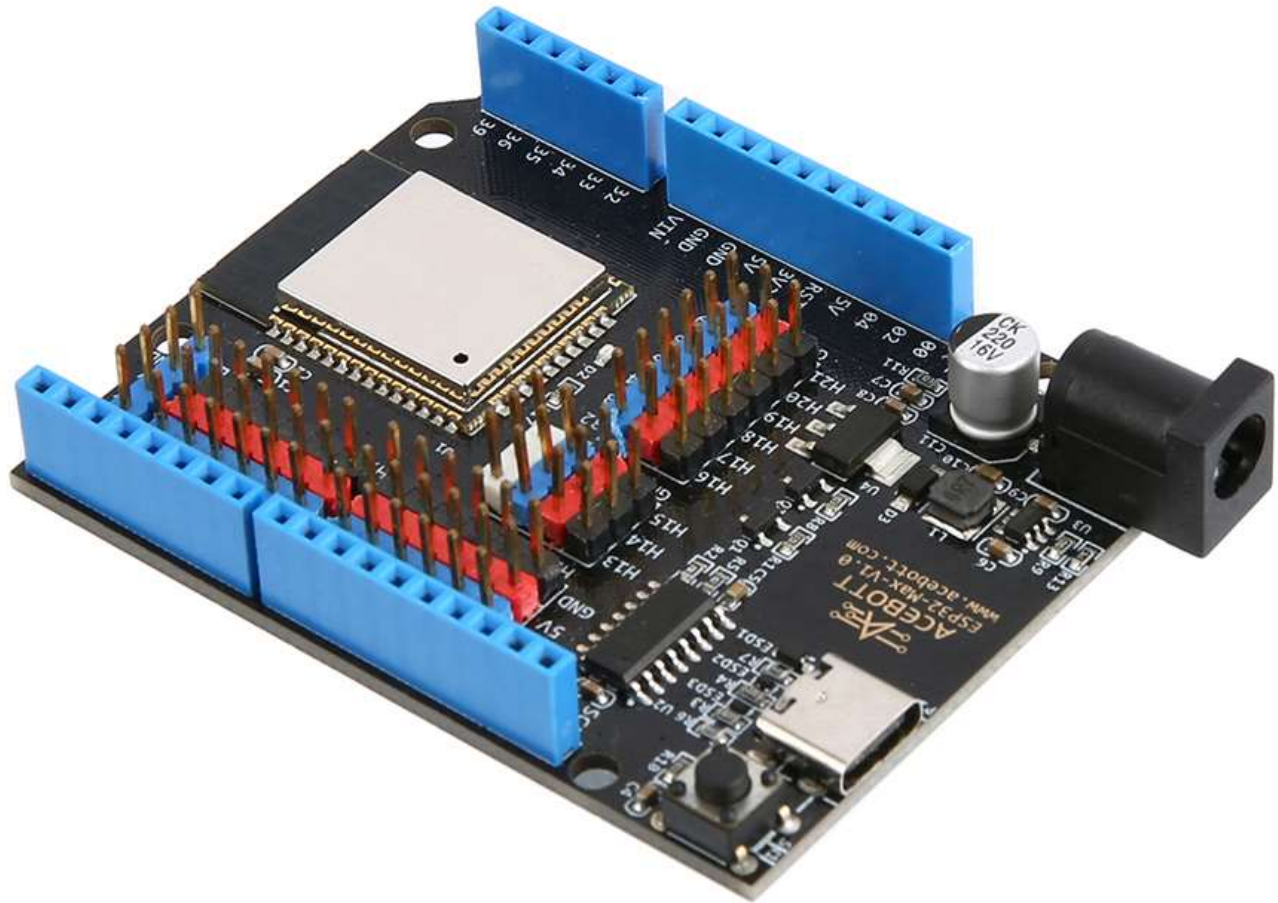


Placa controladora ESP007 Max V008 QA009/QA32/QA1.0



1.Introduction

El Roadshow de **ESP32** La placa controladora Max 1.0 es un microcontrolador de alto rendimiento y bajo consumo de energía, muy adecuado para el desarrollo de la Internet de las cosas.

Cuenta con un procesador dual-core de 240 MHz, 520 KB de RAM y 4 MB de memoria flash. Módulo WiFi y Bluetooth 4.2 integrados, disponibles para comunicación inalámbrica. Con 34 pines GPIO, puede conectar y controlar varios periféricos.

Si quieres saber más sobre ESP32, haga clic [aquí](#)

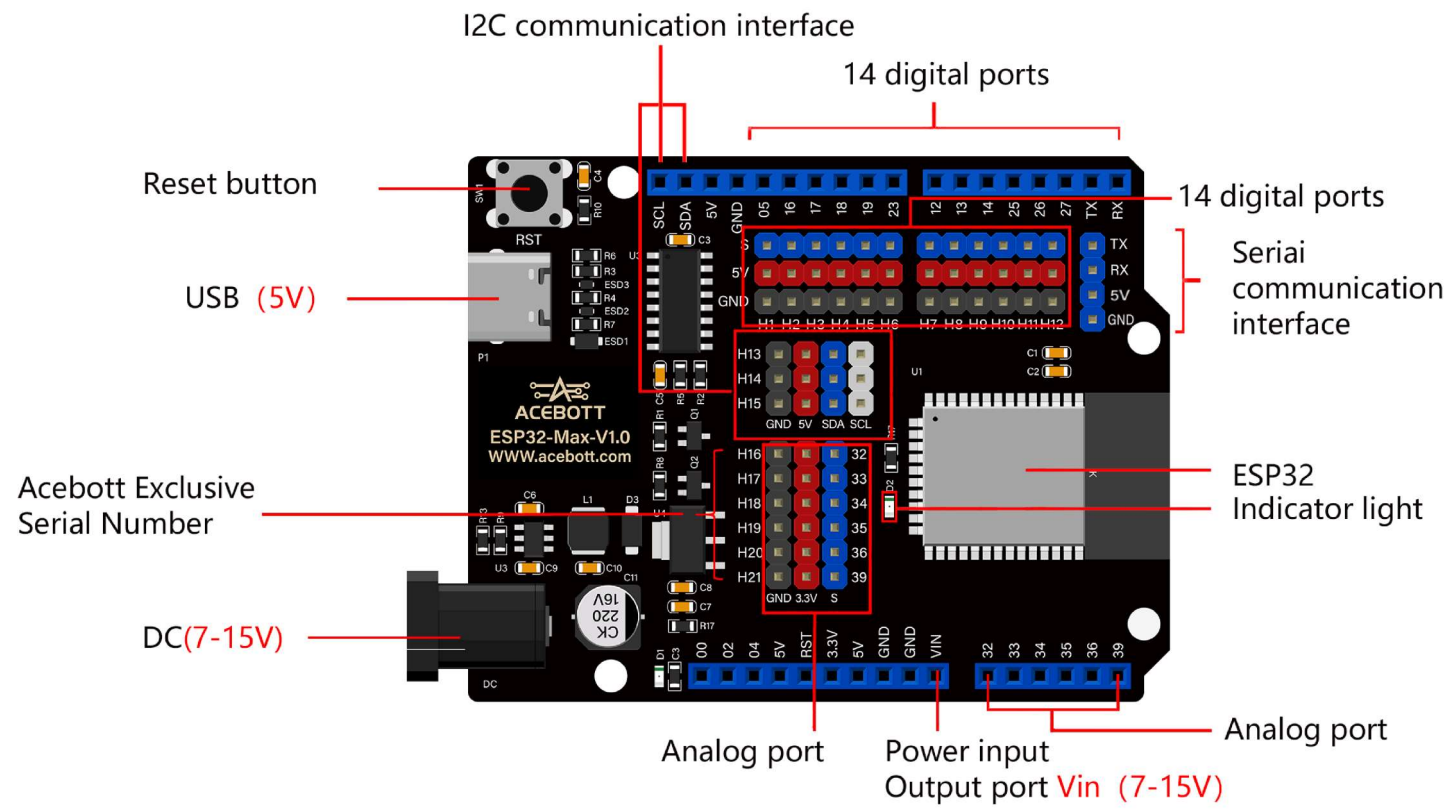
2.Features

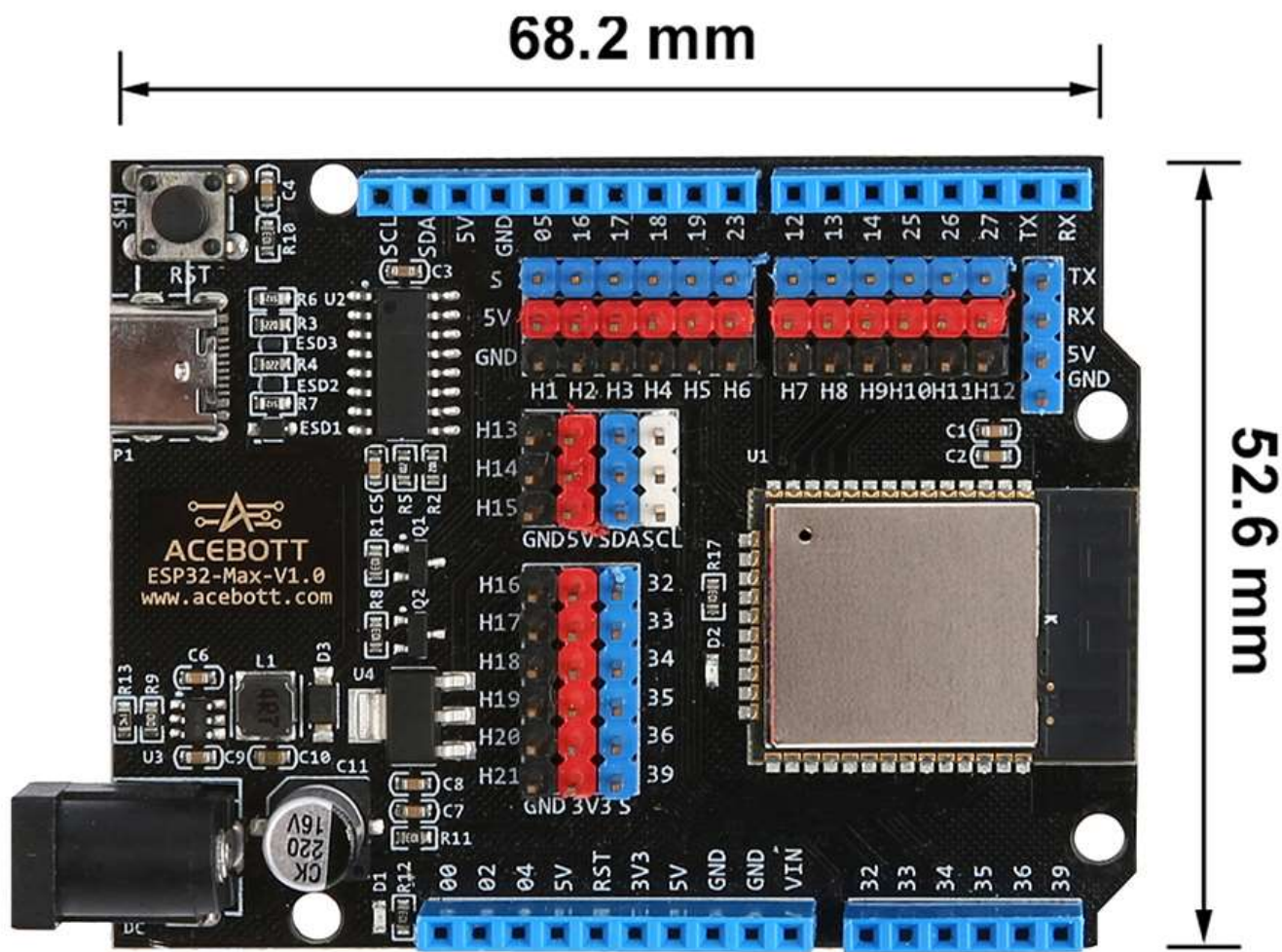
- 1. Actualice la interfaz tipo C para una mayor compatibilidad
- 2. Todos los pines IO se exportan para facilitar el desarrollo.
- 3. No se necesita una placa de pruebas adicional. Tiene pines macho y hembra, y los pines están claramente coloreados para facilitar el cableado.
- 4. La interfaz Tipo C está equipada con un diodo de protección contra descarga electrostática y un diodo de supresión de voltaje transitorio para proteger el chip contra averías electrostáticas y daños de varios pulsos de sobretensión.

3.Specifications

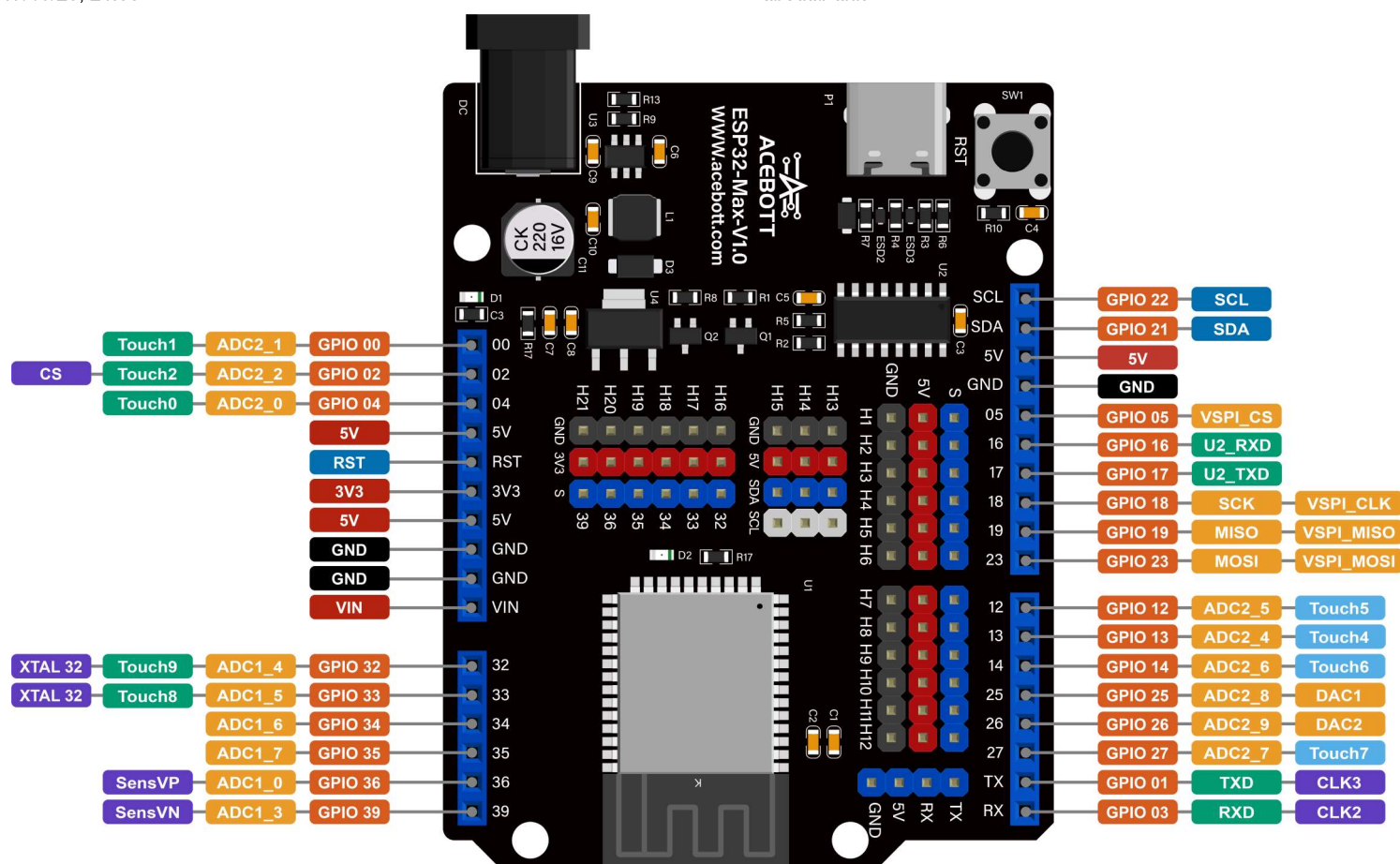
Conectividad:	Wi-Fi Bluetooth LE
Viruta	ESP-WROOM-32
Reloj	240MHz
ROM	448KB
SRAM	520KB
FLASH	4MB
Interfaces	UART I2C SPI PUEDE

Conectividad:	Wi-Fi Bluetooth LE
Voltajes de entrada	6-18V
Asignación de pines	25 (DIGITAL) 15 (ANALÓGICO) 25 (PWM) 2 (DAC) 2 (UART) 2 (SPI) 1 (I2C)





4. Descripción de PCB



La figura ilustra los pines GPIO y las funciones correspondientes de la placa controladora ESP32 Max 1.0, incluidos los pines ADC, DAC, PWM, I2C, SPI, etc.

Solo pin de entrada: GPIO34/GPIO35/GPIO36/GPIO39

Pin I2C predeterminado: GPIO 21 (SDA)/GPIO 22 (SCL)

Pin DAC: Entrada y salida de datos GPIO 25 (DAC1)/entrada y salida de datos GPIO 26 (DAC2)

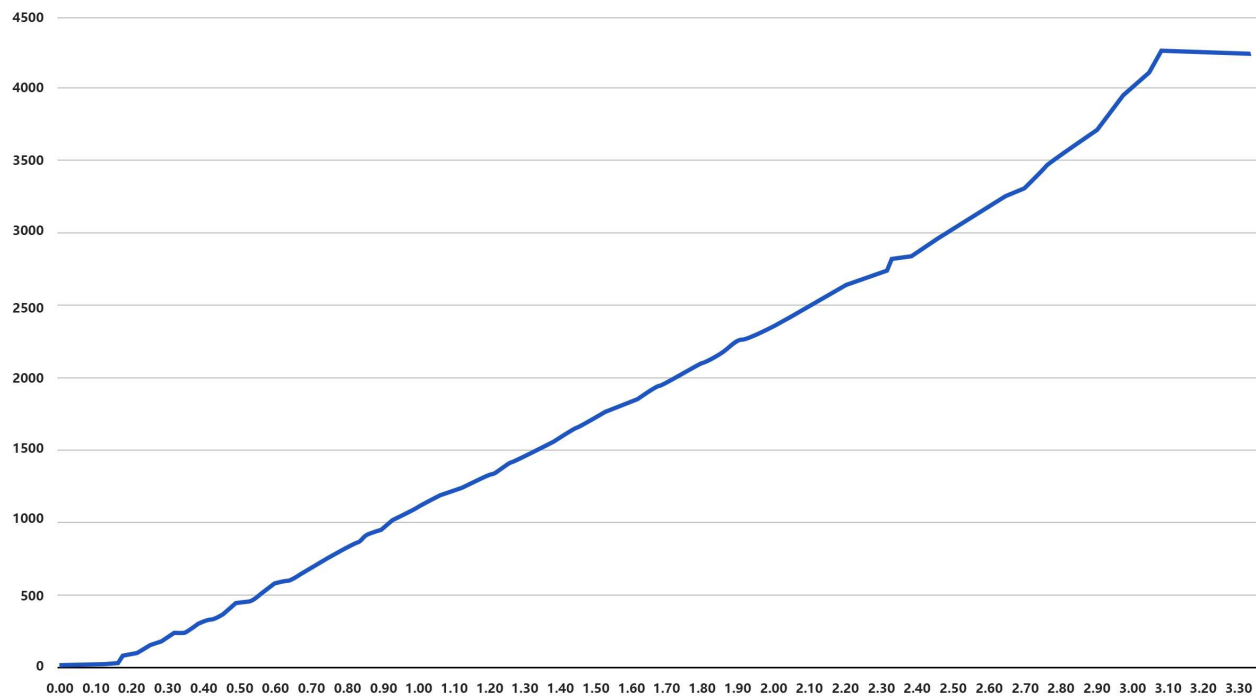
5.Nota

(1) El pin ADC2 no se puede utilizar cuando se utiliza Wi-Fi. Por lo tanto, si no puede obtener el valor del GPIO ADC2 en Wi-Fi, considere utilizar el GPIO ADC1 en su lugar.

(2) El canal de entrada del ADC tiene una resolución de 12 bits. Esto significa que puede obtener lecturas analógicas entre 0 y 4095, donde el valor analógico 0 corresponde a 0 V y el valor analógico 4095 corresponde a 3.3 V. También puede configurar la resolución del canal, así como el rango del ADC en el código.

(3) Los pines del ADC ESP32 no tienen un comportamiento lineal. Es posible que no pueda distinguir entre 0 y 0.1 V, 3.2 y 3.3 V. Tenga esto en cuenta al utilizar los pines del ADC y obtendrá una imagen de datos similar a la que se

muestra a continuación.



6. Uso detallado con el software ARDUINO como se indica a continuación:

Paso 1 | Descargar el entorno Arduino (IDE)

[haga clic aquí](#)

Paso 2 | Instalar el complemento ESP32 en el IDE de Arduino

[haga clic aquí](#)

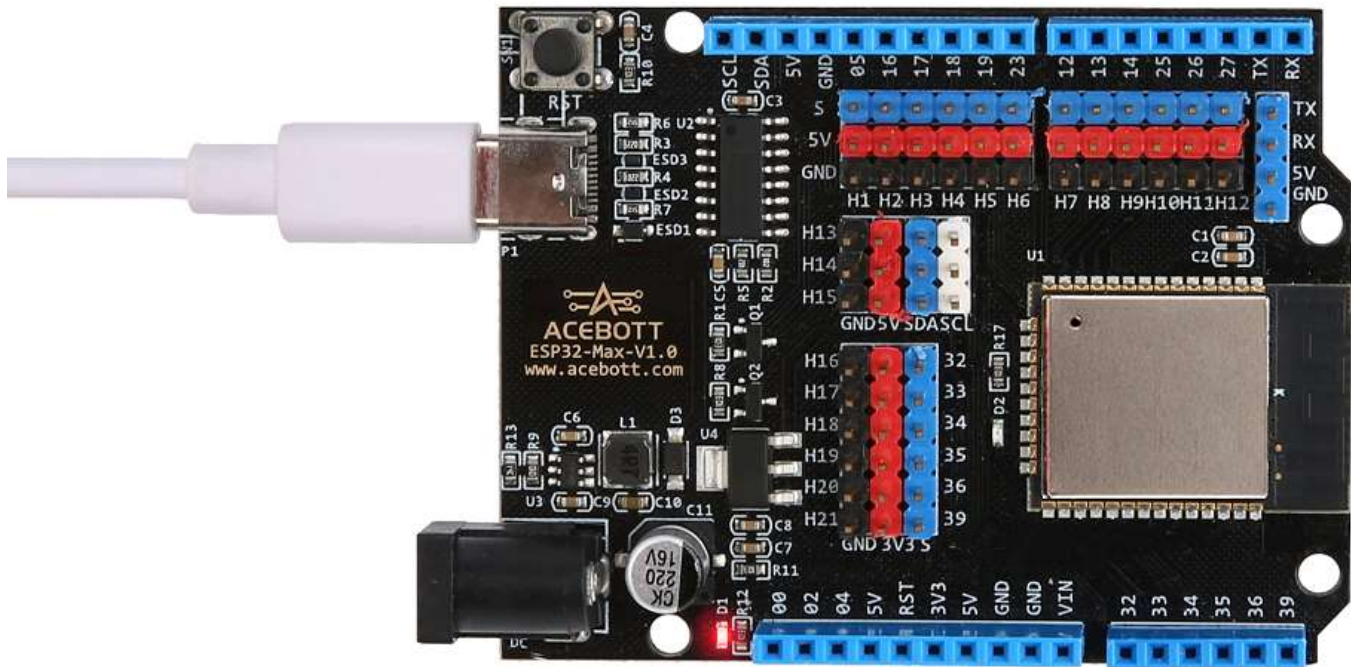
Paso 3 | Descargar el controlador CH340

Sistema MAC

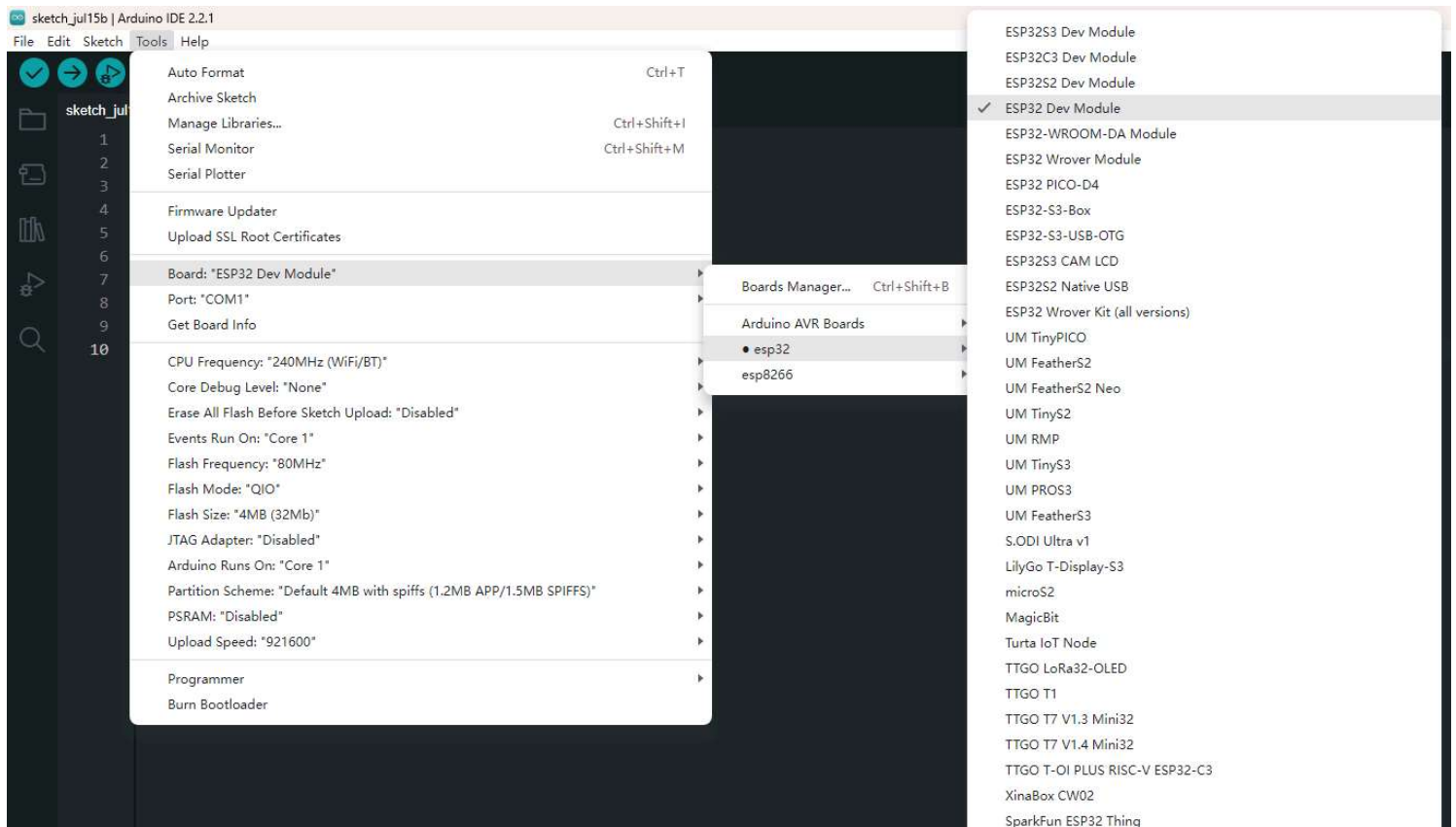
Sistema de Windows

Paso 4 | Subir el programa de iluminación LED

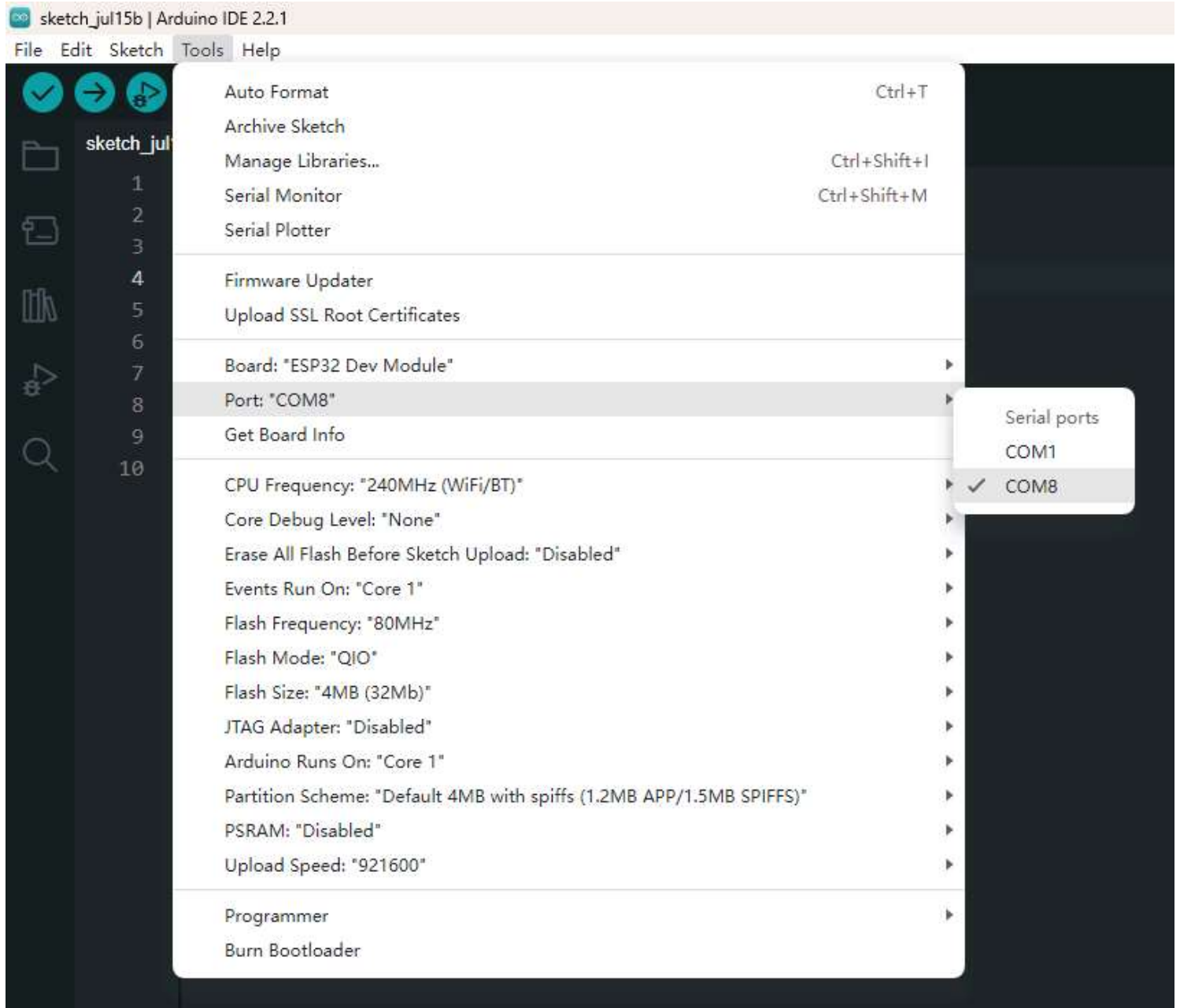
Conecte la placa a la computadora usando un cable de datos tipo C. El LED de encendido rojo debería encenderse.



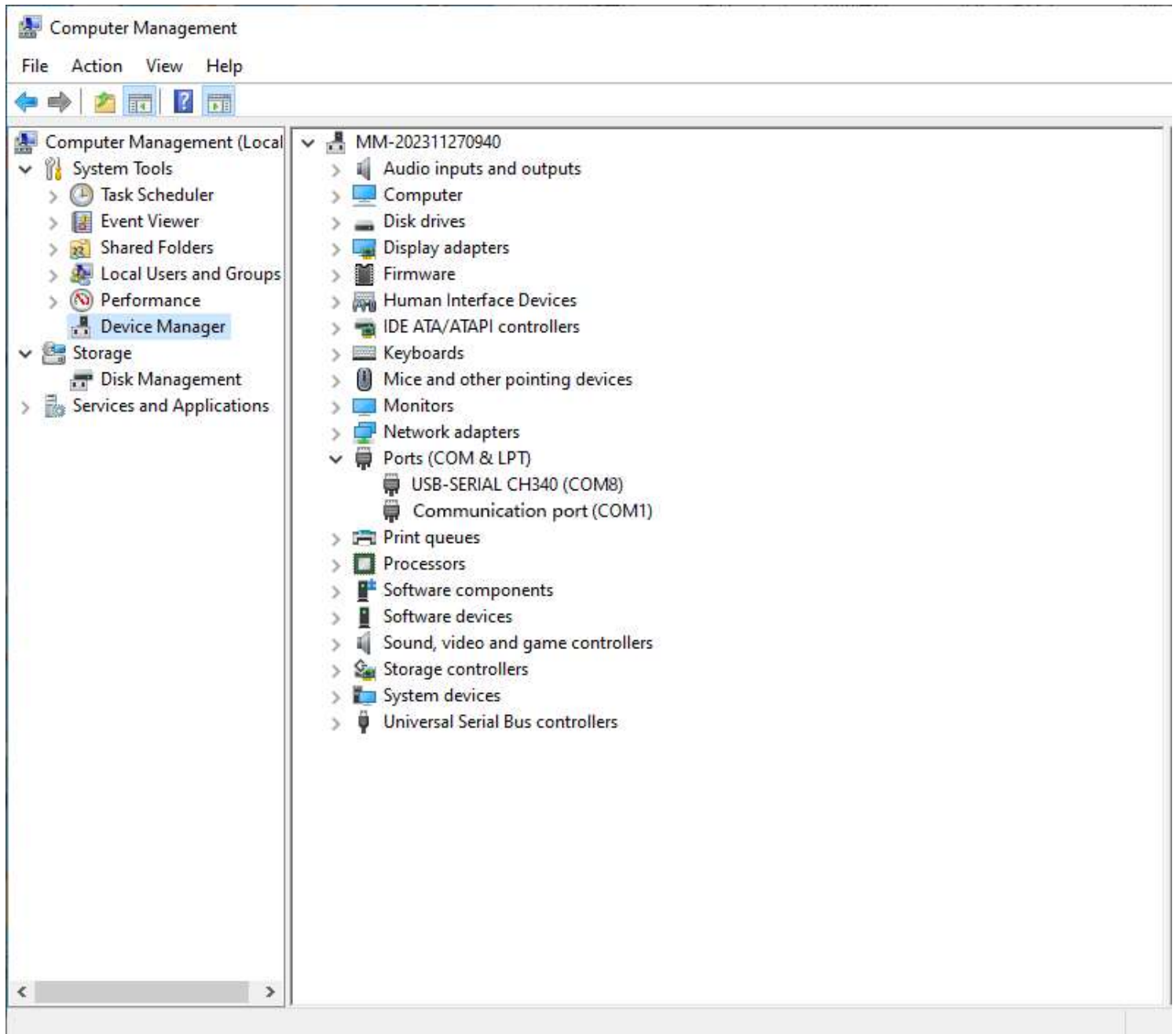
Seleccione “ESP32” -> “Módulo de desarrollo ESP32” en el menú Herramientas > Placa



Seleccione el dispositivo serial de la placa en el menú Herramientas | Puerto serial. Es probable que sea COM8 o superior (COM1 y COM2 suelen estar reservados para los puertos seriales de hardware). Para averiguarlo, puede desconectar la placa y volver a abrir el menú; la entrada que desaparece debería ser la placa. Vuelva a conectar la placa y seleccione ese puerto serial. Aquí debe seleccionar COM 8 como se muestra a continuación.



También puede encontrar los puertos correctos que se muestran en el Administrador de dispositivos.



Subir el programa.

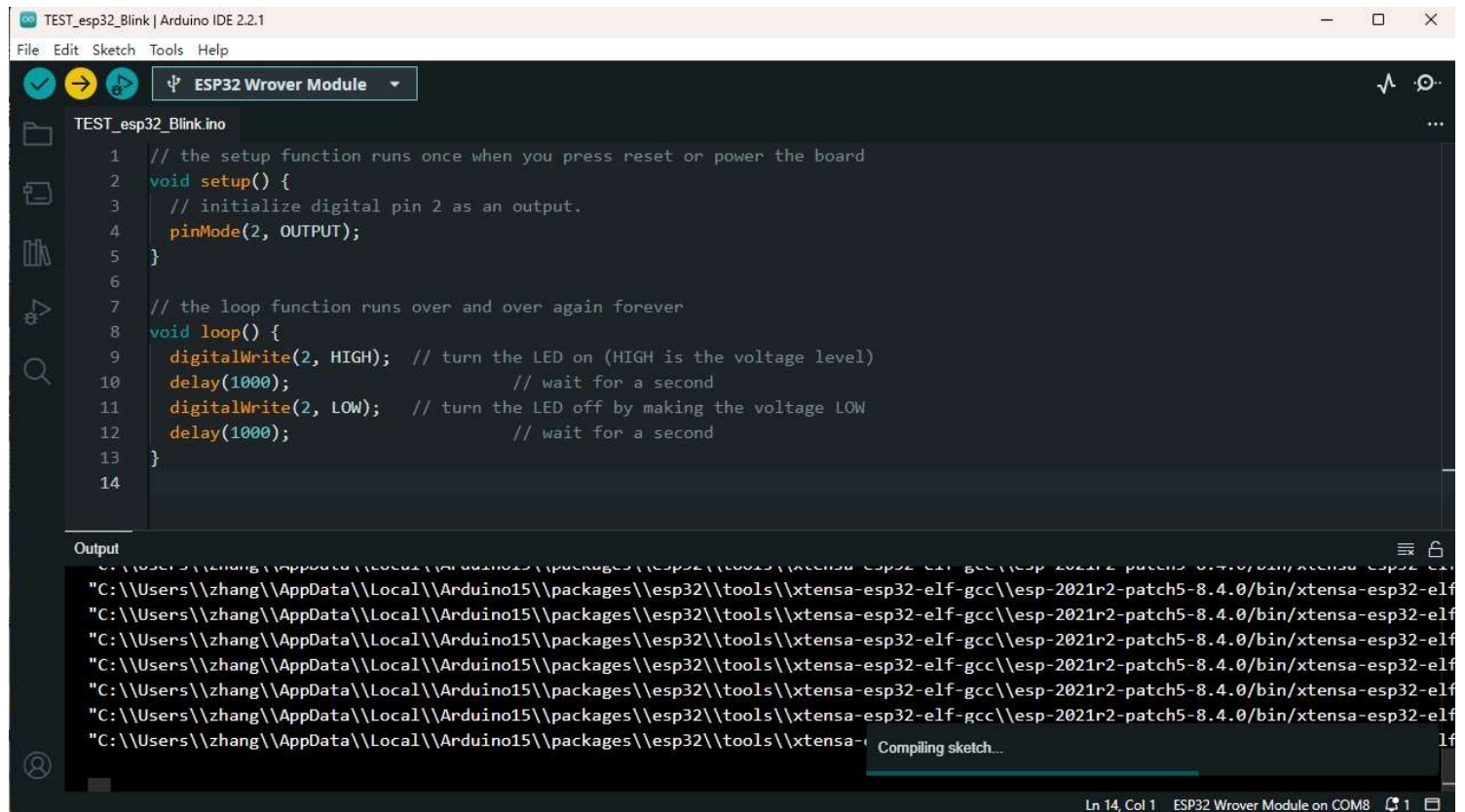
```

1// the setup function runs once when you press reset or power the board
2void setup() {
3  // initialize digital pin 2 as an output.
4  pinMode(2, OUTPUT);
5}
6
7// the loop function runs over and over again forever
8void loop() {
9  digitalWrite(2, HIGH); // turn the LED on (HIGH is the voltage level)
10 delay(1000);           // wait for a second
11 digitalWrite(2, LOW);  // turn the LED off by making the voltage LOW

```

```
12 delay(1000);           // wait for a second
13}
```

Haga clic en el botón “Cargar” para cargar el código. aparecerá en la barra de estado.



Espere unos segundos. Si la carga se realizó correctamente, aparecerá el mensaje “Carga finalizada”.

The screenshot shows the Arduino IDE 2.2.1 interface. The top menu bar includes File, Edit, Sketch, Tools, and Help. The toolbar shows icons for opening a file, saving, and uploading. The 'Tools' dropdown menu is open, showing 'ESP32 Wrover Module' selected. The main editor displays the sketch 'TEST_esp32_Blink.ino' with the following code:

```
1 // the setup function runs once when you press reset or power the board
2 void setup() {
3   // initialize digital pin 2 as an output.
4   pinMode(2, OUTPUT);
5 }
6
7 // the loop function runs over and over again forever
8 void loop() {
9   digitalWrite(2, HIGH); // turn the LED on (HIGH is the voltage level)
10  delay(1000);           // wait for a second
11  digitalWrite(2, LOW);  // turn the LED off by making the voltage LOW
12  delay(1000);           // wait for a second
13 }
14
```

The 'Output' window at the bottom shows the upload progress and status:

```
Writing at 0x00000000... (0 %)
Writing at 0x00046191... (88 %)
Writing at 0x0004bb64... (100 %)
Wrote 253504 bytes (137604 compressed) at 0x00010000 in 2.3 seconds (effective 894.2 kbit/s)...
Hash of data verified.

Leaving...
Hard resetting via RTS pin...
```

A notification box at the bottom right of the output window says 'Done uploading.' The status bar at the bottom indicates 'Ln 14, Col 1' and 'ESP32 Wrover Module on COM8'.

Y el LED de la placa parpadea.

Si las pruebas fueron exitosas, puedes intentar cargar el siguiente código. Cuando se encuentre una red en el entorno, se obtendrá el número y el nombre de todas las redes cercanas y se mostrarán en el puerto serial, y el indicador se encenderá.

```
1/*  
2When a network is found in the environment,  
3the number and name of all nearby networks will be obtained and displayed in the serial port,  
4and the blue indicator will light up.  
5*/  
6#include "WiFi.h"  
7void setup()  
8{  
9    Serial.begin(115200);  
10   pinMode(02,OUTPUT);  
11   //set WiFi to station mode and disconnect from an AP if it was previously connected  
12   WiFi.mode(WIFI_STA);  
13   WiFi.disconnect();
```



```
14   delay(100);
15   Serial.println("Setup done");
16}
17void loop()
18{
19   Serial.println("scan start");
20   // WiFi.scanNetworks will return the number of networks found
21   int n = WiFi.scanNetworks();
22   Serial.println("scan done");
23   if (n == 0) {
24       Serial.println("no networks found");
25   } else {
26       Serial.print(n);
27       Serial.println(" networks found");
28       digitalWrite(2, HIGH); //the blue indicator lights up
29       for (int i = 0; i < n; ++i) {
30           //print SSID and RSSI for each network found
31           Serial.print(i + 1);
32           Serial.print(": ");
33           Serial.print(WiFi.SSID(i));
34           Serial.println((WiFi.encryptionType(i) == WIFI_AUTH_OPEN)? " ":"*");
35           delay(10);
36       }
37   }
38   Serial.println("");
39   // wait a bit before scanning again
40   delay(5000);
41}
```

El diagrama del efecto de visualización del puerto serie es el siguiente:

```
scan start
scan done
20 networks found
1: ACEBOTT*
2: HxSmart*
3: HxSmart*
4: QY2021*
5: JC-WIFI*
6: dxs*
7: ChinaNet-rwbn*
8: ChinaNet-9VQg*
9: HTX_2.4G*
10: ChinaNet-QM4V*
```

7. Lista de paquetes

Placa controladora ESP32 Max V1.0 * 1 pieza

Cable tipo C * 1 pieza

8. Consigue uno ahora

Negocio B2B: información@cebott.com

Website

Tienda Amazon EE. UU.