|  |
| --- |
| Montana State University  Department of Mechanical and Industrial Engineering  ETME 499R CAPSTONE: MECHANICAL ENGINEERING TECHNOLOGY DESIGN II  and  EMEC 499R CAPSTONE: MECHANICAL ENGINEERING DESIGN II  Motion Capture Gloves  By Colter Girardot, Darren Jonathan, Dylan Raber, Luke Niess  For: Jacob Bernal, Jeffrey Kinkaid, Nadya Modyanova  Prepared to Partially Fulfill the Requirements for EMEC 499/ETME 499  Department of Mechanical Engineering Montana State University  Bozeman, MT 59715  May 4th, 2025 |

***Acknowledgements***

Our team would like to thank Dr. Modyanova for being an amazing sponsor and contact for our project, for keeping our spirits high, and for giving us the insight and perspective on how best to tackle this project.

We would also like to thank Mr. Hunter Caldwell for all the time and effort he put into giving us valuable feedback and assessment while also giving us opportunities to improve upon our work.

Finally, we would like to give thanks to the professors of this course, Prof. Jacob Bernal and Prof. Jeff Kinkaid. Without their guidance, a project like this would be a great deal more difficult to plan out and execute, especially given our lack of experience in real-world project management.

***Executive Summary***

Autism Spectrum Disorder (ASD) is a neurological and developmental disorder that primarily affects one’s cognitive and communication skills. Research is still being conducted on how people with ASD interpret the methods with which we communicate. As such, our project goal is to develop technology to assist with research on what effect hand gestures have on communication and comprehension for those with ASD. The technology being developed is an Arduino-based motion capture (mo-cap) glove, capable of locating, tracking, and sending out a user’s hand movement data in free space. The nature of this research dictates that the glove material be washable and disinfectable, as autoimmune diseases are more prevalent in people with ASD.

The gloves will be used in a plethora of environments, both rural and urban; as such the gloves must be accessible to those in areas with less access to technology resources. This dictates that the design be easy to assemble and repair, with parts also being easy to replace.  Moreover, off-the-shelf options tend to come at a heightened cost with fewer options for data customization, furthering the need for our design to be inexpensive and easy to use.

The design process consisted of various rounds of research and analysis to optimize a current model that will be significantly cheaper and simpler than most other commercial designs. Project management procedures like consistent team & sponsor meetings, task lists, and coordination on overall design decisions allowed each of us to keep each other responsible and to work within our specialized fields without losing focus on the overall design being tested. Throughout the design process, various procedures – such as Pugh charts and a top-down preliminary analysis approach – were used to determine the most optimal components for our design. Preliminary analysis included calculations of specifications such as device current draw, data sampling rate, and power usage. Analysis of the mechanical components included calculations of glove weight, tensile strength, and adhesive strength. At the top level, each glove will consist of 6 main components: the microprocessor, the finger-tip sensors, the finger-knuckle sensors, the glove itself, the power supply, and the communication method.

The microprocessor will serve as the brains of the operation, communicating with the hand-sensors and the researcher’s external device to gather and send hand-movement data. The IMU (Inertial Measurement Unit) finger-tip sensors will track each finger’s movement in 6 axes (X, Y, Z, Roll, Pitch, and Yaw). Additionally, the microprocessor has its own IMU, allowing for hand/palm tracking as well. The finger-knuckle flex sensors will track each finger’s “flex”, or how bent it is. The glove itself will consist of two layers, a vinyl glove layer to ensure cleanliness inside the glove, and an outer layer made up of mainly polyester and spandex, which will house the electronics. The power supply for our initial design will come from an external device via USB connection, so no on-board power-supply will be required. The USB connection can also be used to communicate to the researcher’s device, but a wireless option – in the form of Bluetooth Low Energy (BLE) – may also be tested later in development to allow for wireless operations.

The mo-cap gloves’ main objective is to send hand movement data to an external device. The general process for this starts with the hand sensors collecting movement data. The microprocessor then interprets that data, formats it into a readable data package, and sends it off to the external device (in real time). This project’s requirements do *not* include creating a program or app that can display the data in real time (such as a 3D representation of the hand(s) being measured), although the creation of an app or program that could do this may be considered as a stretch goal later in the project timeline.

Table of Contents

Acknowledgements ………………………………………………………………………………I

Executive Summary ……………………………………………………………………………...II

[Table of Contents IV](#_Toc197291396)

[Revisions 2](#_Toc197291397)

[Final Test Plan with Results 3](#_Toc197291398)

[Final Specifications 12](#_Toc197291399)

[Appendices 14](#_Toc197291400)

[Appendix A: Updated Drawing Package 14](#_Toc197291401)

[Appendix B: Updated Manufacturing Plan 26](#_Toc197291402)

[Appendix C: Other (Miscellaneous) 30](#_Toc197291403)

[Fully Updated Project Schedule 30](#_Toc197291404)

[Final Prototype Glove Picture(s) 30](#_Toc197291405)

[Arduino Code 32](#_Toc197291406)

[Python Code 37](#_Toc197291407)

[Appendix D: User Manual 45](#_Toc197291408)

[What’s Included? 45](#_Toc197291409)

[Safety and Handling 45](#_Toc197291410)

[Setup Instructions 45](#_Toc197291411)

[Glove Calibration 51](#_Toc197291412)

[Assembly Instructions 52](#_Toc197291413)

[How to Use 54](#_Toc197291414)

## Revisions

**Analysis:**

For our analysis, our estimations for overall glove weight ended up being an overestimation (to our benefit fortunately). Weight measurements can be found in the test plan section. For our data rate calculations, our overall sampling rate is limited to around 25 Hz. Our test plan and poster say that we were unable to get a sampling rate above 1.2 Hz, but luckily we were able to resolve this issue after those documents were submitted. Other than that there were no major discrepancies between our analysis and what was recorded.

For manufacturing, there were no major issues with putting together the glove. The one caveat to this being that the lack of a PCB means that there is more of a potential of wires becoming disconnected with prolonged use, especially during disassembly and reassembly. In future iterations of the glove, designing a PCB for the main housing circuitry is recommended to simplify the wiring (and potentially slim down the main housing as well).

**Manufacturing Plan:**

Any and all updates to the manufacturing plan can be found in the manufacturing plan section, which is in Appendix B.

**Drawing Package:**

Any design changes to the drawing package can be found in Appendix A.

**Test Plan/Results:**

Code was optimized, allowing increased sample rate which makes the sample rate test a PASS. This and minor adjustments to the number of team members present at certain tests has been reflected in the final test plan below.

# Final Test Plan with Results

* Weight (done week of 4/7)
  + Goal: To ensure the gloves are no greater than a mass of .5 kg
  + Procedures: weigh entirety of completed build on scale
  + Data acquired: Mass of glove
  + Schedule/Sequencing: Upon a completed build with a housing the team likes, the glove is to be weighed
  + Equipment/Facility: Both gloves required. Scale in makerspace with accuracy up to a tenth of a gram.

**Results:** Each glove totaled a mass of around 90-100 grams. In total the gloves were measured to be around 184 grams. This ensures our design is far below the maximum mass value of .5 kg. The mass observed is based on the components used in the current build. **PASS**

* Adhesive Strength (done week of 4/21)
  + Goal: to ensure housings will not detach under velocities experienced with common hand gestures.
  + Procedures: Wave hand in motion at increasing speeds to observe reactions on the housings.
  + Data Acquired: We are searching for the possibility of a max velocity that the user can move at before detaching the housing. It is ideal that all movements can be made, and that a max speed for detachment is beyond the capabilities of a human. Results will be denoted as pass/fail.
  + Schedule/Sequencing: Find various individuals to complete trials. Minimum of 3. If the product fails, adhesive locations will be altered and retested using the same individual
  + Equipment/Facility: Both gloves required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Completed in a private room with at least one team member present and one individual. This is to ensure proper safety and confidentiality following the team’s submitted IRB form.

**Results:** Instructed hand movements of various velocities both directional and rotational were carried out across multiple tests with no noticeable deterioration or detachment of the housings. **PASS**

* Sampling Rate (done week of 4/21)
  + Goal: Verify a sampling rate of 5-50Hz.
  + Procedures: Set sampling rate to one of the following values: 5 Hz, 10 Hz, 25 Hz, 50 Hz. Begin a timer and generate movement within the device. Stop the timer and observe timestamps within the recorded .csv file.
  + Data acquired: Data will be pass/fail with respect to observation of the timestamps. Each sensor should generate 5 data points per second at a minimum.
  + Schedule/Sequencing: Upon completing the adhesive strength test, individuals will be asked to move in a basic, light motion for 30 seconds. The timer will then be stopped and conclusions drawn from .csv files.
  + Equipment/Facilities: At least 1 glove required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Completed in the same location as adhesive test under same confidentiality regulations.

**Results:** After code optimization, the gloves can sample up to 25 Hz. **PASS**

* Disassembly Capabilities (done week of 4/21)
  + Goal: Verify build is simple, by design, to that of a non-engineering major
  + Procedures: Begin a timer and have the individual disassemble the glove.
  + Data Acquired: The individuals will be prompted to complete a questionnaire following the tests. Within this pertains questions to assembly difficulty.
  + Schedule/Sequencing: Uses the same individual as the previous tests and will be completed within the same 30-minute time frame.
  + Equipment/Facility: At least one glove required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Completed in the same location as adhesive test under same confidentiality regulations.

**Results:** Due to unforeseen issues with regard to participant surveying, as well as time constraints during participant testing, this test was not conducted. **FAIL**

* Storage Size (done week of 4/21)
  + Goal: To confirm that the glove can store up to 1 MB of movement data in flash memory.
  + Procedures: Start with the glove fully functioning (i.e. plugged into a computer with the computer running the data collection program). After confirming that data acquisition is functioning properly, close the data collection program. Continue to move the glove for 1 minute. Unplug the glove’s USB connection. Plug the glove back into the computer. Tell the program to collect stored data. Confirm that the glove was still recording data with appropriate values & time stamps.
  + Data Acquired: Data will be a pass/fail with regards to whether or not data was still being collected after the computer program was closed. Data will be collected on how long it takes to fill up the 1 MB of flash memory with the given sampling rate (between 5-50 Hz).
  + Schedule/Sequencing: This test will be done with the team only and will be completed separately from other tests.
  + Equipment/Facility: At least one glove required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Testing will be done in the Makerspace at one of the tables.

**Results:** Due to time constraints, as well as the priority assessment given to backup storage, the code has not been optimized for storage capabilities. As a result, there are currently no on-board storage backups in place in the event of a disconnect, although the data collected up to the point of disconnect will still be recorded and sent to an Excel doc. **FAIL**

* Calibration Setup Timing (done week of 4/21)
  + Goal: To ensure that calibration is a simple process with response times of <1 second.
  + Procedures: Start with glove not plugged into computer. Open computer program for data reading. Plug glove into computer via USB connection cable. Press the calibrate button and wait for the program to respond with the message “Place your hand on a flat surface with fingers spread, then press next”. Press the button labeled “Next” and start the timer. Stop the timer when the program responds with the message “Now, make a fist with your thumb tucked, then press next”. Confirm that timer reads less than one second. Press the button labeled “Next” and start the timer. Stop the timer when the program responds with the message “Calibration Finished”. Confirm that the timer reads less than one second.
  + Data Acquired: Time between when calibration button is pressed and when calibration is completed (happens twice). The test data will be pass/fail.
  + Schedule/Sequencing: This test will be done with the team only and will be completed separately from other tests.
  + Equipment/Facility: At least one glove required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Testing will be done in the Makerspace at one of the tables.

**Results:** Calibration times were timed to be far less than 1 second for each calibration step. **PASS**

* Calibration reliability (done week of 4/21)
  + Goal: to ensure that glove data is self-consistent
  + Procedure: Test gloves one at a time. Start with glove not plugged into computer. Place one hand palm down on a flat surface with fingers spread. Open computer program for data reading. Plug glove into computer via USB connection table. Run through calibration process and begin data acquisition (do not move the glove). Stop data acquisition after 5 seconds and average the values stored in the Excel file. Repeat test three times, completing three power-on cycles, all without moving the hand with the glove under test. Calculate the percentage error range using all collected, averaged data samples.
  + Data Acquired: Percentage error in the glove system, given identical inputs
  + Schedule/Sequencing: This test will only need 1 person on the team for completion, and as such can be done separately from the other tests.
  + Equipment/Facility: At least one glove required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Testing will be done in the Makerspace at one of the tables.

**Results:** At a standstill, the gloves still measure gravity. As such, when the glove is still an absolute value of the acceleration in all 3 directions should add up to somewhere around 9.8 m/s2. Using this constant to measure error, the average error was calculated to be ~0.1 m/s2. This yields a percent error of around **1%**. **PASS**

* Data Precision (done week of 4/21)
  + Goal: to confirm that the data being collected will not rail against the maximum values allowed for the IMU’s accelerometer and gyroscope.
  + Procedure: Ensure all IMU sensors are set to highest precision mode in the Arduino C code. Place hand flat on table with fingers spread. Open computer program for data reading. Plug glove into computer via USB connection cable. Complete calibration process. Move the glove around in all directions, ideally with very accelerated movements. Now, rotate the glove in all comfortable directions, ideally with very accelerated rotations. After this, stop data collection and unplug the glove.
  + Data Acquired: Glove data maximum positional acceleration and angular velocity will be compared to maximum values set in IMU precision code. Test passes if data collected does not reach maximum value set in code.
  + Schedule/Sequencing: This test will only need 1 person on the team for completion, and as such can be done separately from the other tests.
  + Equipment/Facility: At least one glove required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Testing will be done in the Makerspace at one of the tables.

**Results:** Even with hand movements far outside the comfortable range of motion during gesturing, the gloves did not hit their maximum acceleration and gyroscope readings. **PASS**

* Comfort (done week of 4/21)
  + Procedure: While wearing Arduino gloves, flex and unflex each finger on one hand individually, starting with the thumb and ending with the pinky finger. Repeat five times. Immediately after completion, rate the comfort of these motions on a scale from 1-10. Repeat with other hand.
  + Data Acquired: Average comfort score on a scale from 1-10, as reported by users
  + Schedule/Sequencing: Find various individuals to complete trials. Minimum of 3. This test does not produce applicable data for any other tests and can be done separately.

**Results:** Average comfort rating: 8.33. Lowest comfort rating: 8. **PASS**

* User Manual Readability (done week of 4/21)
  + Goal: To ensure a basic user manual interface including ways the glove may malfunction and how to solve such problems.
  + Procedures: Have individual follow manual with no external help
  + Data Acquired: The individual is to give feedback upon completing reading of the manual. They will assess font size, diagram usage, etc. To help guarantee the user manual is easily understood
  + Schedule/Sequencing: Uses the same individual as the previous tests (adhesive/disassembly) and will be completed within the same 30-minute time frame.
  + Equipment/Facility: At least one glove required, in addition to the user manual

**Results:** Due to unforeseen issues with regard to participant surveying, as well as time constraints during participant testing, this test was not conducted. While this doesn’t necessarily mean that the user manual is hard to follow or unreadable, it is hard to judge without outside input from non-engineers. **FAIL**

* Hypoallergenic Confirmation (done week of 4/21)
  + Goal: to confirm the glove can be worn by any individual and is indeed, hypoallergenic
  + Procedures: Verify manufacturer specifications for the allergen.
  + Data Acquired: We are attempting to find a direct link between skin irritation and the glove material used. This includes both the inner liner used, and the spandex/polyester glove itself. This will be denoted as pass fail
  + Schedule/Sequencing: Find various individuals to complete trials. Minimum of 3. This test does not produce applicable data for any other tests and can be done separately. The individual will be asked about any known allergies to clothing products.
  + Equipment/Facility: At least 1 glove required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Completed in the same location as adhesive test under same confidentiality regulations.

**Results:** Amongst our test participants, none experienced or noted a negative reaction to the glove material. The participants felt that the glove material itself was soft, breathable, and comfortable. **PASS**

* LED Reading (done week of 3/31)
  + Goal: to confirm that the glove component’s LEDs light up properly when powered and connected.
  + Procedures: Start with glove not plugged into computer. Open computer program for data reading. Plug glove into computer via USB connection cable. Check each individual IMU to see if their red LED lights up. Undo the main housing and check to see if the multiplexer’s green LED lights up. Lift the multiplexer and check to see if the Arduino’s green LED lights up.
  + Data Acquired: No direct data from the glove will be captured. The test data will be pass/fail: pass if the LED lights up correctly, fail if otherwise.
  + Schedule/Sequencing: This test will only need 1 person on the team for completion, and as such can be done separately from the other tests.
  + Equipment/Facility: Both gloves required. Computer with Windows 10/11 or MacOS 10.14 (or later) required.

**Results:** LED lights in the IMUs, multiplexer, and Arduino microcontroller all lit up upon plugging the glove’s USB cable into a computer/power source. **PASS**

* Incoming Voltage (done week of 4/7)
  + Goal: To ensure that the glove’s operating voltage is 5V or lower. To confirm that the 5V DC Breakout board has an output voltage tolerance of ±2%. To confirm that all electrical components share a common ground with minimal resistance (<0.1 ohms).
  + Procedures: Start with glove not plugged into computer. Open computer program for data reading. Plug glove into computer via USB connection cable. Attach multimeter positive cable to 5V output pin on USB breakout and attach multimeter negative cable to GND output pin on USB breakout. Record multimeter voltage reading. Move multimeter positive cable to GND output pin and change multimeter reading unit to ohms. Record multimeter resistance reading.
  + Data Acquired: 5V breakout pin actual voltage and GND breakout pin resistance. The test data will be pass/fail.
  + Schedule/Sequencing: This test will only need 1 person on the team for completion, and as such can be done separately from the other tests.
  + Equipment/Facility: Both gloves required. Computer with Windows 10/11 or MacOS 10.14 (or later) required. Multimeter that can measure at least 5 volts and has at least 2-digit decimal precision for resistance required. Testing will be done at either the Makerspace with the Makerspace’s Multimeter or will be done at the Electrical Engineering Lab in 601 Cobleigh Hall with one of the multimeters there.

**Results:** Incoming voltage during regular operation was measured to be between 4.98 and 5 volts, well within our voltage tolerance expectations. Ground resistance during regular operation was measured below 0.01 Ohms, well below our maximum resistance expectations. **PASS**

* Data & Power Confirmation (done week of 4/7)
  + Goal: To confirm that the micro-USB to USB type-A wired connection functions properly between the gloves and a computer running Windows 10/11 or MacOS 10.14 (or later).
  + Procedures: Start with glove not plugged into computer running Windows 10. Open computer program for data reading. Plug glove into computer via USB connection cable. Confirm that power connection has been established through the glove’s LEDs. Confirm that data connection has been established by starting data collection. Stop data collection and unplug glove. Repeat these steps with a computer running Windows 11 and a computer running MacOS 10.14 (or later).
  + Data Acquired: No direct data from the glove’s movement will be recorded here. The test data will be pass/fail with regards to each operating system test.
  + Schedule/Sequencing: This test will only need 1 person on the team for completion, and as such can be done separately from the other tests.
  + Equipment/Facility: Both gloves required. Computer with Windows 10 required. Computer with Windows 11 required. Computer with MacOS 10.14 (or later) required. Testing can be done wherever possible/convenient.

**Results:** The tests involved plugging the glove in and confirming that the circuit obtains power (confirmation via LED). Using the created Python UI, calibration and data collection processes were initiated and then subsequently stopped. Confirmation of data collection was observed by viewing the output .csv file. This process was conducted successfully using 3 different machines running 3 different operating systems – Windows 10, Windows 11, and MacOS 15 (Sequoia). **PASS**

* Wire/Cable Inspection (done week of 3/31)
  + Goal: To confirm that the cables and wires being used are the correct gauge and/or the correct type. To confirm that the JST connectors are rated for >1000 mating cycles.
  + Procedures: Confirm by inspection that the cables being used for flex sensor connections are 30 AWG wires. Confirm by inspection that the USB cable being used for glove-to-computer connection is a micro-USB to USB type-A cable. Confirm that the USB cable has a shielding efficiency of >85% through the cable’s datasheet. Confirm that the JST connectors used with the IMUs are rated for >1000 mating cycles through the connectors’ datasheet.
  + Data Acquired: No direct data from the glove’s movement will be recorded here. The test data will be a total of 4 pass/fail results. Each wire/cable specification will have a pass/fail result.
  + Schedule/Sequencing: This test will only need 1 person on the team for completion, and as such can be done separately from the other tests.
  + Equipment/Facility: Both gloves required. Computer that can look up component datasheets required. Testing can be done wherever possible/convenient.

**Results:** Cables and wires are intact with proper connections as needed. The wires have been soldered safely and correctly. It is recommended to check wiring before each use to avoid incorrect data. **PASS**

* Cost Inspection (done week of 4/7)
  + Goal: To confirm that the cost of the components matches what was estimated for the project.
  + Procedures: Confirm that the total price for the flex sensors is less than or between $80 to $90. Confirm that the total price for the IMU sensors is less than or between $150 to $170. Confirm that the total price for the Arduino microcontrollers is less than or between $55 to $65. Confirm that the total price for the USB breakout boards is less than or between $40 to $45. Confirm that the total price for the USB cables is less than or between $10 to $20. Confirm that the total price for the glove materials is less than or between $20 to $30. Confirm that the total price for extra/miscellaneous components is less than or between $120 to $130. Confirm that the total budget is at or below $700.
  + Data Acquired: No direct data from the glove’s movement will be recorded here. The test data will be a total of 8 pass/fail results. Each cost specification will have a pass/fail result. If a result is a failure, the actual cost will be listed.
  + Schedule/Sequencing: This test will only need 1 person on the team for completion, and as such can be done separately from the other tests.
  + Equipment/Facility: Computer that can refer to our budget document required. Testing can be done wherever possible/convenient.

**Results:** Individual test results:

Flex Sensors: ($12.95 \* 8) + ($7.95 \* 2) = $119.50 X (Over by $29.50)

IMU Sensors: $14.95 \* 10 = $149.50 ✓ (Under by $0.50)

Arduino Microcontrollers: $26.80 \* 2 = $53.60 ✓ (Under by $1.40)

USB Breakout Boards: $1.95 \* 3 = $5.85 ✓ (Under by $34.15)

USB Cables: $5.79 \* 2 = $11.58 ✓ (Within Price Range)

Glove Materials: ($18.98 \* 1) + ($7.99 \* 1) = $26.87 ✓ (Within Price Range)

Extra Miscellaneous Components: $62.32 ✓ (Under by $57.68)

* STEMMA Male Headers Cable = ($0.95 \* 12) = $11.40
* Permatex Silicone Adhesive = ($5.04 \* 1) = $5.04
* 8 Channel I2C Multiplexer = ($6.95 \* 2) = $13.90
* Perfboard = ($6.99 \* 1) = $6.99
* Wires = ($9.99 \* 1) = $9.99
* 3D Printing Materials (Estimated) = ($15 \* 1) = $15.00

Total Budget Needed to Reconstruct Gloves: $429.22 ✓

* Note: There may be additional costs if soldering can’t be done in-house

Overall: Components were purchased or 3D printed in house on campus. While there were a few budget estimations that were over-budget, the total amount for the 2 gloves’ components arrived at $429. The goal was to maintain a construction budget under $700. **PASS**

## Final Specifications

***Functional / User-Driven Specifications***

* Set-up time for users is under 5 minutes.
* Each glove weighs less than 0.5 kg for extended comfort.
* Material is hypoallergenic, breathable, and allows for unrestricted movement.
* Gloves are washable with removable electronics for easy cleaning.
* Each glove includes LED indicators for power and connection.
* Calibration is performed using a single software button for simplicity.
* Operates at a voltage of 5V or lower, with integrated overcurrent protection.
* Features Micro USB to USB type A wired plug-and-play cable for data transfer and power supply.
* Adjustable design ensures a secure fit for different adult hand sizes.

***Performance Specifications***

* Gloves capture tilt (±180°), angular speed (up to 200°/s), and acceleration (up to 16g) for precise motion tracking.
* Supports a minimum sampling rate of 5 Hz, adjustable up to 25 Hz for higher fidelity motion data.
* Include onboard real-time processing for immediate data filtering and compression to optimize performance.
* Sensors are rated to withstand >100,000 flex cycles without functional degradation, ensuring long-term durability.

***Electrical and Wiring Specifications***

* Uses 30 AWG stranded wire for sensor connections, supporting up to 0.9A at 5V, and 26 AWG for power lines, supporting up to 2.2A at 5V.
* Operate through Micro USB 3.0 with a power supply capability of 5V/900mA and a data transfer rate of up to 480 Mbps (USB 2.0 fallback for stability).
* Includes a 5V DC breakout board with a voltage tolerance of ±2% to maintain stable voltage across components.
* Ensures all electronic components share a common ground with a maximum resistance of <0.1 ohms for signal consistency.
* Uses shielded cables with a shielding efficiency of >85% to minimize electromagnetic interference.
* Incorporates JST connectors rated for >1,000 mating cycles to allow modularity and easy replacement of components.

***Interface Specifications***

* 6-foot USB type A to micro-USB 3.0 cable for wired data transfer with a power supply capability of 5V/900mA.
* Includes RGB LED indicators with a brightness level of 20-30 lumens to display power and connectivity.
* User manual will contain protocols and solutions for the ways in which the gloves can malfunction.
* Interface software supports Windows 10/11, MacOS 10.14 or later, and includes compatibility for USB 3.0.
* Includes a single calibration button in the software with a response time of <1 second for quick setup.

***Cost Specifications***

* $120 for flex sensors (10 pieces).
* $150 for IMU sensors (10 pieces).
* $55 for Arduino microcontroller (2 pieces).
* $6 for Micro USB breakout module for power and data (2 pieces).
* $12 for USB type A to micro-USB cable (2 pieces).
* $30 for glove materials (1 pair).
* $70 for miscellaneous components (connectors, wires, casing, adhesive).
* Total budget of around $450.
  + Potentially closer to $500 depending on access to soldering equipment.

# Appendices

## Appendix A: Updated Drawing Package

A diagram of a computer component

AI-generated content may be incorrect.

Figure 01: MoCap component diagram tree

All parts manufactured in-house should follow the same nomenclature as seen above. MCG\_ indicating (MoCap glove), value in the hundreds place indicates the file type: 200 – main assembly, 100 – sub-assembly, and 000 – single component. The tens and ones place are dedicated solely to which part you are working with. Drawings for the modeled component can be seen below.

Following the drawings are ECNs (Engineering Change Notification), created throughout the manufacturing process. The changes are direct products of trial and error upon printing the housing for the components. Each change is linked to a component and can be found in the upper right of the ECN document

A blueprint of a mechanical hand

AI-generated content may be incorrect.

A drawing of a box

AI-generated content may be incorrect.

A blueprint of a piece of metal

AI-generated content may be incorrect.

A blueprint of a device

AI-generated content may be incorrect.

A blueprint of a box

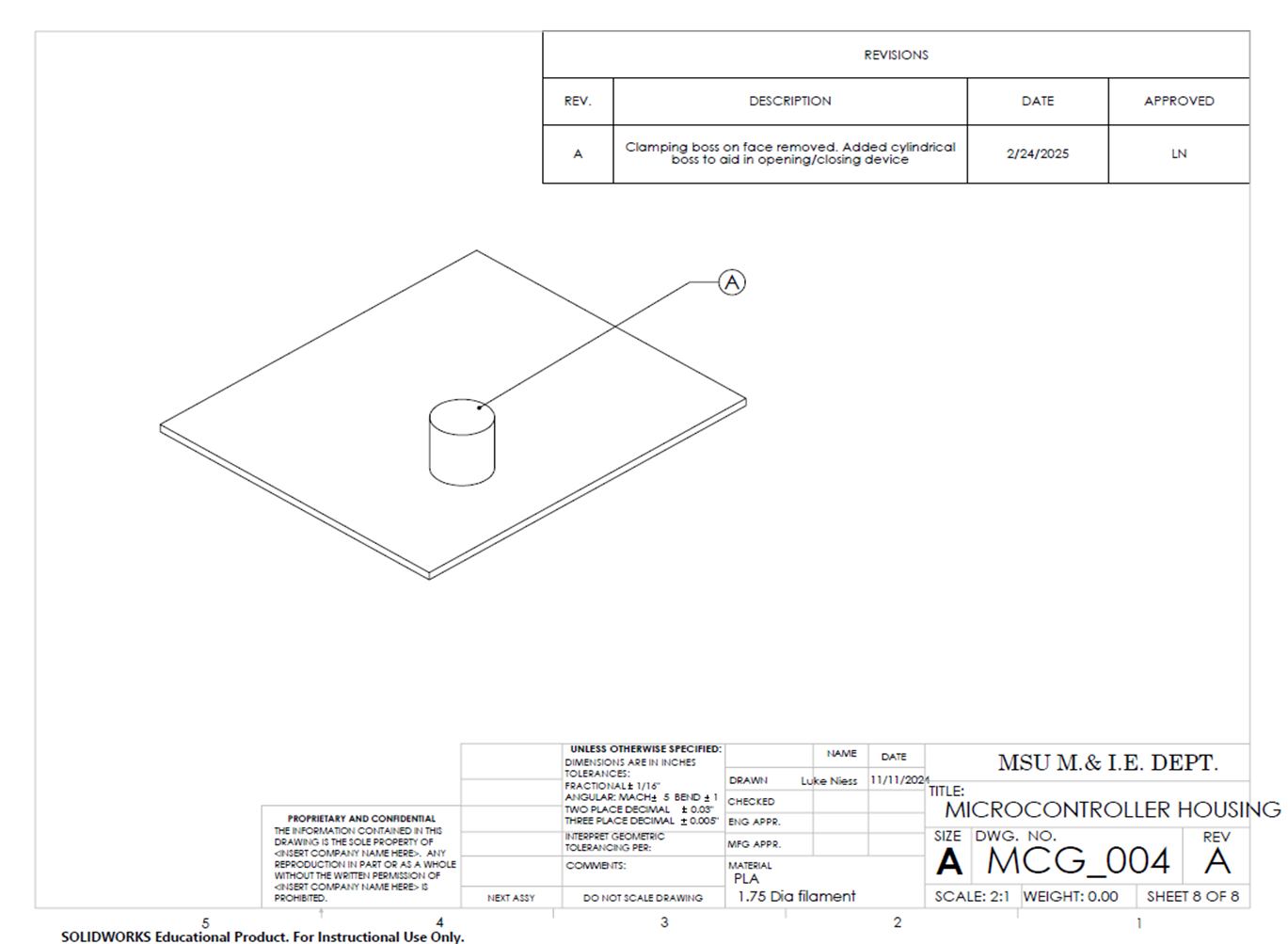
AI-generated content may be incorrect.

A computer drawing of a box

AI-generated content may be incorrect.

A blueprint of a square and a square

AI-generated content may be incorrect.



A screenshot of a document

AI-generated content may be incorrect.

A screenshot of a computer application

AI-generated content may be incorrect.

A black and white form with black lines

AI-generated content may be incorrect.

A close-up of a form

AI-generated content may be incorrect.

## Appendix B: Updated Manufacturing Plan

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Route Sheet: T38P38 Motion Capture Gloves | | | | | | |
| Part No.  MCG\_001 | Part Name  IMU housing | | Planner  Luke Niess | Checked by:  Darren Jonathan | | Date  11/8/2024 |
| Material  PLA | Stock size  1.75 mm | | Comments: | | | |
| No. | Operation description | DEPT | Machine | | Tooling | Length of op |
| 10 | Obtain material | Stock room |  | | NA | 2 mins |
| 20 | Load STL and material | 3D printer | PRUSA MK3.9 | | NA | 10 mins |
| 30 | Print material | 3D printer | PRUSA MK3.9 | | NA | 10 mins |
| 40 | Remove print/clean edges | Inspection |  | | NA | 10 mins |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Route Sheet: T38P38 Motion Capture Gloves | | | | | | |
| Part No.  MCG\_002 | Part Name  Flex senser Fixture | | Planner  Luke Niess | Checked by:  Darren Jonathan | | Date  11/9/2024 |
| Material  PLA | Stock size  1.75 mm | | Comments: create 10 for final assembly. Confirm tolerance of print during inspection. Op lengths are estimates based on experience | | | |
| No. | Operation description | DEPT | Machine | | Tooling | Length of op |
| 10 | Obtain material | Stock room |  | | NA | 2 mins |
| 20 | Load STL and material | 3D printer | Markforged Mark 2 | | NA | 10 mins |
| 30 | Print material | 3D printer | Markforged Mark 2 | | NA | 10 mins |
| 40 | Remove print/clean edges | Inspection |  | | NA | 10 mins |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Route Sheet: T38P38 Motion Capture Gloves | | | | | | |
| Part No.  MCG\_003 | Part Name  Microprocessor housing(base) | | Planner  Luke Niess | Checked by:  Darren Jonathan | | Date  11/9/2024 |
| Material  PLA | Stock size  1.75 mm | | Comments: Create 2 for final assembly, confirm tolerance of print during inspection. Op lengths are estimates based on experience | | | |
| No. | Operation description | DEPT | Machine | | Tooling | Length of op |
| 10 | Obtain material | Stock room |  | | NA | 2 mins |
| 20 | Load STL and material | 3D printer | Markforged Mark 2 | | NA | 10 mins |
| 30 | Print material | 3D printer | Markforged Mark 2 | | NA | 30 mins |
| 40 | Remove print/clean edges | Inspection |  | | NA | 10 mins |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Route Sheet: T38P38 Motion Capture Gloves | | | | | | |
| Part No.  MCG\_004 | Part Name  Microprocessor housing(top) | | Planner  Luke Niess | Checked by:  Darren Jonathan | | Date  11/9/2024 |
| Material  PLA | Stock size  1.75 mm | | Comments: create 2 for final assembly. Confirm tolerance of print during inspection. Op lengths are estimates based on experience | | | |
| No. | Operation description | DEPT | Machine | | Tooling | Length of op |
| 10 | Obtain material | Stock room |  | | NA | 2 mins |
| 20 | Load STL and material | 3D printer | Markforged Mark 2 | | NA | 10 mins |
| 30 | Print material | 3D printer | Markforged Mark 2 | | NA | 10 mins |
| 40 | Remove print/clean edges | Inspection |  | | NA | 10 mins |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Part No.  MCG\_101 | Part Name  Microprocessor housing assembly | | Planner  Luke Niess | Checked by:  Darren Jonathan | | Date  11/9/2024 |
| Material  PLA | Stock size  NA | | Comments: Confirm components in housing are wired safely/correctly | | | |
| No. | Operation description | DEPT | Machine | | Tooling | Length of op |
| 10 | Obtain printed housing components(top and base) | Inspection | NA | | NA | 2 mins |
| 20 | Place microprocessors, multiplexor and usb breakout in housing. | Assembly | NA | | NA | 1 min |
| 30 | Locate .75 mm thru hole on top housing to .75mm extruded pin | Assembly | NA | | NA | 1 min |
| 40 | Lower lid to smaller pin diameter and slide to .25 mm thru hole | Assembly | NA | | NA | 1 min |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Route Sheet: T38P38 Motion Capture Gloves | | | | | |
| Part No.  MCG\_201 | Part Name  Microprocessor housing assembly | | Planner  Luke Niess | Checked by:  Darren Jonathan | Date  11/9/2024 |
| Material  PLA | Stock size  NA | | Comments: “desired location” in op 20 decided through testing | | |
| No. | Operation description | DEPT | Machine | | Length of op |
| 10 | Obtain glove and housing units and all electronic components | Inspection | NA | | 2 mins |
| 20 | Locate IMU, microprocessor, and flex sensor housing to desired location | Assembly | NA | | 5 mins |
| 30 | Use silicon adhesive to attach housing components to the glove, allow to dry completely | Assembly | NA | | 24 hrs |
| 40 | Microprocessor, multiplexer, and usb breakout are to be wired and inserted into respective housing locations | Assembly | NA | | 20 mins |
| 50 | IMU’s are placed in housing and wired to multiplexer | Assembly | NA | | 15 mins |
| 60 | Flex sensors are placed in housing fixture and wired to multiplexer | Assembly | NA | | 10 mins |
| 70 | USB cable is connected to breakout and CPU for power | Assembly | NA | | 2 mins |
| 80 | Device is used as instructed | Research | NA | |  |

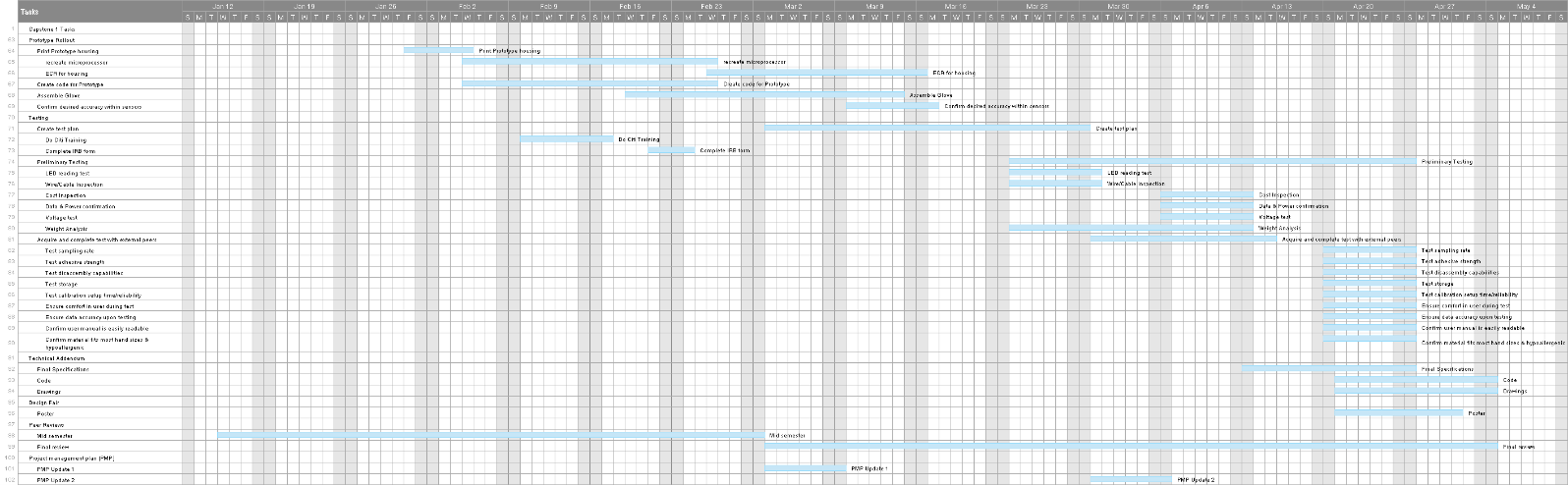
## Appendix C: Other (Miscellaneous)

This appendix contains many miscellaneous components of the Technical Addendum, including our fully updated project schedule (in the form of a Gantt chart), pictures of our final prototype gloves, our Arduino and Python code, and our user manual.

### Fully Updated Project Schedule

A graph with lines and dots

AI-generated content may be incorrect.



### Final Prototype Glove Picture(s)



### Arduino Code

The following is the code used for each of the gloves. The only difference between code for each glove is in line 120, where “L” designates data for the left hand, and “R” designates data for the right hand.

A screenshot of a computer screen

AI-generated content may be incorrect.A screen shot of a computer

AI-generated content may be incorrect.

A screen shot of a computer program

AI-generated content may be incorrect.A screen shot of a computer code

AI-generated content may be incorrect.A screen shot of a computer code

AI-generated content may be incorrect.A screen shot of a computer code

AI-generated content may be incorrect.A screenshot of a computer program

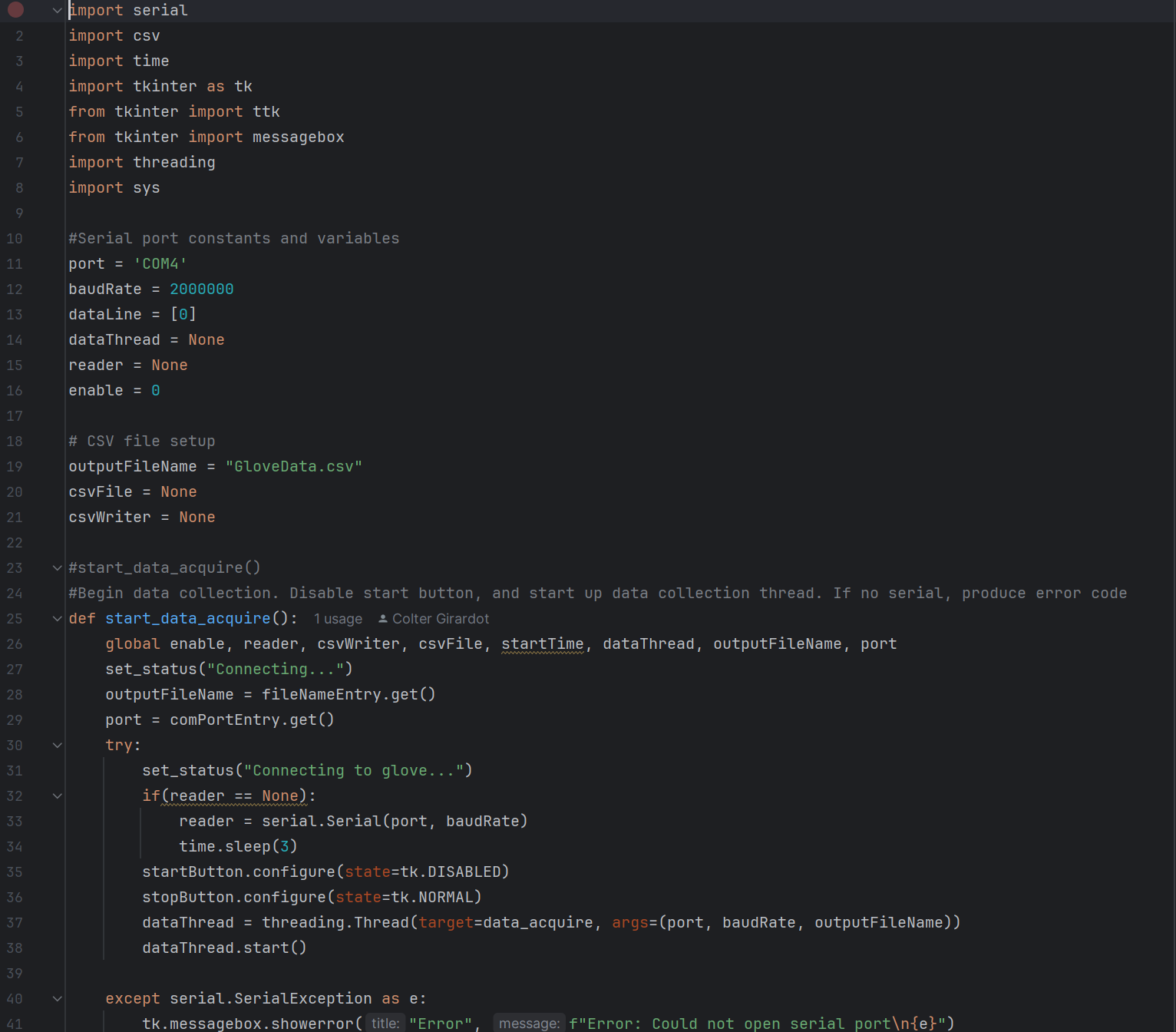
AI-generated content may be incorrect.A screen shot of a computer

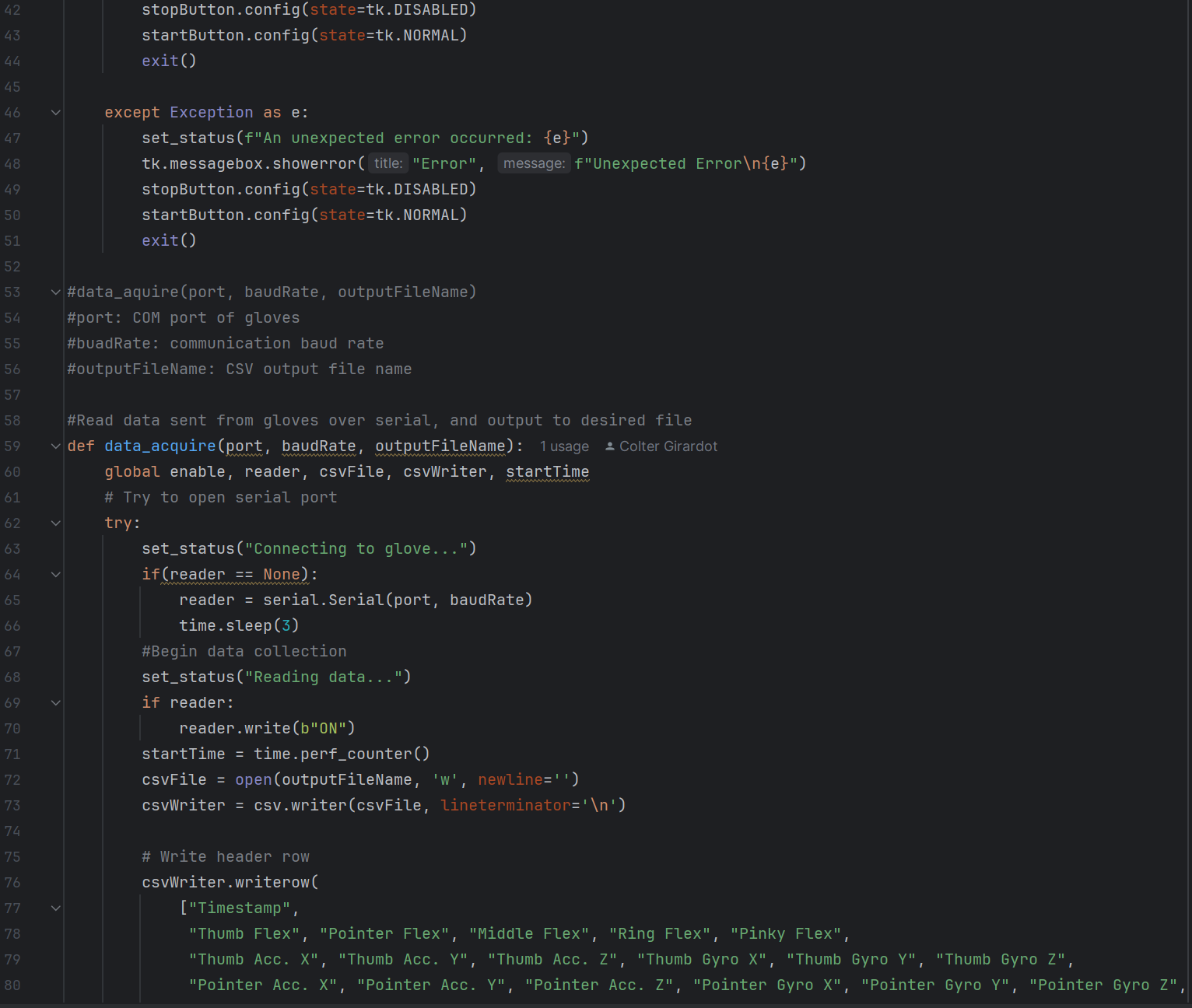
AI-generated content may be incorrect.A black background with white lines

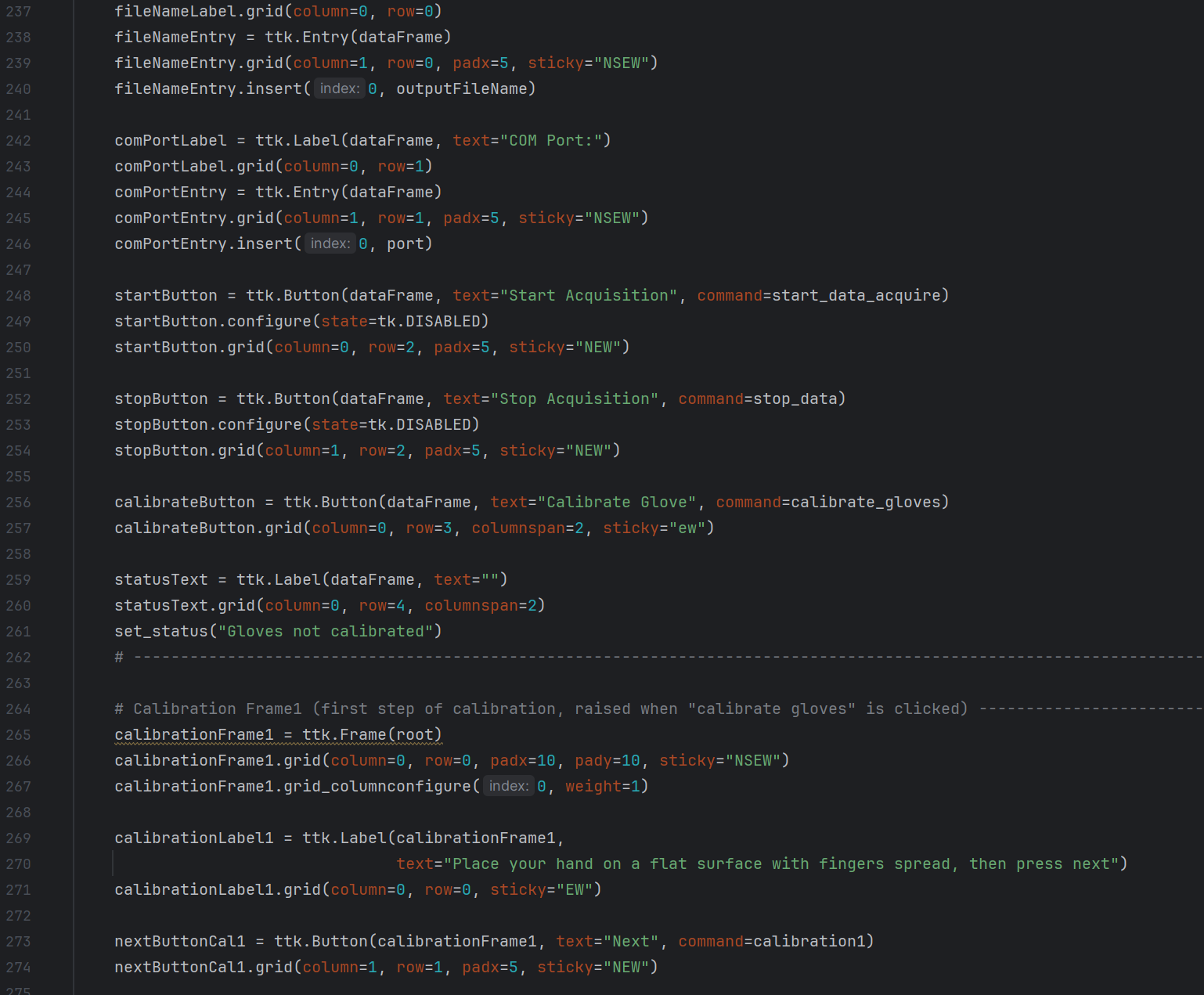
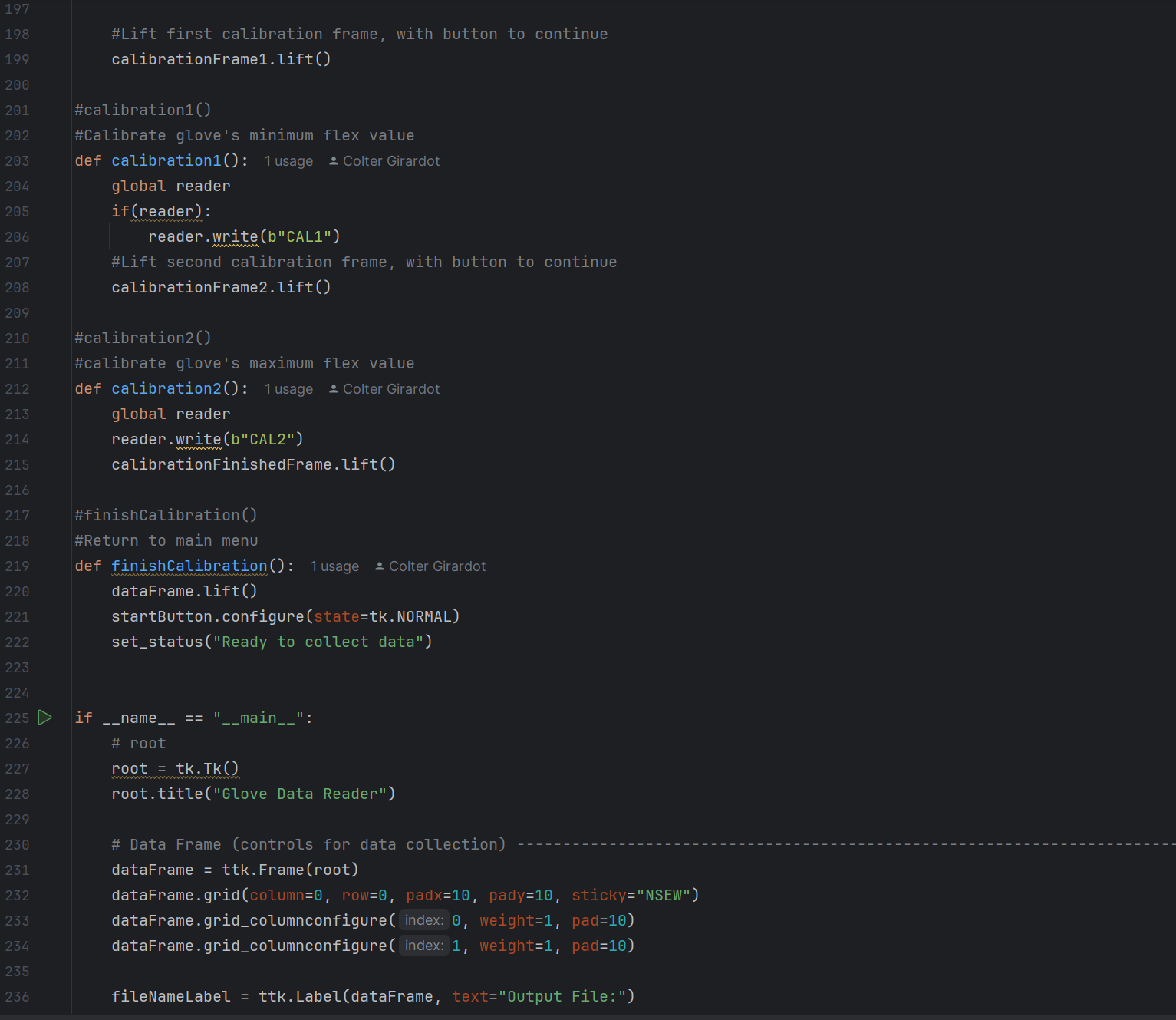
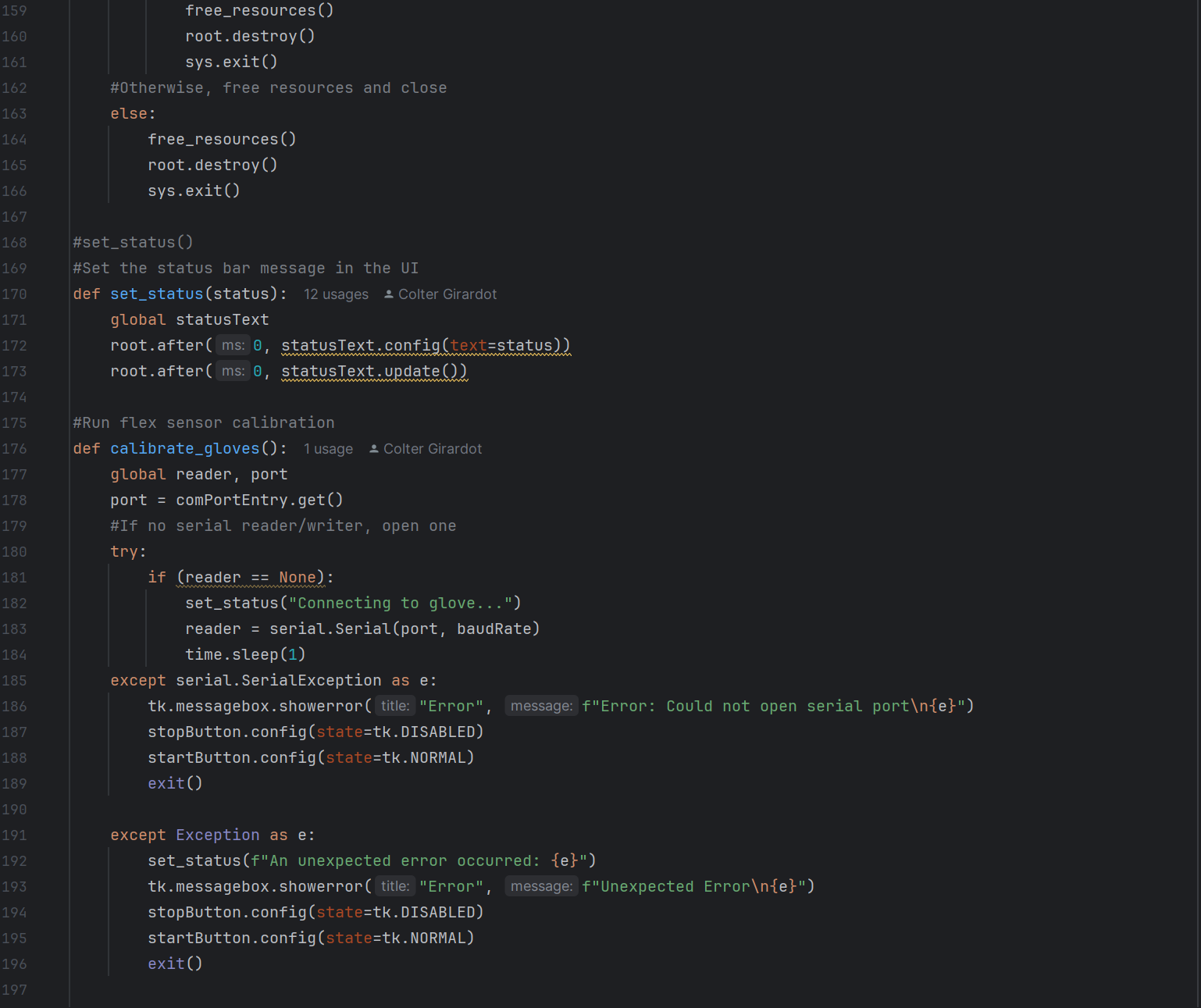
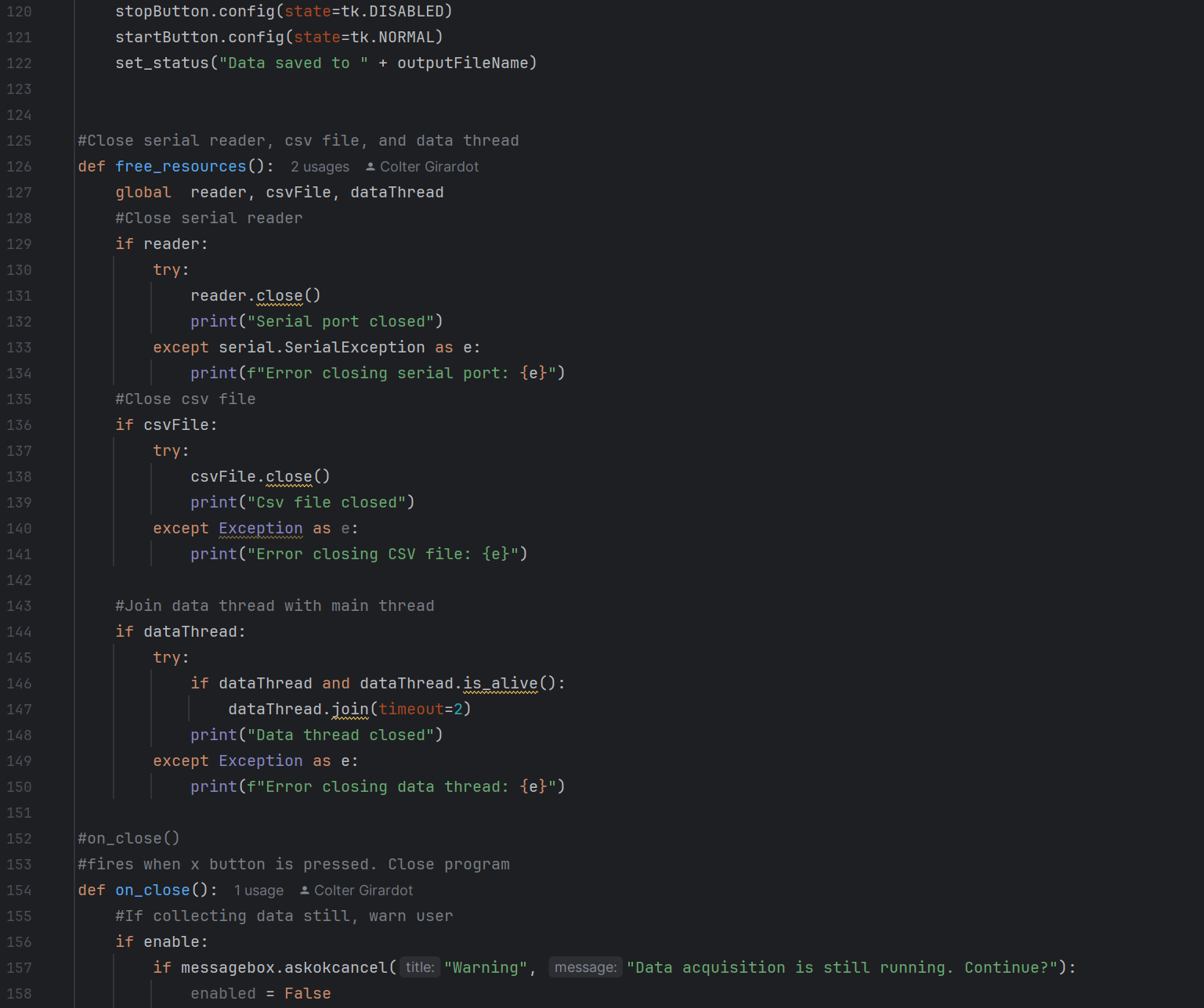
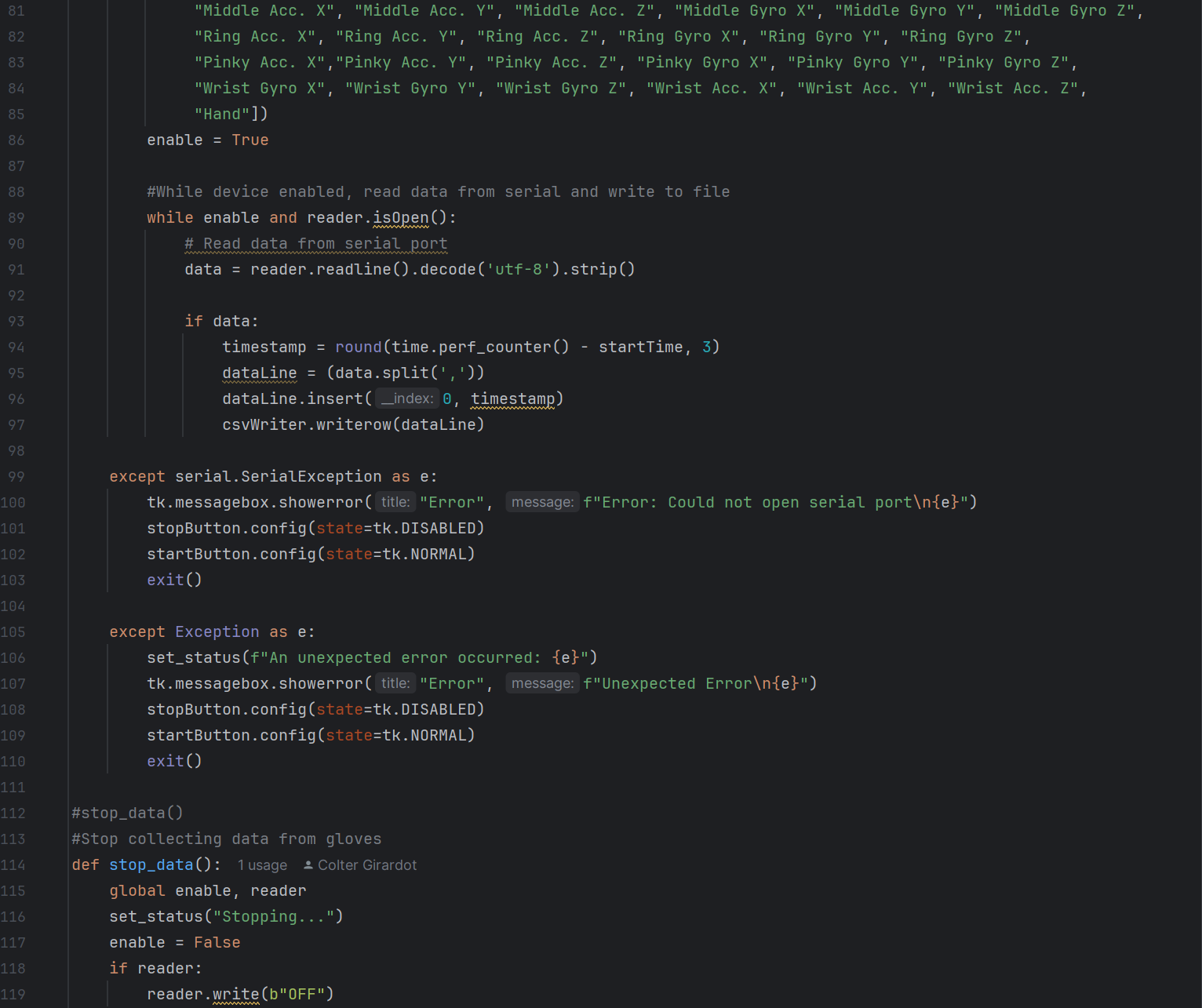
AI-generated content may be incorrect.A computer screen shot of a computer code

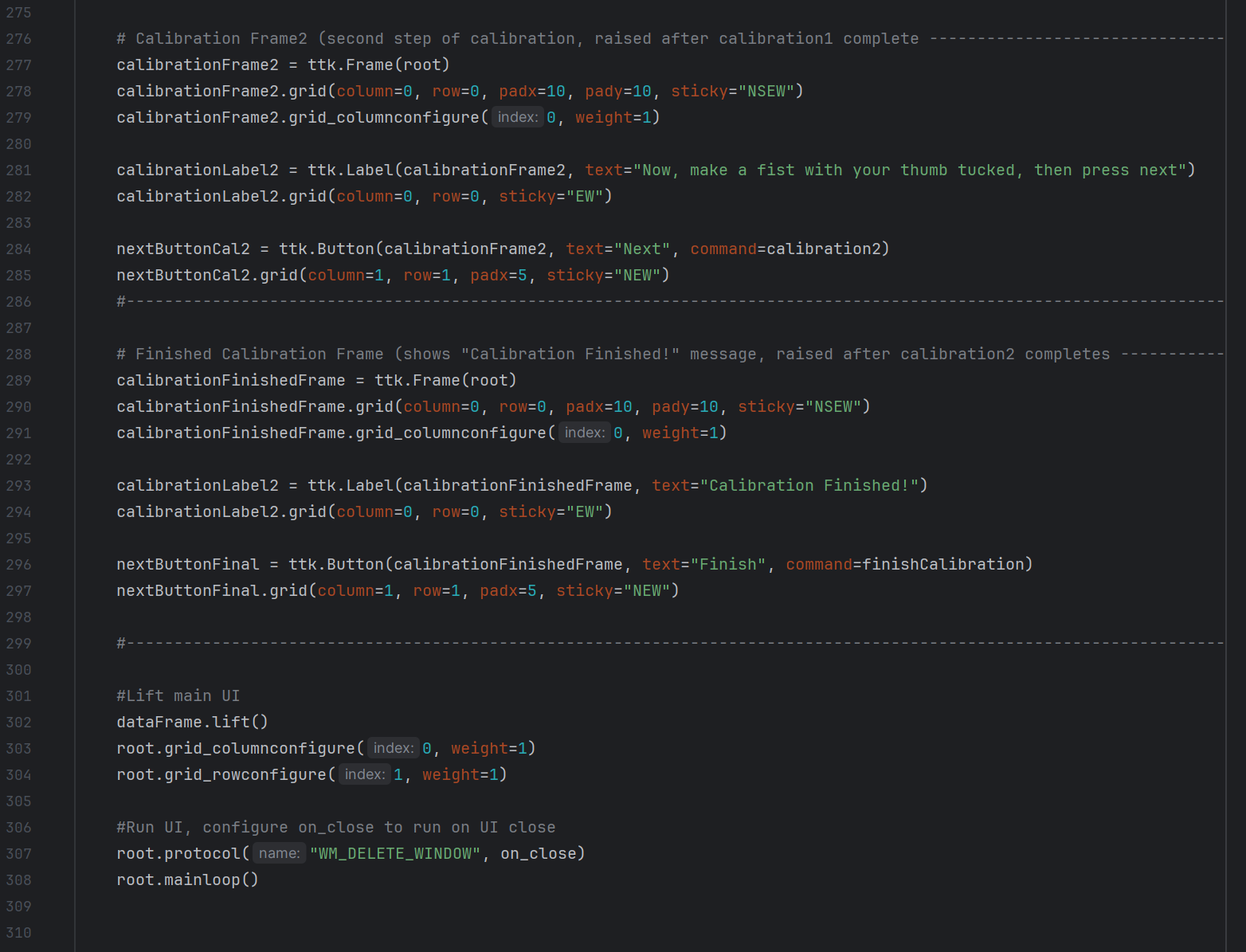
AI-generated content may be incorrect.

### Python Code









## Appendix D: User Manual

**Introduction**

* Product Name: T38P38 MoCap Gloves
* Version: 1.1
* Purpose: To locate the human hand in space and the 3D kinematic motion of a user in free space
* Target Users: Those who work as caregivers to individuals with ASD, or the individual themselves.

### What’s Included?

* 2 polyester/spandex gloves
* 2 USB-C to microUSB cable
* Electronic housings (already assembled)
* 10 Flex sensors
* 10 IMU sensors
* 2 Multiplexor Boards
* 2 Microprocessors
* 2 Perfboards
* Stemma Connectors

### Safety and Handling

* Avoid exposure to high heat and water
* Avoid collisions with electronic housings
* Ensure clean hands and proper hygiene before each use
* If irritation persist due to material, discontinue use
* This product is not a toy! To be used for research purposes only

### Setup Instructions

**Arduino Software Setup Guide**

For our design, we used Arduino for our glove’s code. During normal operation, loading code onto the Arduino manually will not be necessary – it should already have fully functioning code loaded into it. But if the code needs to be altered in some way – or if the Arduino code isn’t working on one of the gloves’ Arduino boards – here are the steps needed to get the Arduino IDE up and running.

Step 1: Head to the Arduino IDE website: <https://www.arduino.cc/en/software/>

Step 2: Download the newest version of Arduino IDE using the correct download option

* Download options just means the right version for your system (i.e. the Windows option if you’re on a Windows computer, the macOS version if you’re on a macOS computer, etc.)

Step 3: After downloading the .exe file, run it and follow the installation instructions

* The software availability option does not matter for our project
* The installation location does not matter, although we suggest going with the recommended folder

Step 4: Once installation is completed, run Arduino IDE

* There should be an icon on your desktop that you can double-click to run.
* If there is not an icon on your desktop, the .exe file is located in the Arduino IDE installation folder
  + Tutorial on where to find this folder: <https://support.arduino.cc/hc/en-us/articles/4412943340178-Open-the-Arduino-IDE-installation-folder>

Step 5: Click on the Library Manager icon on the left side of the screen

* A group of black and white objects

  AI-generated content may be incorrect.

Step 6: Search for and install the following libraries:

* “Arduino\_BMI270\_BMM1” by Arduino
* “Adafruit BusIO” by Adafruit
* “Adafruit ICM20X” by Adafruit
* “Adafruit Unified Sensor” by Adafruit
* “ICM20948\_WE” by Wolfgang Ewald

Step 7: After installing those libraries, you can now proceed to download the code from our GitHub repository (see Arduino Code Load Up Guide for details on this process).

**Arduino Code Load Up Guide**

This guide is for loading our Arduino code onto the gloves’ Arduino microcontrollers. Normally, this does not need to be done. This guide is for if the Arduino code is malfunctioning or if the code needs to be altered in some way, both of which required re-loading of the code.

Step 1: Download the latest version of our code from our GitHub repository: <https://github.com/colter107/T38_P38_Capstone>

* To download the code, click on the “arduino” folder, then click on the folder that starts with “glove\_code\_complete”
* Inside that folder, there should be a file that ends with .ino, we’ll be downloading that, so click on that file
* In the top row of the file, click on the download button

A screenshot of a computer

AI-generated content may be incorrect.

Step 2: Move the .ino file to a new folder somewhere convenient on your computer

Step 3: Open the Arduino file

* Load up the Arduino IDE if it isn’t open already
* Select **File > Open** from the Arduino IDE menus.
* Select the .ino file
* Click the "**Open**" button.

Step 4: Plug in the glove that needs new code to the computer using Arduino IDE

Step 4: In the top right corner of the program, next to the circular symbols, there will be a menu you can click to drop-down, like so:

A screenshot of a computer

AI-generated content may be incorrect.

Select the setting labeled “Arduino Nano 33 BLE” (it might have a different COM, that’s okay)

Step 4.5: If there’s no setting with that label: see Troubleshooting Section

Step 5: Click “Upload” (the arrow facing to the right in the top left corner of the program)

* Depending on the version of Arduino IDE you have installed, the button may be a different logo or in a different spot

A screen shot of a computer

AI-generated content may be incorrect.

Step 6: If the code does not upload, try to wipe the Arduino microcontroller’s code memory:

1. Unplug the glove
2. Open the glove’s main housing to check if there are any disconnected wires (at one or both ends), particular around the plug-in spot. If there are **any** disconnected wires, discontinue use immediately until the wire(s) can be re-soldered.
3. If all the wires are attached at both ends, plug the glove back into the computer
4. While the glove is plugged in, open the glove’s main housing
5. Carefully lift up the green perfboard (green board with holes and wires all around it)
6. Below the perfboard is the Arduino microcontroller, which should have an exposed white reset button on the top of it.
7. Press the reset button twice in quick succession (0.5 seconds or less)
   1. We heavily recommend using a pencil or a thin instrument to get to the button without disturbing the wire locations
8. If the Arduino now has a flashing orange light, good work! You can now try to upload the code using Steps 4 and 5 again.
9. If there are still issues uploading the code, see Troubleshooting Section.
10. If there are no more issues, return the perfboard to its original location and re-seal the main housing.

Step 7: Once the code starts uploading, the Arduino IDE will add a section at the bottom of the program that’ll document progress on uploading the code. Wait until the code is fully uploaded before calibrating or unplugging the glove.

**Python Installation**

To read the data that the gloves output, a Python program has been written. This program is used to calibrate the gloves, connect over USB, and store glove tracking data

1. Navigate to Python’s website: <https://www.python.org/> .
2. From the homepage, hover over downloads, and select your operating system (windows or macOS) from the drop-down menu.
3. Scroll to the section “stable releases”, and select the python 3.13.3 installer for your system. Code for the project is developed in **Python 3.13.3**, but should be backwards compatible with any newer python version.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

1. Follow the instructions in the python installer to install python

**Python Sensor Reader Download**

Code for the project can be found at this repository: [link goes here when public]. The code for the sensor reader is located under /sensor\_reader. Download the file sensorRead3.0, as done above with the Arduino code

**Package Installation**

The sensor reader uses the PySerial package to collect data from the gloves over USB. However, this package does not come pre-installed with python. Thus, we must tell Python to install the package.

1. Open command window

**Windows:** From the start menu search bar, type “cmd” and press enter.

**Mac:** Click the Launchpad icon in the dock. In the search bar, type “terminal” and click on the terminal app.

1. In the terminal window opened, type “pip install pyserial”, then press enter

**Python Sensor Reader Startup**

**Windows:**

1. Locate the file SensorRead3.0.py (downloaded earlier) in your file explorer
2. Right click on the file, select open with > python

A screenshot of a computer

AI-generated content may be incorrect.

**Finding Glove COM Ports**

1. Open the “Arduino IDE” application, as installed earlier.
2. Plug in Gloves
3. In the top toolbar, select tools
4. Under the dropdown, note the name of the port after the header “Port: ” this will be used for setting up the sensor reader

A screenshot of a computer

AI-generated content may be incorrect.

**Python Sensor Reader Usage**

A screenshot of a computer

AI-generated content may be incorrect.The sensor reader program provides a UI to control glove data collection, as shown below.

* **Output File:** File which the glove data will be saved to. Glove data will always be saved to the same file folder as the sensor reader program. This file should end in .csv to save to a comma-separated value file.
* **COM port:** This field must match the COM port the gloves isconnected to. See “finding glove COM ports” for more information.
* **Start Aquisition:** Sends signal to the glove to begin data collection
* **Stop Aquisition:** Stops glove data collection, saving the results to the respective file
* **Calibrate Gloves: Recommended before each use,** this guides users through the glove calibration process. See “Glove Calibration” below.
* **Status Bar (Gloves not calibrated):** Shows messages and important information about the data collection process

### Glove Calibration

To ensure accurate data collection, it is recommended that the gloves be calibrated before each use. Calibrate each glove one at a time

1. When prompted, place hand on flat surface with fingers spread, as shown below

A hand with wires and wires on a wooden surface

AI-generated content may be incorrect.

1. While holding hand being calibrated steady, press “next” in the UI
2. When prompted, move hand being calibrated to a fist with thumb tucked, as shown belowA hand wearing black glove with several small electronics

   AI-generated content may be incorrect.
3. Again, holding hand being calibrated still, press “next” in the UI
4. Wait for the “calibration complete!” message to appear in the UI. Press next to return to the data collection screen.

### Assembly Instructions

Follow the steps below to assemble the Motion Capture Gloves. Ensure you have access to the required STL files, materials, and reference images specified before beginning.

1. **3D Print Components**

* Print the IMU sensor housing, flex sensor anchors, and main housing using the provided STL files.
* Use PLA as the recommended material for all components. ABS or PETG may be used as alternatives if desired.

1. **Attach Housings to the Glove**

* Use a clear silicone RTV adhesive to attach all components to the glove. Ensure the adhesive is flexible, weather-resistant, and safe for fabric.
* Allow the adhesive to cure for 24 hours before proceeding.
* Secure the IMU housings to the fingertips of the glove, 5mm from the top edge.
* Attach the main housing to the back of the glove, 20mm from the base.
* Place the flex sensor anchor over the knuckle area.

1. **Install Heat Shrink Tubing**

* Cut and attach a 20mm segment of 5mm diameter heat shrink tubing between each IMU housing and its corresponding flex sensor anchor.

1. **Insert IMUs into Housings**

* Place each IMU sensor into its corresponding housing.
* Ensure the X-axis of the IMU is oriented toward the tip of the finger for accurate motion tracking.

1. **Complete Electrical Connections**

* Connect all components following the circuit diagram shown below, using the 28 AWG wire.

A diagram of a machine

AI-generated content may be incorrect.

1. **Install Main Circuit**

* Insert the main circuit assembly into the main housing in the order of Arduino nano, circuit board connections, Multiplexer.
* Ensure components are inserted in the correct order of placement as indicated.

1. **Route Flex Sensors**

* Thread each flex sensor through the main housing, then through the flex sensor anchor, heat shrink tube, and finally beneath the IMU housing.
* Refer to image below for the visual guide.A close up of wires

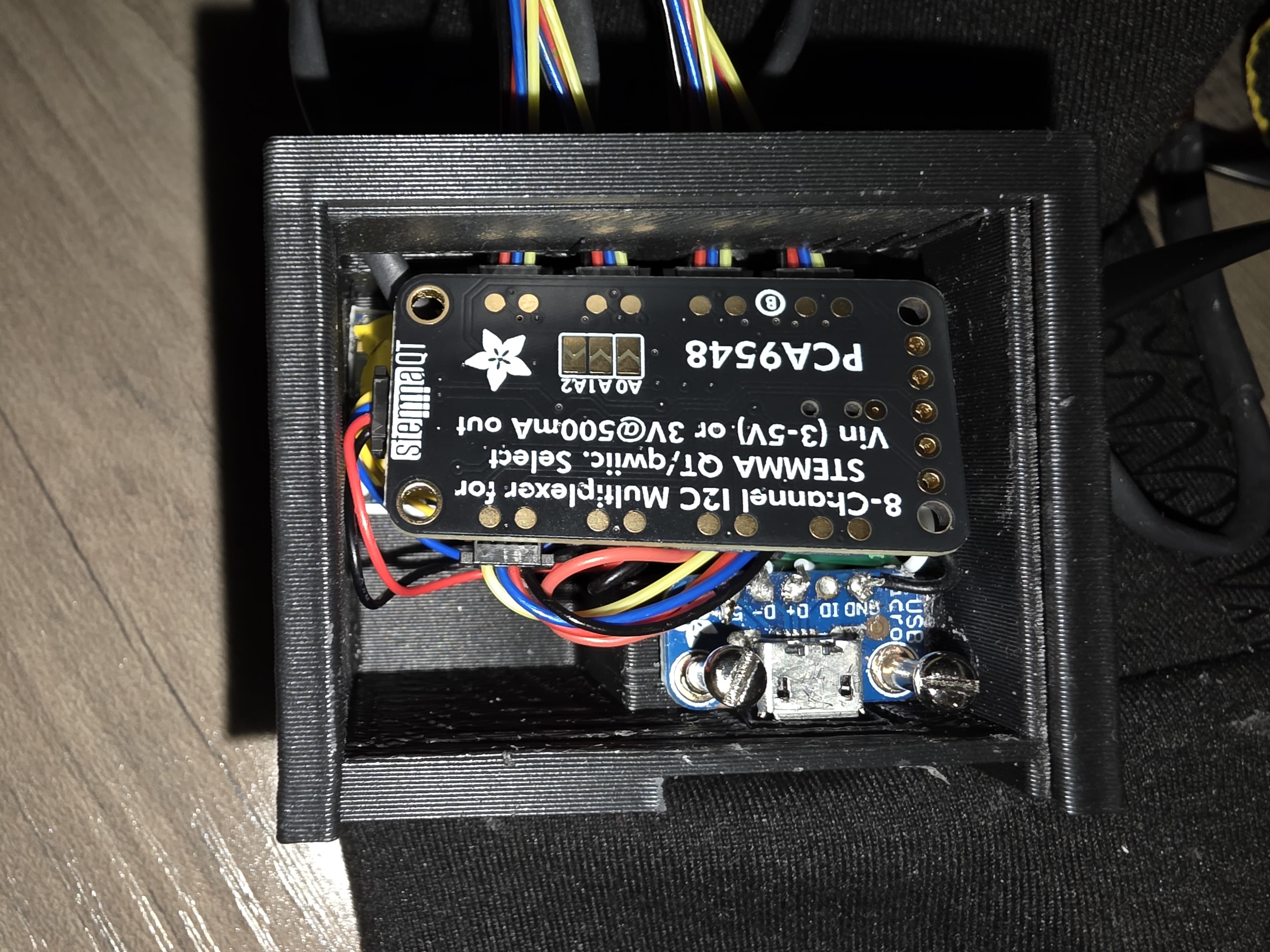
  AI-generated content may be incorrect.

1. **Connect IMUs to Main Circuit**

* Use Stemma connectors to connect each IMU to the Multiplexer.

1. **Install USB Breakout Board**

* Screw the USB breakout board into the main housing according to the positioning in the image below.



1. **Close the Main Housing**

* Place the main housing cover over the assembly and secure it to complete the enclosure.

### How to Use

1. Launch necessary applications
2. Plug in glove to PC to power on
3. Put on liner gloves, then put on the motion capture gloves
4. Begin Python program and start calibration process
5. Upon calibrating gloves, the device is ready to use
6. Initiate collection of hand movement data
7. While recording, wave hands in any motion to collect data
8. Data will be saved into a .csv file (Compatible with programs like Excel that accept .csv files)

**Maintenance & Care**

* When needed, housings should be wiped with an alcohol swab. Make sure to power off the gloves before cleaning. Remove all electronics if the inside of the housing needs to be cleaned. Avoid using water or water-based cleaners! Give the gloves plenty of time to dry (around 10 minutes) before next use.
* Inner liner is to be worn to reduce spreading illness as well as keeping the glove clean
* If the product is soiled, electronics are to be removed and the glove is to be washed by hand. See *4. Setup Instructions* for assembly instruction.

**Troubleshooting**

|  |  |
| --- | --- |
| Problem | Solution |
| Data is not accurate | Recalibrate gloves |
| LED not glowing | Check power cable/connections |
| Gloves not obtaining 5 samples per second |  |
| Arduino IDE does not see glove in drop-down menu | Unplug and replug glove. If the issue persists, restart the computer. If the issue persists, unplug the glove and check to see if there are any loose wires, particular around the plug-in spot. |
| Arduino IDE cannot upload code to a glove | Restart the computer and reset the Arduino code memory again (Step 5.5 in the Arduino Code Load Up Guide). |
| UI shows “Error: cannot open serial port” pop-up | Check USB connection to glove  Check COM port fields for typos |
| UI shows “Permission Denied: (filename).csv” | Ensure output file is not open in any other program |
| UI shows “Permission Denied: (COM port) | Close serial monitor in Arduino IDE |
| Output file shows “(filename).csv is locked for editing” when trying to open | Close sensor reader UI |

## 

**Specifications**

|  |  |
| --- | --- |
| **IMU – Adafruit 4554** |  |
| Gyroscope | 3-axis, up to 2000 degrees/second |
| Accelerometer | 3-axis, up to 16g |
| Interfaces | I2C and SPI |
| Dimensions | 25.7 mm x 17.7 mm x 4.6 mm |
| **Flex sensor – Adafruit 182** |  |
| Resistance range | Flat ~ 10KΩ, Flexed ~ 20 KΩ |
| Life cycle | 1 million flexes |
| Power rating | .5 W continuous, 1 W peak |
| Dimensions | 112.5 mm x 6.38 mm x .5 mm |
| **Multiplexer – Adafruit 5626** |  |
| # of channels | 8 independent I2C channels |
| Voltage compatibility | 3.3 to 5V |
| Dimensions | 40.6 mm x 20.2 mm x 4.8 mm |
| **Microprocessor – Arduino Nano BLE 33** |  |
| Included sensors | 3 – axis accelerometer, gyroscope, and magnetometer |
| I/O voltage | 3.3 V |
| Input voltage | 5 – 18 V |
| Clock speed | 64 MHz |
| Memory | 256 KB SRAM, 1 MB Flash |
| Dimensions | 18mm x 44 mm x 5 mm |