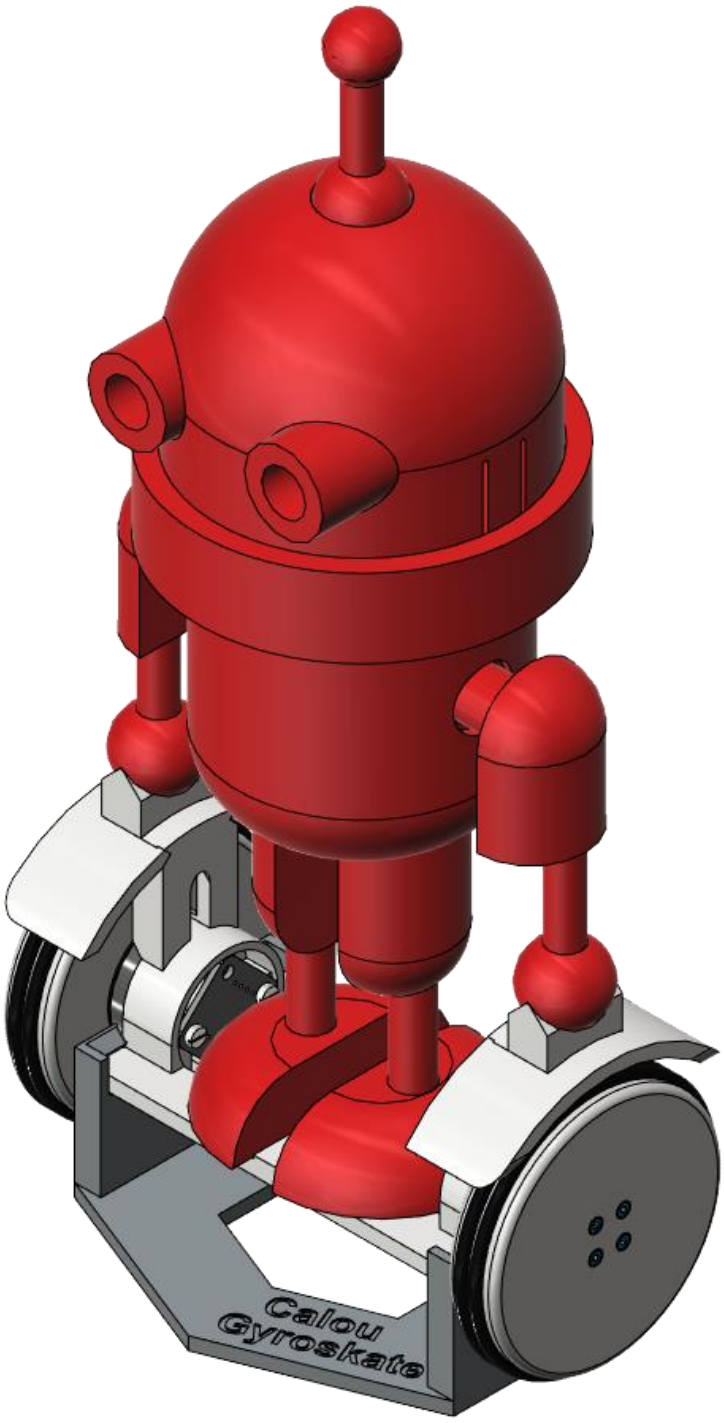
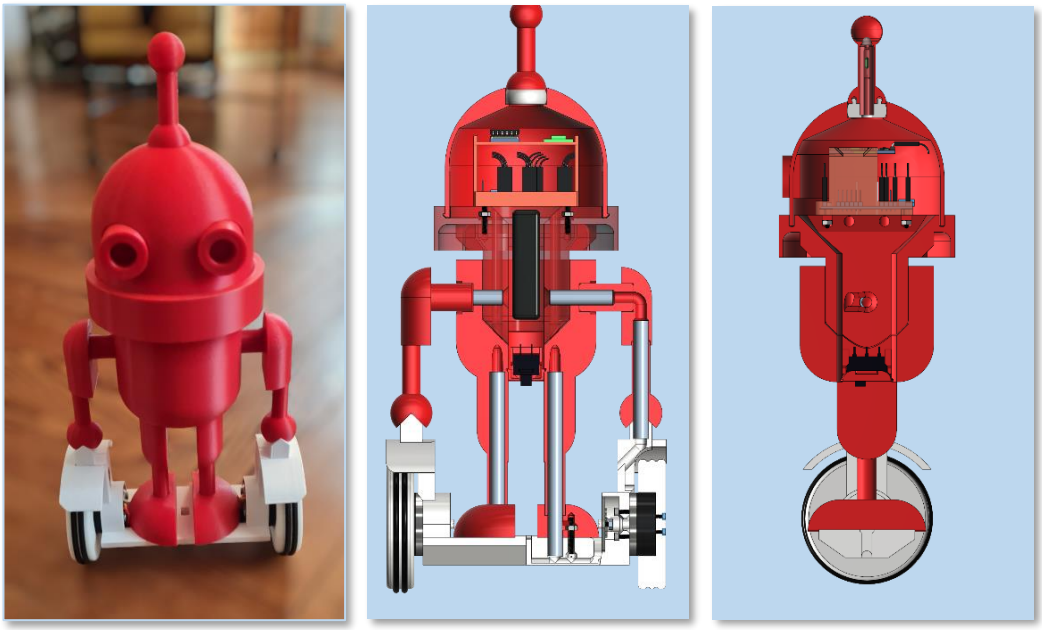
CALOU

self-balancing robot



Summary



CALOU is a two-wheels inverted pendulum controlled by Bluetooth through a smartphone.

It is a personal project entirely manufactured in 3D printing and powered by two RC Brushless BR2804-100T Gimbal Motor and a 2-axis Brushless Gimbal Controller BGC3.1 MOSFET card in a close loop with angle magnetic encoders. Those parts need some modification/adaptation to be assemble. The Controller with the IMU and Bluetooth module HC-05 are located into the head while the Body holds in the center a 500mA – 11.1V – 30C LiPo battery with a switch. The arms and legs of the robot serve as wire guide for motors and encoders.

The program was adapted from an application example developed by Antun Skuric, founder of the SimpleFOC project: an Arduino library implementing a Field Oriented Control (FOC) algorithm for BLDC and Stepper motors. https://github.com/simplefoc/Arduino-FOC-balancer

CALOU is the mascot of the French robotic association Caliban https://www.facebook.com/AssoCaliban

It can be controlled through a smartphone by using a remote Bluetooth app and simple commands. It has a blinking antenna that is activated when switch on.

CALOU demo on YouTube: https://youtu.be/TESaMDyrZCY?si=oM4P1eLyKOn7zRxZ

PROJECT - Apérobot 0x92: https://youtu.be/fIwBQCcEI_Y?si=cOJBYemO8cVf64ot

The Author gives no warranty and accepts no responsibility or liability for the accuracy or completeness of the information and materials contained in this instruction manual. Under no circumstances will the Author be held responsible or liable in any way for any claims, damages, losses, expenses costs or liabilities whatsoever, including, without limitation, any direct, indirect, or consequential damages resulting or arising directly or indirectly from the fabrication or the use of this robot, as well as of its functionalities or lifespan.

Bill of Materials

Materials and quantities are detailed into the excel file: BOM CALOU-GYROSKATE.pdf

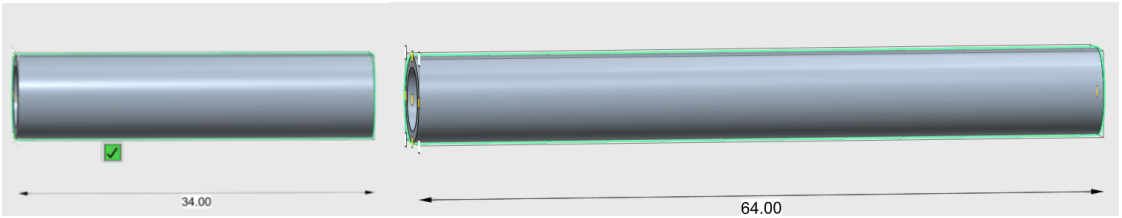
You will find some internet link for the supply of the electronic parts with price. Please note that I have no competing financial interests in those companies, and it is given without any guaranties on the component availabilities and/or price and/or specification changes.

Realization of the mechanical parts

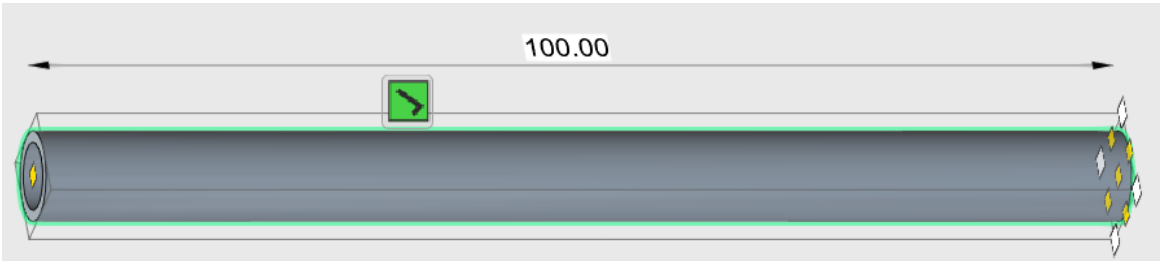
Most of the mechanical parts are made with 3D printing and without support structure. When supports are recommended it will be indicated on the side of the drawing in the list below.

Arms and Legs are reinforced by the used of 8x6mm aluminum tubing. It helps also to reduce the electromagnetic pollution that may jeopardize the analogic encoder signal. The tubes would need to be adjusted in length with a metal saw and then remove any sharp edges with a standard file.

(The tube that I purchased were 100mm each. So, the leg tubes were already at the right size)



2 x Shoulder Tubes 34mm 2 x Arm Tubes 64mm



2 x Leg Tubes 100mm

22 individual STL files are available in the folder "3D_STL".

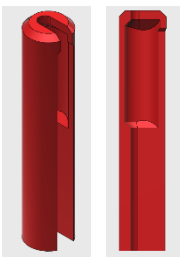
Parts list here below:

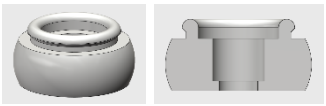
File name: Antenne.stl

File name: CaleLedAntenne.stl

PLA red

Quantity: 1

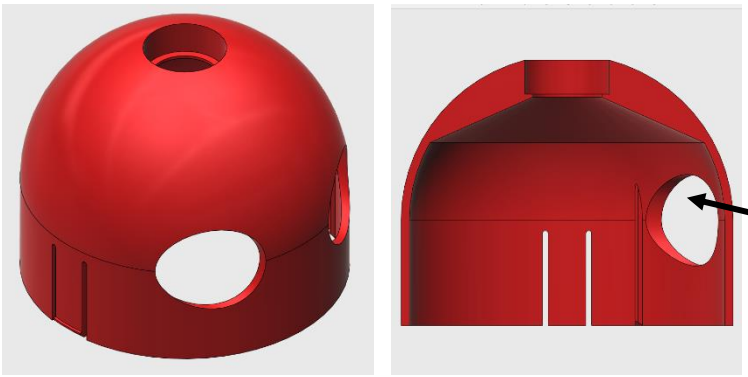




File name: JointLed.stl

TPU white

Quantity: 1

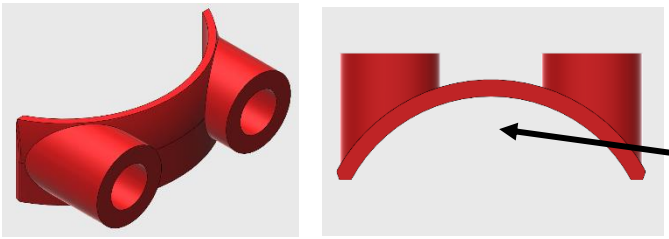


File name: Tête.stl

PLA red

Internal Supports

Quantity: 1

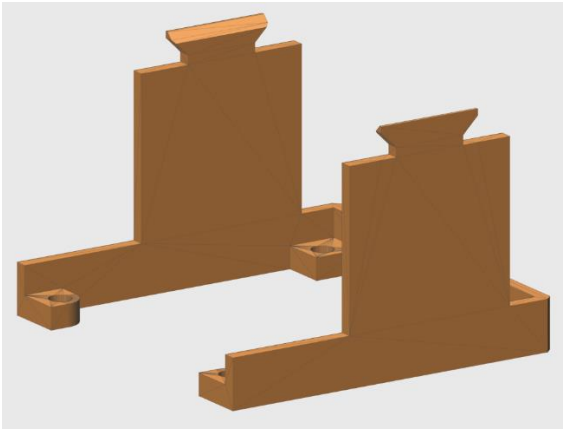


File name: Yeux.stl

PLA red

Supports in contact with Bed

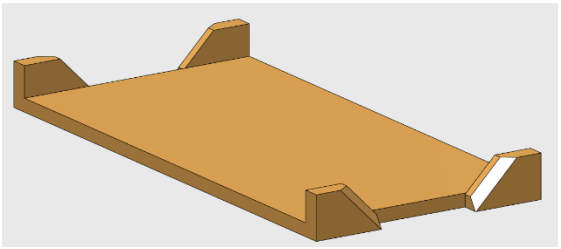
Quantity: 1



File name: SupportBGCboard.stl

PLA (free color)

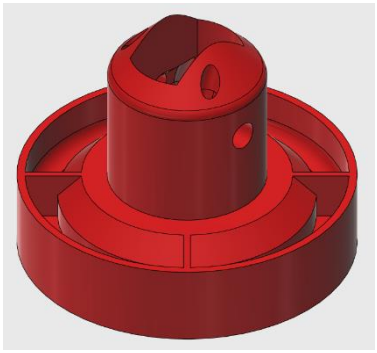
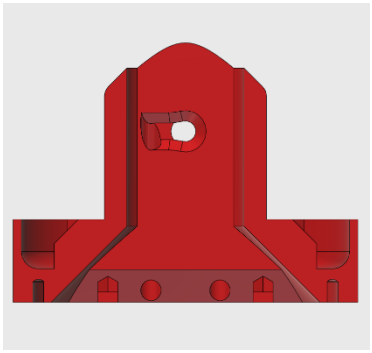
Quantity: 1



File name: SupportIMU&BLE.stl

PLA (free color)

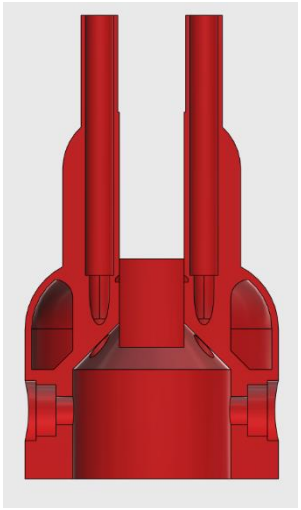
Quantity: 1

File name: CorpsHaut.stl

PLA red

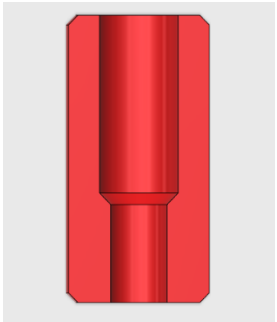
Quantity: 1

File name: CorpsBas-V2.stl

PLA red

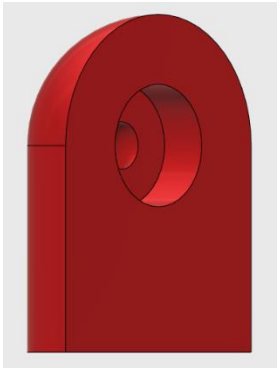
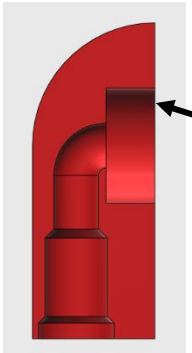
Quantity: 1

File name: AvBras.stl

PLA red

Quantity: 2

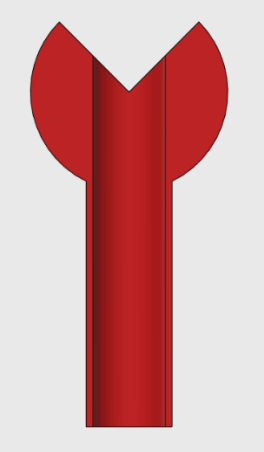



File name: Epaule.stl

PLA red

Internal Supports

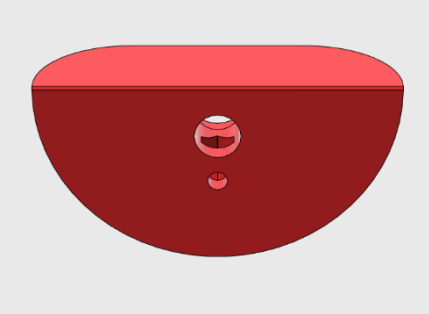
Quantity: 2

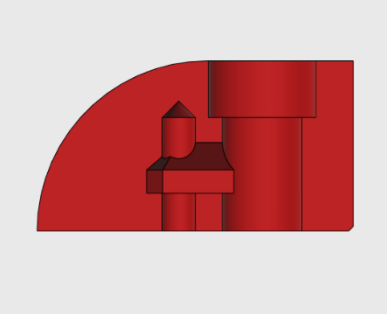


File name: Bras.stl

PLA red

Quantity: 2

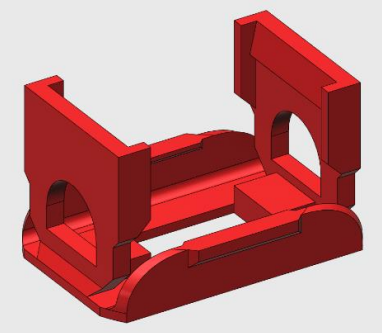




File name: Pied.stl

PLA red

Quantity: 2

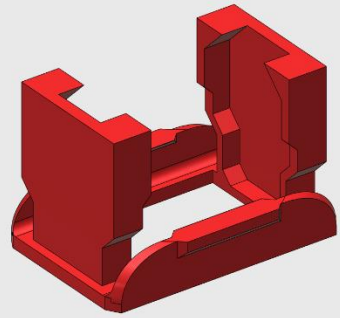


File name: TrappeSwitch-V2.stl

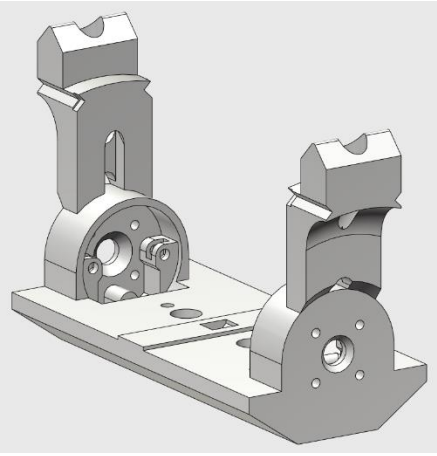
File name: TrappeSwitch-V3.stl

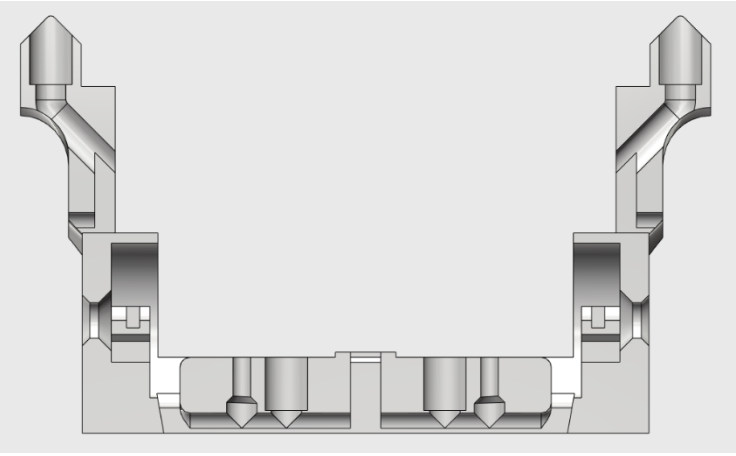
The choice depends of the switch type available

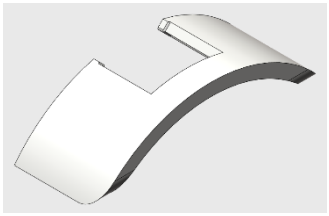
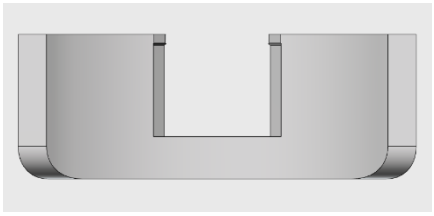
PLA red Quantity: 1



File name: GyroskateCorps.stl - PLA white - Quantity: 1



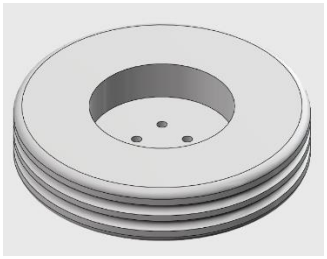
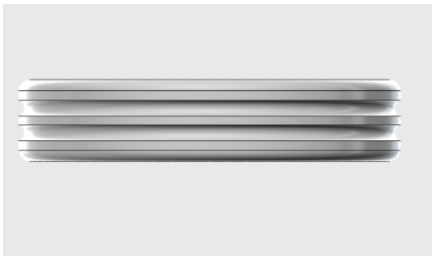


File name: GardeBoue.stl

PLA white

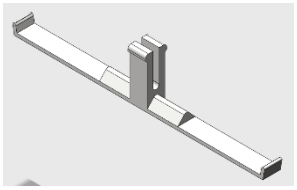
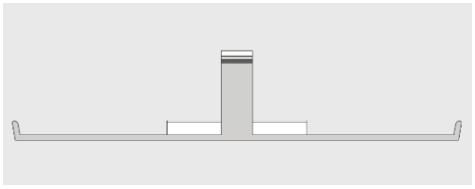
<u>Quantity: 2</u>

File name: Roue.stl

TPU white

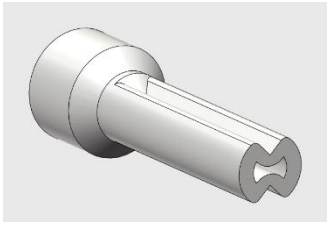
<u>Quantity: 2</u>

File name: CacheFils.stl

PLA white

<u>Quantity: 1</u>

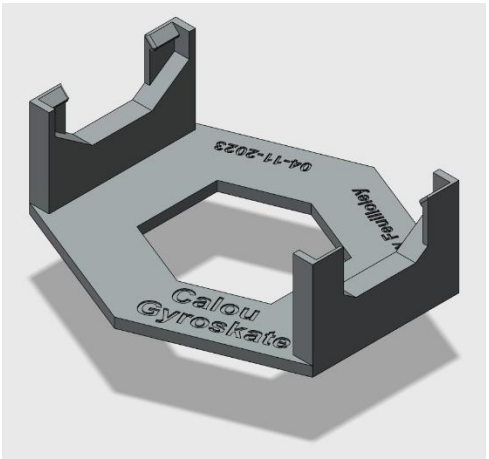


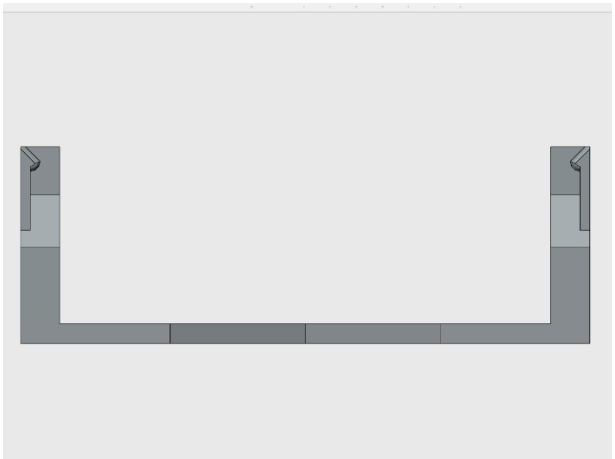

File name: AxeAimant.stl

PLA white

<u>Quantity: 2</u>

File name: Base.stl - PLA (free color) - <u>Quantity: 1</u>



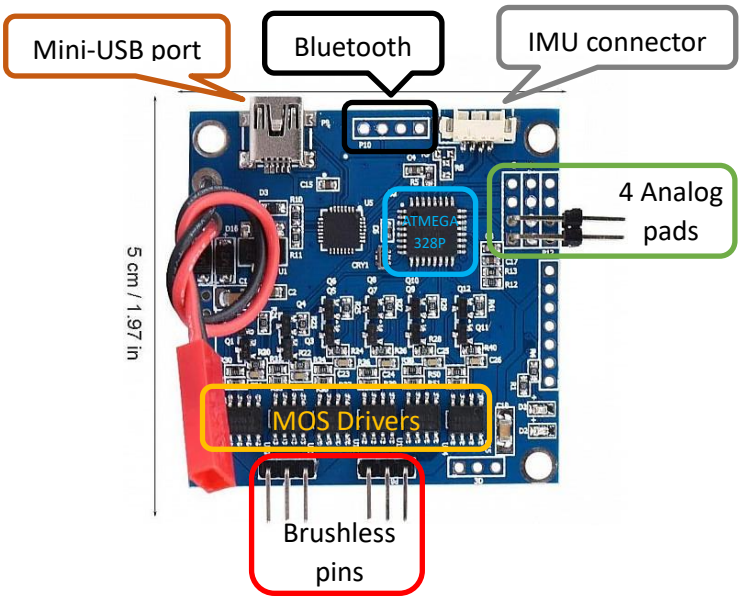


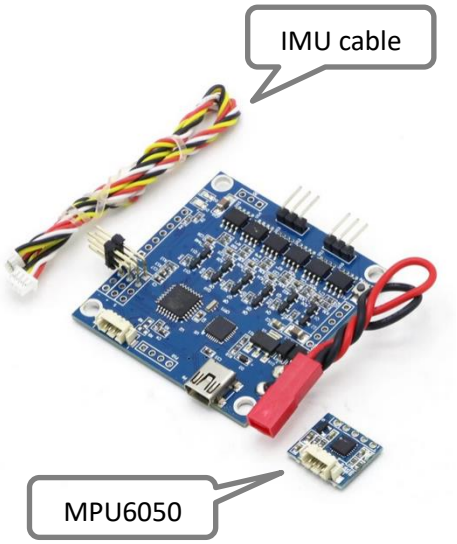
Electronic realization

Converting the Gimbal Controller in an Arduino compatible board:

The BGC3.1 MOSFET 2-axis board is provide with a mini USB port and an ATMEGA 328p microcontroller, the same as on the Arduino Pro Mini so it's possible to reprogram this processor without any problems.

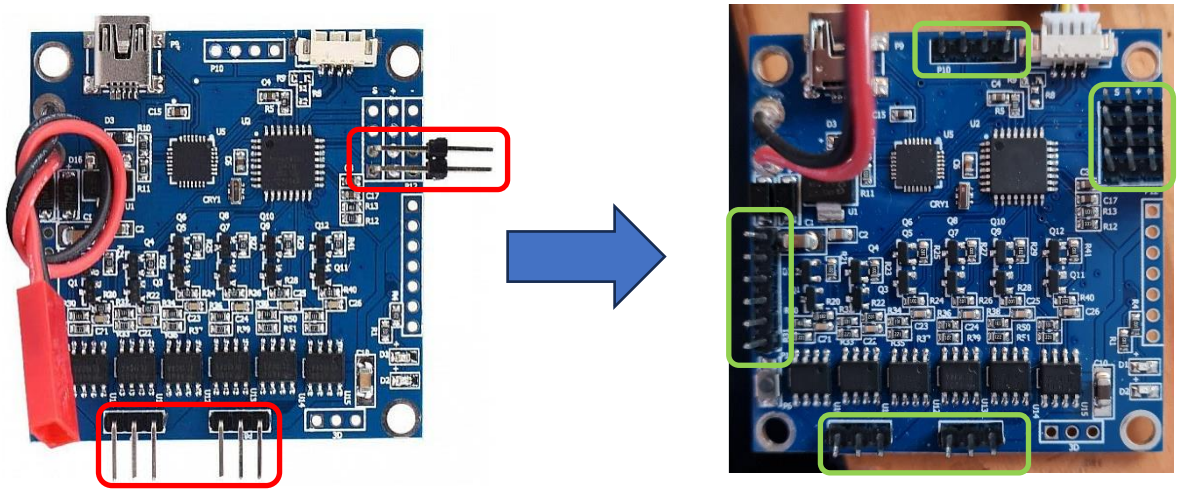
The main advantage of this board land in the connections already done with the MOS Drivers (5A continuous / 10A peak) for control two brushless engines and it is also delivered with an IMU gyroscope MPU6050 already wired and communicating by I2c. Moreover, on the right side we find 4 Analog pads easily accessible normally used by the drone remote and that will be reused as I/O port for connecting 2 angle magnetic encoders AS5600 and drive the LED antenna. On the top we have 4 other free pads that will be used for connecting the Bluetooth module. Electrical supplies are also embedded.





- Modifying steps

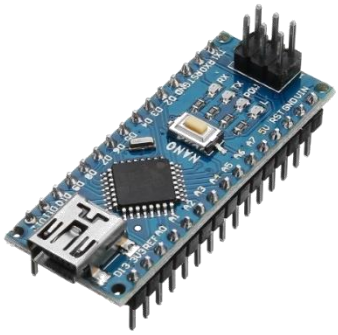
a. Remove the current right angle 2.54mm pin headers (circled in red) and weld new straight mounting pin headers on all the slots circled in green.



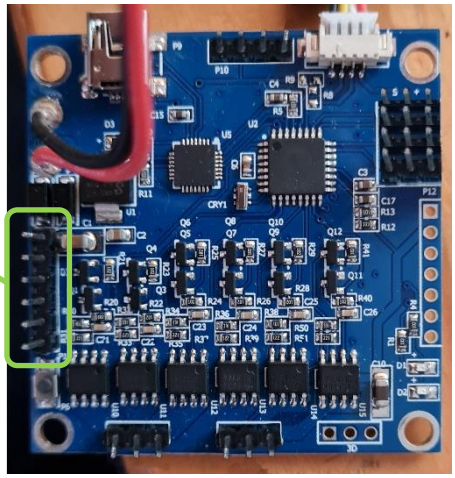
The second action consist in Flashing the Arduino Bootloader.

For this we will be using another Arduino board that will function as a circuit programmer through the ISP port of the Gimbal Controller board (here I took an Arduino nano with the same type of mini-USB connector).

Some general information about how to program an Arduino as an ISP programmer can be find here: **Arduino as ISP and Arduino Bootloaders**

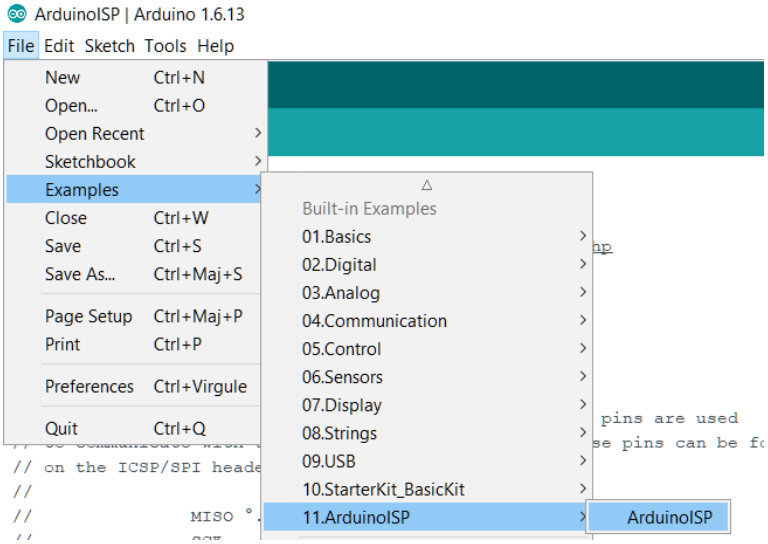


Back of the ISP port



In short you must follow those steps:

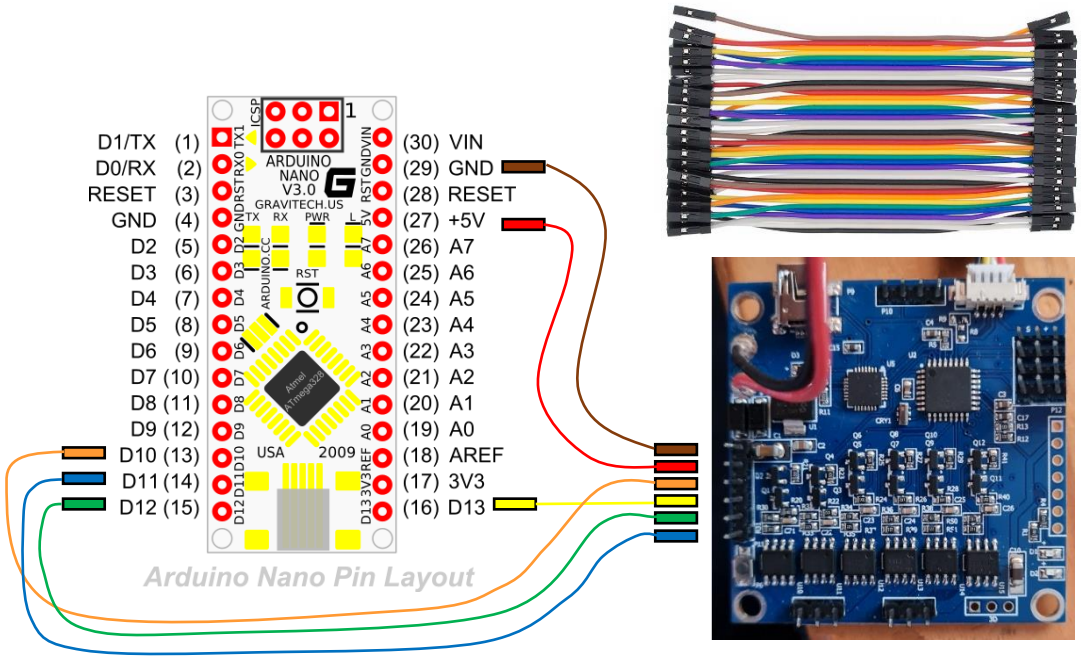
b. Upload this sketch into the Arduino before connecting it to the Gimbal Controller board.
The Arduino that you will use as programmer needs a specific sketch. You find it under Examples > 11. ArduinoISP > ArduinoISP .



- Upload the sketch

c. Connect the ISP port pins of the Gimbal Controller board to the Arduino nano board.

You will need of Female-to-Female Jumper wires Dupont 2.54mm and knowing the pin diagram of your Arduino board. (here after is the diagram for the Arduino nano board)



All the wiring must be done power off. Don't plug the USB's.

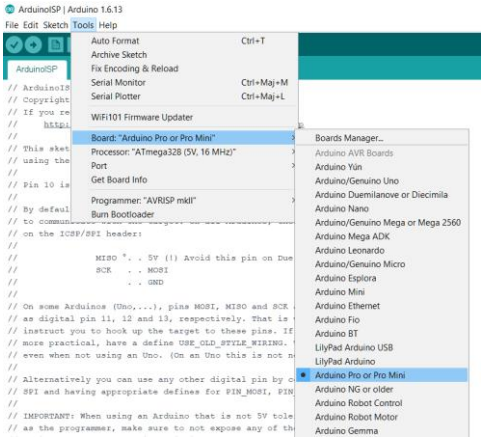
- Connect each individual wire on the ISP port of the Gimbal Controller board.
- Connect the other ends to the pins of the Arduino in respect of the connection table:
  (the colors for the connections are just to help you but don't have any electrical functions)

Arduino nano	BGC3.1 board
GND	GND
5V	+5V
D10	RES
D13	SCK
D12	MISO
D11	MOSI

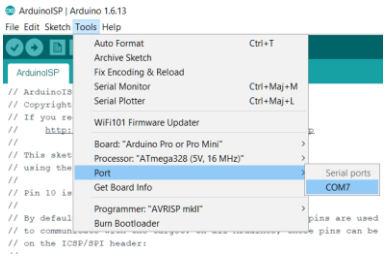
- Once the links done you can connect the USB of the Arduino nano to the computer. The Gimbal microcontroller ATMEGA 328p will be supply through the Arduino during the uploading.

d. Flash the Arduino bootloader

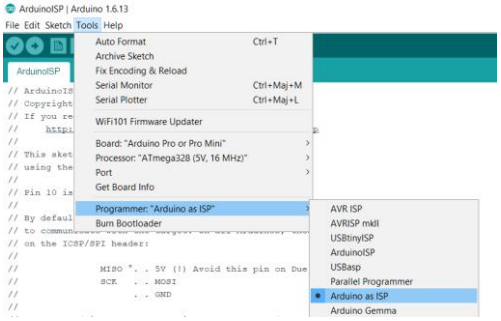
- Go to: Tools > board and select the Arduino pro mini 5V / 16MHz



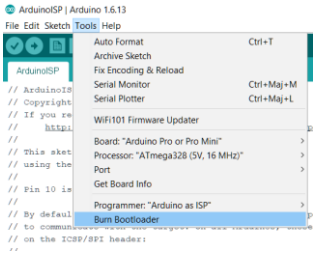
- Go to: Tools > Serial Port and select the COM port where your Arduino nano is connected.



- Go to: Tools > Programmer and select “Arduino as ISP”



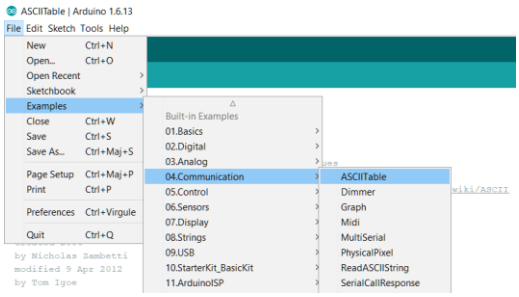
- Go to: Tools > Burn Bootloader



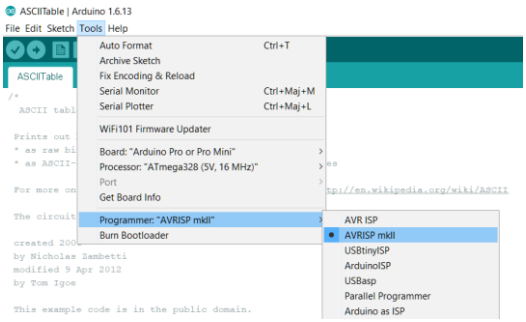
You will need to wait a while and after the message "Done burning bootloader" appears.

e. Configure the BGC board as an Arduino and proceed to a check test.

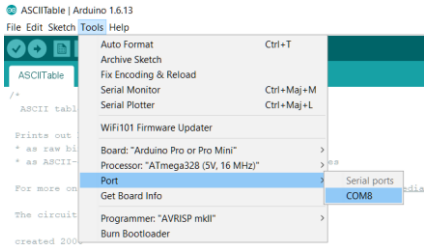
- Disconnect the USB link between the Arduino and the computer.
- Remove all the Jumper Wires. You will no longer need of the Arduino board.
- Connect the BGC board through its USB plug to the computer.
- Go to: Examples > 04. Communication > ASCIITable . (It is a program just for test the board)



- Go to: Tools > Programmer and select “AVRISP mkII”



- Go to: Tools > Serial Port and select the new COM port where your BGC board is connected.

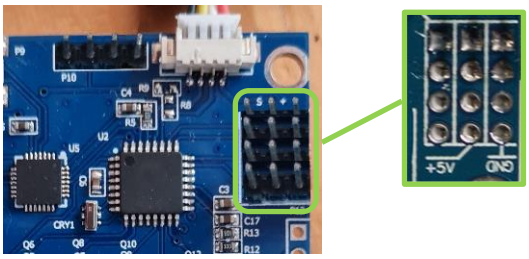


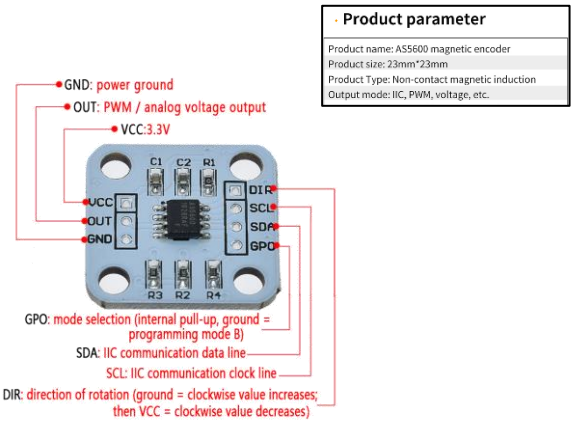
- Upload the sketch
- Open the serial monitor and select the right communication speed “9600 baud”

You should see a scrolling of all the ASCII code meaning that your BGC board has been right configured as an Arduino pro mini 5V / 16MHz.

Adapt the Magnetic Encoder Module for the used:

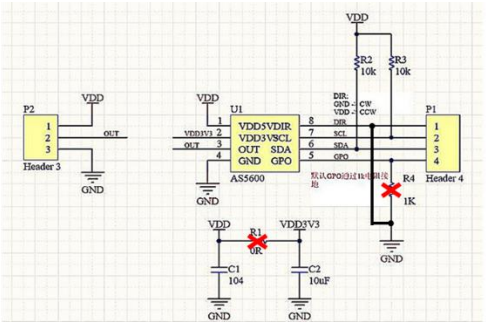
Those type of magnetic encoders are cheap. However, this version was configured to work in 3.3V DC while we have 5V delivered on the 4 pads I/O on the controlling board. Additionally, we need to select a rotation direction as well as the analog voltage output mode.

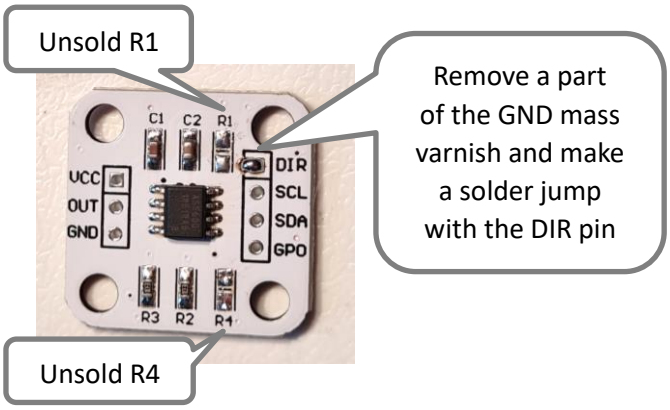




You would need to adapt the two magnetic encoder boards as describe here below.

The schematic and the picture on right show the changes to be made:





- Remove the R1 resistor,
- Remove the R4 resistor,
- Make a short between DIR pin and the ground.

The external connection on VCC/OUT/GND will be done once the servo extension cables installed into the robot.

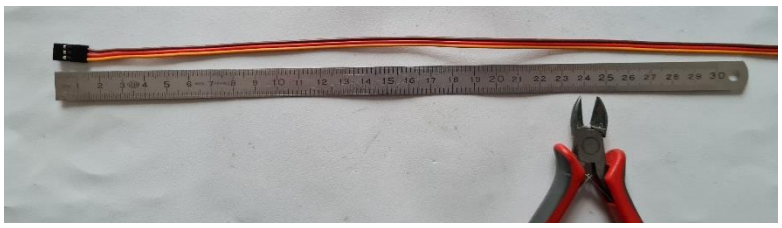
Assembling of the robot

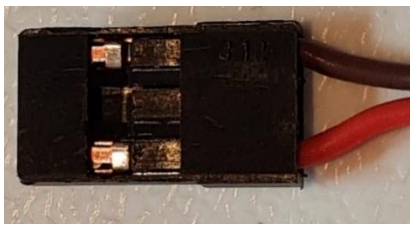
Antenna:

1 x Extension de Cable 3Pin - JR Style
1 x White LED
1 x Resistor 100Ω 1/4W
3D Parts: Antenne; CaleLedAntenne; JointLed

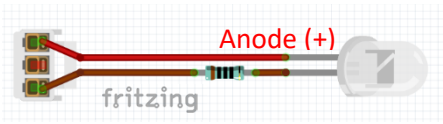
Connect the white LED:

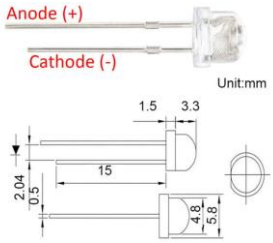
- Start by cutting the extension cable to keep the Male JR plug and around 24cm of cable.
- Separate the Orange wire from the strip to keep only the Brown and Red.
- Unplug the Orange wire terminal from the JR connector with caution to don't break the small plastic tongue. I recommend taking a thin screwdriver and slowly lifting it until the connector pin comes out. Move the terminal of the Red wire to the location of the Orange wire. (See pictures below)





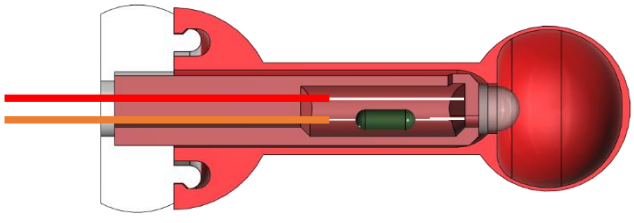
- Weld the LED and the resistor this way. Insulate the resistor with a heat shrink sleeve.





Assemble the antenna:

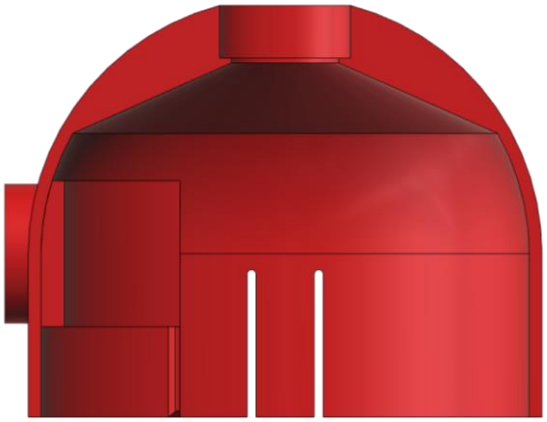
- Insert the LED through the hole into the part "JointLed";
- Slide the "CaleLedAntenna" between the resistor and the LED;
- Insert the "CaleLedAntenne" into the "Antenne" part;
- Clip the "JointLed" on the "Antenne" part by pushing on the base. It can be a little difficult to insert. So, you can use some grease to help if needed.



Calou Head:

3D Parts: 1 x Tête; 1 x Yeux; + The assembled Antenna

- Start by insert the eyes. If it well fits you should be able to remove it for gluing the parts together. I used a cyanoacrylate gel glue for avoid the drops.

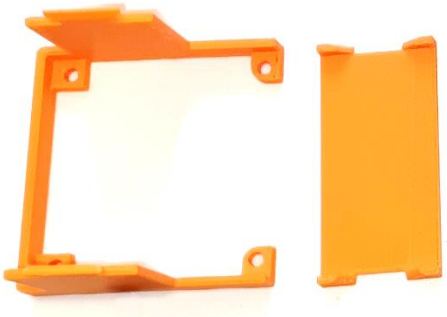


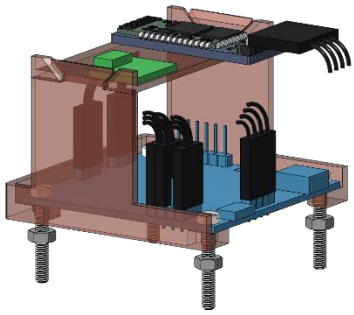
- Once the parts are well assembled, you can mount the antenna in the top housing.

Electronic holder frame:

3D Parts: 1 x SupportBGCboard; 1 x SupportIMU&BLE

- Glue and clip the two parts. There is only one assembling position.





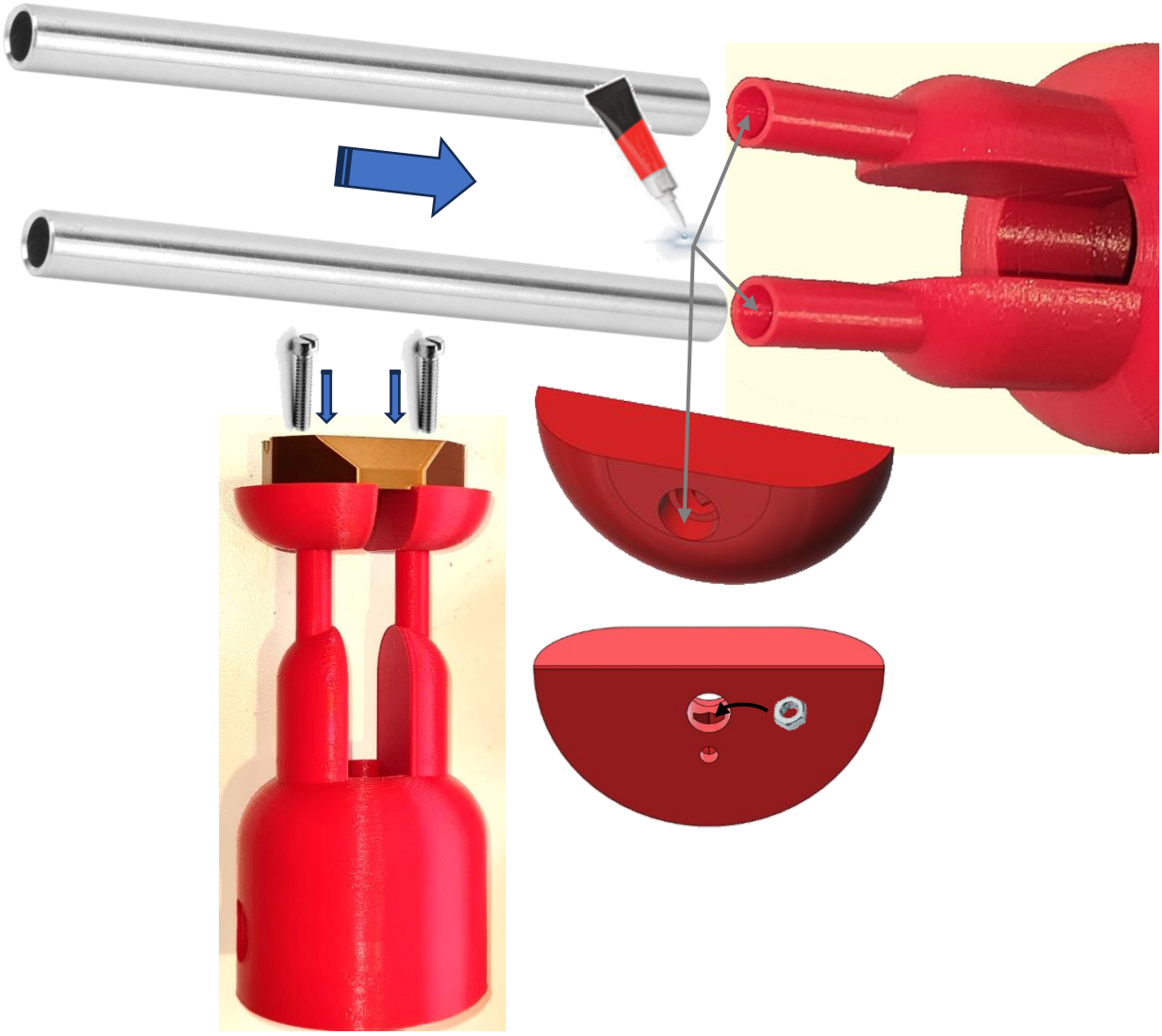
Low body:

2 x Aluminum Leg Tubes 100mm; 2 x Screws (M3x20mm); 2 Nuts M3
3D Parts: 2 x Pied ; 1 x CorpsBas-V2 ; 1 x GyroskateCorps

The goal is to well align the feet with the legs for sticking them together. You can use the Gyroskate body or a dedicated part that I drawn to don't pollute the Gyroskate with the glue drops.

This part is in the list of 3D parts, but it still an option for the assembling: FootSpacer.slt

a. Put some glue into the legs for the sticking of the tubes (grey arrows);
b. Insert the tubes into the legs until it was stopped inside the body;
c. Insert one nut into each foot orifice (you can use tweezers or similar to do it);
d. Put some glue in each cylindrical shape of the foot where the leg will be fixed (grey arrows);
e. Insert each foot the long of the tube and then the legs until it was stopped. Once the feet assembled the nuts can't be removed, so it must well located!
f. Fix and screw the "FootSpacer" in order to get the two feet well positioned;
g. Put the assembly vertical the time for the parts to be well glued;
h. You can then remove the screws and the "FootSpacer".

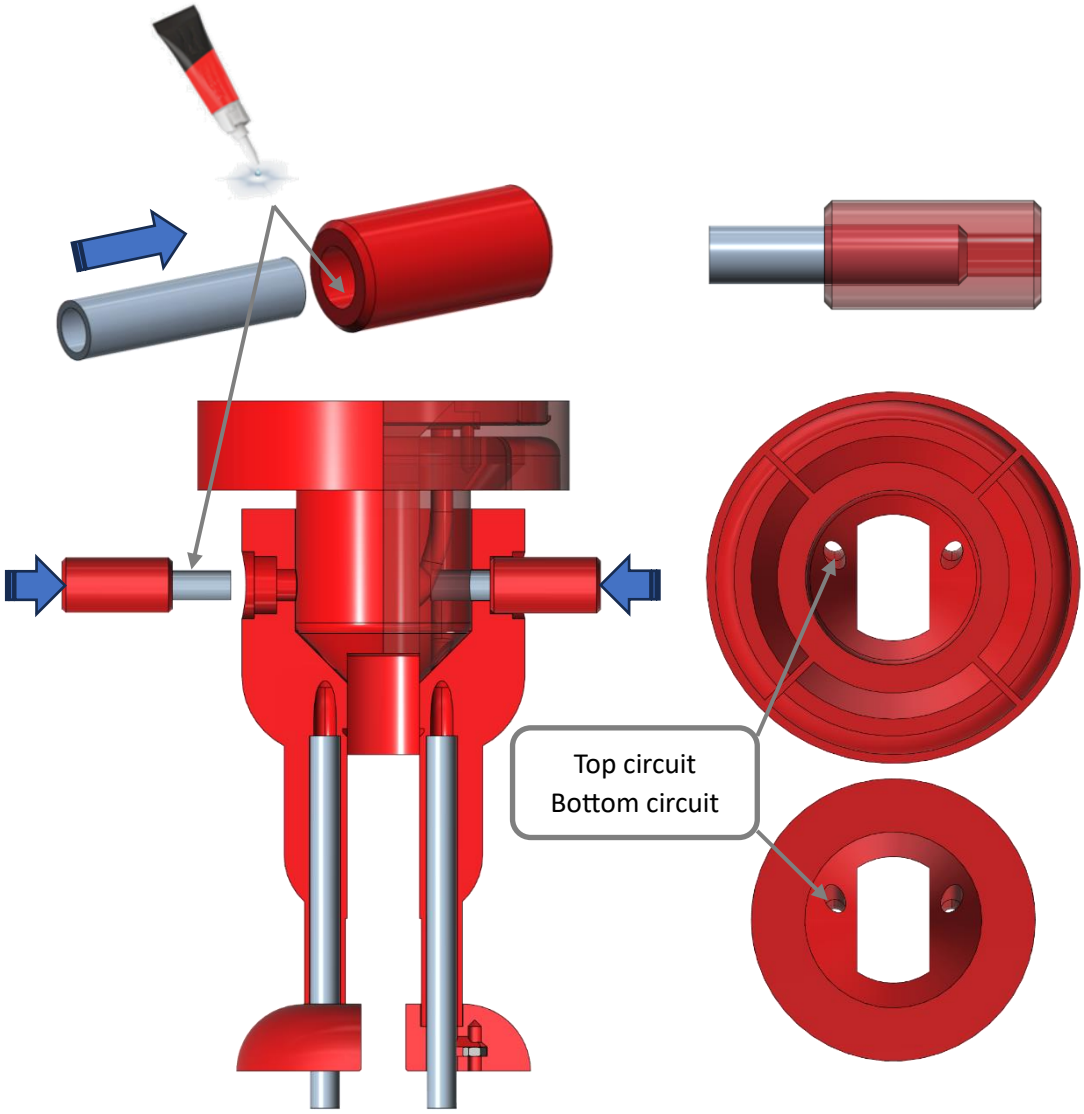


Body and Forearm:

2 x Aluminum shoulder Tubes 34mm
3D Parts : 1 x CorpsHaut; 2 x AvBras; The assembled low Body

We are going to assemble the top and bottom Body with the Forearms.

a. Glue the tubes into the cylindrical parts “AvBras”;
b. Assemble the top and low bodies. Pay attention to the orientation to properly align the circuits of the upper and lower encoder cables.
c. Glue the external side of the tube and insert the two forearms until it was stopped to lock the parts together.

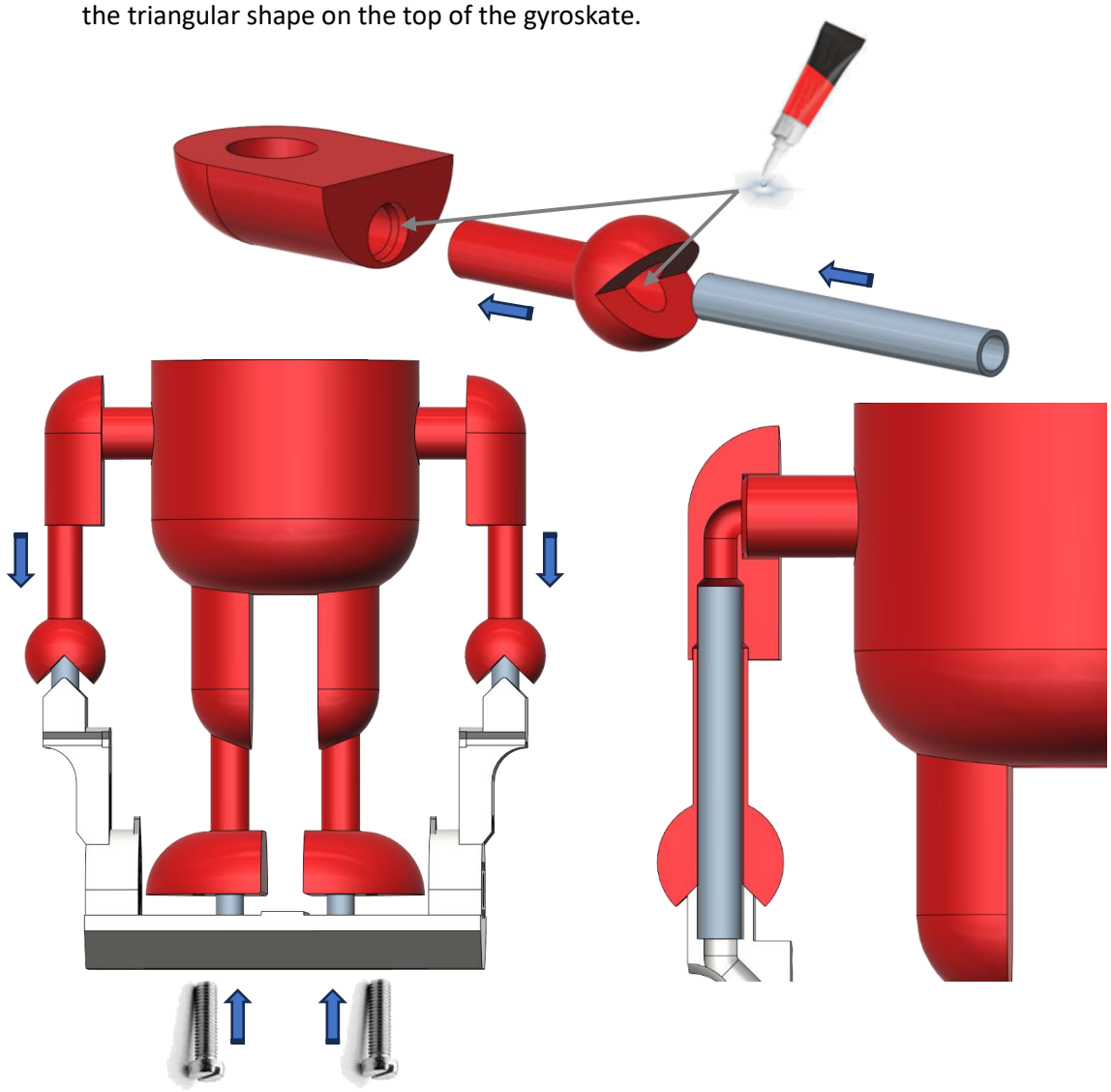


Arms:

2 x Arm Tubes 64mm; 2 x Screws (M3x20mm)
3D Parts : 2 x Bras; 2 x Epaule ; 1 x GyroskateCorps ; The assembled body

We are going to assemble both arms. The arms are locked by the mounting of the body with the gyroskate, so they can be removed to make easy the wiring. The operation described here below must be done as fast as possible before the full sticking.

a. Put some glue in each cylindrical shape of the shoulder where the arm will be fixed (grey arrows);
b. Put some glue into the arm for the sticking of the tube with the arm (grey arrows);
c. Assemble the arm into the shoulder and then the tube into the arm until it was stopped inside the shoulder;
d. Take the body and insert the shoulders into the forearms. Keep it in place with your hand. Take the gyroskate and fix the body and the arms on the gyroskate by inserting the tubes;
e. Screw the feets;
f. Take care that shoulders are well aligned on the body and the robot's hands are well placed on the triangular shape on the top of the gyroskate.



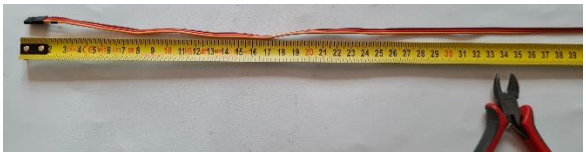
Extension wiring:

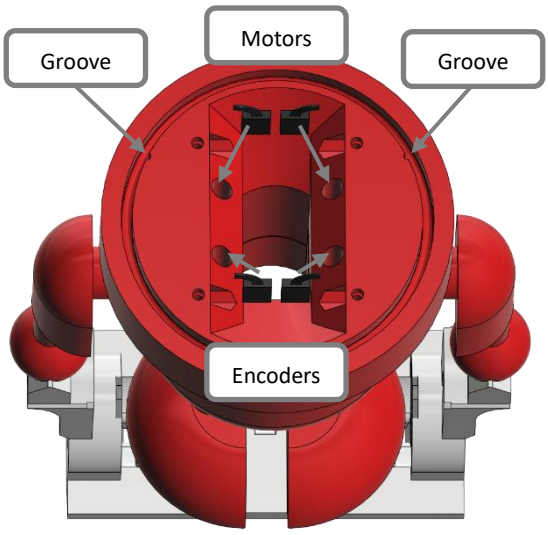
4 X Extensions de Câble Servos 50cm
3D Parts : The assembled body

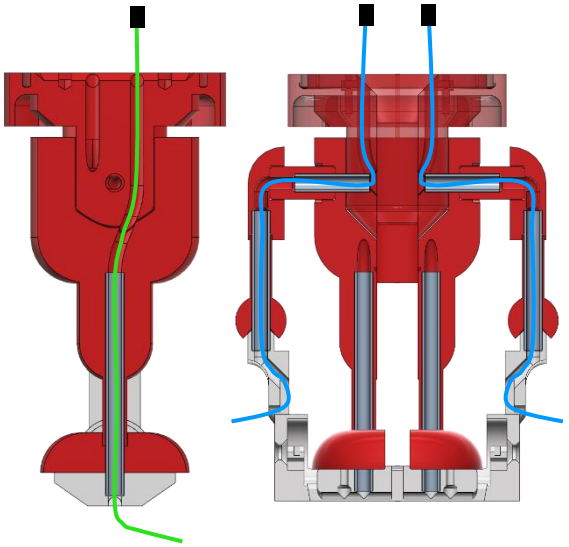
- Start by cutting 2 extension cables to keep the Male JR plug and around 34cm of cable. Those two cables will be the one used to supply the motors.
- Cut the 2 others to keep the Male JR plug and around 38cm of cable. Those two cables will be the one used to connect the magnetic encoders.
- Insert the encoder cables through the top of the robot. Look at the drawing here below to identify the right cable circuits. Once installed the cables must exit from the feet.

Installing the motor cables requires more steps.

- Start by separate the gyroskate from the arms.
- Insert the motor cables through the top of the robot. Once exit through the center of the battery holder you must insert the cable into the forearms tubes. (You can use tweezers or similar to do it);
- Insert all the remaining cable into the shoulder and let it exit through the robot's arms. Put in place the shoulders on the forearms.
- Insert the cables into the respective circuits of the Gyroskate and fix the body and the arms by inserting the tubes inside. (Use the drawings here below to help you)







Top View

Top cable circuits

Side View

Encoder cables circuit

Front View

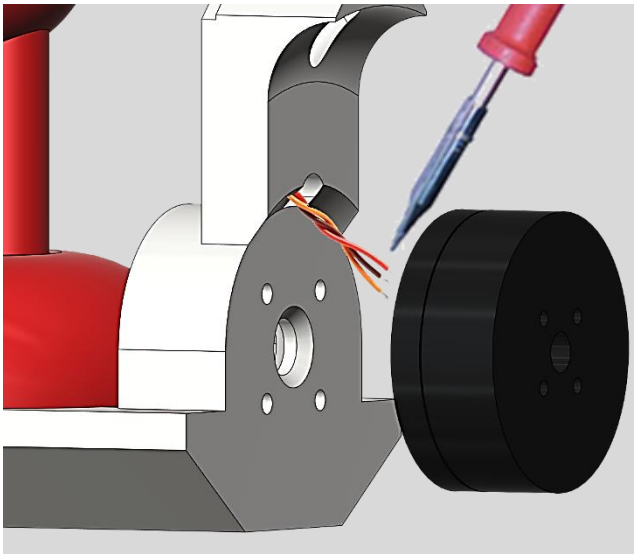
Motor cables circuit

Connect and mount the motors:

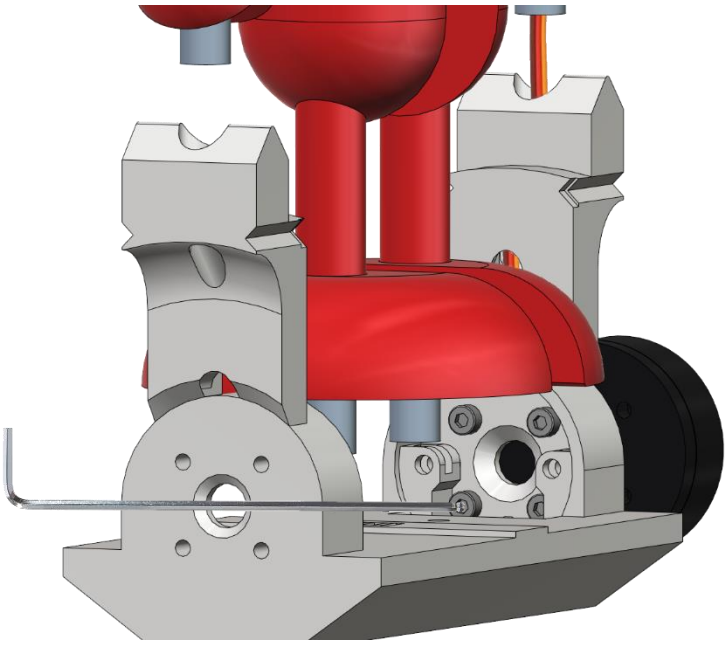
2 X Motors 2804 100KV ; 8 x Screws M2.5x8mm
3D Parts: The assembled robot

We are going to weld the motors to the extension cables and fix them to the gyroskate.

- Unsolder the wires from the motor circled in red and remove the cable. (Pay attention don't heat up too much the pads for don’t damage the motors)
- Weld to the motor the free wires of the extension cable standing on the exit of the gyroskate. Don’t care to the wire’s color because the phase detection will be done later by software.



- Fix the motors on the gyroskate by using the 4 screws M2,5. You would need a 2.5mm Allen wrench with ball end because the access is relatively tricky.

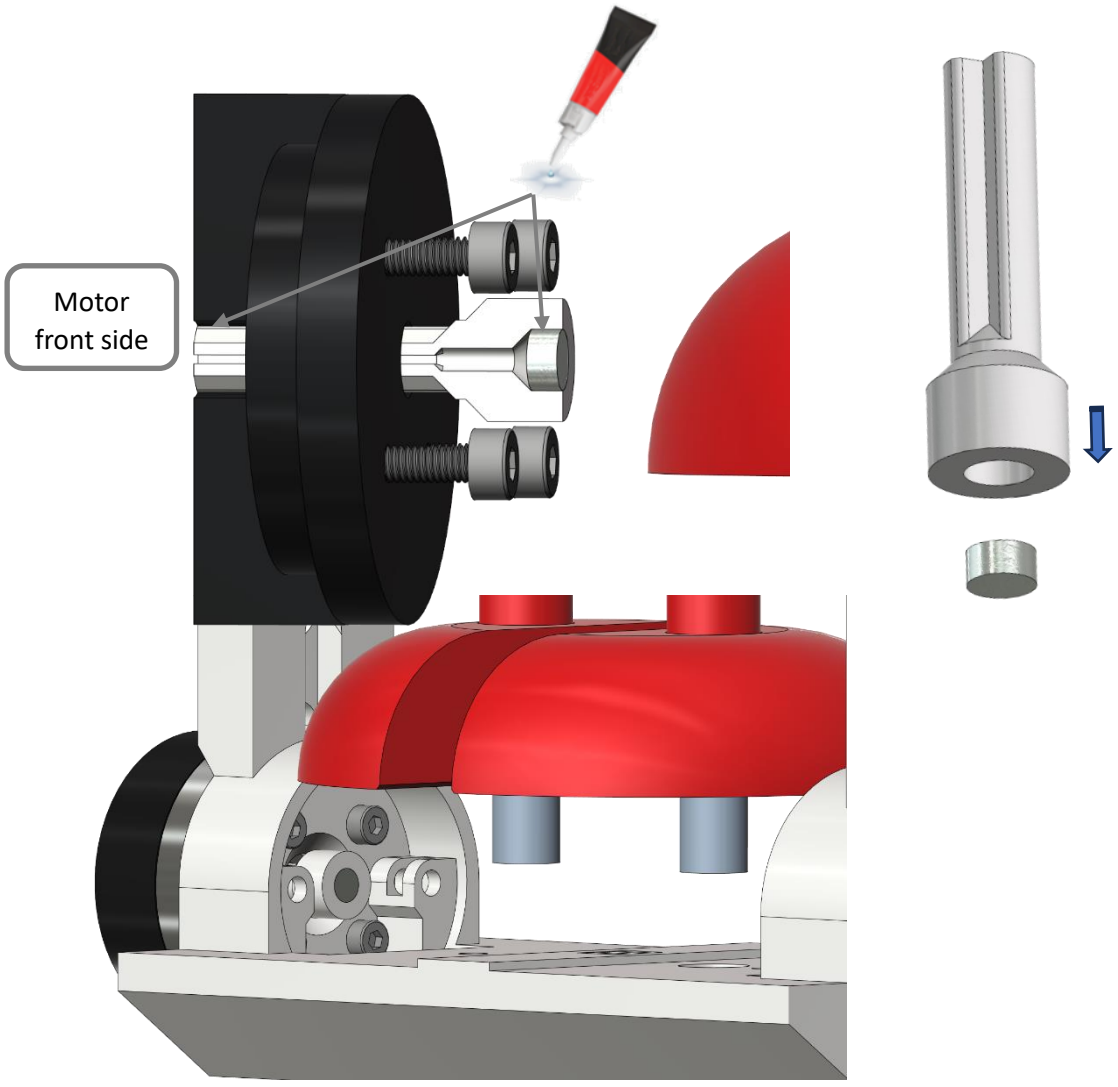


Assemble and mount the magnets:

2 x Magnets Ø4mm x 2mm (usually delivered with the encoder modules)
3D Parts: 2 x AxeAimant ; The assembled robot

Magnets must be well positioned for a good angular reading. A typical distance of 0.5 – 3mm is needed between the magnet and the magnetic sensor. In that case I have put 1.5mm.
Each holder is inserted inside the central core of the motor. Some precaution is required to don't deform the thin part during the assembly to get a good alignment.

- Put a droplet of glue in the magnet holder to guaranty it won't move.
- Put the magnet onto the table and push hard on the round end of the holder to insert it until it reaches the table too.
- Put a droplet of glue into the central core of the motor on the wheel side.
- Insert the magnet holder until it reaches the front side of the motor. Care to push into the motor axis to don't deform the holder.
- Turn the motor to check if the magnet rotates without friction or mechanical resistance.



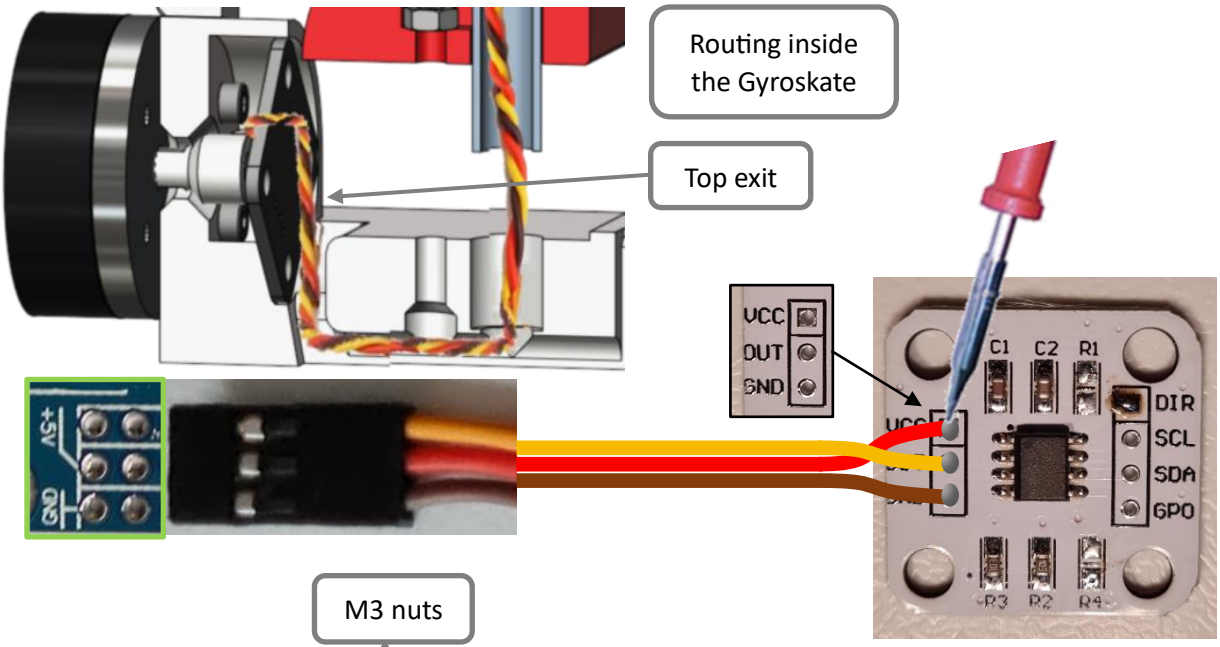
Connect and mount the encoders:

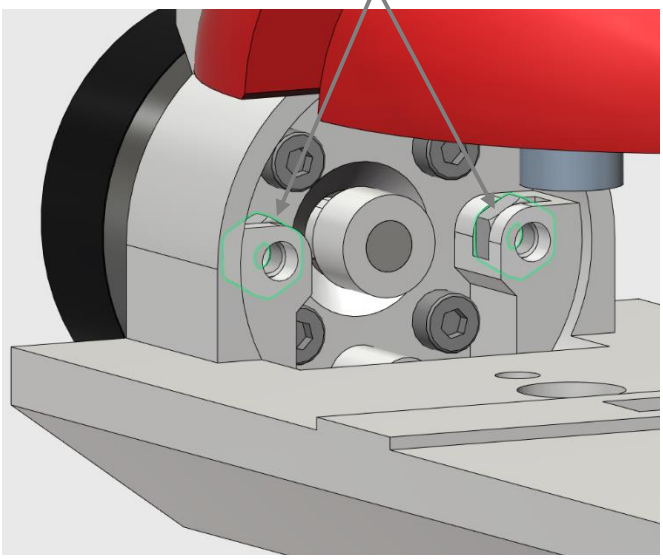
2 X Encoder modules AS5600; 4 x Flat countersunk head Screws M3x8mm; 4 Nuts M3
3D Parts: The assembled robot

The encoders must be fix in front of the magnets. For improve the integration I choose to weld the extension cable directly to the connecting pins of the encoder without intermediate connectors. So, the soldering is done on component side to make it the less visible. However, it is also optionally possible to weld a 3 pin straight mounting pin header for facilitate the interface with the wires.

Be warned to the color match of the wires with the inscriptions on the module!

- Route the cables into the Gyroskate in order to make it exit from the top.
- Weld each wires on the encoders. (Use the drawings here below for the color match)
- Insert two M3 nuts into the encoder brackets.
- Fix each encoder with two flat countersunk head screws M3x8mm



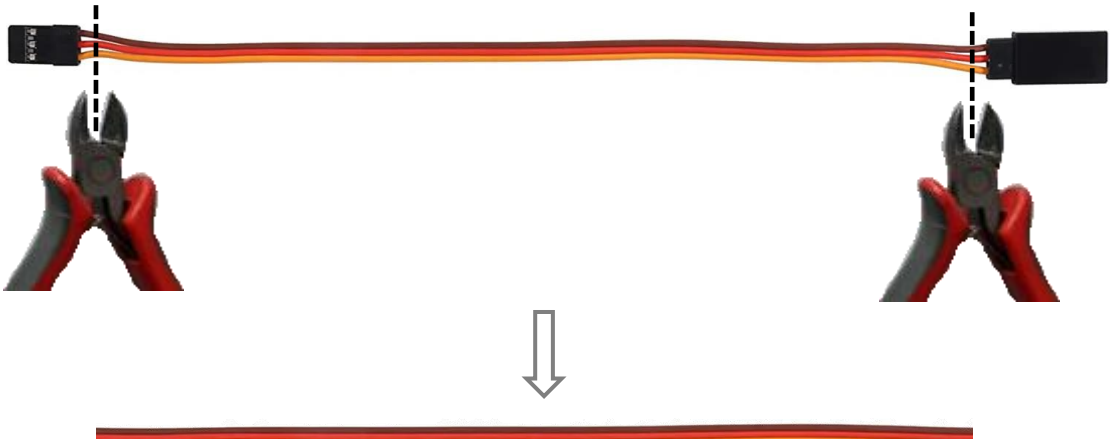


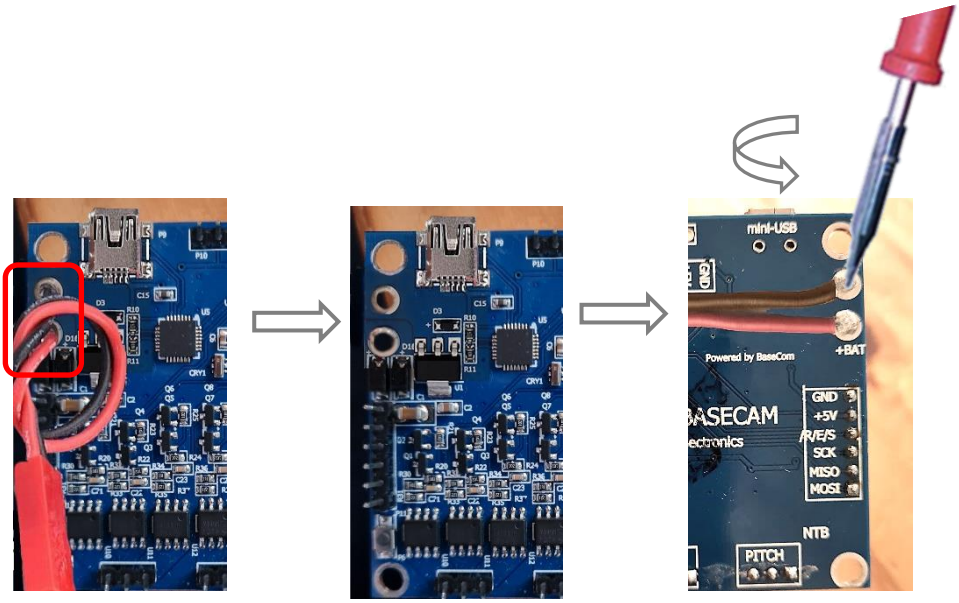
Supply of the board:

1 Extension de Câble 3Pin ; The BGC board;

The supply cable plug of the BGC board must be removed and replaced by a longer cable that will be connect later to a switch. I used one of the 22AWG extension cables, but you can take any compatible cable able to safely handle 7 Amps in continuous.

- Start by cutting both JR plugs of the cable and separate the orange wire from the strip to keep only the brown and red;
- Unsolder the JST plug wires from the BGC board (circled in red). It will be reused later;
- Weld the new cable on the back of the BGC board by respecting the followings color code: (+BAT → Red; GND → Brown)



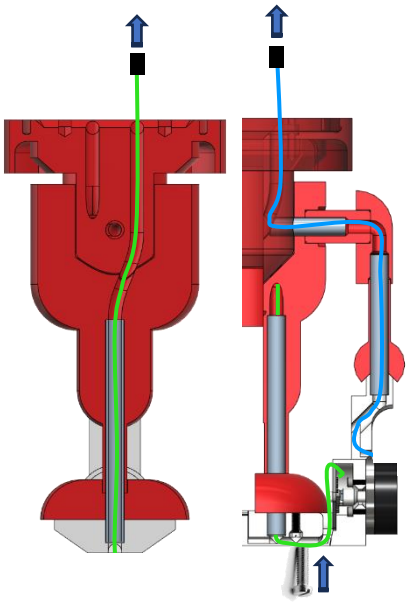


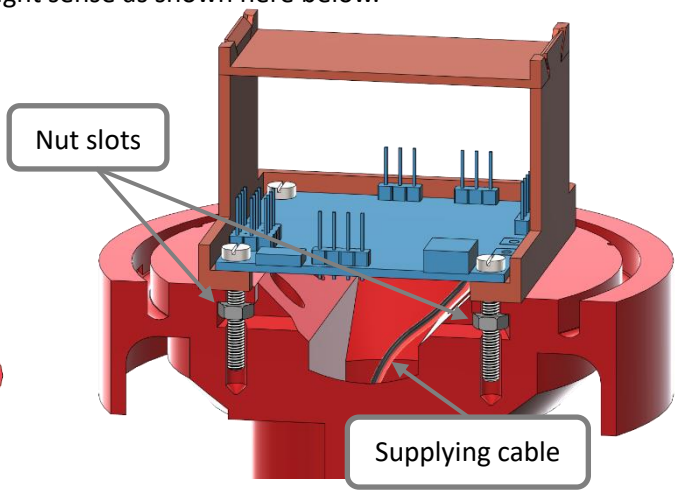
Assemble the command electronic board and connect:

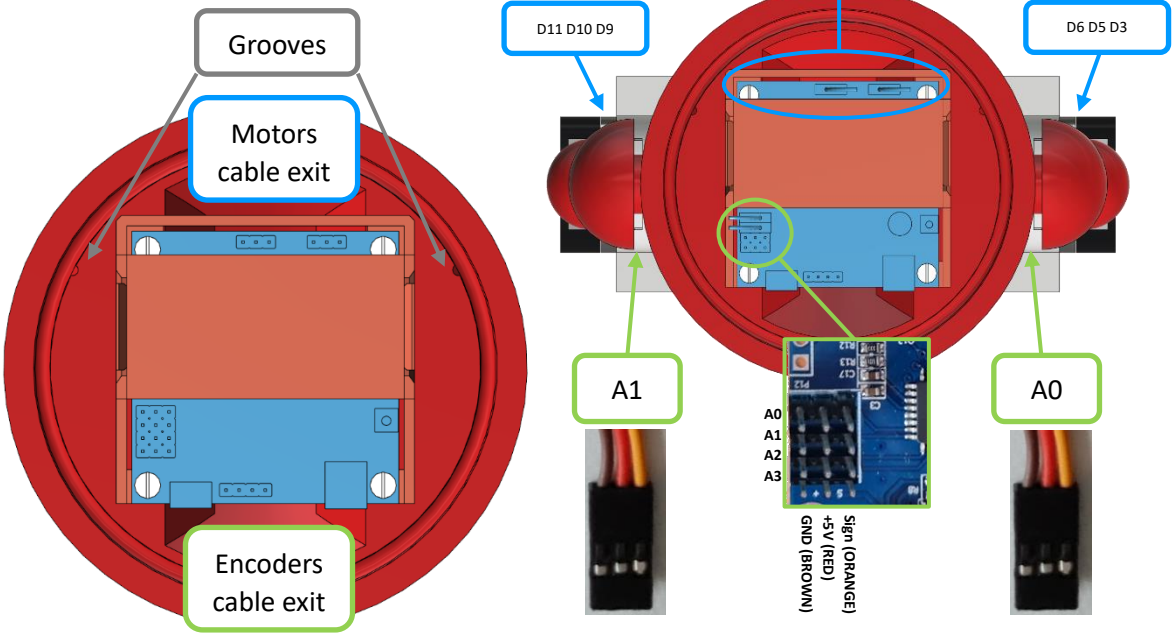
4 x Screws (M3x20mm); 4 Nuts M3; The BGC board
3D Parts: The Electronic holder frame; The assembled robot

You will proceed to the mounting the BGC board and connect the Male JR plugs.

a. Assemble definitively the body on the Gyroskate by inserting the tubes and screwing the feet.
b. Reduce the over lengths of cables by stretching on it with caution through the top.
c. Insert 4 x M3 nuts into the slots on the top Body part;
d. Screw the BGC board with the electronic frame by taking care of the orientation. There are two grooves for align the head that will help you. Look at the location of the cable's exit. The supplying cable is being the sole that must go through the center;
e. Connect the Male JR plugs in the right sense as shown here below.





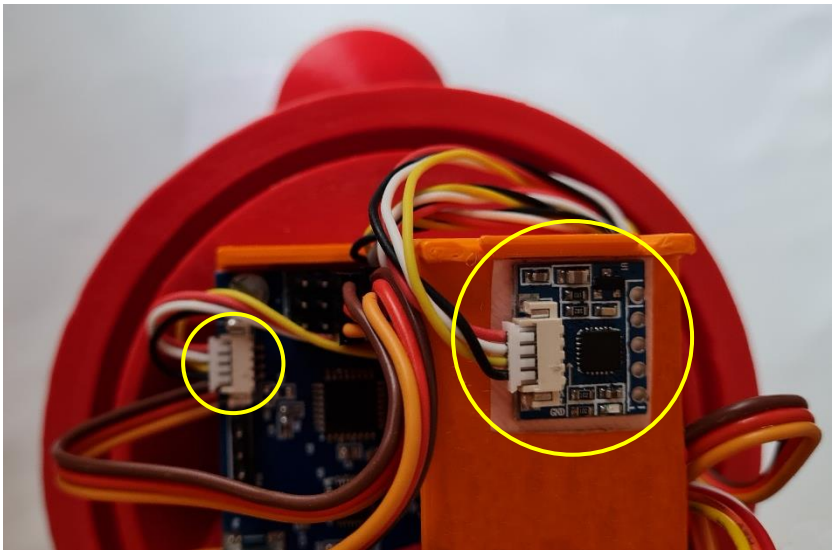


Fix the IMU and connect:

1 x MPU6050 gyroscopic sensor (usually delivered with the BGC board)
3D Parts: The assembled robot

The module that I had was delivered with a double face tape. We will fix it on the top of the electronic holder frame. You need to respect the mounting direction that is very important for the algorithm.

a. Stick the IMU with the connector orientate in the same direction than the one on the command board;
b. Enroll the extra length of cable around the holder frame wall;
c. Connect the cable on the command board.

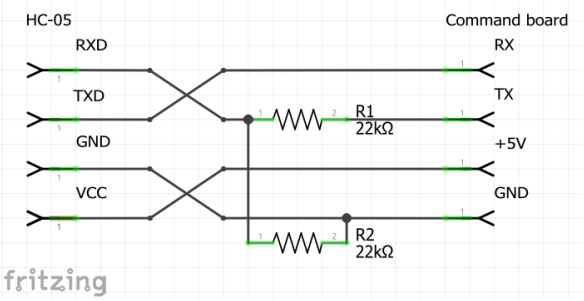


Connect and fix the Bluetooth module:

1 x HC-05 module (6Pin); 1 x Strip of 4 Female to Female wire Dupont 2,54mm – 10cm long.
3D Parts: The assembled robot

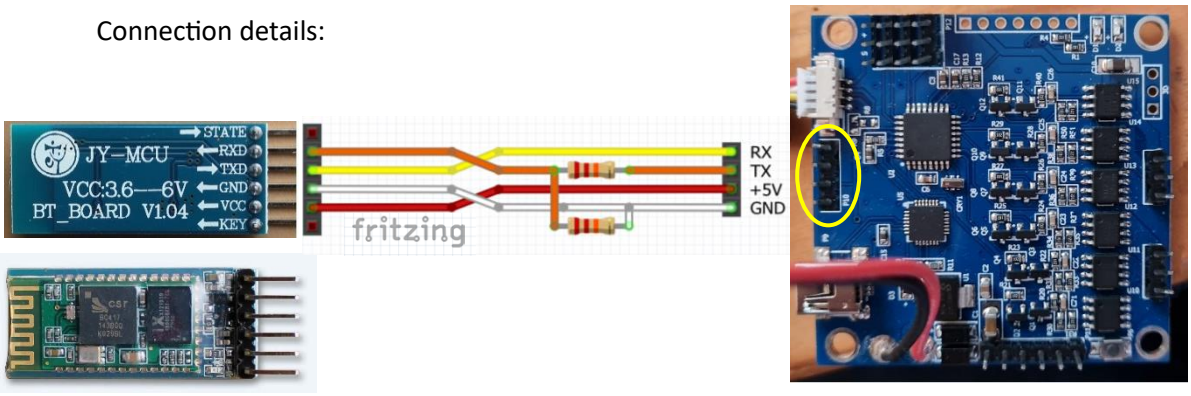
The HC-05 module can be powered from 3.6V to 6V, so it is compliant with the 5V delivered by the command board, but it has a 3.3V logic level voltage. To avoid any damage of the module, a resistor voltage divider from the Arduino TX pin to the HC-05 RX pin is needed to lower the High level from the Arduino to a compatible logic level. This manner data can be transferred safely to the Bluetooth Module.

This is the connecting diagram to apply between the command board and the Bluetooth module:

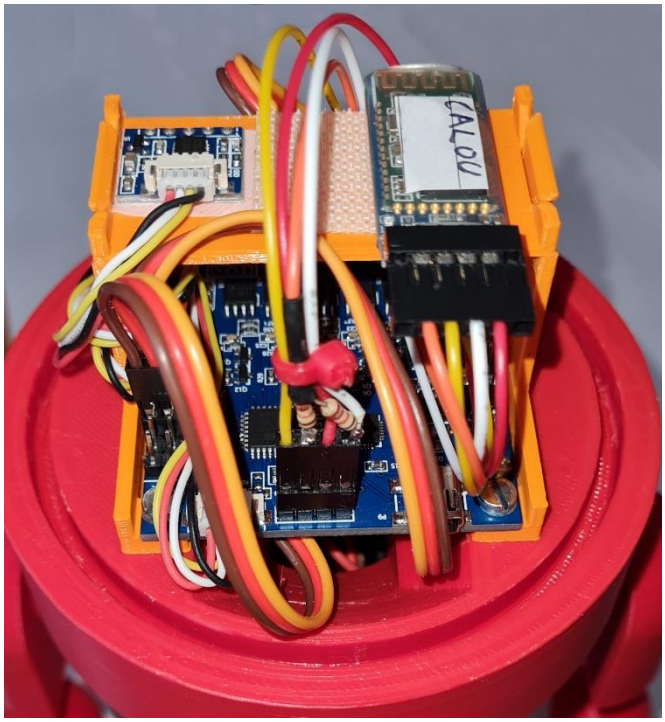


a. We need to manufacture a cable with this voltage level adaptation. You can use a flat cable if it was delivered with the module for modify it or take the 10cm jumpers available.

Connection details:



b. Stick a double face tape on the top of the holder frame and fix on it the module.
Here is an illustration of the mounting.



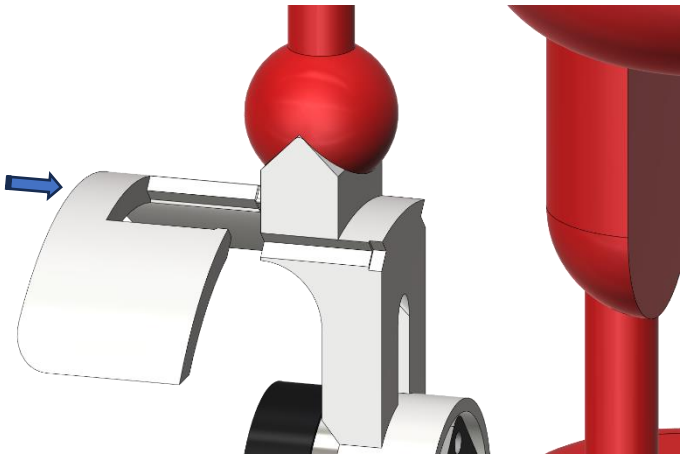
The module is connected to the serial port of the microcontroller that is already used by the FT232R circuit for the USB to serial interface of the command board. Connecting anything to these pins can interfere with that communication, including causing failed uploads to the board. So, it is very important to respect the followings:

<u>Disconnect the Bluetooth module each time you connect the command board to the computer through the USB cable!</u>

Fix the mudguards:

3D Parts: 2 x GardeBoue ; The assembled robot

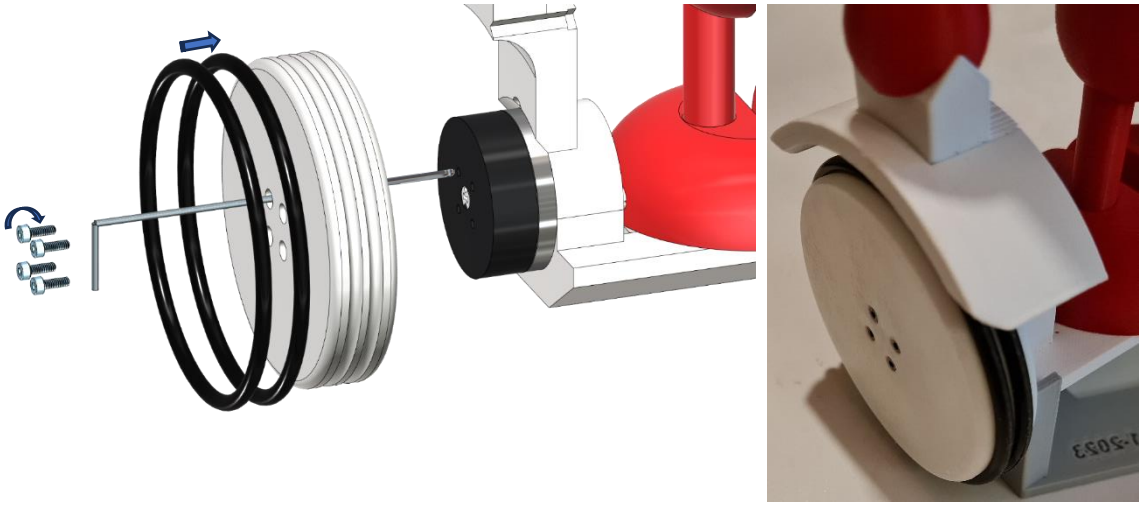
a. Insert each mudguard into the side slits of the Gyroskate until it was clips.



Assemble and fix the wheels:

8 x Screws (M2x6mm) usually delivered with the Motors 2804; 4 x O-Ring 75x68x3,5mm.
3D Parts: 2 x Roue; The assembled robot

a. Mount two O-Ring by wheel into the groves;
b. Fit the wheel on the engine by taking care the holes are well aligned with the threads. You can use the 2mm allen wrench for the alignment before screwing.
c. Do not apply too much torque on the screws as the wheels are printed in soft TPU.

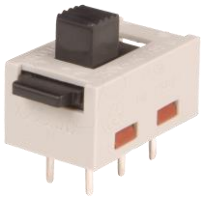


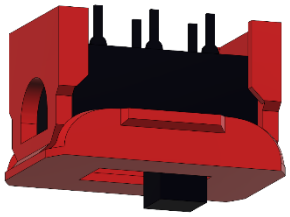
Switch assembly:

1 x JST plug cable (the one we have previously unsold); 1 x ON-OFF Switch
3D Parts: TrappeSwitch-V2 or TrappeSwitch-V3; The assembled robot.

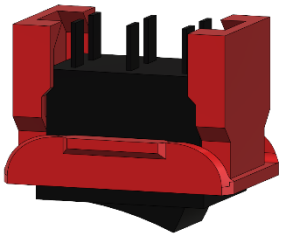
Two versions are proposed depending on the availability of the switch. The most important criteria are to get a double contact ON-OFF switch rated to 10A.

The first version was a slide switch X22205C from Arcolectric. This switch is compatible with the 3D part “TrappeSwitch-V2”, but it is not easy to find.

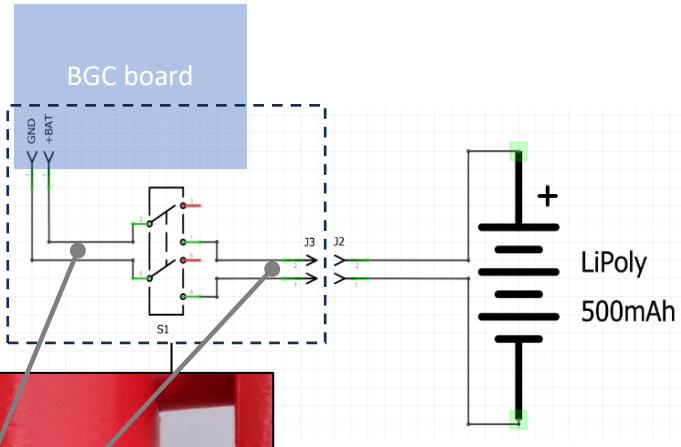




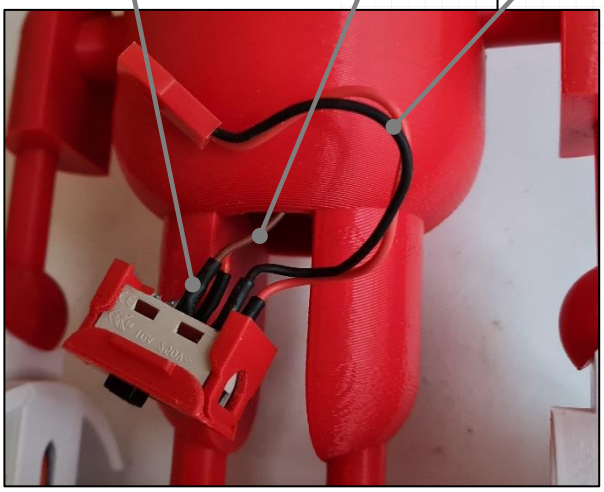
The second version was based on a rocker switch that looks more affordable. This switch is compatible with the 3D part “TrappeSwitch-V3”.



The wiring schematic is relatively simple.



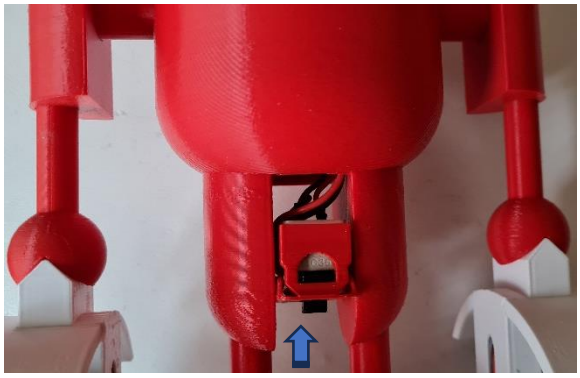
Do not forget to protect the contacts with a heat-shrinkable sleeve:

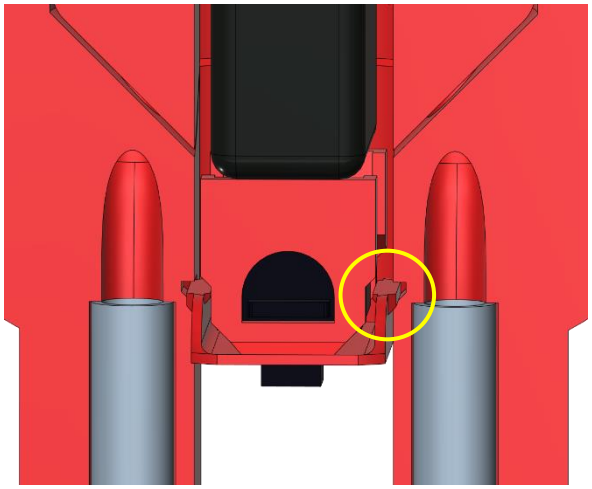


Battery installation:

1 x LiPo Battery 500mAh - 11,1v - (L x l x H = 58 x 32 x16 mm)
3D Parts: The assembled robot

a. Insert the battery (through the side without cables) into the central part of the robot;
b. Check the switch is OFF;
c. Connect the JST plug;
d. Insert the switch holder between the legs and push the full to the top until it is clipped.





Programming the robot

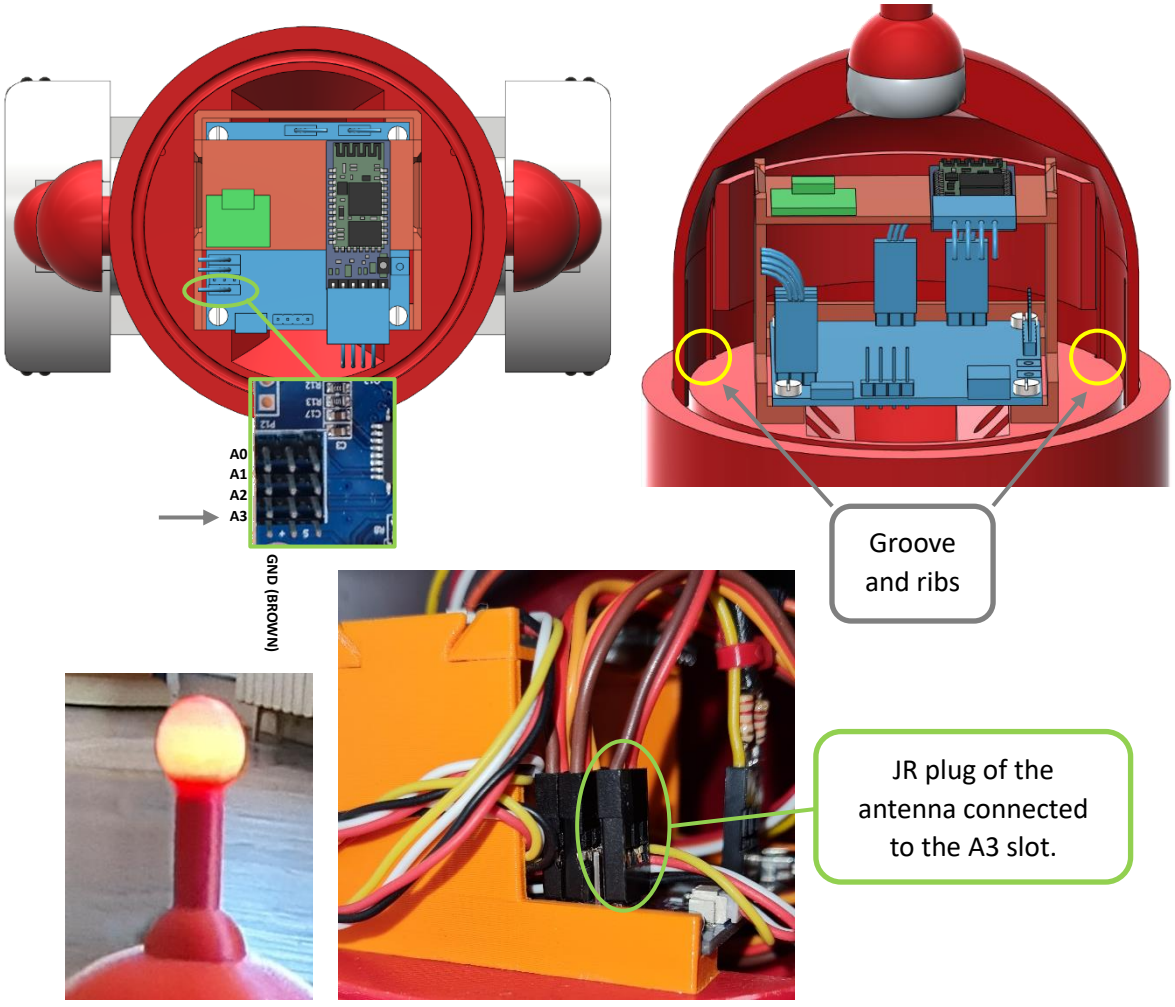
Test programs:

Before upload the main sketch it is highly recommended to test some of the functionalities.

- Test the antenna blinking;
- Test the magnetic sensor reading;
- Find sensor offset and direction of the motors;

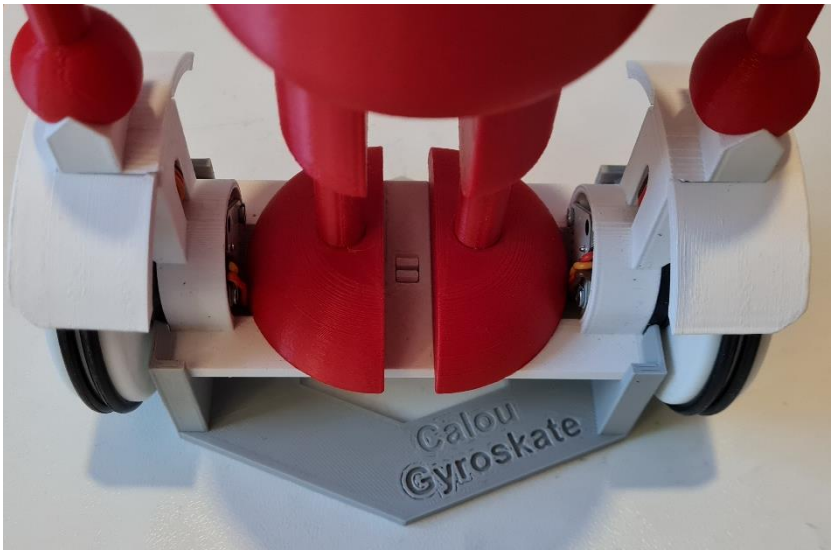
Antenna blinking:

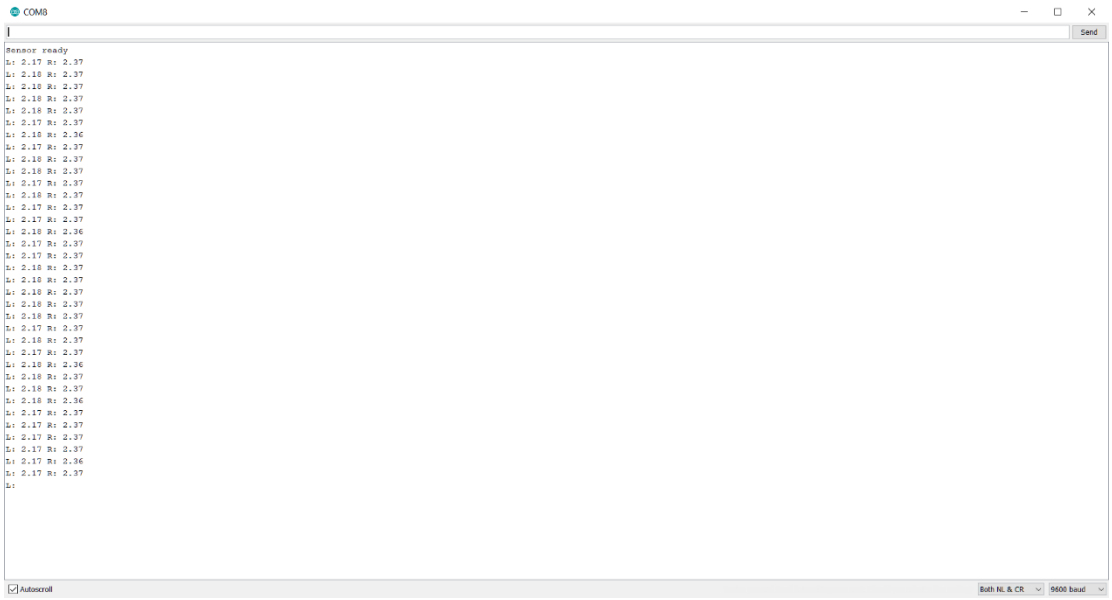
- Switch off the robot and disconnect the Bluetooth module.
- Connect the BGC board through the USB plug to the computer.
- Open and upload the sketch: “antenna.ino”
- Disconnect the BGC board from the computer.
- Connect the Male JR plug of the antenna to the A3 slot. Mind to put the brown cable on the ground side.
- Mount the head. There are two ribs that must enter to the grooves for orientate the head.
- Switch on the robot. The LED must blink within a second interval.



Magnetic sensor reading:

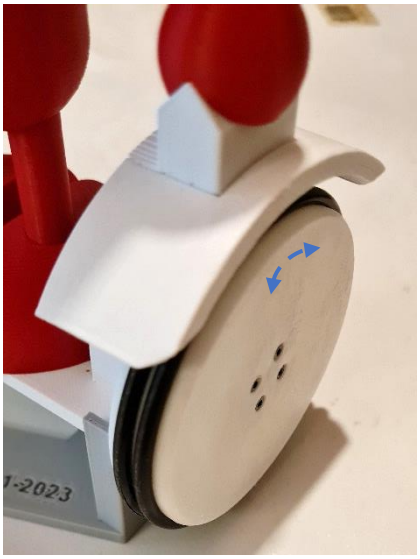
- Clip the robot on the stand base.
- Switch off the robot and disconnect the Bluetooth module. You can also disconnect the plug of the antenna to move the head aside.
- Connect the BGC board through the USB plug to the computer.
- Open and upload the sketch: “test_magnetic_sensor_analog.ino”
- Open the serial monitor and set the baud rate @ 9600.
- You should see a scrolling of the Left and Right sensor angles.
- The angle values should change if you turn manually the wheels.





Find sensor offset and direction of the motors:

- Clip the robot on the stand base and take care that wheels are turning free.
- Switch off the robot and disconnect the Bluetooth module. You can also disconnect the plug of the antenna to move the head aside.
- Connect the BGC board through the USB plug to the computer.
- Open and upload the sketch: “find_sensor_offset_and_direction.ino”
- Once upload switch on the robot and open the serial monitor and set the baud rate @ 9600.
- The Left wheel is doing 3 steps forward and backward and the Right is doing after the same.
- After that you should see the motors offset and direction values print for both. Please note those values and restart the serial monitor two or three time for average the offset values. The direction sense must remain the same.
- Switch off the robot.
- Keep those values. We will use it for update the robot sketch for skip the motor and sensor alignment procedure when turn on.



COM8

```
motorL
Offset: 5.9896 Direction: CCW
motorR
Offset: 0.2498 Direction: CW
To use these values provide them to the robot sketch:
motorL.initFOC(offset, direction)
motorR.initFOC(offset, direction)
If motor are not moving the alignment procedure was not successfull!!
```

Main robot sketch:

Before starting the robot you need to set different values:

Set the sensor offset and direction:

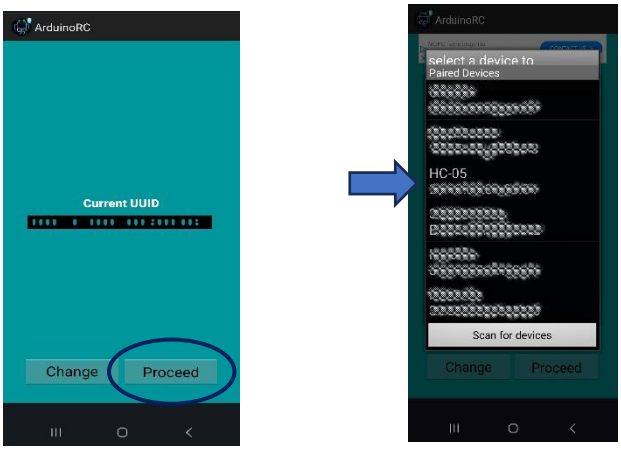
- Clip the robot on the stand base.
- Switch off the robot and disconnect the Bluetooth module. You can also disconnect the plug of the antenna to move the head aside.
- Open the sketch: “Calou-Gyroskate.ino”
- Find the lines “motorL.initFOC();” and “motorR.initFOC();”
- Complete the empty fields with values found at the previous step when running the sketch for find sensor offset and direction. This is what it looks like once filled:
  “motorL.initFOC(5.9896,CCW);“ and “ motorR.initFOC(0.2498,CW); “
  RQ: Don't try those values, it will not work. You need to use the ones you found.
- Save the sketch: “Calou-Gyroskate.ino”

Configure and set the Bluetooth remote:

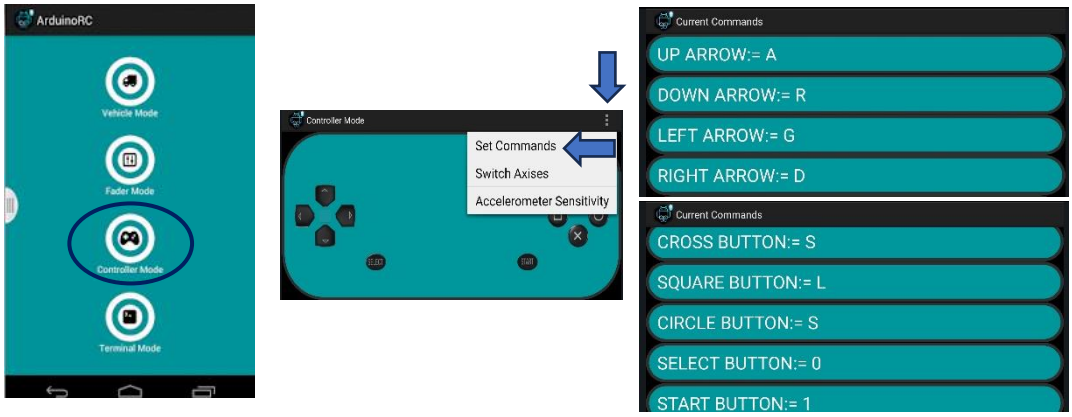
I’ve used a Bluetooth remote app available on Google Playstore called: **Arduino Bluetooth Controller**, however you can use any other Bluetooth application compatible with your device.

https://play.google.com/store/apps/details?id=eu.jahnestacado.arduinorc&pcampaignid=web_share

- Clip the robot on the stand base.
- Switch off the robot and connect the Bluetooth module. You can also disconnect the plug of the antenna to move the head aside.
- Connect the BGC board through the USB plug to the computer. A red LED on the Bluetooth module should blink rapidly.
- Activate the Bluetooth on your smartphone and go the parameters as you normally connect other Bluetooth devices for identify the module. The default name is commonly “HC-05” and the passcode is either 1234 or 0000. Refer you to the supplier’s information.
- Install and open the app **Arduino Bluetooth Controller**
- Clic on “Proceed” button for select the Bluetooth module.
- Once the module paired the red LED should blink slowly.



- Select the “Controller Mode” that emulate a gamepad and enter to the “Set Command” menu through the top right.
- Set the commands with the corresponding capital letter or number.
  UP ARROW:= **A** ; DOWN ARROW:= **R**; LEFT ARROW:= **G**; RIGHT ARROW:= **D**
  CROSS BUTTON:= **S**; SQUARE BUTTON:= **L**; CIRCLE BUTTON:= **S**;
  SELECT BUTTON:= **0**; START BUTTON:= **1**
- The configuration of the Bluetooth remote in ended!



The commands are corresponding to the following robot reactions:
UP ARROW:= Move forward ; DOWN ARROW:= Move Backword;
LEFT ARROW:= Turn left; RIGHT ARROW:= Turn right;
CROSS BUTTON:= Stand still; SQUARE BUTTON:= Stop turning; CIRCLE BUTTON:= Stand still;
SELECT BUTTON:= Balancer disabled; START BUTTON:= Balancer enabled

Set the PID parameters and filtering values:

The PID parameters and filtering values needs to be finetune. More information on the functions by reading the tutorial done for a STM32 Nucleo-64 board on the "SimpleFOCproject" in the Control algorithm chapter: https://github.com/simplefoc/Arduino-FOC-balancer

In short, the controller developed has two parts:
Stabilization loop: This is the loop that make the robot stand vertical. But it would probably never stop moving.
Velocity control loop: This is the loop to make the robot stand still.
In addition, there are filters to smooth the velocity control and the remote control.

- It is possible to keep the values filled in the sketch in case the parts are strictly the same of what I have used. If true, you can jump to the next step of activation.

Here is the extract:

```
// stabilisation pid
PIDController pid_stb{.P = 50, .I = 90, .D = 2, .ramp = 100000, .limit = 10};
// velocity pid
PIDController pid_vel{.P = 0.015, .I = 0.01, .D = 0, .ramp = 2000, .limit = _PI / 10};
// velocity control filtering
LowPassFilter lpf_pitch_cmd{.Tf = 0.35};
// low pass filters for user commands - throttle and steering
LowPassFilter lpf_throttle{.Tf = 0.4};
LowPassFilter lpf_steering{.Tf = 0.01};
```

However, you can tune yourself the settings of the robot by following the steps described below that I extracted from the community discussion forum: https://community.simplefoc.com/

- Clip the robot on the stand base.
- Switch off the robot and disconnect the Bluetooth module. (The Bluetooth module is not mandatory to start PID parameter tuning)
- Connect the BGC board through the USB plug to the computer. Keep it connected during all the setting phases (I advise using a long cable to let the robot free moving). You will have to repeat the uploading of the sketch “Calou-Gyroskate.ino” after each phase of progress in the setting sequence detailed below.
- Unclip the robot from the stand base and then put the robot on the floor with keeping it in the standing position waiting for the stabilization start once switch on. Take care to the falls!

  Setting sequence:
  a. Set all the PID values for stabilization and velocity both to zero.
  b. First start with the PID for stabilization (the hardest to set)
  c. First make the **P value** higher and higher until you notice that it starts to balance a bit. Don’t go too high because it will start shaking!
  d. Increase the **I value** to let it better react to larger disturbances. Don’t go too high because it will start shaking!
  e. Increase **D value** (small steps) to smooth it out, don’t go too high here because that will make the system unstable.
  f. If your robot can balance by itself, then it is time to work on the velocity PID (to let it stand still).
  g. Again, go through sequence “c” to “e” by small steps, but this one will be easier.

- Once the main PID parameters are fixed you can test the movements with the remote for fine tune the filters if needed. (don’t forget to unplug the Bluetooth module while you are connected by the USB cable)

Start the robot:

- Connect the Bluetooth module and the plug of the antenna.
- Mount the head. There are grooves and ribs for the orientation.
- Unclip the robot from the stand base and then put the robot on the floor with keeping it in the standing position waiting for the stabilization start once switch on.
- Paired the Bluetooth module in the remote app and select the “Controller Mode”.
- Press several times the arrows of the remote for increase or decrease the speed and shorten the turning radius. Take care don't go too fast on start.

I hope you enjoyed building this robot. Have fun seeing it moving and stabilize.

Author

Guy FEUILLOLEY

https://www.facebook.com/versat.ilerobot