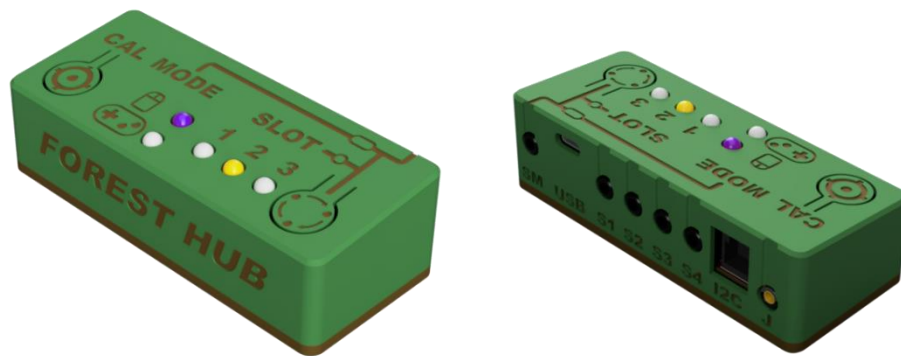


Forest Hub

DESIGN RATIONALE

Introduction

The Forest Hub is an open-source interface that connects to an analog joystick and one or more assistive switches and emulates an HID Mouse and/or HID Gamepad. The device is intended for people who are unable to use traditional input devices like a computer mouse or gaming controller.



Forest Hub

DESIGN RATIONALE

Research

Existing Commercial Options

Title	Quester-switchbox
Link	https://www.pretorianuk.com/quester-switchbox
Author	
License	
Cost	
Notes	

DIY Designs

Title	Enabled Controller Mini
Link	https://github.com/milador/Enabled-Controller-Mini
Author	Milad Hajihassan
License	CC-BY-NC-SA
Cost	
Notes	

Title	Enabled Controller
Link	https://github.com/milador/Enabled-Controller
Author	Milad Hajihassan
License	CC-BY-NC-SA
Cost	
Notes	

Title	Enabled Controller Wireless
Link	https://github.com/milador/Enabled-Controller-Wireless
Author	Milad Hajihassan
License	CC-BY-NC-SA
Cost	
Notes	

Forest Hub

DESIGN RATIONALE

Requirements

Goals

G01	Cost-effective
G02	Maker Manufacturable
G03	

Functional Requirements

F01	Able to operate as both a USB Mouse and a USB Joystick
F02	User should have ability to adjust cursor speed.
F03	User should have ability to initiate calibration.

Non-functional Requirement

NF01	
NF02	
NF03	

Constraints

C01	10 x 10 cm Maximum PCB Size for cost-effective manufacturing
C02	8 x 10 cm Maximum PCB size for free version of Eagle

Forest Hub

DESIGN RATIONALE

Ideation

Key Features

Number of Ports – how many inputs are available for assistive switches

- 4 + 1 for mode/slot switching

Device Buttons – what controls are available on the device itself?

- Mode Switch
- Calibration

Mouse function

- Core Functions
 - o Left click
 - o Right click
 - o Scroll
 - o Middle click
- Additional functions
 - o Double click

User feedback – what indicators are required for the user to determine the state and/or settings of the device?

- Power (is the device powered / on)
- Output Mode (is the device outputting Mouse or Gamepad)
- Slot / Level / Profile (different

Bluetooth Indicator?

User feedback methods – how are the indicators conveyed?

- Individual lights / LEDs
- LED colour
- Display / LCD
- Sound

Slots – enable the user to change between different settings / mappings

- Two Output Indicators and Three Slot Indicators
 - Output Indicators
 - o Mouse

Forest Hub

DESIGN RATIONALE

- Gamepad
- Three Slot Indicators
 - Slot 1
 - Slot 2
 - Slot 3

Calibration – how is a joystick calibrated when attached? When swapped?

- Neutral position calibration
- Extents calibration

Customizability

How can the device be customized to tailor it to the user?

- Enclosure color
- Control the color of the neopixels
- Control the color and pattern of the microcontroller neopixel
 - Graphic
 - Custom lithophane
-

User Experience

Adjust Mouse

- Adjust cursor speed

Switch Slots

- Short press to switch slots
- Long press to switch between mouse and joystick
- LEDs Indicating slot and output mode

Calibration

Short press Calibration button to start neutral position reset.

Long press calibration button to start extent calibration

Set Neutral position on startup

Forest Hub

DESIGN RATIONALE

Layout

- USB in same direction as the inputs
- USB in other direction as the inputs

Mode switch / button / lights in proximity

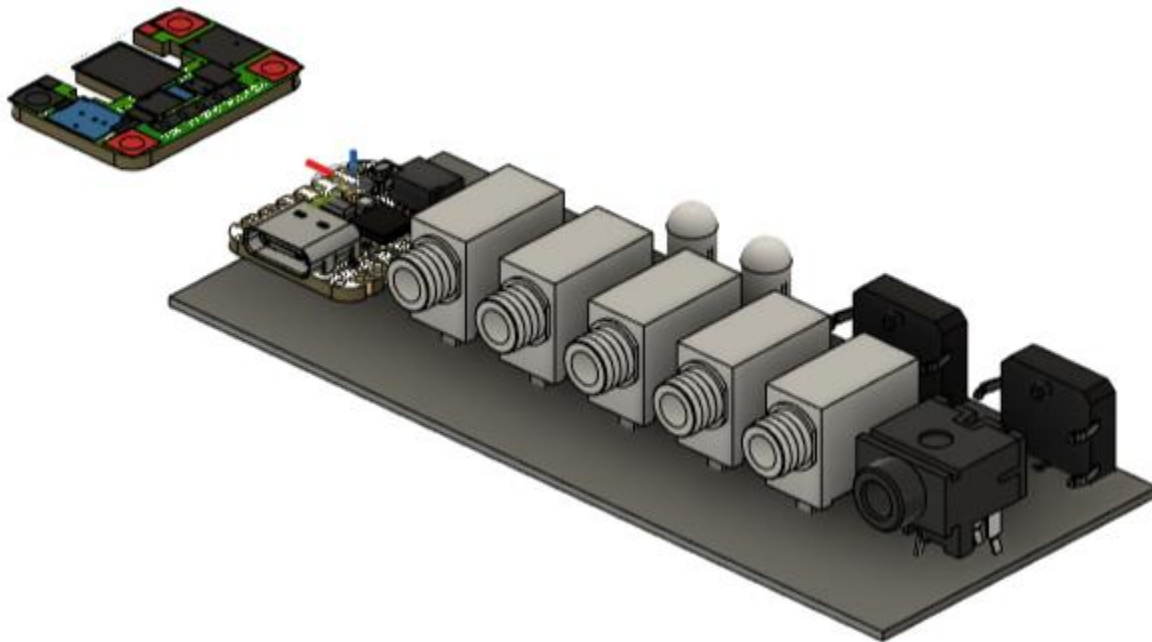
Forest Hub

DESIGN RATIONALE

Conceptual Design

Concept A: Microcontroller + Custom PCB

This concept is based on a microcontroller and a custom PCB that would house the various switch jacks and user interface elements like LEDs and buttons.



4+1 Inputs, 5 Neopixels, Buzzer, SAMD21

- Qt Py SAMD21
- (2) Analog Joystick Jack
- (4) Switch Jacks A, B, C, D
- (1) Mode Button, Calibration button and Mode Switch Jack
- (1) Output Neopixels
 - o Mouse Output Neopixel LED
 - o Joystick Output LED
 - o Slot 1 Neopixel LED
 - o Slot 2 Neopixel LED
 - o Slot 3 Neopixel LED
- (1) Buzzer
- (2) External i2C connection

Forest Hub

DESIGN RATIONALE

User Interface – Neopixels

5mm Through hole NeoPixels could be utilized to provide custom spacing and arrangement of the indicator lights. These can be driven using a single pin, have adjustable brightness, and could be used in different colours based on user preference.

NeoPixel Diffused 5mm Through-Hole LED – 5 Pack

- [Adafruit](#), \$4.95 USD
- [Mouser](#), \$7.18 CAD

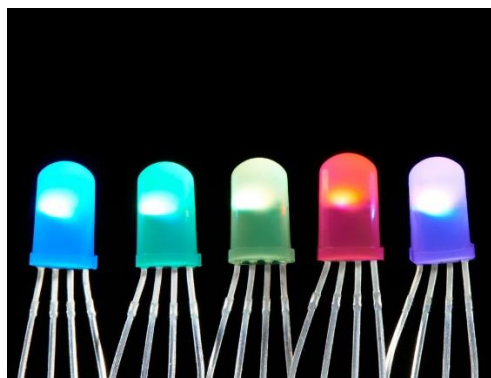


Figure 1. Adafruit NeoPixel Diffused 5mm Through-Hole LED - 5 Pack

Concept B: GPIO Expander or Seesaw-based PCB

This concept is based around a microcontroller that is connected via I2C to a GPIO Expander mounted on a custom PCB. The custom PCB is intended to be a more general purpose I2C device where multiple switch inputs, buttons, and indicator lights could be controlled by a variety of microcontrollers.

Concept C: Feather dev Board + Feather Wings

This concept would utilize the Feather ecosystem for the development board and a combination of Feather Wings and i2C dev boards for additional functions.

OLED Display FeatherWing: <https://www.adafruit.com/product/4650> 128x64 128x64 OLED Display + 3 integrated buttons.

Forest Hub

DESIGN RATIONALE

As a bonus, the Adafruit Joy FeatherWing, which has a built in thumbstick and buttons could be connected to provide an all-in-one solution.

Adafruit Joy FeatherWing for all Feathers

- [Adafruit](https://www.adafruit.com/product/3632), \$9.95 USD

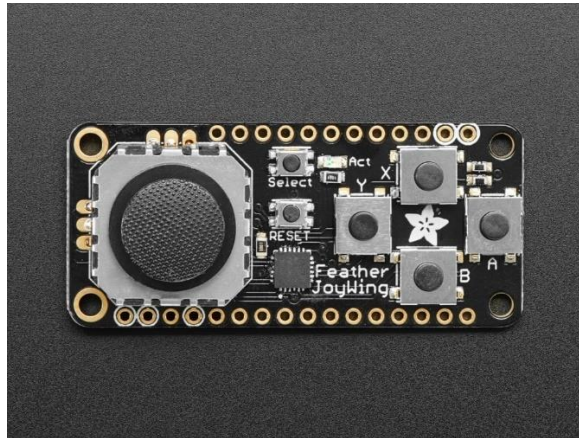


Figure 2. Adafruit Joy FeatherWing. <https://www.adafruit.com/product/3632>

Concept D: DIY / 3D Printed PCB

This concept would avoid the use and associated cost of a custom PCB.

Switch Jacks

Switch and joystick ports could be done using panel mount connectors and individually soldered wires, or using a combination of breadboard-friendly switch jacks soldered into proto board and TRRS breakout boards.

Adafruit TRRS Jack Breakout Board

- [Adafruit](https://www.adafruit.com/product/3632), \$1.75 USD
- [Mouser](https://www.mouser.com), \$2.54 CAD

Forest Hub

DESIGN RATIONALE

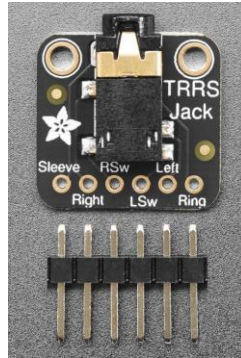


Figure 3. Adafruit TRRS Jack Breakout Board.

Adafruit Breadboard Friendly 3.5mm Stereo Headphone Jack

- [Adafruit](#), \$0.95 USD
- [Mouser](#), \$1.38 CAD

External I2C Connection

Sparkfun RJ11 / RJ25 Breakout

- [Mouser](#), \$2.83



Figure 4. Sparkfun RJ11 Breakout

Accessories

There are a couple options for I2C joysticks that could be used in place of the Analog Joystick:

Adafruit Mini I2C Gamepad with seesaw – STEMMA QT / QWiic



© 2023-2024 by [Neil Squire Society](#).

This work is licensed under the CC BY SA 4.0 License: <http://creativecommons.org/licenses/by-sa/4.0>

Files available at <https://github.com/makersmakingchange/Forest-Joystick-Mouse-Hub>

Forest Hub

DESIGN RATIONALE

- [Adafruit](#), \$7.50 USD
- [Mouser](#), \$10.88 CAD; No stock in Mouser (0 as of 2023-Sep-08)
- [DigiKey](#), \$10.79 CAD; Low stock in Digikey (4 as of 2023-Sep-08)

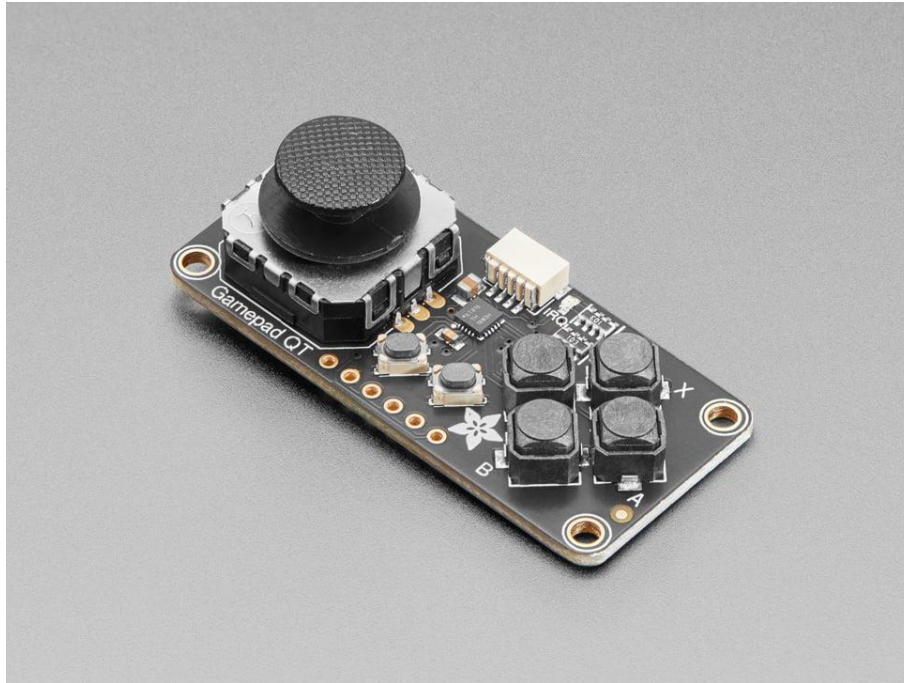


Figure 5. Adafruit Mini I2C Gamepad with seesaw - STEMMA QT / Qwiic

Concept Selection

Concept A was selected as a custom PCB made the most sense for simple, reliable maker manufacturing in the larger quantities expected as part of the ATP Grant. A custom-PCB free version should also be designed to help manage the cost and mitigate the barriers of ordering PCBs for single builds.

Detailed Design

Naming

Tree theme:

- Canopy
- Root
- Hub
- Trunk – where the branches connect
- Stem
- Branch

Forest Hub

DESIGN RATIONALE

- Branch Joystick Mouse Hub?
- The Branch – Joystick Mouse Hub
- The Branch Hub
- The Branch Joystick Hub
- The Branch Joystick Hub – for Gamepad and Mouse Application
- Dendro (relating to trees)
- Arbor (
 - Arboreal
 - Arbor Hub
 - ArborLink
 - Arborium
- Forest
 - Sylvan
 - Silva – latin for forest or wood
 - SilviHub
 - Silvadaptor
 - Boreal
 - Taiga
 - Acadian

Branch

- Bough

Earth / Soil / Dirt / Loam

- Terra
- Topsoil

Canopy

- Crown
- foliage

Word origins

- Dendro- greek dendron – tree
- Xylo- Greek “Xulon” – wood
- Silva- latin silva – “forest” or “woodlands”
- Lignum- latin lignum “wood”
- Arbor- latin arbor – “tree”
- Sylva – variant of silva – forest or wooded area
- Phyto – Greek phyton – “plant”
- Myco- Greek mukes – “fungus” / “mushroom”
- ~~- Ped- Greek pedon – “Soil” or “ground”~~

Forest Hub

DESIGN RATIONALE

Connector / Adapter / Interface

- Link
- Connect
- Nexus
- Port
- Node
- Plug
- Adaptor
- Hub
- Dongle
- ____ Joystick Interface
- ____ Joystick Mouse Interface
- ____ JoyMouse Interface

Mycorrhizal – My-co-rye-zal

	Mycohub	Root
Branch Joystick Mouse Hub		Dendrohub
	Branch Hub	Root Hub
	ArborHub	Denrodongle
		TerraHub
Arbor Interface		
Arbor Joystick Interface	(AJI?)	
____ JoyMouse Interface		
Canopy		
Forest		
Branch		
Boreal		
Canopy Connect Joystick Mouse Interface		
Canopy Connect Joystick Mouse Hub		

Short List

Arbor Joystick Mouse Hub	"Arbor Hub"
Boreal Joystick Mouse Hub	"Boreal Hub"
Branch Joystick Mouse Hub	"Branch Hub"
Canopy Joystick Mouse Hub	"Canopy Hub"
Forest Joystick Mouse Hub	"Forest Hub"

Final Selection: Forest Joystick Mouse Hub



© 2023-2024 by [Neil Squire Society](https://neilsquire.org/).

This work is licensed under the CC BY SA 4.0 License: <http://creativecommons.org/licenses/by-sa/4.0>

Files available at <https://github.com/makersmakingchange/Forest-Joystick-Mouse-Hub>

Forest Hub

DESIGN RATIONALE

Enclosure Design

UI Layout

The Enabled Controller Mini has connections on all sides of the enclosure, and we received feedback that this was undesirable and made it more difficult to manage cables. The decision was to have the reduce the number of cable surfaces to 1 or 2. Two main options were considered: inputs on one side and output on the other side, and everything on one side. The decision was to move all cable connections to the back of the enclosure to make it easier to manage cables, and minimize the visual clutter for the user.

Several options were also considered for the placement of the main user indicators and controls. Locating both the buttons and the indicators on the top surface was selected. This made it easier to activate the buttons with a single hand by pressing down on the enclosure (and avoiding pressing on the side which could cause the enclosure to slide). It also made it possible to put the indicators and controls in proximity to associate their function, and provided greater flexibility in LED positioning.

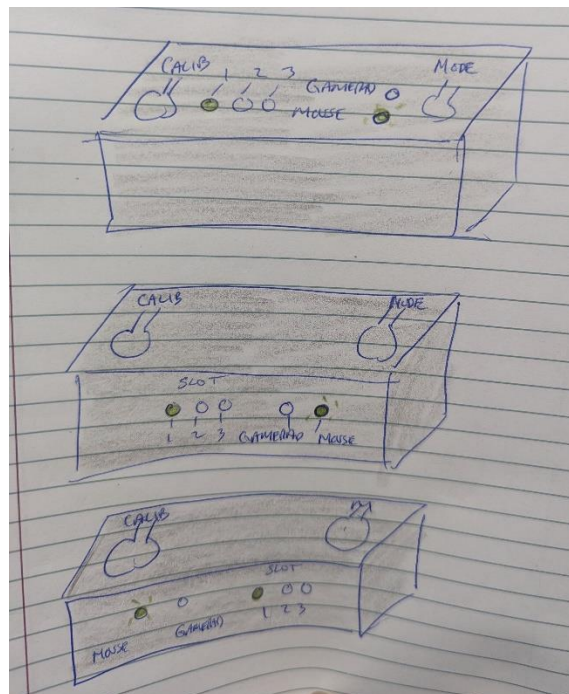


Figure 6. Indicator Layout Options

Forest Hub

DESIGN RATIONALE

Graphics / Labelling

Switch Inputs Jack Naming

S1, S2, S3, and S4 were used for the input switches to help reduce confusion between ABCD (e.g., A and B from a gaming controller) or 1234.

Mode Button

The mode button consists of three circles connected by arrows.

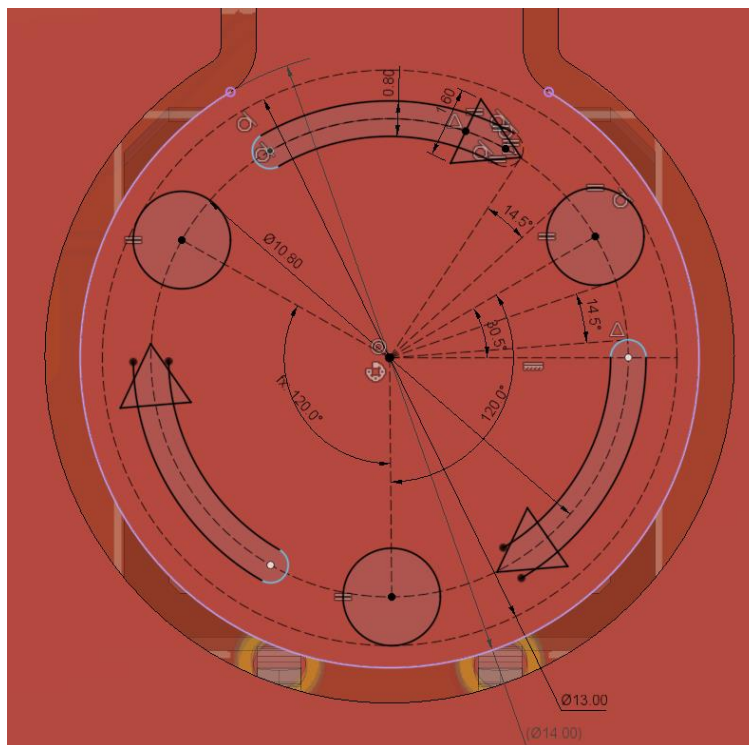


Figure 7. Mode Button Graphic.

Calibration Button

A bullseye design was chosen for the calibration button.

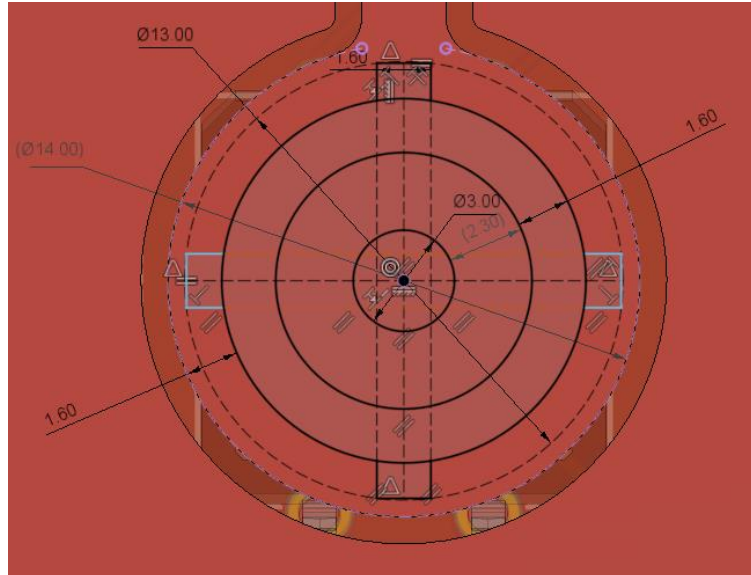


Figure 8. Calibration Button Graphic.

Mouse Output LED

A top-down graphic of a computer mouse was selected. A mouse cursor, mouse animal, and the word “mouse” were also considered. The top-down graphic was the most appropriate given the function and the space available.

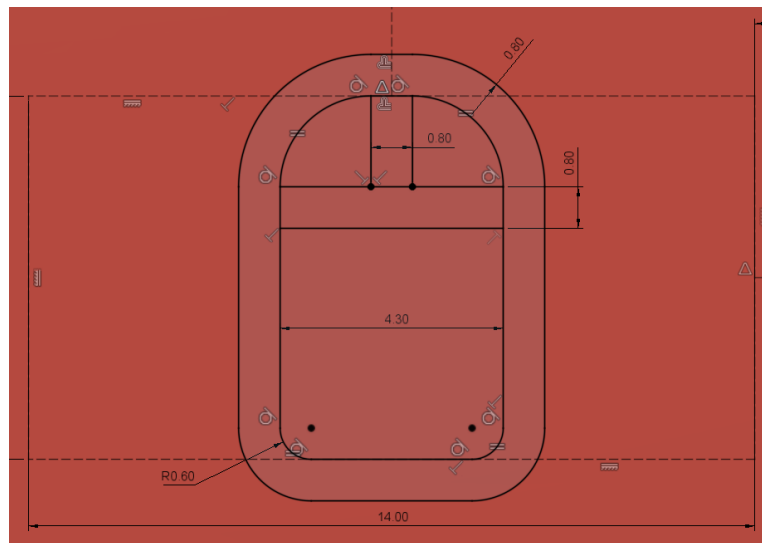


Figure 9. Mouse Output LED Graphic.

Joystick Output LED

A top-down graphic of a traditional gamepad with a directional pad and two buttons was selected.

Forest Hub

DESIGN RATIONALE

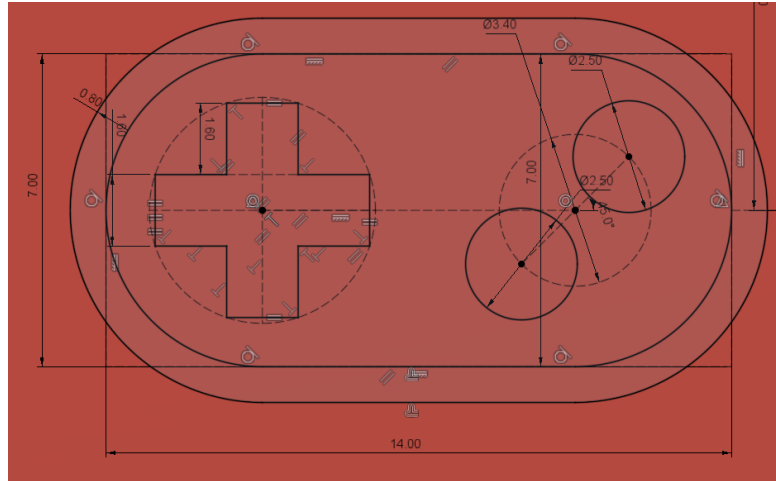


Figure 10. Joystick Output LED Graphic.

Electronics

Electronics Requirements

1. 5X mono jack inputs
2. 1X analog joystick input
3. 1X Buzzer
4. 2X Tactile Switches – Mode Switch and Calibration
5. 1X RJ11 External i2C Connection
6. 5X NeoPixels

Schematic

Buzzer

A buzzer was added to provide audible feedback for those that are unable to see the feedback lights on the enclosure. A [PS1240 Piezo Buzzer from Adafruit](#) was selected as it was used on a similar project to provide audible feedback (the ASTA Switch Training Aid).

To minimize components, a simple circuit with a resistor was selected. A 100 ohm resistor was chosen.

BUZZER

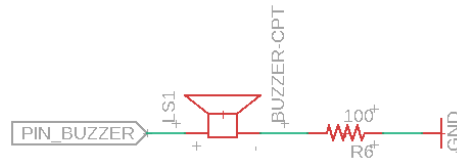


Figure 11. Simple Buzzer Circuit

To maximize sound output, a slightly more complicated circuit with another resistor and a drive transistor would be required. Initial testing showed that the volume and response is likely sufficient.

Multiple Button Input Resistor Ladder

There are more pins required than available GPIO on the microcontroller. To maintain function and the microcontroller, a method for reading multiple input via a single GPIO line was required.

A few options were considered, and ultimately the following circuit was selected so that each input and combination of multiple inputs could be detected by the analog input line. The Mode Switch Jack (SW_M) was chosen as the final leg on the ladder so that any cable or switch resistance would have a smaller effect.

Forest Hub

DESIGN RATIONALE

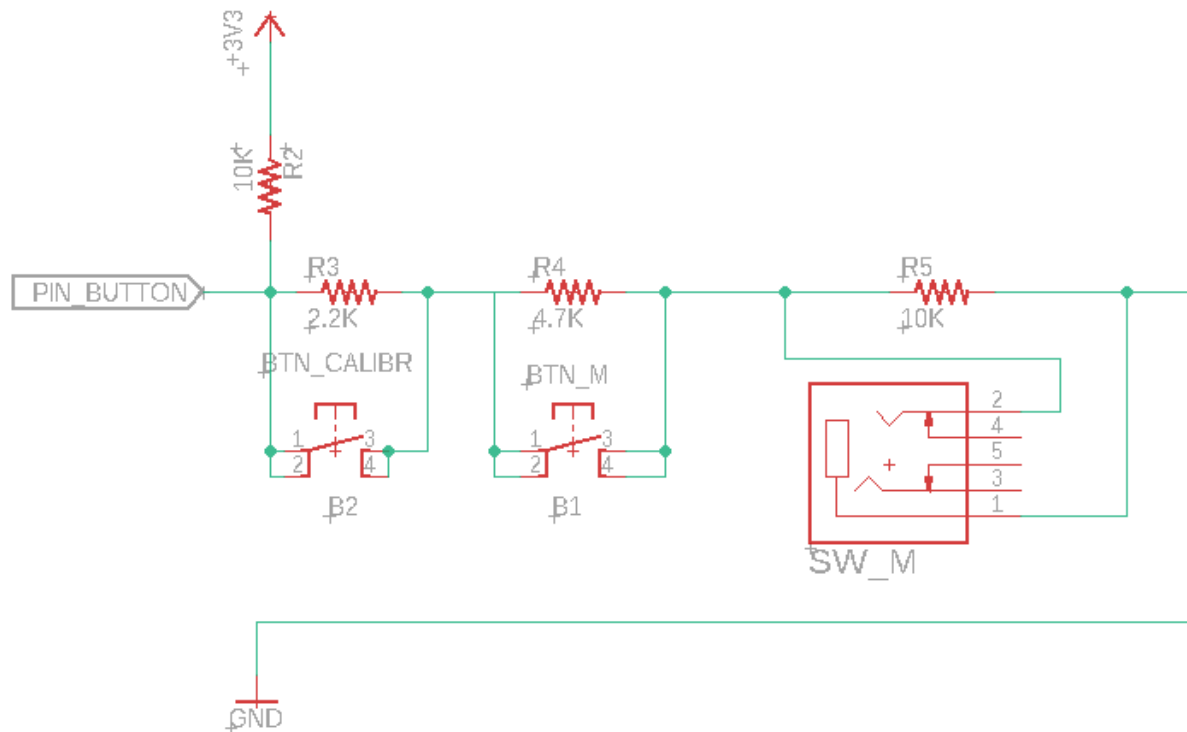


Figure 12. Analog Multiple Input Schematic

Table 1. Resistor Ladder Values

Resistor	Item	Value
R2	Voltage Divider	10K
R3	B2 / Calibration Button	2.2K
R4	B1 / Mode Button	4.7K
R5	SW_M / Mode Switch Jack	10K

Table 2. Theoretical Analog Voltages

B2 State	B1 State	SW_M State	Resistance Top	Resistance Bottom	Voltage	Steps
1	1	1	10000	16900	2.1	643
1	1	0	10000	14700	2.0	609
1	0	1	10000	12200	1.8	563
1	0	0	10000	10000	1.7	512
0	1	1	10000	6900	1.3	418
0	1	0	10000	4700	1.1	327
0	0	1	10000	2200	0.6	185
0	0	0	10000	0	0.0	0

Forest Hub

DESIGN RATIONALE

Pin allocation

The following table lists the inputs and outputs

Table 3: Pin Allocation

Item	Voltage	Gnd	Digital	Analog	Assignment
Joystick Jack	1	1		2	A0, A1
Switch S1 Jack			1		A2
Switch S2 Jack			1		A3
Switch S3 Jack			1		TX
Switch S4 Jack			1		MO
Switch M Jack			1		MI (Resistor Network)
Mode Button				1	MI (Resistor Network)
Calibration Button				(shared with Mode)	MI (Resistor Network)
Buzzer			1 - PWM		SCK
Neopixels	1	1	1		

Forest Hub

DESIGN RATIONALE

I2C Connection: RJ11Connector

A phone cable and a RJ11 Connector approach was selected for the external I2C connection as it has been used in another Open Source assistive technology device, the FAB1. There is a system that uses an RJ45 and ethernet cable to conduct I2C over distances (e.g., SparkFun QwiicBus

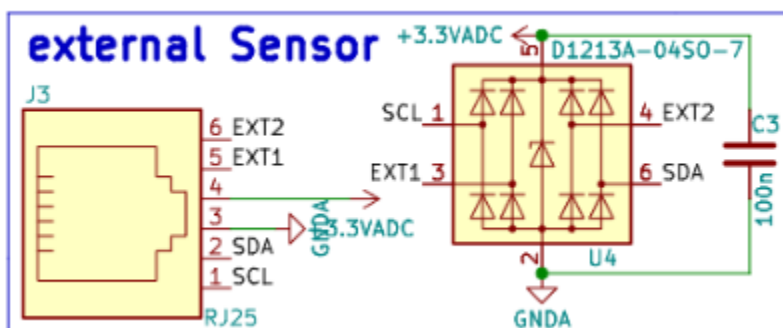


Figure 13. FAB1 External Sensor Wiring Scheme.

https://github.com/asterics/FAB1/blob/SRC_v3.0/hardware/FAB1_schematic.pdf

Table 4. I2C - RJ11 Wiring Options

Pin	Option A - FAB1 Connection	Option B – Match Sparkfun Adapter	Option C – Match i2C Module Breakout Board
1	SCL	GND	GND
2	SDA	GND	3.3V
3	GND	3.3V	GND
4	3.3V	SDA	SCL
5	Not connected	SCL	SDA
6	Not connected	GND	GND

Three options were considered for how to wire connect the I2C connections to the RJ11 connector.

- Option A would be to match FAB1 External Sensor Wiring Scheme. This would have been the preferred option, but it is common for cables and connectors to only have the inner 4 pins connected (i.e., 2,3,4,5).
- Option B would make it possible to solder an RJ11 breakout directly to the Sparkfun STEMMA QT / Qwiic Adapter
- Option C would make it possible to solder an RJ11 breakout directly to a STEMMA QT breakout board.

Version 1.0 of the PCB was released with Option C. It was later discovered that this doesn't work as intended, so later versions of the PCB (V2.0 and up) use Option B. This was also chosen to be consistent with the [LipSync](#) design.

Neopixel Soldering Jig

The five Neopixel LEDs are spaced higher than the PCB so they partially protrude through the top enclosure. To reliably set this height, a 3D printed jig is used. The bottom of the LEDs are at 3 different heights from the board: 11.5, 12.0, and 12.5 mm from the board.

The initial jig design was a single 3D printed piece. The spacing between the 4 leads was too small to have separate holes for each, so a single cavity was used for each.

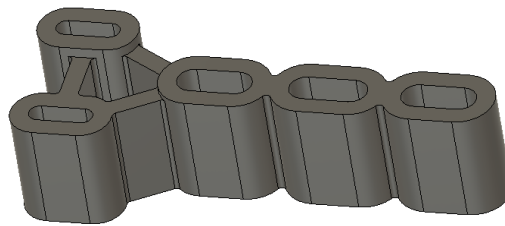


Figure 14. LED Soldering Jig (V1.0)

Assembly with this jig was extremely difficult, as the leads for each of the 5 jigs (i.e., 20 total) had to be aligned into the corresponding holes in the PCB without being able to see them. Several alternatives were considered, including splitting the jig in to two parts, or five individual parts. The option that was chosen for the next version features a similar design to the original with one-piece that has windows to make it easier to see and insert the individual leads.



Figure 15. LED Soldering Jig (V2.0)

Forest Hub

DESIGN RATIONALE

Firmware

Parsing Multiple Switches on Analog Pin

The theoretical values for each of the combination of switch closures is presented in Table 5. This is calculated based on the nominal resistor values. To resolve which combination is pressed, thresholds are setup at midway between the different steps. This should accommodate for any variation in true resistor values due to their tolerance. When the measured voltage is below the threshold, that combination of buttons/switches is likely pressed.

Table 5. Theoretical Voltage for Switch Combinations

SW_M	B1	B2	Theoretical Steps	Threshold
Off	Off	Off	643	
Off	Off	On	609	626
Off	On	Off	562	586
Off	On	On	512	538
On	Off	Off	418	465
On	Off	On	327	373
On	On	Off	184	256
On	On	On	0	93

Non-Volatile Storage

The Forest Hub needs non-volatile storage to store settings when the device is powered off.

The Adafruit QT Py does not have any EEPROM, but is able to use flash. The FlashStorage library (<https://github.com/cmagle/FlashStorage>) is used to handle it more easily.

Variable	Variable Type	Note
isConfiguredFlash	bool	False when first flashed, True after first run
deviceNumberFlash	int	
modelNameFlash	int	
versionNumberFlash	int	Number to track firmware version
deadzoneLevelFlash	int	
cursorSpeedLevelFlash	int	
operatingModeFlash	int	Operating Mode (0

Forest Hub

DESIGN RATIONALE

ledBrightnessFlash	int	0-255 LED Brightness
slotNumberFlash	int	Track index of current settings slot
xOutputMinimumFlash	int	Calibration
xOutputMaximumFlash	int	Calibration
yOutputMinimumFlash	int	Calibration
yOutputMaximumFlash	int	Calibration
xNeutralFlash	int	Calibration
yNeutralFlash	int	Calibration
ledColorRFlash	int	User LED color - red
ledColorGFlash	int	User LED color - green
ledColorBFlash	int	User LED color - blue

Calibration

Calibration is used to ensure that the joystick outputs zeros when the input joystick is released to mechanical neutral, and that the output is maxed then the input joystick is moved to the maximum extent.

Center Reset

The center reset measures the neutral position of the joystick: xNeutral and yNeutral. These numbers are subtracted from the main reading so that when the joystick is in the neutral position, the output is zeroed (i.e., x=0, y=0). The calibration is saved to and read from flash memory, (e.g., xNeutralFlash, yNeutralFlash), though this doesn't do much as the calibration is triggered every time the device is powered.

Extents

The extents calibration ensures that the output is maximized when the joystick is moved to its most extreme position. Currently, these values are just set to a default constant value (e.g., +/- 127) – there is no method or way to dynamically measure the physical min and max of each joystick axis.

These values are set and retrieved from memory: xOutputMinimumFlash, xOutputMaximumFlash, yOutputMinimumFlash, yOutputMaximumFlash.

Forest Hub

DESIGN RATIONALE

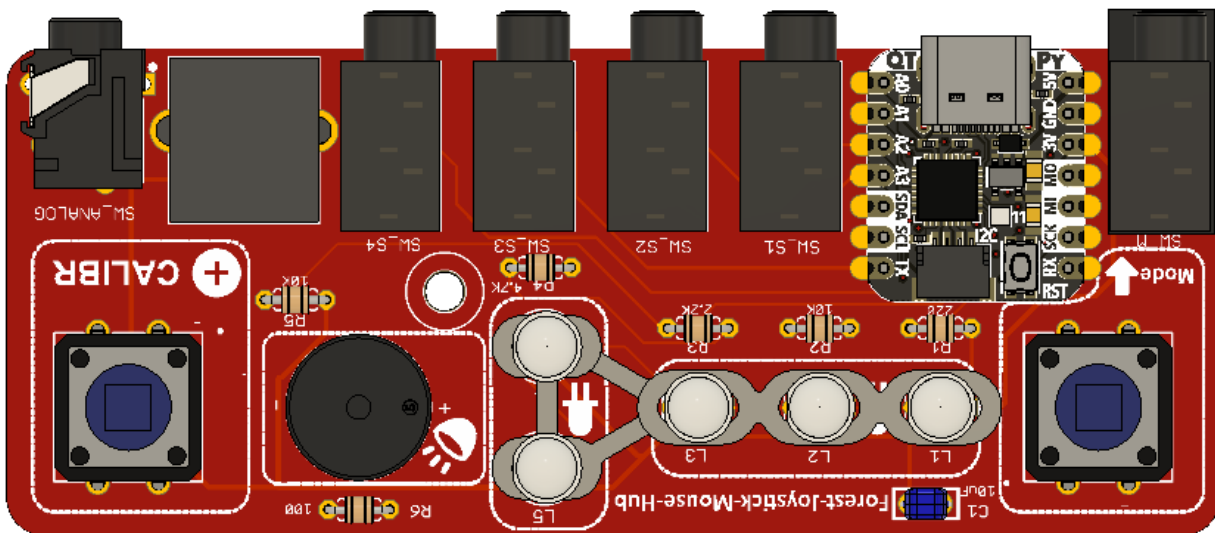
Version 0.1 – PCB Prototype

V0.1 Features

The initial prototype of the Forest Hub included the following features:

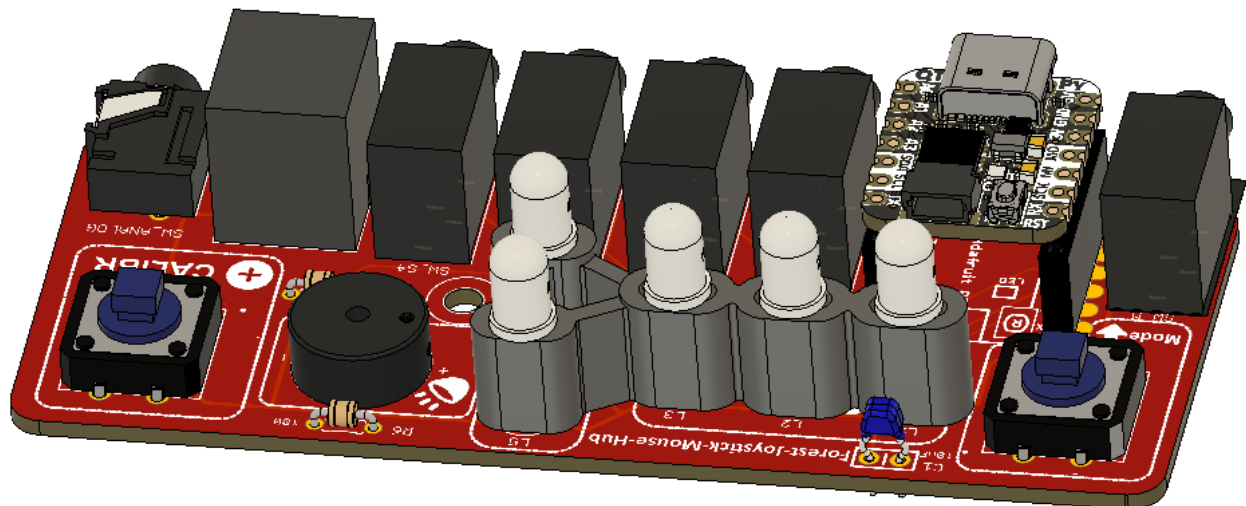
1. (1) Adafruit QT Py SAMD21 Development Board
2. (1) Analog Joystick Input via 3.5 audio TRRS jack
3. (4) Assistive Switch Inputs via 3.5 mm audio mono jacks
4. (1) Assistive Switch Input via 3.5 mm audio mono jack for mode/slot switching
5. (1) Built-in button for mode/slot switching
6. (1) Built-in button for calibration
7. (5) Multicolor LED lights for indicating current slot and output mode
8. (1) Accessory Connection via i2C connected to a RJ25 jack
9. (1) Piezo Buzzer for audio feedback

V0.1 PCB



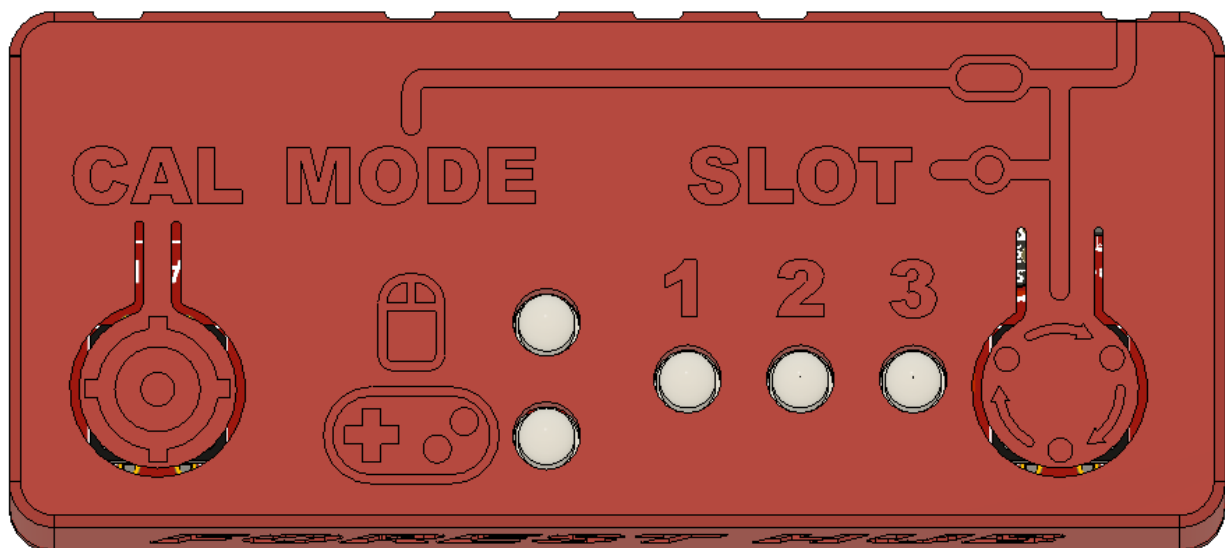
Forest Hub

DESIGN RATIONALE



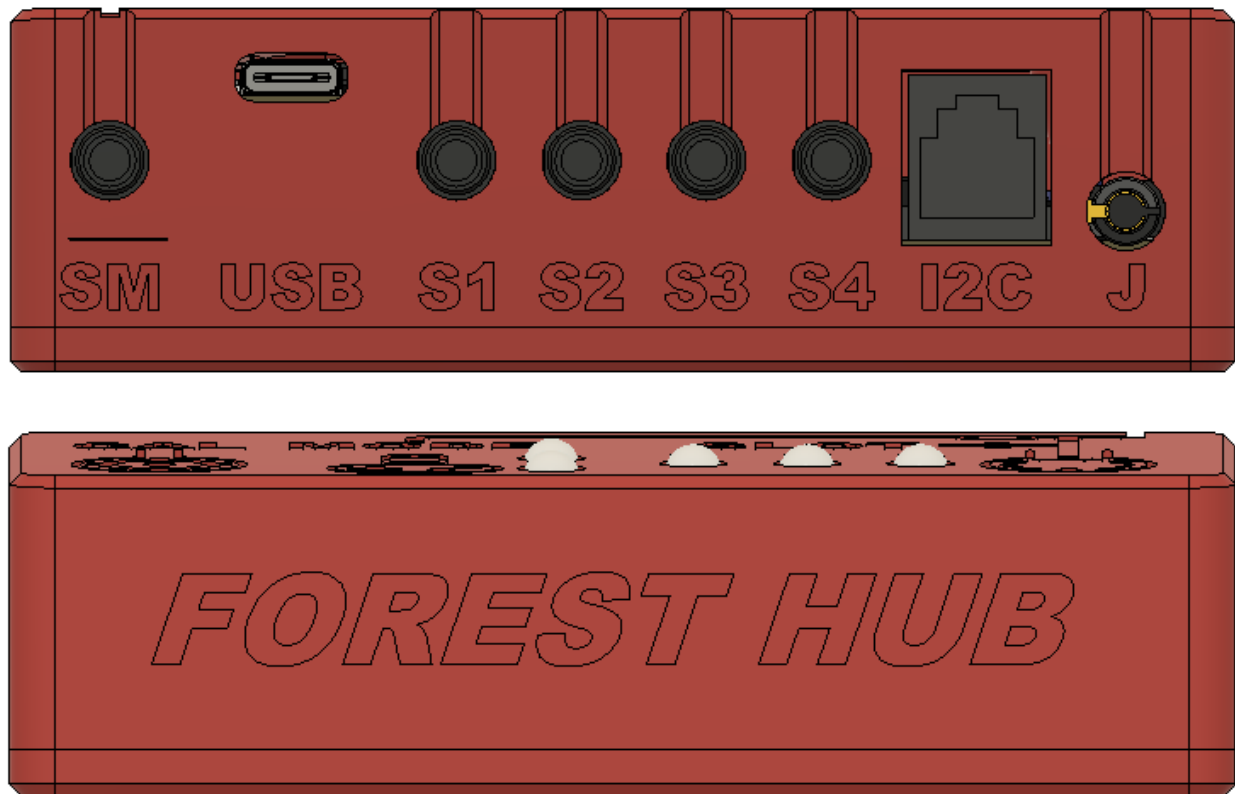
V0.1 Enclosure

The enclosure comprises a top and a bottom that are secured with a snap fit. A third 3D printed component acts as a jig to bring the LEDs up to the proper height to stick through the enclosure.



Forest Hub

DESIGN RATIONALE



V0.1 Build Feedback

1. Some portions of the top enclosure didn't print well – will require a near-perfect first layer to resolve some of the fine details.
2. The capacitor was added late and was not part of the prototype components order.
3. The resistors didn't match the footprint – they were too long to fit properly and had to be soldered in awkwardly.
4. The RJ25 connector is positioned recessed a bit from the edge of the PCB and doesn't match the other audio connectors. Because it is recessed into the enclosure, it is a little difficult to remove the plug when it is inserted.
5. The RJ25 connector has a different footprint than the Sparkfun RJ11 breakout board and is not compatible.
6. It was quite difficult to insert the Neopixel LEDs through the one-piece led soldering jig and into the holes in the PCB.
7. Soldering the Neopixels requires fine dexterity and skill to avoid bridging the connections.
8. The #4 screw didn't thread properly into the base. (Note: This was a modelling error; the sketch was fixed, but the feature kept the original diameter).

Forest Hub

DESIGN RATIONALE

V0.1 Testing

1. The enclosure snap-fits do not securely retain the bottom to the top. If the enclosure is dropped (without the PCB), the parts fly apart.
2. The enclosure buttons need to be pushed further than expected to activate the internal buttons. (Note: There is a mismatch between the height of the ECAD model of tactile button and the actual component, resulting in a greater distance between enclosure button pusher and tactile button.)
3. All components are wired correctly and function – Neopixels, buttons, switch inputs, joystick input, buzzer, and I2C.

V0.1 Opportunities for Improvement

1. PCB Changes
 - a. ~~Add Open Source Hardware certification and logo to PCB~~
 - b. Adjust positioning of RJ25 jack so that edge lines up with audio jacks
 - c. Replace RJ25 component with one that has a compatible footprint with Sparkfun breakout board
 - d. Ensure resistor footprint matches BOM model
 - e. Add silkscreen label to RJ25 connector / I2C on board
2. Consider easier-to-solder LED options rather than Neopixel Leds
3. Add more robust enclosure snap-fit
4. ~~Fix enclosure PCB screw post so that it works with #4 fastener~~
5. Add additional mounting hole or mounting features to support PCB
6. Investigate options for preventing reset when joystick cable is inserted / removed
 - a. Switched jack?
7. Implements extents calibration

Version 1.0

V1.0 Features

The initial release of the Forest Hub includes the following features:

1. (1) Adafruit QT Py SAMD21 Development Board
2. (1) Analog Joystick Input via 3.5 audio TRRS jack
3. (4) Assistive Switch Inputs via 3.5 mm audio mono jacks
4. (1) Assistive Switch Input via 3.5 mm audio mono jack for mode/slot switching

Forest Hub

DESIGN RATIONALE

5. (1) Built-in button for mode/slot switching
6. (1) Built-in button for calibration
7. (5) Multicolor LED lights for indicating current slot and output mode
8. (1) Accessory Connection via i2C connected to a RJ25 jack
9. (1) Piezo Buzzer for audio feedback

V1.0 Opportunities for Improvement / Future Work

1. Consider easier-to-solder LED options rather than Neopixel Leds
2. Add additional mounting hole or mounting features to support PCB
3. Consider additional circuitry (i.e., driving transistor) for piezo buzzer to increase volume.
4. Investigate options for preventing reset when joystick cable is inserted / removed
 - a. Switched jack?
5. Implements extents calibration
6. Develop non-custom PCB option
7. Provide access to microcontroller reset to make it easier to flash firmware

Version 2.0 – Refined PCB

V2.0 Changes

PCB Changes

1. Replaced RJ11 connector with same component that matches SparkFun RJ11 Breakout board.
2. Increased size of resistor footprint to match part
3. Increased size of capacitor footprint
4. Aligned RJ11 Jack with switch jacks
5. Flipped labels to orient board in natural usage orientation
6. Added additional silkscreen labels for component values
- 7.

Firmware

1. Properly implemented neutral calibration.
2. Added basic sound feedback
3. Added slot level changes
- 4.

Enclosure / CAD

1. Increased clearance around USB C connection

Forest Hub

DESIGN RATIONALE

Version 2.0 OFIs

Maker Guide

1. Test fit them into first holes first to get the leads aligned

Forest Hub

DESIGN RATIONALE

Appendix

Resources

Adafruit QT Py

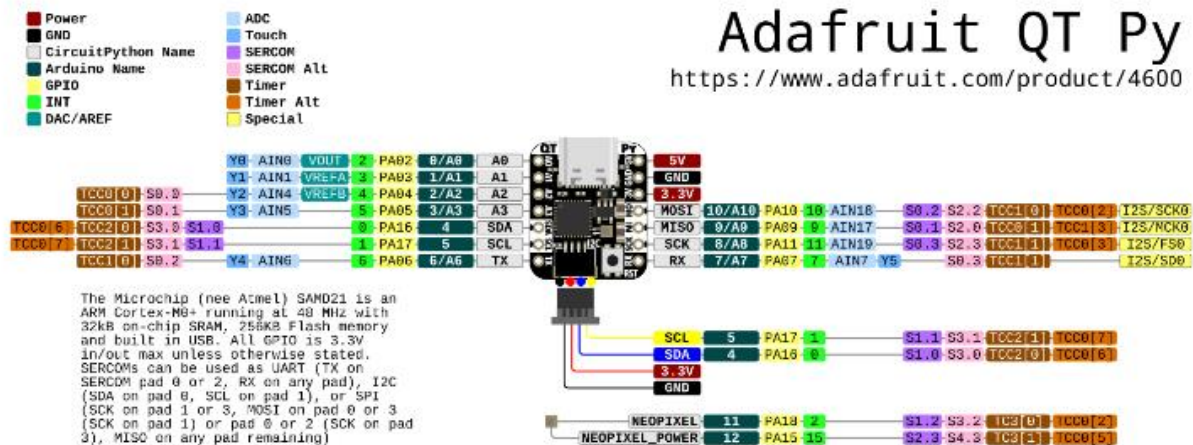


Figure 16. Adafruit QT Py Pinout. CC-BY-SA. <https://learn.adafruit.com/assets/110643>