# Arduino Robotic Arm Controlled by Touch Interface

by Maurizio Miscio

Hello. Today I'm here to see with you how I built a robotic arm with Arduino and a graphic touch interface.

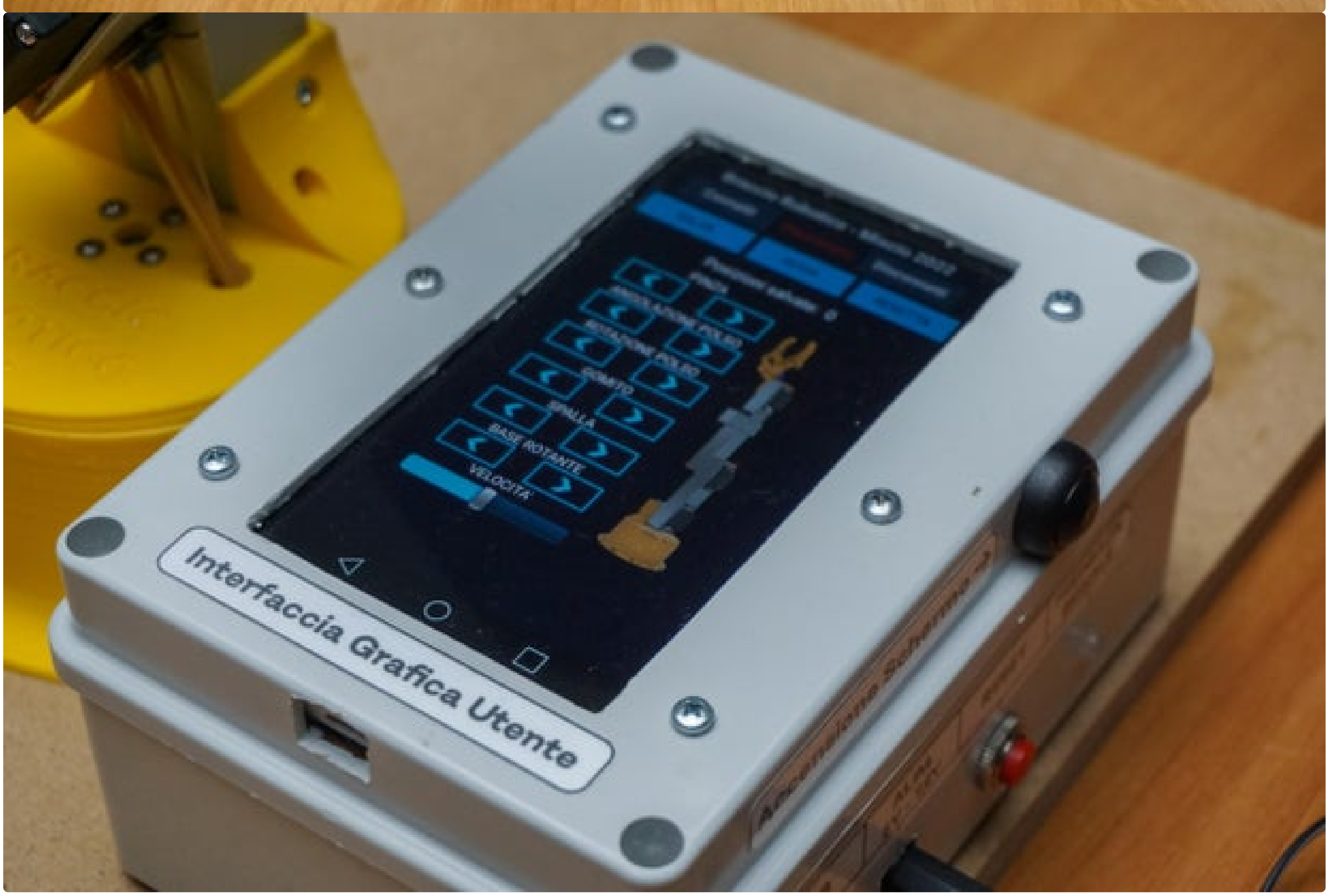
I want to say that for me this is not a simple mixture of various components that build a finished object together but it is something much more, a real project that started from an idea born from my passion months ago and ends today with the publication of this article.

Thanks to the graphic interface it is possible not only to move the various joints in all possible directions but also to program the movements to be performed repeatedly

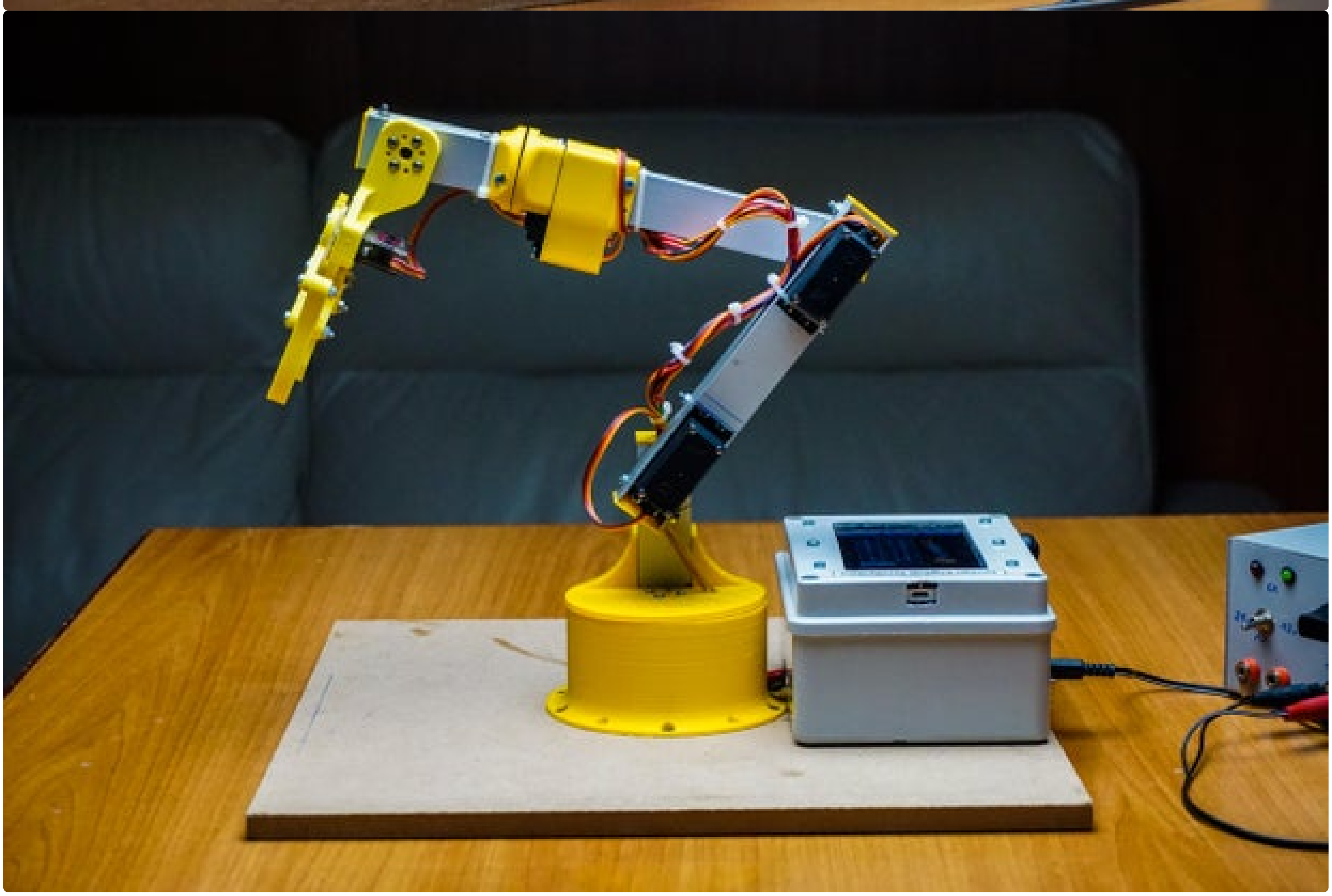
But now, get comfortable and let's see what we need to make it!

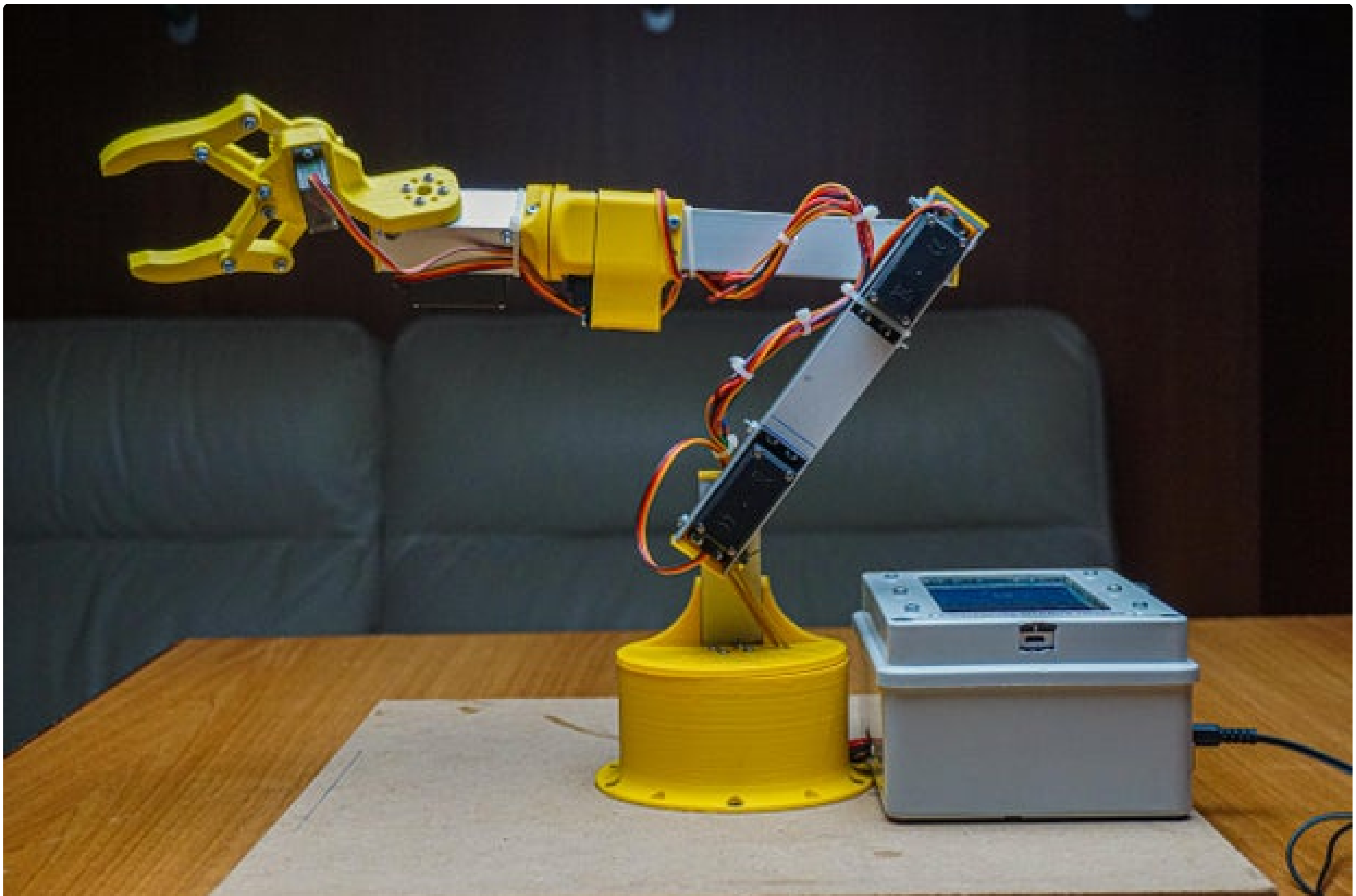
**Supplies:**

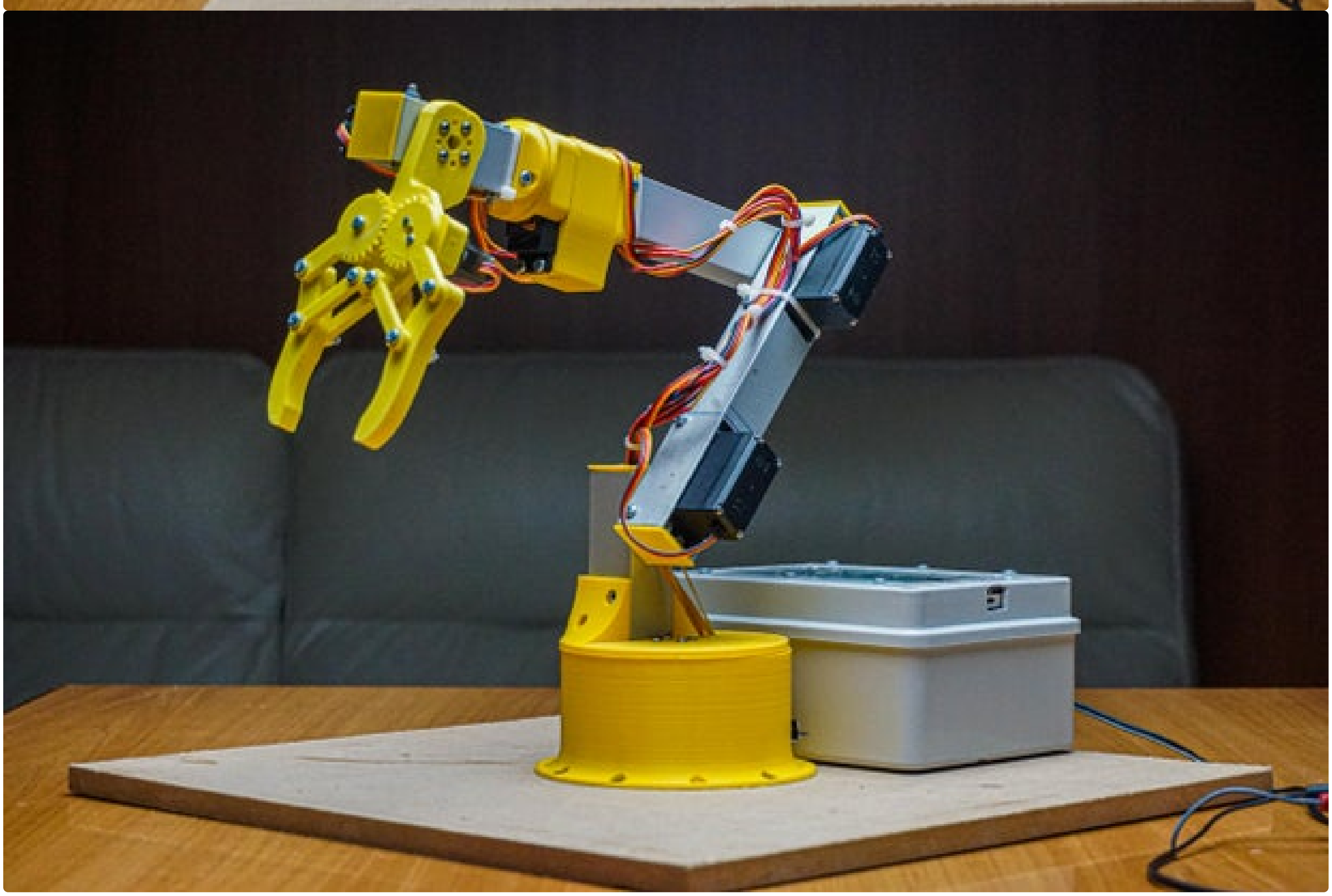
- Square aluminum tube (25mm * 25mm * 1,5mm)
- A fairly powerful power supply (2.5A * 5 motors (but you won't be using them all at the same time)
- Any Arduino board (I used Arduino UNO)
- HC05 Bluetooth module
- One SG90 Servo Motor (I got the one with metal gears)
- 5x MG996R Servomotors (more powerful than SG90)
- Wooden platform (for stability)
- Junction box (for electrical connections)

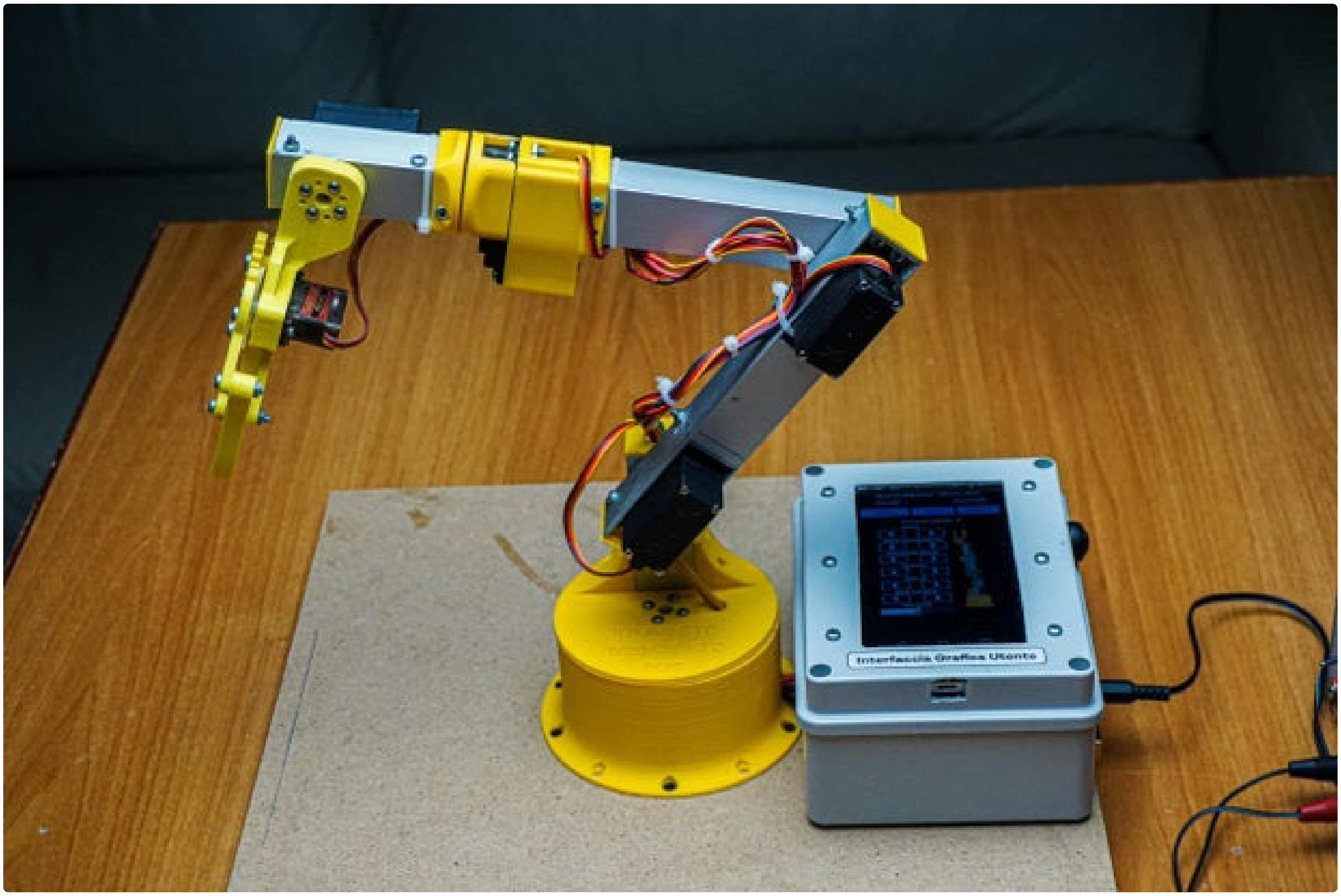

Interfaccia Grafica Utente
ROTAZIONE POLSO
GOMITO
SPALLA
BASE ROTANTE
VELOCITA'


Accensione Schermo →
USB
ALIM.
5V - 7,2V
RESET
FEEDBACK
SERIALE










https://www.youtube.com/watch?v=j-lX9l9-Di4

## Step 1: The Design and the 3D Model

Thanks to this magical Fusion 360 window, you can observe the 3D model of the robotic arm that I had to draw and design long before the entire construction phase began to avoid errors

Starting from the end, I want to retrace all the design choices made:

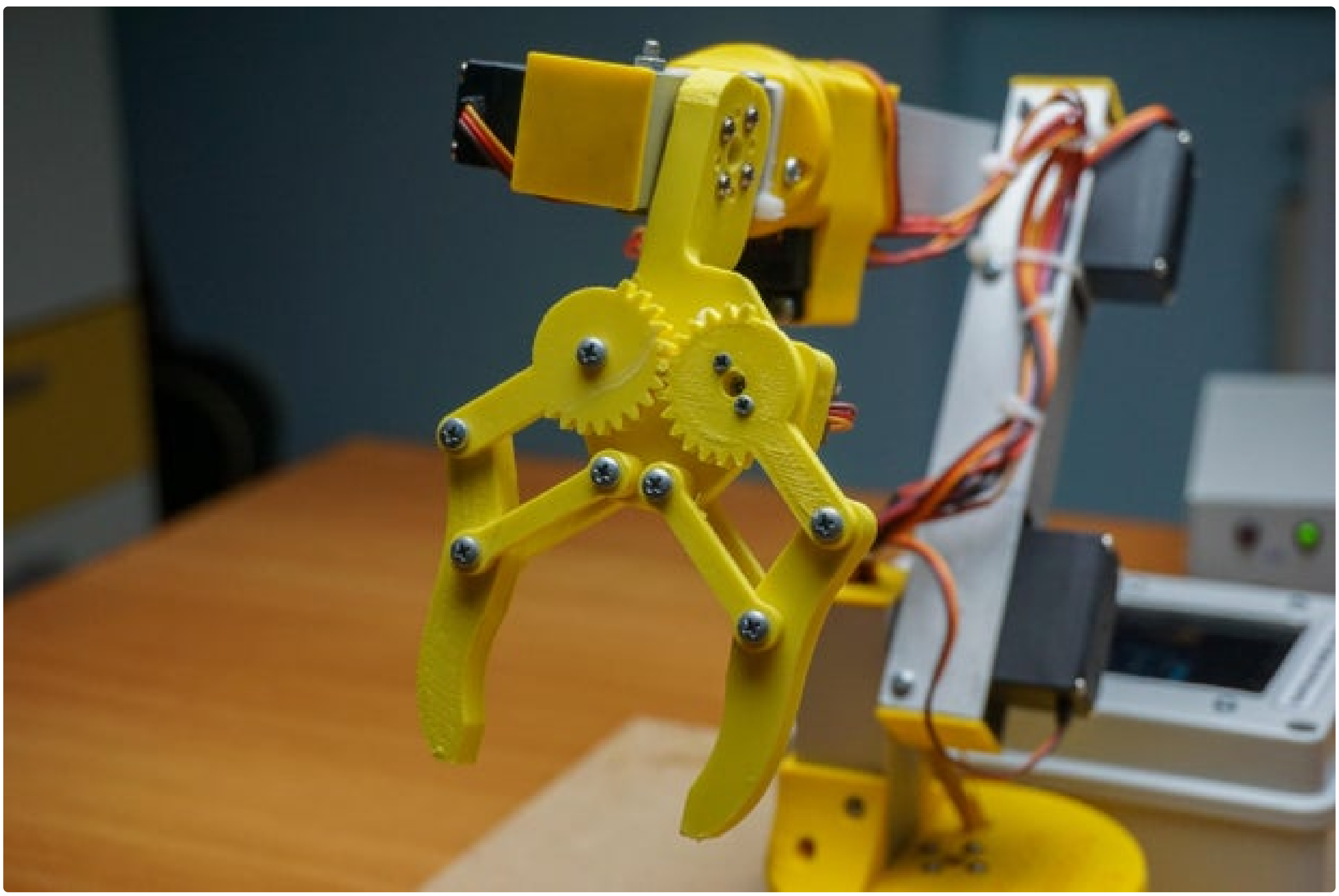
**https://www.instructables.comhttps://a360.co/3lxwtlU**

## Step 2: The Gripper

The gripper, which has been placed at the end of everything, has a not new design that I have already found in many mechanical grippers on sites such as Thingiverse (well-known platform for finding 3D models online) and that I redesigned, adapting it according to this need.

The Clapm is printed all in a few pieces and it is possible to assemble it only thanks to the 3 mm nuts and bolts. Once assembled it was enough to bring the servo to the "0 degrees" position and, keeping the clamp open, position the servo underneath, tighten all the screws and that's it.

I then screwed the gripper to the wrist with four screws to the rotating plate of the wrist angle servo motor.



## Step 3: The Wrist (angle and Rotation)

For the movement of the wrist, the arm needs two motors: one for the angle of the wrist and one for the rotation of the wrist (one that moves it up and down and one that rotates it clockwise or counterclockwise).

For the seat of the first engine, it was necessary to create a seat in the square aluminum tube to make the rear part of the engine come out. The motor (like all motors except for the base and wrist rotation) is mounted with two adapters.

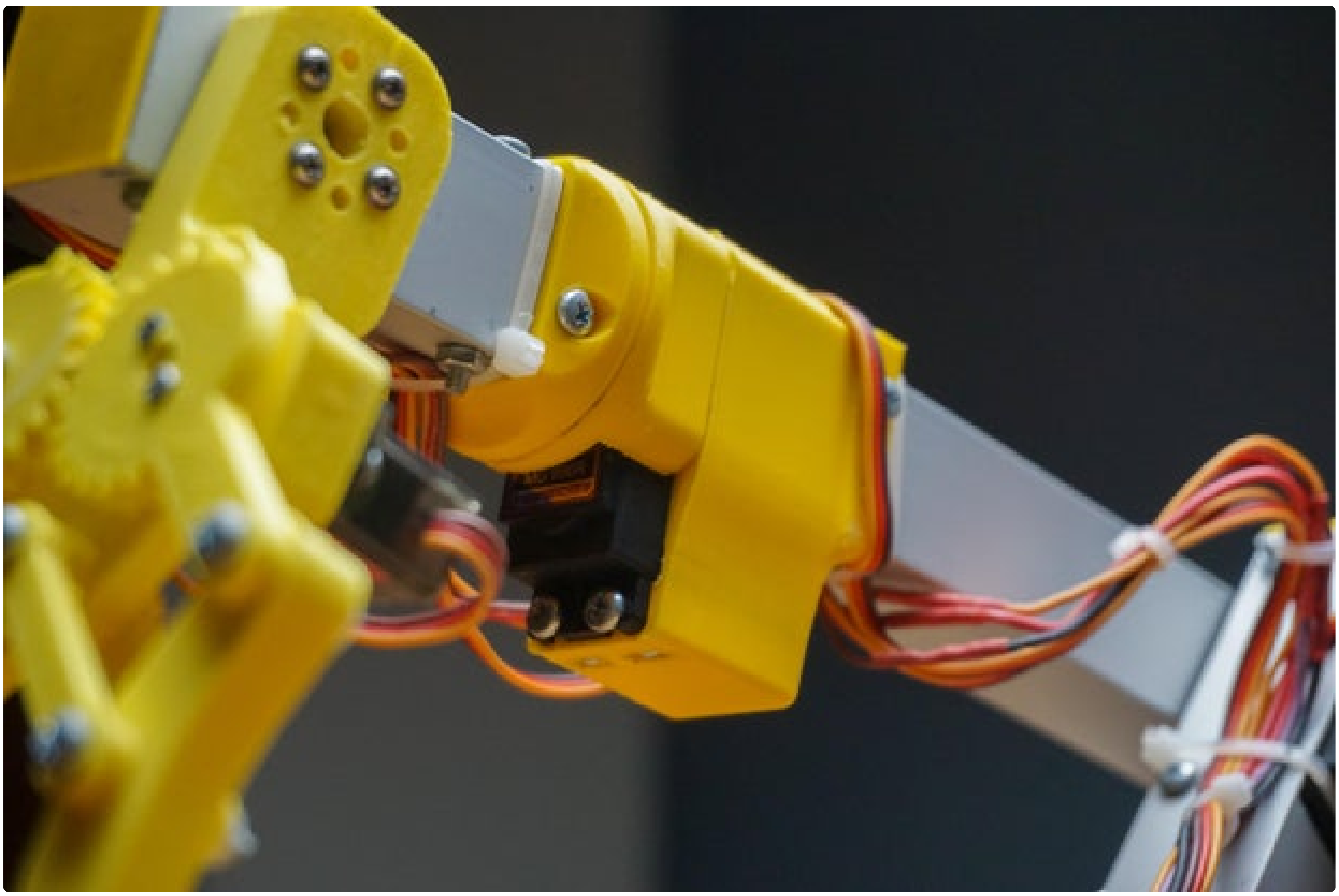
I have made these two adapters in Fusion 360 and with the 3D printer. It was enough to screw the motor to the adapters with four bolts and then mount the adapters laterally to the aluminum profiles (with a special template for the holes that I leave you in the files)

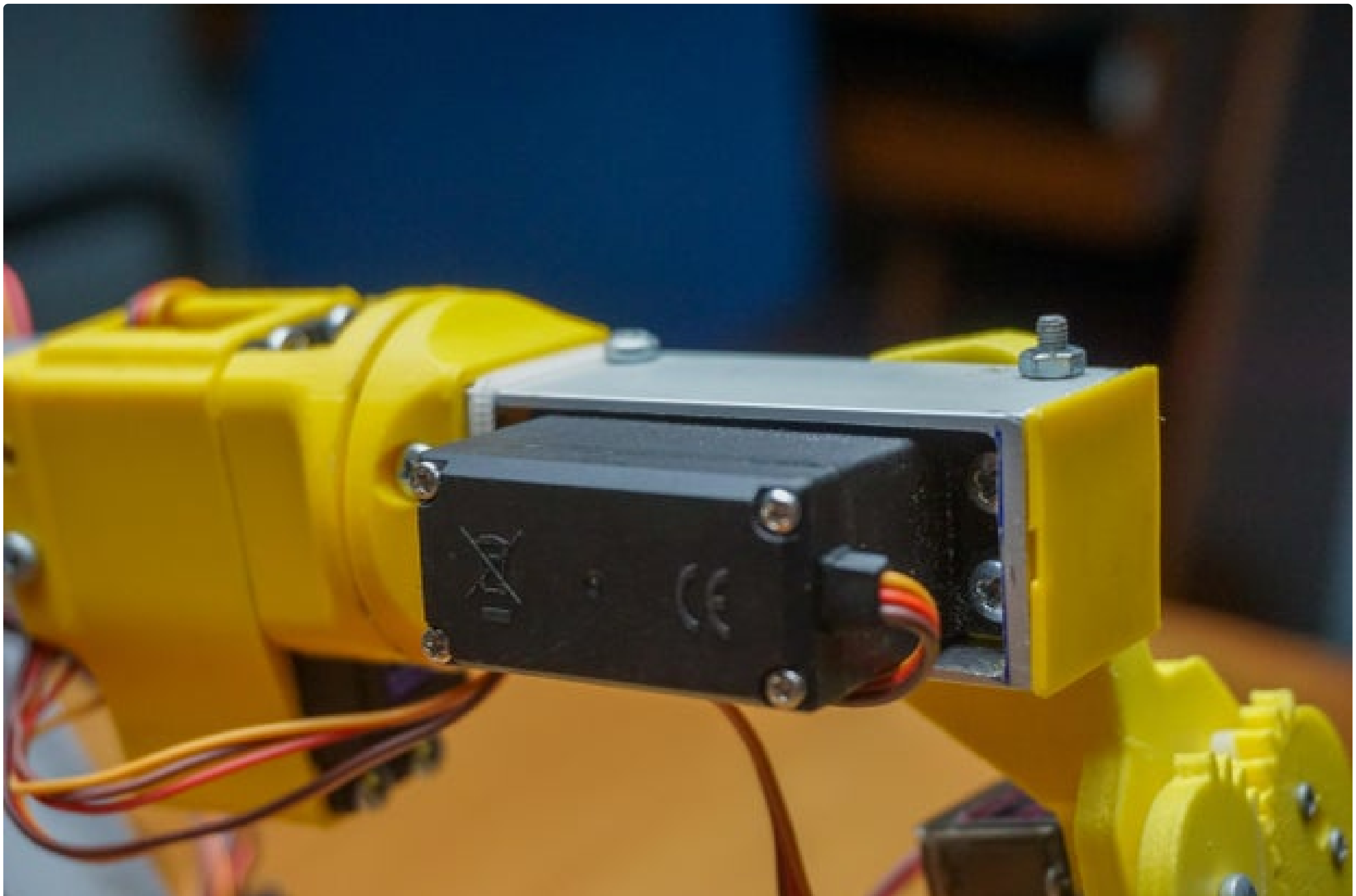
In addition to this it was necessary to drill a hole of 22mm in diameter (same internal dimension of the aluminum profile) to bring out the motor plate which I screwed to the gripper with four screws.

For the second engine (that of the rotation of the wrist) it was much more complex. In fact, it was necessary to mount the motor in such a way that it was directed towards the outside and not towards one side of the aluminum profile.

For this reason, I designed an adapter that "embraced" only two sides of the aluminum square so that the space of the two removed sides was dedicated to house the motor vertically.

The plate of the second motor (wrist rotation) is then screwed to another 3D printed adapter that embraces the aluminum profile (the one more at the end) that holds the last two motors at the end.





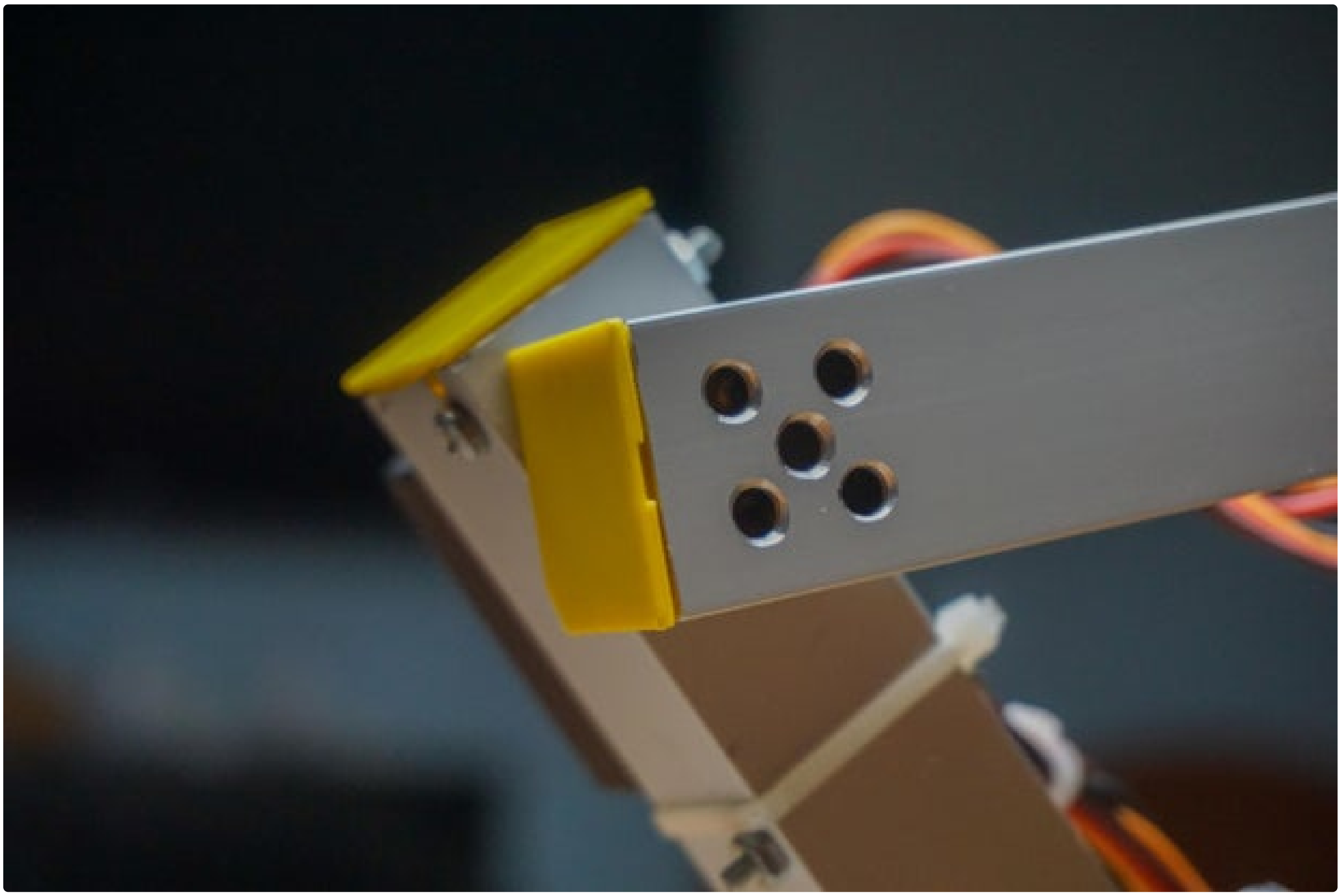
https://www.instructables.com/FW3/3L69/L5JM755C/FW33L69L5JM755C.dwg

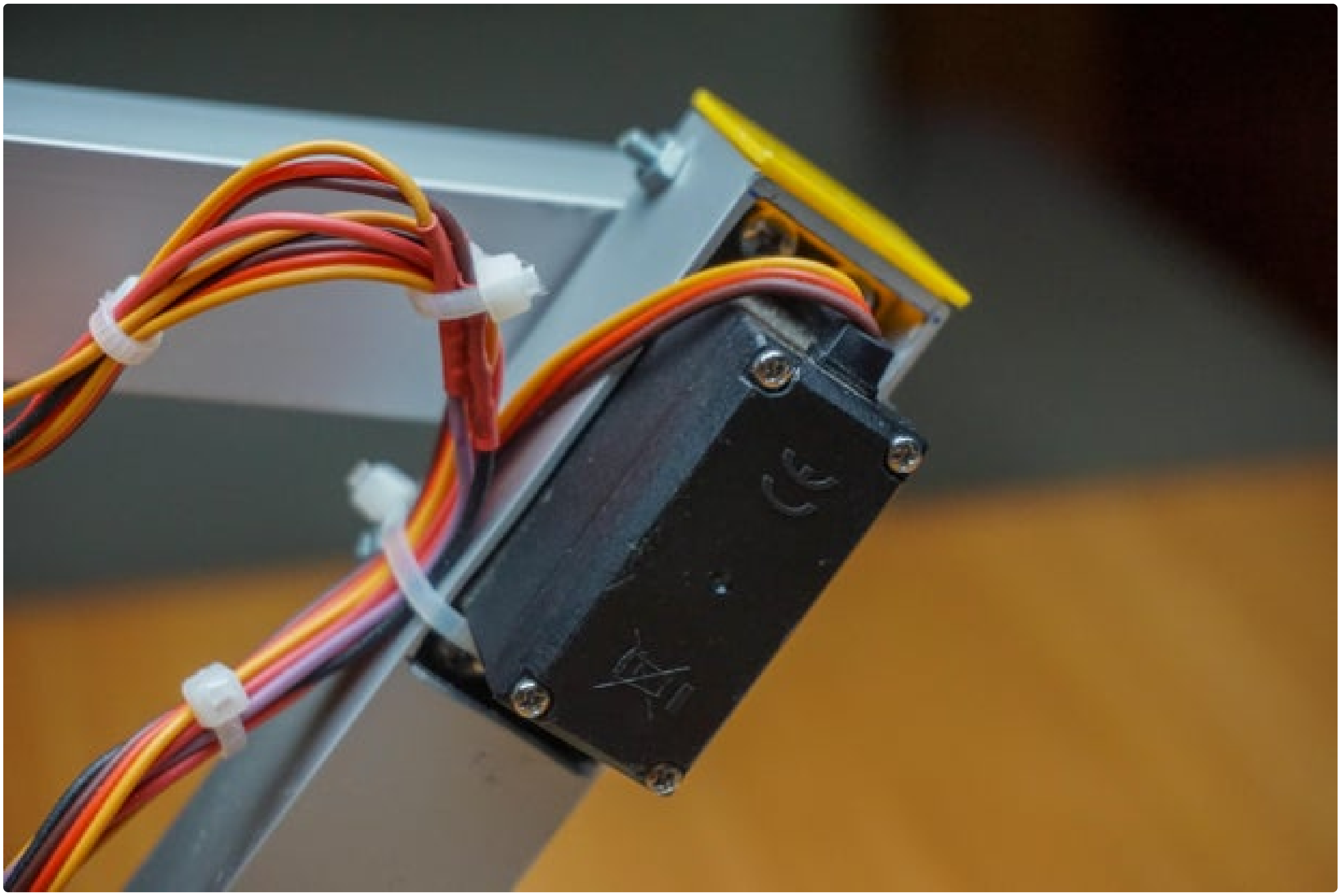
Download

## Step 4: The Elbow

To make the elbow joint I had to insert the motor in the final part of the aluminum profile (with the same technique as for the wrist angle) and I had to connect the rotating plate of the motor to the other aluminum profile with the four screws supplied with the motor (which are used for the assembly of the motor but are also good for the plate).

To screw the motor plate to the aluminum profile that constitutes the forearm, I made some holes in the part near the motor and larger holes in the external part (the ones you see in the photo) in order to let the screwdriver in and screw in the screws.





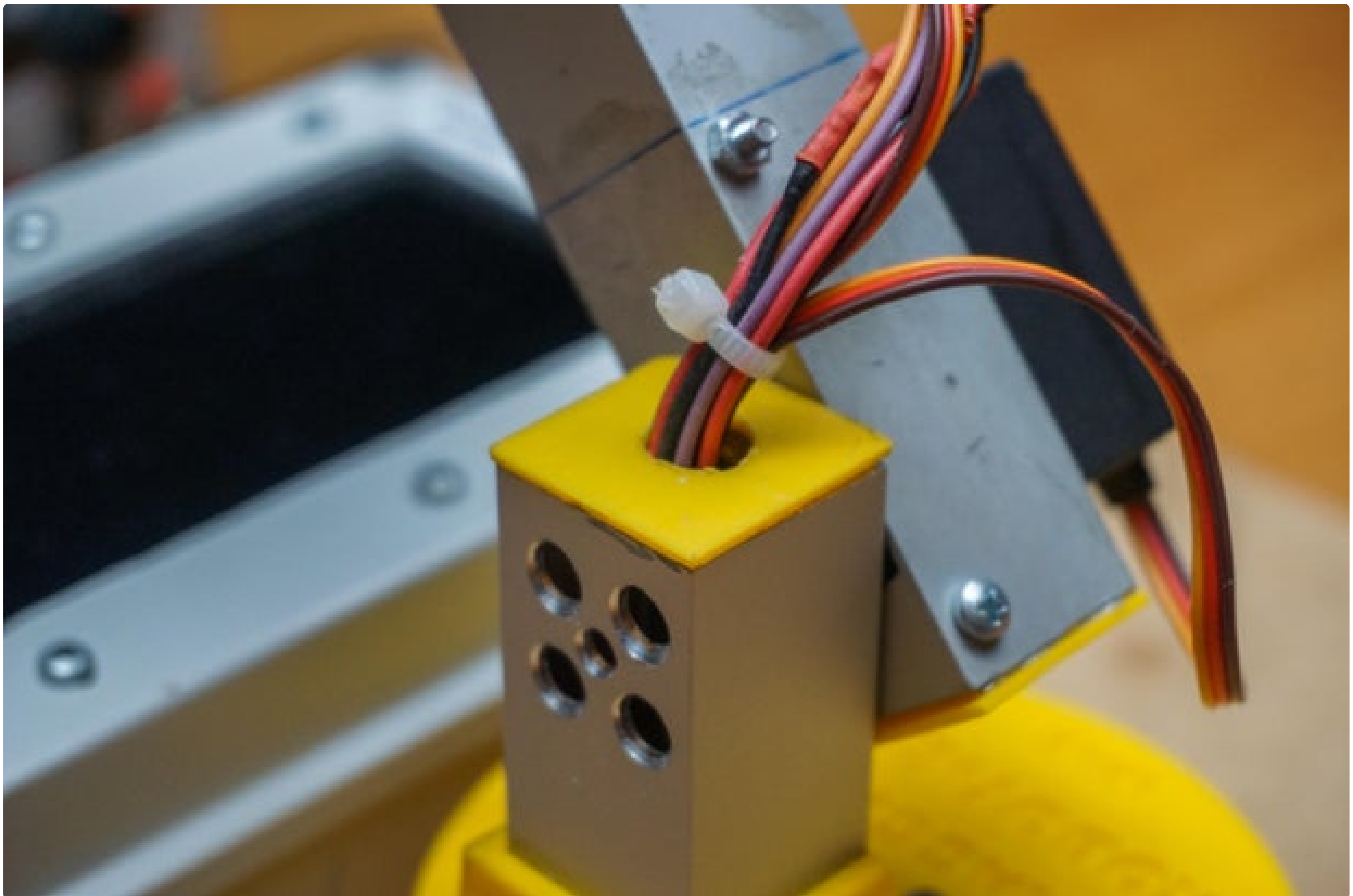
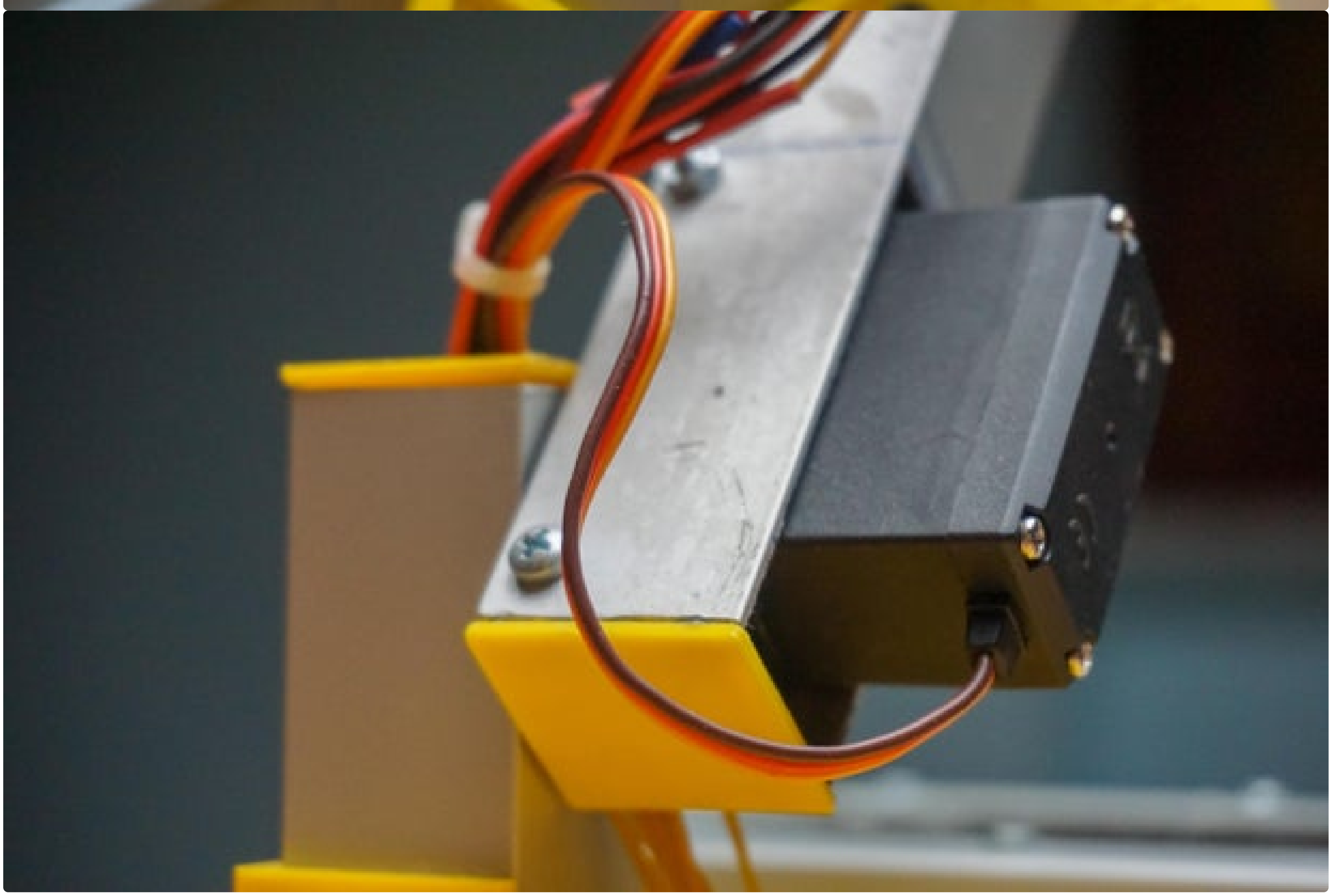
## Step 5: The Shoulder

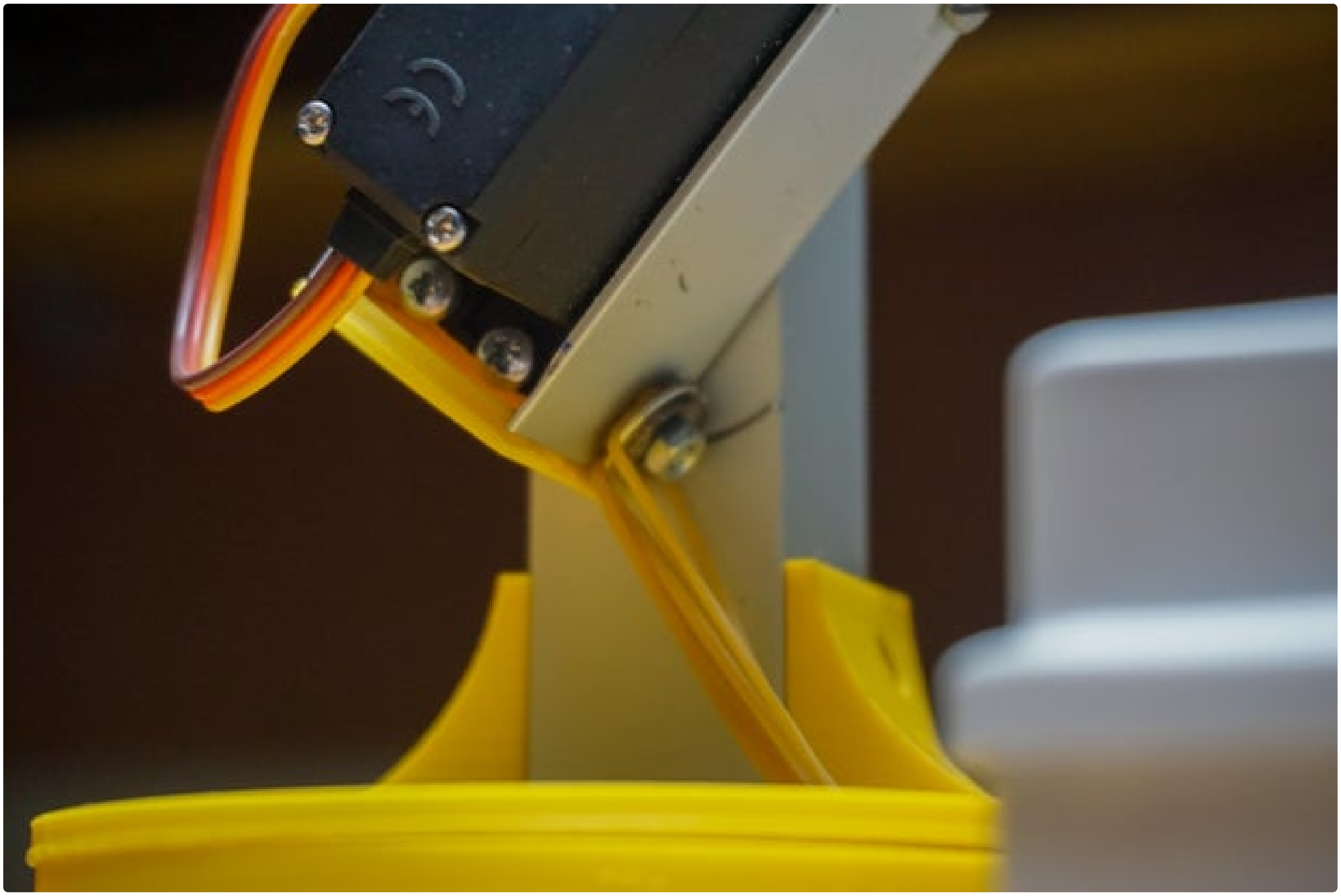
To connect the shoulder to the rotating base I preferred to insert the motor in the arm (not forearm) rather than in the base but it is just a choice, in practice nothing changes (not even in terms of balance and counterweight).

To mount it, same technique: I make a cut in the back of the aluminum to let the end of the motor come out, I make the exit hole of the motor plate and then I mount the motor with the two 3D printed adapters.

The forearm is mounted to the shoulder which is connected to the rotating base with another small piece of aluminum profile which is embraced on three sides by the rotating base so that it can remain stationary.

As you can see from the photos, the motor of the forearm (shoulder) is aided with a series of springs that exert a counter force since, without these, the motor would have to lift the whole arm by itself making a giant effort as well as useless.



## Step 6: The Rotating Base

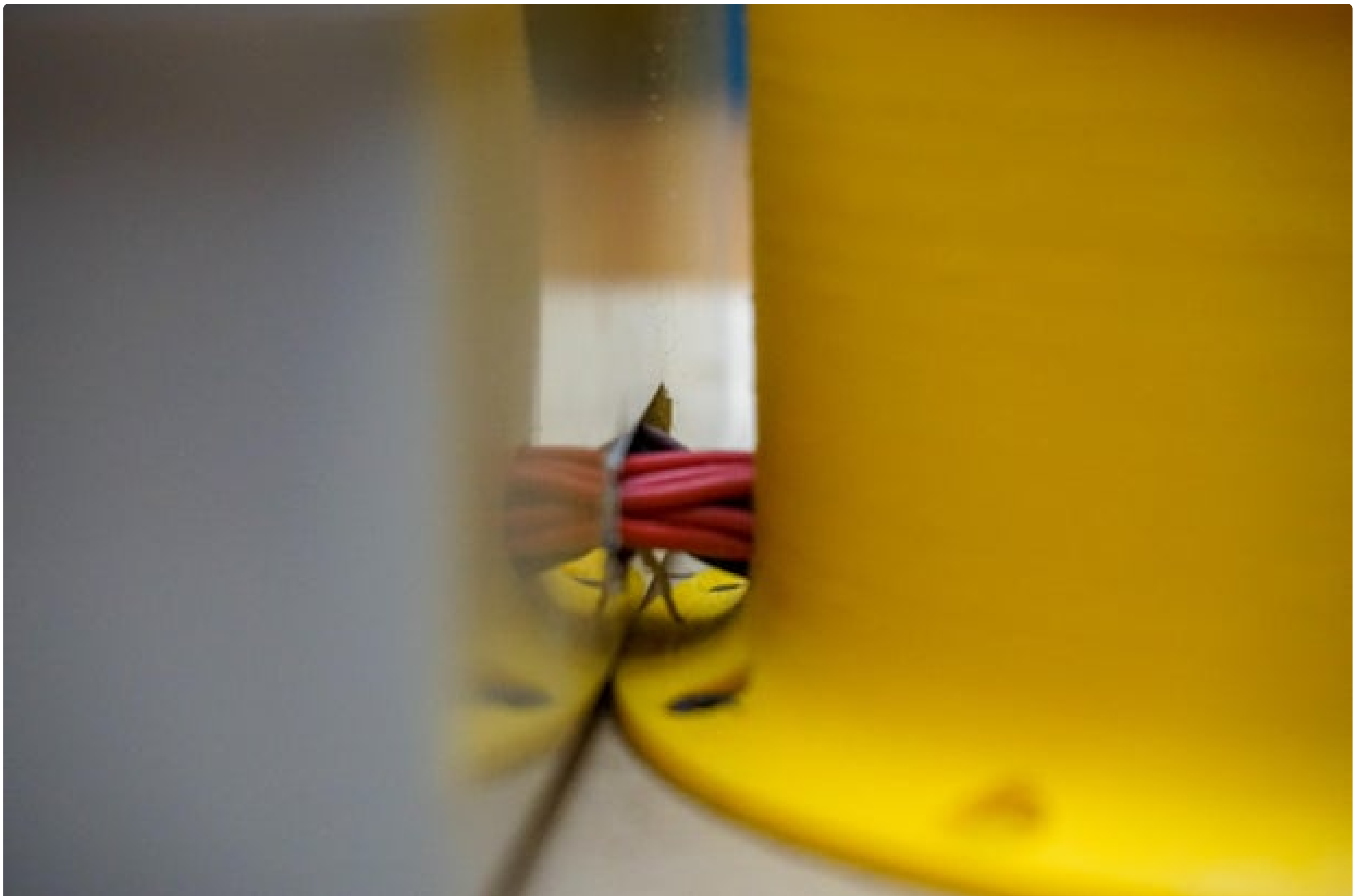
The last motor to mount is the one that controls the rotation of the entire mechanical arm on the vertical axis.

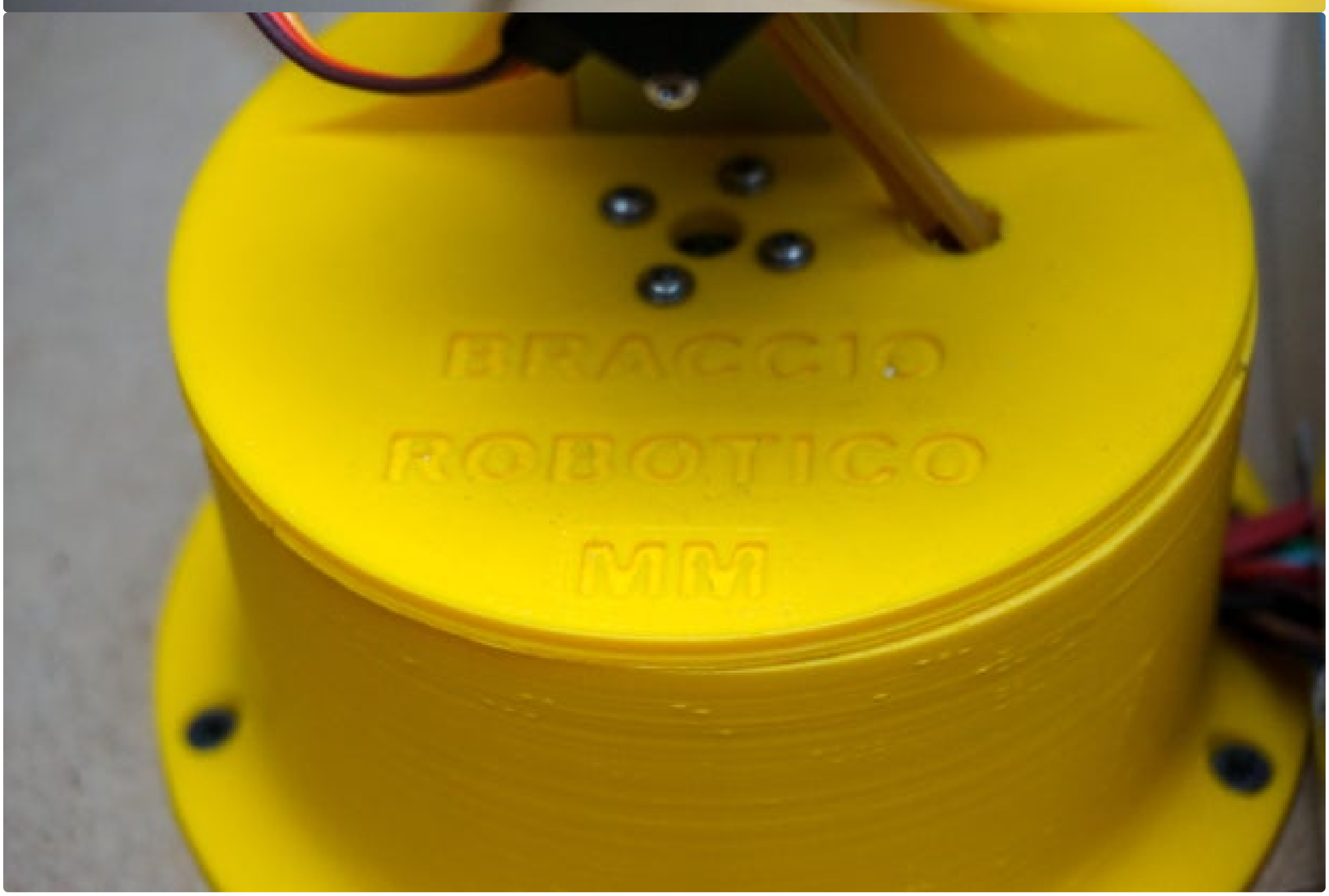
It was enough to insert the motor in the appropriate seat designed and then 3D printed (fixed base) and then I had to screw the rotating plate of the motor to the rotating base (which supports the whole arm).

Regarding the fixed and rotating base, I wanted the rotating base to lean against the fixed one on a thickness of four millimeters, leaving a thread of air to avoid friction.

This design choice allows the rotating base to rest on the four millimeters thick of the fixed base when the weight inclines in that direction. Thanks to the little air left between the rotating base and the fixed base, the motor plate does not twist.

In the fixed base there is a hole that passes all the wires to the junction box where all the electronics are housed.



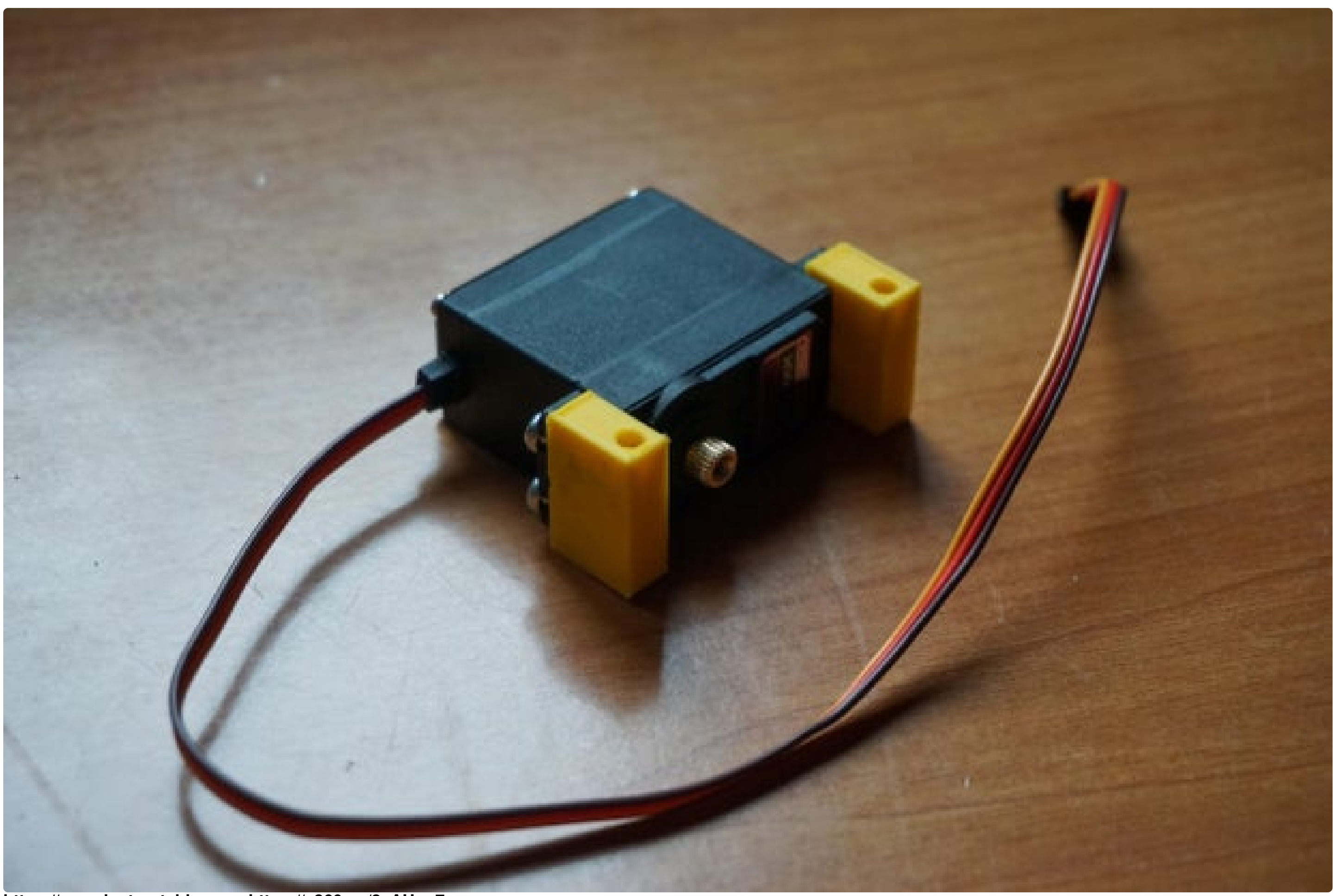

BRACCIO
ROBOTICO
MM

## Step 7: 3D Printing of the Necessary Components

The 3D components to be printed are:

- Fixed Base x1
- Wrist Rotation Adapter x1
- Rotating Base x1
- Wrist Rotation Motor Adapter x1
- Motor Side Wrist Rotation Adapter x1
- Gripper pt1 x1
- Motors Adapter x6
- Gripper pt2 x1
- Gripper pt3 x1
- Gripper pt4 x4
- Gripper pt5 x2
- Cup with hole x1
- Cup x4

The number at the end indicates the number of copies to be made for the respective print



**https://www.instructables.comhttps://a360.co/3yAHuo7**

View in 3D Download
https://www.instructables.com/FOR/5SSM/L5JM77IT/FOR5SSML5JM77IT.stl

View in 3D Download
https://www.instructables.com/FSF/TI54/L5JM77IU/FSFTI54L5JM77IU.stl

View in 3D Download
https://www.instructables.com/F9F/UNFO/L5JM77IV/F9FUNFOL5JM77IV.stl

View in 3D Download
https://www.instructables.com/FMP/ATZG/L5JM77IW/FMPATZGL5JM77IW.stl

View in 3D Download
https://www.instructables.com/FIN/NP8G/L5JM77IX/FINNP8GL5JM77IX.stl

View in 3D Download
https://www.instructables.com/FQM/JLU6/L5JM77IY/FQMJLU6L5JM77IY.stl

View in 3D Download
https://www.instructables.com/FGD/DAEI/L5JM77IZ/FGDDAEIL5JM77IZ.stl

View in 3D Download
https://www.instructables.com/FK2/PZG1/L5JM77J0/FK2PZG1L5JM77J0.stl

View in 3D Download
https://www.instructables.com/FEL/1F7N/L5JM77J1/FEL1F7NL5JM77J1.stl

View in 3D Download
https://www.instructables.com/FG9/Z27I/L5JM77J2/FG9Z27IL5JM77J2.stl

View in 3D Download
https://www.instructables.com/FWP/5CA2/L5JM77J3/FWP5CA2L5JM77J3.stl

View in 3D Download
https://www.instructables.com/F4L/9652/L5JM77J4/F4L9652L5JM77J4.stl

View in 3D Download
https://www.instructables.com/F5C/Q7QY/L5JM77J5/F5CQ7QYL5JM77J5.stl

## Step 8: Building the Arm

Regarding the construction of the robotic arm itself, I don't have much to add to what was said before but let's make a brief summary:

I recommend starting from the end with the assembly of the gripper, which is quite intuitive and simple. Once the caliper is mounted, you will need to mount the SG90 servo in the appropriate space and check that everything moves without friction.

After doing this you will have to build the wrist by cutting a small piece of aluminum (a little longer than the width of the servo) and, after having made the notch for the rear part of the motor and the hole for the rotating ring of the motor, you can mount the wrist angle motor with the two adapters I have attached above.

Once the wrist is composed, you can proceed to screw the gripper to the motor that controls the angle of the wrist.

Once this is done you will have to mount the servo for the rotation of the wrist which has nothing complicated apart from the fact that you have to carve two sides of the aluminum square for a few centimeters in order to leave only two aluminum blades that will support the adapter you need to mount the motor with the ring nut facing the gripper (outside and not to the side)

Between the wrist (as a piece in itself) and the wrist rotation motor I wanted to add another small piece that visually makes the transition from square profile to round profile but it is only a question of aesthetics (so it is optional, if you want you cannot print it)

Finally, to connect the wrist piece to the wrist rotation motor you have to print one of the adapters left in the files that screws onto the wrist piece and screws, on the other side, to the rotating bezel of the motor.

To mount the other two motors, the procedure is the same as that connecting the caliper to the motor at the wrist angle, with the only difference that you will have to drill holes in the aluminum to accommodate the screws that will run through the piece of aluminum and the rotating plate of the motor.

The last piece of aluminum at the base is mounted with 5 bolts (but if you want you can also use less) to the invitation printed in 3d on the rotating base

Last but not least, the motor of the rotating base is fixed on the fixed base (3D printed) in the appropriate seat with four bolts

For convenience I have mounted the fixed base to a larger wooden board (MDF) to give stability to the structure

A junction box must be mounted immediately next to the output of the robotic arm cables to house all the electronics inside.

https://vimeo.com/729623991

## Step 9: The Electric Circuit

Once the construction of the robotic arm is finished, it is time to move on to the electrical connections that require particular attention, especially in the case of motors.

Obviously, all the positives and negatives of the motors must be connected together and only the signal must travel individually to the respective Arduino pins

In addition to this, for the serial communication between the devices that mount the graphic interface it is also necessary to connect a bluetooth module HC05 which, in addition to being a leader in this type of field, is also very simple to connect.

Why bluetooth? Since the device that mounts the graphic interface is located less than 5cm from the Arduino, the best thing to do would have been to control the Arduino (always in serial communication) with the help of the USB port. In fact, it can also be controlled with the USB port as Arduino reads the incoming serial communication and does not care if it comes from Bluetooth or from the USB (so much so that it is also controllable from a computer)

The problem was that the device that mounts the GUI (an old Android phone) has the USB port at the bottom and I was unable (even with a 90-degree micro-USB cable) to fit everything back into the junction box.

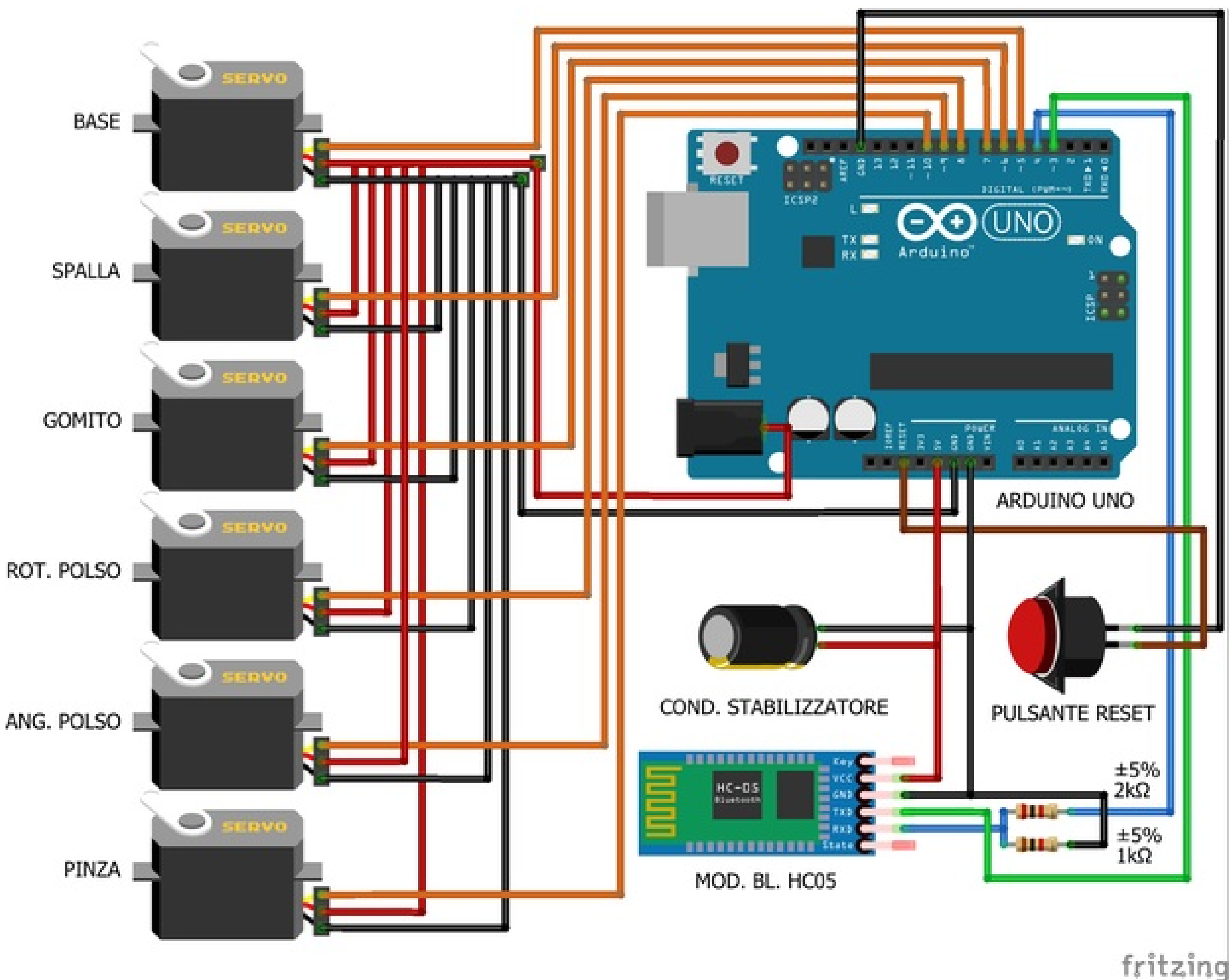
Between the 5V and the negative of the Arduino, I added a 2200µF capacitor to counteract the voltage fluctuations due to the large current draw of the motors when carrying out "against weight" maneuvers

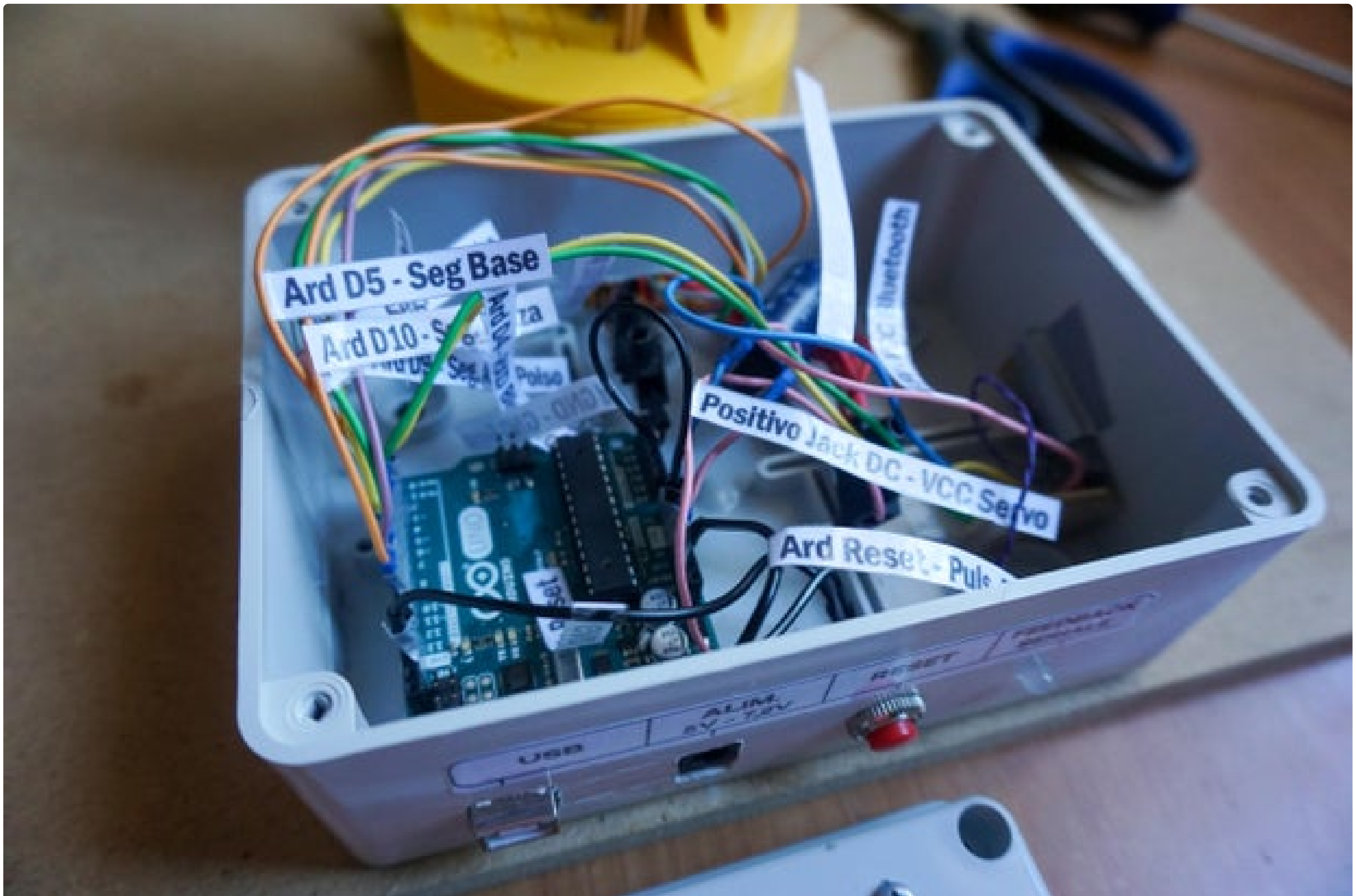
Last but not least, I've added a reset button that connects to the Arduino's GND and RESET pin

Remember that it is imperative that all negatives (GNDs) are linked together with a common negative.

Common negative that with the general positive wire (soldered to the JackDC connector) will supply power to everything from 5V to 7.2V MAX

As you can see from the photo then, I wanted to label every single wire inside the box with an inscription that would indicate what the wire is connected to both on one side and on the other so that in the future, if I had to make changes or carry out repairs) I won't have to force myself to remember.




Ard D5 - Seg Base
Positivo Jack DC - VCC Servo


Accensione Schermo →
USB
ALIM.
5V - 7,2V
RESET
FEEDBACK
SERIALE


Download

https://www.instructables.com/F7R/4GIM/L5JM789I/F7R4GIML5JM789I.fzz

## Step 10: Arduino Code

After you have connected all the wires and, I recommend, checked all the wires, it is time to write and load the code onto the board to start towards the end of the project.

The code you find in the two files (.ino and .txt) is made up of dozens and dozens of ideas from other projects that included serial communication as well as the use of arrays as well as the use of servomotors and for this I say a big thanks to the whole Arduino community that shares their projects because this is how you grow!

(I noticed that if I copy and paste the code on this page it slows down a lot so I prefer to share files)

Not being able to explain the steps in this article, I left many comments within the code to explain what the lines of code indicated are for and the video up here (which is in Italian but i added subtitles in English)


```
/*
       Braccio Robotico Arduino
       Controllato da Smartphone e app creata con "App Inventor"
       by Maurizio Miscio
*/

#include <SoftwareSerial.h>
#include <Servo.h>

Servo servo01;
Servo servo02;
Servo servo03;
Servo servo04;
Servo servo05;
Servo servo06;

SoftwareSerial Bluetooth(3, 4);  // Pin ai queli è collegato il modulo bluetooth (3,4)


int servo1Pos, servo2Pos, servo3Pos, servo4Pos, servo5Pos, servo6Pos;
int servo1PPos, servo2PPos, servo3PPos, servo4PPos, servo5PPos, servo6PPos;
int servo01SP[50], servo02SP[50], servo03SP[50], servo04SP[50], servo05SP[50], servo06SP[50];
int velocita = 20;
int IndiceArray = 0;
int DatiInEntrata;
int m = 0;

void setup() {
```

```
Lo sketch usa 7534 byte (23%) dello spazio disponibile per i programmi. Il massimo è 32256 by
Le variabili globali usano 977 byte (47%) di memoria dinamica, lasciando altri 1071 byte libe
```

https://www.youtube.com/watch?v=S9DObWZRZ9w

Download

**https://www.instructables.com/FC1/SQD1/L5JM7A3L/FC1SQD1L5JM7A3L.txt**

Download

**https://www.instructables.com/FJP/4Y9G/L5JM7A3M/FJP4Y9GL5JM7A3M.ino**

## Step 11: Graphical Interface Firmware

What hides under the screen (control panel) on the junction box is nothing more than an old smartphone I had at home (any device that mounts android and is touch is fine).

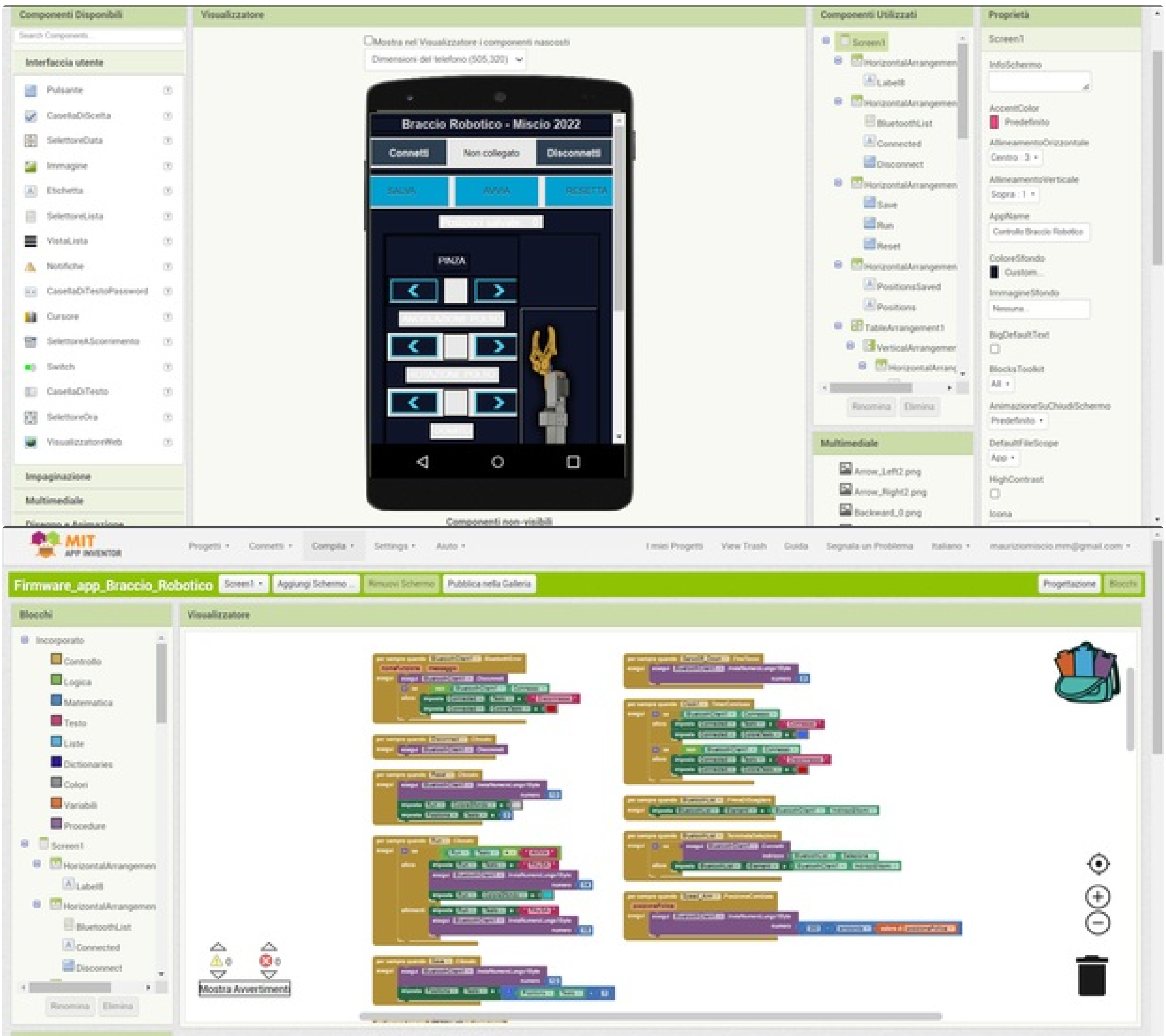
To program the graphical interface of the robotic arm I used MIT's "App Inventor" which is an online software as simple as it is functional divided into two: a part in which to build the graphical interface (GUI) and another in which, with one block diagram, actually program the app.

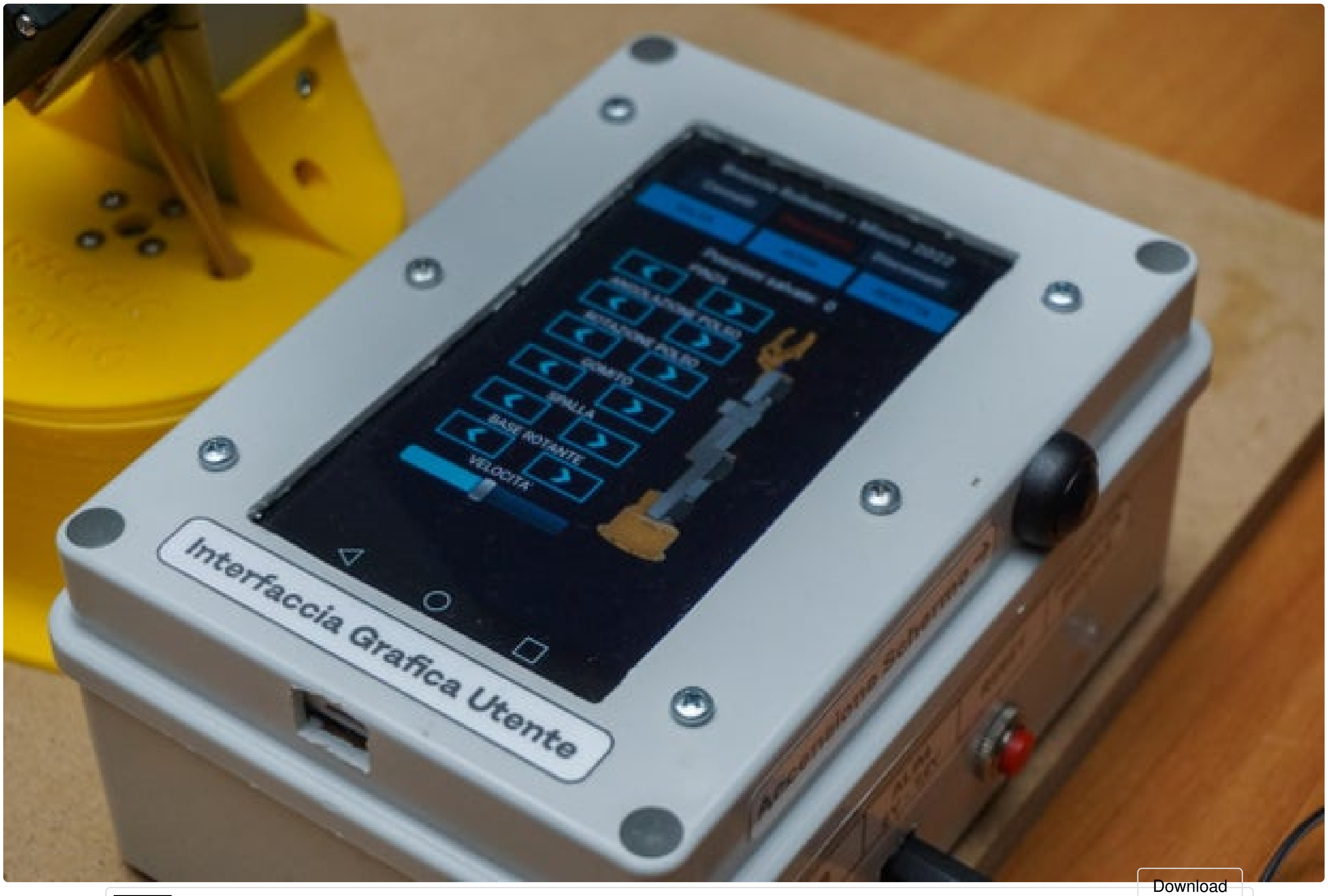
After all, the app just sends serial "codes" to the Arduino via bluetooth and then Arduino, with its code, manages everything.

Obviously, the serial codes can be customized as you wish, the important thing is that they are different from each other.

In the files you can find the Android installation file (.apk).

(if you need the App Inventor file, since Instructables does not support the .aia file type, contact me at mauriziomiscio.mm@gmail.com)






Download

https://www.instructables.com/FRJ/7GI2/L5JM7AD6/FRJ7GI2L5JM7AD6.apk

## Step 12: Finished!

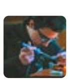
And this article ends too, thank you so much if you have come to the end and I very much hope you enjoyed finding out how this project was made ❤❤❤

## Step 13:

This article is also available here:
https://www.pcbway.com/project/shareproject/Arduino_Robotic_Arm_Controlled_by_Touch_Interface_596a5a9a.html

https://www.youtube.com/watch?v=DUrNzM2hBog

Thank you!!!
I like it


Are you kidding me? Thank YOU for reading the article

Excellent build. In pic 5 there are wires touching sharp plastic. They will eventually wear through. All wire friction/rotation points need to be well managed.

Unfortunately I know but this is the only way I have managed to manage them in order to reduce the risk

'Spiral wrap' is inexpensive and very effective. I'm harping on this because if you don't take care of it now, it will take care of you later. :) Meaning eventually something will happen. However if you Prepare for it now, and Inspect and Maintain later, you'll be fine. This was my professional job.

Hey, this afternoon I'm going to buy the spiral wrap and mount it, thanks for the advice

This is such a good instructable! I hope to follow it one day when I'm brave enough!

Thank you very much. I really appreciate your words

Hi Maurizio, I have been a lecturer for many years, some of which at the Milano Politechnic during the 70/8o's.
Looking at your photos , judging your age and reading this beautifully descriptive essay, I wish I had had students like you during my career! Congratulations from a retired scientist of 75! I look forward to reading about you in the Science Press in the near future! p.s. Can you play that guitar as good? Cheers, O

Wow, I am really very grateful to hear what you said, it gives me a lot of pleasure! Thank you so much really, it really made my day better. Anyway no, I can't play the guitar :)

very cool project, i am gonna make my big end work for school based on this. Just quick question, can you add extencions like learning to grab something and reapeat that without using your phone? example for a elderly woman it is hard to use her hands to eat so this robot would do it for her, and repeat the action like every minute, is that possibly to do that? and sorry for my bad english, love the robot and already a thank you for looking at my comment!

Wow, this is amazing! Yes, you can set actions to repeat even without using the phone but you have to set it from the program

your project is very good

Thank you very much

A youtube video showing it in operation would be nice.

Actually, I agree with you. It doesn't make my life easy to record while I'm building a project but it will certainly be done for the next projects

You don't need a video showing how it was built, although that would be awesome, but what is needed is a video showing the final product. What can it do? What does it do? How does it move? Can I play chess with it?

You can actually play chess if you want. I have a video of when it was just finished, now I am attaching it at the end. Thank you for your suggestion

I like how you attached everything together - the various brackets you designed and 3D printed to hold things together, and the innovative way to mount the servos "inside" the square tubing.

I also like the dual wrist motion, giving a full 5-degees of freedom for the arm; most DIY arms seem to neglect such a thing.

A couple of ideas to think about:

If you want to look at an interesting way to approach such movement (roll and bend) for a "wrist" - look into how some old 1980s "teaching arms" did it (such as the Armdroid, or the TeachMover):

What they did was use a differential gear system of 3 bevel gears, and used cables to drive the gears on each side of the wrist. Driven in opposite directions, the wrist will rotate. Driven in the same direction, the wrist will bend. With other drive scenarios, you could get the wrist to roll and bend at the same time. Usually, the gripper was also cable driven (actually, the whole arm was - or belts were used - to keep the weight of the arm as light as possible - all of the motors, which were steppers, were located in the "shoulder").

Such a "wrist" could be easily 3D printed - and if you didn't want to do the cable drive, you could just mount two servos on either side of the wrist driving the outer bevel gears of the differential.

For your base, rather than relying on a "gap" and the weight of the arm to "bend" and support it - leave a gap, and add an upper plate to the base, and add a circular "groove" to it and the plate the arm is attached to. Put some small bearing balls (copper "BB" shot would be fine) in the groove, and screw everything together - and you have a simple "lazy susan" bearing that would support the arm evenly, yet turn easily.

Wow, what excellent ideas! thanks a lot for the tip. As soon as I can I will inquire about these systems that you have told me that they seem really very interesting. Maybe who knows, I will be able to build a second robotic arm with these implementations! Thank you so much, have a nice day!

I am just curious: in the Arduino code, why did you write 'DatiInEntrata == 0' and other if statements like it? Couldn't you just simply write 'm = DatiInEntrata' to keep the code concise? By the way, your project is really cool.

actually due to an Arduino bug (I think). I tried to use serial input data directly but for some strange reason it didn't work with my arduino. I tried to search on the internet and it concerned the use of the SoftwareSerial so one of the solutions found on the internet was to "convert" the incoming data into other data

Great job! I love it! Your build video was fun to watch too. Thanks a lot for sharing:) Can't wait to see your next projects! And good luck with the contest!

Wow, thanks a lot

Now all is left for me to do is make it waterproof

Well, it might be difficult haha

Very nice of you to post this in both English and Italian! You're awesome :D

Sharing with the world what I do and what I learn and learn thanks to the sharing of others is what I want to do in life and what satisfies me most so I'm happy to do it.