

MiniPro STM32H750

硬件参考手册

V1.0

—正点原子 MiniPro STM32H750 开发板教程

修订历史:

版本	日期	修改内容
V1.0	2021/06/27	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 MiniPro STM32H750 开发板资源初探	7
1.1.1 MiniPro STM32H750 硬件设计特点	7
1.1.2 MiniPro STM32H750 硬件基本参数	7
1.1.3 MiniPro STM32H750 硬件资源分布	8
1.1.4 MiniPro STM32H750 硬件资源列表	9
1.2 MiniPro STM32H750 开发板资源说明	10
1.2.1 硬件资源说明	10
1.2.2 MiniPro STM32H750 IO 引脚分配	14
1.3 MiniPro STM32H750 对比说明	17
第二章 实验平台硬件资源详解	18
2.1 开发板原理图详解	18
2.1.1 MCU	18
2.1.2 引出 IO 口	19
2.1.3 USB 串口/串口 1 选择接口	20
2.1.4 JTAG/SWD	21
2.1.5 LCD 模块接口	21
2.1.6 复位电路	22
2.1.7 启动模式设置接口	22
2.1.8 EEPROM	23
2.1.9 SPI FLASH (QSPI FLASH)	23
2.1.10 温湿度传感器接口	24
2.1.11 光敏传感器	24
2.1.12 红外接收头	25
2.1.13 无线模块接口	25
2.1.14 LED	25
2.1.15 按键	26
2.1.16 OLED/摄像头模块接口	26
2.1.17 有源蜂鸣器	27
2.1.18 TF 卡接口	28
2.1.19 ATK 模块接口	28
2.1.20 多功能端口	28

2.1.21 电源	29
2.1.22 电源输入输出接口	30
2.1.23 USB 通信接口	30
2.1.24 USB 串口	31
2.2 开发板使用注意事项	32

内容简介

本手册主要介绍 MiniPro STM32H750 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 MiniPro STM32H750 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《STM32H750 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 MiniPro STM32H750 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, MiniPro STM32H750 开发板资源初探；
- 1.2, MiniPro STM32H750 开发板资源说明；

1.1 MiniPro STM32H750 开发板资源初探

自从 2011 年上市以来，正点原子 MiniSTM32 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 10W 套。2018 年 ST 推出了高性价比的 STM32H750 系列，为满足大家对高性能的需求，因此在 MiniSTM32F103 的基础上，我们改用最新的 STM32H750 系列 MCU，差异化设计（具体差异见 1.3 节），推出了这款 MiniPro STM32H750 开发板。

下面我们开始介绍 MiniPro STM32H750。

1.1.1 MiniPro STM32H750 硬件设计特点

MiniPro STM32H750 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。我们引出了数十个 IO 口，可以极大的方便大家扩展及使用。
- 3) **资源充足。**主芯片采用 STM32H750VBT6，自带 128KB FLASH 和 1060KB SRAM，并外扩 16M 字节 FLASH，满足大数据存储需求。
- 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，MiniPro STM32H750 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 80%（数量）。

1.1.2 MiniPro STM32H750 硬件基本参数

MiniPro STM32H750 硬件基本参数如表 1.1.2.1 所示：

项目	说明
产品型号	ATK-DNH750M V1
CPU	STM32H750VBT6, LQFP100
引出 IO	46 个
外形尺寸	12.2mm*8mm
工作电压	5V (USB)
工作电流	1mA~155mA ¹ (@5V)
工作温度	0℃~+70℃

表 1.1.2.1 MiniPro STM32H750 硬件基本参数

注 1：1mA 对应 CPU 在复位情况下，裸板的工作电流；155mA 对应 CPU 正常运行时裸板的工作电流。

1.1.3 MiniPro STM32H750 硬件资源分布

MiniPro STM32H750 的硬件资源分布如图 1.1.3.1 所示：

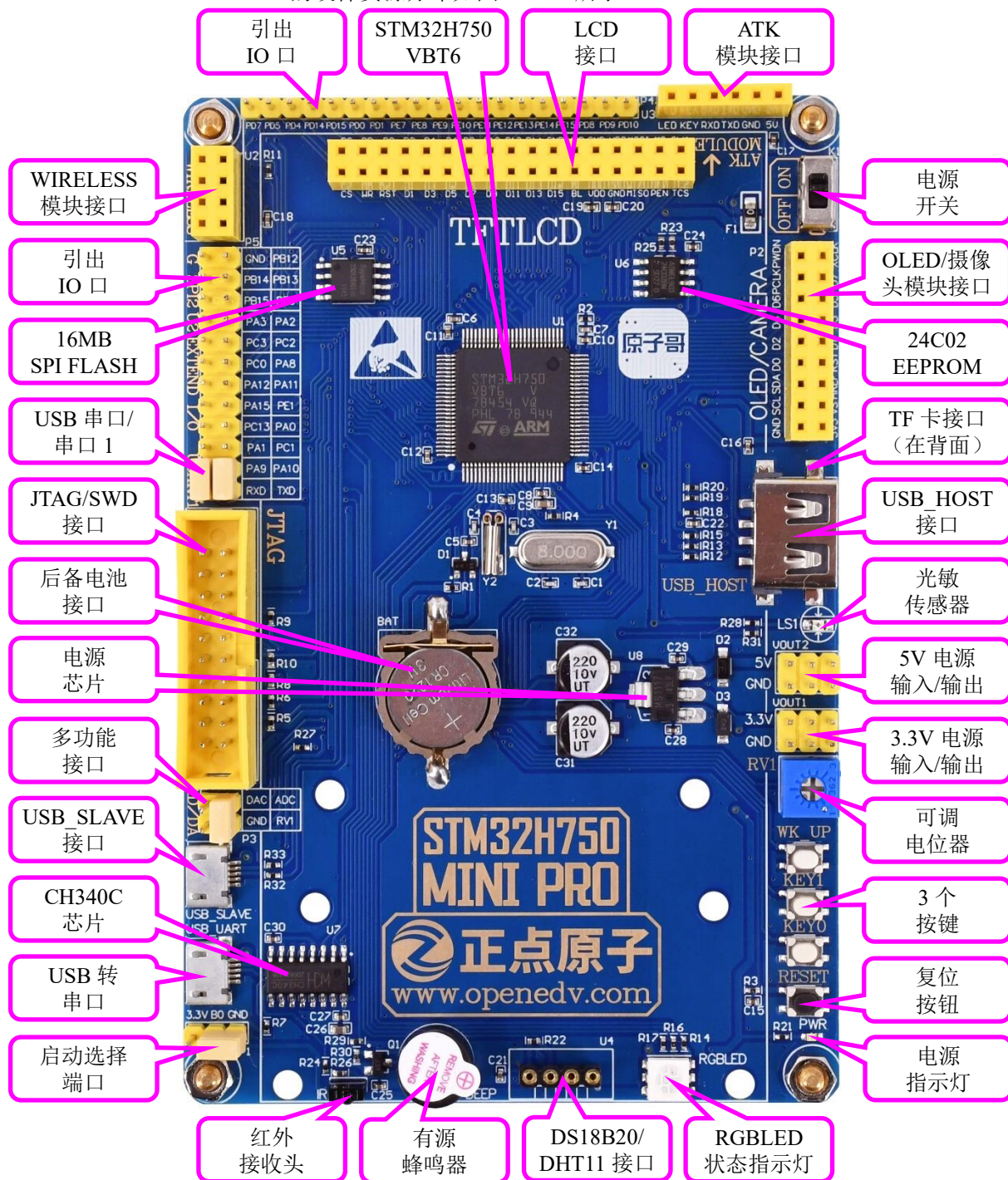


图 1.1.3.1 MiniPro STM32H750 硬件资源分布图

1.1.4 MiniPro STM32H750 硬件资源列表

MiniPro STM32H750 的硬件资源列表如表 1.1.4.1 所示：

资源	数量	说明
CPU	1 个	STM32H750VBT6; FLASH: 128KB; SRAM: 1060KB;
SPI FLASH	1 个	16MB
EEPROM	1 个	2Kb (256B)
电源指示灯	1 个	蓝色
状态指示灯	1 个	RGB 三色 (红色、绿色、蓝色)
复位按键	1 个	用于 MCU&LCD 的复位
功能按键	3 个	KEY0、KEY1、KEY_UP (具备唤醒功能)
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
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
无线模块接口	1 个	可以接 NRF24L01 等无线模块
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
USB 主机接口	1 个	用于 USB_HOST (主机) 通信
TF 卡接口	1 个	用于接 TF 卡
JTAG/SWD 调试口	1 个	用于仿真调试、下载代码等
多功能接口	1 组	用于 DAC/ADC/TPAD/RV1 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
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	45 个	用于扩展使用

表 1.1.4.1 MiniPro STM32H750 的硬件资源列表

1.2 MiniPro STM32H750 开发板资源说明

MiniPro STM32H750 资源说明,我们将分为两个部分:硬件资源说明和 MiniPro STM32H750 IO 引脚分配。

1.2.1 硬件资源说明

这里我们详细介绍 MiniSTM32 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(U2),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. 引出 IO 口(总共有 2 处)

这是开发板 IO 引出端口,总共两组主 IO 引出口: P4 和 P5, P4 采用 1*19 单排针引出, P3 采用 2*12 双排针引出, 总共引出 40 个 IO 口。另外, P3 和 JTAG 接口分别引出 2 个和 4 个 IO 口, 这样通过排针引出的 IO 总数为: 46 个。

如果使用的时候 IO 口还不够用的话,可以考虑自行把摄像头模块接口座子换成排针,这样还可以引出: 14 个 IO 口。

3. 16MB SPI FLASH

这是开发板外扩的 SPI FLASH 芯片(U5),容量为 128Mbit,也就是 16M 字节,可用于存储程序和用户数据,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。

注意: SPI FLASH 采用 QSPI 接口,支持地址映射,该 FLASH 同时存储程序和用户数据。

4. USB 串口/串口 1

这是 USB 串口同 STM32H750VBT6 的串口 1 进行连接的接口(P5),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 MiniSTM32 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 MiniSTM32 开发板上。

5. JTAG/SWD 接口

这是 MiniSTM32 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!**

6. 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。

7. 电源芯片

这是开发板的电源稳压芯片，型号为：AMS1117-3.3。因为 STM32 是 3.3V 供电的，所以我们需要将 USB 的 5V 电压转换为 3.3V，这个芯片就是将 5V 转换为 3.3V 的线性稳压芯片。

8. 多功能端口

这是 1 个由 4 个排针组成的一个接口（P3）。不过大家可别小看这 4 个排针，这组端口通过组合可以实现的功能有：ADC 采集、DAC 输出、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。

9. USB SLAVE

这是开发板板载的一个 Micro 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 Micro 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 Micro 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。

10. CH340C 芯片

这是开发板板载的 USB 转串口芯片，型号为：CH340C。有了这个芯片，我们就可以实现 USB 转串口，从而能实现 USB 下载代码，串口通信等。

11. USB 转串口

这是开发板板载的另外一个 Micro 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。

12. 启动选择端口

这是开发板板载的启动模式选择端口（P1），不同于 STM32F103 系列，STM32H7 系列只有 1 个 BOOT 设置脚，用于选择复位后 STM32 的启动模式。

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 2.1.7 小节。

13. 红外接收头

这是开发板的红外接收头（IR），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。

MiniSTM32 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

14. 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。

15. DS18B20/DHT11 接口

这是开发板的一个复用接口（U4），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，使用上是十分方便灵活的。

16. RGBLED 状态指示灯

这是开发板板载的 1 个 RGBLED 灯，由红色（LEDR）、绿色（LEDG）、蓝色（LEDB）三个 LED 组成，通过 3 个 IO 口控制，经过组合调光，理论上该 LED 可以实现各种颜色的显示。

我们一般的应用 3 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。MiniPro STM32H750 开发板几乎每个实例都使用了 LED 来指示程序的运行状态。

17. 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。

18. 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。

19. 3 个按键

这是开发板板载的 3 个机械式输入按键（KEY0、KEY1 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 2 个是普通按键，可以用于人机交互的输入，这 2 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 和 KEY1 是低电平有效，大家在使用的时候留意一下。

20. 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 P1 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。

21. 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 MiniPro STM32H750 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）。

22. 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）。

23. 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。

24. USB_HOST 接口

这是开发板板载的一个 USB 主机接口(USB-A 座)，由于 STM32H750 的 USB 是支持 HOST 的，所以我们可以通过这个 USB-A 座来连接：U 盘、USB 鼠标、USB 键盘、USB 手柄等 USB 从设备，从而实现 USB 主机功能。

注意，由于 USB HOST 和 USB SLAVE 是共用 PA11 和 PA12，所以两者不可以同时使用。

25. TF 卡接口（在背面）

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

26. 24C02 EEPROM

这是开发板板载的 EEPROM 芯片（U6），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。

27. 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（P2），如果是 OLED 模块，靠下插即可（上边两个孔位悬空）。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。

28. 电源开关

这是开发板板载的电源开关（K1）。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。

29. 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（U3），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。

30. 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。

31. STM32H750VBT6

这是开发板的核心芯片（U1），型号为：STM32H750VBT6。该芯片内部资源非常丰富，详见表 1.2.1.1：

STM32H750VBT6 资源					
内核	Cortex M7	低功耗定时器	5	USART	9
主频	480Mhz	通用定时器	10	CAN	2
FLASH	128KB	高级定时器	2	SDIO	2
SRAM	1060KB	16 位 ADC	3	FMC	1
封装	LQFP100	ADC 通道数	16	DMA	4
IO 数量	82	12 位 DAC	2	RTC	1
工作电压	3.3V	SPI	6	IIC	4
USB OTG	2	QUADSPI	1	SAI	4
ETH MAC	100M	JPEG Codec	支持	DCMI	支持

表 1.2.1.1 STM32H750VBT6 内部资源表

1.2.2 MiniPro STM32H750 IO 引脚分配

为了让大家更快更好的使用我们的 MiniPro STM32H750 开发板，这里特地将 MiniPro STM32H750 开发板主芯片：STM32H750VBT6 的 IO 资源分配做了一个总表，以便大家查阅。MiniPro STM32H750 的 IO 引脚分配总表如表：1.2.2.1 所示：

MiniPro STM32H750 开发板 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
22	PA0	WK_UP		Y	1, 按键 KEY_UP 2, 可以做待机唤醒脚(WKUP)
23	PA1	KEY0		Y	接按键 KEY0
24	PA2	GBC_RX		N	1, ATK MODULE 接口的 RXD 脚 2, 串口 2 的 TX 脚
25	PA3	GBC_RX		N	1, ATK MODULE 接口的 TXD 脚 2, 串口 2 的 RX 脚
28	PA4	DCMI_HREF	STM_DAC	Y	1, DAC_OUT1 输出脚 2, OLED/CAMERA 接口的 HREF 脚
29	PA5	STM_ADC		Y	ADC 输入引脚
30	PA6	DCMI_PCLK		Y	OLED/CAMERA 接口的 PCLK 脚
31	PA7	DCMI_RESET		N	OLED/CAMERA 接口的 RESET 脚
67	PA8	REMOTE_IN	DCMI_XCLK	N	1, 接 LF0038GKLL-1 红外接收头 2, OLED/CAMERA 接口的 XCLK 脚
68	PA9	USART1_TX		Y	串口 1 TX 脚，默认连接 CH340 的 RX (P5 设置)
69	PA10	USART1_RX		Y	串口 1 RX 脚，默认连接 CH340 的 TX (P5 设置)
70	PA11	USB_DM		Y	USB D-引脚 (P11 设置)
71	PA12	USB_DP		Y	USB D+引脚 (P11 设置)
72	PA13	JTMS	SWDIO	N	JTAG/SWD 仿真接口, 没接任何外设
76	PA14	JTCK	SWDCLK	N	JTAG/SWD 仿真接口, 没接任何外设
77	PA15	JTDI	KEY1	N	1, JTAG 仿真口 (JTDI) 2, KEY1
34	PB0	T_SCK		Y	TFTLCD 接口触摸屏 SCK 信号
35	PB1	T_PEN		Y	TFTLCD 接口触摸屏 PEN 信号
36	PB2	QSPI_BK1_CLK		N	QSPI 时钟信号, 接 NOR FLASH
89	PB3	JTDO	T_MOSI	N	1, JTAG 仿真口 (JTDO) 2, T_MOSI 接口的 SDA 脚
90	PB4	JTRST	LEDR	N	1, JTAG 仿真口 (JTRST) 2, LEDR
91	PB5	LCD_BL		Y	TFTLCD 接口背光控制脚
92	PB6	QSPI_BK1_NCS		N	QSPI 片选信号, 接 NOR FLASH
93	PB7		DCMI_VSYNC	Y	OLED/CAMERA 接口的 VSYNC 脚
95	PB8		DCMI_D6	Y	OLED/CAMERA 接口的 D6 脚
96	PB9		DCMI_D7	Y	OLED/CAMERA 接口的 D7 脚

46	PB10	IIC_SCL	DCMI_SCL	N	1, 接 24C02 2, OLED/CAMERA 接口的 SCL 脚
47	PB11	IIC_SDA	DCMI_SDA	N	1, 接 24C02 2, OLED/CAMERA 接口的 SDA 脚
51	PB12	SPI2_CS		Y	没有连接任何器件
52	PB13	SPI2_SCK		Y	接 WIRELESS 接口的 SCK 信号
53	PB14	SPI2_MISO		Y	接 WIRELESS 接口的 SCK 信号
54	PB15	SPI2_MOSI		Y	接 WIRELESS 接口的 SCK 信号
15	PC0	NRF_CE		Y	WIRELESS 接口 CE 引脚
16	PC1		LIGHT_SENSOR	N	接光敏二极管
17	PC2	GBC_KEY		Y	ATK-MODULE 接口的 KEY 引脚
18	PC3	NRF_IRQ	GBC_LED	Y	1, WIRELESS 接口 IRQ 信号 2, ATK-MODULE 接口的 LED 引脚
32	PC4	DCMI_PWDN		Y	OLED/CAMERA 接口的 PWDN 脚
33	PC5	T_CS		Y	TFTLCD/RGBLCD 接口触摸屏 CS 信号
63	PC6		DCMI_D0	Y	OLED/CAMERA 接口的 D0 脚
64	PC7		DCMI_D1	Y	OLED/CAMERA 接口的 D1 脚
65	PC8	SDIO_D0	DCMI_D2	N	1, TF 卡接口的 D0 2, OLED/CAMERA 接口的 D2 脚
66	PC9	SDIO_D1	DCMI_D3	N	1, TF 卡接口的 D1 2, OLED/CAMERA 接口的 D3 脚
78	PC10	SDIO_D2		N	TF 卡接口的 D2
79	PC11	SDIO_D3	DCMI_D4	N	1, TF 卡接口的 D3 2, OLED/CAMERA 接口的 D4 脚
80	PC12	SDIO_CLK		Y	TF 卡接口的 SCK
7	PC13	1WIRE_DQ		N	接单总线接口(U11), 即 DHT11/DS18B20
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 IO
81	PD0	FMC_D2		Y	FMC 总线数据线 D2
82	PD1	FMC_D3		Y	FMC 总线数据线 D3
83	PD2	SDIO_CMD		N	TF 卡接口的 CMD
84	PD3		DCMI_D5	Y	OLED/CAMERA 接口的 D5 脚
85	PD4	FMC_NOE		Y	FMC 总线 NOE(RD) (LCD)
86	PD5	FMC_NWE		Y	FMC 总线 NWE(WR) (LCD)
87	PD6	T_MISO		Y	TFTLCD/RGBLCD 接口触摸屏 MOSI 信号
88	PD7	FMC_NE1		Y	FMC 总线的片选信号 1, 为 TFTLCD 片选信号
55	PD8	FMC_D13		Y	FMC 总线数据线 D13 (LCD)
56	PD9	FMC_D14		Y	FMC 总线数据线 D14 (LCD)
57	PD10	FMC_D15		Y	FMC 总线数据线 D15 (LCD)
58	PD11	QSPI_BK1_IO0		N	QSPI 数据线, 接 NOR FLASH
59	PD12	QSPI_BK1_IO1		N	QSPI 数据线, 接 NOR FLASH

60	PD13	QSPI_BK1_IO3		N	QSPI 数据线，接 NOR FLASH
61	PD14	FMC_D0		Y	FMC 总线数据线 D0 (LCD)
62	PD15	FMC_D1		Y	FMC 总线数据线 D1 (LCD)
97	PE0		NRF_CS	Y	WIRELESS 接口 CS 信号
98	PE1			Y	引出到 P5
1	PE2	QSPI_BK1_IO2		N	QSPI 数据线，接 NOR FLASH
2	PE3	FMC_A19		Y	FMC 总线地址线 A19
3	PE4	BEEP		N	蜂鸣器控制线
4	PE5	LEDB		N	LEDB
5	PE6	LEDG		N	LEDG
37	PE7	FMC_D4		Y	FMC 总线数据线 D4 (LCD)
38	PE8	FMC_D5		Y	FMC 总线数据线 D5 (LCD)
39	PE9	FMC_D6		Y	FMC 总线数据线 D6 (LCD)
40	PE10	FMC_D7		Y	FMC 总线数据线 D7 (LCD)
41	PE11	FMC_D8		Y	FMC 总线数据线 D8 (LCD)
42	PE12	FMC_D9		Y	FMC 总线数据线 D9 (LCD)
43	PE13	FMC_D10		Y	FMC 总线数据线 D10 (LCD)
44	PE14	FMC_D11		Y	FMC 总线数据线 D11 (LCD)
45	PE15	FMC_D12		Y	FMC 总线数据线 D12 (LCD)

表 1.2.2.1 MiniPro STM32H750 IO 资源分配总表

表 1.2.2.1 中，引脚栏即 STM32H750VBT6 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 MiniPro STM32H750 对比说明

正点原子 MiniPro H750 开发板相对于 Mini F103 版本，主要变化如表 1.3.1 所示：

编号	对比项	MiniPro H750 VS Mini F103 开发板		说明
		Mini F103	MiniPro H750	
1	主芯片	STM32F103RCT6	STM32H750VBT6	性能暴涨
2	SPI FLASH	25Q64	25Q128	容量更大
3	LED	2 个单色	1 个 RGB 全彩	应用更广
4	摄像头接口	无	有	新增功能
5	USB HOST 接口	无	有	新增功能
6	光敏传感器	无	有	新增功能
7	蜂鸣器	无	有	新增功能
8	板子尺寸	10cm * 8cm	12.2cm * 8cm	尺寸更大

表 1.3.1 MiniH750 VS MiniF103 对比表

从表 1.3.1 可以看出，MiniPro STM32H750 开发板在 Mini STM32F103 版本的基础上更换了更强大的主芯片，并添加了更多的接口和外设，能满足用户对高性能应用的使用需求。

大家可以根据自己的实际情况，来选择使用 Mini F103 还是 MiniPro H750。

第二章 实验平台硬件资源详解

本章，我们将节将向大家详细介绍正点原子 MiniPro STM32H750 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 MiniPro STM32H750 开发板选择的是 STM32H750VBT6 作为 MCU，该芯片功能非常强大，它拥有的资源包括：1060KB SRAM、128KB FLASH（支持运行 QSPI FLASH 代码）、高达 480Mhz 主频、内置硬件 JPEG 编解码器、带 FMC 外部总线等，详见 1.2 节的表 1.2.1.1。

该芯片性价比极高，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：

MCU_A			U1A
WK_UP	PA0	22	PA0/TIM2_CH1/TIM2_ETR/TIM5_CH1/TIM5_ETR/TIM15_BKIN/USART2_CTS_NSS/UART4_TX/SDMMC2_CMD/SAI2_SD_B0/ETH_MII_CRS/ADC_INP16/WKUP0
KPY0	PA1	23	PA1/TIM2_CH2/TIM2_ETR/TIM5_CH2/USART2_RTS/UART4_RX/QUADSPI_BK1_IO3/SAI2_MCK_B0/ETH_MII_RX_CLK/ETH_RMII_RET_CLK/LCD_R2/ADC1_INN16/INP17
GPIO_RX	PA2	24	PA2/TIM2_CH3/TIM5_CH3/PHM1_OUT/TIM15_CH1/USART7_TX/SAI2_SCK_B0/ETH_MDIO/MDIOS_MDIO1/CD_R1/ADC12_INP11/WKUP1
GPIO_TX	PA3	25	PA3/TIM2_CH4/TIM5_CH4/PHM1_OUT/TIM15_CH2/USART2_RX1/CD_R2/OTG_HS_ULPI_D0/ETH_MII_C0/LCD_B5/ADC12_INP15
DCMI_HREF	STM_DAC	PA4	PA4/TIM1_ETR/SP1_NSS/DS1_WS/SP1_NSS/DS1_WS/USART2_CK/SP6_NSS/OTG_HS_S0/LCD_B1/SDMMC1_D1/CD_VSYNC/ADC12_INP18/DAC1_OUT1
DCMI_PCLK	STM_ADC	PA5	PA5/TIM2_CH1/TIM2_ETR/TIM5_CH1/SP1_CK/SP6_SCK/OTG_HS_ULPI_C0/LCD_B4/ADC12_DN18/BDP/DAC1_OUT2
DCMI_RESET	REMOTE_IN	PA6	PA6/TIM1_BKIN/TIM3_CH1/TIM5_BKIN/SP1_MISO/DS1_SDI/SP6_MISO/TIM13_CH1/TIM8_BKIN/COMP12/MDIOS_MDC/TIM1_BKIN/COMP12/DCMI_PIXCLK/LCD_G2/ADC12_INP3
DCMI_XCLK	USART1_TX	PA7	PA7/TIM1_CH1/TIM3_CH2/TIM8_CH1/SP1_MOSI/DS1_SDO/SP6_MOSI/TIM14_CH1/ETH_MII_RX_DV/ETH_RMII_CRS_DV/FMC_SDNF/ADC12_INN3_INP7/OPAMP1_VINM
	USART1_RX	PA8	PA8/MC0/TIM4_CH1/HTIM4_CH2/TIM8_BKIN/DS1_SCK/USART1_CK/OTG_FS_S0/UART7_RX/TIM8_BKIN2/COMP12/LCD_B5/LCD_B6
	USB_DM	PA9	PA9/TIM1_CH2/HTIM4_CH2/LPUART1_TX1/CD_SBA/SP2_SCK/DS2_CK/USART1_TX/CD_CAN1_RX/CD_MODE/ETH_TX_ER/DCMI_D0/LCD_R5/OTG_FS_VBUS
	USB_DP	PA10	PA10/TIM1_CH3/HTIM4_CH2/LPUART1_RX/USART1_RX/CD_CAN1_TX/CD_MODE/OTG_FS_ID/MDIOS_MDIO1/LCD_B4/DCMI_D1/LCD_B1
	J1MS	PA11	PA11/TIM1_CH4/HTIM4_CH1/LPUART1_CTS/SP2_NSS/DS2_WS/UART4_RX/USART1_CTS_NSS/CD_CAN1_RX/OTG_FS_DM/LCD_R4
	J1CK	PA12	PA12/TIM1_ETR/HTIM4_CH1/LPUART1_RTS/SP2_SCK/DS2_CK/UART4_TX/USART1_RTS/SAI2_FS_B/CD_CAN1_TX/OTG_FS_DFL/CD_R5
	J1DI	PA13	PA13/TIM5_SWDO
KEY1	J1DI	PA14	PA14/IOCS_SWICK
	J1DI	PA15	PA15/HTIM2_CH1/TIM2_ETR/HTIM4_CH1/HTIM4_CH2/SP1_NSS/DS1_WS/SP1_NSS/DS1_WS/SP6_NSS/UART4_RTS/UART7_TX
I_SCK	PH0	31	PH0/TIM1_CH2/TIM3_CH1/TIM8_CH2/DFSDM_CKOUT/UART4_CTS/LCD_R5/OTG_HS_ULPI_D0/ETH_MII_RXD0/LCD_G1/ADC12_INN5_INP9/OPAMP1_VINP/COMP1_INP
I_PEN	PH1	35	PH1/TIM1_CH3/TIM3_CH2/TIM8_CH3/DFSDM_DAT1N/LCD_R6/OTG_HS_ULPI_D1/ETH_MII_RXD1/LCD_G0/ADC12_INP5/COMP1_INM
QSPI_BK1_CLK	PH2	36	PH2/SAI1_D1/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
J1DO	PH3	39	PH3/HTIM2_CH2/HTIM4_CH1/SP1_CK/SP3_SCK/DS3_CK/SP6_SCK/SDMMC2_D2/UART7_RX
LEDR	LCD_B1	PH4	PH4/HTIM2_CH1/TIM4_CH2/HTIM4_CH1/HTIM4_CH2/SP1_MISO/DS1_SDI/SP6_MISO/DS1_SDI/SP6_MISO/SDMMC2_D3/UART7_TX
	QSPI_BK1_CS	PH5	PH5/TIM1_BKIN/TIM3_CH2/HTIM4_CH1/HTIM4_CH2/SC1/HTIM4_CH2/SC1/USART1_TX/LPUART1_TX/CD_CAN2_TX/QUADSPI_BK1_NSS/OTG_HS_ULPI_D7/ETH_PPS_OUT/FMC_SDCKE1/DCMI_D10/UART5_RX
DCMI_VSYNC	PH6	92	PH6/TIM7_CH1N/TIM4_CH2/HTIM4_CH2/SC1/HTIM4_CH2/SC1/USART1_TX/LPUART1_TX/CD_CAN2_TX/QUADSPI_BK1_NSS/OTG_HS_ULPI_D3/ETH_MII_RX_ER/LCD_G4
DCMI_D6	PH8	95	PH8/TIM7_CH1/TIM4_CH1/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
DCMI_D7	PH9	96	PH9/TIM7_CH1/TIM4_CH1/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
DCMI_D8	PH10	46	PH10/TIM7_CH2/TIM4_CH2/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
DCMI_SDA	PH11	47	PH11/TIM7_CH1/TIM4_CH1/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
DCMI_SCL	PH12	51	PH12/TIM7_CH1/TIM4_CH1/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
SP2_CS	PH13	52	PH13/TIM7_CH1/TIM4_CH1/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
SP2_SCK	PH14	53	PH14/TIM7_CH1/TIM4_CH1/DFSDM_CKIN/SAI1_SD_A/SP3_MOSI/DS3_SDO/SAI1_SD_A/QUADSPI_CLK/SAI1_D1/ETH_TX_ER/COMP1_INP/RTC_OUT
SP2_MISO	PH15	54	PH15/RTC_REFIN/TIM1_CH3N/TIM2_CH2/TIM8_CH3N/SAI1_RX/SP2_MOSI/DS2_SDO/DFSDM_DAT1N/USART1_CTS_NSS/CD_CAN2_TX/OTG_HS_ULPI_D6/ETH_MII_TXD0/OTG_HS_ID/TIM1_BKIN/COMP12/UART5_RX
SP2_MOSI	PH15	54	PH15/RTC_REFIN/TIM1_CH3N/TIM2_CH2/TIM8_CH3N/SAI1_RX/SP2_MOSI/DS2_SDO/DFSDM_DAT1N/USART1_CTS_NSS/CD_CAN2_TX/OTG_HS_ULPI_D6/ETH_MII_TXD0/OTG_HS_ID/TIM1_BKIN/COMP12/UART5_RX
NRF_CE	PC0	15	PC0/DFSDM_CKIN0/DFSDM_DAT1N/SAI2_FS_B/OTG_HS_ULPI_STP/FMC_SDNW/LCD_R5/ADC12_INP10
LIGHT_SENSOR	PC1	16	PC1/SAI1_D1/DFSDM_DAT1N0/DFSDM_CKIN4/SP2_MOSI/DS2_SDO/SAI1_SD_A/SAI1_SD_A/SDMMC2_CK/SAI1_D1/ETH_MDIO/MDIOS_MDC/ADC12_INN10_INP11/WKUP5
GPIO_AE	PC2	17	PC2/DFSDM_CKIN1/SP2_MISO/DS2_SDO/DFSDM_CKOUT/OTG_HS_ULPI_DIR/ETH_MII_TXD2/FMC_SDN0/ADC12_INN11_INP12
GPIO_LED	PC3	18	PC3/DFSDM_DAT1N1/SP2_MOSI/DS2_SDO/OTG_HS_ULPI_MX/ETH_MII_TX_CLK/FMC_SDN0/ADC12_INN12_INP13
DCMI_PWDN	PC4	32	PC4/DFSDM_CKIN2/DS1_MCK/SPDIFRX_IN2/ETH_MII_RXD0/ETH_RMII_RXD0/FMC_SDN0/ADC12_INP4/OPAMP1_VOUT/COMP1_INM
I_CS	PC5	33	PC5/SAI1_D3/DFSDM_DAT1N2/SPDIFRX_IN3/SAI1_D3/ETH_MII_RXD1/ETH_RMII_RXD1/FMC_SDN0/COMP1_OUT/ADC12_INN1_INP9/OPAMP1_VINM
DCMI_D0	SD1_D0	61	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D1	SD1_D1	62	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D2	SD1_D2	63	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D3	SD1_D3	64	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D4	SD1_D4	65	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D5	SD1_D5	66	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D6	SD1_D6	67	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D7	SD1_D7	68	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D8	SD1_D8	69	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D9	SD1_D9	70	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D10	SD1_D10	71	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D11	SD1_D11	72	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D12	SD1_D12	73	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D13	SD1_D13	74	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D14	SD1_D14	75	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D15	SD1_D15	76	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D16	SD1_D16	77	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D17	SD1_D17	78	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D18	SD1_D18	79	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D19	SD1_D19	80	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D20	SD1_D20	81	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D21	SD1_D21	82	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D22	SD1_D22	83	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D23	SD1_D23	84	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D24	SD1_D24	85	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D25	SD1_D25	86	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D26	SD1_D26	87	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D27	SD1_D27	88	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D28	SD1_D28	89	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D29	SD1_D29	90	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D30	SD1_D30	91	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D31	SD1_D31	92	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D32	SD1_D32	93	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D33	SD1_D33	94	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D34	SD1_D34	95	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D35	SD1_D35	96	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D36	SD1_D36	97	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D37	SD1_D37	98	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D38	SD1_D38	99	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D39	SD1_D39	100	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D40	SD1_D40	101	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D41	SD1_D41	102	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D42	SD1_D42	103	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D43	SD1_D43	104	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D44	SD1_D44	105	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D45	SD1_D45	106	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D46	SD1_D46	107	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D47	SD1_D47	108	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D48	SD1_D48	109	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D49	SD1_D49	110	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D50	SD1_D50	111	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D51	SD1_D51	112	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D52	SD1_D52	113	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D53	SD1_D53	114	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D54	SD1_D54	115	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D55	SD1_D55	116	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D56	SD1_D56	117	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D57	SD1_D57	118	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D58	SD1_D58	119	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D59	SD1_D59	120	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D60	SD1_D60	121	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D61	SD1_D61	122	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D62	SD1_D62	123	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D63	SD1_D63	124	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D64	SD1_D64	125	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D65	SD1_D65	126	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D66	SD1_D66	127	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D67	SD1_D67	128	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D68	SD1_D68	129	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D69	SD1_D69	130	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D70	SD1_D70	131	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D71	SD1_D71	132	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D72	SD1_D72	133	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D73	SD1_D73	134	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D74	SD1_D74	135	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D75	SD1_D75	136	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D76	SD1_D76	137	PC6/HTIM4_CH1/TIM4_CH1/TIM8_CH1/DFSDM_CKIN1/DS2_SCK/USART6_TX/SDMMC1_D0/HTIM4_CH1/HTIM4_CH2/SP2_MOSI/DS2_SDO/ADC12_INN12_INP13
DCMI_D77	SD1_D77	138	PC6/HTIM4_CH1/TIM4_CH

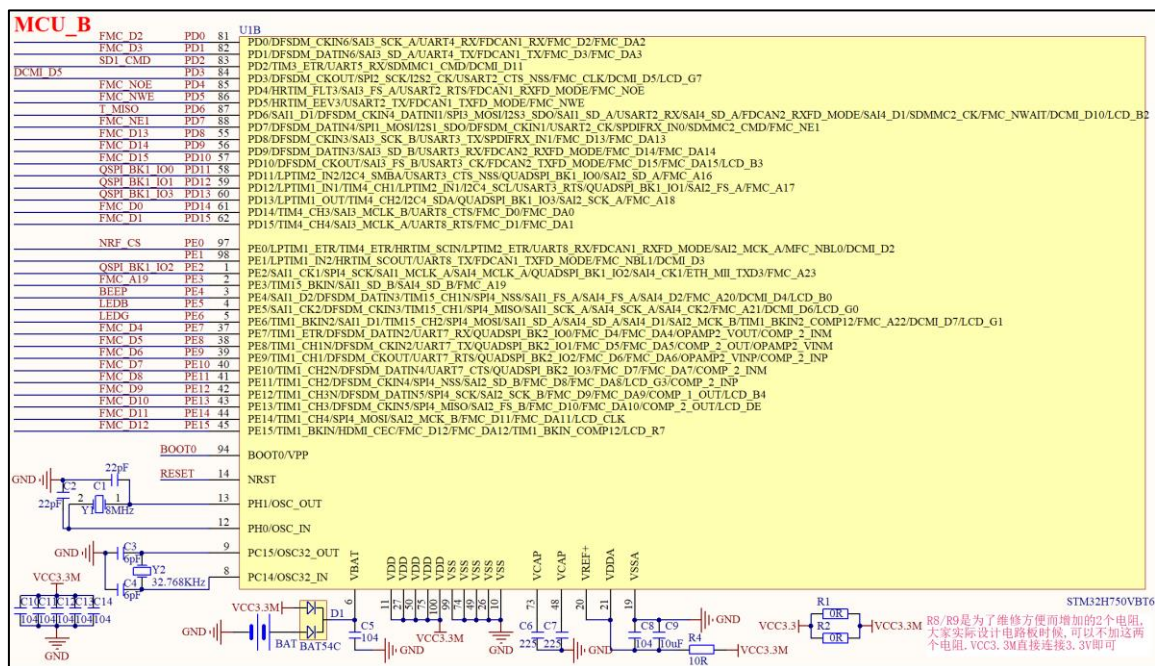


图 2.1.1.1-2 MCU 部分原理图 (B)

图中 U1 为我们的主芯片：STM32H750VBT6（原理图将其分成 A/B 两部分）。

这里主要讲解以下 2 个地方：

- 1，后备区域供电脚 VBAT 脚的供电采用 CR1220 纽扣电池和 VCC3.3 混合供电的方式，在有外部电源（VCC3.3）的时候，CR1220 不给 VBAT 供电，当外部电源断开时则由 CR1220 给其供电。这样，VBAT 总是有电的，以保证 RTC 的走时以及后备寄存器的内容不丢失。
- 2，图中的 R1 和 R2 用隔离 MCU 部分和外部的电源，这样的设计主要是考虑了后期维护，如果 3.3V 电源短路，可以断开这两个电阻，来确定是 MCU 部分短路，还是外部短路，有助于生产和维修。当然大家在自己的设计上，这两个电阻是完全可以去掉的。

2.1.2 引出 IO 口

正点原子 MiniPro STM32H750 通过排针引出了 46 个 IO 口，如图 2.1.2.1 所示：

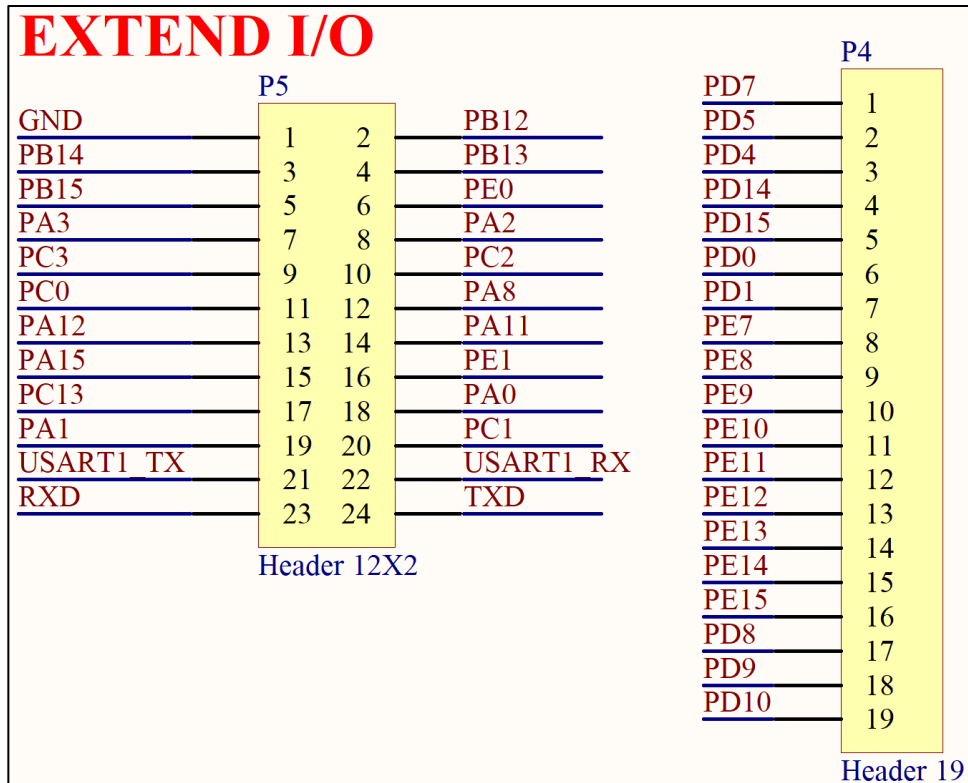


图 2.1.2.1 引出 IO 口

图中 P4 和 P5 为 MCU 主 IO 引出口，这两组排针共引出了 40 个 IO 口，STM32H750VBT6 总共有 82 个 IO，我们通过排针方式共引出了 46 个，这两组主引出排针，总共引出了 40 个 IO，剩下的 6 个 IO 口分别通过：P3（PA4&PA5）和 JTAG（PB3、PB4、PA13 和 PA14）两个接口引出。

如果大家在使用的时候需要扩展 IO 比较多，可以考虑把摄像头座换成排针，这样可以多引出 15 个 IO 口（合计 61 个 IO 口），不过一般我们开发板引出的 46 个 IO 口也是够用了的。

2.1.3 USB 串口/串口 1 选择接口

正点原子 MiniPro STM32H750 板载的 USB 串口和 STM32H750VBT6 的串口是通过 P5 的 21~24 脚连接起来的，如图 2.1.3.1 所示：

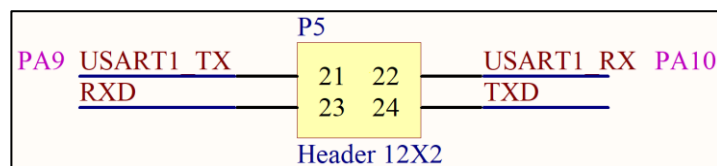


图 2.3.1.1 USB 串口/串口 1 选择接口

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（PA10）和 USART1_TX（PA9）则是相对于 STM32H750VBT6 来说的。这样，通过对接，就可以实现 USB 串口和 STM32H750VBT6 的串口通信了。

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 MiniPro STM32H750 板载的标准 20 针 JTAG/SWD 接口电路如图 2.1.4.1 所示：

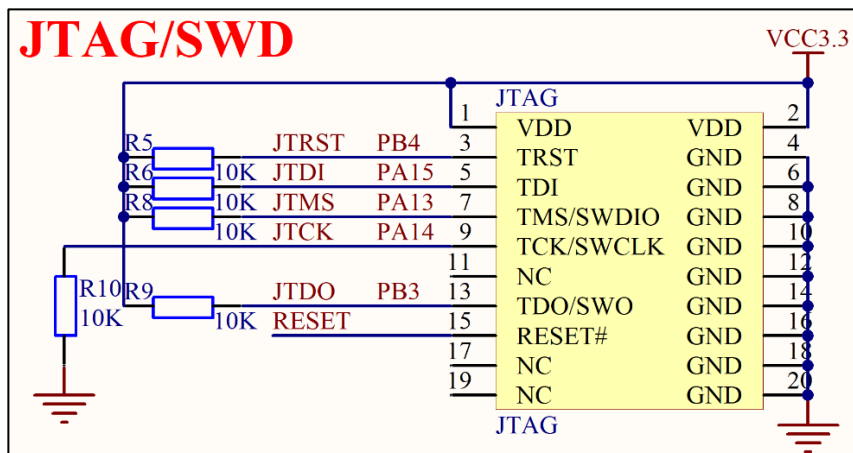


图 2.1.4.1 JTAG/SWD 接口

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 LCD 模块接口

正点原子 MiniPro STM32H750 板载的 LCD 模块接口电路如图 2.1.5.1 所示：

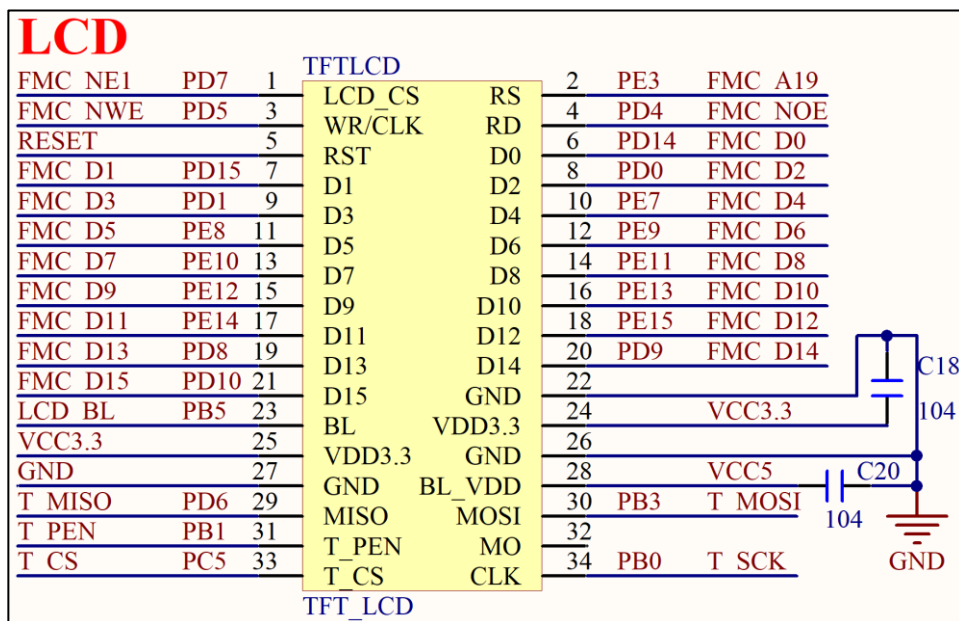


图 2.1.5.1 LCD 模块接口

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 STM32H750 的 FMC 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 PD6/PB3/PB1/PB0/PC5 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 PB5 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。

2.1.6 复位电路

正点原子 MiniPro STM32H750 的复位电路如图 2.1.6.1 所示：

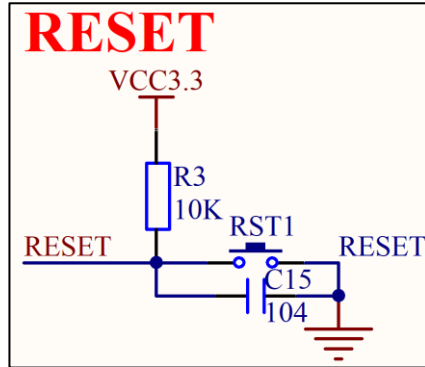


图 2.1.6.1 复位电路

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 C15 构成了上电复位电路。同时，开发板把 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。

2.1.7 启动模式设置接口

正点原子 MiniPro STM32H750 的启动模式设置端口电路如图 2.1.7.1 所示：

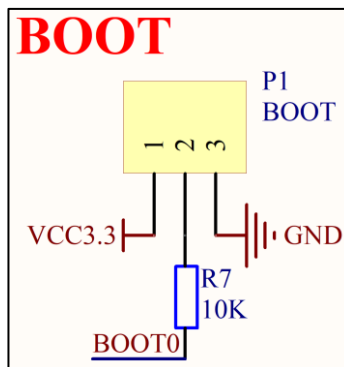


图 2.1.7.1 启动模式设置接口

上图的 BOOT0 用于设置 STM32 的启动方式，其对应启动模式如表 2.1.7.1 所示：

启动模式选择		启动地址
BOOT0	启动地址选项字节	
0	BOOT_ADD0[15:0]	由用户选项字节 BOOT_ADD0[15:0]决定启动地址，ST 出厂默认的启动地址为：0X0800 0000 的 Flash 地址
1	BOOT_ADD1[15:0]	由用户选项字节 BOOT_ADD1[15:0]决定启动地址，ST 出厂默认的启动地址为：0X1FF0 0000 的系统存储器地址

表 2.1.7.1 BOOT0、BOOT1 启动模式表

表 2.1.7.1 中, BOOT_ADD0 和 BOOT_ADD1 的高 16 位地址可以由用户任意设置, 其设置范围为: 0X0000 0000~0X3FFF 0000, 涵盖了整个 FLASH 区域、SRAM 区域和 TCMRAM 区域, 基本上, 可以设置从任意地址启动 (低 16 位必须是 0), 通过 FLASH_BOOT_PRGR 寄存器设置。

在出厂的时候, ST 默认给 BOOT_ADD0 和 BOOT_ADD1 编程为: 0X0800 0000 和 0X1FF0 0000 分别对应用户 FLASH 的起始地址和系统存储器地址, 用于执行用户代码或者进入 BOOTLOADER 状态。一般情况下我们设置 B00T0 为低电平即可, 即从 0X0800 0000 的 FLASH 地址启动, 执行用户代码。

这里需要注意两点:

1, STM32H750 虽然也支持串口下载 (B00T0=1, 从系统存储器启动), 但目前没有比较好的支持 STM32H7 的串口下载软件, 所以, 大家必须自备仿真器 (DAP/STLINK) 一个, 用来下载和调试代码。

2, STM32H750 内部仅 1 个 FLASH 扇区, 大小为 128KB, 用于存储用户代码。一般需要通过 QSPI 外扩 SPI FLASH 来扩展 FLASH 容量, 以满足大项目需求。

2.1.8 EEPROM

正点原子 MiniPro STM32H750 板载的 EEPROM 电路如图 2.1.8.1 所示:

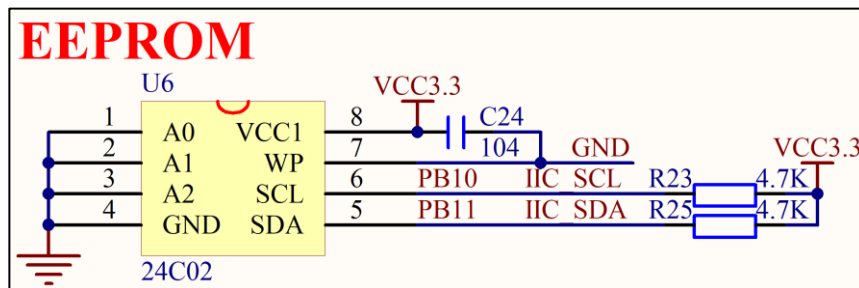


图 2.1.8.1 EEPROM

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 和 IIC_SDA 分别接在 MCU 的 PB10 和 PB11 上, 这里我们虽然接到了 STM32 的硬件 IIC 上, 但是并不提倡大家使用硬件 IIC, 因为 ST 的硬件 IIC 不太好用。

另外, PB10 和 PB11 还连接在 DCMI 接口的 DCMI_SCL 和 DCMI_SDA, 请分时复用。

2.1.9 SPI FLASH (QSPI FLASH)

正点原子 MiniPro STM32H750 板载的 SPI FLASH 电路如图 2.1.9.1 所示:

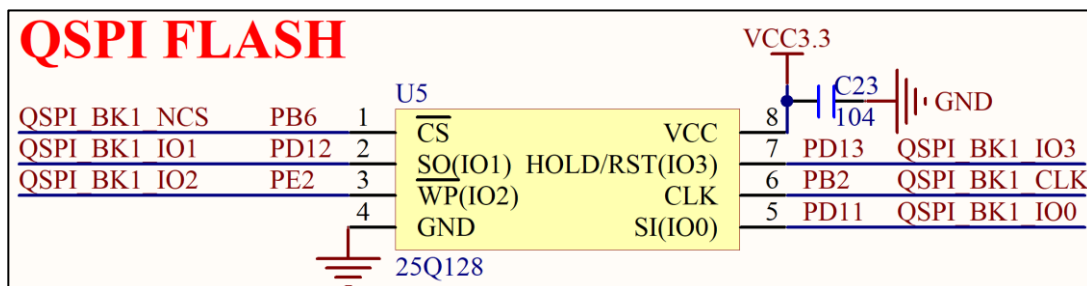


图 2.1.9.1 SPI FLASH 芯片

SPIFLASH 芯片型号为 25Q128（可选品牌为：华邦或博雅等都可以），这里我们使用的是四线（QSPI）接口，容量为 128Mb，也就是 16M 字节，通过 H750 的 QSPI 接口访问，IO 连接关系如上图所示。

由于 STM32H750 是 Value Line 产品线，因此内部 FLASH 都很小（只有 128KB），但是它可以通过 QSPI 接口访问外部 QSPI FLASH 并进行地址映射，从而运行外部 QSPI FLASH 的代码，所以该 SPI FLASH 芯片有 2 个作用：

- 1，作为程序存储器，保存需要执行的程序代码。
- 2，作为数据存储器，保存程序运行过程中所需要用到的数据。

注意：由于 H750 的 QSPI 接口是不支持加密访问的，因此我们一般需要将程序分成 2 个部分存储，核心的关键的代码，存储在 H750 内部 FLASH，非关键的，占用空间大的，存储在外部的 QSPI FLASH。这样既可以保证程序的安全性，又可以保证大代码可以正常下载运行。

2.1.10 温湿度传感器接口

正点原子 MiniPro STM32H750 板载的温湿度传感器接口电路如图 2.1.10.1 所示：

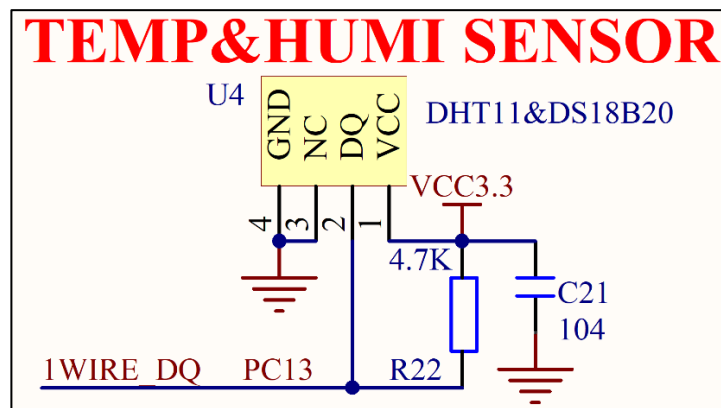


图 2.1.10.1 温湿度传感器接口

该接口（U4）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 PC13 上。

2.1.11 光敏传感器

正点原子 MiniPro STM32H750 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 2.1.11.1 所示：

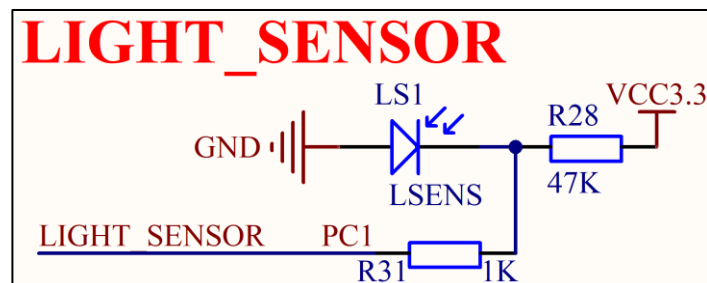


图 2.1.11.1 光敏传感器电路

图中的 LS1 就是光敏传感器，其实就是一个光敏二极管（PTSM021），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 ADC123_INP11（ADC1/2/3 的通道 11）上面，即 PC1 引脚。

2.1.12 红外接收头

正点原子 MiniPro STM32H750 板载的红外接收头电路如图 2.1.12.1 所示：

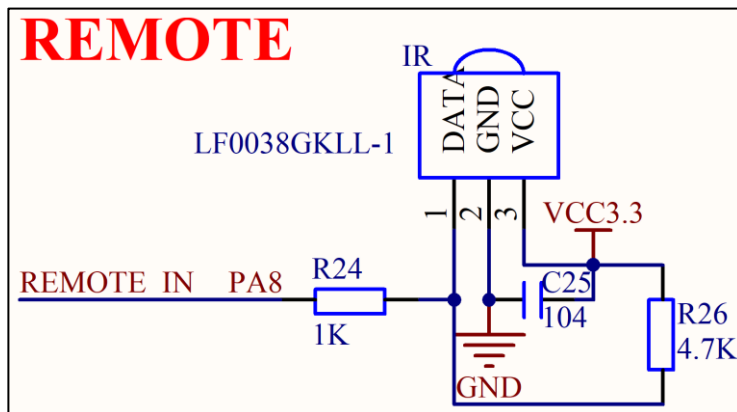


图 2.1.12.1 红外接收头

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 PA8 上。

注意：PA8 同时连接 DCMI_XCLK，因此，当用到 DCMI_XCLK 信号时，红外遥控功能和 DCMI 接口要分时复用。

2.1.13 无线模块接口

正点原子 MiniPro STM32H750 板载的无线模块接口电路如图 2.1.13.1 所示：

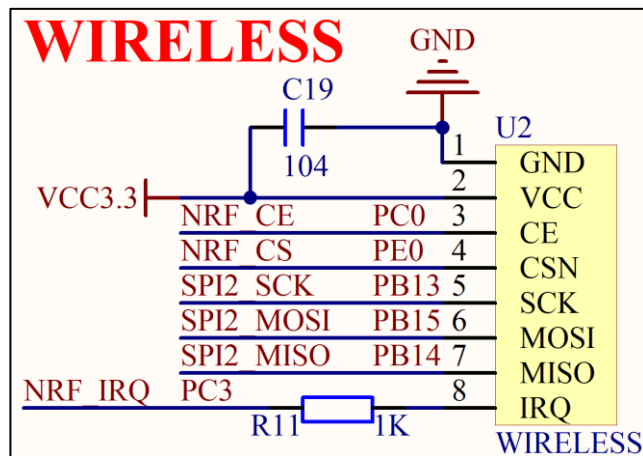


图 2.1.13.1 无线模块接口

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 PC0/PE0/PC3 上，而另外 3 个 SPI 信号则连接 MCU 的 SPI2，对应 IO 分别是：PB13/PB14/PB15。

2.1.14 LED

正点原子 MiniPro STM32H750 板载总共有 2 个 LED，其原理图如图 2.1.14.1 所示：

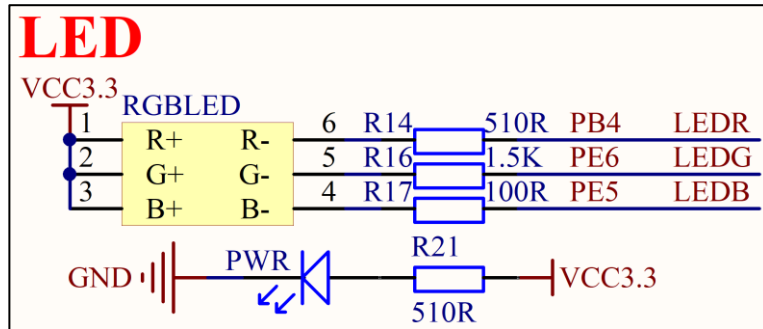


图 2.1.14.1 LED

其中 PWR 是系统电源指示灯，为蓝色。RGBLED 是 RGB 三色全彩 LED，红灯（LEDR）连接 PB4、绿灯（LEDG）连接 PE6、蓝灯（LEDB）连接 PE5。

2.1.15 按键

正点原子 MiniPro STM32H750 板载总共有 3 个输入按键，其原理图如图 2.1.15.1 所示：

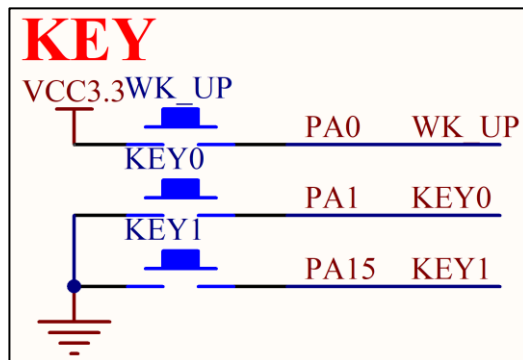


图 2.1.15.1 输入按键

KEY0 和 KEY1 用作普通按键输入，分别连接在 PA1 和 PA15 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。

2.1.16 OLED/摄像头模块接口

正点原子 MiniPro STM32H750 板载了一个 OLED/摄像头模块接口，其原理图如图 2.1.16.1 所示：

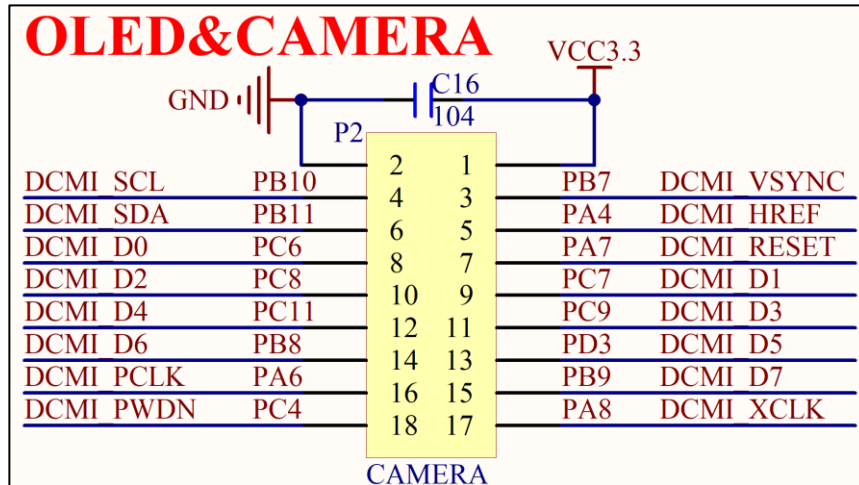


图 2.1.16.1 OLED/摄像头模块接口

图中 P2 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 DCMI_PWDN 和 DCMI_XCLK 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。

特别注意：

- 1, DCMI_SCL 和 DCMI_SDA 和 24C02 共用 PB10 和 PB11，他们要分时复用！
- 2, DCMI_D2、DCMI_D3、DCMI_D4 和 SDIO 共用 PC8/PC9/C11，他们也要分时复用！
- 3, DCMI_HREF 和 STM_DAC 共用 PA4，他们要分时复用！
- 4, DCMI_XCLK 和 REMOTE_IN 共用 PA8，他们要分时复用！

2.1.17 有源蜂鸣器

正点原子 MiniPro STM32H750 板载了一个有源蜂鸣器，其原理图如图 2.1.17.1 所示：

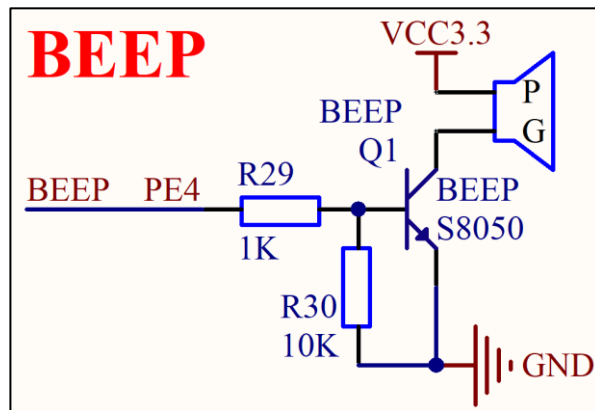


图 2.1.17.1 有源蜂鸣器

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，R30 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 PE4 上面，PE4 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。

2.1.18 TF 卡接口

正点原子 MiniPro STM32H750 板载了一个 TF 卡（小卡/Micro SD 卡）接口，其原理图如图 2.1.18.1 所示：

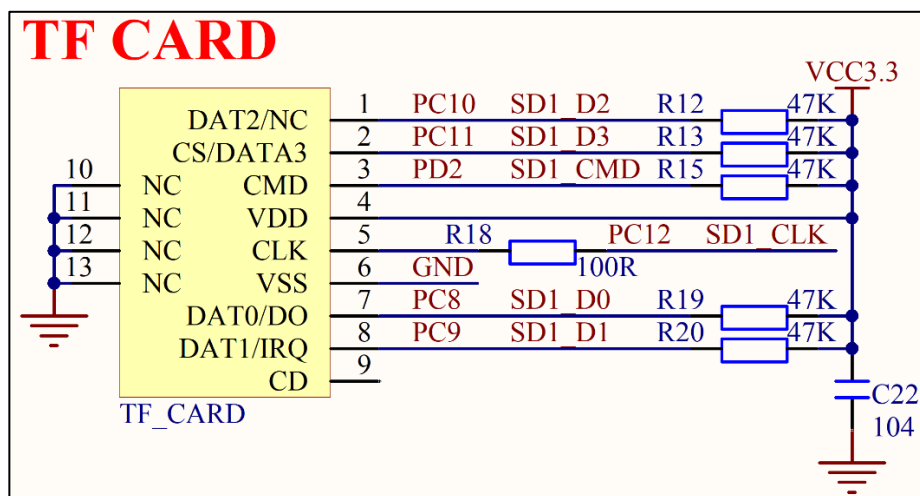


图 2.1.18.1 TF 卡接口

图中 TF_CARD 为 TF 卡接口，采用 SDIO 方式驱动，理论上最大速度可以达到 50MB/S，非常适合需要高速存储的情况。

注意：TF 卡接口和 DCMI 共用 PC8/PC9/C11，他们要分时复用！

2.1.19 ATK 模块接口

正点原子 MiniPro STM32H750 板载了 ATK 模块接口，其原理图如图 2.1.19.1 所示：

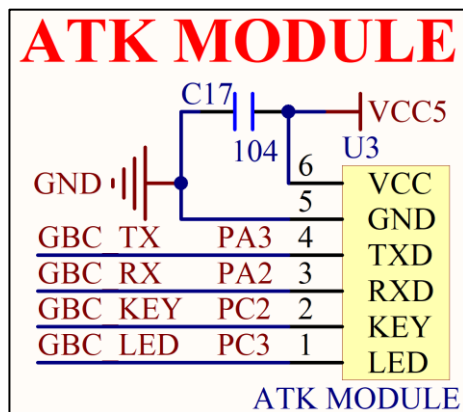


图 2.1.19.1 ATK 模块接口

如图所示，U3 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。

注意：GBC_LED 和 NRF_IRQ 共用 PC3，因此这两个外设不能同时使用，不过可以分时复用。

2.1.20 多功能端口

正点原子 MiniPro STM32H750 板载的多功能端口，是由 P3 构成的一个 4PIN 端口，其原理图如图 2.1.20.1 所示：



其中 2 脚连接滑动电阻 RV1 的滑动端，其电压可以通过 RV1 调节（范围是：0~3.3V）。另外，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。

P1 多功能端口，实物图如图 2.1.20.2 所示：



图 2.1.20.2 组合后的多功能端口

下面我们来看看，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，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）；2，ADC 和 GND 组成一组 ADC 输入；3，ADC 和 DAC 分别用作独立 IO 使用。

当使用 1 个跳线帽的时候：1，DAC 和 ADC 组成一个自输出测试，用 MCU 的 ADC 来测试 MCU 的 DAC 输出。2，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 5 个功能，所以，只要设计合理，1+1 是大于 2 的。

2.1.21 电源

正点原子 MiniPro STM32H750 开发板板载的电源供电部分，其原理图如图 2.1.21.1 所示：

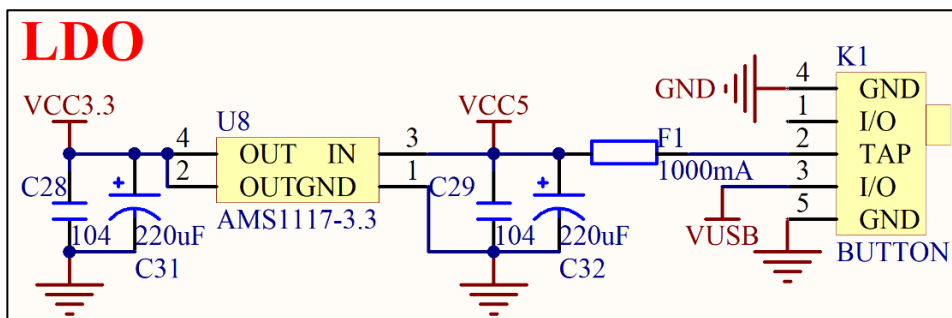


图 2.1.21.1 电源

图中 U8 是一个稳压芯片 (LDO)，将 5V 电压稳压成 3.3V，给开发板提供 3.3V 电源。K1

是开发板的总电源开关，F1 为 1000ma 自恢复保险丝，用于保护 USB。

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 2.1.23 节进行介绍。

2.1.22 电源输入输出接口

正点原子 MiniPro STM32H750 开发板板载了两组简单电源输入输出接口，其原理图如图 2.1.22.1 所示：

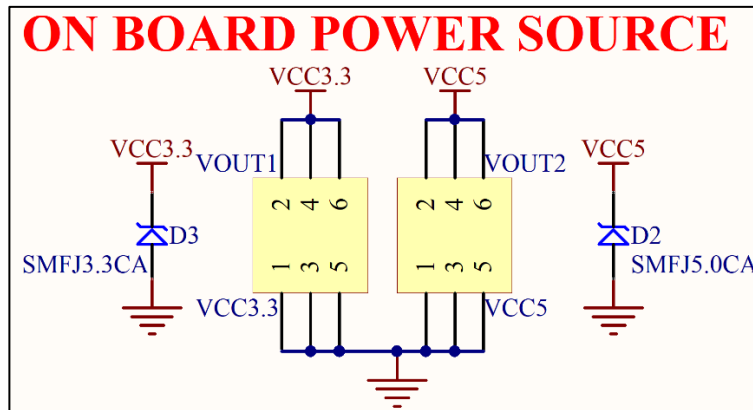


图 2.1.22.1 电源

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 500ma），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 D2 和 D3 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。

2.1.23 USB 通信接口

正点原子 MiniPro STM32H750 板载了两个 USB 通信接口（USB_Slave 和 USB_HOST），其原理图如图 2.1.23.1 所示：

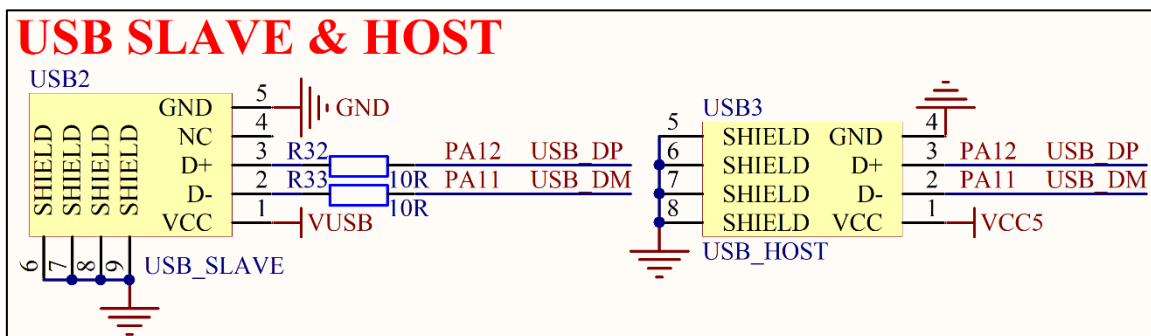


图 2.1.23.1 USB_Slave 接口

图中两个 USB 口：

USB_Slave 是 USB 从机接口，使用的是 Micro USB 座，通过 USB 线连接电脑可以用于 USB 从机通信，如：USB CDC、USB MSC、USB HID 等。

USB_HOST 是 USB 主机接口，是一个 USB A 座，可以用于 USB 主机通信，如外接：USB

鼠标、USB 键盘、USB 手柄、USB 扫码枪等。

注意：这两个 USB 共用 PA11 和 PA12，因此他们不能同时使用，必须分时复用。此外，USB_Slave 接口还可以用给开发板供电，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。

2.1.24 USB 串口

正点原子 MiniPro STM32H750 开发板板载了一个 USB 串口，其原理图如图 2.1.24.1 所示：

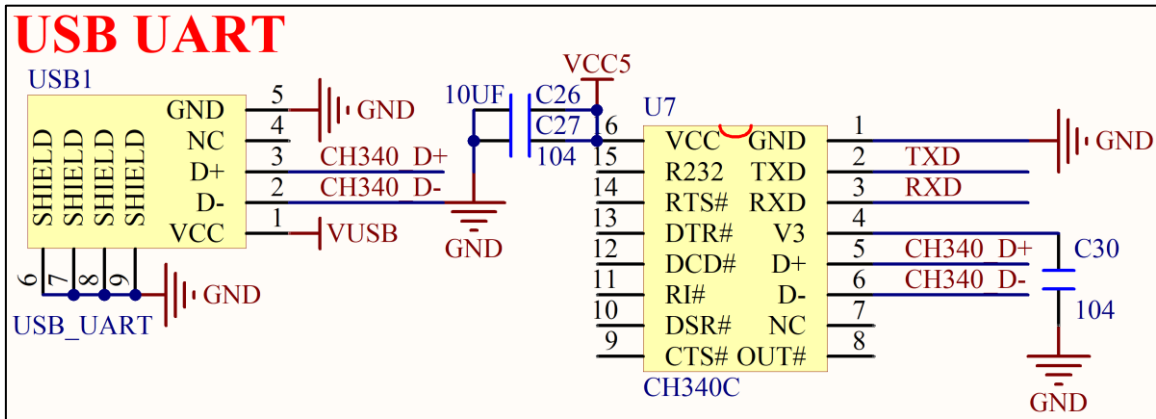


图 2.1.24.1 USB 串口

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

USB_UART 是一个 Micro USB 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 MiniPro STM32H750 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, 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，建议可以同时插 2 个 USB 口，并插上 JTAG，这样供电可以更足一些。
- 2, JTAG 接口有几个信号（JTDI/JTDO/JTRST）被 LCD、按键、LED 等占用了，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**。
- 3, 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 PA8 就不怎么适合用做输入检测，因为他接了红外传感器，随时可能会受到干扰。
- 4, 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。
- 5, 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 MiniPro STM32H750 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。