

RUIZHI ESP32

RUIZHI ESP32-WROOM-32D Mini Development Board User Manual

Model: ESP32 | Brand: RUIZHI

1. INTRODUCTION

The RUIZHI ESP32-WROOM-32D Mini Development Board is a versatile and powerful Wi-Fi and Bluetooth Low Energy (BLE) MCU module. Designed for a wide range of applications, it offers robust performance and flexible development options, making it suitable for IoT projects, embedded systems, and more. This manual provides essential information for setting up, operating, and maintaining your ESP32 development board.

2. KEY FEATURES

- **Compact Design:** Small size for easy integration into various products.
- **Powerful Performance:** Equipped with a dual-core processor, supporting LWIP protocol and FreeRTOS for exceptional project performance.
- **Integrated Connectivity:** Combines Wi-Fi and Bluetooth 2-in-1 functionalities, offering flexible connectivity options.
- **Multiple Operating Modes:** Supports AP (Access Point), STA (Station), and AP+STA coexistence modes to meet diverse connectivity needs.
- **Lua Programming Support:** Simplifies development, compatible with users of all programming levels.
- **Type-C Interface:** Modern and convenient USB-C port for power and data.
- **Stable USB-C to TTL CH340 Chip:** Ensures reliable serial communication.
- **Automatic Download:** Facilitates easy program uploading and execution without manual switching.
- **Windows System Compatibility:** Compatible with Cygwin and MSYS32 environments for flexible development.

3. PACKAGE CONTENTS

Each package contains:

- 3 x RUIZHI ESP32-WROOM-32D Mini Development Boards

4. SETUP INSTRUCTIONS

To begin using your RUIZHI ESP32 development board, follow these steps:

1. **Driver Installation:** Connect the ESP32 board to your computer using a USB Type-C cable. The board uses a CH340 USB-to-serial chip. If your operating system does not automatically install the necessary drivers, you may need to download and install them manually. Drivers are typically available from the CH340 manufacturer's website or common ESP32 development resources.
2. **Development Environment Setup:**
 - **Arduino IDE:** Install the Arduino IDE. Then, add the ESP32 board manager URL to the IDE preferences and install the ESP32 boards package. This allows you to program the ESP32 using the Arduino framework.
 - **ESP-IDF:** For more advanced development, install the Espressif IoT Development Framework (ESP-IDF). Follow the official Espressif documentation for detailed installation instructions for your operating system (Windows, macOS, Linux).
 - **Lua Programming:** If you plan to use Lua, ensure your development environment supports Lua for ESP32, such as NodeMCU firmware.
3. **Connecting to a Breadboard/Breakout Board:** The ESP32 board can be inserted into a breadboard or a dedicated breakout board for easier access to its pins. Ensure correct orientation and secure connection.

Application ESP32 CH340

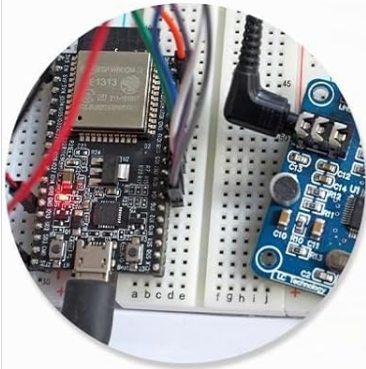
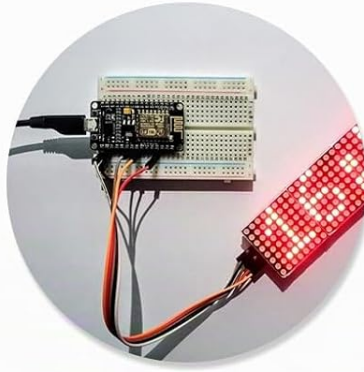


Image: The ESP32 development board being connected to a larger breakout board, illustrating how to expand pin access for prototyping.

5. OPERATING INSTRUCTIONS

Once your development environment is set up, you can program and operate the ESP32 board.

5.1. Programming the ESP32

1. **Connect:** Ensure the ESP32 board is connected to your computer via the USB Type-C cable.
2. **Select Board and Port:** In your Arduino IDE or other development environment, select "ESP32 Dev Module" (or similar) as the board and choose the correct COM port assigned to your ESP32.
3. **Upload Code:** Write or load your desired code (e.g., a simple "Blink" sketch or a Wi-Fi example). Click the "Upload" button. The CH340 chip facilitates automatic download, so manual boot mode selection is typically not required.
4. **Monitor Serial Output:** Open the Serial Monitor in your IDE to view debug messages or data from your ESP32.

5.2. Wi-Fi and Bluetooth Modes

The ESP32 supports three primary operating modes for wireless communication:

- **Station (STA) Mode:** The ESP32 connects to an existing Wi-Fi network (e.g., your home router) as a client device.
- **Access Point (AP) Mode:** The ESP32 creates its own Wi-Fi network, allowing other devices to connect to it.
- **AP+STA Coexistence Mode:** The ESP32 operates as both an Access Point and a Station simultaneously.

Bluetooth functionality can be implemented using the ESP-IDF or specific libraries within the Arduino framework, enabling communication with other Bluetooth-enabled devices.

6. PINOUT DIAGRAM

Understanding the pinout is crucial for connecting peripherals and designing circuits. Refer to the diagram below for the pin assignments of the RUIZHI ESP32-WROOM-32D Mini Development Board.

ESP32 Connect

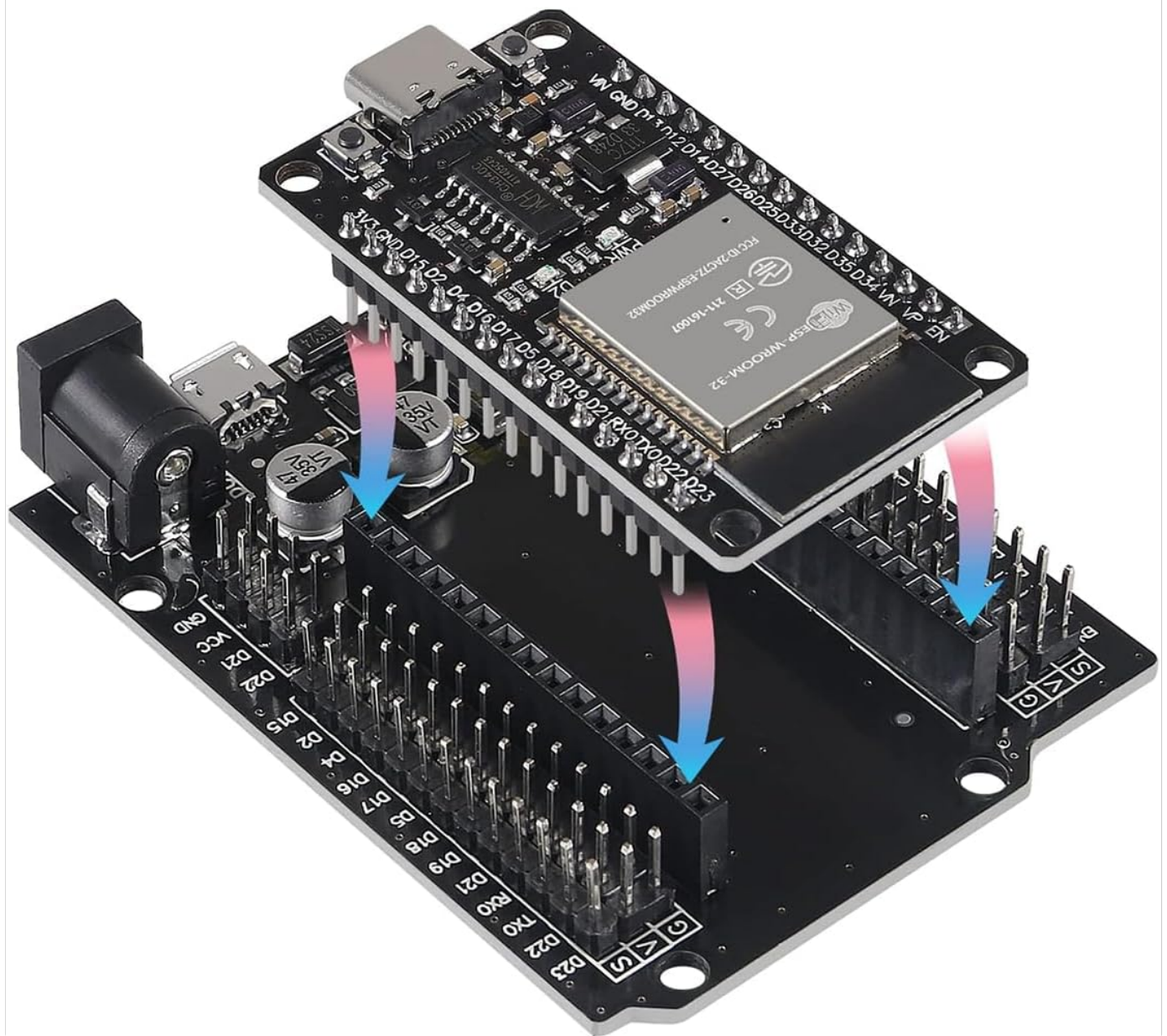


Image: Detailed pinout diagram for the ESP32-WROOM-32D module, showing the function of each pin including power, ground, GPIO, ADC, DAC, and communication interfaces.

Note: While this diagram provides a general overview, always consult the official Espressif documentation for the most accurate and up-to-date pin functionality and limitations, especially concerning strapping pins and their boot-time behavior.

7. EXAMPLE APPLICATIONS

The RUIZHI ESP32 development board is suitable for a wide array of projects, including:

- Internet of Things (IoT) devices

- Home automation systems
- Wearable electronics
- Smart sensors and actuators
- Robotics and drone control
- Educational projects

Détail ESP32

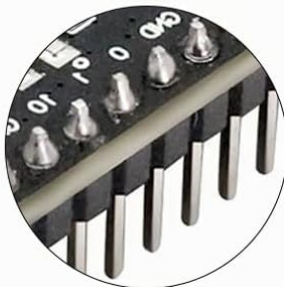
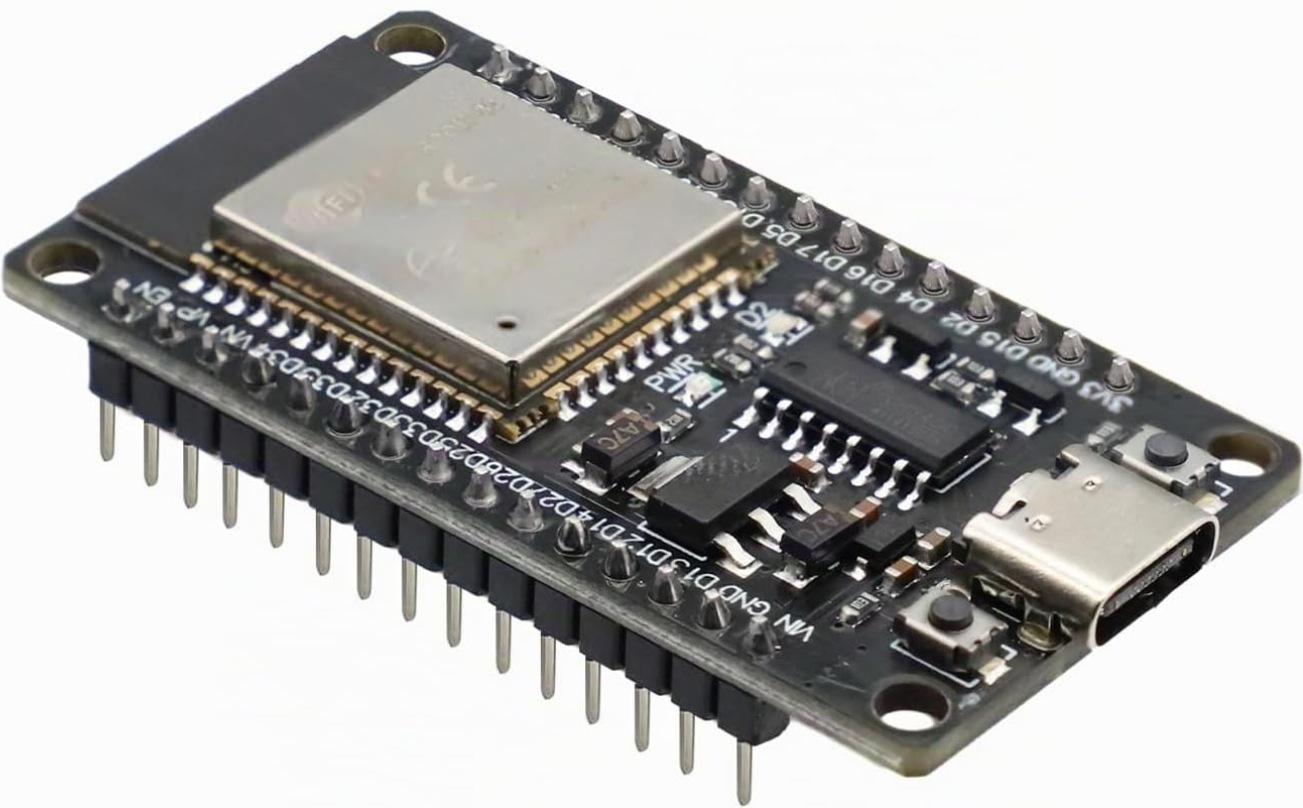


Image: Examples of the ESP32 board integrated into different projects, such as controlling an LED matrix and connecting to other modules.

8. TECHNICAL SPECIFICATIONS

Feature	Detail
Brand	RUIZHI

Feature	Detail
Model Name	ESP32
Model Number	YY1-0092
Part Number	XU143
Connectivity Technology	Bluetooth, Wi-Fi
Wireless Communication Standard	Bluetooth
Operating System	FreeRTOS
CPU Manufacturer	Espressif
Number of Processors	2 (Dual-core)
RAM Memory Technology	LPDDR
Total USB Ports	1 (Type-C)
Compatible Devices	Multiple devices
Number of Items in Package	3

Taille ESP32

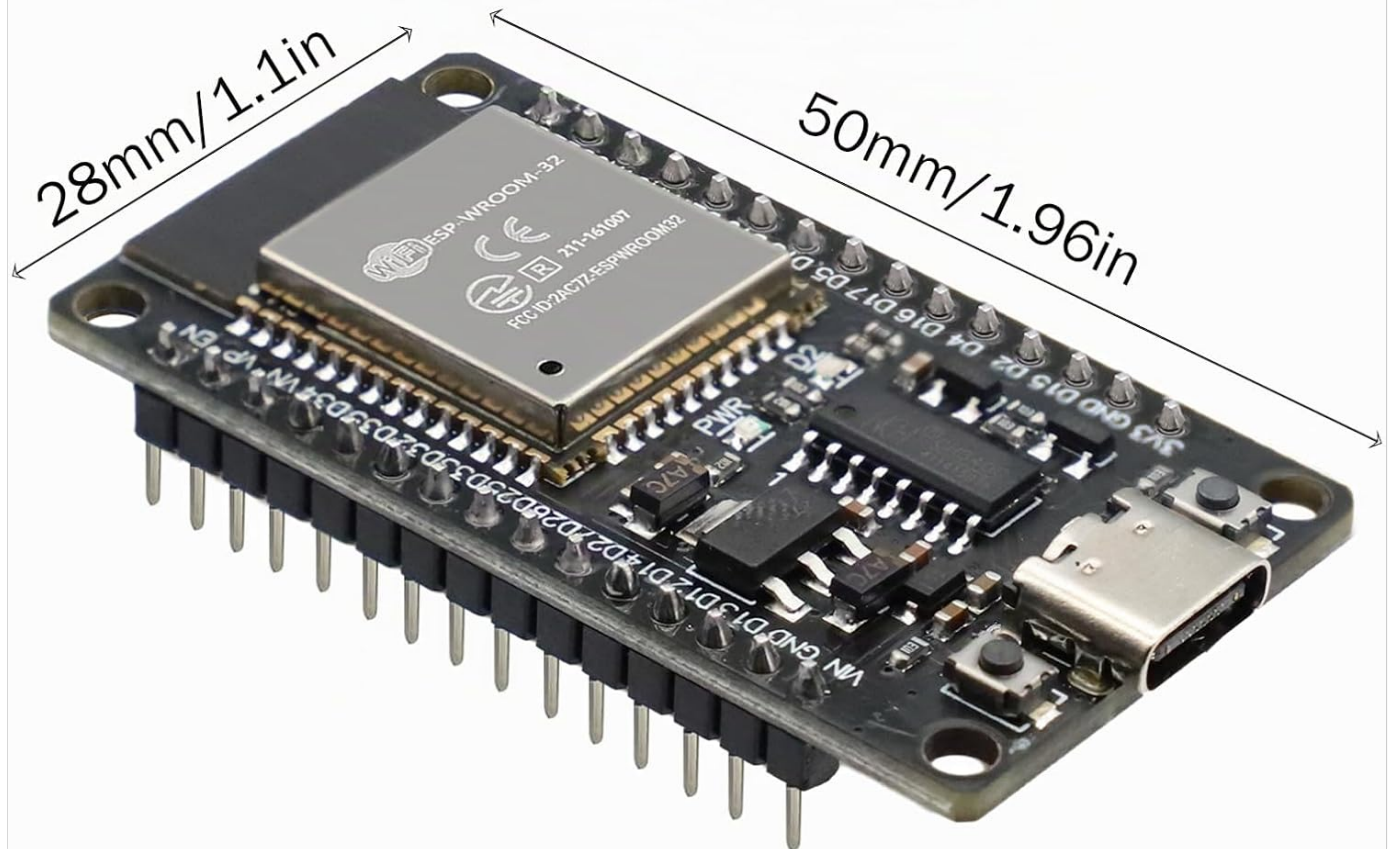


Image: The ESP32 development board displaying its physical dimensions, approximately 50mm (1.96in) in length and 28mm (1.1in) in width.

Avant

Arrière

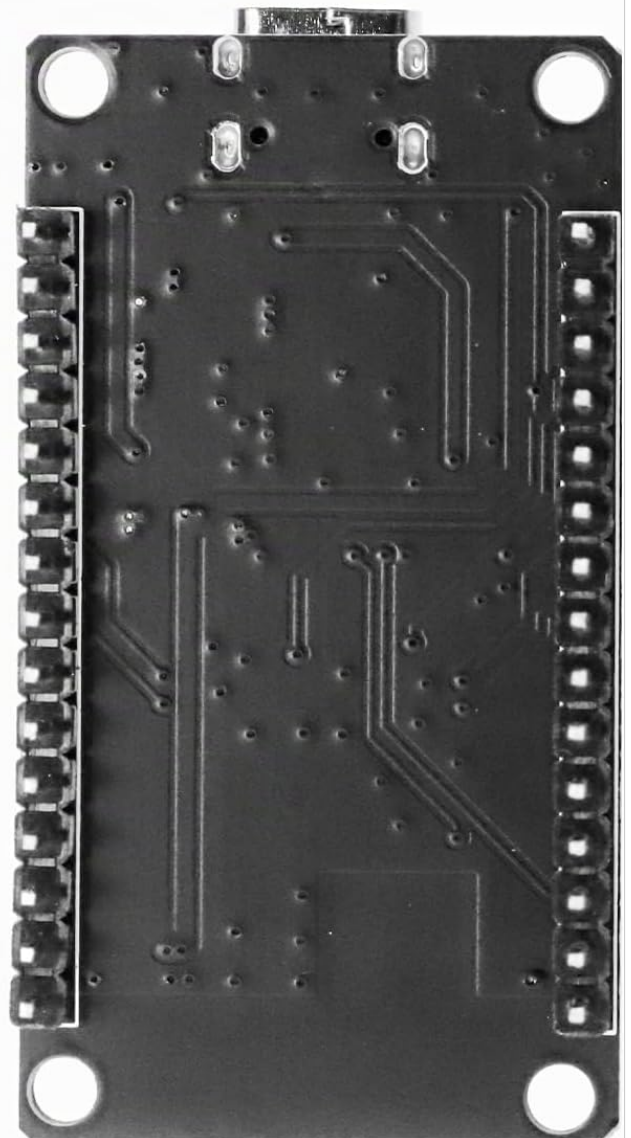
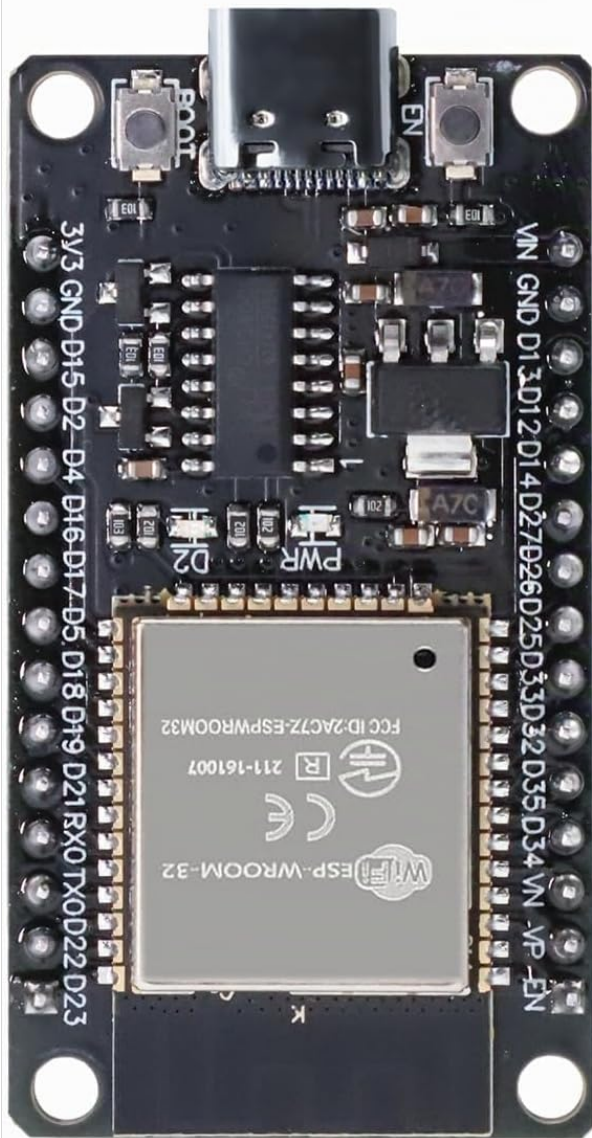


Image: A close-up view of the ESP32 board, highlighting key components such as the ESP-WROOM-32 module, USB-C port, and various integrated circuits.

9. MAINTENANCE

To ensure the longevity and reliable operation of your RUIZHI ESP32 development board, follow these maintenance guidelines:

- **Handle with Care:** Avoid dropping the board or subjecting it to physical shock.
- **Static Discharge:** Always handle the board in an anti-static environment or take precautions to prevent electrostatic discharge (ESD), which can damage sensitive electronic components.
- **Keep Dry:** Protect the board from moisture and liquids. Operate it in a dry environment.
- **Cleanliness:** Keep the board free from dust and debris. Use a soft, dry brush or compressed air for cleaning. Avoid using liquid cleaners.
- **Proper Storage:** When not in use, store the board in an anti-static bag or a protective enclosure to prevent damage.

- **Power Supply:** Ensure you use a stable and appropriate power supply (typically 5V via USB-C or VIN pin) to prevent damage from overvoltage or undervoltage.

10. TROUBLESHOOTING

If you encounter issues with your ESP32 development board, consider the following troubleshooting steps:

- **Board Not Detected:**
 - Verify that the USB Type-C cable is securely connected to both the board and your computer.
 - Ensure the CH340 USB-to-serial driver is correctly installed on your computer. Check Device Manager (Windows) or `/dev/tty*` (Linux/macOS) for the serial port.
 - Try a different USB cable or USB port on your computer.
- **Upload Errors:**
 - Confirm that the correct board type (e.g., "ESP32 Dev Module") and COM port are selected in your IDE.
 - Ensure no other application is using the serial port.
 - If automatic upload fails, try manually putting the ESP32 into bootloader mode by holding down the "BOOT" button, pressing and releasing "EN" (Reset), then releasing "BOOT" before uploading.
 - Check for syntax errors in your code.
- **Wi-Fi/Bluetooth Connectivity Issues:**
 - Double-check your Wi-Fi SSID and password in your code.
 - Ensure the ESP32 is within range of the Wi-Fi access point or Bluetooth device.
 - Verify that your code correctly initializes and manages Wi-Fi/Bluetooth connections.
- **Unexpected Behavior/Crashes:**
 - Check your power supply. Insufficient power can lead to instability.
 - Review your code for logic errors, memory leaks, or incorrect pin assignments.
 - Use the Serial Monitor to capture any error messages or stack traces.

For more detailed troubleshooting, refer to the extensive online resources and community forums available for ESP32 development.

11. WARRANTY INFORMATION

Specific warranty details for the RUIZHI ESP32-WROOM-32D Mini Development Board are not provided in the product information. Please refer to the retailer's return policy or contact the seller directly for warranty terms and conditions.

12. CUSTOMER SUPPORT

For technical assistance, product inquiries, or support, please contact your point of purchase or refer to the RUIZHI official website (if available) for contact information. Online communities and forums dedicated to ESP32 development are also excellent resources for troubleshooting and project guidance.

ASK A QUESTION ABOUT THIS MANUAL

Ask about setup, troubleshooting, compatibility, parts, safety, or missing instructions. Manuals+ will review the question and use this page's manual context to help answer it.

Name

Anonymous

Email

you@example.com

After a successful submission, we securely hash the supplied account or form email before sending it to Microsoft Advertising for conversion attribution. [Privacy details](#).

Question

Example: How do I reset this device or fix this error?

Details

Model number, symptoms, what you tried, or the section of the manual you are using.

Submit question