

战舰 V4 硬件参考手册

V1.0

—正点原子战舰 STM32F103 开发板教程

修订历史:

版本	日期	修改内容
V1.0	2021/06/20	第一次发布



正点原子公司名称：广州市星翼电子科技有限公司

原子哥在线教学平台：www.yuanzige.com

开源电子网 / 论坛：www.openedv.com

正点原子官方网站：www.alientek.com

正点原子淘宝店铺：<https://openedv.taobao.com>

正点原子 B 站视频：<https://space.bilibili.com/394620890>

电话：020-38271790 传真：020-36773971

请下载原子哥 APP，数千讲视频免费学习，更快更流畅。

请关注正点原子公众号，资料发布更新我们会通知。



扫码关注正点原子公众号



扫码下载“原子哥”APP

内容简介	6
第一章 实验平台简介	7
1.1 战舰 V4 开发板资源初探	7
1.1.1 战舰 V4 硬件设计特点	7
1.1.2 战舰 V4 硬件基本参数	7
1.1.3 战舰 V4 硬件资源分布	8
1.1.4 战舰 V4 硬件资源列表	8
1.2 战舰 V4 开发板资源说明	10
1.2.1 硬件资源说明	10
1.2.2 战舰 V4 IO 引脚分配	15
1.3 战舰 V4 升级说明	19
第二章 实验平台硬件资源详解	20
2.1 开发板原理图详解	20
2.1.1 MCU	20
2.1.2 引出 IO 口	22
2.1.3 USB 串口/串口 1 选择接口	23
2.1.4 JTAG/SWD	24
2.1.5 SRAM	24
2.1.6 LCD 模块接口	25
2.1.7 复位电路	26
2.1.8 启动模式设置接口	26
2.1.9 RS232 串口/JOYPAD 接口	27
2.1.10 RS485 接口	27
2.1.11 CAN/USB 接口	28
2.1.12 EEPROM	28
2.1.13 光敏传感器	29
2.1.14 SPI FLASH	29
2.1.15 温湿度传感器接口	29
2.1.16 红外接收头	30
2.1.17 无线模块接口	30
2.1.18 LED	31
2.1.19 按键	31
2.1.20 TPAD 电容触摸按键	31

2.1.21 OLED/摄像头模块接口	32
2.1.22 有源蜂鸣器	32
2.1.23 TF 卡接口	33
2.1.24 ATK 模块接口	33
2.1.25 多功能端口	34
2.1.26 以太网接口 (RJ45)	35
2.1.27 板载喇叭	37
2.1.28 音频编解码	37
2.1.29 电源	38
2.1.30 电源输入输出接口	39
2.1.31 USB 串口	40
2.2 开发板使用注意事项	40

内容简介

本手册主要介绍战舰 V4 的硬件资源，包括：实验平台简介、实验平台硬件资源详解以及使用注意事项等。通过本手册的学习，大家将会对战舰 V4 开发板的硬件有一个比较全面的了解，对后续的软件学习及程序设计非常有帮助。

本手册是《STM32F103 开发指南》的重要补充教程，强烈建议大家在学习相关例程前，先学习本手册！

第一章 实验平台简介

本章主要介绍我们的实验平台：正点原子战舰 V4 STM32F103 开发板。通过本章的学习，您将对我们后面使用的实验平台有个大概了解，为后面的学习做铺垫。

本章将分为如下两节：

- 1.1，战舰 V4 开发板资源初探；
- 1.2，战舰 V4 开发板资源说明；

1.1 战舰 V4 开发板资源初探

自从 2012 年上市以来，正点原子战舰 STM32F103 开发板广受客户好评，并常年稳居淘宝 STM32 系列开发板销量冠军，总销量超过 5W 套。最新的战舰 STM32F103 V4 开发板，则是根据广大客户反馈，在战舰 V3 的基础上进行改进而来（具体改变见 1.3 节），下面我们开始介绍战舰 V4。

1.1.1 战舰 V4 硬件设计特点

战舰 V4 STM32F103 开发板硬件设计特点包括：

- 1) **接口丰富**。板子提供十来种标准接口，可以方便的进行各种外设的实验和开发。
- 2) **设计灵活**。板上很多资源都可以灵活配置，以满足不同条件下的使用。我们引出了除晶振占用的 IO 口外的所有 IO 口，可以极大的方便大家扩展及使用。另外板载一键下载功能，可避免频繁设置 B0、B1 的麻烦，仅通过 1 根 USB 线即可实现 STM32 的开发。
- 3) **资源充足**。主芯片采用自带 512K 字节 FLASH 的 STM32F103ZET6，并外扩 1M 字节 SRAM 和 16M 字节 FLASH，满足大内存需求和大数据存储。板载高性能音频编解码芯片、双 RS232 串口、百兆网卡、光敏传感器以及各种接口芯片，满足各种应用需求。
- 4) **人性化设计**。各个接口都有丝印标注，且用方框框出，使用起来一目了然；部分常用外设大丝印标出，方便查找；接口位置设计合理，方便顺手。资源搭配合理，物尽其用。
- 5) **国产化程度高**。为了支持国产芯片的发展和推广，正点原子优选国产好芯，战舰 V4 开发板上凡是能用国产替代的芯片，全部使用国产芯片，国产化率达到 85%（数量）。

1.1.2 战舰 V4 硬件基本参数

战舰 V4 硬件基本参数如表 1.1.2.1 所示：

项目	说明
产品型号	ATK- DNF103P V4
CPU	STM32F103ZET6, LQFP144
引出 IO	110 个
外形尺寸	121mm*160mm
工作电压	5V（USB）、DC6V~15V（DC005）
工作电流	120mA~160mA ¹ （@5V）
工作温度	0℃~+70℃

表 1.1.2.1 战舰 V4 硬件基本参数

注 1：120mA 对应 CPU 在复位情况下，裸板的工作电流；160mA 对应 CPU 正常运行时裸板的工作电流。

1.1.3 战舰 V4 硬件资源分布

战舰 V4 的硬件资源分布如图 1.1.3.1 所示：

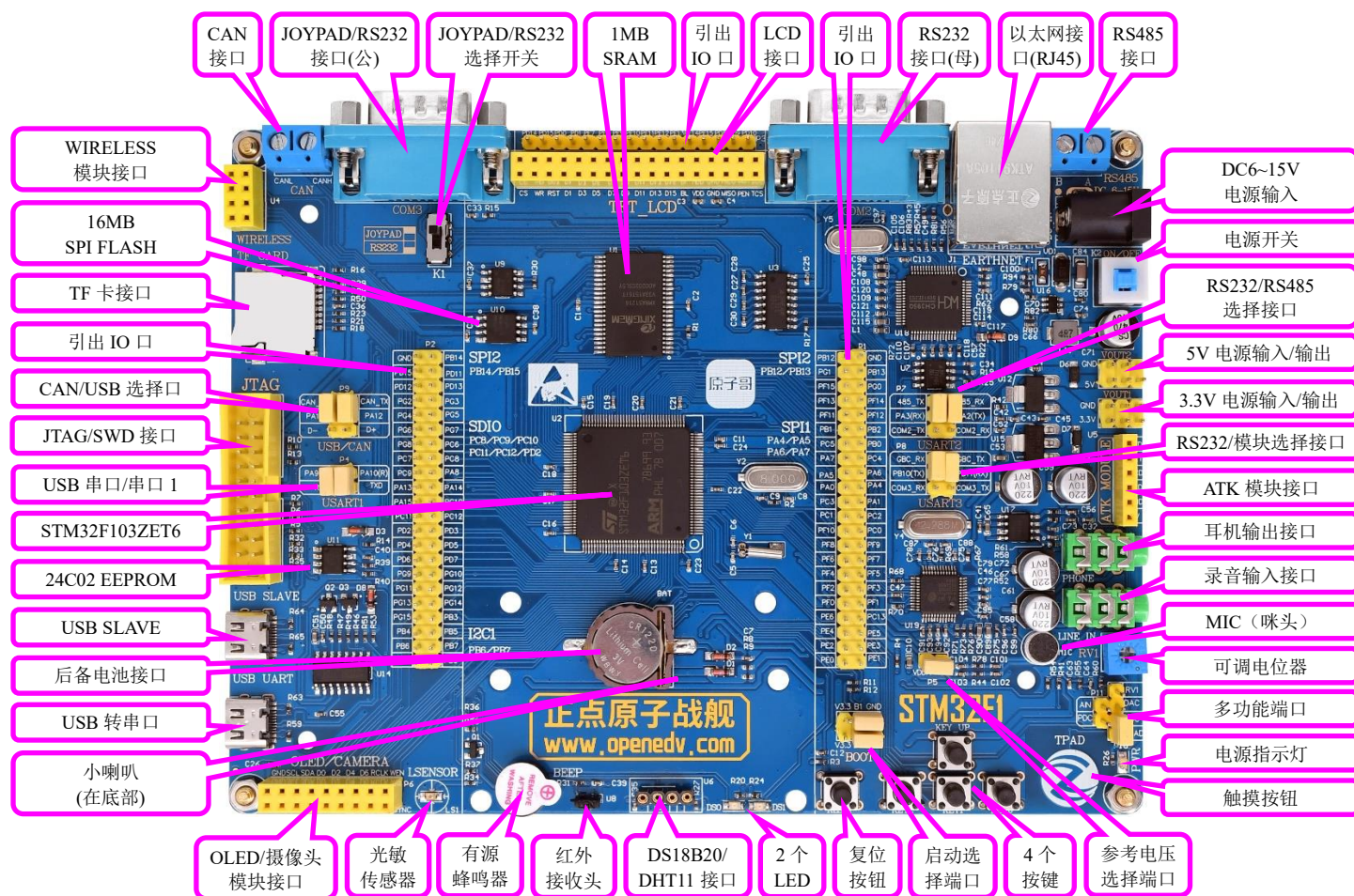


图 1.1.3.1 战舰 V4 硬件资源分布图

1.1.4 战舰 V4 硬件资源列表

战舰 V4 的硬件资源列表如表 1.1.4.1 所示：

资源	数量	说明
CPU	1 个	STM32F103ZET6; FLASH: 512KB; SRAM: 64KB;
SRAM	1 个	1MB (B: 字节, 下同) , 16 位并口访问
SPI FLASH	1 个	16MB
EEPROM	1 个	2Kb (256B)
电源指示灯	1 个	蓝色
状态指示灯	2 个	红色 (DS0); 绿色 (DS1);
复位按键	1 个	用于 MCU&LCD 的复位
功能按键	4 个	KEY0、KEY1、KEY2、KEY_UP (具备唤醒功能)
电容触摸按键	1 个	TPAD, 用于电容触摸按键
电源开关	1 个	控制整个板子供电
可调电位器	1 个	用于设置 RV1 的电源, 方便 ADC 实验测试

蜂鸣器	1 个	有源蜂鸣器，用于发出提示音
板载扬声器	1 个	在开发板背面，用于播放音乐
红外接收头	1 个	用于红外接收，配备红外遥控器
光敏传感器	1 个	用于感应环境光照强度
音频编解码芯片	1 个	VS1053，用于 WAV、MP3、OGG、FLAC 等音频编解码
无线模块接口	1 个	可以接 NRF24L01 等无线模块
CAN 接口	1 个	用于 CAN 通信，带 120R 终端电阻
RS485 接口	1 个	用于 RS485 通信，带 120R 终端电阻
RS232 接口	2 个	用于 RS232 通信，提供一公一母两路 RS232 接口
游戏手柄接口	1 个	用于接 FC 游戏手柄，和公头 RS232 接口共用一个 DB9
数字温湿度传感器接口	1 个	支持 DS18B20、DHT11 等数字温湿度传感器
ATK 模块接口	1 个	支持正点原子各种模块产品（蓝牙/GPS/MPU6050 等）
LCD 接口	1 个	支持正点原子 2.8/3.5/4.3/7 寸等多种 TFTLCD 模块
摄像头接口	1 个	和 OLED 共用一个接口，支持正点原子各种摄像头模块
OLED 模块接口	1 个	和摄像头模块接口共用，支持正点原子各种 OLED 模块
USB 转串口	1 个	用于 USB 转 TTL 串口通信
USB 从机接口	1 个	用于 USB SLAVE（从机）通信
RS232/手柄选择开关	1 个	用于选择 COM3 DB9 口接 RS232 还是 FC 游戏手柄
RS232/485 选择接口	1 个	用于选择 PA2/PA3 做 RS232 还是 RS485 通信
RS232/模块选择接口	1 个	用于选择 PB10/PB11 做 RS232 还是 ATK Module 接口
CAN/USB 选择接口	1 个	用于选择 PA11/PA12 做 CAN 还是 USB 通信
TF 卡接口	1 个	用于接 TF 卡
10M/100M 网口	1 个	用于以太网通信
JTAG/SWD 调试口	1 个	用于仿真调试、下载代码等
录音咪头（MIC）	1 个	用于录音
立体声音频输出接口	1 个	用于外接耳机、功放等
立体声录音输入接口	1 个	用于外接音源输入
多功能接口	1 组	用于 DAC/ADC/PWM DAC/AUDIO IN/TPAD 等互联
5V 电源输入/输出口	1 组	用于 5V 电源接入/对外提供 5V 电压
3.3V 电源输入/输出口	1 组	用于 3.3V 电源接入/对外提供 3.3V 电压
参考电压设置接口	1 个	用于选择 ADC 参考电压
直流电源输入接口	1 个	支持 DC6V~15V 直流电源输入，采用 DC005 接口
启动模式选择配置接口	1 个	用于设置 STM32 启动模式
后备电池接口	1 个	用于 RTC 后备电池
引出 IO	110	除 RTC 晶振占用的 2 个 IO 口外，其他 IO 口全引出
一键下载电路	1 个	正点原子专利电路，方便使用串口下载代码

表 1.1.4.1 战舰 V4 的硬件资源列表

1.2 战舰 V4 开发板资源说明

战舰 V4 资源说明，我们将分为两个部分：硬件资源说明和战舰 V4 IO 引脚分配。

1.2.1 硬件资源说明

这里我们详细介绍战舰 STM32F103 的各个部分（图 1.1.3.1 中的标注部分）的硬件资源，我们将按逆时针的顺序依次介绍。

1. WIRELESS 模块接口

这是开发板板载的无线模块接口（U4），可以外接 NRF24L01/RFID 等无线模块。从而实现无线通信等功能。注意：接 NRF24L01 模块进行无线通信的时候，必须同时有 2 个模块和 2 个板子，才可以测试，单个模块/板子例程是不能测试的。

2. 16MB SPI FLASH

这是开发板外扩的 SPI FLASH 芯片（U10），容量为 128Mbit，也就是 16M 字节，可用于存储字库和其他用户数据，满足大容量数据存储要求。当然如果觉得 16M 字节还不够用，你可以把数据存放在外部 TF 卡。

3. TF 卡接口

这是开发板板载的一个 TF 卡接口（也叫 Micro SD 卡），SDIO 方式驱动，TF 卡容量选择范围非常宽（最大可达 TB 级），有了这个 TF 卡接口，就可以满足海量数据存储的需求。

4. 引出 IO 口（总共有三处）

这是开发板 IO 引出端口，总共有三组主 IO 引出口：P1、P2 和 P3。其中，P1 和 P2 分别采用 2*22 排针引出，总共引出 86 个 IO 口，P3 采用 1*16 排针，按顺序引出 FSMC_D0~D15 等 16 个 IO 口。而 STM32F103ZET6 总共只有 112 个 IO，除去 RTC 晶振占用的 2 个 IO，还剩下 110 个，前面三组主引出排针，总共引出：102 个 IO，剩下的通过：P4、P7、P8 和 P9 引出。

5. CAN/USB 选择口

这是一个 CAN/USB 的选择接口（P9），因为 STM32 的 USB 和 CAN 是共用一组 IO（PA11 和 PA12），所以我们通过跳线帽来选择不同的功能，以实现 USB/CAN 实验。

6. JTAG/SWD 接口

这是战舰开发板板载的 20 针标准 JTAG 调试口（JTAG），该 JTAG 口直接可以和 DAP、JLINK 或者 STLINK 等调试器（仿真器）连接，同时由于 STM32 支持 SWD 调试，这个 JTAG 口也可以用 SWD 模式来连接。

用标准的 JTAG 调试，需要占用 5 个 IO 口，有些时候，可能造成 IO 口不够用，而用 SWD 则只需要 2 个 IO 口，大大节约了 IO 数量，但他们达到的效果是一样的，所以我们**强烈建议仿真器使用 SWD 模式！**

7. USB 串口/串口 1

这是 USB 串口同 STM32F103ZET6 的串口 1 进行连接的接口（P4），标号 RXD 和 TXD 是 USB 转串口的 2 个数据口（对 CH340 来说），而 PA9(TXD)和 PA10(RXD)则是 STM32 的串口 1 的两个数据口（复用功能下）。他们通过跳线帽对接，就可以和连接在一起了，从而实现 STM32 的程序下载以及串口通信。

设计成 USB 串口，是出于现在电脑上串口正在消失，尤其是笔记本，几乎清一色的没有串口。所以板载了 USB 串口可以方便大家下载代码和调试。而在板子上并没有直接连接在一起，则是出于使用方便的考虑。这样设计，你可以把战舰开发板当成一个 USB 转 TTL 串口，来和其他板子通信，而其他板子的串口，也可以方便地接到战舰开发板上。

8. STM32F103ZET6

这是开发板的核心芯片 (U2)，型号为：STM32F103ZET6。该芯片具有 64KB SRAM、512KB FLASH、2 个基本定时器、4 个通用定时器、2 个高级定时器、2 个 DMA 控制器（共 12 个通道）、3 个 SPI、2 个 IIC、5 个串口、1 个 USB、1 个 CAN、3 个 12 位 ADC、1 个 12 位 DAC、1 个 SDIO 接口、1 个 FSMC 接口以及 112 个通用 IO 口。

9. 24C02 EEPROM

这是开发板板载的 EEPROM 芯片 (U11)，容量为 2Kb，也就是 256 字节。用于存储一些掉电不能丢失的重要数据，比如系统设置的一些参数/触摸屏校准数据等。有了这个就可以方便的实现掉电数据保存。

10. USB SLAVE

这是开发板板载的一个 Type C USB 头 (USB_SLAVE)，用于 USB 从机 (SLAVE) 通信，一般用于 STM32 与电脑的 USB 通信。通过此接口，开发板就可以和电脑进行 USB 通信了。

开发板总共板载了 2 个 Type C USB 头，一个 (USB_UART) 用于 USB 转串口，连接 CH340 芯片；另外一个 (USB_SLAVE) 用于 STM32 内带的 USB。同时开发板可以通过此接头供电，板载两个 Type C USB 头（不共用），主要是考虑了使用的方便性，以及可以给板子提供更大的电流（两个 USB 都接上）这两个因素。

11. 后备电池接口

这是 STM32 后备区域的供电接口 (BAT)，可安装 CR1220 电池（默认安装了），可以用来给 STM32 的后备区域提供能量，在外部电源断电的时候，维持后备区域数据的存储，以及 RTC 的运行。

12. USB 转串口

这是开发板板载的另外一个 Type C USB 头 (USB_UART)，用于 USB 连接 CH340 芯片，从而实现 USB 转 TTL 串口。同时，此接头也是开发板电源的主要提供口。

13. 小喇叭

这是开发板自带的一个 8Ω 2W 的小喇叭，安装在开发板的背面，并带了一个小音腔，可以用来播放音频。该喇叭由 HT6872 单声道 D 类功放 IC 驱动，最大输出功率可达 2W。

特别注意：HT6872 受 VS1053 的 GPIO4 控制，必须程序上控制 VS1053 的 GPIO4 输出 1，才可以控制 HT6872 工作，从而听到声音。默认条件下 (GPIO4=0) HT6872 是关闭的。

14. OLED/摄像头模块接口

这是开发板板载的一个 OLED/摄像头模块接口 (P6)，如果是 OLED 模块，靠左插即可（右边两个孔位悬空）。如果是摄像头模块（正点原子提供），则刚好插满。通过这个接口，可以分别连接 2 种外部模块，从而实现相关实验。

15. 光敏传感器

这是开发板板载的一个光敏传感器 (LS1)，通过该传感器，开发板可以感知周围环境光线的变化，从而可以实现类似自动背光控制的应用。

16. 有源蜂鸣器

这是开发板的板载蜂鸣器 (BEEP)，可以实现简单的报警/闹铃等功能。

17. 红外接收头

这是开发板的红外接收头 (U8)，可以实现红外遥控功能，通过这个接收头，可以接受市面上常见的各种遥控器的红外信号，大家甚至可以自己实现万能红外解码。当然，如果应用得当，该接收头也可以用来传输数据。

战舰开发板给大家配备了一个小巧的红外遥控器，该遥控器外观如图 1.2.1.1 所示：



图 1.2.1.1 红外遥控器

18. DS18B20/DHT11 接口

这是开发板的一个复用接口（U6），该接口由 4 个镀金排孔组成，可以用来接 DS18B20/DS1820 等数字温度传感器。也可以用来接 DHT11 这样的数字温湿度传感器。实现一个接口，2 个功能。不用的时候，大家可以拆下上面的传感器，放到其他地方去用，使用上是非常方便灵活的。

19. 2 个 LED

这是开发板板载的两个 LED 灯（DS0 和 DS1），DS0 是红色的，DS1 是绿色的，主要是方便大家识别。

我们一般的应用 2 个 LED 足够了，在调试代码的时候，使用 LED 来指示程序状态，是非常不错的一个辅助调试方法。战舰开发板几乎每个实例都使用了 LED 来指示程序的运行状态。

20. 复位按钮

这是开发板板载的复位按键（RESET），用于复位 STM32，还具有复位液晶的功能，因为液晶模块的复位引脚和 STM32 的复位引脚是连接在一起的，当按下该键的时候，STM32 和液晶一并被复位。

21. 启动选择端口

这是开发板板载的启动模式选择端口（BOOT），STM32 有 BOOT0（B0）和 BOOT1（B1）两个启动选择引脚，用于选择复位后 STM32 的启动模式，作为开发板，这两个是必须的。在开发板上，我们通过跳线帽选择 STM32 的启动模式。关于启动模式的说明，请看 2.1.8 小节。

22. 4 个按键

这是开发板板载的 4 个机械式输入按键（KEY0、KEY1、KEY2 和 KEY_UP），其中 KEY_UP 具有唤醒功能，该按键连接到 STM32 的 WAKE_UP（PA0）引脚，可用于待机模式下的唤醒，在不使用唤醒功能的时候，也可以做为普通按键输入使用。

其他 3 个是普通按键，可以用于人机交互的输入，这 3 个按键是直接连接在 STM32 的 IO 口上的。这里注意 KEY_UP 是高电平有效，而 KEY0、KEY1 和 KEY2 是低电平有效，大家在使用的时候留意一下。

23. 参考电压选择端口

这是 STM32 的参考电压选择端口（P5），我们默认是接开发板的 3.3V（VDDA）。如果大家想设置其他参考电压，只需要把你的参考电压源接到 Vref+ 和 GND 即可。

24. 触摸按钮

这是开发板板载的一个电容触摸输入按键（TPAD），利用电容充放电原理，实现触摸按键检测。

25. 电源指示灯

这是开发板板载的一颗蓝色的 LED 灯（PWR），用于指示电源状态。在电源开启的时候（通过板上的电源开关控制），该灯会亮，否则不亮。通过这个 LED，可以判断开发板的上电情况。

26. 多功能端口

这是 1 个由 6 个排针组成的一个接口（P10&P11）。不过大家可别小看这 6 个排针，这可是本开发板设计的很巧妙的一个端口（由 P10 和 P11 组成），这组端口通过组合可以实现的功能有：ADC 采集、DAC 输出、PWM DAC 输出、外部音频输入、电容触摸按键、DAC 音频、PWM DAC 音频、DAC ADC 自测等，所有这些，你只需要 1 个跳线帽的设置，就可以逐一实现。

27. 可调电位器

这是一个 3362 型可调电位器（RV1），通过它可以调节 RV1 端口电压（范围：0~3.3V），当我们用杜邦线连接 P10 的 RV1 和 ADC 后，在 ADC 实验的时候，就可以通过它调整 ADC 的输入电压，方便大家测试。

28. MIC（咪头）

这是开发板的板载录音输入接口（MIC），该咪头直接接到 VS1053 的输入上，可以用来实现录音功能。

29. 录音输入接口

这是开发板板载的外部录音输入接口（LINE_IN），通过咪头我们只能实现单声道的录音，而通过这个 LINE_IN，我们可以实现立体声录音。

30. 耳机输出接口

这是开发板板载的音频输出接口（PHONE），该接口可以插 3.5mm 的耳机，当 VS1053 放音的时候，就可以通过在该接口插入耳机，欣赏音乐。

31. ATK 模块接口

这是开发板板载的一个正点原子通用模块接口（U5），目前可以支持正点原子开发的 GPS、蓝牙、LORA、手势识别、激光测距和 MPU6050 等模块，直接插上对应的模块，就可以进行开发。后续我们将开发更多兼容该接口的其他模块，实现更强大的扩展性能。

32. RS232/模块选择接口

这是开发板板载的一个 RS232（COM3）/ATK 模块接口（U5）选择接口（P8），通过该选择接口，我们可以选择 STM32 的串口 3 连接在 COM3 还是连接在 ATK 模块接口上面，以实现不同的应用需求。这样的设计还有一个好处，就是我们的开发板还可以充当 RS232 到 TTL 串口的转换（注意，这里的 TTL 高电平是 3.3V）。

33. 3.3V 电源输入/输出

这是开发板板载的一组 3.3V 电源输入输出排针（2*3）（VOUT1），用于给外部提供 3.3V 的电源，也可以用于从外部接 3.3V 的电源给板子供电。

大家在实验的时候可能经常会为没有 3.3V 电源而苦恼不已，有了战舰开发板，你就可以很方便的拥有一个简单的 3.3V 电源（USB 供电的时候，最大电流不能超过 500mA，外部供电的时候，最大可达 1000mA）。

34. 5V 电源输入/输出

这是开发板板载的一组 5V 电源输入输出排针（2*3）（VOUT2），该排针用于给外部提供 5V 的电源，也可以用于从外部接 5V 的电源给板子供电。

同样大家在实验的时候可能经常会为没有 5V 电源而苦恼不已，正点原子充分考虑到了大

家需求，有了这组 5V 排针，你就可以很方便的拥有一个简单的 5V 电源（USB 供电的时候，最大电流不能超过 500mA，外部供电的时候，最大可达 1000mA）。

35. RS232/485 选择接口

这是开发板板载的 RS232（COM2）/485 选择接口（P7），因为 RS485 基本上就是一个半双工的串口，为了节约 IO，我们把 RS232（COM2）和 RS485 共用一个串口，通过 P7 来设置当前是使用 RS232（COM2）还是 RS485。这样的设计还有一个好处。就是我们的开发板既可以充当 RS232 到 TTL 串口的转换，又可以充当 RS485 到 TTL485 的转换。（注意，这里的 TTL 高电平是 3.3V）。

36. 电源开关

这是开发板板载的电源开关（K2）。该开关用于控制整个开发板的供电，如果切断，则整个开发板都将断电，电源指示灯（PWR）会随着此开关的状态而亮灭。

37. DC6~15V 电源输入

这是开发板板载的一个外部电源输入口（DC_IN），采用标准的直流电源插座。开发板板载了 DC-DC 芯片，用于给开发板提供高效、稳定的 5V 电源。由于采用了 DC-DC 芯片，所以开发板的供电范围比较宽，大家可以很方便的找到合适的电源（只要输出范围在 DC6~15V 的基本都可以）来给开发板供电。在耗电比较大的情况下，比如用到 4.3 屏/7 寸屏/网口的时候，建议使用外部电源供电，可以提供足够的电流给开发板使用。

38. RS485 总线接口

这是开发板板载的 RS485 总线接口（RS485），通过 2 个端口和外部 485 设备连接。这里提醒大家，RS485 通信的时候，必须 A 接 A，B 接 B。否则可能通信不正常！另外，开发板自带了终端电阻（120Ω）。

39. 以太网接口（RJ45）

这是开发板板载的网口（EARTHNET），可以用来连接网线，实现网络通信功能。该接口使用 CH395Q 作为网络芯片，该芯片自带协议栈，支持 10M/100M 网络，通过 SPI 接口连接。

40. RS232 接口（母）

这是开发板板载的一个 RS232 接口（COM2），通过一个标准的 DB9 母头和外部的串口连接。通过这个接口，我们可以连接带有串口的电脑或者其他设备，实现串口通信。

41. LCD 接口

这是开发板板载的 LCD 模块接口，该接口兼容正点原子全系列 TFTLCD 模块（MCU 屏），包括：2.8 寸、3.5 寸、4.3 寸和 7 寸等 TFTLCD 模块，并且支持电阻/电容触摸功能。

42. 1MB SRAM

这是开发板外扩的 SRAM 芯（U1）片，容量为 8M 位，也就是 1M 字节，这样，对大内存需求的应用（比如 GUI），就可以很好的实现了

43. JOYPAD/RS232 选择开关

这是开发板板载的一个游戏手柄接口（JOYPAD）和 RS232 接口选择开关（K1），开发板的游戏手柄接口和 RS232 接口共用 COM3，它们需要分时复用。当插游戏手柄时，K1 需要打在 JOYPAD 位置，此时，该接口（COM3）可以用来连接 FC 手柄（红白机/小霸王游戏机手柄），这样大家可以在开发板上编写游戏程序，直接通过手柄玩游戏。当作为串口使用时，K1 需要打在 RS232 位置。

44. JOYPAD/RS232 接口（公）

这是开发板板载的一个游戏手柄/RS232 接口（COM3），通过一个标准的 DB9 公头和外部的 FC 手柄/RS232 串口连接。具体用作接游戏手柄接口还是 RS232 接口，可通过 K1 开关进行选择。

45. CAN 接口

这是开发板板载的 CAN 总线接口 (CAN)，通过 2 个端口和外部 CAN 总线连接，即 CANH 和 CANL。这里提醒大家：CAN 通信的时候，必须 CANH 接 CANH，CANL 接 CANL，否则可能通信不正常！这里，开发板也是自带了终端电阻 (120Ω)。

1.2.2 战舰 V4 IO 引脚分配

为了让大家更快更好的使用我们的战舰 V4 开发板，这里特地将战舰 V4 开发板主芯片：STM32F103ZET6 的 IO 资源分配做了一个总表，以便大家查阅。战舰 V4 的 IO 引脚分配总表如表：1.2.2.1 所示：

战舰 V4 IO 资源分配表					
引脚编号	GPIO	连接资源		完全独立	连接关系说明
34	PA0	WK_UP		Y	1, 按键 KEY_UP 2, 可以做待机唤醒脚(WKUP)
35	PA1	STM_ADC	TPAD	Y	ADC 输入引脚，同时做 TPAD 检测脚
36	PA2	USART2_TX	485_RX	Y	1, RS232 串口 2(COM2)RX 脚(P7 设置) 2, RS485 RX 脚(P7 设置)
37	PA3	USART2_RX	485_TX	Y	1, RS232 串口 2(COM2)TX 脚(P7 设置) 2, RS485 TX 脚(P7 设置)
40	PA4	STM_DAC	GBC_KEY	Y	1, DAC_OUT1 输出脚 2, ATK-MODULE 接口的 KEY 引脚
41	PA5	SPI1_SCK	VS_SCK CH395Q_SCK	N	SPI1 的 SCK 脚，连接 VS1053/CH395Q 的 SCK
42	PA6	SPI1_MISO	VS_MISO CH395Q_MISO	N	SPI1 的 MISO 脚，连接 VS1053/CH395Q 的 MISO
43	PA7	SPI1_MOSI	VS_MOSI CH395Q_MOSI	N	SPI1 的 MOSI 脚，连接 VS1053/CH395Q 的 MOSI
100	PA8	OV_VSYNC	PWM_DAC	N	1, OLED/CAMERA 接口的 VSYNC 脚 2, PWM_DAC 输出脚
101	PA9	USART1_TX		Y	串口 1 TX 脚，默认连接 CH340 的 RX(P4 设置)
102	PA10	USART1_RX		Y	串口 1 RX 脚，默认连接 CH340 的 TX(P4 设置)
103	PA11	USB_D-	CAN_RX	Y	1, USB D-引脚(P9 设置) 2, CAN_RX 引脚(P9 设置)
104	PA12	USB_D+	CAN_TX	Y	1, USB D+引脚(P9 设置) 2, CAN_TX 引脚(P9 设置)
105	PA13	JTMS	SWDIO	N	JTAG/SWD 仿真接口, 没接任何外设 注意：如要做普通 IO，需先禁止 JTAG&SWD
109	PA14	JTCK	SWDCLK	N	JTAG/SWD 仿真接口, 没接任何外设 注意：如要做普通 IO，需先禁止 JTAG&SWD
110	PA15	JTDI	GBC_LED	N	1, JTAG 仿真口(JTDI) 2, ATK-MODULE 接口的 LED 引脚(使用时，需先禁止 JTAG，才可以当普通 IO 使用)

46	PB0	LCD_BL		Y	TFTLCD 接口背光控制脚
47	PB1	T_SCK		Y	TFTLCD 接口触摸屏 SCK 信号
48	PB2	BOOT1	T_MISO	N	1, BOOT1, 启动选择配置引脚(仅上电时用) 2, TFTLCD 接口触摸屏 MISO 信号
133	PB3	JTDO	OV_WEN	N	1, JTAG 仿真口(JTDO) 2, OLED/CAMERA 接口 WEN 脚(使用时, 需先禁止 JTAG, 才可以当普通 IO 使用)
134	PB4	JTRST	OV_RCLK	N	1, JTAG 仿真口(JTRST) 2, OLED/CAMERA 接口 RCLK 脚(使用时, 需先禁止 JTAG, 才可以当普通 IO 使用)
135	PB5	LED0		N	接 DS0 LED 灯(红色)
136	PB6	IIC_SCL		N	接 24C02 的 SCL
137	PB7	IIC_SDA		N	接 24C02 的 SDA
139	PB8	BEEP		N	接蜂鸣器(BEEP)
140	PB9	REMOTE_IN		N	接 LF0038 红外接收头
69	PB10	USART3_TX		Y	1, RS232 串口 3(COM3)RX 脚(P8 设置+K1 选择) 2, JOYPAD_DAT(P8 设置+K1 选择) 3, ATK-MODULE 接口的 RXD 脚(P8 设置)
70	PB11	USART3_RX		Y	1, RS232 串口 3(COM3)TX 脚(P8 设置+K1 选择) 2, JOYPAD_LAT(P8 设置+K1 选择) 3, ATK-MODULE 接口的 TXD 脚(P8 设置)
73	PB12	F_CS		N	25Q128 的片选信号
74	PB13	SPI2_SCK		N	25Q128 和 WIRELESS 接口的 SCK 信号
75	PB14	SPI2_MISO		N	25Q128 和 WIRELESS 接口的 MISO 信号
76	PB15	SPI2_MOSI		N	25Q128 和 WIRELESS 接口的 MOSI 信号
26	PC0	OV_D0		Y	OLED/CAMERA 接口的 D0 脚
27	PC1	OV_D1		Y	OLED/CAMERA 接口的 D1 脚
28	PC2	OV_D2		Y	OLED/CAMERA 接口的 D2 脚
29	PC3	OV_D3		Y	OLED/CAMERA 接口的 D3 脚
44	PC4	OV_D4		Y	OLED/CAMERA 接口的 D4 脚
45	PC5	OV_D5		Y	OLED/CAMERA 接口的 D5 脚
96	PC6	OV_D6		Y	OLED/CAMERA 接口的 D6 脚
97	PC7	OV_D7		Y	OLED/CAMERA 接口的 D7 脚
98	PC8	SDIO_D0		N	SD 卡接口的 D0
99	PC9	SDIO_D1		N	SD 卡接口的 D1
111	PC10	SDIO_D2		N	SD 卡接口的 D2
112	PC11	SDIO_D3		N	SD 卡接口的 D3
113	PC12	SDIO_SCK		Y	SD 卡接口的 SCK
7	PC13	VS_DREQ		N	接 VS1053 芯片的 DREQ 脚
8	PC14		RTC 晶振	N	接 32.768K 晶振, 不可用做 IO

9	PC15		RTC 晶振	N	接 32.768K 晶振，不可用做 I/O
114	PD0	FSMC_D2		N	FSMC 总线数据线 D2 (LCD/SRAM 共用)
115	PD1	FSMC_D3		N	FSMC 总线数据线 D3 (LCD/SRAM 共用)
116	PD2	SDIO_CMD		N	SD 卡接口的 CMD
117	PD3	OV_SCL	JOY_CLK	N	1, OLED/CAMERA 接口的 SCL 信号 2, JOYPAD 的 CLK 信号 (接在 COM3 口)
118	PD4	FSMC_NOE		N	FSMC 总线 NOE (RD) (LCD/SRAM 共用)
119	PD5	FSMC_NWE		N	FSMC 总线 NWE (WR) (LCD/SRAM 共用)
122	PD6	OV_WRST		Y	OLED/CAMERA 接口的 WRST 信号
123	PD7	CH395Q_RST	RS485_RE	N	1, CH395Q 复位引脚 2, RS485 RE 引脚
77	PD8	FSMC_D13		N	FSMC 总线数据线 D13 (LCD/SRAM 共用)
78	PD9	FSMC_D14		N	FSMC 总线数据线 D14 (LCD/SRAM 共用)
79	PD10	FSMC_D15		N	FSMC 总线数据线 D15 (LCD/SRAM 共用)
80	PD11	FSMC_A16		N	FSMC 总线地址线 A17 (SRAM 专用)
81	PD12	FSMC_A17		N	FSMC 总线地址线 A18 (SRAM 专用)
82	PD13	FSMC_A18		N	FSMC 总线地址线 A19 (SRAM 专用)
85	PD14	FSMC_D0		N	FSMC 总线数据线 D0 (LCD/SRAM 共用)
86	PD15	FSMC_D1		N	FSMC 总线数据线 D1 (LCD/SRAM 共用)
141	PE0	FSMC_NBL0		N	FSMC 总线 NBL0 (SRAM 专用)
142	PE1	FSMC_NBL1		N	FSMC 总线 NBL1 (SRAM 专用)
1	PE2	KEY2		Y	接按键 KEY2
2	PE3	KEY1		Y	接按键 KEY1
3	PE4	KEY0		Y	接按键 KEY0
4	PE5	LED1		N	接 DS1 LED 灯 (绿色)
5	PE6	VS_RST		N	接 VS1053 芯片的 RST 脚
58	PE7	FSMC_D4		N	FSMC 总线数据线 D4 (LCD/SRAM 共用)
59	PE8	FSMC_D5		N	FSMC 总线数据线 D5 (LCD/SRAM 共用)
60	PE9	FSMC_D6		N	FSMC 总线数据线 D6 (LCD/SRAM 共用)
63	PE10	FSMC_D7		N	FSMC 总线数据线 D7 (LCD/SRAM 共用)
64	PE11	FSMC_D8		N	FSMC 总线数据线 D8 (LCD/SRAM 共用)
65	PE12	FSMC_D9		N	FSMC 总线数据线 D9 (LCD/SRAM 共用)
66	PE13	FSMC_D10		N	FSMC 总线数据线 D10 (LCD/SRAM 共用)
67	PE14	FSMC_D11		N	FSMC 总线数据线 D11 (LCD/SRAM 共用)
68	PE15	FSMC_D12		N	FSMC 总线数据线 D12 (LCD/SRAM 共用)
10	PF0	FSMC_A0		N	FSMC 总线地址线 A0 (SRAM 专用)
11	PF1	FSMC_A1		N	FSMC 总线地址线 A1 (SRAM 专用)
12	PF2	FSMC_A2		N	FSMC 总线地址线 A2 (SRAM 专用)
13	PF3	FSMC_A3		N	FSMC 总线地址线 A3 (SRAM 专用)
14	PF4	FSMC_A4		N	FSMC 总线地址线 A4 (SRAM 专用)
15	PF5	FSMC_A5		N	FSMC 总线地址线 A5 (SRAM 专用)

18	PF6	VS_XDCS		N	接 VS1053 芯片的 XDCS 脚
19	PF7	VS_XCS		N	接 VS1053 芯片的 XCS 脚
20	PF8	LIGHT_SENSOR		N	接光敏传感器 (LS1)
21	PF9	T_MOSI		Y	TFTLCD 接口触摸屏 MOSI 信号
22	PF10	T_PEN		Y	TFTLCD 接口触摸屏 PEN 信号
49	PF11	T_CS		Y	TFTLCD 接口触摸屏 CS 信号
50	PF12	FSMC_A6		N	FSMC 总线地址线 A6 (SRAM 专用)
53	PF13	FSMC_A7		N	FSMC 总线地址线 A7 (SRAM 共用)
54	PF14	FSMC_A8		N	FSMC 总线地址线 A8 (SRAM 专用)
55	PF15	FSMC_A9		N	FSMC 总线地址线 A9 (SRAM 专用)
56	PG0	FSMC_A10		N	FSMC 总线地址线 A10 (SRAM/LCD 共用)
57	PG1	FSMC_A11		N	FSMC 总线地址线 A11 (SRAM 专用)
87	PG2	FSMC_A12		N	FSMC 总线地址线 A12 (SRAM 专用)
88	PG3	FSMC_A13		N	FSMC 总线地址线 A13 (SRAM 专用)
89	PG4	FSMC_A14		N	FSMC 总线地址线 A14 (SRAM 专用)
90	PG5	FSMC_A15		N	FSMC 总线地址线 A15 (SRAM 专用)
91	PG6	NRF_IRQ	CH395Q_INT	N	1, WIRELESS 接口 IRQ 信号 2, CH395Q 中断信号
92	PG7	NRF_CS		Y	WIRELESS 接口的 CS 信号
93	PG8	NRF_CE		Y	WIRELESS 接口的 CE 信号
124	PG9	FSMC_NE2		N	FSMC 总线的片选信号 2, 为 CH395Q 片选信号
125	PG10	FSMC_NE3		N	FSMC 总线的片选信号 3, 为外部 SRAM 片选信号
126	PG11	1WIRE_DQ		N	单总线接口 (U6) 数据线, 接 DHT11/DS18B20
127	PG12	FSMC_NE4		Y	FSMC 总线的片选信号 4, 为 LCD 片选信号
128	PG13	OV_SDA		N	OLED/CAMERA 接口的 SDA 脚
129	PG14	OV_RRST		Y	OLED/CAMERA 接口的 RRST 脚
132	PG15	OV_OE		Y	OLED/CAMERA 接口的 OE 脚

表 1.2.2.1 战舰 V4 IO 资源分配总表

表 1.2.2.1 中, 引脚栏即 STM32F103ZET6 的引脚编号; GPIO 栏则表示 GPIO; 连接资源栏表示了对应 GPIO 所连接到的网络; 独立栏, 表示该 IO 是否可以完全独立 (不接其他任何外设和上下拉电阻) 使用, 通过一定的方法, 可以达到完全独立使用该 IO, Y 表示可做独立 IO, N 表示不可做独立 IO; 连接关系栏, 则对每个 IO 的连接做了简单的介绍。

该表在: A 盘→3, 原理图 文件夹下有提供 Excel 格式, 并注有详细说明和使用建议, 大家可以打开该表格的 Excel 版本, 详细查看。

1.3 战舰 V4 升级说明

正点原子战舰 V4 开发板相对于 V3 版本，主要变化如表 1.3.1 所示：

编号	对比项	正点原子战舰 STM32 开发板		说明
		V3 版本	V4 版本	
1	SPI FLASH 芯片	W25Q128	NM25Q128/BY25Q128	换国产芯片
2	10/100M 以太网	DM9000	CH395Q	带协议栈，使用简单
3	DCDC 电源芯片	MP2359	JW5060T	换国产芯片
4	USB 转串口	CH340G	CH340C	改进设计
5	RS232 接口芯片	SP3232	TP3232	换国产芯片
6	RS485 接口芯片	SP3485	TP8485	换国产芯片
7	CAN 接口芯片	TJA1050	SIT1050	换国产芯片
8	SRAM 芯片	IS62WV51216	XM8A51216	换国产芯片
9	内存卡接口	SD 卡	TF 卡	更小更通用
10	USB 接口	Mini USB	Type C USB	更主流
11	OV_SCL/OV_SDA 上拉	无	有	更合理
12	可调电位器	无	有	更方便
13	红外接收头	HS0038	LF0038	更小巧

表 1.3.1 V4 版本 VS V3 版本硬件变更表

从表 1.3.1 可以看出，战舰 V4 开发板在 V3 版本的基础上进行了不少改进，主要包括国产器件的大量使用、网口芯片修改、内存卡和 USB 接口更改、增加可调电位器等修改。

以上修改基本不涉及到 IO 口变动，因此大家以前编写的代码，很多都是可以直接在战舰 V4 上面运行的，或者只需要经过很少的改动（如部分芯片驱动的小修改），就可以完成适配。

第二章 实验平台硬件资源详解

本章，我们将向大家详细介绍正点原子战舰 STM32F103 各部分的硬件原理图，让大家对该开发板的各部分硬件原理有个深入理解，并向大家介绍开发板的使用注意事项，为后面的学习做好准备。

本章将分为如下两节：

- 2.1，开发板原理图详解；
- 2.2，开发板使用注意事项；

2.1 开发板原理图详解

2.1.1 MCU

正点原子战舰 STM32 开发板选择的是 STM32F103ZET6 作为 MCU，该芯片是 STM32F103 里面配置非常强大的了，它拥有的资源包括：64KB SRAM、512KB FLASH、2 个基本定时器、4 个通用定时器、2 个高级定时器、2 个 DMA 控制器（共 12 个通道）、3 个 SPI、2 个 IIC、5 个串口、1 个 USB、1 个 CAN、3 个 12 位 ADC、1 个 12 位 DAC、1 个 SDIO 接口、1 个 FSMC 接口以及 112 个通用 IO 口。该芯片的配置十分强悍，并且还带外部总线（FSMC）可以用来外扩 SRAM 和连接 LCD 等，通过 FSMC 驱动 LCD，可以显著提高 LCD 的刷屏速度，是 STM32F1 家族常用型号里面，最高配置的芯片了，所以我们选择了它作为我们战舰板的主芯片。MCU 部分的原理图如图 2.1.1.1-1 和图 2.1.1.1-2（由于 MCU 引脚比较多，因此我们把原理图分成 2 部分，方便查看）所示：





这里主要讲解以下 3 个地方:

3, 图中 P5 是参考电压选择端口。我们开发板默认是接板载的 3.3V 作为参考电压, 如果大家想用自己的参考电压, 则把你的参考电压接入 Vref+ 即可。

正点原子战舰 STM32F103 引出了 STM32F103ZET6 的所有 IO 口，如图 2.1.2.1 所示：

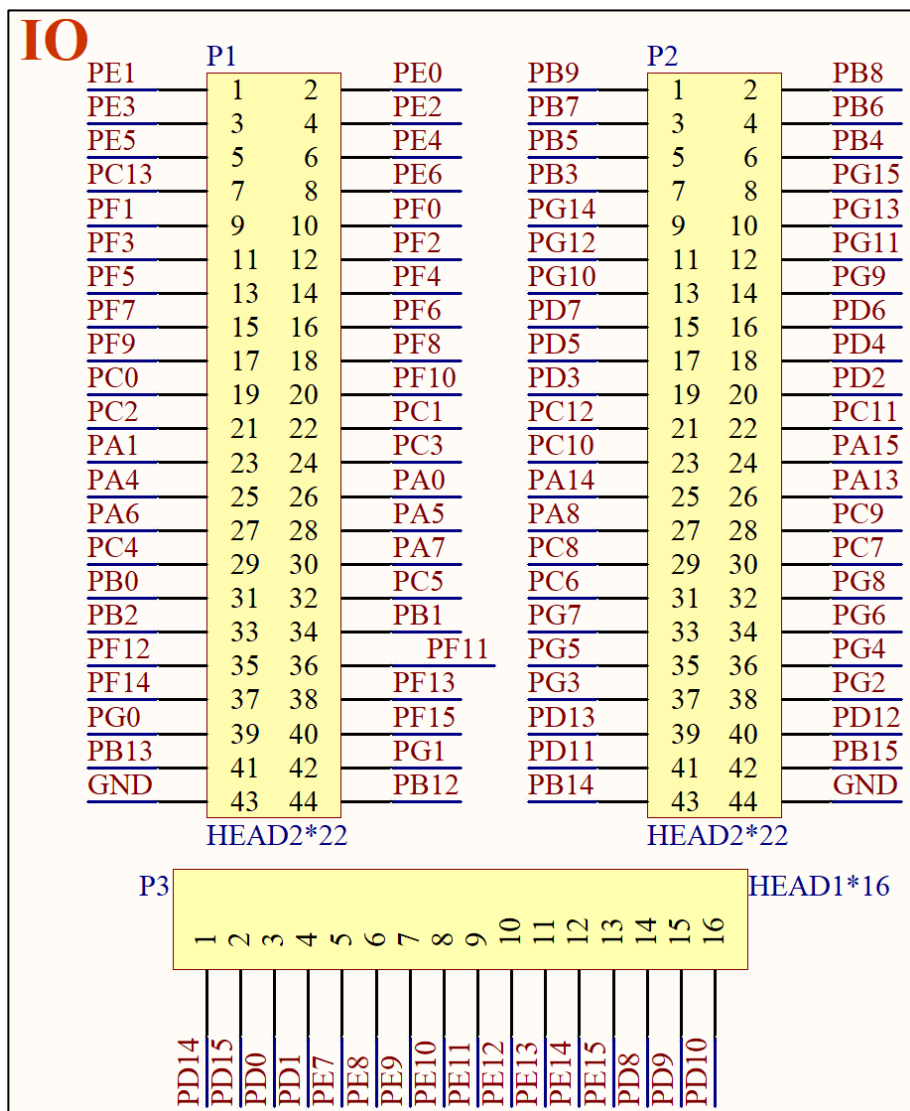


图 2.1.2.1 引出 IO 口

图中 P1、P2 和 P3 为 MCU 主 IO 引出口，这三组排针共引出了 102 个 IO 口，STM32F103ZET6 总共有 112 个 IO，除去 RTC 晶振占用的 2 个，还剩 110 个，这三组主引出排针，总共引出了 102 个 IO，剩下的 8 个 IO 口分别通过：P4（PA9&PA10）、P7（PA2&PA3）、P8（PB10&PB11）和 P9（PA11&PA12）等 4 组排针引出。

2.1.3 USB 串口/串口 1 选择接口

正点原子战舰 STM32F103 板载的 USB 串口和 STM32F103ZET6 的串口是通过 P4 连接起来的，如图 2.1.3.1 所示：

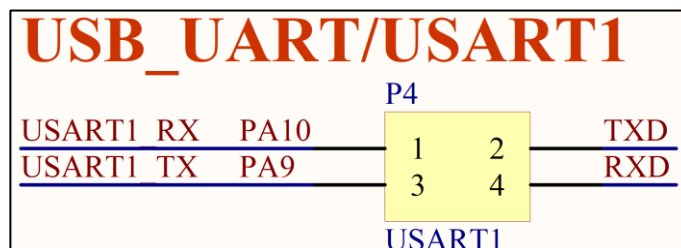


图 2.3.1.1 USB 串口/串口 1 选择接口

图中 TXD/RXD 是相对 CH340 来说的，也就是 USB 串口的发送和接收脚。而 USART1_RX 和 USART1_TX 则是相对于 STM32F103ZET6 来说的。这样，通过对接，就可以实现 USB 串口和 STM32F103ZET6 的串口通信了。同时，P4 是 PA9 和 PA10 的引出口。

这样设计的好处就是使用上非常灵活。比如需要用到外部 TTL 串口和 STM32 通信的时候，只需要拔了跳线帽，通过杜邦线连接外部 TTL 串口，就可以实现和外部设备的串口通信了；又比如我有个板子需要和电脑通信，但是电脑没有串口，那么你就可以使用开发板的 RXD 和 TXD 来连接你的设备，把我们的开发板当成 USB 转 TTL 串口用了。

2.1.4 JTAG/SWD

正点原子战舰 STM32F103 板载的标准 20 针 JTAG/SWD 接口电路如图 2.1.4.1 所示：

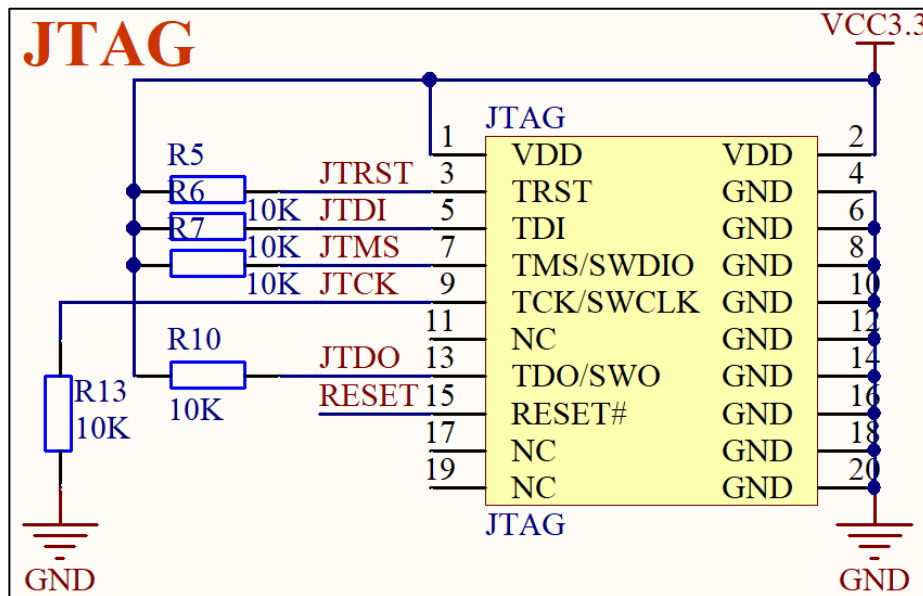


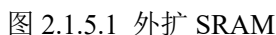
图 2.1.4.1 JTAG/SWD 接口

这里，我们采用的是标准的 JTAG 接法，但是 STM32 还有 SWD 接口，SWD 只需要最少 2 跟线（SWCLK 和 SWDIO）就可以下载并调试代码了，这同我们使用串口下载代码差不多，而且速度非常快，能调试。所以建议大家在设计产品的时候，可以留出 SWD 来下载调试代码，而摒弃 JTAG。STM32 的 SWD 接口与 JTAG 是共用的，只要接上 JTAG，你就可以使用 SWD 模式了（其实并不需要 JTAG 这么多线），当然，你的调试器必须支持 SWD 模式，DAP、ST LINK、JLINK 和 ULINK 等都支持 SWD 调试。

特别提醒，JTAG 有几个信号线用来接其他外设了，但是 SWD 是完全没有接任何其他外设的，所以在使用的時候，**推荐大家一律使用 SWD 模式!!!**

2.1.5 SRAM

正点原子战舰 STM32F103 外扩了 1M 字节的 SRAM 芯片，如图 2.1.5.1 所示，注意图中的地址线标号，是以 XM8A51216 为模版的，但是和 XM8A51216 的 datasheet 标号有出入，不过，因为地址的唯一性，这并不会影响我们使用 XM8A51216（特别提醒：地址线可以乱，但是数据线必须一致!!），因此，该原理图对这两个芯片都是可以正常使用的。



2.1.6 LCD 模块接口

LCD

Pin	Function	Pin	Function	Pin	Function
1	FSMC NE4	17	FSMC D11	1	FSMC A10
2	FSMC D2	18	FSMC D12	2	FSMC NOE
3	FSMC D3	19	FSMC D13	3	FSMC D0
4	FSMC D4	20	FSMC D14	4	FSMC D1
5	FSMC D5	21	FSMC D15	5	FSMC D2
6	FSMC D6	22		6	FSMC D3
7	FSMC D7	23	LCD BL	7	FSMC D4
8	FSMC D8	24	VCC3.3	8	FSMC D5
9	FSMC D9	25	VCC3.3	9	FSMC D6
10	FSMC D10	26		10	FSMC D7
11	FSMC D11	27	VDD3.3	11	FSMC D8
12	FSMC D12	28	VCC5	12	FSMC D9
13	FSMC D13	29	T MISO	13	FSMC D10
14	FSMC D14	30	T MOSI	14	FSMC D11
15	FSMC D15	31	T PEN	15	FSMC D12
16	FSMC D16	32		16	FSMC D13
17	FSMC D17	33	T CS	17	FSMC D14
18	FSMC D18	34	T SCK	18	FSMC D15
19	FSMC D19			19	FSMC D16
20	FSMC D20			20	FSMC D17
21	FSMC D21			21	FSMC D18
22	FSMC D22			22	FSMC D19
23	FSMC D23			23	FSMC D20
24	FSMC D24			24	FSMC D21
25	FSMC D25			25	FSMC D22
26	FSMC D26			26	FSMC D23
27	FSMC D27			27	FSMC D24
28	FSMC D28			28	FSMC D25
29	FSMC D29			29	FSMC D26
30	FSMC D30			30	FSMC D27
31	FSMC D31			31	FSMC D28
32	FSMC D32			32	FSMC D29
33	FSMC D33			33	FSMC D30
34	FSMC D34			34	FSMC D31

TFT LCD

图 2.1.6.1 LCD 模块接口

图中 TFT_LCD 是一个通用的液晶模块接口，支持正点原子全系列 TFTLCD 模块，包括：2.4 寸、2.8 寸、3.5 寸、4.3 寸和 7 寸等尺寸的 TFTLCD 模块。LCD 接口连接在 STM32F103ZET6 的 FSMC 总线上面，可以显著提高 LCD 的刷屏速度。

图中的 T_MISO/T_MOSI/T_PEN/T_SCK/T_CS 连接在 MCU 的 PB2/PF9/PF10/PB1/PF11 上，这些信号用来实现对液晶触摸屏的控制（支持电阻屏和电容屏）。LCD_BL 连接在 MCU 的 PB0 上，用于控制 LCD 的背光。液晶复位信号 RESET 则是直接连接在开发板的复位按钮上，和 MCU 共用一个复位电路。

2.1.7 复位电路

正点原子战舰 STM32F103 的复位电路如图 2.1.7.1 所示：

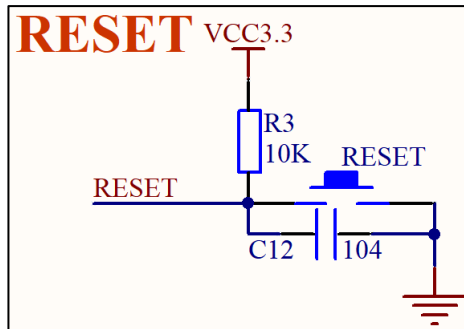


图 2.1.7.1 复位电路

因为 STM32 是低电平复位的，所以我们设计的电路也是低电平复位的，这里的 R3 和 C12 构成了上电复位电路。同时，开发板把 TFT_LCD 的复位引脚也接在 RESET 上，这样这个复位按钮不仅可以用来复位 MCU，还可以复位 LCD。

2.1.8 启动模式设置接口

正点原子战舰 STM32F103 的启动模式设置端口电路如图 2.1.8.1 所示：

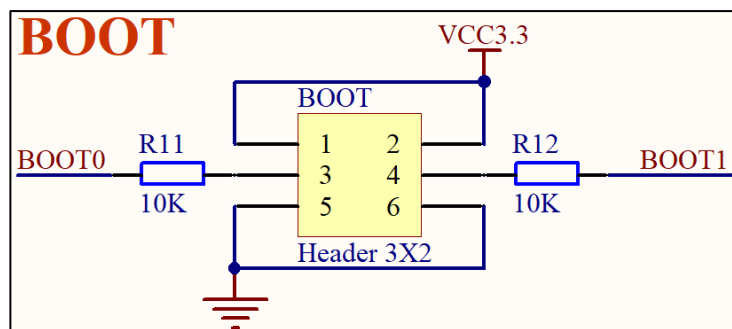


图 2.1.8.1 启动模式设置接口

上图的 BOOT0 和 BOOT1 用于设置 STM32 的启动方式，其对应启动模式如表 2.1.8.1 所示：

BOOT0	BOOT1	启动模式	说明
0	X	用户闪存存储器	用户闪存存储器，也就是 FLASH 启动
1	0	系统存储器	系统存储器启动，用于串口下载代码
1	1	SRAM 启动	SRAM 启动，用于在 SRAM 中调试代码

表 2.1.8.1 BOOT0、BOOT1 启动模式表

按照表 2.1.8.1，一般情况下如果我们想用串口下载代码，则必须配置 BOOT0 为 1，BOOT1 为 0，而如果想让 STM32 一按复位键就开始跑代码，则需要配置 BOOT0 为 0，BOOT1 随便设

置都可以。这里正点原子战舰 STM32F103 专门设计了一键下载电路，通过串口的 DTR 和 RTS 信号，来自动配置 BOOT0 和 RST 信号，因此不需要用户来手动切换他们的状态，直接串口下载软件自动控制，可以非常方便的下载代码。

2.1.9 RS232 串口/JOYPAD 接口

正点原子战舰 STM32F103 板载了一公一母两个 RS232 接口，其中 COM3 不但可以接 RS232 还可以接游戏手柄（JOYPAD），电路原理图如图 2.1.9.1 所示：

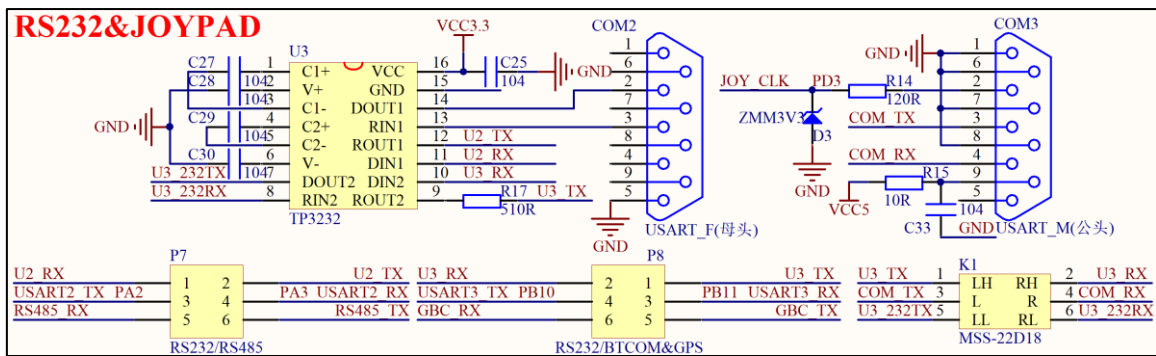


图 2.1.9.1 RS232 串口

因为 RS232 电平不能直接连接到 STM32，所以需要有一个电平转换芯片。这里我们选择的是 TP3232（也可以用 SP3232）来做电平转接，同时图中的 P7 用来实现 RS232(COM2)/RS485 的选择，P8 用来实现 RS232(COM3)/ATK 模块接口的选择，以满足不同实验的需要。

图中 COM2 是母头，COM3 是公头，而且 COM3 可以接 RS232 串口或者接 FC 游戏手柄（JOYPAD），具体选择哪个功能，则是通过 K1 开关来切换（请看板载丝印）。使用的时候，要特别注意，K1 先设置对了，再去接 RS232 串口/FC 游戏手柄。

图中 USART2_TX/USART2_RX 连接在 MCU 的串口 2 上（PA2/PA3），所以这里的 RS232(COM2)/RS485 都是通过串口 2 来实现的。图中 RS485_TX 和 RS485_RX 信号接在 SP3485 的 DI 和 RO 信号上。

而图中的 USART3_TX/USART3_RX 则是连接在 MCU 的串口 3 上（PB10/PB11），所以 RS232(COM3)/ATK 模块接口都是通过串口 3 来实现的。图中 GBC_RX 和 GBC_TX 连接在 ATK 模块接口 U5 上面。

因为 P7/P8 的存在，其实还带来另外一个好处，就是我们可以把开发板变成一个 RS232 电平转换器，或者 RS485 电平转换器，比如你买的核心板，可能没有板载 RS485/RS232 接口，通过连接战舰 STM32F103 的 P7/P8 端口，就可以让你的核心板拥有 RS232/RS485 的功能。

2.1.10 RS485 接口

正点原子战舰 STM32F103 板载的 RS485 接口电路如图 2.1.10.1 所示：

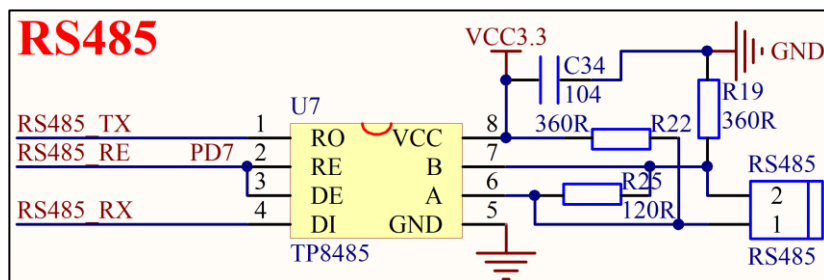


图 2.1.10.1 RS485 接口

RS485 电平也不能直接连接到 STM32，同样需要电平转换芯片。这里我们使用 TP8485 来做 485 电平转换，其中 R25 为终端匹配电阻，而 R22 和 R19，则是两个偏置电阻，以保证静默状态时，485 总线维持逻辑 1。

RS485_RX/RS485_TX 连接在 P7 上面，通过 P7 跳线来选择是否连接在 MCU 上面，RS485_RE 则是直接连接在 MCU 的 IO 口（PD7）上的，该信号用来控制 TP8485 的工作模式（高电平为发送模式，低电平为接收模式）。

另外，**特别注意**：RS485_RE 和 CH395Q_RST 共同接在 PD7 上面，在同时用到这两个外设的时候，需要注意下。

2.1.11 CAN/USB 接口

正点原子战舰 STM32F103 板载的 CAN 接口电路以及 STM32 USB 接口（Type C 接口）电路如图 2.1.11.1 所示：

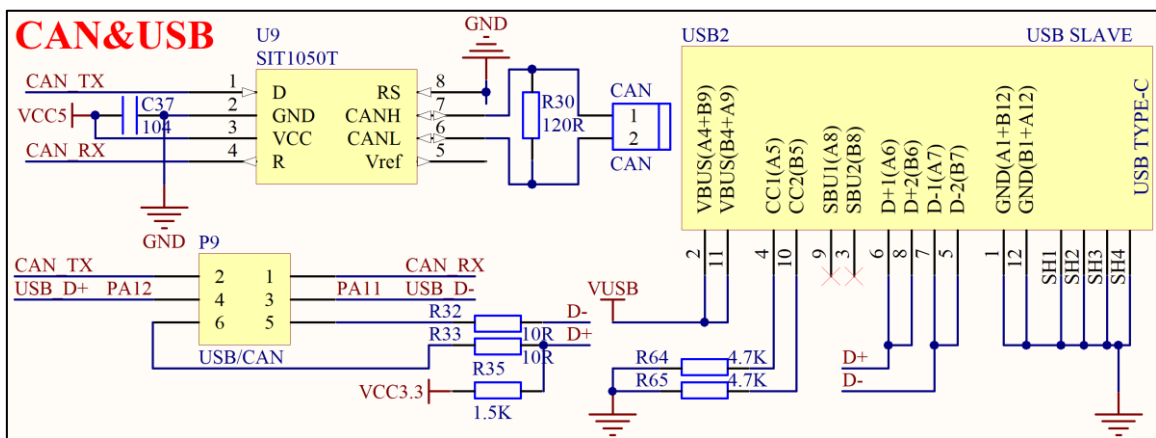


图 2.1.11.1 CAN/USB 接口

CAN 总线电平也不能直接连接到 STM32，同样需要电平转换芯片。这里我们使用 SIT1050T 来做 CAN 电平转换，其中 R30 为终端匹配电阻。

USB_D+/USB_D-连接在 MCU 的 USB 口（PA12/PA11）上，同时，因为 STM32 的 USB 和 CAN 共用这组信号，所以我们通过 P9 来选择使用 USB 还是 CAN。

USB_SLAVE 可以用来连接电脑，实现 USB 读卡器或 USB 虚拟串口等 USB 从机实验。另外，该接口还具有供电功能，VUSB 为开发板的 USB 供电电压，通过这个 USB 口，就可以给整个开发板供电了。

2.1.12 EEPROM

正点原子战舰 STM32F103 板载的 EEPROM 电路如图 2.1.12.1 所示：

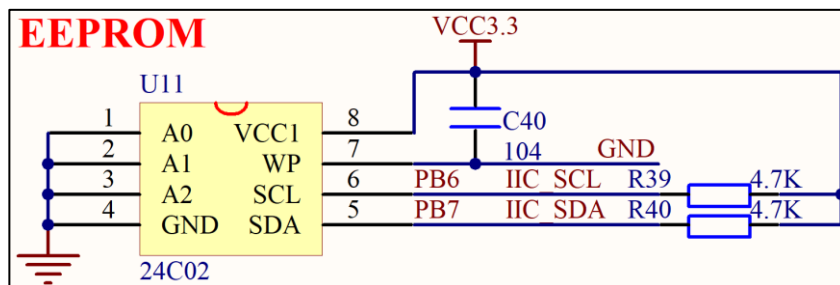


图 2.1.12.1 EEPROM

EEPROM 芯片我们使用的是 24C02，该芯片的容量为 2Kb，也就是 256 个字节，对于我们

普通应用来说是足够的。当然，你也可以选择换大容量的芯片，因为我们的电路在原理上是兼容 24C02~24C512 全系列 EEPROM 芯片的。

这里我们把 A0~A2 均接地，对 24C02 来说也就是把地址位设置成了 0 了，写程序的时候要注意这点。IIC_SCL 接在 MCU 的 PB6 上，IIC_SDA 接在 MCU 的 PB7 上，这里我们虽然接到了 STM32 的硬件 IIC 上，但是我们并不提倡使用硬件 IIC，因为 STM32 的 IIC 是鸡肋！请谨慎使用。

2.1.13 光敏传感器

正点原子战舰 STM32F103 板载了一个光敏传感器，可以用来感应周围光线的变化，该部分电路如图 2.1.13.1 所示：

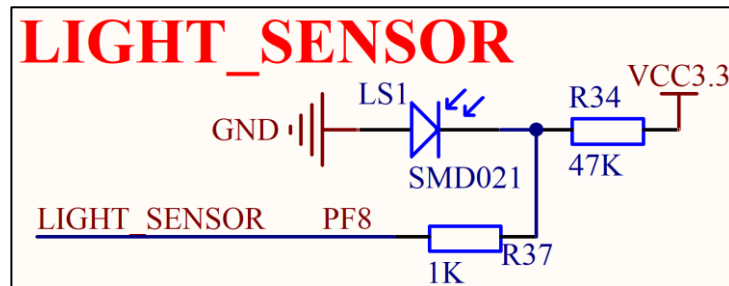


图 2.1.13.1 光敏传感器电路

图中的 LS1 就是光敏传感器，其实就是一个光敏二极管（PTSMD021），周围环境越亮，电流越大，反之电流越小，即可等效为一个电阻，环境越亮阻值越小，反之越大，从而通过读取 LIGHT_SENSOR 的电压，即可知道周围环境光线强弱。LIGHT_SENSOR 连接在 MCU 的 ADC3_IN6（ADC3 通道 6）上面，即 PF8 引脚。

2.1.14 SPI FLASH

正点原子战舰 STM32F103 板载的 SPI FLASH 电路如图 2.1.14.1 所示：

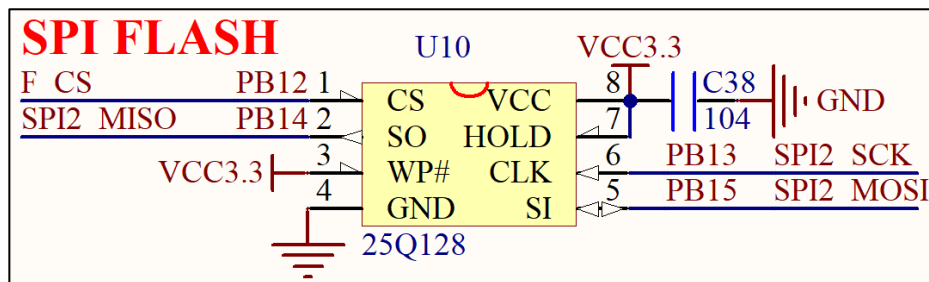


图 2.1.14.1 SPI FLASH 芯片

SPIFLASH 芯片型号为 25Q128（可选品牌为：诺存、华邦、博雅等都可以），该芯片的容量为 128Mb，也就是 16M 字节。该芯片和 NRF24L01 共用一个 SPI（SPI2），通过片选来选择使用某个器件，在使用其中一个器件的时候，请务必禁止另外一个器件的片选信号。

图中 F_CS 连接在 MCU 的 PB12 上，SPI2_SCK/SPI2_MOSI/SPI2_MISO 则分别连接在 MCU 的 PB13/PB15/PB14 上。

2.1.15 温湿度传感器接口

正点原子战舰 STM32F103 板载的温湿度传感器接口电路如图 2.1.15.1 所示：

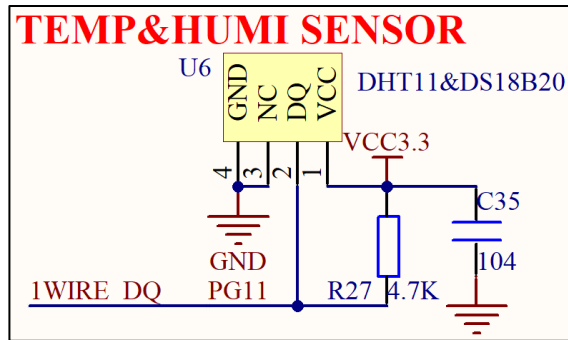


图 2.1.15.1 温湿度传感器接口

该接口（U6）支持 DS18B20/DS1820/DHT11 等单总线数字温湿度传感器。1WIRE_DQ 是传感器的数据线，该信号连接在 MCU 的 PG11 上。

2.1.16 红外接收头

正点原子战舰 STM32F103 板载的红外接收头电路如图 2.1.16.1 所示：

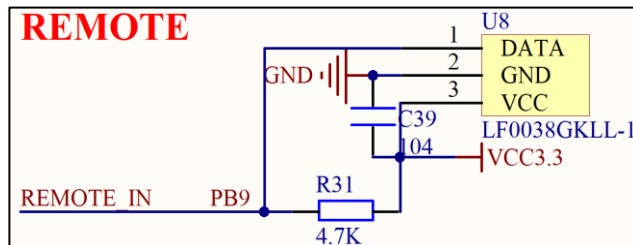


图 2.1.16.1 红外接收头

LF0038 是一个通用的红外接收头，几乎可以接收市面上所有红外遥控器的信号，有了它，就可以用红外遥控器来控制开发板了。REMOTE_IN 为红外接收头的输出信号，该信号连接在 MCU 的 PB9 上。

2.1.17 无线模块接口

正点原子战舰 STM32F103 板载的无线模块接口电路如图 2.1.17.1 所示：

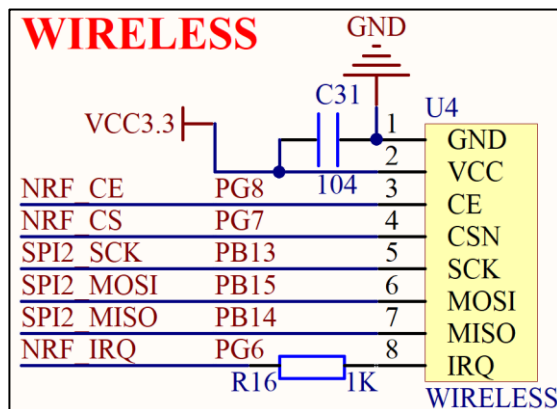


图 2.1.17.1 无线模块接口

该接口用来连接 NRF24L01 或者 RFID 等无线模块，从而实现开发板与其他设备的无线数据传输（注意：NRF24L01 不能和蓝牙/WIFI 连接）。NRF24L01 无线模块的最大传输速度可以达到 2Mbps，传输距离最大可以到 30 米左右（空旷地，无干扰）。

NRF_CE/NRF_CS/NRF_IRQ 连接在 MCU 的 PG8/PG7/PG6 上，而另外 3 个 SPI 信号则和

SPI FLASH 共用，接 MCU 的 SPI2。这里**需要注意**的是 PG6 还接了 DM9000_INT 这个信号，所以在使用 NRF_IRQ 中断引脚的时候，不能和 DM9000 同时使用，不过，如果没用到 NRF_IRQ 中断引脚，那么 DM9000 和无线模块就可以同时使用了。

2.1.18 LED

正点原子战舰 STM32F103 板载总共有 3 个 LED，其原理图如图 2.1.18.1 所示：

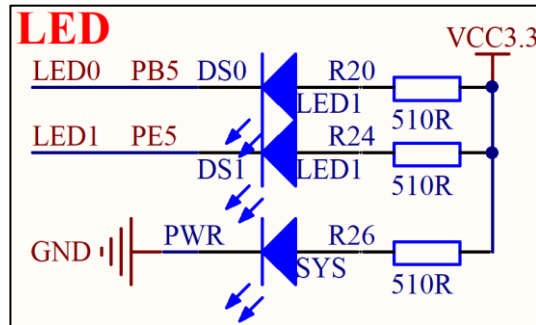


图 2.1.18.1 LED

其中 PWR 是系统电源指示灯，为蓝色。LED0(DS0)和 LED1(DS1)分别接在 PB5 和 PE5 上。为了方便大家判断，我们选择了 DS0 为红色的 LED，DS1 为绿色的 LED。

2.1.19 按键

正点原子战舰 STM32F103 板载总共有 4 个输入按键，其原理图如图 2.1.19.1 所示：

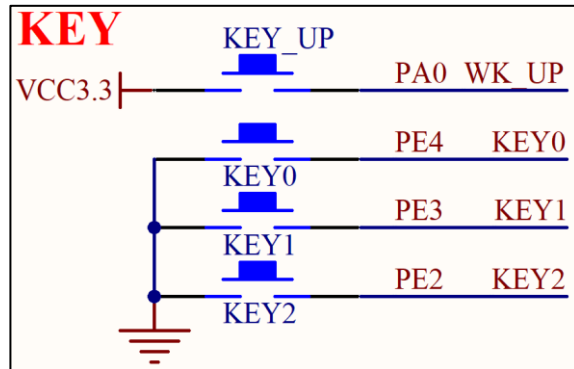


图 2.1.19.1 输入按键

KEY0、KEY1 和 KEY2 用作普通按键输入，分别连接在 PE4、PE3 和 PE2 上，这里并没有使用外部上拉电阻，但是 STM32 的 IO 作为输入的时候，可以设置上下拉电阻，所以我们使用 STM32 的内部上拉电阻来为按键提供上拉。

KEY_UP 按键连接到 PA0(STM32 的 WKUP 引脚)，它除了可以用作普通输入按键外，还可以用作 STM32 的唤醒输入。注意：这个按键是高电平触发的。

2.1.20 TPAD 电容触摸按键

正点原子战舰 STM32F103 板载了一个电容触摸按键，其原理图如图 2.1.20.1 所示：

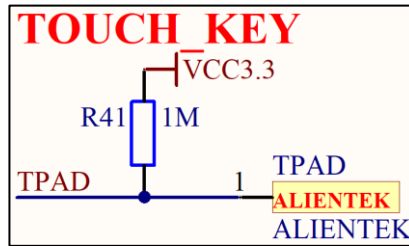


图 2.1.20.1 电容触摸按键

图中 1M 电阻是电容充电电阻，TPAD 并没有直接连接在 MCU 上，而是连接在多功能端口（P10）上面，通过跳线帽来选择是否连接到 STM32。多功能端口，我们将在 2.1.25 节介绍。电容触摸按键的原理我们将在后续的实战篇里面介绍。

2.1.21 OLED/摄像头模块接口

正点原子战舰 STM32F103 板载了一个 OLED/摄像头模块接口，其原理图如图 2.1.21.1 所示：

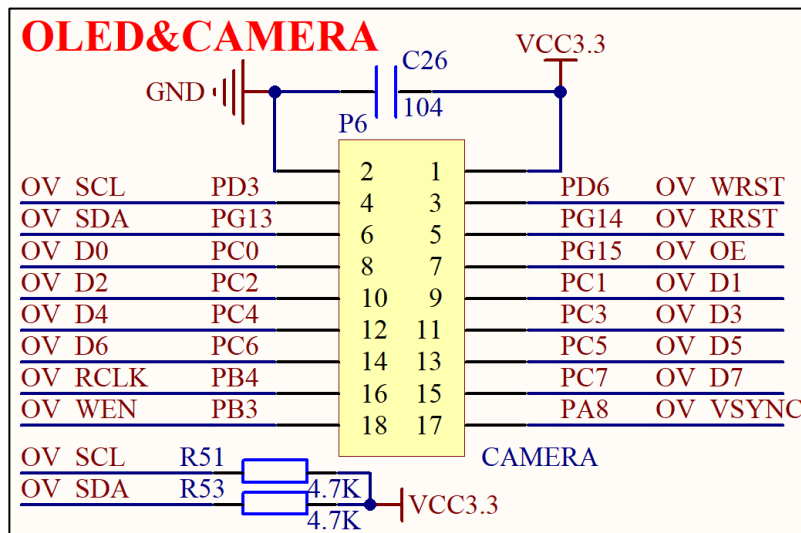


图 2.1.21.1 OLED/摄像头模块接口

图中 P6 是接口可以用来连接正点原子 OLED 模块或者正点原子 摄像头模块。如果是 OLED 模块，则 OV_WEN 和 OV_VSYNC 不需要接（在板上靠左插即可），如果是摄像头模块，则需要用到全部引脚。

其中，OV_SCL/OV_SDA/FIFO_WRST/OV_RRST/OV_OE 这 5 个信号是分别连接在 MCU 的 PD3/PG13/PD6/PG14/PG15 上面，OV_D0~OV_D7 则连接在 PC0~7 上面（放在连续的 IO 上，可以提高读写效率），OV_RCLK/OV_WEN/OV_VSYNC 这 3 个信号是分别连接在 MCU 的 PB4/PB3/PA8 上面。其中 PB3 和 PB4 又是 JTAG 的 JTRST/JTDO 信号，所以在使用 OV7725/OV7670 的时候，不要用 JTAG 仿真，要选择 SWD 模式（所以我们建议大家直接用 SWD 模式来连接我们的开发板，这样所有的实验都可以仿真！）。

特别注意：OV_SCL 和 JOY_CLK 共用 PD3，OV_VSYNC 和 PWM_DAC 共用 PA8，他们必须分时复用。另外，OV_SCL 和 OV_SDA 我们加了 4.7K 上拉电阻，在使用的时候，需要注意这个问题。

2.1.22 有源蜂鸣器

正点原子战舰 STM32F103 板载了一个有源蜂鸣器，其原理图如图 2.1.22.1 所示：

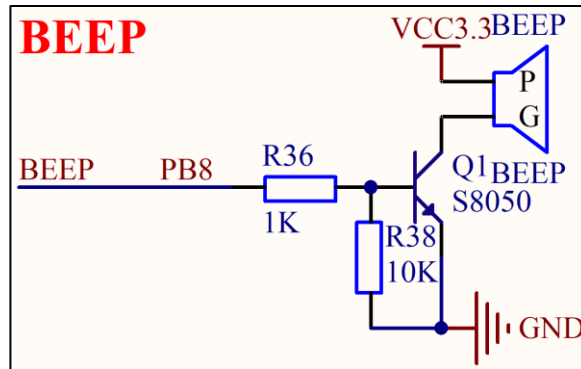


图 2.1.22.1 有源蜂鸣器

有源蜂鸣器是指自带了震荡电路的蜂鸣器，这种蜂鸣器一接上电就会自己震荡发声。而如果是无源蜂鸣器，则需要外加一定频率（2~5Khz）的驱动信号，才会发声。这里我们选择使用有源蜂鸣器，方便大家使用。

图中 Q1 是用来扩流，R38 则是一个下拉电阻，避免 MCU 复位的时候，蜂鸣器可能发声的现象。BEEP 信号直接连接在 MCU 的 PB8 上面，PB8 可以做 PWM 输出，所以大家如果想玩高级点（如：控制蜂鸣器“唱歌”），就可以使用 PWM 来控制蜂鸣器。

2.1.23 TF 卡接口

正点原子战舰 STM32F103 板载了一个 TF 卡（小卡/Micro SD 卡）接口，其原理图如图 2.1.23.1 所示：

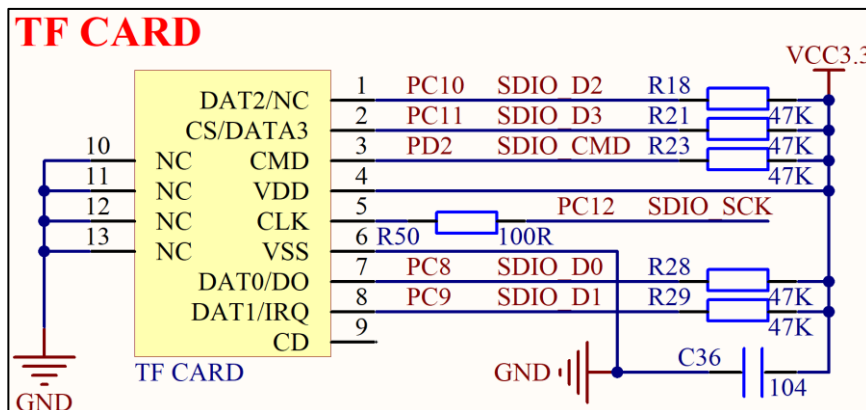


图 2.1.23.1 TF 卡接口

图中 TF_CARD 为 TF 卡接口，采用 4 位 SDIO 方式驱动，理论上最大速度可以达到 12MB/S，非常适合需要高速存储的情况。图中：SDIO_D0/SDIO_D1/SDIO_D2/SDIO_D3/SDIO_SCK/SDIO_CMD 分别连接在 MCU 的 PC8/PC9/PC10/PC11/PC12/PD2 上面。

2.1.24 ATK 模块接口

正点原子战舰 STM32F103 板载了 ATK 模块接口，其原理图如图 2.1.24.1 所示：

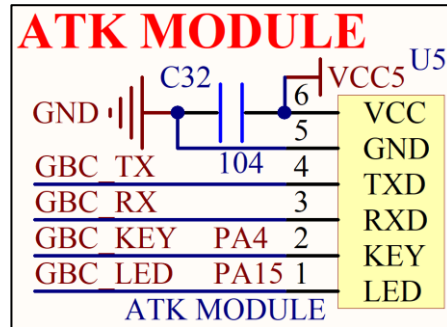


图 2.1.24.1 ATK 模块接口

如图所示，U5 是一个 1*6 的排座，可以用来连接正点原子推出的一些模块，比如：蓝牙串口模块、GPS 模块、MPU6050 模块等。有了这个接口，我们连接模块就非常简单，插上即可工作。

图中：GBC_TX/GBC_RX 可通过 P8 排针，选择接入 PB11/PB10（即串口 3），详见 2.1.9 节。而 GBC_KEY 和 GBC_LED 则分别连接在 MCU 的 PA4 和 PA15 上面。**特别注意：**GBC_KEY 与 PWM_DAC 共用 PA4，GBC_LED 和 JTDI 共用 PA15，在使用的时候，要注意这个问题。

2.1.25 多功能端口

正点原子战舰 STM32F103 板载的多功能端口，是由 P10 和 P11 构成的一个 6PIN 端口，其原理图如图 2.1.25.1 所示：

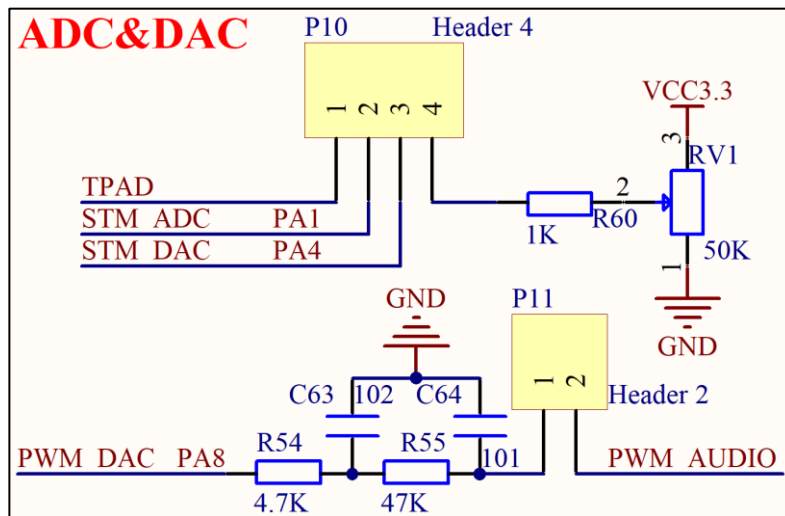


图 2.1.25.1 多功能端口

从上图，大家可能还看不出这个多功能端口的全部功能，别担心，下面我们会详细介绍。

首先介绍右侧的 P10，其中 TPAD 为电容触摸按键信号，连接在电容触摸按键上。STM_ADC 和 STM_DAC 则分别连接在 PA1 和 PA4 上，用于 ADC 采集或 DAC 输出。当需要电容触摸按键的时候，我们通过跳线帽短接 TPAD 和 STM_ADC，就可以实现电容触摸按键（利用定时器的输入捕获）。STM_DAC 信号则既可以用作 DAC 输出，也可以用作 ADC 输入，因为 STM32 的该管脚同时具有这两个复用功能。**特别注意：**STM_DAC 与摄像头的 GBC_KEY 共用 PA4，所以他们不可以同时使用，但是可以分时复用。P10 第四脚接可调电位器 RV1 的滑动端，通过调整 RV1 可以改变 P10 第 4 脚的电压（0~3.3V 之间）。

我们再来看看 P11，PWM_DAC 连接在 MCU 的 PA8，是定时器 1 的通道 1 输出，后面跟一个二阶 RC 滤波电路，其截止频率为 33.8Khz。经过这个滤波电路，MCU 输出的方波就变为

直流信号了。PWM_AUDIO 是一个音频输入通道，它连接到 TDA1308 和 HT6872 的输入，输出到耳机/扬声器。**特别注意：**PWM_DAC 和 OV_VSYNC 共用 PA8，所以 PWM_DAC 和摄像头模块，不可以同时使用，不过，可以分时复用。

单独介绍完了 P10 和 P11，我们再来看看他们组合在一起的多功能端口，如图 2.1.25.2 所示：

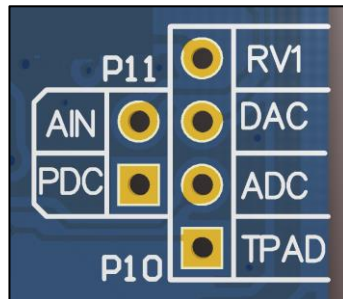


图 2.1.25.2 组合后的多功能端口

图中 AIN 是 PWM_AUDIO，PDC 是滤波后的 PWM_DAC 信号。下面我们来看看通过 1 个跳线帽，这个多功能接口可以实现哪些功能。

当不用跳线帽的时候：1，AIN 和 GND（从其他地方接 GND，下同）组成一个音频输入通道。2，PDC 和 GND 组成一个 PWM_DAC 输出；3，DAC 和 GND 组成一个 DAC 输出/ADC 输入（因为 DAC 脚也刚好也可以做 ADC 输入）；4，ADC 和 GND 组成一组 ADC 输入；5，TPAD 和 GND 组成一个触摸按键接口，可以连接其他板子实现触摸按键。

当使用 1 个跳线帽的时候：1，AIN 和 PDC 组成一个 MCU 的音频输出通道，实现 PWM DAC 播放音乐。2，AIN 和 DAC 同样可以组成一个 MCU 的音频输出通道，也可以用来播放音乐。3，DAC 和 ADC 组成一个自输出测试，用 MCU 的 ADC 来测试 MCU 的 DAC 输出。4，PDC 和 ADC，组成另外一个自输出测试，用 MCU 的 ADC 来测试 MCU 的 PWM DAC 输出。5，ADC 和 TPAD，组成一个触摸按键输入通道，实现 MCU 的触摸按键功能。6，DAC 和 RV1，组成一个 ADC 测试，通过 RV1 可调电阻可以调整电压值。

从上面的分析，可以看出，这个多功能端口可以实现 11 个功能，所以，只要设计合理，1+1 是大于 2 的。

2.1.26 以太网接口（RJ45）

正点原子战舰 STM32F103 板载了一个以太网接口(RJ45)，其原理图如图 2.1.26.1 所示：

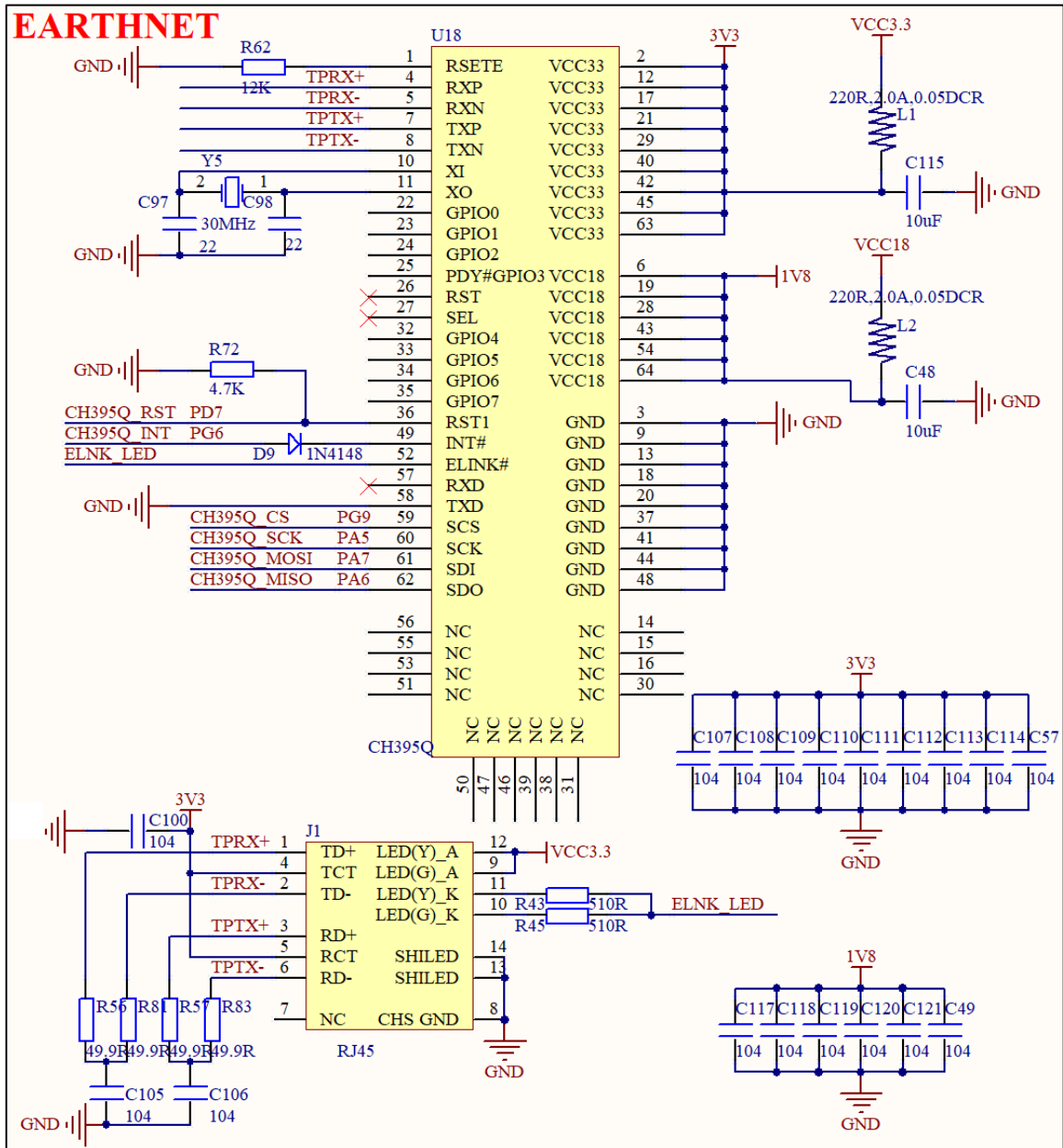


图 2.1.26.1 以太网接口电路

STM32F1 本身并不带网络功能，战舰 STM32 开发板板载了一颗 CH579Q 网络接口芯片，用于给 STM32F1 提供网络接口，该芯片集成以太网 MAC 层与网络协议栈，使用非常方便，有了它，战舰 STM32 开发板就可以实现网络相关的功能了。

CH579Q 和 STM32F103ZET6 的连接通过 SPI 接口连接（SPI1），总共 6 个信号线，连接关系为：CH579Q_RST/ CH579Q_INT/CH579Q_CS/CH579Q_SCK/CH579Q_MOSI/CH579Q_MISO 分别接在 PD7/PG6/PG9/PA5/PA7/PA6 上。SPI1 上面还挂了 VS1053B，它和 CH579Q 通过不同片选分时复用，互不影响。

特别注意：CH579Q_RST 和 RS485_RE 共用 PD7，CH579Q_INT 和 NRF_IRQ 共用 PG6，在使用的时候，注意要分时复用。另外，CH579Q 的中断引脚，我们加了 D9 二极管，防止干扰 NRF_IRQ，所以，在战舰板上，仅支持低电平有效的中断方式。

2.1.27 板载喇叭

正点原子战舰 STM32 开发板板载了一个小喇叭（扬声器），通过 D 类功放驱动，其原理图如图 2.1.27.1 所示：

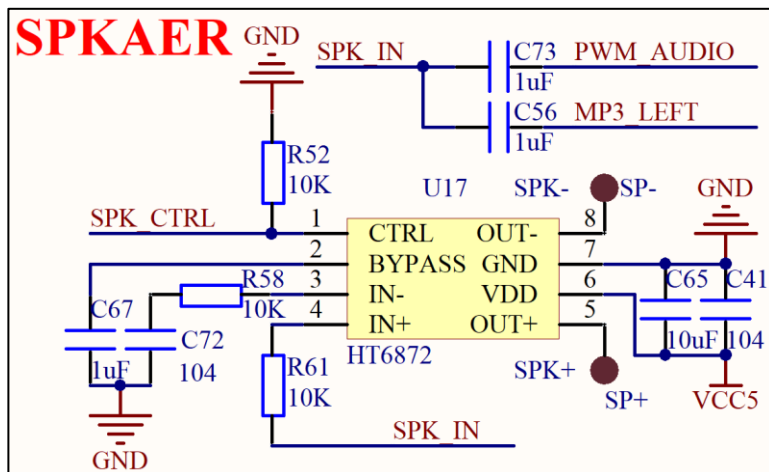


图 2.1.27.1 喇叭输出电路

HT6872 是一款低 EMI，防削顶失真的，单声道免滤波 D 类音频功率放大器。在 6.5V 电源，10%THD+N，4Ω 负载条件下，输出 4.71W 功率，在各类音频终端应用中维持高效率并提供 AB 类放大器的性能。在战舰 STM32 开发板上面，我们采用它来驱动板载的 8Ω 2W 喇叭。

图中 SP-和 SP+是喇叭焊接焊盘（在开发板底部），开发板板载的喇叭就是焊接在这两个焊盘上。

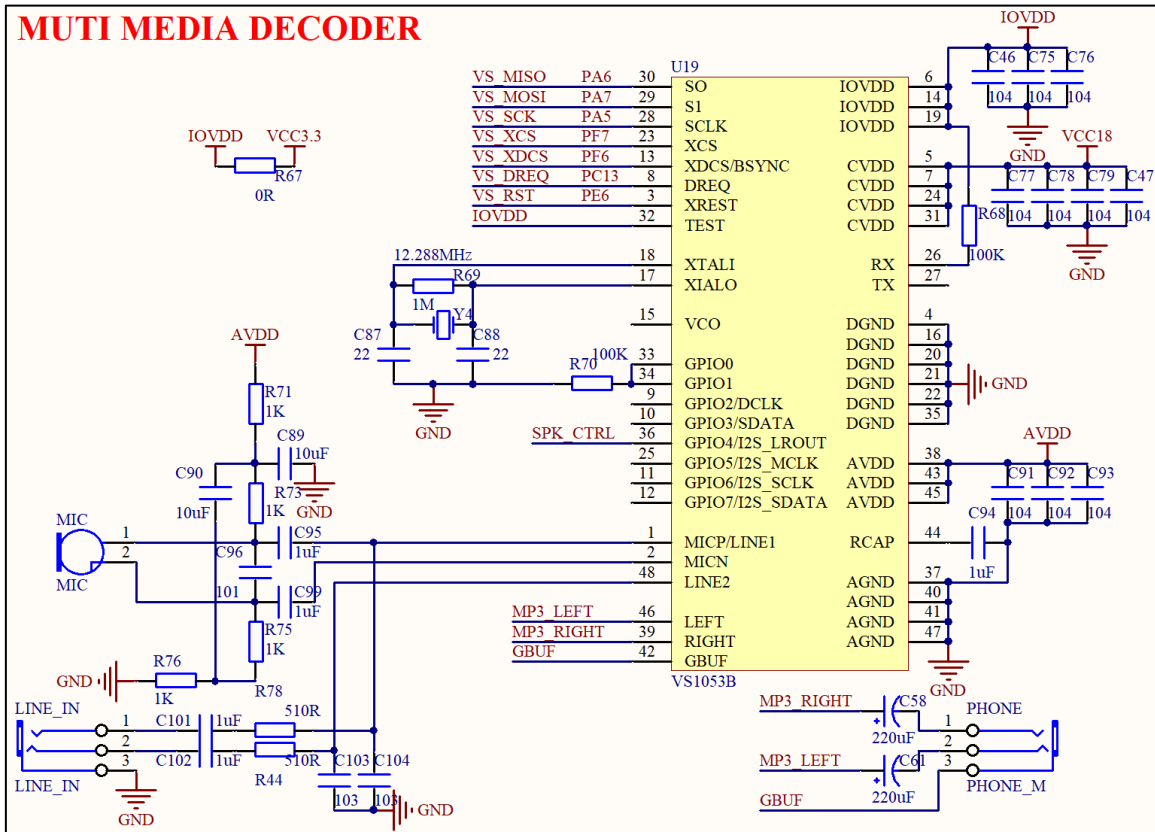
SPK_IN 是 HT6872 的音频信号输入端，该信号来自 MP3_LEFT/PWM_AUDIO，同时耦合进来。

SPK_CTRL 则由 VS1053 的 GPIO4 控制（见 2.1.28 节），当 SPK_CTRL 为低电平的时候，HT6872 进入关断模式，当 SPK_CTRL 为高电平的时候，HT6872 正常工作，这里我们加了 10K 的下拉电阻，所以，默认情况下，HT6872 是关断的，也就是喇叭并不会发声。我们必须在程序上，控制 VS1053 的 GPIO4 输出高电平，才可以使板载喇叭发声。

有了板载喇叭，我们就可以直接通过板载喇叭欣赏开发板播放的音乐或者其他音频了，更加人性化。

2.1.28 音频编解码

正点原子战舰 STM32 开发板板载 VS1053 音频编解码芯片，其原理图如图 2.1.28.1 所示：



2.1.28.1 音频编解码芯片

VS1053 是一颗单片 OGG/MP3/AAC/WMA/MIDI 音频解码器，通过 patch 可以实现 FLAC 的解码，同时该芯片可以支持 IMA ADPCM 编码，通过 patch 可以实现 OGG 编码。相比它的前辈：VS1003，VS1053 性能提升了不少，比如支持 OGG 编解码，支持 FLAC 解码，同时音质上也有比较大的提升，还支持空间效果设置。

图中 MP3_LEFT/MP3_RIGHT 这两个信号是 VS1053 的音频输出接口，输出到耳机/板载喇叭（详见：2.1.27 节）。VS1053 通过 7 根线连接到 MCU，VS_MISO/V_S_MOSI/V_S_SCK/V_S_XCS/V_S_XDCS/V_S_DREQ/V_S_RST 这 7 根线分别连接到 MCU 的 PA6/PA7/PA5/PF7/PF6/PC13/PE6 上，VS1053 通过 STM32 的 SPI1 访问，和 CH579Q 共用 SPI1。

图中的 SPK_CTRL 连接在 VS1053 的 GPIO4 上面，用于控制 HT6872 是否工作，从而控制板载喇叭是否出声，要让板载喇叭发声，必须通过软件控制 VS1053 的 GPIO4 输出高电平，否则板载喇叭关闭。

2.1.29 电源

正点原子战舰 STM32 开发板板载的电源供电部分，其原理图如图 2.1.29.1 所示：

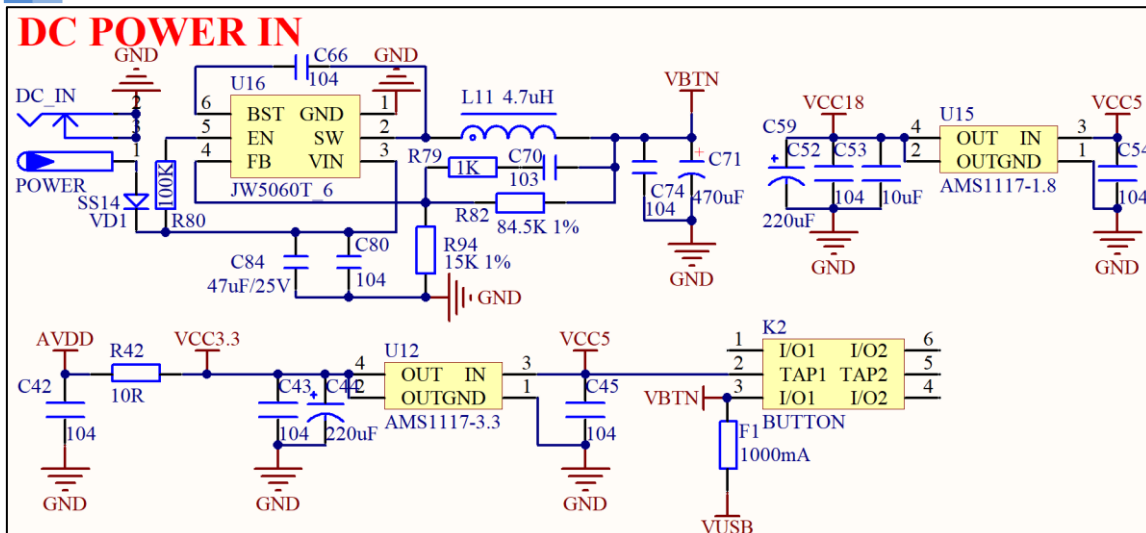


图 2.1.29.1 电源

图中，总共有 3 个稳压芯片：U12/U15/U16，DC_IN 用于外部直流电源输入，范围是 DC6~15V，输入电压经过 U16 DC-DC 芯片转换为 5V 电源输出，其中 VD1 是防反接二极管，避免外部直流电源极性搞错的时候，烧坏开发板。K2 为开发板的总电源开关，F1 为 1000ma 自恢复保险丝，用于保护 USB。U12 为 3.3V 稳压芯片，给开发板提供 3.3V 电源，而 U15 则是 1.8V 稳压芯片，供 VS1053 的 CVDD 使用。

这里还有 USB 供电部分没有列出来，其中 VUSB 就是来自 USB 供电部分，我们将在 2.1.31 节进行介绍。

2.1.30 电源输入输出接口

正点原子战舰 STM32 开发板板载了两组简单电源输入输出接口，其原理图如图 2.1.30.1 所示：

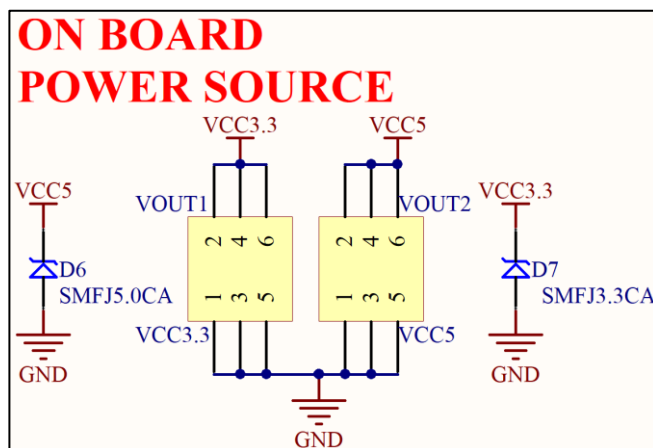


图 2.1.30.1 电源

图中，VOUT1 和 VOUT2 分别是 3.3V 和 5V 的电源输入输出接口，有了这 2 组接口，我们可以通过开发板给外部提供 3.3V 和 5V 电源了，虽然功率不大（最大 1000ma），但是一般情况都够用了，大家在调试自己的小电路板的时候，有这两组电源还是比较方便的。同时这两组端口，也可以用来由外部给开发板供电。

图中 D6 和 D7 为 TVS 管，可以有效避免 VOUT 外接电源/负载不稳的时候（尤其是开发板外接电机/继电器/电磁阀等感性负载的时候），对开发板造成的损坏。同时还能一定程度防止外

接电源接反，对开发板造成的损坏。

2.1.31 USB 串口

正点原子战舰 STM32 开发板板载了一个 USB 串口，其原理图如图 2.1.31.1 所示：

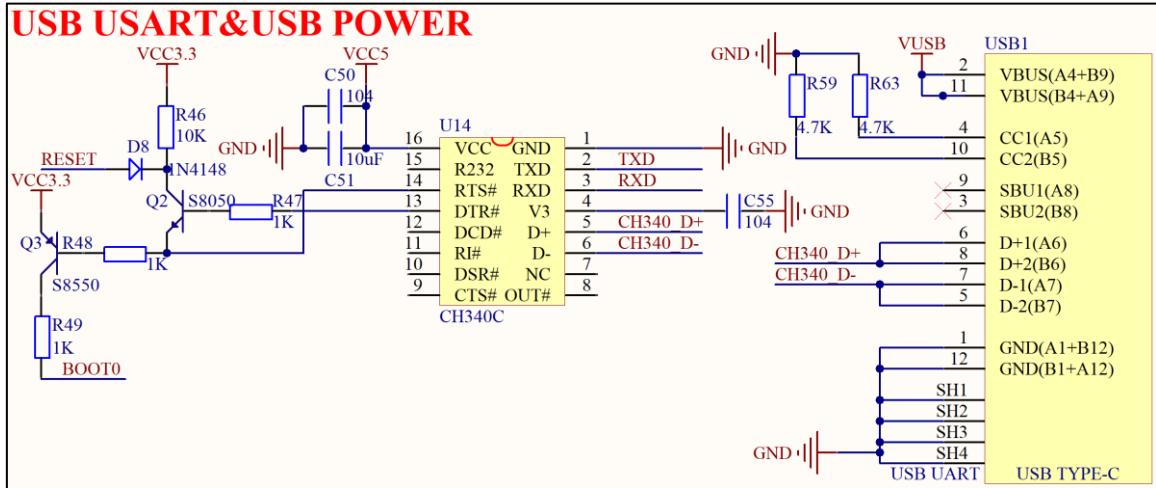


图 2.1.32.1 USB 串口

USB 转串口芯片，我们选择的是 CH340C，无需外部晶振，是 CH340G 的升级版，非常好用。

图中 Q2 和 Q3 的组合构成了我们开发板的一键下载电路，**只需要在 flymcu 软件设置：DTR 的低电平复位，RTS 高电平进 BootLoader**。就可以一键下载代码了，而不需要手动设置 B0 和按复位了。其中，RESET 是开发板的复位信号，BOOT0 则是启动模式的 B0 信号。

一键下载电路的具体实现过程：首先，mcuisp 控制 DTR 输出低电平，则 DTR_N 输出高，然后 RTS 置高，则 RTS_N 输出低，这样 Q3 导通了，BOOT0 被拉高，即实现设置 BOOT0 为 1，同时 Q2 也会导通，STM32F1 的复位脚被拉低，实现复位。然后，延时 100ms 后，mcuisp 控制 DTR 为高电平，则 DTR_N 输出低电平，RTS 维持高电平，则 RTS_N 继续为低电平，此时 STM32F1 的复位引脚，由于 Q2 不再导通，变为高电平，STM32F1 结束复位，但是 BOOT0 还是维持为 1，从而进入 ISP 模式，接着 mcuisp 就可以开始连接 STM32F1，下载代码了，从而实现一键下载。

USB_UART 是一个 USB TypeC 座，提供 CH340C 和电脑通信的接口，同时可以给开发板供电，VUSB 就是来自电脑 USB 的电源，USB_UART 是本开发板的主要供电口。

2.2 开发板使用注意事项

为了让大家更好的使用正点原子战舰 STM32 开发板，我们在这里总结该开发板使用的时候尤其要注意的一些问题，希望大家在使用的时候多多注意，以减少不必要的问题。

- 1，开发板一般情况是由 USB_UART 口供电，在第一次上电的时候由于 CH340C 在和电脑建立连接的过程中，导致 DTR/RTS 信号不稳定，会引起 STM32 复位 2~3 次左右，这个现象是正常的，后续按复位键就不会出现这种问题了。
- 2，1 个 USB 供电最多 500mA，且由于导线电阻存在，供到开发板的电压，一般都不会有 5V，如果使用了很多大负载外设，比如 4.3 寸屏、7 寸屏、网络、摄像头模块等，那么可能引起 USB 供电不够，所以如果是使用 4.3 屏/7 寸屏的朋友，或者同时用到多个模块的时候，**建议大家使用一个独立电源供电**。如果没有独立电源，建议可以同时插 2 个 USB 口，并插上 JTAG，这样供电可以更足一些。

- 3, JTAG 接口有几个信号 (JTDI/JTDO/JTRST) 被 GBC_LED (ATK MODULE) / OV_WEN (摄像头模块) / OV_RCLK (摄像头模块) 占用了, 所以在调试这些模块的时候, 请大家选择 SWD 模式, 其实**最好就是一直用 SWD 模式**。
- 4, 当你想使用某个 IO 口用作其他用处的时候, 请先看看开发板的原理图, 该 IO 口是否有连接在开发板的某个外设上, 如果有, 该外设的这个信号是否会对你的使用造成干扰, 先确定无干扰, 再使用这个 IO。比如 PB8 就不怎么适合再用做其他输出, 因为他接了蜂鸣器, 如果你输出高电平就会听到蜂鸣器的叫声了。
- 5, 开发板上的跳线帽比较多, 大家在使用某个功能的时候, 要先查查这个是否需要设置跳线帽, 以免浪费时间。
- 6, 当液晶显示白屏的时候, 请先检查液晶模块是否插好 (拔下来重新插试试), 如果还不行, 可以通过串口看看 LCD ID 是否正常, 再做进一步的分析。

至此, 本手册的实验平台 (正点原子战舰 STM32 开发板) 的硬件部分就介绍完了, 了解了整个硬件对我们后面的学习会有很大帮助, 有助于理解后面的代码, 在编写软件的时候, 可以事半功倍, 希望大家细读! 另外正点原子开发板的其他资料及教程更新, 都可以在技术论坛 www.openedv.com/forum.php 下载到, 大家可以经常去这个论坛获取更新的信息。