

XBugBot

How to Hack a wireless Xbox controller to make an autonomous robot!

Introduction:

This project came about when trying to make a low cost autonomous robot for use in education. The idea being that students would be able to identify with the Xbox controller and enjoy hacking it into something far more interesting!

The Concept:

This was a concept to make a low cost autonomous programmable robot for education using low cost every day things that the students could identify with – ie. The Xbox Controller. There are of course several existing product in this market most notably the Lego NXT and the Vex Systems. Both of these allow users to build and program autonomous robots, but there are two key differences, 1) On the XBugBot the programming and the processor are not separate! 2) The price! The XBugBot can be build for around 1/10th of the cost of the competing products!

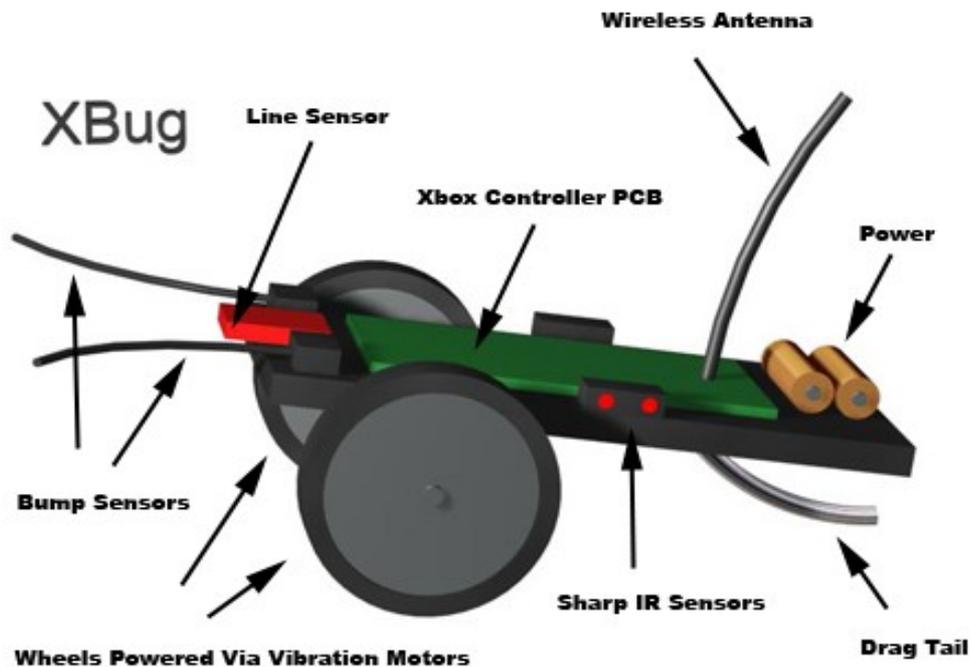


The humble Xbox Controller.

What is the Xbox controller?

Well once connected to the PC it becomes a low cost wireless DAQ (Data Acquisition System), it has several digital inputs (Buttons) and several analogue inputs (Joy Sticks etc.), plus two motor control outputs (Vibration Motors).

What can it become? - Meet XBugBot:



Hack your Xbox Controller and add a couple of wheels and sensors and you get an XBugBot!

How is it Controlled?

Using the FlowStone graphical programming language from DSPRobotics:

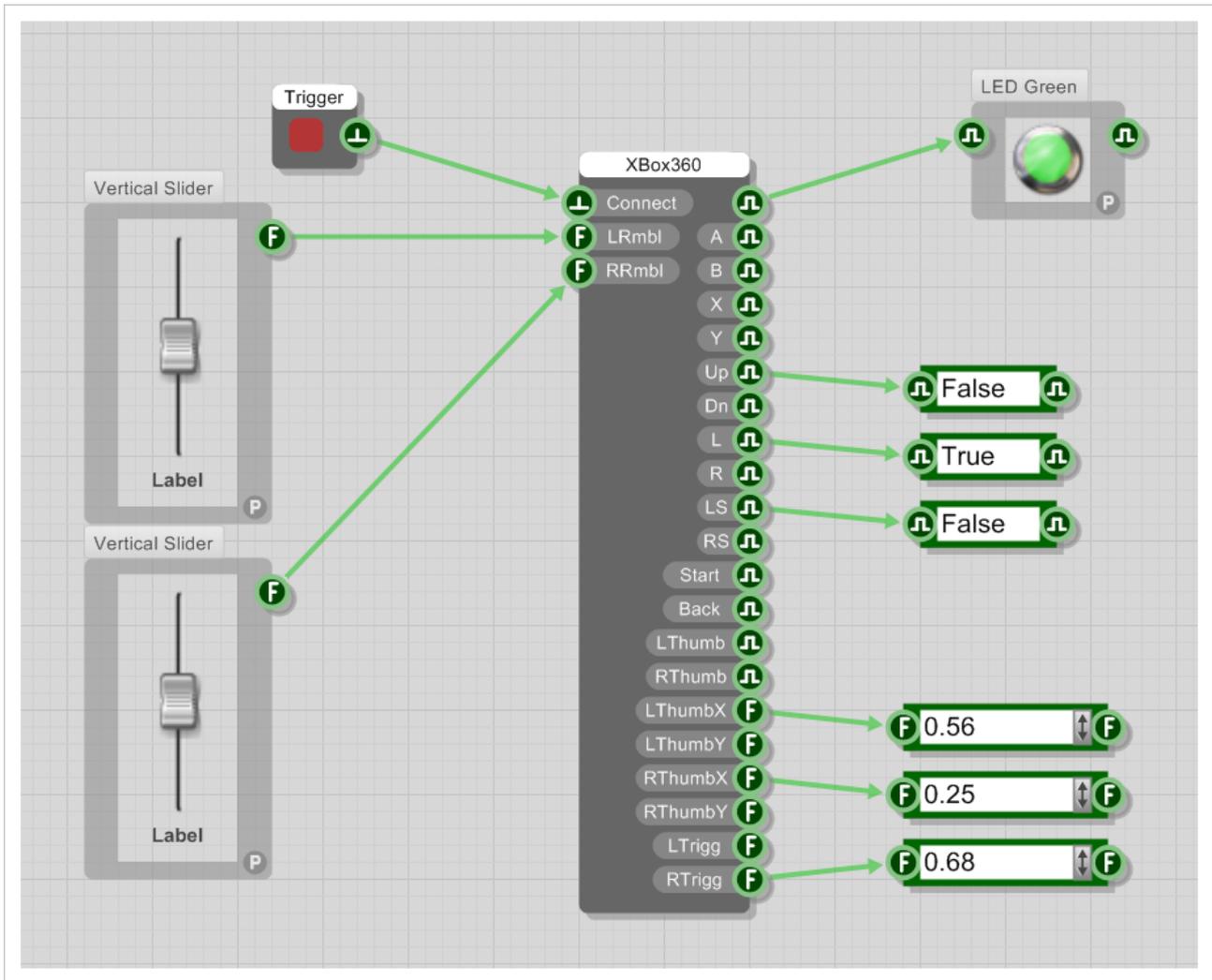
FLOWSTONE

(www.dsrobotics.com)

FlowStone is an awesome teaching tool for science, technology, engineering and mathematics (STEM) as is graphical so students can easily relate and understand it. Also the Xbox Controller is already pre-programmed into FlowStone so you can just use the Xbox Module and start programming your Bot.

Programming and Processor:

On many systems used in education the programming and the processor are separate. What this means is that you have to write your program on a computer and then download it into the device to test it. If it's not right then you have to repeat the process again, and again. This serves no purpose and just prolongs programming time and limits the actual learning process as you can't view the live data. The different with the XBugBot is that the program can be developed on the PC and run on the PC in real time, no downloading, and no compiling! This is done using the FlowStone graphical programming language and communicating to the hardware wirelessly using the Xbox Controllers own wireless connection. This is very similar to the Formula 1 or Indy Car race telemetry, with one important difference; you are in control and can change the program in real time!



The Xbox Controller Module in FlowStone

The Kit of Parts:

- 1 x FlowStone Software
- 1 x Wireless Xbox controller
- 1 x Xbox USB wireless receiver
- 2 x small motors and gearboxes eg. Solarbotics Motor 8
- 2 x wheels eg. Solarbotics 2- 5/8 " wheels
- 3 x IR range sensors eg. Sharp GP2D12
- 3 x line sensing kit or line detection sensors and resistors eg. TCRT5000
- 2 x micro switches & whiskers eg. LynxMotion Bump Switch Kit
- 1 x battery holder with switch eg. 2xAA Battery Holder with Switch
- 1 x castor wheel. Eg ½" ball Caster wheel
- 1 x 6mm Sintra board
- A few bits, cable ties, tape, sticky Velcro, solder, screws etc.



Some of the parts used

Let get going!

Step 1 - Dismantle the Xbox controller:

Remove all of the screws from the Xbox controller case most of the screws are fairly obvious but there is one hidden screw in the middle:

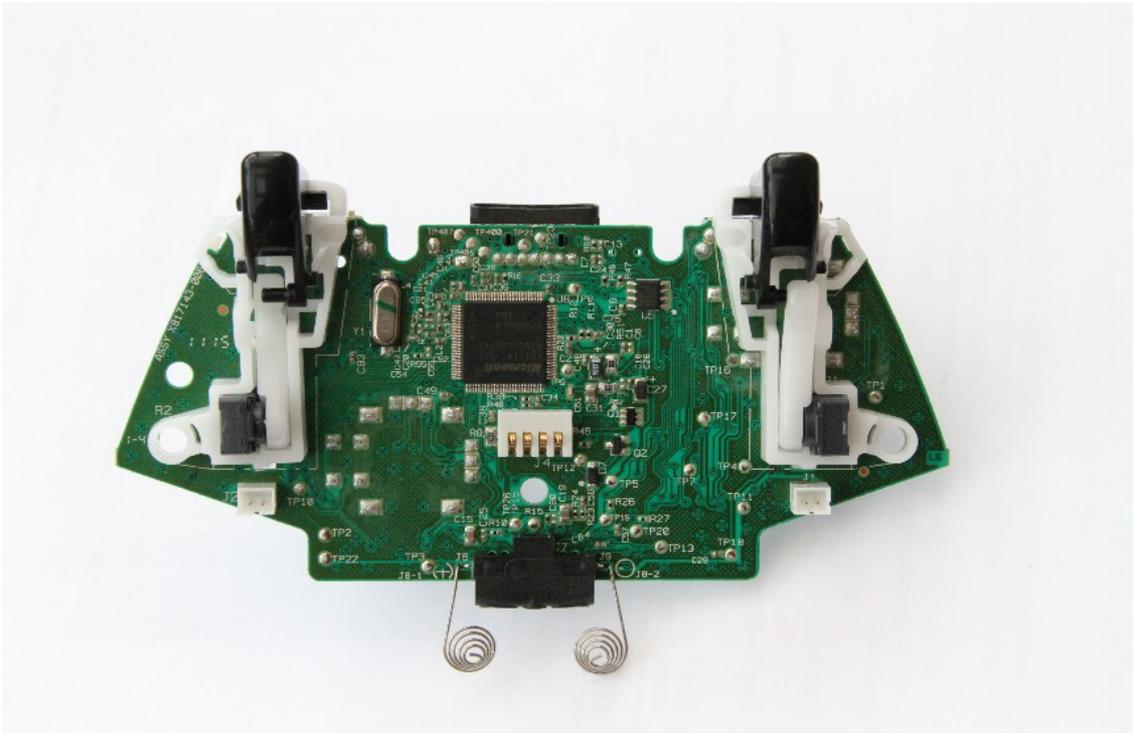


There is one hidden screw!

You might need a special screwdriver but it is possible using a flat one of the right size.

Step 2 – Remove the PCB

Unplug the motors and the PCB should just come out :



The Xbox Controller PCB

Step 3 – Remove the triggers:

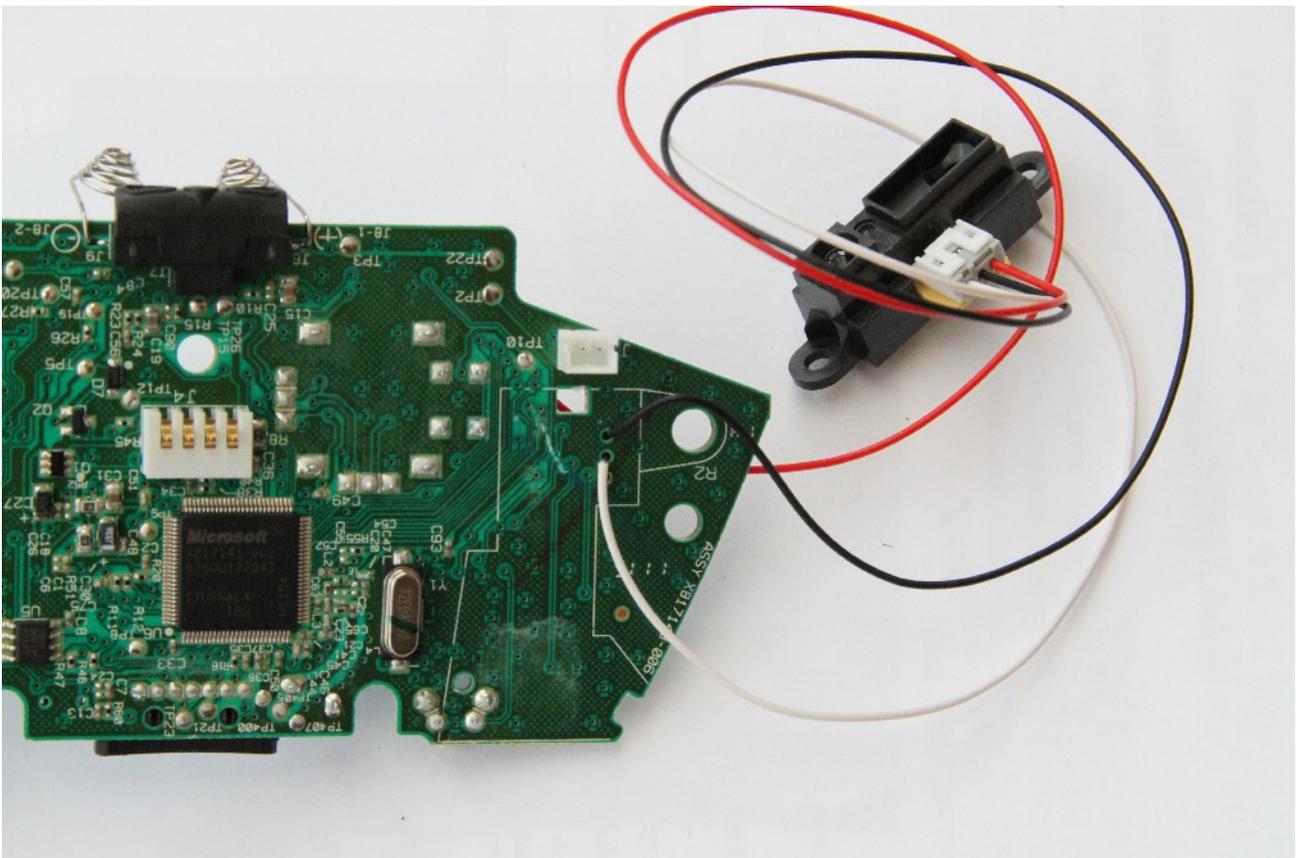
In order to mount the board flat you will need to remove the two triggers (seen above in white plastic). The easiest way to do this is to de-solder the two grey pots and remove them from the board.

Step 4 – Solder the side distance sensors:

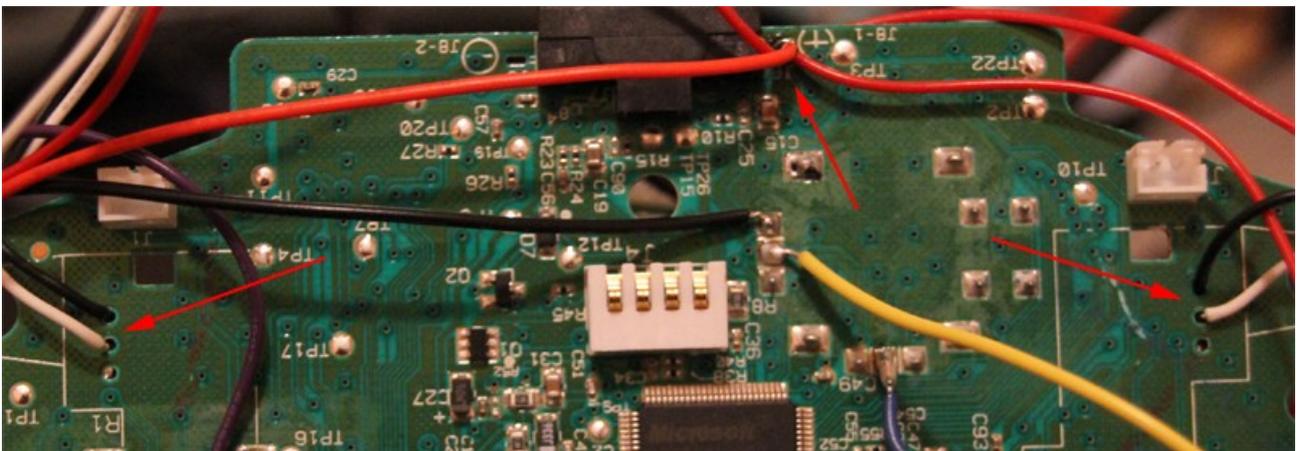
With the two grey trigger pots removed you can use these inputs for the side sensors (2 x Sharp GP2D12):



Sharp GP2D12 IR Sensor



Solder the Black wire using the top hole, and the White wire using the middle hole.



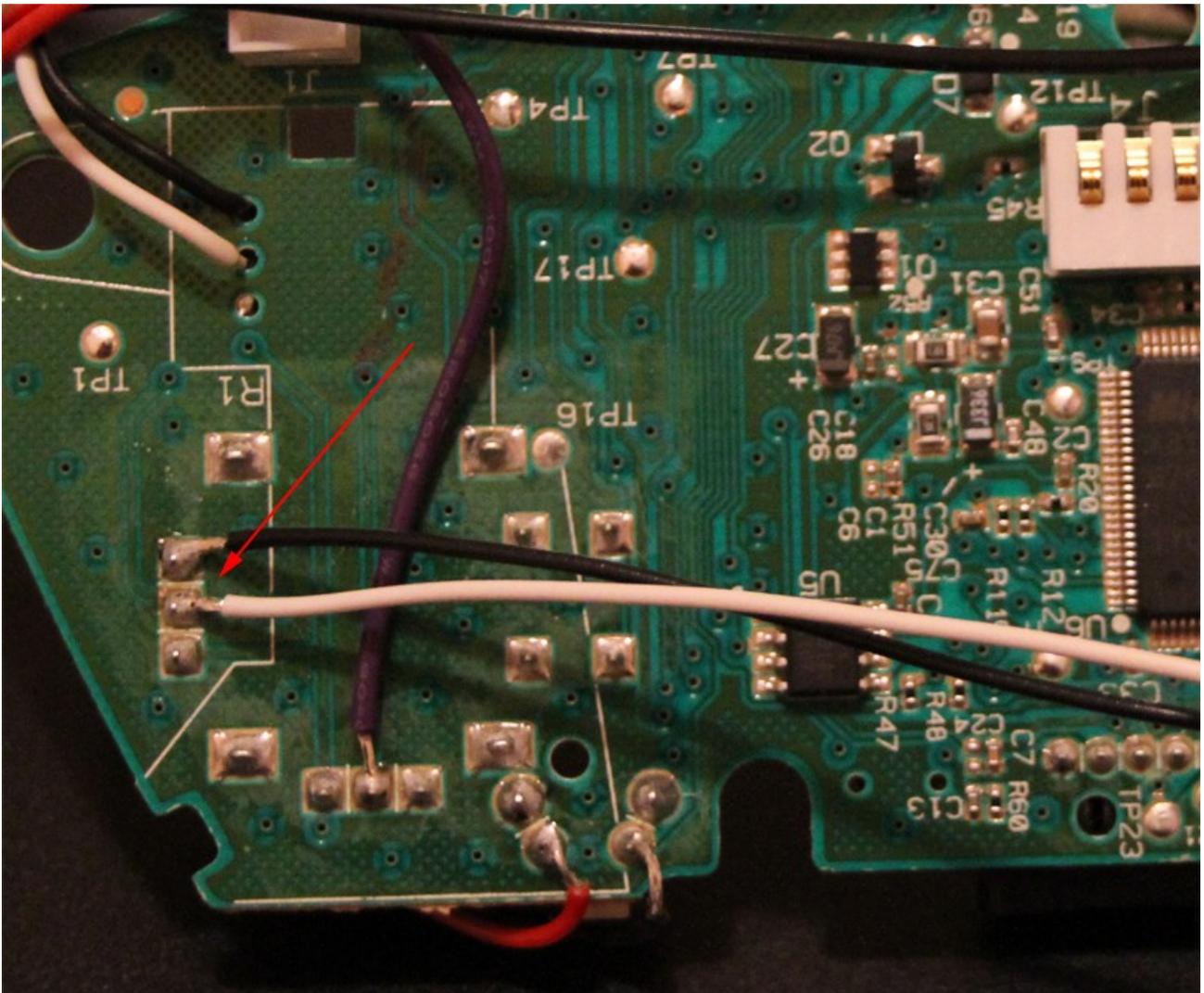
The Red wire needs to go to the +ve input from the battery, repeat this step for the second sensor on the other side.

Step 5 – Solder Bump switches

These are 'NO' (Normally Open) micro switches generally used for emergency stopping. ie. If it hits anything stop the motors.

Step 6 – Solder the front IR distance sensor:

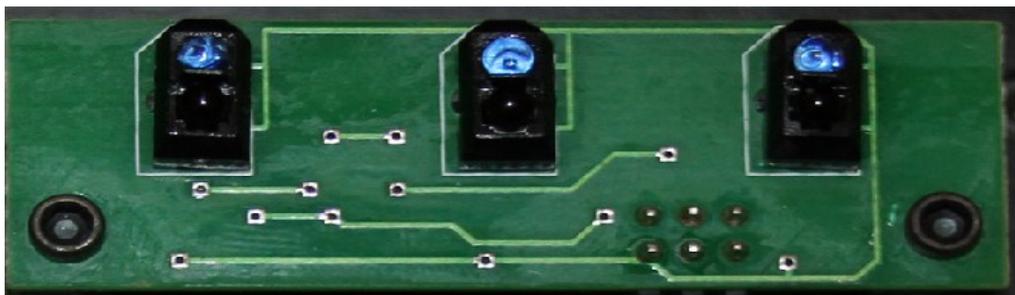
This sensor will ultimately be mounted on the front, solder wires as shown:



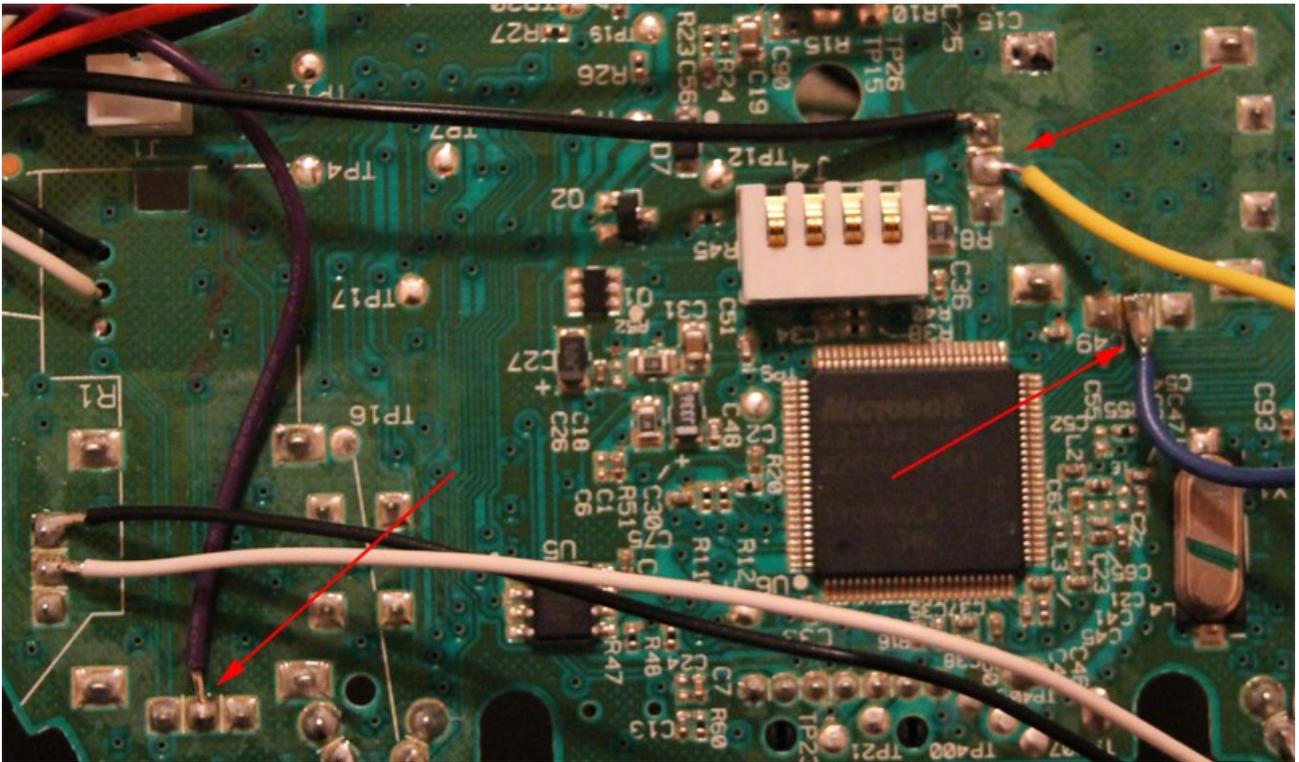
The Black and White wires from the front Sharp IR Sensor. The Red wire goes to the +ve from the battery just like the side IR sensors.

Step 7 – Solder the Line Following Sensors:

The three line following sensors need to be connected to the three remaining analogue inputs from the Joy sticks, we left the joy sticks in place and the signal were sufficient without removing them.



Line Sensor Board



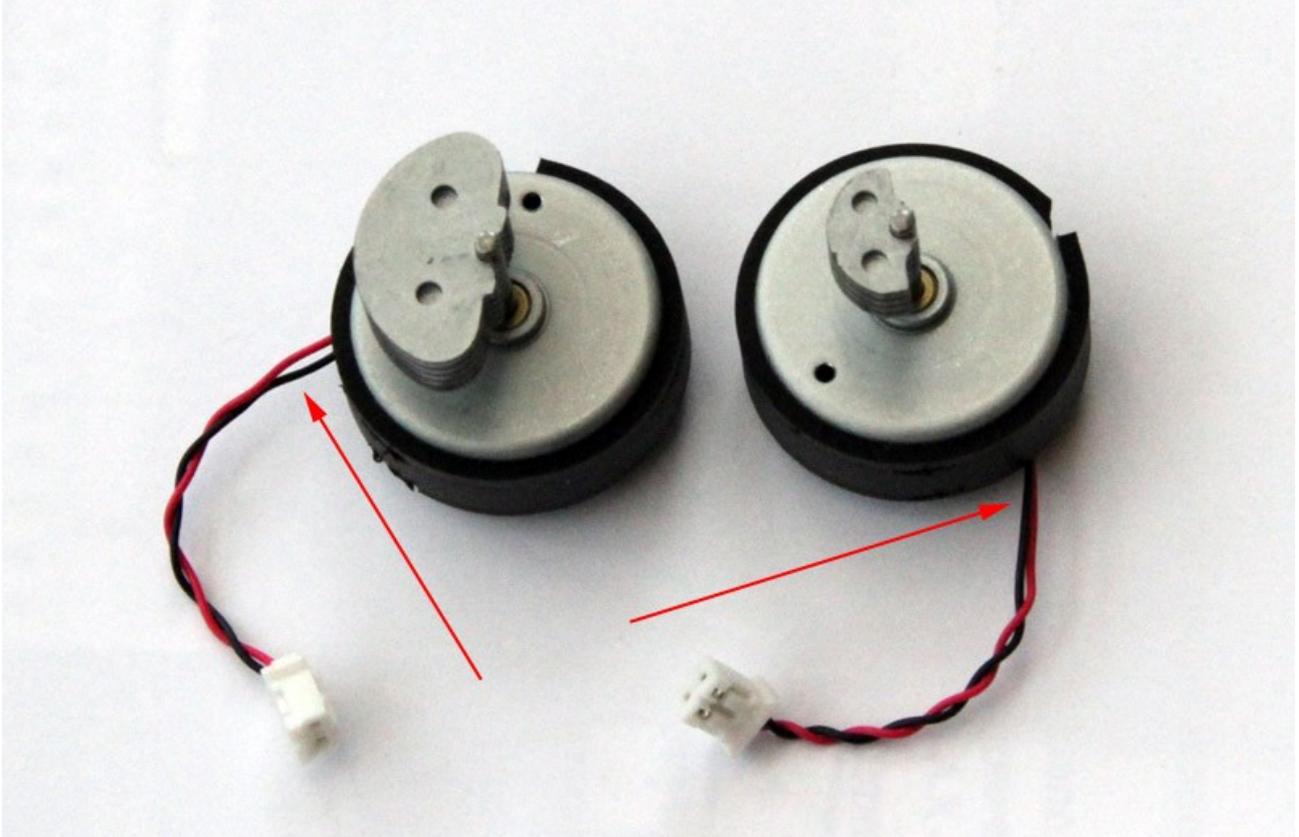
The centre pin of each joystick pot is the signal in. The Black wire (next to the yellow one) is the ground.

Step 8 – Wiring the Motors:



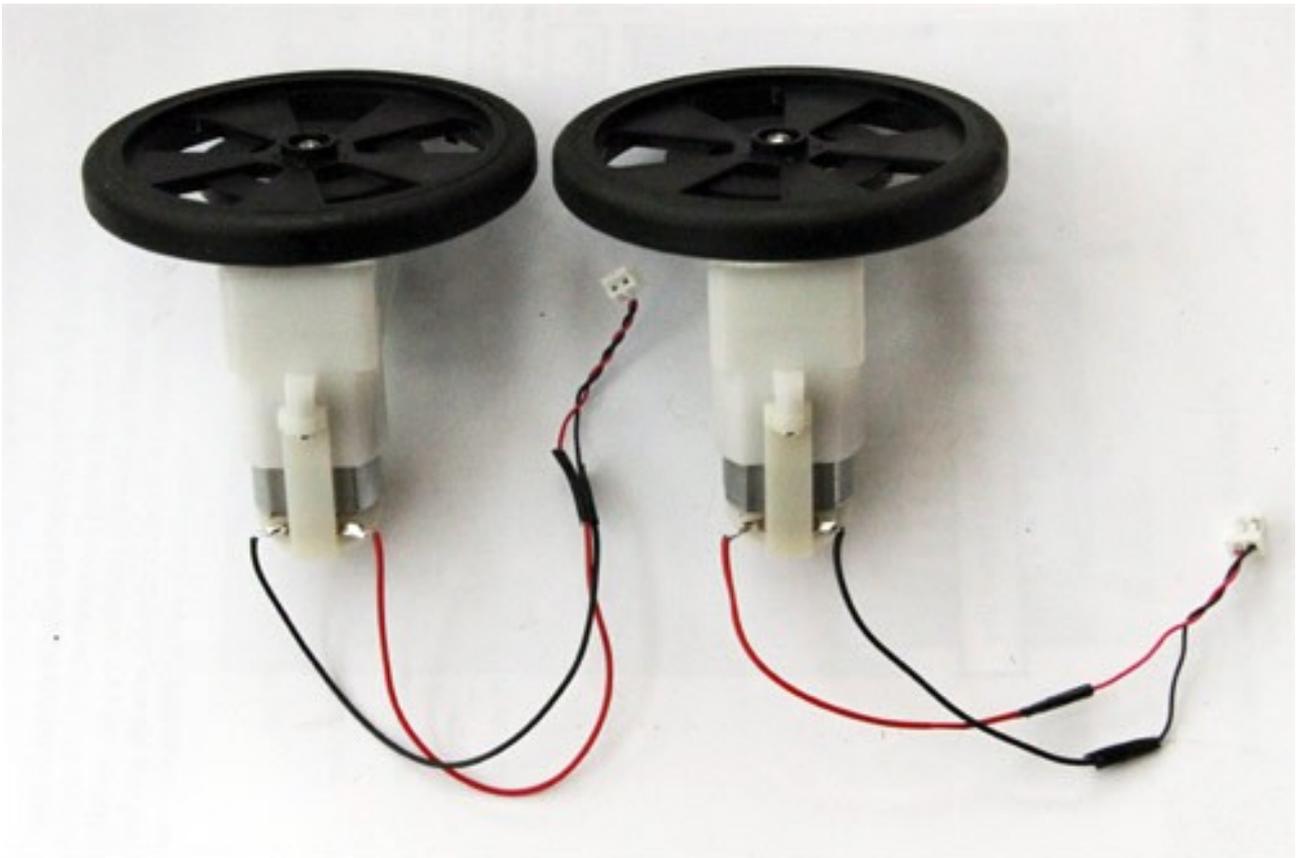
The two small geared motors (unfortunately the Xbox vibration motors were not powerful enough) need connected to the PWM motor output connectors.

To do this cut the wires off the vibration motors (near the motor) so that you can re-use the connectors and wires:



Cut the wires here on the vibration motors

Now solder these wires to the new motors. You will also need a couple of inches of spare wire to extend these so that you can position them on the chassis later:



Note the extended wires and also the polarity. The motors need to be wired up in opposite directions!

You can also add the wheels at this stage.

Step 9 – Adding the Battery Box:

The 2 x AA battery holder can now be soldered to the Xbox board (without batteries at this stage – just to be safe):



Black and Red battery wires

Step 10 – Making the Chassis:

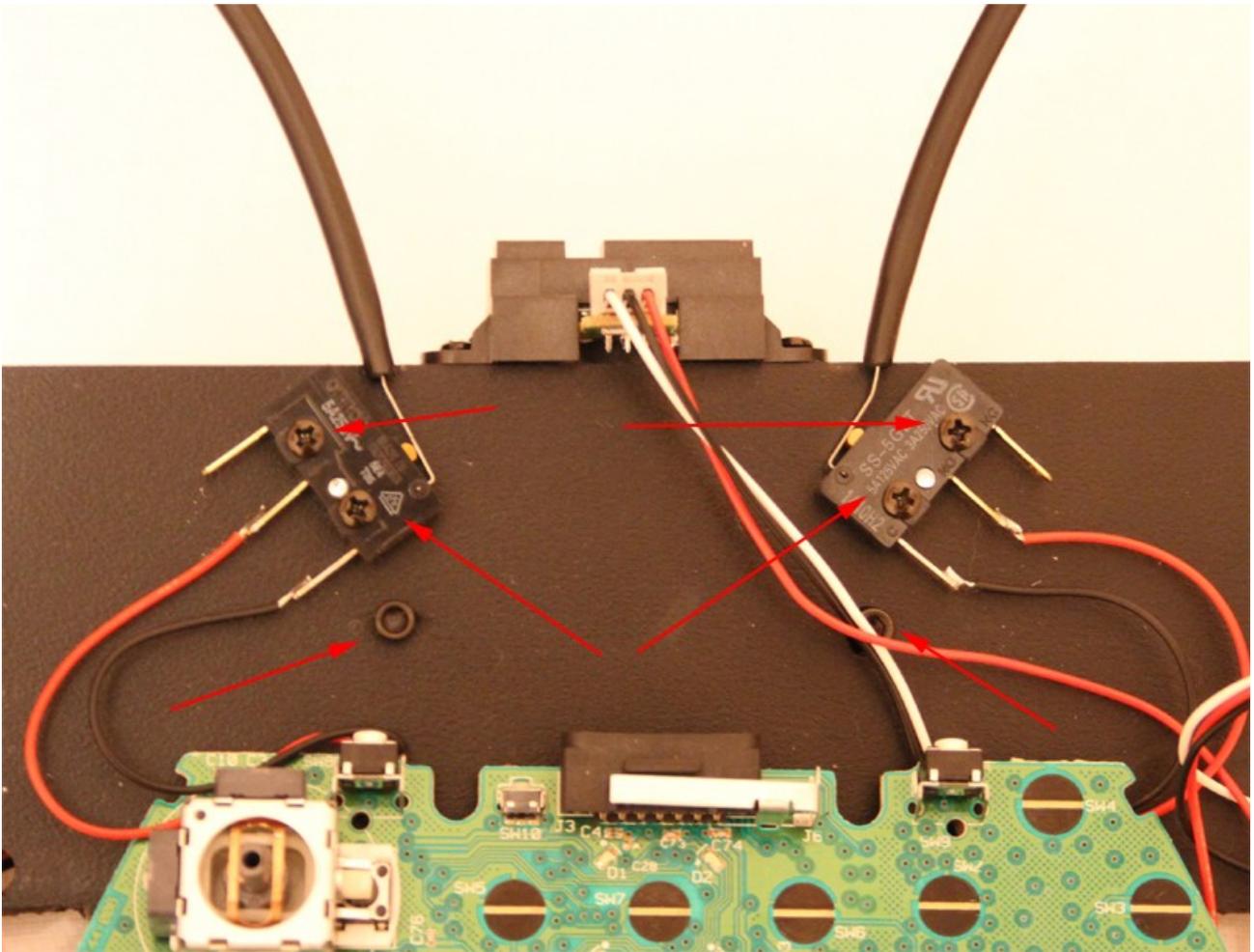
Using some 6mm Sintra board cut out your XBugBot body. We used a Jigsaw to do ours making a nice shape



Cut the Sintra board measuring roughly 8" (20cm) x 6" (15cm), the tail is 1" (2.5cm) wide & 4" (10cm) deep.

Step 11 – Drill some mounting holes:

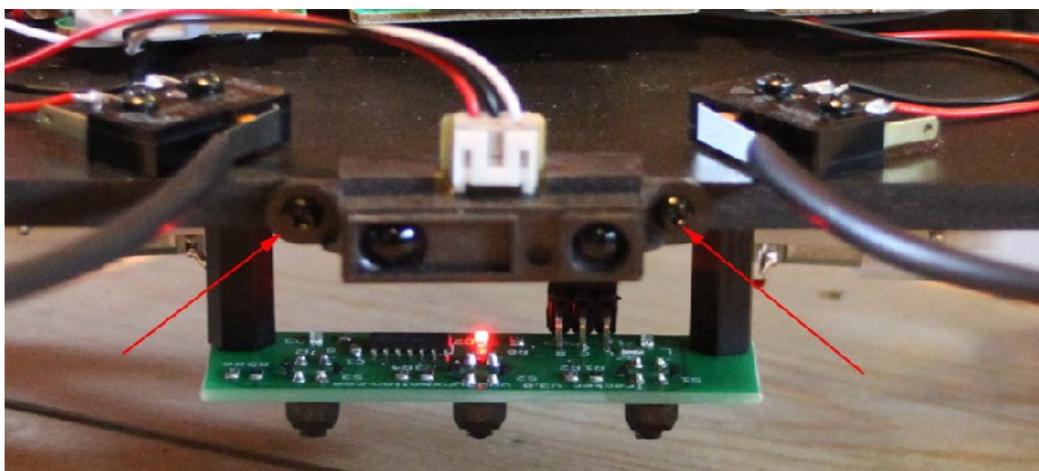
To mount the Bump sensors and line following sensors you will need to drill 6 holes:



The six mounting holes with the sensors mounted.

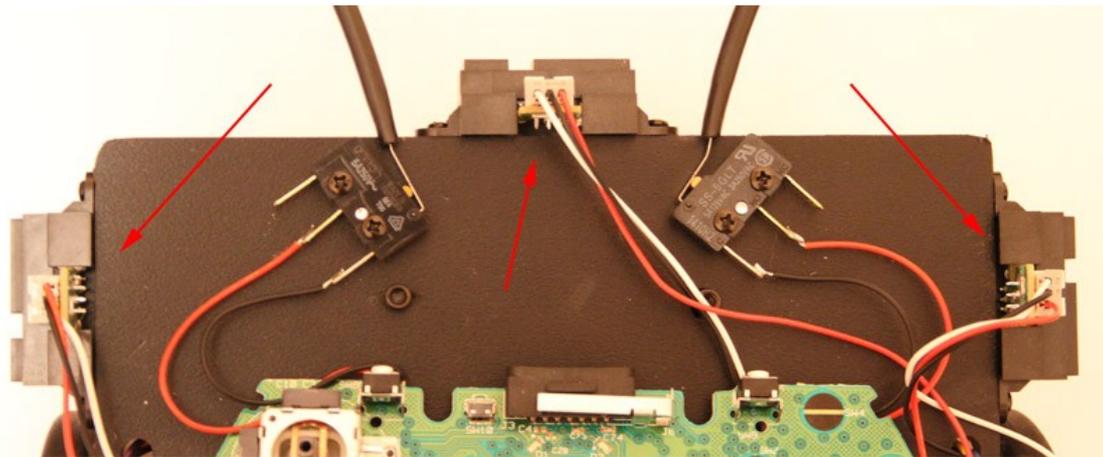
Step 12 – Mount the sensors:

To mount the Sharp IR sensors use some small self tapping screws:



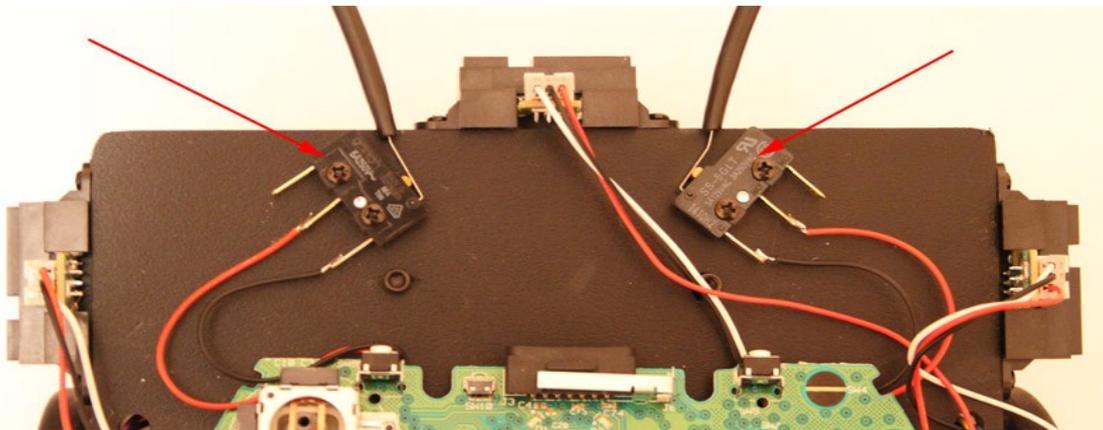
Self tapping screws used for IR sensors

Mount the two side IR sensors in the same way:



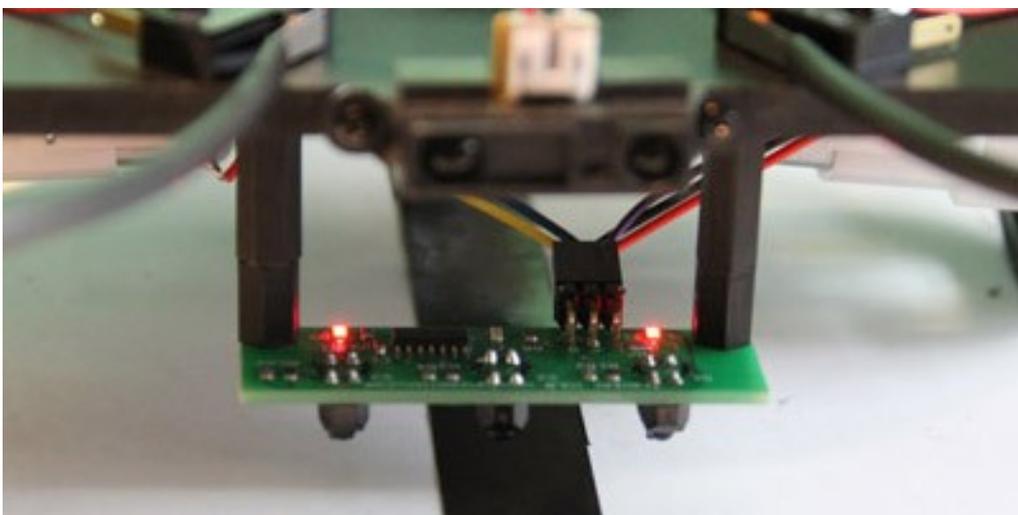
Location of the Sharp IR sensors

Mount the bump switches at an angle using the mounting holes you drilled:



Bump switch Mounting

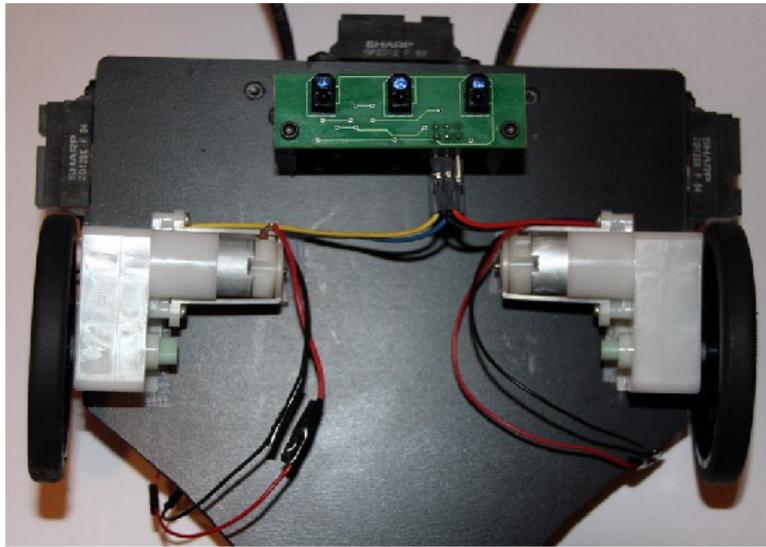
Mount the Line Following sensors on the underside using stand-offs to leave around a 1/2" (1cm) gap to the ground once the wheels are added:



Line Sensor Mounting

Step 13 – Mounting the Wheels and Motors:

Using sticky Velcro stick the motors to the underside of the chassis about half way from front to back:

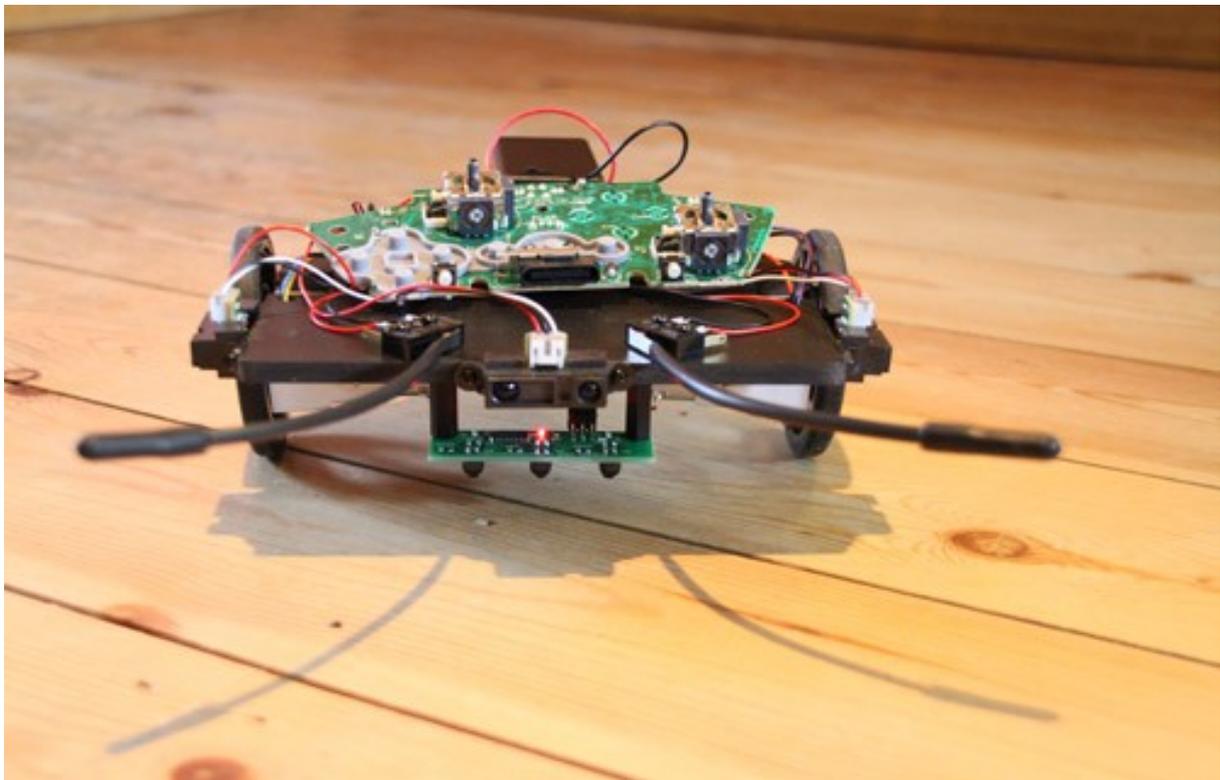


Do the same for the rear caster wheel mounting this right on the end of the tail.

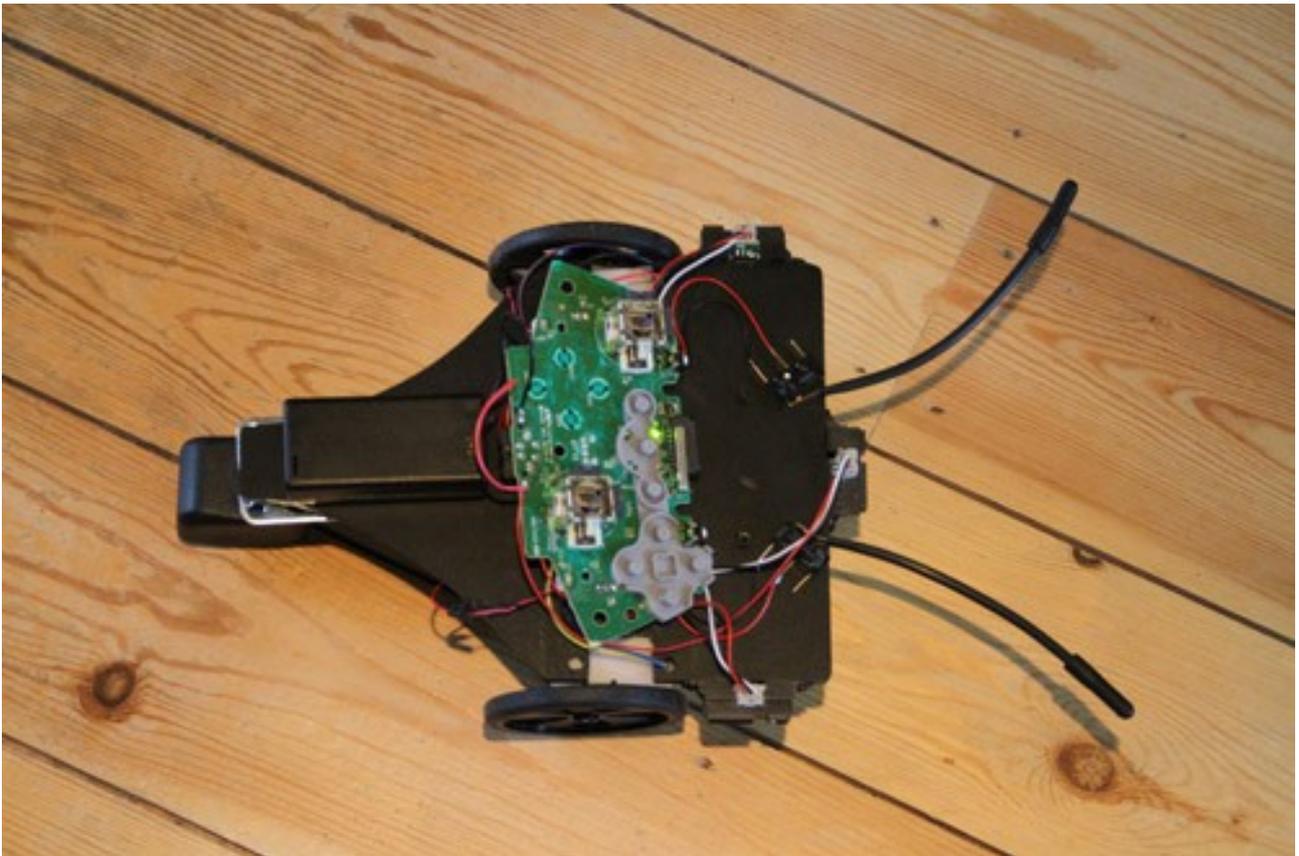
Step 14 – Mounting the battery holder:

Again using sticky Velcro mount the battery holder on the top of the tail.

Finished!



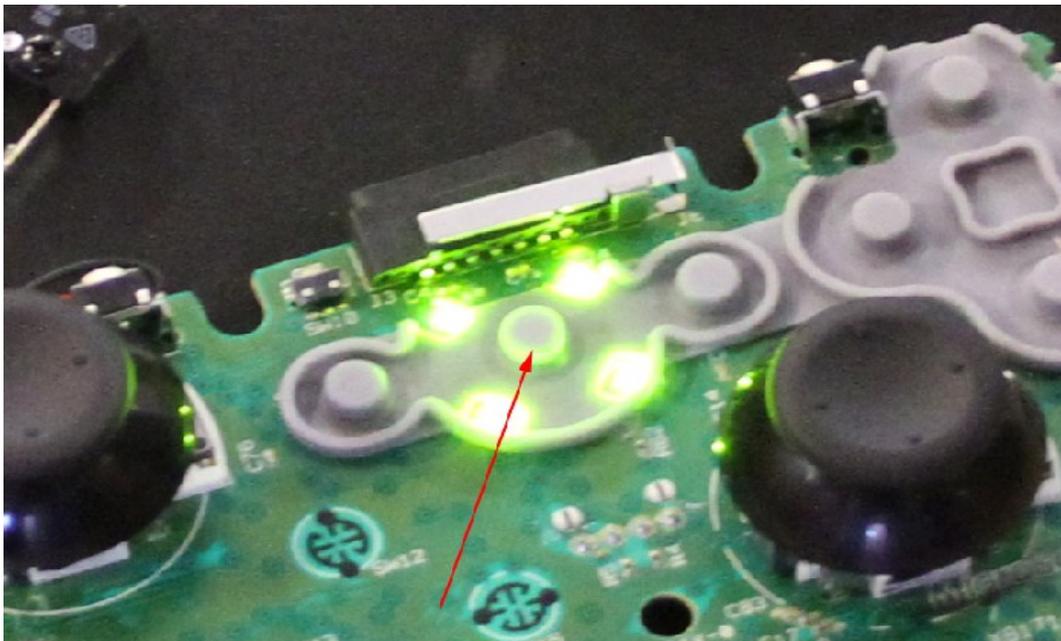
The finished XBugBot!



Top View

Testing with FlowStone:

Switch on the battery box, then you will need to press the Xbox On button. To do this it works best if you leave the rubber pads on the top or even glue them to the board:



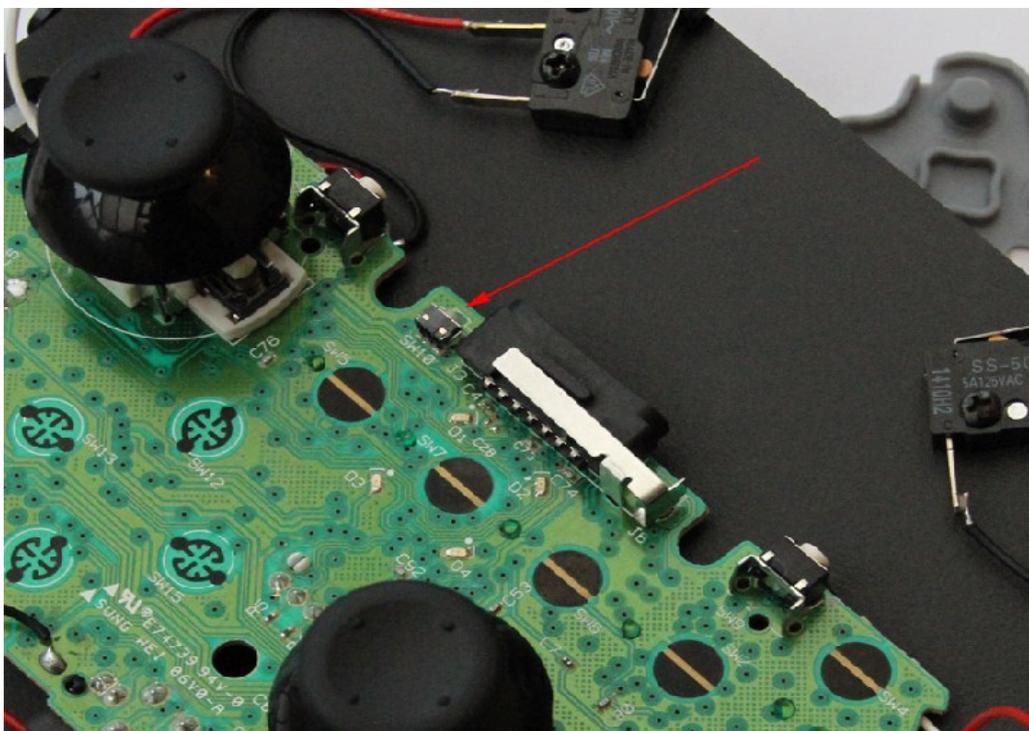
Xbox Controller 'On' Button

Next you will need to 'pair' your Xbox controller to the Xbox receiver. Start by pressing the button on the Xbox receiver:



Xbox Wireless Receiver

Then press the small button on the Xbox board:



Xbox Pairing button

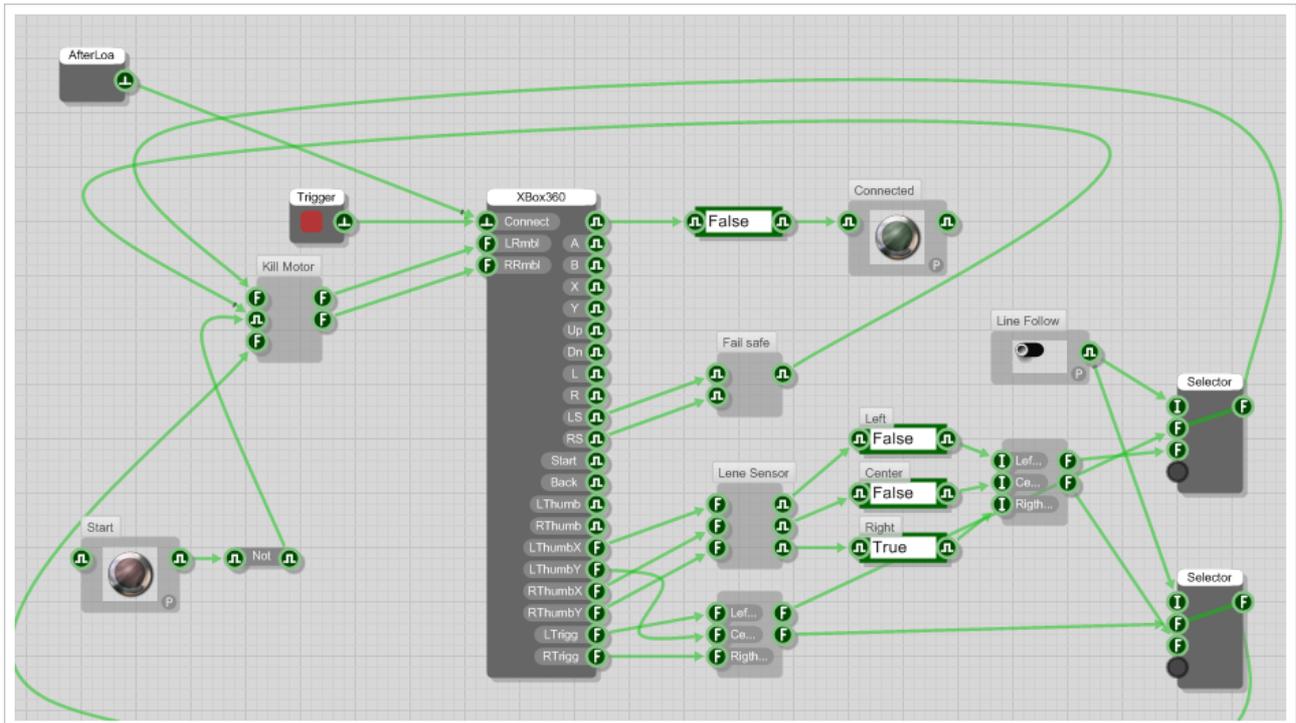
Once paired just one of the green LED's will be lit on the Xbox board.

DSPRobotics FlowStone:

Now all the hardware is built and installed you can download the FlowStone software from here: <http://www.dsrobotics.com/download.php>

There are three versions of FlowStone, FREE, Enterprise and Pro. You can get your XBugBot going using FlowStone FREE however if you want to make a bigger projects or make an EXE then you will need to upgrade to one of the paid versions.

Here is an example XBugBot FlowStone project:



XBugBot FlowStone test project.

You can download the test project from here:

<http://www.dsrobotics.com/support/viewtopic.php?f=11&t=340>

Here is the video of the first test run:

<http://www.youtube.com/watch?v=1xGA5bDD-Ug>

Enjoy!