CRASH WORKS

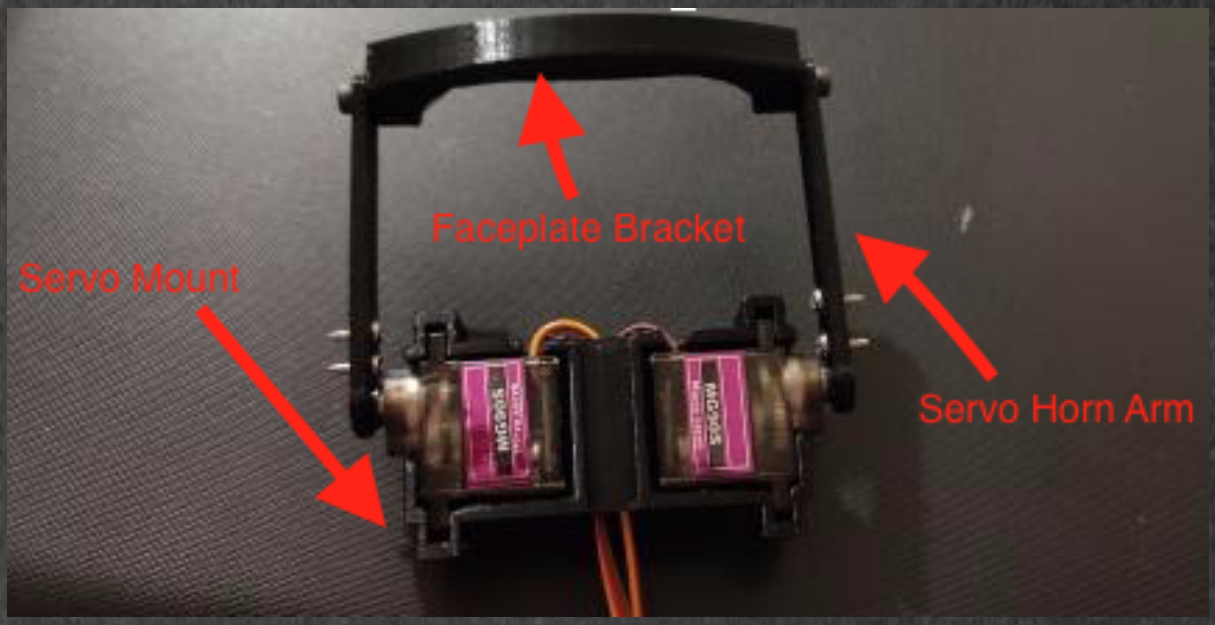
Iron Man Faceplate Tutorial

1. **Print out all of the items from the Tower Pro or Non-Tower Pro Folders Shown in image. If you are unsure which servo you have, the Tower Pro servos have their "wings" further down the body than the Clone variants. Note: you may need to trim extra area around the cable hole in the servo mount – The cable location seems to differ on some brands of servo just clip away some extra with flush cutters.**
   ******Note: SG90's are plastic geared & not suitable for this project. ******

2. **Your servos are supplied with arm (horn) options, find the one pictured below.**

3. **Trim off one side of the arm (horn) using flush cutters or nail clippers.**

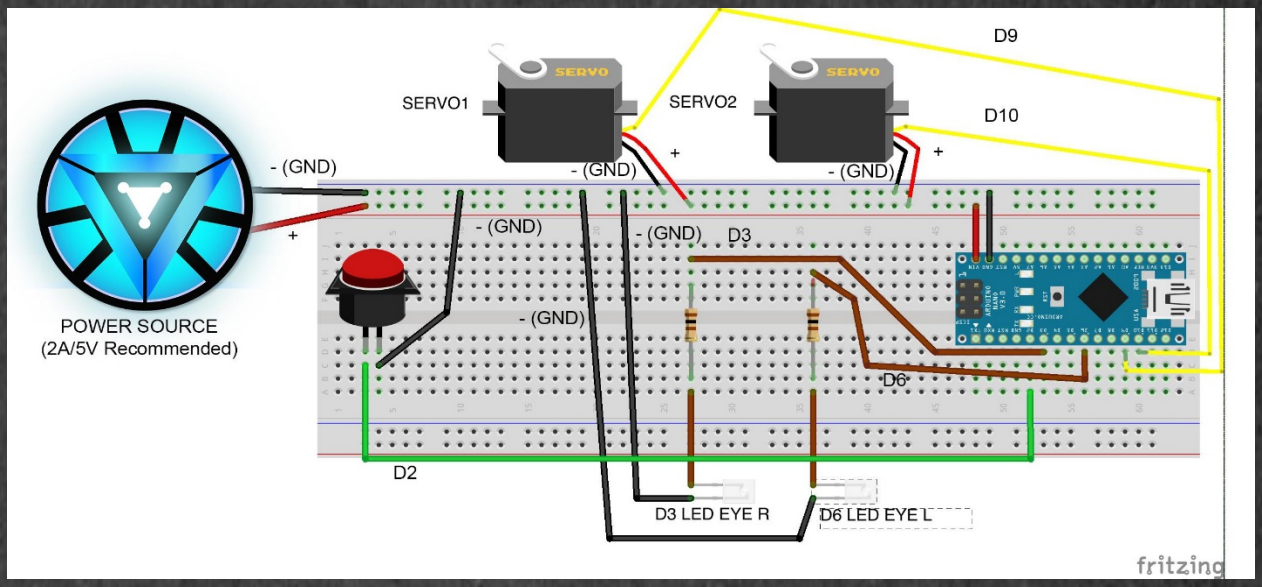
4. **Place your servos into the servo mount and attach with self-tapping screws. ***Note: Wiring comes out of the servo mount to the front of the helmet.*** Attach the servo horns into the holes on the servo horn arms._ Using the horn screws that came with your servos, screw through the horn into the servo horn arms. These screws are self-tapping and will make their own hole through the 3D printed part. Once you have screwed both screws in, remove them and screw them back through the 3D printed part first (the direction shown in the image). ****Note: Sand smooth any areas that are 3d printed and rotation is needed e.g servo horn arms to faceplate bracket. ******



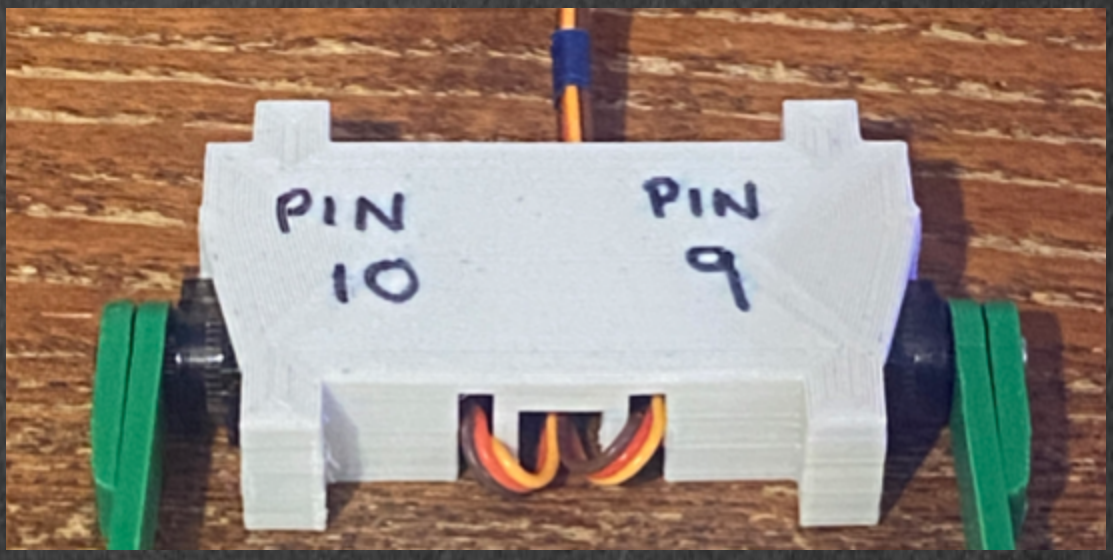
5. **Clip the extra thread off with flush cutters.**

6. **Mount the servo horn arms onto the servos as pictured above using the screws supplied with your servos.**

7. **Connect your wiring as per the image below into your breadboard.**



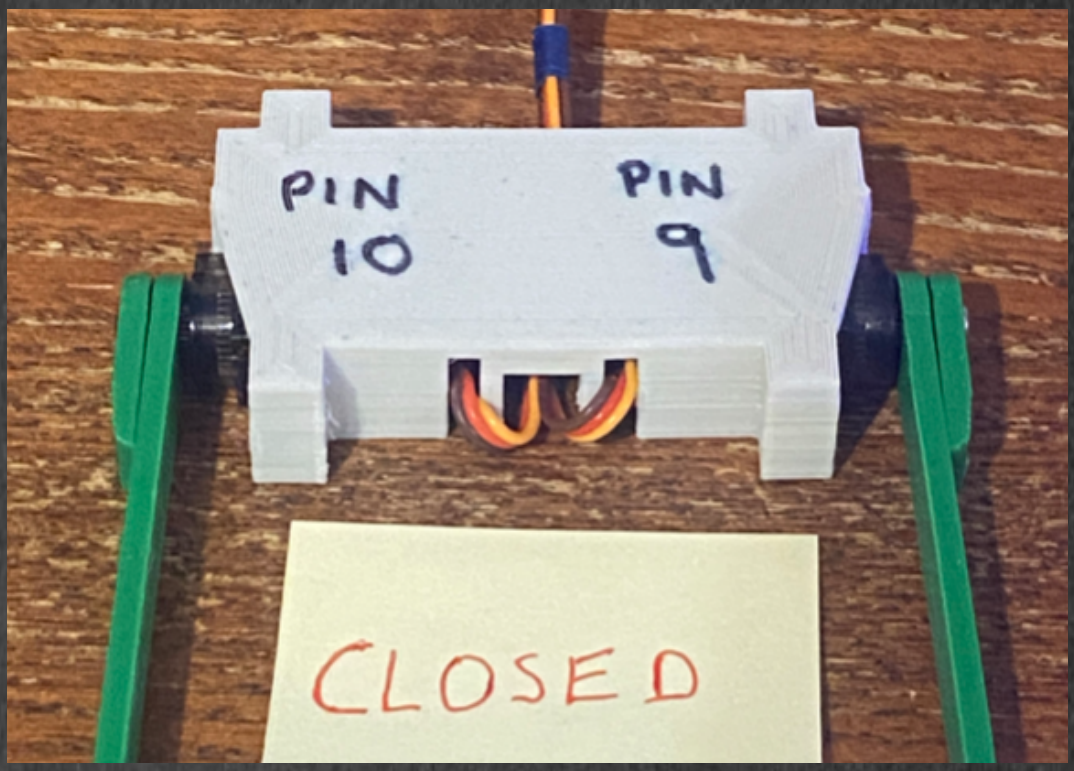
8. **Your servos should be connected to pin 9 and 10 that attach to the servos as labelled in the image below.**



9. **Upload the code to your Arduino. Click the link Iron Man Arduino Servo Code (https://github.com/crashworks3d/Iron_Man_Servo) to go to our GitHub site to download the code and for additional information.** ****** Note you will need the "VarSpeedServo" library installed in your Arduino IDE, since this code uses Servo speed control ******

10. **Connect the power to your Arduino and test the code. Your servos will automatically move to the "faceplate closed" position. Pressing the switch will cycle the servos open and closed with the LED's (eyes) turning on with the servos in the closed position – If the LEDs light up with the servos in the up (faceplate open) position your servos are connected incorrectly. Disconnect your power. Swap servo pins 9 and 10 and re-power.** ******Note: Your servos will move in opposite directions via the coding. When housed in the servo mount, they will move correctly. ******

Setting the Servo Angles



11. **Place your servos on a flat surface. Using your switch set the servo arms to the down (faceplate closed LEDs on) position as in the image below.**

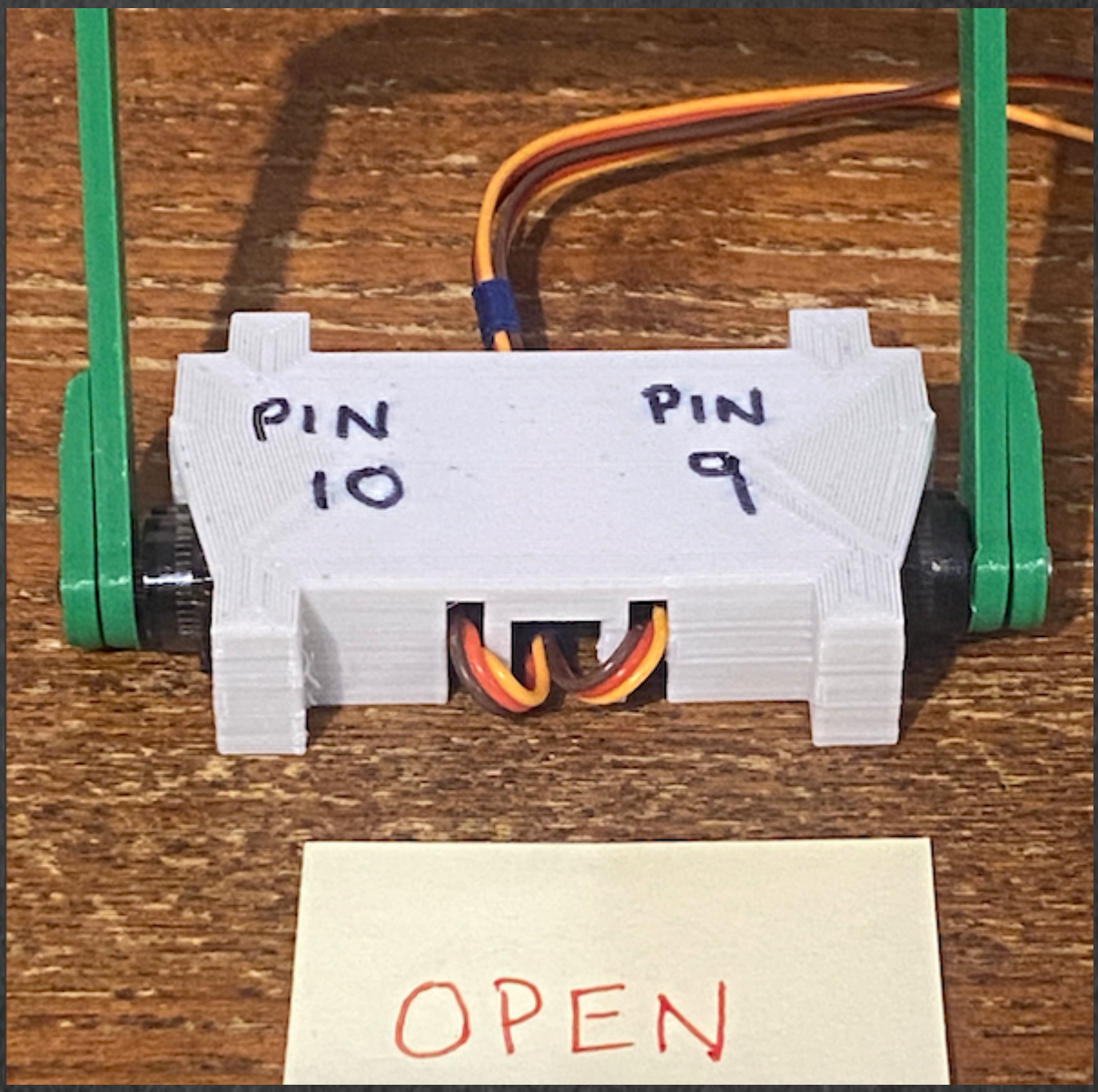
12. **Adjust your closed angles in the code as shown below at this point you will be using the *servo1_ClosePos = 160 & servo2_ClosePos = 20* until the arms are parallel with each other.** ******Note: maximum angles for the servos are 180° and 0°. When you have changed the angles in the code you will need to re-upload to your Arduino and re-power. (see example code block below)******

// In Dual Servo Configuration the servos move in opposing directions, so the angles of the servos will be opposite to each other.

// Normal Servo range is 0° ~ 180°, for initial setup the range has been adjusted to 20° ~ 160°, this allows for a 20° adjustment at both ends of the servo range.

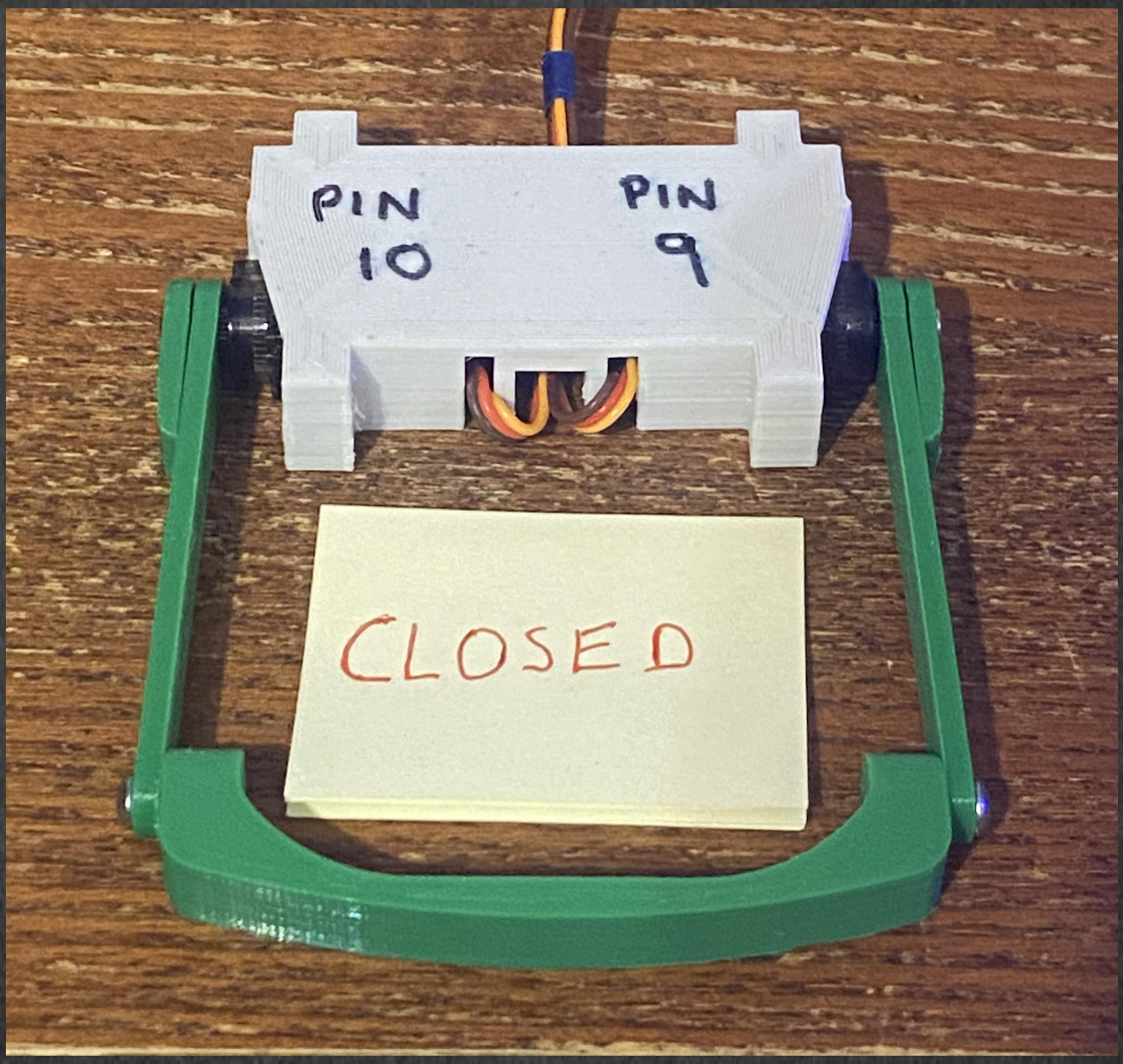
```
const int servo1_OpenPos = 20; // set the open position of servo 1
const int servo2_OpenPos = 160; // set the open position of servo 2
const int servo1_ClosePos = 160; // set the closed position of servo 1
const int servo2_ClosePos = 20; // set the closed position of servo 2
```

13. Once your closed angles are adjusted using the switch place the servos in the open (LEDs off) position see image below. Follow the previous steps to ensure your servo arms are also parallel in this position using the code *servo1_OpenPos = 20 & servo2_OpenPos = 160*.



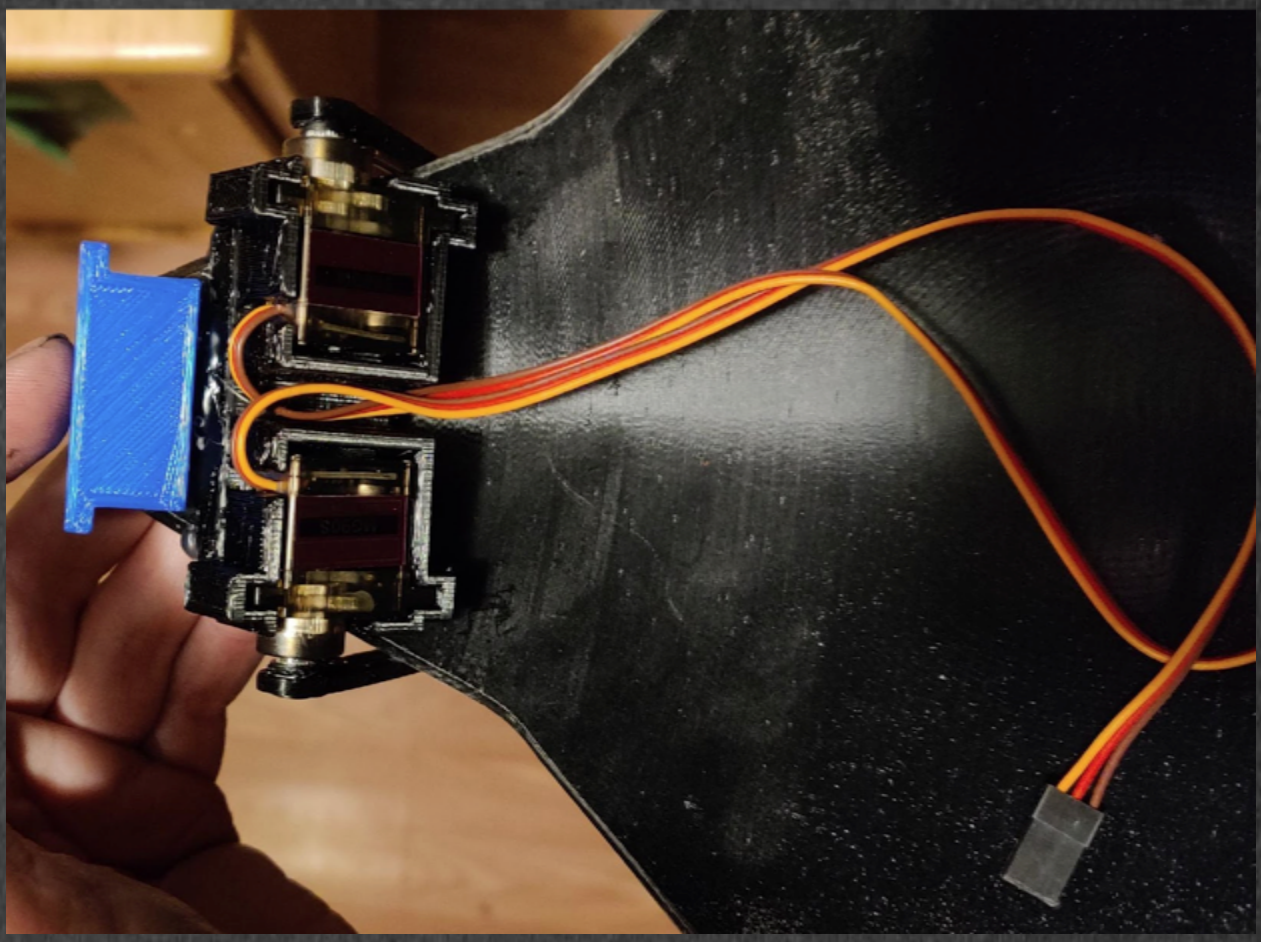
14. **Once your servo arms are parallel at both the open and closed positions attach the faceplate bracket to the arms with M3 screws. ****Note: The faceplate bracket is already threaded. **** Tighten the screws until the faceplate bracket will not move and then loosen them one turn so that the faceplate bracket can rotate freely.**

    Use the switch to open and close the servos to further ensure the arms are parallel with the faceplate bracket attached.



Attaching the Mechanicals

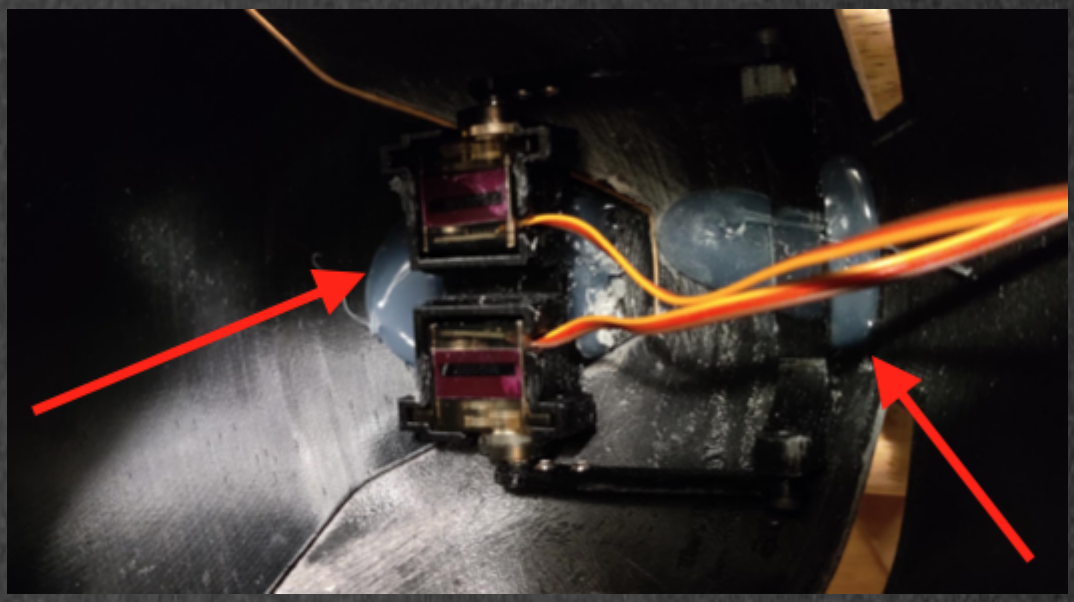
15. **(Mk85) place the servo alignment tool on the top of the widows peak, hold in place and butt the servo mount up to it. Ensure your servo mount is central (left to right) on the widow's peak. *** Note: *Make sure the Servos are set to the maximum "upward arm position" to ensure clearance on the sides of the Widow's peak when they are fully extended upward.* *****



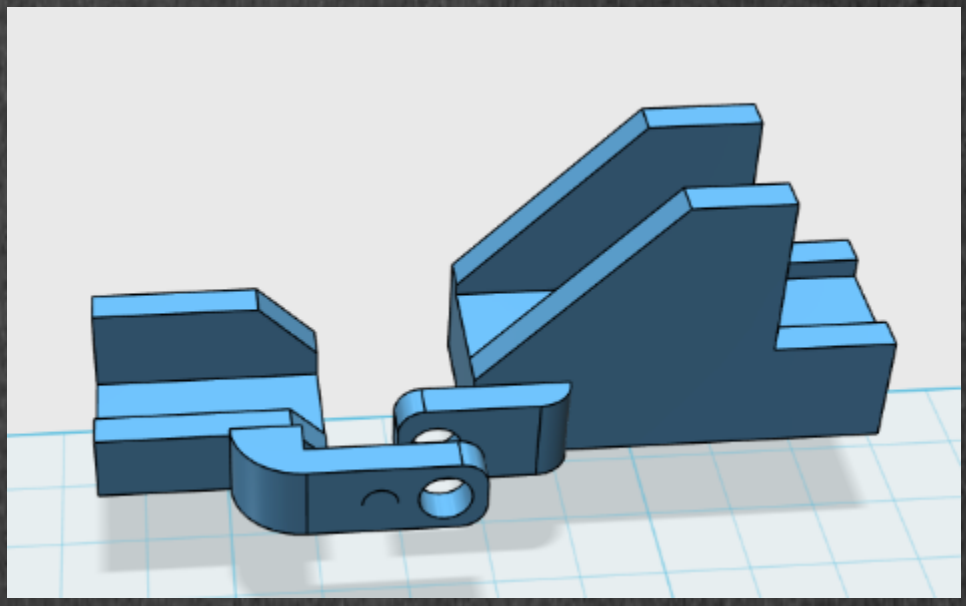
16. **Hot glue the back of the servo mount (side that is away from the tool) in place. Once the glue is holding, then remove the tool and glue the front of the servo mount in place. ****Note: Once this step is completed, make sure to power the servos and run the code so they move to the fully downward position, before proceeding to the next step.* *****

17. **Put the faceplate on, and using pieces of credit card sized plastic shim between the helmet and the faceplate. Hold the faceplate and helmet together with painters (masking) tape.**

18. **With the Servo arms down, and the Faceplate bracket attached to the arms, make sure it properly contacts the faceplate above the “eyes”. You may need to adjust your downward angles in the Arduino code to find the proper position where the arms are as far down as they can go and properly contact the faceplate. When properly set, the faceplate should remain closed and flush with the helmet. Then hot glue the faceplate bracket onto the faceplate – Again we'll be cleaning this up later.**



19. **Assemble the two helper arm tool pieces loosely with an m3 nut and bolt through the eyelets.**



20. ***Open the faceplate***. **Place the helper arm alignment tool on the section of the helmet (not faceplate) shown in the below picture. Tighten but do not overtighten the nut setting the angle. The arm mount will be placed in the square cut-out of the helper arm alignment tool also pictured below.**

21. Hot glue the arm mount in place using the helper arm alignment tool as a guide.

22. Remove the helper arm alignment tool and transfer to the same location on the opposite side of the helmet. Repeat steps 20 & 21 to adhere your second arm mount in place.

23. Attach the helper arms loosely to the arm mounts that are now attached to the helmet using M3 screws.

24. Attach the other two mounts (which will be later adhered to the faceplate) to the holes on the ends of the helper arms with M3 screws.

25. Locate the area you are going to be attaching the mounts to on the faceplate. There is a lip on the inside of the faceplate that the mounts will butt up to. ****Note: This is easier to locate now, before you glue the faceplate mounts on. ****

26. *Close the faceplate.*

27. **Put some hot glue on the area that the mount is going to attach to (on the faceplate), swing the arm & mount over the glue until it butts up against the inside lip, and push down to glue in place.**

28. **Repeat step 27 on the other side of the helmet/faceplate so that both mounts are attached to the faceplate.**

29. **Tighten all 4 screws on the helper arms and remove one turn to ensure the helper arms can move (rotate) freely.**

30. **Using the switch test your code. Your faceplate should now move up and down smoothly. If your faceplate is catching on any areas of the helmet locate where it is binding and sand away some of the plastic in that area.**
******Remember it's easier to take a small amount off, and not so easy to replace it. ******

31. **Once you are fully satisfied with the movement of your faceplate you can go back and weld (solder or 3D pen) the areas around where you hot glued.**
******Note: Do not weld or use the 3D pen on areas that have hot glue on them - Secure pieces that are unglued first. ******

32. **When you have secured all of the hot glued pieces to the helmet you can remove the remaining visible hot glue with isopropyl alcohol.**

33. **With the hot glue removed you can now secure the pieces that were under the hot glue with solder weld or 3D pen.**

34. **Using your switch test your code.**

Adjusting your open and close angles for fitment

35. **Now that you have your faceplate attached you may want to adjust your faceplate open and/or close range. Change the value of the angles in the code in 1° or 2° steps until you reach the fitment you desire.**

```
// In Dual Servo Configuration the servos move in opposing directions, so the angles of the servos will be opposite to each other.

// Normal Servo range is 0° ~ 180°, for initial setup the range has been adjusted to 20° ~ 160°, this allows for a 20° adjustment at both ends of the servo range.
const int servo1_OpenPos = 20; // set the open position of servo 1
const int servo2_OpenPos = 160; // set the open position of servo 2
const int servo1_ClosePos = 160; // set the closed position of servo 1
const int servo2_ClosePos = 20; // set the closed position of servo 2
```

To finish your helmet, install your choice of eyes and route the wiring as desired. Be sure to route your wiring in a way that will not affect the motion of the moving parts you have just installed. Choose a location to mount your Arduino and electronic components if you wish inside the helmet or keep them free for external use.

Troubleshooting

- **If the closing motion of your faceplate is too quick (i.e. crashing into the jaw) reduce close speed in the coding.**

  // Declare variables for servo control

  const int servoCloseSpeed = 100; // set the speed of the servo close function

  (Speed value 0 = slowest 255 = maximum)

- **If the servos are struggling to lift the faceplate ensure you are providing enough power and not powering them from the Arduino. A 9v battery is NOT capable of delivering the amperage required. A power bank or lipo capable of at least 2A is required.**

- **If the servo arms or top of the faceplate are colliding with the helmet in the up position, reduce the open servo angles.**

  const int servo1_OpenPos = 20; // set the open position of servo 1

  const int servo2_OpenPos = 160; // set the open position of servo 2

  (Servo angle values 0-180 both must be adjusted. Eg "20" "60" adjusted by 2 degrees would become "22" "158")

- **If your servos are "jittery" or the switch is not responding as expected go back through your wiring and ensure all of your ground wires are connected back to the power supply.**

- **Ensure you have used 100Ω resistor for each LED or appropriate resistive value for the colour of LED / Amperage rating of your LED.**

- **Servos hum at the open or closed position. Your angles are set incorrectly and forcing the servos to try to go further than they are able. Reduce your angles in the code.**

A special thanks goes out to all the members of Crash Works 3D.
Crash Works 3D is a collaboration between:
Warlordxxx
Dropwire
Jeet K
Remco
Cranshark - i3 Creations LLC