Programmable Trackpad

Group 18
Taylor Barnes - CpE
Jonah Halili - CpE
Brian Modica - CpE
Bradley Vanderzalm - CpE

Project Motivation

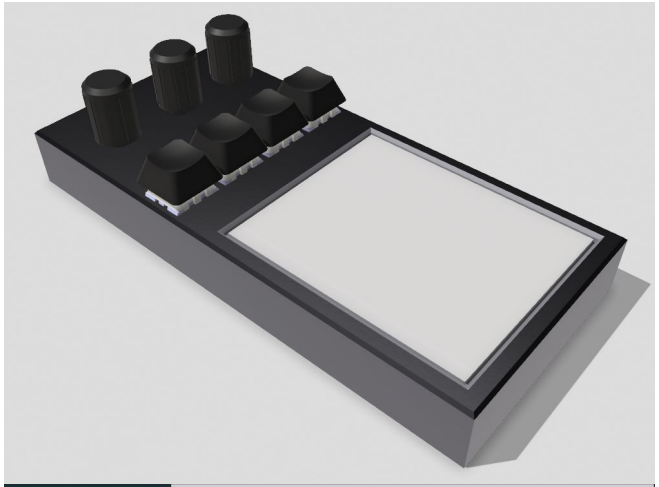
- The main goal of the programmable trackpad is to increase a user's productivity while using a computer
- The device will allow for usage of convenient short cuts
- Similar devices exists for the computer mouse, but nothing for the trackpad
- The intention is for this device to completely replace the default laptop trackpad

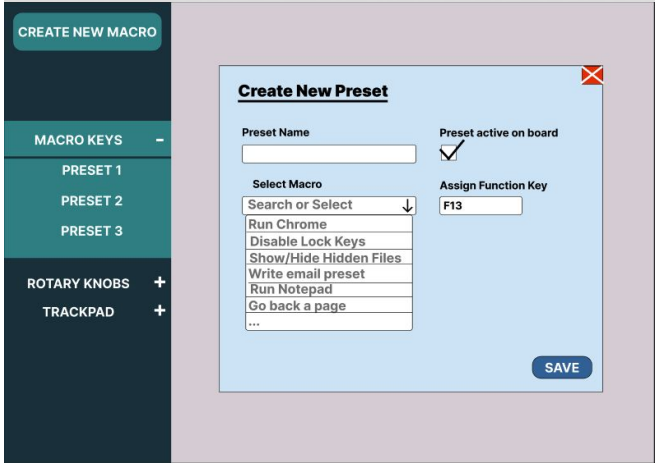
+

= ?

Project Concept

- Hardware device that acts as a fully-functioning trackpad.
- Buttons and rotary encoders on the device are assigned to macros that the computer executes.
- The user customizes macros via a graphical user interface.





Goals and Objectives

Goal	**Objective** (how we plan to achieve said goal)
Reduce common and repetitive tasks	Add buttons with macro key capabilities that are programmable.
Convenient	Make the device wireless with Bluetooth capability.
Ergonomic	Support ambidextrous users.
Low Learning Curve	Application with user-friendly interface to program macro keys.
Customizable for user	**Hardware** - Ability to easily remove keys to the user's liking. **Software** - Application should store custom macros for multiple users of the device.

Functions

Function	Description
4 Mechanical Keys	Capable of macro and keybind function.
3 Rotary Encoders	Capable of audio mixer, adjusting windows, etc. functions, (per-application functionality).
USB Connection	For charging the battery or having a wired connection.
Bluetooth Connection	Main connection for using the device.
Touchpad/Trackpad	Mouse replacement offering ergonomics.
4 Mouse Buttons	Availability changes based on dominant hand usage.
Power Switch	Turn the device on or off.
Application User Interface	Main ability to program and customize hardware keys with macros

Requirements

- The device shall weigh less than or equal to 1 lb, and not exceed the dimensions 5” x 5” x 2”
  - This increases portability
- The device shall have a latency of less than or equal to 48 ms
  - The trackpad on the device should offer a snappy and precise experience
- The device shall have 4 mechanical switches and 3 rotary encoders
  - User defined shortcuts are a must
- The key caps for the mechanical switches shall be hot swappable
- The device shall connect to a users PC via USB cable and Bluetooth
- The battery of the device shall have a use time of greater than or equal to 10 hours, and should gain a full charge in less than three hours

Market Analysis

Apple Magic Trackpad

- Price: $129.99
- Similarities to our device:
  - Small form factor
  - Fits conveniently on a desk
- Differences from our device:
  - No physical buttons
  - Little to no customizability
  - No Windows support

Mousetrapper Advance 2.0

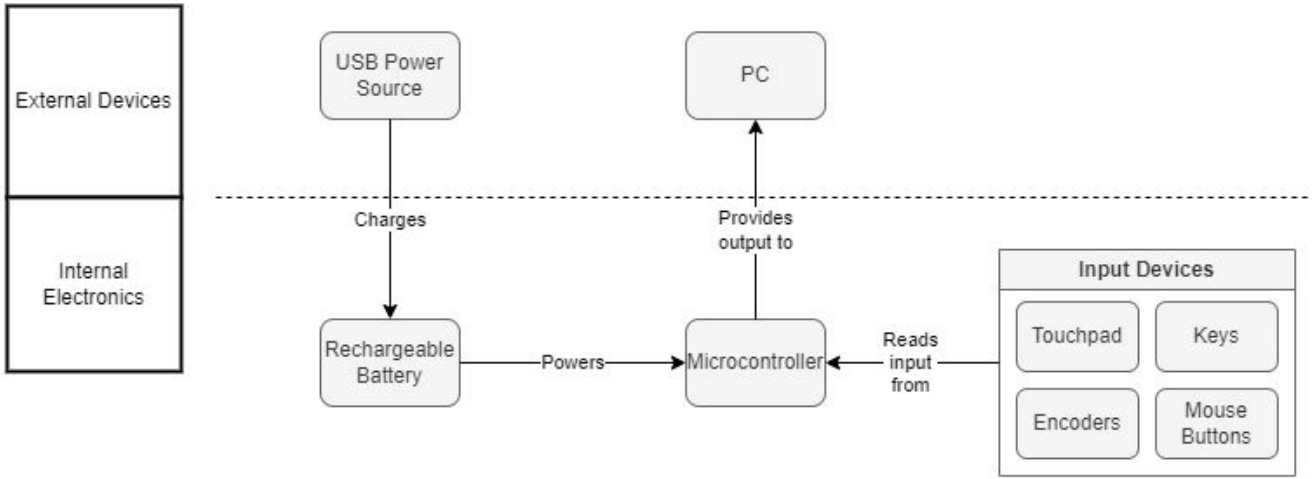
- Price: $200 - $300
- Similarities to our device:
  - Physical and customizable buttons
- Differences from our device:
  - Much larger
  - Wired functionality only

Keymecher MANO-703

- Price: $39.99
- Similarities to our device:
  - Has macro keys
  - Fits conveniently on a desk
- Differences from our device:
  - Macro key functions are hard coded, not customizable

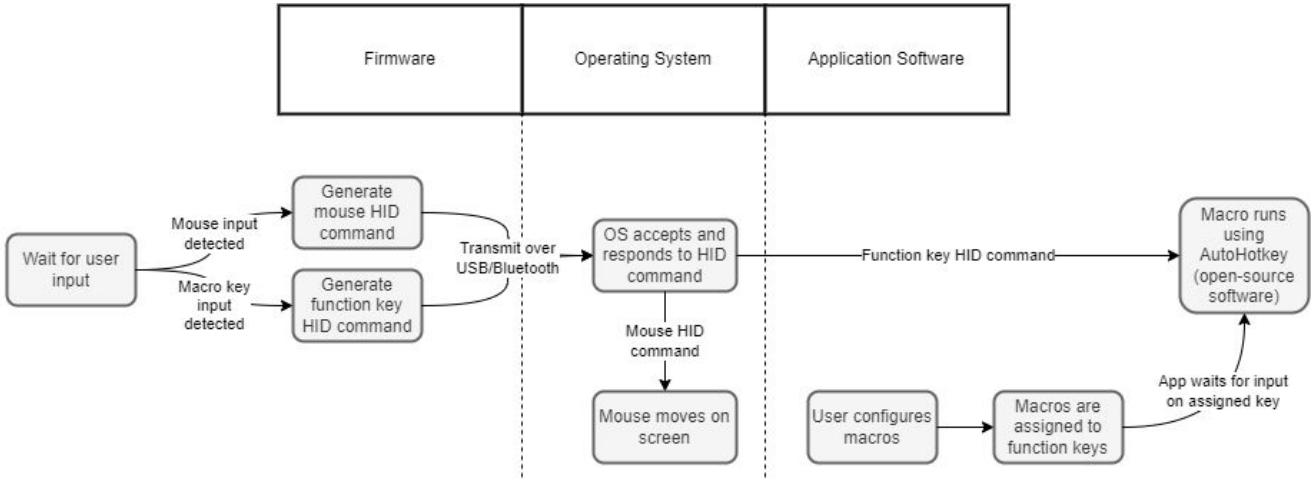
Hardware Overview

- The hardware device consists of a power system and an I/O system.
- A microcontroller handles data and transmits it to the user's PC.



Software Overview

- The device’s software operates on three levels: the device firmware, the PC’s operating system, and the application.
- The firmware processes user input and transmits to PC.
- The PC’s operating system interprets HID commands (mouse and keyboard).
- The application executes user-defined macros.



Technology Investigation: Hardware

Battery technology comparison

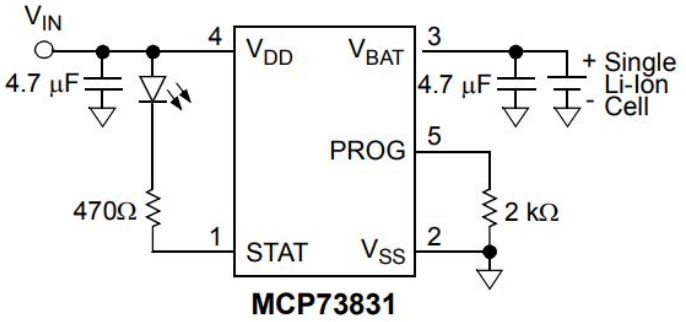
- LiPo batteries are the common choice for small devices like cell phones.
- Many microcontrollers and voltage regulators are designed to work with typical LiPo batteries.

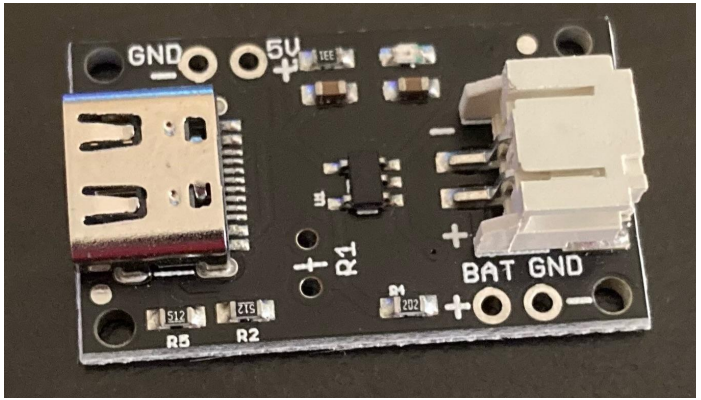
Technology	**Size**	**Capacity**	**Efficiency**	**Cost**	**Protection**
Lead-acid	Largest	Medium	Medium	Lowest	None
Lithium Ion	Medium	Medium	Best	Highest	None
Lithium Polymer	**Smallest**	**Medium**	**Best**	**Highest**	**Built-in**
Nickel Metal Hydride	Medium	Best	Medium	Medium	None

Technology Investigation: Hardware

Battery charger

- To charge the battery, the device needs a chip that will deliver constant current at the battery's rated voltage.
- The TP4056 and MCP7383X-2 both accomplish this task, so they were both considered.
- The MCP7383X-2 was chosen because it is simpler and has better documentation.





Technology Investigation: Hardware

Voltage regulation

- The device electronics use three different voltages.
  - 5V for USB
  - 3.3V for microcontroller
  - 3.7-4.2V battery
- Voltage converters are needed to regulate the 3.3V and 5V nodes.
- The tables show comparisons between various considered parts.

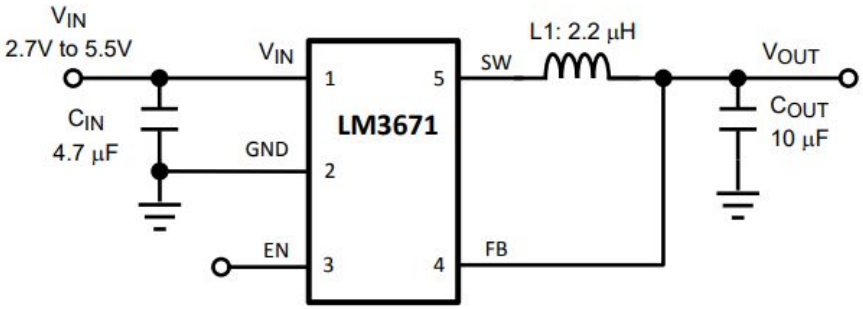
3.3V Chip	Input Voltage	Current Output	Number of extra parts (inductors and capacitors) needed per documentation
TPS62203	2.5-6 volts	300 mA	3
AP63203	3.8-32 volts	2000 mA	4
LM3671	2.7-5.5 volts	600 mA	3

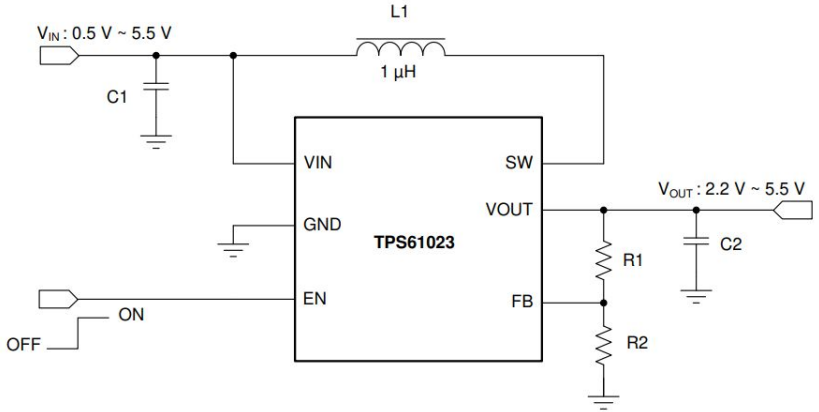
5V Chip	Input Voltage	Number of external components required
TPS61023	0.5-5.5 volts	3
ME2108	0.9-6.5 volts	4

Technology Investigation: Hardware

Voltage regulation

- For the 3.3V regulator, we chose the LM3671 because it is designed specifically for LiPo batteries.
- For the 5V regulator, we chose the TPS61023 because it has more thorough documentation.





Technology Investigation: Hardware

Input Units: Keyboard Switches

- Linear, tactile, clicky, silent
- Actuation force
- Sound level
- Price ($8 - $15 for a pack of 10)

Technology Investigation: Hardware

Switch	**Actuation Force**	**Behavior**	**Sound Level**
Gateron Red	45 grams	Linear	Quiet
Gateron Brown	**55 grams**	**Tactile**	**Slightly Audible**
Gateron Blue	60 grams	Clicky	Very Audible
Boba U4T	62 or 68 grams	Tactile	Slightly Audible
Gateron Black Ink v2	60 grams	Linear	Slightly Audible
Kailh Box Jade	65 grams	Clicky	Very Audible

Technology Investigation: Hardware

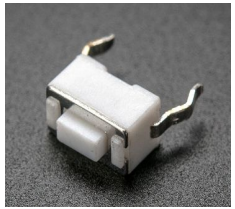
Input Units

Rotary Encoders

- Not many to choose from, generic ones did just the job
- We wanted haptic feedback between detents while turning the knob
- Potentially have another vertical input for more customizability
- Why not use a potentiometer
    - Did not achieve what we potentially wanted in terms of customizability

Trackpad/Touchpad Left and Right Click Switches

- Any tactile button that had enough height to reach the left and right button plates



Technology Investigation: Hardware

Microcontroller Requirements

- 17 GPIO pins
  - 6 encoders
  - 4 keys
  - 4 touchpad
  - 2 mouse buttons
  - 1 orientation switch
- Programmable with SWD
- Bluetooth-capable
- USB-capable

Technology Investigation: Hardware

Microcontroller

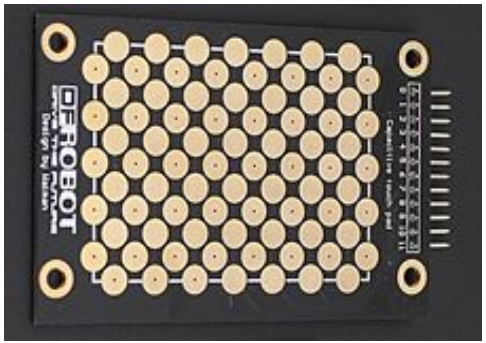
- ATMega32: Meets GPIO requirements, but does not have built-in Bluetooth/USB interfaces.
- nRF52840: Has necessary GPIO, is capable of generating Bluetooth and USB messages, does not have a Bluetooth antenna.
- ESP32: Similar specs to nRF, slightly less powerful.
- MDBT50Q: Combines the nRF MCU with a Bluetooth antenna.

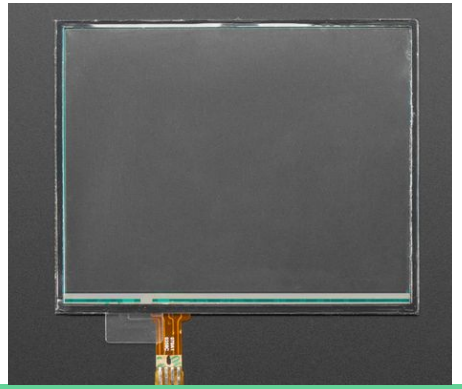
MCU	GPIO Pins	Bluetooth	USB	Programming Method
ATMega32	32	None	None	Arduino IDE, Microchip Studio
nRF52840	48	On-chip	On-chip	nRF5 SDK, Arduino IDE, CircuitPython
ESP32	34	On-chip	On-chip	Arduino IDE

Technology Investigation: Hardware

Touchpads

Touchpad	Size	Price	Integration	LCD Display	Power Drain
AliExpress RGB Touch Display Module	4.5” Diagonal	$15	9 Through-Hole Pins	Yes	High
DFRobot Capacitive Touch Kit	3.2” Diagonal	$20	12 Through-Hole Pins	No	Medium-Low
Adafruit Resistive Touch Screen	3.7” Diagonal	$6	4 Pin Flat Flex Cable	No	Low





Technology Investigation: Software

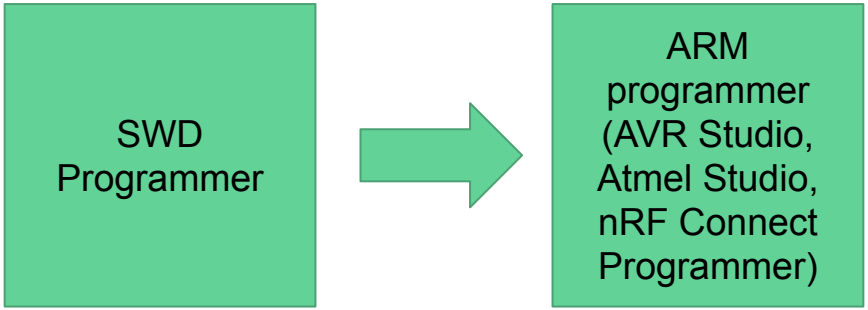
HID Libraries and Firmware

- CircuitPython to connect the I/O pins to respective input units
- USB and Bluetooth connection
- Keyboard, Rotary Encoder, X and Y coordinate reading
- Arduino also has libraries but CircuitPython was more involved with what we wanted
- KMK and QMK for keyboard firmware was also considered

Technology Investigation: Software

SWD Programmer and Connector

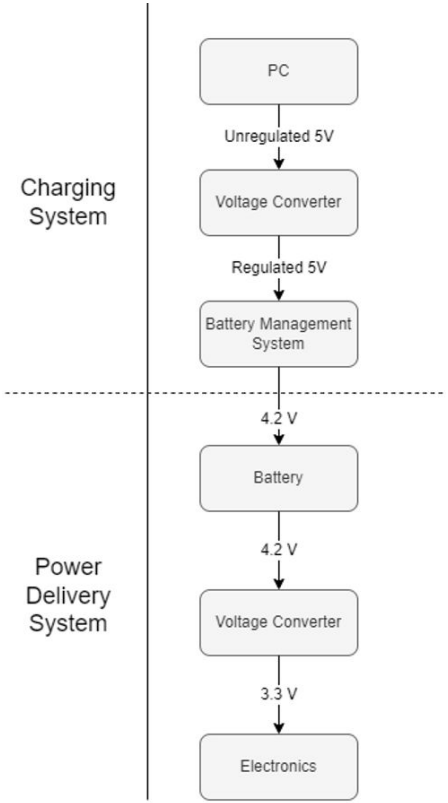
- This allows us to program the blank microcontroller and bluetooth module
- Install the necessary CircuitPython capabilities



Design Details: Hardware

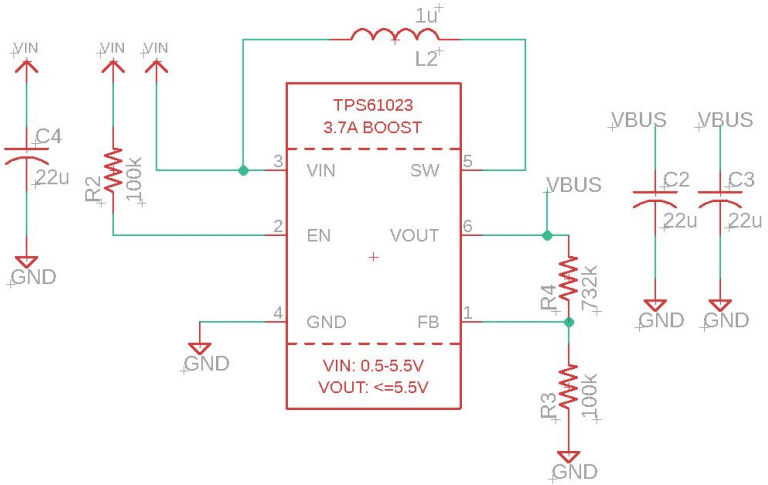
Power System

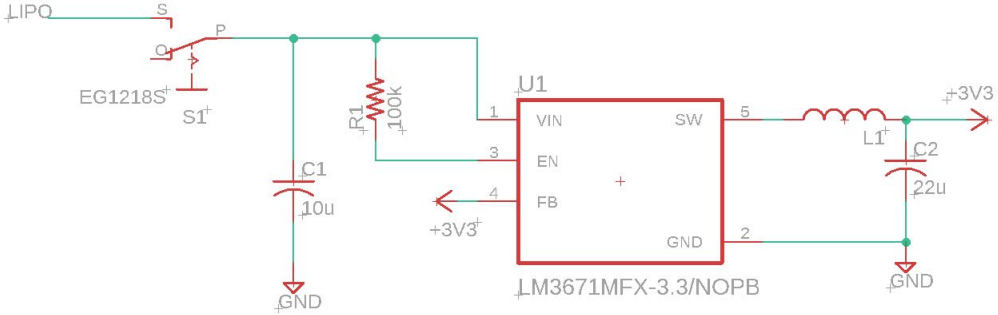
- The PC supplies 5V power, which is then regulated by the TPS61023.
- The regulated 5V power activates the MCP7383X-2, which charges the battery.
- The battery supplies 3.7-4.2V power, which is then regulated by the LM3671.
- The regulated 3.3V power activates the device electronics.

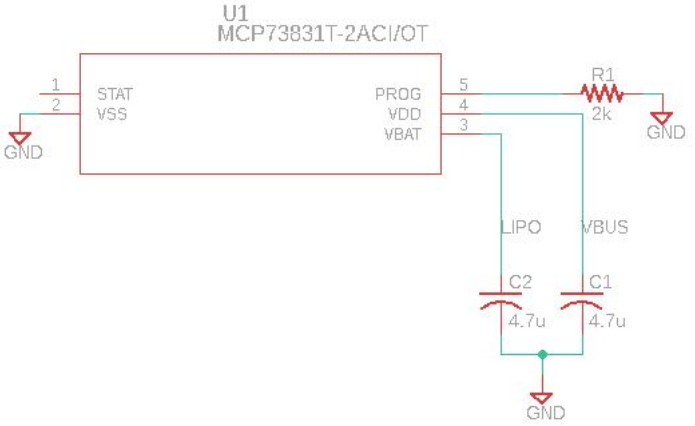


Design Details: Hardware

Power System Schematics

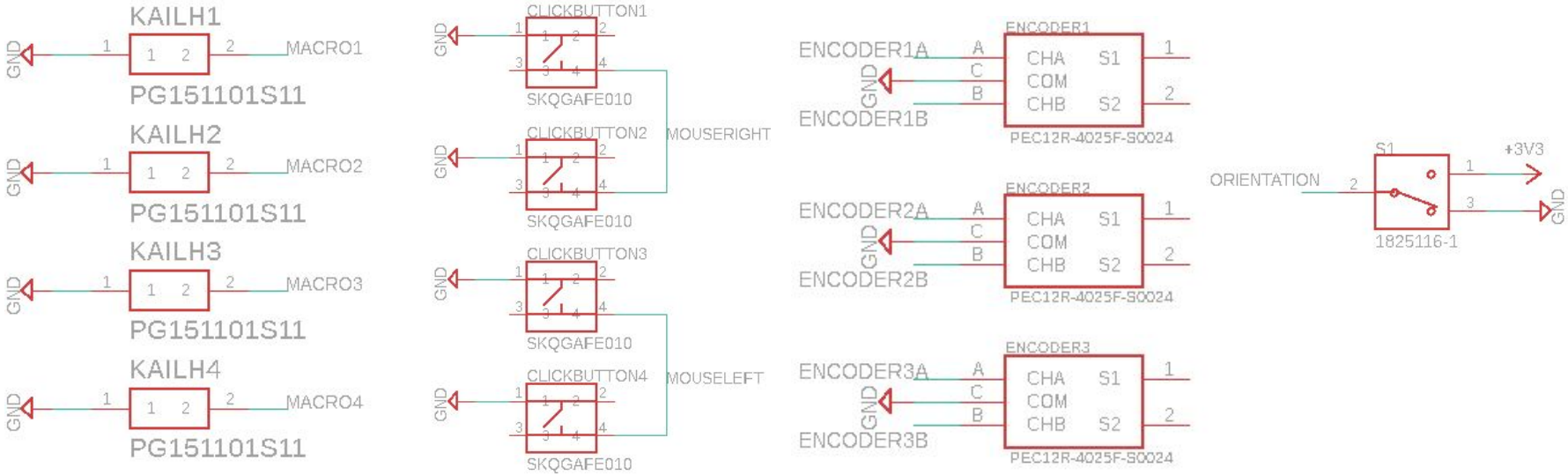






Design Details: Hardware

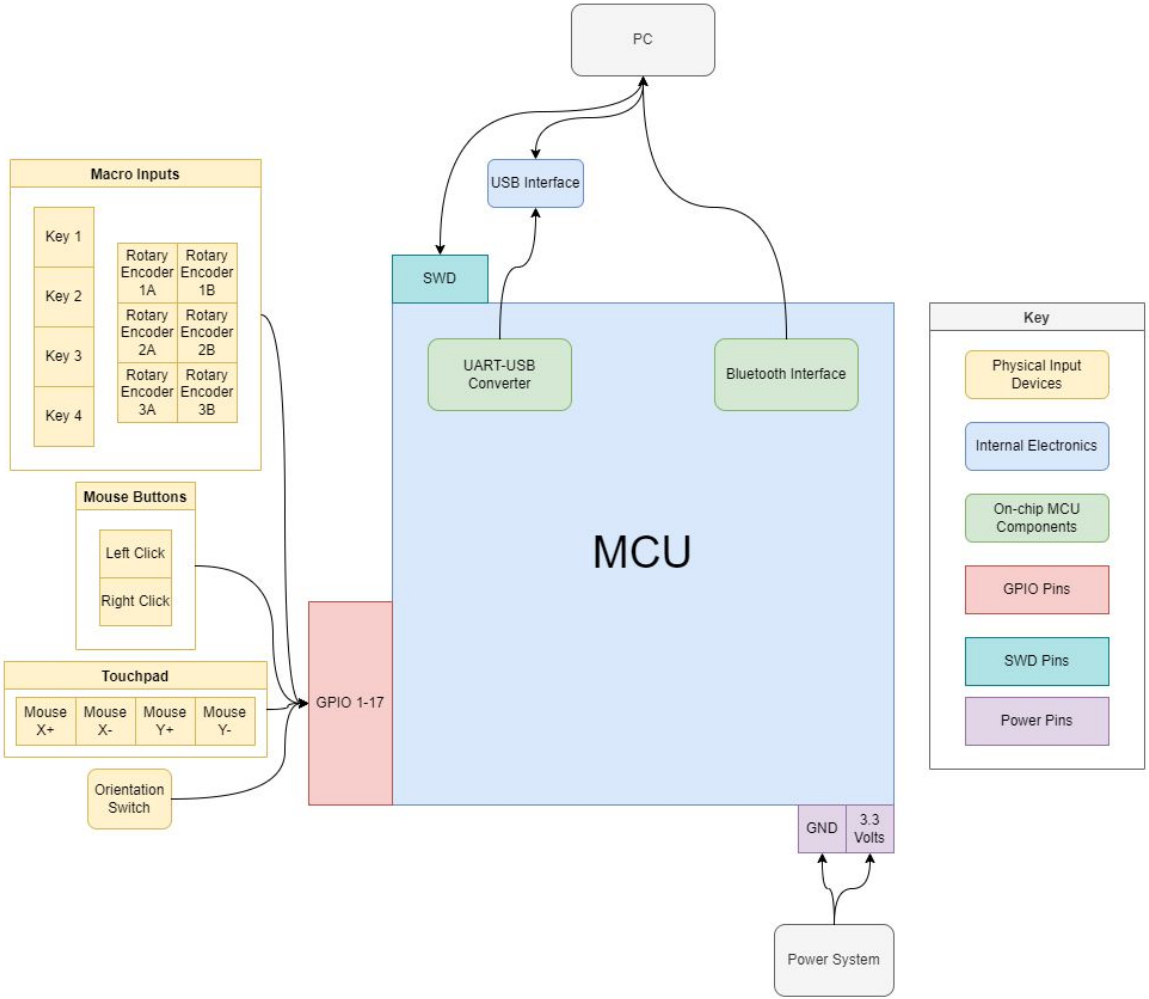
Input Unit Schematics



Design Details: Hardware

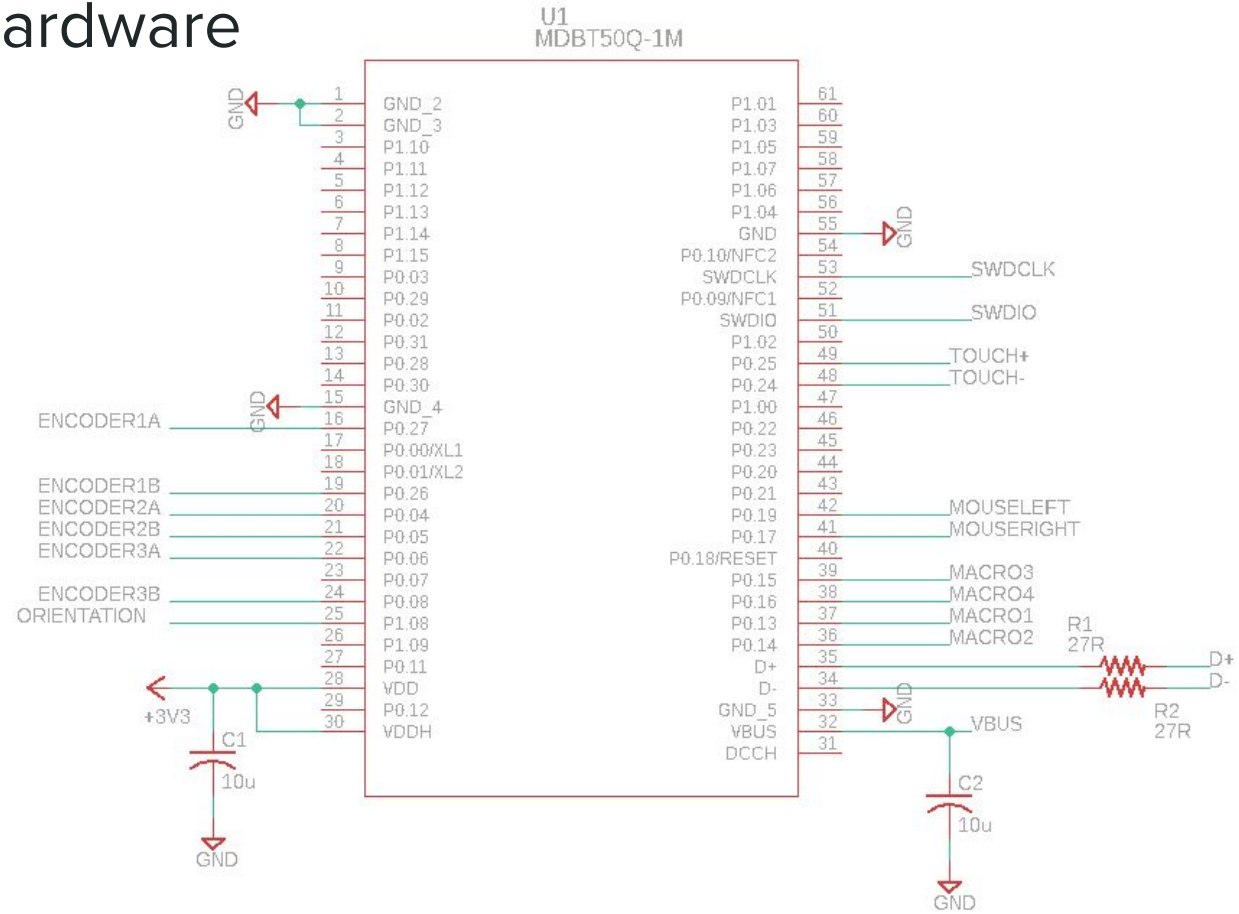
Microcontroller

- The MCU is powered with 3.3V from the power system.
- The input units connect to the MCU via GPIO pins.
- The MCU's Bluetooth and USB interfaces with the user's PC.
- The PC can be used to program the MCU on its SWD pins.



Design Details: Hardware

Microcontroller Schematic

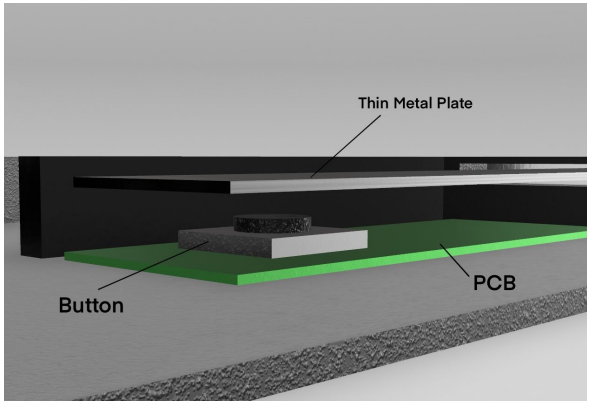


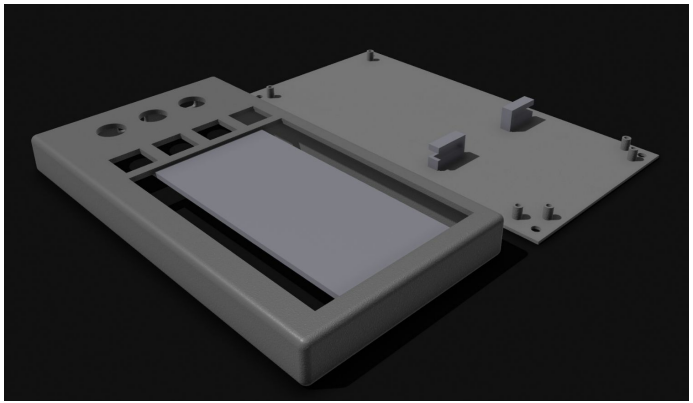
Design Details: Hardware

Chassis

What we needed to consider

- Form factor
- How it would hold the electronics
- Cutouts for input units
- Ambidextrous while being comfortable and ergonomic





Design Details: Hardware

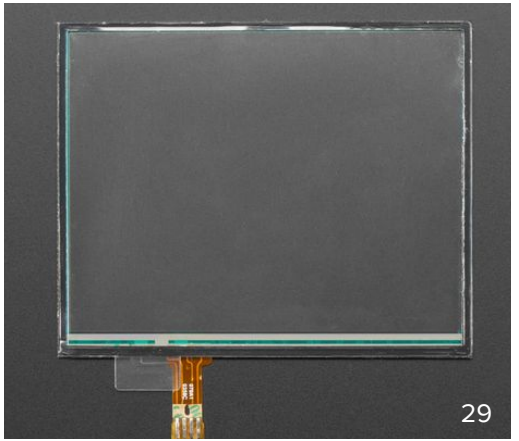
SWD Programmer

- We had to work with a 3rd party programmer due to availability and cost restrictions
- Was able to work with the J-Link and SEGGER software

Design Details: Hardware

Adafruit Resistive Touch Screen

- This touchscreen will pass X and Y coordinates to the MCU via a 4 pin ribbon cable
- The Adafruit HID library will be used to take these coordinates and translate it to mouse movement
- Calculations within the code are made to reduce unintended jitteriness and jumps in mouse movement



Design Details: Firmware

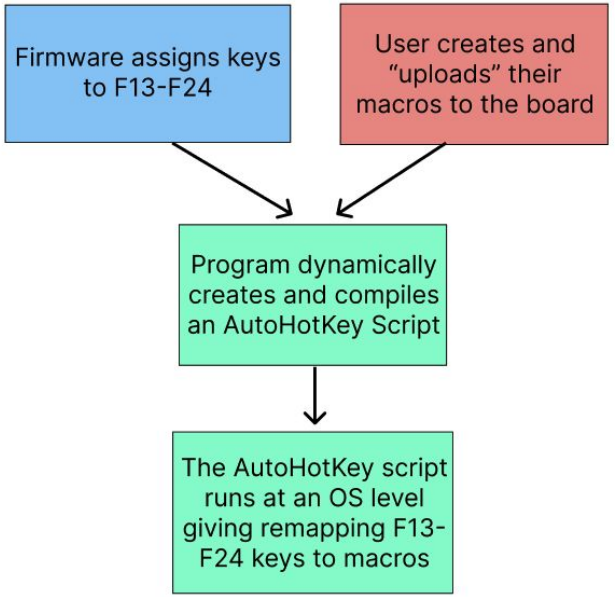
The main functionalities we needed were: keyboard HID, mouse HID, rotary encoder controllers, and Bluetooth

- CircuitPython allowed us to do all that and all this will be preprogrammed where it will interact with AHK
- Main idea was to set the keys to keys people wouldn't typically use (F13 - F17)
  - These would be set to macros or whatever the user wanted through the application
  - This would be done using the AHK scripting
- Rotary Encoders each need two function keys for each channel on the encoder
  - F18-F24

Design Detail: Application Software

A graphical user interface (GUI) is needed for the user to design macros and “upload” them to the device.

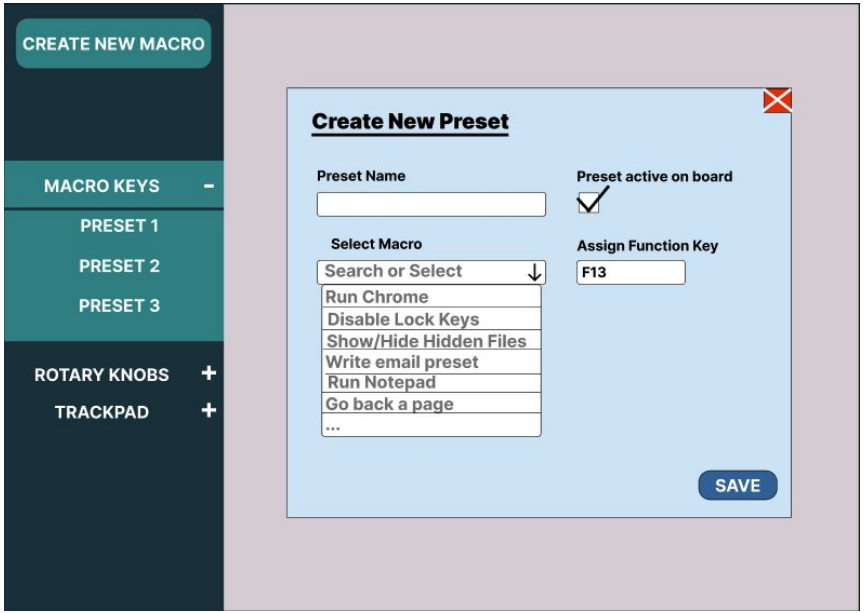
- User interface with forms and dropdowns for the user to design their own Macros.
- Dynamically create and compile one AutoHotKey script that communicates with Windows at the OS level to perform the macro.

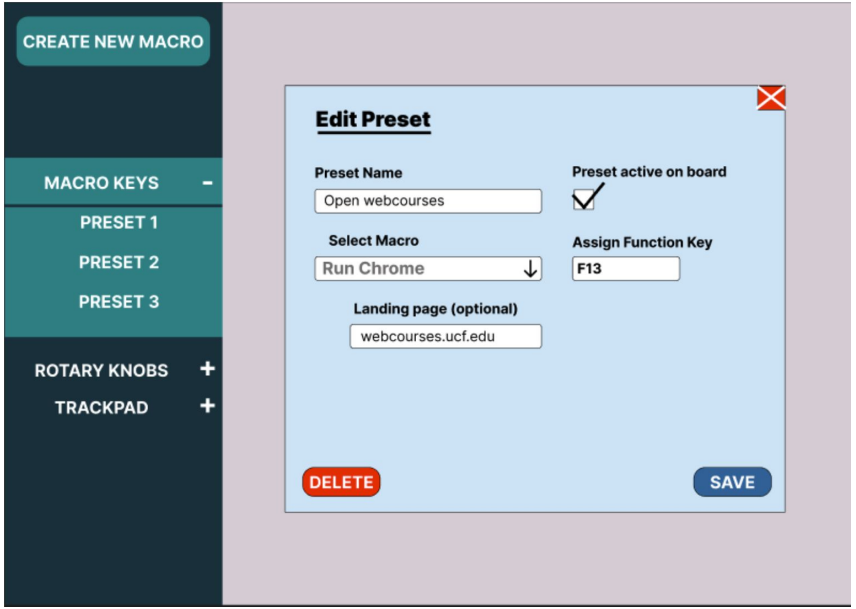


Design Detail: Technology Used in Application software

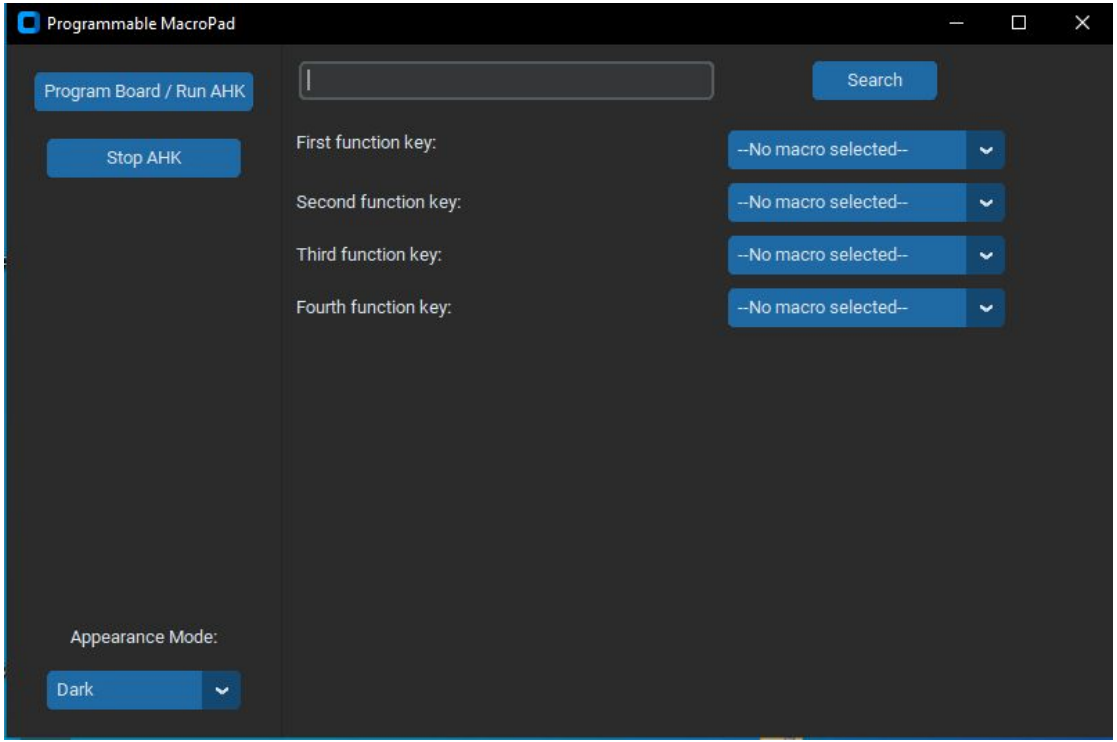
	Frontend	Backend
Technology	TKinter GUI Library	AutoHotKey (AHK), scripting language for windows
Coding Language	Python	Python and AHK macros performing at the OS Level
Features	Open source library, abundant documentation	Dynamically create AHK macro scripts based on the user's choice. Abundant documentation
Compatible with Windows OS?	Yes	Yes
Cost	Free	Free

Design Detail: Figma Application Design Sketch

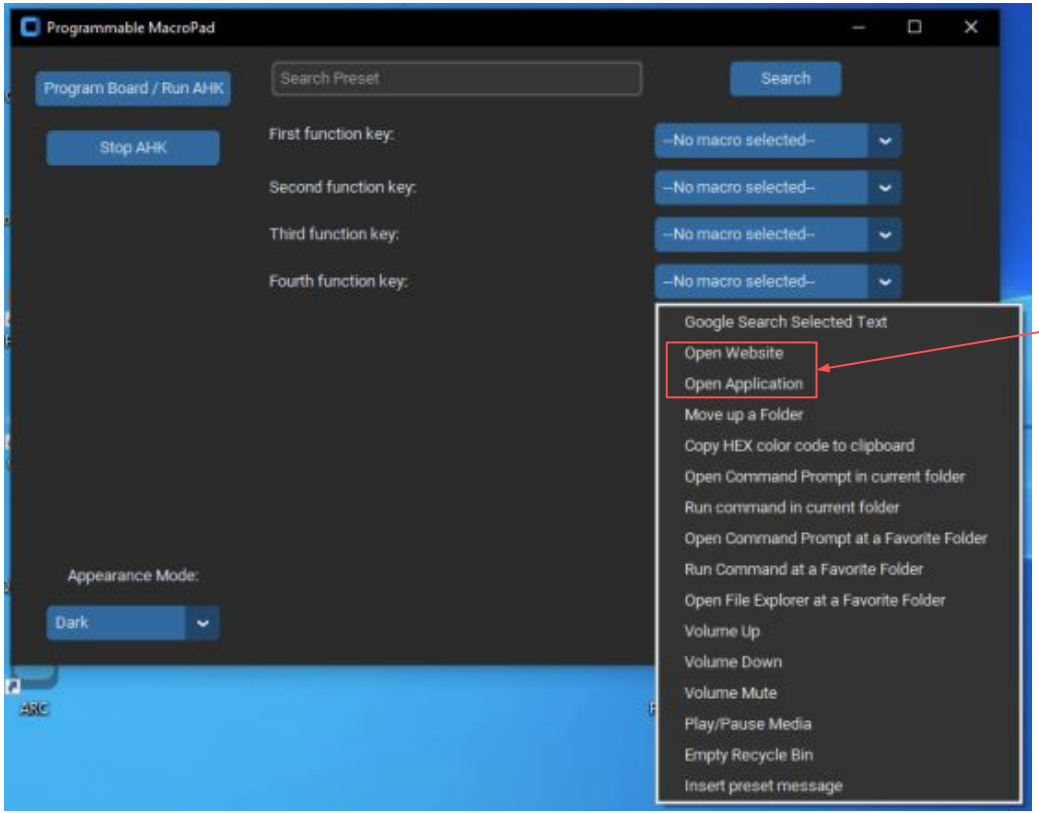




Design Detail: Prototype GUI Design



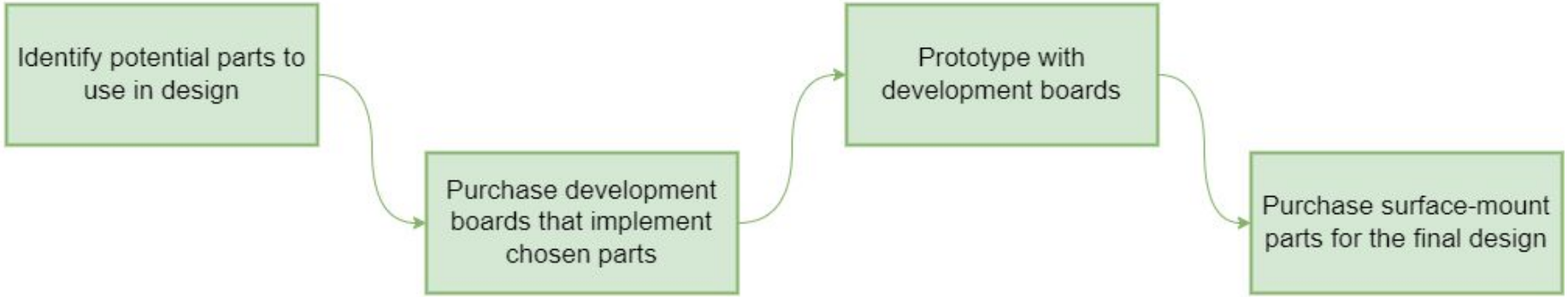
Design Detail Current Progress - Functional AHK Macros



Todo: Implement customizability
"Open Website" and "Open Application" are static, user should be able to choose the website for example.

Fully Functional that don't require user input - examples
"Google Search Selected Text", "Copy HEX color code", or Volume Control. User input isn't required, performs the macro successfully.

Finance: Strategy



Finance: Development Budget

- These materials were purchased as part of the prototyping phase of development.
- Total development budget: $142.17

Item	Price	Purpose
3.7" Touchpad	$11.45	
Touchpad to USB breakout board	$17.59	Prototyping touch functions
500 mAh LiPo battery	$6.99	
1700 mAh LiPo battery	$10.99	
18650 battery charger	$5.99	
USB-C Battery Management board	$8.99	
USB-C Battery Charging board	$12.99	
TPS61023 Development Board	$3.56	
LM3671 Buck Converter	$11.22	Prototyping power system
Rotary Encoders	$9.00	
Mech Switch Breakout Board	$3.25	Prototyping HID commands
MDBT50Q	$12.95	
Itsbitsy nRF52840 Express	$27.20	Prototyping microcontroller

Finance: Bill of Materials

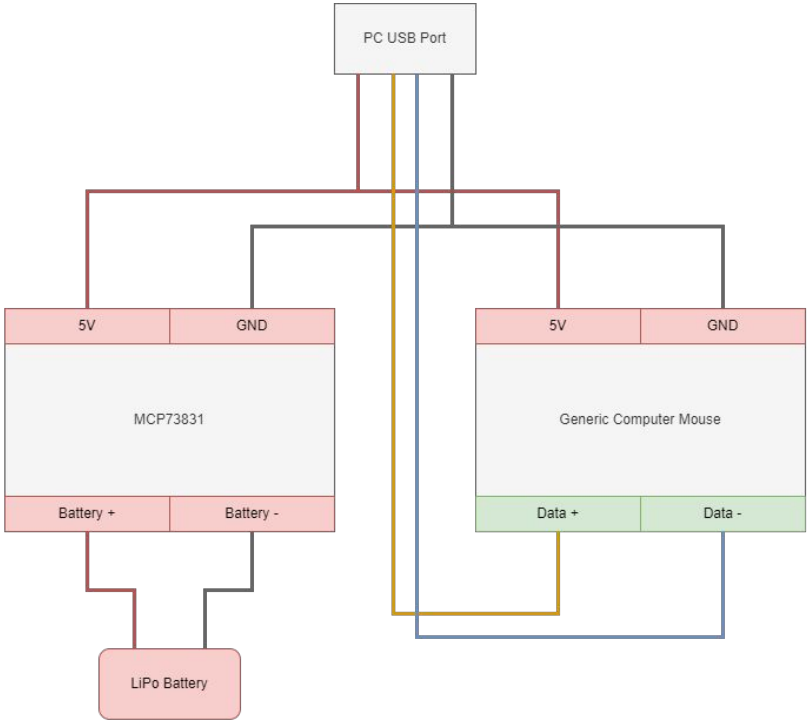
- These are the materials that would be necessary in order to manufacture one Programmable Trackpad.
- Total cost: $49.92

Item	Quantity	Price
MDBT50Q	1	$12.95
500 mAh LiPo battery	1	$6.99
LM3671 3.3V Converter	1	$1.61
TPS61023 5V Converter	1	$1.23
MCP73831 Battery Management System	1	$0.77
3.7" Touchpad	1	$11.45
Rotary Encoder	3	$2.90
Kailh Socket Pack	1	$2.50
PCB	1	$2.00
Ribbon Cable Receptacle	1	$0.36
Micro USB Receptacle	1	$1.36

Current Progress

Charging and Transmission Prototype

- The original prototype attempted to charge a lithium polymer battery on a PC's USB port while a mouse is plugged into the same USB port.
- This prototype failed because the combination of devices caused the USB port to dip well below 5 volts, which could not power either the BMS or the mouse.
- Subsequent prototypes involved a 5 volt regulator to ensure that the BMS and mouse would both be adequately supplied.



Current Progress

Bluetooth/USB Encoder and Mech Switch Prototype

- ItsyBitsy development board bluetooth connection, input units would only work for a bit then cut out on battery
  - Most likely due to the battery pin damage from past testing
  - With a simple full charge of the battery and cleaning up the foundation code, it worked flawlessly

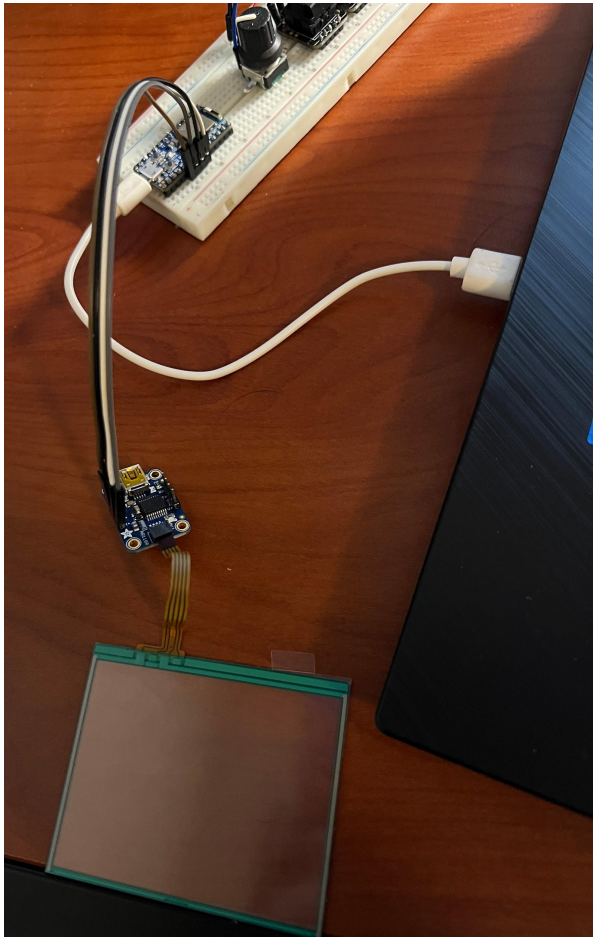
Programming a Blank Microcontroller

- Mainstream SWD Programmers weren't available
  - Was something we all were unfamiliar with
  - Looking to a 3rd party programmer

Current Progress

Touchpad Prototype

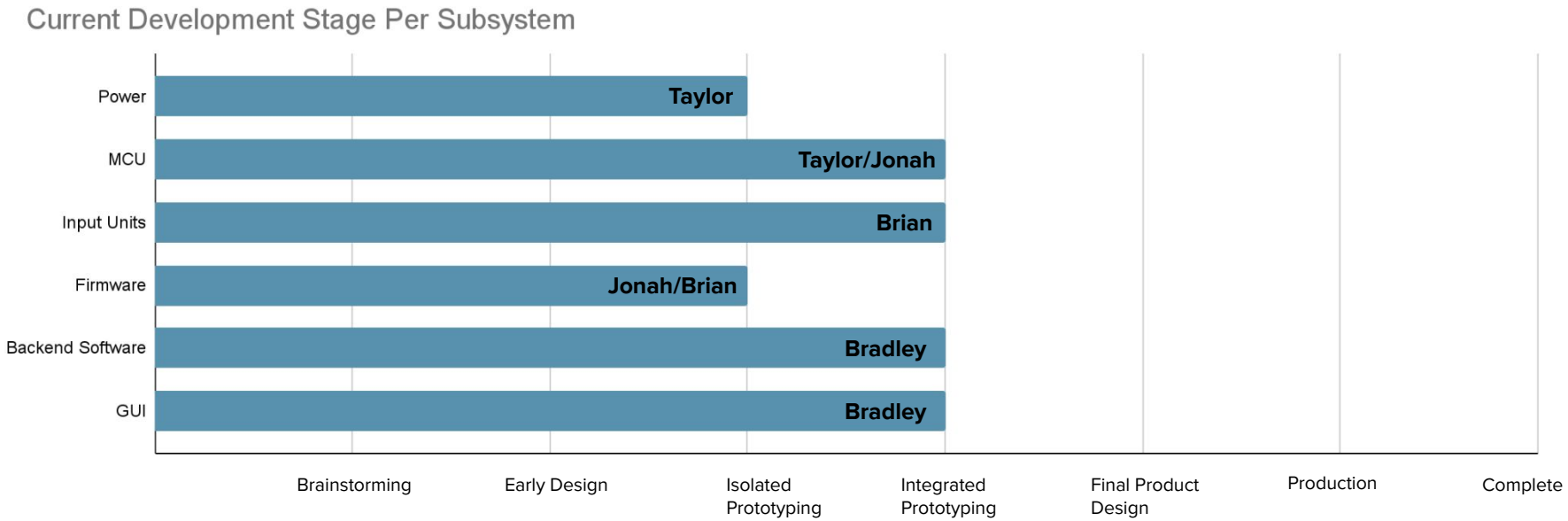
- While prototyping the touchpad, we initially faced difficulties getting the touchpad to translate inputs into smooth cursor movements
- We tried a few different Circuit Python libraries and experimented with the functions that each one had to offer
    - Some libraries wouldn't read the touchpad data, while others would move the mouse in a jittery and imprecise way
- We ended up finding a solution within the Adafruit HID Library which has a function that moves the mouse directly to an x, y position on the screen
    - We simply fed the stream of coordinate data through this function



Current Progress - Application

- AutoHotKey scripts can be dynamically created based on the user's input
- AutoHotKey can be programmatically compiled and then communicates with Windows to perform the desired shortcut macro
- User interface customizability is limited to a dropdown at the moment. Some macros will require user input such as "Open Application" or "Open Website" where the user must supply additional information to have the macro fit their shortcut needs.
- An additional dropdown is needed for the rotary encoder. This dropdown would only includes macros that would make sense for the encoder such as Volume Control, brightness adjustment, scrolling etc.

Current Progress



Next Steps

- Test firmware programming on microcontroller.
- Add more macro functions to application.
- Add rotary encoder support to application.
- Design PCB using existing schematics.
- Finalize chassis design.
- Implement mouse click mechanism.
- Integrate all subsystems.
- More user customization and custom macros in application software