**Animatronic EVE**

Version 2.0

*Mike Owen*

A white robot with a black circle

Description automatically generated

# Parts required

* *3 x MG90S or SG90 Servos*
* *1 x MG946-180 or similar*
* *1 x DFPlayer Mini MP3 player*
* *1 x Arduino Nano ATMEGA328P*
* *1 x Polulu Maestro 6 channel*
* *2 x* [*Adafruit 1.5” OLED Screen*](https://www.adafruit.com/product/1431)
* *1 x m4 or m0 Teensy (not AVR or 32u4 boards) – I used an M4 Express*
* *1 x* [*USB pinboard type C*](https://www.aliexpress.com/item/1005001984554555.html)
* *1 x* [*Touch Sensor*](https://www.aliexpress.com/item/32451378348.html)
* *1 x* [*40mm Speaker*](https://www.aliexpress.com/item/32836767822.html)
* *1m WS2811 RGB LED 144leds/m*
* *a Paperclip*
* *Primer filler paint*
* *Heat shrink*
* *Wet & Dry Sandpaper (240/400/800/1200/2000)*
* *Gloss White Spray Paint (with 2k Clearcoat ideally)*
* *Window Tint film*
* *Cyanoacrylate (super glue)*
* *Double sided tape or Hot Glue Gun*
* *M2 & M3 Screws (assorted lengths)*
* *Soldering Iron & Solder*
* *Wire (assorted lengths – Servo extension & female to female jumper wires desirable)*

# Print Instructions

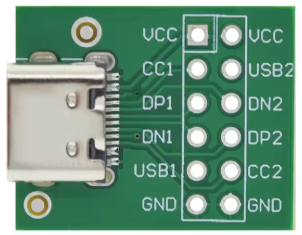
* Body - 25% infill, upside down orientation, tree supports with additional supporting where the base stand slots into it.
* Base Top – 45% infill, tree support
* Base Bottom – 15% infill, support recommended
* Head – at least 25% infill, tree support
* Glass cover – transparent Resin
* Light bar – anything dark/opaque to limit light bleed
* Arms – 25% infill, upright orientation with tree support

# Assembly

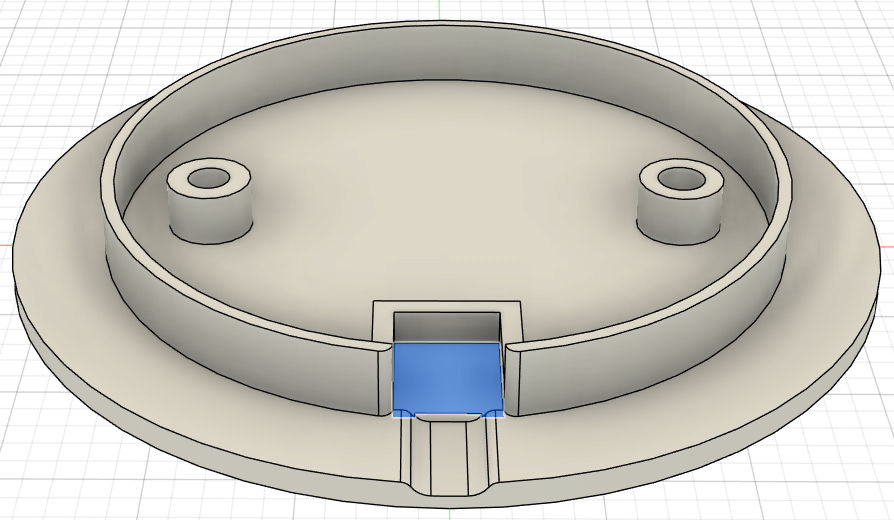
There aren’t many parts so assembly is fairly straight forward, the electronics tend to be the tricker part.

## Base - Bottom

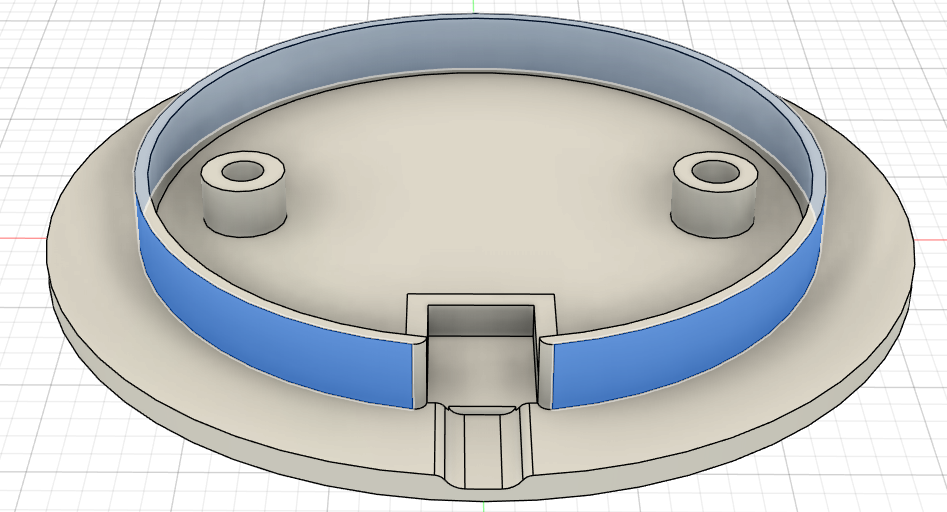
Solder Header pins to the USB board to both VCC and Ground pins.1 Set will be used to power the LEDs and touch sensor in the base and the other everything else from body upwards.



Insert the USB board into the slot in the base as below. Secure in place using the USB cover and screwing in 2 x M3 screws



Cut a length of WS2811 strip to the same length as the outer circumference of the inner ring of the bottom base. Using a female to female jumper cut in half and solder in a 3rd wire in a Y shape. Cover solder joint with heat shrink. This should give you one header for the USB board, one header for the touch sensor and a spare wire which can be soldered to the LED strip. Repeat so you have one for both +ve and -ve. Solder a separate length (~30cm) of wire to data line of the LED. This will later be pulled up through the stand and body and connected up to the controller in the head. Stick the LED to the rim as show below with either double sided tape or using a hot glue gun.

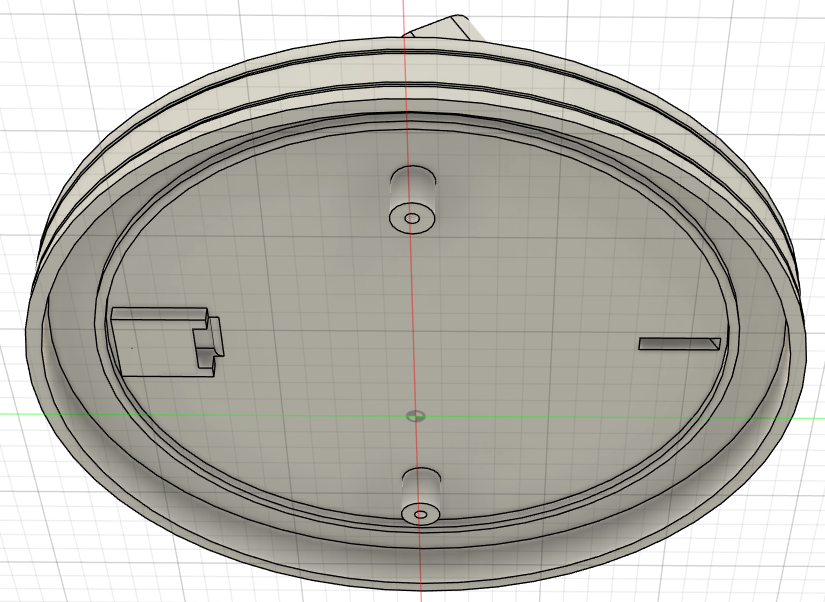


## Base – Top

Solder on header pins with the pins pointing up from this side so that the round sensor side can be flush with the underside of the Base top. Attach the -ve(GND) and +ve(VCC) Y shaped cables created earlier and connect a ~30cm female to female jumper to the I/O pin, this will be routed up through the base/body and connected to the controller.

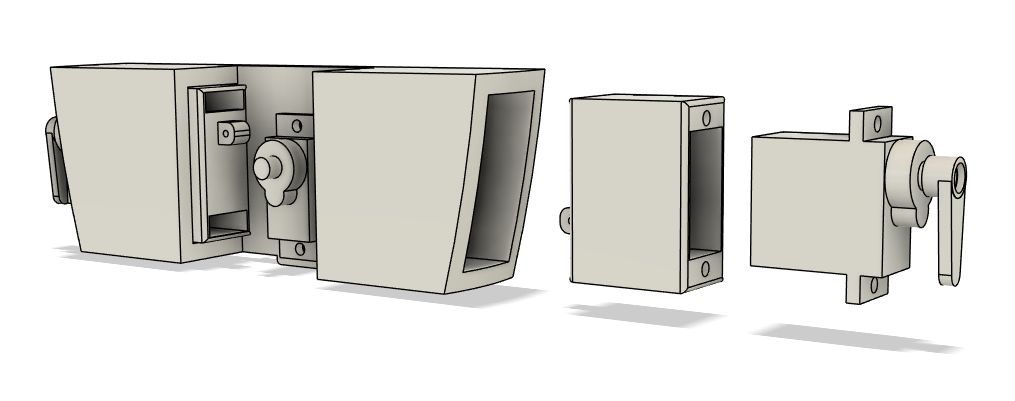


Insert touch sensor into the slot shown below, sensor side facing upwards towards EVE. Fit the Bottom and Top Base pieces together aligning with the wholes on the base and screw in from the bottom with M3 Screws



## Arm – Servo Mechanism

Insert the two MG90S servos into the Servo Slides pushing the servo lead through the hole on the back on the same side the cable exists the servo. Note the servo slides one has the connector hole at the top and one at the bottom. Servo’s are orientated as shown below with the servo gear at the top on both sides. The middle servo is mounted from the rear (pic shows it from the front).



Bend a paperclip or other similar sized wire as below. This is a rough guide adjust length as necessary. With the servo set to fully retracted and servo slides in position where the arms are flush with the side of the body measure between the hole in the servo and the servo slide.

3 mm

3 mm

4 mm

4 mm

18 mm

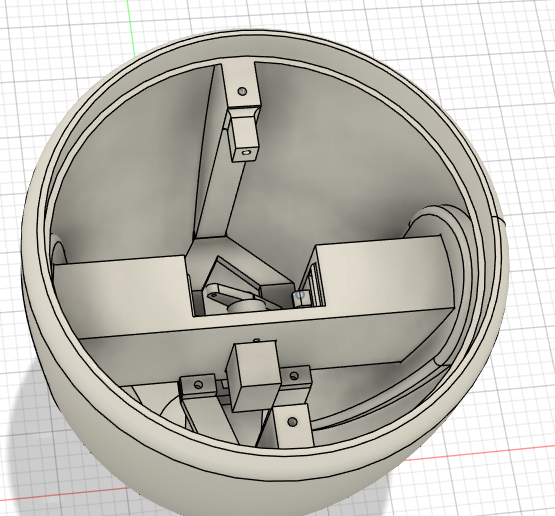
Install the middle servo with the screws provided and thread the paperclip connector through the servo horn and servo slide as shown.

A metal box with a metal handle

Description automatically generated with medium confidence

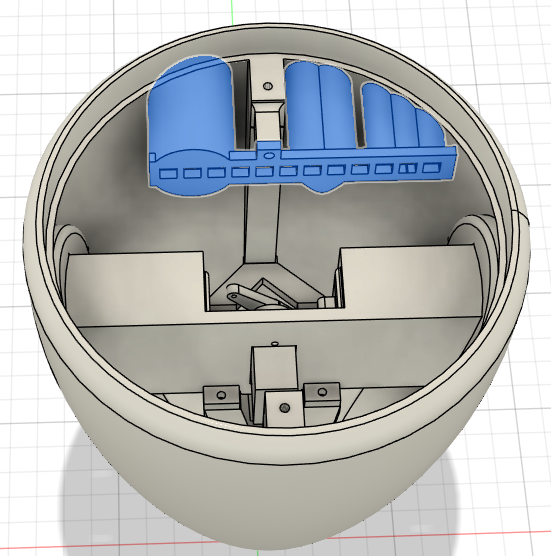
## Body Assembly

The Servo Mechanism can be press fit into the body via the cut out on the top. Insert in one arm slot first and then gentle press down to slide into the second and seat on the support. Use M3 screws to secure in place.

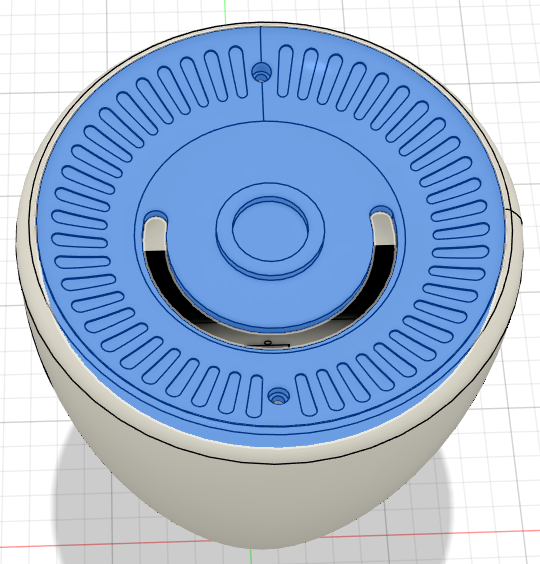


Insert the lightbar into place and use an m3 screw to hold in place. Grab a 15cm Servo extension lead, cut off the female end and solder to the second LED strip of 12 LEDS, check the arrow direction on the LEDs to make sure you solder to the correct end. Insert LED’s into the light bar, use a small amount of hot glue to fix in place if required.

Note: LED’s may not be visible if painting so consider this part optional



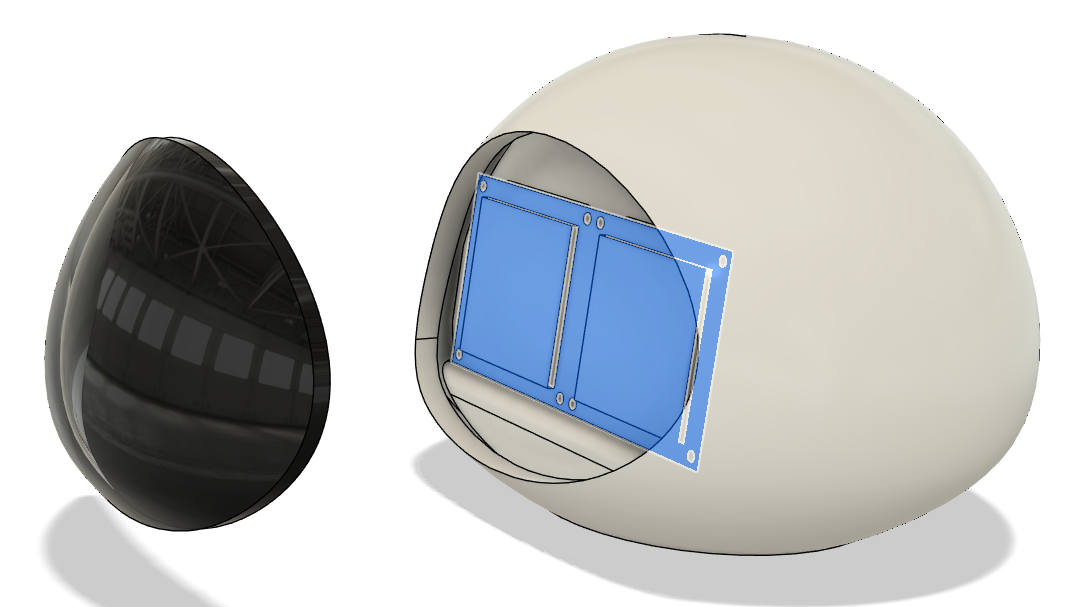
Before proceeding with the next piece complete wiring for the base, the light bar and then pull all the servo leads up through the neck piece before press fitting the neck and securing with 2 x M3 screws. A round Servo horn for the MG946 can be hot glued into the centre piece. The weight of the head will hold it in place, so the horn doesn’t need to be screwed into Servo.



## Head

Do this part once you have your electronics sorted given the inside of the head is going to get cramped. Attach the two OLED screens to the head using M2 screws. The Header pins are at the top of the OLED and should have room to push the jumper cables through the given cut out.

For the Glass cover unless you have tinted your resin you can either spray a tinted clearcoat over it or use Car Window Tinting film. Follow the tinting instructions to make sure no air bubbles show.

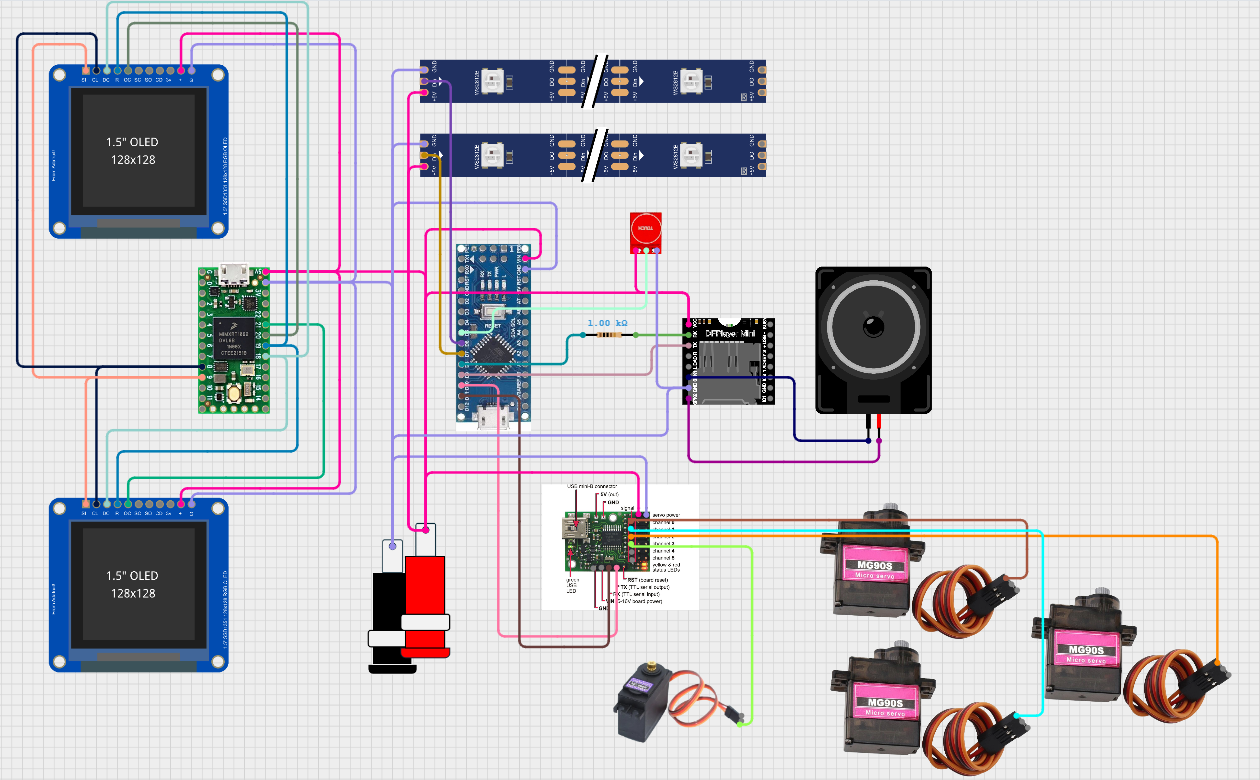


The MG946 is installed with servo gear in the middle of the head. The Servo can be screwed into the Head Base either from the top or bottom depending on your preference for head height above the body. Use M3 Screws to connect the base to the head



# Wiring

This could really do with a custom PCB being made to tidy this up as the end result looks like this.



**OLED Wiring to Teensy (M4 Express)**

* A computer chip with many different colored numbers

  Description automatically generated with medium confidenceSI to Pin MOSI
* CL to Pin SCK
* DC to Pin D7
* R to Pin D9
* OC to Pin 10 or 11 (Left/Right Eye)
* + to +ve (5v)
* G to -ve (GND)

**Arduino Nano Wiring**

* DFPlayer RX to ~1kohm Resistor to Pin D8
* DFPlayer TX to Pin D9
* Maestro TX to Pin D10
* Maestro RX to Pin D11
* Touch Sensor I/O to D5
* Base LED Strip to Pin D6
* Chest LED Strip to Pin D4

# Code

<https://github.com/ShimmerNZ/EVE-2.0>

### Pololu Maestro

* Channel 0 – Arm Extender
* Channel 1 - Left Arm
* Channel 2 – Right Arm
* Channel 3 – Neck

The below should be the configuration when the arms are down and inside the bottom:

A screenshot of a computer

Description automatically generated

Copy the contents of Code.txt from the Github repo to the Script tab and click “Apply Settings”. You can create your own sequence just ensure that it always finishes with the arms fully retracted. To trigger via the Arduino on the script page make sure for the Sequence subroutines they are ended with “quit” instead of “return” (the frame subroutines should be “return”).

### Teensy 4.0 Express

The code used is based off the Uncanny Eyes project which has more extensive documentation available here: <https://learn.adafruit.com/animated-electronic-eyes/overview>

The primary changes are to the Sclera and Eyelid graphics. You can modify those files to adjust how Eve’s eye shape and blinking looks. To do this open a terminal navigate to the convert folder and execute the below:

python tablegen.py EveEye/sclera.png EveEye/iris.png EveEye/lid-upper-symmetrical.png EveEye/lid-lower-symmetrical.png EveEye/lid-upper.png EveEye/lid-lower.png 2 > EveEye.h

This will produce a file called EveEye.h this will need to be copied to ..\EveTeensy\Eve\Graphics

If you have problems with compiling, open the EveEye.h file in a text editor (e.g. Notepad++) and change the encoding to ANSI or UTF-8 as it will error if UTF-16

### Arduino Nano

The Body LED code was never completed as I painted mine which meant I couldn’t see the LED shine through. There is however a base light chasing effect with white light that is included.