ESP32

Hardware Design Guidelines



Release master
Espressif Systems
Nov 09, 2026

Table of contents

Table of contents i

1 Latest Version of This Document 3
- 1.1 About This Document . . . 3
  - 1.1.1 Introduction . . . 3
  - 1.1.2 Latest Version of This Document . . . 3
- 1.2 Product Overview . . . 3
- 1.3 Schematic Checklist . . . 4
  - 1.3.1 Overview . . . 4
  - 1.3.2 Power Supply . . . 5
  - 1.3.3 Chip Power-up and Reset Timing . . . 7
  - 1.3.4 Flash and PSRAM . . . 8
  - 1.3.5 Clock Source . . . 9
  - 1.3.6 RF . . . 11
  - 1.3.7 UART . . . 13
  - 1.3.8 SPI . . . 13
  - 1.3.9 Strapping Pins . . . 13
  - 1.3.10 GPIO . . . 14
  - 1.3.11 ADC . . . 15
  - 1.3.12 External Capacitor . . . 16
  - 1.3.13 SDIO . . . 16
  - 1.3.14 Touch Sensor . . . 17
  - 1.3.15 Ethernet MAC . . . 18
- 1.4 PCB Layout Design . . . 18
  - 1.4.1 General Principles of PCB Layout for the Chip . . . 19
  - 1.4.2 Power Supply . . . 19
  - 1.4.3 Crystal . . . 22
  - 1.4.4 RF . . . 22
  - 1.4.5 Flash and PSRAM . . . 25
  - 1.4.6 External RC . . . 25
  - 1.4.7 UART . . . 25
  - 1.4.8 General Principles of PCB Layout for Modules (Positioning a Module on a Base Board) . . . 27
  - 1.4.9 SDIO . . . 27
  - 1.4.10 Touch Sensor . . . 28
  - 1.4.11 Typical Layout Problems and Solutions . . . 31
- 1.5 Download Guidelines . . . 31
- 1.6 Related Documentation and Resources . . . 32
  - 1.6.1 ESP32 Modules . . . 32
  - 1.6.2 ESP32 Development Boards . . . 32
  - 1.6.3 Other Related Documentation and Resources . . . 33
- 1.7 Glossary . . . 33
- 1.8 Revision History . . . 33
- 1.9 Disclaimer and Copyright Notice . . . 34
  - 1.9.1 Terms of Use for Reference Designs . . . 35

This document provides guidelines for the ESP32 SoC.



Schematic Checklist



PCB Layout Design



Download Guidelines



Resources

Chapter 1

Latest Version of This Document

Check the link to make sure that you use the latest version of this document: https://docs.espressif.com/projects/esp-hardware-design-guidelines/en/latest/esp32/index.html

1.1 About This Document

1.1.1 Introduction

The hardware design guidelines advise on how to integrate ESP32 into a product. These guidelines will help to achieve optimal performance of your product, ensuring technical accuracy and adherence to Espressif ' s standards. The guidelines are intended for hardware and application engineers.

The document assumes that you possess a certain level of familiarity with the ESP32 SoC. In case you lack prior knowledge, we recommend utilizing this document in conjunction with the ESP32 Series Datasheet.

1.1.2 Latest Version of This Document

Check the link to make sure that you use the latest version of this document: https://docs.espressif.com/projects/esp-hardware-design-guidelines/en/latest/esp32/index.html

1.2 Product Overview

ESP32 is a system on a chip that integrates the following features:

- Wi-Fi (2.4 GHz band)
- Bluetooth®
- Dual high-performance Xtensa® 32-bit LX6 CPU cores
- Ultra Low Power coprocessor
- Multiple peripherals

Powered by 40 nm technology, ESP32 provides a robust, highly-integrated platform, which helps meet the continuous demands for efficient power usage, compact design, security, high performance, and reliability. Typical application scenarios for ESP32 include:

- Smart Home

- Industrial Automation
- Health Care
- Consumer Electronics
- Smart Agriculture
- POS Machines
- Service Robot
- Audio Devices
- Generic Low-power IoT Sensor Hubs
- Generic Low-power IoT Data Loggers
- Cameras for Video Streaming
- Speech Recognition
- Image Recognition
- SDIO Wi-Fi + Bluetooth Networking Card
- Touch and Proximity Sensing

For more information about ESP32, please refer to ESP32 Series Datasheet.

Note: Unless otherwise specified, "ESP32" used in this document refers to the series of chips, instead of a specific chip variant.

1.3 Schematic Checklist

1.3.1 Overview

The integrated circuitry of ESP32 requires only 20 electrical components (resistors, capacitors, and inductors) and a crystal, as well as an SPI flash. The high integration of ESP32 allows for simple peripheral circuit design. This chapter details the schematic design of ESP32.

The following figure shows a reference schematic design of ESP32. It can be used as the basis of your schematic design.

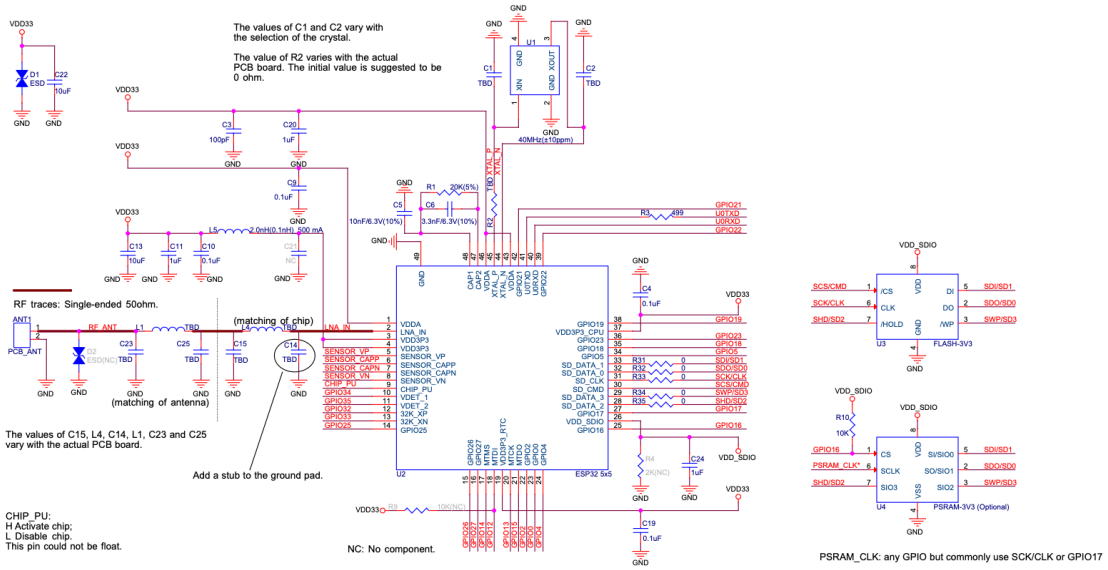

Fig. 1: ESP32 Reference Schematic

Note that Figure *ESP32 Reference Schematic* shows the connection method for quad 3.3 V external flash/PSRAM.

- The SCLK for PSRAM can be assigned to any available GPIO (configured by software). It is recommended to use GPIO17, or share the clock line with flash, i.e., SD_CLK.
- When using ESP32-D0WDR2-V3 (with in-package quad 3.3 V PSRAM), the connection method for external flash is as shown in *ESP32 Reference Schematic*. GPIO16 needs a pull-up resistor, with a typical value of 10 kΩ.
- When using a module with built-in ESP32-D0WDR2-V3, since VDD_SDIO is not led out, the pull-up on GPIO16 needs to be connected to an external 3.3 V power supply. This will result in additional power consumption. If low power consumption is required, use a pull-up resistor of 1 MΩ at the maximum.
- When using ESP32-U4WDH (with built-in quad 3.3 V flash), the internal flash connection is as shown in *ESP32 Schematic for Quad 3.3 V In-Package Flash*.
- When using in-package flash/PSRAM, there is no need to install resistors on the SPI traces, nor to pay special attention to the SPI traces.
- Whether R9 needs to be installed should be determined according to the ESP32 Chip Series Datasheet > Section *Internal LDO (VDD_SDIO) Voltage Control*.

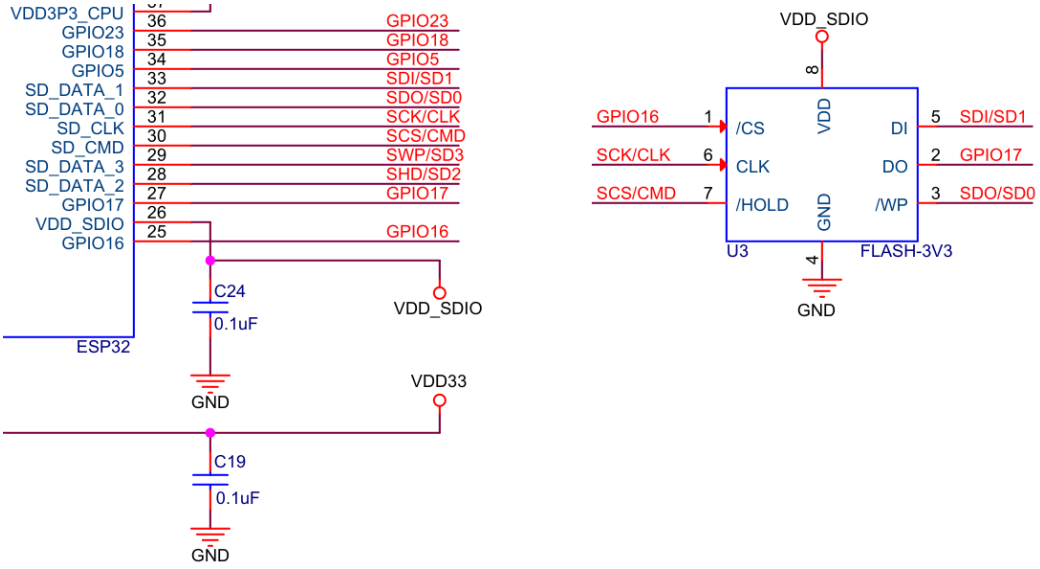

Fig. 2: ESP32 Schematic for Quad 3.3 V In-Package Flash

Any basic ESP32 circuit design may be broken down into the following major building blocks:

- *Power supply*
- *Chip power-up and reset timing*
- *Flash and PSRAM*
- *Clock source*
- *RF*
- *UART*
- *SPI*
- *Strapping pins*
- *External capacitor*
- *GPIO*
- *ADC*
- *SDIO*
- *Touch sensor*

The rest of this chapter details the specifics of circuit design for each of these sections.

1.3.2 Power Supply

The general recommendations for power supply design are:

- When using a single power supply, the recommended power supply voltage is 3.3 V and the output current is no less than 500 mA.
- It is suggested to add an ESD protection diode and at least 10 µF capacitor at the main power entrance (where the external power supply enters the PCB).

The power scheme is shown in Figure *ESP32 Power Scheme*.

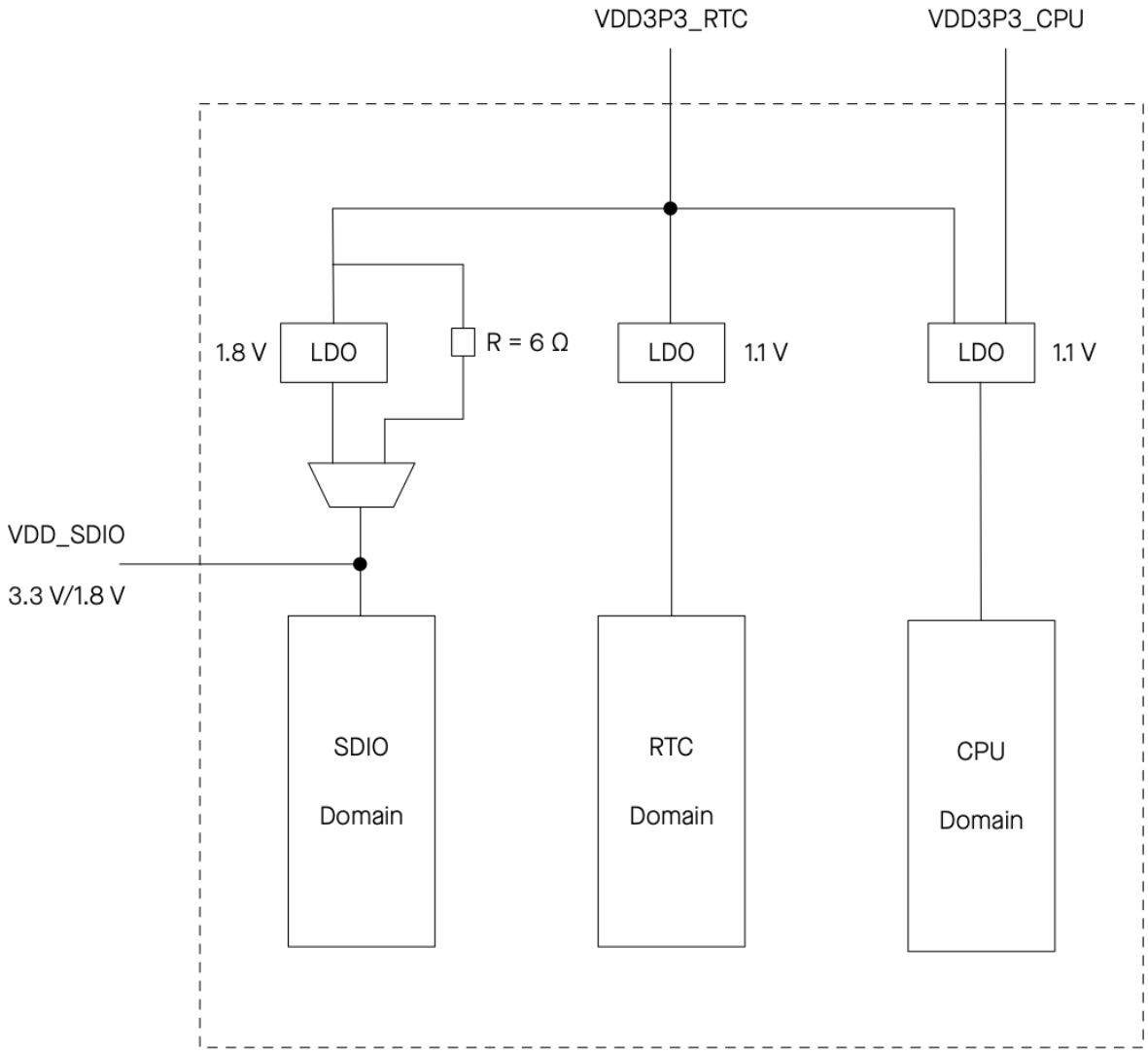

Fig. 3: ESP32 Power Scheme

More information about power supply pins can be found in ESP32 Series Datasheet > Section *Power Supply*.

Digital Power Supply

ESP32 has pin37 VDD3P3_CPU as the digital power supply pin, with an operating voltage range of 1.8 V ~ 3.6 V. ESP32 also has pin 20 VDD3P3_RTC as the RTC and partial digital power supply pin, with an operating voltage range of 2.3 V ~ 3.6 V. It is recommended to add a 0.1 µF capacitor close to the digital power supply pins in the circuit.

Pin VDD_SDIO serves as the power supply for the external device at either 1.8 V or 3.3 V (default).

- When VDD_SDIO is in 1.8 V mode, it is powered by ESP32' s internal LDO, which can provide a maximum current of 40 mA and an output voltage range of 1.65 V ~ 2.0 V. It is recommended to add a 2 kΩ pull-down resistor and a 4.7 µF capacitor to ground at the VDD_SDIO pin.

- When VDD_SDIO is in 3.3 V mode, it is powered by VDD3P3_RTC through an internal resistor of about 6 Ω. Therefore, there will be some voltage drop between VDD_SDIO and VDD3P3_RTC. It is recommended to add a 1 µF filter capacitor close to the VDD_SDIO pin.

Attention:

- When using VDDVDD_SDIO_SPI as the power supply pin for in-package or off-package 3.3 V flash/PSRAM, please ensure that VDD3P3_RTC remains above 3.0 V to meet the operating voltage requirements of the flash/PSRAM, considering the voltage drop mentioned earlier.
- Note that VDD3P3_RTC cannot supply power alone; all power supplies must be powered on at the same time.

Depending on the value of EFUSE_SDIO_FORCE, the VDD_SDIO voltage can be controlled in two ways, as Table *VDD_SDIO Voltage Control* shows.

Table 1: VDD_SDIO Voltage Control

EFUSE_	MTDI	EFUSE_	Voltage	VDD_SDIO Power Source
0	0	Ignored	3.3 V	VDD3P3_RTC via R_{SPI} (default)
0	1	Ignored	1.8 V	Flash Voltage Regulator
1	Ignored	0	1.8 V	Flash Voltage Regulator
1	Ignored	1	3.3 V	VDD3P3_RTC via R_{SPI}

VDD_SDIO can also be driven by an external power supply.

It is recommended to use the VDD_SDIO output to supply power to external or internal flash/PSRAM.

Analog Power Supply

ESP32's VDDA (pin 1/43/46) and VDD3P3 (pin 3/4) are the analog power supply pins, working at 2.3 V ~ 3.6 V.

For VDD3P3, when ESP32 is transmitting signals, there may be a sudden increase in the current draw, causing power rail collapse. Therefore, it is highly recommended to add a 10 µF capacitor to the power rail, which can work in conjunction with the 1 µF capacitor(s) or other capacitors.

It is suggested to add an extra 10 µF capacitor at the main power entrance. If the main power entrance is close to VDD3P3, then the two 10 µF capacitors can be merged into one.

Add an LC circuit to the VDD3P3 power rail to suppress high-frequency harmonics. The inductor's rated current is preferably 500 mA and above.

For the remaining capacitor circuits, please refer to *ESP32 Reference Schematic*.

1.3.3 Chip Power-up and Reset Timing

ESP32's CHIP_PU pin can enable the chip when it is high and reset the chip when it is low.

When ESP32 uses a 3.3 V system power supply, the power rails need some time to stabilize before CHIP_PU is pulled up and the chip is enabled. Therefore, CHIP_PU needs to be asserted high after the 3.3 V rails have been brought up.

To reset the chip, keep the reset voltage V_{IL_nRST} in the range of (NA ~ 0.6) V. To avoid reboots caused by external interferences, make the CHIP_PU trace as short as possible.

Figure *ESP32 Power-up and Reset Timing* shows the power-up and reset timing of ESP32.

Table *Description of Timing Parameters for Power-up and Reset* provides the specific timing requirements.

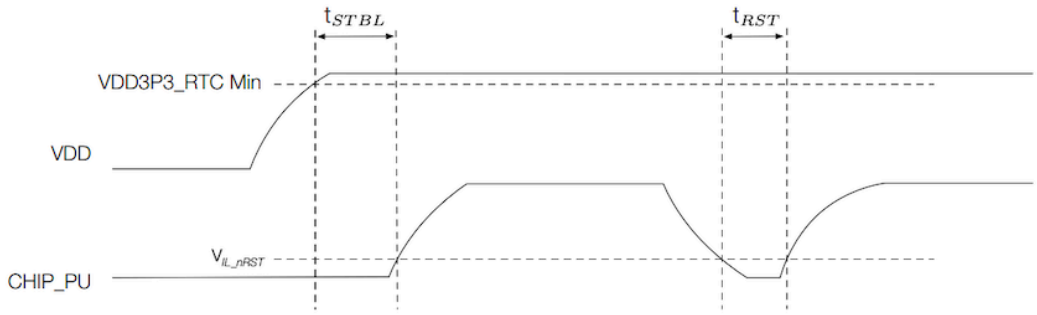

Fig. 4: ESP32 Power-up and Reset Timing

Table 2: Description of Timing Parameters for Power-up and Reset

Parameter	Description	Minimum (µs)
t_{STBL}	Time reserved for the power rails to stabilize before the CHIP_PU pin is pulled high to activate the chip	50
t_{RST}	Time reserved for CHIP_PU to stay below V_{IL_nRST} to reset the chip	50

Attention:

- CHIP_PU must not be left floating.
- To ensure the correct power-up and reset timing, it is advised to add an RC delay circuit at the CHIP_PU pin. The recommended setting for the RC delay circuit is usually R = 10 kΩ and C = 1 µF. However, specific parameters should be adjusted based on the characteristics of the actual power supply and the power-up and reset timing of the chip.
- If the user application has one of the following scenarios:
  - Slow power rise or fall, such as during battery charging.
  - Frequent power on/off operations.
  - Unstable power supply, such as in photovoltaic power generation.

  Then, the RC circuit alone may not meet the timing requirements, which may prevent the chip from entering normal operating mode or cause flash erase operations to occasionally fail to complete. In this case, please reserve a power monitor chip to reset the chip when the power supply is abnormal, and the threshold of the power monitor chip is recommended to be around 3.0 V.

1.3.4 Flash and PSRAM

ESP32 requires in-package or off-package flash to store application firmware and data. In-package PSRAM or off-package PSRAM is optional.

In-Package Flash and PSRAM

The tables list the pin-to-pin mapping between the chip and in-package flash/PSRAM. Please note that the following chip pins can connect at most one flash and one PSRAM. That is to say, when there is only flash in the package, the pin occupied by flash can only connect PSRAM and cannot be used for other functions; when there is only PSRAM, the pin occupied by PSRAM can only connect flash; when there are both flash and PSRAM, the pin occupied cannot connect any more flash or PSRAM.

Table 3: Pin-to-Pin Mapping Between Chip and In-Package Flash

ESP32-U4WDH	In-Package Flash (4 MB)
SD_DATA_1	IO0/DI
GPIO17	IO1/DO
SD_DATA_0	IO2/WP#
SD_CMD	IO3/HOLD#
SD_CLK	CLK
GPIO16	CS#
GND	VSS
VDD_SDIO	VDD

Table 4: Pin-to-Pin Mapping Between Chip and In-Package PSRAM

ESP32-D0WDR2-V3	In-Package PSRAM (2 MB)
SD_DATA_1	SIO0/SI
SD_DATA_0	SIO1/SO
SD_DATA_3	SIO2
SD_DATA_2	SIO3
SD_CLK	SCLK
GPIO16	CE#
GND	VSS
VDD_SDIO	VDD

Off-Package Flash and PSRAM

To reduce the risk of software compatibility issues, it is recommended to use flash and PSRAM models officially validated by Espressif. For detailed model selection, consult the sales or technical support team. If VDD_SDIO is used to supply power, make sure to select the appropriate off-package flash and RAM according to the power voltage on VDD_SDIO (1.8 V/3.3 V). It is recommended to add zero-ohm resistor footprints in series on the SPI communication lines. These footprints provide flexibility for future adjustments, such as tuning drive strength, mitigating RF interference, correcting signal timing, and reducing noise, if needed.

1.3.5 Clock Source

ESP32 supports two external clock sources:

- *External crystal clock source (Compulsory)*
- *RTC clock source (Optional)*

External Crystal Clock Source (Compulsory)

The ESP32 firmware only supports 40 MHz crystal.

The circuit for the crystal is shown in Figure *ESP32 Schematic for External Crystal*. Note that the accuracy of the selected crystal should be within ±10 ppm.

Please add a series inductor on the XTAL_P clock trace. Initially, it is suggested to use an inductor of 0 Ω to reduce the impact of high-frequency crystal harmonics on RF performance, and the value should be adjusted after an overall test.

The initial values of external capacitors C1 and C2 can be determined according to the formula:

$$C_L = \frac{C1 \times C2}{C1 + C2} + C_{stray}$$

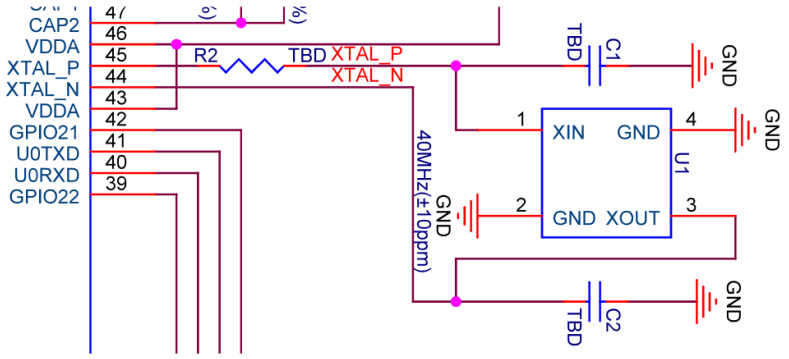

Fig. 5: ESP32 Schematic for External Crystal

where the value of C_L (load capacitance) can be found in the crystal's datasheet, and the value of C_{stray} refers to the PCB's stray capacitance. The values of C1 and C2 need to be further adjusted after an overall test as below:

1. Select TX tone mode using the Certification and Test Tool.
2. Observe the 2.4 GHz signal with a radio communication analyzer or a spectrum analyzer and demodulate it to obtain the actual frequency offset.
3. Adjust the frequency offset to be within ±10 ppm (recommended) by adjusting the external load capacitance.

- When the center frequency offset is positive, it means that the equivalent load capacitance is small, and the external load capacitance needs to be increased.
- When the center frequency offset is negative, it means the equivalent load capacitance is large, and the external load capacitance needs to be reduced.
- External load capacitance at the two sides are usually equal, but in special cases, they may have slightly different values.

Note:

- Defects in the manufacturing of crystal (for example, large frequency deviation of more than ±10 ppm, unstable performance within the operating temperature range, etc) may lead to the malfunction of ESP32, resulting in a decrease of the RF performance.
- It is recommended that the amplitude of the crystal is greater than 500 mV.
- When Wi-Fi or Bluetooth connection fails, after ruling out software problems, you may follow the steps mentioned above to ensure that the frequency offset meets the requirements by adjusting capacitors at the two sides of the crystal.

RTC Clock Source (Optional)

ESP32 supports an external 32.768 kHz crystal to act as the RTC clock. The external RTC clock source enhances timing accuracy and consequently decreases average power consumption, without impacting functionality.

Figure *ESP32 Schematic for 32.768 kHz Crystal* shows the schematic for the external 32.768 kHz crystal.

Please note the requirements for the 32.768 kHz crystal:

- Equivalent series resistance (ESR) ≤ 70 kΩ.
- Load capacitance at both ends should be configured according to the crystal's specification.

The parallel resistor R is used for biasing the crystal circuit (5 MΩ < R ≤ 10 MΩ).

For chip revisions v1.0 or v1.1, it is not recommended to use a 32.768 kHz crystal. For chip revisions v3.0 or higher, a 32.768 kHz crystal can be used, but the parallel resistor R must be installed.

When using AT firmware, the 32.768 kHz crystal is enabled by default. To prevent interference from floating crystal pins that may trigger the 32.768 kHz crystal, it is recommended to reserve grounding resistors at the 32K_XP and

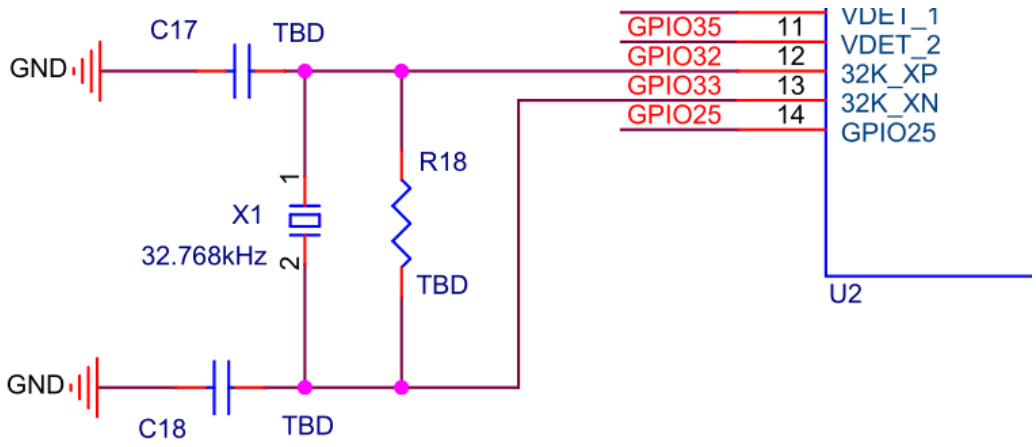

Fig. 6: ESP32 Schematic for 32.768 kHz Crystal

32K_XN pins, but do not install them.

If the RTC clock source is not required, then the pins for the 32.768 kHz crystal can be used as GPIOs.

1.3.6 RF

RF Circuit

ESP32 ' s RF circuit is mainly composed of three parts, the RF traces on the PCB board, the chip matching circuit, the antenna and the antenna matching circuit. Each part should meet the following requirements:

- For the RF traces on the PCB board, 50 Ω impedance control is required.
- For the chip matching circuit, it must be placed close to the chip. A CLC structure is preferred.
  - The CLC structure is mainly used to adjust the impedance point and suppress harmonics.
  - The RF matching circuit is shown in Figure *ESP32 Schematic for RF Matching*.
- For the antenna and the antenna matching circuit, to ensure radiation performance, the antenna ' s characteristic impedance must be around 50 Ω. Adding a CLC matching circuit near the antenna is recommended to adjust the antenna. However, if the available space is limited and the antenna impedance point can be guaranteed to be 50 Ω by simulation, then there is no need to add a matching circuit near the antenna.
- It is recommended to include ESD protection devices for the antenna to mitigate electrostatic interference.

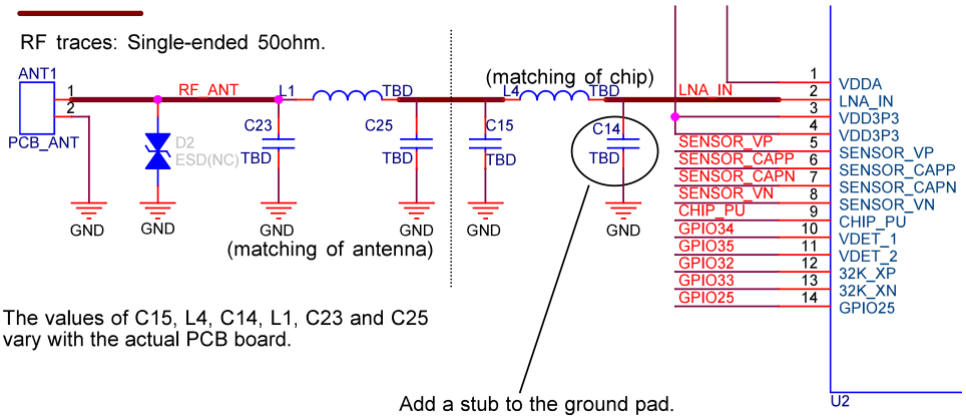

Fig. 7: ESP32 Schematic for RF Matching

RF Tuning

The RF matching parameters vary with the board, so the ones used in Espressif modules could not be applied directly. Follow the instructions below to do RF tuning.

Figure *ESP32 RF Tuning Diagram* shows the general process of RF tuning.

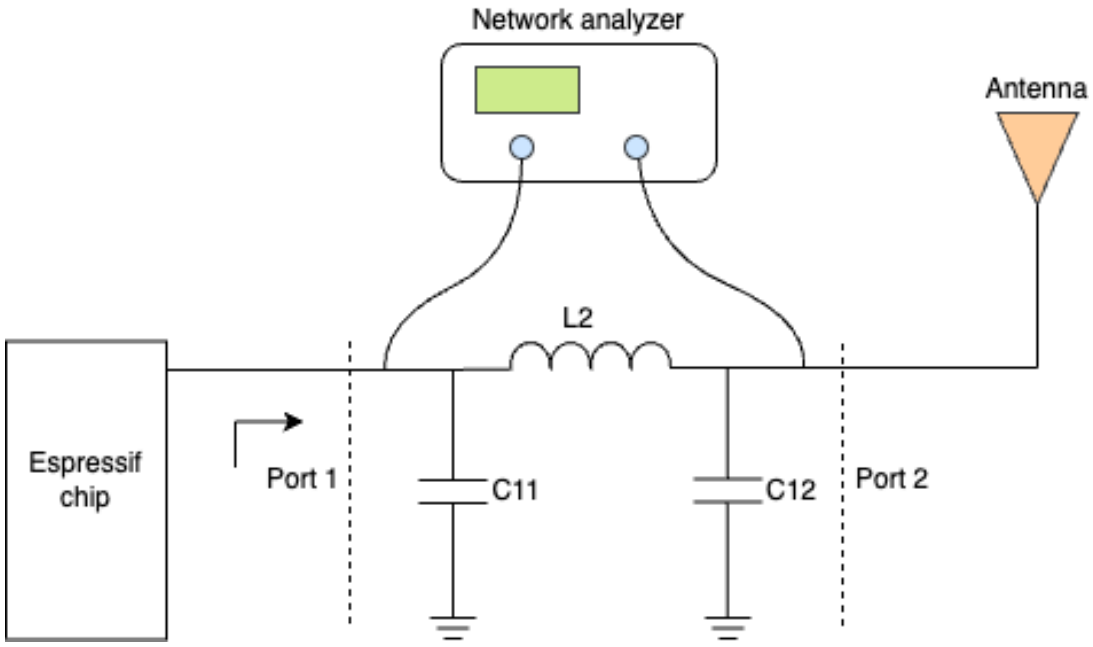

Fig. 8: ESP32 RF Tuning Diagram

In the matching circuit, define the port near the chip as Port 1 and the port near the antenna as Port 2. S11 describes the ratio of the signal power reflected back from Port 1 to the input signal power, the transmission performance is best if the matching impedance is conjugate to the chip impedance. S21 is used to describe the transmission loss of signal from Port 1 to Port 2. If S11 is close to the chip conjugate point 25+j0 and S21 is less than -35 dB at 4.8 GHz and 7.2 GHz, the matching circuit can satisfy transmission requirements.

Connect the two ends of the matching circuit to the network analyzer, and test its signal reflection parameter S11 and transmission parameter S21. Adjust the values of the components in the circuit until S11 and S21 meet the requirements. If your PCB design of the chip strictly follows the PCB design stated in Chapter *PCB Layout Design*, you can refer to the value ranges in Table *Recommended Value Ranges for Components* to debug the matching circuit.

Table 5: Recommended Value Ranges for Components

Reference Designator	Recommended Value Range	Serial No.
C11	1.2 ~ 1.8 pF	GRM0335C1H1RXBA01D
L2	2.0 ~ 3.0 nH	LQP03TN2NXB02D
C12	1.8 ~ 1.2 pF	GRM0335C1H1RXBA01D

Please use 0201 packages for RF matching components and add a stub to the first capacitor in the matching circuit at the chip end.

Note: If RF function is not required, it is recommended not to initialize the RF stack in firmware. In this case, the RF pin can be left floating. However, if RF function is enabled, make sure an antenna is connected. Operation without an antenna may result in unstable behavior or potential damage to the RF circuit.

1.3.7 UART

ESP32 includes 3 UART interfaces, UART0, UART1, and UART2. U0TXD and U0RXD are GPIO1 and GPIO3 by default. Other UART signals can be mapped to any available GPIOs by software configurations.

Usually, UART0 is used as the serial port for download and log printing. For instructions on download over UART0, please refer to Section *Download Guidelines*. It is recommended to connect a 499 Ω series resistor to the U0TXD line to suppress harmonics.

For application communication, use UART interfaces other than UART0 if possible. Add a series resistor on the TX line to suppress harmonics.

When using the AT firmware, please note that the UART GPIO is already configured (refer to Hardware Connection). It is recommended to use the default configuration.

1.3.8 SPI

When using the SPI function, to improve EMC performance, add a series resistor (or ferrite bead) and a capacitor to ground on the SPI_CLK trace. If space allows, it is recommended to also add a series resistor and capacitor to ground on other SPI traces. Ensure that the RC/LC components are placed close to the pins of the chip or module.

1.3.9 Strapping Pins

At each startup or reset, a chip requires some initial configuration parameters, such as in which boot mode to load the chip, etc. These parameters are passed over via the strapping pins. After reset, the strapping pins work as normal function pins.

GPIO0, GPIO2, GPIO5, MTDI, and MTDO are strapping pins.

All the information about strapping pins is covered in ESP32 Series Datasheet > Chapter *Boot Configurations*.

For strapping pin information related to VDD_SPI, please refer to Section *Digital Power Supply*.

In this section, we will mainly cover the strapping pins related to boot mode.

After chip reset is released, the combination of GPIO0 and GPIO2 controls the boot mode. See Table *Boot Mode Control*.

Table 6: Boot Mode Control

Boot Mode	GPIO0	GPIO2
Default Config	1	0
SPI Boot	1	Any value
Joint Download Boot[1]	0	0

Signals applied to the strapping pins should have specific *setup time* and *hold time*. For more information, see Figure *Setup and Hold Times for Strapping Pins* and Table *Description of Timing Parameters for Strapping Pins*.

Table 7: Description of Timing Parameters for Strapping Pins

Parameter	Description	Minimum (ms)
t_{SU}	Time reserved for the power rails to stabilize before the chip enable pin (CHIP_PU) is pulled high to activate the chip.	0
t_{H}	Time reserved for the chip to read the strapping pin values after CHIP_PU is already high and before these pins start operating as regular IO pins.	3

[1] Joint Download Boot mode supports the following download methods:
- UART Download Boot
- SDIO Download Boot

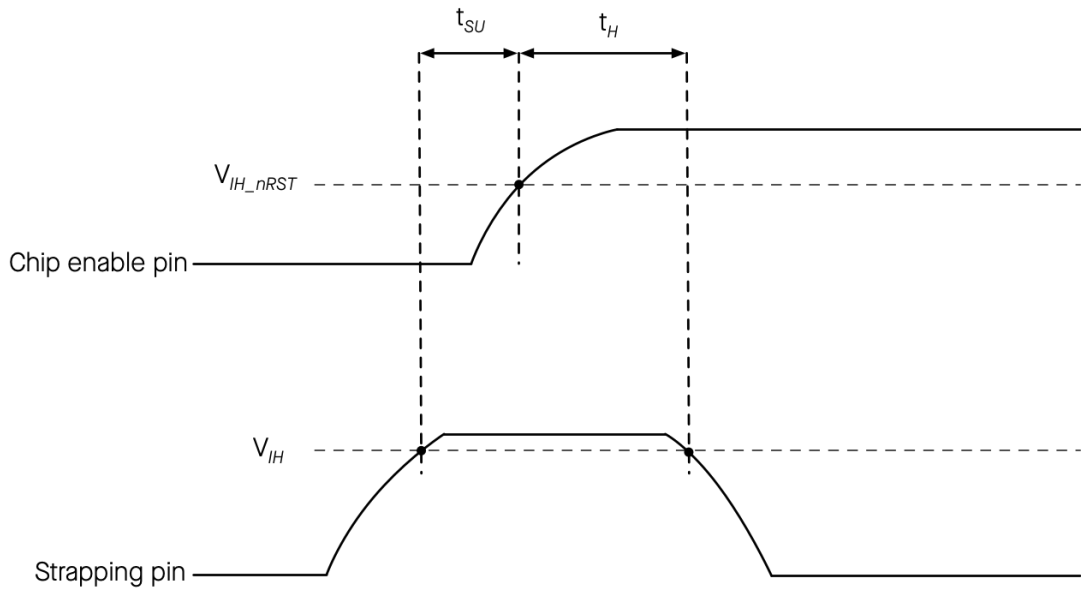

Fig. 9: Setup and Hold Times for Strapping Pins

Attention:

- It is recommended to place a pull-up resistor at the GPIO0 pin.
- Do not add high-value capacitors at GPIO0, or the chip may enter download mode.

1.3.10 GPIO

The pins of ESP32 can be configured via IO MUX or GPIO matrix. IO MUX provides the default pin configurations (see ESP32 Series Datasheet > Appendix *ESP32 Consolidated Pin Overview*), whereas the GPIO matrix is used to route signals from peripherals to GPIO pins. For more information about IO MUX and GPIO matrix, please refer to ESP32 Technical Reference Manual > Chapter *IO MUX and GPIO Matrix*.

Some peripheral signals have already been routed to certain GPIO pins, while some can be routed to any available GPIO pins. For details, please refer to ESP32 Series Datasheet > Section *Peripherals*.

When using GPIOs, please:

- Pay attention to the states of strapping pins during power-up.
- Pay attention to the default GPIO configurations after reset (see the table below). For unused pins in the high-impedance state without an internal pull-up or pull-down, it is recommended to add a pull-up or pull-down resistor or enable the internal pull during software initialization to avoid extra power consumption, selecting the direction as required by the external circuit.
- Avoid using the pins already occupied by flash/PSRAM.
- Note that GPIO34 and above are input-only IOs and do not have internal pull-up or pull-down resistors. If needed, please add appropriate external resistors.
- Only GPIOs in the VDD3P3_RTC power domain can be controlled in Deep-sleep mode.

Table 8: IO Pin Default Configuration

No.	Name	Power	At Reset	After Reset
1	VDDA			
2	LNA_IN	VDD3P3		

continues on next page

Table 8 – continued from previous page

No.	Name	Power	At Reset	After Reset
3	VDD3P3			
4	VDD3P3			
5	SENSOR_VP	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
6	SENSOR_CAPP	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
7	SENSOR_CAPN	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
8	SENSOR_VN	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
9	CHIP_PU	VDD3P3_RTC		
10	VDET_1	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
11	VDET_2	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
12	32K_XP	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
13	32K_XN	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
14	GPIO25	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
15	GPIO26	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
16	GPIO27	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=0
17	MTMS	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=1, wpu
18	MTDI	VDD3P3_RTC	oe=0, ie=1, wpd	oe=0, ie=1, wpd
19	VDD3P3_RTC	VDD3P3_RTC		
20	MTCK	VDD3P3_RTC	oe=0, ie=0	oe=0, ie=1, wpd
21	MTDO	VDD3P3_RTC	oe=0, ie=1, wpu	oe=0, ie=1, wpu
22	GPIO2	VDD3P3_RTC	oe=0, ie=1, wpd	oe=0, ie=1, wpd
23	GPIO0	VDD3P3_RTC	oe=0, ie=1, wpu	oe=0, ie=1, wpu
24	GPIO4	VDD3P3_RTC	oe=0, ie=1, wpd	oe=0, ie=1, wpd
25	GPIO16	VDD_SDIO	oe=0, ie=0	oe=0, ie=1
26	VDD_SDIO			
27	GPIO17	VDD_SDIO	oe=0, ie=0	oe=0, ie=1
28	SD_DATA_2	VDD_SDIO	oe=0, ie=1, wpu	oe=0, ie=1, wpu
29	SD_DATA_3	VDD_SDIO	oe=0, ie=1, wpu	oe=0, ie=1, wpu
30	SD_CMD	VDD_SDIO	oe=0, ie=1, wpu	oe=0, ie=1, wpu
31	SD_CLK	VDD_SDIO	oe=0, ie=1, wpu	oe=0, ie=1, wpu
32	SD_DATA_0	VDD_SDIO	oe=0, ie=1, wpu	oe=0, ie=1, wpu
33	SD_DATA_1	VDD_SDIO	oe=0, ie=1, wpu	oe=0, ie=1, wpu
34	GPIO5	VDD3P3_CPU	oe=0, ie=1, wpd	oe=0, ie=1, wpd
35	GPIO18	VDD3P3_CPU	oe=0, ie=0	oe=0, ie=1
36	GPIO23	VDD3P3_CPU	oe=0, ie=0	oe=0, ie=1
37	VDD3P3_CPU			
38	GPIO19	VDD3P3_CPU	oe=0, ie=0	oe=0, ie=1
39	GPIO22	VDD3P3_CPU	oe=0, ie=0	oe=0, ie=1
40	U0RXD	VDD3P3_CPU	oe=0, ie=1, wpu	oe=0, ie=1, wpu
41	U0TXD	VDD3P3_CPU	oe=0, ie=1, wpu	oe=0, ie=1, wpu
42	GPIO21	VDD3P3_CPU	oe=0, ie=0	oe=0, ie=1
43	VDDA			
44	XTAL_N	VDDA		
45	XTAL_P	VDDA		
46	VDDA			
47	CAP2	VDDA		
48	CAP1	VDDA		

- ie –input enable
- oe –output enable
- wpu –internal weak pull-up resistor enable
- wpd –internal weak pull-down resistor enable

1.3.11 ADC

Table below shows the correspondence between ADC channels and GPIOs.

Table 9: ADC Function

GPIO Pin Name	ADC Function
SENSOR_VP	ADC1_CH0
SENSOR_CAPP	ADC1_CH1
SENSOR_CAPN	ADC1_CH2
SENSOR_VN	ADC1_CH3
32K_XP	ADC1_CH4
32K_XN	ADC1_CH5
VDET_1	ADC1_CH6
VDET_2	ADC1_CH7
GPIO4	ADC2_CH0
GPIO0	ADC2_CH1
GPIO2	ADC2_CH2
MTDO	ADC2_CH3
MTCK	ADC2_CH4
MTDI	ADC2_CH5
MTMS	ADC2_CH6
GPIO27	ADC2_CH7
GPIO25	ADC2_CH8
GPIO26	ADC2_CH9

Please add a 0.1 µF filter capacitor between ESP pins and ground when using the ADC function to improve accuracy.

When RTC peripherals (SAR ADC1/SAR ADC2/AMP) is powered on, the inputs of GPIO36 (SENSOR_VP) and GPIO39 (SENSOR_VN) will be pulled down for approximately 80 ns. If SENSOR_VP and SENSOR_VN are used as GPIOs in the design, while ADC is supported by other pins, then software should disregard the glitch. Optionally, use SENSOR_VP and SENSOR_VN as ADC pins.

ADC1 is recommended over ADC2 as the latter cannot be used when Wi-Fi function is enabled.

The calibrated ADC results after hardware calibration and software calibration are shown in the list below. For higher accuracy, you may implement your own calibration methods.

- When ATTEN=0 and the effective measurement range is 100 ~ 950 mV, the total error is ±23 mV.
- When ATTEN=1 and the effective measurement range is 100 ~ 1250 mV, the total error is ±30 mV.
- When ATTEN=2 and the effective measurement range is 150 ~ 1750 mV, the total error is ±40 mV.
- When ATTEN=3 and the effective measurement range is 150 ~ 2450 mV, the total error is ±60 mV.

1.3.12 External Capacitor

Figure *ESP32 Schematic for External Capacitor* shows the schematic of components connected to pin47 CAP2 and pin48 CAP1.

C5 (10 nF) that connects to CAP1 should be of 10% tolerance and is required for proper operation of ESP32.

RC circuit between CAP1 and CAP2 pins may be omitted under certain conditions. This circuit is used when entering Deep-sleep mode. During this process, to minimize power consumption, the voltage to power ESP32 internals is dropped from 1.1 V to around 0.7 V. The RC circuit is used to minimize the period of the voltage drop. If removed, this process will take longer and the power consumption in Deep-sleep will be higher. If particular application of ESP32 is not using Deep-sleep mode, or power consumption is less critical, then this circuit is not required.

1.3.13 SDIO

ESP32 supports an SD/SDIO/MMC host controller and an SD/SPI slave controller, compliant with the SDIO 2.0 specification. For the SDIO host/slave controller, ESP32 provides two sets of slot GPIOs for use. Since the GPIOs

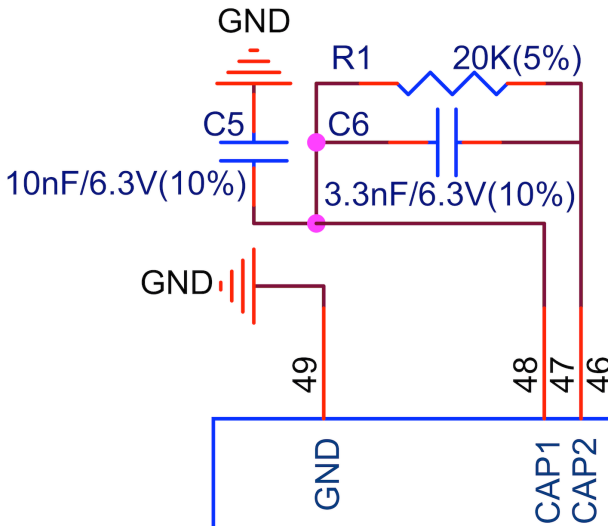

Fig. 10: ESP32 Schematic for External Capacitor

in slot0 are by default connected to flash, please connect the GPIOs and signal lines according to slot1 as shown in Table *SDIO Pin Configuration*.

Table 10: SDIO Pin Configuration

	CMD	CLK	DAT0	DAT1	DAT2	DAT3	Note
Slot0	GPIO11	GPIO6	GPIO7	GPIO8	GPIO9	GPIO10	Used to connect flash by default. Not recommended for other use.
Slot1	GPIO15	GPIO14	GPIO2	GPIO4	GPIO12	GPIO13	Multiplexed with JTAG, touch, EMAC, and strapping functions.

When connecting GPIOs in slot1 to signal lines, please note:

- When ESP32 works as an SDIO host, it is recommended to add pull-up resistors on the used pins. Unused pins can be utilized for other purposes.
- When ESP32 works as an SDIO slave, add pull-up resistors on all pins, regardless of whether these pins are used for SDIO or not. Unused pins cannot be used for other purposes.
- Since there are strapping pins among the SDIO GPIOs, enabling pull-up resistors for SDIO may cause some conflicts. For specific solutions, please refer to SD Pull-up Requirements.

For more information on SDIO configuration, please refer to API References.

1.3.14 Touch Sensor

ESP32 has 10 capacitive-sensing GPIOs, which detect variations induced by touching or approaching the GPIOs with a finger or other objects. The low-noise nature of the design and the high sensitivity of the circuit allow relatively small pads to be used. Arrays of pads can also be used, so that a larger area or more points can be detected.

Attention: ESP32 touch sensor has not passed the Conducted Susceptibility (CS) test for now, and thus has limited application scenarios.

Table below shows the correspondence between touch sensor channels and GPIOs.

Table 11: Touch Sensor Functions

GPIO Pin Name	Touch Sensor Function
GPIO4	TOUCH00
GPIO0	TOUCH1
GPIO2	TOUCH2
MTDO	TOUCH3
MTCK	TOUCH4
MTDI	TOUCH5
MTMS	TOUCH6
GPIO27	TOUCH7
32K_XN	TOUCH8
32K_XP	TOUCH9

When using the touch function, it is recommended to populate a series resistor at the chip side to reduce the coupling noise and interference on the line, and to strengthen the ESD protection. The recommended resistance is from 470 Ω to 2 kΩ, preferably 510 Ω. The specific value depends on the actual test results of the product.

1.3.15 Ethernet MAC

ESP32 provides a media access control (MAC) interface that complies with the IEEE-802.3-2008 standard for Ethernet communication. The ESP32-Ethernet-Kit board only supports the Reduced Media-Independent Interface (RMII). The allocation of the ESP32 pins to the RMII interface is shown in the table below.

Table 12: Ethernet MAC

Pin Name	Function 6	RMII (int_osc)	RMII (ext_osc)
GPIO0	EMAC_TX_CLK	CLK_OUT(O)	EXT_OSC_CLK(I)
GPIO21	EMAC_TX_EN	TX_EN(O)	TX_EN(O)
GPIO19	EMAC_TXD0	TXD[0](O)	TXD[0](O)
GPIO22	EMAC_TXD1	TXD[1](O)	TXD[1](O)
GPIO27	EMAC_RX_DV	CRS_DV(I)	CRS_DV(I)
GPIO25	EMAC_RXD0	RXD[0](I)	RXD[0](I)
GPIO26	EMAC_RXD1	RXD[1](I)	RXD[1](I)
GPIO16	EMAC_CLK_OUT	CLK_OUT(O)	–
GPIO17	EMAC_CLK_OUT_180	CLK_OUT_180(O)	–
Any GPIO	–	MDC(O)	MDC(O)
Any GPIO	–	MDIO(IO)	MDIO(IO)

For an Ethernet solution, it is recommended to use GPIO0 as the clock input. Be aware that GPIO0 also acts as a strapping pin and it can be affected by the clock during chip power-up and may enter download mode as a result. Therefore, make sure you have turned off the clock output on the PHY side before powering up the chip. The ESP32-Ethernet-Kit board uses GPIO to control PHY's reset pin and turn off the clock output. However, not all PHYs support this design, so it is crucial to verify functionality during testing. If this design cannot be implemented, consider alternative methods to ensure that GPIO0 remains unaffected during power-up. For a reference design please see ESP32-Ethernet-Kit User Guide.

If you need to use Wi-Fi and Ethernet simultaneously, do not use the internal APLL clock to generate the RMII clock, as this can lead to clock instability. Instead, use the RMII clock from the PHY side or an external clock source. For further details, please refer to ESP32-Ethernet-Kit User Guide > RMII Clock Selection.

1.4 PCB Layout Design

This chapter introduces the key points of how to design an ESP32 PCB layout using an ESP32 module (see Figure *ESP32 Reference PCB Layout*) as an example.

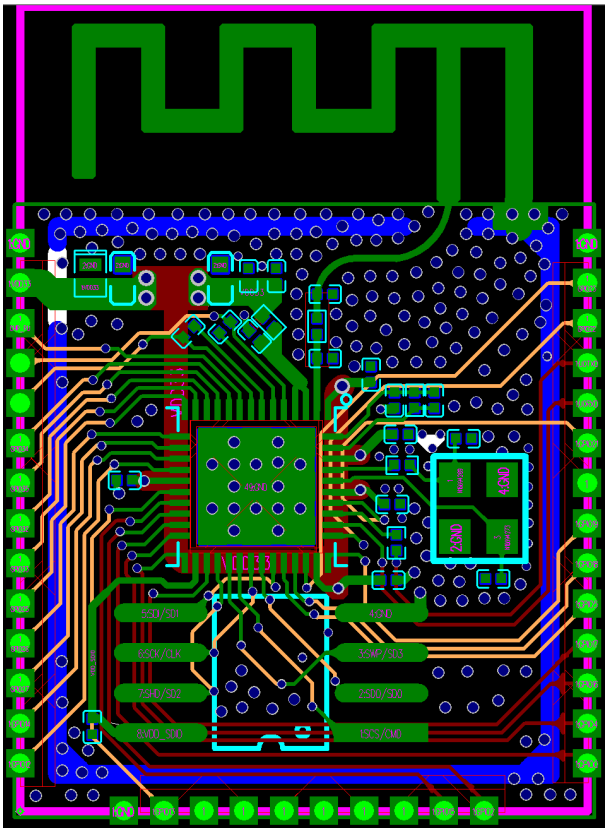

Fig. 11: ESP32 Reference PCB Layout

1.4.1 General Principles of PCB Layout for the Chip

It is recommended to use a four-layer PCB design:

- Layer 1 (TOP): Signal traces and components.
- Layer 2 (GND): No signal traces here to ensure a complete GND plane.
- Layer 3 (POWER): GND plane should be applied to better isolate the RF and crystal. Route power traces and a few signal traces on this layer, provided that there is a complete GND plane under the RF and crystal.
- Layer 4 (BOTTOM): Route a few signal traces here. It is not recommended to place any components on this layer.

A two-layer PCB design can also be used:

- Layer 1 (TOP): Signal traces and components.
- Layer 2 (BOTTOM): Do not place any components on this layer and keep traces to a minimum. Please make sure there is a complete GND plane for the chip, RF, and crystal.

1.4.2 Power Supply

Four-Layer PCB Design

Figure *ESP32 Power Traces in a Four-Layer PCB Design* shows the power traces in a four-layer PCB design.

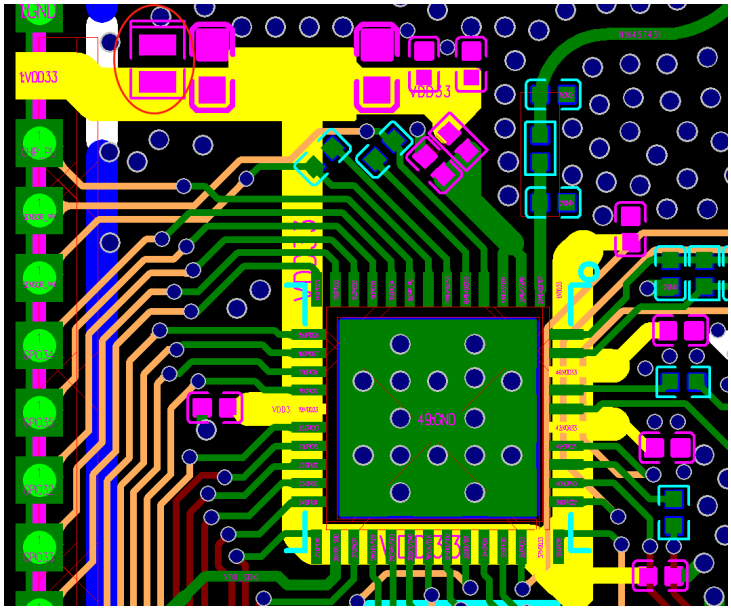

Fig. 12: ESP32 Power Traces in a Four-Layer PCB Design

- A four-layer PCB design is recommended. Whenever possible, route the power traces on the inner layers (not the ground layer) and connect them to the chip pins through vias. There should be at least two vias if the main power traces need to cross layers. The drill diameter on other power traces should be no smaller than the width of the power traces.
- The yellow highlighted traces in Figure *ESP32 Power Traces in a Four-Layer PCB Design* are the 3.3 V power traces. The width of the main power traces should be no less than 25 mil. The width of VDD3P3 power traces should be no less than 20 mil. The recommended width of other power traces is 12 ~ 15 mil. Ensure the power traces are surrounded by ground copper.
- The red circles in *ESP32 Power Traces in a Four-Layer PCB Design* show ESD protection diodes. Place them close to the power input. Add a 10 µF capacitor before the power trace enters the chip. You can also add a 0.1 µF or 1 µF capacitor in parallel. After that, the power trace can branch out in a star-shaped layout to reduce coupling between different power pins.
- The power supply for pin 3 and pin 4 is RF related, so please place a 10 µF capacitor for each pin. You can also add a 0.1 µF or 1 µF capacitor in parallel.
- Add a CLC/LC filter circuit near pin 3 and pin 4 to suppress high-frequency harmonics. The power trace can be routed at a 45-degree angle to maintain distance from adjacent RF traces. Except for the 10 µF capacitor, it is recommended to use 0201 components. This allows the filter circuit for pin 3 and pin 4 to be placed closer to the pins, with a GND isolation layer separating them from surrounding RF and GPIO traces, while also maximizing the placement of ground vias. Using 0201 components enables placing a via to the bottom layer at the first capacitor near the chip, while maintaining a keep-out area on other layers, further reducing harmonic interference. See Figure *ESP32 Power Stub*.

- In Figure *ESP32 Power Traces in a Four-Layer PCB Design*, the 10 µF capacitor is shared by the analog power supply VDD3P3, and the power entrance since the analog power is close to the chip power entrance. If the chip power entrance is not near VDD3P3, it is recommended to add a 10 µF capacitor to both the chip power entrance and VDD3P3.
- Place appropriate decoupling capacitors at the rest of the power pins. Ground vias should be added close to the capacitor's ground pad to ensure a short return path.
- The ground pad at the bottom of the chip should be connected to the ground plane through at least nine ground vias.
- The ground pads of the chip and surrounding circuit components should make full contact with the ground copper pour rather than being connected via traces.
- If you need to add a thermal pad EPAD under the chip on the bottom of the module, it is recommended to employ a square grid on the EPAD, cover the gaps with solder paste, and place ground vias in the gaps, as shown in Figure *ESP32 Power Traces in a Four-Layer PCB Design*. This helps effectively reduce solder leakage

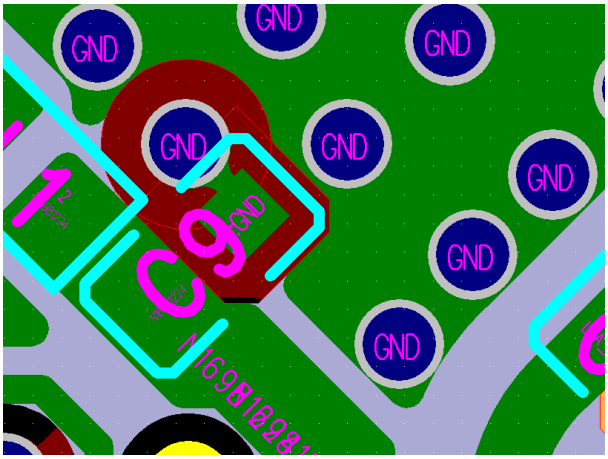

Fig. 13: ESP32 Power Stub

issues when soldering the module EPAD to the substrate.

Two-Layer PCB Design

Figure *ESP32 Power Traces in a Two-Layer PCB Design* shows the power traces in a two-layer PCB design.

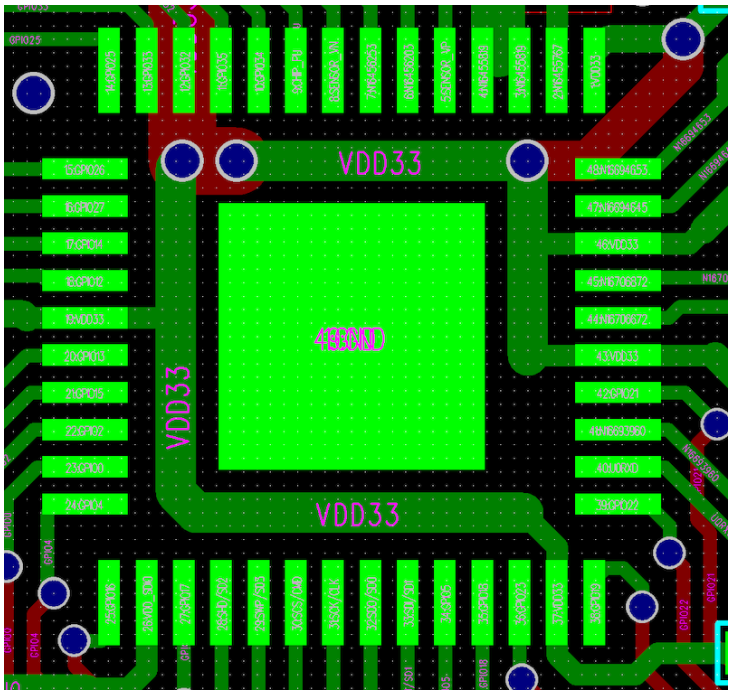

Fig. 14: ESP32 Power Traces in a Two-Layer PCB Design

- For a two-layer design, ensure to provide a continuous reference ground for the chip, RF, and crystal oscillator, as shown in the figure above.
- In the figure above, the trace VDD33 represents the 3.3 V power trace. Unlike a four-layer design, the power trace should be routed on the top layer as much as possible. Therefore, the thermal pad in the center of the chip should be reduced in size, allowing the power trace to pass between the signal pads and the thermal pad. Vias to the bottom layer should only be used when absolutely necessary.

- Other layout considerations are the same as for a four-layer design.

1.4.3 Crystal

Figure *ESP32 Crystal Layout* shows a reference layout for crystal on ESP32.

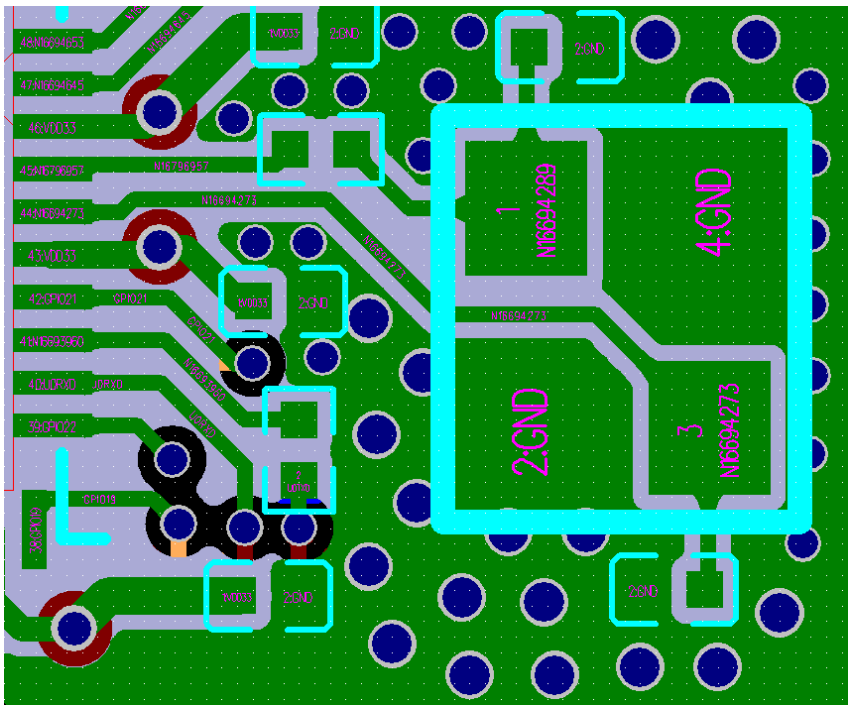

Fig. 15: ESP32 Crystal Layout

The layout of the crystal should follow the guidelines below:

- Ensure a complete GND plane for the RF, crystal, and chip.
- The crystal should be placed far from the clock pin to avoid interference on the chip. The gap should be at least 2.7 mm. It is good practice to add high-density ground vias stitching around the clock trace for better isolation.
- There should be no vias for the clock input and output traces.
- Components in series to the crystal trace should be placed close to the chip side.
- The external matching capacitors should be placed on the two sides of the crystal, preferably at the end of the clock trace, but not connected directly to the series components. This is to make sure the ground pad of the capacitor is close to that of the crystal.
- Do not route high-frequency digital signal traces under the crystal. It is best not to route any signal trace under the crystal. The vias on the power traces on both sides of the crystal clock trace should be placed as far away from the clock trace as possible, and the two sides of the clock trace should be surrounded by ground copper.
- As the crystal is a sensitive component, do not place any magnetic components nearby that may cause interference, for example large inductance component, and ensure that there is a clean large-area ground plane around the crystal.

1.4.4 RF

RF Layout on Four-layer PCB

The RF trace is routed as shown highlighted in pink in Figure *ESP32 RF Layout in a Four-layer PCB Design*.

The RF layout should meet the following guidelines:

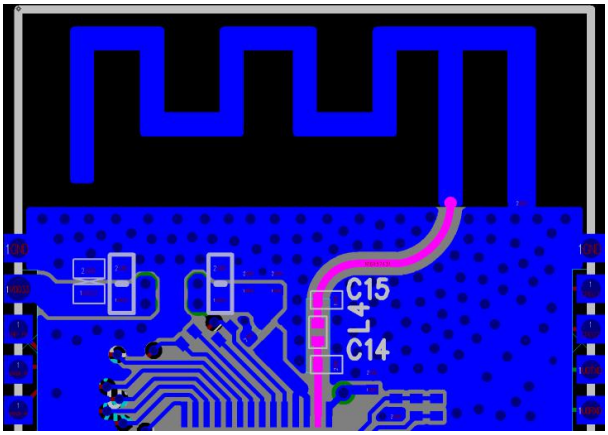

Fig. 16: ESP32 RF Layout in a Four-layer PCB Design

- The RF trace should have a 50 Ω characteristic impedance. The reference plane is the layer next to the chip. For designing the RF trace at 50 Ω impedance, you could refer to the PCB stack-up design shown below.

Thickness (mm)	Impedance (Ohm)	Gap (mil)	Width (mil)	Gap (mil)
-	50	12.2	12.6	12.2

Stack up	Material	Base copper (oz)	Finished Layer Thickness (mil)	DK
CM			0.4	4
L1_Top	Finished Copper 1 oz	0.33	0.8 (Min)	
PP	7628 TG150 RC50%		8.22	4.6
L2_Gnd		1	1.2	
Core	Core		Adjustable	4.6
L3_Power		1	1.2	
PP	7628 TG150 RC50%		8.22	4.6
L4_Bottom	Finished Copper 1 oz	0.33	0.8 (Min)	
SM			0.4	4

Fig. 17: ESP32 PCB Stack-up Design

- A CLC matching circuit is required for chip tuning. Please use 0201 components and place them close to the pin in a zigzag. In other words, the two capacitors should not be oriented in the same direction to minimize interference.
- Add a stub on the ground pad of the grounding capacitor near the chip side in the matching circuit to suppress the second harmonics. It is preferable to keep the stub length 15 mil, and determine the stub width according to the PCB stack-up so that the characteristic impedance of the stub is 100 Ω ± 10%. The reference plane is the third layer, so the area under the trace on the second layer should be cleared. The trace highlighted in Figure *ESP32 Stub in a Four-layer PCB Design* is the stub. Note that a stub is not required for package types

of 0402 and above.

- It is recommended to keep all layers clear under the IPEX antenna connector. See Figure *ESP32 IPEX Antenna Connector Layout*.
- For PCB antennas, make sure to validate them through both simulation and real-world testing on a development board. It is recommended to include an additional CLC matching circuit for antenna tuning. Place this circuit as close to the antenna as possible.

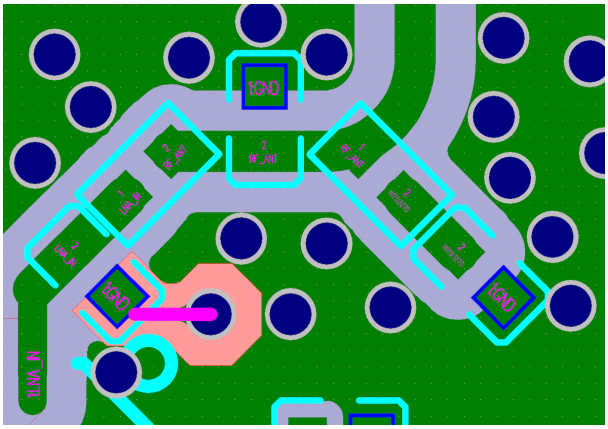

Fig. 18: ESP32 Stub in a Four-layer PCB Design

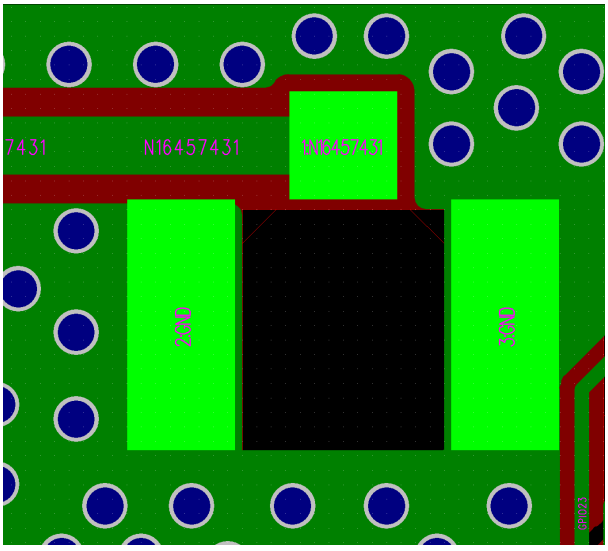

Fig. 19: ESP32 IPEX Antenna Connector Layout

- The RF trace should have a consistent width and not branch out. It should be as short as possible with dense ground vias around for interference shielding.
- The RF trace should be routed on the outer layer without vias, i.e., should not cross layers. The RF trace should be routed at a 135° angle, or with circular arcs if trace bends are required.
- The ground plane on the adjacent layer needs to be complete. Do not route any traces under the RF trace whenever possible.
- There should be no high-frequency signal traces routed close to the RF trace. The RF antenna should be placed away from high-frequency components, such as crystals, DDR SDRAM, high-frequency clocks, etc. In addition, the USB port, USB-to-serial chip, UART signal lines (including traces, vias, test points, header pins, etc.) must be as far away from the antenna as possible. The UART signal line should be surrounded by ground copper and ground vias.

RF Layout on Two-layer PCB

In a two-layer PCB design, the RF trace is routed as shown highlighted in pink in Figure *ESP32 RF Layout in a Two-layer PCB Design*. The width of the RF trace should be greater than that of the RF trace in a four-layer board and is normally over 20 mil. The actual width depends on the impedance formula where impedance-relevant parameters may vary depending on the number of PCB layers.

Other good practices for routing RF traces in four-layer PCB designs still apply to two-layer board designs.

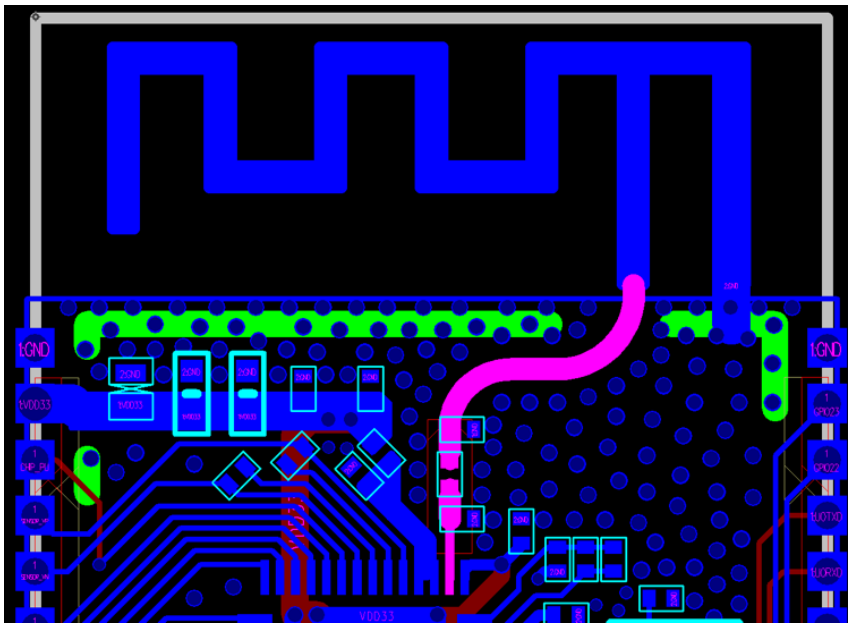

Fig. 20: ESP32 RF Layout in a Two-layer PCB Design

1.4.5 Flash and PSRAM

The layout for flash and PSRAM should follow the guidelines below:

- Place the zero-ohm resistors in series on the SPI lines close to ESP32.
- Route the SPI traces on the inner layer (e.g., the third layer) whenever possible, and add ground copper and ground vias around the clock and data traces of SPI separately.
- If the flash and PSRAM are located far from ESP32, it is recommended to place appropriate decoupling capacitors both at VDD_SDIO and near the flash and PSRAM power supply.

Figure *ESP32 Flash and PSRAM Layout* shows an example of flash (U3) and PSRAM (U4) layout.

1.4.6 External RC

External resistors and capacitors should be placed close to the chip pins, and there should be no vias around the traces. Please ensure that 10 nF capacitors are placed close to the pins.

1.4.7 UART

Figure *ESP32 UART Layout* shows the UART layout.

The UART layout should meet the following guidelines:

- The series resistor on the U0TXD trace needs to be placed close to the chip side and away from the crystal.

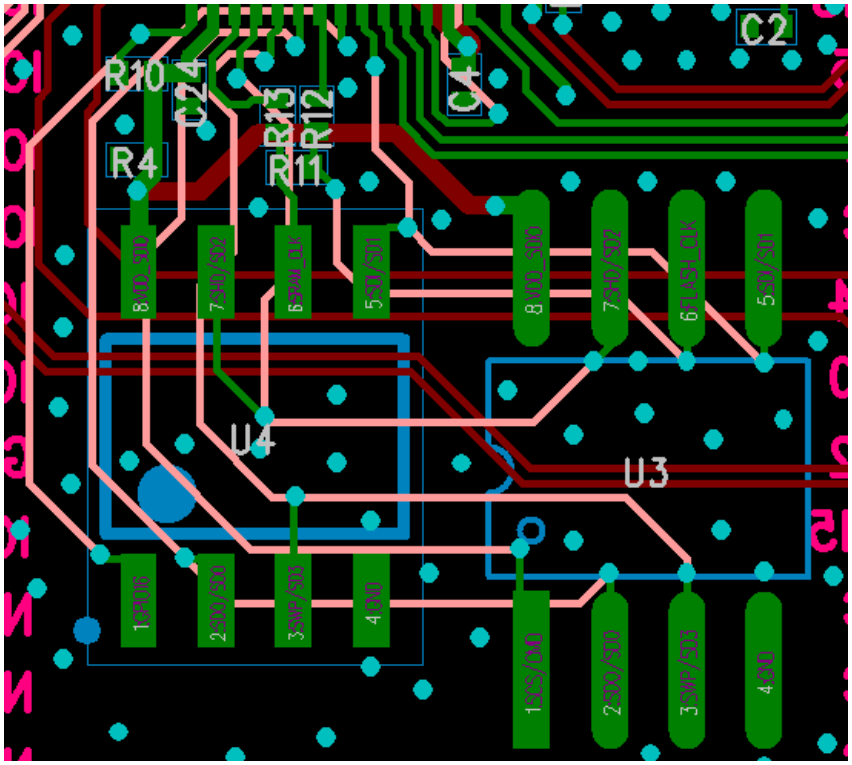

Fig. 21: ESP32 Flash and PSRAM Layout

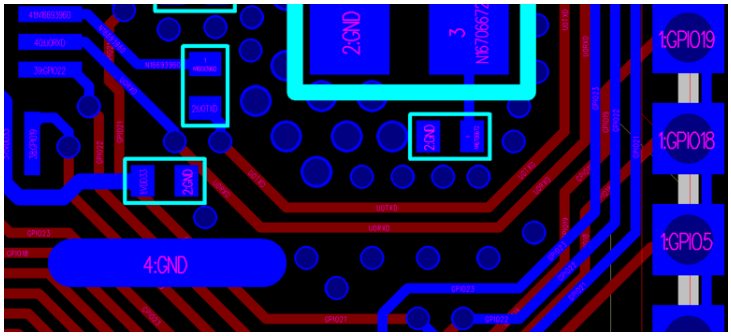

Fig. 22: ESP32 UART Layout

- The U0TXD and U0RXD traces on the top layer should be as short as possible.
- The UART trace should be surrounded by ground copper and ground vias stitching.

1.4.8 General Principles of PCB Layout for Modules (Positioning a Module on a Base Board)

If module-on-board design is adopted, attention should be paid while positioning the module on the base board. The interference of the baseboard on the module's antenna performance should be minimized.

It is suggested to place the module's on-board PCB antenna outside the base board, and the feed point of the antenna close to the edge of the base board. In the following example figures, positions with mark ✓ are strongly recommended, while positions without a mark are not recommended.

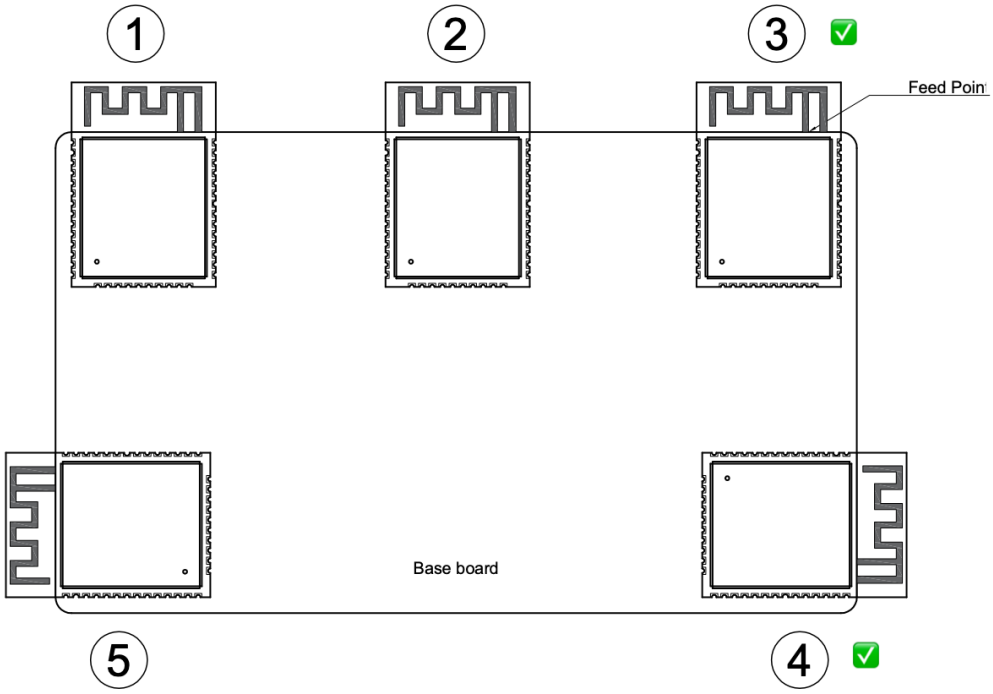

Fig. 23: Placement of ESP32 Modules on Base Board (antenna feed point on the right)

If the antenna cannot extend beyond the board edge, the feed point should still be placed as close to the board edge as possible. Then cut off the base board on both sides of the antenna and below it to minimize the impact of the base board material on the PCB antenna and provide a sufficiently large clearance area for the PCB antenna. Note that the module should not be placed in the center of the board with clearance created by hollowing out on all four sides. Figure *Keepout Zone for ESP32 Module's Antenna (Antenna feed point on the right)* shows the suggested clearance area. Please note that sufficient ground copper and dense ground vias should be placed on the base board near the antenna.

After the base board is placed in the end product, please consider the impact of the housing on the antenna during end-product design. Ensure that the PCB antenna on the base board also has a sufficiently large clearance area inside the housing. A clearance of at least 15 mm is recommended in all directions.

Please note that the final end product should be tested for throughput and communication range to ensure RF performance.

1.4.9 SDIO

The SDIO layout should follow the guidelines below:

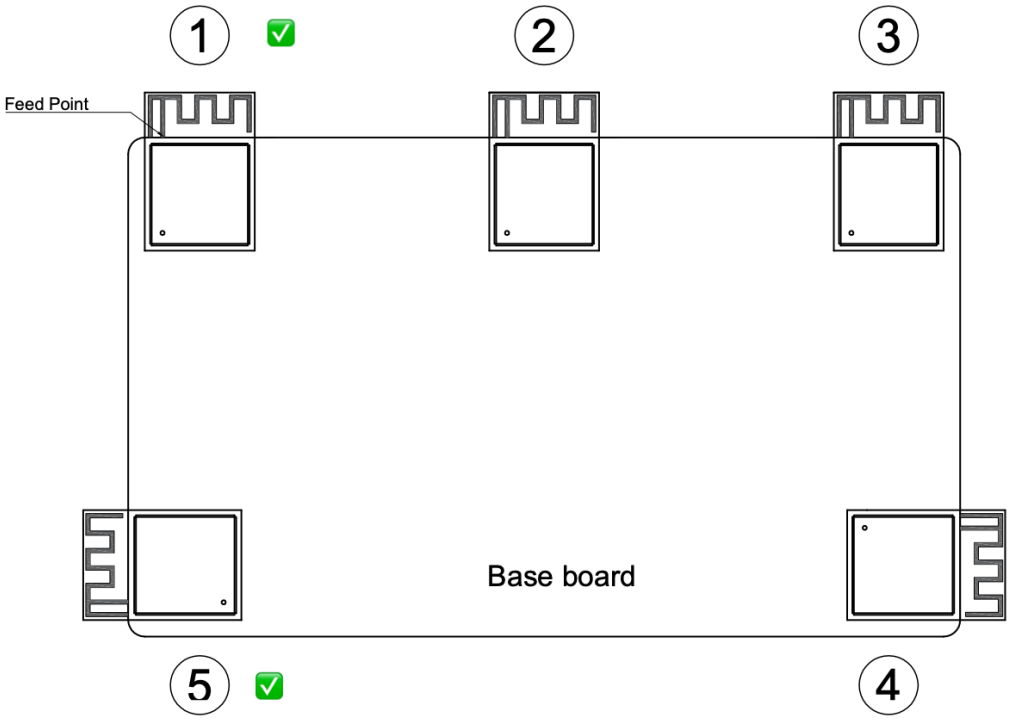

Fig. 24: Placement of ESP32 Modules on Base Board (antenna feed point on the left)

- Minimize parasitic capacitance of SDIO traces as they involve high-speed signals.
- The trace lengths for SDIO_CMD and SDIO_DATA0 ~ SDIO_DATA3 should be within ± 50 mil of the SDIO_CLK trace length. Use serpentine routing if necessary.
- For SDIO routing, maintain a 50 Ω single-ended impedance with a tolerance of ±10%.
- Keep the total trace length from SDIO GPIOs to the master SDIO interface as short as possible, ideally within 2000 mil.
- Ensure that SDIO traces do not cross layers. Besides, a reference plane (preferably a ground plane) must be placed beneath the traces, and continuity of the reference plane must be ensured.
- It is recommended to surround the SDIO_CLK trace with ground copper.

1.4.10 Touch Sensor

ESP32 offers up to 10 capacitive IOs that detect changes in capacitance on touch sensors due to finger contact or proximity. The chip's internal capacitance detection circuit features low noise and high sensitivity. It allows to use touch pads with smaller area to implement the touch detection function. You can also use the touch panel array to detect a larger area or more test points.

Figure *ESP32 Typical Touch Sensor Application* depicts a typical touch sensor application.

To prevent capacitive coupling and other electrical interference to the sensitivity of the touch sensor system, the following factors should be taken into account.

Electrode Pattern

The proper size and shape of an electrode improves system sensitivity. Round, oval, or shapes similar to a human fingertip are commonly applied. Large size or irregular shape might lead to incorrect responses from nearby electrodes.

Figure *ESP32 Electrode Pattern Requirements* shows the proper and improper size or shape of electrode. Please note that the examples illustrated in the figure are not of actual scale. It is suggested to use a human fingertip as reference.

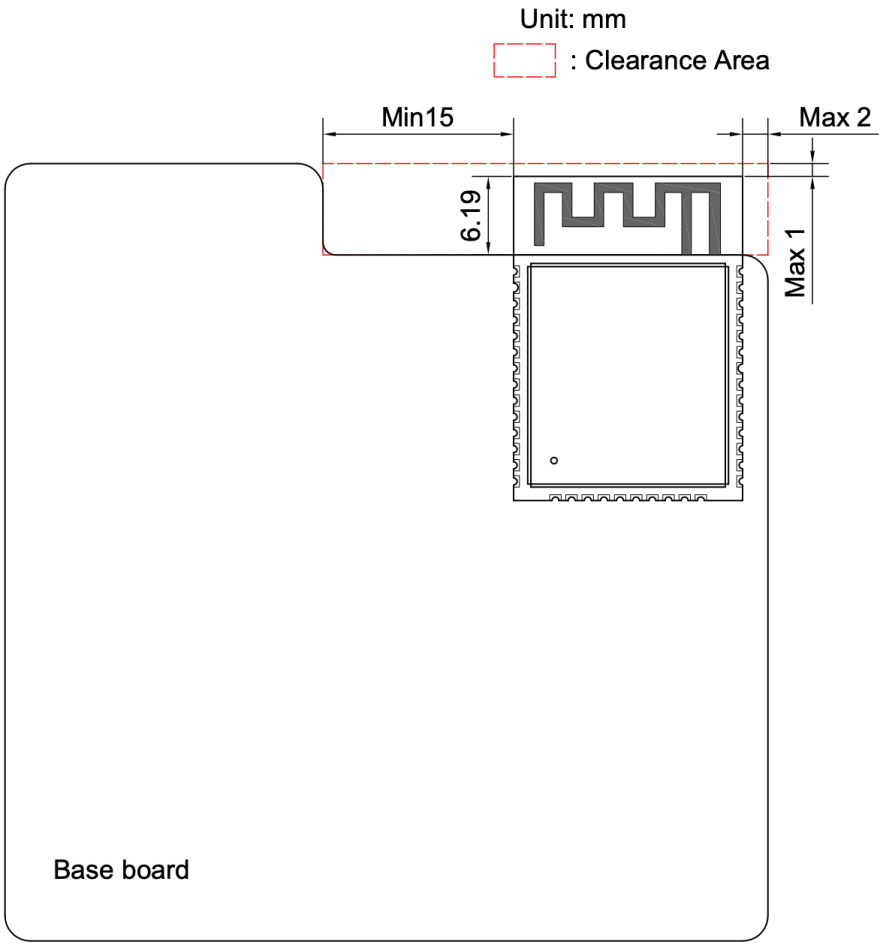

Fig. 25: Keepout Zone for ESP32 Module’ s Antenna (Antenna feed point on the right)

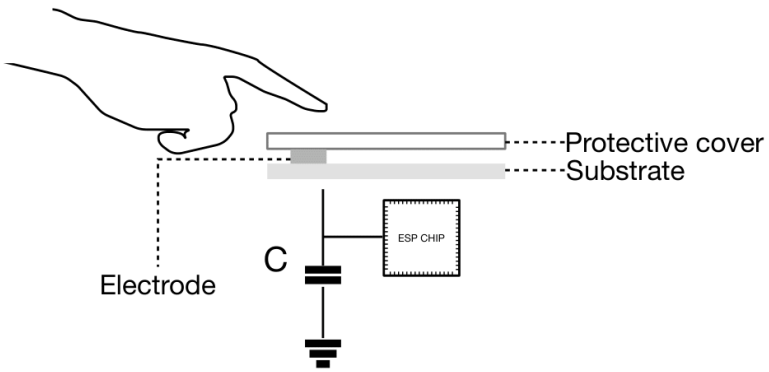

Fig. 26: ESP32 Typical Touch Sensor Application

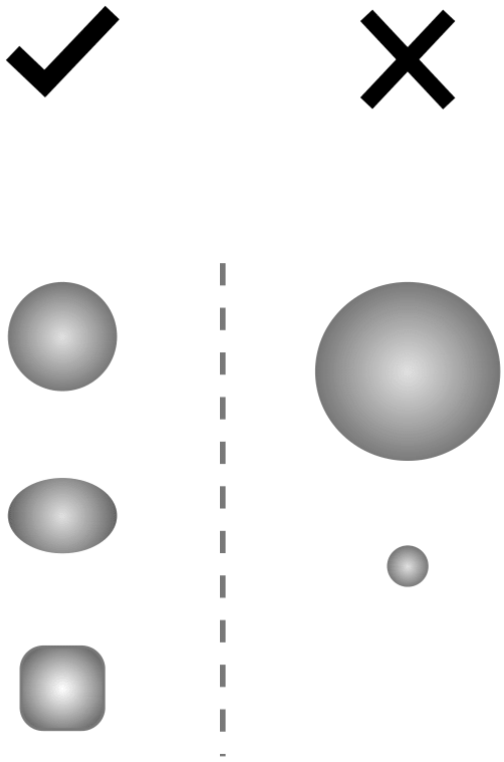

Fig. 27: ESP32 Electrode Pattern Requirements

PCB Layout

Figure *ESP32 Sensor Track Routing Requirements* illustrates the general guidelines to routing traces. Specifically,

- The trace should be as short as possible and no longer than 300 mm.
- The trace width (W) can not be larger than 0.18 mm (7 mil).
- The alignment angle (R) should not be less than 90°.
- The trace-to-ground gap (S) should be in the range of 0.5 mm to 1 mm.
- The electrode diameter (D) should be in the range of 8 mm to 15 mm.
- Hatched ground should be added around the electrodes and traces.
- The traces should be isolated well and routed away from that of the antenna.

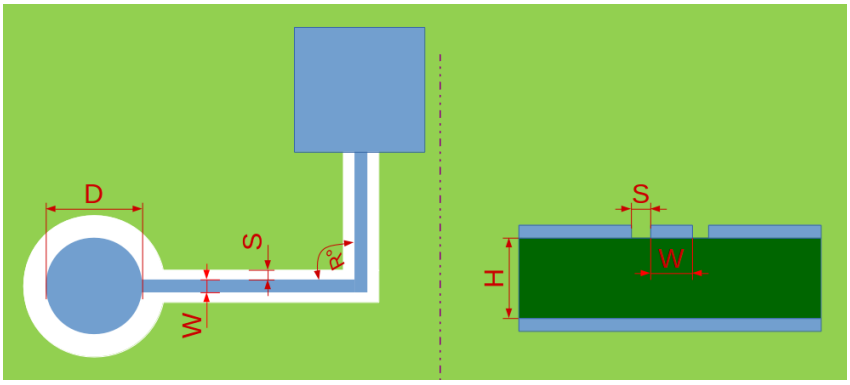

Fig. 28: ESP32 Sensor Track Routing Requirements

Note: For more details on the hardware design of the touch sensor, please refer to Touch Sensor Application Note.

1.4.11 Typical Layout Problems and Solutions

The voltage ripple is not large, but the TX performance of RF is rather poor.

Analysis: The voltage ripple has a strong impact on the RF TX performance. It should be noted that the ripple must be tested when ESP32 is in the normal working mode. The ripple increases when the power gets high in a different mode.

Generally, the peak-to-peak value of the ripple should be <80 mV when ESP32 sends MCS7@11n packets, and <120 mV when ESP32 sends 11 MHz@11b packets.

Solution: Add a 10 µF filter capacitor to the branch of the power trace (the branch powering the chip's analog power pin). The 10 µF capacitor should be as close to the analog power pin as possible for small and stable voltage ripples.

When ESP32 sends data packages, the voltage ripple is small, but RF TX performance is poor.

Analysis: The RF TX performance can be affected not only by voltage ripples, but also by the crystal itself. Poor quality and big frequency offsets of the crystal decrease the RF TX performance. The crystal clock may be corrupted by other interfering signals, such as high-speed output or input signals. In addition, high-frequency signal traces, such as the SDIO traces and UART traces under the crystal, could also result in the malfunction of the crystal. Besides, sensitive components or radiating components, such as inductors and antennas, may also decrease the RF performance.

Solution: This problem is caused by improper layout for the crystal and can be solved by re-layout. Please refer to Section *Crystal* for details.

When ESP32 sends data packages, the power value is much higher or lower than the target power value, and the EVM is relatively poor.

Analysis: The disparity between the tested value and the target value may be due to signal reflection caused by the impedance mismatch on the transmission line connecting the RF pin and the antenna. Besides, the impedance mismatch will affect the working state of the internal PA, making the PA prematurely access the saturated region in an abnormal way. The EVM becomes poor as the signal distortion happens.

Solution: Match the antenna's impedance with the π-type circuit on the RF trace, so that the impedance of the antenna as seen from the RF pin matches closely with that of the chip. This reduces reflections to the minimum.

TX performance is not bad, but the RX sensitivity is low.

Analysis: Good TX performance indicates proper RF impedance matching. Poor RX sensitivity may result from external coupling to the antenna. For instance, the crystal signal harmonics could couple to the antenna. If the TX and RX traces of UART cross over with RF trace, they will affect the RX performance, as well. If there are many high-frequency interference sources on the board, signal integrity should be considered.

Solution: Keep the antenna away from crystals. Do not route high-frequency signal traces close to the RF trace. Please refer to Section *RF* for details.

1.5 Download Guidelines

You can download firmware to ESP32 via UART.

To download via UART:

1. Before the download, make sure to set the chip or module to Joint Download Boot mode, according to Table *Boot Mode Control*.
2. Power up the chip or module and check the log via the UART0 serial port. If the log shows "waiting for download", the chip or module has entered Joint Download Boot mode.
3. Use the `Flash Download Tool` to flash firmware into flash via UART.
4. After the firmware has been downloaded, pull GPIO0 high or leave it floating to make sure that the chip or module enters SPI Boot mode.
5. Power up the chip or module again. The chip will read and execute the new firmware during initialization.

Note:

- For firmware download instructions, see also ESP Product Firmware Download Instructions.
- For how to check serial port output, see also Establish Serial Connection with ESP32.
- It is advised to download the firmware only after the "waiting for download" log shows via the serial port.
- Serial tools cannot be used simultaneously with the Flash Download Tool on one COM port.

1.6 Related Documentation and Resources

1.6.1 ESP32 Modules

This section lists only sample and mass-production ESP32 modules. For the complete list, see the Modules page on the Espressif website.

- ESP32-MINI-1
- ESP32-MINI-1U
- ESP32-WROOM-32E
- ESP32-WROOM-32UE
- ESP32-WROVER-E
- ESP32-WROVER-IE

For module reference designs please refer to:

- Download links

Note: Use the following tools to open the files in module reference designs:

- .DSN files: OrCAD Capture V16.6
- .pcb files: Pads Layout VX.2. If you cannot open the .pcb files, please try importing the .asc files into your software to view the PCB layout.

For the terms applicable to the use of these reference designs, please refer to *Terms of Use for Reference Designs* in *Disclaimer and Copyright Notice*.

1.6.2 ESP32 Development Boards

This section lists only sample and mass-production ESP32 development boards. For the complete list, see the Development Boards page on the Espressif website.

- ESP32-DevKitC
- ESP32-DevKitM-1
- ESP32-Ethernet-Kit
- ESP32-LyraT-Mini

1.6.3 Other Related Documentation and Resources

- Chip Datasheet (PDF)
- Technical Reference Manual (PDF)
- Chip Errata
- ESP32 Chip Variants
- Espressif KiCad Library
- ESP Product Selector
- Regulatory Certificates
- User Forum (Hardware)
- Technical Support
- ESP-FAQ

1.7 Glossary

The glossary contains terms and acronyms that are used in this document.

Term	Description
CLC	Capacitor-Inductor-Capacitor
DDR SDRAM	Double Data Rate Synchronous Dynamic Random-Access Memory
ESD	Electrostatic Discharge
LC	Inductor-Capacitor
PA	Power Amplifier
RC	Resistor-Capacitor
RTC	Real-Time Clock
Zero-ohm resistor	A zero-ohm resistor acts as a placeholder in the circuit, allowing for the replacement with a higher-ohm resistor based on specific design requirements.

1.8 Revision History

Table 13: Revision History

Date	Version	Release Notes
2025-09-17	v1.9	•***PCB Layout Design*** – Added Figure *Placement of ESP32 Modules on Base Board (antenna feed point on the left)* – Updated Figure *Placement of ESP32 Modules on Base Board (antenna feed point on the right)*
2025-07-23	v1.8	Improved wording throughout the document
2025-05-23	v1.7	•***PCB Layout Design*** – Section *SDIO*: Updated descriptions about the SDIO layout guidelines – Section *Crystal*: Updated descriptions about the crystal layout guidelines
2024-12-27	v1.6	•***Schematic Checklist*** – Section *RTC Clock Source (Optional)*: Updated descriptions about the 32.768 crystal and chip revisions
2024-12-09	v1.5	•***ESP32 Modules***: – Added download links to module reference designs
2024-11-15	v1.4	•***Schematic Checklist*** – Section *SPI*: Newly added section
2024-10-15	v1.3	•***Schematic Checklist*** – Section *Ethernet MAC*: Newly added section – Section *UART*: Updated the AT related description
2024-01-09	v1.2	•***Schematic Checklist*** – Section *RF Tuning*: Updated RF matching description
2023-12-25	v1.1	•***PCB Layout Design*** – Section *Crystal*: Updated crystal PCB layout
2023-12-22	v1.0	Migrated ESP32 Hardware Design Guidelines from PDF to HTML format. During the migration from PDF to HTML format, minor updates, improvements, and clarifications were made throughout the documentation. If you would like to check previous versions of the document, please submit documentation feedback.

1.9 Disclaimer and Copyright Notice

Information in this document, including URL references, is subject to change without notice.

All third party's information in this document is provided as is with no warranties to its authenticity and accuracy.

No warranty is provided to this document for its merchantability, non-infringement, fitness for any particular purpose,

nor does any warranty otherwise arising out of any proposal, specification or sample.

All liability, including liability for infringement of any proprietary rights, relating to use of information in this document is disclaimed. No licenses express or implied, by estoppel or otherwise, to any intellectual property rights are granted herein.

The Wi-Fi Alliance Member logo is a trademark of the Wi-Fi Alliance. The Bluetooth logo is a registered trademark of Bluetooth SIG.

All trade names, trademarks and registered trademarks mentioned in this document are property of their respective owners, and are hereby acknowledged.

1.9.1 Terms of Use for Reference Designs

The module reference designs provided in this documentation are open-source materials and may be used in product development, including commercial applications, provided that appropriate product verification and testing are carried out prior to use.

Use of these reference designs does not grant any rights to use Espressif's trademarks. In particular:

- Prior to incorporating a reference design into a product, any Espressif trademarks appearing on the PCB silkscreen, such as the Espressif logo or module part numbers, must be replaced with the manufacturer's own markings.
- Where the schematics or PCB layout are modified and the resulting design is released for commercial use, all Espressif trademarks and other proprietary markings must likewise be removed from the released schematics and PCB files.