RoboBrrd

by **RobotGrrl** on November 13, 2011

Table of Contents


Author:RobotGrrl RobotGrrl.com
Robot builder and Mac & iOS developer. Check out my website!

Intro: RoboBrrd

RoboBrrd is an robot / animatronic character whose purpose morphs to mirror that of the virtual world. It is designed to be used as a tangible real world interface to virtual world learning applications. As a standalone robot, RoboBrrd is an entertaining platform that can be used to learn about robotics, Arduino, circuits, and programming.

Here is a teaser video for this RoboBrrd:
http://www.youtube.com/watch?v=dD5KpyTdO5A

Here are other videos about RoboBrrd:
http://www.youtube.com/user/RobotGrrl91#grid/user/CAEAA1300E7BF036 (Big green RoboBrrd)
http://www.youtube.com/user/RobotGrrl91#grid/user/55260A24369DD65D (Learning Pet)

This Instructable will guide you through creating a RoboBrrd- all the way from the circuits to programming to the felt decorations. We will also include reasoning behind our design choices to further enlighten the Instructable.

Here is what a FIRST Lego League team coach (Eaglesnest Robotics) had to say about RoboBrrd:

"We are so excited about this [RoboBrrd] project. What a great way to introduce microcontrollers to our young roboticist. We so appreciate all the work you are doing to make Robotbrrd a great activity for our students. [...] Our goal is to grab their imaginations and run with them. Hopefully after building this project, they will not only want to learn more about programming and buidling robots but create inventions of their own. Thank you so much for helping Eaglesnest Robotics accomplish this task. We hope to create future engineers and Scientist, and also students who are like yourself."

With that said, read on to the next step for the inspiration, motivation, and future of RoboBrrd. Or you can jump right into the build on Step 2! Happy RoboBrrd'ing!

Step 1: Inspiration, History, Motivation

The Beginning

The history of RoboBrrd is quite extensive, but really fun. Back in March 2011, we were contacted by Adafruit Industries asking to make some robot videos for their Ask an Engineer show (before Google+ existed). They didn't have any requirements for the robot, which allowed me to brainstorm about building a robot that can be made out of common materials, such as pencils and coffee stir sticks.

We chose to look at birds as inspiration for this robot.The interesting thing with birds is that they can be anthropomorphized extremely well, examples being Woody the Woodpecker and Donald Duck. I decided to put my own spin on a bird robot, and create one that is an iconic cube shape. The beak mechanism would be similar to a scotch-yoke mechanism to push both halves of the beak open, as opposed to lifting one beak half.

RoboBrrd started out as a 30cm version, with notable pencil eraser tops as eyelashes. This robot was fun to experiment with various behaviours and applications. We added in my robot mesh networking to have it communicate with my other robot, MANOI. Also, we created a Processing sketch that used the RoboBrrd as a tool to make "cosmic soap" artwork from a fluid dynamics meets physics simulation. It even appeared in an Adafruit Show & Tell through IRC, so viewers could message it and it would dance around. To top it all off, it was mentioned in MAKE Magazine Volume 27!

Learning Pet

Fast forward to September 2011, the Open Hardware Summit announced they were having a scholarship contest. They only announced this a little less than a week before the conference, and I wanted to create a RoboBrrd that I could bring and show around to everyone. The 10cm version of RoboBrrd was created from start to finish over 4.5 days (don't ask how much time I slept during those days), and was able to work with the Android Accessory Development Kit or Processing to demonstrate the virtual to real world connection, with a focus on learning. The prize was $2000, which would have been able to get us started on being able to create more robust versions of RoboBrrd that we could give to students. Unfortunately, we didn't win.

Thankfully, the story doesn't stop here. The RoboBrrds were featured on MAKE Blog after the Maker Faire, and Adafruit Industries has kindly given me some more servos to create another RoboBrrd, with perfect timing for this amazing laser cutter contest opportunity.

Motivation

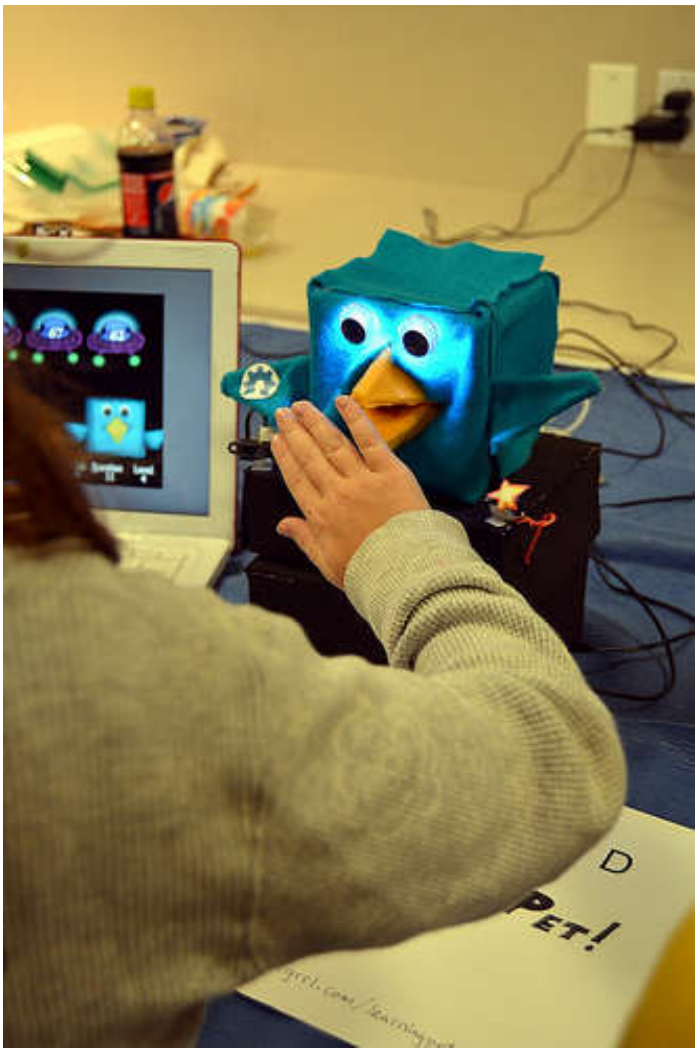
People think I'm crazy with all of my RoboBrrds and staying awake to crazy hours in the morning working on them. The main motivation behind these robots is the challenge of creating a tangible embodied character that can interact with the virtual world. I was heavily inspired by a project at the MIT Media Lab with a similar physical-virtual world premise, but wanted to make something that can be taken everywhere and thrown into a backpack at the end of a school day. Something where the learning can be brought with you everywhere, and interface with anything. RoboBrrd would mainly serve as a physical tie-in to the virtual world learning, aimed at K-6 students.

Future

Our ultimate goal is to be able to create RoboBrrds and work together with schools to give RoboBrrds to students as DIY kits that they put together, and then use as a tool. This is extremely valuable, as the students will have a special connection with their RoboBrrd since /they/ put it together and are able to fix it. They are learning about various subjects while using their creation. It is a magical experience.

We have had a lot of fun with the RoboBrrd platform, and it is only just beginning. We are extremely excited about this, as we already have had some RoboBrrd fan artwork sent to us. We refined RoboBrrd so that there is a clear Instructable to build one, so everyone can get started.

With that said, let's go and build a RoboBrrd!



Step 2: Setup

Before we embark on this journey of building a RoboBrrd, there are some parts that you will need and some tools!

Parts needed:

Electronics
- 2 Large Servos
- 2 Micro Servos
- 2 Light Dependant Resistors aka Photocells
- 2 RGB LEDs
- 1 Arduino
- 1 Proto Screwshield
- 1 Spool of wire
- 1 USB B Cable for Arduino (you probably already have one)
- 1 Speaker

- 2x 1k Ohm resistors
- 6x 100 Ohm resistors
- 4x 3 Pin double sided male headers - for connecting the servo plugs

Voltage Regulator Circuit
- 1x LM317
- 1x 5k Trimpot
- 2x 10uF Electrolytic capacitors
- 1x 280 Ohm Resistor
- 7x 100 Ohm Resistors
- 2x 1N4001 Diodes
- 1x 2.1cm DC barel jack
- 2x 2-pin Male headers
- 1x 2-Terminal screw terminal
- 3x 5-pin Female headers

DC Motor Circuit
- 3x 0.1uF Ceramic capacitors
- 1x TIP 120
- 1x 1N4001 Diode
- 1x 100uF Electrolytic capacitor
- 1x 2-Terminal screw terminal

Construction
- Popsicle sticks (~$3 at Walmart)
- Coffee stir sticks ($1 at Dollarama)
- Craft sticks (~$4 at Walmart)
- Paints, felt, foam (~$10 at Walmart)
- Pipecleaners (~$3 at Walmart)
- Googely eyes (~$2 at Walmart)
- Dual Lock ($5 at Staples)
- Sticky Tack ($5 at Staples)
- Strong sculpting wire ($2 at local hobby store?)

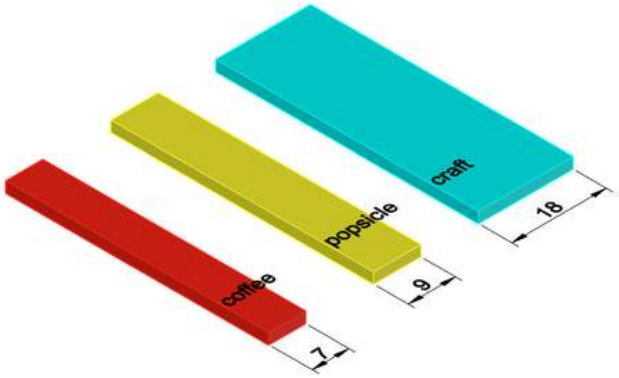
Tools:
- Awl (sharp poking tool)
- Xacto knife
- Scissors
- Pliers
- Wire strippers
- Wire cutters
- Soldering iron
- Solder & wire
- Desoldering pump
- File
- Screwdriver
- Ruler
- Pencil
- Sharpie
- Hot glue gun
- Glue sticks

Timeline
Building a RoboBrrd takes some time if you are working by yourself. If you have someone helping you, it can be done in about 8 hours. Below is an approximate amount of time that it took me to create this RoboBrrd for each of the steps.

- 2.50 hr : Faces Construction
- 1.50 hr : Beak & Wings Construction
- 1.00 hr : Base Construction
- 2.00 hr : Beak Mechanism
- 0.50 hr : Wings & Servos
- 0.50 hr : Rotational Servo
- 1.50 hr : Hula Hoop
- 1.25 hr : LM317
- 0.50 hr : TIP 120 & DC Motor
- 2.00 hr : Proto-Screw Shield
- 0.75 hr : Wiring
- 0.50 hr : Calibration & Testing
- 1.50 hr : Felting
- 0.75 hr : Foaming
- 0.75 hr : Behaviours
- 1.50 hr : Dances
- 0.50 hr : Final Touches
- 1.00 hr : Above & Beyond

Total time: 20.5 hours (it is best to split this across multiple days - perhaps 2 weekends)

Note: All of the dimensions in this Instructable are in centimeters, with the exception of the CAD illustrations which are in millimeters.



Step 3: Faces Construction

There are three different faces that we need to create to form the cube of RoboBrrd's skeleton. These are the bottom face, the side faces, and the front & back face.

Bottom Face

The bottom face is designed with a large hole in the back to route the wires from the LEDs & LDRs and servos into the base where the controller board will be. There is a 3 popsicle stick wide platform in the middle in order for the dual-lock to attach to the rotational servo in the base.

Side Faces

For the side faces, you will need to create two for the left and right side of the bottom face. The only difference between the two sides will be the positioning of the ledge. The ledge is always closest to the back of the RoboBrrd. The ledge in the side faces is used to neatly route the LED & LDR wires, allowing them to not interfere with the movement of the wings and beak mechanism.

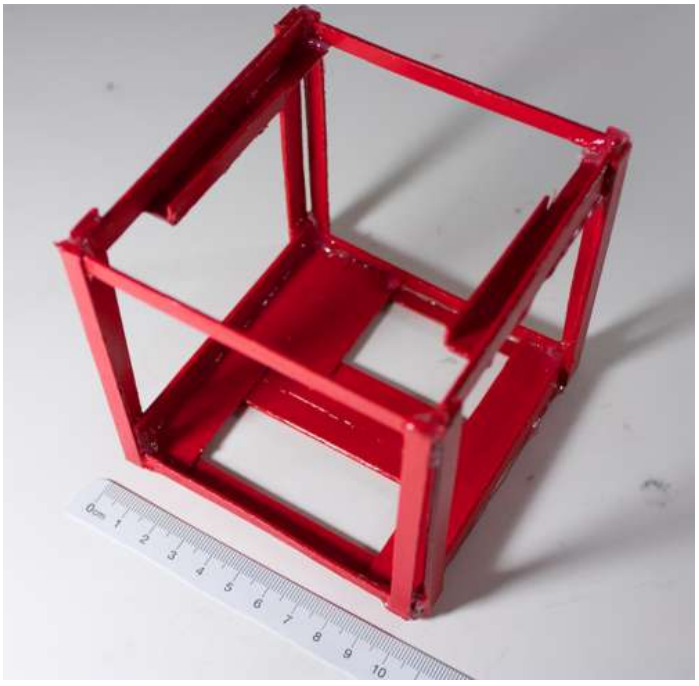
Front & Back Face

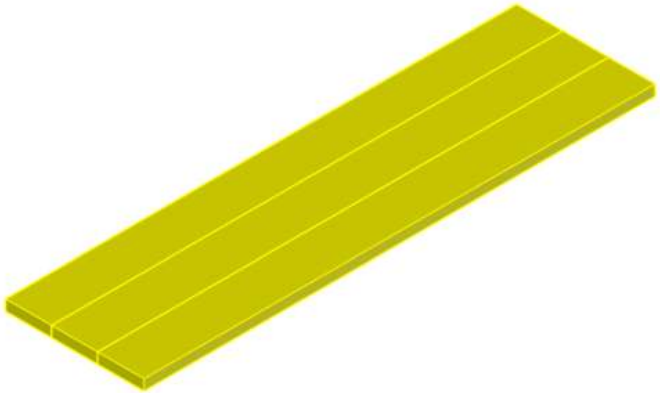
The front and back face are constructed out of coffee stir sticks as their main structural purpose is to help the side faces remain perpendicular to the bottom face.

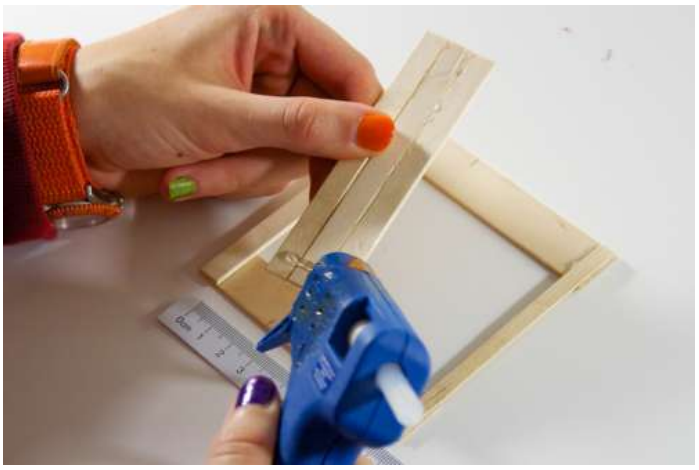
The front face is specifically used to mount the beak halves. This will be in more detail in a later step.

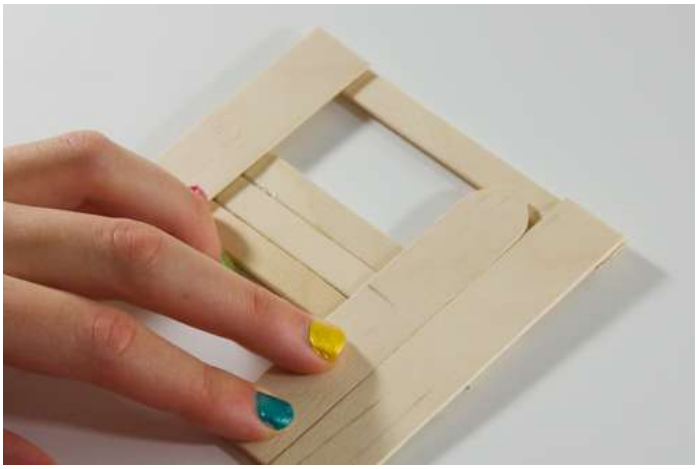
Attaching

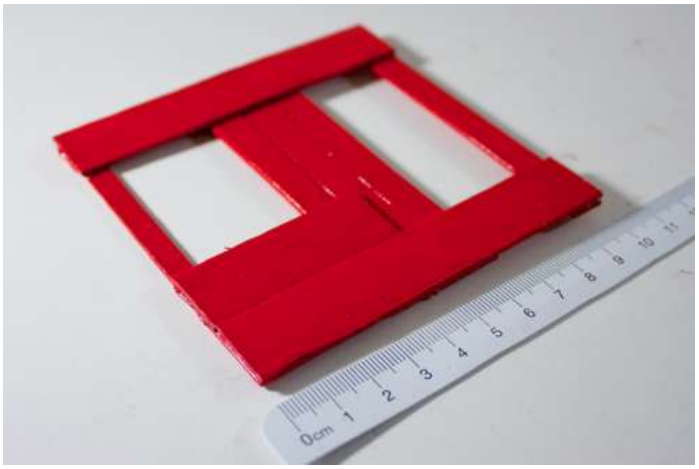
When all of the faces are constructed, they need to be attached. It is important to ensure that the faces are perpendicular to each other. Placing the hot glue on the edges of the faces that are being mounted is usually the best tactic for this.

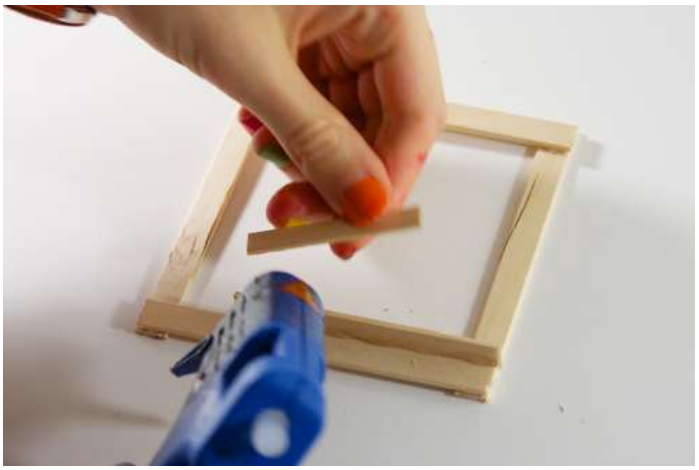


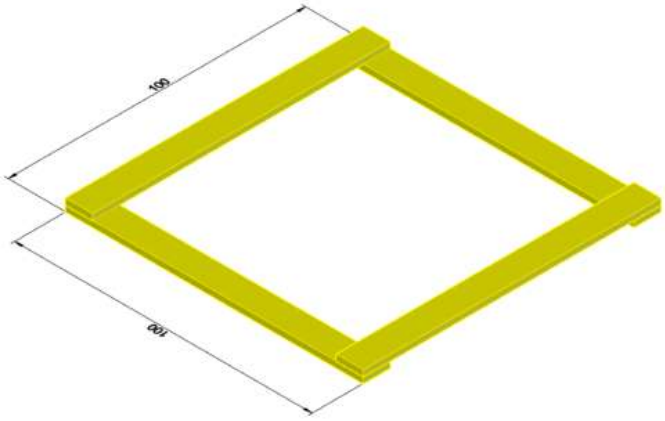


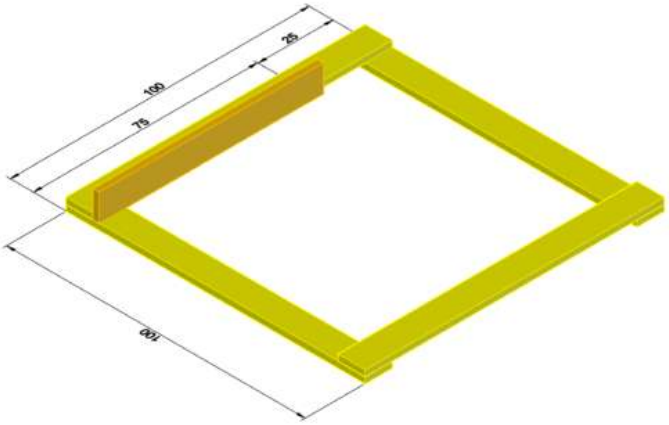


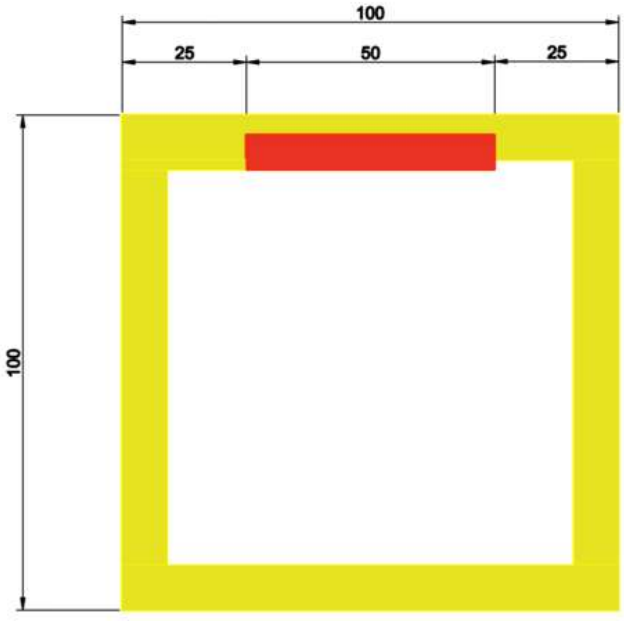


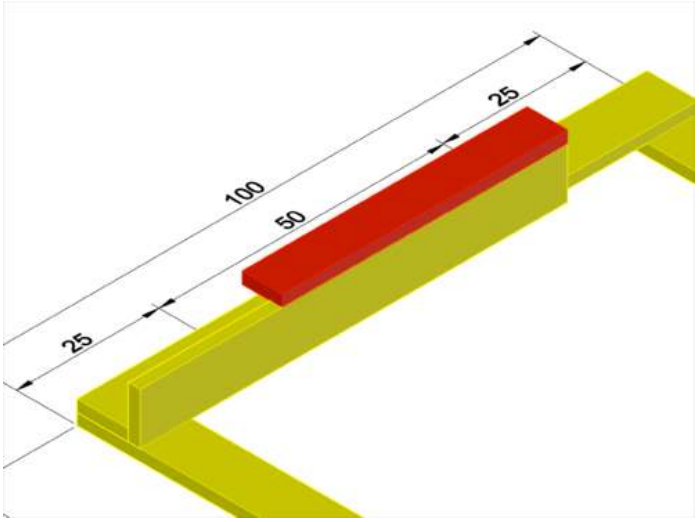


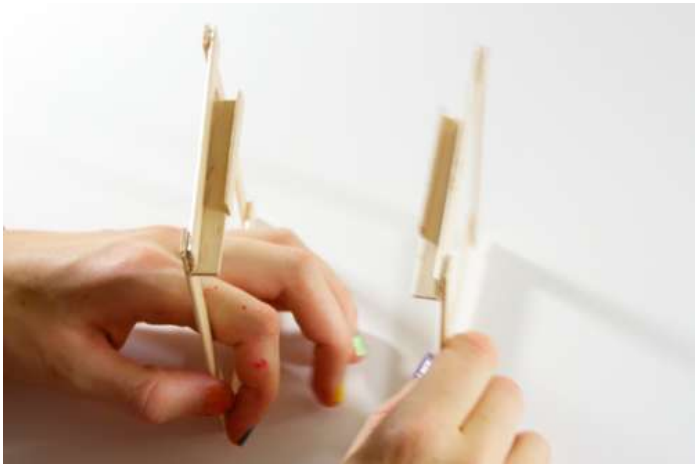


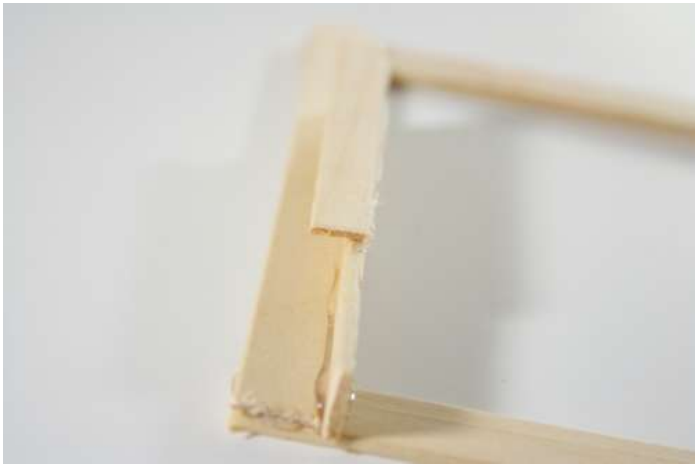

100
100

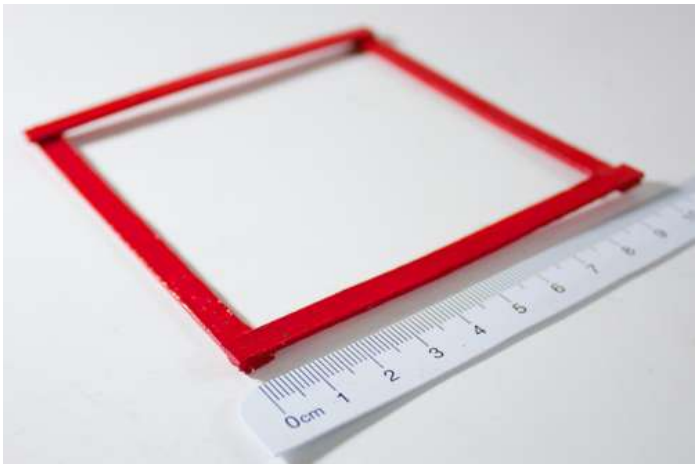

25
100
75
100

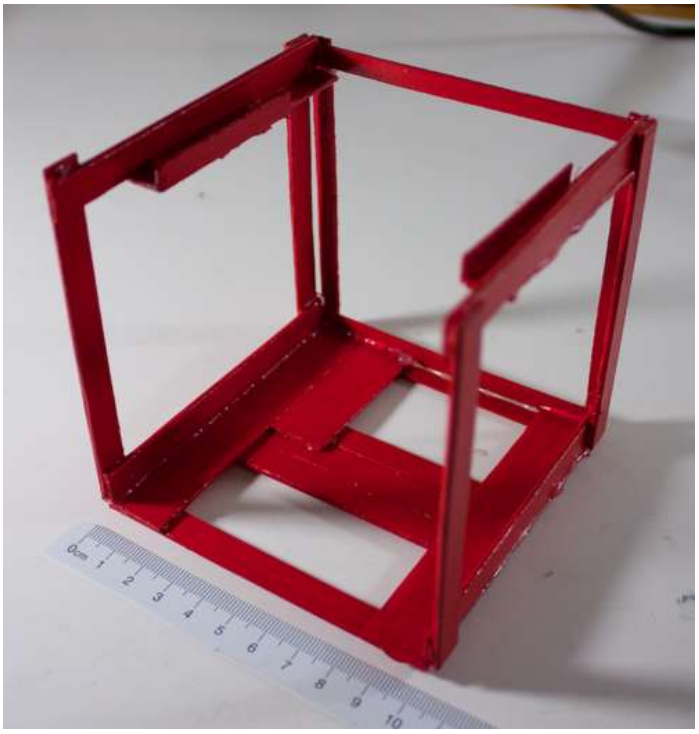

100
25
50
25
100


25
100
50
25









Step 4: Beak & Wings Construction

Beak

The beak is designed to be as light as possible, but also sturdy. We need to create two beak halves. They are both identical in construction.

There are two coffee stir sticks as the bridge of the nose to allow for an additional coffee stir stick to be inserted. This will be done in a later step.

The popsicle stick for the beak is used for attaching to the axels from the servo. This attachment is done using a loop of wire, which allows for some flexibility.

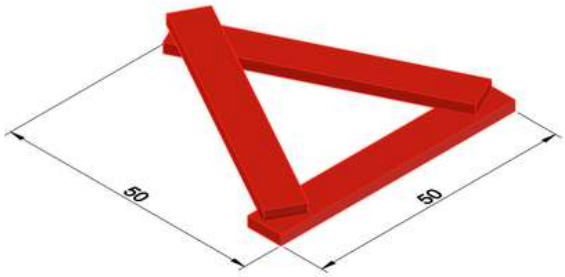
We will eventually be mounting the beak halves on the front face by threading a coffee stir stick through the side of the nose.

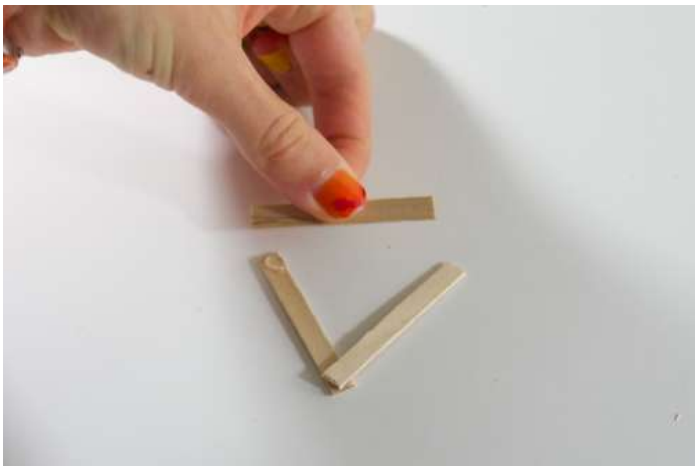
Wings

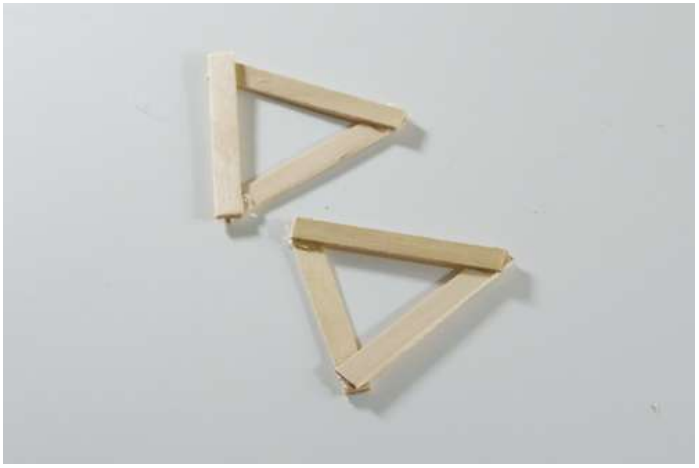
We need to create two wings for RoboBrrd, one for each side. The only aspect of the construction that will have to be mirrored is the diagonal piece attaching the bottom of the wing to the vertical coffee stir stick.

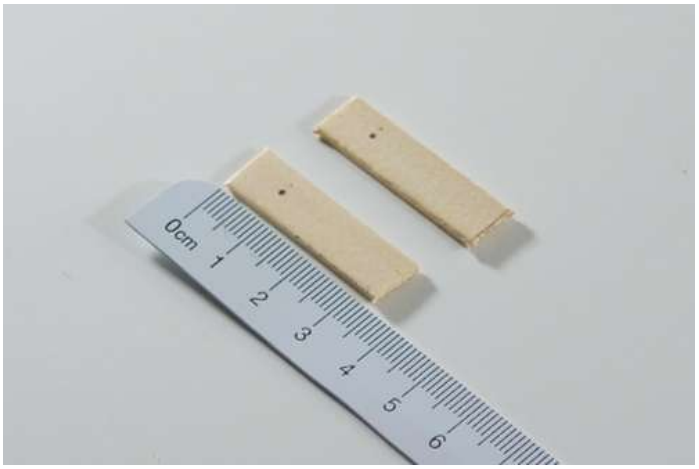
For the wings, we want to keep them as light as possible, but still give them a good structure for when we are placing felt over them. This is why there is a diagonal piece only on the front side of the wing. There are two coffee stir sticks that attach the wing onto the servo arm on each side. This allows for a better attachment to the servo.

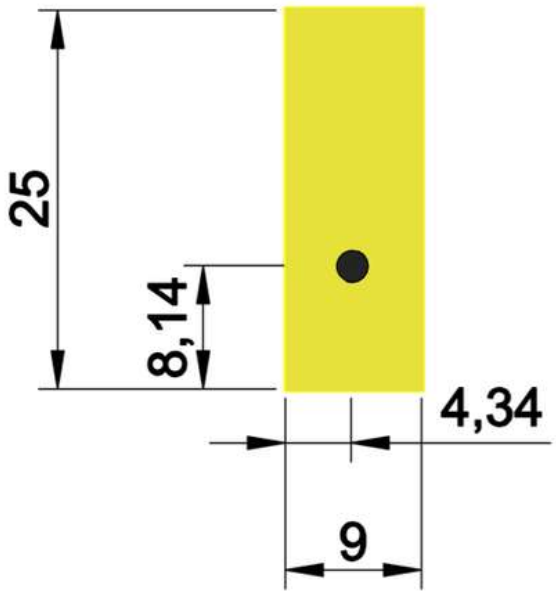
On the other side of the servo arm, we have a counter balance. This uses two pennies in order to move the balance point of the entire wing construction closer to the middle. Theoretically, this will alleviate some stress on the servos.

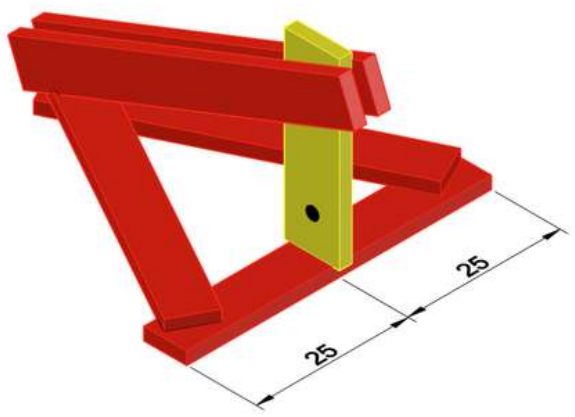


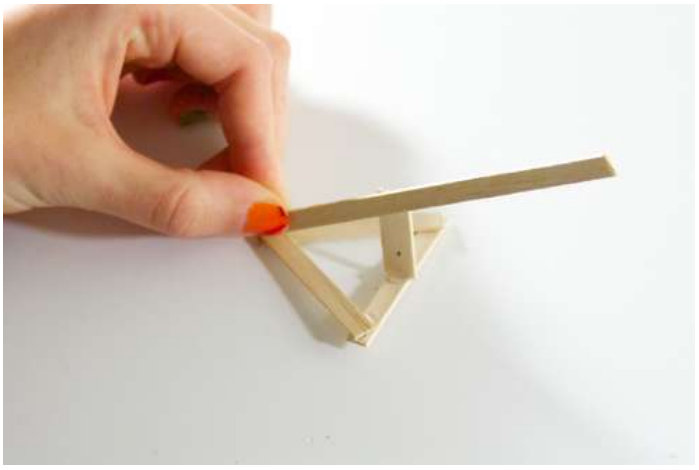


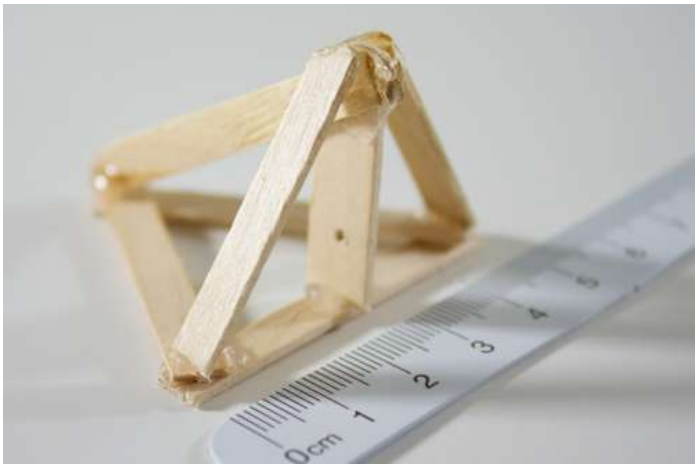



0cm
1
2
3
4
5
6

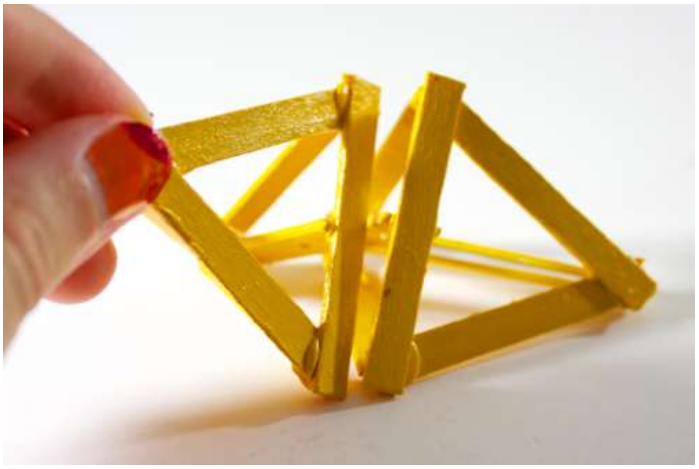

25
8,14
4,34
9

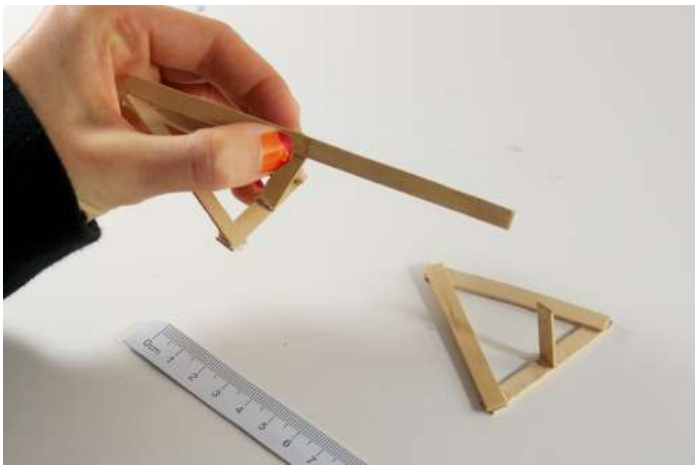

25
25

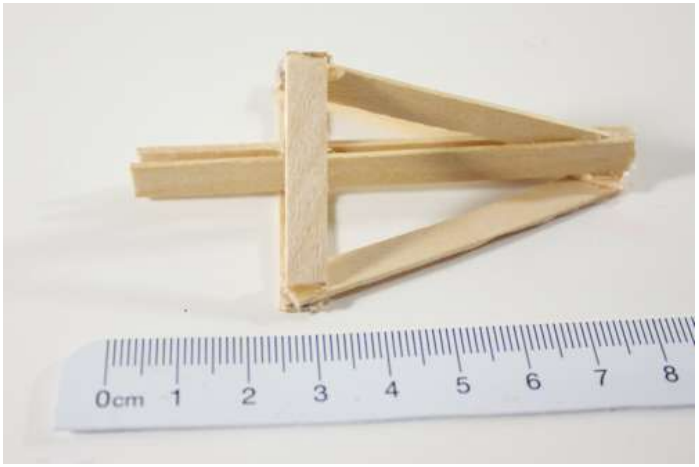


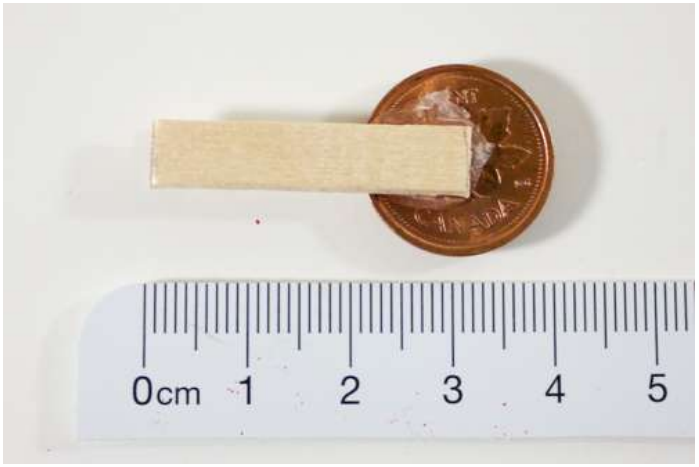

0cm
1
2
3
4

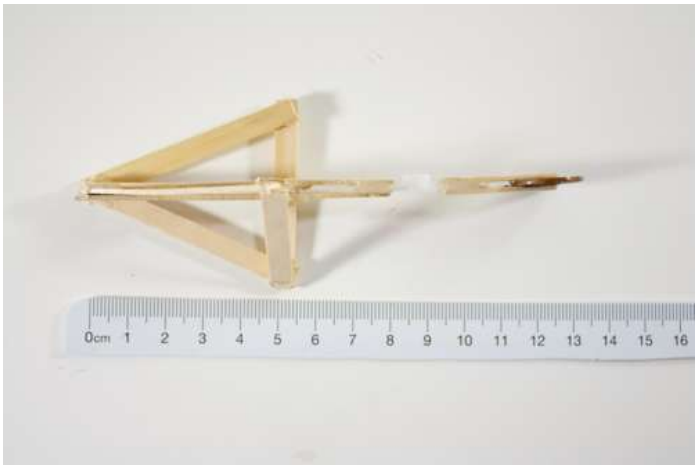

0cm 1 2 3 4 5 6

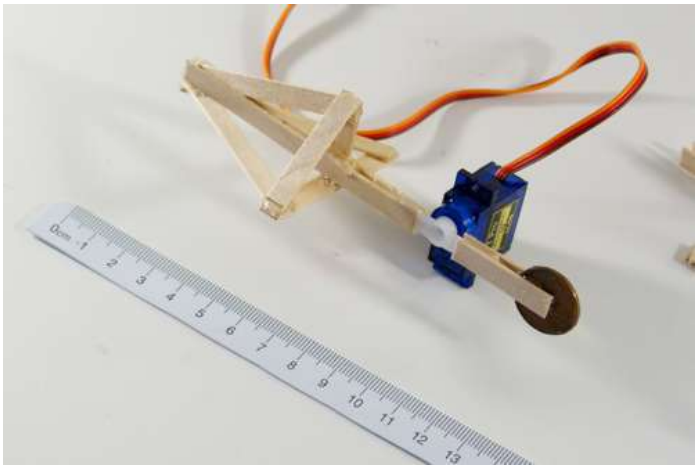


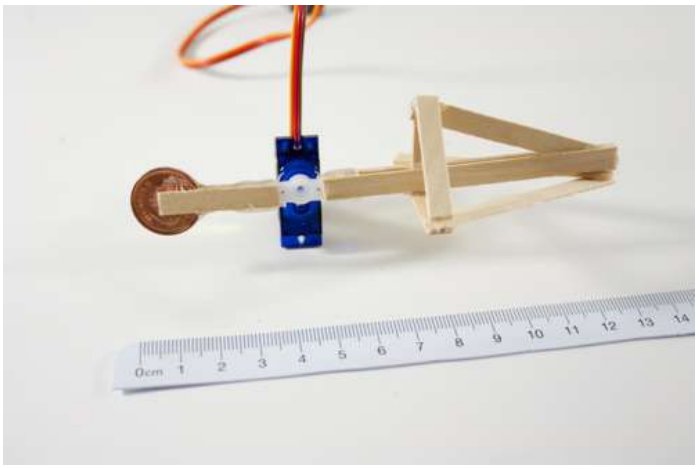


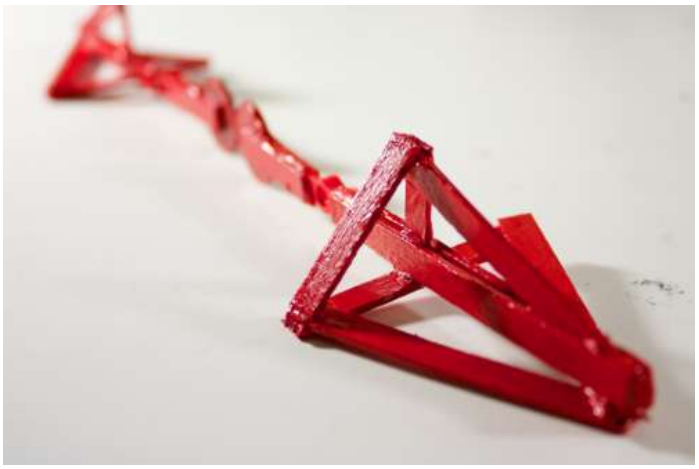

0cm 1 2 3 4 5 6 7 8

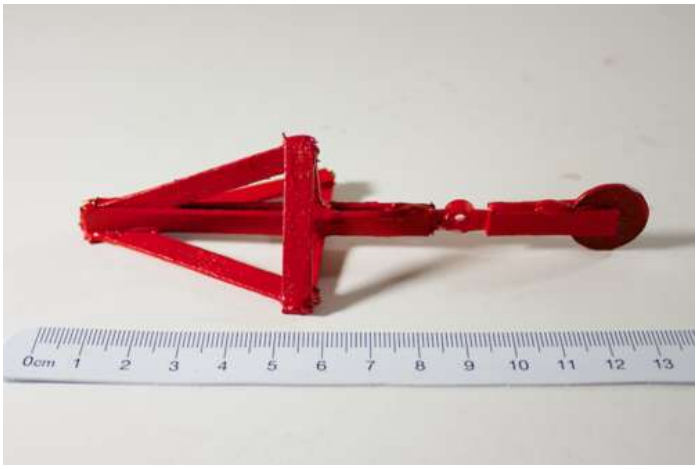

0cm 1 2 3 4 5


0cm 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16




0cm 1 2 3 4 5 6 7 8 9 10 11 12 13 14





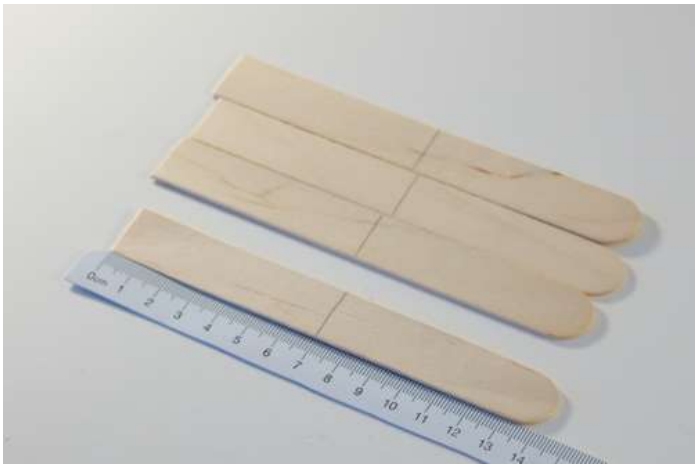
Step 5: Base Construction

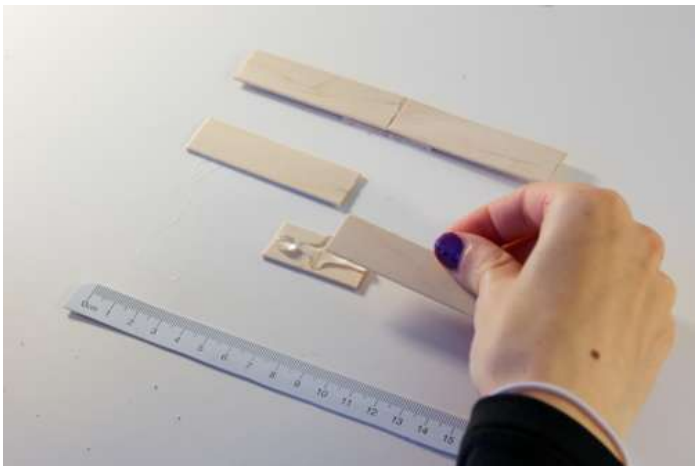
The base is where the rotational servo, battery, and controller board will be housed. It is also good to keep in mind that it will also be where all of the wires will be situated. The controller board will be placed to the right of the servo, and the battery and speaker to the left of the servo.

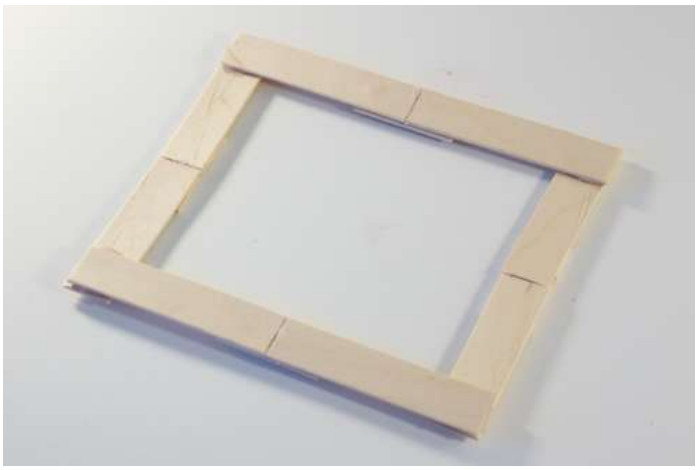
The rotational servo will be placed directly in the middle of the base, and it is what will be bearing all of the weight of the faces construction above. We will use the circular servo horn on the rotational servo as it has the largest surface area. On top of here, we will have a small popsicle stick construction which will have a piece of dual lock on it. On the bottom face of RoboBrrd we will attach the other piece of dual lock. This will allow us to be able to take off the main RoboBrrd structure without having to dismantle everything.

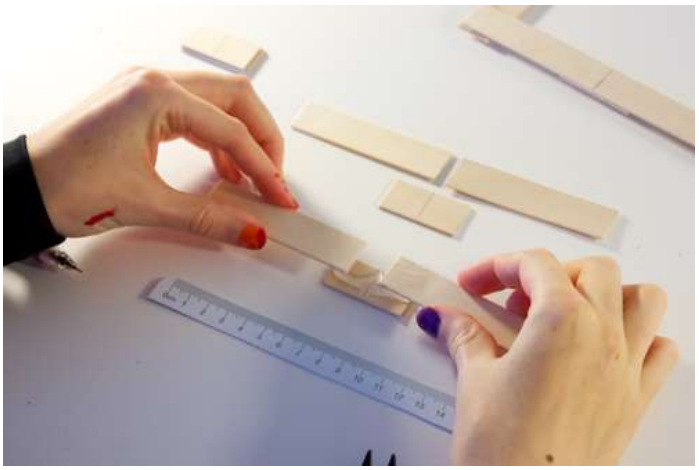
If you need some more room for the controller board, move the rotational servo to the left a few millimetres. For this RoboBrrd, we had to move the rotational servo 3mm to the left to allow for enough room for the controller board.

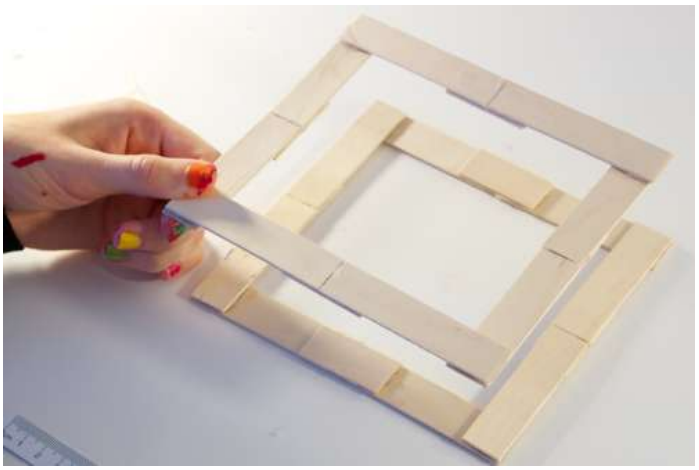
The easiest way to construct the base is to make the bottom and top 'flat' pieces first. Then, attach the perpendicular stands to the bottom. To finish, glue the top piece onto the stands.

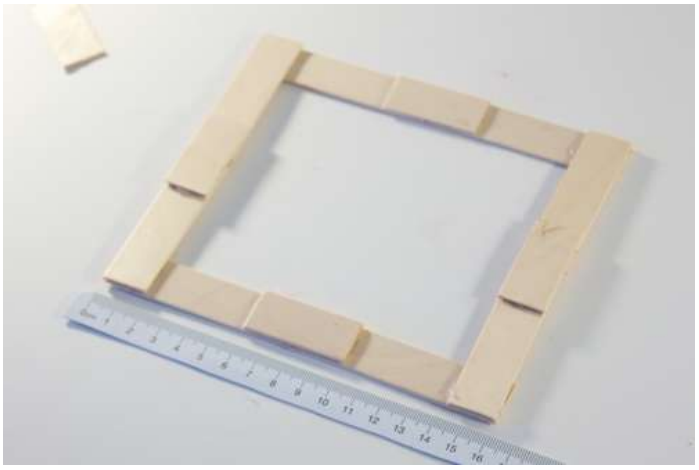


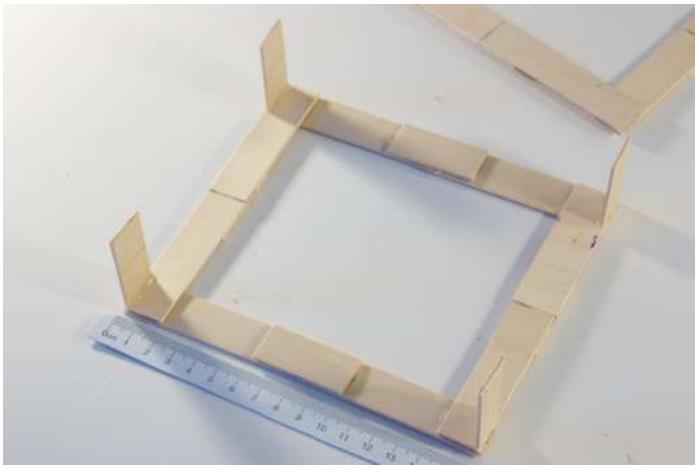


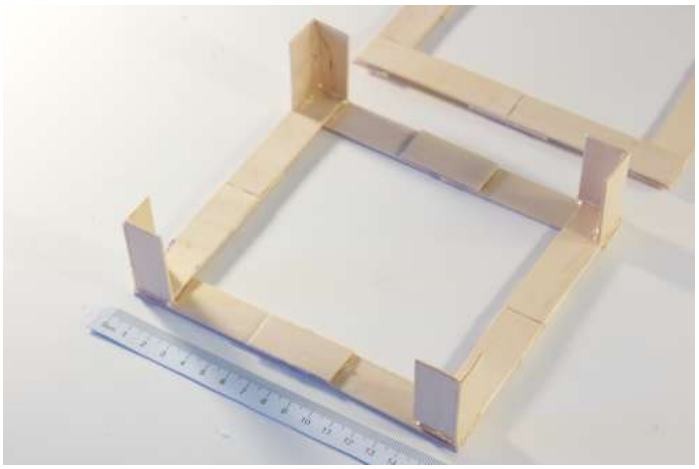


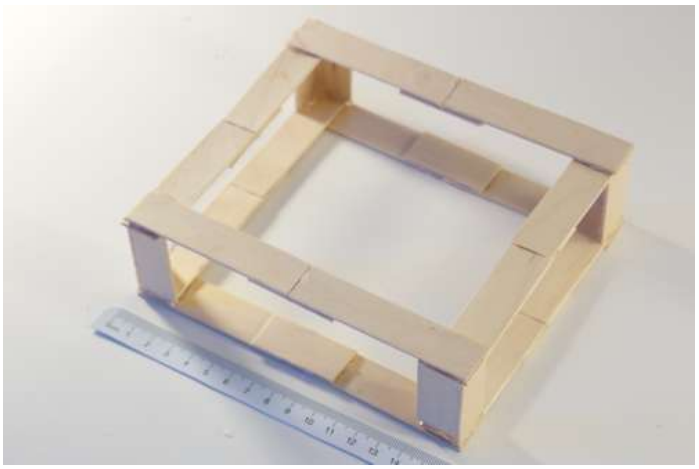


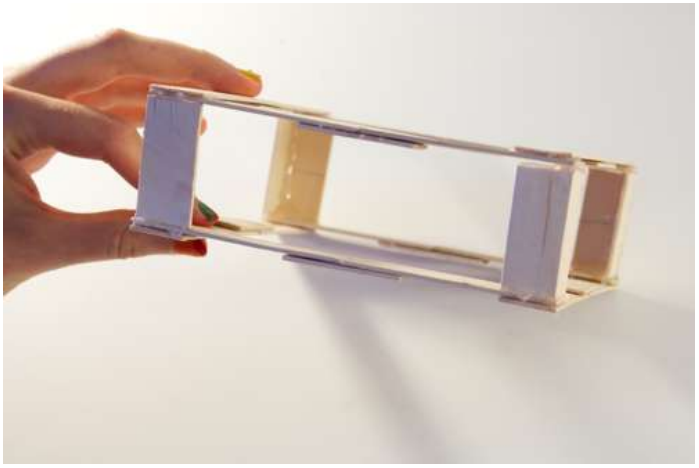


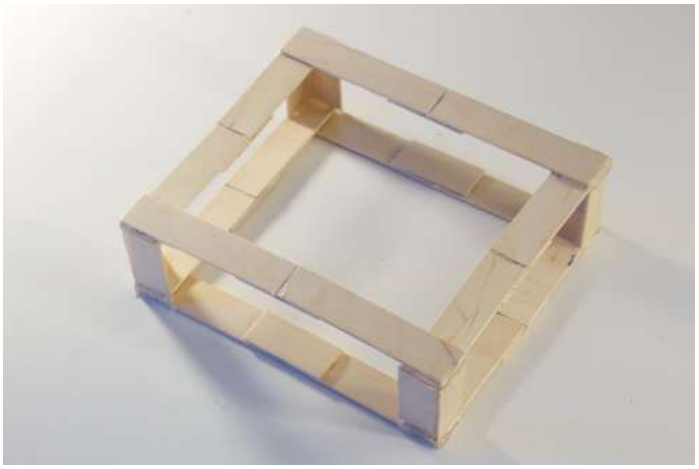


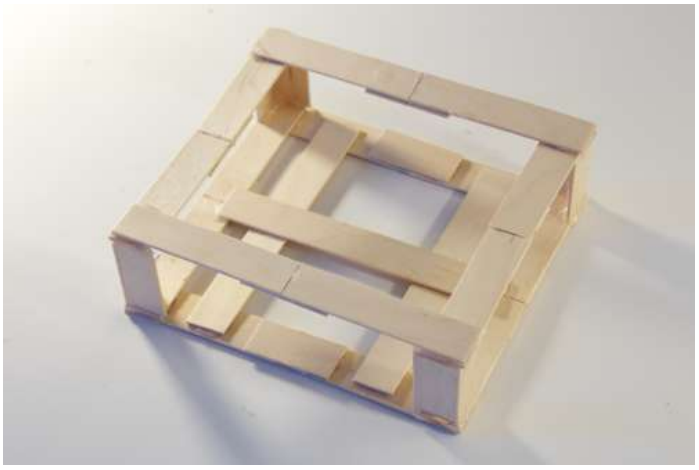


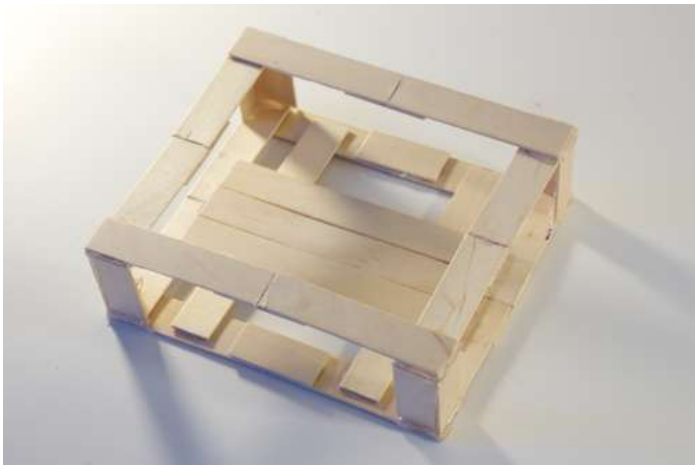


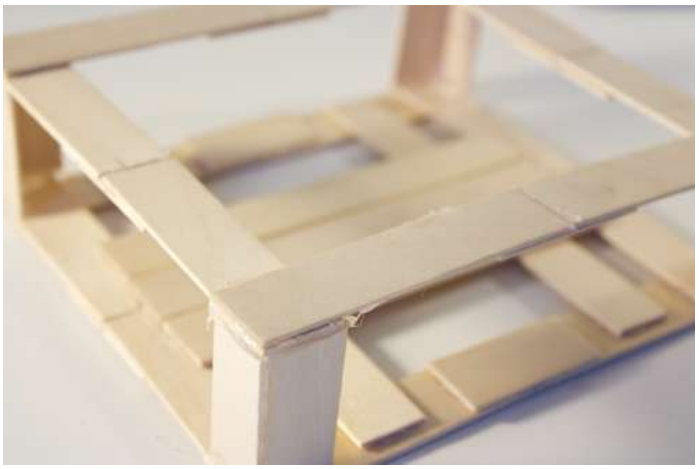


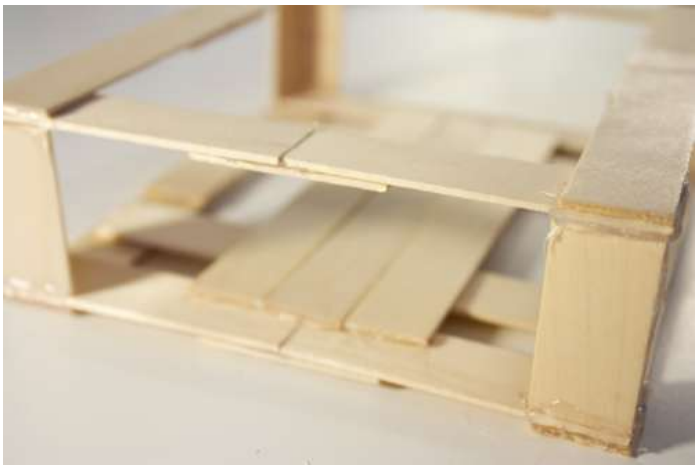


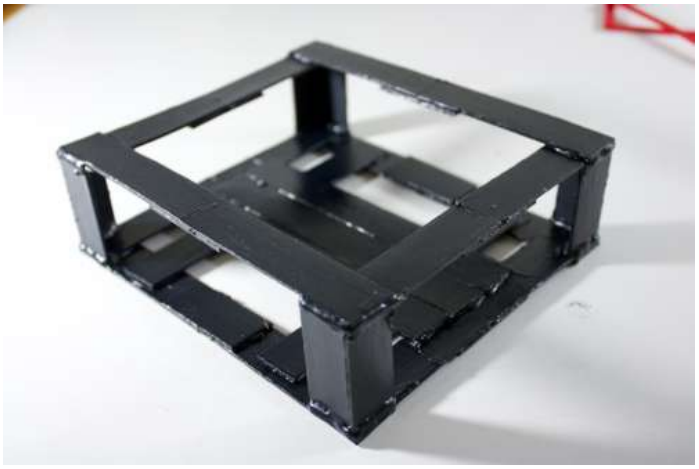


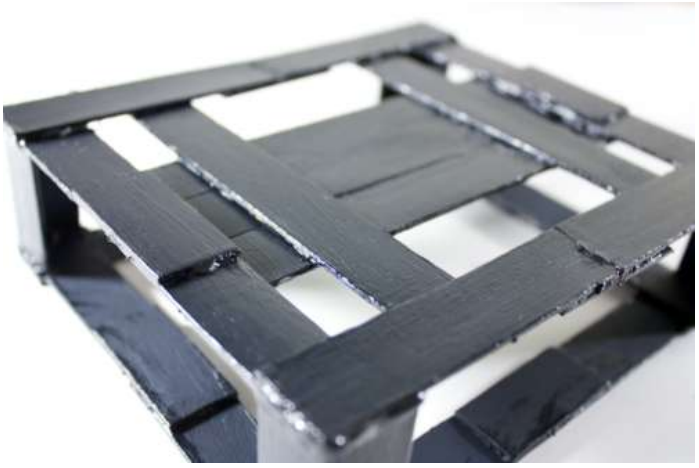












Step 6: Beak Mechanism

The beak mechanism is the “creme de la creme” of RoboBrrd. To illustrate its elegance in words, the beak mechanism allows for the two beak halves to open at once, rather than lifting one half of the beak.

To begin with the beak mechanism, we first have to mount the servo for the beak. This should be placed in the middle of the back left platform. We will then attach a 4-point servo arm to the servo and glue on the popsicle stick arm. The popsicle stick arm should have a hole near the end, and it should also be securely glued onto the servo’s servo arm.

We will thread the beak halves with a 10cm piece of coffee stir stick. Now, in order to make the rotation of the beaks on these coffee stir sticks about a more central fulcrum, we need to cut some small ‘blocker’ pieces that will stop the beak from falling. There should be a couple of millimetres of clearance between the blocker and coffee stir stick.

Next, we need to mount this coffee stir stick to the face. Sometimes adjustments have to be made, so it is better to first measure everything and temporarily mount them with sticky tack.

When both halves are mounted onto their coffee stir sticks, you should ensure that there is enough room to open and close the beak. The gap between the two beak halves should be a little less than 1cm. When testing this, it is important to be rotating the beaks about their fulcrum, which will be closest to the back of the beak where the vertical popsicle stick is. The blocker that was just mounted should be assisting you with this.

Now we have to measure the length of the axels that connect the beak to the popsicle servo arm. To do this, it is critical that the popsicle servo arm’s hole is located directly where the middle of the two beaks will be.

To measure the length of the axels, take a blank popsicle stick and hold it on the beak where the hole is, and move the other end to where the popsicle servo arm is. Draw a line for where the axel should be cut. If this is done correctly, they should be the same (or very close, within a few mm) length.

Poke holes in each end of the axel and attach them with some solid core wire to the beak and popsicle servo arm.

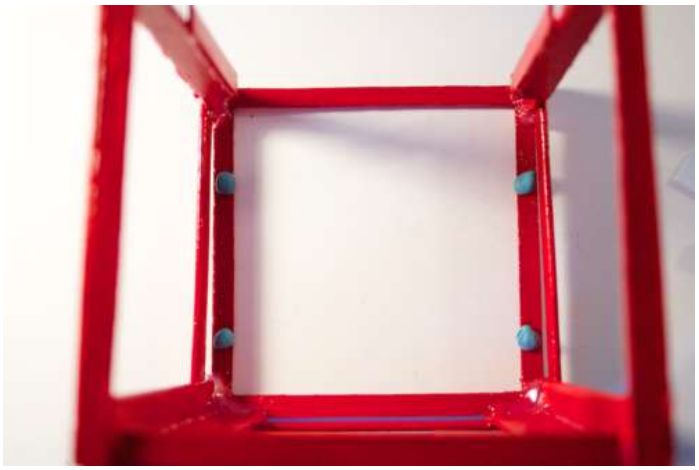
Test the mechanism by manually moving the servo. You should check and see if the beak is able to smoothly open and close. If it does not, see where the modifications need to be made. Usually it is something blocking something else from moving.

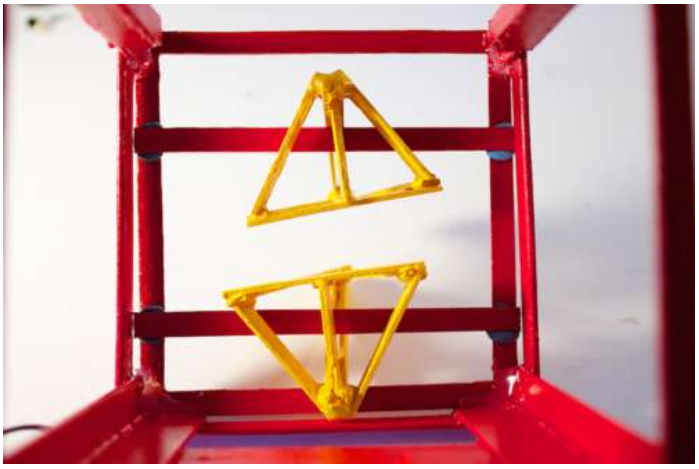
When the testing is complete, you can remove the sticky tack from the coffee stir sticks and mount them using hot glue.

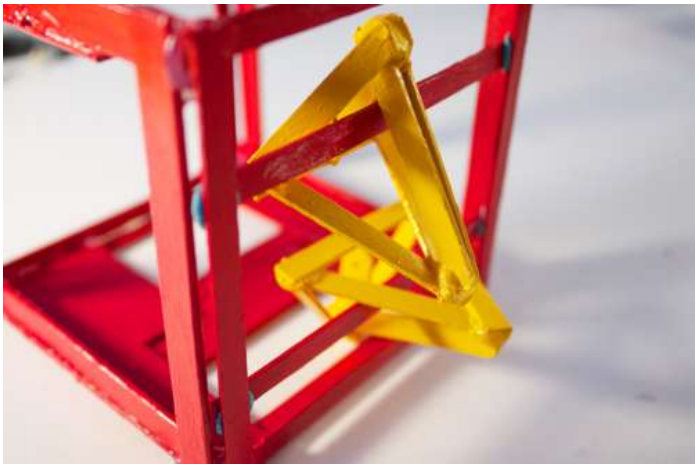
Next, we need to prevent the beak from moving horizontally. We will use small coffee stir sticks mounted vertically to assist with this task. They should be placed about 1cm apart from the center to allow for some wiggle room.

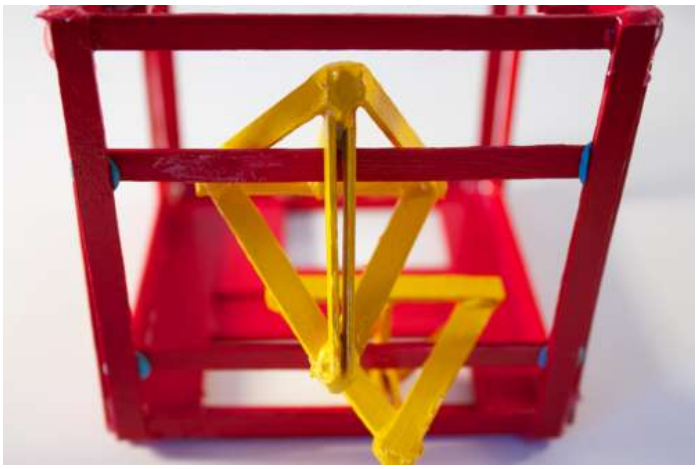
As always, test the mechanism to ensure that it is working properly. To make the wires not come unraveled, loop them tightly to the popsicle sticks and use hot glue to keep the windings secure.

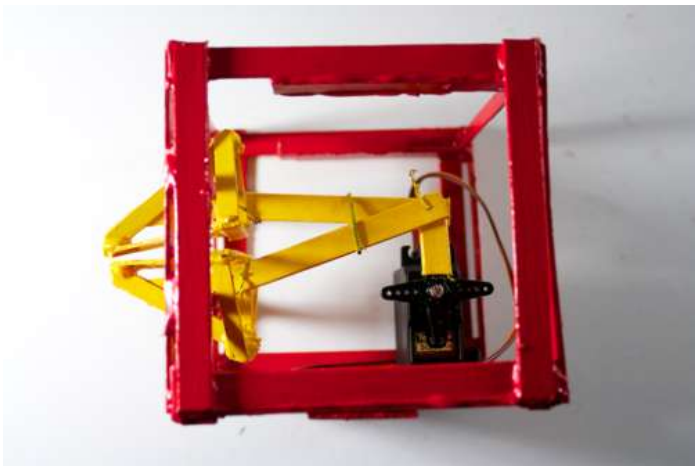
In programming the beak, it will be important to properly set the boundaries for this servo for the open and closed positions. Doing so will ensure that we are not going to cause any accidental damage on the front face.

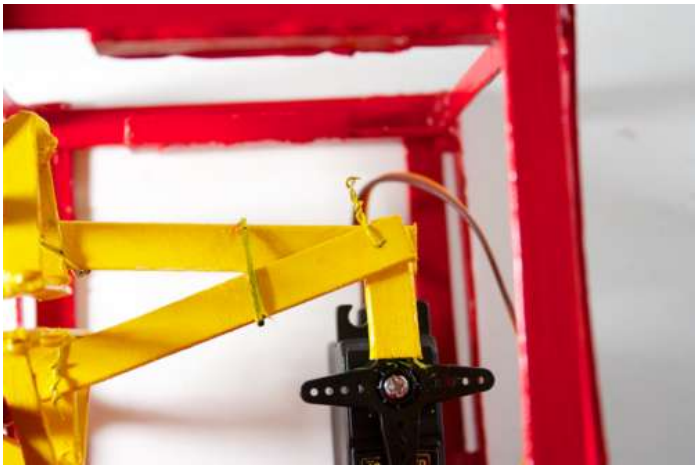


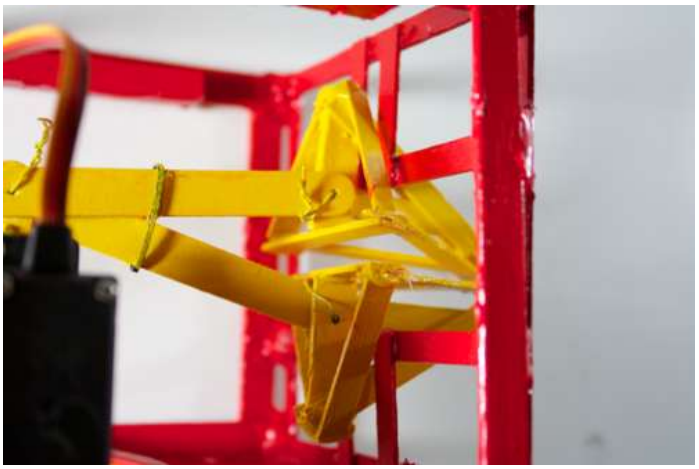


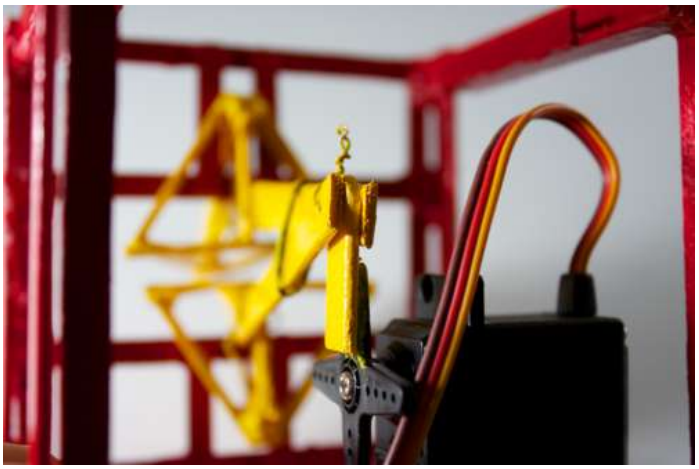


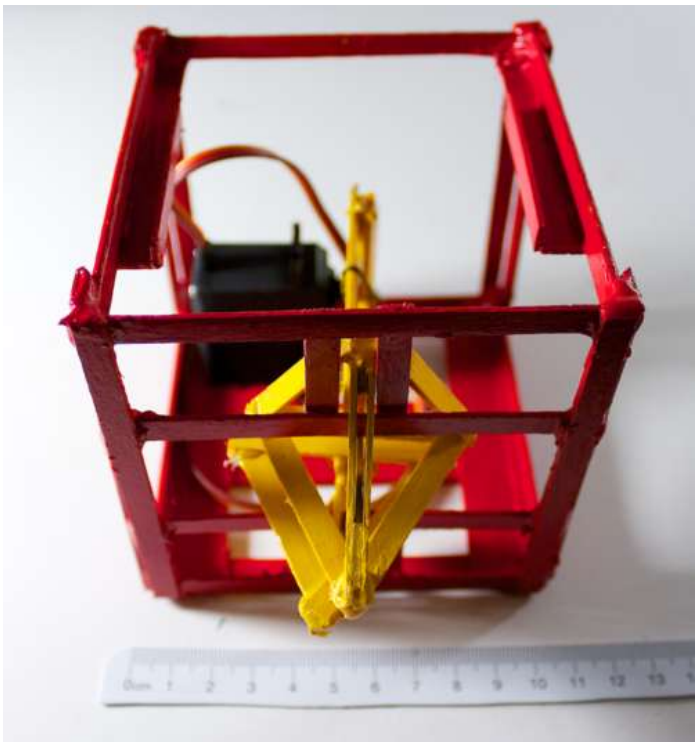


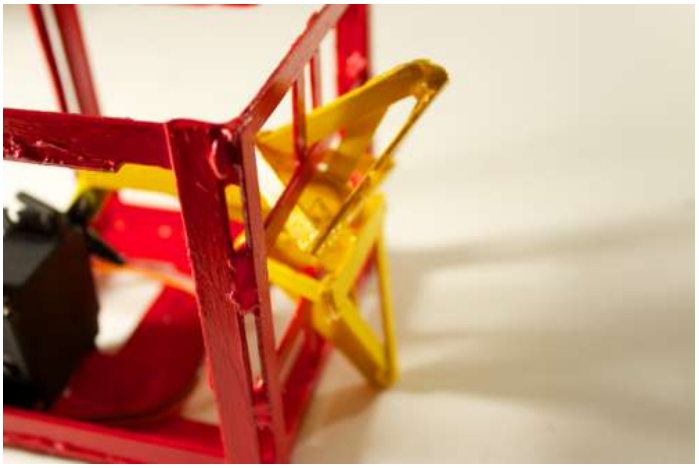


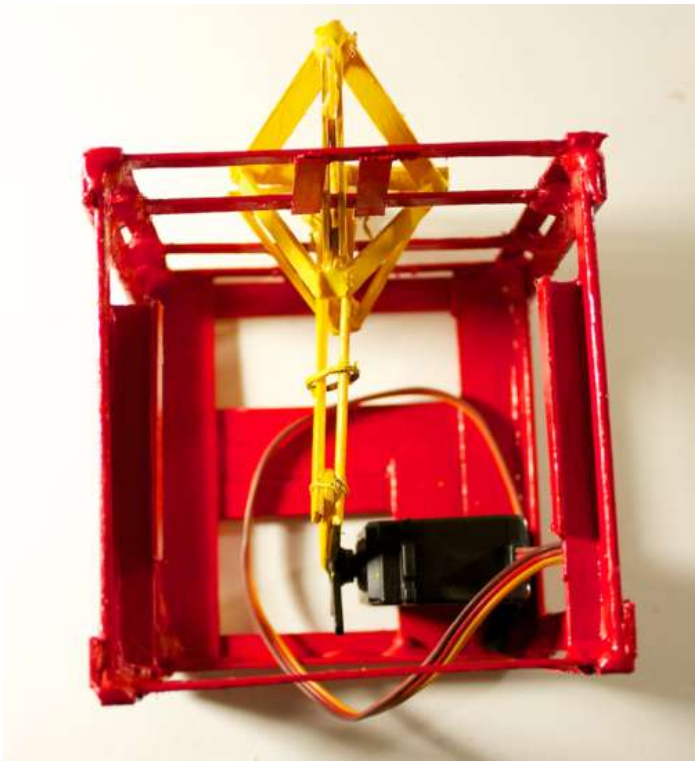


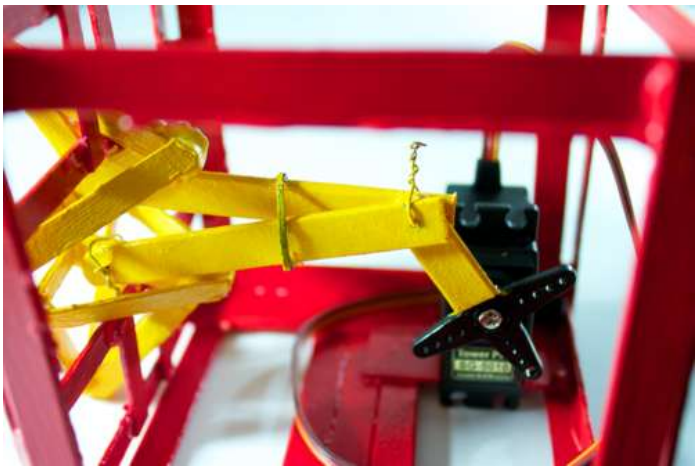


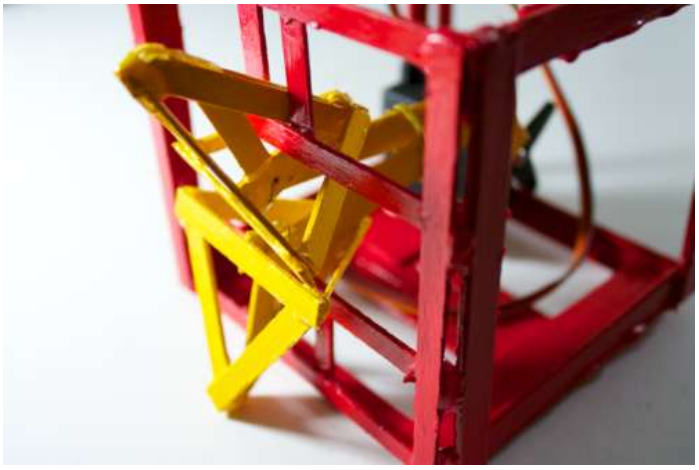


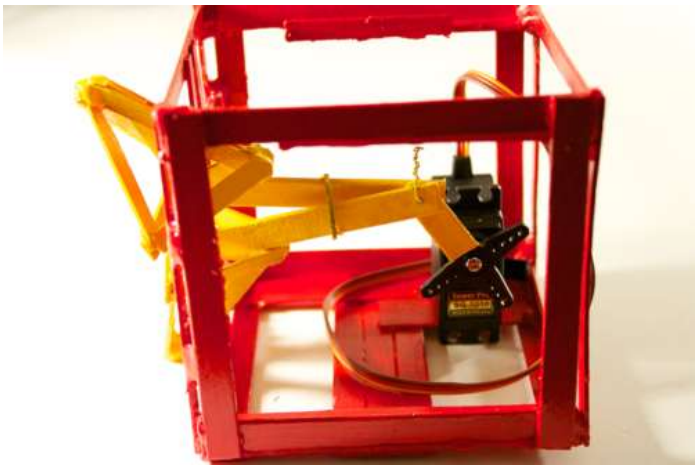










Step 7: Wings & Servos

The wing servos will be mounted on the side near the front of the RoboBrrd structure. There should be enough room for the wing to exit out the side of the structure. In order to mount them correctly, the bottom mounting plate of the micro servo will have to be cut off and filed down. The side face and bottom platform will provide enough stability for the wing servo when hot glued in.

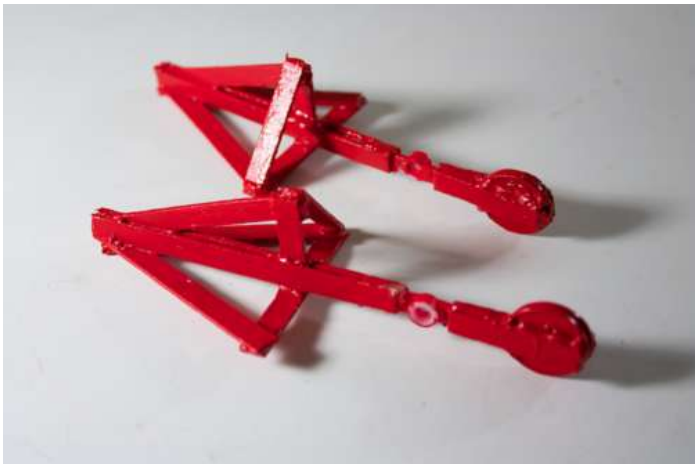
Before mounting the wing construction's servo arm to the servo, we need to ensure that the servo's boundaries are at the right positions. It will be useful to use an alternative servo horn for this testing. The servo should be able to move a few degrees past the vertical point near the top, and the same with the bottom. When testing the opposite side's servo, the adjustments will have to be mirrored. When this is complete, move the servo to the middle.

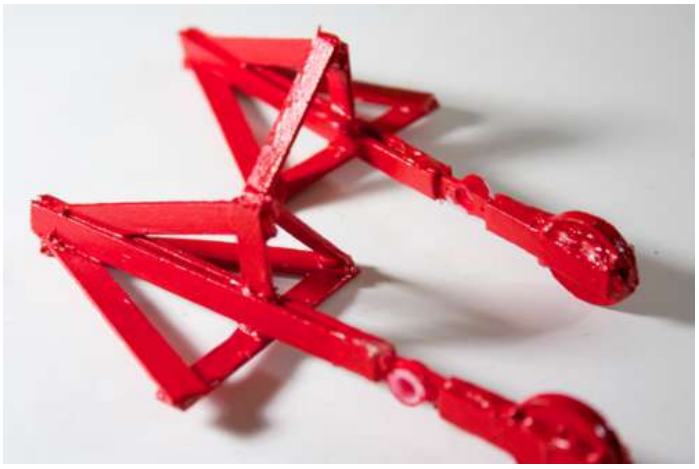
The wings will be mounted to the servo using the servo arm that was used within the wing construction. We will mount them directly in the middle to ensure the proper servo boundaries.

We should be able to move the wings until the wing servo arm hits the bottom of the RoboBrrd structure, and until the wing hits the side of the RoboBrrd structure as it is nearing the top.

The counter balance should have enough room to move freely within the hole in the bottom face. The two counter balances from the opposite wings should not hit each other. There should also be enough clearance to not interfere with the bottom beak in the beak mechanism.

When everything above is tested and complete, add the screw to tie the servo horn to the servo more securely.





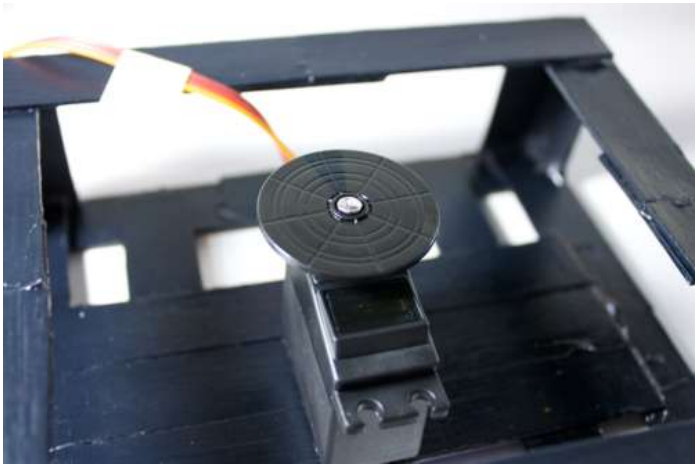
Step 8: Rotational Servo

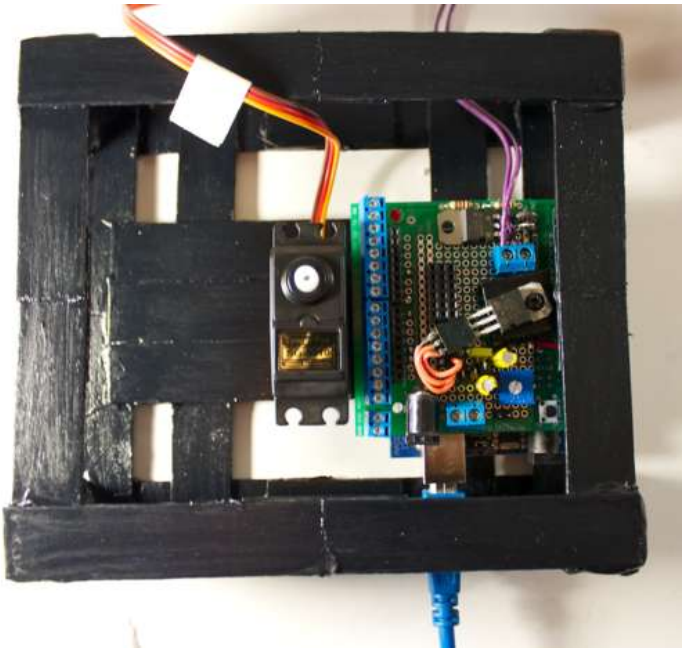
As mentioned in the previous step “Base Construction”, the rotational servo will be mounted directly in the middle of the base. The servo horn on the rotational servo should be the exact center (or close to it) of the base. If the controller board needs more space, the rotational servo can be moved a couple of millimetres to the left. For this RoboBrrd, we had to move the rotational servo 3mm to the left to allow for enough room for the controller board.

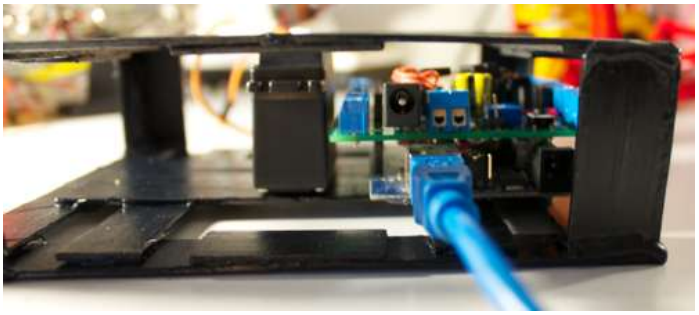
First we have to check the boundaries of the rotational servo. Using a 4 point servo horn, see that it can travel around pointing from the left to the right. When finished, move the servo horn directly to the middle.

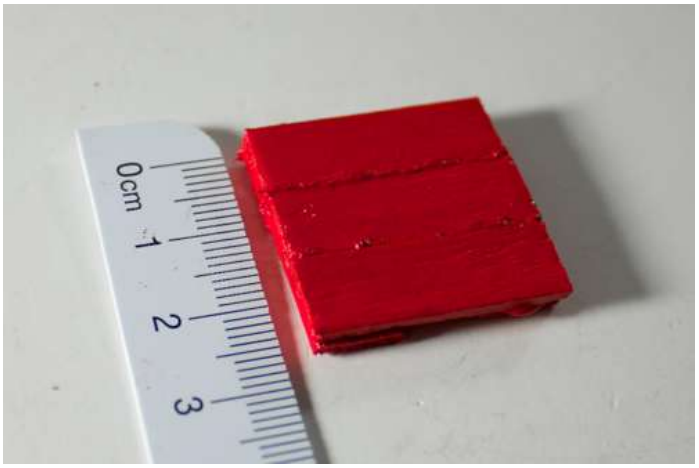
We will use the circular servo horn on the rotational servo as it has the largest surface area. On top of here, we will have a small popsicle stick construction. Before mounting the construction to the servo horn, ensure the screw is in the servo horn.

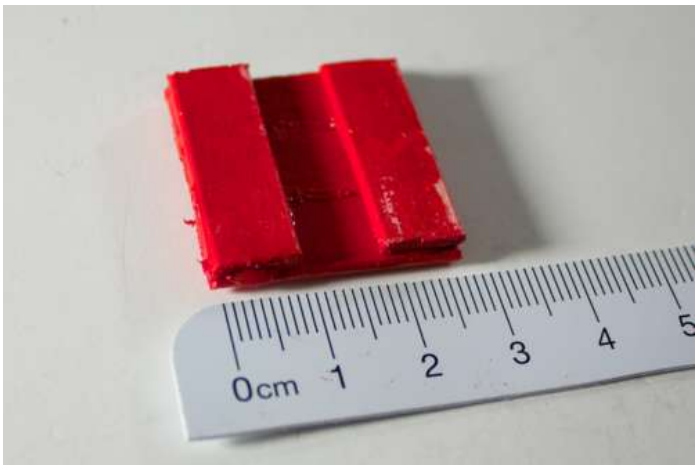
Place a piece of dual lock on the popsicle stick construction. On the bottom face of RoboBrrd we will attach the other piece of dual lock. This will allow us to be able to take off the main RoboBrrd structure without having to dismantle everything.

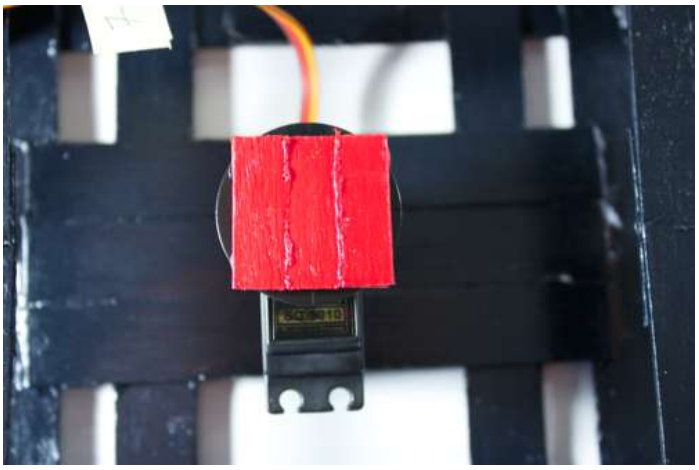


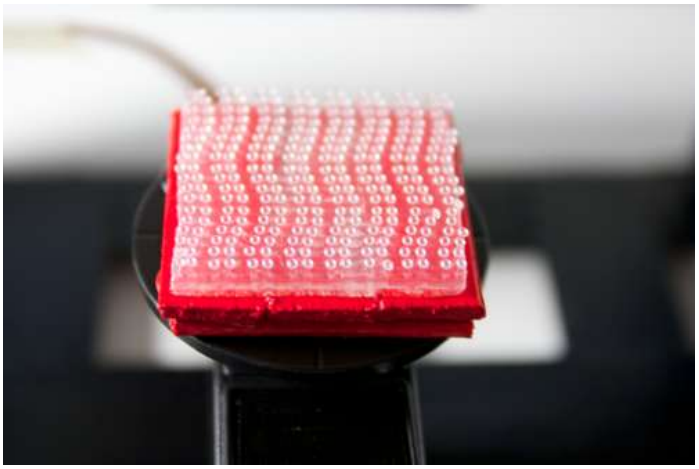


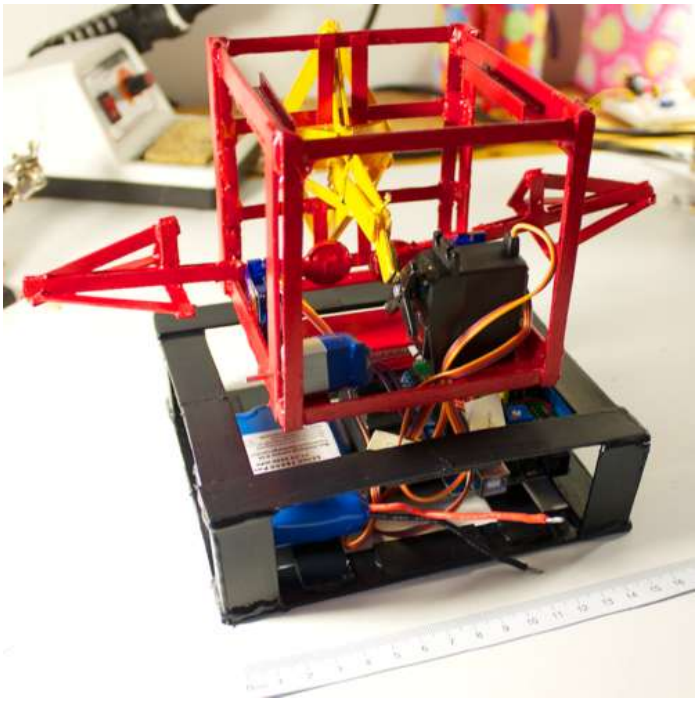


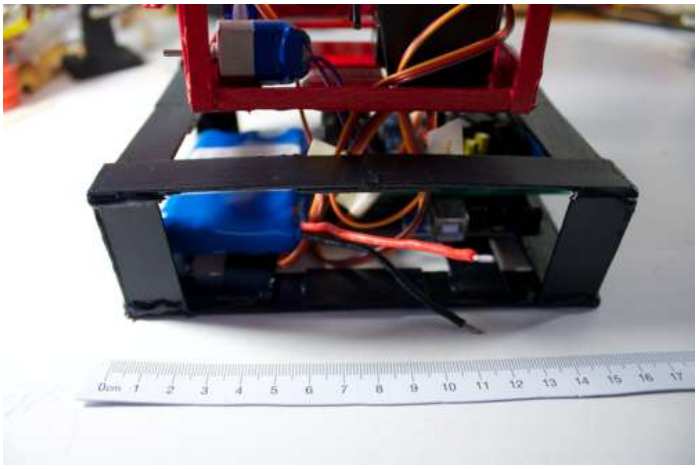

0cm
1
2
3


0cm
1
2
3
4









Step 9: Hula Hoop

The hula hoop uses a DC motor with two ends of craft sticks stuck together to push up on a ring of pipecleaners. The hoop of pipecleaners is attached to the back of the RoboBrrd structure, wound around the back and bottom face near the middle.

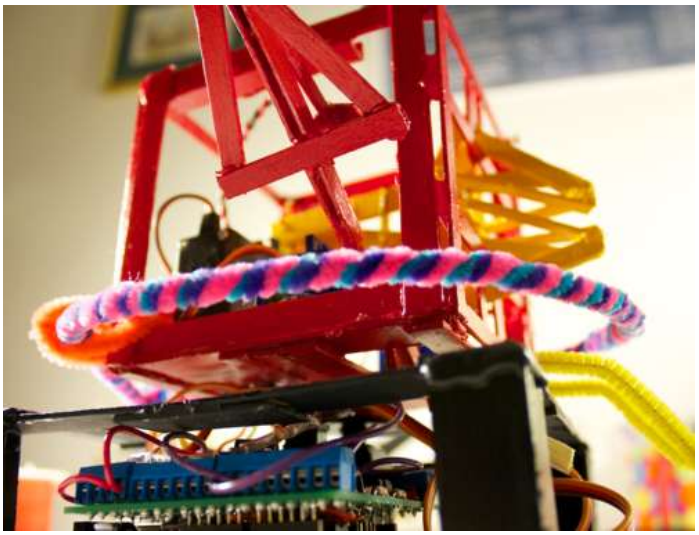
The DC motor for the hula hoop is mounted in the back left corner of the RoboBrrd structure. It uses the left face, bottom platform, and a little piece of the back face for support.

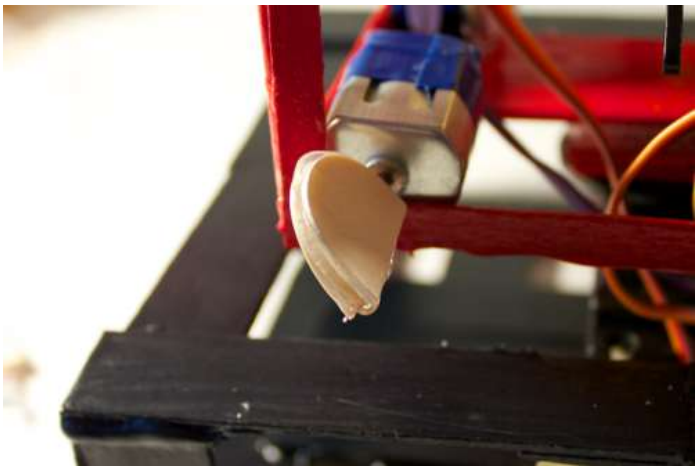
In order for the hula hoop to 'float' near the front of the RoboBrrd, we need to push down on the back of the hoop. We use a piece of sturdy wire to do this.

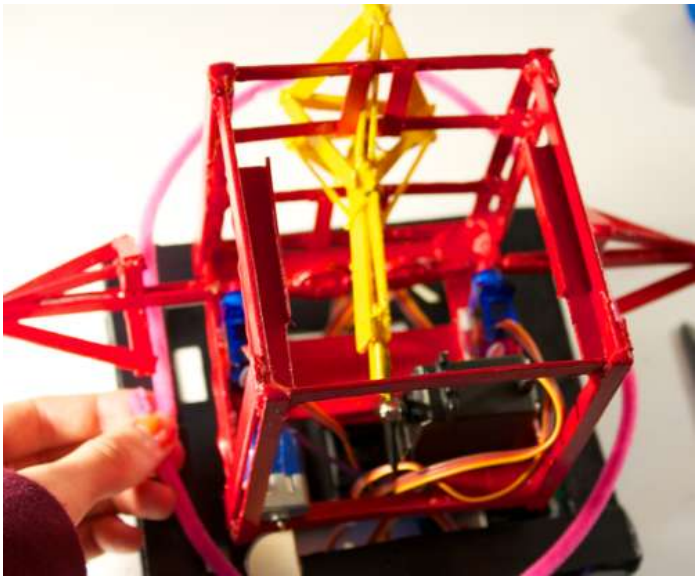
To lift the hoop up, we use some diamond shaped windings of pipecleaner, pictured above in orange. They are attached to the structure of the RoboBrrd more securely using hot glue. However, they are still able to be shaped to change the way the hoop jiggles.

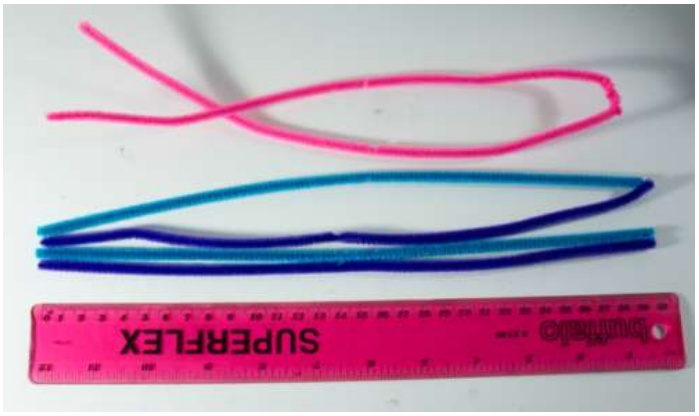
It will take some testing to determine the best shape for the hoop to be coming in contact with the craft stick ends on the DC motor. Ideally, the shape shouldn't allow the motor to stall too often.

Depending on how long you allow the motor to run, you can get interesting hula hoop results. We will be creating the circuit for the DC motor in a later step.

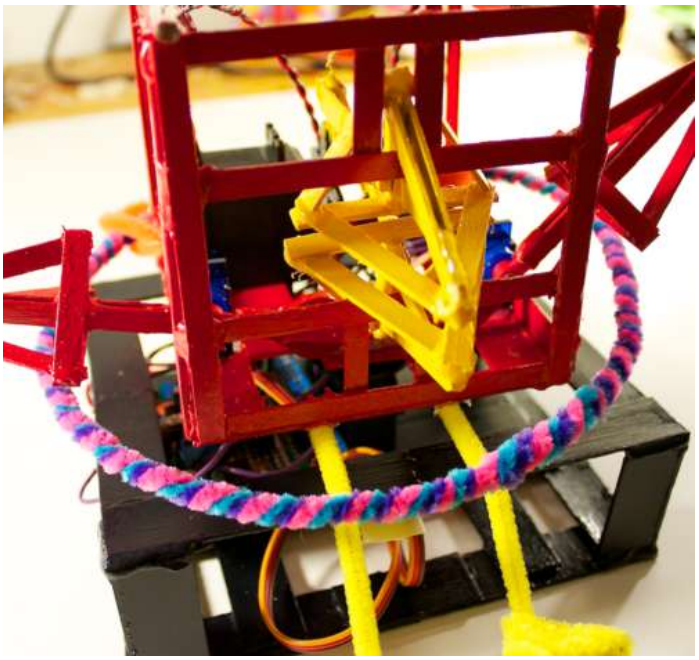


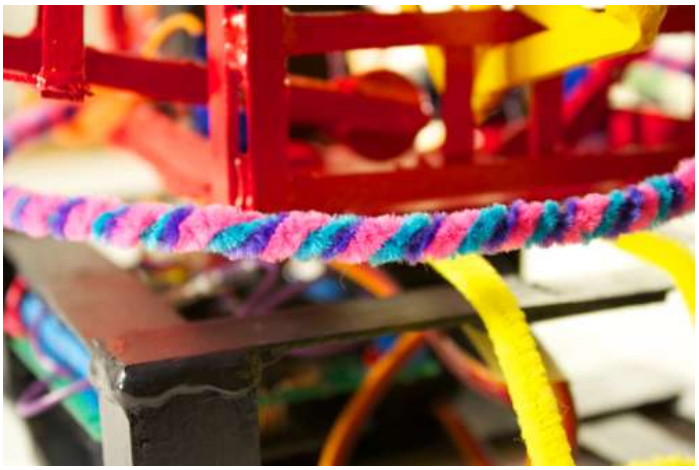


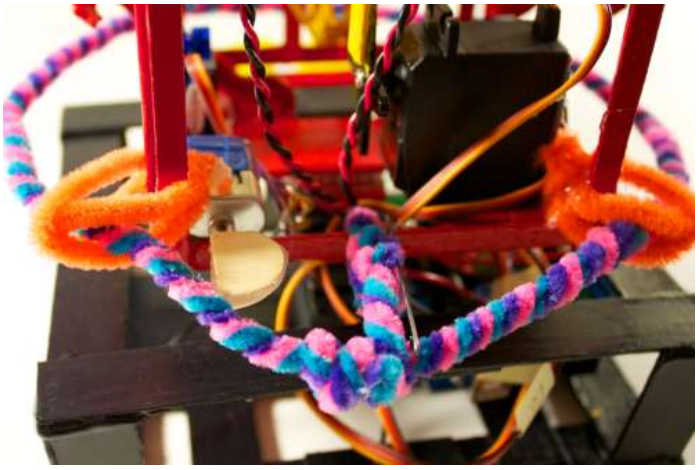


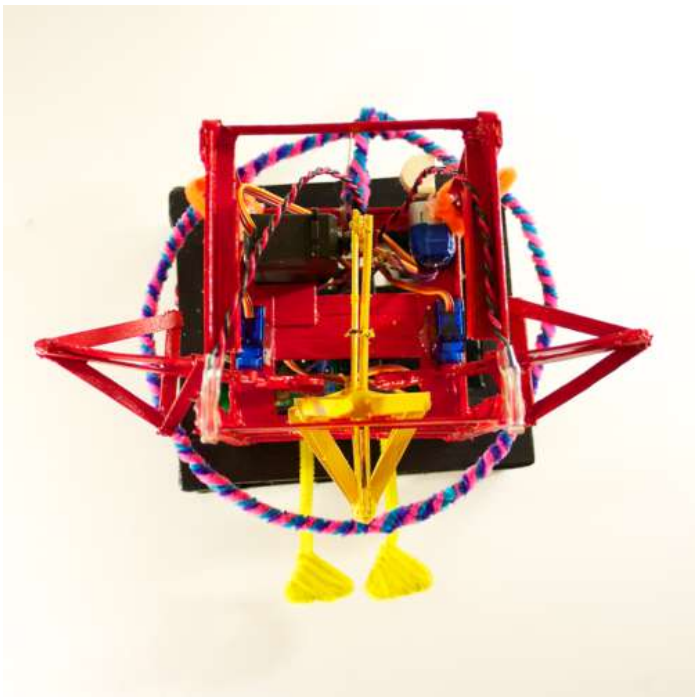

SUPERFLEX

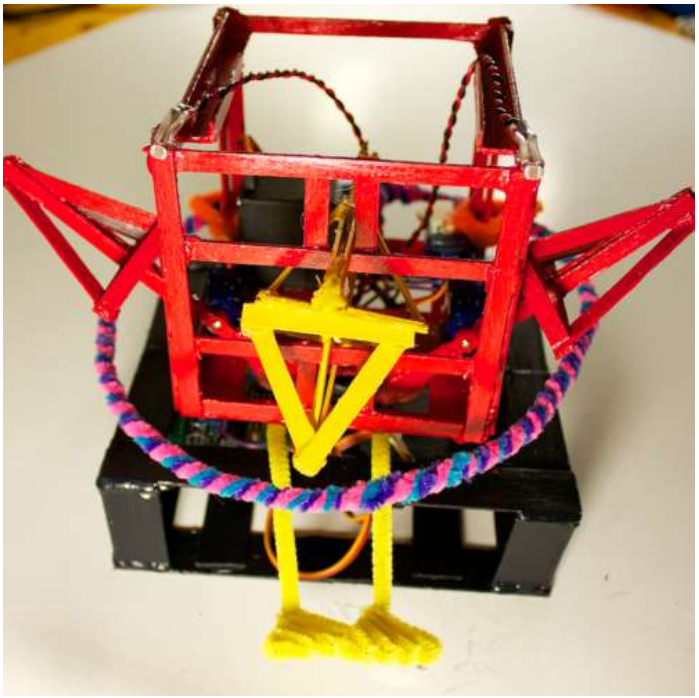

104
104











Step 10: LM317 Voltage Regulator

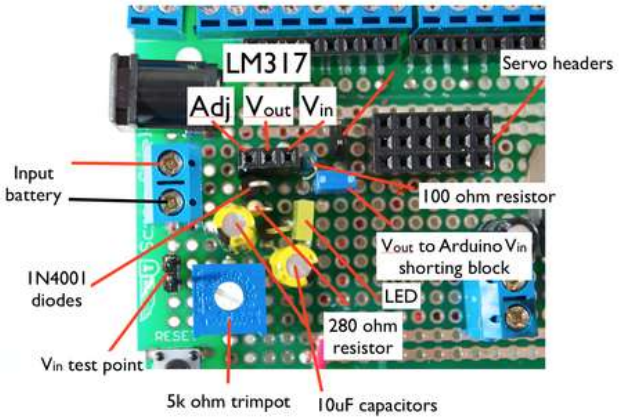
In order to be able to provide enough power for our servos, we need to have a separate power supply since the Arduino cannot supply the servos with enough current. The voltage range for these servos is around 5V. Since most batteries are not 5V we need to use a voltage regulator.

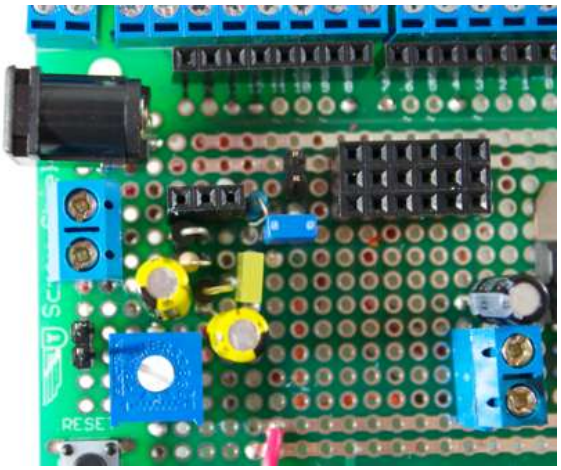
The LM317 is an adjustable voltage regulator that you can change to fit your power supply. You can find the datasheet for the LM317 here . We will be using the same circuit as on page 9.

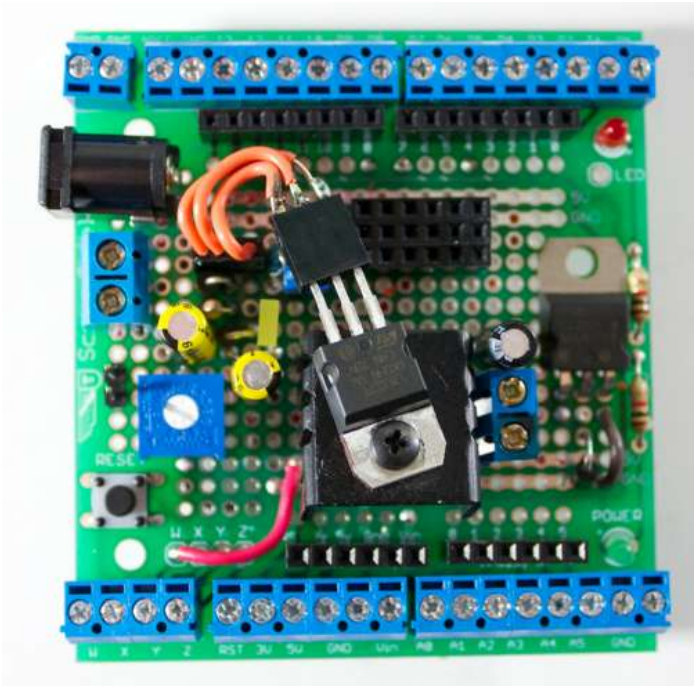
The important part for the LM317 voltage regulator is to adjust the 5k trimpot so that the output voltage is at 5V. The LED can be used as a visual indicator for the voltage. If the LED is extremely bright or burning up, then you may have too much output voltage and your servos may be at risk.

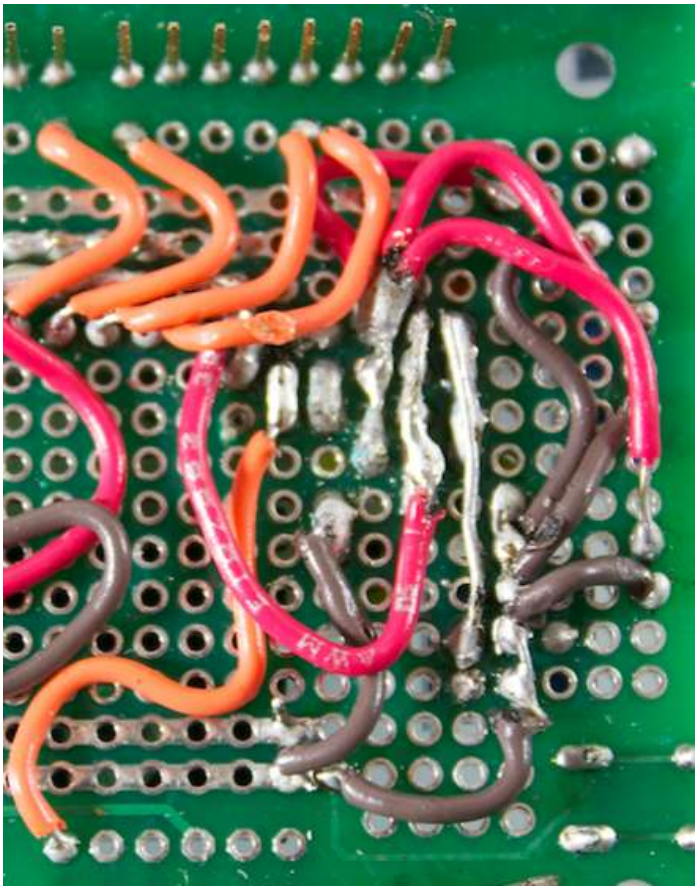
Depending on how much higher your supply voltage is compared to 5V, there may be a considerable amount of heat generated. In this RoboBrrd, I am using 11.1V, which means the LM317 heats up quickly. I use a medium sized heatsink to help dissipate the heat.

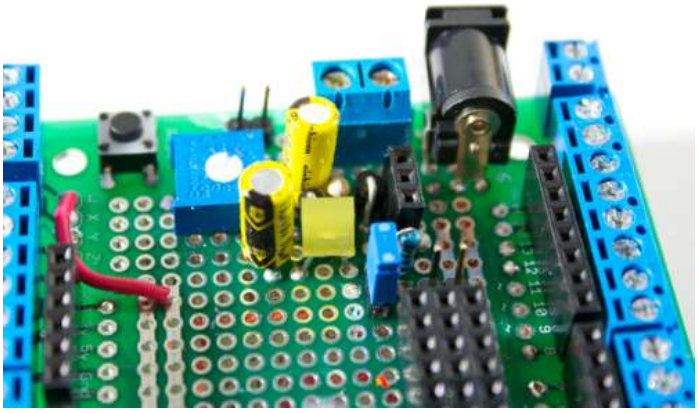
On the proto-screwshield we want to design for maximum fixability. Therefore, we use female headers for the LM317 to plug into in case of the unlikely event that the LM317 stops working and we need to replace it.

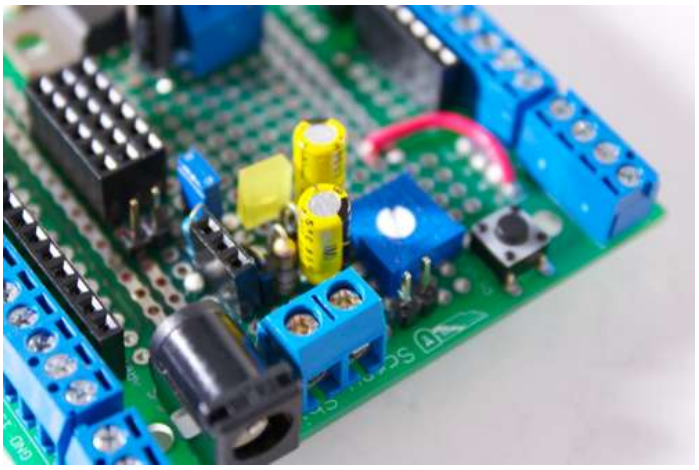

LM317
Adj
Vout
Vin
Servo headers
Input battery
100 ohm resistor
Vout to Arduino Vin shorting block
LED
1N4001 diodes
280 ohm resistor
Vin test point
5k ohm trimpot
10uF capacitors

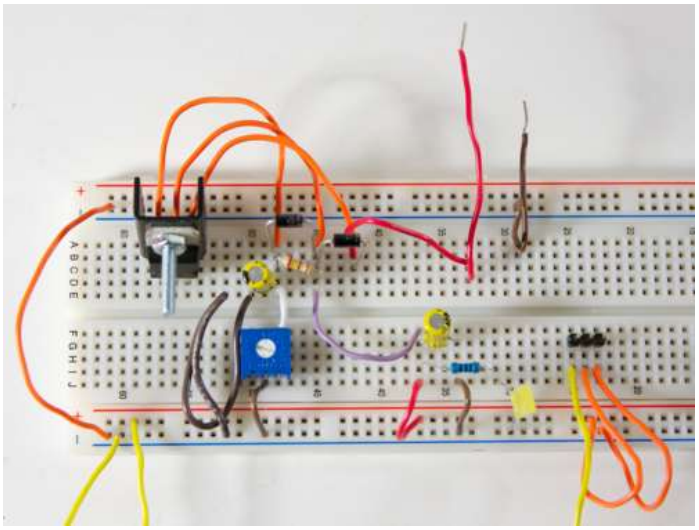


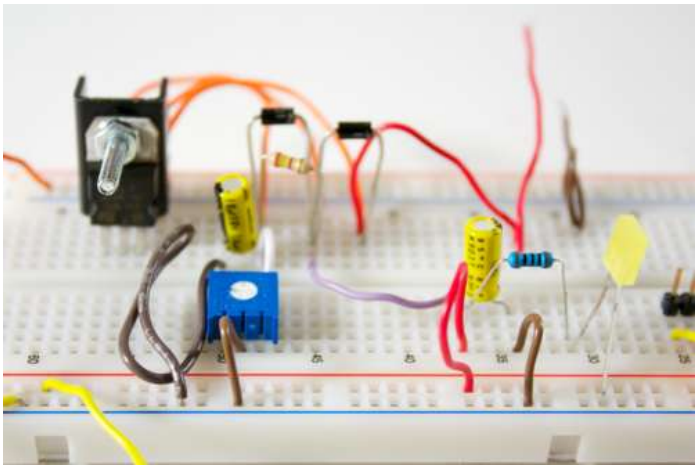












Step 11: TIP120 & DC Motor

For the hula hoop to move, we want to use a DC motor to push up the pipecleaners enough to provide a small jiggle.

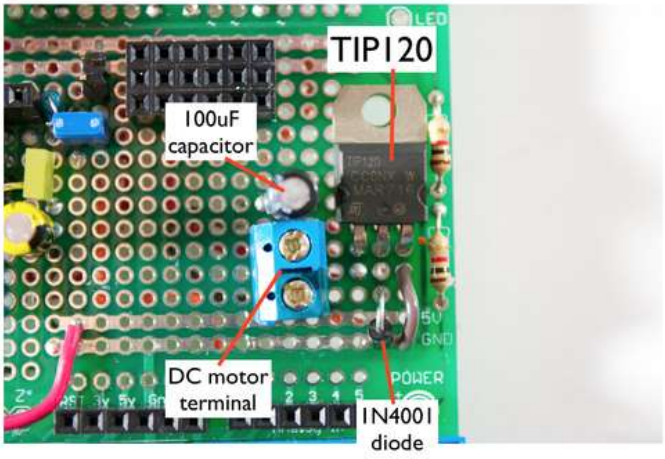
For this purpose, we only need to control the motor in one direction, so we can use a TIP 120 to switch the voltage.

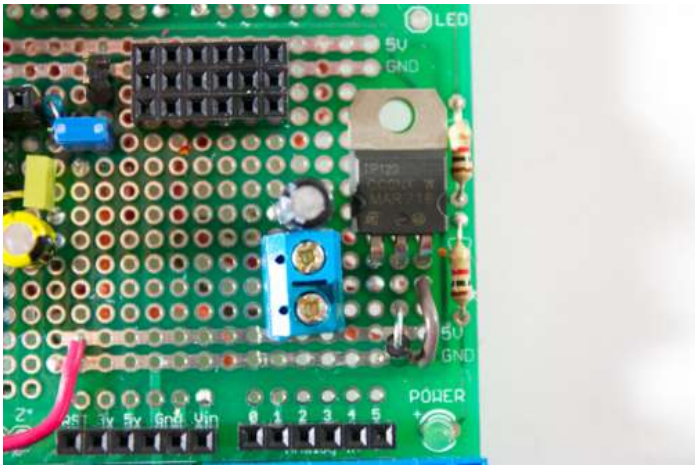
DC motors are noisy and need filtering. This is why we use a 100uF capacitor across the power supply. We use smaller 0.1uF ceramic capacitors near the motor to filter out sharp spikes.

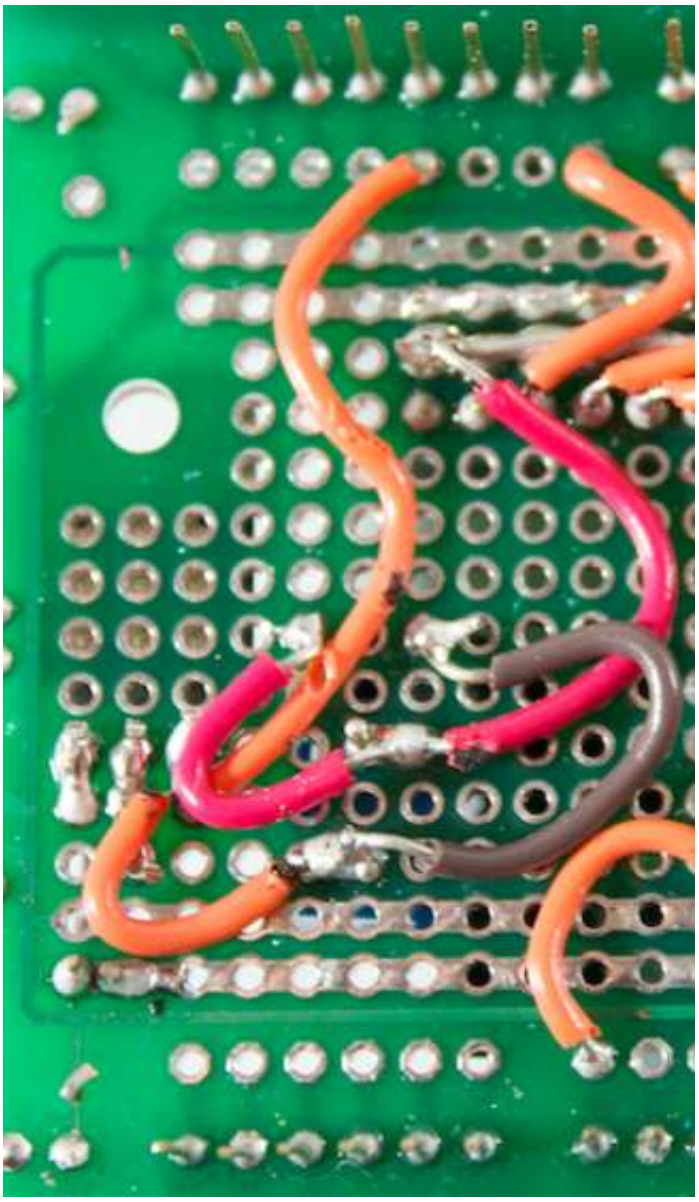
Although the TIP120 has an internal diode across the base and emitter, using another diode for our specific rating is always a good idea to prevent against back EMF.

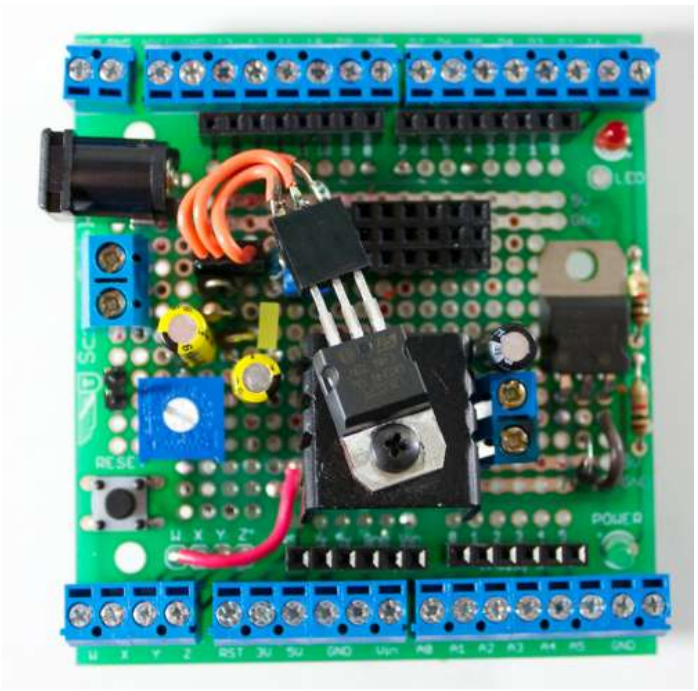
On the proto-screwshield we use a screw terminal to attach the leads of the motor. Alternatively, you can solder them on.

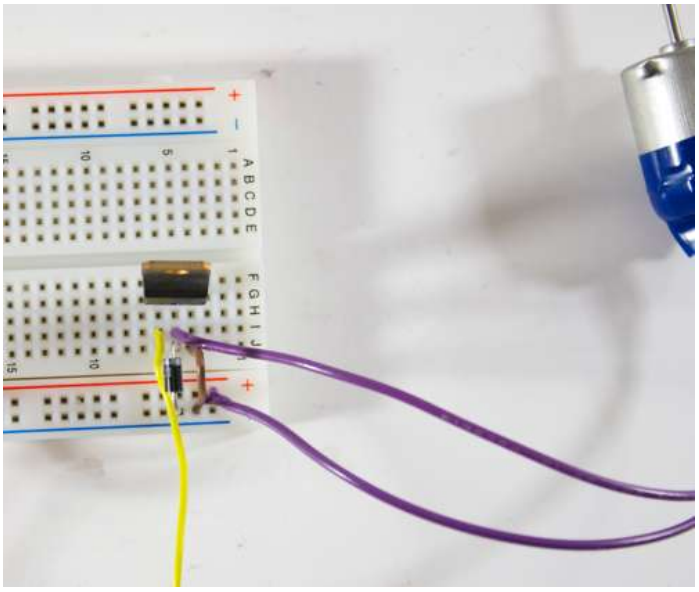
You will want to test how everything behaves and modify the circuit accordingly for your motor.

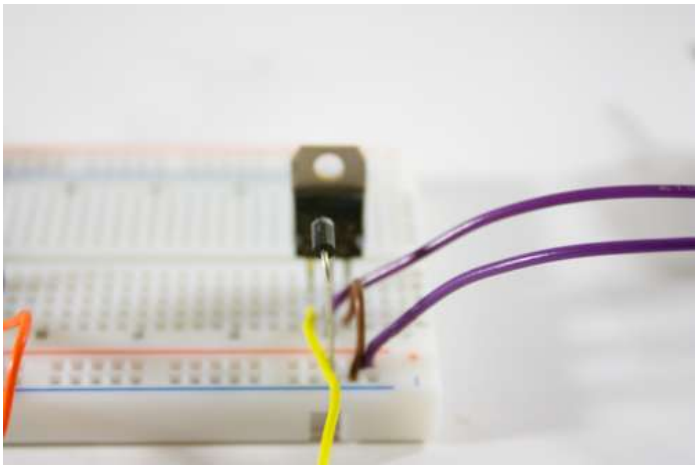







LED
RESET
POWER
W X Y Z
RST 3V 5V GND Vin A0 A1 A2 A3 A4 A5 GND


A B C D E
F G H I J



Step 12: Proto-Screw Shield

There is a really cool and elegant art form to laying out a circuit. In this RoboBrrd, I aimed to try and keep as many ‘straight lines’ as possible. I also aimed to use colour coded wire for specific lines, such as red for voltage, brown for ground, and orange for signals. We are using the WXYZ terminals of the proto-screw shield as the Arduino’s 5V in order to have room to connect the LED and LDR 5V wires.

In addition to the circuits created in the previous steps, there are some other components that have to be mounted to the shield.

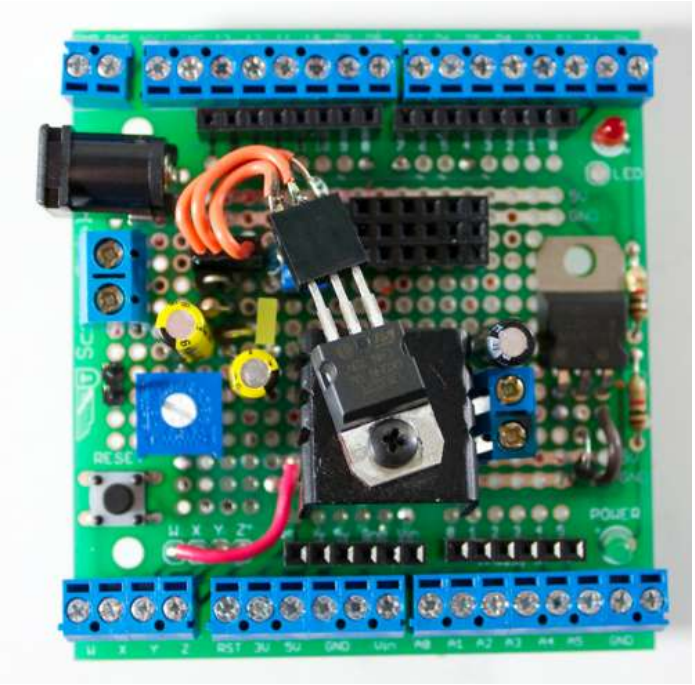
The servo headers we use are female as we can use double sided male headers to attach the servo cable to the headers. The side closest to the screw terminals is ground.

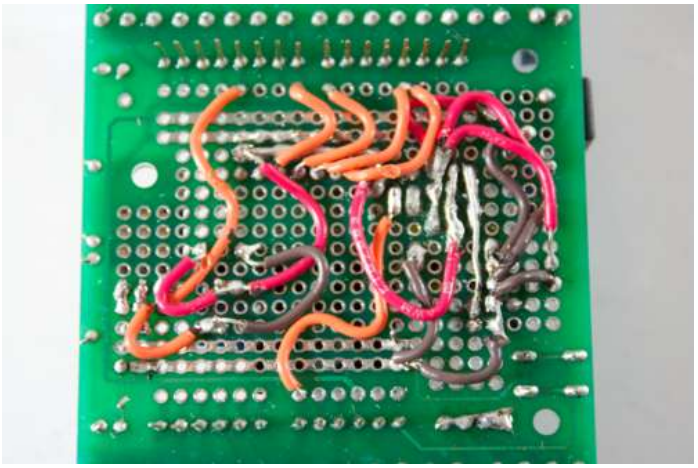
To attach our input voltage we have the choice of a DC barrel jack or a screw terminal. Either can be used, and having both allows for good flexibility for whatever power source we need to have connected.

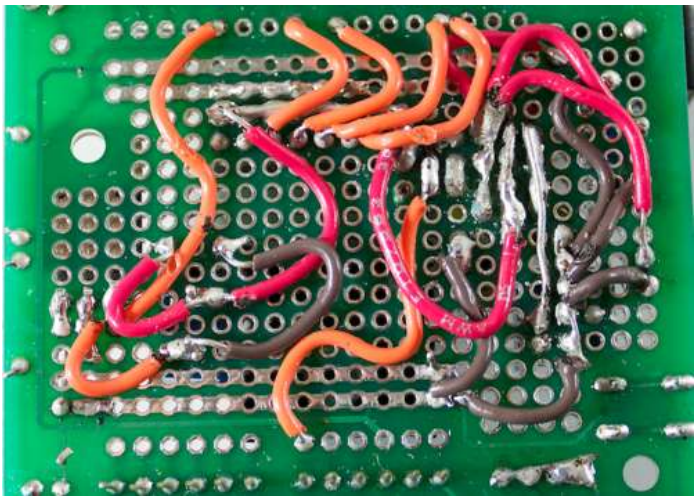
We also have a shorting block that can allow the voltage out of the LM317 to be used as the Arduino’s voltage in. While this method does work, we recommend using two power supplies as it will be safer for your Arduino.

There are two test points on the board that will allow you to test the input voltage and the voltage out of the LM317. These are very handy for debugging, as you can plug in a header and use a multimeter to measure the voltages.

There is a large space on the board that we use to place the LM317 and heatsink. Since the design was to be modular with the LM317, we use wires to extend the LM317 to this space. It works quite well as it is easy to remove the LM317, but it remains in place and is able to dissipate heat.







Step 13: Wiring

In order to make the wiring for the RoboBrrd as tidy as possible, ensure that all of your wires have some extra length. This will allow us to be able to wire in all of the connections outside of the base which will make it much easier.

The wiring for the RGB LEDs has 100 ohm resistors on each colour lead. We use heat shrink to ensure that none of the leads will be touching eachother.

The RGB LEDs will be mounted on the front face below the top coffee stir stick. There should be a little ridge on the LED that will assist with making it stick to the edge of the coffee stir stick. We will want to make sure that these are as perpendicular and straight as possible, as we will be adding the googely eyes to it.

The wiring for the LDRs is similar to the RGB LEDs. There is also a splice near the end of the wire that breaks out the signal wire from the ground. The ground has a 1k ohm pull down resistor on it, which is also heat shrinked to another short piece of wire to extend the reach of the ground.

The LDRs will be mounted on the top corners of the RoboBrrd structure. Gluing them should be trivial as there is enough room, and the heat shrink provides a good service to press into the glue.

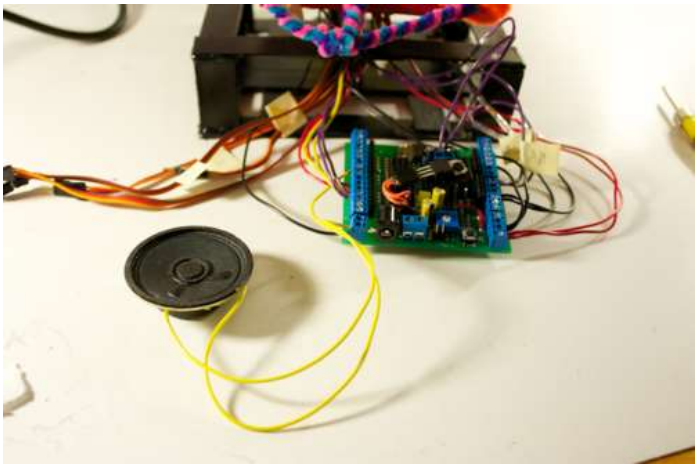
The wires from the LEDs and LDRs can be twisted together to save space. They should be routed into the ledges on the side faces in order to not interfere with the beak mechanism or wings. From here, the wires can drop down into the hole and into the base.

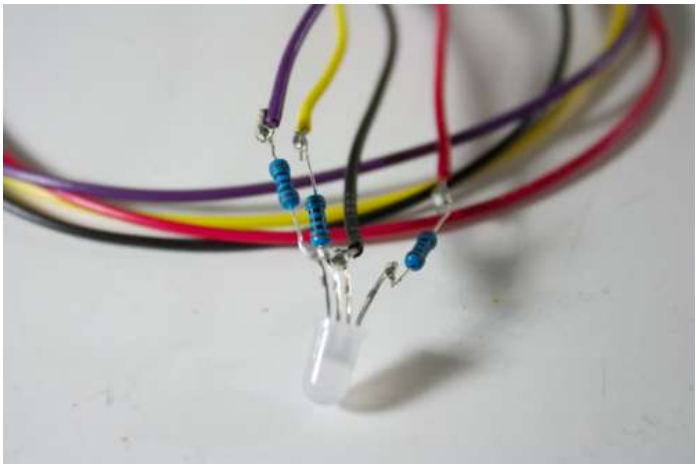
For attaching the wires to the proto-screw shield, I typically try to allow the wires that may most often need to come out near the top ‘layer’. In this case, it would be the servos and DC motor. Since they will be going over all of the other wires, we will leave them to last.

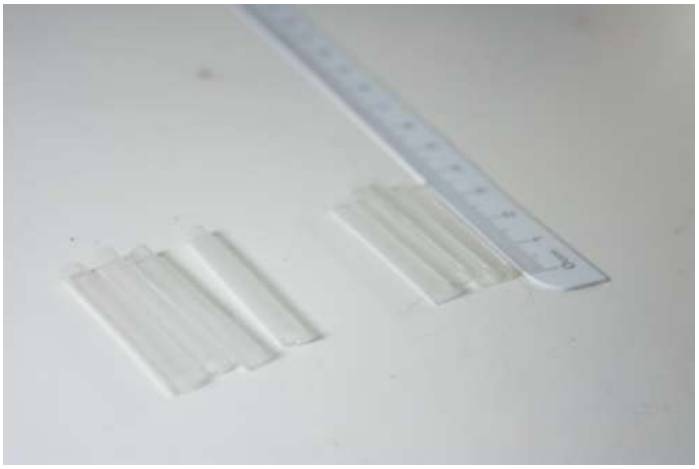
To make the wiring look neater, try to make the wires go all in one direction. This will also make it easier to place the shield back into the base. You may have to tilt the shield to have it go back into the base.

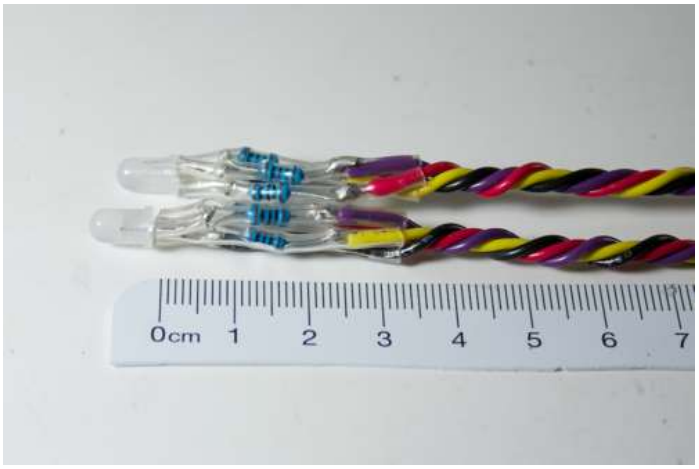
Once the shield is back in the base, attach the servo connections, as they may protrude higher than the base. Next, attach the Arduino through the side of the base.

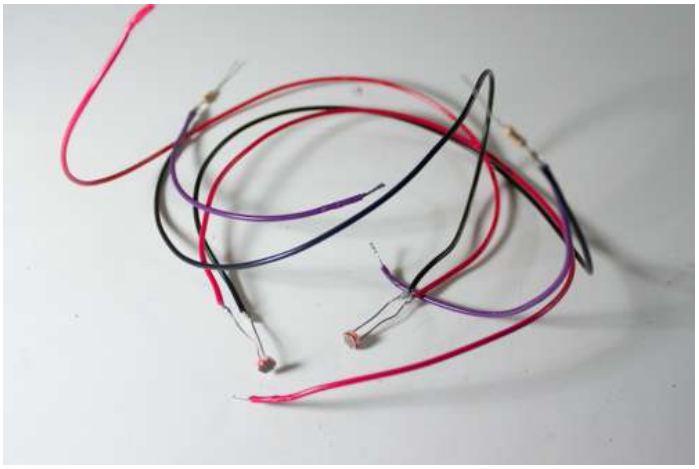
Give everything a test run to make sure that nothing was nicked while moving the shield back into the base. It will also be good to ensure that no wires may tangle the rotational servo.

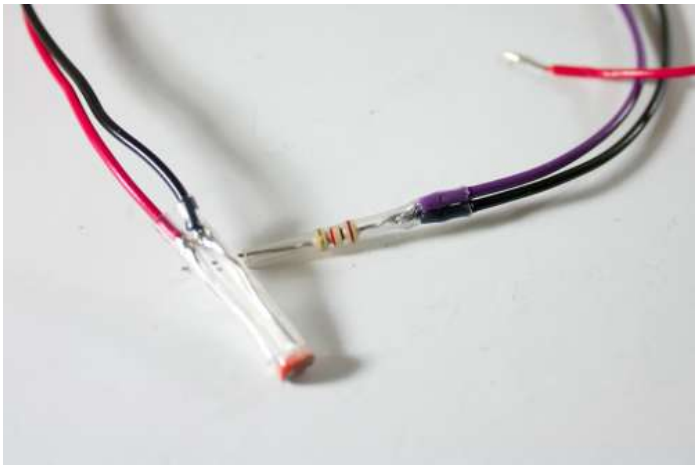




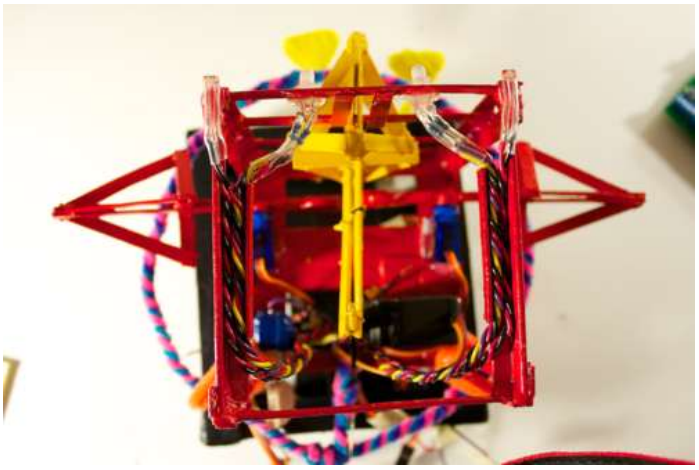


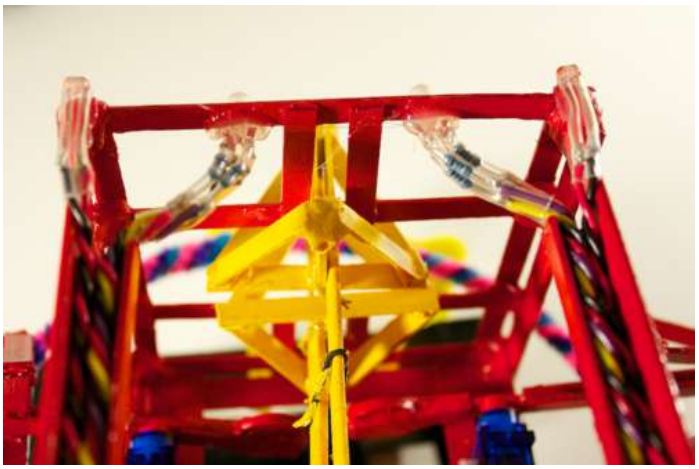

0cm 1 2 3 4 5 6 7

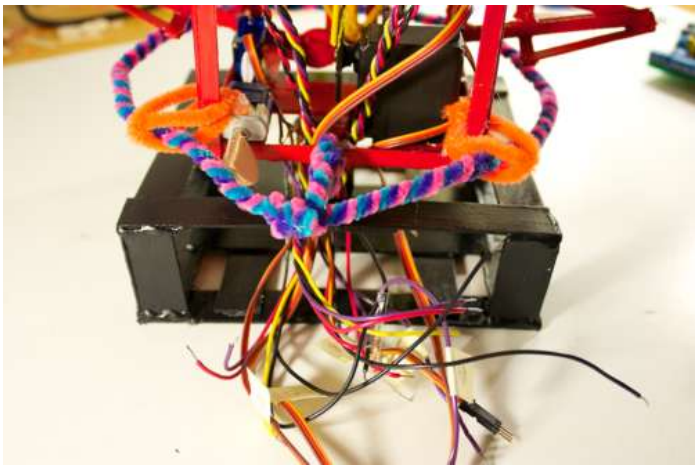


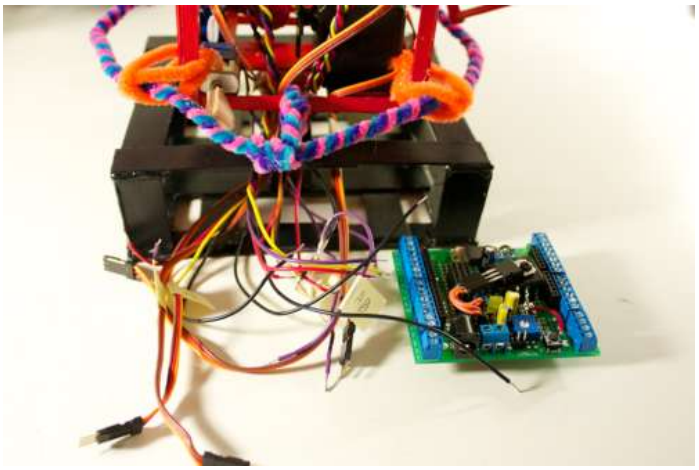


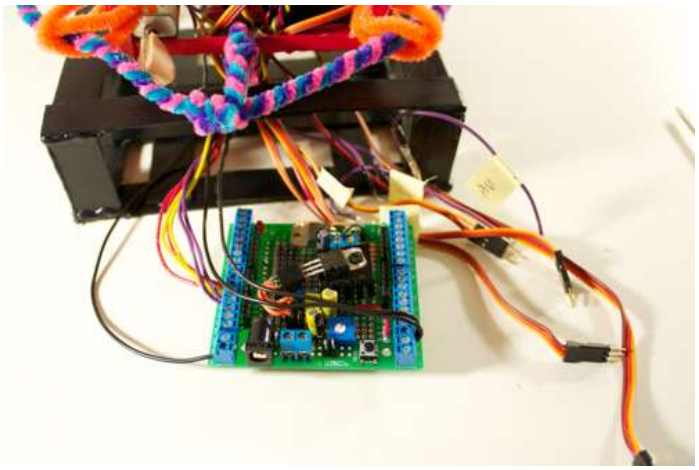

SUPERFLEX

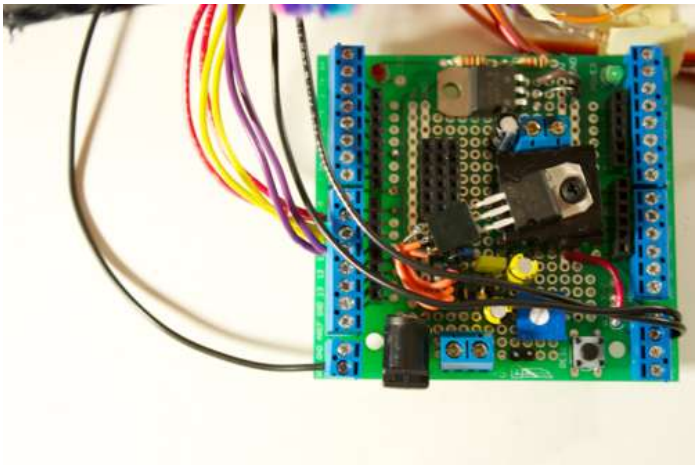


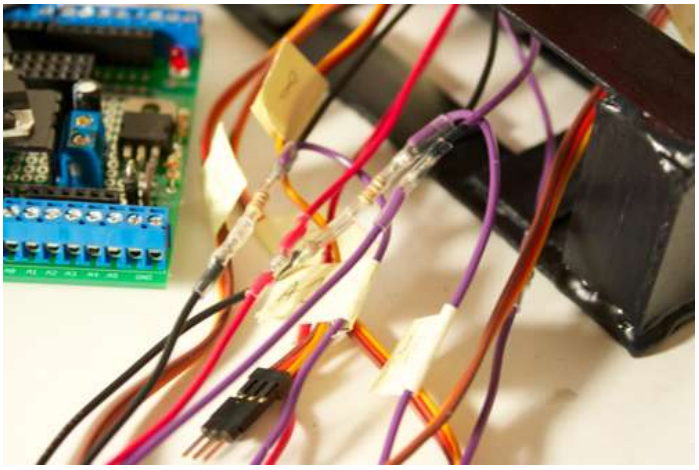


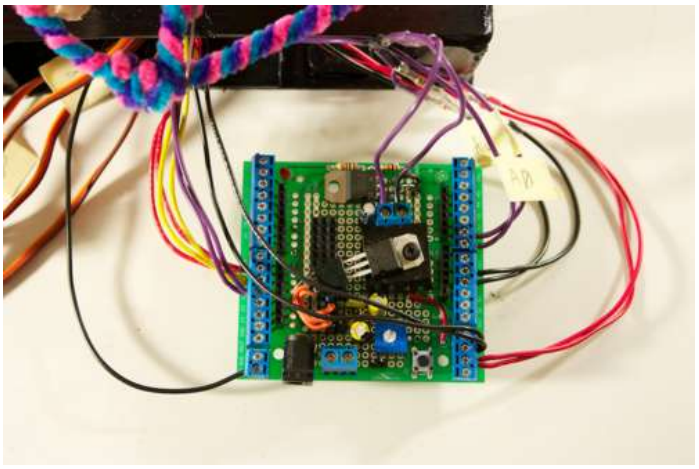


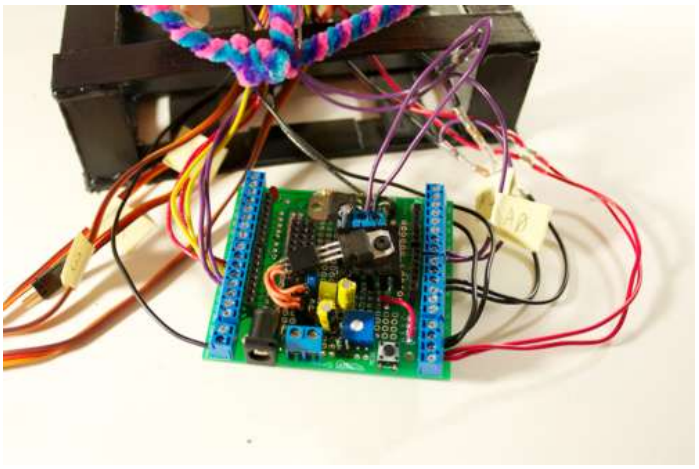


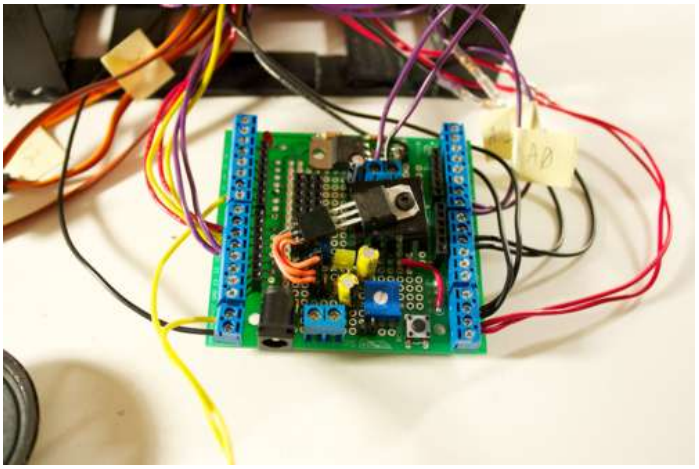


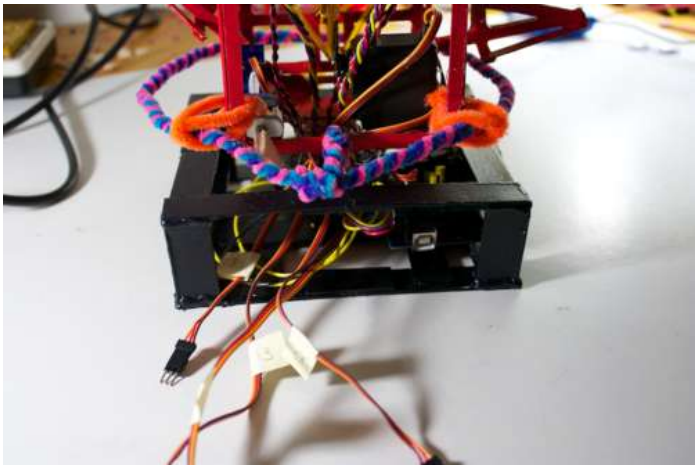


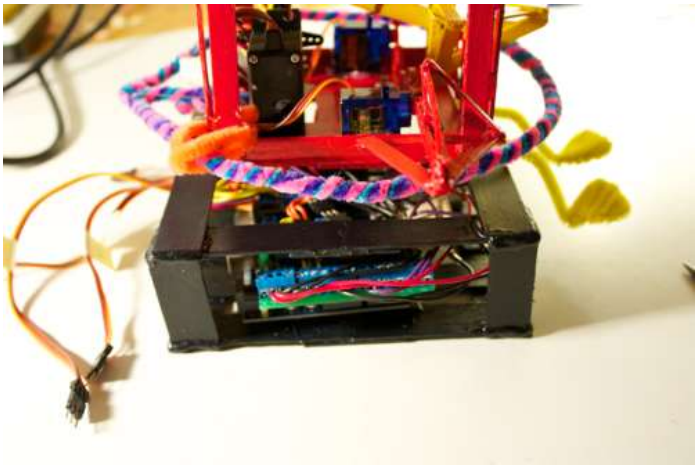


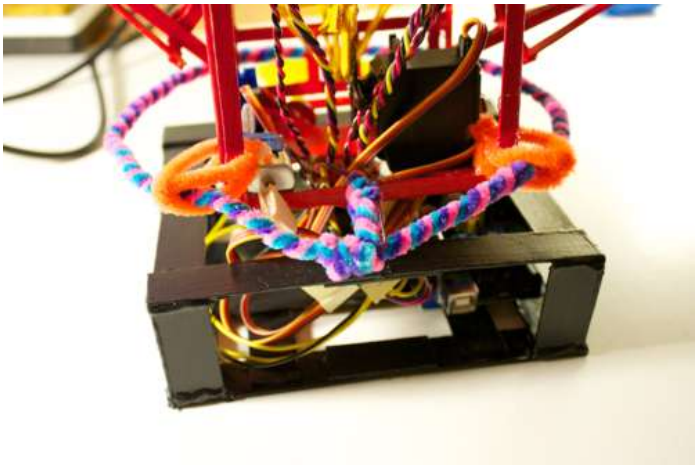


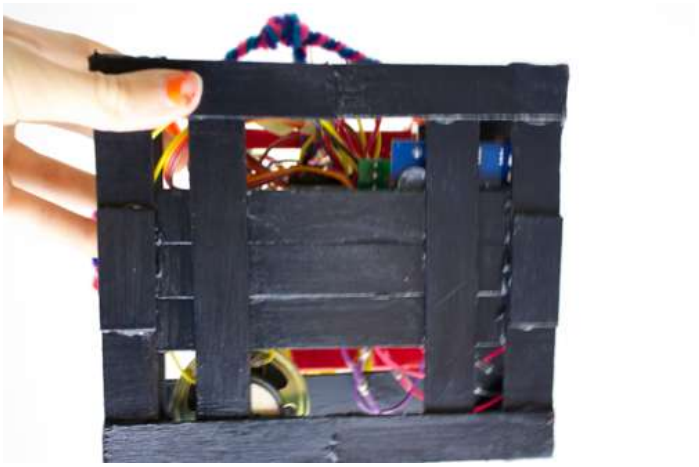












Step 14: Calibration & Testing

Now is a good time to go ahead and test all of the servos and calibrate them. We will also test the RGB LEDs, LDRs, and speaker.

For calibrating the servo, what we need to do is narrow in on the acceptable range of motion. Meaning, we need to find the upper and lower bounds for each servo. Here is an example of a function for calibrating the beak servo. The full source code for this RoboBrrd can be found on GitHub here .

The idea behind the calibration piece of code is that the servo will slowly start to move. When it reaches the open position, you can send an ‘s’ through the Serial Monitor, and it will print what the current position is. Remember this number as it will be the lower bound.

Then send ‘g’ for the servo to move the other direction. When it reaches the closed beak, send an ‘s’ again, and remember the number to be used for the upper bound.

Now you can hard code in these numbers, and be able to use them as a reference for when you are writing to the servo.

To test the servo, we will open and close the beak with a delay of 2 seconds in between. We should check and see that the mechanism is working properly, and that the beak is able to open and close.

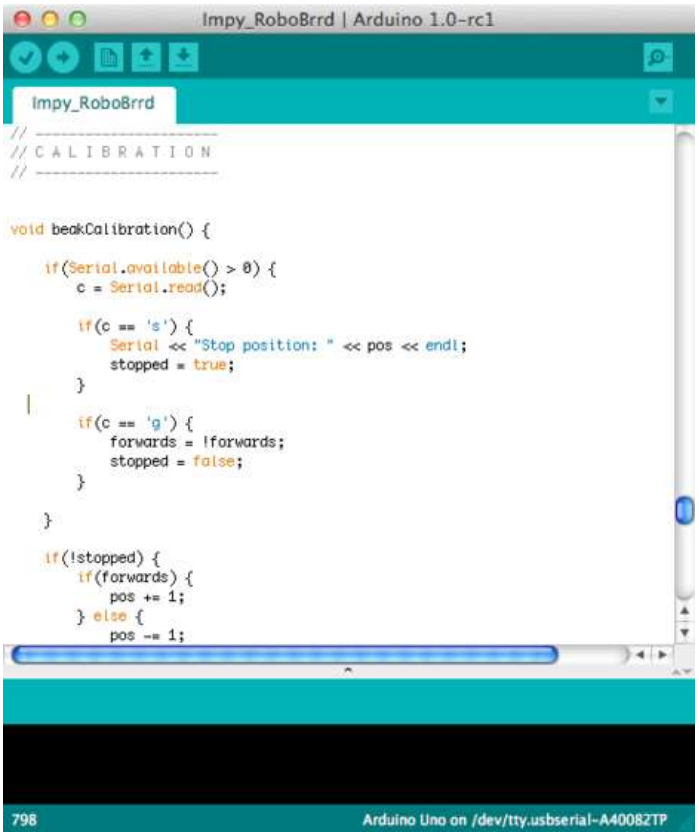
We will repeat this same process for the wing servos.

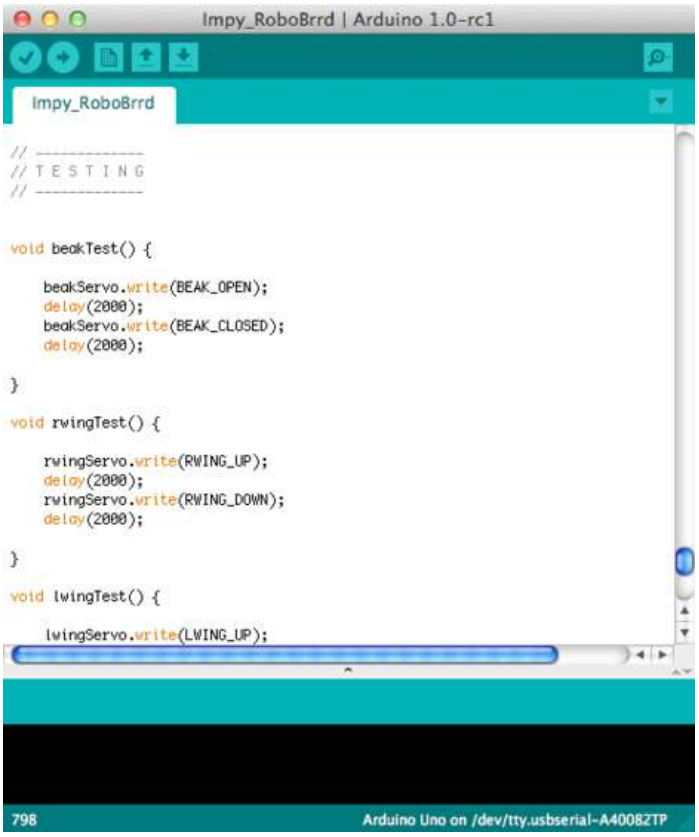
For the LDRs, we want to check to see if they are working. We will use the Streaming library to help us quickly print out the readings to the Serial Monitor. You can find the LDR test function in the code on GitHub.

Test the LDRs by moving your hand in front of them, or shining a bright light on them. Do the values have a noticeable difference? If not, check that the wires are plugged in.

The RGB LEDs can be tested by simply setting the various pins to LOW or HIGH.

Be sure to fix anything that is buggy or not working smoothly before moving on to the next step.





Step 15: Felting

Now we need to make the RoboBrrd not look like a skeleton anymore! We will be adding felt to the outside edges of RoboBrrd. The most difficult areas to felt are the front face and the beak.

For the front face, you will want to begin with a piece of felt that is a couple of cm larger than the size of the front face (10cm). From there, we will poke some holes for where the LEDs will appear through. When doing so, you will want to make sure that the top edge (with some excess felt on it) will be remaining straight, parallel with the top edge of the front face.

Now, we have to measure and estimate where we should be cutting for the beak opening. The best way to do this is by holding the felt on the front face, and marking it from the inside of the RoboBrrd. We will want to mark where the beak edges are, creating a diamond shape.

Begin by cutting a small ‘plus’ in the middle of the diamond. Place the felt on RoboBrrd’s face and see how much more longer the plus sides have to be. The beak should be poking through the plus a little bit. Keep on repeating this process until the entire beak can pop through, and the felt is flush with the front face. Now you can glue the felt onto the front face on all edges.

For the beak, we will need to divide the process into felting the ‘top’ sides of the beak (the one with the nose bridge), and the ‘inside’ of the beak (where teeth would be if birds had teeth).

Take a piece of felt and measure to see how long it would need to be to wrap around the entire top side of the beak. Cut it to the length, and add a little bit of glue in the middle near the side of the longest edge. Stick this onto the bridge of the nose as close as you can get it to where it meets the front face. If it sticks to anything other than the beak, be sure to cut it.

By pressing the felt across the nose bridge and onto the sides of the beak, you can create a nice smooth shape. You will want to ensure that some of the felt is attaching to the edges of the beak, which may be hiding behind the front face, on the inside of the RoboBrrd.

When everything is smooth, add glue to the edges of the beak and press the felt onto it. When it dries, cut away the excess felt, leaving an extra 3mm. We will wrap this extra felt onto the inside of the beak. Add some glue to the inside of the beak nearest to the edges and press the extra felt into it. The tip of the beak may need some additional gluing.

Repeat this same process for the other beak half.

Now we will be adding the felt for the inside of the beak. We will want the beak to be open at its widest point for this. Take a piece of felt and measure how wide it should be and how long. For the length, add an extra centimetre. Start by gluing the top-center of the piece of felt to the tip of the top beak half.

From there, you will want to move the felt down, towards the tip of the bottom beak. There should be some extra slack in the middle creating a bump. The bump should be about a little less than a centimetre tall. Glue the bottom-center of the piece to the tip of the bottom beak half.

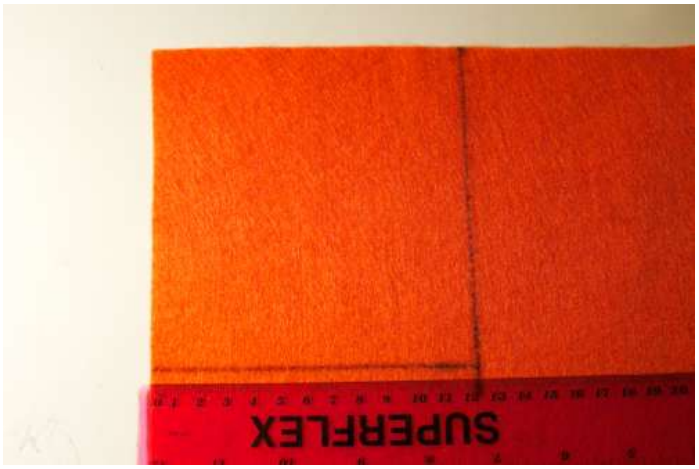
Now, press the bump of excess felt through the opening between the two beak halves. This is used for creating a seamless feel between the two beak halves. Ensure the felt is laying flush with the insides of the beak. When it is, add glue to the inside edges of the beak and press the felt into it. Cut off all the excess felt, and it is done!

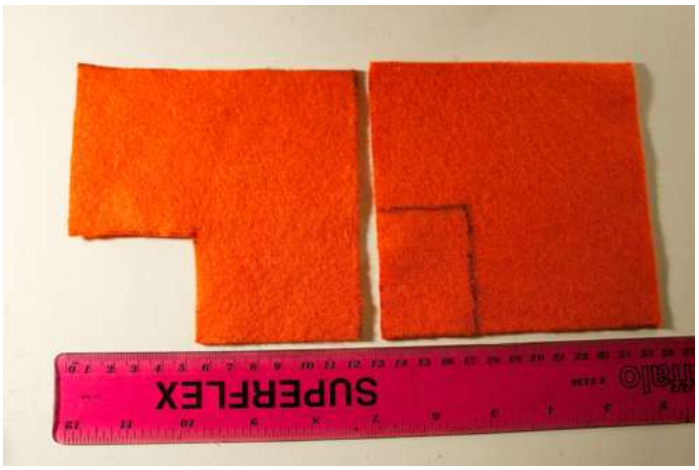
For the side faces, create a 10cm by 10cm square (with a bit extra if you want to be able to trim it to length), and add an opening for the wing servos. Be sure to only glue this along the top edge, and along the side edge to where the opening is. We do not want glue on the bottom or opposite side edge as we will need to lift the flap to be able to see the insides of the RoboBrrd.

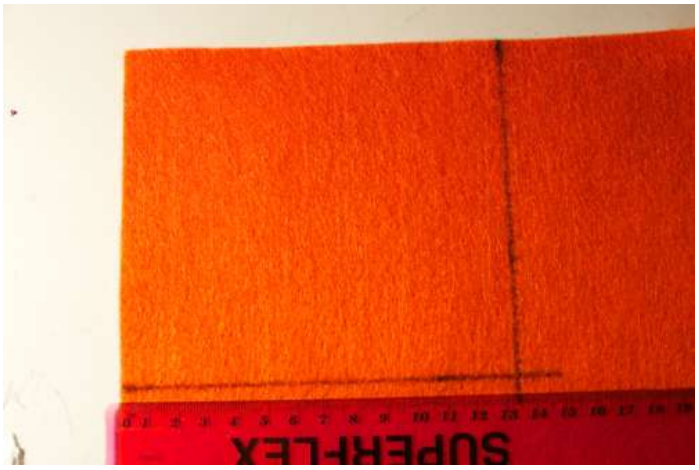
The back and top face is a similar approach. Only the top edge will be glued on, so we can lift it to reveal the inner workings of the RoboBrrd.

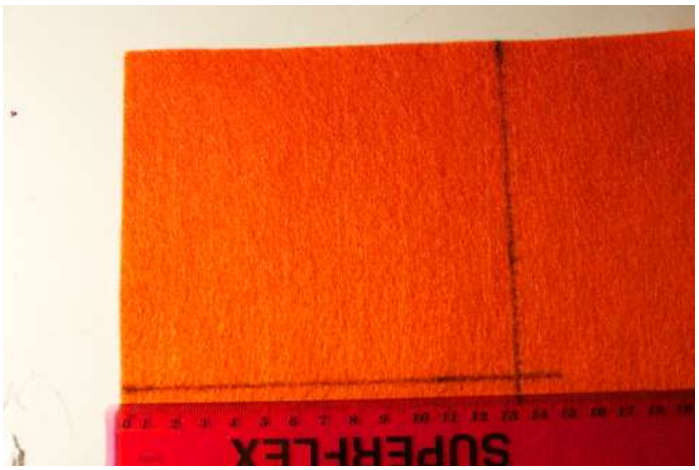
For the wings, we will do the same process as we did with felting the top of the beaks. Cut the felt to the proper length, then add glue to the top of the diagonal piece. Then smooth out the felt for the other edges, add glue, and press the felt into it. Cut away the excess felt, leaving an extra 3mm to be used to wrap around the the bottom of the wing. Add glue and press the extra felt into it, and the wings are done!

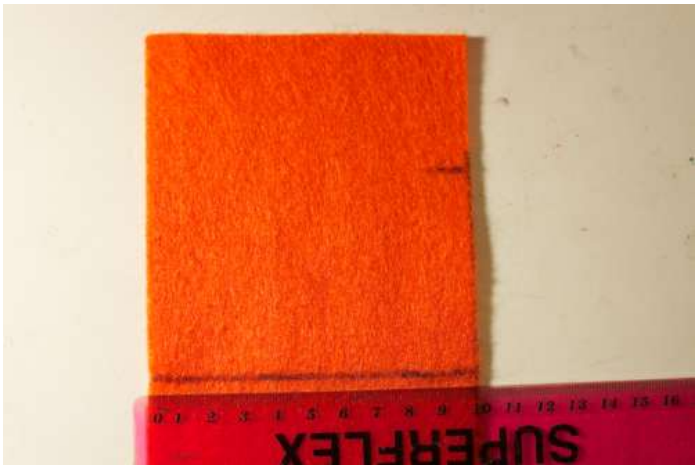
When everything is felted, we can move on to the next step.

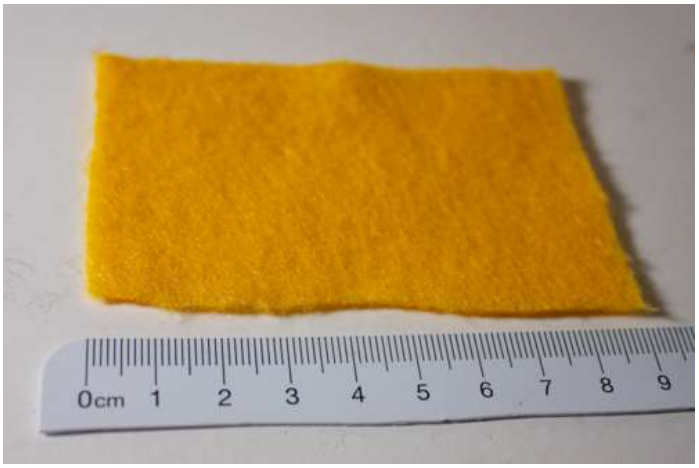


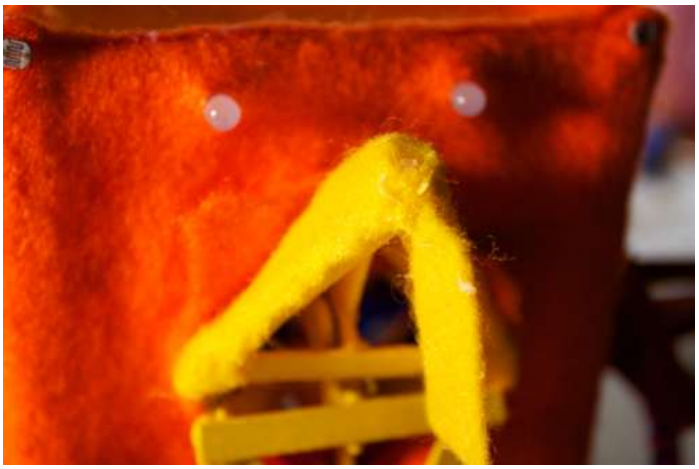

SUPERFLEX

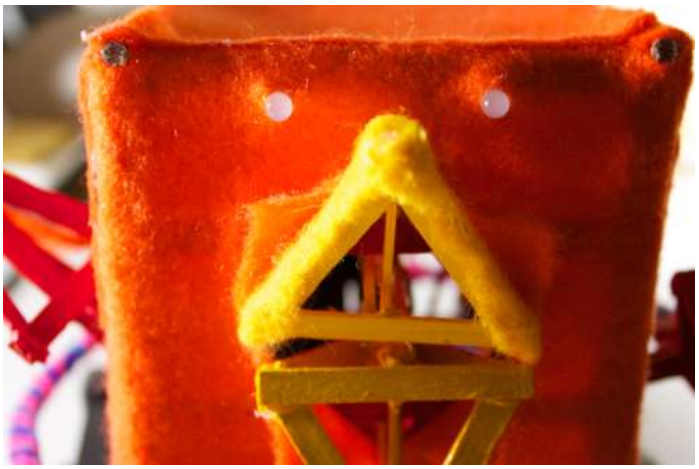

SUPERFLEX


SUPERFLEX


SUPERFLEX


0cm 1 2 3 4 5 6 7 8 9






0cm
1
2
3

Step 16: Foaming

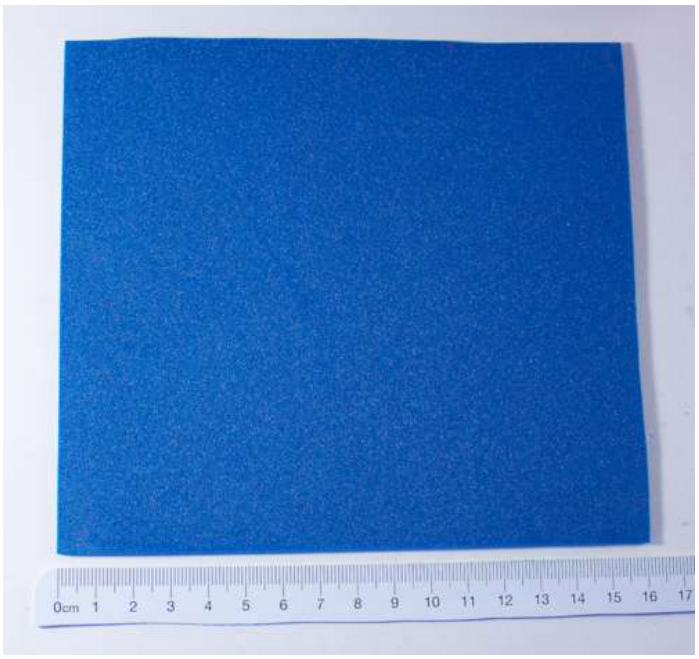
We should cover the base to ensure no wires get snagged and no electronics fall out. We'll use some foam for this, cutting two pieces each for the bottom/top side, the left/right side, and the front/back side.

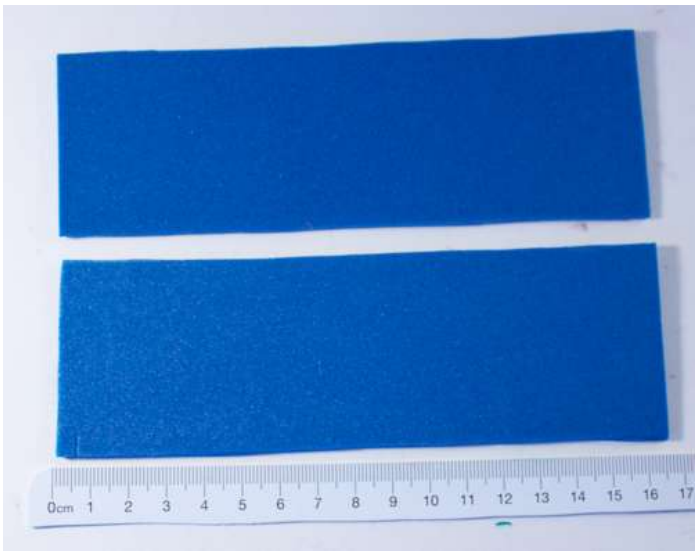
For the bottom, left/right, and front/back sides the cutting is straight forward- cut it to the length. For gluing all the sides on (except for the bottom and front), only glue the top edge as you will be able to then lift the foam and see the electronics inside. We can use velcro later on to secure the foam pieces to the base to keep them closed.

The top side will be similar to a skirt, where we will wrap it around the base. In order to make this, cut a rectangle the same size as your base (approximately- we can always trim later). Find the center of the rectangle and cut a small square of about 4cm. Cut a straight line leading from the center to the middle of the longest side.

Wrap the 'skirt' around the base of the RoboBrrd and test the rotation. If the foam is getting caught on anything, expand the dimensions of the square and test again. Once it all fits, glue the front side onto the base. From here, we can add velcro to the other sides to secure it in place.

The velcroing can be done as you see necessary. For this RoboBrrd, we used 3 small squares per side near the corners and in the middle. This ensures for a smooth look.




0cm 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17

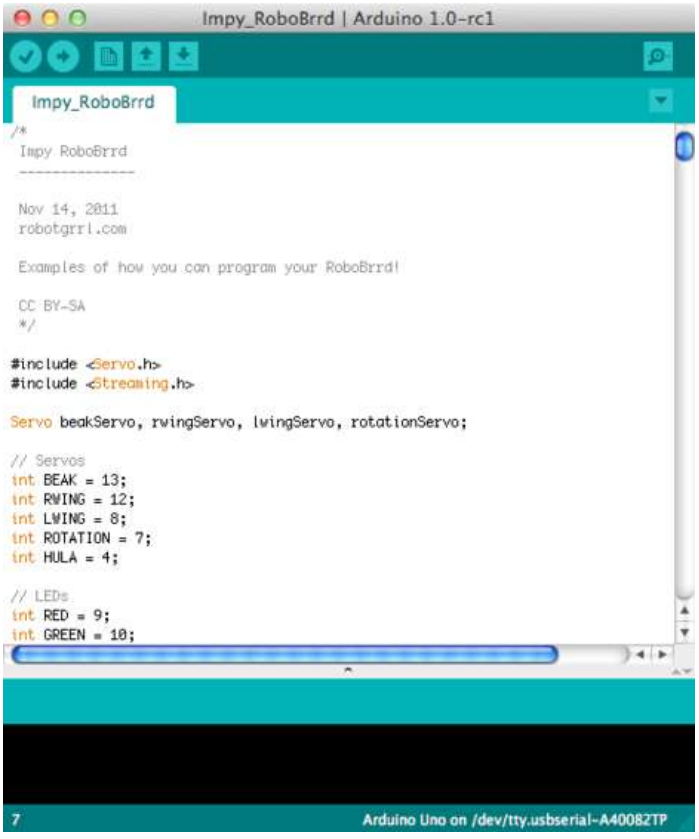
Step 17: Programming

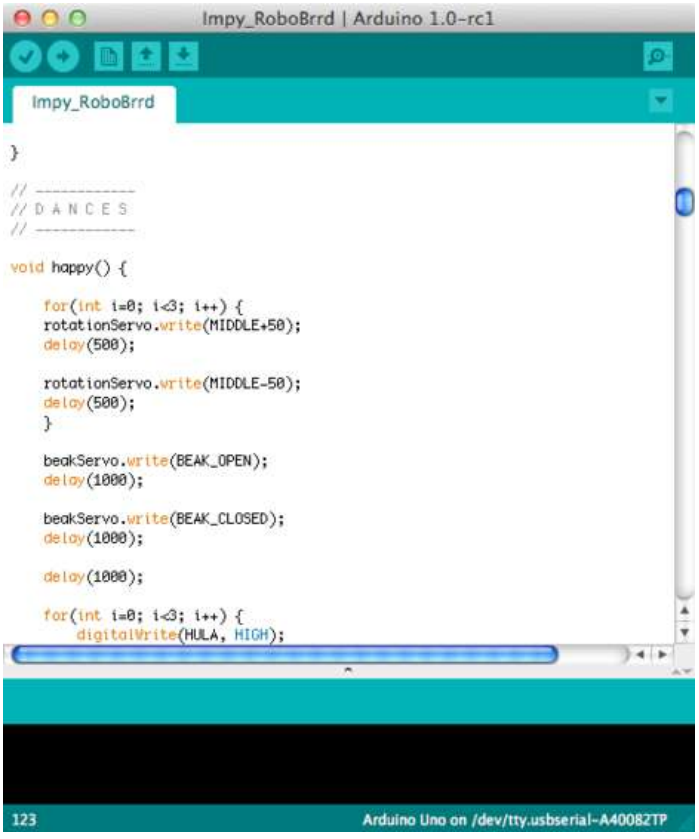
Programming RoboBrrd is the fun part! Here, we can tell the RoboBrrd what to do and how to do it! The best way to learn the code is to look at my example on GitHub . It is extensive, yet very clear.

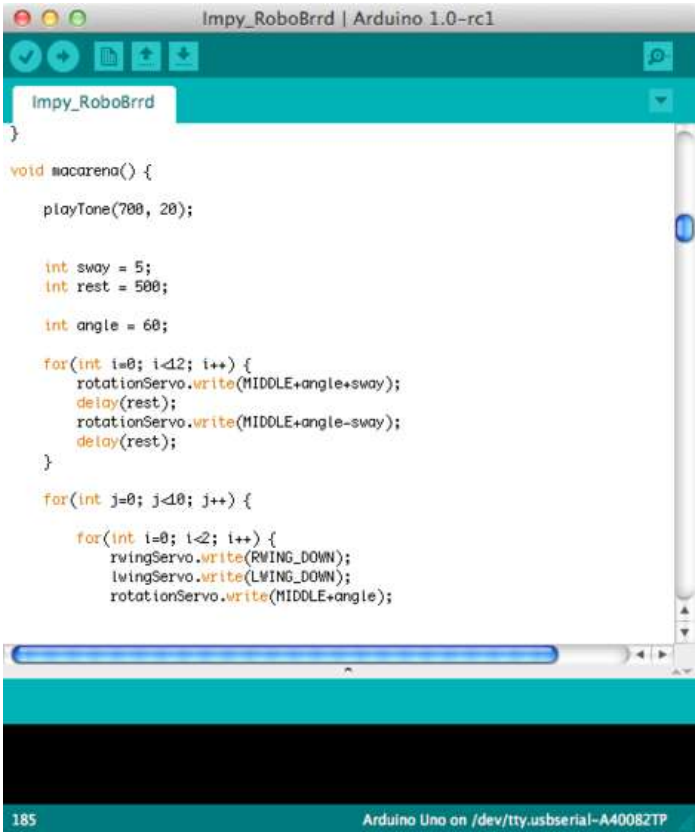
The behaviours of the RoboBrrd are only limited by your imagination. Since there are two LDRs on RoboBrrd, we can create a simple photovore 'light chasing' (or hiding) behaviour that is quite fun.

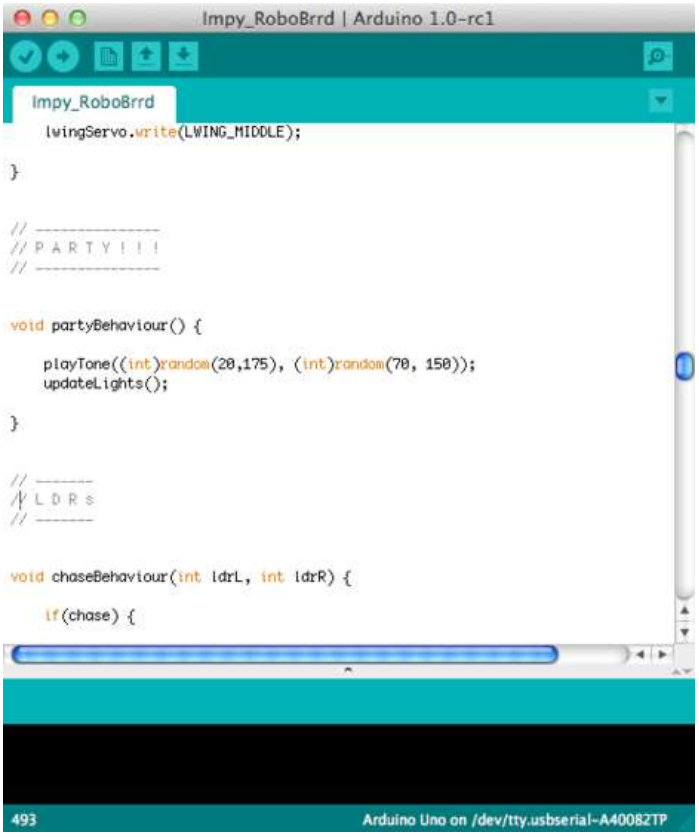
The dances for RoboBrrd usually consist of lots of rotational servo movements, as well as wing flapping.

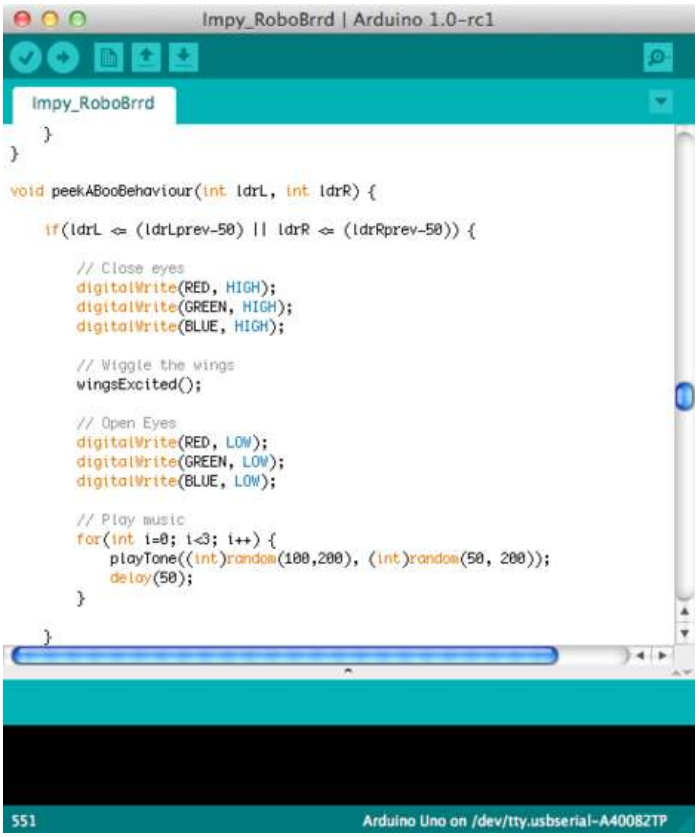
What are you waiting for? Download the code, open it in Arduino, and begin tinkering :)

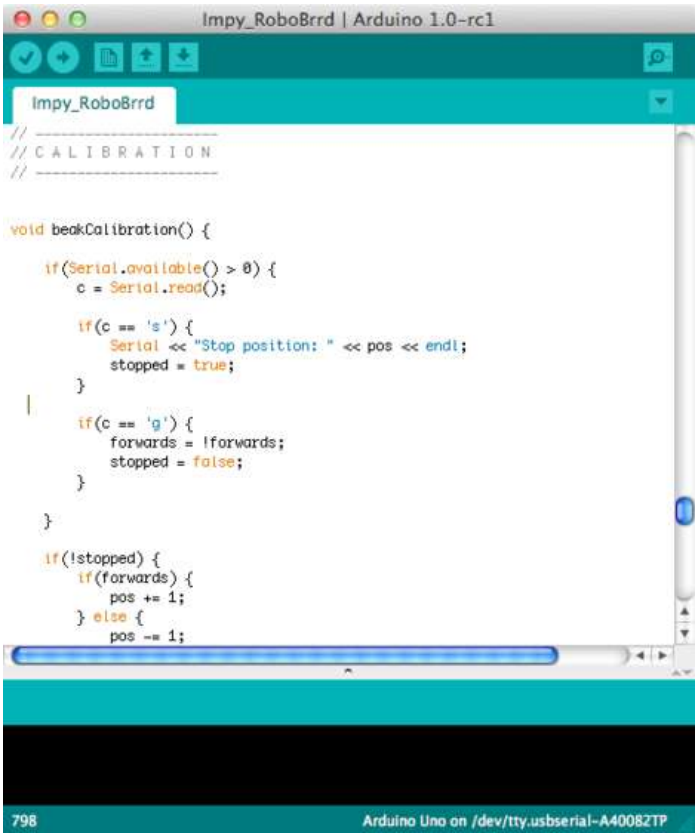


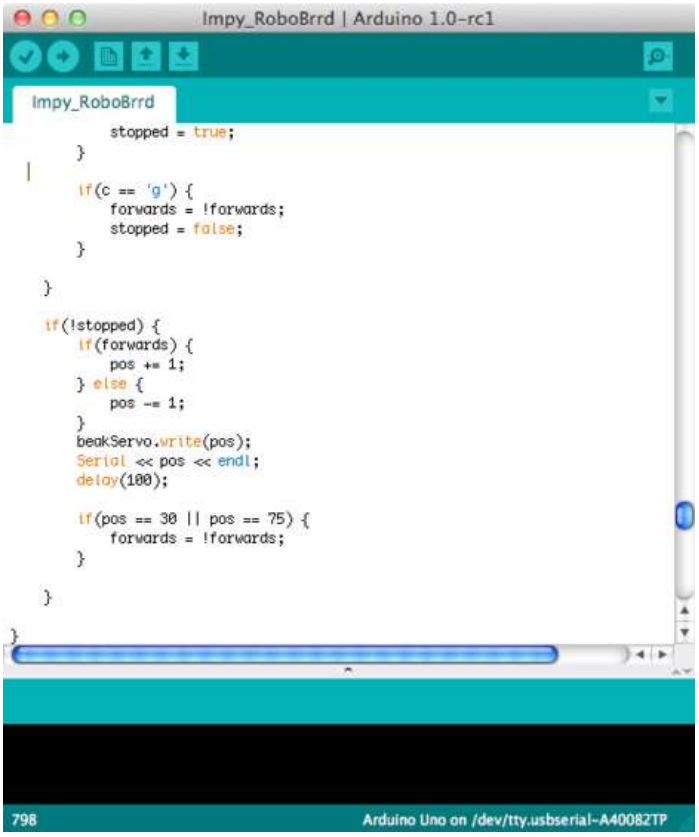


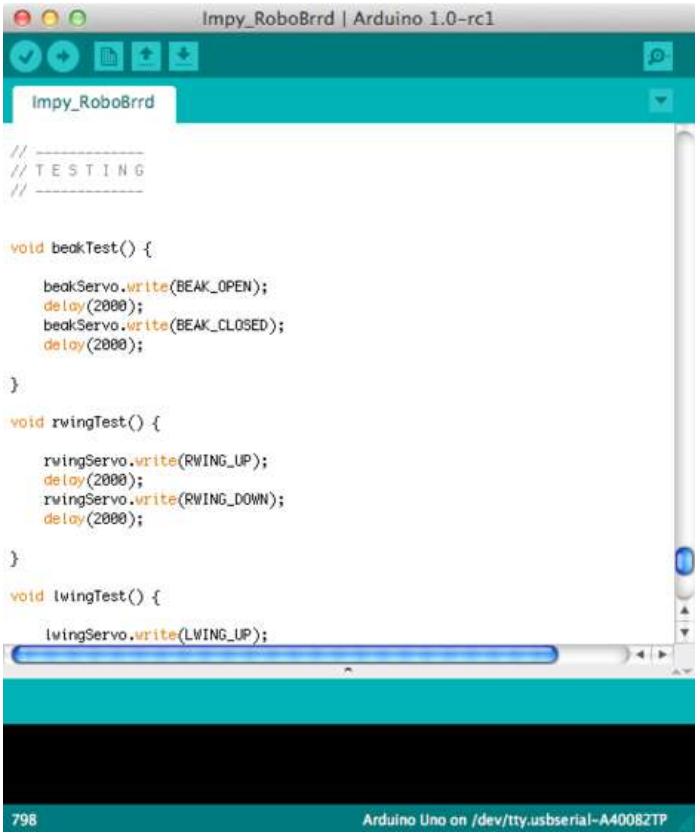

Impy_RoboBrrd | Arduino 1.0-rc1
Impy_RoboBrrd
}
void macarena() {
playTone(700, 20);
int sway = 5;
int rest = 500;
int angle = 60;
for(int i=0; i<12; i++) {
rotationServo.write(MIDDLE+angle+sway);
delay(rest);
rotationServo.write(MIDDLE+angle-sway);
delay(rest);
}
for(int j=0; j<10; j++) {
for(int i=0; i<2; i++) {
rwingServo.write(RWING_DOWN);
lwingServo.write(LWING_DOWN);
rotationServo.write(MIDDLE+angle);
185
Arduino Uno on /dev/tty.usbserial-A40082TP

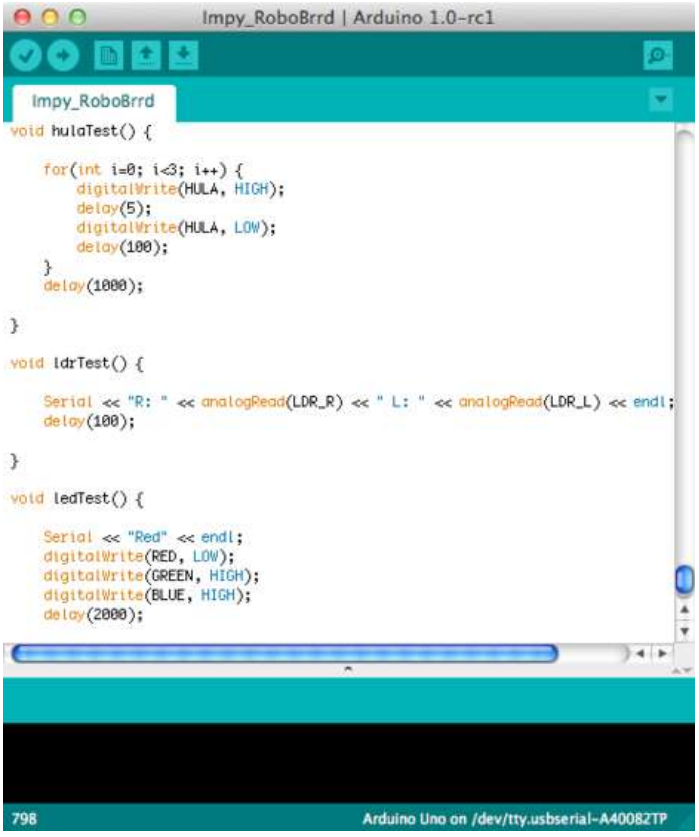

Impy_RoboBrrd | Arduino 1.0-rc1
Impy_RoboBrrd
lwingServo.write(LWING_MIDDLE);
}
// ---------------
// P A R T Y ! ! !
// ---------------
void partyBehaviour() {
playTone((int)random(20,175), (int)random(70, 150));
updateLights();
}
// -------
// L D R s
// -------
void chaseBehaviour(int ldrL, int ldrR) {
if(chase) {
493
Arduino Uno on /dev/tty.usbserial-A40082TP


Impy_RoboBrrd | Arduino 1.0-rc1
Impy_RoboBrrd
}
}
void peekABooBehaviour(int ldrL, int ldrR) {
if(ldrL <= (ldrLprev-50) || ldrR <= (ldrRprev-50)) {
// Close eyes
digitalWrite(RED, HIGH);
digitalWrite(GREEN, HIGH);
digitalWrite(BLUE, HIGH);
// Wiggle the wings
wingsExcited();
// Open Eyes
digitalWrite(RED, LOW);
digitalWrite(GREEN, LOW);
digitalWrite(BLUE, LOW);
// Play music
for(int i=0; i<3; i++) {
playTone((int)random(100,200), (int)random(50, 200));
delay(50);
}
}
551
Arduino Uno on /dev/tty.usbserial-A40082TP


Impy_RoboBrrd | Arduino 1.0-rc1
Impy_RoboBrrd
// ---------------------
// C A L I B R A T I O N
// ---------------------
void beakCalibration() {
if(Serial.available() > 0) {
c = Serial.read();
if(c == 's') {
Serial << "Stop position: " << pos << endl;
stopped = true;
}
if(c == 'g') {
forwards = !forwards;
stopped = false;
}
}
if(!stopped) {
if(forwards) {
pos += 1;
} else {
pos -= 1;
798
Arduino Uno on /dev/tty.usbserial-A40082TP


Impy_RoboBrrd | Arduino 1.0-rc1
Impy_RoboBrrd
stopped = true;
}
if(c == 'g') {
forwards = !forwards;
stopped = false;
}
}
if(!stopped) {
if(forwards) {
pos += 1;
} else {
pos -= 1;
}
beakServo.write(pos);
Serial << pos << endl;
delay(100);
if(pos == 30 || pos == 75) {
forwards = !forwards;
}
}
}
798
Arduino Uno on /dev/tty.usbserial-A40082TP


Impy_RoboBrrd | Arduino 1.0-rc1
Impy_RoboBrrd
// ------------
// TESTING
// ------------
void beakTest() {
beakServo.write(BEAK_OPEN);
delay(2000);
beakServo.write(BEAK_CLOSED);
delay(2000);
}
void rwingTest() {
rwingServo.write(RWING_UP);
delay(2000);
rwingServo.write(RWING_DOWN);
delay(2000);
}
void lwingTest() {
lwingServo.write(LWING_UP);
798
Arduino Uno on /dev/tty.usbserial-A40082TP


Impy_RoboBrrd | Arduino 1.0-rc1
Impy_RoboBrrd
void hulaTest() {
for(int i=0; i<3; i++) {
digitalWrite(HULA, HIGH);
delay(5);
digitalWrite(HULA, LOW);
delay(100);
}
delay(1000);
}
void ldrTest() {
Serial << "R: " << analogRead(LDR_R) << " L: " << analogRead(LDR_L) << endl;
delay(100);
}
void ledTest() {
Serial << "Red" << endl;
digitalWrite(RED, LOW);
digitalWrite(GREEN, HIGH);
digitalWrite(BLUE, HIGH);
delay(2000);
798
Arduino Uno on /dev/tty.usbserial-A40082TP

Step 18: Final Touches

Let's add some final touches to this RoboBrrd to make it look snazzy.

The most necessary addition are googely eyes over the RGB LEDs! To mount the eyes, place some hot glue in the middle of the back of the googely eye, and gently press it into the LED. You should see the back flex inwards a bit as you are pressing it. This is okay, but make sure it doesn't flex inwards too much otherwise the eye won't be able to jiggle. Ensure that the eye is as parallel as possible to the front face. Hold it until the glue has dried. It should be surprisingly sturdy!

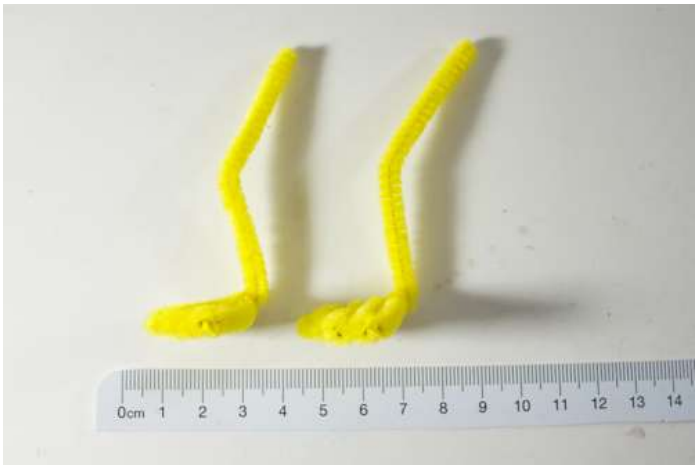
Next, add on the legs to the RoboBrrd. We use pipe cleaners for this. They attach to the bottom of the RoboBrrd.

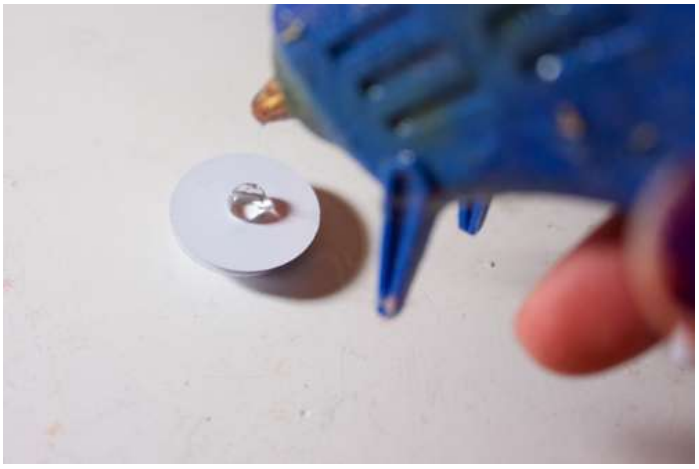
Now we can add some additions to the faces of the RoboBrrd, for example a lightning bolt and some stripes.

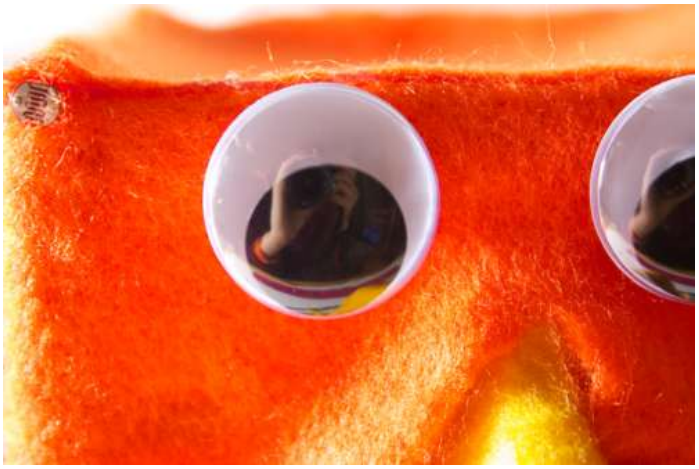
Be sure to paint the craft sticks on the DC motor for the hula hoop if not done already.

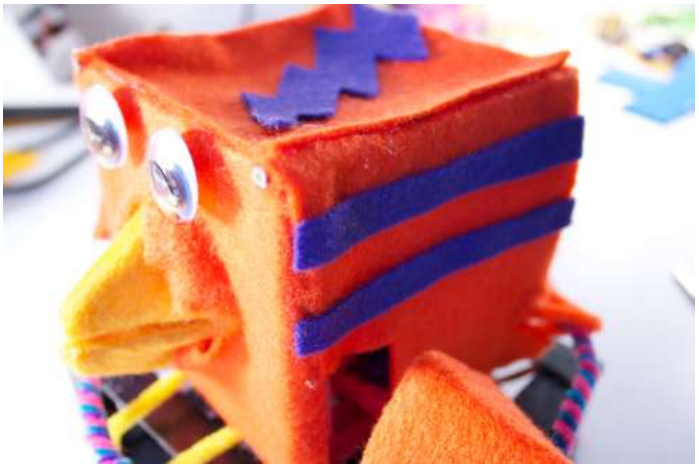
Cut a small opening in the back panel of the foam for the base. This will be for the USB cable for the Arduino.

If you want to add feathers to the RoboBrrd, go ahead! Plus, add on some other extra accessories. Bells on the wings would be an interesting musical touch.









Step 19: Above & Beyond

Congrats! You are now completed the RoboBrrd construction. Now it is time to take it to the next level. What will you do with your RoboBrrd? In the 2nd picture is a drawing of a RoboBrrd by a fabulous young robot/artist named Alec! Way to go!

The possibilities for RoboBrrd are endless.

You can interface RoboBrrd with the 'Math in Space' Processing program that was created for a previous RoboBrrd! You can grab the code on GitHub here . Try modifying it to make it work with your RoboBrrd. (Hint: instead of using a distance sensor, try using the LDRs!)

If you were working with someone else who was also creating a RoboBrrd, you can have them interact with each other if you add on an XBee to them! Create a mini mesh network and share messages from the RoboBrrds in between them.

Other ideas are to make the RoboBrrd sing, send out virtual tweets, and add more sensors to it! Adding something like a pulse sensor would be neat. The RoboBrrd can alert you if you have no pulse (wait, that doesn't make much sense!). :)

Go wild with it. RoboBrrd is an awesome robot to explore different movements and behaviours with!

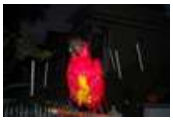
Huge thanks to Adafruit Industries for the servos, electronics, inspiration and celebration of robots, Arduinos, and OSHW.

Huge thanks to all of the fans of RoboBrrd out there, and the people who inspire me to make funky robots.

Thanks to David Cuartielles for the great photos of RoboBrrd at Maker Faire NYC 2011 (used in Step 1).

If you have built a RoboBrrd, leave a comment with your feedback, and create your own Instructable documenting your process!

Related Instructables



Animatronic Fawkes the Phoenix by makermike

Arduino Wireless Animatronic Hand by njkl44

Robotic Talking Turret by RazorConcepts

Arduino animatronics- make your awesome costumes more awesome! by Honus

Animatronics: Lion Dance (video) by tonll

SPROT, my SimPleROboT by sbateson

Comments

1 comments **Add Comment**

mikeasaurus says:
Such great creativity and accessibility, awesome!

Nov 15, 2011. 7:23 AM **REPLY**