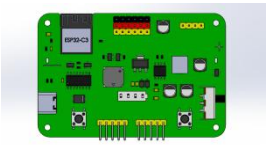
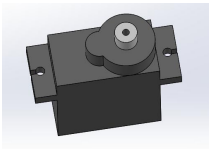
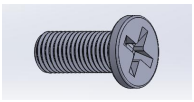
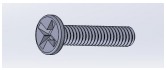
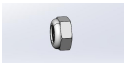
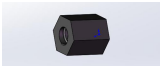
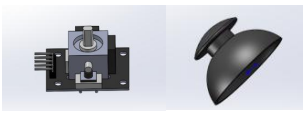
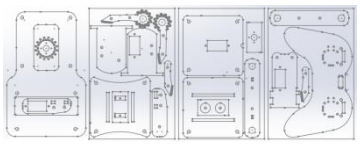
Table of Contents

1. Introduction to eArm

SIYEENOVE eArm robot is an ESP32 based robotic arm kit designed around maker education and programming learning.

Features:
-The eArm robot is very simple and easy to build. You can assemble it and start playing with it in two or three hours.
-We have pre-burned the program on the control board, so you can run it without uploading the code.
-The robot arm has an onboard battery holder, a power indicator, and you can directly use the USB cable to charge the battery on the robot.
-The joystick handle provides a stable connection and easy control method, and it is very convenient and flexible to control the robot arm.
-You don't need to prepare additional WiFi or upload code, just press the right button of the joystick to switch between Web APP and joystick mode.
-Connect your phone to the WiFi named eArm with one click and enter the IP address we provide in the browser, then you can connect to eArm and remotely control it.
-In the last section of this document, it is an introduction to Arduino programming and examples, which allows you to gain a deeper understanding of how Arduino controls robots..

1.1 Product List

Esp32 Control Board 1PCS	MG90s Servo 4PCS	M4x10mm Screw 20PCS
		
M3x14mm Screw 2PCS	M3x10mm Screw 18PCS	M2x10mm Screw 8PCS
		
M1.4x5mm Screw 6PCS	M4 Nat 4PCS	M3 Self-locking Nat 8PCS
		
M2 Nut 8PCS	M3x7mm Nylon Column 4PCS	M3x28mm Nylon Column 4PCS
		
M3x40mm Nylon Column 2PCS	M4x20mm Nylon Column 8PCS	M3x6mm Nylon Screw 8PCS
		
Flange Bearing 4PCS	Turntable 3PCS	5P Dupont Wire 2PCS
		
Joystick 2PCS	USB Cable 1PCS	A Set of Acrylic Sheet
		
M3 Screwdriver 1PCS	M2.0 Screwdriver 1PCS	Wrench 1PCS
		

Note：You need to buy a 18650 lithium battery yourself!

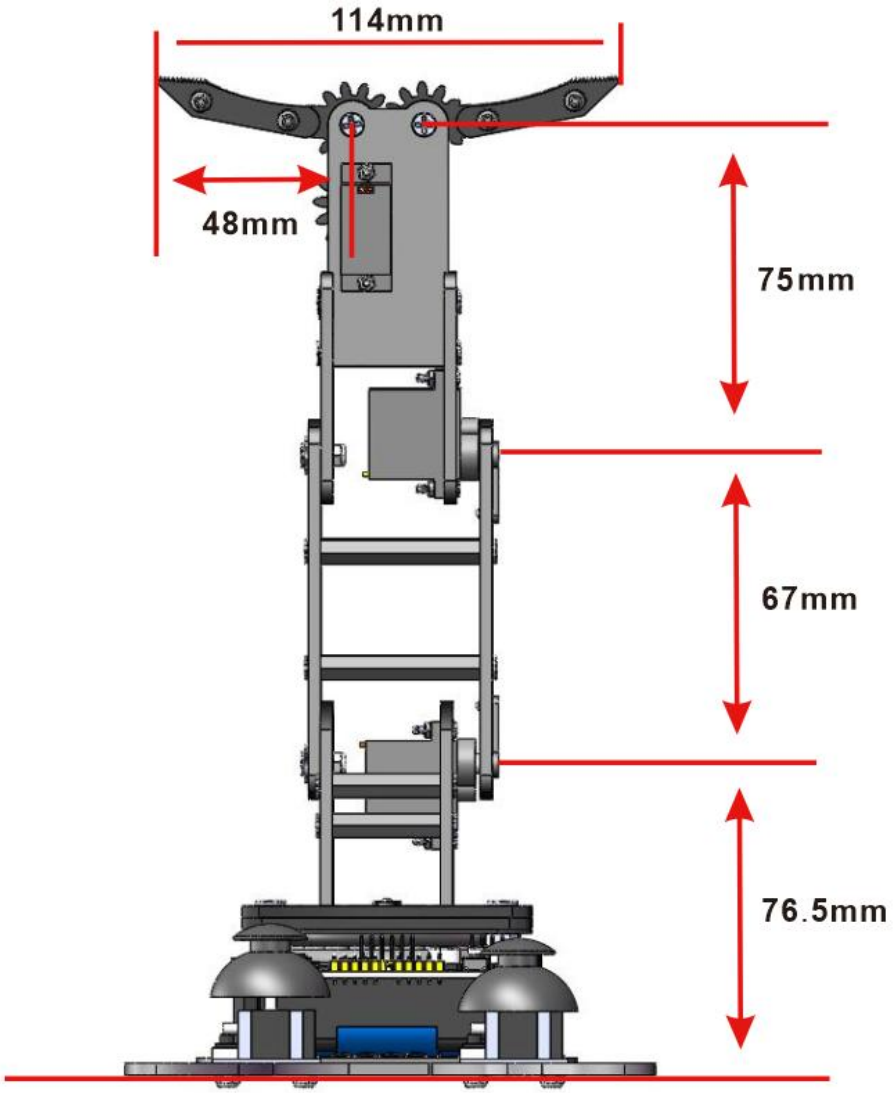
Parameters of 18650 lithium battery

- Model: 18650 lithium battery
- Top type: Both button top and flat top are acceptable
- Capacity: >2000mAh
- Maximum charging voltage: 4.2V
- Nominal voltage: 3.7V
- End-off voltage: 2.75V
- Minimum charging current: >2A
- Minimum discharge current: >4A

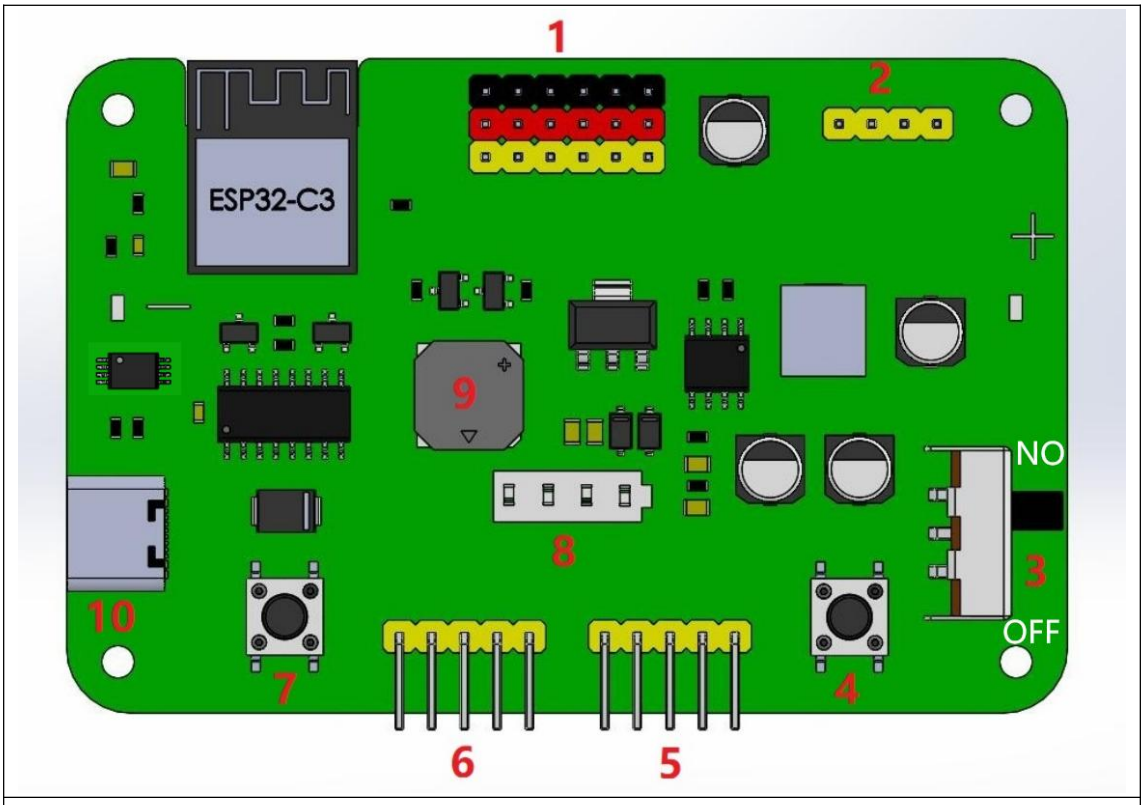
5

1.2 Parameters:

Control Board	eArm-ESP32-C3-MINI-1
Programming language	Arduino
Charge/discharge	5V/2A
Servo type	MG90S 180°
Remote control method	Joystick or Web APP
Assembly time required	1-2 hours
Product size	15X9.5X26MM
Packaging size	18.5X12.5X4.3MM
Applicable age	for experienced players of any age or beginners aged 12+
Battery required	a 18650 battery(Not included)



1.3 ESP32 Control Baord



1: IO port, 5V/2A driving capability.

2: IO port and serial port expansion port.

3: Power switch.

4: Battery display button.

5: IO port, 3.3V/500mA driving capability.

6: IO port, 3.3V/500mA driving capability.

7: Reset button.

8: Battery level indicator LED.

9: Buzzer.

10: Type-c USB Port for charging or uploading code

Notes：

3：When the power switch is in the "ON" state, if the control board is connected to the 18650 lithium battery, the control board will work normally, and the lithium battery cannot be charged through the USB port. When the power switch is in the "OFF" state, turn off the power of the control board and charge the lithium battery through the USB port.

10：The Type-c USB port only charges the battery or uploads the code, and does not power the 5V pin.

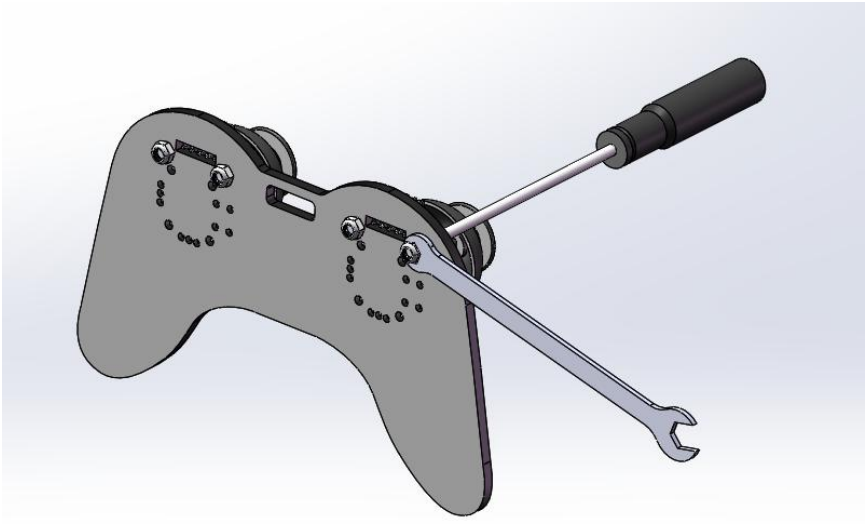
7

2. Assemble the eArm

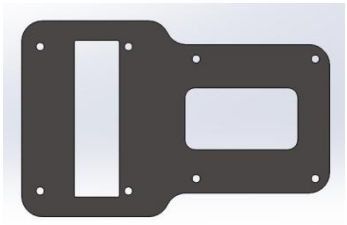
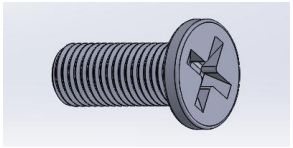
Notes

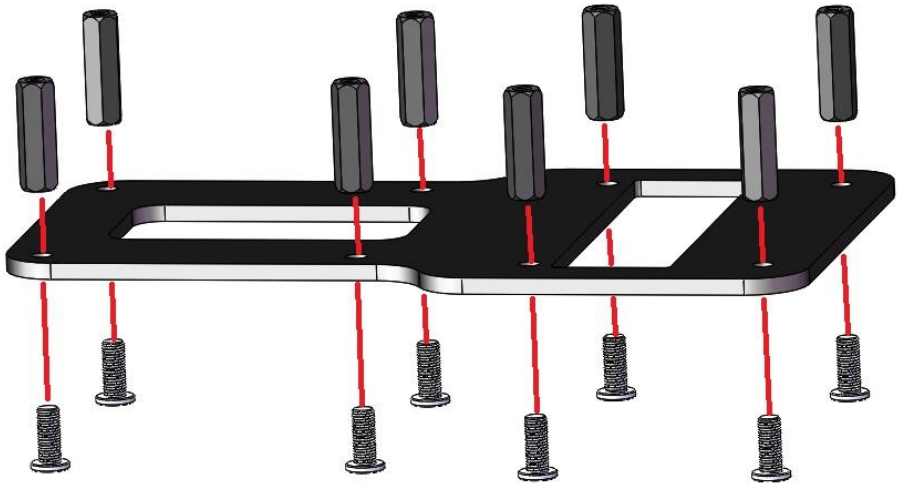
The surface of the acrylic board is covered with a protective film, please tear it off before installation!

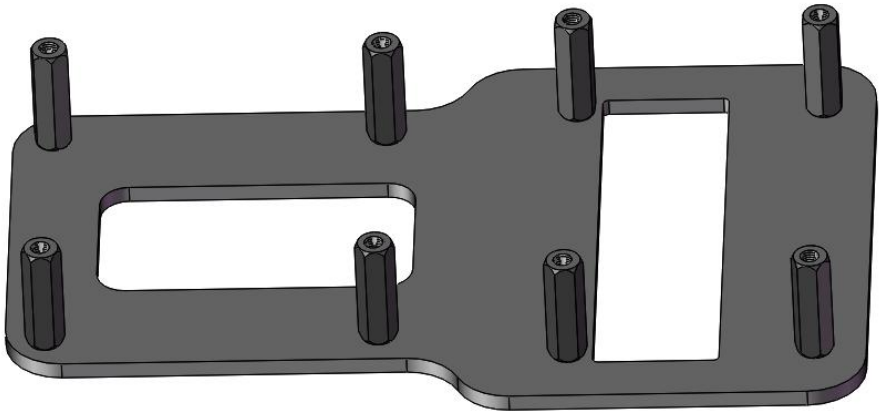
Self-locking nuts are difficult to tighten, please use a screwdriver and wrench to tighten!



Step 1 Robot Base

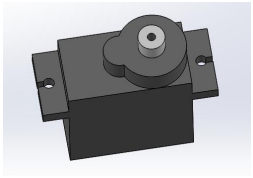
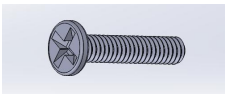
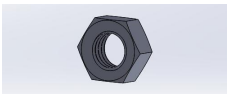
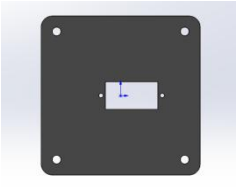
Acrylic Sheet 1PCS	M4X20MM Nylon Column 8PCS	M4X10MM Screw 8PCS
		

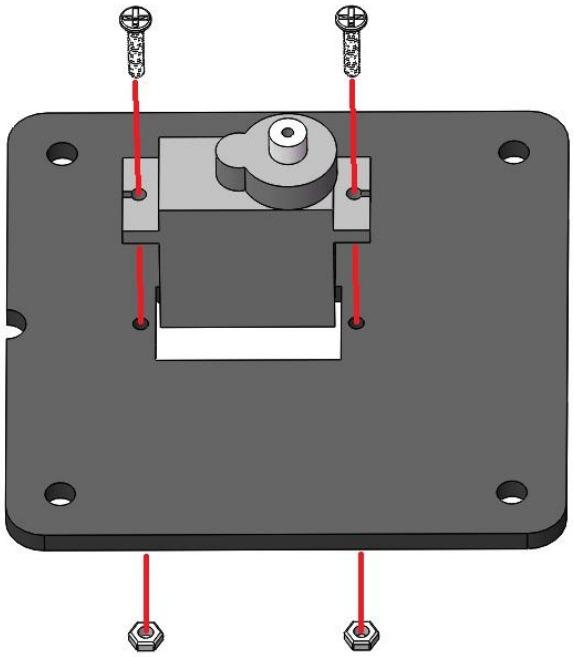


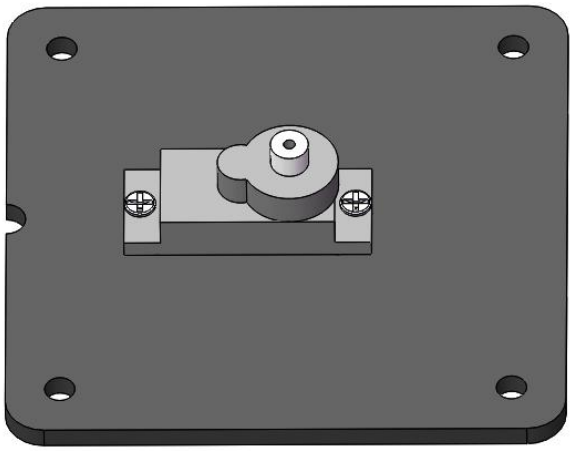


9

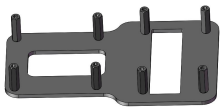
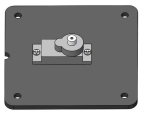
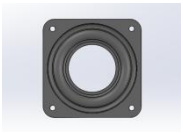
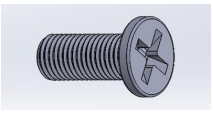
Step 2 Servo A on robot base

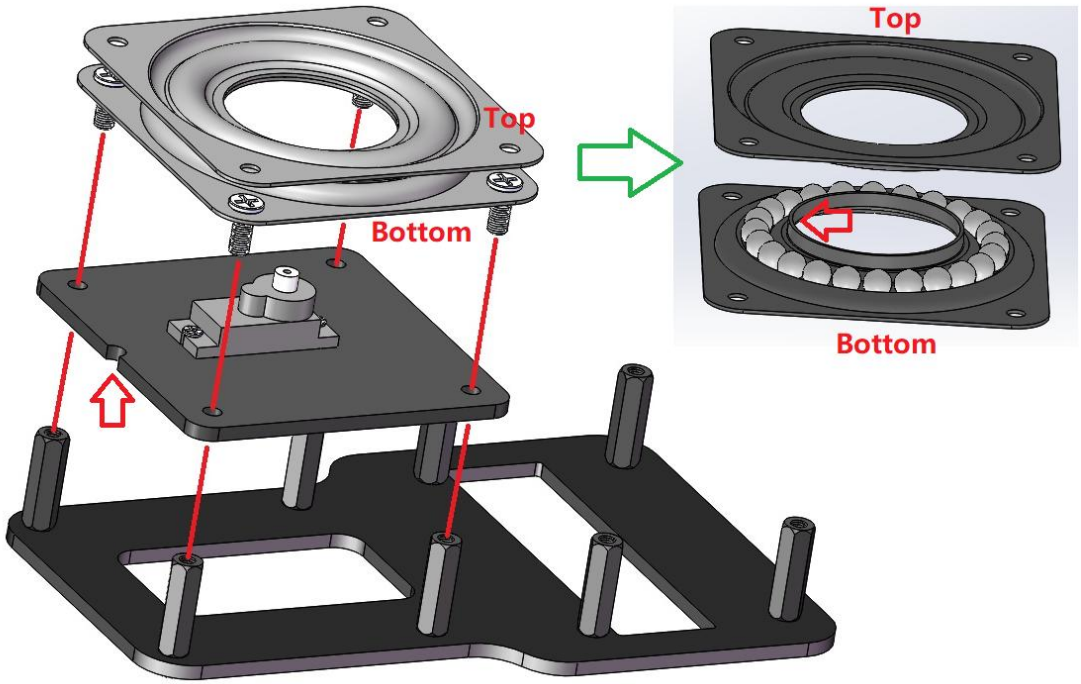
Servo 1PCS	M2x10MM Screw 2PCS	M2 Nut 2PCS	Acrylic Sheet 1PCS
			

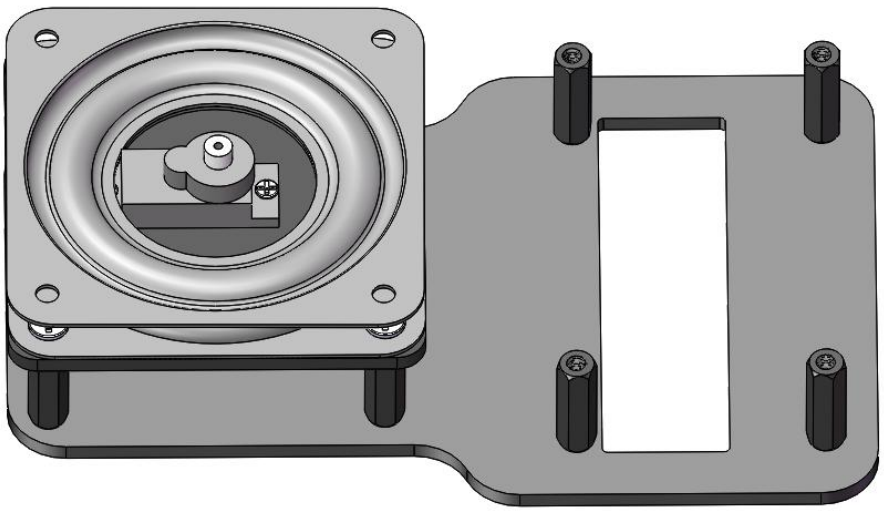




Step 3 Turntable

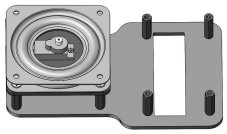
Part in Step 1	Part in Step 2	Turntable 1PCS	M4X10MM Screw 4PCS
			

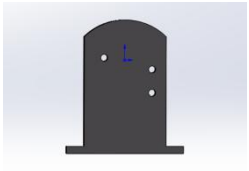
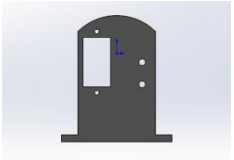
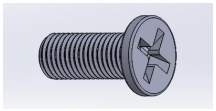
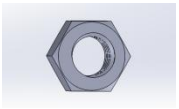


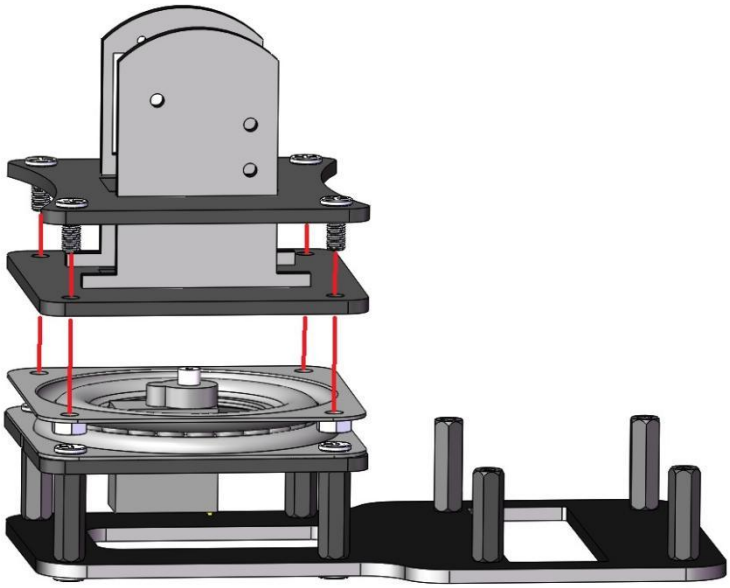


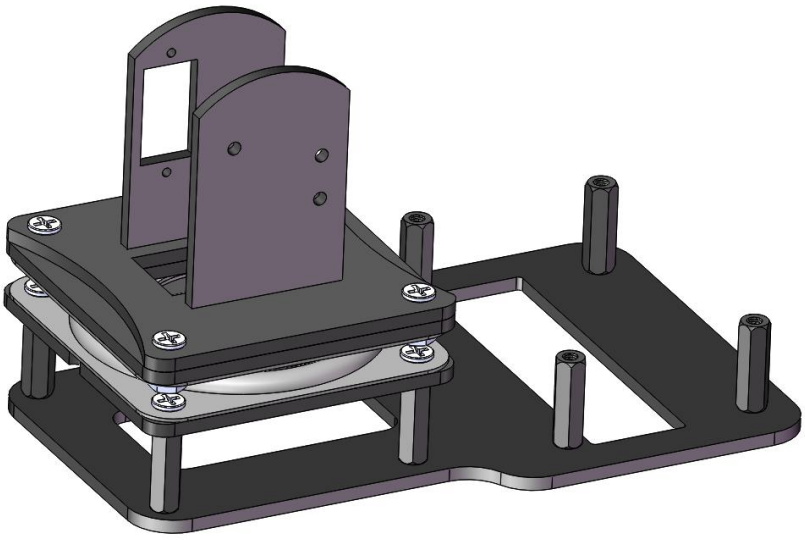
11

Step 4 Lower arm

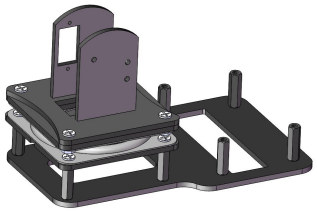
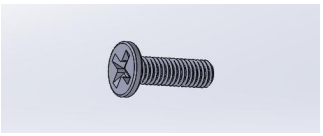
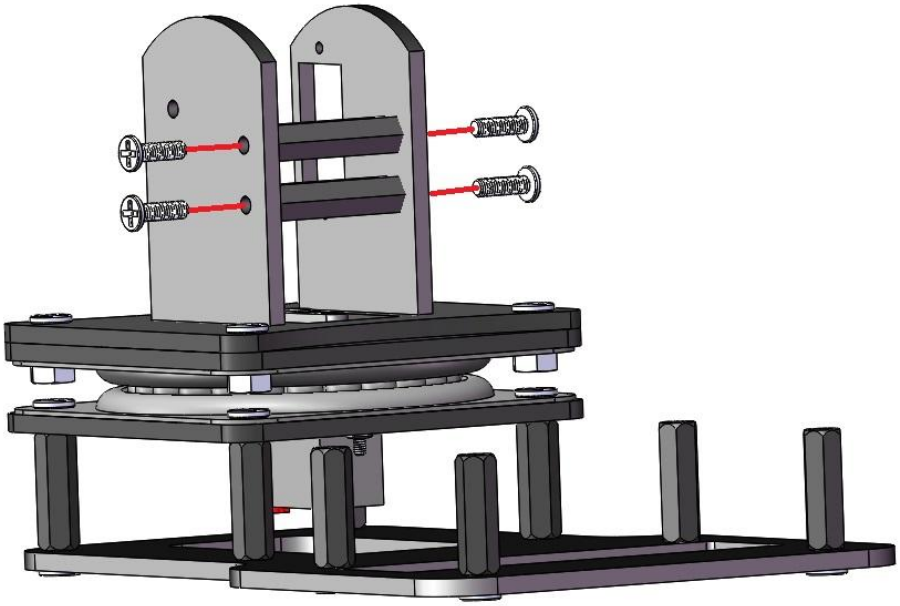
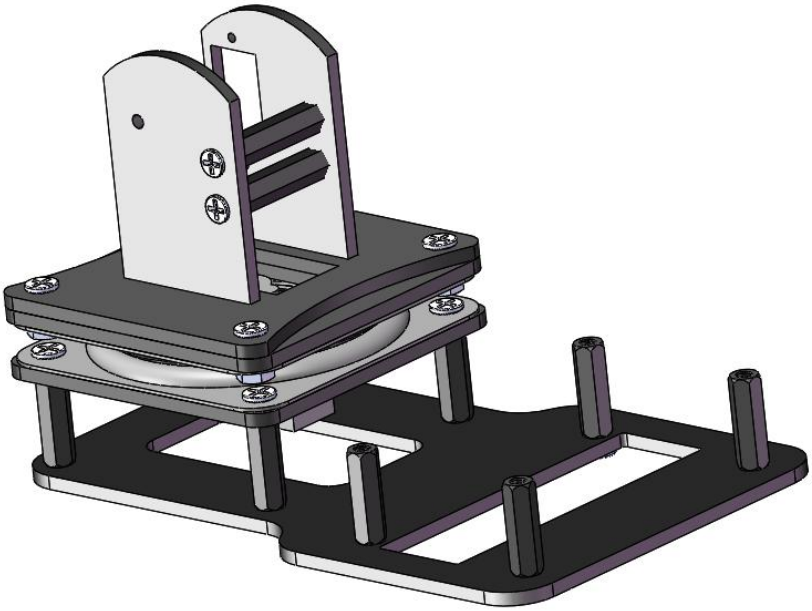
Part in Step 3	Acrylic Sheet 1PCS	Acrylic Sheet 1PCS
		

Acrylic Sheet 1PCS	Acrylic Sheet 1PCS	M4X10MM Screw 4PCS	M4 Nut 4PCS
			



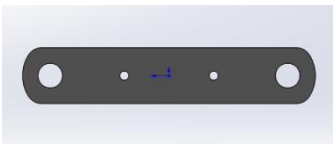
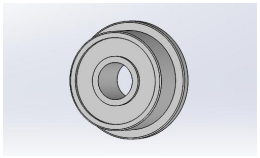
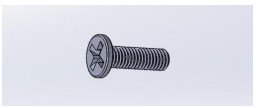
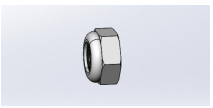
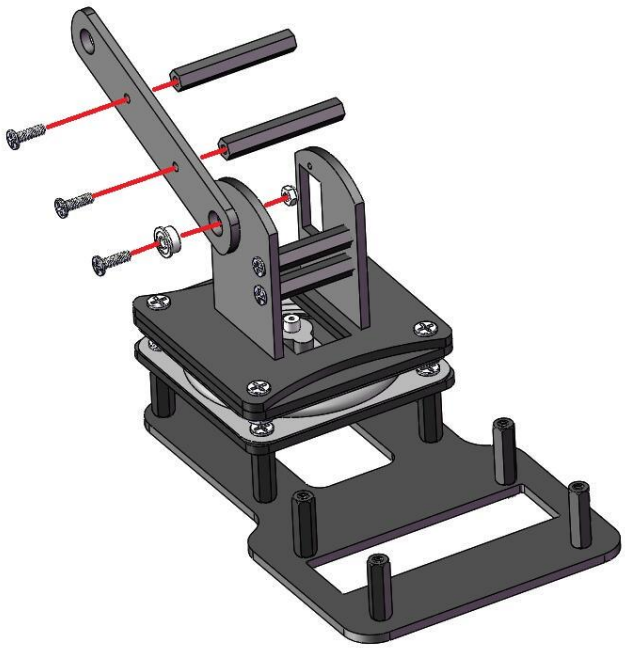


Step 5 Fix the lower arm

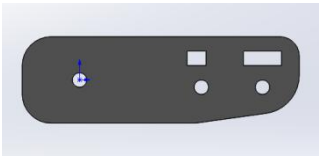
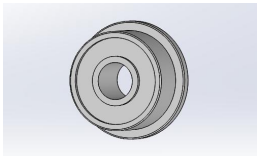
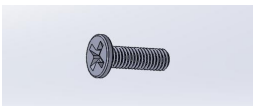
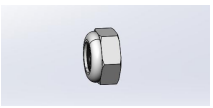
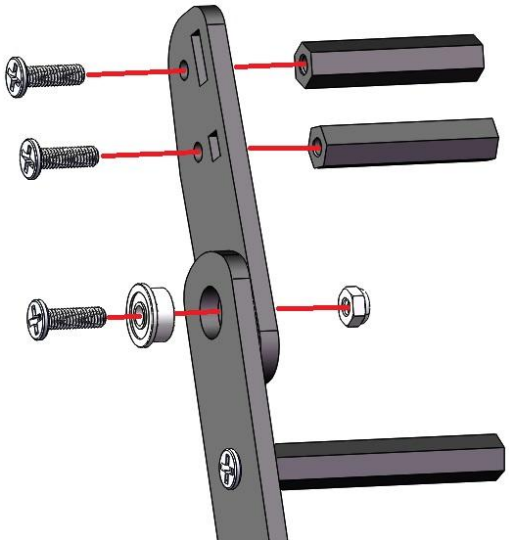
Part in Step 4	M3x28MM Nylon Column 2PCS	M3X10MM Screw 4PCS
		
		
		

13

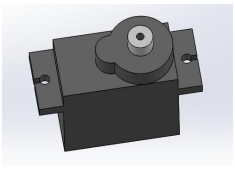
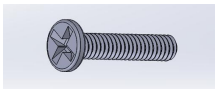
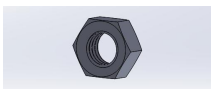
Step 6 Middle arm

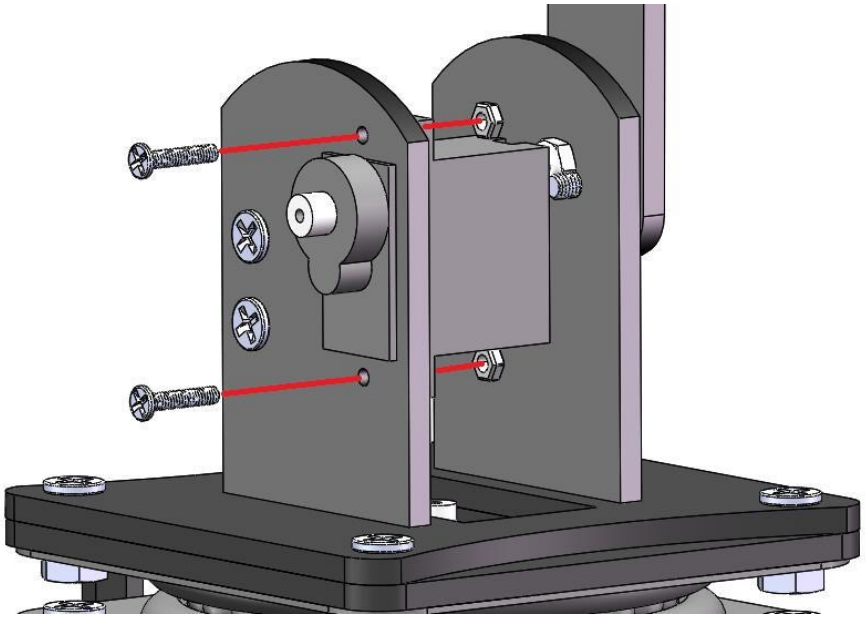
Part in Step 5	Acrylic Sheet 1PCS	Flange Bearing 1PCS
		
M3x40MM Nylon Column 2PCS	M3X10MM Screw 3PCS	M3 Self-locking Nut 1PCS
		
		
		

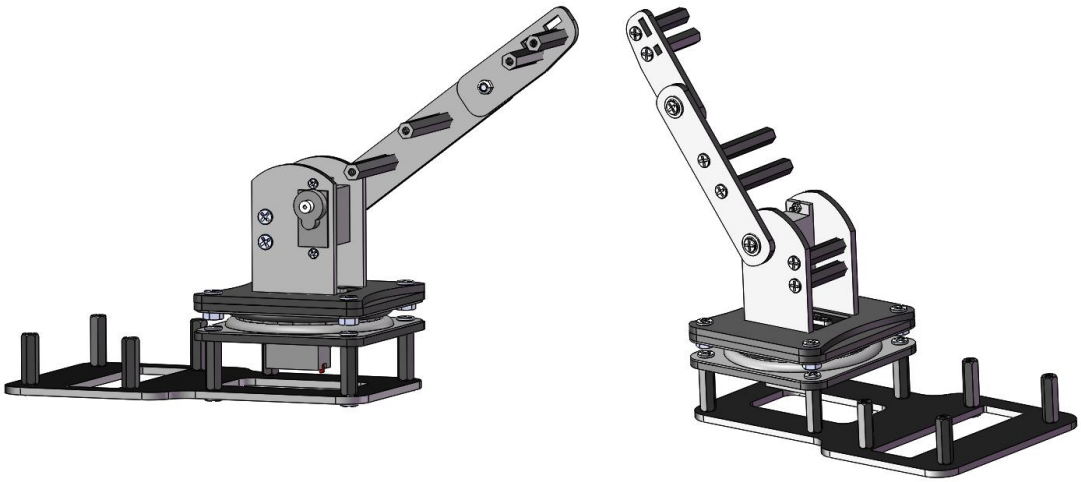
Step 7 Upper arm

Part in Step 6	Acrylic Sheet 1PCS	Flange Bearing 1PCS
		
M3x28MM Nylon Column 2PCS	M3X10MM Screw 3PCS	M3 Self-locking Nut 1PCS
		
		
		

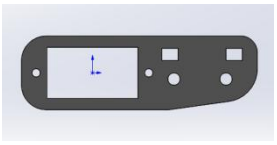
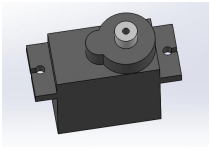
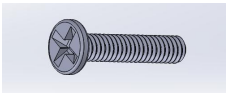
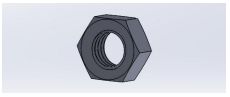
Step 8 Servo B on lower arm

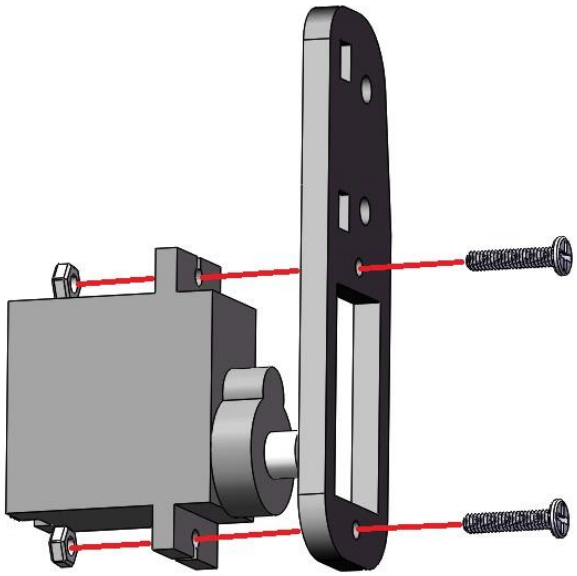
Part in Step 7	Servo 1PCS	M2x10MM Screw 2PCS	M2 Nut 2PCS
			

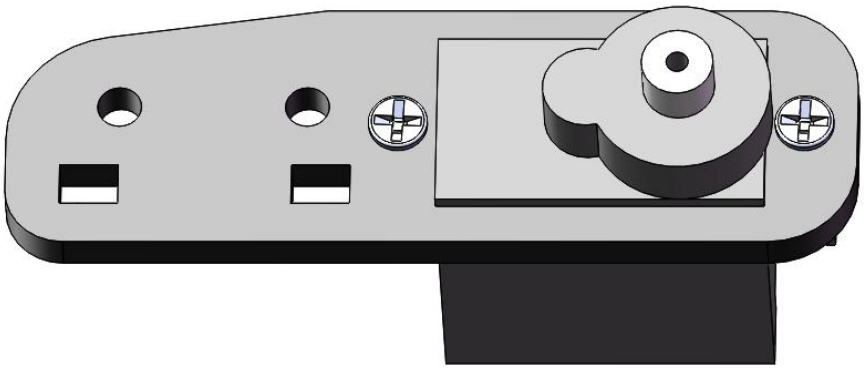




Step 9 Servo C on middle arm

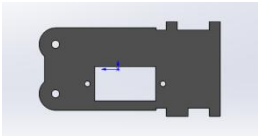
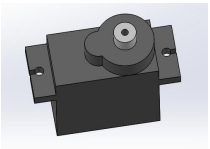
Acrylic Sheet 1PCS	Servo 1PCS	M2x10MM Screw 2PCS	M2 Nut 2PCS
			

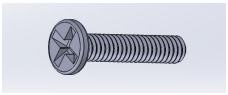
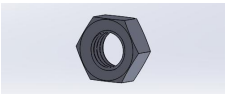


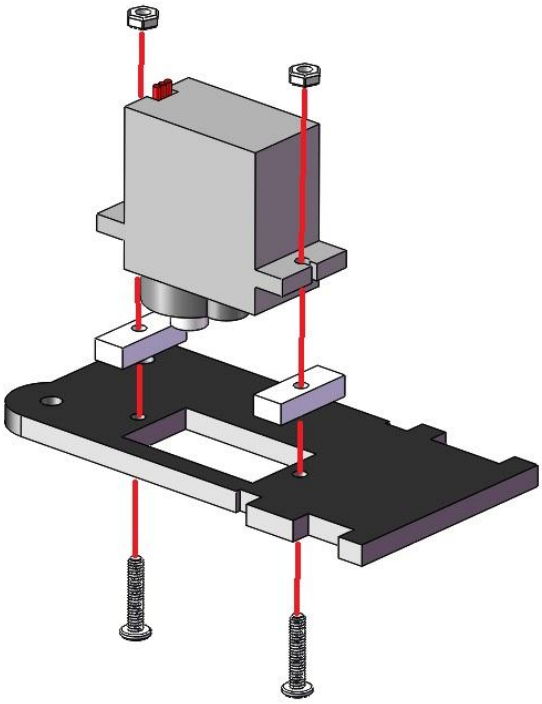


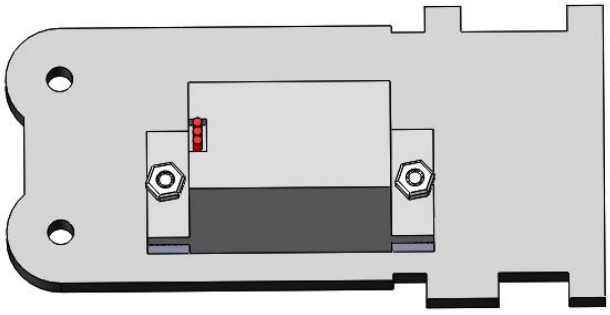
17

Step 10 Servo D on upper arm

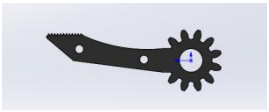
Acrylic Sheet 1PCS	Servo 1PCS
	

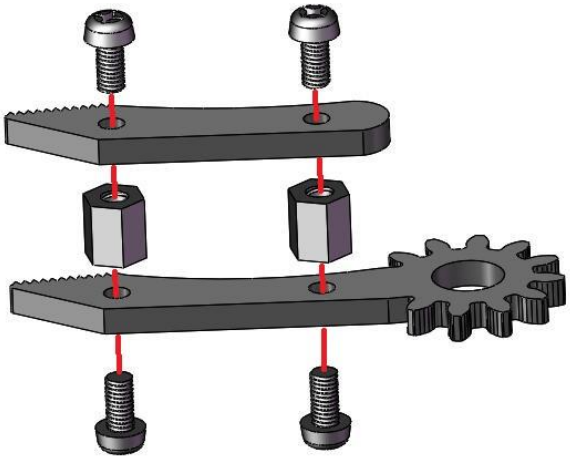
Acrylic Sheet 2PCS	M2x10MM Screw 2PCS	M2 Nut 2PCS
		

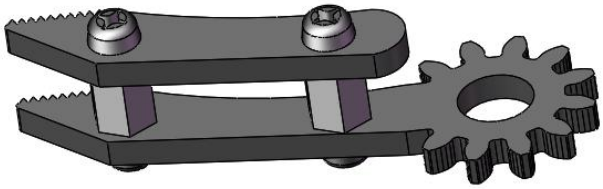




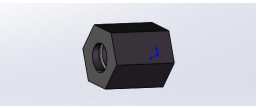
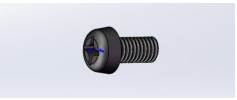
Step 11 Left claw

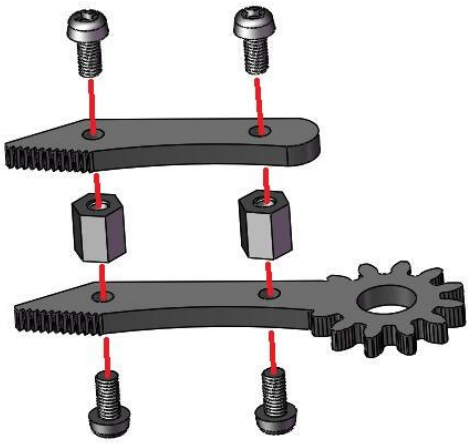
Acrylic Sheet 1PCS	Acrylic Sheet 1PCS	M3x7MM Nylon Column 2PCS	M3x6MM Nylon Screw 4PCS
			

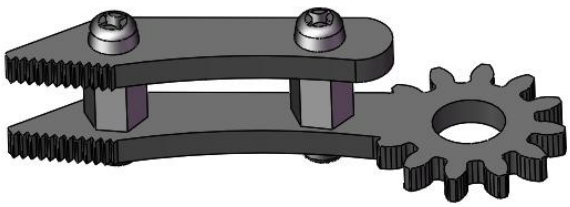




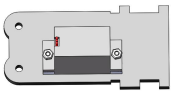
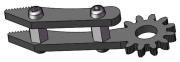
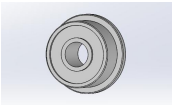
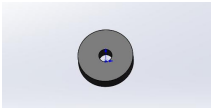
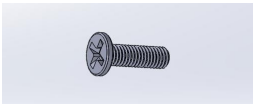
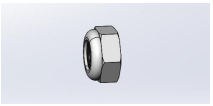
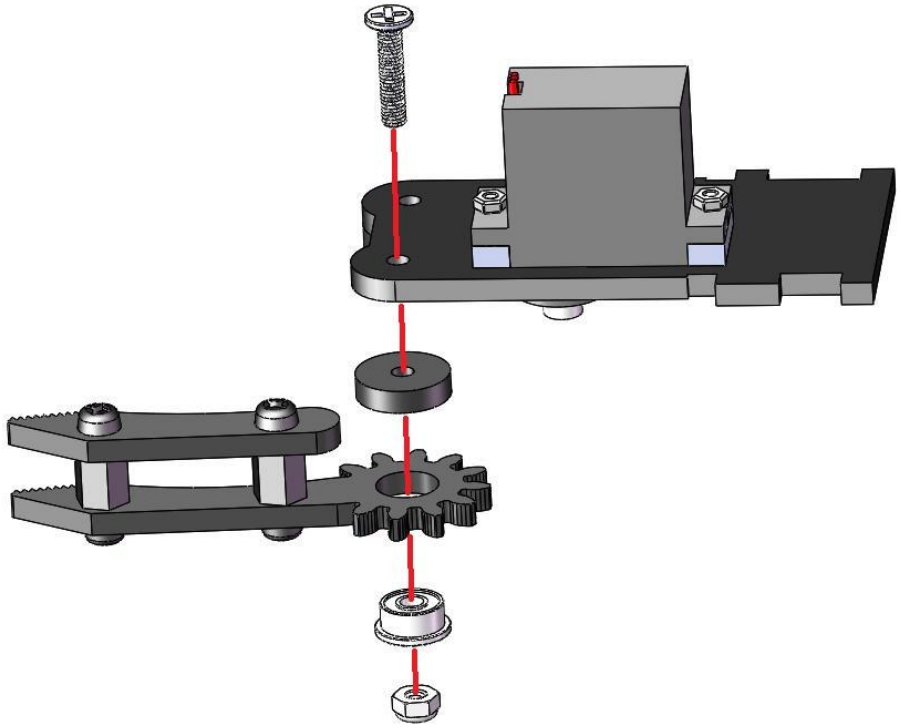
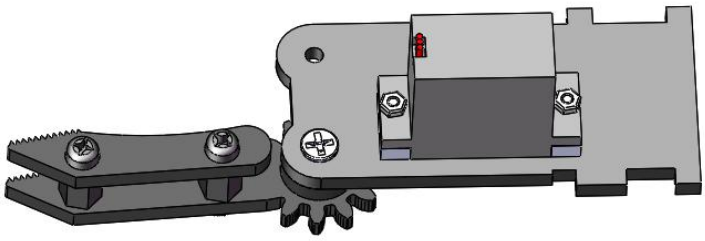
Step 12 Right claw

Acrylic Sheet 1PCS	Acrylic Sheet 1PCS	M3x7MM Nylon Column 2PCS	M3x6MM Nylon Screw 4PCS
			



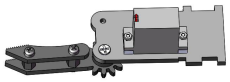
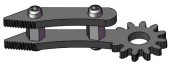
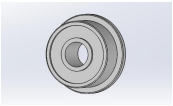
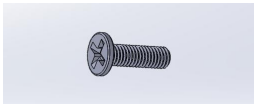
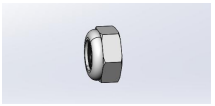
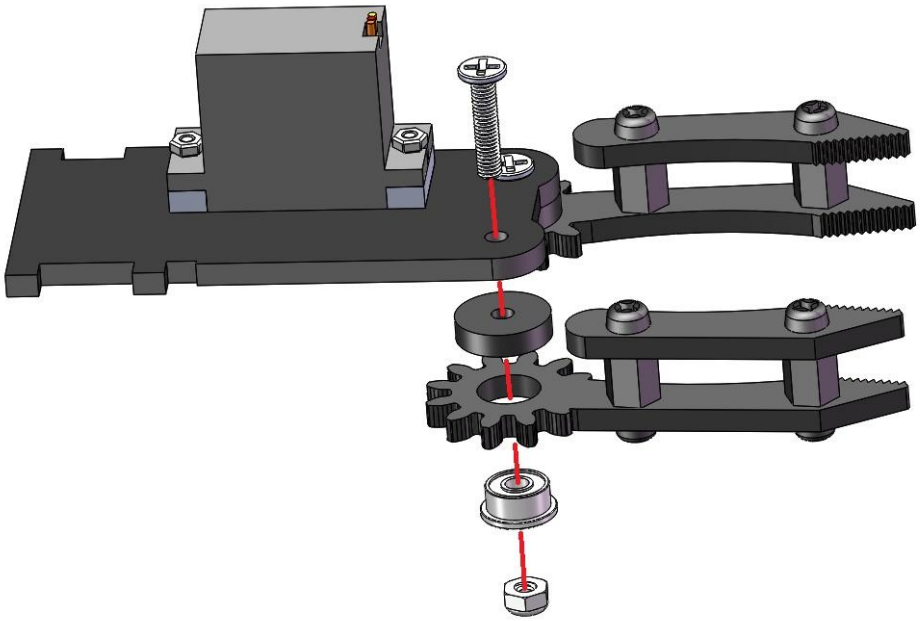
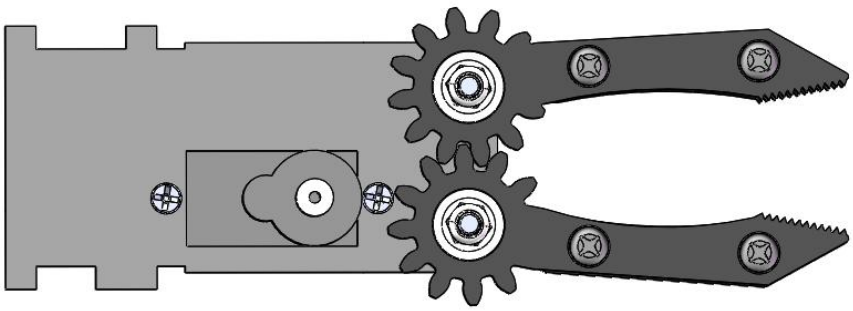


Step 13 Assembling left claw

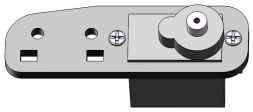
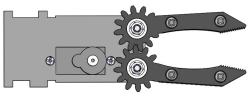
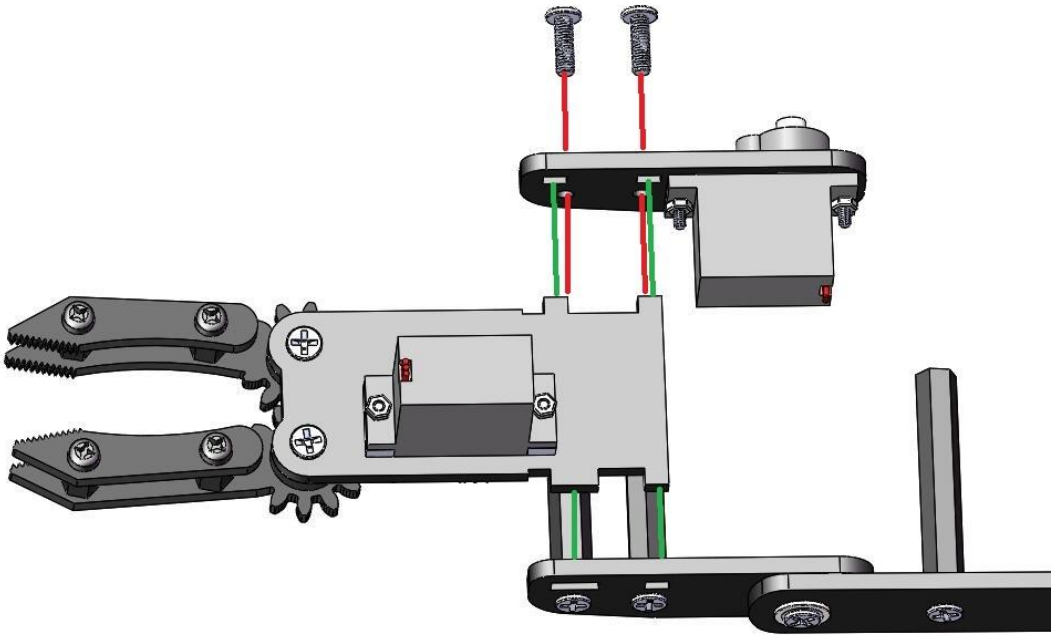
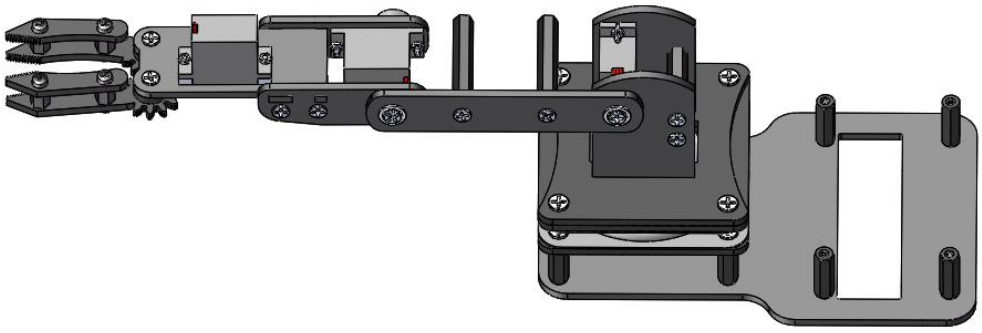
Part in Step 10	Part in Step 11	Flange Bearing 1PCS
		
Acrylic 1PCS	M3X14MM Screw 1PCS	M3 Self-locking Nut 1PCS
		
		
		

21

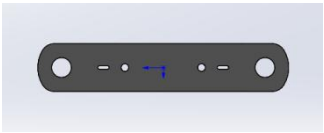
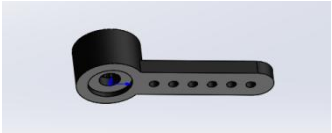
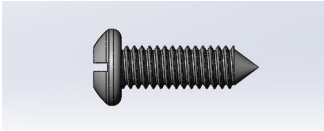
Step 14 Assembling right claw

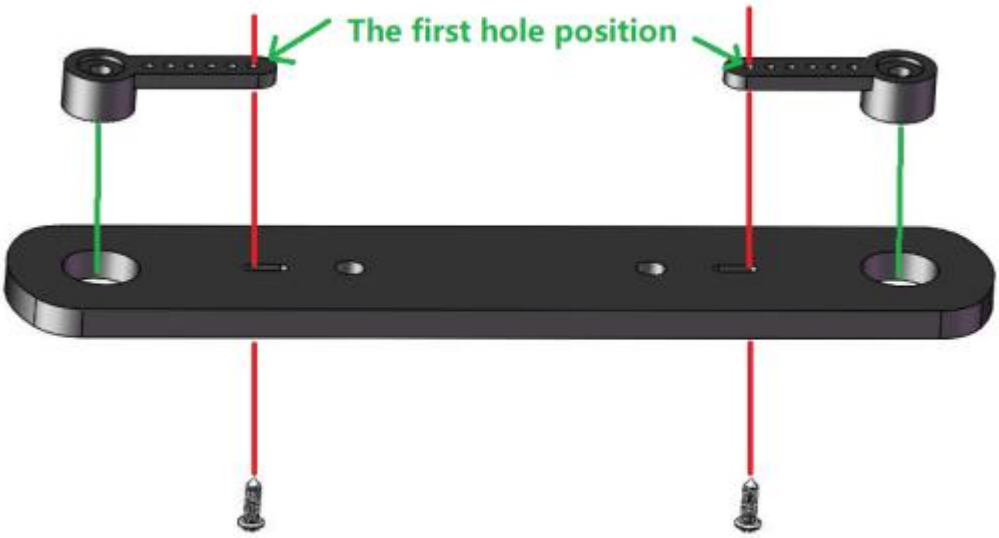
Part in Step 13	Part in Step 12	Flange Bearing 1PCS
		
Acrylic 1PCS	M3X14MM Screw 1PCS	M3 Self-locking Nut 1PCS
		
		
		

Step 15 Claw and Servo D

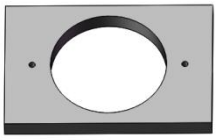
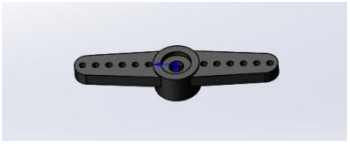
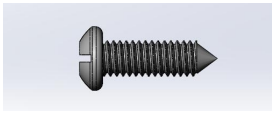
Part in Step 8	Part in Step 9
	
Part in Step 14	M3X10MM Screw 2PCS
	
	
	

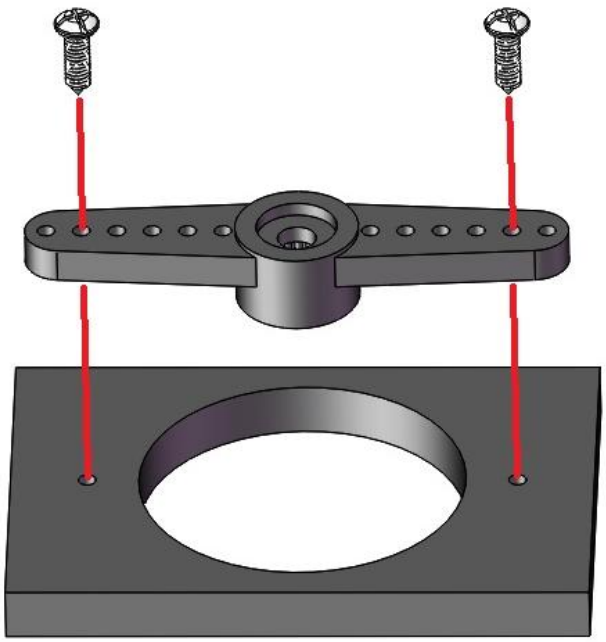
Step 16 Part for connect Servo B and Servo C

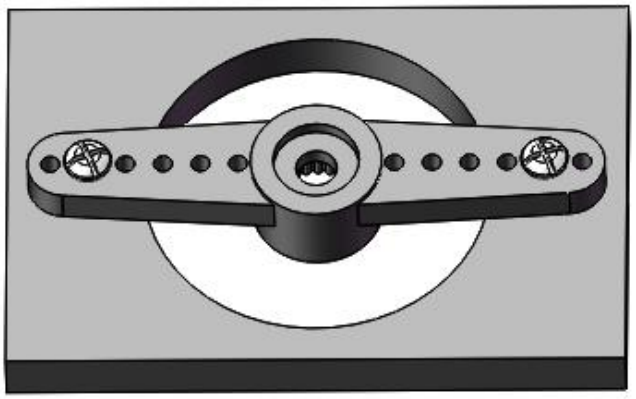
Acrylic Sheet 1PCS	Servo Arm 2PCS	M1.4X5 Self-locking Screw 2PCS
		



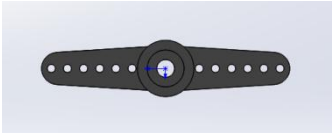
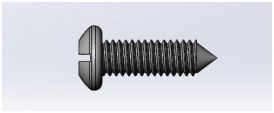
Step 17 Part fixed to Servo A

Acrylic Sheet 1PCS	Servo Arm 1PCS	M1.4X5 Self-locking Screw 2PCS
		

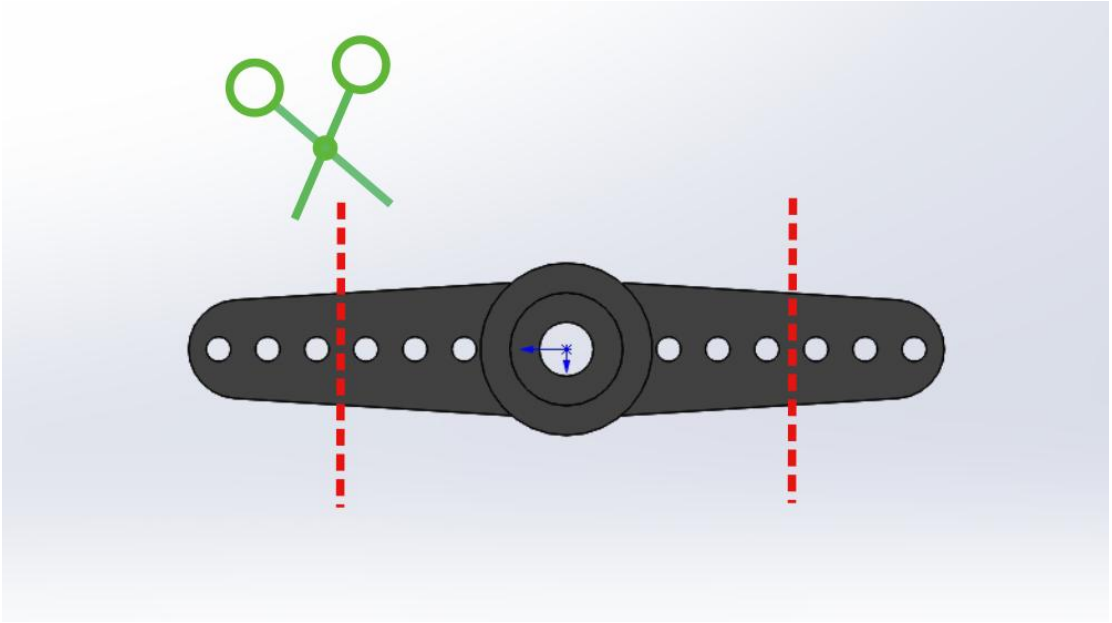


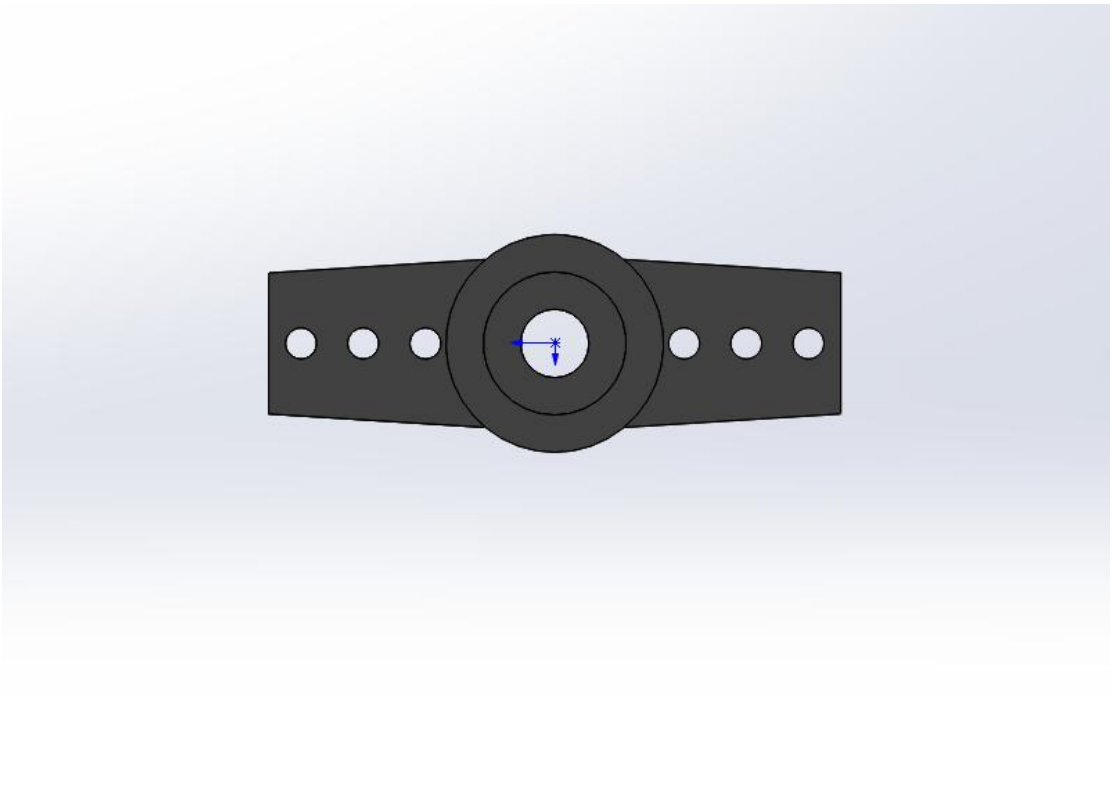


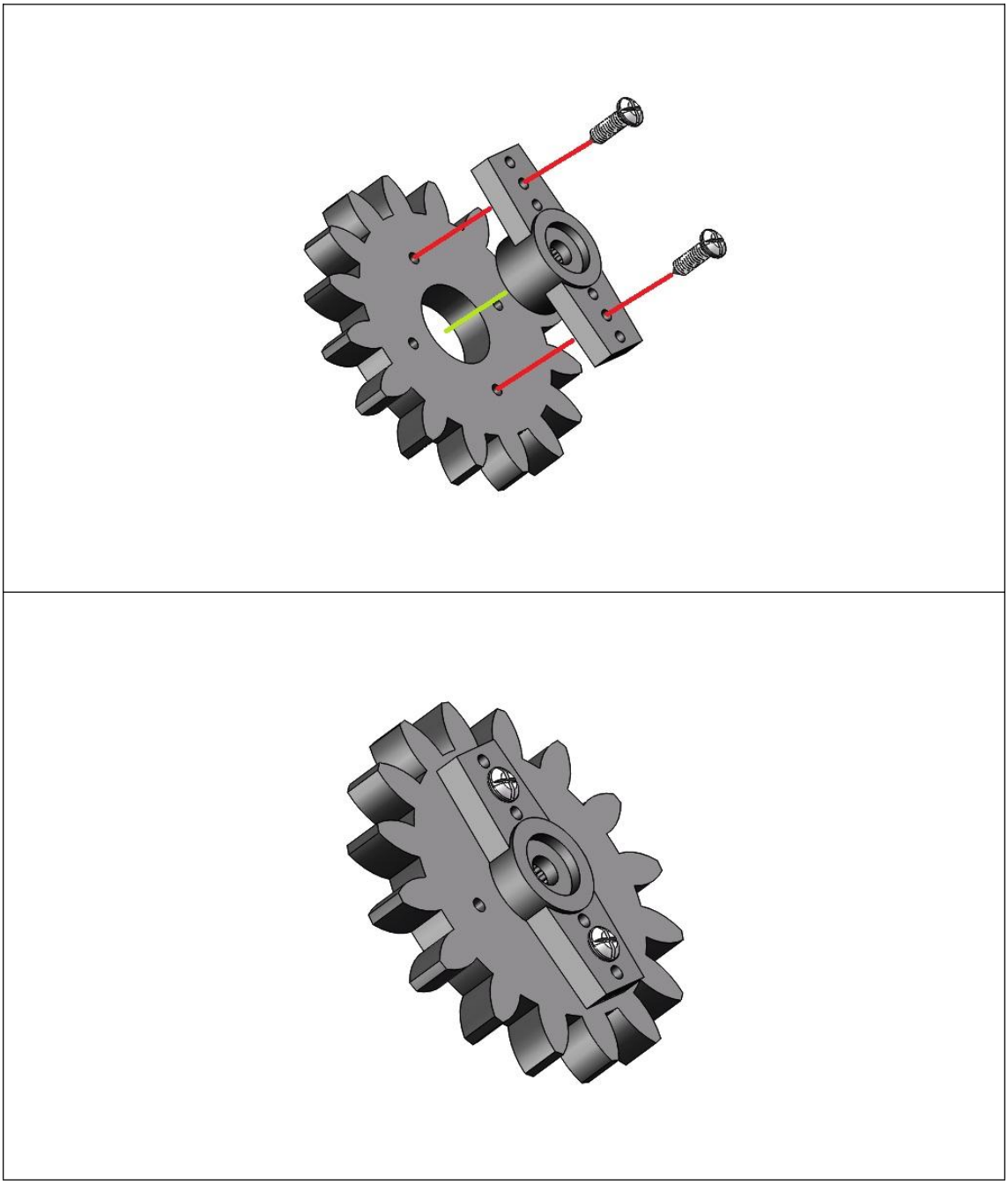
Step 18 Part fixed to Servo D

Acrylic Sheet 1PCS	Servo Arm 1PCS	M1.4X5 Self-locking Screw 2PCS
		

We have already cut it and you can find it in one of the bags!

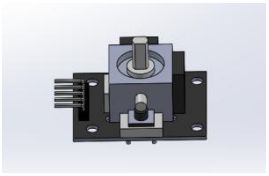


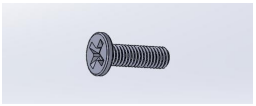
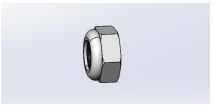


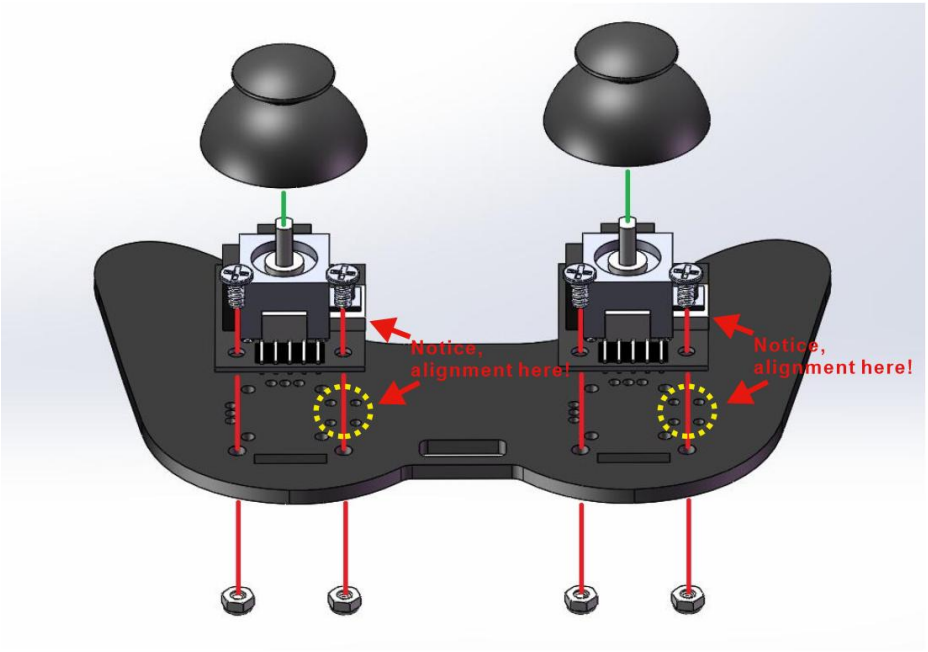


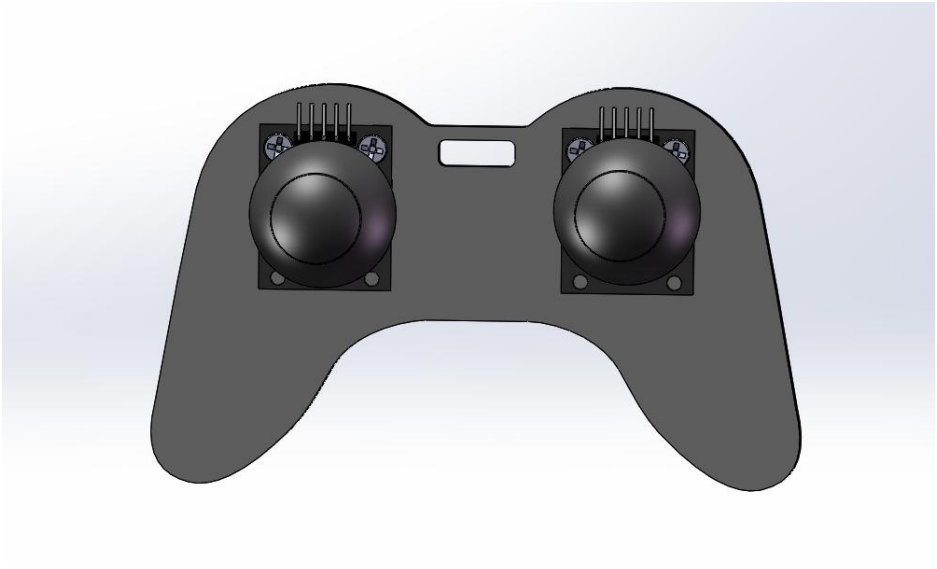
27

Step 19 Joystick

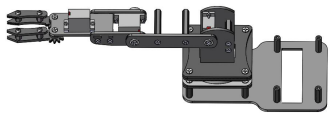
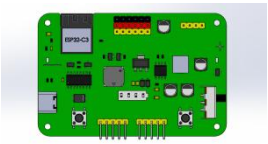
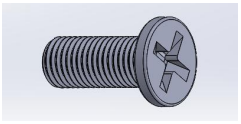
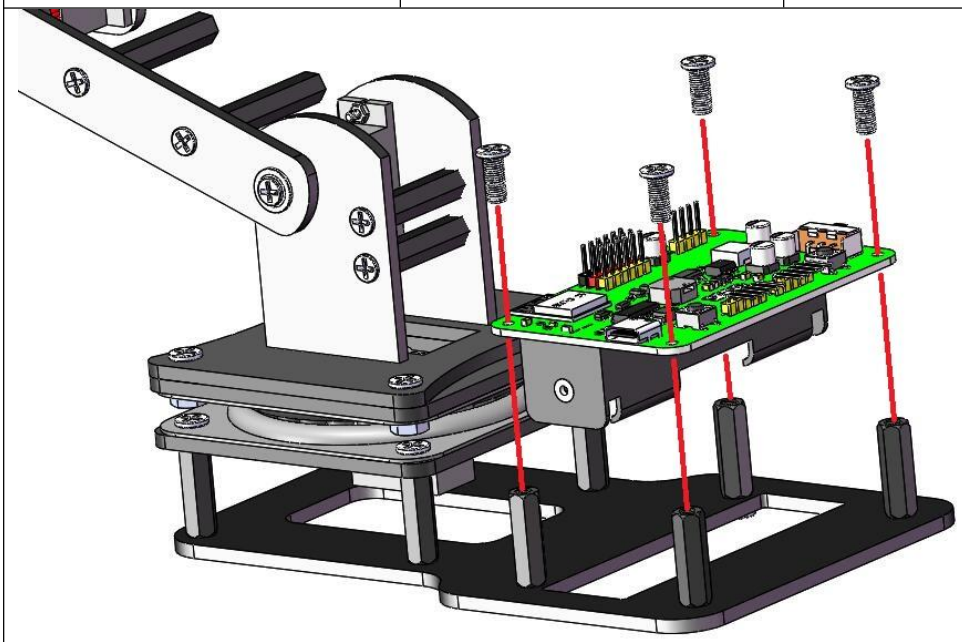
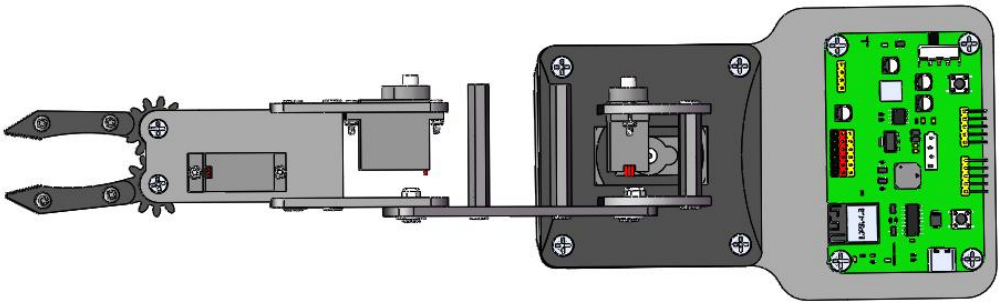
Acrylic Sheet 1PCS	Joystick Module 2PCS
	

Joystick Hat 2PCS	M3X10MM Screw 4PCS	M3 Self-locking Nut 4PCS
		

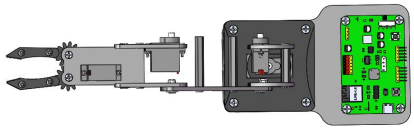
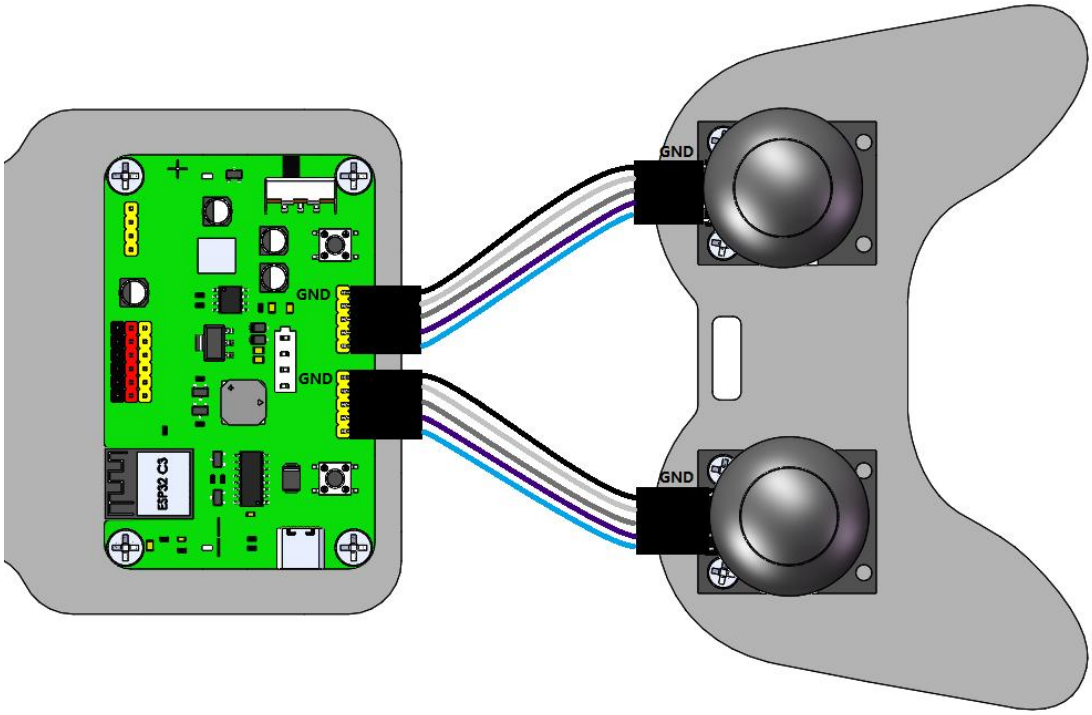
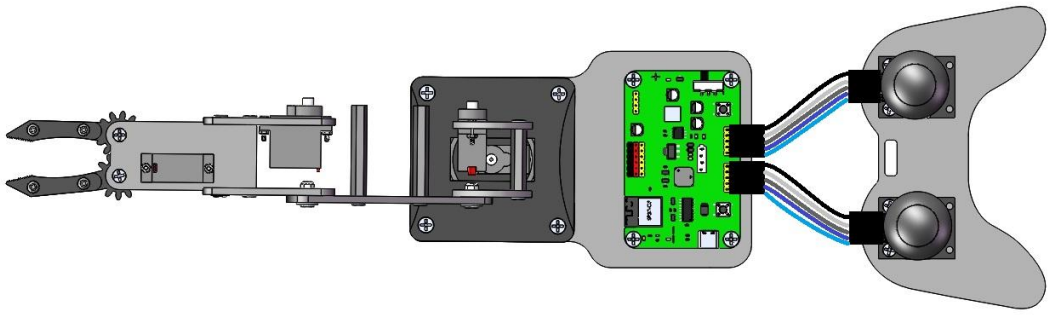




Step 20 ESP32 Board

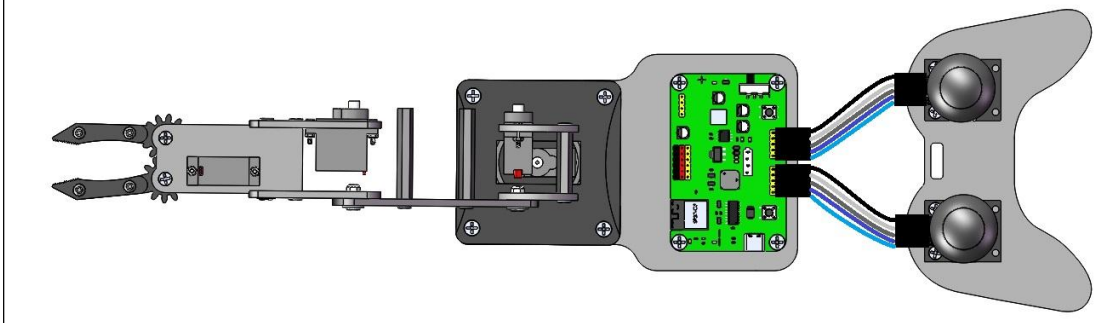
Part in Step 15	ESP32 Control Board 1PCS	M4X10MM Screw 4PCS
		
		
		

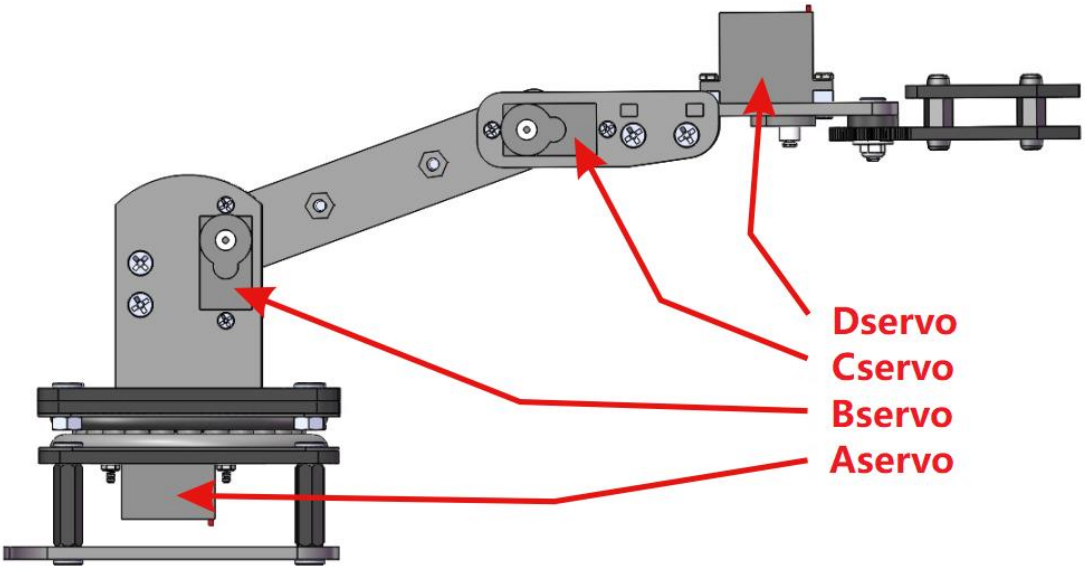
Step 21 Connecting joystick

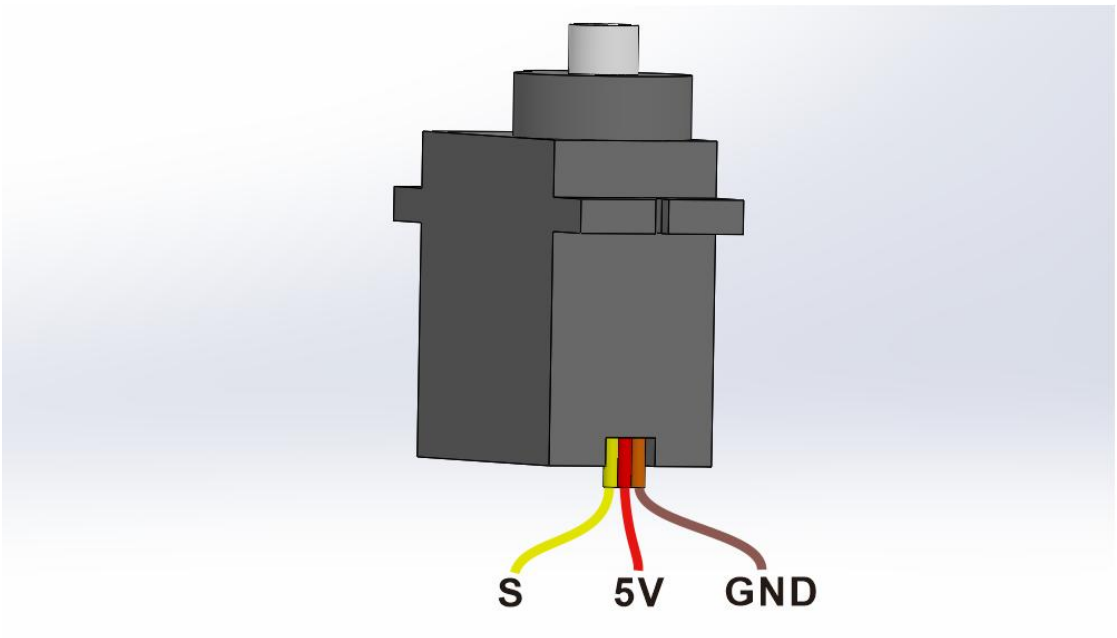
Part in Step 19	Part in Step 20
	
5P Dupont Wire 2PCS	
	
	
	

Step 22 Connecting the servo A,B,C,D to ESP32 board

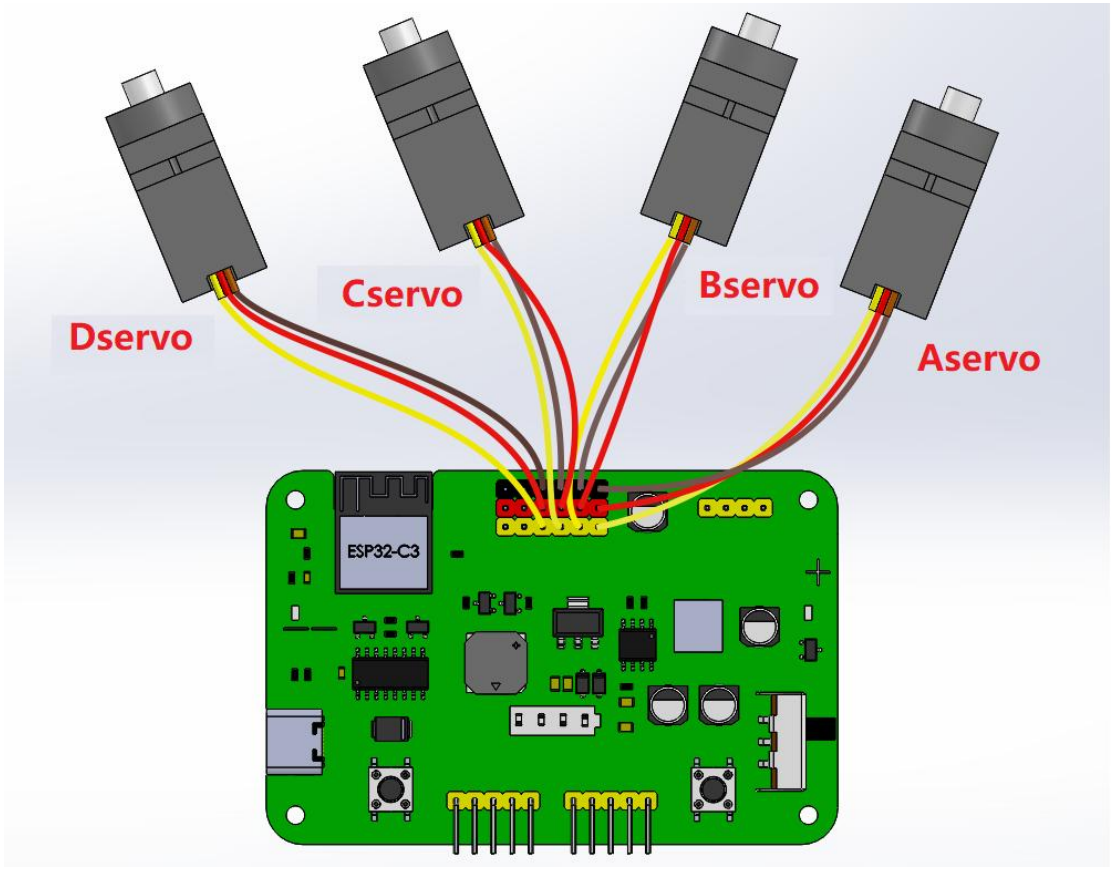
Part in Step 21



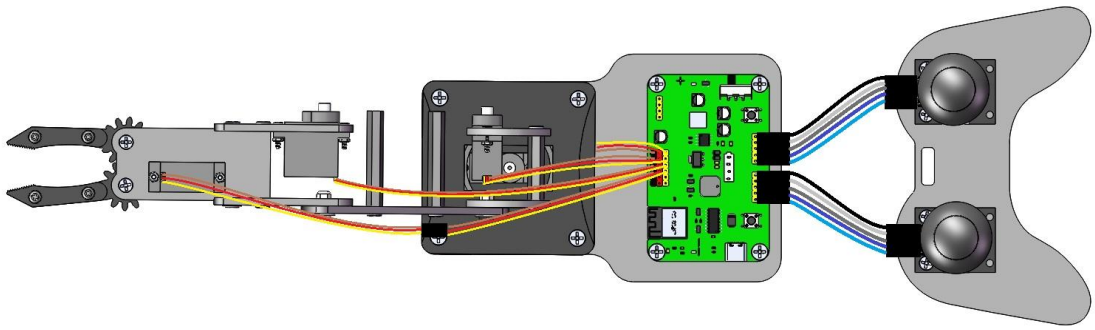




Dservo=GPIO7	Cservo=GPIO6	Bservo=GPIO5	Aservo=GPIO4

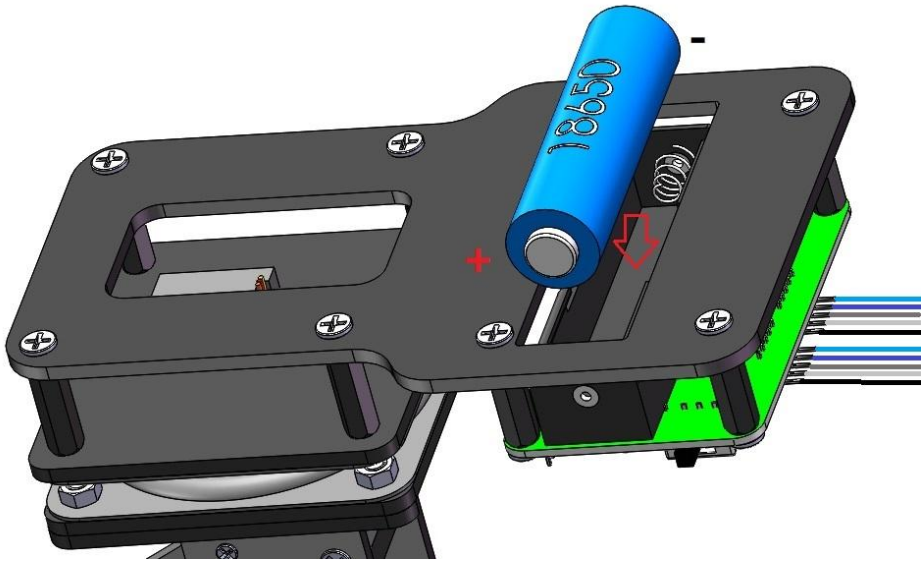


Correct wiring diagram of all servos



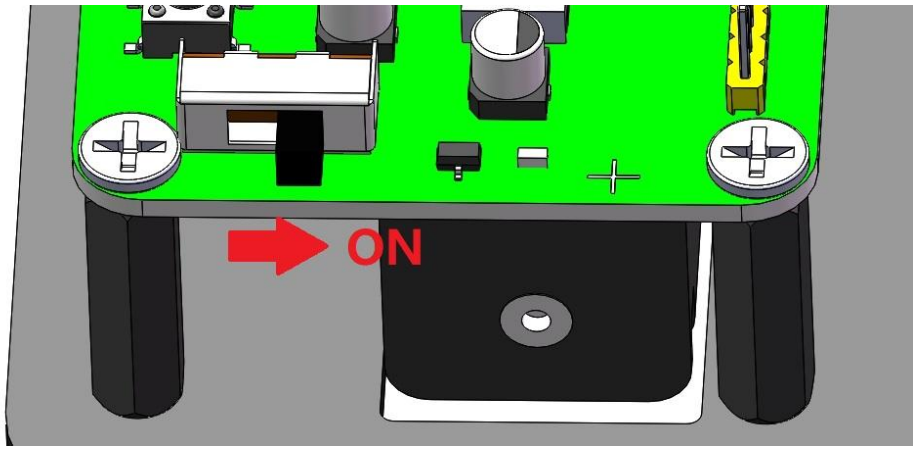
Step 23 Power the ESP32 board

Install a 18650 lithium battery (**need to be purchased by yourself**)



Very important: Turn on the power switch！ ！ ！

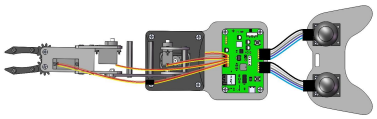
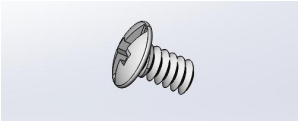
We have already **burned the program** on the control board. After power is turned on, the control board can set all the servos to **90 degrees**.

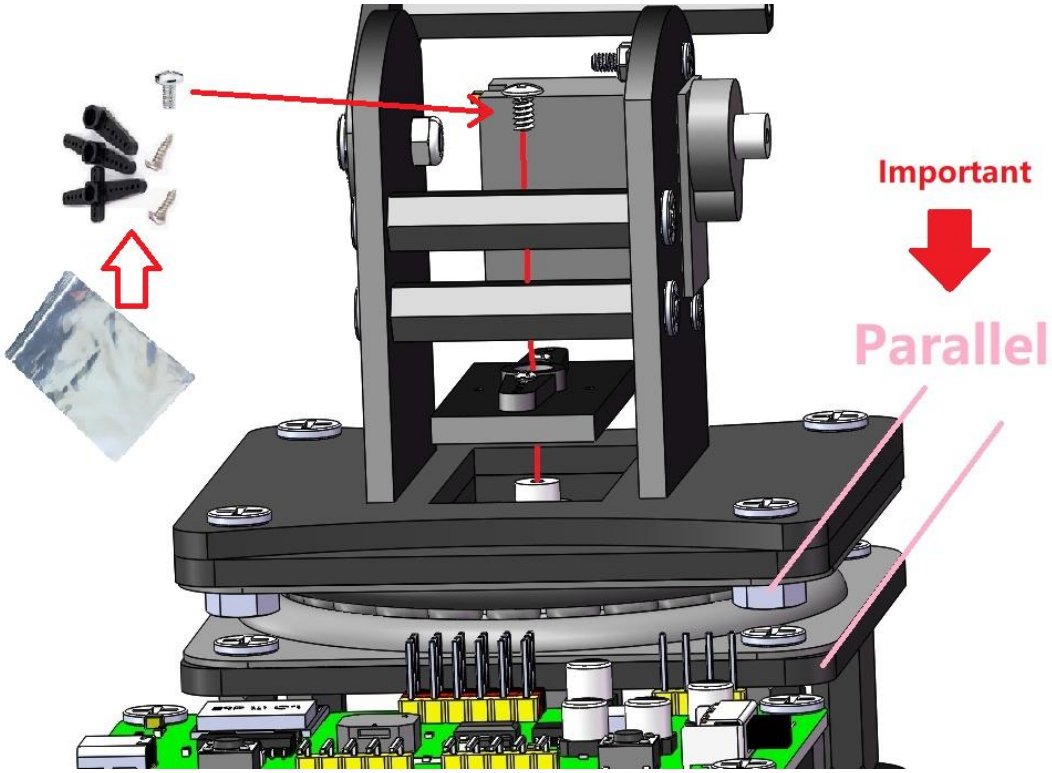


1. Make sure the battery is fully charged!
2. When you turn on the power switch, you will hear the servo turning!
3. In the following installation steps, please keep the power on to prevent the servo angle from being changed!

33

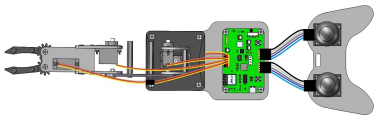
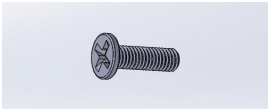
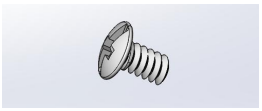
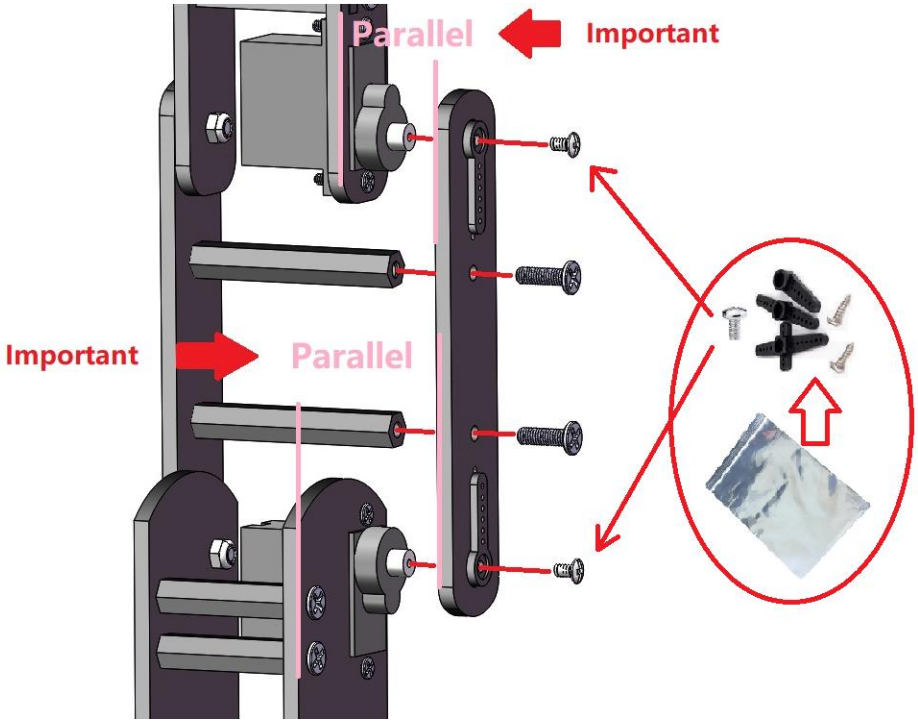
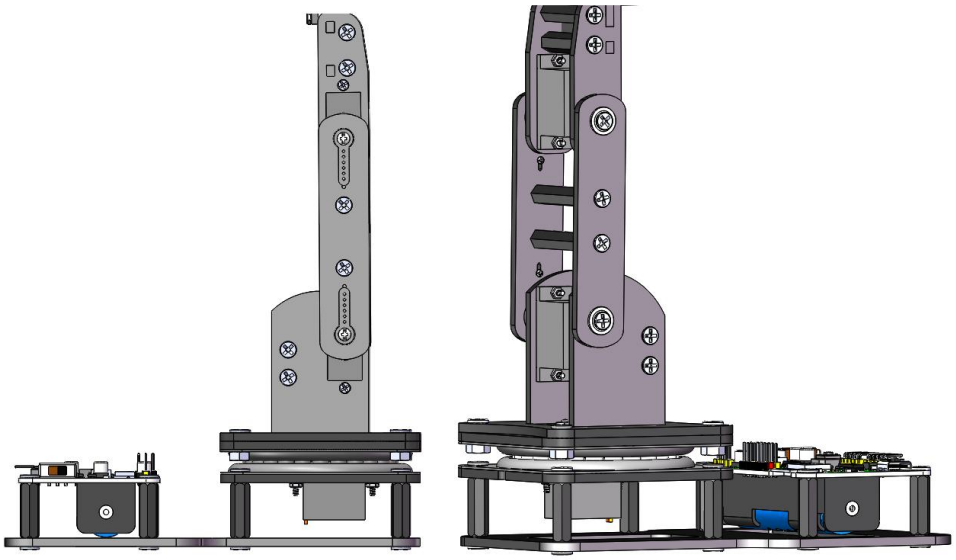
Step 24 Fix the turntable with the servo A

Part in Step 23	Part in Step 17	M2.5X4mm Screw 1PCS
		

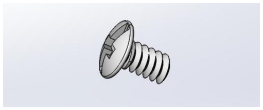


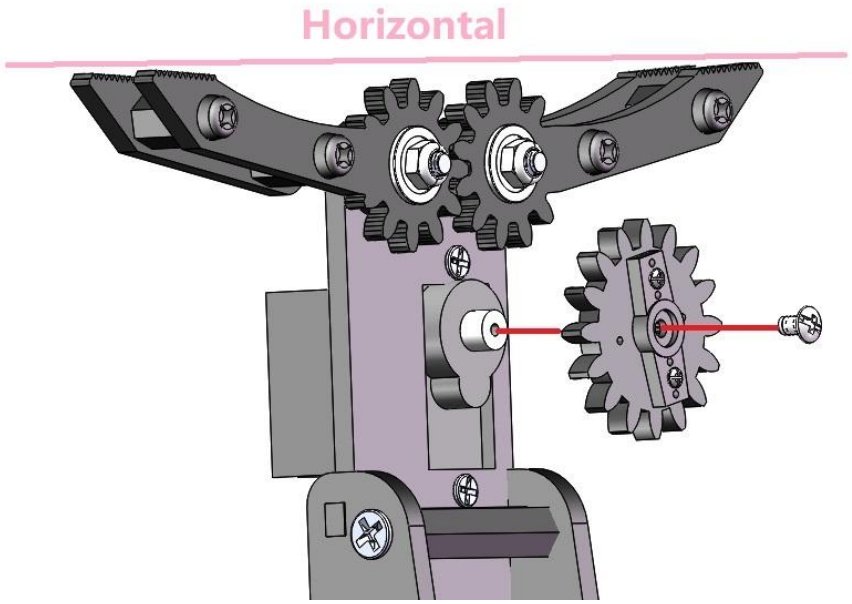
34

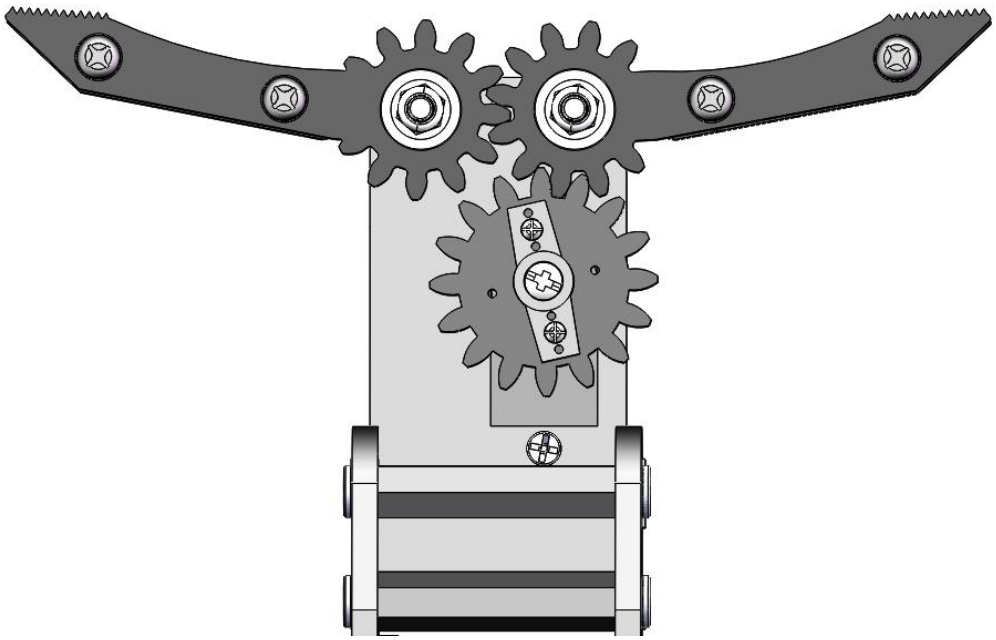
Step 25 Assembling servo B and servo C together

Part in Step 24	Part in Step 16
	
M3X10MM Screw 2PCS	M2.5X4mm Screw 2PCS
	
	
	

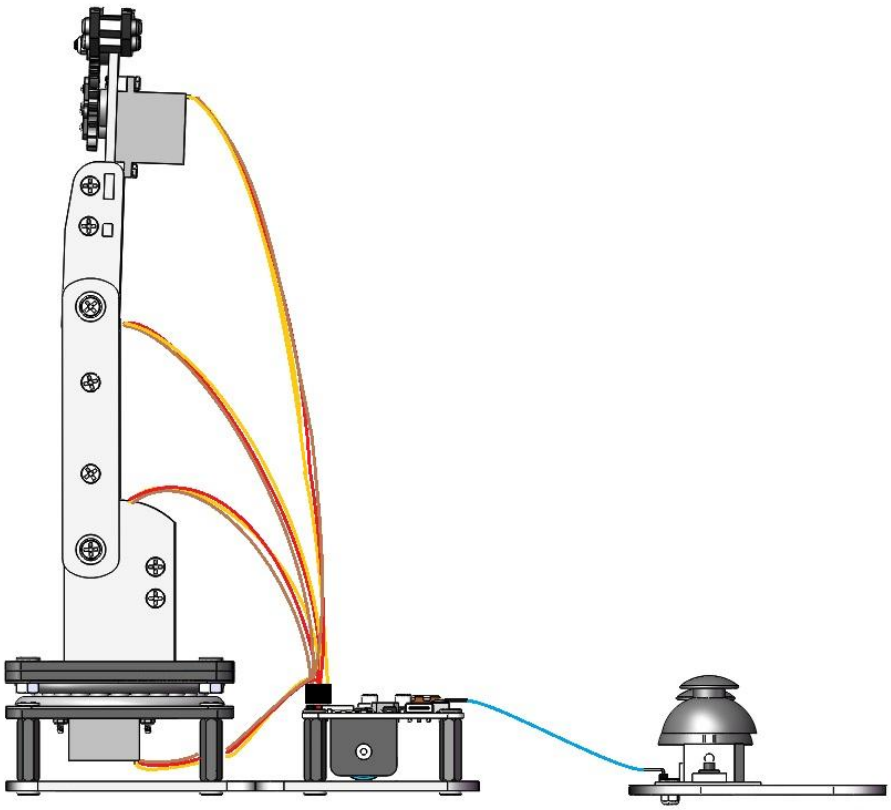
Step 26 Servo D and claw

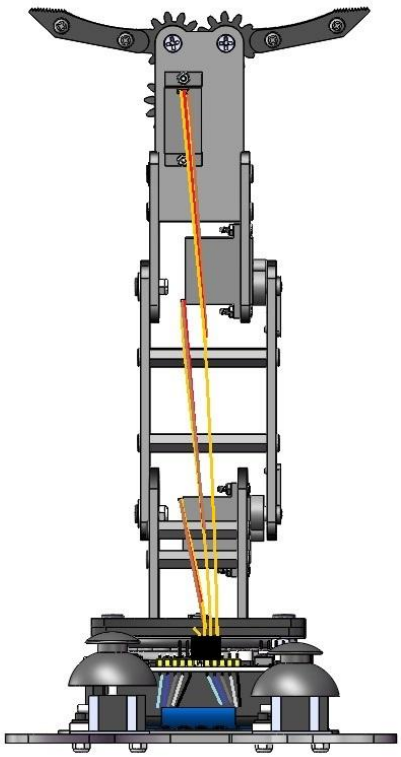
Part in Step 25	Part in Step 18	M2.5X4mm Screw 1PCS
		





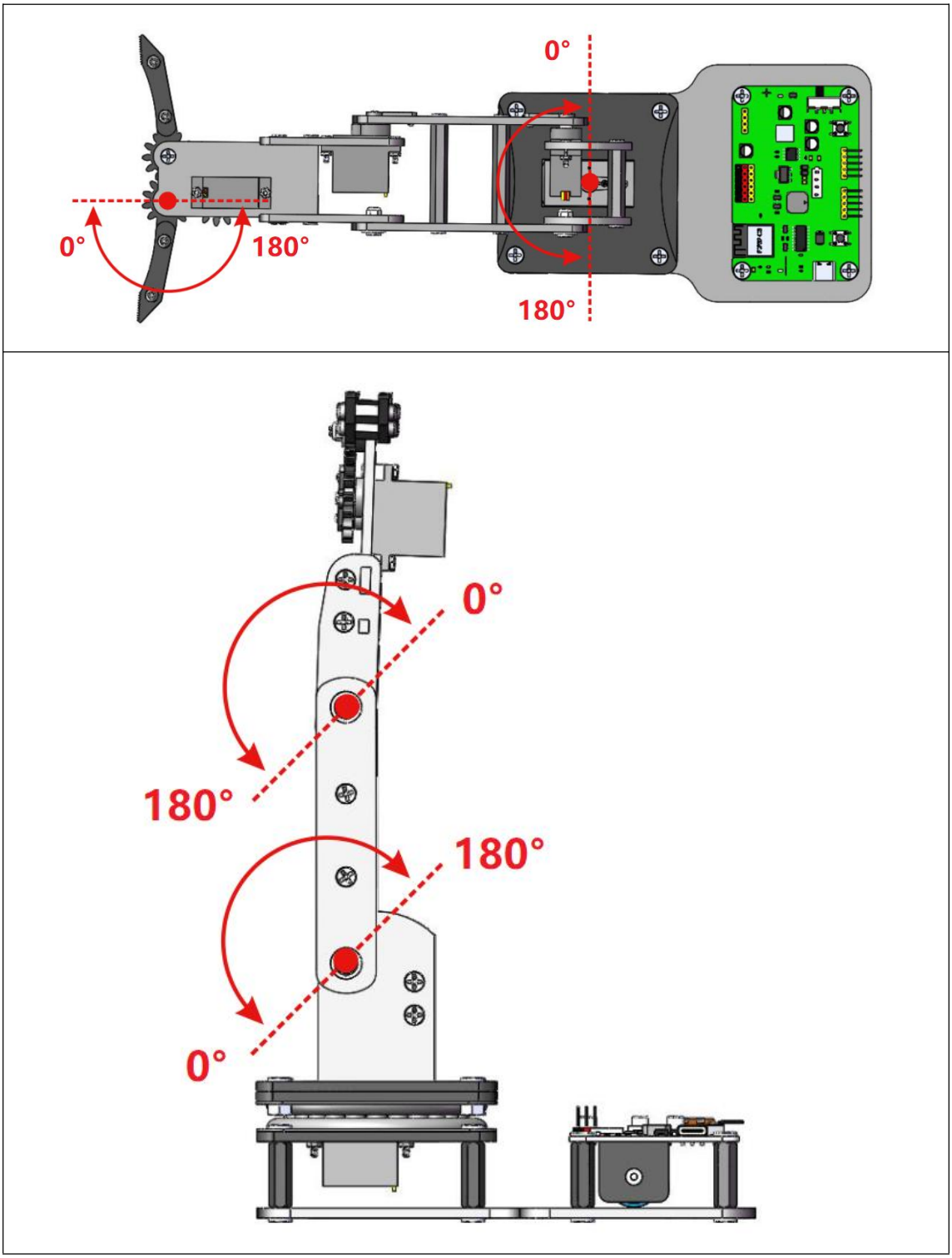
Step 27 Complete assembly





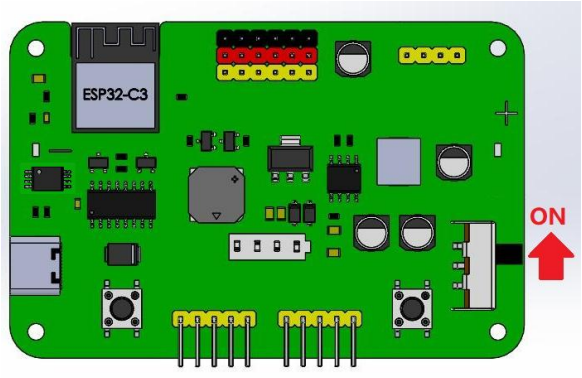
3. Playing with the Robot Arm

3.1 Degrees of freedom



3.2 Control the eArm with a Joystick

Turn the power switch ON the eArm to the **"ON"** state.

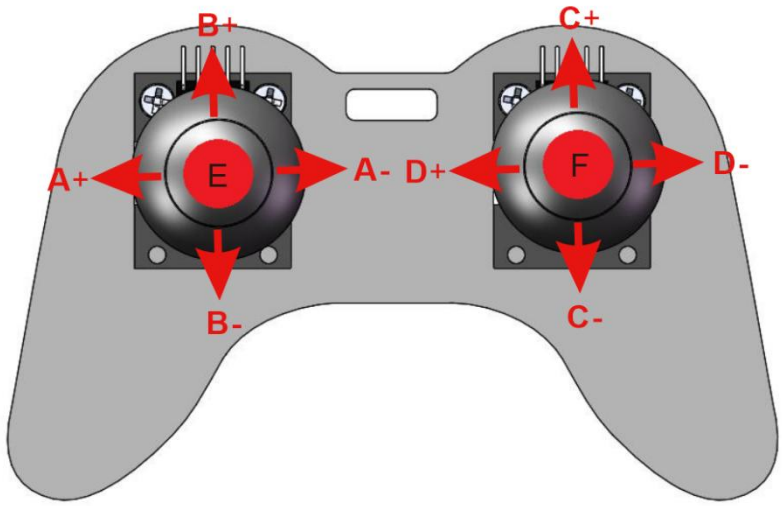


1.Make sure the 18650 lithium battery is installed.
2. Make sure the battery is fully charged!

Mode switch:

After the robot is powered on, the default control mode is joystick control. You can switch the control mode by pressing the "F" button on the right joystick. The modes are "Joystick handle" and "Web App" control modes.

When you press "F" on the right joystick, the buzzer sounds once and the mode switches to Web APP control mode, and the joystick control will not work.

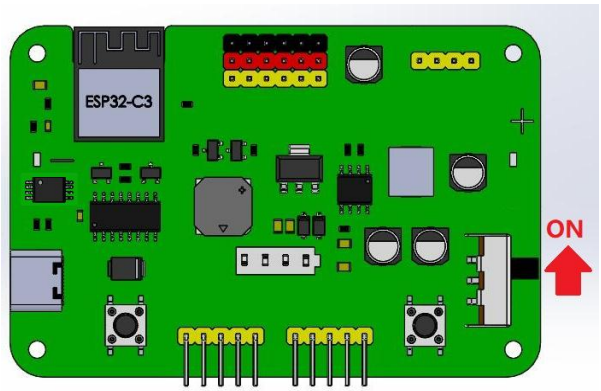


A+: Control servo A, the robot arm turns left.	A+: Control servo A, the robot arm turns right.

B+: Control servo B, the posterior arm rise.	B+: Control servo B, the rear arm moves downward.
C+: Control servo C, the forearm rise.	C+: Control servo C, the front arm moves downward.
D+: Control servo D, the robot arm clamp opens.	D+: Control servo D, the robot arm clamp closes.
E: Controls whether the buzzer sounds	F: Switch between Joystick handle and Web APP control mode.

3.4 Control the eArm with Web App

Turn the power switch ON the eArm to the **"ON"** state.



1.Make sure the 18650 lithium battery is installed.
2. Make sure the battery is fully charged!

Turn on the WLAN on the phone and select **eArm**

After connecting to Wifi, your phone may pop up a window saying it cannot connect to the network, please ignore it!

40

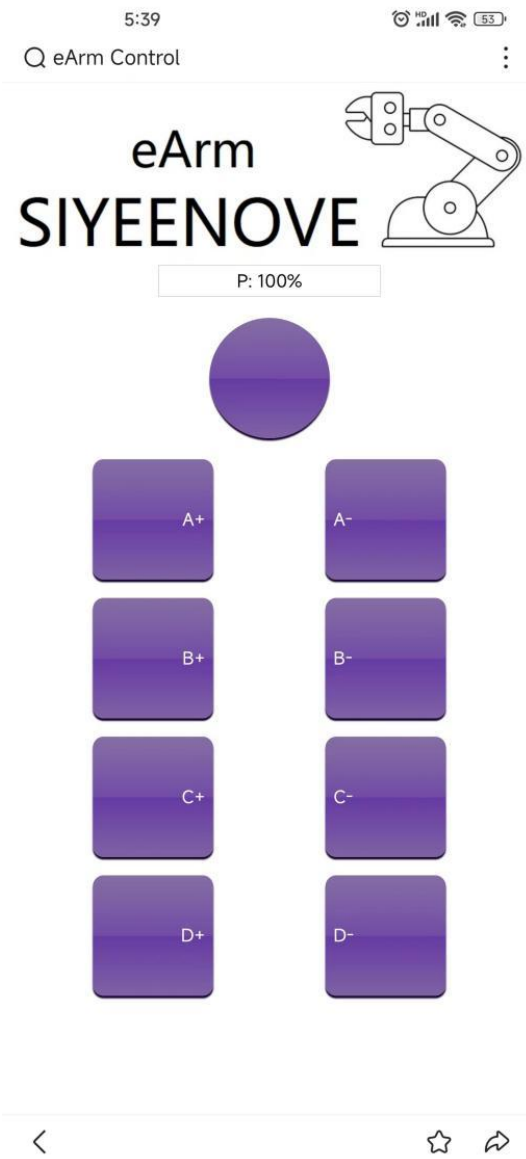
Turn off the mobile phone's **mobile data**, otherwise you may not be able to use the Web APP.

Open your phone's browser and link to it by typing **"192.168.4.1"** in the address bar.

The following page should appear in your browser, now you can control the eArm!

Note: You can use any device that can connect to the eArm Wifi to control it (Web App). Please note that you should only use one device to connect and control the eArm at a time. Multiple devices may not be able to control the eArm at the same time.

41

	Round button: Turn the buzzer on or off A+: Control servo A, the robot arm turns left. A-: Control servo A, the robot arm turns right. B+: Control servo B, the posterior arm is raised. B-: Control servo B, the posterior arm is down. C+: Control servo C, the forearm is raised. C-: Control servo C, the forearm is down. D+: Control servo D, the robot arm clamp is closed. D-: Control servo D, the robot arm clamp is opened.

42

4. Programming

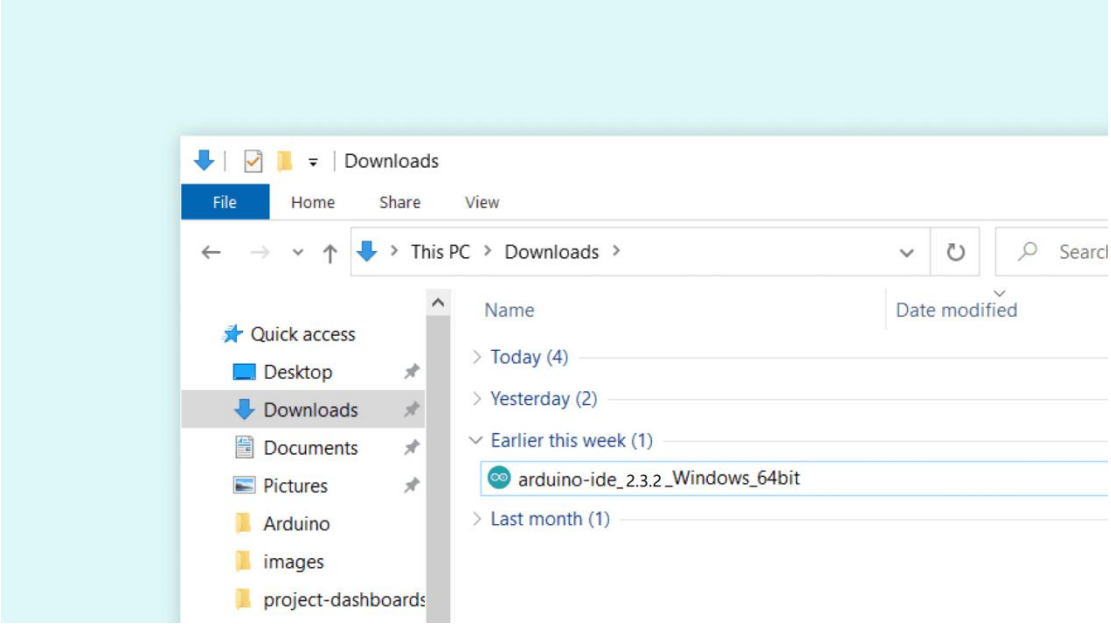
4.1 Install Arduino IDE

The latest version of Arduino IDE can be downloaded from the Arduino official website：https://www.arduino.cc/en/software

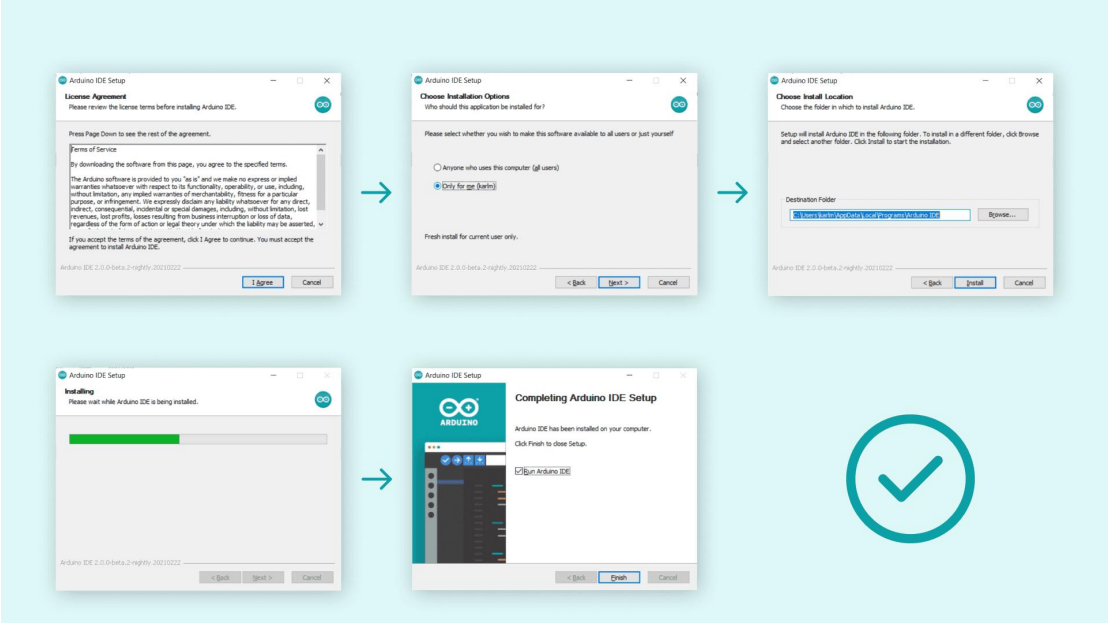
In the following we will demonstrate the installation of **Windows Win 10 and newer, 64 bits** on PC.

Installation

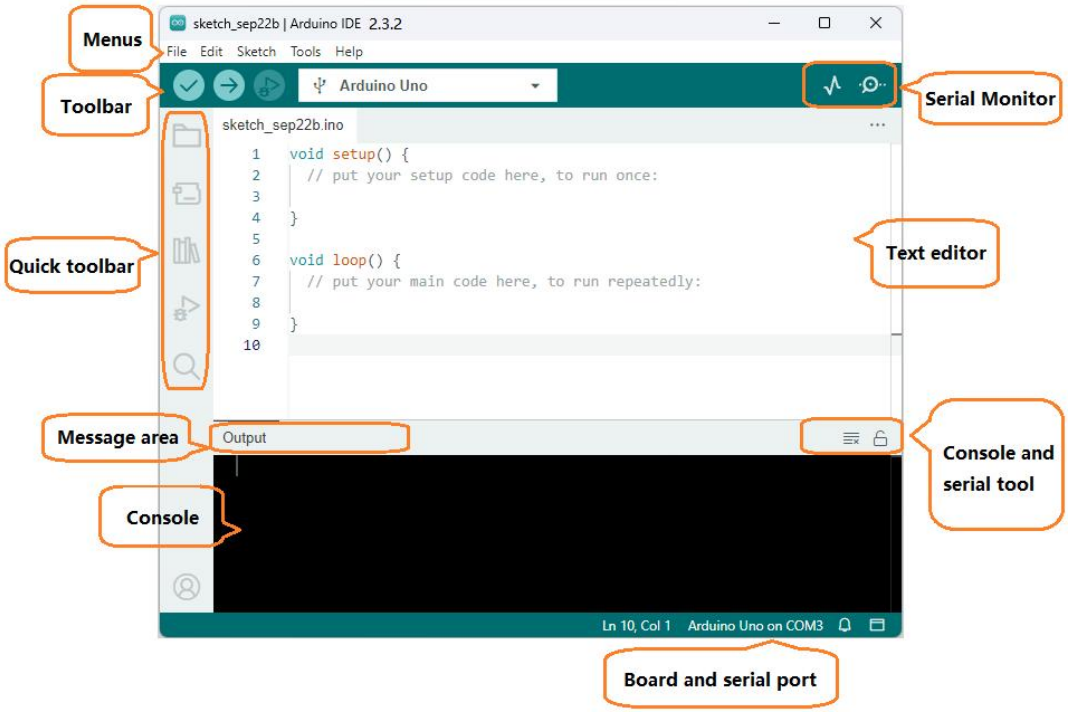
To install the Arduino IDE 2.x.x on a Windows computer, simply run the file downloaded from the software page.



Follow the pop-up window prompts to complete the installation process.



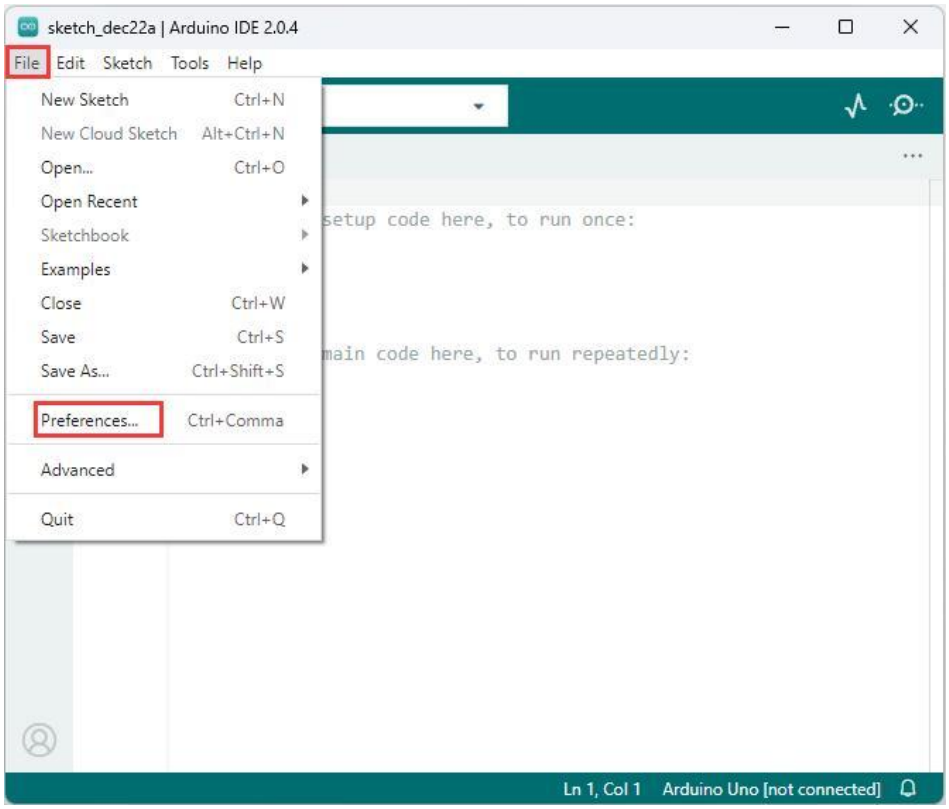
The homepage of Arduino IDE.



44

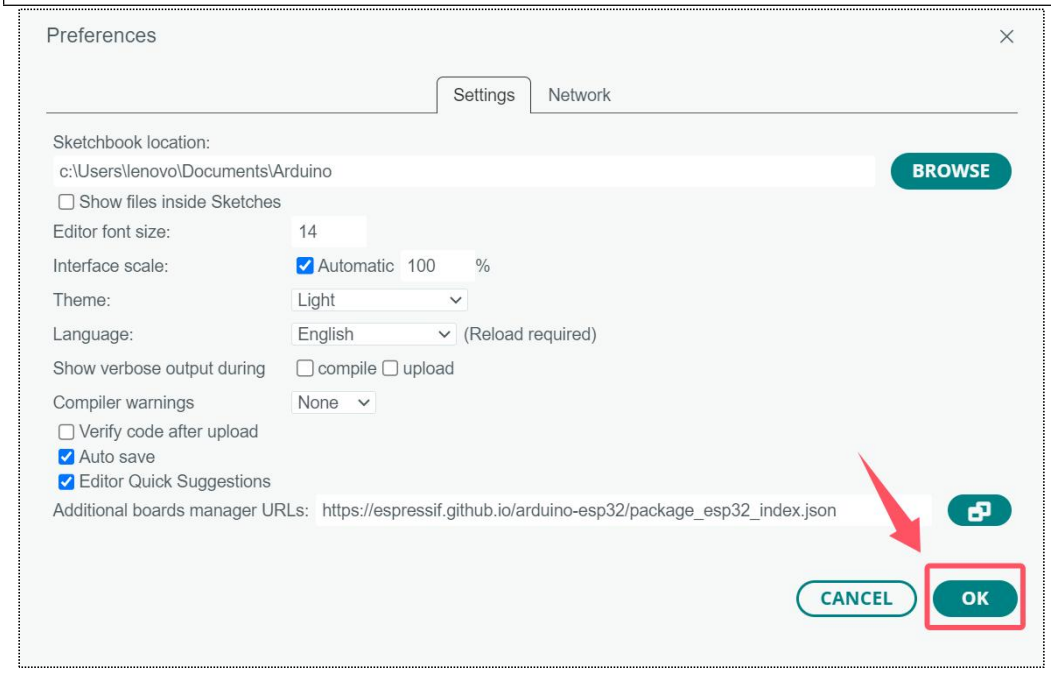
4.2 Install ESP32 development environment for Arduino IDE

Open the arduino IDE，click “File” > “Preferences”，as shown below:

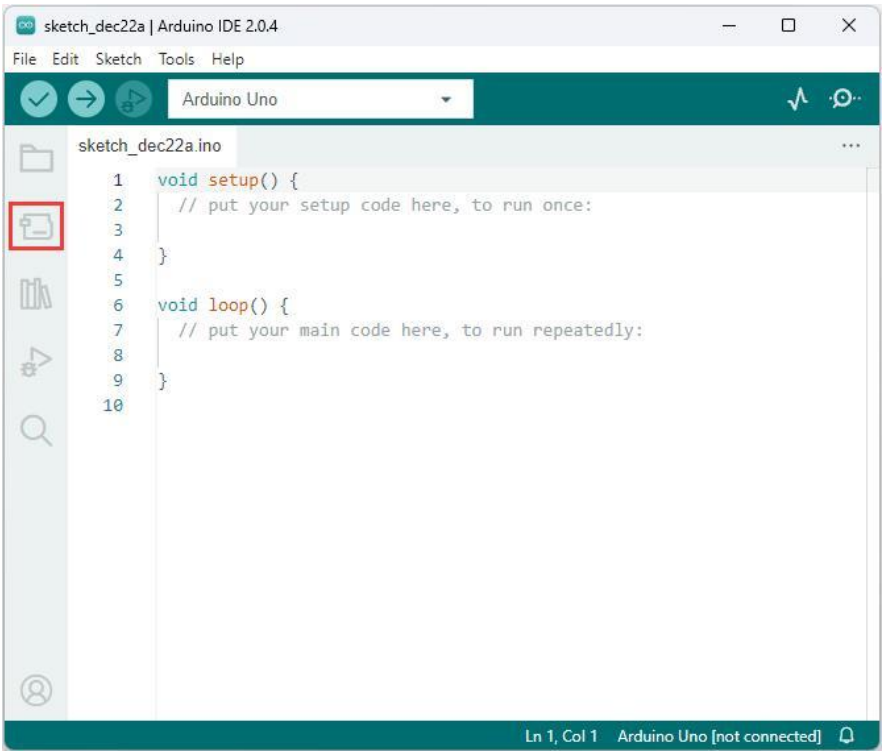


Scroll down the pop-up window and copy the link below into the **Additional boards manager URLs** text box. Then click “**OK**”.

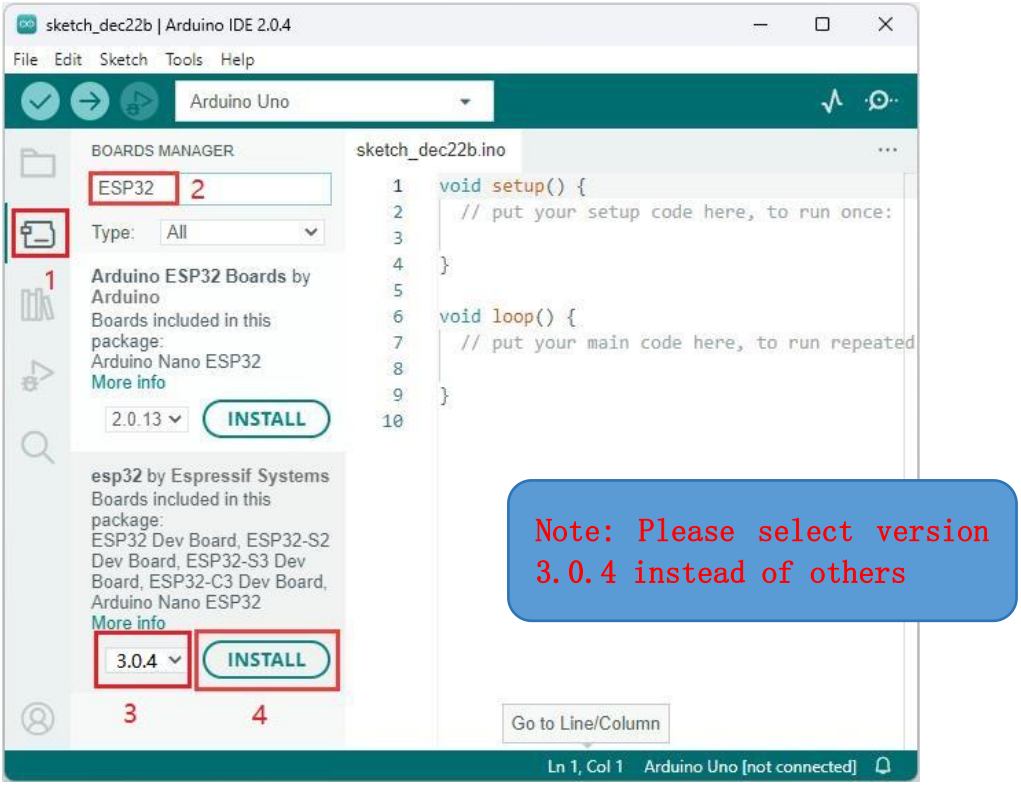
https://espressif.github.io/arduino-esp32/package_esp32_index.json



Click "**Boards Manager**":

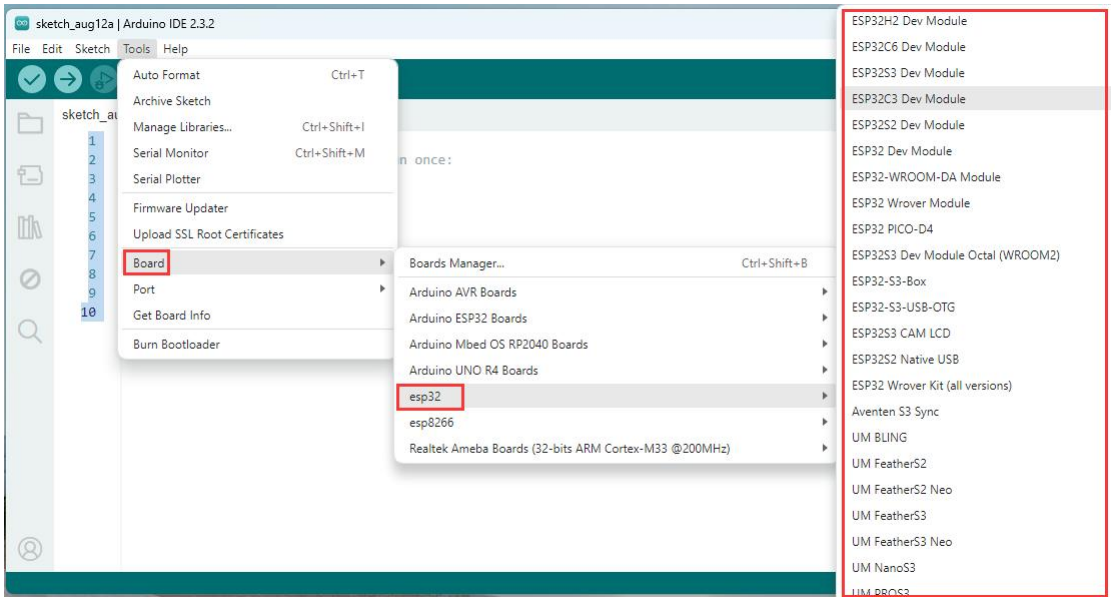


Search for **ESP32** in the Boards Manager, select **esp32 by Espressif Systems in 3.0.4** version and click **INSTALL**



46

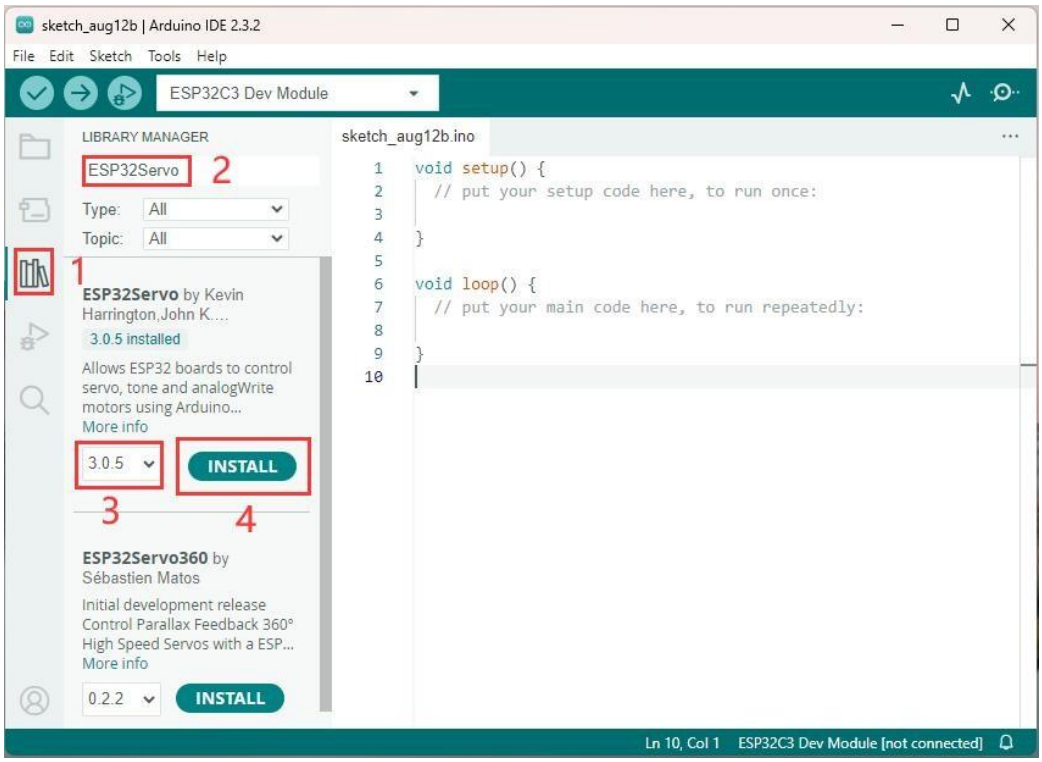
After successfully installing the ESP32 board, click the **"Tools"->"Board"->"ESP32"** menu, and you can find many ESP32 boards.



4.3 Installing the library file

Method 1:

Open the library manager of Arduino IDE and type "**ESP32Servo**" in the search bar, select the library file named ESP32Servo and version **3.0.5** and click Install.

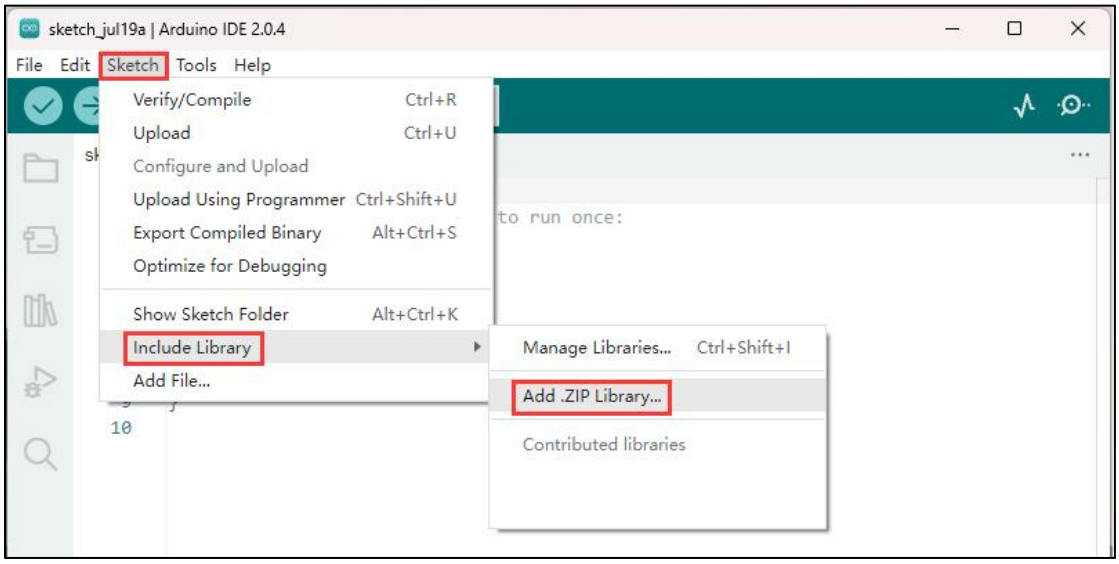


47

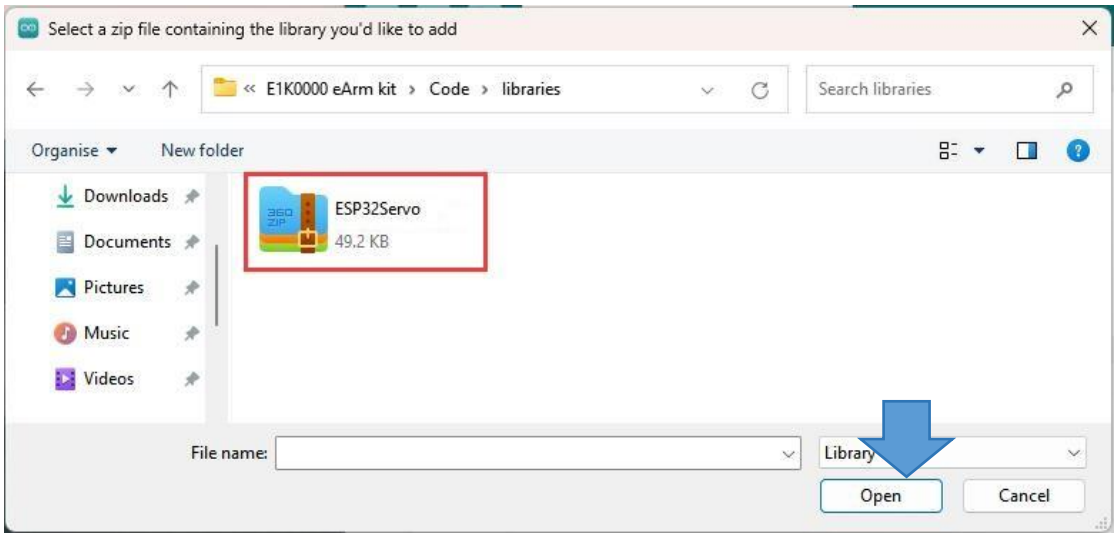
Method 2:

We provide a library zip file in the tutorial package, in the following folder:

Import the library zip file from the Arduino IDE as follows:



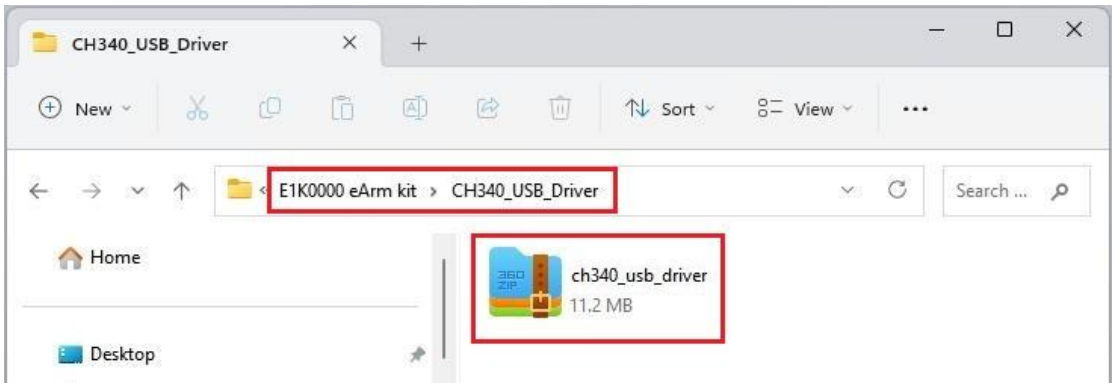
Click **Open** to install the library file.



48

4.4 Install CH340 USB driver

Driver files is provided in our tutorial package:

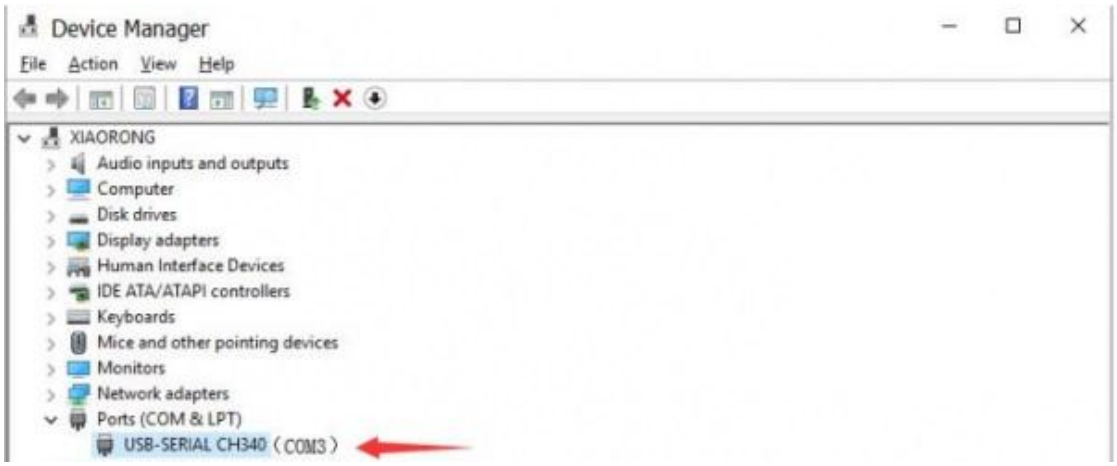


Install the driver on your Windows system

Unzip the file we provided and double-click **"SETUP"** to run the installer:

If the driver is successfully installed, when the esp32 board is connected to the computer through the USB cable, under the path **"Computer" -> "Properties" -> "Device manager"**, you can find **"USB-SERIAL CH340 (COM3)"**, as shown below:

49



Note: Driver display symbol is "USB-SERIAL CH340 (COMx)", "x" can be any number, it depends on what number your computer assigns to the ESP32 device.

Install the driver on your MAC system

Link to download the driver:

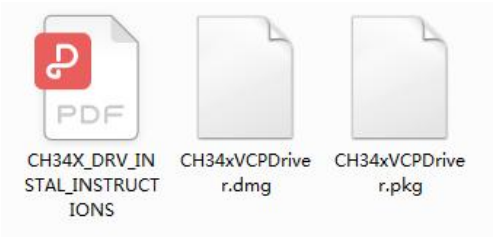
http://www.wch-ic.com/downloads/CH341SER_EXE.html

download

relation files

file name	file content
CH341SER.ZIP	CH340/CH341 USB to serial port Windows driver, supports Windows XP/Vista/7/8/8.1/10/11/ SERVER 2003/2008/2012/2016/2019/2022 -32/64bit, Microsoft WHQL Certified, supports USB to 3-line and 9-line serial port.
CH341SER_LINUX.ZIP	Linux driver for USB to serial port, supports CH340 and CH341, supports 32/64-bit operating systems.
CH341SER_MAC.ZIP	For CH340/CH341/CH342/CH343/CH344/CH347/CH9101/CH9102/CH9103/CH9104/CH9143, USB to serial port VCP vendor driver of macOS , supports OS X 10.9~10.15 , OS X 11 (Big Sur) and above , contains installation guide documents.
CH340DS1.PDF	CH340 datasheet, USB bus converter chip which converts USB to serial port, printer port, IrDA SIR etc. Integrated clock, supports various operating systems, chip information can be customized, this datasheet is about USB to serial port and USB to Infrared Adapter SIR.
CH341DS1.PDF	CH341 datasheet, USB bus converter chip with multiple communication interfaces, such as USB to serial port/parallel port/printer port/I2C interface etc. Drivers support for Windows/Linux/Android/Mac, etc. The datasheet is the description of USB to serial port and printer port.
CH341SER_ANDROID.ZIP	CH340/CH341/CH342/CH343/CH344/CH347/CH9101/CH9102/CH9103/CH9104/CH9143 USB to serial port Android driver-free application library and software, for Android OS 4.4 and above in USB Host mode, without loading Android kernel driver and without root access operation. Includes apk installer, lib library file (Java Driver), App Demo routines (USB to UART Demo project SDK).

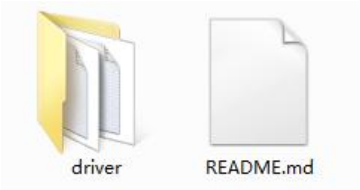
Download the driver from the website and unzip the file to a local installation directory. You will get a PDF installation guide and two installation packages in different formats. For details, see the PDF installation guide.



Install the driver on your Linux system

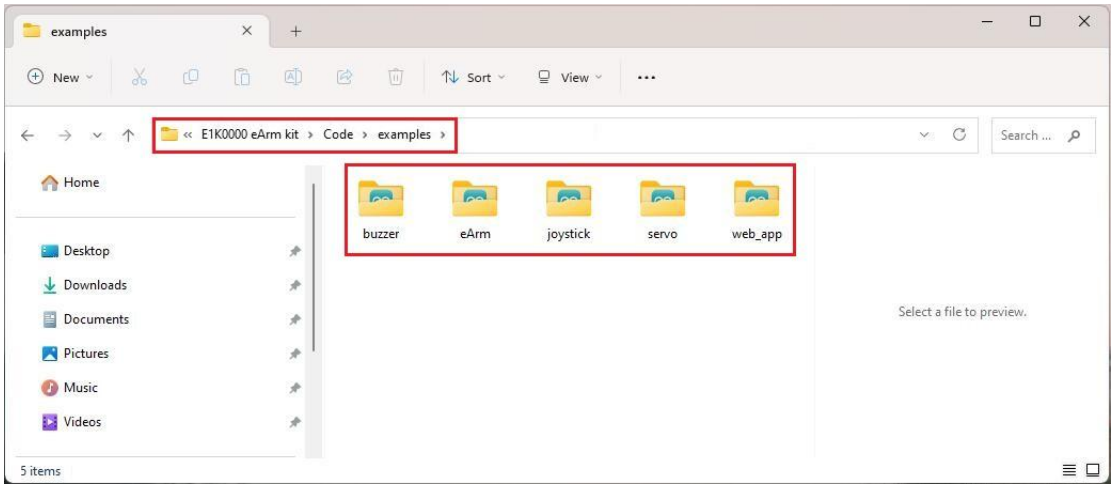
Drivers are almost certainly built into your Linux kernel already and it will probably just work as soon as you plug it in. If not you can download the Linux CH340 Driver (but I'd recommend just upgrading your Linux install so that you get the "built in" one).

If you must install it by yourself, check the "README.md" file in the zip package.



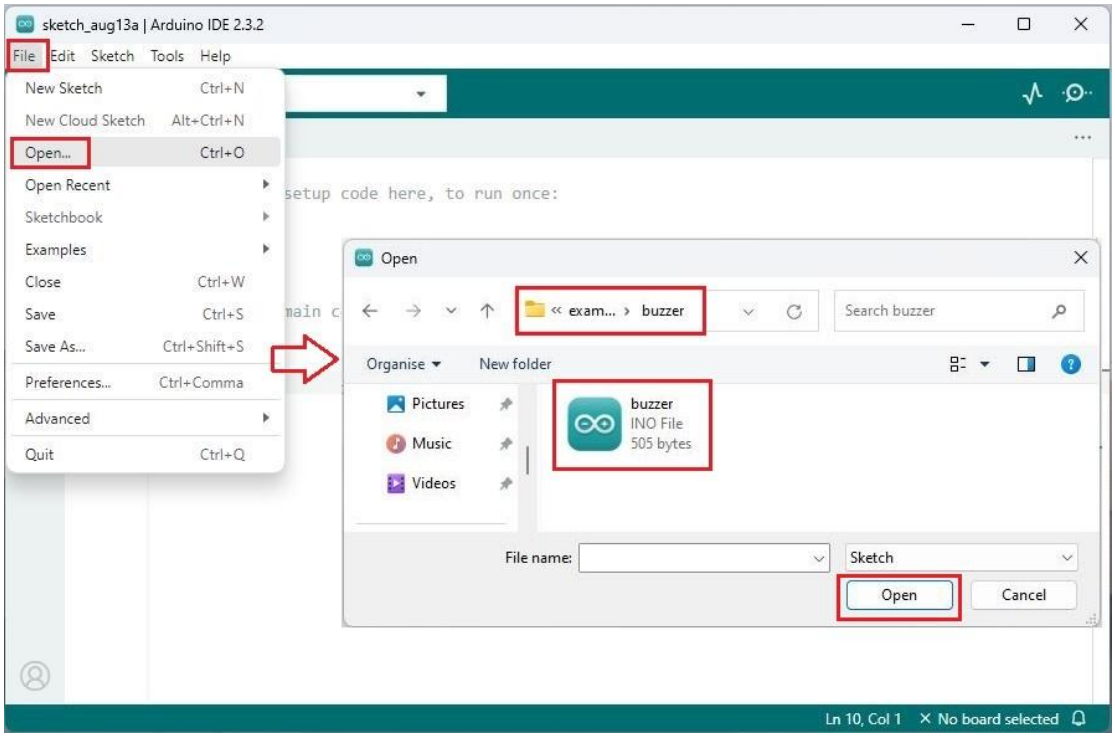
4.5 Basic lessons

The sample code for all the basic lessons is stored in the following folders:

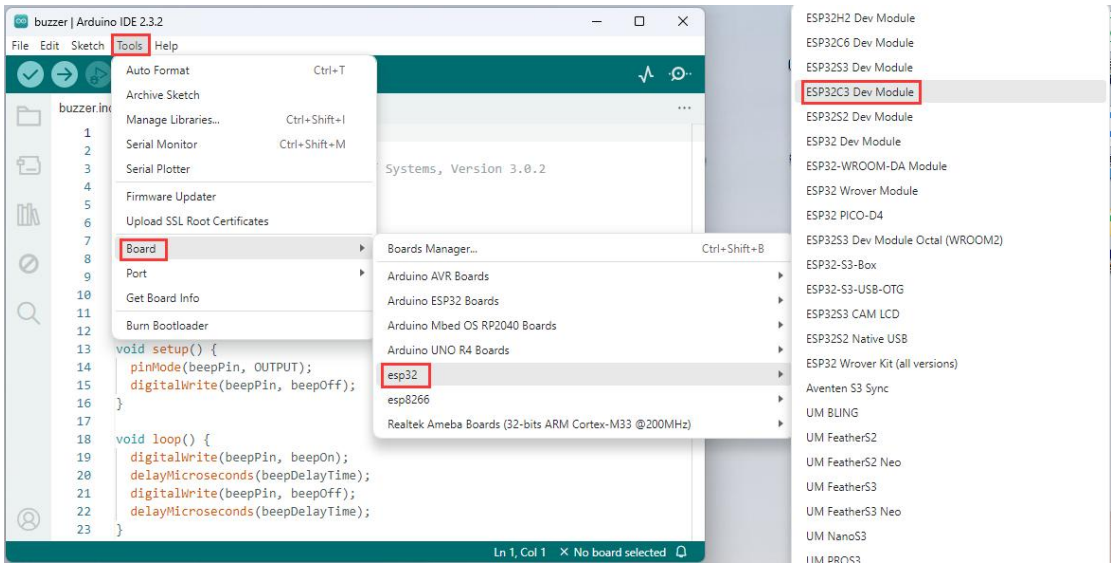


4.5.1 Onboard buzzer

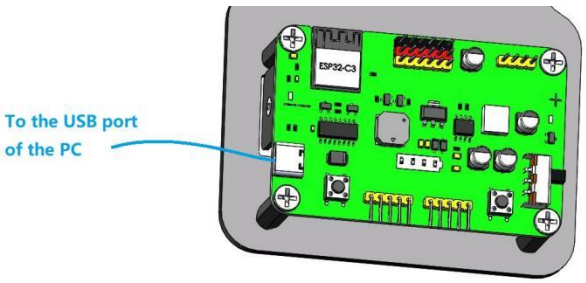
Open the "buzzer" example code in the Arduino IDE:



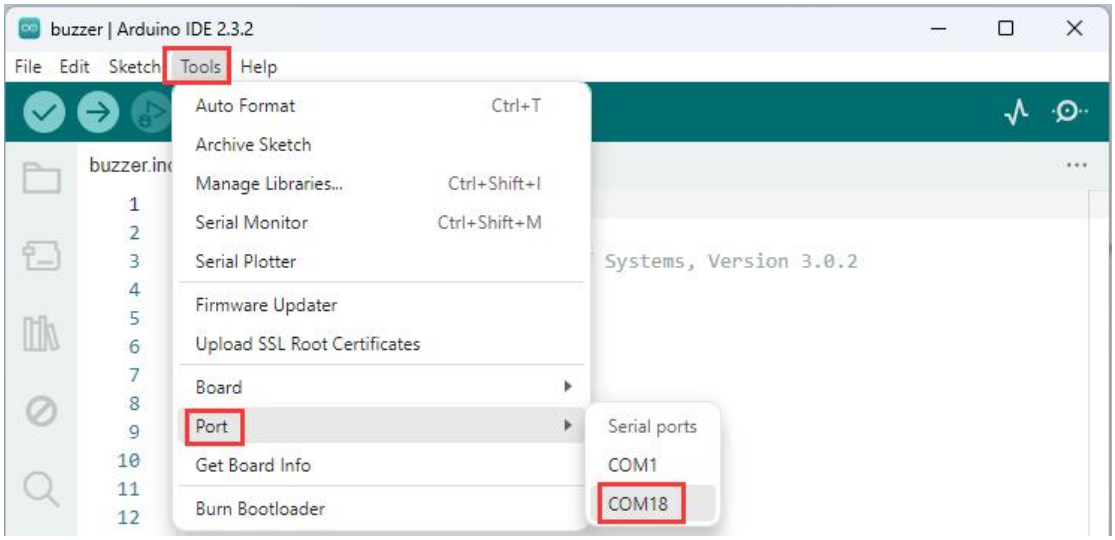
Select board type（ESP32C3 Dev Module）：



Use the USB cable to connect the PC and eArm, as shown below:



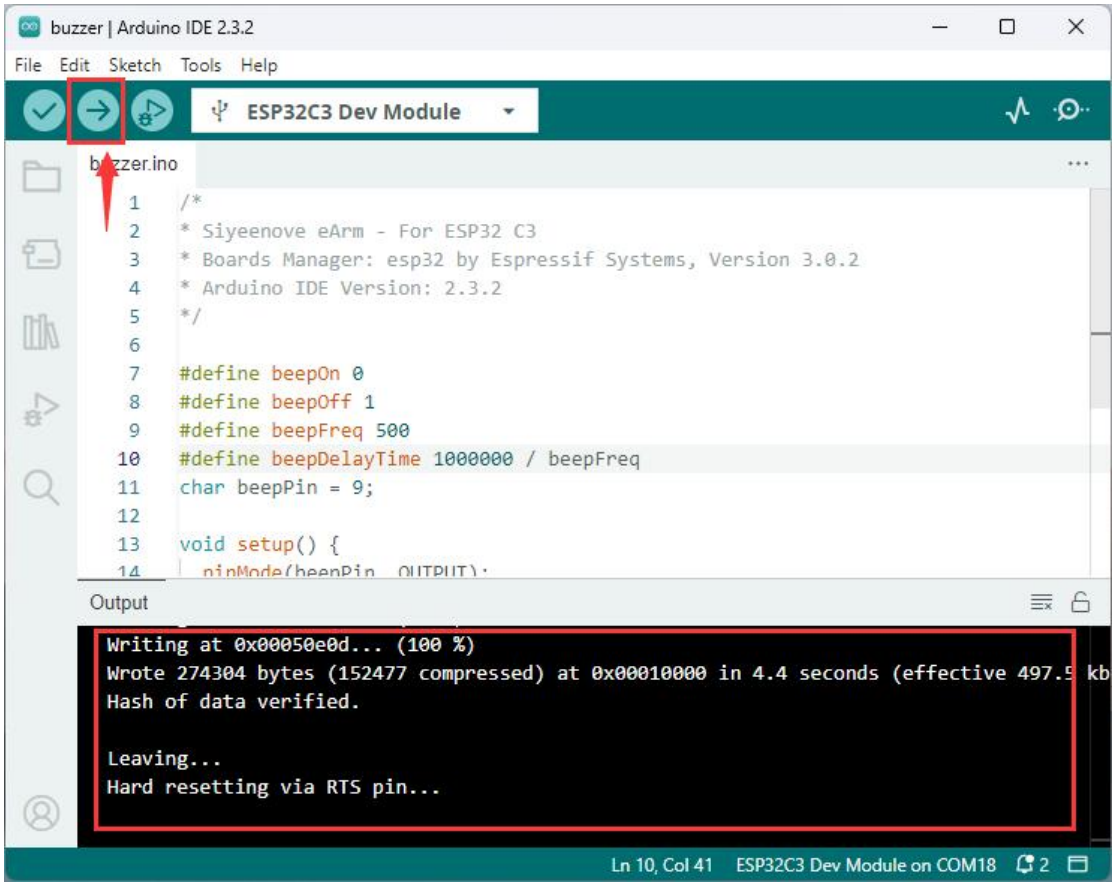
Select Port:



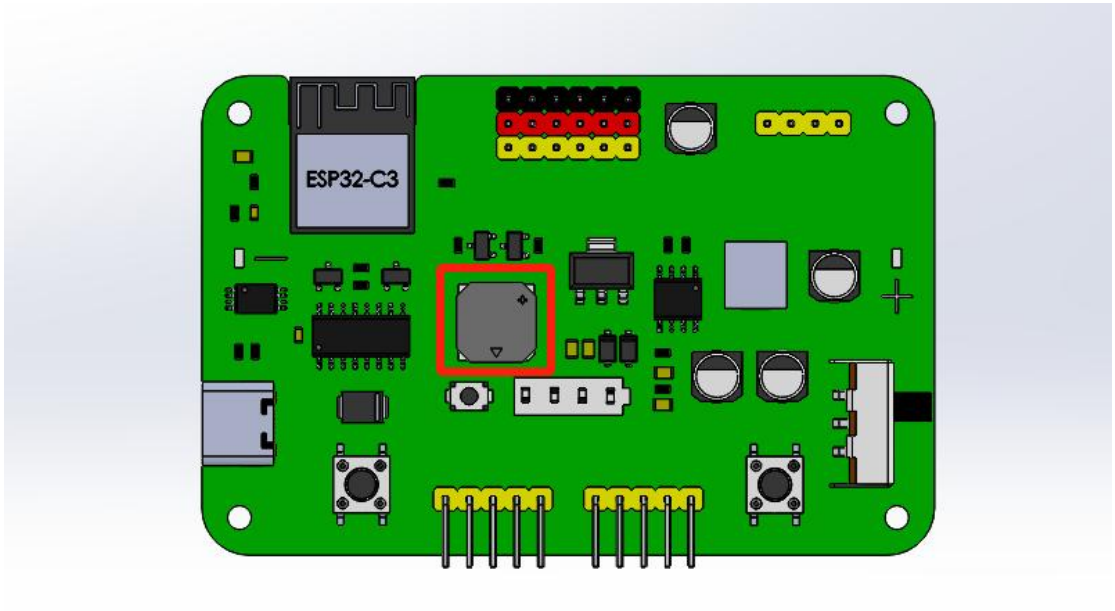
Note: The port in the above picture is COM18. Your computer may have a different numbered port. Please select it according to the port assigned by the device manager on your computer. Your computer must have the CH340 driver installed for the port number to be displayed in the device manager.

53

Click to upload the code：



After the code is uploaded successfully, the buzzer on the ESP board will keep making sounds:

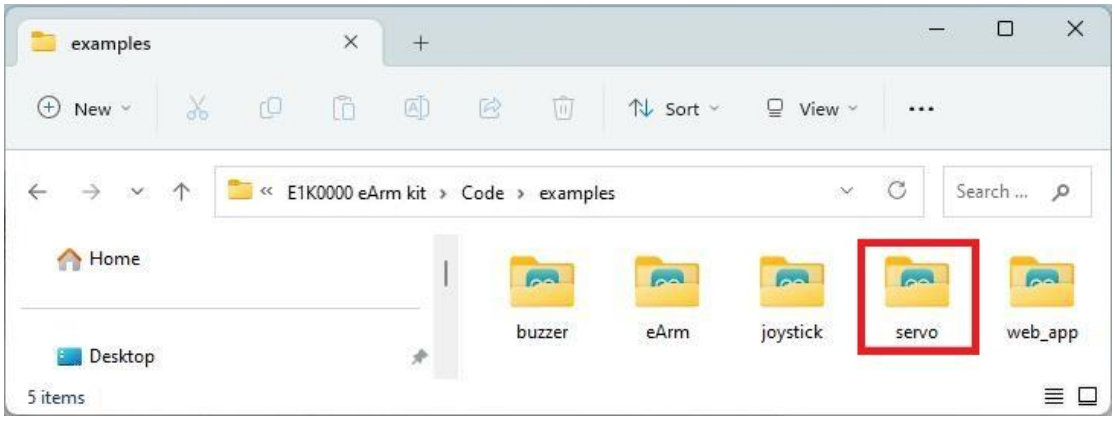


Code Explanation：

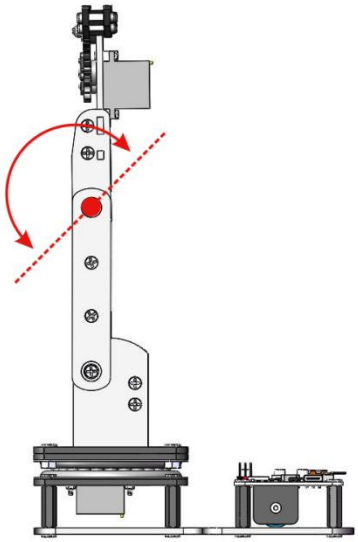
#define beepOn 0	Define a constant "beepOn" with a value of 0.
#define beepDelayTime 1000000 / beepFreq	Define a constant "beepDelayTime" with a value of: 1000000 / beepFreq.
char beepPin = 9;	Define a character variable "beepPin" with a value of 9, mapped to the control pin of the buzzer.
void setup() { ... }	Built-in function of Arduino IDE, only runs once when powered on.
pinMode(beepPin, OUTPUT);	Set pin 9 to output mode.
digitalWrite(beepPin, beepOff);	Pin 9 outputs a high level to turn off the buzzer.
void loop() { ... }	Built-in function of Arduino IDE, runs in a loop all the time when powered on.
delayMicroseconds(beepDelayTime);	Built-in function of Arduino IDE, delay in microseconds.

4.5.2 To drive a servo

Please refer to section 4.3.1 to upload the "servo" code to the esp32 control board:



After the code is uploaded successfully, the eArm's servo C will swing 0-180, 180-0 in a cycle:



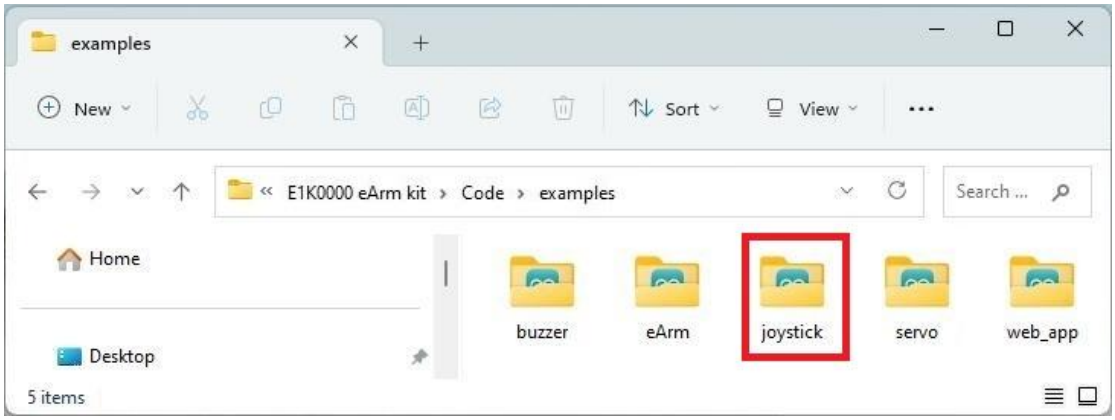
Code Explanation：

#include <ESP32Servo.h>	Include an ESP32 servo driver library.
Servo Aservo;	Define a servo class to drive Servo.
ESP32PWM::allocateTimer(0);	Allow the servo driver library to use timer 0.
Aservo.setPeriodHertz(50);	Set the pulse drive frequency of the driving servo Aservo to 50 Hz.
Aservo.attach(AservoPin, 500, 2400);	Set the driving servo Aservo to use the AservoPin pin to drive, the latest pulse width is 500 microseconds, and the maximum pulse width is 2400 microseconds.
Aservo.write(aServoAngle);	Set the driving servo Aservo to rotate to the angle aServoAngle.
for(statement1;condition;statement2){ … }	For loop statement, each time statement1 is executed, it determines whether the condition is true. If so, the statements in the curly braces are executed, otherwise the loop is interrupted; Then statement2 is executed, and the condition is determined again, and the loop is repeated.

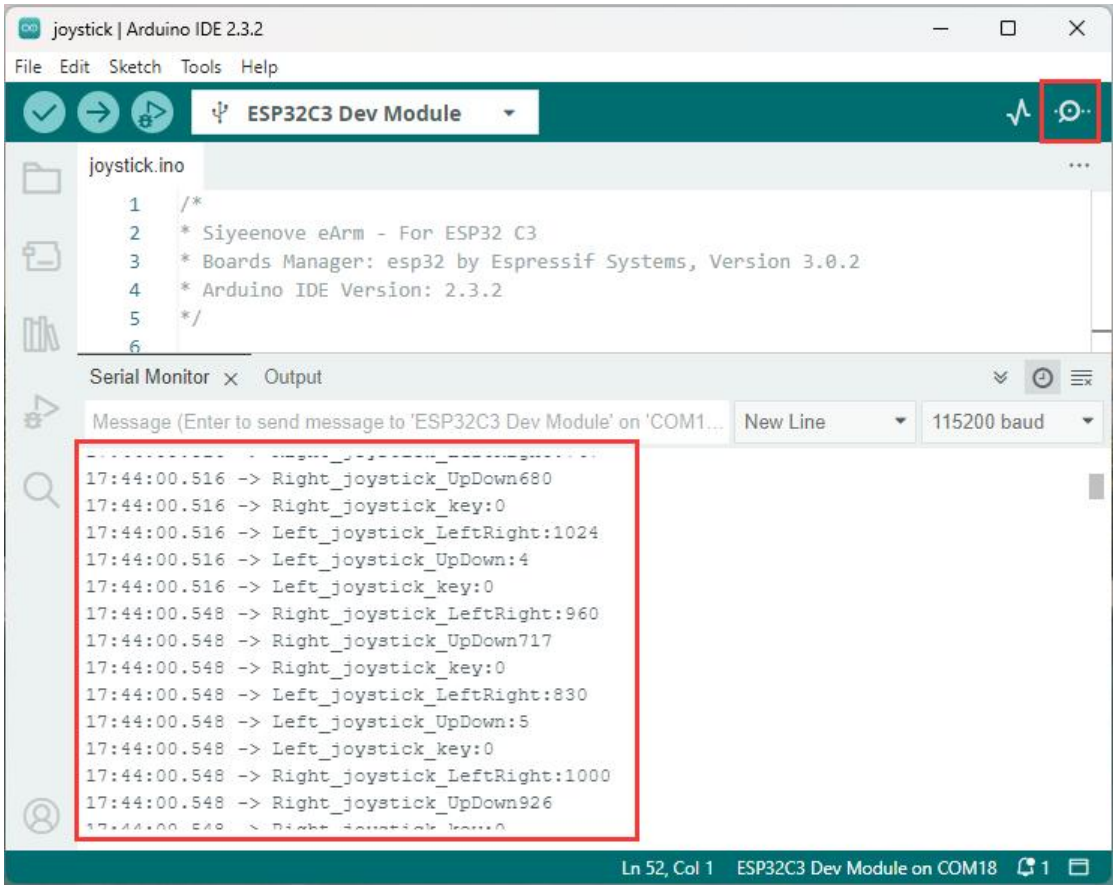
56

4.5.3 Read the value of the joystick

Please refer to section 4.3.1 to upload the "joystick" code to the esp32 control board:



Then open the serial port monitor of Arduino IDE, shake the handle left, right, up, down, or press the joystick down, and the serial port will print the joystick value:

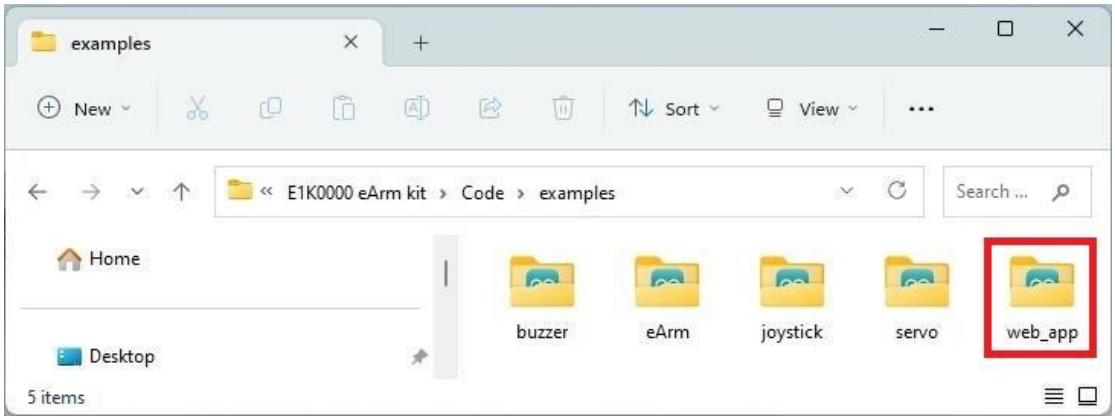


Code Explanation：

Serial.begin(115200);	Set the serial port baud rate to 115200
pinMode(left_joystick_KeyPin, INPUT);	Set the key pin of the left joystick to "left_joystick_KeyPin" and the mode to output.
pinMode(right_joystick_KeyPin, INPUT_PULLUP);	Set the key pin of the right joystick to "right_joystick_KeyPin", the mode to output, and with a pull-up resistor.
Serial.print("Left_joystick_LeftRight:");	The serial port prints a string without line breaks.
Serial.println(analogRead(left_joystick_LeftRightPin));	The serial port prints analog values and wraps.
analogRead(left_joystick_LeftRightPin);	Read the value of the left joystick shaking left and right.
delay(1000);	Delay 1000 milliseconds.

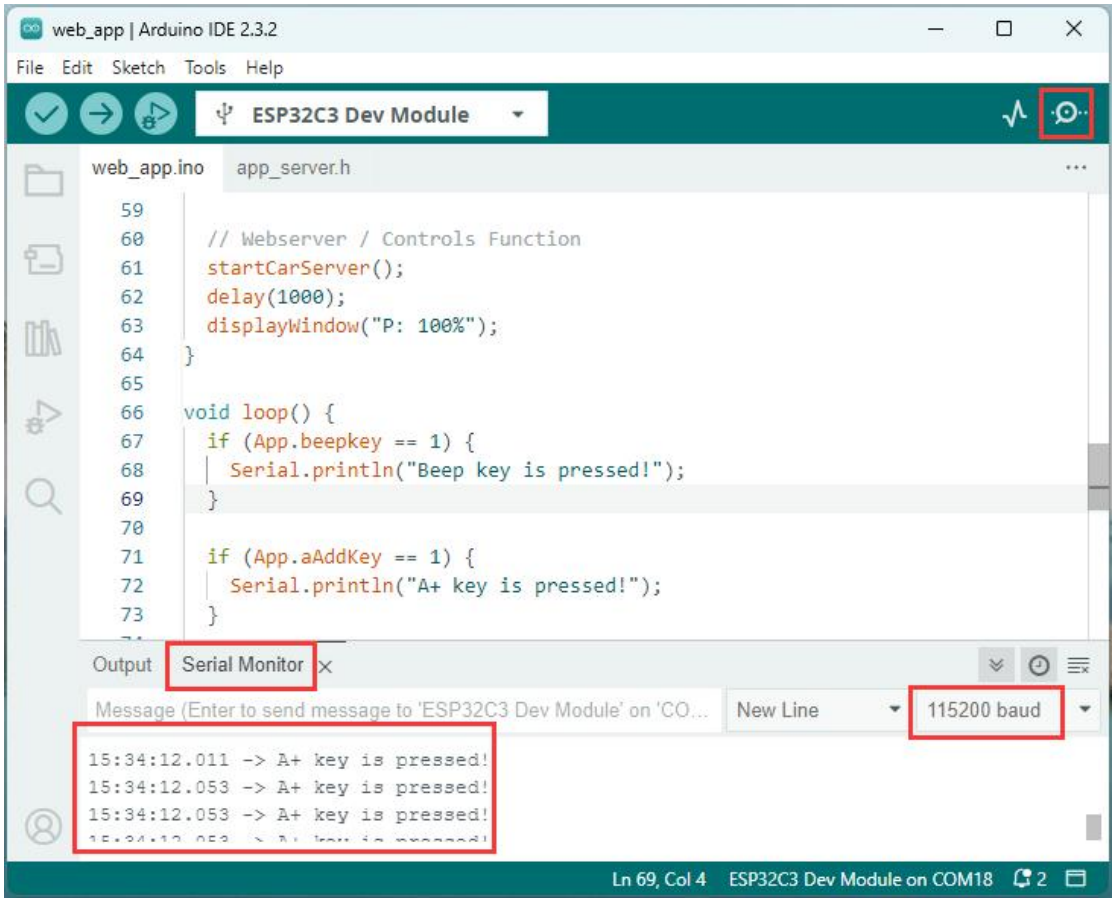
4.5.4 Web APP

Please refer to section 4.3.1 and upload the "web_app" code to the esp32 control board:



Refer to "Section 3.3" to connect "eArm" to wifi and use the web app, then open the serial monitor of Arduino IDE and click the button of the web app. The serial monitor will print the string corresponding to the button:

58



Code Explanation:

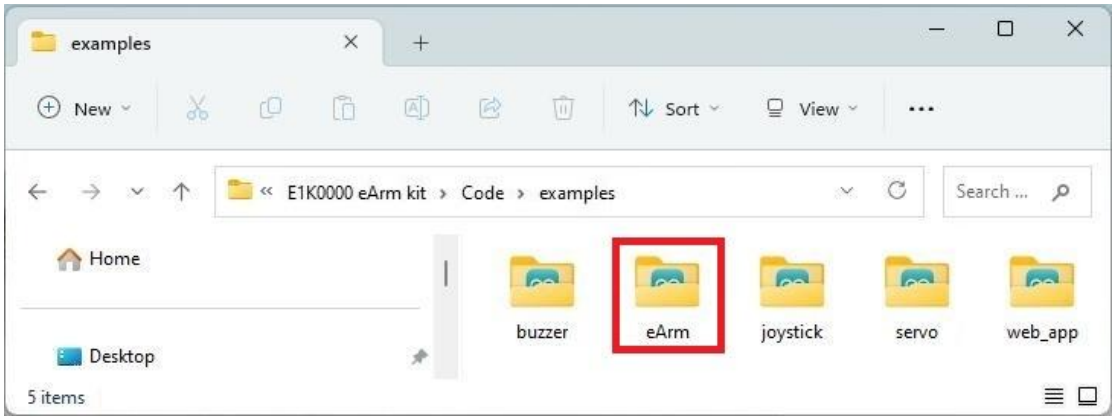
#include "app_server.h"	Include the "app_server.h" header file from the same folder.
bool ap = 1;	Define a Boolean variable "ap" and assign it a value of 1.
const char* ssid = "eArm";	Define a constant character pointer and assign a value to it.
int channel = 11;	Define an integer variable "channel" and assign it a value of 11.
WiFi.mode(WIFI_STA);	Set Wi-Fi to "station" mode.
WiFi.begin(ssid, password);	Set the Wi-Fi name and password, and start Wi-Fi.
while(condition) {...}	While conditional loop statement, if the condition is true, the loop executes the statements in the curly braces; otherwise, the loop is interrupted.
WiFi.status();	Read the wifi connection status.
WiFi.localIP();	Read the wifi IP address in "station" mode.
WiFi.mode(WIFI_AP);	Set the wifi to AP mode.
WiFi.softAP(ssid, password, channel, hidden, maxconnection);	In AP mode, set the wifi name, password, channel, whether to hide the name, and the

	maximum number of connections.
WiFi.softAPIP();	Read the IP address of the wifi in AP mode.
startCarServer();	Start the web server.
displayWindow("P: 100%");	Display the string on the front end of the web page.
if(condition){ ... }	If the condition is true, the statements in the curly braces are executed; otherwise, they are not executed.
App.beepkey	The state variable value of the buzzer button on the front end of the web page.

4.5.5 To Control the Robotic Arm

This sample code is also the source code of our factory burned firmware. Upload the code to the control board according to the following method, and you can also restore the factory function.

Please refer to section 4.3.1 to upload the "eArm" code to the esp32 control board:



For how to control the robot, please refer to "Chapter 3".

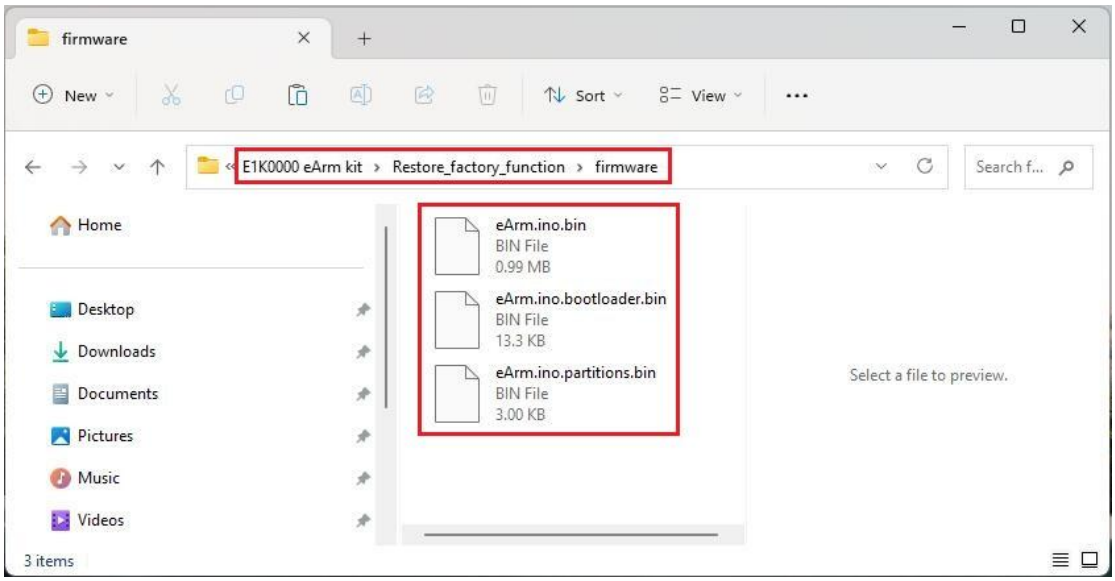
Code Explanation:

TaskHandle_t TASK_HandleOne = NULL;	Create a thread handle.
xTaskCreate(...);	Initialize the thread.
void TASK_ONE(void* param) { ... }	Thread function, which runs in parallel with the loop() function.

5. Factory Reset Function

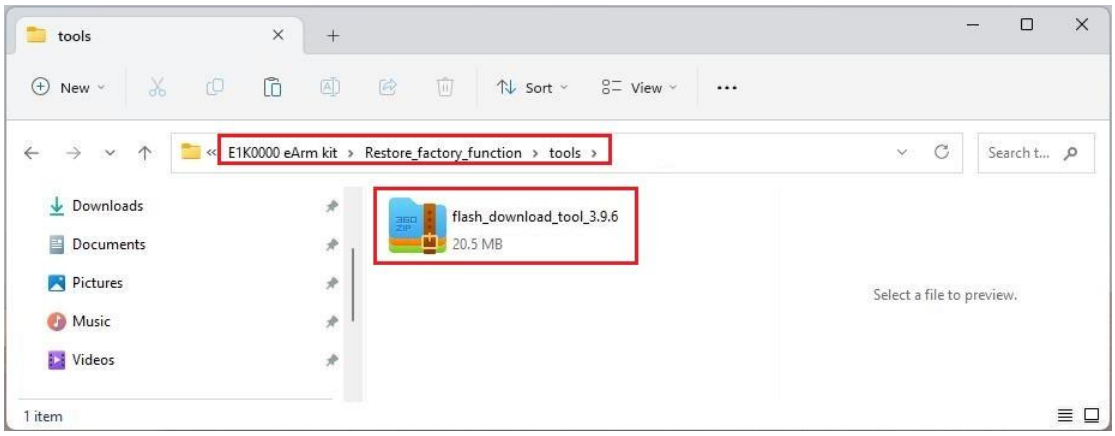
5.1 firmware

We provide factory firmware files, which are saved in the following files:



5.2 Burning tool

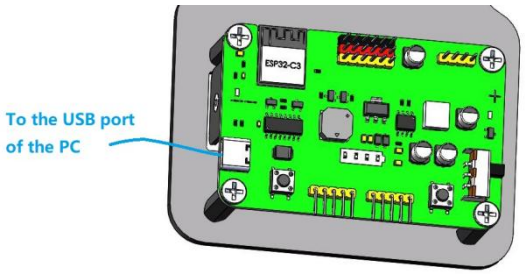
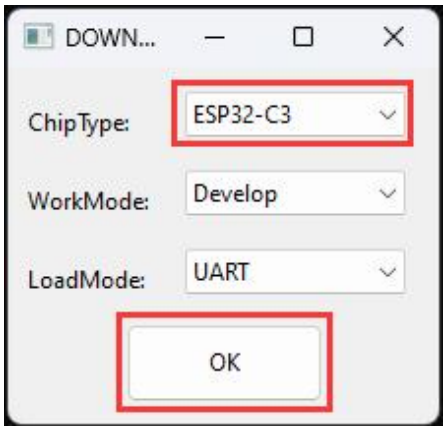
The burning tool is in the following files:



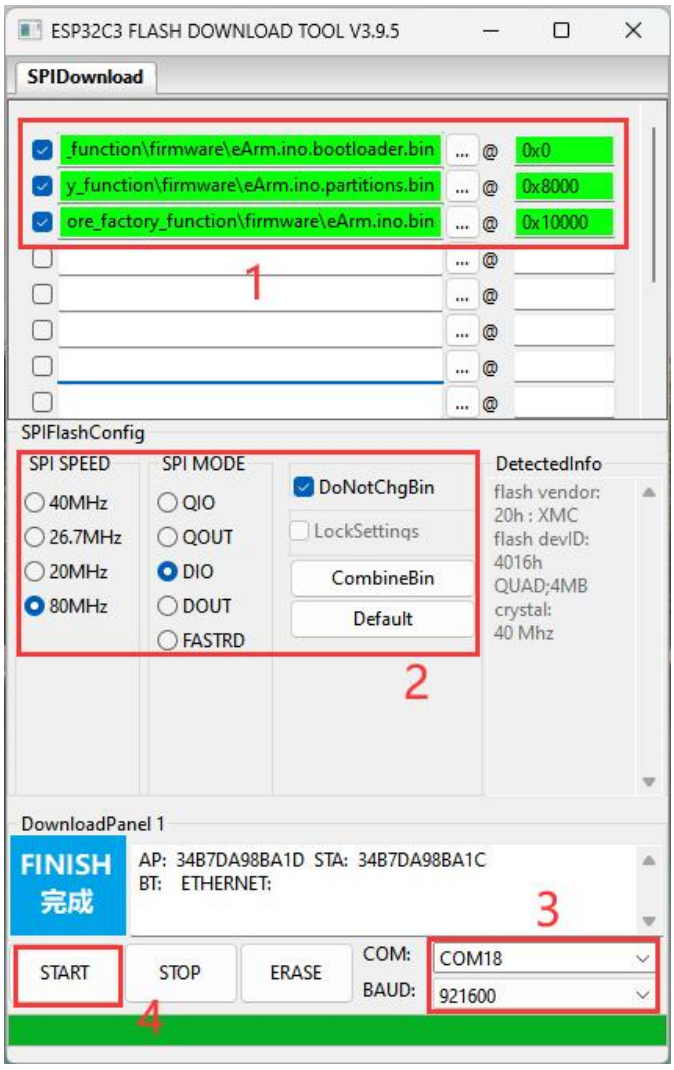
You can also download the latest version of the burning tool from the official website: https://www.espressif.com.cn/zh-hans/support/download/other-tools

61

5.3 How to burn firmware

Use a USB cable to connect the PC and eArm, as shown below:	Start the burning tool:
	

Select the firmware we provided and fill in the burning address correctly:



You must have installed the CH340 driver and turn on the switch of the ESP32 board, otherwise the COM port will not be found!

62

6. QA

6.1 Unable to recognize USB serial ports

☛Do you use a USB cable with data communication function?

☛Does the USB cable connect well?

☛Is the CH340 USB driver installed?

6.2 Robotic arm doesn't work

☛Please make sure that the 18650 lithium battery is connected!

☛Is the battery electricity sufficient?

☛Whether to turn on the power switch during installation to initialize the angle of the servo.

6.3 Other problems

☛Please check whether the assembly is correct?

☛Please check whether the battery power is sufficient?

☛Please check whether the battery used in accordance with the specifications?

7. Contact us

If you encounter technical problems or want to share your ideas and opinions with us, please contact us at any time.

✉ siyeenove@outlook.com

http://siyeenove.com