

AUTODESK
Instructables

Biodata Sonification

By [electricityforprogress](#) in [CircuitsElectronics](#)



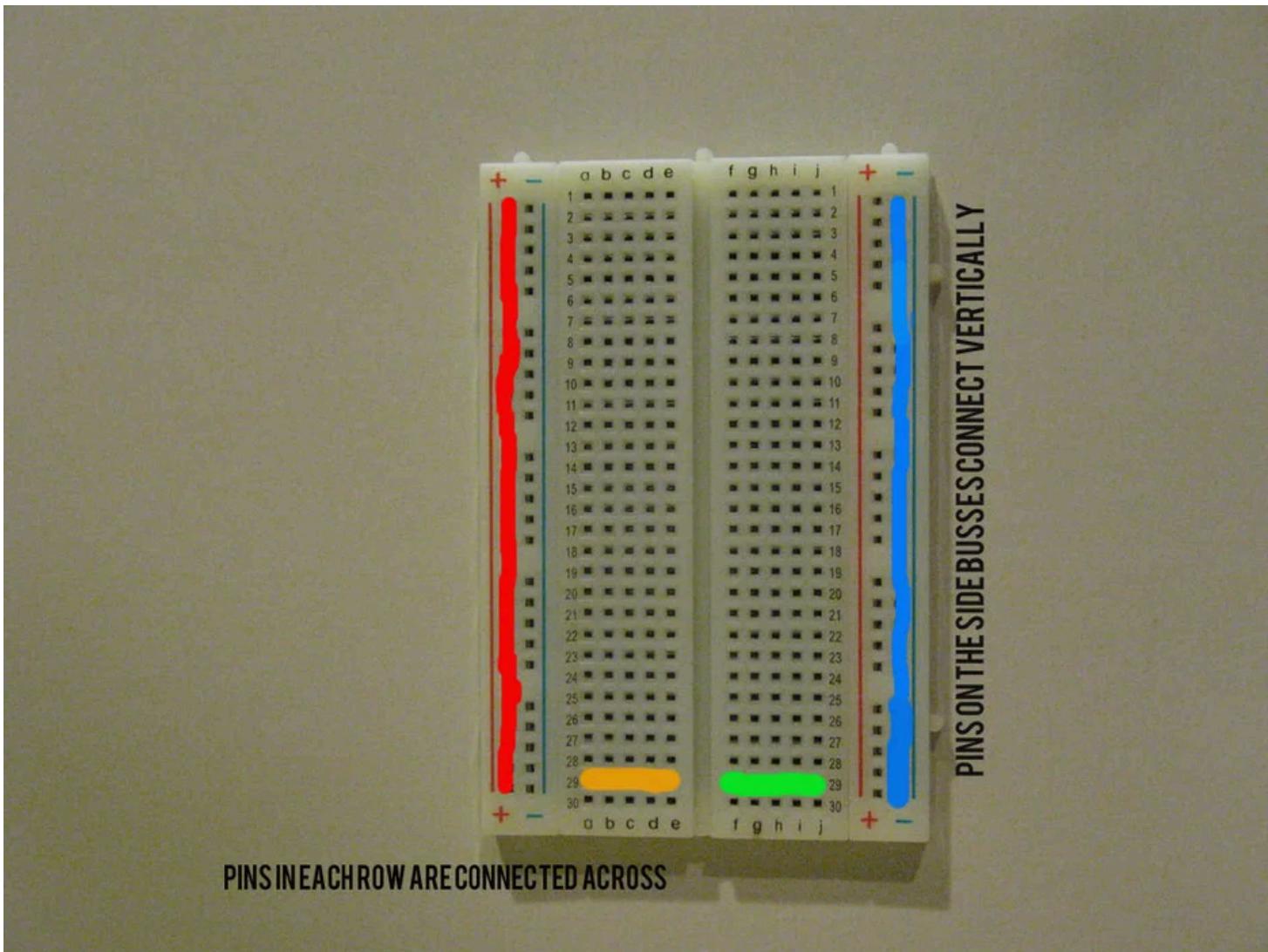
Introduction: Biodata Sonification



Generate MIDI notes based on changes in Galvanic Conductance across two probes.

For the latest code version and updated tutorials please go to electricityforprogress.com and checkout my github project <https://github.com/electricityforprogress/BiodataSonificationBreadboardKit>

Step 1: Solderless Breadboard

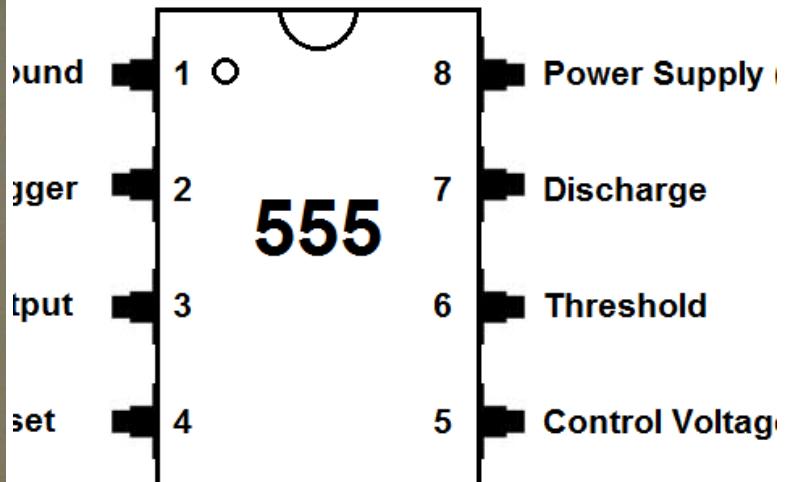
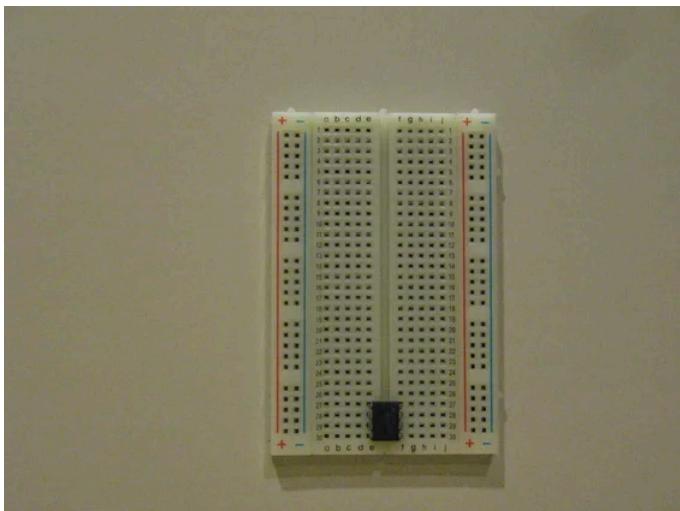


A key tool in electronics experimentation is the Solderless Breadboard. Allowing users to connect components together and reconfigure easily, the Breadboard lets newcomers to electronics and seasoned engineers to prototype designs and connect electronic systems easily.

Breadboards have a series of holes which are electrically connected. Horizontal rows run across the Breadboard in Terminal Strips of 5 connected points points and are marked with the letters abcde and fghij. A large divide down the middle of the breadboard separates the horizontal rows, this facilitates the use of Dual Inline Package (DIP) microchips. On the sides of the breadboard are vertical columns of holes, usually marked with Red and Blue lines. These vertical columns are used most often for power connections (positive voltage and ground), and are called a 'Bus'. We will be attaching all of our Positive and Ground connections to these Buses on each side of the breadboard. In a later step we will tie together the Grounds and the Positive Buses on each side of the breadboard.

In order to 'connect' two electronic components, we simply place the leads (or 'legs') of the parts into adjacent horizontal holes. This allows a user to connect multiple components together using each horizontal row of 5 points.

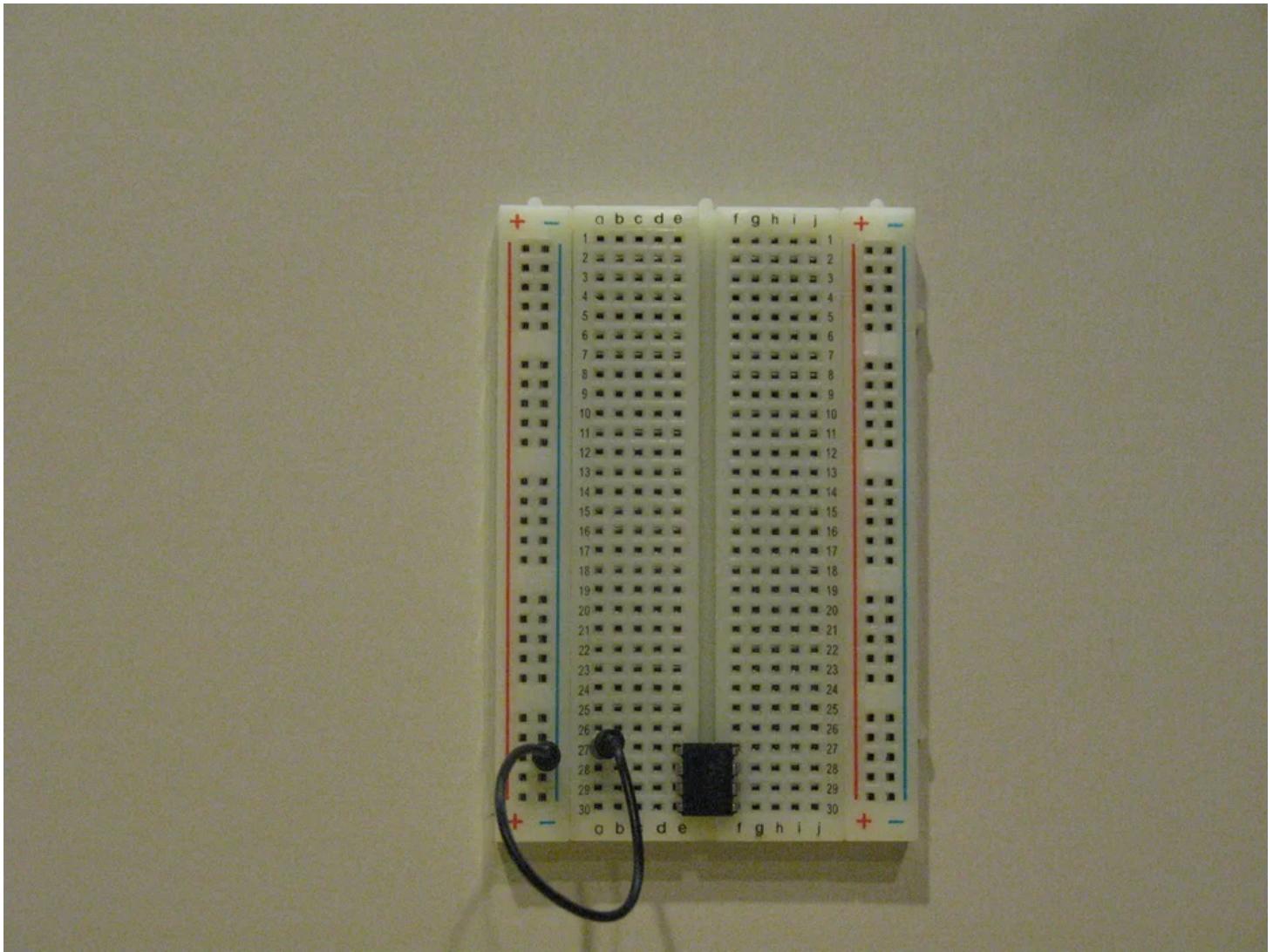
Step 2: Insert 555 Timer



The 555 timer is an 8 pin DIP microchip, which we will configure as an astable multivibrator capable of measuring electrical conductivity. Orient the chip so that Pin 1 is at the top - you will see a small circle near pin 1 on the chip, also see the diagram which identifies each of the pins on the 555 Timer.

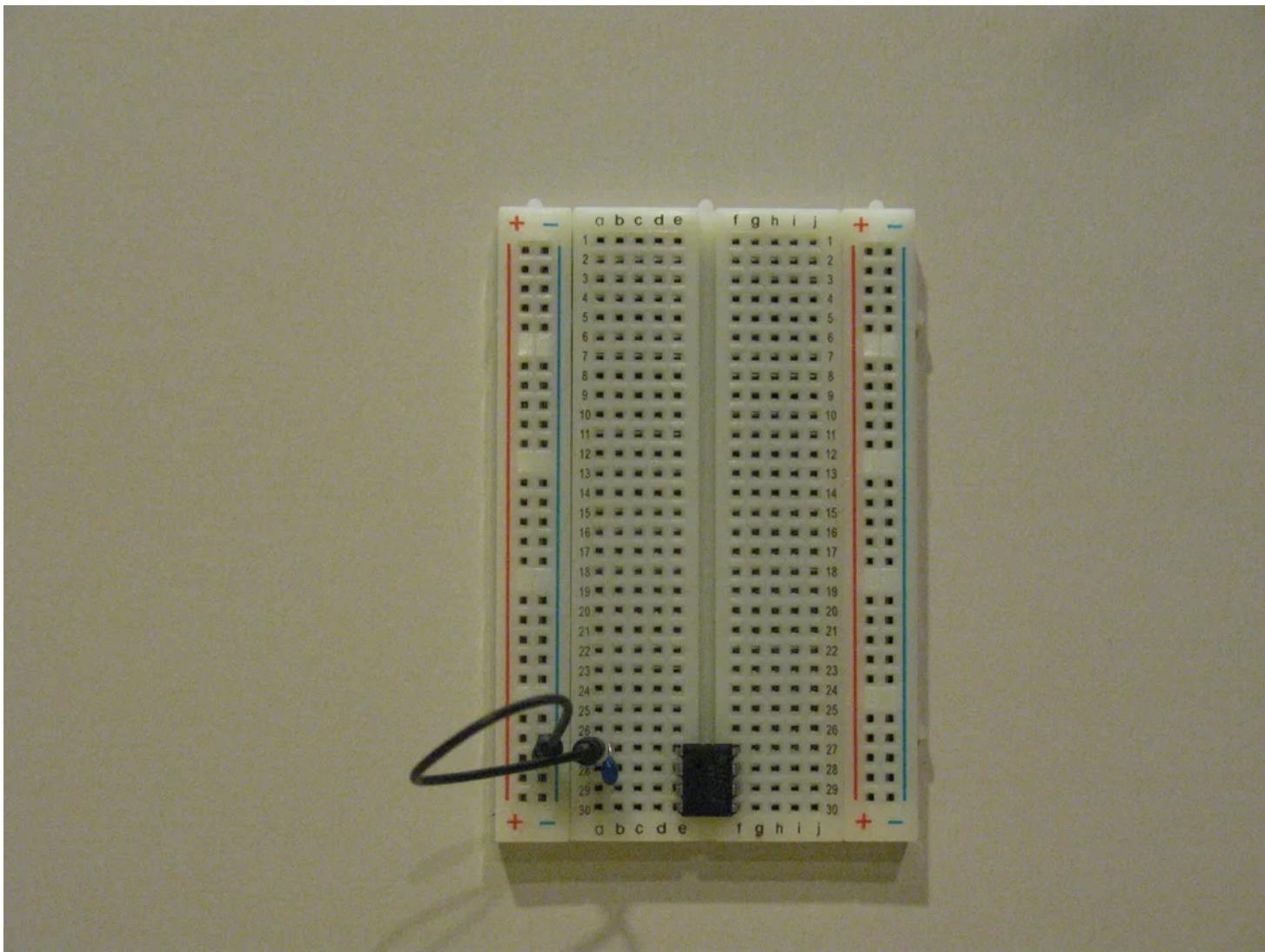
Place the 555 timer at the bottom of the Breadboard. The breadboard is arranged with a gap down the middle, the microchip should span across this gap. The rows of the breadboard are numbered, we will be inserting the 555 timer in rows 27, 28, 29, and 30, with pin 1 in row 27.

Step 3: Pin 1 to Ground



Attaching the 555 Pin 1 to Ground, add a jumper wire from row 27 column A to the Ground Bus.

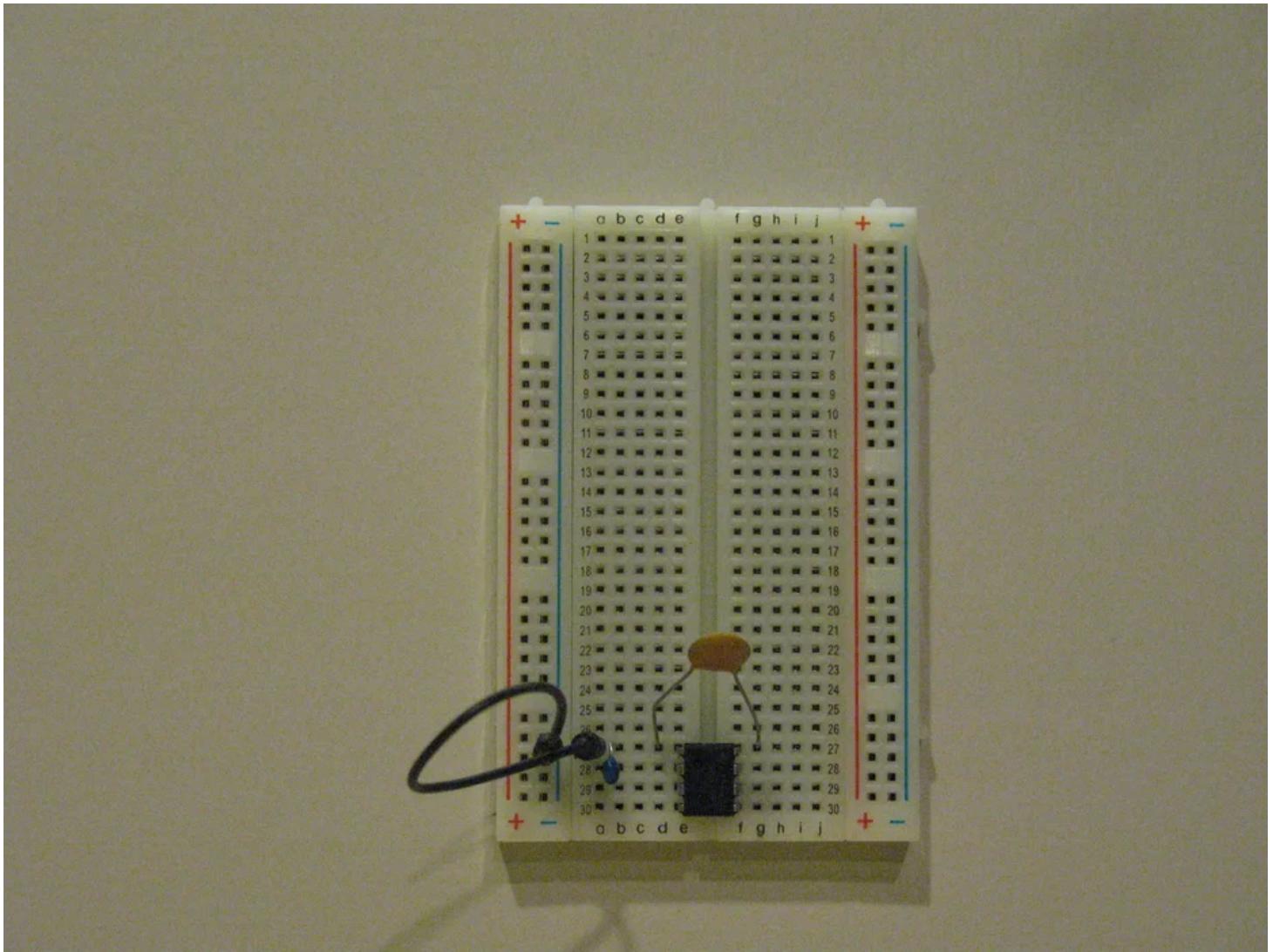
Step 4: Timing Capacitor C1



Connect the timing Capacitor C1 (0.0042uF) between Pin 1 and Pin 2 of the 555 Timer. Insert the tiny blue capacitor into rows 27 and 28 in column B.

This capacitor sets the overall frequency range of the timer, here we use a very small value in order to get the highest resolution of pulses out of the 555 as we measure fluctuations in electric capacitance across the two probes.

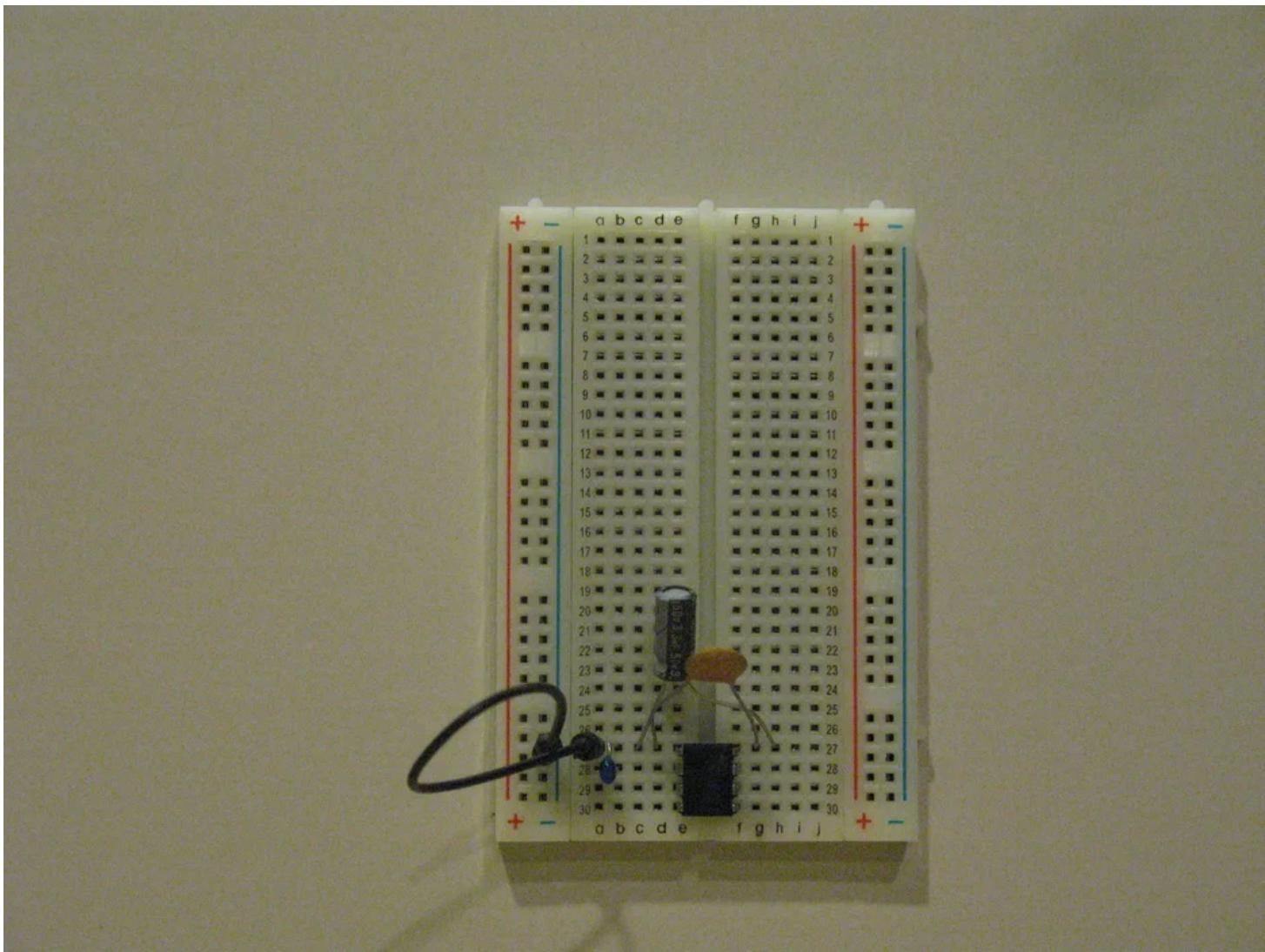
Step 5: Decoupling Capacitor C2



Connect the high frequency decoupling capacitor C2 (1uF) across the 555 Timer's positive and ground, pins 1 and 8 in row 27, column D and G.

It can be helpful to trim the legs of the capacitor, for a better fit on the breadboard, but be careful to leave enough space for the legs to span the microchip and fully connect with the breadboard sockets.

Step 6: Decoupling Electrolytic Capacitor C3



Connect the low frequency decoupling Electrolytic Capacitor C3 (41uF) across the 555 Timer's positive and ground, pins 1 and 8 in row 27, column C and H.

Note that Electrolytic capacitors are polarized, identifying the negative end with a white stripe down the side of the cap; ensure that the negative side of the capacitor goes to Pin 1 (Ground) column C and the positive side of the capacitor goes to Pin 8 (Positive) column H.

Step 7: LED Output

Add the Red LED to the output pin 3 of the 555 Timer Row 29 pin A and across to the Ground Bus. Place the longer lead of the LED (anode) in Row 29 Column A, with the shorter leg of the LED in one of the Ground Bus holes.

**- LEDs are polarized and must be inserted in the correct orientation. The LED's Cathode leg (negative) can be identified by a flattened edge on the side of the LED, and the positive Anode can be identified by the longer leg. LED's polarity and color can be identified using a simple button battery, by sliding the battery in between the LED leads, you will either see the LED glow or not, try turning the battery the other direction. The LED will illuminate when the battery + (wide flat) end is connected to the Anode (longer leg) and the battery - (smaller button) is connected to the Cathode Ground leg. Grab a CR2032 3v button battery and try it out!

After you get everything working in the last step, you can come back and trim the legs of the LED if desired.

NOTICE: under all normal circumstances, a resistor would be added between the output pin and the LED. In order to simplify the build of this kit, the current limiting resistors have been omitted. We have included resistors for each LED in the kit. Modified instructions including current limiting resistors will be provided as an appendix.

Step 8: Jumper 555 Trigger to Threshold

Connect a Jumper wire between Pin 2 and Pin 6 of the 555 Timer Row 28 column D to Row 29 Column G.

This attaches the threshold and the trigger pins of the 555 timer, which form the input connection for the primary electrode.

Step 9: Jumper 555 Reset to V+

Connect Pin 4 of the 555 Timer to the Positive Bus using a Jumper wire Row 30 Column D to the Positive Bus

Connect Pin 8 of the 555 Timer to the Positive Bus using a Jumper wire Row 27 Column I to the Positive Bus

(add image and step for 555 VCC to V+)

Step 10: Resistor R1 100K 555 Discharge to Positive Bus

Connect Resistor R1 (100k) between Pin 7 of the 555 and the Positive Bus. Place one side of the Resistor in Row 28 Column J and the other side of the resistor to the Positive Bus.

Step 11: Probe Input Jack

The Probe input is an 3.5mm mono jack, which connects to the breadboard through two soldered pins. While its a tight spot, the header pins soldered to the jack will fit into Row 28 and 29 Column H.

The header pins have been added to the jacks to make it easier for the user to build the kit. Please note that excess stress on the jack or pins may cause damage to the solder connection. If your kit does not have the header pins soldered to the jack, please see the appendix for soldering instructions for the jack and header.

Step 12: Positive Bus Jumper

Connect the Positive Bus on both sides of the breadboard by inserting a Jumper wire between the top highest points on the left and right (red) Power Bus.

Step 13: Ground Bus Jumper

Connect the Ground Bus on both sides of the breadboard by inserting a Jumper wire between the top highest points on the left and right (blue) Ground Bus.

Step 14: Testing the Galvanometer

Now we are ready to hookup some batteries and test the Galvanometer we just built from the 555 Timer.

Insert 3 AA batteries into the black Battery box, ensure the power switch on the box is in the 'OFF' position. Attach the Battery box Red wire to the Breadboard Positive (red) Bus, attach the Battery box Black wire to the Breadboard Ground (blue) Bus. Now slide the power switch on the battery box to 'ON'. The LED should be illuminated, showing the 555 timer is powered on.

Attach the white electrode leads (don't bother using the sticky pads yet) to the 3.5mm jack connecting to the Galvanometer. By touching the metal button ends of the electrodes with your fingers, you will be able to see the LED flash based on changes in conductivity. Touching the electrodes very lightly can show the LED flash on and off slowly, by squeezing the electrodes really hard the LED flashes very fast, appearing like the LED remains lit or slightly dims.

Step 15: Insert ATMEGA328 28pin DIP

Your MIDIsprout Kit comes with a preprogrammed ATMEGA328 micro controller, with fuses set to run at 8Mhz on the internal oscillator (Fuses: Low-E2 High-D9 Ext-FF) , and preloaded with the MIDIsprout firmware. This 28 pin DIP has two parallel rows of 14 pins.

Insert the 328p chip at the top of the breadboard, identifying Pin 1 by the small circle on the chip, into Rows 1 - 14 spanning the DIP across the gap in Columns E and F.

******To easily reprogram and experiment, it is possible to add a 16Mhz oscillator on pins 9 and 10 of the breadboard, and program using an arduino Uno board with modifications of the MIDIsprout code. The ATMEGA328 can also be reprogrammed through ICSP with an external programmer (other arduino) and a maze of Jumper wires ;)

****Also as an addendum, MIDIsprout Kit can be built using the previous steps to assemble the Galvanometer, with the breadboard attached directly to an Arduino Uno! Stay tuned...**

For reference, the code preloaded into the current version MIDIsprout :

Arduino Code:

MIDI_PsychoGalvanometer (https://codebender.cc/sketch:49812?referrer=electricityforprogress) by Download electricityforprogress (https://codebender.cc/user/electricityforprogress?referrer=electricityforprogress)

MIDI_PsychoGalvanometer.ino

LEDFader.cpp

LEDFader.h

```
2  MIDI_PsychoGalvanometer v021           (/?      (https://codebender.cc/sketch:49812?referrer=electricityforprogress)
3  Accepts pulse inputs from a Galvanic Conductance sensor
4  consisting of a 555 timer set as an astablemultivibrator and two electrodes
5  Through sampling pulse widths and identifying the locations, MIDI Note and
6  are generated. Features include Threshold, Scaling, Control Number, and
7  using PWM through an RC Low Pass filter.
8  -----
9
10 #include <LEDFader.h> //manage LEDs without delay() jgillick/arduino-LEDFader
11
12 //*****
13 //set scaled values, sorted array, first element scale length
14 int scaleMajor[] = {7,1, 3, 5, 6, 8, 10, 12};
15 int scaleDiaMinor[] = {7,1, 3, 4, 6, 8, 9, 11};
16 int scaleIndian[] = {7,1, 2, 2, 5, 6, 9, 11};
17 int scaleMinor[] = {7,1, 3, 4, 6, 8, 9, 11};
18 int scaleChrom[] = {12,1,2,3,4,5,6,7,8,9,10,11,12};
19 int *scaleSelect = scaleChrom; //initialize scaling
20 int root = 0; //initialize for root
21 //*****
22
23 const byte interruptPin = INT0; //galvanometer input
24 const byte knobPin = A0; //knob analog input
25
26 const byte samplesize = 10; //set sample array size
27 const byte analysize = samplesize - 1; //trim for analysis array
28
29 const byte polyphony = 5; //above 8 notes may run out of ram
30 byte channel = 1; //setting channel to 11 or 12 often helps simply computation
31 int noteMin = 36; //C2 - keyboard note minimum
32 int noteMax = 96; //G7 - keyboard note maximum
33 byte note[8]; //sends each note out CHAN 1-4, for use with General MIDI 1
34 byte controlNumber = 80; //set to mappable control, low values may interfere
```

To program your Arduino from your browser, install the codebender app or [Arduino Create](#). Agent for codebender. Learn more (<https://codebender.cc/static/plugin>).

Please select a board

Step 16: Power the ATMEGA328

Attach the VCC pin on the 328 to the Positive Bus using a Jumper between Row 7 Column A and the Positive Bus.

Step 17: Ground the ATMEGA328

Attach the Ground pin on the 328 to the Ground Bus using a Jumper between Row 8 Column B and the Ground Bus.

Step 18: Power the ATMEGA328 (analog)

Attach the analog Voltage pin on the 328 to the Positive Bus using a Jumper between Row 9 Column J and the Positive Bus.

Step 19: Ground the ATMEGA328 (analog)

Attach the Ground pin on the 328 to the Ground Bus using a Jumper between Row 7 Column J and the Ground Bus.

Step 20: 555 Timer Output to ATMEGA328 Input

Connect the output pin from the 555 Timer to the Input Pin 4 on the 328 with a Jumper wire between 555 Timer pin 3 Row 29 Column D and Row 4 Column D.

Here the digital output of the 555 triggers an interrupt pin on the 328, INT0, which measures and compares pulse durations.

Step 21: Knob

The included knob should be prepared by gently bending its three legs (bend all three at the same time) so the knob can stand vertically. Insert the Knob onto the left side of the breadboard in Column A Rows 19, 20, and 21.'

Step 22: Knob Wiper to ATMEGA328 Analog Input

Connect the center pin of the Knob to the Analog Input (A0) of the 328 using a Jumper wire. Attach a jumper between the Knob Row 20 Column E and 328 (A0 pin) Row 6 Column G.

Step 23: MIDI Jack

Insert the MIDI Jack into the breadboard. Prepare the jack by identifying the two pointed mounting pins located at the front of the MIDI jack and bending them upward to point out the front of the MIDI jack. Place the MIDI jack on the right side of the breadboard, with the jack facing the right side. Insert the MIDI jack into Column I and J, Rows 18, 19, 21, 23, and 24. The five MIDI jack pins will fit (snuggly) into the breadboard, be careful not to push too hard.

Step 24: MIDI Data Pin to ATMEGA328 Tx

Connect the MIDI Data output pin to the ATMEGA328 serial Transmit (Tx) pin, by attaching a jumper between Column F Row 23 (MIDI Data pin 5) and Column B Row 3 (328 Tx).

Step 25: MIDI Power Resistor to V+

Connect a resistor between the MIDI power pin (4) and V+ using a 220 Ohm resistor connected to Column H Row 19 (MIDI power) and the Positive Bus on the right side of the board.

Step 26: MIDI Ground Jumper

Connect the MIDI Ground pin to the Ground bus using a Jumper wire between Column F Row 21 (MIDI Ground) and the Ground Bus.

Step 27: Knob Positive Voltage

Connect the Knob positive voltage pin to the Positive Bus using a jumper between Column D Row 19 and the Positive Bus.

Step 28: Knob Ground

Connect the Knob Ground pin to the Ground Bus using a jumper between Column D Row 21 and the Ground Bus.

Step 29: LEDs (red)

There are 5 colored LEDs in the MIDIsprout which provide a light show and indication of the state of the MIDI notes being played.

Connect the LED (red) Anode - long leg to Column A Row 5 and the LED Cathode to the Ground Bus.

**- For simplicity, we are omitting current limiting resistors in this build, please see the appendix for steps to include resistors with the LEDs.

Step 30: LEDs (yellow)

Connect the LED (yellow) Anode - long leg to Column A Row 11
Connect the LED (red) Anode - long leg to Column A Row 5 and the LED Cathode to the Ground Bus.
and the LED Cathode to the Ground Bus.

Step 31: LEDs (green)

Connect the LED (green) Anode - long leg to Column A Row 12 and the LED Cathode to the Ground Bus.

Step 32: LEDs (blue)

Connect the LED (blue) Anode - long leg to Column J Row 14 and the LED Cathode to the Ground Bus.

Step 33: LEDs (white)

Connect the LED (white) Anode - long leg to Column J Row 13 and the LED Cathode to the Ground Bus.

Step 34: 16MHz Crystal Oscillator PlaceHolder

The 16MHz crystal oscillator should be added on pins 9 and 10 of the ATMEGA328 Row 9 and 10 Column C. The part is not polarized and the crystal can be inserted into pins 9 and 10 in either orientation.

Step 35: Battery Pack

Attach the battery pack to the breadboard by placing the battery pack Red wire into the breadboard Positive Voltage Bus and the Back wire into the breadboard Ground Bus. Insert 3 AA batteries and switch on the battery box. With the power on the LED by the 555 Galvanometer should illuminate.

Connect the electrode leads to the jack at the bottom of the breadboard, and touch the two button ends of the leads. The Galvanometer LED should flash in response to the conductivity across your fingers.

Step 36: Biodata Sonification

When the electrode leads are touched or attached using gel pads, the MIDIsprout program will detect small changes in conductivity and represent these changes as MIDI notes and colorful lights!

Connecting a MIDI cable from the MIDI jack on the bread board, the MIDIsprout Kit can be attached to synthesizers, keyboards, sound generators, and computers supporting MIDI to produce sounds in reaction to the MIDI notes.

By turning the knob, the Threshold/Sensitivity of the MIDIsprout can be adjusted. By decreasing the threshold, smaller fluctuations in conductance from the galvanometer can be detected; by increasing the threshold, larger changes are required in order to produce notes. During long term installations, I use a low threshold setting which produces a pleasant babbling stream of MIDI data. For public interactive events with multiple plants, I turn the threshold up rather high, which results in MIDI notes only being produced when a person gets very close or physically touches the plant.