

American Sign Language Alphabet Interpreter

By

Nicholas DeNardo

Michael Genovese

Timothy Wong

Final Report for ECE 445, Senior Design, Spring 2017

TA: Jacob Bryan

3 May 2017

Project No. 3

Abstract

We engineered a wearable, American Sign Language alphabet interpretation system to aid the Deaf and hard-of-hearing in their interactions with those who are not fluent in American Sign Language. We describe the design of subsystems which capture the state of the users hand, interpret the letter being signed, announce the letter being signed, and power the entire system. In the end, our system achieves greater than ninety-five percent classification accuracy but is not suitable for all day, portable use.

Contents

1	Introduction	1
1.1	Objective	1
1.2	High Level Requirements.	1
2	Design.	2
2.1	Sensors Module.	2
2.1.1	Flex Sensors	3
2.1.2	Contact Sensors	7
2.1.3	Accelerometers.	11
2.2	Microcontroller	12
2.2.1	Recognition Algorithm	12
2.3	Power.	13
2.3.1	Battery Pack	13
2.3.2	Power Converter	14
2.3.3	Power Consumption	14
2.4	Output.	15
2.4.1	Speaker Module.	15
2.4.2	Status LEDs	17
3	Cost	18
4	Conclusion.	19
4.1	Accomplishments	19
4.2	Future work	19
4.3	Ethical considerations	19
	Reference	21
	Appendix A Flex Sensor Cost Analysis	22
	Appendix B Requirement and Verification Table.	23
	Appendix C Supporting Diagrams	26

1 Introduction

1.1 Objective

As it stands, there exists a communication barrier between those who can use spoken language and those who cannot use spoken language. Many of those in the latter group who reside in North America rely on American Sign Language (ASL) to communicate due to their inability to effectively produce or process audible language. Yet, relatively few members of the hearing community possess the ability to interpret American Sign Language [1]. Acknowledgement of this inability enlightens us to the fact that an effective sign language translation system could allow hearing and non-hearing individuals to communicate effectively with each other.

Despite the magnitude and severity of this issue, relatively little progress has been made toward developing a wearable sign language interpretation system. The most noteworthy effort comes from two University of Washington students who developed SignAloud, a glove-based system that translates American Sign Language into audible English words [2]. Despite this products award-winning functionality and performance, it fails to abolish the communication barrier between hearing non-hearing individuals due to its lack of portability. Because this system relies on a nearby computer to perform the interpretation process, this product cannot be widely deployed for everyday use. A truly practical system would provide the accurate interpretation of SignAloud without the aid of external, cumbersome hardware.

We endeavored to take the first step toward effortless communication between the hearing and the deaf by developing a wearable device that audibly translates the ASL alphabet into the English alphabet. While ASL contains signs for many words, there are many instances in which fingerspelling, the process of spelling a word using the twenty-six letters of the ASL alphabet, must be employed to convey meaning. Furthermore, because fingerspelling relies on only twenty-six signs that involve only the right hand, constructing a system that is capable of interpreting only the ASL alphabet is must more tractable than constructing a system that is capable of interpreting all of American Sign Language. Yet, by developing a wearable device capable of audibly translating the ASL alphabet into the English alphabet, we lay the foundation for a practical American Sign Language translation system.

To do so, we engineered the hardware to describe the state of the user's hand, to decipher the letter being signed, to translate the letter into audible English, and to keep the system running all day long. These four tasks were accomplished by the sensors, microcontroller, output, and power supply modules which are described below. In the end, our system was able to achieve our goal of interpreting the ASL alphabet with greater than 95% correctness, though it was not suitable for portable, all day use.

1.2 High Level Requirements

- The system must audibly translate all twenty-six letters of the American Sign Language alphabet into the English alphabet with ninety-five percent correctness.
- The system must be able to function properly for six continuous hours without having its batteries charged or replaced.
- The form factor of the system must not significantly impede the users ability to sign.

2 Design

The interpreter system can be divided into the four main components shown in the block diagram of Figure 1. Paramount to overall system function is the sensors module which contains the hardware needed to describe the state of the user's hand. The sensors module functions by collecting data from contact sensors which detect when fingers are in contact with one another, accelerometers which determine the orientations of the user's hand and index finger, and flex sensors which describe each finger's level of articulation. Data from the sensors module is sent to the microcontroller via digital input ports, a serial peripheral interface bus (SPI), and an inter-integrated circuit (I^2C). After processing the data from the sensors module, the microcontroller transmits an interpretation over SPI to the output module. The output module will announce the letter being signed over a speaker and will illuminate LEDs to inform the user that the system is operational and that a letter has been announced. Additionally, the power module supplies 5V and 3.3V to the sensors and output modules and 3.3V to the microcontroller unit.

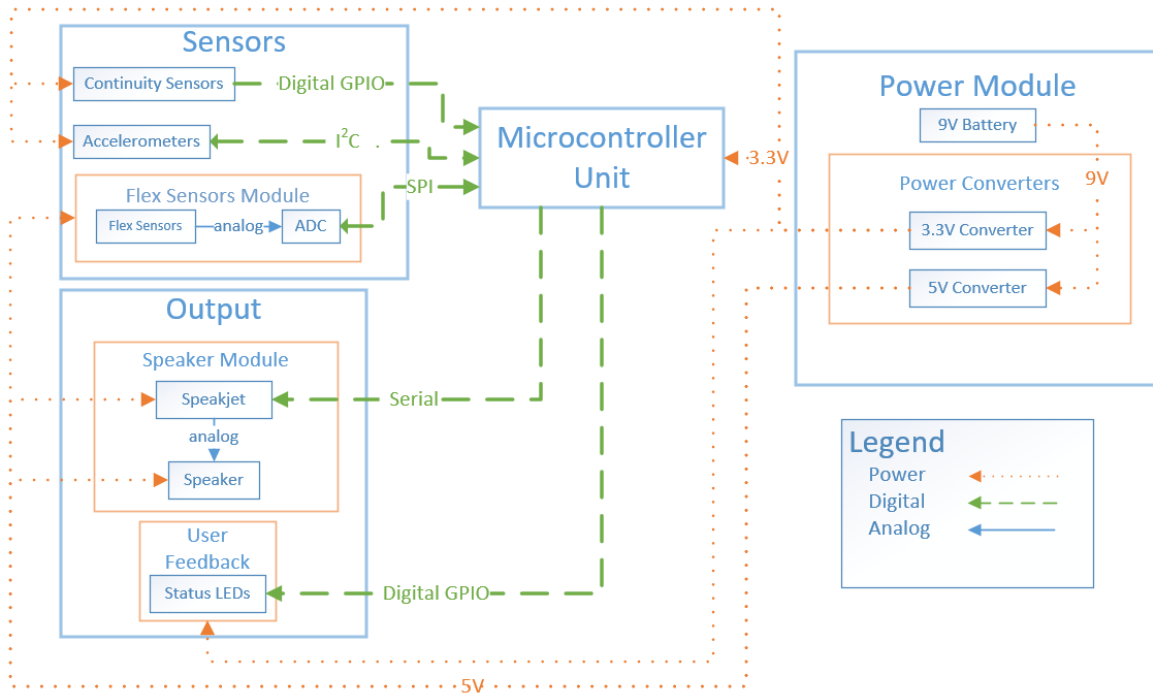


Figure 1: System Level Block Diagram.

2.1 Sensors Module

The purpose of the sensors subsystem is to capture information on the state of the users hand so that the microcontroller may determine the letter being signed. The information that this system seeks to capture is the extent to which each of the users fingers are bent, how the fingers are making contact with one another, and how both the middle digit of the users index finger and the palm of the users hand are oriented. This information is captured by the flex sensors, contact sensors, and accelerometers submodules, respectively, which are integrated in the printed circuit board (PCB) shown in Figure 17 in Appendix C. The sensors module receives 5V and 3.3V supplies from the power module and transmits the data it collects to the

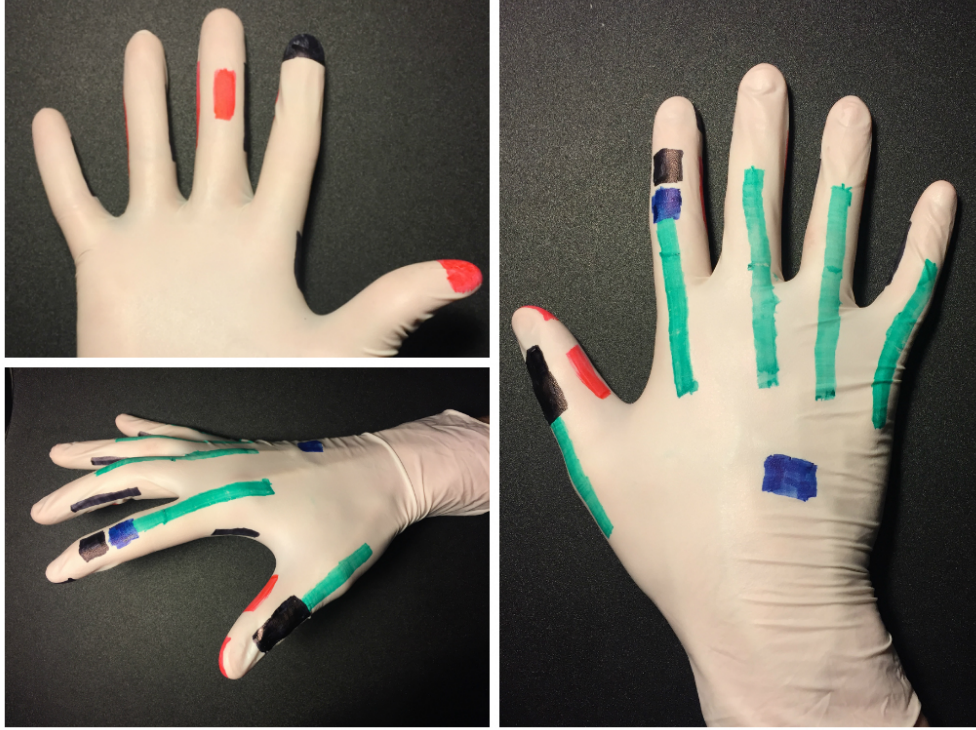


Figure 2: Sensor Configuration. The green stripes represent flex sensors, the blue squares denote accelerometers, and the red and black rectangles represent the high and low sides of contact sensors, respectively.

microcontroller for processing over digital input ports, SPI, and I²C. Accurate and precise data collection and transmission by the sensor subsystem is critical to the success of the overall system because the system’s ability to interpret the letter being signed is inherently linked to the quantity and quality of the information it gathers on the state of the user’s hand.

In the design phase, we required that the sensors module produce a unique feature vector for each letter so that the system could interpret all twenty-six letters with ninety-five percent accuracy 70 mW of power. The system’s accurate classification evidences that the sensors module produces a unique feature vector for each letter, though it has also been shown that on average the multidimensional vector space of sensor outputs with three flex sensor states and three accelerometer states has a Hamming distance of 2. For these reasons, we can verify that the sensors module satisfies the design requirements.

2.1.1 Flex Sensors

The flex sensors subsystem provides the microprocessor with data on the extent to which each of the users fingers and thumb are bent. This information is collected by measuring the output voltages of voltage divider circuits formed between flex sensors and resistors using an analog-to-digital converter (ADC). Each output voltage is described by Equation 1 where V_{OUT} is the output voltage, V_{IN} is the input voltage, R_{flex} is the resistance of the flex sensor, and R_{OUT} is the resistance of the output resistor. The information that this subsystem provides aids the microprocessor in determining which letter is being signed by producing voltages characteristic of only a certain subset of letters. The schematic for this circuit is shown in Figure 18. in Appendix C

$$V_{OUT} = \frac{R_{OUT}}{R_{flex} + R_{OUT}} V_{IN} \quad (1)$$

Initially, we sought to implement the flex sensors module using piezoelectric flex sensors manufactured by Spectra. During preliminary testing, we found that these sensors had an unsatisfactorily small resistance range of less than $100\text{ k}\Omega$ and slow settling times as can be seen in Figure 3. To correct these issues, we manufactured our own optical flex sensors. These devices operate on the principles depicted in Figure 4 and consist of a blue, 3mm light-emitting diode (LED), a poor waveguide, and a cadmium sulfide GM5539 photoresistor. When there is little bending in the waveguide, most of the light reaches the photoresistor and its resistance is low. As the bending increases, less light reaches the photoresistor and its resistance increases. The performance of these flex sensors is characterized in Figure 5. Figure 6 displays the improved response time in this new device. In the end, our in-house design reduced the response time significantly and increased the resistance range by a factor of 1000.

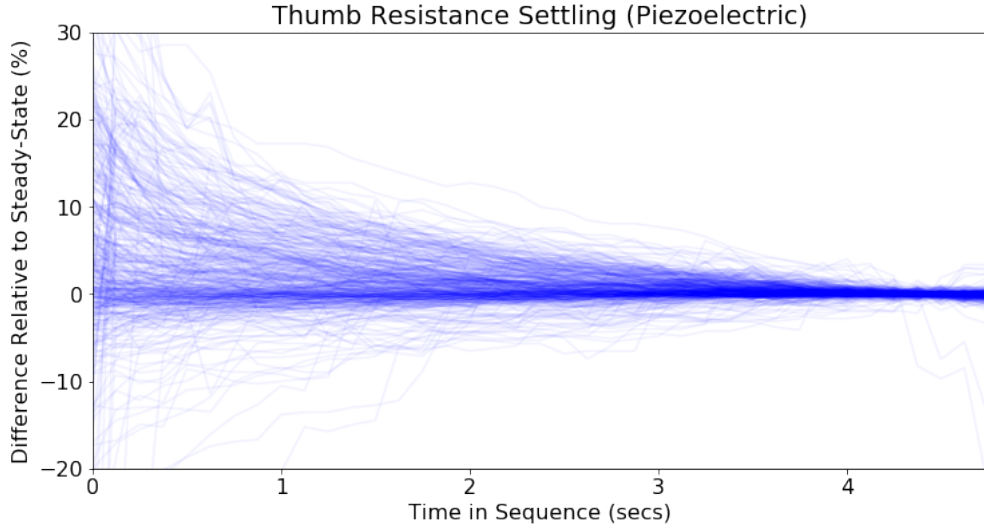


Figure 3: Resistance response of piezoelectric flex sensor.

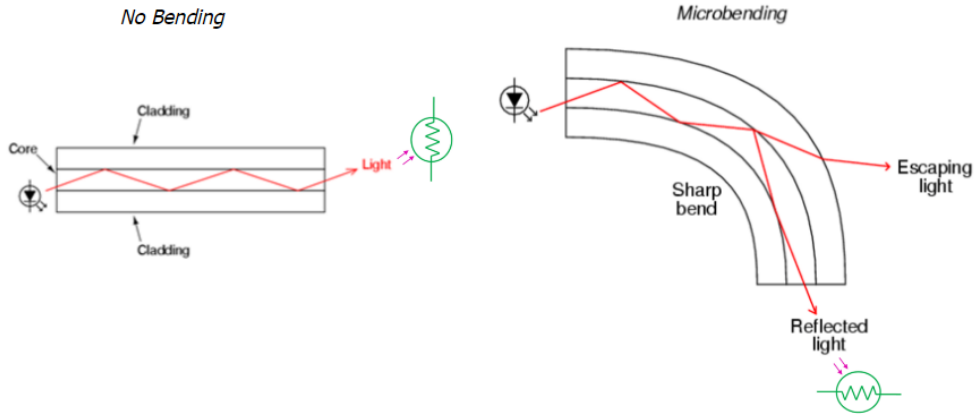


Figure 4: Principle of operation of optical flex sensors [3].

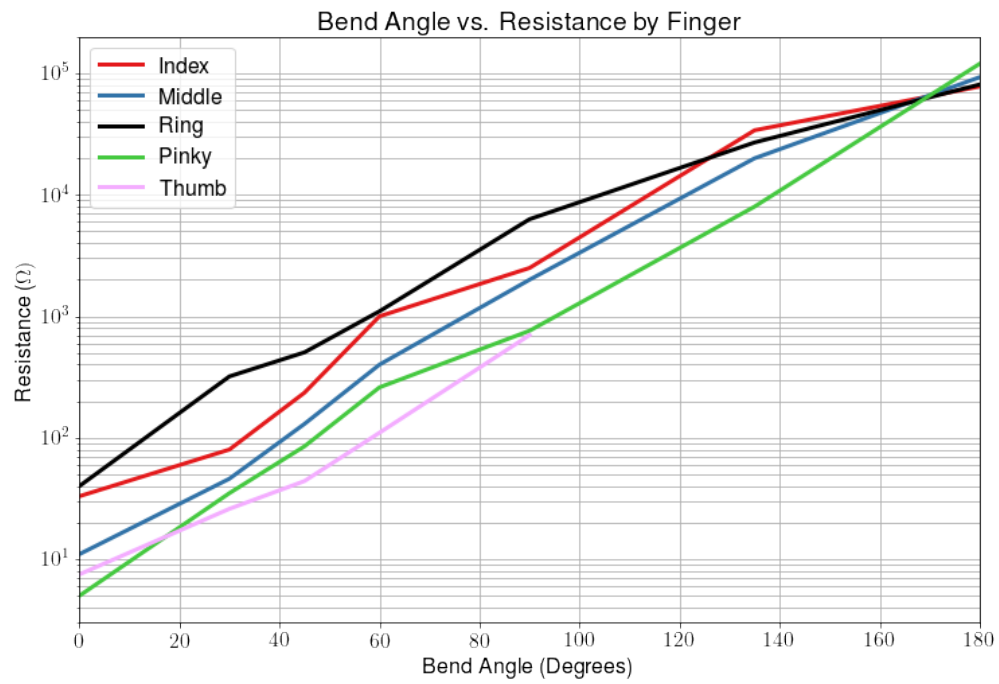


Figure 5: Performance of optical flex sensors.

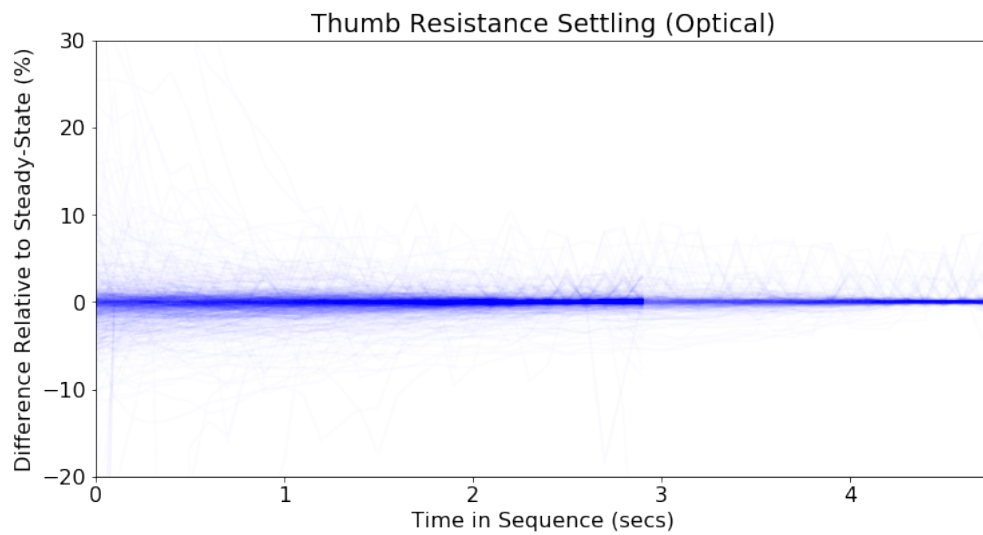


Figure 6: Resistance response of optical flex sensor.

Moreover, our custom flex sensors provide the core functionality of the circuit by varying in resistance as they bend with a digit. Specifically, the resistance in a flex sensor adhered to a digit increases as that digit becomes more jointed. Figure 5 shows that the resistance of the flex sensor can become as large as 100 M Ω at high levels of bending. This fact led us to select the flex sensor as the dividing resistor and not as the output resistor in the voltage divider circuit because its resistance can be significant when compared with the input impedance of the MCP3008 ADC at high levels of bending [4].

Additionally, to convert the resistance of the flex sensor into a voltage, we employed a voltage divider circuit. While other topologies exist for producing voltage measurements of resistances, alternatives such as common source amplifiers would increase the cost and design complexity of the flex sensor subsystem without providing substantial improvements in performance. For this reason, the voltage divider circuit was selected over all other alternatives for converting the flex sensor resistance into a voltage for the MCU.

In designing the voltage divider circuit, we first faced the decisions of selecting an input voltage, V_{IN} . To find a suitable input voltage, we first took the partial derivative of Equation 1 with respect to R_{flex} , as shown in Equation 2. This equation shows us that the magnitude of the change in output voltage for a given change in flex sensor resistance increases with increasing input voltage. For this reason, we desired a large input voltage. However, the MCP3008 ADC is only rated to receive input voltages up to 5V. For this reason, we set the input voltage to the voltage divider circuit, V_{IN} , to be 5V. This decision guaranteed that the output of the voltage divider would be less than 5V and that the ADC would operate reliably.

$$\frac{\partial V_{OUT}}{\partial R_{flex}} = \frac{-R_{OUT}}{(R_{flex} + R_{OUT})^2} V_{IN} \quad (2)$$

With the input voltage for each voltage divider set, we set about determining suitable output resistors for the voltage divider for each flex sensor. The objective was to make the flex sensor's resistance with little bending appear small when compared to the output resistance and to make the flex sensor's resistance with high bending appear large when compared to the output resistance. To do so, we sought to match the output resistance of the voltage dividers for the pinky, ring, middle, and index fingers with the resistance of each respective flex sensor when the user formed the letter 'e'. Doing so made the output resistor a suitable comparison to the flex sensor because the letter 'e' produces moderate finger bending for the pinky, ring, middle, and index fingers. To determine a suitable dividing resistance for the thumb, we took the average resistance of the thumb flex sensor when the user formed the letters 'a' and 'b'. These letters represent states of high and low thumb bending, and so the average of thumb flex sensor resistance in these states provides a suitable output resistor value.

Figures 20 through 24 in Appendix C show that the voltages produced for different letters may vary by hundreds of millivolts. For this reason, we knew that we needed an ADC capable of detecting voltage variations on the order of ten millivolts. Equation 3 defines the voltage resolution, V_{RES} , of an ADC where the reference voltage, V_{REF} is $V_{IN} = 5V$. When we solve this equation for N , the number of bits the ADC uses to represent the voltage, in Equation 4, we find that we need an ADC that represents voltages with at least nine bits. The MCP3008 from Microchip provides us with the ten bits of resolution that we need as well as a multiplexing capability to support all five flex sensor voltage divider circuits. Additionally, the MCP3008 has an SPI serial interface which is supported by the LPC11U37 microprocessor incorporated in this system. For these reasons, we determined that the MCP3008 was a suitable choice for the ADC of the

flex sensor module.

$$\frac{V_{REF}}{2^N} = V_{RES} \quad (3)$$

$$N = \log_2\left(\frac{V_{REF}}{V_{RES}}\right) = \log_2\left(\frac{5V}{0.010V}\right) = 8.96 < 10 \quad (4)$$

To verify that the flex sensors module functions properly, we ensured that the ADC communicated output voltage values to the microcontroller that differed by more than 5 mV, the resolution of the ADC, for the high, moderate, and low resistance states. Because this proved to be the case, we verified that the flex sensors module functions correctly.

In the end, this module receives a 5 V supply from the power module and communicates how bent the users fingers are to the microprocessor over an SPI interface. This subsystem is critical to the success of the overall system because without reliable data on the extent to which each of the users fingers are bent each letter of the ASL alphabet would not be able to be uniquely identified with ninety-five percent correctness.

2.1.2 Contact Sensors

To enable the system to determine how the users fingers are grouped together, we strategically placed contact sensors across the glove as can be seen in the physical diagram of Figure 2. These sensors perform several specific functions. Firstly, these sensors determine if adjacent fingers are in lateral contact with one another. Secondly, these sensors determine if the tip of the thumb is in contact with the inside of the pinky finger, with the inside of the ring finger, or with the tip of the index finger. Finally, these sensors determine if the underside of the middle finger is in contact with the top of the index finger or with the top of the thumb. In doing so, the contact sensors serve a critical role by gathering the information that allows the system to differentiate between similar signs. Without these sensors and the information they provide, the system would not be able to translate the letters of the ASL alphabet into the English alphabet with ninety-five percent correctness. Furthermore, these sensors receive a 3.3V supply from the power module and transmit the information they collect to the microprocessor across a digital port.

Each contact sensor was constructed by sewing patches of conductive fabric at the locations shown in Figure 2. The patches represented by red rectangles in Figure 2 are connected to a 3.3V rail from the power module through a 200 k Ω resistor. These patches serve as the high sides of the contact sensors. The patches represented by black rectangles in Figure 2 are connected to ground. These patches will be referred to as the low sides of the contact sensors. Figure 7 shows the circuit model for each contact sensor. As shown in Figure 7, we represent contact between the two sides of the contact sensor by a single-pole, single-throw (SPST) switch. When contact is made between the two sides of a contact sensor, the switch is closed and the output is connected to ground. Conversely, when the high side of a contact sensor is not in contact with a patch connected to ground, the circuit is effectively open in steady-state. As a result, the voltage at the output node will be near 3.3V.

In practice, opening the switch in Figure 7 will not open the circuit because the input impedance of the microprocessor pins is not infinite. According to the data sheet for the ARM LPC11u37 microprocessor, the input impedance of the digital pins is 2.5 M Ω . Thus, the output of the contact sensor is the output

of a voltage divider with the $200\text{ k}\Omega$ resistor serving as the dividing resistor and the input pin serving as the output resistor. Additionally, the data sheet for the microprocessor also states that logical high inputs are those inputs whose voltages rise above 2.31 V [5]. To ensure that the voltages produced by the contact sensors when no contact is made are always interpreted as logically high, we require that voltage rise above 3 V . From this requirement, we knew that we needed a resistor between the 3.3 V supply and the high side of the contact sensor that would force the output voltage above 3 V when the switch in Figure 7 is open. By applying Equation 5 and solving for R , we found that we needed a resistance below $250\text{ k}\Omega$. Realizing this constraint, we sought to maximize the resistance to conserve power, provide a reasonable resistance margin to ensure correct functionality, and minimize the component cost. In response, we decided to implement the resistor using a $\$0.03$ Panasonic $250\text{ k}\Omega$ thick film resistor with a 1% tolerance and a 250 mW power rating in a 1206 surface mount package [6].

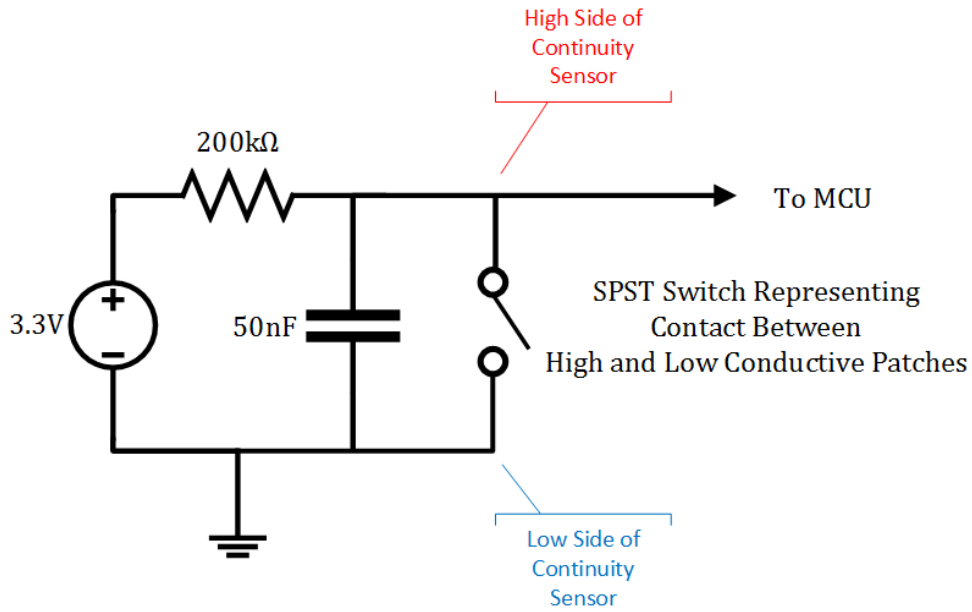


Figure 7: Contact Sensor Circuit Model

$$V_{OUT} = \frac{R_{MCU}}{R + R_{MCU}} V_{IN} \quad (5)$$

Furthermore, because the contact sensors resemble mechanical switches, the signals they produce require debouncing. Figure 8 shows a signal produced by a contact sensor during a low-to-high transition with the filtering capacitor removed. This waveform shows that the bouncing occurs during the first 10 ms after initial contact is made between the two sides of the contact sensors. As a result, we concluded that the bouncing occurs at a frequency above 100 Hz . Consequently, we knew that we could attenuate the bouncing during the low-to-high transition by incorporating a low-pass filter with a 3 dB bandwidth of 100 Hz . To do so, we applied Equation 6 which defines the 3 dB bandwidth for a low pass filter, solved for the capacitance, C , and placed a 50 nF capacitor across the two sides of the contact sensor. Figure 9 displays the signal produced by a contact sensor with this filtering capacitor in place. Clearly, we see that this capacitor severely attenuates the amplitude of the bouncing. Additionally, Figure 10 and Figure 11 show that the filtering capacitor also attenuates bouncing during the high-to-low transition, though not as severely. Some bouncing remains

during the high-to-low transition because the signal generated by the contacts during this transition is not truly low-pass filtered but is merely slowed by the discharging of the capacitor through the parasitic resistance in the wires and contacts. Though not ideal, the attenuated bouncing that remained during the high-to-low transition did not affect the functionality of the system.

$$f_{3dB} = \frac{1}{RC} \quad (6)$$

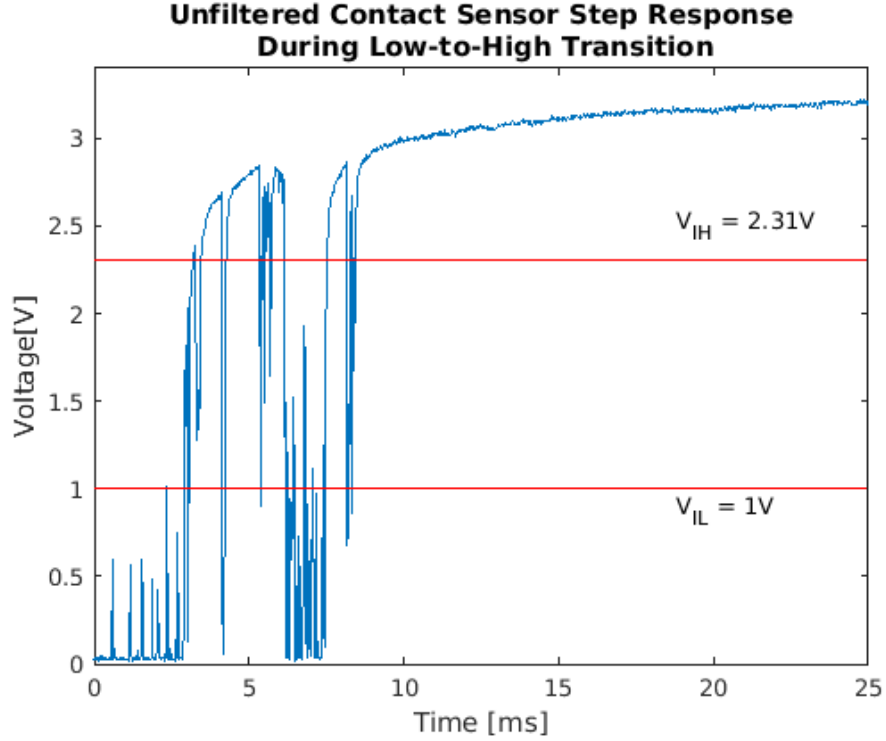


Figure 8: Unfiltered contact sensor step response during low-to-high transition.

To verify this module, we needed to ensure that the microcontroller received the appropriate logic signals according to the state of each contact sensor and that each contact sensor consumed less than 1.0 mW. Because we were able to observe the correct logic values with no bouncing read back by the microcontroller over a serial interface, we were able to verify that this module produces and transmits the desired data. Additionally, by measuring the current drawn from the 3.3 V by each contact sensor, we were able to determine that each contact sensor consumes an average of 54.6 μ W. With these two conditions verified, we can confirm that this module satisfies the design requirement.

All told, this module receives a 3.3 V supply from the power module and communicates how the user's fingers are in contact with one another to the microprocessor over digital ports. This subsystem is critical to the success of the overall system because without reliable data on how the user's fingers are grouped together each letter of the ASL alphabet would not be able to be uniquely identified with ninety-five percent correctness.

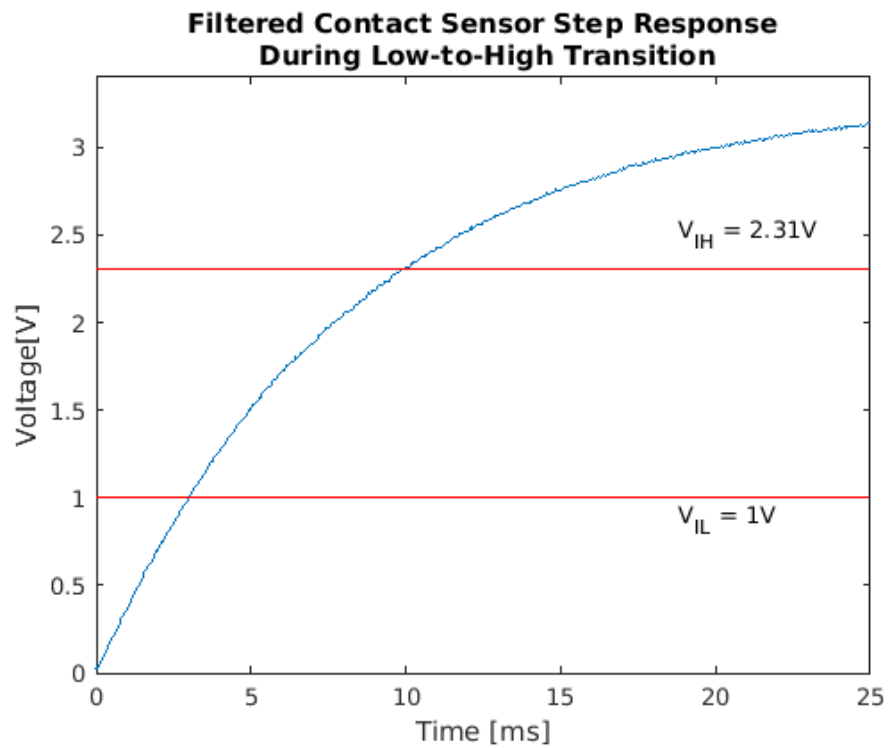


Figure 9: Filtered contact sensor waveform during low-to-high transition.

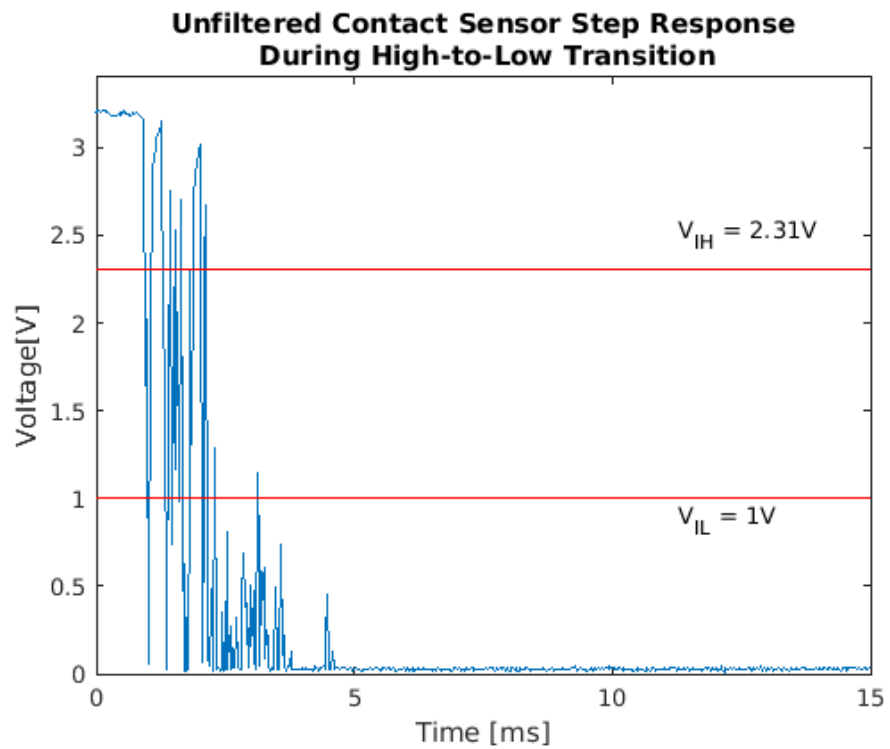


Figure 10: Unfiltered contact sensor step response during high-to-low transition.

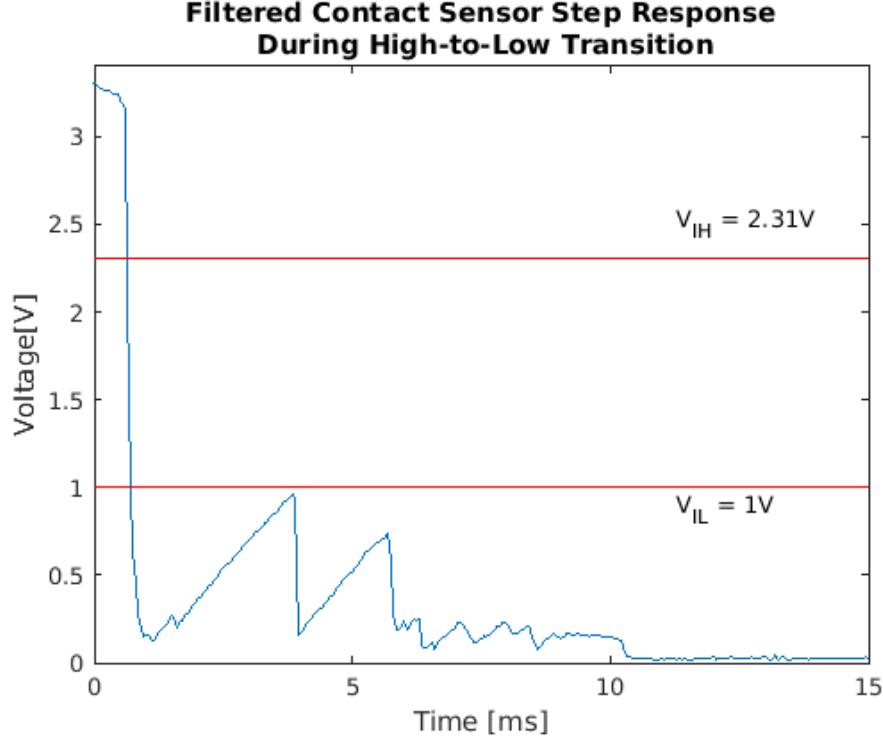


Figure 11: Filtered contact sensor waveform during high-to-low transition.

2.1.3 Accelerometers

The purpose of the accelerometers module is to detect both the orientation of the palm and the orientation of the middle digit of the index finger with respect to the surface of the earth. This information allows the system to differentiate between signs which are rotations of one another and to determine when the hand is in motion. Without these accelerometers, the system would not be able to differentiate between three sets of two letters. Thus, the system would be incapable of meeting the requirement that it translate all twenty-six letters of the ASL alphabet into the English alphabet with ninety-five percent correctness. The placement and physical implementation of the accelerometers can be seen in Figures 2 and 16.

To implement the accelerometers, we employed the MPU6050 3-axis accelerometer from InvenSense. The MPU6050 offers the competitive cost of many other accelerometers with the added benefit of a 3-axis gyroscope that can be used to filter the accelerometer data and provide a more accurate description of the orientation of the hand. Additionally, the MPU6050 also contains an on-board ADC that allows it to communicate with the microcontroller over an I²C interface. This feature simplifies the implementation of the accelerometers module and prevents us from exhausting all the analog inputs of the microcontroller and having to purchase additional ADCs. The interface for the MPU6050 breakout board that we used is shown in Figure 25.

To verify proper functionality of the accelerometers, we needed to observe the standard deviation in the accelerometer's acceleration measurements along each axis when at rest and ensure that each is below 0.5g. This test produced values of 0.005g, 0.003g, and 0.005g for the standard deviation in the acceleration along the x, y, and z-axes, respectively. Because each of these is well below 0.5g, we can conclude that the

accelerometers module meets the requirement.

2.2 Microcontroller

For this project the LPC11U37 was used. this chip was chosen since its 32 bit processor and IDE supports C programs, it supports I2C, SPI, and UART communication protocols, and has plenty of GPIO pins. The SPI ports were used for communication with the MCP3008 ADC. The I²C ports were used for communication with the MPU6050 accelerometer. The UART ports were used to communicate with the speaker module. Continuity sensors were connected to the microcontroller via the GPIO pins. Figure 19 shows the schematic for the microcontroller unit.

Table 1: Microcontroller Features

Feature	Value
Chip	LPC11U37
Architecture	ARM Cortex-M0
Input Voltage	1.8 V - 3.6 V
Current Draw	7 mA
Software IDE	LCPXpresso
Processor Speed	12 MHz - 50 MHz
Memory	10 kB SRAM
Flash	128 kB
GPIO Pins	40
Communication Protocol	I ² C, SPI, UART

2.2.1 Recognition Algorithm

The recognition algorithm is primarily based on the use of a Support Vector Machine (SVM) to classify sensor inputs as one of 27 states. Our system recognizes each of the 26 letters of the alphabet as well as an additional ‘no-guess’ state. Sample raw sensor data can be found in Table 2.

Table 2: Sample raw sensor data

Flex Sensor						Accelerometer						Continuity Sensor					
Label	<i>Index</i>	<i>Middle</i>	<i>Ring</i>	<i>Pinky</i>	<i>Thumb</i>	<i>Index X</i>	<i>Index Y</i>	<i>Index Z</i>	<i>Hand X</i>	<i>Hand Y</i>	<i>Hand Z</i>	<i>InterI</i>	<i>InterM</i>	<i>InterR</i>	<i>InterT</i>	<i>TipT</i>	<i>SuperM</i>
A	163	27	80	127	554	-0.34	0.95	0.23	0.16	-0.90	0.44	0	0	0	1	1	1
B	844	682	785	885	369	-0.09	-0.97	0.31	-0.11	-0.99	0.07	0	0	0	1	1	1
C	775	550	663	819	619	-0.96	-0.05	0.22	-0.04	0.79	0.60	0	0	0	1	1	1

An SVM is a relatively simple supervised machine learning algorithm. From initial training data, a set of support vectors are calculated. The computed support vectors act as simple classifiers by linearly dividing the state space in two regions, ‘in-class’ and ‘out-class’. The ‘in-class’ region represents an area of the state space that likely belongs to the specific letter, conversely the ‘out-class’ region represents an area which is unlikely to be the specific letter. Unseen data is then classified by evaluating which region of the state space the input data point lies in. Every support vector acts as a vote for the current recognized class. The class with the most total votes represents the most likely recognized letter.

To increase the linear separability of the input data, we preprocess the raw sensor data using Linear Discrim-

inant Analysis (LDA). The goal of LDA is to the minimize in-class scattering, while maximizing out-class scattering. That is, data points that are members of the same class should be tightly clustered while members of other classes should be spread out as far as possible. Based on the same data used to train the SVM, an optimal transformation matrix is computed. This transformation matrix projects the input data into a new, more easily separable space.

2.3 Power

Two of the requirements of the ASL interpreter are that the glove needs to be portable and last for a full day's use. Due to these requirements the power module needs to supply power from the battery and convert the voltage to another level efficiently.

2.3.1 Battery Pack

A battery pack for this project was designed to power sensor module, microcontroller, and speaker module. The battery pack consist of two PP3 9 V battery holders. A PP3 sized 9 V alkaline battery was chosen as a power source due to its wide availability, minimal safety risk compared to lithium polymer battery, and mAh rating. The redundancy in the number of battery holders allows the user to swap in an additional battery without having to turn the glove off, or use two batteries to greatly extend the operating time of the glove.

Figure 12 shows the measurements for the battery voltage drop when the glove is on. The battery voltage after 4 hours is at 7.68 V which is above the 6.5 V voltage cutoff for our converters.

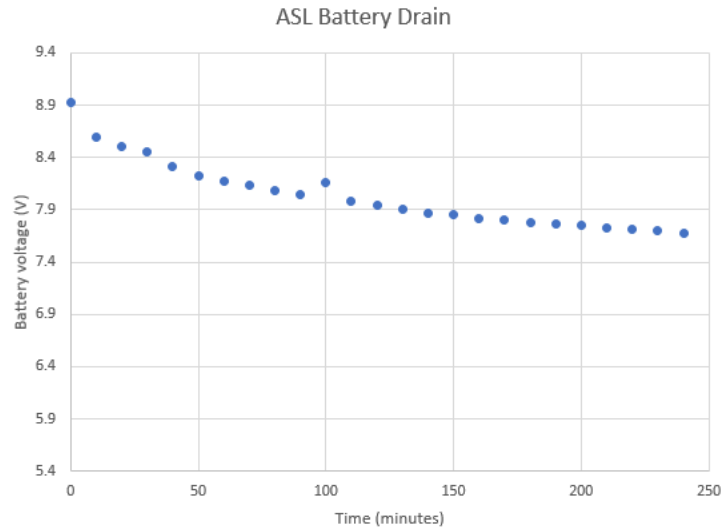


Figure 12: Graph of battery pack voltage over time

Table 3: Battery Features

Features	Value
Battery	9 V
Supply Voltage range	10.5 V -5.4 V
Package size	PP3
Rated Capacity	500 mAh

2.3.2 Power Converter

The power converter for the interpreter supplies 5 V for the output module and accelerometers, and 3.3 V for the microcontroller, flex sensors, and continuity sensors. two buck converters were used to step down the voltage from 9 V to 3.3 v and 5 V. The TPS563208 was selected for its high switching frequency, adjustable feedback network for controlling the output voltage, and soft startup feature. Figure 26 in the appendix shows the schematic for the 3.3 V and 5 V power supply using the TPS563208. The LM317 was left out in the final design. Resistance values for the feedback network were chosen based equation 7. The circuit for the TPS563208 is based on the application circuit provided [7].

$$V_{out} = 0.768 \times \left(1 + \frac{R_1}{R_2}\right) \quad (7)$$

Table 4: TPS563208 Features

Features	Value
Chip	TPS563208
Input voltage range	4.5 V to 17 V
Output voltage range	0.76 V - 7 V
Switching frequency	580 kHz
Output current	3.0 A

Figure 27 in the appendix shows the waveform of the output from the 5 V converter module. The measured ripple ignoring the switching noise is 10 mV peak to peak. The switching noise has a peak to peak ripple of 334 mV. Thus the voltage maximum at the output is 5.27 V, which is below the 5.5 V absolute voltage maximum for the components in the system [4]. Figure 28 in the appendix shows the output voltage waveform of the 3.3 V converter. The voltage ripple ignoring the switching noise is 9.25 mV. The switching noise ripple is 243 mV. Thus the voltage maximum from the converter is 3.47 V, which is below the 3.6 mV Voltage measurement for the microcontroller [5].

2.3.3 Power Consumption

Table 5: Module Power Consumption

Component	Quantity	Voltage (V)	Current (A)	Power (W)
LPC11U37	1	3.3	7	0.0198
Continuity sensor	6	3.3	0.15	0.00297
Accelerometer	2	5	3.9	0.039
Speaker module (inactive)	1	5	25	0.125
Speaker module (active)	1	5	425	2.125
Flex sensor module	5	3.3	10	0.165
Status LED	1	3.3	10	0.033

Table 5 has the power consumption of each of the modules. In the inactive state, where the speaker is not being used, the power consumption is measured to be 0.385 W. When the speaker is being used the power consumption is 2.385 W.

Figure 13 is a oscilloscope measurement of the current and current when the speaker produces sound. The test configuration for this measurement was a separate power supply for the speaker module, and current

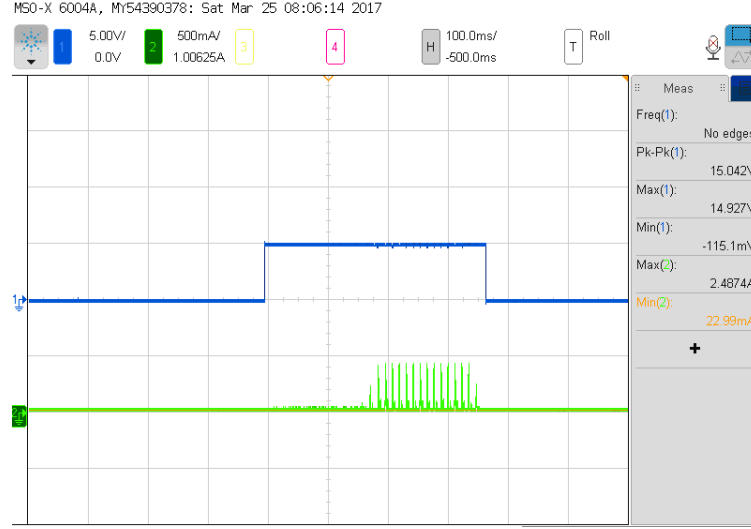


Figure 13: The yellow waveform is the voltage and the green trace is the current waveform. Note the oscilloscope measurements on the gray side panel are incorrect.

probe and voltage on the input to the speaker module. The peak sound pressure measurement during the test was 60 dB. The waveform shows that the speaker module operates in the inactive state a majority of the time with the exception of a few milliseconds. Thus the impact of the large power consumption of the speaker is not as significant as the active power consumption suggest.

2.4 Output

The purpose of the output module is to provide audible and visual feedback to the listener and user of the ASL alphabet glove. The output module consist of two parts, the speaker module which will pronounce a letter, and status LEDs to let the user know the device is on and when the speaker module is pronouncing a letter.

2.4.1 Speaker Module

The purpose of the speaker module is to communicate with the microcontroller to produce a sound. The approach to tackling this problem was inspired by the speech synthesis chip in the Speak and Spell. This approach allows the speaker module to be compact, have lower performance requirement, and be easier to implement at little cost to sound quality.

The speaker module with the microcontroller over UART to receive the letter to be pronounced. The data sent goes to the Cypress CY8C29466 microcontroller to convert the serial input to 8 bit output for the speakjet. Based on the 8 bit input, the speakjet will synthesize sound. The sound output of the speakjet is then goes through a low pass filter and amplifier to bring the output to an audible level.

Figure 15 shows the waveform of the output of the speakjet on the top half, and the Fast Fourier Transform of the waveform on the bottom half. From the FFT of the speakjet when a letter is being pronounced, most of the audible sound from the speakjet chip occurs at frequencies greater than 500 Hz. A low pass filter was designed to dampen the higher frequencies. The equation 8 was used to determine the resistance value for the low pass filter.

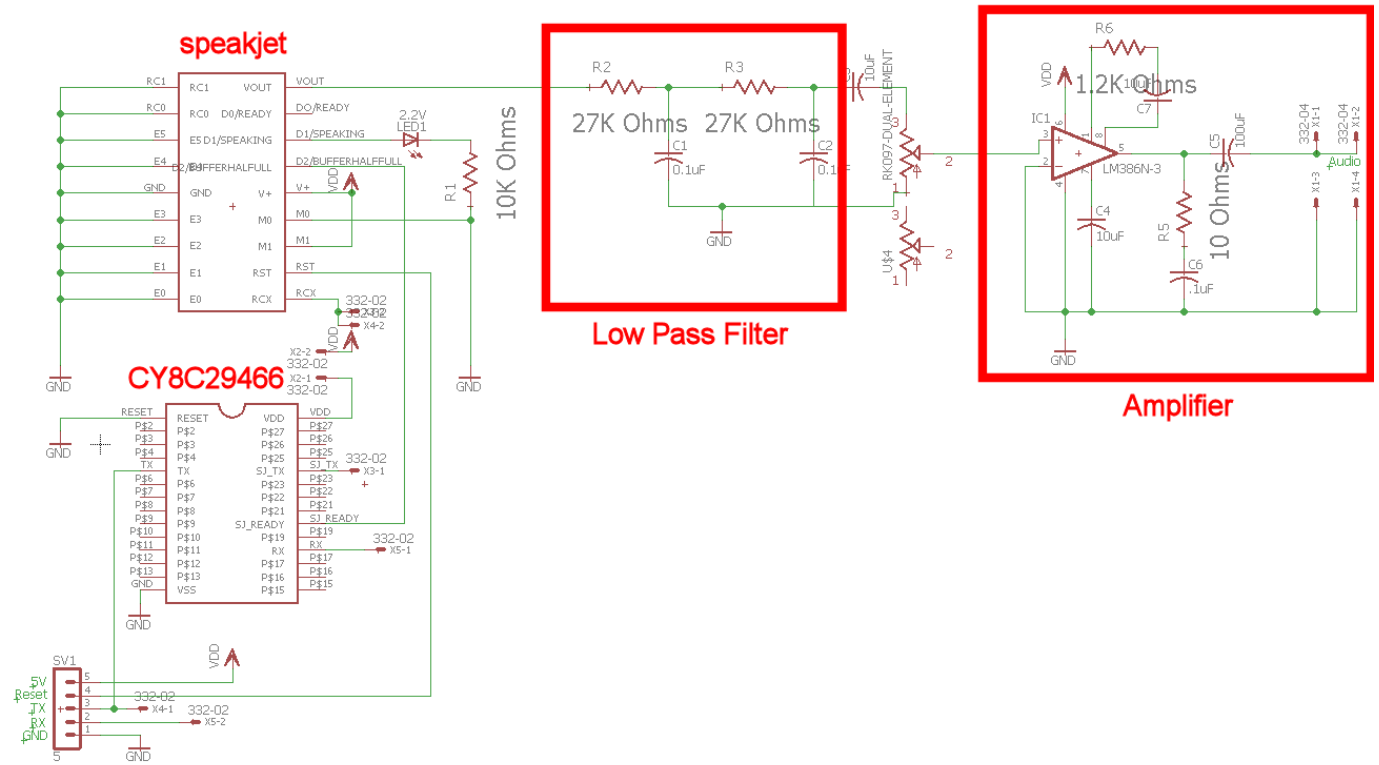


Figure 14: Schematic for the speaker module

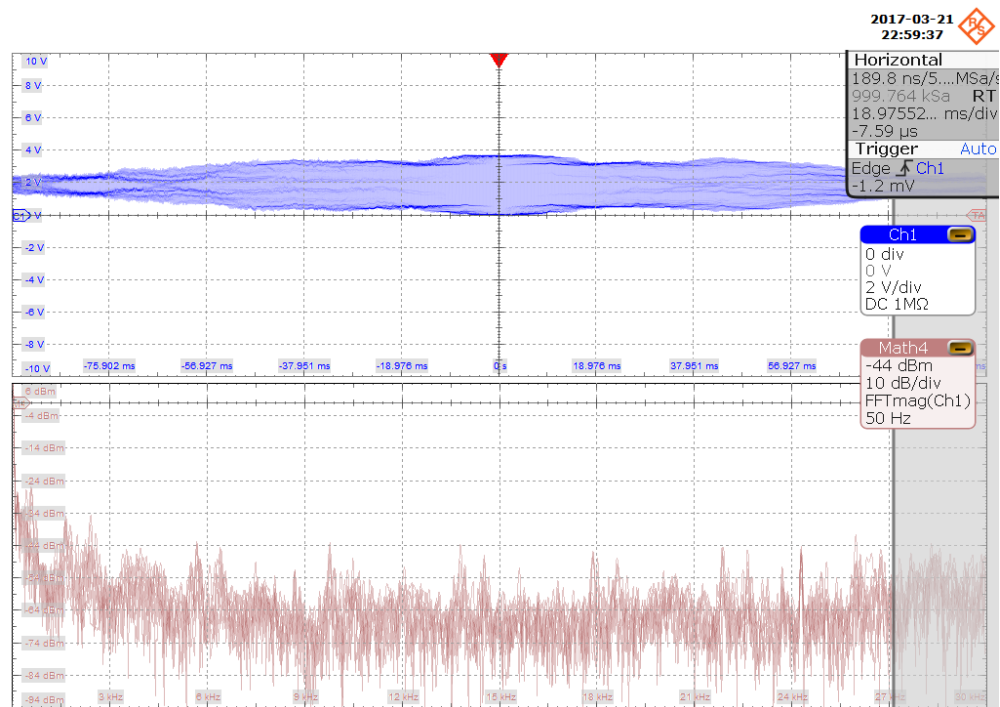


Figure 15: Fast fourier transform of the output of the speakjet

$$f_c = \frac{1}{2\pi RC} \quad (8)$$

The LM386n-3 was chosen for the amplifier circuit for its less noisy output at higher gains compared to the NE553 amplifier. The amplifier design used was based on the application circuit provided [8]. The gain for the amplifier can be selected by changing the value of the resistor and the capacitor across pins 1 and 8. Equation 9 is the formula for determining the amplifier gain.

For the glove to be usable, the output volume needs to be greater than 50 dB. The sound pressure level (SPL) from the speaker module was measured one meter from the speaker with a decibel meter. The measured SPL was 68.6 dB which exceeds the requirement.

$$A = \frac{30000}{150 + \frac{1350 \times R}{1350 + R}} \quad (9)$$

2.4.2 Status LEDs

Status LEDs were used to provide feedback to the user. Resistance values were chosen for the status LEDs to be visible from 1 meter away. Device operations we decided to give LED indicator for are an indicator for when the speaker is pronouncing a letter and when the device is on.

3 Cost

Itemized costs of all parts and labor can be found in Table 6. While we do not consider the project as a whole commercially viable, our optical flex sensor may be viable. A detailed flex sensor cost analysis can be found in Appendix A.

Table 6: Total costs by parts and labor

PARTS				
Part Name	Distributor	Retail Cost	Quantity	Total
Buck Converter	Mouser	\$ 3.17	2	\$ 6.34
9v Alkaline Battery	Mouser	\$ 2.37	1	\$ 2.37
Various Passive Elements	DigiKey	\$ 5.00	1	\$ 5.00
PCBs	PCB Way	\$ 0.50	10	\$ 5.00
Power Module				\$ 18.71
Microcontroller	Chip1Stop	\$ 4.74	1	\$ 4.74
Various Passive Elements	DigiKey	\$ 5.00	1	\$ 5.00
PCBs	PCB Way	\$ 2.50	10	\$ 25.00
Controller Module				\$ 34.74
Speaker	Sparkfun	\$ 1.95	1	\$ 1.95
SpeakJet	Sparkfun	\$ 24.95	1	\$ 24.95
Various Passive Elements	DigiKey	\$ 5.00	1	\$ 5.00
Output Module				\$ 31.90
Micro Fuel Line Tubing	Home Depot	\$ 4.75	1	\$ 4.75
Accelerometer	Sparkfun	\$ 9.95	2	\$ 19.90
Conductive Thread	Sparkfun	\$ 3.95	1	\$ 3.95
10-bit ADC	Adafruit	\$ 3.75	1	\$ 3.75
Various Passive Elements	DigiKey	\$ 5.00	1	\$ 5.00
PCBs	PCB Way	\$ 0.50	10	\$ 5.00
Sensors Module				\$ 42.35
Glove	Amazon	\$ 12.99	1	\$ 12.99
Miscellaneous				\$ 12.99
PARTS TOTAL				\$ 127.70
LABOR				
Team Member	Hourly Rate	Total Hours	Expense Multiplier	Total Cost
Nick DeNardo	\$ 33.50	160	2.5	\$ 13,400.00
Tim Wong	\$ 33.50	160	2.5	\$ 13,400.00
Mike Genovese	\$ 33.50	160	2.5	\$ 13,400.00
LABOR TOTAL				\$ 40,200.00
GRAND TOTAL				\$ 40,327.70

4 Conclusion

4.1 Accomplishments

Our team has been very successful in the development of a system capable of reliably recognizing all letters of the ASL alphabet. Although we trained and evaluated the system on a single user, we achieved a classification accuracy of 98.7%.

We also developed an improved flex sensor design over available commercial alternatives. Most notably, our flex sensor's operate over a much larger range of resistance than current alternatives. In practice, our sensors exhibited a resistance range larger than commercial alternatives by a factor of 1000. Further, we constructed our design at a fraction of the cost.

4.2 Future work

Our primary goal for future work would be to increase the robustness of our system. We developed and optimized our recognition algorithm for a single user. Although we found our system was still reasonably robust to an entirely new and unseen user, there remains an appreciable decrease in recognition accuracy. Our current system does not take into consideration differences in hand shapes and individual user's fingerspelling form. In the future we would like to assemble a larger, more diverse dataset to prevent our system from overfitting to a single user's fingerspelling idiosyncrasies.

Additionally, due to resource constraints on our selected microcontroller, we unable to store all the required software on the microcontroller. Instead we used a Raspberry Pi to perform all necessary computation. With a larger and more performant microcontroller we could implement all the required software on-board without relying on external processing.

Further, by using the microcontroller we originally developed, we could greatly reduce the required size of the total system. In our designs current form, the system is not practical to wear regularly. By combining all PCBs and prioritizing form factor more, we believe the size and weight of our system could be reduced.

4.3 Ethical considerations

The greatest safety concern in our proposed design is the battery. The relatively large energy density of the batteries makes the battery pack susceptible to various thermal and electrical hazards. Sustained skin exposure to temperatures greater than 48°C can result in third degree burns within 5 minutes [9]. As thermal buildup is an inevitable product of energy storage and discharge, we have designed our battery pack to promote air flow and convective heat dissipation to the environment. To mitigate the risks involved with short circuits, our design includes current limiting circuitry around the battery. Further, we have selected alkaline batteries over more hazardous chemistries like lithium-ion batteries.

As with many wearable technology applications, there is an inherent requirement that numerous electrical components be coupled closely with the users body. The effects of long term exposure of electronics positioned closely along the body is still an area of open research. In a similar vein to the potential negative effects of prolonged cell phone radiation [10], there is the potential that wearable electronics may be shown to have negative effects for long term users. In accordance with #9 of the IEEE Code of Ethics [11], to avoid potential harm to users of our product, it is important the we maintain a cautionary position on any potential adverse

or harmful effects of wearable technology.

In addition to a concern for our users safety, we also consider the safety of development a main priority. As we move through the development process, there are several potential hazards that may not be present in the final product. For example, while in development, a battery will not be used to power the device. This exposes all those working with our prototypes to a larger potential source of voltage and current. Developer and early user safety is an equally important concern to that of end user safety.

As stated in 1.7 of the ACM Code of Ethics [12], "...communication technology enables the collection and exchange of personal information on a scale unprecedented in the history...", therefore it is of notable concern to fully protect the users privacy. A potential compromise in the security of the device could allow a malicious actor to obtain a users entire conversation. It is critical that we preempt such a breach of privacy. As such, our design has prioritized the need for the device to be entirely self-contained. Performing all necessary data processing on board the device and without maintaining any logs, the potential for a data leak is greatly reduced.

We strive to fulfill guidelines set forth by #3 and #7 of the IEEE Code of Ethics by representing all our technical claims honestly and willingly accepting criticism of our work. We will meaningfully credit all those who make contributions to our project.

References

- [1] B. Lane, Hoffmeister, *A journey into the deaf-world*. San Diego, CA: DawnSignPress, 1996.
- [2] U. of Washington, “Uw undergraduate team wins \$10,000 lemelson-mit student prize for gloves that translate sign language,” Apr 2016. [Online]. Available: <http://www.washington.edu/news/2016/04/12/uw-undergraduate-team-wins-10000-lemelson-mit-student-prize-for-gloves-that-translate-sign-language/>
- [3] *Optical data communication*, Webpage, Learning Electronics, accessed: May 1, 2017. [Online]. Available: http://www.learningelectronics.net/vol_4/chpt_14/5.html
- [4] *MCP3004/3008*, Datasheet, Microchip, 2008, accessed: April 26, 2017. [Online]. Available: <https://stevieb9.github.io/rpi-adc-mcp3008/datasheet/MCP3008.pdf>
- [5] “LPC1110/11/12/13/14/15,” Datasheet, NXP, 2014, accessed: Feb. 24, 2017. [Online]. Available: https://www.nxp.com/documents/data_sheet/LPC111X.pdf
- [6] “ERJ-8ENF2003V Panasonic - Mouser Europe,” Web page, Mouser Electronics, 2017, accessed: Feb. 24, 2017. [Online]. Available: <http://www.mouser.com/ProductDetail/Panasonic/ERJ-8ENF2003V/?qs=sGAEpiMZZMtIubZbdhIBINbsJb0hdM3zoIr%2fb1JCZdI%3d>
- [7] *TPS56320x 4.5-V to 17-V Input, 3-A Synchronous Step-Down Voltage Regulator in SOT-23*, Datasheet, Texas Instruments, 2015, accessed: February 25, 2017. [Online]. Available: <http://www.ti.com/lit/ds/symlink/tps563201.pdf>
- [8] *LM386M-1/LM386MX-1 Low Voltage Audio Power Amplifier*, Datasheet, Texas Instruments, 2004, accessed: March 16, 2017. [Online]. Available: <http://www.ti.com/lit/ds/symlink/lm386.pdf>
- [9] C. of Kentucky, “Time and temperature relationship to serious burns,” *Cabinet for Health and Family Services*, 2013, accessed: Feb. 3, 2017. [Online]. Available: <http://chfs.ky.gov/NR/rdonlyres/2CFE26ED-59A5-4F60-91A7-E2B5C0E9D000/0/BurnsintheElderly091213.pdf>
- [10] N. C. Institute, “Cell phones and cancer risk,” 2016. [Online]. Available: <https://www.cancer.gov/about-cancer/causes-prevention/risk/radiation/cell-phones-fact-sheet>
- [11] IEEE, “Ieee code of ethics,” 2017. [Online]. Available: <http://www.ieee.org/about/corporate/governance/p7-8.html>
- [12] ACM, “Code of ethics,” 2017. [Online]. Available: <http://ethics.acm.org/code-of-ethics>

Appendix A Flex Sensor Cost Analysis

As an alternative to commercially available piezoelectric flex sensors, we constructed our own optical based flex sensors. Table 7 summarizes the itemized costs if the sensors were to be commercially produced in bulk. Listed costs represent the unit cost of the item if purchased at the listed quantity.

Table 7: Itemized cost for single optical flex sensor

FLEX SENSOR PARTS					
Part Name	Distributor	Bulk Cost	Bulk Quantity	Quantity	Total
4-in Micro Tubing	US Plastic	\$ 0.02	3000 ft	1	\$ 0.02
4-in Heat Shrink	DigiKey	\$ 0.05	3000 ft	1	\$ 0.05
Resistor	DigiKey	¢ 0.20	1000 pcs	1	¢ 0.20
Photoresistor	Alibaba	\$ 0.08	1000 pcs	1	\$ 0.08
3mm LED	DigiKey	\$ 0.08	1000 pcs	1	\$ 0.08
6-in Wire	DigiKey	\$ 0.04	1000 ft	4	\$ 0.15
				TOTAL	\$ 0.58

Appendix B Requirement and Verification Table

Table 8: System Requirements and Verifications

Power Module			
Requirement	Verification	Verified	Points
Battery module must be able to power the device for 4 hours of continuous use.	1. Attach ammeter between battery module and power module 2. Fingerspell the alphabet for 15 minutes 3. Verify the battery's mAh rating is 16 times larger than the average measured current	No	12
9.0 V to 5.0 V DC buck converter must supply a DC voltage of $5.0\text{ V} \pm 0.1\text{ V}$ with a maximum current of 200 mA.	1. Attach 24Ω resistor across the converter output as load 2. Attach multimeter across load 3. Ensure the voltage it is within the range 4.9 V - 5.1 V.	Yes	4.5
9.0 V to 3.3 V DC buck converter must supply a DC voltage of $3.3\text{ V} \pm 0.1\text{ V}$ with a maximum current of 100 mA.	1. Attach 33Ω resistor across the converter output as load 2. Attach multimeter across load 3. Ensure the voltage it is within the range 3.2 V - 3.4 V.	Yes	4.5
Module Total:			12
Controller Module			
Requirement	Verification	Verified	Points
Sensor input should deterministically map sensor inputs to character outputs.	1. Set the microcontroller to listen to serial in for data 2. Send a random feature vector over serial 3. Verify output of system is the same on all attempts 4. Repeat for five additional random sensor inputs	Yes	1
Controller module recognition algorithm makes character predictions at 15 Hz.	1. Monitor the serial out port of the microcontroller 2. While wearing the glove, fingerspell the alphabet 3. Using the microcontroller serial output, verify predictions are made at a minimum of 15 Hz	Yes	3
Character must be recognized within 1 second of completing the entire gesture.	1. While wearing the glove, fingerspell the entire alphabet 2. Using a timer, measure the latency between gesture and output 3. Ensure the latency never exceeds 1 second	Yes	3

Continued on next page

Table 8 – continued from previous page

Requirement	Verification	Verified	Points
Repeated characters must be recognized within 3 seconds of the previous character's identification.	1. While wearing the glove, fingerspell the entire alphabet 2. Immediately after recognition, fingerspell the character again 3. Ensure the latency never exceeds 3 second	No	3
Entire controller module should draw less than 40 mW.	1. Attach ammeter between power module and controller module 2. Fingerspell the alphabet 3. Ensure no more than 12 mA is drawn on average	Yes	2
Classification of characters should, in total, be at least 95% accurate.	1. While wearing the glove, fingerspell the entire alphabet five times 2. Ensure there are no more than six misidentifications.	Yes	5
Module Total:			17
Sensor Module			
Requirement	Verification	Verified	Points
The sensor subsystem must produce a unique feature vector after quantization for each letter of the ASL alphabet.	1. Monitor the serial out port of the microcontroller 2. While wearing the glove, fingerspell the entire alphabet 3. Verify every letter produces a unique feature vector	Yes	5
The worst case power dissipation in each continuity sensor must be less than 1 mW.	1. Attach an ammeter to the input of the continuity sensor 2. Touch the positive side to the negative side 3. Ensure no more than 0.33 mA flows through the circuit 4. Repeat for the five remaining continuity sensors.	Yes	3
Each flex sensor circuit must produce voltages that differ from the threshold voltages by more than the resolution of the ADC when the flex sensors are in the high, moderate, and low resistance states.	1. Attach a multimeter to the output of the flex sensor circuit 2. Measure the voltage nearest the threshold resistance in the high resistance state. 3. Repeat for the moderate and low resistance states. 4. Ensure there is minimum of 0.01 V difference between each measured voltage and the threshold voltages. 5. Repeat for the four remaining flex sensors.	Yes	3

Continued on next page

Table 8 – continued from previous page

Requirement	Verification	Verified	Points
Each contact sensor should produce a voltage that is read by the microcontroller as digital high when contact is made and as digital low when contact is not made.	1. Attach the contact sensors to the microcontroller across a digital port. 2. Make and break contact between the sides of the contact sensors. 3. Ensure that the microcontroller reads high and low values appropriately.	Yes	3
The standard deviation of each output of the accelerometer is less than 0.5 G.	1. Place the accelerometer flat on a table 2. Monitor the serial out port on the microcontroller 3. Print out all quantized values of the accelerometer over serial 4. Record data for 1 minute 5. Ensure the standard deviation of each output is less than 0.5 G	Yes	1
Module Total:			15
Output Module			
Requirement	Verification	Verified	Points
The speaker must be able to output 50 db of audio from a distance of one meter.	1. Place an audio noise meter capable of measuring decibels one meter from the glove 2. Fingerspell a letter 3. Verify at least 50 db was measured	Yes	2
Entire output module should draw less than 700 mW.	1. Attach ammeter between power module and output module 2. Fingerspell the alphabet 3. Ensure no more than 140 mA is drawn on average	Yes	2
The status LEDs must be bright enough to be comfortably seen from one meter away.	1. Stand one meter from glove in a well light room 2. Verify all status LEDs are clearly visible.	Yes	2
Module Total:			6
Total Points:			50

Appendix C Supporting Diagrams

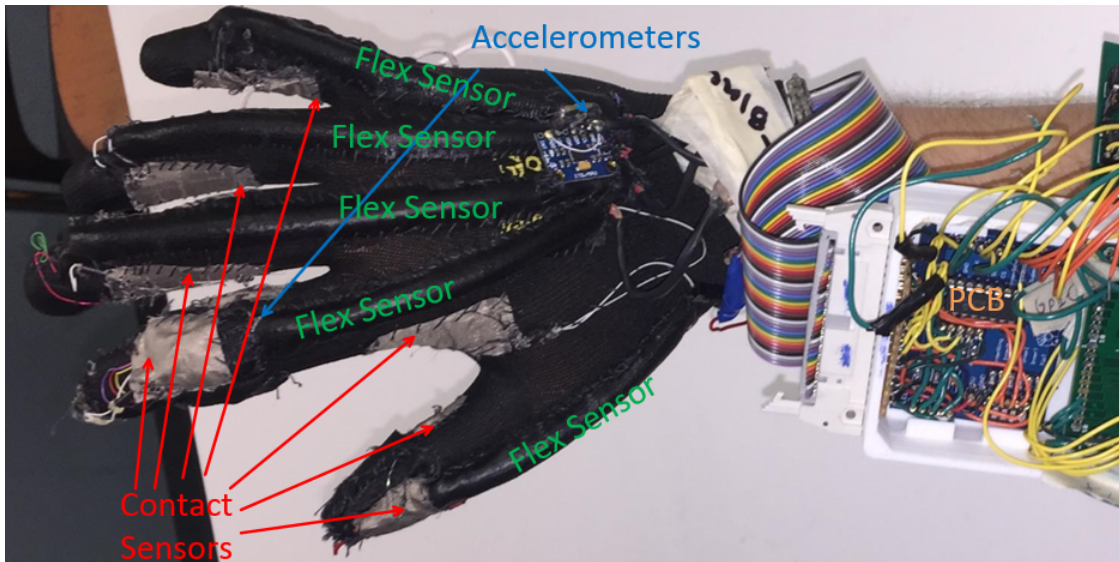


Figure 16: Sensors Module

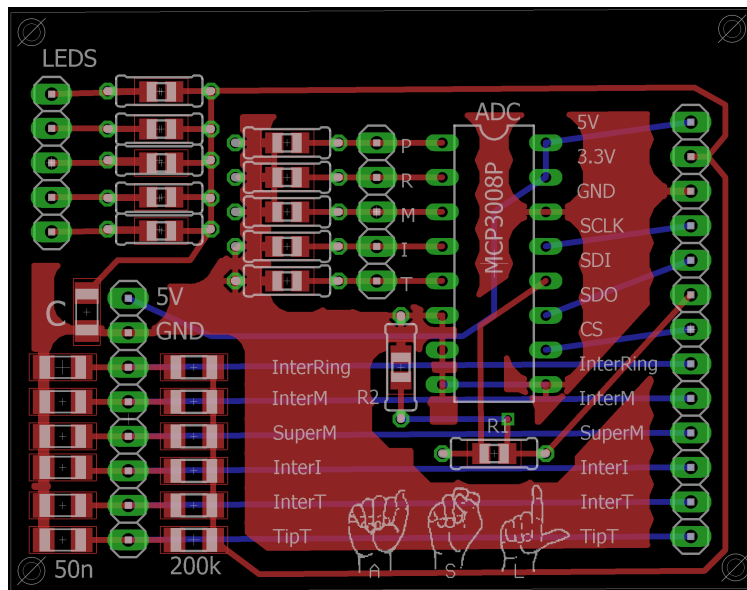


Figure 17: Sensors Module Printed Circuit Board. This circuit board sends signals to the sensors mounted on the glove, processes the signals it receives from the sensors, and produces and digital signals for the microprocessor.

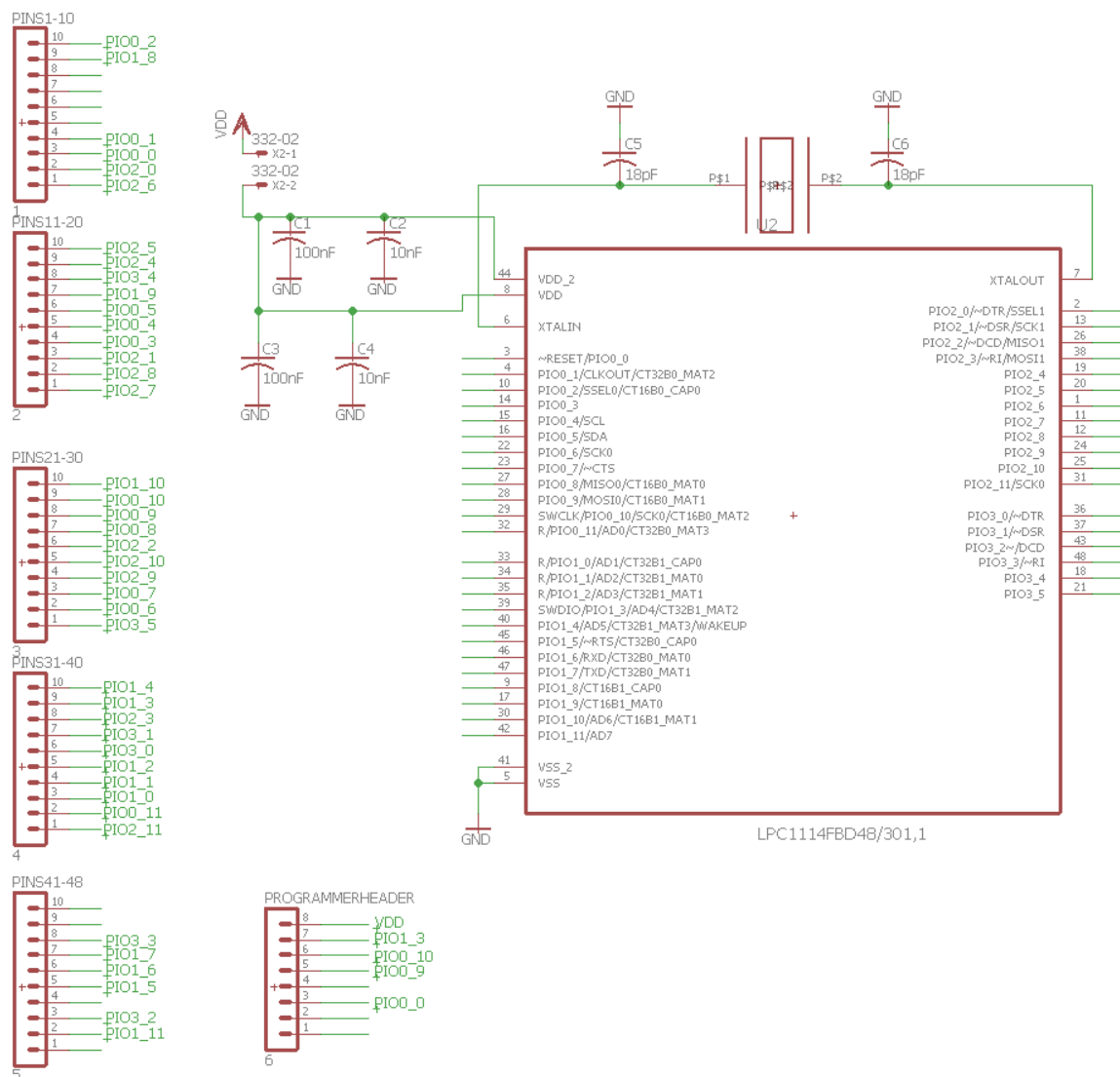


Figure 19: Schematic for the microcontroller.

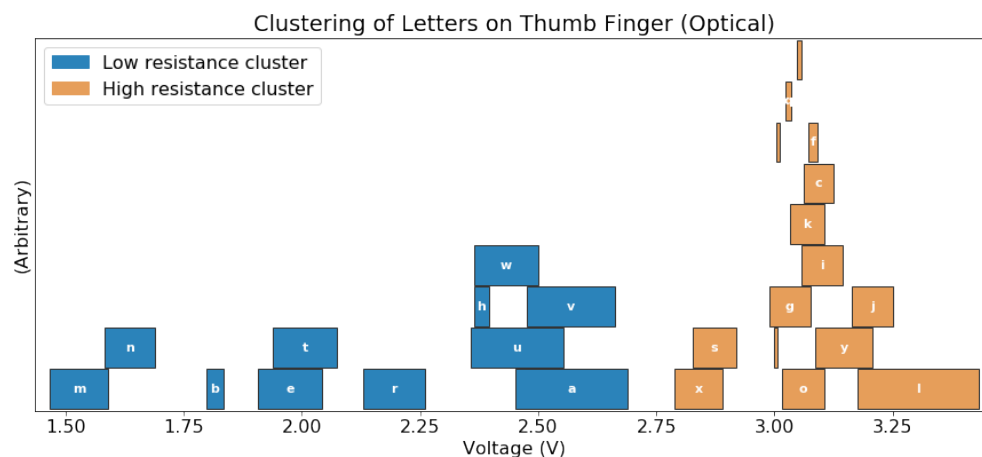


Figure 20: Letter clustering by resistance of thumb flex sensor.

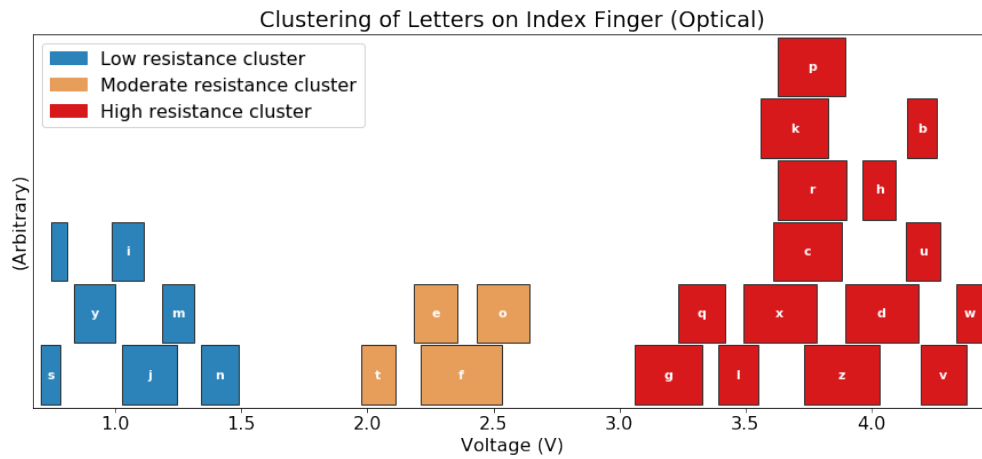


Figure 21: Letter clustering by resistance of index flex sensor.

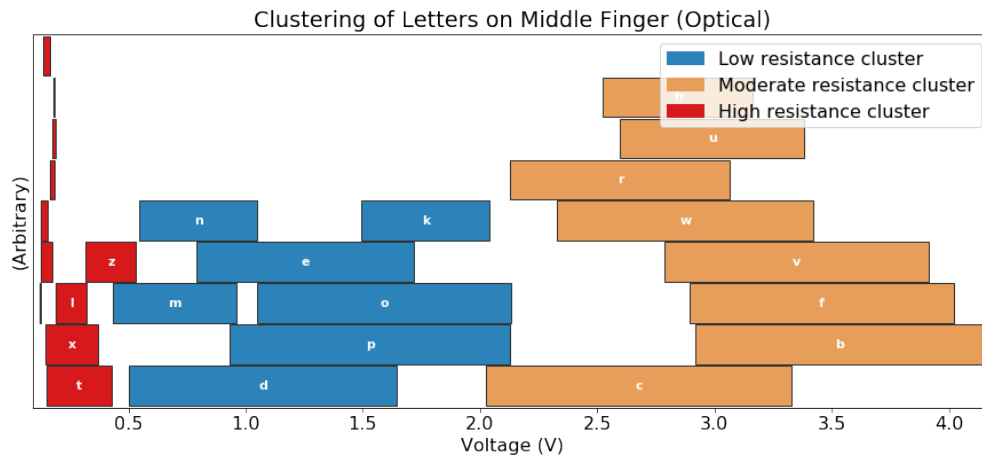


Figure 22: Letter clustering by resistance of middle flex sensor.

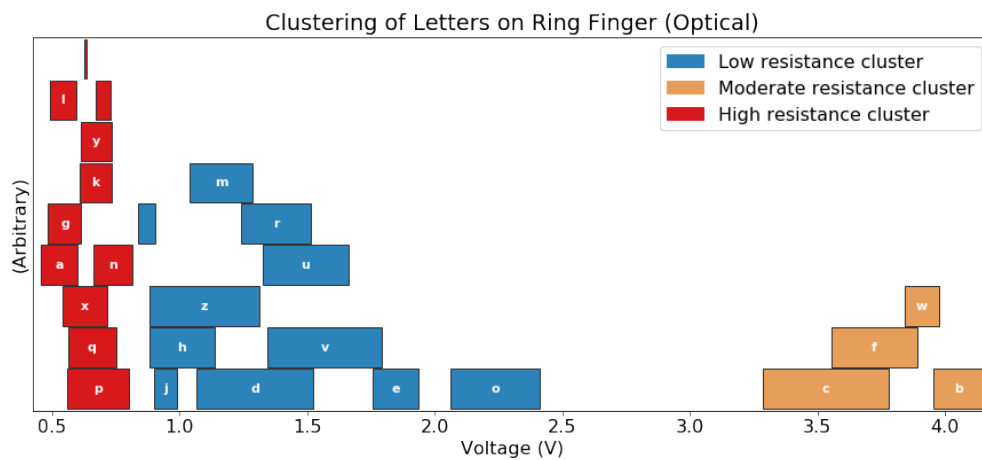


Figure 23: Letter clustering by resistance of ring flex sensor.

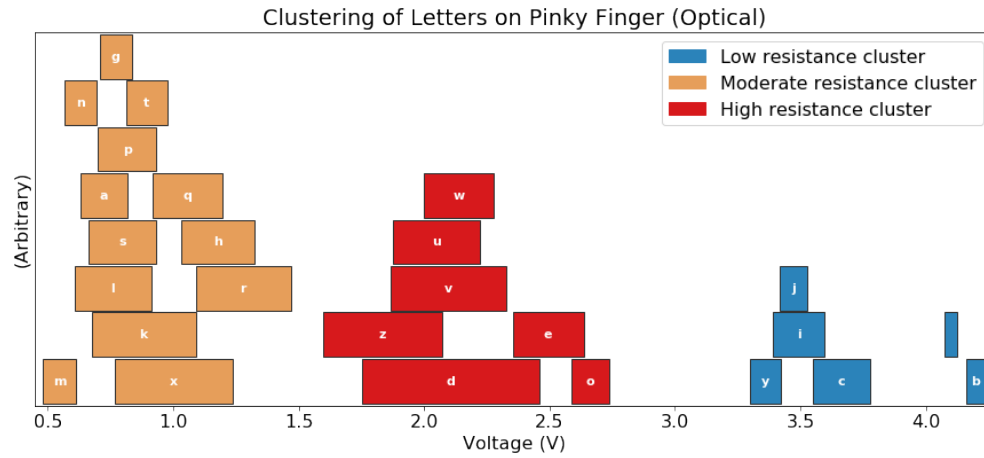


Figure 24: Letter clustering by resistance of pinky flex sensor.

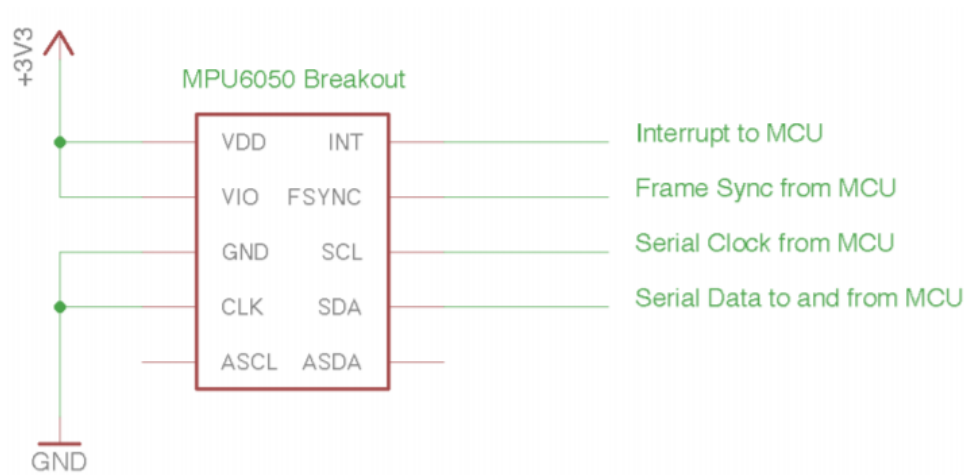


Figure 25: Schematic of accelerometer module interface.



Figure 28: Transient ripple for 3.3V output