



ACEBOTT

O Tutorial do Braço Robótico

Prefácio

Nossa Empresa

ACEBOTT STEM Education Tech Co., Ltd

Fundada no Vale do Silício da China em 2013, a ACEBOTT é uma líder em soluções para educação STEM. Temos uma equipe de 150 pessoas, incluindo membros das áreas de pesquisa e desenvolvimento, vendas e logística. Nosso objetivo é fornecer produtos e serviços de educação STEM de alta qualidade para nossos clientes. Trabalhamos em conjunto com especialistas em educação STEM e nossos parceiros comerciais para desenvolver produtos STE de sucesso. Nossa fábrica própria também oferece serviços de CEM para nossos clientes, incluindo personalização de logotipo nas embalagens dos produtos e placas de circuito impresso(PCB).

Nosso Tutorial

Este curso e o kit de aprendizado de braço robótico foram projetados para crianças e adolescentes com 8 anos ou mais, com o objetivo de proporcionar um entendimento mais aprofundado sobre a placa de desenvolvimento ESP32, o conhecimento sobre braços robóticos e hardware eletrônico. Se você deseja aprender sobre braços robóticos, este kit pode fornecer conhecimento e etapas práticas para ajudá-lo a montar seu próprio braço robótico.

Através deste kit, você pode:

- 1.Aprender a usar de forma eficaz a placa de desenvolvimento ESP32, incluindo baixar códigos, entender suas características e programar no Arduino IDE.
- 2.Construir uma base sólida em programação com base na linguagem C, já que o ESP32 utiliza uma linguagem simplificada de C/C++ para controlar circuitos e sensores.

- 3.Explorar o funcionamento do módulo servo motor e entender como vários servos trabalham em conjunto em projetos de braço robótico.
- 4.Seguir o tutorial para montar passo a passo seu próprio braço robótico usando o kit ACEBOTT, aprimorando suas habilidades de maker.
- 5.Implementar funções básicas no projeto do braço robótico, como controle por joystick, aprendizado por demonstração, controle via página web e controle por aplicativo.
- 6.Desenvolver uma compreensão completa dos conceitos de braço robótico, preparando-se para estudos mais avançados no futuro.

Em resumo, o braço robótico ACEBOTT é um kit de aprendizado baseado em ESP32 projetado especialmente para iniciantes. Com este kit, os usuários podem obter uma compreensão abrangente sobre o funcionamento da placa de controle e dos servos no braço robótico. Seguindo os tutoriais fornecidos no kit, estudantes de diferentes faixas etárias podem adquirir conhecimentos valiosos sobre braços robóticos e construir com sucesso seus próprios projetos.

Atendimento ao Cliente

A ACEBOTT é uma empresa de tecnologia educacional STEM dinâmica e em rápido crescimento, que se empenha em oferecer produtos excelentes e serviços de qualidade que atendam às suas expectativas. Valorizamos seu feedback e incentivamos você a nos enviar um e-mail para support@acebott.com com quaisquer comentários ou sugestões que possa ter.

Nossos engenheiros experientes estão dedicados a resolver prontamente quaisquer problemas ou dúvidas que você possa ter sobre nossos produtos. Garantimos uma resposta dentro de 24 horas durante os dias úteis.

Siga-nos

Escaneie os códigos QR para nos seguir e obter assistência técnica e as últimas novidades.

Temos uma grande comunidade que é muito útil para resolver problemas, e também contamos com uma equipe de suporte pronta para responder a qualquer pergunta.



Código QR do grupo ACEBOTT no FB



Código QR do YouTube

Conteúdo

Lição 1: Compreensão do Hardware e Instalação do Software	1
I.Compreensão do Hardware	2
II.Instalação do Software	3
III.Compreendendo os Servos	32
Lição2 Montagem do Braço Robótico	36
I.Lista de Peças	36
II.Lista de Componentes Estruturais	37
III.Passos de Montagem	38
Lição 3: Controle do Braço Robótico com Joystick	62
I.Controle dos Servos	62
II.Compreendendo o Módulo Joystick	63
III.Movimentos Básicos do Braço Robótico Controlado pelo Joystick	65
IV.Tarefas de Expansão	68
Lição 4: As Coordenadas Espaciais do Braço Robótico	69
I.O Sistema de Coordenadas Cartesianas	69
II.Sistema de Coordenadas da Junta	70
III.Cinemática Direta e Inversa	71
IV.Diagrama de Coordenadas do Braço Robótico	73
V.Instruções de Calibração do Braço Robótico	74
VI.Movendo Pontos de Coordenadas Espaciais	75
Lição 5: Empilhamento com Braço Robótico	79
I.Programa de Paletização com Braço Robótico	80
II.Tarefas de Expansão	82
Lição 6: Ensino e Aprendizado do Braço Robótico	83
I.Programa de Ensino	83
II.Tarefas de Expansão	85
Lição 7: Controle Web do Braço Robótico	87
I.Programa de Controle Web	87
II.Login na Página Web	88
III.Tarefas de Expansão	90
Lição 8: Controle do Braço Robótico pelo APP	92
I.Download do APP	92
II.Controle do Braço Robótico pelo APP	94

Lição 1: Compreensão do Hardware e Instalação do Software

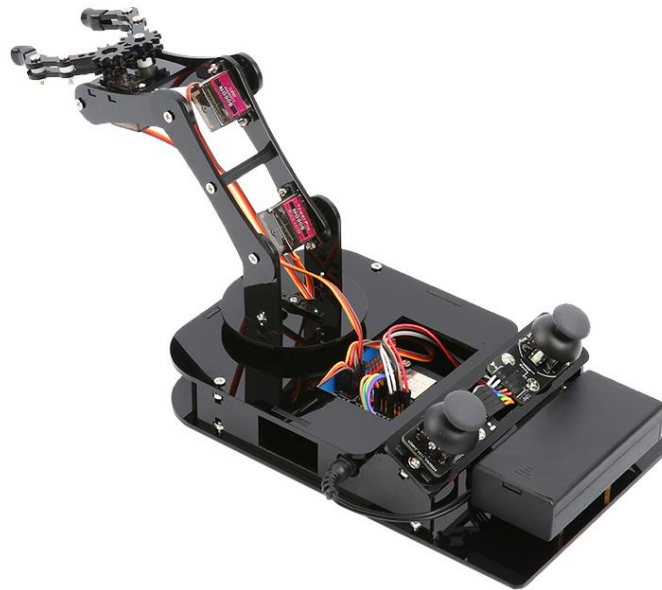
Os braços robóticos são dispositivos mecânicos automatizados amplamente utilizados no campo da robótica. O braço é um componente crucial do mecanismo de execução do robô, projetado para transportar a peça de trabalho agarrada até uma posição determinada. No setor industrial, a aplicação de braços robóticos pode substituir os seres humanos em tarefas de produção monótonas e repetitivas ou em operações de processamento em ambientes perigosos e severos.

Um braço robótico é composto por três partes básicas: o corpo principal, o sistema de acionamento e o sistema de controle. O corpo principal inclui principalmente o chassi, o ombro, o cotovelo e o efector final. Dependendo dos diferentes cenários de aplicação, o corpo principal do braço robótico pode ser dividido em configurações de quatro eixos, cinco eixos, seis eixos ou até configurações multi-eixos. O sistema de acionamento inclui dispositivos de potência e mecanismos de transmissão, sendo os componentes principais o redutor e o motor de servo, que acionam o braço robótico para realizar as ações correspondentes. O sistema de controle emite sinais de comando para o sistema de acionamento de acordo com o programa de entrada, controlando o movimento do braço robótico.

O braço robótico deste tutorial é um braço robótico de quatro eixos. O primeiro eixo é o servo do chassi, o segundo eixo é o servo do ombro, o terceiro eixo é o servo do cotovelo, e o quarto eixo é o servo da garra.

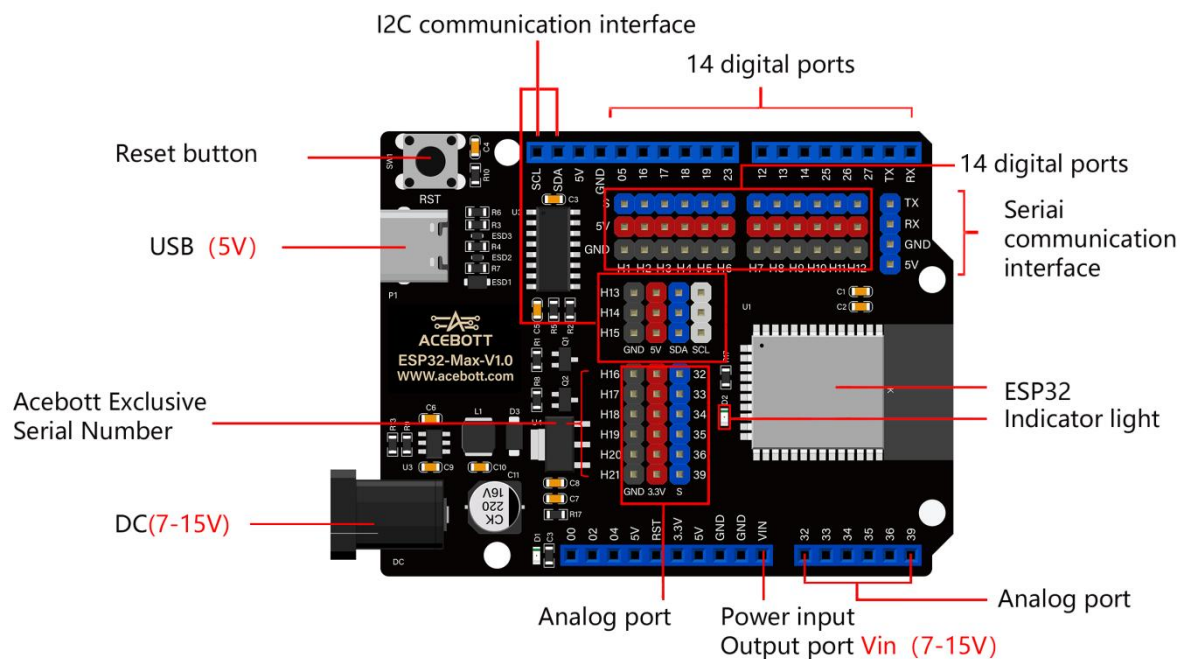
O braço robótico deste tutorial utiliza uma placa controladora ESP32, um controlador amplamente utilizado, e a programação é feita na IDE Arduino. Os métodos de controle incluem controle por joystick, controle via web e controle por aplicativo.

Como o braço robótico é controlado especificamente? Vamos continuar com o tutorial e aprender juntos.



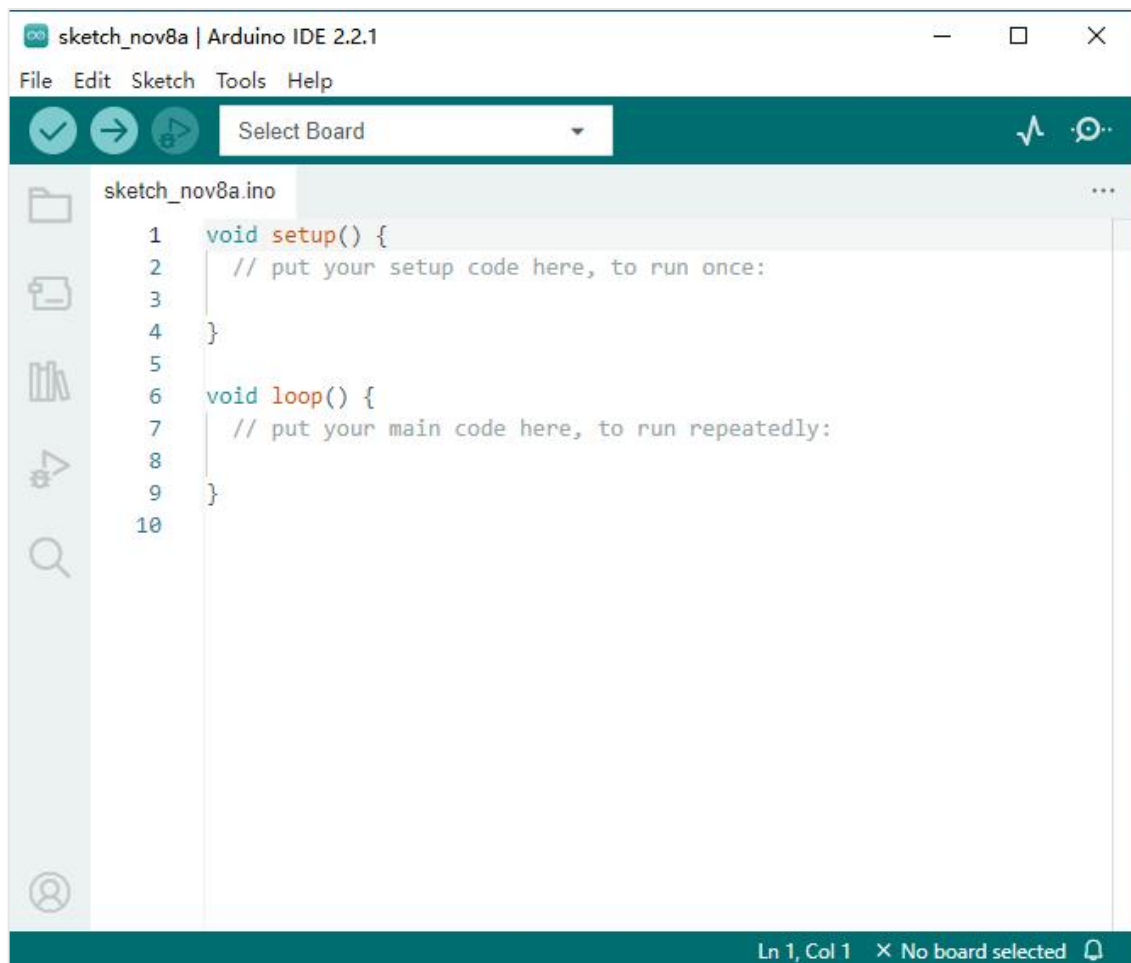
I. Compreensão do Hardware

A placa controladora ESP32 é um microcontrolador de baixo consumo de energia e alto desempenho, ideal para o desenvolvimento de IoT (Internet das Coisas). Ela possui um processador de núcleo duplo operando a 240 MHz, 520 KB de RAM e 4 MB de memória flash. Inclui módulos Wi-Fi e Bluetooth 4.2 integrados para comunicação sem fio. Com 34 pinos GPIO, a ESP32 oferece suporte para conectividade e controle de diversos periféricos.



II.Instalação do Software

No projeto do braço robótico, usaremos principalmente a IDE Arduino como o software de programação. A IDE Arduino é uma plataforma de programação de código aberto compatível com diversas placas, como Arduino UNO, ESP32, ESP8266, STM32 e outras. Usando a IDE Arduino, você simplesmente escreve o código do seu programa dentro da IDE e, em seguida, o envia para a placa controladora. O programa instruirá a placa controladora sobre as ações que devem ser realizadas.



1.Instalando a IDE Arduino

Primeiro, abra o site oficial onde a IDE Arduino pode ser baixada:

<https://www.arduino.cc/en/Main/Software>

De acordo com o sistema operacional do usuário, escolha a versão correspondente do software.

(1) Instalação no sistema Windows

①Clique com o mouse na posição mostrada na figura.

Atenção: Devido a possíveis problemas de compatibilidade com a versão mais recente, recomenda-se instalar a versão 2.2.1 da IDE Arduino.



Arduino IDE 2.2.1

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

DOWNLOAD OPTIONS

- Windows** Win 10 and newer, 64 bits
- Windows** MSI installer
- Windows** ZIP file
- Linux** AppImage 64 bits (X86-64)
- Linux** ZIP file 64 bits (X86-64)
- macOS** Intel, 10.14: "Mojave" or newer, 64 bits
- macOS** Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

Link para a versão 2.2.1:<https://github.com/arduino/arduino-ide/releases>

2.2.1

Changed

- Bump Arduino Firmware Uploader dependency to 2.4.1 (#2206)

Fixed

- Fix missing translations for русский and Türkçe (#2201)

Full Changelog: [2.2.0...2.2.1](#)

Assets 14

arduino-ide_2.2.1_Linux_64bit.AplImage	175 MB	Aug 31, 2023
arduino-ide_2.2.1_Linux_64bit.zip	175 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_64bit.dmg	179 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_64bit.zip	175 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_arm64.dmg	171 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_arm64.zip	166 MB	Aug 31, 2023
arduino-ide_2.2.1_Windows_64bit.exe	137 MB	Aug 31, 2023
arduino-ide_2.2.1_Windows_64bit.msi	147 MB	Aug 31, 2023
arduino-ide_2.2.1_Windows_64bit.zip	179 MB	Aug 31, 2023
stable-linux.yml	396 Bytes	Sep 3, 2023
stable-mac.yml	853 Bytes	Aug 31, 2023
stable.yml	365 Bytes	Aug 31, 2023

②Selezione JUSTDOWNLOAD.

Download Arduino IDE & support it's progress

Since the release 1.x release in March 2015, the Arduino IDE has been downloaded **77,917,375** times — impressive! Help its development with a donation.

\$3

\$5

\$10

\$25


\$50

Other

CONTRIBUTE AND DOWNLOAD

or

JUST DOWNLOAD



Learn more about donating to Arduino.

Stay in the Loop: Join Our Newsletter!

As a beginner or advanced user, you can find inspiring projects and learn about cutting-edge Arduino products through our **weekly newsletter!**


☐ I confirm to have read the [Privacy Policy](#) and to accept the [Terms of Service](#) *

☐ I would like to receive emails about special deals and commercial offers from Arduino.

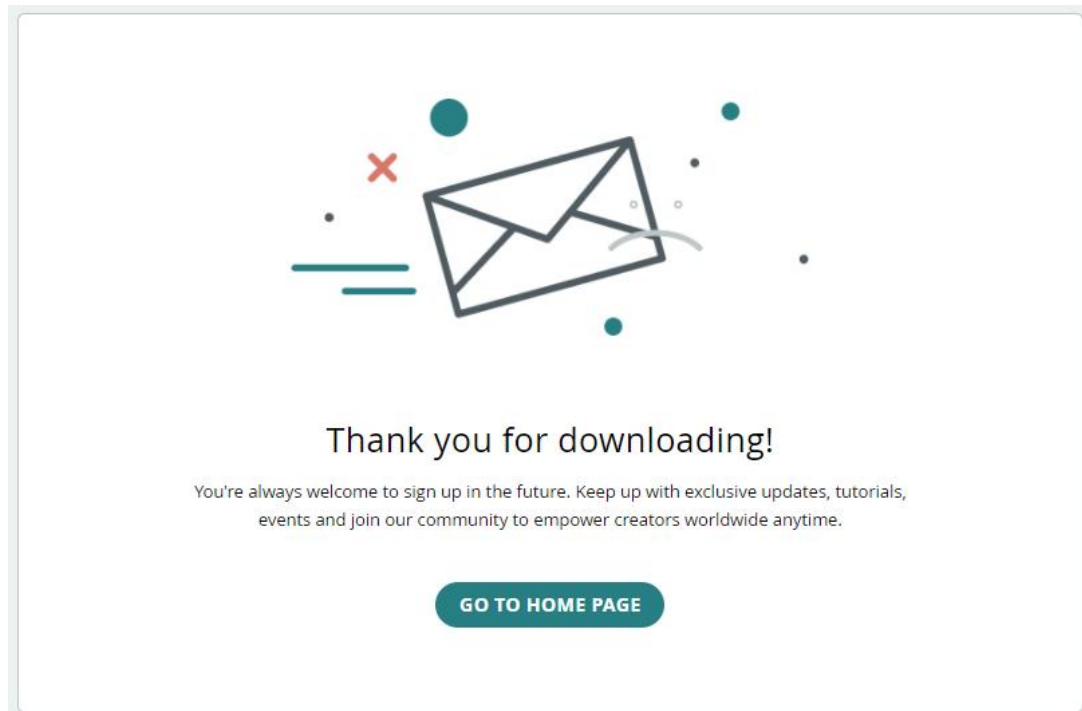
SUBSCRIBE & DOWNLOAD

or

JUST DOWNLOAD



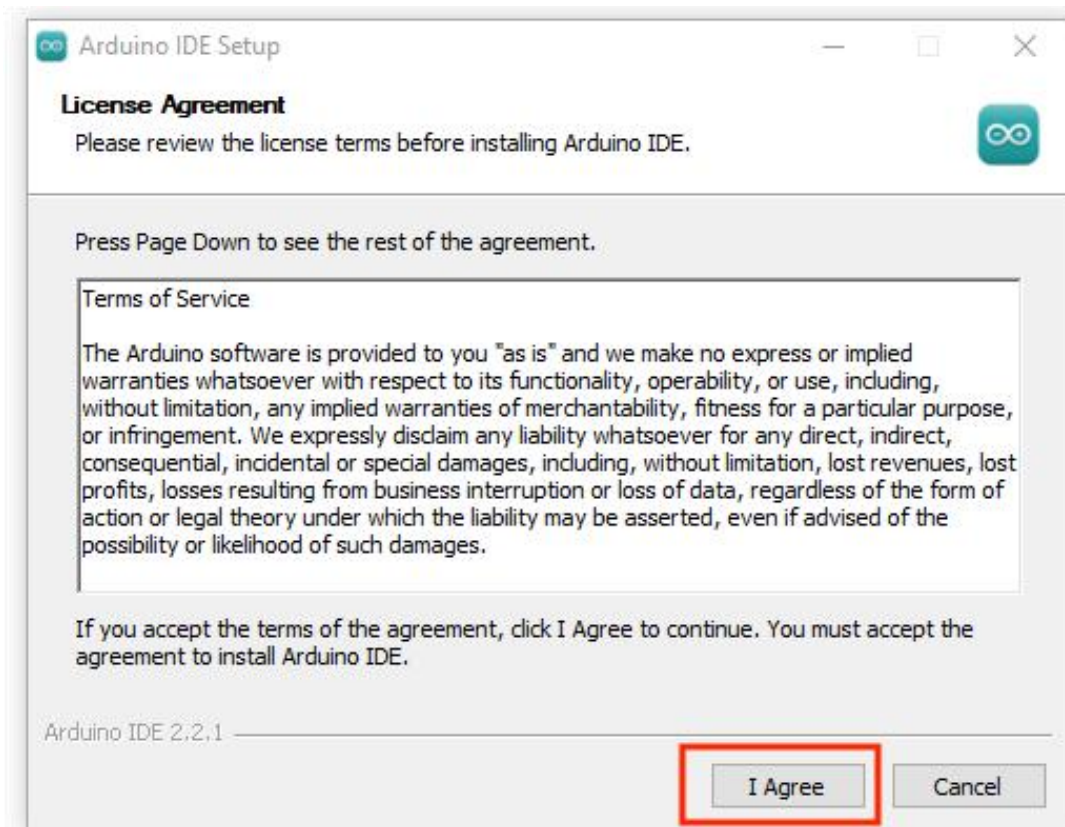
③Quando a seguinte tela aparecer, isso indicará que a IDE Arduino está sendo baixada.



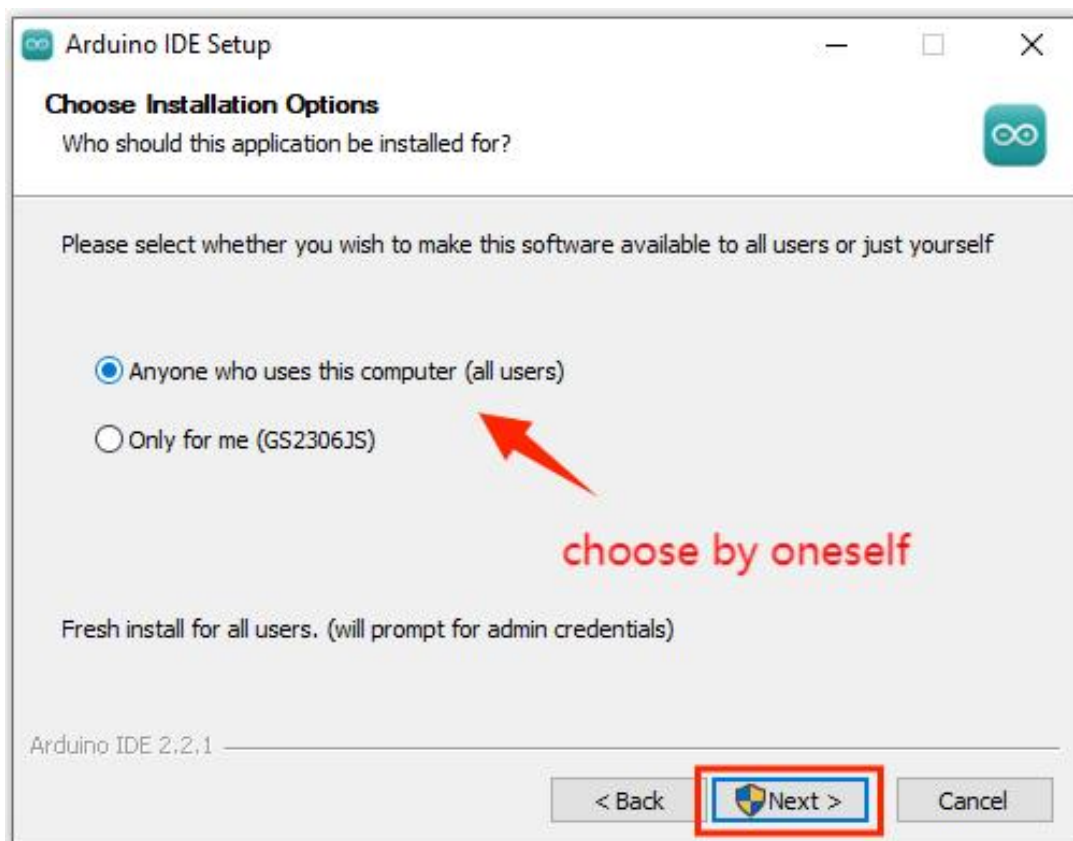
④ Quando o download for concluído, o ícone do arquivo aparecerá. Clique em Instalar software.



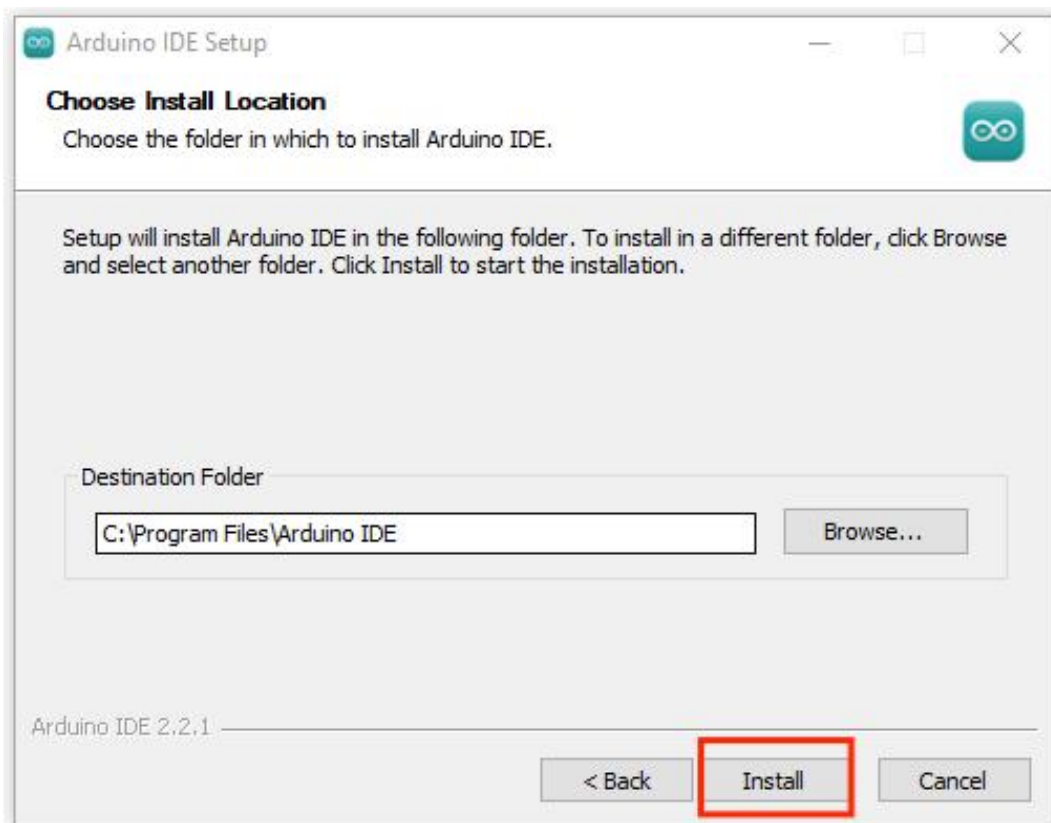
⑤Após a instalação, a seguinte tela aparecerá. Selecione "I Agree".



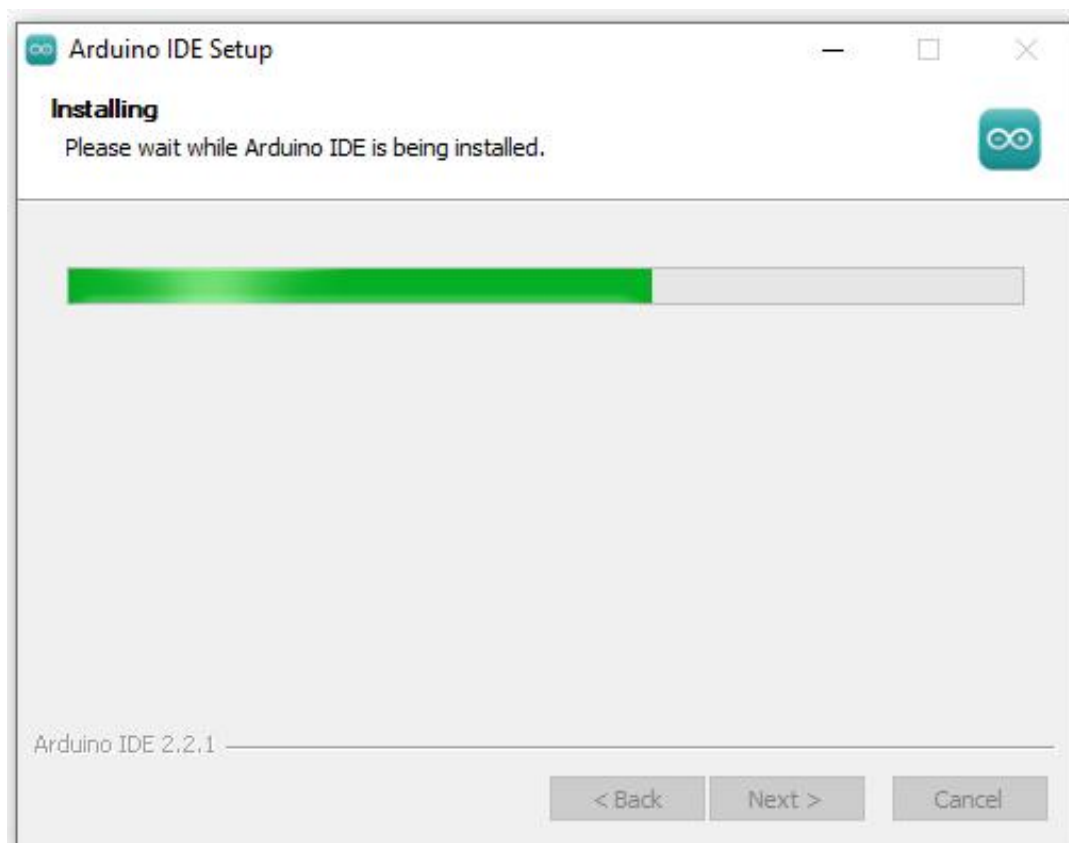
⑥Após selecionar "I Agree", a seguinte tela aparecerá. Selecione "Next".



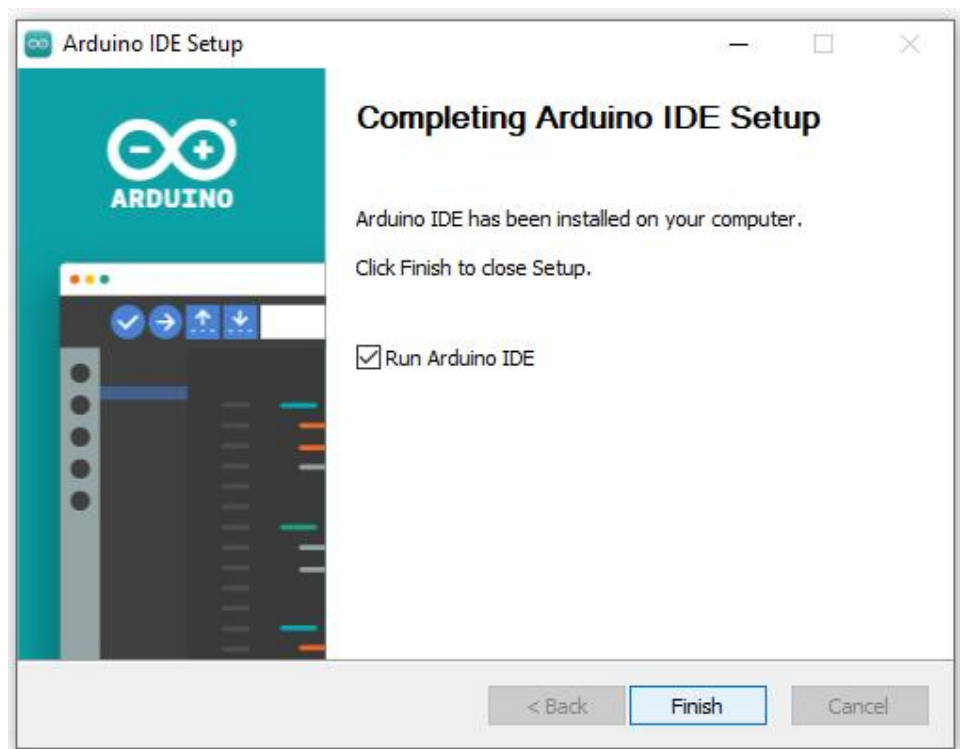
⑦ Seleccione "Next" e a seguinte tela aparecerá. Seleccione "Install".



⑧ Software da IDE Arduino instalado.



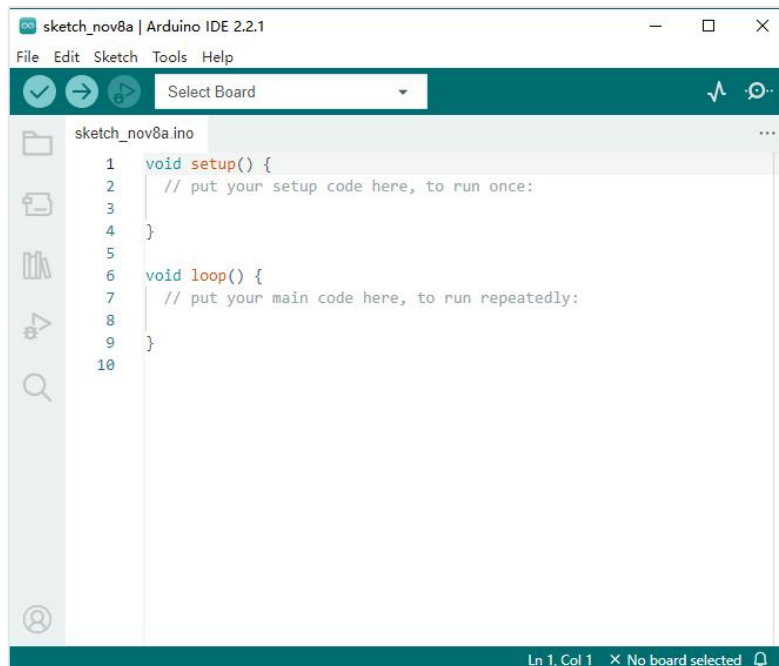
⑨ Quando a instalação for concluída, clique em Concluir.



⑩ Após a instalação, um ícone de atalho da IDE Arduino aparecerá na sua área de trabalho.



⑪ Após abrir, a seguinte interface do software aparecerá.



(2) Instalação no Mac OS

①Clique com o mouse na posição mostrada na figura.

Atenção: Devido a possíveis problemas de compatibilidade com a versão mais recente, recomenda-se instalar a versão 2.2.1 da IDE Arduino.



Arduino IDE 2.2.1

The new major release of the Arduino IDE is faster and even more powerful! In addition to a more modern editor and a more responsive interface it features autocompletion, code navigation, and even a live debugger.

For more details, please refer to the [Arduino IDE 2.0 documentation](#).

Nightly builds with the latest bugfixes are available through the section below.

SOURCE CODE

The Arduino IDE 2.0 is open source and its source code is hosted on [GitHub](#).

DOWNLOAD OPTIONS

Windows Win 10 and newer, 64 bits
Windows MSI installer
Windows ZIP file

Linux AppImage 64 bits (X86-64)
Linux ZIP file 64 bits (X86-64)

macOS Intel, 10.14: "Mojave" or newer, 64 bits
macOS Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

Link para a versão 2.2.1:<https://github.com/arduino/arduino-ide/releases>

2.2.1**Changed**

- Bump Arduino Firmware Uploader dependency to 2.4.1 (#2206)

Fixed

- Fix missing translations for русский and Türkçe (#2201)

Full Changelog: [2.2.0...2.2.1](#)

▼ Assets

14

arduino-ide_2.2.1_Linux_64bit.AplImage	175 MB	Aug 31, 2023
arduino-ide_2.2.1_Linux_64bit.zip	175 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_64bit.dmg	179 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_64bit.zip	175 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_arm64.dmg	171 MB	Aug 31, 2023
arduino-ide_2.2.1_macOS_arm64.zip	166 MB	Aug 31, 2023
arduino-ide_2.2.1_Windows_64bit.exe	137 MB	Aug 31, 2023
arduino-ide_2.2.1_Windows_64bit.msi	147 MB	Aug 31, 2023
arduino-ide_2.2.1_Windows_64bit.zip	179 MB	Aug 31, 2023

②Selecione JUSTDOWNLOAD.


Download Arduino IDE & support it's progress

Since the release 1.x release in March 2015, the Arduino IDE has been downloaded **77,917,375** times — impressive! Help its development with a donation.

CONTRIBUTE AND DOWNLOAD

or

JUST DOWNLOAD



[Learn more about donating to Arduino.](#)

Stay in the Loop: Join Our Newsletter!

As a beginner or advanced user, you can find inspiring projects and learn about cutting-edge Arduino products through our **weekly newsletter!**

email *


☐ I confirm to have read the [Privacy Policy](#) and to accept the [Terms of Service](#) *

☐ I would like to receive emails about special deals and commercial offers from Arduino.

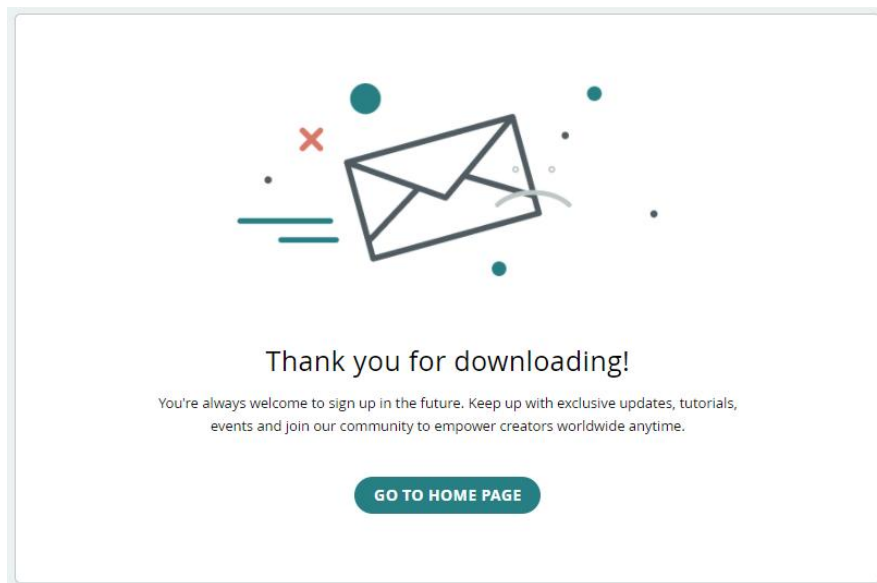
SUBSCRIBE & DOWNLOAD

or

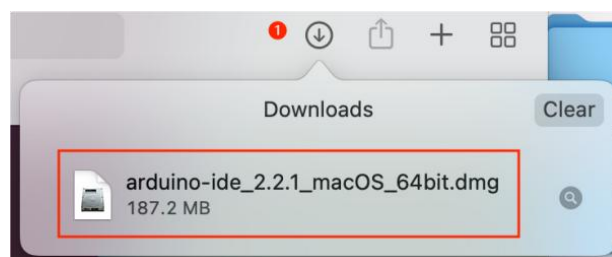
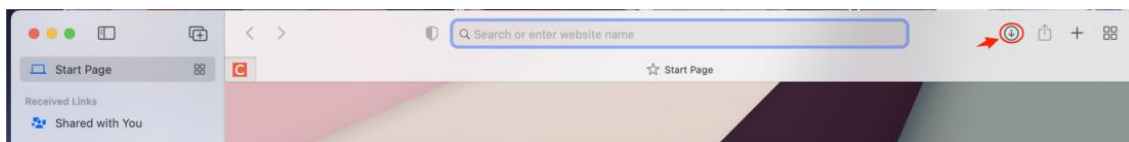
JUST DOWNLOAD



③Quando a seguinte tela aparecer, a IDE Arduino estará sendo baixada.



④Quando o download for concluído, clique no ícone de download no seu navegador e localize o instalador da IDE Arduino.



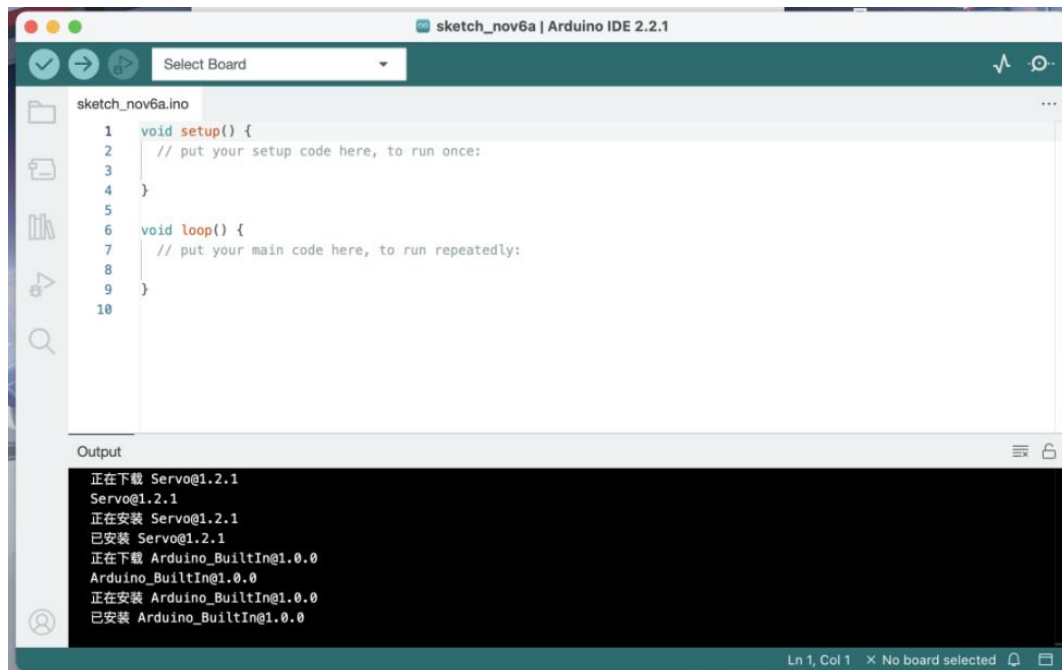
⑤Clique no pacote de instalação para que a tela de instalação apareça. Basta selecionar o ícone da IDE Arduino e movê-lo para a pasta Aplicativos para instalar o programa.



⑥Na área de trabalho, localize a IDE Arduino e abra-a.



⑦Após abrir, você verá a seguinte interface do software.



2.Instale o driver serial (ignore se já estiver instalado)

A porta serial é uma interface de comunicação do computador, geralmente usada para transmissão de dados entre o computador e outros dispositivos (como modems, sensores, impressoras, microcontroladores, etc.). A porta serial USB é a interface de comunicação mais comumente utilizada.

Geralmente, após a instalação do software da IDE Arduino pela primeira vez, é necessário instalar o driver da porta serial para transmitir o programa editado para a placa controladora.

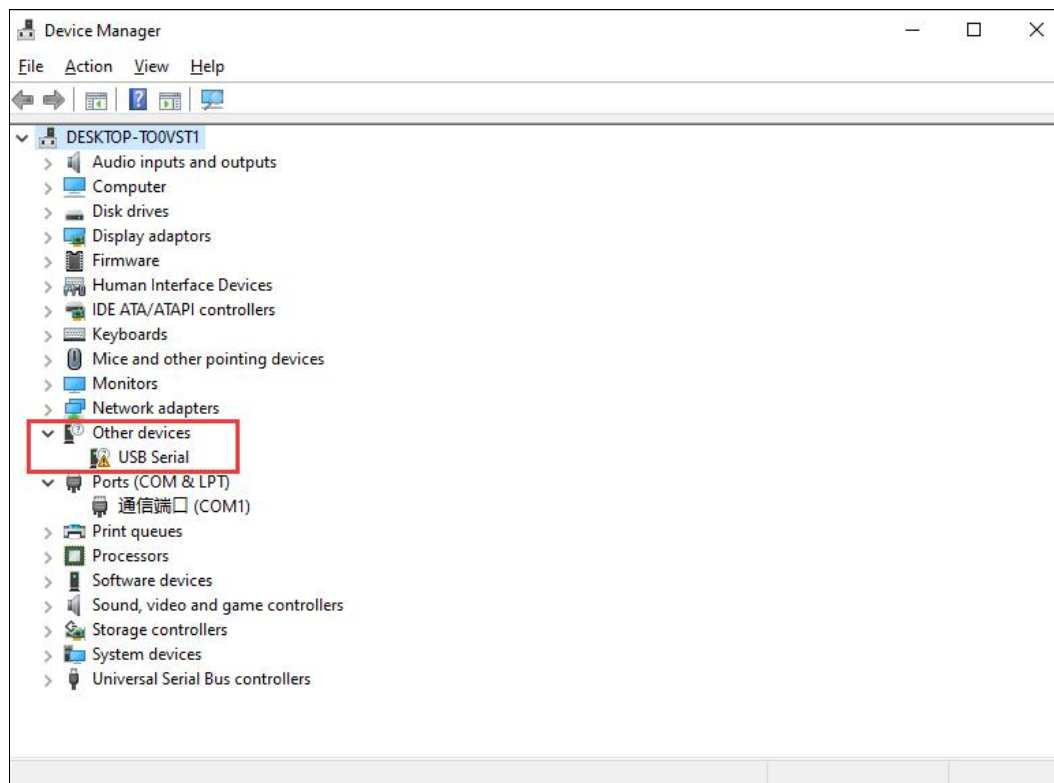
O chip USB-para-serial da placa controladora ESP32 é o CH340, portanto, você precisa instalar o driver para esse chip. Após conectar a placa controladora ao computador com um cabo USB, o driver geralmente será instalado automaticamente nos sistemas MacOS e Windows. Se o driver não for instalado automaticamente, será necessário instalá-lo manualmente.

O driver pode ser baixado da internet, ou você pode encontrar a pasta chamada "[CH340 Driver](#)" no pacote de recursos fornecido por nós. Escolha o pacote apropriado de acordo com o tipo do seu sistema operacional e, em seguida, siga os passos abaixo para instalar o driver.

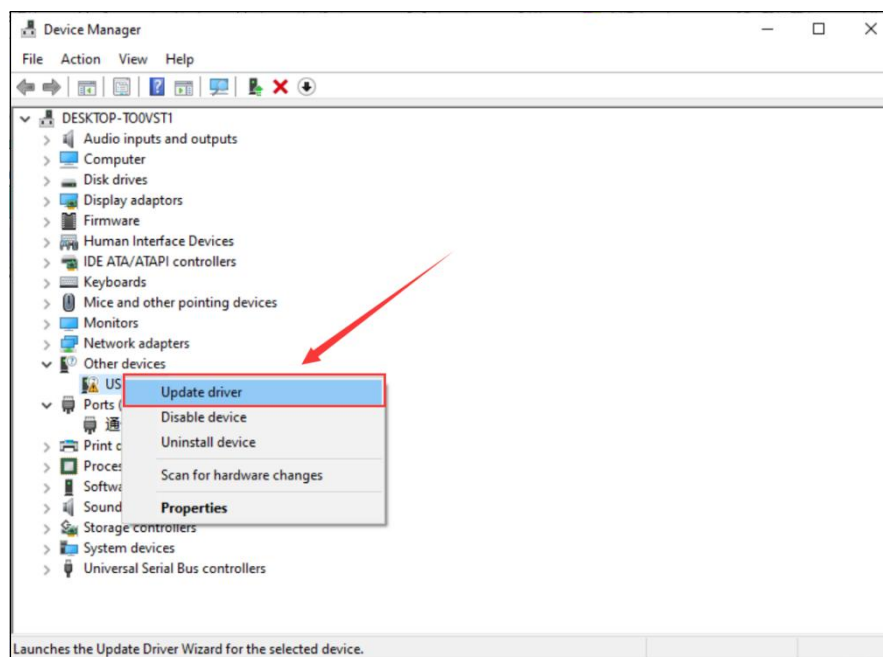
(1)Modo de instalação do driver no sistema Windows

①Insira uma extremidade do cabo USB na placa controladora ESP32 e a outra extremidade em uma porta USB do seu computador.

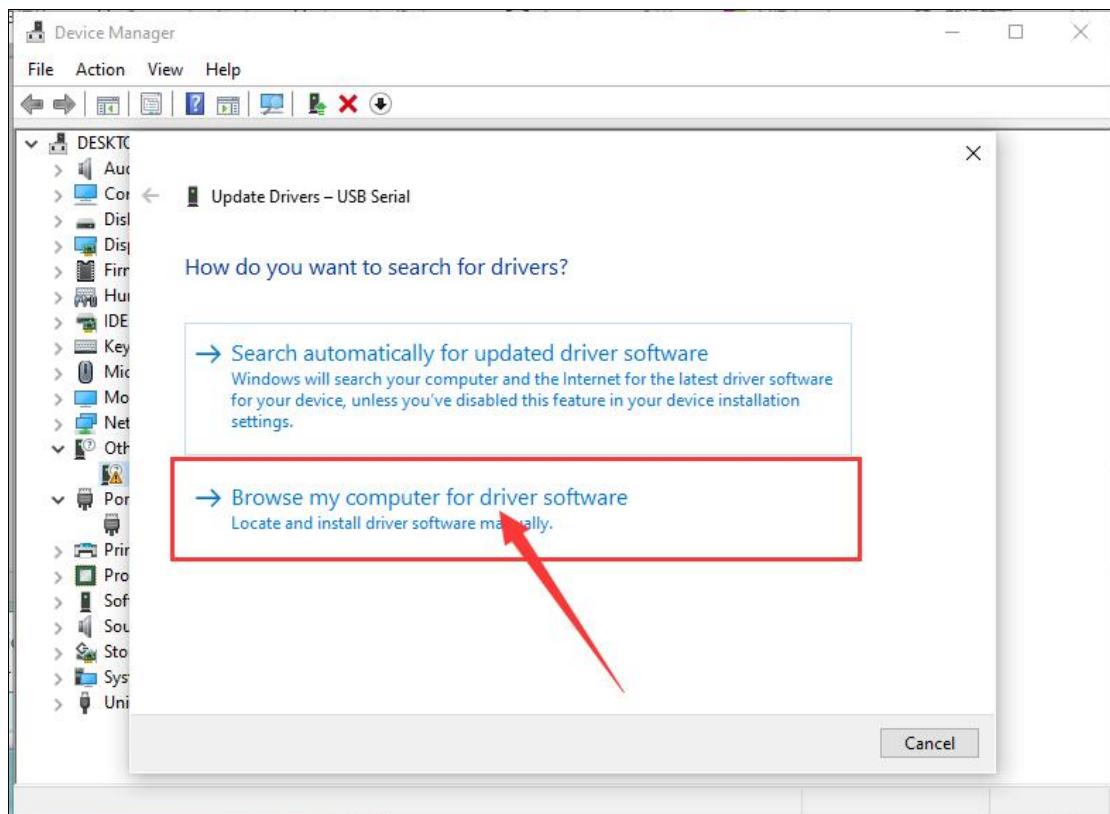
②Ao conectar a placa controladora ESP32 ao computador pela primeira vez, clique com o botão direito em "My Computer" -> "Properties" -> clique em "Device Manager". Em "Other devices", você verá "USB-Serial" ou "Unknown Device".



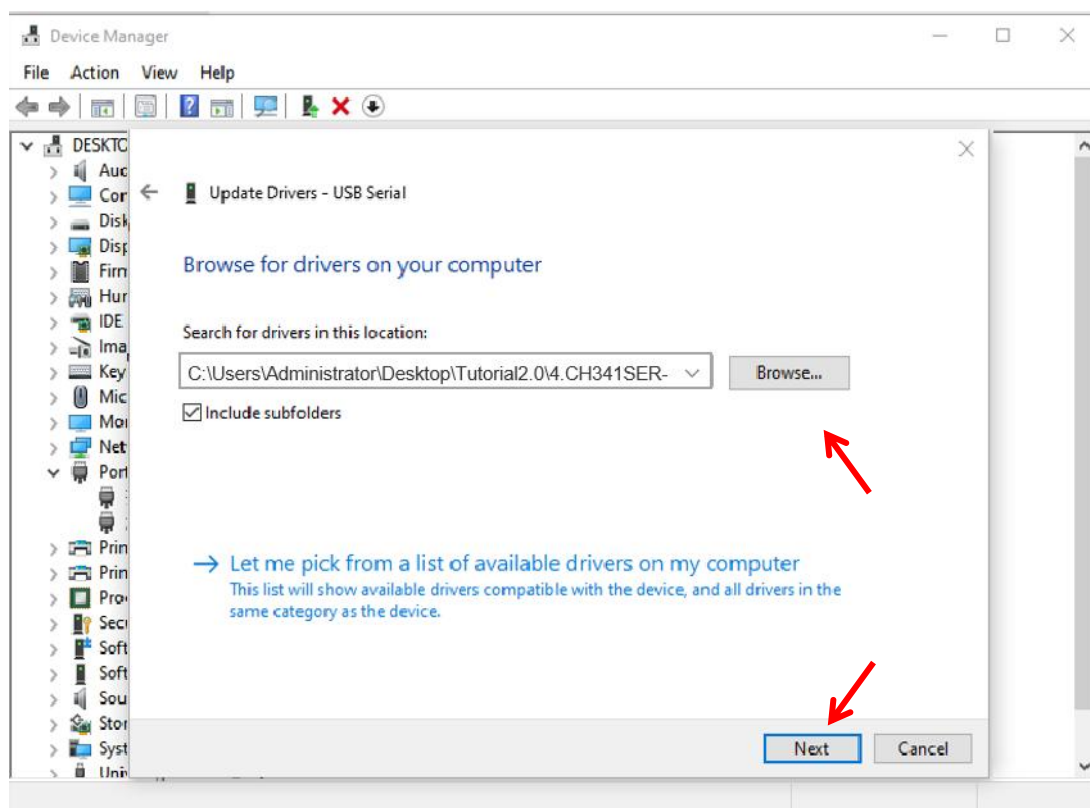
③Clique com o botão direito no dispositivo e selecione a opção do menu superior ("Update driver"), conforme mostrado na imagem abaixo.



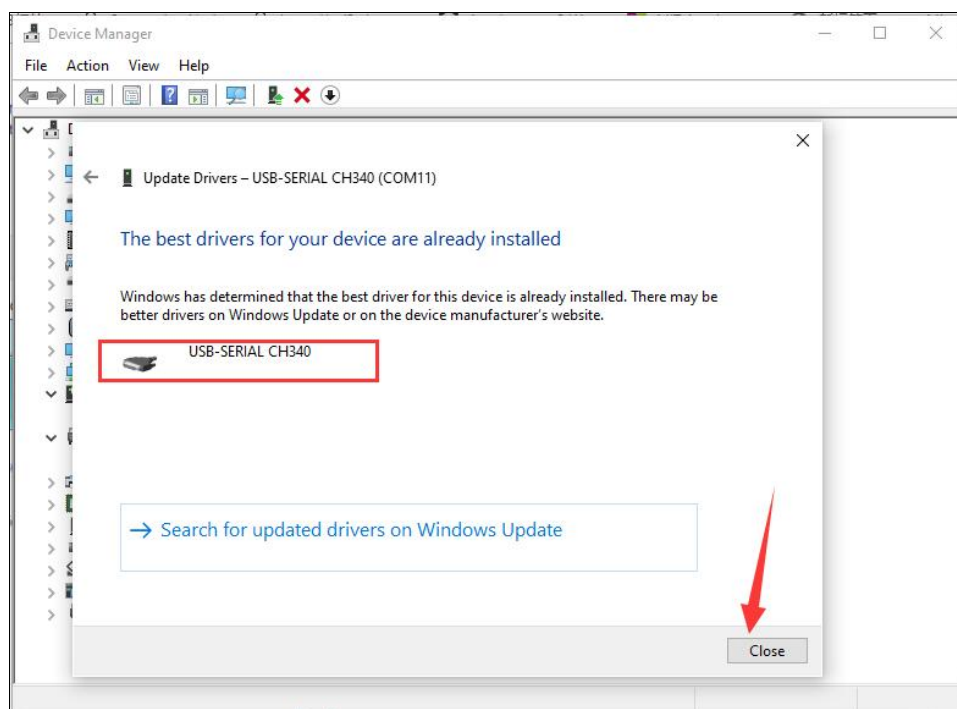
④Em seguida, você será solicitado a escolher entre "Search automatically for updated driver software" ou "Browse my computer for driver software", conforme mostrado na imagem abaixo. Nessa página, selecione "Browse my computer for driver software".



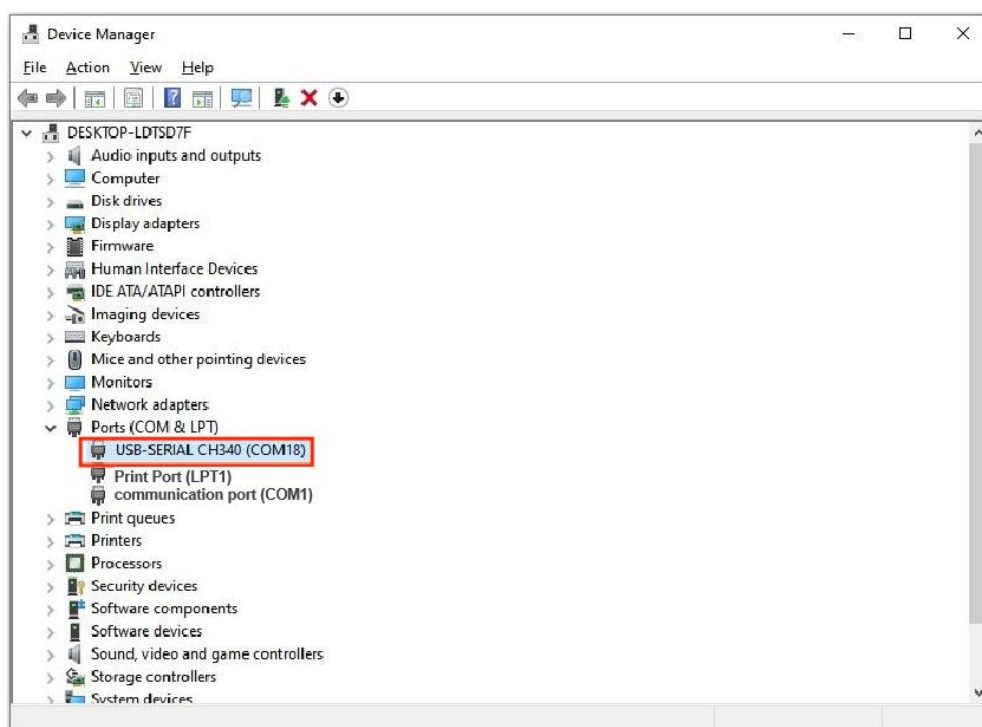
⑤ Em seguida, adicione o caminho para os arquivos do driver.



⑥Após a instalação do software ser concluída, você receberá uma mensagem de confirmação. Quando a instalação for finalizada, clique em "Close".

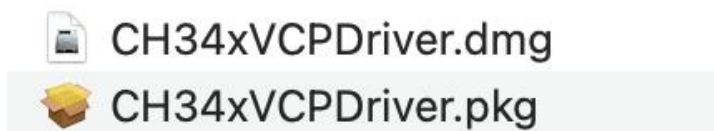


⑦Para confirmar se a instalação foi bem-sucedida, conecte uma extremidade do cabo USB à placa controladora ESP32 e a outra extremidade em uma porta USB do seu computador. Clique com o botão direito em "My Computer" -> "Properties" -> clique em "Device Manager". Depois que a placa controladora estiver conectada, a instalação bem-sucedida será exibida conforme indicado na imagem abaixo.

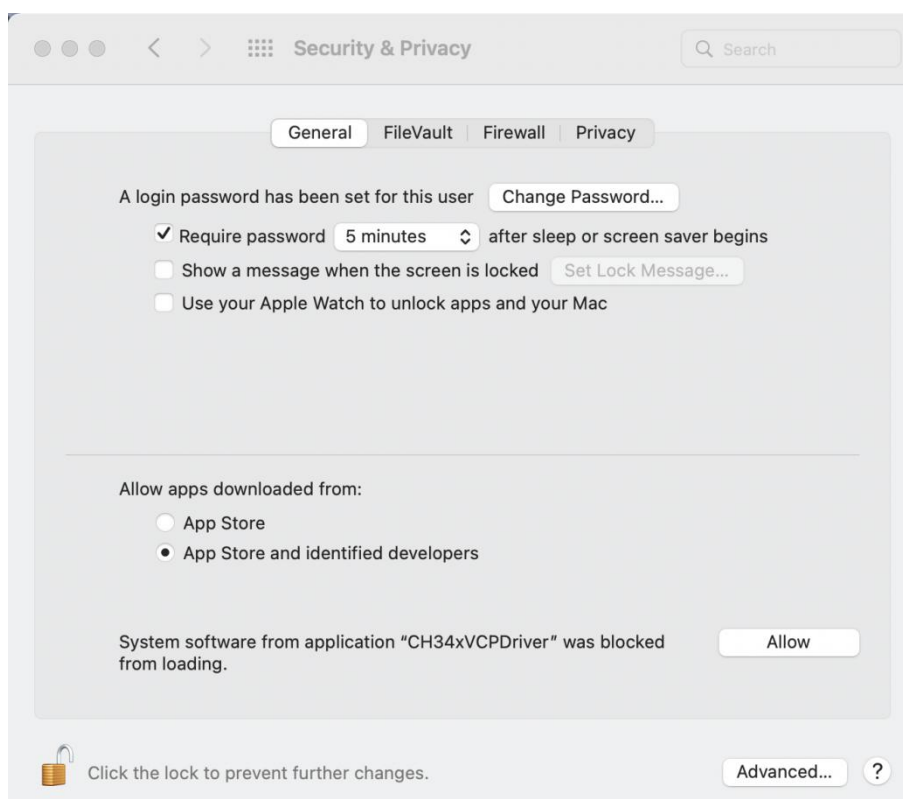


(2) Modo de instalação do driver no sistema macOS

Abra a pasta "CH340 Driver file-mac" e você verá o arquivo de instalação do driver que fornecemos. Em geral, o driver no formato pkg é instalado por padrão, mas se o Rosetta não for suportado no macOS 11.0 ou posterior, instale o driver no formato dmg.

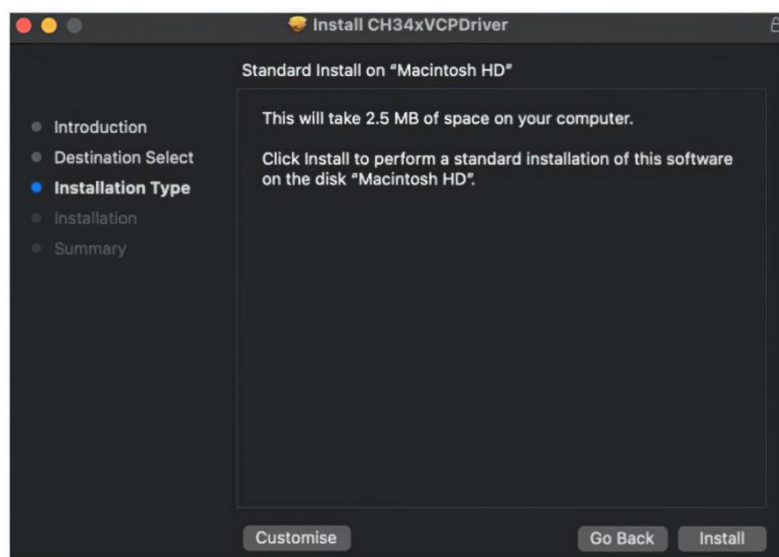
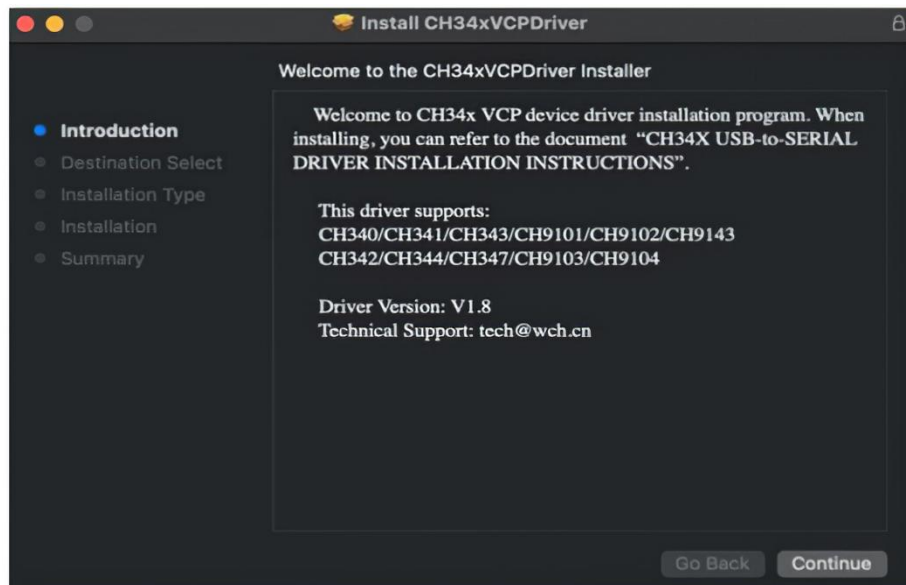


Antes de instalar, vá em "System Preferences" -> "Security & Privacy" -> "General" e selecione "Mac App Store and identified developers" na seção "Allow applications to be downloaded from the following addresses" para que o driver funcione corretamente.

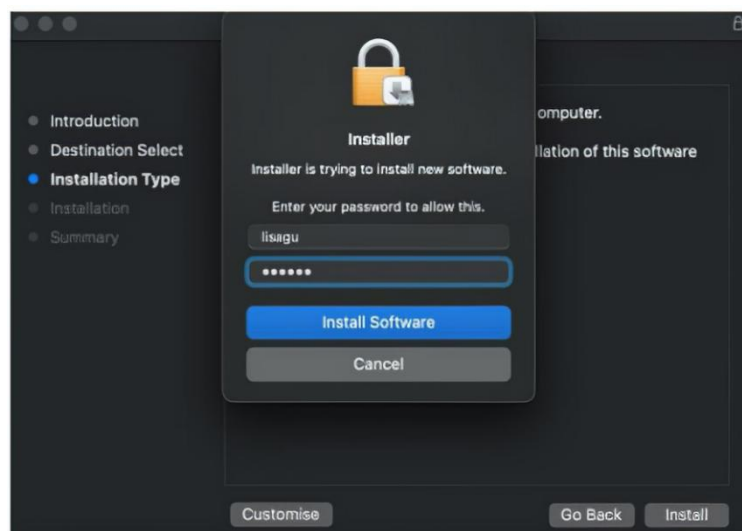


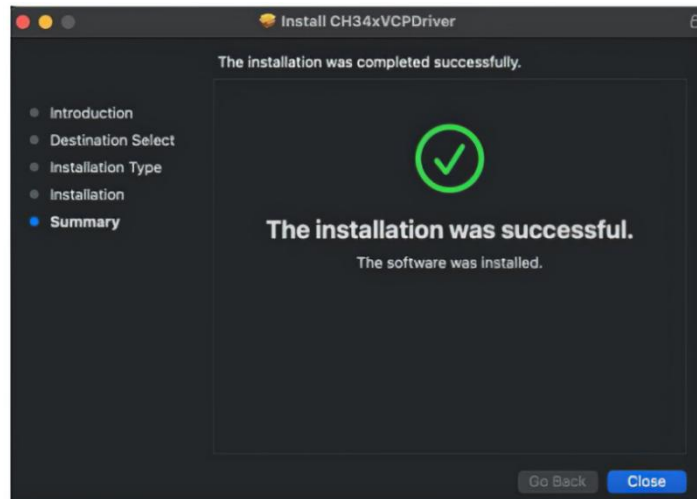
1).Instale o driver no formato pkg

①Clique no arquivo do driver -> Continuar -> Instalar.

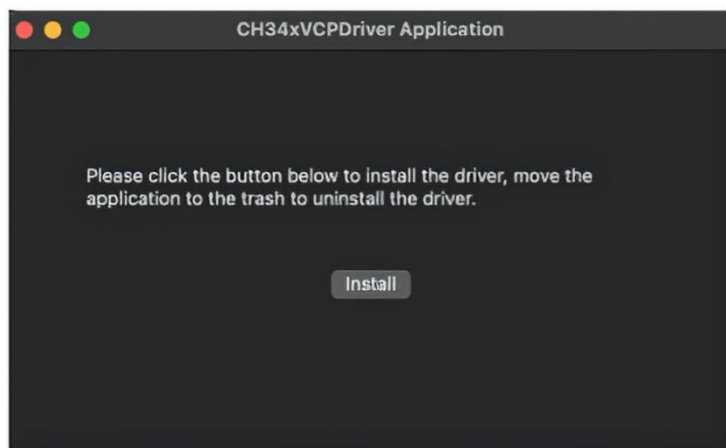


②A instalação foi bem-sucedida.





③ Instalar o driver no formato pkg no macOS 11.0 ou posterior: abra "LaunchPad" -> "CH34xVCPDriver" -> Instalar.

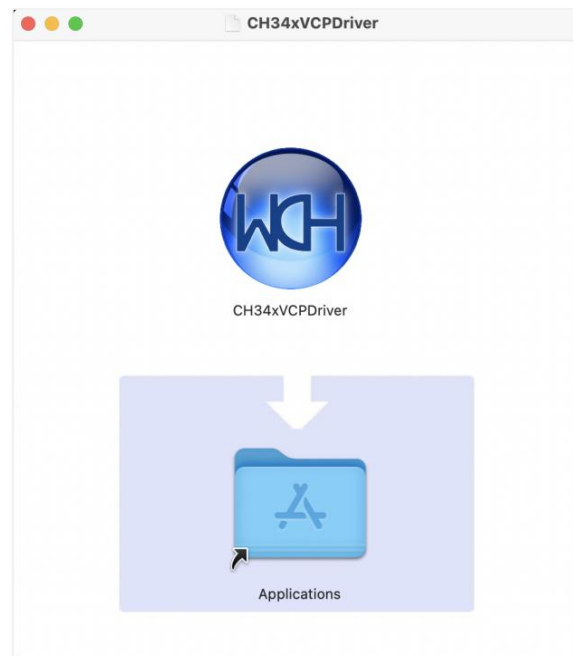


④ Ao usar o macOS 10.9 até o macOS 10.15, clique em Reiniciar para reiniciar o seu computador e execute as seguintes etapas após o reinício.



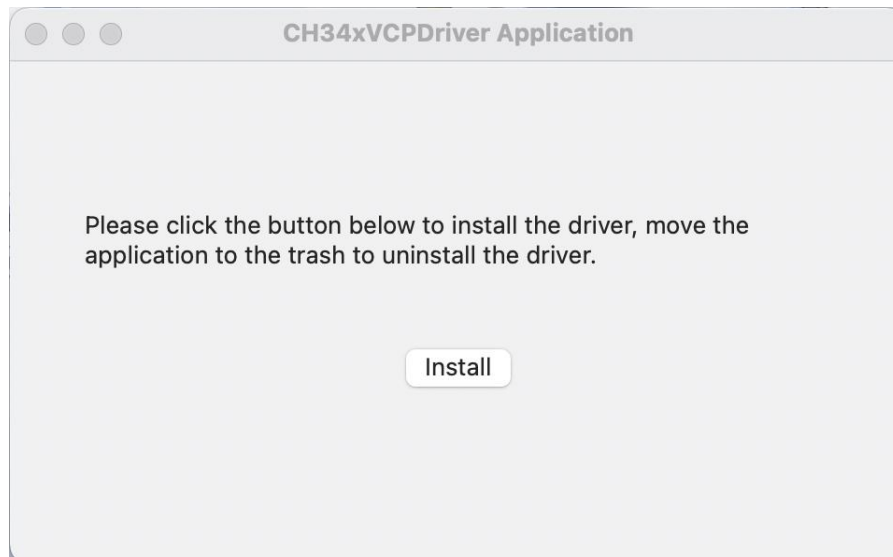
2) Instalar o driver no formato dmg (Atenção: Se o arquivo pkg da Etapa A for instalado com sucesso, pule a Etapa B)

① Instale o driver dmg, clique no arquivo dmg e arraste "CH34xVCPDriver" para a pasta de aplicativos do sistema operacional.

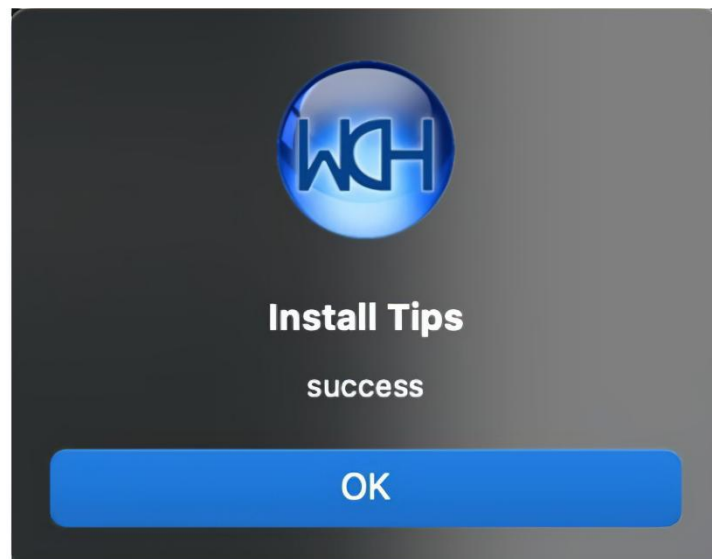


② Em seguida, abra "LaunchPad" -> "CH34xVCPDriver" -> Instalar.



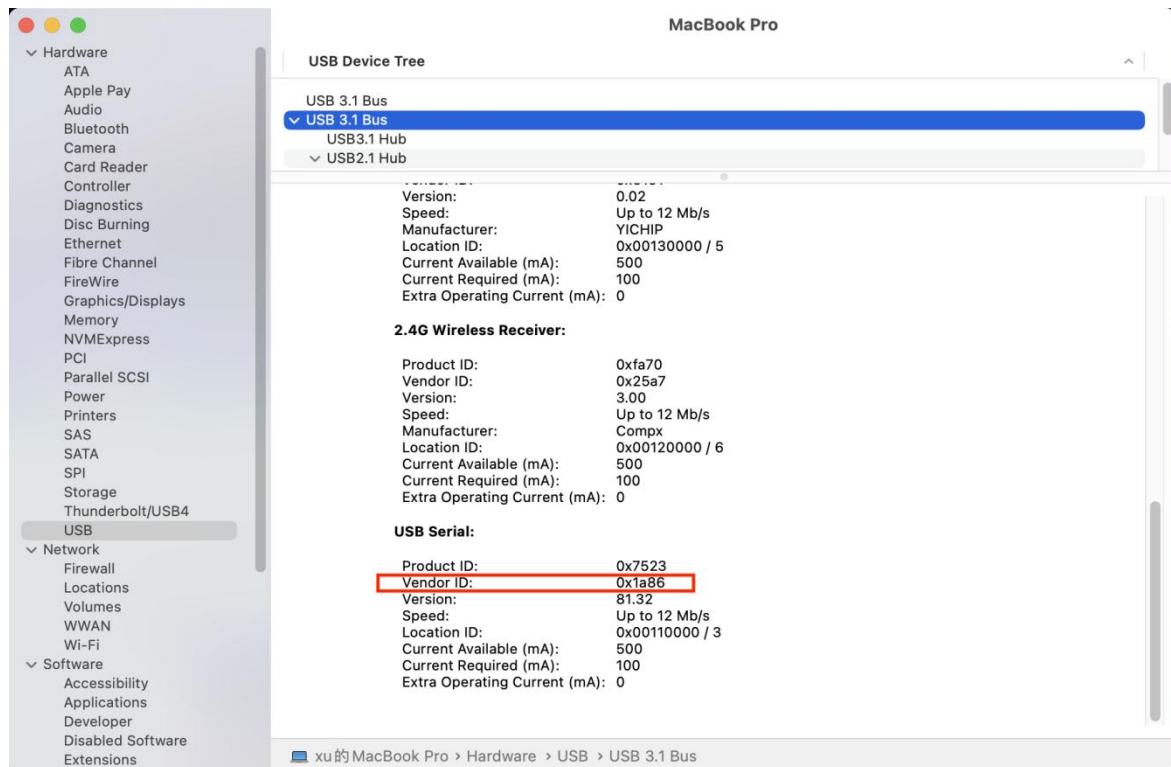


③A instalação foi bem-sucedida.



3)Verifique se o driver da porta serial CH340 está instalado

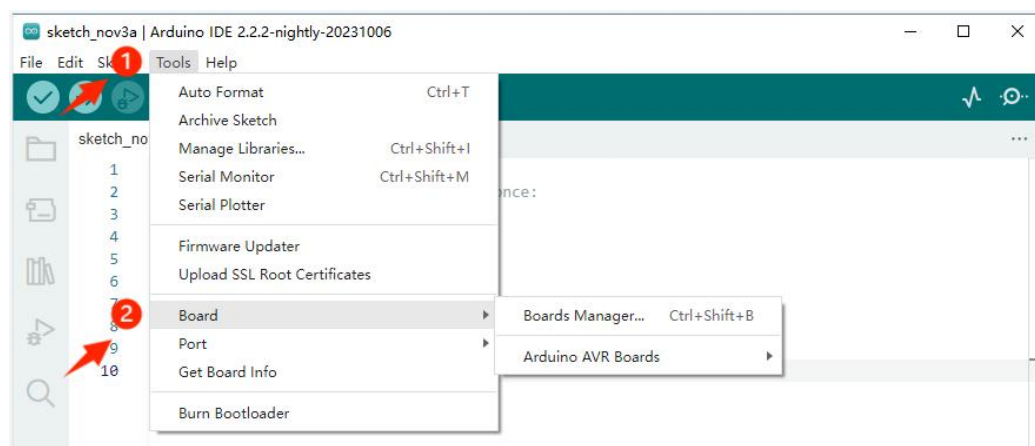
Quando a placa controladora estiver inserida na porta USB, abra Relatório do Sistema -> Hardware -> USB. À direita estará o dispositivo USB. Se o dispositivo USB estiver funcionando corretamente, você encontrará um dispositivo com o "Vendor ID" [0x1a86].



3.Instalar a biblioteca ESP32

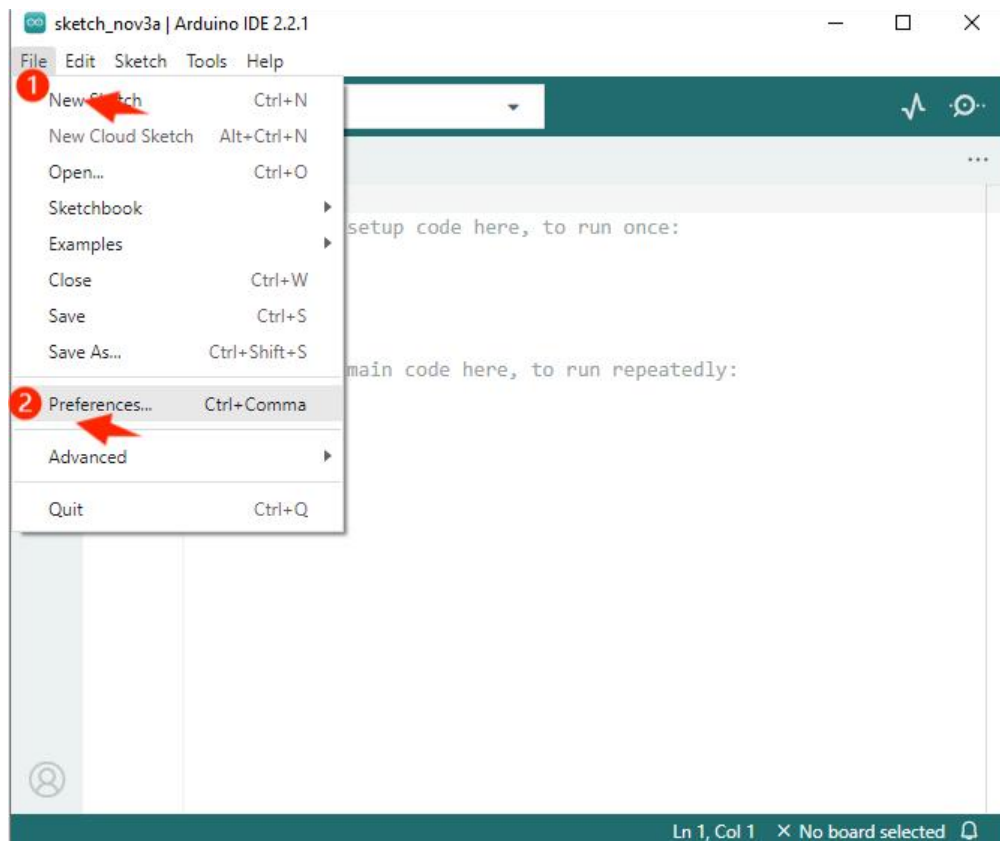
Como a placa controladora usada no braço robótico é a ESP32, também é necessário adicionar a biblioteca da ESP32 para programar a placa ESP32 na IDE Arduino.

Ao abrir a IDE Arduino, selecione Tools>Board, você verá que a IDE Arduino só tem as Placas Arduino AVR e não a ESP32.

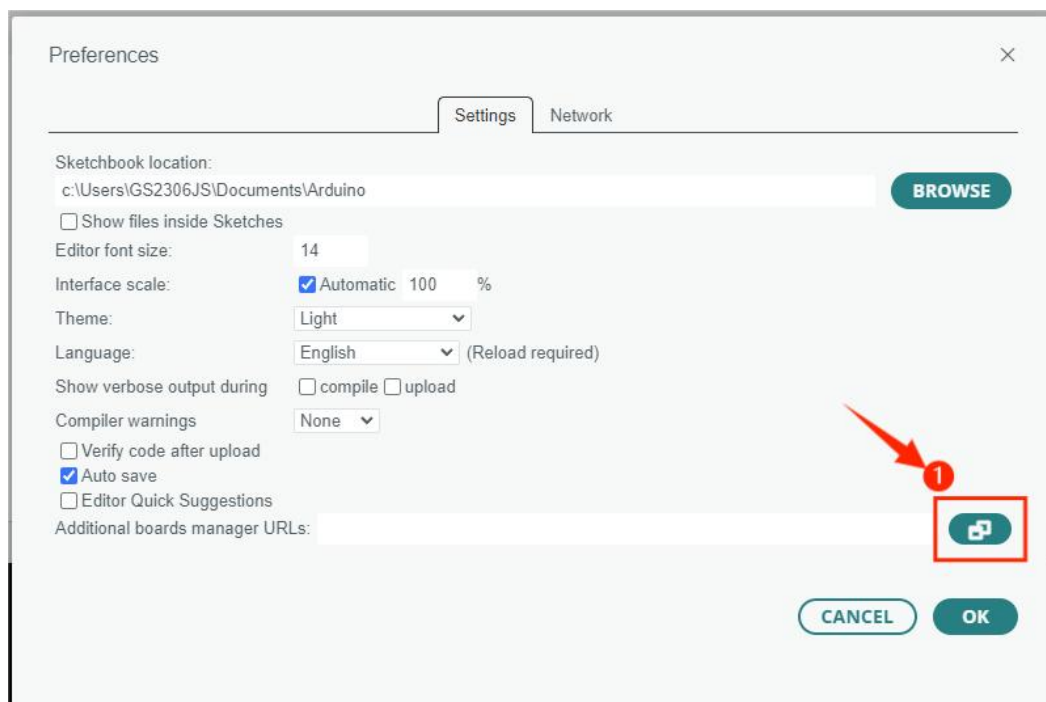


A seguir, precisamos instalar a biblioteca ESP32. Siga estes passos:

① Abra File > Preferences

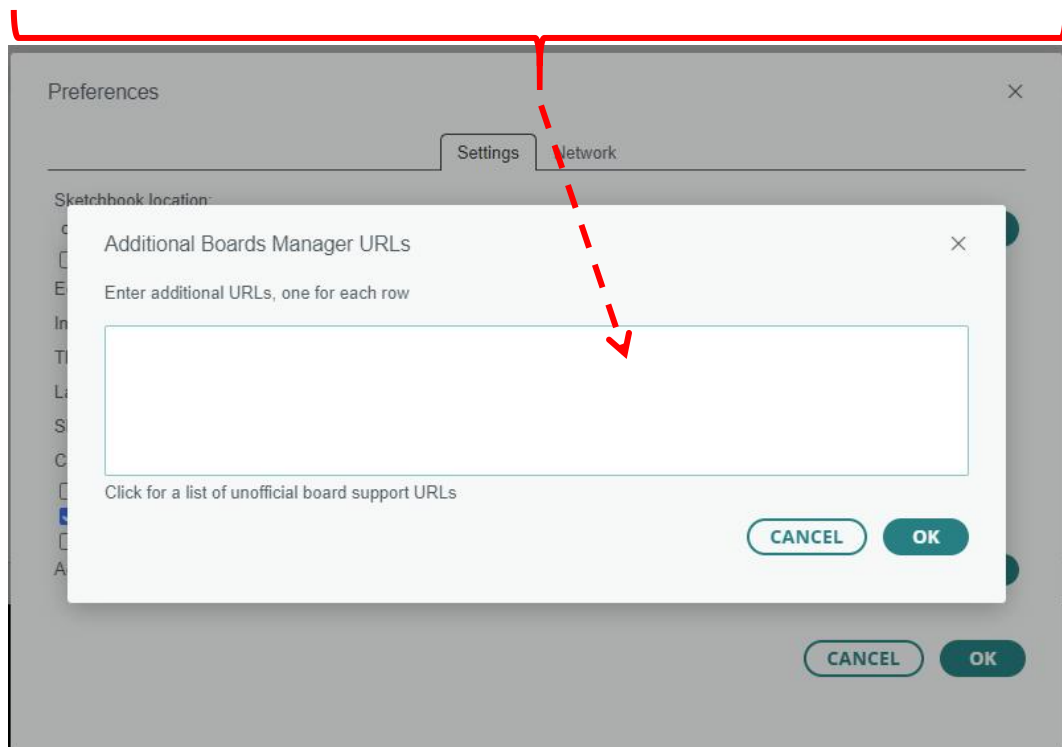


② Adicione o URL de gerenciamento da placa controladora.

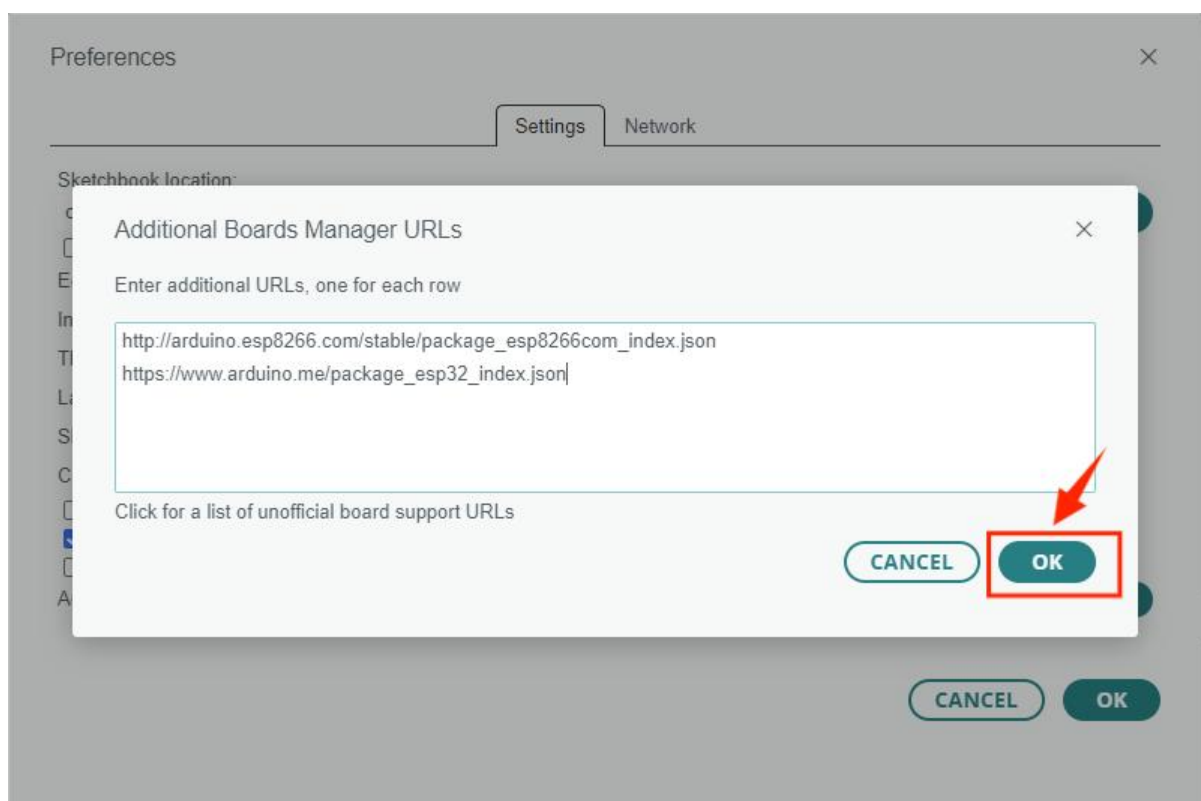


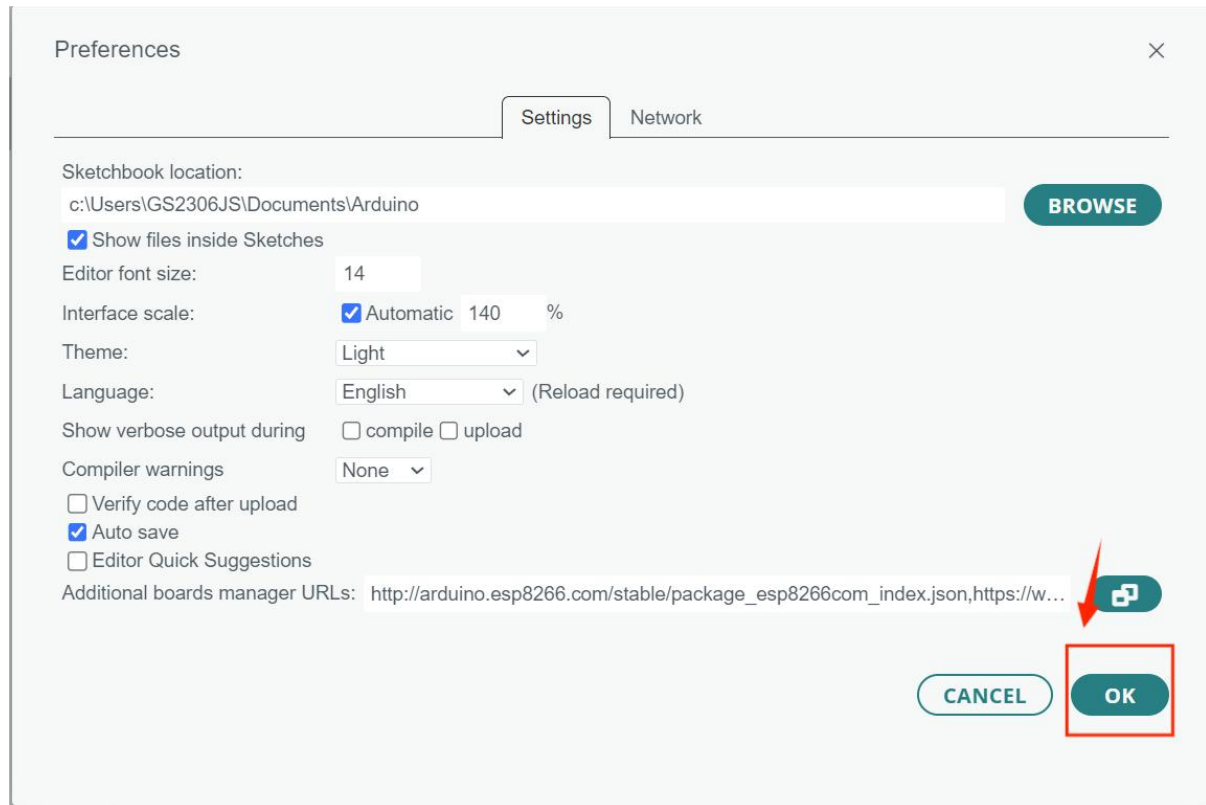
③ Copie o URL na caixa de texto inferior e adicione-o em "URLs adicionais do Gerenciador de Placas"

http://arduino.esp8266.com/stable/package_esp8266com_index.json
https://www.arduino.me/package_esp32_index.json

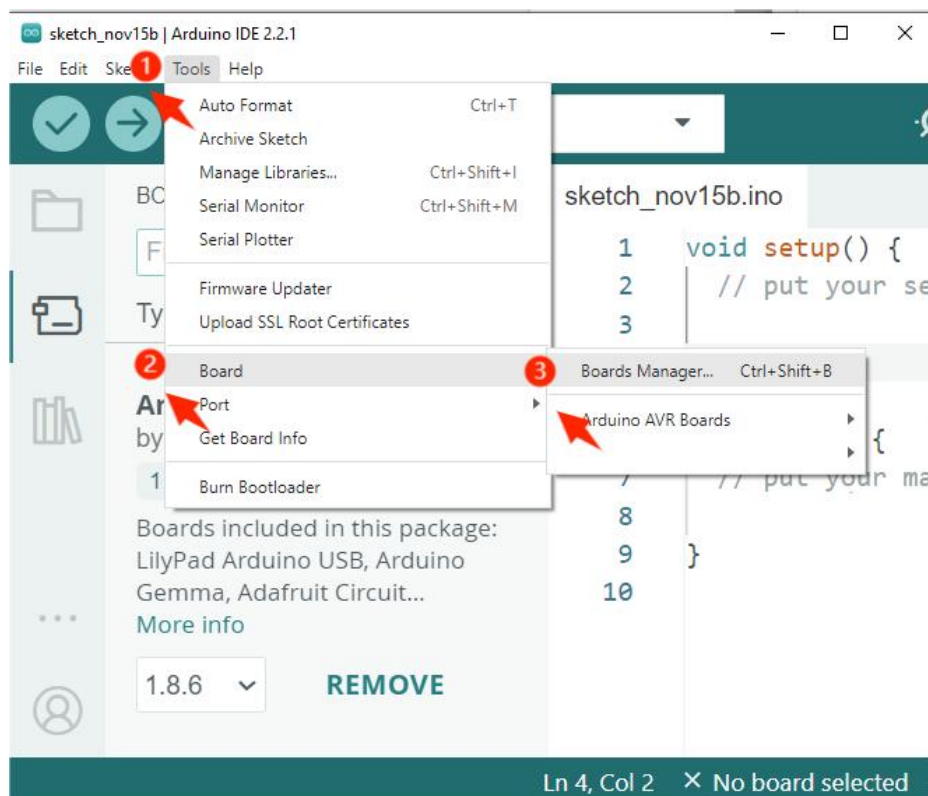


④ Após adicionar o URL, clique em "OK".



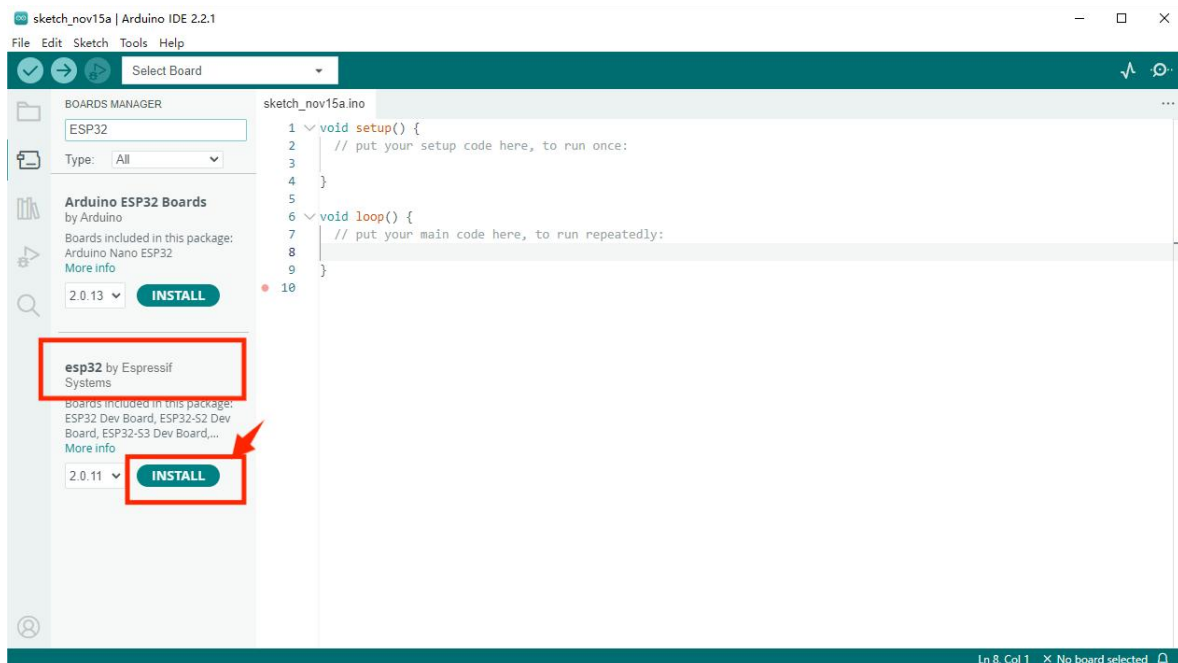


⑤Clique em Tools > Board > Boards Manager.

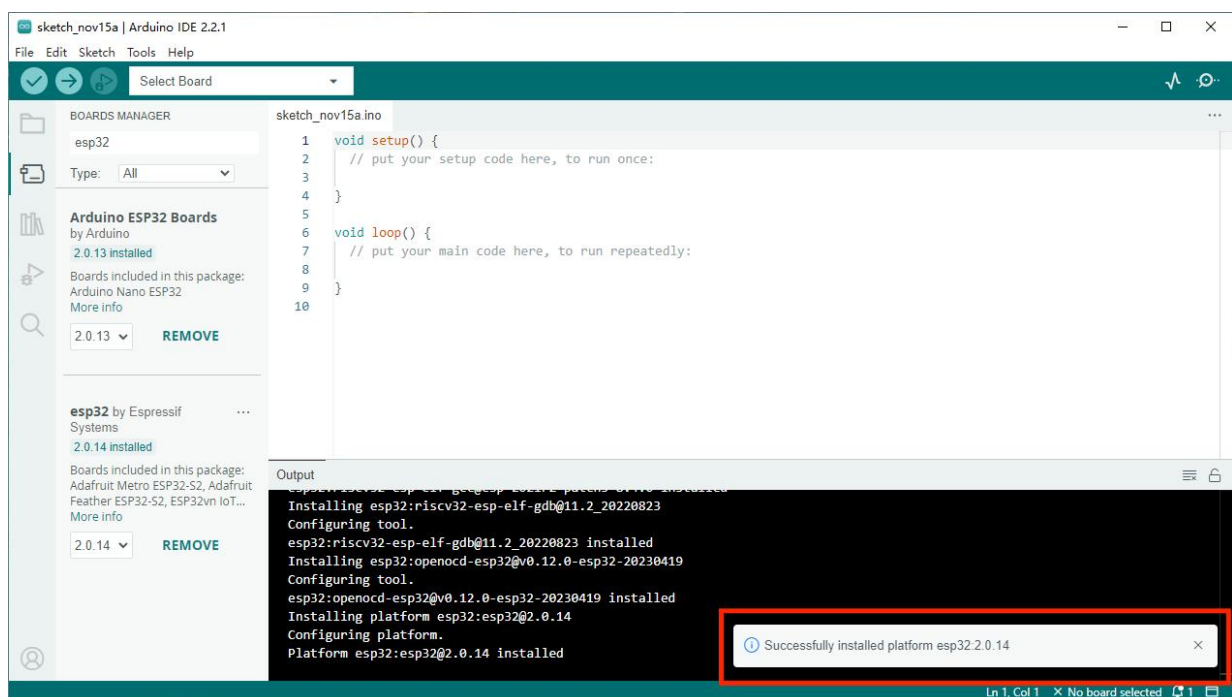


⑥Pesquise por "ESP32" na barra de pesquisa do GERENCIADOR DE PLACAS e instale-o.

Atenção: Instale a versão 2.0.12 do ESP32, pois as versões mais recentes são incompatíveis com as bibliotecas do tutorial, o que pode causar erros no programa! Se você já instalou a versão 3.0, desinstale-a e reinstale a versão 2.0 do ESP32.

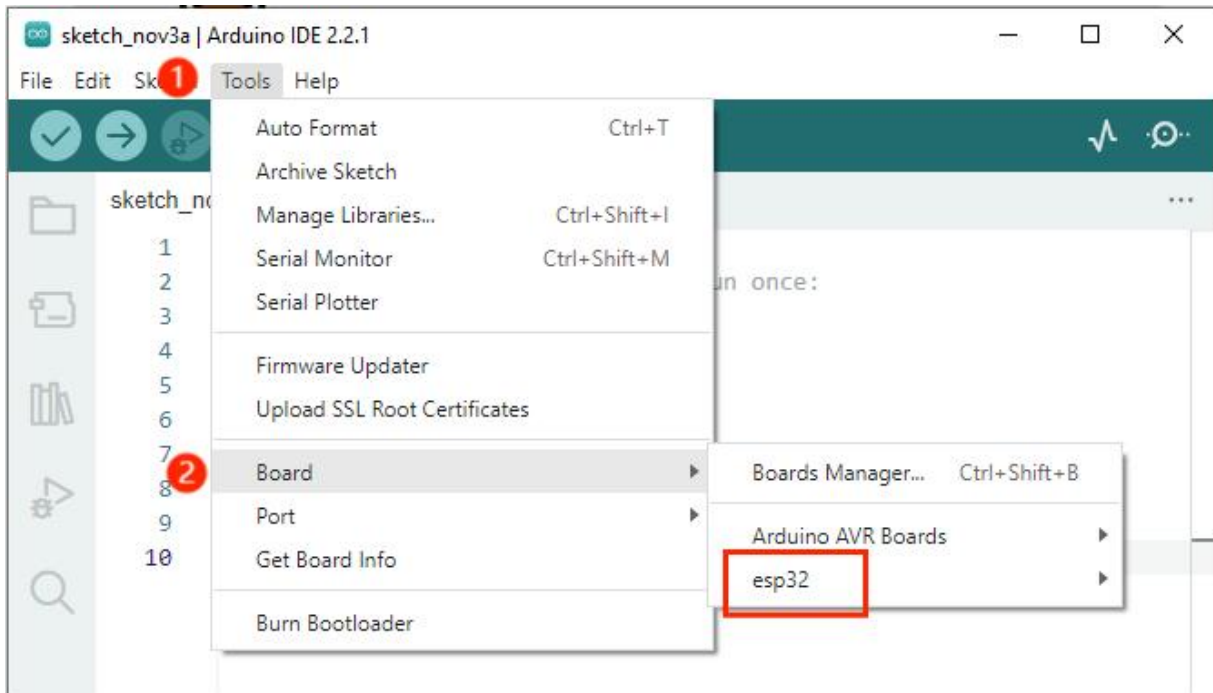


⑦Quando a seguinte interface aparecer, aguarde até que a instalação seja concluída, e então feche a IDE Arduino.



Atenção: Como o pacote de instalação está hospedado no GitHub, ele pode ser afetado pela velocidade da rede. Se a instalação falhar, tente várias vezes.

⑧Reabra a IDE Arduino, selecione Tools > Board, e você verá a placa ESP32 aparecer.



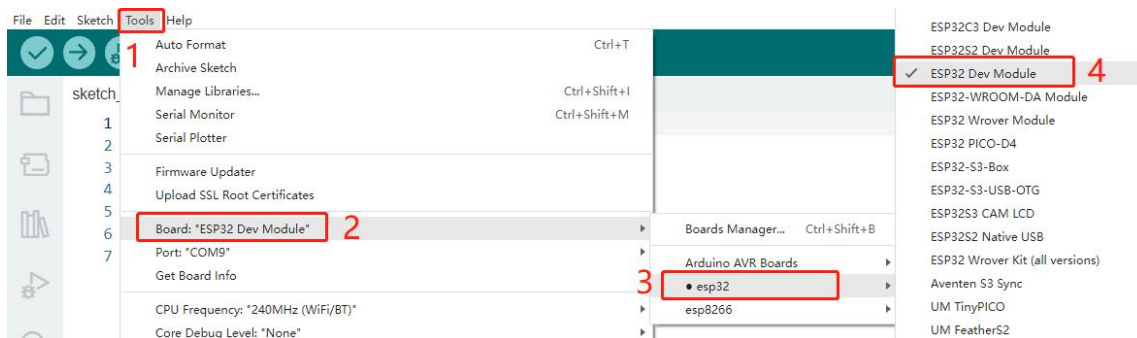
4.Add library files

Abra o arquivo "[Add libraries](#)" em "português\Arduino (Aprendiz Experiente)\4.Adicionar arquivos de bibliotecas" e siga as instruções para adicioná-las.

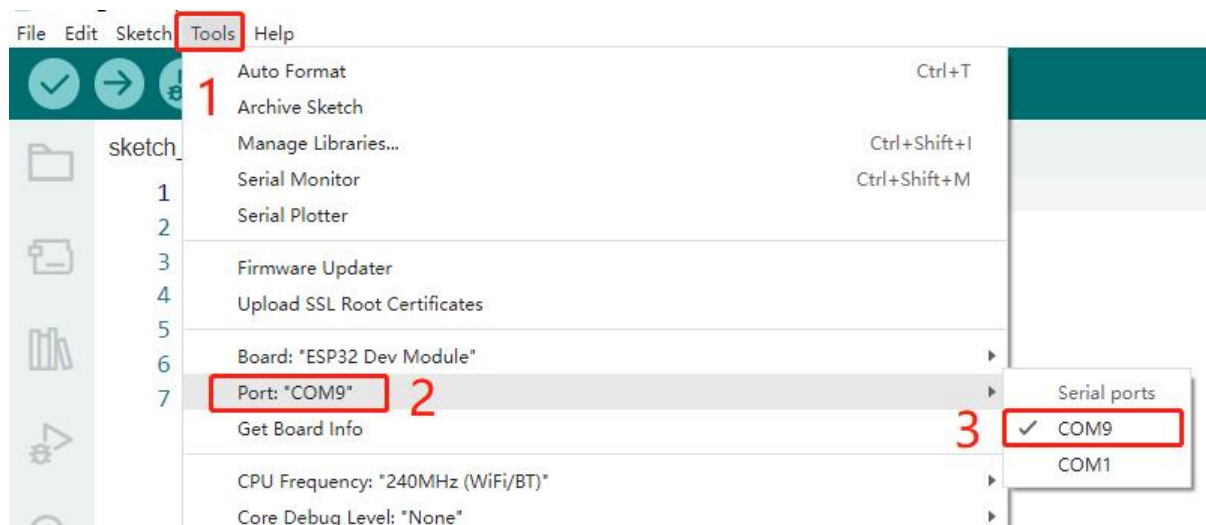
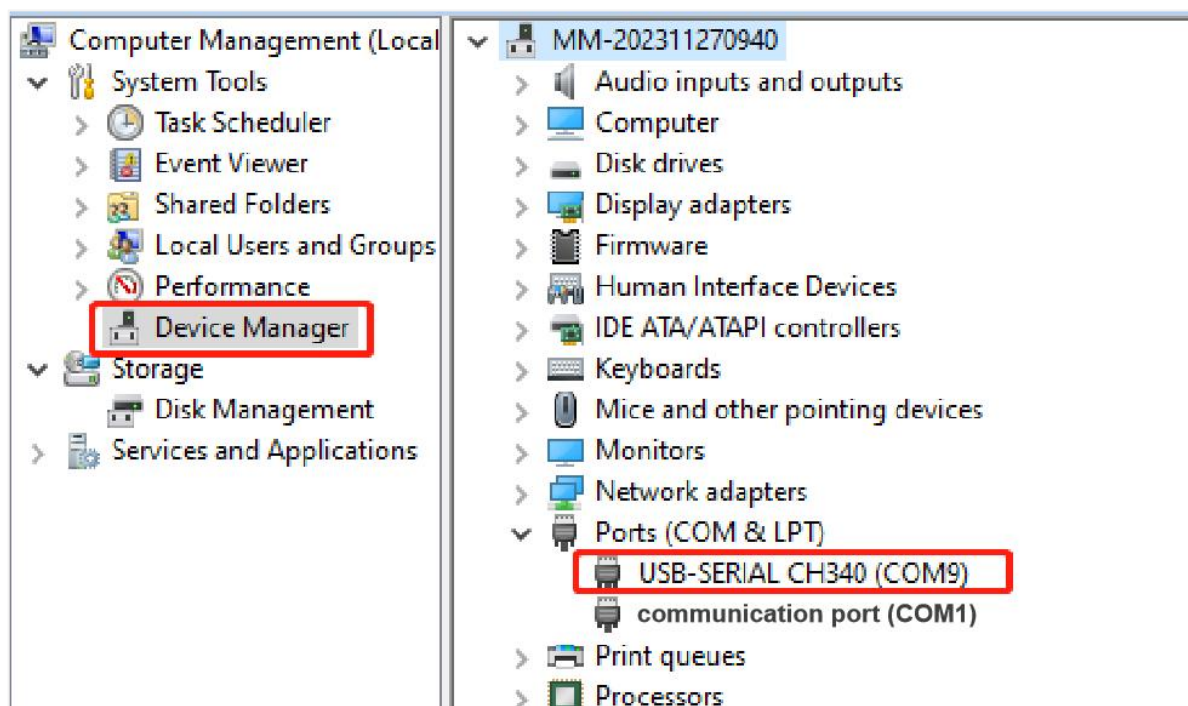
5.Testar o ambiente de desenvolvimento

Após instalar a IDE Arduino e a biblioteca de extensão da placa ESP32, você pode testar se o ambiente de desenvolvimento foi configurado com sucesso com um programa simples. Siga os seguintes passos:

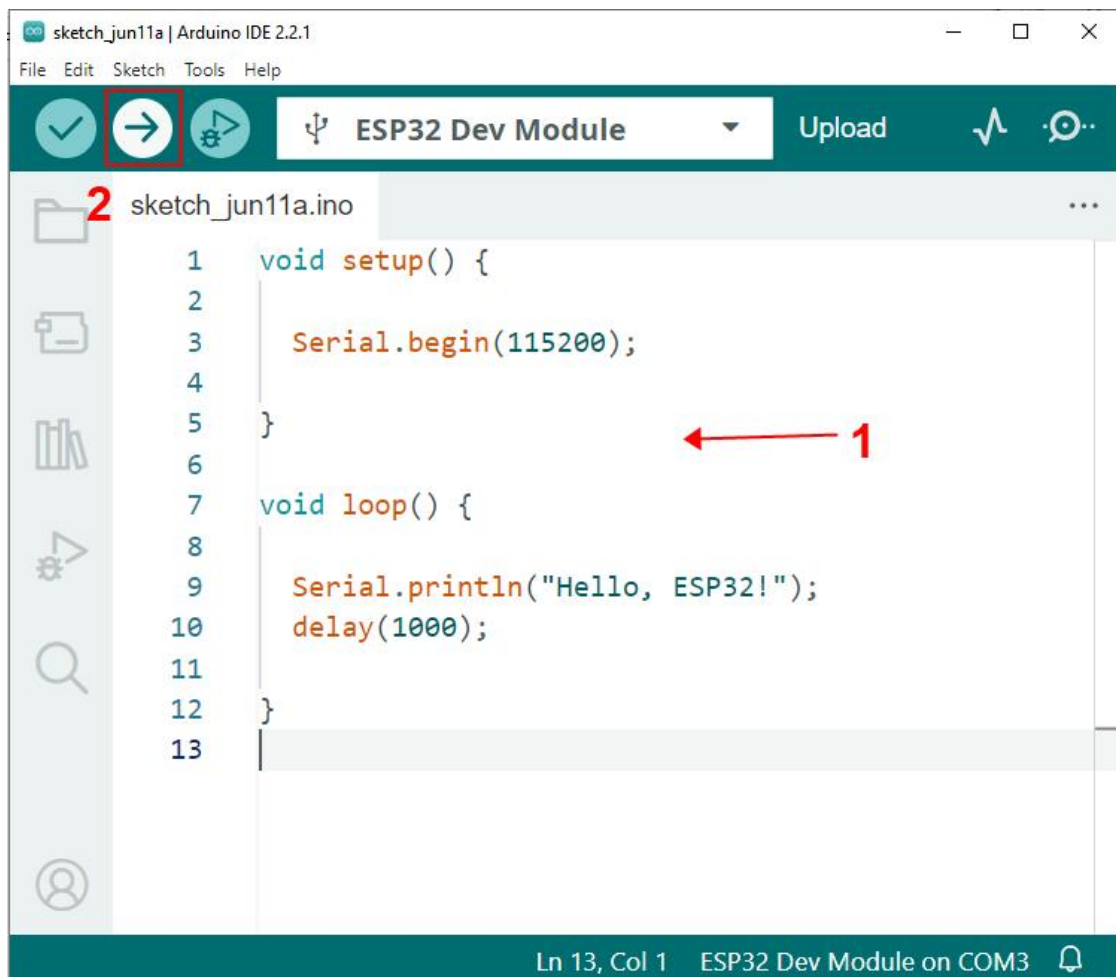
①Conecte a placa controladora ao seu computador > Abra a IDE Arduino > Clique em Ferramentas > Selecione ESP32 > Escolha (ESP32 Dev Module).

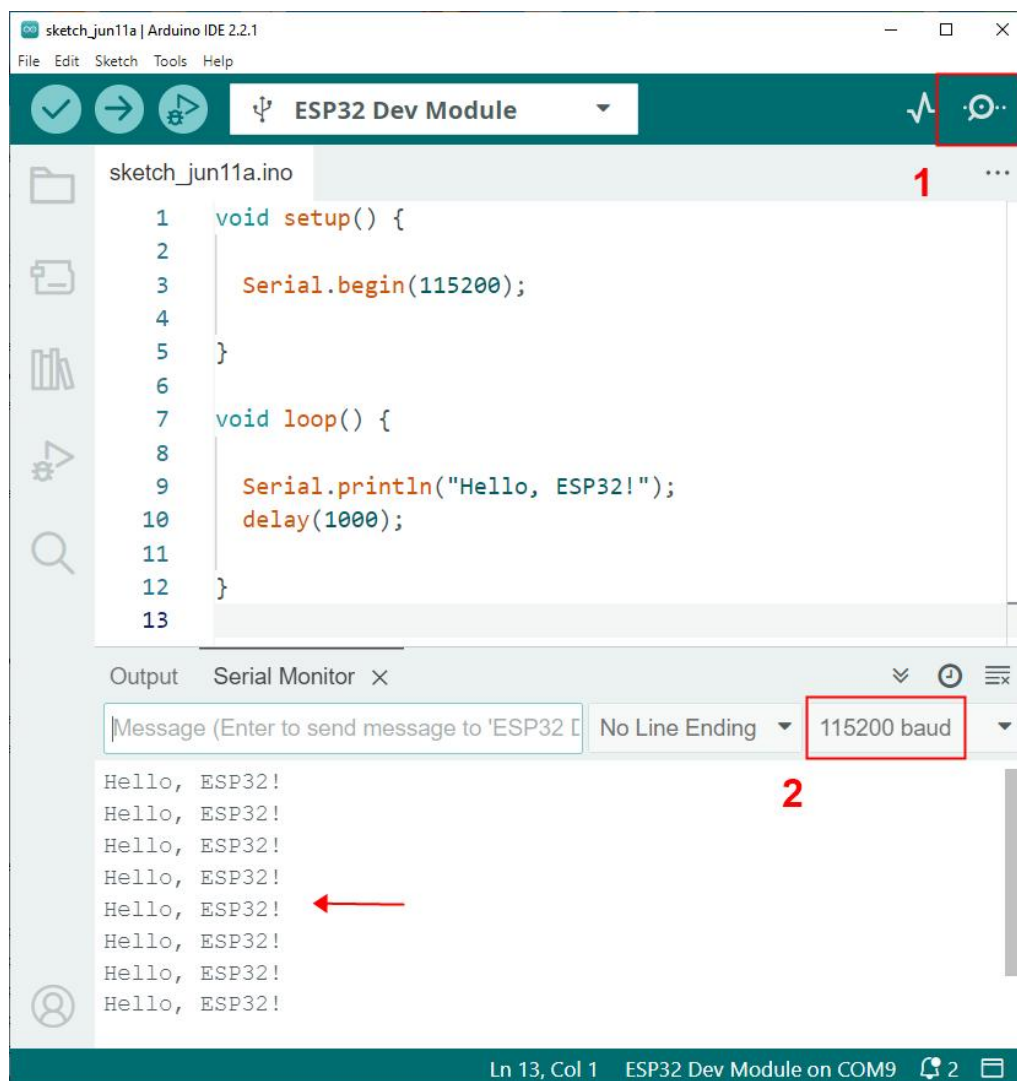


②Selecione a porta serial (você pode verificar o número da porta serial no Gerenciador de Dispositivos do computador e, em seguida, verificar se a porta correspondente aparece abaixo. Como cada placa tem um número COM diferente, escolha o número COM exibido de acordo).



③ Abra o arquivo 'Hello_esp32.ino' em 'português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 1\Hello_esp32', conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora e a porta corretas, faça o upload do código para a placa controladora ESP32, selecione a taxa de transmissão de 115200. Você verá o monitor serial exibindo constantemente 'Hello, ESP32!'.





III.Compreendendo os Servos

1.Introdução aos Servos

A estrutura principal do servo é mostrada na figura a seguir, que é composta principalmente por várias partes: carcaça, conjunto de engrenagens de velocidade variável, motor, potenciômetro ajustável, placa de circuito de controle e volante.

O princípio de funcionamento é controlar a placa de circuito para receber sinais de controle da fonte de sinal e acionar o motor para rotacionar. O conjunto de engrenagens reduz a velocidade do motor várias vezes e amplifica o torque de saída do motor de forma correspondente. O potenciômetro e o estágio final do conjunto de

engrenagens giram juntos para medir o ângulo real de rotação do eixo do servo. A placa de circuito de controle recebe o feedback do potenciômetro para determinar o ângulo real do motor e compará-lo com o ângulo alvo. Se houver erro, ela controla o servo para girar até a posição do ângulo alvo.

O fluxo de trabalho é o seguinte: Sinal de Controle → Placa de Circuito Eletrônico → Rotação do Motor → Redução do Conjunto de Engrenagens → Rotação do Servo → Feedback do Ângulo Real do Motor → A Placa de Circuito de Controle ajusta a posição do motor com base no feedback para o ângulo alvo.

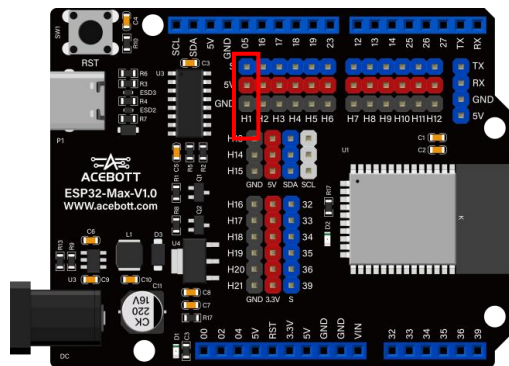
2.A definição dos pinos do servo

① Normalmente, um servo possui três fios de controle: fio de alimentação, fio de terra e fio de sinal.



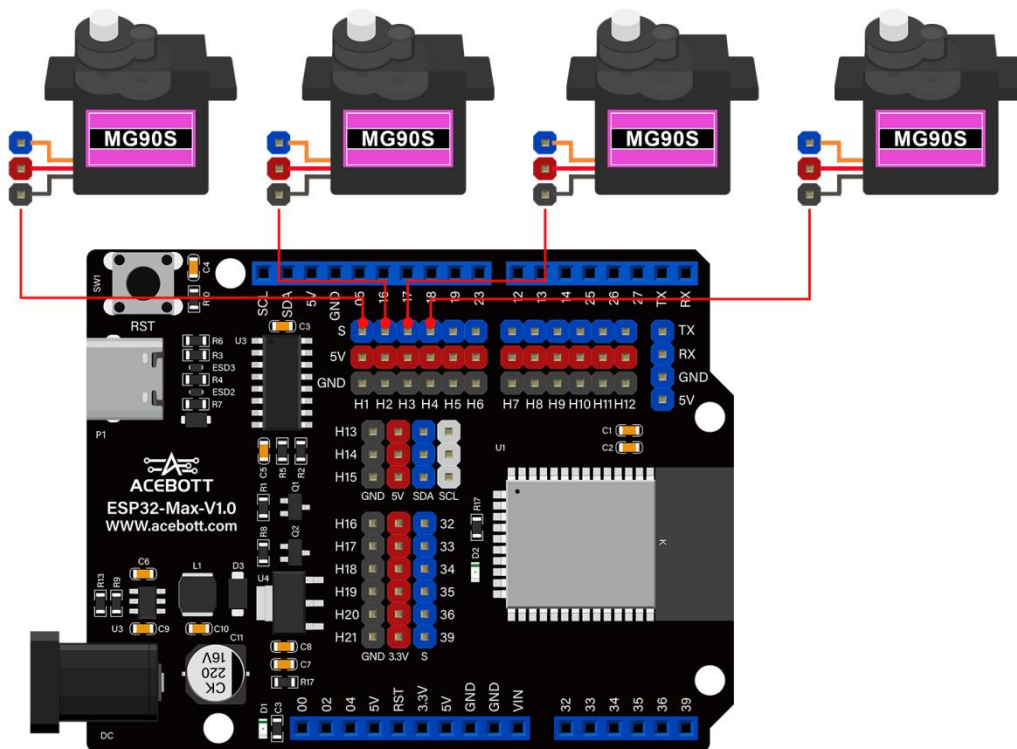
② Definições dos pinos do servo: Fio marrom - GND, Fio vermelho - 5V, Fio laranja - sinal.

③ Conecte o servo à placa controladora ESP32 de acordo com o diagrama a seguir.

Servo	Placa de controle ESP32	Diagrama
Fio marrom	GND	
Fio vermelho	5V	
Fio laranja	GPIO5	

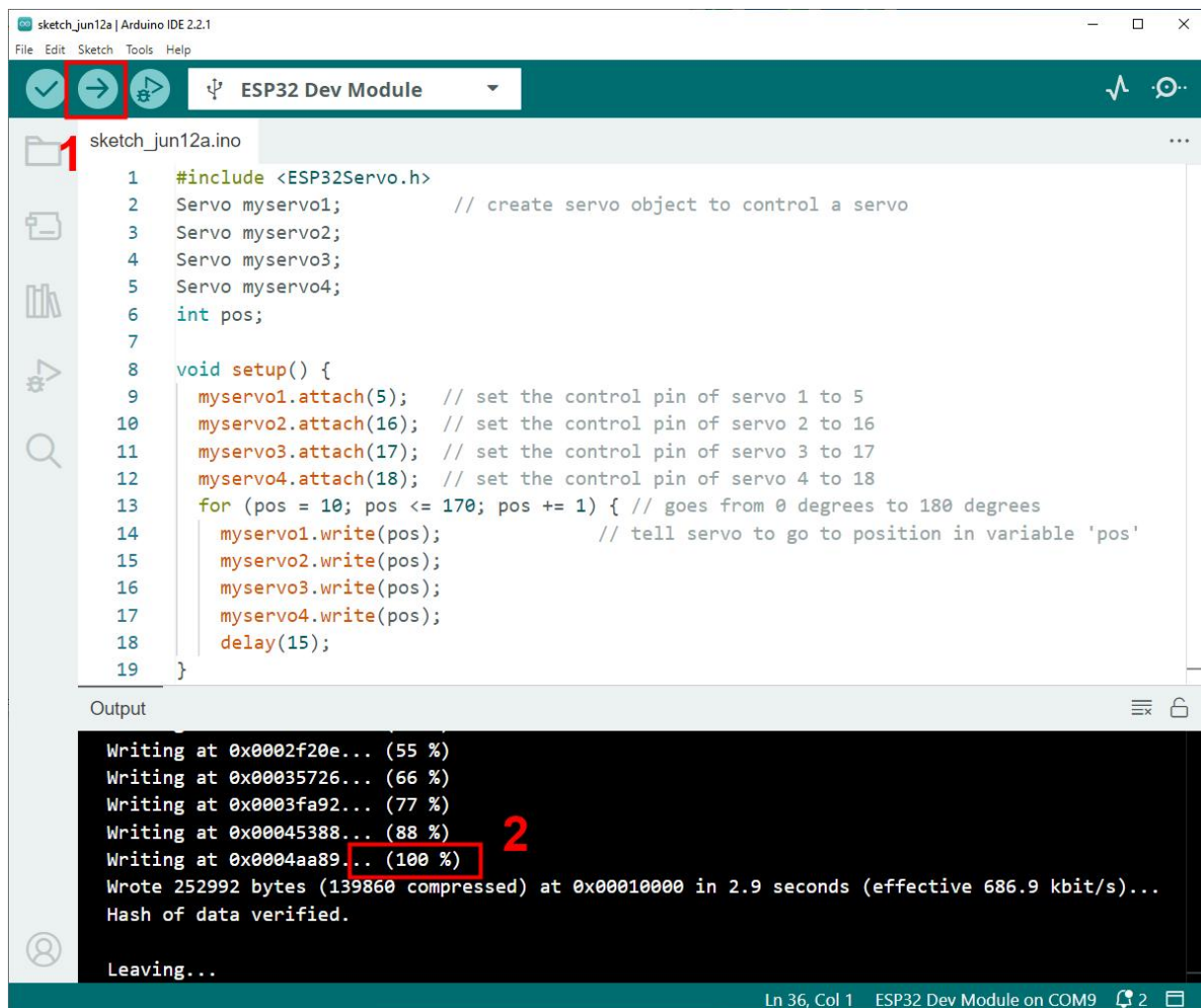
3.Teste do Servo

①Conecte os quatro servos conforme mostrado no diagrama.



②Abra o arquivo '[Servo_test.ino](#)' em português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 1\Servo_test, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora, processador e porta corretos, e faça o upload do código para a placa controladora ESP32.

Atenção: Para manter uma saída de energia estável, você precisa instalar as baterias 18650 no compartimento de baterias, conectá-las à porta de alimentação da placa controladora e ligá-las na posição "on" antes de operar.



```
sketch_jun12a.ino
1  #include <ESP32Servo.h>
2  Servo myservo1;          // create servo object to control a servo
3  Servo myservo2;
4  Servo myservo3;
5  Servo myservo4;
6  int pos;
7
8  void setup() {
9      myservo1.attach(5);   // set the control pin of servo 1 to 5
10     myservo2.attach(16);  // set the control pin of servo 2 to 16
11     myservo3.attach(17);  // set the control pin of servo 3 to 17
12     myservo4.attach(18);  // set the control pin of servo 4 to 18
13     for (pos = 10; pos <= 170; pos += 1) { // goes from 0 degrees to 180 degrees
14         myservo1.write(pos);           // tell servo to go to position in variable 'pos'
15         myservo2.write(pos);
16         myservo3.write(pos);
17         myservo4.write(pos);
18         delay(15);
19     }
20 }
```

Output

```
Writing at 0x0002f20e... (55 %)
Writing at 0x00035726... (66 %)
Writing at 0x0003fa92... (77 %)
Writing at 0x00045388... (88 %)
Writing at 0x0004aa89... (100 %)
Wrote 252992 bytes (139860 compressed) at 0x00010000 in 2.9 seconds (effective 686.9 kbit/s)...
Hash of data verified.
Leaving...
```

Ln 36, Col 1 ESP32 Dev Module on COM9

③Se o servo estiver funcionando corretamente, ele irá girar de 0 graus para 180 graus, depois de 180 graus de volta para 0 graus, e finalmente se estabilizará na posição de 90 graus.

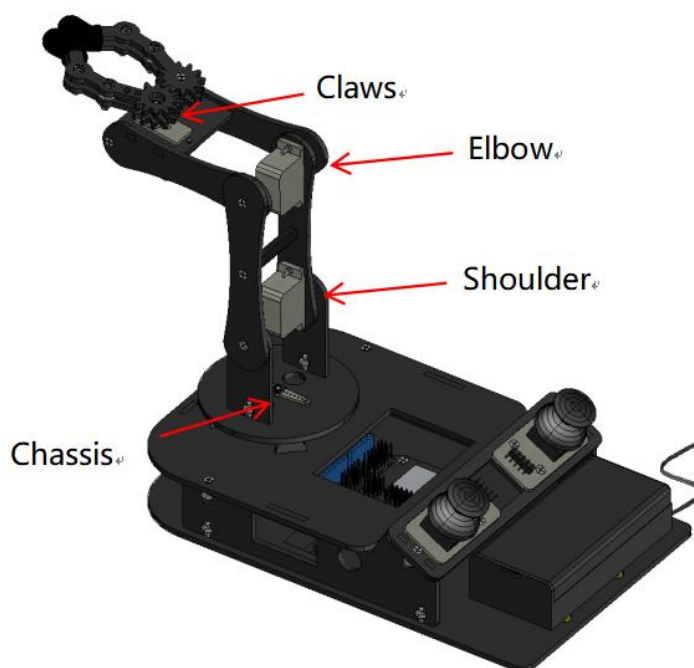
4.Programa de Zeração do Motor do Servo

Antes de instalar o braço robótico, é importante pré-gravar o programa de zeração do motor do servo para facilitar a montagem suave de sua estrutura.




























Abra o arquivo '[servo_90.ino](#)' em português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 1\servo_90, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora, processador e porta corretos, e faça o upload do código para a placa controladora ESP32.

Lição2 Montagem do Braço Robótico

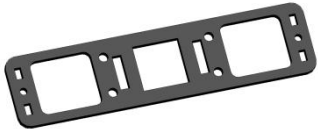

Antes da montagem, para facilitar o processo e distinguir os servos nas diferentes posições, nomearemos os servos de cima para baixo como Garra, Cotovelo, Ombro e Chassi.



I.Lista de Peças

 ESP32 Max V1.0 Controller Board 1PC	 Acrylic Board 1Set	 Servo MG90 9G 4PCS	 F-F 1P Dupont Wire 12PCS	 Joystick Module 2PCS	 USB Cable 1M 1PC
 18650 Battery Holder 1PC	 Nylon Cable Ties 2PCS	 Screwdriver 1PC	 L-Angled Socket Spanner 1PC	 Non-Slip Mat 6PCS	 Non-Slip Sleeve 4PCS
 M3*12MM Dual-pass Copper Pillar 9PCS	 M3*8MM Flat Head Screws 17PCS	 M3*10MM Flat Head Screws 24PCS	 M3*14MM Round Head Screws 5PCS	 M3 Nickel-Plated Nuts 24PCS	 M3 Nickel-Plated Lock Nuts 4PCS
 M2*10MM Round Head Screws 10PCS	 M2 Nickel-Plated Nuts 10PCS	 M1.7*6 Large Round Flat Head Tapping Screws 10PCS	 M3*22MM Flat Head Screws 2PCS	 M3*35 Nylon Column 1PC	 M3*40 Nylon Column 1PC
 M3*3 Nylon Gasket 6PCS	 M3*6 Nylon Gasket 2PCS	 Block 4PCS			

II.Lista de Componentes Estruturais

Estrutura de Acrílico	Quantidade	Imagem
Placa de base	1	
Suporte da placa de base	1	
	1	
	1	
Placa de montagem do joystick	1	
Placa de montagem do servo do chassi	1	
Pino	4	
Disco do chassi	1	
Suporte do disco 1	1	

Suporte do disco 2	1	
Suporte de ombro 1	1	
Suporte de ombro 2	1	
Suporte de cotovelo 1	1	
Suporte de cotovelo 2	1	
Placa de montagem do servo da garra	1	
Componente da garra 1	2	
Componente da garra 2	2	

III.Passos de Montagem

Atenção: Se você quiser assistir ao vídeo de montagem, clique no link abaixo.

<https://www.youtube.com/watch?v=RtOe7knGhkl>

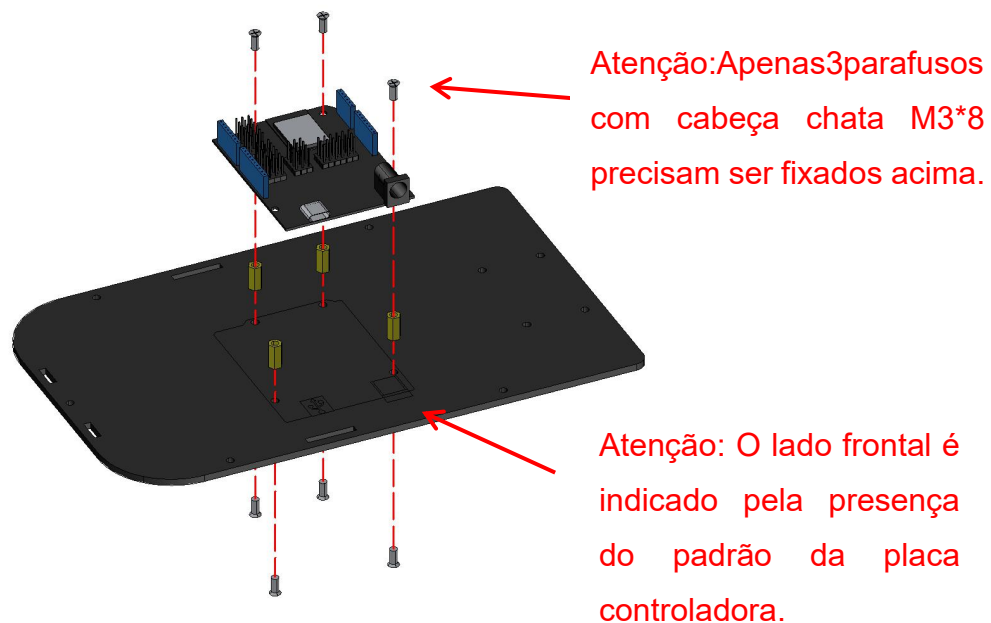
Ou escaneie o código QR abaixo.



1. Remova o papel protetor anexado à estrutura de acrílico

2. Instale a placa controladora ESP32 na placa base

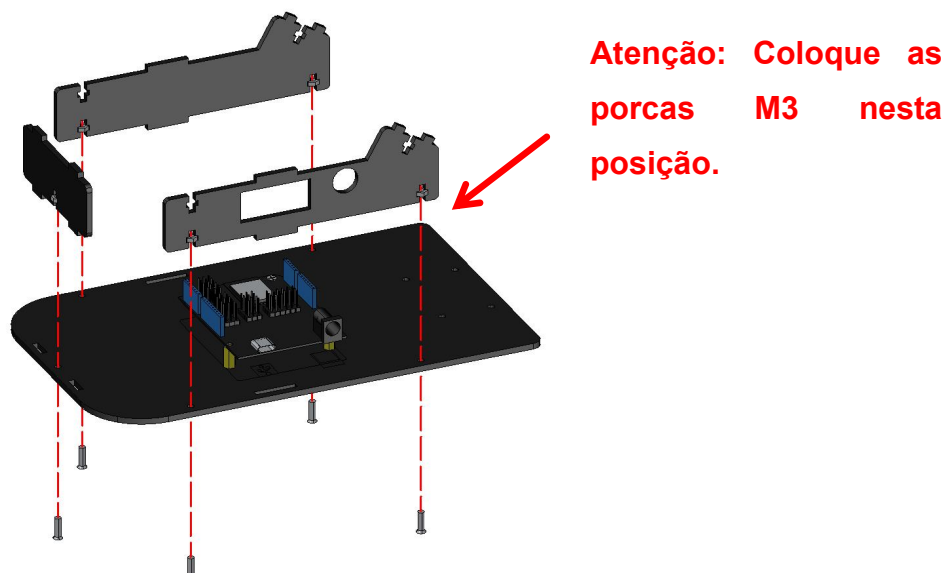
Lista de Componentes	
Nome	Quantidade
esp32 Controller Board	1
Base Plate	1
M3*8 Flat Head Screw	7
M3*12 Double-pass Copper Pillar	4



3.Instalando a Placa Base

Lista de Componentes	
Nome	Quantidade
Base Plate Bracket	3
M3*10 Flat Head Screw	5
M3 Nut	5

Atenção: Segure as porcas de ambos os lados com a mão antes de apertar os parafusos.

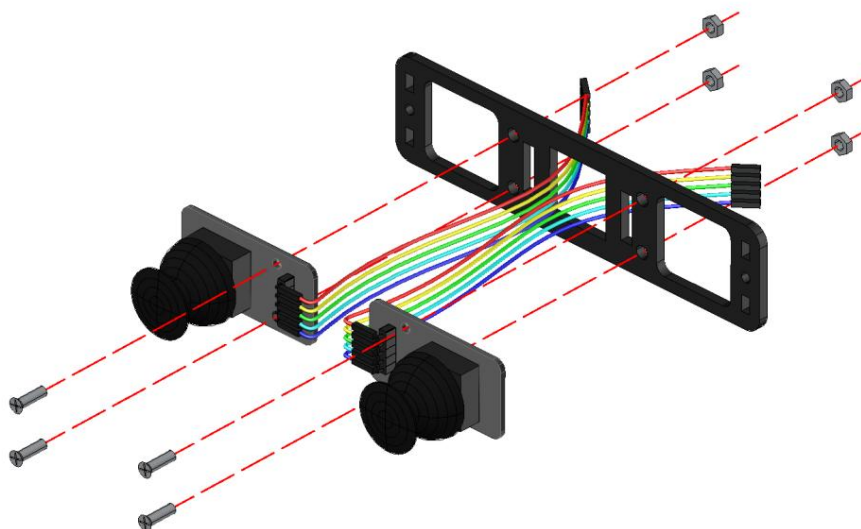


4.Instalando o módulo de joystick

Lista de Componentes	
Nome	Quantidade

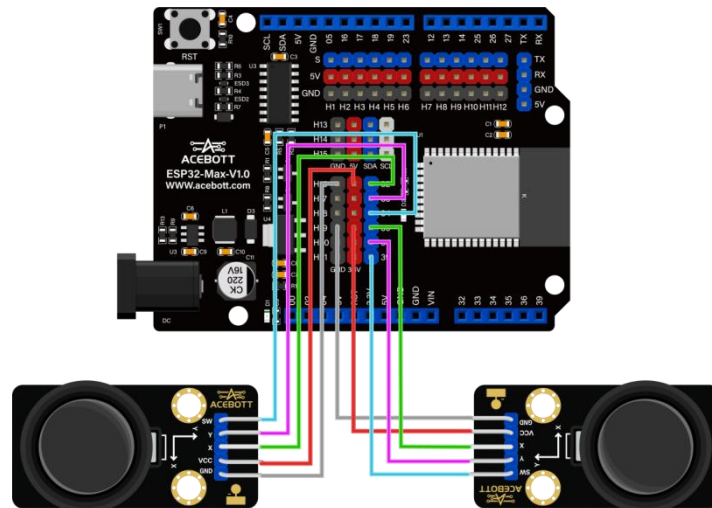
Joystick Module	2
Joystick Mounting Plate	1
M3*10 Flat Head Screws	4
M3 Nuts	4

Atenção: Quando o módulo de joystick estiver conectado ao fio Dupont, a cor real do fio Dupont pode ser diferente da figura, então, você só precisa seguir a identificação dos pinos no módulo de joystick para realizar a conexão.



Este passo requer conectar os fios dos dois módulos de joystick à placa principal primeiro. Para o módulo de joystick esquerdo, conecte SW ao pino 34, X ao pino 32, Y ao pino 33, e conecte VCC e GND aos pinos VCC e GND na mesma linha do pino 32. Para o módulo de joystick direito, conecte SW ao pino 39, X ao pino 35, Y ao pino 36, e conecte VCC e GND aos pinos VCC e GND na mesma linha do pino 35.

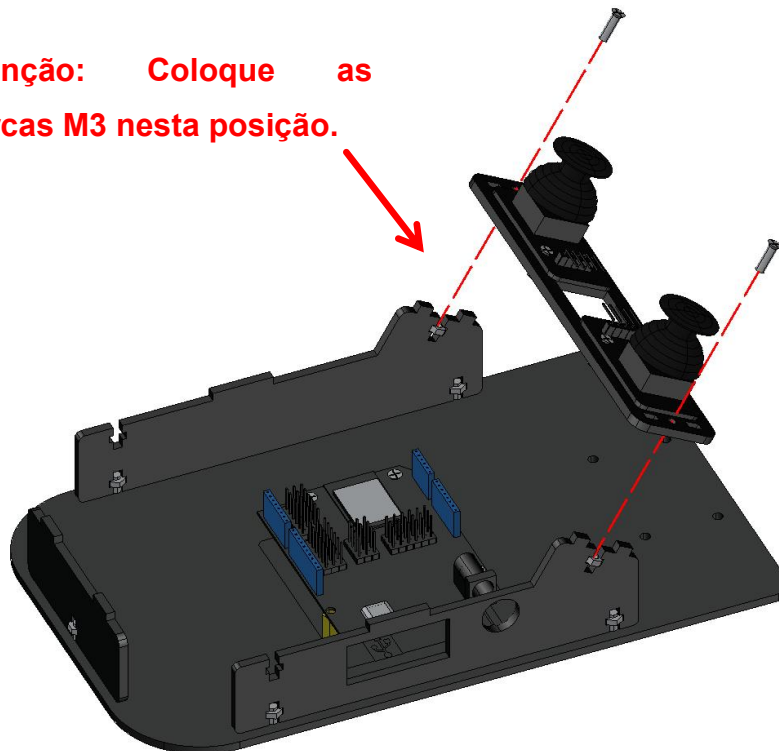
Atenção: Certifique-se de seguir rigorosamente as instruções de fiação ao conectar o módulo à placa controladora ESP32. Uma fiação incorreta pode causar um curto-circuito e danificar a placa controladora ESP32.



5.Instalar a placa de montagem do joystick

Lista de Componentes	
Nome	Quantidade
M3*10 Flat Head Screw	2
M3 Nut	2

Atenção: Coloque as porcas M3 nesta posição.

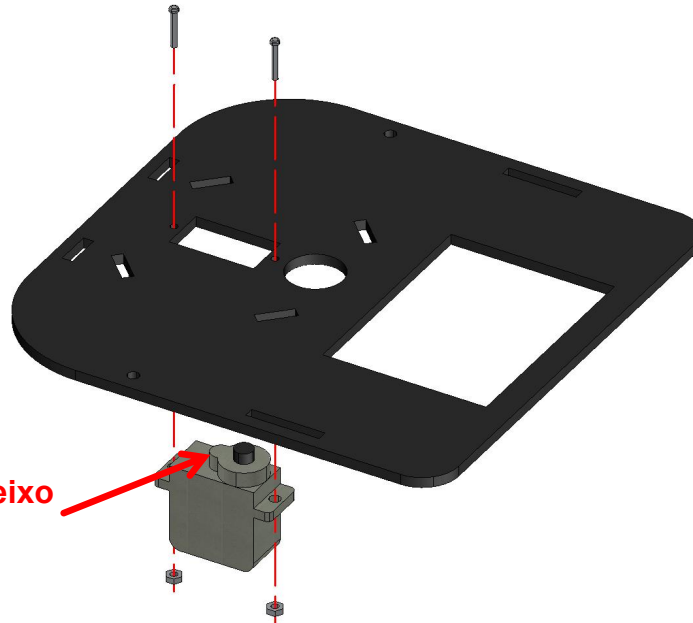


6.Instalar o servo do chassi

Lista de Componentes	
Nome	Quantidade

Servo	1
Chassis Servo Mounting Plate	1
M2*10 Round Head Screw	2
M2 Nut	2

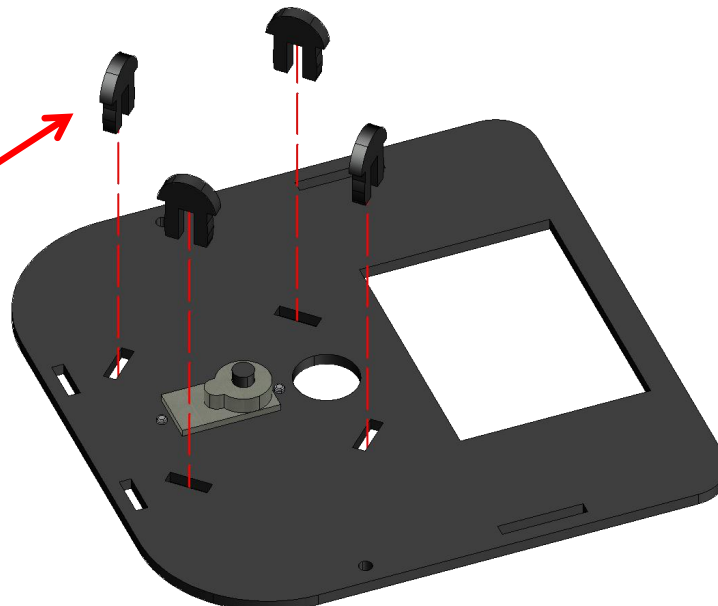
Atenção:
orientação do eixo
do servo.



7.Instalar o pino

Lista de Componentes	
Nome	Quantidade
Pin	4

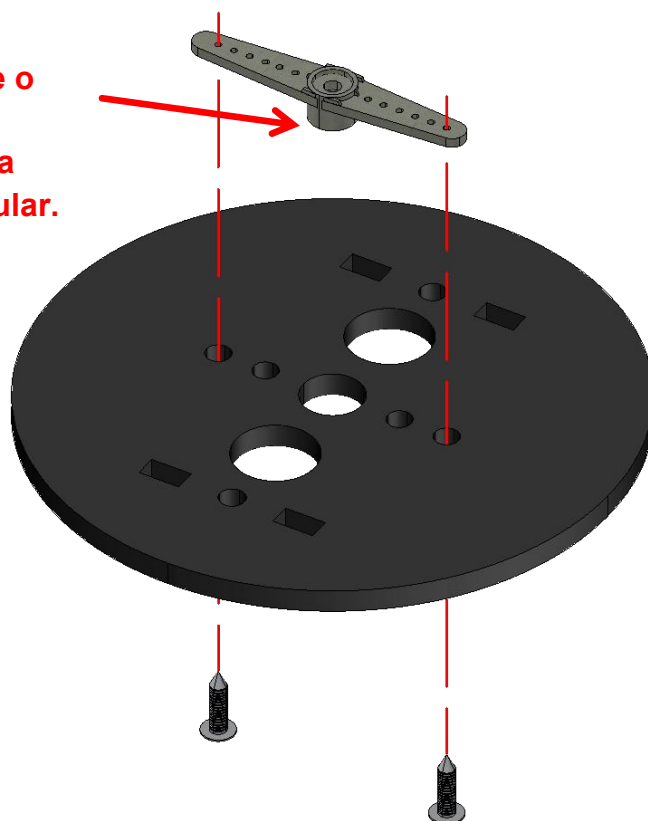
Atenção: Não
pressione o
pino com
muita força
para evitar
que ele
quebre.



8.Instalar o volante do servo do chassi

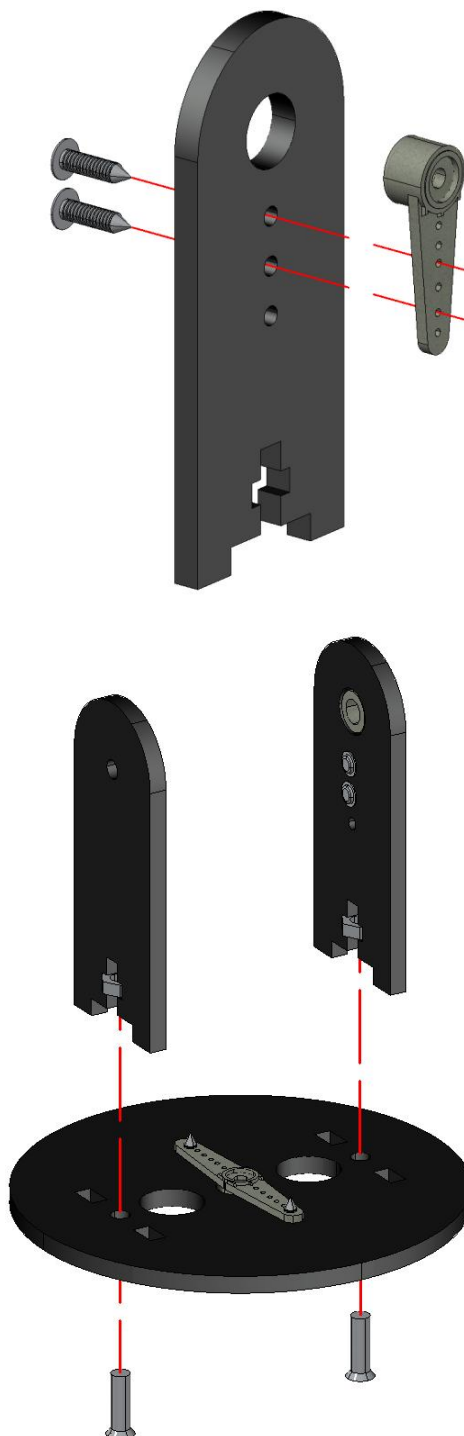
Lista de Componentes	
Nome	Quantidade
Straight Steering Wheel	1
Chassis Disk	1
M1.7*6 Large Round Flat Head Tapping Screw	2

Atenção: Instale o volante com a saliência voltada para o furo circular.



9.Instale o suporte do disco

Lista de Componentes	
Nome	Quantidade
Half Straight Steering Wheel	1
Disk bracket 1	1
Disk bracket 2	1
M1.7*6 Large Round Flat Head Tapping Screws	2
M3*10 Flat Head Screws	2
M3 Nuts	2

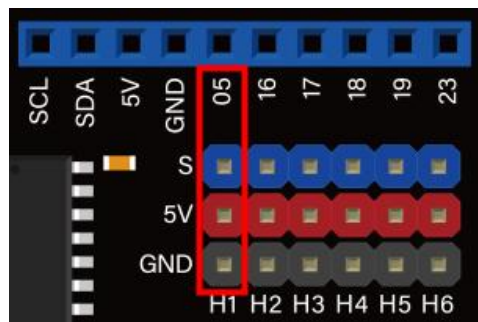


10.Fixe a estrutura do chassi do braço robótico

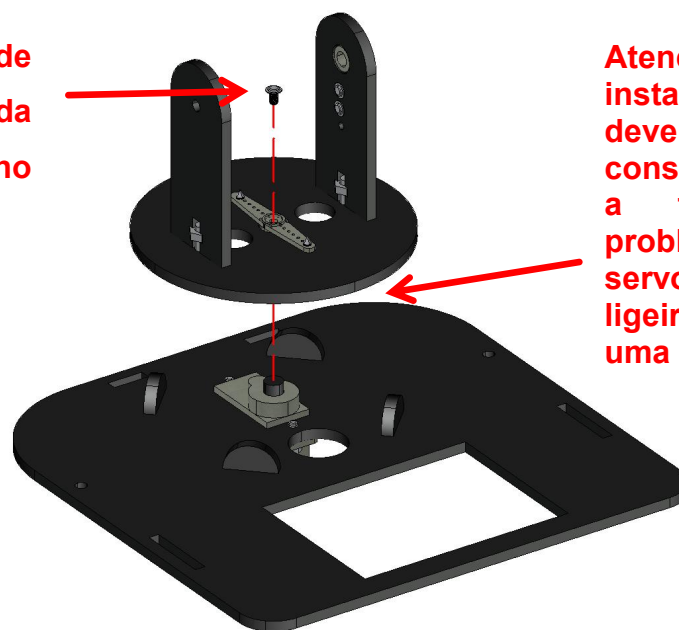
Lista de Componentes	
Nome	Quantidade
M2.5*4 Round Head Screw	1

Attention: ①Antes desta etapa de instalação, conecte o servo ao pino GPIO5 da placa de controle e, em seguida, ligue a alimentação da placa de controle para manter o servo na posição de 90°.

②Para evitar danos ao mecanismo de direção durante a instalação, não gire o eixo do mecanismo de direção.



Os parafusos de cabeça redonda M2.5*4 estão no saco do servo.



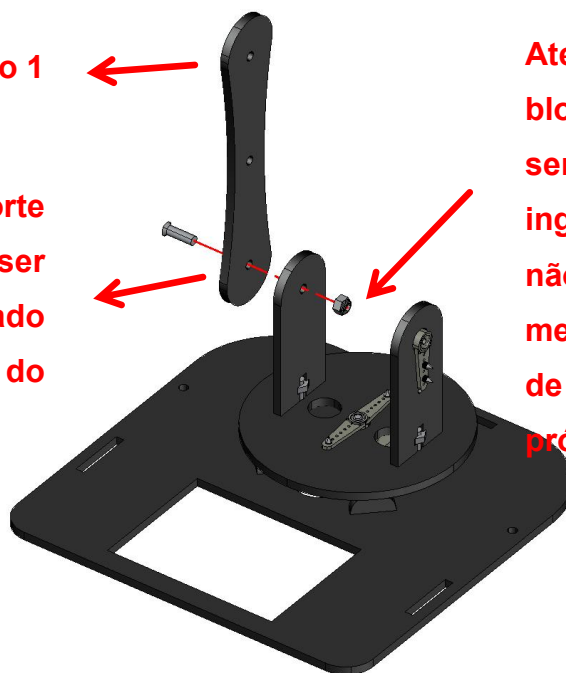
Atenção: A direção de instalação desta estrutura deve ser o mais consistente possível com a figura. Devido a problemas de precisão do servo, pode haver alguma ligeira desviação, o que é uma situação normal.

11.Instalar o suporte do ombro

Lista de Componentes	
Nome	Quantidade
Shoulder Bracket 1	1
Elbow Bracket 1	1
M3*35 Nylon Column	1
M3*10 Flat Head Screw	3
M3 Nickel-Plated Lock Nut	2

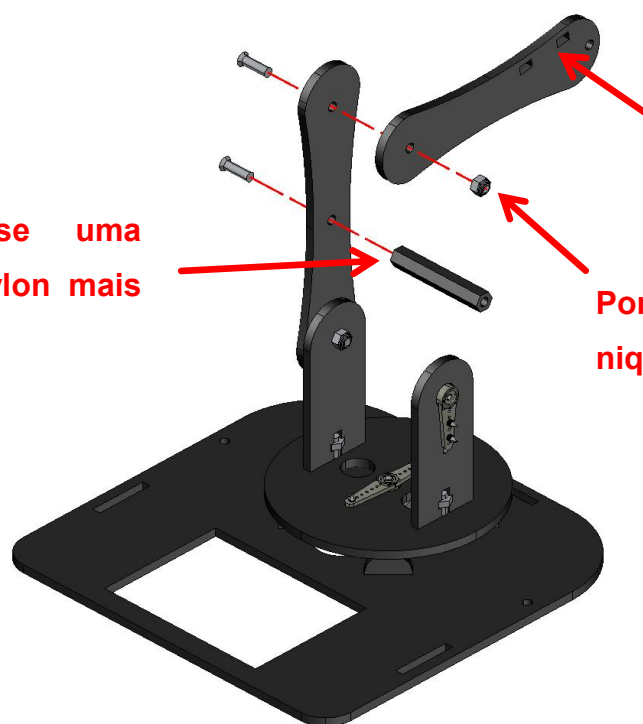
O suporte do ombro 1

Atenção: O suporte do ombro 1 deve ser fixado no lado externo do suporte do disco.



Atenção: A porca de bloqueio niquelada pode ser fixada com uma chave inglesa em ângulo, mas não a aperte demais. O mesmo vale para a porca de bloqueio niquelada no próximo passo.

Atenção: Use uma coluna de nylon mais curta aqui.

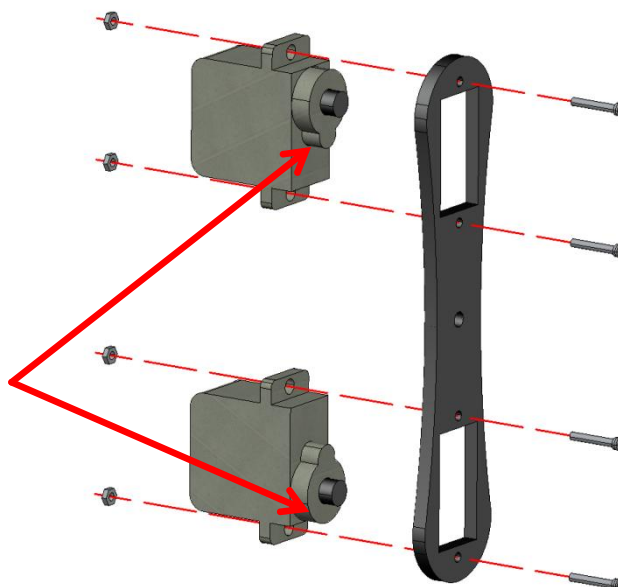


Porcas de trava niqueladas M3

Lista de Componentes

Nome	Quantidade
Servo	2
Shoulder Bracket 2	1
M2*10 Round head screw	4
M2 Nut	4

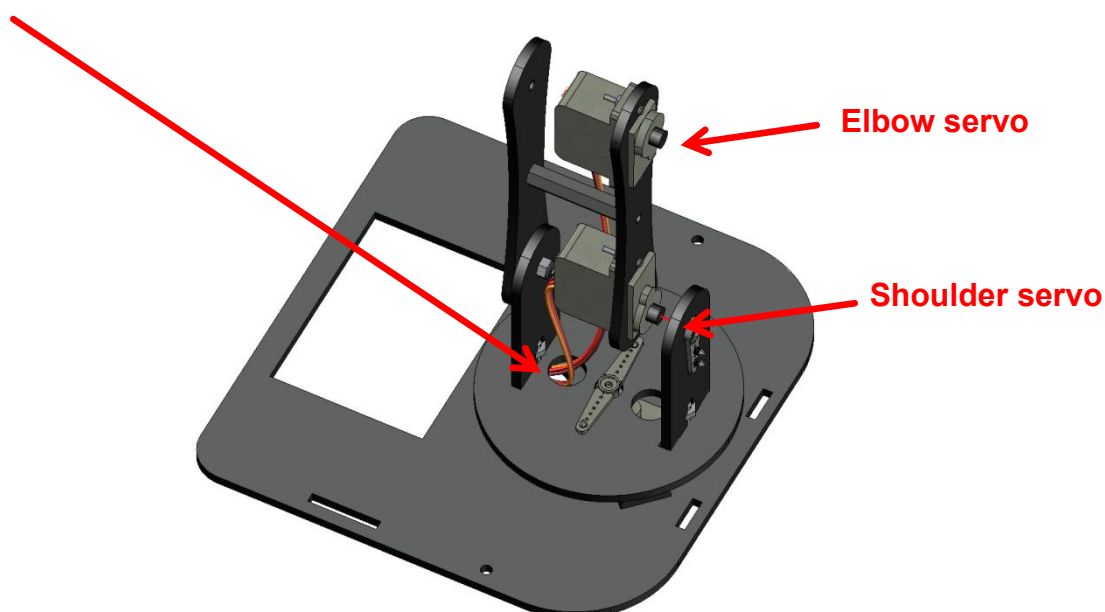
Atenção: A direção dos dois eixos de direção deve ser correta.



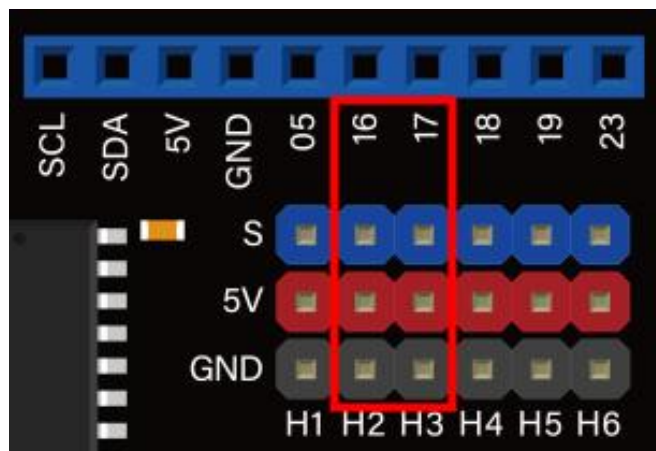
Conectando o servo do ombro e o servo do cotovelo ao ESP32.

Atenção: ①Primeiro, desconecte o cabo USB, desligue a fonte de alimentação da placa controladora e coloque o servo do braço robótico em um estado sem energia;

②Em seguida, gire suavemente o orifício roscado no disco do chassi para alinhá-lo com o orifício roscado na base. Passe os fios Dupont do servo do ombro e do servo do cotovelo do braço robótico pelos dois orifícios roscados.

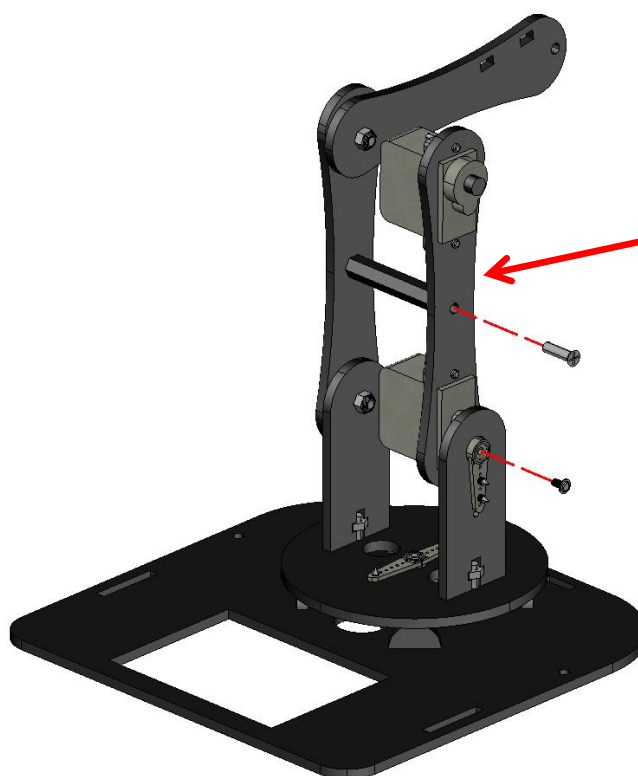


③Conecte o fio do servo do ombro ao pino GPIO16 e o fio do servo do cotovelo ao pino GPIO17.



④Em seguida, ligue a alimentação novamente e mantenha os servos da base, do ombro e do cotovelo na posição de 90°.

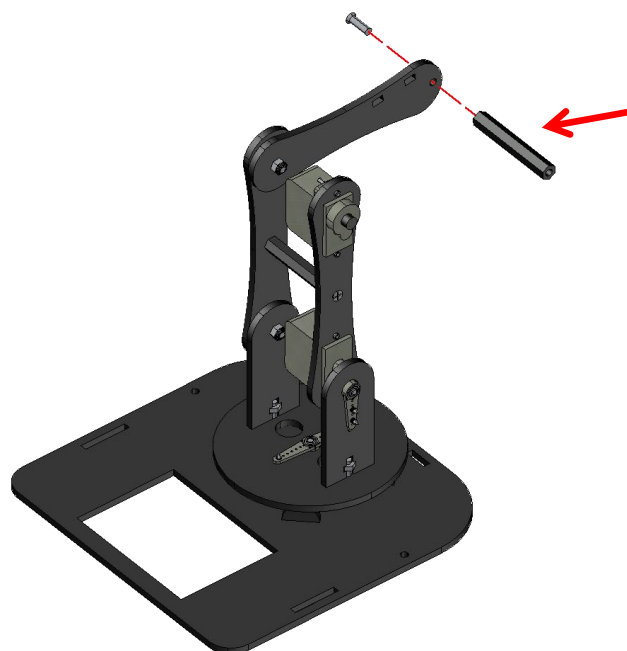
Lista de Componentes	
Nome	Quantidade
M2.54 Round Head Screw	1
M3*10 Flat Head Screw	1



Atenção: O suporte do ombro deve ser instalado verticalmente ao chão antes de apertar os parafusos do motor servo e da coluna de nylon.

12.Fixar a coluna de nylon M340

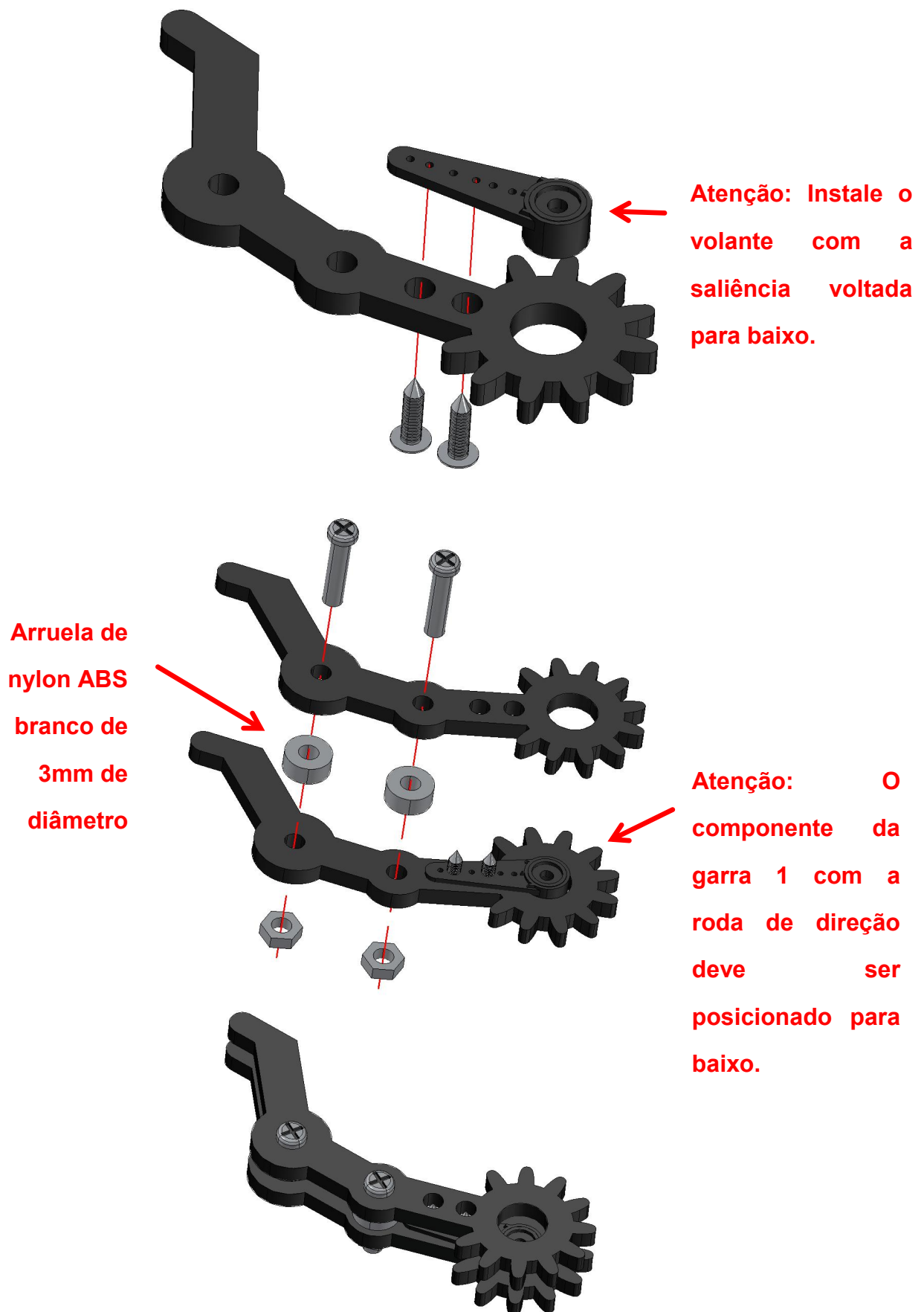
Lista de Componentes	
Nome	Quantidade
M3*40 Nylon Column	1
M3*10 Flat Head Screw	1

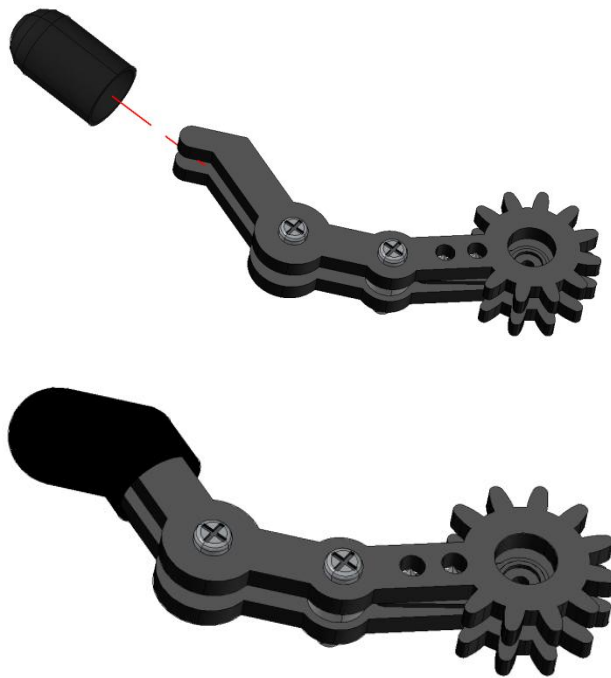


Atenção: Use
uma coluna de
nylon mais
longa aqui.

13.Instalar a estrutura da garra esquerda do braço robótico

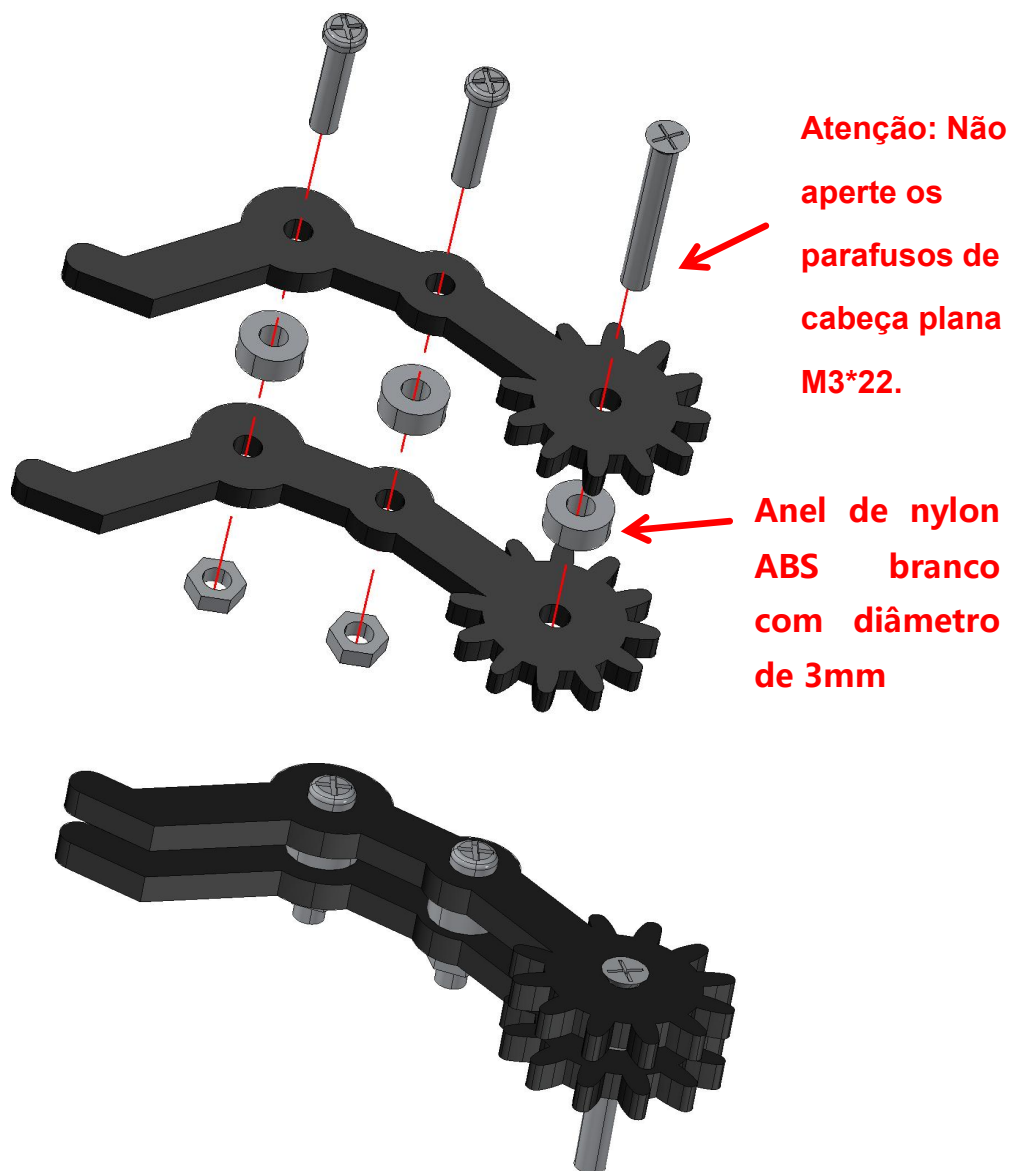
Lista de Componentes	
Nome	Quantidade
Half Straight Steering Wheel	1
Claw Component 1	2
M1.7*6 Large Round Flat Head Tapping Screws	2
M3*14 Round Head Screws	2
3MM White ABS Nylon Gaske	2
M3 Nuts	2
Non-Slip Sleeve	1



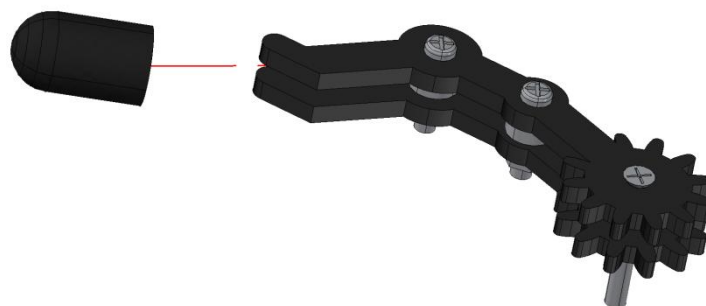


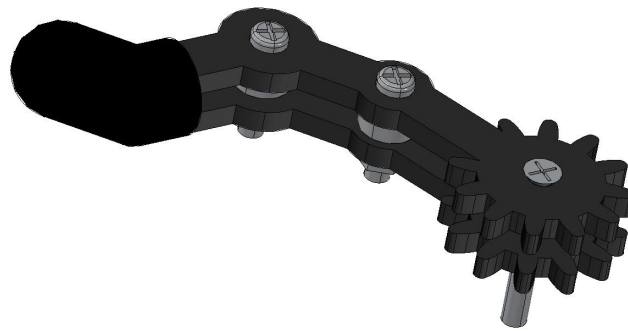
14.Instalar a estrutura da pata direita do braço robótico

Lista de Componentes	
Nome	Quantidade
Claw Component 2	2
M3*22 Flat Head Screws	1
M3*14 Round Head Screws	2
3MM White ABS Nylon Gaske	3
M3 Nuts	2
Non-Slip Sleeve	1



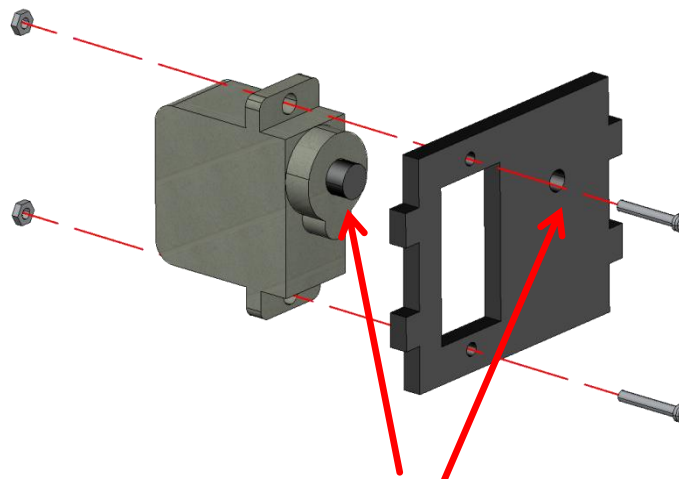
Instalar o tapete antiderrapante.





15.Fixar a placa de montagem do servo da garra do braço robótico

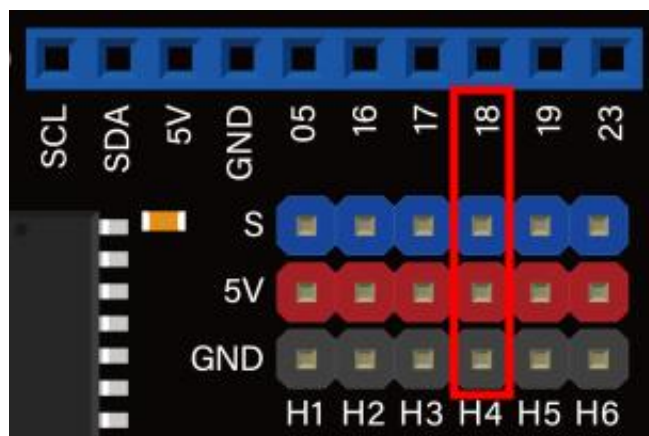
Lista de Componentes	
Nome	Quantidade
Servo	1
Claws Servo Mounting Plate	1
M2*10 Round Head Screw	2
M2 Nut	2



Atenção: Certifique-se de que o eixo da direção e o orifício circular estejam posicionados para cima.

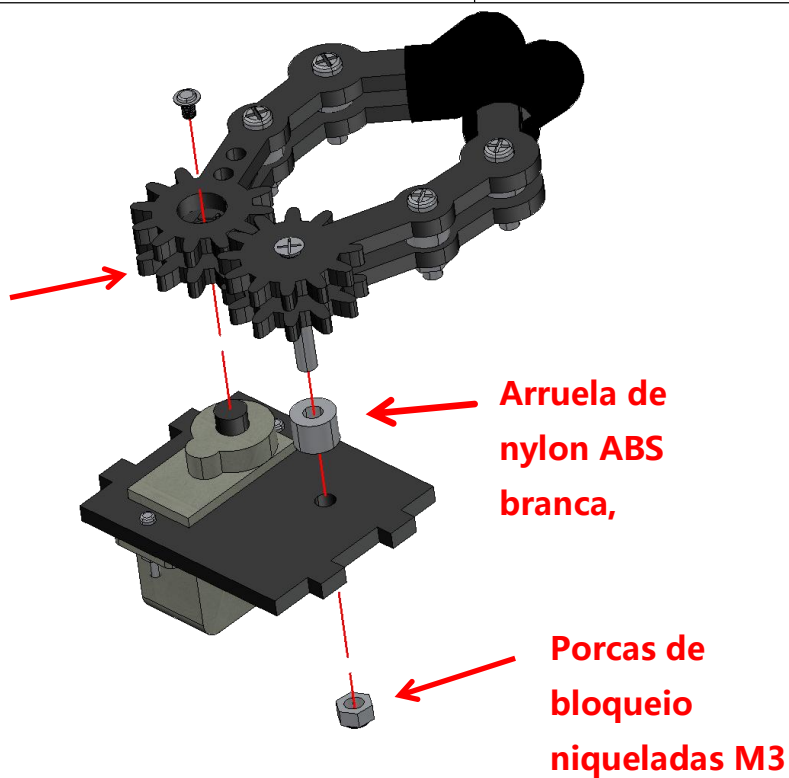
Conecte o fio do servo da garra ao pino GPIO18.

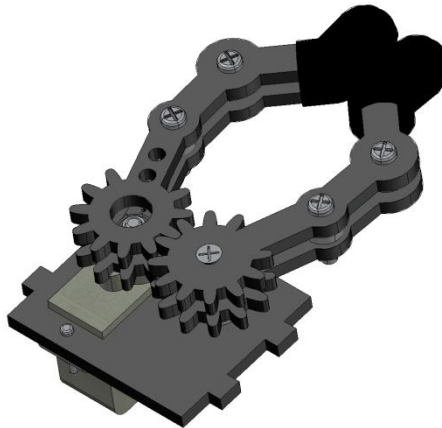
Atenção: Para esta etapa, por favor, ligue a placa controladora e certifique-se de que o servo da garra permaneça na posição de 90°.



Lista de Componentes	
Nome	Quantidade
M2.5*4 Round Head Screw	1
6MM White ABS Nylon Gaske	1
M3 Nickel-Plated Lock Nut	1

Atenção: A forma inicial da garra é fechada.

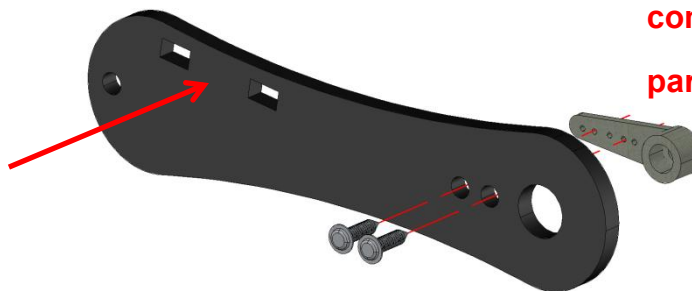




Lista de Componentes

Nome	Quantidade
Half Straight Steering Wheel	1
Elbow Bracket 2	1
M1.7*6 Large Round Flat Head Tapping Screws	2
M3*10 Flat Head Screw	1
M2.5*4 Round Head Screw	1

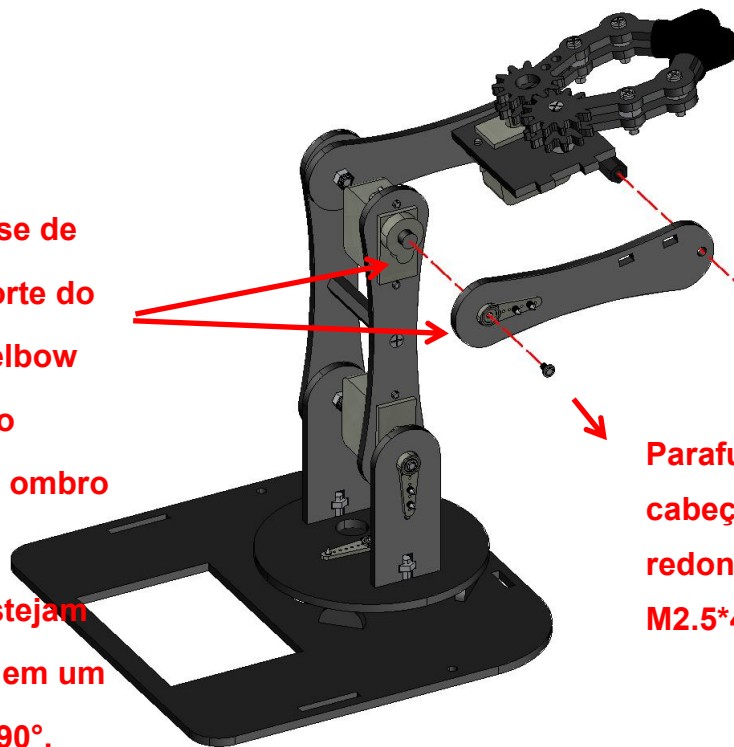
Atenção:
Certifique-se de que
o buraco retangular
esteja voltado para
cima.



Atenção: Instale o volante
com a saliência voltada
para o buraco circular.

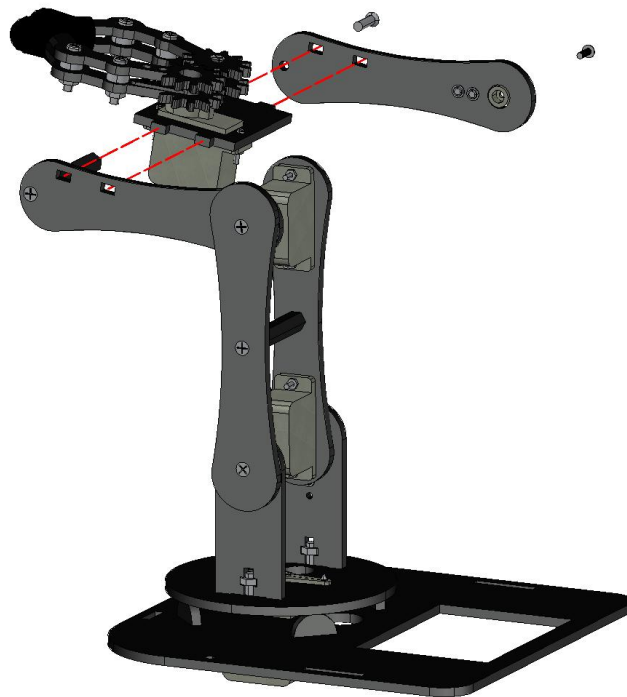
Atenção:

Certifique-se de que o suporte do cotovelo (elbow bracket) e o suporte do ombro (shoulder bracket) estejam instalados em um ângulo de 90°.



Parafuso de cabeça redonda M2.5*4

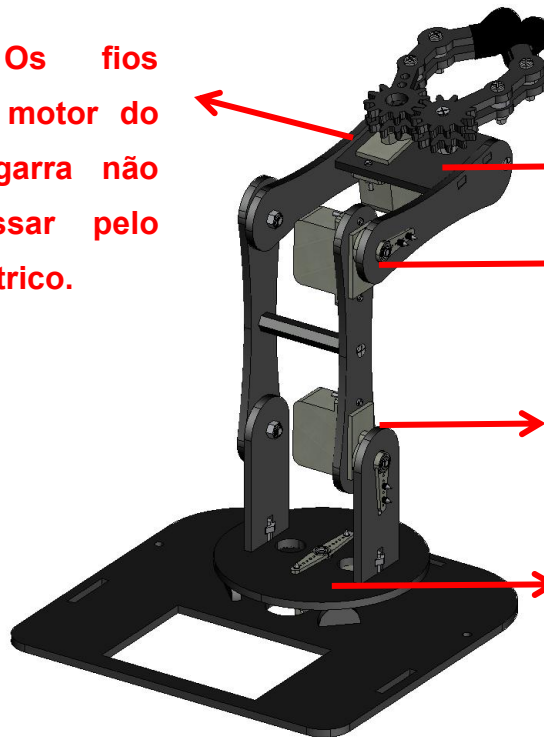
Parafuso de cabeça redonda M2.5*4



16.Organize os fios dos servos

Por favor, certifique-se de que os fios DuPont de todos os quatro servos estejam corretamente conectados à placa de controle.

Atenção: Os fios DuPont do motor do servo da garra não devem passar pelo furo concêntrico.

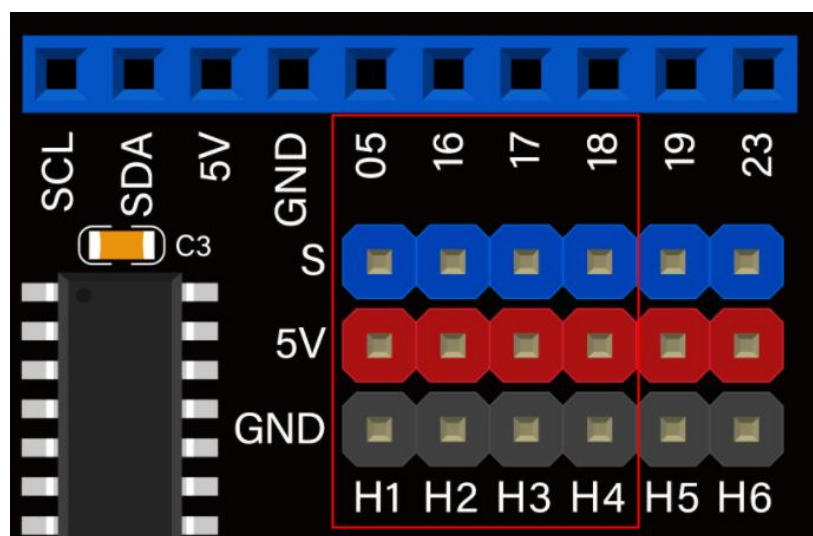


O motor servo da garra está conectado ao pino 18.

O motor servo do cotovelo está conectado ao pino 17.

O motor servo do ombro está conectado ao pino 16.

O motor servo do chassi está conectado ao pino 5.

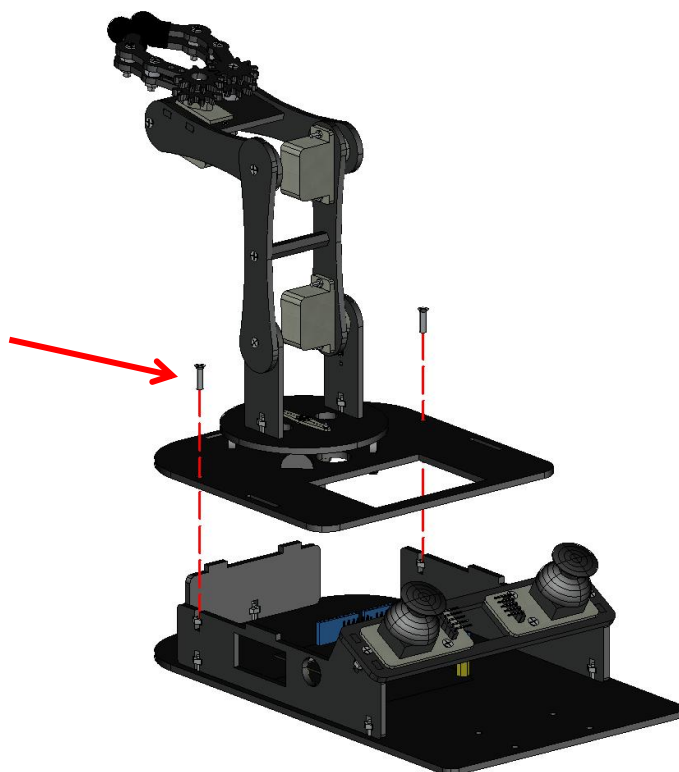




17.Fixação da base do braço robótico

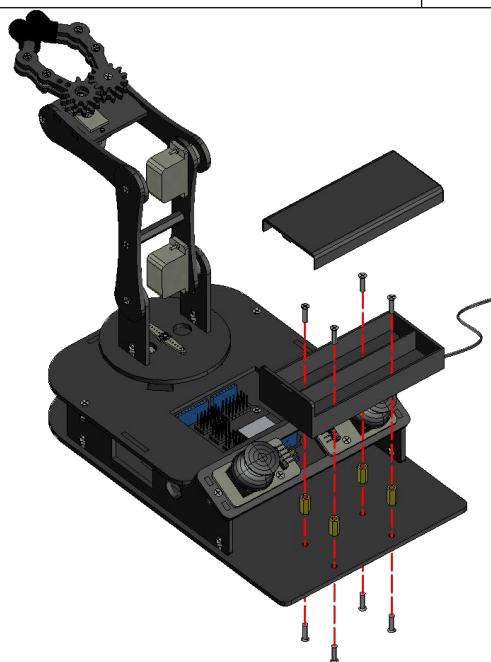
Lista de Componentes	
Nome	Quantidade
M3*10 Flat Head Screws	2
M3 Nuts	2

Atenção:
Certifique-se de
que todos os fios
dos quatro
motores de servo
estejam
conectados à
placa controladora
antes de apertar
os parafusos.



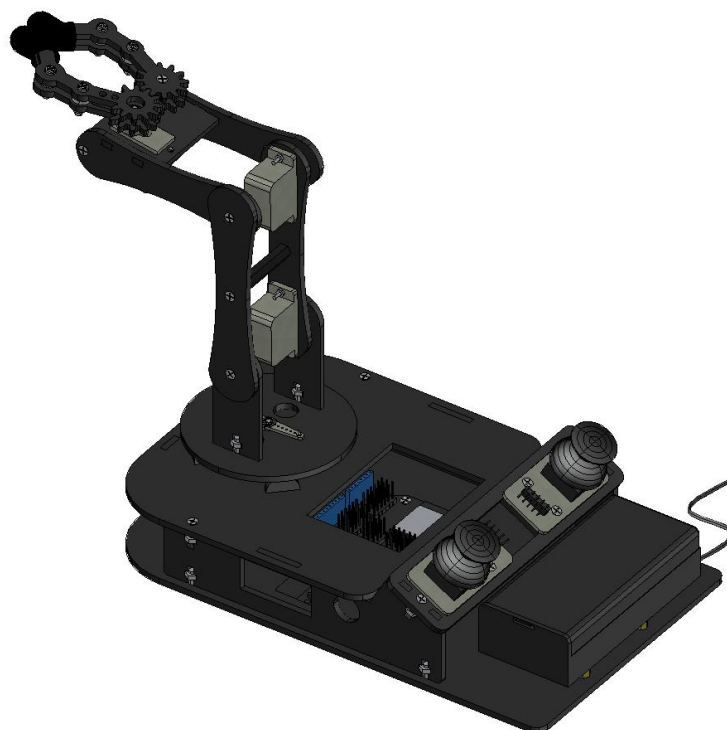
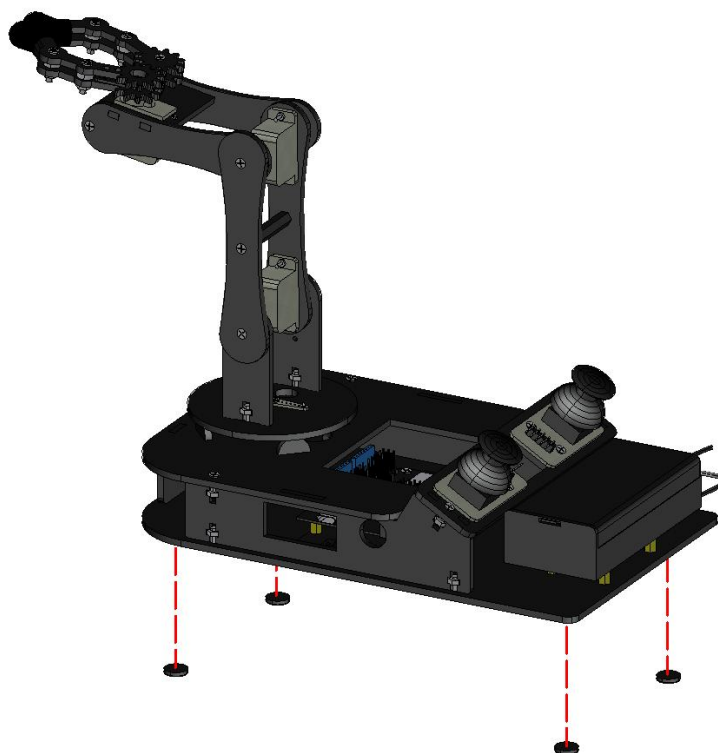
18.Fixação do Suporte da Bateria

Lista de Componentes	
Nome	Quantidade
18650 Battery Holder	1
M3*8 Flat Head Screws	8
M3*12 Double-pass Copper Pillar	4



19.Instalar o Tapete Antiderrapante

Retire o filme adesivo do tapete antiderrapante e cole o tapete nas quatro extremidades na parte inferior da base do braço robótico.



O braço robótico está agora totalmente montado!

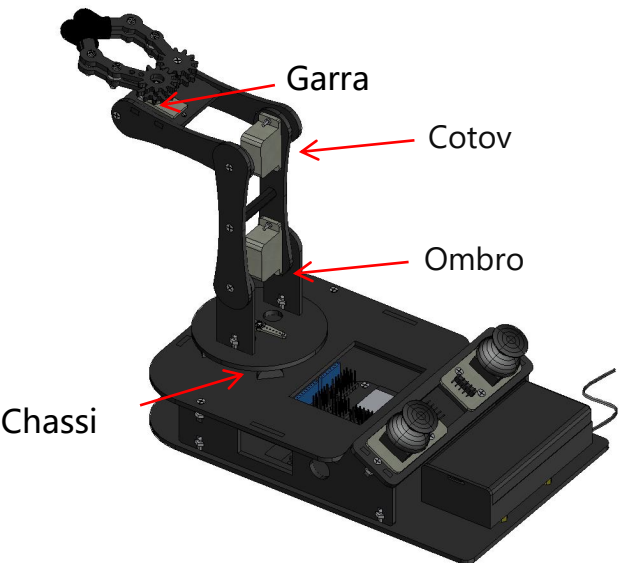
Lição 3: Controle do Braço Robótico com Joystick

I. Controle dos Servos

1. Descrição dos pinos dos servos do braço robótico

Após instalar o braço robótico, cada servo assume funções diferentes. Para controlar o ângulo de cada servo programaticamente e obter diversas funcionalidades, é essencial conhecer os números dos pinos correspondentes a cada servo.

O braço robótico utiliza um total de 4 servos, com os números dos pinos correspondentes conforme abaixo:

NO.	Número do pino	Posição do servo	Figura
1	GPIO18	Garras	
2	GPIO17	Cotovelo	
3	GPIO16	Ombro	
4	GPIO5	Chassi	

2. Princípios de movimento de cada junta no braço robótico

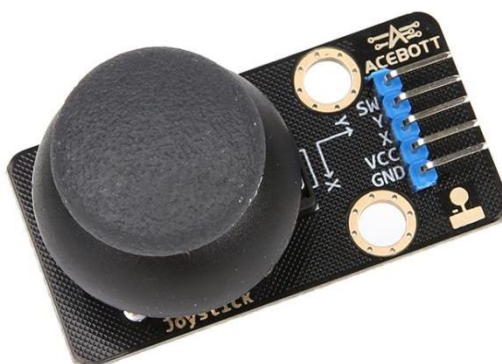
NO.	Número do pino	Posição do servo	Padrões de movimento
1	GPIO18	Garras	Quanto maior o ângulo do servo, mais ampla a garra se abre
2	GPIO17	Cotovelo	Quanto maior o ângulo do servo, mais alto o cotovelo do braço robótico se move para cima
3	GPIO16	Ombro	Quanto maior o ângulo do servo, mais baixo o ombro do braço robótico se move para baixo

4	GPIO5	Chassi	Quanto maior o ângulo do servo, mais a base do braço robótico gira para a esquerda
---	-------	--------	------------------------------------------------------------------------------------

II.Compreendendo o Módulo Joystick

1.Introdução ao Módulo Joystick

O módulo joystick é composto por dois potenciômetros deslizantes e um botão. Quando você move o joystick, os valores de resistência dos potenciômetros mudam, resultando nos valores de tensão correspondentes nos eixos X/Y. Pressionar o joystick aciona o botão, fazendo com que o sinal SW se torne baixo. Este módulo é comumente utilizado em aplicações como modelos RC, controles de videogame, veículos controlados remotamente e estabilizadores de câmera.



2.Testando a Saída Serial do Módulo Joystick

Quando você move o joystick, pode monitorar os valores correspondentes de X/Y em tempo real imprimindo-os no Monitor Serial da IDE do Arduino.

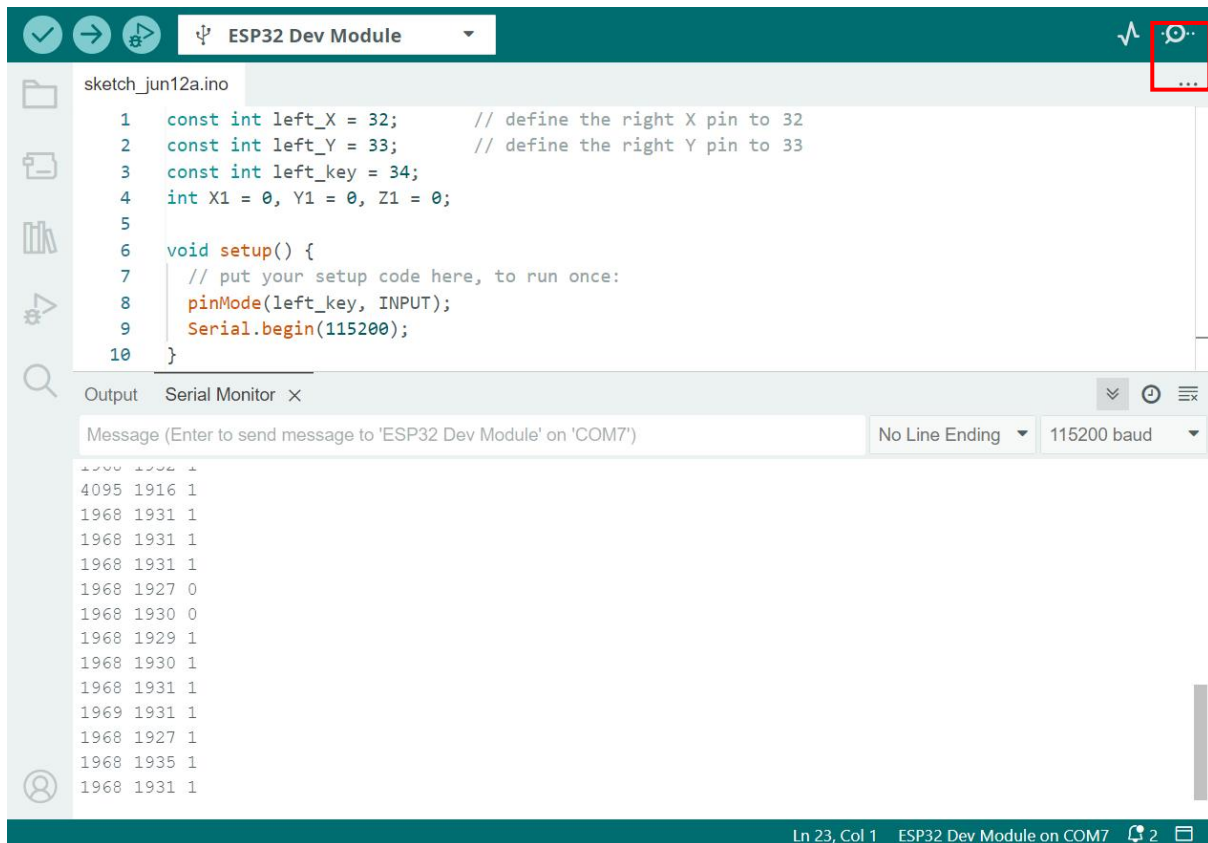
Abra o arquivo '[Joystick_test.ino](#)' em português\ACECode (Iniciante)\2.ACECode Program\Lição 3\Joystick_test, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora correta, o processador e a porta, e faça o upload do código para a placa controladora ESP32.

Código de Exemplo:

```
const int left_X = 32; //Define the right X pin to 32
const int left_Y = 33; //Define the right Y pin to 33
const int left_key = 34;
int X1 = 0, Y1 = 0, Z1 = 0;

void setup() {
  pinMode(left_key, INPUT);
  Serial.begin(115200);
}

void loop() {
  X1 = analogRead(left_X); //Read the right X value
  Serial.print(X1);
  Y1 = analogRead(left_Y); //Read the right Y value
  Serial.print(" ");
  Serial.print(Y1);
  Z1 = digitalRead(left_key); //Read the right Z value
  Serial.print(" ");
  Serial.println(Z1);
  delay(1000);
}
```



III.Movimentos Básicos do Braço Robótico Controlado pelo Joystick

As ações básicas de controle de um braço robótico incluem principalmente rotação para a esquerda e para a direita, movimento do cotovelo para cima e para baixo, e abertura e fechamento da garra. Após dominar esses movimentos básicos, você pode combinar e expandir outras ações com base nesses fundamentos.

Atenção: Após ligar o braço robótico, é proibido girar o servo diretamente com a mão para evitar danos ao servo.

1.Programa de Controle pelo Joystick

Abra o arquivo '[JoyStick Controlled Robot Arm.ino](#)' em português\ACECode (Iniciante)\2.ACECode Program\Lição 3\JoyStick_Controlled_Robot_Arm, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora correta, o processador e a porta, e faça o upload do código para a placa controladora ESP32.

Código de Exemplo:

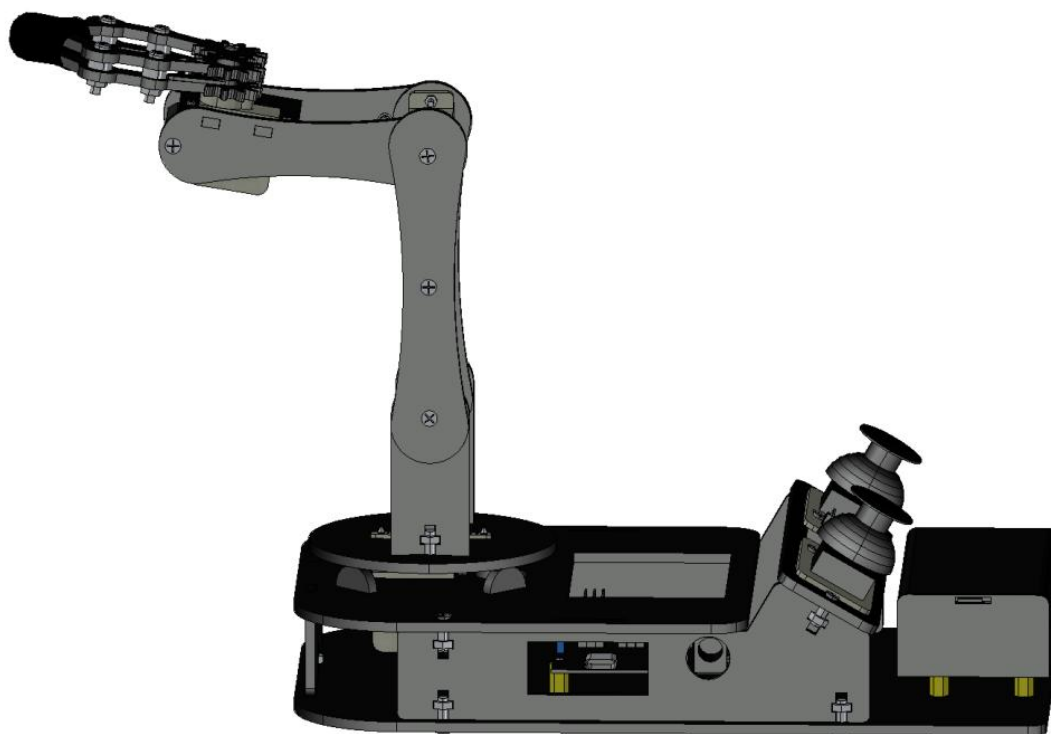
```
#include <ACB_ARM.h> //Add the Robot arm libraries
ACB_ARM ARM;

void setup() {
  ARM.ARM_init(5,16,17,18); //The parameters are four Servo pins
  ARM.JoyStick_init(32,33,34,35,36,39); //Joystick initialization
}

void loop() {
  ARM.get_JoyStick();
  if (ARM.JoyY1 < 50) { //chassis left
    ARM.chassis_angle = ARM.chassis_angle + 1;
    ARM.JoyChassisCmd(ARM.chassis_angle);
  }
  if (ARM.JoyY1 > 3500) { //chassis right
    ARM.chassis_angle = ARM.chassis_angle - 1;
    ARM.JoyChassisCmd(ARM.chassis_angle);
  }
  if (ARM.JoyX1 < 50) { //Shoulder down
    ARM.shoulder_angle = ARM.shoulder_angle + 1;
    ARM.JoyShoulderCmd(ARM.shoulder_angle);
  }
  if (ARM.JoyX1 > 4000) { //Shoulder up
    ARM.shoulder_angle = ARM.shoulder_angle - 1;
    ARM.JoyShoulderCmd(ARM.shoulder_angle);
  }
  if (ARM.JoyX2 < 50) { //Elbow up
    ARM.elbow_angle = ARM.elbow_angle + 1;
    ARM.JoyElbowCmd(ARM.elbow_angle);
  }
  if (ARM.JoyX2 > 4000) { //Elbow down
    ARM.elbow_angle = ARM.elbow_angle - 1;
    ARM.JoyElbowCmd(ARM.elbow_angle);
  }
  if (ARM.JoyY2 > 4000) { // Claws open
    ARM.claws_angle = ARM.claws_angle + 1;
    ARM.JoyClawsCmd(ARM.claws_angle);
  }
  if (ARM.JoyY2 < 50) { // Claws close
    ARM.claws_angle = ARM.claws_angle - 1;
    ARM.JoyClawsCmd(ARM.claws_angle);
  }
}
```

2. Padrões de controle pelo Joystick

Após fazer o upload do programa, você notará que os quatro braços do braço robótico estão dispostos em forma de "7", que é a postura inicial. Você pode então usar o joystick para controlar o braço robótico.



A seguir, uma introdução às funções do joystick:

Joystick	Direção	Posição da articulação	Lei do movimento
Esquerdo joystick	Esquerda	Chassi	Virar para a esquerda
	Direita	Chassi	Virar para a direita
	Cima	Ombro	Movimento para cima
	Baixo	Ombro	Movimento para baixo
Direito joystick	Cima	Cotovelo	Movimento para cima
	Baixo	Cotovelo	Movimento para baixo
	Esquerda	Garras	Garras abertas
	Direita	Garras	Garras fechadas

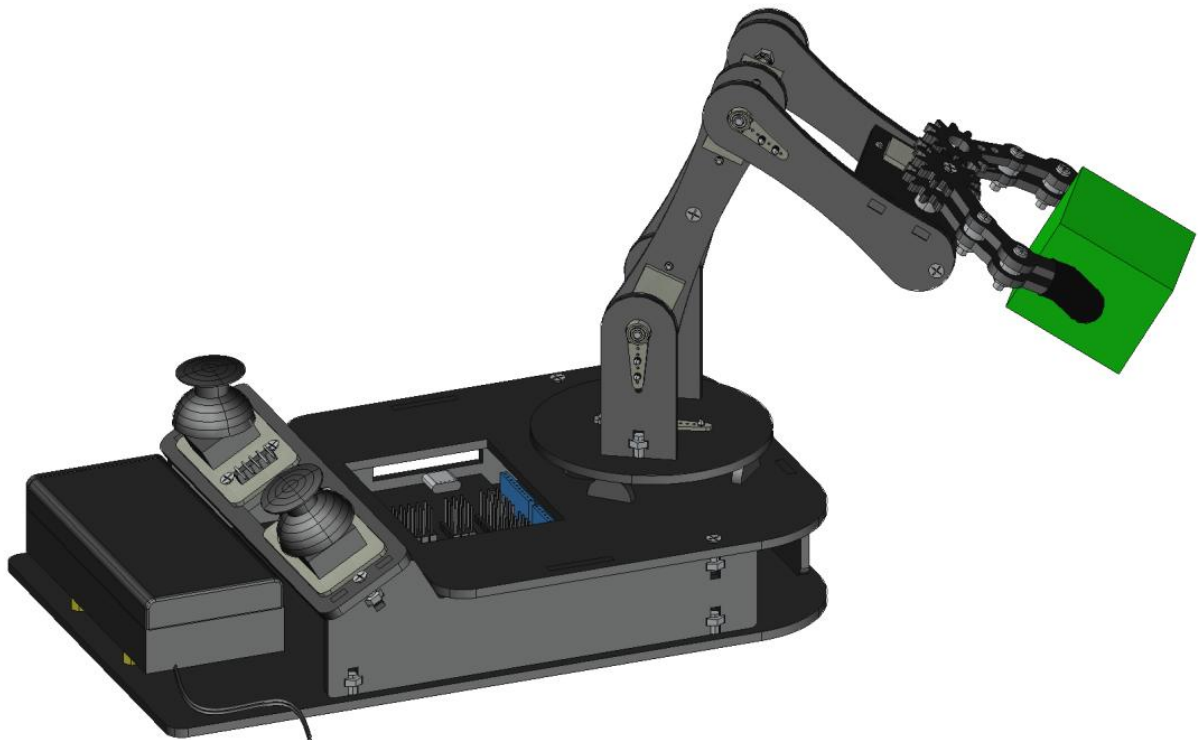
IV.Tarefas de Expansão

Com base nos padrões operacionais básicos do joystick no braço robótico, a seguir podemos alcançar a funcionalidade de usar o joystick para controlar o braço robótico em tarefas de manipulação de objetos.

Descrição da Tarefa:

Usando o método de controle do módulo joystick, manobre o braço robótico para pegar um objeto do ponto A e transportá-lo até o ponto B para colocação.

Atenção: Defina as posições dos pontos A e B conforme desejado.

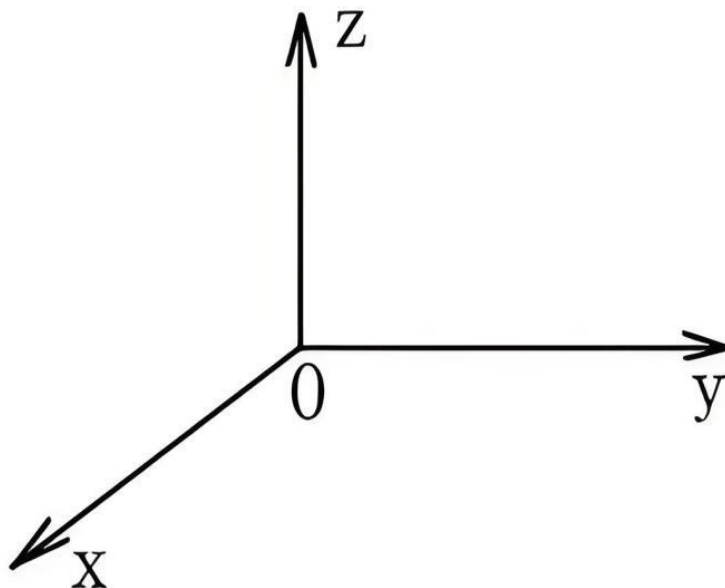


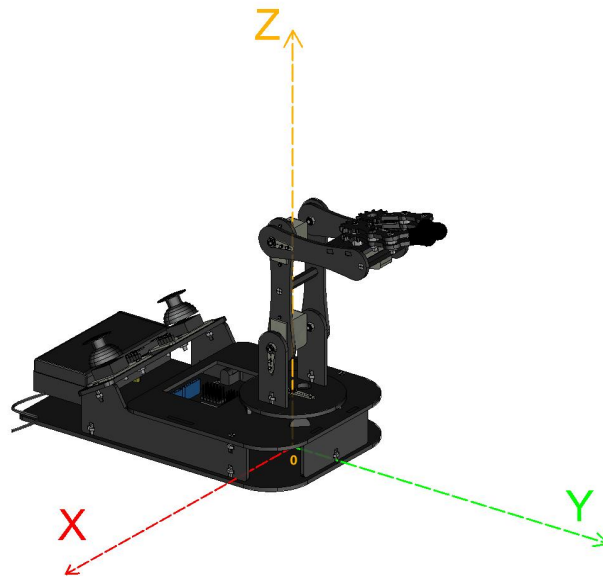
Lição 4: As Coordenadas Espaciais do Braço Robótico

As coordenadas espaciais do braço robótico desempenham um papel crucial em seu controle e programação. Através de coordenadas espaciais precisas, o braço robótico pode alcançar um posicionamento preciso, otimizar o planejamento de movimentos, evitar obstáculos de forma eficaz, realizar operações precisas e, assim, melhorar os níveis de automação e inteligência.

I.O Sistema de Coordenadas Cartesianas

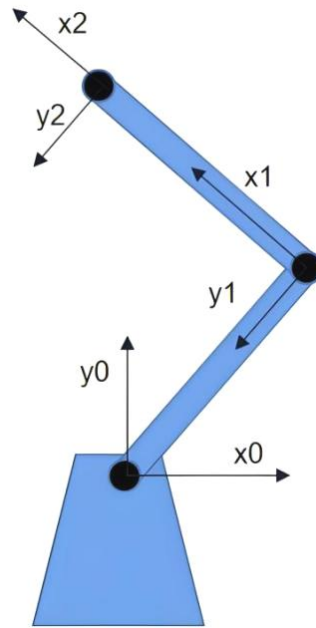
No controle de movimento espacial do braço robótico, o sistema de coordenadas cartesianas é um dos sistemas de coordenadas mais comumente utilizados. Trata-se de um sistema matemático para descrever a posição de pontos no espaço. No espaço tridimensional, o sistema de coordenadas cartesianas é composto por três eixos mutuamente perpendiculares (x , y , z). A interseção desses três eixos é a origem (O) do sistema de coordenadas. Neste tutorial, a origem do sistema de coordenadas cartesianas está localizada no centro do disco do servo do chassi do braço robótico.





II.Sistema de Coordenadas da Junta

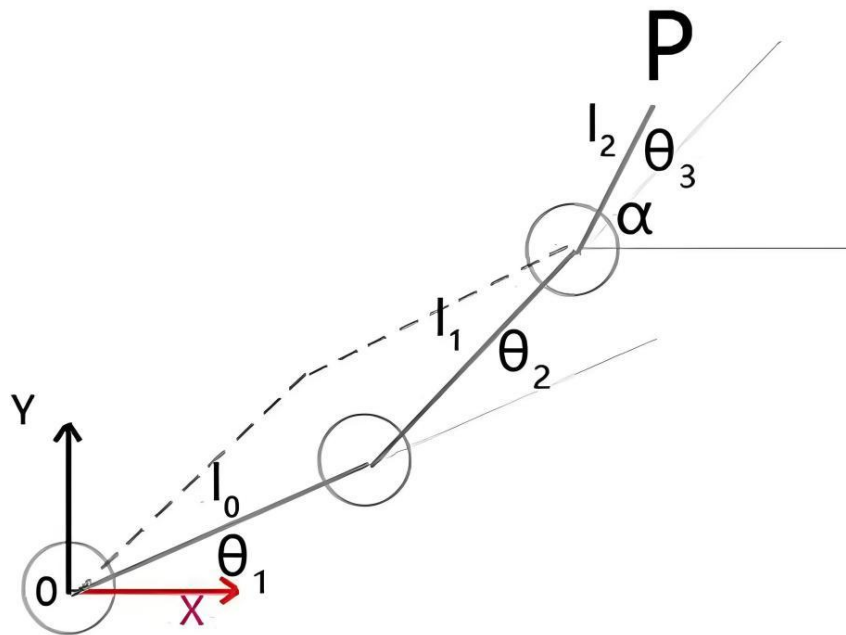
Além do sistema de coordenadas cartesianas, cada junta do braço robótico possui seu próprio sistema de coordenadas, conhecido como sistema de coordenadas da junta. Sua origem geralmente está localizada no ponto de conexão da junta, e seus eixos são definidos ao longo do eixo de rotação da junta. Cada sistema de coordenadas da junta está associado a uma coordenada de junta que descreve o ângulo de rotação ou a extensão daquela junta. Como cada junta do braço robótico pode girar ou se estender, os sistemas de coordenadas das juntas podem se transformar com base na postura atual do braço robótico.



III. Cinemática Direta e Inversa

A cinemática direta refere-se ao cálculo da posição e orientação do braço robótico em coordenadas cartesianas com base nos ângulos das juntas. A cinemática inversa, por outro lado, envolve o cálculo dos ângulos das juntas com base nas coordenadas cartesianas para alcançar o movimento desejado do braço robótico.

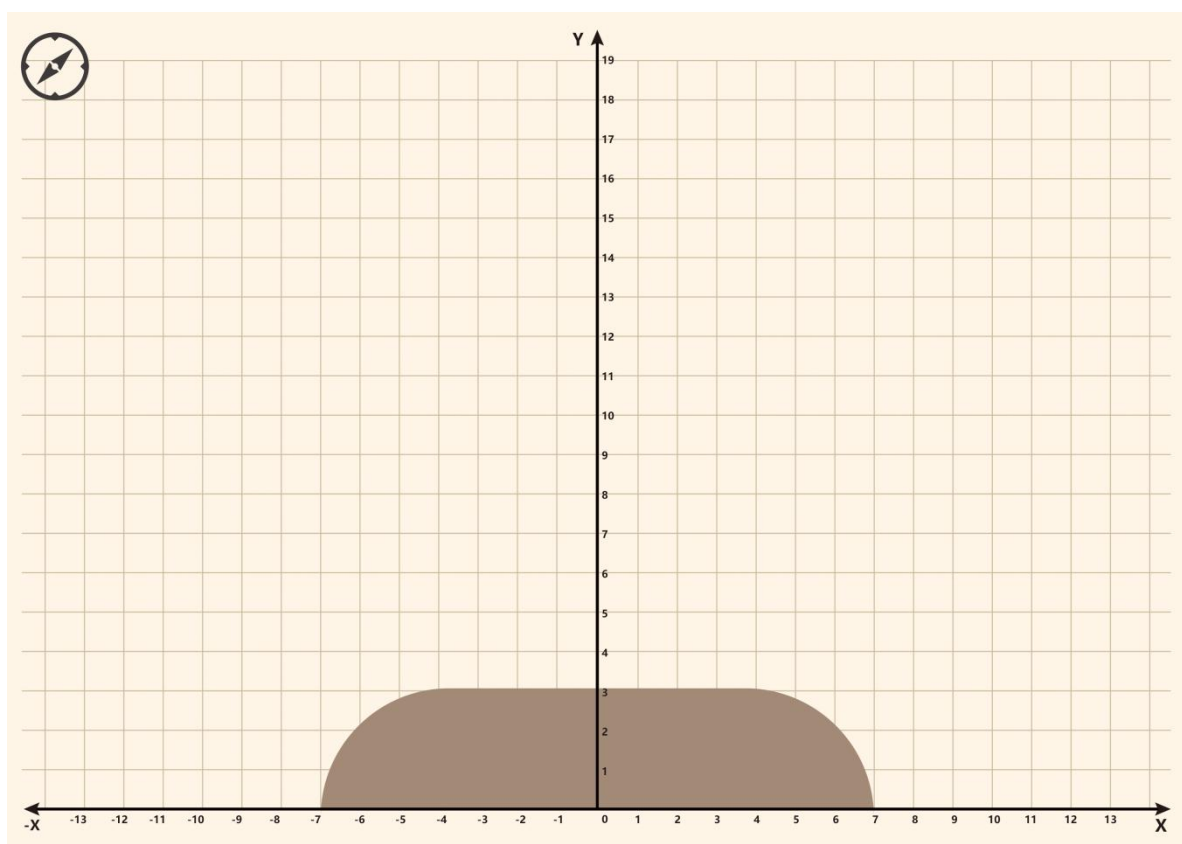
O problema da cinemática direta pode ser resolvido por meio de transformações matriciais, ou seja, comparando a matriz de transformação de cada sistema de coordenadas das juntas com as fórmulas de transformação de coordenadas do braço robótico para determinar os ângulos de cada junta. O problema da cinemática inversa é geralmente mais complexo, exigindo a solução de um conjunto de equações não lineares e, muitas vezes, apresentando múltiplas soluções.



Neste tutorial, iremos construir principalmente um sistema de coordenadas cartesianas com o centro da roda de direção do chassi como o ponto de origem. Quando as coordenadas de posição especificadas (X, Y, Z) forem inseridas, o arquivo de biblioteca do tutorial utilizará automaticamente o algoritmo de cinemática inversa para calcular as coordenadas de cada junta do braço mecânico e, em seguida, convertê-las nos ângulos de cada servo de junta. Assim, a extremidade do manipulador será controlada para alcançar a posição do ponto alvo nas coordenadas espaciais.

IV. Diagrama de Coordenadas do Braço Robótico

Inevitavelmente, haverá alguns erros no processo de montagem do braço robótico. Para calibrar melhor o braço robótico, precisamos usar o diagrama de coordenadas. O diagrama de coordenadas do braço robótico é composto pelas coordenadas X e Y, com o ponto de interseção de X e Y sendo a origem do mapa. O intervalo de coordenadas de X é $[-13, 13]$, e o intervalo de coordenadas de Y é $[0, 19]$. O retângulo chanfrado na área sombreada é a posição de referência para o braço robótico. Coloque a borda superior da base do braço robótico contra essa forma.



[【 Clique para obter o arquivo PDF do diagrama de coordenadas do braço robótico 】](#)

Atenção: Por favor, imprima o diagrama do braço robótico em papel A4 de acordo com o arquivo PDF.

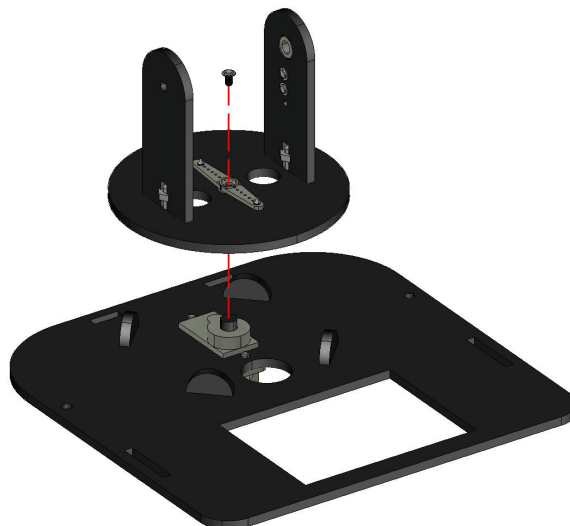
V.Instruções de Calibração do Braço Robótico

O braço robótico é composto por múltiplos servos, e no trabalho prático, pode haver certos erros. Para reduzir esses erros, adicionamos instruções no programa de controle do braço robótico que permitem o ajuste dos erros do braço.

```
ARM.Chassis_angle_adjust(8); // Default 0  
ARM.Slight_adjust(0,2); // Default 0,0
```

Primeiro, o comando 'ARM.Chassis_angle_adjust(8)' é usado para ajustar a variação do ângulo central do servo do chassi. O valor padrão do ponto central é 90 graus, com ângulos variando de 90 a 180 graus no lado esquerdo do eixo -X, e de 0 a 90 graus no lado direito do eixo X.

No entanto, devido à precisão dos engrenagens durante a instalação, não é garantido que o servo do chassi e o servo de direção estarão exatamente a 90 graus. Eles podem estar ligeiramente inclinados para a esquerda ou para a direita. Portanto, nesse ponto, precisamos calibrá-los no programa. Por exemplo, se estiver inclinado 8 graus para a direita, precisamos aumentar o valor de desvio. Assim, nos parênteses do comando 'ARM.Chassis_angle_adjust()', você deve inserir 8. Se o deslocamento for de 8 graus para a esquerda, então você deve subtrair o valor do desvio, ou seja, 'ARM.Chassis_angle_adjust()' com -8 nos parênteses.



Em seguida, temos o comando 'ARM.Slight_adjust(0,2)'. Esse comando possui dois parâmetros, ambos com valor padrão 0. Suponha que haja um pequeno erro de

desvio quando o efetor final do braço robótico atinge um ponto espacial especificado. Nesse caso, este comando é usado para fazer ajustes finos.

O primeiro parâmetro corresponde ao semieixo positivo X do braço robótico. Se o efetor final inclinar 1 grau para a direita, você deve escrever 1; se inclinar 1 grau para a esquerda, deve escrever -1. Se não houver inclinação, você escreverá 0.

O segundo parâmetro corresponde ao semieixo negativo X do braço robótico. Se o efetor final inclinar 1 grau para a direita, você deve escrever 1; se inclinar 1 grau para a esquerda, deve escrever -1. Se não houver inclinação, você escreverá 0.

Agora, por favor, consulte o diagrama de coordenadas e, de acordo com as coordenadas espaciais abaixo do programa e a sua situação real, ajuste os parâmetros de calibração.

VI.Movendo Pontos de Coordenadas Espaciais

Abra o arquivo " [Inverse Kinematics.ino](#) " em português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 4\Inverse_Kinematics, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora correta, o processador e a porta, e faça o upload do código para a placa controladora ESP32.

Código Exemplo:


```
#include <ACB_ARM.h> //Add the Robot arm libraries
ACB_ARM ARM;

void setup() {
  ARM.Chassis_angle_adjust(8);
  ARM.Slight_adjust(0,2);
  ARM.ARM_init(5,16,17,18); //The parameters are four Servo pins
  Serial.begin(115200);
}

void loop() {
  if (Serial.available()) {
    String input = Serial.readStringUntil('\n');
    int x, y, z, claws; //Define variables for coordinates x, y, z, and claws
    sscanf(input.c_str(), "%d %d %d %d", &x, &y, &z, &claws);
    if (claws < 90 || claws > 180) {
      Serial.println("Invalid claw value. Claw value must be between 90 and 180.");
      return;
    }
    Serial.print("X:");
    Serial.print(x);
    Serial.print(",Y:");
    Serial.print(y);
    Serial.print(",Z:");
    Serial.print(z);
    Serial.print("  Claws Angle:");
    Serial.print(claws);
    Serial.println(" ");
    ARM.ClawsCmd(claws);
    delay(1000);
    ARM.PtpCmd(x, y, z);
  }
}
```



Após fazer o upload do programa, insira quatro valores no monitor serial: a coordenada X, a coordenada Y, a coordenada Z e o ângulo de abertura da garra (variando de 90 graus a 180 graus). Esses quatro valores precisam ser separados por espaços. Após inserir os valores, pressione Enter.

ARM.PtpCmd(x, y, z) é um comando de controle de coordenadas do braço robótico. Através deste comando, o efector final do braço robótico se move para a posição especificada nas coordenadas espaciais, onde x é a coordenada do eixo X, y é a coordenada do eixo Y e z é a coordenada do eixo Z.

ARM.JoyClawsCmd(claws) é um comando de controle da garra do braço robótico. Este comando controla a abertura e o fechamento da garra do efector final, onde "claws" representa o parâmetro de ângulo para o servo da garra, com uma faixa de entrada de 90 a 180 graus.

Se as coordenadas estiverem corretas, o braço robótico se moverá até o ponto especificado no espaço. Se o monitor serial exibir "Out of range!" após a inserção das coordenadas, isso significa que os valores inseridos excedem o alcance do braço,

pois a faixa de movimento está dentro de uma área esférica. Nesse caso, você precisará ajustar os valores e reentrá-los.

Lição 5: Empilhamento com Braço Robótico

Na era atual de rápido desenvolvimento tecnológico, os braços robóticos se tornaram uma parte indispensável da indústria moderna, dos serviços comerciais e da vida cotidiana. Com sua notável flexibilidade, precisão e eficiência, eles transformaram completamente os modos de operação tradicionais.

Especialmente, a tecnologia de paletização com braço robótico é amplamente utilizada em diversos campos, particularmente em cenários que exigem manuseio de itens com alta eficiência, onde seu valor de aplicação é especialmente importante. Por exemplo, em logística e armazenagem, a tecnologia de paletização com braço robótico pode melhorar significativamente a eficiência e a precisão do manuseio de mercadorias, reduzir os custos com mão de obra e minimizar erros humanos. Comparado à paletização manual, a tecnologia de paletização com braço robótico oferece vantagens como alta eficiência, boa estabilidade e facilidade de operação.



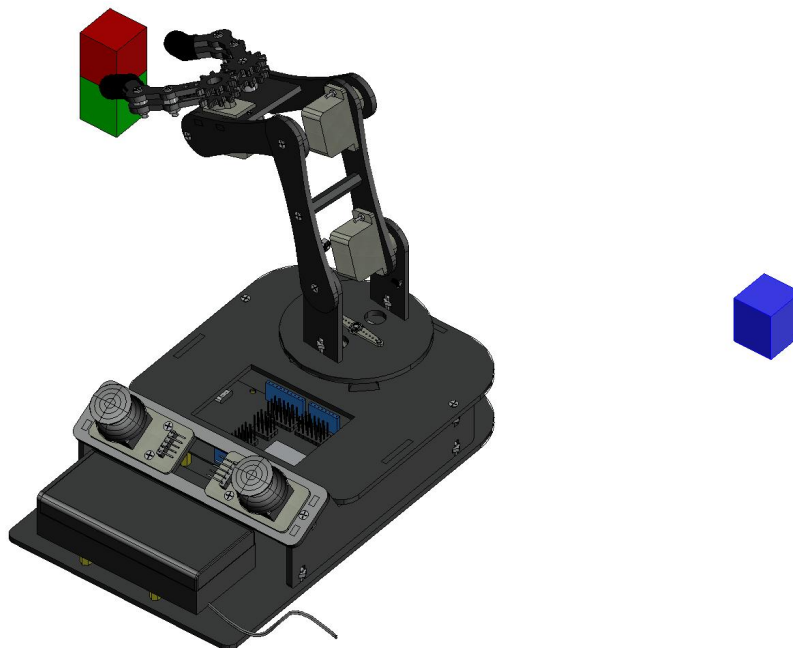
Com o avanço da tecnologia e a demanda do mercado, a tecnologia de paletização com braço robótico verá aplicações ainda mais amplas no futuro. Portanto, nesta

lição, aprenderemos a implementar funções básicas de paletização usando um braço robótico.

I. Programa de Paletização com Braço Robótico

Abra o arquivo “ [Robot_Arm_Stacking.ino](#) ” em português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 5\Robot_Arm_Stacking, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora correta, o processador e a porta, e faça o upload do código para a placa controladora ESP32.

Primeiro, empilhe dois blocos verticalmente e coloque-os nas coordenadas (-7,13) no mapa do braço robótico, garantindo que o ponto central dos blocos esteja alinhado com o ponto de coordenadas.



Código de Exemplo:

```
#include <ACB_ARM.h> //Add the Robot arm libraries
ACB_ARM ARM;

bool RunningState = true;

int startx = 7, starty = 13, startz = 2; //Initial coordinates
int midz = 15; //Height of middle point
int endx = -7, endy = 13, endz = 0; //End point coordinates
int openAngle = 130, closeAngle = 90;
int count = 2; //Number of blocks
int i = 0;

void setup() {
  ARM.Chassis_angle_adjust(8); //Chassis error calibration
  ARM.Slight_adjust(0,2); //Small calibration error
  ARM.ARM_init(5,16,17,18); //The parameters are four Servo pins
  Serial.begin(115200);
}

void loop() {
  while(i < count) {
    ARM.ClawsCmd(openAngle); //open claws
    delay(1000);
    ARM.PtpCmd(startx, starty, startz-i*2); //The target point is the upper object
    delay(1000);
    ARM.ClawsCmd(closeAngle); //close claws
    delay(1000);
    ARM.PtpCmd(startx, starty, midz); //Lift the object after picking it up
    delay(1000);
    ARM.PtpCmd(endx, endy, midz); //Rotate above the target point
    delay(1000);
    ARM.PtpCmd(endx, endy, endz+i*2); //Placing Objects
    delay(1000);
    ARM.ClawsCmd(openAngle);
    delay(1000);
    ARM.PtpCmd(endx, endy, midz); //Raise arms
    delay(1000);
    i = i + 1;
  }
  ARM.Zero(); //Servo initialization
}
```

Após fazer o upload do programa, podemos observar que o efetor final do braço robótico atingirá as coordenadas iniciais dos blocos, em seguida, pegará o bloco superior, o transportará para as coordenadas de destino e o colocará no local. Em seguida, ele retornará para as coordenadas iniciais dos blocos, pegará o bloco inferior, o transportará para as coordenadas de destino e, finalmente, o empilhará em cima do primeiro bloco.

II.Tarefas de Expansão

Com base no mapa de coordenadas do braço robótico, já conhecemos o padrão de paletização do braço robótico. Desde que saibamos as coordenadas de posição no mapa, podemos fazer o braço robótico completar a função de paletização. A seguir, utilizaremos o mapa para completar a função de paletização do braço robótico.

Descrição da tarefa:

Com base no exemplo do programa de paletização, tente modificar as coordenadas da posição inicial e as coordenadas do ponto final dos objetos no programa para permitir que o braço robótico realize a paletização em diferentes posições.

Atenção: Os dois blocos precisam ser colocados empilhados juntos. Garanta que as coordenadas inseridas não excedam o intervalo do mapa. Além disso, durante o processo de captura do braço robótico, pode haver alguns erros devido a questões de posicionamento ou precisão.

Lição 6: Ensino e Aprendizado do Braço Robótico

Ensinar um braço robótico envolve o operador definir um caminho de movimento fixo para o braço robótico seguir, permitindo que ele trabalhe de acordo com etapas predefinidas.

Ensinar um braço robótico pode ser dividido em três etapas. A primeira etapa é a ação de ensino, onde o operador define um caminho de movimento fixo para o braço robótico. A segunda etapa é a ação de armazenamento, onde o sistema de controle do braço robótico grava as ações ensinadas. A terceira etapa é a reprodução do ensino, onde o braço robótico repete as ações registradas durante o processo de ensino.

Nesta lição, usaremos o módulo joystick para ensinar e aprender os movimentos do braço robótico.

I.Programa de Ensino

Abra o arquivo “ [Memory Controlled Robot Arm.ino](#) ” em português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 6\Memory_Controlled_Robot_Arm, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa controladora correta, o processador e a porta, e faça o upload do código para a placa controladora ESP32.

Código de Exemplo:


```
#include <ACB_ARM.h> //Add the Robot arm libraries
ACB_ARM ARM;

void setup() {
  ARM.ARM_init(5,16,17,18); //The parameters are four Servo pins
  ARM.Joystick_init(32,33,34,35,36,39); //Joystick initialization
  Serial.begin(115200); //set the baud rate to 115200
}

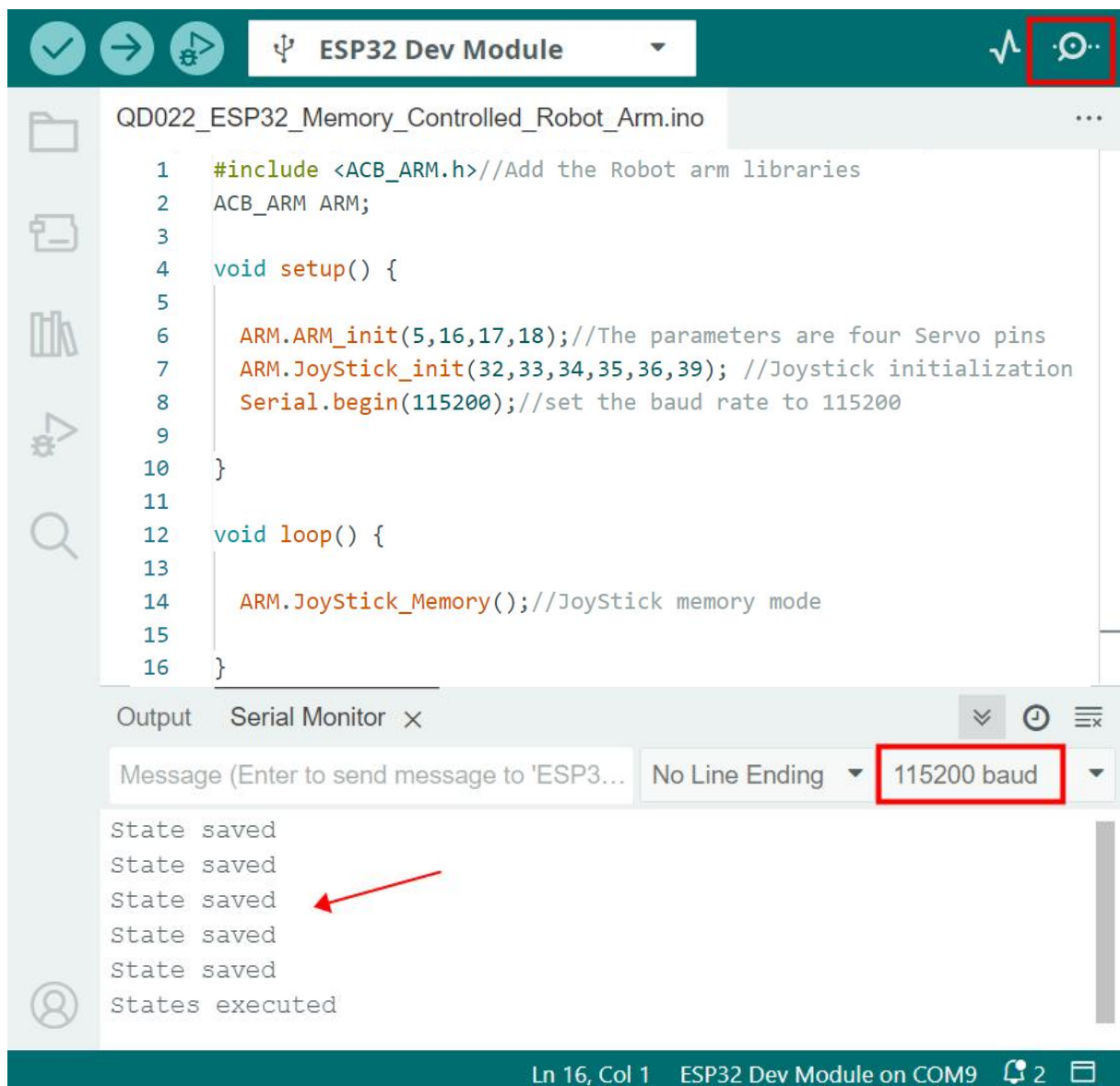
void loop() {
  ARM.Joystick_Memory(); //JoyStick memory mode
}
```

Atenção: ①Ao determinar o primeiro e o último ponto de posição do braço robótico, é necessário salvá-los prontamente;

②A função de memória pode salvar até 20 grupos de ações por vez.

Instrução de operação de ensino:

Joystick	Operação do Joystick	ações
Joystick Esquerdo	pressionamento curto	salvar ação
	pressionamento longo	Limpar ação
Joystick Direito	pressionamento curto	Executar ação



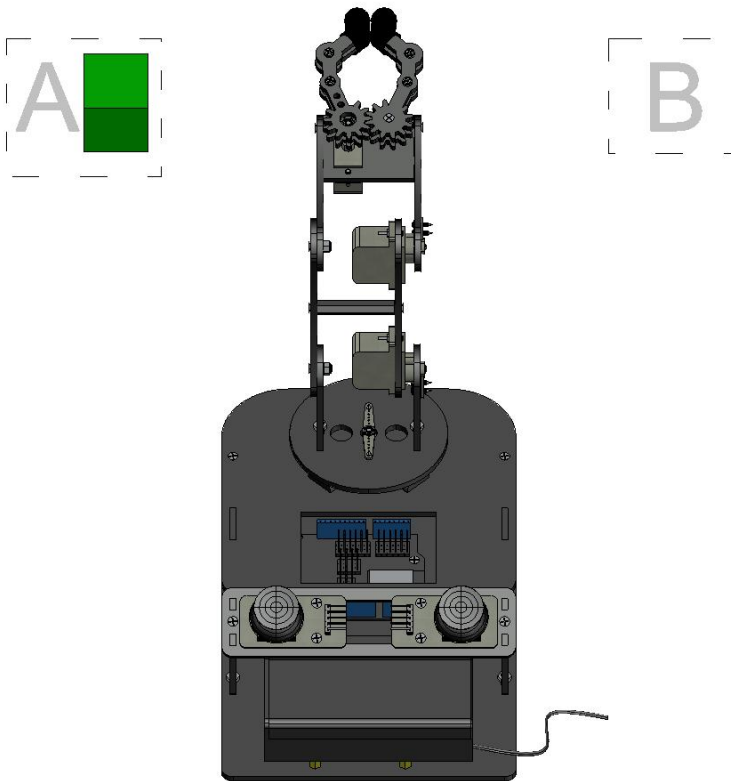
II. Tarefas de Expansão

Com base nas regras operacionais básicas de ensino e aprendizagem do braço robótico, a seguir, podemos usar o módulo joystick para realizar as funções de ensino e aprendizagem no transporte de objetos com o braço robótico.

Descrição da tarefa:

Use o módulo joystick para controlar o braço robótico, pegar um bloco do ponto A no mapa de coordenadas, transportá-lo até o ponto B e, em seguida, retornar à posição inicial. Esse processo envolve ensino e aprendizagem.

Atenção: Defina as posições dos pontos A e B como desejar.



Lição 7: Controle Web do Braço Robótico

Com o desenvolvimento contínuo da tecnologia de comunicação sem fio e da tecnologia da Internet das Coisas (IoT), a tecnologia de controle remoto está amplamente sendo usada em diversos campos. Ela permite que os usuários realizem um controle remoto preciso de dispositivos terminais à longa distância. Existem vários tipos de tecnologias de comunicação sem fio, e este tutorial foca principalmente em como usar a tecnologia de comunicação Wi-Fi para controlar remotamente um braço robótico.

A tecnologia de comunicação WiFi é um tipo de tecnologia de Rede Local Sem Fio (WLAN) que permite que dispositivos eletrônicos como smartphones, tablets e laptops se conectem sem fio à Internet ou a uma rede local (LAN). A tecnologia de comunicação WiFi conecta dispositivos à mesma rede utilizando roteadores sem fio ou pontos de acesso (APs), permitindo que eles transmitam e recebam dados entre si.

O controle de dispositivos baseado na web é uma das principais aplicações da tecnologia de comunicação WiFi, amplamente utilizada em áreas como casas inteligentes e indústrias inteligentes. O controle de dispositivos baseado na web conecta dispositivos e terminais de controle pela Internet. A interação entre dispositivos e controladores pode ser realizada através de simples protocolos HTTP. Quando um dispositivo se conecta a um controlador, o controlador fornece uma interface web simples que os usuários podem acessar por meio de uma página da web para controlar o dispositivo.

A seguir, usaremos uma página da web para controlar remotamente a operação do braço robótico.

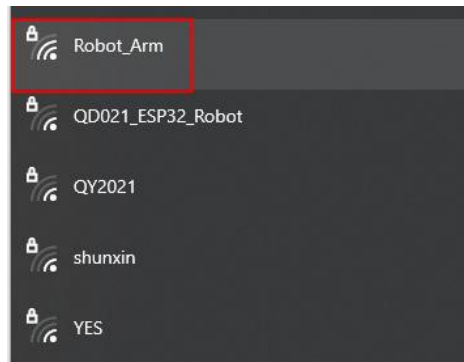
I. Programa de Controle Web

Abra o arquivo "[Web_Controlled_Robot_Arm.ino](#)" em português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 7\Web_Controlled_Robot_Arm, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa

controladora correta, o processador e a porta, e faça o upload do código para a placa controladora ESP32.

II.Login na Página Web

Após o upload bem-sucedido, em seguida, use seu computador ou celular para procurar redes Wi-Fi. Conecte-se ao ponto de acesso Wi-Fi chamado 'Robot_Arm' com a senha 12345678, conforme mostrado na imagem abaixo.



Após a conexão bem-sucedida, digite "192.168.4.1" na barra de endereços do seu navegador. A interface da página web aparecerá conforme mostrado na imagem abaixo.

Not secure 192.168.4.1

Robot Arm

Claw Open

Chassis left Chassis right

Claw Close

Shoulder Up

Elbow Down Elbow Up

Shoulder Down

Slide Control

Claws : 117 Enter Value

Elbow : 90 Enter Value

Shoulder: 90 Enter Value

Chassis : 90 Enter Value

Custom mode

MODE 1

Start Save Run Repeat Start Clear Action

Spatial coordinate

X: Enter Value Y: Enter Value Z: Enter Value Confirm

The value of x ranges from -19 to 19.
 The value of y ranges from 0 to 19.
 The value of z ranges from 0 to 27.
 Note: The value range is the point within the sphere.

No.	Funcionalidade da Página Web	Descrição da Função
1	Controle por botão	Controlar o movimento do braço robótico através dos botões na página web.
2	Controle deslizante	<p>Controle o movimento do braço robótico movendo os controles deslizantes ou inserindo um ângulo na caixa de entrada na página web.</p> <p>Atenção: Mova o controle deslizante lentamente; quanto mais rápido você movê-lo, mais rápido o braço robótico se moverá.</p>

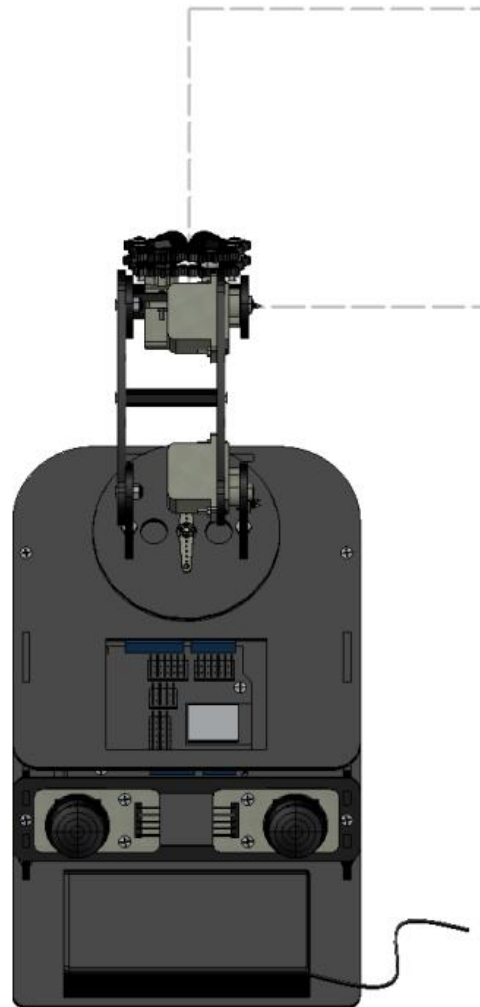
3	Salvar ação	<p>Você pode salvar um total de 6 conjuntos de ações para o braço robótico (Modo 1 a 6), com cada conjunto podendo armazenar até 20 ações diferentes. O processo específico de operação é o seguinte:</p> <p>①Clicando em "Iniciar" o botão muda para "Finalizar", depois siga para "Salvar" a ação. Com base no seu caminho de ação, clique em "Salvar" passo a passo. Lembre-se de clicar em "Salvar" tanto para a posição inicial quanto para a posição final.</p> <p>②Clique em "Finalizar" para concluir o salvamento da ação;</p> <p>③Clique em "Executar" para realizar uma ação salva na memória;</p> <p>④Clique em "Repetir Início" para repetir a ação gravada;</p> <p>⑤Clique em "Resetar" para redefinir o grupo de ações.</p>
4	Posicionamento espacial	<p>Digite as coordenadas espaciais x, y, z, e depois clique em "Confirmar". O braço robótico se moverá para as coordenadas espaciais especificadas.</p> <p>Atenção: Abaixo dos campos de entrada x, y, z, há descrições correspondentes ao intervalo de valores permitido. Se os valores inseridos estiverem fora do intervalo especificado, por favor, insira-os novamente.</p>

III.Tarefas de Expansão

De acordo com o controle do braço robótico pela página web, a seguir usaremos três métodos de controle para realizar a função de salvamento de ações para o braço robótico na página web.

Descrição da tarefa:

- (1)Use os botões para controlar as garras do braço robótico e desenhar ações semelhantes a um quadrado no mapa, salvando-as no modo1;
- (2)Use o slider para controlar as garras do braço robótico e desenhar uma ação semelhante a um quadrado no mapa, salvando-a no modo2;
- (3)O posicionamento espacial é usado para controlar a garra do braço robótico e desenhar uma ação semelhante a um quadrado no mapa, salvando-a no modo3.



Lição 8: Controle do Braço Robótico pelo APP

No tutorial anterior, aprendemos a controlar o braço robótico com um joystick e uma página web. Para controlar o braço robótico de maneira mais conveniente, escolhemos usar o APP móvel como interface de controle, permitindo que o usuário controle o braço robótico através do aplicativo. A seguir, veremos como controlar o funcionamento do braço robótico por meio de um APP móvel.

I.Download do APP

(1)Se você estiver usando um dispositivo iOS, pesquise pela palavra-chave "ACEBOTT" na App Store e faça o download. Se estiver usando um dispositivo Android, pesquise pela palavra-chave "ACEBOTT" na Google Play Store e faça o download. O ícone é mostrado abaixo.



Atenção:

- ①Este tutorial é aplicável à versão 2.0 ou superior do APP ACEBOTT. Você pode clicar no botão de configurações no canto superior esquerdo do APP para visualizar o número da versão do software. Certifique-se de que a versão do software que você está usando atende aos requisitos;
- ②Se precisar atualizar a versão do software ACEBOTT, você pode seguir o método indicado neste tutorial para baixar a versão mais recente do APP.

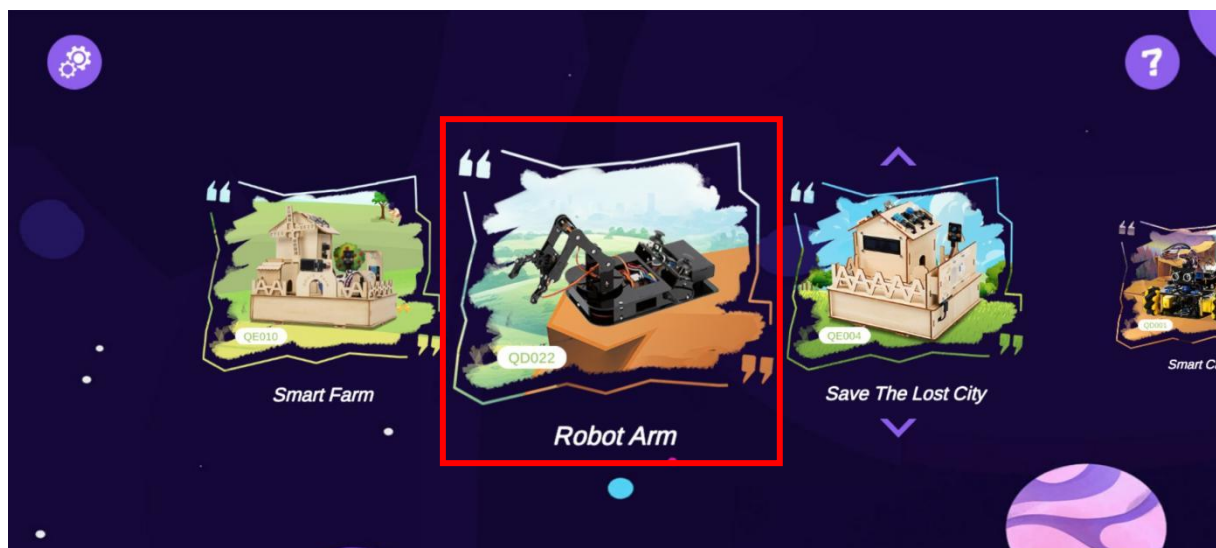
(2)Abra o app para acessar a tela de introdução.



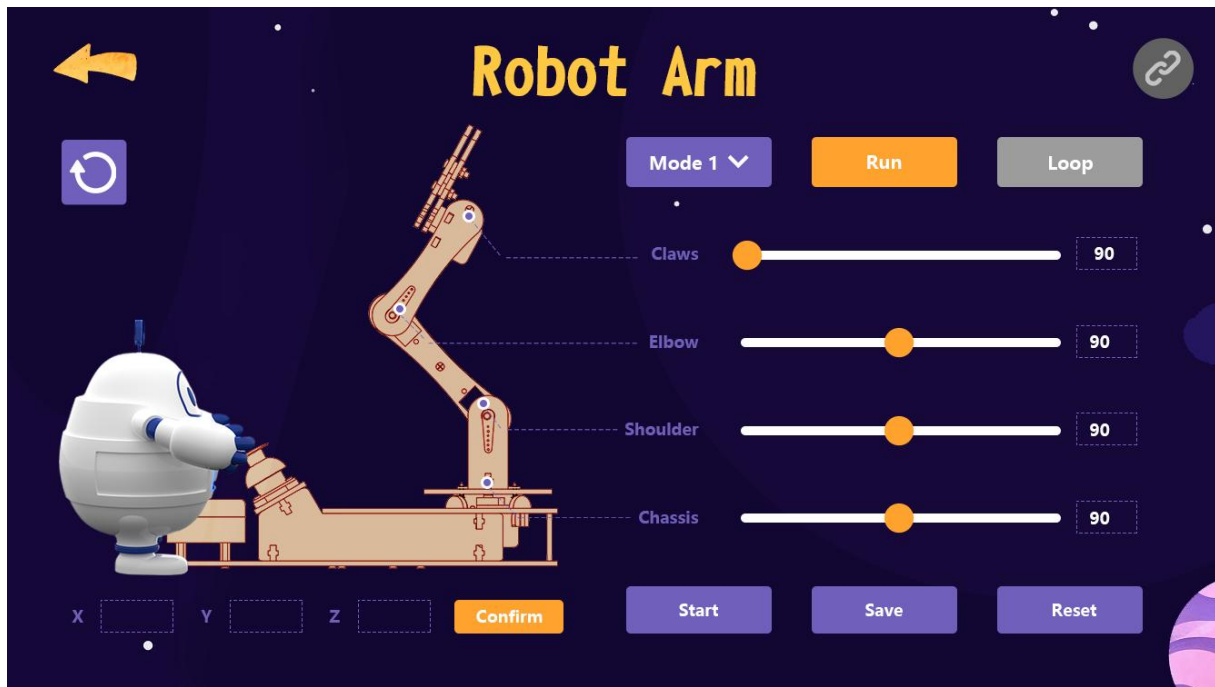
(3) Acesse a tela de seleção e escolha o braço robótico.

Atenção: Se você precisar assistir ao vídeo de operação do APP, clique no link abaixo.

<https://youtu.be/0JtV29RbKQs>



(4) Acesse a interface de controle do braço robótico (ainda não é possível controlá-lo diretamente, pois o programa precisa ser carregado primeiro).



II. Controle do Braço Robótico pelo APP

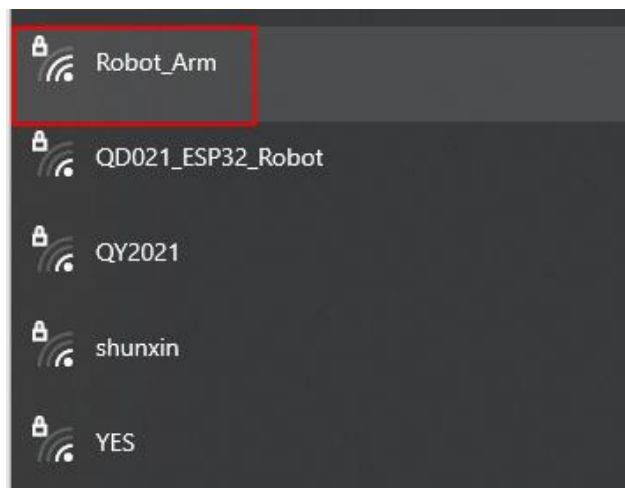
1. Carregue o programa Arduino para controle do braço robótico pelo APP.

Antes de usar o APP para controlar o braço robótico, é necessário carregar o programa Arduino que permite a comunicação entre o braço robótico e o APP.

Abra o arquivo ["APP Controlled Robot Arm.ino"](#) em português\Arduino (Aprendiz Experiente)\2.Programa Arduino\Lição 8\APP_Controlled_Robot_Arm, conecte a placa controladora ESP32 ao computador com um cabo USB, selecione a placa de controle correta, o processador e a porta, e faça o upload do código para a placa controladora ESP32.

2. Conectar à WiFi do Braço Robótico

Procure redes WiFi no seu computador ou celular e conecte-se ao ponto de acesso WiFi chamado 'Robot_Arm' com a senha 12345678, conforme mostrado na imagem abaixo.

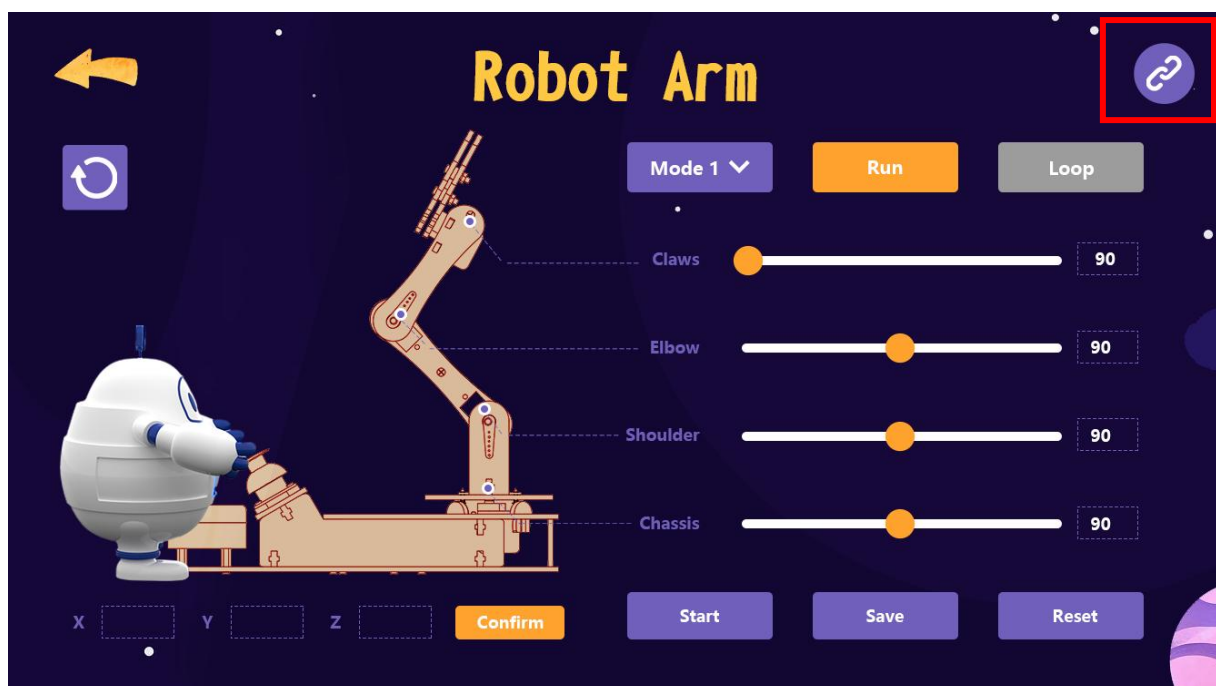


Atenção: O nome do ponto de acesso e a senha são pré-definidos no programa, mas os usuários podem personalizá-los. Quando tivermos múltiplos braços robóticos, podemos distinguir cada um utilizando nomes de WiFi diferentes.

```
const char* ssid = "Robot_Arm";// Wifi name
const char* password = "12345678";// WiFi password
```

3. Usando os controles do APP

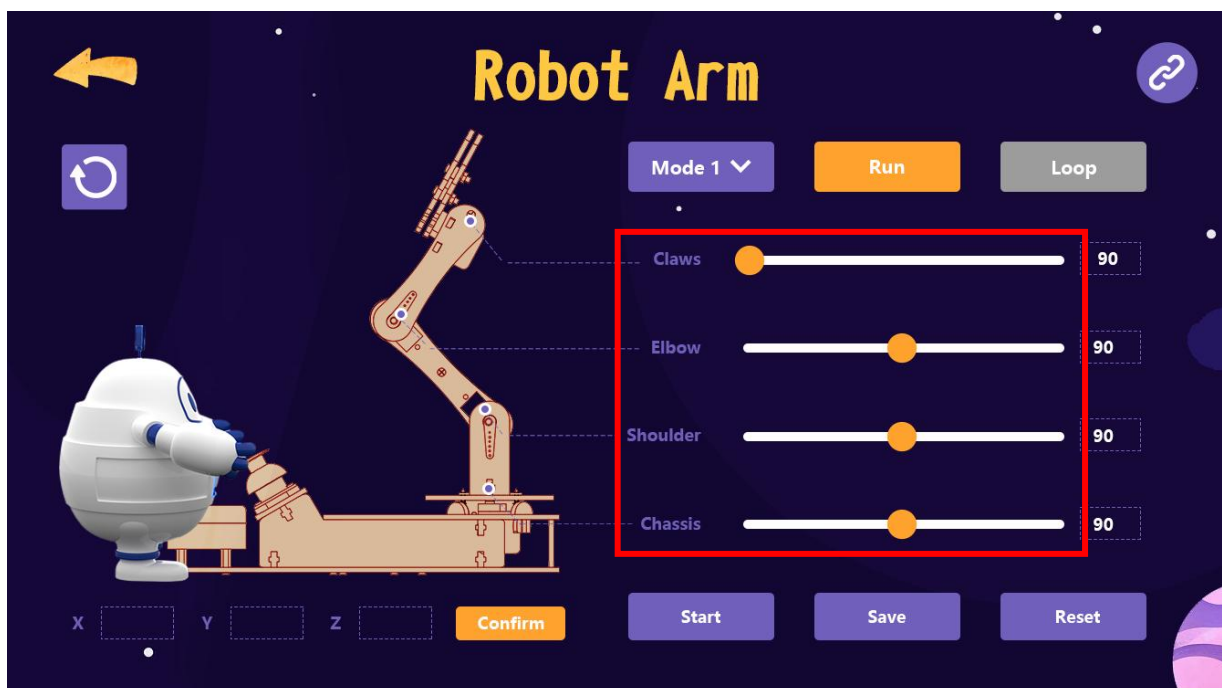
Após conectar à WiFi, clique no ícone de conexão no canto superior direito do APP para completar a conexão.



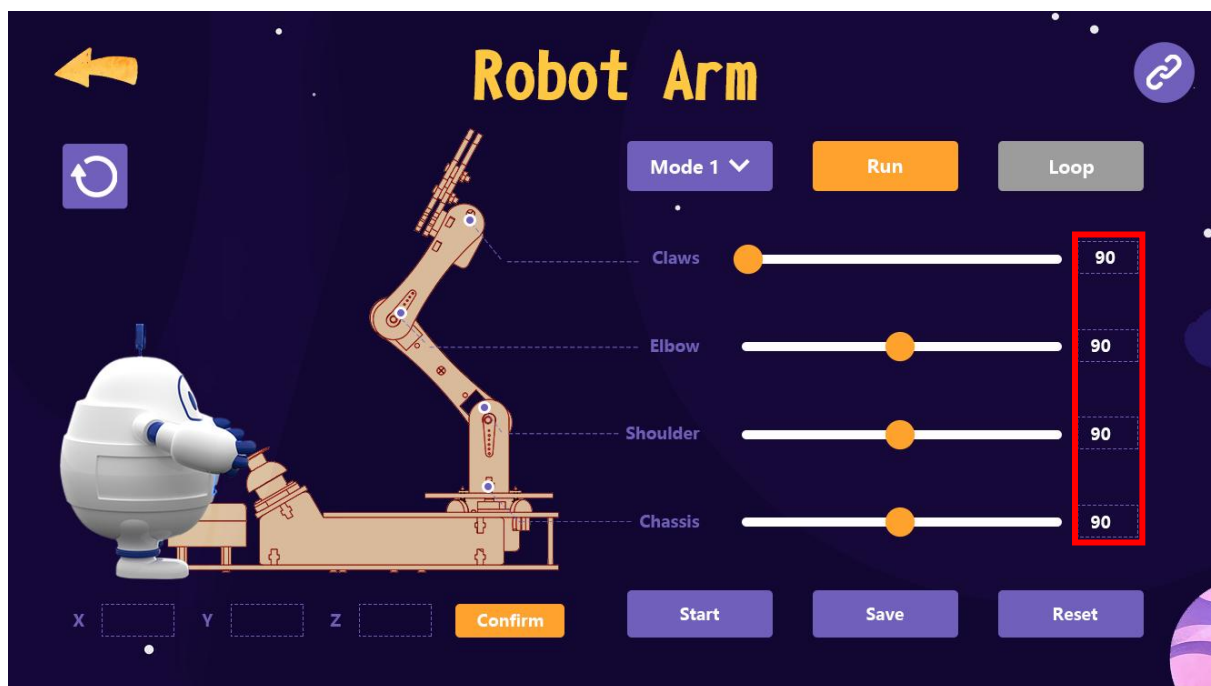
Após concluir as operações acima, retorne à interface mostrada abaixo novamente, e então o controle do braço robótico pode ser realizado. As principais ações de controle são: controle deslizante, controle por caixa de entrada, modo personalizado (iniciar, finalizar, salvar, executar, resetar), função de posicionamento espacial e função de recuperação de posição.

Introdução à função do APP do braço robótico:

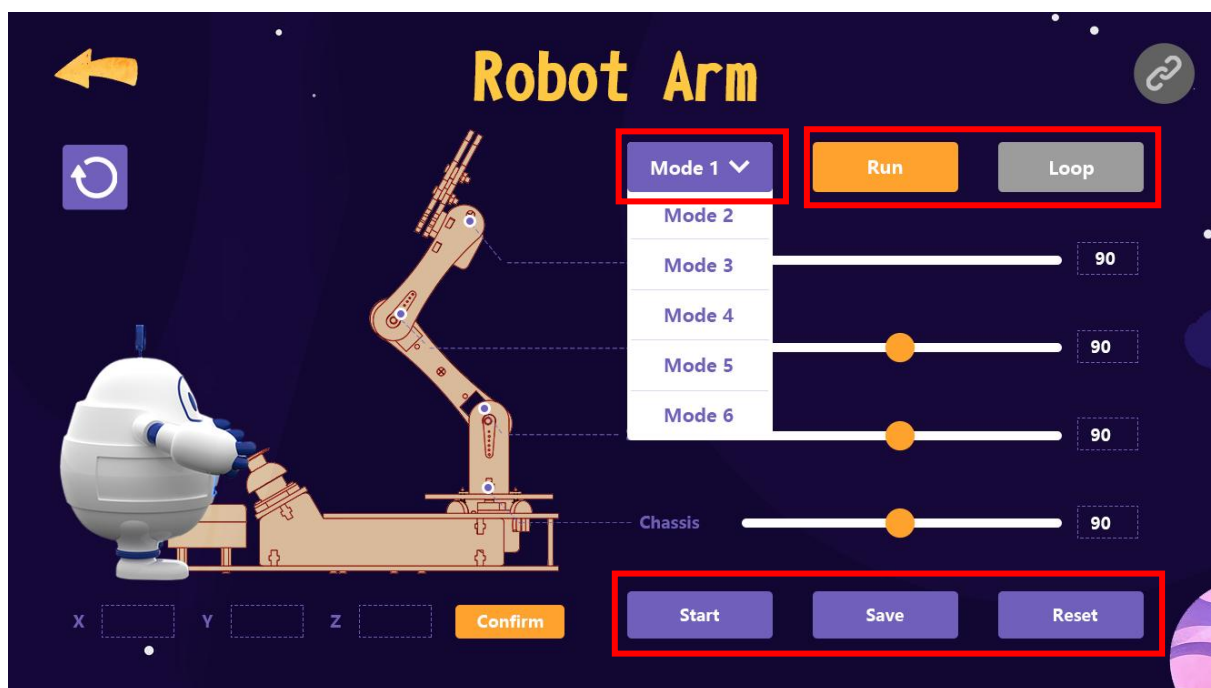
(1) **Controle deslizante:** Mova os controles deslizantes correspondentes aos diferentes servos do braço robótico para ajustar sua postura e orientação.



(2) **Controle por caixa de entrada:** Ao lado de cada controle deslizante, há uma caixa de entrada onde você pode inserir o ângulo desejado do servo para controlar a postura e orientação do braço robótico.

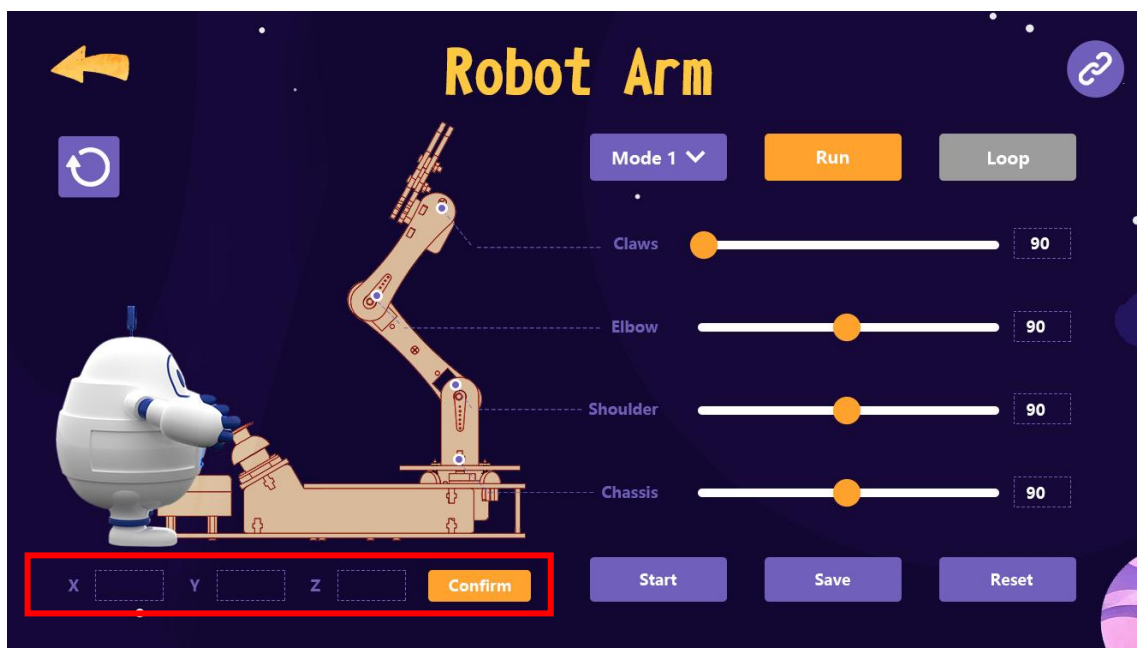


(3)**Salvar ação:** Clique em 'mode1', haverá 6 modos para escolher (Modo 1~6), e até 20 grupos de ações diferentes podem ser salvos em cada modo. Clique em 'Run' para executar uma ação memorizada, clique em 'Loop' para repetir a ação memorizada e clique em 'Reset' para apagar a ação memorizada. O processo de operação específico é o mesmo que o controle pela página web.

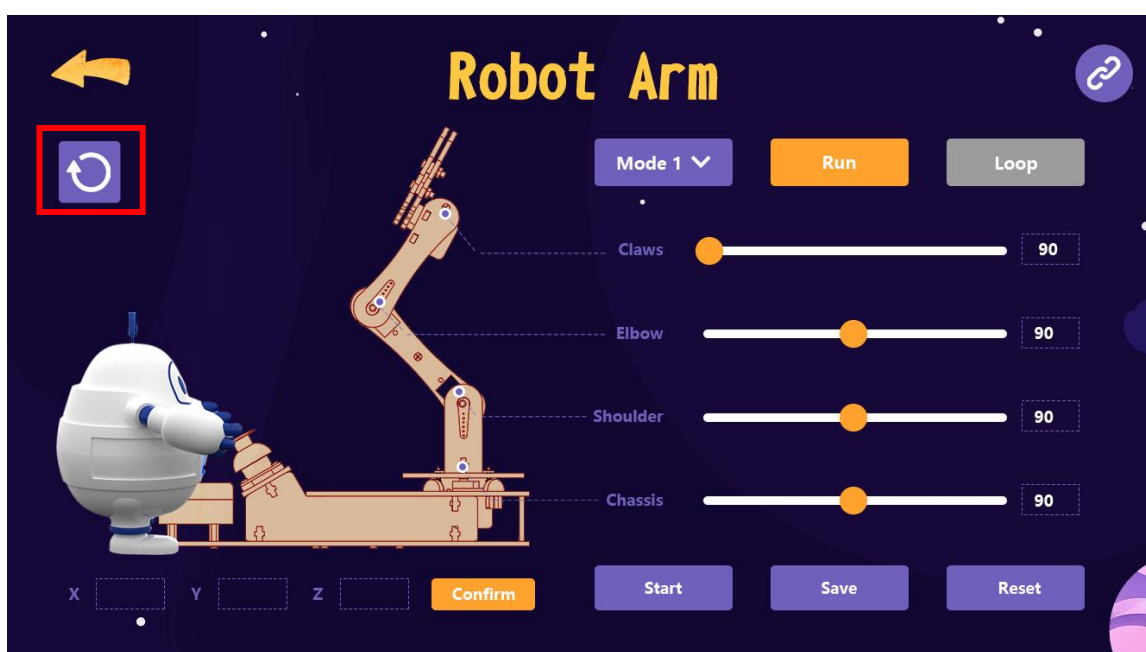


(4)**Posicionamento espacial:** Insira as coordenadas espaciais x, y, z, e o braço robótico se moverá até o ponto de coordenadas espaciais especificado.

Atenção: Devido à estrutura do braço robótico, existem limitações em seu alcance de movimento. Ao inserir as coordenadas, se os valores excederem o alcance de movimento do braço robótico, você pode inseri-los novamente.



(5)**Inicialização de posição:** Clique no ícone de atualização no canto superior esquerdo, e o braço robótico retornará à sua posição inicial.



Siga-nos

Escaneie os códigos QR para nos seguir e obter suporte para solução de problemas e as últimas notícias. Temos uma comunidade muito grande, que é muito útil para solucionar problemas, e também contamos com uma equipe de suporte pronta para responder a qualquer dúvida.



Código QR do Grupo ACEBOTT



Código QR do YouTube