ESP32 Series

Datasheet Version 4.8

2.4 GHz Wi-Fi + Bluetooth® + Bluetooth LE SoC

Including:

ESP32-D0WD-V3

ESP32-D0WDR2-V3

ESP32-U4WDH

ESP32-S0WD – Not Recommended for New Designs (NRND)

ESP32-D0WD – Not Recommended for New Designs (NRND)

ESP32-D0WDQ6 – Not Recommended for New Designs (NRND)

ESP32-D0WDQ6-V3 – Not Recommended for New Designs (NRND)

www.espressif.com

Product Overview

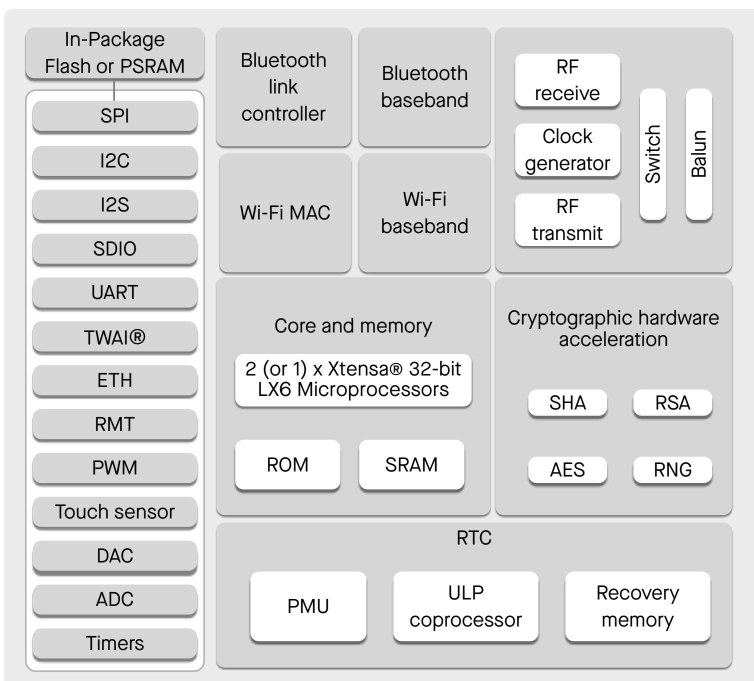
ESP32 is a single 2.4 GHz Wi-Fi-and-Bluetooth combo chip designed with the TSMC low-power 40 nm technology. It is designed to achieve the best power and RF performance, showing robustness, versatility and reliability in a wide variety of applications and power scenarios.

The ESP32 series of chips includes ESP32-D0WD-V3, ESP32-D0WDR2-V3, ESP32-U4WDH, ESP32-S0WD (NRND), ESP32-D0WDQ6-V3 (NRND), ESP32-D0WD (NRND), and ESP32-D0WDQ6 (NRND), among which,

- ESP32-S0WD (NRND), ESP32-D0WD (NRND), and ESP32-D0WDQ6 (NRND) are based on chip revision v1 or chip revision v1.1.
- ESP32-D0WD-V3, ESP32-D0WDR2-V3, ESP32-U4WDH, and ESP32-D0WDQ6-V3 (NRND) are based on chip revision v3.0 or chip revision v3.1.

For details on part numbers and ordering information, please refer to Section 1 ESP32 Series Comparison. For details on chip revisions, please refer to *ESP32 Chip Revision v3.0 User Guide* and *ESP32 Series SoC Errata*.

The functional block diagram of the SoC is shown below.



ESP32 Functional Block Diagram

Features

Wi-Fi

- 802.11b/g/n
- 802.11n (2.4 GHz), up to 150 Mbps
- WMM
- TX/RX A-MPDU, RX A-MSDU
- Immediate Block ACK
- Defragmentation
- Automatic Beacon monitoring (hardware TSF)
- Four virtual Wi-Fi interfaces
- Simultaneous support for Infrastructure Station, SoftAP, and Promiscuous modes
  Note that when ESP32 is in Station mode, performing a scan, the SoftAP channel will be changed.
- Antenna diversity

Bluetooth®

- Compliant with Bluetooth v4.2 BR/EDR and Bluetooth LE specifications
- Class-1, class-2 and class-3 transmitter without external power amplifier
- Enhanced Power Control
- +9 dBm transmitting power
- NZIF receiver with –94 dBm Bluetooth LE sensitivity
- Adaptive Frequency Hopping (AFH)
- Standard HCI based on SDIO/SPI/UART
- High-speed UART HCI, up to 4 Mbps
- Bluetooth 4.2 BR/EDR and Bluetooth LE dual mode controller
- Synchronous Connection-Oriented/Extended (SCO/eSCO)
- CVSD and SBC for audio codec
- Bluetooth Piconet and Scatternet
- Multi-connections in Classic Bluetooth and Bluetooth LE
- Simultaneous advertising and scanning

CPU and Memory

- Xtensa® single-/dual-core 32-bit LX6 microprocessor(s)
- CoreMark® score:
  - 1 core at 240 MHz: 504.85 CoreMark; 2.10 CoreMark/MHz

- 2 cores at 240 MHz: 994.26 CoreMark; 4.14 CoreMark/MHz

- 448 KB ROM
- 520 KB SRAM
- 16 KB SRAM in RTC
- QSPI supports multiple flash/SRAM chips

Clocks and Timers

- Internal 8 MHz oscillator with calibration
- Internal RC oscillator with calibration
- External 2 MHz ~ 60 MHz crystal oscillator (40 MHz only for Wi-Fi/Bluetooth functionality)
- External 32 kHz crystal oscillator for RTC with calibration
- Two timer groups, including 2 × 64-bit timers and 1 × main watchdog in each group
- One RTC timer
- RTC watchdog

Advanced Peripheral Interfaces

- 34 programmable GPIOs
  - Five strapping GPIOs
  - Six input-only GPIOs
  - Six GPIOs needed for in-package flash (ESP32-U4WDH) and in-package PSRAM (ESP32-D0WDR2-V3)
- 12-bit SAR ADC up to 18 channels
- Two 8-bit DAC
- 10 touch sensors
- Four SPI interfaces
- Two I2S interfaces
- Two I2C interfaces
- Three UART interfaces
- One host (SD/eMMC/SDIO)
- One slave (SDIO/SPI)
- Pulse count controller
- Ethernet MAC interface with dedicated DMA and IEEE 1588 support
- TWAI®, compatible with ISO 11898-1 (CAN Specification 2.0)
- RMT (TX/RX)

- Motor PWM
- LED PWM up to 16 channels

Power Management

- Fine-resolution power control through a selection of clock frequency, duty cycle, Wi-Fi operating modes, and individual power control of internal components
- Five power modes designed for typical scenarios: Active, Modem-sleep, Light-sleep, Deep-sleep, Hibernation
- Power consumption in Deep-sleep mode is 10 μA
- Ultra-Low-Power (ULP) coprocessors
- RTC memory remains powered on in Deep-sleep mode

Security

- Secure boot
- Flash encryption
- 1024-bit OTP, up to 768-bit for customers
- Cryptographic hardware acceleration:
  - AES
  - Hash (SHA-2)
  - RSA
  - ECC
  - Random Number Generator (RNG)

Applications

With low power consumption, ESP32 is an ideal choice for IoT devices in the following areas:

- 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

Note:

Check the link or the QR code to make sure that you use the latest version of this document:
https://www.espressif.com/documentation/esp32_datasheet_en.pdf

Contents

List of Tables

List of Figures

1 ESP32 Series Comparison

1.1 Nomenclature

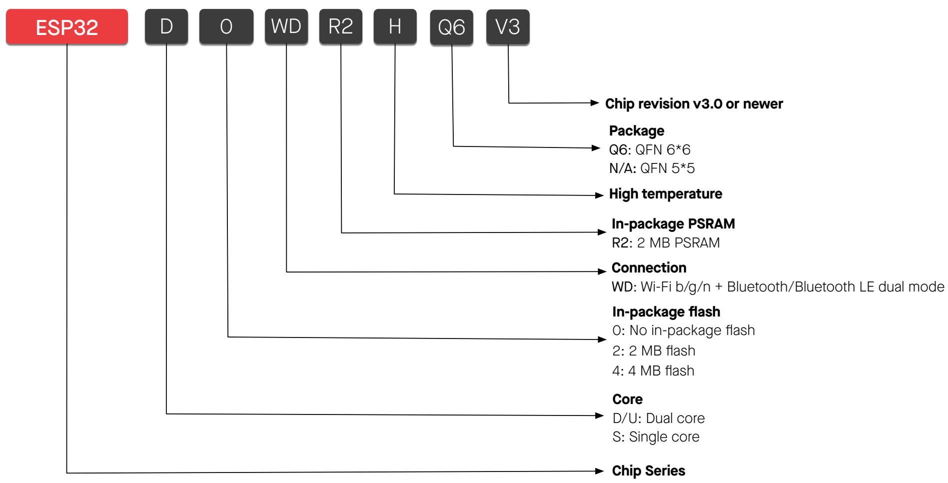


Figure 1-1. ESP32 Series Nomenclature

1.2 Comparison

Table 1-1. ESP32 Series Comparison

Ordering code[1]	Core	Chip Revision[2]	In-Package Flash/PSRAM	Package	VDD_SDIO Voltage
ESP32-D0WD-V3	Dual core	v3.0/v3.1[4]	—	QFN 5*5	1.8 V/3.3 V
ESP32-D0WDR2-V3	Dual core	v3.0/v3.1[4]	2 MB PSRAM	QFN 5*5	3.3 V
ESP32-U4WDH	Dual core[3]	v3.0/v3.1[4]	4 MB flash[6]	QFN 5*5	3.3 V
ESP32-D0WDQ6-V3 (NRND)	Dual core	v3.0/v3.1[4]	—	QFN 6*6	1.8 V/3.3 V
ESP32-D0WD (NRND)	Dual core	v1.0/v1.1[5]	—	QFN 5*5	1.8 V/3.3 V
ESP32-D0WDQ6 (NRND)	Dual core	v1.0/v1.1[5]	—	QFN 6*6	1.8 V/3.3 V
ESP32-S0WD (NRND)	Single core	v1.0/v1.1[5]	—	QFN 5*5	1.8 V/3.3 V

[1] All above chips support Wi-Fi b/g/n + Bluetooth/Bluetooth LE Dual Mode connection. For details on chip marking and packing, see Section 6 Packaging.

[2] Differences between ESP32 chip revisions and how to distinguish them are described in *ESP32 Series SoC Errata*.

[3] ESP32-U4WDH will be produced as dual-core instead of single core. See PCN-2021-021 for more details.

[4] The chips will be produced with chip revision v3.1 inside. See PCN20220901 for more details.

[5] The chips will be produced with chip revision v1.1 inside. See PCN20220901 for more details.

[6] The in-package flash supports:
- More than 100,000 program/erase cycles
- More than 20 years data retention time

2 Pins

2.1 Pin Layout

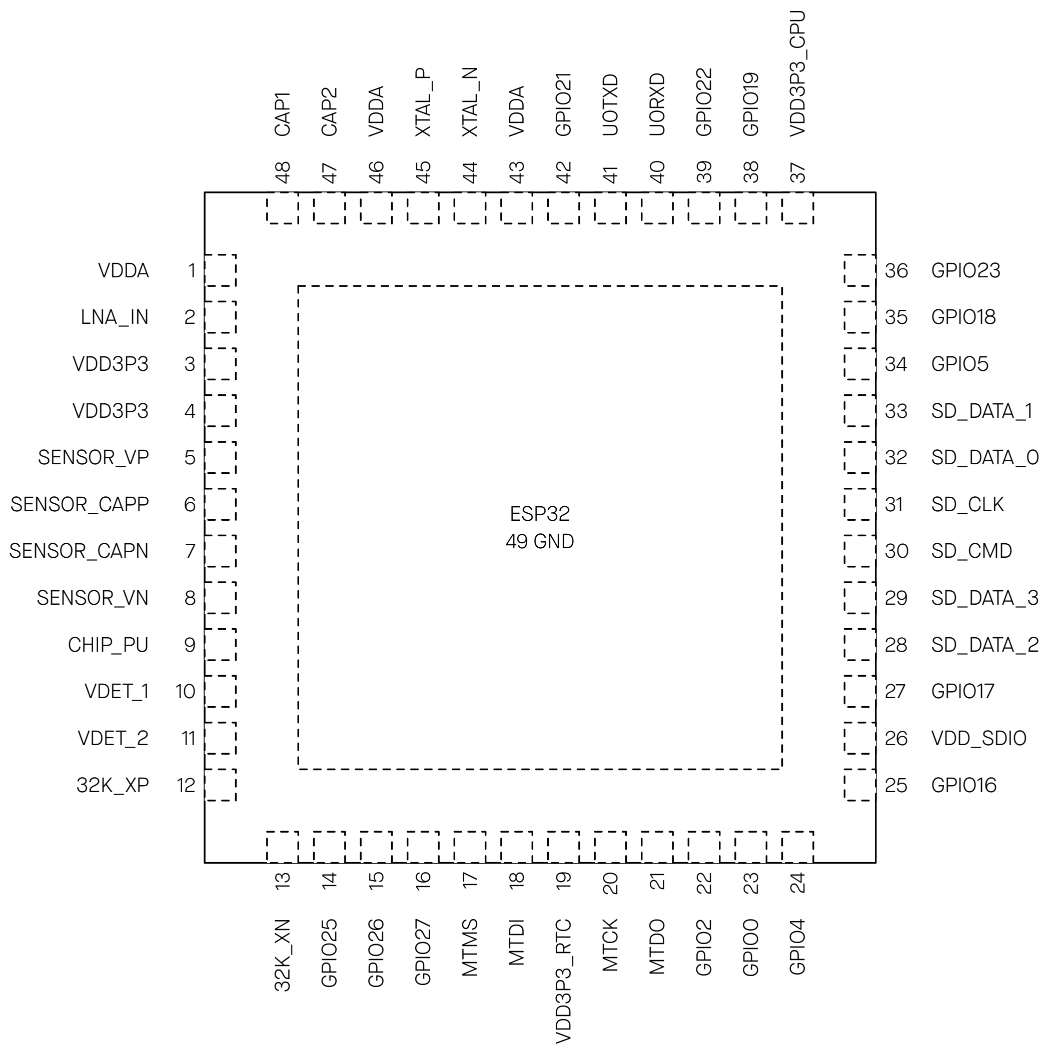


Figure 2-1. ESP32 Pin Layout (QFN 6*6, Top View)

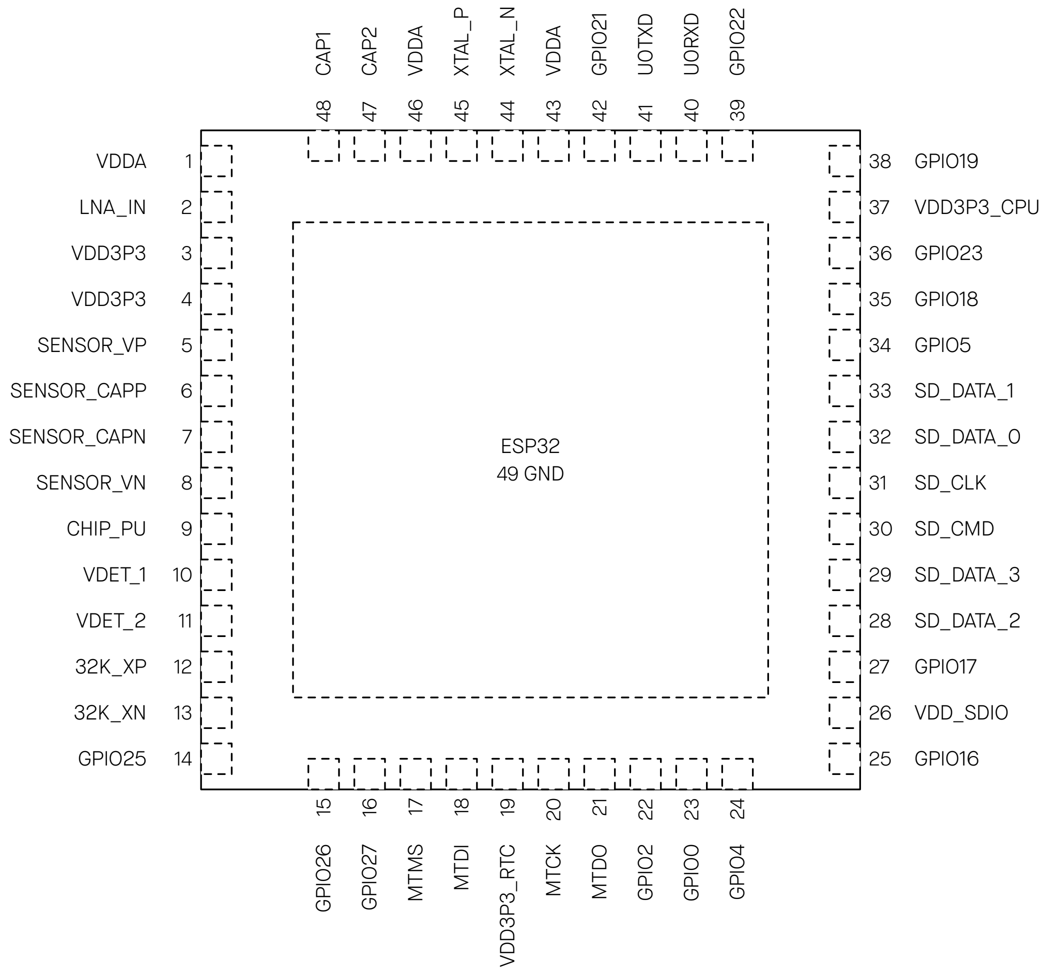


Figure 2-2. ESP32 Pin Layout (QFN 5*5, Top View)

2.2 Pin Overview

Table 2-1. Pin Overview

Name	No.	Type	Function
Analog			
VDDA	1	P	Analog power supply (2.3 V ~ 3.6 V)
LNA_IN	2	I/O	RF input and output
VDD3P3	3	P	Analog power supply (2.3 V ~ 3.6 V)
VDD3P3	4	P	Analog power supply (2.3 V ~ 3.6 V)
VDD3P3_RTC			
SENSOR_VP	5	I	GPIO36, ADC1_CH0, RTC_GPIO0
SENSOR_CAPP	6	I	GPIO37, ADC1_CH1, RTC_GPIO1
SENSOR_CAPN	7	I	GPIO38, ADC1_CH2, RTC_GPIO2
SENSOR_VN	8	I	GPIO39, ADC1_CH3, RTC_GPIO3
CHIP_PU	9	I	High: On; enables the chip Low: Off; the chip shuts down Note: Do not leave the CHIP_PU pin floating.
VDET_1	10	I	GPIO34, ADC1_CH6, RTC_GPIO4
VDET_2	11	I	GPIO35, ADC1_CH7, RTC_GPIO5
32K_XP	12	I/O	GPIO32, ADC1_CH4, RTC_GPIO9, TOUCH9, 32K_XP (32.768 kHz crystal oscillator input)
32K_XN	13	I/O	GPIO33, ADC1_CH5, RTC_GPIO8, TOUCH8, 32K_XN (32.768 kHz crystal oscillator output)
GPIO25	14	I/O	GPIO25, ADC2_CH8, RTC_GPIO6, DAC_1, EMAC_RXD0
GPIO26	15	I/O	GPIO26, ADC2_CH9, RTC_GPIO7, DAC_2, EMAC_RXD1
GPIO27	16	I/O	GPIO27, ADC2_CH7, RTC_GPIO17, TOUCH7, EMAC_RX_DV
MTMS	17	I/O	GPIO14, ADC2_CH6, RTC_GPIO16, TOUCH6, EMAC_TXD2, HSPICLK, HS2_CLK, SD_CLK, MTMS
MTDI	18	I/O	GPIO12, ADC2_CH5, RTC_GPIO15, TOUCH5, EMAC_TXD3, HSPIQ, HS2_DATA2, SD_DATA2, MTDI
VDD3P3_RTC	19	P	Input power supply for RTC IO (2.3 V ~ 3.6 V)
MTCK	20	I/O	GPIO13, ADC2_CH4, RTC_GPIO14, TOUCH4, EMAC_RX_ER, HSPID, HS2_DATA3, SD_DATA3, MTCK
MTDO	21	I/O	GPIO15, ADC2_CH3, RTC_GPIO13, TOUCH3, EMAC_RXD3, HSPICS0, HS2_CMD, SD_CMD, MTDO

Name	No.	Type	Function
GPIO2	22	I/O	GPIO2, ADC2_CH2, RTC_GPIO12, TOUCH2, HSPIWP, HS2_DATA0, SD_DATA0
GPIO0	23	I/O	GPIO0, ADC2_CH1, RTC_GPIO11, TOUCH1, EMAC_TX_CLK, CLK_OUT1,
GPIO4	24	I/O	GPIO4, ADC2_CH0, RTC_GPIO10, TOUCH0, EMAC_TX_ER, HSPIHD, HS2_DATA1, SD_DATA1
			VDD_SDIO
GPIO16	25	I/O	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
VDD_SDIO	26	P	Output power supply: 1.8 V or the same voltage as VDD3P3_RTC
GPIO17	27	I/O	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
SD_DATA_2	28	I/O	GPIO9, HS1_DATA2, U1RXD, SD_DATA2, SPIHD
SD_DATA_3	29	I/O	GPIO10, HS1_DATA3, U1TXD, SD_DATA3, SPIWP
SD_CMD	30	I/O	GPIO11, HS1_CMD, U1RTS, SD_CMD, SPICS0
SD_CLK	31	I/O	GPIO6, HS1_CLK, U1CTS, SD_CLK, SPICLK
SD_DATA_0	32	I/O	GPIO7, HS1_DATA0, U2RTS, SD_DATA0, SPIQ
SD_DATA_1	33	I/O	GPIO8, HS1_DATA1, U2CTS, SD_DATA1, SPID
			VDD3P3_CPU
GPIO5	34	I/O	GPIO5, HS1_DATA6, VSPICS0, EMAC_RX_CLK
GPIO18	35	I/O	GPIO18, HS1_DATA7, VSPICLK
GPIO23	36	I/O	GPIO23, HS1_STROBE, VSPID
VDD3P3_CPU	37	P	Input power supply for CPU IO (1.8 V ~ 3.6 V)
GPIO19	38	I/O	GPIO19, U0CTS, VSPIQ, EMAC_TXD0
GPIO22	39	I/O	GPIO22, U0RTS, VSPIWP, EMAC_TXD1
U0RXD	40	I/O	GPIO3, U0RXD, CLK_OUT2
U0TXD	41	I/O	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2
GPIO21	42	I/O	GPIO21, VSPIHD, EMAC_TX_EN
			Analog
VDDA	43	P	Analog power supply (2.3 V ~ 3.6 V)
XTAL_N	44	O	External crystal output
XTAL_P	45	I	External crystal input
VDDA	46	P	Analog power supply (2.3 V ~ 3.6 V)
CAP2	47	I	Connects to a 3.3 nF (10%) capacitor and 20 kΩ resistor in parallel to CAP1

Name	No.	Type	Function
CAP1	48	I	Connects to a 10 nF series capacitor to ground
GND	49	P	Ground

Notes for Table 2-1 Pin Overview:

1. Function names:

CLK_OUT...	clock output
SPICLK HSPICLK VSPICLK	SPI clock signal
HS..._CLK	SDIO Master clock signal
SD_CLK	SDIO Slave clock signal
EMAC_TX_CLK EMAC_RX_CLK	EMAC clock signal
U..._RTS U..._CTS	UART0/1/2 hardware flow control signals
U..._RXD U..._TXD	UART0/1/2 receive/transmit signals
MTMS MTDI MTCK MTDO	JTAG interface signals
GPIO...	General-purpose input/output with signals routed via the GPIO matrix. For more details on the GPIO matrix, see *ESP32 Technical Reference Manual* > Chapter *IO MUX and GPIO Matrix*。

2. Regarding highlighted cells, see Section 2.3.1 Restrictions for GPIOs and RTC_GPIOs.

3. For a quick reference guide to using the IO_MUX, Ethernet MAC, and GPIO Matrix pins of ESP32, please refer to Appendix ESP32 Pin Lists.

2.3 IO Pins

2.3.1 Restrictions for GPIOs and RTC_GPIOs

All IO pins of the ESP32 have GPIO and some have RTC_GPIO pin functions. However, these IO pins are multifunctional and can be configured for different purposes based on the requirements. Some IOs have restrictions for usage. It is essential to consider their multiplexed nature and the limitations when using these IO pins.

In Table 2-1 Pin Overview some pin functions are highlighted, specically:

- GPIO – **Input only pins**, output is not supported due to lack of pull-up/pull-down resistors.
- GPIO – allocated for communication with in-package flash/PSRAM and NOT recommended for other uses. For details, see Section 2.6 Pin Mapping Between Chip and Flash/PSRAM.
- GPIO – have one of the following important functions:
  - **Strapping pins** – need to be at certain logic levels at startup. See Section 3 *Boot Configurations*.
  - **JTAG interface** – often used for debugging.
  - **UART interface** – often used for debugging.

See also Appendix A.1 – Notes on ESP32 Pin Lists.

2.4 Analog Pins

Table 2-2. Analog Pins

Pin No.	Pin Name	Pin Type	Pin Function
1	LNA_IN	I/O	Low Noise Amplifier (LNA) input signal, Power Amplifier (PA) output signal
4	CHIP_PU	I	High: on, enables the chip (Powered up). Low: off, the chip powers off (powered down). Note: Do not leave the CHIP_PU pin floating.
53	XTAL_N	—	External clock input/output connected to chip's crystal or oscillator. P/N means differential clock positive/negative.
54	XTAL_P	—	

2.5 Power Supply

2.5.1 Power Pins

ESP32's digital pins are divided into three different power domains:

- VDD3P3_RTC
- VDD3P3_CPU
- VDD_SDIO

Table 2-3. Power Pins

Pin No.	Pin Name	Direction	Power Supply: Power Domain / Other	Power Supply: IO Pins
3	VDD3P3	Input	Analog power domain	
4	VDD3P3	Input	Analog power domain	
19	VDD3P3_RTC[1]	Input	RTC and part of Digital power domains	RTC IO
26	VDD3P3_SDIO[3]	Input/Output	Analog power domain	
37	VDD3P3_CPU[2]	Input	Digital power domain	Digital IO
43	VDDA	Input	Analog power domain	
46	VDDA	Input	Analog power domain	
49	GND	–	External ground connection	

[1] VDD3P3_RTC is also the input power supply for RTC and CPU.

[2] VDD3P3_CPU is also the input power supply for CPU.

[3] VDD_SDIO connects to the output of an internal LDO whose input is VDD3P3_RTC. When VDD_SDIO is connected to the same PCB net together with VDD3P3_RTC, the internal LDO is disabled automatically.

2.5.2 Power Scheme

The power scheme is shown in Figure 2-3 ESP32 Power Scheme.

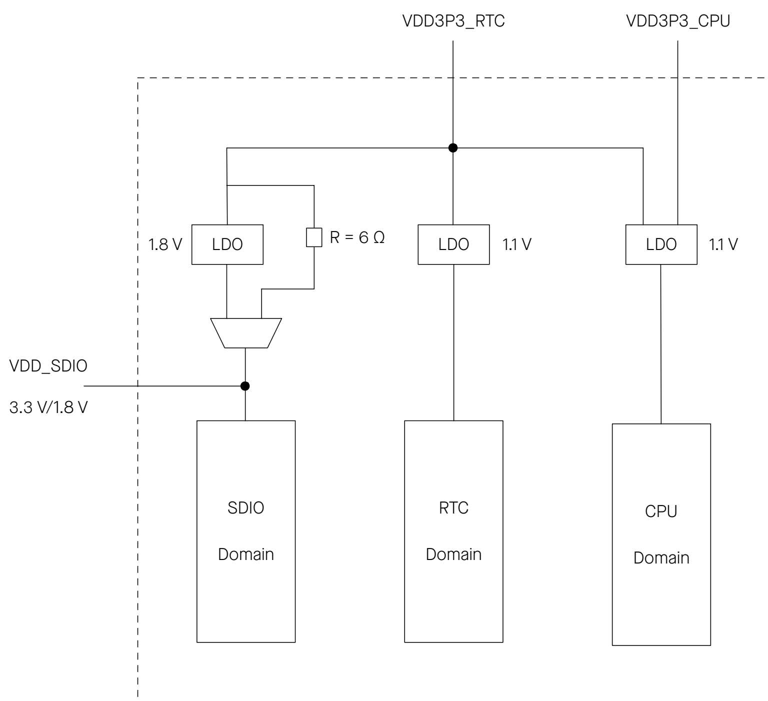


Figure 2-3. ESP32 Power Scheme

The internal LDO can be configured as having 1.8 V, or the same voltage as VDD3P3_RTC. It can be powered off via software to minimize the current of flash/SRAM during the Deep-sleep mode.

2.5.3 Chip Power-up and Reset

Once the power is supplied to the chip, its power rails need a short time to stabilize. After that, CHIP_PU – the pin used for power-up and reset – is pulled high to activate the chip. For information on CHIP_PU as well as power-up and reset timing, see Figure 2-4 and Table 2-4.

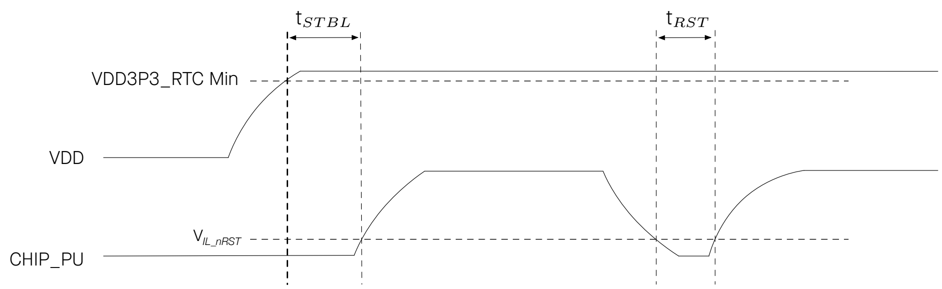


Figure 2-4. Visualization of Timing Parameters for Power-up and Reset

Table 2-4. Description of Timing Parameters for Power-up and Reset

Parameter	Description	Min (μs)
t_{STBL}	Time reserved for the 3.3 V 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 (see Table 5-3)	50

- In scenarios where ESP32 is powered up and down repeatedly by switching the power rails, while there is a large capacitor on the VDD33 rail and CHIP_PU and VDD33 are connected, simply switching off the CHIP_PU power rail and immediately switching it back on may cause an incomplete power discharge cycle and failure to reset the chip adequately.
  An additional discharge circuit may be required to accelerate the discharge of the large capacitor on rail VDD33, which will ensure proper power-on-reset when the ESP32 is powered up again.
- When a battery is used as the power supply for the ESP32 series of chips and modules, a supply voltage supervisor is recommended, so that a boot failure due to low voltage is avoided. Users are recommended to pull CHIP_PU low if the power supply for ESP32 is below 2.3 V.

Notes on power supply:

- The operating voltage of ESP32 ranges from 2.3 V to 3.6 V. When using a single-power supply, the recommended voltage of the power supply is 3.3 V, and its recommended output current is 500 mA or more.
- PSRAM and flash both are powered by VDD_SDIO. If the chip has an in-package flash, the voltage of VDD_SDIO is determined by the operating voltage of the in-package flash. If the chip also connects to an external PSRAM, the operating voltage of external PSRAM must match that of the in-package flash. This also applies if the chip has an in-package PSRAM but also connects to an external flash.

- When VDD_SDIO 1.8 V is used as the power supply for external flash/PSRAM, a 2 kΩ grounding resistor should be added to VDD_SDIO. For the circuit design, please refer to *ESP32 Hardware Design Guidelines*.
- When the three digital power supplies are used to drive peripherals, e.g., 3.3 V flash, they should comply with the peripherals' specifications.

2.6 Pin Mapping Between Chip and Flash/PSRAM

Table 2-5 lists the pin-to-pin mapping between the chip and the in-package flash/PSRAM. The chip pins listed here are not recommended for other usage.

For the data port connection between ESP32 and off-package flash/PSRAM please refer to Table 2-6.

Table 2-5. Pin-to-Pin Mapping Between Chip and In-Package Flash/PSRAM

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[1]	VDD
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[2]	CE#
GND	VSS
VDD_SDIO[1]	VDD

Table 2-6. Pin-to-Pin Mapping Between Chip and Off-Package Flash/PSRAM

Chip Pin	Off-Package Flash
SD_DATA_1/SPID	IO0/DI
SD_DATA_0/SPIQ	IO1/DO
SD_DATA_3/SPIWP	IO2/WP#
SD_DATA_2/SPIHD	IO3/HOLD#
SD_CLK	CLK
SD_CMD	CS#
GND	VSS
VDD_SDIO	VDD

Cont'd on next page

Table 2-6 – cont'd from previous page

Chip Pin	Off-Package PSRAM
Chip Pin	Off-Package PSRAM
SD_DATA_1	SIO0/SI
SD_DATA_0	SIO1/SO
SD_DATA_3	SIO2
SD_DATA_2	SIO3
SD_CLK/GPIO17[3]	SCLK
GPIO16[2]	CE#
GND	VSS
VDD_SDIO	VDD

Note:

1. As the in-package flash (ESP32-U4WDH) and the in-package PSRAM (ESP32-D0WDR2-V3) operate at 3.3 V, VDD_SDIO must be powered by VDD3P3_RTC via a 6 Ω resistor. See Figure 2-3 ESP32 Power Scheme.
2. If GPIO16 is used to connect to PSRAM's CE# signal, please add a pull-up resistor at the GPIO16 pin. See *ESP32-WROVER-E Datasheet* > Figure *Schematics of ESP32-WROVER-E*.
3. SD_CLK and GPIO17 pins are available to connect to the SCLK signal of external PSRAM.
   - If SD_CLK pin is selected, one GPIO (i.e., GPIO17) will be saved. The saved GPIO can be used for other purposes. This connection has passed internal tests, but relevant certification has not been completed.
   - Or GPIO17 pin is used to connect to the SCLK signal. This connection has passed relevant certification, see certificates for ESP32-WROVER-E.

   Please select the proper pin for your specific applications.

3 Boot Configurations

The chip allows for configuring the following boot parameters through strapping pins and eFuse bits at power-up or a hardware reset, without microcontroller interaction.

- **Chip boot mode**
  - Strapping pin: GPIO0 and GPIO2
- **Internal LDO (VDD_SDIO) Voltage**
  - Strapping pin: MTDI
  - eFuse bit: EFUSE_SDIO_FORCE and EFUSE_SDIO_TIEH
- **U0TXD printing**
  - Strapping pin: MTDO
- **Timing of SDIO Slave**
  - Strapping pin: MTDO and GPIO5
- **JTAG signal source**
  - eFuse bit: EFUSE_DISABLE_JTAG

The default values of all the above eFuse bits are 0, which means that they are not burnt. Given that eFuse is one-time programmable, once an eFuse bit is programmed to 1, it can never be reverted to 0. For how to program eFuse bits, please refer to *ESP32 Technical Reference Manual* > Chapter *eFuse Controller*.

The default values of the strapping pins, namely the logic levels, are determined by pins' internal weak pull-up/pull-down resistors at reset if the pins are not connected to any circuit, or connected to an external high-impedance circuit.

Table 3-1. Default Configuration of Strapping Pins

Strapping Pin	Default Configuration	Bit Value
GPIO0	Pull-up	1
GPIO2	Pull-down	0
MTDI	Pull-down	0
MTDO	Pull-up	1
GPIO5	Pull-up	1

To change the bit values, the strapping pins should be connected to external pull-down/pull-up resistances. If the ESP32 is used as a device by a host MCU, the strapping pin voltage levels can also be controlled by the host MCU.

All strapping pins have latches. At system reset, the latches sample the bit values of their respective strapping pins and store them until the chip is powered down or shut down. The states of latches cannot be changed in any other way. It makes the strapping pin values available during the entire chip operation, and the pins are freed up to be used as regular IO pins after reset.

The timing of signals connected to the strapping pins should adhere to the *setup time* and *hold time* specifications in Table 3-2 and Figure 3-1.

Table 3-2. Description of Timing Parameters for the Strapping Pins

Parameter	Description	Min (ms)
t_{SU}	*Setup time* is the time reserved for the power rails to stabilize before the CHIP_PU pin is pulled high to activate the chip.	0
t_H	*Hold time* is the 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.	1

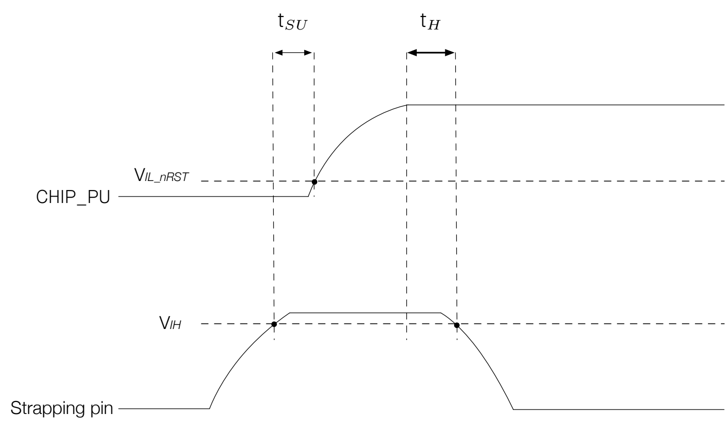


Figure 3-1. Visualization of Timing Parameters for the Strapping Pins

3.1 Chip Boot Mode Control

GPIO0 and GPIO2 control the boot mode after the reset is released. See Table 3-3 *Chip Boot Mode Control*.

Table 3-3. Chip Boot Mode Control

Boot Mode	GPIO0	GPIO2
SPI Boot Mode	**1**	Any value
Joint Download Boot Mode [2]	0	0

[1] **Bold** marks the default value and configuration.
[2] Joint Download Boot mode supports the following download methods:
- SDIO Download Boot
- UART Download Boot

In Joint Download Boot mode, the detailed boot flow of the chip is put below 3-2.

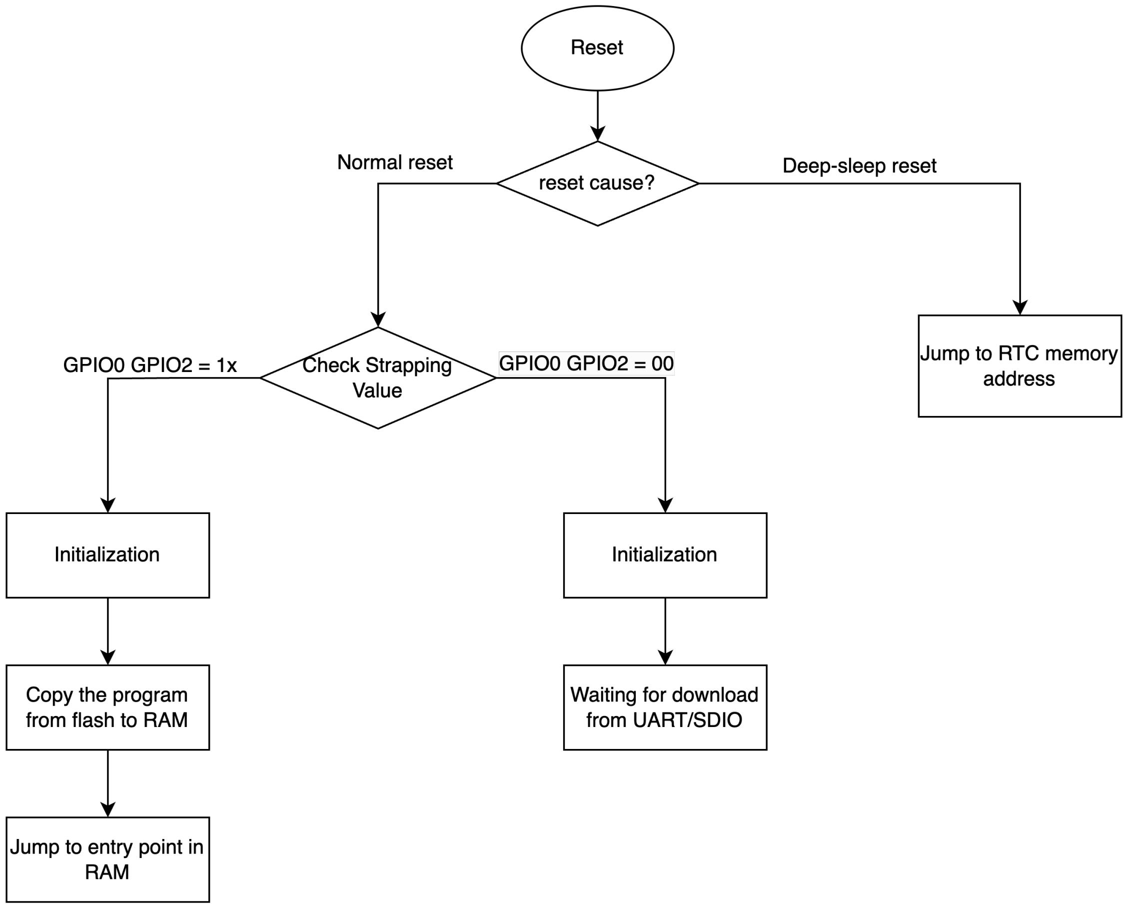


Figure 3-2. Chip Boot Flow

uart_download_dis controls boot mode behaviors:

It permanently disables Download Boot mode when uart_download_dis is set to 1 (valid only for ESP32 ECO V3).

3.2 Internal LDO (VDD_SDIO) Voltage Control

The required VDD_SPI voltage for the chips of the ESP32 Series can be found in Table 1-1 *Comparison*.

MTDI is used to select the VDD_SDIO power supply voltage at reset:

- MTDI = 0 (by default), VDD_SDIO pin is powered directly from VDD3P3_RTC. Typically this voltage is 3.3 V. For more information, see Section 2.5.2 *Power Scheme*.
- MTDI = 1, VDD_SDIO pin is powered from internal 1.8 V LDO.

This functionality can be overridden by setting EFUSE_SDIO_FORCE to 1, in which case the EFUSE_SDIO_TIEH determines the VDD_SDIO voltage:

- EFUSE_SDIO_TIEH = 0, VDD_SDIO connects to 1.8 V LDO.
- EFUSE_SPI_TIEH = 1, VDD_SDIO connects to VDD3P3_RTC.

3.3 U0TXD Printing Control

During booting, the strapping pin MTDO can be used to control the U0TXD Printing, as Table 3-4 shows.

Table 3-4. U0TXD Printing Control

U0TXD Printing Control	MTDO
Enabled [1]	**1**
Disabled	0

[1] **Bold** marks the default value and configuration.

3.4 Timing Control of SDIO Slave

The strapping pin MTDO and GPIO5 can be used to control the timing of SDIO slave, see Table 3-5 *Timing Control of SDIO Slave*.

Table 3-5. Timing Control of SDIO Slave

Edge behavior	MTDO	GPIO5
Falling edge sampling, falling edge output	0	0
Falling edge sampling, rising edge output	0	1
Rising edge sampling, falling edge output	1	0
Rising edge sampling, rising edge output	**1**	**1**

[1] **Bold** marks the default value and configuration.

3.5 JTAG Signal Source Control

If EFUSE_DISABLE_JTAG is set to 1, the source of JTAG signals can be disabled.

4 Functional Description

4.1 CPU and Memory

4.1.1 CPU

ESP32 contains one or two low-power Xtensa® 32-bit LX6 microprocessor(s) with the following features:

- 7-stage pipeline to support the clock frequency of up to 240 MHz (160 MHz for ESP32-S0WD (NRND))
- 16/24-bit Instruction Set provides high code-density
- Support for Floating Point Unit
- Support for DSP instructions, such as a 32-bit multiplier, a 32-bit divider, and a 40-bit MAC
- Support for 32 interrupt vectors from about 70 interrupt sources

The single-/dual-CPU interfaces include:

- Xtensa RAM/ROM Interface for instructions and data
- Xtensa Local Memory Interface for fast peripheral register access
- External and internal interrupt sources
- JTAG for debugging

For information about the Xtensa® Instruction Set Architecture, please refer to *Xtensa® Instruction Set Architecture (ISA) Summary*.

4.1.2 Internal Memory

ESP32's internal memory includes:

- 448 KB of ROM for booting and core functions
- 520 KB of on-chip SRAM for data and instructions
- 8 KB of SRAM in RTC, which is called RTC FAST Memory and can be used for data storage; it is accessed by the main CPU during RTC Boot from the Deep-sleep mode.
- 8 KB of SRAM in RTC, which is called RTC SLOW Memory and can be accessed by the ULP coprocessor during the Deep-sleep mode.
- 1 Kbit of eFuse: 256 bits are used for the system (MAC address and chip configuration) and the remaining 768 bits are reserved for customer applications, including flash-encryption and chip-ID.
- In-package flash or PSRAM

Note:
Products in the ESP32 series differ from each other, in terms of their support for in-package flash or PSRAM and the size of them. For details, please refer to Section 1 ESP32 Series Comparison.

4.1.3 External Flash and RAM

ESP32 supports multiple external QSPI flash and external RAM (SRAM) chips. More details can be found in *ESP32 Technical Reference Manual* > Chapter *SPI Controller*. ESP32 also supports hardware encryption/decryption based on AES to protect developers' programs and data in flash.

ESP32 can access the external QSPI flash and SRAM through high-speed caches.

- Up to 16 MB of external flash can be mapped into CPU instruction memory space and read-only memory space simultaneously.
  - When external flash is mapped into CPU instruction memory space, up to 11 MB + 248 KB can be mapped at a time. Note that if more than 3 MB + 248 KB are mapped, cache performance will be reduced due to speculative reads by the CPU.
  - When external flash is mapped into read-only data memory space, up to 4 MB can be mapped at a time. 8-bit, 16-bit and 32-bit reads are supported.
- External RAM can be mapped into CPU data memory space. SRAM up to 8 MB is supported and up to 4 MB can be mapped at a time. 8-bit, 16-bit and 32-bit reads and writes are supported.

Note:
After ESP32 is initialized, firmware can customize the mapping of external RAM or flash into the CPU address space.

4.1.4 Address Mapping Structure

The structure of address mapping is shown in Figure 4-1. The memory and peripheral mapping is shown in Table 4-1.

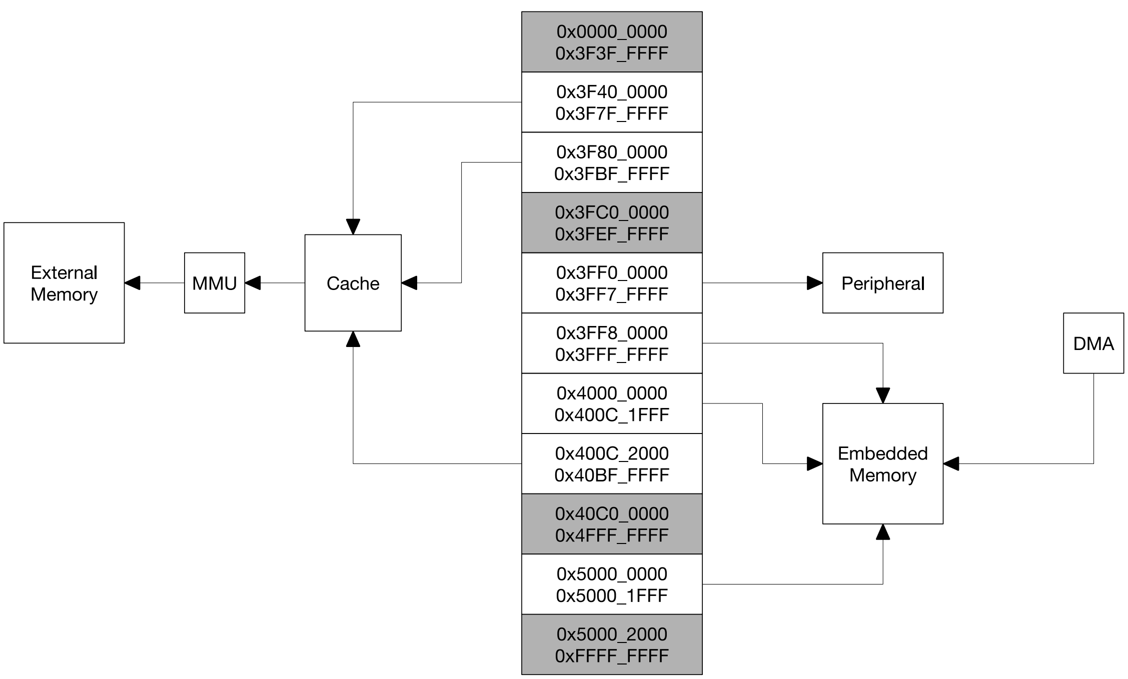


Figure 4-1. Address Mapping Structure

Table 4-1. Memory and Peripheral Mapping

Category	Target	Start Address	End Address	Size
Embedded Memory	Internal ROM 0	0x4000_0000	0x4005_FFFF	384 KB
	Internal ROM 1	0x3FF9_0000	0x3FF9_FFFF	64 KB
	Internal SRAM 0	0x4007_0000	0x4009_FFFF	192 KB
	Internal SRAM 1	0x3FFE_0000	0x3FFF_FFFF	128 KB
		0x400A_0000	0x400B_FFFF	
	Internal SRAM 2	0x3FFA_E000	0x3FFD_FFFF	200 KB
	RTC FAST Memory	0x3FF8_0000	0x3FF8_1FFF	8 KB
		0x400C_0000	0x400C_1FFF	
	RTC SLOW Memory	0x5000_0000	0x5000_1FFF	8 KB
External Memory	External Flash	0x3F40_0000	0x3F7F_FFFF	4 MB
		0x400C_2000	0x40BF_FFFF	11 MB+248 KB
	External RAM	0x3F80_0000	0x3FBF_FFFF	4 MB
Peripheral	DPort Register	0x3FF0_0000	0x3FF0_0FFF	4 KB
	AES Accelerator	0x3FF0_1000	0x3FF0_1FFF	4 KB
	RSA Accelerator	0x3FF0_2000	0x3FF0_2FFF	4 KB
	SHA Accelerator	0x3FF0_3000	0x3FF0_3FFF	4 KB
	Secure Boot	0x3FF0_4000	0x3FF0_4FFF	4 KB
	Cache MMU Table	0x3FF1_0000	0x3FF1_3FFF	16 KB
	PID Controller	0x3FF1_F000	0x3FF1_FFFF	4 KB
	UART0	0x3FF4_0000	0x3FF4_0FFF	4 KB
	SPI1	0x3FF4_2000	0x3FF4_2FFF	4 KB
	SPI0	0x3FF4_3000	0x3FF4_3FFF	4 KB
	GPIO	0x3FF4_4000	0x3FF4_4FFF	4 KB
	RTC	0x3FF4_8000	0x3FF4_8FFF	4 KB
	IO MUX	0x3FF4_9000	0x3FF4_9FFF	4 KB
	SDIO Slave	0x3FF4_B000	0x3FF4_BFFF	4 KB
	UDMA1	0x3FF4_C000	0x3FF4_CFFF	4 KB
	I2S0	0x3FF4_F000	0x3FF4_FFFF	4 KB
	UART1	0x3FF5_0000	0x3FF5_0FFF	4 KB
	I2C0	0x3FF5_3000	0x3FF5_3FFF	4 KB
	UDMA0	0x3FF5_4000	0x3FF5_4FFF	4 KB
	SDIO Slave	0x3FF5_5000	0x3FF5_5FFF	4 KB
	RMT	0x3FF5_6000	0x3FF5_6FFF	4 KB
	PCNT	0x3FF5_7000	0x3FF5_7FFF	4 KB
	SDIO Slave	0x3FF5_8000	0x3FF5_8FFF	4 KB
	LED PWM	0x3FF5_9000	0x3FF5_9FFF	4 KB
	eFuse Controller	0x3FF5_A000	0x3FF5_AFFF	4 KB
	Flash Encryption	0x3FF5_B000	0x3FF5_BFFF	4 KB
	PWM0	0x3FF5_E000	0x3FF5_EFFF	4 KB
	TIMG0	0x3FF5_F000	0x3FF5_FFFF	4 KB
	TIMG1	0x3FF6_0000	0x3FF6_0FFF	4 KB
	SPI2	0x3FF6_4000	0x3FF6_4FFF	4 KB
	SPI3	0x3FF6_5000	0x3FF6_5FFF	4 KB

Category	Target	Start Address	End Address	Size
Peripheral	SYSCON	0x3FF6_6000	0x3FF6_6FFF	4 KB
	I2C1	0x3FF6_7000	0x3FF6_7FFF	4 KB
	SDMMC	0x3FF6_8000	0x3FF6_8FFF	4 KB
	EMAC	0x3FF6_9000	0x3FF6_AFFF	8 KB
	TWAI	0x3FF6_B000	0x3FF6_BFFF	4 KB
	PWM1	0x3FF6_C000	0x3FF6_CFFF	4 KB
	I2S1	0x3FF6_D000	0x3FF6_DFFF	4 KB
	UART2	0x3FF6_E000	0x3FF6_EFFF	4 KB
	PWM2	0x3FF6_F000	0x3FF6_FFFF	4 KB
	PWM3	0x3FF7_0000	0x3FF7_0FFF	4 KB
	RNG	0x3FF7_5000	0x3FF7_5FFF	4 KB

4.1.5 Cache

ESP32 uses a two-way set-associative cache. Each of the two CPUs has 32 KB of cache featuring a block size of 32 bytes for accessing external storage.

For details, see *ESP32 Technical Reference Manual* > Chapter *System and Memory* > Section *Cache*.

4.2 System Clocks

4.2.1 CPU Clock

Upon reset, an external crystal clock source is selected as the default CPU clock. The external crystal clock source also connects to a PLL to generate a high-frequency clock (typically 160 MHz).

In addition, ESP32 has an internal 8 MHz oscillator. The application can select the clock source from the external crystal clock source, the PLL clock or the internal 8 MHz oscillator. The selected clock source drives the CPU clock directly, or after division, depending on the application.

4.2.2 RTC Clock

The RTC clock has five possible sources:

- external low-speed (32 kHz) crystal clock
- external crystal clock divided by 4
- internal RC oscillator (typically about 150 kHz, and adjustable)
- internal 8 MHz oscillator
- internal 31.25 kHz clock (derived from the internal 8 MHz oscillator divided by 256)

When the chip is in the normal power mode and needs faster CPU accessing, the application can choose the external high-speed crystal clock divided by 4 or the internal 8 MHz oscillator. When the chip operates in the low-power mode, the application chooses the external low-speed (32 kHz) crystal clock, the internal RC clock or the internal 31.25 kHz clock.

4.2.3 Audio PLL Clock

The audio clock is generated by the ultra-low-noise fractional-N PLL.

For details, see *ESP32 Technical Reference Manual* > Chapter *Reset and Clock*.

4.3 RTC and Low-power Management

4.3.1 Power Management Unit (PMU)

With the use of advanced power-management technologies, ESP32 can switch between different power modes.

- Power modes
  - **Active mode**: The chip radio is powered up. The chip can receive, transmit, or listen.
  - **Modem-sleep mode**: The CPU is operational and the clock is configurable. The Wi-Fi/Bluetooth baseband and radio are disabled.
  - **Light-sleep mode**: The CPU is paused. The RTC memory and RTC peripherals, as well as the ULP coprocessor are running. Any wake-up events (MAC, SDIO host, RTC timer, or external interrupts) will wake up the chip.
  - **Deep-sleep mode**: Only the RTC memory and RTC peripherals are powered up. Wi-Fi and Bluetooth connection data are stored in the RTC memory. The ULP coprocessor is functional.
  - **Hibernation mode**: The internal 8 MHz oscillator and ULP coprocessor are disabled. The RTC recovery memory is powered down. Only one RTC timer on the slow clock and certain RTC GPIOs are active. The RTC timer or the RTC GPIOs can wake up the chip from the Hibernation mode.

Table 4-2. Power Consumption by Power Modes

<table>
<tr><th>Power mode</th><th colspan="3">Description</th><th>Power Consumption</th></tr>
<tr><td rowspan="3">Active (RF working)</td><td colspan="3">Wi-Fi Tx packet</td><td rowspan="3">Please refer to Table 5-4 for details.</td></tr>
<tr><td colspan="3">Wi-Fi/BT Tx packet</td></tr>
<tr><td colspan="3">Wi-Fi/BT Rx and listening</td></tr>
<tr><td rowspan="6">Modem-sleep</td><td rowspan="6">The CPU is powered up.</td><td rowspan="2">240 MHz *</td><td>Dual-core chip(s)</td><td>30 mA ~ 68 mA</td></tr>
<tr><td>Single-core chip(s)</td><td>N/A</td></tr>
<tr><td rowspan="2">160 MHz *</td><td>Dual-core chip(s)</td><td>27 mA ~ 44 mA</td></tr>
<tr><td>Single-core chip(s)</td><td>27 mA ~ 34 mA</td></tr>
<tr><td rowspan="2">Normal speed: 80 MHz</td><td>Dual-core chip(s)</td><td>20 mA ~ 31 mA</td></tr>
<tr><td>Single-core chip(s)</td><td>20 mA ~ 25 mA</td></tr>
<tr><td>Light-sleep</td><td colspan="3">-</td><td>0.8 mA</td></tr>
<tr><td rowspan="3">Deep-sleep</td><td colspan="3">The ULP coprocessor is powered up.</td><td>150 μA</td></tr>
<tr><td colspan="3">ULP sensor-monitored pattern</td><td>100 μA @1% duty</td></tr>
<tr><td colspan="3">RTC timer + RTC memory</td><td>10 μA</td></tr>
<tr><td>Hibernation</td><td colspan="3">RTC timer only</td><td>5 μA</td></tr>
<tr><td>Power off</td><td colspan="3">CHIP_PU is set to low level, the chip is powered down.</td><td>1 μA</td></tr>
</table>

- * Among the ESP32 series of SoCs, ESP32-D0WD-V3, ESP32-D0WDR2-V3, ESP32-U4WDH, ESP32-D0WD (NRND), ESP32-D0WDQ6 (NRND), and ESP32-D0WDQ6-V3 (NRND) have a maximum CPU frequency of 240 MHz, ESP32-S0WD (NRND) has a maximum CPU frequency of 160 MHz.

- When Wi-Fi is enabled, the chip switches between Active and Modem-sleep modes. Therefore, power consumption changes accordingly.
- In Modem-sleep mode, the CPU frequency changes automatically. The frequency depends on the CPU load and the peripherals used.
- During Deep-sleep, when the ULP coprocessor is powered on, peripherals such as GPIO and RTC I2C are able to operate.
- When the system works in the ULP sensor-monitored pattern, the ULP coprocessor works with the ULP sensor periodically and the ADC works with a duty cycle of 1%, so the power consumption is 100 μA.

4.3.2 Ultra-Low-Power Coprocessor

The ULP coprocessor and RTC memory remain powered on during the Deep-sleep mode. Hence, the developer can store a program for the ULP coprocessor in the RTC slow memory to access the peripheral devices, internal timers and internal sensors during the Deep-sleep mode. This is useful for designing applications where the CPU needs to be woken up by an external event, or a timer, or a combination of the two, while maintaining minimal power consumption.

For details, see *ESP32 Technical Reference Manual* > Chapter *ULP Coprocessor*.

4.4 Timers and Watchdogs

4.4.1 General Purpose Timers

There are four general-purpose timers embedded in the chip. They are all 64-bit generic timers which are based on 16-bit prescalers and 64-bit auto-reload-capable up/down-timers.

The timers feature:

- A 16-bit clock prescaler, from 2 to 65536
- A 64-bit timer
- Configurable up/down timer: incrementing or decrementing
- Halt and resume of time-base counter
- Auto-reload at alarming
- Software-controlled instant reload
- Level and edge interrupt generation

For details, see *ESP32 Technical Reference Manual* > Chapter *Timer Group*.

4.4.2 Watchdog Timers

The chip has three watchdog timers: one in each of the two timer modules (called the Main Watchdog Timer, or MWDT) and one in the RTC module (called the RTC Watchdog Timer, or RWDT). These watchdog timers are intended to recover from an unforeseen fault causing the application program to abandon its normal sequence. A watchdog timer has four stages. Each stage may trigger one of three or four possible actions upon the expiry of its programmed time period, unless the watchdog is fed or disabled. The actions are:

interrupt, CPU reset, core reset, and system reset. Only the RWDT can trigger the system reset, and is able to reset the entire chip, including the RTC itself. A timeout value can be set for each stage individually.

During flash boot the RWDT and the first MWDT start automatically in order to detect, and recover from, booting problems.

The watchdogs have the following features:

- Four stages, each of which can be configured or disabled separately
- A programmable time period for each stage
- One of three or four possible actions (interrupt, CPU reset, core reset, and system reset) upon the expiry of each stage
- 32-bit expiry counter
- Write protection that prevents the RWDT and MWDT configuration from being inadvertently altered
- SPI flash boot protection
  If the boot process from an SPI flash does not complete within a predetermined time period, the watchdog will reboot the entire system.

For details, see *ESP32 Technical Reference Manual* > Chapter *Watchdog Timers*.

4.5 Cryptographic Hardware Accelerators

ESP32 is equipped with hardware accelerators of general algorithms, such as AES (FIPS PUB 197), SHA (FIPS PUB 180-4), RSA, and ECC. The chip also supports independent arithmetic, such as large-number modular multiplication and large-number multiplication. The maximum operation length for RSA, ECC, large-number modular multiplication, and large-number multiplication is 4096 bits.

The hardware accelerators greatly improve operation speed and reduce software complexity. They also support code encryption and dynamic decryption, which ensures that code in the flash will not be hacked.

4.6 Radio and Wi-Fi

The radio module consists of the following blocks:

- 2.4 GHz receiver
- 2.4 GHz transmitter
- bias and regulators
- balun and transmit-receive switch
- clock generator

4.6.1 2.4 GHz Receiver

The 2.4 GHz receiver demodulates the 2.4 GHz RF signal to quadrature baseband signals and converts them to the digital domain with two high-resolution, high-speed ADCs. To adapt to varying signal channel conditions, RF filters, Automatic Gain Control (AGC), DC offset cancelation circuits and baseband filters are integrated in the chip.

4.6.2 2.4 GHz Transmitter

The 2.4 GHz transmitter modulates the quadrature baseband signals to the 2.4 GHz RF signal, and drives the antenna with a high-powered Complementary Metal Oxide Semiconductor (CMOS) power amplifier. The use of digital calibration further improves the linearity of the power amplifier, enabling state-of-the-art performance in delivering up to +20.5 dBm of power for an 802.11b transmission and +18 dBm for an 802.11n transmission.

Additional calibrations are integrated to cancel any radio imperfections, such as:

- Carrier leakage
- I/Q phase matching
- Baseband nonlinearities
- RF nonlinearities
- Antenna matching

These built-in calibration routines reduce the amount of time required for product testing, and render the testing equipment unnecessary.

4.6.3 Clock Generator

The clock generator produces quadrature clock signals of 2.4 GHz for both the receiver and the transmitter. All components of the clock generator are integrated into the chip, including all inductors, varactors, filters, regulators and dividers.

The clock generator has built-in calibration and self-test circuits. Quadrature clock phases and phase noise are optimized on-chip with patented calibration algorithms which ensure the best performance of the receiver and the transmitter.

4.6.4 Wi-Fi Radio and Baseband

ESP32 implements a TCP/IP and full 802.11 b/g/n Wi-Fi MAC protocol. It supports the Basic Service Set (BSS) STA and SoftAP operations under the Distributed Control Function (DCF). Power management is handled with minimal host interaction to minimize the active-duty period.

The ESP32 Wi-Fi Radio and Baseband support the following features:

- 802.11b/g/n
- 802.11n MCS0-7 in both 20 MHz and 40 MHz bandwidth
- 802.11n MCS32 (RX)
- 802.11n 0.4 μs guard-interval
- up to 150 Mbps of data rate
- Receiving STBC 2×1
- Up to 20.5 dBm of transmitting power
- Adjustable transmitting power
- Antenna diversity
  ESP32 supports antenna diversity with an external RF switch. One or more GPIOs control the RF switch and selects the best antenna to minimize the effects of channel fading.

4.6.5 Wi-Fi MAC

The ESP32 Wi-Fi MAC applies low-level protocol functions automatically. They are as follows:

- Four virtual Wi-Fi interfaces
- Simultaneous Infrastructure BSS Station mode/SoftAP mode/Promiscuous mode
- RTS protection, CTS protection, Immediate Block ACK
- Defragmentation
- TX/RX A-MPDU, RX A-MSDU
- TXOP
- WMM
- CCMP (CBC-MAC, counter mode), TKIP (MIC, RC4), WAPI (SMS4), WEP (RC4) and CRC
- Automatic beacon monitoring (hardware TSF)

4.7 Bluetooth

The chip integrates a Bluetooth link controller and Bluetooth baseband, which carry out the baseband protocols and other low-level link routines, such as modulation/demodulation, packet processing, bit stream processing, frequency hopping, etc.

4.7.1 Bluetooth Radio and Baseband

The Bluetooth Radio and Baseband support the following features:

- Class-1, class-2 and class-3 transmit output powers, and a dynamic control range of up to 21 dB
- $\pi/4$ DQPSK and 8 DPSK modulation
- High performance in NZIF receiver sensitivity with a minimum sensitivity of -94 dBm
- Class-1 operation without external PA
- Internal SRAM allows full-speed data-transfer, mixed voice and data, and full piconet operation
- Logic for forward error correction, header error control, access code correlation, CRC, demodulation, encryption bit stream generation, whitening and transmit pulse shaping
- ACL, SCO, eSCO, and AFH
- A-law, μ-law, and CVSD digital audio CODEC in PCM interface
- SBC audio CODEC
- Power management for low-power applications
- SMP with 128-bit AES

4.7.2 Bluetooth Interface

- Provides UART HCI interface, up to 4 Mbps
- Provides SDIO/SPI HCI interface

- Provides PCM/I2S audio interface

4.7.3 Bluetooth Stack

The Bluetooth stack of the chip is compliant with the Bluetooth v4.2 BR/EDR and Bluetooth LE specifications.

4.7.4 Bluetooth Link Controller

The link controller operates in three major states: standby, connection and sniff. It enables multiple connections, and other operations, such as inquiry, page, and secure simple-pairing, and therefore enables Piconet and Scatternet. Below are the features:

- Classic Bluetooth
  - Device Discovery (inquiry, and inquiry scan)
  - Connection establishment (page, and page scan)
  - Multi-connections
  - Asynchronous data reception and transmission
  - Synchronous links (SCO/eSCO)
  - Master/Slave Switch
  - Adaptive Frequency Hopping and Channel assessment
  - Broadcast encryption
  - Authentication and encryption
  - Secure Simple-Pairing
  - Multi-point and scatternet management
  - Sniff mode
  - Connectionless Slave Broadcast (transmitter and receiver)
  - Enhanced power control
  - Ping
- Bluetooth Low Energy
  - Advertising
  - Scanning
  - Simultaneous advertising and scanning
  - Multiple connections
  - Asynchronous data reception and transmission
  - Adaptive Frequency Hopping and Channel assessment
  - Connection parameter update
  - Data Length Extension

- Link Layer Encryption
- LE Ping

4.8 Digital Peripherals

4.8.1 General Purpose Input / Output Interface (GPIO)

ESP32 has 34 GPIO pins which can be assigned various functions by programming the appropriate registers. There are several kinds of GPIOs: digital-only, analog-enabled, capacitive-touch-enabled, etc. Analog-enabled GPIOs and Capacitive-touch-enabled GPIOs can be configured as digital GPIOs.

Most of the digital GPIOs can be configured as internal pull-up or pull-down, or set to high impedance. When configured as an input, the input value can be read through the register. The input can also be set to edge-trigger or level-trigger to generate CPU interrupts. Most of the digital IO pins are bi-directional, non-inverting and tristate, including input and output buffers with tristate control. These pins can be multiplexed with other functions, such as the SDIO, UART, SPI, etc. (More details can be found in the Appendix, Table IO_MUX.) For low-power operations, the GPIOs can be set to hold their states.

For details, see Section 4.10 *Peripheral Pin Configurations*, Appendix A –ESP32 Pin Lists and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.2 Serial Peripheral Interface (SPI)

ESP32 features three SPIs (SPI, HSPI and VSPI) in slave and master modes in 1-line full-duplex and 1/2/4-line half-duplex communication modes.

Features of General Purpose SPI (GP-SPI)

- Programmable data transfer length, in multiples of 1 byte
- Four-line full-duplex/half-duplex communication and three-line half-duplex communication support
- Master mode and slave mode
- Programmable CPOL and CPHA
- Programmable clock

For details, see *ESP32 Technical Reference Manual* > Chapter *SPI Controller*.

Pin Assignment

For SPI, the pins are multiplexed with GPIO6 ~ GPIO11 via the IO MUX. For HSPI, the pins are multiplexed with GPIO2, GPIO4, GPIO12 ~ GPIO15 via the IO MUX. For VSPI, the pins are multiplexed with GPIO5, GPIO18 ~ GPIO19, GPIO21 ~ GPIO23 via the IO MUX.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.3 Universal Asynchronous Receiver Transmitter (UART)

The UART in the ESP32 chip facilitates the transmission and reception of asynchronous serial data between the chip and external UART devices. It consists of two UARTs in the main system, and one low-power LP UART.

Feature List

- Programmable baud rate
- RAM shared by TX FIFOs and RX FIFOs
- Supports input baud rate self-check
- Support for various lengths of data bits and stop bits
- Parity bit support
- Asynchronous communication (RS232 and RS485) and IrDA support
- Supports DMA to communicate data in high speed
- Supports UART wake-up
- Supports both software and hardware flow control

For details, see *ESP32 Technical Reference Manual* > Chapter *UART Controller*.

Pin Assignment

The pins for UART can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.4 I2C Interface

ESP32 has two I2C bus interfaces which can serve as I2C master or slave, depending on the user’s configuration.

Feature List

- Two I2C controllers: one in the main system and one in the low-power system
- Standard mode (100 Kbit/s)
- Fast mode (400 Kbit/s)
- Up to 5 MHz, yet constrained by SDA pull-up strength
- Support for 7-bit and 10-bit addressing, as well as dual address mode
- Supports continuous data transmission with disabled Serial Clock Line (SCL)
- Supports programmable digital noise filter

Users can program command registers to control I2C interfaces, so that they have more flexibility.

For details, see *ESP32 Technical Reference Manual* > Chapter *I2C Controller*.

Pin Assignment

For regular I2C, the pins used can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.5 I2S Interface

The I2S Controller in the ESP32 chip provides a flexible communication interface for streaming digital data in multimedia applications, particularly digital audio applications.

Feature List

- Master mode and slave mode
- Full-duplex and half-duplex communications
- A variety of audio standards supported
- Configurable high-precision output clock
- Supports PDM signal input and output
- Configurable data transmit and receive modes

For details, see *ESP32 Technical Reference Manual* > Chapter *I2S Controller*.

Pin Assignment

The pins for the I2S Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.6 Remote Control Peripheral

The Remote Control Peripheral (RMT) controls the transmission and reception of infrared remote control signals.

Feature List

- Eight channels for sending and receiving infrared remote control signals
- Independent transmission and reception capabilities for each channel
- Clock divider counter, state machine, and receiver for each RX channel
- Supports various infrared protocols

For details, see *ESP32 Technical Reference Manual* > Chapter *Remote Control Peripheral*.

Pin Assignment

The pins for the Remote Control Peripheral can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.7 Pulse Counter Controller (PCNT)

The pulse counter controller (PCNT) is designed to count input pulses by tracking rising and falling edges of the input pulse signal.

Feature List

- Eight independent pulse counter units
- Each pulse counter unit has a 16-bit signed counter register and two channels
- Counter modes: increment, decrement, or disable
- Glitch filtering for input pulse signals and control signals
- Selection between counting on rising or falling edges of the input pulse signal

For details, see *ESP32 Technical Reference Manual* > Chapter *Pulse Count Controller*.

Pin Assignment

The pins for the Pulse Count Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.8 LED PWM Controller

The LED PWM Controller (LEDC) is designed to generate PWM signals for LED control.

Feature List

- Sixteen independent PWM generators
- Maximum PWM duty cycle resolution of 20 bits
- Eight independent timers with 20-bit counters, configurable fractional clock dividers and counter overflow values
- Adjustable phase of PWM signal output
- PWM duty cycle dithering
- Automatic duty cycle fading

For details, see *ESP32 Technical Reference Manual* > Chapter *LED PWM Controller*.

Pin Assignment

The pins for the LED PWM Controller can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.9 Motor Control PWM

The Pulse Width Modulation (PWM) controller can be used for driving digital motors and smart lights. The controller consists of PWM timers, the PWM operator and a dedicated capture sub-module. Each timer provides timing in synchronous or independent form, and each PWM operator generates a waveform for one PWM channel. The dedicated capture sub-module can accurately capture events with external timing.

Feature List

- Three PWM timers for precise timing and frequency control
  - Every PWM timer has a dedicated 8-bit clock prescaler
  - The 16-bit counter in the PWM timer can work in count-up mode, count-down mode, or count-up-down mode
  - A hardware sync can trigger a reload on the PWM timer with a phase register. It will also trigger the prescaler' restart, so that the timer' s clock can also be synced, with selectable hardware synchronization source
- Three PWM operators for generating waveform pairs
  - Six PWM outputs to operate in several topologies
  - Configurable dead time on rising and falling edges; each set up independently
  - Modulating of PWM output by high-frequency carrier signals, useful when gate drivers are insulated with a transformer
- Fault Detection module
  - Programmable fault handling in both cycle-by-cycle mode and one-shot mode
  - A fault condition can force the PWM output to either high or low logic levels
- Capture module for hardware-based signal processing
  - Speed measurement of rotating machinery
  - Measurement of elapsed time between position sensor pulses
  - Period and duty cycle measurement of pulse train signals
  - Decoding current or voltage amplitude derived from duty-cycle-encoded signals of current/voltage sensors
  - Three individual capture channels, each of which with a 32-bit time-stamp register
  - Selection of edge polarity and prescaling of input capture signals
  - The capture timer can sync with a PWM timer or external signals

For details, see *ESP32 Technical Reference Manual* > Chapter *Motor Control PWM*.

Pin Assignment

The pins for the Motor Control PWM can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.10 SD/SDIO/MMC Host Controller

An SD/SDIO/MMC host controller is available on ESP32.

Feature List

- Supports two external cards
- Supports SD Memory Card standard: version 3.0 and version 3.01)
- Supports SDIO Version 3.0
- Supports Consumer Electronics Advanced Transport Architecture (CE-ATA Version 1.1)
- Supports Multimedia Cards (MMC version 4.41, eMMC version 4.5 and version 4.51)

The controller allows up to 80 MHz clock output in three different data-bus modes: 1-bit, 4-bit, and 8-bit modes. It supports two SD/SDIO/MMC4.41 cards in a 4-bit data-bus mode. It also supports one SD card operating at 1.8 V.

For details, see *ESP32 Technical Reference Manual* > Chapter *SD/MMC Host Controller*.

Pin Assignment

The pins for SD/SDIO/MMC Host Controller are multiplexed with GPIO2, GPIO4, GPIO6 ~ GPIO15 via IO MUX.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.11 SDIO/SPI Slave Controller

ESP32 integrates an SD device interface that conforms to the industry-standard SDIO Card Specification Version 2.0, and allows a host controller to access the SoC, using the SDIO bus interface and protocol. ESP32 acts as the slave on the SDIO bus. The host can access the SDIO-interface registers directly and can access shared memory via a DMA engine, thus maximizing performance without engaging the processor cores.

Feature List

The SDIO/SPI slave controller supports the following features:

- SPI, 1-bit SDIO, and 4-bit SDIO transfer modes over the full clock range from 0 to 50 MHz
- Configurable sampling and driving clock edge
- Special registers for direct access by host
- Interrupts to host for initiating data transfer
- Automatic loading of SDIO bus data and automatic discarding of padding data
- Block size of up to 512 bytes
- Interrupt vectors between the host and the slave, allowing both to interrupt each other
- Supports DMA for data transfer

For details, see *ESP32 Technical Reference Manual* > Chapter *SDIO Slave Controller*.

Pin Assignment

The pins for SDIO/SPI Slave Controller are multiplexed with GPIO2, GPIO4, GPIO6 ~ GPIO15 via IO MUX.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.12 TWAI® Controller

The Two-wire Automotive Interface (TWAI®) is a multi-master, multi-cast communication protocol designed for automotive applications. The TWAI controller facilitates the communication based on this protocol.

Feature List

- Compatible with ISO 11898-1 protocol (CAN Specification 2.0)
- Standard frame format (11-bit ID) and extended frame format (29-bit ID)
- Bit rates:
  - From 25 Kbit/s to 1 Mbit/s in chip revision v0.0/v1.0/v1.1
  - From 12.5 Kbit/s to 1 Mbit/s in chip revision v3.0/v3.1
- Multiple modes of operation: Normal, Listen Only, and Self-Test
- 64-byte receive FIFO
- Special transmissions: single-shot transmissions and self reception
- Acceptance filter (single and dual filter modes)
- Error detection and handling: error counters, configurable error interrupt threshold, error code capture, arbitration lost capture

For details, see *ESP32 Technical Reference Manual* > Chapter *Two-wire Automotive Interface (TWAI)*.

Pin Assignment

The pins for the Two-wire Automotive Interface can be chosen from any GPIOs via the GPIO Matrix.

For more information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.8.13 Ethernet MAC Interface

An IEEE-802.3-2008-compliant Media Access Controller (MAC) is provided for Ethernet LAN communications. ESP32 requires an external physical interface device (PHY) to connect to the physical LAN bus (twisted-pair, fiber, etc.). The PHY is connected to ESP32 through 17 signals of MII or nine signals of RMII.

Feature List

- 10 Mbps and 100 Mbps rates
- Dedicated DMA controller allowing high-speed transfer between the dedicated SRAM and Ethernet MAC
- Tagged MAC frame (VLAN support)

- Half-duplex (CSMA/CD) and full-duplex operation
- MAC control sublayer (control frames)
- 32-bit CRC generation and removal
- Several address-filtering modes for physical and multicast address (multicast and group addresses)
- 32-bit status code for each transmitted or received frame
- Internal FIFOs to buffer transmit and receive frames. The transmit FIFO and the receive FIFO are both 512 words (32-bit)
- Hardware PTP (Precision Time Protocol) in accordance with IEEE 1588 2008 (PTP V2)
- 25 MHz/50 MHz clock output

For details, see ESP32 Technical Reference Manual > Chapter *Ethernet Media Access Controller (MAC)*.

Pin Assignment

For information about the pin assignment of Ethernet MAC Interface, see Section 4.10 *Peripheral Pin Configurations* and ESP32 Technical Reference Manual > Chapter *IO_MUX and GPIO Matrix*.

4.9 Analog Peripherals

4.9.1 Analog-to-Digital Converter (ADC)

ESP32 integrates two 12-bit SAR ADCs and supports measurements on 18 channels (analog-enabled pins). The ULP coprocessor in ESP32 is also designed to measure voltage, while operating in the sleep mode, which enables low-power consumption. The CPU can be woken up by a threshold setting and/or via other triggers.

Table 4-3 describes the ADC characteristics.

Table 4-3. ADC Characteristics

Parameter	Description	Min	Max	Unit
DNL (Differential nonlinearity)	RTC controller; ADC connected to an external 100 nF capacitor; DC signal input; ambient temperature at 25 °C; Wi-Fi&Bluetooth off	–7	7	LSB
INL (Integral nonlinearity)		–12	12	LSB
Sampling rate	RTC controller	—	200	ksps
	DIG controller	—	2	Msps

Notes:

- When atten = 3 and the measurement result is above 3000 (voltage at approx. 2450 mV), the ADC accuracy will be worse than described in the table above.
- To get better DNL results, users can take multiple sampling tests with a filter, or calculate the average value.

- The input voltage range of GPIO pins within VDD3P3_RTC domain should strictly follow the DC characteristics provided in Table 5-3. Otherwise, measurement errors may be introduced, and chip performance may be affected.

By default, there are ±6% differences in measured results between chips. ESP-IDF provides couple of calibration methods for ADC1. Results after calibration using eFuse Vref value are shown in Table 4-4. For higher accuracy, users may apply other calibration methods provided in ESP-IDF, or implement their own.

Table 4-4. ADC Calibration Results

Parameter	Description	Min	Max	Unit
Total error	Atten = 0, effective measurement range of 100 ~ 950 mV	–23	23	mV
	Atten = 1, effective measurement range of 100 ~ 1250 mV	–30	30	mV
	Atten = 2, effective measurement range of 150 ~ 1750 mV	–40	40	mV
	Atten = 3, effective measurement range of 150 ~ 2450 mV	–60	60	mV

For details, see *ESP32 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

Pin Assignment

With appropriate settings, the ADCs can be configured to measure voltage on 18 pins maximum. For detailed information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.9.2 Digital-to-Analog Converter (DAC)

Two 8-bit DAC channels can be used to convert two digital signals into two analog voltage signal outputs. The design structure is composed of integrated resistor strings and a buffer. This dual DAC supports power supply as input voltage reference. The two DAC channels can also support independent conversions.

For details, see *ESP32 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

Pin Assignment

The DAC can be configured by GPIO 25 and GPIO 26. For detailed information about the pin assignment, see Section 4.10 *Peripheral Pin Configurations* and *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix*.

4.9.3 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.

Pin Assignment

The 10 capacitive-sensing GPIOs are listed in Table 4-5.

Table 4-5. Capacitive-Sensing GPIOs Available on ESP32

Capacitive-Sensing Signal Name	Pin Name
T0	GPIO4
T1	GPIO0
T2	GPIO2
T3	MTDO
T4	MTCK
T5	MTDI
T6	MTMS
T7	GPIO27
T8	32K_XN
T9	32K_XP

For details, see *ESP32 Technical Reference Manual* > Chapter *On-Chip Sensors and Analog Signal Processing*.

Note:
ESP32 Touch Sensor has not passed the Conducted Susceptibility (CS) test for now, and thus has limited application scenarios.

4.10 Peripheral Pin Configurations

Table 4-6. Peripheral Pin Configurations

Interface	Signal	Pin	Function
ADC	ADC1_CH0	SENSOR_VP	Two 12-bit SAR ADCs
	ADC1_CH1	SENSOR_CAPP	
	ADC1_CH2	SENSOR_CAPN	
	ADC1_CH3	SENSOR_VN	
	ADC1_CH4	32K_XP	
	ADC1_CH5	32K_XN	
	ADC1_CH6	VDET_1	
	ADC1_CH7	VDET_2	
	ADC2_CH0	GPIO4	
	ADC2_CH1	GPIO0	
	ADC2_CH2	GPIO2	
	ADC2_CH3	MTDO	
	ADC2_CH4	MTCK	
	ADC2_CH5	MTDI	
	ADC2_CH6	MTMS	
	ADC2_CH7	GPIO27	
	ADC2_CH8	GPIO25	
	ADC2_CH9	GPIO26	
DAC	DAC_1	GPIO25	Two 8-bit DACs
	DAC_2	GPIO26	
Touch Sensor	TOUCH0	GPIO4	Capacitive touch sensors
	TOUCH1	GPIO0	
	TOUCH2	GPIO2	
	TOUCH3	MTDO	
	TOUCH4	MTCK	
	TOUCH5	MTDI	
	TOUCH6	MTMS	
	TOUCH7	GPIO27	
	TOUCH8	32K_XN	
	TOUCH9	32K_XP	
JTAG	MTDI	MTDI	JTAG for software debugging
	MTCK	MTCK	
	MTMS	MTMS	
	MTDO	MTDO	

Interface	Signal	Pin	Function
SD/SDIO/MMC Host Controller	HS2_CLK	MTMS	Supports SD memory card V3.01 standard
	HS2_CMD	MTDO	
	HS2_DATA0	GPIO2	
	HS2_DATA1	GPIO4	
	HS2_DATA2	MTDI	
	HS2_DATA3	MTCK	
Motor PWM	PWM0_OUT0~2	Any GPIO Pins	Three channels of 16-bit timers generate PWM waveforms. Each channel has a pair of output signals, three fault detection signals, three event-capture signals, and three sync signals.
	PWM1_OUT_IN0~2		
	PWM0_FLT_IN0~2		
	PWM1_FLT_IN0~2		
	PWM0_CAP_IN0~2		
	PWM1_CAP_IN0~2		
	PWM0_SYNC_IN0~2		
	PWM1_SYNC_IN0~2		
SDIO/SPI Slave Controller	SD_CLK	MTMS	SDIO interface that conforms to the industry standard SDIO 2.0 card specification
	SD_CMD	MTDO	
	SD_DATA0	GPIO2	
	SD_DATA1	GPIO4	
	SD_DATA2	MTDI	
	SD_DATA3	MTCK	
UART	U0RXD_in	Any GPIO Pins	Three UART devices with hardware flow-control and DMA
	U0CTS_in		
	U0DSR_in		
	U0TXD_out		
	U0RTS_out		
	U0DTR_out		
	U1RXD_in		
	U1CTS_in		
	U1TXD_out		
	U1RTS_out		
	U2RXD_in		
	U2CTS_in		
	U2TXD_out		
	U2RTS_out		
I2C	I2CEXT0_SCL_in	Any GPIO Pins	Two I2C devices in slave or master mode
	I2CEXT0_SDA_in		
	I2CEXT1_SCL_in		
	I2CEXT1_SDA_in		
	I2CEXT0_SCL_out		
	I2CEXT0_SDA_out		
	I2CEXT1_SCL_out		
	I2CEXT1_SDA_out		

Interface	Signal	Pin	Function
LED PWM	ledc_hs_sig_out0~7	Any GPIO Pins	16 independent channels @80 MHz clock/RTC CLK. Duty accuracy: 16 bits.
	ledc_ls_sig_out0~7		
I2S	I2S0I_DATA_in0~15	Any GPIO Pins	Stereo input and output from/to the audio codec; parallel LCD data output; parallel camera data input. **Note:** I2S0_CLK and I2S1_CLK can only be mapped to GPIO0, U0RXD (GPIO3), or U0TXD (GPIO1) via IO MUX by selecting GPIO functions CLK_OUT1, CLK_OUT2, and CLK_OUT3. For more information, see *ESP32 Technical Reference Manual* > Chapter *IO_MUX and GPIO Matrix* > Table *IO MUX Pad Summary*.
	I2S0O_BCK_in		
	I2S0O_WS_in		
	I2S0I_BCK_in		
	I2S0I_WS_in		
	I2S0I_H_SYNC		
	I2S0I_V_SYNC		
	I2S0I_H_ENABLE		
	I2S0O_BCK_out		
	I2S0O_WS_out		
	I2S0I_BCK_out		
	I2S0I_WS_out		
	I2S0O_DATA_out0~23		
	I2S1I_DATA_in0~15		
	I2S1O_BCK_in		
	I2S1O_WS_in		
	I2S1I_BCK_in		
	I2S1I_WS_in		
	I2S1I_H_SYNC		
	I2S1I_V_SYNC		
	I2S1I_H_ENABLE		
	I2S1O_BCK_out		
	I2S1O_WS_out		
	I2S1I_BCK_out		
	I2S1I_WS_out		
	I2S1O_DATA_out0~23		
	I2S0_CLK	GPIO0, U0RXD, or U0TXD	
	I2S1_CLK		
RMT	RMT_SIG_IN0~7	Any GPIO Pins	Eight channels for an IR transmitter and receiver of various waveforms
	RMT_SIG_OUT0~7		
General Purpose SPI	HSPIQ_in/_out	Any GPIO Pins	Standard SPI consists of clock, chip-select, MOSI and MISO. These SPIs can be connected to LCD and other external devices. They support the following features: • Both master and slave modes; • Four sub-modes of the SPI transfer format; • Configurable SPI frequency; • Up to 64 bytes of FIFO and DMA.
	HSPID_in/_out		
	HSPICLK_in/_out		
	HSPI_CS0_in/_out		
	HSPI_CS1_out		
	HSPI_CS2_out		
	VSPIQ_in/_out		
	VSPID_in/_out		
	VSPICLK_in/_out		
	VSPI_CS0_in/_out		
	VSPI_CS1_out		
	VSPI_CS2_out		

Interface	Signal	Pin	Function
Parallel QSPI	SPIHD	SD_DATA_2	Supports Standard SPI, Dual SPI, and Quad SPI that can be connected to the external flash and SRAM
	SPIWP	SD_DATA_3	
	SPICS0	SD_CMD	
	SPICLK	SD_CLK	
	SPIQ	SD_DATA_0	
	SPID	SD_DATA_1	
	HSPICLK	MTMS	
	HSPICS0	MTDO	
	HSPIQ	MTDI	
	HSPID	MTCK	
	HSPIHD	GPIO4	
	HSPIWP	GPIO2	
	VSPICLK	GPIO18	
	VSPICS0	GPIO5	
	VSPIQ	GPIO19	
	VSPID	GPIO23	
	VSPIHD	GPIO21	
	VSPIWP	GPIO22	
EMAC	EMAC_TX_CLK	GPIO0	Ethernet MAC with MII/RMII interface
	EMAC_RX_CLK	GPIO5	
	EMAC_TX_EN	GPIO21	
	EMAC_TXD0	GPIO19	
	EMAC_TXD1	GPIO22	
	EMAC_TXD2	MTMS	
	EMAC_TXD3	MTDI	
	EMAC_RX_ER	MTCK	
	EMAC_RX_DV	GPIO27	
	EMAC_RXD0	GPIO25	
	EMAC_RXD1	GPIO26	
	EMAC_RXD2	U0TXD	
	EMAC_RXD3	MTDO	
	EMAC_CLK_OUT	GPIO16	
	EMAC_CLK_OUT_180	GPIO17	
	EMAC_TX_ER	GPIO4	
	EMAC_MDC_out	Any GPIO Pins	
	EMAC_MDI_in	Any GPIO Pins	
	EMAC_MDO_out	Any GPIO Pins	
	EMAC_CRS_out	Any GPIO Pins	
	EMAC_COL_out	Any GPIO Pins	

Interface	Signal	Pin	Function
Pulse Counter	pcnt_sig_ch0_in0	Any GPIO Pins	Operating in seven different modes, the pulse counter captures pulse and counts pulse edges.
	pcnt_sig_ch1_in0		
	pcnt_ctrl_ch0_in0		
	pcnt_ctrl_ch1_in0		
	pcnt_sig_ch0_in1		
	pcnt_sig_ch1_in1		
	pcnt_ctrl_ch0_in1		
	pcnt_ctrl_ch1_in1		
	pcnt_sig_ch0_in2		
	pcnt_sig_ch1_in2		
	pcnt_ctrl_ch0_in2		
	pcnt_ctrl_ch1_in2		
	pcnt_sig_ch0_in3		
	pcnt_sig_ch1_in3		
	pcnt_ctrl_ch0_in3		
	pcnt_ctrl_ch1_in3		
	pcnt_sig_ch0_in4		
	pcnt_sig_ch1_in4		
	pcnt_ctrl_ch0_in4		
	pcnt_ctrl_ch1_in4		
	pcnt_sig_ch0_in5		
	pcnt_sig_ch1_in5		
	pcnt_ctrl_ch0_in5		
	pcnt_ctrl_ch1_in5		
	pcnt_sig_ch0_in6		
	pcnt_sig_ch1_in6		
	pcnt_ctrl_ch0_in6		
	pcnt_ctrl_ch1_in6		
	pcnt_sig_ch0_in7		
	pcnt_sig_ch1_in7		
	pcnt_ctrl_ch0_in7		
	pcnt_ctrl_ch1_in7		
TWAI	twai_rx	Any GPIO Pins	Compatible with ISO 11898-1 protocol (CAN Specification 2.0)
	twai_tx		
	twai_bus_off_on		
	twai_clkout		

5 Electrical Characteristics

5.1 Absolute Maximum Ratings

Stresses above those listed in Table 5-1 *Absolute Maximum Ratings* may cause permanent damage to the device. These are stress ratings only and normal operation of the device at these or any other conditions beyond those indicated in Section 5.2 Recommended Power Supply Characteristics is not implied. Exposure to absolute-maximum-rated conditions for extended periods may affect device reliability.

Table 5-1. Absolute Maximum Ratings

Parameter	Description	Min	Max	Unit
VDDA, VDD3P3, VDD3P3_RTC, VDD3P3_CPU, VDD_SDIO	Allowed input voltage	–0.3	3.6	V
I_{output} [1]	Cumulative IO output current	—	1200	mA
T_{STORE}	Storage temperature	–40	150	°C

[1] The product proved to be fully functional after all its IO pins were pulled high while being connected to ground for 24 consecutive hours at ambient temperature of 25 °C.

5.2 Recommended Power Supply Characteristics

Table 5-2. Recommended Power Supply Characteristics

Parameter	Description	Min	Typ	Max	Unit
VDDA, VDD3P3_RTC, VDD3P3, VDD_SDIO (3.3 V mode) *note 1*	Voltage applied to power supply pins per power domain	2.3/3.0 *note 2*	3.3	3.6	V
VDD3P3_CPU	Voltage applied to power supply pin	1.8	3.3	3.6	V
I_{VDD}	Current delivered by external power supply	0.5	—	—	A
T *note 3*	Operating temperature	–40	—	125	°C

1. - VDD_SDIO works as the power supply for the related IO, and also for an external device. Please refer to the Appendix IO_MUX of this datasheet for more details.
   - VDD_SDIO can be sourced internally by the ESP32 from the VDD3P3_RTC power domain:
     - When VDD_SDIO operates at 3.3 V, it is driven directly by VDD3P3_RTC through a 6 Ω resistor, therefore, there will be some voltage drop from VDD3P3_RTC.
     - When VDD_SDIO operates at 1.8 V, it can be generated from ESP32's internal LDO. The maximum current this LDO can offer is 40 mA, and the output voltage range is 1.65 V ~ 2.0 V.
   - VDD_SDIO can also be driven by an external power supply.
   - Please refer to Section 2.5.2 Power Scheme, for more information.
2. - Chips with a 3.3 V flash or PSRAM in-package: this minimum voltage is 3.0 V;
   - Chips with no flash or PSRAM in-package: this minimum voltage is 2.3 V;
   - For more information, see Section 1 ESP32 Series Comparison.
3. - The operating temperature of ESP32-U4WDH ranges from –40 °C to 105 °C, due to the in-package flash.
   - The operating temperature of ESP32-D0WDR2-V3 ranges from –40 °C to 85 °C, due to the in-package PSRAM.
   - For other chips that have no in-package flash or PSRAM, their operating temperature is –40 °C ~ 125 °C.

5.3 DC Characteristics (3.3 V, 25 °C)

Table 5-3. DC Characteristics (3.3 V, 25 °C)

<table>
<tr><th>Parameter</th><th colspan="2">Description</th><th>Min</th><th>Typ</th><th>Max</th><th>Unit</th></tr>
<tr><td>C_{IN}</td><td colspan="2">Pin capacitance</td><td>—</td><td>2</td><td>—</td><td>pF</td></tr>
<tr><td>V_{IH}</td><td colspan="2">High-level input voltage</td><td>0.75×VDD[1]</td><td>—</td><td>VDD[1]+0.3</td><td>V</td></tr>
<tr><td>V_{IL}</td><td colspan="2">Low-level input voltage</td><td>–0.3</td><td>—</td><td>0.25×VDD[1]</td><td>V</td></tr>
<tr><td>I_{IH}</td><td colspan="2">High-level input current</td><td>—</td><td>—</td><td>50</td><td>nA</td></tr>
<tr><td>I_{IL}</td><td colspan="2">Low-level input current</td><td>—</td><td>—</td><td>50</td><td>nA</td></tr>
<tr><td>V_{OH}</td><td colspan="2">High-level output voltage</td><td>0.8×VDD[1]</td><td>—</td><td>—</td><td>V</td></tr>
<tr><td>V_{OL}</td><td colspan="2">Low-level output voltage</td><td>—</td><td>—</td><td>0.1×VDD[1]</td><td>V</td></tr>
<tr><td rowspan="3">I_{OH}</td><td rowspan="3">High-level source current (VDD[1] = 3.3 V, V_{OH} >= 2.64 V, output drive strength set to the maximum)</td><td>VDD3P3_CPU power domain [1, 2]</td><td>—</td><td>40</td><td>—</td><td>mA</td></tr>
<tr><td>VDD3P3_RTC power domain [1, 2]</td><td>—</td><td>40</td><td>—</td><td>mA</td></tr>
<tr><td>VDD_SDIO power domain [1, 3]</td><td>—</td><td>20</td><td>—</td><td>mA</td></tr>
<tr><td>I_{OL}</td><td colspan="2">Low-level sink current (VDD[1] = 3.3 V, V_{OL} = 0.495 V, output drive strength set to the maximum)</td><td>—</td><td>28</td><td>—</td><td>mA</td></tr>
<tr><td>R_{PU}</td><td colspan="2">Resistance of internal pull-up resistor</td><td>—</td><td>45</td><td>—</td><td>kΩ</td></tr>
<tr><td>R_{PD}</td><td colspan="2">Resistance of internal pull-down resistor</td><td>—</td><td>45</td><td>—</td><td>kΩ</td></tr>
<tr><td>V_{IL_nRST}</td><td colspan="2">Low-level input voltage of CHIP_PU to shut down the chip</td><td>—</td><td>—</td><td>0.6</td><td>V</td></tr>
</table>

1. Please see Table IO_MUX for IO's power domain. VDD is the I/O voltage for a particular power domain of pins.
2. For VDD3P3_CPU and VDD3P3_RTC power domain, per-pin current sourced in the same domain is gradually reduced from around 40 mA to around 29 mA, V_{OH}>=2.64 V, as the number of current-source pins increases.
3. For VDD_SDIO power domain, per-pin current sourced in the same domain is gradually reduced from around 30 mA to around 10 mA, V_{OH}>=2.64 V, as the number of current-source pins increases.

5.4 RF Current Consumption in Active Mode

The current consumption measurements are taken with a 3.3 V supply at 25 °C of ambient temperature at the RF port. All transmitters' measurements are based on a 50% duty cycle.

Table 5-4. Current Consumption Depending on RF Modes

Work Mode	Min	Typ	Max	Unit
Transmit 802.11b, DSSS 1 Mbps, POUT = +19.5 dBm	—	240	—	mA
Transmit 802.11g, OFDM 54 Mbps, POUT = +16 dBm	—	190	—	mA
Transmit 802.11n, OFDM MCS7, POUT = +14 dBm	—	180	—	mA
Receive 802.11b/g/n	—	95 ~ 100	—	mA
Transmit BT/BLE, POUT = 0 dBm	—	130	—	mA
Receive BT/BLE	—	95 ~ 100	—	mA

5.5 Reliability

Table 5-5. Reliability Qualifications

Test Item	Test Conditions	Test Standard
HTOL (High Temperature Operating Life)	125 °C, 1000 hours	JESD22-A108
ESD (Electro-Static Discharge Sensitivity)	HBM (Human Body Mode) [1] ± 2000 V	JS-001
	CDM (Charge Device Mode) [2] ± 500 V	JS-002
Latch up	Current trigger ± 200 mA	JESD78
	Voltage trigger 1.5 × VDD_{max}	
Preconditioning	Bake 24 hours @125 °C Moisture soak (level 3: 192 hours @30 °C, 60% RH) IR reflow solder: 260 + 0 °C, 20 seconds, three times	J-STD-020, JESD47, JESD22-A113
TCT (Temperature Cycling Test)	–65 °C / 150 °C, 500 cycles	JESD22-A104
Autoclave Test	121 °C, 100% RH, 96 hours	JESD22-A102
uHAST (Highly Accelerated Stress Test, unbiased)	130 °C, 85% RH, 96 hours	JESD22-A118
HTSL (High Temperature Storage Life)	150 °C, 1000 hours	JESD22-A103

1. JEDEC document JEP155 states that 500 V HBM allows safe manufacturing with a standard ESD control process.
2. JEDEC document JEP157 states that 250 V CDM allows safe manufacturing with a standard ESD control process.

5.6 Wi-Fi Radio

Table 5-6. Wi-Fi Radio Characteristics

Parameter	Description	Min	Typ	Max	Unit
Operating frequency range ^{note1}	—	2412	—	2484	MHz
Output impedance ^{note2}	-	-	*note 2*	—	Ω
TX power ^{note3}	11n, MCS7	12	13	14	dBm
	11b mode	18.5	19.5	20.5	dBm
Sensitivity	11b, 1 Mbps	—	–98	—	dBm
	11b, 11 Mbps	—	–88	—	dBm
	11g, 6 Mbps	—	–93	—	dBm
	11g, 54 Mbps	—	–75	—	dBm
	11n, HT20, MCS0	—	–93	—	dBm
	11n, HT20, MCS7	—	–73	—	dBm
	11n, HT40, MCS0	—	–90	—	dBm
	11n, HT40, MCS7	—	–70	—	dBm
Adjacent channel rejection	11g, 6 Mbps	—	27	—	dB

Parameter	Description	Min	Typ	Max	Unit
	11g, 54 Mbps	—	13	—	dB
	11n, HT20, MCS0	—	27	—	dB
	11n, HT20, MCS7	—	12	—	dB

1. Device should operate in the frequency range allocated by regional regulatory authorities. Target operating frequency range is configurable by software.
2. The typical value of the Wi-Fi radio output impedance is different between chips in different QFN packages. For chips in a QFN 6×6 package, the value is 30+j10 Ω. For chips in a QFN 5×5 package, the value is 35+j10 Ω.
3. Target TX power is configurable based on device or certification requirements.

5.7 Bluetooth Radio

5.7.1 Receiver –Basic Data Rate

Table 5-7. Receiver Characteristics –Basic Data Rate

Parameter	Description	Min	Typ	Max	Unit
Sensitivity @0.1% BER	—	–90	–89	–88	dBm
Maximum received signal @0.1% BER	—	0	—	—	dBm
Co-channel C/I	—	—	+7	—	dB
Adjacent channel selectivity C/I	F = F0 + 1 MHz	—	—	–6	dB
	F = F0 –1 MHz	—	—	–6	dB
	F = F0 + 2 MHz	—	—	–25	dB
	F = F0 –2 MHz	—	—	–33	dB
	F = F0 + 3 MHz	—	—	–25	dB
	F = F0 –3 MHz	—	—	–45	dB
Out-of-band blocking performance	30 MHz ~ 2000 MHz	–10	—	—	dBm
	2000 MHz ~ 2400 MHz	–27	—	—	dBm
	2500 MHz ~ 3000 MHz	–27	—	—	dBm
	3000 MHz ~ 12.5 GHz	–10	—	—	dBm
Intermodulation	—	–36	—	—	dBm

5.7.2 Transmitter –Basic Data Rate

Table 5-8. Transmitter Characteristics –Basic Data Rate

Parameter	Description	Min	Typ	Max	Unit
RF transmit power ^{note1}	—	—	0	—	dBm
Gain control step	—	—	3	—	dB
RF power control range	—	–12	—	+9	dBm
+20 dB bandwidth	—	—	0.9	—	MHz
Adjacent channel transmit power	F = F0 ± 2 MHz	—	–47	—	dBm
	F = F0 ± 3 MHz	—	–55	—	dBm
	F = F0 ± > 3 MHz	—	–60	—	dBm

Parameter	Description	Min	Typ	Max	Unit
$\Delta\ f1_{\text{avg}}$	—	—	—	155	kHz
$\Delta\ f2_{\text{max}}$	—	133.7	—	—	kHz
$\Delta\ f2_{\text{avg}}/\Delta\ f1_{\text{avg}}$	—	—	0.92	—	—
ICFT	—	—	–7	—	kHz
Drift rate	—	—	0.7	—	kHz/50 μs
Drift (DH1)	—	—	6	—	kHz
Drift (DH5)	—	—	6	—	kHz

1. There are in total eight power levels from level 0 to level 7, with transmit power ranging from –12 dBm to 9 dBm. When the power level rises by 1, the transmit power increases by 3 dB. Power level 4 is used by default and the corresponding transmit power is 0 dBm.

5.7.3 Receiver –Enhanced Data Rate

Table 5-9. Receiver Characteristics –Enhanced Data Rate

Parameter	Description	Min	Typ	Max	Unit
$\pi/4$ DQPSK					
Sensitivity @0.01% BER	—	–90	–89	–88	dBm
Maximum received signal @0.01% BER	—	—	0	—	dBm
Co-channel C/I	—	—	11	—	dB
Adjacent channel selectivity C/I	F = F0 + 1 MHz	—	–7	—	dB
	F = F0 –1 MHz	—	–7	—	dB
	F = F0 + 2 MHz	—	–25	—	dB
	F = F0 –2 MHz	—	–35	—	dB
	F = F0 + 3 MHz	—	–25	—	dB
	F = F0 –3 MHz	—	–45	—	dB
8DPSK					
Sensitivity @0.01% BER	—	–84	–83	–82	dBm
Maximum received signal @0.01% BER	—	—	–5	—	dBm
C/I c-channel	—	—	18	—	dB
Adjacent channel selectivity C/I	F = F0 + 1 MHz	—	2	—	dB
	F = F0 –1 MHz	—	2	—	dB
	F = F0 + 2 MHz	—	–25	—	dB
	F = F0 –2 MHz	—	–25	—	dB
	F = F0 + 3 MHz	—	–25	—	dB
	F = F0 –3 MHz	—	–38	—	dB

5.7.4 Transmitter –Enhanced Data Rate

Table 5-10. Transmitter Characteristics –Enhanced Data Rate

Parameter	Description	Min	Typ	Max	Unit
RF transmit power (see note under Table 5-10)	—	—	0	—	dBm

Parameter	Description	Min	Typ	Max	Unit
Gain control step	—	—	3	—	dB
RF power control range	—	–12	—	+9	dBm
$\pi/4$ DQPSK max w0	—	—	–0.72	—	kHz
$\pi/4$ DQPSK max wi	—	—	–6	—	kHz
$\pi/4$ DQPSK max |wi + w0|	—	—	–7.42	—	kHz
8DPSK max w0	—	—	0.7	—	kHz
8DPSK max wi	—	—	–9.6	—	kHz
8DPSK max |wi + w0|	—	—	–10	—	kHz
$\pi/4$ DQPSK modulation accuracy	RMS DEVM	—	4.28	—	%
	99% DEVM	—	100	—	%
	Peak DEVM	—	13.3	—	%
8 DPSK modulation accuracy	RMS DEVM	—	5.8	—	%
	99% DEVM	—	100	—	%
	Peak DEVM	—	14	—	%
In-band spurious emissions	F = F0 ± 1 MHz	—	–46	—	dBm
	F = F0 ± 2 MHz	—	–40	—	dBm
	F = F0 ± 3 MHz	—	–46	—	dBm
	F = F0 +/–> 3 MHz	—	—	–53	dBm
EDR differential phase coding	—	—	100	—	%

5.8 Bluetooth LE Radio

5.8.1 Receiver

Table 5-11. Receiver Characteristics –Bluetooth LE

Parameter	Description	Min	Typ	Max	Unit
Sensitivity @30.8% PER	—	–94	–93	–92	dBm
Maximum received signal @30.8% PER	—	0	—	—	dBm
Co-channel C/I	—	—	+10	—	dB
Adjacent channel selectivity C/I	F = F0 + 1 MHz	—	–5	—	dB
	F = F0 –1 MHz	—	–5	—	dB
	F = F0 + 2 MHz	—	–25	—	dB
	F = F0 –2 MHz	—	–35	—	dB
	F = F0 + 3 MHz	—	–25	—	dB
	F = F0 –3 MHz	—	–45	—	dB
Out-of-band blocking performance	30 MHz ~ 2000 MHz	–10	—	—	dBm
	2000 MHz ~ 2400 MHz	–27	—	—	dBm
	2500 MHz ~ 3000 MHz	–27	—	—	dBm
	3000 MHz ~ 12.5 GHz	–10	—	—	dBm
Intermodulation	—	–36	—	—	dBm

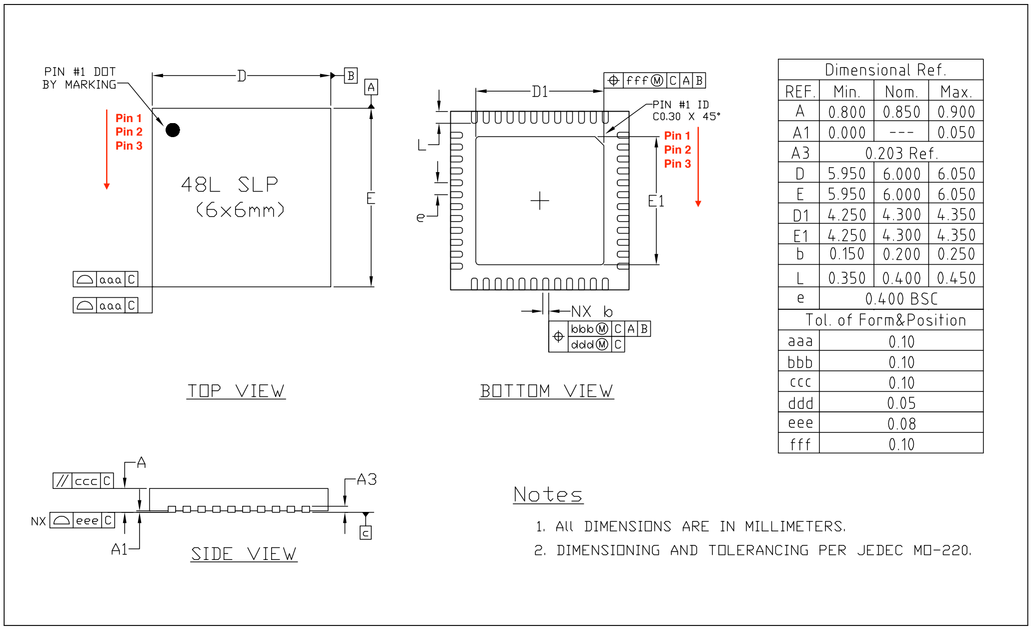
5.8.2 Transmitter

Table 5-12. Transmitter Characteristics –Bluetooth LE

Parameter	Description	Min	Typ	Max	Unit
RF transmit power (see note under Table 5-8)	—	—	0	—	dBm
Gain control step	—	—	3	—	dB
RF power control range	—	–12	—	+9	dBm
Adjacent channel transmit power	F = F0 ± 2 MHz	—	–52	—	dBm
	F = F0 ± 3 MHz	—	–58	—	dBm
	F = F0 ± > 3 MHz	—	–60	—	dBm
$\Delta\ f1_{\text{avg}}$	—	—	—	265	kHz
$\Delta\ f2_{\text{max}}$	—	247	—	—	kHz
$\Delta\ f2_{\text{avg}}/\Delta\ f1_{\text{avg}}$	—	—	0.92	—	—
ICFT	—	—	–10	—	kHz
Drift rate	—	—	0.7	—	kHz/50 μs
Drift	—	—	2	—	kHz

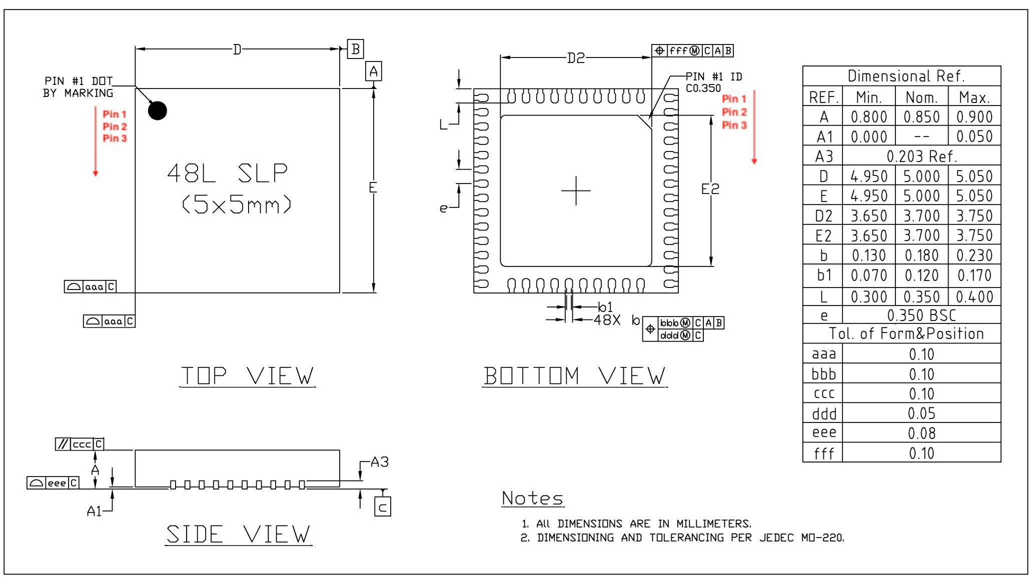
6 Packaging

- For information about tape, reel, and chip marking, please refer to *Espressif Chip Packaging Information*.
- The pins of the chip are numbered in anti-clockwise order starting from Pin 1 in the top view. For pin numbers and pin names, see also pin layout figures in Section 2.1 Pin Layout.



Dimensional Ref.			
REF.	Min.	Nom.	Max.
A	0.800	0.850	0.900
A1	0.000	---	0.050
A3	0.203 Ref.		
D	5.950	6.000	6.050
E	5.950	6.000	6.050
D1	4.250	4.300	4.350
E1	4.250	4.300	4.350
b	0.150	0.200	0.250
L	0.350	0.400	0.450
e	0.400 BSC		
Tol. of Form&Position			
aaa	0.10		
bbb	0.10		
ccc	0.10		
ddd	0.05		
eee	0.08		
fff	0.10		

Figure 6-1. QFN48 (6×6 mm) Package



Dimensional Ref.			
REF.	Min.	Nom.	Max.
A	0.800	0.850	0.900
A1	0.000	--	0.050
A3	0.203 Ref.		
D	4.950	5.000	5.050
E	4.950	5.000	5.050
D2	3.650	3.700	3.750
E2	3.650	3.700	3.750
b	0.130	0.180	0.230
b1	0.070	0.120	0.170
L	0.300	0.350	0.400
e	0.350 BSC		
Tol. of Form&Position			
aaa	0.10		
bbb	0.10		
ccc	0.10		
ddd	0.05		
eee	0.08		
fff	0.10		

Figure 6-2. QFN48 (5×5 mm) Package

Related Documentation and Resources

Related Documentation

- ESP32 Technical Reference Manual – Detailed information on how to use the ESP32 memory and peripherals.
- ESP32 Hardware Design Guidelines – Guidelines on how to integrate the ESP32 into your hardware product.
- ESP32 ECO and Workarounds for Bugs – Correction of ESP32 design errors.
- *Certificates*
  https://espressif.com/en/support/documents/certificates
- *ESP32 Product/Process Change Notifications (PCN)*
  https://espressif.com/en/support/documents/pcns
- *ESP32 Advisories* – Information on security, bugs, compatibility, component reliability.
  https://espressif.com/en/support/documents/advisories
- *Documentation Updates and Update Notification Subscription*
  https://espressif.com/en/support/download/documents

Developer Zone

- ESP-IDF Programming Guide for ESP32 – Extensive documentation for the ESP-IDF development framework.
- *ESP-IDF* and other development frameworks on GitHub.
  https://github.com/espressif
- *ESP32 BBS Forum* – Engineer-to-Engineer (E2E) Community for Espressif products where you can post questions, share knowledge, explore ideas, and help solve problems with fellow engineers.
  https://esp32.com/
- *The ESP Journal* – Best Practices, Articles, and Notes from Espressif folks.
  https://blog.espressif.com/
- See the tabs *SDKs and Demos*, *Apps*, *Tools*, *AT Firmware*.
  https://espressif.com/en/support/download/sdks-demos

Products

- *ESP32 Series SoCs* – Browse through all ESP32 SoCs.
  https://espressif.com/en/products/socs?id=ESP32
- *ESP32 Series Modules* – Browse through all ESP32-based modules.
  https://espressif.com/en/products/modules?id=ESP32
- *ESP32 Series DevKits* – Browse through all ESP32-based devkits.
  https://espressif.com/en/products/devkits?id=ESP32
- *ESP Product Selector* – Find an Espressif hardware product suitable for your needs by comparing or applying filters.
  https://products.espressif.com/#/product-selector?language=en

Contact Us

- See the tabs *Sales Questions*, *Technical Enquiries*, *Circuit Schematic & PCB Design Review*, *Get Samples* (Online stores), *Become Our Supplier*, *Comments & Suggestions*.
  https://espressif.com/en/contact-us/sales-questions

Appendix A –ESP32 Pin Lists

A.1. Notes on ESP32 Pin Lists

Table 6-1. Notes on ESP32 Pin Lists

No.	Description
1	In Table IO_MUX, the boxes highlighted in yellow indicate the GPIO pins that are input-only. Please see the following note for further details.
2	GPIO pins 34-39 are input-only. These pins do not feature an output driver or internal pull-up/pull-down circuitry. The pin names are: SENSOR_VP (GPIO36), SENSOR_CAPP (GPIO37), SENSOR_CAPN (GPIO38), SENSOR_VN (GPIO39), VDET_1 (GPIO34), VDET_2 (GPIO35).
3	The pins are grouped into four power domains: VDDA (analog power supply), VDD3P3_RTC (RTC power supply), VDD3P3_CPU (power supply of digital IOs and CPU cores), VDD_SDIO (power supply of SDIO IOs). VDD_SDIO is the output of the internal SDIO-LDO. The voltage of SDIO-LDO can be configured at 1.8 V or be the same as that of VDD3P3_RTC. The strapping pin and eFuse bits determine the default voltage of the SDIO-LDO. Software can change the voltage of the SDIO-LDO by configuring register bits. For details, please see the column "Power Domain" in Table IO_MUX.
4	The functional pins in the VDD3P3_RTC domain are those with analog functions, including the 32 kHz crystal oscillator, ADC, DAC, and the capacitive touch sensor. Please see columns "Analog Function 0 ~ 2" in Table IO_MUX.
5	These VDD3P3_RTC pins support the RTC function, and can work during Deep-sleep. For example, an RTC-GPIO can be used for waking up the chip from Deep-sleep.
6	The GPIO pins support up to six digital functions, as shown in columns "Function 0 ~ 5" In Table IO_MUX. The function selection registers will be set as "*N*", where *N* is the function number. Below are some definitions: • SD_* is for signals of the SDIO slave. • HS1_* is for Port 1 signals of the SDIO host. • HS2_* is for Port 2 signals of the SDIO host. • MT* is for signals of the JTAG. • U0* is for signals of the UART0 module. • U1* is for signals of the UART1 module. • U2* is for signals of the UART2 module. • SPI* is for signals of the SPI01 module. • HSPI* is for signals of the SPI2 module. • VSPI* is for signals of the SPI3 module.

No.	Description
7	Each column about digital "Function" is accompanied by a column about "Type". Please see the following explanations for the meanings of "type" with respect to each "function" they are associated with. For each "Function-*N*", "type" signifies: • I: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is still from this pin. • I1: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is always "1". • I0: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is always "0". • O: output only. • T: high-impedance. • I/O/T: combinations of input, output, and high-impedance according to the function signal. • I1/O/T: combinations of input, output, and high-impedance, according to the function signal. If a function is not selected, the input signal of the function is "1". For example, pin 30 can function as HS1_CMD or SD_CMD, where HS1_CMD is of an "I1/O/T" type. If pin 30 is selected as HS1_CMD, this pin's input and output are controlled by the SDIO host. If pin 30 is not selected as HS1_CMD, the input signal of the SDIO host is always "1".
8	Each digital output pin is associated with its configurable drive strength. Column "Drive Strength" in Table IO_MUX lists the default values. The drive strength of the digital output pins can be configured into one of the following four options: • 0: ~5 mA • 1: ~10 mA • 2: ~20 mA • 3: ~40 mA The default value is 2. The drive strength of the internal pull-up (wpu) and pull-down (wpd) is ~75 μA.
9	Column "At Reset" in Table IO_MUX lists the status of each pin during reset, including input-enable (ie=1), internal pull-up (wpu) and internal pull-down (wpd). During reset, all pins are output-disabled.
10	Column "After Reset" in Table IO_MUX lists the status of each pin immediately after reset, including input-enable (ie=1), internal pull-up (wpu) and internal pull-down (wpd). After reset, each pin is set to "Function 0". The output-enable is controlled by digital Function 0.
11	Table Ethernet_MAC is about the signal mapping inside Ethernet MAC. The Ethernet MAC supports MII and RMII interfaces, and supports both the internal PLL clock and the external clock source. For the MII interface, the Ethernet MAC is with/without the TX_ERR signal. MDC, MDIO, CRS and COL are slow signals, and can be mapped onto any GPIO pin through the GPIO-Matrix.
12	Table GPIO Matrix is for the GPIO-Matrix. The signals of the on-chip functional modules can be mapped onto any GPIO pin. Some signals can be mapped onto a pin by both IO-MUX and GPIO-Matrix, as shown in the column tagged as "Same input signal from IO_MUX core" in Table GPIO Matrix.

No.	Description
13	*In Table GPIO_Matrix, the column "Default Value if unassigned" records the default value of the an input signal if no GPIO is assigned to it. The actual value is determined by register GPIO_FUNC*m*_IN_INV_SEL and GPIO_FUNC*m*_IN_SEL. (The value of *m* ranges from 1 to 255.)

A.2. GPIO_Matrix

Table 6-2. GPIO_Matrix

Signal No.	Input Signals	Default Value If Unassigned*	Same Input Signal from IO_MUX Core	Output Signals	Output Enable of Output Signals
0	SPICLK_in	0	yes	SPICLK_out	SPICLK_oe
1	SPIQ_in	0	yes	SPIQ_out	SPIQ_oe
2	SPID_in	0	yes	SPID_out	SPID_oe
3	SPIHD_in	0	yes	SPIHD_out	SPIHD_oe
4	SPIWP_in	0	yes	SPIWP_out	SPIWP_oe
5	SPICS0_in	0	yes	SPICS0_out	SPICS0_oe
6	SPICS1_in	0	no	SPICS1_out	SPICS1_oe
7	SPICS2_in	0	no	SPICS2_out	SPICS2_oe
8	HSPICLK_in	0	yes	HSPICLK_out	HSPICLK_oe
9	HSPIQ_in	0	yes	HSPIQ_out	HSPIQ_oe
10	HSPID_in	0	yes	HSPID_out	HSPID_oe
11	HSPICS0_in	0	yes	HSPICS0_out	HSPICS0_oe
12	HSPIHD_in	0	yes	HSPIHD_out	HSPIHD_oe
13	HSPIWP_in	0	yes	HSPIWP_out	HSPIWP_oe
14	U0RXD_in	0	yes	U0TXD_out	1'd1
15	U0CTS_in	0	yes	U0RTS_out	1'd1
16	U0DSR_in	0	no	U0DTR_out	1'd1
17	U1RXD_in	0	yes	U1TXD_out	1'd1
18	U1CTS_in	0	yes	U1RTS_out	1'd1
23	I2S0O_BCK_in	0	no	I2S0O_BCK_out	1'd1
24	I2S1O_BCK_in	0	no	I2S1O_BCK_out	1'd1
25	I2S0O_WS_in	0	no	I2S0O_WS_out	1'd1
26	I2S1O_WS_in	0	no	I2S1O_WS_out	1'd1
27	I2S0I_BCK_in	0	no	I2S0I_BCK_out	1'd1
28	I2S0I_WS_in	0	no	I2S0I_WS_out	1'd1
29	I2CEXT0_SCL_in	1	no	I2CEXT0_SCL_out	1'd1
30	I2CEXT0_SDA_in	1	no	I2CEXT0_SDA_out	1'd1
31	pwm0_sync0_in	0	no	sdio_tohost_int_out	1'd1
32	pwm0_sync1_in	0	no	pwm0_out0a	1'd1
33	pwm0_sync2_in	0	no	pwm0_out0b	1'd1
34	pwm0_f0_in	0	no	pwm0_out1a	1'd1

Signal No.	Input Signals	Default Value If Unassigned*	Same Input Signal from IO_MUX Core	Output Signals	Output Enable of Output Signals
35	pwm0_f1_in	0	no	pwm0_out1b	1'd1
36	pwm0_f2_in	0	no	pwm0_out2a	1'd1
37	—	0	no	pwm0_out2b	1'd1
39	pcnt_sig_ch0_in0	0	no	—	1'd1
40	pcnt_sig_ch1_in0	0	no	—	1'd1
41	pcnt_ctrl_ch0_in0	0	no	—	1'd1
42	pcnt_ctrl_ch1_in0	0	no	—	1'd1
43	pcnt_sig_ch0_in1	0	no	—	1'd1
44	pcnt_sig_ch1_in1	0	no	—	1'd1
45	pcnt_ctrl_ch0_in1	0	no	—	1'd1
46	pcnt_ctrl_ch1_in1	0	no	—	1'd1
47	pcnt_sig_ch0_in2	0	no	—	1'd1
48	pcnt_sig_ch1_in2	0	no	—	1'd1
49	pcnt_ctrl_ch0_in2	0	no	—	1'd1
50	pcnt_ctrl_ch1_in2	0	no	—	1'd1
51	pcnt_sig_ch0_in3	0	no	—	1'd1
52	pcnt_sig_ch1_in3	0	no	—	1'd1
53	pcnt_ctrl_ch0_in3	0	no	—	1'd1
54	pcnt_ctrl_ch1_in3	0	no	—	1'd1
55	pcnt_sig_ch0_in4	0	no	—	1'd1
56	pcnt_sig_ch1_in4	0	no	—	1'd1
57	pcnt_ctrl_ch0_in4	0	no	—	1'd1
58	pcnt_ctrl_ch1_in4	0	no	—	1'd1
61	HSPICS1_in	0	no	HSPICS1_out	HSPICS1_oe
62	HSPICS2_in	0	no	HSPICS2_out	HSPICS2_oe
63	VSPICLK_in	0	yes	VSPICLK_out_mux	VSPICLK_oe
64	VSPIQ_in	0	yes	VSPIQ_out	VSPIQ_oe
65	VSPID_in	0	yes	VSPID_out	VSPID_oe
66	VSPIHD_in	0	yes	VSPIHD_out	VSPIHD_oe
67	VSPIWP_in	0	yes	VSPIWP_out	VSPIWP_oe
68	VSPICS0_in	0	yes	VSPICS0_out	VSPICS0_oe
69	VSPICS1_in	0	no	VSPICS1_out	VSPICS1_oe
70	VSPICS2_in	0	no	VSPICS2_out	VSPICS2_oe
71	pcnt_sig_ch0_in5	0	no	ledc_hs_sig_out0	1'd1
72	pcnt_sig_ch1_in5	0	no	ledc_hs_sig_out1	1'd1
73	pcnt_ctrl_ch0_in5	0	no	ledc_hs_sig_out2	1'd1
74	pcnt_ctrl_ch1_in5	0	no	ledc_hs_sig_out3	1'd1
75	pcnt_sig_ch0_in6	0	no	ledc_hs_sig_out4	1'd1
76	pcnt_sig_ch1_in6	0	no	ledc_hs_sig_out5	1'd1
77	pcnt_ctrl_ch0_in6	0	no	ledc_hs_sig_out6	1'd1
78	pcnt_ctrl_ch1_in6	0	no	ledc_hs_sig_out7	1'd1

Signal No.	Input Signals	Default Value If Unassigned*	Same Input Signal from IO_MUX Core	Output Signals	Output Enable of Output Signals
79	pcnt_sig_ch0_in7	0	no	ledc_ls_sig_out0	1'd1
80	pcnt_sig_ch1_in7	0	no	ledc_ls_sig_out1	1'd1
81	pcnt_ctrl_ch0_in7	0	no	ledc_ls_sig_out2	1'd1
82	pcnt_ctrl_ch1_in7	0	no	ledc_ls_sig_out3	1'd1
83	rmt_sig_in0	0	no	ledc_ls_sig_out4	1'd1
84	rmt_sig_in1	0	no	ledc_ls_sig_out5	1'd1
85	rmt_sig_in2	0	no	ledc_ls_sig_out6	1'd1
86	rmt_sig_in3	0	no	ledc_ls_sig_out7	1'd1
87	rmt_sig_in4	0	no	rmt_sig_out0	1'd1
88	rmt_sig_in5	0	no	rmt_sig_out1	1'd1
89	rmt_sig_in6	0	no	rmt_sig_out2	1'd1
90	rmt_sig_in7	0	no	rmt_sig_out3	1'd1
91	—	—	—	rmt_sig_out4	1'd1
92	—	—	—	rmt_sig_out6	1'd1
94	twai_rx	1	no	rmt_sig_out7	1'd1
95	I2CEXT1_SCL_in	1	no	I2CEXT1_SCL_out	1'd1
96	I2CEXT1_SDA_in	1	no	I2CEXT1_SDA_out	1'd1
97	host_card_detect_n_1	0	no	host_ccmd_od_pullup_en_n	1'd1
98	host_card_detect_n_2	0	no	host_rst_n_1	1'd1
99	host_card_write_prt_1	0	no	host_rst_n_2	1'd1
100	host_card_write_prt_2	0	no	gpio_sd0_out	1'd1
101	host_card_int_n_1	0	no	gpio_sd1_out	1'd1
102	host_card_int_n_2	0	no	gpio_sd2_out	1'd1
103	pwm1_sync0_in	0	no	gpio_sd3_out	1'd1
104	pwm1_sync1_in	0	no	gpio_sd4_out	1'd1
105	pwm1_sync2_in	0	no	gpio_sd5_out	1'd1
106	pwm1_f0_in	0	no	gpio_sd6_out	1'd1
107	pwm1_f1_in	0	no	gpio_sd7_out	1'd1
108	pwm1_f2_in	0	no	pwm1_out0a	1'd1
109	pwm0_cap0_in	0	no	pwm1_out0b	1'd1
110	pwm0_cap1_in	0	no	pwm1_out1a	1'd1
111	pwm0_cap2_in	0	no	pwm1_out1b	1'd1
112	pwm1_cap0_in	0	no	pwm1_out2a	1'd1
113	pwm1_cap1_in	0	no	pwm1_out2b	1'd1
114	pwm1_cap2_in	0	no	pwm2_out1h	1'd1
115	pwm2_flta	1	no	pwm2_out1l	1'd1
116	pwm2_fltb	1	no	pwm2_out2h	1'd1
117	pwm2_cap1_in	0	no	pwm2_out2l	1'd1
118	pwm2_cap2_in	0	no	pwm2_out3h	1'd1
119	pwm2_cap3_in	0	no	pwm2_out3l	1'd1
120	pwm3_flta	1	no	pwm2_out4h	1'd1

Signal No.	Input Signals	Default Value If Unassigned*	Same Input Signal from IO_MUX Core	Output Signals	Output Enable of Output Signals
121	pwm3_fltb	1	no	pwm2_out4l	1'd1
122	pwm3_cap1_in	0	no	—	1'd1
123	pwm3_cap2_in	0	no	twai_tx	1'd1
124	pwm3_cap3_in	0	no	twai_bus_off_on	1'd1
125	—	—	—	twai_clkout	1'd1
140	I2S0I_DATA_in0	0	no	I2S0O_DATA_out0	1'd1
141	I2S0I_DATA_in1	0	no	I2S0O_DATA_out1	1'd1
142	I2S0I_DATA_in2	0	no	I2S0O_DATA_out2	1'd1
143	I2S0I_DATA_in3	0	no	I2S0O_DATA_out3	1'd1
144	I2S0I_DATA_in4	0	no	I2S0O_DATA_out4	1'd1
145	I2S0I_DATA_in5	0	no	I2S0O_DATA_out5	1'd1
146	I2S0I_DATA_in6	0	no	I2S0O_DATA_out6	1'd1
147	I2S0I_DATA_in7	0	no	I2S0O_DATA_out7	1'd1
148	I2S0I_DATA_in8	0	no	I2S0O_DATA_out8	1'd1
149	I2S0I_DATA_in9	0	no	I2S0O_DATA_out9	1'd1
150	I2S0I_DATA_in10	0	no	I2S0O_DATA_out10	1'd1
151	I2S0I_DATA_in11	0	no	I2S0O_DATA_out11	1'd1
152	I2S0I_DATA_in12	0	no	I2S0O_DATA_out12	1'd1
153	I2S0I_DATA_in13	0	no	I2S0O_DATA_out13	1'd1
154	I2S0I_DATA_in14	0	no	I2S0O_DATA_out14	1'd1
155	I2S0I_DATA_in15	0	no	I2S0O_DATA_out15	1'd1
156	—	—	—	I2S0O_DATA_out16	1'd1
157	—	—	—	I2S0O_DATA_out17	1'd1
158	—	—	—	I2S0O_DATA_out18	1'd1
159	—	—	—	I2S0O_DATA_out19	1'd1
160	—	—	—	I2S0O_DATA_out20	1'd1
161	—	—	—	I2S0O_DATA_out21	1'd1
162	—	—	—	I2S0O_DATA_out22	1'd1
163	—	—	—	I2S0O_DATA_out23	1'd1
164	I2S1I_BCK_in	0	no	I2S1I_BCK_out	1'd1
165	I2S1I_WS_in	0	no	I2S1I_WS_out	1'd1
166	I2S1I_DATA_in0	0	no	I2S1O_DATA_out0	1'd1
167	I2S1I_DATA_in1	0	no	I2S1O_DATA_out1	1'd1
168	I2S1I_DATA_in2	0	no	I2S1O_DATA_out2	1'd1
169	I2S1I_DATA_in3	0	no	I2S1O_DATA_out3	1'd1
170	I2S1I_DATA_in4	0	no	I2S1O_DATA_out4	1'd1
171	I2S1I_DATA_in5	0	no	I2S1O_DATA_out5	1'd1
172	I2S1I_DATA_in6	0	no	I2S1O_DATA_out6	1'd1
173	I2S1I_DATA_in7	0	no	I2S1O_DATA_out7	1'd1
174	I2S1I_DATA_in8	0	no	I2S1O_DATA_out8	1'd1
175	I2S1I_DATA_in9	0	no	I2S1O_DATA_out9	1'd1

Signal No.	Input Signals	Default Value If Unassigned*	Same Input Signal from IO_MUX Core	Output Signals	Output Enable of Output Signals
176	I2S1I_DATA_in10	0	no	I2S1O_DATA_out10	1'd1
177	I2S1I_DATA_in11	0	no	I2S1O_DATA_out11	1'd1
178	I2S1I_DATA_in12	0	no	I2S1O_DATA_out12	1'd1
179	I2S1I_DATA_in13	0	no	I2S1O_DATA_out13	1'd1
180	I2S1I_DATA_in14	0	no	I2S1O_DATA_out14	1'd1
181	I2S1I_DATA_in15	0	no	I2S1O_DATA_out15	1'd1
182	—	—	—	I2S1O_DATA_out16	1'd1
183	—	—	—	I2S1O_DATA_out17	1'd1
184	—	—	—	I2S1O_DATA_out18	1'd1
185	—	—	—	I2S1O_DATA_out19	1'd1
186	—	—	—	I2S1O_DATA_out20	1'd1
187	—	—	—	I2S1O_DATA_out21	1'd1
188	—	—	—	I2S1O_DATA_out22	1'd1
189	—	—	—	I2S1O_DATA_out23	1'd1
190	I2S0I_H_SYNC	0	no	pwm3_out1h	1'd1
191	I2S0I_V_SYNC	0	no	pwm3_out1l	1'd1
192	I2S0I_H_ENABLE	0	no	pwm3_out2h	1'd1
193	I2S1I_H_SYNC	0	no	pwm3_out2l	1'd1
194	I2S1I_V_SYNC	0	no	pwm3_out3h	1'd1
195	I2S1I_H_ENABLE	0	no	pwm3_out3l	1'd1
196	—	—	—	pwm3_out4h	1'd1
197	—	—	—	pwm3_out4l	1'd1
198	U2RXD_in	0	yes	U2TXD_out	1'd1
199	U2CTS_in	0	yes	U2RTS_out	1'd1
200	emac_mdc_i	0	no	emac_mdc_o	emac_mdc_oe
201	emac_mdi_i	0	no	emac_mdo_o	emac_mdo_o_e
202	emac_crs_i	0	no	emac_crs_o	emac_crs_oe
203	emac_col_i	0	no	emac_col_o	emac_col_oe
204	pcmfsync_in	0	no	bt_audio0_irq	1'd1
205	pcmclk_in	0	no	bt_audio1_irq	1'd1
206	pcmdin	0	no	bt_audio2_irq	1'd1
207	—	—	—	ble_audio0_irq	1'd1
208	—	—	—	ble_audio1_irq	1'd1
209	—	—	—	ble_audio2_irq	1'd1
210	—	—	—	pcmfsync_out	pcmfsync_en
211	—	—	—	pcmclk_out	pcmclk_en
212	—	—	—	pcmdout	pcmdout_en
213	—	—	—	ble_audio_sync0_p	1'd1
214	—	—	—	ble_audio_sync1_p	1'd1
215	—	—	—	ble_audio_sync2_p	1'd1
224	—	—	—	sig_in_func224	1'd1

Signal No.	Input Signals	Default Value If Unassigned*	Same Input Signal from IO_MUX Core	Output Signals	Output Enable of Output Signals
225	—	—	—	sig_in_func225	1'd1
226	—	—	—	sig_in_func226	1'd1
227	—	—	—	sig_in_func227	1'd1
228	—	—	—	sig_in_func228	1'd1

A.3. Ethernet_MAC

Table 6-3. Ethernet_MAC

Pin Name	Function6	MII (int_osc)	MII (ext_osc)	RMII (int_osc)	RMII (ext_osc)
GPIO0	EMAC_TX_CLK	TX_CLK (I)	TX_CLK (I)	CLK_OUT(O)	EXT_OSC_CLK(I)
GPIO5	EMAC_RX_CLK	RX_CLK (I)	RX_CLK (I)	—	—
GPIO21	EMAC_TX_EN	TX_EN(O)	TX_EN(O)	TX_EN(O)	TX_EN(O)
GPIO19	EMAC_TXD0	TXD[0](O)	TXD[0](O)	TXD[0](O)	TXD[0](O)
GPIO22	EMAC_TXD1	TXD[1](O)	TXD[1](O)	TXD[1](O)	TXD[1](O)
MTMS	EMAC_TXD2	TXD[2](O)	TXD[2](O)	—	—
MTDI	EMAC_TXD3	TXD[3](O)	TXD[3](O)	—	—
MTCK	EMAC_RX_ER	RX_ER(I)	RX_ER(I)	—	—
GPIO27	EMAC_RX_DV	RX_DV(I)	RX_DV(I)	CRS_DV(I)	CRS_DV(I)
GPIO25	EMAC_RXD0	RXD[0](I)	RXD[0](I)	RXD[0](I)	RXD[0](I)
GPIO26	EMAC_RXD1	RXD[1](I)	RXD[1](I)	RXD[1](I)	RXD[1](I)
U0TXD	EMAC_RXD2	RXD[2](I)	RXD[2](I)	—	—
MTDO	EMAC_RXD3	RXD[3](I)	RXD[3](I)	—	—
GPIO16	EMAC_CLK_OUT	CLK_OUT(O)	—	CLK_OUT(O)	—
GPIO17	EMAC_CLK_OUT_180	CLK_OUT_180(O)	—	CLK_OUT_180(O)	—
GPIO4	EMAC_TX_ER	TX_ERR(O)*	TX_ERR(O)*	—	—
In GPIO Matrix*	—	MDC(O)	MDC(O)	MDC(O)	MDC(O)
In GPIO Matrix*	—	MDIO(IO)	MDIO(IO)	MDIO(IO)	MDIO(IO)
In GPIO Matrix*	—	CRS(I)	CRS(I)	—	—
In GPIO Matrix*	—	COL(I)	COL(I)	—	—
*Notes: 1. The GPIO Matrix can be any GPIO. 2. The TX_ERR (O) is optional.					

A.4. IO_MUX

For the list of IO_MUX pins, please see the next page.

IO_MUX

Pin No.	Power Supply Pin	Analog Pin	Digital Pin	Power Domain	Analog Function0	Analog Function1	Analog Function2	RTC Function0	RTC Function1	Function0	Type	Function1	Type	Function2	Type	Function3	Type	Function4	Type	Function5	Type	Drive Strength (2'd2: 20 mA)	At Reset	After Reset
1	VDDA			VDDA supply in																				
2		LNA_IN		VDD3P3																				
3	VDD3P3			VDD3P3 supply in																				
4	VDD3P3			VDD3P3 supply in																				
5		SENSOR_VP		VDD3P3_RTC		ADC1_CH0		RTC_GPIO0		GPIO36	I			GPIO36	I								oe=0, ie=0	oe=0, ie=0
6		SENSOR_CAPP		VDD3P3_RTC		ADC1_CH1		RTC_GPIO1		GPIO37	I			GPIO37	I								oe=0, ie=0	oe=0, ie=0
7		SENSOR_CAPN		VDD3P3_RTC		ADC1_CH2		RTC_GPIO2		GPIO38	I			GPIO38	I								oe=0, ie=0	oe=0, ie=0
8		SENSOR_VN		VDD3P3_RTC		ADC1_CH3		RTC_GPIO3		GPIO39	I			GPIO39	I								oe=0, ie=0	oe=0, ie=0
9		CHIP_PU		VDD3P3_RTC																				
10		VDET_1		VDD3P3_RTC		ADC1_CH6		RTC_GPIO4		GPIO34	I			GPIO34	I								oe=0, ie=0	oe=0, ie=0
11		VDET_2		VDD3P3_RTC		ADC1_CH7		RTC_GPIO5		GPIO35	I			GPIO35	I								oe=0, ie=0	oe=0, ie=0
12		32K_XP		VDD3P3_RTC	XTAL_32K_P	ADC1_CH4	TOUCH9	RTC_GPIO9		GPIO32	I/O/T			GPIO32	I/O/T							2'd2	oe=0, ie=0	oe=0, ie=0
13		32K_XN		VDD3P3_RTC	XTAL_32K_N	ADC1_CH5	TOUCH8	RTC_GPIO8		GPIO33	I/O/T			GPIO33	I/O/T							2'd2	oe=0, ie=0	oe=0, ie=0
14			GPIO25	VDD3P3_RTC	DAC_1	ADC2_CH8		RTC_GPIO6		GPIO25	I/O/T			GPIO25	I/O/T					EMAC_RXD0	I	2'd2	oe=0, ie=0	oe=0, ie=0
15			GPIO26	VDD3P3_RTC	DAC_2	ADC2_CH9		RTC_GPIO7		GPIO26	I/O/T			GPIO26	I/O/T					EMAC_RXD1	I	2'd2	oe=0, ie=0	oe=0, ie=0
16			GPIO27	VDD3P3_RTC		ADC2_CH7	TOUCH7	RTC_GPIO17		GPIO27	I/O/T			GPIO27	I/O/T					EMAC_RX_DV	I	2'd2	oe=0, ie=0	oe=0, ie=0
17			MTMS	VDD3P3_RTC		ADC2_CH6	TOUCH6	RTC_GPIO16		MTMS	I0	HSPICLK	I/O/T	GPIO14	I/O/T	HS2_CLK	O	SD_CLK	I0	EMAC_TXD2	O	2'd2	oe=0, ie=0	oe=0, ie=1, wpu
18			MTDI	VDD3P3_RTC		ADC2_CH5	TOUCH5	RTC_GPIO15		MTDI	I1	HSPIQ	I/O/T	GPIO12	I/O/T	HS2_DATA2	I1/O/T	SD_DATA2	I1/O/T	EMAC_TXD3	O	2'd2	oe=0, ie=1, wpd	oe=0, ie=1, wpd
19	VDD3P3_RTC			VDD3P3_RTC supply in																				
20			MTCK	VDD3P3_RTC		ADC2_CH4	TOUCH4	RTC_GPIO14		MTCK	I1	HSPID	I/O/T	GPIO13	I/O/T	HS2_DATA3	I1/O/T	SD_DATA3	I1/O/T	EMAC_RX_ER	I	2'd2	oe=0, ie=0	oe=0, ie=1, wpd
21			MTDO	VDD3P3_RTC		ADC2_CH3	TOUCH3	RTC_GPIO13	I2C_SDA	MTDO	O/T	HSPICS0	I/O/T	GPIO15	I/O/T	HS2_CMD	I1/O/T	SD_CMD	I1/O/T	EMAC_RXD3	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
22			GPIO2	VDD3P3_RTC		ADC2_CH2	TOUCH2	RTC_GPIO12	I2C_SCL	GPIO2	I/O/T	HSPIWP	I/O/T	GPIO2	I/O/T	HS2_DATA0	I1/O/T	SD_DATA0	I1/O/T			2'd2	oe=0, ie=1, wpd	oe=0, ie=1, wpd
23			GPIO0	VDD3P3_RTC		ADC2_CH1	TOUCH1	RTC_GPIO11	I2C_SDA	GPIO0	I/O/T	CLK_OUT1	O	GPIO0	I/O/T					EMAC_TX_CLK	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
24			GPIO4	VDD3P3_RTC		ADC2_CH0	TOUCH0	RTC_GPIO10	I2C_SCL	GPIO4	I/O/T	HSPIHD	I/O/T	GPIO4	I/O/T	HS2_DATA1	I1/O/T	SD_DATA1	I1/O/T	EMAC_TX_ER	O	2'd2	oe=0, ie=1, wpd	oe=0, ie=1, wpd
25			GPIO16	VDD_SDIO						GPIO16	I/O/T			GPIO16	I/O/T	HS1_DATA4	I1/O/T	U2RXD	I1	EMAC_CLK_OUT	O	2'd2	oe=0, ie=0	oe=0, ie=1
26	VDD_SDIO			VDD_SDIO supply out/in																				
27			GPIO17	VDD_SDIO						GPIO17	I/O/T			GPIO17	I/O/T	HS1_DATA5	I1/O/T	U2TXD	O	EMAC_CLK_OUT_180	O	2'd2	oe=0, ie=0	oe=0, ie=1
28			SD_DATA_2	VDD_SDIO						SD_DATA2	I1/O/T	SPIHD	I/O/T	GPIO9	I/O/T	HS1_DATA2	I1/O/T	U1RXD	I1			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
29			SD_DATA_3	VDD_SDIO						SD_DATA3	I0/O/T	SPIWP	I/O/T	GPIO10	I/O/T	HS1_DATA3	I1/O/T	U1TXD	O			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
30			SD_CMD	VDD_SDIO						SD_CMD	I1/O/T	SPICS0	I/O/T	GPIO11	I/O/T	HS1_CMD	I1/O/T	U1RTS	O			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
31			SD_CLK	VDD_SDIO						SD_CLK	I0	SPICLK	I/O/T	GPIO6	I/O/T	HS1_CLK	O	U1CTS	I1			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
32			SD_DATA_0	VDD_SDIO						SD_DATA0	I1/O/T	SPIQ	I/O/T	GPIO7	I/O/T	HS1_DATA0	I1/O/T	U2RTS	O			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
33			SD_DATA_1	VDD_SDIO						SD_DATA1	I1/O/T	SPID	I/O/T	GPIO8	I/O/T	HS1_DATA1	I1/O/T	U2CTS	I1			2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
34			GPIO5	VDD3P3_CPU						GPIO5	I/O/T	VSPICS0	I/O/T	GPIO5	I/O/T	HS1_DATA6	I1/O/T			EMAC_RX_CLK	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
35			GPIO18	VDD3P3_CPU						GPIO18	I/O/T	VSPICLK	I/O/T	GPIO18	I/O/T	HS1_DATA7	I1/O/T					2'd2	oe=0, ie=0	oe=0, ie=1
36			GPIO23	VDD3P3_CPU						GPIO23	I/O/T	VSPID	I/O/T	GPIO23	I/O/T	HS1_STROBE	I0					2'd2	oe=0, ie=0	oe=0, ie=1
37	VDD3P3_CPU			VDD3P3_CPU supply in																				
38			GPIO19	VDD3P3_CPU						GPIO19	I/O/T	VSPIQ	I/O/T	GPIO19	I/O/T	U0CTS	I1			EMAC_TXD0	O	2'd2	oe=0, ie=0	oe=0, ie=1
39			GPIO22	VDD3P3_CPU						GPIO22	I/O/T	VSPIWP	I/O/T	GPIO22	I/O/T	U0RTS	O			EMAC_TXD1	O	2'd2	oe=0, ie=0	oe=0, ie=1
40			U0RXD	VDD3P3_CPU						U0RXD	I1	CLK_OUT2	O	GPIO3	I/O/T							2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
41			U0TXD	VDD3P3_CPU						U0TXD	O	CLK_OUT3	O	GPIO1	I/O/T					EMAC_RXD2	I	2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu
42			GPIO21	VDD3P3_CPU						GPIO21	I/O/T	VSPIHD	I/O/T	GPIO21	I/O/T					EMAC_TX_EN	O	2'd2	oe=0, ie=0	oe=0, ie=1
43	VDDA			VDDA supply in																				
44		XTAL_N		VDDA																				
45		XTAL_P		VDDA																				
46	VDDA			VDDA supply in																				
47		CAP2		VDDA																				
48		CAP1		VDDA																				
Total Number	8	14	26																					

Notes:

- wpu: weak pull-up;
- wpd: weak pull-down;
- ie: input enable;
- oe: output enable;
- Please see *Table: Notes on ESP32 Pin Lists* for more information.（请参考表：管脚清单说明。）

Revision History

Date	Version	Release notes
2025.01	v4.8	• Section 3 *Boot Configurations*: Fixed the typo about JTAG signal source control • Section 2.2 *Pin Overview*: Added a note about JTAG interface signals • Table 2-5 *Pin Mapping Between Chip and Flash/PSRAM*: Modified a note about VDD_SDIO
2024.09	v4.7	• Table 5-2 *Recommended Power Supply Characteristics*: Deleted a note about VDD3P3_RTC limitation • Section 4.1.1 *CPU*: Fixed the link to Cadence Xtensa ISA Summary • Section 4.8.7 *Pulse Counter Controller (PCNT)*: Fixed the typo in the Feature List
2024.08	v4.6	Improved the formatting, structure, and wording in the following sections: • Section 2 *Pins* • Section 3 *Boot Configurations* (used to be named as "Strapping Pins") • Section 4 *Functional Description*
2024.02	v4.5	Section 2.5.3 Chip Power-up and Reset: Updated the link to the VDD_SDIO 1.8 V circuit design to *ESP32 Hardware Design Guidelines*
2023.12	v4.4	Table 1-1 Comparison: Added information about flash under the table
2023.07	v4.3	• Updated formatting throughout the document • Updated wording in some sections • Added a new section 2.3.1 Restrictions for GPIOs and RTC_GPIOs • Added a new section 4.1.5 Cache
2023.01	v4.2	• Removed contents about hall sensor according to PCN20221202 • Section 4.9.3 Touch Sensor: Added a note about limited applications of touch sensor
2022.12	v4.1	• Section 4.1.1 CPU: Added link to *Xtensa® Instruction Set Architecture (ISA) Summary* • Table 1-1 Comparison: Updated the description about chip revision upgrade

Cont'd on next page

Cont'd from previous page

Date	Version	Release notes
2022.10	v4.0	• Section Product Overview: Updated the description • Table 2-6 Pin Mapping Between Chip and Flash/PSRAM: Added two notes below the table • Section 2.5.2 Power Scheme: Added a new item to "Notes on power supply" • Updated Figure 1-1 ESP32 Series Nomenclature • Table 1-1 Comparison: Added a new column "VDD_SDIO Voltage" • Section 4.8.12 TWAI® Controller: Updated the bit rates • Added Not Recommended for New Designs (NRND) label to ESP32-S0WD
2022.03	v3.9	• Added a new chip variant ESP32-D0WDR2-V3 • Added Table 2-5 Pin Mapping Between Chip and Flash/PSRAM and Table 2-6 Pin Mapping Between Chip and Flash/PSRAM • Updated Figure 6-2 QFN48 (5×5 mm) Package • Updated Appendix IO_MUX • Updated Table 4-6 Peripheral Pin Configurations • Section 3.1 Chip Boot Mode Control: Added links to *ESP32 Technical Reference Manual*
2021.10	v3.8	• Upgraded ESP32-U4WDH variant from single-core to dual-core, see PCN-2021-021. The single-core version coexists with the new dual-core version around December 2, 2021. The physical product is subject to batch tracking. • Section CPU and Memory: Added CoreMark® score • Section 4.8.12 TWAI® Controller: Updated the description • Added Not Recommended for New Designs (NRND) label to the ESP32-D0WDQ6-V3 variant • Section 6 Packaging: Provided a link to *Espressif Chip Package Information* • Updated Section Bluetooth
2021.07	v3.7	• Removed ESP32-D2WD variant • Section 4.7.1 Bluetooth Radio and Baseband: Updated wording • Updated pin function numbers starting from Function0 • Added Not Recommended for New Designs (NRND) label to ESP32-D0WD and ESP32-D0WDQ6 variants

Cont'd on next page

Cont'd from previous page

Date	Version	Release notes
2021.03	V3.6	• Updated Figure Block Diagram • Updated Table 5-5 Reliability • Updated Figure 2-3 ESP32 Power Scheme • Updated Table 5-2 Recommended Power Supply Characteristics • Updated the notes below Table 2-4 Description of Timing Parameters for Power-up and Reset • Table 4-1, 4-6, Section 4.8.12 TWAI® Controller: Added more information about TWAI®
2021.01	V3.5	• Table 2-1 Pin Overview: Updated the description for CAP2 from 3 nF to 3.3 nF • Section Advanced Peripheral Interfaces: Added TWAI® • Updated Figure Block Diagram • Appendix IO_MUX: Updated the reset values for MTCK, MTMS, GPIO27
2020.04	V3.4	• Added one chip variant: ESP32-U4WDH • Updated some figures in Table 4-2, 5-6, 5-7, 5-9, 5-11, 5-12 • Table 5-7 Receiver –Basic Data Rate: Added a note under the table
2020.01	V3.3	• Added two chip variants: ESP32-D0WD-V3 and ESP32-D0WDQ6-V3. • Added a note under Table 4-3 Analog-to-Digital Converter (ADC)
2019.10	V3.2	• Updated Figure 2-4 Visualization of Timing Parameters for Power-up and Reset
2019.07	V3.1	• Table 2-1 Pin Overview: Added pin-pin mapping between ESP32-D2WD and the in-package flash under the table • Updated Figure 1-1 ESP32 Series Nomenclature
2019.04	V3.0	• Section 3 *Boot Configurations* (used to be named as "Strapping Pins"): Added information about the setup and hold times for the strapping pins
2019.02	V2.9	• Table 2-1 Pin Overview: Applied new formatting • Table 4-6 Peripheral Pin Configurations: Fixed typos with respect to the ADC1 channel mappings

Cont'd on next page

Cont'd from previous page

Date	Version	Release notes
2019.01	V2.8	• Changed the RF power control range in Table 5-7, 5-10, and 5-12 from –12 ~ +12 to –12 ~ +9 dBm; • Small text changes
2018.11	V2.7	• Updated Section Applications • Table IO_MUX: Updated pin statuses at reset and after reset
2018.10	V2.6	• Section 6 Packaging: Updated QFN package drawings
2018.08	V2.5	• Table 5-1 Absolute Maximum Ratings: Added "Cumulative IO output current" • Table 5-3 DC Characteristics (3.3 V, 25 °C): Added more parameters • Appendix IO_MUX: Changed the power domain names to be consistent with the pin names
2018.07	V2.4	• Deleted information on Packet Traffic Arbitration (PTA); • Added Figure 2-4 Visualization of Timing Parameters for Power-up and Reset • Table 4-2 Power Management Unit (PMU): Added the current consumption figures for dual-core SoCs • Updated Section 4.9.1 Analog-to-Digital Converter (ADC)
2018.06	V2.3	• Table 4-2 Power Management Unit (PMU): Added the current consumption figures at CPU frequency of 160 MHz

Cont'd on next page

Cont'd from previous page

Date	Version	Release notes
2018.05	V2.2	• Table 2-1 Pin Overview: Changed the voltage range of VDD3P3_RTC from 1.8-3.6 V to 2.3-3.6 V • Updated Section 2.5.2 Power Scheme • Updated Section 4.1.3 External Flash and RAM • Updated Table 4-2 Power Management Unit (PMU) • Removed content about temperature sensor; Changes to electrical characteristics: • Updated Table 5-1 Absolute Maximum Ratings • Added Table 5-2 Recommended Power Supply Characteristics • Added Table 5-3 DC Characteristics (3.3 V, 25 °C) • Added Table 5-5 Reliability • Table 5-7 Receiver –Basic Data Rate: Updated the values of "Gain control step" and "Adjacent channel transmit power" • Table 5-10 Transmitter –Enhanced Data Rate: Updated the values of "Gain control step", "$\pi/4$ DQPSK modulation accuracy", "8 DPSK modulation accuracy", and "In-band spurious emissions" • Table 5-12 Transmitter: Updated the values of "Gain control step" and "Adjacent channel transmit power"
2018.01	V2.1	• Deleted software-specific features; • Deleted information on LNA pre-amplifier; • Specified the CPU speed and flash speed of ESP32-D2WD; • Section 2.5.2 Power Scheme: Added notes
2017.12	V2.0	• Section 6 Packaging: Added a note on the sequence of pin number
2017.10	V1.9	• Table 2-1 Pin Overview: Updated the description of pin CHIP_PU • Section 2.5.2 Power Scheme: Added a note • Section 3 *Boot Configurations* (used to be named as "Strapping Pins"): Updated the description of the chip's system reset • Section 4.6.4 Wi-Fi Radio and Baseband: Added a description of antenna diversity and selection • Table 4-2 Power Management Unit (PMU): Deleted "Association sleep pattern", added notes to Active sleep and Modem-sleep
2017.08	V1.8	• Added Table 4-6 Peripheral Pin Configurations • Figure Block Diagram: Corrected a typo

Cont'd on next page

Cont'd from previous page

Date	Version	Release notes
2017.08	V1.7	• Section Bluetooth: Changed the transmitting power to +12 dBm; the sensitivity of NZIF receiver to –97 dBm • Table 2-1 Pin Overview: Added a note • section 4.1.1 CPU: Added 160 MHz clock frequency • Section 4.6.4 Wi-Fi Radio and Baseband: Changed the transmitting power from 21 dBm to 20.5 dBm • Section 4.7.1 Bluetooth Radio and Baseband: Changed the dynamic control range of class-1, class-2 and class-3 transmit output powers to "up to 24 dBm"; changed the dynamic range of NZIF receiver sensitivity to "over 97 dB" • Table 4-2 Power Management Unit (PMU): Added two notes • Updated Section 4.8.1 General Purpose Input / Output Interface (GPIO) • Updated Section 4.8.11 SDIO/SPI Slave Controller • Updated Table 5-1 Absolute Maximum Ratings • Table 5.4 RF Current Consumption in Active Mode: Changed the duty cycle on which the transmitters' measurements are based to 50%. • Table 5-6 Wi-Fi Radio: Added a note on "Output impedance" • Table 5-7, 5-9, 5-11: Updated parameter "Sensitivity" • Table 5-7, 5-10, 5-12: Updated parameters "RF transmit power" and "RF power control range"; added parameter "Gain control step" • Deleted Chapters: "Touch Sensor" and "Code Examples"; • Added a link to certification download.
2017.06	V1.6	• Section Complete Integration Solution: Changed the number of external components to 20 • Section 4.8.1 General Purpose Input / Output Interface (GPIO): Changed the number of GPIO pins to 34
2017.06	V1.5	• Section CPU and Memory: Changed the power supply range • Section 2.5.2 Power Scheme: Updated the note • Updated Table 5-1 Absolute Maximum Ratings • Table Notes on ESP32 Pin Lists: Changed the drive strength values of the digital output pins in Note 8 • Added the option to subscribe for notifications of documentation changes

Cont'd on next page

Cont'd from previous page

Date	Version	Release notes
2017.05	V1.4	• Section Clocks and Timers: Added a note to the frequency of the external crystal oscillator • Section 3 *Boot Configurations* (used to be named as "Strapping Pins"): Added a note • Updated Section 4.3 RTC and Low-power Management • Table 5-1 Absolute Maximum Ratings: Changed the maximum driving capability from 12 mA to 80 mA • Table 5-6 Wi-Fi Radio: Changed the input impedance value of 50Ω to output impedance value of 30+j10 Ω • Table Notes on ESP32 Pin Lists: Added a note to No.8 • Table IO_MUX: Deleted GPIO20
2017.04	V1.3	• Added Appendix Notes on ESP32 Pin Lists • Updated Table 5-6 Wi-Fi Radio • Updated Figure 2-2 ESP32 Pin Layout (QFN 5*5, Top View)
2017.03	V1.2	• Table 2-1 Pin Overview: Added a note • Section 4.1.2 Internal Memory: Updated the note
2017.02	V1.1	• Added Section 1 ESP32 Series Comparison • Updated Section MCU and Advanced Features • Updated Section Block Diagram • Updated Section 2 Pins • Updated Section CPU and Memory • Updated Section 4.2.3 Audio PLL Clock • Updated Section 5.1 Absolute Maximum Ratings • Updated Section 6 Packaging • Updated Section Related Documentation and Resources
2016.08	V1.0	First release.