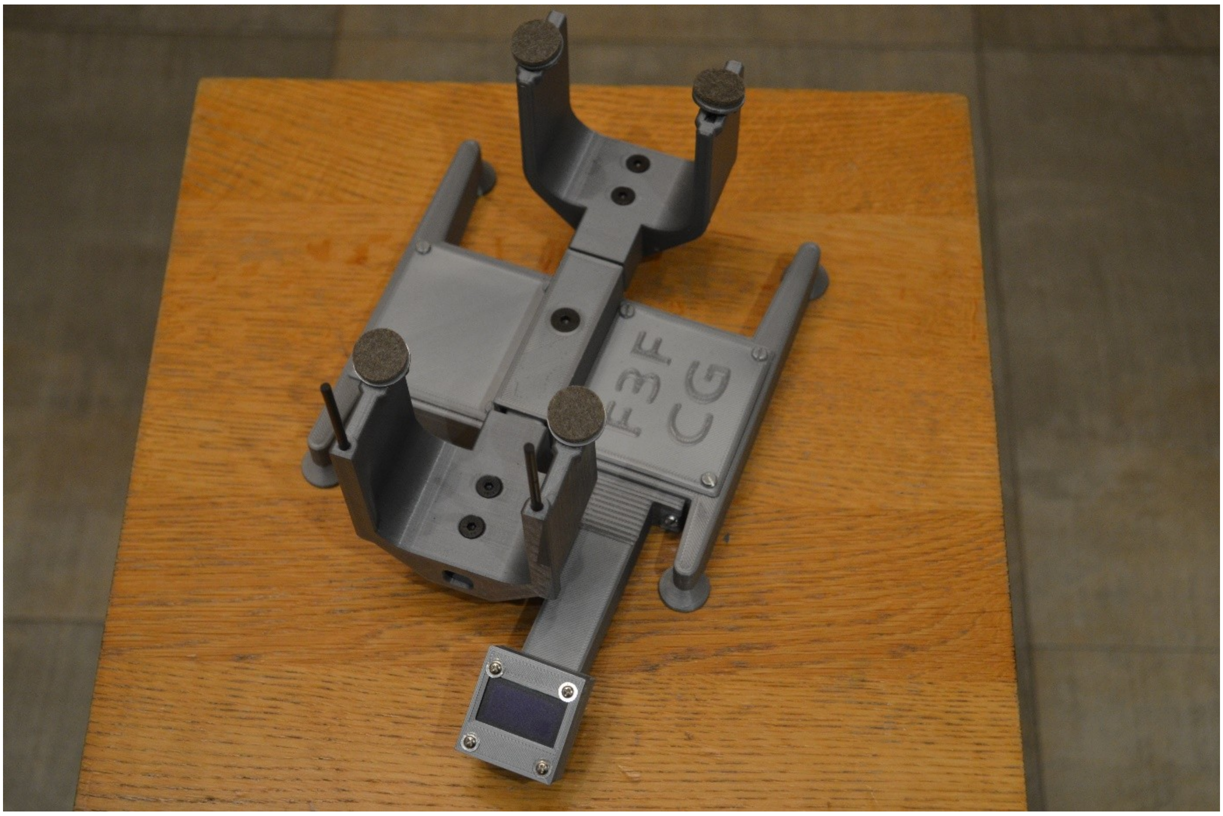
alain.desandre@wanadoo.fr

Arduino-based CG scale for F3x gliders (and others).

Enjoy !!



1 Version History

Version	Date	Motif
1.0	26.02.2018	Creation
1.1	17.03.2018	Update bypass A7 pin if not the right arduino card.
1.2	19.03.2018	Update schematic diagram introduction.
1.3	26.03.2018	Update sketch download step by step.
1.4	22/05/2018	Update new sketch Calibrate_bothHX711.ino to simplify the calibration phase. update troubleshooting on arduino regulator problem.
1.5	09/01/2022	Update : Introduction of STL files for larger arms to be able to use the scale with models with a larger than F3F spindle. Add principles of operation of the CG Scale.

2 Introduction-

This document presents the instructions to build a CG scale that can be used for FNx gliders.

At the end of 2017, many posts appeared on the RC forums, presenting a digital scale dedicated to the determination of the CG for F3F and F3B gliders.

While searching the net for information on this digital scale sales over 130 euros in various shops, I came across the site of **Olav Kallhovd**, Norwegian F3F pilot (thanks to him), (https: / /github.com/olkal/CG_scale), which has made in open source all the information needed to build a scale very similar to commercial models by using two Arduino boards and a 3D printer.

The photo below shows the scale designed by Olav. All the elements needed for buid are available on his site, including ".stl" files for 3D printing.

This scale can be used for gliders type F3F, F3B and F5J, the different widths being available for 3D printing.

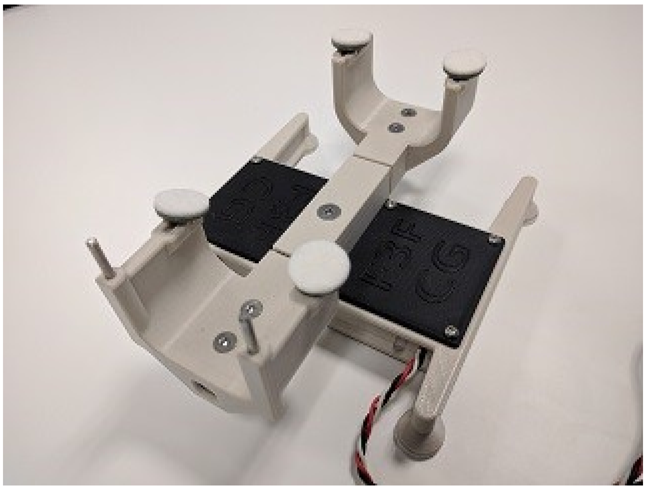


***Figure** 1 : The model originally designed by Olav Kallhovd with a remote screen.*

After having built a model identical to that initially proposed by Olav, with the help of colleagues from the National Aeroclub of Electricians and Gazier (ANEG) (Damien Donsez, Christophe Nocchi and Fred Hours), we have introduced some modifications to facilitate and simplify the construction of this scale:

- The initial 1602 LCD screen has been replaced by a 0.96 " OLED display in I2C. It is also possible to use a 1602 LCD display in I2C and you will find on the site below the Arduino files for the three versions (Olav base version 1602, OLED 0.96 " I2C and 1602 I2C). Using the I2C for the screen simplifies wiring and uses only **one Arduino board** instead of 2,

- The external led visible on the above picture has been replaced by the builtin LED on the Arduino boards. If you print in "clear" color PLA as in the picture, the blinking of the led in the base remains very visible,

- A small PCB has been designed to facilitate the integration of the few electronics components. This PCB fits into the base of the scale, and the associated Fritzing and Gerber files are also available on the github below. With these files, you can order the PCB directly from the Fritzing

Fab website (https://aisler.net/fritzing) or Seed Fusion website (https://www.seeedstudio.com/fusion_pcb.html). It is quite possible to do without this PCB to build the balance, by wiring the various components to one another wire to wire ... but it is less simple and more delicate !

- Some stl were modified to enlarge a few holes and we created an additional stl to integrate a connecting arm to connect the OLED screen to the base.

All these modifications, including the Fritzing file and the gerber for the PCB, are available in Open Sources at https://github.com/adesandr/CG_Scale_OlkalBreakout. When finished, the 0.96 "OLED screen scale looks like the picture below.

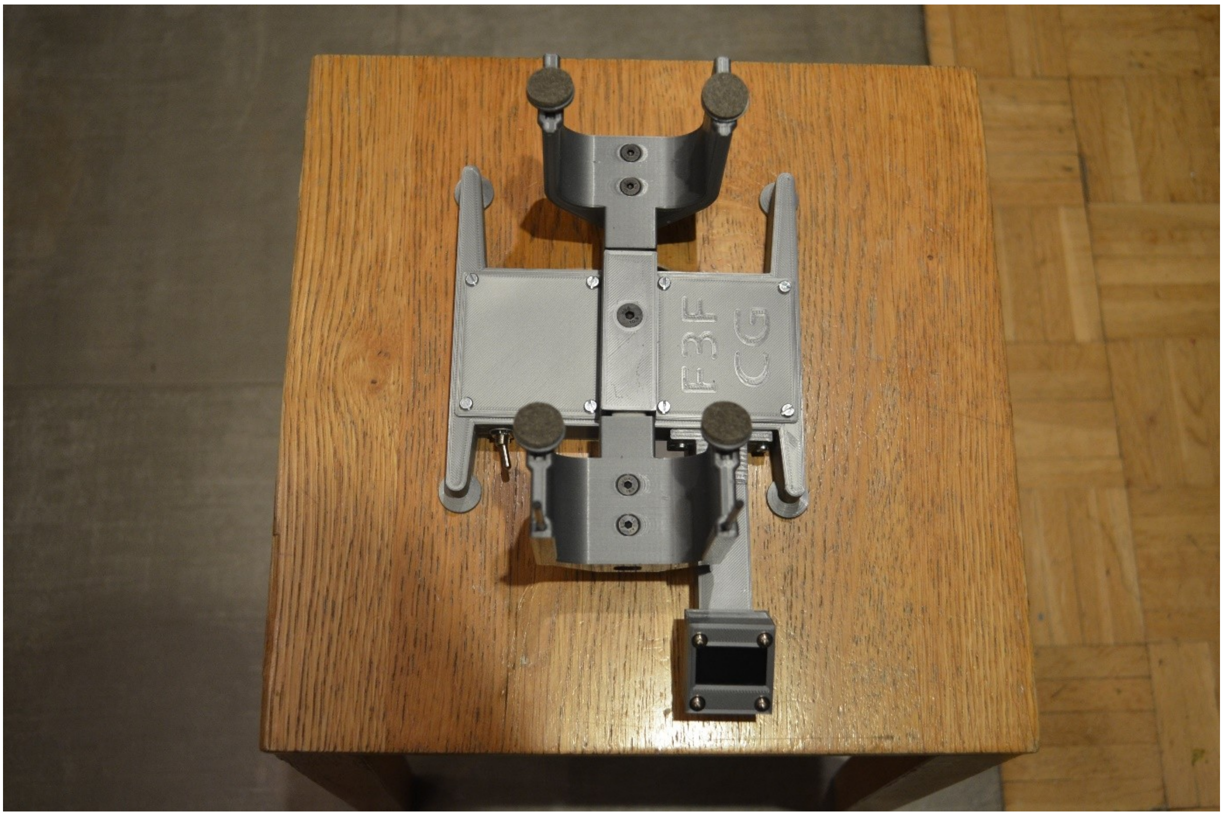


***Figure** 2 **:** Finished CG Scale in OLED 0,96" version*

The steps to build the F3F version with 0.96 " OLED display in I2C are presented in the following chapters.

3 Principles of operation of the CG scale

The CG scale is based on the use of two strain gauges.

3.1 What is a strain gauge?

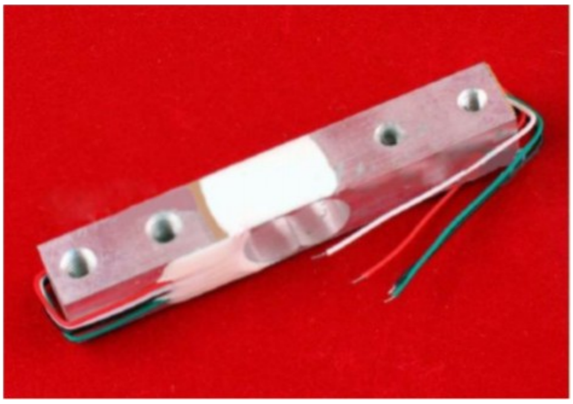


Figure 3 : Strain gauge example

A gauge consists of a very thin wire glued to a very thin support (thickness between 20 µm and 30µm). Most of its length is distributed parallel to a fixed direction Ɛx. Larger wires are used to solder cables to connect the gauge to a conditioner. When it is desired to know the elongation of a structure along a given direction, the gauge is placed, wires parallel to that direction. The measurement of the resistance variations is related to the relative elongation or longitudinal deformation.

> **It is possible to find strain gauges for weighing from a few hundred grams to several hundred kilograms. For the scale, depending on the version, we use a gauge of 2 Kg at the front and 3 Kg at the back for the F3F and F5J versions, and gauges of 10kg at the front and back for the XL version. Of course, the larger the target weight of the gauge, the less accurate the weighing is at low weights. All gauges work identically, only the calibration factor differs.**

3.2 How a strain gauge works ?

Since the strands of wire constituting the gauge are mainly aligned along the Ɛx direction, (except for the connecting loops between successive strands), it is admitted that the wire undergoes the same deformations as the surface to which it is bonded.

If we consider a wire that is subjected to tension (or compression), within the limits of its elastic range, it elongates under the effect of the load, while its cross-section decreases. If Ɛ is the relative elongation, the diameter undergoes a relative decrease, i.e., -µƐ, µ being Poisson's ratio, a figure close to 0.3 for most metals.

Knowing that the resistance of a conductive wire is: R = ρ * L/S (with ρ = resistivity, L = length and S = section), we obtain by logarithmic derivation and by making the approximation of a constant resistivity, we get ΛR/R = K * ΛL/L with K which is a constant that depends on the metals considered (K = 2 for constantan, nickel chromium, K = 3.2 for invar and K = 0.5 for manganin).

On the basis of this formula, a strain gauge will allow us to detect a very small resistance variation by measuring a voltage variation using a Wheatstone bridge.

3.3 Application to the CG scale

The stressing of a gauge on the CG scale is shown below.

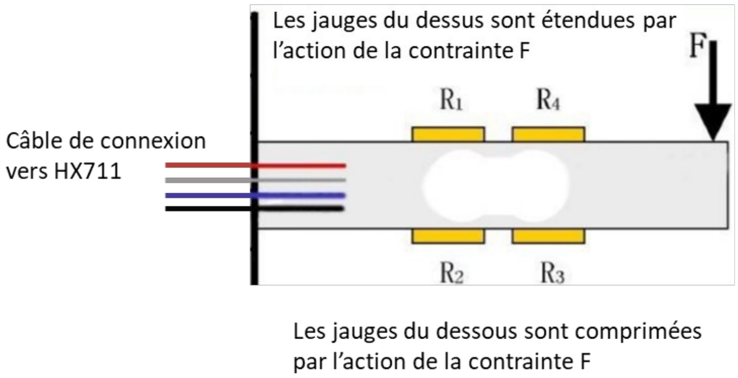


Figure 4 : *Gauge stressing*

Each gauge of the scale is fixed by one side to the base of the scale. When the gauge is deformed by the weight of the glider, the corresponding stress is detected by a change in the value of resistors R1, R2, R3 and R4. These changes are converted into voltage variations which can be converted by a weight equivalent.

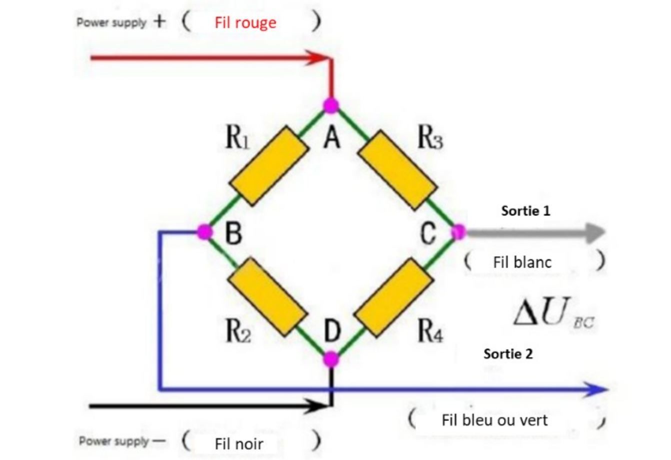


Figure 5 : *Wheastone bridge equivalence.*

The four resistors of the bridge are connected to strain gauges. The bridge is designed to output a stable value under excitation voltage that can vary from one gauge to another (hence the need for a calibration process). When a stress is applied, the resistance values will vary slightly, producing a voltage variation that can be correlated to the applied force depending on the material used.

3.4 Coupling with the HX711

The HX711 circuit is a two-channel (A & B) 24-bit analog/digital converter that operates with a maximum voltage of 5.5V, nominal 5V. An integrated regulator allows to stabilize the excitation voltage which cannot be higher than 5V. The HX711 has an analog amplifier with an adjustable gain of 64 or 128 on channel A and 32 on channel B (which we will not use). The HX711 can operate with a sampling rate of 10 Hz or 80 Hz (by unsoldering a resistor). The noise characteristics given are 50 nV for a gain of 128 and a rate of 10 Hz (this is the configuration chosen for the balance) and 90 nV for 80 Hz.

Note that the gauges used are given for a variation of 0.017% / 10°C for zero and 0.014% / 10°C at full load. Similarly the "creep", i.e. the long-term drift is given for a variation of 0.015% / 30 min. These characteristics for gauges of this price seem extremely low and in use, the "creep" announced seems relatively optimistic, without having dug more into the subject. These characteristics are however largely sufficient to determine the CG of a F3F glider.

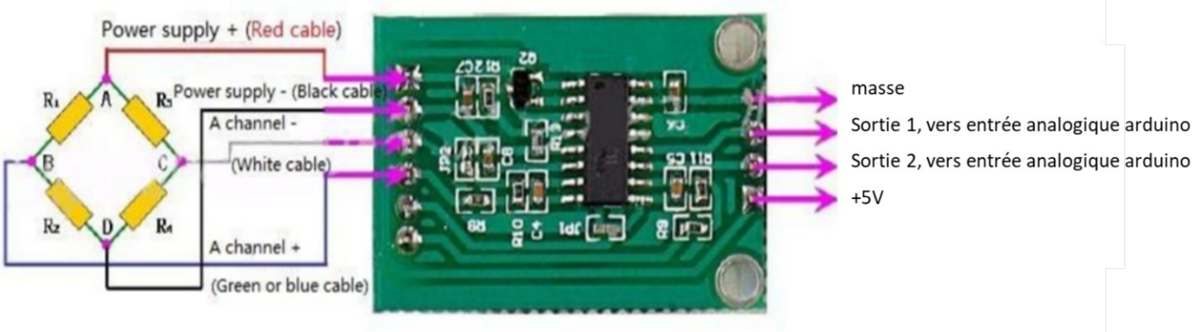


***Figure 6 :** HX711 connection schematic*

4 Printing and build of the mechanicals elements

4.1 Bill of material

BOM is available at the following URL : https://github.com/adesandr/CG_Scale_OlkalBreakout/blob/master/Documentation/BOM.pdf.

All components are available on Hobbyking, Ebay, Amazon or equivalent.

Taking into account the cost of PLA for 3D printing, the various versions cost about 40 €.

The scale is built in one day for the initial version and half a day for the I2c versions, including calibration, depending on your experience with a soldering iron and your knowledge of the Arduino environment. I can only encourage you to start, it is extremely simple and the tutorial is a bit long to present step by step the different stages of the build.

I draw your attention to the two warnings present in the file for the purchase of the HX711 cards and the Arduino Pro mini 5V board. Some HX711 cards generate a noise level that is too high and impact the measurements and some Arduino boards do not have the I2c bus control pins A6 / A7 correctly placed in relation to the PCB (this point is however easy to deal with, see troubleshooting in end of document).

4.2 Print the frame

The STL files needed to print the scale are available on my github in the "STL print files" directory. In this directory you will find several subdirectories:

- The "Generics files" subdirectory which contains the files not specific to a scale version. The files in this subdirectory correspond to a version of the scale for a zone of F3F.
- The "F3F Version" subdirectory contains files specific to the scale version with arms for an F3F zone,
- The sub-directorate "F5J version" which contains the files specific to the version of the scale with arms for a F5J time zone,
- The "LCD version" subdirectory which contains the files specific to the version of the scale for an LCD type screen (and not OLED),
- The "XL version" sub-directorate which contains the files specific to the version of the scale for 220x290 arms in order to use the scale for gliders with a fuselage larger than F3F or F5J fuselages.

The following picture shows the various elements of the CG Scale after printing with associated .stl files for a F3F version. Note that the lower part of the screen case is to be glued using epoxy on the "arm".

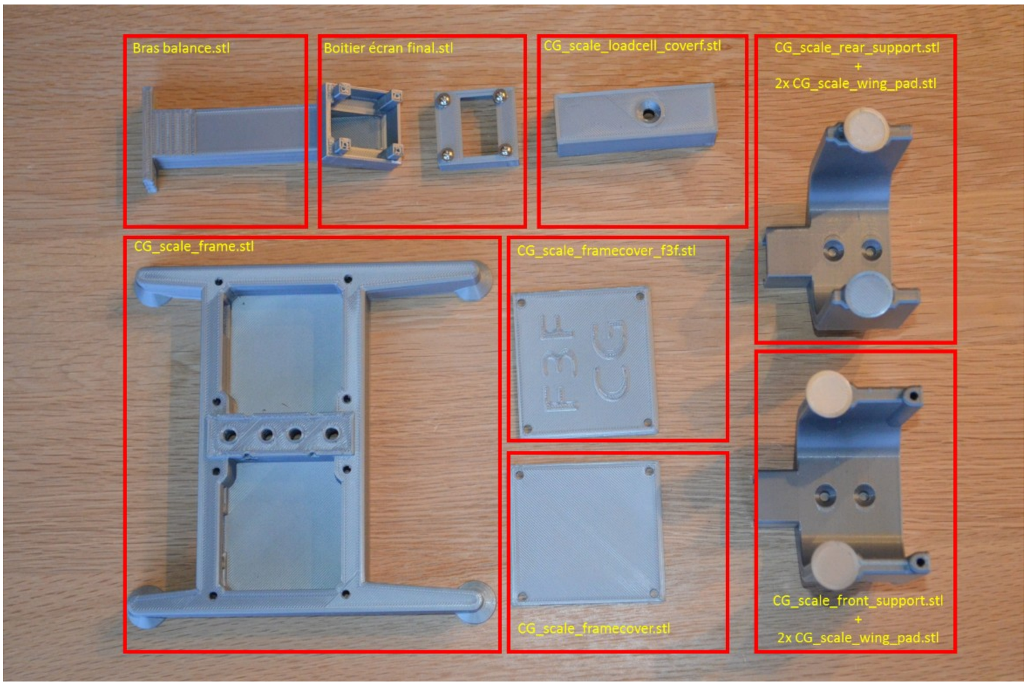


***Figure** 7 : The various elements of the scale with associated .stl files.*

A 3D printer with a minimum 220x220 plate is required. PLA material with generous infill is recommended. As an exemple, The Ice Filament PLA available on Amazon works well, as well as the Hobby King PLA, with 2 wall thicknesses and a 20% fill. The printing of all the parts of the scale takes about **twenty hours**.

Parts must be printed in the same orientation as in STL files.

Penetrating holes are binded for better print quality (continuous perimeter), these have to be drilled trough after printing.

4.3 Build the CG Scale frame

4.3.1 Step 1 : Strain gauges installation.

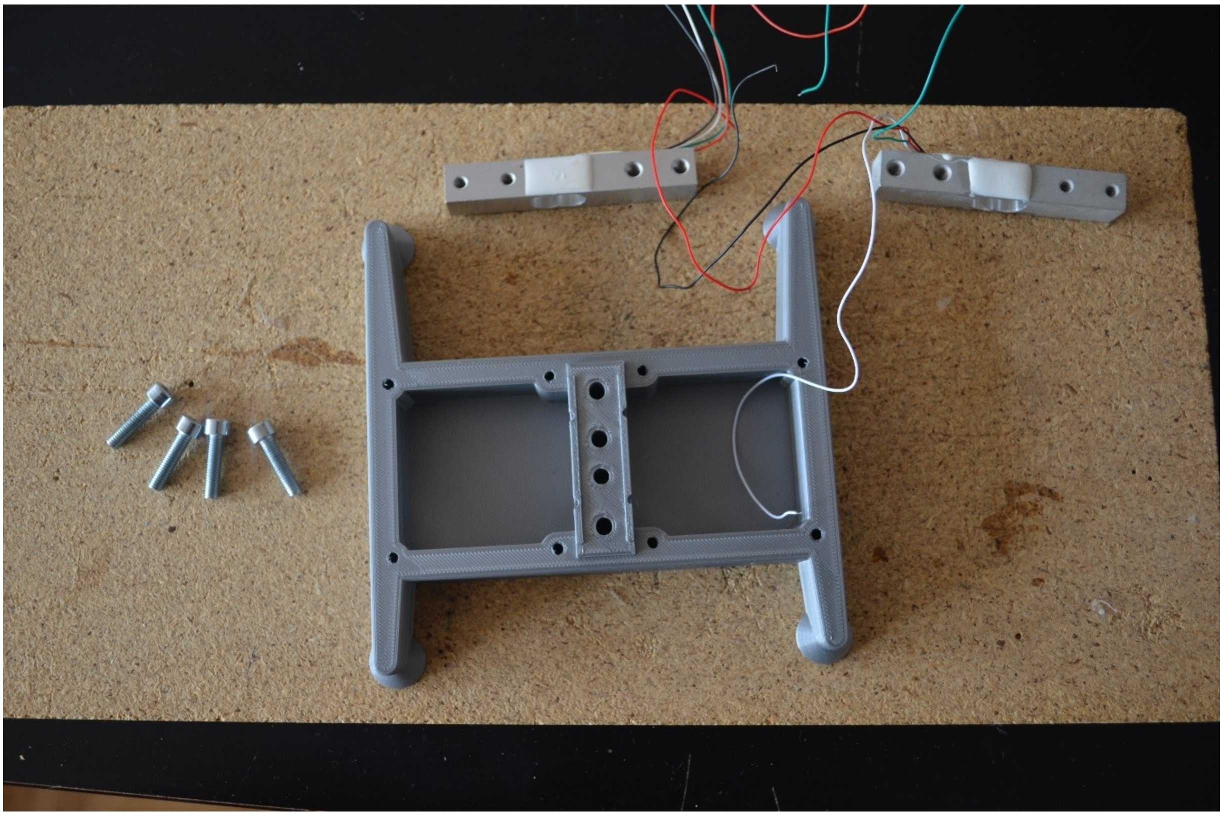


***Figure** 8 : Step 1 – Strain gauges installation.*

Strain gauges are installed on the frame of the scale using 4 HEX screws 5 mm x 20 mm. The contact surfaces of the strain gauges must be flat, and may require minor sanding. Check the overall alignment of the two strain gauges to insure the accuracy of the scale.

The 2 kg gauge is installed at the front of the scale, the 3 kg gauge is installed at the rear. The front of the scale is shown down in the picture of Figure 3.

Avoid cutting the wires on the gauges, otherwise the Wheatstone bridge equilibrium that is built into each gauge will be modified, which would slightly alter the accuracy of the scale.

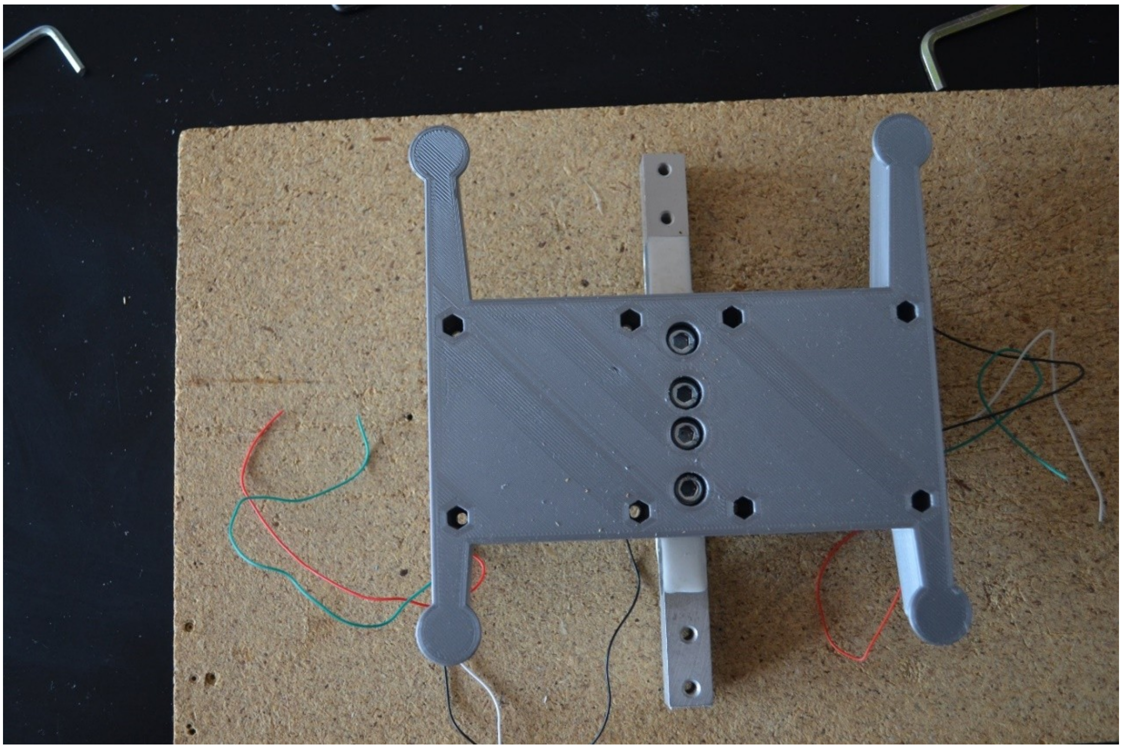


***Figure** 9 **:** Step 1 – strain gauges screwed onto the base of the scale.*

4.3.2 Step 2 : Install the gauges cap.

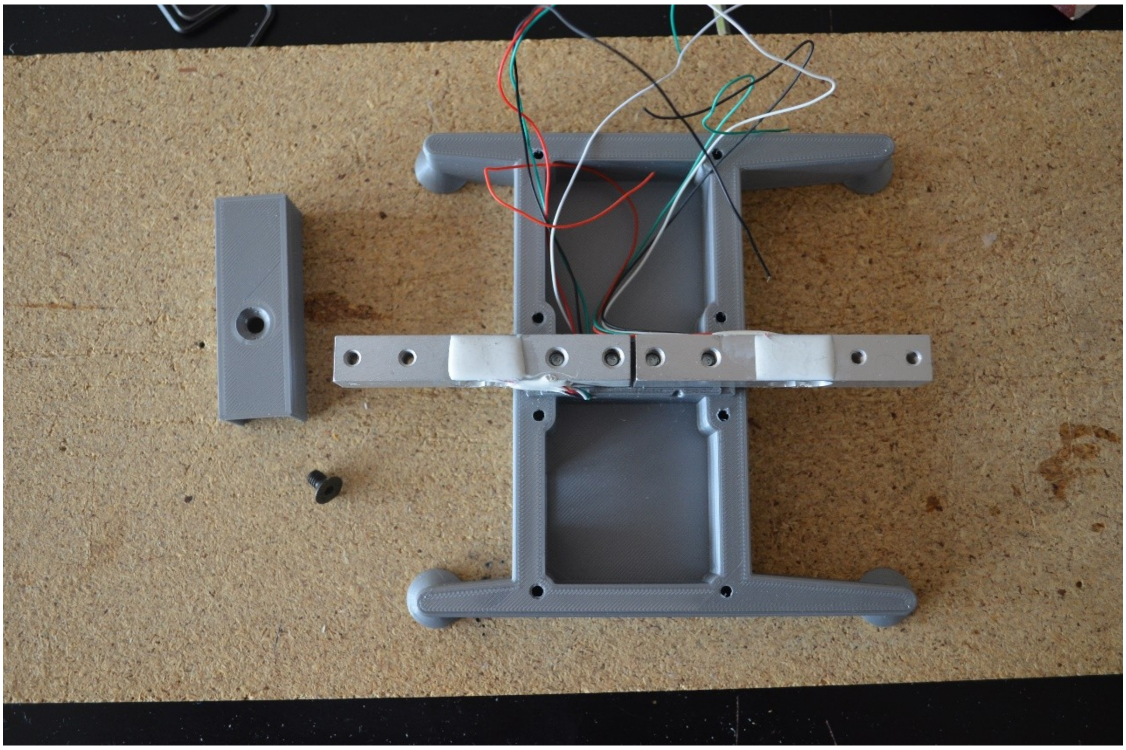


***Figure** 10 **:** Step 2 – Install the gauges cap*

The gauge cap is installed using a HEX M5 x 8 mm flat head screw.

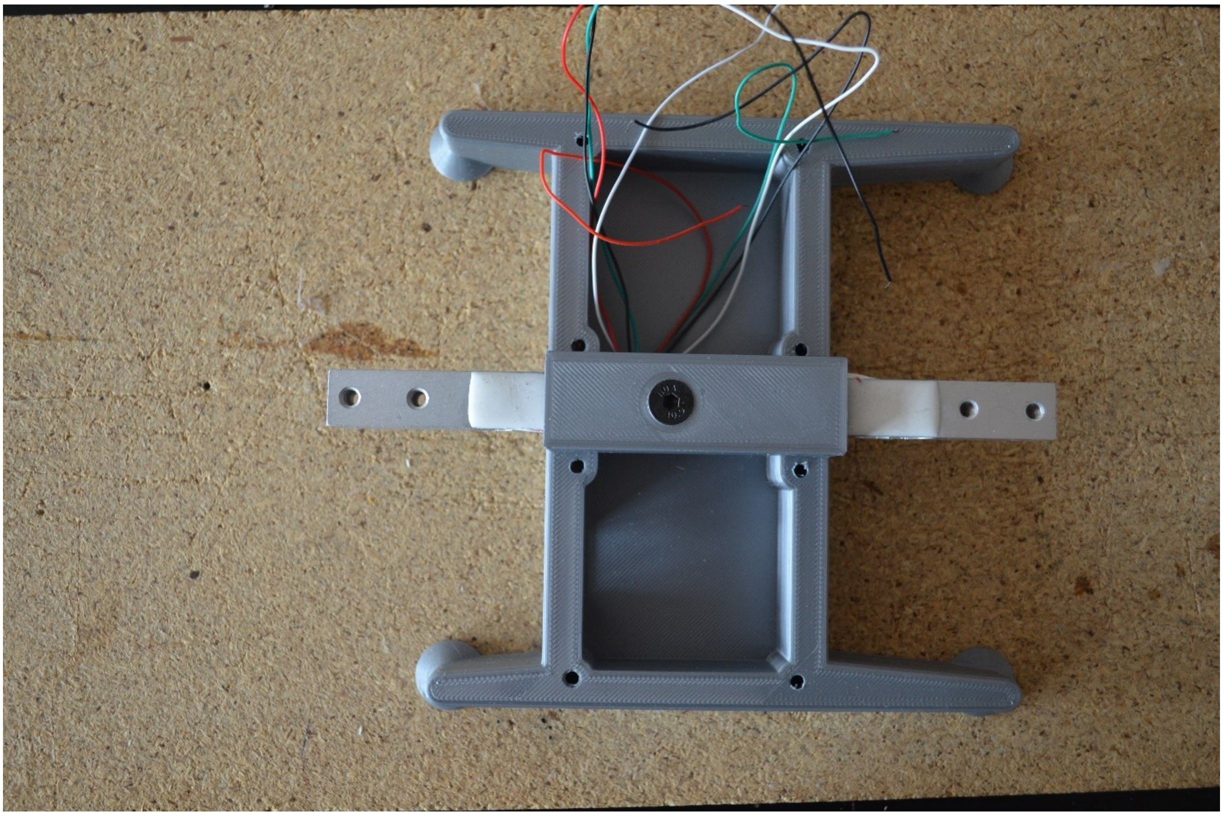


***Figure** 11 : Step 2 – Gauges cap Installation using a HEX M5 x 8 mmm flat head screw.*

4.3.3 Step 3 : Install the glider supports.

Both front and rear brackets are installed using 4 M5 x 16 mm flat head screws..

IMPORTANT: The 4 wing pads are angled for wing dihedral wings, and must be installed with correct orientation. The hinging for the wings pads must be loose to avoid any binding and to give some allowance for different wing dihedrals. Use a 2mm drill bit on the hole in the wing pads if they are tight.

Attach soft sticker pads (furniture type) on wing pads and under the frame.

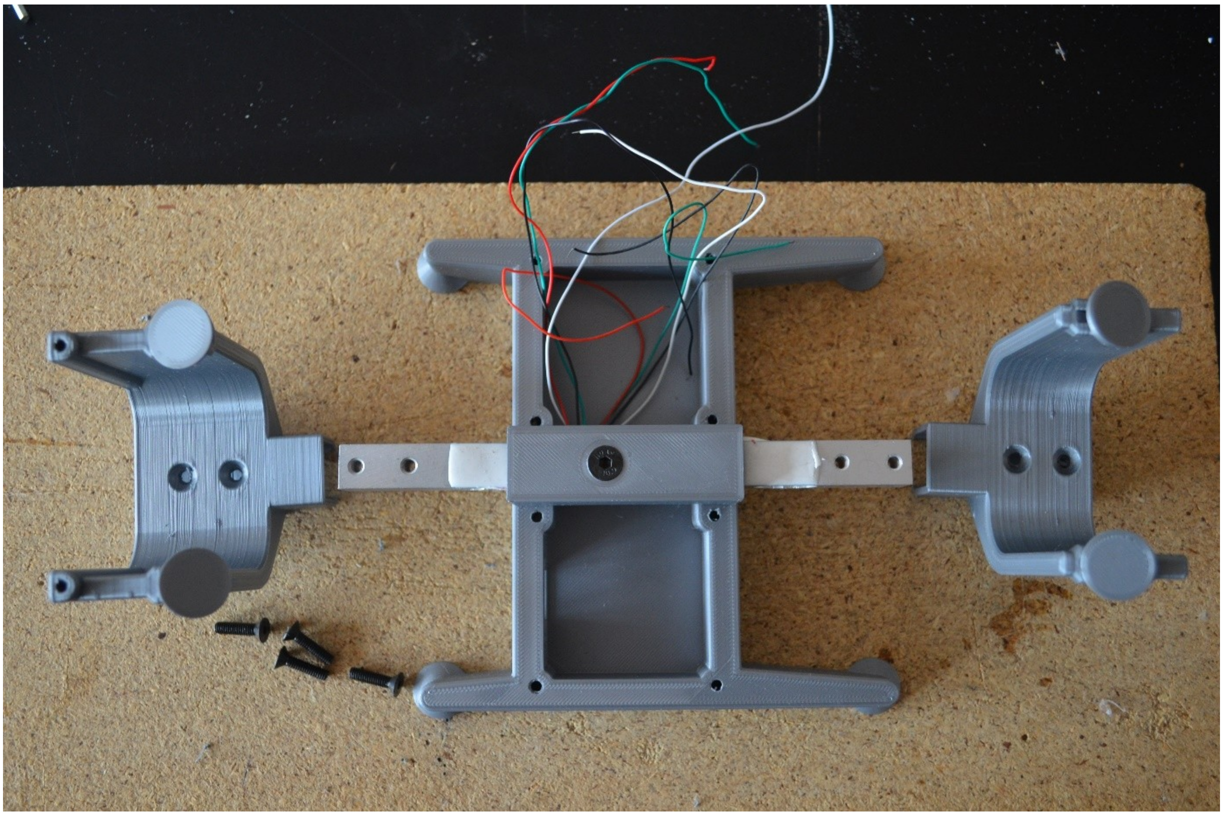


***Figure** 12 : Step 3 – Both glider brackets are installed using two HEX M5 x 16 mm flat head srews..*

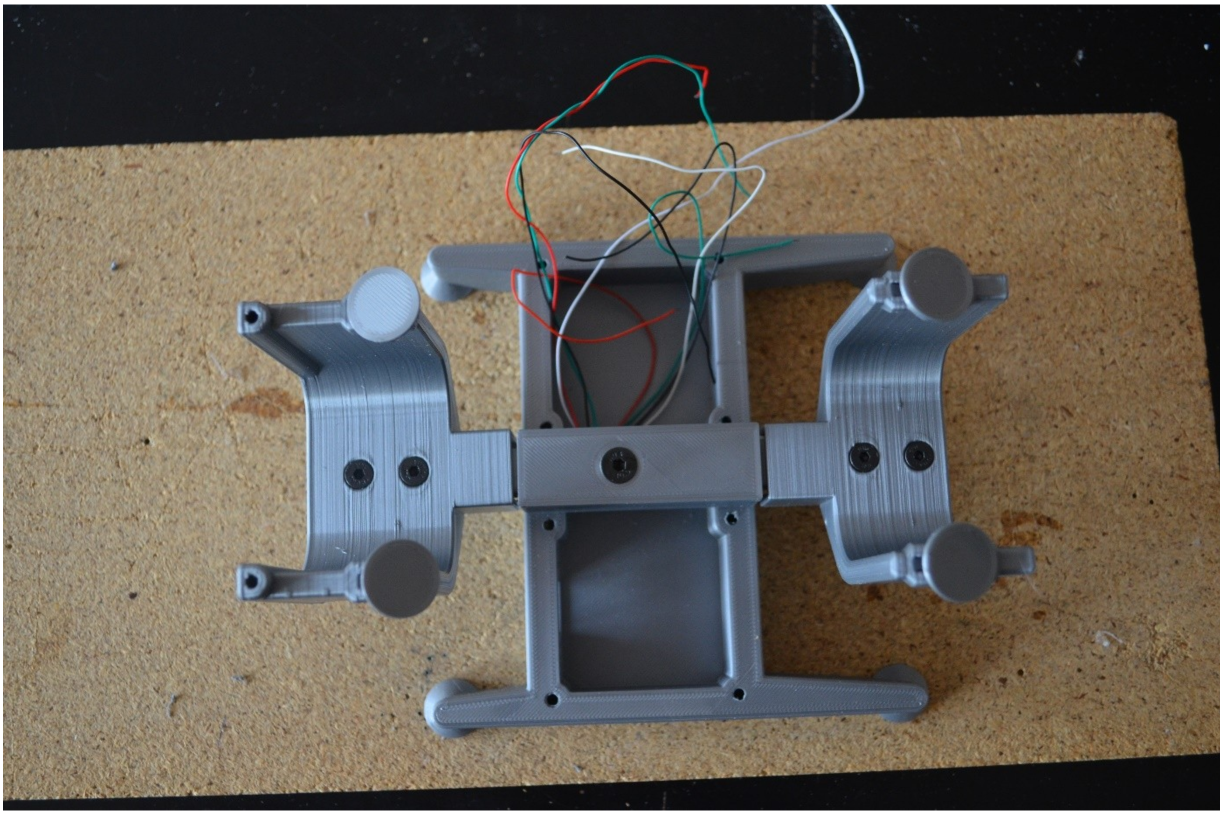


***Figure** 13 : Step 3 – Wing pads installation.*

4.3.4 Step 4 : Fix the support for the OLED screen

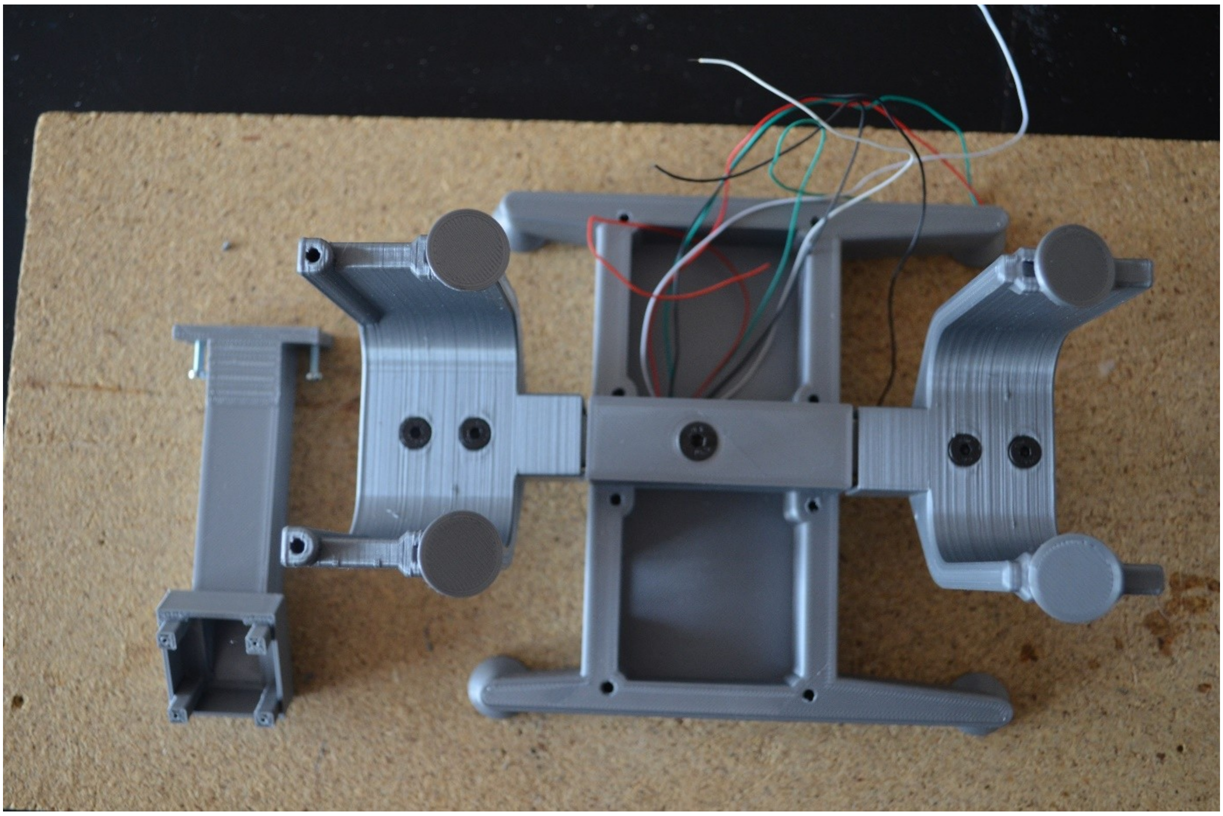


***Figure** 14 : Step 4 – OLED screen support installation*

The screen support is secured with two HEX M3 x 20 mm flat head screws. Note that the base of the screen box must be glued in advance using standard epoxy, on the arm of the screen.

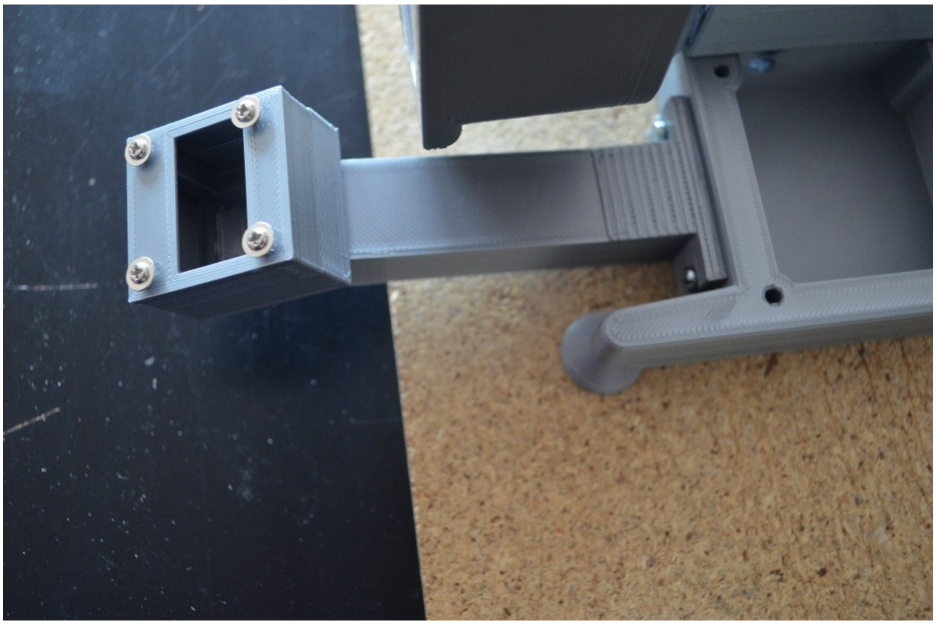


***Figure** 15 : Step 4 – Screen support installed with two HEX M3 x 20 mm flat head screws.*

To complete this step, release the 6 mm hole in the frame provided for the installation of a toggle switch and glue two caron rods of 3 mm x 45 m holes of both brackets.

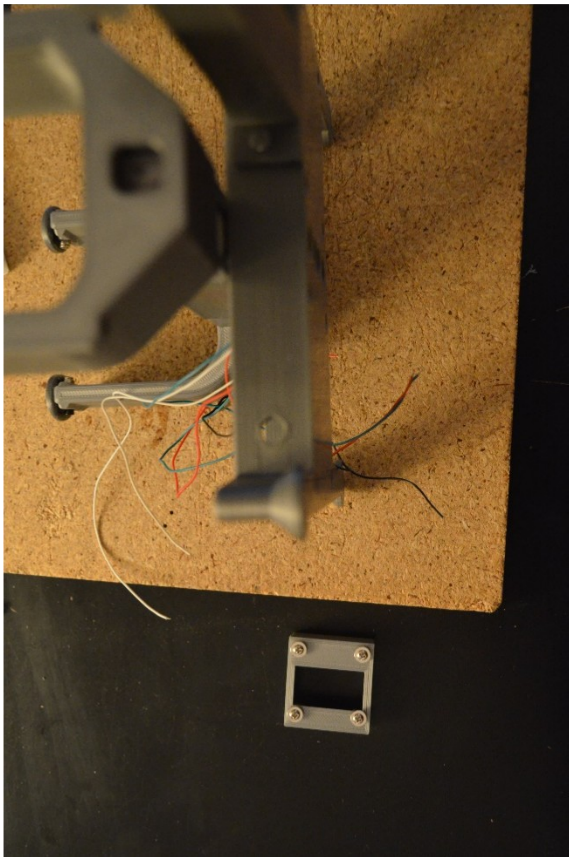


Figure 16 : *Step 4 – Use a Dremel with a 6 mm drill bit to open the the toggle switch hole.*

So congratulations, the assembly of the of the scale components is finished and we can move to the integration of the electronic parts..

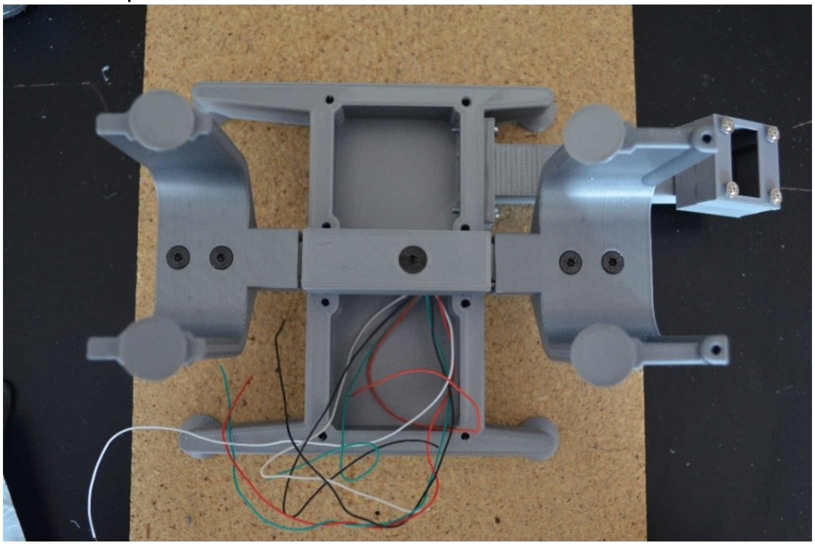


***Figure** 17 : Step 4 – The balance finished with all the assembled mechanical elements.*

4.4 Electronics parts integration

4.4.1 Bill of materials

The following picture shows the components needed to mount the scale. A 4.7 KΩ resistor and a 10 KΩ resistor are required for the voltage divider bridge that measures battery voltage.

A 4-wire ribbon connects the PCB to the 0.96 "OLED display.

Four 2.54 screw terminal blocks are used to weld on both HX711s to simplify the connection of strain gages. It's possible to do without it, but the build is much simpler by using them.

Two HX711 boards and one Arduino Pro mini 5V board are also used, as well as a toggle switch and a LiPo battery of maximum dimension 41mm x 21mm x 16mm.

A small prototyping plate (or breadboard) will also be useful to help make some welds (about 3 €).

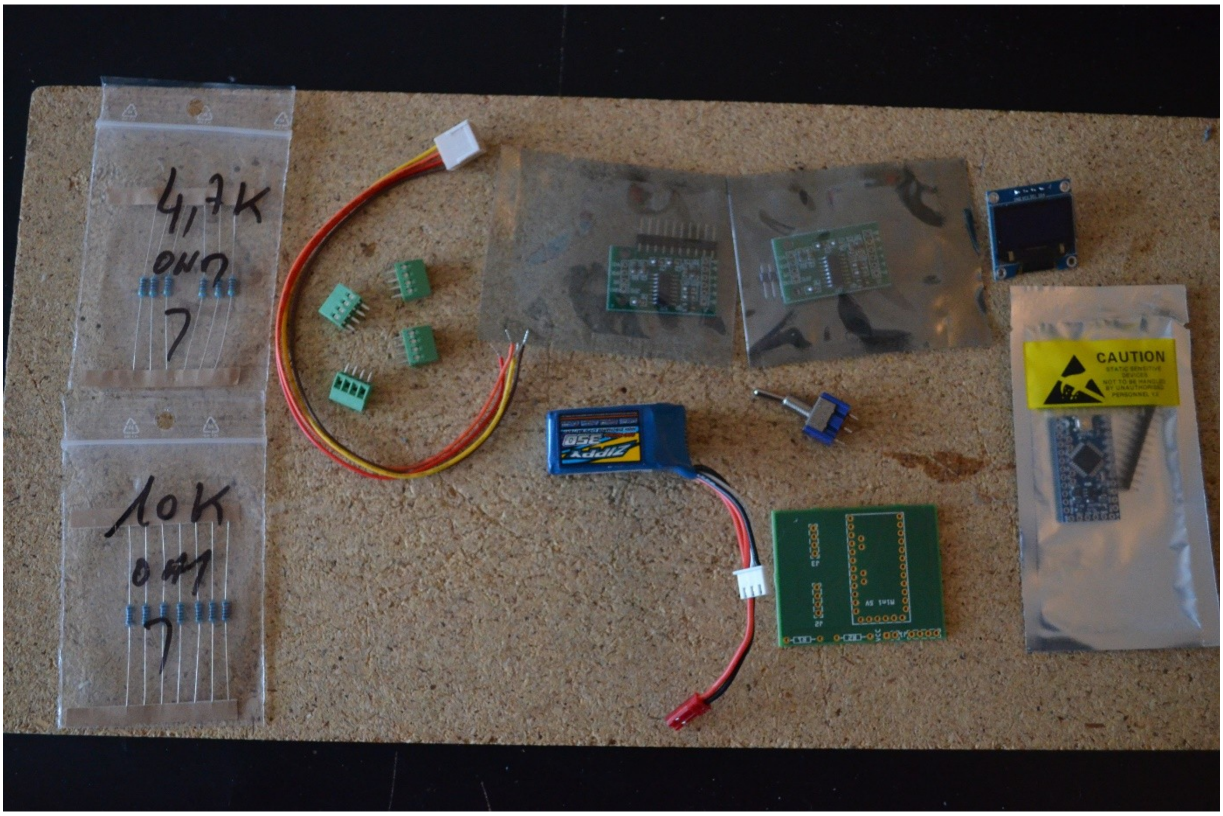


***Figure** 18 **:** Electronics parts necessary to build the scale.*

Since the Arduino Pro mini board does not have a USB port, it is also necessary to purchase a TTL-USB programmer. You can buy a programmer of this type on Amazon, https://www.amazon.fr/gp/product/B06Y38P7P2/ref=oh_aui_detailpage_o07_s00?ie=UTF8&psc=1 .
This type of programmer costs between 7 € and 8 €.

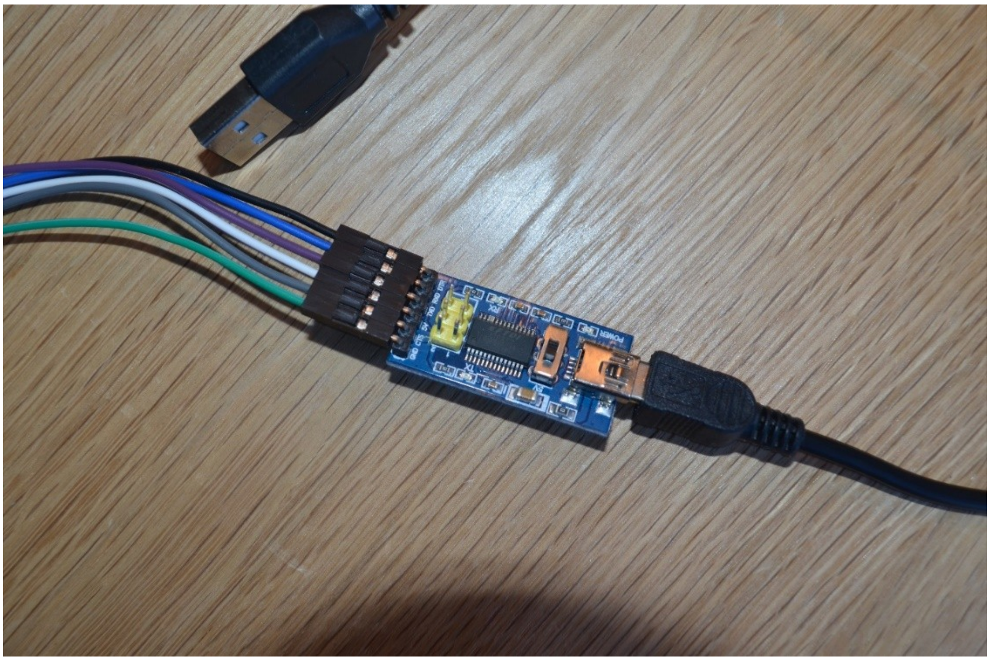


***Figure** 19 : TTL-USB 3.3V 5 V interface (Goaxing Tech as an exemple).*

4.4.2 Schematic diagram

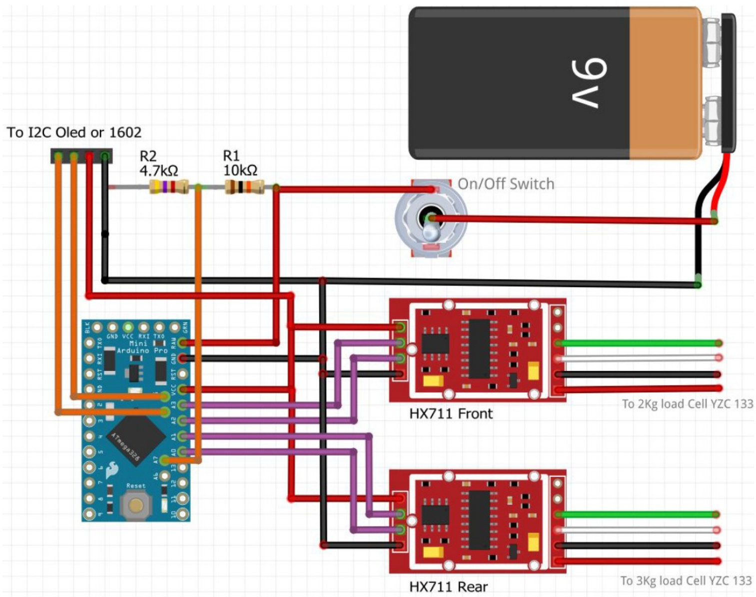


***Figure** 20 : Schematic diagram of the sigle arduino version using I2C display.*

The structure of the scale is pretty simple and is shown in the diagram above.

The battery (Lipo 2s 7.4 v easier to integrate into the right compartment of the base of the scale), passes through a toggle switch and feeds the arduino card. A voltage divider bridge connected to pin A7 of the Arduino board allows reading of the supply voltage. The regulated 5V of the arduino board is distributed on both HX711 and the I2C screen, as well as the ground. The I2C bus control SDA (pin A4) and SCL (pinA5) signals are connected to the I2C display. The control signals of both HX711 are connected to pins A0 to A3 of the arduino board

4.4.3 Step 5 : Prepare the Arduino board

We will start by soldering the programming connector to the Arduino board. For that a small breadboard will facilitate us welds. The programming connector must be welded "upwards" to facilitate the programming of the scale when the various components have been integrated into the base.

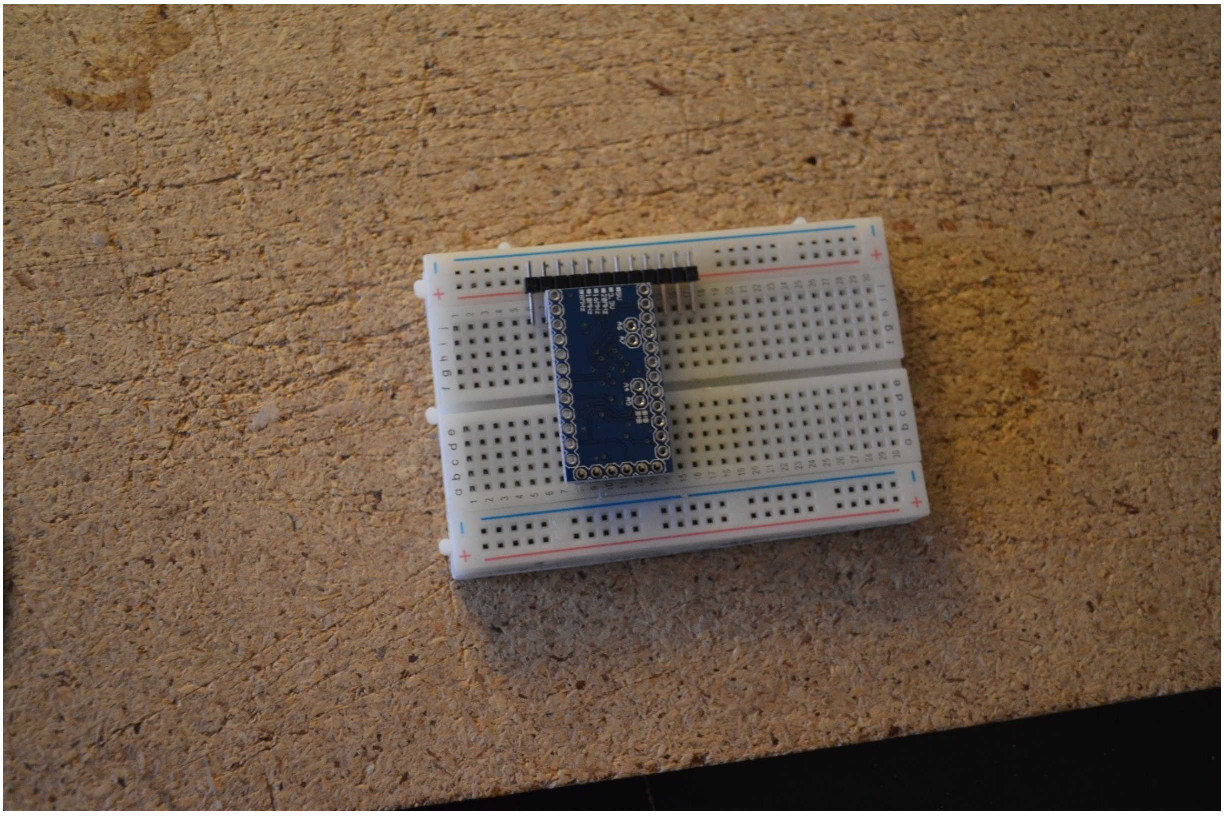


***Figure** 21 **:** Step 5 – soldering the programing connector on the Arduino board.*

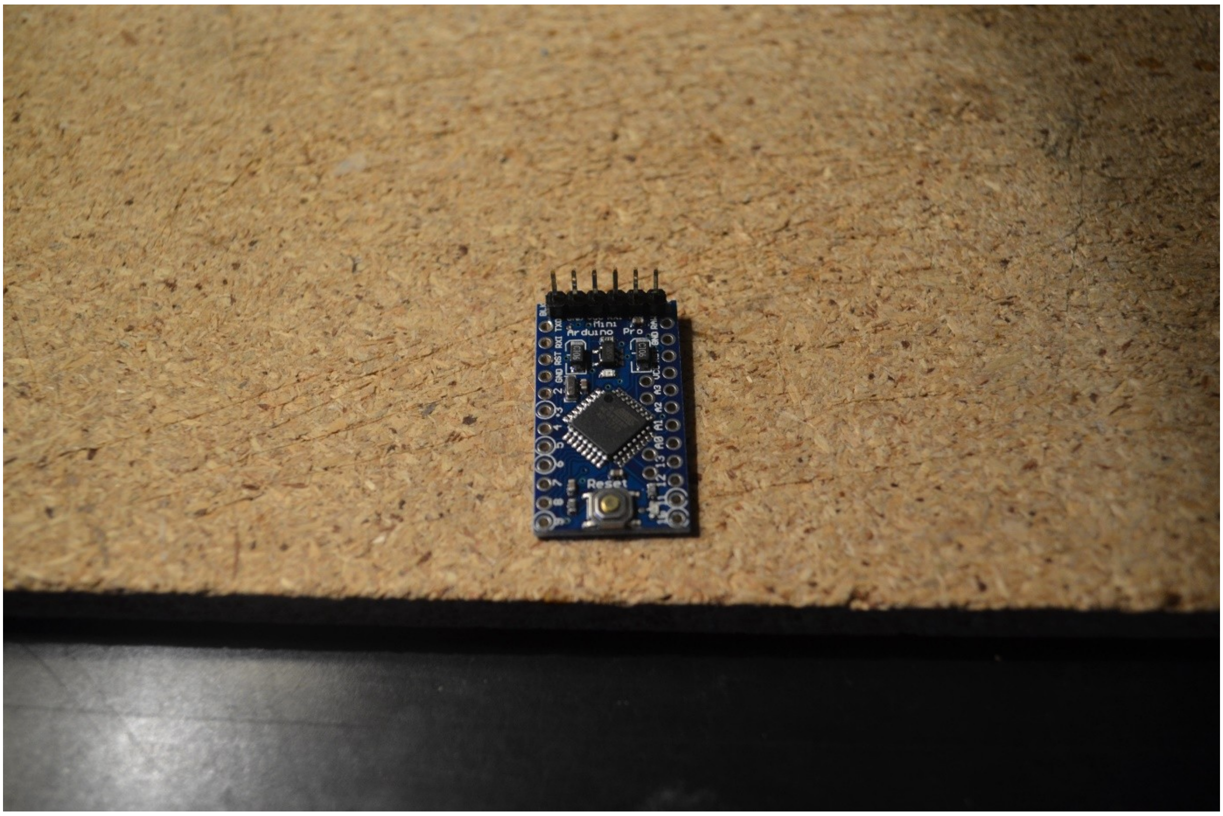


***Figure** 22 **:** Step 5 – the programing connector must be soldered « upwards »*

Once the connector is soldered on the board, I recommend that you perform a functional test of the board.

To do this, connect the Arduino board to your PC via the TTL-USB programmer, respecting the polarity (including GND and VCC). For this test of the Arduino board, only the VCC and GND cables are to be connected between the Arduino board and the programmer.

The wiring may change regarding your TTL-USB and your arduino board . Nethertheless, the logic is the following :

- **GND TTL-USB <-> GND arduino board,**
- **VCC TTL-USB <-> VCC arduino board (or 5V depends thhe serigraphy),**
- **TX0 TTL-USB <-> RX0 arduino board,**
- **RX0 TTL-USB <-> TX0 arduino board,**
- **CTS TTL-USB <-> CTS ou GRN arduino board,**
- **DTR TTL-USB <-> BLK arduino board.**

All Arduino boards are delivered with a program that flashes the led installed on the card, if your card is functional, after connection to the PC, the LED should flash with a frequency of about 1s.

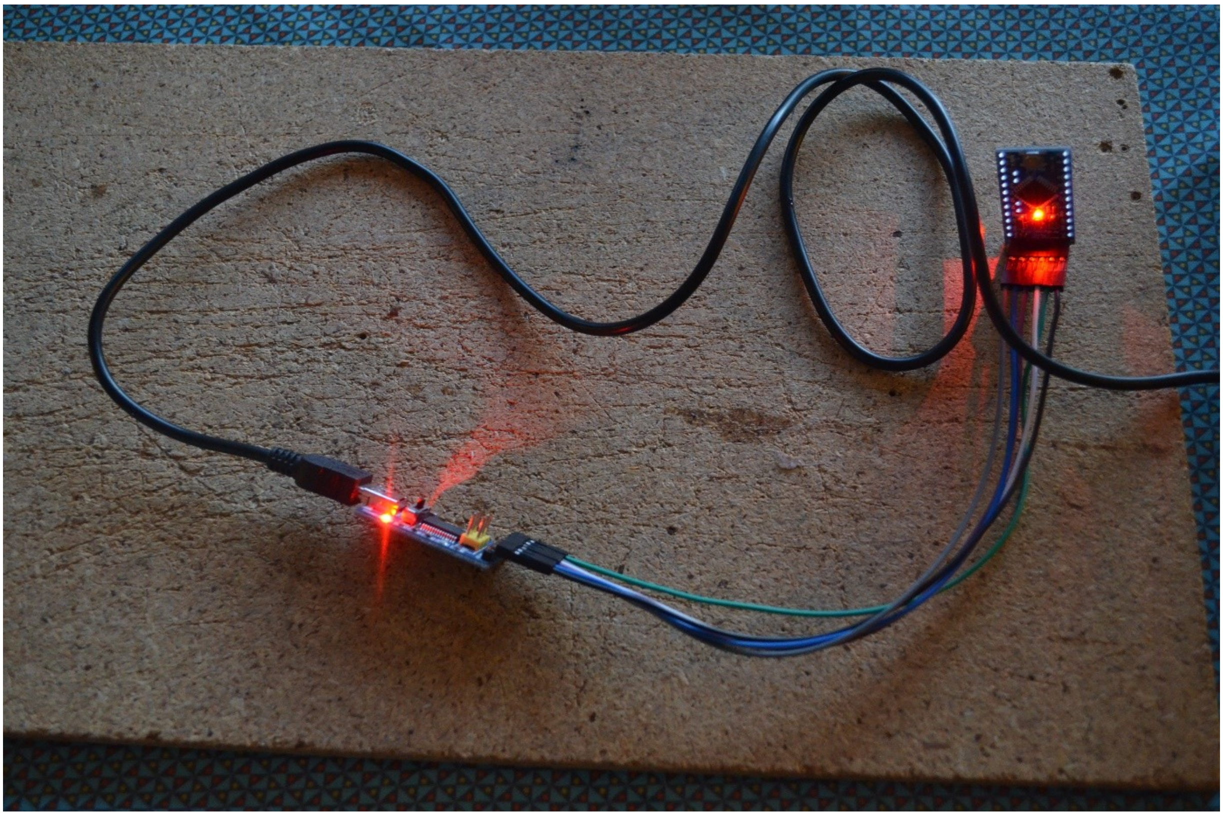


***Figure** 23 : **Step** 5 – Arduino board functionnal check after programmer connector soldering.*

4.4.4 Step 6 : Prepare the PCB

PCB preparation consists of soldering a 10 kΩ resistor at location R1 and a 4.7 kΩ resistor at R2.

Also solder the Arduino board to the PCB (pins: raw, gnd, A0, A1, A2, A3, A4, A5, A7). Only these pins are used.

Finally weld a ground wire and a power wire on the connector J1 (allow a little margin for the wire, 15 cm can be adapted later).

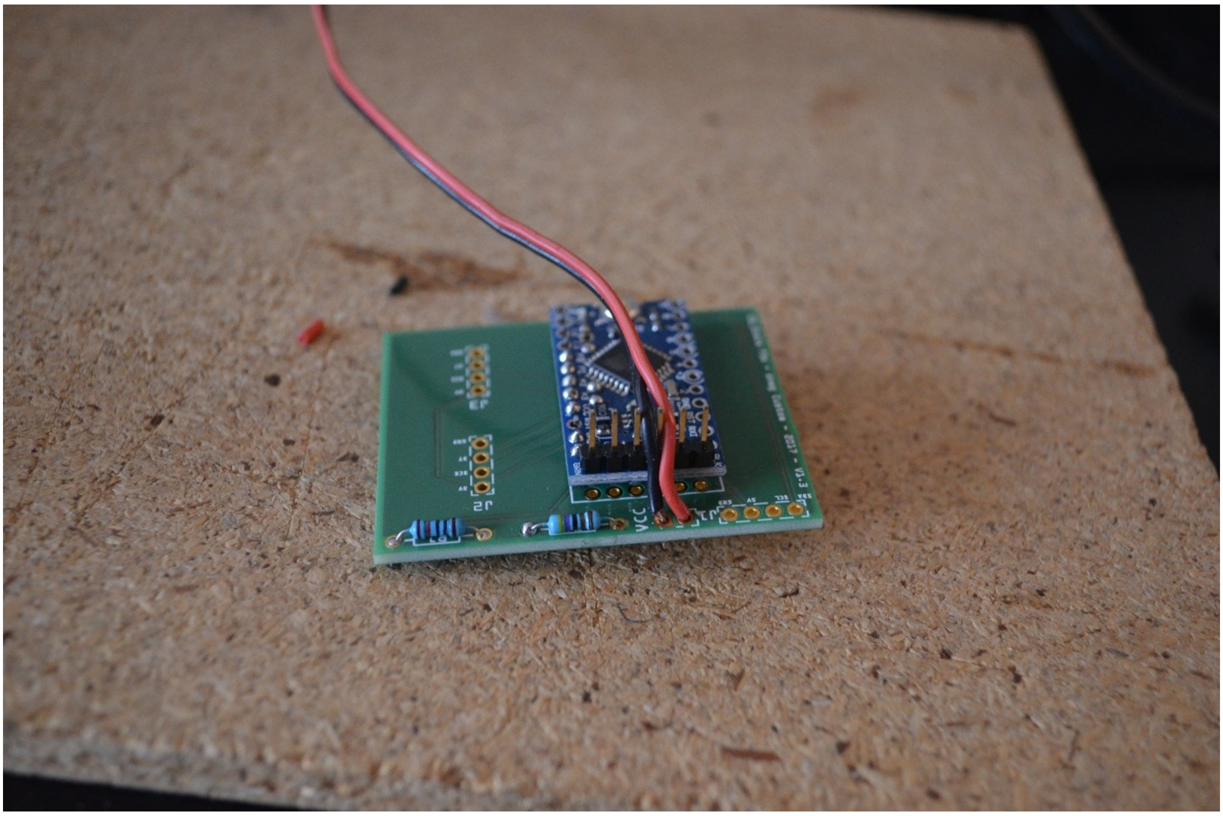


***Figure** 24 **:** Step 6 – PCB preparation*

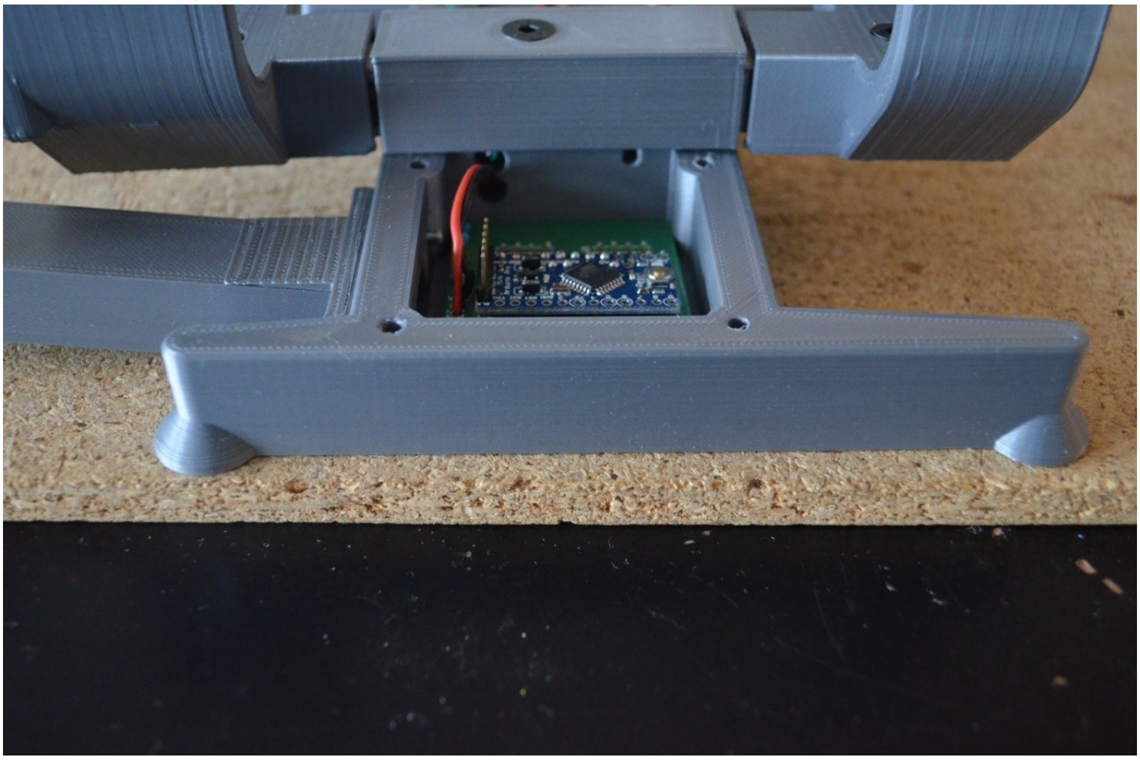


***Figure** 25 **:** Step 6 – PCB installation test into the base.*

4.4.1 Step 7 : Setting up the switch.

Solder the power wire from the PCB to one of the switch tabs.

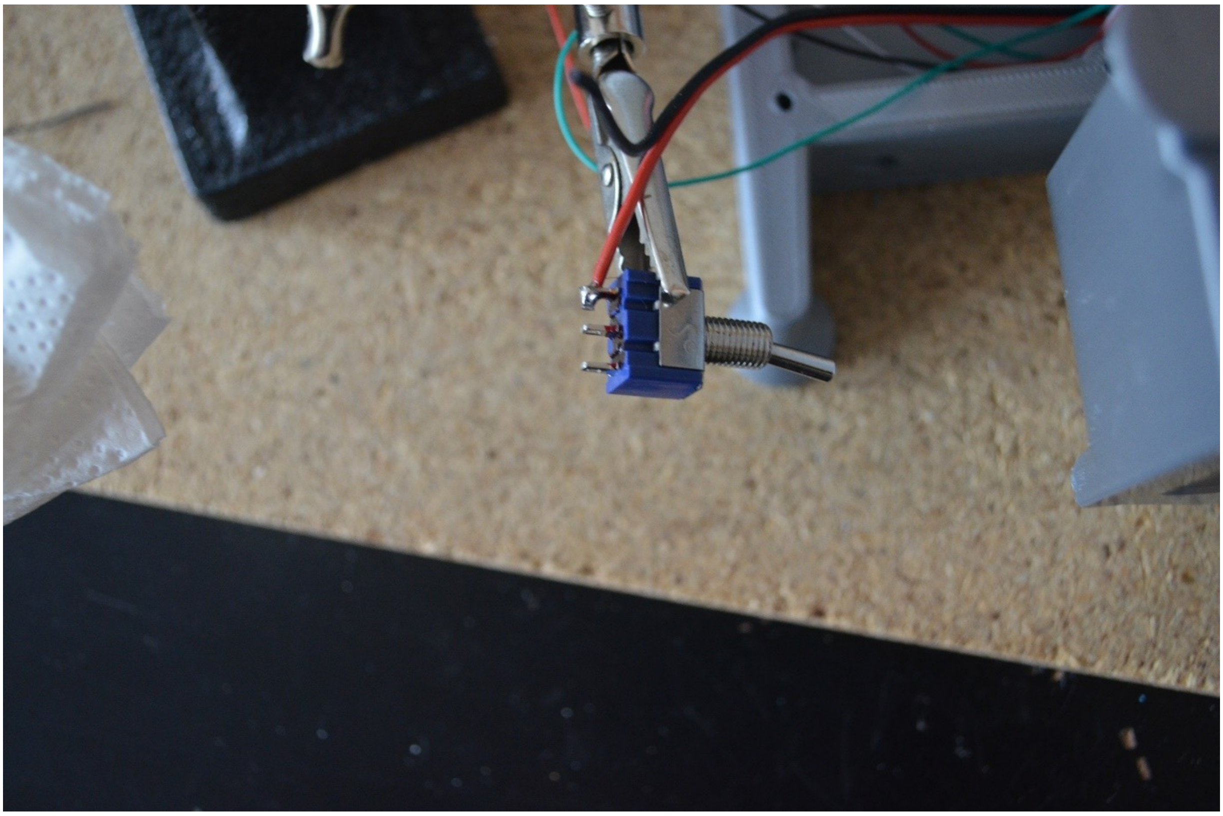


***Figure** 26 **:** Step 7 – Setting up the switch.*

Weld a second power wire to the adjacent tab of the switch. With this power wire and the ground wire from the PCB, install a connector to connect the LiPo battery.

Chosse any kind. The installation of this connector may require the use of crimping pliers (8 € at Hobby King). You can by also a pre-solded connector.

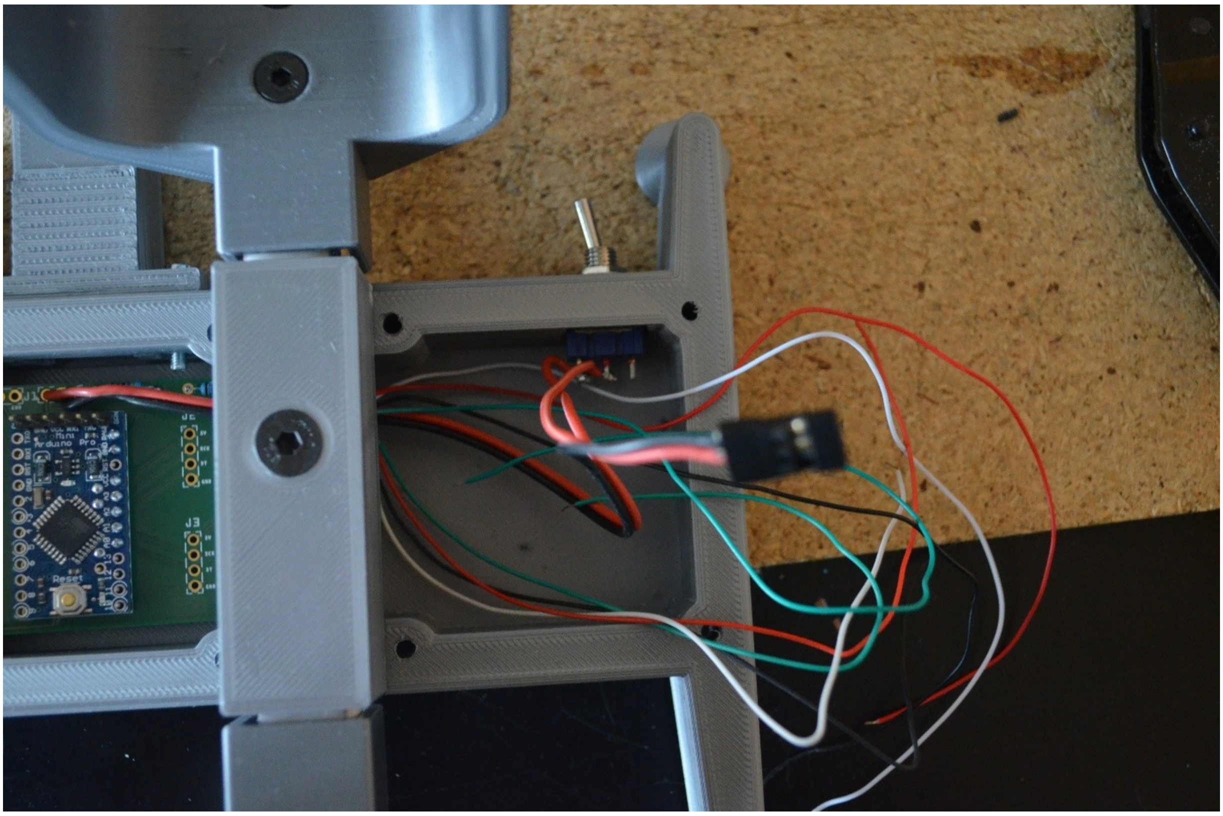


***Figure** 27 : Step 7 – The switch and the connector for the LiPo battery.*

4.4.1 Step 8 : Wiring the strain gauges.

Prepare 8 strands of Kynar or equivalent (30 AWG gauge). The interest of Kynar is that it is mono filament and relatively not brittle. You can also use servo wire.

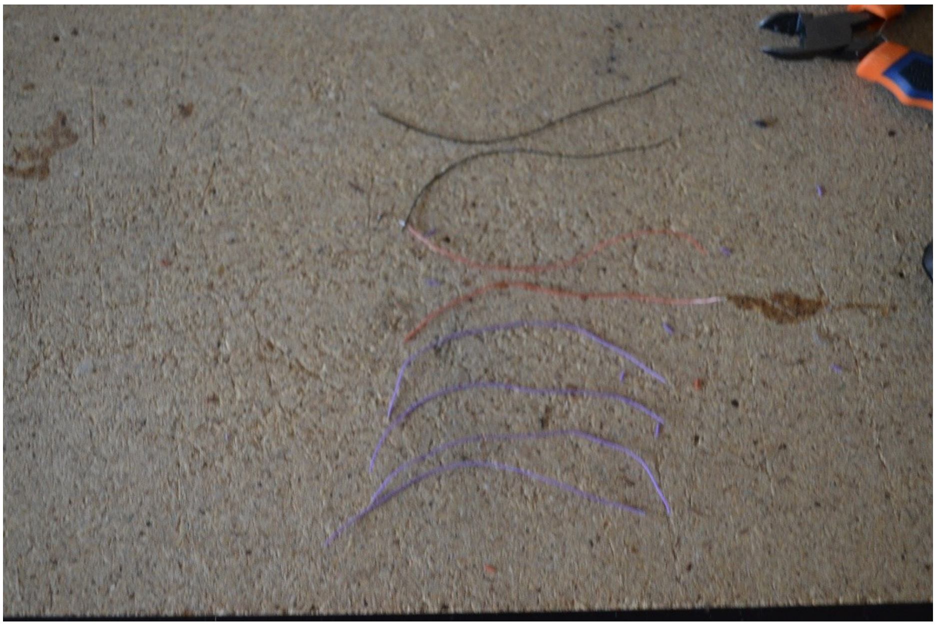


***Figure** 28 : Step 8 – Prepare 8 strands of Kynar (12 cm).*

Solder these strands of Kynar to the PCB as shown in the photo below.

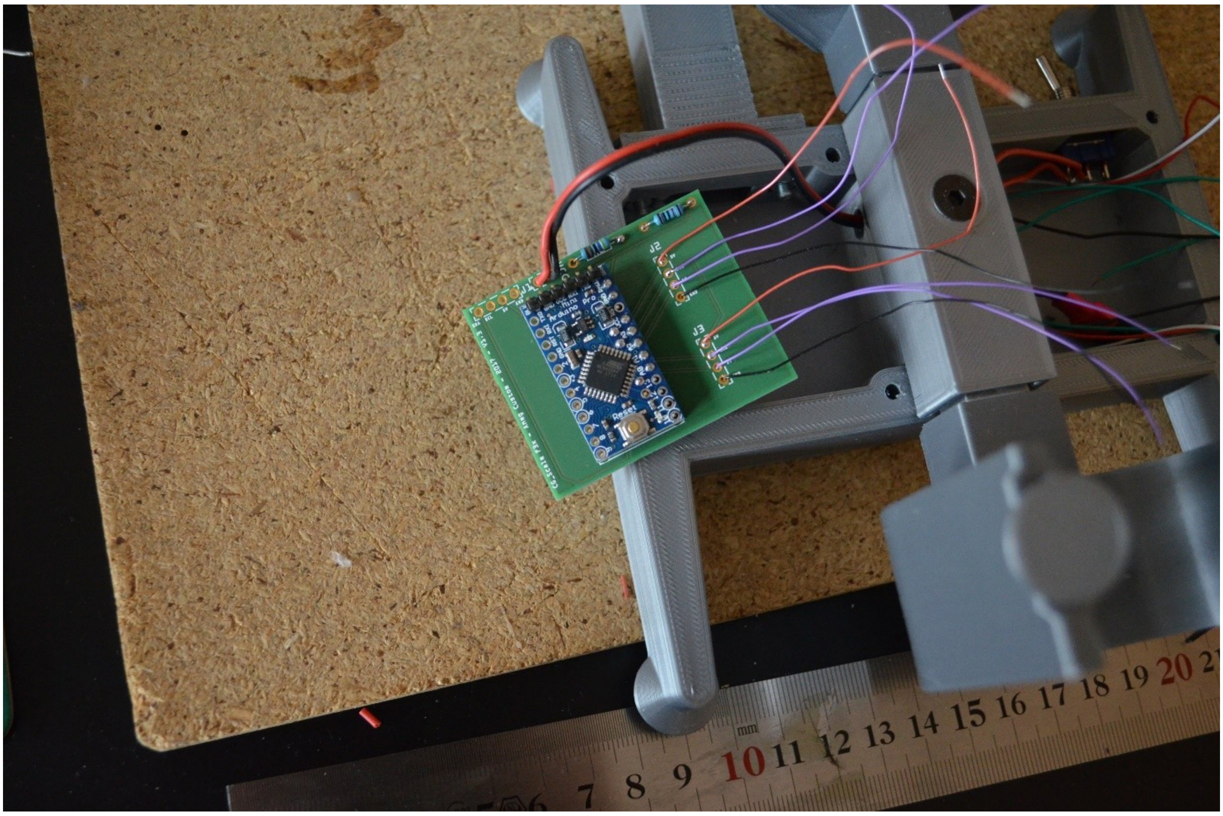


***Figure** 29 **:** Step 8 – Solder there strands of Kynar on the PCB.*

Then hold in place with a little hot glue to prevent one of the connections from breaking during assembly.

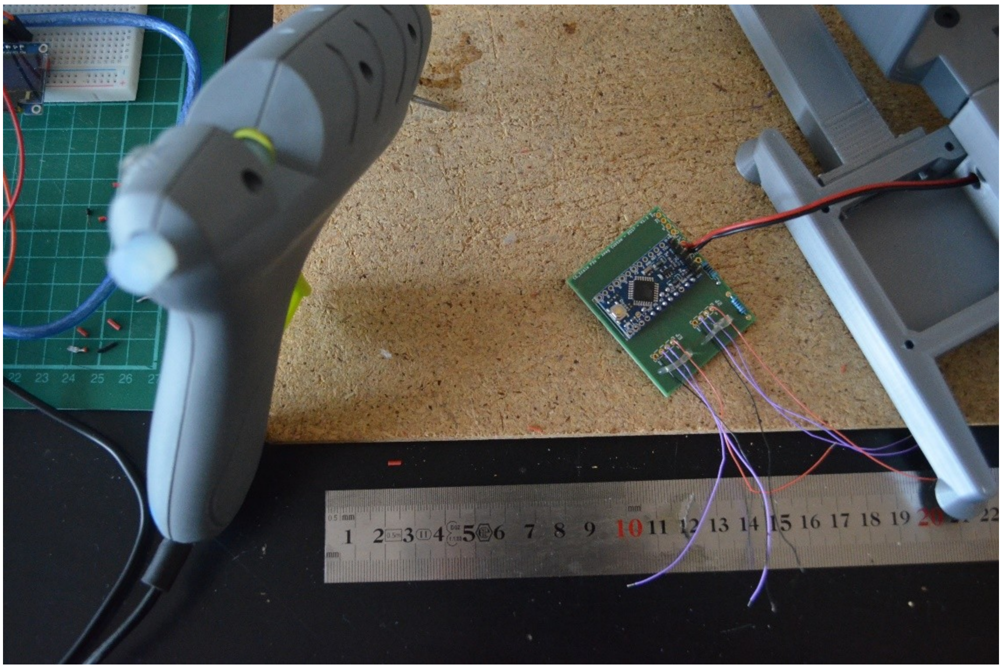


***Figure** 30 **:** Step 8 – Hold the 8 strandss of Kynar with hot glue.*

4.4.1 Step 9 : Prepare both HX711 board

Solder the screw terminals on both HX711 boards.

On each board, a terminal block must be soldered on the pins GND, DT, SCK, VCC and the other terminal block must be soldered on pins E +, E-, A- and A +.

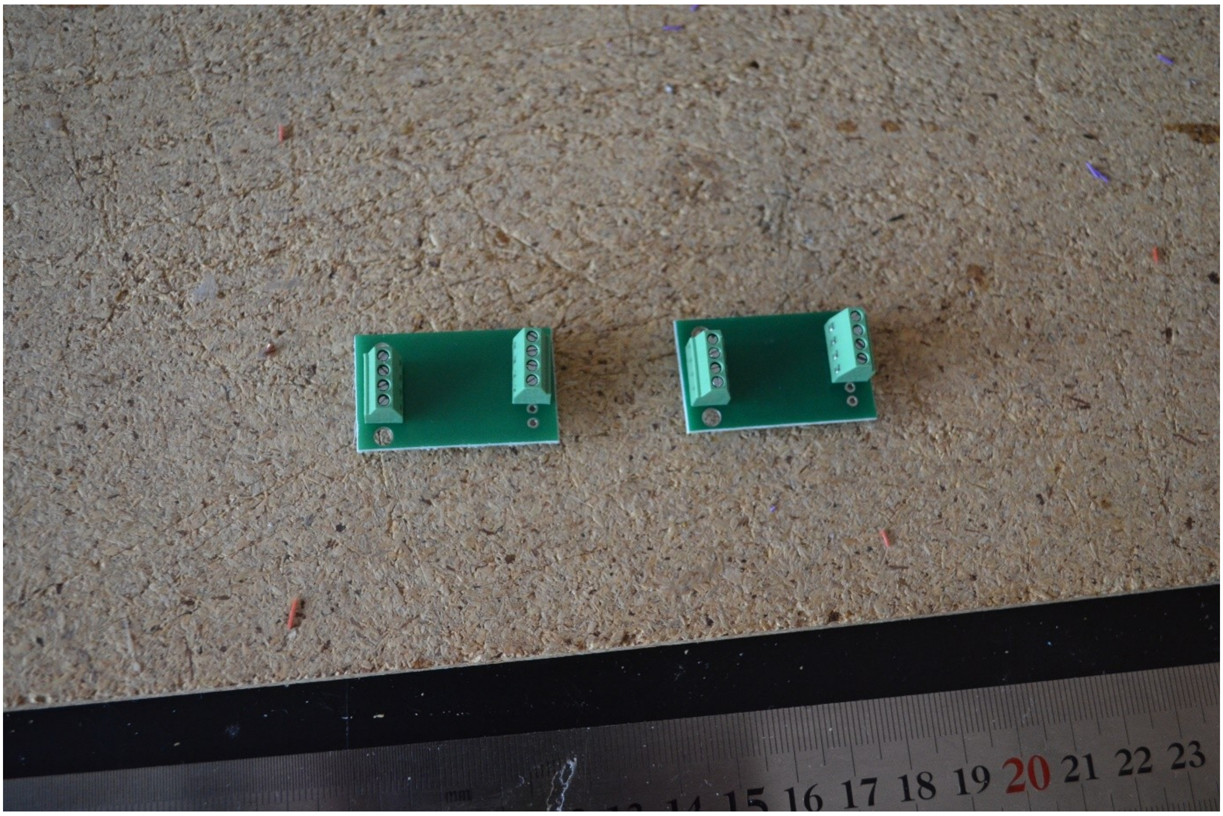


***Figure** 31 : Step 9 – Prepare both HX711 board.*

4.4.1 Step 10 : Integrate both HX711 board and power.

The HX711 board of the front strain gauge must be connected to the Kynar strands from J2 respecting:

- Black wire on GND,
- Red wire on VCC,
- A3 on SCK,
- A2 on DT.

The HX711 board of the front strain gauge must be connected to the Kynar strands from J3 respecting:

- Black wire on GND,
- Red wire on VCC,
- A1 on SCK,
- A0 on DT.

> **Be carefull to respect GND et VCC. HX711 are not protected against polarity inversions.**

The strain gauges are usually equipped with 4 wires of black, red, white and green colors. On each board connect the wires coming from the respective gauges (front gauge on the HX711 wired on J2 and rear gauge on the HX711 wired on J3) respecting :

- Black wire on E+,
- Red wire on E-,
- White wire on A-,
- Green wire on A+.
-

The following two pictures show the integration of the two gauges in the compartments of the balance base. The right and the left are heard looking at the scale by positioning the rear support of the balance in front of you

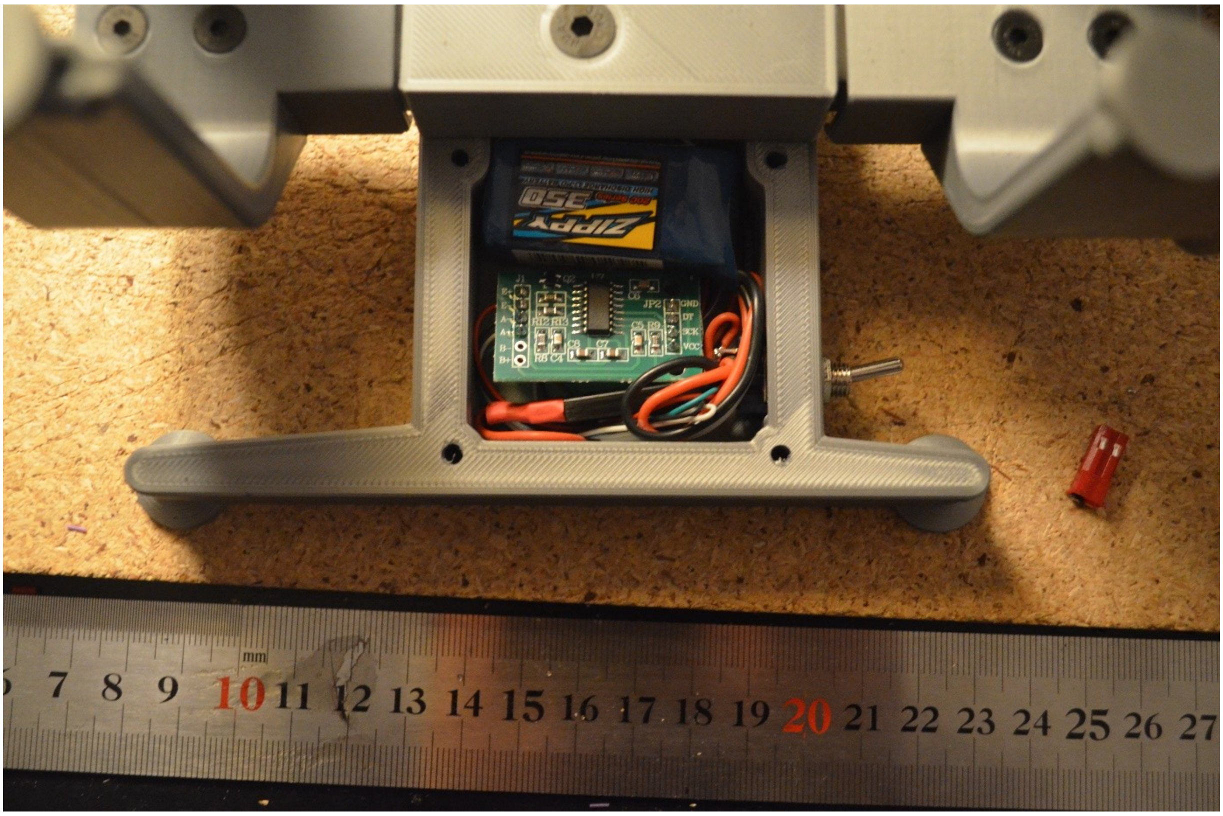


***Figure** 32 : Step 10 – HX711 and battery Integration into the right compartment.*

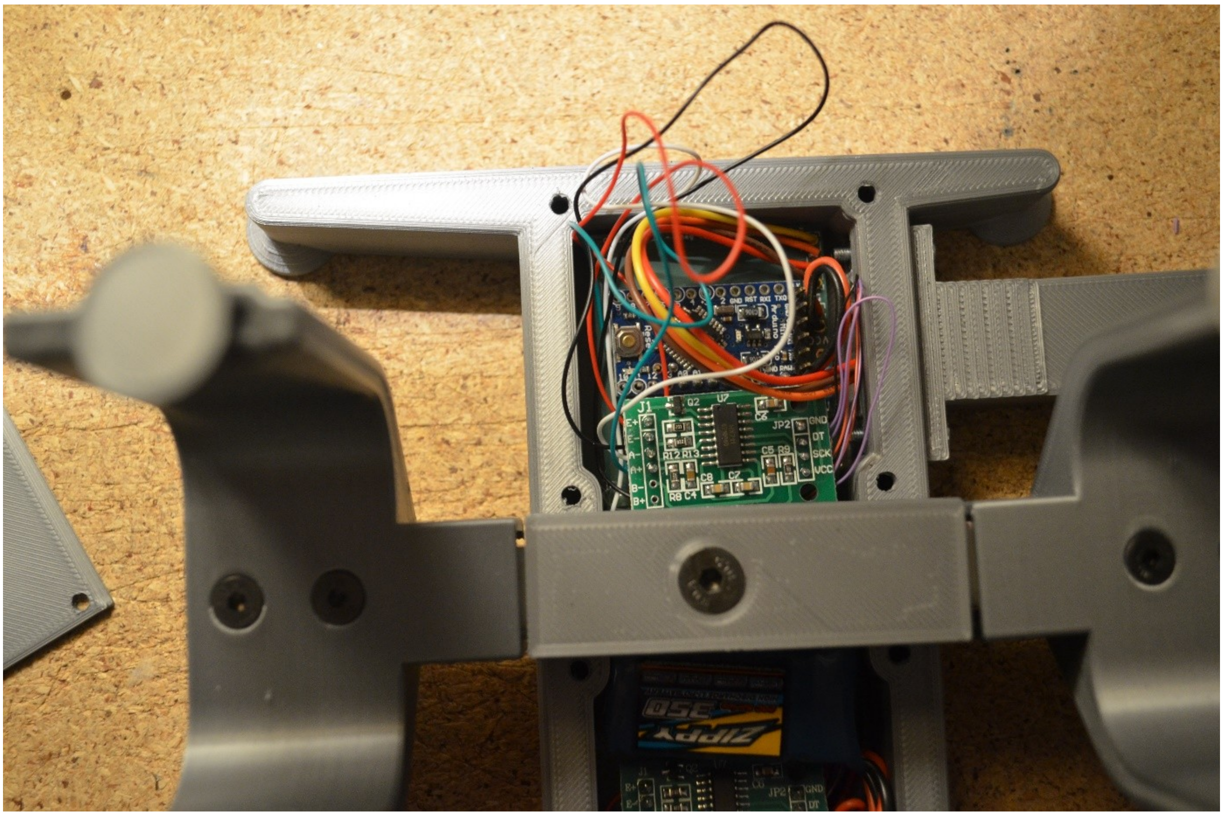


***Figure** 33 : Step 10 – HX711 and Arduino board integration into the left compartment.*

4.4.1 Step 11 : OLED 0,96'' display integration

Solder the 4-wire connector to the PCB and connect the connector to the OLED display.

> **Be carefull to respect GND et VCC. OLED display is not protected against polarity inversions.**

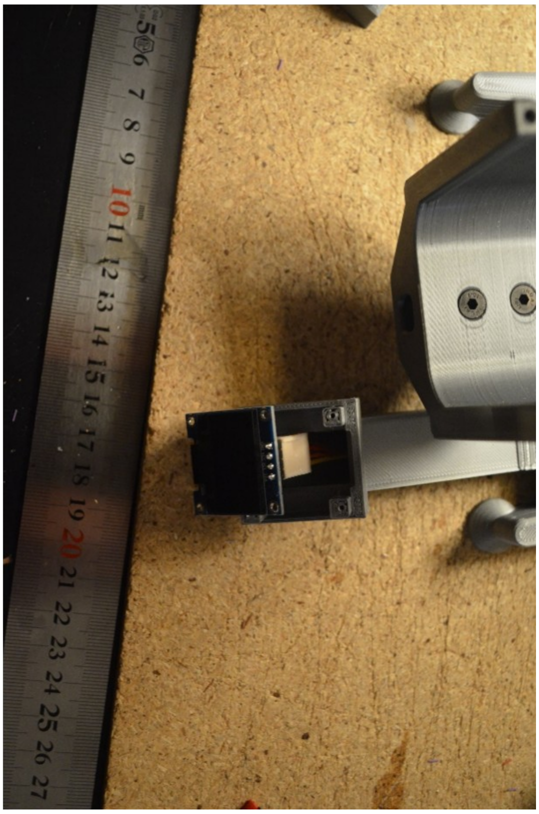


***Figure** 34 : Step 11 – OLED 0,96'' display integration*

Close the display case with 4 screws 1 mm round head.

> **Be careful to be very "delicate" in closing the screen cover. The OLED screen is extremely fragile and the glass breaks very easily, making the screen unusable. It is important to check that the cache closes without constraints, and to take it back to Dremel otherwise.**

And here is the balance and totally mounted. We can proceed to the installation of the Arduino environment, the download of the software and the calibration of the balance. You have to re-open the cache marked "F3F CG" to do this.

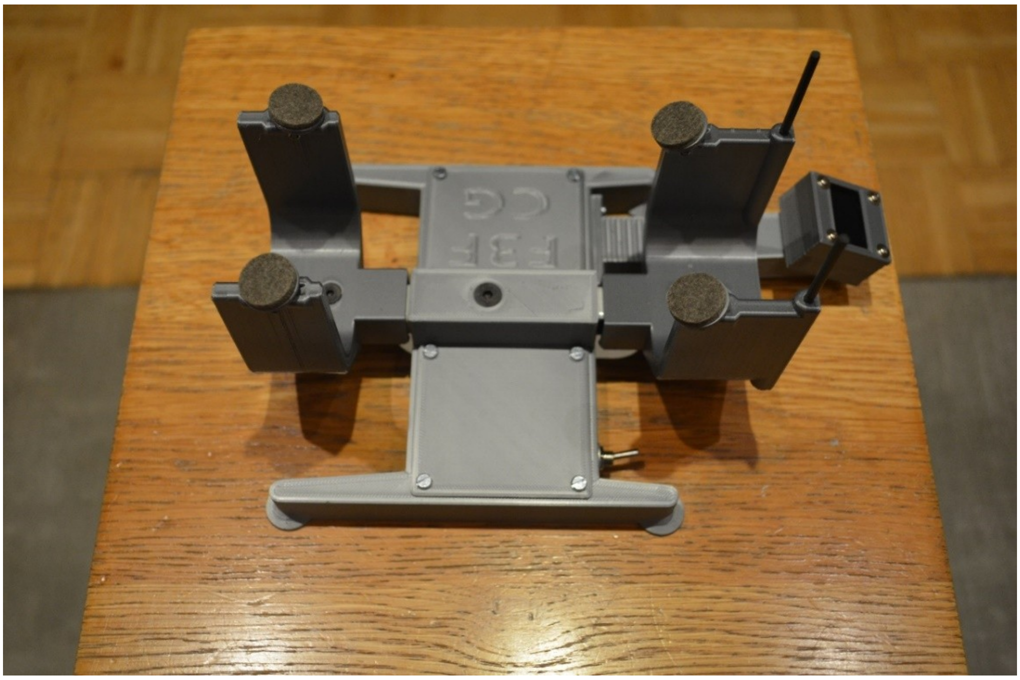


***Figure** 35 : Step 11 – The scale is finished, congratulations take a chair and taste a fresh beer !*

5 Getting Started with the Arduino Environment.

Download the latest version of Arduino software on this page. https://www.arduino.cc/download_handler.php

The download is in compressed zip format. When the download is finished, unzip the downloaded file: you get an Arduino-00XX directory whose structure must be kept. Open this folder: you should see some files and subdirectories.

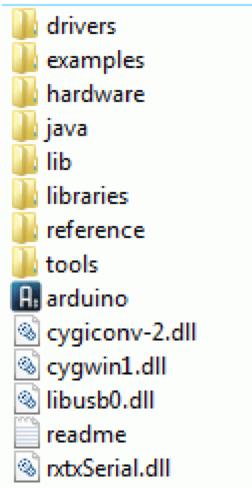


***Figure** 36 : Arduino IDE tree*

Locate the file with the Arduino icon that corresponds to the Arduino software. Note that this directory is "portable" and can be put where you want on your computer, or even on a USB stick or an external hard drive. The Arduino software will run without any problem.

5.1 Connect the Arduino board to the computer.

Now connect your Arduino board to your computer using your USB programmer. The LED (green or red depending on the Arduino board) power supply (PWR) should light up.

5.2 Install the USB driver for the virtual serial port.

When you connect the card to the computer for the first time, Windows should start the driver installation process.

In Windows Vista and above, the driver should normally be installed automatically. If it's not the case :

- Go to Start> Control Panel> System> Device Manager
- Go to Com Port and LPT> Locate Serial USB Port> Right Click> Update Driver
- Then select the driver in the / drivers / FTDI_USB subdirectory of the previously downloaded Arduino-00xx directory.
- Validate the differents steps.

You can now check that the drivers have been installed by opening the Control Panel> System> Device Manager. You should find in the section Ports LPT and COM a "USB Serial Port": it is the USB port of the Arduino board. Note in passing the port number.

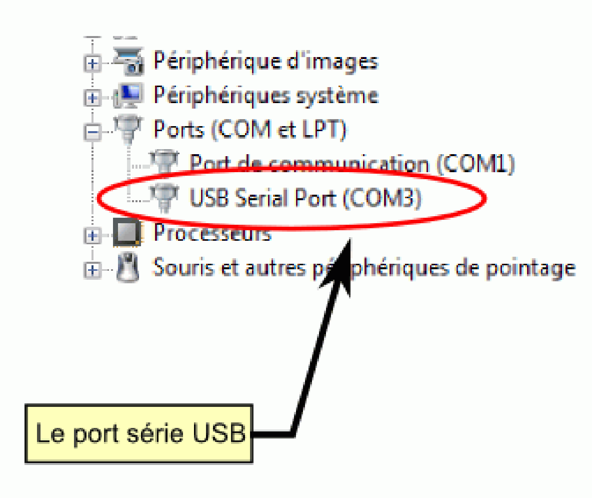
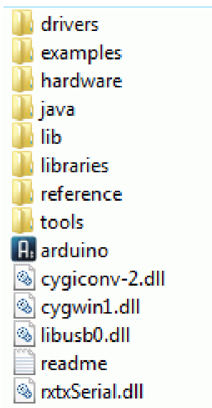


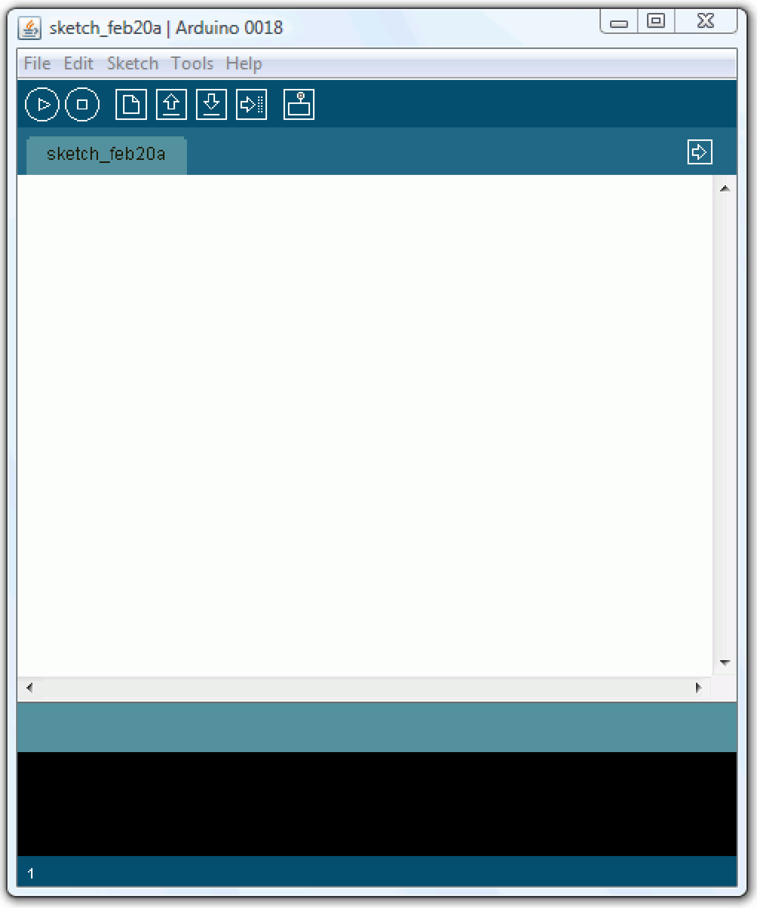
***Figure** 37 : USB port connected to Arduino board via USB / TTL programmer.*

5.3 Run the Arduino IDE

Now, launch the Arduino IDE by double-clicking the Arduino icon twice in the previously downloaded directory:

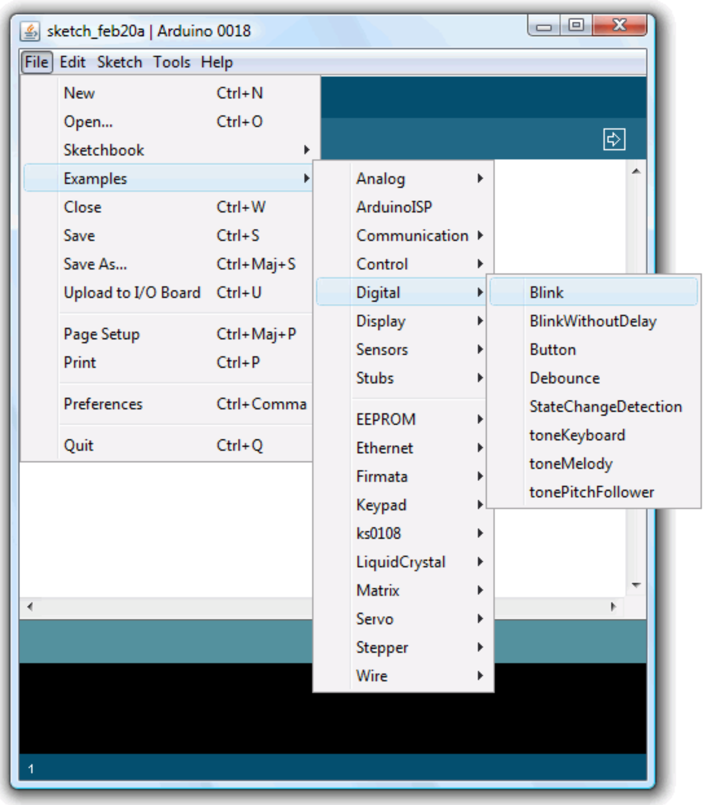


The following windows could appears.

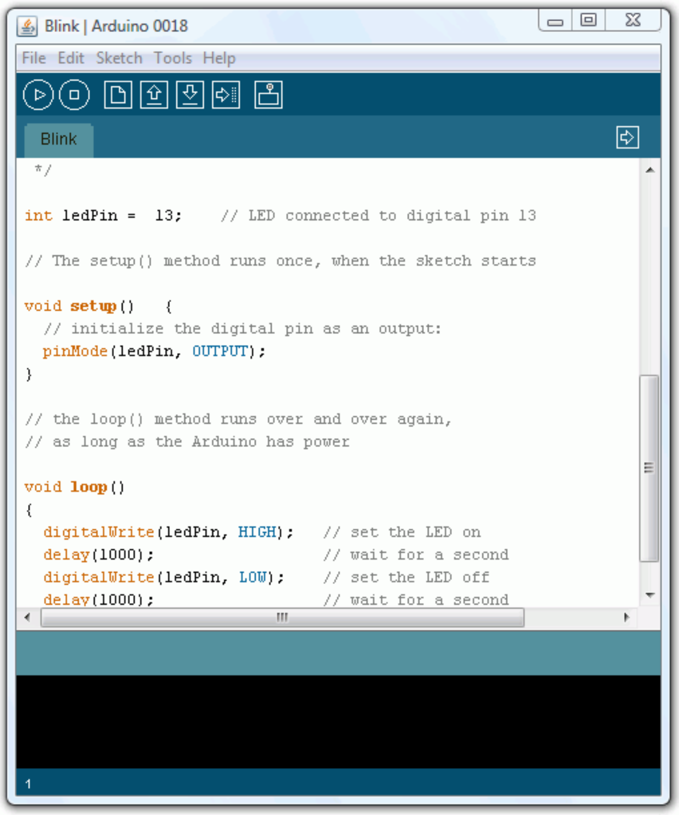


5.4 Open the Blink example program

Now open the example program "Blink" which flashes the LED of the card connected to pin 13. To do so, go to menu File> Examples> Digital> Blink

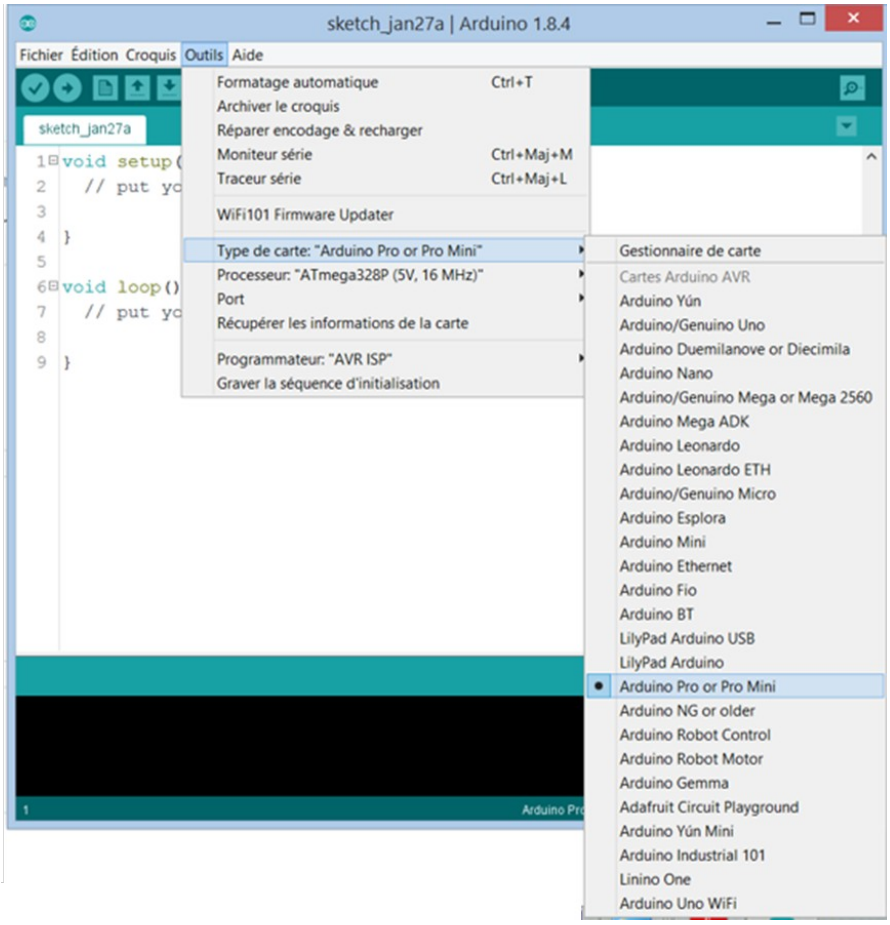


The program code should now appear in the editor window. Note that you could have written yourself a test program but to begin, it's not the simplest way to discover the IDE ...



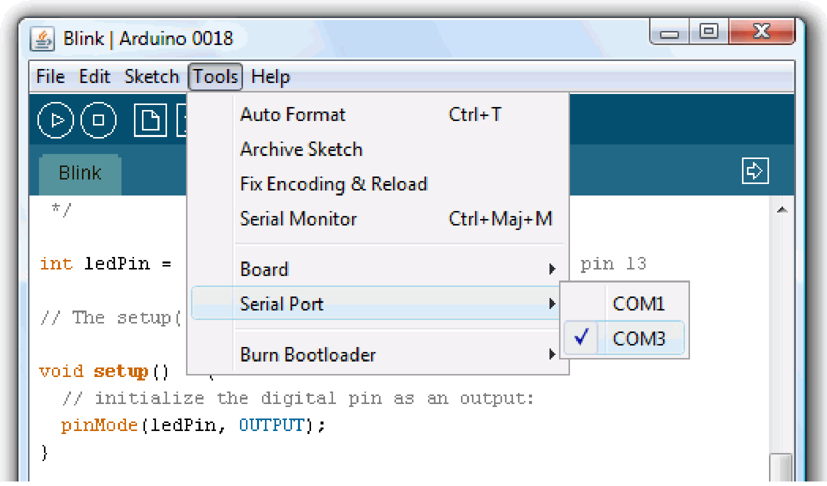
5.5 Select your Arduino Board.

You must now select your Arduino board from the Tools> Board menu: for new boards with an ATmega328, select Arduino Pro or Pro Mini.



5.6 8. Select the serial port.

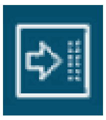
Now you must select the serial port used for communication with the Arduino board from the Tools> Serial Port menu. This will probably be COM port 3 or higher (COM1 and COM2 ports are usually reserved for hardware serial ports). To find out which port it is, you can disconnect your Arduino board and reopen the Tools> Serial Port menu: the entry that has disappeared is probably that of the Arduino board port. Reconnect the board, reopen the Tools> Serial Port menu and select the serial port that reappeared :



5.7 9. Download the Blink program inside your Arduino board.

ow, once you have selected the correct serial port and the correct Arduino board, click the UPLOAD button on the toolbar, or select File> Upload to I / O board (File> Transfer to map). Just click on the "UPLOAD" button on the Arduino software toolbar.

Note that the transfer of the program starts with a recompilation of the sketch (under the Arduino IDE a program is called a sketch).

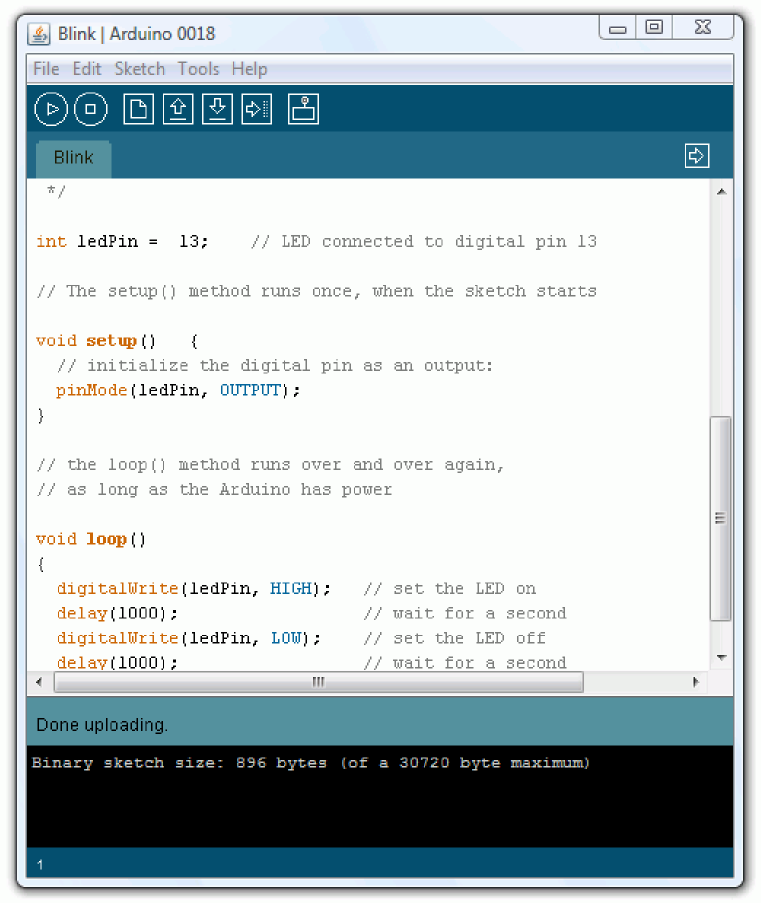


You must press the reset button on the card just before starting the transfer and hold it until the message below appears in the bottom window of the Arduino IDE.
At this moment, release the "reset" button: MAGIC INSTANT, you must see the download start with the display of a progress bar in the form of a sequence of characters "*". The end of the download is announced by the display of a message « Thank you » !!!

***Figure** 38 : Hold the "reset" button on the Arduino board until you see the message above and then release the "reset" button to start the download.*

You should see the LEDs of the RX and TX lines of your USB-TTL programmer flashing rapidly, indicating that the program is well transferred. During transfer, the button turns yellow and the Arduino software displays a message indicating that the transfer is in progress :

Once the transfer is complete, the Arduino software must display a message ("Done uploading") indicating that the transfer is well done, or show error messages ...



Once the transfer is complete, the bootloader is active a small second ("listen" to see if a new program arrives ...) once the card is reset at the end of the transfer; then the last program programmed in the card runs. So you should see the LED on pin 13 flashing (orange color).

If that's the case, congratulations! You have successfully programmed and operated your Arduino board.

6 Install additional libraries

Two libraries (or three libraries depending on the version you want to build) are to be installed to compile the .ino file that will be downloaded in the arduino card:

- The HX711_ADC library, which allows you to interface with ... HX711,

If you build the OLED I2C version :

- Adafruit_SSD1306
- Adafruit_GFX

If you build version 1602 I2C :

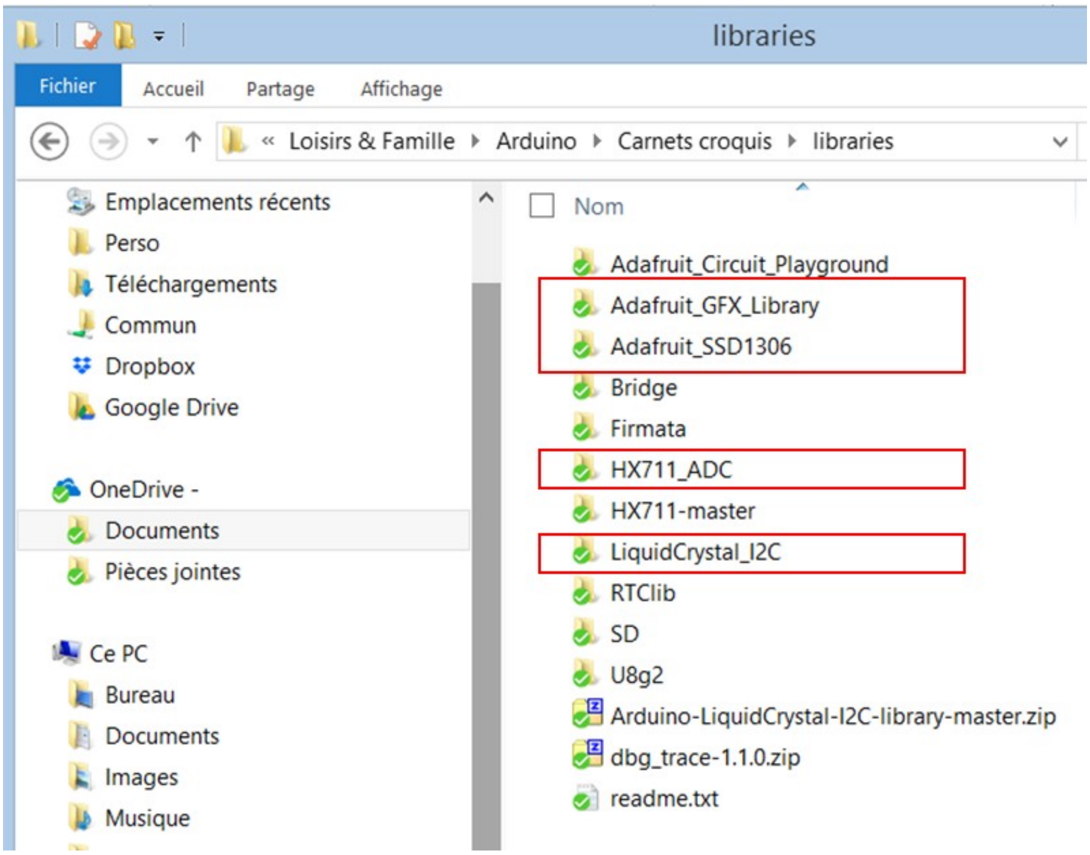
- LiquidCrystal_I2C

If you build the 1602 base version (installed by default in the arduino environment) :

- LiquidCrystal

All of these libraries are installed from the Library Manager in the Sketch / Include Library / Manage Libraries ... menu. It allows you to find a detailed list of the libraries present, to update them when an evolution has been detected in your library. creator, and to install new ones automatically. This list can be sorted by theme, or by state (installed, to update ...).

By default, these libraries are installed at the location of your sketchbook.



7 6 Download software and calibrate ADC

> **Note that for all the following operations, the toggle switch must be in the "off" position because it is your PC that powers the Arduino board..**

7.1 Initialize the calculation parameters of the CG.

In line 77 and 78 (case of sketch CG_scale_I2c_Oled) of the sketch two values are to modify according to your balance:

```
74
75 //*** configuration:
76 //*** set dimensional calibration values:
77 const long WingPegDist = 1214; //calibration value in 1/10mm,
78 const long LEstopperDist = 306; //calibration value 1/10mm,
```

WingPegDist is the distance in 1/10 of a mm projected between the two axes of the pads of the front support and the rear support.

LEstopperDist corresponds to the distance in 1/10 mm projected between the axis of the front support and the shutter rod of the wings of the glider.

These two distances are to be measured with a caliper.

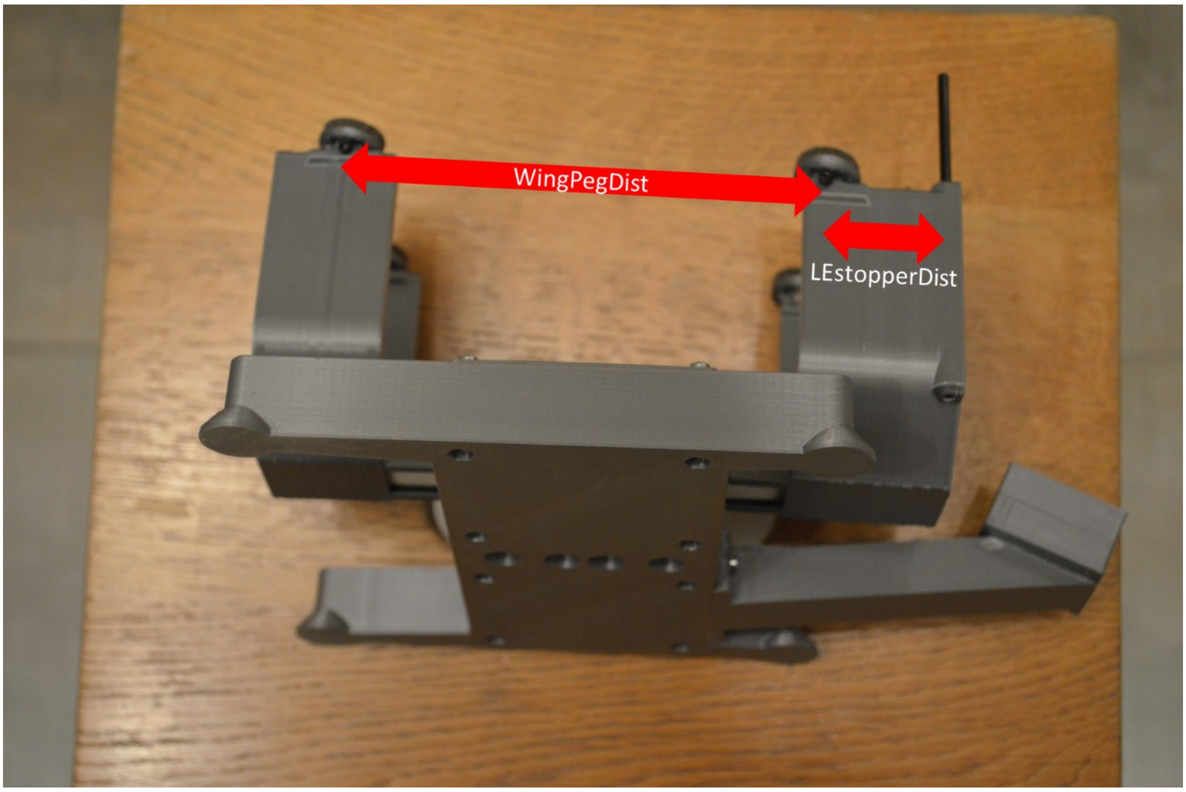
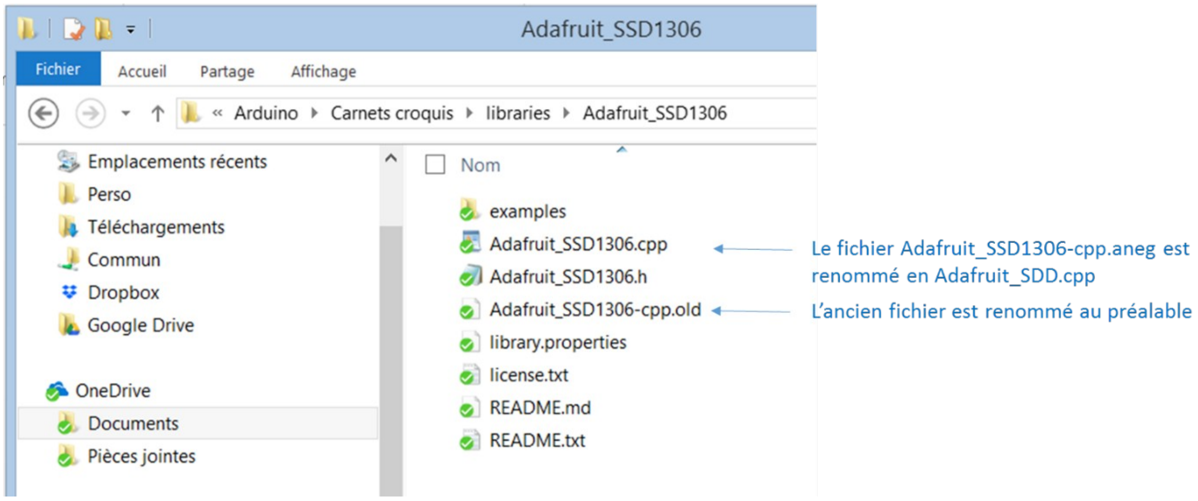


***Figure** 39 : Etape 11 – WingPegDist & LEstopperDist.*

7.1 Download the software.

Once your Arduino environment is set up and the necessary libraries are installed, all you have to do is download the CG_scale_I2c_Oled sketch (or CG_scale_I2c_1602 or CG_scale depending on your version) in the Arduino board. To do this proceed as for the sketch Blink and wait for the end of the operation. After the download, the card will restart and a home screen must appear on the screen Oled 0.96 ".

> **By default, the balance's home screen is configured with the Adafruit logo. If you want to have on the home screen the logo of the ANEG, it is necessary to copy the file Adafruit_SSD1306-cpp.aneg which you will find in the directorium "Arduino files" of the Github, by renaming it Adafruit_SSD1306.cpp at the level from the Adafruit_SSD1306 library, taking care to save the old file in Adafruit_SSD1306-cpp.old. You can also design your own logo by following the procedure in Chapter 6.**



7.2 6.2 Calibrating the HX711

Use two weights or two ballasts whose weights you know by weighing them on another scale. Two solutions are at your disposal, the first being the simplest and the fastes.

7.2.1 First way : Use the Calibrate_bothHX711.ino sketch

Download the sketch Calibrate_bothHX711.ino in the arduino card, open the serial monitor and follow the instructions displayed on the monitor, placing a reference weight on the gauge "front" to match the measured weight with the reference weight and repeat the operation by putting a weight on the gauge "rear" :

- The 'F' (for front) and 'R' (for rear) keys are used to select the gauge to be calibrated.
- The 'l' (-1.0) and 'L' (-10.0) keys allow to decrease the calibration factor of the corresponding quantity.
- The 'h' (+1.0) and 'H' (+10.0) keys increase the calibration factor of the corresponding quatity.

Once the two gauges have been calibrated, report the values of the two calibration factors in line 80 and 81 of the sketch CG_scale_I2c_Oled.ino and reload this sketch in the arduino.

```
80 const float ldcell_1_calfactor = 897.0;
81 const float ldcell_2_calfactor = 745.0;
```

7.2.2 Second way : use directly the CG_scale-I2c_Oled sketch.

In line 94 of the sketch (case of sketch CG_scale_I2c_Oled), start by setting the variable output to 0 and to upload the sketch in the Arduino board.

```
92 void setup() {
93   //***
94   output = 1; //change to 1 for OLED, output = 0 for Serial terminal
95   //***
96
```

Open the serial monitor in the Tools menu. Periodically the weights measured by each HX711 via the strain gauges (weight_Ldcell_1 and weight Ldcell_2) are displayed.

We will describe the procedure for the HX711 connected to the front support. Put one of your reference weights on the front support and read the value measured by the scale.

If the measurement displayed on the monitor of the serial monitor is greater than the reference weight, in line 80 of the sketch, change the value assigned to ldcell_1_calfactor by assigning it a lower value (from -50 for example) and reload the sketch and note the result of your modification. Repeat until the value read on the serial monitor is close to the value of your reference weight.

```
80 const float ldcell_1_calfactor = 897.0;
81 const float ldcell_2_calfactor = 745.0;
```

Repeat the process by placing your second reference weight on the back support by changing the value assigned to ldcell_2_calfactor.

Whatever the solution implemented, when both HX711 are calibrated, you can disconnect your programmer from the Arduino board, close the cover of the base marked "F3F CG" with 4 screws M3 x 20 mm round head and your scale is ready to us.

8 How to use the scale

Operation is extremely simple :

1. Turn on the scale and wait for "Batt: X.YY V" and "Wt: 0000" to appear on the screen.
2. If "Wt:" is not at zero, turn off the scale and go back to 1.
3. Place the glider in the center of the supports, the wings on the rods of the front support.
4. Total weight and location of the CG (distance from the leading edge) will be displayed.

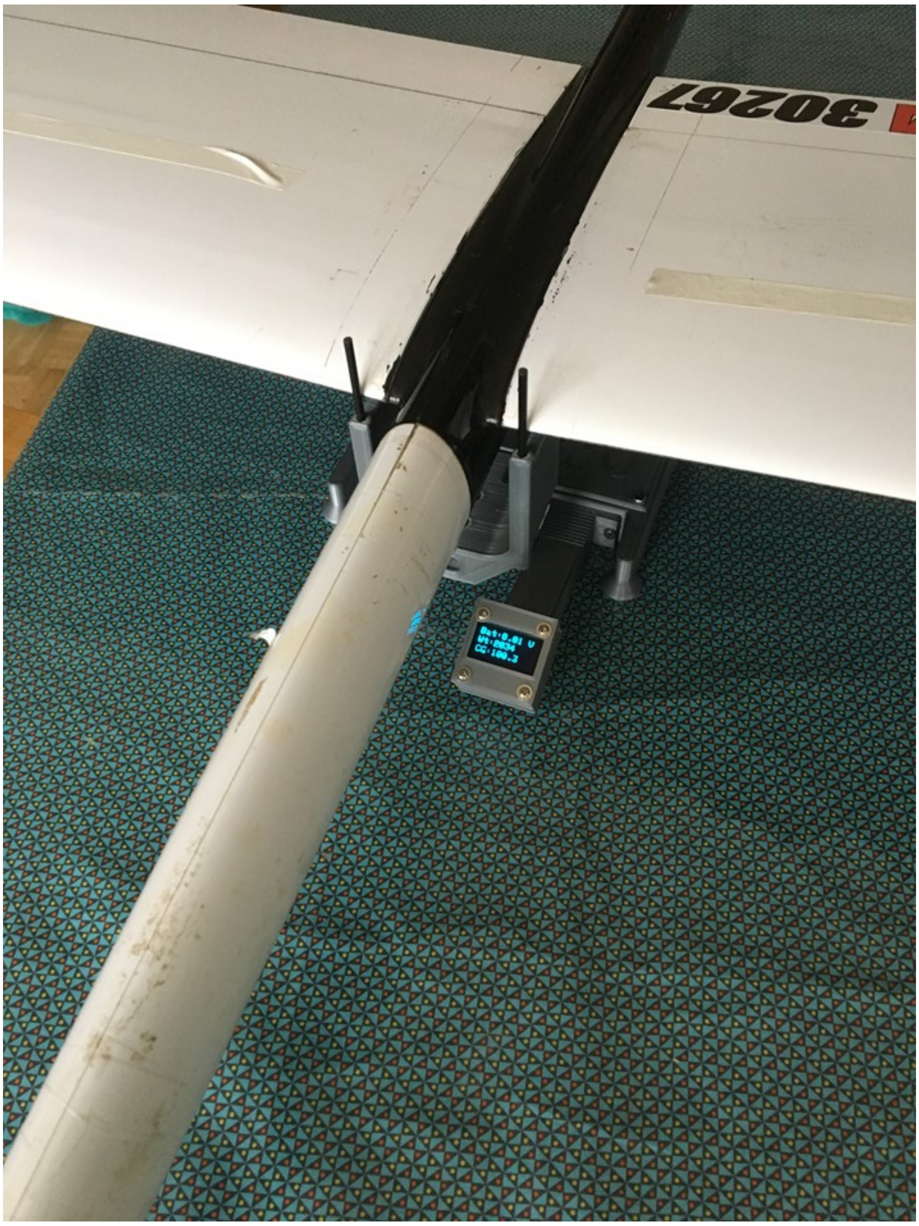
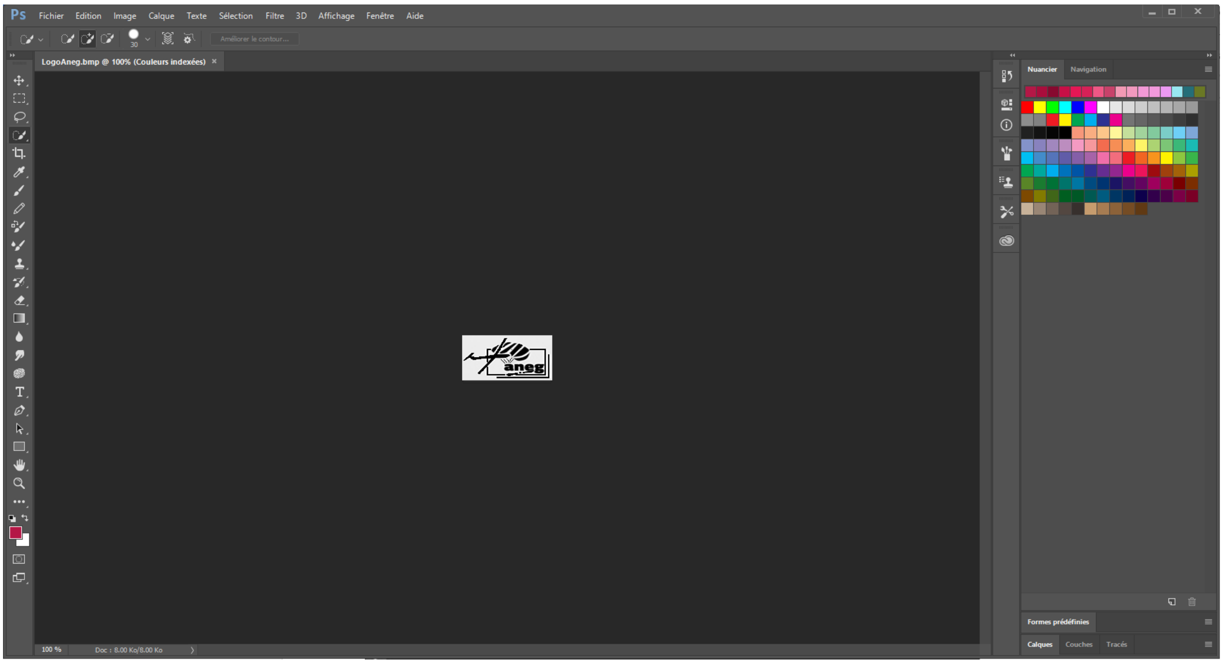


***Figure** 40 **:** The scale in operation.*

9 Change adafruit OLED Logo.

This chapter has been reused and modified from the site https://www.instructables.com/id/Change-Adafruit-Library-OLED-Logo/

9.1 Step1 : create your logo graphic



Use the graphics editor of your choice to create a 8 bit bitmap of your logo in this instance making the size 128 x 64 or whatever size your OLED or LCD is.

It must be saved a a 8 bit bitmap as no other file format or type will work.

Save the bitmap and move onto converting it into code

> **Also make sure the background is actually white (#ffffff) and the foreground (the logo or image) is black (#000000) these color steps are important as the software we use to convert the graphic to code won't work properly if these two color values are not used..**

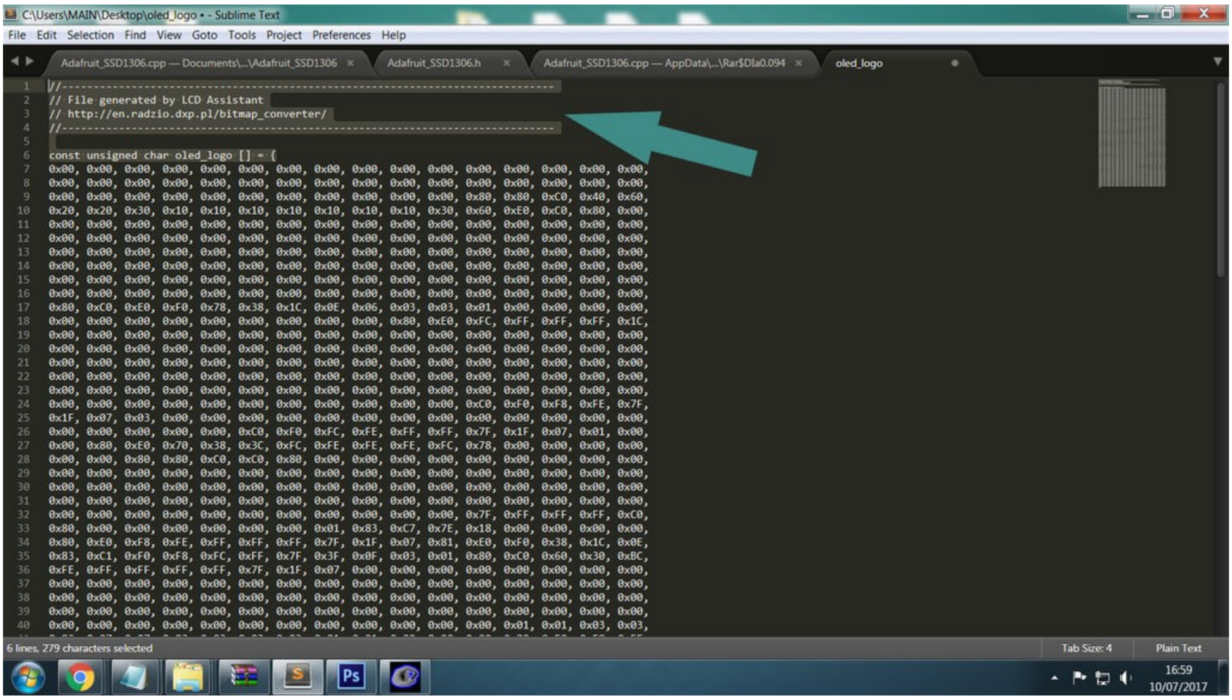
9.2 Step 2: Use LCD Assistant to convert your bitmap to code.

Download the free bitmap to code converter called LCD Assistant the link for it is http://en.radzio.dxp.pl/bitmap_converter/.

Open the program and then from the file menu load in your bitmap graphic. LCD Assistant should automatically detect the size of the graphic in this case 128 x 64 and for every other setting leave everything else as is.

Then again from the file menu select output and save the output file to somewhere handy on your computer such as the Desktop for example

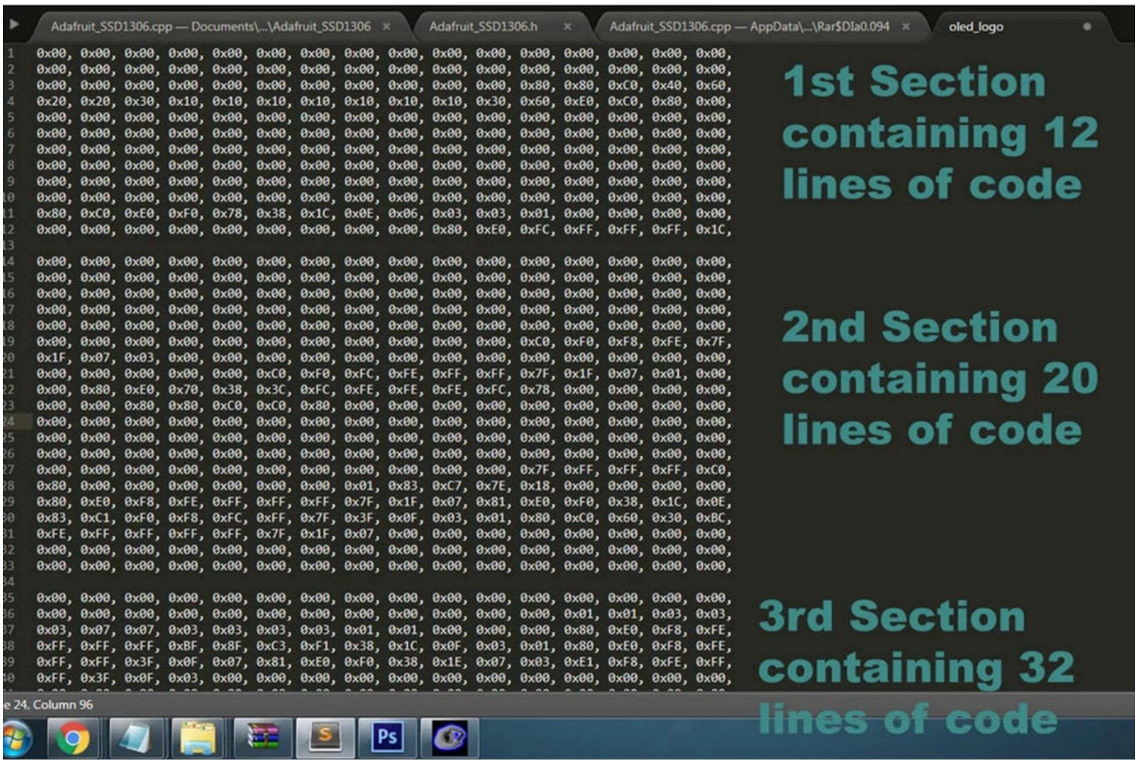
9.3 Step 3 : Edit Logo Hex Code Output File



Open the file generated by LCD Assistant in any text editor even Notepad will do. Here I am using notepad.

You want to delete everything apart from the actual hex code so remove the part at the top of the file so the first line should just be code **and at the bottom delete the ending " }; " symbols**.

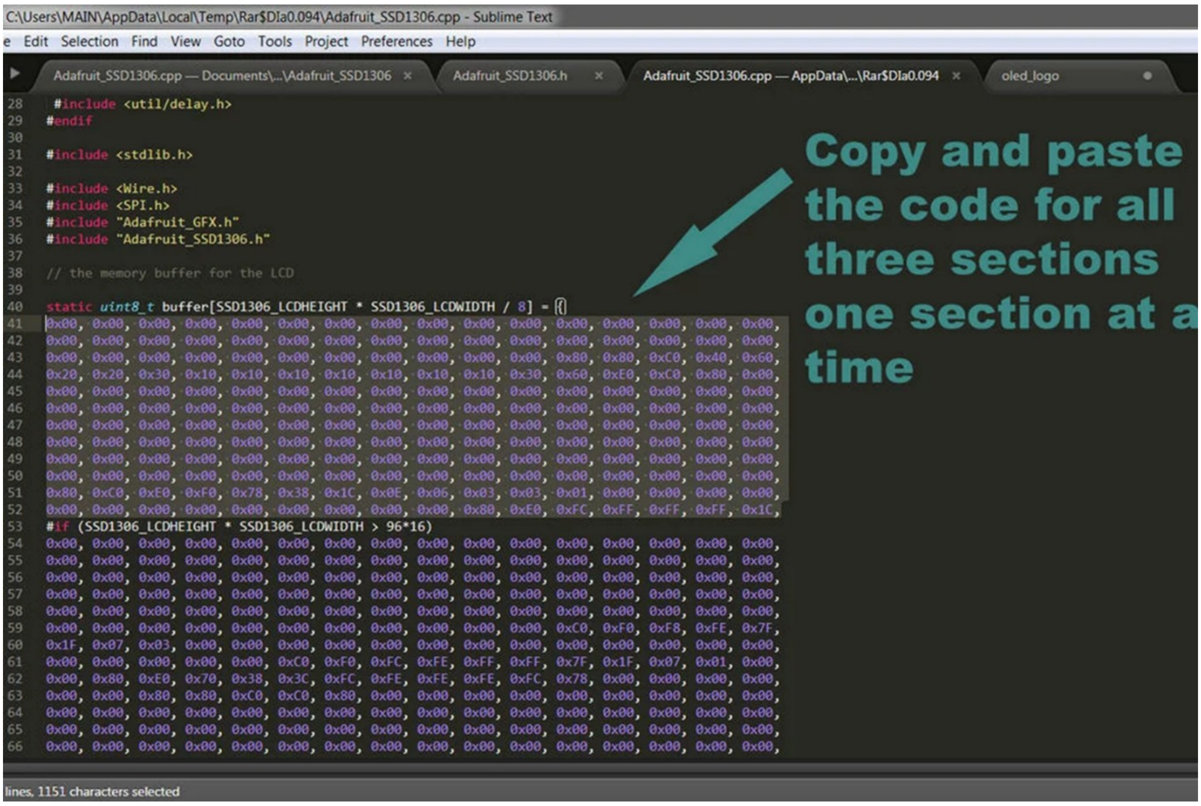
9.4 Step 4: Break Logo Code into three sections.



In this step we just want to break the logo hex code into three sections so we end up with three separate sections. The first of which will have 12 lines of code, the second 20 lines of code and the third secton 32 lines of code to make a total of 64 lines of code which is the same height as your OLED... 64. (If you are using a different size then you will have more or less lines of code to match the height of your display)

All you do to separate these sections is to add a blank line in between them.

9.5 Step 5 : Copy your logo code to replace the adafruit logo code.



For this step you need to already have the Adafruit_SSD1306 library downloaded and installed. If you are using the Arduino web ide rather than the standard version that runs on your computer you will need to add the library once customized as a custom library because you are unable to edit the library using the web / cloud version and be sure to include the custom version into your sketch and not include the pre installed version or this won't work.

In the Adafruit_SSD1306 library, copy the Adafruit_SSD1306.cpp file to Adafruit_SSD-cpp.old.

> **Important: do not rename the file to <something.cpp>, because at compile time the arduino IDE tries to compile all the .cpp files contained in a library and you would have multiple redefinitions and thus an impossibility to compile.**

Inside the Adafruit_SSD1306 library open the file named Adafruit_SSD1306.cpp using a text editor.

Scroll down into you find the following section starting with this line:

static uint8_t buffer[SSD1306_LCDHEIGHT * SSD1306_LCDWIDTH / 8] = {

You will then copy and paste replacing the code with your new logo code into each section:

static uint8_t buffer[SSD1306_LCDHEIGHT * SSD1306_LCDWIDTH / 8] = {

CODE SECTION ONE GOES HERE

#if (SSD1306_LCDHEIGHT * SSD1306_LCDWIDTH > 96*16)

CODE SECTION TWO GOES HERE

#if (SSD1306_LCDHEIGHT == 64)

CODE SECTION THREE GOES HERE

#endif #endif };

Leave everything exactly as it is being careful to only replace the code in each section and nothing else otherwise it won't work.

Once all that is done save the file.

Recompile the sketch CG_scale_I2c_Oled and upload it. Assuming everything has been done correctly your logo will now replace that of Adafruit Industries.

10 Troubleshooting

Symptôme	Résolution
Bakelite smell at the balance connection.	You probably reversed the connection of the GND and the VCC, either on one of the HX711s or on the OLED screen. Replace the component
The TTL-USB programmer can not be detected by the PC	Your USB driver is probably not correctly installed. The FTDI driver comes with the Arduino IDE and is located on the path C: \ <Your installation path> \ Arduino-1.8.4 \ Drivers \ FTDI USB Drivers. Try to reinstall it. The driver is also available on the FTDI Chip website at : http://www.ftdichip.com/FTDrivers.htm
Your arduino pro mini card does not have the A7 pin in the right place.	Wrap the A7 pin on your board, to the A7 pin of the PCB before soldering the arduino board to the PCB.
My Arduino board is well connected, the serial port is clearly visible and configured in the "tools" menu, as well as the right board, but the download does not start.	Hold down the "reset" button on the Arduino board until the beginning of the download appears and release the "reset" button (see. Figure 38).
My Arduino board is well connected, the serial port is clearly visible and configured in the "tools" menu, as well as the right board, but the download does not start.	Check the wiring of the USB-TTL programmer to the Arduino board, not all arduino pro mini cards have the same distribution of connection pins to the USB-TTL programmer.
After connecting the battery and opening the circuit with the toggle switch, the "power" led on the arduino card glowed stealthily and went out. The "power" led no longer lights up by doing the operation again.	The arduino card's input regulator probably "burned" due to a short circuit in the power circuit, either due to a false contact (GND and +) on the battery connector, or to the VCC on the PCB (wires are touching). It is still possible to recover the card by feeding it in 5V, via a step-down regulator (less than 1 euro), connected to the VCC and GND pins of the arduino card. The display of the battery voltage will be fanciful ... but you can reuse the card and the entire assembly.