EyesBot Driver
RobotShop App
Assembly and Use

EyesBot Hardware Specification

DISCLAIMER

The contents of this document are believed to be correct, but the author, copyright holder, heirs, and anyone else associated with creating or distributing this document disclaim any responsibility for any injury, consequential or incidental damage associated with its use. This document and any accompanying code are distributed without any warrantee of any kind, expressed or implied.

Working with electronics you may injure yourself or others and damage or destroy property. You do this at your own risk. The author of this document, the distributors of this document, EyesBot, LLC, its owners, employees and heirs take no responsibility for your use of this document

1 Getting Ready

This document describes how to create a robot that will hold an iPod or iPhone and permit the iOS device to talk with an Arduino which will control motors using a motor shield. The robot is controlled by means of a web interface which is provided by the EyesBot Driver app, which is described in section 2, below.

1.1 *Parts*

Part Name	RobotShop Part Number	Quantity	Comments
DFRobot Leonardo Microcontroller with Xbee socket	RB-Dfr-217	1	
DFRobot Arduino Compatible Motor Shield (1A)	RB-Dfr-04	1	
170 Tie Point Mini Breadboard	RB-Spa-139	1	
GM9 - Gear Motor 9 - 90 Degree Shaft	RB-Sbo-07	2	
Dagu Caster 20 mm	RB-Dag-22	1	
Plastic Wheel (Black)	RB-Sbo-59	2	
Light Sensor Mini Photocell (called photodetectors in most places in the document)	RB-Dfr-317	8	Only 5 are used for communication in the current version of EyesBot Driver
Optional Parts			
Jumpers	RB-Dfr-353 or RB-Dfr-133	1	Not included in kit (since most hobbyists have some jumper wire on hand)
2n3904 Transistor	RB-Dfr-236	1	Needed for headlights, but not included in kit as many hobbyists have a low power NPN transistor on hand (any that will handle 100 mA or so will do).
68 Ohm Resistors	RB-Ibo-94	2	Needed for headlights, but not included in kit as many hobbyists have some resistors on hand. The actual value of the resistor will depend on what the parameters of

			your LED are (see its datasheet to determine the resistor to use).
1K Ohm Resistor		1	Needed for headlights, but not included in kit as many hobbyists have some resistors on hand. This is the transistor base resistor.
White LED		2	Needed for headlights. Not included in kit but available from many sources. Check the datasheet as you may need to adjust the 68 Ohm resistor value.
4xAA Square Battery Holder with Cover	RB-Dfr-313	1	Not included in the kit. Any holder will do and you can substitute a 7.4 V Lipo for the 4 AA batteries.
Cardboard/ABS Plastic		About 20cm x 10cm piece about 6mm deep	For robot body and iPod holder. These instructions assume you will use cardboard. If you use ABS plastic, it will result in a much more robust robot
Solder		Small amount	Needed for headlights and motor connections
Glue		Small amount	Needed to attach motors to body and assemble iPod holder
iPod		1	With EyesBot Driver app, as discussed below
Clear adhesive tape			A small piece of wide, clear, adhesive tape is needed to keep the photodetectors in place
Black electrical tape			This is for screening out light, and, since the tape will be in contact with the leads of the photodetector, it must be electrical tape. Duct tape can be used for any non-electrical part.

1.2 ***Tools***

Razor knife or scissors – This will be used to cut out cardboard pieces

Soldering iron – For connecting leads to the motor and soldering the components that make up the headlights.

Tweezers – To position the photodetectors and place the jumpers. In a sense this is optional, but it is extremely difficult without tweezers.

2 EyesBot Driver App

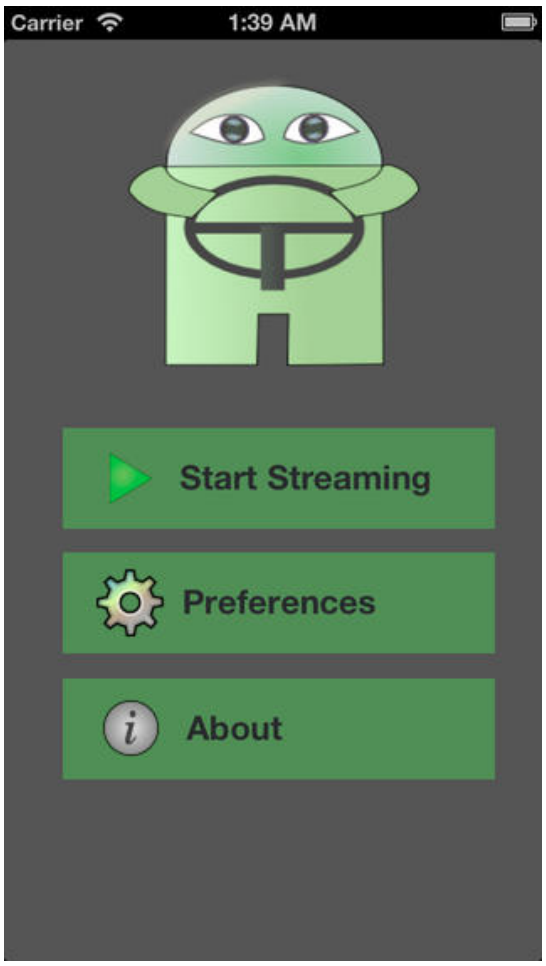
The brains of the robot will be an iPod or iPhone running the EyesBot Driver app. Please download it prior to starting to put the robot together. The app can be downloaded from the app store at:

https://itunes.apple.com/us/app/eyesbot-driver/id654272975?ls=1&mt=8

After you download EyesBot Driver to your iPod or iPhone, please start it up.

2.1 *Startup screen*

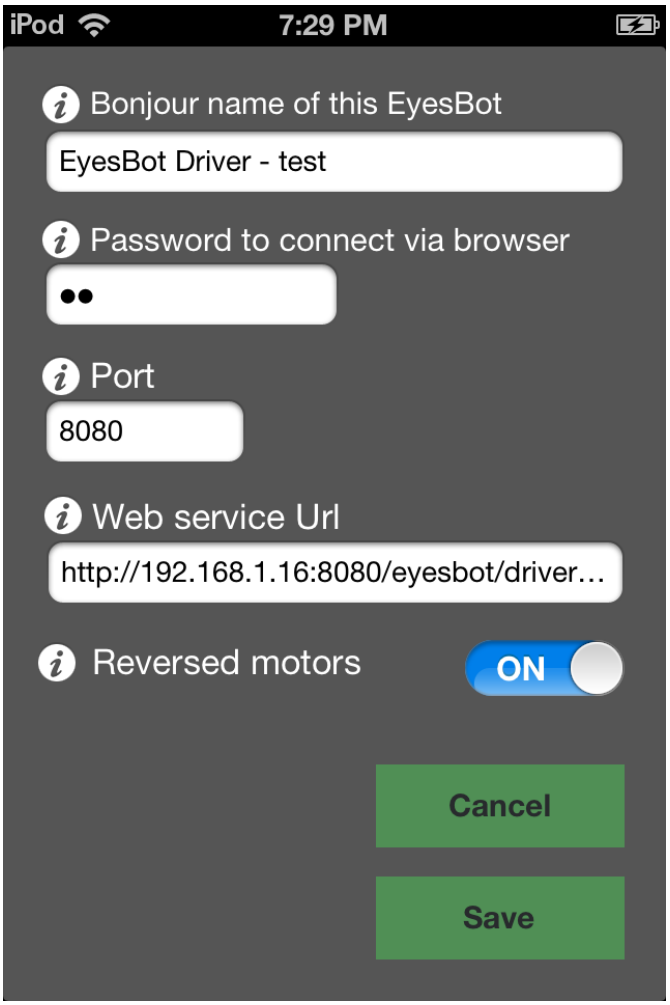
The first screen that you will see has three buttons. The two of interest for our purposes are "Start Streaming" and "Preferences". "Start Streaming" will start the server on your iOS device that streams video and accepts commands from the EyesBot Driver web interface. The first button you should click is "Preferences".



2.2 *Preferences screen*

The preferences screen, shown below, lets you change the Bonjour name of the instance of EyesBot Driver, which is only useful if you are using a Bonjour browser, and also set the password required to connect and the port number on which the server runs. If you leave the password blank anyone with access to your network can connect to EyesBot Driver, so be sure to set it. The port number defaults to 8080, but you can change it to anything that will work on your network. The web services URL is discussed in this blog post: http://eyesbot.com/blog?preload=eyesbot_web_services.txt , but isn't necessary to build this robot.

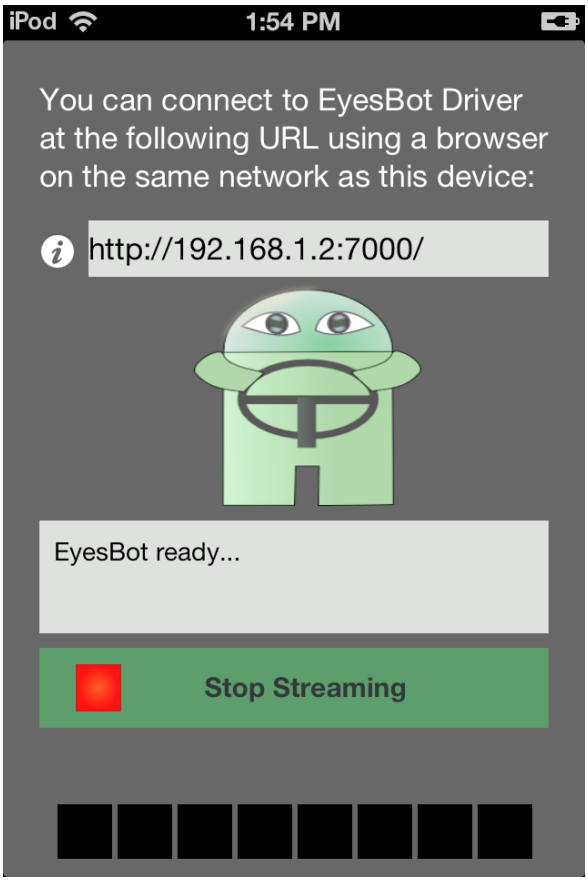
The way that the motors are deployed on this robot is "reversed", that is, the back end of the motor is pointing forward, so you will need to turn the reversed motors switch on or turning left and right will be reversed.



2.3 *Streaming screen*

You arrive at the streaming screen by clicking the "Start Streaming" button. The key points on this screen are:

- The address at the top, which you will need to type into your browser in order to connect to EyesBot Driver
- The gray area which the server uses to display state changes, such as someone logging in.
- Stop streaming, which stops the server and takes you back to the startup screen.
- The black squares along the bottom which you will align with some photo-detectors when creating the robot body so that the iOS device can communicate with an Arduino



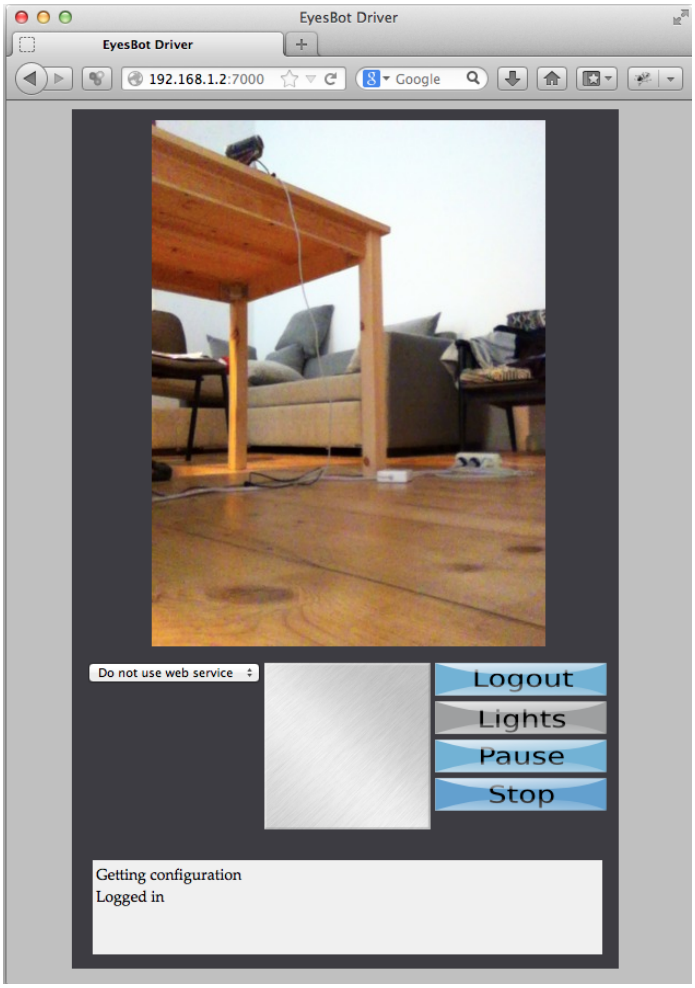
2.4 *Web interface*

When you type the address from the streaming screen into a browser which is on the same network as the iOS device, you will see the following interface:

If you don't see this interface, you likely have a firewall between the system that you have the browser running on and the iOS device, or you may have typed the address in wrong.

To log in, type in the password you set in the preferences tab. If you didn't set a password, no password will be requested.

After you have logged in successfully, you will see a screen showing what the iOS device is pointed at. To see how the robot talks with an Arduino, click and drag around on the "trackpad" in the center below the video image. This will cause the lights on the streaming screen to change, which will be read by the robot body that is discussed later in this document.

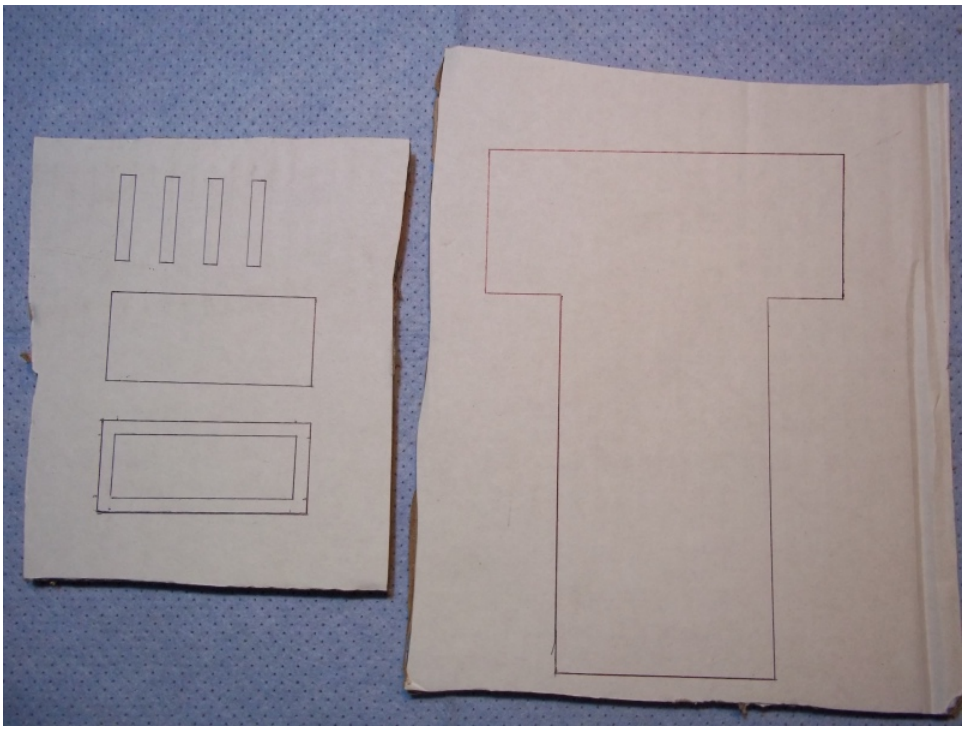


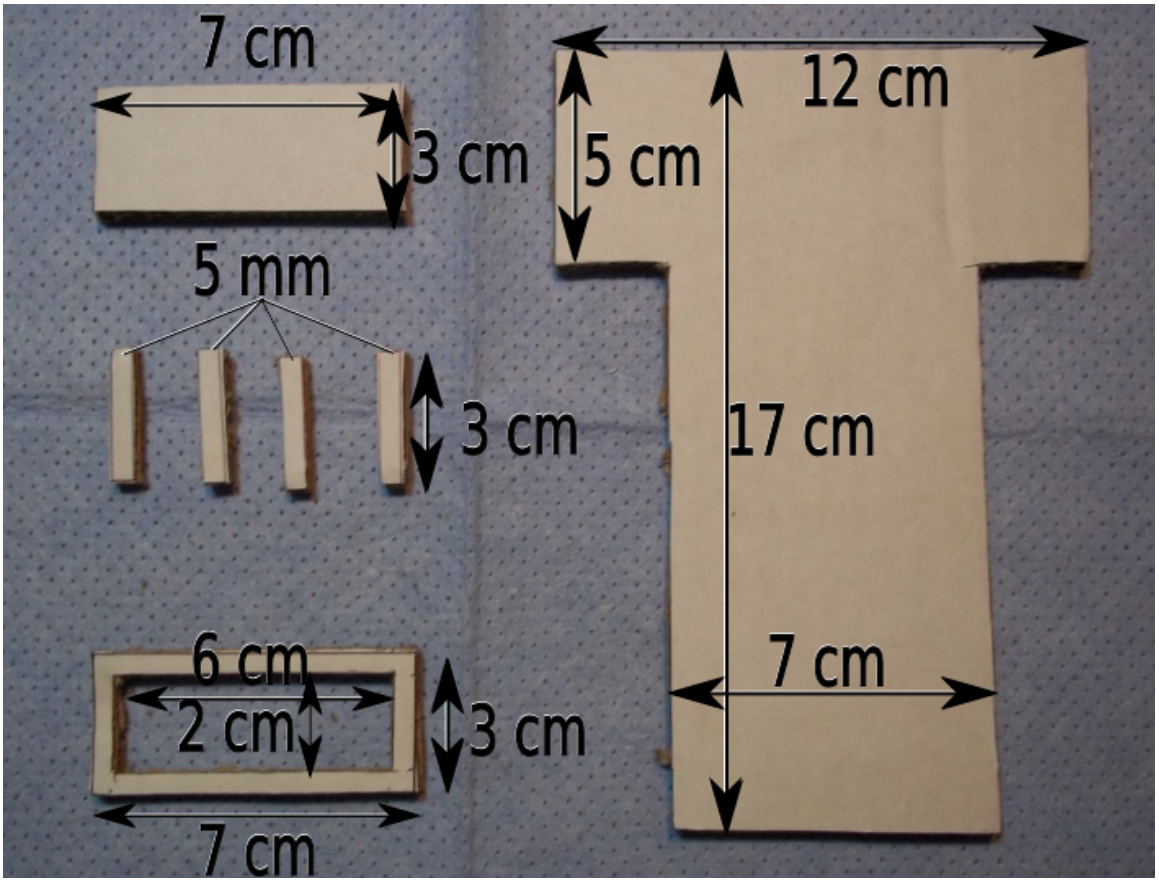
3 Platform and iPod/iPhone Holder

In this long section, you will create a robot. The directions assume you will be using cardboard, but you can substitute ABS plastic if you have the necessary tools and want a sturdier robot.

3.1 *Template*

Start out by selecting some sturdy cardboard. It best to use cardboard that is about 6 mm thick, as it is easy to compress cardboard if you need to compress it to create a snug fit with your iOS device, but it becomes weak if it needs to be stretched to fit your iOS device. When you have the cardboard, trace out the outlines of the parts you need on it. You will be cutting this out with a razor knife or sharp scissors. The second image has the dimensions.

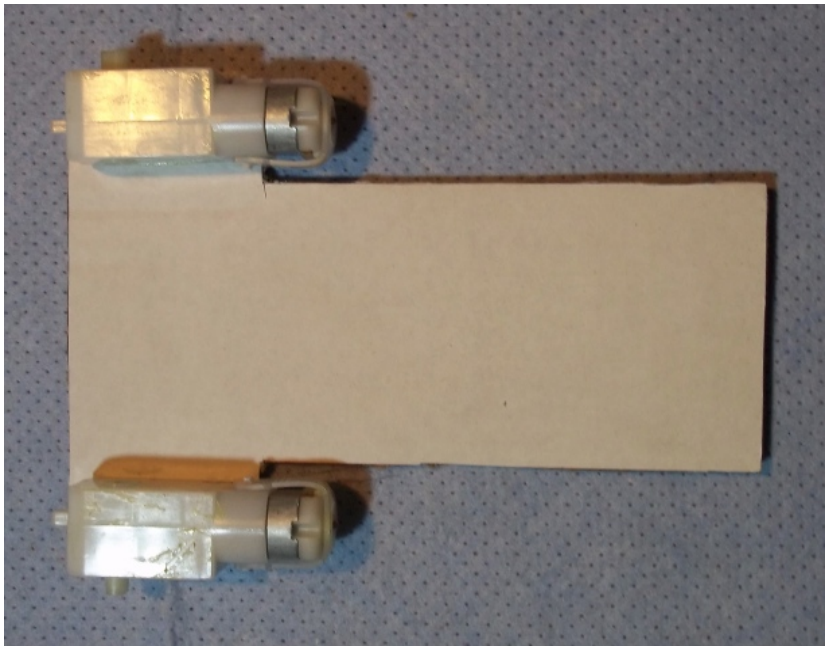



7 cm
3 cm
5 mm
3 cm
6 cm
2 cm
3 cm
7 cm
12 cm
5 cm
17 cm
7 cm

The large piece of cardboard serves as the robot platform, the other pieces will be used to create the iPod holder.

3.2 *Attach Motors*

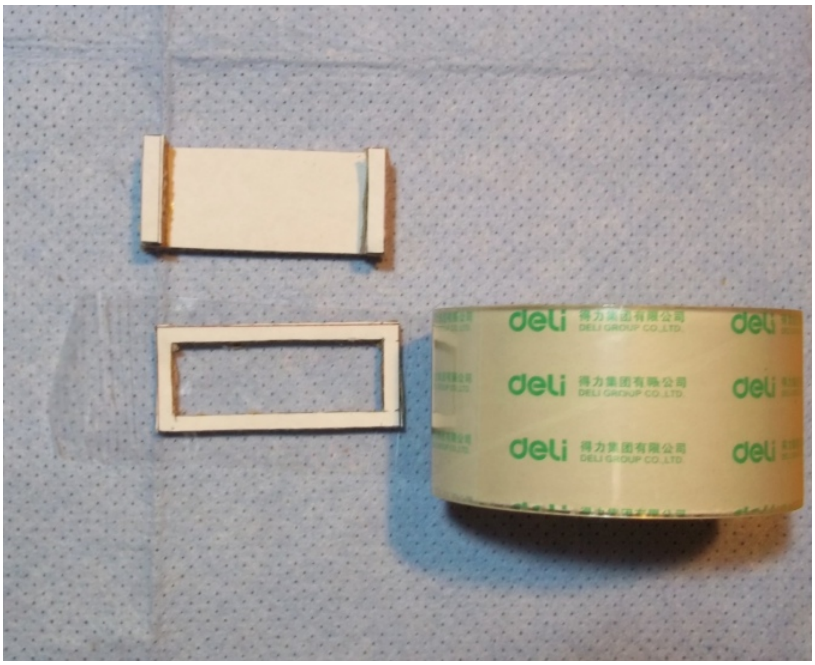
Next, attach the motor as shown below. If you want to attach them permanently, use glue, otherwise double sided sticky tape will permit easier removal.



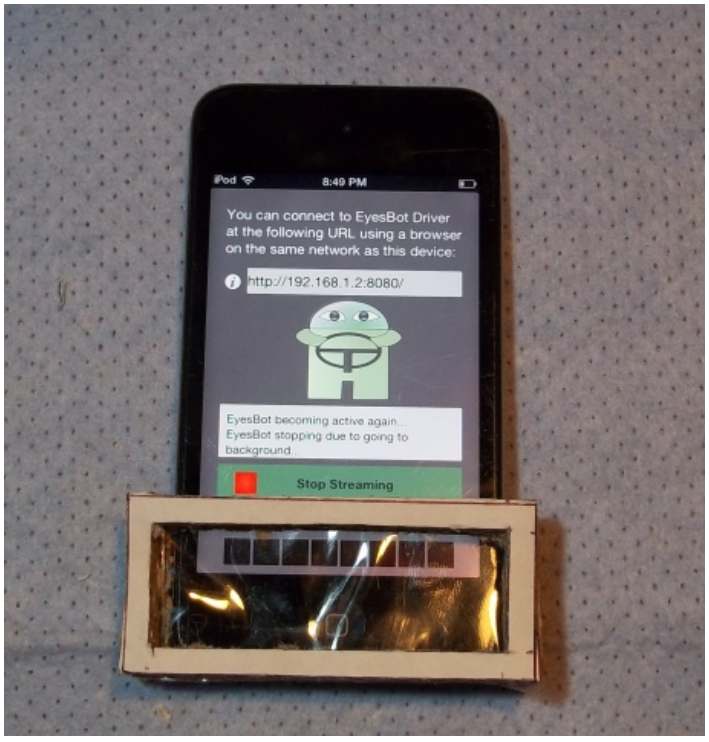
3.3 *Assemble iPod/iPhone Holder*

A simple way to create a holder out of cardboard, transparent tape, and glue is shown in this section. If you want to make the holder out of a plastic like ABS, the instructions are essentially the same, but you will obviously need a router to cut the plastic. The only tool you need for this section is a pair of scissors or a razor knife. If you use a razor knife, please use all necessary safety equipment.

First, take the solid piece of cardboard that is 7cm X 3cm and glue the four small cardboard spacers onto it. You want the spacers to add at least 7 mm of depth for an iPd (use 6 pieces, 3 on each side, if you are using an iPhone, since and iPhone is 7.6mm for an iPhone 5 and 9.3mm for an iPhone 4). Also take the cardboard piece with the center cut out and put transparent tape on it, with the tape facing into the cutout area so that it adheres to the square piece that has its center cut out. Photo shown below



The next step is to put the front on the holder. Glue the front in place as shown. Note that the excess tape has been removed and that the sticky side of the tape is facing away from the iPod. The holder should have a little excess depth that you can remove at this point by compressing the cardboard until the iPod fits snuggly. This will help to keep out ambient light.

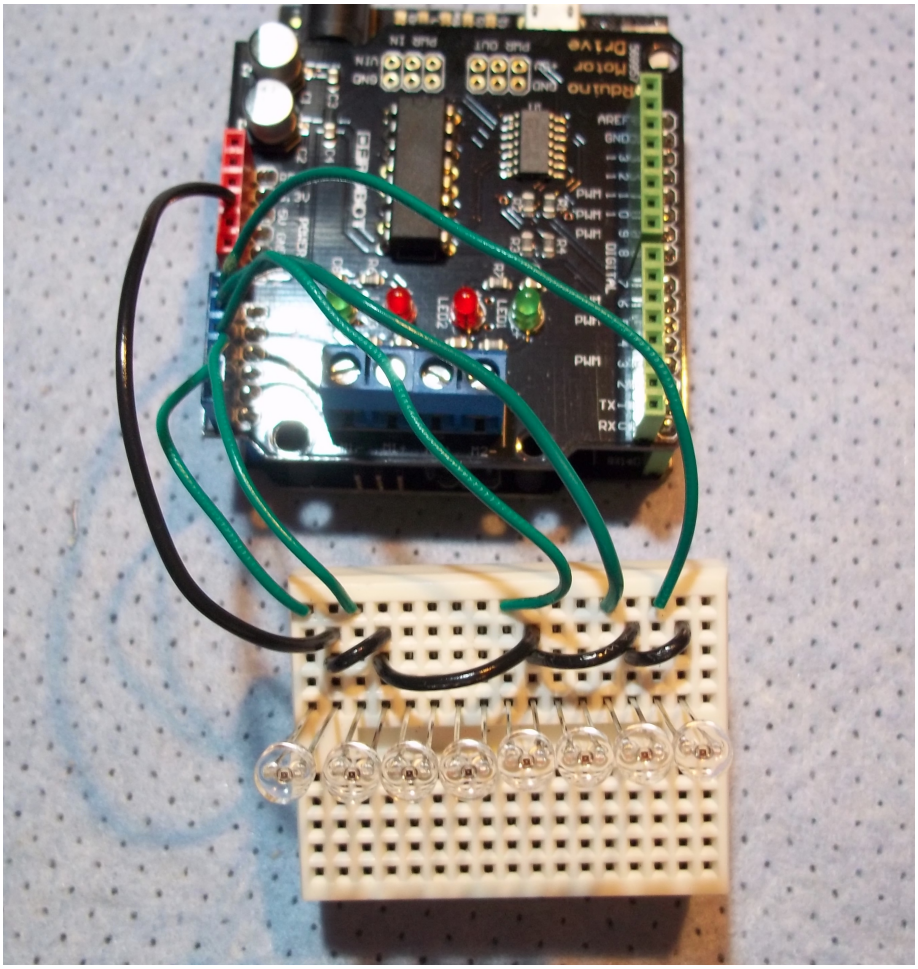


3.4 *Assemble Arduino and shield, put photodetectors and jumpers into breadboard*

First, snap the motor shield into place on the Leonardo board.

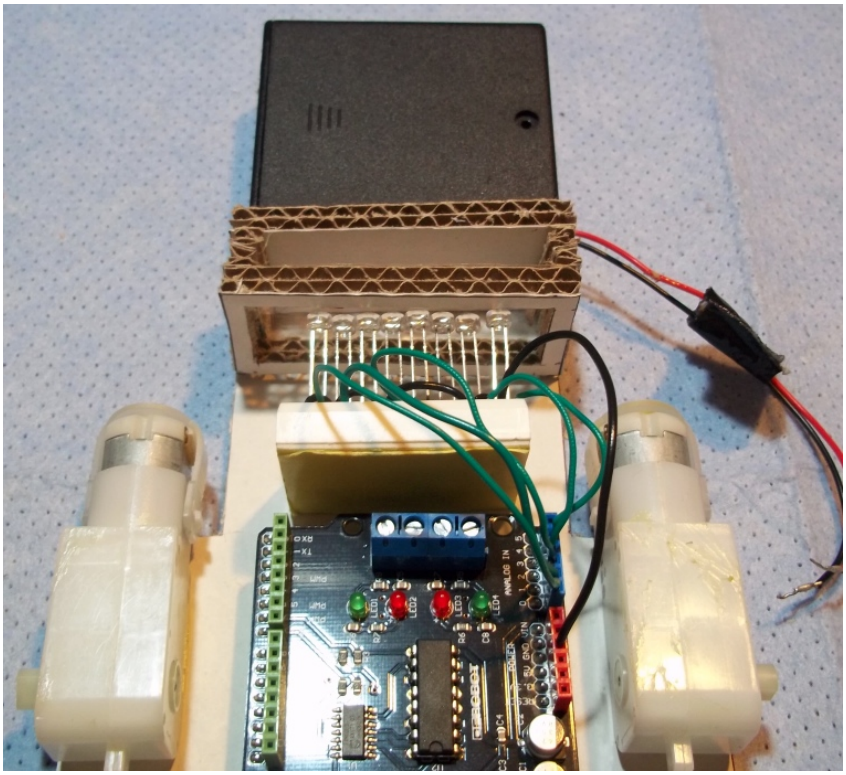
Next, put the breadboard facing you and place the eight photo detectors into the lowest hole on the top part of the breadboard (this leave the rightmost connection point on this row empty). Place the longer leg, the anode, on the left with the board facing you. Next, cut some jumper wire to size (the black wire in the photo) and daisychain the jumpers so that all of them will connect the cathodes together. The leftmost black jumper should connect to one of the ground sockets on the motor shield.

Finally, connect the rightmost photodetector anode to the analog 4 socket, and the other detector anodes to analog 3 through 0, going from left to right with the board facing you. These connections are shown with the green jumpers in the photo below.



3.5 *Layout items on robot base*

Next, you will see how everything will lay out when it is all done. Refer to the photo below



3.6 *Create photodetector iPod holder assembly*

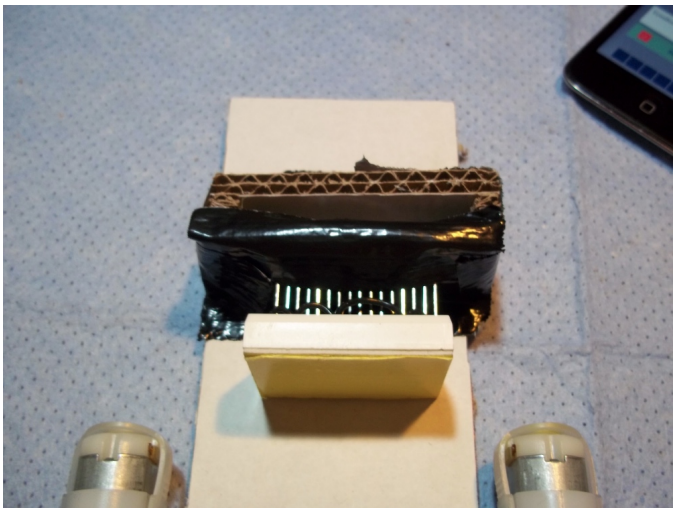
This is the hard part.

First start EyesBot Driver, start it streaming, and then place it in the holder. You will notice that the black squares line up with the photodetectors. You now need to push the photodetectors towards the screen while ensuring that all of the detectors stay lined up with the corresponding black square on the iOS device screen. When the detectors are in touch with the sticky side of the tape, they will be held in place in the correct position. See the photo below to see the photodetectors properly lined up with the black squares.

3.7 *Add ambient light shielding and attach holder to body*

This next step uses electrical tap to screen out ambient light and attach the holder to the robot platform.

DO NOT USE DUCT TAPE OR ANY NON-ELECTRICAL TAPE – Most tapes will conduct electricity and will cause you to have a weak signal if they bridge any two photodetector leads. If you don't have any electrical tape, put paper on both sides of the leads and tape it into place so that the leads touch paper rather than tape.



Tape the holder firmly in place, but **the tape that covers the top of the photo detectors should be done in a way that is easy to peel back**, since you will likely need to peel it back frequently while building the robot as you will knock the photodetectors out of alignment and need to reposition them.

3.8 *Loading the sketch and testing*

Congratulations on getting to this point, its time to load the sketch onto the Arduino and test to see whether the photodetectors are properly positioned and hooked up.

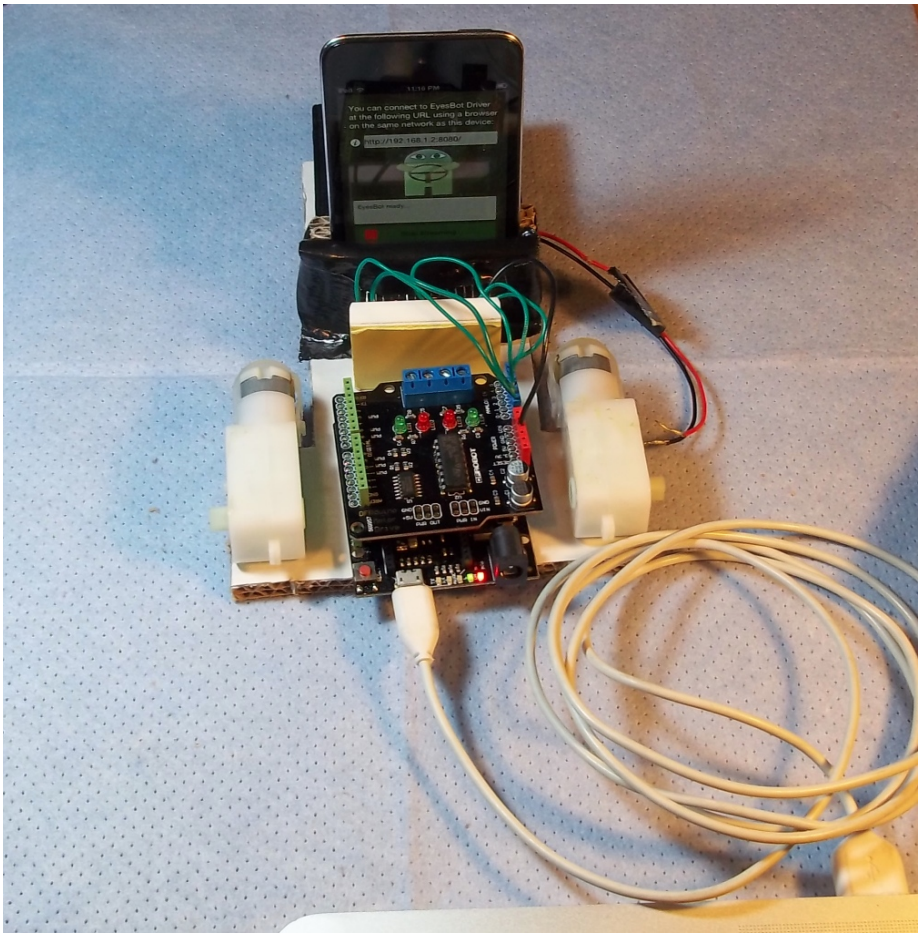
Do NOT connect the battery to Vin and ground at this point, the power will come from the USB cable.

In the bundle of files that you loaded along with the app, look for:

EyesBotDriver_1_2_RS.ino

This is the sketch that you will be using, so move it to a convenient place on your hard drive, start up the Arduino IDE, and load the sketch. Next, look for the text:
//use these for debugging
and uncomment the next six lines.

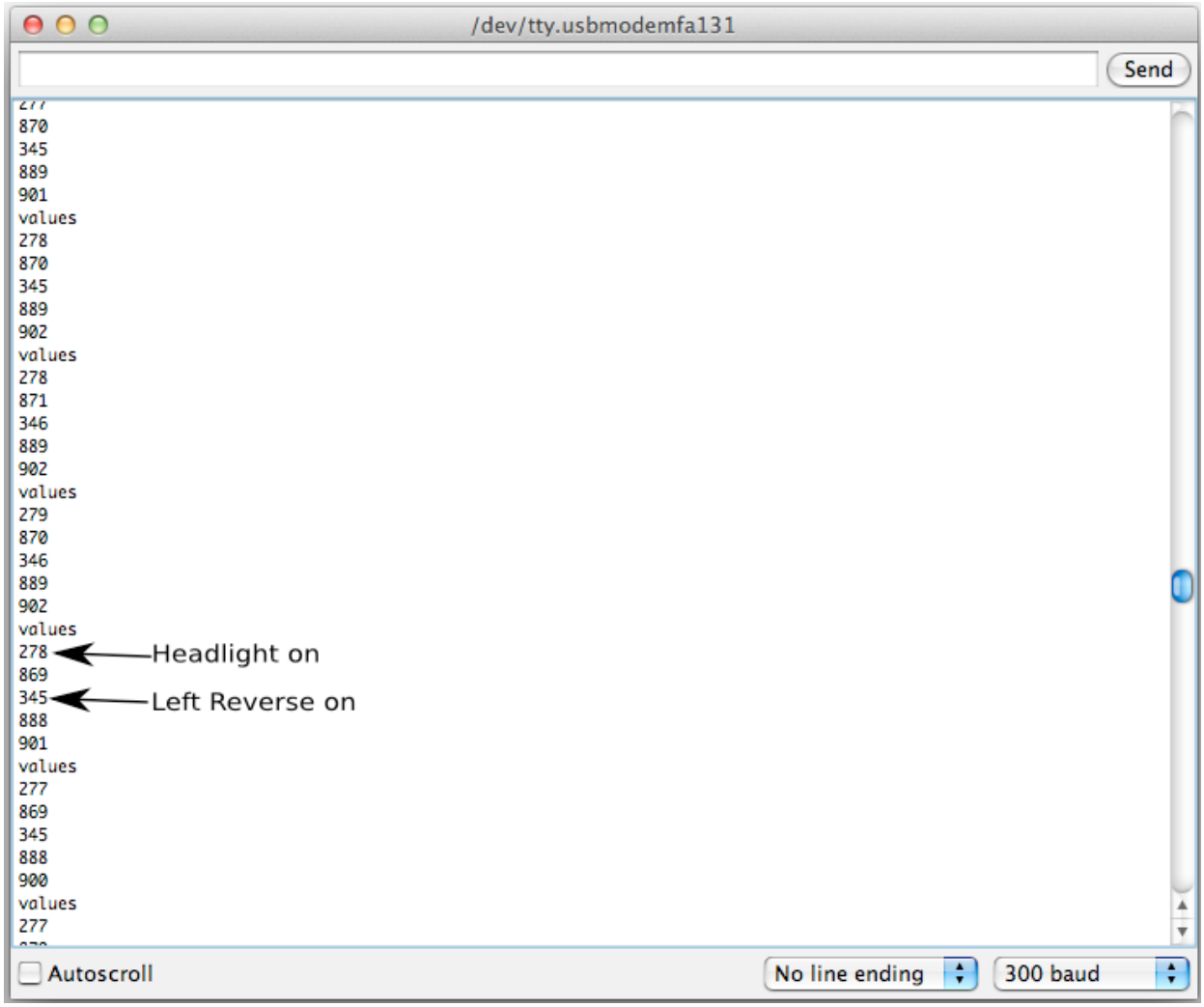
You should now startup EyesBot Driver, start it streaming, and put the iPod into the holder:



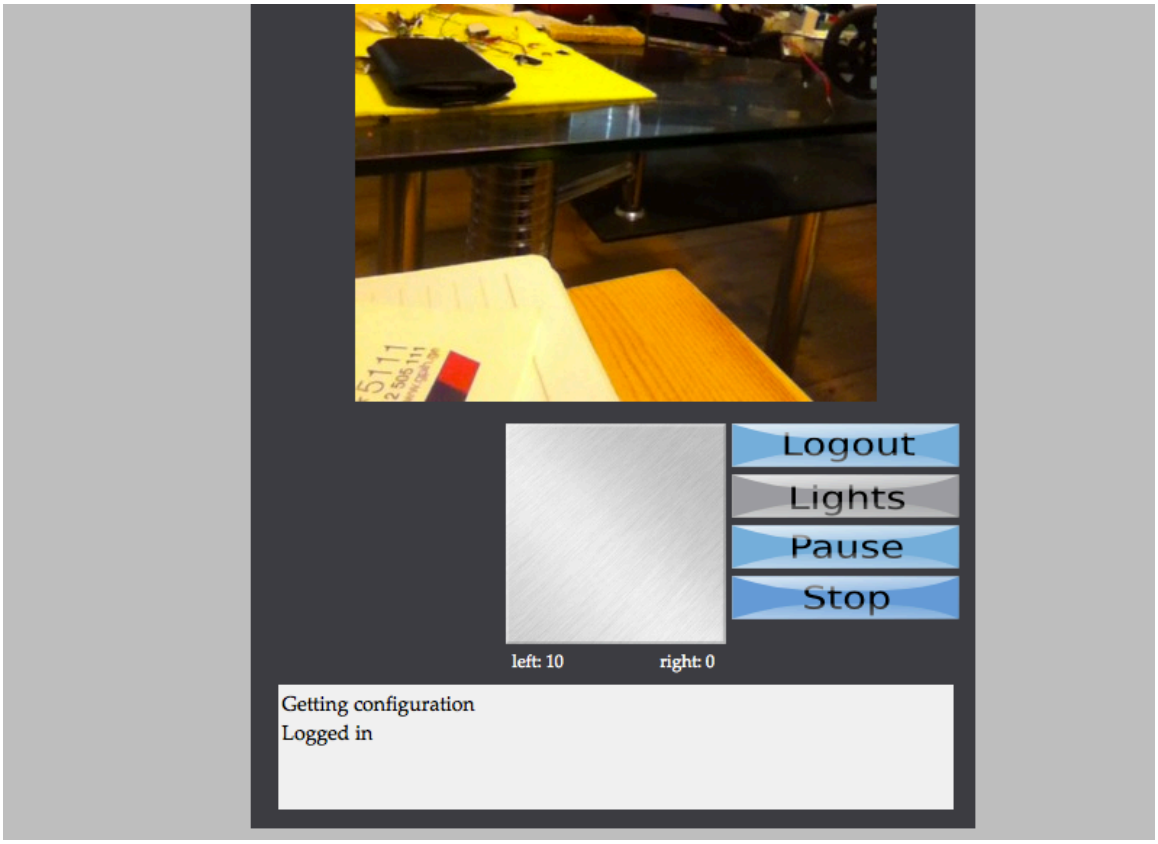
You should start up a browser and connect the browser to the address given on the streaming interface.

Now upload the sketch and open your serial monitor (press the button that looks like  in the top right corner of the Arduino IDE). Set your serial monitor baud rate to 300. The values should be around 900 for all pins. If any values are below 800, make sure that the any tape that touches the phototransistor leads is electrical tape, that there are no wires touching that should not be, and that ambient light is shielded out of the iPod holder with the black tape.

Now, use the browser interface to turn on the Lights (click the light button). Make sure the browser has the keyboard focus and that the light button turns white. There are no headlights on the robot yet, but it will change the resistance values. The first resistance value should drop. In my case, it dropped to about 300.



Next, click drag on the “trackpad” on the EyesBot driver web interface. If you drag to the corners, you should be able to set left and right to 10 and -10 independently (the left and right below the trackpad on the web interface refer to left and right velocity, which can vary between -10 and 10).



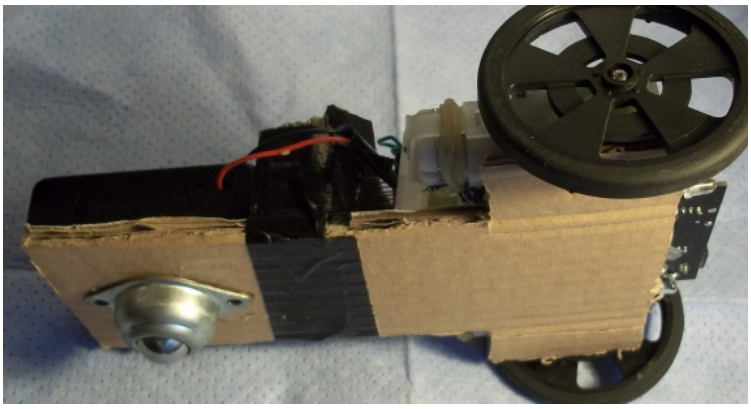
As you set each velocity to its maximum or minimum value, the corresponding values reported on the serial monitor will drop.

If the drop is weak, the mostly likely cause is that the alignment of a photodetector is off. Peel back the ambient light shield and make sure that all of the photodetectors are aligned properly with the corresponding squares on the iOS device screen.

After you have all the detectors aligned properly, you may want to put a little glue around the photodetectors to hold them in place. Make sure that the photodetectors are tightly adhering to the tape if you glue them, since otherwise the glue will be drawn into the gap between the tape and photodetector and lessen the sensitivity of the photodetector. Many types of glue are weak conductors, please be sure to avoid contacting the leads of the photodetectors if you glue them.

3.9 *Attach wheels and casters*

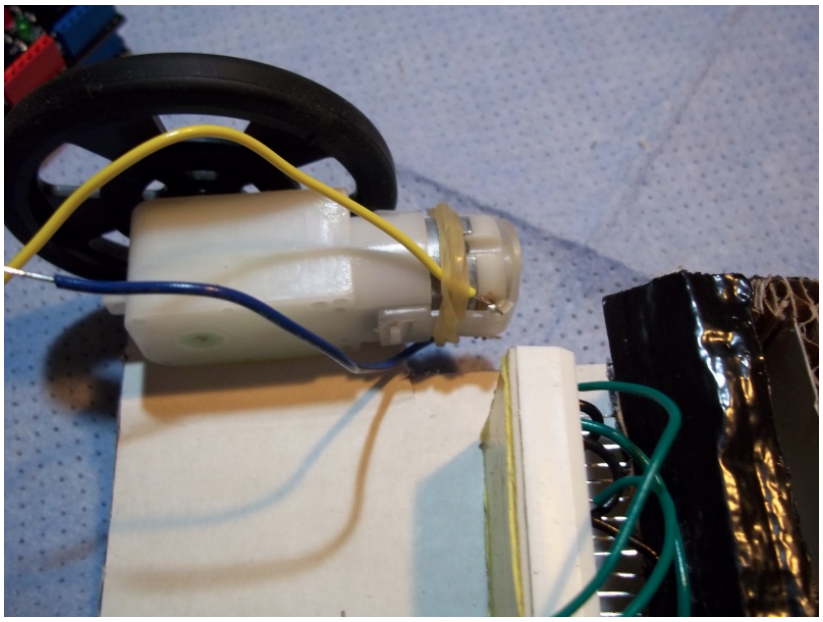
Before your robot starts rolling, you will need to glue the caster in place on the bottom and attach the wheels. Prior to turning the robot to the side, be sure to use electrical tape to hold the Arduino and battery holder in place.



3.10 *Soldering motors and headlights*

At this point you are about set to really start using the robot. You will need to connect the motors to the motor controller shield and also add headlights. Since both involve soldering, you may want to get everything set up and then do all the soldering at once.

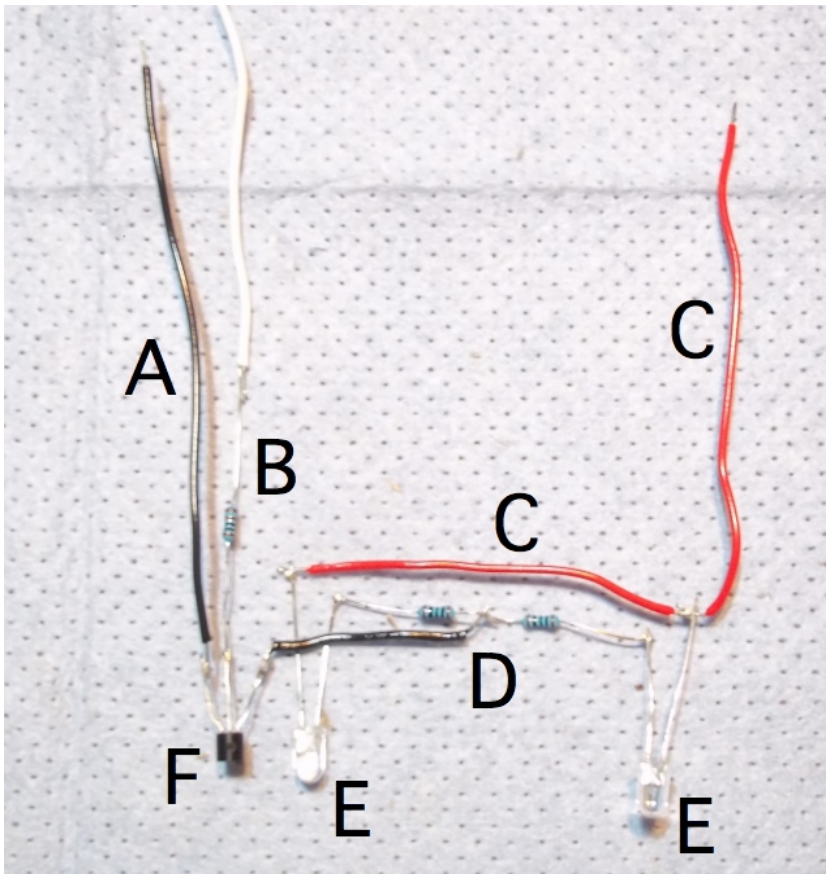
To connect the motor shield, you will need to solder leads onto the motors as shown.



The yellow leads are soldered to the top tab on the right motor and the bottom tab of the left motor. The blue leads are soldered to the other tabs. I typically wrap rubber bands around the leads and the motor so that you don't end up twisting the tab on the motor off as you move the leads around.

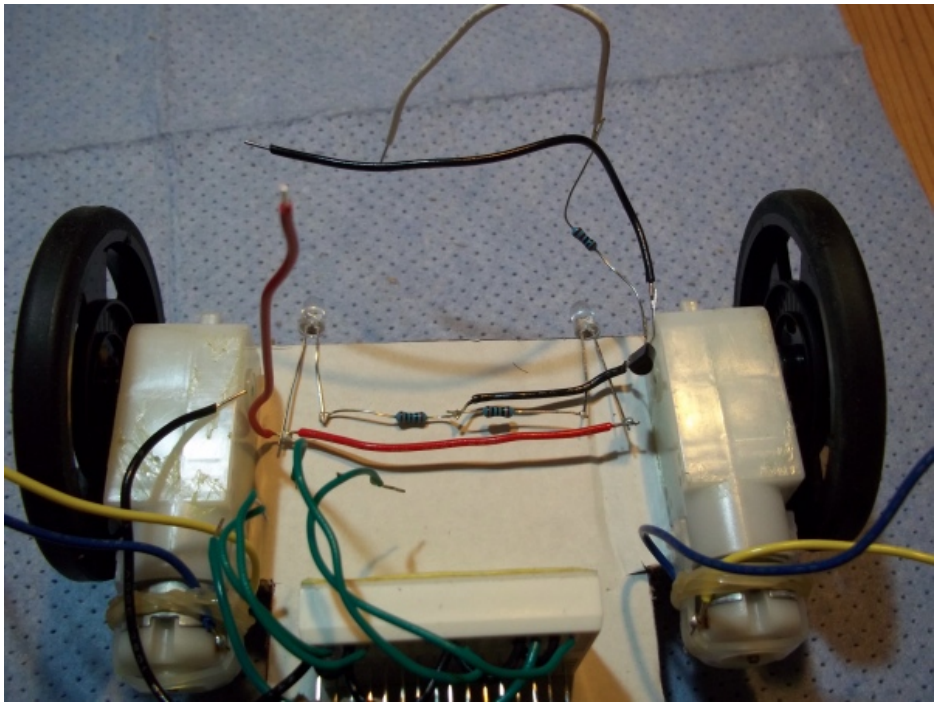
After they are soldered into place, you should create the headlights, then the headlights and leads from the motors can be attached to the motor shield at the same time.

To create the headlights, you will need to solder the 2N3904 (or similar NPN transistor), short sections of wire, the 1K base resistor and the 68 Ohm LED resistors as shown below.

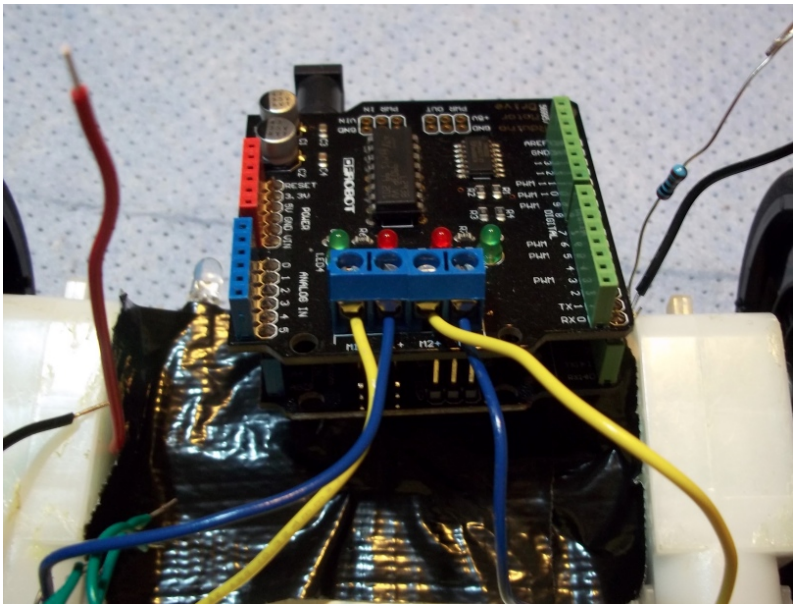


A. This black hookup wire connects the emitter pin (left pin with the flat side facing away from you and the leads of the transistor pointed away) to a ground pin on the 1A motor shield.
B. This 1K base resistor and short section of white hookup wire connects the base (center) pin to digital pin 12 on the motor shield
C. These sections of red white connect the LED anodes to each other and to the +5 v supply on the motor shield
D. These 68 Ohm resistors and short section of black hookup wire connect the LED cathodes to the collect on the transistor
E. Two white LEDs
F. 2N3904 transistor or similar

Once this is soldered, it should be place on the robot platform as shown and then taped into place with electrical tape:

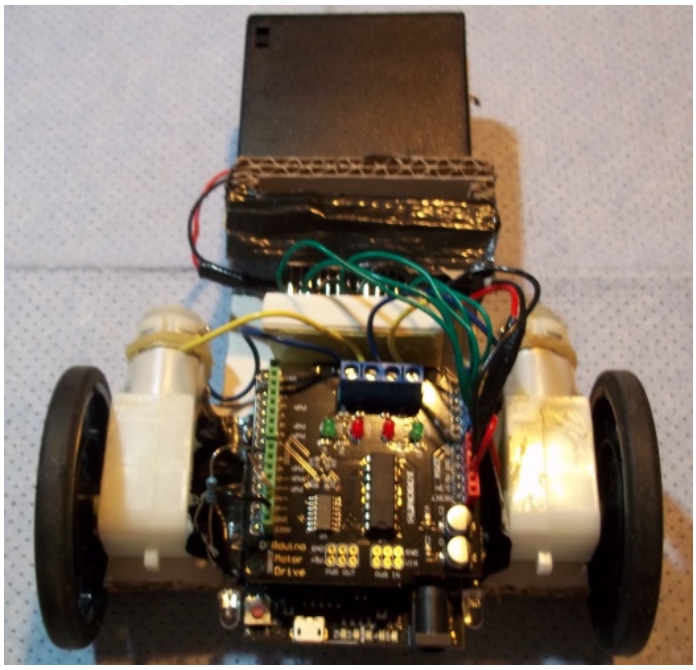


After the headlight assembly is covered with electrical tape, the Leonardo and motor shield should be put back into place and the leads from the motors attached to the motor shield.



Next, connect the red lead from the headlights to the +5V pin on the motor shield, the black lead from the headlights to ground (there is a ground pin on the light green header side of the motor shield) and the white lead from the headlights to digital pin 12 on the motor shield.

After everything is in place, you can trim away some of the excess wire and reroute some of them so that the final robot is a bit neater looking:



At this point, check the alignment of the photodetectors again as you will likely push something out of alignment as you attach the motors and headlights. To fix it, peel back the ambient light shield and, with the iPod in place and the streaming interface showing, move the detectors back into place. Re-test the resistance from the detectors as done in section 3.8 to make sure they are properly positioned, since this is the last stage prior to real use of the robot. When you re-test, make sure the batteries are not connected to the motor shield and comment out the contents of the body of the Motor1 and Motor2 functions.

4 Troubleshooting and use

In this section, you start to use your robot, and learn a bit about troubleshooting

4.1 *Using*

When you are ready to use EyesBot Driver, make sure to:

- Fully charge the iPod/iPhone battery since streaming video to a web interface is battery intensive
- Turn on the iOS device's "Do Not Disturb" feature in Settings

- Make sure you use the robot away from any drop-offs or dangerous situations.

Now you can startup EyesBot Driver on your iPod or iPhone, press the "Start Streaming" button, and put it into place in your robot body and turn on the robot. Connect to the address showing on the streaming interface with a browser and then you should be able to move your robot around with the trackpad on the web interface.

4.2 *Battery life*

If you are using 4 AA alkaline batteries, you will have about 6000 mAh of power. Since the motors each draw about 75 mA when running and the Arduino idling takes 50 mA, the total current draw is something over 200 mA. Assuming it is 300 mA, 4 AA should last less than 20 hours. The iPod battery will be drawing quite a bit of current to stream video to the web interface, and will likely last a half hour or less before needed a re-charge, particularly if the iOS device's battery is older.

4.3 *Troubleshooting*

There are a lot of systems that need to work together in order for the robot to properly function. This is a limited guide to help you track done problems that may arise.

4.3.1 Network

The iPod and the system you run the browser on need to be on the same network and the ports used for communications can't be blocked by a firewall.

4.3.2 Robot

Don't use Ni-Cd cells as they supply 1.2V, which will be too low (unless you use 5 of them).

4.3.3 iPod

Be sure the iPod is fully charged, since streaming video consumes a lot of power. Also, it is best to turn on "Do Not Disturb" in the iPod settings so no incoming notifications make the robot act oddly. Additionally, if the low battery warning pops up, it may make the robot behave oddly.

4.3.4 Browser

The browser must support Javascript for the web interface to operate. On some mobile devices Javascript won't be enabled, so it may not work on those devices.

4.4 *Contact*

If you have questions, please send email to support@eyesbot.com and specify that you have a question about the EyesBot Driver RobotShop app.