6DOF Robotic Arm

Manual

AK-ARM

Table of Contents

Component List

Component Name	QTY
6DOF Robotic Arm Frame Kit	1
MG995 Servo Motors	6
Servo Discs	6

Frame Kit Parts:

Component Name	QTY	Code
Multi Function Brackets	4	3
Long U Shaped Brackets	3	2
Mechanical Claw	1	1
L Shaped Bracket	1	4
Bass	1	10
4 Packs of Required Screw Nuts	1	11
Cup Bearing	1	5
Big Rotative Holder	2	6
Round Disc Cover For Motor	1	9
Small Rotative Holder	2	7 & 8

Screws:

Component Name	QTY
M3 x 40 pillars	4
M3 x 6+6 pillars	4
M3 x 10 pillars	2
M3 x 6 pillars	2
M3 x 14 Nut	2
M3 x 10 nut	6
M3 x 8 Nut	2
M3 x 6 nut	4
M3 x 4 nut	4
M3 x 6 nut	4
M4 x 8 nut	24
M4 screw nut	24
M3 x 10 nut	3
M3 x 7 nut	12
M3 x 6 nut	8
M3 x 5 nut	18
M3 screw nut	15
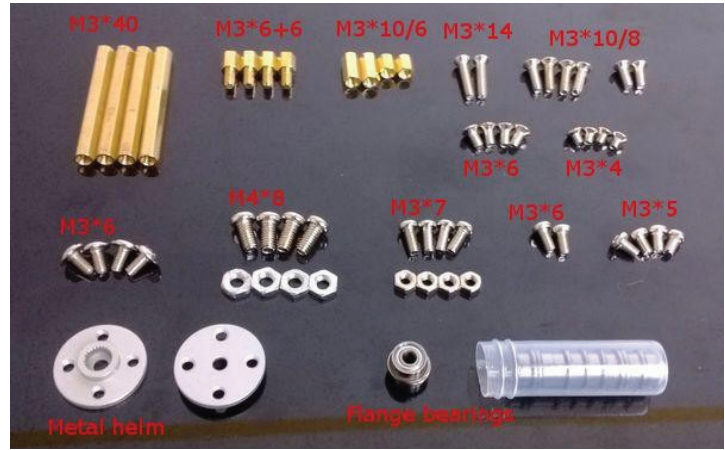
Tools Required:

(not included)

Straight Sharp Tweezers

Straight Blade Screwdriver
Phillips Head Screwdriver

Long Nose Pliers

5V 10A Switching Power Supply

A robotic arm is a type of mechanical arm usually programmable with similar functions to a human arm. The arm may be the sum of the mechanism or may be part of a more complex robot. The links of such a manipulator are connected by joints allowing either rotational motion or translational displacement. The links of the manipulator can be considered to form a kinematic chain. The terminus of the kinematic chain of the manipulator is called the end effector and it is analogous to the human hand.

In this manual You will learn how to assemble as well as code a robotic arm. This kit inculdes the frame and motors that are required to build the 6DOF Robotic arm. Shown below is the diagram for the robot arm.
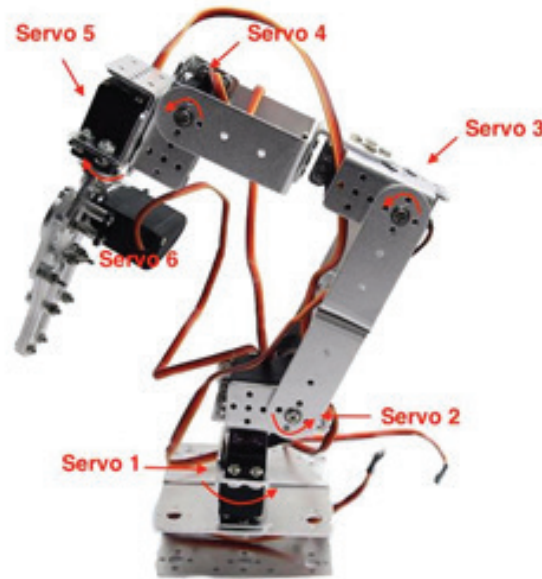
What is 6 DOF:

DOF stands for Degrees of freedom. It means that in how many directions or axis's the ARM can move freely. This arm you are building uses 6 angles of freedom. Six degrees of freedom (6DoF) refers to the freedom of movement of a rigid body in three-dimensional space. The body is free to change position as forward/backward (surge), up/down (heave), left/right (sway) translation in three perpendicular axes, combined with changes in orientation through rotation about three perpendicular axes, often termed yaw (normal axis), pitch (transverse axis), and roll (longitudinal axis).

Installing Mechanical Parts

Step 1: Installing Mechanical Parts

Required:
- 1 x Multifunction bracket
- 1 x small rotative holder
- 4 x Round M4 x 6 screws
- 4 x M4 Nut

With the help of one multifunction bracket and one small rotative holder, you need to affix these parts together using the round M4x6 screws as shown in the figure below.
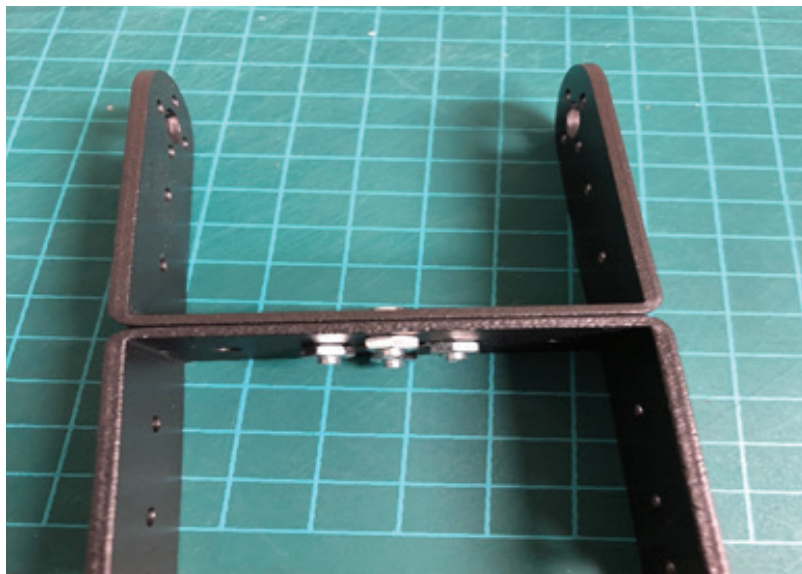

Step 2: Installing The Long Arm Of The Robot Structure

Required:
- 2 x U Long bracket
- 4 x Round M3x7 screw
- 4 x M3 Nut

There are two long U shape brackets in the package. Attach those both using M3 screw and nuts. Below is the picture shown after attaching both U shape brackets.

Step 3: Installing Mechanical Arm Structure

Required:
- 1 x multi function bracket
- 2 x L-shaped bracket
- 1 x long U-shaped bracket
- 8 x Round M3 x 7 screw
- 8 x M3 Nut

Take one L-shaped bracket, one Multi function bracket and one long U- shaped bracket. The picture below demonstrates how to assemble. First affix the U shape bracket to the L- shape bracket and then affix it to the multi-functional bracket.
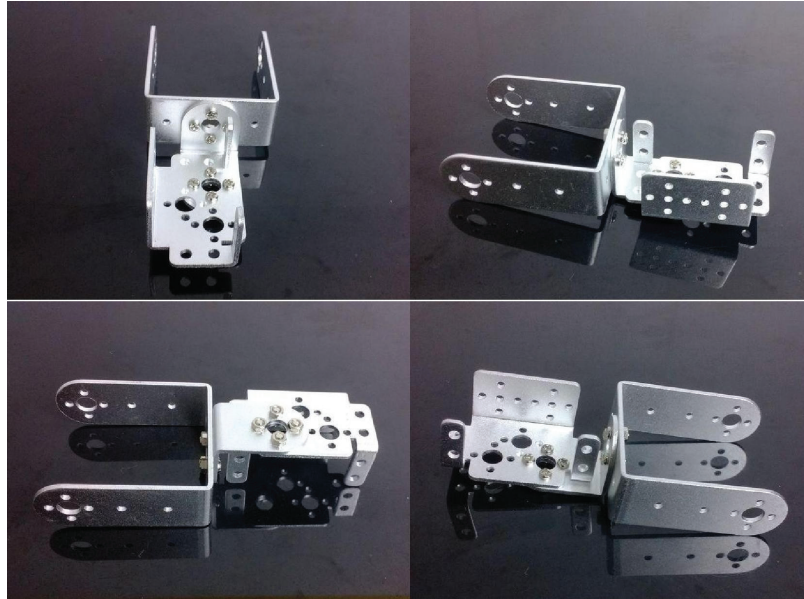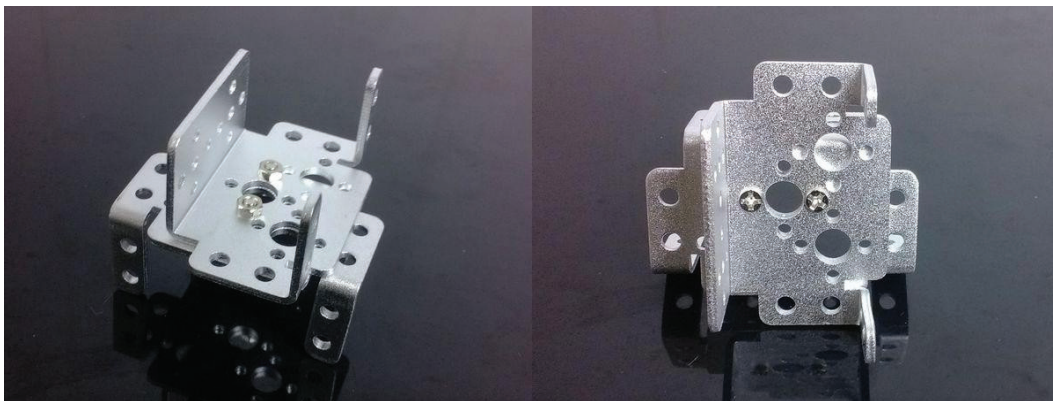

Step 4: Installing The Manipulator Rotational Structure

Required:
- 2 x Multifunction bracket
- 2 x flat head M3 x 8 screw
- 2 x M3 Nut

Above you used 2 multi-function bracket and the remaining 2 is used in this step. The two multi-function brackets are affixed together as shown in the figure below using Flat head M3 x 8 nuts and M3 screws..

Step 5: Installing the Claw Structure

Required:
- 1 x Servo disc
- 1 x Mechanical Claw
- 2 x Round M3x5 screws.

In this step you are going to install the claw structure. The claw given in the package does not have screws on one of it's sides. You can see in the picture below one circle on the claw has screws and the other does not. The reason being is so you can mount the servo motor to the claw. Now affix the servo disc to the claw, as shown in the picture below.
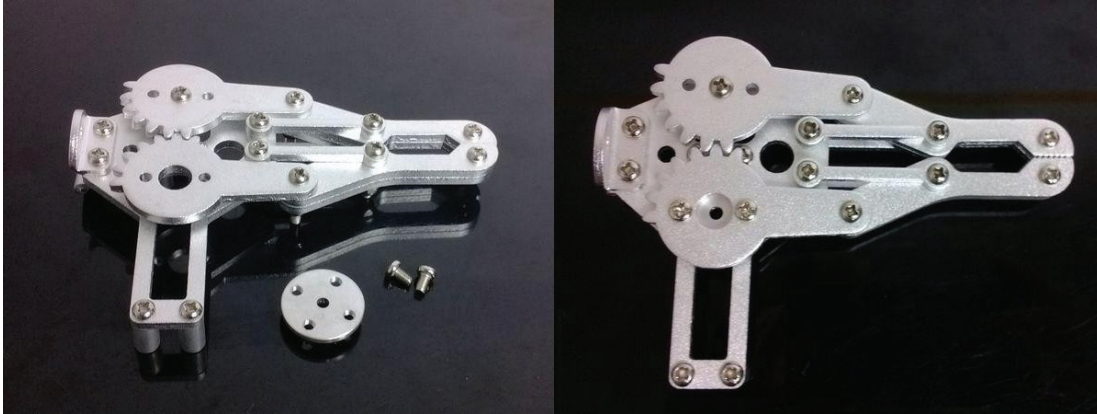
Installing The Mechanical Hand Joints & Servo Motors

All the servo motors are to be adjusted to its middle position before installing. Additionally, you can use a servo tester and PC software to adjust to the middle position value 1500.

Step 1: Installing Servo motor to the Manipulator Rotary Joints

Required:
- 1 x Round Disc Cover
- 1 x Servo Motor
- 4 x Round M4x8 screws
- 4 x M4 Nuts

In this step, servo motor is installed to the Round disc cover as shown in the figure below.
Use Round M4x8 screws and nuts for mounting.
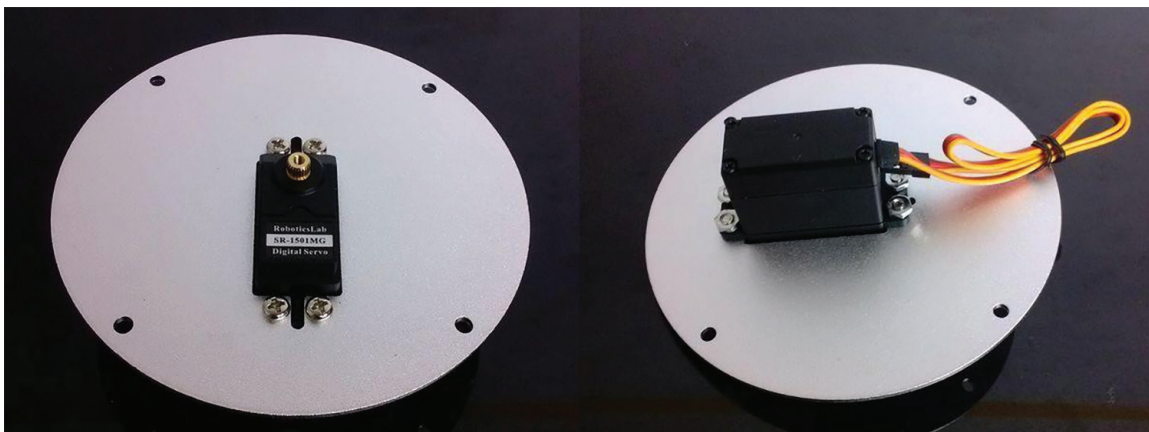
Step 2: Install The Long Arm Joints Of The Robot

Required:
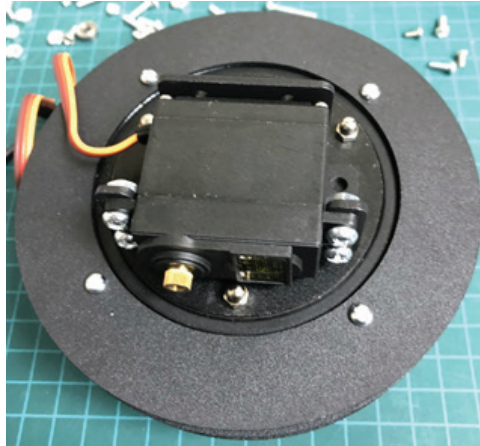- 1 x Mounted UP cover structure
- 1 x Servo motor
- 1 x Round M3x10 screws
- 4 x Round M4x8 screws
- 4 x M4 nut

Here you are going to mount one servo motor to the rotative holder you assembled in Step 1: Installing Mechanical Parts. Please follow the diagram shown below.

Step 3: Installing Mechanical Short Arm Joints

Required:
- 1 x Mounted short arm structure
- 1 x Servo motors
- 1 x Round M3x10 screws
- 4 x Round M4x8 screws
- 4 x M4 Nut

Affix the motor to the before fixed joints that you assembled in Step 2: Installing The Long Arm Of The Robot Structure. Use M3x10 and M4x8 screws.
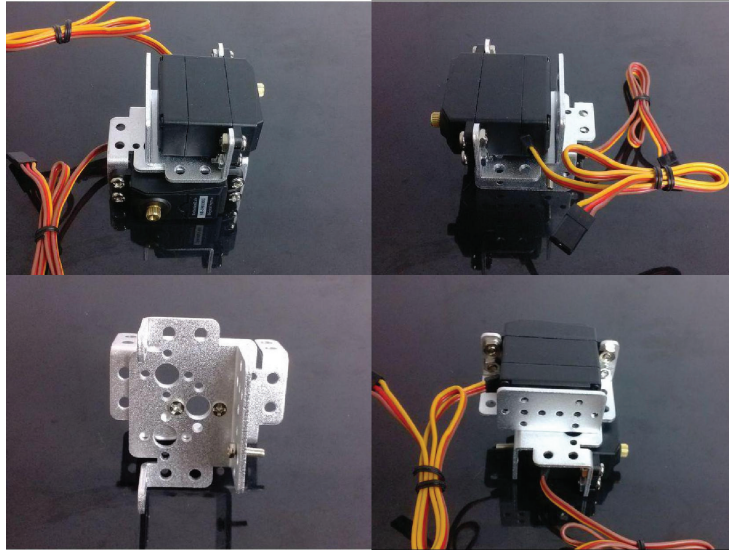
Step 4: Installing Manipulator Rotary Joints

Required:
- 1 x Mounted rotating structure
- 2 x Servo
- 1 x Round M3x10 screws
- 8 x Round M4x8 screws
- 8 x M4 Nut

Affix the motor to the mounted rotating structure you assembled in Step 4: Installing The Manipulator Rotational Structure. Use M3x10 and M4x8 screws.
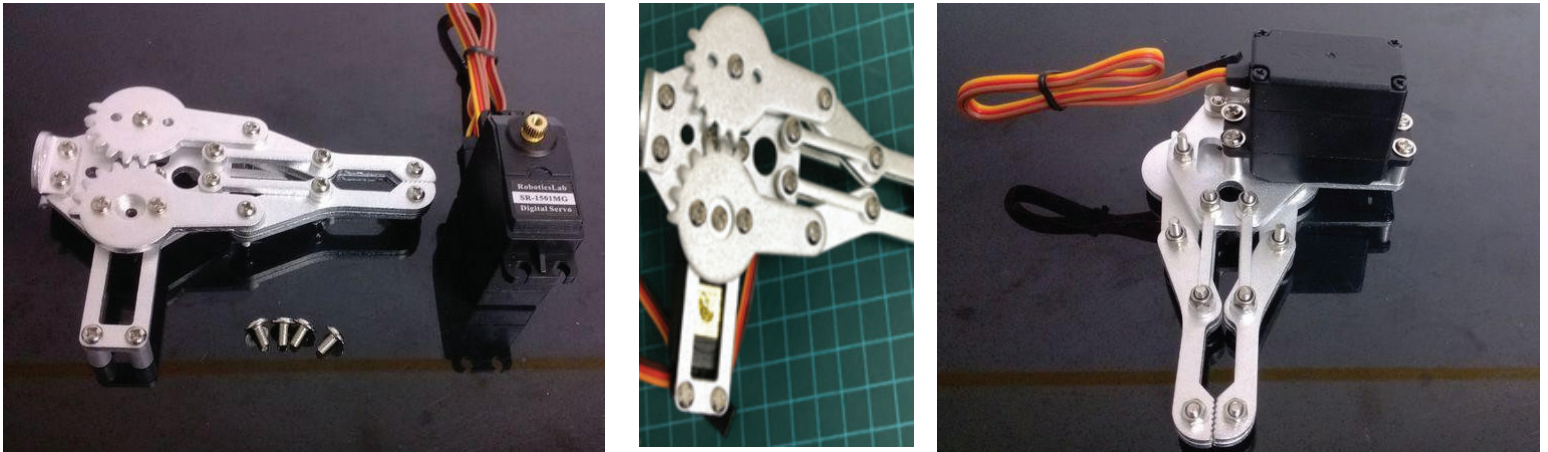

Step 5: Installing claw joint

Required:
- 1 x Mounting claw structure
- 1 x Servo motor
- 4 x M3x6 screws
- 1 x Round M3x6 Screw

Now take the claw joint that you assembled in Step 5: Installing the Claw structure, and affix the servo motor as shown in the figure below. You can use M3x6 screws for it. You must affix it to the servo disc and mounts on the claw joint.

Other Parts for the Claw

In these step you are going to join all the parts of the arm you assembled.

Step 1: Mechanical Arm disk

Mount all base parts as shown in the below figure using above assembled parts.
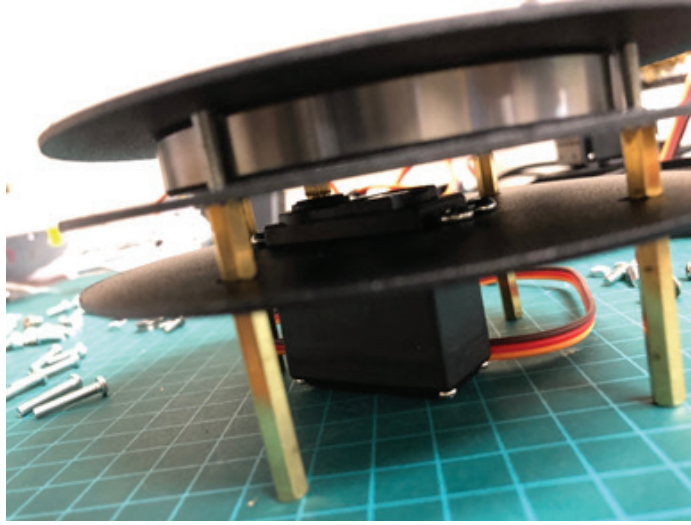
Step 2: Mechanical Arm Claw

Required:
- 1 x mounted rotary joints
- 1 x mechanical claws
- 2 x servo disc
- 3 x Round M3x6 screw

Take the mounted rotary joint and affix the claw that has the servo motor already installed to it. Join the parts by using the servo disc and M3x6 screws.

Step 3: Mechanical Short Arm

Required:
- 1 x the mounted mechanical claw on the previous step
- 1 x robot short arm joints
- 1 x Servo disc
- 4 x round M3x5 screws
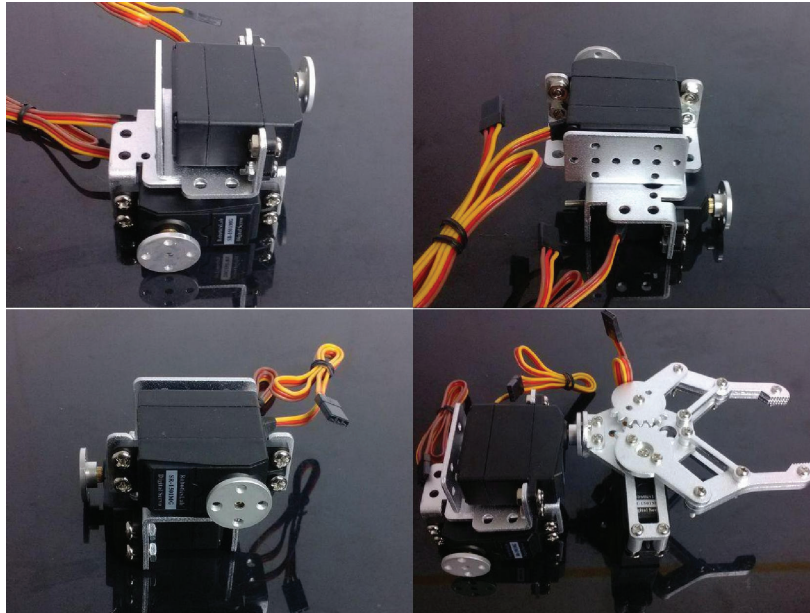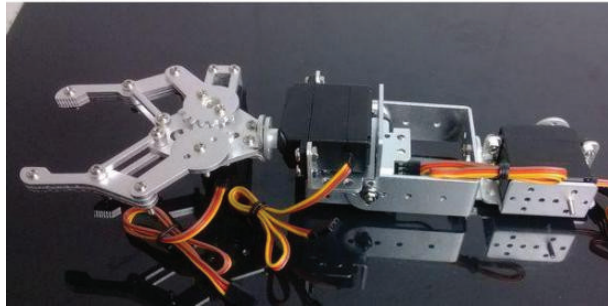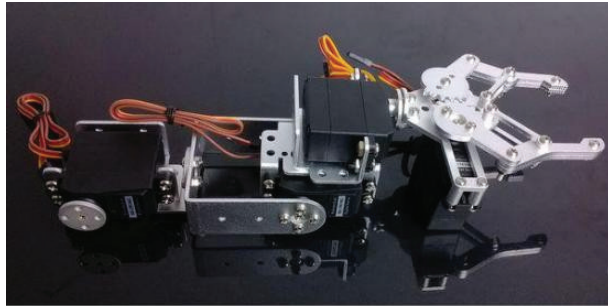- 1 x round M3x6 screw
- 1 x flange bearings
- 1 x M3 nut

Now it is time to Mount the small arm you assembled in the pervious steps. Now affix the claw from the above step to the robot short arms. They are affixed together by using the M3x5 and M3x6 screws and use the flange baring for the other side of the bracket. (Do not forget this step, otherwise your brackets will not be stable). The picture below shows the flange bearing arrangements.
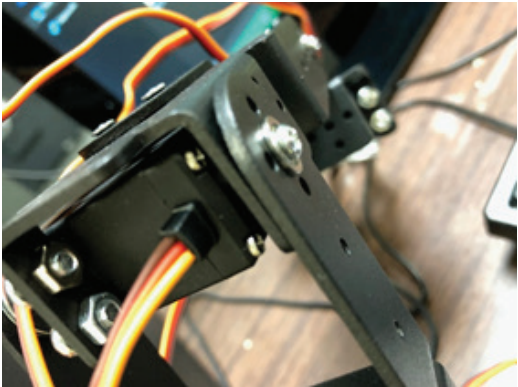
Step 4: Mechanical Long Arm

Required:
- 1 x mounted robot arm done in the previous step
- 1 x Long-arm joints
- 1 x Servo disc
- 4 x round M3x5 screw
- 1 x round M3x6 screw
- 1 x flange bearing
- 1 x M3 nut

Now attached the long U-shaped brackets which were joint togehter in the previous step, to the mechincal short arm. Use the required screws, nuts, flange bearing and servo disc mention above to attach it to the motor. The final structure should look like the image below.
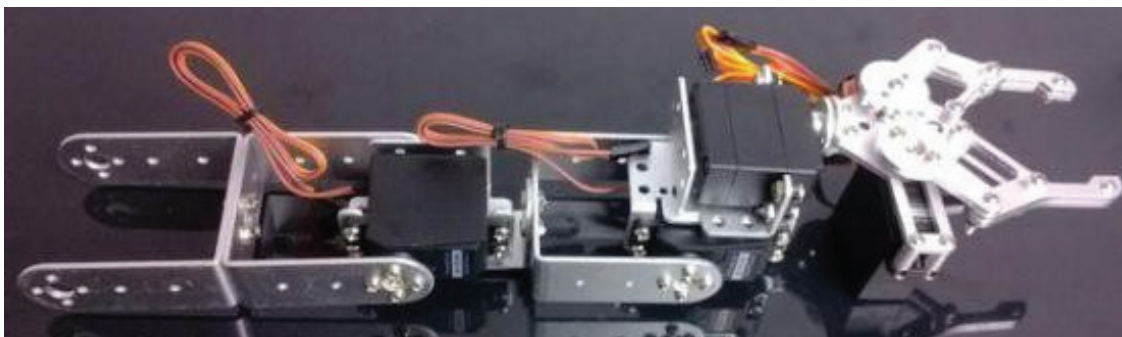
Step 5: Connecting Robot Base Disc

Required:
- 1 x Robot long arm
- 1 x mounted robot disc base
- 4 x round M3x5 screw
- 1 x round M3x6 screw
- 1 x flange bearings
- 1 x M3 Nut

Now it is time to attach the rotating base that was assembled in the previous steps. Use the servo disc and the flange bearing to affix the long U-shaped brackets from the robot long arm to the rotating base.
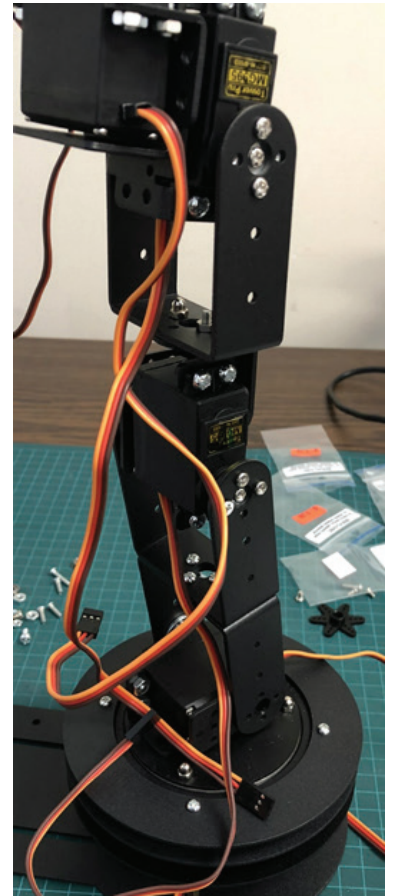
Step 6: Connect The Robot Big Base

Required:
- 1 x Mounted robot arm in the previous step
- 1 x Robot arms big base
- 4 x flat head M3x6 screw

Last step in to assembling the arm, is to affix the big base that is provided using long spacers. Below is the picture of what the arm should resemble once the big base is attached.

Now that the arm is assembled is it time do the connections and code it.

Connections & Coding

In the next steps you will learn the connections and how to make this robotic arm automatic or manual control. This arm can be used for many applications, such as controlling the arm using Bluetooth, WIFI or a automatic picking and placing robot. Since there are many applications to this arm, only one application is explained in the following steps.

Controling Robotic ARM Using A Joystick

In this application you are going learn how to connect and code a joystick to the arm so that you can control the arm using a joystick.
Furthermore, for other applications and code you can follow the GitHub link provided below.

Required Electronic Components:

Additional to the above completed robotic frame, you need some other electronics components to make it work automatically.
List for components required (Not include in the kit)
  • One Arduino NANO (ABRA Part No: ABRANANO)
  • Three Joysticks (ABRA Part No: AM-JOYSTICK)
  • Connecting jumper wires (Male-female and Male-Male)
  • Breadboard (ABRA Part No: ABRA-6)
  • I2C 16 channel PWM servo motor controller (ABRA Part no: MOT-PCA9685-A)

Connection Diagram
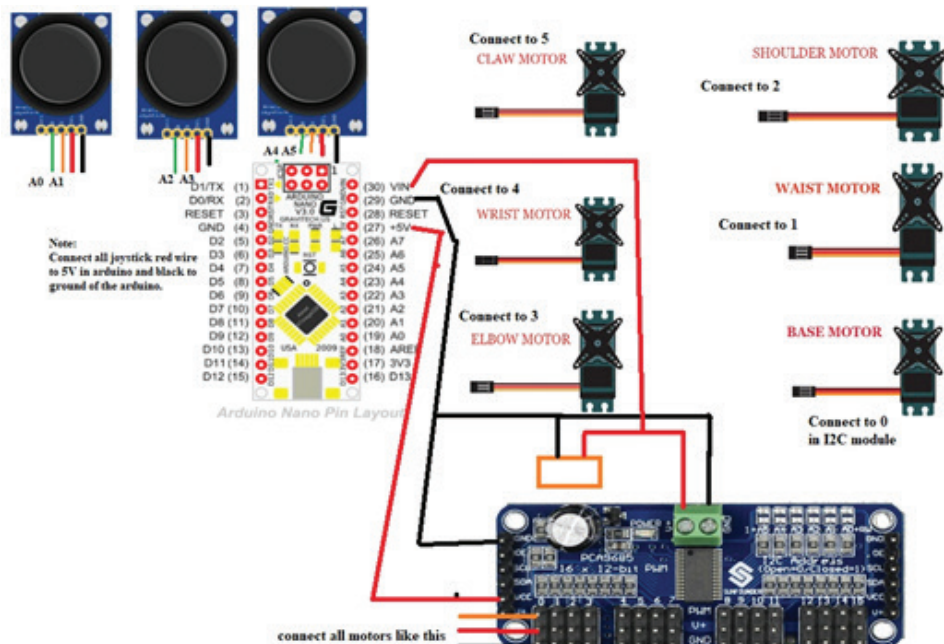
NOTE: Use an external power supply for servo motors

Source Code

After completing the connections as explained above, copy this code to the ARDUINO IDE and upload to the Arduino NANO board. Please look at the below link for ARDUINO IDE installation and upload details.

Link1: https://startingelectronics.org/software/arduino/installing-arduino-software-windows-10/

Link 2: https://www.arduino.cc/en/Guide/ArduinoNano

Here is the code:

```
/************************************************
 //ABRA ELECTRONICS
  //6DOF ARM ROBOTIC
 *************************************************/
#include "HCPCA9685.h"
#include <Wire.h>
#include <Adafruit_PWMServoDriver.h>
#include <Servo.h>
// called this way, it uses the default address 0x40
Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver();
// you can also call it with a different address you want
//Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(0x41);
// you can also call it with a different address and I2C interface
//Adafruit_PWMServoDriver pwm = Adafruit_PWMServoDriver(&Wire, 0x40);

// Depending on your servo make, the pulse width min and max may vary, you
// want these to be as small/large as possible without hitting the hard stop
// for max range. You'll have to tweak them as necessary to match the servos you
// have!
#define SERVOMIN  150 // this is the 'minimum' pulse length count (out of 4096)
#define SERVOMAX  600 // this is the 'maximum' pulse length count (out of 4096)
#define JoyX1 A1 // Joystick X1 pin connected to A0 on the UNO
#define JoyY1 A0 // Joystick Y1 pin connected to A1 on the UNO
#define JoyX2 A2 // Joystick X1 pin connected to A0 on the UNO
#define JoyY2 A3 // Joystick Y1 pin connected to A1 on the UNO
#define JoyX2 A6 // Joystick X1 pin connected to A0 on the UNO
#define JoyY2 A7 // Joystick Y1 pin connected to A1 on the UNO
#define I2CAdd 0x40
HCPCA9685 HCPCA9685(I2CAdd);
// our servo # counter

uint8_t servonum0 = 0;
uint8_t servonum1 = 1;
uint8_t servonum2 = 2;
uint8_t servonum3 = 3;
uint8_t servonum4 = 4;
uint8_t servonum5 = 5;

Servo servo,servo1,servo2,servo3,servo4,servo5;
void setup() {


servo.attach(servonum0);
servo1.attach(servonum1);
servo2.attach(servonum2);
servo3.attach(servonum3);
servo4.attach(servonum4);
servo5.attach(servonum5);

pwm.begin();

pwm.setPWMFreq(30);  // Analog servos run at ~60 Hz updates

delay(10);
}

// you can use this function if you'd like to set the pulse length in seconds
// e.g. setServoPulse(0, 0.001) is a ~1 millisecond pulse width. its not precise!
void setServoPulse(uint8_t n, double pulse) {
double pulselength;

  pulselength = 1000000;   // 1,000,000 us per second
```

```
  pulselength /= 30;   // 60 Hz

  pulselength /= 4096;  // 12 bits of resolution

  pulse *= 1000000;  // convert to us
  pulse /= pulselength;

  pwm.setPWM(n, 0, pulse);
}

void loop()
{


  int analogValue1 = analogRead(A0); // read the analog input

  int analogValue2 = analogRead(A1); // read the analog input

  int analogValue3 = analogRead(A2); // read the analog input

  int analogValue4 = analogRead(A3); // read the analog input

  int analogValue5 = analogRead(A6); // read the analog input

  int analogValue6 = analogRead(A7); // read the analog input

  // if your sensor's range is less than 0 to 1023, you'll need to
  // modify the map() function to use the values you discovered:
  // move the servo using the angle from the sensor:

servonum0 = map(analogValue1, 0, 1023, 0, 255);

 delay(100);

servonum1 = map(analogValue2, 0, 1023, 0, 255);

  delay(100);
servonum2 = map(analogValue3, 0, 1023, 0,130);

   delay(100);
servonum3 = map(analogValue4, 0, 1023,0,255);

   delay(100);

servonum4 = map(analogValue5, 0, 1023, 0, 255);

   delay(100);
servonum5 = map(analogValue6, 0, 1023, 50, 150 );
delay(100);

HCPCA9685.Servo(0, servonum0);
HCPCA9685.Servo(1, servonum1);
HCPCA9685.Servo(2, servonum2);
HCPCA9685.Servo(3, servonum3);
HCPCA9685.Servo(4, servonum4);
HCPCA9685.Servo(5, servonum5);

servo.write(servonum0);
servo1.write(servonum1);
servo2.write(servonum2);
servo3.write(servonum3);
servo4.write(servonum4);
servo5.write(servonum5);
delay(10);
}
```

OTHER APPLICATION PROJECT LINKS:
Robotic Arm using Bluetooth and android device
Robotic arm using Serial monitor.
(for the code and connections of this robotic, please go to our website and look for AK-ARM. You will find updated link of GitHub)