Accessible Gomoku

Alex Dickhans [*], Quinn Hedge [†], Justin Brooks [‡], Deanne Thomas [§]
Lecture Section 010, ECEN 1400

December 5, 2025

Abstract

The core aim of this project was to design and construct an accessible, inclusive game board for Gomoku (and potential other strategy games) that actively promotes community and socialization among people of all abilities. This was achieved through different accessibility features on both the actual game borad and the input devices. For output, the board will be both physical and tactile features via changes in textures and engraved shapes. For input, the user has the ability to plug in input devices that coincide with a given player's disability. We will create 2 input devices, one for fine motor controls and one for coarse motor controls. Due to cost and time constraints, we will limit the board to a 5x5 size, but we also plan to include multiple modes and documentation of the game.

[*] alex.dickhans@colorado.edu
[†] quinn.hedge@colorado.edu
[‡] justin.brooks-1@colorado.edu
[§] deanne.thomas@colorado.edu

Contents

1 Inspiration

Around the world, there are abundant examples of engineering projects designed to be accessible to people from all walks of life. These, while incredible and useful, often share a common and inadvertent side effect. When targeting accessibility for a given demographic, it is easy to create a project that is useful for one group and relatively ineffective for other groups. In this project, we hope to address this issue. This project is meant to be compatible with a variety of abilities and limitations, making it effective in communities and as a tool for social connections. By creating a clear and fun user interface, we hope to reduce barriers between those of different abilities.

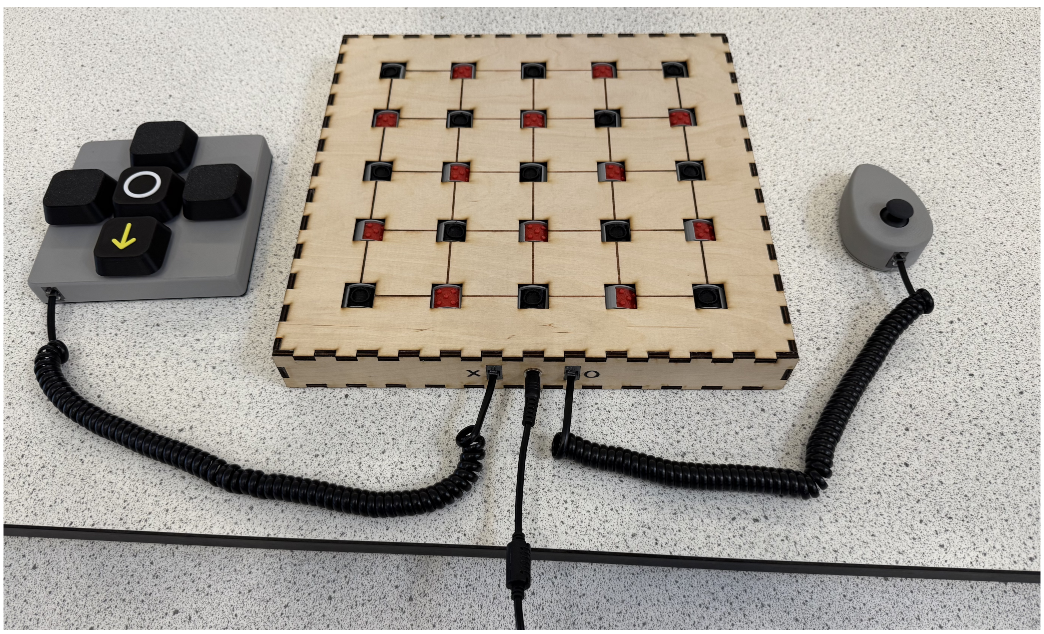


Figure 1: Final Gomoku Board with Coarse (left) and Fine (right) motor controllers.

2 Overview

2.1 Components

- **Controllers:** We developed two distinct input devices to demonstrate the flexibility of the game board in accommodating varied physical capabilities. One was created for fine motor control while the other for coarse motor input, thus showing how the system can be precisely tailored to meet specific user needs.
- **Gameboard:** The $5x5$ board dimension was specifically chosen to maximize versatility, enabling the users to play multiple different games beyond just Gomoku.
- **Tiles:** The playing tiles feature two distinct sides, which are easily identifiable through touch.

2.2 Team Assignments

- **Mechanical/Packaging:** Justin, Alex
- **Input Design:** Alex, Quinn, Deanne
- **Gameboard Electronics:** Alex
- **Software:** Deanne, Alex
- **Game Design/Rulebook:** Quinn

3 Technical Components

3.1 Design Overview

The game consists of two critical parts: the main board and the controllers. The wooden enclosure of the main board holds the 25 game squares, each composed of a servo, servo mount, and display tile. The controllers each have unique enclosures and electronics to serve their different purposes. The fine motor control has a sleek and ergonomic enclosure that contains a SparkFun joystick. The coarse motor control has a large, rectangular enclosure with large, easily distinguished buttons attached to key switched. Both controllers connect to the main board via RJ9 connectors, allowing 2-way UART communication that enables haptic response. The connection method creates an ease of development of new specialized controllers that connect via the same connector.

3.2 Main Board

The main board of the game is a focal point for our project, as it is where all the actual game play takes place. In order for this project to be successfully accessible, it is important that the main board epitomizes that, being as simple and direct to use and interface with as possible. The primary components of the board are the enclosure, game squares, and electronics, creating a tactile display that can be used to select a game mode, play a game, and display a victor.

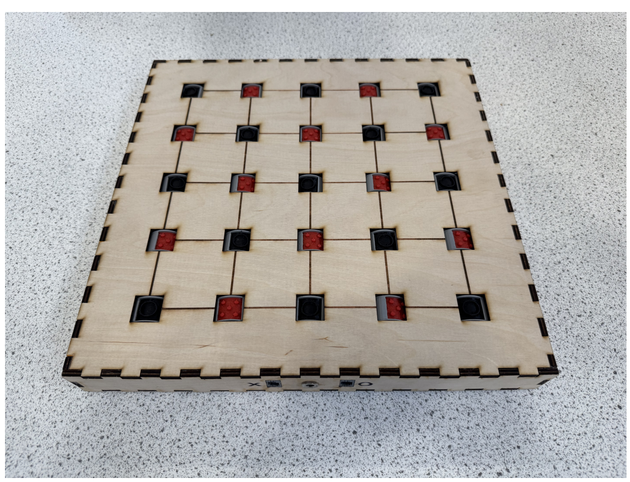


Figure 2: Completed main board assembly.

3.2.1 Enclosure

We constructed the enclosure of the main game board from laser-cut plywood. The sides of the board were individually cut with teeth so that the board could be easily glued together. The underside is not glued, but instead secured with screws for access and maintenance. The top of the board has an engraved design, reminiscent of the traditional Go board on which Gomoku is usually played. The 5X5 array of square holes in the top allows the display tiles to be inserted. One side of the board has holes designed to contain the power port and the ports for both controllers. The result is a clean, sleek board that is easy and aesthetically pleasing to use, with easy access for any changes of fixes that need to be made. The drawing can be seen in figure 3.

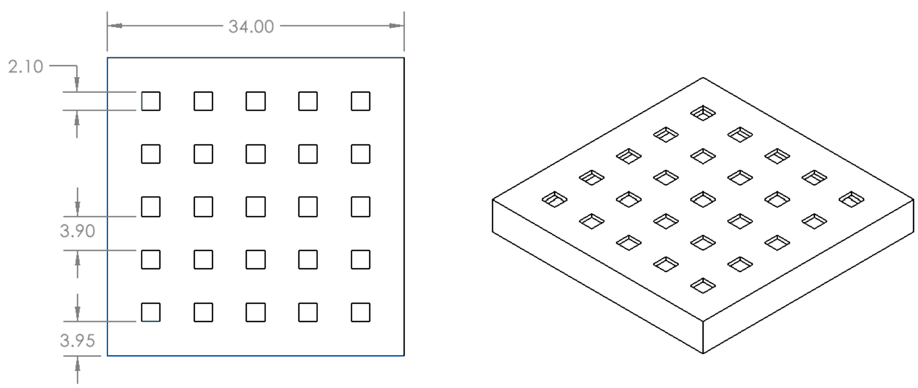


Figure 3: Main Board Enclosure

3.2.2 Game Square Design

Display Tile
For the display of each game square, we designed and 3d printed a rotating, tactile piece. One side represents "X" and is colored red, the other side represents "O" and is colored black, and the in-between is flat and gray. In game play, the display tile can be seen and felt, representing the tiles placed by each player in a Gomoku game, or lack thereof. The bottom of the tile has an inset to fit a servo horn, allowing easy attachment and replacement. The drawing can be seen in figure 4.

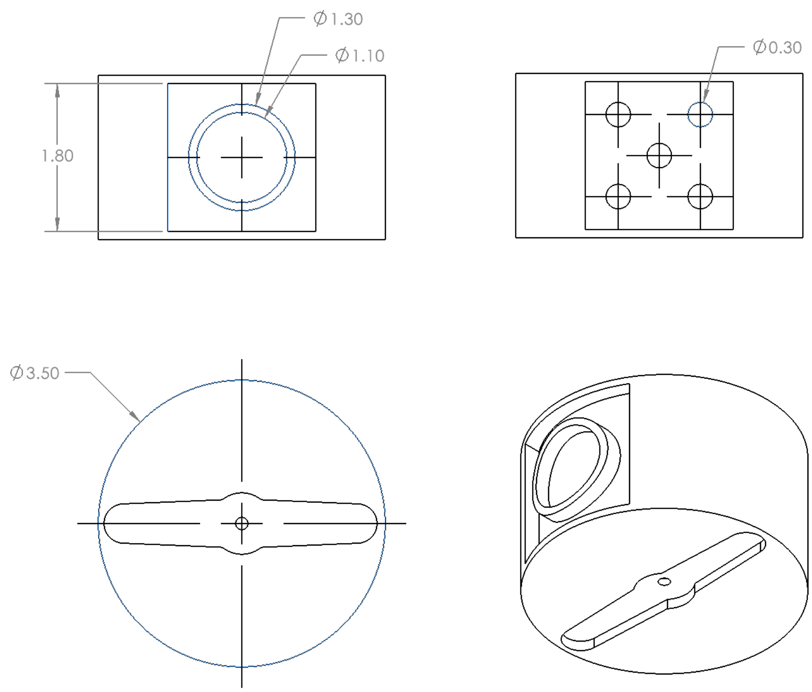


Figure 4: Servo-attached display tile

Servo Mount
To mount the piece to the board, we designed a 3d printed piece that allows the servo to snap in and out without any tools, allowing for much simpler repair and maintenance. The drawing can be seen in figure 5.

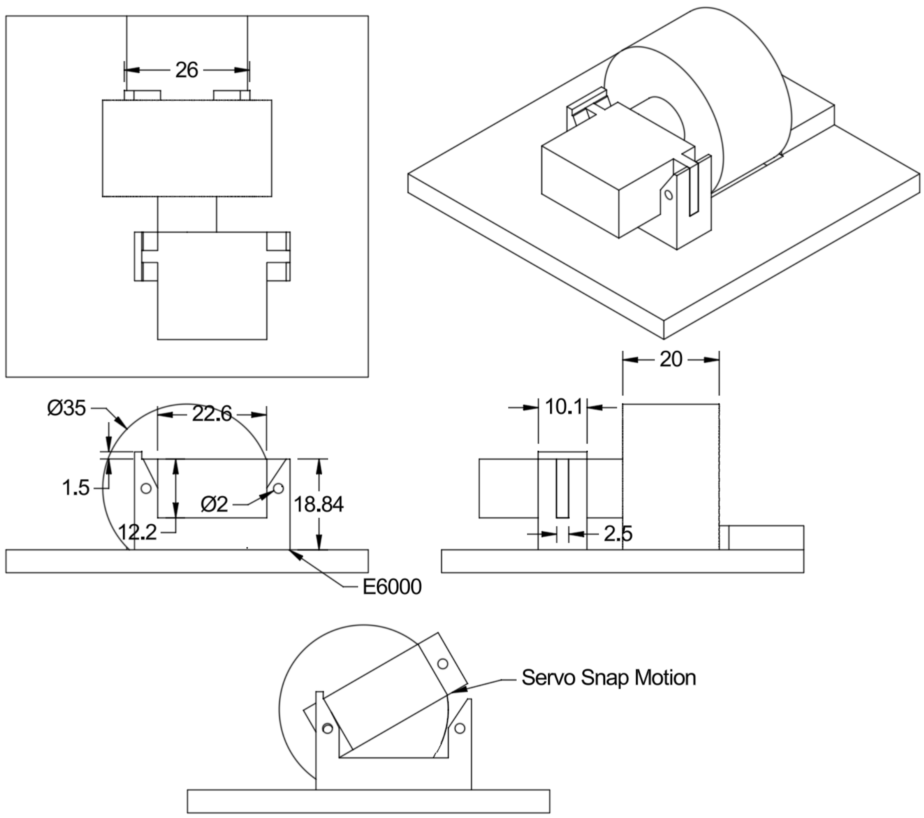


Figure 5: Servo snap-in mount drawing

3.2.3 Electronics

Electronic Component Choice

Component	Function / Specification	Justification for Selection
Arduino Mini	Atmega328P Microcontroller. Small footprint, 5V operation, 14 Digital I/O (6 PWM), 8 Analog Inputs.	Selected for small size, and more than enough I/O for the limited communication needed (UART/I^2C).
PCA9685 Module	I^2C PWM controller. Controls up to 16 PWM outputs with power distribution.	Required to control all the servos on the main board with the limited IO on the Atmega328P. Provided a much simpler and more elegant solution than changing microcontrollers.
5V Power Supply	5V 10A generic power supply with barrel jack.	Provides the necessary power source for the game board. The current limit of 10 A was plenty to control up to 15 of the servos simultaneously.
RJ9 Female Connectors	Allows power and data transmission over COTS telephone cables.	4P4C connector allows for GND, 5V, TX, and RX wires.
RJ9 Wires	COTS RJ9 telephone wires.	Tested, flexible, robust, easy to use connector that has well understood preformance.
Servo Wire Expanders	Adds 10cm of length to PWM servo wires (SIG, 5V, GND).	10cm was the perfect distance to allow almost every connection to use 1 or less expander, and only 2 needed to use multiple expanders.

Table 1: Main board electrical component selection

Schematic Design

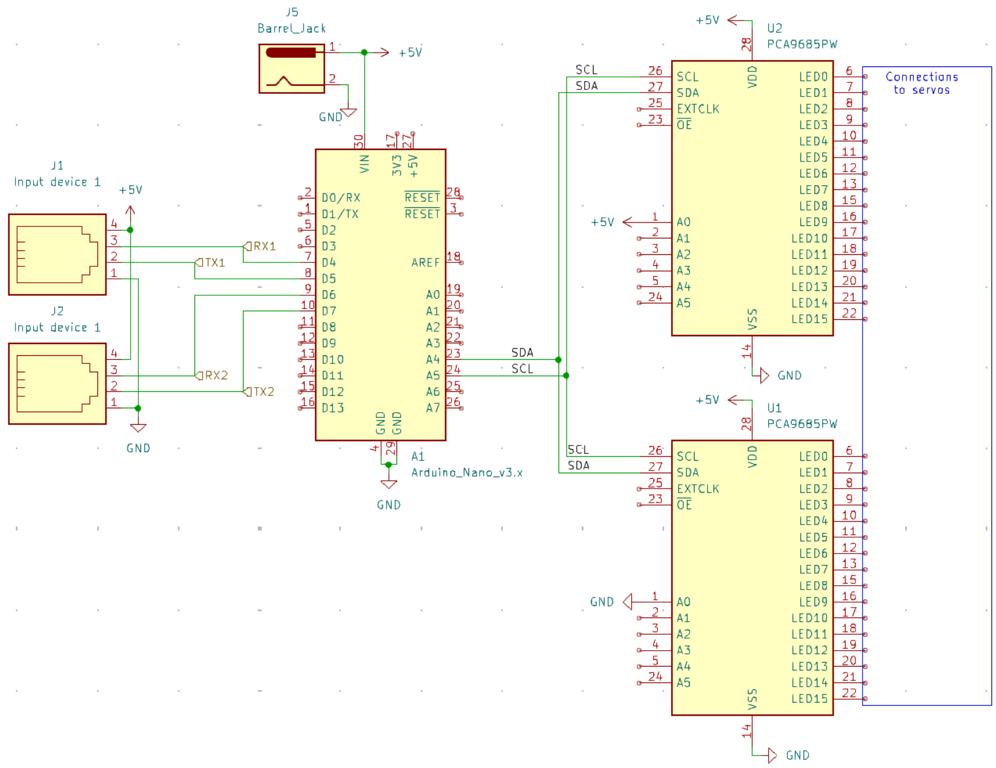


Figure 6: Main Board Simplified Schematic

3.2.4 Code

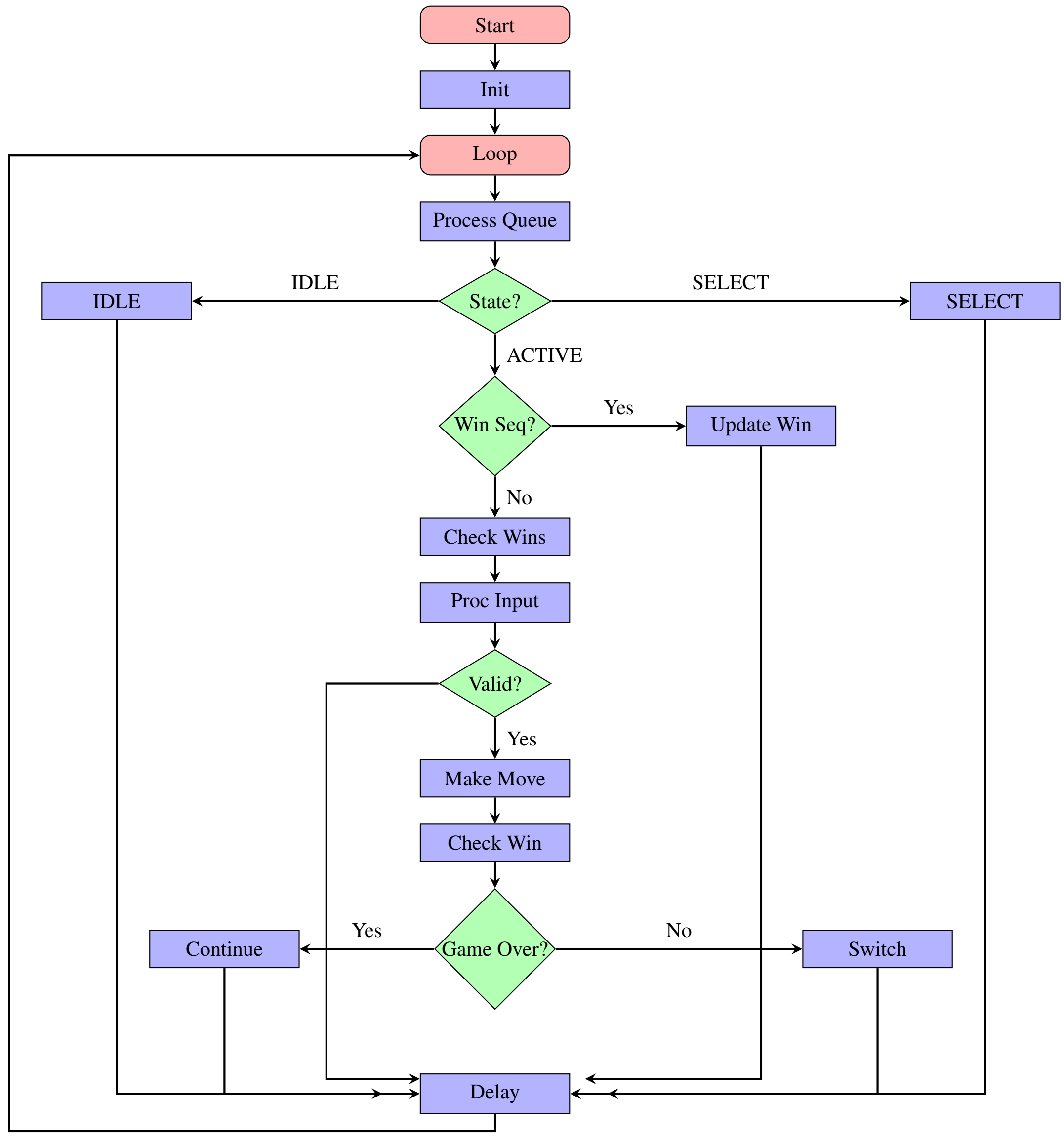


Figure 7: Main board flow chart

Communication Protocol
For each of the controllers, there will be an RJ9 Connector with power and UART. Power, GND and 5V on pins 1 and 2, respectively, have a current capacity of up to 100mA. The UART, TX and RX are on pins 3 and 4 of the RJ9 connector respectively. UART communication is character-based has the following data communication for the input and haptic feedback.

Cmd	Description
Main Board → Player Controller (Haptics & Status)	
`'y'`	**Yes / Valid Move**: Triggers a short haptic pattern (`...`) to indicate a successful action or valid move.
`'n'`	**No / Invalid Move**: Triggers a long haptic pattern (`--`) to indicate an error or invalid move (e.g., space occupied).
`'t'`	**Turn Notification**: Triggers a signal pattern (`-.-`) to notify the player it is their turn.
`'w'`	**Win Notification**: Triggers a specific win haptic pattern (`....-`).
`'l'`	**Lose Notification**: Triggers a specific lose haptic pattern (`--..`).
Player Controller → Main Board (Game Inputs)	
`'u'`	**Up**: Move cursor up / Navigate menu.
`'d'`	**Down**: Move cursor down / Navigate menu.
`'l'`	**Left**: Move cursor left / Previous game mode.
`'r'`	**Right**: Move cursor right / Next game mode.
`'s'`	**Select**: Confirm selection / Place piece.

This communication protocol makes it easy to develop new devices that fit seamlessly into the existing board and is able to fully interface with all the features.

3.2.5 Manufacturing/Assembly

1. Laser cut the pieces of the board from 3/16 in wood
2. 3D print the individual components (display tiles, servo mounts, corner screw points)

3. Use a utility knife to chamfer 2 of the edges of each display tile square, as shown in figure 8. Chamfer must allow room for the display tiles to spin, but not too much so that it effects the look of the front of the board.

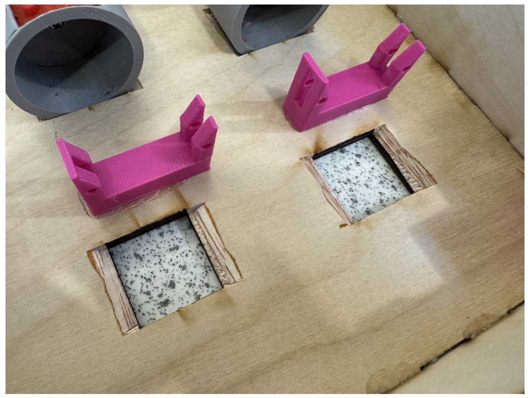


Figure 8: Manually-cut chamfers of each board square. Note: the alignment of the chamfers.

4. Attach all of the servo horns to the servos in a consistent position that allows for both sides of the game piece to be shown. Ensure all pieces are consistent.

5. Glue each of the servos onto the board in a consistent orientation as shown in figure 9.

6. Route wires for each of the servos under the next one in the row, as shown in figure 9.

Figure 9: Wires for servos routed under the row.

7. Solder power wires to each individual component (PCA9685 modules, and Arduino Mini).
8. Bridge the first solder pad for one of the I^2C address pins on the PCA9685 device.
9. Solder wires for I^2C between the Arduino Mini and the PCA9685 Modules according to the schematic.
10. Glue the sides of the enclosure on the bottom. Ensure that the barrel jack is connected on the side opposite to that of where the servos are pointing (see figure 10). Wait for glue to cure.

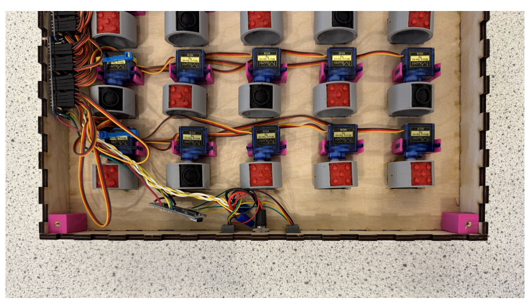


Figure 10: Side piece glued to the top of the board with servos. Note the side where the barrel jack was placed.

11. Insert threaded inserts into the corner pieces and glue them into the board waiting for them to cure. Place as in figure 11.

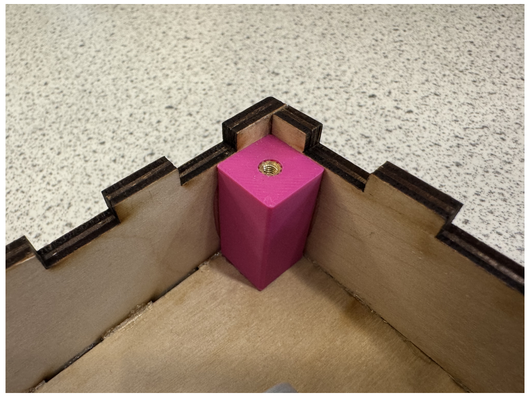


Figure 11: Screw attachment points, glued in with the brass threaded inserts.

12. Use wire nuts to connect the power cables together inside the board.
13. Connect the wiring for all the servos to the PCA9685. Look at figure 12 for wiring reference.

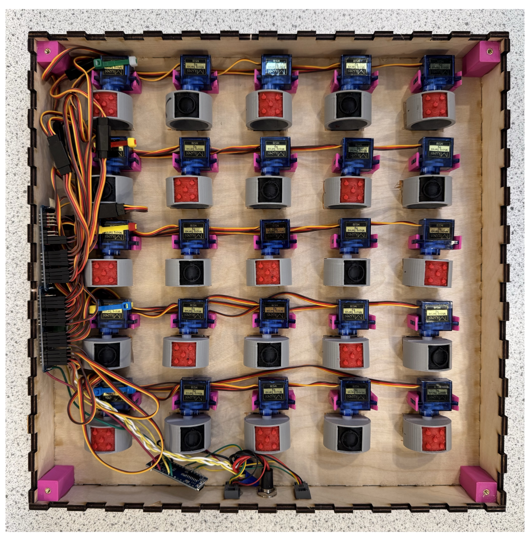


Figure 12: Main wiring. Note the organization of the wiring under the servos.

14. Screw on the bottom side of the box using M3 screws, while making sure not to pinch any wires. See figure 13.

Figure 13: Bottom of main board, with M3 screws attached.

3.3 Fine Motor Input System

The fine motor input is the controller catered towards users that have fine motor capabilities. The purpose of this in an accessibility-focused project is as a proof of our over-all goal; to provide a smooth, fun experience for all. Having different input systems demonstrates the potential for more to be developed and the way multiple different input systems can interact and work together. This controller is designed to be ergonomic and simple to use, featuring a joystick that allows the user to move their "cursor" on the board in 8 directions and select a tile by pressing down on the joystick. The controller also features haptic response to indicate moves and turns, and accomplishes 2-way communication via an RJ9 connector. A picture of the controller can be seen in figure 14.

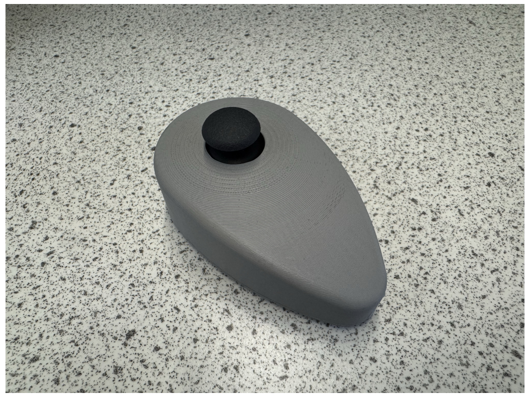


Figure 14: Fine motor controller final product

3.3.1 Enclosure/Packaging

The enclosure for the controller was designed and 3d printed. The rounded, tapering body was designed to be easy and comfortable to hold, providing an ergonomic experience. The base of the controller has a raised section on which the joystick is installed using screws. The enclosure contains the rest of the electronics with the bottom plate being attached with screws. There is a port at the back of the controller to contain the female RJ9 connector. The drawing can be seen in figure 15.

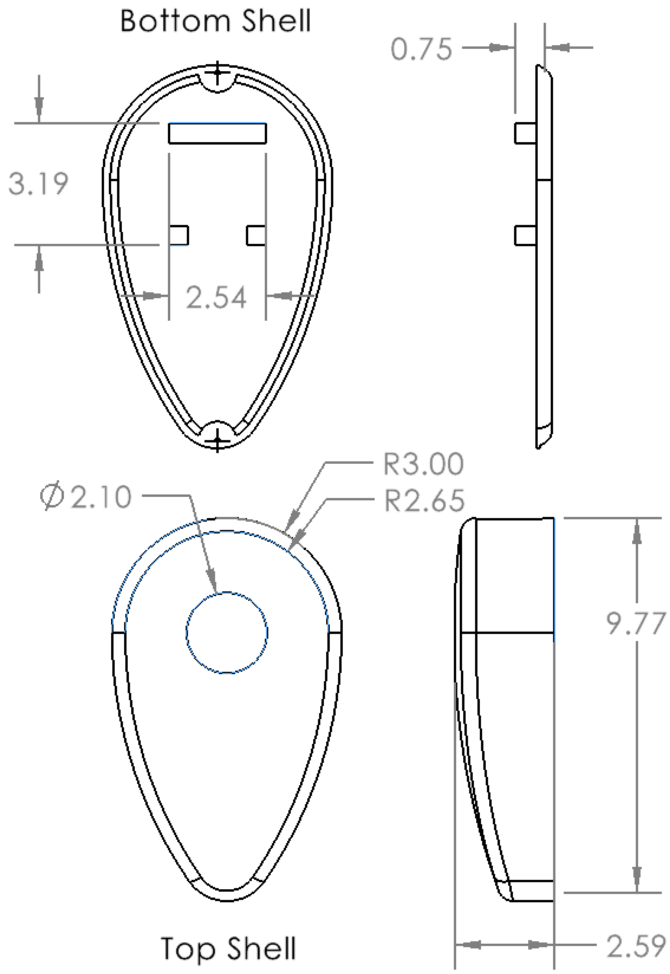


Figure 15: Fine Motor Input Casing

3.3.2 Electronics

Electronic Component Choice

Component	Function / Specification	Justification for Selection
Arduino Mini	Atmega328P Microcontroller. Small footprint, 5V operation, 14 Digital I/O (6 PWM), 8 Analog Inputs.	Selected for built-in analog capabilities and UART communications.
Sparkfun Joystick Module	2D analog joystick with press-down switch	Allows for the user to intuitively input what the desired movement direction is. COTS module allows for well-tested performance.
RJ9 Female Connectors	Allows power and data transmission over COTS telephone cables.	Selected for power and communication to main board.
Haptic Motor	COTS haptic motor, similar to commercial game controller haptics.	Selected for for easy integration and reliable haptic feedback for user.
NPN Transistor (2N2222A)	Low-power transistor.	Selected for higher current control of haptic motor.

Table 2: Fine controller electrical component selection

Schematic Design

For fine motor control, we are effectively only combining 3 components, the RJ9 connector for communication, a haptic motor (with transistor for power), and an analog joystick with digital push button. All of these components, with the modules, only need to be connected via direct traces, no additional passives are needed.

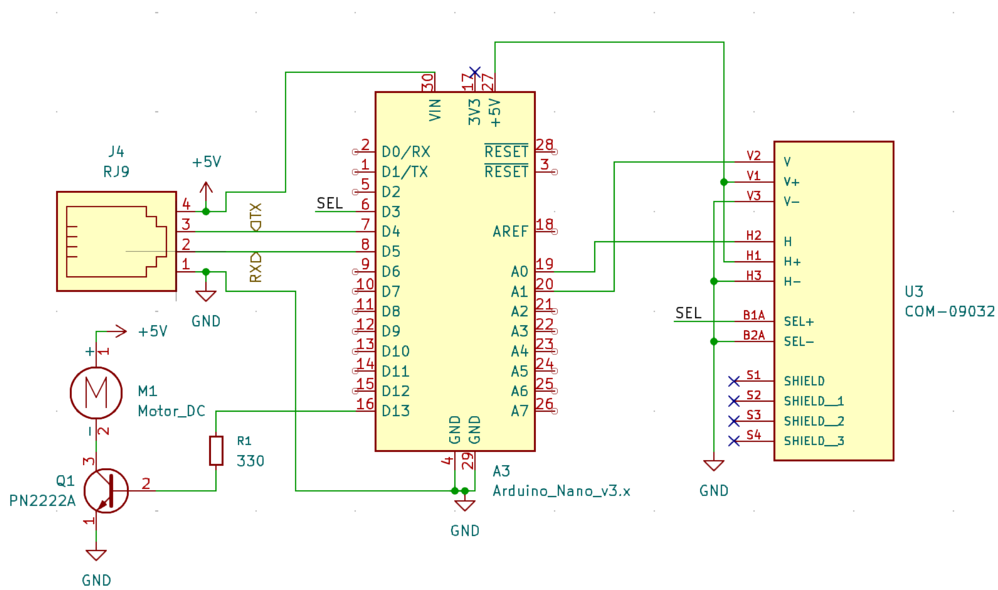


Figure 16: Fine Motor Control Simplified Schematic

3.3.3 Code

The code for the fine motor control has 2 main components: reading and transmitting data from the joystick, and controlling the haptics in the controllers. To read data from the joystick the code uses analog inputs, which it then debounces (ensures no accidental double presses) and sends simple UART characters to the main controller to indicate a move has been made. For the haptics, it has a variety of 1-character haptic patterns in it's memory, which it receives from the main board. These commands are then processed and used to move the haptic motor at the right times. A detailed flow chart of the code is in figure 17.

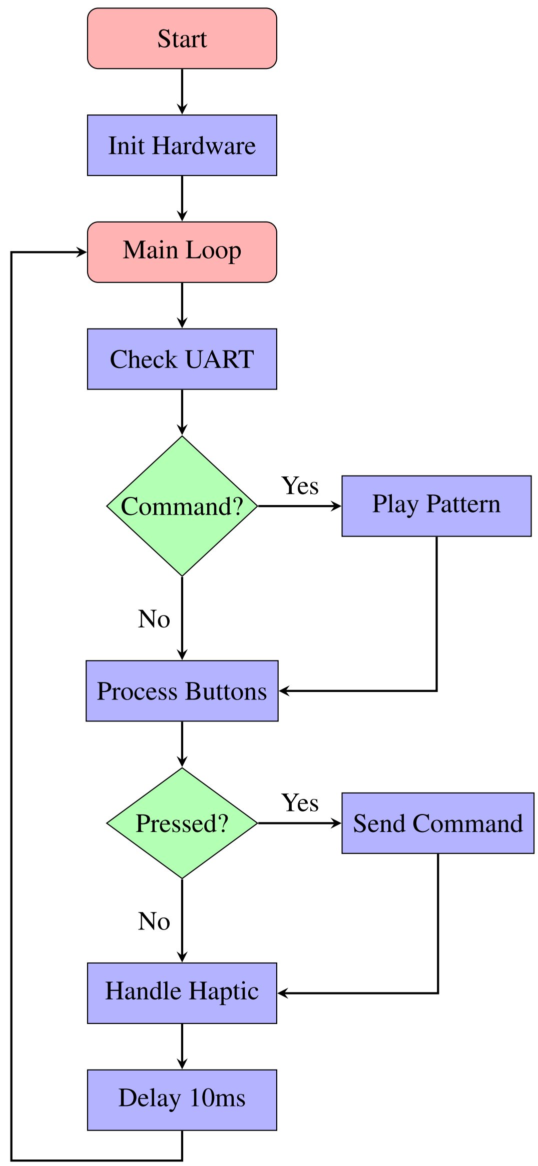


Figure 17: Fine Motor Software Flowchart

3.3.4 Manufacturing/Assembly

1. 3D print the individual components (top/bottom casings).
2. Make holes for the joystick screws.

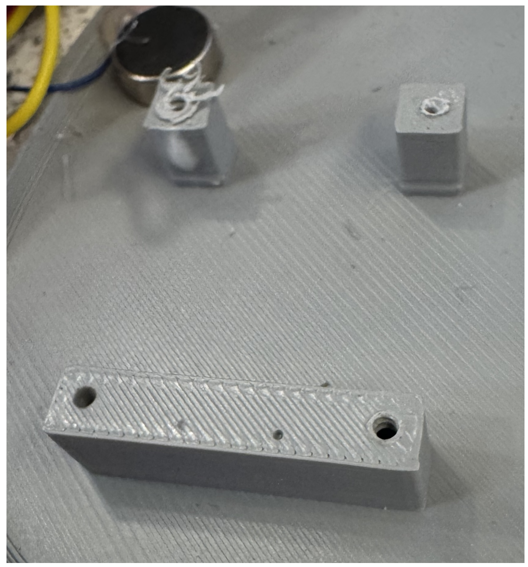


Figure 18: Holes in the bottom housing to hold the joystick.

3. Solder Connections to RJ9 connector, haptic motor, transistor, and joystick to arduino mini according to the schematic.
4. Screw in Joystick using self-tapping screws.
5. Attach and secure haptic motor and wired connections with hot glue.
6. Press-fit RJ9 connector into top casing. Should look similar to figure 19.

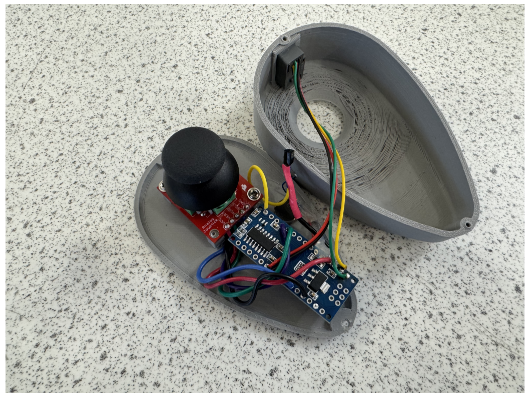


Figure 19: Fine controller insides.

7. Screw in top and bottom board components with self-tapping screws.

3.4 Coarse Motor Input System

The coarse motor input is the controller catered towards users that have coarse motor capabilities. The purpose is to provide an input method that requires minimal precision, allowing individuals with limited motor control to interact with the game effectively. This controller features 5 large keyboard-style keys, up, down, left, right, and a central key for select. Like the fine motor system, it demonstrates our goal of making an inclusive experience through various input systems. This controller also features haptic response to indicate moves and turns, and accomplishes 2-way communication via an RJ9 connector. A picture of the controller can be seen in figure 20.

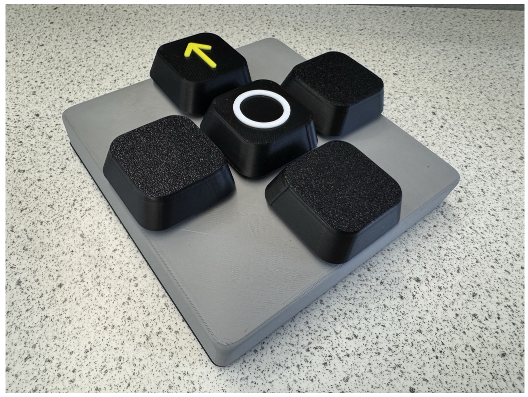


Figure 20: Coarse motor controller final product

3.4.1 Enclosure/Packaging

We decided to fully 3D print the enclosure for the coarse motor control in 2 separate parts. The top part hold the keyswitches and keycaps, along with allowing room for the arduino and other electrical components. The enclosure is closed up using a flat 3d-printed 3 mm bottom plate. The entire enclosure was printed out of PLA on an BambuLab A1-mini with no special settings.

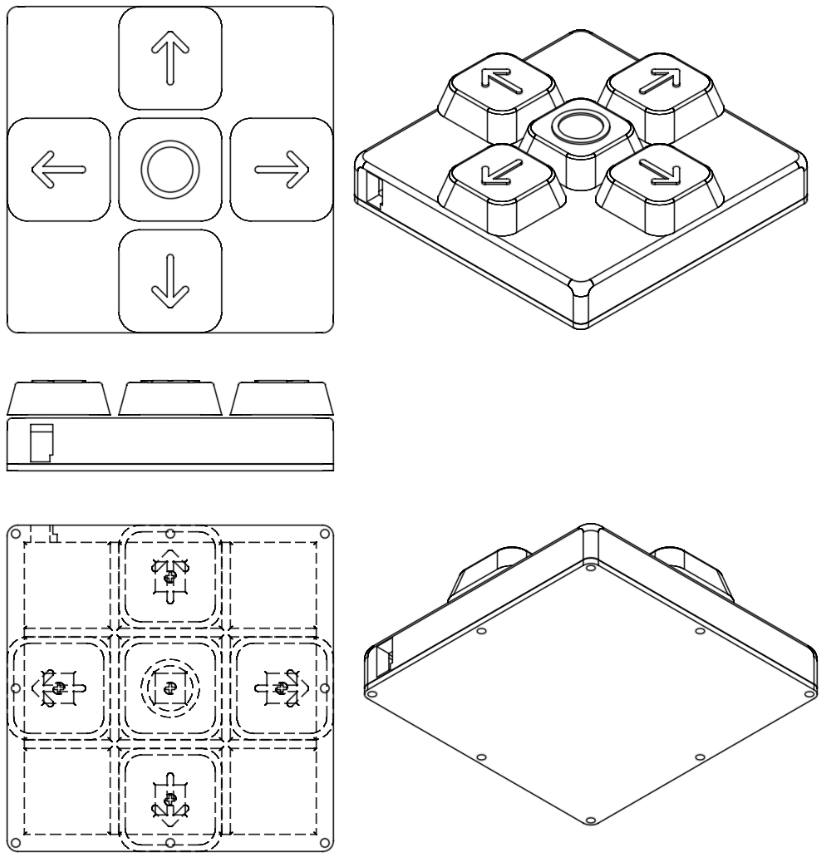


Figure 21: Coarse motor control drawing

The keycaps were printed using multi-color functionality of the BambuLab A1-mini and attached directly to the keyswitches using the cross connection on the bottom. The drawing of the keycap is shown in figure 22.

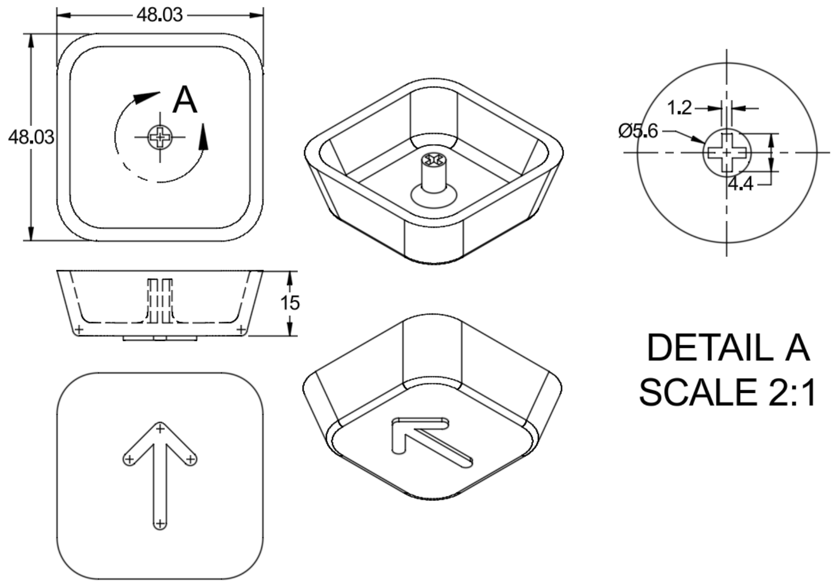


Figure 22: Drawing of keycap with arrow

3.4.2 Electronics

Electronic Component Choice

Component	Function / Specification	Justification for Selection
Arduino Mini	Atmega328P Microcontroller. Small footprint, 5V operation, 14 Digital I/O (6 PWM), 8 Analog Inputs.	Selected for built-in analog capabilities and UART communications.
Cherry MX Blue Switches	Mechanical clicky keyswitch.	Selected for clicky, audible feedback and easy mechanical integration.
RJ9 Female Connectors	Allows power and data transmission over COTS telephone cables.	Selected for power and communication to main board.
Haptic Motor	COTS haptic motor, similar to commercial game controller haptics.	Selected for for easy integration and reliable haptic feedback for user.
NPN Transistor (2N2222A)	Low-power transistor.	Selected for higher current control of haptic motor.

Table 3: Coarse electrical component selection

Schematic Design

For coarse motor control, we are effectively only combining 3 components, the RJ9 connector for communication, a haptic motor (with transistor for power), and keyswitch digital push buttons. All of these components, with the modules, only need to be connected via direct traces, no additional passives are needed.

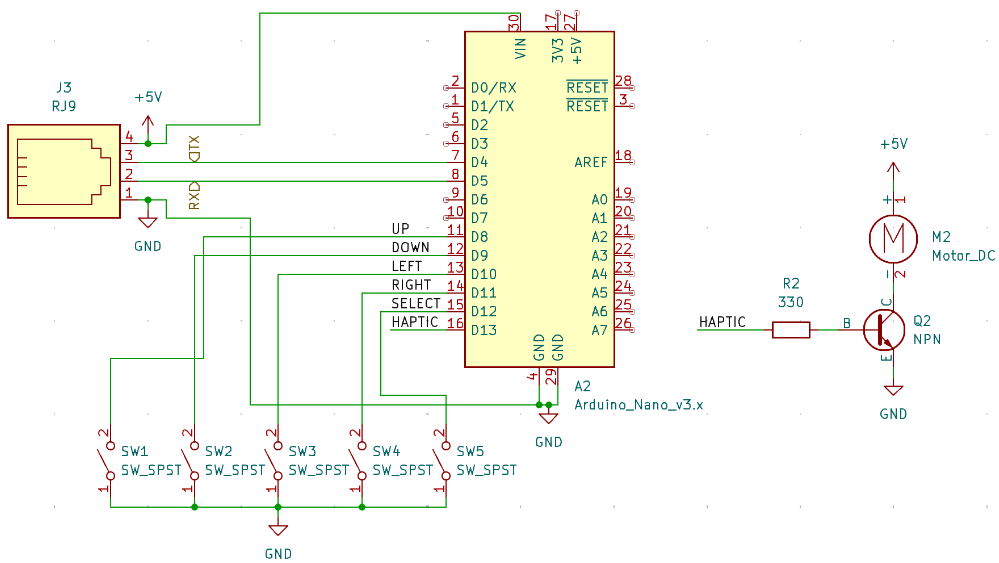


Figure 23: Coarse Motor Control Simplified Schematic

3.4.3 Code

The code for the coarse motor control has 2 main components: reading and transmitting data from the buttons, and controlling the haptics in the controllers. To read data from the buttons the code uses digital inputs, which it then debounces (ensures no accidental double presses) and sends simple UART characters to the main controller to indicate a move has been made. For the haptics, it has a variety of 1-character haptic patterns in it's memory, which it receives from the main board. These commands are then processed and used to move the haptic motor at the right times. A detailed flow chart of the code is in figure 24.

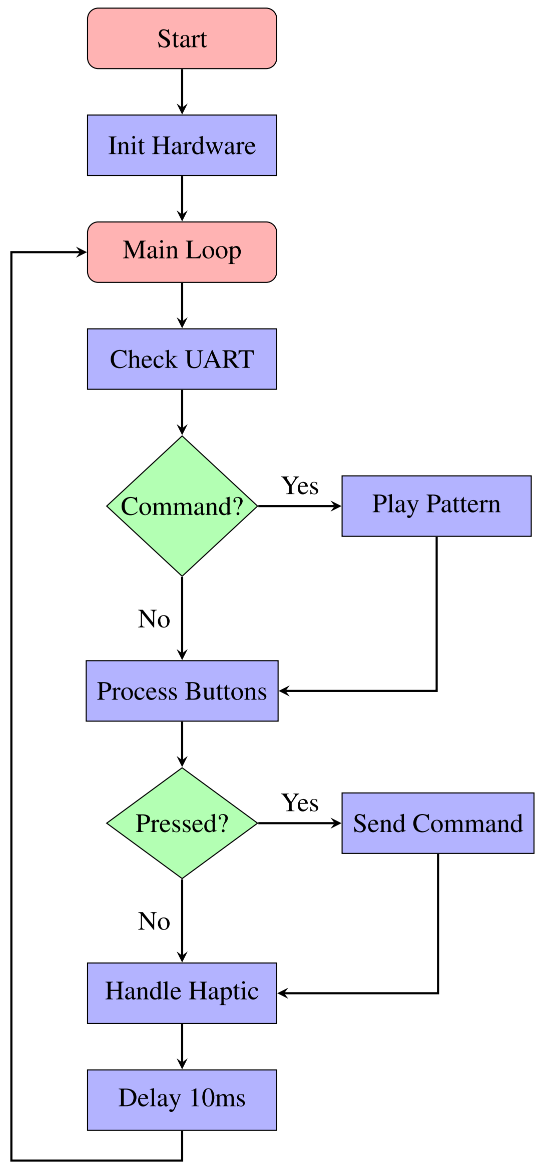


Figure 24: Coarse Motor Control Flow Chart

3.4.4 Manufacturing/Assembly

1. 3D print the individual components (top/bottom casings, keycaps)
2. Use soldering iron to heat up and insert threaded inserts into top casing. See figure 25.

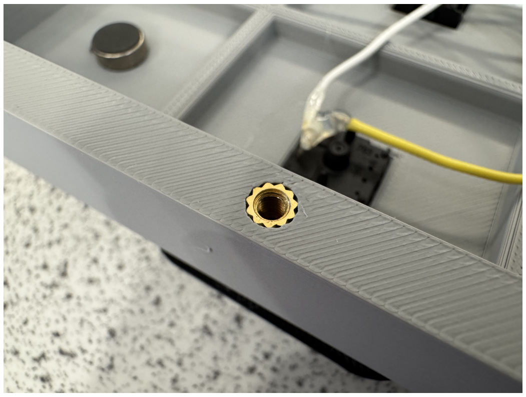


Figure 25: Threaded inserts, melted into the coarse motor control top casing.

3. Snap keyswitches and press-fit RJ9 connector into top casing.
4. Solder connections to keyswitches and RJ9 and their respective pins in the schematic.
5. Solder Haptic connection to transistor and pins on Arduino Mini.
6. Attach and secure haptic motor and connections with hot glue. See final inside assembly in figure 26.

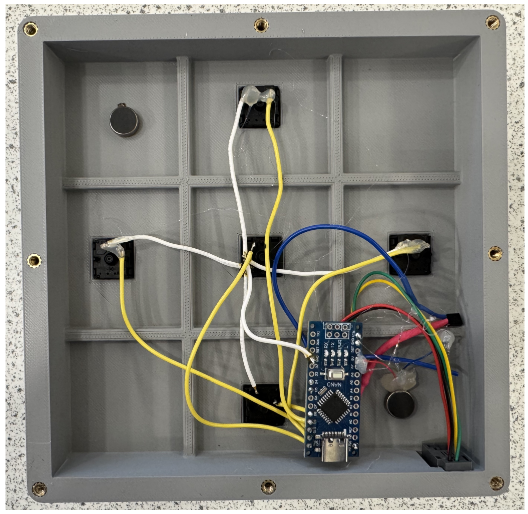


Figure 26: Coarse motor controls inside wiring.

7. Screw in top and bottom casings with M3 screws. At this step it should look like figure 27.

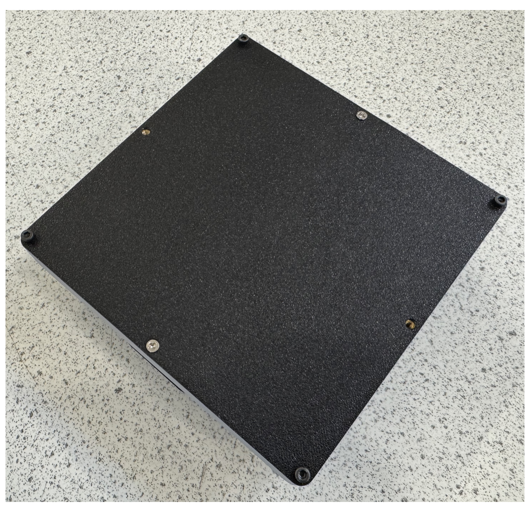
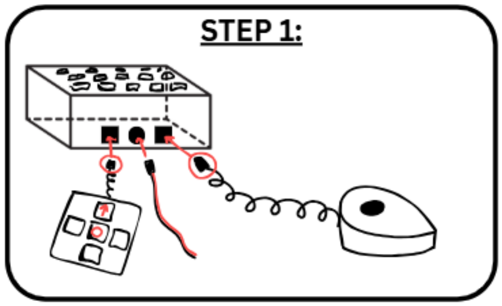
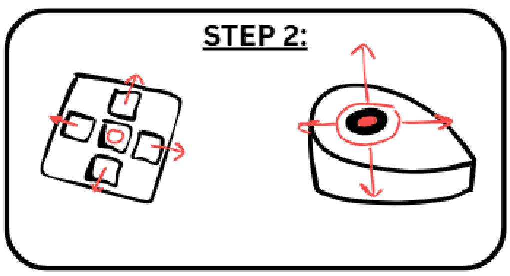


Figure 27: Coarse motor control bottom after being closed up.

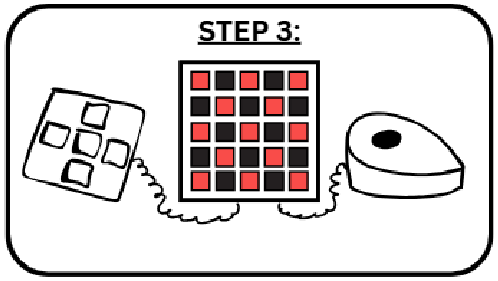
4 Gameplay Guide



To set up the game, players must plug their respective controllers into the RJ9 ports on the gameboard, along with plugging in the power cable. The board should have an initial state where all 25 game squares are clear (displaying gray faces).



Next, players will move their controllers left or right to select the desired game mode they would like to play. When one is chosen, the player should press the select button on their controller to confirm the game mode. Then, a game will start. Players will use their controllers to move their "cursor" around the board until they choose a spot to play. They will then use the select button to "place" their piece (select their tile). Once an available tile is selected, it will rotate to represent that player's assigned symbol and color (player 1 will have a red X, and player 2 will have black dots). The controllers also have a haptic feedback system built in. A short vibration will indicate that a piece was placed or a valid move occurred. A longer vibration will indicate that the chosen tile is already occupied. The haptics will also vibrate to let a player know it's their turn.



Players will then pass turns back and forth until a win condition or a tie is met. Once this happens, the board will display the victor, and the game will be over.

5 Bill of Materials

Item	QTY	Price	Subtotal
Board Squares	25	5.00	125.00
Micro-controller	3	4.00	12.00
PCA9685 Modules	2	6.00	12.00
1/8" Plywood	1	10.00	10.00
3D Printed Components	150	0.02	3.00
Joystick	1	10.00	10.00
Key-switches	6	1.50	9.00
Haptic Motors	5	0.80	4.00
RJ9 Connectors/Wires	2	2.00	4.00
Total			189

6 Conclusion

The goal of this project was to design and construct an accessible and inclusive game board for Gomoku, with a focus on fostering community among people of all abilities. Specifically, we wanted to address the issue where accessibility-minded solutions, when focused on a target demographic, can become relatively ineffective for other groups. This was achieved through different accessibility features on both the game board and input devices. For output, the main board provides tactile features via engraved shapes and moving pieces. The playing tiles feature two distinct sides that are easily identifiable through touch. For input, the user has the ability to plug in the chosen input device that best suits their abilities. We developed two unique input controllers to accommodate these different levels of ability: one utilizing a joystick for finer motor controls, and the other utilizing large keyswitches that are easier to press. Both controllers connect to the main board via RJ9 connectors, allowing 2-way communication. The use of RJ9 connectors also allows for the development and integration of new input devices that can accommodate other disabilities. By creating a system compatible with a variety of abilities and limitations, this project has the potential to expand the cross-playability for diverse communities.

7 Appendix A: Code

Code can be found on GitHub alexDickhas/ecen1400-final.