

# ESP32 硬件设计指南



**Espressif Systems**

2016 年 12 月 9 日

# 关于本手册

《**ESP32 硬件设计指南**》提供了 ESP32 系列产品的硬件信息，包括 ESP32 芯片，ESP-WROOM-32 模组以及 ESP32-DevKitC 开发板。

## 相关资源

有关 ESP32 的其他文档及资源，请登录乐鑫官网查看：[ESP32 资源](#)。

## 发布说明

日期	版本	发布说明
2016.12	V1.0	首次发布。
2016.12	V1.1	更新表 <a href="#">4</a> 。

## 免责申明和版权公告

本文中的信息，包括供参考的 URL 地址，如有变更，恕不另行通知。文档“按现状”提供，不负任何担保责任，包括对适销性、适用于特定用途或非侵权性的任何担保，和任何提案、规格或样品在他处提到的任何担保。

本文档不负任何责任，包括使用本文档内信息产生的侵犯任何专利权行为的责任。本文档在此未以禁止反言或其他方式授予任何知识产权使用许可，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 © 2016 乐鑫所有。保留所有权利。

# 目录

<b>1 产品概述</b>	6
1.1 基础协议	6
1.1.1 Wi-Fi	6
1.1.2 蓝牙	6
1.2 应用	7
1.3 功能框图	8
<b>2 管脚定义</b>	9
2.1 管脚布局	9
2.2 管脚定义	9
2.3 Strapping 管脚	11
<b>3 电路图和版图设计</b>	13
3.1 原理图设计	13
3.1.1 电源	14
3.1.1.1 数字电源	14
3.1.1.2 模拟电源	14
3.1.2 上电时序与复位	15
3.1.2.1 上电时序	15
3.1.2.2 复位	15
3.1.3 Flash	15
3.1.4 晶振	16
3.1.4.1 外部时钟参考 (必选)	16
3.1.4.2 RTC 时钟 (可选)	17
3.1.5 射频 (RF)	17
3.1.6 传感器外部采样电容	17
3.1.7 外置阻容	18
3.2 版图布局	18
3.2.1 独立的 ESP32 模组的版图设计	18
3.2.1.1 版图设计通用要点	18
3.2.1.2 电源	19
3.2.1.3 晶振	19
3.2.1.4 射频	20
3.2.2 ESP32 作为从设备的版图设计	20
3.2.3 版图设计常见问题	21
3.2.3.1 为什么电源纹波并不大，但射频的 Tx 性能很差？	21
3.2.3.2 为什么芯片发包时，电源纹波很小，但射频的 Tx 性能不好？	22
3.2.3.3 为什么 ESP32 发包时，仪器测试到的 power 值比 target power 值要高或者低很多，且 EVM 比较差？	22
3.2.3.4 为什么芯片的 Tx 性能没有问题，但 Rx 的灵敏度不好？	22
<b>4 开发硬件介绍</b>	23
4.1 ESP-WROOM-32 模组	23
4.1.1 概述	23

4.1.2 管脚说明	24
4.1.3 使用注意事项	25
4.2 ESP32-DevKitC (开发板)	26
4.2.1 概述	26
4.2.2 原理图	27
4.2.2.1 电源原理图	27
4.2.2.2 USB 转串口原理图	27
<b>5 典型应用案例</b>	<b>28</b>
5.1 UART 串口转 Wi-Fi 智能硬件	28
5.2 ESP32-Lyra (智能音频平台)	28

## 表格

1	管脚定义	9
2	Strapping 管脚	12
3	WROOM-32 管脚说明	24
4	UART 接口的管脚定义	28

# 插图

1	ESP32 功能框图	8
2	ESP32 管脚布局	9
3	ESP32 原理图	13
4	ESP32 数字电源	14
5	ESP32 模拟电源	15
6	ESP32 Flash	16
7	ESP32 晶振	16
8	ESP32 晶振 (RTC 时钟)	17
9	ESP32 射频电源图	17
10	ESP32 传感器采样电容	18
11	ESP32 外置电容	18
12	ESP32 版图设计	19
13	ESP32 晶振设计	20
14	ESP32 射频部分版图设计	20
15	PAD/TV Box 平面位置规划框架	21
16	ESP-WROOM-32 模组外观图	23
17	ESP-WROOM-32 模组外观尺寸图	23
18	ESP32-DevKitC 尺寸图	26
19	ESP32-DevKitC 布局图	26
20	ESP32-DevKitC 电源原理图	27
21	USB 转串口原理图	27

# 1. 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) 超低功耗的 40 纳米工艺，拥有最佳的功耗性能、射频性能、稳定性、通用性和可靠性，适用于各种应用和不同功耗需求。

ESP32 是业内集成度领先的 Wi-Fi + 蓝牙解决方案，外部元器件仅有 16 个，并且集成了天线开关、射频 balun、功率放大器、低噪放大器、过滤器、电源管理模块和先进的自校准电路，极大减少了印刷电路板 (PCB) 的面积。

ESP32 还集成了先进的自校准电路，实现了动态自动调整，以消除外部电路的缺陷，更好地适应外部环境的变化。因此，ESP32 的批量生产不需要昂贵的专用 Wi-Fi 测试设备。

## 1.1 基础协议

### 1.1.1 Wi-Fi

- 802.11 b/g/n/e/i
- 802.11 n (2.4 GHz)，速度高达 150 Mbps
- 802.11 e: QoS 机制实现无线多媒体技术
- WMM-PS, UAPSD
- A-MPDU 和 A-MSDU 帧聚合技术
- 块回复
- 分片和重组
- Beacon 自动监测 / 扫描
- 802.11 i 安全特性：预认证和 TSN
- 支持 WPA / WPA2 / WPA2-Enterprise / WPS 加密
- 基础结构型网络 (Infrastructure BSS) Station 模式 / SoftAP 模式
- Wi-Fi Direct (P2P)、P2P 发现、P2P GO 模式和 P2P 电源管理
- UMA 兼容和认证
- 天线分集与选择

### 1.1.2 蓝牙

- 蓝牙 v4.2 完整标准，包含传统蓝牙 (BR/EDR) 和低功耗蓝牙 (BLE)
- 支持标准 Class-1、Class-2 和 Class-3，且无需外部功率放大器
- 加强的精准功率控制
- 输出功率高达 +10 dBm
- NZIF 接收器具有 -98 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
- 速度高达 4 Mbps 的高速 UART HCI
- 支持 BT 4.2 controller 和 host 协议栈
- 服务发现协议 (SDP)
- 通用访问应用 (GAP)
- 安全管理协议 (SMP)
- 低功耗蓝牙
- ATT / GATT
- HID
- 可支持所有基于 GATT 的低功耗蓝牙应用
- SPP-Like 低功耗蓝牙数据透传协议
- BLE Beacon
- A2DP / AVRCP / SPP, HSP / HFP, RFCOMM
- CVSD 和 SBC 音频编解码算法
- 蓝牙微微网 (Piconet) 和散射网 (Scatternet)

## 1.2 应用

- 通用低功耗 IoT 传感器 Hub
- 通用低功耗 IoT 记录器
- 相机的视频流传输
- OTT 电视盒 / 机顶盒设备
- 音乐播放器
  - 网络音乐播放器
  - 音频流媒体设备
- Wi-Fi 玩具
  - 计数器
  - 玩具防丢器
- Wi-Fi 语音识别设备
- 耳麦
- 智能插座

- 家庭自动化
- Mesh 网络
- 工业无线控制
- 婴儿监控器
- 传感器网络
- 可穿戴电子产品
- Wi-Fi 位置感知设备
- 安全 ID 标签
- 健康医疗
  - 运动监测和防丢报警器
  - 温度记录仪

### 1.3 功能框图

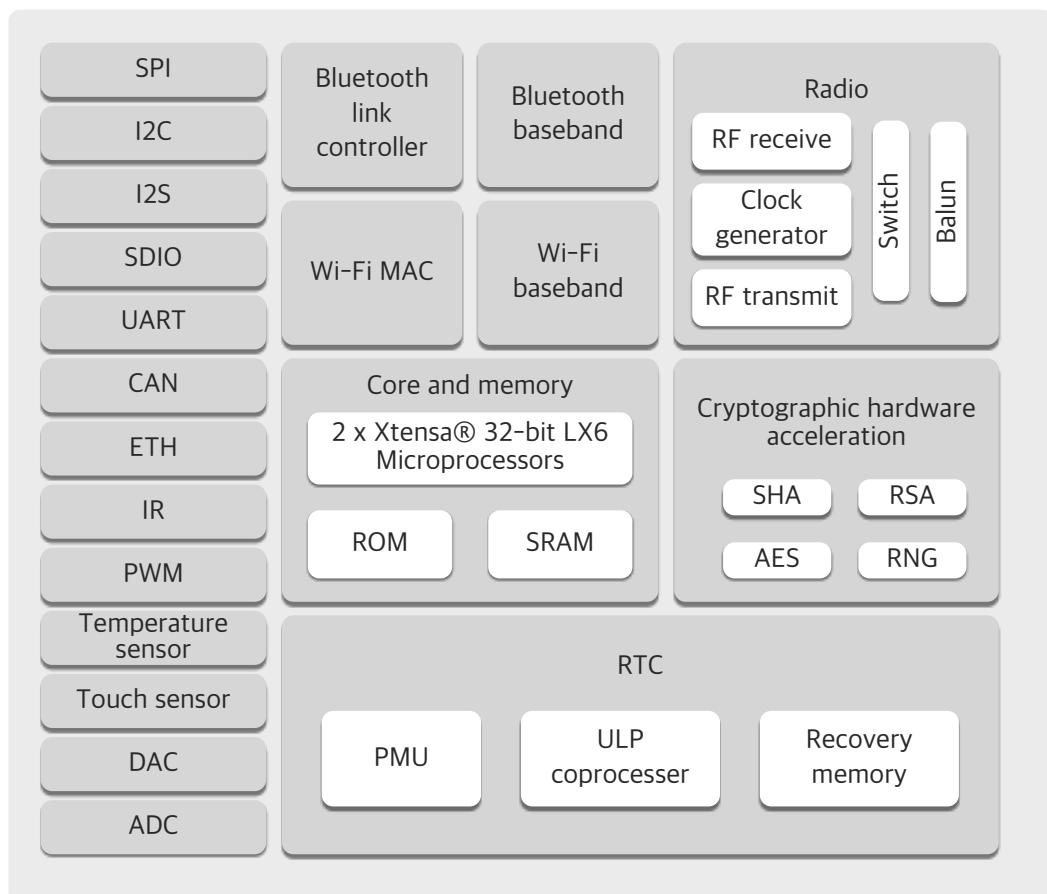


图 1: ESP32 功能框图

## 2. 管脚定义

### 2.1 管脚布局

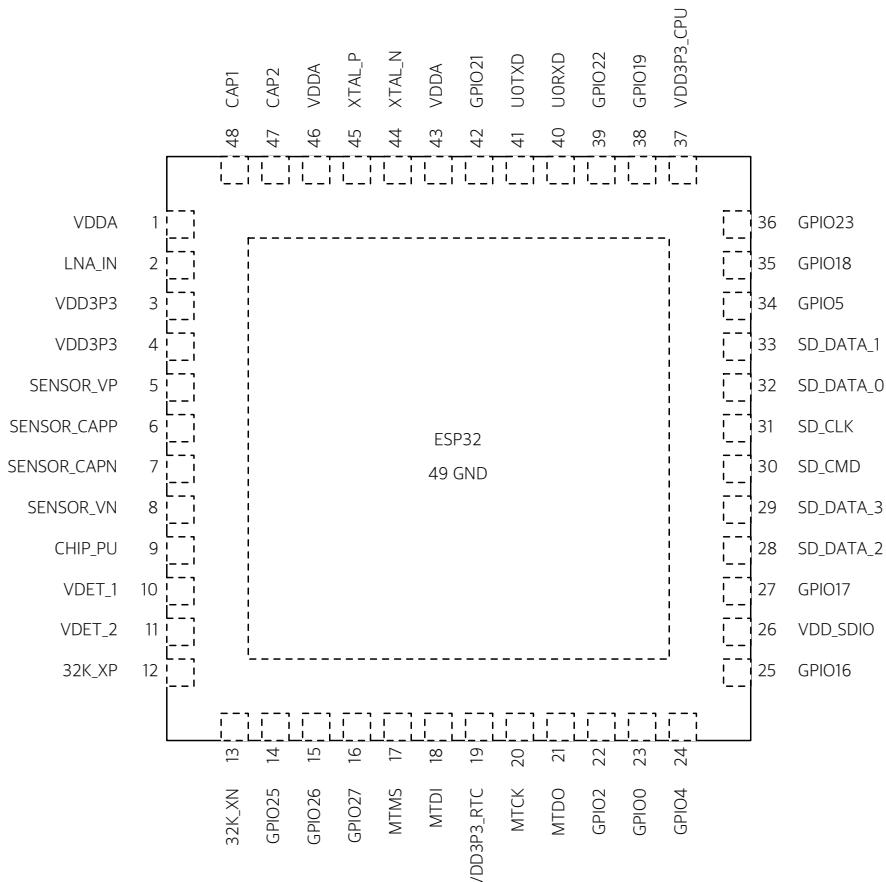


图 2: ESP32 管脚布局

### 2.2 管脚定义

表 1: 管脚定义

名称	编号	类型	功能
模拟			
VDDA	1	P	模拟电源 (2.3V ~ 3.6V)
LNA_IN	2	I/O	射频输入和输出
VDD3P3	3	P	放大器电源 (2.3V ~ 3.6V)
VDD3P3	4	P	放大器电源 (2.3V ~ 3.6V)
VDD3P3_RTC			
SENSOR_VP	5	I	GPIO36, ADC_PRE_AMP, ADC1_CH0, RTC_GPIO0 注意: 作为 ADC_PRE_AMP 使用时, 将 270 pF 电容从 SENSOR_VP 连接到 SENSOR_CAPP 上。

名称	编号	类型	功能
SENSOR_CAPP	6	I	GPIO37, ADC_PRE_AMP, ADC1_CH1, RTC_GPIO1 注意：作为 ADC_PRE_AMP 使用时，将 270 pF 电容从 SENSOR_VP 连接到 SENSOR_CAPP 上。
SENSOR_CAPN	7	I	GPIO38, ADC1_CH2, ADC_PRE_AMP, RTC_GPIO2 注意：作为 ADC_PRE_AMP 使用时，将 270 pF 电容从 SENSOR_VN 连接到 SENSOR_CAPN 上。
SENSOR_VN	8	I	GPIO39, ADC1_CH3, ADC_PRE_AMP, RTC_GPIO3 注意：作为 ADC_PRE_AMP 使用时，将 270 pF 电容从 SENSOR_VN 连接到 SENSOR_CAPN 上
CHIP_PU	9	I	芯片使能（高电平有效） 高电平：上电，芯片正常工作； 低电平：断电，芯片以最小功率工作； 注意：不能让 CHIP_PU 管脚悬浮。
VDET_1	10	I	GPIO34, ADC1_CH6, RTC_GPIO4
VDET_2	11	I	GPIO35, ADC1_CH7, RTC_GPIO5
32K_XP	12	I/O	GPIO32, 32K_XP (32.768 kHz 晶振输入), ADC1_CH4, TOUCH9, RTC_GPIO9
32K_XN	13	I/O	GPIO33, 32K_XN (32.768 kHz 晶振输出), ADC1_CH5, TOUCH8, RTC_GPIO8
GPIO25	14	I/O	GPIO25, DAC_1, ADC2_CH8, RTC_GPIO6, EMAC_RXD0
GPIO26	15	I/O	GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1
GPIO27	16	I/O	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
MTMS	17	I/O	GPIO14, ADC2_CH6, TOUCH6, RTC_GPIO16, MTMS, HSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2
MTDI	18	I/O	GPIO12, ADC2_CH5, TOUCH5, RTC_GPIO15, MTDI, HSPIQ, HS2_DATA2, SD_DATA2, EMAC_TXD3
VDD3P3_RTC	19	P	RTC IO 电源输入 (1.8V ~ 3.3V)
MTCK	20	I/O	GPIO13, ADC2_CH4, TOUCH4, RTC_GPIO14, MTCK, HSPIID, HS2_DATA3, SD_DATA3, EMAC_RX_ER
MTDO	21	I/O	GPIO15, ADC2_CH3, TOUCH3, RTC_GPIO13, MTDO, HSPICS0, HS2_CMD, SD_CMD, EMAC_RXD3
GPIO2	22	I/O	GPIO2, ADC2_CH2, TOUCH2, RTC_GPIO12, HSPIWP, HS2_DATA0, SD_DATA0
GPIO0	23	I/O	GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK
GPIO4	24	I/O	GPIO4, ADC2_CH0, TOUCH0, RTC_GPIO10, HSPIHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
VDD_SDIO			
GPIO16	25	I/O	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
VDD_SDIO	26	P	1.8V 或 3.3V 电源输出
GPIO17	27	I/O	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
SD_DATA_2	28	I/O	GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD
SD_DATA_3	29	I/O	GPIO10, SD_DATA3, SPIWP, HS1_DATA3, U1TXD
SD_CMD	30	I/O	GPIO11, SD_CMD, SPICS0, HS1_CMD, U1RTS
SD_CLK	31	I/O	GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS

名称	编号	类型	功能
SD_DATA_0	32	I/O	GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS
SD_DATA_1	33	I/O	GPIO8, SD_DATA1, SPID, HS1_DATA1, U2CTS
VDD3P3_CPU			
GPIO5	34	I/O	GPIO5, VSPICS0, HS1_DATA6, EMAC_RX_CLK
GPIO18	35	I/O	GPIO18, VSPICLK, HS1_DATA7
GPIO23	36	I/O	GPIO23, VSPID, HS1_STROBE
VDD3P3_CPU	37	P	CPU IO 电源输入 (1.8V ~ 3.3V)
GPIO19	38	I/O	GPIO19, VSPIQ, U0CTS, EMAC_TXD0
GPIO22	39	I/O	GPIO22, VSPIWP, U0RTS, 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
模拟			
VDDA	43	I/O	模拟电源 (2.3V ~ 3.6V)
XTAL_N	44	O	外部晶振输出
XTAL_P	45	I	外部晶振输入
VDDA	46	P	PLL 电源 (2.3V ~ 3.6V)
CAP2	47	I	并联 3 nF 电容和 20 kΩ 电阻到 CAP1
CAP1	48	I	串联 10 nF 电容到地
GND			
GND	49	P	接地

**注意：**

GPIO36、GPIO37、GPIO38、GPIO39、GPIO34、GPIO35 仅可用作输入。

## 2.3 Strapping 管脚

ESP32 共有 6 个 Strapping 管脚。

- MTDI/GPIO12：内部下拉
- GPIO0：内部上拉
- GPIO2：内部下拉
- GPIO4：内部下拉
- MTDO(GPIO15)：内部上拉
- GPIO5：内部上拉

软件可以读取寄存器 “GPIO\_STRAPPING” 中这 6 个位的值。

在芯片上电复位过程中，Strapping 管脚对电平采样并存储到锁存器中，锁存为 “0” 或 “1”，并一直保持到芯片掉电或关闭。

每一个 Strapping 管脚都会连接内部上拉/下拉。如果一个 Strapping 管脚没有外部连接或者连接的外部线路处于高阻抗状态，内部弱上拉/下拉将决定 Strapping 管脚输入电平的默认值。

为改变 Strapping 比特的值，用户可以应用外部下拉/上拉电阻，或者应用主机 MCU 的 GPIO 控制 ESP32 上电复位时的 Strapping 管脚电平。

复位后，Strapping 管脚和普通管脚功能相同。

配置 Strapping 管脚的详细启动模式请参阅表 2。

**表 2: Strapping 管脚**

内置 LDO (VDD_SDIO) 电压					
管脚	默认	3.3V	1.8V		
MTDI(GPIO12)	下拉	0	1		
系统启动模式					
管脚	默认	SPI Flash 启动模式	下载启动模式		
GPIO0	上拉	1	0		
GPIO2	下拉	无关项	0		
系统启动过程中，UOTXD 输出 log 打印信息					
管脚	默认	UOTXD 翻转	UOTXD 静止		
MTDO(GPIO15)	上拉	1	0		
SDIO 从机信号输入输出时序					
管脚	默认	下降沿输入 下降沿输出	下降沿输入 上升沿输出	上升沿输入 下降沿输出	上升沿输入 上升沿输出
MTDO(GPIO15)	上拉	0	0	1	1
GPIO5	上拉	0	1	0	1

**说明:**

固件可以通过配置一些寄存器比特位，在启动后改变“内置 LDO (VDD\_SDIO) 电压”和“SDIO 从机信号输入输出时序”的设定。

### 3. 电路图和版图设计

ESP32 的核心电路仅由 16 个电阻电容电感、1 个无源晶振及 1 个 SPI Flash 组成。ESP32 集成了完整的射频功能，包括天线开关、射频 balun、功率放大器、低噪放大器、过滤器、电源管理模块和先进的自校准电路。

ESP32 的高度集成使其外围电路设计变得简单，为了能够更好地确保 ESP32 工作性能，下文将详细介绍 ESP32 的原理图以及版图设计。

#### 3.1 原理图设计

ESP32 的核心电路图如图 3 所示。

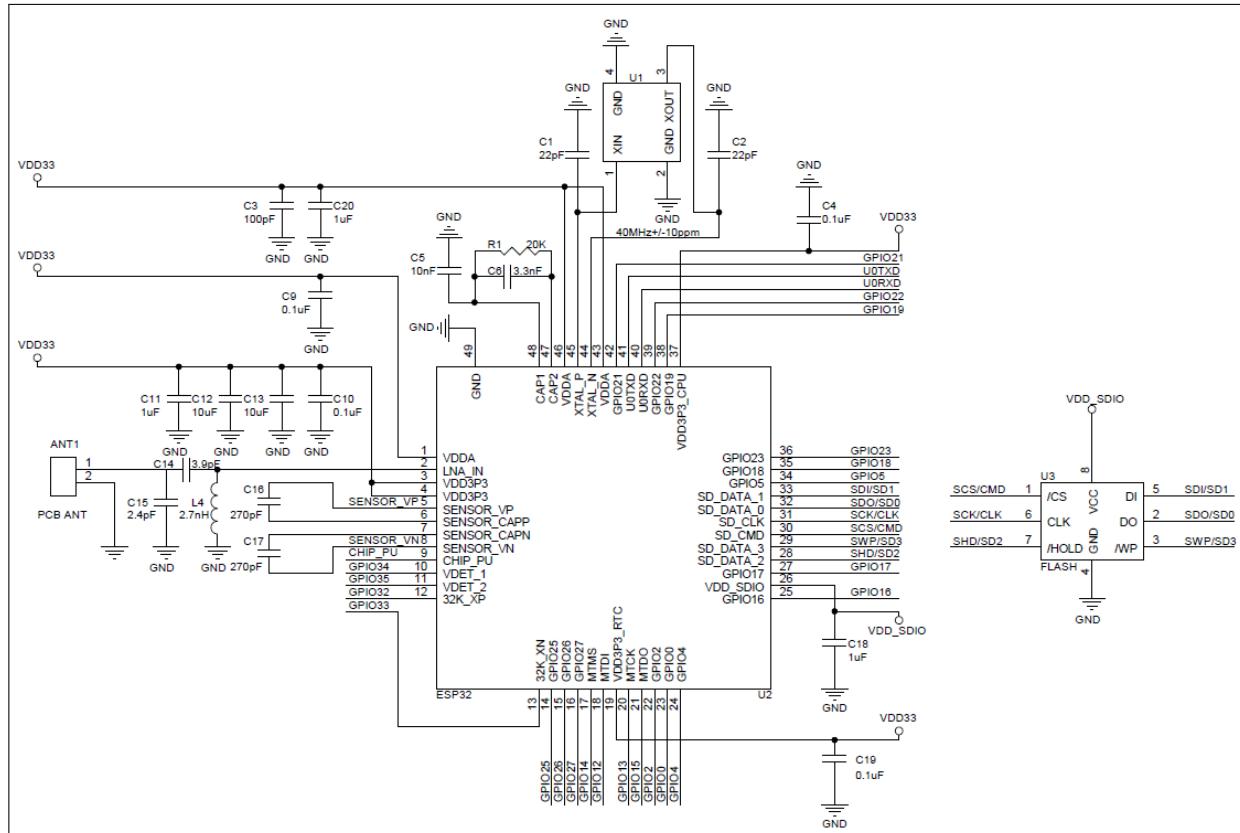


图 3: ESP32 原理图

ESP32 的核心电路图的设计有七个部分注意事项：

- 电源
- 上电时序与复位
- Flash
- 晶振
- 射频
- 传感器外部采样电容
- 外置阻容

下文将分别对这七个部分进行描述。

### 3.1.1 电源

#### 3.1.1.1 数字电源

ESP32 的 Pin19 及 Pin37 分别为 RTC 电源管脚和 CPU 电源输入管脚。数字电源工作电压范围为 1.8V ~ 3.3V。建议在电路中靠近数字电源管脚处增加滤波电容。

另外，VDD\_SDIO 内部的 LDO 可提供 1.8V 或 3.3V 电源输出给外部电路使用，最大供给电流在 40 mA 左右。可在 VDD\_SDIO 靠近管脚处添加 1  $\mu$ F 的滤波电容。当 VDD\_SDIO 与 VDD3P3\_RTC 连接在一起时，VDD\_SDIO 内部 LDO 将关闭。

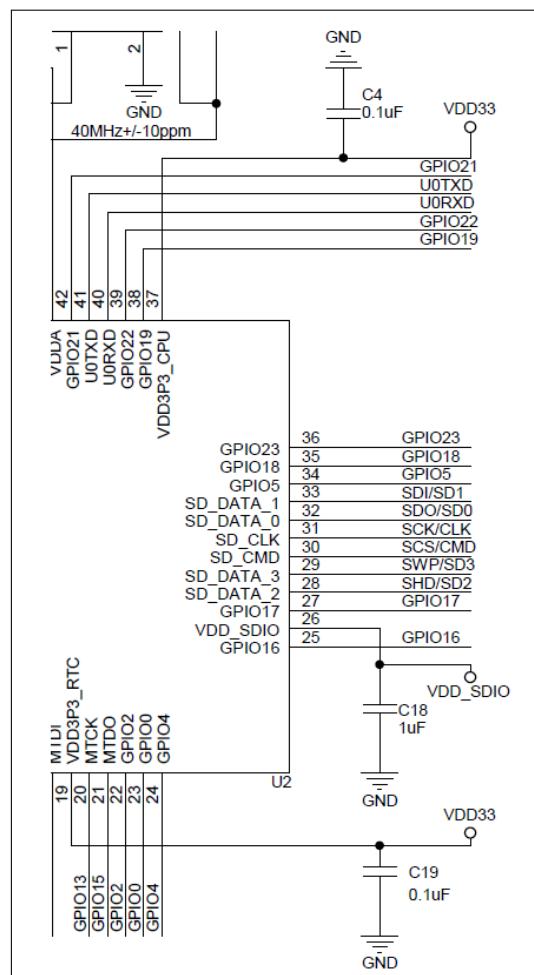


图 4: ESP32 数字电源

#### 3.1.1.2 模拟电源

ESP32 的 Pin1、Pin43、Pin46 为模拟电源，Pin3、Pin4 为功率放大器电源。该部分电源需要注意的是当 ESP32 工作在 Tx 时，瞬间电流会加大，往往引起电源的轨道塌陷。所以在电路设计时建议在电源走线上增加一个 0603 封装的 10  $\mu$ F 电容，该电容可与 0402 封装的 0.1  $\mu$ F 电容搭配使用。

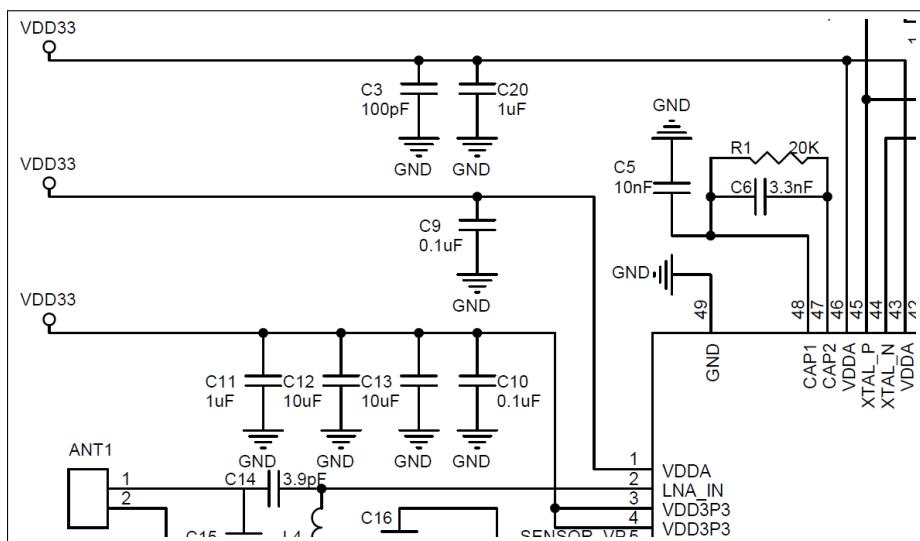


图 5: ESP32 模拟电源

### 3.1.2 上电时序与复位

#### 3.1.2.1 上电时序

ESP32 使用 3.3V 作为统一的系统电源，所以上电时序上只需遵循：ESP32 的 Pin9 CHIP\_PU 使能管脚上电要晚于或同时与系统电源 3.3V 上电。

##### 注意：

如果将 CHIP\_PU 连接到电源管理芯片，则由电源管理芯片控制 ESP32 的上电。这时，由于电源管理芯片的 GPIO 输出高低电平开关 Wi-Fi 时会产生脉冲电流，为了避免 CHIP\_PU 端上电电平不稳定，需加上 RC 延时 ( $R = 1 \text{ k}\Omega$ ,  $C = 100 \text{ nF}$ ) 电路。

#### 3.1.2.2 复位

ESP32 的复位可使用 CHIP\_PU 管脚，当 CHIP\_PU 为低电平时，ESP32 会下电。为防止外界干扰引起重启，CHIP\_PU 引线需尽量短一些，且最好加上拉电阻和对地电容。

##### 注意：

该管脚不可悬空。

### 3.1.3 Flash

ESP32 最多支持 4 个 16 MB 的外部 QSPI Flash 和 SRAM。目前采用的 Demo Flash 为 SPI Flash，ROM 大小为 4 MB，封装为 SOIC\_8 (SOP\_8)，使用 VDD\_SDIO 输出电源供电。需注意的是，需根据设置的 VDD\_SDIO 输出电源电压大小选择合适工作电压的 Flash。设计中可以考虑在 Pin21 (SD\_CLK) 管脚上串联一个 0402 封装的电阻，连接到 Flash CLK 管脚上。此电阻的作用主要为：降低驱动电流，减小串扰和外部干扰，调节时序等。

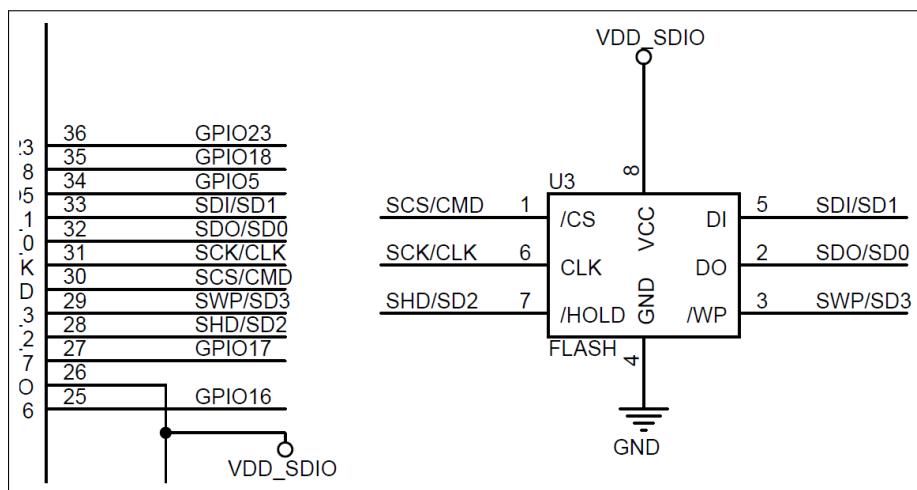


图 6: ESP32 Flash

### 3.1.4 晶振

ESP32 外部可以有 2 个晶振提供时钟源，即外部晶振时钟源和 RTC 时钟源。

#### 3.1.4.1 外部时钟参考（必选）

目前固件版本支持的晶体频率为 40 MHz, 26 MHz 及 24 MHz, 使用时请注意需在下载工具中选择对应晶体类型。晶振外部输入输出的对地调节电容 C1、C2 可不设为固定值，该值范围在 6 pF ~ 22 pF，具体值需要通过对系统测试后进行调节确定。基于目前市场中主流晶振的情况，一般 26 MHz 晶振的输入输出所加电容 C1、C2 在 10 pF 及以内；一般 40 MHz 晶振的输入输出所加电容为  $C_1 \geq 10\text{pF}$ 、 $C_2 \leq 22\text{pF}$ 。选用的晶振自身精度需在  $\pm 10\text{ PPM}$ ，工作温度范围为  $-20^\circ\text{C} \sim 85^\circ\text{C}$ 。

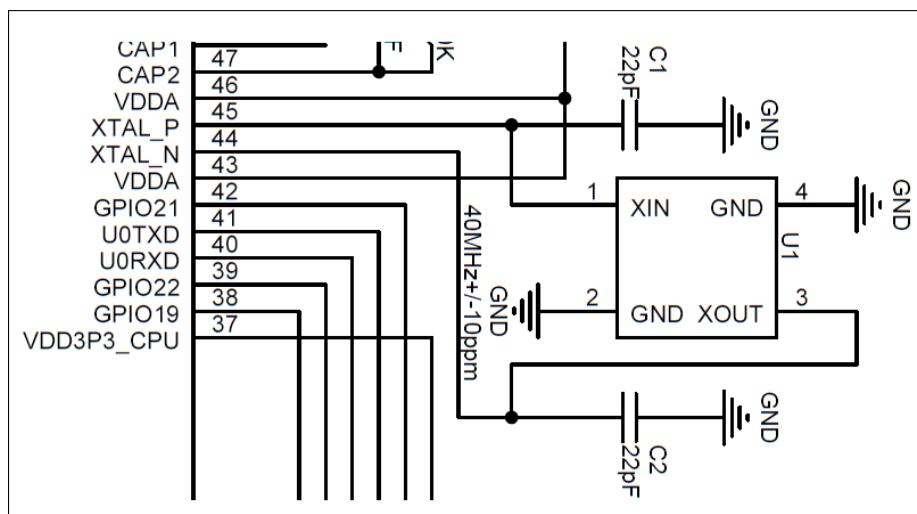


图 7: ESP32 晶振

#### 注意：

尽管 ESP32 内部带有自校准功能，但是晶振本身的质量问题，比如自身频偏过大，工作温度稳定性不高等仍然会影响 ESP32 的正常工作，导致射频指标性能下降。

### 3.1.4.2 RTC 时钟 (可选)

ESP32 支持外置 32 kHz 的时钟振荡器作为 RTC 睡眠时钟。

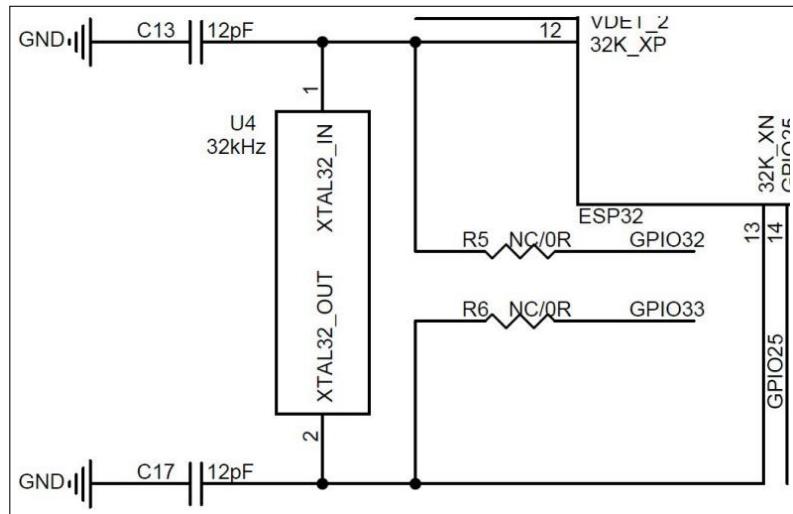


图 8: ESP32 晶振 (RTC 时钟)

#### 注意:

如果不需该 RTC 时钟源，则 Pin12 32K\_XP 和 Pin13 32K\_XN 也可配置为数字的 GPIO 口使用。

### 3.1.5 射频 (RF)

设计时需添加  $\pi$  型匹配网络对天线进行匹配。

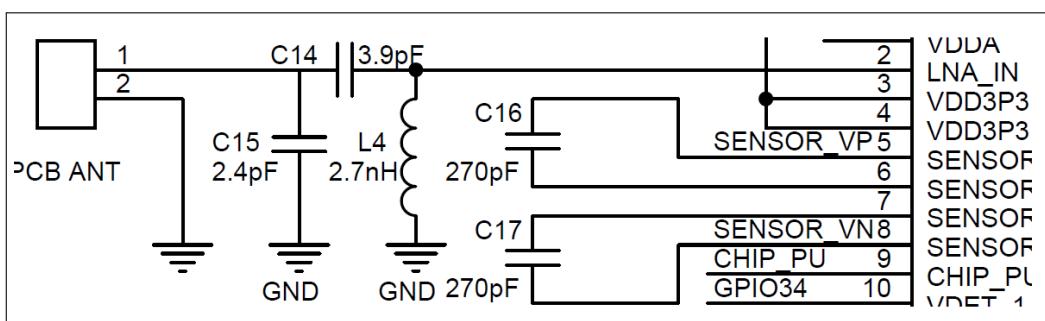


图 9: ESP32 射频电源图

### 3.1.6 传感器外部采样电容

SENSOR\_VP 和 SENSOR\_CAPP, SENSOR\_CAPN 和 SENSOR\_VN 间外接的电容 (270 pF) 作为芯片内部开关放大器的采样电容使用。如果去掉外部这两个采样电容，SENSOR\_VP、SENSOR\_CAPP、SENSOR\_CAPN 及 SENSOR\_VN 也可用作普通的 ADC。

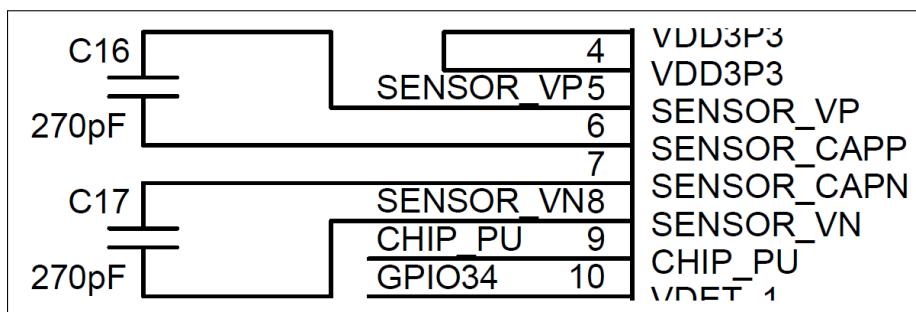


图 10: ESP32 传感器采样电容

### 3.1.7 外置阻容

ESP32 的 Pin47 CAP2 和 Pin48 CAP1 的连接电路请参见图 11。CAP1 所连的 C5, 10 nF 需要较高精度；CAP1 与 CAP2 之间的 RC 电路建议参考图中所示。若取消这部分电路，将会对芯片在 Deep-sleep 状态下产生细微影响。

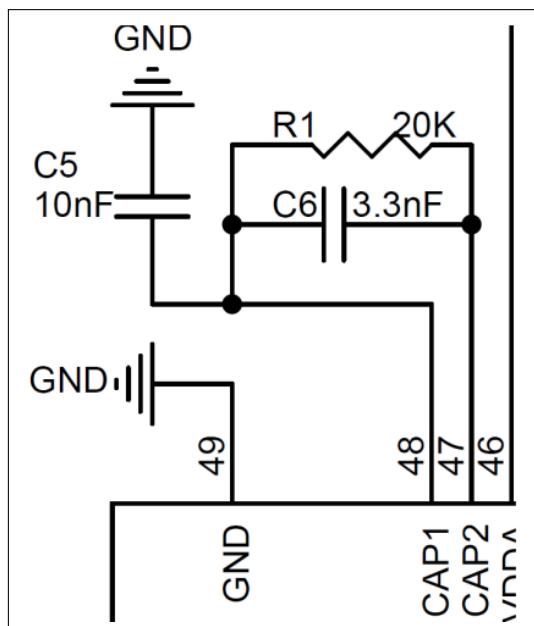


图 11: ESP32 外置电容

## 3.2 版图布局

版图布局分为以下两部分来介绍：

- 独立的 ESP32 模组版图设计
- ESP32 作为从设备版图设计

### 3.2.1 独立的 ESP32 模组的版图设计

#### 3.2.1.1 版图设计通用要点

建议采用四层板设计，即：

- 第一层为顶层，主要用于走信号线和摆件。
- 第二层为地层，不走信号线，保证一个完整的地平面。
- 第三层为电源层。为了使射频及晶振部分可以得到更好的屏蔽，因此在保证射频及晶振部分下方完整地平面的情况下，可适度走信号线。
- 第四层为底层，不建议摆件，将电源走在这层上。

### 3.2.1.2 电源

图 12 中黄色高亮信号线即为 3.3V 电源走线。电源走线的线宽必须  $> 20 \text{ mil}$ 。电源走线进入 ESP32 模拟电源管脚 (Pin1, 3, 4, 43, 46) 前需添加一个 0603 的  $10 \mu\text{F}$  电容及一个 0402 的  $0.1 \mu\text{F}$  的电容。如图 12 中 C13 ( $10 \mu\text{F}$ ) 放置在电源入口处，C10 靠近芯片的模拟电源管脚摆放。

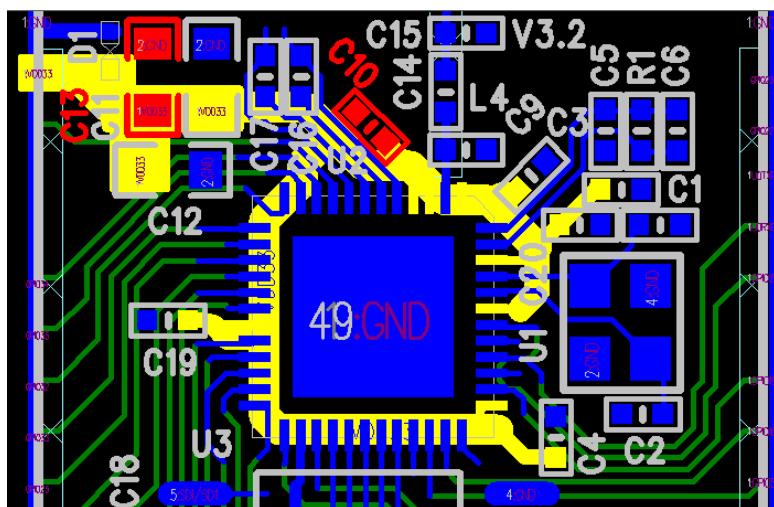


图 12: ESP32 版图设计

电源走线尽量走在第四层（底层），须通过过孔连接至顶层芯片管脚处。在过孔处理上，钻孔直径需不小于电源走线的宽度，钻孔焊盘的直径建议是钻孔直径的 1.5 倍。

### 3.2.1.3 晶振

晶振设计请参考图 13，另外：

- 晶振需离芯片时钟管脚稍远一些放置 (**gap=2.7 mm 为优**)，防止晶振干扰到芯片。同时晶振走线须用地包起来周围密集地孔屏蔽隔离。
- 晶振的时钟走线不可打孔走线，即不能跨层。晶振的时钟走线不可交叉，跨层交叉也不行。
- 晶振外接的对地调节电容请靠近晶振左右两侧摆放，并尽量置于时钟走线连接末端。
- 晶振下方 4 层都不能走高频数字信号，最佳情况是晶振下方不走任何信号线，晶振顶层的铺铜区域越大越好。
- 晶振为敏感器件，晶振周围不能有磁感应器件，比如大电感等。

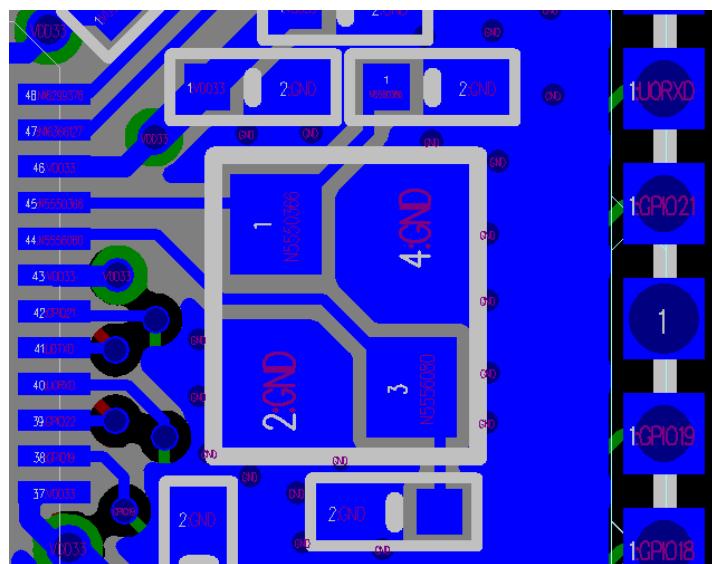


图 13: ESP32 晶振设计

### 3.2.1.4 射频

射频走线须做  $50\Omega$  单端阻抗控制，须保证相邻层完整地平面。射频走线线宽请注意保持一致，不可分支走线。射频走线长度须尽量短，并注意周围密集地孔屏蔽。

射频走线不可有过孔，即不能跨层走线，且尽量使用  $135^\circ$  角走线或是圆弧走线。

射频走线上需预留一个  $\pi$  型匹配电路，且  $\pi$  型匹配电路需靠近芯片端放置。

射频走线附近不能有高频信号线。射频上的天线必须远离所有传输高频信号的器件，比如晶振，DDR，一些高频时钟（SDIO\_CLK）等。

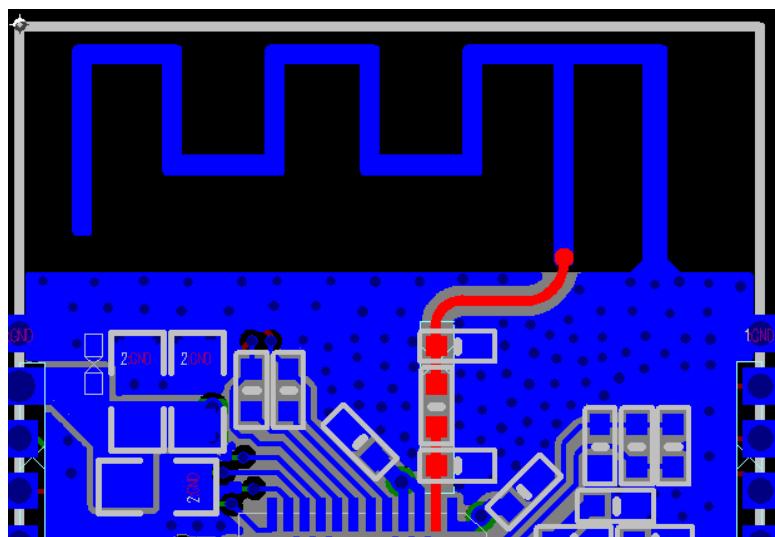


图 14: ESP32 射频部分版图设计

### 3.2.2 ESP32 作为从设备的版图设计

ESP32 作为从设备，搭载其他主 CPU 应用时，在布局设计上比模组设计更加要注意的是信号完整性。系统的复杂性，高频信号的增多对于 ESP32 的干扰也增多，需要注意避免这些干扰影响 ESP32 正常工作。这里我们以 PAD 或 TV Box 的主板作为示例进行分析。

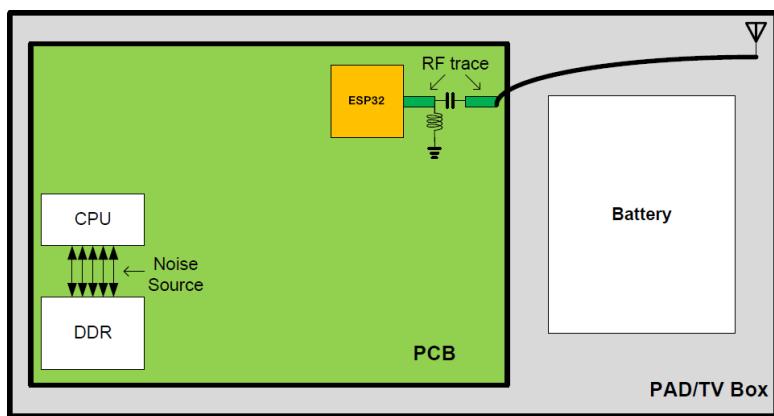


图 15: PAD/TV Box 平面位置规划框架

CPU 和 DDR 之间的数字信号是高频噪声的主要制造者，会从空中干扰 Wi-Fi 射频，所以在做系统设计时，应注意以下几点：

- 从图 15 中可以看出，ESP32 需在 PCB 边上，远离 CPU 与 DDR，即远离噪声源。从位置上增加距离使得干扰能量随距离增加而衰减，继而减小噪声耦合。
- ESP32 与 CPU 之间通过 SDIO 通信时，最好在 SDIO 的 6 根信号线上串联一个  $200\Omega$  的电阻，减小驱动电流，继而减小干扰，同时也可以消除 SDIO 走线长度不一致引起的时序问题。
- 在天线选型上，尽量不使用 PCB 板载天线，因为 PCB 板载天线受到的干扰比较大，容易把干扰耦合进来影响射频性能。最好使用外置天线，可以通过电缆线引出远离 PCB 板，这样板上高频干扰信号对 Wi-Fi 的影响会减弱。
- 在布线过程中，应注意 CPU 与 MEM 之间的高频信号走线控制，严格按照高频信号处理规则走线（详情可参考 DDR 走线控制文档），尽量做到把 CLK 单独包地处理，data 或 addr 线成组进行包地处理。
- 如果在系统设计中牵涉到电机等高功率器件，则务必要把 Wi-Fi 部分电路的返回路径（GND）与其他高功率器件的返回路径（GND）分离开来，通过导线把 2 个返回路径（GND）连接起来。
- 天线要尽量远离其他高频噪声源，如 LCD、HDMI、摄像传感器、USB 以及其他高频信号。

### 3.2.3 版图设计常见问题

#### 3.2.3.1 为什么电源纹波并不大，但射频的 Tx 性能很差？

##### 现象分析

电源纹波可极大地影响射频的 Tx 性能。测量电源纹波时需注意电源纹波必须是在 ESP32 正常发包下测试。随着不同模式下功率的改变，电源纹波也会随之变化，发包功率越高，导致的纹波越大。

一般情况下，发送 11n MCS7 的包时，电源纹波必须  $< 100 \text{ mV}$ 。发送 11b/11m 时，电源纹波必须  $< 120 \text{ mV}$ 。

##### 解决方法

在电源支路上（支路为 ESP32 模拟电源管脚）添加一个  $10 \mu\text{F}$  的滤波电容。 $10 \mu\text{F}$  的电容必须靠近芯片的模拟电源管脚，越近纹波会越小越稳定。

### 3.2.3.2 为什么芯片发包时，电源纹波很小，但射频的 Tx 性能不好？

#### 现象分析

射频的 Tx 性能不仅受电源纹的影响，还受到晶振的影响。晶振的本身质量不好，频偏过大会影响射频的 Tx 性能。或者晶振受到高频信号干扰，比如晶振的输入输出信号线走线跨层交叉，使得晶振的输入信号耦合到输出信号上，输出信号耦合到输入信号上，也会影响射频的 Tx 性能。另外，如果晶振的下方有其他高频信号走线，比如 SDIO 走线，UART 走线等，也会导致晶振无法正常工作。最后，晶振旁边有感性器件或辐射器件，比如大电感、天线等也会导致芯片的射频性能不好。

#### 解决方法

此问题主要是在布局上，可以重新布局，详见章节 [3.2 版图布局](#)。

### 3.2.3.3 为什么 ESP32 发包时，仪器测试到的 power 值比 target power 值要高或者低很多，且 EVM 比较差？

#### 现象分析

当仪器检测到的 power 值与 target power 相差较大，可能是由于芯片射频管脚输出到天线这一段传输线上阻抗不匹配导致信号在传输过程中有反射。其次，阻抗不匹配会影响到芯片内部 PA 的工作状态，使得 PA 非正常过早进入饱和区域，继而使得信号失真度高，EVM 自然会变差。

#### 解决方法

射频走线上预留了一个  $\pi$  型电路，可以根据需求对天线进行阻抗匹配，使得从芯片射频管脚往天线端看去，阻抗接近芯片端口阻抗。

### 3.2.3.4 为什么芯片的 Tx 性能没有问题，但 Rx 的灵敏度不好？

#### 现象分析

芯片的 Tx 性能没有问题意味着射频端的阻抗匹配也没有问题。Rx 灵敏度不好的可能原因是外界干扰耦合到天线上，比如晶振离天线非常近，或是 UART 的 Tx 与 Rx 走线穿过射频走线等。如果 ESP32 是作为从设备设计，那么主板上将存在非常多的高频信号干扰源，需根据主板设计来考量信号完整性的问题。

#### 解决方法

请确保天线远离晶振，且射频走线附近不要走高频信号，具体可参考 [3.2 版图布局](#)。

## 4. 开发硬件介绍

### 4.1 ESP-WROOM-32 模组

#### 4.1.1 概述

乐鑫为客户提供贴片式模组 ESP-WROOM-32。该模组的射频性能已调试到最佳状态。初期使用 ESP32 进行测试或二次开发时，建议采购我司提供的模组。该贴片式模组外形如图 16 所示。

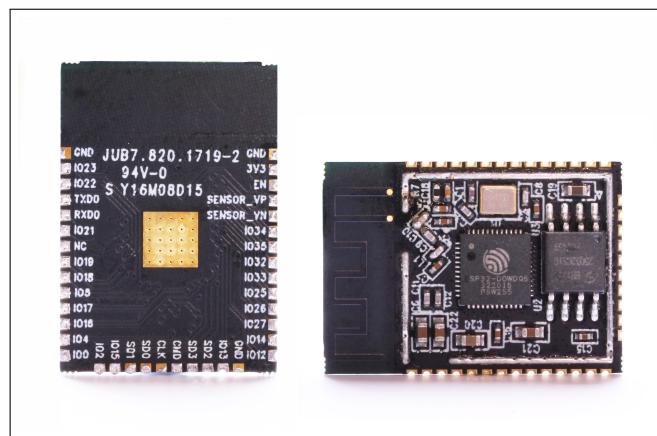


图 16: ESP-WROOM-32 模组外观图

ESP-WROOM-32 尺寸为 18 × 25.5 毫米，使用封装大小为 SOP8-208 mil 的 Flash，2 dBi 的 PCB 板载天线。ESP-WROOM-32 的尺寸图如下：

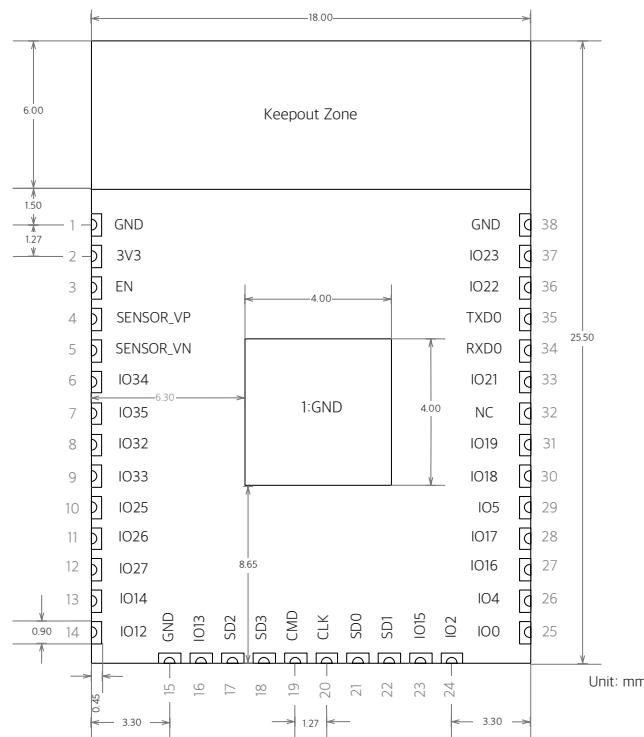


图 17: ESP-WROOM-32 模组外观尺寸图

### 4.1.2 管脚说明

该贴片式模组已引出 38 个管脚，管脚说明如表 3 所示。

表 3: WROOM-32 管脚说明

序号	名称	功能
1	GND	接地
2	3V3	供电
3	EN	使能芯片信号。高电平有效。
4	SENSOR_VP	GPI36, SENSOR_VP, ADC_H, ADC1_CH0, RTC_GPIO0
5	SENSOR_VN	GPI39, SENSOR_VN, ADC1_CH3, ADC_H, RTC_GPIO3
6	IO34	GPI34, ADC1_CH6, RTC_GPIO4
7	IO35	GPI35, ADC1_CH7, RTC_GPIO5
8	IO32	GPIO32, XTAL_32K_P (32.768 kHz crystal oscillator input), ADC1_CH4, TOUCH9, RTC_GPIO9
9	IO33	GPIO33, XTAL_32K_N (32.768 kHz crystal oscillator output), ADC1_CH5, TOUCH8, RTC_GPIO8
10	IO25	GPIO25, DAC_1, ADC2_CH8, RTC_GPIO6, EMAC_RXD0
11	IO26	GPIO26, DAC_2, ADC2_CH9, RTC_GPIO7, EMAC_RXD1
12	IO27	GPIO27, ADC2_CH7, TOUCH7, RTC_GPIO17, EMAC_RX_DV
13	IO14	GPIO14, ADC2_CH6, TOUCH6, RTC_GPIO16, MTMS, HSPICLK, HS2_CLK, SD_CLK, EMAC_TXD2
14	IO12	GPIO12, ADC2_CH5, TOUCH5, RTC_GPIO15, MTDI, HSPIQ, HS2_DATA2, SD_DATA2, EMAC_TXD3
15	GND	接地
16	IO13	GPIO13, ADC2_CH4, TOUCH4, RTC_GPIO14, MTCK, HSPID, HS2_DATA3, SD_DATA3, EMAC_RX_ER
17	SHD/SD2	GPIO9, SD_DATA2, SPIHD, HS1_DATA2, U1RXD
18	SWP/SD3	GPIO10, SD_DATA3, SPIWP, HS1_DATA3, U1TXD
19	SCS/CMD	GPIO11, SD_CMD, SPICS0, HS1_CMD, U1RTS
20	SCK/CLK	GPIO6, SD_CLK, SPICLK, HS1_CLK, U1CTS
21	SDO/SD0	GPIO7, SD_DATA0, SPIQ, HS1_DATA0, U2RTS
22	SDI/SD1	GPIO8, SD_DATA1, SPID, HS1_DATA1, U2CTS
23	IO15	GPIO15, ADC2_CH3, TOUCH3, MTDO, HSPICS0, RTC_GPIO13, HS2_CMD, SD_CMD, EMAC_RXD3
24	IO2	GPIO2, ADC2_CH2, TOUCH2, RTC_GPIO12, HSPIWP, HS2_DATA0, SD_DATA0
25	IO0	GPIO0, ADC2_CH1, TOUCH1, RTC_GPIO11, CLK_OUT1, EMAC_TX_CLK
26	IO4	GPIO4, ADC2_CH0, TOUCH0, RTC_GPIO10, HSPIHD, HS2_DATA1, SD_DATA1, EMAC_TX_ER
27	IO16	GPIO16, HS1_DATA4, U2RXD, EMAC_CLK_OUT
28	IO17	GPIO17, HS1_DATA5, U2TXD, EMAC_CLK_OUT_180
29	IO5	GPIO5, VSPICS0, HS1_DATA6, EMAC_RX_CLK
30	IO18	GPIO18, VSPICLK, HS1_DATA7
31	IO19	GPIO19, VSPIQ, U0CTS, EMAC_TXD0

序号	名称	功能
32	NC	-
33	IO21	GPIO21, VSPIHD, EMAC_TX_EN
34	RXD0	GPIO3, U0RXD, CLK_OUT2
35	TXD0	GPIO1, U0TXD, CLK_OUT3, EMAC_RXD2
36	IO22	GPIO22, VSPIWP, U0RTS, EMAC_TXD1
37	IO23	GPIO23, VSPID, HS1_STROBE
38	GND	接地

### 4.1.3 使用注意事项

- 贴片模组使用单一管脚供电，客户只需外接 1 个 3.3V 的电源即可。该 3.3V 电源既可为模拟电路供电，也可为数字电路供电。
- EN 管脚为 Wi-Fi 使能管脚，正常工作时需把 EN 管脚置高电平。
- 贴片模组提供两种工作模式，一种是 UART 下载模式，另一种是 Flash 启动模式。其中，UART 下载模式下，用户可以通过烧录工具选择把程序烧录至 Flash 或 Memory 中。当程序烧录在 Memory 中时，程序只能在本次上电期间运行，断电后 Memory 将会自动清空。当程序烧录在 Flash 中时，程序会被保存在 Flash 中，可随时调用。
- 将 GND、RXD、TXD 接出外接 USB 转 UART 工具下载、打印 log 以及通信。

出厂模式下的模组 Flash 已下载初始固件。如需自行重新烧录固件，烧录 Flash 的操作步骤如下：

1. 烧录前，需要设置模组在 UART 下载模式下工作；
2. 将 IO0 和 IO2 下拉到低电平；
3. 上电。通过串口查看是否进入 Download 模式。
4. 通过 ESP Flash 下载工具将程序固件烧录进 Flash 中；
5. 烧录结束后，IO5 需保持高电平，IO0 需上拉切换至高电平，进入 SPI 启动模式下工作；
6. 重新上电，芯片初始化时会从 Flash 中读取程序运行。

#### 注意：

- 整个操作过程都可通过 UART 打印 log 信息查看芯片运行是否正常。当无法烧录或程序无法运行时，可通过串口打印信息查看芯片初始化时设置的工作模式是否正常。
- 串口打印工具和烧录工具不能同时把串口端口打开。
- 请至乐鑫官网下载最新版 ESP Flash 下载工具：[Flash 下载工具](#)。

## 4.2 ESP32-DevKitC (开发板)

### 4.2.1 概述

乐鑫提供 ESP32-DevKitC 开发板，以供客户快速评估、熟悉产品，便于进行二次应用开发。ESP32-DevKitC 的外观尺寸如图 18 所示。

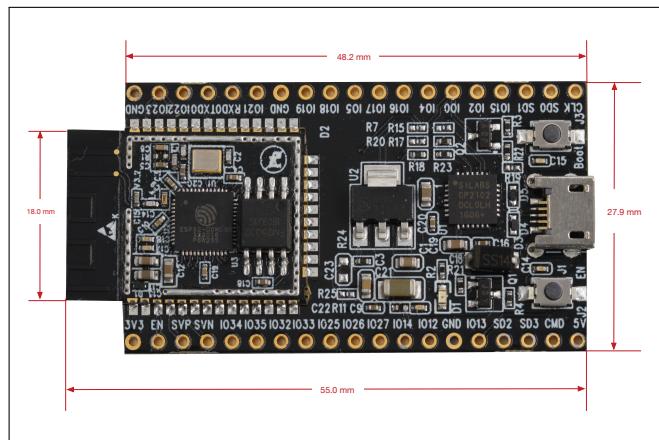


图 18: ESP32-DevKitC 尺寸图

该开发板将 ESP-WROOM-32 模组的管脚全部引出至排针，具备 ESP32 芯片所支持的所有功能，用户可以根据自己的需求使用跳线连接至外部设备。ESP32-DevKitC 的接口如图 19 所示。更多关于开发板的管脚的描述，请参考章节 4.1.2 ESP-WROOM-32 管脚说明。

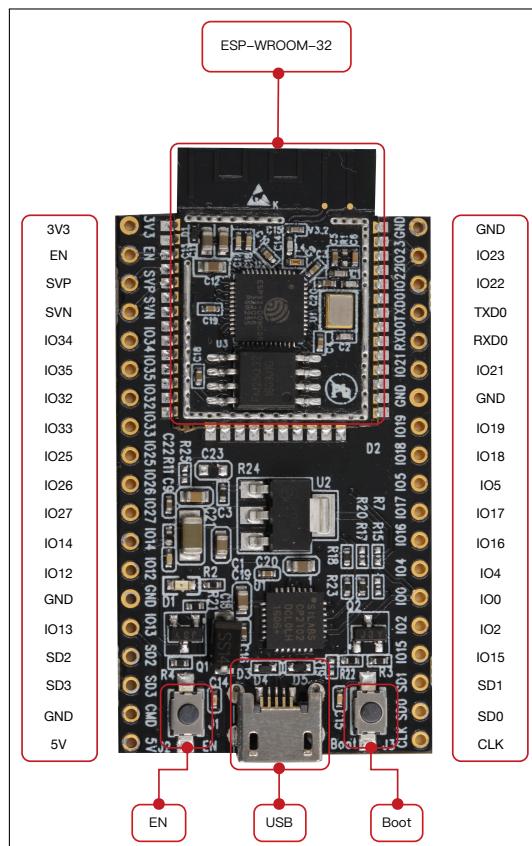


图 19: ESP32-DevKitC 布局图

## 4.2.2 原理图

### 4.2.2.1 电源原理图

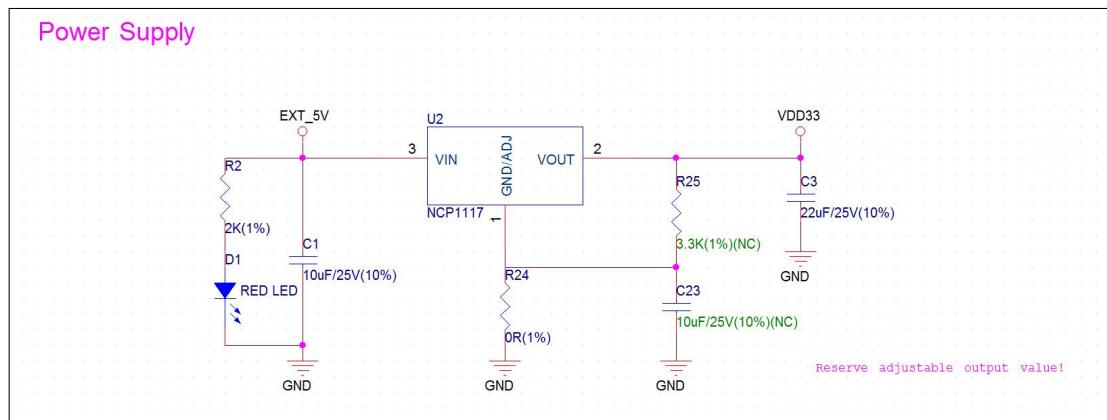


图 20: ESP32-DevKitC 电源原理图

### 4.2.2.2 USB 转串口原理图

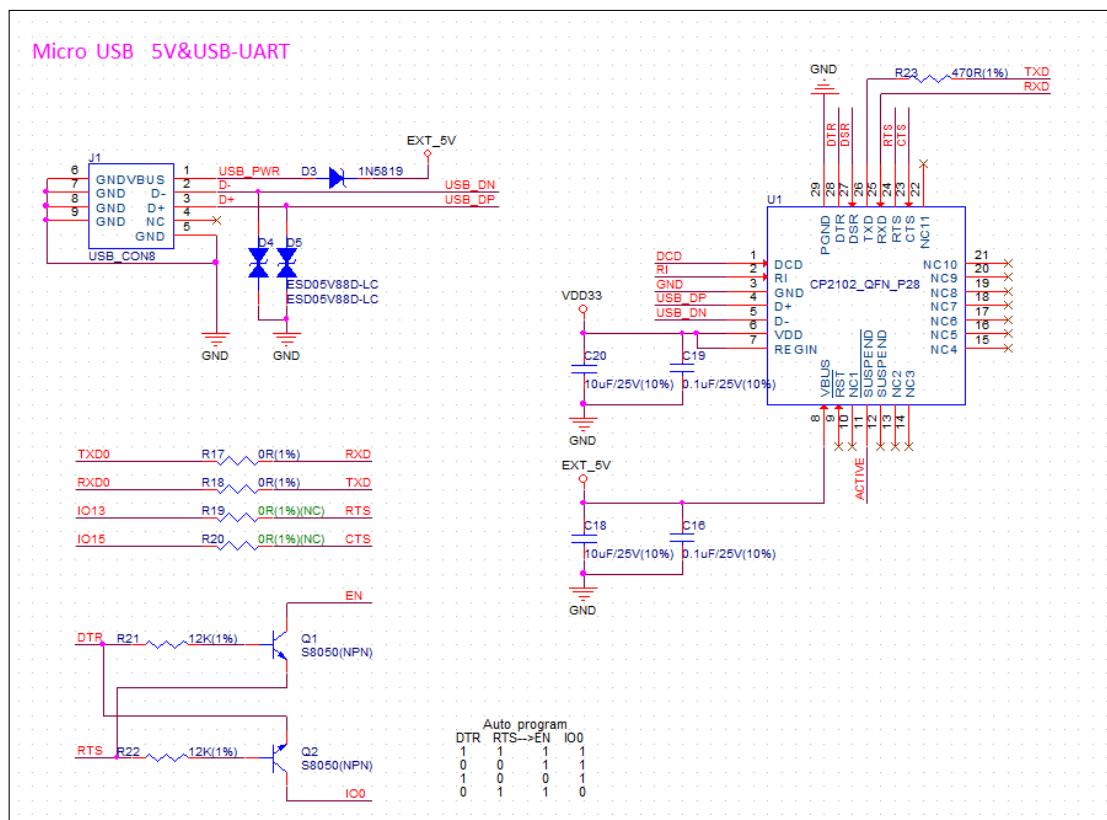


图 21: USB 转串口原理图

## 5. 典型应用案例

### 5.1 UART 串口转 Wi-Fi 智能硬件

基于目前定义的 2 个 UART 接口的管脚定义如表 4 所示。

表 4: UART 接口的管脚定义

类别	管脚定义	功能
UART0	(Pin34) U0RXD + (Pin35) U0TXD	可用来打印 log 信息
UART1	(Pin25) U1RXD + (Pin27) U1TXD	可用来收发命令

应用案例：ESP32 开发板（详见章节 4.2 ESP32-DevKitC）。

### 5.2 ESP32-Lyra (智能音频平台)

ESP32-Lyra 是乐鑫着眼物联网、倾力打造的高性价比智能音频平台。基于 ESP32 双核处理器、Wi-Fi + BT，具有语音识别、音频播放、云端服务等功能。ESP32-Lyra 平台可应用于无线音响系统、早教机系统、机器人语音系统、智能家居系统等。

ESP32-Lyra 智能音频平台具有以下特性：

- 音频接口丰富、扩展性强
- 支持触摸按键
- 支持多种音频格式：WMA、ALAC、AAC、FLAC、OPUS、MP3、WAV、OGG
- 支持多种无线音频标准：DLNA、AirPlay、QPlay
- 支持多种云平台：喜马拉雅、YunOS、亚马逊
- 支持多种配网协议：ESP-TOUCH、ALINK、JoyLink3.0、AirKiss