

# DroneBot Workshop

Build your own Electronics, IoT, Drones and Robots – Welcome to the Workshop!



[Home](#) [Arduino](#) [Raspberry Pi](#) [ESP32](#) [Electronics](#) [Robots](#) [Forum](#) [YouTube](#)  
[About](#)

## Assembling the MeArm Robotic Arm

The MeArm is a small and inexpensive robotic arm with 4 degrees of freedom. Originally developed as an educational tool for schoolchildren the MeArm has since become a popular device for robotics experimenters of all ages. Its simple open source design makes it an ideal tool for anyone developing software and electronics. Projects created with the MeArm can then be ported for use on a more expensive and capable full-sized robotics arm.

24

## Table of Contents

[1 Open Source](#)

[2 MeArm Kit Contents](#)

[2.1 MeArm Acrylic Parts](#)

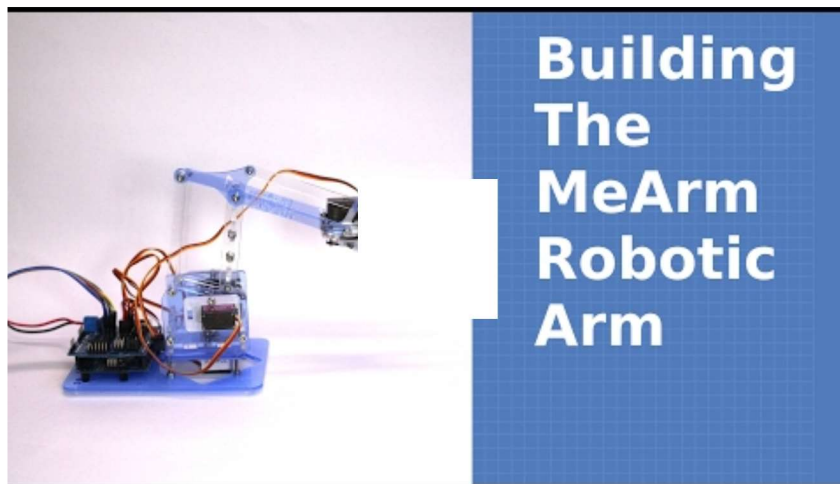
[2.2 MeArm Fixings Package Contents](#)

[2.3 MeArm Servos](#)

[3 MeArm Servo Calibration](#)

[3.1 MeArm Servo Calibration 1 – Arduino Calibrate](#)

[3.2 MeArm Servo Calibration 2](#)



The MeArm is a great introduction to robotics, as well as a useful tool for seasoned experimenters

## Open Source

As the MeArm is an open source project the plans for building it are available freely on the Internet. To build a MeArm from scratch requires just four small servo motors and a material such as acrylic or wood to construct the arm itself. It is also possible to assemble a MeArm using parts produced with a 3D Printer.

For those of us who don't have the facilities to cut or produce all of the MeArm components there is another option. The MeArm kit, available from Mime in the UK, contains all of the parts required to build a MeArm. This is the route that I have chosen to create my own MeArm and this is what I have documented in this article and its associated video.

There are over 50 steps that you need to take to assemble a MeArm and I'm going to go over all of them in this article, so as a result this is a very long document (and a very long video). For the most part the assembly went smoothly, however I did encounter a few errors in the MeArm manual that I needed to work around. This included one major omission that actually resulted in my needing to take apart and reassemble my MeArm! If you follow this guide along with the MeArm manual you'll be assured of successfully completing the build without having to resort to such drastic measures.

### – Servo Arms

#### 4 Building the MeArm

- 4.1 Assembly – Steps 1, 2, 3
- 4.2 Assembly – Steps 5, 6, 7, 8 (forgot 4 – put feet on!)
- 4.3 Assembly – Steps 10, 11 (MeArm has no step 9!)
- 4.4 Assembly – Step 12
- 4.5 Assembly – Steps 13, 14
- 4.6 Assembly – Step 15
- 4.7 Assembly – Steps 16, 17
- 4.8 Assembly – Step 18
- 4.9 Assembly – Steps 19, 20, 21 (22 not shown, with Arduino)
- 4.10 Assembly – Step 23
- 4.11 Assembly – Steps 24, 25
- 4.12 Assembly – Step 26
- 4.13 Assembly – Step 27
- 4.14 Assembly – Step 28
- 4.15 Assembly – Step 29
- 4.16 Assembly – Step 30
- 4.17 Assembly – Step 31
- 4.18 Assembly – Step 32
- 4.19 Assembly – Step 33 (34 not shown, with Arduino)
- 4.20 Assembly – Step 35
- 4.21 Step 36 & 37 SNAFU! – Parts do not exist!
- 4.22 Assembly – Step 38, 39 & 40



Despite these small setbacks I found the MeArm to be a quality product that will no doubt be useful for several future robotics experiments. Now that it's built I have moved on to designing and constructing an advanced controller for the arm which I will share with you in the next article in the MeArm series.

So grab yourself a MeArm kit and let's get building!

## MeArm Kit Contents

The MeArm Kit is available in different colors with two "flavors" – with or without a controller (which Mime calls a "brains board"). I chose to purchase the kit without a controller as I intended to build my own controller. This is a popular option and the Mime website has instructions for using both an Arduino and a Raspberry Pi with the MeArm. For my initial build I chose to use an Arduino. Naturally I chose a blue MeArm to match my workshop!



When I first received the MeArm Kit I was surprised as to how small the box was, it didn't look like you could actually fit all of the parts into such a small container. However this is just a testament to the efficiency of the MeArm design and upon opening the tiny box I was relieved to discover that everything was indeed included.

[4.23 Assembly – Step 41, 42, 43 & 44](#)

[4.24 Assembly – Step 45](#)

[4.25 Assembly – Step 46](#)

[4.26 Assembly – Step 47](#)

[4.27 Assembly – Step 48](#)

[4.28 Assembly – Step 49, 50 & 51](#)

[4.29 Missing Feet \(from step 4\)](#)

[4.30 Assembly – Step 52](#)

---

## [5 Finishing the MeArm](#)

[5.1 MeArm Conclusion](#)

---

## [6 Resources](#)

---

## Welcome to the Newsletter!

Let's keep in touch!

Subscribe to the DroneBot Workshop Newsletter and be the first to find out about new projects and new features on the website.

No spam - just useful information and updates sent to you every second week. I'd love to be a regular visitor to your Inbox!

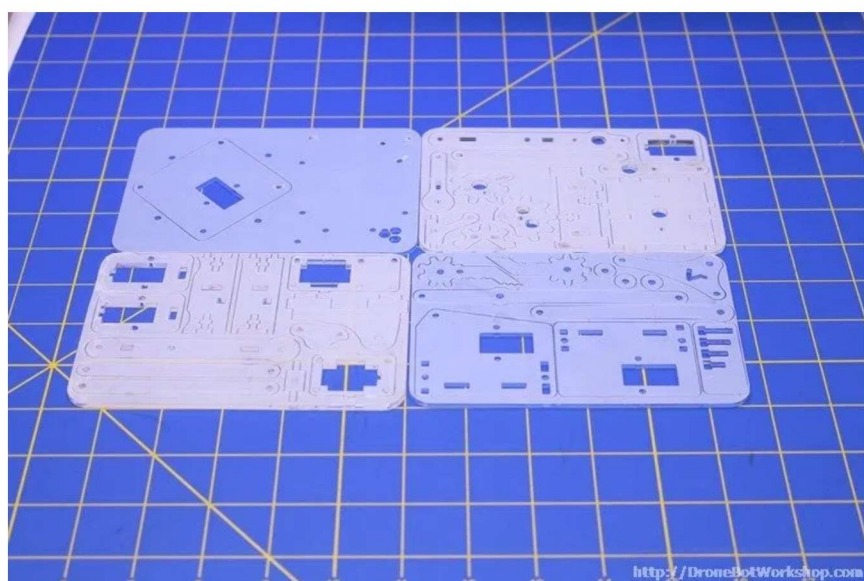
[Subscribe Today!](#)



No instruction manual is packaged with the MeArm Kit, instead you can download a PDF document from the Mime website. I downloaded and printed the MeArm Assembly Manual Arduino Edition, my copy was Revision 1.0 September 2015. As I purchased my kit in December 2016 there are a few changes to the design that were not reflected in the manual, this is to be expected and with the exception of one change did not cause a lot of confusion.

## MeArm Acrylic Parts

The MeArm is constructed of acrylic plastic and within the kit you'll find four sheets, each precut with all of the MeArm parts. This is actually where I made my major error and rather than make you read through to the end of the article I'll tell you what that was right now.



The four sheets each come with a white plastic backing on one side. What I didn't realize was that you NEED TO REMOVE THE PLASTIC before beginning construction! Unfortunately I didn't do this and as a result my initial build didn't work very well. Once I rebuilt the MeArm without the plastic backing on the components everything worked a lot better! In retrospect I should have realized this when I started, in my own meager defense the manual could of perhaps mentioned this step but it didn't. So don't make the same error

## Recent Articles

[Using GPS Modules with Arduino & Raspberry Pi](#)

[Arduino Nano RP2040 Connect – Arduino meets Raspberry Pi](#)

[Arduino Nano 33 IoT Board](#)

[Build an ESP32CAM Robot Car](#)

[Learning Python – A List of Free Resources](#)

## Popular Articles



**Stepper Motors with Arduino – Getting Started with Stepper Motors**

February 10, 2018

By [DroneBot Workshop](#) | 80 Comments

Stepper motors are used in a variety of devices ranging from 3D printers and CNC machines to Blu Ray drives,...



**Controlling DC Motors with the L298N Dual H-Bridge and an Arduino**

March 11, 2017

By [DroneBot Workshop](#) |



I did – take the plastic off before you begin construction.

The photos (and video) of the MeArm assembly I've included here show the parts with the plastic still attached. While this is incorrect the rest of the assembly procedure is still valid. As an added bonus the plastic does make it easier to see the pieces in the photos and the video, so it wasn't a complete failure!

## MeArm Fixings Package Contents

The “Fixings Package” contains all of the nuts and screws that you'll need to assemble the MeArm. The screws are all M3 metric fasteners and in many cases they self-tap into the acrylic. The parts list is as follows:

- 10 x M3 Nut
- 6 x M3-6mm Screws
- 15 x M3-8mm Screws
- 3 x M3-10mm Screws
- 8 x M3-12mm Screws
- 4 x M3-20mm Screws

In addition the Fixings Package also contains four stick-on feet for the MeArm base (which ironically I neglected to install in Step 4, I corrected that near the end of the build) and a 2.5 mm hex Allen key. The Allen key is required to tighten the screws, in my case I used a 2.5 mm hex screwdriver from my toolbox as I found this a bit easier to work with.



145 Comments

Controlling DC Motors is an essential skill for constructing robots and other hobby projects. An easy way to control DC...



### Getting started with the ESP32-CAM

May 24, 2020

By [DroneBot Workshop](#) | 86 Comments

Today we will look at the amazing ESP32-CAM module from A-Thinker. This 10-dollar module features a 2MP camera, microSD card...



### Getting started with ESP32

April 2, 2020

By [DroneBot Workshop](#) | 51 Comments

Time to move up to another microcontroller, the ESP32. This amazing device has multiple I/O ports, WiFi, Bluetooth and BLE,...

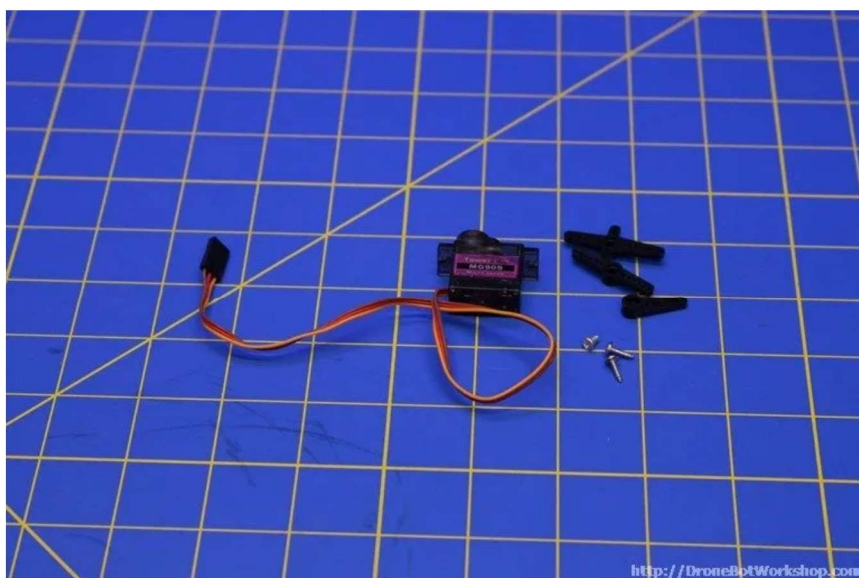


### Using Inexpensive 433MHz Transmitter

After confirming that I was provided with the correct quantities of hardware I separated the nuts and screws and placed them into a small parts box with individual compartments for each one. If you don't have a box like this you could simply use a egg carton to organize the parts.

## MeArm Servos

The kit comes with four MG90S servo motors. These popular servo motors are the metal gear equivalent of the even more popular SG90S motors which I'm sure you've probably used before. Each servo is in its own package and each one is packed with three styles of servo arm, as well as with a the screws required to use the arms and attach them to the servo shaft.



In addition to the servo motors there is also a short 3-pin extension cable provided. This is used to extend the reach of the claw servo motor cable.

## MeArm Servo Calibration

Prior to beginning construction of the MeArm you will need to calibrate the servo motors. Follow along and I'll show you how that's done. You'll need an Arduino, some jumper wires, an external 5 volt power supply capable of enough current for the four servos and a permanent marker pen.

## and Receive Modules with Arduino

February 17, 2018

By [DroneBot Workshop](#) |  
42 Comments

Those inexpensive RF transmitter and receiver modules that you can get on eBay and Amazon are perfect when you need...



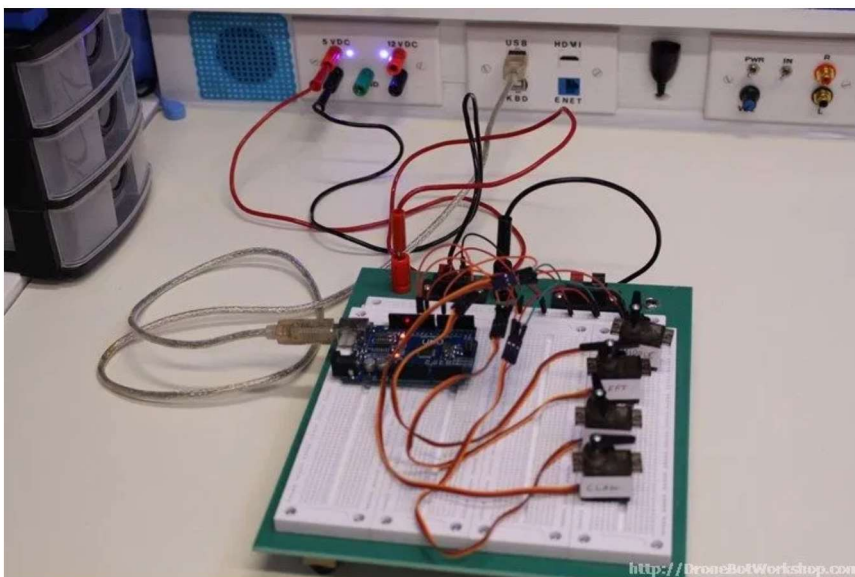


## MeArm Servo Calibration 1 – Arduino Calibrate

The MeArm manual has a diagram that shows how to hook the four servo motors up to an Arduino Uno. While the hookup diagram is indeed correct two of the colors illustrated for the servo motor wires didn't match the ones on the motors I was supplied. You'll probably run into this problem as well, so here is the list of the actual servo motor wire colors and how they "translate" to the ones shown in the MeArm manual:

- Yellow = Orange (This is the PWM control wire)
- Black = Brown (This is the Ground wire)
- Red = Red (This is the 5 Volt power supply wire)

The PWM wires for the motors are connected to four of the digital outputs on the Arduino. If you're using a different type of Arduino you can move these to other outputs, providing you select outputs that are capable of PWM. If you do that you'll need to modify the supplied Arduino sketch accordingly.



24

Speaking of the sketch, the one in the manual is missing the trailing curly bracket so you'll need to add that in or it won't compile. An alternative (and easier) method is to just copy the code from here. You can also download the sketch from <https://codebender.cc/sketch:148456>.



```
1  #include <Servo.h>
2
3  Servo middle, left, right, claw ; // creates 4 "servo objects"
4
5  void setup()
6  {
7    Serial.begin(9600);
8    middle.attach(11); // attaches the servo on pin 11 to the middle object
9    left.attach(10); // attaches the servo on pin 10 to the left object
10   right.attach(9); // attaches the servo on pin 9 to the right object
11   claw.attach(6); // attaches the servo on pin 6 to the claw object
12 }
13
14 void loop()
15 {
16   middle.write(90); // sets the servo position according to the value(degr
17   left.write(90); // does the same
18   right.write(90); // and again
19   claw.write(25); // yes you've guessed it
20   delay(300); // doesn't constantly update the servos which can fry them
21 }
```

Again make sure to use an external 5 volt supply to drive the motors, any supply that can provide at least 2 amperes will suffice. Don't try using the 5 volt output from the Arduino as its internal voltage regulator can't supply enough current.

Once you have everything hooked up power it up. You should see the motors move to a set position. Take your permanent marker pen and draw a line across the servo motor body and the motor spindle. This line serves as a calibration line in case the motors get moved during assembly. The MeArm manual illustrates this pretty well.

## MeArm Servo Calibration 2 – Servo Arms

After calibration the manual has you go through the step of attaching servo arms to the motors. As it turns out you'll be removing these arms before installing the motors, so I believe this step is just there so you can identify the motors during assembly. Nonetheless it isn't very time consuming so follow the instructions for using the small machine screw included with each servo and attach the arms as shown in the diagram.

Now that you're done you can power down the Arduino and remove the motors. We are ready to start actual construction





of our MeArm!

## Building the MeArm

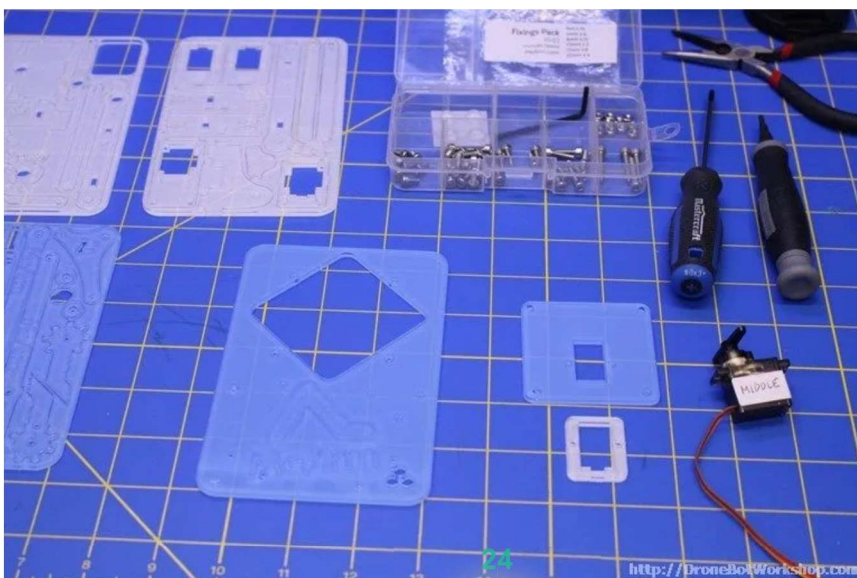
As I mentioned at the beginning this is a very long article, as the manual illustrates over 50 steps to complete construction.

The step numbers in the following text correspond to the ones in the manual and when I refer to the “diagram” I’m talking about the diagram within the MeArm manual, so grab your manual and let’s get started.

And remember – **PEEL OFF THE PLASTIC** from the acrylic parts!

### Assembly – Steps 1, 2, 3

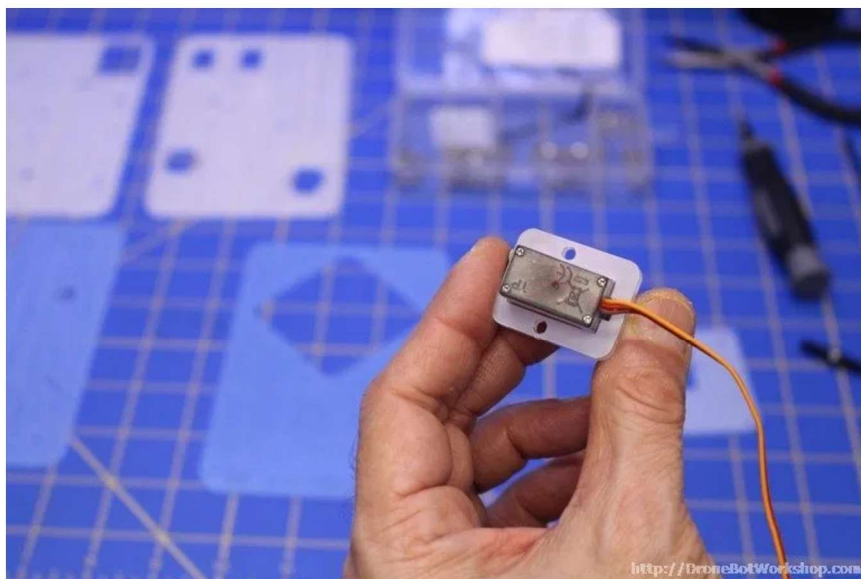
We’ll begin construction by assembling the “Pivot Servo Plate”, this is the base plate that holds the base motor for the MeArm. You’ll be inserting the base motor into its servo collar, an arrangement that is used to hold all of the servo motors in place. No nuts are used as the screws all self-tap into the acrylic material.



One important thing to note is the orientation of the motor in the collar. The collar has a notch in one side, this is the side of the servo that the wires enter into. You will also want to be sure that you get the motor and collar assembly facing the



right direction. Just remember that the shaft of the motor should be centered in the Pivot Servo Plate and you'll be fine.



## Assembly – Steps 5, 6, 7, 8 (forgot 4 – put feet on!)

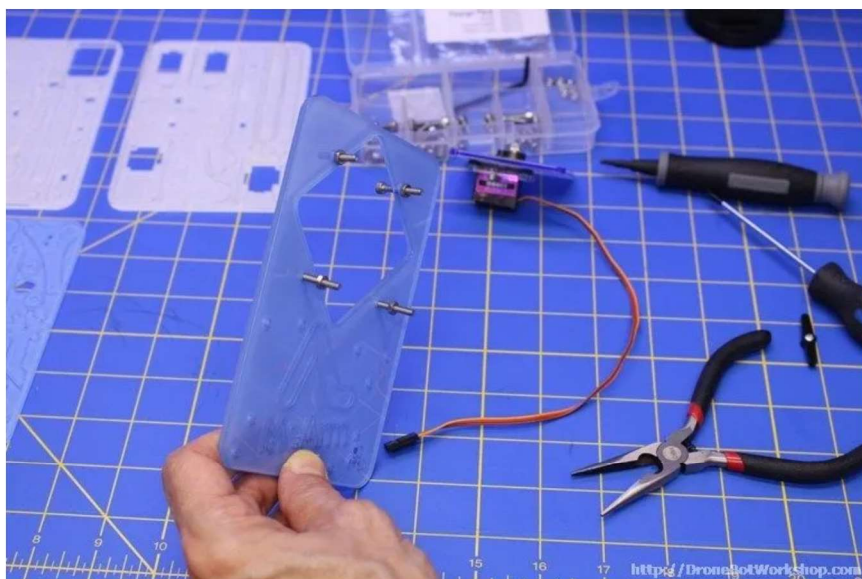
Quick note, I missed step 4 which is to place the four “feet” onto the base of the MeArm. They just peel and stick, so do that now. Make sure not to obscure the mounting holes for the Arduino Uno that have been drilled into the base plate.

Steps 5 through 8 are to mount the Pivot Servo Plate onto the base. This is accomplished with the four 20mm screws provided in the Fixings Package (they are the longest screws in the package). You'll also use four of the M3 nuts to secure the assembly to the base.

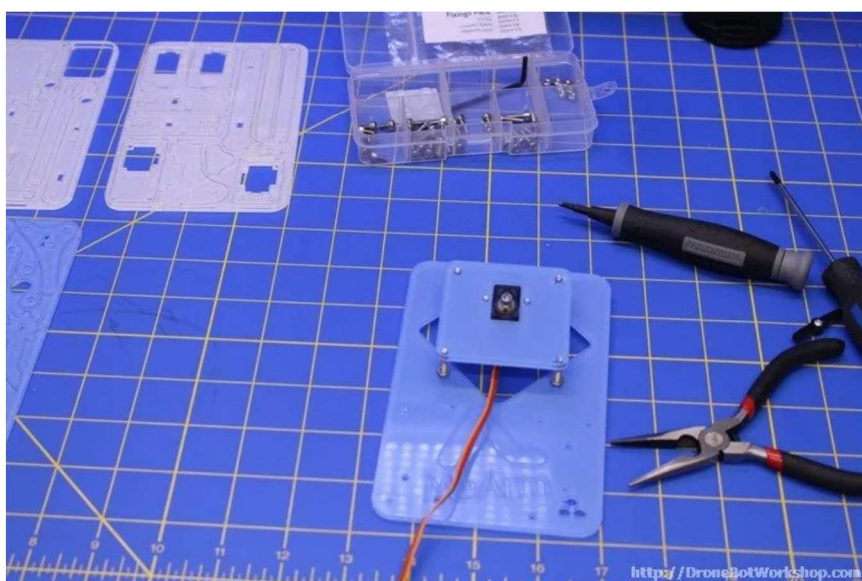
24







You'll want to be sure to get the tops of the screws flush with the material in the Pivot Servo Plate. Once you've done that you can finger-tighten the nuts flush with the base to complete these steps. Later, once the arm is finished, you can use pliers or a metric wrench to tighten them more securely.

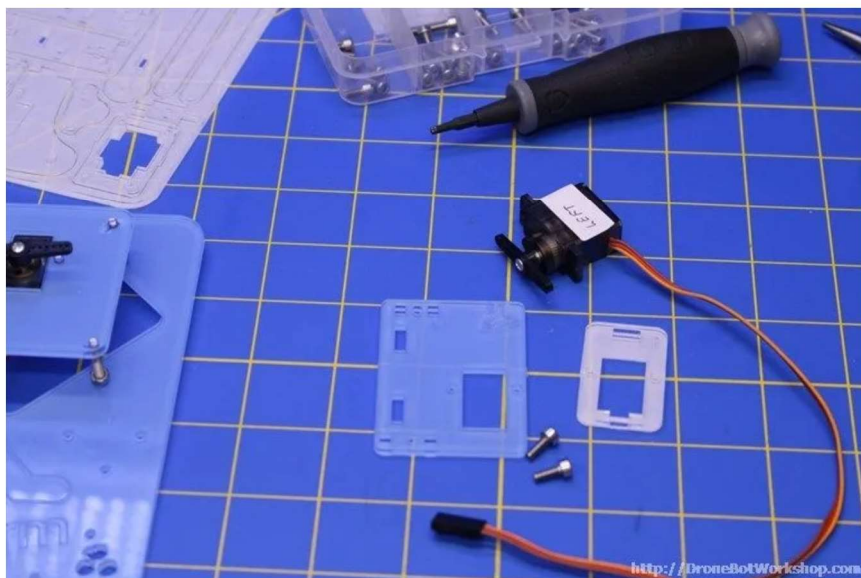


## Assembly – Steps 10, 11 (MeArm has no step 9!)

No I didn't omit a step again, the manual doesn't seem to have a step 9!



In these steps you'll be mounting another servo motor using a servo collar, in this case it's the Left servo which you will attach to the Left Arm Servo Plate. As with the previous motor pay attention to the orientation of the collar in respect to the motor.



When you tighten the 8mm screws used to secure the motor don't overdo it. They need to be tight enough to firmly hold the motor but not too tight or the acrylic will warp and possible crack.

## Assembly – Step 12

Now that the left motor is secured to the Left Arm Servo Plate it is time to reinstall it's servo arm and attach it to the Servo Double Arm. The manual illustrates this step pretty well. You'll be using the two self-tapping screws provided with the motor to attach it to the Servo Double Arm.

24







## Assembly – Steps 13, 14

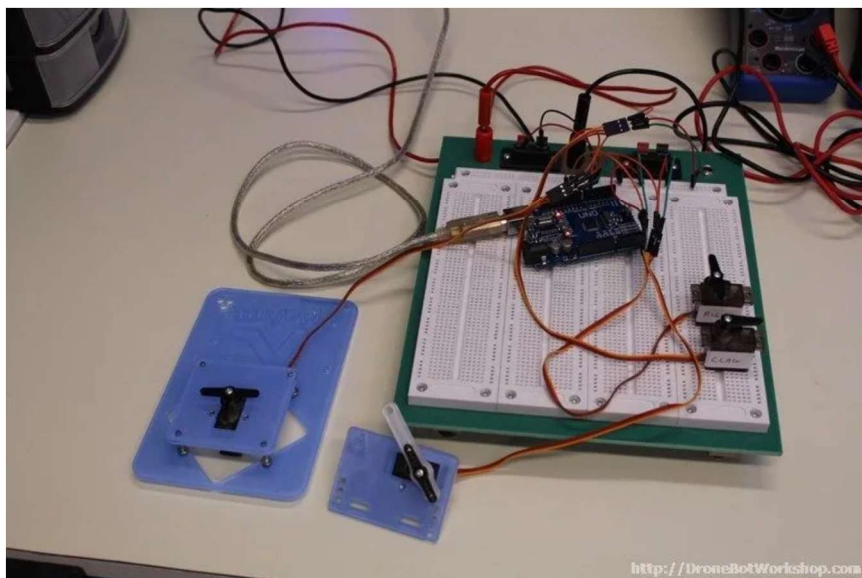
Now that the servo arm is attached to the Servo Double Arm you'll need to mount it on the Left servo motor. It's important to note the orientation of the arm, it should end up in the 9-o'clock position as illustrated in the manual. The arm attaches using the machine screw you used earlier when calibrating the motor.



## Assembly – Step 15



Step 15 is optional. In this step you reattach the motor to the Arduino, power it up and check that the servo is in the correct 9-o'clock position. I'd advise doing it as if the servo arms are incorrectly mounted it will come back to bite you when you finish the MeArm assembly.



## Assembly – Steps 16, 17

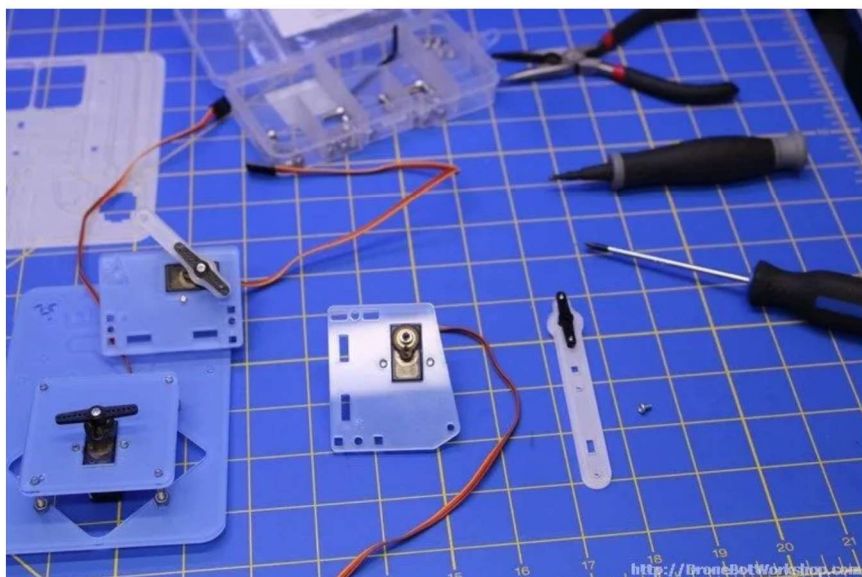
It's time to move on to the Right servo motor. As with the other ones it is attached using a servo collar, in this case to the Right Arm Servo Plate.





## Assembly – Step 18

In Step 18 we attach a servo arm to the Right Arm Base Joint using the two self-tapping screws provided with the servo motor.

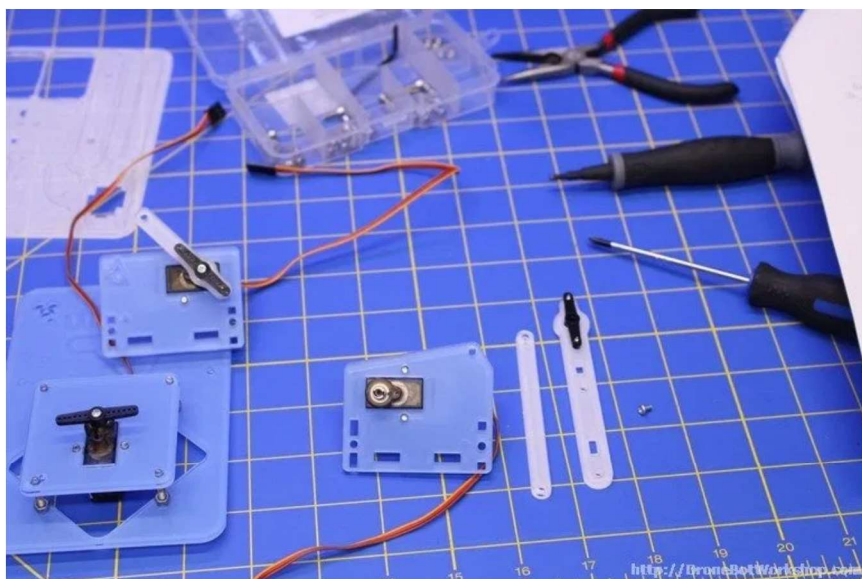


## Assembly – Steps 19, 20, 21 (22 not shown, with Arduino)

In Step 19 we will attach one of the Parallel Linkages to the Right Arm Servo Plate using a 6mm screw. Follow the diagram in the manual and be sure not to overtighten the screw as the Parallel Linkage needs to be able to move.

24





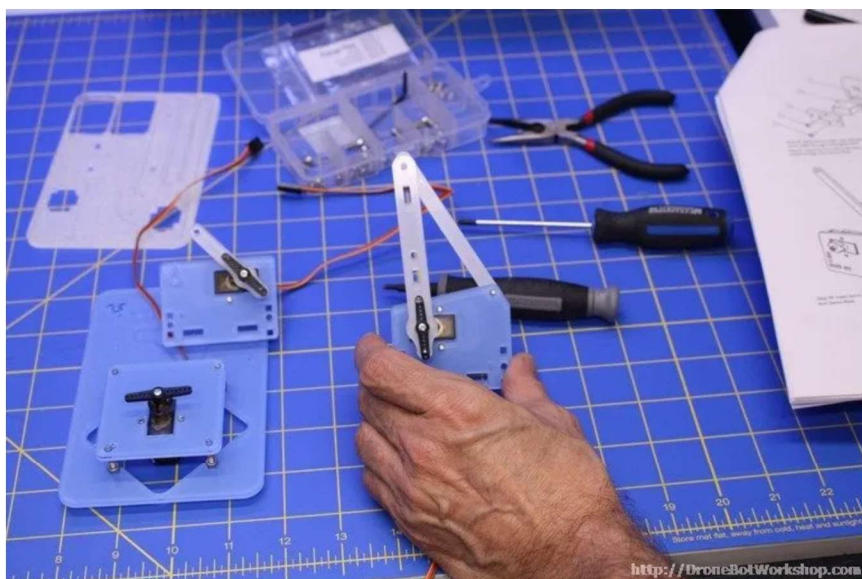
In Steps 20-21 the Right Arm Base Joint is attached to the Right servo motor using the machine screw from the servo motor pack. Note the orientation, it's pretty well a mirror image of the Servo Double Arm that you attached to the Left motor.



Step 22 isn't illustrated here, it's optional but worthwhile. Again you'll hook up the motor to the Arduino, power it up and verify that the orientation is correct. I advise that you don't omit this step, better to find out now that it's misaligned than to have to pull the whole assembly apart later!

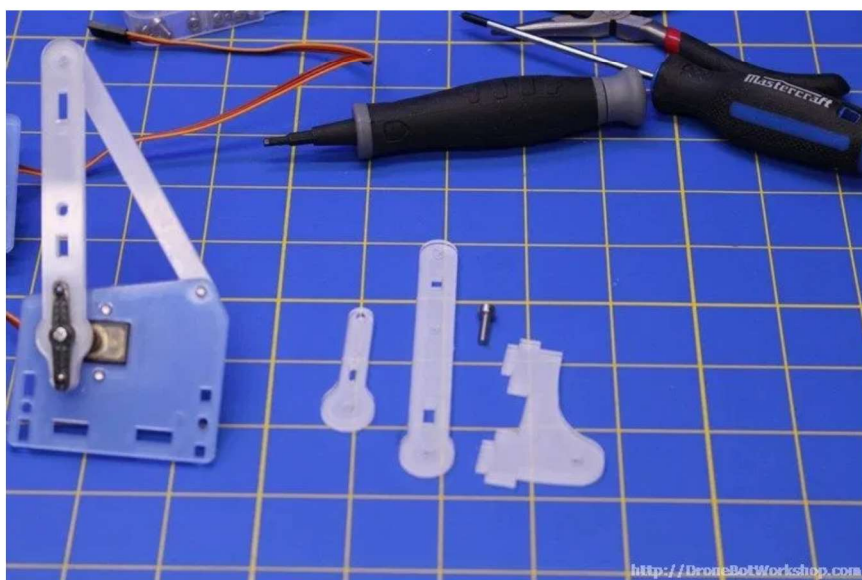






## Assembly – Step 23

This is amusing. The part we are about to assemble is called a “Pig”. I’m not exactly sure how it got this name (perhaps someone can enlighten me on that) but that’s what it’s called. So let’s build a pig!



24

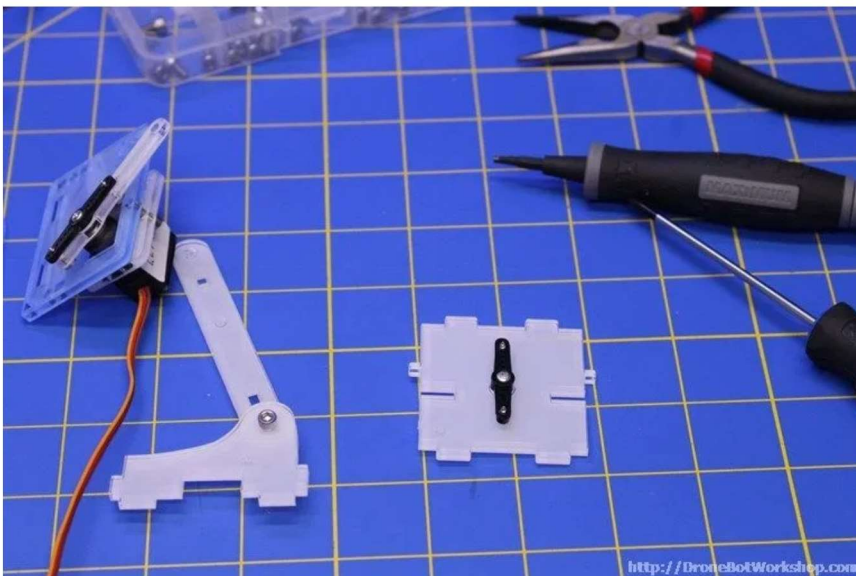
The “Pig” consists of three parts – the Short Servo Arm, the Left Arm Base Joint and the Left Arm Mount Tablet. These components are held together using a 10mm screw, which self-taps into the Short Servo Arm. Don’t over tighten the screw as our “pig” needs to move!





## Assembly – Steps 24, 25

We will now attach a servo arm to the Arm Bottom plate, this is the base of the MeArm. Again you'll use a couple of the self-tapping screws that came with the servo motor.



Step 25 is simply to leave the machine screw in the assembly to make it easier to attach it later on. It's good advice so do it!

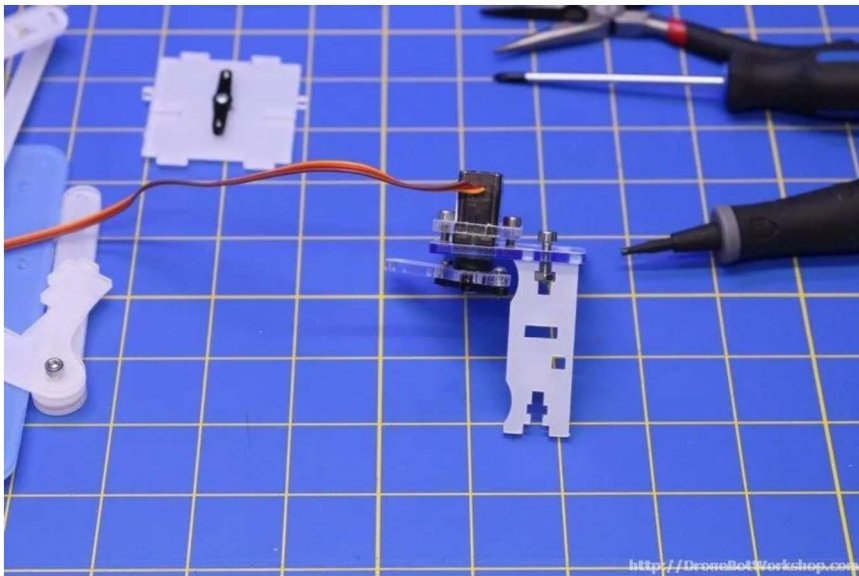
## Assembly – Step 26

Step 26 is to attach the Front Arm Cross Member to the Left Arm Servo Plate and motor assembly. This is done with a 12





mm screw and nut, which are used to secure the member once it is snapped into place.



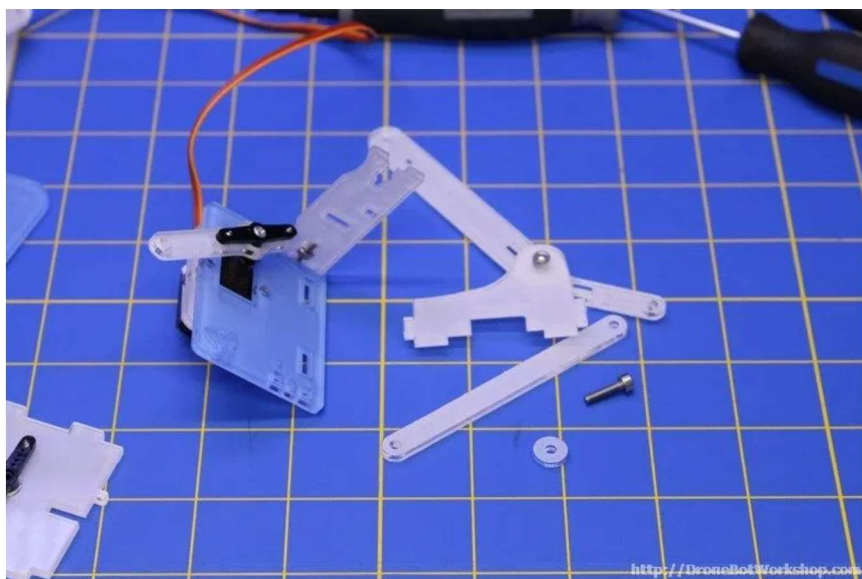
I performed this step by taking the instructions literally, mounting the nut to the screw and then snapping the whole assembly together. While this did work it was a bit awkward and in retrospect I wish that I'd snapped the Front Arm Cross Member to the Left Arm Servo Plate first and then installed the screw and nut. See what works best for you (the video shows that I'm struggling a bit with this step and with subsequent steps that use the same mounting arrangement).

## Assembly – Step 27

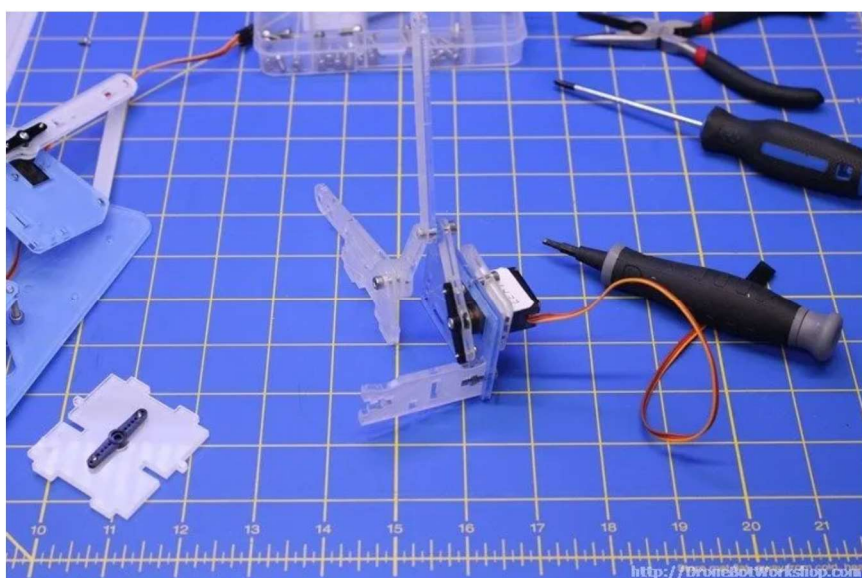
Time for the Pig! In this step the Pig is attached to a Parallel Linkage and the Front Arm Cross Member and then to the Long Servo Arm Extension. Follow the diagram carefully and you'll do just fine. Make sure to note the position of the spacer you'll need to complete this step.

24





Here is a view of the completed assembly:

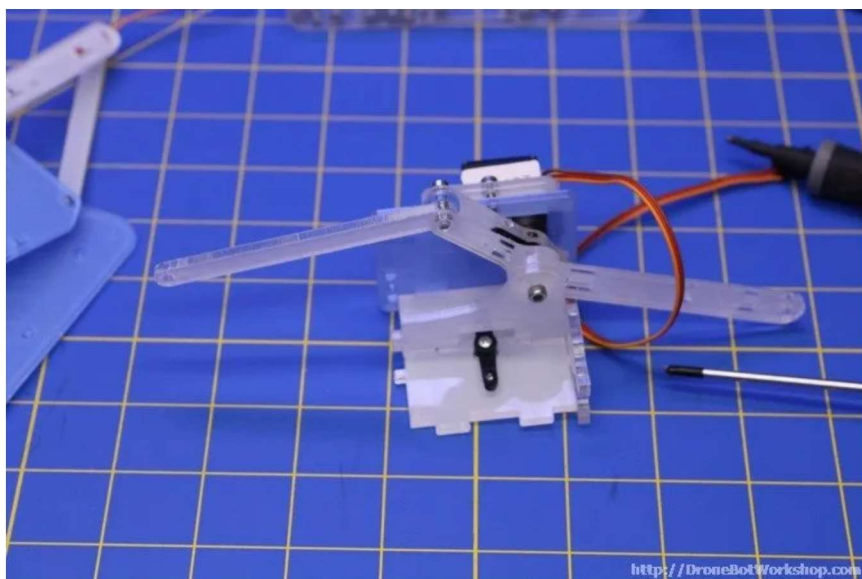


## Assembly – Step 28

In Step 28 you'll mount the Pig assembly you just put together to the Arm Base Plate. Everything snaps together, although it will take a bit of fiddling to make all the parts line up. Make sure to keep that small machine screw in place on the Arm Base Plate, you'll need it soon.

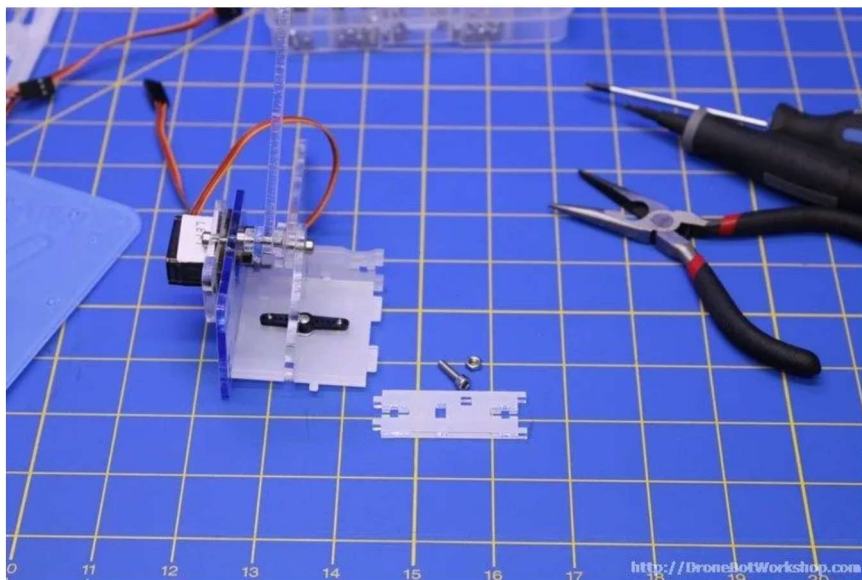






## Assembly – Step 29

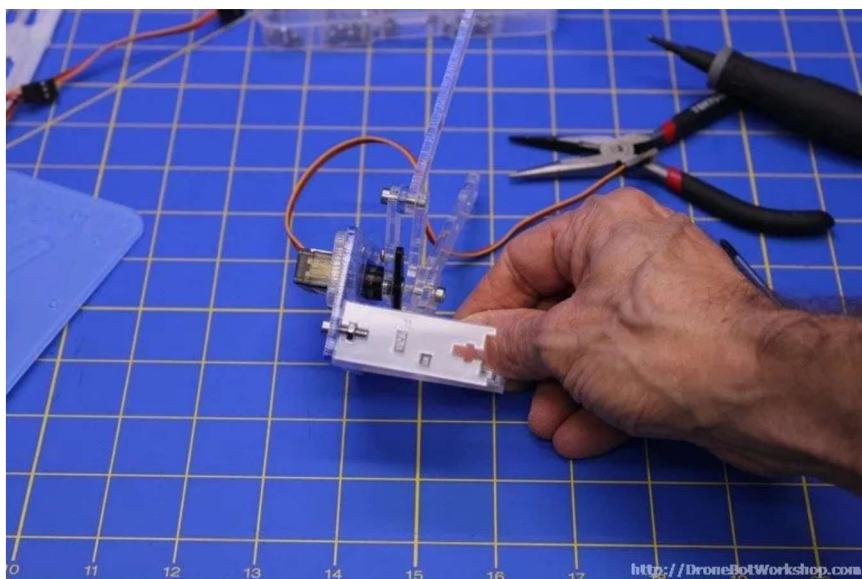
Now it's time for the Rear Arm Cross Member. Like it's Front counterpart this is held together with a combination of a 12mm screw and nut and by lining up the slots in the respective pieces. You'll also be mating it to the Arm Base Plate.



24

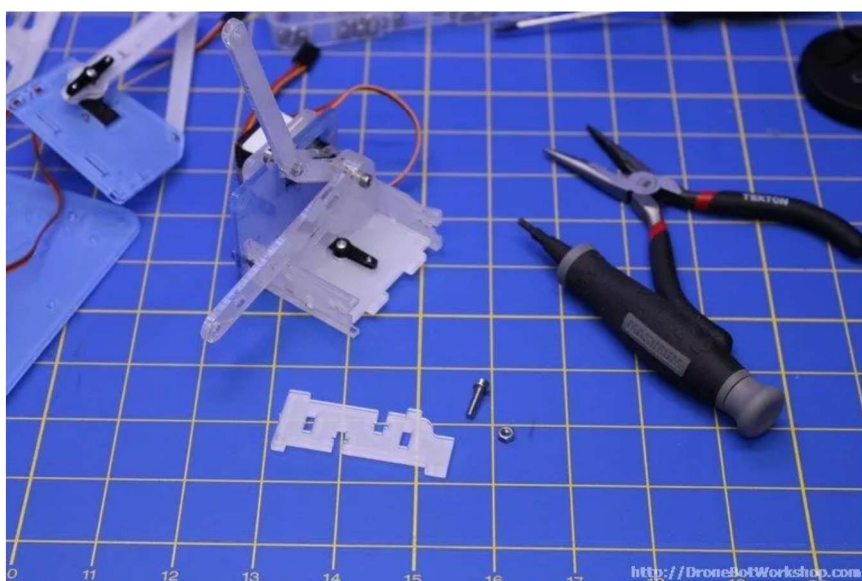
I found this to be a bit of a struggle (as you can see in the video) but with perseverance it will all fall into place. Once again I installed the screw and nut first but now wish I'd put them in last. Next time I build a MeArm I'll know better!





## Assembly – Step 30

The same mounting arrangement with the 12mm screw and nut, this time I got smart and put them on at the end. In this step we are installing the Main Arm Cross Web.



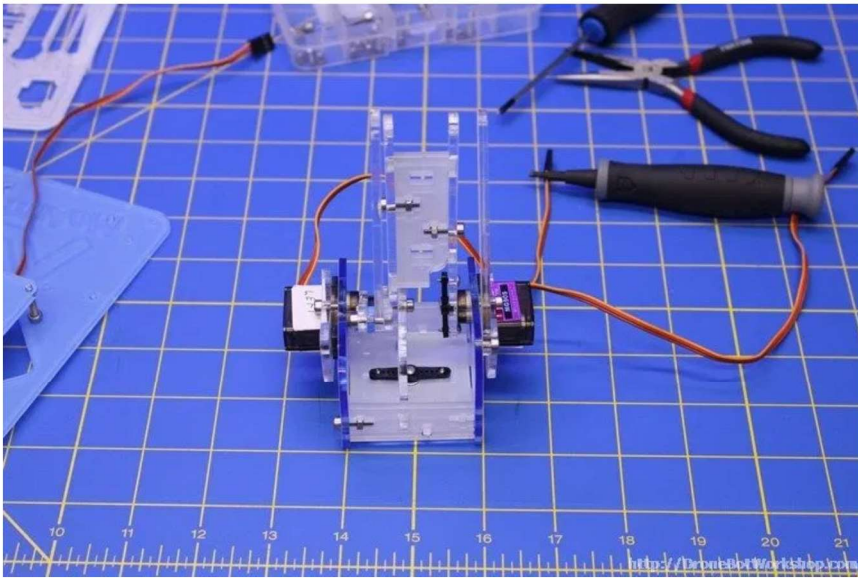
## Assembly – Step 31 24

Now we are going to mate the left and right sections of the arm. Although the manual warns that this is a bit tricky I actually didn't find it too difficult (I thought Step 29 was much harder to perform). Still you'll want to heed the warning about



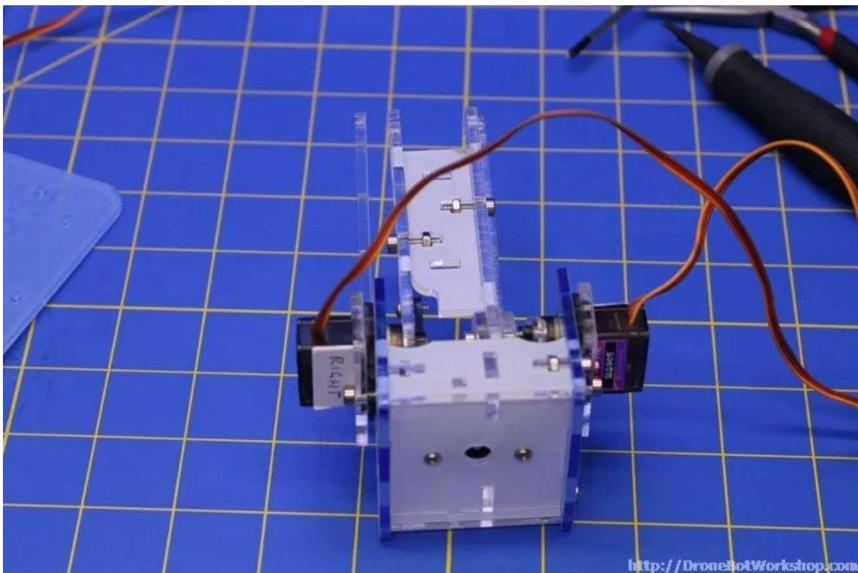


being gentle and having patience.



## Assembly – Step 32

This step is really just a continuation of the previous step. We are going to secure the two halves together using a pair of 12mm screws and some nuts.



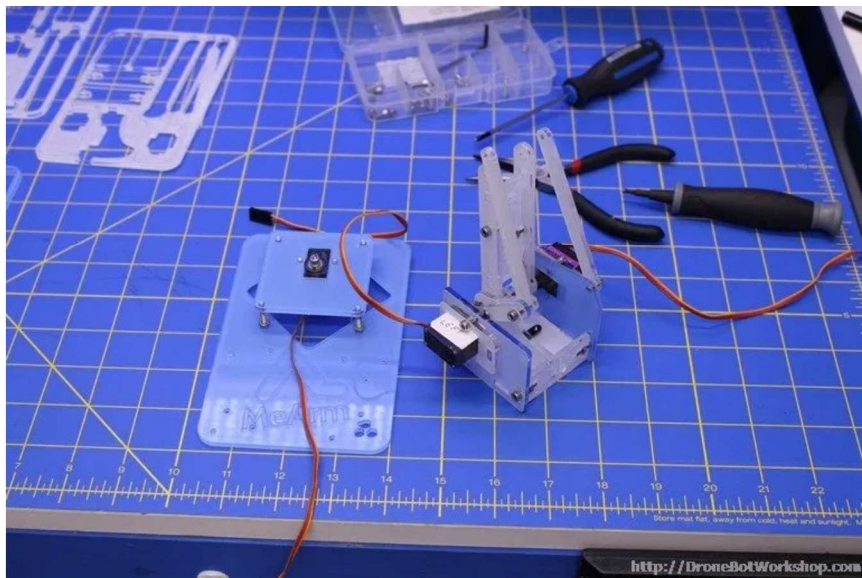
24

## Assembly – Step 33 (34 not shown, with Arduino)

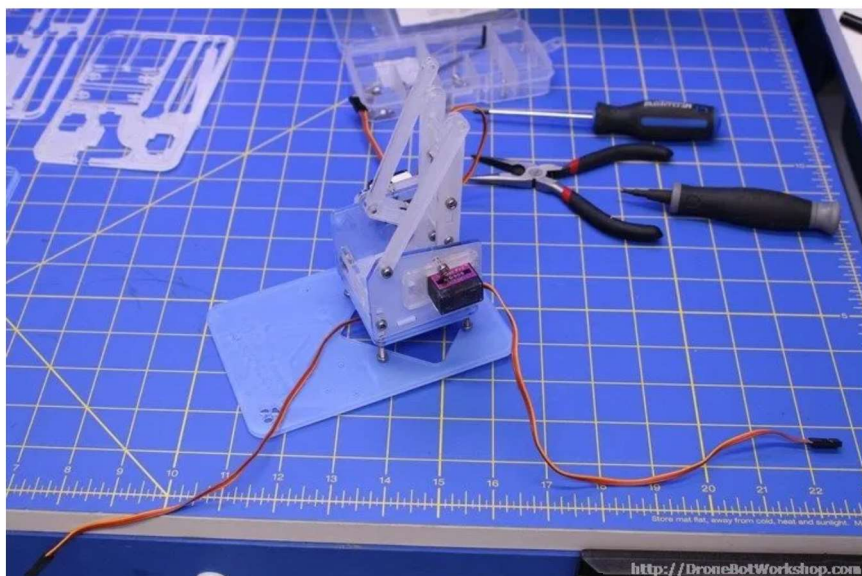
Remember that machine screw you were told to leave on the



Arm Base Plate? Now it's time to put it to use, mounting the Arm Base Plate to the motor on the Pivot Servo Plate. It's a pretty straightforward operation, just be sure to follow the illustration in the manual and line it up correctly.



With this step completed the MeArm is really starting to take shape!



24

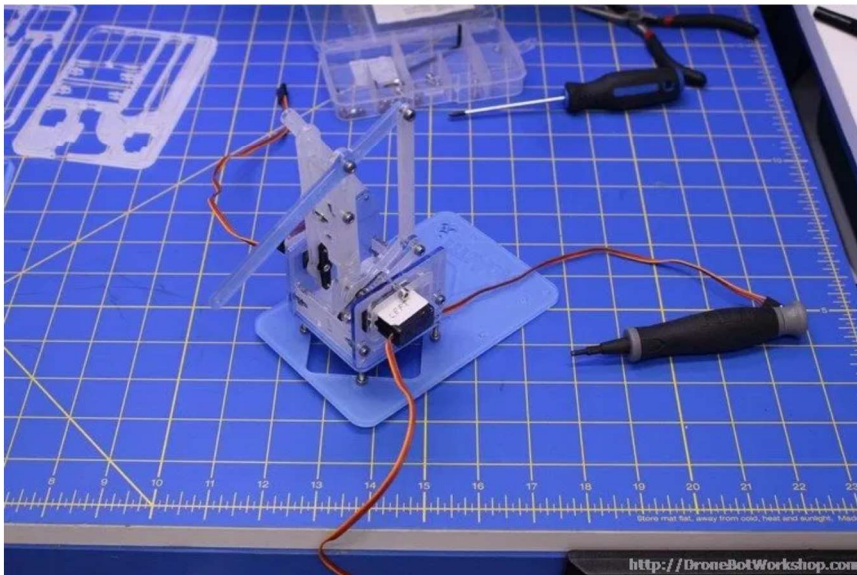
Step 34, which I haven't illustrated here, is to check the calibration with the Arduino. I would advise you to take the time to do that now.





## Assembly – Step 35

In step 35 you will be attaching the Left Wrist Joint to the assembly. It's held on with two 3mm screws. Don't over-tighten the screws as they serve as pivot points to allow the MeArm to move.



## Step 36 & 37 SNAFU! – Parts do not exist!

Well it was going pretty smoothly up until now. But all good things must come to an end I suppose, and this is where I met the end.

Steps 36 and 37 in the manual tell you to install the Wrist Joint Connector between the Left Wrist Joint and the Right Wrist Joint. Looks pretty simple in the diagram. But upon looking at my MeArm I discovered two things:

1. Part #18, the “Wrist Joint Connector”, was nowhere to be found. No part numbered 18 existed in my kit.
2. There also was no part #31, which was supposed to be the Right Wrist Joint.

Clearly the MeArm kit that I'd purchased in December 2016 was a bit different than the one that was documented for assembly in September 2015. So what to do?



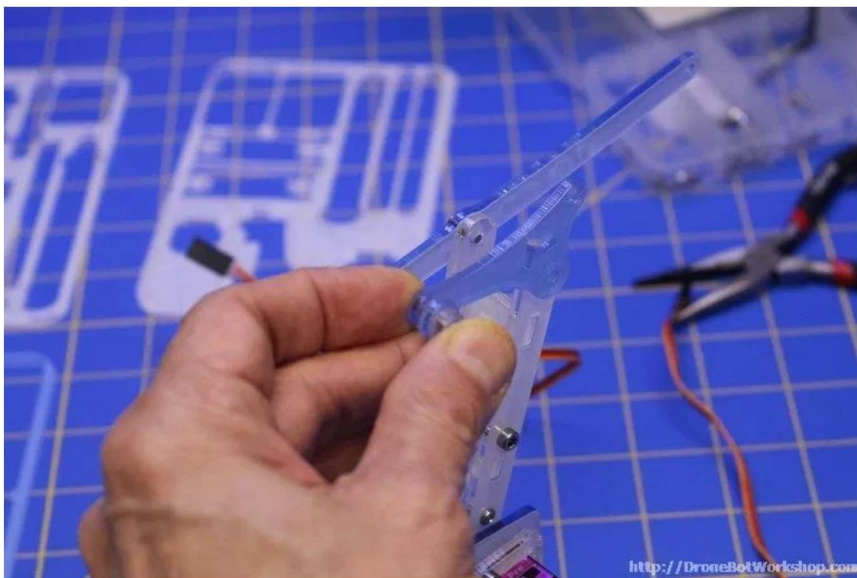
Upon observing the MeArm as it was so far and examining the parts I was provided I came to the following conclusions, which turn out to be correct:

1. There is no “Wrist Joint Connector”. The (existing) Left Wrist Joint doesn’t have any holes to accommodate it.
2. An unnumbered part that was just labeled “MeArm” is the Left Wrist Joint.

So at this point I proceeded by ignoring these instructions and moving on to step 38. It turned out to be the correct decision. Perhaps in future updates to the manual this will be corrected.

## Assembly – Step 38, 39 & 40

Now we will secure the unlabeled “Right Wrist Joint” with the Parallel Linkage Connector. Follow the video and the manual and it will all make sense. You’ll also be using the connector to attach to two Parallel Linkages that you’ve already installed.



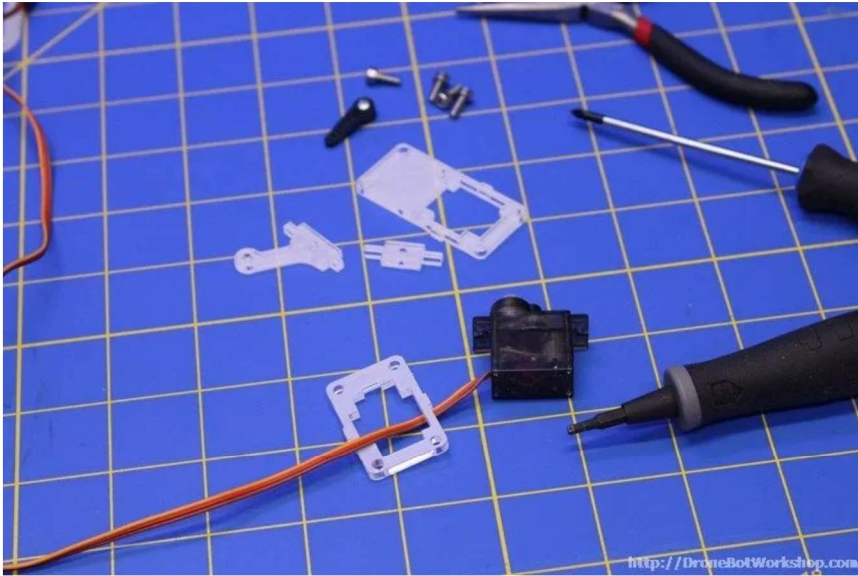
Make sure not to over-tighten these screws.

## Assembly – Step 41, <sup>24</sup>42, 43 & 44

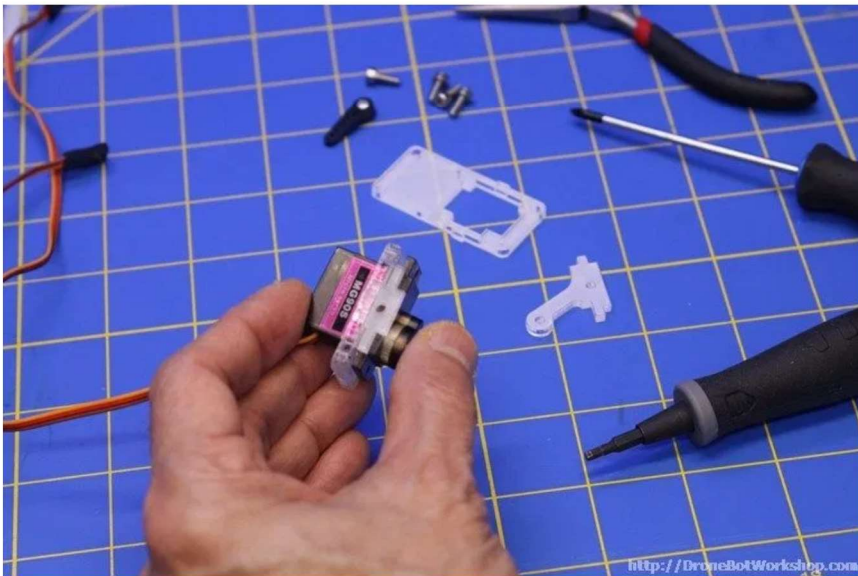
We are now on the final stretch now, it's time to work on the claw.





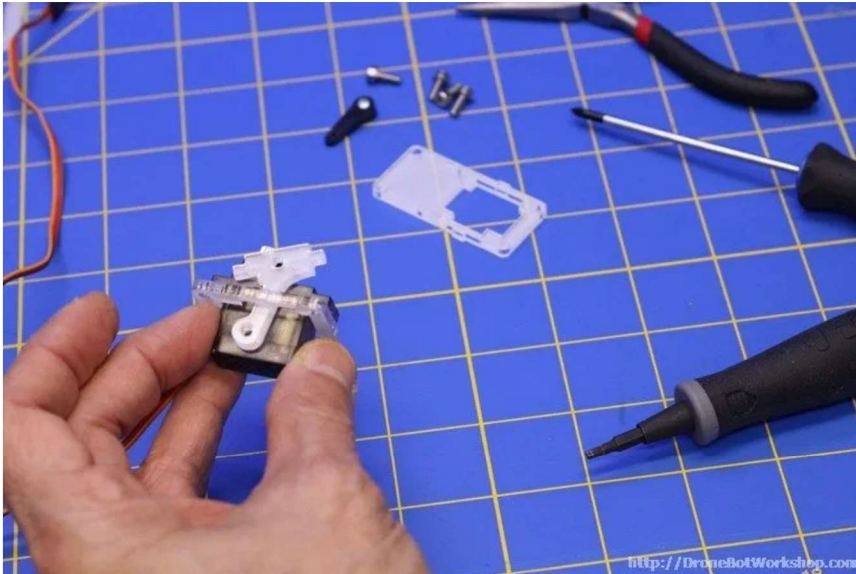


We will start by mounting the claw servo motor using a servo clamp collar. This is similar to the way we mounted the other three motors with one exception – there are two pieces (the Left and Right Wrist Attachment) that need to be inserted into the clamp collar before you put it all together.



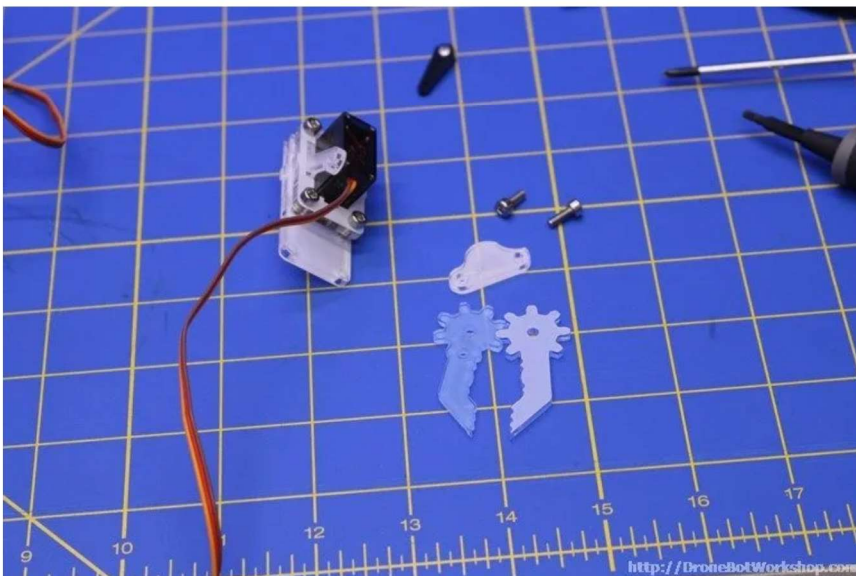
Follow the diagram carefully and watch the video to be certain that you've put these pieces in correctly. Four 8mm screws hold the pieces together.





## Assembly – Step 45

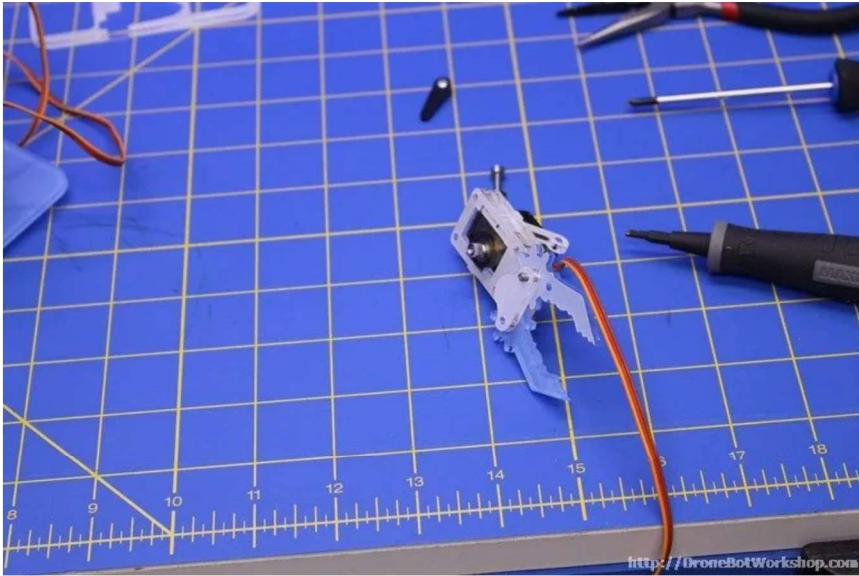
Step 45 is to install the grippers. Although the manual only tells you to install the Left Gripper you really need to put both of them in.



Mount the grippers onto the claw motor assembly using two screws, one 8mm and one 12mm. Below the grippers you'll be installing a Gripper plate. Again careful observance of the diagram and the video should clear up any confusion you might have regarding this step.

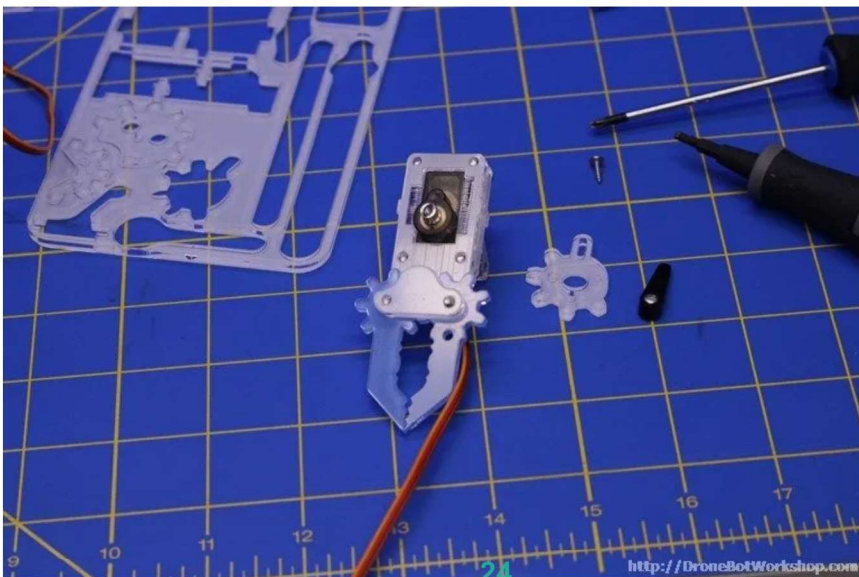






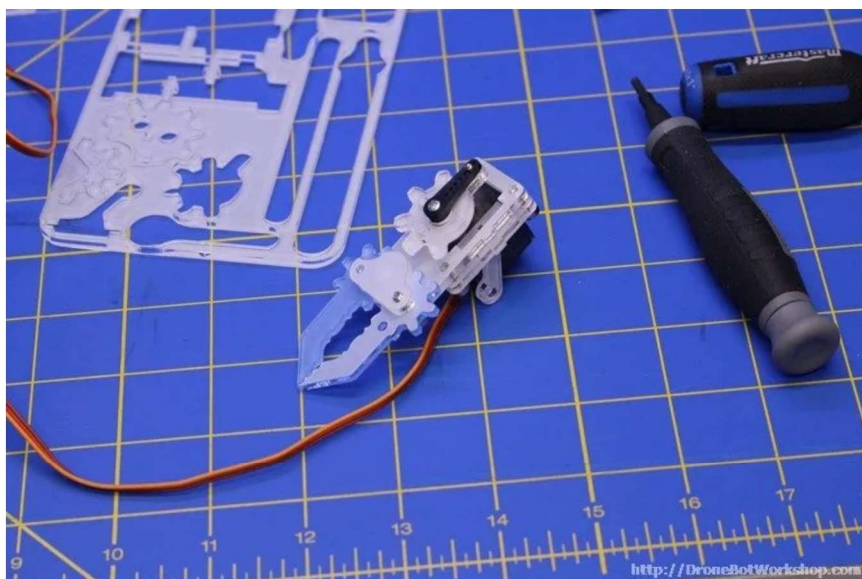
## Assembly – Step 46

Now we'll use the single servo arm (provided with the servo motor and used earlier during calibration) and one of the self-tapping screws (also provided with the servo motor) to attach the Top Servo Gear. The manual illustrates this step pretty nicely.



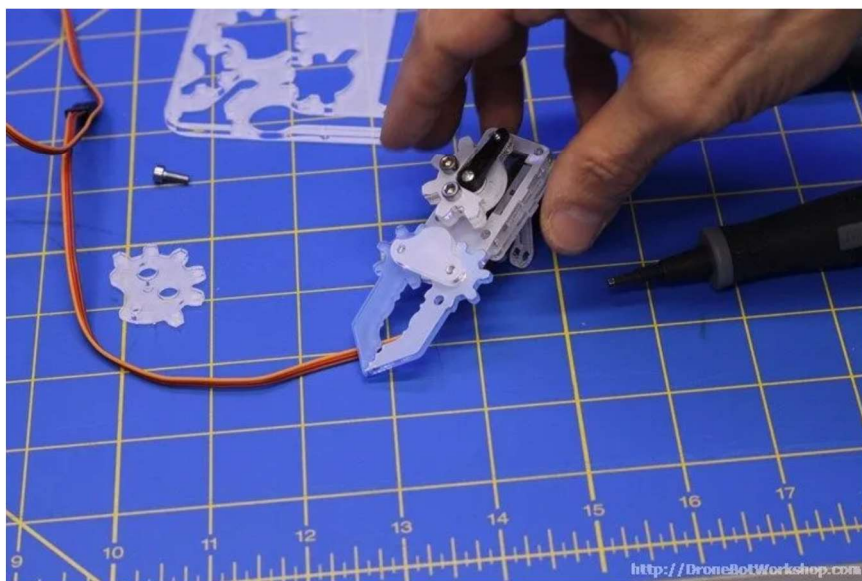
When that is done we use the machine screw to attach this assembly to the Claw servo motor.





## Assembly – Step 47

Step 47 is to connect the Bottom Servo Gear to the Top Servo Gear. This small piece is fastened using a 6mm screw.



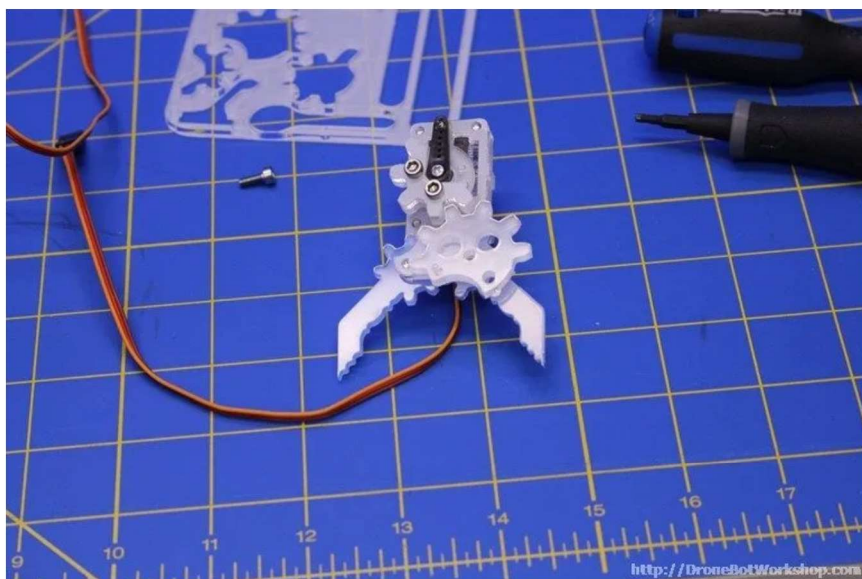
## Assembly – Step 48

24

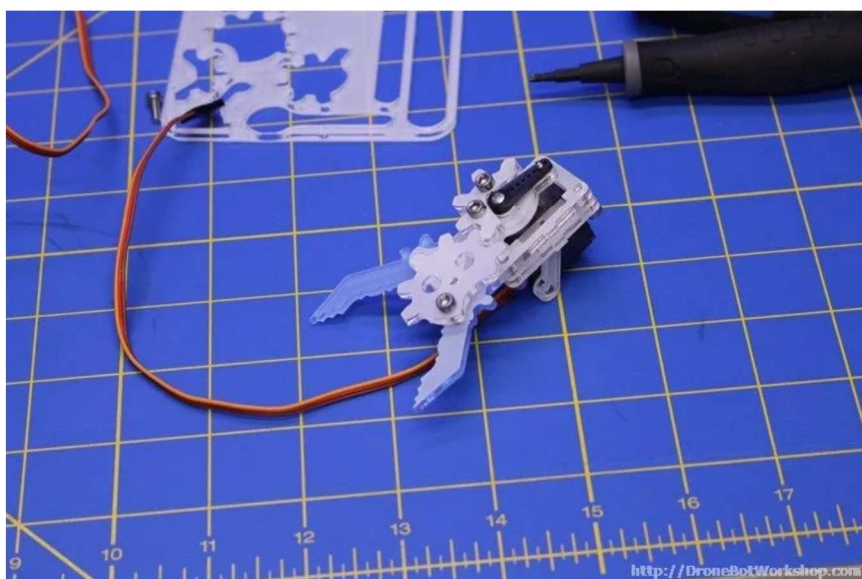
In this step we will attach the Gripper Gear to the claw assembly. The diagram for this in the manual is incorrect (and the manual makes note of this error), it shows the grippers in a closed position but instead they need to be open.







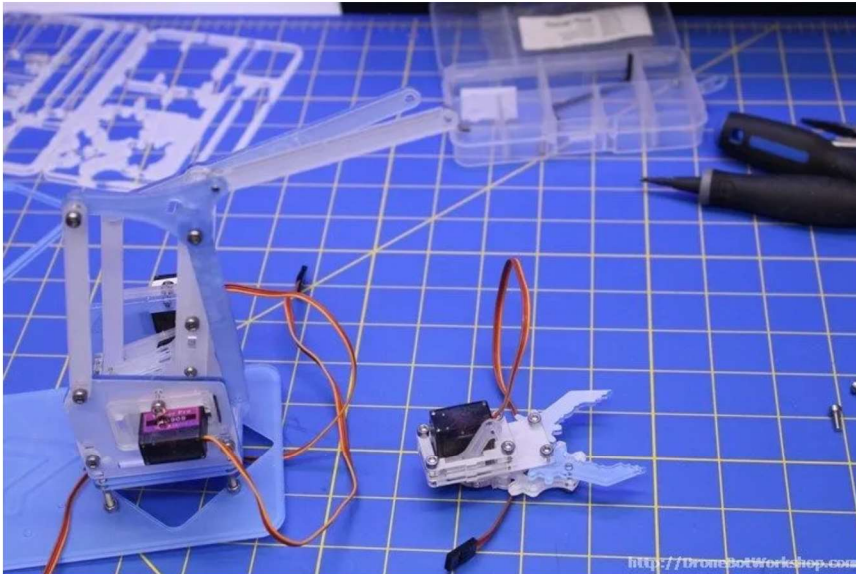
The Gripper Gear is attached using the 12mm screw we installed in Step 45 and an additional 6mm screw. Take a look at the video if you find the manual confusing.



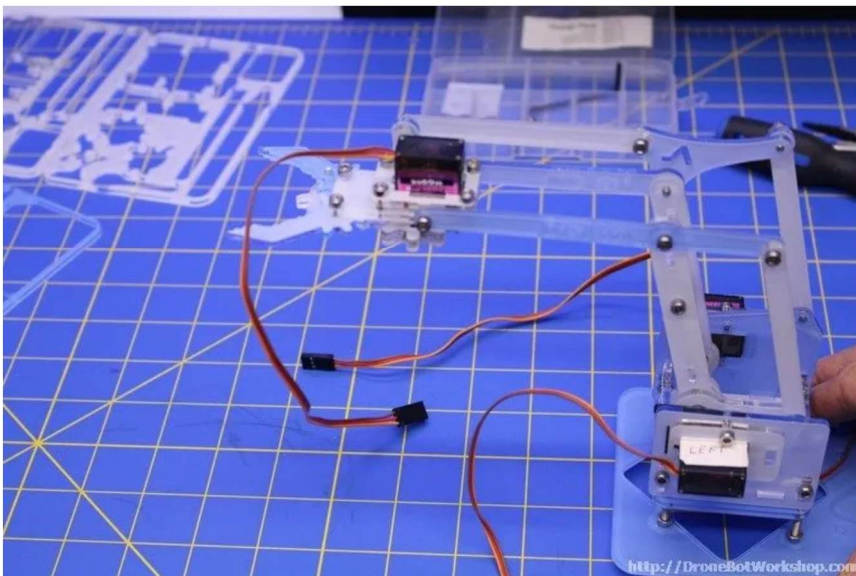
## Assembly – Step 49, 50 & 51

The MeArm is nearly complete! It's now time to attach the claw assembly to the rest of the arm. This is done using three 8mm screws, two on the right and one on the left. You'll also need the remaining spacer, which is used to align the Parallel Linkage on the right side.





I found it easier to complete this step by placing the arm on its side, as shown in the video.



## Missing Feet (from step 4)

Yes I missed step 4, sticking on the feet! I'll assume you have already done this, but if you haven't <sup>24</sup> then peel the feet off of the backing material and stick them on the base. Make sure not to block the mounting holes that you'll be using for the Arduino.

## Assembly – Step 52





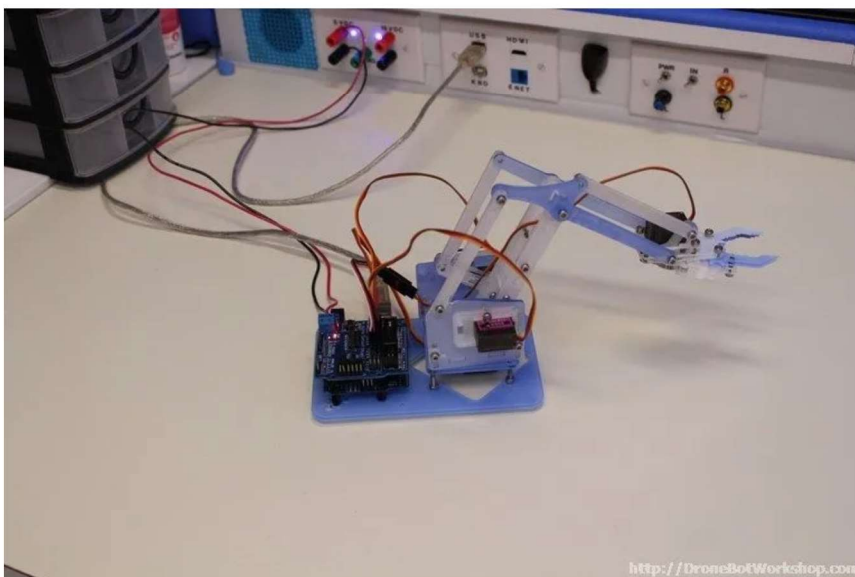
The last step in constructing the MeArm is to simply thread the cable from the claw servo motor through the guide in the Main Arm Cross Web. After you've done that attach the small servo extension cable to it so that it can be connected to the Arduino.

## Finishing the MeArm

Your MeArm is completed, all that remains to do is hook it up to a controller and try it out.

If you are using an Arduino Uno you'll find that there are mounting holes already drilled into the base to accommodate it. Unfortunately the MeArm Kit doesn't come with spacers so you'll need to supply your own.

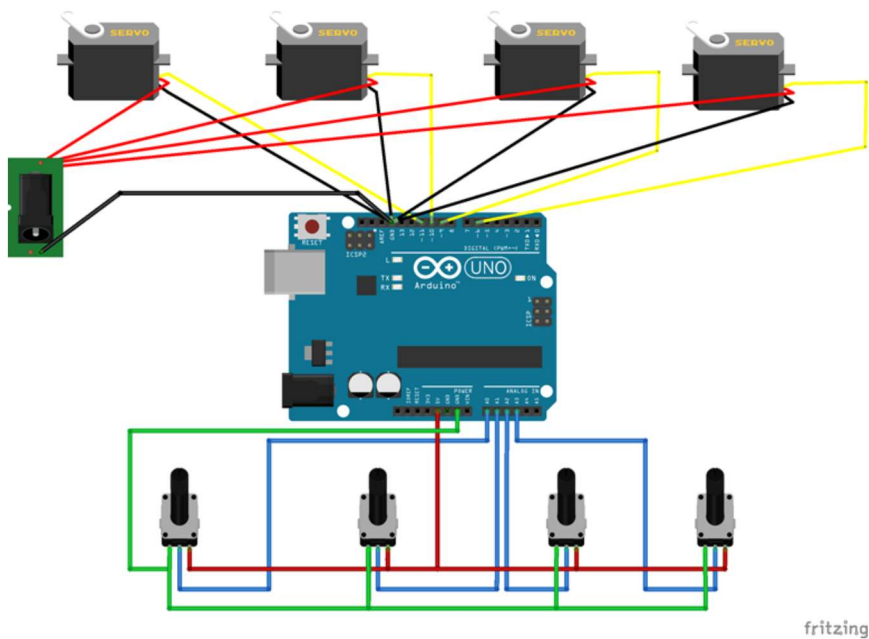
Although I plan on building the "Ultimate MeArm Controller" (which I'll document in the next article and video in this series) I did start out using an Arduino Uno and you probably should too, just to put the MeArm through its paces.



I attached an I/O shield to my Arduino Uno to simplify connecting the MeArm servos and four potentiometers. This handy shield provides connectors for the servos and the ability to power them off of a separate 5 volt power supply.

Here is a diagram of the hookup to the Arduino:





Once I had this connected I wrote a simple sketch to drive the servos based upon the analog inputs from the four potentiometers. I've included it here, feel free to modify it to suit your needs.

```

1  #include <Servo.h>
2
3  Servo middle, left, right, claw ; // Creates 4 "servo objects"
4
5  int middlePin = A0; // Middle Pot Analog Input Pin
6  int leftPin = A1; // Left Pot Analog Input Pin
7  int rightPin = A2; // Right Pot Analog Input Pin
8  int clawPin = A3; // Claw Pot Analog Input Pin
9
10 int middleValue = 0; // Middle Pot Value
11 int leftValue = 0; // Left Pot Value
12 int rightValue = 0; // Right Pot Value
13 int clawValue = 0; // Claw Pot Value
14
15
16
17
18 void setup()
19 {
20   middle.attach(11); // attaches the servo on pin 11 to the middle object
21   left.attach(10); // attaches the servo on pin 10 to the left object
22   right.attach(9); // attaches the servo on pin 9 to the right object
23   claw.attach(6); // attaches the servo on pin 6 to the claw object
24 }
25
26 void loop()
27 {
28   // Read the Pot Values
29   middleValue = analogRead(middlePin);

```



```
31 leftValue = analogRead(leftPin);
32 rightValue = analogRead(rightPin);
33 clawValue = analogRead(clawPin);
34
35 // Convert to values between 0 and 180 for the servos
36 middleValue = map(middleValue, 0, 1023, 0, 180) ;
37 leftValue = map(leftValue, 0, 1023, 0, 180) ;
38 rightValue = map(rightValue, 0, 1023, 0, 180) ;
39 clawValue = map(clawValue, 0, 1023, 0, 180) ;
40
41 // Write values to servos
42 middle.write(middleValue); // Middle servo position
43 left.write(leftValue); // Left servo position
44 right.write(rightValue); // Right servo position
45 claw.write(clawValue); // Claw servo position
46
47 delay(400); // Short Delay
48 }
```

Power everything up and give your MeArm a try. If you find some of the joints binding you may have overtightened some of the screws, use the supplied Allen key to take care of those.

Congratulations, your MeArm is complete!

## MeArm Conclusion

I found the MeArm to be an enjoyable project to build. It was pretty well documented although I'm hoping that for future iterations they update the manual to reflect the current design (and of course I still think they should have told me to PEEL OFF THE PLASTIC). But despite these minor setbacks I think the kit is a good value and would recommend it to anyone who wants to develop software and sensors for a robotics arm.

In the next article and video in this series I'll go over the construction of a deluxe MeArm controller. Please stay tuned for that and enjoy your new MeArm!

## Resources

24

[Arduino Sketches](#) – The Arduino Sketches used in this article

[MeArm Manual](#) – The MeArm Manual for Arduino

[Purchase a MeArm](#) – Mime Industries MeArm page



[MeArm Laser Cut Files](#) – Cut your own MeArm parts from these files from Thingiverse

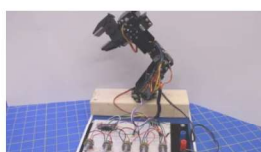
## Related



### [Programming the Arduino Pro Mini](#)

April 13, 2019

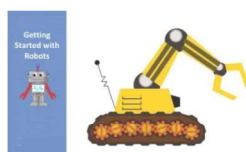
In "Arduino"



### [DFRobot 5 DOF Robot Arm - Building the Robotic Arm](#)

June 3, 2018

In "Arduino"



### [Getting Started Building Robots](#)

February 6, 2016

In "Robots"

Tagged on: [Robot Project](#)

DroneBot Workshop February 14, 2017

Project, Robots 24 Comments

[← Getting Started Building Robots](#)

[Build a Robot Car with Speed Sensors →](#)

## If you have a question...

Comments about this article are encouraged and appreciated. However, due to the large volume of comments that I receive, it may not be possible for me to answer you directly here on the website.

You are much more likely to get answers to technical questions by making a post on the [DroneBot Workshop Forum](#). Your post will be seen not only by myself, but by a large group of tech enthusiasts who can quickly answer your





question. You may also add code samples, images and videos to your forum posts.

Having said that, please feel free to leave constructive comments here. Your input is always welcome. Please note that all comments may be held for moderation.

*Join the discussion*

**B** *I* U         

## 24 COMMENTS



Oldest ▼


**Inam Ulla**  4 years ago

Hello sir,  
how should i connect the potentiometer in this project and how to connect a i/o shield to arduino

+ 0 —  Reply

Author

**DroneBot Workshop**  4 years ago

 Reply to [Inam Ulla](#)

Hi Inam The I/O Shield that I used is an Arduino Shield, it just snaps onto the Arduino Uno and has the digital output pins marked. It makes it a lot easier to connect the servo's and the potentiometers, but you can actually do it without. Just connect the servos the same way you were instructed to in the MeArm manual and the code will work. If



you do use the shield remember to power the servos with their own 5 volt supply, don't use the 5 volts from the Arduino. The four potentiometers each have one side grounded and... [Read more »](#)

+ -1 - ➔ Reply

**Inam Ulla** ⌚ 4 years ago

which version or model of i/o shield you have used and how is the connection for potentiometer and servo in the i/o shield  
plzzzzzz tell me

+ 0 - ➔ Reply

Author

**DroneBot Workshop** ⌚ 4 years ago

🗨 Reply to [Inam Ulla](#)

An example of the shield is this one at eBay – <http://ebay.to/2m6t64S>. If you search “Arduino Sensor Shield” you’ll come up with a bunch of them. Mine is made by Sainsmart, it’s “Arduino Sensor Shield v5.0”. Each analog input to the Arduino has a corresponding connector on the shield. Each connector has 3 pins, labeled G (for Ground), V (for voltage, the +5volts) and S (for sensor). Use the A0 to A3 inputs as I described in my last comment. Connect one end of each potentiometer to the inputs “G” pin, the other end to the “V” pin. Connect the wiper (i.e.... [Read more »](#)

+ 0 - ➔ Reply

**Inam Ulla** ⌚ 4 years ago





Thankyou for your kind reply.Can you please tell me without using the shield i can construct the mearm or not

+ 0 — ↩ Reply

**Inam Ulla** ⌚ 4 years ago

I also request you to send the picture of the shield while you have connected to servos and arduino and potentiometer from nearer because in the video it is from far away which is difficult to appear my whatsapp no. is +918884918707 or email me. I hope you will not deny my request. Thankyou

+ 0 — ↩ Reply

**Inam Ulla** ⌚ 4 years ago

Sir please reply me. I want to submit this project at the last of this month in my college. and i don't know why this motors are not working properly

+ 0 — ↩ Reply

**Inam Ulla** ⌚ 4 years ago

As you told that to connect the middle ( wiper pin of the potentiometer) to the analogue input on the shield which connects the analoge pin in the arduino board.

But what about the 3pins of the servo motor where should it be connected on the shield.

+ 0 — ↩ Reply

24

Author

**DroneBot Workshop** ⌚ 4 years ago

↩ Reply to [Inam Ulla](#)

I have added a Fritzing diagram to the



article that shows the hookup to the Arduino Uno. The motor hookup is identical to the one illustrated on page 3 of the MeArm Manual.

the shield just makes the connections a lot easier, it is literally “plug & play” as it is already designed to have connections for the servos. Just be sure to power the servos from their own 5 volt power supply.

+ 0 — ➤ Reply

Author

**DroneBot Workshop** ⌚ 4 years ago

🗨 Reply to [Inam Ulla](#)

Hi Inam

I have just added a diagram that explains the hookup to the Arduino Uno, it's near the code at the bottom of the article.

+ 0 — ➤ Reply

**Inam Ulla** ⌚ 4 years ago

Thankyou for your kind reply. Sir can you tell what is the need of the serial.begin(4200) in the programme.

+ 0 — ➤ Reply

**Inam Ulla** ⌚ 4 years ago

sorry it is Thankyou for your kind reply. Sir can you tell what is the need of the serial.begin(9600) in the programme.

+ 0 — ➤ Reply

Author





**DroneBot Workshop** 4 years ago

Reply to [Inam Ulla](#)

Hi Inam

Good catch, it actually isn't required!

I developed the sketch originally by reading back the values sent to the motors via the serial connection.

Afterwards I removed the serial functions (as they were just used during development) but I neglected to remove the "serial.begin(9600)" statement.

Thank you for pointing out the error, hope you've been able to get your MeArm working. I'm building a more advanced controller for the MeArm now and will publish the plans for it soon.

+ 0 — Reply

**Jithu** 4 years ago

Great video, helped me build up the mechanical parts.

A question regarding the sainsmart Arduino sensor shield – power separation related stuff

– We should remove the SEL jumper adjacent to shield power screws right ?

– Should I plug in a jumper between the shield external power GND and something to have a common GND (or is it already taken care by the shield ?)

+ 0 — Reply

24

Author

**DroneBot Workshop** 4 years ago

Reply to [Jithu](#)




Glad you enjoyed the video Jithu.

– Yes you are correct, you need to remove the jumper so that you can use the external power supply for the servo motors.

– The shield already has a common ground connection that's internally connected to the negative line of the external power supply, so no need to supply one yourself.

+ 0 –  Reply

**Meme**  4 years ago


Very good video!

Just one question, what type of potentiometers are you using? Could 10K linear potentiometers work?

+ 0 –  Reply

Author

**DroneBot Workshop**  4 years ago


 Reply to [Meme](#)

Sorry for the late reply to your question.

10k linear pots would work just perfectly, in fact that's exactly what I'm using. Any linear taper pot over 5k would work well, under 5k and the 4 pots would start to consume a bit more current than I would want to waste on heating resistors!

+ 0 –  Reply

24

**Varun**  3 years ago

Hi I've finished the MeArm construction and I just need to build the controller. I am planning to do it without the shield and I was wondering where I



can buy the potentiometers and also would it be possible for someone to explain the green and red wiring for them as they seem to connect to multiple potentiometers together. I don't have a wire that can do this. Thanks

+ 2 — ↩ Reply

**Varun** ⌚ 3 years ago

I also have a breadboard I can use to help with the wiring.

Thanks

+ 0 — ↩ Reply

**Greg S** ⌚ 2 years ago

Great instructions – thanks! I am having trouble with the Left servo calibration and hope you can help. Your instructions and the MeArm manual both say to calibrate it to align with the part – and both mention 9-o-clock position. But your photo at Steps 13, 14 and Step 15, Step 27 and also the diagram in the manual all show the part pointing closer to 10:30 or so. And furthermore it appears that Step 28 requires the servo to be in the 10:30 position in order to fit the 12mm screw into it. Did I miss something somewhere? Can... [Read more »](#)

+ 7 — ↩ Reply

**sumanth** ⌚ 1 year ago

Hi Sir thanks for the detailed instructions.

I bought new me arm Kit. In kit i found "Gripper Gear" is missing for Claw. is it possible to build myself and what could the dimension to build it. Or is it possible for you to send the 3d design for Gripper Gear to build

+ 0 — ↩ Reply





**Moeketsi Mpanya** 10 months ago

I'm a 66 year-old very first timer in robot arm assembly and am scared of making a fatal mistake. What mistake(s) could result in the robot arm getting damaged or non-functional? What should I watch out for?

+ 0 — Reply

**nick** 2 months ago

Reply to [Moeketsi Mpanya](#)

+ 0 — Reply

**nick** 2 months ago

+ 0 — Reply

## Projects

Arduino Projects

Drone Projects

Electronics Projects

Internet of Things Projects

Raspberry Pi Projects

Robot Projects

## Tutorials

Arduino Tutorials

Drone Tutorials

Electronics Tutorials

Internet of Things Tutorials

Raspberry Pi Tutorials

Robot Tutorials

Software Tutorials

## What's New

Using GPS Modules with Arduino & Raspberry Pi

Arduino Nano RP2040 Connect – Arduino meets Raspberry Pi

Arduino Nano 33 IoT Board

Build an ESP32CAM Robot Car

Learning Python – A List of Free Resources

Raspberry Pi Pico with

## What's Popular

Getting started with the ESP32-CAM

Getting started with ESP32

Stepper Motors with Arduino - Getting Started with Stepper Motors

Measure DC Voltage and Current with an Arduino

Convert an ATX Computer Supply to a Bench Power Supply

[CircuitPython](#)

[Raspberry Pi Pico – How to Interface \(almost\) Everything!](#)

[Measure DC Voltage and Current with an Arduino](#)

[Getting Started with PlatformIO](#)

[Using Rotary Encoders with Arduino](#)

## Workshop Connections



Copyright © 2021 DroneBot Workshop.

[Home](#) [Arduino](#) [Raspberry Pi](#) [Drones](#) [Robots](#) [Electronics](#) [IoT](#) [Reviews](#) [Tutorials](#) [About Us](#) [Contact Us](#)  
[Privacy and Cookies](#)

24

