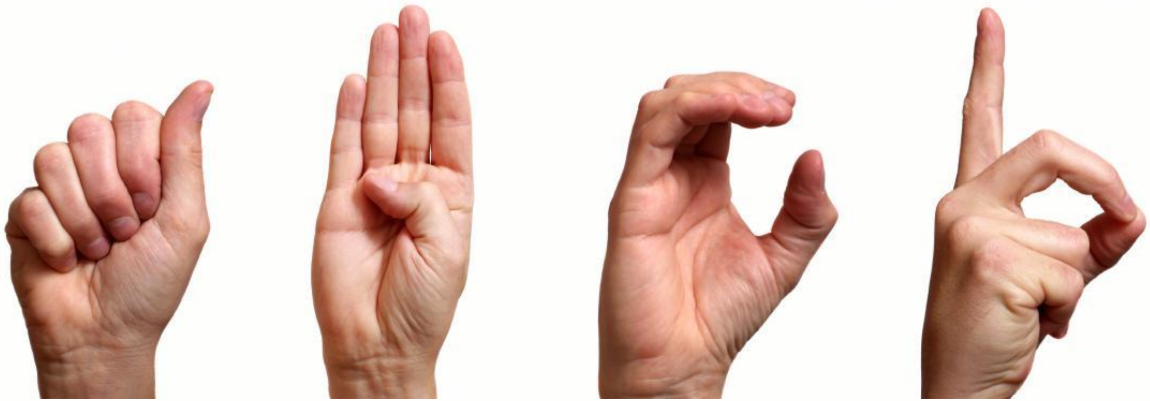
Sign Language Interpreter Glove



Group 24

Christopher Delgado	**Electrical Engineering**
Emmanuel Hernandez	**Electrical Engineering**
Jason Balog	**Electrical Engineering**
Ramon Santana	**Electrical Engineering**

Thank You to our generous sponsors!!!

BOEING

leidos

Motivation

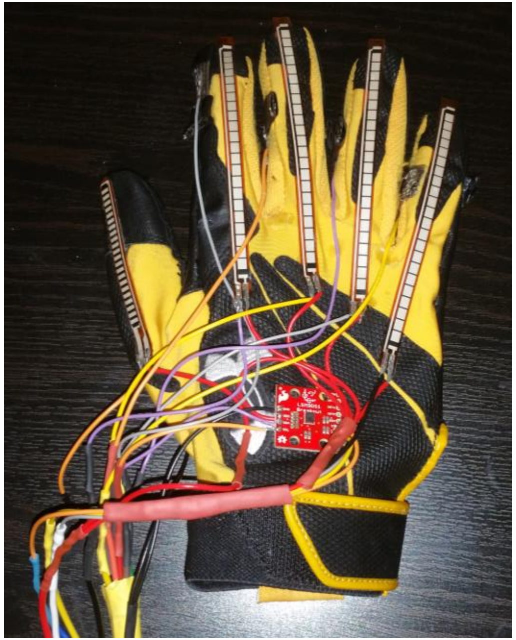
- The Survey of Income and Program Participation (SIPP) – estimates that about 1,000,000 are functionally deaf in the United States.
- The World Health Organization estimates that over 5% of the world's population – 360 million people – has disabling hearing loss (328 million adults and **32 million children**).
- The original motivation to pursue this project comes from one of our team members who has experienced the difficulty of communicating with his speech-impaired sister.

Goals and Objectives

- Our objective is to establish communication between a sign language speaker and a non-sign language speaker. Any letter the user signs will be displayed through a user interface where the non ASL-speaker can read the letter. We also want to implement a learning mode, where the user has the option to learn the American Sign Language letters.

- **Hardware**
    - Flex sensors
    - Contact sensors
    - Accelerometer
    - Gyroscope
    - MCU: ATMega328
    - Bluetooth Low Energy

- **Software**
    - Android Mobile Application

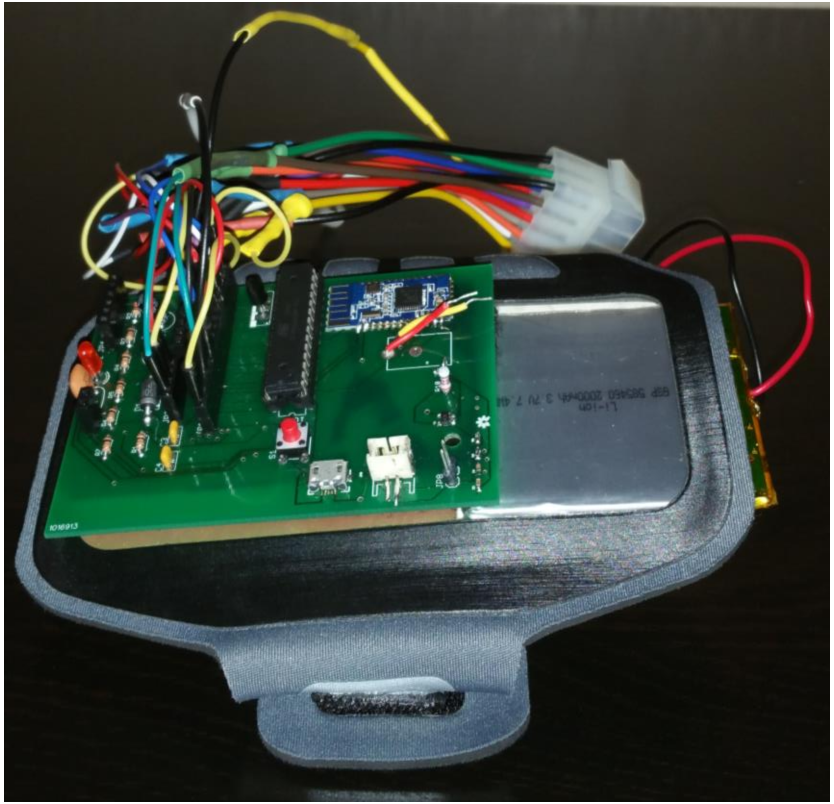


Specifications

- Lightweight
- Portable
- Energy Efficient

Glove Weight	< 1.5 lb.
Lithium Ion Battery	3.7 V 2000mAh
Battery Life	13 hours

Specifications Table



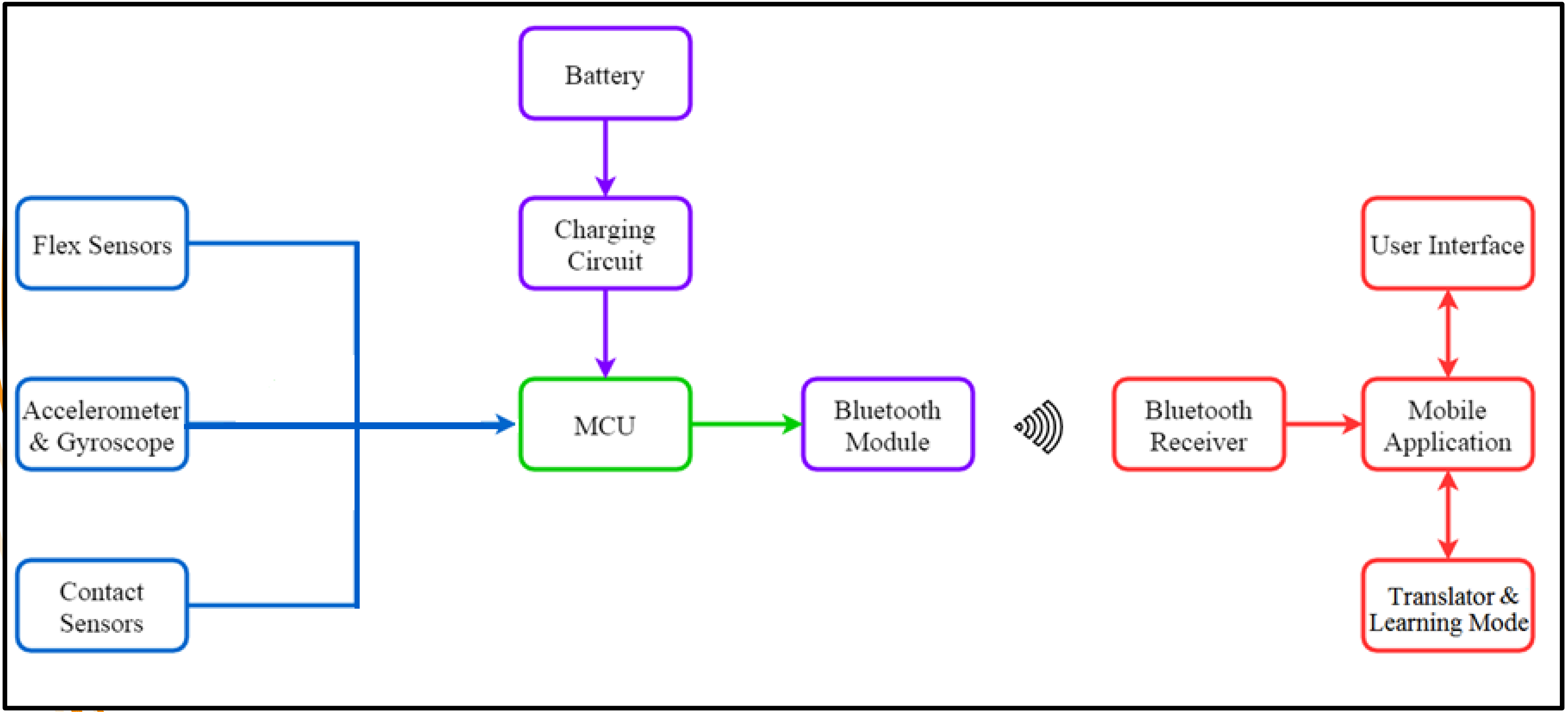
Related Standards

- IEEE Standards 802.15.1™-2005(Revision of IEEE Std 802.15.1-2002)
  -Establishes a communication standard optimized to serve a variety of medical and non-medical applications.

- Safety standards for Lithium-ion batteries (International Electrotechnical Commission (IEC) and International Organization for Standardization (ISO).
  -IEC 62133-2: safety requirement for portable battery cells
  -IEC 62660: batteries for EV/HEV applications
  -IEC 61427: secondary cells and batteries for renewable energy storage

- Google's Android developers have set up a multitude of standards and qualification on the different aspects of an android application.
  -UX-B1: App uses common user interface patterns and conventional use of icons

Block Diagram

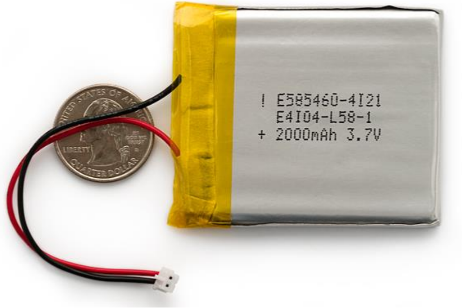


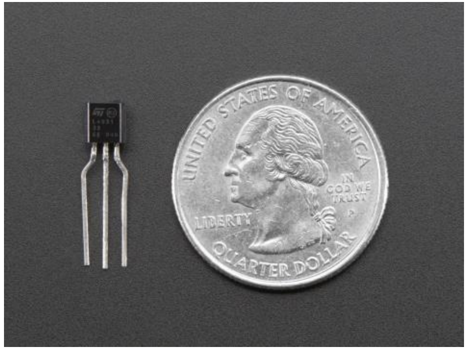
Significant Hardware Decisions - Wireless Communication

WiFi	Bluetooth Low Energy	NFC
Pros: - Security (WPA2) - Range 150-300 feet	**Pros:** - Range 20-120 feet - Consumes less power than Wi-Fi	**Pros:** - Consumes less power than Bluetooth Low Energy
Cons: - Needs Router - Consumes lots of Power - Not portable friendly	**Cons:** - No cons for SLIG ☺	**Cons:** - Limited Range (about 0-4 inches)

Significant Hardware Decisions - Battery & Regulators

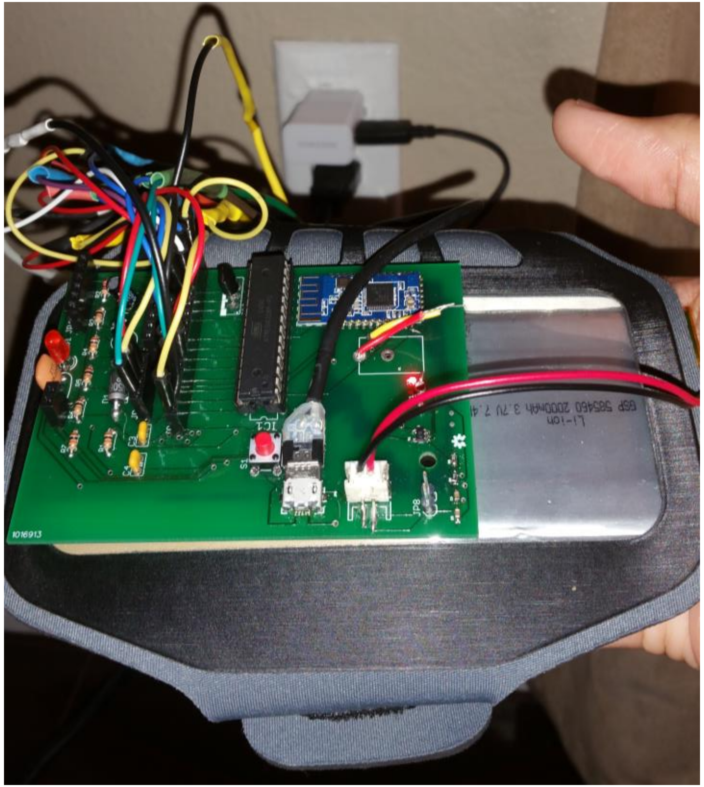
- **Polymer Lithium Ion Battery - 2000mAh**
  - 3.7V at 2000mAh
  - Built-in protection against over voltage and over current
  - Self-discharge rate <8% per month
- **Regulators - TI LP2985**
  - The purpose of voltage regulators is to keep a constant voltage level
  - Dropout Voltage: 0.4 V
  - Output Voltage: 3.3 V





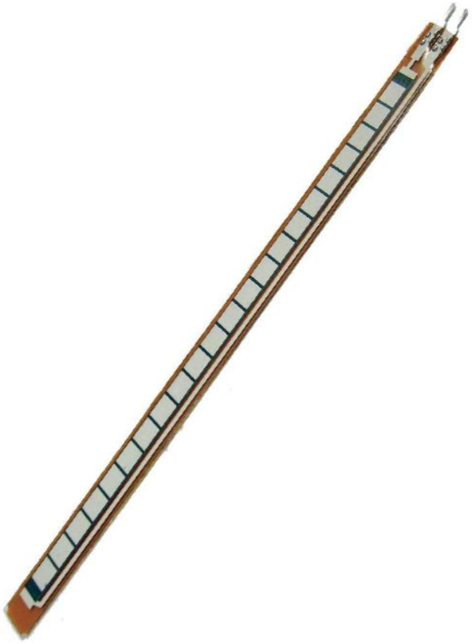
Significant Hardware Decisions - Charging

- **Charger input:**
  - Micro-USB
  - Charge current 500mA
- **Charger output:**
  - Single-Cell batteries Only
  - Lithium-Polymer or Lithium-Ion Only
  - Max Voltage: 4.2 V
- **Top challenges were:**
  - Learning how to solder small electronics



Significant Hardware Decisions – Flex Sensors

- Flex sensors will be the primary sensors employed in this projects.
- Flex sensors will be used to detect the degree to which each finger is bent on the hand performing the sign language gesture.
- Each letter will have a specific configuration based on expected outputs of the five flex sensors.



Significant Hardware Decisions – Flex Sensors

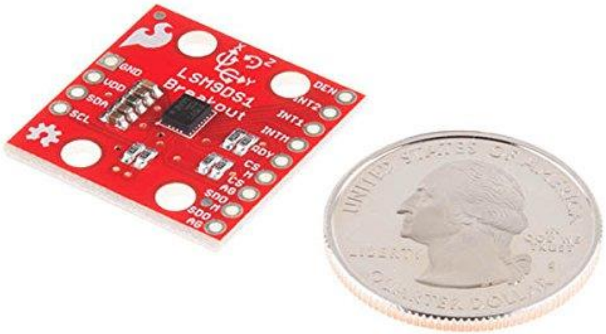
SpectraSymbol Long Flex Sensor	Images Two-Directional Bi-Flex Sensor	Tactilus Flex
Pros: - widely available - accurate output	**Pros:** - built in pressure sensors	**Pros:** - claims high durability (>35 million cycles) - high quality leads
Cons: - fragile leads	**Cons:** - irregular shape - sold exclusively by manufacturer	**Cons:** - increased cost - sold exclusively by manufacturer

Significant Hardware Decisions – Gyroscope and Accelerometer

- Accelerometers and gyroscopes can be used to measure these type of parameters, which are crucial in identifying certain sign language letters.

- Examples include "**j**" and "**z**" or distinguishing between “**g**” and “**q**”.

- The group is using the SparkFun 9 Degrees of Freedom IMU Breakout - LSM9DS1.

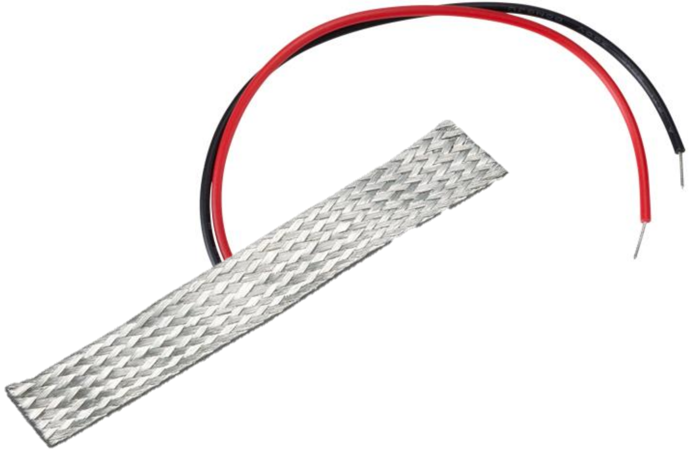


Significant Hardware Decisions – Contact Sensors

- Contact sensors were crucial in telling apart the following pairs of sign language: **R**, **U**, and **V**, **S** and **T** and **M** and **N**.

- The contact sensors were made with strips of copper braids that were connected to the power supply through wire leads.

- The glove detects when two contact sensors come in contact creating a closed circuit.



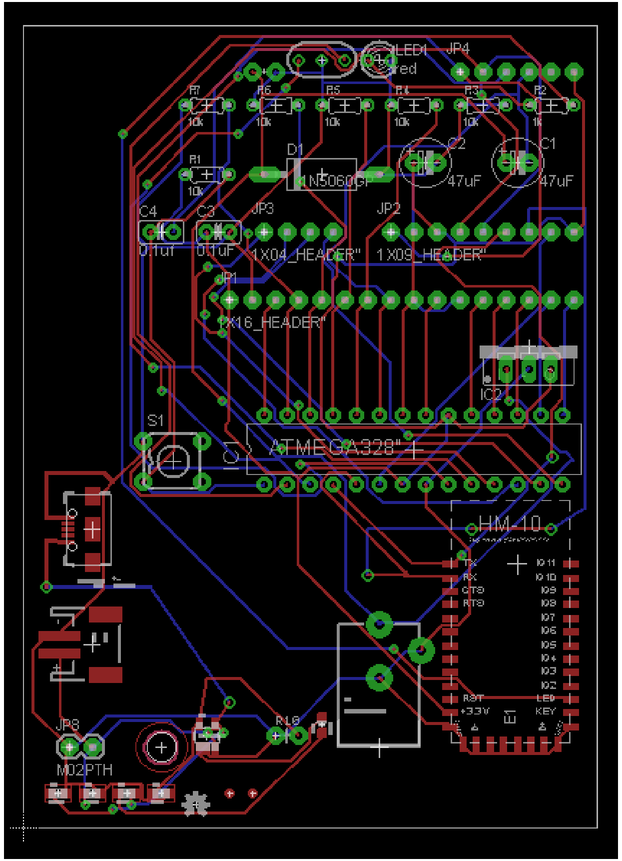
Significant Hardware Decisions – PCB

- The PCB was designed using Eagle CAD software and fabricated by Bay Area Circuits and most parts were sourced from Digikey.com

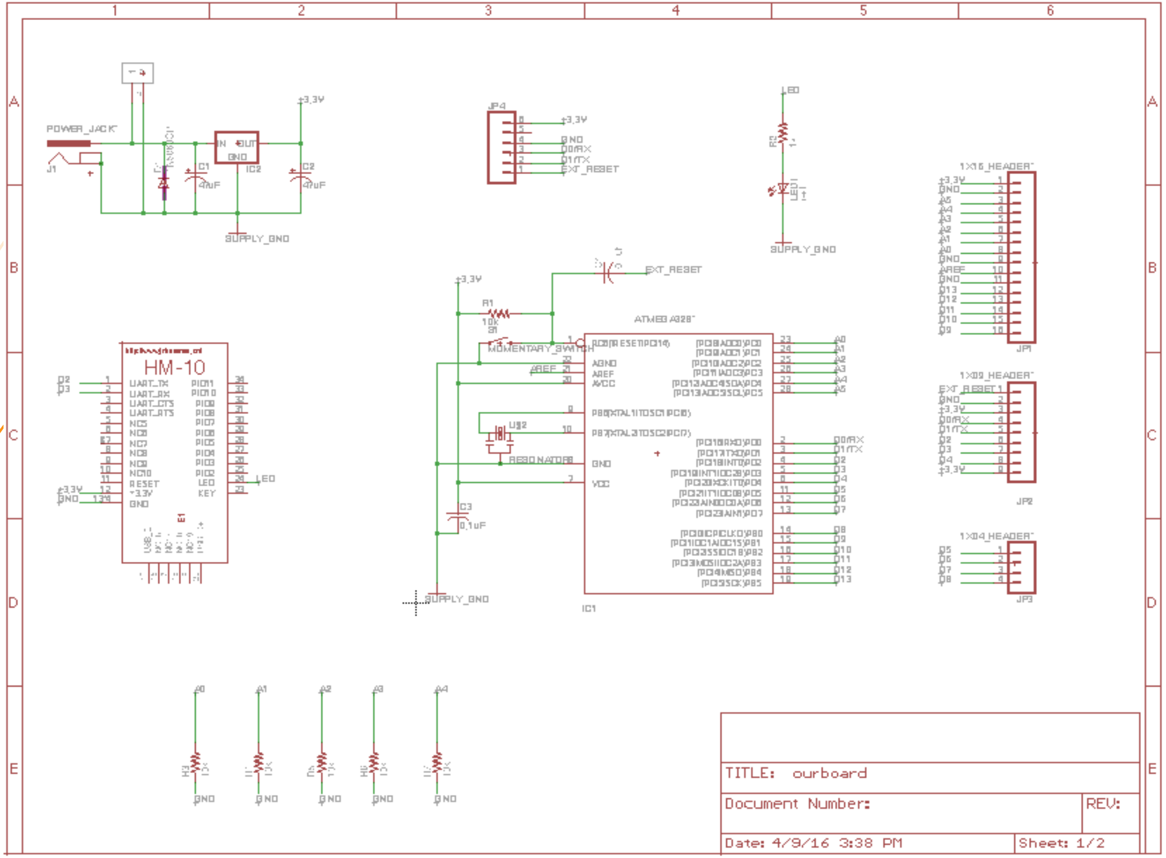
- It was designed as two layer board measuring 2.5” by 3.5” and is meant to be worn on the forearm.

- Designing and building the PCB board was the biggest challenge with the hardware.

- Top challenges were:
  - Learning Eagle CAD
  - Correctly Wiring Components
  - Ordering the write components



Significant Hardware Decisions – PCB



Significant Hardware Decisions – Microcontroller

- The unit that is chosen will have to employ at least 6 analog input pins (with included ADC units), have digital i/o pins, and also be able to establish a line of serial communication.

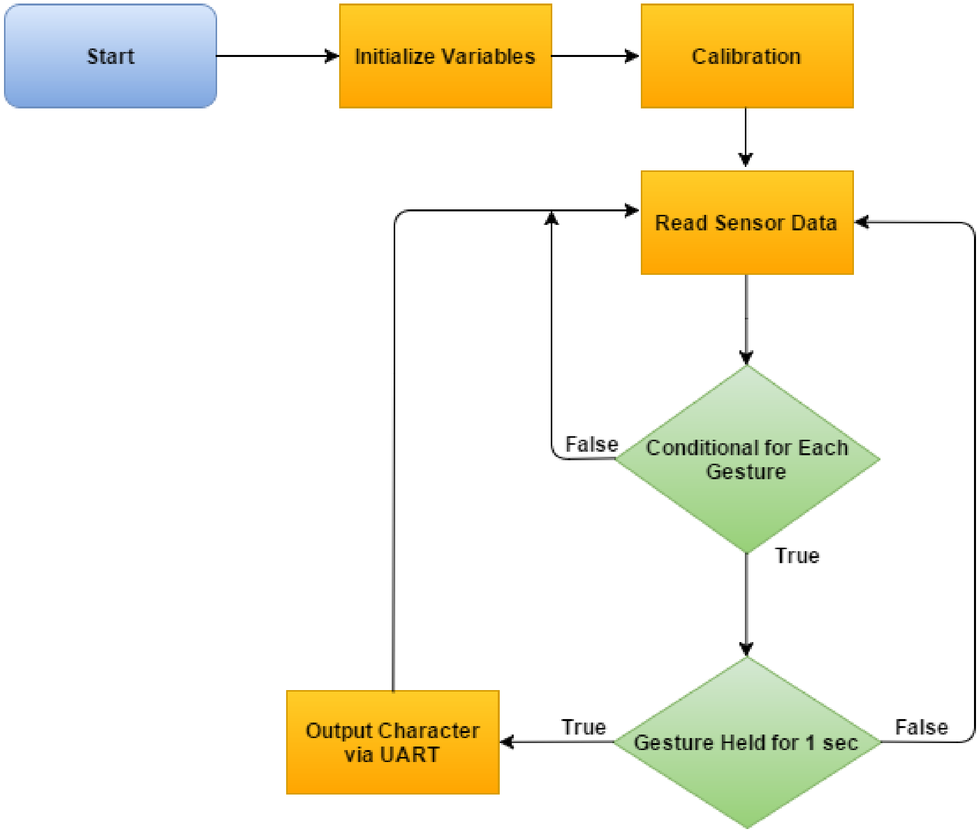
- When considering the microcontroller unit, it is also important to note that the two units come with their own native programming environments.

Table 3.2: MSP430 vs. ATMega32 comparison

Feature	MSP430	ATMega32
Analog Input Pins	8	12
Digital Input Pins	8	20
Random Access Memory	512 Bytes	2.5 Kilobytes
Data Bus	16 bits	8 bits
Speed	16 MHz	16 MHz
Cost/Vendor	$9.99 → mouser.com	$24.95 → ebay.com

Gesture Recognition Software

- The gesture recognition system will use thresholds to determine what hand gesture is being performed.
- The ranges of values that are included in every type of hand gesture determine the minimum and maximum limits to the conditional statements.



Calibration Process

- The calibration process occurs during the ‘setup’ process of the program, before the infinite loop begins

- Calibration will capture the minimum and maximum input values from each sensor, and apply the mapping function.

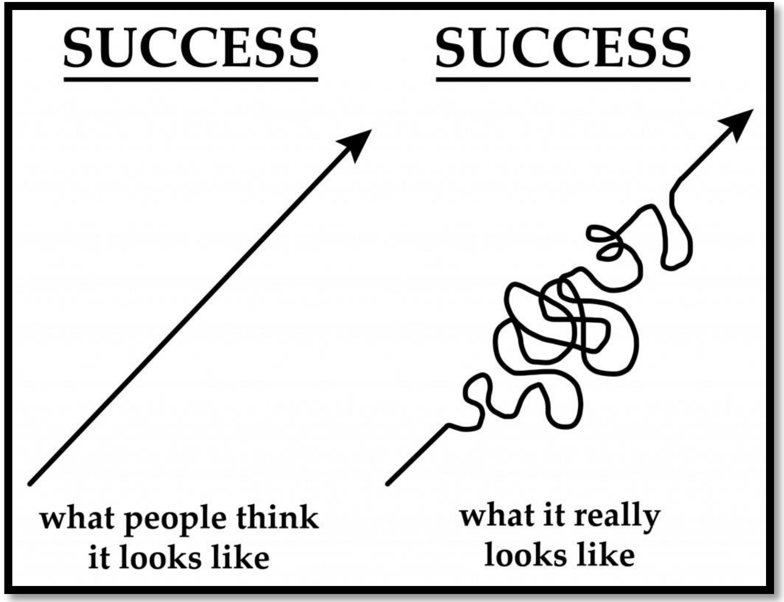
- Calibration is needed because analog flex sensors provide volatile data that needs to be normalized.

Gesture Recognition Software

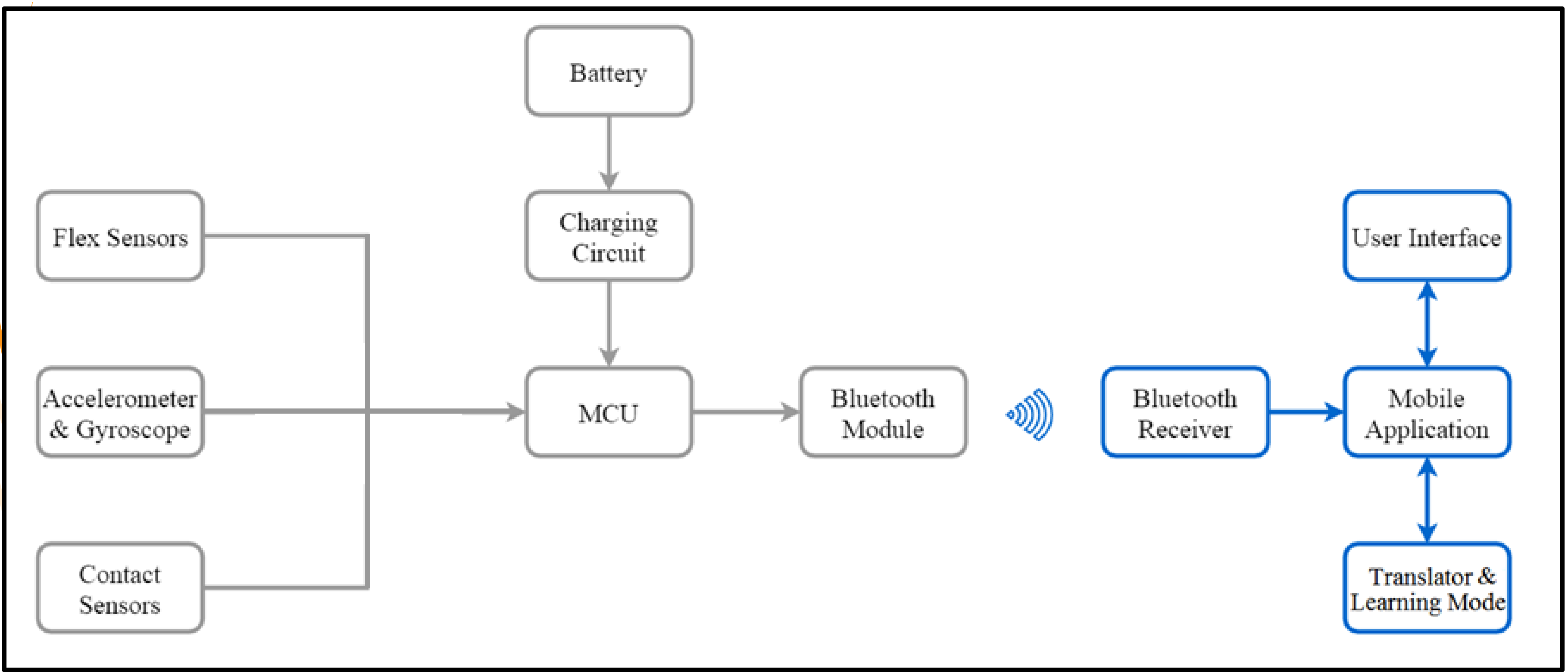
- Program scans sensors for data at one second intervals
- Arduino “map” function is used to apply calibration
- Normalized sensor values and motion data are used in conditional statements that determine gesture being performed

Gesture Recognition Set-Backs

- Contact sensors needed to have internal pull-up resistors enabled due to floating i/o pins
- Had to use SPI as opposed to I2C for the motion unit due to a lack of enough analog input pins
- Many distinct gestures unexpectedly produced similar flex sensor data and required methods of distinction



Software Components: Mobile Application



Mobile Application: Overview

- The mobile application is an important feature of this project that will serve as the user interface for the sign language glove.
- The mobile application is responsible for wirelessly displaying hand gestures performed by the glove onto a mobile phone screen as text.
- In order for the mobile application to be successful, the design of the app should consider the user, be simple and elegant, and meet all design requirements.

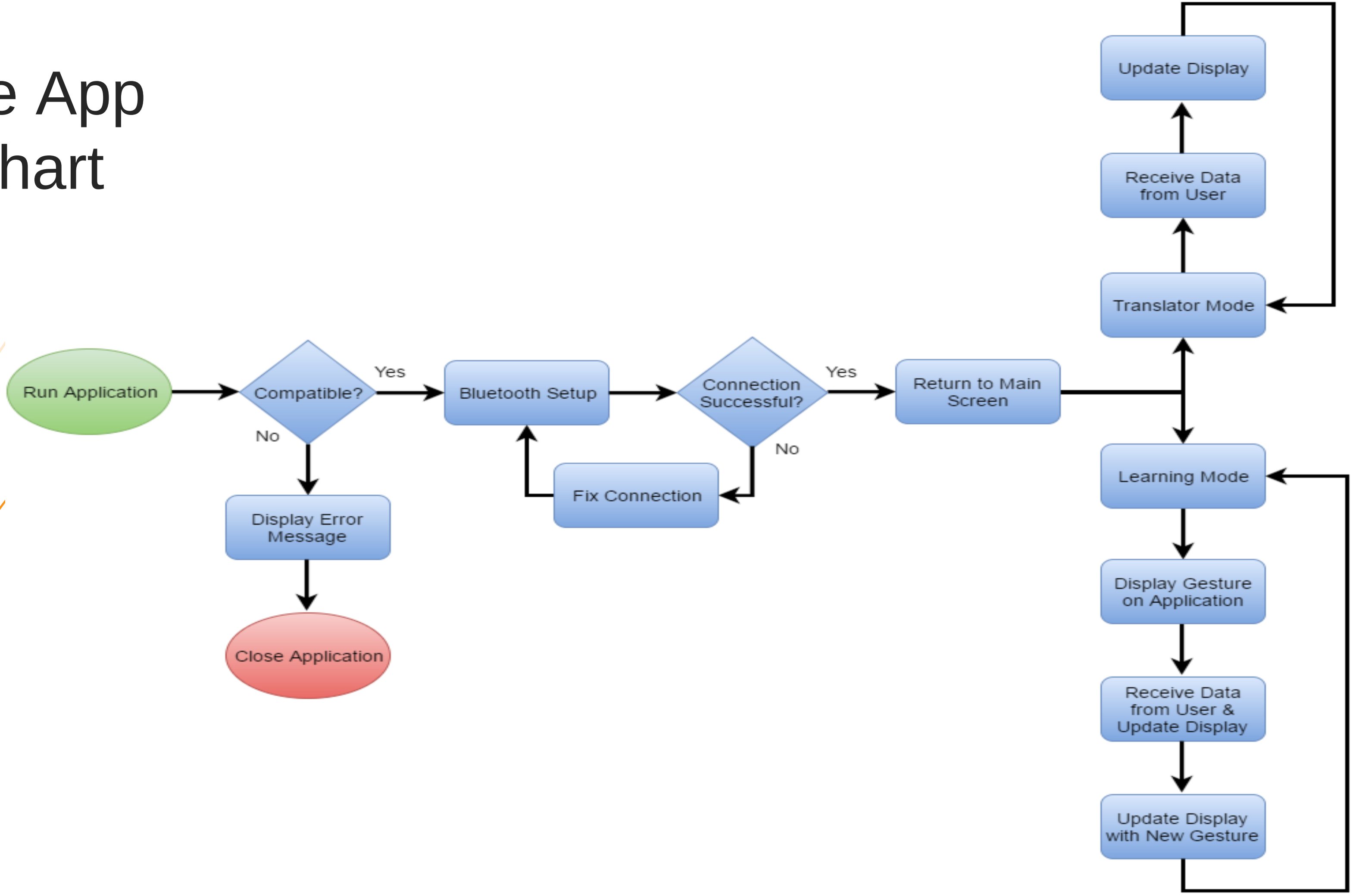
Potential Mobile Platforms

- **Android**
  - Programming language: Java
  - IDE: Android Studio
    - Open source and cross-platform compatible
  - BLE compatible
  - Everyone in our group owns one!
- **iOS**
  - Programming Languages: Swift or Objective-C
  - IDE: Xcode
    - Only Mac OS compatible
  - BLE compatible
- **Windows**
  - Programming Languages: C++, C#, and Visual Basic
  - IDE: Visual Studio
  - BLE compatible

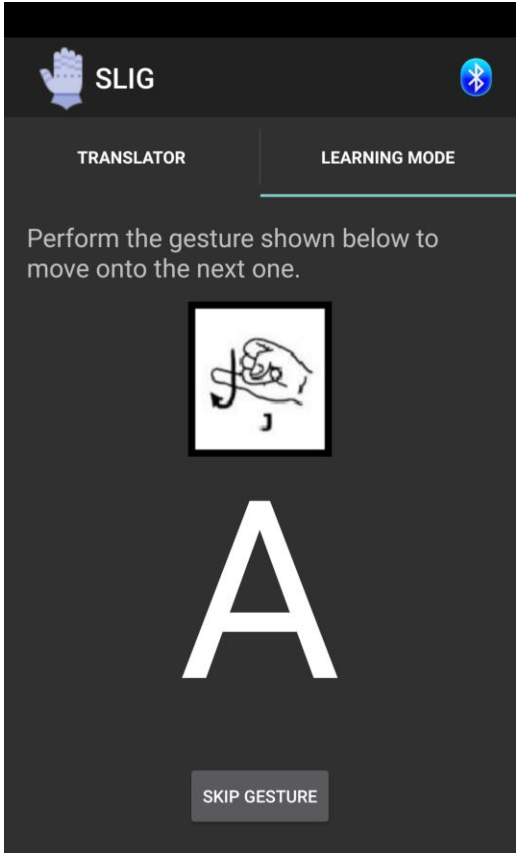
Android

Minimum Requirements	
Device Type	Android
Mobile Platform	Smartphone
Bluetooth Version	Bluetooth Low Energy v4.0
Platform Version	Android 4.3
Codename	Jelly Bean-MR2
Android API Level	18

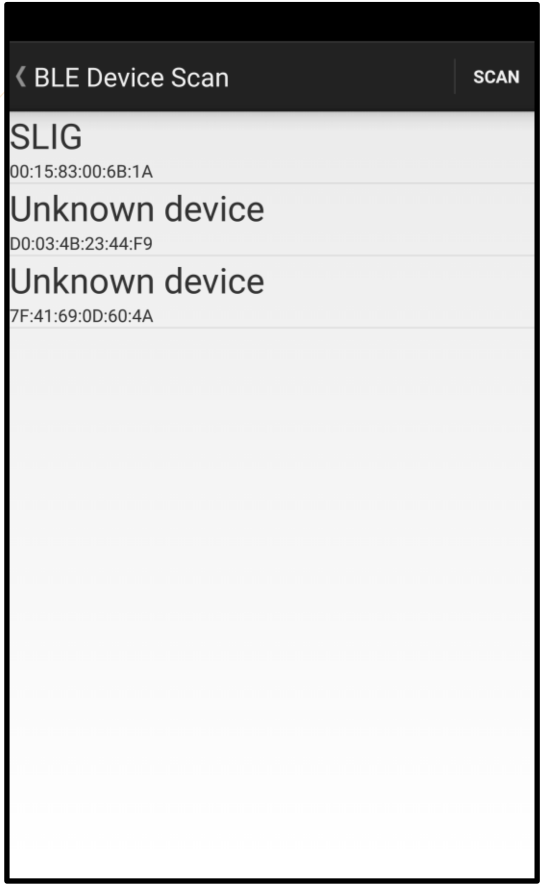
Mobile App Flowchart

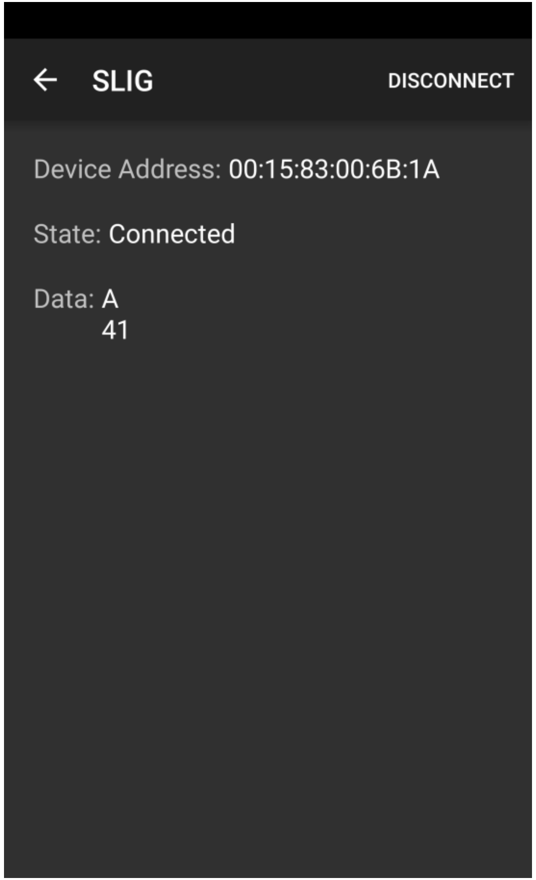


User Interface



User Interface





Budget And Financing

- **Sponsored by:** Boeing and Leidos
- **Initial Budget Estimate:** $400
- **Final Project Cost:** $825

Part Description	Price per Unit ($)	Quantity	Cost ($)
Flex sensors 4.5"	$12.95	15	$194.25
Flex sensors 2.2"	$7.95	1	$7.95
Development Board	$20.87	3	$62.61
ATmega328P	$13.99	3	$41.97
Bluetooth Module	$11.29	4	$45.16
Accelerometer & Gyroscope	$29.55	2	$59.10
Battery	$22.20	1	$22.20
PCB	$49.60	1	$49.60
PCB Components	$49.76	1	$49.76
2nd PCB Components	$52.99	1	$52.99
3rd PCB Components	$34.08	1	$34.08
Glove and Wristband	$20.22	2	$40.44
Glove and Armband	$45.00	1	$45.00
Soldering Materials et al	$120.00	1	$120.00
Total cost	-	-	**825.11**
Boeing & Leidos sponsorship	-	-	**$359.80**
Difference			**$465.31**

Questions ?