



! VERSION COMPATIBILITY NOTES

Please note: The ESP32-S3-Touch-LCD-2.8 V1 version has been discontinued. Shipments have switched to the V2 version as of June 5, 2026.

- V1 version features the ST7789 driver chip and CST328 capacitive touch chip.
- V2 version features the ST7789 driver chip and CST3530 capacitive touch chip.
- You can identify the version by the label on the back of the product.

ESP32-S3-Touch-LCD-2.8



The [ESP32-S3-Touch-LCD-2.8](#) is a microcontroller development board that supports 2.4GHz Wi-Fi and Bluetooth BLE 5. It integrates large capacity Flash and PSRAM and has onboard 2.8inch capacitive touch screen, can smoothly run GUI programs such as LVGL. Combined with various peripheral interfaces, it is suitable for the quick development of the HMI and other ESP32-S3 applications.

If you are looking for:

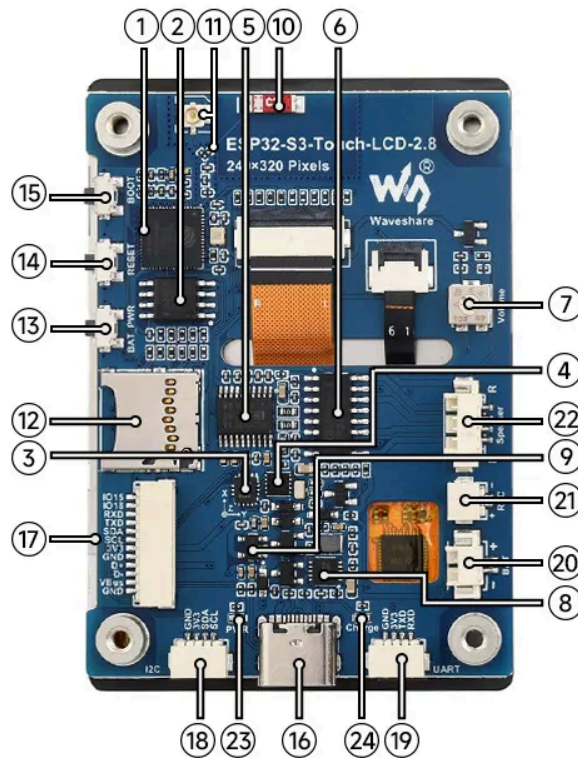
- **Coding with AI tools:** Click **Ask AI** and select an AI tool to ask questions about this page, or choose **Copy page** to copy its content and paste it into an AI tool for coding assistance.
- **Schematics, datasheets, and example code:** See [Resources](#)
- **Arduino development:** See [Working with Arduino](#)
- **ESP-IDF development:** See [Working with ESP-IDF](#)
- **Troubleshooting:** See [Product FAQ](#) or contact [Technical Support](#)

SKU	Product
27690	ESP32-S3-Touch-LCD-2.8

Features

- Powered by a high-performance Xtensa 32-bit LX7 dual-core processor, with a main frequency of up to 240MHz
- Supports 2.4 GHz Wi-Fi (802.11 b/g/n) and Bluetooth 5 (LE) with an onboard antenna
- Onboard 512 KB SRAM and 384 KB ROM, with stacked 16 MB Flash and 8 MB PSRAM
- Onboard 2.8inch capacitive touch screen, 240 × 320 resolution, 262K colors
- Supports I2C interface control for capacitive touch, with 5-point touch, and supports interrupts
- Exposes UART, I2C, and some I/O interfaces; integrated full-speed USB serial port
- Onboard speaker, QMI8658 6-axis IMU, RTC clock sensor, TF card slot, and battery charging management module
- Supports flexible clock, multiple power modes, and other precise controls to enable low-power operation in various scenarios.

Onboard Resources



1. **ESP32-S3R8** Dual-core processor, operating frequency up to 240MHz
2. **16MB Flash**
3. **QST Attitude Sensor** QMI8658 (6-axis gyroscope + accelerometer)
4. **RTC chip** PCF85063 RTC clock
5. **PCM5101 audio decoder chip**
6. **Power amplifier chip**
7. **Volume adjustment knob**
8. **Battery charging management chip**
9. **ME6217C33M5G** Low dropout LDO, current (Max) 800mA
10. **Onboard ceramic antenna**
11. **IPEX1 Connector and Switch Resistor** Switching to use the external antenna via resoldering the resistor
12. **TF card slot**
13. **Battery Power Control Button** Relevant driver program is required
14. **RESET button**
15. **BOOT button**
16. **USB Type-C Interface** For programming and log output
17. **12PIN Multi-function Interface**

18. **I2C header** Internally connected to other chips; only external I2C devices may be connected, cannot be remapped to other functions
19. **UART Port**
20. **Battery header** MX1.25 2PIN connector for 3.7V lithium battery, supports charging and discharging
21. **RTC battery header** For connecting a rechargeable RTC battery
22. **Speaker header** Comes with 8Ω 2W 2030 speaker
23. **Power indicator**
24. **Charging indicator** When a system battery is connected, it stays on while charging and turns off when fully charged (status indeterminate when no system battery is connected)

Interface Description

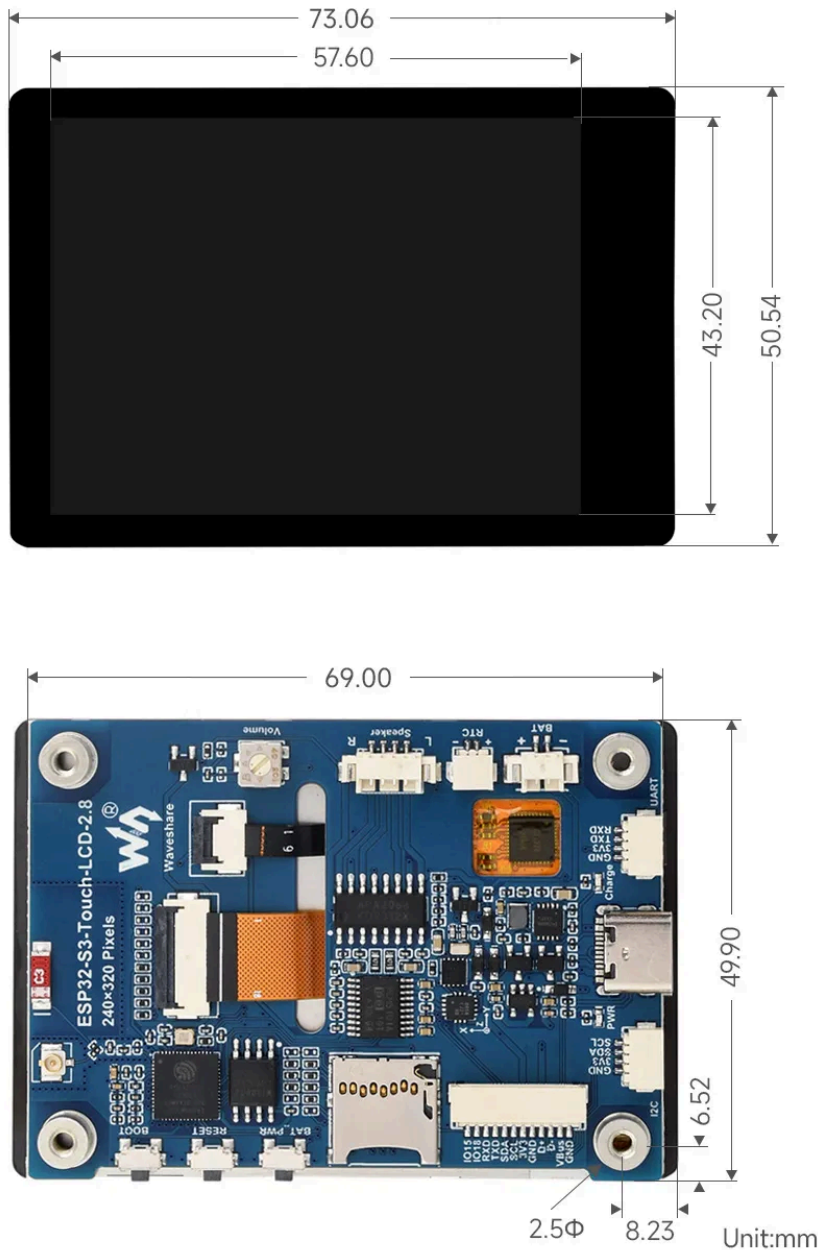
When using the ESP32-S3-Touch-LCD-2.8, it is important to understand the following interface descriptions.

▶ **12PIN Wire Interface Pin Mapping Table (click to expand)**

▶ **I2C Interface Pin Mapping Table (click to expand)**

▶ **UART Interface Pin Mapping Table (click to expand)**

Dimensions



Development Methods

The ESP32-S3-Touch-LCD-2.8 supports both the Arduino IDE and ESP-IDF development frameworks, giving developers flexible options. You can choose the development tool that best suits your project needs and personal preferences.

Arduino generally offers a gentler learning curve and may be more approachable for beginners and hobbyists. ESP-IDF provides advanced tooling and finer control over system behavior, making it better suited to complex projects and applications with demanding performance requirements.

- **Arduino IDE** is a convenient, flexible, and easy-to-use open-source electronics prototyping platform. It requires minimal foundational knowledge, allowing for rapid development after a short learning period. Arduino has a huge global user community, providing a vast amount of open-source code, project examples, and tutorials, as well as a rich library ecosystem that encapsulates complex functions, enabling developers to implement various features rapidly. You can refer to the **Working with Arduino** to complete the initial setup, and the tutorial also provides related example programs for reference.
- **ESP-IDF**, short for Espressif IoT Development Framework, is a professional development framework launched by Espressif Systems for its ESP series of chips. It is based on C language development and includes compilers, debuggers, flashing tools, etc. It supports development via command line or integrated development environments (such as Visual Studio Code with the Espressif IDF plugin), which provides features like code navigation, project management, and debugging. We recommend using VS Code for development. For the specific configuration process, please refer to the **Working with ESP-IDF**. The tutorial also provides relevant example programs for reference.

 [Give Feedback](#)