

# STM32F103 HAL库 made by 正点原子

---

## 前言

- Cortex-M3 采用 ARM V7 构架，不仅支持 Thumb-2 指令集，而且拥有很多新特性。较之ARM7 TDMI，Cortex-M3 拥有更强劲的性能、更高的代码密度、位带操作、可嵌套中断、低成本、低功耗等众多优势。
- 拥有FSMC、TIMER、SPI、IIC、USB、CAN、IIS、SDIO、ADC、DAC、RTC、DMA 等众多外设及功能，具有极高的集成度。
- STM32 仅 M3 内核就拥有 F100、F101、F102、F103、F105、F107、F207、F217 等 8 个系列上百种型号，具有 QFN、LQFP、BGA 等封装可供选择。同时 STM32还推出了 STM32L 和 STM32W 等超低功耗和无线应用型的 M3 芯片
- 优异的实时性能。84 个中断，16 级可编程优先级，并且所有的引脚都可以作为中断输入
- 杰出的功耗控制。STM32 各个外设都有自己的独立时钟开关，可以通过关闭相应外设的时钟来降低功耗。
- STM32 的开发不需要昂贵的仿真器，只需要一个串口即可下载代码，并且支持 SWD 和 JTAG 两种调试口。

## 硬件篇

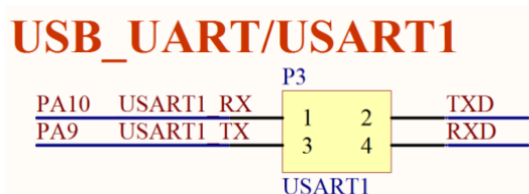
外设略：

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）、

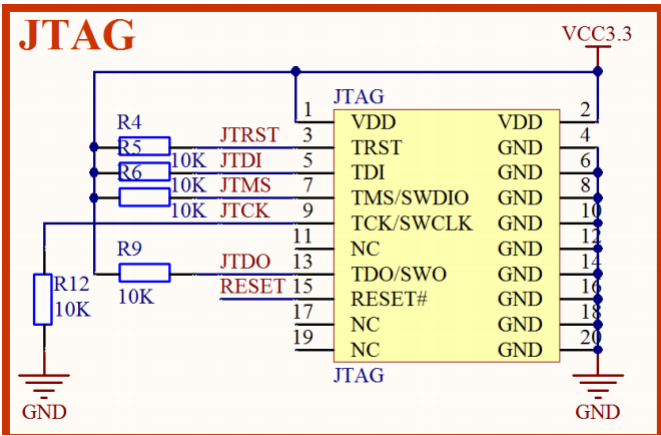
3 个 SPI、2 个 IIC、5 个串口、1 个 USB、1 个 CAN、3 个 12 位 ADC、1 个 12 位 DAC、1 个SDIO 接口、1 个 FSMC 接口以及 112 个通用 IO 口。

## USB连接电路：



图中 TXD/RXD 是相对 CH340G 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

## JTAG/SWD

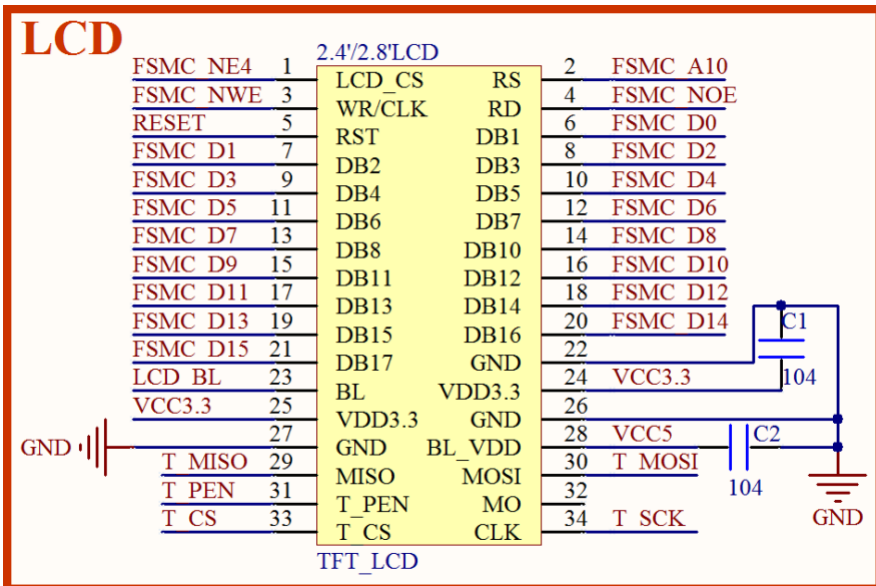


JLINK V7/V8、ULINK2 和 ST LINK 等都支持 SWD 调试。

SWD 只需要最少 2跟线（SWCLK 和 SWDIO）就可以下载并调试代码了

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，所以在使用的时候，推荐大家一律使用 SWD 模式！！

## LCD模块接口



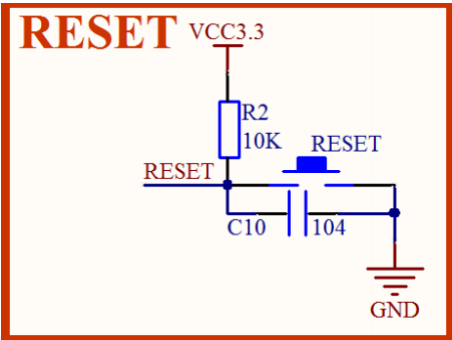
通用的液晶模块接口，支持ALIENTEK全系列TFTLCD 模块，包括：

2.4 寸、2.8 寸、3.5 寸、4.3 寸和 7 寸等尺寸的 TFTLCD 模块

连接在 STM32F103ZET6的 FSMC 总线上面，可以显著提高 LCD 的刷屏速度。

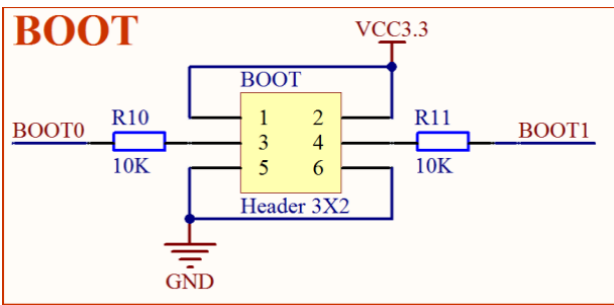
图中的 T\_MISO/T\_MOSI/T\_PEN/T\_SCK/T\_CS 连接在 MCU 的 PB2/PF9/PF10/PB1/PF11 上，  
这些信号用来实现对液晶触摸屏的控制（支持电阻屏和电容屏）。LCD\_BL 连接在 MCU 的 PB0 上，用于控制 LCD 的背光。液晶复位信号 RESET 则是直接连接在开发板的复位按钮上，和 MCU 共用一个复位电路

复位电路



低电平复位，可复位MCU和LCD

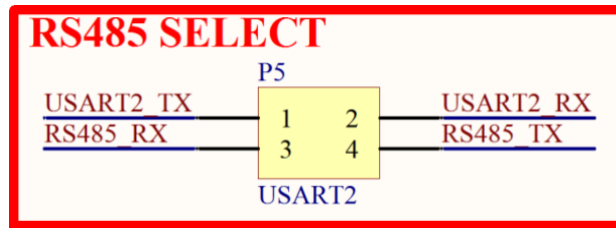
启动模式设置端口



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
1	1	SRAM启动	SRAM启动，用于在SRAM中调试代码

- 用串口下载代码，则必须配置BOOT0为1，BOOT1为 0
- 一按复位键就开始跑代码，则需要配置 BOOT0 为 0，BOOT1 随便设
- 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。

## RS485选择接口

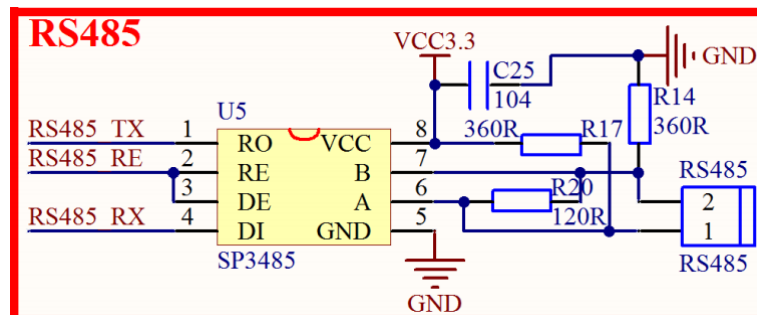


USART2\_RX 和 USART2\_TX 分别连接在 STM32F103ZET6 的 PA3 和 PA2 上面，RS485\_TX 和 RS485\_RX 则分别连接在 SP3485 的 **RO** 和 **DI** 引脚

用 2 个跳线帽短接 P5 的 1 和 3、2 和 4，即可实现 RS485 接口连接在 STM32 的串口 2 上面，完成 RS485 通信。当拔了这两个跳线帽的时候，该接口可以实现 2 个功能：

- 1，作为 PA2 和 PA3 的引出 IO 口；
- 2，开发板的 RS485 部分，作为 **RS485 转 TTL** 模块使用。

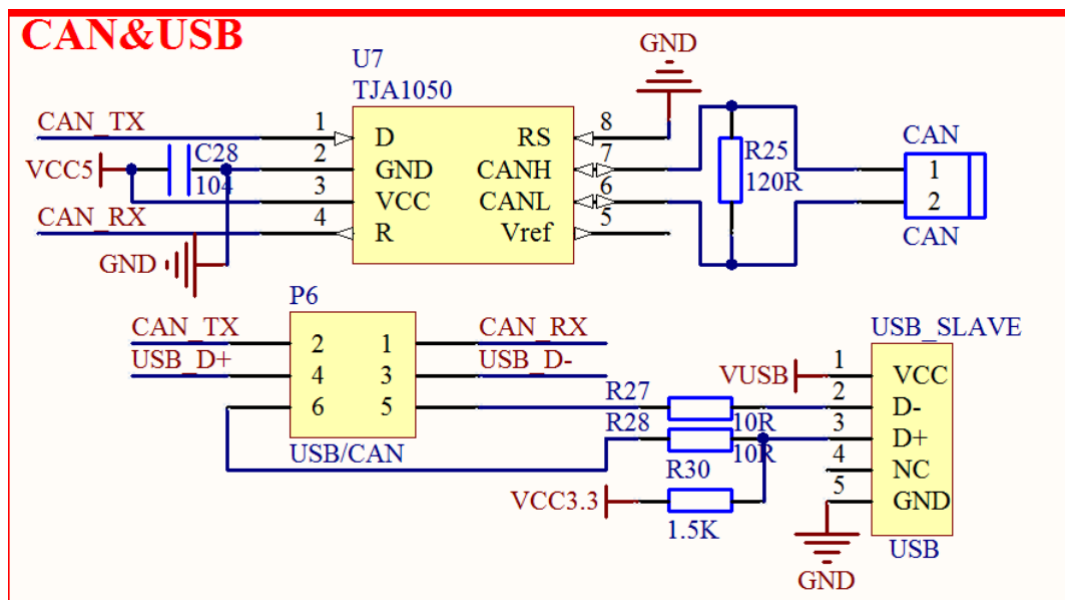
## RS485接口



使用 **SP3485** 来做 485 电平转换，其中 R20 为终端匹配电阻，而 R14 和 R17，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 P5 上面，通过 **P5 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 SP3485 的工作模式（高电平为发送模式，低电平为接收模式）。

## CAN/USB接口

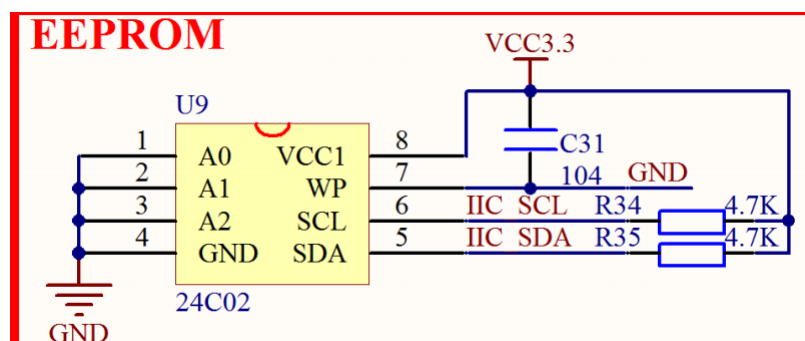


使用 TJA1050来做 CAN 电平转换，其中 R25 为终端匹配电阻。

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 P6 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。

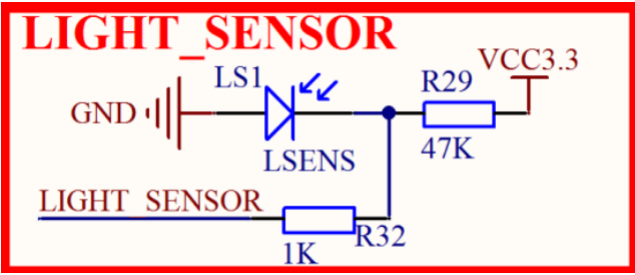
## EEPROM



该芯片的容量为 **2Kb**，也就是 256 个字节

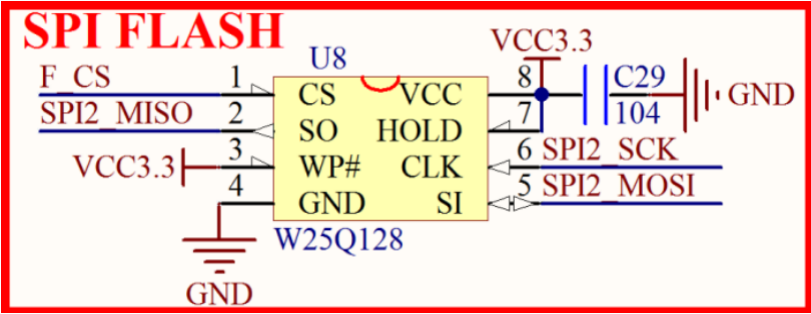
A0~A2 均接地，对 24C02 来说也就是把地址位设置成了 0

# 光敏传感器



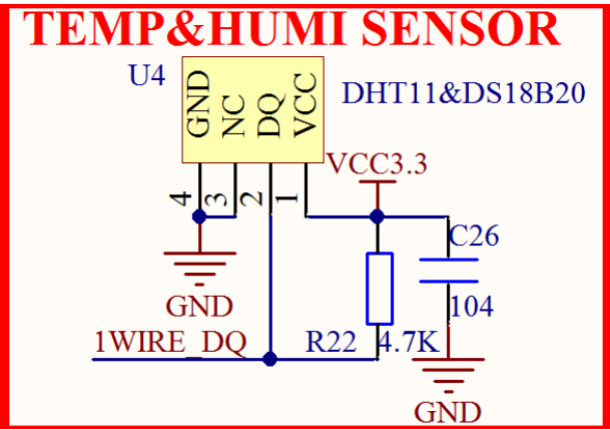
LIGHT\_SENSOR 连接在 MCU 的 ADC3\_IN6（ADC3 通道 6）上面，即 PF8 引脚

# SPI FLASH



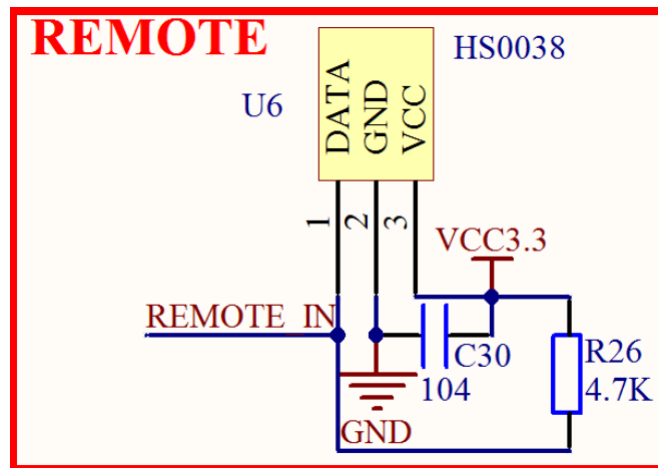
SPI FLASH 芯片型号为 W25Q128，该芯片的容量为 128Mb，也就是 16M 字节。该芯片和 NRF24L01 共用一个 SPI（SPI2），通过片选来选择使用某个器件，在使用其中一个器件的时候，请务必禁止另外一个器件的片选信号。

# 湿温度传感器



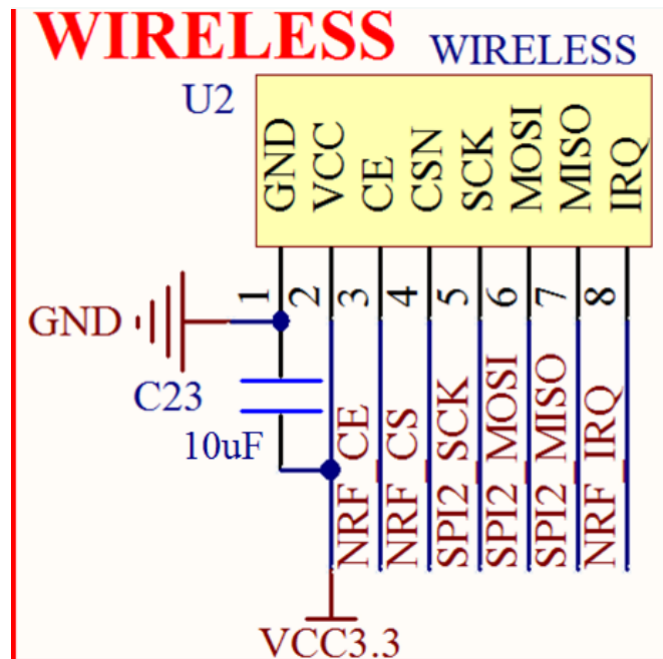
1WIRE\_DQ 是传感器的数据线，该信号连接在 MCU 的 PG11 上。

## 红外接收头



REMOTE\_IN 为红外接收头的输出信号，该信号连接在MCU 的 PB9 上

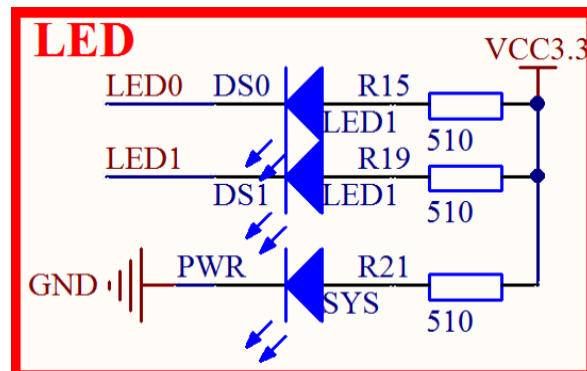
## 无线模块接口



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）。

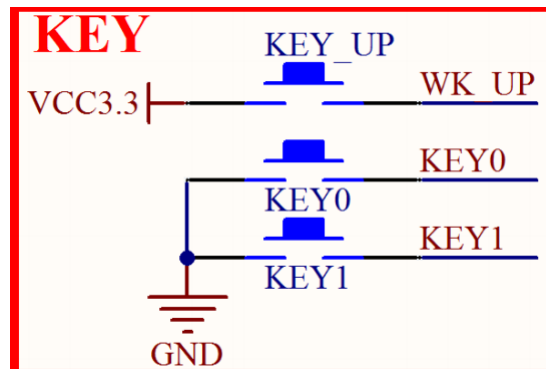
NRF\_CE/NRF\_CS/NRF\_IRQ 连接在 MCU 的 PG8/PG7/PG6 上，而另外 3 个 SPI 信号则和 SPI FLASH 共用，接 MCU 的 SPI2。

## LED

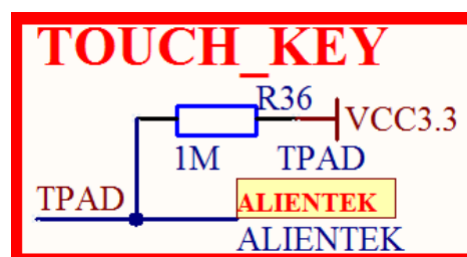


其中 PWR 是系统电源指示灯，为蓝色。LED0(DS0)和 LED1(DS1)分别接在 PB5 和 PE5 上，选择 DS0 为红色的 LED，DS1 为绿色的 LED。

## 按键



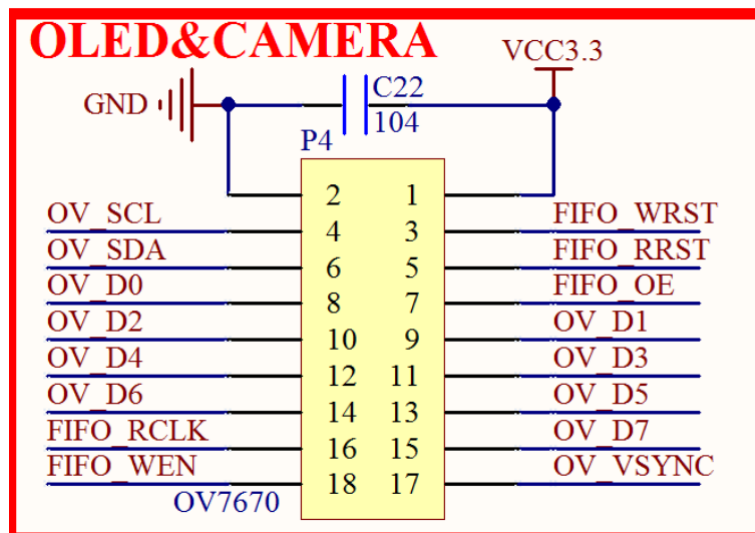
## TPAD电容介绍



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 AD/DA 组合端口（P7）上面，通过跳线帽来选择是否连接到 STM32。



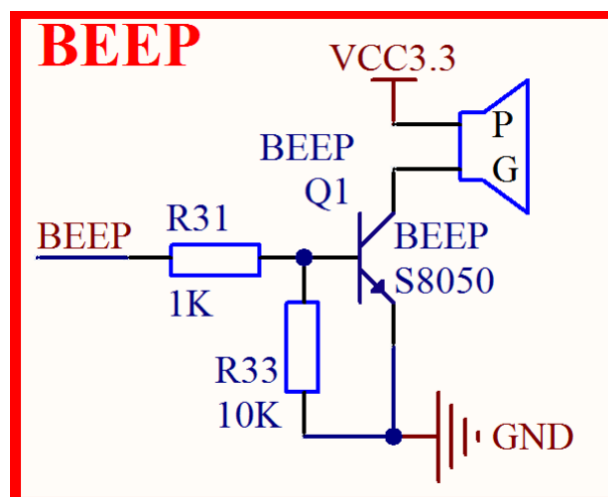
## OLED/CAM



如果是 OLED 模块，则 FIFO\_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。

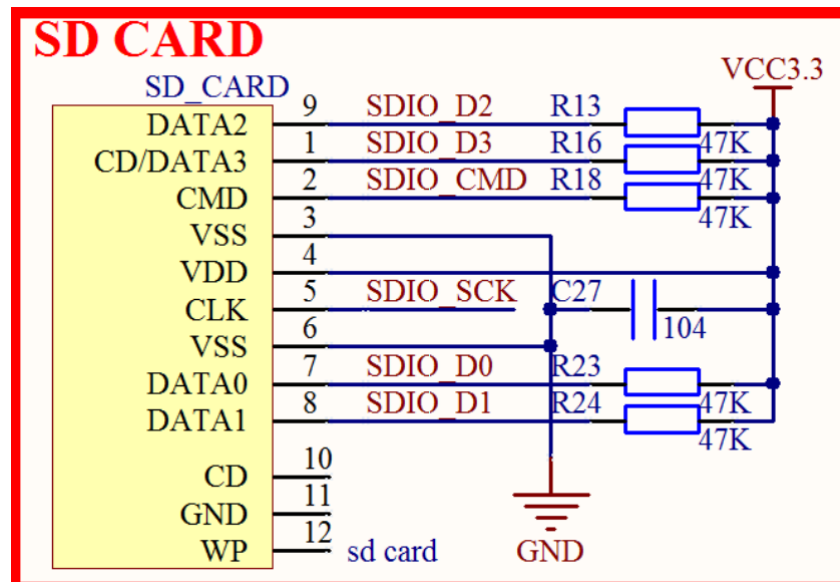
其中，OV\_SCL/OV\_SDA/FIFO\_Wrst/FIFO\_Rrst/FIFO\_OE 这 5 个信号是分别连接在 MCU 的 PD3/PG13/PD6/PG14/PG15 上面，OV\_D0~OV\_D7 则连接在 PC0~7 上面（放在连续的 IO 上，可以提高读写效率），FIFO\_RCLK/FIFO\_WEN/OV\_VSYNC 这 3 个信号是分别连接在 MCU 的 PB4/PB3/PA8 上面。其中 PB3 和 PB4 又是 JTAG 的 JTRST/JTDO 信号，所以在使用 OV7725 的时候，不要用 JTAG 仿真，要选择 SWD 模式

## 蜂鸣器



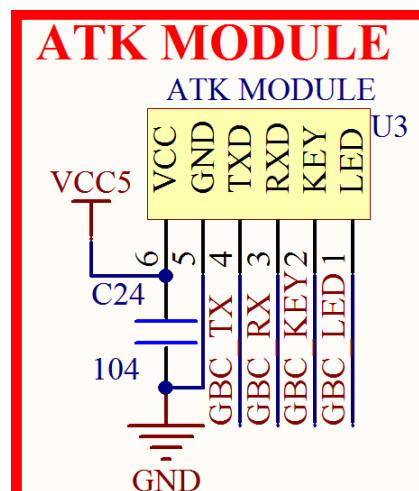
BEEP 信号直接连接在 MCU 的 PB8 上面，PB8 可以做 PWM 输出

## SD卡



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

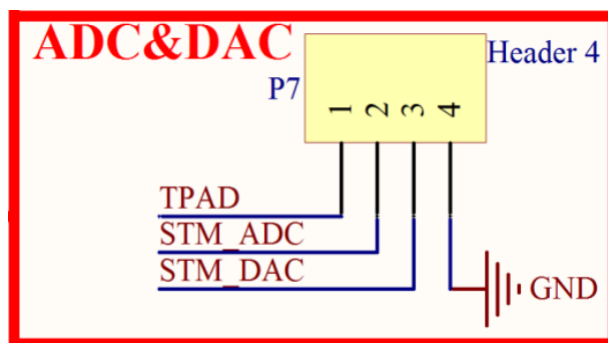
## ATK



GBC\_TX/GBC\_RX 分别连接在 MCU 的 PB11/PB10（即串口 3）上面，而 GBC\_KEY 和 GBC\_LED 则分别连接在 MCU 的 PA4 和 PA15 上面。

GBC\_KEY 与 STM\_DAC共用 PA4，GBC\_LED 和 JTDI 共用 PA15

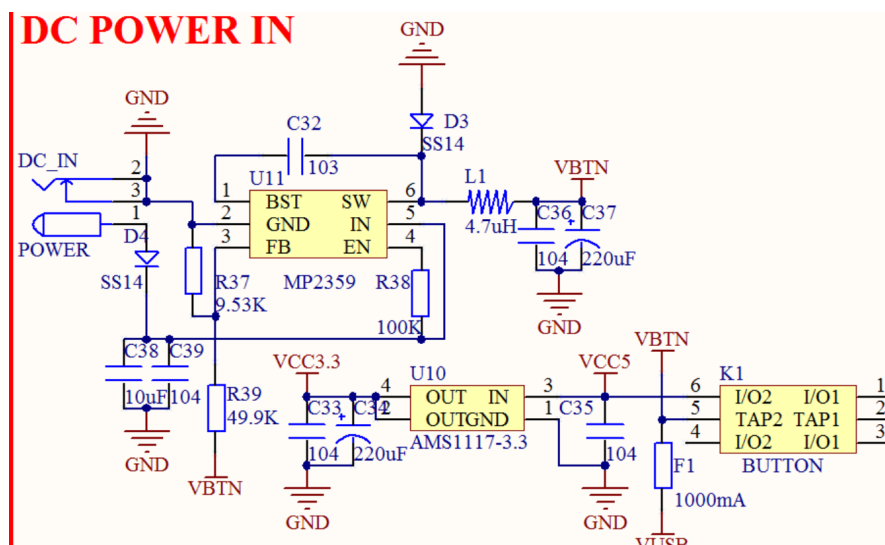
## AD/DA



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（默认是连接的），就可以实现电容触摸按键（利用定时器的输入捕获）。另外，STM\_ADC 还可以用与 AD 采集。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。

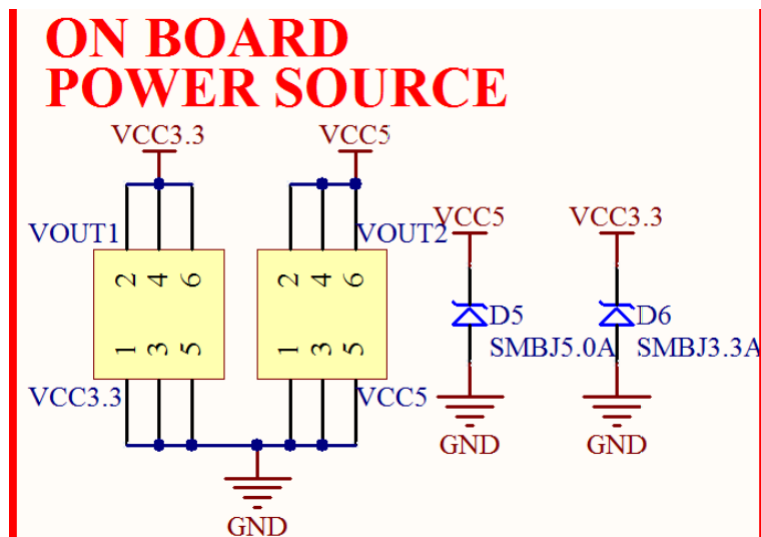
特别注意：STM\_DAC 与摄像头的 GBC\_KEY 共用 PA4，所以他们不可以同时使用，但是可以分时复用

# POWER



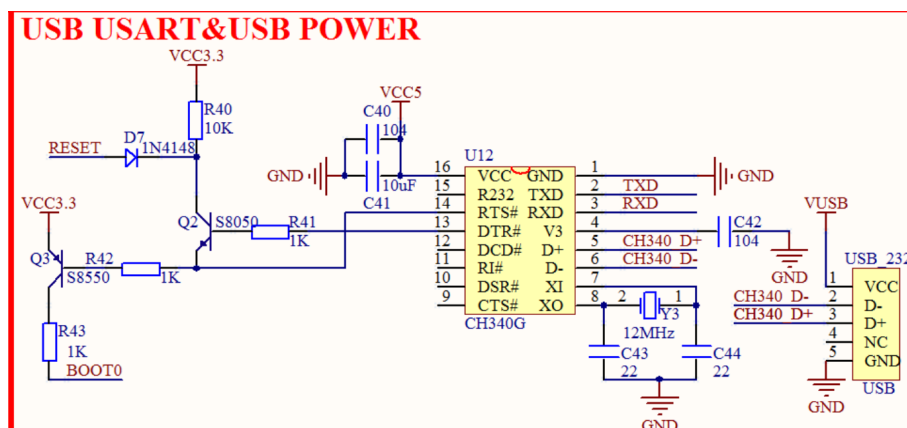
DC\_IN 用于外部直流电源输入，范围是 DC6~24V，输入电压经过 U11 DC-DC 芯片转换为 5V 电源输出，其中 D4 是防反接二极管，避免外部直流电源极性搞错的时候，烧坏开发板。K1 为开发板的总电源开关，F1 为 1000ma 自恢复保险丝，用于保护 USB。U10 为 3.3V 稳压芯片，给开发板提供 3.3V 电源。

## POWER INPUT AND OUTPUT



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）

## USB SERIAL



USB 转串口，我们选择的是 CH340G，是国内芯片公司南京沁恒的产品，稳定性经测试还不错，所以还是多支持下国产。

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\_232 是一个 MiniUSB 座, 提供 CH340G 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\_232 是本开发板的主要供电口。