EC5R28T

2.8-inch TN ESP32-C5 Display Module

User Manual

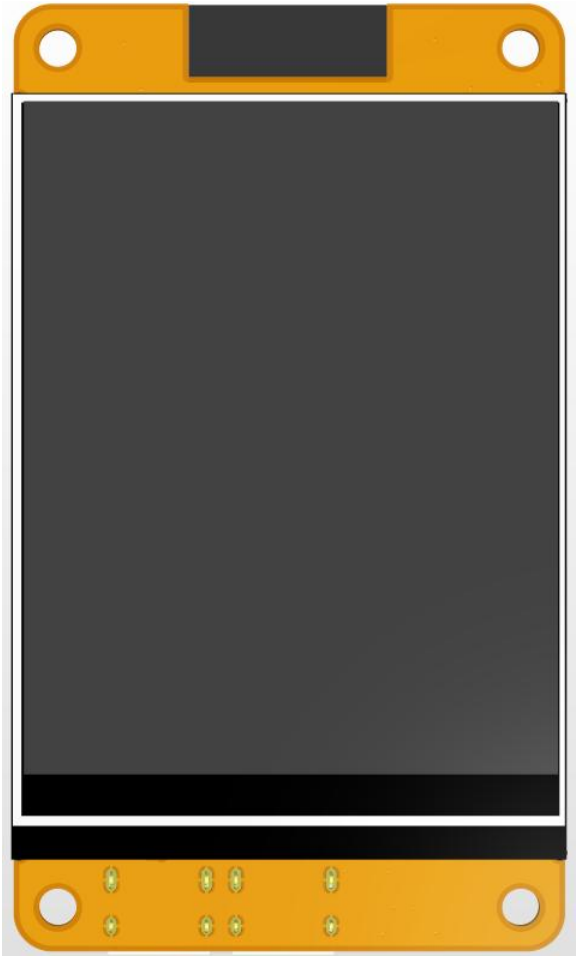

Table of Contents

1. Data Package Description

The data directory is shown in the figure below:

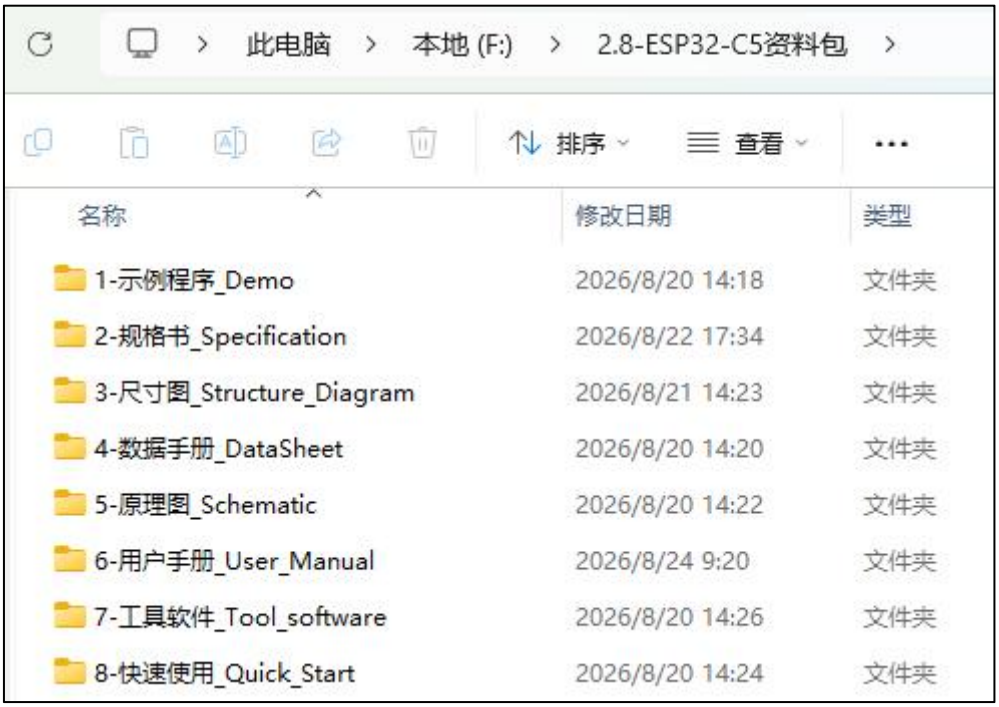

Figure 1.1 Product Data Package Directory

Directory	Content Description
1-Example_Programs_Demo	Example program code, third-party software libraries that the example programs depend on, software development environment setup documentation, and example program documentation.
2-Specification	Display module product specification.
3-Structure_Diagram	Display module product dimension drawing
4-DataSheet	LCD display driver ST7789 datasheet, resistive touchscreen driver XPT2046 datasheet, ESP32-C5 main controller datasheet and hardware design guidance document, USB-to-serial IC (CH340C) datasheet, AMS1117 voltage regulator datasheet, audio amplifier chip FM8002E datasheet, and built-in chip RGB tri-color LED datasheet.
5-Schematic	Product hardware schematic, ESP32-C5 module IO resource allocation table
6-User_Manual	Product user documentation

7-Tool_Software	Wi-Fi and Bluetooth test APP and debugging tools, ESP32 Flash download tool software, character font extraction software, image extraction software, JPG image processing software, and serial debugging tools.
8-Quick_Start	Bin files to be flashed, flash download tool, and usage documentation.

2. Software Usage Instructions

The display module software development steps are as follows:

A. Set up the ESP32 platform software development environment;

B. If needed, import third-party software libraries as the development foundation;

C. Open the software project to be debugged, or create a new software project;

D. Power on the display module, compile and download the debug program, then check the software running results;

E. If the software results do not meet expectations, continue modifying the program code, then compile and download until the results meet expectations;

For detailed instructions on the above steps, see the documentation in the "1-Example_Programs_Demo" directory.

3. Hardware Usage Instructions

3.1. Display Module Hardware Resources Introduction

The module hardware resources are shown in the following two figures:

Figure 3.1 Module Hardware Resources 1

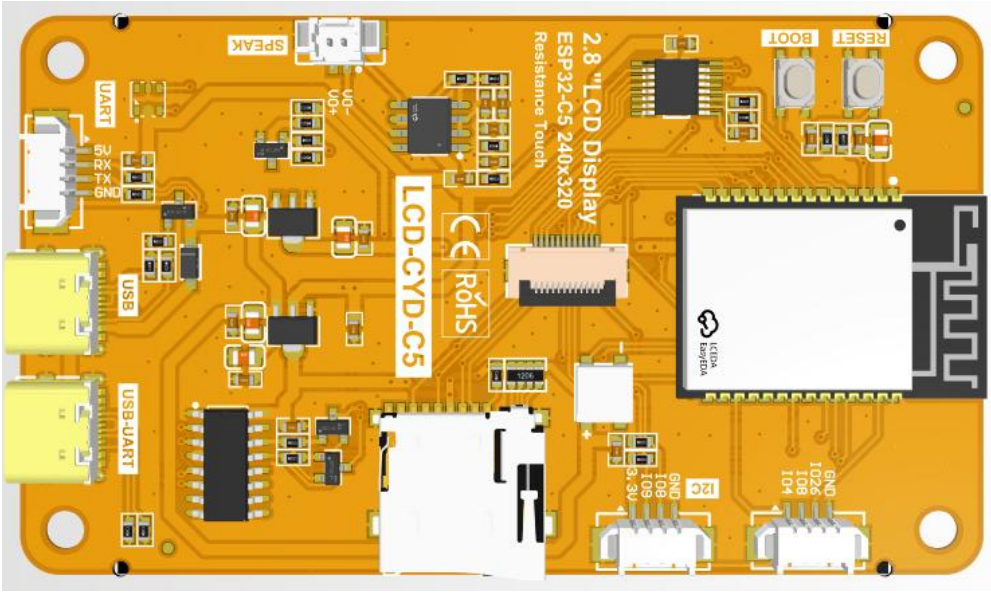

Figure 3.2 Module Hardware Resources 2

Hardware resources are described as follows:

1) LCD Screen

The LCD screen display size is 2.8 inches, the driver IC is ST7789, and the resolution is 240x320. It is connected to the ESP32-C5 using a 4-wire SPI communication interface.

A、ST7789 Controller Introduction

The ST7789 controller supports a maximum resolution of 240x320 and has a 172800-byte GRAM. It also supports 8-bit and 9-bit parallel data buses, 6-bit RGB interface, as well as 3-wire, 4-wire, and 2-data-line SPI serial interfaces. Since parallel control requires a large number of IO pins, SPI serial control is the most commonly used. The ST7789P3 also supports 65K and 262K RGB color display with rich colors, and supports rotation display, scrolling display, and video playback, providing diverse display methods.

The ST7789P3 controller uses 16-bit (RGB565) to control a single pixel display, so each pixel can display up to 65K colors. Pixel address setting is performed in row-column order, and the increment/decrement direction is determined by the scanning method. The ST7789P3 display method works by first setting the address and then setting the color value.

B. SPI Communication Protocol Introduction

The 4-wire SPI bus write mode timing is shown in the figure below:

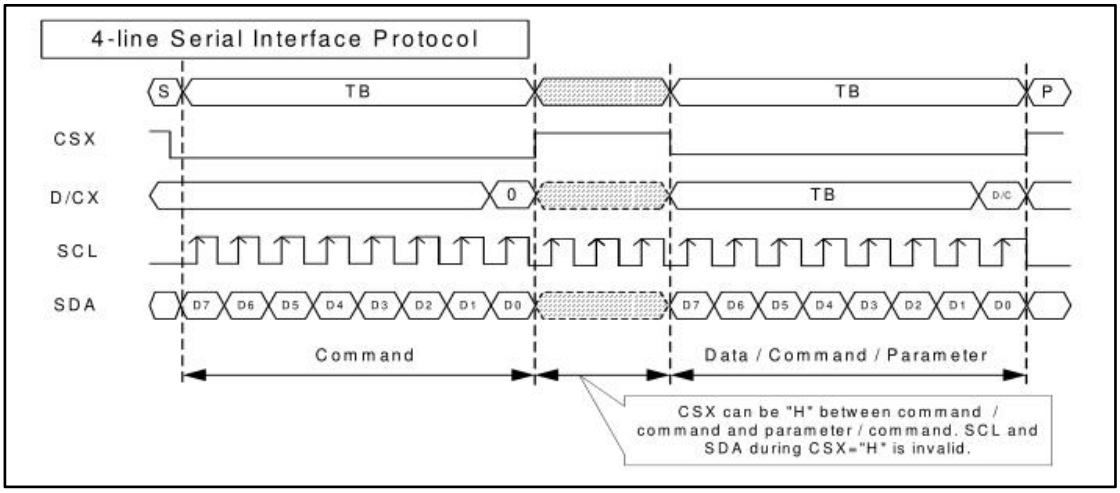

Figure 3.3 4-wire SPI Bus Write Mode Timing

CSX is the slave chip select. The chip is enabled only when CSX is at a low level.

D/CX is the data/command control pin of the chip. When DCX is low, a command is written; when high, data is written.

SCL is the SPI bus clock. One bit of data is transmitted on each rising edge;

SDA is the data transmitted via SPI, transmitting 8 bits of data at a time. The data format is shown in the figure below:

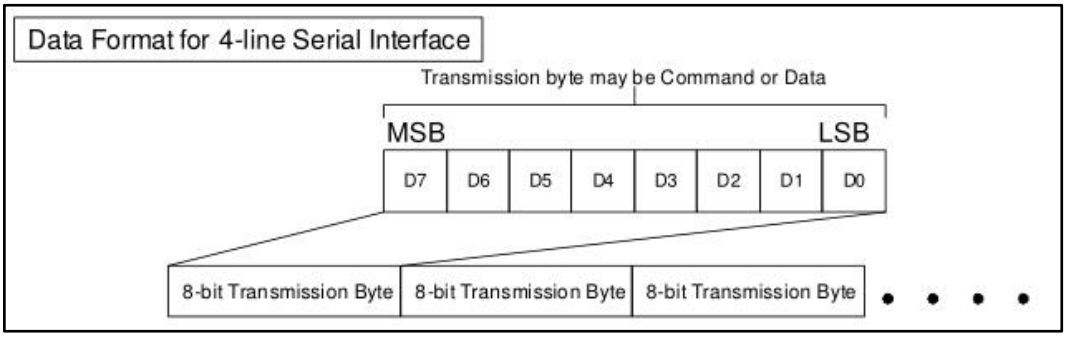

Figure 3.4 SPI Data Transmission Format

MSB first, transmitted first.

For SPI communication, data has transmission timing, which is a combination of clock phase (CPHA) and clock polarity (CPOL)

combination:

The level of CPOL determines the idle state level of the serial synchronous clock. CPOL = 0 means low level. CPOL has little effect on the transmission protocol;

The level of CPHA determines whether data is sampled on the first or second clock transition edge. When CPHA = 0, data is sampled on the first transition edge;

These two combinations result in four SPI communication modes. SPI0 is commonly used, i.e., CPHA = 0, CPOL = 0.

2) Resistive Touchscreen

The resistive touchscreen size is 2.8 inches, connected to the XPT2046 controller IC through four pins: XL, XR, YU, and YD.

3) ESP32-C5 Module

Built-in ESP32-C5 chip, RISC-V 32-bit single-core processor, supporting clock frequencies up to 240 MHz. It has 320KB ROM, 384KB HP SRAM, 16KB LP SRAM, 8MB OPI PSRAM, and external 16MB QSPI Flash. Supports 2.4 & 5 GHz dual-band Wi-Fi, Bluetooth V6.0, and low-power Bluetooth. It has 22 GPIO pins, supporting GPIO, SPI, parallel IO, UART, I2C, I2S, RMT (TX/RX), pulse counter, LED PWM, USB serial/JTAG controller, MCPWM, CAN FD controller, SDIO slave controller, bit regulator, event task matrix, ADC, temperature sensor, undervoltage monitor, analog voltage comparator, system timer, general-purpose timer, RTC timer, watchdog timer, etc.

4) **MicroSD Card Slot**

Connected to the ESP32-C5 using SPI communication, supporting MicroSD cards of various capacities.

5) **RGB Tri-color LED**

Uses an RGB tri-color LED with a built-in IC. Only one GPIO is needed to input different timing waveforms to control the LED color and frequency.

6) **Serial Port**

External serial port module for module serial communication.

7) **USB-to-Serial and One-click Download Circuit**

The core component is CH340C. One end connects to the computer USB, and the other end connects to the ESP32 serial port, implementing USB to-TTL serial conversion. Additionally, a one-click download circuit is included, which automatically enters download mode when downloading programs, without the need for external touch.

8) **BOOT Button**

After the display module is powered on, pressing the button pulls IO28 low. If IO28 is pulled low at the moment of power-on or during ESP32-C5 reset, the module enters download mode. In other cases, it can be used as a regular button.

9) **USB Interface**

Contains two circuits: USB analog serial port and USB-to-serial IC. Both USB interfaces can serve as the main power supply interface and program download interface for the display module. Both can connect to the ESP32-C5 for downloading and serial communication, and can also provide power.

10) **5V to 3.3V Voltage Regulator Circuit**

Contains two circuits: MCU power 5V-to-3.3V circuit and non-MCU power 5V-to-3.3V circuit. The MCU power 5V-to-3.3V circuit is dedicated to powering the MCU unit, while the non-MCU power 5V-to-3.3V circuit is dedicated to powering peripheral circuits other than the MCU. The core component of the circuit is the AMS1117-3.3 linear voltage regulator (LDO). This regulator chip has a maximum input voltage of 18V, a recommended maximum operating input voltage of 12V, a

fixed output of 3.3V, an output accuracy of ±2%, a maximum output current of 1A, and overcurrent limit and thermal shutdown protection. The LDO has a typical dropout voltage of 1.2V at 1A output current, can stably output 3.3V voltage, and its output capacity can fully meet the voltage and current requirements of the display module.

11) RESET Button

After the display module is powered on, pressing the button pulls the ESP32-C5 reset pin low (default state is pulled high), thereby implementing the reset function.

12) Extended Pins

Three idle IO pins of the ESP32-C5 chip, brought out for peripheral use.

13) Backlight Control Circuit

The core component is the BSS138 MOSFET. One end of this circuit connects to the backlight control pin on the ESP32 main controller, and the other end connects to the negative terminal of the LCD backlight LED. When the backlight control pin is pulled high, the backlight turns on; otherwise, it turns off.

14) Speaker Interface

The terminal block must be connected vertically. Used for connecting monaural speakers and loudspeakers.

15) Audio Amplifier Circuit

The core component is the FM8002E audio amplifier IC. One end of this circuit connects to the audio codec IC output interface, and the other end connects to the speaker interface. The function of this circuit is to drive small-power speakers or loudspeakers to produce sound.

16) I2C Peripheral Interface

A 4-wire horizontal interface. It brings out the I2C bus of the ESP32-C5 for connecting external I2C devices.

17) Extended FPC Interface

Ten multiplexed IO pins of the ESP32-C5 chip, including SPI, IIC, USB, and other protocol interfaces, brought out for peripheral use.

3.2. Display Module Schematic Detailed Explanation

1) Type-C Interface Circuit

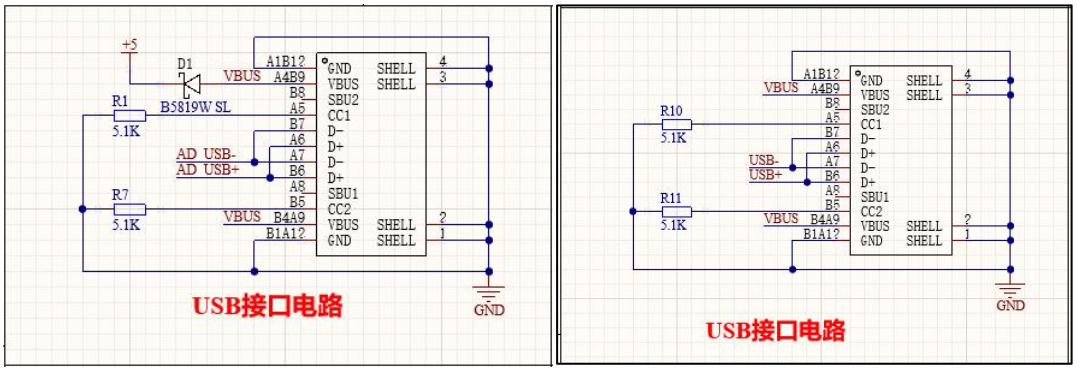

Figure 3.5 Type-C Interface Circuit

In this circuit, D1 is a Schottky diode used to prevent reverse current. R1, R7, R10, and R11 are pull-down resistors. USB is the Type-C receptacle. The display module receives power, downloads programs, and communicates serially through the USB Type-C connection. +5V and GND are the positive supply voltage and ground signal, USB- and USB+ are USB differential signals transmitted to the ESP32-C5 USB interface IO, and AD_USB- and AD_USB+ are USB differential signals transmitted to the USB-to-serial auto-download circuit.

2) MCU Power 5V to 3.3V Voltage Regulator Circuit / Non-MCU Power 5V to 3.3V Voltage Regulator Circuit

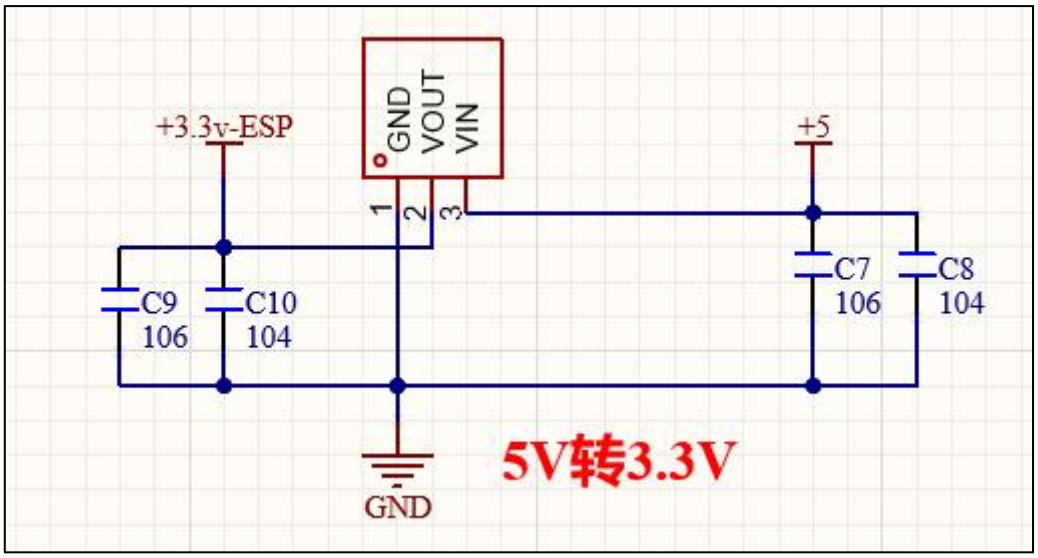

Figure 3.6 MCU Power 5V to 3.3V Voltage Regulator Circuit

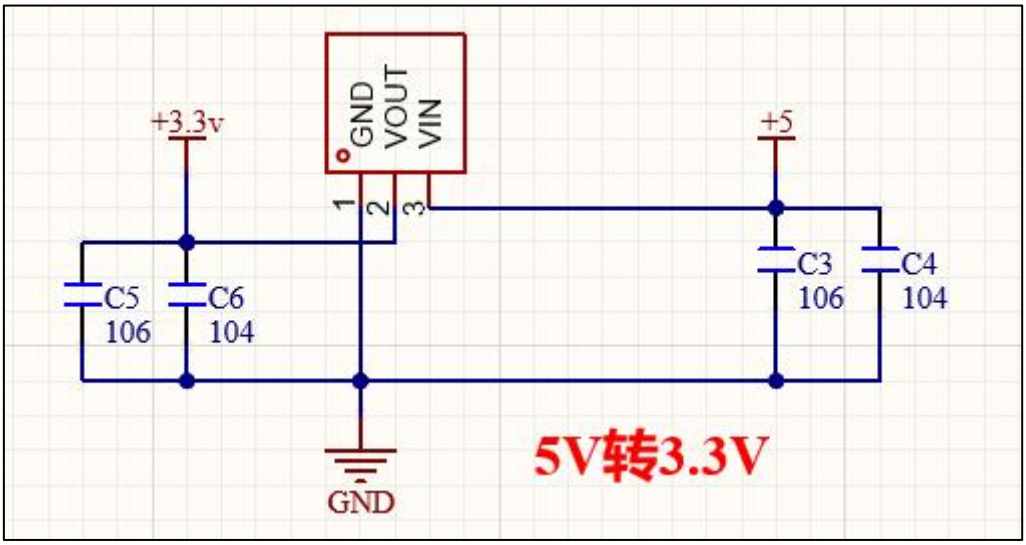

Figure 3.7 Non-MCU Power 5V to 3.3V Voltage Regulator Circuit

In this circuit, C3 through C10 are bypass filter capacitors used to maintain input and output voltage stability. U2 and U3 are 5V-to-3.3V LDOs, model AMS1117. Since most circuits on the display module require 3.3V power supply, while the Type-C interface input power is basically 5V, a voltage regulator conversion circuit is needed. The MCU power 5V-to-3.3V voltage regulator circuit supplies power to the MCU. The non-MCU power 5V-to-3.3V voltage regulator circuit supplies power to all circuits except the MCU.

3) Resistive Touchscreen Interface Circuit

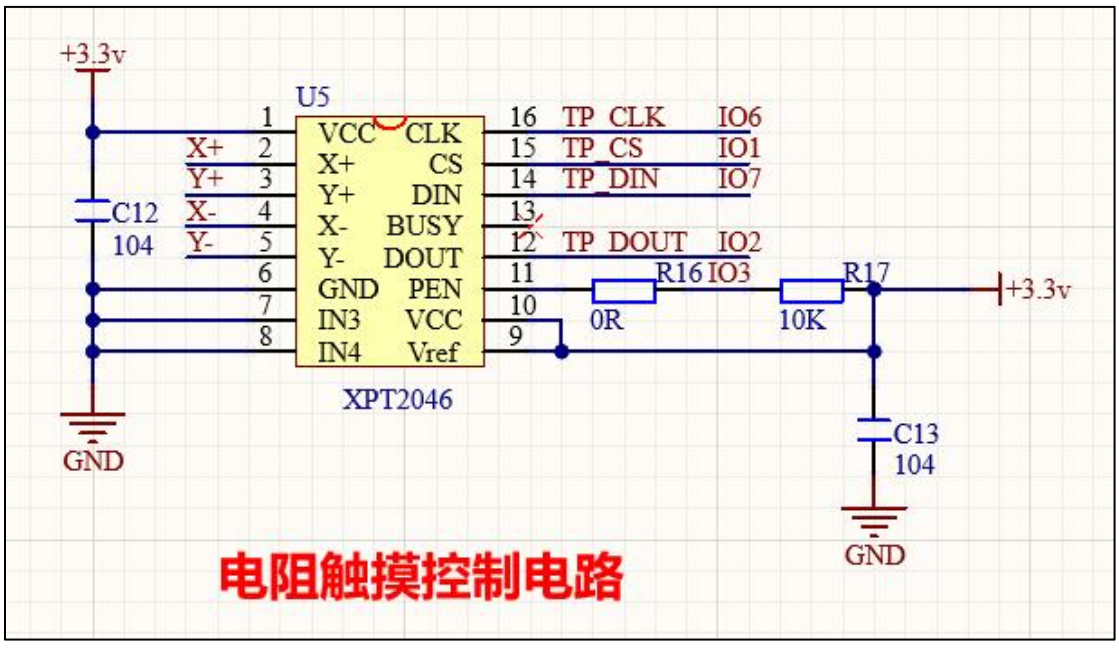

Figure 3.8 Resistive Touchscreen Interface Circuit

In this circuit, C12 and C13 are bypass filter capacitors used to maintain input voltage stability. R16 is a switchable resistor; if not soldered, interrupt IO3 is not connected; it is soldered by default. R17 is a pull-up resistor used to maintain the default high-level state of the pin. U5 is the XPT2046 controller IC. The function of this IC is to obtain the coordinate voltage value of the resistive touchscreen touch point through the four pins X+, X-, Y+, and Y-, then perform ADC conversion and transmit the ADC value to the ESP32 main controller. The ESP32 main controller then converts the ADC value into pixel coordinates of the display screen. The IO3 pin is the touch interrupt pin; it outputs a low level when a touch event occurs. Note: SPI bus is shared.

4) Audio Amplifier Circuit

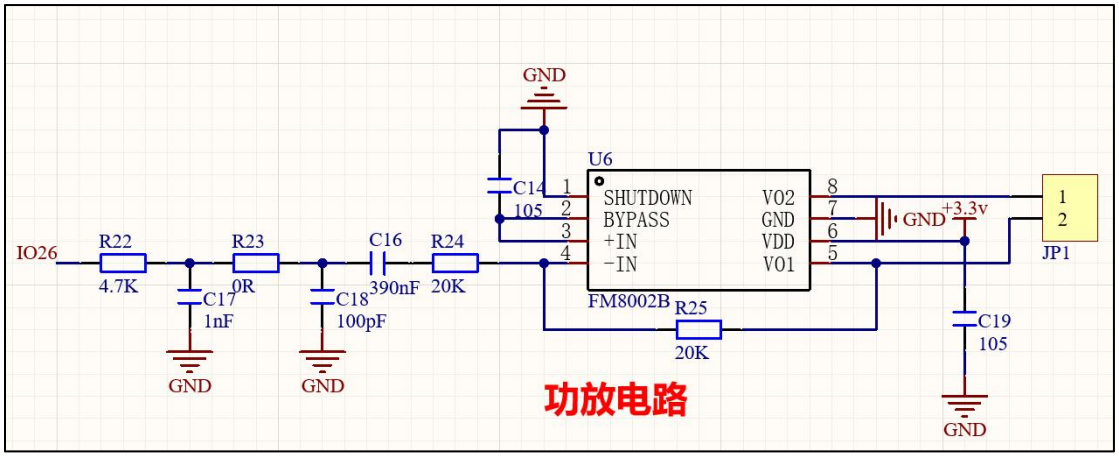

Figure 3.9 Audio Amplifier Circuit

In this circuit, R22 and C17 form a first-stage RC low-pass filter circuit, R23 and C18 form a second-stage RC low-pass filter circuit, and R24 and R25 are gain adjustment resistors for the FM8002B amplifier. The audio output uses a differential output method. When the internal parameters of the chip are fixed, simultaneously reducing the resistance values of R24 and R25 increases the circuit gain and the volume of the external speaker. C16 is the audio input DC-blocking coupling capacitor. C14 and C19 are power supply and output filter capacitors. R23 is a 0-ohm resistor used for signal path shorting. JP1 is the speaker/loudspeaker interface. U6 is the FM8002B monaural Class D audio amplifier IC. IO26 inputs a PWM audio signal, which passes through two-stage RC filtering and a DC-blocking capacitor, is amplified by the FM8002B, and then differentially output through the

VO1 and VO2 pins to drive the speaker. The SHUTDOWN pin is directly connected to ground, and the chip is permanently in the working enabled state.

5) ESP32-C5 Main Controller Circuit

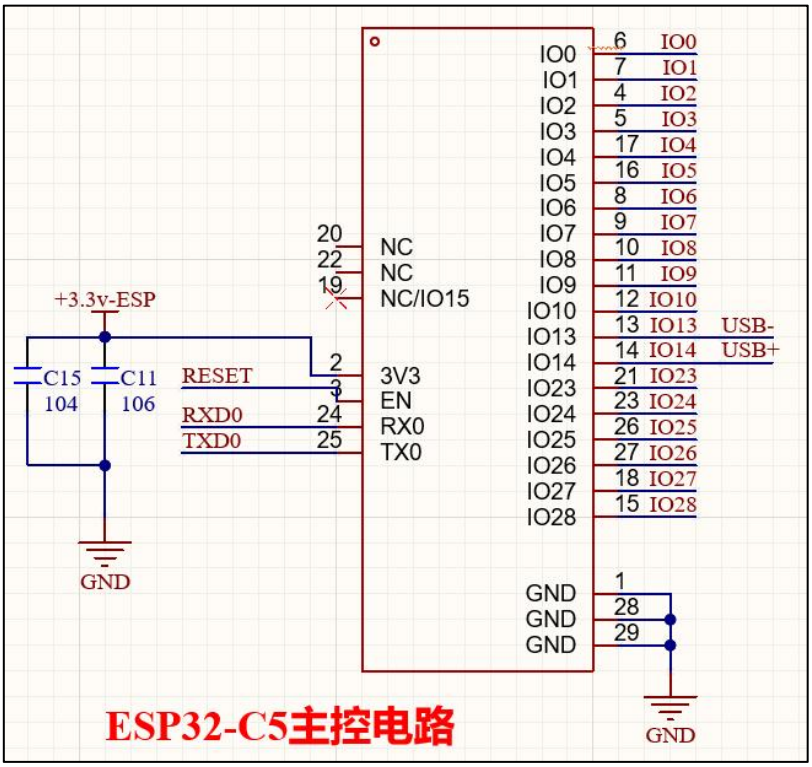

Figure 3.10 ESP32-C5 Main Controller Circuit

In this circuit, U4 is the ESP32-C5 module. C15 and C16 are power supply filters, and the rest are all externally routed pins.

6) Button Reset Circuit

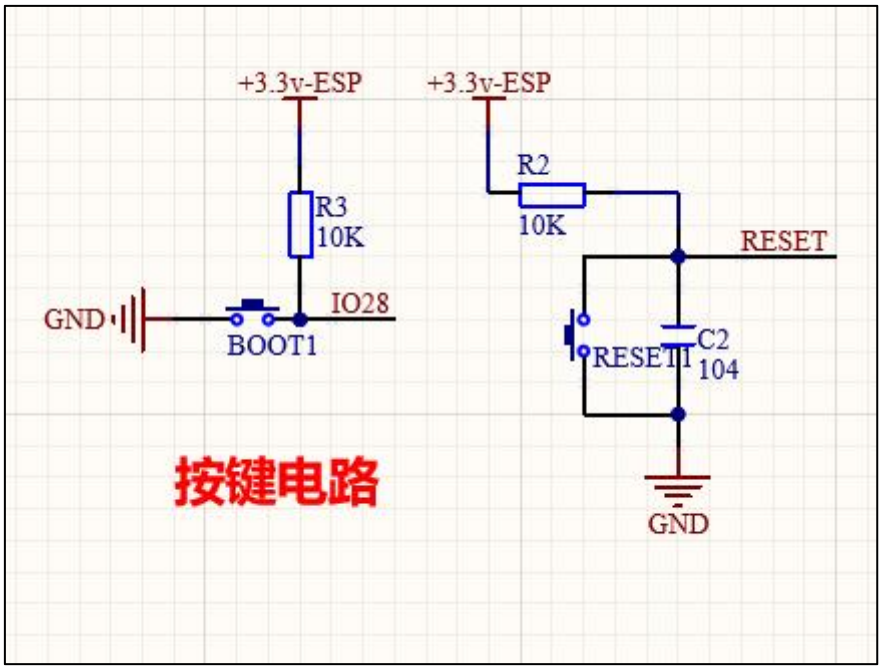

Figure 3.11 Button Reset Circuit

In the RESET circuit, RESET1 is the button, R2 is the pull-up resistor, and C2 is the delay capacitor. Reset principle:

A、After power-on, C2 charges. At this point C2 is equivalent to a short circuit, the RESET pin is grounded, and the ESP32-C5 enters the reset

state;

B、When C2 finishes charging, C2 is equivalent to an open circuit, the RESET pin is pulled high, the ESP32-C5 reset ends,

and it enters normal working state;

C. Press RESET1, the RESET pin is grounded, the ESP32-C5 enters the reset state, and C2 discharges through KEY1;

D. Release RESET1, C2 charges. At this point C2 is equivalent to a short circuit, the RESET pin is grounded, and the ESP32 is still in

the reset state. After C2 finishes charging, the RESET pin is pulled high, the ESP32-C5 reset ends, and it enters

normal working state;

If the reset is unsuccessful, the capacitance of C2 can be appropriately increased to extend the low-level duration of the RESET pin.

In the BOOT button circuit, BOOT1 is the button and R3 is the pull-up resistor.

The default state of IO28 is high level. When BOOT

is pressed, it becomes low level. By holding BOOT1 and powering on or resetting, the ESP32 will enter download mode. In other

cases, BOOT1 can be used as a regular button.

7) Serial Module Interface Circuit

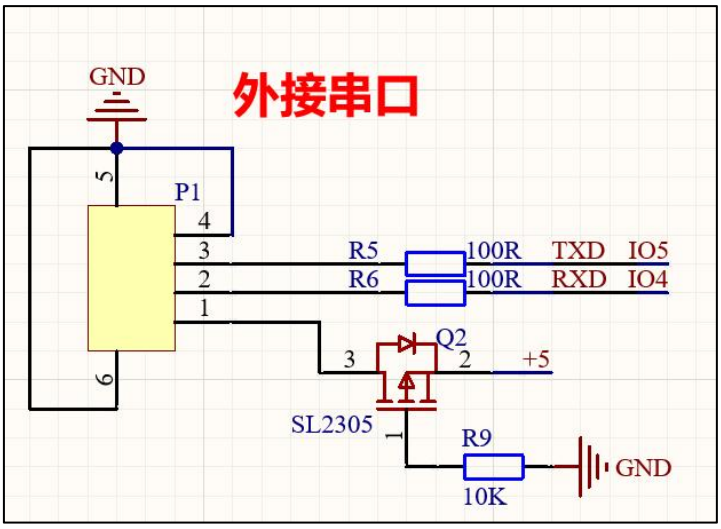

Figure 3.12 Serial Module Interface Circuit

In this circuit, P1 is a 4-pin 1.25mm pitch connector, R5 and R6 are impedance balancing resistors, and Q2 is a MOSFET that controls the 5V

input power. R9 is a pull-down resistor. RXD and TXD are connected to serial pins, and the other two

pins are for power supply. Q2 and R9 form a reverse-connection protection circuit.

8) Extended IO Circuit

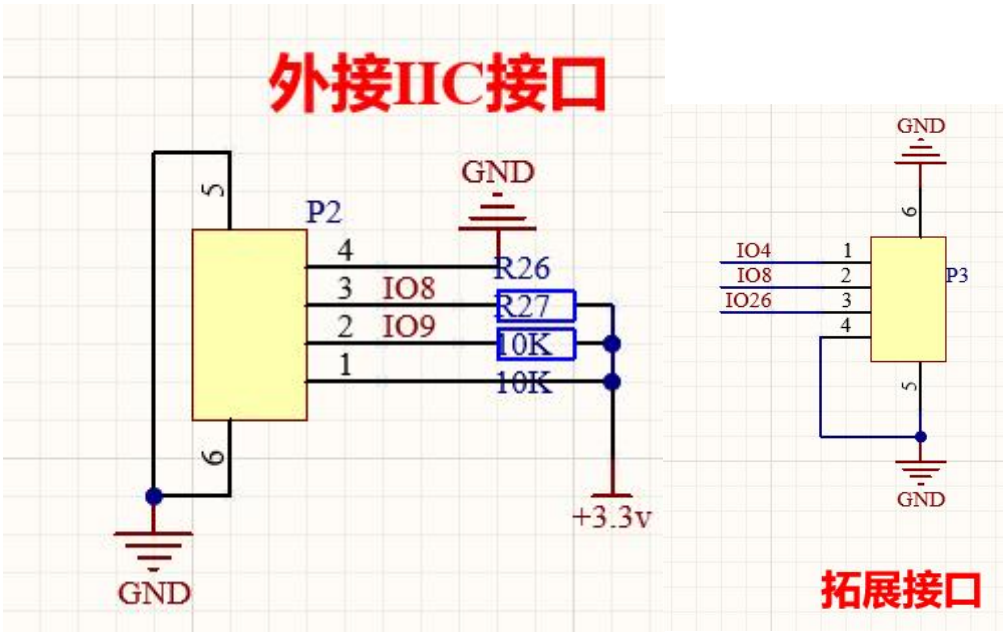

Figure 3.13 Extended IO Interface Circuit

In this circuit, P2 and P3 are 4-pin 1.25mm pitch connectors. IO4, IO8, and IO26 are three idle IO pins that can be connected to peripherals. IO8 and IO9 are externally expanded I2C interface pins that can connect to external I2C devices. Note: Due to the external pull-up resistors, they cannot be used as regular IO pins.

9) Extended FPC Circuit

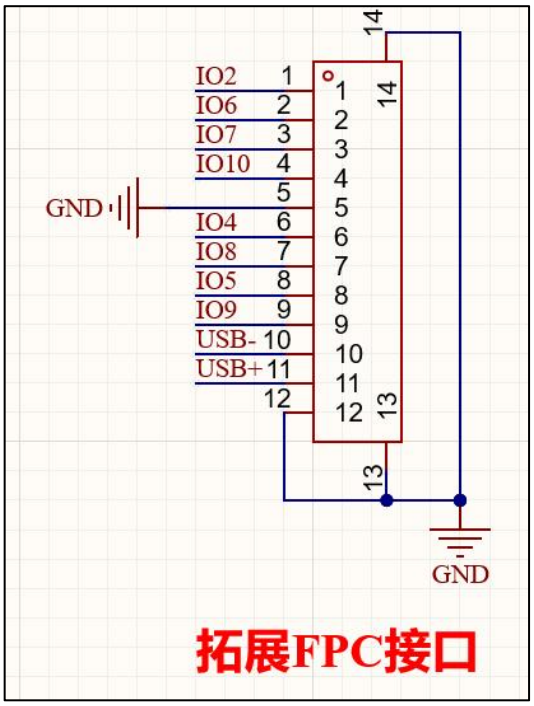

Figure 3.14 Extended FPC Circuit

In this circuit, FPC is a 12-pin FPC connector. It connects to multiplexed IO pins

including IO2, IO6, IO7, IO10, IO4, IO8, IO5, IO9, IO13, and IO14. Pay attention to the IO multiplexing situation when using.

10) 18P LCD Ribbon Cable Soldering Interface

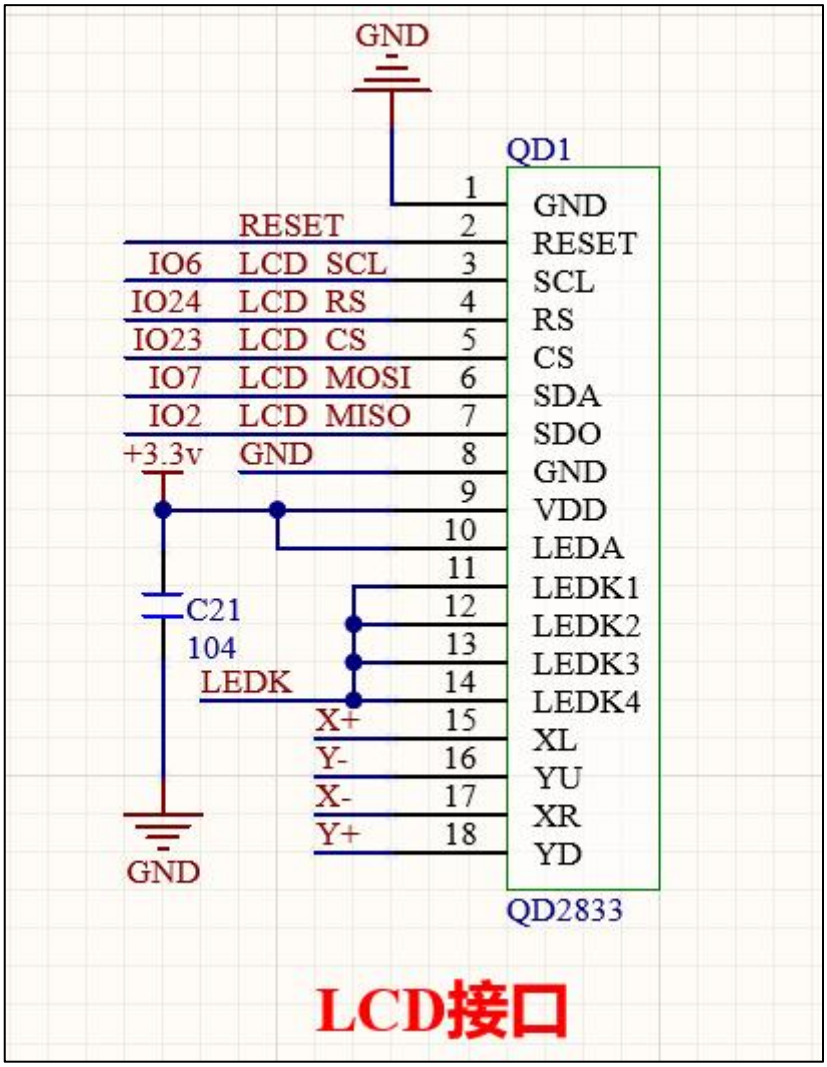

Figure 3.15 18P LCD Ribbon Cable Soldering Interface

In this circuit, C21 is a bypass filter capacitor, and QD1 is an 18-pin 0.8mm pitch LCD soldering interface. QD1 has

resistive touchscreen signal pins, LCD voltage pins, SPI communication pins, control pins, and backlight circuit

pins. The ESP32-C5 controls the LCD screen and touchscreen through these pins. Note: SPI bus is shared.

11) LCD Backlight Control Circuit

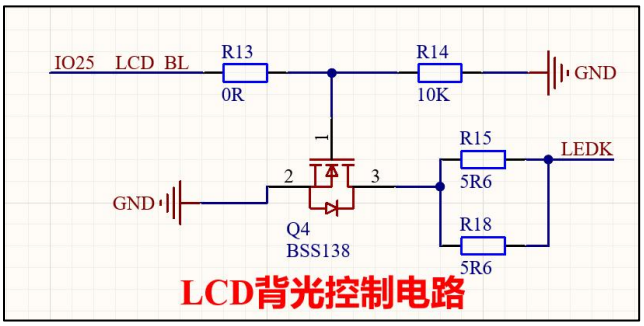

Figure 3.16 LCD Backlight Control Circuit

In this circuit, R13 is a debug resistor, temporarily reserved. Q4 is an N-channel MOSFET, R14 is the Q4 gate pull-down resistor, and R15 and R18 are backlight current-limiting resistors used in parallel. The LCD backlight LEDs are connected in parallel, with the positive terminal connected to 3.3V and the negative terminal connected to the drain of Q4. When the control pin LCD_BL outputs a high level, the drain and source of Q4 are conductive, the LCD backlight negative terminal is grounded, and the backlight LEDs conduct and emit light. When the control pin LCD_BL outputs a low level, the drain and source of Q4 are cut off, the LCD backlight negative terminal is floating, and the backlight LEDs do not conduct. In the default state, the LCD backlight is off. Reducing the resistance values of R15 and R18 can increase the maximum brightness of the backlight. Additionally, the LCD_BL pin can accept a PWM signal to adjust the LCD backlight brightness.

12) RGB Tri-color LED Control Circuit

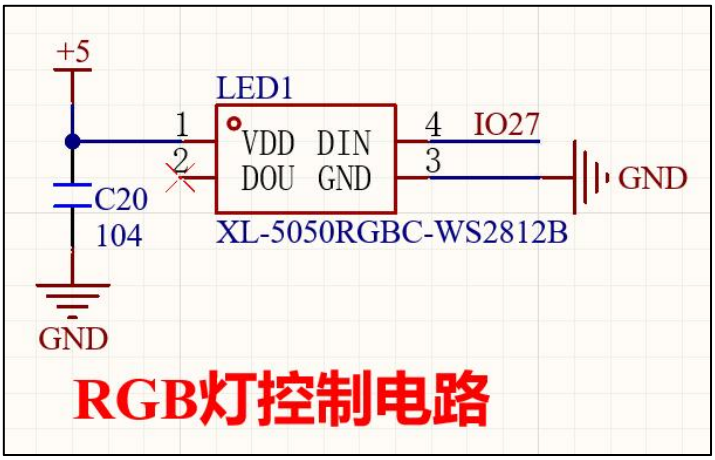

Figure 3.17 RGB Tri-color LED Control Circuit

In this circuit, LED1 is an RGB tri-color LED with a built-in control IC. It uses 5V power supply and requires only one

IO pin for control, saving IO resources. The principle is to drive different colored LEDs to emit light by inputting different timing waveforms through the RGB_LED signal.

13) MicroSD Card Slot Interface Circuit

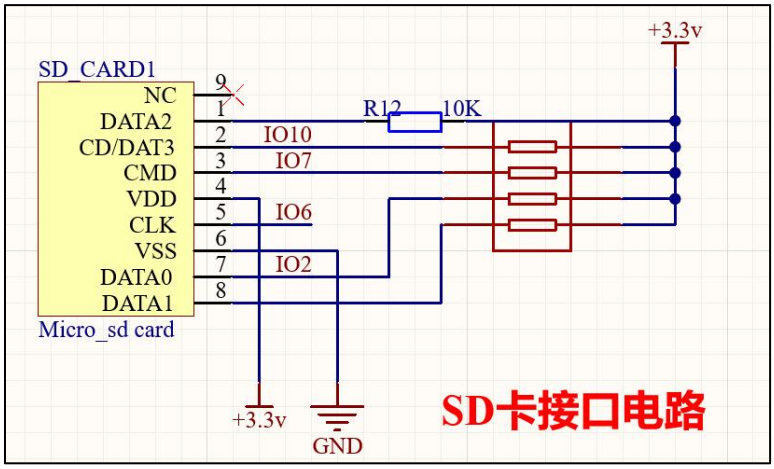

Figure 3.18 MicroSD Card Slot Interface Circuit

In this circuit, SD_CARD1 is the MicroSD card slot. R12 and the 10K resistor array are pull-up resistors for each pin. This interface circuit uses SPI communication. It supports high-speed storage MicroSD cards. Note: SPI bus is shared.

3.3. Display Module Usage Precautions

1) When connecting to the externally routed IO pins, pay attention to the IO usage to avoid mismatched connections with program code definitions.
2) Since the touchscreen, SD card, and display functions all use the same SPI bus, CS is used to distinguish devices. Be careful not to repeatedly initialize SPI in the code.
3) Both USB ports support serial flashing, but the flashing methods are different: The USB-UART port uses CH340 and relies on hardware circuitry to automatically reset and enter download mode; the USB port uses native USB-IO to complete flashing through the chip built-in USB protocol.
4) IO28 belongs to the ESP32-C5. If the code configures IO28 incorrectly, an abnormal power-on level will force entry into download mode, causing startup abnormalities.
5) Use this product safely and properly.