



## Close Encounters Hat

Created by Leslie Birch



Last updated on 2018-08-22 03:42:23 PM UTC

## Guide Contents

Guide Contents	2
Overview	3
Tools and Supplies	4
Circuit Diagram	6
Prepare Sensor and Piezo Buzzer	7
Stitch Parts in Hat	8
Hat Power	9
Arduino Code	10
NeoPixel Überguide: Arduino Library Installation ( <a href="https://adafru.it/nBF">https://adafru.it/nBF</a> )	10
CircuitPython Code	13
Wear It!	16

## Overview

The Close Encounters hat is a nod to the incredible movie -- with lights and sounds that mimic the mother ship, all stowed under the brim. A photocell triggers the action for GEMMA, while the NeoPixels and piezo buzzer provide the show. Get ready to change up that next kiss or rock a dark movie theater.

This guide was written for the 'original' Gemma board, but can be done with either the original or M0 Gemma. We recommend the Gemma M0 as it is easier to use and is more compatible with modern computers!



## Tools and Supplies



If you order the [GEMMA Sensor Pack \(https://adafru.it/dMD\)](https://adafru.it/dMD), you'll have most everything you need. The one exception is the amount of NeoPixels. Since there are five lights/sounds for Close Encounters, you may want to purchase 6 additional NeoPixels to supplement the 4 that come with the kit. That way you'll have lights for the front and back of your hat.

This intermediate level wearables project requires some soldering and sewing. We recommend checking out the following guides before you begin:

- [Adafruit Guide To Excellent Soldering \(https://adafru.it/drl\)](https://adafru.it/drl)
- [Adafruit NeoPixel Überguide \(https://adafru.it/dhw\)](https://adafru.it/dhw)
- [Introducing GEMMA \(https://adafru.it/dgH\)](https://adafru.it/dgH) or [Introducing Gemma M0 guide \(https://adafru.it/yeq\)](https://adafru.it/yeq)

Make sure you have the Adafruit Arduino software installed for GEMMA, or hop on [Codebender \(https://adafru.it/dwj\)](https://adafru.it/dwj).

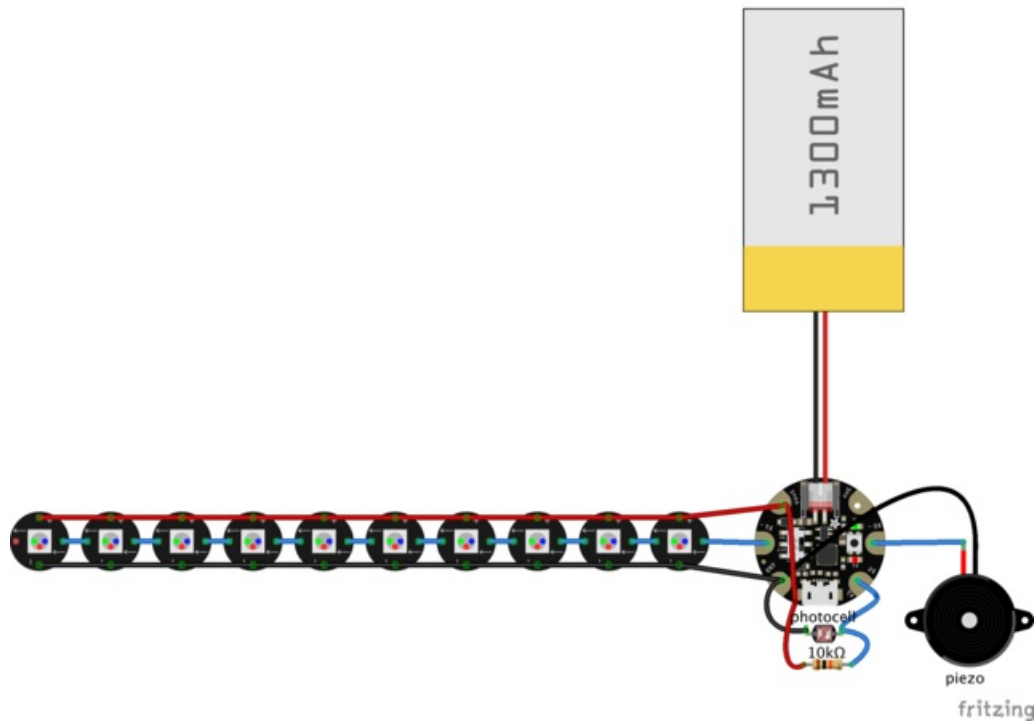
For this project you'll need:

- [Gemma M0 \(https://adafru.it/ytb\)](https://adafru.it/ytb) or [Original Gemma \(http://adafru.it/1222\)](http://adafru.it/1222) (M0 type is recommended!)
- [NeoPixels \(Total of 10\) \(https://adafru.it/dxP\)](https://adafru.it/dxP)
- [Photo Cell \(http://adafru.it/161\)](http://adafru.it/161)
- [Piezo Buzzer \(https://adafru.it/dyP\)](https://adafru.it/dyP)
- [500mAh LiPoly Battery \(https://adafru.it/dhK\)](https://adafru.it/dhK) and [Charger \(https://adafru.it/dpG\)](https://adafru.it/dpG)
- [10k Resistor \(https://adafru.it/xfm\)](https://adafru.it/xfm)
- [Conductive Thread \(https://adafru.it/dME\)](https://adafru.it/dME)
- [Alligator clips \(https://adafru.it/dxS\)](https://adafru.it/dxS)
- [Soldering iron \(http://adafru.it/180\)](http://adafru.it/180) and [Solder \(http://adafru.it/145\)](http://adafru.it/145)
- [Heat Shrink Tubing \(http://adafru.it/344\)](http://adafru.it/344)
- [Wire strippers \(http://adafru.it/527\)](http://adafru.it/527), and [Flush Diagonal Cutters \(https://adafru.it/dxQ\)](https://adafru.it/dxQ)

- (optional but very helpful) [Third Hand Tool \(https://adafru.it/dxR\)](https://adafru.it/dxR)
- One stainless steel snap
- [Sewing needle \(https://adafru.it/dgC\)](https://adafru.it/dgC) and Thread
- Scissors
- Adhesive Velcro
- Bucket Hat (we recommend getting one size up to save room for the battery)

## Circuit Diagram







Click to enlarge.









This diagram uses the original Gemma but you can also use the Gemma M0 with the exact same wiring!

- Gemma D0/A2 pin <--> Piezo (+)
- Gemma D1/A0 pin <--> NeoPixels
- Gemma D2/A1 pin <--> Photocell <--> 10k Resistor
- Gemma Vout <--> Resistor and NeoPixel (+)
- Gemma Gnd <--> Photocell, NeoPixel (-), Piezo (-)

## Prepare Sensor and Piezo Buzzer

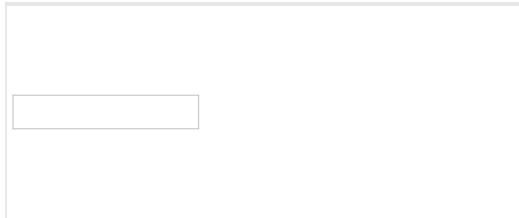

	<p>The photo cell will be your sensor to detect darkness and activate your hat. Start by bending the two wires slightly apart. Then, using a needle nose pliers, bend the legs at a 90 degree angle so that while the sensor points outward, the legs will sit nicely against your hat.</p>
	<p>Cut two small pieces of heat shrink tubing and place them on the legs of the photoresistor. Then use a heat gun or blow dryer on hottest setting to shrink the tube in place.</p>
	<p>Now, take your needle nose pliers and bend the tips of the legs into circles. Be gentle as the wires are fragile.</p>
	<p>Insert the legs of the photo cell into the grommet on the hat closest to the GEMMA. Gently wriggle them in one at a time, so they fit nicely against the inside of the hat.</p>
	<p>Let's move on to the piezo buzzer. This tiny piece will create the sounds for your hat. Start by stripping 1/4" off each wire. Do this very gently as the wires can break easily. Tin the wires with a bit of solder.</p>
	<p>Now solder each wire to the end of the snap pieces. It doesn't matter which snap piece you choose, as we are only using them as conductive pieces that allow us to stitch onto fabric. Okay, speaking of stitches, it's sewing time!</p>

## Stitch Parts in Hat

	<p>Start by marking a place for GEMMA and the 10 NeoPixels using chalk or ink. GEMMA is more undercover if you keep it behind a side seam under the brim. Mark a circle so it will be obvious where it goes. Evenly mark small dots for the NeoPixels around the brim.</p>
	<p>Make sure your NeoPixels are arranged so the arrows point in the same direction away from GEMMA's D1 pin. Using conductive thread, stitch the D1, Ground and Vout lines. You can follow the stitch lines on the hat to keep the rows tidy.</p>
	<p>It helps to use an invisible stitch, where you just pick up a thread of the fabric each time you make a stitch. That way stitches won't show on the front side. Notice the GEMMA is a little bit further from the edge to allow room for stitches around it.</p>
	<p>Make sure the photo cell is resting with the curled legs on top of the lining near the brim. On the GEMMA, stitch the GND pin to one leg and the D2 pin to the other leg on the sensor (it does not matter which leg, the sensor is not polar).</p>
	<p>Move the piezo gently in place so the snaps are not touching any conductive thread. It's okay for the wires to cross conductive thread because they are shielded. Stitch the black wire snap to D0 and the red wire snap to Vout.</p>
	<p>Next load some plain black thread into your needle and gently stitch the red and black wires to the hat, so the piezo won't move. Nice, now it's battery time!</p>



## Hat Power

	<p>Put a small piece of adhesive velcro onto the battery, making sure the wire won't have to twist going into the GEMMA. You may wish to tape the wire junctions to provide additional strain relief. Place the other side of the velcro inside the lining of the hat, directly above the GEMMA.</p>
	<p>The battery rests nicely in the pocket of the lining and the wire has an easy reach to the GEMMA. Now it's time to load the code.</p>

## Arduino Code

The Arduino code presented below works equally well on all versions of GEMMA: v1, v2 and M0. But if you have an M0 board, consider using the CircuitPython code on the next page of this guide, no Arduino IDE required!

If this is your first time using GEMMA, work through the [Introducing G \(https://adafru.it/cHH\)](https://adafru.it/cHH)EMMA (<https://adafru.it/cHH>) or [Introducing Gemma M0 guide \(https://adafru.it/yeq\)](https://adafru.it/yeq) first; you need to customize some settings in the Arduino IDE. Once you have it up and running (test the 'blink' sketch), then follow the instructions on the following page for installing the NeoPixel library:

### NeoPixel Überguide: Arduino Library Installation (<https://adafru.it/nBF>)

Plug in your circuit and load up the sketch below:

```
/*
Close Encounters hat with 10 neopixels by Leslie Birch for Adafruit Industries.
Notes play with each corresponding light.
Based on code by Becky Stern, Mike Barela and T Main for Adafruit Industries
http://learn.adafruit.com/adafruit-trinket-modded-stuffed-animal/animal-sounds
*/

const int speaker = 0;      // the number of the speaker
#define PHOTOCELL 1        // cDS photocell on A1

// this section is Close Encounters Sounds
#define toneC    1046.50
#define toneG    1567.98
#define tonec    2093.00
#define toned    2349.32
#define tonee    2637.02
#define tonep    0

#define darkness_min 512 // analog 0-1024 0-512 (light, 513-1023 dark)

long vel = 20000;

// This section is NeoPixel Variables

#include <Adafruit_NeoPixel.h>

#define PIN 1

// Parameter 1 = number of pixels in strip
// Parameter 2 = pin number (most are valid)
// Parameter 3 = pixel type flags, add together as needed:
//   NEO_KHZ800  800 KHz bitstream (most NeoPixel products w/WS2812 LEDs)
//   NEO_KHZ400  400 KHz (classic 'v1' (not v2) FLORA pixels, WS2811 drivers)
//   NEO_GRB     Pixels are wired for GRB bitstream (most NeoPixel products)
//   NEO_RGB     Pixels are wired for RGB bitstream (v1 FLORA pixels, not v2)
//Adafruit_NeoPixel strip = Adafruit_NeoPixel(10, 1, NEO_GRB + NEO_KHZ800);
Adafruit_NeoPixel strip = Adafruit_NeoPixel(10, 1);

void setup() {
```

```

pinMode(speaker, OUTPUT);

Serial.println("setup");

//This is for Neopixel Setup
strip.begin();
strip.show(); // Initialize all pixels to 'off'
}

void loop()
{
  // turn lights and audio on when dark
  // less than 50% light on analog pin
  if ( analogRead(PHOTOCELL) > darkness_min ) {

    alien(); // Close Encounters Loop

  }

}

// the sound producing function

void beep (unsigned char speakerPin, int frequencyInHertz, long timeInMilliseconds)
{ // http://web.media.mit.edu/~leah/LilyPad/07_sound_code.html
int x;
long delayAmount = (long)(1000000/frequencyInHertz);
long loopTime = (long)((timeInMilliseconds*1000)/(delayAmount*2));
for (x=0;x<loopTime;x++)
{
digitalWrite(speakerPin,HIGH);
delayMicroseconds(delayAmount);
digitalWrite(speakerPin,LOW);
delayMicroseconds(delayAmount);
}
}

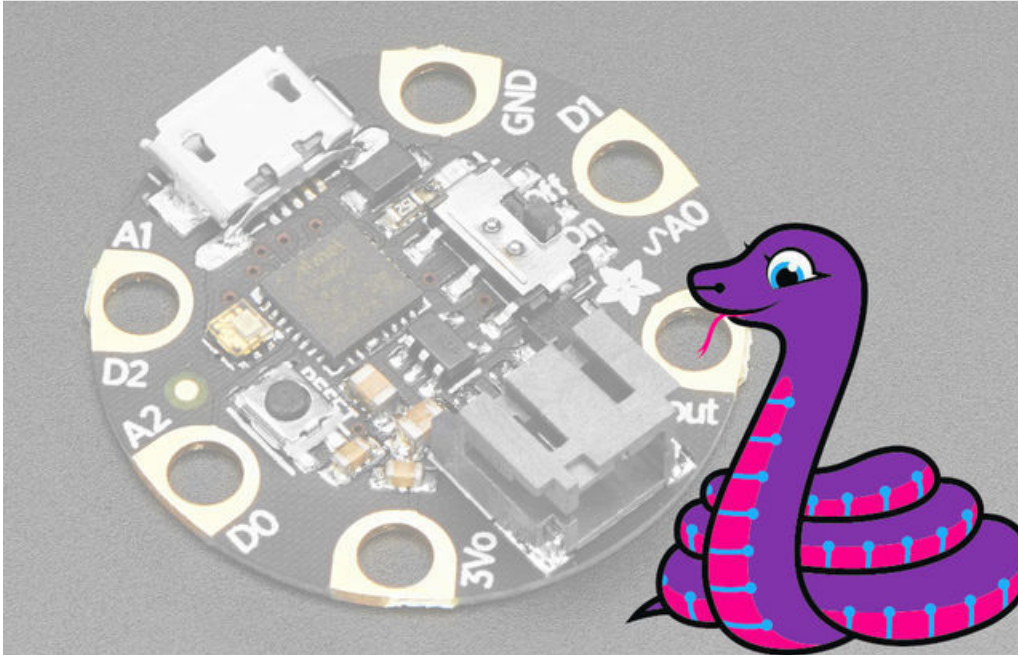
// Generate the Close Encounters LEDs
void alien() {

  strip.setBrightness(64);
  strip.setPixelColor(8, 255, 255, 0); //yellow front
  strip.setPixelColor(3, 255, 255, 0); //yellow back
  strip.show();
  beep(speaker,toned,1000);
  delay(25);
  strip.setPixelColor(8, 0, 0, 0); //clear front
  strip.setPixelColor(3, 0, 0, 0); //clear back
  strip.show();
  delay(25);
  strip.setPixelColor(7, 255, 0, 255); //pink front
  strip.setPixelColor(2, 255, 0, 255); //pink back
  strip.show();
  beep(speaker,toned,1000);
  delay(25);
  strip.setPixelColor(7, 0, 0, 0); //clear front
  strip.setPixelColor(2, 0, 0, 0); //clear back
  strip.show();
  delay(25);
}

```

```
strip.setPixelColor(4, 128, 255, 0); //green front
strip.setPixelColor(9, 128, 255, 0); //green back
strip.show();
beep(speaker, toneC, 1000);
delay(25);
strip.setPixelColor(4, 0, 0, 0); //clear front
strip.setPixelColor(9, 0, 0, 0); //clear back
strip.show();
delay(25);
strip.setPixelColor(5, 0, 0, 255); //blue front
strip.setPixelColor(0, 0, 0, 255); //blue back
strip.show();
beep(speaker, toneC, 1000);
delay(75);
strip.setPixelColor(5, 0, 0, 0); //clear front
strip.setPixelColor(0, 0, 0, 0); //clear back
strip.show();
delay(100);
strip.setPixelColor(6, 255, 0, 0); //red front
strip.setPixelColor(1, 255, 0, 0); //red back
strip.show();
beep(speaker, toneG, 1000);
delay(100);
strip.setPixelColor(6, 0, 0, 0); //clear front
strip.setPixelColor(1, 0, 0, 0); //clear back
strip.show();
delay(100);
}
```

## CircuitPython Code



These directions are specific to the “M0” GEMMA board. The original GEMMA with an 8-bit AVR microcontroller doesn’t run CircuitPython...for those boards, use the Arduino sketch on the “Arduino code” page of this guide.

Below is CircuitPython code that works similarly (though not exactly the same) as the Arduino sketch shown on a prior page. To use this, plug the GEMMA M0 into USB...it should show up on your computer as a small **flash drive**...then edit the file “main.py” with your text editor of choice. Select and copy the code below and paste it into that file, **entirely replacing its contents** (don’t mix it in with lingering bits of old code). When you save the file, the code should **start running almost immediately** (if not, see notes at the bottom of this page).

If GEMMA M0 doesn’t show up as a drive, follow the [GEMMA M0 guide](#) link above to prepare the board for CircuitPython.

```
# Close Encounters Hat with 10 NeoPixels
# ported from Leslie Birch's Arduino to CircuitPython
#
# Photocell voltage divider center wire to GPIO #2 (analog 1)
# and output tone to GPIO #0 (digital 0)

import time

import analogio
import board
import neopixel
import simpleio

# Initialize input/output pins
photocell_pin = board.A1 # cds photocell connected to this ANALOG pin
speaker_pin = board.D0 # speaker is connected to this DIGITAL pin
pixpin = board.D1 # pin where NeoPixels are connected
numpix = 10 # number of neopixels`
```

```

darkness_min = (2 ** 16 / 2) # more dark than light > 32k out of 64k
photocell = analogio.AnalogIn(photocell_pin)
strip = neopixel.NeoPixel(pixpin, numpix, brightness=.4)

# this section is Close Encounters Sounds
toned = 294
tonee = 330
tonec = 262
toneC = 130
toneg = 392

def alien():
    strip[8] = (255, 255, 0) # yellow front
    strip[3] = (255, 255, 0) # yellow back
    simpleio.tone(speaker_pin, toned, 1) # play tone for 1 second

    time.sleep(.025)

    strip[8] = (0, 0, 0) # clear front
    strip[3] = (0, 0, 0) # clear back

    time.sleep(.025)

    strip[7] = (255, 0, 255) # pink front
    strip[2] = (255, 0, 255) # pink back
    simpleio.tone(speaker_pin, tonee, 1) # play tone for 1 second

    time.sleep(.025)

    strip[7] = (0, 0, 0) # clear front
    strip[2] = (0, 0, 0) # clear back

    time.sleep(.025)

    strip[4] = (128, 255, 0) # green front
    strip[9] = (128, 255, 0) # green back
    simpleio.tone(speaker_pin, tonec, 1) # play tone for 1 second

    time.sleep(.025)

    strip[4] = (0, 0, 0) # clear front
    strip[9] = (0, 0, 0) # clear back

    time.sleep(.025)

    strip[5] = (0, 0, 255) # blue front
    strip[0] = (0, 0, 255) # blue back
    simpleio.tone(speaker_pin, toneC, 1) # play tone for 1 second

    time.sleep(.075)

    strip[5] = (0, 0, 0) # clear front
    strip[0] = (0, 0, 0) # clear back

    time.sleep(.1)

    strip[6] = (255, 0, 0) # red front
    strip[1] = (255, 0, 0) # red back

```

```
simpleio.tone(speaker_pin, toneg, 1) # play tone for 1 second

time.sleep(.1)

strip[6] = (0, 0, 0) # clear front
strip[1] = (0, 0, 0) # clear back

time.sleep(.1)

# Loop forever...
while True:

    # turn lights and audio on when dark
    # (less than 50% light on analog pin)
    if photocell.value > darkness_min:
        alien() # close Encounters Loop
```

This code requires the **neopixel.mpy** and **simpleio.mpy** libraries. A factory-fresh board will have `neopixel.py` installed. If you've just reloaded the board with CircuitPython, create the "lib" directory and obtain [neopixel.mpy and simpleio.mpy from the latest Adafruit\\_CircuitPython\\_Bundle \(https://adafru.it/y8E\)](#).

Wear It!



If you have done everything correctly, your hat should light up and make the famous notes when it is dark. You can also trigger this reaction yourself by cupping your hand around the grommet that holds the photo cell. Have fun and let everyone know that you are the biggest Spielberg fan around!