

1、外观展示

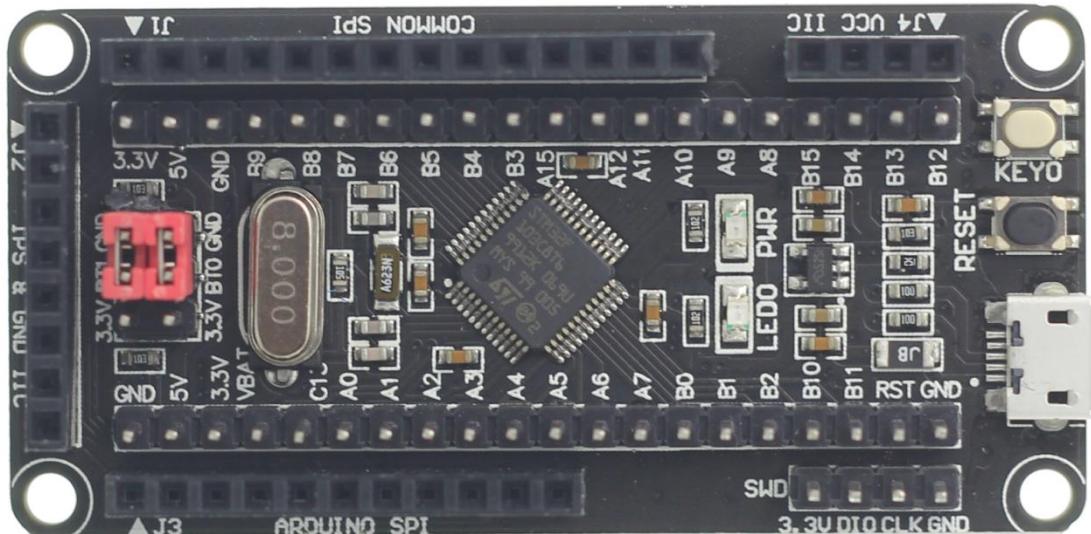


图 1. 正面



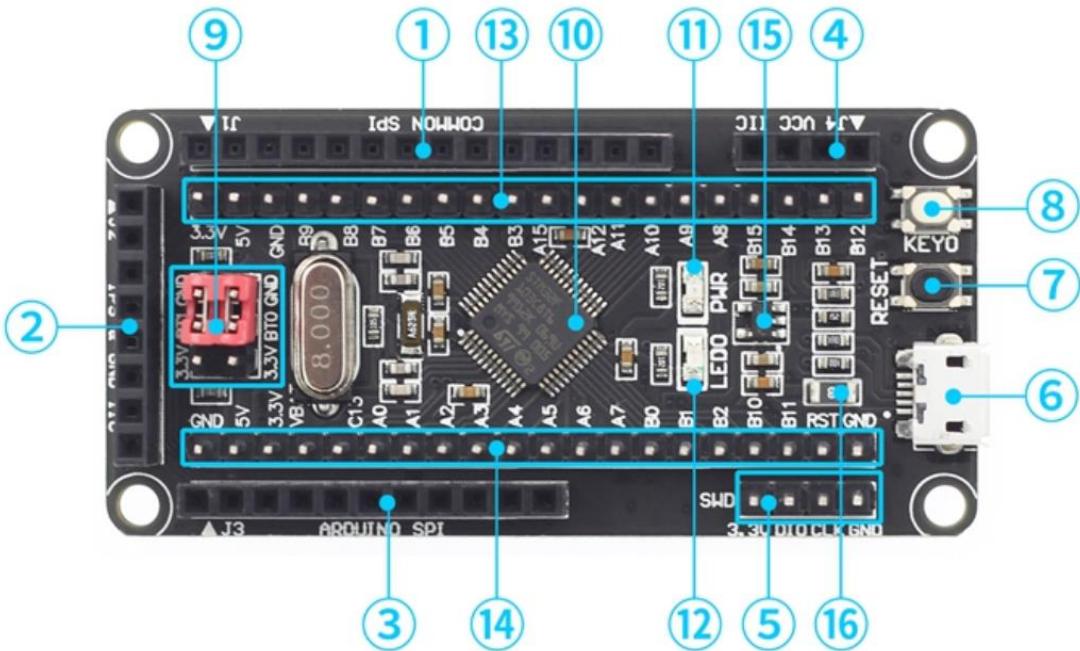


图 3. STM32F103C8T6 最小系统板硬件资源图

- | | | | |
|----------------------|-----------------|--------------------|----------|
| ①J1 直插接口 | ②J2 直插接口 | ③J3 直插接口 | ④J4 直插接口 |
| ⑤SWD 下载仿真接口 | ⑥USB 通信接口 | ⑦RESET 复位按键 | |
| ⑧KEYO 测试按键 | ⑨BT0/BT1 启动选择端口 | ⑩STM32F103C8T6 主芯片 | |
| ⑪红色的 LED 电源指示灯 (PWR) | ⑫蓝色 LED 测试灯 | ⑬扩展 IO | ⑭扩展 IO |
| ⑮3.3V 输出稳压 IC | ⑯过流保护保险丝 | | |

硬件资源具体说明如下：

- STM32F103C8T6 主芯片，LQFP48 封装，主频：72MHz，内部 FLASH：64KB，内部 SRAM：20KB
- 红色的 LED 电源指示灯 (PWR)，用于指示最小系统板上电情况
- USB 通信接口可用于最小系统板和其他设备进行 USB 通信，也可用于供电
- SWD 下载仿真接口可用于 SWD 下载和仿真，此 4 针接口中 DIO (PA13) 和 CLK (PA14) 可做普通 IO 口使用
- LED0 为蓝色 LED 测试灯，接在 PC13 引脚上，可用于测试 PC13 引脚输出，也可用于指示程序运行状态，当然也实现一些用户需要的灯光效果
- 扩展 IO 共有两组，除了 RTC 晶振占用的两个引脚 (PC14 和 PC15) 和 SWD 接口引出的两个引脚，其他的引脚都通过扩展 IO 引出来了，方便客户接入各种外设进行开发
- 此款最小系统板还针对我们现有的使用 SPI 和 IIC 接口的显示模块专门设置了 J1~J4 四个直插接口，它们的具体使用说明如下：
 - ◆ J1 接口用于直插普通的使用 SPI 接口的 LCD 模块，包括带触摸和不带触摸功能的

- ◆ J2 接口用于直插使用 SPI 接口的 IPS 屏模块和 OLED 模块，另外还兼容直插第一引脚为 GND 的使用 IIC 接口的 OLED 模块
- ◆ J3 接口用于直插在 Arduino 单片机上使用的 SPI 接口的 LCD 模块
- ◆ J4 接口用于直插第一引脚为 VCC 的使用 IIC 接口的 OLED 模块
- BT0/BT1 启动选择端口用于选择最小系统板复位后的启动模式。它们是通过跳帽来选择连接 VCC（高电平）还是 GND（低电平）。相关的启动模式说明如下：

BT0	BT1	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

- KEY0 按键接在 PA0 引脚上，默认为下拉输入，高电平输入表示按键按下。此按键可用于待机模式下唤醒最小系统板。当不使用按键唤醒功能时，可做普通按键使用，可用于用户其他按键功能开发，例如控制 LED 灯开启等等。
- 复位按键（RESET）和 STM32F103C8T6 主芯片的复位引脚连在一起，用于复位主芯片。当不需要此复位按键时，用户可以通过连接最小系统板上扩展出来的 RST 引脚来实现复位功能。
- 3.3V 输出稳压 IC 用于稳定输出电压，因为该最小系统板的输入电压为 5V 左右，而主 IC 的最佳工作电压为 3.3V，所以需要将 5V 转为 3.3V 稳定输出，保证主 IC 正常工作。
- 过流保护保险丝用于保护主 IC 和稳压 IC 不因电流过大而烧坏。因为最小系统板发生短路或者电源和地接反都会导致电流突然增大，会超过 IC 的极限值。

3、原理图详解

STM32F103C8T6 最小系统板的原理图包括如下 11 部分：

- USB 接口电路
- LED 灯控制电路
- 启动模式选择电路
- SWD 接口电路
- 复位控制电路
- 电源控制电路

- STM32F103C8T6 主控电路
- 外部电池供电电路
- 按键电路
- 外部扩展 IO 接口电路
- 液晶屏模块接口电路

整体电路原理图见资料包中的原理图 PDF 文档

各部分的原理图详细说明如下：

A、USB 接口电路

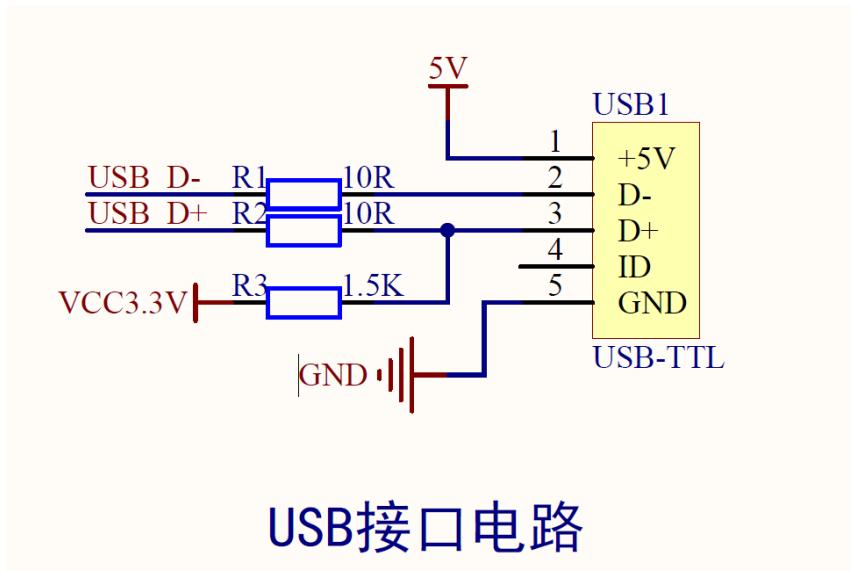
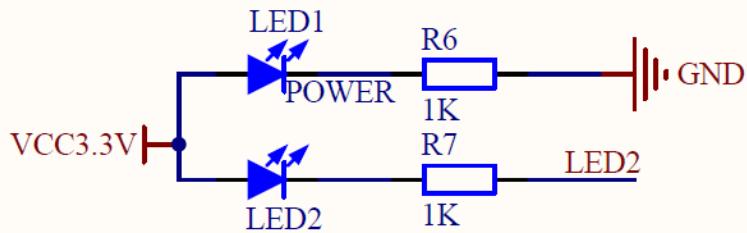


图 4. USB 接口原理图

其中 D- 和 D+ 分别接到 STM32F103C8T6 主芯片的 PA11 和 PA12 上，R1 和 R2 用于平衡阻抗，如果要求不是很严格，可以去掉。R3 为 D+ 引脚上拉电阻，使 D+ 引脚默认为高电平输出，一旦最小系统板接到 PC 上，PC 机的 USB 口检测到 D+ 脚为高电平，就会提示有 USB 设备接入，然后自动安装 USB 驱动。

B、LED 灯控制电路

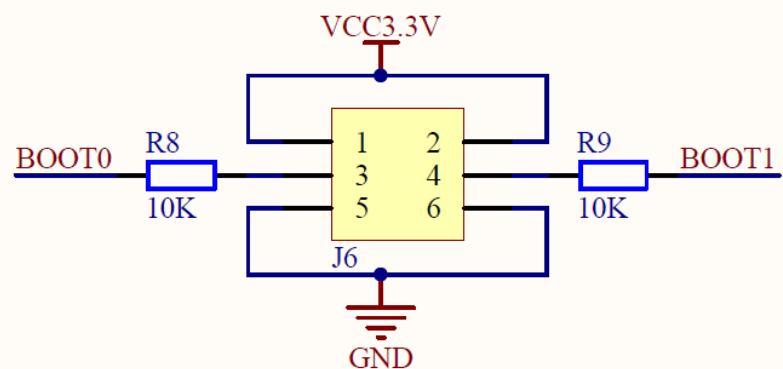


LED灯控制电路

图 5. LED 灯控制原理图

其中 LED1 为电源指示灯，当最小系统板正常上电后，该指示灯会亮。LED2 为用户控制的 LED 灯，接在主芯片的 PC13 引脚上，一旦 PC13 引脚输出低电平（默认为高电平），LED2 就会亮。R6 和 R7 为 LED 灯限流电阻，阻值越小，灯越亮，但是电流不能超过 LED 灯所能承受的最大电流。

C、启动模式选择电路

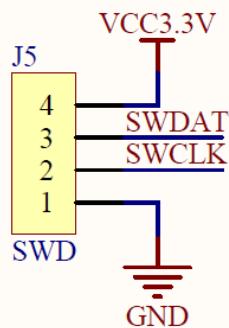


启动模式选择电路

图 6. 启动模式选择原理图

其中 BOOT0 和 BOOT1 分别接在主芯片的 BOOT0 和 PB2（BOOT1）引脚上，J6 为两排三列的排针，VCC3.3V 为高电平，GND 为低电平。BOOT0 和 BOOT1 分别通过跳帽连接高电平和低电平。R8 和 R9 为上下拉电阻（接高电平时，为上拉电阻；接低电平时，为下拉电阻）

D、SWD 接口电路

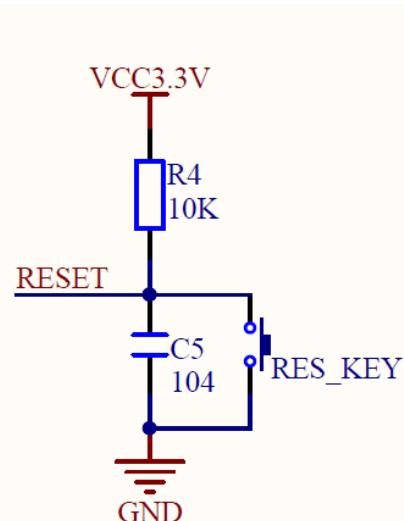


SWD接口电路

图 7. SWD 接口原理图

其中 SWDAT 和 SWCLK 分别接在主芯片的 PA13 和 PA14 引脚上。下载器接到该接口上就可以进行 SWD 下载和仿真了。如果不需要进行下载和仿真，SWDAT 和 SWCLK 可以做普通引脚使用。

E、复位控制电路



复位控制电路

图 8. 复位控制原理图

其中 RESET 接在主芯片的复位引脚上，R4 为上拉电阻，C5 为旁路电容，RES_KEY 为复位按键。该电路可以用于上电复位和按键复位。当最小系统板刚上电时，C5 会

立即进行充电，此时 C5 两端相当于短路，RESET 脚接到 GND，主芯片复位引脚被拉低，进入复位状态，随着 C5 充电完毕，C5 两端相当于开路，RESET 脚被 R4 上拉到高电平，复位完成，所以只要 C5 容值选择恰当，保证 RESET 引脚被拉低的时间大于主芯片需要的复位时间，就可以实现上电复位。在运行过程中，一旦 RES_KEY 按键按下，RESET 接到 GND，主芯片进入复位状态，RES_KEY 松开，RESET 被拉高，复位完成，从而实现按键复位。

F、电源控制电路

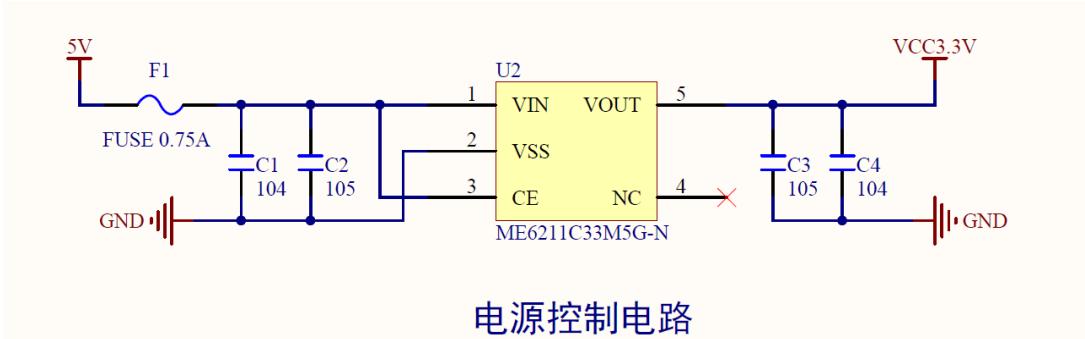


图 9. 电源控制原理图

其中 C1 和 C2 为输入端（5V）滤波电容，C3 和 C4 为输出端（3.3V）滤波电容，保证两端电压稳定。ME6211C33M5G 为专业的电源管理 IC，负责将 5V 输入转换成稳定的 3.3V 输出。F1 为过流保护保险丝，当最小系统板发生短路，电路中电流瞬间增大，有可能会超过 IC 的极限电流值，这样会导致 IC 被烧毁，F1 就是用来防止该情况发生，一旦电流过大，超过 F1 的保护电流极限值，F1 立即断开，这样整个电路就相当于断路，从而起到保护作用，当电路电流恢复正常，F1 又会恢复连接，从而保证电路正常工作。当然 F1 的极限电流值必须小于 IC 的极限电流值。

G、STM32F103C8T6 主控电路

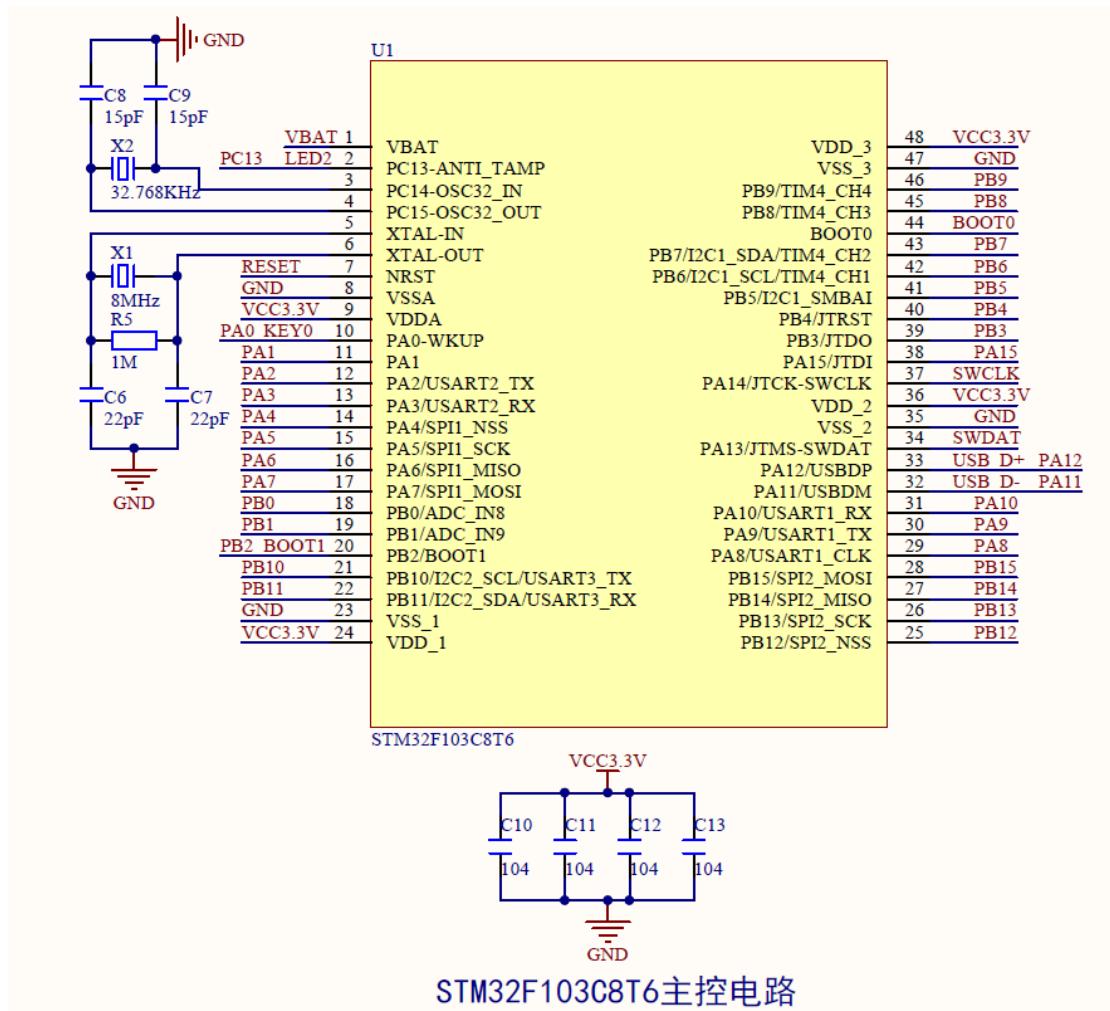
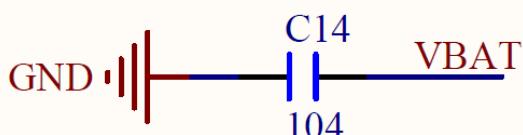


图 10. STM32F103C8T6 原理图

其中主控电路包含四部分：RTC 时钟晶振电路、主芯片时钟晶振电路、主芯片接口电路、主芯片电源滤波电路。RTC 时钟晶振用的是 32.768kHz，C8 和 C9 为晶振频率调节电容，这两个电容的容值需要根据 PCB 板的设计调整出最佳值。主芯片时钟使用 8MHz 的晶振，C6 和 C7 容值基本和晶振匹配就行。C10、C11、C12、C13 都是滤波电容，保证主芯片供电稳定。剩下的电路就是主 IC 的引脚，分别接到扩展口和外设上。

H、外部电池供电电路



外部电池供电电路

图 11. 外部电池供电原理图

其中 C14 为滤波电容，VBAT 接到主芯片电池供电引脚上，这是为了给 RTC 实时时钟模块供电。一旦主芯片断电，使用外部电池供电可以保证 RTC 实时时钟不间断或者不被复位。此电路就是起滤波作用，保证电池供电电压稳定。

I、按键电路

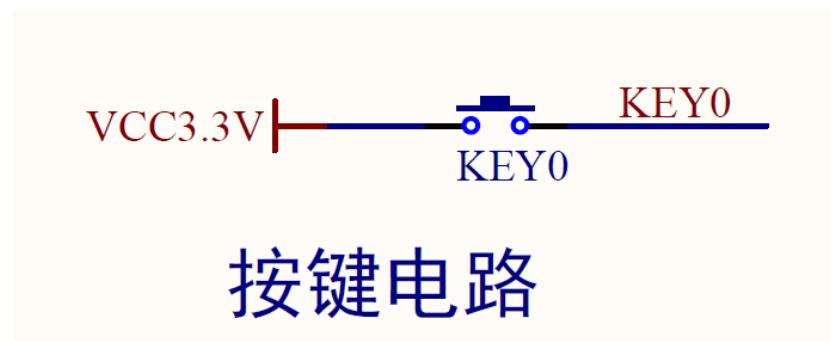
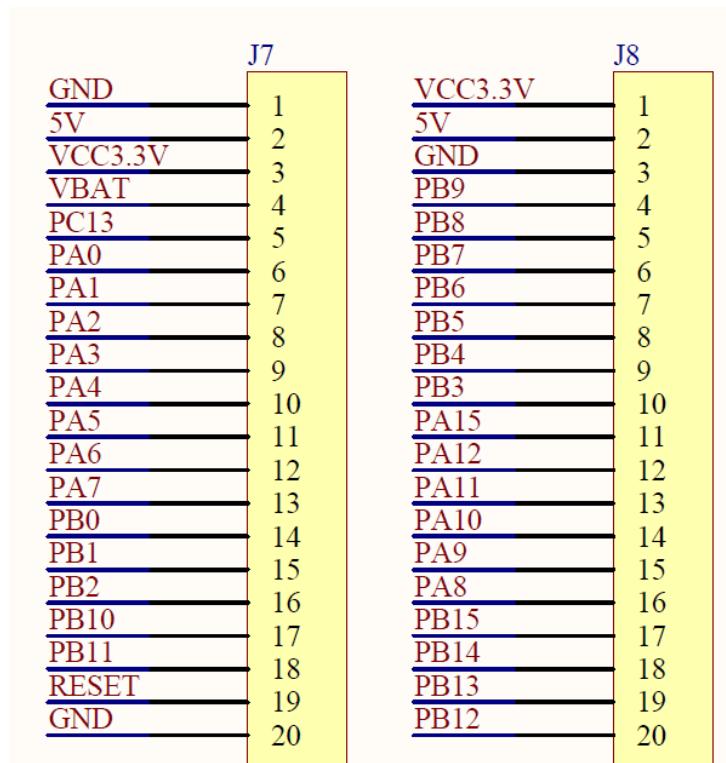


图 12. 按键原理图

其中 KEY0 按键接在主芯片的 PA0 引脚上，默认为下拉输入，按键按下时，引脚会变成上拉输入，根据此状态就可以判断按键是否按下。

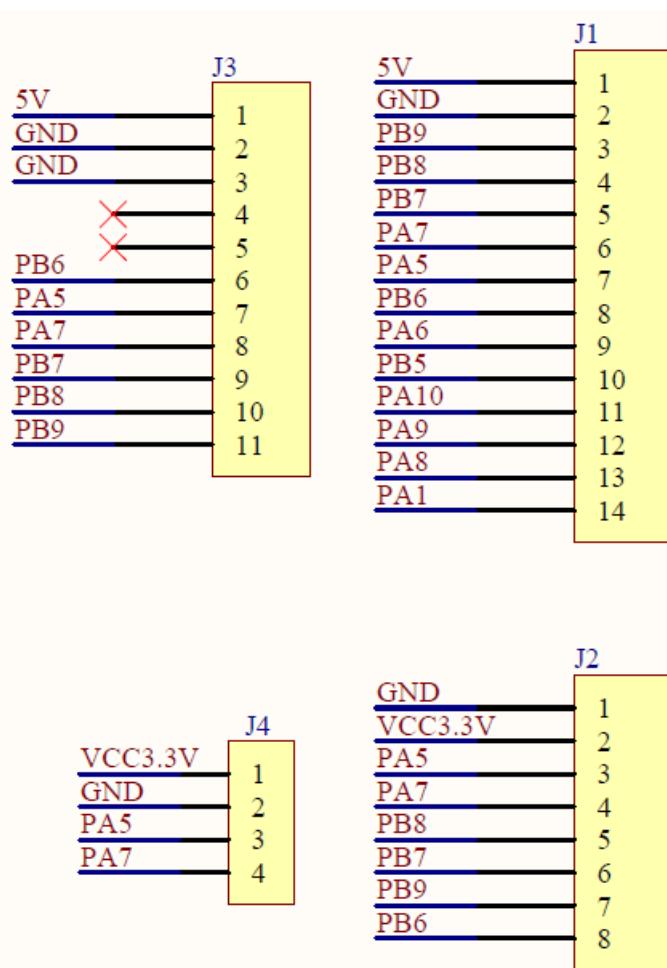
J、外部扩展 I/O 接口电路



外部扩展 I/O 接口

图 13. 按键原理图

K、液晶屏模块接口电路



液晶模块接口

图 13. 液晶屏模块接口原理图

其中：

- ✧ J1 接口用于直插普通的使用 SPI 接口的 LCD 模块，包括带触摸和不带触摸功能的
- ✧ J2 接口用于直插使用 SPI 接口的 IPS 屏模块和 OLED 模块，另外还兼容直插第一引脚为 GND 的使用 IIC 接口的 OLED 模块
- ✧ J3 接口用于直插在 Arduino 单片机上使用的 SPI 接口的 LCD 模块
- ✧ J4 接口用于直插第一引脚为 VCC 的使用 IIC 接口的 OLED 模块

4、程序下载

STM32F103C8T6 最小系统板提供了 SWD 下载、串口下载两种下载方式。关于此两种下载方式的具体说明见资料包中下载方法文件夹。

5、示例说明

暂时提供了 5 个测试示例：

- A、LED 灯测试；
- B、按键测试；
- C、LCD 测试（包含触摸功能）；
- D、RTC 测试；
- E、串口测试；

A、LED 测试

此程序用于测试 GPIO 口输出功能，控制 LED0 闪烁。当程序运行时，LED0 会持续亮 500ms 和灭 500ms。

B、按键测试

此程序用于测试 GPIO 口输入功能，使用 KEY0 按键控制 LED0 亮和灭。当程序运行时，按一次 KEY0，LED0 就会亮，再按一次 KEY0，LED0 就会灭。

C、LCD 测试（包含触摸功能）

此程序用于测试硬件 SPI 功能和 GPIO 输入输出功能。使用 2.8 寸 ILI9341 模块，直插到 J1 接口，当程序运行时，LCD 会有相应内容的显示，触摸屏也会有反应。值得注意的地方是触摸屏的校准参数保存在内部 flash 的最后 20 个字节里面，如果运行的程序需要用到最后 20 个字节，那么校准参数就不能再写入 flash 了。只能在程序里面写入变量，否则会导致程序运行不正常。

D、RTC 测试

此程序用于测试 RTC 实时时钟显示，将实时时钟的时分秒、日期、星期等信息通过 LCD 显示出来。

E、串口测试

此程序用于测试串口读写功能。使用 XCOM 串口调试工具实现和最小系统板进行串口通信。