GNG2101

Design Project User and Product Manual

Duo Click: Designers of Accessible Clickers

Submitted by:

DuoClick: Group 3.5

Sophie Kim, 300387263

Maya Buduru, 300376286

Abdulmalek Housroum, 300334858

Fdaa Alhalaki, 300334855

2024-12-03

University of Ottawa

Table of Contents

Table of Contents ii

List of Figures v

List of Tables vii

List of Acronyms and Glossary viii

1 Introduction 1

1.1.1 Document Structure 1

1.1.2 Purpose of the Document 2

1.1.3 Intended Audience 2

1.1.4 Safety and Privacy Considerations 2

2 Overview 3

2.1.1 The Problem 3

2.1.2 The User's Needs 3

2.1.3 Why the MCB is Better 3

2.1.4 Key Features 3

2.1.5 System Architecture 4

2.2 Conventions 4

2.3 Cautions & Warnings 4

3 Getting started 5

3.1 Configuration Considerations 5

3.2 User Access Considerations 5

3.3 Accessing/setting up the System 5
3.4 System Organization & Navigation 5
3.5 Exiting the System 5
4 Using the System 6
5 Troubleshooting & Support 9
5.1 Error Messages or Behaviors 9
5.2 Special Considerations 9
5.3 Maintenance 10
5.4 Support 10
5.4.1 Problem Reporting 10
5.4.2 Security Incident Handling 10
6 Product Documentation 12
6.1 Subsystem 1 – Software 12
6.1.1 BOM (Bill of Materials) 12
6.1.2 Equipment list 12
6.1.3 Instructions 12
6.2 Subsystem 2 – Circuits 19
6.2.1 BOM (Bill of Materials) 19
6.2.2 Equipment list 20
6.2.3 Instructions 20
6.3 Subsystem 3 – Casing 23
6.3.1 BOM (Bill of Materials) 23

6.3.2 Equipment List 23
6.3.3 Instructions 24
6.4 Testing & Validation 33
7 Conclusions and Recommendations for Future Work 35
8 Bibliography 37
APPENDICES 38
9 APPENDIX I: Design Files 38
10 APPENDIX II: Other Appendices 39

List of Figures

Figure 1: Downloading Arduino IDE (Arduino, Software Downloads) 13

Figure 2: Installing Mouse.h Library 14

Figure 3: Including Mouse.h Library 15

Figure 4: Declaration Code 16

Figure 5: Code for Setup Function 16

Figure 6: Code for Loop Function 17

Figure 7: Dropdown Menu to Select Board 18

Figure 8: Selecting Board and Port 19

Figure 9: Wire Attachments 21

Figure 10: Circuit Diagram 22

Figure 11: Connections of Electronic Components 22

Figure 12: Casing Isometric View 25

Figure 13: Casing Technical Drawing 26

Figure 14: Button Holder Isometric View 27

Figure 15: Button Holder Technical Drawings 28

Figure 16: Stabilizing Block Isometric View 29

Figure 17: Stabilizing Block Technical Drawings 29

Figure 18: Button Caps Isometric View 30

Figure 19: Button Caps Technical Drawings 31

Figure 20: Button Plate Isometric View 31

Figure 21: Button Plate Technical Drawings 32

Figure 22: Downloading UltiMaker Cura Slicer (UltiMaker, 2024) 33

List of Tables

Table 1. Acronyms viii

Table 3: Common Issues 9

Table 4: Software BOM 12

Table 5: Circuits BOM 19

Table 6: Casing BOM 23

Table 7: Button/Spring Durability Test 34

Table 8. Referenced Documents 38

List of Acronyms

Table 1. Acronyms

Acronym	Definition
BOM	Bill of Materials
CAD	Computer-Aided-Design
IDE	Integrated Development Environment
MCB	Mouse Click Buttons
USB	Universal Serial Bus

1 Introduction

This User and Product Manual (UPM) provides detailed instructions for using, maintaining, and reproducing the **Mouse Click Buttons (MCB)**, a device designed to enhance accessibility for wheelchair users. The MCB enables users to activate right and left mouse clicks using buttons mounted near their joystick, offering seamless integration with their existing setup. The device connects to computers via USB, providing a reliable and straightforward solution.

The manual assumes the user has access to a powered wheelchair equipped with a joystick and a computer with USB connectivity. It is organized for ease of use, ensuring non-technical users can confidently set up and operate the MCB.

1.1.1 Document Structure

This manual is organized as follows:

1. **Introduction**: Explains the purpose of the MCB, the document structure, and the intended audience.
2. **Overview**: Details the problem the MCB solves, the user needs it addresses, and its unique benefits.
3. **Getting Started**: Provides instructions for initial setup and installation.
4. **Using the System**: Describes the operation of the MCB, including its features.
5. **Troubleshooting & Support**: Offers solutions for common issues and guidance on maintenance.
6. **Product Documentation**: Includes technical details such as the Bill of Materials (BOM), testing, and validation records.
7. **Conclusions and Recommendations**: Summarizes the project and suggests areas for improvement.

8. **Bibliography and Appendices**: References and supplementary resources, including design files and schematics.

1.1.2 Purpose of the Document

The primary goals of this document are to:

- Enable users to set up and operate the MCB with ease.
- Provide clear maintenance and troubleshooting instructions.
- Offer comprehensive documentation to facilitate future reproductions or improvements.

1.1.3 Intended Audience

This document is for:

- **Wheelchair Users**: Enhancing accessibility for computer interaction.
- **Caregivers**: Assisting users with setup and operation.
- **Technicians and Developers**: Supporting efforts to improve or reproduce the device.

1.1.4 Safety and Privacy Considerations

- **Safety**: Always turn off the wheelchair before installing or adjusting the MCB to prevent accidental movement.
- **Hygiene**: The MCB is designed with cleanable materials, ensuring safe and sanitary use.
- **Privacy**: The device does not store or transmit any data, safeguarding user privacy.

2 Overview

2.1.1 The Problem

Many wheelchair users who rely on a joystick for mobility and as a computer cursor lack accessible solutions for right and left mouse clicks. Traditional computer mice and touchpads are often unusable for individuals with limited dexterity, creating barriers to digital accessibility.

2.1.2 The User's Needs

Users need a solution that:

- Integrates seamlessly with their joystick setup.
- Offers ergonomic, easy-to-use buttons for mouse clicks.
- Is simple to install, maintain, and clean.
- Provides a reliable USB connection for direct computer interaction.

2.1.3 Why the MCB is Better

The MCB is a unique and effective solution because:

- **Plug-and-Play**: It connects via USB with no additional drivers or software required.
- **Durable and Hygienic**: Designed for daily use and easy cleaning.
- **Secure Mounting**: The buttons are securely mounted near the joystick but can be quickly removed if necessary.

2.1.4 Key Features

- **USB Connectivity**: Reliable and easy to set up with any USB-compatible computer.
- **Ergonomic Design**: Buttons positioned for effortless access near the joystick.
- **Robust Construction**: Built for durability and consistent performance.

2.1.5 System Architecture

The MCB consists of:

1. **Hardware Components**:
    a. Two buttons for right and left mouse clicks.
    b. A microcontroller that processes button inputs.
    c. Mounting hardware for secure attachment to the wheelchair.
2. **Connectivity**: A direct USB connection ensures consistent performance without the need for wireless pairing or additional configuration.

2.2 Conventions

In this manual:

- **Action Required**: Instructions for user actions begin with **Action** in bold.
- **Caution**: Important safety notes are highlighted under **Caution**.
- **Note**: Additional tips and helpful information are provided under **Note**.

2.3 Cautions & Warnings

- **Caution**: Ensure the wheelchair is powered off during installation to prevent accidents.
- **Caution**: The buttons are designed to withstand firm presses; however, avoid using tools or excessive external force beyond typical finger pressure to ensure the device's longevity.
- **Warning**: Do not expose the device to water or excessive moisture to prevent damage.

3 Getting started

3.1 Configuration Considerations

The mouse buttons are configurated such that the cord is already plugged into the device. If the cord is not connected to the device, it can be plugged into the Micro port on the side.

3.2 User Access Considerations

Users with mobility impairments may have issues connecting the device to a computer or attaching it to the wheelchair. If that is the case, they may require help from another individual to set up the system.

3.3 Accessing/setting up the System

First, secure the device onto the Velcro pad of the wheelchair, with the Arduino port facing towards the computer. If the cord connecting the device through the device's port is not already connected, plug it in. After, plug in the USB end of the cord into a USB port in the desired computer.

3.4 System Organization & Navigation

The system is made up of the device and a Micro USB Cable (USB A Male to Micro B Male). The cable connects to the mouse buttons through a port on the side.

3.5 Exiting the System

To exit the system, unplug the device from the computer, and remove it from the Velcro pad. Store in a dry environment.

4 Using the System

The following sub-sections provide detailed, step-by-step instructions on how to use the various functions or features of the DUO CLICK Mouse Buttons. Also, here is a link for a video demonstration of the Mouse:

https://youtube.com/shorts/sc4C8VbngQ0?si=A4Xr5ClbRW24fK4c

4.1 Click Buttons

4.1.1 Left, Right or Double Click

- **Input:** Press the left, right button on the mouse click device. Press the left button twice to do a double click.
    - The button is tactile, requiring low-force actuation for accessibility.
- **Output:** The system sends a signal to the connected computer to execute a left, right, or double-click action.
- **Special Instructions:**
    - Ensure the device is securely attached to the wheelchair armrest.
    - Confirm that the button is pressed fully to register the input and USB cable is plugged to the computer.
- **Caveats:**
    - Avoid excessive force, as it may affect durability despite robust testing.

4.2 Connectivity

4.2.1 USB Wired Connection

- **Input:** Connect the device to the computer using the provided USB cable.
- **Output:** Establishes a stable wired connection for input signals.
- **Special Instructions:**

    - Ensure the USB cable is securely connected to both the device and the computer.
    - Verify the cable is not damaged or obstructed.
- **Caveats:**
    - The operating range is limited to the cable's length (1.5 meters).

4.3 Mounting the Device

4.3.1 Velcro Attachment

- **Input:** Use the Velcro strap to mount the device onto the wheelchair armrest.
- **Output:** Secures the device in place for optimal usability.
- **Special Instructions:**
    - Ensure the Velcro strap is tightly secured but not overtightened to prevent damage.
    - Test the attachment for stability before use.
- **Caveats:**
    - Ensure the mounting surface is clean for proper adhesion.

4.4 Durability and Maintenance

4.4.1 Drop Resistance

- **Input:** Subject to drop/impact tests up to 1 meter during operation.
- **Output:** Device remains functional and intact.
- **Special Instructions:**
    - Use padded surfaces for testing to prevent damage.
- **Caveats:**
    - Frequent or deliberate impacts may degrade performance over time.

4.4.2 Component Replacement

- **Input:** Remove the device cover using a screwdriver to access internal components.
- **Output:** Facilitates part replacement or repair.
- **Special Instructions:**
    - Follow the assembly manual for reattaching the cover securely.
    - Use only compatible replacement parts to maintain performance.
- **Caveats:**
    - Avoid disassembling without proper tools or training.

5 Troubleshooting & Support

This section provides simple and clear steps to identify and resolve issues that may arise while using the **Mouse Click Buttons (MCB)**. It also includes maintenance tips and support information to ensure the device remains in good working condition.

5.1 Error Messages or Behaviors

Below are common issues users may experience, their likely causes, and steps to resolve them.

Table 2: Common Issues

Issue	Possible Cause	Solution
Buttons not responding	Loose USB connection or cable damage	**Action**: Check if the USB cable is securely connected to the computer. Replace the cable if damaged.
Computer not recognizing the device	Faulty USB port or driver conflict	**Action**: Try plugging the USB cable into a different USB port. Restart the computer if necessary.
Delayed response from buttons	Dirt or debris affecting the button mechanism	**Action**: Clean the buttons with a damp cloth and ensure they are free from obstructions.
Buttons feel stuck	Physical wear or debris inside the buttons	**Action**: Gently press the button multiple times to release it. Clean around the button edges.

5.2 Special Considerations

- **Proper Installation**: Always ensure the MCB is securely mounted near the joystick before use. Misalignment may affect usability.
- **Power Off During Setup**: To avoid accidental joystick movement, ensure the wheelchair is powered off when installing or adjusting the MCB.
- **USB Port Compatibility**: Use USB 2.0 or higher ports for the best performance.

5.3 Maintenance

Regular maintenance can prevent issues and extend the life of the MCB. Follow these steps:

1. **Weekly Cleaning**:
    a. Use a damp, lint-free cloth to wipe the buttons and USB cable.
    b. Ensure no debris is obstructing the button mechanisms.
2. **Monthly Inspection**:
    a. Check the mounting hardware to ensure it is securely attached.
    b. Inspect the USB cable for signs of wear or damage.
3. **Button Testing**:
    a. Press each button to ensure they respond correctly.
    b. Replace buttons if they show significant wear or lack of response.

5.4 Support

If you experience issues that cannot be resolved with the above steps, assistance is available.

5.4.1 Problem Reporting

To report a problem:

1. Describe the issue in detail, including:
    a. The steps leading to the problem.
    b. Any error messages or unusual behaviors.
2. Attach pictures or videos (if possible) to help explain the issue.
3. Email the details to the support contact above.

5.4.2 Security Incident Handling

If the device is lost, damaged, or suspected of misuse:

1. Disconnect the USB cable and power off the wheelchair.
2. Report the incident to the contact listed above with a detailed description of the issue.

With these steps, you can effectively troubleshoot, maintain, and seek support for the MCB, ensuring a reliable and smooth experience.

6 Product Documentation

6.1 Subsystem 1 – Software

6.1.1 BOM

Table 3: Software BOM

Item	Quantity	Price	Use	Link
Mouse.h Arduino Library	1	$ 0	To write code that can implement mouse clicker functions	https://docs.arduino.cc/libraries/mouse/
CANADUINO® Pro Micro Atmega32u4, 5V, 16MHz	1	$ 19.10	For code to be uploaded onto	https://www.amazon.ca/dp/B01N4TVIQX?ref=ppx_yo2ov_dt_b_fed_asin_title

6.1.2 Equipment list

- Laptop or computer
- Micro USB Cable (USB A Male to Micro B Male)
- Arduino IDE (free, downloaded from https://www.arduino.cc/en/software)

6.1.3 Instructions

1. Download the Arduino IDE from the Arduino website (arduino.cc/en/software).

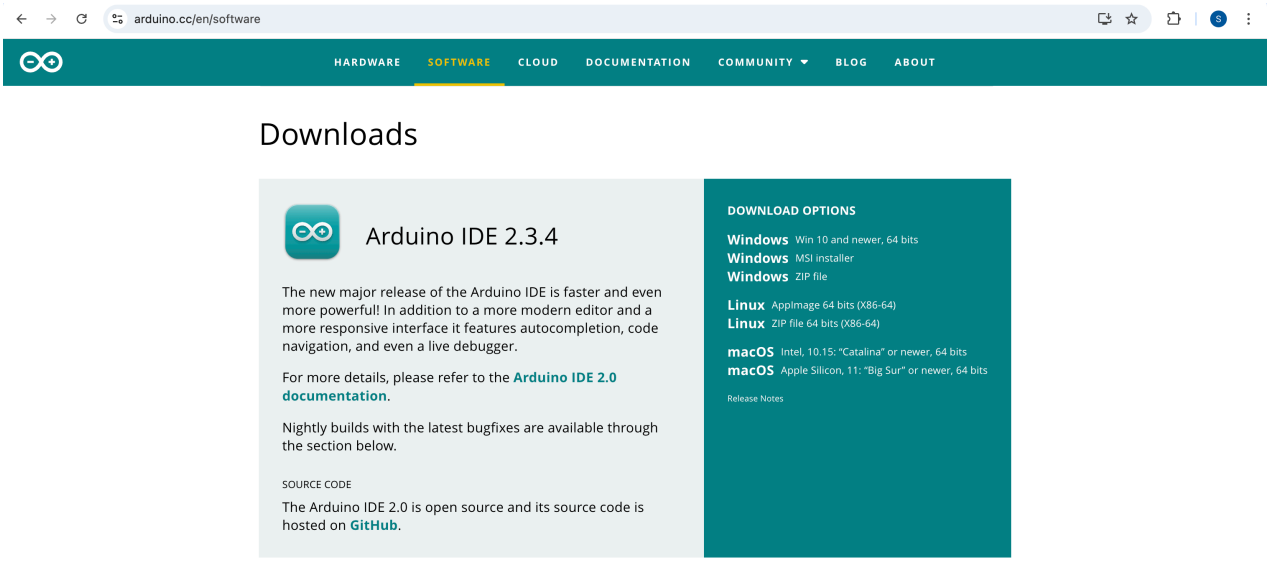


Figure 1: Downloading Arduino IDE (Arduino, Software Downloads)

2. Open the IDE application and download Arduino's Mouse libary.

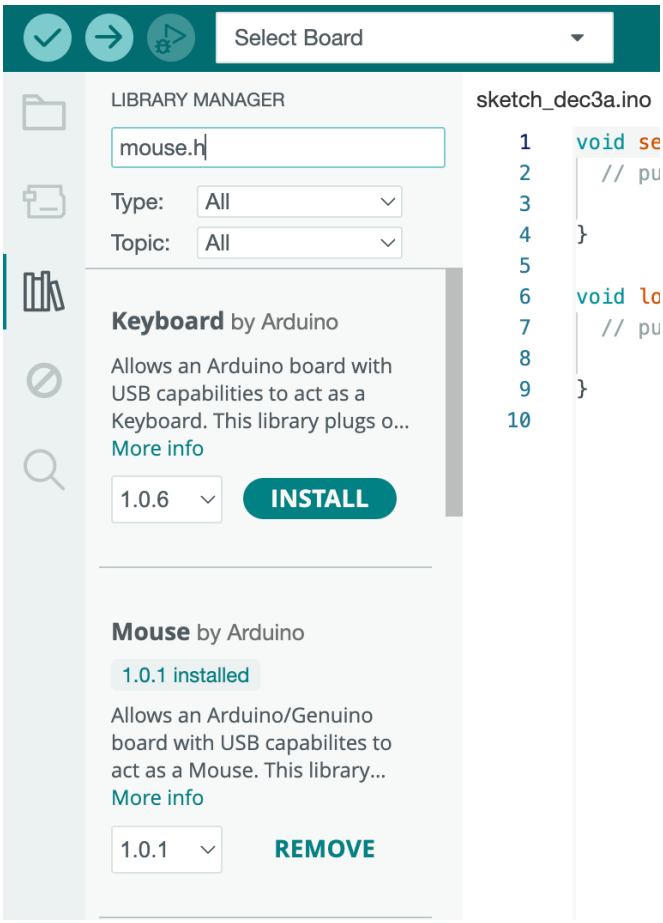


Figure 2: Installing Mouse.h Library

3. Select “Sketch” on the toolbar > “Include Library” > “Mouse”

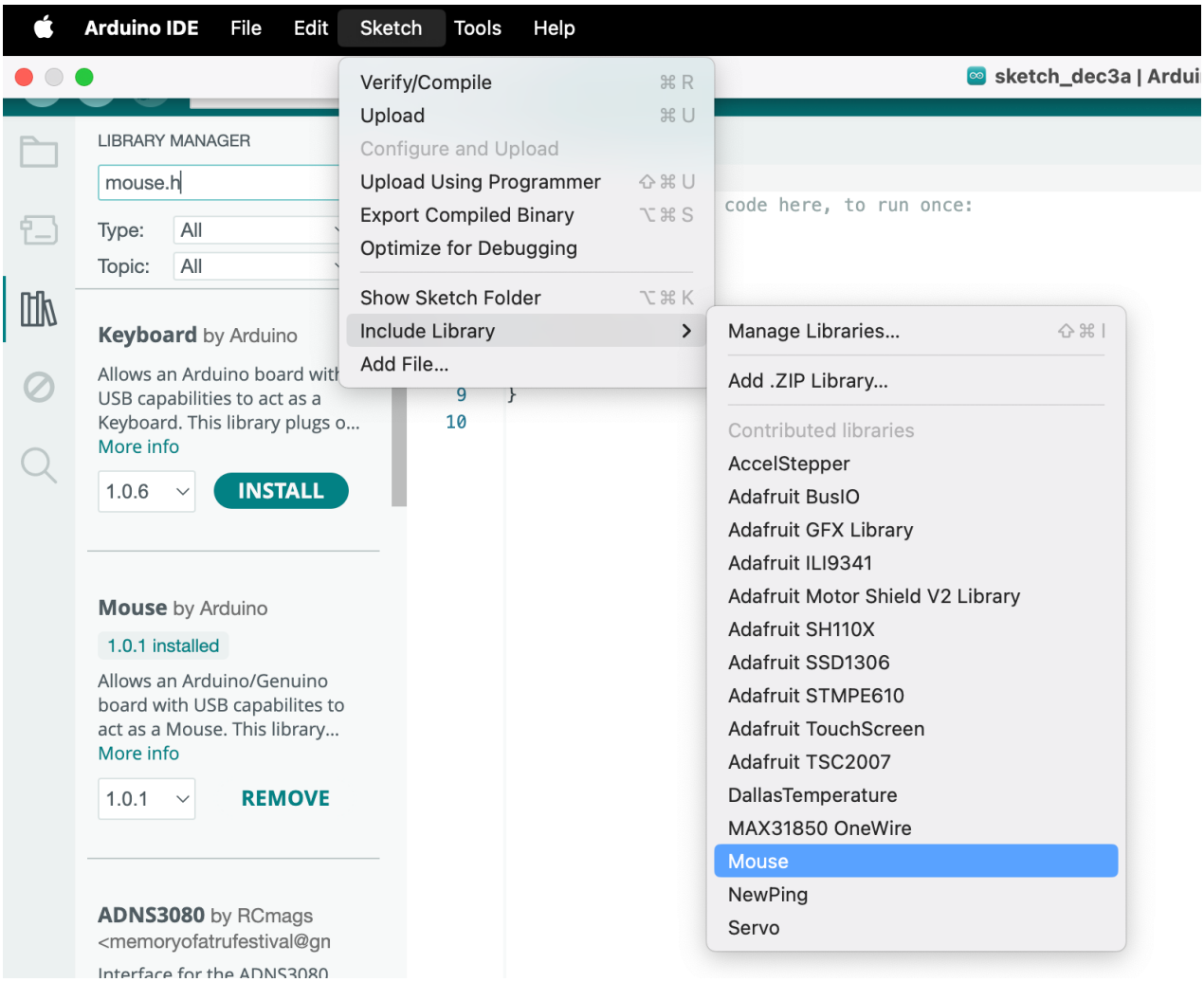


Figure 3: Including Mouse.h Library

4. Write code for the mouse clicker functions.

```
#include <Mouse.h> // Include the Mouse library

const int leftClickPin = 3;   // Pin for left-click button
const int rightClickPin = 2;  // Pin for right-click button

bool leftButtonPressed = false;   // State tracking for left button
bool rightButtonPressed = false;  // State tracking for right button
bool dragging = false;            // State tracking for dragging action
unsigned long rightButtonPressTime = 0;  // To track how long right button is pressed
const unsigned long holdThreshold = 500; // Hold time in milliseconds for drag action
```

Figure 4: Declaration Code

```

void setup() {
  // put your setup code here, to run once:
  // Initialize button pins as input with an internal pull-up resistor
  pinMode(leftClickPin, INPUT_PULLUP);
  pinMode(rightClickPin, INPUT_PULLUP);

  // Initialize the Mouse library
  Mouse.begin();

  // Debug message
  Serial.begin(115200);
  Serial.println("Mouse control initialized");
}
```

Figure 5: Code for Setup Function

```
25
26  void loop() {
27    // put your main code here, to run repeatedly:
28    // Check for left-click button press and release
29    Serial.println("in loop func");
30    if (digitalRead(leftClickPin) == LOW && !leftButtonPressed) {
31      Mouse.click(MOUSE_LEFT);            // Perform left mouse click
32      Serial.println("Left click");        // Debug output
33      leftButtonPressed = true;           // Set state to prevent multiple clicks
34    }
35    // Reset state when left button is released
36    if (digitalRead(leftClickPin) == HIGH && leftButtonPressed) {
37      leftButtonPressed = false;
38    }
39
40    // Right-click or drag based on hold duration
41    if (digitalRead(rightClickPin) == LOW) {
42      if (!rightButtonPressed) {
43        rightButtonPressTime = millis();   // Record the time when button is first pressed
44        rightButtonPressed = true;
45      }
46
47      // Check if the button is held for the specified threshold
48      if (!dragging && (millis() - rightButtonPressTime > holdThreshold)) {
49        Mouse.press(MOUSE_LEFT);           // Start drag (hold left mouse button)
50        Serial.println("Dragging started"); // Debug output
51        dragging = true;                   // Set dragging state
52      }
53    }
54
55    // If the right button is released
56    if (digitalRead(rightClickPin) == HIGH && rightButtonPressed) {
57      if (dragging) {
58        Mouse.release(MOUSE_LEFT);         // Stop drag (release left mouse button)
59        Serial.println("Dragging stopped");  // Debug output
60        dragging = false;                   // Reset dragging state
61      } else {
62        // If not dragging, it's a right-click
63        Mouse.click(MOUSE_RIGHT);          // Perform right-click
64        Serial.println("Right click");       // Debug output
65      }
66      rightButtonPressed = false;           // Reset the press state
67    }
68
69    delay(10); // Small delay for stability
70  }
```

Figure 6: Code for Loop Function

5. Connect CANADUINO Pro Micro to the laptop or computer, click on “Select Board” dropdown located at the top left corner.

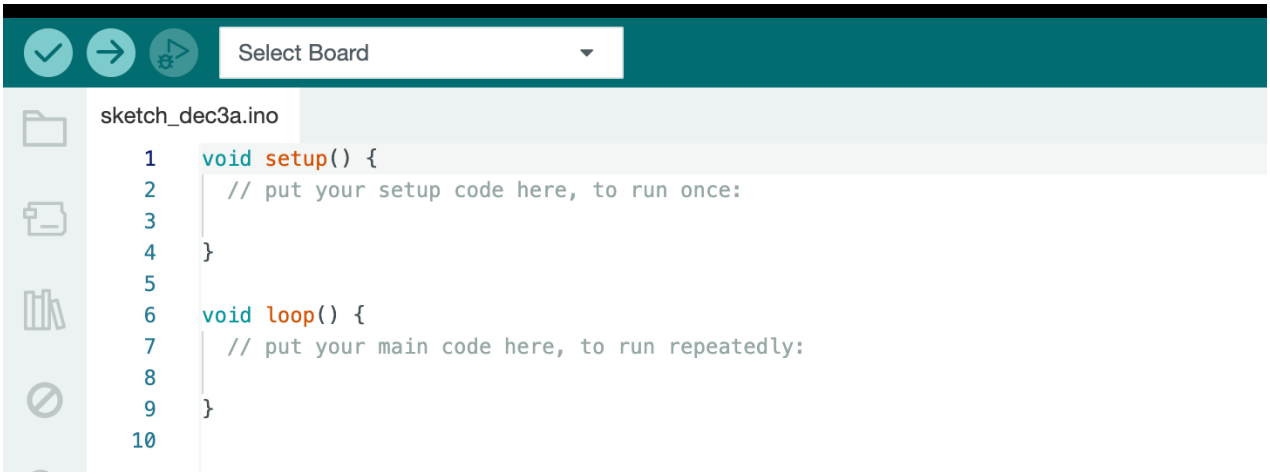


Figure 7: Dropdown Menu to Select Board

Select the corresponding board & USB connector on the IDE.

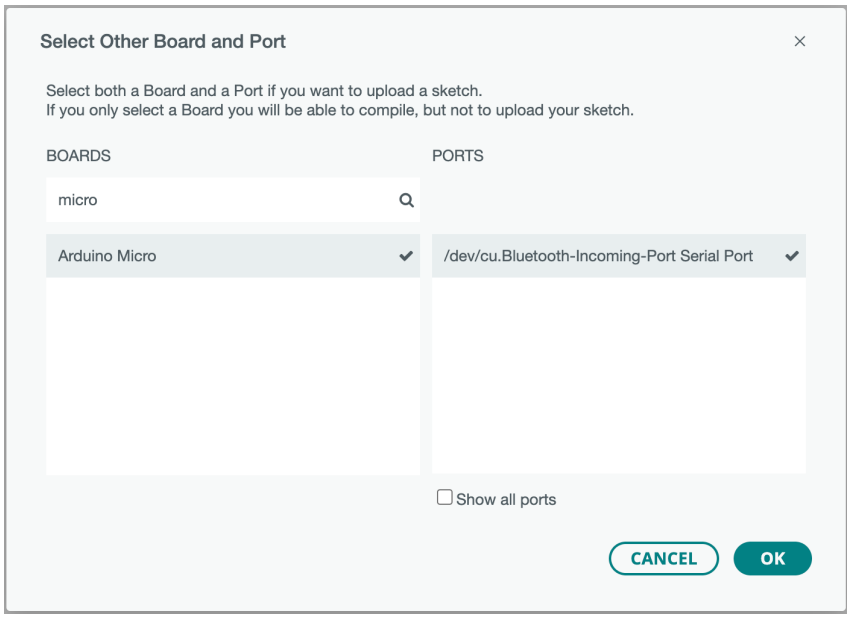


Figure 8: Selecting Board and Port

6. Upload the code onto the Pro Micro.

6.2 Subsystem 2 – Circuits

6.2.1 BOM

Table 4: Circuits BOM

Item	Quantity	Price	Use	Link
CANADUINO® Pro Micro Atmega32u4, 5V, 16MHz	1	$ 19.10	For code to be uploaded onto	https://www.amazon.ca/dp/B01N4TVIQX?r

*This is already mentioned in BOM for software.				ef=ppx_yo2ov_dt_b_fed_asin_title
12mm Momentary On/Off Button	2	$ 3.00	To send left and right click signals to the microcontroller	https://makerstore.ca/shop/ols/products/12mm-latching-on-off-button
Jumper Wires	4	$ 0	To connect the buttons to the microcontroller	N/A (free from Makerspace)
Heat Shrink Tubing	2	$0	To cover exposed wires	N/A (free from Makerspace)
Micro USB Cable (6.6 ft), A Male to Micro B Male	1	$8.99	To power the microcontroller	https://www.amazon.ca/dp/B099K78GW4?ref=ppx_pop_mob_ap_share

6.2.2 Equipment list

- Wire cutter
- Soldering iron
- Blow drier (for heat shrink tubing)

6.2.3 Instructions

1. Solder the pins of the microcontroller, if it did not come pre-soldered.

2. Connect each of the pins of the buttons to one red wire (for power connection) and one black wire (for ground connection). It can be soldered on, or attached by alternative means, as seen in the picture below.

Figure 9: Wire Attachments

3. Make the rest of the connections as seen in the picture below. Like in step 2, either solder the connections or attach by alternative means. Ensure that the wire lengths are short (picture below does not reflect the appropriate wire lengths).

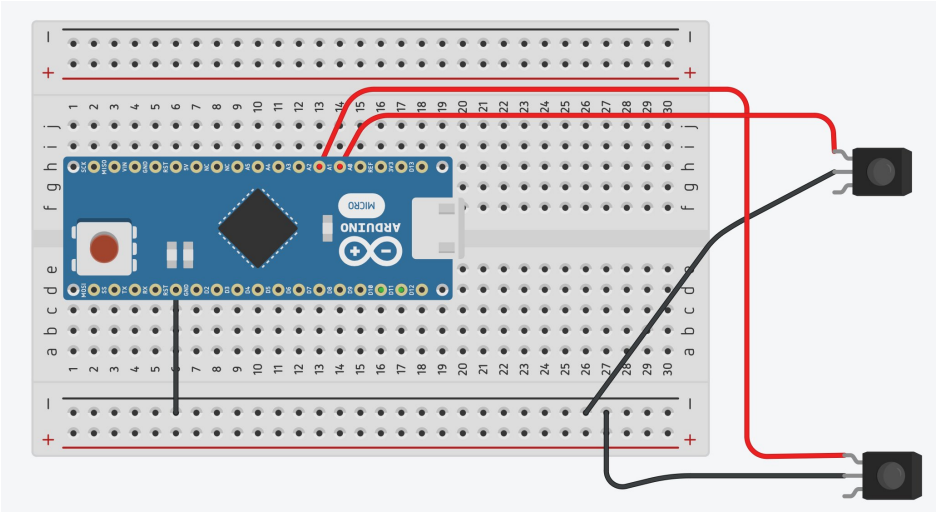


Figure 10: Circuit Diagram

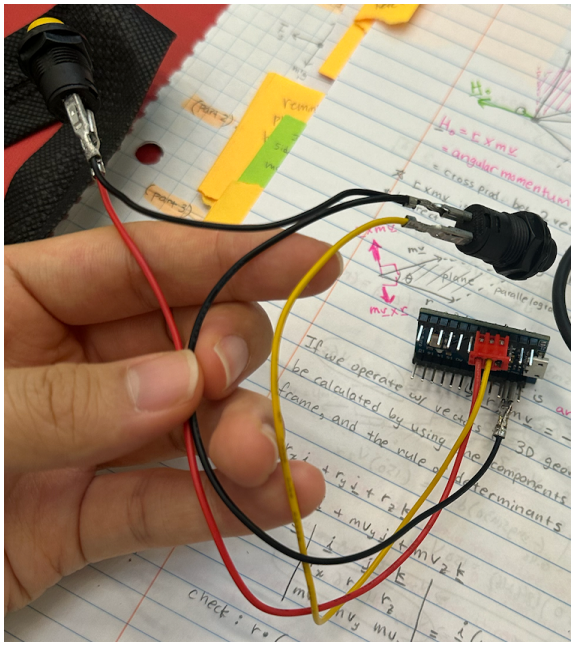


Figure 11: Connections of Electronic Components

4. Connect the microcontroller with a power source to verify that the left- and right-click functions properly.

6.3 Subsystem 3 – Casing

6.3.1 BOM

Table 5: Casing BOM

Item	Quantity	Price	Use	Link
PLA Filament	1	$ 0	To make 3D prints	N/A (free from Makerspace)
Springs	2 (However, comes as a pack of different sizes)	$ 10.61	To provide resistance for the button mechanism	https://www.amazon.ca/Cilky-Compressed-Assorted-Stainles-Compression/dp/B0CR8W8D8G?crid=1V0UAY3M768HB&dib=eyJ2IjoiMSJ9.eIJGp3CMo4-LgvNC0wbFYTeH5qLhXYgGj1YvjOGrssoVNwua89bsJZJ1a8lZWaI4duXSaV-epdma8IS8Gbf9TJcaFHn4nXJ5oYazO6vZyD_f6Upmn4DRG_-upi1FK5xCUjpKJllqkgUU1FYGd9yjYN8y6XbC8mDM1EWfMU-wLA8IypkDDUQe-KXnaR9HZt5FQNe3YGMtqf2dEjxzi1JWLQkx5YcEuyXbRCcEll0XWsymE0NIO8zNU4JGrlbzbpip_s5fA1q9G9-aZPcP06Z9YuiVtxUAGjUhXiIF8ywSeF0.XzSESlNCUyy2MKU-N5-1qJl7eiXL4kQdyAfWEXTX4YU&dib_tag=se&keywords=springs%2BconicaI&qid=1733281287&sprefix=springs%2Bconical%2Caps%2C90&sr=8-5&th=1
Screws	4	$ 0	To screw in the components together	N/A (free from Makerspace)

6.3.2 Equipment List

- UltiMaker 3D printer

- UltiMaker Cura 5.9 3D Printing Slicer (free, download from https://ultimaker.com/software/ultimaker-cura/)
- Wire cutter
- Sanding paper
- Pliers
- Rulers, calipers

6.3.3 Instructions

1. Make sketches of the designs for each component.
2. Design the components using computer aided design (CAD) software, such as SolidWorks.

Casing:

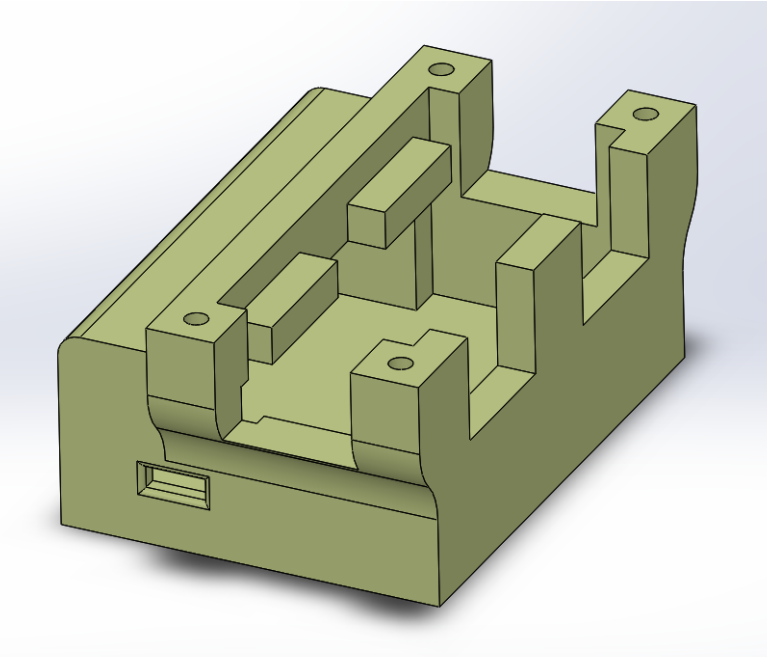


Figure 12: Casing Isometric View

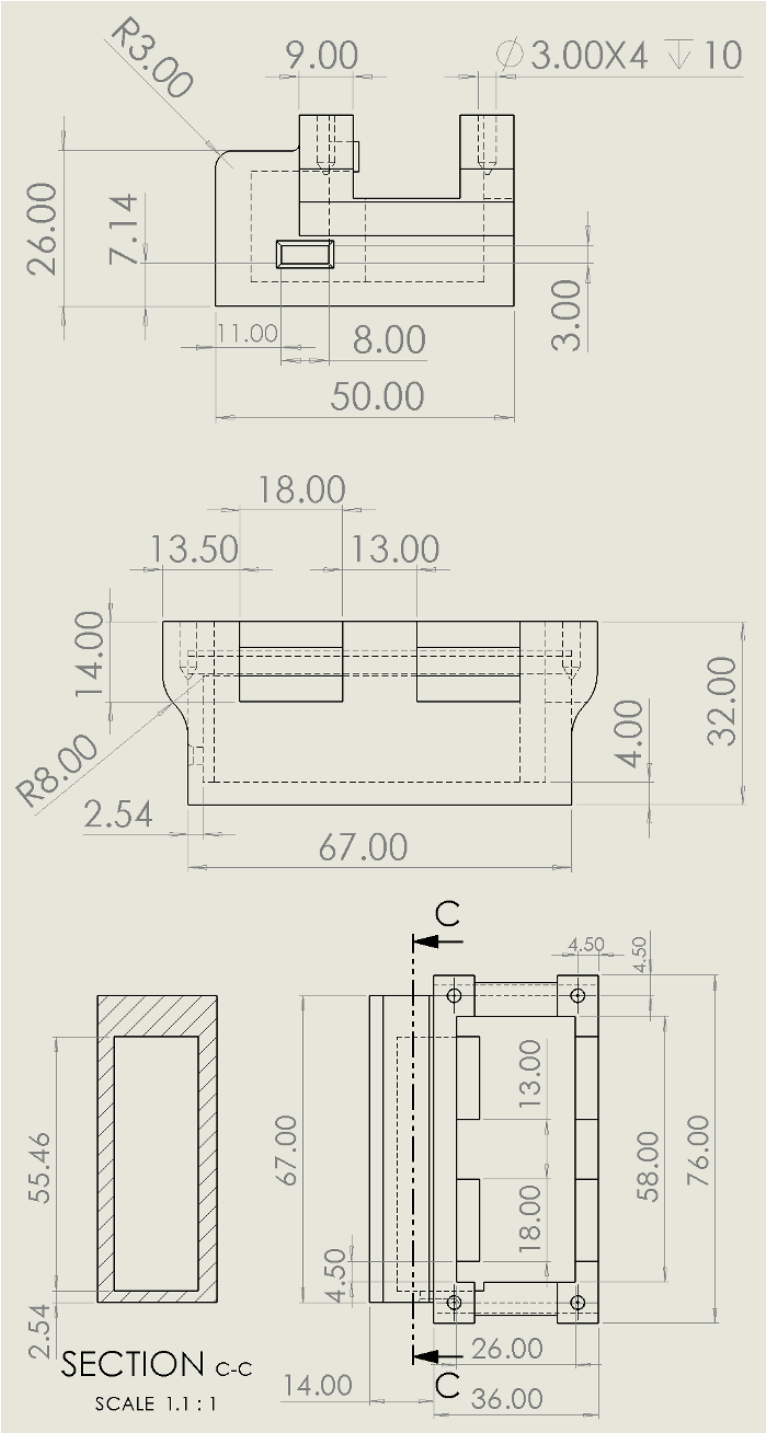


Figure 13: Casing Technical Drawing

Button Holder:

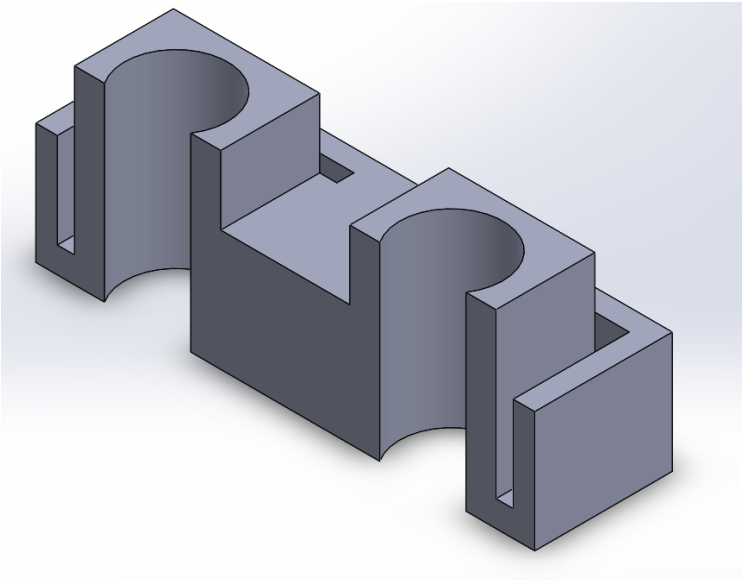


Figure 14: Button Holder Isometric View

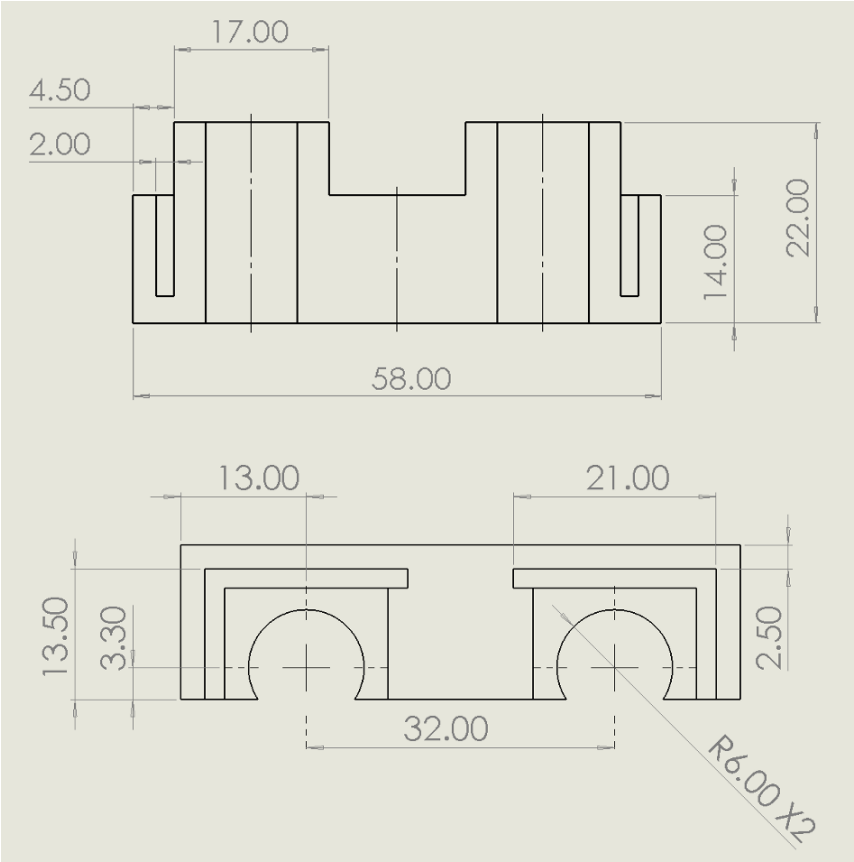


Figure 15: Button Holder Technical Drawings

Stabilizing Block:

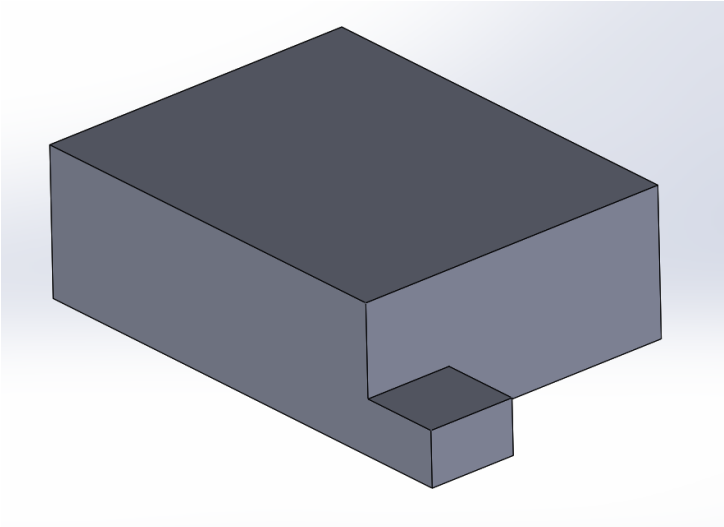


Figure 16: Stabilizing Block Isometric View

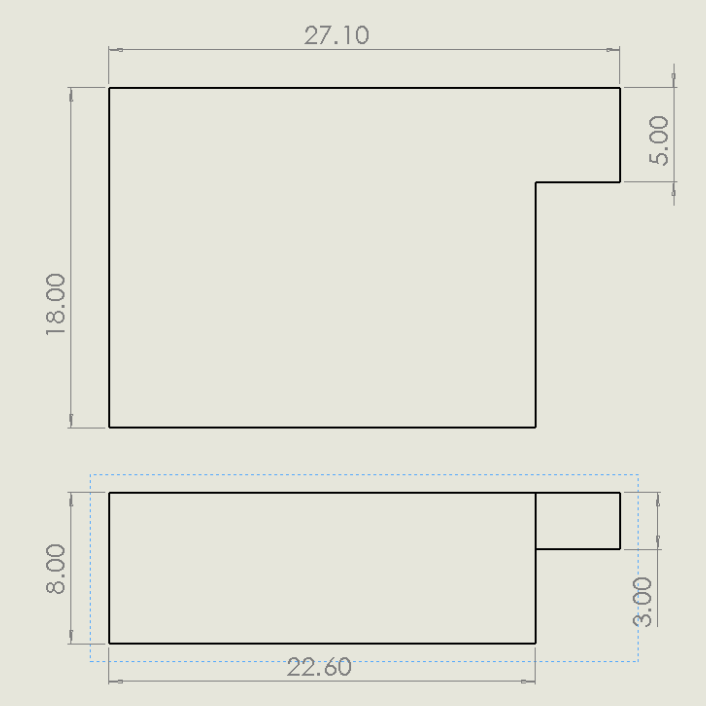


Figure 17: Stabilizing Block Technical Drawings

Button Caps:

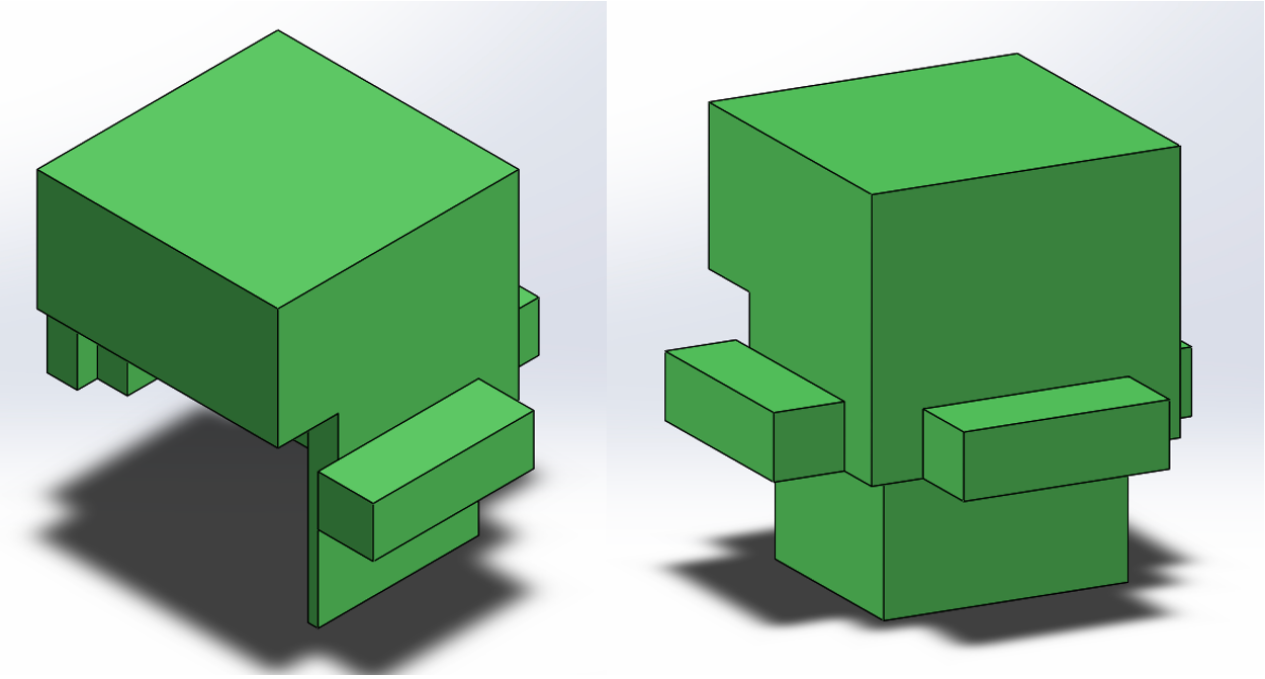
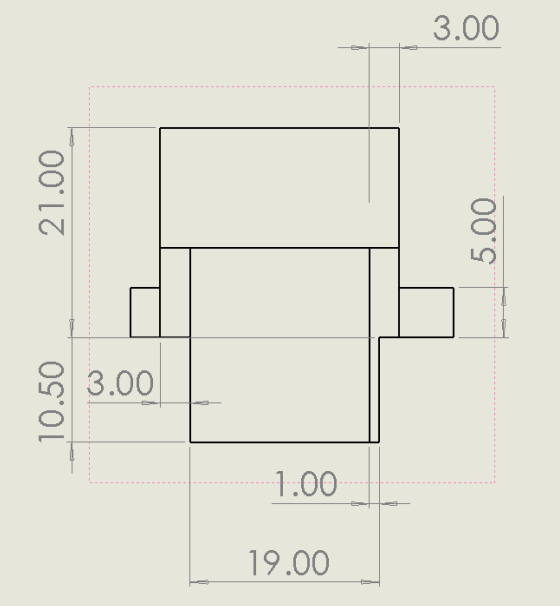


Figure 18: Button Caps Isometric View



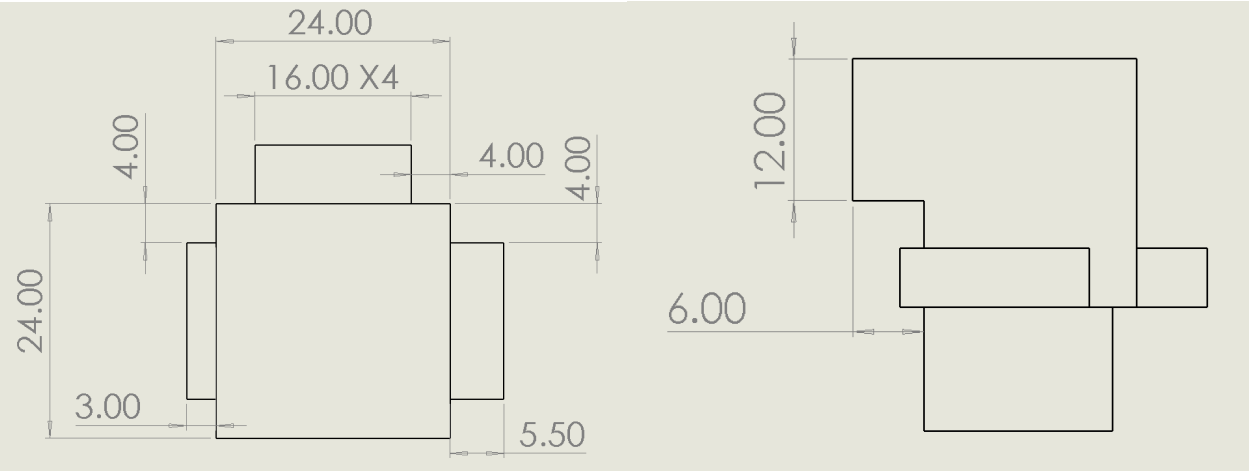


Figure 19: Button Caps Technical Drawings

Button Plate:

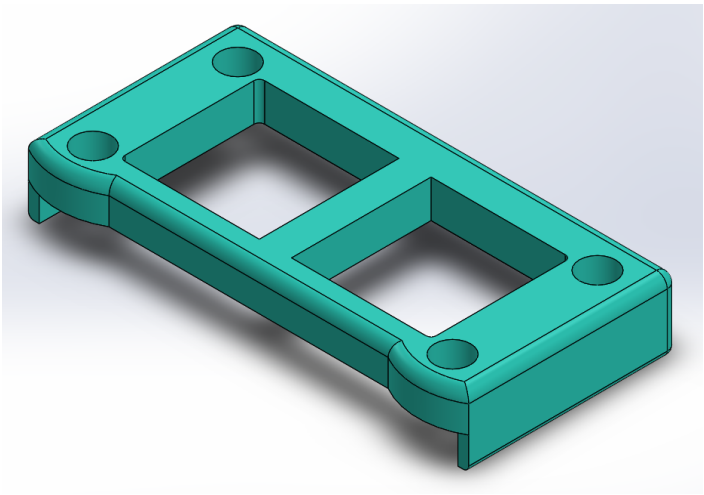


Figure 20: Button Plate Isometric View

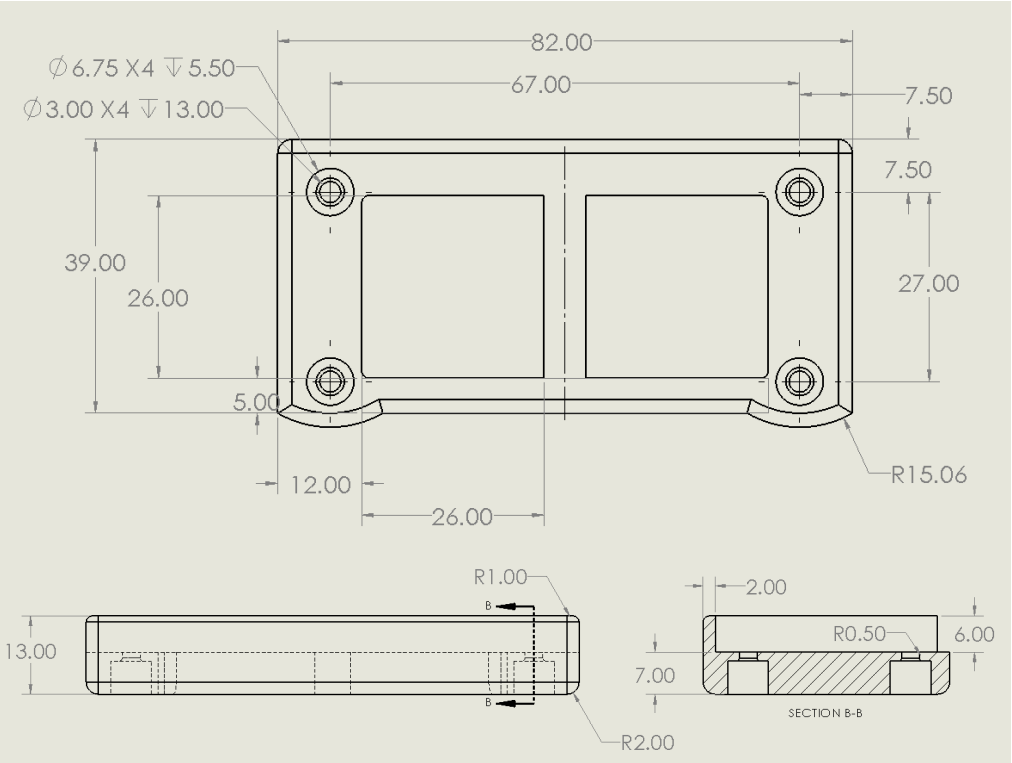


Figure 21: Button Plate Technical Drawings

3. Download the CAD files as .stl files.
4. Slice the .stl files using a 3D printing slicer software, such as UltiMaker Cura (downloaded from UltiMaker website shown below).

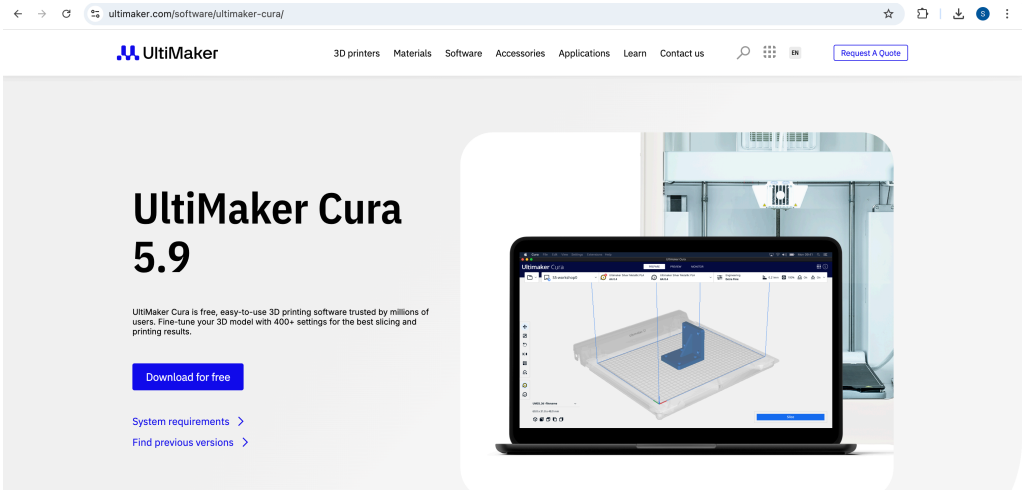


Figure 22: Downloading UltiMaker Cura Slicer (UltiMaker, 2024)

Customize the print settings, making reasonable assumptions to achieve reasonable print times. Nozzle sizes of 0.4mm or less are preferred. Some non-negotiables are:

   a. Use 80-100% infill for maximum strength.
   b. Enable support if part has portions that are “floating.”
5. Save each sliced file onto SD cards.
6. Insert SD card into a 3D printer of the correct nozzle size. Start the printer.
7. Once prints are finished, remove the supports as needed.
8. Assemble all the components.
   a. Insert Pro Micro microcontroller inside the casing.
   b. Insert the stabilizing piece to secure the microcontroller in place.
   c. Secure the buttons inside the button holder.
   d. Insert the button holder in the casing.
   e. Mount the springs on the two buttons.
   f. Cover the buttons with springs with the button caps.
   g. Hold all the components down with the button plate and screw the plate in.

6.4 Testing & Validation

Button Durability Test:

The durability of the button/spring system was tested by repeatedly pressing one of the buttons with a force of 9-10 N (a forceful press).

Table 6: Button/Spring Durability Test

Number of Clicks	Condition of Button / Spring System
10	The condition of the system remains unchanged in comparison to before the test.
50	
100	
150	

Impact Test:

10 drop tests were done to ensure that the clicker than withstand impact from being dropped from a height of 1m, which is approximately the height of the wheelchair arm (from the ground) where the clicker will be installed.

Table 8: Button/Spring Durability Test

Surface Type	Condition of Clicker after 5 Drops from a Height of 1m
Cardboard Surface	The condition of the clicker remains unchanged in comparison to before the test.
Hard Floor	

While durability tests were done, it is recommended that the clicker is not dropped or used aggressively to extend its lifetime.

7 Conclusions and Recommendations for Future Work

Throughout our project, we faced numerous challenges that tested our adaptability, resource management, and proactive planning. These experiences have provided us with valuable lessons and insights for the future.

1. **Lessons Learned:**
    a. **Adaptability:** Unexpected issues, such as the Bluetooth module failing to connect to the computer and the need to redesign our prototype because of updated client requirements, emphasized the importance of flexibility and responsiveness to changing needs.
    b. **Resource Management:** Running out of filament and encountering 3D printer malfunctions demonstrated the need for us to do contingency planning and maintaining backups to prevent delays.
    c. **Time Management:** The delays caused by troubleshooting the hardware and software side and redesigning taught us the importance of stricter timelines and early completion of tasks just in case there are any unexpected issues.
    d. **Technical Knowledge:** Deeper expertise in Bluetooth functionality and purchasing additional components for testing could have accelerated progress and improved outcomes, and we could have identified this earlier.
2. **Outstanding Issues:**
    a. **Bluetooth Functionality:** The module's failure to connect remains a critical issue that hindered the prototype's performance. Further exploration of hardware and software integration is necessary so we can achieve our original prototype plan.
    b. **Material and Equipment Limitations:** The malfunctioning 3D printer and lack of filament delayed testing and iterations, requiring us to do better resource allocation and backups.
3. **Implications for the Future:**
    a. **Proactive Planning:** Setting stricter timelines will ensure deadlines are met and provide more time for adjustments.
    b. **Early User Engagement:** Engaging with clients earlier in the process and conducting user testing will help clarify requirements and help us adjust in a timely manner.

c. **Incremental Goals:** Establishing clear milestones, such as achieving Bluetooth functionality in the first prototype, will allow for early identification and resolution of critical challenges.
d. **Skill Development:** Enhancing technical understanding, particularly of Bluetooth and related technologies, will improve the quality and reliability of future projects.
a. By implementing these strategies, we can minimize risks to improve project outcomes and better complete the user needs.

8 Bibliography

Arduino, T. A. (n.d.). *Software Downloads*. Arduino. https://www.arduino.cc/en/software

Ultimaker Cura. UltiMaker Cura 5.9. (2024, November 19).

https://ultimaker.com/software/ultimaker-cura/

APPENDICES

9 APPENDIX I: Design Files

Makerepo Link: https://makerepo.com/SophieKim/2232.duoclick

Table **7***. Referenced Documents*

Document Name	Document Location and/or URL	Issuance Date
Project Deliverable A	MakeRepo Project Files https://makerepo.com/SophieKim/2232.duoclick	Dec 3, 2024
Project Deliverable B, C, D	Same as above	Dec 3, 2024
Project Deliverable E, F, G, I	Same as above	Dec 3, 2024

10 APPENDIX II: Other Appendices