BUILD GUIDE

This document is to aid in the construction of your very own Teensy-based wind instrument. There's a lot of small notes to make that would be hard to include in a video so I'm putting it all here. **NOTE: this is an intermediate project. If you want to build one of these, you should have some experience in building Arduino circuits and have a general idea of how the components work. I do this stuff for fun and I'm providing these files for free for no other reason than that I want to. Making time to help people troubleshoot can be difficult. Always approach advice from the internet with a healthy dose of skepticism and do your own research before beginning any DIY projects.**

Printed stuff

You'll need to print one of everything, to include both halves of the case and the mouthpiece. I printed these in PLA with 3 perimeters, 4 shells, 10% infill. You won't need any support material as long as you print the mouthpiece standing up and the case halves with the solid back on the build plate.

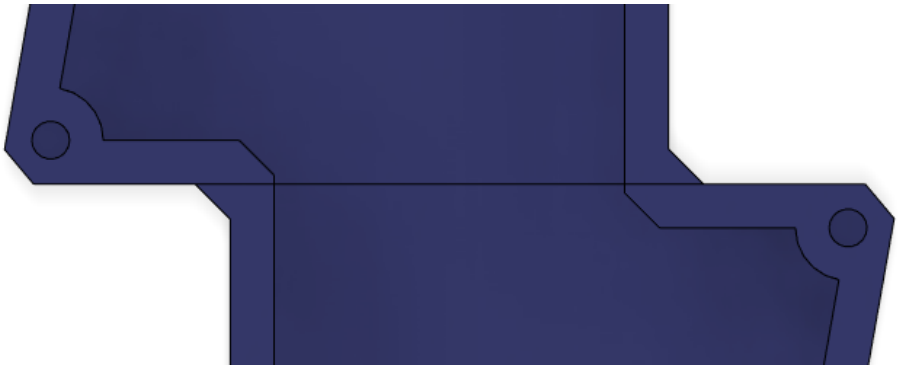
These are all of the non-printed parts that you'll need to buy:

- Teensy 3.2
- Teensy audio shield (this can be omitted if only mono out is needed but the code will need some small modifications)
- MPR121 breakout board (https://learn.adafruit.com/adafruit-mpr121-12-key-capacitive-touch-sensor-breakout-tutorial)
- 7805 5V regulator (https://components101.com/ics/7805-voltage-regulator-ic-pinout-datasheet)
- MPXV4006GC7U, MPXV4006GC6U/C6T1 pressure sensors (choose one)
- An analog joystick. I took mine from an xbox 360 controller
- Some capacitors (100u & 10u electro; 470p & 2x 100n ceramic)
- 4148 diode
- 8x M3 nuts, 4x M3x8 screws, and 4x 15mm pcb standoffs (for mounting the joystick) I got all of these in a kit from eBay
- 2x M3ishx10mm self tapping screws for mounting the mouthpiece. I used some salvaged screws, find something that fits
- 6x M3x10 or #4-40x5/16" machine screws and matching heat set inserts
- Power switch, SPDT sub mini toggle
- ¼" mono audio jack
- ⅛" stereo audio jack
- 1ft of 7/64" ID (3mm), ¼" OD (6mm) vacuum tube. Only about 2" (50mm) will be used but it helps to have extra if you drill too many holes
- Some hookup wires
- Some kinda stripboard, dotboard, or protoboard
- Pin headers (both male & female)
- 9V battery clip

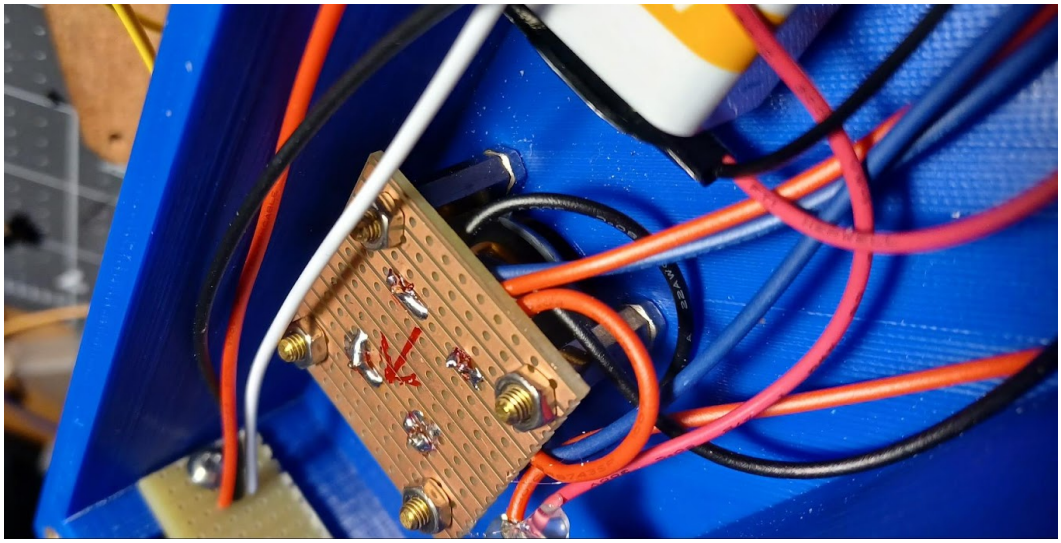
- 12 of something conductive. 3M sells copper tape which works, I had some custom PCB's made which turned out to be kind of a pain. You could also try hardware of some kind, I've used bolts before which worked okay. This part may need some experimentation.

BUILDING THE CASE

Print out the case parts and prepare to glue them together. There are special glues for PLA but I've found that regular ol epoxy does the job just fine. Before gluing, sand the parts at the glue joint to ensure good adhesion. For reference, the parts should line up as below:

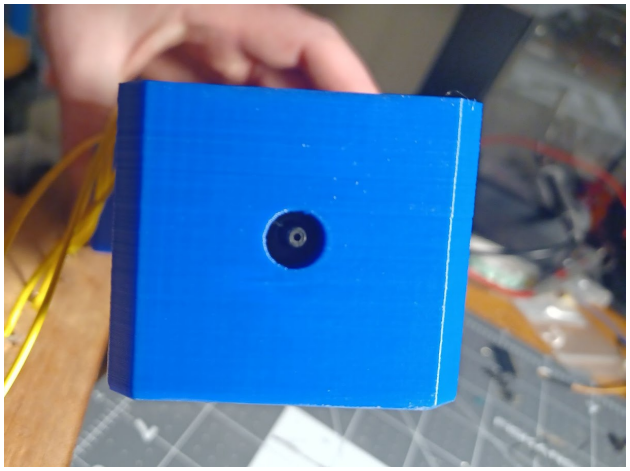


Also, make sure that the bottom of both parts are well aligned for glue up. I used masking tape to help keep them in place and left them on a known flat surface clamped overnight. After allowing time for the glue to dry, clean up any squeeze out and apply the heat set inserts with a soldering iron. At this time I also selected the screws used to mount the joystick and pressure sensor. The joystick needs about 18mm of standoff inside the case to get the fit right, so I used a 15mm pcb standoff stacked on a nut. The M3 screw goes through the case, held in place with an M3 nut. You should have enough threads left over to thread on the pcb standoff (male/female type), and the joystick sits on top secured with another nut.



The mounting of the pressure sensor can be a bit tricky due to the tight spacing, I managed to get in there with a long screwdriver with a magnet on the shank. Screwing into 3D printed parts makes them prone to splitting along the layer lines so I heated up the screws with a lighter before installing to set the threads. The timing for this is tricky, you want the screw to soften the plastic but still be solid enough to set threads instead of squishing. Just a quick swipe back and forth with the lighter should be enough.

Next, print out the templates for the pcb's and the faceplate. The pcb's can be made from any kind of soldered breadboard, but if you're using a board with connections (like stripboard) make sure that the traces are cut before soldering the pressure sensor or joystick down. Attach the templates to the breadboard of choice and cut out the pcb's to the lines and drill the holes as marked. You'll also need to drill some smaller holes for the mounting tabs of the joystick but those will need to be marked by eye. Just do your best to get it centered. Depending on which pressure sensor you were able to get it may be SMD or thru hole mounted, this will change the direction that the breadboard will have to face when it's inside the case. Either one will work just fine, just be careful to align the sensor on the board so that it is somewhat centered when looking into the case. This will help you get the tubing on later.



For the faceplate, choose a material that is around ⅛" (3mm) thick and large enough to fit the final part. Attach the printed template to the part and cut around the lines and drill the holes. It is helpful to leave a small margin of error around the perimeter to save for mounting on the case, then trace the outline of the case onto the faceplate to get a good fit. I used a 4mm drill bit for the 6 mounting holes and a 6mm bit for the 12 touch plate holes. If you plan on making the faceplate out of wood, be sure to apply finish after achieving the final shape but before applying the touch plates.

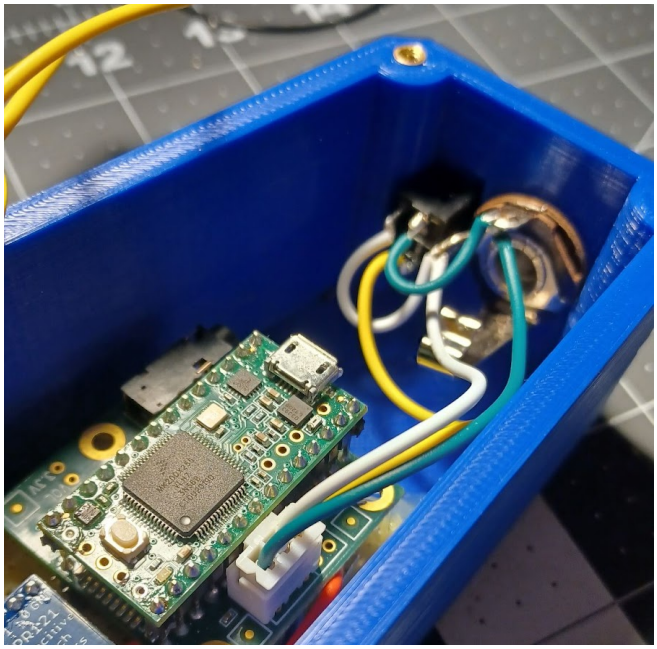
ELECTRONICS

Prepare your breadboard of choice as needed, when using stripboard it is helpful to place all parts on it without soldering, and mark all of the traces that need to be cut. Traces can be cut easily with a drill bit spun by hand or VERY CAREFULLY with a drill. Follow the schematic for placing of parts, it's a nice touch to socket the MPR121 and Teensy in case you need to replace or reuse them later. There are 3 power rails for this project, as follows:

- 9V rail (Source: 9V battery): from 9V+ battery terminal to switch, switch to 7805 IN pin
- 5V rail (Source: 7805 output): 7805 OUT pin to Teensy VIN pin & pressure sensor VIN pin
- 3.3V rail (Source: Teensy 3.3V pin): Teensy 3.3V pin to MPR121 VCC pin & joystick V+ pins. Also the 4148 diode connects between the ADC pin of the pressure sensor to 3.3V to limit the sensor output.

NOTE: before soldering the Teensy, cut the VCC trace. If this trace is not cut then the Teensy cannot be plugged in to USB when installed in this project. If the trace is cut, then the Teensy must receive power from another source (i.e. the project) to turn on. https://www.pjrc.com/teensy/external_power.html

Next, solder your components to the prepared breadboard. Be careful to note the correct direction of all components, especially the 5V regulator. There are lots of different versions and some of them have different pinouts, check the datasheet for yours before installing. When using the Audio Shield with Teensy, it should be soldered on top of the Teensy with double pin headers. PJRC has the option to buy the pin headers with the Teensy and Shield, this is the easiest way to get them. I also recommend using pin headers to connect the audio shield outputs to the jacks, this makes the Teensy fully removable when also using a socket to connect to the breadboard.



For soldering the hookup wires for all of the off-board components, think through how they will be mounted inside and leave yourself enough slack to get them placed in position. You'll want a little extra length to help get things placed during final assembly.

Soldering the touch plates to the MPR121 pins can be tricky, as they need to be installed on the faceplate first. What I did was solder the hookup wires to each touch plate and glue all of them to the faceplate, then solder each wire to its own pin on the MPR121. If using another method, do a dry run first and think through the steps.

When you believe you are done soldering, take a break. Coming back to the project later will help to highlight any misplacements or soldering issues. Check over all solder joints for weak connections and bridges. All mistakes can be fixed at this point but be sure of your work before powering up. Before powering up any project, I like to check the power connections for shorts with a multimeter in continuity mode. Compare each power rail to the others and to GND, if there is full continuity (the BEEP) between any of these then track down that issue before powering up. Remember that capacitors may cause momentary continuity before charging up.

CODE

After all soldering is complete and the hardware is functioning, upload the code to the Teensy. As long as you're able to find the parts listed at the beginning, your code will require minimal modifications. The most likely thing that will need adjustments is the joystick. Joystick values are all over the place, even on the 2 axes mine were significantly different. The main thing to keep in mind is that the map() values (lines 270, 271, 282, 283) should be close to the highest and lowest range of the joystick axes (pushed all the way in one direction), and the threshold values (lines 105-108) should bound the readout when the joystick snaps to center. If you have issues, enable the serial monitor and read out the joystick values with the Teensy connected to your PC. This can be done by uncommenting the Serial.print lines in the code. Remember that you will need to cut the VCC trace as mentioned above or your Teensy will blow up.

If you choose not to buy the Audio Shield, the code will require modifications to run. Follow these resources to understand what needs changing, and remember that with this modification only mono output is available. The code will also need to be changed in the Audio System Design Tool. As is, the output is sent to I2S1 which is the Audio Shield, this will need to be changed to DAC.

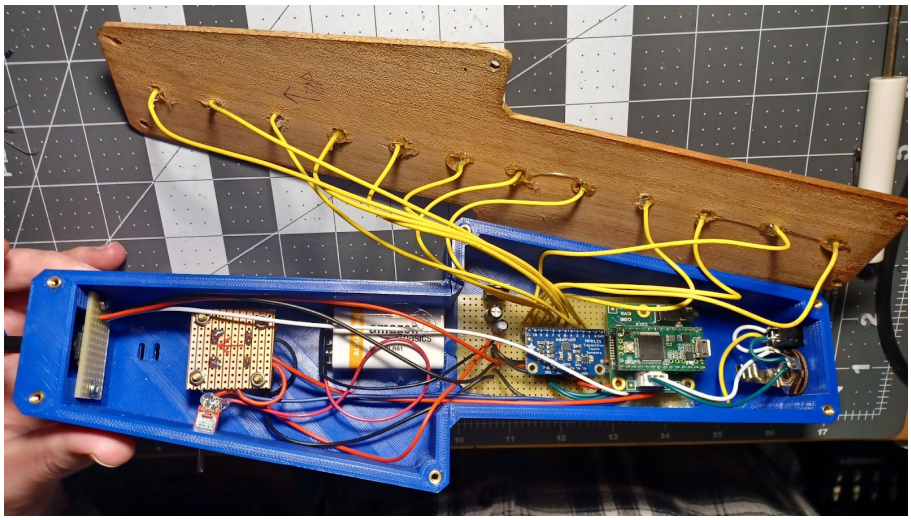
- Notes & Volts: https://www.youtube.com/watch?v=c39e8FvTYRA
- PJRC: https://www.pjrc.com/teensy/teensy31.html

Most of the time the MPR121 sets thresholds that work very well for most conductive surfaces. However, if you find strange things happening with the detection or releasing of the touchplates, you may need to adjust the sensor thresholds by modifying the values in Line 254: cap.setThresholds (touch, release). This page describes some relevant definitions, there's plenty of forum posts about this as well:
https://adafruit.github.io/Adafruit_MPR121/html/class_adafruit___m_p_r121.html

Another code modification is that the I2C address for the MPR121 may need to be changed, depending on what version of sensor you end up with. Have a look at the documentation for the version you have (or the official version for the bootleg one you got on AliExpress). The default address is set in Line 253: “0x5A” but per the datasheet addresses up to D are allowed. Theoretically, you could build this instrument with up to 48 notes if you use all 4 addresses matched to 4 MPR121’s.

If you want to do any other mods to this code, I do think that you should, but my ability to help you is limited. I know just enough about coding to be able to put this kind of stuff together and I’m sure you could write a 30 minute presentation about how much the code could be improved. But I can tell you that it works, which is my only goal. Your best bet is to follow the resources in the .ino file, that’s how I figured out how to put all of this together.

FINAL ASSEMBLY



At this point, you should have all parts printed and joined, the faceplate should be finalized, all parts connected, and you should have verified that the instrument powers on and the inputs are being recognized. If you need to later, there should be enough space in the case to sneak a USB onto the Teensy for reprogramming or debugging.

Place the pressure sensor into the case first and secure it with the 2 screws. You should have a bit of wiggle room to get the sensor port centered in the hole. Next, install the power switch and mind the orientation so you know which way is On / Off. After the power switch, mount the stackup for the joystick: screw goes into the case with a nut on the inside, then the pcb standoff, then place the joystick, then a final set of nuts to secure. Verify that the joystick moves and doesn’t get hung up anywhere, you may need to center it up manually.

Next, fit the audio jacks into place and tighten the nuts to secure them. Place the main board in after and secure it with some hot glue. Leave enough room to fit a USB cable onto the Teensy when it’s in the case, you may want to reprogram it later. Also mind that the 7805 may get warm, so don’t place it directly against the side of the case. Lastly, make sure the power switch is in the Off position, plug in the 9V battery, and hot glue it into the case. I know this isn’t

the neatest solution but it's easy enough to remove once in a while. Feel free to come up with better mounting solutions if you want.

Screw the faceplate onto the case, cut down a small section of vacuum tubing and drill a few small holes in it. I ended up going with 4x 1.6mm holes, but try it out and adjust as you see fit. You should have enough tubing for several attempts. Place the mouthpiece on the other end of the tubing, and if all has gone according to plan, you should now have an assembled instrument. Power it up, connect up some headphones or an amp, blow into the mouthpiece and push some buttons. Flail around wildly as you rejoice in your mastery of the elements, don your wizard hat and coat, make haste for the town square to charm the locals with your whimsical music abilities.

TROUBLESHOOTING

If the locals aren't swooning at your musical abilities, the only possible reason is that the instrument isn't adjusted properly. There's a lot of small parts that can be tuned up and depend on the exact parts installed. Most likely, the issues are related to setting up the joysticks or setting the thresholds on the touch plates. Try to narrow down the problem to one component: if no audio is coming out, is that because the filter isn't opening, the envelopes aren't triggering, or the audio isn't connected? The serial monitor is your friend here, place flags inside your functions that will only trip if the function works, print values out to the serial monitor. Or, if you've made many modifications, try reverting to the original version and only making one change. Be scienticious and you will prevail.