

# ESP32 系列芯片

## 技术规格书 版本 4.6

2.4 GHz Wi-Fi + 传统蓝牙 + 低功耗蓝牙 SoC

### 包括:

ESP32-D0WD-V3

ESP32-D0WDR2-V3

ESP32-U4WDH

ESP32-S0WD – [不推荐用于新设计 \(NRND\)](#)

ESP32-D0WD – [不推荐用于新设计 \(NRND\)](#)

ESP32-D0WDQ6 – [不推荐用于新设计 \(NRND\)](#)

ESP32-D0WDQ6-V3 – [不推荐用于新设计 \(NRND\)](#)



# 产品概述

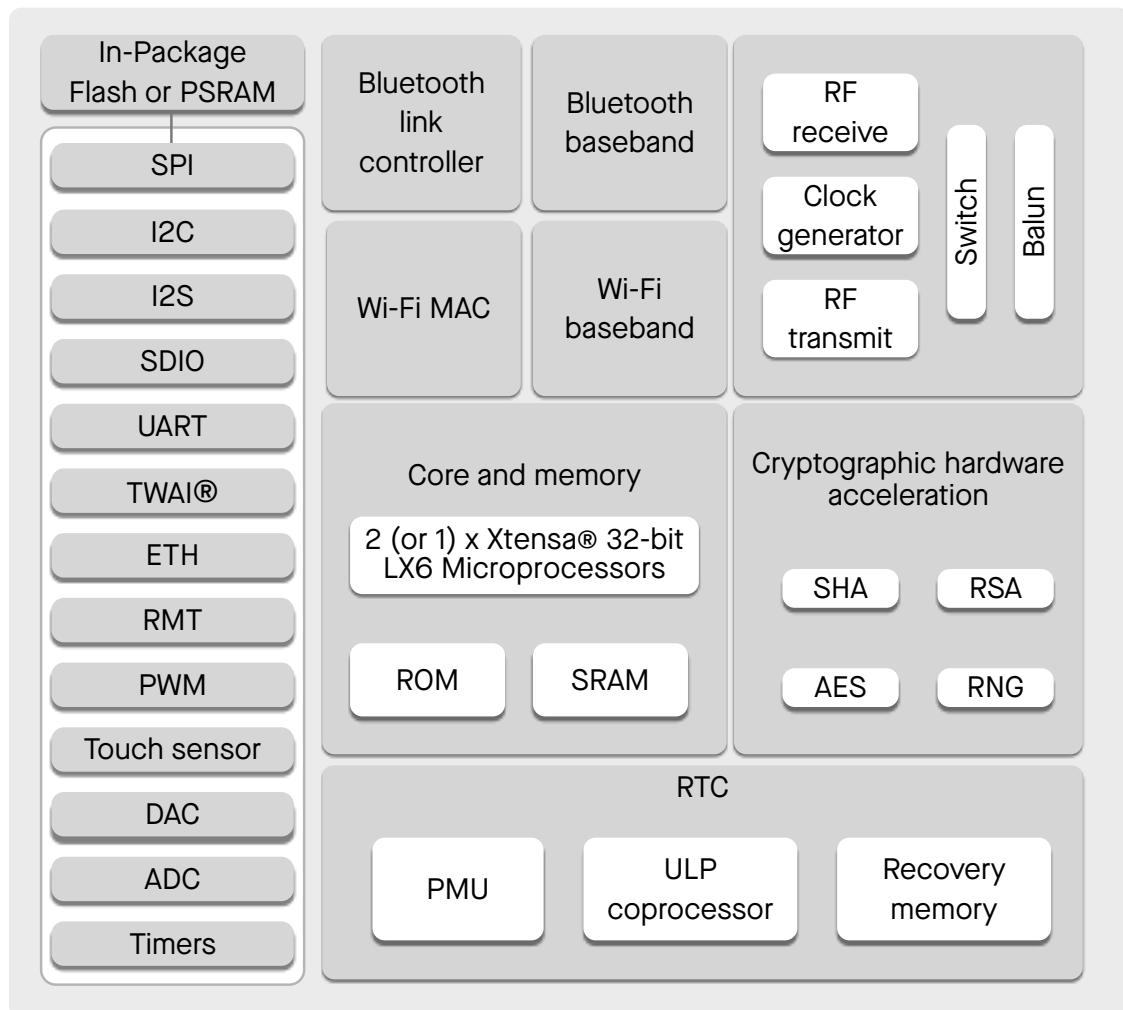
ESP32 是集成 2.4 GHz Wi-Fi 和蓝牙双模的单芯片方案，采用台积电 (TSMC) 低功耗 40 纳米工艺，具有超高的射频性能、稳定性、通用性和可靠性，以及超低的功耗，满足不同的功耗需求，适用于各种应用场景。

目前 ESP32 系列的产品型号包括 ESP32-DOWD-V3、ESP32-DOWDR2-V3、ESP32-U4WDH、ESP32-SOWD ([NRND](#))、ESP32-DOWD ([NRND](#))、ESP32-DOWDQ6 ([NRND](#)) 和 ESP32-DOWDQ6-V3 ([NRND](#))，其中

- ESP32-SOWD ([NRND](#))、ESP32-DOWD ([NRND](#))、ESP32-DOWDQ6 ([NRND](#)) 基于芯片版本 v1.0 或 v1.1；
- ESP32-DOWD-V3、ESP32-DOWDR2-V3、ESP32-U4WDH 和 ESP32-DOWDQ6-V3 ([NRND](#)) 基于芯片版本 v3.0 或 v3.1。

产品型号说明和订购信息请参考章节 [1 ESP32 系列型号对比](#)。有关芯片版本的更多信息，请参考文档 [《ESP32 芯片版本 v3.0 使用指南》](#) 和 [《ESP32 系列芯片勘误表》](#)。

芯片的功能框图如下图所示。



ESP32 功能框图

## 产品特性

### Wi-Fi

- 支持 IEEE 802.11b/g/n 协议
- 802.11n (2.4 GHz) 速度高达 150 Mbps
- 无线多媒体 (WMM)
- 帧聚合 (TX/RX A-MPDU, RX A-MSDU)
- 立即块确认 (Immediate Block ACK)
- 重组 (Defragmentation)
- Beacon 自动监测 (硬件 TSF)
- 4 个虚拟 Wi-Fi 接口
- 同时支持基础结构型网络 (Infrastructure BSS) Station 模式/SoftAP 模式/混杂模式  
请注意 ESP32 在 Station 模式下扫描时，SoftAP 信道会同时改变
- 天线分集

### 蓝牙

- 蓝牙 v4.2 完整标准，包含传统蓝牙 (BR/EDR) 和低功耗蓝牙 (Bluetooth LE)
- 支持标准 Class-1、Class-2 和 Class-3，且无需外部功率放大器
- 增强型功率控制 (Enhanced Power Control)
- 输出功率高达 +9 dBm
- NZIF 接收器具有 -94 dBm 的 BLE 接收灵敏度
- 自适应跳频 (AFH)
- 基于 SDIO/SPI/UART 接口的标准 HCI
- 高速 UART HCI，最高可达 4 Mbps
- 支持蓝牙 4.2 BR/EDR 和 Bluetooth LE 双模 controller
- 同步面向连接/扩展同步面向连接 (SCO/eSCO)
- CVSD 和 SBC 音频编解码算法
- 蓝牙微微网 (Piconet) 和散射网 (Scatternet)
- 支持传统蓝牙和低功耗蓝牙的多设备连接
- 支持同时广播和扫描

### CPU 和存储

- Xtensa® 32-bit LX6 单/双核处理器
- CoreMark® 得分：
  - 单核，主频 240 MHz: 504.85 CoreMark; 2.10 CoreMark/MHz

- 双核，主频 240 MHz: 994.26 CoreMark; 4.14 CoreMark/MHz
- 448 KB ROM
- 520 KB SRAM
- 16 KB RTC SRAM
- QSPI 支持多个 flash/SRAM

## 时钟和定时器

- 内置 8 MHz 振荡器，支持自校准
- 内置 RC 振荡器，支持自校准
- 支持外置 2 MHz 至 60 MHz 的主晶振（如果使用 Wi-Fi/蓝牙功能，则目前仅支持 40 MHz 晶振）
- 支持外置 32 kHz 晶振，用于 RTC，支持自校准
- 2 个定时器群组，每组包括 2 个 64-bit 通用定时器和 1 个主系统看门狗
- 1 个 RTC 定时器
- RTC 看门狗

## 高级外设接口

- 34 个 GPIO 口
  - 5 个作为 strapping 管脚
  - 6 个仅为输入管脚
  - 6 个用于连接封装内 flash/PSRAM (ESP32-D0WD-R2-V3, ESP32-U4WDH)
- 12-bit SAR ADC，多达 18 个通道
- 2 个 8-bit D/A 转换器
- 10 个触摸传感器
- 4 个 SPI
- 2 个 I2S
- 2 个 I2C
- 3 个 UART
- 1 个 Host SD/eMMC/SDIO
- 1 个 Slave SDIO/SPI
- 带有专用 DMA 的以太网 MAC 接口，支持 IEEE 1588
- TWAI<sup>®</sup>，兼容 ISO 11898-1 (CAN 规范 2.0)
- RMT (TX/RX)
- 电机 PWM
- LED PWM，多达 16 个通道

## 功耗管理

- 通过选择时钟频率、占空比、Wi-Fi 工作模式和单独控制内部器件的电源，实现精准电源控制
- 针对典型场景设计的五种功耗模式：Active、Modem-sleep、Light-sleep、Deep-sleep、Hibernation
- Deep-sleep 模式下功耗低至  $10 \mu\text{A}$
- 超低功耗协处理器 (ULP)
- Deep-sleep 模式下 RTC 存储器仍保持工作

## 安全机制

- 安全启动
- flash 加密
- 1024 位 OTP，用户可用的高达 768 位
- 加密硬件加速器：
  - AES
  - Hash (SHA-2)
  - RSA
  - ECC
  - 随机数生成器 (RNG)

## 应用

低功耗芯片 ESP32 专为物联网 (IoT) 设备而设计，应用领域包括：

- 智能家居
- 工业自动化
- 医疗保健
- 消费电子产品
- 智慧农业
- POS 机
- 服务机器人
- 音频设备
- 通用低功耗 IoT 传感器集线器
- 通用低功耗 IoT 数据记录器
- 摄像头视频流传输
- 语音识别
- 图像识别
- SDIO Wi-Fi + 蓝牙网卡
- 触摸和接近感应

说明:

点击链接或扫描二维码确保您使用的是最新版本的文档:

[https://www.espressif.com/documentation/esp32\\_datasheet\\_cn.pdf](https://www.espressif.com/documentation/esp32_datasheet_cn.pdf)



# 目录

<b>产品概述</b>	2
产品特性	3
应用	5
<b>1 ESP32 系列型号对比</b>	11
1.1 命名规则	11
1.2 型号对比	11
<b>2 管脚</b>	12
2.1 管脚布局	12
2.2 管脚概述	14
2.3 IO 管脚	17
2.3.1 GPIO 和 RTC_GPIO 的限制	17
2.4 模拟管脚	17
2.5 电源	17
2.5.1 电源管脚	17
2.5.2 电源管理	18
2.5.3 芯片上电和复位	19
2.6 芯片与 flash/PSRAM 的管脚对应关系	20
<b>3 启动配置项</b>	22
3.1 芯片启动模式控制	23
3.2 内置 LDO (VDD_SDIO) 电压控制	24
3.3 UOTXD 打印控制	25
3.4 SDIO 从机信号输入输出时序控制	25
3.5 JTAG 信号源控制	25
<b>4 功能描述</b>	26
4.1 CPU 和存储	26
4.1.1 CPU	26
4.1.2 片上存储	26
4.1.3 封装外 flash 和 RAM	26
4.1.4 地址映射结构	27
4.1.5 Cache	29
4.2 系统时钟	29
4.2.1 CPU 时钟	29

---

4.2.2	RTC 时钟	29
4.2.3	音频 PLL 时钟	29
4.3	RTC 和低功耗管理	30
4.3.1	电源管理单元 (PMU)	30
4.3.2	超低功耗协处理器	31
4.4	定时器和看门狗	31
4.4.1	通用定时器	31
4.4.2	看门狗定时器	31
4.5	加密硬件加速器	32
4.6	射频和 Wi-Fi	32
4.6.1	2.4 GHz 接收器	32
4.6.2	2.4 GHz 发射器	32
4.6.3	时钟生成器	32
4.6.4	Wi-Fi 射频和基带	33
4.6.5	Wi-Fi MAC	33
4.7	蓝牙	33
4.7.1	蓝牙射频和基带	33
4.7.2	蓝牙接口	34
4.7.3	蓝牙协议栈	34
4.7.4	蓝牙链路控制器	34
4.8	数字外设	35
4.8.1	通用输入/输出接口 (GPIO)	35
4.8.2	串行外设接口 (SPI)	35
4.8.3	通用异步收发器 (UART)	36
4.8.4	I <sub>2</sub> C 接口	36
4.8.5	I <sub>2</sub> S 接口	37
4.8.6	红外遥控	37
4.8.7	脉冲计数控制器	38
4.8.8	LED PWM 控制器	38
4.8.9	电机控制脉宽调制器 (MCPWM)	39
4.8.10	SD/SDIO/MMC 主机控制器	39
4.8.11	SDIO/SPI 从机控制器	40
4.8.12	双线汽车接口	40
4.8.13	以太网 MAC 接口	41
4.9	模拟外设	42
4.9.1	模/数转换器 (ADC)	42
4.9.2	数/模转换器 (DAC)	42
4.9.3	触摸传感器	43
4.10	外设管脚分配	44

## 5 电气特性

5.1	绝对最大额定值	49
5.2	建议工作条件	49
5.3	直流电气特性 (3.3 V, 25 °C)	50
5.4	Active 模式下的 RF 功耗	50
5.5	可靠性	51

5.6	Wi-Fi 射频	51
5.7	经典蓝牙射频	52
5.7.1	接收器 - 基础数据率 (BR)	52
5.7.2	发射器 - 基础数据率 (BR)	52
5.7.3	接收器 - 增强数据率 (EDR)	53
5.7.4	发射器 - 增强数据率 (EDR)	53
5.8	低功耗蓝牙射频	54
5.8.1	接收器	54
5.8.2	发射器	55
<b>6</b>	<b>封装</b>	56
<b>7</b>	<b>相关文档和资源</b>	57
<b>附录 A - ESP32 管脚清单</b>		58
A.1.	管脚清单说明	58
A.2.	GPIO_Matrix	60
A.3.	Ethernet_MAC	65
A.4.	IO_MUX	65
<b>修订历史</b>		67

# 表格

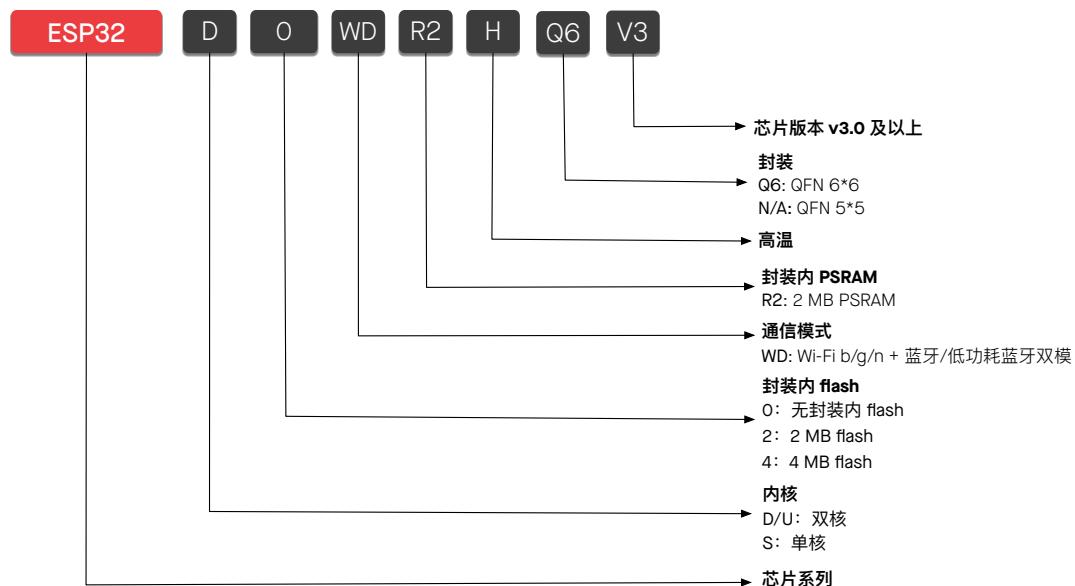
1-1	ESP32 系列芯片对比	11
2-1	管脚概述	14
2-2	模拟管脚	17
2-3	电源管脚	18
2-4	上电和复位时序参数说明	19
2-5	芯片与封装内 flash/PSRAM 的管脚对应关系	20
2-6	ESP32 和封装外 flash/PSRAM 的连接关系	20
3-1	Strapping 管脚的默认配置	22
3-2	Strapping 管脚的时序参数说明	23
3-3	芯片启动模式控制	23
3-4	UOTXD 打印控制	25
3-5	SDIO 从机信号输入输出时序控制	25
4-1	存储器和外设地址映射	27
4-2	不同功耗模式下的功耗	30
4-3	ADC 特性	42
4-4	ADC 校准结果	42
4-5	ESP32 上的电容式传感 GPIO	43
4-6	外设管脚分配	44
5-1	绝对最大额定值	49
5-2	建议工作条件	49
5-3	直流电气特性 (3.3 V, 25 °C)	50
5-4	射频功耗参数	50
5-5	可靠性	51
5-6	Wi-Fi 射频特性	51
5-7	接收器特性 - 基础数据率 (BR)	52
5-8	发射器特性 - 基础数据率 (BR)	52
5-9	接收器特性 - 增强数据率 (EDR)	53
5-10	发射器特性 - 增强数据率 (EDR)	53
5-11	低功耗蓝牙接收器特性	54
5-12	低功耗蓝牙发射器特性	55
7-1	管脚清单说明	58
7-2	GPIO_Matrix	60
7-3	Ethernet_MAC	65

## 插图

1-1	ESP32 系列芯片命名规则	11
2-1	ESP32 管脚布局（封装为 QFN 6*6，顶视图）	12
2-2	ESP32 管脚布局（封装为 QFN 5*5，顶视图）	13
2-3	ESP32 电源管理	18
2-4	上电和复位时序参数图	19
3-1	Strapping 管脚的时序参数图	23
3-2	芯片启动流程	24
4-1	地址映射结构	27
6-1	QFN48 (6×6 mm) 封装	56
6-2	QFN48 (5×5 mm) 封装	56

# 1 ESP32 系列型号对比

## 1.1 命名规则



## 1.2 型号对比

表 1-1. ESP32 系列芯片对比

订购代码 <sup>1</sup>	内核	芯片版本 <sup>2</sup>	封装内 flash/PSRAM	封装	VDD_SDIO 电压
ESP32-DOWD-V3	双核	v3.0/v3.1 <sup>4</sup>	—	QFN 5*5	1.8 V/3.3 V
ESP32-DOWDR2-V3	双核	v3.0/v3.1 <sup>4</sup>	2 MB PSRAM	QFN 5*5	3.3 V
ESP32-U4WDH	双核 <sup>3</sup>	v3.0/v3.1 <sup>4</sup>	4 MB flash <sup>6</sup>	QFN 5*5	3.3 V
ESP32-DOWDQ6-V3 ( <u>NRND</u> )	双核	v3.0/v3.1 <sup>4</sup>	—	QFN 6*6	1.8 V/3.3 V
ESP32-DOWD ( <u>NRND</u> )	双核	v1.0/v1.1 <sup>5</sup>	—	QFN 5*5	1.8 V/3.3 V
ESP32-DOWDQ6 ( <u>NRND</u> )	双核	v1.0/v1.1 <sup>5</sup>	—	QFN 6*6	1.8 V/3.3 V
ESP32-SOWD ( <u>NRND</u> )	单核	v1.0/v1.1 <sup>5</sup>	—	QFN 5*5	1.8 V/3.3 V

<sup>1</sup> 所有芯片型号均支持 Wi-Fi b/g/n + 蓝牙/低功耗蓝牙双模通信模式。更多关于芯片丝印和包装的信息，请参考章节 [6 封装](#)。

<sup>2</sup> ESP32 芯片版本信息及其区分方式见 [《ESP32 系列芯片勘误表》](#)。

<sup>3</sup> ESP32-U4WDH 芯片将从单核升级为双核，见 [PCN-2021-021](#)。

<sup>4</sup> 该芯片型号使用的芯片版本将由 v3.0 变更为 v3.1，见 [PCN20220901](#)。

<sup>5</sup> 该芯片型号使用的芯片版本将由 v1.0 变更为 v1.1，见 [PCN20220901](#)。

<sup>6</sup> Flash 支持：

- 至少 10 万次编程/擦除周期
- 至少 20 年数据保留时间

## 2 管脚

### 2.1 管脚布局

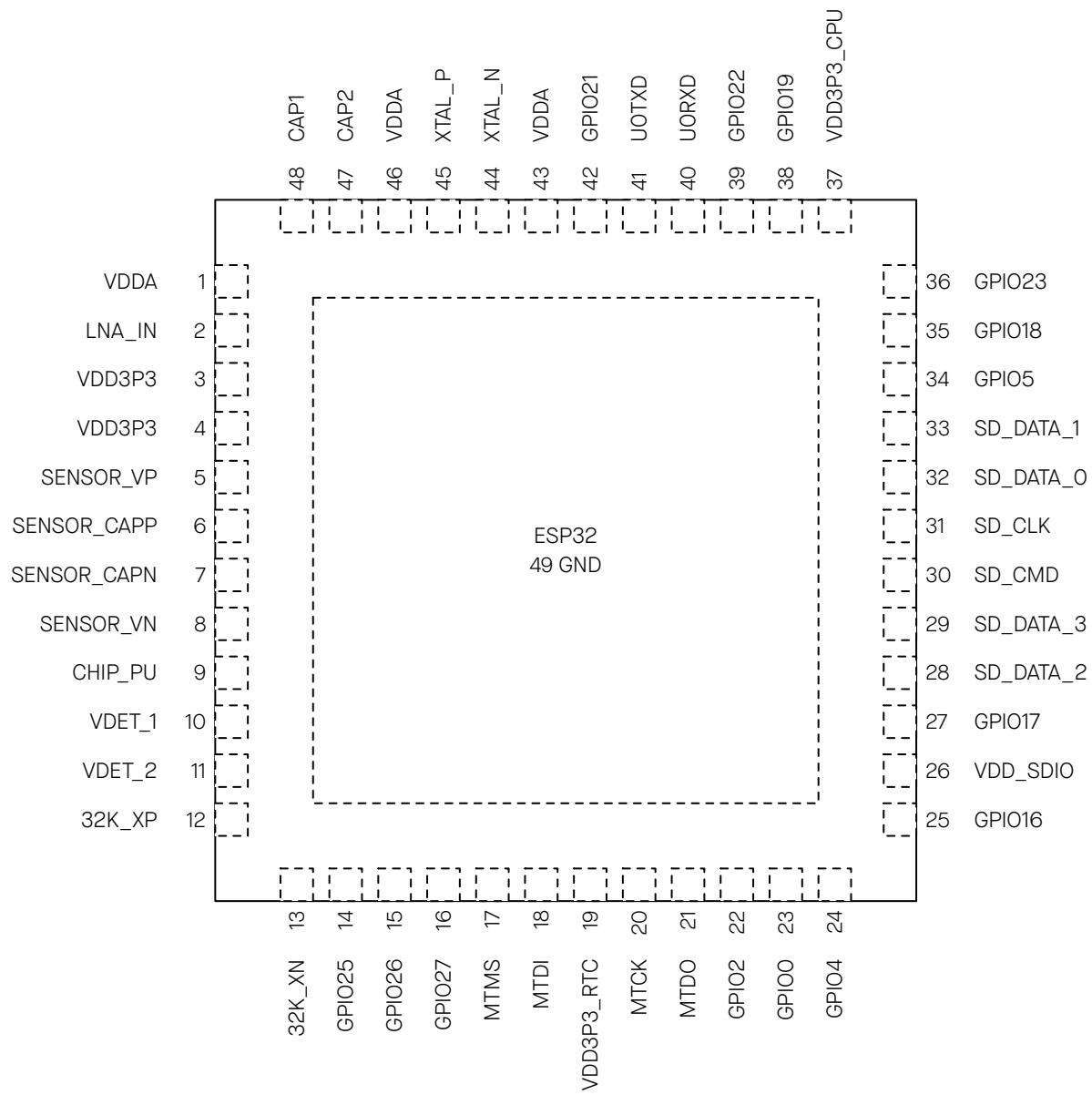


图 2-1. ESP32 管脚布局 (封装为 QFN 6\*6, 顶视图)

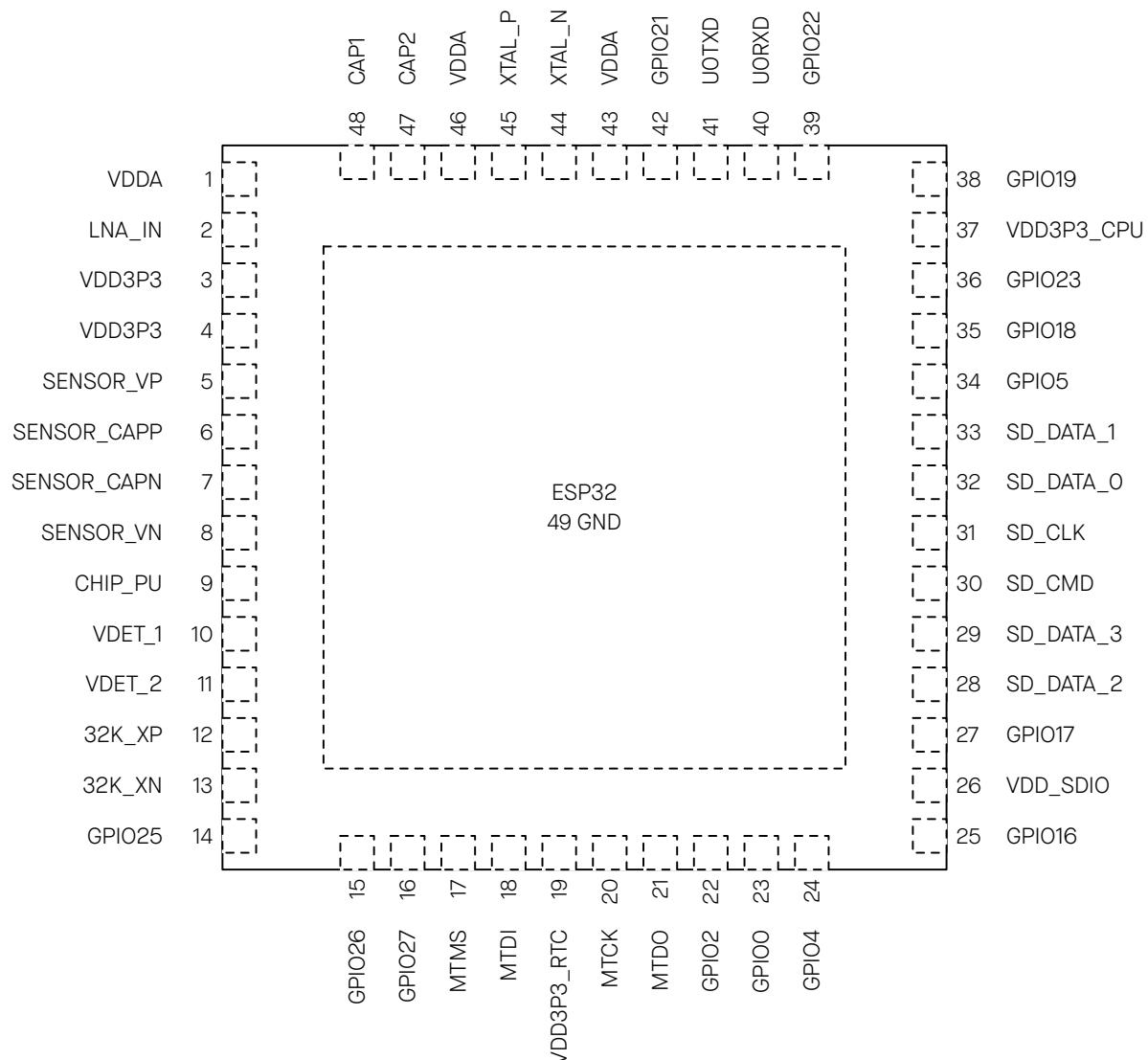


图 2-2. ESP32 管脚布局 (封装为 QFN 5\*5, 顶视图)

## 2.2 管脚概述

表 2-1. 管脚概述

名称	No.	类型	功能			
模拟						
VDDA	1	P	模拟电源 (2.3 V ~ 3.6 V)			
LNA_IN	2	I/O	射频输入和输出			
VDD3P3	3	P	模拟电源 (2.3 V ~ 3.6 V)			
VDD3P3	4	P	模拟电源 (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	高电平: 芯片使能; 低电平: 芯片关闭; 注意: 不能让 CHIP_PU 管脚浮空。			
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 晶振输入)			
32K_XN	13	I/O	GPIO33, ADC1_CH5, RTC_GPIO8, TOUCH8, 32K_XN (32.768 kHz 晶振输出)			
GPIO25	14	I/O	GPIO25, ADC2_CH8, RTC_GPIO6, DAC_1, EMAC_RXDO			
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	RTC IO 电源输入 (2.3 V ~ 3.6 V)			
MTCK	20	I/O	GPIO13, ADC2_CH4, RTC_GPIO14, TOUCH4, EMAC_RX_ER, HSPIID, HS2_DATA3, SD_DATA3, MTCK			
MTDO	21	I/O	GPIO15, ADC2_CH3, RTC_GPIO13, TOUCH3, EMAC_RXD3, HSPICSO, HS2_CMD, SD_CMD, MTDO			

名称	No.	类型	功能						
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_CHO,	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	1.8 V 或 VDD3P3_RTC 电源输出						
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,	SPICSO		
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,	VSPICSO,	EMAC_RX_CLK			
GPIO18	35	I/O	GPIO18,	HS1_DATA7,	VSPICLK				
GPIO23	36	I/O	GPIO23,	HS1_STROBE,	VSPID				
VDD3P3_CPU	37	P	CPU IO 电源输入 (1.8 V ~ 3.6 V)						
GPIO19	38	I/O	GPIO19,	UOCTS,	VSPIQ,	EMAC_TXDO			
GPIO22	39	I/O	GPIO22,	UORTS,	VSPIWP,	EMAC_TXD1			
UORXD	40	I/O	GPIO3,	UORXD,	CLK_OUT2				
UOTXD	41	I/O	GPIO1,	UOTXD,	CLK_OUT3,	EMAC_RXD2			
GPIO21	42	I/O	GPIO21,		VSPIHD,	EMAC_TX_EN			
模拟									
VDDA	43	P	模拟电源 (2.3 V ~ 3.6 V)						
XTAL_N	44	O	外部主晶振输出						
XTAL_P	45	I	外部主晶振输入						
VDDA	46	P	模拟电源 (2.3 V ~ 3.6 V)						
CAP2	47	I	并联 3.3 nF (10%) 电容和 20 kΩ 电阻到 CAP1						

名称	No.	类型	功能
CAP1	48	I	串联 10 nF 电容到地
GND	49	P	接地

表 2-1 管脚概述 说明：

## 1. 功能名称：

CLK_OUT...	时钟输出
SPICLK	
HSPICLK	SPI 时钟 (Clock) 信号
VSPICLK	
HS..._CLK	SDIO 主机时钟 (Clock) 信号
SD_CLK	SDIO 从机时钟 (Clock) 信号
EMAC_TX_CLK	
EMAC_RX_CLK	EMAC 时钟 (Clock) 信号
U..._RTS	
U..._CTS	UART0/1/2 硬件流控信号
U..._RXD	
U..._TXD	UART0/1/2 接收/发送信号
GPIO...	通用输入输出，信号通过 GPIO 交换矩阵传输。更多关于 GPIO 交换矩阵的信息，详见 <a href="#">《ESP32 技术参考手册》&gt; 章节 IO MUX 和 GPIO 交换矩阵 (GPIO, IO MUX)</a> 。

- 高亮的单元格，详见章节 [2.3.1 GPIO 和 RTC\\_GPIO 的限制](#)。
- IO\_MUX、Ethernet MAC、GPIO Matrix 请见附件[管脚清单](#)。

## 2.3 IO 管脚

### 2.3.1 GPIO 和 RTC\_GPIO 的限制

ESP32 的所有 IO 管脚都具备 GPIO 功能，其中部分还具有 RTC\_GPIO 功能。不过，这些 IO 管脚是多功能的，可以根据需求进行不同功能的配置，也有一些管脚有使用限制，需要特别注意。

表 2-1 中，部分管脚功能有高亮标记，具体如下：

- **GPIO** – 仅作输入管脚，没有输出功能，不带输出驱动器或内置上拉/下拉电路。
- **GPIO** – 用于与封装内 flash/PSRAM 通讯，不建议作其他用途。更多信息，详见章节 2.6 芯片与 flash/PSRAM 的管脚对应关系。
- **GPIO** – 具有以下重要功能之一：
  - **Strapping 管脚** – 启动时逻辑电平需为特定值。详见章节 3 启动配置项。
  - **JTAG 接口** – 通常用于调试功能。
  - **UART 接口** – 通常用于调试功能。

附录 A.1 – 管脚清单说明 也可参考。

## 2.4 模拟管脚

表 2-2. 模拟管脚

管脚序号	管脚名称	管脚类型	管脚功能
1	LNA_IN	I/O	低噪声放大器 (RF LNA) 输入信号，功率放大器 (RF PA) 输出信号
4	CHIP_PU	I	高电平：芯片使能（上电）； 低电平：芯片关闭（掉电）； 注意不能让 CHIP_PU 管脚浮空
53	XTAL_N	—	连接芯片有源晶振或无源晶振的外部时钟输入/输出。
54	XTAL_P	—	P/N 指差分时钟负极/正极端。

## 2.5 电源

### 2.5.1 电源管脚

ESP32 的数字管脚可分为 3 种不同的电源域：

- VDD3P3\_RTC
- VDD3P3\_CPU
- VDD\_SDIO

表 2-3. 电源管脚

管脚序号	管脚名称	方向	电源	
			电源域 / 其他	IO 管脚
3	VDD3P3	输入	模拟电源域	
4	VDD3P3	输入	模拟电源域	
19	VDD3P3_RTC <sup>1</sup>	输入	RTC 及部分数字电源域	RTC IO
26	VDD3P3_SDIO <sup>3</sup>	输入/输出	模拟电源域	
37	VDD3P3_CPU <sup>2</sup>	输入	数字电源域	数字 IO
43	VDDA	输入	模拟电源域	
46	VDDA	输入	模拟电源域	
49	GND	-	外部接地	

<sup>1</sup> VDD3P3\_RTC 同时是 RTC 和 CPU 的输入电源。

<sup>2</sup> VDD3P3\_CPU 是 CPU 的输入电源。

<sup>3</sup> VDD\_SDIO 与一个内置 LDO 的输出相连，该内置 LDO 的输入是 VDD3P3\_RTC。当 VDD\_SDIO 与 VDD3P3\_RTC 连接在相同的电源上，内置 LDO 会自动关闭。

## 2.5.2 电源管理

电源管理如图 2-3 ESP32 电源管理 所示。

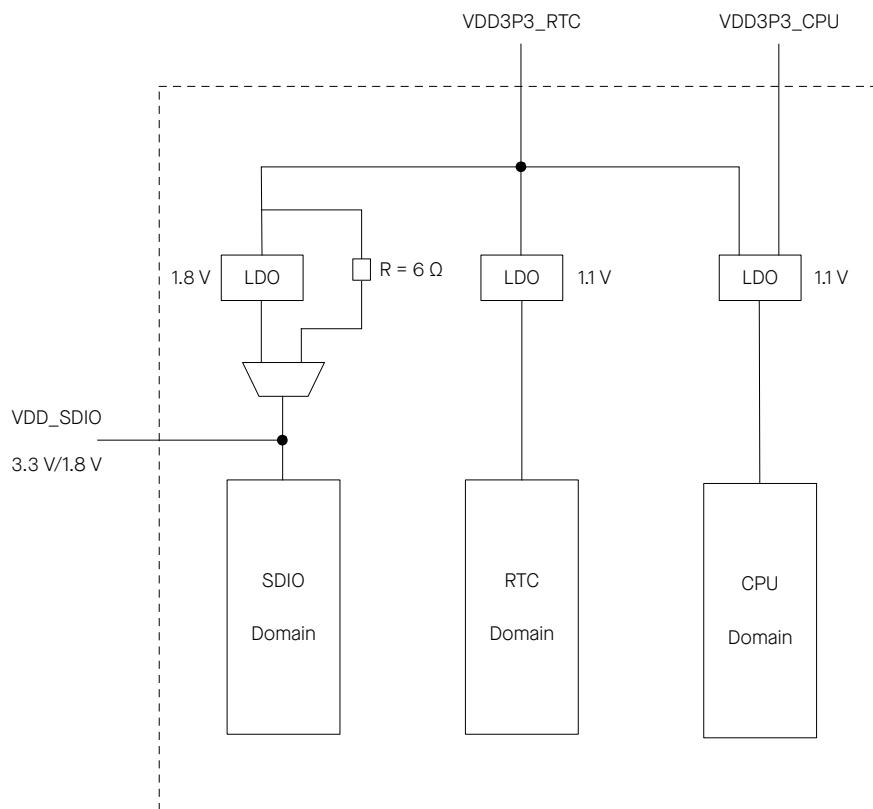


图 2-3. ESP32 电源管理

内置 LDO 可被配置成 1.8 V 或与 VDD3P3\_RTC 相同的电压。在 Deep-sleep 模式下，为了使 flash 电流降到最低，可以通过软件关闭内置 LDO。

### 2.5.3 芯片上电和复位

芯片上电后，其电源轨需要一点时间方可稳定。之后，用于上电和复位的管脚 CHIP\_PU 拉高，激活芯片。更多关于 CHIP\_PU 及上电和复位时序的信息，请见图 2-4 和表 2-4。

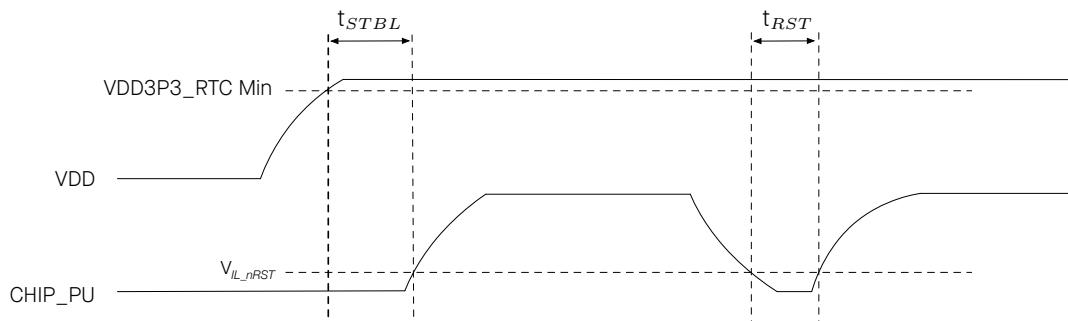


图 2-4. 上电和复位时序参数图

表 2-4. 上电和复位时序参数说明

参数	说明	最小值 ( $\mu\text{s}$ )
$t_{STBL}$	CHIP_PU 管脚拉高激活芯片前，3.3 V 系统电源达到稳定所需的时间	50
$t_{RST}$	CHIP_PU 电平低于 $V_{IL\_nRST}$ (具体数值参考表 5-3) 从而复位芯片的时间	50

- 在实际应用中，如果需要快速反复开关 VDD33，且 VDD33 外围电路上有大电容，且 CHIP\_PU 与 VDD33 相连，在先关后开的动作中，CHIP\_PU 电平降到 0 的过程会非常缓慢，在下一次重新上电时，CHIP\_PU 还来不及降到足够低的电平，从而导致不能充分复位芯片。此时需要额外的放电电路来加速 VDD33 上大电容的放电。
- 当使用电池给 ESP32 系列芯片和模组供电时，为避免电池电压过低导致芯片进入异常状态不能正常启动，一般推荐外接 Power Supply Supervisor。建议检测到供给 ESP32 的电压低于 2.3 V 时将 ESP32 的 CHIP\_PU 脚拉低。

#### 关于电源电压的说明：

- ESP32 工作电压范围为 2.3 V ~ 3.6 V；使用单电源供电时，建议电源电压为 3.3 V，输出电流需要达到 500 mA 及以上。
- PSRAM 和 flash 均由 VDD\_SDIO 进行供电。如果芯片已经内置了 flash，则 VDD\_SDIO 的电压既已确定。此时，如果芯片需要另外连接外部 PSRAM 时，则外部 PSRAM 电压应与内置 flash 的工作电压匹配。上述情况同时也适用于内置了 PSRAM，但需要外接 flash 的情况。
- 当使用 VDD\_SDIO 的 1.8 V 模式给 flash/PSRAM 等外设供电时，需要在 VDD\_SDIO 上接个对地  $2\text{ k}\Omega$  的电阻，电路原理图可参考 [《ESP32 硬件设计指南》](#)。
- 在使用这些数字电源为外设供电的使用场景中，比如 3.3 V flash，电源电压还应满足外设的使用要求。

## 2.6 芯片与 flash/PSRAM 的管脚对应关系

表 2-5 列出了芯片与封装内 flash/PSRAM 的管脚对应关系。这些管脚不建议用于其它功能。

芯片与封装外 flash/PSRAM 的推荐连接管脚见表 2-6。

表 2-5. 芯片与封装内 flash/PSRAM 的管脚对应关系

ESP32-U4WDH	封装内 Flash (4 MB)
SD_DATA_1	IO0/DI
GPIO17	IO1/DO
SD_DATA_0	IO2/WP#
SD_CMD	IO3/HOLD#
SD_CLK	CLK
GPIO16	CS#
GND	VSS
VDD_SDIO <sup>1</sup>	VDD
<b>ESP32-D0WDR2-V3 封装内 PSRAM (2 MB)</b>	
SD_DATA_1	SIO0/SI
SD_DATA_0	SIO1/SO
SD_DATA_3	SIO2
SD_DATA_2	SIO3
SD_CLK	SCLK
GPIO16 <sup>2</sup>	CE#
GND	VSS
VDD_SDIO <sup>1</sup>	VDD

表 2-6. ESP32 和封装外 flash/PSRAM 的连接关系

芯片管脚	封装外 flash
SD_DATA_1/SPIID	IO0/DI
SD_DATA_0/SPIQ	IO1/DO
SD_DATA_3/SPIWP	IO2/WP#
SD_DATA_2/SPIHD	IO3/HOLD#
SD_CLK	CLK
SD_CMD	CS#
GND	VSS
VDD_SDIO	VDD
芯片管脚	封装外 PSRAM
SD_DATA_1	SIO0/SI
SD_DATA_0	SIO1/SO
SD_DATA_3	SIO2
SD_DATA_2	SIO3
SD_CLK/GPIO17 <sup>3</sup>	SCLK
GPIO16 <sup>2</sup>	CE#

见下页

表 2-6 - 接上页

芯片管脚	封装外 PSRAM
GND	VSS
VDD_SDIO	VDD

**说明:**

1. ESP32-U4WDH 的封装内 flash 以及 ESP32-DOWDR2-V3 的封装内 PSRAM 均运行在 3.3V 电压下, 此时, VDD\_SDIO 会通过  $6\Omega$  的电阻从 VDD3P3\_RTC 取电, 见图 2-3。
2. 使用 GPIO16 连接 PSRAM 的 CE# 信号时, 请在 GPIO16 处增加一个上拉电阻。请参考 [《ESP32-WROVER-E 技术规格书》](#) > 电路原理图。
3. 连接外部 PSRAM 的 SCLK 信号时, 可使用芯片的 SD\_CLK 管脚或 GPIO17 管脚。
  - 如果选择使用 SD\_CLK 管脚, 则可以节省一个 GPIO 管脚, 即 GPIO17。节省出来的 GPIO 可用于其它用途。该连接方式已通过内部测试, 但相关认证尚未完成。
  - 或者选择使用 GPIO17 管脚。该连接方式已通过相关认证, 请参考 [ESP32-WROVER-E](#) 证书。

请根据具体的应用需求, 选择合适的管脚。

### 3 启动配置项

芯片在上电或硬件复位时，可以通过 Strapping 管脚和 eFuse 位配置如下启动参数，无需微处理器的参与：

- 芯片启动模式
  - Strapping 管脚：GPIO0 和 GPIO2
- 内置 LDO (VDD\_SDIO) 电压
  - Strapping 管脚：MTDI
  - eFuse 参数：EFUSE\_SDIO\_FORCE 和 EFUSE\_SDIO\_TIEH
- UOTXD 打印
  - Strapping 管脚：MTDO
- SDIO 从机信号输入输出时序
  - Strapping 管脚：MTDO 和 GPIO5
- JTAG 信号源
  - Strapping 管脚：MTCK, MTMS, MTDI 和 MTDO
  - eFuse 参数：EFUSE\_DISABLE\_JTAG

上述 eFuse 位的默认值均为 0，也就是说没有烧写过。eFuse 只能烧写一次，一旦烧写为 1，便不能恢复为 0。有关烧写 eFuse 的信息，请参考 [《ESP32 技术参考手册》> 章节 eFuse 控制器](#)。

上述 strapping 管脚如果没有连接任何电路或连接的电路处于高阻抗状态，则其默认值（即逻辑电平值）取决于管脚内部弱上拉/下拉电阻在复位时的状态。

表 3-1. Strapping 管脚的默认配置

Strapping 管脚	默认配置	值
GPIO0	上拉	1
GPIO2	下拉	0
MTDI	下拉	0
MTDO	上拉	1
GPIO5	上拉	1

要改变 strapping 管脚的值，可以连接外部下拉/上拉电阻。如果 ESP32 用作主机 MCU 的从设备，strapping 管脚的电平也可通过主机 MCU 控制。

所有 strapping 管脚都有锁存器。系统复位时，锁存器采样并存储相应 strapping 管脚的值，一直保持到芯片掉电或关闭。锁存器的状态无法用其他方式更改。因此，strapping 管脚的值在芯片工作时一直可读取，strapping 管脚在芯片复位后作为普通 IO 管脚使用。

Strapping 管脚的信号时序需遵循表 3-2 和图 3-1 所示的 建立时间和 保持时间。

表 3-2. Strapping 管脚的时序参数说明

参数	说明	最小值 (ms)
$t_{SU}$	建立时间，即拉高 CHIP_PU 激活芯片前，电源轨达到稳定所需的时间	0
$t_H$	保持时间，即 CHIP_PU 已拉高、strapping 管脚变为普通 IO 管脚开始工作前，可读取 strapping 管脚值的时间	1

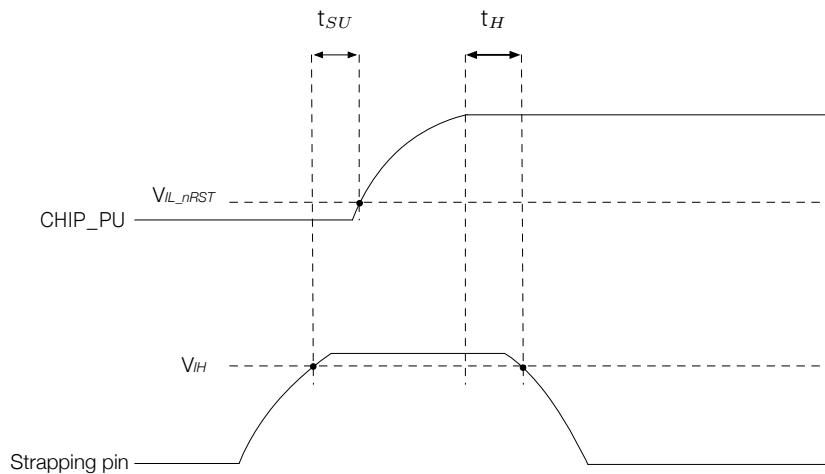


图 3-1. Strapping 管脚的时序参数图

### 3.1 芯片启动模式控制

复位释放后，GPIO0 和 GPIO2 共同决定启动模式。详见表 3-3 芯片启动模式控制。

表 3-3. 芯片启动模式控制

启动模式	GPIO0	GPIO2
SPI Boot 模式 <sup>1</sup>	1	任意值
Joint Download Boot 模式 <sup>2</sup>	0	0

<sup>1</sup> 加粗表示默认值和默认配置。

<sup>2</sup> Joint Download Boot 模式下支持以下下载方式：

- SDIO Download Boot
- UART Download Boot

在 Joint Download Boot 模式下，芯片启动的具体流程见图 3-2。

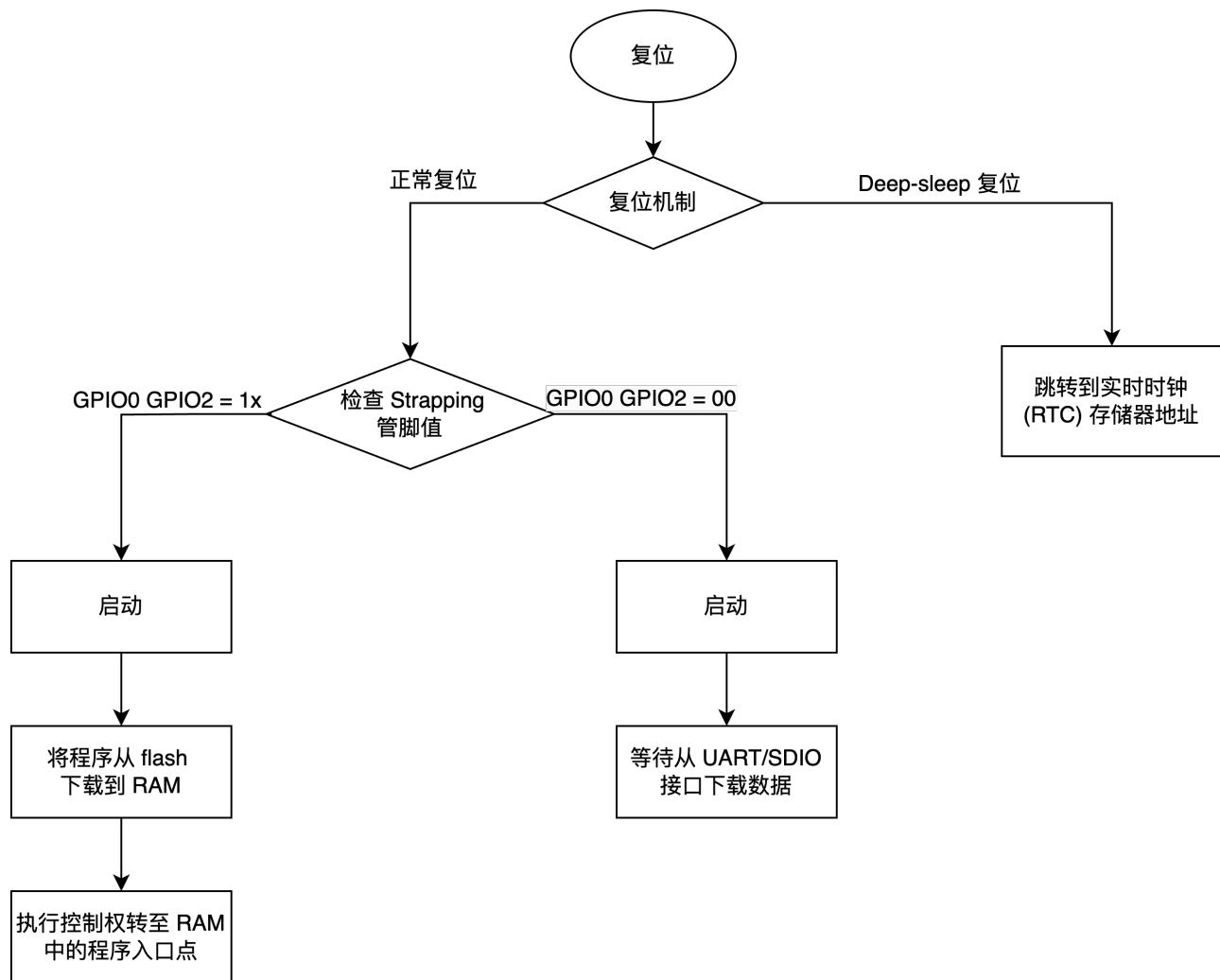


图 3-2. 芯片启动流程

`uart_download_dis` 可用于控制启动模式的具体行为：

如果此 `uart_download_dis` 设置为 1，则永久禁用下载启动模式（仅 ESP32 ECO V3 支持）。

## 3.2 内置 LDO (VDD\_SDIO) 电压控制

ESP32 系列芯片所需的 VDD\_SDIO 电压请参考表 1-1 型号对比。

芯片复位时，MTDI 可用于选择 VDD\_SDIO 电压：

- MTDI = 0 时（默认），VDD\_SDIO 由 VDD3P3\_RTC 供电（电压典型值为 3.3 V）；更多信息请参考章节 2.5.2 电源管理。
- MTDI = 1 时，VDD\_SDIO 可选择由内置 LDO 供电（电压为 1.8 V）。

`EFUSE_SDIO_FORCE` 设置为 1 时，可关闭上述功能。此时 VDD\_SDIO 电压由 `EFUSE_SDIO_TIEH` 的值决定：

- `EFUSE_SDIO_TIEH` = 0 时，VDD\_SDIO 连接 1.8 V LDO；
- `EFUSE_SDIO_TIEH` = 1 时，VDD\_SDIO 连接 VDD3P3\_RTC。

### 3.3 UOTXD 打印控制

在系统启动过程中，MTDO 可用于控制 UOTXD 打印，如表 3-4 所示。

表 3-4. UOTXD 打印控制

UOTXD 打印控制	MTDO
使能 <sup>1</sup>	1
关闭	0

<sup>1</sup> 加粗表示默认值和默认配置。

### 3.4 SDIO 从机信号输入输出时序控制

MTDO 和 GPIO5 管脚可用于控制 SDIO 从机信号输入输出时序。详见表 3-5 SDIO 从机信号输入输出时序控制。

表 3-5. SDIO 从机信号输入输出时序控制

沿控制	MTDO	GPIO5
下降沿采样下降沿输出	0	0
下降沿采样上升沿输出	0	1
上升沿采样下降沿输出	1	0
上升沿采样上升沿输出	1	1

<sup>1</sup> 加粗表示默认值和默认配置。

### 3.5 JTAG 信号源控制

在系统启动早期阶段，MTCK，MTMS，MTDI 和 MTDO 可用于控制 JTAG 信号源。

设置 EFUSE\_DISABLE\_JTAG 为 1 可以关闭 JTAG 信号源。

## 4 功能描述

### 4.1 CPU 和存储

#### 4.1.1 CPU

ESP32 搭载低功耗 Xtensa® LX6 32-bit 单/双核处理器，具有以下特性：

- 7 级流水线架构，支持高达 240 MHz 的时钟频率（除 ESP32-SOWD ([NRND](#)) 为 160 MHz）
- 16-bit/24-bit 指令集提供高代码密度
- 支持浮点单元 (FPU)
- 支持 DSP 指令，例如 32-bit 乘法器、32-bit 除法器和 40-bit 累加乘法器 (MAC)
- 支持来自约 70 个中断源的 32 个中断向量

单/双核处理器接口包括：

- Xtensa RAM/ROM 指令和数据接口
- 用于快速访问外部寄存器的 Xtensa 本地存储接口
- 具有内外中断源的中断接口
- 用于调试的 JTAG 接口

有关 Xtensa® 指令集架构 (ISA) 的说明可以参考 [Xtensa® Instruction Set Architecture \(ISA\) Summary](#)。

#### 4.1.2 片上存储

ESP32 片上存储包括：

- 448 KB 的 ROM，用于程序启动和内核功能调用
- 用于数据和指令存储的 520 KB 片上 SRAM
- RTC 快速存储器，为 8 KB 的 SRAM，可以在 Deep-sleep 模式下 RTC 启动时用于数据存储以及被主 CPU 访问
- RTC 慢速存储器，为 8 KB 的 SRAM，可以在 Deep-sleep 模式下被协处理器访问
- 1 Kbit 的 eFuse，其中 256 bit 为系统专用 (MAC 地址和芯片设置)；其余 768 bit 保留给用户程序，这些程序包括 flash 加密和芯片 ID
- 封装内 flash 或 PSRAM

**说明：**

ESP32 芯片是否有封装内 flash/PSRAM，以及封装内 flash/PSRAM 的大小与芯片型号有关。详见章节 [1 ESP32 系列型号对比](#)。

#### 4.1.3 封装外 flash 和 RAM

ESP32 支持多个外部 QSPI flash 和静态随机存储器 (SRAM)。详情可参考 [《ESP32 技术参考手册》> SPI 控制器章节](#)。ESP32 还支持基于 AES 的硬件加解密功能，从而保护开发者 flash 中的程序和数据。

ESP32 可通过高速缓存访问封装外 QSPI flash 和 SRAM:

- 封装外 flash 可以同时映射到 CPU 指令和只读数据空间。外部 flash 最大可支持 16 MB。
  - 当映射到 CPU 指令空间时，一次最多可映射 11 MB + 248 KB。如果一次映射超过 3 MB + 248 KB，则 cache 性能可能由于 CPU 的推测性读取而降低。
  - 当映射到只读数据空间时，一次最多可以映射 4 MB。支持 8-bit、16-bit 和 32-bit 读取。
- 封装外 SRAM 可映射到 CPU 数据空间。封装外 SRAM 最大可支持 8 MB。一次最多可映射 4 MB。支持 8-bit、16-bit 和 32-bit 访问。

**说明：**

芯片启动完成后，软件可以自定义封装外 SRAM 或 flash 到 CPU 地址空间的映射。

#### 4.1.4 地址映射结构

ESP32 的地址映射结构如图 4-1 所示。ESP32 存储器和外设地址映射如表 4-1 所示。

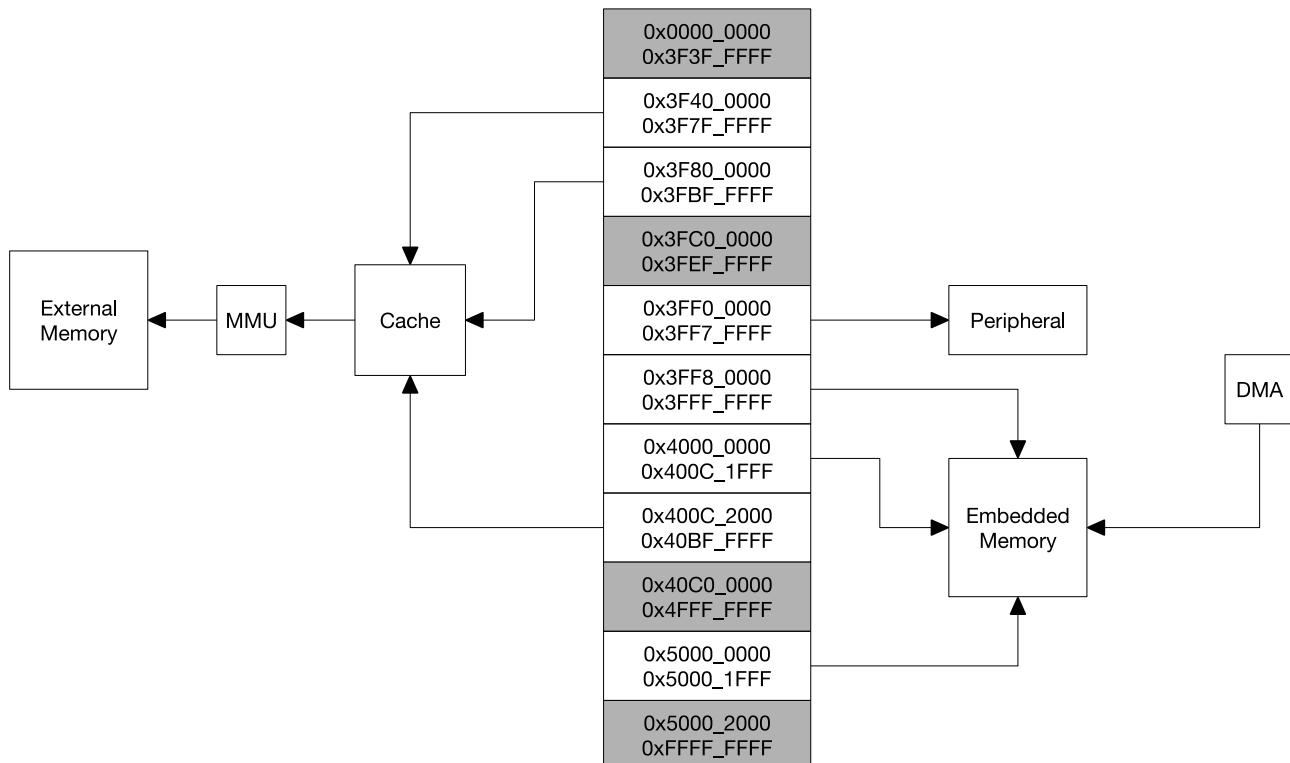


图 4-1. 地址映射结构

表 4-1. 存储器和外设地址映射

种类	目标	开始地址	结束地址	大小
片上存储器	内部 ROM 0	0x4000_0000	0x4005_FFFF	384 KB
	内部 ROM 1	0x3FF9_0000	0x3FF9_FFFF	64 KB
	内部 SRAM 0	0x4007_0000	0x4009_FFFF	192 KB
	内部 SRAM 1	0x3FFE_0000	0x3FFF_FFFF	128 KB
		0x400A_0000	0x400B_FFFF	

种类	目标	开始地址	结束地址	大小
片外存储器	内部 SRAM 2	0x3FFA_E000	0x3FFD_FFFF	200 KB
	RTC 快速存储器	0x3FF8_0000	0x3FF8_1FFF	8 KB
		0x400C_0000	0x400C_1FFF	
	RTC 慢速存储器	0x5000_0000	0x5000_1FFF	8 KB
片外存储器	外部 flash	0x3F40_0000	0x3F7F_FFFF	4 MB
		0x400C_2000	0x40BF_FFFF	11 MB+248 KB
	片外 RAM	0x3F80_0000	0x3FBF_FFFF	4 MB
外设	DPort 寄存器	0x3FF0_0000	0x3FF0_0FFF	4 KB
	AES 加速器	0x3FF0_1000	0x3FF0_1FFF	4 KB
	RSA 加速器	0x3FF0_2000	0x3FF0_2FFF	4 KB
	SHA 加速器	0x3FF0_3000	0x3FF0_3FFF	4 KB
	安全启动	0x3FF0_4000	0x3FF0_4FFF	4 KB
	Cache MMU Table	0x3FF1_0000	0x3FF1_3FFF	16 KB
	PID 控制器	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 控制器	0x3FF5_A000	0x3FF5_AFFF	4 KB
	Flash 加密	0x3FF5_B000	0x3FF5_BFFF	4 KB
	PWM0	0x3FF5_E000	0x3FF5_EFFF	4 KB
	TIMG0	0x3FF5_F000	0x3FF5_FFFF	4 KB
	TIMG1	0x3FF6_0000	0x3FF6_0FFF	4 KB
	SPI2	0x3FF6_4000	0x3FF6_4FFF	4 KB
	SPI3	0x3FF6_5000	0x3FF6_5FFF	4 KB
外设	SYSCON	0x3FF6_6000	0x3FF6_6FFF	4 KB
	I2C1	0x3FF6_7000	0x3FF6_7FFF	4 KB
	SDMMC	0x3FF6_8000	0x3FF6_8FFF	4 KB
	EMAC	0x3FF6_9000	0x3FF6_AFFF	8 KB
	TWAI	0x3FF6_B000	0x3FF6_BFFF	4 KB
	PWM1	0x3FF6_C000	0x3FF6_CFFF	4 KB
	I2S1	0x3FF6_D000	0x3FF6_DFFF	4 KB

种类	目标	开始地址	结束地址	大小
	UART2	0x3FF6_E000	0x3FF6_EFFF	4 KB
	PWM2	0x3FF6_F000	0x3FF6_FFFF	4 KB
	PWM3	0x3FF7_0000	0x3FF7_0FFF	4 KB
	RNG	0x3FF7_5000	0x3FF7_5FFF	4 KB

### 4.1.5 Cache

ESP32 的两个 CPU 各有一组容量为 32 KB、块大小为 32 字节、两路组相连的 cache，用以访问外部存储器。

详细信息请参考 [《ESP32 技术参考手册》> 章节 系统和存储器 > 小节 Cache](#)。

## 4.2 系统时钟

### 4.2.1 CPU 时钟

一旦芯片重启，外置主晶振时钟源会被选为默认的 CPU 时钟。这个外置晶振也会与 PLL 连接产生一个高频时钟（通常为 160 MHz）。

另外，ESP32 内置了一个 8 MHz 的振荡器。应用程序可以在外置主晶振、PLL 时钟和内置 8 MHz 时钟中选择一个作为时钟源。根据不同的应用程序，被选择的时钟源直接或在分频之后驱动 CPU 时钟。

### 4.2.2 RTC 时钟

RTC 时钟有 5 种可能的时钟源：

- 外置低速 (32 kHz) 晶振时钟
- 外置主晶振的 4 分频时钟
- 内置 RC 振荡器（通常为 150 kHz，频率可调节）
- 内置 8 MHz 振荡器时钟
- 内置 31.25 kHz 时钟（由内置 8 MHz 振荡器时钟经 256 分频生成）

当芯片处于正常功耗模式且需要更快速的 CPU 访问时，应用程序可选择外置主晶振的 4 分频时钟或者内置 8 MHz 振荡器时钟。当芯片在低功耗模式下运行时，应用程序可选择外部低速 (32 kHz) 晶振时钟、内置 RC 振荡器时钟或内置 31.25 kHz 时钟。

### 4.2.3 音频 PLL 时钟

音频时钟由超低噪声 fractional-N PLL 生成。

详细信息请参考 [《ESP32 技术参考手册》> 章节 复位和时钟](#)。

## 4.3 RTC 和低功耗管理

### 4.3.1 电源管理单元 (PMU)

ESP32 采用了先进的电源管理技术，可以在不同的功耗模式之间切换。

- 功耗模式

- **Active 模式**: 芯片射频处于工作状态。芯片可以接收、发射和侦听信号。
- **Modem-sleep 模式**: CPU 可运行，时钟可被配置。Wi-Fi/蓝牙基带和射频关闭。
- **Light-sleep 模式**: CPU 暂停运行。RTC 存储器和外设以及 ULP 协处理器运行。任何唤醒事件 (MAC、SDIO 主机、RTC 定时器或外部中断) 都会唤醒芯片。
- **Deep-sleep 模式**: CPU 和大部分外设都会掉电，只有 RTC 存储器和 RTC 外设处于工作状态。Wi-Fi 和蓝牙连接数据存储在 RTC 中。ULP 协处理器可以工作。
- **Hibernation 模式**: 内置的 8 MHz 振荡器和 ULP 协处理器均被禁用。RTC 内存恢复电源被切断。只有 1 个位于低速时钟上的 RTC 时钟定时器和某些 RTC GPIO 在工作。RTC 时钟定时器或 RTC GPIO 可以将芯片从 Hibernation 模式中唤醒。

设备在不同的功耗模式下有不同的电流消耗，详情请见下表。

表 4-2. 不同功耗模式下的功耗

功耗模式	描述			功耗	
Active (射频工作)	Wi-Fi Tx packet			详见表 5-4	
	Wi-Fi/BT Tx packet				
	Wi-Fi/BT Rx 和侦听				
Modem-sleep	CPU 处于工作状态	240 MHz *	双核芯片	30 mA ~ 68 mA	
		单核芯片	N/A		
		160 MHz *	双核芯片	27 mA ~ 44 mA	
		单核芯片	27 mA ~ 34 mA		
		正常速度: 80 MHz	双核芯片	20 mA ~ 31 mA	
			单核芯片	20 mA ~ 25 mA	
Light-sleep	-			0.8 mA	
Deep-sleep	ULP 协处理器处于工作状态			150 $\mu$ A	
	超低功耗传感器监测方式			100 $\mu$ A @1% duty	
	RTC 定时器 + RTC 存储器			10 $\mu$ A	
Hibernation	仅有 RTC 定时器处于工作状态			5 $\mu$ A	
关闭	CHIP_PU 脚拉低，芯片处于关闭状态			1 $\mu$ A	

- \* ESP32 系列芯片中, ESP32-D0WD-V3、ESP32-D0WDR2-V3、ESP32-U4WDH、ESP32-D0WD ([NRND](#))、ESP32-D0WDQ6-V3 ([NRND](#)) 和 ESP32-D0WDQ6 ([NRND](#)) 的 CPU 最大频率为 240 MHz; ESP32-S0WD ([NRND](#)) 的 CPU 最大频率为 160 MHz。
- 在 Wi-Fi 开启的场景中，芯片会在 Active 和 Modem-sleep 模式之间切换，功耗也会在两种模式间变化。
- Modem-sleep 模式下，CPU 频率自动变化，频率取决于 CPU 负载和使用的外设。
- Deep-sleep 模式下，仅 ULP 协处理器处于工作状态时，可以操作 GPIO 及低功耗 I2C。
- 当系统处于超低功耗传感器监测模式时，ULP 协处理器和传感器周期性工作，ADC 以 1% 占空比工作，系统功耗典型值为 100  $\mu$ A。

### 4.3.2 超低功耗协处理器

ULP 处理器和 RTC 存储器在 Deep-sleep 模式下仍保持工作状态。因此，开发者可以将 ULP 协处理器的程序存放在 RTC 慢速存储器中，使其能够在 Deep-sleep 模式下访问外设、内置定时器和内置传感器。在 CPU 需要由外部事件、定时器或者这些事件的组合来唤醒的应用中，可用于保持最低的功耗。

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 超低功耗协处理器 (ULP)。

## 4.4 定时器和看门狗

### 4.4.1 通用定时器

芯片内置 4 个 64-bit 通用定时器，具有 16-bit 分频器和 64-bit 可自动重载的向上/向下计时器。

定时器特性：

- 16-bit 时钟分频器，分频系数为 2 至 65536
- 64-bit 计时器
- 计时器方向可配置：递增或递减
- 软件控制计数暂停和继续
- 定时器超时自动重载
- 软件控制的即时重载
- 电平触发中断和边沿触发中断

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 定时器组 (TIMG)。

### 4.4.2 看门狗定时器

芯片中有 3 个看门狗定时器：2 个定时器模块中各有 1 个（称作主看门狗定时器，即 MWDT），RTC 模块中也有 1 个（称作 RTC 看门狗定时器，即 RWDT）。意外的软件或硬件问题会导致应用程序工作失常，而看门狗定时器可以帮助系统从中恢复。看门狗定时器有 4 个阶段。如果当前阶段超过预定时间，但没有喂狗或关闭看门狗定时器，可能引发以下 3 到 4 种动作中的 1 种。这些动作是：中断、CPU 复位、内核复位和系统复位。其中，只有 RWDT 能够触发系统复位，将复位包括 RTC 在内的整个芯片。每个阶段的超时时间长度均可单独设置。

在 flash 启动期间，RWDT 和第一个 MWDT 会自动开启，以便检测和修复启动问题。

看门狗具有以下特性：

- 4 个阶段，每一阶段都可被单独配置或关闭
- 各阶段时间段可被编程
- 如阶段超时，会采取 3 到 4 种可能动作中的 1 种（中断、CPU 复位、内核复位和系统复位）
- 32-bit 超时计数器
- 写保护，防止 RWDT 和 MWDT 配置被误写
- SPI flash 启动保护；如果在预定时间内，来自 SPI flash 的启动过程没有完成，看门狗会重启整个系统。

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 看门狗定时器 (WDT)。

## 4.5 加密硬件加速器

ESP32 配备硬件加速器，支持一些通用加密算法，比如 AES (FIPS PUB 197)、SHA (FIPS PUB 180-4)、RSA 和 ECC 等，还支持大数乘法、大数模乘等独立运算。硬件加速器支持的 RSA、ECC、大数乘法和大数模乘运算最大长度可达 4096 bit。

硬件加速器极大提高了运算速度，显著减小了软件的复杂度。硬件加速器还支持对 flash 的加密与动态解密，确保 flash 中的代码不被窃取。

## 4.6 射频和 Wi-Fi

芯片射频包含以下主要模块：

- 2.4 GHz 接收器
- 2.4 GHz 发射器
- 偏置 (bias) 和线性稳压器
- Balun 和收发切换器
- 时钟生成器

### 4.6.1 2.4 GHz 接收器

2.4 GHz 接收器将 2.4 GHz 射频信号解调为正交基带信号，并用 2 个高精度、高速的 ADC 将后者转为数字信号。为了适应不同的信道情况，芯片集成了 RF 滤波器、自动增益控制 (AGC)、DC 偏移补偿电路和基带滤波器。

### 4.6.2 2.4 GHz 发射器

2.4 GHz 发射器将正交基带信号调制为 2.4 GHz 射频信号，使用大功率互补金属氧化物半导体 (CMOS) 功率放大器驱动天线。数字校准进一步改善了功率放大器的线性，使得 802.11b 无线传输功率可达 +20.5 dBm，802.11n 无线传输功率可达 +18 dBm。

为了抵消射频接收器的瑕疵，芯片还另增了校准措施，例如：

- 载波泄露消除
- I/Q 相位匹配
- 基带非线性抑制
- 射频非线性抑制
- 天线匹配

这些内置校准措施缩短了产品的测试时间，不再需要测试设备。

### 4.6.3 时钟生成器

时钟生成器为接收器和发射器生成 2.4 GHz 正交时钟信号，所有部件均集成于芯片上，包括电感、变容二极管、环路滤波器、线性稳压器和分频器。

时钟生成器含有内置校准电路和自测电路。运用拥有自主知识产权的校准算法，正交时钟相噪特性在片上经过算法优化处理（该算法拥有自主知识产权），以确保接收器和发射器达到最佳性能。

#### 4.6.4 Wi-Fi 射频和基带

芯片支持 TCP/IP 协议，完全遵循 802.11b/g/n Wi-Fi MAC 协议栈，支持分布式控制功能 (DCF) 下的基本服务集 (BSS) STA 和 SoftAP 操作。支持通过最小化主机交互来优化有效工作时长，以实现功耗管理。

Wi-Fi 射频和基带支持以下特性：

- 802.11b/g/n
- 802.11n MCS0-7 支持 20 MHz 和 40 MHz 带宽
- 802.11n MCS32 (RX)
- 802.11n 0.4  $\mu$ s 保护间隔
- 数据率高达 150 Mbps
- 接收 STBC 2×1
- 发射功率高达 20.5 dBm
- 可调节的发射功率
- 天线分集

芯片支持带有外部射频开关的天线分集与选择。外部射频开关由一个或多个 GPIO 管脚控制，用来选择最合适的天线以减少信道衰减的影响。

#### 4.6.5 Wi-Fi MAC

Wi-Fi MAC 自行支持的底层协议功能如下：

- 4 个虚拟 Wi-Fi 接口
- 同时支持基础结构型网络 (Infrastructure BSS) Station 模式/SoftAP 模式/混杂模式
- RTS 保护，CTS 保护，立即块确认 (Immediate Block ACK)
- 重组 (Defragmentation)
- TX/RX A-MPDU, RX A-MSDU
- TXOP
- 无线多媒体 (WMM)
- CCMP (CBC-MAC, 计数器模式)、TKIP (MIC, RC4)、WAPI (SMS4)、WEP (RC4) 和 CRC
- 自动 Beacon 监测 (硬件 TSF)

### 4.7 蓝牙

芯片集成了蓝牙链路控制器和蓝牙基带，支持基带协议和其他底层链路协议，例如调制/解调、包处理、比特流处理和跳频等。

#### 4.7.1 蓝牙射频和基带

芯片蓝牙射频和基带支持以下特性：

- Class-1、Class-2 和 Class-3 发射输出功率，动态控制范围高达 21 dB

- $\pi/4$  DQPSK 和 8 DPSK 调制
- NZIF 接收器灵敏度高，最小灵敏度达 -94 dBm
- 无需外部 PA 即可支持 Class-1 操作
- 内置 SRAM 支持全速数据传送、混合语音和数据以及完整的微微网 (Piconet) 运行
- 用于前向纠错、包头错误控制、接入码检查、CRC、解调、加密比特流生成、白化和发送脉冲成形的逻辑电路
- ACL、SCO、eSCO 和 AFH
- PCM 接口中的 A-law、 $\mu$ -law 和 CVSD 数字音频编解码
- SBC 音频编解码
- 低功耗应用的电源管理
- 带有 128-bit AES 的 SMP

#### 4.7.2 蓝牙接口

- 提供 UART HCI 接口，速度高达 4 Mbps
- 提供 SDIO/SPI HCI 接口
- 提供 PCM/I2S 音频接口

#### 4.7.3 蓝牙协议栈

芯片的蓝牙协议栈支持蓝牙 v4.2 BR/EDR 和 Bluetooth LE 标准。

#### 4.7.4 蓝牙链路控制器

链路控制器主要可实现 3 种操作：Standby、Connection 和 Sniff。可实现多路连接以及查询、寻呼和安全简易配对 (SSP) 等操作，因此能够组建微微网 (Piconet) 和散射网 (Scatternet)。以下为链路控制器的主要特性：

- 传统蓝牙
  - 设备发现（查询和查询扫描）
  - 建立连接（寻呼和寻呼扫描）
  - 多连接
  - 支持异步数据收发
  - 同步连接 (SCO/eSCO)
  - 主从切换
  - 自适应跳频 (AFH) 和信道选择
  - 加密广播
  - 授权和加密
  - 安全简易配对 (SSP)
  - 多点连接和散射网管理

- Sniff (呼吸) 模式
- 非连接的从模式广播 (收发)
- 增强型功率控制
- Ping
- 低功耗蓝牙
  - 广播
  - 扫描
  - 支持同时广播和扫描
  - 多连接
  - 异步数据收发
  - 自适应跳频和信道选择
  - 连接参数更新
  - 支持扩展的数据包长度
  - 链路层加密
  - LE Ping

## 4.8 数字外设

### 4.8.1 通用输入/输出接口 (GPIO)

ESP32 共有 34 个 GPIO 管脚，通过配置对应的寄存器，可以为这些管脚分配不同的功能，包括如下几类 GPIO：只有数字功能的 GPIO、带模拟功能的 GPIO、带电容触摸功能的 GPIO 等。带模拟功能的 GPIO 和带电容触摸功能的 GPIO 可以被配置为数字 GPIO。

大部分带数字功能的 GPIO 都可以被配置为内部上拉/下拉，或者被设置为高阻。当被配置为输入时，可通过读取寄存器获取输入值。输入管脚也可以被设置为通过边缘触发或电平触发来产生 CPU 中断。大部分数字 IO 管脚都是双向、非反相和三态的，包括带有三态控制的输入和输出缓冲器。这些管脚可以复用作其他功能，例如 SDIO、UART、SPI 等（更多信息请参考附录 [IO\\_MUX](#)）。当芯片低功耗运行时，GPIO 可被设定为保持状态。

详细信息请参考 章节 [4.10 外设管脚分配](#)，附件 [管脚清单](#) 和 [《ESP32 技术参考手册》](#) > 章节 [IO MUX 和 GPIO 交换矩阵 \(GPIO, IO MUX\)](#)。

### 4.8.2 串行外设接口 (SPI)

ESP32 共有 3 组 SPI (SPI、HSPI 和 VSPI) 接口，可以在主机或从机模式，在 1-line 全双工或 1/2/4-line 半双工通信模式下工作。

#### GP-SPI (通用 SPI) 的特性

- 数据传输长度以 byte 为单位可配置
- 支持四线全双工/半双工通信和三线半双工通信
- 主机模式和从机模式时钟可配置

- 时钟极性 (CPOL) 和时钟相位 (CPHA) 可配置
- 时钟可配置

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 SPI 控制器。

### 管脚分配

SPI 接口的管脚通过 IO MUX 与 GPIO6 ~ GPIO11 复用。HSPI 接口通过 IO MUX 与 GPIO2, GPIO4, GPIO12 ~ GPIO15 复用。VSPI 接口通过 IO MUX 与 GPIO5, GPIO18 ~ GPIO19, GPIO21 ~ GPIO23 复用。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.3 通用异步收发器 (UART)

ESP32 芯片中的 UART 用于芯片与外部 UART 设备之间的异步串行数据传输和接收。ESP32 由两个在主系统中的 UART 和一个低功耗 LP UART 组成。

#### 特性

- 可编程收发波特率
- RAM 由 TX FIFO 和 RX FIFO 共用
- 支持输入信号波特率自检功能
- 支持多种数据长度和停止位的长度
- 支持奇偶校验位
- 支持异步通信 (RS232 和 RS485) 和 IrDA
- 支持 DMA 高速数据通信
- 支持 UART 唤醒模式
- 支持软件流控和硬件流控

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 *UART 控制器 (UART)*。

### 管脚分配

UART 的管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.4 I2C 接口

ESP32 有 2 个 I2C 总线接口，根据用户的配置，总线接口可以用作 I2C 主机或从机模式。

#### 特性

- 两个 I2C 控制器：一个在主系统中 (I2C)，一个在低功耗系统中 (LP I2C)
- 支持标准模式 (100 Kbit/s)
- 支持高速模式 (400 Kbit/s)
- 速度最高可达 5 MHz，但受制于 SDA 上拉强度

- 支持 7 位和 10 位寻址以及双地址寻址模式
- 支持关闭 SCL 时钟实现连续数据传输
- 支持可编程数字噪声滤波功能

用户可以配置指令寄存器来控制 I2C 接口，从而实现更多灵活的应用。

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 I2C 控制器 (I2C)。

### 管脚分配

I2C 的管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 4.10 外设管脚分配 和附件管脚清单。

## 4.8.5 I2S 接口

ESP32 芯片中的 I2S 控制器为多媒体应用程序提供了一种灵活的通信接口，特别适用于数字音频应用。

### 特性

- 支持主机模式和从机模式
- 支持全双工和半双工通信
- 支持多种音频标准
- 可配置高精度输出时钟
- 支持 PDM 信号输入输出
- 收发数据模式可配置

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 I2S 控制器 (I2S)。

### 管脚分配

I2S 的管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 4.10 外设管脚分配 和附件管脚清单。

## 4.8.6 红外遥控

红外遥控 (RMT) 用于控制发送和接收红外遥控信号。

### 特性

- 八个通道，用于发送和接收红外遥控信号
- 每个通道具有独立的发送和接收功能
- 每个发送通道都有时钟分频器计数器、状态机和发射器
- 支持多种红外协议

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 红外遥控 (RMT)。

### 管脚分配

RMT 管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.7 脉冲计数控制器

脉冲计数控制器 (PCNT) 旨在通过跟踪输入脉冲信号的上升沿和下降沿来计数。

#### 特性

- 八个独立的脉冲计数器，每个计数器有两个通道
- 递增、递减或停止计数模式
- 输入脉冲信号和控制信号的毛刺滤波
- 选择在输入脉冲信号的上升沿或下降沿进行计数

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 脉冲计数控制器 (PCNT)。

### 管脚分配

脉冲计数控制器的管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.8 LED PWM 控制器

LED PWM 控制器 (LEDC) 用于生成用于 LED 控制的 PWM 信号。

#### 特性

- 十六个独立的 PWM 生成器
- 最大 PWM 占空比精度为 20 位
- 八个独立的定时器，具有 20 位计数器、可配置的时钟小数分频器和计数器溢出值
- 可调节 PWM 信号输出的相位
- PWM 占空比微调
- 占空比自动渐变

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 LED PWM 控制器 (LEDC)。

### 管脚分配

LED PWM 管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.9 电机控制脉宽调制器 (MCPWM)

PWM 控制器可以用于驱动数字马达和智能灯。该控制器包含 PWM 定时器、PWM 操作器和 1 个专用的捕捉子模组。定时器可以同步定时，也可以独立运行。每个 PWM 执行器为 1 个 PWM 通道生成波形。专用的捕捉子模组可以精确捕捉外部定时事件。

#### 特性

- 三个 PWM 定时器，用于精确计时和频率控制
  - 每个 PWM 定时器都有一个专用的 8 位时钟预分频器
  - PWM 定时器中的 16 位计数器的工作模式包括：递增计数模式，递减计数模式，递增递减循环计数模式
  - 硬件同步可以触发 PWM 定时器重载，重载值位于相位寄存器中；同时触发预分频的重启，从而同步定时器的时钟，硬件同步源可选择
- 三个 PWM 操作器，用于生成波形对
  - 六个 PWM 输出，可在几种拓扑结构中运行
  - 死区时间在上升沿和下降沿可配置，并可分别设置
  - 通过高频载波信号调制 PWM 输出，在使用变压器隔离栅极驱动器时可发挥巨大作用
- 故障检测模块
  - 出现故障时，可选择在逐周期模式或一次性模式下处理
  - 故障条件可强制 PWM 输出高或低电平
- 捕获模块，用于基于硬件的信号处理
  - 旋转电机的速度测量
  - 位置传感器脉冲之间的间隔时间测量
  - 脉冲序列信号的周期和占空比测量
  - 从电流/电压传感器的占空比编码信号导出的解码电流或电压振幅
  - 3 个独立的捕获通道，各具备一个 32 位的时间戳寄存器
  - 输入捕获信号可以预分频，边沿极性可选
  - 捕获定时器可以与 PWM 定时器或外部信号同步

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 电机控制脉宽调制器 (PWM)。

#### 管脚分配

MCPWM 管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 4.10 外设管脚分配 和附件管脚清单。

### 4.8.10 SD/SDIO/MMC 主机控制器

ESP32 集成一个 SD/SDIO/MMC 主机控制器。

## 特性

- 支持两个外部卡
- 支持 SD 存储卡 3.0 和 3.01 标准
- 支持 SDIO 3.0 版本
- 支持 CE-ATA 1.1 版本
- 支持多媒体卡 (MMC 4.41 版本、eMMC 4.5 版本和 4.51 版本)

控制器实现了高达 80 MHz 的时钟输出，并且支持 3 种数据总线模式：1 bit、4 bit 和 8 bit。在 4-bit 数据总线模式中，可以支持 2 个 SD/SDIO/MMC4.41 卡，还支持 1 个以 1.8 V 电压工作的 SD 卡。

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 SD/MMC 主机控制器。

## 管脚分配

SD/SDIO/MMC 主机控制器管脚通过 IO MUX 与 GPIO2, GPIO6 ~ GPIO15 复用。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.11 SDIO/SPI 从机控制器

ESP32 芯片中的 SDIO 2.0 从机控制器提供了对安全数字输入/输出 (SDIO) 设备接口的硬件支持，允许 SDIO 主机通过 SDIO 总线协议访问 ESP32。

## 特性

- 时钟范围为 0 至 50 MHz，支持 SPI、1-bit SDIO 和 4-bit SDIO 的传输模式
- 采样和驱动的时钟边沿可配置
- 主机可直接访问的专用寄存器
- 可中断主机，启动数据传输
- 支持自动填充 SDIO 总线上的发送数据，同样支持自动丢弃 SDIO 总线上的填充数据
- 字节块大小可达 512 字节
- 主机与从机间有中断向量可以相互中断对方
- 用于数据传输的 DMA

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 SDIO 从机控制器。

## 管脚分配

SDIO/SPI 从机控制器的管脚通过 IO MUX 与 GPIO2, GPIO6 ~ GPIO15 复用。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.12 双线汽车接口

双线汽车接口 (TWAI<sup>®</sup>) 是一种为车载应用设计的多主机、多播通信协议。TWAI 控制器用于芯片使用该协议的通信。

## 特性

- 兼容 ISO 11898-1 协议 (CAN 规范 2.0)
- 支持标准格式 (11-bit 标识符) 和扩展格式 (29-bit 标识符)
- 支持速率:
  - 芯片版本 v0.0/v1.0/v1.0 支持 25 Kbit/s ~ 1 Mbit/s 位速率
  - 芯片版本 v3.0/v3.1 支持 12.5 Kbit/s ~ 1 Mbit/s 位速率
- 支持多种操作模式：正常模式、只听模式和自测模式
- 64 字节接收 FIFO
- 特殊发送：单次发送和自发自收
- 接收滤波器（支持单滤波器和双滤波器模式）
- 错误检测与处理：错误计数、错误报警限制可配置、错误代码捕捉和仲裁丢失捕捉

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 双线汽车接口 (TWAI)。

## 管脚分配

TWAI 管脚可以为任意 GPIO，通过 GPIO 交换矩阵配置。

更多关于管脚分配的信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.8.13 以太网 MAC 接口

ESP32 为以太网通信提供了一个符合 IEEE-802.3-2008 标准的媒体访问控制器 (MAC) 接口。ESP32 需要一个外部物理层接口芯片 (PHY) 来连接实体 LAN 总线 (双绞线、光纤等)。物理层接口芯片通过 17 个 MII 信号或 9 个 RMII 信号与 ESP32 连接。

## 特性

- 10 Mbps 和 100 Mbps 的速率
- 专用的 DMA 控制器实现以太网 MAC 接口与专用 SRAM 之间的高速传输
- 带标记的 MAC 帧 (支持 VLAN)
- 半双工 (CSMA/CD) 和全双工操作
- MAC 控制子层 (控制帧)
- 32-bit CRC 自动生成和消除
- 用于单播和组播地址 (广播和组地址) 的多种地址过滤模式
- 记录每个收发帧的 32-bit 状态码
- 内部 FIFO 用于缓存发射和接收帧。发送 FIFO 和接收 FIFO 均为 512 字 (32-bit)
- 符合 IEEE 1588 2008 (PTP V2) 标准的硬件 PTP (精确时间协议)
- 25 MHz/50 MHz 的时钟输出

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 以太网 (MAC)。

## 管脚分配

关于以太网 MAC 接口管脚分配的信息，请参考 章节 4.10 外设管脚分配 和附件管脚清单。

## 4.9 模拟外设

### 4.9.1 模/数转换器 (ADC)

ESP32 集成了 2 个 12 位 SAR ADC，共支持 18 个模拟通道输入。为了实现更低功耗，ESP32 的 ULP 协处理器也可以在睡眠方式下测量电压，此时，可通过设置阈值或其他触发方式唤醒 CPU。

表 4-3 描述了 ADC 特性：

表 4-3. ADC 特性

参数	描述	最小值	最大值	单位
DNL (差分非线性)	RTC 控制器；ADC 外接 100 nF 电容；	-7	7	LSB
	输入为 DC 信号；常温 25 °C；Wi-Fi&BT 关闭	-12	12	LSB
采样速度	RTC 控制器	-	200	kspS
	DIG 控制器	-	2	MspS

说明：

- Atten = 3 时，测量值大于 3000（电压值约为 2450 mV）之后，精度会比上表所述低。
- 使用过滤器多次采样或计算平均值可以获得更好的 DNL 结果。
- VDD3P3\_RTC 电源域的 GPIO 管脚输入电压范围请严格遵守表 4-3 直流电气特性规范，否则会引入 ADC 测量误差，且影响芯片正常工作。

默认情况下，芯片之间会有  $\pm 6\%$  的测量差异。ESP-IDF 提供了对 ADC1 的多种[校准方法](#)。使用 eFuse Vref 参考值校准后的结果如表 4-4 所示。用户如需要更高的精度可选用其他方法自行校准。

表 4-4. ADC 校准结果

参数	描述	最小值	最大值	单位
总误差	Atten = 0, 有效测量范围 100 ~ 950 mV	-23	23	mV
	Atten = 1, 有效测量范围 100 ~ 1250 mV	-30	30	mV
	Atten = 2, 有效测量范围 150 ~ 1750 mV	-40	40	mV
	Atten = 3, 有效测量范围 150 ~ 2450 mV	-60	60	mV

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 片上传感器与模拟信号处理。

## 管脚分配

通过适当的设置，最多可配置 18 个管脚的 ADC，用于电压模数转换。关于管脚分配的详细信息，请参考 章节 4.10 外设管脚分配 和附件管脚清单。

### 4.9.2 数/模转换器 (DAC)

ESP32 有 2 个 8-bit DAC 通道，将 2 路数字信号分别转化为 2 个模拟电压信号输出，两个通道可以独立地工作。DAC 电路由内置电阻串和 1 个缓冲器组成。这 2 个 DAC 可以作为参考电压使用。

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 片上传感器与模拟信号处理。

### 管脚分配

可配置 GPIO 25 和 GPIO 26 管脚用于数模转换。关于管脚分配的详细信息，请参考 章节 [4.10 外设管脚分配](#) 和附件[管脚清单](#)。

### 4.9.3 触摸传感器

ESP32 提供了多达 10 个电容式传感 GPIO，能够探测由手指或其他物品直接接触或接近而产生的电容差异。这种设计的低噪声特性和电路的高灵敏度支持使用相对较小的触摸板。也可以使用触摸板阵列以探测更大区域或更多点。

### 管脚分配

表 4-5 列出了 10 个电容式传感 GPIO。

表 4-5. ESP32 上的电容式传感 GPIO

电容式传感信号名称	管脚名称
T0	GPIO4
T1	GPIO0
T2	GPIO2
T3	MTDO
T4	MTCK
T5	MTDI
T6	MTMS
T7	GPIO27
T8	32K_XN
T9	32K_XP

详细信息请参考 [《ESP32 技术参考手册》](#) > 章节 片上传感器与模拟信号处理。

**说明:**

ESP32 触摸传感器目前尚无法通过射频抗扰度测试系统 (CS) 认证，应用场景有所限制。

## 4.10 外设管脚分配

表 4-6. 外设管脚分配

接口	信号	管脚	功能
ADC	ADC1_CH0	SENSOR_VP	2 个 12-bit SAR ADC
	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	
ADC	ADC2_CH0	GPIO4	2 个 12-bit SAR ADC
	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	2 个 8-bit DAC
	DAC_2	GPIO26	
触摸传感器	TOUCH0	GPIO4	电容式触摸传感器
	TOUCH1	GPIO0	
	TOUCH2	GPIO2	
	TOUCH3	MTDO	
	TOUCH4	MTCK	
	TOUCH5	MTDI	
	TOUCH6	MTMS	
	TOUCH7	GPIO27	
	TOUCH8	32K_XN	
	TOUCH9	32K_XP	
JTAG	MTDI	MTDI	软件调试 JTAG
	MTCK	MTCK	
	MTMS	MTMS	
	MTDO	MTDO	
SD/SDIO/MMC 主机控制器	HS2_CLK	MTMS	支持 V3.01 标准 SD 内存卡
	HS2_CMD	MTDO	
	HS2_DATA0	GPIO2	
	HS2_DATA1	GPIO4	
	HS2_DATA2	MTDI	
	HS2_DATA3	MTCK	

接口	信号	管脚	功能
电机 PWM	PWMO_OUT0~2	任意 GPIO 管脚	16-bit 计时器的 3 路通道可产生 PWM 波形，每路包括 1 对输出信号、3 个故障检测信号、3 个事件捕捉信号，及 3 个同步信号。
	PWM1_OUT_IN0~2		
	PWMO_FLT_IN0~2		
	PWM1_FLT_IN0~2		
	PWMO_CAP_IN0~2		
	PWM1_CAP_IN0~2		
	PWMO_SYNC_IN0~2		
	PWM1_SYNC_IN0~2		
SDIO/SPI 从机控制器	SD_CLK	MTMS	SDIO 接口，符合 SDIO V2.0 卡行业标准。
	SD_CMD	MTDO	
	SD_DATA0	GPIO2	
	SD_DATA1	GPIO4	
	SD_DATA2	MTDI	
	SD_DATA3	MTCK	
UART	U0RXD_in	任意 GPIO 管脚	3 路 UART 设备，支持硬件流控制和 DMA
	UOCTS_in		
	UODSR_in		
	UOTXD_out		
	UORTS_out		
	UODTR_out		
	U1RXD_in		
	U1CTS_in		
	U1TXD_out		
	U1RTS_out		
	U2RXD_in		
	U2CTS_in		
	U2TXD_out		
	U2RTS_out		
I2C	I2CEXTO_SCL_in	任意 GPIO 管脚	2 路 I2C 设备，支持主机或从机模式
	I2CEXTO_SDA_in		
	I2CEXT1_SCL_in		
	I2CEXT1_SDA_in		
	I2CEXTO_SCL_out		
	I2CEXTO_SDA_out		
	I2CEXT1_SCL_out		
	I2CEXT1_SDA_out		
LED PWM	ledc_hs_sig_out0~7	任意 GPIO 管脚	16 路独立通道；采用 @80 MHz 时钟/RTC 时钟。占空比精确度：16-bit。
	ledc_ls_sig_out0~7		

接口	信号	管脚	功能
I2S	I2SOI_DATA_in0~15	任意 GPIO 管脚	用于串行立体声数据的输入输出，并行 LCD 数据的输出，camera 并行数据的输入。  注：I2SO_CLK 和 I2S1_CLK 仅可通过 IO MUX 映射至 GPIO0、UORXD (GPIO3) 或 UOTXD (GPIO1)，选择 GPIO 功能 CLK_OUT1、CLK_OUT2 或 CLK_OUT3。更多信息，请参考 <a href="#">《ESP32 技术参考手册》&gt; 章节 IO_MUX 和 GPIO Matrix &gt; IO MUX Pad 列表。</a>
	I2SOO_BCK_in		
	I2SOO_WS_in		
	I2SOI_BCK_in		
	I2SOI_WS_in		
	I2SOI_H_SYNC		
	I2SOI_V_SYNC		
	I2SOI_H_ENABLE		
	I2SOO_BCK_out		
	I2SOO_WS_out		
	I2SOI_BCK_out		
	I2SOI_WS_out		
	I2SOO_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		
	I2SO_CLK	GPIO0、UORXD、或 UOTXD	8 路 IR 收发器，支持不同波形标准。
	I2S1_CLK		
红外遥控器	RMT_SIG_IN0~7	任意 GPIO 管脚	8 路 IR 收发器，支持不同波形标准。
	RMT_SIG_OUT0~7		
通用 SPI	HSPIQ_in/_out	任意 GPIO 管脚	Standard SPI 接口包括时钟、片选、MOSI 和 MISO。这些接口可连接至 LCD 等外设设备，支持以下功能： <ul style="list-style-type: none"><li>• 主机和从机模式；</li><li>• 4 种模式的 SPI 传输格式；</li><li>• 可配置 SPI 频率；</li><li>• 高达 64 字节 FIFO 和 DMA。</li></ul>
	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		

接口	信号	管脚	功能
并行 QSPI	SPIHD	SD_DATA_2	支持 Standard SPI、Dual SPI 和 Quad SPI，可以连接外部 Flash 和 SRAM。
	SPIWP	SD_DATA_3	
	SPICSO	SD_CMD	
	SPICLK	SD_CLK	
	SPIQ	SD_DATA_0	
	SPID	SD_DATA_1	
	HSPICLK	MTMS	
	HSPICSO	MTDO	
	HSPIQ	MTDI	
	HSPID	MTCK	
	HSPIHD	GPIO4	
	HSPIWP	GPIO2	
	VSPICLK	GPIO18	
	VSPICSO	GPIO5	
	VSPIQ	GPIO19	
EMAC	VSPIID	GPIO23	以太网 MAC、MII/RMII 接口
	VSPIHD	GPIO21	
	VSPIWP	GPIO22	
	EMAC_TX_CLK	GPIO0	
	EMAC_RX_CLK	GPIO5	
	EMAC_TX_EN	GPIO21	
	EMAC_TXDO	GPIO19	
	EMAC_TXD1	GPIO22	
	EMAC_TXD2	MTMS	
	EMAC_TXD3	MTDI	
	EMAC_RX_ER	MTCK	
	EMAC_RX_DV	GPIO27	
	EMAC_RXDO	GPIO25	
	EMAC_RXD1	GPIO26	
	EMAC_RXD2	UOTXD	
	EMAC_RXD3	MTDO	
	EMAC_CLK_OUT	GPIO16	
	EMAC_CLK_OUT_180	GPIO17	
	EMAC_TX_ER	GPIO4	
	EMAC_MDC_out	任意 GPIO 管脚	
	EMAC_MDI_in	任意 GPIO 管脚	
	EMAC_MDO_out	任意 GPIO 管脚	
	EMAC_CRS_out	任意 GPIO 管脚	
	EMAC_COL_out	任意 GPIO 管脚	

接口	信号	管脚	功能
脉冲计数器	pcnt_sig_ch0_in0	任意 GPIO 管脚	脉冲计数器通过 7 种模式捕捉脉冲并对脉冲边沿计数。
	pcnt_sig_ch1_in0		
	pcnt_ctrl_ch0_in0		
	pcnt_ctrl_ch1_in0		
	pcnt_sig_ch0_in1		
	pcnt_sig_ch1_in1		
	pcnt_ctrl_ch0_in1		
	pcnt_ctrl_ch1_in1		
	pcnt_sig_ch0_in2		
	pcnt_sig_ch1_in2		
	pcnt_ctrl_ch0_in2		
	pcnt_ctrl_ch1_in2		
	pcnt_sig_ch0_in3		
	pcnt_sig_ch1_in3		
	pcnt_ctrl_ch0_in3		
	pcnt_ctrl_ch1_in3		
	pcnt_sig_ch0_in4		
	pcnt_sig_ch1_in4		
	pcnt_ctrl_ch0_in4		
	pcnt_ctrl_ch1_in4		
	pcnt_sig_ch0_in5		
	pcnt_sig_ch1_in5		
	pcnt_ctrl_ch0_in5		
	pcnt_ctrl_ch1_in5		
	pcnt_sig_ch0_in6		
	pcnt_sig_ch1_in6		
	pcnt_ctrl_ch0_in6		
	pcnt_ctrl_ch1_in6		
	pcnt_sig_ch0_in7		
	pcnt_sig_ch1_in7		
	pcnt_ctrl_ch0_in7		
	pcnt_ctrl_ch1_in7		
TWAI	twai_rx	任意 GPIO 管脚	兼容 ISO11898-1 协议 (CAN 规范 2.0)
	twai_tx		
	twai_bus_off_on		
	twai_clkout		

## 5 电气特性

### 5.1 绝对最大额定值

超出表 5-1 绝对最大额定值 的绝对最大额定值可能导致器件永久性损坏。这只是强调的额定值，不涉及器件在这些或其它条件下超出章节 5.2 建议工作条件 技术规格指标的功能性操作。长时间暴露在绝对最大额定条件下可能会影响设备的可靠性。

表 5-1. 绝对最大额定值

参数	说明	最小值	最大值	单位
VDDA, VDD3P3, VDD3P3_RTC, VDD3P3_CPU, VDD_SDIO	允许输入电压	-0.3	3.6	V
$I_{output}^1$	IO 输出总电流	—	1200	mA
$T_{STORE}$	存储温度	-40	150	°C

<sup>1</sup> 在 25 °C 的环境温度下连续 24 小时保持所有 IO 管脚拉高并接地，设备工作完全正常。

### 5.2 建议工作条件

表 5-2. 建议工作条件

参数	说明	最小值	典型值	最大值	单位
VDDA, VDD3P3, VDD3P3_RTC <sup>note 1</sup> , VDD_SDIO (3.3 V mode) <sup>note 2</sup>	建议输入电压	2.3/3.0 <sup>note 3</sup>	3.3	3.6	V
VDD3P3_CPU	建议输入电压	1.8	3.3	3.6	V
$I_{VDD}$	输入总电流	0.5	—	—	A
$T$ <sup>note 4</sup>	工作温度	-40	—	125	°C

- 写 eFuse 时，VDD3P3\_RTC 应至少 3.3 V。
- VDD\_SDIO 为 IO 的供电电源，通常也作为外接设备的供电电源。具体信息可参考附录 [IO\\_MUX](#)。
  - VDD\_SDIO 可由 ESP32 的 VDD3P3\_RTC 生成：
    - 当 VDD\_SDIO 处于 3.3 V 模式时，由 VDD3P3\_RTC 通过约  $6\ \Omega$  电阻直接供电。因此，VDD\_SDIO 相对 VDD3P3\_RTC 会有一定电压降。
    - 当 VDD\_SDIO 处于 1.8 V 模式时，由 ESP32 内部的 LDO 产生。LDO 能提供的最大电流为 40 mA，输出电压范围为 1.65 V ~ 2.0 V。
  - VDD\_SDIO 也可由外部电源供电。
  - 更多信息请参考章节 [2.5.2 电源管理](#)。
- 带有封装内 3.3 V flash/PSRAM 的芯片，此处电压最小值为 3.0 V；
  - 无封装内 flash/PSRAM 的芯片，此处电压最小值为 2.3 V；
  - 更多信息请查看表 [1 ESP32 系列型号对比](#)。
- 由于 ESP32-U4WDH 的内置 flash 工作温度为 -40 °C ~ 105 °C，所以 ESP32-U4WDH 芯片的整体工作温度为 -40 °C ~ 105 °C。
  - 由于 ESP32-DOWDR2-V3 的内置 PSRAM 工作温度为 -40 °C ~ 85 °C，所以 ESP32-DOWDR2-V3 芯片的整体工作温度为 -40 °C ~ 85 °C。
  - ESP32 系列中其他型号的芯片无内置 flash 或 PSRAM，工作温度为 -40 °C ~ 125 °C。

### 5.3 直流电气特性 (3.3 V, 25 °C)

表 5-3. 直流电气特性 (3.3 V, 25 °C)

参数	说明	最小值	典型值	最大值	单位
$C_{IN}$	管脚电容	—	2	—	pF
$V_{IH}$	高电平输入电压	$0.75 \times VDD^1$	—	$VDD^1 + 0.3$	V
$V_{IL}$	低电平输入电压	—0.3	—	$0.25 \times VDD^1$	V
$I_{IH}$	高电平输入电流	—	—	50	nA
$I_{IL}$	低电平输入电流	—	—	50	nA
$V_{OH}$	高电平输出电压	$0.8 \times VDD^1$	—	—	V
$V_{OL}$	低电平输出电压	—	—	$0.1 \times VDD^1$	V
$I_{OH}$	高电平拉电流 ( $VDD^1 = 3.3$ V, $V_{OH} \geq 2.64$ V, 管脚输出强度设为 最大值)	VDD3P3_CPU 电源域 <sup>1, 2</sup>	—	40	mA
		VDD3P3_RTC 电源域 <sup>1, 2</sup>	—	40	mA
		VDD_SDIO 电源域 <sup>1, 3</sup>	—	20	mA
$I_{OL}$	低电平灌电流 ( $VDD^1 = 3.3$ V, $V_{OL} = 0.495$ V, 管脚输出强度设为最大值)	—	28	—	mA
$R_{PU}$	上拉电阻	—	45	—	kΩ
$R_{PD}$	下拉电阻	—	45	—	kΩ
$V_{IL\_nRST}$	CHIP_PU 关闭芯片的低电平输入电压	—	—	0.6	V

1. VDD 是 I/O 的供电电源，具体请参考附录 IO\_MUX。
2. VDD3P3\_CPU 和 VDD3P3\_RTC 电源域的单个管脚的拉电流随管脚数量增加而减小，从约 40 mA 减小到约 29 mA。
3. VDD\_SDIO 电源域的单个管脚的拉电流随管脚数量增加而减小，从约 30 减小到约 10 mA。

### 5.4 Active 模式下的 RF 功耗

下列功耗数据是基于 3.3 V 电源、25 °C 环境温度，在 RF 接口处完成的测试结果。所有发射数据均基于 50% 的占空比测得。

表 5-4. 射频功耗参数

工作模式	最小值	典型值	最大值	单位
发送 802.11b, DSSS 1 Mbps, POUT = +19.5 dBm	—	240	—	mA
发送 802.11g, OFDM 54 Mbps, POUT = +16 dBm	—	190	—	mA
发送 802.11n, OFDM MCS7, POUT = +14 dBm	—	180	—	mA
接收 802.11b/g/n	—	95 ~ 100	—	mA
发送 BT/BLE, POUT = 0 dBm	—	130	—	mA
接收 BT/BLE	—	95 ~ 100	—	mA

## 5.5 可靠性

表 5-5. 可靠性

测试项目	测试条件	测试标准
HTOL (高温工作寿命)	125 °C, 1000 小时	JESD22-A108
ESD (静电放电敏感度)	HBM (人体放电模式) <sup>1</sup> ± 2000 V	JS-001
	CDM (充电器件模式) <sup>2</sup> ± 500 V	JS-002
闩锁测试 (Latch-up)	过电流 ± 200 mA	JESD78
	过电压 $1.5 \times VDD_{max}$	
预处理测试	烘烤: 125 °C, 24 小时	J-STD-020
	浸泡: 三级 (30 °C, 60% RH, 192 小时)	JESD47
	回流焊: 260 + 0 °C, 20 秒, 三次	JESD22-A113
TCT (温度循环测试)	-65 °C / 150 °C, 500 次循环	JESD22-A104
高压蒸煮试验	121 °C, 100% RH, 96 小时	JESD22-A102
uHAST(无偏压高加速温湿度应力试验)	130 °C, 85% RH, 96 小时	JESD22-A118
HTSL (高温贮存寿命)	150 °C, 1000 小时	JESD22-A103

1. JEDEC 文档 JEP155 规定: 500 V HBM 能够在标准 ESD 控制流程下安全生产。

2. JEDEC 文档 JEP157 规定: 250 V CDM 能够在标准 ESD 控制流程下安全生产。

## 5.6 Wi-Fi 射频

表 5-6. Wi-Fi 射频特性

参数	说明	最小值	典型值	最大值	单位
工作信道中心频率范围 <sup>1</sup>	—	2412	—	2484	MHz
输出阻抗 <sup>2</sup>	—	-	见说明 2	—	Ω
输出功率 <sup>3</sup>	11n, MCS7	12	13	14	dBm
	11b 模式	18.5	19.5	20.5	dBm
灵敏度	11b, 1 Mbps	—	-98	—	dBm
	11b, 11 Mbps	—	-88	—	dBm
	11g, 6 Mbps	—	-93	—	dBm
	11g, 54 Mbps	—	-75	—	dBm
	11n, HT20, MCS0	—	-93	—	dBm
	11n, HT20, MCS7	—	-73	—	dBm
	11n, HT40, MCS0	—	-90	—	dBm
	11n, HT40, MCS7	—	-70	—	dBm
邻道抑制	11g, 6 Mbps	—	27	—	dB
	11g, 54 Mbps	—	13	—	dB
	11n, HT20, MCS0	—	27	—	dB
	11n, HT20, MCS7	—	12	—	dB

1. 工作信道中心频率范围应符合国家或地区的规范标准。软件可以配置工作信道中心频率范围。

2. ESP32 Wi-Fi 射频的输出阻抗典型值与封装尺寸有关。封装为 QFN 6×6 的 ESP32 芯片的输出阻抗为  $30+j10 \Omega$ ；封装为 QFN 5×5 的 ESP32 芯片的输出阻抗为  $35+j10 \Omega$ 。
3. 根据产品或认证的要求，用户可以配置目标功率。

## 5.7 经典蓝牙射频

### 5.7.1 接收器 - 基础数据率 (BR)

表 5-7. 接收器特性 - 基础数据率 (BR)

参数	说明	最小值	典型值	最大值	单位
灵敏度 @0.1% BER	—	-90	-89	-88	dBm
最大接收信号 @0.1% BER	—	0	—	—	dBm
共信道抑制比 C/I	—	—	+7	—	dB
邻道选择性抑制比 C/I	$F = F_0 + 1 \text{ MHz}$	—	—	-6	dB
	$F = F_0 - 1 \text{ MHz}$	—	—	-6	dB
	$F = F_0 + 2 \text{ MHz}$	—	—	-25	dB
	$F = F_0 - 2 \text{ MHz}$	—	—	-33	dB
	$F = F_0 + 3 \text{ MHz}$	—	—	-25	dB
	$F = F_0 - 3 \text{ MHz}$	—	—	-45	dB
带外阻塞	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
互调	—	-36	—	—	dBm

### 5.7.2 发射器 - 基础数据率 (BR)

表 5-8. 发射器特性 - 基础数据率 (BR)

参数	说明	最小值	典型值	最大值	单位
射频发射功率 (见表下方说明)	—	—	0	—	dBm
增益控制步长	—	—	3	—	dB
射频功率控制范围	—	-12	—	+9	dBm
20 dB 带宽	—	—	0.9	—	MHz
邻道发射功率	$F = F_0 \pm 2 \text{ MHz}$	—	-47	—	dBm
	$F = F_0 \pm 3 \text{ MHz}$	—	-55	—	dBm
	$F = F_0 \pm > 3 \text{ MHz}$	—	-60	—	dBm
$\Delta f_{1\text{avg}}$	—	—	—	155	kHz
$\Delta f_{2\text{max}}$	—	133.7	—	—	kHz
$\Delta f_{2\text{avg}}/\Delta f_{1\text{avg}}$	—	—	0.92	—	—
ICFT	—	—	-7	—	kHz
漂移速率	—	—	0.7	—	kHz/50 $\mu$ s

参数	说明	最小值	典型值	最大值	单位
偏移 (DH1)	—	—	6	—	kHz
偏移 (DH5)	—	—	6	—	kHz

**说明:**

从 0 到 7，共有 8 个功率级别，发射功率范围从 -12 dBm 到 9 dBm。功率电平每增加 1 时，发射功率增加 3 dB。默认情况下使用功率级别 4，相应的发射功率为 0 dBm。

**5.7.3 接收器 - 增强数据率 (EDR)**

表 5-9. 接收器特性 - 增强数据率 (EDR)

参数	说明	最小值	典型值	最大值	单位
$\pi/4$ DQPSK					
灵敏度 @0.01% BER	—	-90	-89	-88	dBm
最大接收信号 @0.01% BER	—	—	0	—	dBm
共信道抑制比 C/I	—	—	11	—	dB
邻道选择性抑制比 C/I	$F = F_0 + 1$ MHz	—	-7	—	dB
	$F = F_0 - 1$ MHz	—	-7	—	dB
	$F = F_0 + 2$ MHz	—	-25	—	dB
	$F = F_0 - 2$ MHz	—	-35	—	dB
	$F = F_0 + 3$ MHz	—	-25	—	dB
	$F = F_0 - 3$ MHz	—	-45	—	dB
8DPSK					
灵敏度 @0.01% BER	—	-84	-83	-82	dBm
最大接收信号 @0.01% BER	—	—	-5	—	dBm
共信道抑制比 C/I	—	—	18	—	dB
邻道抑制比 C/I	$F = F_0 + 1$ MHz	—	2	—	dB
	$F = F_0 - 1$ MHz	—	2	—	dB
	$F = F_0 + 2$ MHz	—	-25	—	dB
	$F = F_0 - 2$ MHz	—	-25	—	dB
	$F = F_0 + 3$ MHz	—	-25	—	dB
	$F = F_0 - 3$ MHz	—	-38	—	dB

**5.7.4 发射器 - 增强数据率 (EDR)**

表 5-10. 发射器特性 - 增强数据率 (EDR)

参数	说明	最小值	典型值	最大值	单位
射频发射功率 (见表 5-8 下方说明)	—	—	0	—	dBm
增益控制步长	—	—	3	—	dB
射频功率控制范围	—	-12	—	+9	dBm
$\pi/4$ DQPSK max w0	—	—	-0.72	—	kHz

参数	说明	最小值	典型值	最大值	单位
$\pi/4$ DQPSK max $w_i$	—	—	-6	—	kHz
$\pi/4$ DQPSK max $ w_i + w_0 $	—	—	-7.42	—	kHz
8DPSK max $w_0$	—	—	0.7	—	kHz
8DPSK max $w_i$	—	—	-9.6	—	kHz
8DPSK max $ w_i + w_0 $	—	—	-10	—	kHz
$\pi/4$ DQPSK 调制精度	RMS DEVM	—	4.28	—	%
	99% DEVM	—	100	—	%
	Peak DEVM	—	13.3	—	%
8 DPSK 调制精度	RMS DEVM	—	5.8	—	%
	99% DEVM	—	100	—	%
	Peak DEVM	—	14	—	%
带内杂散发射	$F = F_0 \pm 1$ MHz	—	-46	—	dBm
	$F = F_0 \pm 2$ MHz	—	-40	—	dBm
	$F = F_0 \pm 3$ MHz	—	-46	—	dBm
	$F = F_0 +/\rightarrow 3$ MHz	—	—	-53	dBm
EDR 差分相位编码	—	—	100	—	%

## 5.8 低功耗蓝牙射频

### 5.8.1 接收器

表 5-11. 低功耗蓝牙接收器特性

参数	说明	最小值	典型值	最大值	单位
灵敏度 @30.8% PER	—	-94	-93	-92	dBm
最大接收信号 @30.8% PER	—	0	—	—	dBm
共信道抑制比 C/I	—	—	+10	—	dB
邻道抑制比 C/I	$F = F_0 + 1$ MHz	—	-5	—	dB
	$F = F_0 - 1$ MHz	—	-5	—	dB
	$F = F_0 + 2$ MHz	—	-25	—	dB
	$F = F_0 - 2$ MHz	—	-35	—	dB
	$F = F_0 + 3$ MHz	—	-25	—	dB
	$F = F_0 - 3$ MHz	—	-45	—	dB
带外阻塞	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
互调	—	-36	—	—	dBm

### 5.8.2 发射器

表 5-12. 低功耗蓝牙发射器特性

参数	说明	最小值	典型值	最大值	单位
射频发射功率 (见表 5-8 下方说明)	—	—	0	—	dBm
增益控制步长	—	—	3	—	dB
射频功率控制范围	—	-12	—	+9	dBm
邻道发射功率	$F = FO \pm 2 \text{ MHz}$	—	-52	—	dBm
	$F = FO \pm 3 \text{ MHz}$	—	-58	—	dBm
	$F = FO \pm > 3 \text{ MHz}$	—	-60	—	dBm
$\Delta f_{1\text{avg}}$	—	—	—	265	kHz
$\Delta f_{2\text{max}}$	—	247	—	—	kHz
$\Delta f_{2\text{avg}}/\Delta f_{1\text{avg}}$	—	—	0.92	—	—
ICFT	—	—	-10	—	kHz
漂移速率	—	—	0.7	—	kHz/50 $\mu\text{s}$
偏移	—	—	2	—	kHz

## 6 封装

- 有关卷带、载盘和产品标签的信息，请参阅 [《乐鑫芯片包装信息》](#)。
- 俯视图中，芯片管脚从 Pin 1 位置开始按逆时针方向编号。关于管脚序号和名称的详细信息，请参考章节 [2.1 管脚布局](#)。

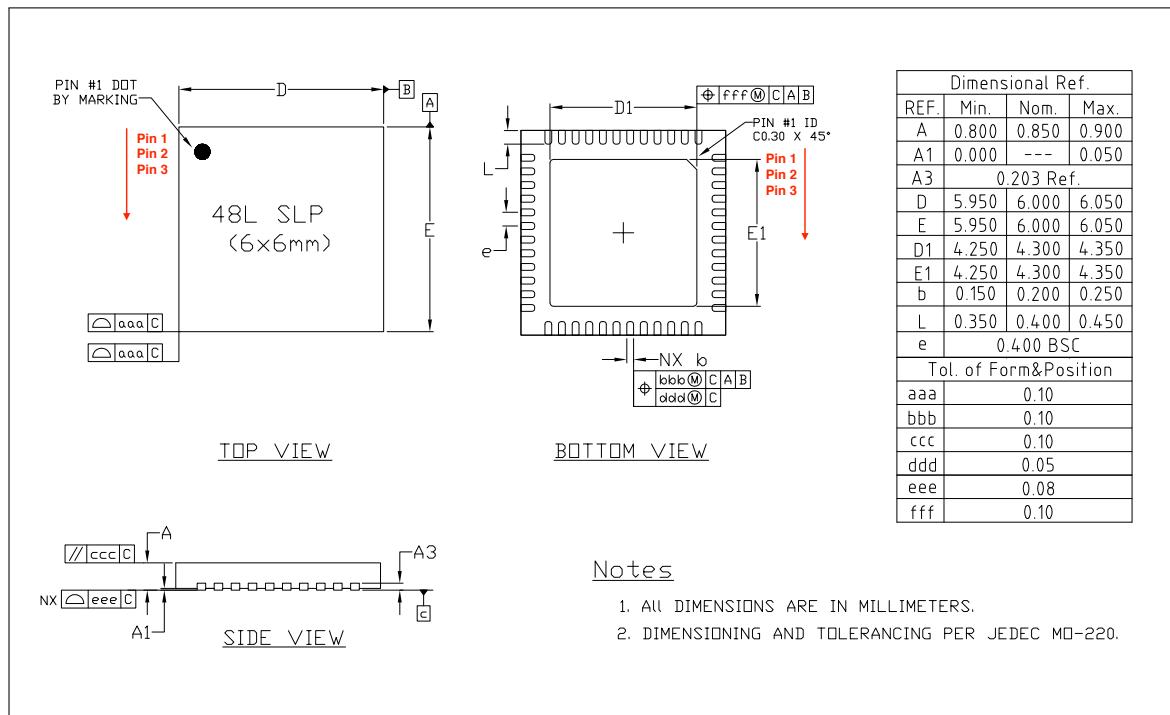


图 6-1. QFN48 (6x6 mm) 封装

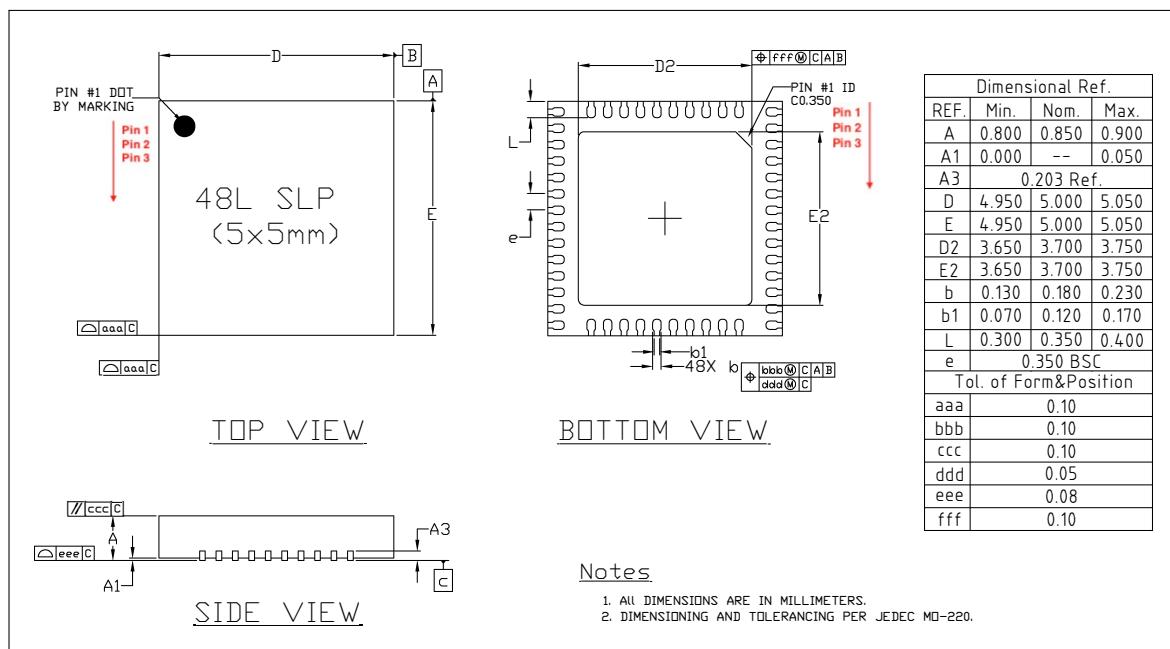


图 6-2. QFN48 (5x5 mm) 封装

## 7 相关文档和资源

### 相关文档

- [《ESP32 技术参考手册》](#) – 提供 ESP32 芯片的存储器和外设的详细使用说明。
- [《ESP32 硬件设计指南》](#) – 提供基于 ESP32 芯片的产品设计规范。
- [《ESP32 勘误表及解决办法》](#) – 提供关于 ESP32 芯片的设计问题的说明和解决方案。
- 证书  
<https://espressif.com/zh-hans/support/documents/certificates>
- ESP32 产品/工艺变更通知 (PCN)  
<https://espressif.com/zh-hans/support/documents/pcns>
- ESP32 公告 – 提供有关安全、bug、兼容性、器件可靠性的信息  
<https://espressif.com/zh-hans/support/documents/advisories>
- 文档更新和订阅通知  
<https://espressif.com/zh-hans/support/download/documents>

### 开发者社区

- [《ESP32 ESP-IDF 编程指南》](#) – ESP-IDF 开发框架的文档中心。
- ESP-IDF 及 GitHub 上的其它开发框架  
<https://github.com/espressif>
- ESP32 论坛 – 工程师对工程师 (E2E) 的社区，您可以在这里提出问题、解决问题、分享知识、探索观点。  
<https://esp32.com/>
- *The ESP Journal* – 分享乐鑫工程师的最佳实践、技术文章和工作随笔。  
<https://blog.espressif.com/>
- SDK 和演示、App、工具、AT 等下载资源  
<https://espressif.com/zh-hans/support/download/sdks-demos>

### 产品

- ESP32 系列芯片 – ESP32 全系列芯片。  
<https://espressif.com/zh-hans/products/socs?id=ESP32>
- ESP32 系列模组 – ESP32 全系列模组。  
<https://espressif.com/zh-hans/products/modules?id=ESP32>
- ESP32 系列开发板 – ESP32 全系列开发板。  
<https://espressif.com/zh-hans/products/devkits?id=ESP32>
- ESP Product Selector (乐鑫产品选型工具) – 通过筛选性能参数、进行产品对比快速定位您所需要的产品。  
<https://products.espressif.com/#/product-selector?language=zh>

### 联系我们

- 商务问题、技术支持、电路原理图 & PCB 设计审阅、购买样品（线上商店）、成为供应商、意见与建议  
<https://espressif.com/zh-hans/contact-us/sales-questions>

# 附录 A - ESP32 管脚清单

## A.1. 管脚清单说明

表 7-1. 管脚清单说明

序号	说明
1	<a href="#">IO_MUX</a> 表格中，黄色单元格表示这几个 GPIO 管脚仅为输入管脚。
2	以下管脚仅为输入管脚，不带输出驱动器或内置上拉 / 下拉电路：SENSOR_VP (GPIO36)、SENSOR_CAPP (GPIO37)、SENSOR_CAPN (GPIO38)、SENSOR_VN (GPIO39)、VDET_1 (GPIO34)、VDET_2 (GPIO35)。
3	所有管脚分为 4 个电源域：VDDA（模拟电源）、VDD3P3_RTC（RTC 电源）、VDD3P3_CPU（数字 IO 和 CPU 电源）、VDD_SDIO（SDIO IO 电源）。VDD_SDIO 是内置 SDIO-LDO 的输出源。SDIO-LDO 的配置电压为 1.8 V 或与 VDD3P3_RTC 相同。Strapping 管脚和 eFuse bit 共同决定 SDIO-LDO 的默认电压。此外，软件还可以配置寄存器来强制改变 SDIO-LDO 电压。详见 <a href="#">IO_MUX</a> 表格中“Power Domain”一栏。
4	VDD3P3_RTC 域中的功能管脚具有模拟功能，包括 32 kHz 晶体振荡器、ADC、DAC 以及电容式触摸传感器。请参考 <a href="#">IO_MUX</a> 表格中“Analog Function 0 ~ 2”各栏。
5	这些 VDD3P3_RTC 管脚支持 RTC Function 功能，能够在 Deep-sleep 模式下使用。例如，RTC-GPIO 可用于 Deep-sleep 模式下的芯片。
6	GPIO 管脚最多支持 <a href="#">IO_MUX</a> 表格中“Function 0 ~ 5”各栏中所列的 6 个数字功能。功能选择寄存器应设置为“N”，其中 N 为功能编号。以下为常用模块的信号前缀定义： <ul style="list-style-type: none"><li>• SD_* 是 SDIO 从机的信号。</li><li>• HS1_* 是 SDIO 主机端口 1 的信号。</li><li>• HS2_* 是 SDIO 主机端口 2 的信号。</li><li>• MT* 是 JTAG 的信号。</li><li>• UO* 是 UART0 模块的信号。</li><li>• U1* 是 UART1 模块的信号。</li><li>• U2* 是 UART2 模块的信号。</li><li>• SPI* 是 SPI01 模块的信号。</li><li>• HSPI* 是 SPI2 模块的信号。</li><li>• VSPI* 是 SPI3 模块的信号。</li></ul>

序号	说明
7	<p>每个数字 Function 栏均对应一列 Type。请参考下列描述来理解每个 Function 所对应 Type 的意义。对于任意功能 Function-<i>N</i> 而言，Type 所指的是：</p> <ul style="list-style-type: none"> <li>• I: 仅为输入。如果选择了 Function-<i>N</i> 以外的其他功能，则该管脚的输入信号仍会连接到 Function-<i>N</i> 的输入信号。</li> <li>• I1: 仅为输入。如果选择了 Function-<i>N</i> 以外的其他功能，则 Function-<i>N</i> 的输入信号恒为 1。</li> <li>• IO: 仅为输入。如果选择 Function-<i>N</i> 以外的其他功能，则 Function-<i>N</i> 的输入信号恒为 0。</li> <li>• O: 仅为输出。</li> <li>• T: 高抗阻。</li> <li>• I/O/T: 该功能信号包含输入、输出和高抗阻组合。</li> <li>• I1/O/T: 该函数信号包含输入、输出和高抗阻组合。如果选择了 Function-<i>N</i> 以外的其他功能，则 Function-<i>N</i> 的输入信号恒为 1。</li> </ul> <p>例如，30 号管脚可作为 HS1_CMD 或 SD_CMD 使用，其中 HS1_CMD 的类型为 I1/O/T。如果 30 号管脚被选作 HS1_CMD，则该管脚的输入和输出均由 SDIO 主机控制。如果 30 号管脚未被选作 HS1_CMD，则对 SDIO 主机的输入信号总为 1。</p>
8	<p>每一个数字输出管脚的驱动强度可以配置。<a href="#">IO_MUX</a> 表格中的“Drive Strength”栏列出了默认值。其中，驱动强度可通过寄存器配置为以下四种：</p> <ul style="list-style-type: none"> <li>• 0: ~5 mA</li> <li>• 1: ~10 mA</li> <li>• 2: ~20 mA</li> <li>• 3: ~40 mA</li> </ul> <p>默认值为 2。</p> <p>内置上拉 (wpu) 和内置下拉 (wpd) 驱动强度为 ~75 <math>\mu</math>A。</p>
9	<a href="#">IO_MUX</a> 表格中“At Reset”栏列出了复位时每个管脚的状态，包括输入使能 (ie=1)、内置上拉 (wpu) 和内置下拉 (wpd)。复位时，所有管脚都处于输出禁用状态。
10	<a href="#">IO_MUX</a> 表格中“After Reset”栏列出了复位后每个管脚的即刻状态，包括输入使能 (ie=1)、内置上拉 (wpu) 和内置下拉 (wpd)。复位后，每个管脚均设定为“Function 0”。输出使能由其数字 Function 0 控制。
11	表格 <a href="#">Ethernet_MAC</a> 用于描述 Ethernet MAC 模块内的信号映射。Ethernet MAC 支持 MII 和 RMII 两种接口，同时支持内部 PLL 时钟和外部时钟源。对于 MII 接口来说，Ethernet MAC 可配 TX_ERR 信号，也可不配 TX_ERR 信号。MDC、MDIO、CRS 和 COL 均为慢速信号，可通过 GPIO 矩阵映射到任何 GPIO 管脚。
12	表格 <a href="#">GPIO_Matrix</a> 用于描述 GPIO 交换矩阵。片上功能模组所发出的信号可映射到任何 GPIO 管脚上。如表格 <a href="#">GPIO_Matrix</a> 中，“信号可经由 IO_MUX 输出”栏说明的一样，有些信号既可通过 IO_MUX，也可通过 GPIO 矩阵映射到管脚上。
13	* 表格 <a href="#">GPIO_Matrix</a> 中，“默认值”一栏表示未分配 GPIO 时，输入信号的默认值。信号的实际值由寄存器位 GPIO_FUNC <i>m</i> _IN_INV_SEL 和寄存器域 GPIO_FUNC <i>m</i> _IN_SEL 共同决定。（ <i>m</i> 的取值范围为 1~255。）

## A.2. GPIO\_Matrix

表 7-2. GPIO\_Matrix

序号	输入信号	默认值 *	信号可经由 IO_MUX 输出	输出信号	输出信号的 输出使能信号
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	SPICSO_in	0	yes	SPICSO_out	SPICSO_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	HSPICSO_in	0	yes	HSPICSO_out	HSPICSO_oe
12	HSPIHD_in	0	yes	HSPIHD_out	HSPIHD_oe
13	HSPIWP_in	0	yes	HSPIWP_out	HSPIWP_oe
14	UORXD_in	0	yes	UOTXD_out	1'd1
15	UOCTS_in	0	yes	UORTS_out	1'd1
16	UODSR_in	0	no	UODTR_out	1'd1
17	U1RXD_in	0	yes	U1TXD_out	1'd1
18	U1CTS_in	0	yes	U1RTS_out	1'd1
23	I2S00_BCK_in	0	no	I2S00_BCK_out	1'd1
24	I2S10_BCK_in	0	no	I2S10_BCK_out	1'd1
25	I2S00_WS_in	0	no	I2S00_WS_out	1'd1
26	I2S10_WS_in	0	no	I2S10_WS_out	1'd1
27	I2S01_BCK_in	0	no	I2S01_BCK_out	1'd1
28	I2S01_WS_in	0	no	I2S01_WS_out	1'd1
29	I2CEXTO_SCL_in	1	no	I2CEXTO_SCL_out	1'd1
30	I2CEXTO_SDA_in	1	no	I2CEXTO_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
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

序号	输入信号	默认值 *	信号可经由 IO_MUX 输出	输出信号	输出信号的 输出使能信号
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	VSPICSO_in	0	yes	VSPICSO_out	VSPICSO_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
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

序号	输入信号	默认值 *	信号可经由 IO_MUX 输出	输出信号	输出信号的 输出使能信号
87	rmt_sig_in4	0	no	rmt_sig_out0	1'd1
88	rmt_sig_in5	0	no	rmt_sig_out1	1'd1
89	rmt_sig_in6	0	no	rmt_sig_out2	1'd1
90	rmt_sig_in7	0	no	rmt_sig_out3	1'd1
91	—	—	—	rmt_sig_out4	1'd1
92	—	—	—	rmt_sig_out6	1'd1
94	twai_rx	1	no	rmt_sig_out7	1'd1
95	I2CEXT1_SCL_in	1	no	I2CEXT1_SCL_out	1'd1
96	I2CEXT1_SDA_in	1	no	I2CEXT1_SDA_out	1'd1
97	host_card_detect_n_1	0	no	host_ccmd_od_pullup_en_n	1'd1
98	host_card_detect_n_2	0	no	host_RST_n_1	1'd1
99	host_card_write_prt_1	0	no	host_RST_n_2	1'd1
100	host_card_write_prt_2	0	no	gpio_sd0_out	1'd1
101	host_card_int_n_1	0	no	gpio_sd1_out	1'd1
102	host_card_int_n_2	0	no	gpio_sd2_out	1'd1
103	pwm1_sync0_in	0	no	gpio_sd3_out	1'd1
104	pwm1_sync1_in	0	no	gpio_sd4_out	1'd1
105	pwm1_sync2_in	0	no	gpio_sd5_out	1'd1
106	pwm1_f0_in	0	no	gpio_sd6_out	1'd1
107	pwm1_f1_in	0	no	gpio_sd7_out	1'd1
108	pwm1_f2_in	0	no	pwm1_out0a	1'd1
109	pwm0_cap0_in	0	no	pwm1_out0b	1'd1
110	pwm0_cap1_in	0	no	pwm1_out1a	1'd1
111	pwm0_cap2_in	0	no	pwm1_out1b	1'd1
112	pwm1_cap0_in	0	no	pwm1_out2a	1'd1
113	pwm1_cap1_in	0	no	pwm1_out2b	1'd1
114	pwm1_cap2_in	0	no	pwm2_out1h	1'd1
115	pwm2_fita	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_fita	1	no	pwm2_out4h	1'd1
121	pwm3_fltb	1	no	pwm2_out4l	1'd1
122	pwm3_cap1_in	0	no	—	1'd1
123	pwm3_cap2_in	0	no	twai_tx	1'd1
124	pwm3_cap3_in	0	no	twai_bus_off_on	1'd1
125	—	—	—	twai_clkout	1'd1
140	I2S0I_DATA_in0	0	no	I2S0O_DATA_out0	1'd1
141	I2S0I_DATA_in1	0	no	I2S0O_DATA_out1	1'd1
142	I2S0I_DATA_in2	0	no	I2S0O_DATA_out2	1'd1
143	I2S0I_DATA_in3	0	no	I2S0O_DATA_out3	1'd1

序号	输入信号	默认值 *	信号可经由 IO_MUX 输出	输出信号	输出信号的 输出使能信号
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	I2S10_DATA_out0	1'd1
167	I2S1I_DATA_in1	0	no	I2S10_DATA_out1	1'd1
168	I2S1I_DATA_in2	0	no	I2S10_DATA_out2	1'd1
169	I2S1I_DATA_in3	0	no	I2S10_DATA_out3	1'd1
170	I2S1I_DATA_in4	0	no	I2S10_DATA_out4	1'd1
171	I2S1I_DATA_in5	0	no	I2S10_DATA_out5	1'd1
172	I2S1I_DATA_in6	0	no	I2S10_DATA_out6	1'd1
173	I2S1I_DATA_in7	0	no	I2S10_DATA_out7	1'd1
174	I2S1I_DATA_in8	0	no	I2S10_DATA_out8	1'd1
175	I2S1I_DATA_in9	0	no	I2S10_DATA_out9	1'd1
176	I2S1I_DATA_in10	0	no	I2S10_DATA_out10	1'd1
177	I2S1I_DATA_in11	0	no	I2S10_DATA_out11	1'd1
178	I2S1I_DATA_in12	0	no	I2S10_DATA_out12	1'd1
179	I2S1I_DATA_in13	0	no	I2S10_DATA_out13	1'd1
180	I2S1I_DATA_in14	0	no	I2S10_DATA_out14	1'd1
181	I2S1I_DATA_in15	0	no	I2S10_DATA_out15	1'd1
182	—	—	—	I2S10_DATA_out16	1'd1
183	—	—	—	I2S10_DATA_out17	1'd1
184	—	—	—	I2S10_DATA_out18	1'd1
185	—	—	—	I2S10_DATA_out19	1'd1

序号	输入信号	默认值 *	信号可经由 IO_MUX 输出	输出信号	输出信号的 输出使能信号
186	—	—	—	I2S10_DATA_out20	1'd1
187	—	—	—	I2S10_DATA_out21	1'd1
188	—	—	—	I2S10_DATA_out22	1'd1
189	—	—	—	I2S10_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
227	—	—	—	sig_in_func227	1'd1
228	—	—	—	sig_in_func228	1'd1

### A.3. Ethernet\_MAC

表 7-3. Ethernet\_MAC

Pin Name	Function6	MII (int_osc)	MII (ext_osc)	RMII (int_osc)	RMII (ext_osc)
GPIO0	EMAC_TX_CLK	TX_CLK (I)	TX_CLK (I)	CLK_OUT(O)	EXT_OSC_CLK(I)
GPIO5	EMAC_RX_CLK	RX_CLK (I)	RX_CLK (I)	—	—
GPIO21	EMAC_TX_EN	TX_EN(O)	TX_EN(O)	TX_EN(O)	TX_EN(O)
GPIO19	EMAC_TXDO	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_RXDO	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)
UOTXD	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

IO\_MUX 管脚清单详见下一页。

IO_MUX																								
Pin No.	Power Supply Pin	Analog Pin	Digital Pin	Power Domain	Analog Function0	Analog Function1	Analog Function2	RTC Function0	RTC Function1	Function0	Type	Function1	Type	Function2	Type	Function3	Type	Function4	Type	Function5	Type	Drive Strength (2'd2: 20 mA)	At Reset	After Reset
1	VDDA			VDDA supply in																				
2		LNA_IN		VDD3P3																				
3	VDD3P3			VDD3P3 supply in																				
4	VDD3P3			VDD3P3 supply in																				
5	SENSOR_VP	VDD3P3_RTC		ADC1_CH0	RTC_GPIO00			GPIO36	I		GPIO36	I										oe=0, ie=0	oe=0, ie=0	
6	SENSOR_CAPP	VDD3P3_RTC		ADC1_CH1	RTC_GPIO01			GPIO37	I		GPIO37	I										oe=0, ie=0	oe=0, ie=0	
7	SENSOR_CAPN	VDD3P3_RTC		ADC1_CH2	RTC_GPIO02			GPIO38	I		GPIO38	I										oe=0, ie=0	oe=0, ie=0	
8	SENSOR_VN	VDD3P3_RTC		ADC1_CH3	RTC_GPIO03			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_GPIO04			GPIO34	I		GPIO34	I										oe=0, ie=0	oe=0, ie=0	
11	VDET_2	VDD3P3_RTC		ADC1_CH7	RTC_GPIO05			GPIO35	I		GPIO35	I										oe=0, ie=0	oe=0, ie=0	
12	32K_XP	VDD3P3_RTC	XTAL_32K_P	ADC1_CH4	TOUCH9	RTC_GPIO09		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_GPIO08		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_GPIO06			GPIO25	I/O/T		GPIO25	I/O/T										2'd2	oe=0, ie=0	oe=0, ie=0
15	GPIO26	VDD3P3_RTC	DAC_2	ADC2_CH9	RTC_GPIO07			GPIO26	I/O/T		GPIO26	I/O/T										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										2'd2	oe=0, ie=0	oe=0, ie=0
17	MTMS	VDD3P3_RTC		ADC2_CH6	TOUCH6	RTC_GPIO16		MTMS	IO	HSPICLK	I/O/T	GPIO14	I/O/T	HS2_CLK	O	SD_CLK	IO	EMAC_RXD2	O	2'd2	oe=0, ie=0	oe=0, ie=1, wpu		
18	MTDI	VDD3P3_RTC		ADC2_CH5	TOUCH5	RTC_GPIO15		MTDI	II	HSPIQ	I/O/T	GPIO12	I/O/T	HS2_DATA2	II/O/T	SD_DATA2	II/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	II	HSPID	I/O/T	GPIO13	I/O/T	HS2_DATA3	II/O/T	SD_DATA3	II/O/T	EMAC_RX_ER	I	2'd2	oe=0, ie=0	oe=0, ie=1, wpd		
21	MTDO	VDD3P3_RTC		ADC2_CH3	TOUCH3	RTC_GPIO13	I2C_SDA	MTDO	O/T	HSPICSO	I/O/T	GPIO15	I/O/T	HS2_CMD	II/O/T	SD_CMD	II/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	HSPWP	I/O/T	GPIO16	I/O/T	HS2_DATA0	II/O/T	SD_DATA0	II/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	HSPIH	I/O/T	GPIO4	I/O/T	HS2_DATA1	II/O/T	SD_DATA1	II/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	II/O/T	U2RXD	II	EMAC_CLK_OUT	O	2'd2	oe=0, ie=0	oe=0, ie=1			
26	VDD_SDIO	VDD_SDIO	VDD_SDIO supply out/in																					
27	GPIO17	VDD_SDIO						GPIO17	I/O/T		GPIO17	I/O/T	HS1_DATA5	II/O/T	U2TXD	O	EMAC_CLK_OUT_1B0	O	2'd2	oe=0, ie=0	oe=0, ie=1			
28	SD_DATA_2	VDD_SDIO						SD_DATA2	II/O/T	SPIHD	I/O/T	GPIO09	I/O/T	HS1_DATA2	II/O/T	U1RXD	II				2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu	
29	SD_DATA_3	VDD_SDIO						SD_DATA3	II/O/T	SPIWP	I/O/T	GPIO10	I/O/T	HS1_DATA3	II/O/T	U1TXD	O				2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu	
30	SD_CMD	VDD_SDIO						SD_CMD	II/O/T	SPICSO	I/O/T	GPIO11	I/O/T	HS1_CMD	II/O/T	U1RTS	O				2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu	
31	SD_CLK	VDD_SDIO						SD_CLK	IO	SPICLK	I/O/T	GPIO16	I/O/T	HS1_CLK	O	U1CTS	II				2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu	
32	SD_DATA_0	VDD_SDIO						SD_DATA0	II/O/T	SPIQ	I/O/T	GPIO07	I/O/T	HS1_DATA0	II/O/T	U2RTS	O				2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu	
33	SD_DATA_1	VDD_SDIO						SD_DATA1	II/O/T	SPIR	I/O/T	GPIO08	I/O/T	HS1_DATA1	II/O/T	U2CTS	II				2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu	
34	GPIO5	VDD3P3_CPU						GPIO5	I/O/T	VSPICSO	I/O/T	GPIO05	I/O/T	HS1_DATA6	II/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	II/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	IO					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	UOCTS	II			EMAC_RXD0	O	2'd2	oe=0, ie=0	oe=0, ie=1		
39	GPIO22	VDD3P3_CPU						GPIO22	I/O/T	VSPWP	I/O/T	GPIO22	I/O/T	UORTS	O			EMAC_RXD1	O	2'd2	oe=0, ie=0	oe=0, ie=1		
40	U0RXD	VDD3P3_CPU						U0RXD	II	CLK_OUT2	O	GPIO03	I/O/T								2'd2	oe=0, ie=1, wpu	oe=0, ie=1, wpu	
41	U0TXD	VDD3P3_CPU						U0TXD	O	CLK_OUT3	O	GPIO01	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	VSPIH	I/O/T	GPIO21	I/O/T					EMAC_RX_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。)

## 修订历史

日期	版本	发布说明
2024.08	v4.6	<p>优化以下章节的格式、结构和表述：</p> <ul style="list-style-type: none"> <li>• 章节 2 管脚</li> <li>• 章节 3 启动配置项（曾用名“Strapping 管脚”）</li> <li>• 章节 4 功能描述</li> </ul>
2024.02	v4.5	章节 2.5.3 芯片上电和复位：将 VDD_SDIO 1.8 V 模式的参考电路原理图链接至《ESP32 硬件设计指南》
2023.12	v4.4	表 1-1 型号对比：增加 flash 的规格信息
2023.07	v4.3	<ul style="list-style-type: none"> <li>• 全文格式更新</li> <li>• 部分章节描述更新</li> <li>• 新增章节 2.3.1 GPIO 和 RTC_GPIO 的限制</li> <li>• 新增章节 4.1.5 Cache</li> </ul>
2023.01	v4.2	<ul style="list-style-type: none"> <li>• 删除霍尔传感器相关内容，具体请见 <a href="#">PCN20221202</a></li> <li>• 章节 4.9.3 触摸传感器：增加应用限制的说明</li> </ul>
2022.12	v4.1	<ul style="list-style-type: none"> <li>• 章节 4.1.1 CPU：增加 <a href="#">Xtensa® 指令集架构 (ISA) 说明</a> 的链接</li> <li>• 表 1 ESP32 系列型号对比：更新有关芯片版本升级的说明</li> </ul>
2022.10	v4.0	<ul style="list-style-type: none"> <li>• 章节 <a href="#">产品概述</a>：更新部分描述</li> <li>• 表 2-6 芯片与 flash/PSRAM 的管脚对应关系：新增两条说明</li> <li>• 章节 2.5.2 电源管理：新增一条关于对电源电压的说明</li> <li>• 更新图 1-1 ESP32 系列芯片命名规则</li> <li>• 表 1 <a href="#">ESP32 系列型号对比</a>：新增一列“VDD_SDIO 电压”</li> <li>• 章节 4.8.12 双线汽车接口：更新 TWAI 支持的速率</li> <li>• 标注 ESP32-S0WD 芯片型号为 <a href="#">不推荐用于新设计 (NRND)</a></li> </ul>
2022.03	v3.9	<ul style="list-style-type: none"> <li>• 新增一款芯片型号：ESP32-D0WDR2-V3</li> <li>• 章节 4 功能描述：添加相关链接</li> <li>• 更新图 6-2 QFN48 (5×5 mm) 封装</li> <li>• 更新附录 IO_MUX</li> <li>• 更新表 4-6 外设管脚分配</li> <li>• 新增表 2-5 芯片与 flash/PSRAM 的管脚对应关系 和表 2-6 芯片与 flash/PSRAM 的管脚对应关系</li> </ul>

见下页

## 接上页

日期	版本	发布说明
2021.10	v3.8	<ul style="list-style-type: none"> <li>ESP32-U4WDH 芯片从单核升级为双核, 见 <a href="#">PCN-2021-021</a>。2021 年 12 月 2 日前后为新旧版本过渡, 实物情况以批次跟踪为准。</li> <li>章节 <a href="#">CPU 和存储</a>: 新增 CoreMark® 得分</li> <li>章节 <a href="#">4.8.12 双线汽车接口</a>: 更新描述</li> <li>章节 <a href="#">6 封装</a>: 添加 <a href="#">《乐鑫芯片包装信息》</a> 链接</li> <li>标注 ESP32-DOWDQ6-V3 芯片型号为 <a href="#">不推荐用于新设计 (NRND)</a></li> <li>更新章节 <a href="#">蓝牙</a></li> </ul>
2021.07	v3.7	<ul style="list-style-type: none"> <li>移除 ESP32-D2WD 芯片型号</li> <li>章节 <a href="#">4.71 蓝牙射频和基带</a>: 更新描述</li> <li>更新管脚功能编号, 编号从 Function0 开始</li> <li>标注 ESP32-DOWD 和 ESP32-DOWDQ6 芯片型号为 <a href="#">不推荐用于新设计 (NRND)</a></li> </ul>
2021.03	V3.6	<ul style="list-style-type: none"> <li>更新图 <a href="#">功能框图</a></li> <li>更新表 <a href="#">5.5 可靠性</a></li> <li>更新表 <a href="#">5-2 建议工作条件</a></li> <li>表 <a href="#">2-4 上电和复位时序参数说明</a>: 更新说明</li> <li>更新图 <a href="#">2-3 ESP32 电源管理</a></li> <li>表 <a href="#">4-1 地址映射结构</a>、表 <a href="#">4-6 外设管脚分配</a>、表 <a href="#">GPIO_Matrix</a>、章节 <a href="#">4.8.12 双线汽车接口</a>: 增加 TWAI® 相关信息</li> </ul>
2021.01	V3.5	<ul style="list-style-type: none"> <li>更新图 <a href="#">功能框图</a></li> <li>表 <a href="#">2-1 管脚概述</a>: 将 CAP2 并联电容由 3 nF 更新为 3.3 nF</li> <li>章节 <a href="#">高级外设接口</a>: 增加双线汽车接口 (TWAI®, 兼容 ISO11898-1)</li> <li>附录 <a href="#">GPIO_Matrix</a>: 更新 MTCK、MTMS、GPIO27 的复位状态</li> </ul>
2020.04	V3.4	<ul style="list-style-type: none"> <li>增加一款芯片型号: ESP32-U4WDH</li> <li>更新表格 <a href="#">4-2 电源管理单元 (PMU)</a>, <a href="#">5-6 Wi-Fi 射频</a>, <a href="#">5-7 接收器 - 基础数据率 (BR)</a>, <a href="#">5-9 接收器 - 增强数据率 (EDR)</a>, <a href="#">5-11 接收器</a>, <a href="#">5-12 发射器</a> 中的一些数据</li> <li>表 <a href="#">5-7 接收器 - 基础数据率 (BR)</a>: 增加一条说明</li> </ul>
2020.01	V3.3	<ul style="list-style-type: none"> <li>增加两个芯片型号: ESP32-DOWD-V3 和 ESP32-DOWDQ6-V3</li> <li>表 <a href="#">4-3 模/数转换器 (ADC)</a>: 增加一条说明</li> </ul>

见下页

接上页

日期	版本	发布说明
2019.10	V3.2	<ul style="list-style-type: none"> <li>更新图 <a href="#">2-4 上电和复位时序参数图</a></li> </ul>
2019.07	V3.1	<ul style="list-style-type: none"> <li>表 <a href="#">2-1 管脚概述</a>: 增加 ESP32-D2WD 与内置 flash 的管脚对应关系</li> <li>图 <a href="#">1-1 ESP32 系列芯片命名规则</a> 增加产品型号</li> </ul>
2019.04	V3.0	<ul style="list-style-type: none"> <li>章节 <a href="#">3 启动配置项</a> (曾用名“Strapping 管脚”): 增加关于 CHIP_PU 上电前和上电后 Strapping 管脚的建立时间和保持时间的说明</li> </ul>
2019.02	V2.9	<ul style="list-style-type: none"> <li>表 <a href="#">2-1 管脚概述</a>: 更新“管脚描述”的格式</li> <li>表 <a href="#">4-6 外设管脚分配</a>: 修改有关 ADC1 管脚映射的笔误</li> </ul>
2019.01	V2.8	<ul style="list-style-type: none"> <li>表 <a href="#">5-7 接收器 - 基础数据率 (BR)</a>、表 <a href="#">5-10 发射器 - 增强数据率 (EDR)</a> 和表 <a href="#">5-12 发射器</a>: 将“射频功率控制范围”从-12 ~ +12 改为-12 ~ +9 dBm</li> <li>一些文字修改</li> </ul>
2018.11	V2.7	<ul style="list-style-type: none"> <li>更新章节 <a href="#">应用</a></li> <li>表 <a href="#">IO_MUX</a>: 更新管脚复位状态</li> </ul>
2018.10	V2.6	<ul style="list-style-type: none"> <li>章节 <a href="#">6 封装</a>: 更新两个封装图</li> </ul>
2018.08	V2.5	<ul style="list-style-type: none"> <li>表 <a href="#">5-1 绝对最大额定值</a>: 增加“IO 输出总电流”</li> <li>表 <a href="#">4-3 模/数转换器 (ADC)</a>: 增加各个电源域的拉电流平均值</li> <li>表 <a href="#">IO_MUX</a>: 更新电源域的名称</li> </ul>
2018.07	V2.4	<ul style="list-style-type: none"> <li>删除通信包仲裁 (PTA) 相关内容</li> <li>章节 <a href="#">2.5.2 电源管理</a>: 增加芯片上电、复位时序图和参数说明</li> <li>表 <a href="#">4-2 电源管理单元 (PMU)</a>: 增加双核芯片的功耗</li> <li>章节 <a href="#">4.9.1 模/数转换器 (ADC)</a>: 增加 ADC 特性和校准结果</li> </ul>
2018.06	V2.3	<ul style="list-style-type: none"> <li>表 <a href="#">4-2 电源管理单元 (PMU)</a>: 增加 160 MHz 功耗</li> </ul>

见下页

接上页

日期	版本	发布说明
2018.05	V2.2	<ul style="list-style-type: none"> <li>• 表 <a href="#">2.2 管脚概述</a>: 将 VDD3P3_RTC 的电压范围由 1.8-3.6V 改为 2.3-3.6V</li> <li>• 更新章节 <a href="#">2.5.2 电源管理</a></li> <li>• 更新章节 <a href="#">4.1.3 封装外 flash 和 RAM</a></li> <li>• 更新表 <a href="#">4-2 电源管理单元 (PMU)</a></li> <li>• 删除有关温度传感器内容</li> </ul> <p>电气特性相关的更新:</p> <ul style="list-style-type: none"> <li>• 更新表 <a href="#">5-1 绝对最大额定值</a></li> <li>• 增加表 <a href="#">5-2 建议工作条件</a></li> <li>• 增加表 <a href="#">4-3 模/数转换器 (ADC)</a></li> <li>• 增加表 <a href="#">5.5 可靠性</a></li> <li>• 表 <a href="#">5-7 接收器 - 基础数据率 (BR)</a>: 更新“增益控制步长”、“邻道发射功率”参数</li> <li>• 表 <a href="#">5-10 发射器 - 增强数据率 (EDR)</a>: 更新“增益控制步长”，“<math>\pi/4</math> DQPSK 调制精度”、“8 DPSK 调制精度”和“带内杂散发射”参数</li> <li>• 表 <a href="#">5-12 发射器</a>: 更新“增益控制步长”、“邻道发射功率”参数</li> </ul>
2018.01	V2.1	<ul style="list-style-type: none"> <li>• 删除部分软件相关的内容</li> <li>• 删除超低噪声前置模拟放大器相关的内容</li> <li>• 说明 ESP32-D2WD 的 CPU 速率为 160 MHz, 嵌入式 flash 为 40 MHz</li> <li>• 章节 <a href="#">2.5.2 电源管理</a>: 增加说明</li> </ul>
2017.12	V2.0	<ul style="list-style-type: none"> <li>• 章节 <a href="#">6 封装</a>: 增加管脚方向的说明</li> </ul>
2017.10	V1.9	<ul style="list-style-type: none"> <li>• 表 <a href="#">2-1 管脚概述</a>: 更新管脚 CHIP_PU 的描述</li> <li>• 章节 <a href="#">2.5.2 电源管理</a>: 增加一条说明</li> <li>• 章节 <a href="#">3 启动配置项 (曾用名 “Strapping 管脚”)</a>: 修改关于芯片系统复位的描述</li> <li>• 章节 <a href="#">4.6.4 Wi-Fi 射频和基带</a>: 增加天线分集和选择的描述</li> <li>• 表 <a href="#">4-2 电源管理单元 (PMU)</a>: 删除“关联睡眠方式”，增加关于 Active sleep 和 Modem-sleep 的说明</li> </ul>
2017.08	V1.8	<ul style="list-style-type: none"> <li>• 增加表 <a href="#">4-6 外设管脚分配</a></li> <li>• 图 <a href="#">功能框图</a>: 修改一处笔误</li> </ul>

见下页

## 接上页

日期	版本	发布说明
2017.08	V1.7	<ul style="list-style-type: none"> <li>章节 <a href="#">Bluetooth</a>: 将输出功率改为“高达 12 dBm”，将 NZIF 接收器的 BLE 接收灵敏度改为 -97 dBm</li> <li>表 <a href="#">2-1 管脚概述</a>: 增加一条说明</li> <li>章节 <a href="#">4.1.1 CPU</a>: 增加 160 MHz 时钟频率</li> <li>章节 <a href="#">4.6.4 Wi-Fi 射频和基带</a>: 将发射功率改为 20.5 dBm</li> <li>章节 <a href="#">4.7.1 蓝牙射频和基带</a>: 将 Class-1、Class-2 和 Class-3 发射输出功率的动态控制范围改为“高达 24 dB”，将 NZIF 接收器的动态控制范围改为“超过 97 dB”</li> <li>表 <a href="#">4-2 电源管理单元 (PMU)</a>: 更新不同功耗模式下的功耗，并增加两条说明</li> <li>更新章节 <a href="#">4.8.1 通用输入/输出接口 (GPIO)</a></li> <li>更新章节 <a href="#">4.8.11 SDIO/SPI 从机控制器</a></li> <li>更新表 <a href="#">5-1 绝对最大额定值</a></li> <li>表 <a href="#">5-4 Active 模式下的 RF 功耗</a>: 更新射频功耗参数，并将发射数据所基于的占空比改为 50%</li> <li>表 <a href="#">5-6 Wi-Fi 射频</a>: 增加关于“输出阻抗”的说明</li> <li>表 <a href="#">5-7、5-9、5-11</a>: 更新“灵敏度”参数</li> <li>表 <a href="#">5-7、5-10、5-12</a>: 更新“射频发射功率”，“射频功率控制范围”参数，并增加“增益控制步长”参数</li> <li>删除章节触摸传感器和示例代码</li> <li>增加产品证书下载 <a href="#">链接</a></li> </ul>
2017.06	V1.6	<ul style="list-style-type: none"> <li>章节高集成度: 将外部元器件个数改为 20</li> <li>章节 <a href="#">4.8.1 通用输入/输出接口 (GPIO)</a>: 将 GPIO 管脚个数改为 34</li> </ul>
2017.06	V1.5	<ul style="list-style-type: none"> <li>章节 <a href="#">CPU 和存储</a>: 更新供电电压范围</li> <li>章节 <a href="#">2.5.2 电源管理</a>: 更新说明</li> <li>更新表 <a href="#">5-1 绝对最大额定值</a></li> <li>表 <a href="#">管脚清单说明</a>: 更新数字输出管脚的驱动强度</li> <li>增加文档变更通知</li> </ul>
2017.05	V1.4	<ul style="list-style-type: none"> <li>章节 <a href="#">时钟和定时器</a>: 增加对于外置晶振频率的说明</li> <li>章节 <a href="#">3 启动配置项 (曾用名“Strapping 管脚”)</a>: 增加说明</li> <li>表 <a href="#">5-1 绝对最大额定值</a>: 将最大驱动能力由 12 mA 改为 80 mA</li> <li>表 <a href="#">5-6 Wi-Fi 射频</a>: 将输入阻抗值 <math>50\Omega</math> 改为输出阻抗值 <math>30+j10\Omega</math></li> <li>更新章节 <a href="#">4.3 RTC 和低功耗管理</a></li> <li>表 <a href="#">管脚清单说明</a>: 增加一条说明</li> <li>表 <a href="#">IO_MUX</a>: 删除 GPIO20</li> </ul>

见下页

接上页

日期	版本	发布说明
2017.04	V1.3	<ul style="list-style-type: none"> <li>• 增加附录 管脚清单说明</li> <li>• 更新表格 5-6 Wi-Fi 射频</li> <li>• 更新图 2-2 ESP32 管脚布局 (封装为 QFN 5*5, 顶视图)</li> </ul>
2017.03	V1.2	<ul style="list-style-type: none"> <li>• 表 2-1 管脚概述: 增加说明</li> <li>• 章节 4.1.2 片上存储: 增加说明</li> </ul>
2017.02	V1.1	<ul style="list-style-type: none"> <li>• 增加章节 1 ESP32 系列型号对比</li> <li>• 更新章节 MCU 和高级特性</li> <li>• 更新章节功能框图</li> <li>• 更新章节 2 管脚</li> <li>• 更新章节 CPU 和存储</li> <li>• 更新章节 4.2.3 音频 PLL 时钟</li> <li>• 更新章节 5.1 绝对最大额定值</li> <li>• 更新章节 6</li> <li>• 更新章节学习资源</li> </ul>
2016.08	V1.0	首次发布



## 免责声明和版权公告

本文档中的信息，包括供参考的 URL 地址，如有变更，恕不另行通知。

本文档可能引用了第三方的信息，所有引用的信息均为“按现状”提供，乐鑫不对信息的准确性、真实性做任何保证。

乐鑫不对本文档的内容做任何保证，包括内容的适销性、是否适用于特定用途，也不提供任何其他乐鑫提案、规格书或样品在他处提到的任何保证。

乐鑫不对本文档是否侵犯第三方权利做任何保证，也不对使用本文档内信息导致的任何侵犯知识产权的行为负责。本文档在此未以禁止反言或其他方式授予任何知识产权许可，不管是明示许可还是暗示许可。

Wi-Fi 联盟成员标志归 Wi-Fi 联盟所有。蓝牙标志是 Bluetooth SIG 的注册商标。

文档中提到的所有商标名称、商标和注册商标均属其各自所有者的财产，特此声明。

版权归 © 2024 乐鑫信息科技（上海）股份有限公司。保留所有权利。

[www.espressif.com](http://www.espressif.com)