K-9 2.0. WiFi Controlled and Autonomous Robot Dog.

by **steve-gibbs5** on May 30, 2014

Table of Contents

Intro: K-9 2.0. WiFi Controlled and Autonomous Robot Dog.

Features

- Sturdy wooden (MDF) and acrylic design
- Fully microprocessor controlled using the EZ-Robot EZ-B v4 controller and EZ-Builder software platform
- Full drive movement, forward, reverse, turn left, turn right
- Twin 30w 12v drive motors with PVC wheels and rubber tyres
- Heavy duty servo and bearing steering system

- Multiple control options including...
  - Fully WiFi controlled Via laptop, tablet PC or smartphone
  - 3G/4G LTE controlled via Smartphone
  - Voice, facial, colour, glyph recognition controlled
  - Rotating ultrasonic distance/ avoidance sensor to help control autonomous movement

- Speech synthesizer that allows verbal communication
- Fully customized chatbot program to allow dynamic conversation between K-9 and humans
- Multi-movement head, looks up, down, left and right
- Autonomous movement abilities and personality generator
- Retractable LED torch/flashlight from K-9's nose (his laser weapon in the TV shows)
- Multi-coloured sound reactive EL lighting panel for his mouth
- Independent rotating ears (both move in same or opposite directions from one another)
- Multi-coloured RGB LED eye panel, (i2c connection) (eyes can blink, show emotion and patterns)
- Front facing camera for no latency video streaming, vision tracking and still photo capture
- Side facing camera for video and still photo capture
- Removable 7" Android tablet PC for K-9s VDU (visual display unit) on his side panel
- Servo controlled blue under chassis LED lighting with 8 flashing modes
- Blue edge lit EL lighting for dorsal panel and collar
- Multiple drive speeds via H-Bridge and PWM control
- Security remote power On/Off to K-9's power systems (so no unauthorized operation)
- Wagging telescopic tail

- Ability to navigate with room mapping
  - Removable dorsal control panel with...
  - Servo controlled illuminated LED buttons with 6 flashing modes
  - LED battery meter to monitor main drive battery
  - Programmable dot matrix scrolling message display that can be removed

- Removable magnetic inspection panel
- Ability to inform of incoming phone calls and emails, and who is calling or sending them
- Built so body can be removed from rolling chassis for easy maintenance
- Media player which can be controlled using voice recognition or WiFi control
- Verbally tells people the latest news/weather/traffic reports and more using RSS feeds
- Can entertain by playing games, telling jokes, telling fun facts, and trying to sing songs

- And he can do the whole "Line following" thing too.

K-9 2.0. He really is a man's, or woman's best friend.

Introduction

I've been a massive fan of the loveable metal pooch from Doctor Who ever since I was I kid and he is my one of my all time favourite companions. I have always wanted my own K-9 unit and recently found some spare time in my hands so finally had the chance to build my own.

When I made my Cardboard K-9 a while back as a proof of concept, I mentioned in the related Instructable that I was planning making a full size robot version with a load of robotic bells and whistles added, and if I did so, I promised that I would write an Instructable to go with the build if I ever got around to it. Well after a little research, a lot if head scratching, then finally drawing up some plans, I'm pleased to say that I finally did get around to making another K-9 unit which I am calling "K-9, 2.0", and true to my word here is the Instructable which includes all the plans I drew up for the build, photos during the build, and instructions of how it all went together, which I hope will be easy enough to understand and will hopefully go towards helping anyone who plans to build their own version. He took me about 4 months to complete including programming, working a couple of hours in the evening every day.

My take on this build looks a little different to the original design, and at the risk of upsetting any K-9 purists, let me explain. I took the slightly controversial decision to give my K-9 a little bit of a facelift in regards to measurements and the overall look to give him a slightly more modern look and something that was a little bit different to the norm, but still very much keeping in the spirit of the original design, so I hope any die hard K-9 fans reading this will see why I made the changes, and hope you like it. I would love to hear your thoughts and opinions.

So with that said, let's move on to the build itself. The body is mostly made up from 9mm MDF board, used 2x1 Wooden batons for supports and fixings, 3mm transparent blue acrylic sheet for the snout, dorsal and lower body panels. I used the transparent type as it was a lot cheaper than purchasing opaque acrylic or Perspex sheet, but I had a couple of ideas of what I could do. I also used a peace of 2mm white plastic sheet cut and molded for the top and rear of the head. When it came to colour, I had already decided to go with deep blue panels but I just couldn't decide what shade of grey to do the rest of the body as it would be a one shot deal without wasting money on respraying it. I eventually settled with metallic silver as it would be a clean modern look and thought it would off set the blue panels quite well.

The rolling chassis is again made from MDF and 2x1 batons, and include two 30w 12v drive motors, one heavy duty servo and Lazy Susan bearing, two 10 inch tough plastic wheels which came from a 12volt kiddies ride on sports car, custom rubber tyres, battery meter, on/off bypass switch, speed controls, a 12v rechargeable battery or 11.1v LiPo battery . I was going to go down the power wheelchair route, but I looked at these ride on cars motors as they were a lot cheaper and have enough torque. The front wheel came from an old pushchair.

The robotics side of things came courtesy of EZ-Robot which included servos and fixings, wiring, RGB LED eye panel, ultrasonic sensor, cameras, and the brains of the outfit, the EZ-B v4 controller kit which is powered with a 7.4v LiPo battery, and the EZ-Builder programming software. It can all be controlled by PC, laptop, tablet, smartphone, joystick, voice recognition, glyph recognition, and more ways that are being added all the time. As you can see, there are many options to physically control K-9, but with a little bit of programming, he also has the ability to be autonomous. K-9 would have turned out very different, and probably not as good as he is now if it wasn't for the guys at EZ-Robot and their community. Best robot shop I have ever found.

These are the other bits and peace's, either bits I purchased from online stores, or items I had saved up for this build. They are...

- Telescopic antenna for his tail.
- LED flashing Christmas lights.
- Sound activated EL lighting wires for his mouth.
- Mountain bike tires attached the the ride on car wheels to prevent wheel spin.
- Illuminated push switches of various colours for the control panel.
- Steering rack boot for the base of his tail.
- Air duct tubing for his neck.
- Speaker grill and two old rubber antennas for his ears
- A super bright pen sized torch/flashlight for his nose laser (keeping it child friendly, and it's more useful than a laser pointer which was my original idea)
- Rubber hose insulator for his bumpers.
- A nice looking draw handled (close to the original look)
- A Bluetooth speaker to hear K-9 talk.
- And a sheet of blue acrylic.

And speaking of hearing K-9 talk, as with my cardboard version, I have set up and extensively trained a Chatbot from Pandorabots.com. This allows somebody to have a two way dynamic conversation with K-9, just like they were talking to another person. K-9 uses a SAPI 5 compliant speech synthesis voice which was purchased from Cepstral voice solutions. As you can hear in the videos, he has a British accent with a slight robotic voice effect. He speaks very clearly and is easy to understand. He is programmed with fun facts, jokes, can sing a song (sort of), and engage with what ever subject the user is talking about. By using the internet he can also tell people the latest local or international news or weather reports, tell you the latest traffic reports before you go to work, or keep you informed of the latest sports results. He can be spoken to, using the speech recognition control built in to EZ-Builder using a headset microphone, but I use an iPhone text to speech engine as the recognition is highly accurate and second to none. this is connected to the software via a VCN application that connects the iPhone to the computer running the EZ-Builder project. The voice recognition also controls some of his functions such as activating his torch, turning lights on and off and more. Watch the videos for a demonstration.

I have worked with wood in the past, but I am by no means an experienced wood worker so I probably could have done certain things a better way, but that being said, I am quite pleased with the way it turned out and it surly was a great learning curve, and was fun to build. So if you do decide to build your own K-9 unit and you have a better way of doing things, then by all means feel free to change things around, and I would be interested to hear what you would have done differently. So before we start I just want to mention that the measurements I have used for this build are not the true measurements for the original prop used in the TV shows, but ones I have come up with myself by studying photos and still video shots, and adapted them to accommodate the components used for my build. For some more accurate great prop measurements and plans check out the incredible K-9 build by Podpadstudios. Their build was part of the reason for getting me kick started on my K-9 project, and it is such a cool build too.

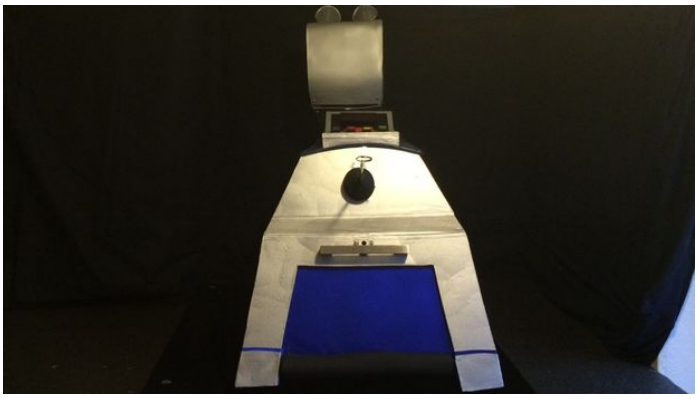
There may look like a lot of steps in this Instructable, but they mostly contain pictures of the build with brief descriptions of how things went together. I chose not to write a lot of detailed steps of the build as a picture speak a thousand words, and I have many of them to share with you.

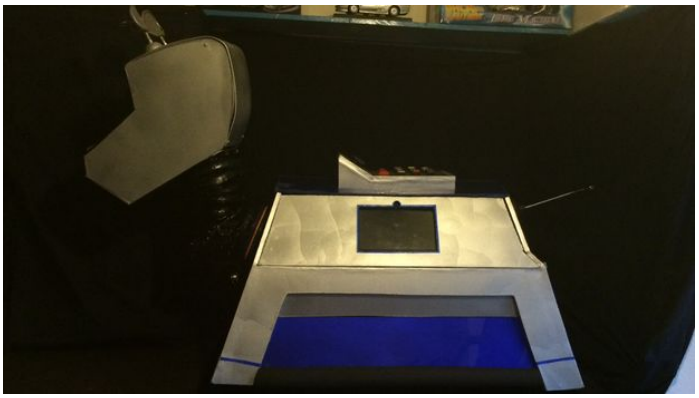
Video slideshow of K-9's build process.

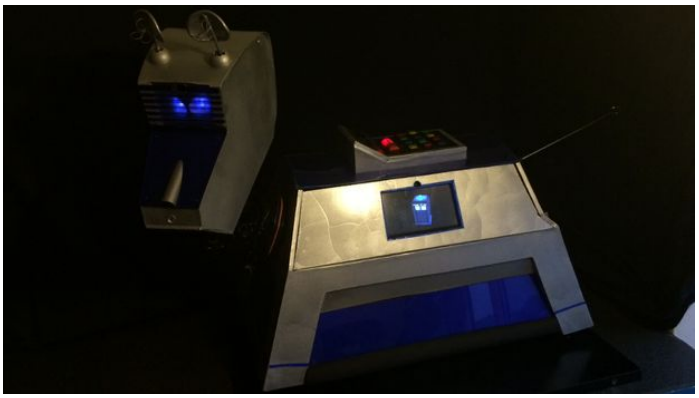
Video demonstration of K-9 in action.

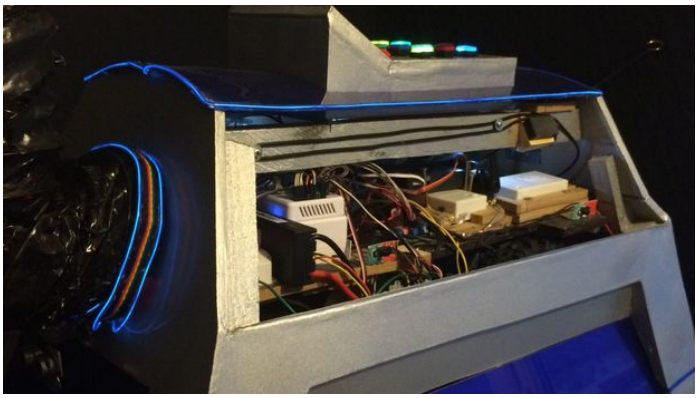
Video demonstration showing K-9's mobile control.

So let's get to it.

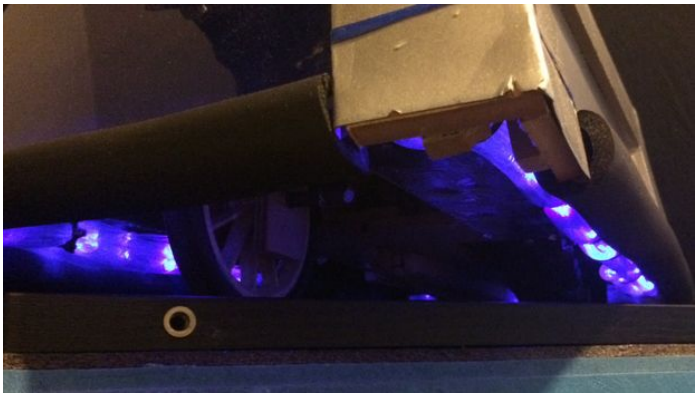






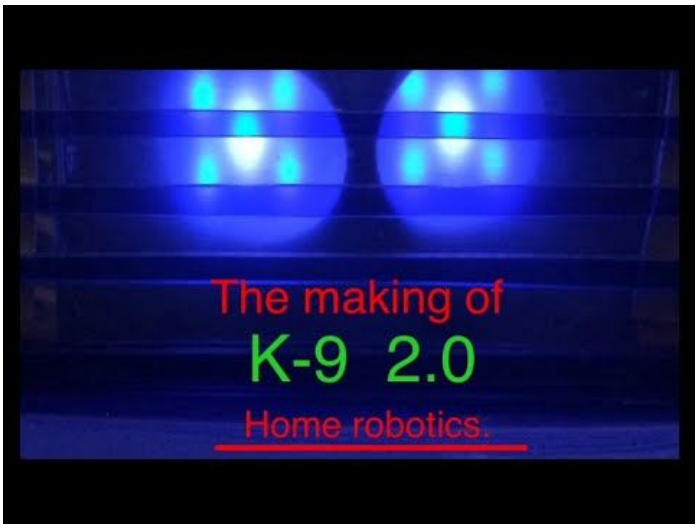



ezrobot
Explore
Shop
Software




K 9


K9


The making of
K-9 2.0
Home robotics.

Step 1: Tools and Equipment.

A brief list of the tools I used.

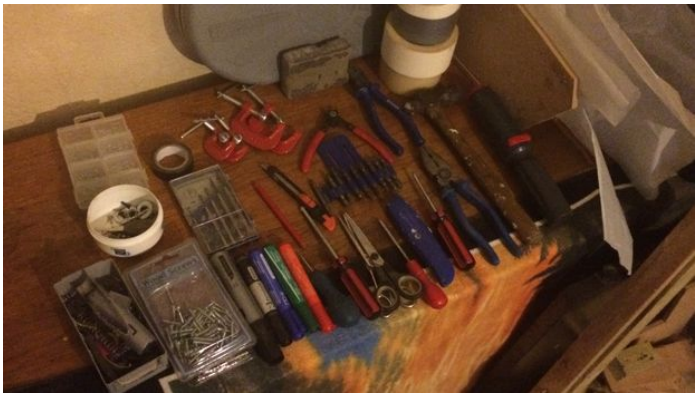
- Hammer
- pliers
- Phillips screwdriver
- Stanley knife
- Small tipped slotted screwdriver
- Pair of scissors
- Centre punch
- Red, blue and black permanent markers
- Wax pencil
- Craft knife
- Power screwdriver
- Various power driver bits
- Wire cutter/strippers
- Wire snippers
- Metal clamps
- Various size Phillips screws
- Various size nuts and bolts
- Electrical tape
- Hack saw with wooden and metal cutting blades
- Fine grain sand paper

Power tools I used were

- Cordless drill
- Jigsaw with wood blade and fine tooth blade for cutting acrylic
- Heat gun/paint stripper
- Power sander
- Nail gun
- Laptop for build plans and robotic programming

And the adhesives I used were

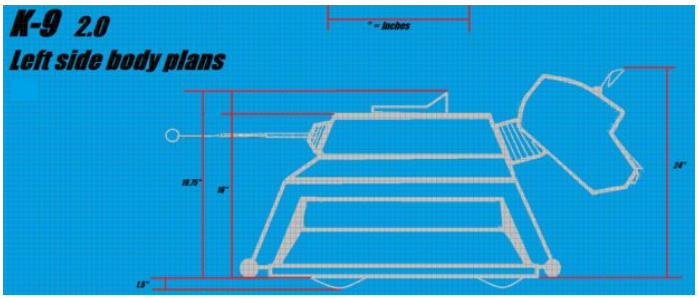
- Heavy duty multi purpose "Liquid Nails"
- Grab adhesive for filling gaps
- Super glue
- Low tack masking tape
- Aluminium tape
- Bag of 100, 30cmZip ties

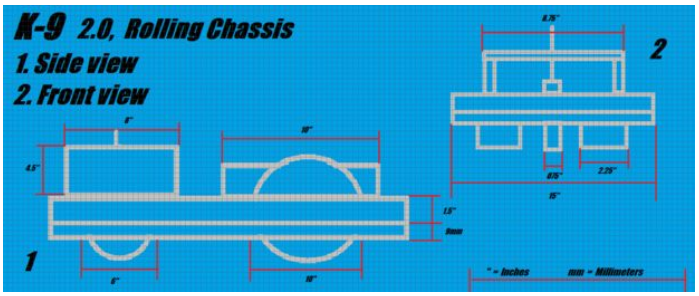


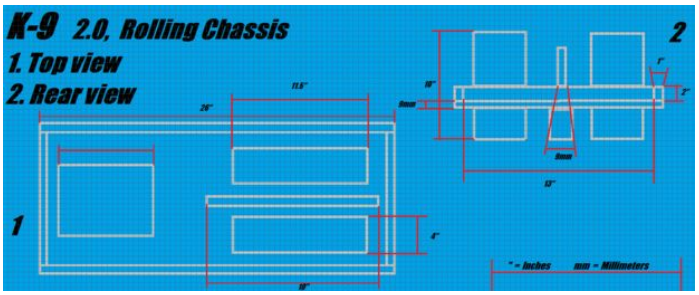
Step 2: Blueprint Plans for K-9, 2.0.

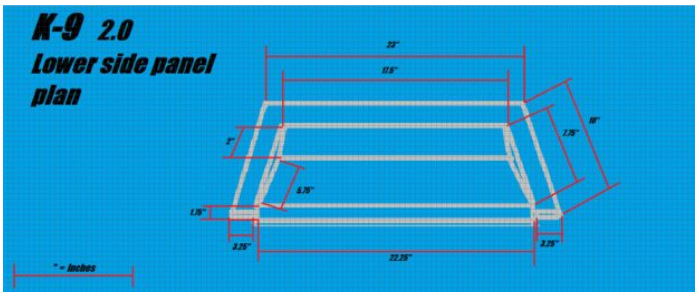
The blueprints I have supplied show all the main measurements I used for
my build. Anyone who would like to use these blueprints, along with the build photos, I hope you find them useful and easy to understand. As I mentioned previously, the measurements in these plans are for the build featured in this Instructable and not for the exact size for the original TV show prop, although there is not much different between the two. For something much closer to the original prop plans, have a look at Podpadstudios Instructable, or visit this page for "Lets make Robots" website which also has some plans. I have also added the relevant blueprints to each of the steps that are applicable for this build.

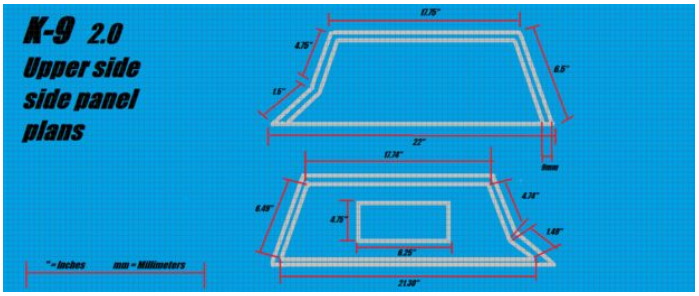
So lets start off with the drive section. That's up next.

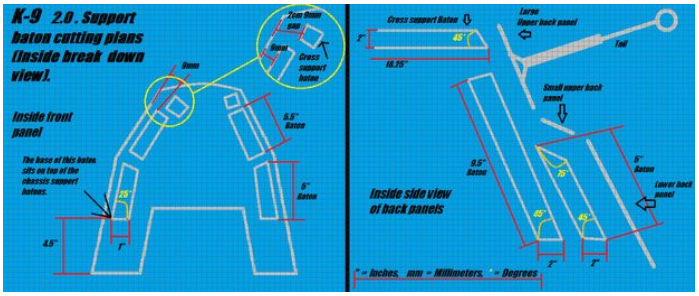


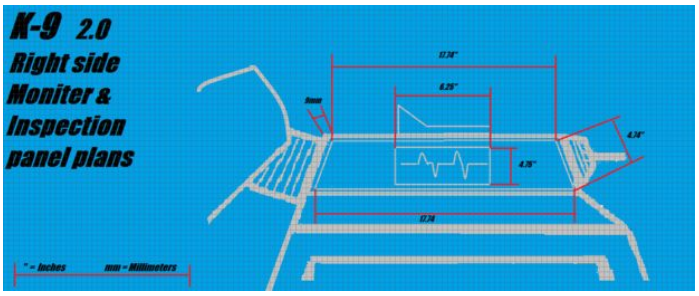


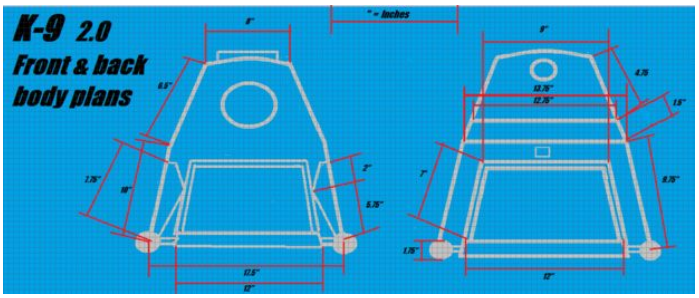


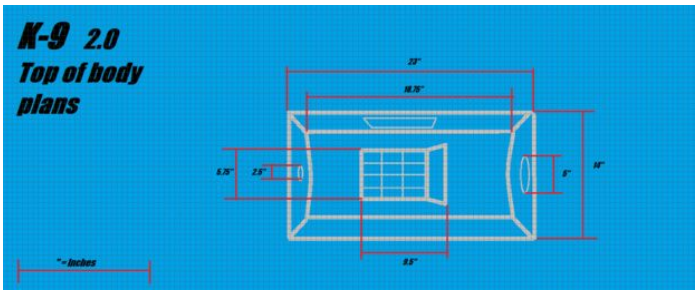


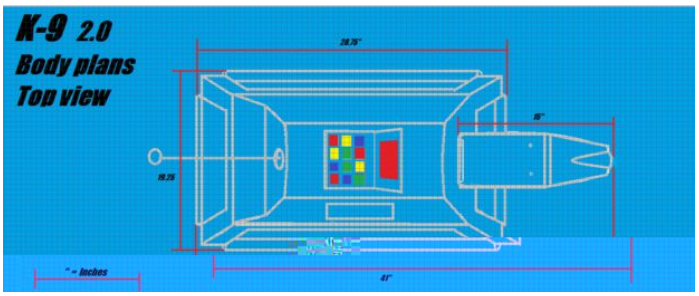


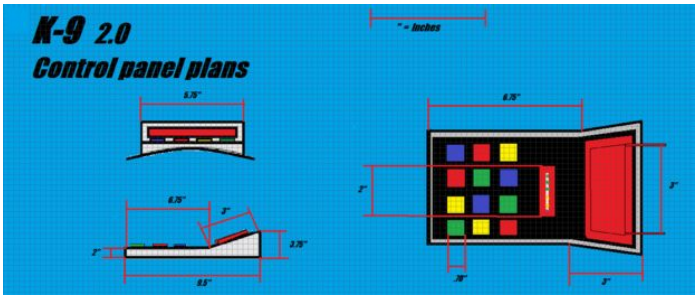








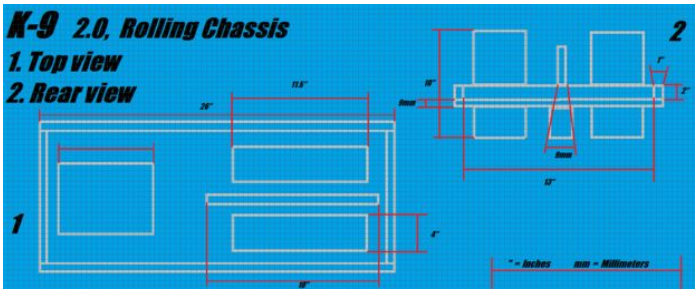


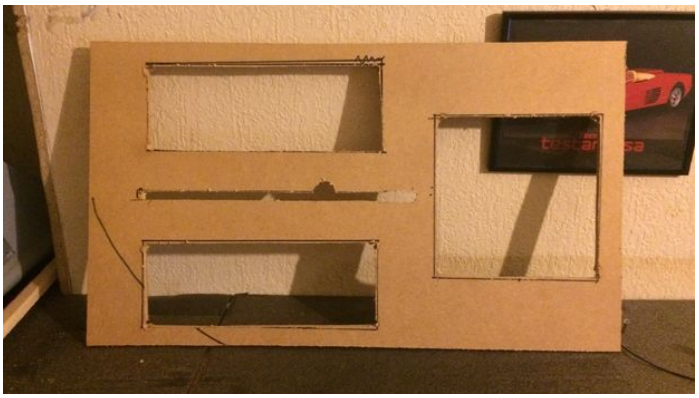


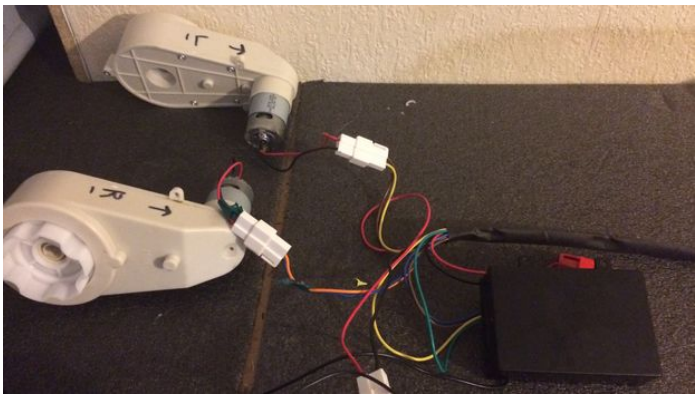
Step 3: Making the Chassis.

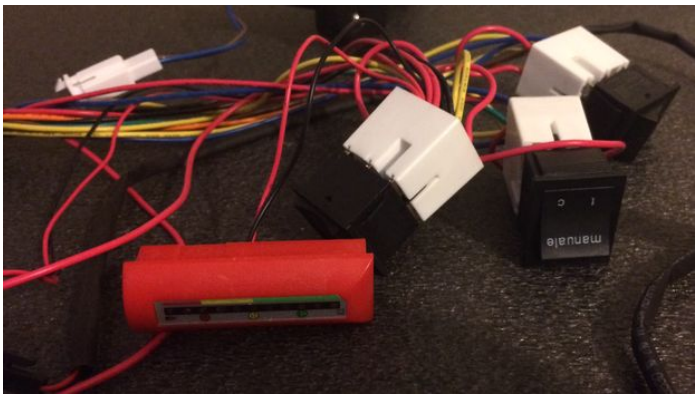
From this point on, you may want to refer to the blueprints and
accompanying build photos for clarification of measurements and hard to explain areas.

So, starting from the ground up, I'll begin with the drive system. I measured and cut a sheet of MDF, marked up where the drive wheels and front steering wheel were to sit and for the drivetrain board, and cut out with a Jigsaw. Then using a smaller peace of MDF, I cut it to size, drilled a hole for the motors to sit and another hole for the axle. I then screwed the motors to the board and slotted the drivetrain in to the chassis and secured, followed by inserting the axle and attaching the wheels. Then some small peace's of wood were cut and attached to make the steering section. The wheels were plastic so there was not much traction. To over come this, I used an old mountain bike tyre, cut it to size and trimmed off the tyre walls, an attached it to the wheel using some nuts and bolts. This would make them easy to remove if they needed replacing, and the head of the bolts sit on the base of the tyre between the treads so they don't rub on the ground, only the tyre tread would.

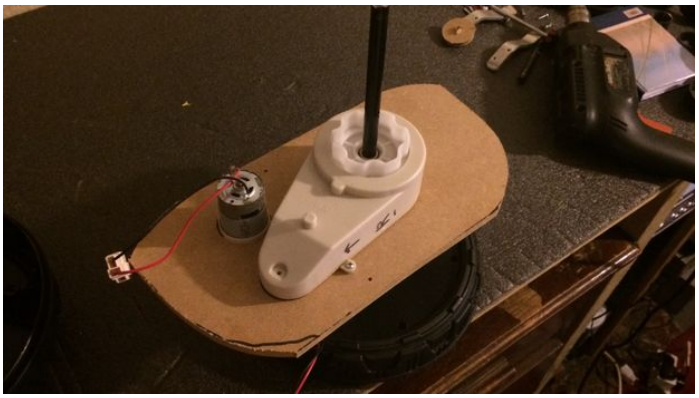




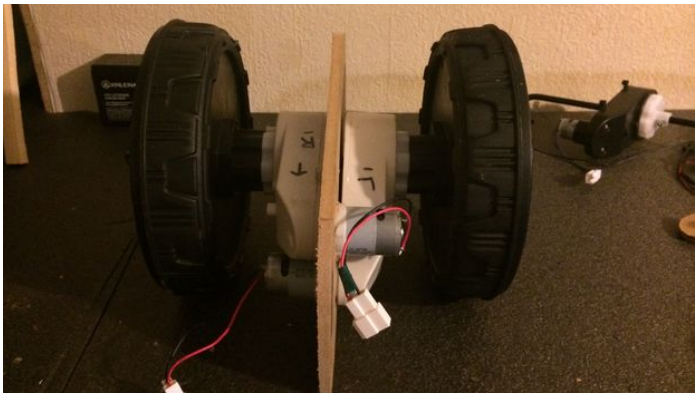


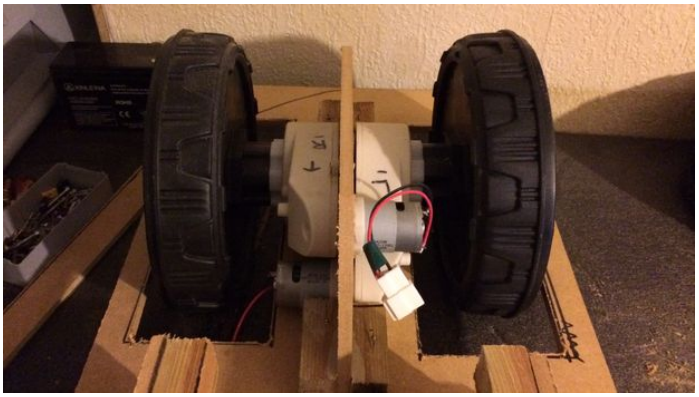

manuale

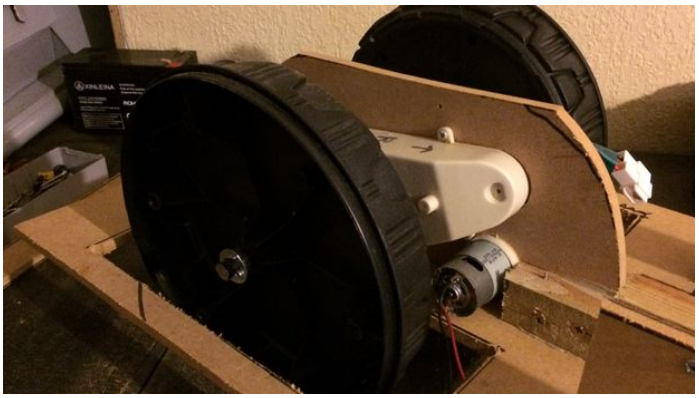

XINLEINA
ROHS
Pb
Akai

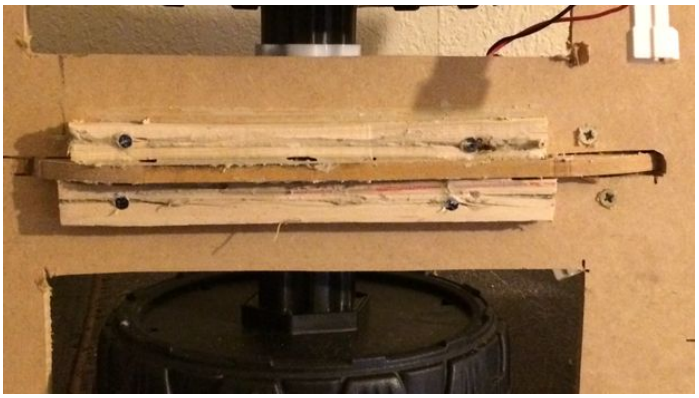


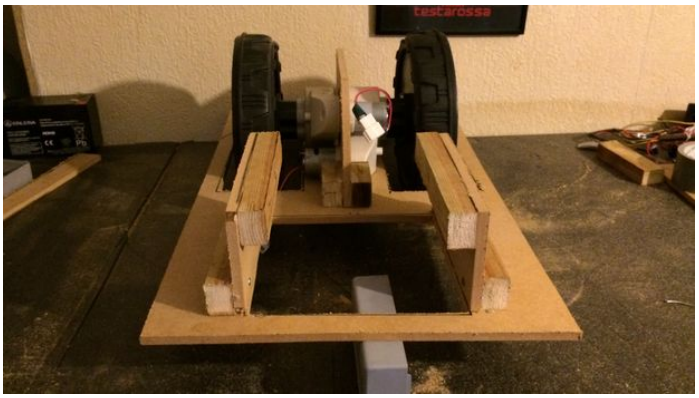

testarossa

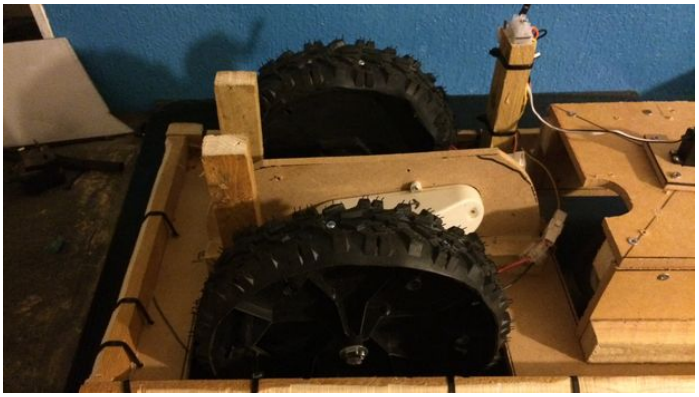








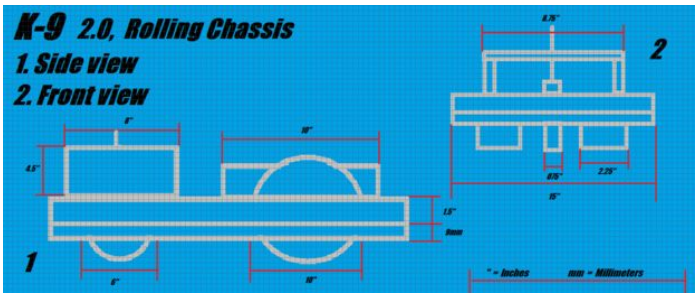


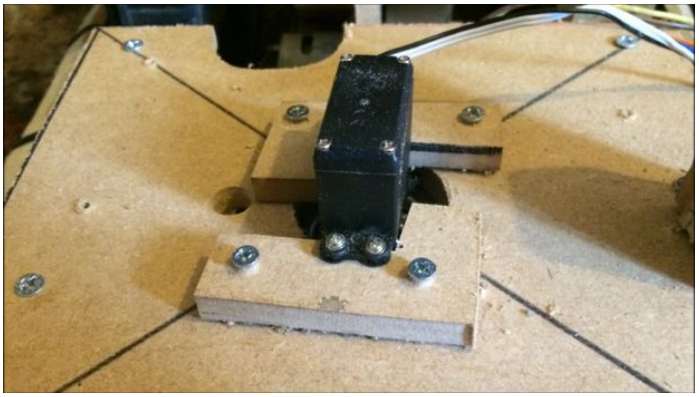


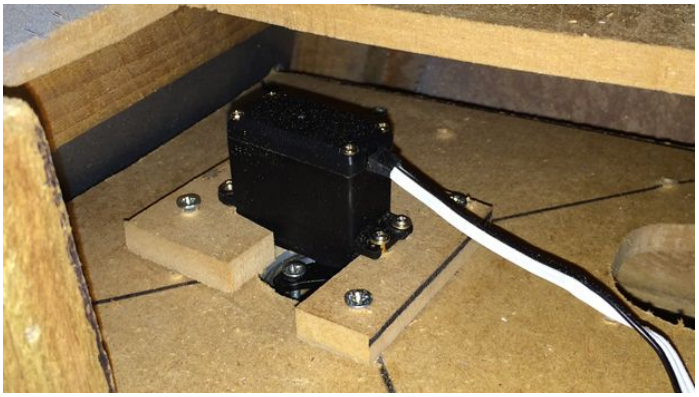
Step 4: Chassis Steering.

Stage 2

I had to do some modification to the steering system design as I originally was going to use a third 12v motor. I decided against this and opted for a Heavy duty servo and lazy Susan bearing. This would give great and more accurate steering control. I cut a peace of MDF to make a steering platform, measured the centre and drilled a 2" hole for a servo horn to fit through. After fixing the steering platform down I cut and fixed three peace's of MDF to create a bracket, drilled a hole through each side and thread the axle through, along with the wheel and secured it with split pins. Then I screwed down the bearing on to the top of the bracket. I centred the servo to 90 degrees, screwed the servo down on to the top of the platform with the servo horn going in to the bored hole. Finally I screwed the top side of the bearing to the under side of the platform, centered the wheel and screwed the servo horn on to the top of the steering bracket.

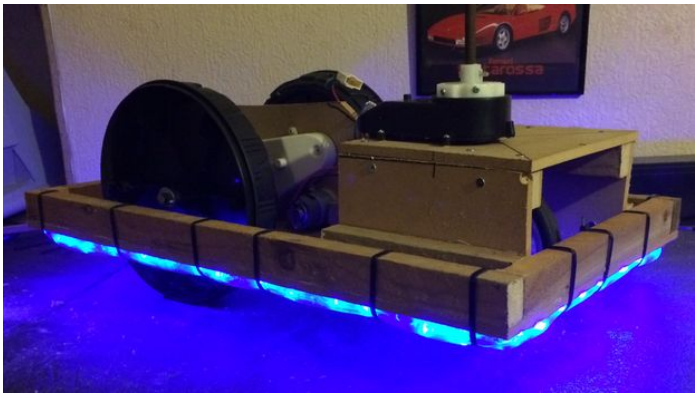






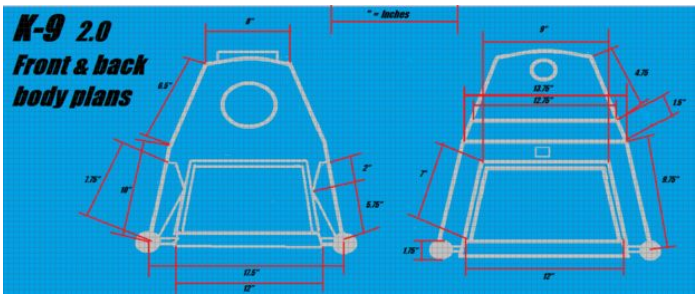
Step 5: Under Chassis Lighting.

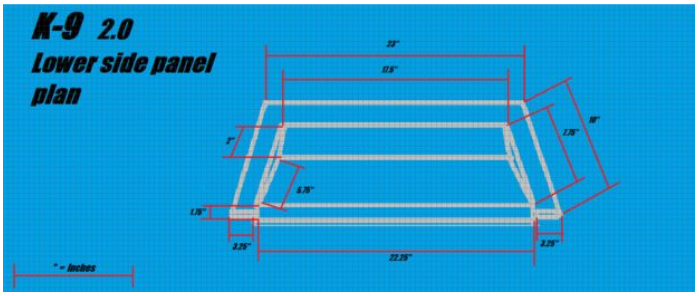
A cheap but very effective way of getting some bright and cool looking lighting under the chassis was to use a chain of 40 battery powered blue LED fairy lights which has eight flashing modes, a length of clear tubing, some zip ties, and some aluminium tape.I measured two lengths of tubing so they would fit around the each side of the chassis, cut them to size and cut down the length of them with a knife. I fed the LED lights in to one length of the tube and bunched up the bulbs evenly so the were close together. Once done I fitted the second length of tubing over the first to cover the knife cut I made.I then covered the entire base of the chassis with aluminium tape to act as a reflector. Then I drilled some holes evenly around the edge of the chassis base. Finally I attached the lighting tube to the chassis using zip ties that were fed through the drilled holes. The LED's battery compartment would be placed in to the body later on.

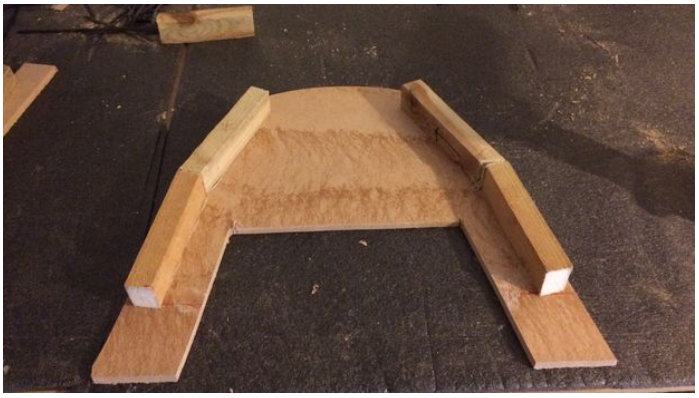


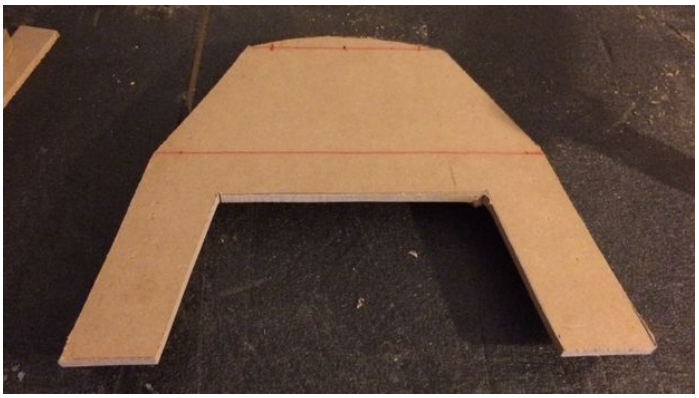
Step 6: Body Frame.

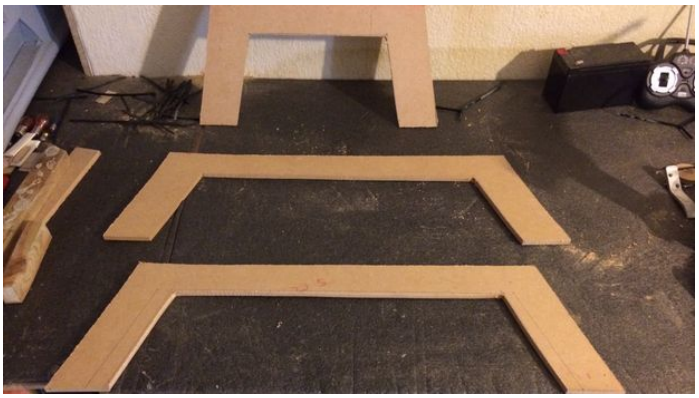
To start the body I first made the frame by measuring and cutting some 1x1" batons. Then they were screwed together to form the overall body shape. Four further, short, batons were cut and fixed inside of the four corners so they would rest on the chassis. The frame was then rested on to the chassis to insure a good fit, sat evenly, and checked ground clearance for the body panels later on. Adjustments were made where necessary.

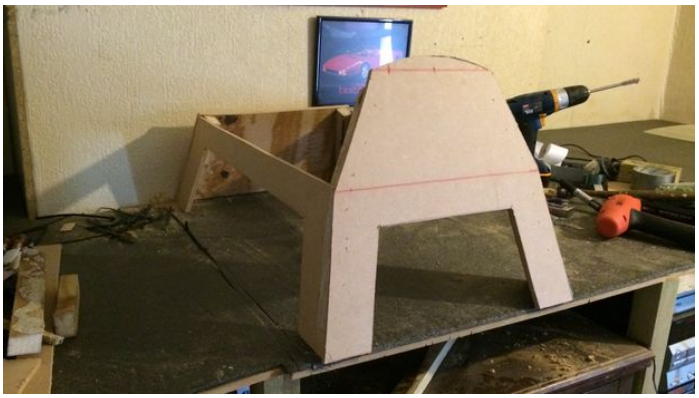






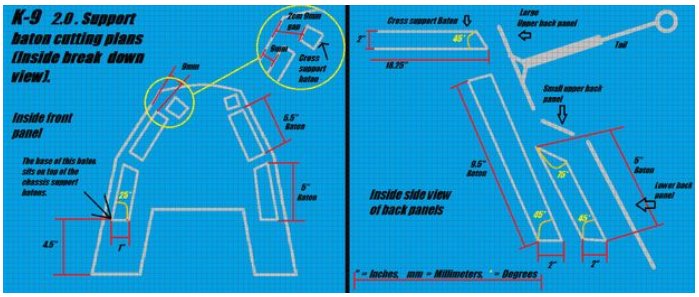


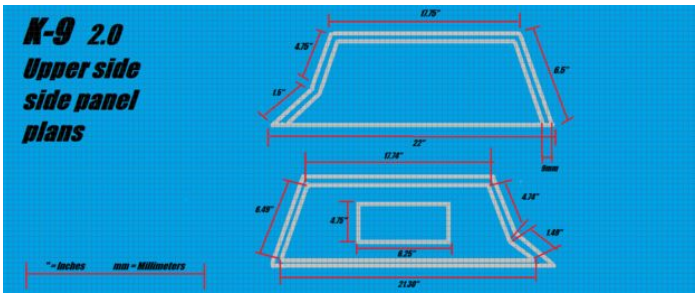


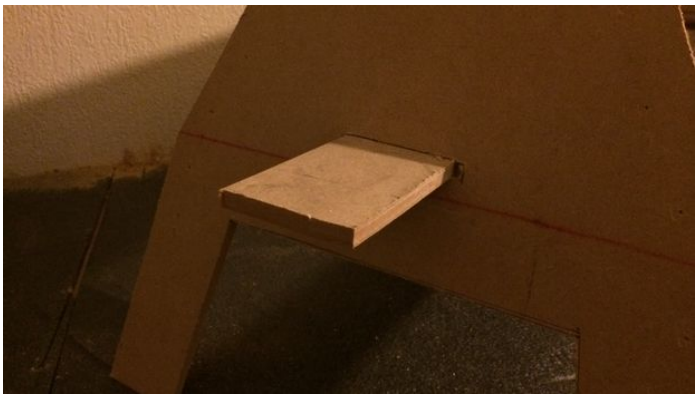


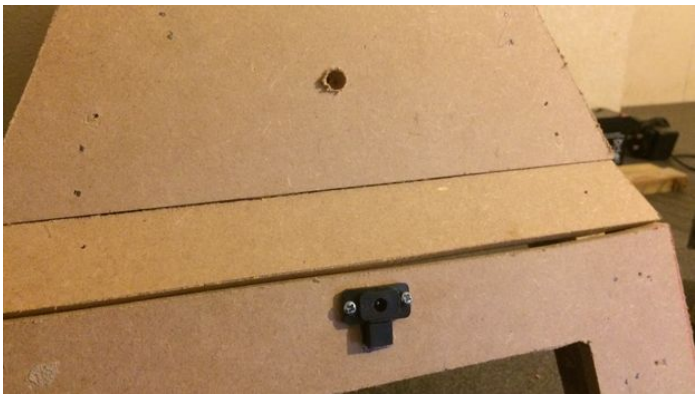
Step 7: Side Body Panels.

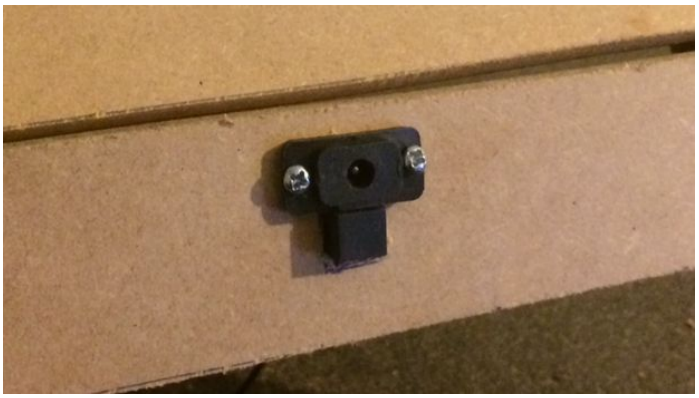
The MDF body panels were now measured and cut to size. These were then fitted front first, then the rear panels, then the lower side panels. The difference with the side panels were that on the left side, the upper panel was fitted too. The right side was left off as this would be the removable inspection panel. All of the edges and gaps were filled and sanded ready for spray painting. Three coats of grey primer were applied, followed by five coats of metallic silver paint and three coats of clear lacquer. A charging port was then added to the rear panel. This is where a lead acid battery could be charged if needed.

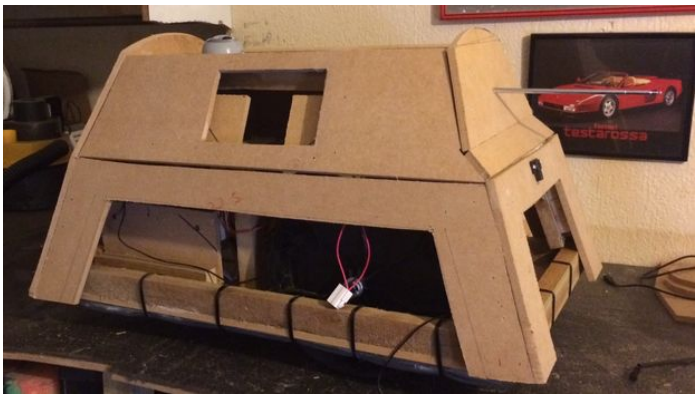


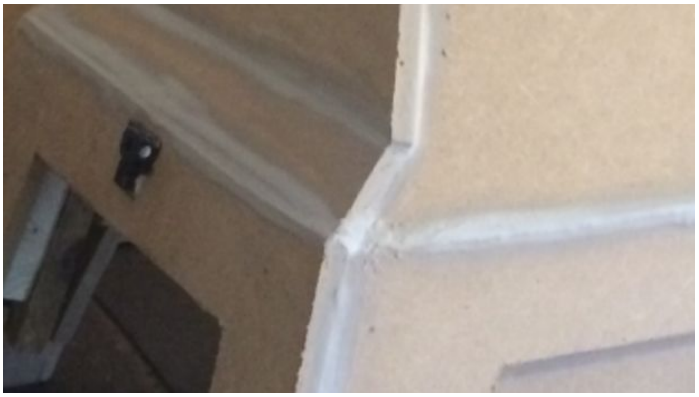

K-9 2.0
Upper side
side panel
plans
" = Inches
mm = Millimeters

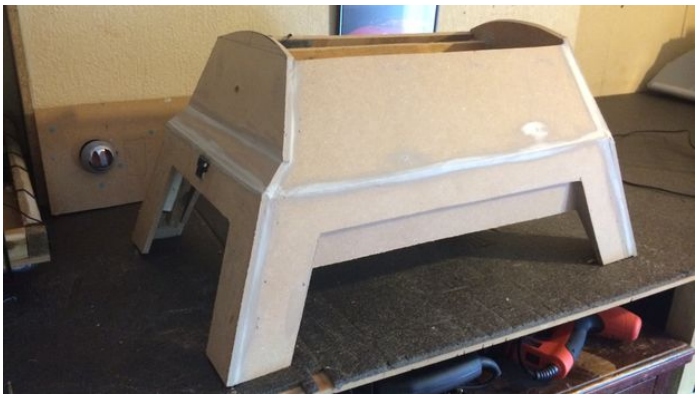


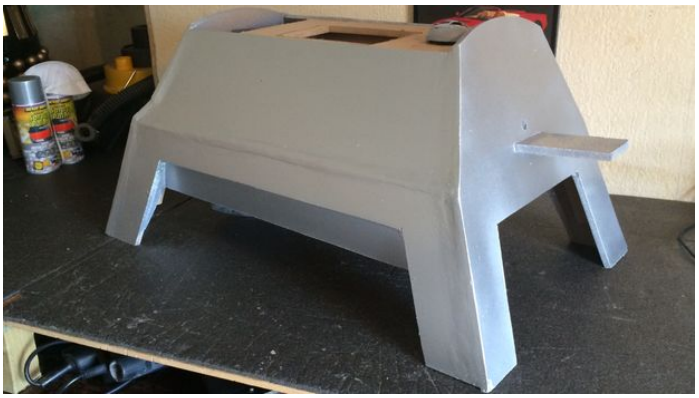





testarossa







Step 8: Inspection Panel and VDU Housing.

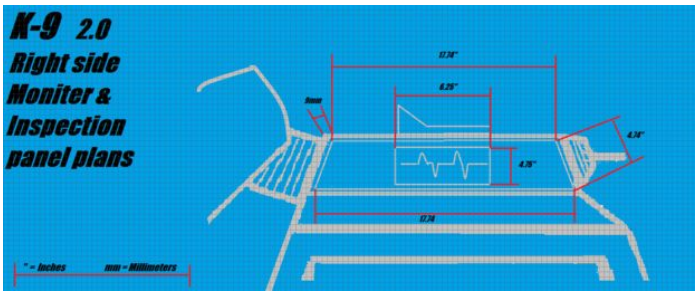
A little work needs to be done on the inspection panel, mainly to house the tablet PC and fix a magnetic fastener to have it fix to the body.

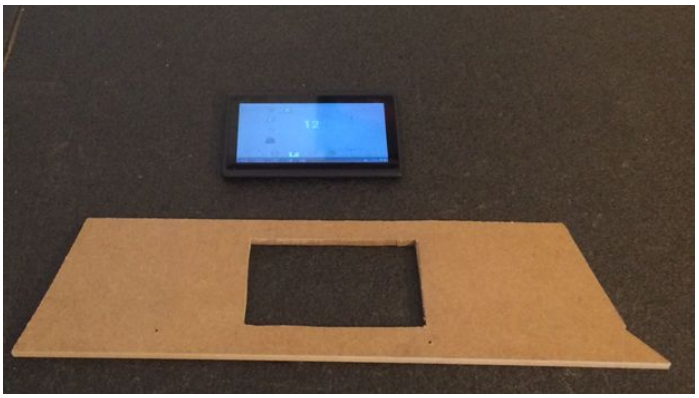
1. Sand the left and right side edges of inspection panel down a millimetre or 2 to insure the panel can be removed from the main body, but doesn't leave to much of a gap when it's in place.
2. Measure the size of the tablet PC screen dimensions, mark on the inspection panel and cut out.
3. Lay the inspection panel on the work surface facing down.
4. Place the tablet PC on to the back of the panel over the cut out, insure any part of the screen is not obscured by any of the panel by carefully turning it over to check.
5. Then carefully place panel back on to the work surface keeping the tablet still and in place, then draw a line around the tablet and remove.
6. Using some MDF cut offs, measure two peaces , and one peace ,
7. Cut some cardboard strips to the same sizes and superglue to the MDF strips.
8. Apply adhesive to the 3 strips and place on to panel around the outside of the line marking, and remove any access adhesive.
9. Place the tablet back in to position to make sure the MDF strips are straight and the tablet is not to lose or to tight of a fit (leave a couple of millimetres gap for easy tablet removal, but also so it has minimal room to slide about). Leave for adhesive to set.
10. Cut 2 MDF squares measuring 7cm square.
11. Place the 2 squares on to each side of the 3 strips and check to see if the tablet slides in and out easily. Add some more strips of cardboard to pad them out if necessary.
12. Apply adhesive to the 3 strips, attach the 2 square peaces and leave to set.
13. Mark exactly where the tabled front facing camera is on to the edge of the inspection panel.
14. Using a "saw" drill bit, carve out a semi circle through the top edge of the VDU housing frame of the inspection panel.
15. Place a scrap peace of wood, the same size as the tablet, in to the VDU housing.
16. Give the panel 2 to 3 coats of primer, 5 to 6 coats of metallic silver paint, and 3 to 4 coats of clear lacquer and leave for at least 24 hours to cure.
17. For the blue edge piping that goes around the VDU housing, I cut some 9mm strips from a blue plastic file folder, applied glue around the edges of the housing then attached the strips.
18. For the outer bezel, I masked around the housing to leave a half centimetre edge showing then simply went around with a blue permanent marker a couple of times.
19. With a small thin plate of steel (measuring 1x2 inches), apply adhesive to one side and attach it to the top edge of the inside of the inspection panel just beside the VDU housing.
20. Sit the inspection panel on to the body and mark the location of the steel plate on to the cross support baton.
21. Remove panel, apply adhesive to the back of a magnet and attach to the support baton between the markings, clamp down and allow adhesive to fully set.

And that takes care of the inspection and visual display unit panel. You can put the panel aside for now as it won't be needed until the end of the build. At this point you can have some fun making some animated graphics and/or find some nice animated screen savers and video clips to download to the tablet to have playing when K-9 is fully activated.

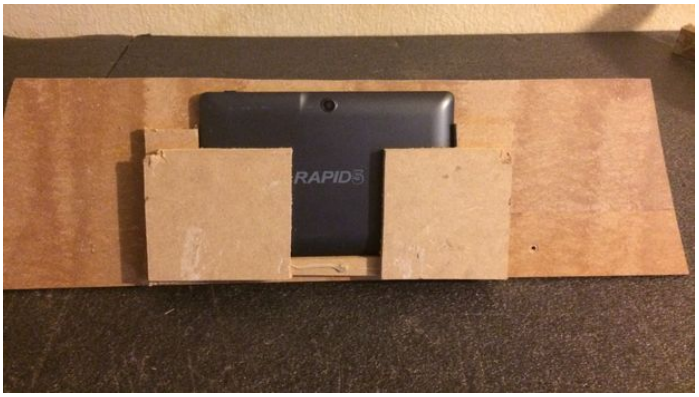
In the next stage we will be looking at a few of the smaller details, mainly the ears, nose cone and tail. Small parts but important never the less.


Tablet PC

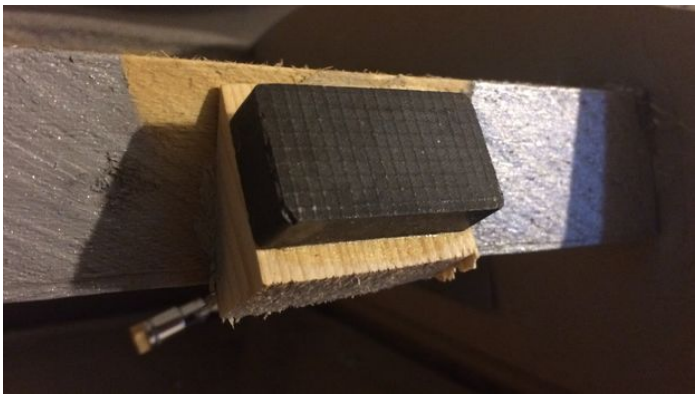

K-9 2.0
Right side
Moniter &
Inspection
panel plans
" = Inches
mm = Millimeters

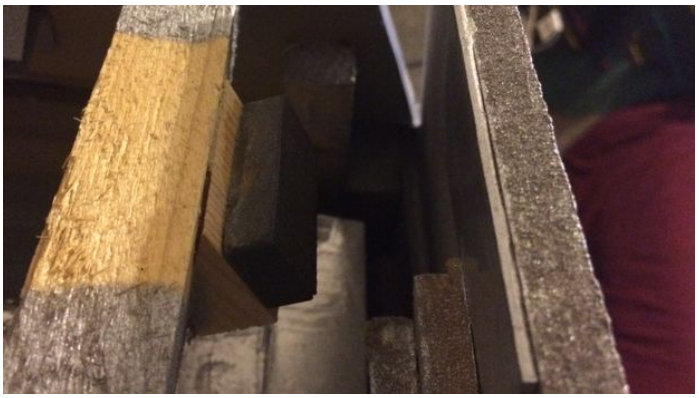


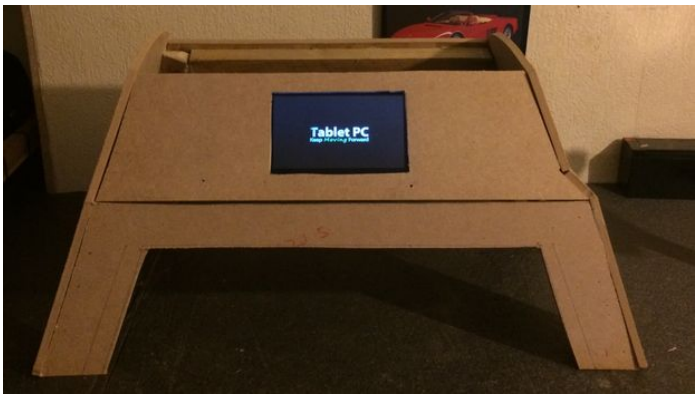

12.0V

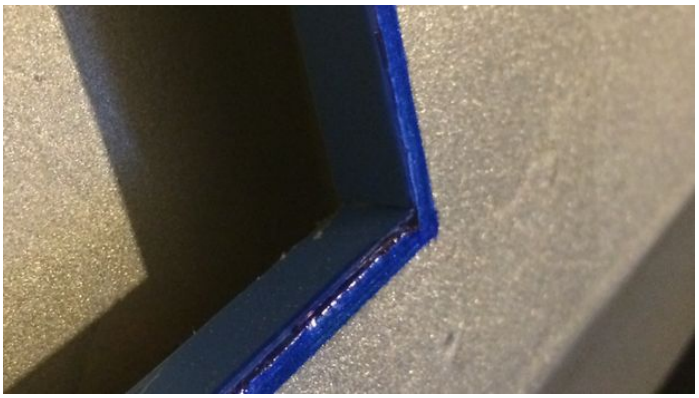


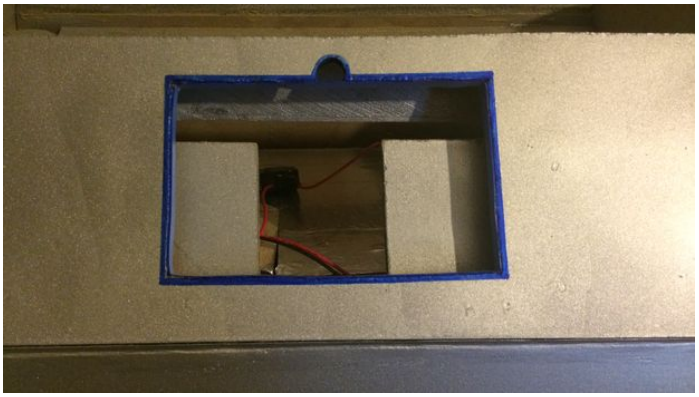

Tablet PC






Tablet PC





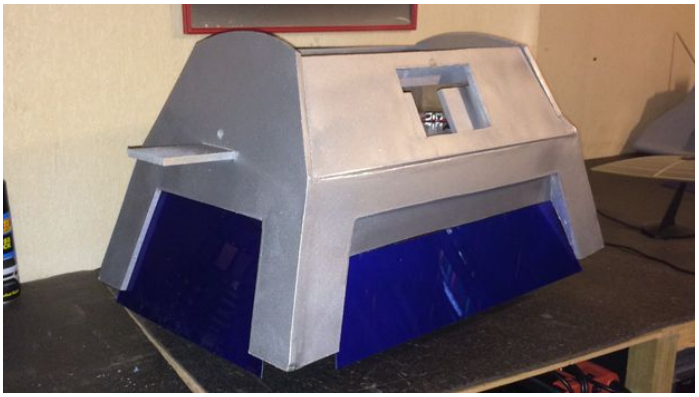
Step 9: Cutting and Attaching Blue Acrylic panels.

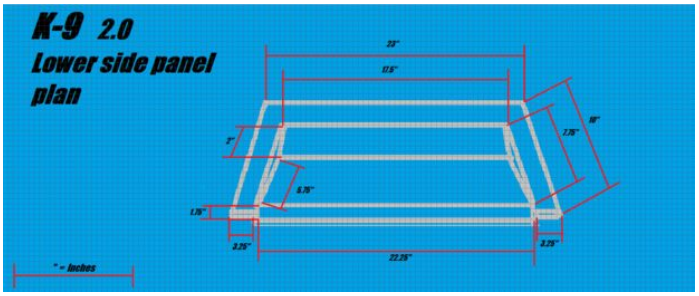
My original idea for the 4 lower body panels made from the acrylic sheet, was to have the entire panels back lit and glow blue, but no matter what I tried to defuse the lights I used, sanding one side of the sheet, using thin paper, repositioning the lights and a few other ideas, it just never looked right so I scrapped the idea. Instead I spray painted one side of the sheet metallic silver with a couple of coats and the end results looked great as it was still a nice shade of deep blue, but glistens when it catches the light. The photos I took doesn't do them justice but check them out anyway for the before and after results. I did also think of tracing a strip of blue EL lighting around the edges of the panels but I thought that K-9's overall finish would make him look like he just rolled of the set of the "Tron" movies and look a bit over the top so I decided against it.

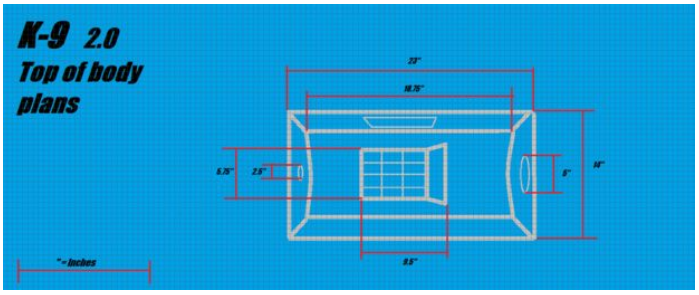
Fitting the acrylic panels.

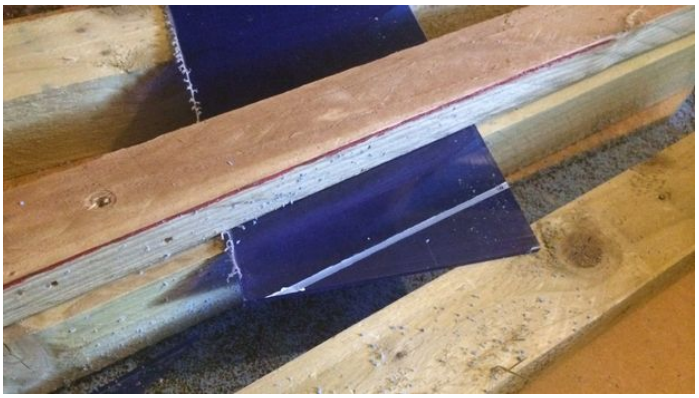
Leaving the protective film on, the acrylic panels were measured, marked and carefully cut using a jig saw with a fine blade, only pulsing the jig saw trigger and not putting any pressure on the saw. Just doing a nice slow cut. This reduced the chance of the acrylic cracking and melting. The holes were drilled around the edge for attaching to the body. The next step was to remove the film from one side of the panels and spray paint them sides with metallic silver.Once dry the rest of the protective film was removed, and the panels were attached to the body using adhesive around the edges, left to set, the screwed on for a secure fit.

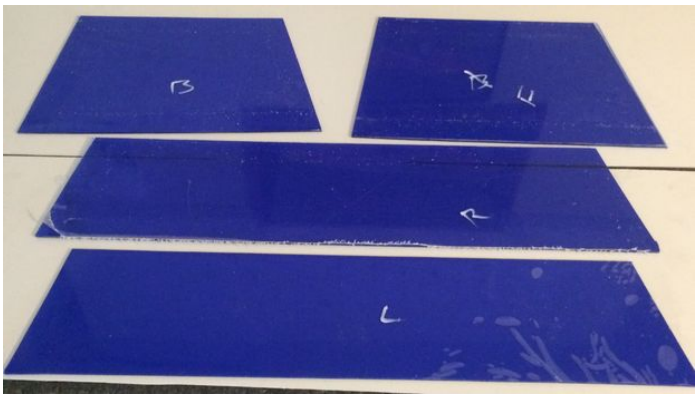
The dorsal panel was measured for the control panel and cut out. Then it was heated with a heat gun and moulded in to a curved shape.





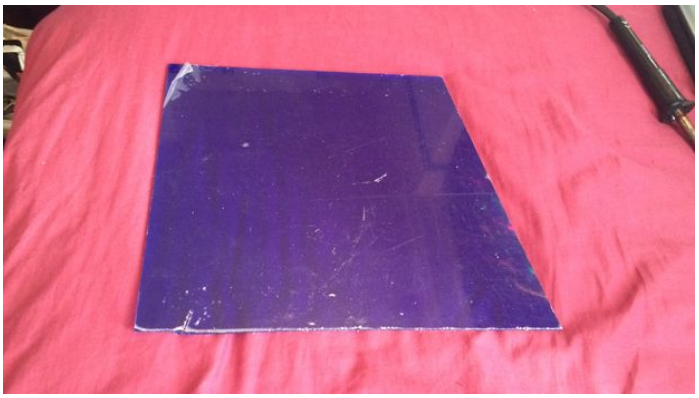


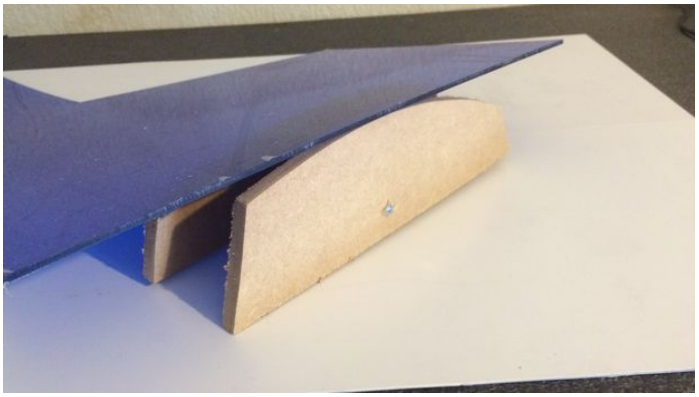


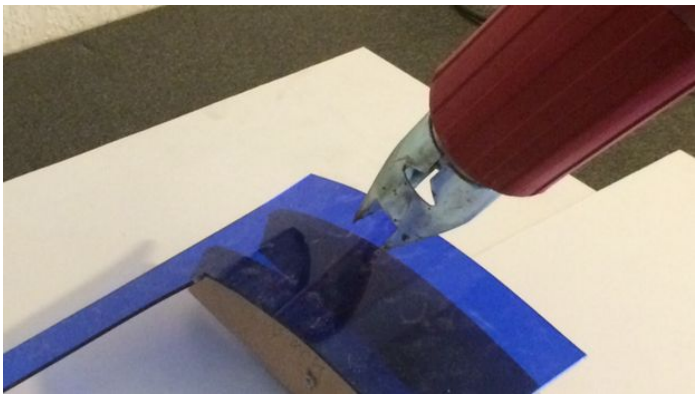

B
B
R
L

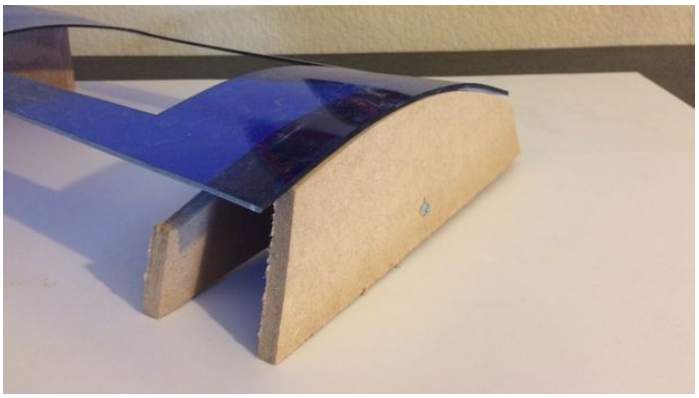

B

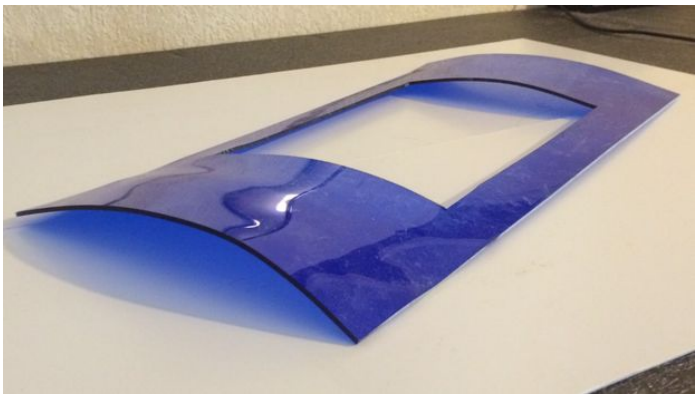

B

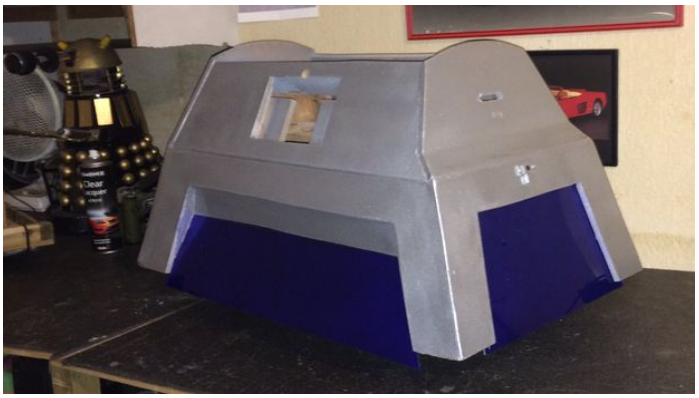


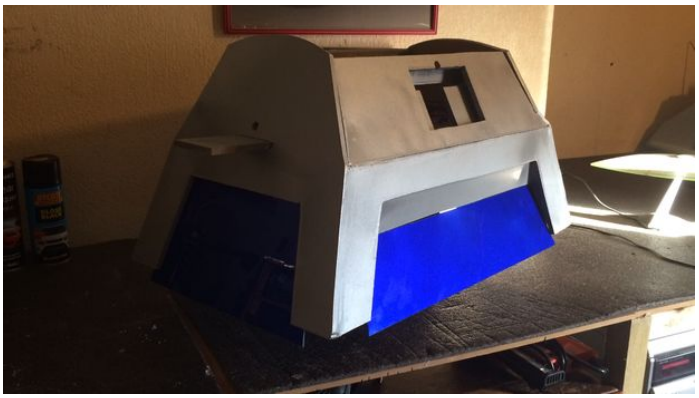












Step 10: Ears, Nose, Tail, and Rear Handle.

Not the "E,N,T" ears, nose and throat clinic, but rather the "E,N,T,H"
ears, nose, tail and handle workshop. Four parts that were quite easy and fun to do and can be done in a few ways, so here's how I made and fitted the parts for my dog.

Part 1. Ears.

1. Using a stereo speaker grill, mark out 2 oval shapes and carefully cut out.
2. With a length of coaxial wire, carefully cut down the length of the outer casing with a craft knife and remove all the inner wire.
3. Feed the spliced wire casing around one of the edge of the speaker grill cut outs so the 2 ends meet and cut off the access wire casing .
4. Peel a small amount of the wire casing back and pipe some super glue down the and replace. Do this on both ends of the casing and leave to set.
5. Repeat the process for the other ear.
6. Following the advice from the Podpadstudios K-9 build, they mentioned that they made the ear stalks to be flexible because of breakage and I wanted to do the same. For this I used 2 rubber antennas I had laying around.
7. Using some thin steel wire, cut off 2 20cm lengths, feed through the back of the speaker grill in the middle, and the width of the rubber antenna.
8. Place the rubber antenna in the middle of the grill and twist the wires around it, a couple of turns with your fingers, then 4 more tight twists with a pair of pliers to secure the antenna.
9. With a marker pen, wrap 1 of the wire lengths around it to make a loop and cut off the access.
10. Do the same for the other to lengths of wire and cut the forth one off at the end of the twist.
11. Move the loops in to position (I had mine positioned wrong in the photos. They needed another 90 degree twist. Easy to do and can be seen in later photos).
12. Repeat the rest of the process for the other ear.
13. Lightly spray 4 to 5 coats with chrome coloured paint. Give them light coats as the holes in the speaker grills can get clogged up with paint.
14. When paint is dry set ears to one side for now as they will be assembled later.

Part 2. Nose cone.

This is an easy one.

1. Using a thick silver market pen, remove the end caps and the insides then rinse with warm soapy water.
2. With a hacksaw, cut through the pen at a shallow angle so the longest part is about 9cm long.
3. Cut the end cap of the marker in half.
4. Paint the end cap silver (unless silver already) then glue back in place on to pen.
5. Lightly sand any signage off until your left with a smooth silver finish.

See. I said it was easy. So put it to one side with the ears for later installation.

Part 3. Tail.

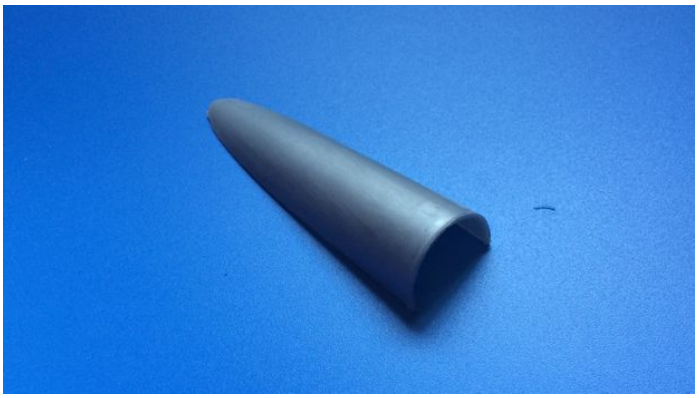
For this section the parts needed are a telescopic aerial, a metal loop clasp from a keyring, a small thin steel sheet (about 2mm thick), a rubber steering rack boot cut down to size, and 4 small screws no more than 8mm long so they don't go through the MDF upper back panel.

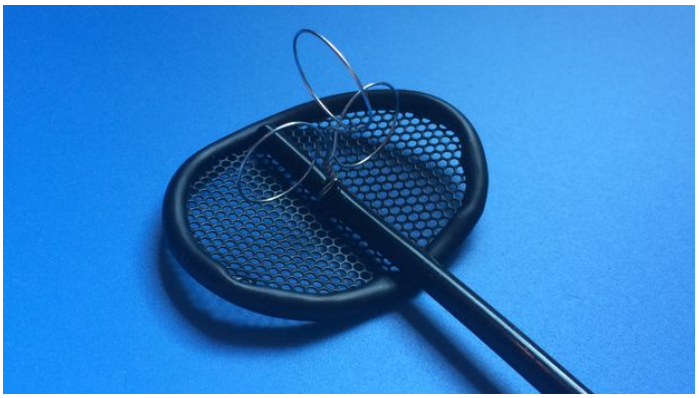
1. Extend the thinner end of the aerial and cut the end off with a pair of wire cutters.
2. Gently squeeze the newly cut end of the aerial then insert a small screwdriver to reform the hole as it may be a bit squashed due to cutting.
3. Hold the keyring clasp with a pair of pliers, and with a second pair straighten out one of the ends to form a 3cm length.
4. Insert the straightened out end of the ring in to the end of the aerial.
5. Squeeze the end of the aerial with a pair if pliers to tighten the grip of the ring. Then on a hard surface carefully hammer the end to make it as tight as possible.
6. Extend the rest of the aerial and drill a small hole through the other thicker end about 7cm from the end.
7. With a saw drill bit (or carefully with a jigsaw) cut a a small slit in the middle of the upper back panel of the body, about 5cm long and just wide enough for the thicker end of the aerial to fit through.
8. Cut two 5x5cm squares from the steel sheet, and bend both squares in half at 90 degrees to make "L" shape brackets.
9. Drill 2 small holes on one side of the brackets and one hole, in the middle, on the other side of the brackets.
10. Screw the brackets to the inside of the upper back panel, just above and below the slit, taking care the screws don't come through the other side. With the 2mm thick bracket and 9mm MDF, 8mm screws will be safe to use.
11. Insert the aerial/tail through the rubber boot, and then through the slit and thread a bolt through the single bracket holes and hole drill through the tail, then tighten with a bolt.
12. Drill 2 small holes through the rubber boot and in to the back panel, one at the top and one at the bottom, then attach with 2 small black screws.
13. The tail should now have a good range of left and right movement. In a few steps time we will be connecting a servo to the tail.

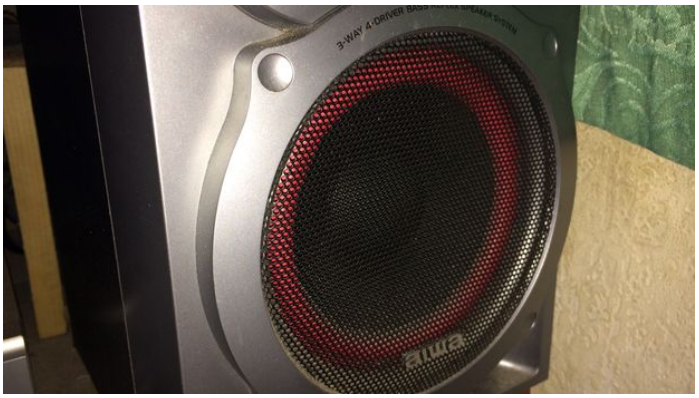
Part 4. The rear handle.

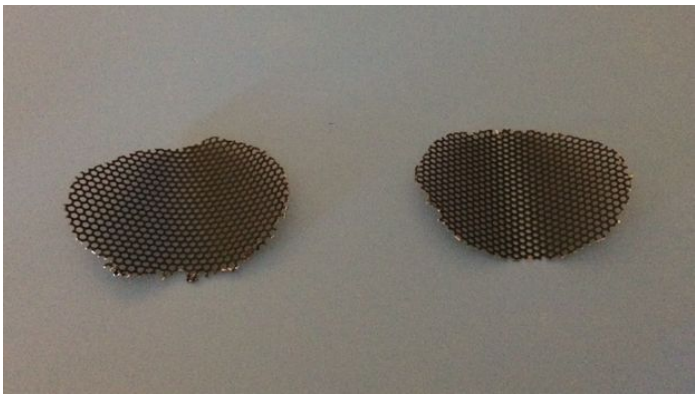
1. Measure the distance of the 2 screw holes of the handle and mark on to the middle of the lower back body panel, just below the charging socket.
2. Drill the 2 holes, insert the screws from the inside, and attach the handle.

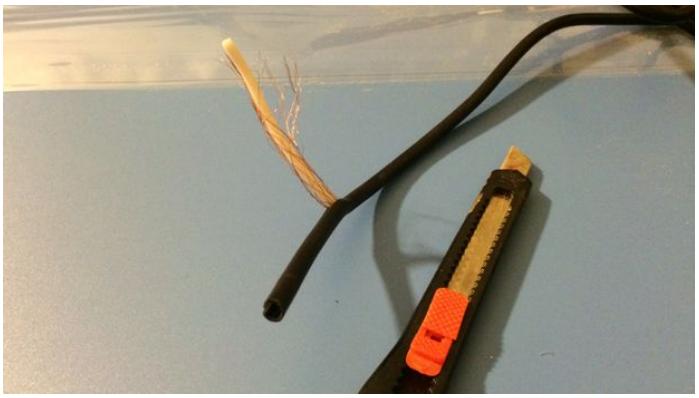
Now this little jobs are done, time to move on the the next major step of the build. The head and neck section.

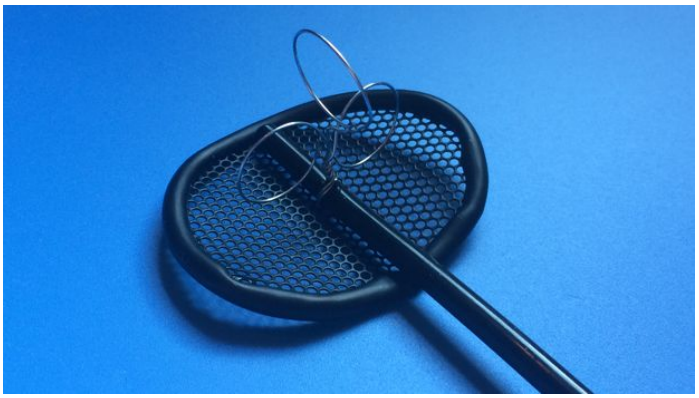


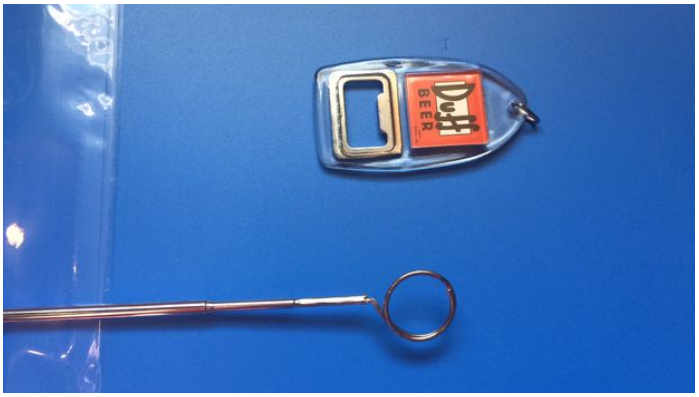


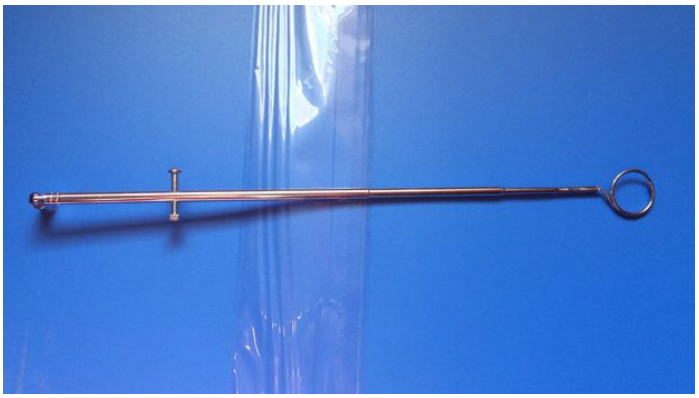

aiwa

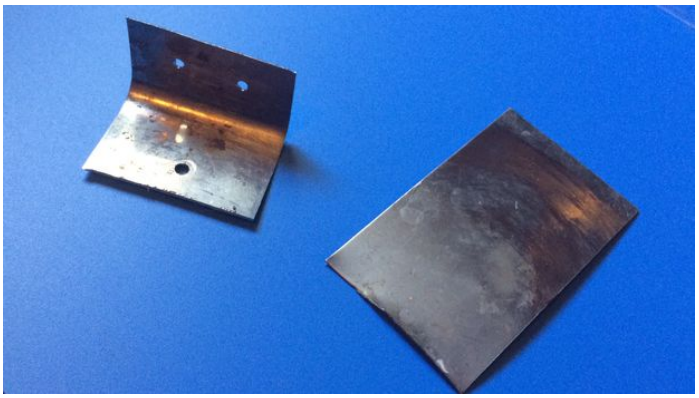


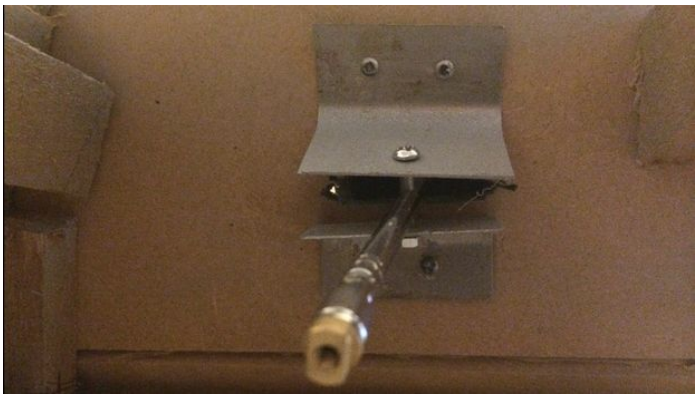





DeH
BEER





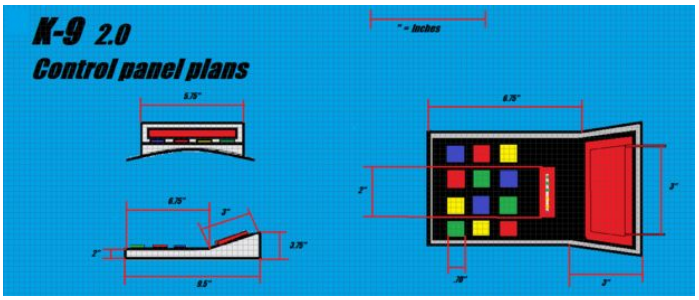


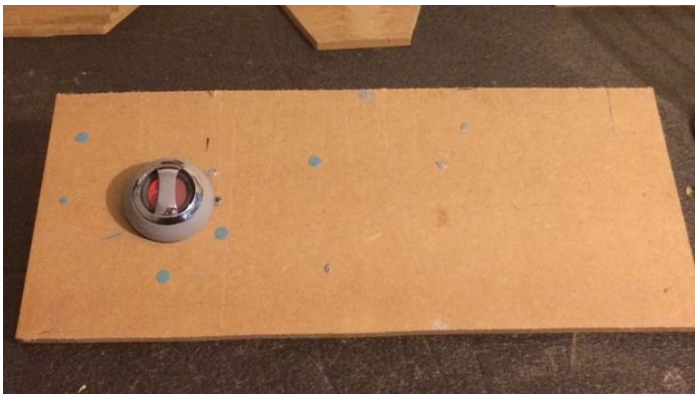

MADE IN CHINA
8 m/m
MARKER

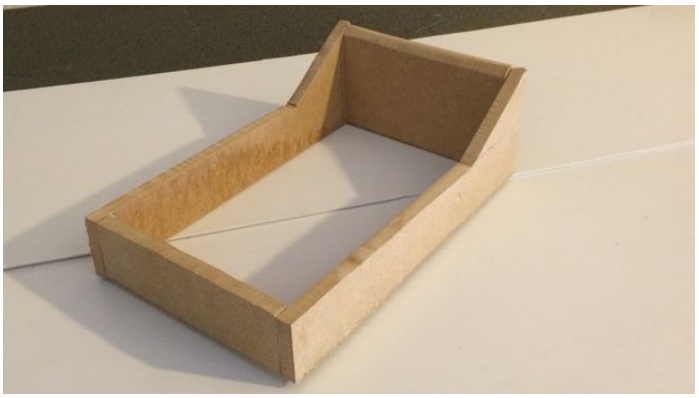
Step 11: Control Panel.

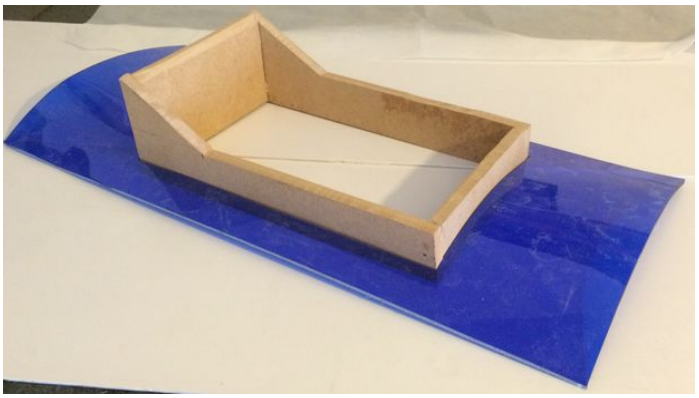
The control panel was next on the list.

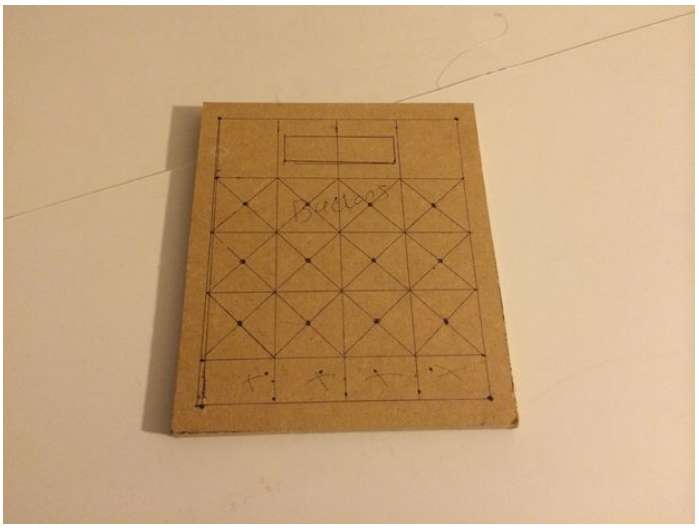
Measuring and cutting more MDF peaches to create the sides, button panel, and matrix display panel. The sides were attached together with adhesive and left to set. The button panel was evenly measured to mark out where the 16 buttons would be place, and the LED battery meter. The the holes were drilled. The two remaining panels were then attached to the frame. The panel was the primed, painted and lacquered. I had 16 coloured push button housings that were stripped out, only leaving the illumination buttons in place. These were placed in to the control panel. Using a cheap set of 20 clear LED fairy lights, 16 of the bulbs were pushed in to the button housings and secured with adhesive. Then the battery mete was fixe in to place. A PVC sheet was measured and cut, a large hole drilled, and the battery compartment and battery meter wires were fed through. Then the PVC sheet was attached to the base of the panel. Finally the Dot Matrix display was attached to the angled part of the panel, and the control panel was set aside until later.

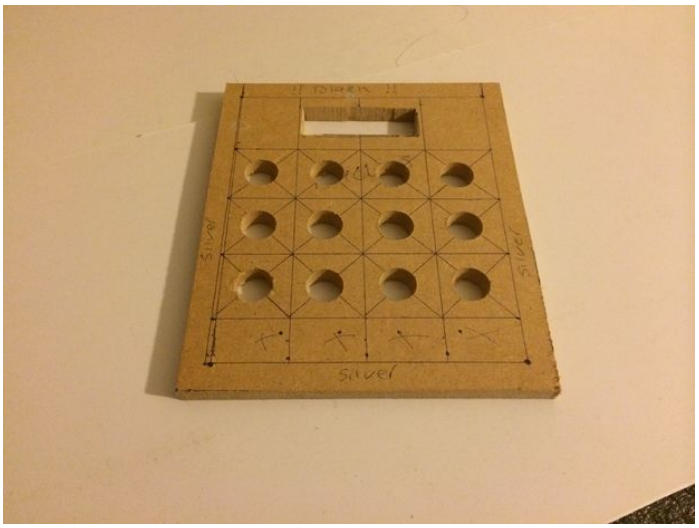


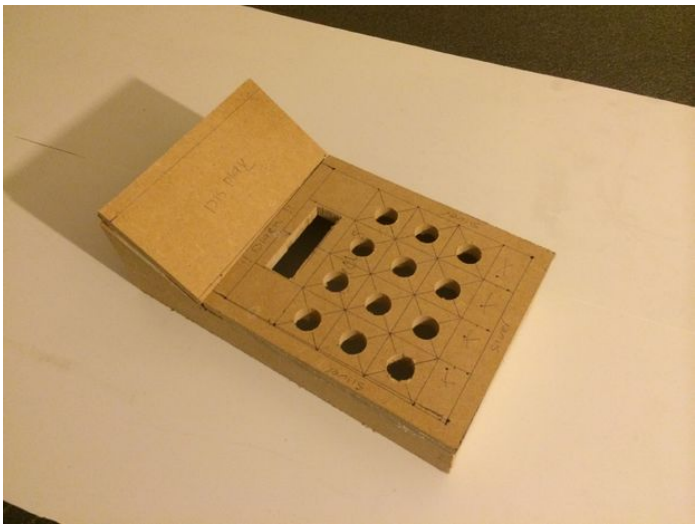


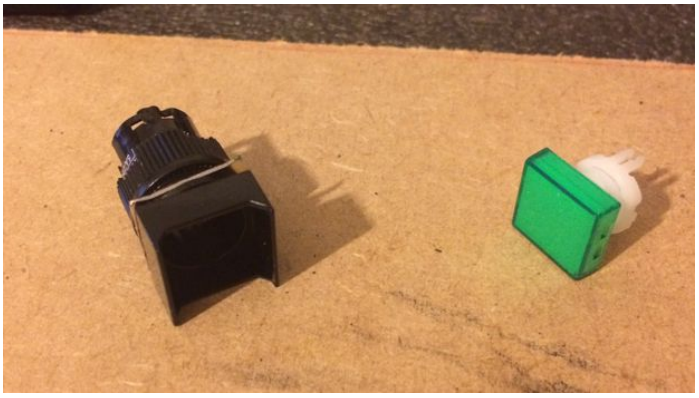


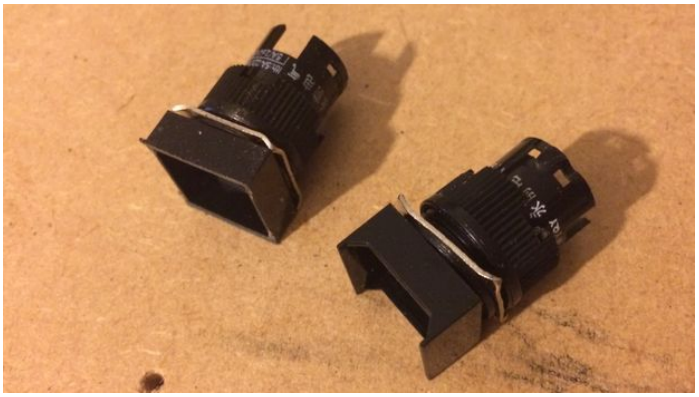


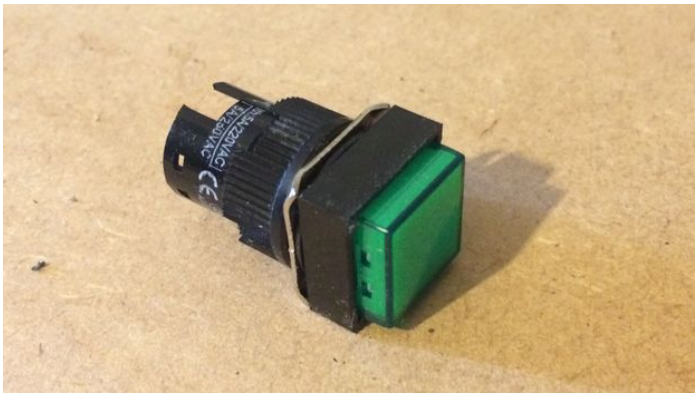


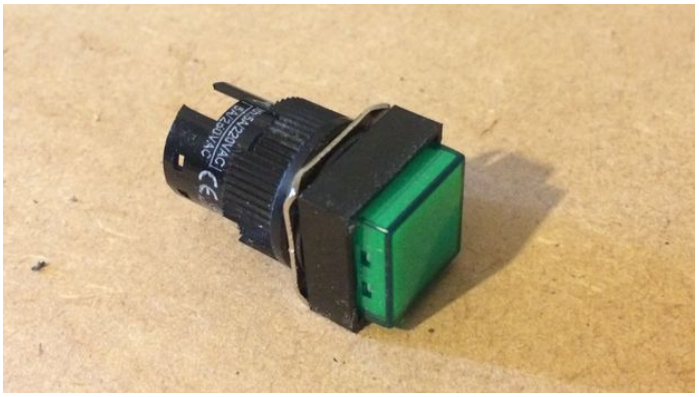

SILVER
SILVER
SILVER

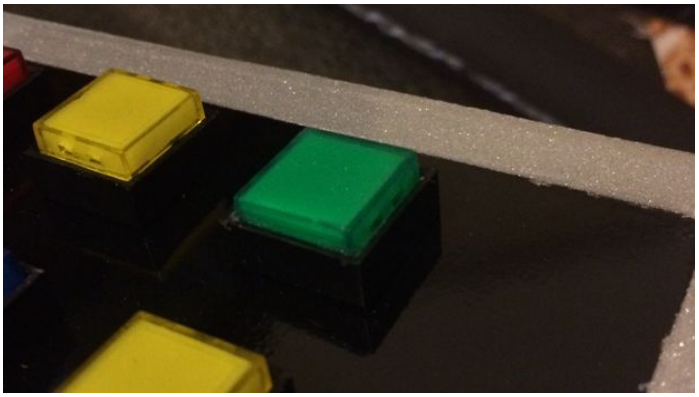


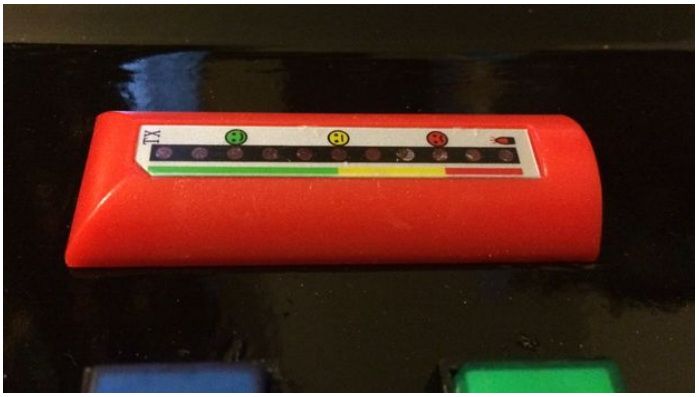








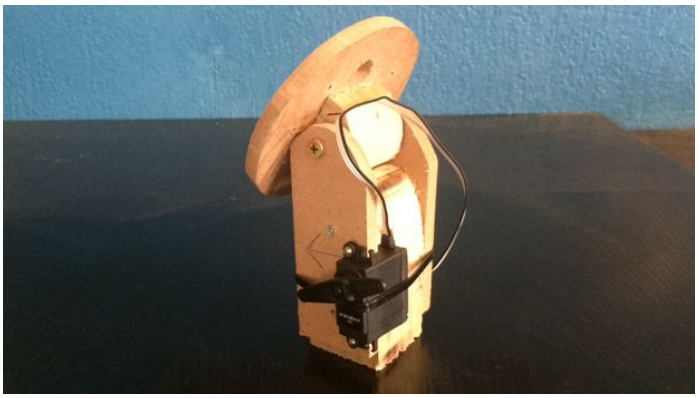


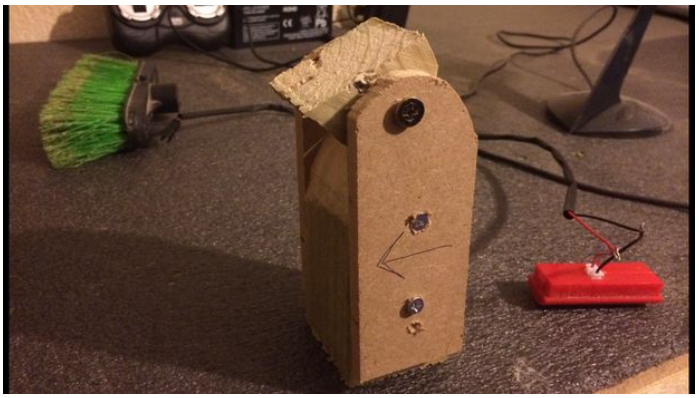


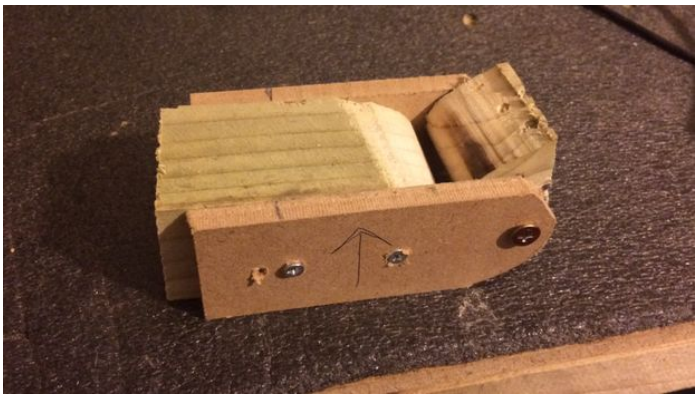
Step 12: The Neck.

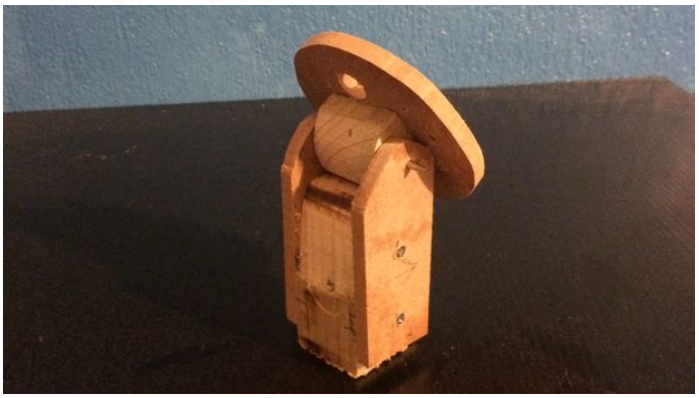
The neck was the next thing to make. Using a short length of 2x2" baton, on end was rounded to a semi circle with a hacksaw. A much shorter peace was cut the same way (see photos). Then two small peaches of MDF were cut, one end rounded, and attached either side of the longer baton. The smaller peace was inserted between the MDF peace's, both curved ends "almost" touching, then a he was drilled through the MDF and short baton peace, then a long screw was inserted through the holes making a pivot.

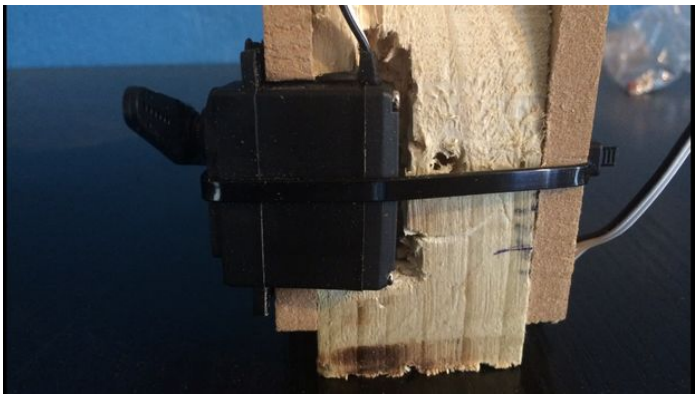
A MDF disc was cut and attached to the pivot. Two electrical terminal blocks were attached either end of the under side of the disc. A further square peace was cut and attached to the base of the neck. Then a Lazy Susan bearing was screwed on to the square MDF peace.A shallow recess was cut in to the baton, and a 90 degree centred Heavy Duty servo with a two sided horn was attached. Two lengths of steel wire were measured, cut, and bent in to shape to create linkages. These were attach to the servo horn and fed in to the terminal blocks. The finished neck mechanism was attached to the next platform protruding from the body ensuring the pre-drilled hole and bearing centre were aligned. Finally a 90 degree centred Heavy Duty servo was attached to the under side of the neck platform and screwed the servo horn to the base of the neck mechanism.



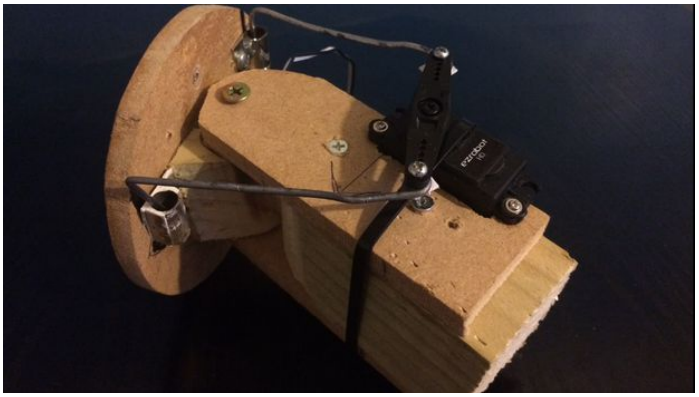


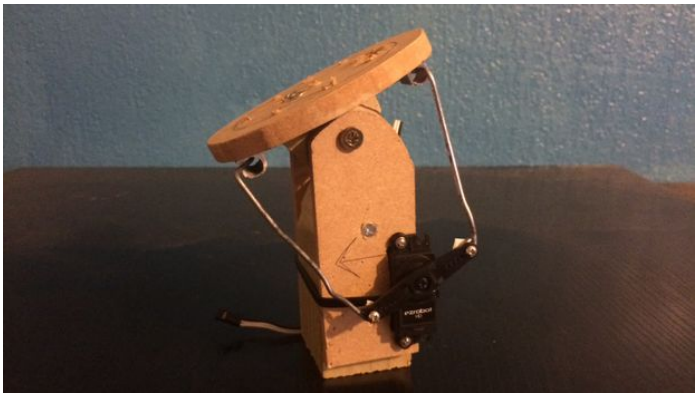


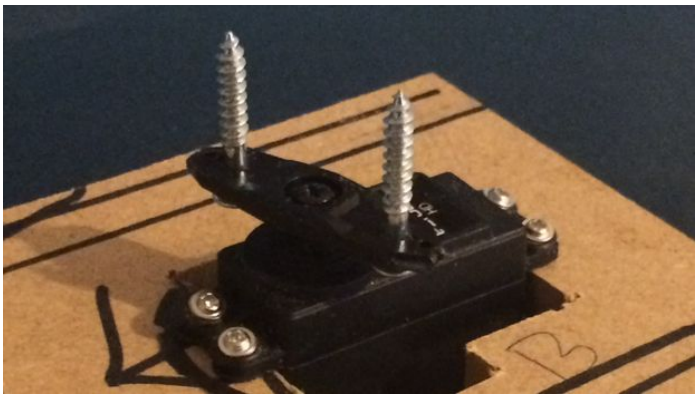


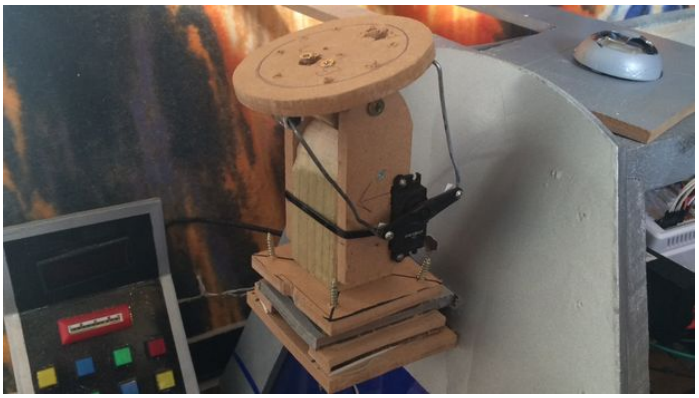


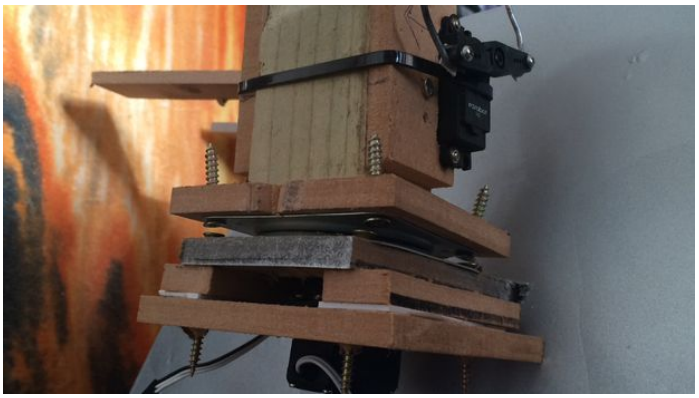

ezrobot





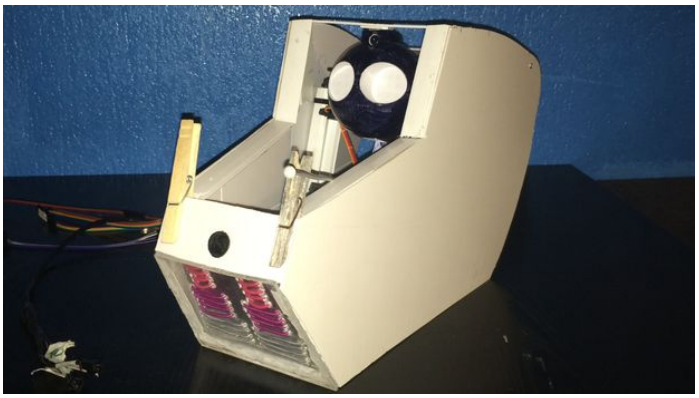


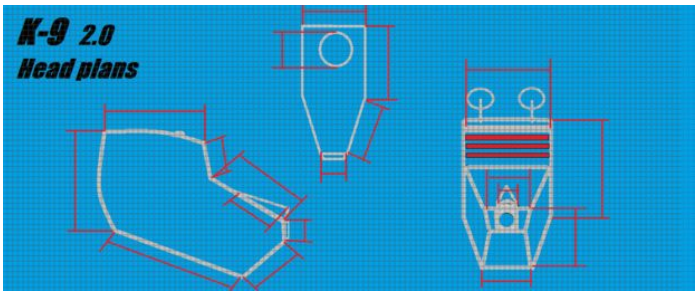


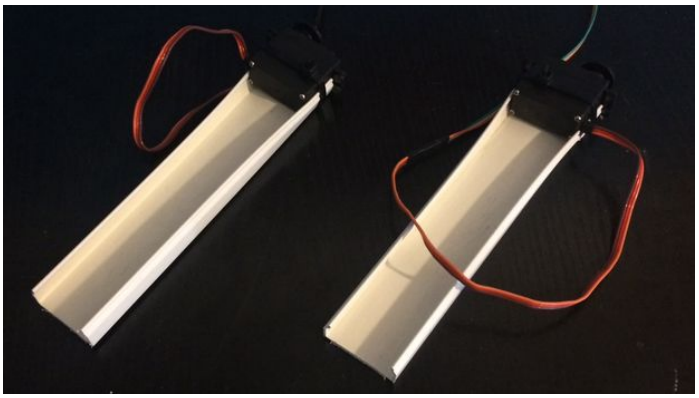


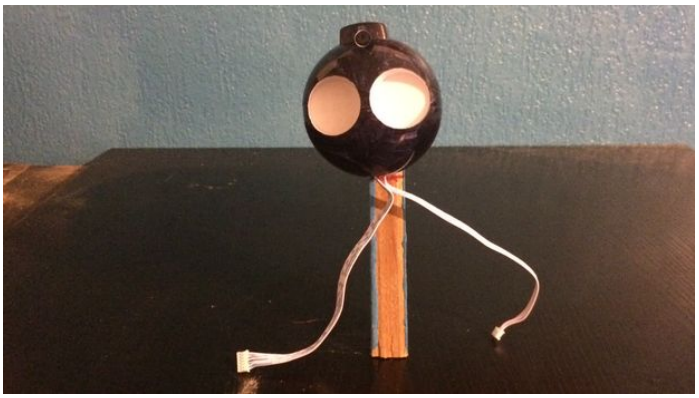
Step 13: Starting the Head.

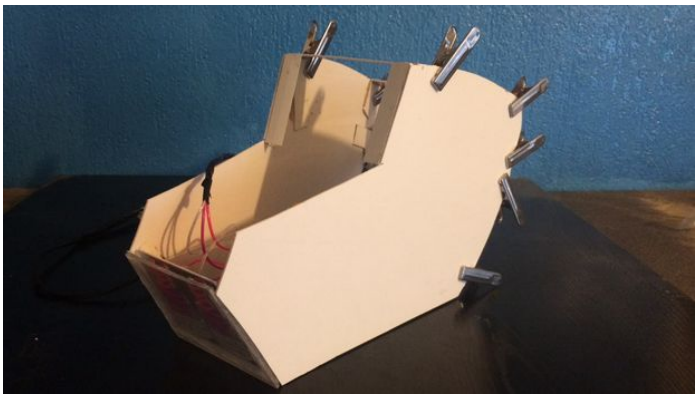
The head was made out of PVC plastic sheet to keep the weight down. The base and side panels were measured and cut to shape, the attached using some PVC wire trunking and clear silicone adhesive. The top/back head panel was measured, cut and heat treated with a heat gun to bend it in to shape. A scrap MDF head template was made to rest the PCV sheet over to heat and shape the panel. The humanoid head (eye and camera array) was fitted inside of the head with ribbon wire extensions attached to the RGB LED and camera wires. Then the two micro servos were fitted along with servo wire extensions for the ear rotation.

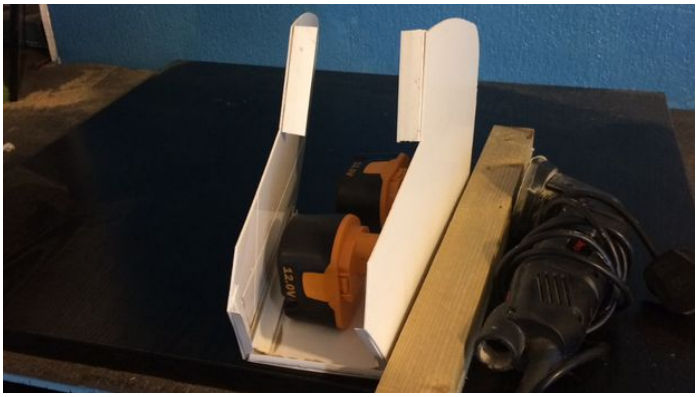




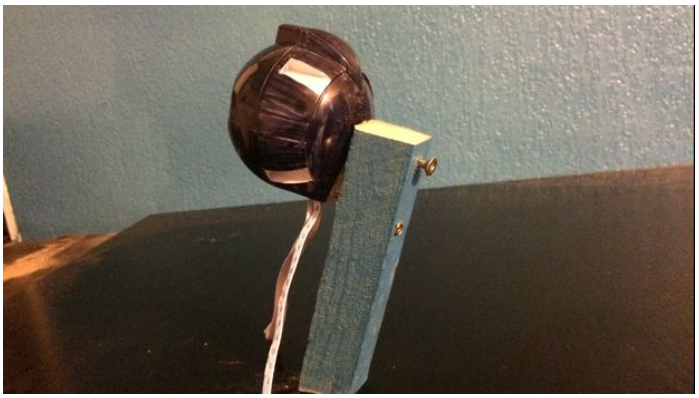


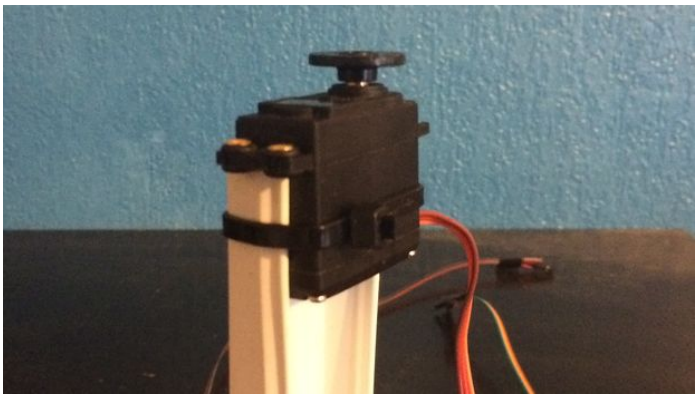


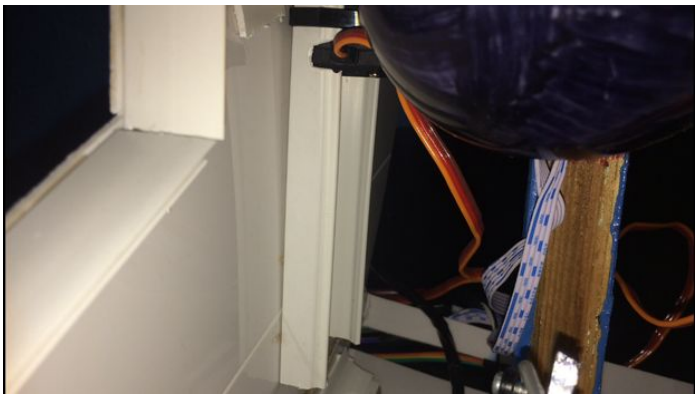


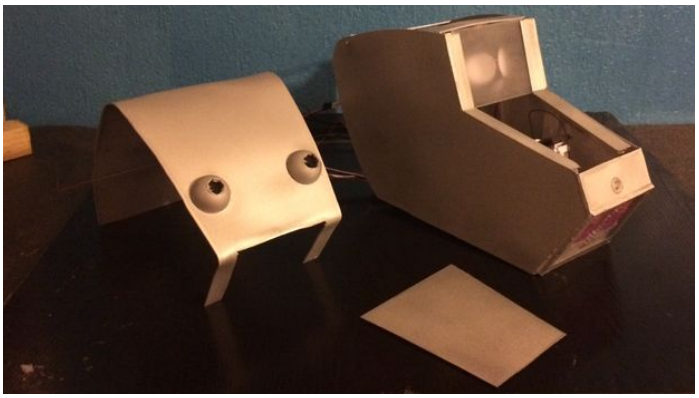



2509
2509



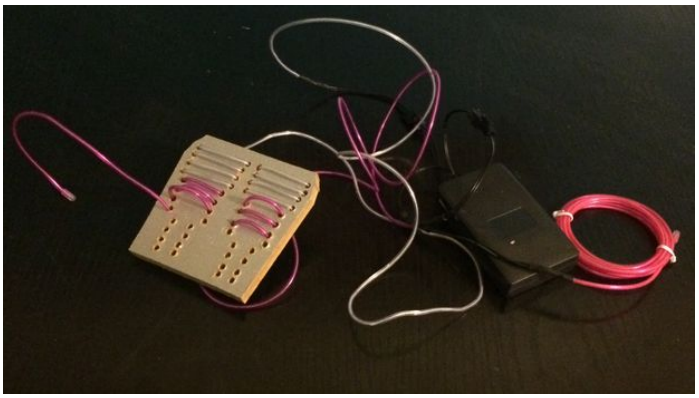


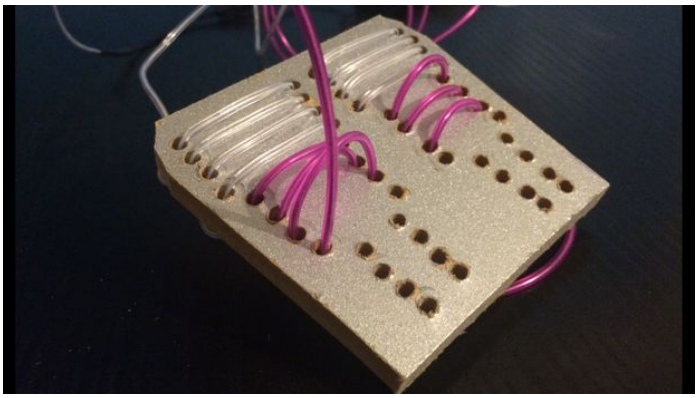


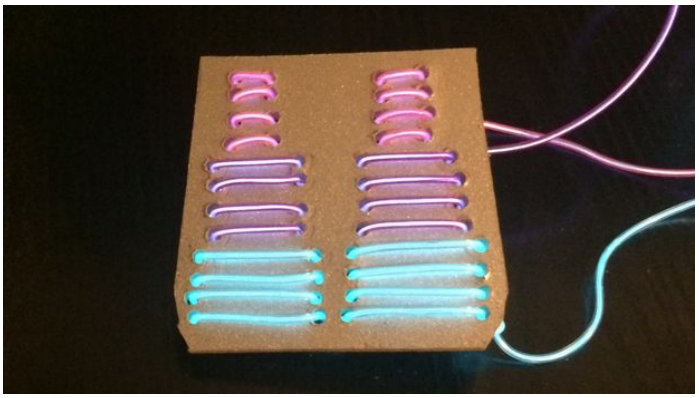


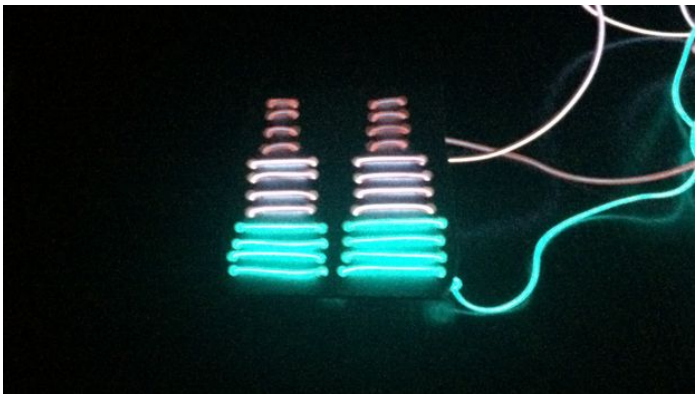
Step 14: Mouth panel.

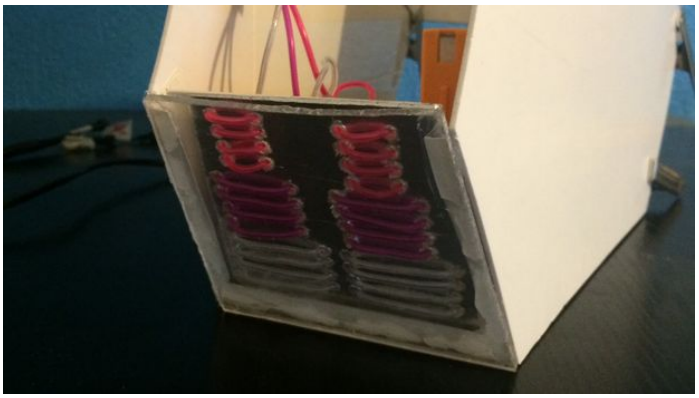
The mouth was made using a peace of PVC which was measured, one side covered with aluminium tape, and evenly drilled to create the mouth pattern. Then the Electroluminescent wires were fed through the holes and pulled tight. A clear peace of perspex was cut and fitted to the outside of the head and the mouth panel fitted in place on the inside.





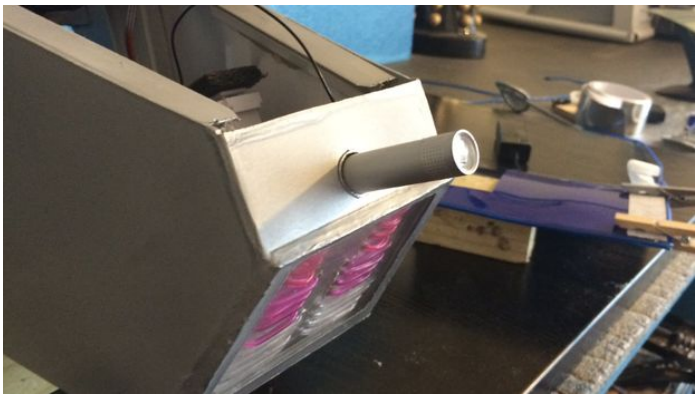


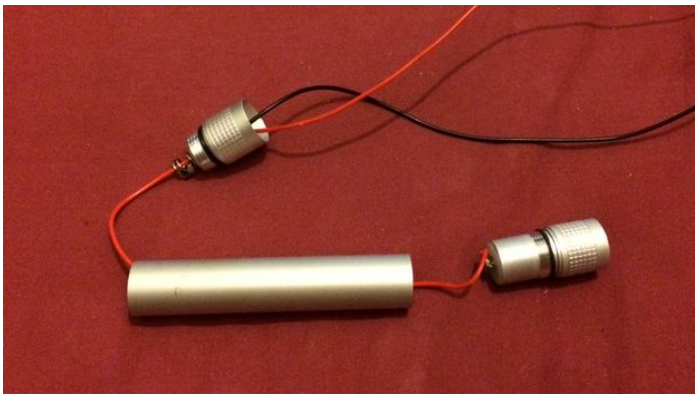




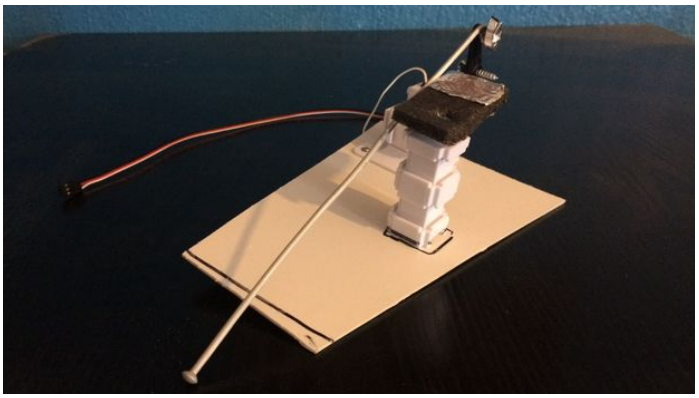
Step 15: Nose section.

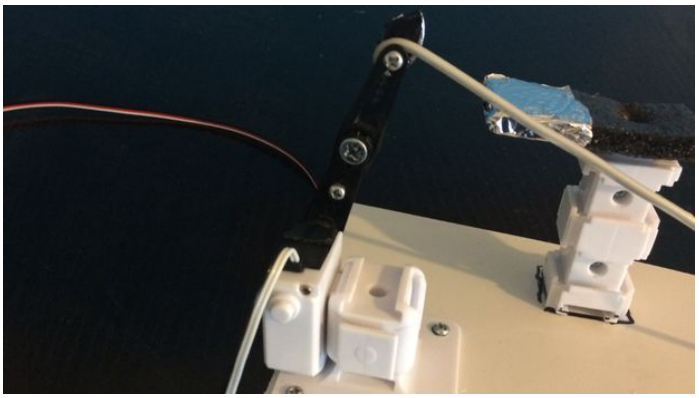
The nose was made using a permanent marker pen shell, a penlight torch, momentary switch, micro servo and PVC peace's. A PVC platform was cut and the servo attached to the rear end. An extended servo arm was made using two horns, with a screw attached to the centre of the arm to activate the switch, and attached to the servo. A momentary switch was the fixed in to position where the screw would meet and press it. A peace of PVC sheet was cut and a hole drilled through the centre. The marker pen shell was fed through the hold and secured. The torch was stripped, wires soldered to it, fixed back together, a hook attached to the end and placed in to the pen shell. One of the torch wires were cut and wired to the switch. Finally a length of steel wire was measured and cut to make a linkage, the attached to the torch hook and servo arm. The nose mechanism and front panel were fitted to the head.

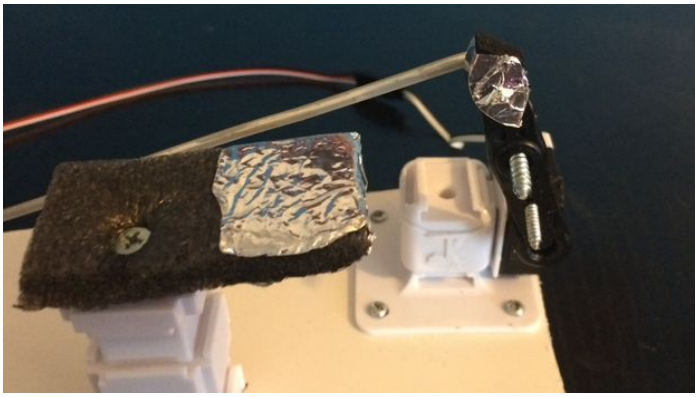


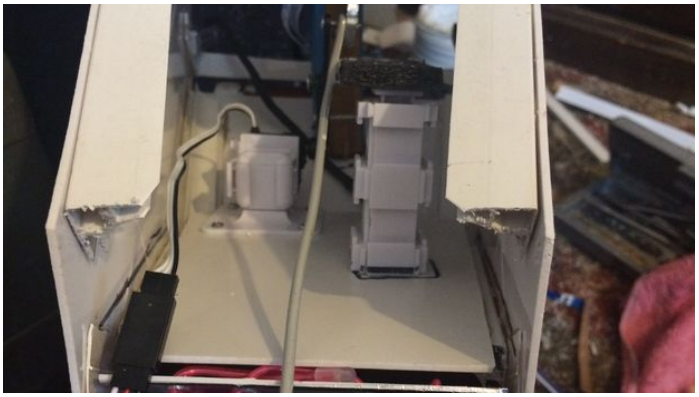


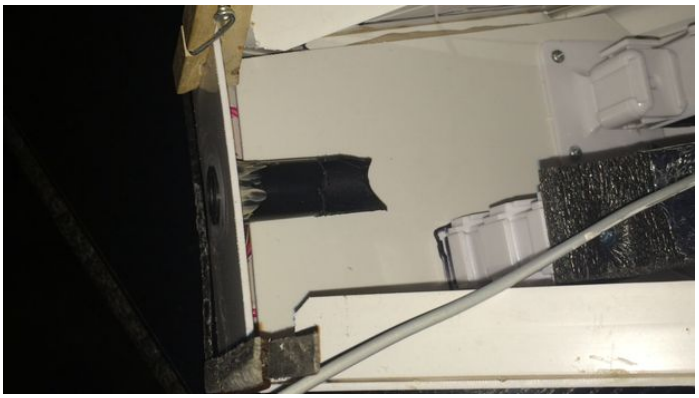

neutral smell
lightfast
waterproof
geruchsneutral
ohne zusatz von xylol / toluol
lichtbeständig
wasserfest
waagerecht lagern







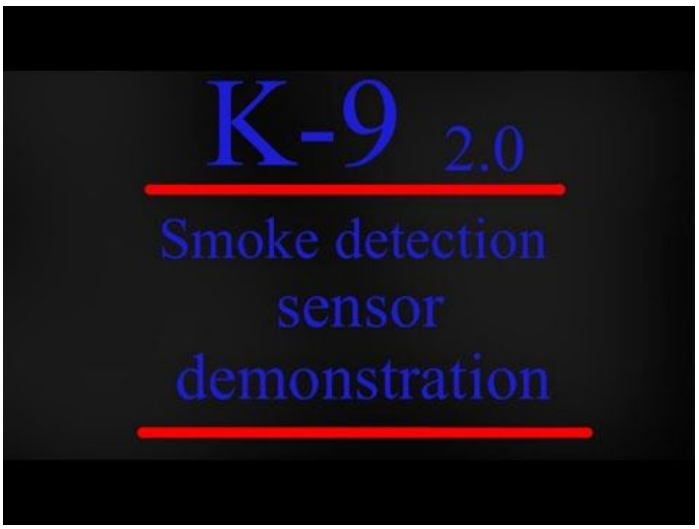


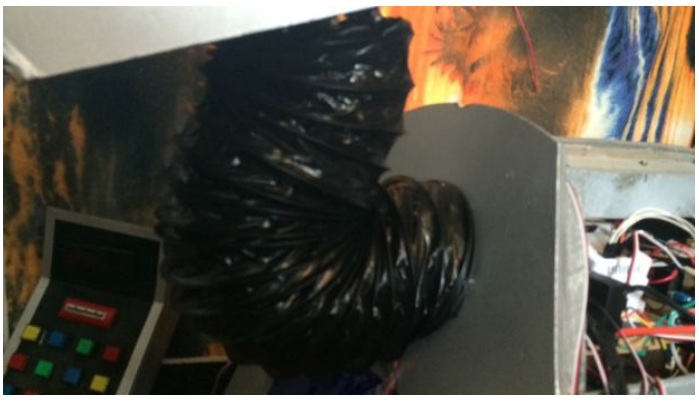


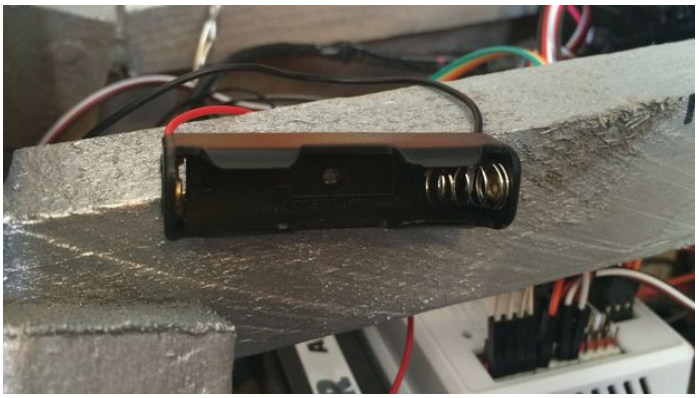
Step 16: Finishing the Head.

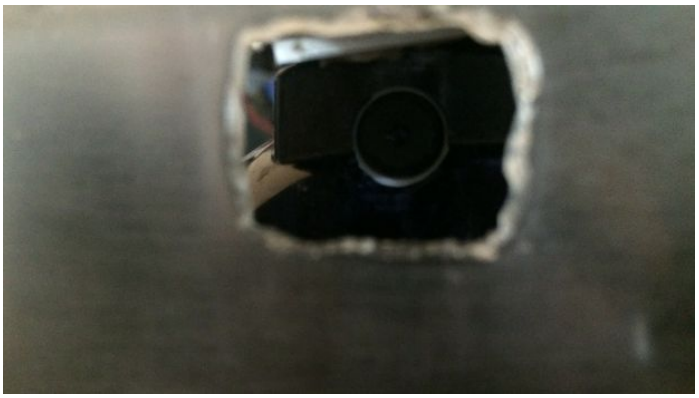
A hole was drilled and all the wires were fed through it. The inside of the head was covered and masked up, the the head and top head panel were primed, painted and lacquered. Once dry everything was unmasked and the top head panel fitted. A clear piece of Perspex was cut (with a small hole cut for the camera), lightly sanded on one side, then attached to the eye panel. Then small lengths of blue acrylic were cut and attached, evenly spaced, to the outer side of the eye panel.

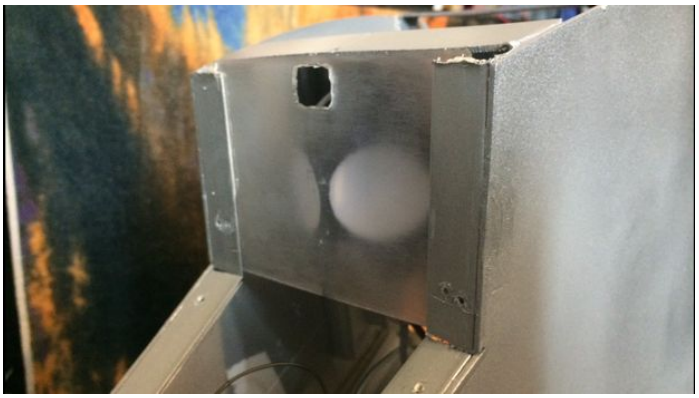
The blue acrylic nose panel was then screwed on. The silver nose fitting made from the jumbo marker pen was attached to the acrylic nose panel. Finally a drop of silicone was applied to the ear servo horns and the ears inserted and left to set.

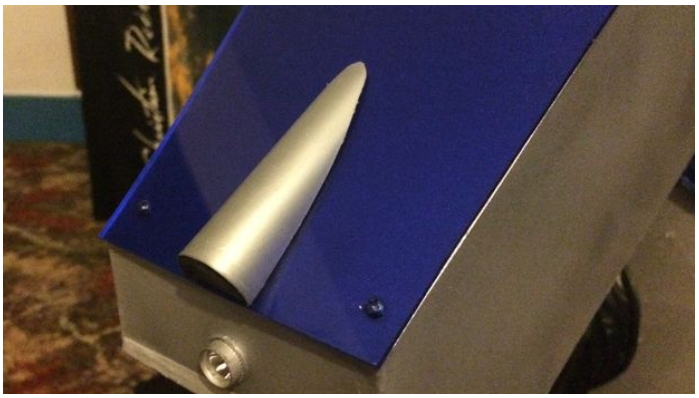


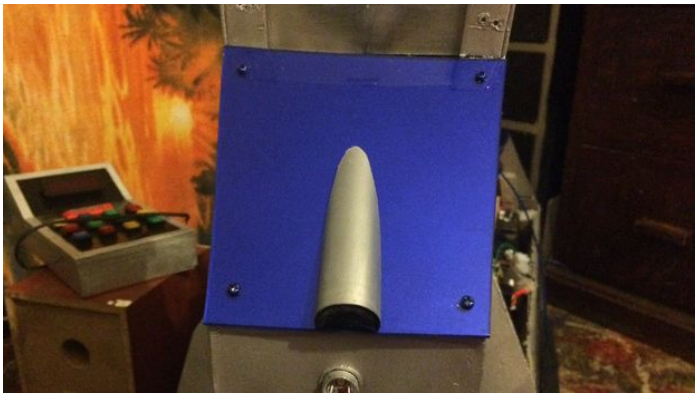


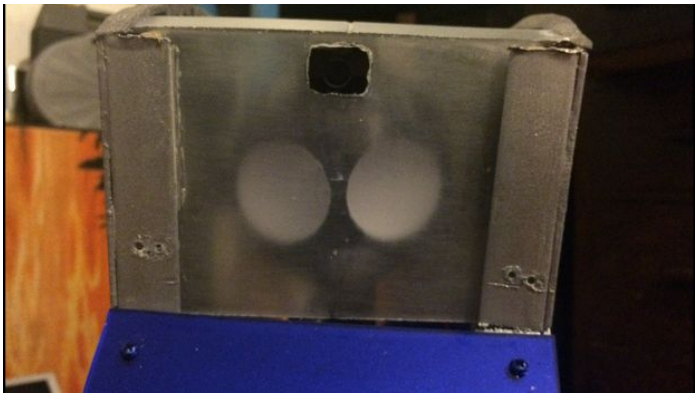


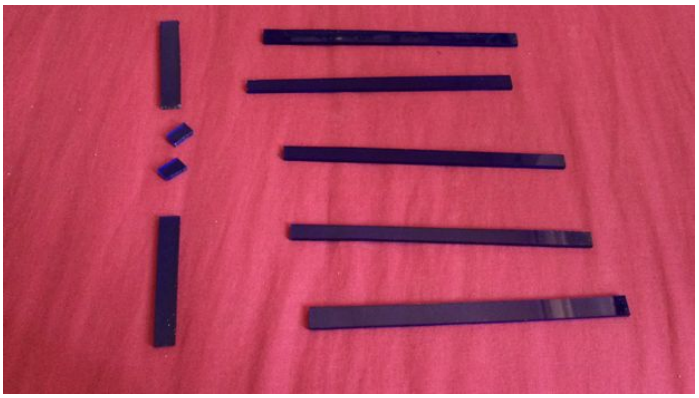


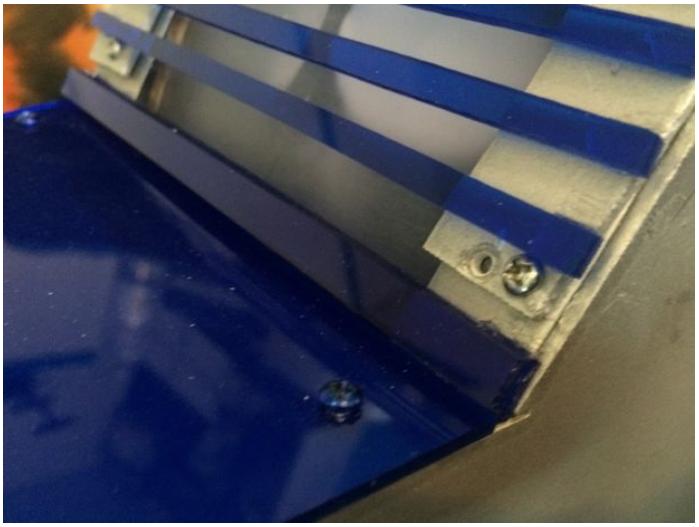












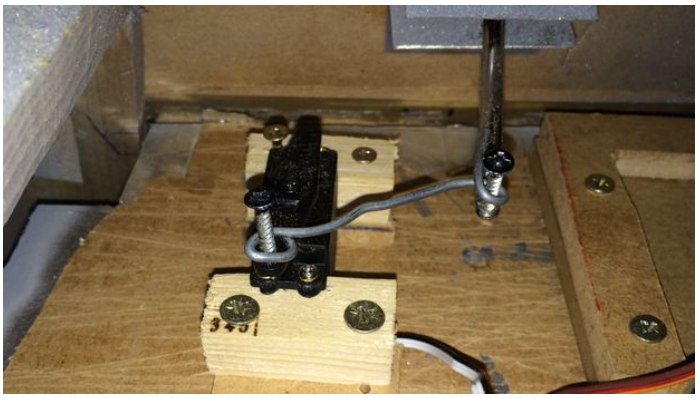
Step 17: The Electronics.

I cut a sheet of MDF to make a platform and fitted inside of the body. This is where the majority of the electronics would be housed. Then it was time to break open the EZ-Robot Development kit and mark out where everything was to sit.

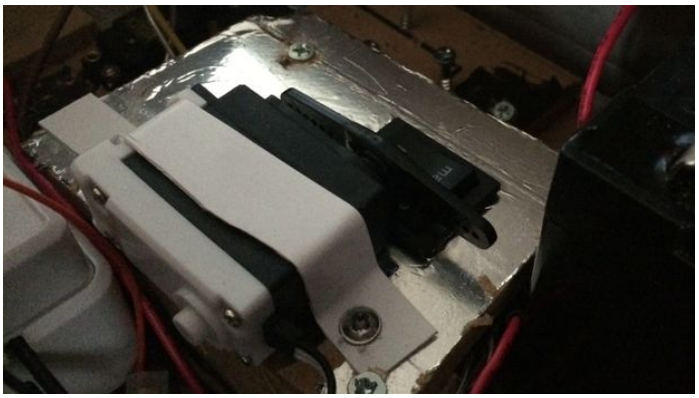
After marking out where the servos would be situated, I started by attaching the tail servo. It was screwed in place and a steel rod was cut and shaped, then attached to the servo horn and tail. Next servo to bed fitted was for the control panel and chassis lighting. Using some scrap MDF peace's, I cut and attached them to the platform then housed the lighting battery packs. Then the servo was fixed in place making sure the servo horn could reach the flashing mode buttons on the battery packs. The next servo was that was fitted was for the dorsal and collar EL wire lighting. Some screws were fixed in place to house the battery pack, and the servo fitted ensuring the servo horn could activate the flashing mode button. The last servo was fitted for the switch that would turn all the lights on and off. The addition of a toggle switch was used for this.

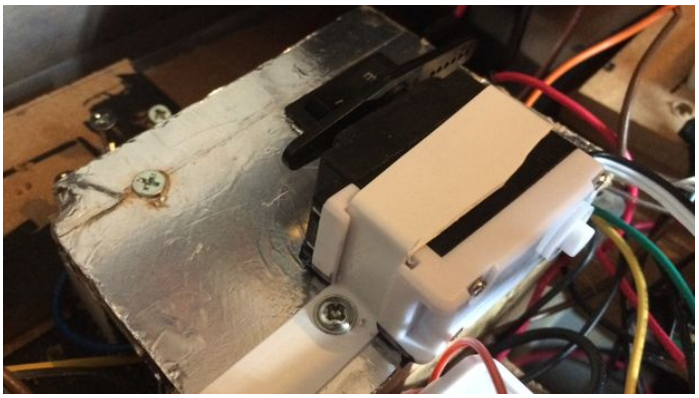
A cut in the ground wires of all the lights would be connected to the toggle switch and fixed in place under the servo horn. When the servo was activated, this would toggle the switch on and off. To remotely power On/Off K-9's main systems, a two channel 12v remote control relay switch was fitted. The power input would be the from the 12v battery and the relay terminals would be wired to the EZ-B microcontroller.H-Bridge motor controller with regulator. The next item to be installed was a two channel 12v, 10 amp motor controller with H-Bridge. The two drive motors were connected and 5v regulator was fitted to the power input terminals. Instead of fitting a telescopic sensor to K-9's eye panel (like the one seen in the TV shows) I wanted to fit something useful. I went for a smoke and gas detection sensor. This was fitted behind one of the blue acrylic eye panel strips and connected to wires u previously installed for this purpose. Now all the electrical peripherals were fitted, it was time to connect them all up and give K-9 a brain. But before that, a little hack was needed.

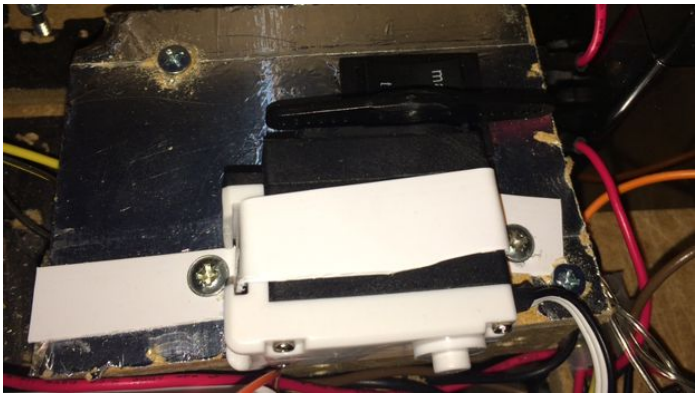

ezrobot
BUILD • PLAY • LEARN

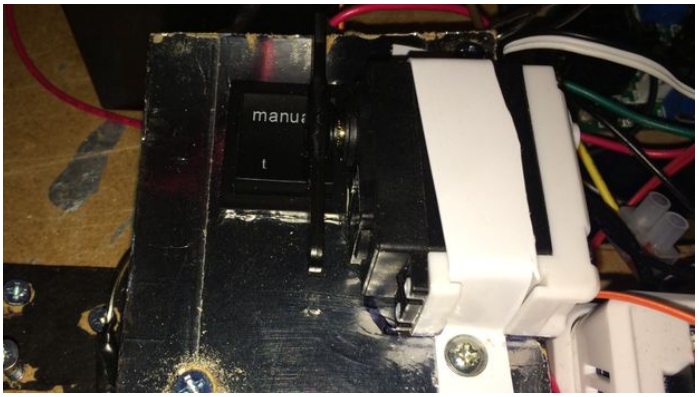



testarossa

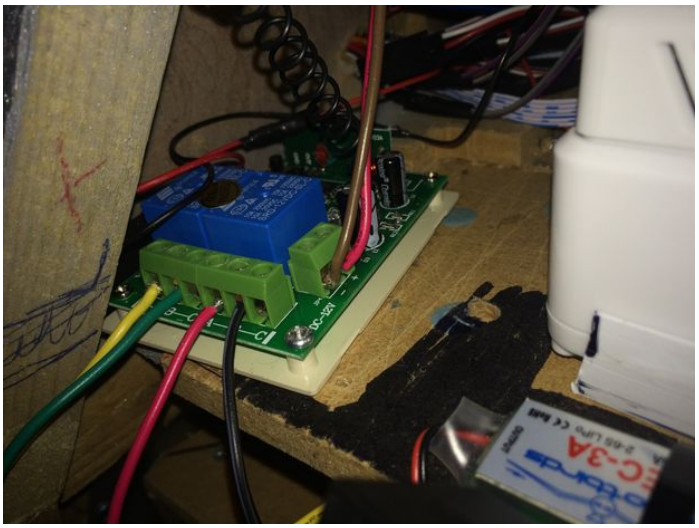


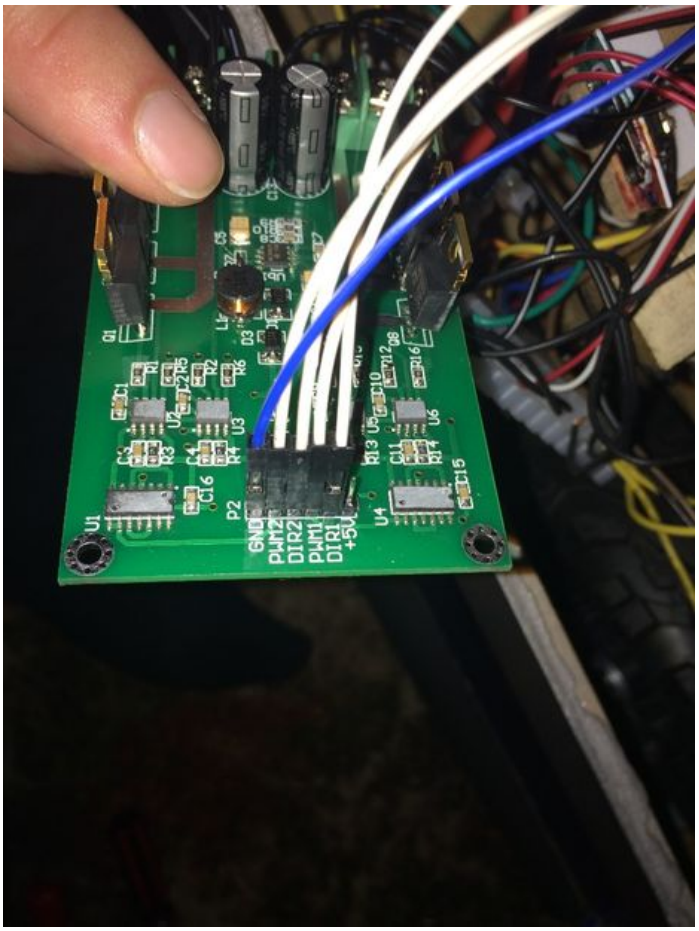


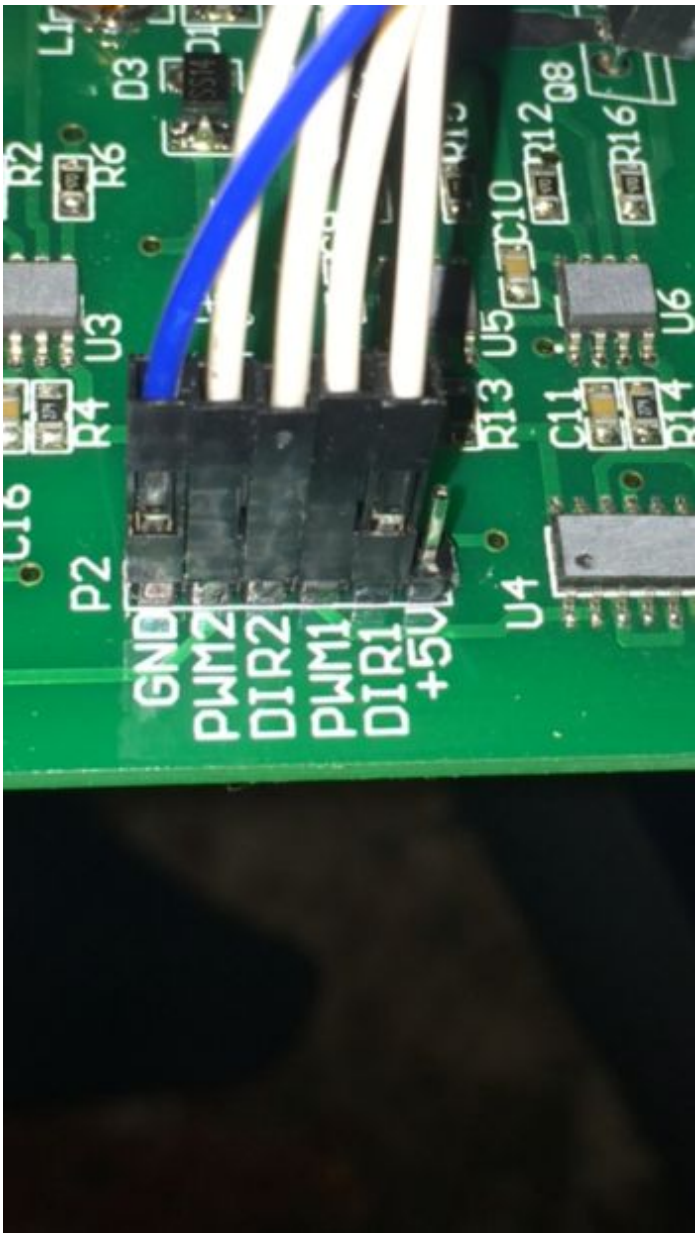


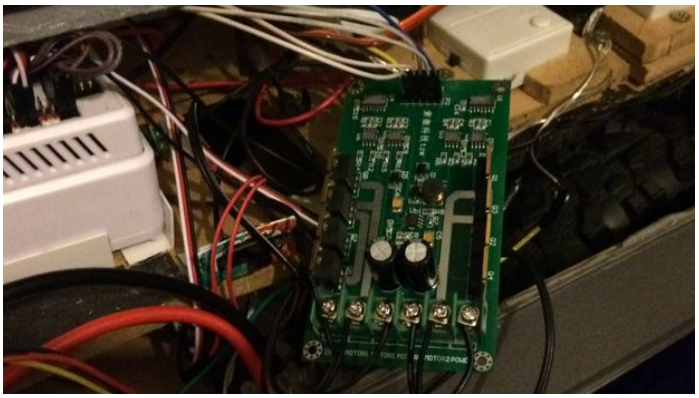

manu

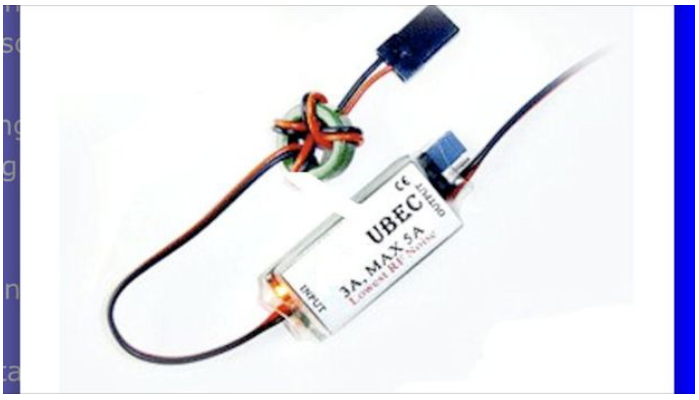

A
B

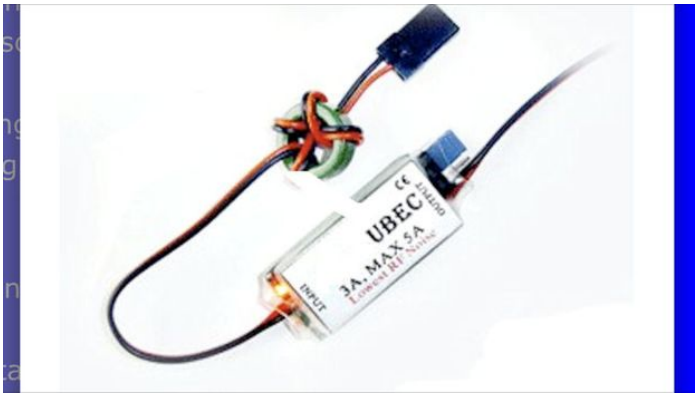


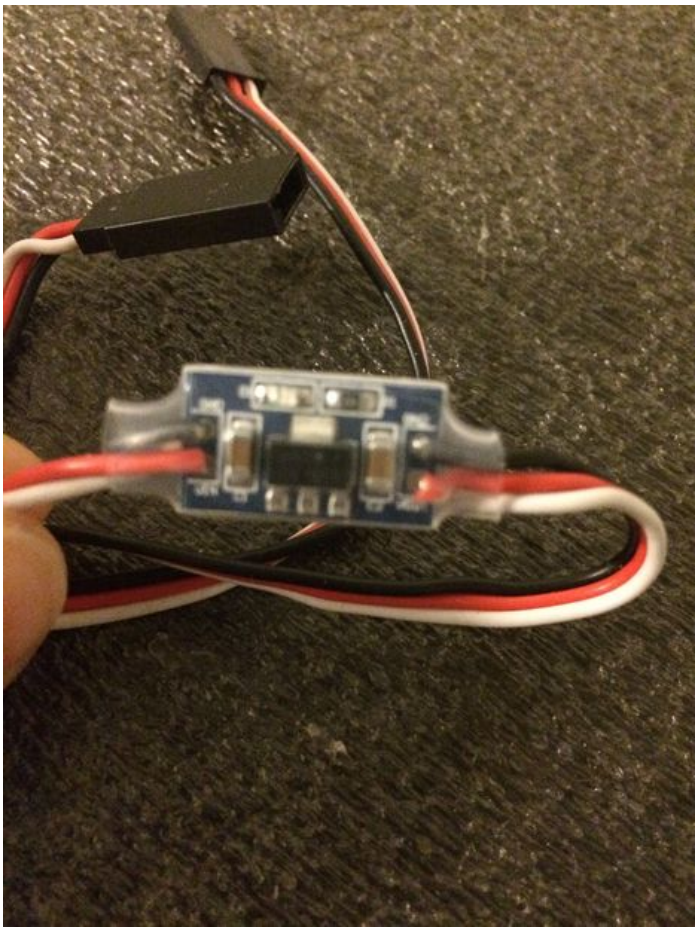

GND
PWM2
DIR2
PWM1
DIR1
+5V

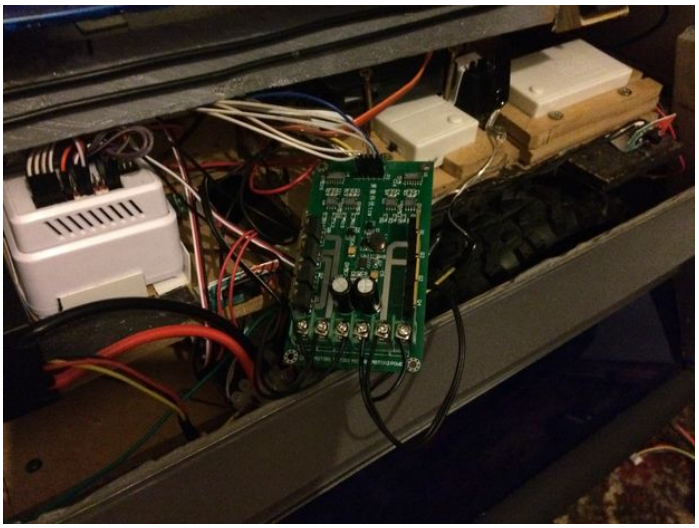

D3
R2
R6
U3
R4
C16
P2
GND
PWM2
DIR2
PWM1
DIR1
+5V
U4
R13
C10
U5
R12
Q8
R16
U6
C11

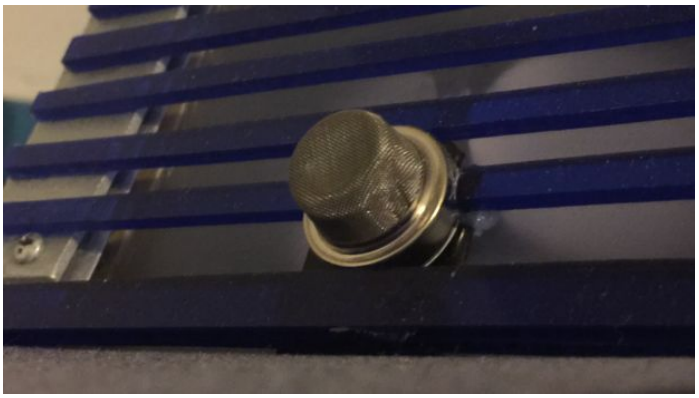


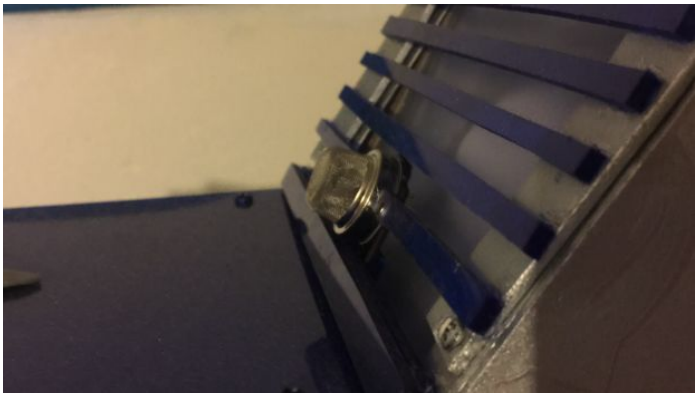

UBEC
3A MAX 5A
INPUT

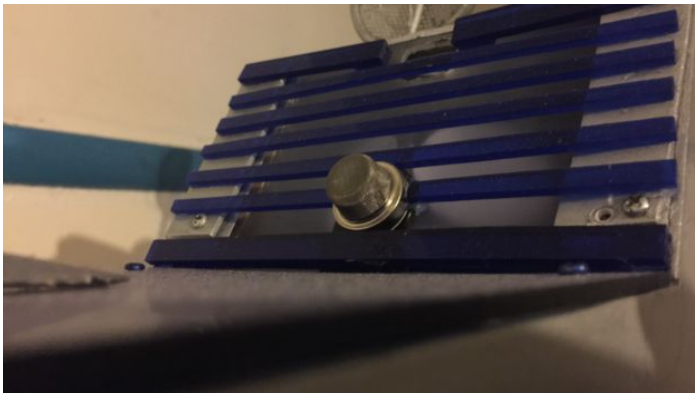

UBEC
3A, MAX 5A











Step 18: EZ-B v4 Sound Breakout.

The EZ-B v4 comes with a small speaker already installed and sounds great, but I was going to use a larger amplified external speaker, so I needed to do a small hack to the EZ-B's circuit board to attach a headphone socket. Here is how it's done.

1) Open up the EZ-B v4 casing and remove the PCB's and small speaker as you won't need this now.

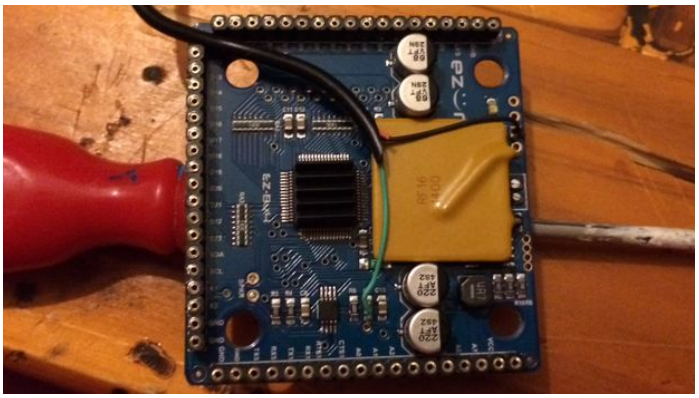
2) Carefully prise the two PCB's apart and put the board with the connection pins to one side.

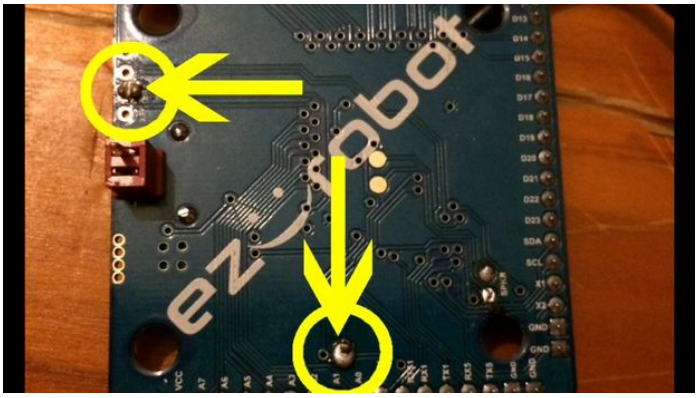
3) Using an old headphone lead, strip the wires and solder the bare copper wire to the "Ground" terminal, and the shielded wire to the "positive" terminal.If your using a non amplified speaker you can solder the wires to the "SPK" solder terminals. This is because the EZ-B v4 Has a small on-board amplifier.

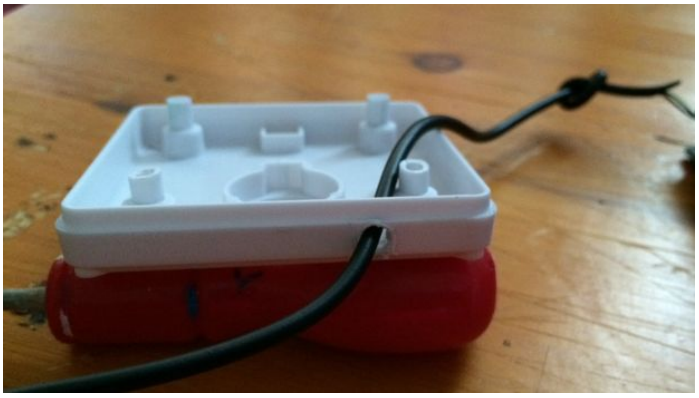
4) Then sandwich the two PCB's back together and replace the casing.

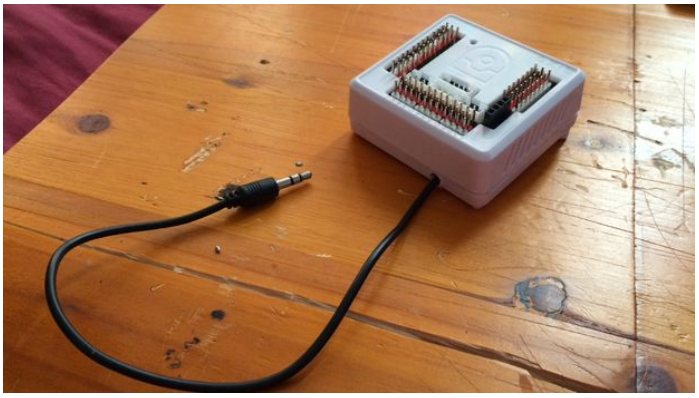
5) You will need to modify the casing a little so the speaker cable can fit through it (as seen in the photos).

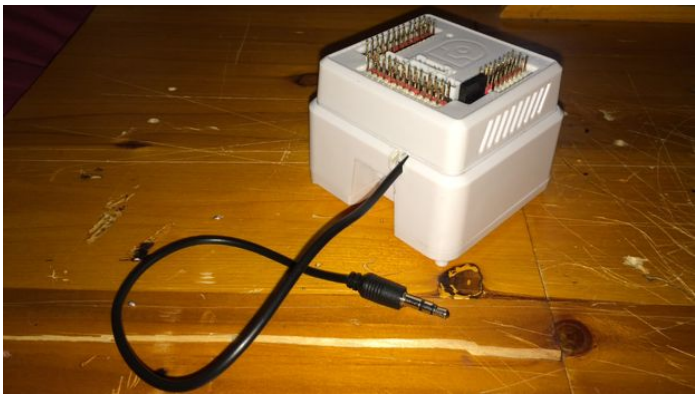
6) Power up the EZ-B, plug to headphone jack in to a speaker and test it. All should be working.





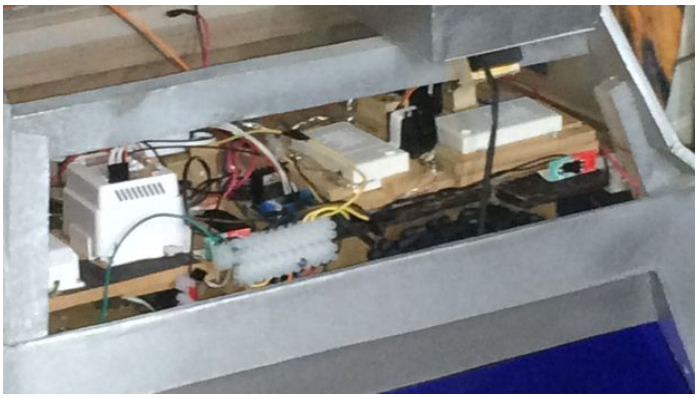






Step 19: Installing and connecting the EZ-B v4 Microcontroller.

Now things get even more interesting. To house and secure the EZ-B, I cut four short lengths of "L" shaped PVC wire trunking, attached them to the inside of the body on the platform, and seated the EZ-B. Then one by one I connected all of the servo cables, 6 pin camera ribbon cable, the 4 pin i2c ribbon cable for the camera analog smoke sensor wires, H-Bridge cables and ultrasonic sensor wires (this will be fitted soon). The wires from the remote relay switch was connected to the EZ-B's barrel jack and connected to the EZ-B. (I also added a power isolation switch between the battery and EZ-B wires, and between the battery and H-Bridge for maintenance and overall system shut down to save on battery life when K-9 wasn't used for long periods).



Step 20: Fitting the Dorsal Panel, Collar and Bumpers.

With the acrylic dorsal panel cut, shaped and protective film removed, I drilled some small holes towards the rear to form a small circle. This would act as a speaker grill. The I glued some strips of blue cardboard underneath the panel around the edges, with just a little overhang. Clear silicon adhering was applied all around, and the EL lighting wire was attached. The excess wire was fed through the hole in the neck ready to make the collar.

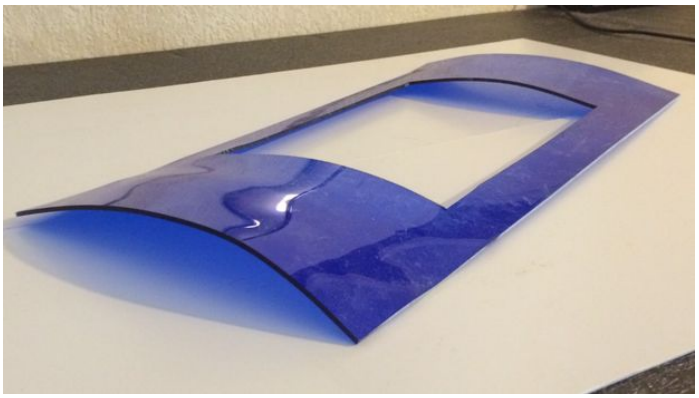
Using a small length of coloured material, I applied a thin line if silicon adhesive each side and laid the EL wire on top. I used clothes pegs to hole the EL wire straight while the silicon set. After a few hours I wrapped the collar around the neck and fastened it together, pulling any excess EL wire back in to the body.

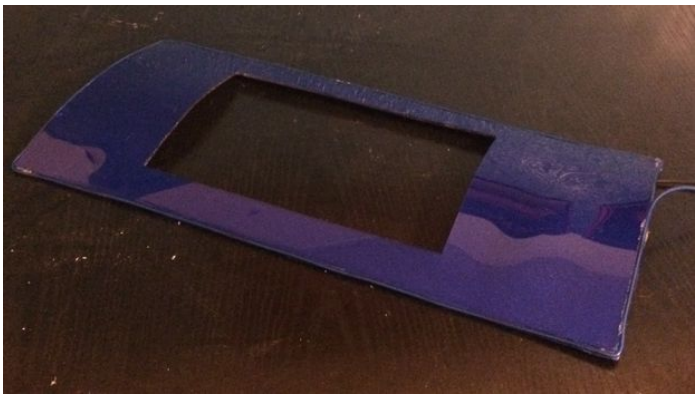
Next I attached a servo to the under side of the neck platform, glued in a rubber grommet to the servo horn, and attached the ultrasonic distant sensor. I fixed the neck covering in to place and connected the extension cables to the servo and sensor. I cut out a dog bone shape out of cardboard, covered it in aluminium tape, cut out two holes and fitted it to the sensor to create a dog tag.

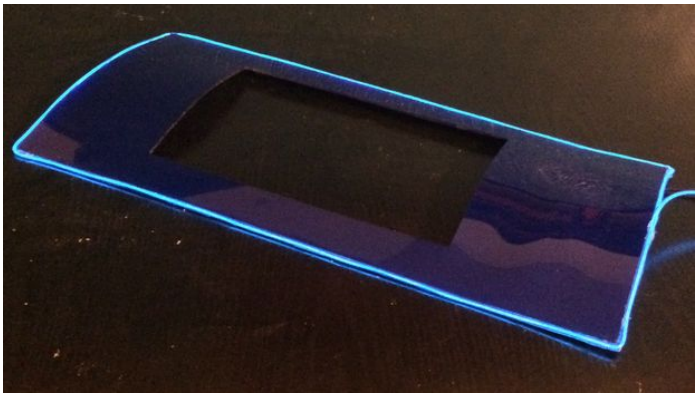
To fit the speaker, I cut out a peace of MDF and bored a hole in the centre. The speaker I used was a rechargeable Bluetooth speaker with a headphone socket. I opened up the speaker, fed the parts through the hole of the MDF, and put the speaker back together. Then I screwed the speaker panel to the top of the body frame supports towards the back of the body and connected the EZ-B speaker jack to the speaker. To finish this section off I placed the audio sensor/battery pack for the EL lighting for the mouth, under the speaker in a housing I made for it. Adjusting the sensitivity of the audio sensor to the correct position made it possible so the EL wire would only flash when K-9 spoke, and not flash due to background noise such as the drive motors.

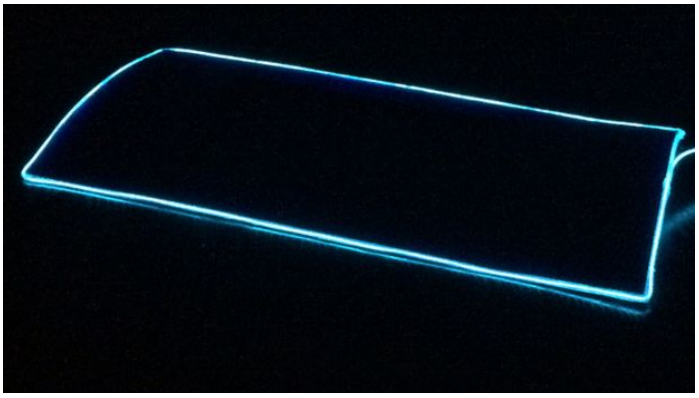
To attach the acrylic dorsal panel to the body, a piped some silicon adhesive to the top of the front and rear body panels, and fitted the acrylic panel. Then the control panel was lowered in to place to sit on top of the body frame supports, putting the battery compartment in to place. The cable of the control panel lights was long enough when the control panel is removed to change batteries of maintenance. Finally the rubber bumpers were fitted to the bottom of the acrylic panels.

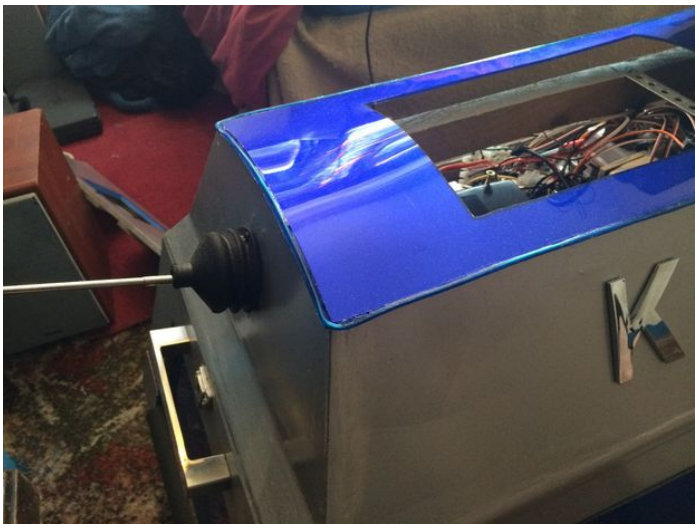
And that pretty much takes care of the build part of the project. K-9 was now cleaned up with a damp cloth, and his batteries were charged up ready for programming.

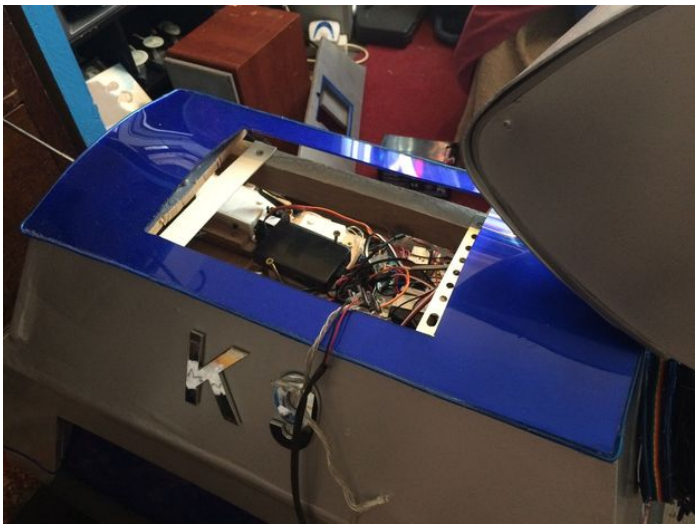


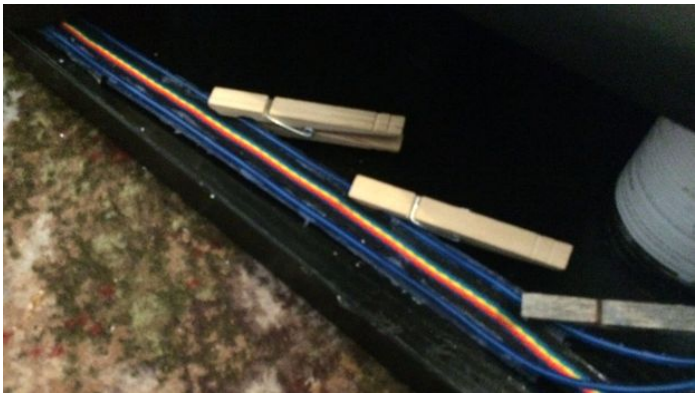


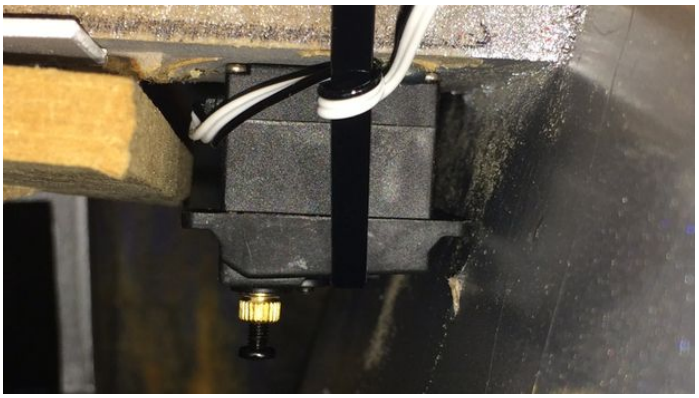


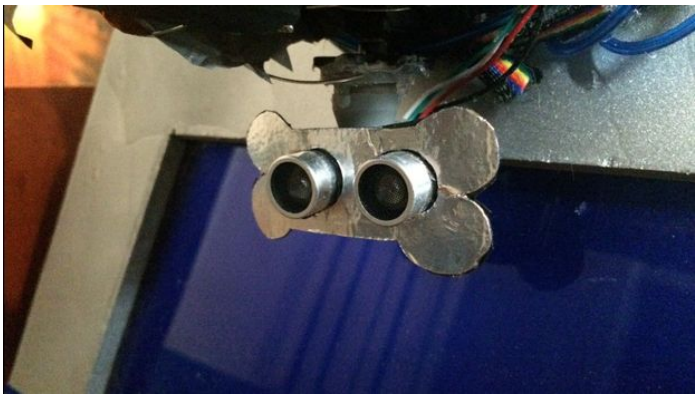


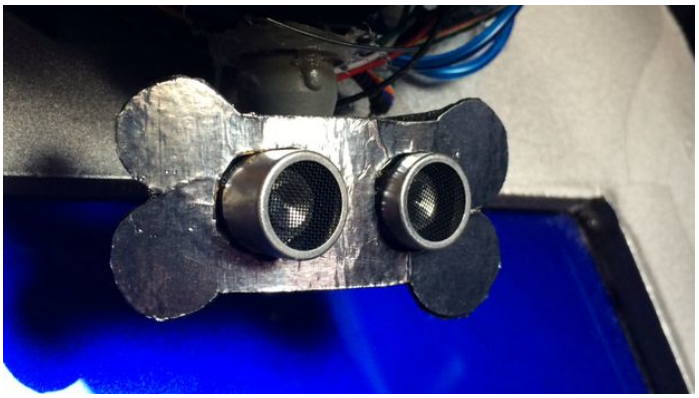


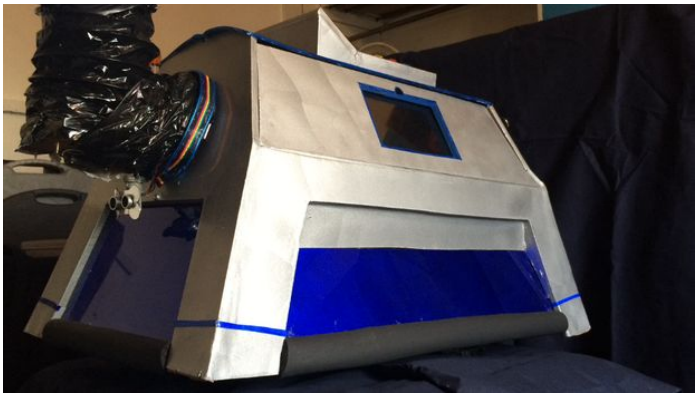












Step 21: Programming.

Now the REAL fun begins. For someone like myself to make K-9 2.0 do all the things that he can do, such as dynamic conversation, react to speech recognition, track colours, objects and peoples faces, and to give him autonomous abilities, these would have just stayed as ideas and dreams. But with the EZ-Robot software "EZ-Builder ", this made all the things mentioned, and more, possible. There was still a lot of figuring out and learning to do, but the way EZ-Builder software is made and available tutorials, as well as a friendly and knowledgeable forum community, these made things a lot easier.

I will supply a few basic scripts for different functions, and show some of K-9's main functions and controls used. I won't write many instructions on scripting and programming hear, as this would be a lot to read, maybe at times confusing, and may not be suitable for your own requirements. But all the information you need is available on EZ-Robots website, where K-9 is also featured on his own "Project Showcase". But if you wanted to have a look and even use some of K-9's functions, programming scripts, and controls, one of K-9's more basic project files is stored in EZ-Robots online project storage facility, the "EZ-Cloud". The project you want to search for is called "K-9 Project MK2". You will need the EZ-Builder software to access this, which is free to download.

So here are some of K-9 2.0's main functions.

Mobile Device Control.

EZ-Robot offers the ability to create you own mobile apps to control your robots, but I decided to go a different direction as I needed the shared WiFi connection to connect to K-9's EZ-Builder project and access the Pandorabot chatbot web server simultaneously, which the EZ-Builder mobiles apps doesn't support just yet. So what I did instead was to use a VPN PC remote application installed on my iPhone. This gives me the ability to view a mirror image of my PC's screen that runs EZ-Builder, as well as use the iPhone's touch screen to control the PC, thus controlling K-9. And using the iPhone's text to speech engine, I can use K-9's speech recognition control and the Pandorabot control. Because the iPhone's text to speech recognition engine is so reliable (at 98% accuracy), this allows myself, or anyone else to have a very reliable conversation with him.

Operating Lights.

With a 180 degree servo in place, a simple script to rotate the servo to 90 degree in either direction, press the flashing mode button and returns to centre position is all that is needed. This script combined with a soundboard sound effect, can the be used with the mobile, speech recognition, Pandorabot AIML code, joystick, or a random autonomous script control.

The servo that powers the lights On/Off switch can be controlled in the same manner. The only addition I used was when K-9 powers up which the activates a script to turn the lights on.

Here's a simple script to change the lighting phase (assuming servo is connected to port D13 on the EZ-B v4):

```
ControlCommand("Sound effects", Track_3)
servo(d13,90)

sleep(500)

Release(d13)
```

Controlling Drive System.

As the drive motors are controlled using a H-Bridge motor controller, a different kind of script is needed. This is dependent on how the H-Bridge is wired up to correctly select the correct motors to turn, the direction they turn, and the speed. The script is used in a script editor or a movement panel which can be accessed via joystick, voice rec, mobile controls, and of course autonomous programming. An added safety feature was added where the input power for the H-Bridge goes through one of the channels on the remote control power circuit. Here I would press button "A" on the key fob to power K-9's systems up. Once a wireless connection is established I would press button "B" on the fob to activate the H-Bridge. This is useful for maintenance too.

Here's a simple script to drive K-9 forwards (assuming H-Bridge is connected to ports D1, D2, D3, D4 on the EZ-B v4):

```
Set(D1, Off)

Set(D3, Off)

PWM(D0, 50)

PWM(D2, 50)
```

Head Movements.

Just like the lighting, the servos used to move the head up/down left/right needs some simple scripts that other controls can call on. The difference here is that the servo speed and range of movement need to be reduced as the head and neck have limited maximum travel limits.

A short script that turns the head right on a mobile control (assuming servo is connected to port D19 on the EZ-B)

```
if(getservo(d19)>10)
servodown(d19,7)

endif

sleep(100)
```

Rotating the Ears.

The ears have independent servos which can turn one at a time, or simultaneously either in opposite or the same directions. Here the speed and amount of times the ears rotate can be changed with different scrips. Two of the scripts I use the most are, one, for a standard quick rotation, and another slower speed turn with a radar sound effect when K-9 enters "Scan Mode"

Here's a simple script to rotate the ears (assuming servos are connected to ports D16, D17 on the EZ-B v4):

```
servo(d16,20)
```

servospeed(d16,1)

servo(d17,160)

servospeed(d17,1)

sleep(700)

servo(d16,160)

servospeed(d16,1)

servo(d17,20)

servospeed(d17,1)

sleep(1000)

servo(d16,20)

servospeed(d16,1)

servo(d17,160)

servospeed(d17,1)

sleep(1000)

servo(d16,90)

servospeed(d16,1)

servo(d17,90)

servospeed(d17,1)

sleep(1000)

Speech Recognition.

Most of K-9's functions are tied in to a speech recognition control with all the necessary command code scripts for all his functions. The speech recognition control uses Microsofts SAPI Speech Recognition engine, so some voice training on your Windows computer, as well as a good quality microphone or headset mic is needed to get more accurate results. I also have a very reliable secondary speech rec system that I mentioned above, using the iPhone's speech rec engine. But back to the main speech rec control. Every script and control in K-9's software project generates a "Control Command". These are essentially single line short cut scripts that access almost everything in the entire project. These command codes are entered in the the speech recognition editor, and the voice commands you want to use to execute the commands are entered exactly the same way you would say them. Multiple voice commands can be added to execute a single script. For example, you say "Hello K-9" and he responds with "Hello. What is your name?". But you add another command where you would say "Greetings K-9" and you would hear the same response. The speech recognition control is a very powerful and useful one.

A small speech recognition script to steer K-9 left

Phrase:

K9 turn left

Command:

Left(255,1200)

Speech Synthesis.

K-9 has a very clear English accent speaking voice with a slight robotic effect. The voice software was purchased from Cepstral, and the voices available have great pronunciation. This uses Microsofts SAPI Text to Speech engine, and is run through the EZ-Builder's "Speech synthesis" control. Here the voice volume, emphasis, and speech rate can be adjusted. The audio is then ported to the on-board speaker under K-9's dorsal panel. The set up is very simple, and the control is very reliable. People love to hear him speak.

A simple script command to here K-9 speak:

SayEZB("Hello. My name is K9, and I can talk.")

Media Player.

The software has two audio sound card controls. One plays audio through the EZ-B speaker (which I use for sound effects and recorded speech phrases), and the other through the PC's sound card. This means that I can connect my PC to my home stereo system via HDMI, WiFi or Bluetooth, and any audio played from the PC sound card control will be heard through my stereo systems speakers. I use a VCN iPhone app to control the audio via touch screen, but what I mostly use is the voice recognition, so can ask K-9 to play a certain music track by a certain artist, or ask for a random song to be played through my home audio system. And of course I can tell him to change, pause, or stop a track from playing. Great entertainment at party's. He does have the option to play videos as well using a similar method, with the difference being that I would use EZ-Builder to access my video files stored in my PC and play them through my TV.

Camera Tracking.

The on-board camera is high quality and can stream video feeds with no latency. Because of this, camera recognition is very good and extremely useful. The camera tracking control has a few functions that are used. These are...

- Factual recognition and tracking,
- Colour recognition and tracking,

- Object recognition and tracking (only basic shape objects can be recognised at the moment, but he is learning all the time so more complex shapes won't be far away),
- Movement recognition and tracking,
- And the ability to scan and recognise QR codes which can be used for navigation for example.

The video stream can be viewed on a PC, smartphone, tablet PC or smart TV. This can be done over a WiFi or cellular connection and can be viewed from almost anywhere in the world. And with the movement recognition control active, this is great for home security as K-9 can be programmed to take certain action's based on anything he see's that is out of the ordinary.

The camera tracking controls are linked to script commands which control his head servo movements, along with drive motor and steering controls. K-9 can also record video and take still photos when asked to do so, or autonomously based on what he see's.

Drive Controls.

The drive controls are mainly written in script form, as the control commands I mentioned earlier (the short cuts) for his drive system can be used in many controls such as speech recognition and touch screen control. As the drive motors are connected to a H-Bridge motor controller, scripts had to be written using values for PWM (pulse width modulation). This is what is used to control the speed at which the motors turn when K-9 is roaming around. Here's a quick example of the code used on the mobile interface. Press "Forward" button once and K-9 moves forwards. Press button again, K-9 stop's.

```
IF (!$pressed)
Set(D1, Off)

Set(D3, Off)

PWM(D0, 40)

PWM(D2, 40)

$pressed = 1 ELSE

Set(D1, Off)

Set(D3, Off)

PWM(D0, 0)

PWM(D2, 0)

$pressed = 0

ENDIF
```

Autonomous Personality.

K-9's autonomous abilities are managed by using a random personality generator and timing scripts which are tied in to the onboard sensors. The personality generator allows K-9 to randomly carry out security patrols of the house, tell a random joke or fun fact, or initiate a conversation. The timing scripts can initiate command codes for various controls, and take actions picked up on sensors such as the onboard smoke/gas detector.

A command code for telling jokes (assuming joke scripts are already programmed)...

```
ControlCommand("Script Manager", ScriptStart, "Random clean jokes")
```

And a script to alert of the detection of smoke or gas (assuming smoke sensor is connected to analog port ADC0)...

```
:loop
$smoke=getADC(adc0)

if($smoke>70)

sayEZB("Steve. My sensors have detected, the presents of smoke, or gas.")

endif

sleep(7000)

goto(loop)
```

Object Avoidance Detection.

The heart of the object detection and avoidance system is an ultrasonic distance sensor, a servo that rotates 180 degrees and a radar control within the EZ-Builder software. The ultrasonic can act as a motion detection sensor that can trigger command codes, or with the servo slowly sweeping from left to right and back, if any objects, such as walls or furniture are detected within a pre-defined range, a script in K-9's drive controls can stop the motors, reverse, and find a path around the object.

Line Following / Room Mapping.

Line following would have been an easy implementation to add, using an IR sensor and some program scripting. But the fact of the matter is, my house does not have lines painted all over the floors. But there was a better way for K-9 to follow lines if he, or myself wished, and the lines would not be visible to the human eye. This would be done by using a floor mapping control. To put it simply, I would get K-9 to move around a room and have him draw a virtual line. This line could the be used for K-9 to follow it once, or to be recorded and followed another time. The recording option also gives the ability to map a room, or number of rooms. Quite handy for when he roams around the house in autonomous mode.

Here is a short script that can be used in "Floor Mapping" (assuming the floor map control is active)

```
ControlCommand("Floor Map", Reset)
```

forward(150,4000)

sleep(2000)

right(100,2000)

sleep(7000)

forward(150,4000)

sleep(2000)

stop()

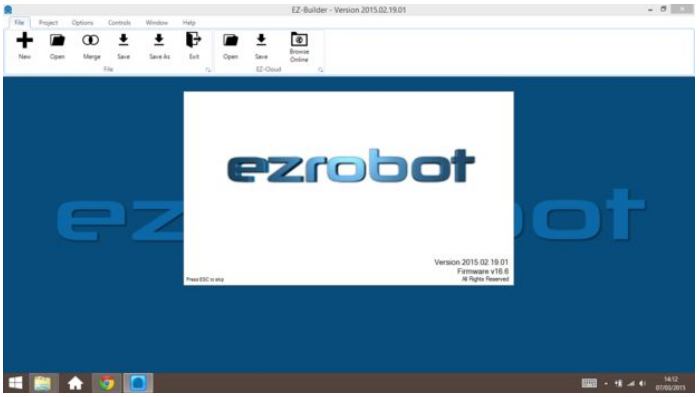
Phone Call and Email Notifications.

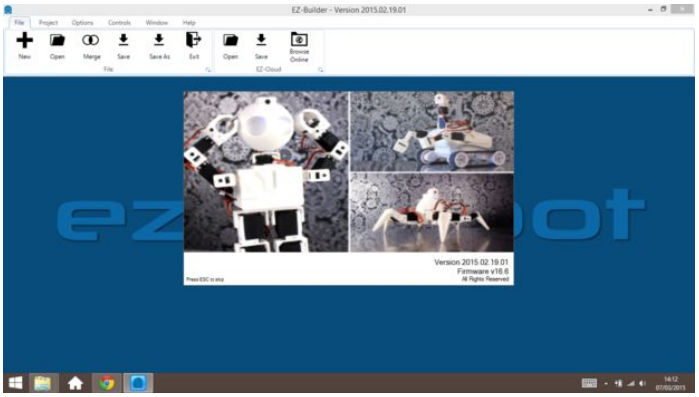
As the onboard speaker under the dorsal panel is Bluetooth enabled, connecting a phone to the speaker can allow K-9 to notify of oncoming phone calls, emails, text messages, FaceBook and Twitter alerts, alarms, notifications and event reminders. But K-9 can go one better my actually telling you who is calling or emailing you. To do this I used the audio editing software "Audacity" and recorded a list of sound files of K-9 using his speech synthesis function saying things such as "You have an incoming phone call from your friend, Richard" and "You have received a text message from your friend, Mark". This was cut and volume adjusted in Audacity with a standard tone overlaid. For example, when there's a phone call you would hear two seconds of a loud tone, the volume would drop 50% and then the voice would be heard. When the voice finishes the sentence the volume of the ringtone rises back to 100%.

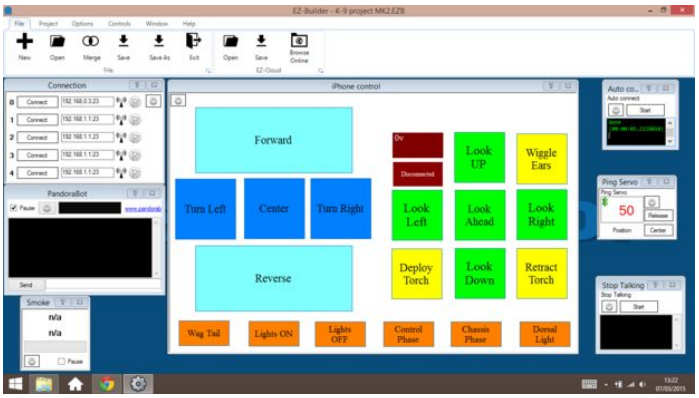
So, for the iPhone I use, once all of the alerts are created, edited and saved, they need to be converted in to ringtone files that the phone can use and saved to your iTunes library.

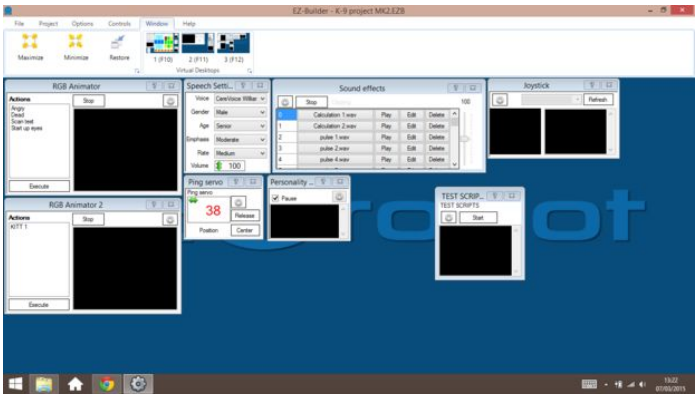
1. On your PC, launch iTunes.
2. From your library, right click on the song you want to use as a ringtone and then select "Get Info".
3. Select the Options tab and then tick the Start Time and Stop Time boxes. What I did in Audacity was to trim all the tones to 29 seconds so this next step wasn't required. Enter times at which you want the ringtone to start and stop. So just check if your tone is 30 seconds (as this is the Max limit) or less. If not, you need to listen to the ringtone first and note down the time you want it to start.
4. Click on "OK". Now right click on the track again, and then select "Create AAC version". iTunes will now convert the ringtone. It will appear as a duplicate track which you can identify it by "Track info" then "File type".
5. Right click on the ringtone and then using the Options tab from the "Get info" menu, untick the start and stop times to return them to their original times then click OK.
6. Now right click on your shorter ringtone track and click "Show in Windows Explorer". Double click the file to add it to the "Tones" section the iTunes library. **NOTE:** The steps outlined above need to be done one ringtone at a time which is a long process depending on how many files you created, but it is a one time thing
7. Delete the original ringtone from the library as it will no longer play.
8. Finally connect your iPhone and upload the new ringtones to the phone, select the correct tone for each of your contacts, alarms or notifications. Connect your phone to the speaker and every time your phone alerts you, K-9 will tell what that alert is.

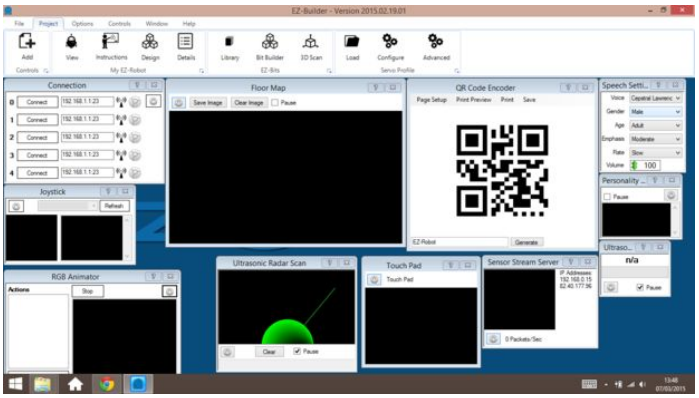
There are many more functions and features that K-9 2.0 has, but it's just too many to list here without giving information overload. But you should get the idea how versatile, useful and unique K-9 is, as a lot of the robots that are around could only dream of doing half of what K-9 is capable of and how well he does it. But I'll say this, if I managed to do it, you can too. The controls in EZ-Builder do make things easy, but the more work you put in to scripting, programming, and learning how to get the most out of all the available controls, the better and more impressive you robot will be. And the same goes with the build itself. The more time you take, and the willingness to learn new skills, the better the final result will be.

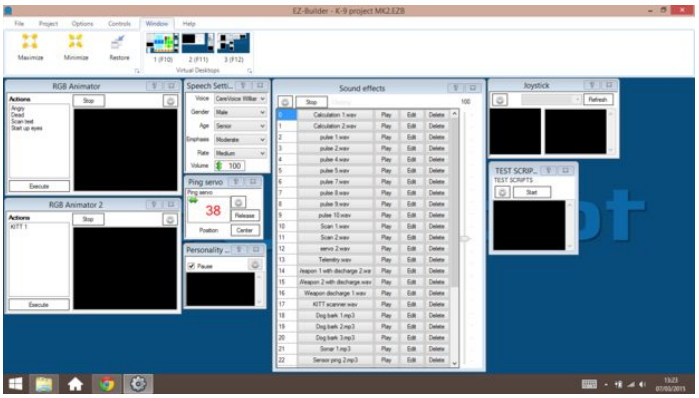


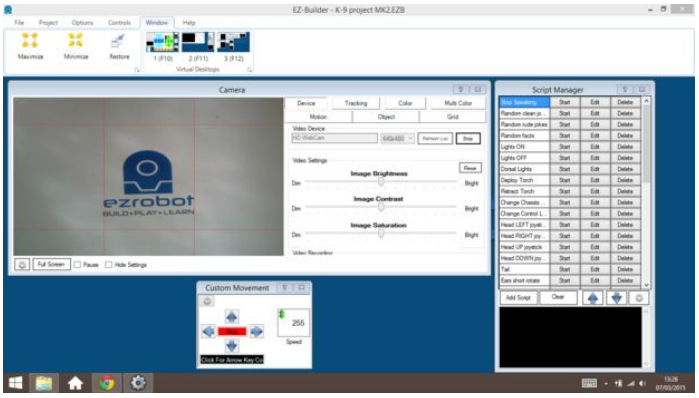


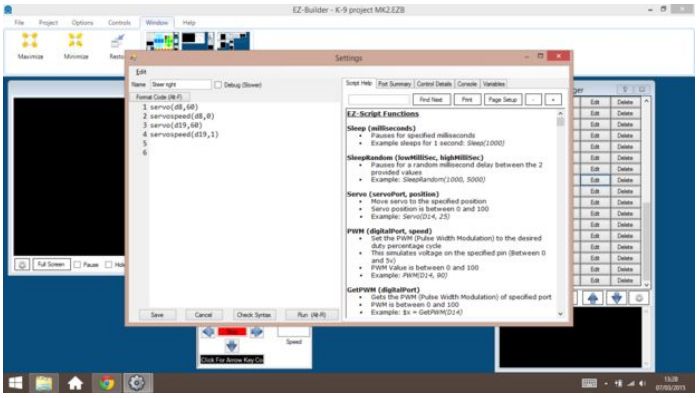


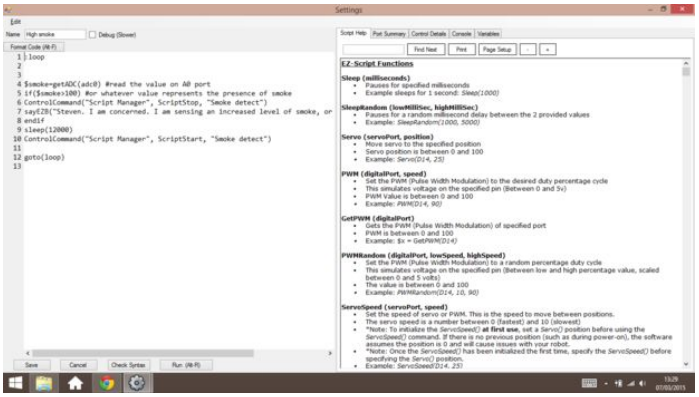


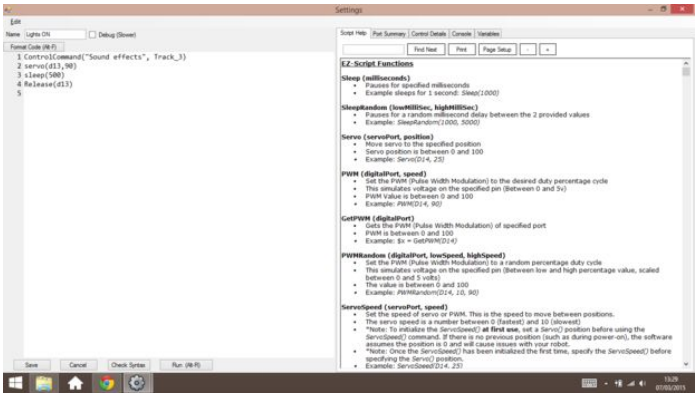

EZ-Builder - Version 2015.02.19.01
Connection
Floor Map
QR Code Encoder
Speech Setti...
Joystick
RGB Animator
Ultrasonic Radar Scan
Touch Pad
Sensor Stream Server
Personality ...
Ultraso...
n/a

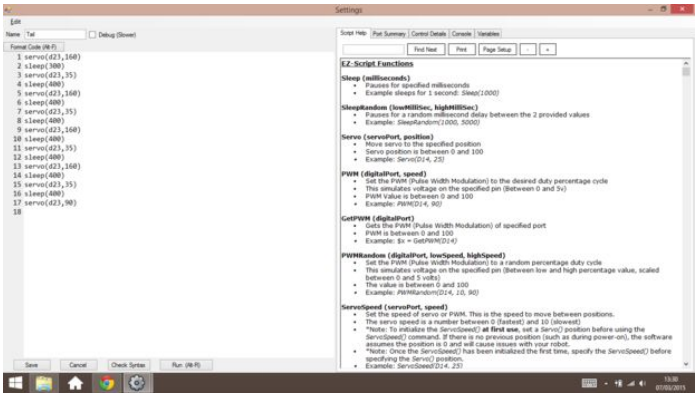

EZ-Builder - K-9 project MK2.EZB
RGB Animator
RGB Animator 2
Sound effects
Joystick
Speech Setti...
Ping servo
38
Personality ...
TEST SCRIP...

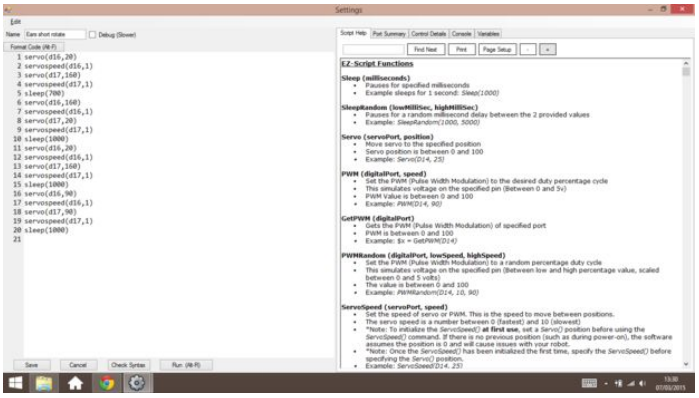

EZ-Builder - K-9 project MK2.EZB
Camera
Script Manager
ezrobot
BUILD • PLAY • LEARN
Image Brightness
Image Contrast
Image Saturation
Custom Movement
255
Speed

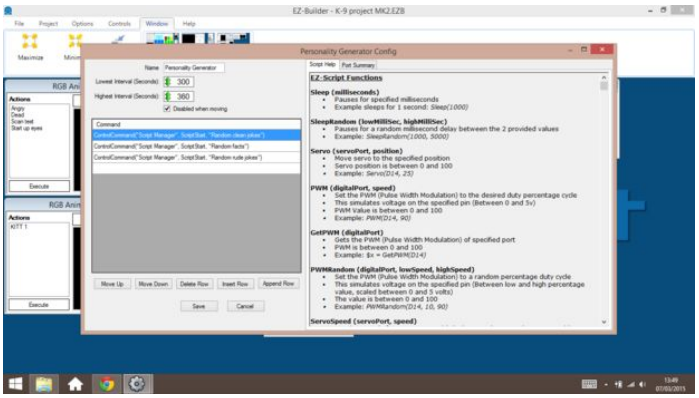

EZ-Builder - K-9 project MK2.EZB
Settings
1 servo(d8,60)
2 servospeed(d8,8)
3 servo(d19,60)
4 servospeed(d19,1)
EZ-Script Functions
Sleep (milliseconds)
SleepRandom (lowMilliSec, highMilliSec)
Servo (servoPort, position)
PWM (digitalPort, speed)
GetPWM (digitalPort)
Save
Cancel
Check Syntax
Run (Alt-R)

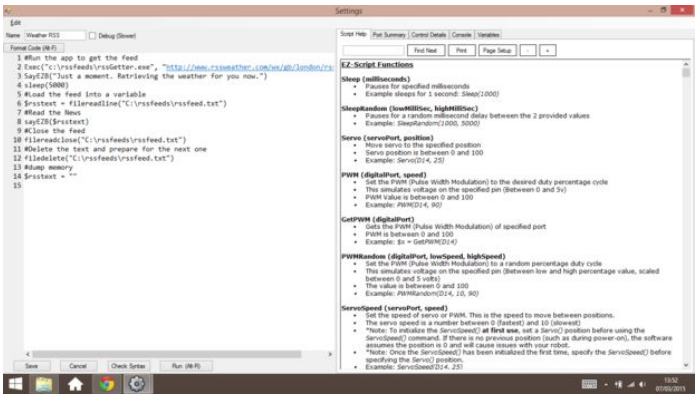

Settings
1 loop
2
3
4 $smoke=getADC(adc0) #read the value on A0 port
5 if($smoke>100) #or whatever value represents the presence of smoke
6 ControlCommand("Script Manager", ScriptStop, "Smoke detect")
7 sayEZB("Steven. I am concerned. I am sensing an increased level of smoke, or
8 endif
9 sleep(12000)
10 ControlCommand("Script Manager", ScriptStart, "Smoke detect")
11
12 goto(loop)
13
EZ-Script Functions
Sleep (milliseconds)
SleepRandom (lowMilliSec, highMilliSec)
Servo (servoPort, position)
PWM (digitalPort, speed)
GetPWM (digitalPort)
PWMRandom (digitalPort, lowSpeed, highSpeed)
ServoSpeed (servoPort, speed)
Save
Cancel
Check Syntax
Run (Alt-R)

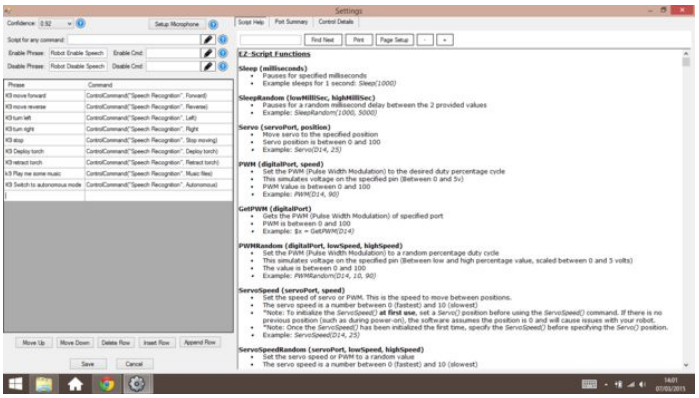

Settings
1 ControlCommand("Sound effects", Track_3)
2 servo(d13,90)
3 sleep(500)
4 Release(d13)
5
EZ-Script Functions
Sleep (milliseconds)
SleepRandom (lowMilliSec, highMilliSec)
Servo (servoPort, position)
PWM (digitalPort, speed)
GetPWM (digitalPort)
PWMRandom (digitalPort, lowSpeed, highSpeed)
ServoSpeed (servoPort, speed)
Save
Cancel
Check Syntax
Run (Alt-R)


Settings
1 servo(d23,160)
2 sleep(300)
3 servo(d23,35)
4 sleep(400)
5 servo(d23,160)
6 sleep(400)
7 servo(d23,35)
8 sleep(400)
9 servo(d23,160)
10 sleep(400)
11 servo(d23,35)
12 sleep(400)
13 servo(d23,160)
14 sleep(400)
15 servo(d23,35)
16 sleep(400)
17 servo(d23,90)
18
EZ-Script Functions
Sleep (milliseconds)
SleepRandom (lowMilliSec, highMilliSec)
Servo (servoPort, position)
PWM (digitalPort, speed)
GetPWM (digitalPort)
PWMRandom (digitalPort, lowSpeed, highSpeed)
ServoSpeed (servoPort, speed)
Save
Cancel
Check Syntax
Run (Alt-R)


Settings
1 servo(d16,20)
2 servospeed(d16,1)
3 servo(d17,160)
4 servospeed(d17,1)
5 sleep(700)
6 servo(d16,160)
7 servospeed(d16,1)
8 servo(d17,20)
9 servospeed(d17,1)
10 sleep(1000)
11 servo(d16,20)
12 servospeed(d16,1)
13 servo(d17,160)
14 servospeed(d17,1)
15 sleep(1000)
16 servo(d16,90)
17 servospeed(d16,1)
18 servo(d17,90)
19 servospeed(d17,1)
20 sleep(1000)
21
EZ-Script Functions
Sleep (milliseconds)
SleepRandom (lowMilliSec, highMilliSec)
Servo (servoPort, position)
PWM (digitalPort, speed)
GetPWM (digitalPort)
PWMRandom (digitalPort, lowSpeed, highSpeed)
ServoSpeed (servoPort, speed)
Save
Cancel
Check Syntax
Run (Alt-R)







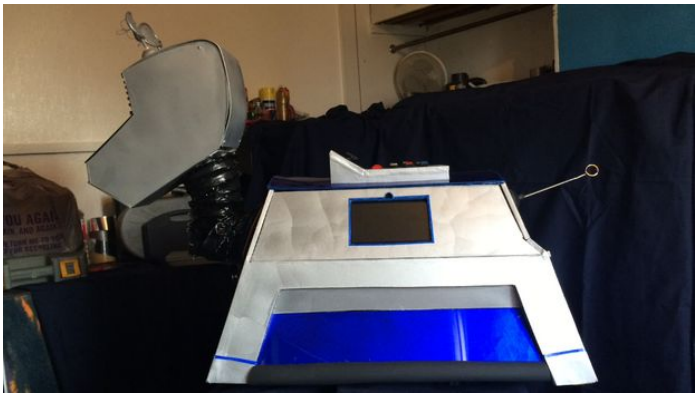
Step 22: Final Thought's, Photos and Videos.

So I'm about finished here. I just want to leave you with a few final thoughts, a few more pictures, and a couple more fun videos for your viewing pleasure. I just want to mention that I do not have any affiliation or work for any of the companies I have supplied product links to, they are included to hopefully help you out if you wanted to use some of my ideas. I also supplied links for EZ-Robot because it is a fantastic robot platform, they are a great company to deal with, and sell quality products as well as offering lots of friendly support. I am so pleased I found them. Like I said at the start, K-9 would have turned out very differently, and certainly not for the better. If you're looking to head out in to the world of robot building, or you're a seasoned robot builder, or indeed if you want to try your hand at building your very own K-9 unit, you should check them out because you won't regret it.

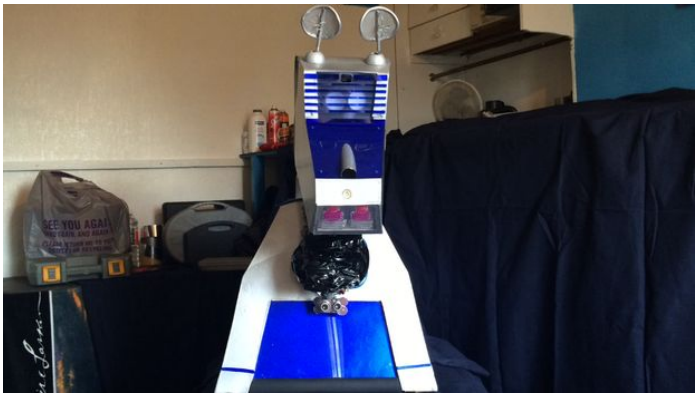
So it's time to sign off now. I hope you found some of the information and pictures provided in this Instructable useful and inspiring, and would love to hear any comments or suggestions you may have. And of course, if you do decide to build you own K-9 unit, I would love to see your work. I am currently working on a new robot project, working name "Project Vic", which I will do an Instructable for, which should be posted in a couple of months time, if your interested that is. So anyway, from myself, and K-9 of course, thanks for reading, and happy building. :)

S G :)

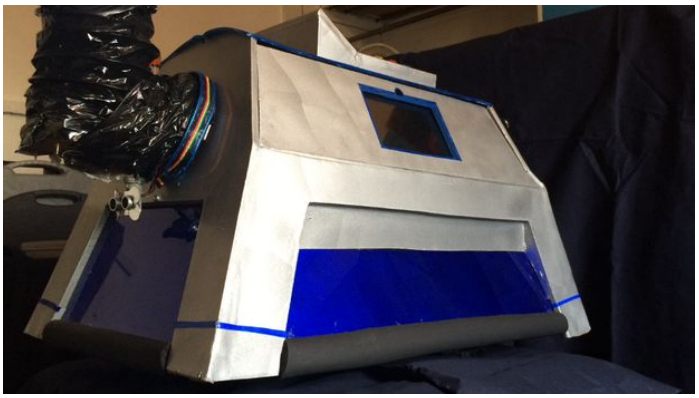

K 9

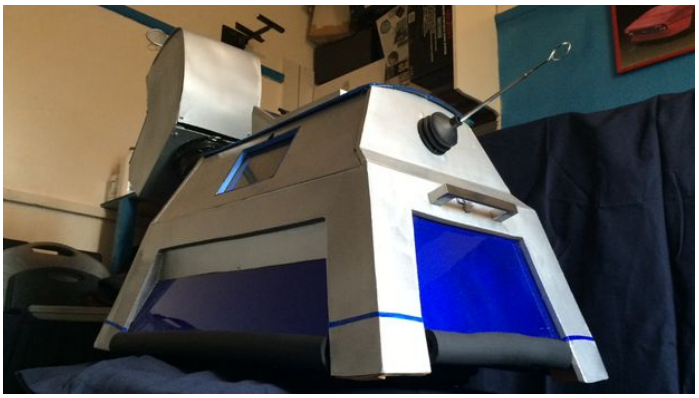



K9

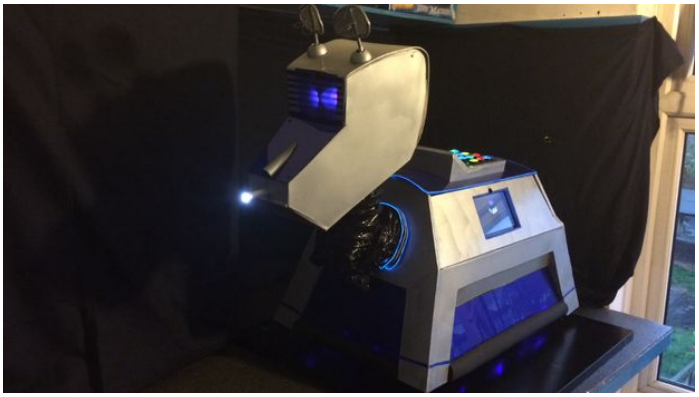


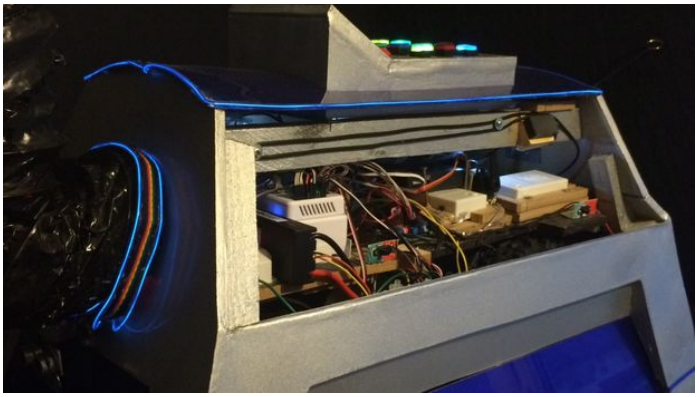

K9






K9





Related Instructables

DIY Mod an Omnibot 80's Robot with Voice, Camera, Servos, Bluetooth by djsures

EZ-Robot Wall.E (Photos) by Djandco

My Autonomous HomeMade Wall-E Robot (Photos) by djsures

Make a scary scarab robot by djsures

BASIC TUTORIAL ON ARTIFICIAL INTELLIGENCE MARKUP LANGUAGE

How to enter the iRobot Create Challenge by jeffreyf

(AIML) VIA NOTEPAD by theseventhsage

Comments

2 comments **Add Comment**

tomatoskins says: Mar 9, 2015. 8:33 AM **REPLY**

This is awesome! The level of detail here is unparalleled. I would suggest changing the order of your first two videos. How the instructable is formatted you need to click the drop down menu to see all your images in the into and that second video of the demonstaration is amazing!

steve-gibbs5 says: Mar 9, 2015. 9:40 AM **REPLY**

@tomatoskins.

Thank you for your kind words, and I'm pleased you like the (not so) little fella. I couldn't "not" take time with the detail, as K-9 was a labour of love and thought he deserved the extra effort. Glad you enjoyed the videos too. Thanks for the "heads up" on the picture structure. On your recommendation, I changed it about a bit and I think it looks a little better now. :)