Click Here for Original Web Version

Maketato - A DIY MakeCode Arcade Handheld for Kids

Sep 25, 2024

In my opinion, game programming provides kids with the most engaging and effective introduction to the field of computer science. Most kids already have a game concept or two knocking around in their head and, with the help of a streamlined development environment, it’s amazing how quickly they pick up logical and algorithmic concepts. In the past few years, Microsoft’s MakeCode Arcade has become my go-to coding tool for kids of all ages. The block-based interface is intuitive for absolute beginners and familiar to those who’ve had prior experience with LEGO’s SPIKE Prime or MIT’s Scratch environments. The ability to flip back and forth between blocks and JavaScript helps more experienced kids transition to true text-based, structured programming. And of course there’s the built-in simulator that provides instand visual (and auditory!) feedback, the thoughtfully designed libraries, the playful tutorials, the excellent documentation… I’m gushing, but trust me, it’s good.

On top of all that, MakeCode allows you to download the games you create and play them on dedicated handhelds. With a bit of parental assistance, it’s possible for kids to construct devices on which can they play games of their own creation, building both their STEAM skills and self-confidence.

This Summer I ran a weeklong camp for my son and his friends where they did just that. Although the program itself was a success, I found putting together kits and lesson plans using existing online resources to be incredibly frustrating. The designs I purchased for the handhelds looked cool but required extensive customization, wouldn’t print properly from my cheap 3D printer and were difficult for the kids to assemble. For the sake of other parents’ and educators’ sanity, I decided to redesign everything from scratch and document it here.

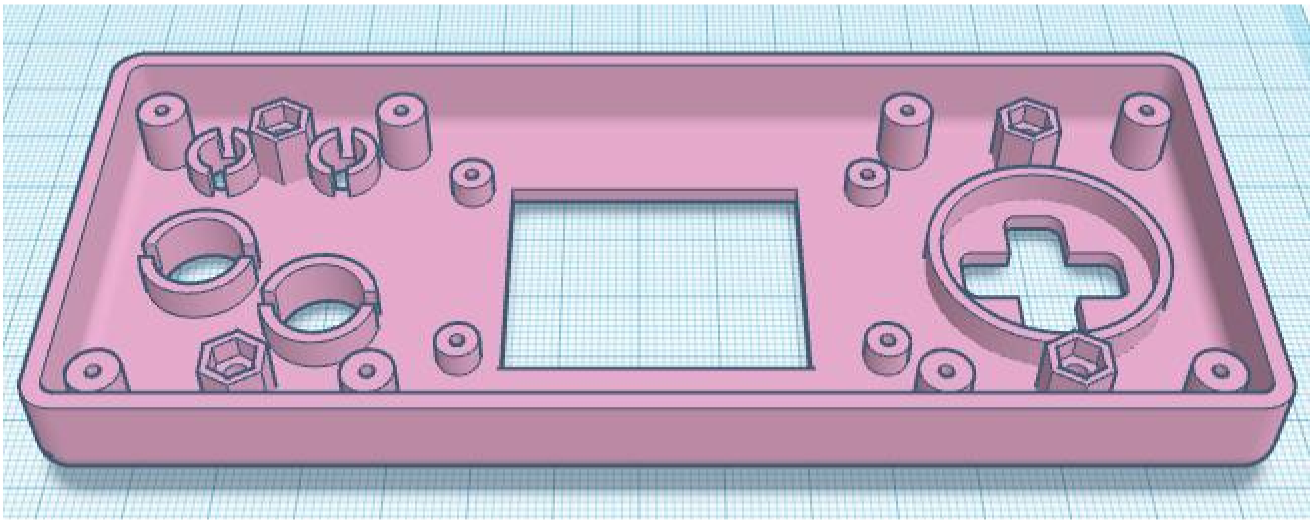
Introducing the Maketato

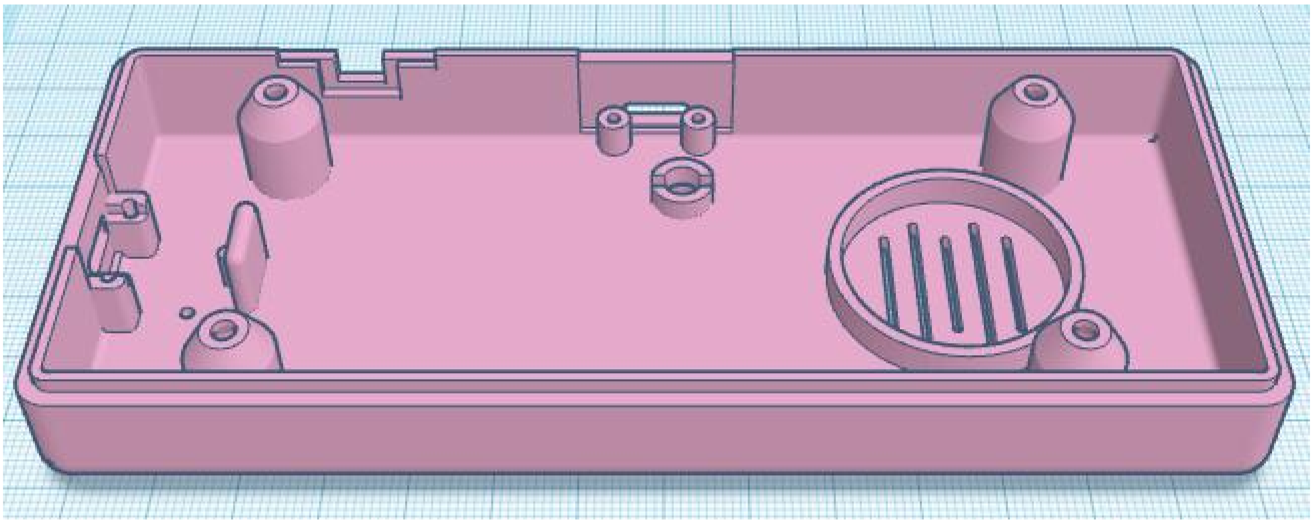
Maketato is a 3D-printed, MakeCode Arcade-compatible[1] handheld that you and your kids can build at home. To keep costs down, it incorporates only cheap, readily-available and reusable parts such as the Raspberry Pi Pico. The enclosure is durable, sized for small hands, and will print quickly and without drama on nearly any 3D printer. Assembly is straightforward enough for a child to perform but not *so* simplified as to eliminate the possibility of mistakes and the learning that comes from them.

To build your own Maketato you’ll need to gather the listed materials, print out the case and other components, and assemble them according to the instructions below. Please note that the current version of the build requires a bit of soldering and, depending on their age and ability, children will need parental supervision and assistance. Not to ruin the vibe, but YOU ACKNOWLEDGE THAT USE OF THESE INSTRUCTIONS IS AT YOUR OWN RISK.

Gathering Materials

3D Models





The Maketato’s case was designed from the ground up to be durable, easy to print and fit comfortably in small hands. The package I’ve made available on Cults3D includes STL files for the front and back cases as well as each of the buttons. I’ve also thrown in an alternative back case design that allows for safer but less convenient AA battery power.

[Download the Maketato Models (Cults3D)]()

Assembled, the AA battery-powered version of the Maketato measures 144mm wide, 60mm tall and 33mm thick. The USB-C / Lithium Polymer variant is more swelte at just 28mm thick.

Required Components

My goal in picking components for this build was to provide maximum reusability at minimal cost. The Raspberry Pi Pico microcontroller at the heart of the Maketato costs about $5 and can be repurposed for countless other educational projects. The same goes for the $7 LCD screen, speaker, jumper wires and buttons. The total component cost per device when building 4 or more should be less than $20, making it viable for classroom and afterschool program use.

Item	Quantity	Buy From
Raspberry Pi Pico	1	Amazon
1.8" ST7735 TFT LCD	1	Amazon or eBay

Item	Quantity	Buy From
1/4 Size Breadboard PCB (ElectroCookie)	2	Amazon
6x6x5mm Tactile Switch	8	Amazon
2W 8 Ohm 28mm Speaker	1	Amazon
15cm Female-to-Female Dupont Jumper Wires	24	Amazon
Breadboard Jumper Wires (or any 24AWG Solid Wire)	10	Amazon
Right-angle Headers	32	Amazon
SPDT Mini Slide Switch	1	Amazon
M2 4mm Hex Socket Head Cap Bolts	18	Amazon
M3 8mm Hex Tapered Flat Head Cap Bolts	4	Amazon
M3 Nuts (Stainless Steel)	4	Amazon

For LiPo Battery Power

Item	Quantity	Where to Buy
TP4057 USB-C Charge Controller	1	Amazon
1100mAh 3.7V LiPo Battery (603449)	1	Amazon
JST 2.0 Female Battery Connector w/Leads	1	Amazon

For 2xAA Battery Power

Item	Quantity	Where to Buy
Nickel Plated AA Battery Contacts	2 pairs	Amazon
NiMH AA Rechargeable Batteries	2	Amazon

Tools

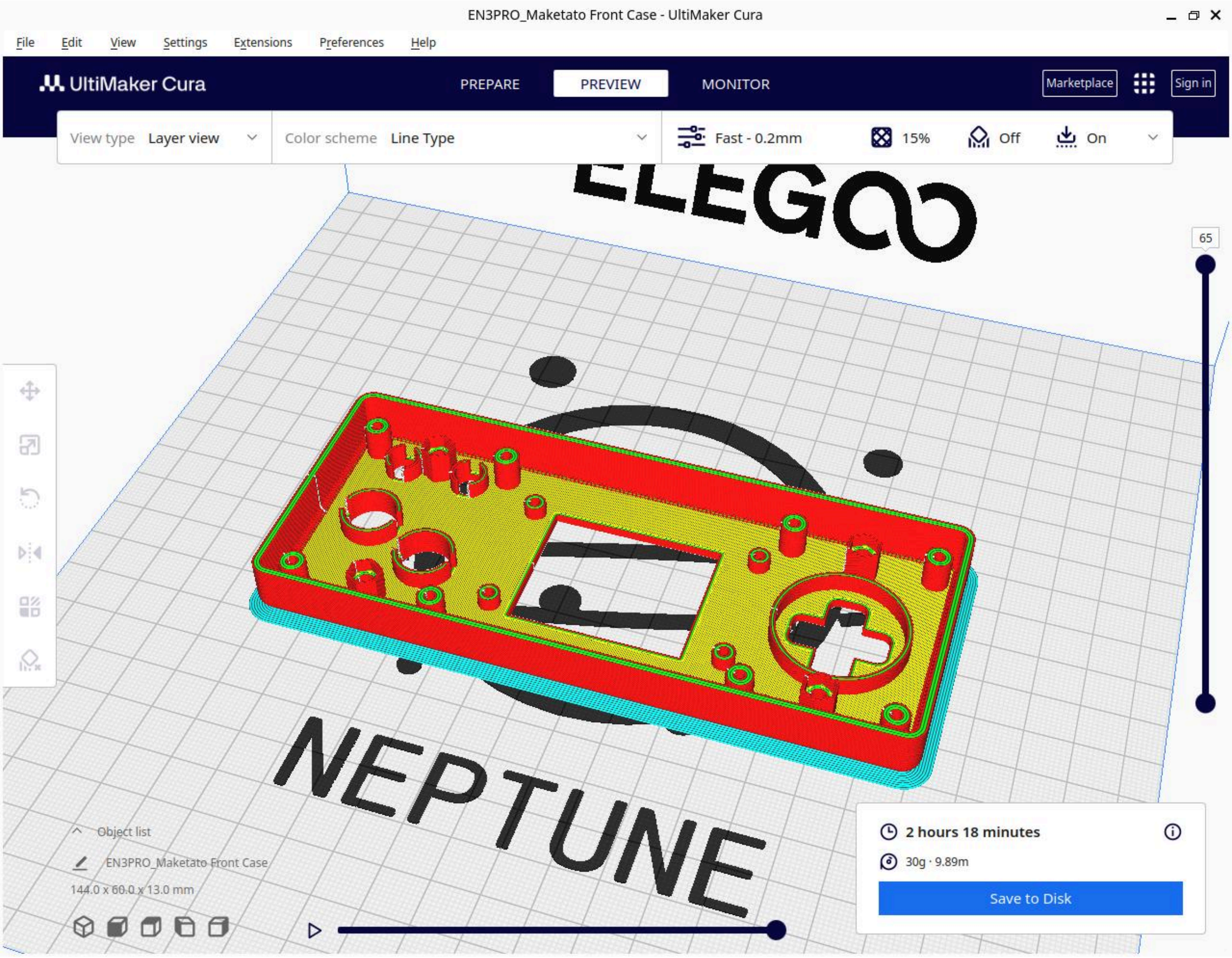
Item	Where to Buy
3D Printer (e.g. my Elegoo Neptune 3 Pro)	Amazon
Soldering Iron (e.g. this Hakko)	Amazon
Solder (e.g. this cheap but adequate Pb-free one)	Amazon
Flush Cutter	Amazon
Wire Stripper (e.g. IRWIN VISE-GRIP)	Amazon
USB-A to Micro USB Cable	Your junk drawer
M2 and M3 Hex Bolt Allen Keys	Typically included with bolts

Printing the Case

Slicing the Models

I’ve been using Ultimaker Cura 5.8.0 to slice the models for my stock Elegoo Neptune 3 Pro FDM printer. The *Balanced - Fast (.2mm layer height)* preset with

Adhesion (brim) enabled and Support disabled works well for the front and back case models. I use 15% Infill for both of the case models. The front case measurements are 144x60x13mm and the version of the back case designed for USB-C charging is 144x60x17mm. The back case of the AA battery-powered variant is a bit taller at 144x60x22mm. Orient the models such that they fit within the printable area of your build plate.



EN3PRO_Maketato Front Case - UltiMaker Cura

View type Layer view

Color scheme Line Type

Fast - 0.2mm

15%

Off

On

2 hours 18 minutes

30g · 9.89m

Object list

EN3PRO_Maketato Front Case

144.0 x 60.0 x 13.0 mm

The buttons can be printed using the same preset or, for cleaner results, the *Balanced - Fine (.1mm layer height)*, again with Adhesion (brim) enabled and Support disabled. You will need to print two copies of the ‘Maketato Button - AB’ model (or Heart / Star button models, which are *cute* and interchangeable), two of the Menu/Restart button models, and a single DPad and Bootsel. You can arrange all the required buttons on the slicer’s virtual build plate to be printed in one go, though the quality will be better if you print them separately or one at a time, using Cura’s ‘Print Sequence’ option.

Printing the Parts

I printed all of my prototypes on an Elegoo Neptune 3 Pro using Elegoo’s own PLA+ filament. I found a nozzle temperature of 215F and build plate temp of 70F worked well for this material at a 60mm/s print speed. Using my recommended settings, the front case prints in just over 2 hours and the back case in about 3.

The buttons pictured here were printed in the same material in beige but for fun I also printed out a set in Anycubic’s ‘PLA Luminous’ glow-in-the-dark filament. They came out great using a lower 200F nozzle temp on the *Balanced - Fine (.1mm layer height)* preset.

Be sure to remove any extraneous burrs, bumps and strings from your prints with a craft knife before continuing to assembly.

Assembling the Handheld

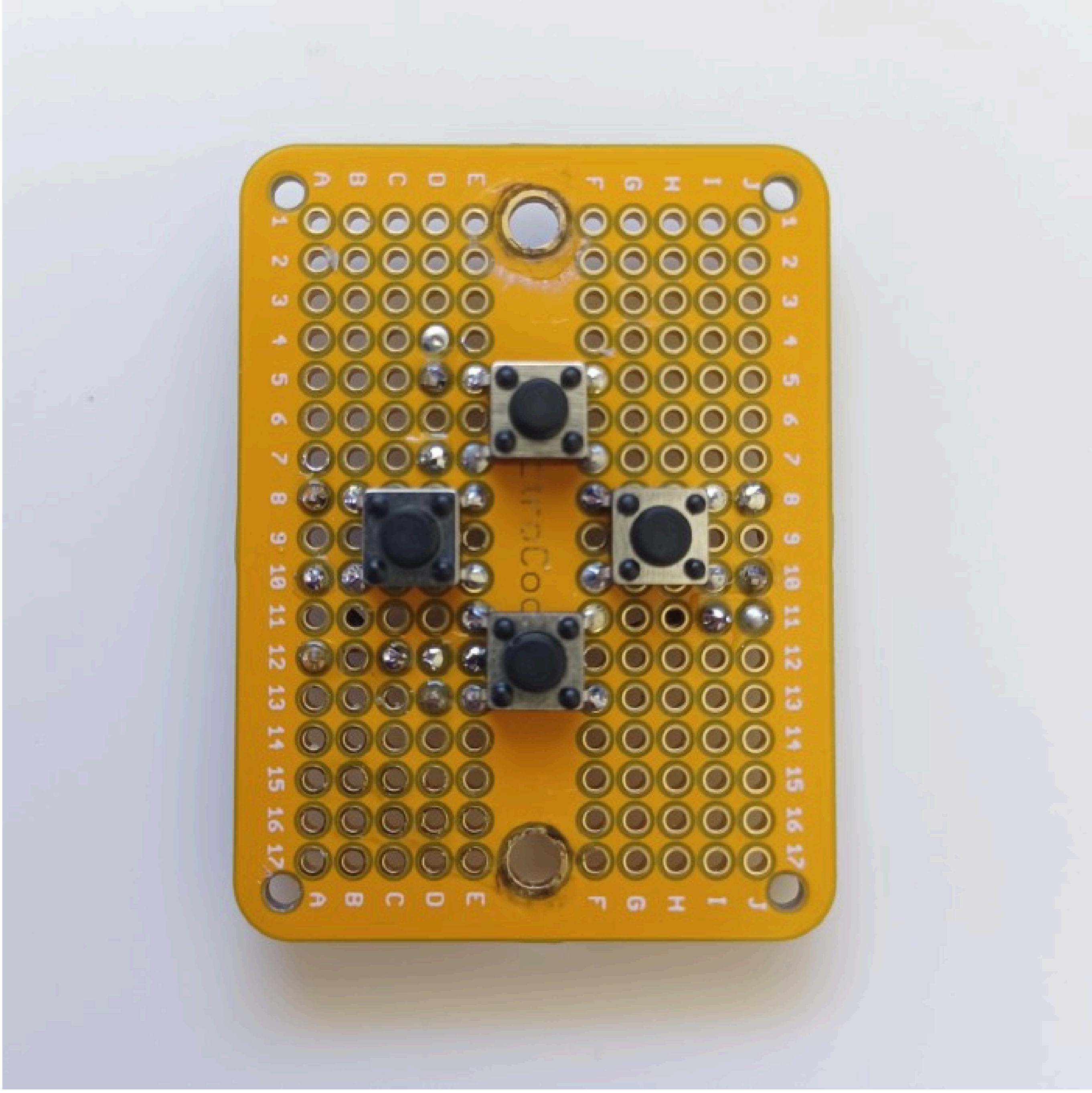
Step 1: Preparing the Controller Boards

The Maketato’s controls are split between two separate boards mounted to the left and right of the screen. Before assembling the front case, we need to complete both controller PCBs, which will require some soldering. If you’ve never soldered before I recommend watching a few tutorials on YouTube before proceding. While the soldering tasks for this project are by no means difficult, you’ll definitely want to familiarize yourself with standard techniques and safety measures. If you’re assisting a child with this build, I suggest knocking out all the soldering ahead of time, leaving them with the safer tasks of wiring and assembly.

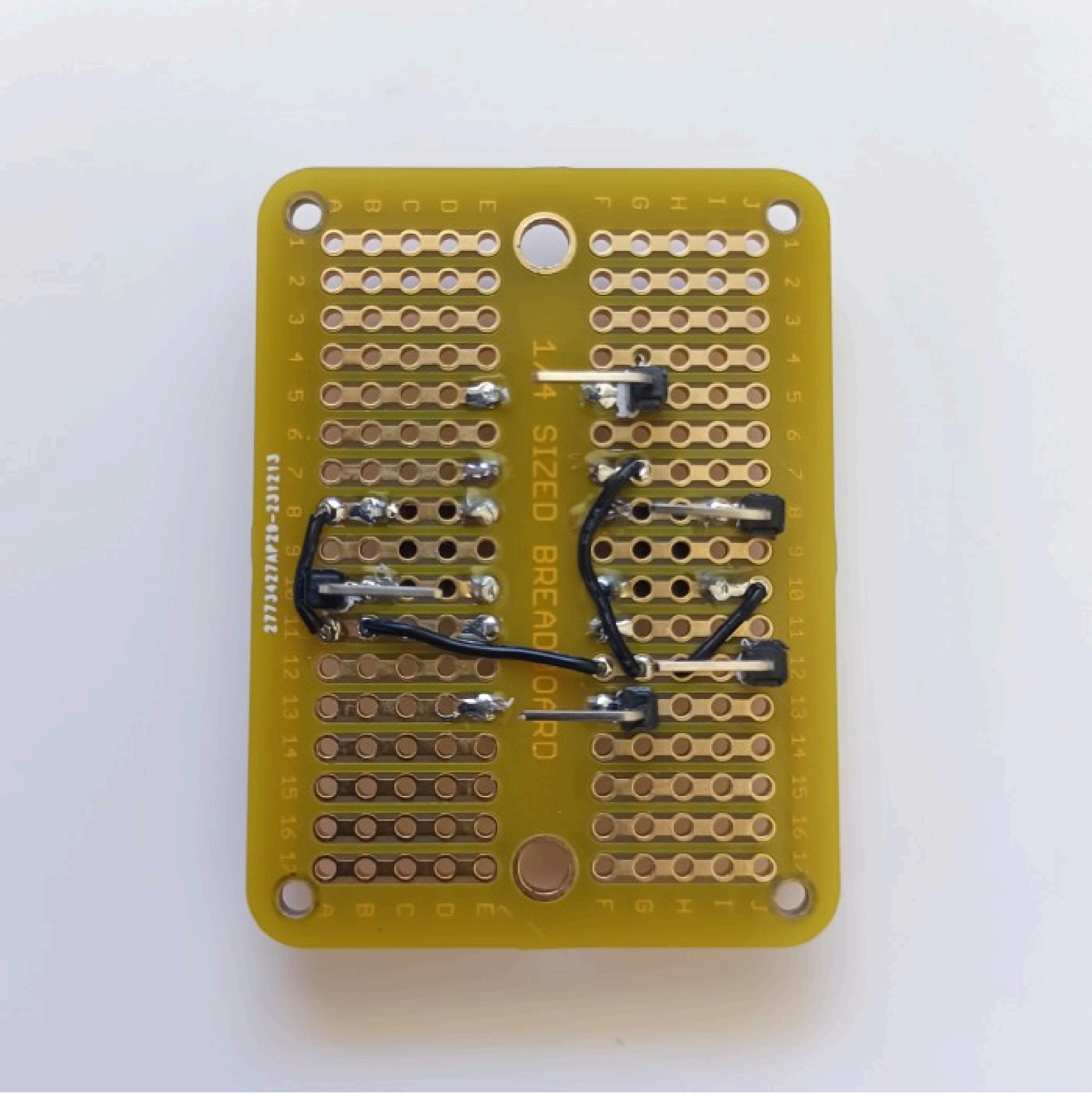
Let’s start with the directional pad. Pop four tactile switches - one for each of the cardinal directions - into a 1/4 size breadboard PCB according to the following layout:

DPad Switch	Board Holes
Left	I10, I8, F10, F8
Right	E10, E8, B10, B8
Up	F7, F5, E7, E5
Down	F13, F11, E13, E11

Press down firmly on each button to ensure it’s sitting flat against the board. Then flip the board over and carefully bend the tips of the legs inward so the buttons remain in place. Solder in all four legs on each switch.



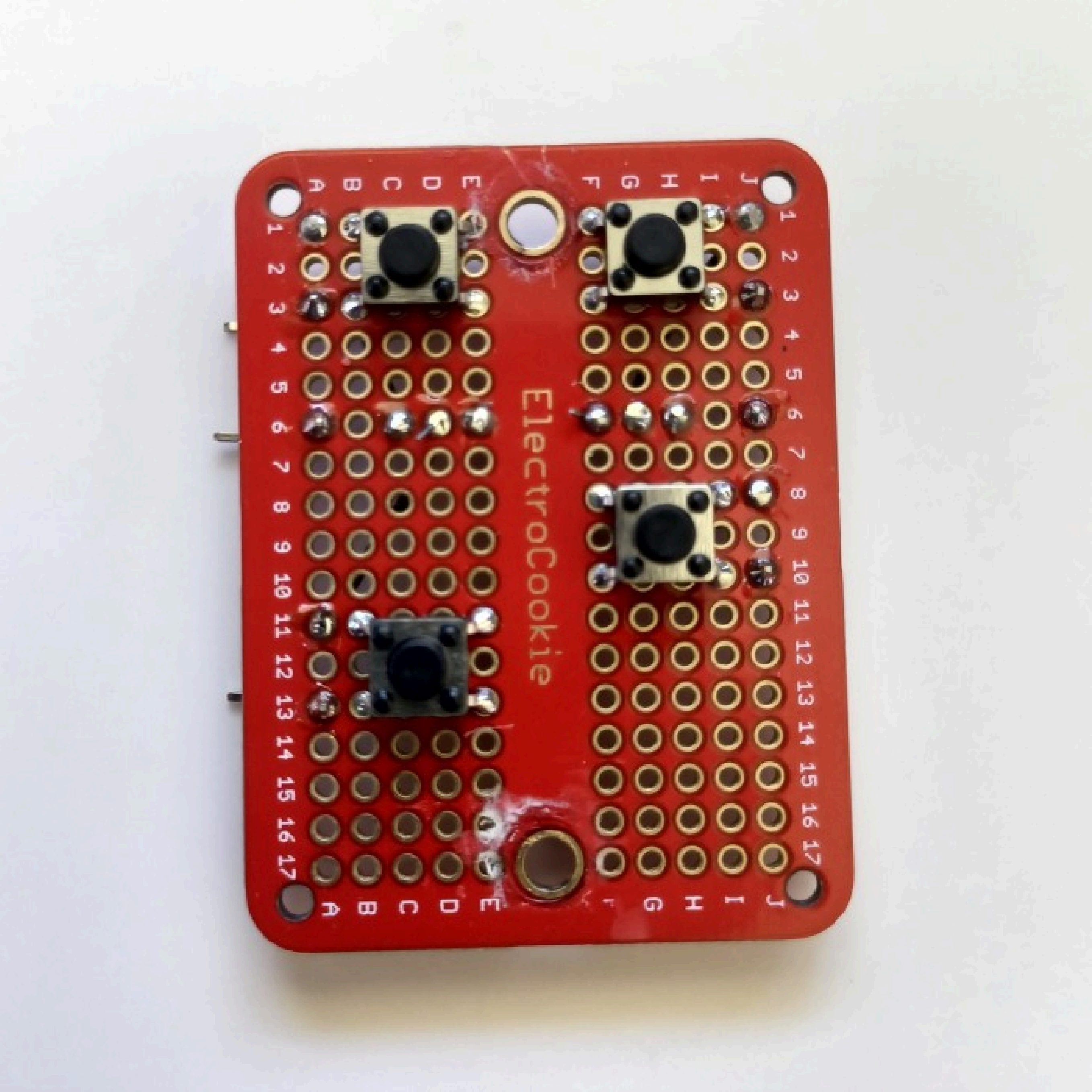
Next, pick an unoccupied row of through holes to be the common Ground. For each of the four switches, run a 24AWG jumper wire from a hole adjacent to any of its legs to our Ground row, trimming excess lenght with the flush snips. Using a pair of mini pliers, snap off 5 individual right-angle headers. Later on, these will allow us to connect the board to the Pi Pico and read directional inputs. For each of the switches, insert a header pin into a hole connected to the leg adjacent to the one wired to the Ground row. Place the final header pin in the last hole of the Ground row. Using heat resistant tape or poster tack to hold the wires and headers in place, then flip the board over and solder them all in.



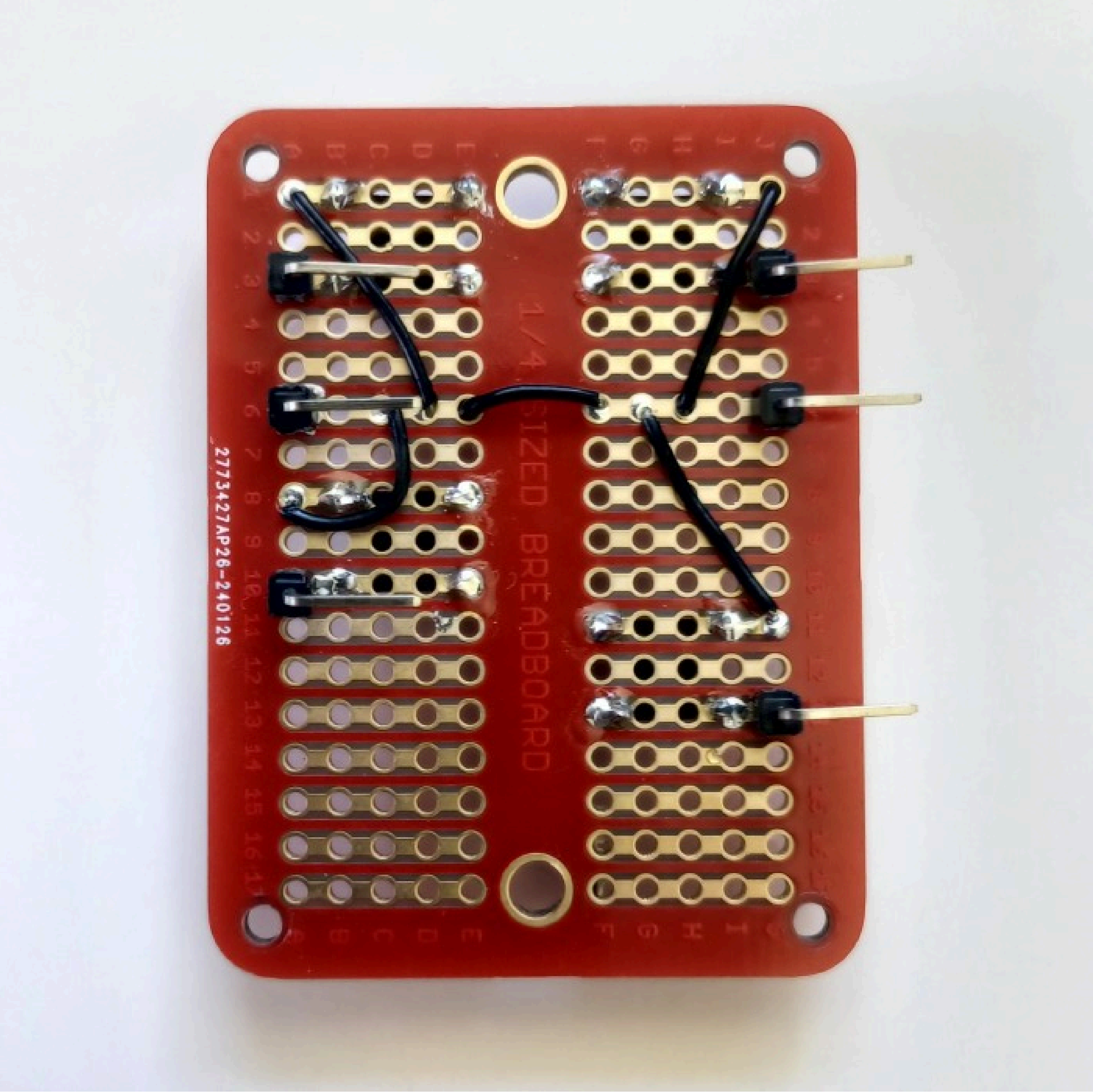
The completed board should like something like the above. If you have a multimeter on hand, it would be good idea to verify that there’s continuity between each switch’s header pin and the common Ground pin when its button is pressed.

Moving on to the action button PCB, follow the same procedure as before but for the following switch layout:

Action Switch	Board Holes
Restart	B1, B3, E1, E3
Menu	F1, F3, I1, I3
A	F8, F10, I8, I10
B	B11, B13, E11, E13



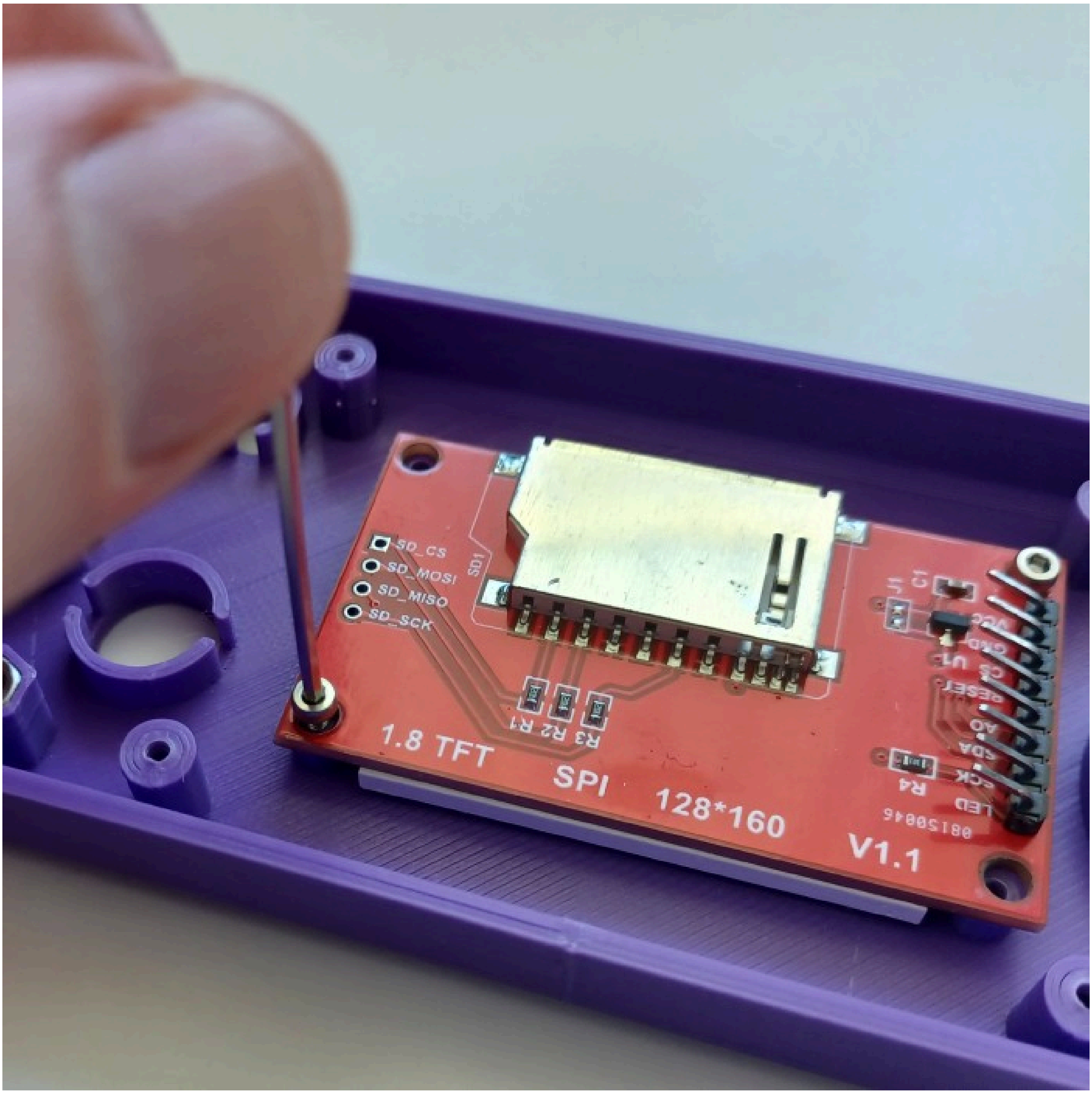
We’ll solder an additional header pin in the Ground row on this board, allowing us to daisy chain the Ground from the directional pad PCB and connect all the switches to a single Ground on the Pi. For my build, I soldered a jumper between the left and right sides of row 6 to make an extended Ground row.



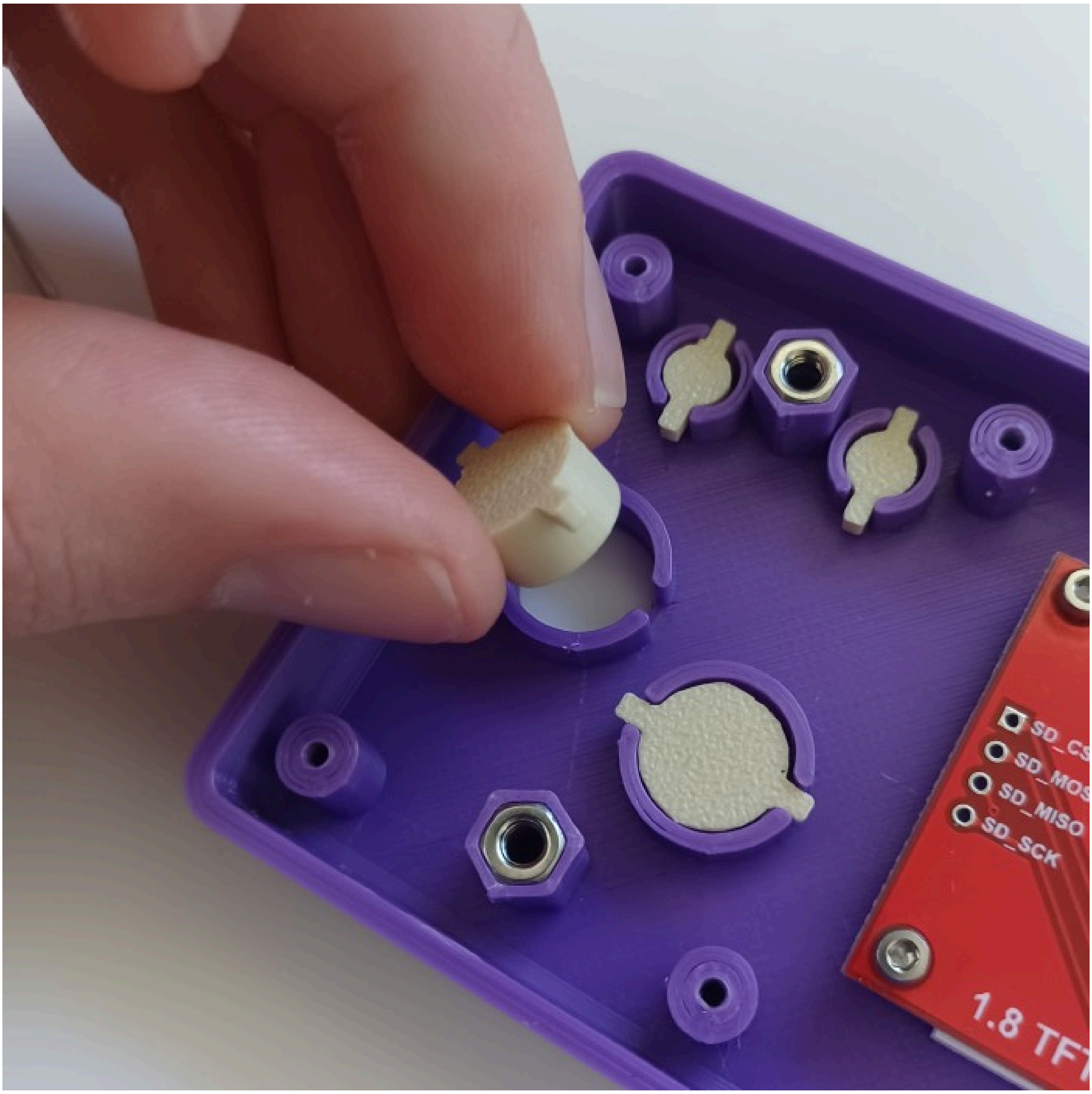
Step 2: Assembling the Front Case

With the two controller boards complete, we can now assemble the front case. First, use the handle of a screw driver to gently tap an M3 nut into each of the four hexagonal posts.

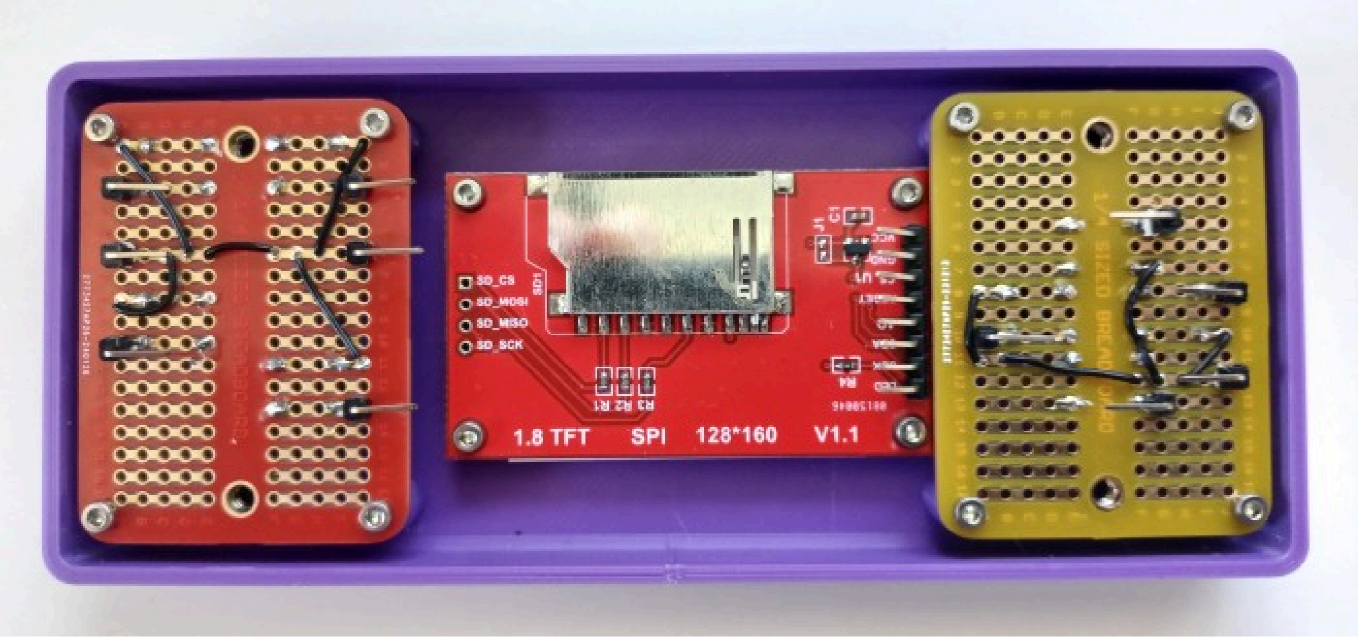
Next, peel the protective plastic off of the LCD screen and place it face-down in the center of the front case with the 8 pin header on the LCD board on the side with the DPad cutout. Align the mounting holes in the LCD board with the four posts around the screen cutout and screw a 4mm M2 bolt into each.



Drop the 3D printed DPad, A, B, Restart and Menu buttons into their respective cutouts, then place the controller PCBs - switch side down, of course - on top of the mounting posts on the left and right sides of the front case. Secure each controller board with four 4mm M2 bolts.



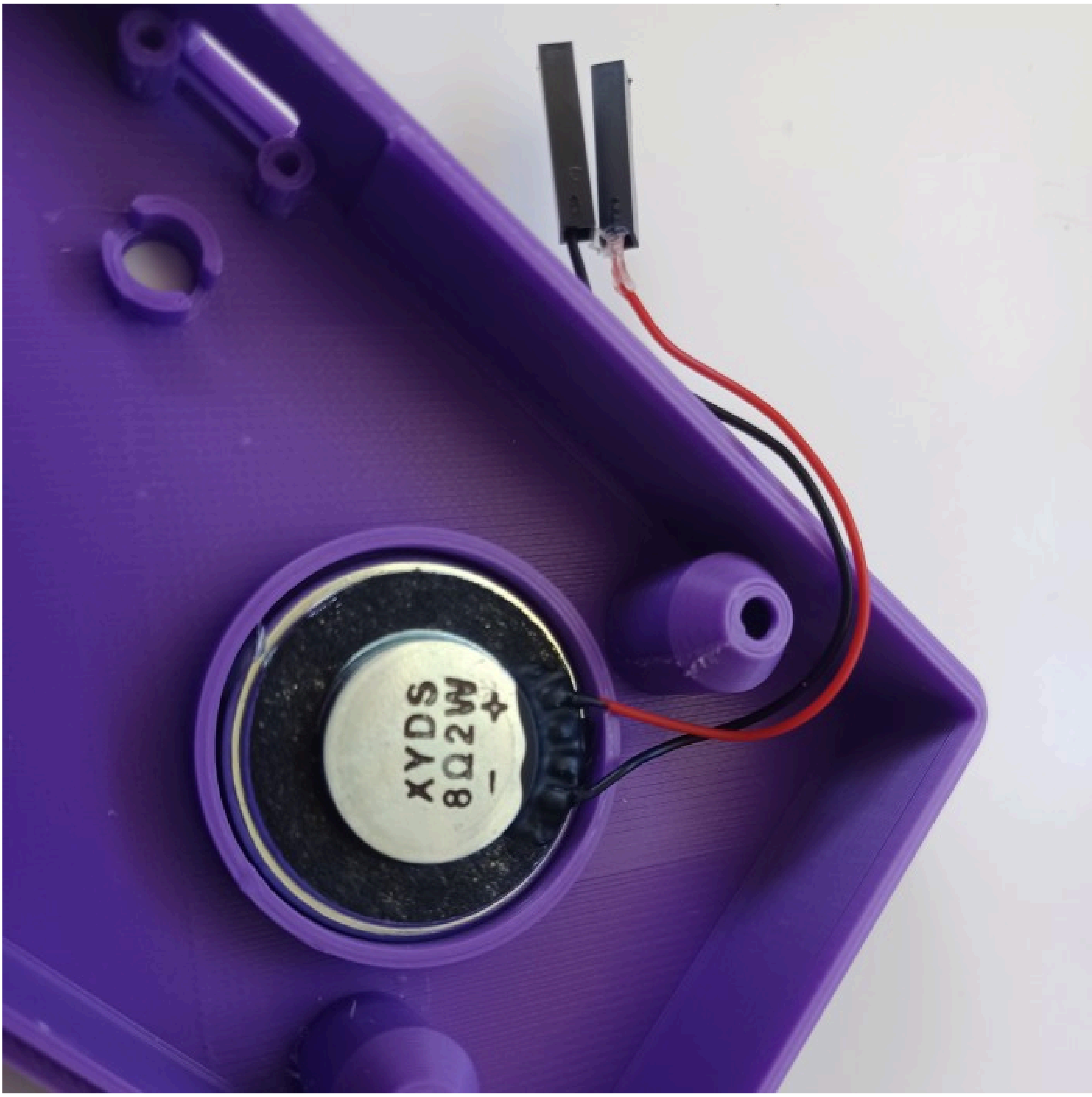
And just like that, we’re halfway done! Feel free to skip ahead to the section on wiring if you’d like to test out the screen and buttons before moving on to the back case.



Step 3: Assembling the Back Case

Dropping in the Speaker

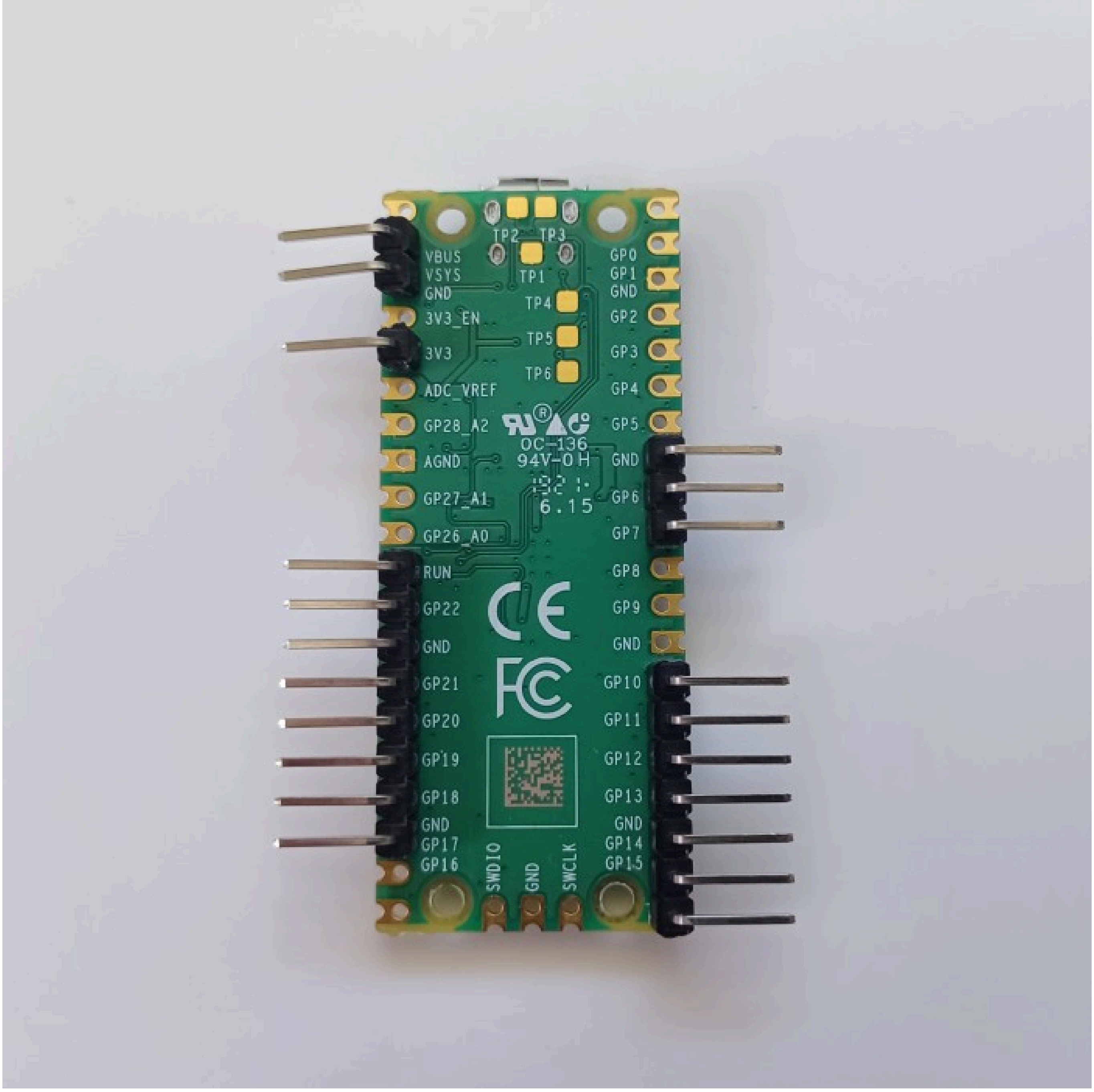
If your speaker doesn’t already have female Dupont connectors on its leads, I recommend putting a couple on there before installing it in the back case. You can do this either by taking a female-female Dupont jumper wire, cutting it half and splicing the ends with the bare speaker leads or by crimping on Dupont connectors with a special tool.



Once the speaker’s been prepped, squirt a blob of hot glue into the inner edge of the back case’s speaker well and squish the speaker into place - cone side down - on top of it.

Preparing the Pi Pico

To simplify wiring up the Pi we’ll be soldering right-angle headers to the underside of the board. For this build, you only need to install headers for the 21 pins listed below, but you if you’d prefer to cover all 40 there’s room for that as well. The poster tack trick shared for the controller board assembly will help with the soldering here as well.



Pin Cluster	Length	Pi Pico Pins
Power Input	2	VSYS, GND
Power Output	1	3V3
LCD & Restart	8	RUN, GP22, GND, GP21, GP20, GP19, GP18, GND
Sound & Menu	3	GND, GP6, GP7
Controller	7	GP10, GP11, GP12, GP13, GND, GP14, GP15

Building the Power Circuitry

The last step before assembling the back case is to build the power circuitry. As was discussed earlier, you have a couple of options here. The option that provides the greatest convenience is a TP4056/TP4057 USB-C charger attached to a small Lithium Polymer (LiPo) battery. If you’re more concerned with safety or if

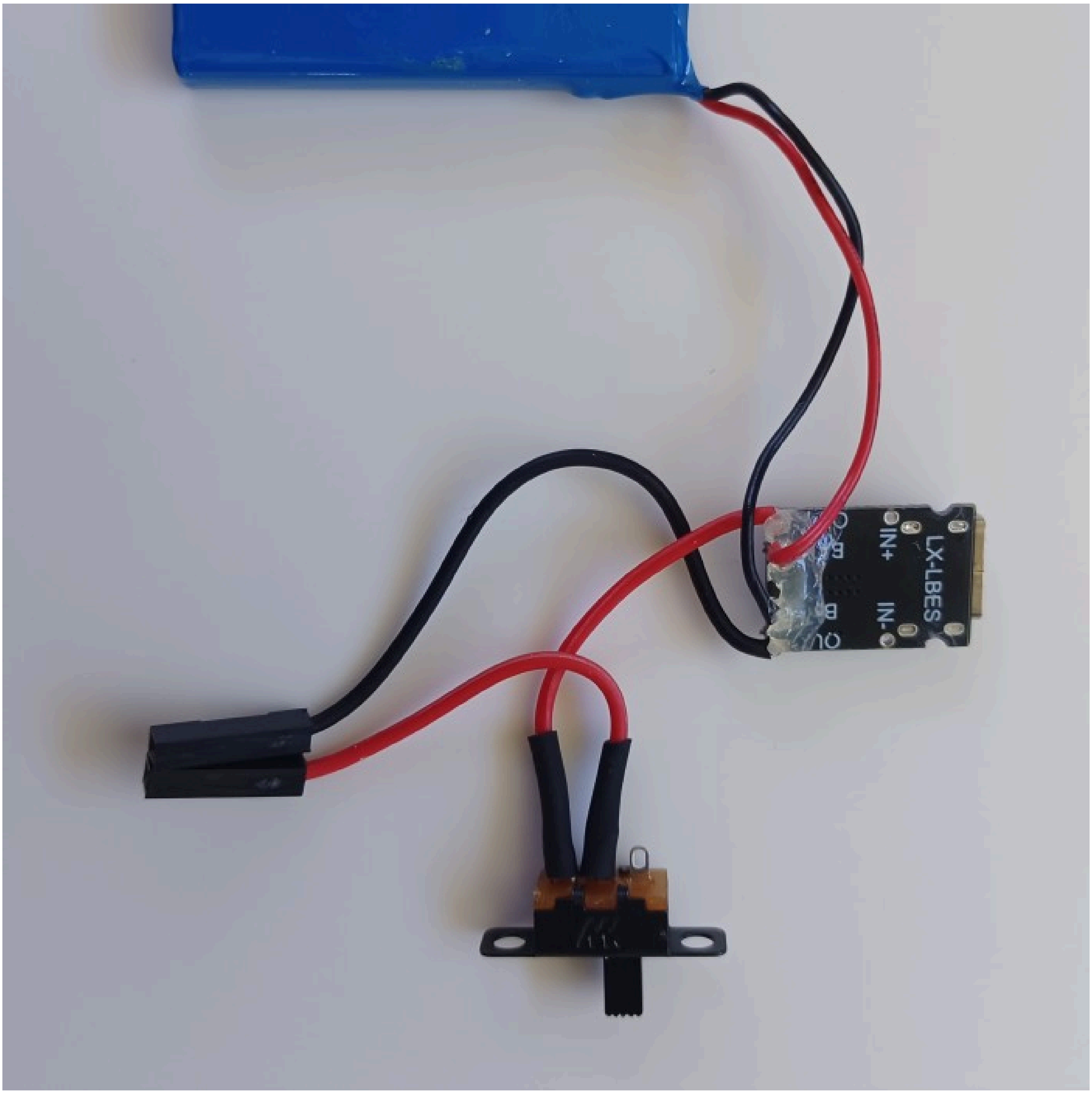
LiPo batteries are not readily available in your area, it's also possible to power the Maketato with a couple of AA batteries.

Option 1: LiPo Battery and USB-C Charger

First, let's prep the wires and connectors we'll need to attach the Pi Pico and charger PCB to the Pi Pico. Find a red female-female Dupont jumper and cut off one end about 4cm from the connector. Cut off another 4cm segment of wire and strip both its ends. Cut a 1cm long piece of heatshrink tubing and, slide it over the wire and then solder one end to the rightmost tab of the slide switch and the other through the 'OUT+' hole on the USB-C charger board. Cut another 1cm segment of heatshrink tubing and slide it over the cut end of the 4cm wire with Dupont connector. Now, strip that end and solder it to the center tab of the slide switch. Slide the both heatshrink segments over the soldered tabs of the slide switch and toast them like marshmallows over the tip of the soldering iron until they're tight over the tabs and wire. Next, cut the end off of a black Dupont jumper wire 5.5cm from the connector. Strip the cut end and solder it through the 'OUT-' hole in the charger board.

Finally, we'll solder the battery wires to the charger. You can either solder the leads of the battery directly to the charger or wire up a suitable connector. If your LiPo battery already has a male JST connector, I recommend the latter as it's the safer and simpler route. If you choose to attach the battery directly to the board, ensure that you do not accidentally short the leads during soldering. Shorting the battery will damage it and may explode - an entirely different sort of STEM lesson and one that you and your kids should avoid.

Once you've decided on your approach, solder the red wire from either the battery or the JST connector through the 'BAT+' hole of the charger and the black wire through 'BAT-'. To prevent shorts during during later assembly and use, I recommend smearing hot glue over the exposed solder points on the board.



You can now secure the USB-C charger board with two 4mm M2 screws and the LiPo battery with double-sided tape. Drop the slide switch into its cutout near the top-left corner of the back case with the unused tab on the left side.

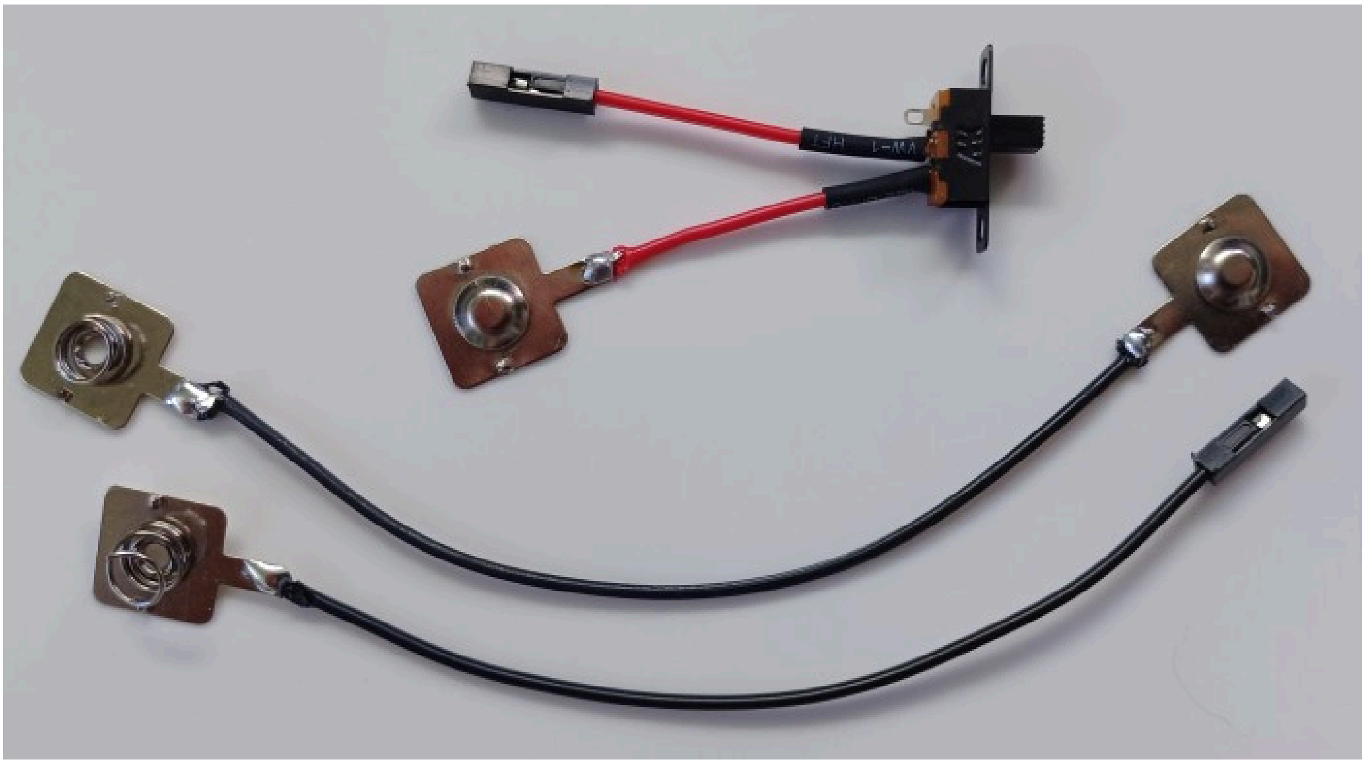
Option 2: AA Batteries

A safer, if less convenient, power option is to use a couple of AA batteries, either standard alkaline ones or NiMH rechargeables. This will require you to print out the AA variant of the back case, install AA battery contacts and solder them to the slide switch.

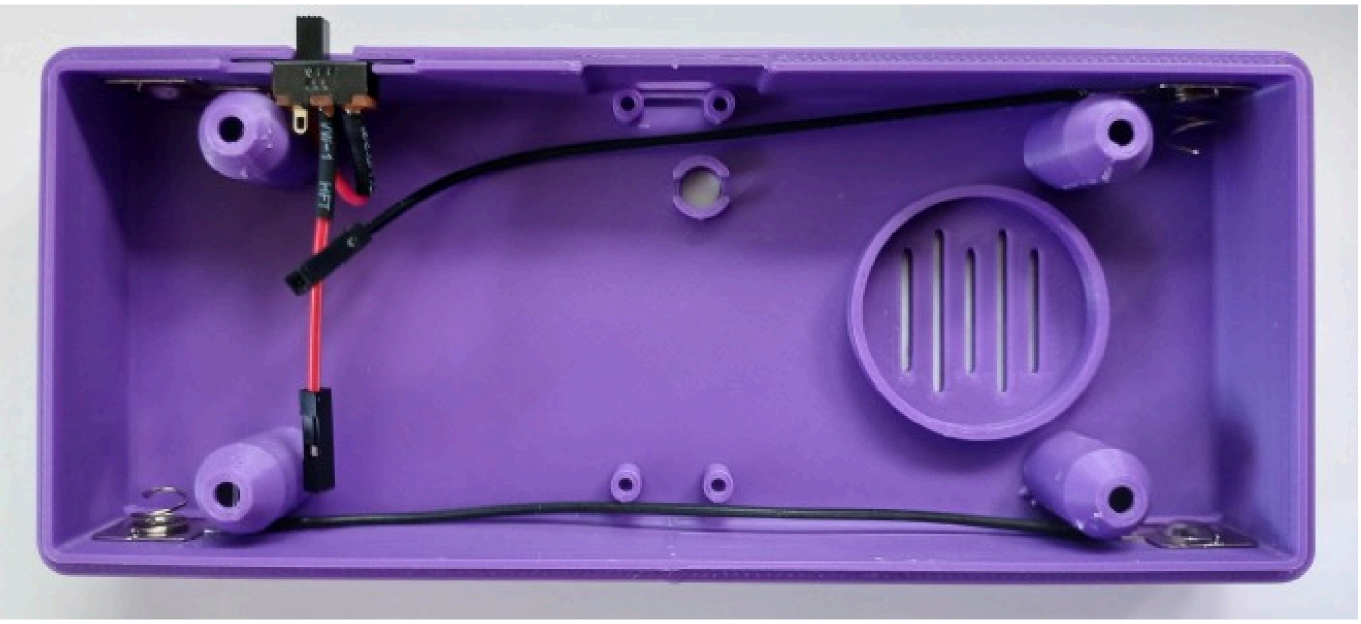
Find a red female-female Dupont jumper and cut off one end about 4cm from the connector. Cut off another 4cm segment of wire and strip both its ends. Cut a 1cm long piece of heatshrink tubing and, slide it over the wire and then solder one end to the rightmost tab of the slide switch and the other through the hole in one of the positive battery terminals (the ones with a bump, not a spring).

Cut another 1cm segment of heatshrink tubing and slide it over the cut end of the 4cm wire with Dupont connector. Now strip that end and solder it to the center tab of the slide switch. Slide the both heatshrink segments over the soldered tabs of the slide switch and toast them like marshmallows over the tip of the soldering iron until they're tight over the tabs and wire.

Take the two negative battery terminals and, using the flush snips, cut their springs down to a 5mm height. Find a spare 15cm or longer Dupont jumper wire (or other stranded wire) and cut off both ends, leaving a 13cm segment. Strip both ends and solder one to the remaining positive contact and the other to one of the negative terminals. Find a black female-female Dupont jumper and cut it 10cm from a connector. Strip the cut end and solder it to the remaining negative terminal.



Now let's get the power circuitry into the back case. If your battery contacts have little locking tabs, pinch them flat with pliers before proceeding. First, drop the power switch into its cutout in the top-right corner of the case, then slide the attached positive terminal the adjacent groove. Next, take the two terminals connected by a 13mm wire and slide the negative one into the bottom-left groove of the case and the positive one into the bottom-right one, routing the along the bottom of the case, behind the screw posts. Finally, slide the other negative terminal into the top-right groove routing the wire around the back of the nearby screw post and between the Bootsel button cutout and upper mounting posts for the Pi Pico.



Insert one AA battery into each side of the case, making sure the springs are in contact with the cathodes (or ‘-’ ends) of the batteries.

Installing the Pi Pico

With the power circuitry installed, we’re nearly ready to drop in the brains of our project - the Pi Pico itself! Pop the 3D printed Bootsel button into the cutout at the top-center of the back case and, if you’re building the AA-powered variant, make sure the black wire from the battery terminal is routed between it and the upper Pi Pico mounts. Now lay the Pi Pico upside-down (with pin names showing) into the back case, sliding its Micro USB connector into the hole at the top. Secure the Pi Pico with four 4mm M2 screws.

The back case is now fully assembled. Let’s wire everything up and start playing some homemade games!

Step 4: Wiring it Up

Grab all of your 15mm female-female Dupont jumpers and let’s wire this baby up. With the Pi Pico installed upside down each pin’s label is visible, making this a pretty straightforward affair. Work through the tables below a row at a time, connecting one end of a jumper to a header on the Pi and the other to the corresponding header on another board.

Speaker

Speaker	Pi Pico
Red (+)	GP7

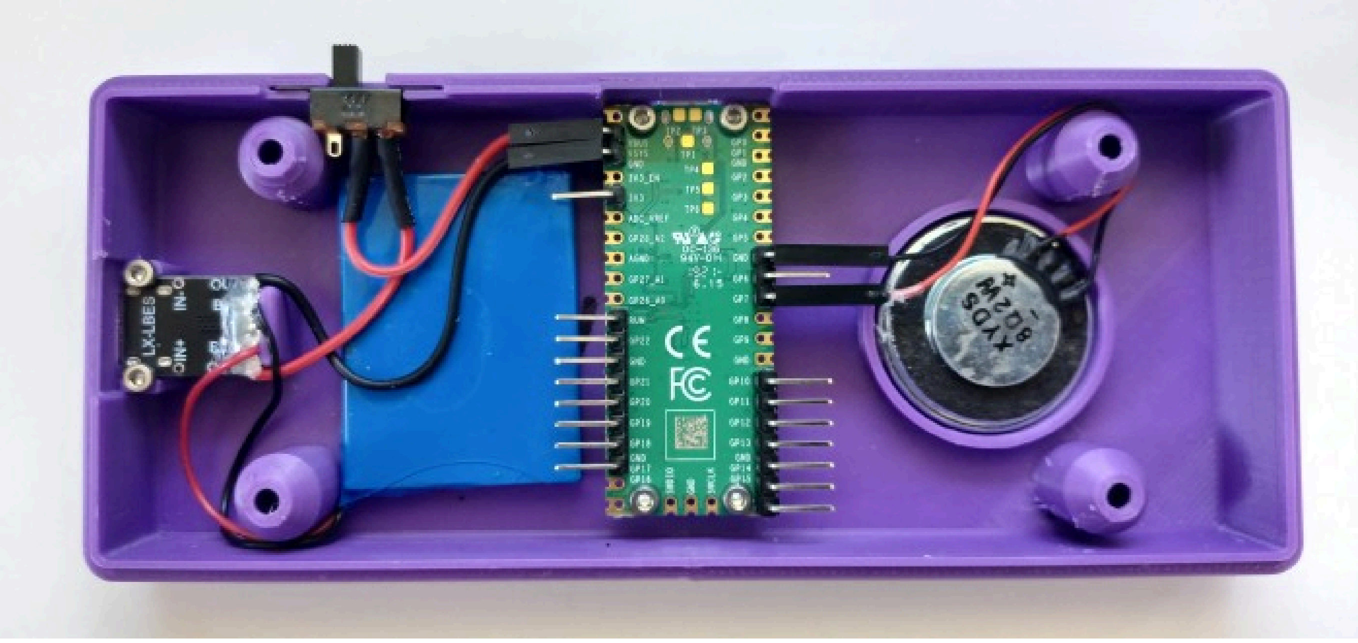
Speaker	Pi Pico
Black (-)	GND (8)

Power Option 1: LiPo Battery & Charger

Charger / Switch	Pi Pico
Switch: Center Tab	VSYS
Charger: OUT-	GND (38)

Power Option 2: 2x AA Batteries

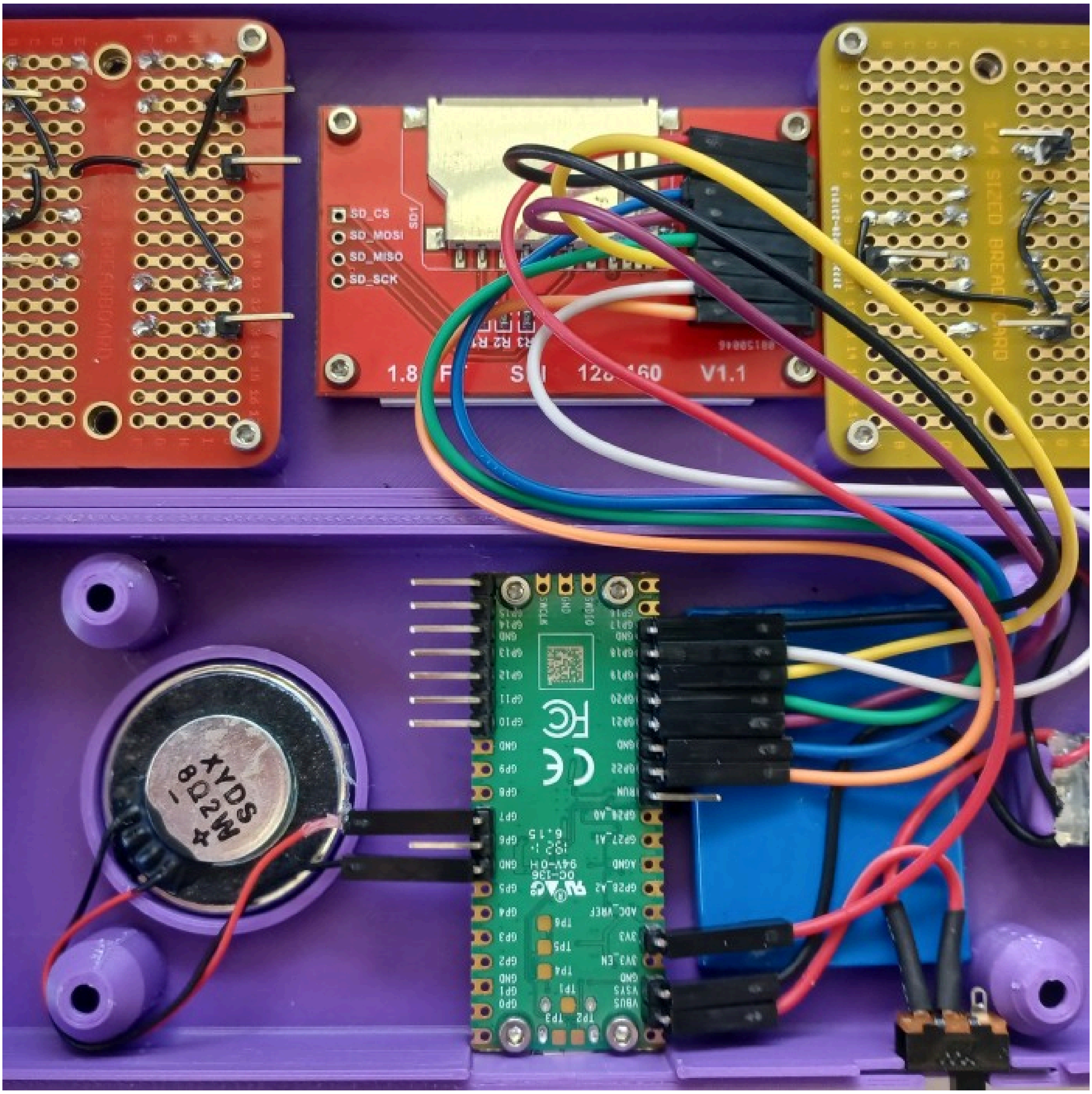
Batteries / Switch	Pi Pico
Switch: Center Tab	VSYS
Right AA Neg. Terminal	GND (38)



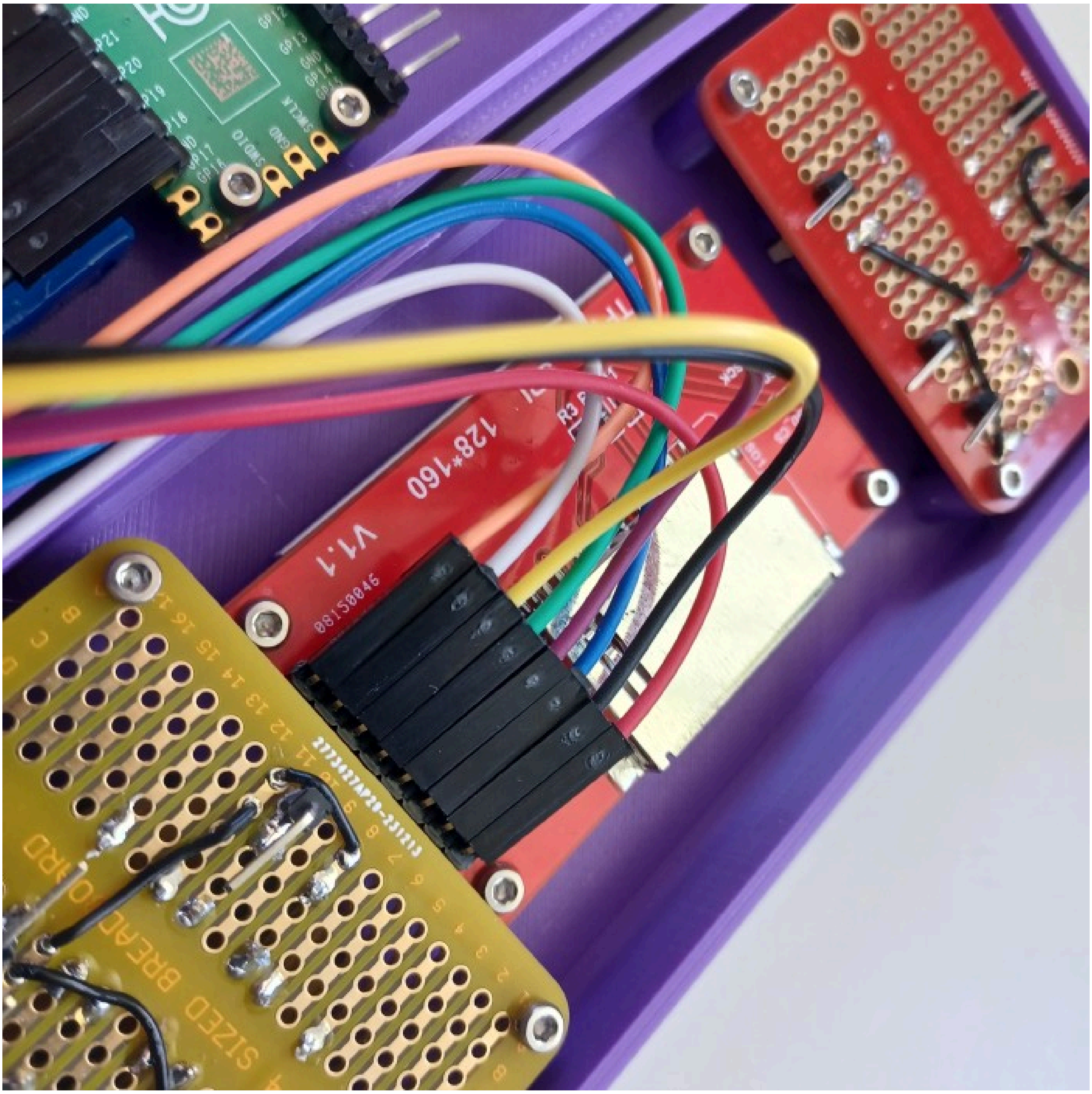
TFT LCD Display

TFT LCD	Pi Pico
VCC	3V3
GND	GND (28)
CS	GND (23)
RESET	GP21
AO	GP20
SDA	GP19
SCK	GP18
LED	GP22 or 3V3

Note: Typically, GP22 would be connected to additional components, allowing you to adjust the screen backlight from MakeCode Arcade’s menu. Connecting the LCD board’s LED pin directly to GP22 without this circuitry results in medium, non-adjustable screen brightness. For a brighter screen, you can instead connect LED to 3V3 (e.g. using a splice).



After connecting jumper wires to all the LCD headers, carefully bend the pins over toward the center until they’re at a 20-30 degree angle to the board. This will make it easier to close the case later.



Controller

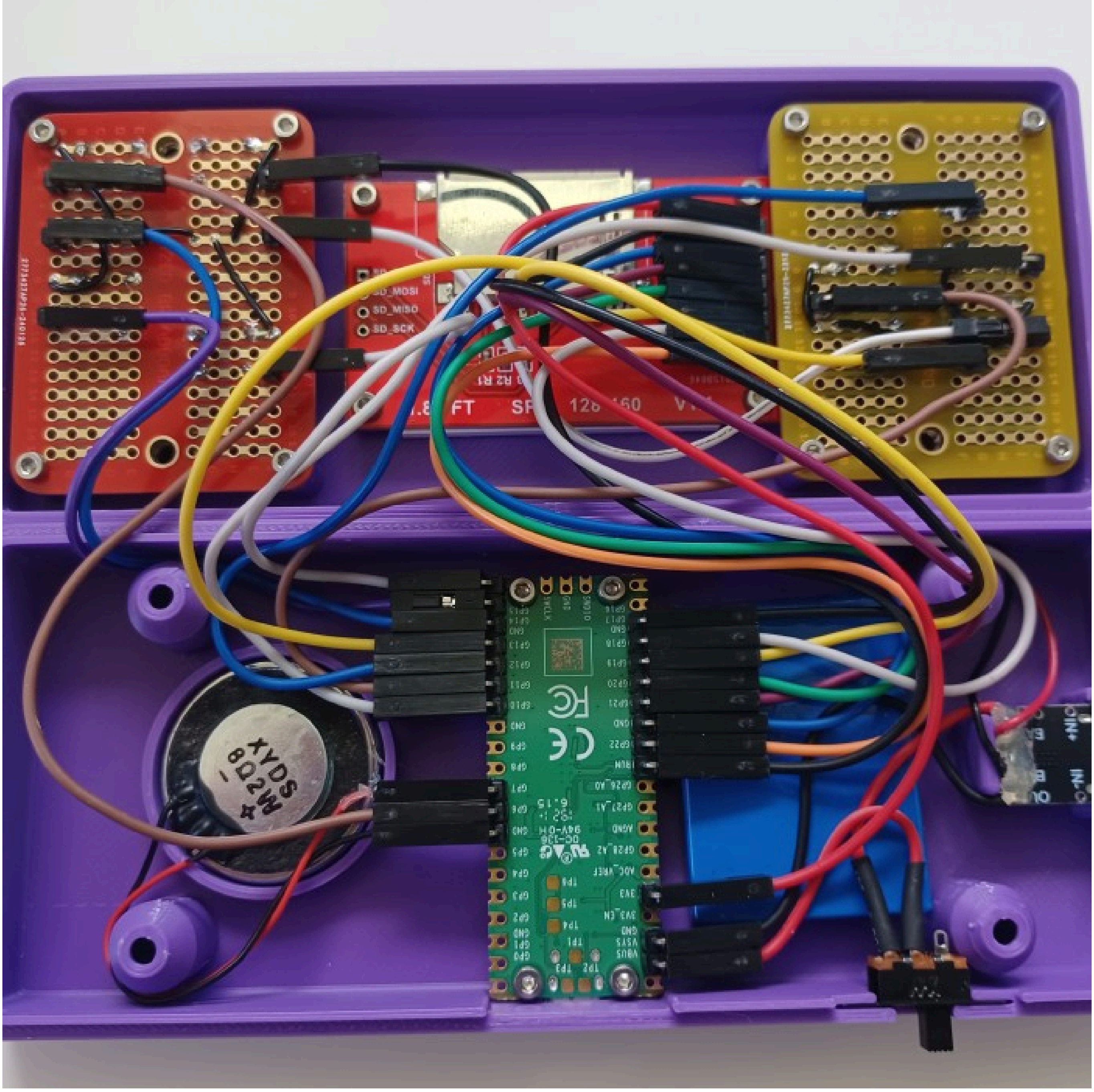
DPad Board	**Pi Pico**
Left	GP10
Right	GP11
Up	GP12
Down	GP13
Ground	Button Board GND or GND

Note: To reduce the number of GND pins used on the Pi Pico, I daisy chain this board’s common Ground to the one on the Button Board.

Button Board	**Pi Pico**
A	GP14

Button Board	Pi Pico
B	GP15
Menu	GP6
Restart	RUN
Ground	GND (18)

That’s it for the wiring! Before sealing up the case, I recommend you hop down to the Firmware Setup section below to test everything out.



Sealing the Case

After verifying that your Maketato is fully functional, slowly bring together the front and back cases, ensuring that no wires are pinched by the screw posts or lip of the back case. The cases should snap together fairly easily. If you feel resistance, check the arrangement of the wires before applying pressure again. Once the two halves are joined with no obvious seam, insert the tapered 8mm M3 bolts into the four wells in the back and tighten them to complete the build.

Firmware Setup and Testing

Before we play our own or others' MakeCode Arcade games on the Maketato, we need to flash a custom firmware configuration to the Pi Pico.

Step 1: Download Firmware Files

Because some ST7735 LCDs are configured differently from others, I've made a couple of firmware configurations available.[2] I recommend starting with the first file and, if you notice rows of rainbow-colored pixels onscreen during testing, repeating the steps below with the second file.

Maketato Config V1 with 1x2 Offset (.uf2)

Maketato Config V1 with 0x0 Offset (.uf2)

Step 2: Flashing the Firmware

To flash the Pi Pico's firmware you'll need a USB-A to Micro USB cable. Plug the USB-A side into the computer (PC, Mac, Chromebook, etc.) with the firmware files, leaving the Micro USB side loose for the moment. Check that the power switch on the Maketato is off (i.e. to the left), then press and hold the 3D printed button near the center of the back case. With the button held down, insert the Micro USB cable into the connector at the top of the Maketato.

After a second or so you may release the button. A volume entitled 'RPI-RP2' should appear in your computer's file browser. Drag and drop one of the above firmware files into the root of the 'RPI-RP2' volume. After copying, the Pi Pico will reboot, causing the volume to disappear, then reappear.

Step 3: Testing with a Game

You're now ready to test out your Maketato! Download the 'Jumpy Platformer' file below and copy it to the Pi Pico by dragging it to the 'RPI-RP2' volume like before. The device will again reboot and this time, should be running a Mario-esque platformer. If that doesn't happen or the game is unplayable, try the troubleshooting steps below.

Jumpy Platformer (From MakeCode Arcade Tutorials) (.uf2)

Step 4: Troubleshooting

Things rarely go perfectly on a first attempt. Don't fret - it's probably just a loose wire or a couple of swapped ones. Here are some suggestions to jump start your troubleshooting:

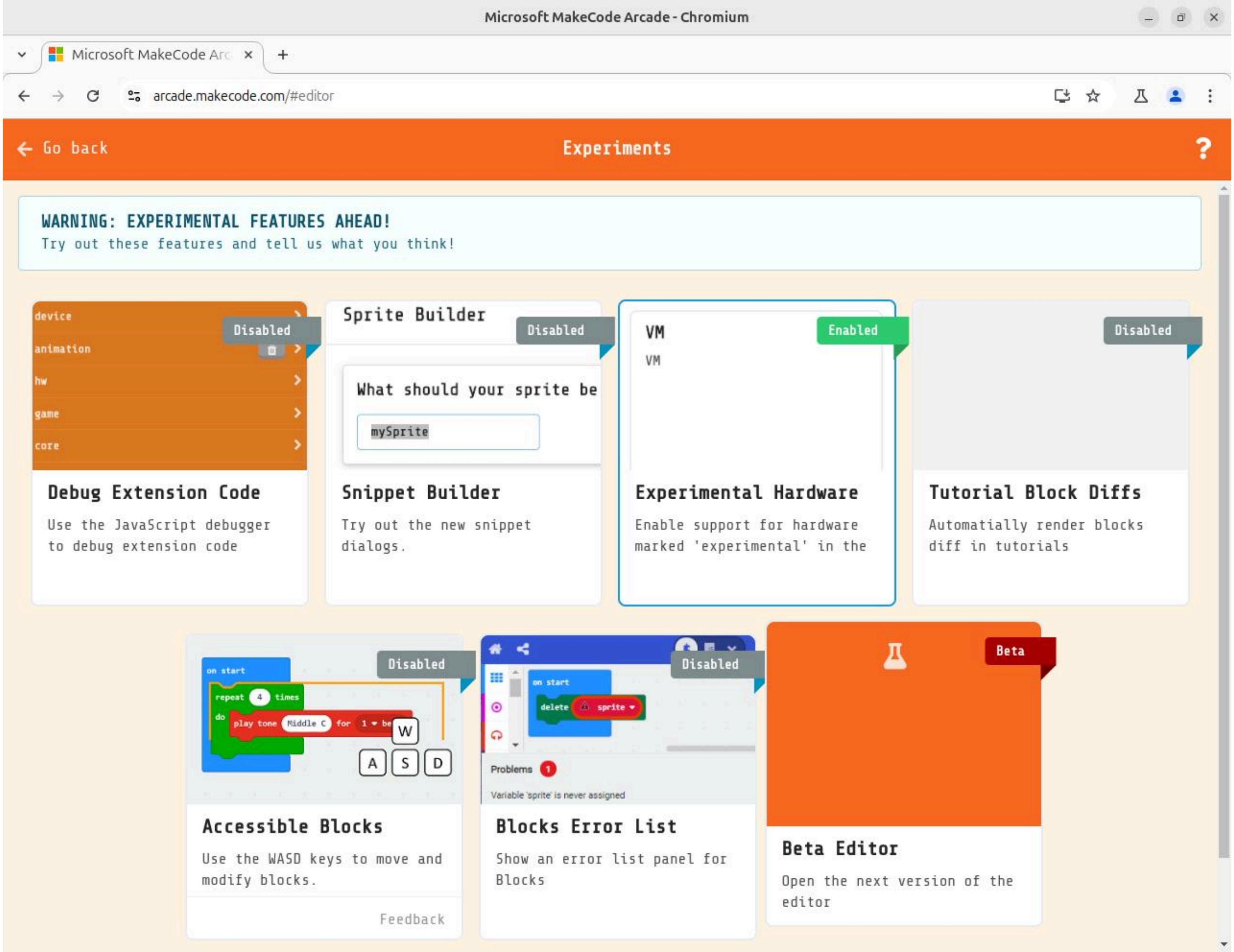
Issue	**Things to Try**
Screen is black	Check power and LCD wiring, flash firmware
Screen is white	Check LCD wiring, flash firmware
Image appears on screen but has rainbow lines on edges	Flash alternate firmware configuration
Image appears on screen but is cut off	Flash alternate firmware configuration
Controls don't work	Check controller wiring
No sound	Check speaker wiring, press Menu button to access volume controls

Coding and Running MakeCode Games

Microsoft’s MakeCode Arcade is a web-based game development environment. On your first visit, I suggest making an account to enable saving projects to the cloud and then diving into one of the tutorials under ‘Beginner Skillmaps’.

Using the Maketato with MakeCode Arcade

To play MakeCode Arcade games on the Maketato we’ll first need to enable support for ‘Experimental Hardware’ by clicking the gear icon in the top-right corner of the page, selecting ‘About…’ and clicking the ‘Experiments’ button in the pop-up. Click on the ‘Experimental Hardware’ option and then return to the Project view.


Microsoft MakeCode Arcade - Chromium
Microsoft MakeCode Arc
arcade.makecode.com/#editor
Go back
Experiments

WARNING: EXPERIMENTAL FEATURES AHEAD!
Try out these features and tell us what you think!

Disabled
Debug Extension Code
Use the JavaScript debugger to debug extension code

Sprite Builder
What should your sprite be
mySprite
Disabled
Snippet Builder
Try out the new snippet dialogs.

VM
Enabled
Experimental Hardware
Enable support for hardware marked 'experimental' in the

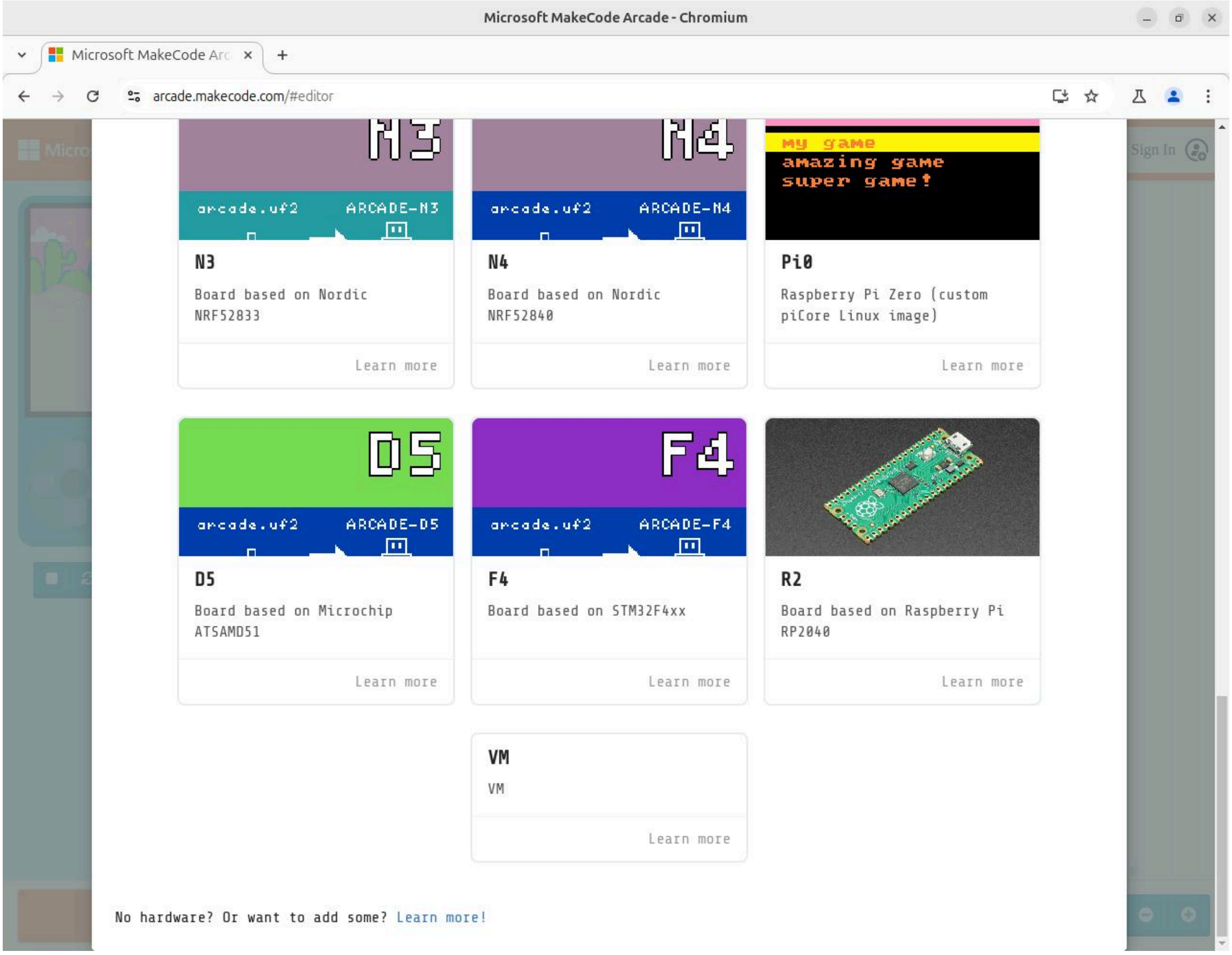
Disabled
Tutorial Block Diffs
Automatially render blocks diff in tutorials

Disabled
Accessible Blocks
Use the WASD keys to move and modify blocks.
Feedback

Disabled
Blocks Error List
Show an error list panel for Blocks

Beta
Beta Editor
Open the next version of the editor

Back in the Project, click on the ‘…’ icon by the ‘Download’ button at bottom-left and select ‘Choose Hardware’. Select the ‘R2 - Board based on Raspberry Pi RP2040’ option near the bottom. Now, when you click the big orange ‘Download’ button you’ll download the game as a .uf2 playable on your Maketato.



N3

Board based on Nordic NRF52833

Learn more

N4

Board based on Nordic NRF52840

Learn more

Pi0

Raspberry Pi Zero (custom piCore Linux image)

Learn more

D5

Board based on Microchip ATSAMD51

Learn more

F4

Board based on STM32F4xx

Learn more

R2

Board based on Raspberry Pi RP2040

Learn more

VM

VM

Learn more

No hardware? Or want to add some? Learn more!

To load a game on your device, first switch off the Maketato, then connect a USB-A to Micro USB cable to your computer. While holding down the ‘Bootsel’ button on the back of the Maketato case, plug the cable into the Micro USB connector on top. The Maketato will turn on and an external drive named ‘RPI-RP2’ will appear on your computer. Drag the previously download .uf2 file to the ‘RPI-RP2’ drive to install the game, which will automically begin playing. You can then disconnect the Micro USB cable and flip the Maketato’s power switch to continue playing.

---

1. The Maketato isn’t a Microsoft-certified MakeCode Arcade device and support for the RP2040 microcontroller on which it’s based remains experimental. Not all MakeCode features are currently supported, most notably JACK-based multiplayer. ↩

2. Some ST7735-based LCD modules map the first physical pixel to the logical point (0,0) while others map it to (1,2). To ensure MakeCode displays a complete image, we need to account for this difference in the configuration file. ↩