

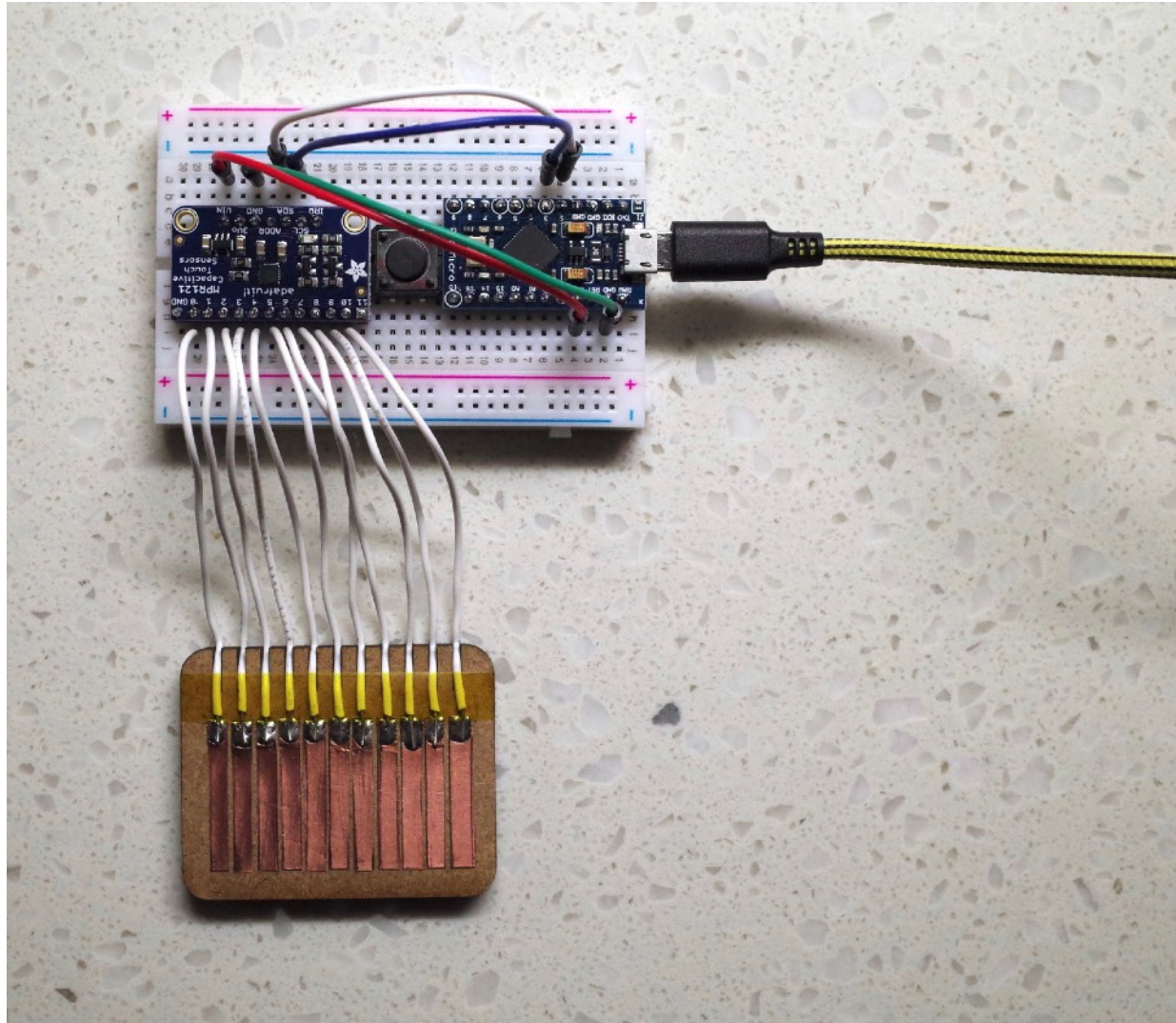
# CONTINUITY

---

*UWTMC*

*Matt Borland - 2019*





## WE'RE EXPLORING CONTINUOUS MUSICAL EXPRESSION

---

*We'll use capacitive touch and some fancy code to explore the basics behind using capacitive touch to measure position.*



# BopPad

Smart Fabric Drum Pad

\$199

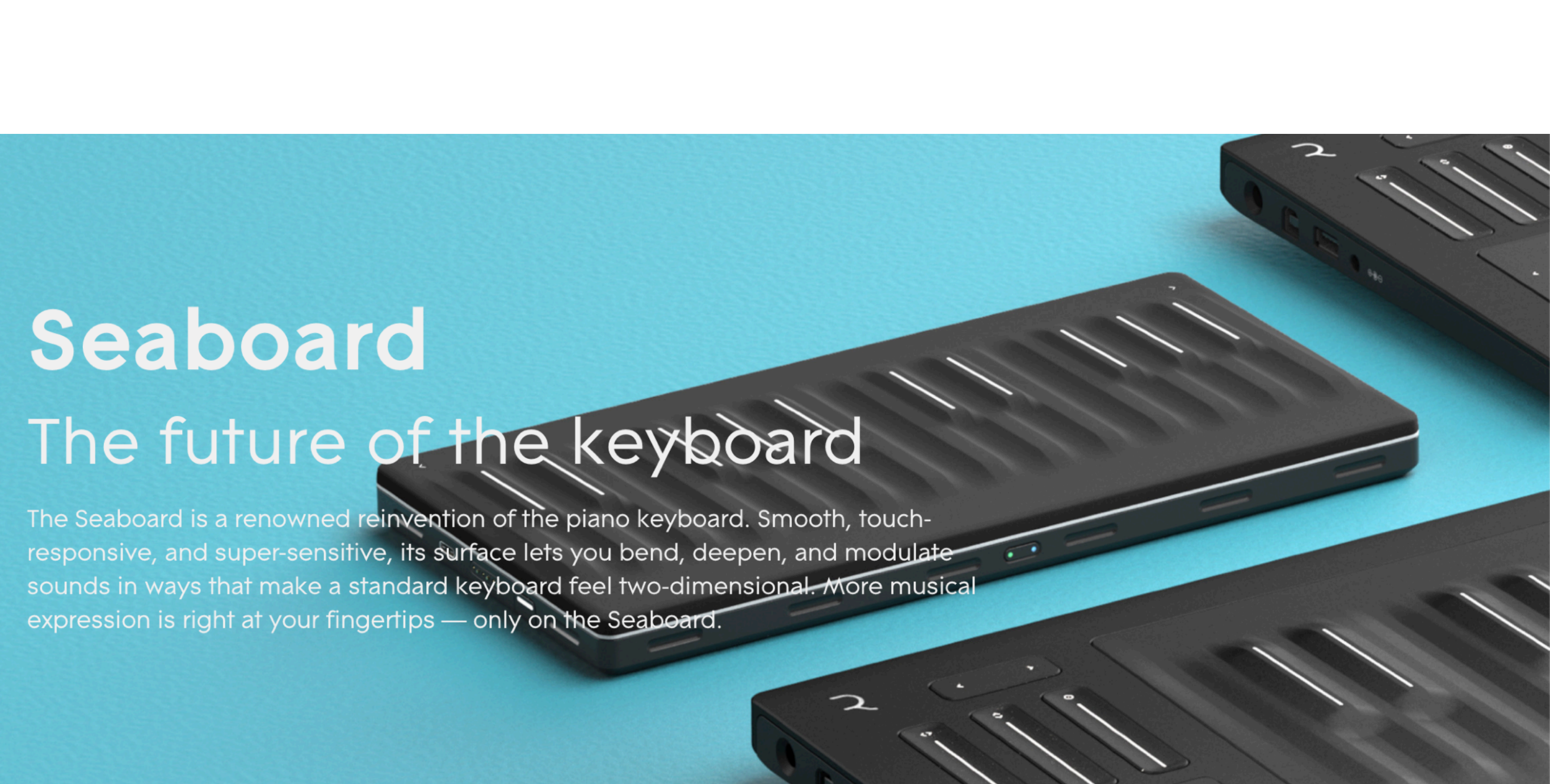
Buy now

<https://www.keithmcmillen.com/products/boppad/>



<https://www.youtube.com/watch?v=FkWHMR3kwoQ>





# Seaboard

## The future of the keyboard

The Seaboard is a renowned reinvention of the piano keyboard. Smooth, touch-responsive, and super-sensitive, its surface lets you bend, deepen, and modulate sounds in ways that make a standard keyboard feel two-dimensional. More musical expression is right at your fingertips — only on the Seaboard.

<https://roli.com/products/seaboard/>





<https://www.youtube.com/watch?v=jh-hzbG5FzI>



# DOWNLOADS

---

*You need two pieces of software. Both are free and multi-platform!*

Download the Arduino IDE



*Arduino IDE - a software platform used to program your microcontroller.*

<https://www.arduino.cc/en/Main/Software>

*Helm - a software synthesizer to make musical sounds with your computer.*

<https://tytel.org/helm/>



# FILES

---



Workshops this term are at CML! All sessions run 3:30-5:30pm at Critical Media Lab, located in Communitech, 151 Charles St. W., Kitchener.

Sept. 18th: Deformation - MIDI Balloons: [DeformationFiles.zip](#)

Oct. 2nd: Gravity - Shruti Box: [GravityFiles.zip](#)

Oct. 23rd: Continuity - Seaboard and Bop Pad: [ContinuityFiles.zip](#)

Nov. 6th: Complexity - Modular Synthesis

Nov. 20th: Exploration - Co-play Patch Tables

Each session will start with an explanation of a musical interface and a discussion of its expressivity, followed by a hands-on build with the sensing technology being explored, then the opportunity to develop improvisational skills through jamming with other participants on the instruments you've just built. Make sure to bring your laptop to be able to program and power your instrument!

*Files are available as a ZIP at <https://uwtmc.com>*



LET'S DO IT!

---

*Matt will explain everything,  
so follow along with him on  
the big screen. Ask questions if  
you're unsure - other people  
are probably in the same boat.*

# HOOKING UP YOUR SENSORS! FOLLOW MB ON THE PROJECTOR

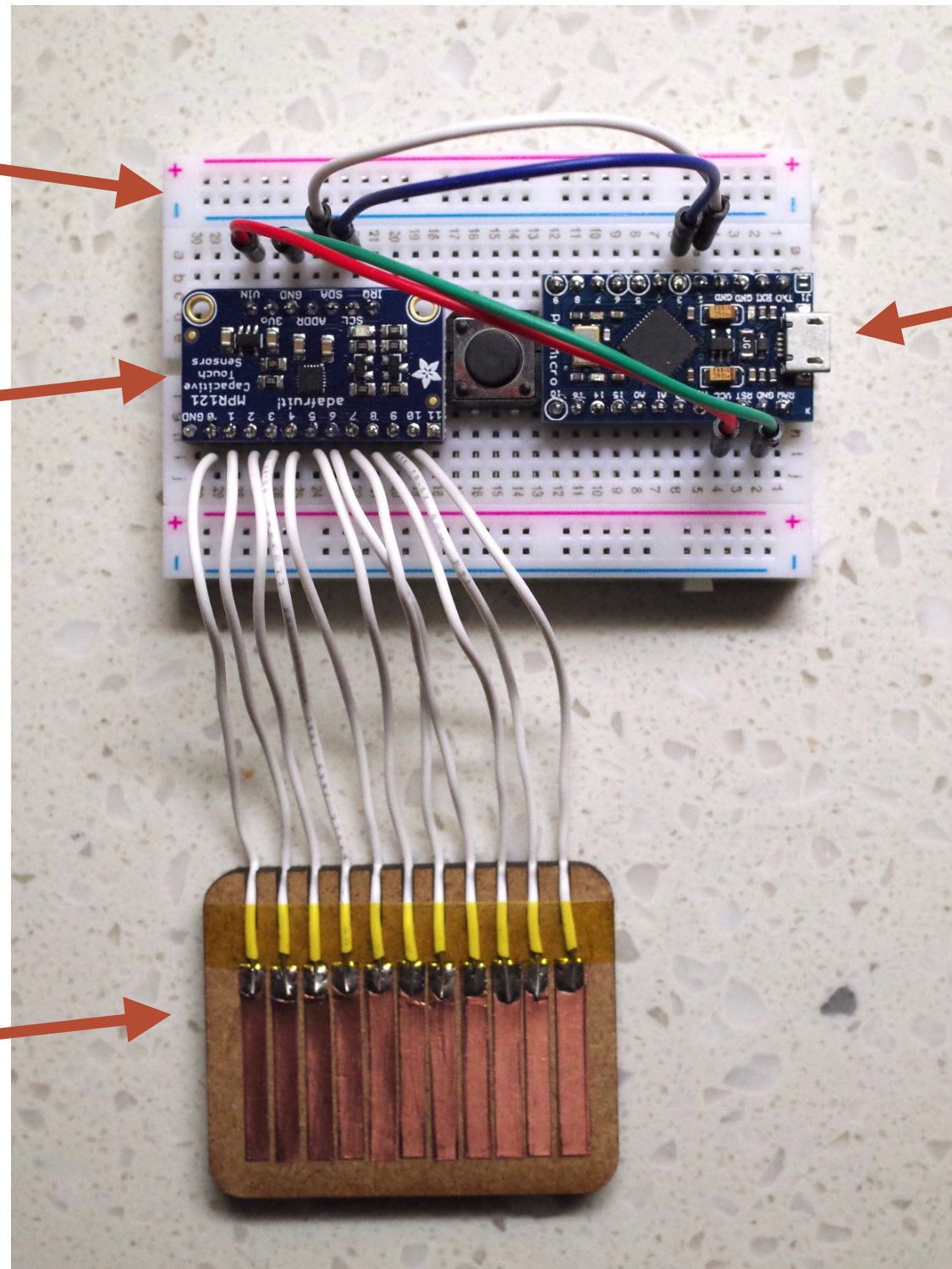
---

Breadboard

Capacitive  
Touch Board

Touch Strip

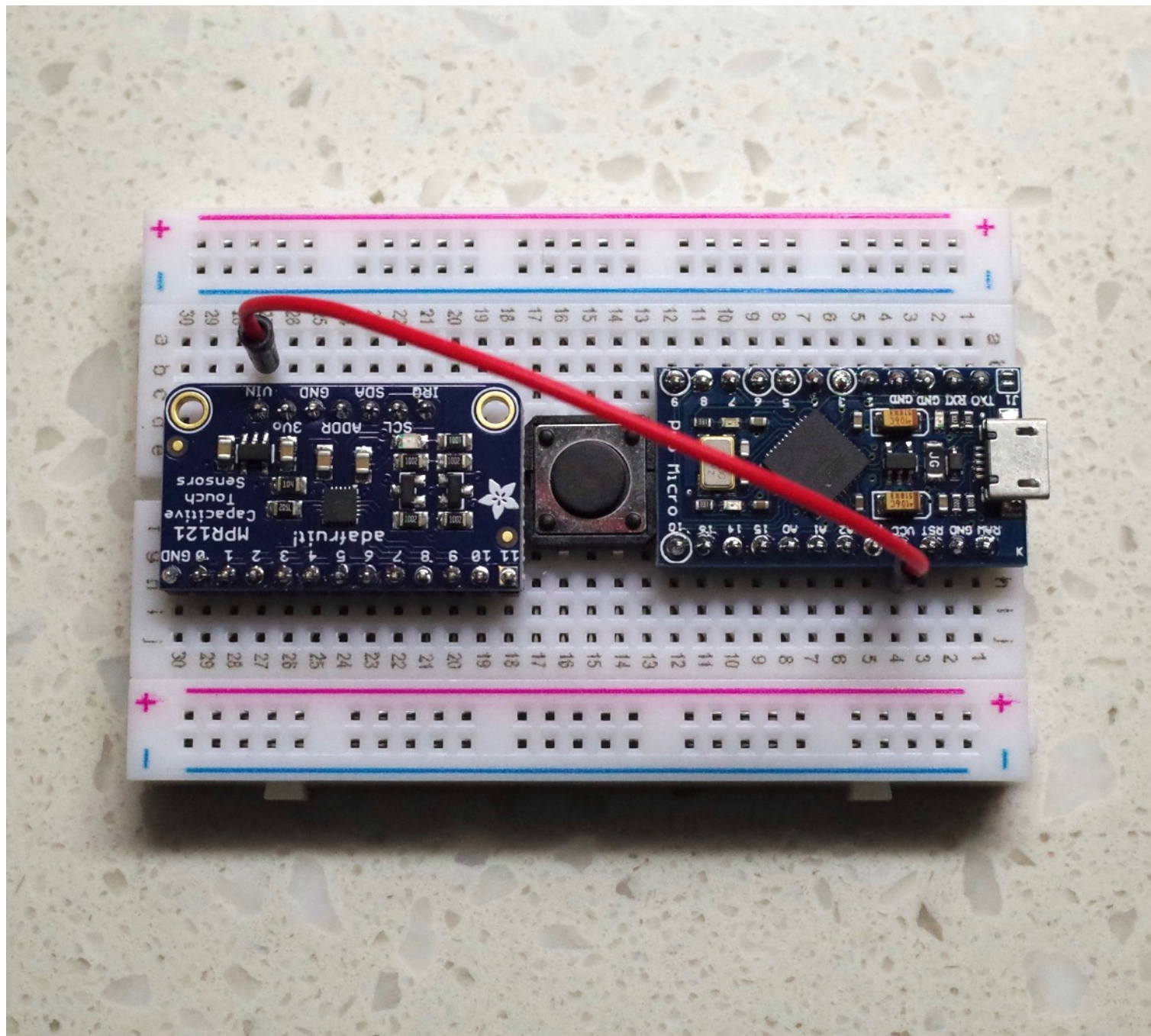
Arduino





# HOOKING UP YOUR SENSORS! FOLLOW MB ON THE PROJECTOR

---



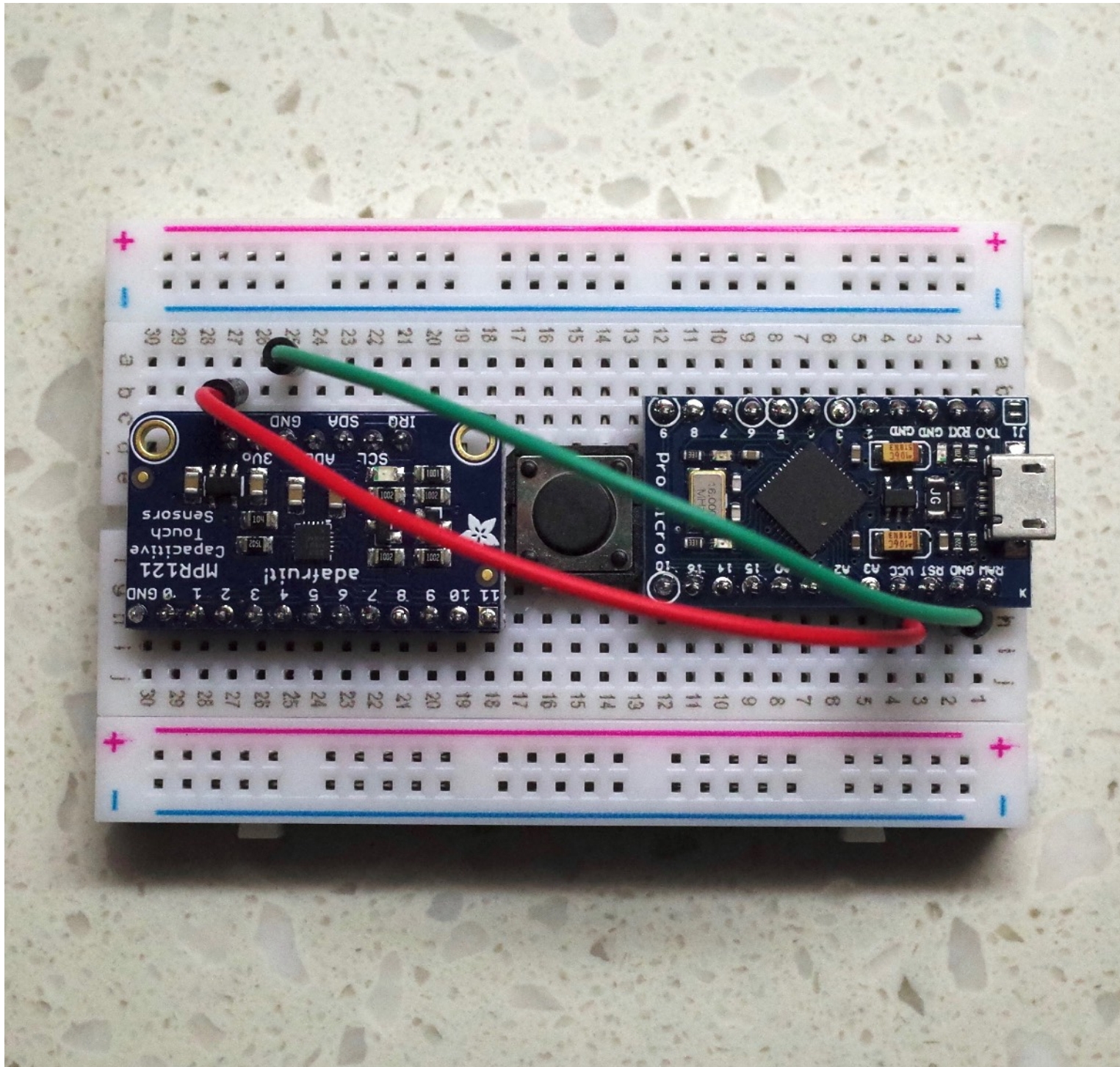
Connect the “VCC” pin on the Arduino board to the “VIN” pin on the Capacitive Touch Board.

**NOTE:** All pin references are to the labels directly on the components, not the breadboard.



# HOOKING UP YOUR SENSORS! FOLLOW MB ON THE PROJECTOR

---

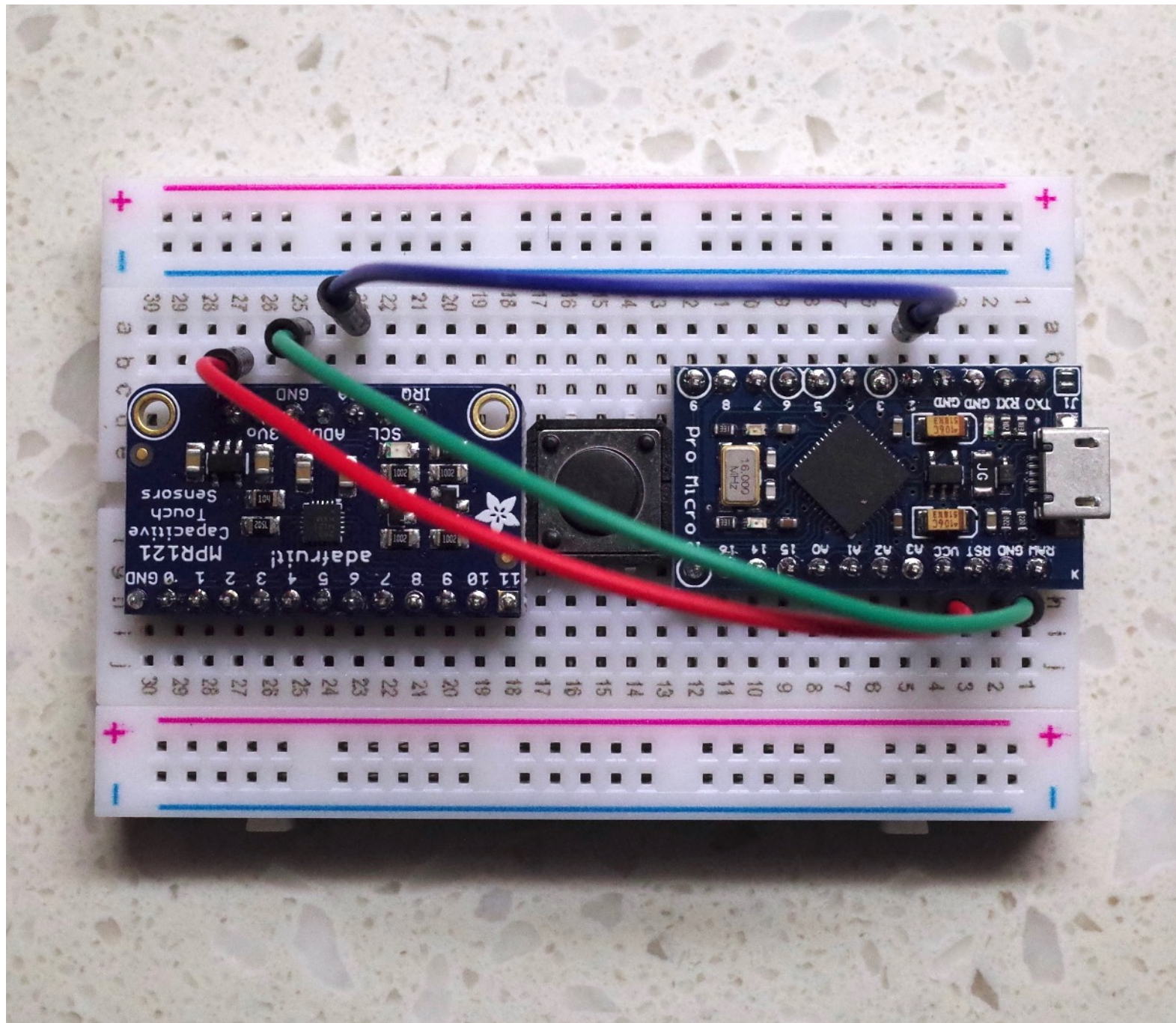


Connect the “GND” pin on the Arduino board to the “GND” pin on the Capacitive Touch Board.



# HOOKING UP YOUR SENSORS! FOLLOW MB ON THE PROJECTOR

---

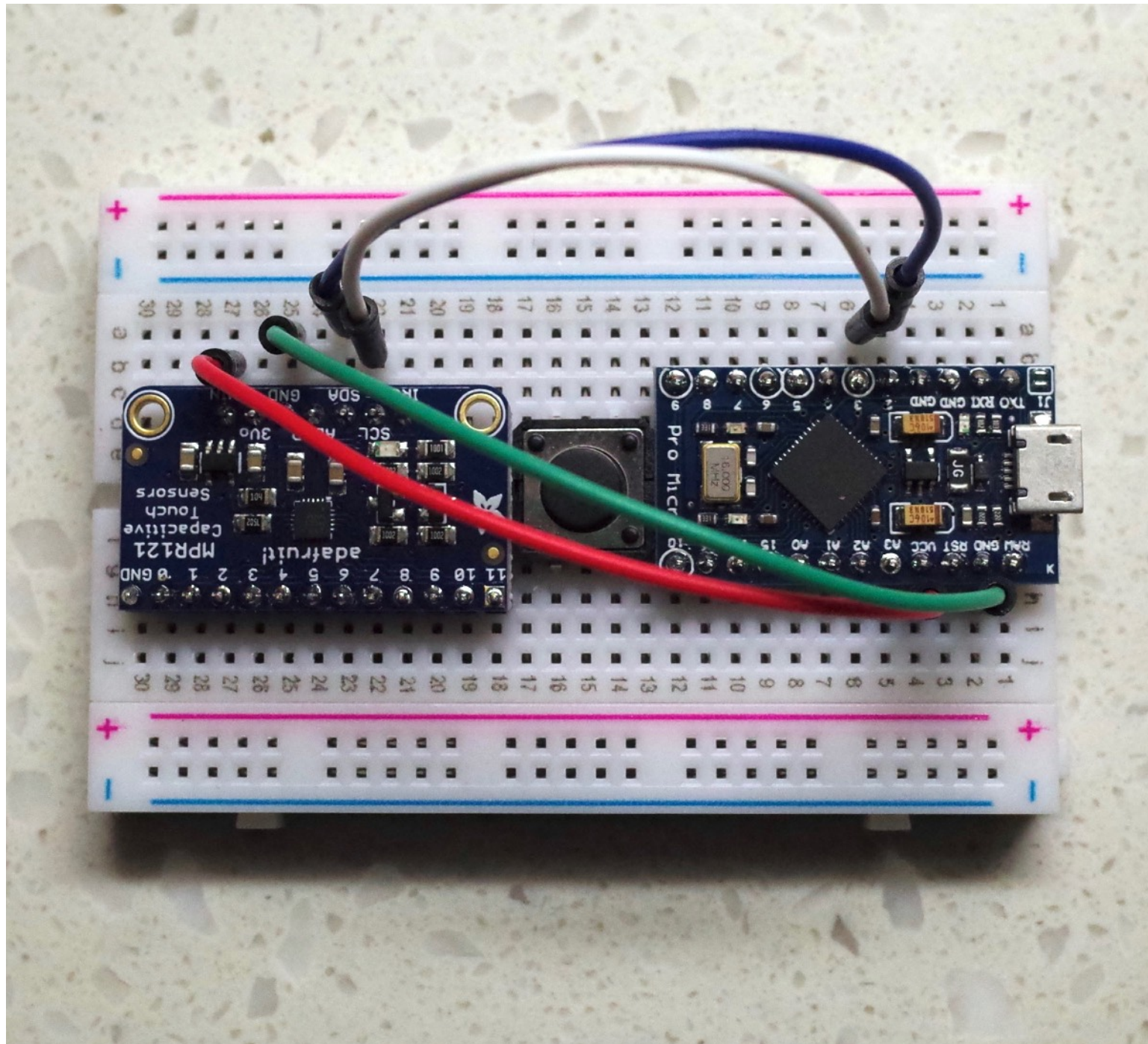


Connect the “2” pin on the Arduino board to the “SDA” pin on the Capacitive Touch Board.



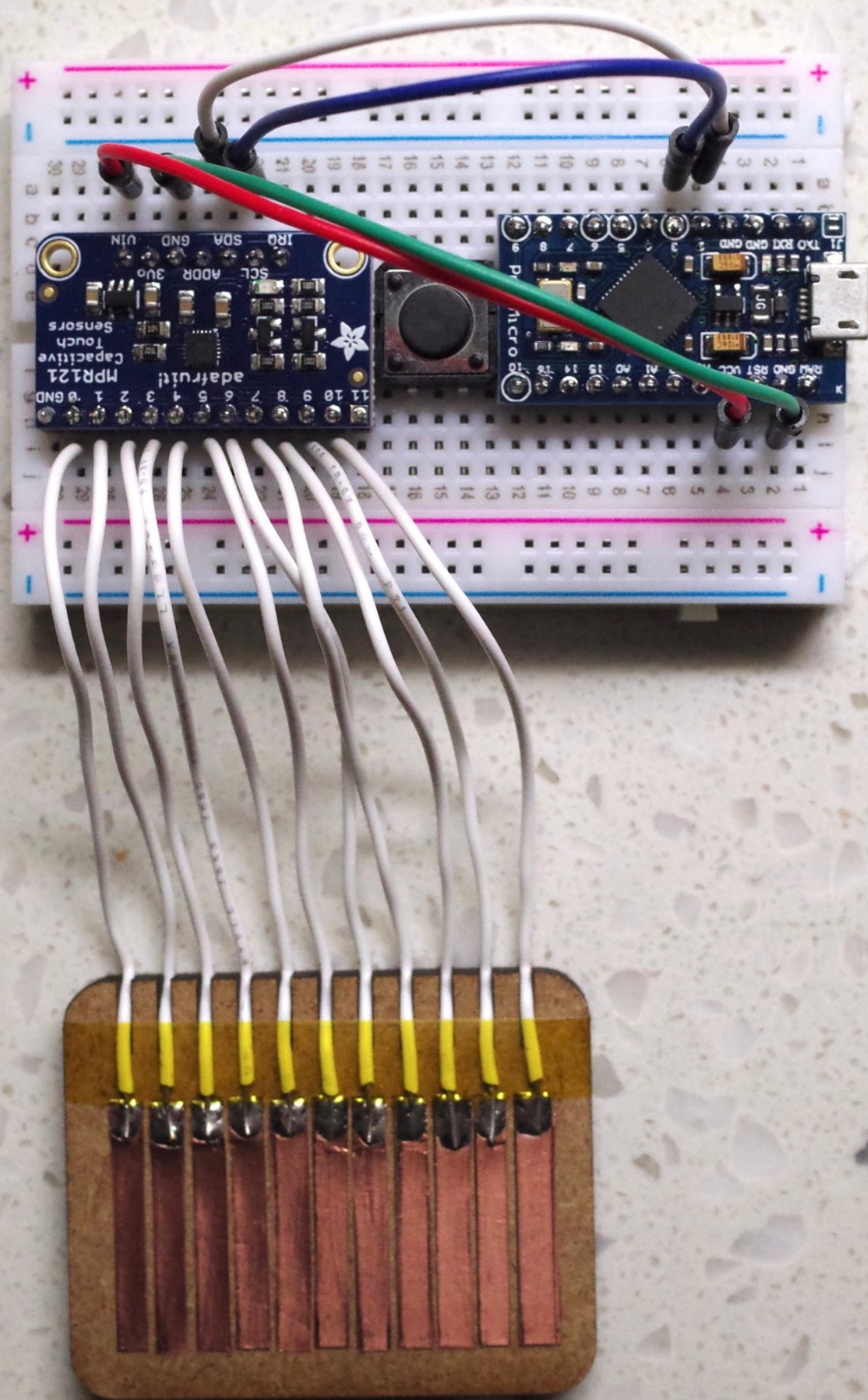
# HOOKING UP YOUR SENSORS! FOLLOW MB ON THE PROJECTOR

---



Connect the “3” pin on the Arduino board to the “SCL” pin on the Capacitive Touch Board.





## HOOKING UP YOUR SENSORS! FOLLOW MB ON THE PROJECTOR

.....

Connect the touch strip wires in order (pay attention that you don't switch any) to the “0” to “10” pins on the Capacitive Touch Board.

## SAFETY CHECK!

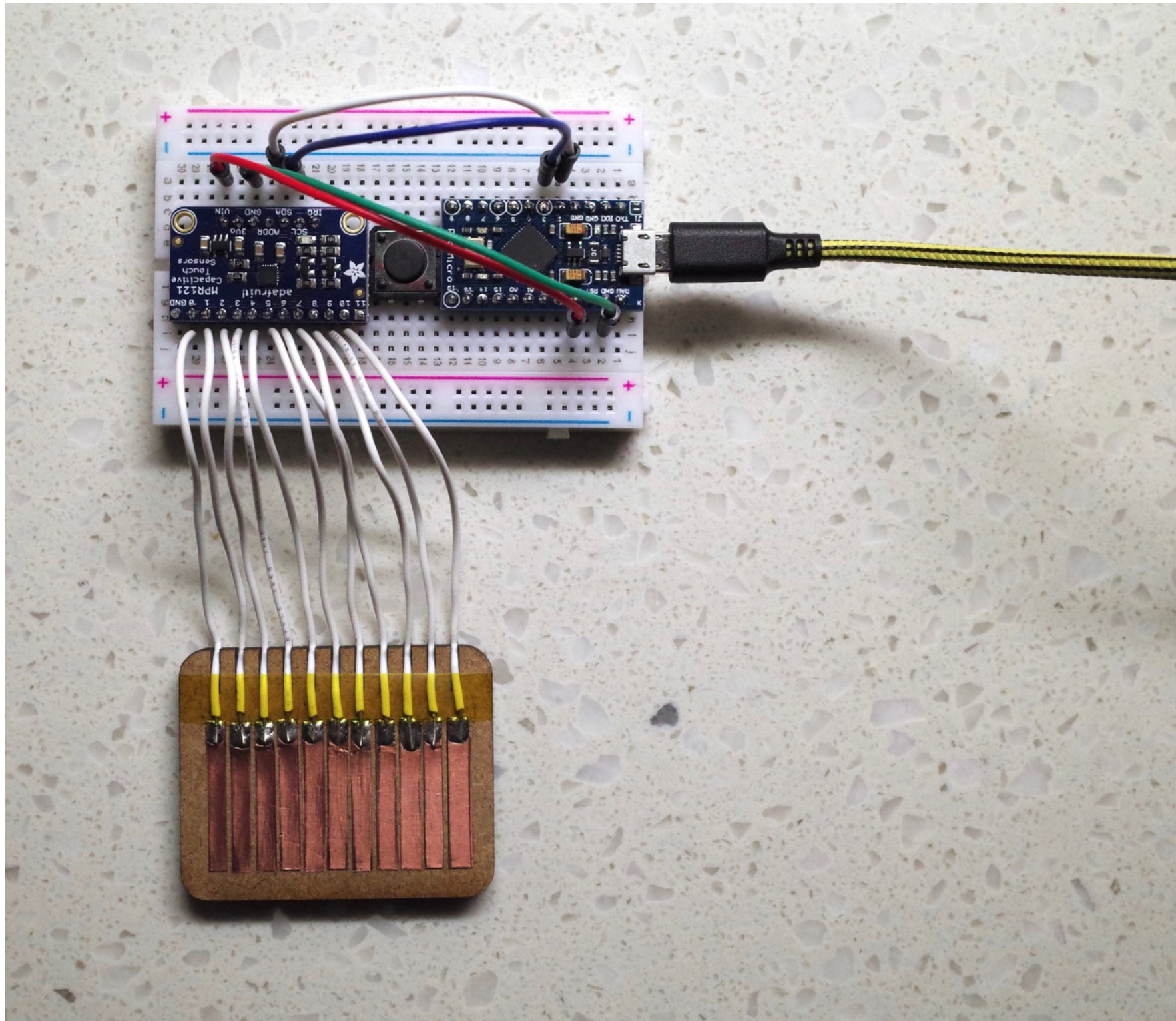
---

*BEFORE PLUGGING THE  
USB CABLE INTO YOUR  
COMPUTER GET MATT TO  
CHECK YOUR BOARD SO  
YOU KNOW IT'S SAFE AND  
NOTHING WILL BREAK!*



# HOOKING UP YOUR SENSORS! PLUG IN USB

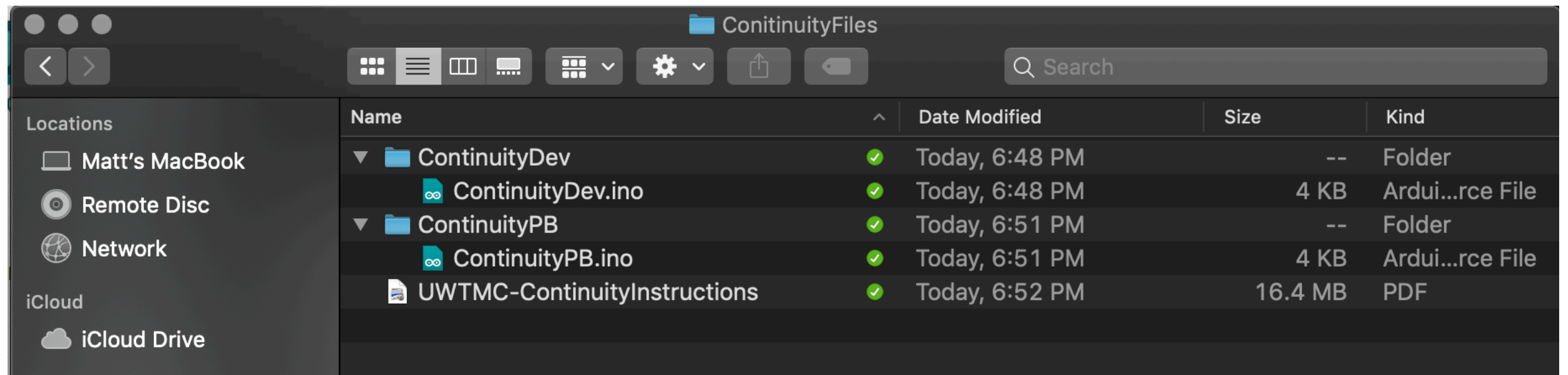
.....



# ON YOUR COMPUTER

---

*Launch Arduino by opening “ContinuityDev.ino”*



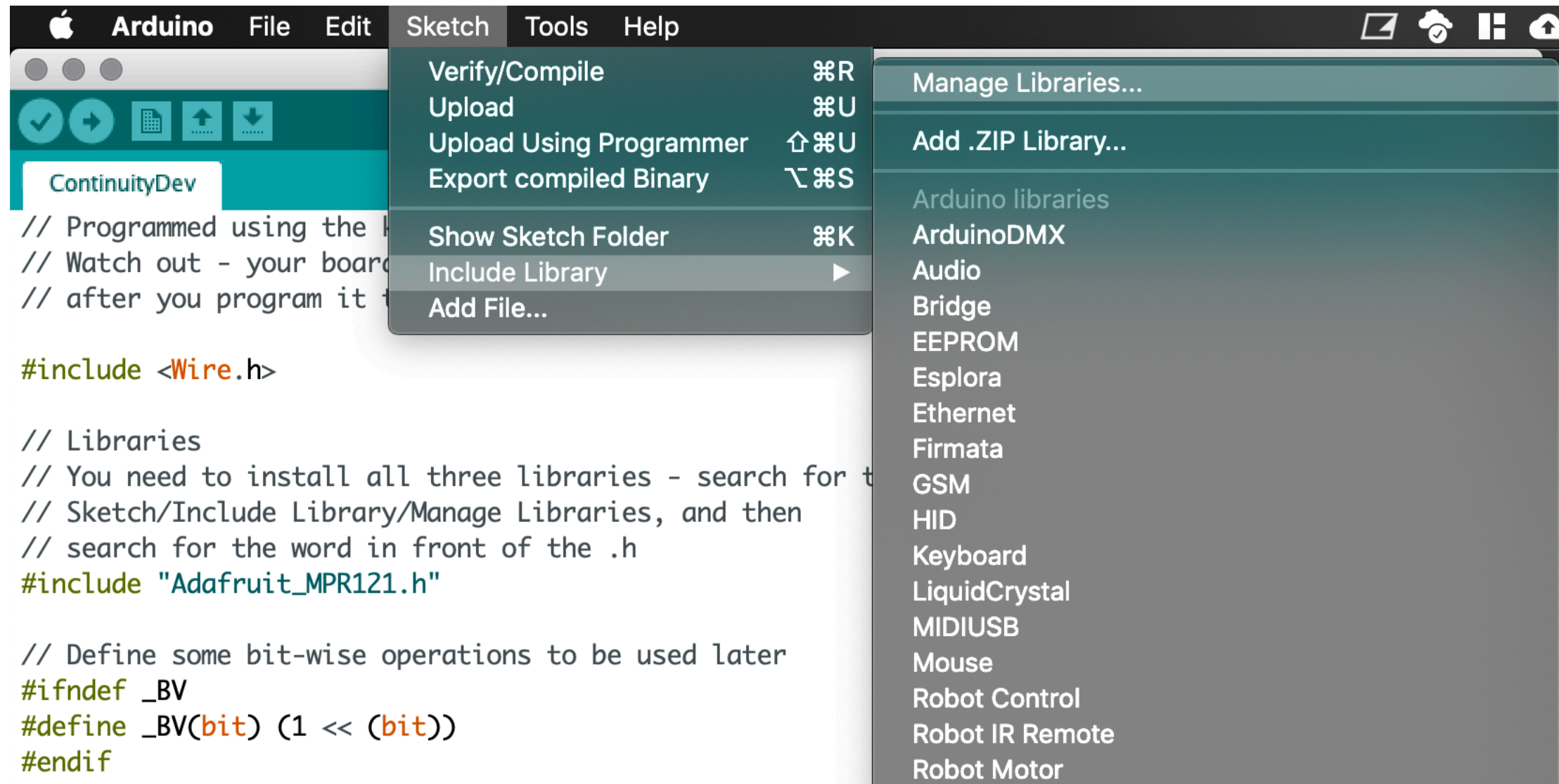
*Arduino IDE - a software platform used to program your microcontroller.*

<https://www.arduino.cc/en/Main/Software>



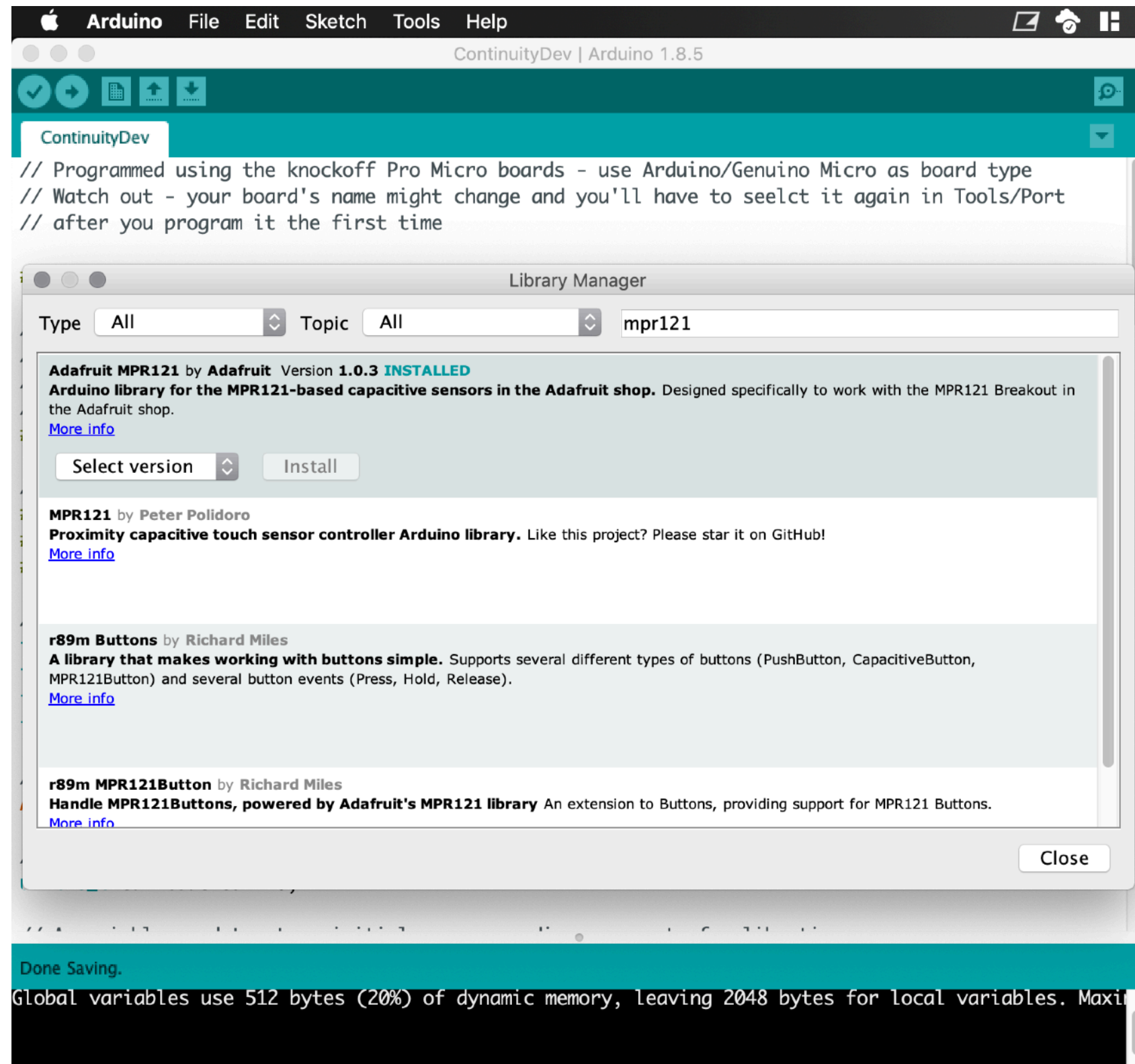
# LIBRARIES – INSTALL THESE TO ADD FUNCTIONS TO YOUR PROGRAM

.....



# LIBRARIES – INSTALL THESE TO ADD FUNCTIONS TO YOUR PROGRAM

---



*We need to add the Adafruit MPR121 library to run the capacitive touch board.*



# PROGRAM YOUR DEVICE – BOARD TYPE

The screenshot displays the Arduino IDE interface. The main window shows a sketch with the following code:

```
// Programmed using the knockoff
// Watch out - your board's name
// after you program it the first time

#include <Wire.h>

// Libraries
// You need to install all three
// Sketch/Include Library/Manager
// search for the word in front of the .h
#include "Adafruit_MPR121.h"

// Define some bit-wise operations to be used later
#ifndef _BV
#define _BV(bit) (1 << (bit))
#endif

// Variables used to calculate centroid value
float centroidArea = 0;
float area = 0;
float prevCentroid = 0;
float currCentroid = 0;

// Initialize the capacitive touch board
Adafruit_MPR121 cap = Adafruit_MPR121();

// Keep track of the last pins touched
// so we know when buttons are 'released'
```

The Tools menu is open, showing the following options:

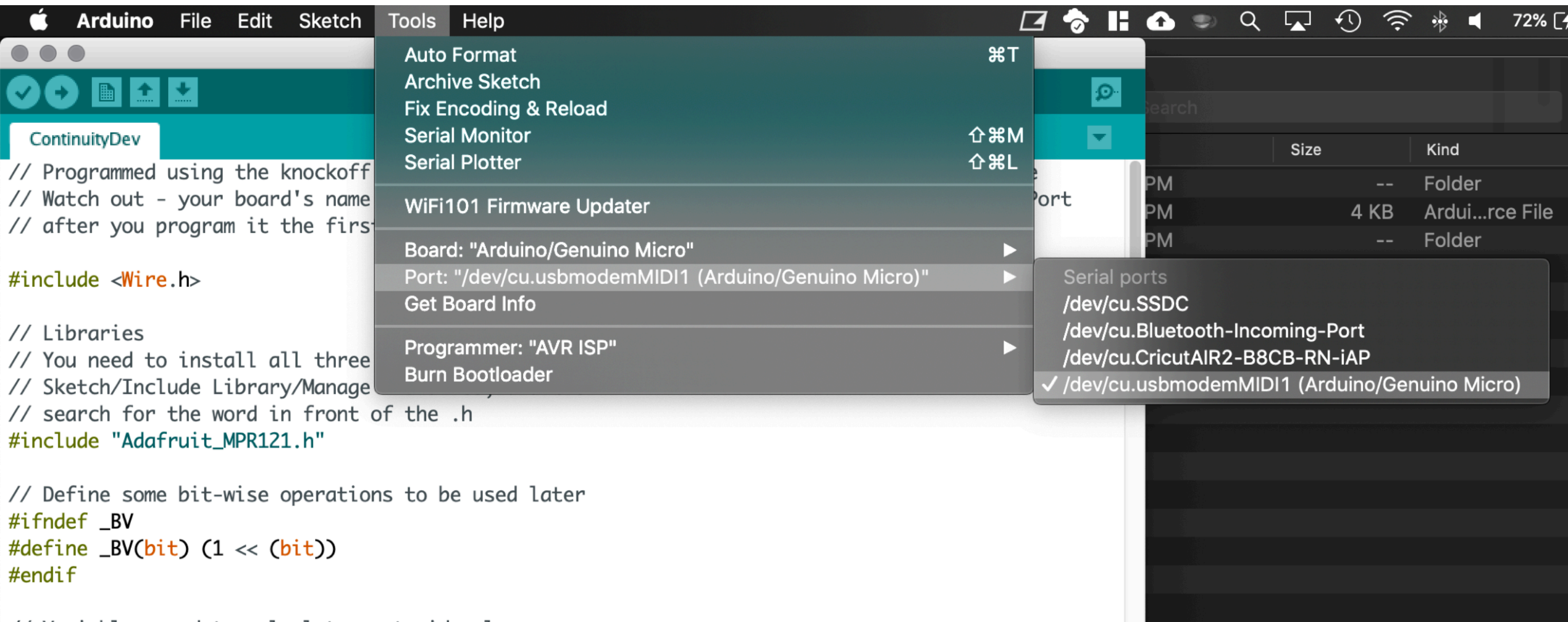
- Auto Format (⌘T)
- Archive Sketch
- Fix Encoding & Reload
- Serial Monitor (⇧⌘M)
- Serial Plotter (⇧⌘L)
- WiFi101 Firmware Updater
- Board: "Arduino/Genuino Micro" (selected)
- Port: "/dev/cu.usbmodemMIDI1 (Arduino/Genuino Micro)"
- Get Board Info
- Programmer: "AVR ISP"
- Burn Bootloader

The Boards Manager window is open, showing a list of boards. The "Arduino/Genuino Micro" board is selected and checked.

Done Saving.

/Users/mattborland/ownCloud/TEACHING/UWTMC/Public Sessions/F2019/Session 1 - Deformation/De

# PROGRAM YOUR DEVICE – PORT



*Sometime this name changes after programming, so you may need to reselect this from time to time.*

*If you get an error that says “board not found” you should check it’s plugged in and that your port is selected correctly.*



# PROGRAM YOUR DEVICE – CONTINUITYDEV.INO

---

*With your arduino plugged in, press the ARROW button to compile the code and send it to the arduino.*



```
Arduino  File  Edit  Sketch  Tools  Help
ContinuityDev | Arduino 1.8.5

ContinuityDev
// Programmed using the knockoff Pro Micro boards - use Arduino/Genuino Micro as board type
// Watch out - your board's name might change and you'll have to select it again in Tools/Port
// after you program it the first time

#include <Wire.h>

// Libraries
// You need to install all three libraries - search for them under
// Sketch/Include Library/Manage Libraries, and then
// search for the word in front of the .h
#include "Adafruit_MPR121.h"

// Define some bit-wise operations to be used later
#ifndef _BV
#define _BV(bit) (1 << (bit))
#endif

// Variables used to calculate centroid value
float centroidArea = 0;
float area = 0;
float prevCentroid = 0;
float currCentroid = 0;

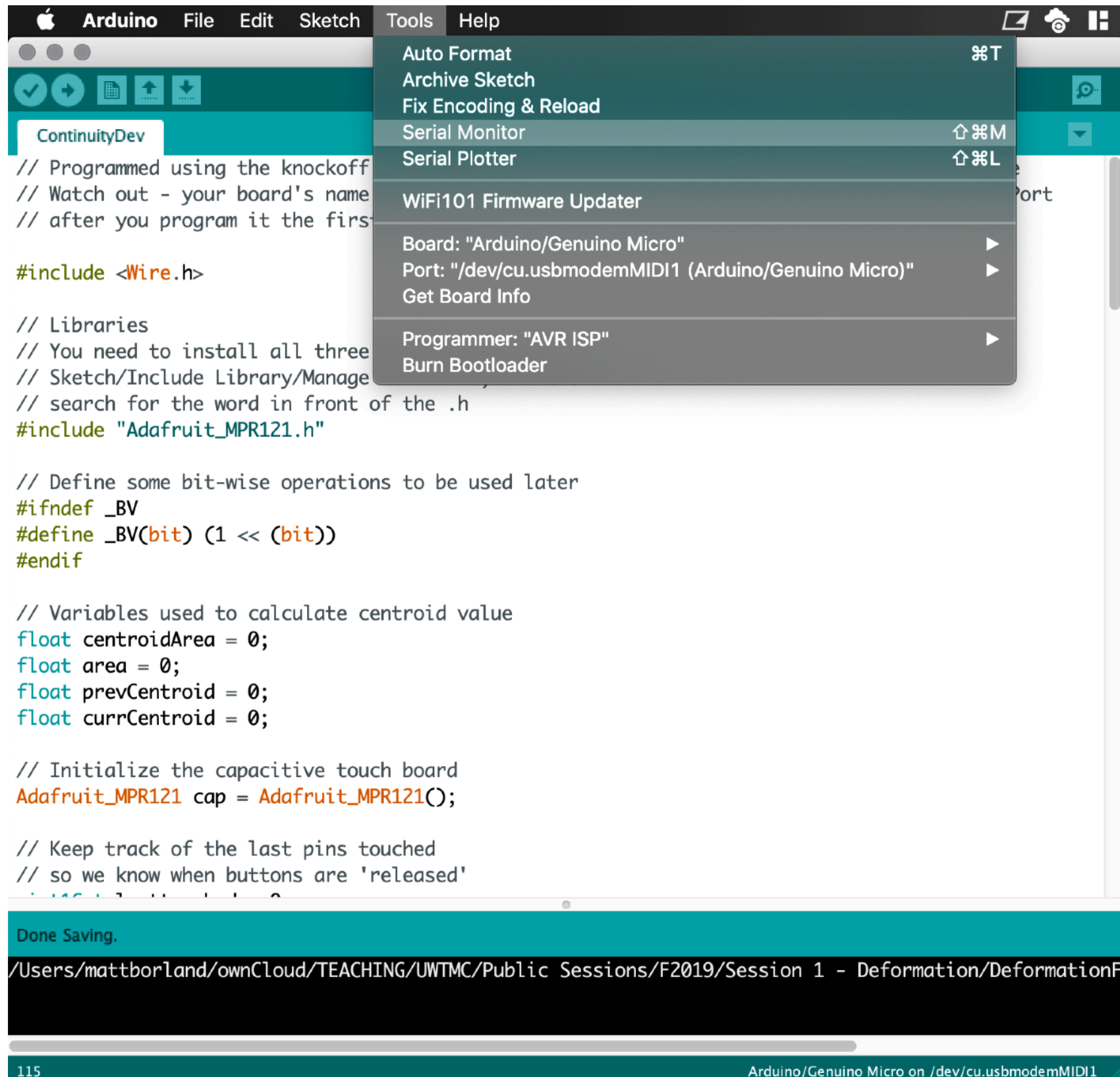
// Initialize the capacitive touch board
Adafruit_MPR121 cap = Adafruit_MPR121();

// Keep track of the last pins touched
// so we know when buttons are 'released'
```

Done Saving.

/Users/mattborland/ownCloud/TEACHING/UWTCM/Public Sessions/F2019/Session 1 - Deformation/DeformationF

# PROGRAM YOUR DEVICE – MONITOR THE SERIAL PORT





# PROGRAM YOUR DEVICE – CHECK POSITION READING

.....

*Slide your finger across the copper strips and see how the values change!*

*If this doesn't work, get help!*

*If it does work you can go onto the next step!*

```
ContinuityDe
Serial.pr
// Defaul
if (!cap
Serial.
while (
}
Serial.pr
// Wait d
delay(100
for(uint8
cal[i]
}
}
// The main
void loop()
// Read f
currtouch
// Reset
centroidA
area = 0;
float f =
float cou
167.00
2.51
106.67
2.92
151.50
3.55
108.00
4.28
117.33
4.95
146.00
5.55
140.00
6.49
134.00
7.46
```

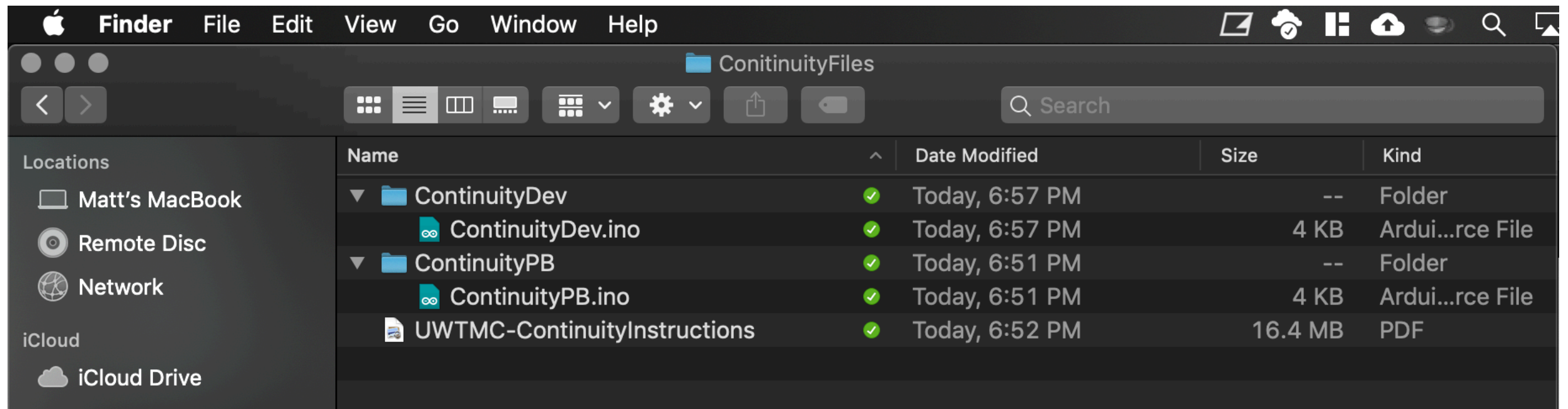
Done uploading

Autoscroll No line ending 115200 baud Clear output

# OPEN A FILE

---

*Launch Arduino and open “ContinuityPB.ino”*



*Arduino IDE - a software platform used to program your microcontroller.*

<https://www.arduino.cc/en/Main/Software>



# LIBRARIES – INSTALL THESE TO ADD FUNCTIONS TO YOUR PROGRAM

---

```
// Libraries
// You need to install all three libraries -
// Search for them under Sketch/Include Library/Manage Libraries,
// Then search for the word in front of the .h
#include <elapsedMillis.h>
#include <MIDIUSB.h>
#include "Adafruit_MPR121.h"
```

*Do the same thing as we did before to install the `elapsedMillis.h` and the `MIDIUSB.h` library.*

*You installed the `MPR121` library earlier.*

*You may already have these installed from a previous week!*

# PROGRAM YOUR DEVICE – CONTINUITYPB.INO

.....

*Now program your board with the new code. This will make your arduino send out MIDI messages to your computer that HELM will turn into sounds.*



```
Arduino  File  Edit  Sketch  Tools  Help
ContinuityPB | Arduino 1.8.5

ContinuityPB
// Programmed using the knockoff Pro Micro boards - use Arduino/Genuino Micro as board type
// Watch out - your board's name might change and you'll have to select it again in Tools/Port
// after you program it the first time

#include <Wire.h>

// Libraries
// You need to install all three libraries - search for them under Tools/Manage Libraries, and then
// search for the word in front of the .h
#include <elapsedMillis.h>
#include <MIDIUSB.h>
#include "Adafruit_MPR121.h"

// Define some bit-wise operations to be used later
#ifndef _BV
#define _BV(bit) (1 << (bit))
#endif

// Variables used to calculate centroid value
float centroidArea = 0;
float area = 0;
float prevCentroid = 0;
float currCentroid = 0;
float pressure = 0;

// Keeps track of the last pins touched
// so we know when buttons are 'released'
uint16_t lasttouched = 0;
uint16_t currtouched = 0;

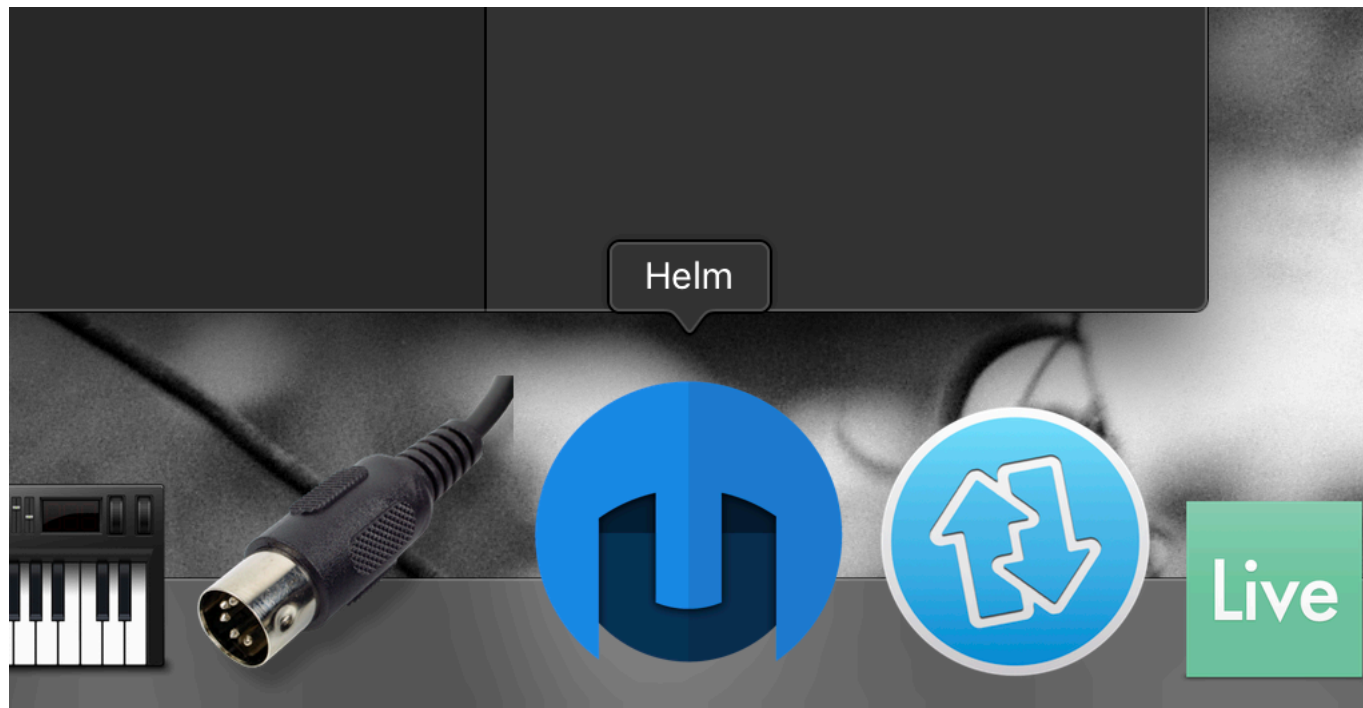
Uploading...

Sketch uses 10266 bytes (35%) of program storage space. Maximum is 28672 bytes.
Global variables use 436 bytes (17%) of dynamic memory, leaving 2124 bytes for local variables. Maximum is 2
```



# OPEN HELM – SOFTWARE SYNTHESIZER

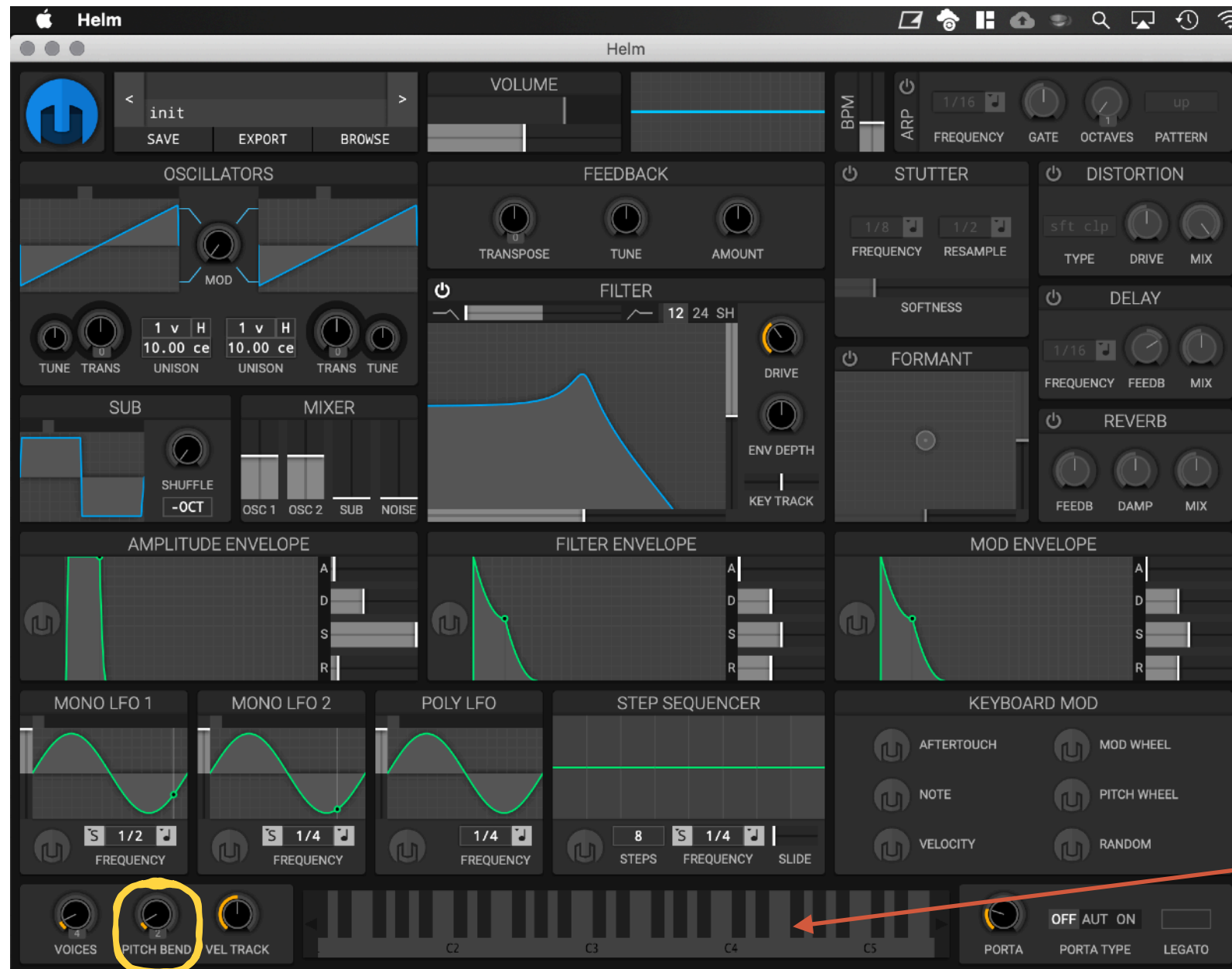
---



*Helm - a software synthesizer  
to make musical sounds with  
your computer.*

<https://tytel.org/helm/>

# SETUP HELM



Adjust the pitch bend amount to change the range of notes your touch strip bends through.

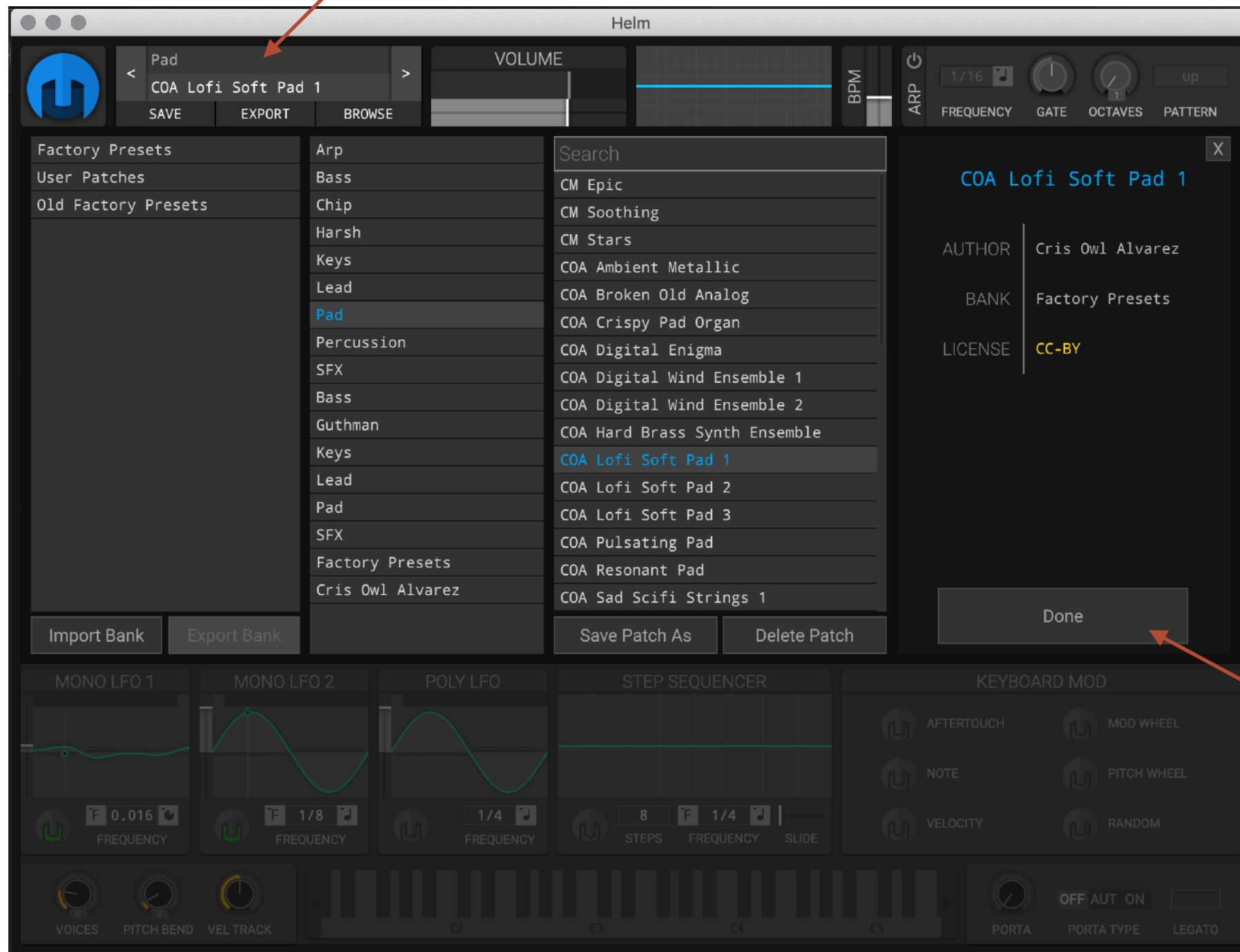
You should see one of the keys of the piano lighting up as you touch the strip!



# SETUP HELM

*Click here to open the menu to select a sound*

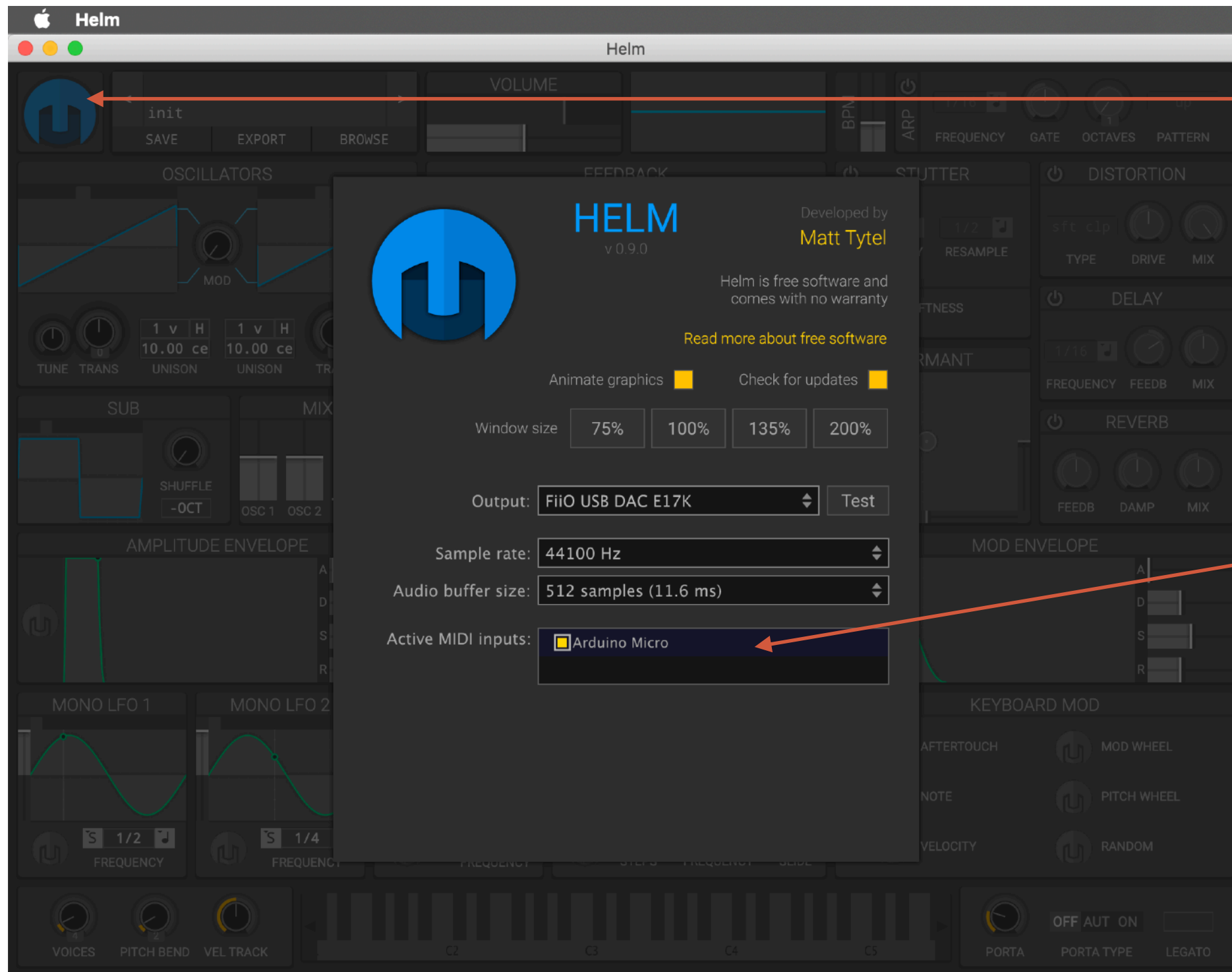
*Make sure your laptop's speakers are on and turned up!*



*Touch the strip to send notes to Helm!*

*Try different sounds, then click done.*

# NOT WORKING? CHECK THE MIDI PORT



*Click the HELMet to get this dialog to open.*

*You should see “Arduino Micro” as a MIDI input, with a yellow box showing it is selected.*