# ESP32 Series

## Datasheet

### Including:

- ESP32-D0WD
- ESP32-D0WDQ6
- ESP32-D2WD
- ESP32-S0WD

Version 3.0
Espressif Systems


www.espressif.com

## About This Guide

This document provides the specifications of ESP32 family of chips.

## Revision History

For any changes to this document over time, please refer to the last page.

## Documentation Change Notification

Espressif provides email notifications to keep customers updated on changes to technical documentation. Please subscribe at www.espressif.com/en/subscribe.

## Certification

Download certificates for Espressif products from www.espressif.com/en/certificates.

## Disclaimer and Copyright Notice

# Contents

# List of Tables

## List of Figures

# 1. Overview

ESP32 is a single 2.4 GHz Wi-Fi-and-Bluetooth combo chip designed with the TSMC ultra-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-D0WDQ6, ESP32-D0WD, ESP32-D2WD, and ESP32-S0WD. For details on part numbers and ordering information, please refer to Part Number and Ordering Information.

## 1.1 Featured Solutions

### 1.1.1 Ultra-Low-Power Solution

ESP32 is designed for mobile, wearable electronics, and Internet-of-Things (IoT) applications. It features all the state-of-the-art characteristics of low-power chips, including fine-grained clock gating, multiple power modes, and dynamic power scaling. For instance, in a low-power IoT sensor hub application scenario, ESP32 is woken up periodically and only when a specified condition is detected. Low-duty cycle is used to minimize the amount of energy that the chip expends. The output of the power amplifier is also adjustable, thus contributing to an optimal trade-off between communication range, data rate and power consumption.

**Note:**
For more information, refer to Section 3.7 RTC and Low-Power Management.

### 1.1.2 Complete Integration Solution

ESP32 is a highly-integrated solution for Wi-Fi-and-Bluetooth IoT applications, with around 20 external components. ESP32 integrates an antenna switch, RF balun, power amplifier, low-noise receive amplifier, filters, and power management modules. As such, the entire solution occupies minimal Printed Circuit Board (PCB) area.

ESP32 uses CMOS for single-chip fully-integrated radio and baseband, while also integrating advanced calibration circuitries that allow the solution to remove external circuit imperfections or adjust to changes in external conditions. As such, the mass production of ESP32 solutions does not require expensive and specialized Wi-Fi testing equipment.

## 1.2 Wi-Fi Key Features

- 802.11 b/g/n
- 802.11 n (2.4 GHz), up to 150 Mbps
- WMM
- TX/RX A-MPDU, RX A-MSDU
- Immediate Block ACK
- Defragmentation
- Automatic Beacon monitoring (hardware TSF)
- 4 × 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

**Note:**
For more information, please refer to Section 3.5 Wi-Fi.

## 1.3 BT Key Features

- Compliant with Bluetooth v4.2 BR/EDR and BLE specifications
- Class-1, class-2 and class-3 transmitter without external power amplifier
- Enhanced Power Control
- +12 dBm transmitting power
- NZIF receiver with –97 dBm BLE 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 BLE dual mode controller
- Synchronous Connection-Oriented/Extended (SCO/eSCO)
- CVSD and SBC for audio codec
- Bluetooth Piconet and Scatternet
- Multi-connections in Classic BT and BLE
- Simultaneous advertising and scanning

## 1.4 MCU and Advanced Features

### 1.4.1 CPU and Memory

- Xtensa® single-/dual-core 32-bit LX6 microprocessor(s), up to 600 MIPS (200 MIPS for ESP32-S0WD, 400 MIPS for ESP32-D2WD)
- 448 KB ROM
- 520 KB SRAM
- 16 KB SRAM in RTC
- QSPI supports multiple flash/SRAM chips

### 1.4.2 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/BT 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

### 1.4.3 Advanced Peripheral Interfaces

- 34 × programmable GPIOs
- 12-bit SAR ADC up to 18 channels
- 2 × 8-bit DAC
- 10 × touch sensors
- 4 × SPI
- 2 × $I^2S$
- 2 × $I^2C$
- 3 × UART
- 1 host (SD/eMMC/SDIO)
- 1 slave (SDIO/SPI)
- Ethernet MAC interface with dedicated DMA and IEEE 1588 support
- CAN 2.0
- IR (TX/RX)
- Motor PWM
- LED PWM up to 16 channels
- Hall sensor

### 1.4.4 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)

## 1.5 Applications (A Non-exhaustive List)

- Generic Low-power IoT Sensor Hub
- Generic Low-power IoT Data Loggers
- Cameras for Video Streaming
- Over-the-top (OTT) Devices
- Speech Recognition
- Image Recognition
- Mesh Network
- Home Automation
  - Light control
  - Smart plugs
  - Smart door locks
- Smart Building
  - Smart lighting
  - Energy monitoring
- Industrial Automation
  - Industrial wireless control
  - Industrial robotics
- Smart Agriculture
  - Smart greenhouses
  - Smart irrigation
  - Agriculture robotics
- Audio Applications
  - Internet music players
  - Live streaming devices
  - Internet radio players
  - Audio headsets
- Health Care Applications
  - Health monitoring
  - Baby monitors
- Wi-Fi-enabled Toys
  - Remote control toys
  - Proximity sensing toys
  - Educational toys
- Wearable Electronics
  - Smart watches
  - Smart bracelets
- Retail & Catering Applications
  - POS machines
  - Service robots

## 1.6 Block Diagram

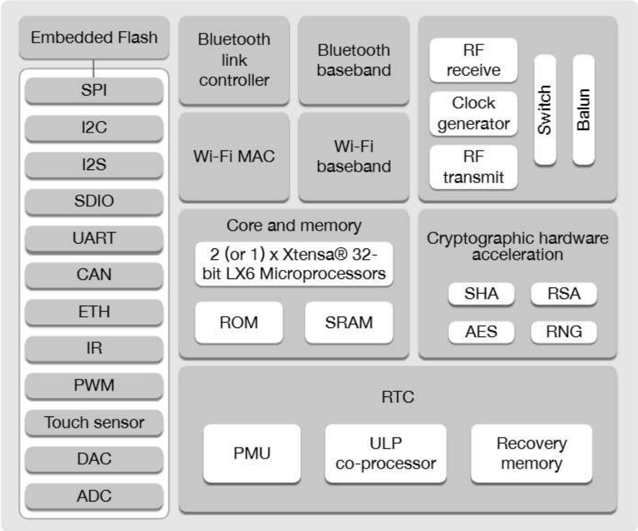



**Figure 1: Functional Block Diagram**

**Note:**

Products in the ESP32 series differ from each other in terms of their support for embedded flash and the number of CPUs they have. For details, please refer to Part Number and Ordering Information.

# 2. Pin Definitions

## 2.1 Pin Layout

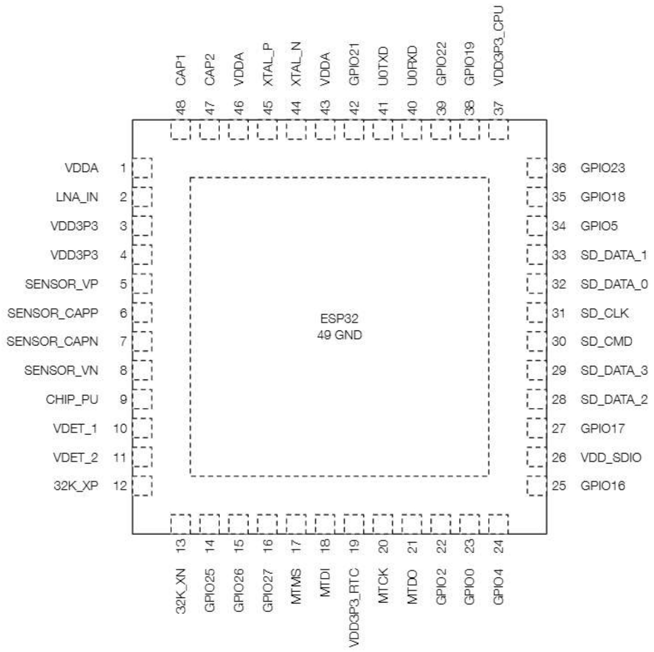



**Figure 2: ESP32 Pin Layout (QFN 6*6, Top View)**

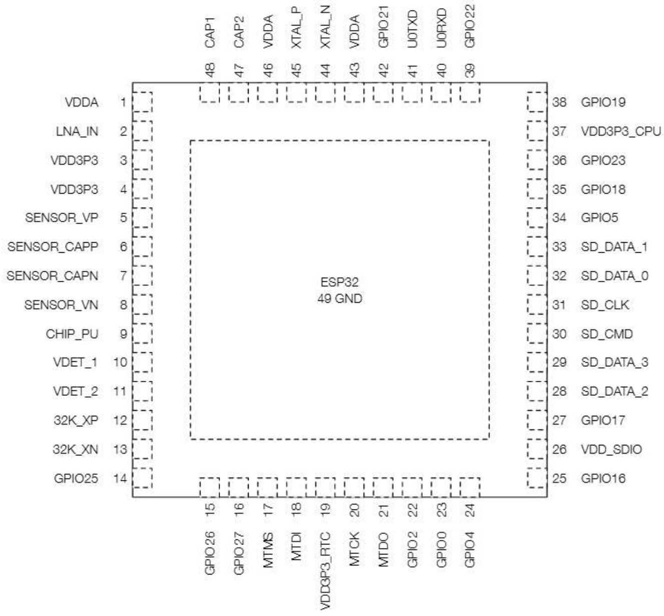



**Figure 3: ESP32 Pin Layout (QFN 5*5, Top View)**

**Note:**
For details on ESP32's part numbers and the corresponding packaging, please refer to Part Number and Ordering Information.

## 2.2 Pin Description

**Table 1: Pin Description**

| 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<br>Low: Off; the chip powers off<br>Note: Do not leave the CHIP_PU pin floating. |

| Name | No. | Type | Function |
|---|---|---|---|
| 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 |
| 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 nF capacitor and 20 kΩ resistor in parallel to CAP1 |
| CAP1 | 48 | I | Connects to a 10 nF series capacitor to ground |
| GND | 49 | P | Ground |

**Note:**

- ESP32-D2WD's pins GPIO16, GPIO17, SD_CMD, SD_CLK, SD_DATA_0 and SD_DATA_1 are used for connecting the embedded flash, and are not recommended for other uses.
- For a quick reference guide to using the IO_MUX, Ethernet MAC, and GIPO Matrix pins of ESP32, please refer to Appendix ESP32 Pin Lists.
- In most cases, the data port connection between the ESP32 and external flash is as follows: SD_DATA0/SPIQ = IO1/DO, SD_DATA1/SPID = IO0/DI, SD_DATA2/SPIHD = IO3/HOLD#, SD_DATA3/SPIWP = IO2/WP#.

## 2.3 Power Scheme

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

- VDD3P3_RTC
- VDD3P3_CPU
- VDD_SDIO

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

VDD3P3_CPU is also the input power supply for CPU.

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. The power scheme diagram is shown below:

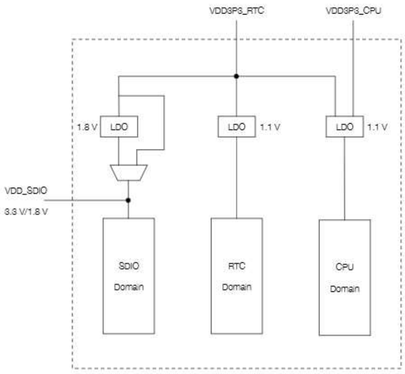



**Figure 4: ESP32 Power Scheme**

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.

**Notes on CHIP_PU:**

- The illustration below shows the ESP32 power-up and reset timing. Details about the parameters are listed in Table 2.

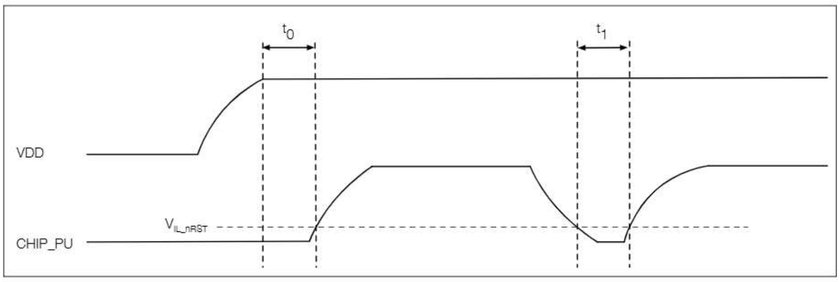



**Figure 5: ESP32 Power-up and Reset Timing**

Table 2: Description of ESP32 Power-up and Reset Timing Parameters

| Parameters | Description | Min. | Unit |
|---|---|---|---|
| $t_0$ | Time between the 3.3 V rails being brought up and CHIP_PU being activated | 50 | $\mu$s |
| $t_1$ | Duration of CHIP_PU signal level < $V_{IL\_nRST}$ (refer to its value in Table 13 DC Characteristics) to reset the chip | 50 | $\mu$s |

- In scenarios where ESP32 is powered on and off 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. Please find the discharge circuit in Figure **ESP32-WROOM-32 Peripheral Schematics**, in *ESP32-WROOM-32 Datasheet*.
- 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. For the reset circuit, please refer to Figure **ESP32-WROOM-32 Peripheral Schematics**, in *ESP32-WROOM-32 Datasheet*.

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

## 2.4 Strapping Pins

ESP32 has five strapping pins:

- MTDI
- GPIO0
- GPIO2
- MTDO
- GPIO5

Software can read the values of these five bits from register "GPIO_STRAPPING".

During the chip's system reset release (power-on-reset, RTC watchdog reset and brownout reset), the latches of the strapping pins sample the voltage level as strapping bits of "0" or "1", and hold these bits until the chip is powered down or shut down. The strapping bits configure the device's boot mode, the operating voltage of VDD_SDIO and other initial system settings.

Each strapping pin is connected to its internal pull-up/pull-down during the chip reset. Consequently, if a strapping pin is unconnected or the connected external circuit is high-impedance, the internal weak pull-up/pull-down will determine the default input level of the strapping pins.

To change the strapping bit values, users can apply the external pull-down/pull-up resistances, or use the host MCU's GPIOs to control the voltage level of these pins when powering on ESP32.

After reset release, the strapping pins work as normal-function pins.

Refer to Table 3 for a detailed boot-mode configuration by strapping pins.

**Table 3: Strapping Pins**

<table>
<tr><th colspan="6">Voltage of Internal LDO (VDD_SDIO)</th></tr>
<tr><td>Pin</td><td>Default</td><td colspan="2">3.3 V</td><td colspan="2">1.8 V</td></tr>
<tr><td>MTDI</td><td>Pull-down</td><td colspan="2">0</td><td colspan="2">1</td></tr>
<tr><th colspan="6">Booting Mode</th></tr>
<tr><td>Pin</td><td>Default</td><td colspan="2">SPI Boot</td><td colspan="2">Download Boot</td></tr>
<tr><td>GPIO0</td><td>Pull-up</td><td colspan="2">1</td><td colspan="2">0</td></tr>
<tr><td>GPIO2</td><td>Pull-down</td><td colspan="2">Don't-care</td><td colspan="2">0</td></tr>
<tr><th colspan="6">Enabling/Disabling Debugging Log Print over U0TXD During Booting</th></tr>
<tr><td>Pin</td><td>Default</td><td colspan="2">U0TXD Toggling</td><td colspan="2">U0TXD Silent</td></tr>
<tr><td>MTDO</td><td>Pull-up</td><td colspan="2">1</td><td colspan="2">0</td></tr>
<tr><th colspan="6">Timing of SDIO Slave</th></tr>
<tr><td>Pin</td><td>Default</td><td>Falling-edge Sampling<br>Falling-edge Output</td><td>Falling-edge Sampling<br>Rising-edge Output</td><td>Rising-edge Sampling<br>Falling-edge Output</td><td>Rising-edge Sampling<br>Rising-edge Output</td></tr>
<tr><td>MTDO</td><td>Pull-up</td><td>0</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>GPIO5</td><td>Pull-up</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
</table>

**Note:**

- Firmware can configure register bits to change the settings of "Voltage of Internal LDO (VDD_SDIO)" and "Timing of SDIO Slave", after booting.
- For ESP32 chips that contain an embedded flash, users need to note the logic level of MTDI. For example, ESP32-D2WD contains an embedded flash that operates at 1.8 V, therefore, the MTDI should be pulled high.

The illustration below shows the setup and hold times for the strapping pin before and after the CHIP_PU signal goes high. Details about the parameters are listed in Table 4.

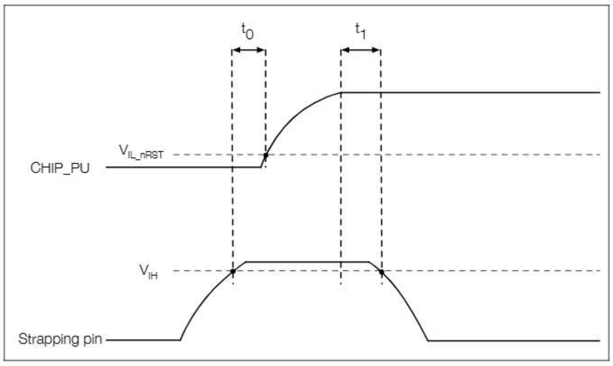



**Figure 6: Setup and Hold Times for the Strapping Pin**

**Table 4: Parameter Descriptions of Setup and Hold Times for the Strapping Pin**

| Parameters | Description | Min. | Unit |
|---|---|---|---|
| $t_0$ | Setup time before CHIP_PU goes from low to high | 0 | ms |
| $t_1$ | Hold time after CHIP_PU goes high | 1 | ms |

# 3. Functional Description

This chapter describes the functions integrated in ESP32.

## 3.1 CPU and Memory

### 3.1.1 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 and ESP32-D2WD)
- 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

### 3.1.2 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 co-processor 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.
- Embedded flash

**Note:**

- Products in the ESP32 series differ from each other, in terms of their support for embedded flash and the size of it. For details, please refer to Part Number and Ordering Information.
- ESP32-D2WD has a 16-Mbit, 40-MHz embedded flash, connected via pins GPIO16, GPIO17, SD_CMD, SD_CLK, SD_DATA_0 and SD_DATA_1.

### 3.1.3 External Flash and SRAM

ESP32 supports multiple external QSPI flash and SRAM chips. More details can be found in Chapter SPI in the *ESP32 Technical Reference Manual*. 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 SRAM 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 SRAM or flash into the CPU address space.

### 3.1.4 Memory Map

The structure of address mapping is shown in Figure 7. The memory and peripheral mapping of ESP32 is shown in Table 5.

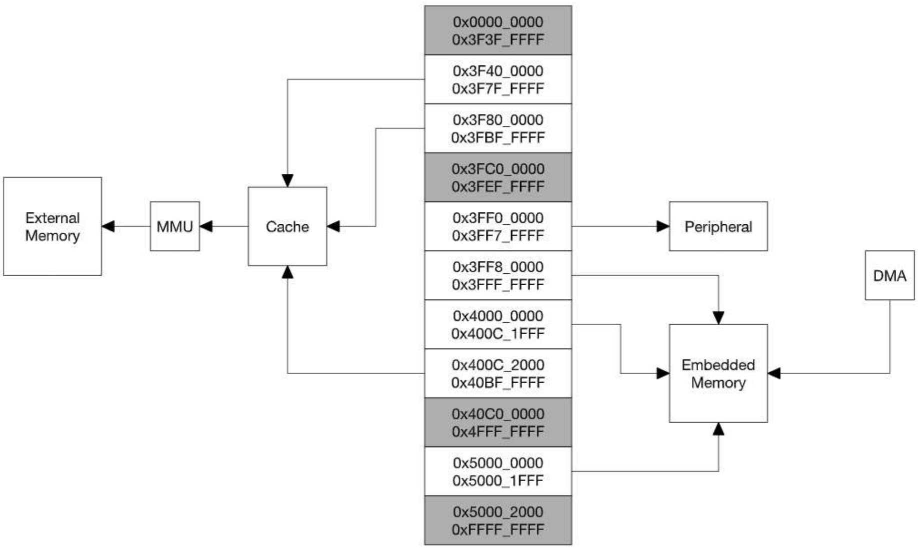



**Figure 7: Address Mapping Structure**

Table 5: 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 SRAM | 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 |

| Category | Target | Start Address | End Address | Size |
|---|---|---|---|---|
| | SPI3 | 0x3FF6_5000 | 0x3FF6_5FFF | 4 KB |
| | 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 |
| | 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 |

## 3.2 Timers and Watchdogs

### 3.2.1 64-bit Timers

There are four general-purpose timers embedded in the ESP32. 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

### 3.2.2 Watchdog Timers

The ESP32 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 ESP32 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.

## 3.3 System Clocks

### 3.3.1 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.

### 3.3.2 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.

### 3.3.3 Audio PLL Clock

The audio clock is generated by the ultra-low-noise fractional-N PLL. More details can be found in Chapter Reset and Clock in the *ESP32 Technical Reference Manual*.

## 3.4 Radio

The ESP32 radio consists of the following blocks:

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

- clock generator

### 3.4.1 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 with ESP32.

### 3.4.2 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.

### 3.4.3 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.

## 3.5 Wi-Fi

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.

### 3.5.1 Wi-Fi Radio and Baseband

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 $\mu$s 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.

### 3.5.2 Wi-Fi MAC

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

- 4 × 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)

## 3.6 Bluetooth

ESP32 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.

### 3.6.1 Bluetooth Radio and Baseband

The ESP32 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 24 dB
- $\pi$/4 DQPSK and 8 DPSK modulation
- High performance in NZIF receiver sensitivity with over 97 dB of dynamic range
- 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, $\mu$-law and CVSD digital audio CODEC in PCM interface

- SBC audio CODEC
- Power management for low-power applications
- SMP with 128-bit AES

### 3.6.2 Bluetooth Interface

- Provides UART HCI interface, up to 4 Mbps
- Provides SDIO / SPI HCI interface
- Provides PCM / I²S audio interface

### 3.6.3 Bluetooth Stack

The Bluetooth stack of ESP32 is compliant with the Bluetooth v4.2 BR / EDR and BLE specifications.

### 3.6.4 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

## 3.7 RTC and Low-Power Management

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

- Power modes
  - Active mode: The chip radio is powered on. 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 co-processor are running. Any wake-up events (MAC, host, RTC timer, or external interrupts) will wake up the chip.
  - Deep-sleep mode: Only the RTC memory and RTC peripherals are powered on. Wi-Fi and Bluetooth connection data are stored in the RTC memory. The ULP co-processor is functional.
  - Hibernation mode: The internal 8-MHz oscillator and ULP co-processor 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 6: Power Consumption by Power Modes**

| Power mode | Description | | | Power consumption |
|---|---|---|---|---|
| Active (RF working) | Wi-Fi Tx packet | | | Please refer to Table 15 for details. |
| | Wi-Fi/BT Tx packet | | | |
| | Wi-Fi/BT Rx and listening | | | |
| Modem-sleep | The CPU is powered on. | 240 MHz * | Dual-core chip(s) | 30 mA ~ 68 mA |
| | | | Single-core chip(s) | N/A |
| | | 160 MHz * | Dual-core chip(s) | 27 mA ~ 44 mA |
| | | | Single-core chip(s) | 27 mA ~ 34 mA |
| | | Normal speed: 80 MHz | Dual-core chip(s) | 20 mA ~ 31 mA |
| | | | Single-core chip(s) | 20 mA ~ 25 mA |
| Light-sleep | - | | | 0.8 mA |
| Deep-sleep | The ULP co-processor is powered on. | | | 150 $\mu$A |
| | ULP sensor-monitored pattern | | | 100 $\mu$A @1% duty |
| | RTC timer + RTC memory | | | 10 $\mu$A |
| Hibernation | RTC timer only | | | 5 $\mu$A |
| Power off | CHIP_PU is set to low level, the chip is powered off. | | | 0.1 $\mu$A |

**Note:**

- * Among the ESP32 series of SoCs, ESP32-D0WDQ6 and ESP32-D0WD have a maximum CPU frequency of 240 MHz, ESP32-D2WD and ESP32-S0WD have 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 co-processor is powered on, peripherals such as GPIO and I²C are able to operate.
- When the system works in the ULP sensor-monitored pattern, the ULP co-processor works with the ULP sensor periodically and the ADC works with a duty cycle of 1%, so the power consumption is 100 $\mu$A.

# 4. Peripherals and Sensors

## 4.1 Descriptions of Peripherals and Sensors

### 4.1.1 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.

### 4.1.2 Analog-to-Digital Converter (ADC)

ESP32 integrates 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.

With appropriate settings, the ADCs can be configured to measure voltage on 18 pins maximum.

Table 7 describes the ADC characteristics.

**Table 7: ADC Characteristics**

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

**Notes:**

- When atten=3 and the measurement result is above 3,000 (voltage at approx. 2,450 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.

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 8. For higher accuracy, users may apply other calibration methods provided in ESP-IDF, or implement their own.

**Table 8: 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 100–1,250 mV | –30 | 30 | mV |
| | Atten=2, effective measurement range of 150–1,750 mV | –40 | 40 | mV |
| | Atten=3, effective measurement range of 150–2,450 mV | –60 | 60 | mV |

### 4.1.3 Hall Sensor

ESP32 integrates a Hall sensor based on an N-carrier resistor. When the chip is in the magnetic field, the Hall sensor develops a small voltage laterally on the resistor, which can be directly measured by the ADC.

### 4.1.4 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.

### 4.1.5 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. The 10 capacitive-sensing GPIOs are listed in Table 9.

**Table 9: 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 |

### 4.1.6 Ultra-Low-Power Co-processor

The ULP processor and RTC memory remain powered on during the Deep-sleep mode. Hence, the developer can store a program for the ULP processor 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.

### 4.1.7 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. The following features are supported on the Ethernet MAC (EMAC) interface:

- 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

### 4.1.8 SD/SDIO/MMC Host Controller

An SD/SDIO/MMC host controller is available on ESP32, which supports the following features:

- Secure Digital memory (SD mem Version 3.0 and Version 3.01)
- Secure Digital I/O (SDIO Version 3.0)
- Consumer Electronics Advanced Transport Architecture (CE-ATA Version 1.1)
- Multimedia Cards (MMC Version 4.41, eMMC Version 4.5 and Version 4.51)

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

### 4.1.9 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.

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

### 4.1.10 Universal Asynchronous Receiver Transmitter (UART)

ESP32 has three UART interfaces, i.e., UART0, UART1 and UART2, which provide asynchronous communication (RS232 and RS485) and IrDA support, communicating at a speed of up to 5 Mbps. UART provides hardware management of the CTS and RTS signals and software flow control (XON and XOFF). All of the interfaces can be accessed by the DMA controller or directly by the CPU.

### 4.1.11 $I^2C$ Interface

ESP32 has two $I^2C$ bus interfaces which can serve as $I^2C$ master or slave, depending on the user's configuration. The $I^2C$ interfaces support:

- Standard mode (100 Kbit/s)
- Fast mode (400 Kbit/s)
- Up to 5 MHz, yet constrained by SDA pull-up strength
- 7-bit/10-bit addressing mode
- Dual addressing mode

Users can program command registers to control $I^2C$ interfaces, so that they have more flexibility.

### 4.1.12 $I^2S$ Interface

Two standard $I^2S$ interfaces are available in ESP32. They can be operated in master or slave mode, in full duplex and half-duplex communication modes, and can be configured to operate with an 8-/16-/32-/48-/64-bit resolution as input or output channels. BCK clock frequency, from 10 kHz up to 40 MHz, is supported. When one or both of the $I^2S$ interfaces are configured in the master mode, the master clock can be output to the external DAC/CODEC.

Both of the $I^2S$ interfaces have dedicated DMA controllers. PDM and BT PCM interfaces are supported.

### 4.1.13 Infrared Remote Controller

The infrared remote controller supports eight channels of infrared remote transmission and receiving. By programming the pulse waveform, it supports various infrared protocols. Eight channels share a 512 x 32-bit block of memory to store the transmitting or receiving waveform.

### 4.1.14 Pulse Counter

The pulse counter captures pulse and counts pulse edges through seven modes. It has eight channels, each of which captures four signals at a time. The four input signals include two pulse signals and two control signals. When the counter reaches a defined threshold, an interrupt is generated.

### 4.1.15 Pulse Width Modulation (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.

### 4.1.16 LED PWM

The LED PWM controller can generate 16 independent channels of digital waveforms with configurable periods and duties.

The 16 channels of digital waveforms operate with an APB clock of 80 MHz. Eight of these channels have the option of using the 8 MHz oscillator clock. Each channel can select a 20-bit timer with configurable counting range, while its accuracy of duty can be up to 16 bits within a 1 ms period.

The software can change the duty immediately. Moreover, each channel automatically supports step-by-step duty increase or decrease, which is useful for the LED RGB color-gradient generator.

### 4.1.17 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. These SPIs also support the following general-purpose SPI features:

- Four modes of SPI transfer format, which depend on the polarity (CPOL) and the phase (CPHA) of the SPI clock
- Up to 80 MHz (The actual speed it can reach depends on the selected pads, PCB tracing, peripheral characteristics, etc.)
- up to 64-byte FIFO

All SPIs can also be connected to the external flash/SRAM and LCD. Each SPI can be served by DMA controllers.

### 4.1.18 Accelerator

ESP32 is equipped with hardware accelerators of general algorithms, such as AES (FIPS PUB 197), SHA (FIPS PUB 180-4), RSA, and ECC, which support independent arithmetic, such as Big Integer Multiplication and Big Integer Modular Multiplication. The maximum operation length for RSA, ECC, Big Integer Multiply and Big Integer Modular Multiplication is 4,096 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.

## 4.2 Peripheral Pin Configurations

**Table 10: 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 | Two 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 | | |
| $I^2C$ | I2CEXT0_SCL_in | Any GPIO Pins | Two $I^2C$ 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 |
| | 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 | | |
| Infrared Remote Controller | 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:<br>• Both master and slave modes;<br>• Four sub-modes of the SPI transfer format;<br>• Configurable SPI frequency;<br>• 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 | | |

# 5. Electrical Characteristics

## 5.1 Absolute Maximum Ratings

Stresses beyond the absolute maximum ratings listed in the table below may cause permanent damage to the device. These are stress ratings only, and do not refer to the functional operation of the device that should follow the recommended operating conditions.

**Table 11: Absolute Maximum Ratings**

| Symbol | Parameter | Min | Max | Unit |
|---|---|---|---|---|
| VDDA, VDD3P3, VDD3P3_RTC, VDD3P3_CPU, VDD_SDIO | Voltage applied to power supply pins per power domain | –0.3 | 3.6 | V |
| $I_{output}$ * | Cumulative IO output current | - | 1,200 | mA |
| $T_{store}$ | Storage temperature | –40 | 150 | °C |

* The chip worked properly after a 24-hour test in ambient temperature at 25 °C, and the IOs in three domains (VDD3P3_RTC, VDD3P3_CPU, VDD_SDIO) output high logic level to ground.

## 5.2 Recommended Operating Conditions

**Table 12: Recommended Operating Conditions**

| Symbol | Parameter | Min | Typical | Max | Unit |
|---|---|---|---|---|---|
| VDDA, VDD3P3_RTC [1]<br>VDD3P3, VDD_SDIO (3.3 V mode) [2] | Voltage applied to power supply pins per power domain | 2.3 | 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 [3] | Operating temperature | –40 | - | 125 | °C |

1. When writing eFuse, VDD3P3_RTC should be at least 3.3 V.
2. • 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 Power Scheme, section 2.3, for more information.
3. The operating temperature of ESP32-D2WD ranges from –40 °C ~ 105 °C, due to the flash embedded in it. The other chips in this series have no ebedded flash, so their range of operating temperatures is –40 °C ~ 125 °C.

## 5.3 DC Characteristics (3.3 V, 25 °C)

**Table 13: DC Characteristics (3.3 V, 25 °C)**

<table>
<tr><th>Symbol</th><th colspan="2">Parameter</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">Pull-up resistor</td><td>-</td><td>45</td><td>-</td><td>kΩ</td></tr>
<tr><td>$R_{PD}$</td><td colspan="2">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 power off the chip</td><td>-</td><td>-</td><td>0.6</td><td>V</td></tr>
</table>

**Notes:**

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 Reliability Qualifications

**Table 14: Reliability Qualifications**

| Reliability tests | Standards | Test conditions | Result |
|---|---|---|---|
| Electro-Static Discharge Sensitivity (ESD), Charge Device Mode (CDM) [1] | JEDEC EIA/JESD22-C101 | ±500 V, all pins | Pass |
| Electro-Static Discharge Sensitivity (ESD), Human Body Mode (HBM) [2] | JEDEC EIA/JESD22-A114 | ±1500 V, all pins | Pass |
| Latch-up (Over-current test) | JEDEC STANDARD NO.78 | ±50 mA ~ ±200 mA, room temperature, test for IO | Pass |
| Latch-up (Over-voltage test) | JEDEC STANDARD NO.78 | 1.5 × Vmax, room temperature, test for $V_{supply}$ | Pass |
| Moisture Sensitivity Level (MSL) | J-STD-020, MSL 3 | 30 °C, 60% RH, 192 hours, IR × 3 @260 °C | Pass |

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

## 5.5 RF Power-Consumption Specifications

The power 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 15: RF Power-Consumption Specifications**

| Mode | Min | Typ | Max | Unit |
|---|---|---|---|---|
| Transmit 802.11b, DSSS 1 Mbps, POUT = +19.5 dBm | - | 240 | - | mA |
| Transmit 802.11b, OFDM 54 Mbps, POUT = +16 dBm | - | 190 | - | mA |
| Transmit 802.11g, 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 |

## 5.6 Wi-Fi Radio

**Table 16: Wi-Fi Radio Characteristics**

| Description | Min | Typical | Max | Unit |
|---|---|---|---|---|
| Input frequency | 2412 | - | 2484 | MHz |
| Output impedance* | - | * | - | Ω |
| Tx power | | | | |
| Output power of PA for 72.2 Mbps | 13 | 14 | 15 | dBm |
| Output power of PA for 11b mode | 19.5 | 20 | 20.5 | dBm |
| Sensitivity | | | | |
| DSSS, 1 Mbps | - | –98 | - | dBm |
| CCK, 11 Mbps | - | –91 | - | dBm |
| OFDM, 6 Mbps | - | –93 | - | dBm |
| OFDM, 54 Mbps | - | –75 | - | dBm |
| HT20, MCS0 | - | –93 | - | dBm |
| HT20, MCS7 | - | –73 | - | dBm |
| HT40, MCS0 | - | –90 | - | dBm |
| HT40, MCS7 | - | –70 | - | dBm |
| MCS32 | - | –89 | - | dBm |
| Adjacent channel rejection | | | | |
| OFDM, 6 Mbps | - | 37 | - | dB |
| OFDM, 54 Mbps | - | 21 | - | dB |
| HT20, MCS0 | - | 37 | - | dB |
| HT20, MCS7 | - | 20 | - | dB |

*The typical value of ESP32's Wi-Fi radio output impedance is different in chips of different QFN packages. For ESP32 chips with a QFN 6x6 package (ESP32-D0WDQ6), the value is 30+j10 Ω. For ESP32 chips with a QFN 5x5 package (ESP32-D0WD, ESP32-D2WD, ESP32-S0WD), the value is 35+j10 Ω.

## 5.7 Bluetooth Radio

### 5.7.1 Receiver – Basic Data Rate

**Table 17: Receiver Characteristics – Basic Data Rate**

| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Sensitivity @0.1% BER | - | - | –94 | - | 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 18: Transmitter Characteristics – Basic Data Rate**

| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| RF transmit power | - | - | 0 | - | dBm |
| Gain control step | - | - | 3 | - | dBm |
| 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 |
| $\Delta f1_{avg}$ | - | - | - | 155 | kHz |
| $\Delta f2_{max}$ | - | 133.7 | - | - | kHz |
| $\Delta f2_{avg}/\Delta f1_{avg}$ | - | - | 0.92 | - | - |
| ICFT | - | - | –7 | - | kHz |
| Drift rate | - | - | 0.7 | - | kHz/50 $\mu$s |
| Drift (DH1) | - | - | 6 | - | kHz |
| Drift (DH5) | - | - | 6 | - | kHz |

### 5.7.3 Receiver – Enhanced Data Rate

Table 19: Receiver Characteristics – Enhanced Data Rate

| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| $\pi/4$ DQPSK | | | | | |
| Sensitivity @0.01% BER | - | - | –90 | - | 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 | - | 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 20: Transmitter Characteristics – Enhanced Data Rate

| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| RF transmit power | - | - | 0 | - | dBm |
| Gain control step | - | - | 3 | - | dBm |
| 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 |

| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| | 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 21: Receiver Characteristics – BLE

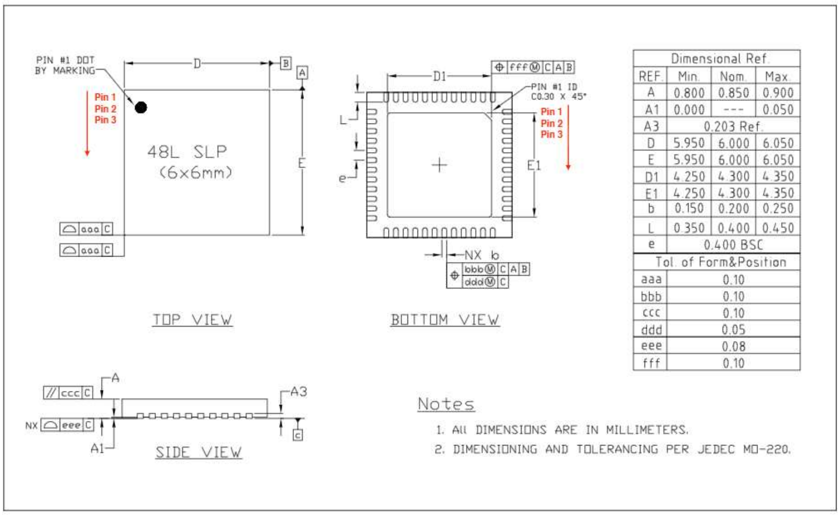
| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| Sensitivity @30.8% PER | - | - | –97 | - | dBm |
| Maximum received signal @30.8% PER | - | 0 | - | - | dBm |
| Co-channel C/I | - | - | +10 | - | dB |
| Adjacent channel selectivity C/I | F = F0 + 1 MHz | - | –5 | - | dB |
| | 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 22: Transmitter Characteristics – BLE

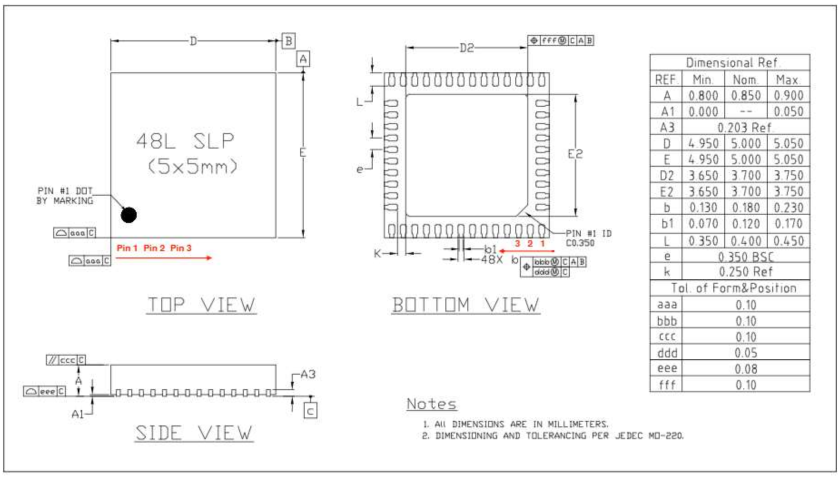
| Parameter | Conditions | Min | Typ | Max | Unit |
|---|---|---|---|---|---|
| RF transmit power | - | - | 0 | - | dBm |
| Gain control step | - | - | 3 | - | dBm |
| RF power control range | - | –12 | - | +9 | dBm |
| Adjacent channel transmit power | F = F0 ± 2 MHz | - | –52 | - | dBm |
| | F = F0 ± 3 MHz | - | –58 | - | dBm |
| | F = F0 ± > 3 MHz | - | –60 | - | dBm |
| $\Delta f1_{avg}$ | - | - | - | 265 | kHz |
| $\Delta f2_{max}$ | - | 247 | - | - | kHz |
| $\Delta f2_{avg}/\Delta f1_{avg}$ | - | - | –0.92 | - | - |
| ICFT | - | - | –10 | - | kHz |
| Drift rate | - | - | 0.7 | - | kHz/50 $\mu$s |
| Drift | - | - | 2 | - | kHz |

# 6. Package Information




| 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 8: QFN48 (6x6 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 | 0.350 | 0.400 | 0.450 |
| e | 0.350 BSC | | |
| k | 0.250 Ref. | | |
| Tol. of Form&Position | | | |
| aaa | 0.10 | | |
| bbb | 0.10 | | |
| ccc | 0.10 | | |
| ddd | 0.05 | | |
| eee | 0.08 | | |
| fff | 0.10 | | |

Figure 9: QFN48 (5x5 mm) Package

**Note:**
The pins of the chip are numbered in an anti-clockwise direction from Pin 1 in the top view.

# 7. Part Number and Ordering Information

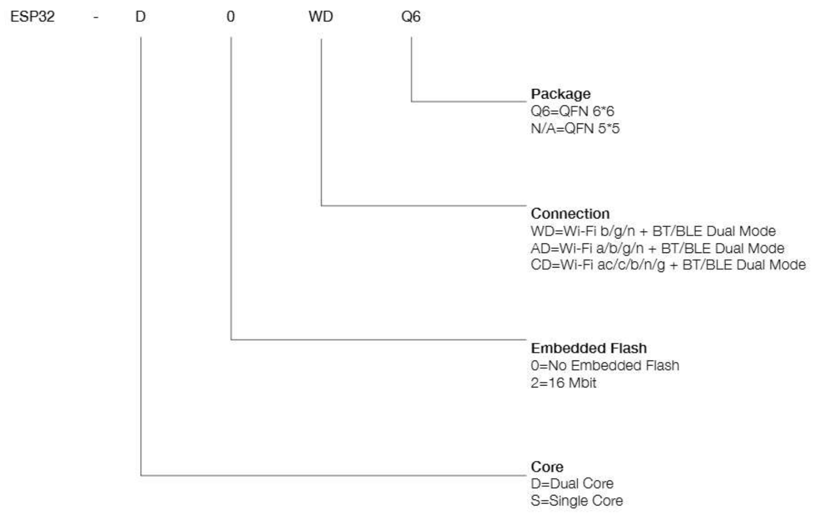



**Figure 10: ESP32 Part Number**

The table below provides the ordering information of the ESP32 series of chips.

**Table 23: ESP32 Ordering Information**

| Ordering code | Core | Embedded flash | Connection | Package |
|---|---|---|---|---|
| ESP32-D0WDQ6 | Dual core | No embedded flash | Wi-Fi b/g/n + BT/BLE Dual Mode | QFN 6*6 |
| ESP32-D0WD | Dual core | No embedded flash | Wi-Fi b/g/n + BT/BLE Dual Mode | QFN 5*5 |
| ESP32-D2WD | Dual core | 16-Mbit embedded flash (40 MHz) | Wi-Fi b/g/n + BT/BLE Dual Mode | QFN 5*5 |
| ESP32-S0WD | Single core | No embedded flash | Wi-Fi b/g/n + BT/BLE Dual Mode | QFN 5*5 |

# 8. Learning Resources

## 8.1 Must-Read Documents

Click on the following links to access documents related to ESP32.

- *ESP-IDF Programming Guide*
  It hosts extensive documentation for ESP-IDF, ranging from hardware guides to API reference.
- *ESP32 Technical Reference Manual*
  The manual provides detailed information on how to use the ESP32 memory and peripherals.
- ESP32 Hardware Resources
  The zip files include schematics, PCB layout, Gerber and BOM list.
- *ESP32 Hardware Design Guidelines*
  The guidelines provide recommended design practices when developing standalone or add-on systems based on the ESP32 series of products, including the ESP32 chip, the ESP32 modules and development boards.
- *ESP32 AT Instruction Set and Examples*
  This document introduces the ESP32 AT commands, explains how to use them, and provides examples of several common AT commands.
- *Espressif Products Ordering Information*

## 8.2 Must-Have Resources

Here are the ESP32-related must-have resources.

- ESP32 BBS
  This is an Engineer-to-Engineer (E2E) Community for ESP32, where you can post questions, share knowledge, explore ideas, and solve problems together with fellow engineers.
- ESP32 GitHub
  ESP32 development projects are freely distributed under Espressif's MIT license on GitHub. This channel of communication has been established to help developers get started with ESP32 and encourage them to share their knowledge of ESP32-related hardware and software.
- ESP32 Tools
  This is a webpage where users can download ESP32 Flash Download Tools and the zip file "ESP32 Certification and Test".
- ESP-IDF
  This webpage links users to the official IoT development framework for ESP32.
- ESP32 Resources
  This webpage provides the links to all available ESP32 documents, SDK and tools.

# Appendix A – ESP32 Pin Lists

## A.1. Notes on ESP32 Pin Lists

Table 24: 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 1~3" 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 1~6" In Table IO_MUX. The function selection registers will be set as "*N*-1", where *N* is the function number. Below are some definitions:<br>• SD_* is for signals of the SDIO slave.<br>• HS1_* is for Port 1 signals of the SDIO host.<br>• HS2_* is for Port 2 signals of the SDIO host.<br>• MT* is for signals of the JTAG.<br>• U0* is for signals of the UART0 module.<br>• U1* is for signals of the UART1 module.<br>• U2* is for signals of the UART2 module.<br>• SPI* is for signals of the SPI01 module.<br>• HSPI* is for signals of the SPI2 module.<br>• 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:<br>• I: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is still from this pin.<br>• I1: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is always "1".<br>• I0: input only. If a function other than "Function-*N*" is assigned, the input signal of "Function-*N*" is always "0".<br>• O: output only.<br>• T: high-impedance.<br>• I/O/T: combinations of input, output, and high-impedance according to the function signal.<br>• 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".<br>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:<br>• 0: ~5 mA<br>• 1: ~10 mA<br>• 2: ~20 mA<br>• 3: ~40 mA<br>The default value is 2.<br>The drive strength of the internal pull-up (wpu) and pull-down (wpd) is ~75 $\mu$A. |
| 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 1". The output-enable is controlled by digital Function 1. |
| 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_FUNCm_IN_INV_SEL and GPIO_FUNCm_IN_SEL. (The value of m ranges from 1 to 255.) |

## A.2. GPIO_Matrix

Table 25: 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 |
| 35 | pwm0_f1_in | 0 | no | pwm0_out1b | 1'd1 |

| Signal No. | Input signals | Default value if unassigned | Same input signal from IO_MUX core | Output signals | Output enable of output signals |
|---|---|---|---|---|---|
| 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 |
| 79 | pcnt_sig_ch0_in7 | 0 | no | ledc_ls_sig_out0 | 1'd1 |

| Signal No. | Input signals | Default value if unassigned | Same input signal from IO_MUX core | Output signals | Output enable of output signals |
|---|---|---|---|---|---|
| 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 | - | - | - | 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 |
| 121 | pwm3_fltb | 1 | no | pwm2_out4l | 1’d1 |

| Signal No. | Input signals | Default value if unassigned | Same input signal from IO_MUX core | Output signals | Output enable of output signals |
|---|---|---|---|---|---|
| 122 | pwm3_cap1_in | 0 | no | - | 1'd1 |
| 123 | pwm3_cap2_in | 0 | no | - | 1'd1 |
| 124 | pwm3_cap3_in | 0 | no | - | 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 |
| 176 | I2S1I_DATA_in10 | 0 | no | I2S1O_DATA_out10 | 1'd1 |
| 177 | I2S1I_DATA_in11 | 0 | no | I2S1O_DATA_out11 | 1'd1 |

| Signal No. | Input signals | Default value if unassigned | Same input signal from IO_MUX core | Output signals | Output enable of output signals |
|---|---|---|---|---|---|
| 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 |
| 225 | - | - | - | sig_in_func225 | 1'd1 |
| 226 | - | - | - | sig_in_func226 | 1'd1 |

| Signal No. | Input signals | Default value if unassigned | Same input signal from IO_MUX core | Output signals | Output enable of output signals |
|---|---|---|---|---|---|
| 227 | - | - | - | sig_in_func227 | 1'd1 |
| 228 | - | - | - | sig_in_func228 | 1'd1 |

## A.3. Ethernet_MAC

**Table 26: 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 Function1 | Analog Function2 | Analog Function3 | RTC Function1 | RTC Function2 | Function1 | Type | Function2 | Type | Function3 | Type | Function4 | Type | Function5 | Type | Function6 | 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 | ADC_H | ADC1_CH0 | | RTC_GPIO0 | | GPIO36 | I | | | GPIO36 | I | | | | | | | | oe=0, ie=0 | oe=0, ie=0 |
| 6 | | SENSOR_CAPP | | VDD3P3_RTC | ADC_H | ADC1_CH1 | | RTC_GPIO1 | | GPIO37 | I | | | GPIO37 | I | | | | | | | | oe=0, ie=0 | oe=0, ie=0 |
| 7 | | SENSOR_CAPN | | VDD3P3_RTC | ADC_H | ADC1_CH2 | | RTC_GPIO2 | | GPIO38 | I | | | GPIO38 | I | | | | | | | | oe=0, ie=0 | oe=0, ie=0 |
| 8 | | SENSOR_VN | | VDD3P3_RTC | ADC_H | 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=1 |
| 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 |
| 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 |
| 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 |
|---|---|---|
| 2019.04 | V3.0 | Added information about the setup and hold times for the strapping pins in Section 2.4: Strapping Pins. |
| 2019.02 | V2.9 | Applied new formatting to Table 1: Pin Description;<br>Fixed typos with respect to the ADC1 channel mappings in Table 10: Peripheral Pin Configurations. |
| 2019.01 | V2.8 | Changed the RF power control range in Table 18, Table 20 and Table 22 from –12 ~ +12 to –12 ~ +9 dBm;<br>Small text changes. |
| 2018.11 | V2.7 | Updated Section 1.5;<br>Updated pin statuses at reset and after reset in Table IO_MUX. |
| 2018.10 | V2.6 | Updated QFN package drawings in Chapter 6: Package Information. |
| 2018.08 | V2.5 | • Added "Cumulative IO output current" entry to Table 11: Absolute Maximum Ratings;<br>• Added more parameters to Table 13: DC Characteristics;<br>• Changed the power domain names in Table IO_MUX to be consistent with the pin names. |
| 2018.07 | V2.4 | • Deleted information on Packet Traffic Arbitration (PTA);<br>• Added Figure 5: ESP32 Power-up and Reset Timing in Section 2.3: Power Scheme;<br>• Added the power consumption of dual-core SoCs in Table 6: Power Consumption by Power Modes;<br>• Updated section 4.1.2: Analog-to-Digital Converter (ADC). |
| 2018.06 | V2.3 | Added the power consumption at CPU frequency of 160 MHz in Table 6: Power Consumption by Power Modes. |
| 2018.05 | V2.2 | • Changed the voltage range of VDD3P3_RTC from 1.8-3.6V to 2.3-3.6V in Table 1: Pin Description;<br>• Updated Section 2.3: Power Scheme;<br>• Updated Section 3.1.3: External Flash and SRAM;<br>• Updated Table 6: Power Consumption by Power Modes;<br>• Deleted content about temperature sensor;<br>Changes to electrical characteristics:<br>• Updated Table 11: Absolute Maximum Ratings;<br>• Added Table 12: Recommended Operating Conditions;<br>• Added Table 13: DC Characteristics;<br>• Added Table 14: Reliability Qualifications;<br>• Updated the values of "Gain control step" and "Adjacent channel transmit power" in Table 18: Transmitter Characteristics - Basic Data Rate;<br>• Updated the values of "Gain control step", "$\pi$/4 DQPSK modulation accuracy", "8 DPSK modulation accuracy" and "In-band spurious emissions" in Table 20: Transmitter Characteristics – Enhanced Data Rate;<br>• Updated the values of "Gain control step", "Adjacent channel transmit power" in Table 22: Transmitter Characteristics - BLE. |

| Date | Version | Release notes |
|---|---|---|
| 2018.01 | V2.1 | • Deleted software-specific features;<br>• Deleted information on LNA pre-amplifier;<br>• Specified the CPU speed and flash speed of ESP32-D2WD;<br>• Added notes to Section 2.3: Power Scheme. |
| 2017.12 | V2.0 | Added a note on the sequence of pin number in Chapter 6. |
| 2017.10 | V1.9 | • Updated the description of the pin CHIP_PU in Table 1;<br>• Added a note to Section 2.3: Power Scheme;<br>• Updated the description of the chip's system reset in Section 2.4: Strapping Pins;<br>• Added a description of antenna diversity and selection to Section 3.5.1;<br>• Deleted "Association sleep pattern" in Table 6 and added notes to Active sleep and Modem-sleep. |
| 2017.08 | V1.8 | • Added Table 4.2 in Section 4;<br>• Corrected a typo in Figure 1. |
| 2017.08 | V1.7 | • Changed the transmitting power to +12 dBm; the sensitivity of NZIF receiver to -97 dBm in Section 1.3;<br>• Added a note to Table 1 Pin Description;<br>• Added 160 MHz clock frequency in section 3.1.1;<br>• Changed the transmitting power from 21 dBm to 20.5 dBm in Section 3.5.1;<br>• Changed the dynamic control range of class-1, class-2 and class-3 transmit output powers to "up to 24 dBm"; and changed the dynamic range of NZIF receiver sensitivity to "over 97 dB" in Section 3.6.1;<br>• Updated Table 6: Power Consumption by Power Modes, and added two notes to it;<br>• Updated sections 4.1.1, 4.1.9;<br>• Updated Table 11: Absolute Maximum Ratings;<br>• Updated Table 15: RF Power Consumption Specifications, and changed the duty cycle on which the transmitters' measurements are based by 50%.<br>• Updated Table 16: Wi-Fi Radio Characteristics and added a note on "Output impedance" to it;<br>• Updated parameter "Sensitivity" in Table 17, 19, 21;<br>• Updated parameters "RF transmit power" and "RF power control range", and added parameter "Gain control step" in Table 18, 20, 22;<br>• Deleted Chapters: "Touch Sensor" and "Code Examples";<br>• Added a link to certification download. |
| 2017.06 | V1.6 | Corrected two typos:<br>• Changed the number of external components to 20 in Section 1.1.2;<br>• Changed the number of GPIO pins to 34 in Section 4.1.1. |
| 2017.06 | V1.5 | • Changed the power supply range in Section: 1.4.1 CPU and Memory;<br>• Updated the note in Section 2.3: Power Scheme;<br>• Updated Table 11: Absolute Maximum Ratings;<br>• Changed the drive strength values of the digital output pins in Note 8, in Table 24: Notes on ESP32 Pin Lists;<br>• Added the option to subscribe for notifications of documentation changes. |

| Date | Version | Release notes |
|---|---|---|
| 2017.05 | V1.4 | • Added a note to the frequency of the external crystal oscillator in Section 1.4.2: Clocks and Timers;<br>• Added a note to Section 2.4: Strapping Pins;<br>• Updated Section 3.7: RTC and Low-Power Management;<br>• Changed the maximum driving capability from 12 mA to 80 mA, in Table 11: Absolulte Maximum Ratings;<br>• Changed the input impedance value of 50Ω, in Table 16: Wi-Fi Radio Characteristics, to output impedance value of 30+j10 Ω;<br>• Added a note to No.8 in Table 24: Notes on ESP32 Pin Lists;<br>• Deleted GPIO20 in Table IO_MUX. |
| 2017.04 | V1.3 | • Added Appendix: ESP32 Pin Lists;<br>• Updated Table: Wi-Fi Radio Characteristics;<br>• Updated Figure: ESP32 Pin Layout (for QFN 5*5). |
| 2017.03 | V1.2 | • Added a note to Table: Pin Description;<br>• Updated the note in Section: Internal Memory. |
| 2017.02 | V1.1 | • Added Chapter: Part Number and Ordering Information;<br>• Updated Section: MCU and Advanced Features;<br>• Updated Section: Block Diagram;<br>• Updated Chapter: Pin Definitions;<br>• Updated Section: CPU and Memory;<br>• Updated Section: Audio PLL Clock;<br>• Updated Section: Absolute Maximum Ratings;<br>• Updated Chapter: Package Information;<br>• Updated Chapter: Learning Resources. |
| 2016.08 | V1.0 | First release. |