ECE 445 Design Document

Glove For Programmable Prosthetic Hand

Team Number: 55

Ryan Metzger (rtm4)

Quang Nguyen (quangpn2)

Sohil Pokharna (sohilp2)

Mentor TA: Barbhai, Sainath

Date: 2/20/2023

Table of Contents

| 1. | Introd | luction | 2 |
|----|--------|------------------------------|----|
| | 1.1. | Problem | 2 |
| | 1.2. | Solution | 2 |
| | 1.3. | Functions | 2 |
| | 1.4. | Benefits | 2 |
| | 1.5. | Features | 2 |
| | 1.6. | Visual Aid | 3 |
| | 1.7. | High-level requirements list | 3 |
| 2. | Desig | 'n | |
| | 2.1. | Block Diagram | 4 |
| | 2.2. | Physical Design | 5 |
| | 2.3. | Subsystem 1 | 6 |
| | 2.4. | Subsystem 2 | 7 |
| | 2.5. | Subsystem 3 | 13 |
| | 2.6. | Subsystem 4 | 13 |
| | 2.7. | Tolerance Analysis | 14 |
| 3. | Costa | and Schedule | |
| | 3.1. | Cost Analysis | 17 |
| | 3.2. | Schedule | 18 |
| 4. | Ethics | s and Safety | 19 |
| 5. | Appe | ndix | 21 |
| 6. | Refre | nces | 26 |

1 Introduction

1.1 Problem: Modern robotic prosthetics may achieve fine motor control through predefined hand motions encoded into the prosthetic. Modern prosthetics may have the ability to save preset positions but don't have the ability to adjust and tweak positions on the fly. We plan to implement a hardware/software solution that is able to measure the positions of a functional organic hand and translate this motion into a prosthetic hand in order for this prosthetic to mimic this motor control on the move. With features such as mirroring we are able to have 2 hands, 1 organic and simulated prosthetic, we are able to introduce a level of dynamic programmability. Also introducing multiple preset positions through organic hand gestures, the user can recall most used positions for convenience. By adding sensors to individual fingers, we can combine combinations of gestures in order to control the prosthetic beyond mirror mode and be able to change the preset positions using these gesture controls.

1.2 Solution: We propose to create a glove with flex sensors and hall effects that can measure the motion of the fingers and detect gestures that the user gives. From this the user can then control a robotic hand with their organic hand making it easier to adjust position as well as record motions for the robotic hand to execute.

1.3 Functions: The product is supposed to record an organic hand and either save positions from the organic hand or mimic the movement of the organic hand.

1.4 Benefits: It provides the user with the ability to make quick and easy changes to their robotic hand while on the go. It also allows the user to mimic the users organic hand's movement.

1.5 Features: The parts of this project that make it marketable is the quick and easy re-programmability on the fly and the ability to mimic the user's hand movement.

1.6 Visual Aid:



Figure 1.6.1: Visual Aid for Glove for Programmable Prosthetic Hand

1.7 High-level requirements list:

- 1. Have the ability to detect each of the 5 fingers' movement and also move the digital/physical robotic hand appropriately
- 2. Must detect 4 gestures at minimum
- 3. Must have the ability to move to 3 pre-set positions and mirroring mode

2 Design:

2.1 Block Diagram:



Further details about each of the substems are showin in sections 2.3 to 2.6. The prosthetic hand subsystem is the subsystem that encompasses the robotic hand for hte dmeo. It is made up of servos and an Aduino to control it. The prosthetic hand subsystem gets the information of the sensors from the processing and communication subsystem. This subsystem has the micro in it and gets its information via blue tooth. The motion and gesture detection unit is made up of 4 hall-effect sensors and 5 flex sensors to get the orientation of the hand and the mode the glove is in. Finally the power and battery subsystem powers everything.

2.2 Physical Design



In the physical design you can see that the hall effect sensors are located in the tips of the four fingers and the magnet resides on the thumb. The flex sensors have to cover the middle and top knuckle at minimum. The pcb and batter pack will either reside on the wrist or the back of the hand.

2.3 Subsystem 1: Processing/Communication Unit Microcontroller -

The Microcontroller is responsible for processing Motion/Gesture Unit inputs, processing the data and sending the results via Bluetooth to the Hand. This unit will also feature small LEDS that give the user feedback on what mode they are in/the preset options they are selecting.

Microcontroller must contain on-chip support for bluetooth LE and be able to run off 3.3V. The microcontroller must have on-chip Flash in order to support saving user configurations or an off chip-flash will interface via SPI to achieve the same function. Ideally the micro will have a power saving/sleep mode that will increase battery life of the system. The microcontroller must have at least 5 analog in pins and 4 digital in pins at the minimum and be able to run off the voltage regulator ripple specs. The User Feedback LEDs must be able to operate off of 3.3V.

| Requirement | Verification | |
|--|---|--|
| Make sure the processor can distinguish between all the different hall effects and set the LEDs to the correct modes | a) Check to see if we are successfully saving the different present values for the present positions b) When the system is complete go through all of the gestures and see if the LED signifying what mode is correct as well as the robotic hand going to the correct position as shown in the gestures table below | |
| 2) Check to see if Bluetooth can transmit and receive information | a) First check if we can transmit data between the micro and the Arduino. try and send a bit stream from the micro to the Arduino b) Try and send which hall effects are triggered and send the resistance values of the flex sensors from the micro to the Arduino | |
| 3) Flex sensors are successfully read | a) First get the input of the values of all the flex sensors and make sure we get any input at all b) Then we need to at minimum do a two-point calibration for the flex sensors. Have it at its minimum value and measure the resistance and its | |

2.4 Subsystem 2: Motion/Gesture Detections Unit.

The Motion/Gesture Unit is used to collect data about the user's finger positions as well as to control the unlocking/programmability it sends data to the Processing Unit.

The Detection Unit must consist of 5 Analog Flex Sensors and 4 Hall Effect Sensors that will all interface with the microcontroller for sensor feedback and draw power at 3.3V from the Power/Battery System. The noise must be filtered and magnetics tuned so that each hall effect sensor input is distinguishable from one another and can be repeatedly triggered. The flex sensors will interface with a resistor of the same nominal Resistance to form a voltage divider to read flex/finger positions may need to consider input impedance of ADC pins as some flex sensors have high impedance.

| Gestures | Meaning | |
|-----------|---|--|
| \square | Turns mirroring mode on | |
| | Mirroring mode is when the robotic hand copies the movement of the hand | |
| | | |

| 20 | Turns preset mode on |
|-----|---|
| X 2 | Preset mode is when the user is in this mode the gestures labeled preset 1 - 3 will set the hand in the saved configuration the user wants it in. Also while in this mode gestures recorded 1-3 will record the hand position of the user and save it as the preset position. For example, if the user does the gesture "record 2" the glove will record the position of the hand and then save the position of the hand that way if "preset mode 2" is triggered the robotic hand will go to that position. |
| | Preset 1: this will set the robotic hand to the first of the three preset positions |
| | |

| X2 | Record 1:This sets the glove in record mode and will wait 5 seconds to take a snapshot of the user's hand to override the saved position for preset mode 1 |
|----|--|
| | Preset 2: this will set the robotic hand to the second of the three preset positions |

| X 2 | Record 2:This sets the glove in record mode and will wait 5 seconds to take a snapshot of the user's hand to override the saved position for preset mode 2 |
|-----|--|
| | Preset 3: this will set the robotic hand to the third of the three preset positions |

| X 2 | Record 3: This sets the glove in record mode and will wait 5 seconds to take a snapshot of the user's hand to override the saved position for preset mode 3 |
|-----|---|
| X 3 | Turns all modes off, in this mode the hand just waits to be put in either mirroring mode or preset mode before moving the robotic hand |

| Requirements | Verification |
|---|--|
| The hall effects getting triggered should set the modes/try and move the hand to preset positions | 1 while designing the system touch the magnet to the hall effect and see in the code if the correct flags are set high |
| 2) The flex sensors bending causes the finger to move to correct location with ±5% | 1a) First through the software make sure we can see the change of resistance for the flex sensors and see the full range of motion b) then measure the angle of the finger and compare it to the angle of the robot. Ex of how the angle is measured: |

2.5 Subsystem 3: Power/Battery System.

The Power/Batter System will take a LiPo rechargeable battery and regulate the voltage/current to 3.3V so that our system functions. This is responsible for powering the Motion/Gesture Unit and the Processing Unit.

The power/battery system must be able to provide a $3.3V \pm 0.1$ to power the microcontroller and various sensors. The system's main power supply should be a rechargeable LiPo battery. The supply IC/battery should be able to supply at least 40 mA peak to allow for the system to operate(see power analysis). The battery should be able to power the glove system for a minimum of 8 hours.

| Requirement | Verification | |
|--|--|--|
| Able to power both the sensors and microcontroller for 8 hours in continuous transmission mode | We will test by placing the glove into mirroring mode which is the most intensive mode since the glove must continuously transmit data through bluetooth. Since our code transmits continuously regardless of movement in mirroring mode, we will set a timer and time how long it takes for the battery to drain in the glove. | |

2.6 Subsystem 4 (Prosthetic Hand):

This robotic hand will be used to demonstrate the effectiveness of our project and will interface with the Communication/Processing Unit via Bluetooth. The physical hand used to demonstrate capabilities of the project will communicate with the glove and microcontroller via bluetooth. 5 digits must be modeled and the angle of the finger will be adjustable. This hand will also have to respond to the 4 gesture commands such as prerecorded positions and mode switching. We plan to model our hand using an open source robotic hand from InMoove.[6.2] Since our project is focused on modeling the organic hand and mimicking the movement into the robotic hand, we opted to use an open source hand as a way to devote more time into the sensing of the hand rather than the design of the robotic prosthetic.

This robotic hand uses the MG996 Servo Motor as the robotic hand is designed to integrate them into the model, this servo is able to provide a maximum of 1.49 N*m of torque at

6 V. These servos will transmit movement through multi braided fishing line wrapped around a 3D printed pulley since the breaking point of the fishing line will be beyond what the servos can provide, and the multi braided line provides repeated abrasion resistance compared to single filament line. For the 3D print material of the hand we are using PLA or Polylactic Acid filament. Compared to similar materials such as PETG or ABS, PLA is much easier to work/print with, has a lower cost per kilogram, and overall has a higher rigidity. Though PETG and ABS have a higher temperature resistance before deformation, the temperature range we plan to use the robotic hand in will be well within the acceptable range for PLA since PLA begins to deform at 140 Degrees Fahrenheit. For the fasteners and hardware we plan to receive assistance from the ECE Machine Shop such as springs, screws, nuts, etc.

To control the hand we are using an Arduino Nano BLE which is used to control all the servos in the hand. This Arduino will then communicate with the sensor glove's microcontroller via bluetooth. The Arduino and Servos will be powered using 2x 3.7V, 18650 Li-ion Batteries placed in series in order to provide a 7.4 V for the Arduino and Servos.

| Requirements | Verification |
|--|--|
| Replicate hand position and commands given by sensor glove | Develop several test positions and quantify the angles of the fingers as defined in the sensing verification. Then measure the error of the robotic hand and compare against measured test points. |

2.7 Tolerance Analysis:

An aspect to our project that could lead to a large road block is trying to hit our goal of 8 hours of continuous usage. To tackle this problem we decided to base our tolerance analysis around calculating the estimated energy that our system requires in order to operate see Fig 4.1 and Fig 4.2.

First sensor power consumption was determined based on possible component's worst case operating voltages/current draws. For the CPU, an average current draw was found from the manufacturer's website. The bluetooth power was estimated based on the microcontroller manufacturer's transmission specs for -6dB operation.

Next to find energy we separated calculations by burst power draw(BLE) and continuous average power draw(everything else). To find BLE energy consumption we found the time for one transmission and multiplied it by the total number of transmissions per day to get total transmission time. The total transmissions per day was found by multiplying 8Hrs by 30Hz(the transmission rate of our system). Next, we added all the continuous power draw quantities and multiplied by 8 Hrs to find the energy consumption of non-BLE components. Finally, we accounted for inefficiency in the power regulation IC by multiplying the total energy consumption by 1.2(80% efficiency).

In the end, we found out that our system requires 3.2kJ to operate. Our LiPo will provide around 11kJ, meaning that we can meet the 8Hr operation specification easily with some room for error.

$$\frac{3.3N}{2} \sum_{\substack{k \le 151k \\ k \le$$



Unit case H for the 1 particle BLE Energy $53 \times 10^{-3} \left(\begin{array}{c} 4815 + 100 - 1000 \\ 1000 + 100 \end{array} \right) 1000 = 28.51 J + 10^{-3} \\ 1000 + 100 \\ 1000 + 100 \end{array} \right) + 1000 \\ + 1000$

Figure 2.7.2: power calculations

3 Cost and Schedule

3.1 Cost Analysis

Labor:

| \$/Hour | Estimated Hours | Employees |
|---------|-----------------|------------|
| 45 | 60 | 3 |
| | | |
| | Labor Cost | \$8,100.00 |

Parts For Glove:

| Part Name | Manufacturer | Model | Unit Price(\$) | # Units | Cost(\$) |
|---------------------------|---------------------------|-----------------------|----------------|------------|----------|
| lex Sensor Spectra Symbol | | FS-L-0055-253-ST | 7.95 | 5 | 39.75 |
| Glove Pair | ve Pair Home Depot | | 15 | 1 | 15 |
| Hall Effect | Texas Instruments | DRV5032AJDBZT | 1.45 | 4 | 5.8 |
| Magnet K&J Manetics | | D62 (K and J Magnetic | 0.61 | 1 | 0.61 |
| BLE Micro Adafruit | | 1528-2835-ND | 8.75 | 1 | 8.75 |
| 3.3V LDO | Diodes Incoporated | AP7387Q-33Y-13 | 1.55 | 1 | 1.55 |
| Schottky Diode | Fairchild | Bat54 | 0.31 | 1 | 0.31 |
| 7.4V LiPo | Blomik | 850mAh | 11.99 | 1 | 11.99 |
| 25k Resistors | TE Connectivity | CPF-A-0805B25KE | 0.48 | 5 | 2.4 |
| 330 Ohm Resistor | ROHM | SDR03EZPD3300 | 0.22 | 3 | 0.66 |
| 0.1uF Cap | Murata | GRT155R71C104KE01 | 0.1 | 8 | 0.8 |
| 1uF Cap | Kemet | C2220X105K2RACAUT | 0.5 | 2 | 1 |
| PCB | N/A | N/A | 5 | 1 | 5 |
| 5050 SMD LED | ROHM | CSL1901VW1 | 3 | 0.5 | 1.5 |
| | | | | Glove Cost | \$93.62 |

Parts For Robotic Hand:

| Part Name | Manufacturer | Model | Unit Price(\$) | # Units | Cost(\$) |
|------------------|--------------|-------------|----------------|------------|----------|
| Power Supply | MEAN WELL | 323282 | 15 | 1 | 15 |
| Servos | Towerpro | MG996R | 4.831 | 5 | 24.155 |
| Nano 33 BLE | Ardunio | Nano 33 BLE | 19.99 | 1 | 19.99 |
| Printer Filament | Polymaker | PLA | 10 | 1 | 10 |
| Fishing Line | Berkly Line | BGB20-22 | 10 | 1 | 10 |
| | | | | Robot Cost | \$79.15 |

Total Cost:

\$8100 (Labor) + \$172.77 (Parts) = \$8272.77

3.2 Schedule:

| Week | Objectives | Notes |
|------|---|---|
| 2/20 | Design Document complete, work on PCB layout, start 3d printing hand, work on firmware | All of us will work on the design document, Quang will start 3d printing the hand, Ryan will work on the PCB layout, Sohil will work on the firmware |
| 2/27 | Get PCB checked, work on Firmware, assemble 3D printed hand/ more 3D printing | Ryan will get V1 of the PCB checked, Sohil will have a dev board and can start validating some of the firmware, Quang will assemble the robotic hand/ 3d print more parts |
| 3/6 | Order PCB, continue working on Firmware, start testing robotic hand | This week we will order our PCB and work more on the firmware. We will also see if we can control the robotic hand by itslef (Ryan, Quang, and Sohil) |
| 3/13 | Spring break | N/A |
| 3/20 | Continue working on firmware | Should be finishing up the firmware for each specific subsystem (Ryan, Quang, and Sohil) |
| 3/27 | Assemble PCB's (main PCB for controlling and the PCB's that hold the halleffect sensors), continue working on firmware, print hall effect sensor mounts and other small 3d prints | Ryan will take point on assembling all the PCB's with help from Sohil and Quang. Quang will be in charge of making sure all the 3D printing is complete, Sohil is in charge of trying to integrate all the different parts of the firmware together |
| 4/3 | General Debugging | This week we will mainly be working on debugging all the issues with the firmware and if needed hardware debugging as well (Ryan, Quang, and Sohil) |
| 4/10 | General Debugging | Continue debugging issues as they come |

| | | up (Ryan, Quang, and Sohil) |
|------|---|--|
| 4/17 | Start working on Final Paper and continue debugging | Start working on final paper and continue debugging (Ryan, Quang, and Sohil) |
| 4/24 | Get demo ready | Run a demo by the TA |
| 5/1 | Work on Final Paper | Work on Final Paper |

4 Ethics and Safety

Our team made sure to comply with both the IEEE ethics and safety guidelines as well as the ACM code of ethics while detailing out our project. The safety of our users is our number one priority and we want to make sure that we are not abusing or using anyone in the creation and distribution of our product. This includes our users who will most likely be disabled people. Some issues we can see with the usage of our project is that it is a wearable so we want to make sure that the usage of our project doesn't cause harm to anyone. Since our sensor glove directly interfaces with a user's limb, there is a risk of injury from exposed circuitry and electric shock. We can ensure our users safety by correctly grounding all components, insulating exposed metal and leads, and using an electrically insulating material on the glove. In doing this we ensure the safety of our users. [6.1]

Furthermore to comply with section 1.5 in IEEE ethics and safety guidelines we have all parts of the project will be looked over by all team members, credit all sources we use/draw inspiration from, and will ensure the high quality of our work and keep each other responsible. Furthermore we will also be open to criticism and suggestions from our TA and our peers and make necessary adjustments to our methods, design, and practices if quality or a breach of ethics is called into question. This also relates with Section 1.3 in the ACM code of ethics where we will make sure to gain the knowledge or seek out help when working on technical aspects that we have never tried before, since a lack of experience working on systems and ideas can lead to dangerous hardware for both us as the creators and the final end users. Since our project involves many disciplines of engineering from mechanical design, communications, and coding, we will try our best to stick to our strengths to ensure the most qualified team member is working on the respective aspect of the project.[6.4]

Also according to IEEE 1.4, we must avoid unlawful activities both technically and professionally. In order to ensure that our wireless communication schemes are within compliance as required by the FCC on bluetooth, we will reference the manuals and datasheets provided with our microcontroller and arduino and if a fault is found within our use cases, we will consult an experienced mentor for guidance and conduct research in order to remedy the fault.

5 Appendix:

| Table 5-1. Device Comparison | | | | | | |
|------------------------------|----------------------|----------------------|----------------|------------------|-----------------------|--|
| VERSION | MAXIMUM THRESHOLD | MAGNETIC RESPONSE | OUTPUT TYPE | SAMPLING RATE | PACKAGES AVAILABLE | |
| DRV5032DU | 3.9 mT | Unipolar | Push-pull | 20 Hz | SOT-23, X2SON, TO-92 | |
| DRV5032FA | 4.8 mT | Omnipolar | Push-pull | 20 Hz | SOT-23, X2SON, TO-92 | |
| DRV5032FB | | Omnipolar | Push-pull | 5 Hz | SOT-23, TO-92 | |
| DRV5032FC | | Omnipolar | Open-drain | 20 Hz | SOT-23, TO-92 | |
| DRV5032FD | | Unipolar | Push-pull | 20 Hz | X2SON, TO-92 | |
| DRV5032AJ | 9.5 mT | Omnipolar | Open-drain | 20 Hz | SOT-23, X2SON, TO-92 | |
| DRV5032ZE | 63 mT | Omnipolar | Open-drain | 20 Hz | SOT-23, TO-92 | |

TI Application Notes [6.5] : Figure 5.1

We decided to choose a hall effect sensor with medium sensitivity. This is to allow us to use a relatively small magnet, but gives the sensor some noise immunity in case the user brushes by another magnetic source, and tries to negate multiple fingers triggering at the same time by requiring a stronger field to switch the hall sensors.

Use Equation 2 for the cylinder shown in Figure 9-2:

$$\vec{B} = \frac{B_r}{2} \left(\frac{D+T}{\sqrt{((0.5C)^2 + (D+T)^2)}} - \frac{D}{\sqrt{(0.5C)^2 + D^2}} \right)$$

TI Application Notes [6.5] : Figure 5.2



Details of the magnet D62 [6.6]: Figure 5.3

Based on a 9.5 mT maximum threshold of the DRV5032AJ. Based on a desired trigger distance D of around 0.25 inches, the magnet D62 was chosen with a Br of 13,200 Gauss and C = $\frac{3}{8}$ " and T = $\frac{1}{8}$ ". This magnet is small enough to fit on the back of the thumb.

Plugging D62 coefficients into the field strength equation yields 9.54 mT at 0.25 inches away, just enough to switch the hall effect sensor at the desired distance.

Power Considerations

LDO

Features

- Wide Input Voltage Range: 5V to 60V
- Maximum Output Current: 150mA
- Dropout Voltage:
 V_{DROP} = 700mV @Iout = 100mA (Typ.)
 V_{DROP} = 1100mV @Iout = 150mA (Typ.)
- Low Quiescent Current: 2µA (Typ.)
- High Output Voltage Accuracy: ±2%
- High PSRR: 70dB @1kHz
- Excellent Line/Load Regulation
- Thermal Shutdown Function
- Short Current Protection Function
- Output Current Limit

The LDO will help regulate the voltage coming out of the LiPo without causing too much of a voltage drop as the LiPo drains. This will allow us to support both a 4.2V and 7.4V LiPo depending on what we finalize the design to use.

The LDO has a short circuit protection circuit, and we are placing a schottky diode in series with the LDO input in order to provide reverse polarity protection from the lipo being plugged in with the wrong polarity. This should cover problems that could occur between the LiPo and Power Circuitry subsystem interface.'

Flex Sensors

BASIC FLEX SENSOR CIRCUIT:



Application Circuit from the flex sensor manufacturer. We will use a similar configuration in order to read the voltage from the flex sensor. Instead of the impedance buffer, we will be connecting the circuit to the inputs of the microcontroller ADC pins.

Microcontroller Module



This module includes the nRF51822 microcontroller with an antenna already inside. This module allows us to use the nRF51822 BLE microcontroller without having to worry about antenna design and meeting FCC regulations as this module comes with FCC certifications for the antenna. We will be programming the module using the Arduino Environment if we can, and if that doesn't work out will we be using Keil to handle the programming via C. We will flash the microcontroller using the STLINKv2 debugger/programmer.

6 References:

6.1 "IEEE code of Ethics," *IEEE*. [Online]. Available: https://www.ieee.org/about/corporate/governance/p7-8.html. [Accessed: 09-Feb-2023].

6.2 "Hand and Forarm," *InMoov*. [Online]. Available: https://inmoov.fr//hand-and-forarm/. [Accessed: 09-Feb-2023].

6.3 J. N. Ingram, K. P. Körding, I. S. Howard, and D. M. Wolpert, "The statistics of Natural Hand Movements," *Experimental brain research*, Jun-2008. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2636901/. [Accessed: 09-Feb-2023].

6.4 "The code affirms an obligation of computing professionals to use their skills for the benefit of society.," *Code of Ethics*. [Online]. Available: https://www.acm.org/code-of-ethics. [Accessed: 09-Feb-2023].

6.5 "DRV5032 Ultra-Low-Power digital-switch hall effect sensor datasheet ..." [Online]. Available:

https://www.ti.com/lit/ds/symlink/drv5032.pdf?HQS=dis-mous-null-mousermode-dsf-pf-null-w we&ts=1662340198537&ref_url=https%253A%252F%252Fwww.mouser.com%252F. [Accessed: 24-Feb-2023].

6.6 "D62," *K&J Magnetics: D62*. [Online]. Available: https://www.kjmagnetics.com/proddetail.asp?prod=D62. [Accessed: 23-Feb-2023].