

ESP32-S3-Relay-1CH

From Waveshare Wiki

Jump to: navigation, search

Introduction

[Expand]

ESP32-S3-Relay-1CH



(<https://www.waveshare.com/esp32-s3-relay-1ch.htm>)

Electrical and Relay Safety Instructions

- This product must be operated by professional electricians or qualified personnel. During use, ensure electrical safety, leakage protection, and proper insulation.
- Before installing, maintaining, or replacing the relay device, always turn off the power and unplug the device.
- Do not attempt to disassemble the relay device to avoid damage or the risk of electric shock.
- Properly install and place the relay device. Do not use it in humid, overheated, flammable, or explosive environments to prevent accidents caused by improper installation or use.

1. Load Matching

- Ensure the relay's rated voltage and current match the load. Do not exceed the rated capacity.
- For inductive loads (motors, coils, lamps, etc.), the starting current may be much higher than the rated current. Choose a relay with sufficient current margin.

2. Short Circuit and Overcurrent Protection

- Install a **fuse** or **circuit breaker** in the relay circuit to prevent damage due to short circuits or accidental overcurrent.
- Ensure the load circuit has no short circuits during wiring, and select protection components with appropriate current ratings if necessary.

3. Arc and Switching Protection

- Relay switching generates arcs, which can cause contact wear or welding.
- For inductive loads, it is recommended to use **RC snubber circuits** or **varistors** for arc suppression.

4. Installation Environment

- Do not use the relay in humid, high-temperature, flammable, explosive, or dusty environments.

- Install the relay securely to avoid vibrations or shocks that may cause misoperation or damage.

5. Power-Off Operation

- Always cut off power before maintenance, wiring, or replacing the relay to ensure personnel and device safety.
- Latching relays are only powered when changing state. Avoid strong vibrations or strong magnetic fields while the relay is unpowered.

6. Status Confirmation

- After powering on, confirm or reset the relay status as needed to prevent abnormal operation caused by transportation, installation, or external disturbances.
- Avoid power interruption during relay operation to prevent uncertain status or contact damage.

7. Regular Inspection

- Periodically inspect relay contacts, terminals, and insulation to ensure proper operation.
- If abnormal heating, odor, or burn marks are detected, immediately cut off power and replace the relay.

Usage Guide

ESP32-S3-Relay-1CH currently provides **Arduino IDE** development tool and framework, and you can choose the right development tool according to your project needs and personal habits.

Development Tool

Arduino IDE



(/wiki/File:270px-Arduino-IDE-logo.jpg)

Arduino IDE is an open source electronic prototyping platform, convenient and flexible, easy to get started. After a simple learning, you can start to develop quickly. At the same time, Arduino has a large global user community, providing an abundance of open source code, project examples and tutorials, as well as rich library resources, encapsulating complex functions, allowing developers to quickly implement various functions.

Components Preparation

- ESP32-S3-Relay-1CH x1
- USB cable (Type-A male to Type-C male) x1

- Modbus RTU Relay (<https://www.waveshare.com/Modbus-RTU-Relay.htm>) x 1
- USB TO RS232/485 (<https://www.waveshare.com/USB-TO-RS485-B.htm>) x 1

Before operating, it is recommended to browse the table of contents to quickly understand the document structure. For smooth operation, please read the FAQ carefully to understand possible problems in advance. All resources in the document are provided with hyperlinks for easy download.

Working with Arduino

This chapter introduces setting up the Arduino environment, including the Arduino IDE, management of ESP32 boards, installation of related libraries, program compilation and downloading, as well as testing demos. It aims to help users master the development board and facilitate secondary development.



(/wiki/File:Arduino-flow-04.png)

Environment Setup

Download and Install Arduino IDE

- Click to visit the Arduino official website (<https://www.arduino.cc/en/software>), select the corresponding system and system bit to download

Arduino IDE 2.3.3

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** ApplImage 64 bits (X86-64)
- Linux** ZIP file 64 bits (X86-64)
- macOS** Intel, 10.15: "Catalina" or newer, 64 bits
- macOS** Apple Silicon, 11: "Big Sur" or newer, 64 bits

[Release Notes](#)

(/wiki/File:ESP32-S3-AMOLED-1.91-Ar-software-01.png)

- Run the installer and install all by default

Install ESP32 Development Board

- Regarding ESP32-related motherboards used with the Arduino IDE, the '*esp32 by Espressif Systems*' library must be installed first.
- According to **Board installation requirement**, it is generally recommended to use **Install Online**. If online installation fails, use **Install Offline**
- For the installation tutorial, please refer to Arduino board manager tutorial (https://www.waveshare.com/wiki/Arduino_Board_Managers_Tutorial)
- ESP32-S3-Relay-1CH required development board installation instructions

Board name	Board installation requirement	Version number requirement
esp32 by Espressif Systems	"Install Offline" / "Install Online"	3.0.x

Install Library

- When installing Arduino libraries, there are usually two ways to choose from: **Install online** and **Install offline**. **If the library installation requires offline installation, you must use the provided library file**
For most libraries, users can easily search and install them through the online library manager of the Arduino software. However, some open-source libraries or custom libraries are not synchronized to the Arduino Library Manager, so they cannot be acquired through online searches. In this case, users can only manually install these libraries offline.
- For library installation tutorial, please refer to Arduino library manager tutorial (https://www.waveshare.com/wiki/Arduino_Library_Manager_Tutorial)
- ESP32-S3-Relay-1CH library file is stored in the demo, click here to jump: [ESP32-S3-Relay-1CH Demo](#)

- ESP32-S3-Relay-1CH library file installation instructions

Library Name	Description	Version	Library Installation Requirement
ArduinoJson	Lightweight JSON library	v6.21.4	"Install Online" or "Install Offline"
PubSubClient	MQTT message subscription publishing library	v2.8.0	"Install Online" or "Install Offline"
NTPClient	Network time synchronization client library	v3.2.1	"Install Online" or "Install Offline"

Run the First Arduino Demo

If you are just getting started with ESP32 and Arduino, and you don't know how to create, compile, flash, and run Arduino ESP32 programs, then please expand and take a look. Hope it can help you!

[\[Expand\]](#)

Demo

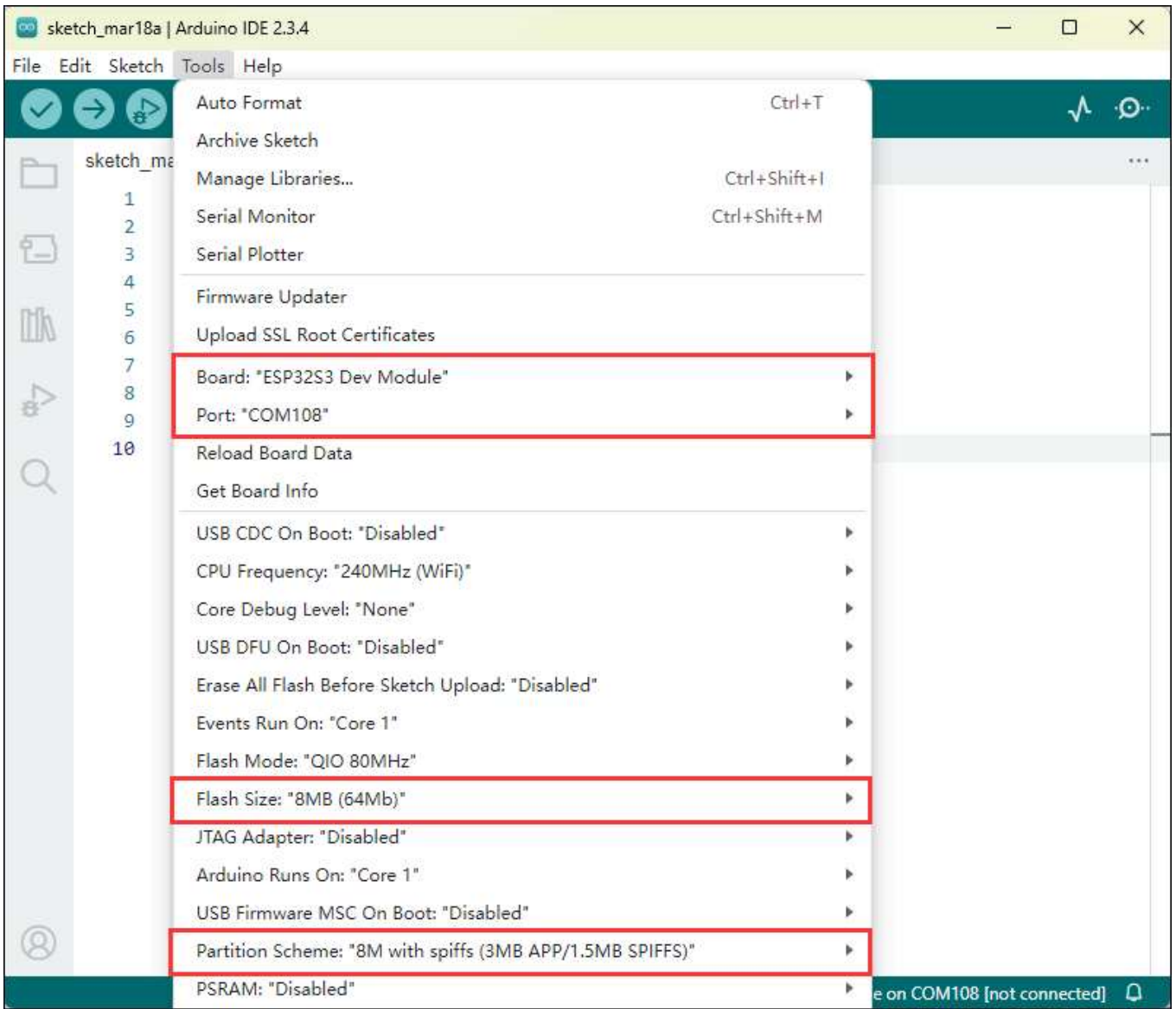


(/wiki/File:Demo-flow-01.png)

- ESP32-S3-Relay-1CH demos

Demo	Basic Description	Dependency Library
01_MAIN_WIFI_AP	RS485 interface control, Bluetooth control, Bluetooth send IP, Web page control (short distance)	Can be flashed directly Web pages are only available if they are connected to device WIFI
02_MAIN_WIFI_STA	RS485 interface control, Bluetooth control, Bluetooth send IP, Web page control (short distance)	Need to modify the WIFI network to connect to Web pages can only be used within the intranet
03_MAIN_WIFI_MQTT	RS485 interface control, Bluetooth control, Bluetooth transmit IP, Waveshare cloud control (long distance)	Need to modify the WIFI network to connect to The device must be created in the Waveshare cloud
04_MAIN_ALL	RS485 interface control, Bluetooth control, Bluetooth send IP, Web page control (short distance), Waveshare cloud control (long distance)	Need to modify the WIFI network to connect to The device must be created in the Waveshare cloud Web pages can only be used within the intranet

Arduino Project Parameter Setting



(/wiki/File:ESP32-S3-Relay-6CH-demo-10.png)

01_MAIN_WIFI_AP

Demo description

- This demo implements the control of 1 relay switch through WiFi, Bluetooth, and RS485. The AP mode of WiFi is enabled in this demo

Precautions

- Can be flashed directly
- Web pages are only available if they are connected to device WIFI

RS485 Control

- [Jump to RS485 Control Tutorial \(https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-RS485\)](https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-RS485)

Bluetooth Control

- [Jump to Bluetooth Control Tutorial \(https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Bluetooth\)](https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Bluetooth)

Web Page Control

- [Jump to Web Control Tutorial \(https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Web\)](https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Web), this demo is in AP mode

02_MAIN_WIFI_STA

Demo description

- This example implements the control of 1 relay switch through WiFi, Bluetooth, and RS485. The STA mode of WiFi is enabled in this example

Precautions

- Need to modify the WIFI network to connect to
- The Web page only supports controlling the device and this product to be used under the same network, if you use the mobile phone control, you need to turn off the mobile network

RS485 Control

- [Jump to RS485 Control Tutorial \(https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-RS485\)](https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-RS485)

Bluetooth Control

- [Jump to Bluetooth Control Tutorial \(https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Bluetooth\)](https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Bluetooth)

Web Page Control

- Jump to Web Control Tutorial (<https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Web>), this demo is in STA mode

03_MAIN_WIFI_MQTT

Demo description

- This example controls 1 relay through MQTT, Bluetooth, and RS485 communication methods. It uses ESP32 as the main control unit, supports Wi-Fi and Bluetooth connections, and provides remote control based on the MQTT protocol

Precautions

- Need to modify the the WIFI network to connect to
- Must create the device in Waveshare Cloud

RS485 Control

- Jump to RS485 Control Tutorial (<https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-RS485>)

Bluetooth Control

- Jump to Bluetooth Control Tutorial (<https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Bluetooth>)

Waveshare Cloud Control

- Jump to Waveshare Cloud Control (https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Waveshare_Cloud)

04_MAIN_ALL

Demo description

- This example is a collection of RS485 interface control, Bluetooth control, Web page control (near distance), and Waveshare cloud control (long distance).

Precautions

- Need to modify the the WIFI network to connect to
- The Web page only supports controlling the device and this product to be used under the same network, if you use the mobile phone control, you need to turn off the mobile network
- Must create the device in Waveshare Cloud

RS485 Control

- Jump to RS485 Control Tutorial (<https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-RS485>)

Bluetooth Control

- Jump to Bluetooth Control Tutorial (<https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Bluetooth>)

Web Page Control

- Jump to Web Control Tutorial (<https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Web>), this demo is in AP mode

Waveshare Cloud Control

- Jump to Waveshare Cloud Control (https://www.waveshare.com/wiki/ESP32-S3-Relay-1CH-Waveshare_Cloud)

External Expansions

RS485 Extended Relay Channels

[\[Expand\]](#)

Flash Firmware Flashing and Erasing

- The current demo provides test firmware, which can be used to test whether the onboard device functions properly by directly flashing the test firmware
- bin file path:

```
..\ESP32-S3-Relay-1CH\Firmware\Factory bin
```

Flash firmware flashing and erasing (https://www.waveshare.com/wiki/Flash_Firmware_Flashing_and_Erasing) for reference

Working with Homeassistant

The product can be controlled online through the Homeassistant built on the Raspberry Pi, and the relevant operations can be found in Reference link (https://www.waveshare.com/wiki/Using_Modbus_RTU_Relay_with_Homeassistant)

Resources

Demo

- ESP32-S3-Relay-1CH Demo (<https://files.waveshare.com/wiki/ESP32-S3-Relay-1CH/ESP32-S3-Relay-1CH-Demo.zip>)

Schematic

- ESP32-S3-Relay-1CH Schematic (<https://files.waveshare.com/wiki/ESP32-S3-Relay-1CH/ESP32-S3-Relay-1CH-schematic.pdf>)

Datasheets

ESP32-S3

- ESP32-S3 Datasheet (https://files.waveshare.com/wiki/common/Esp32-s3_datasheet_en.pdf)
- ESP32-S3 Technical reference manual (https://files.waveshare.com/wiki/common/Esp32-s3_technical_reference_manual_en.pdf)

Software Tools

Arduino

- Arduino IDE Official download link (<https://www.arduino.cc/en/software/>)

- ESP32_Arduino offline package (https://drive.google.com/drive/folders/1Pcs_A4FKWvdSHnz9IEBYqOpr-noTMblv?usp=sharing)

VScode

- VScode official website (<https://code.visualstudio.com/download>)

Thonny

- Thonny official download link (<https://thonny.org/>)

Debugging Tools

- Flash_download_tool (https://dl.espressif.com/public/flash_download_tool.zip)
- SSCOM (https://files.waveshare.com/wiki/ESP32-S3-Relay-6CH/SSCOM5.13.1_For_ESP32_S3_Relay_6CH.zip)
- Bluetooth debugging tool (nRF Connect) (<https://www.nordicsemi.com/Products/Development-tools/nRF-Connect-for-mobile>)

Other Resource Links

- ESP32-Arduino official documentation (<https://docs.espressif.com/projects/arduino-esp32/en/latest/index.html>)

FAQ

Question: After the module downloads the demo and re-downloads it, why sometimes it can't connect to the serial port or the flashing fails?

Answer:

- Long press the BOOT button, press RESET at the same time, then release RESET, then release the BOOT button, at this time the module can enter the download mode, which can solve most of the problems that can not be downloaded.

Question: Why does the module keep resetting and flicker when viewed the recognition status from the device manager?

Answer:

- It may be due to Flash blank and the USB port is not stable, you can long-press the BOOT button, press RESET at the same time, and then release RESET, and then release the BOOT button, at this time the module can enter the download mode to flash the firmware (demo) to solve the situation.

Question: How to deal with the first compilation of the program being extremely slow?**Answer:**

- It's normal for the first compilation to be slow, just be patient

Question: How to handle the display "waiting for download..." on the serial port after successfully ESP-IDF flashing?**Answer:**

- If there is a reset button on the development board, press the reset button; if there is no reset button, please power it on again

Question: What should I do if I can't find the AppData folder?**Answer:**

- Some AppData folders are hidden by default and can be set to show.
- English system: Explorer->View->Check "Hidden items"
- Chinese system: File Explorer -> View -> Display -> Check "Hidden Items"

Question: How do I check the COM port I use?**Answer:**

- Windows system:

①View through Device Manager: Press the Windows + R keys to open the "Run" dialog box; input devmgmt.msc and press Enter to open the Device Manager; expand the "Ports (COM and LPT)" section, where all COM ports and their current statuses will be listed.

②Use the command prompt to view: Open the Command Prompt (CMD), enter the "mode" command, which will display status information for all COM ports.

③Check hardware connections: If you have already connected external devices to the COM port, the device usually occupies a port number, which can be determined by checking the connected hardware.

- Linux system:

- ① Use the `dmesg` command to view: Open the terminal.
- ① Use the `ls` command to view: Enter `ls /dev/ttyS*` or `ls /dev/ttyUSB*` to list all serial port devices.
- ③ Use the `setserial` command to view: Enter `setserial -g /dev/ttyS*` to view the configuration information of all serial port devices.

Question: Why does the program flashing fail when using a MAC device?

Answer:

- Install MAC Driver (https://files.waveshare.com/wiki/common/CH34XSER_MAC.7z) and flash again.

Question: Why is there no output after successfully flashing the code with no issues?

Answer:

- Check the schematic diagram for different development boards with Type-C interfaces, and handle the output accordingly:
 - For development boards with direct USB output, `printf` function is supported for printing output. If you want to support output via the Serial function, you will need to enable the USB CDC On Boot feature or declare `HWCDC`.
 - For development boards with UART to USB conversion, both `printf` and Serial functions are supported for printing output, and there is no need to enable USB CDC On Boot.

Question: Please note!

Answer:

- The factory demo is for learning only, if it is used for practical application, please optimize the demo logic by yourself.

Question: When controlling other devices using RS485, it is not sensitive or communication fails?

Answer:

- Please move the jumper cap to 120R and try again. Some RS485 devices require a 120R resistor to be connected in series

Support

Technical Support

If you need technical support or have any feedback/review, please click the **Submit Now** button to submit a ticket, Our support team will check and reply to you within 1 to 2 working days. Please be patient as we make every effort to help you to resolve the issue.

Working Time: 9 AM - 6 PM GMT+8
(Monday to Friday)

Submit Now (<https://service.waveshare.com/>)

*Retrieved from "<https://www.waveshare.com/w/index.php?title=ESP32-S3-Relay-1CH&oldid=107042>
(<https://www.waveshare.com/w/index.php?title=ESP32-S3-Relay-1CH&oldid=107042>)"*
