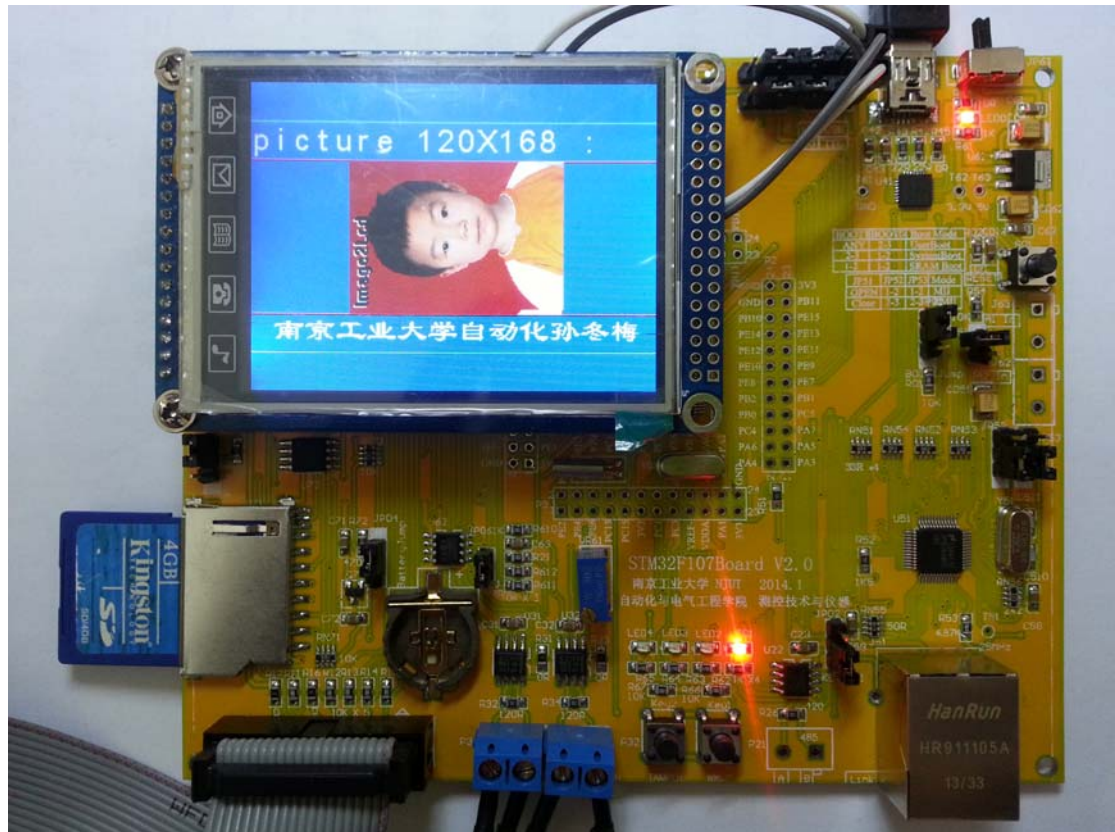


1.外形

板正面



2.硬件系统结构及使用 IO 口资源

板上主要芯片资源:

STM32f107VC (CPU)

1117-33

CP2102

W25Q16B W25×16 (16M Flash 4M*8bit)

AtmelHD012 (24C02) E2PROM(1K-128*8bit)

SP3485

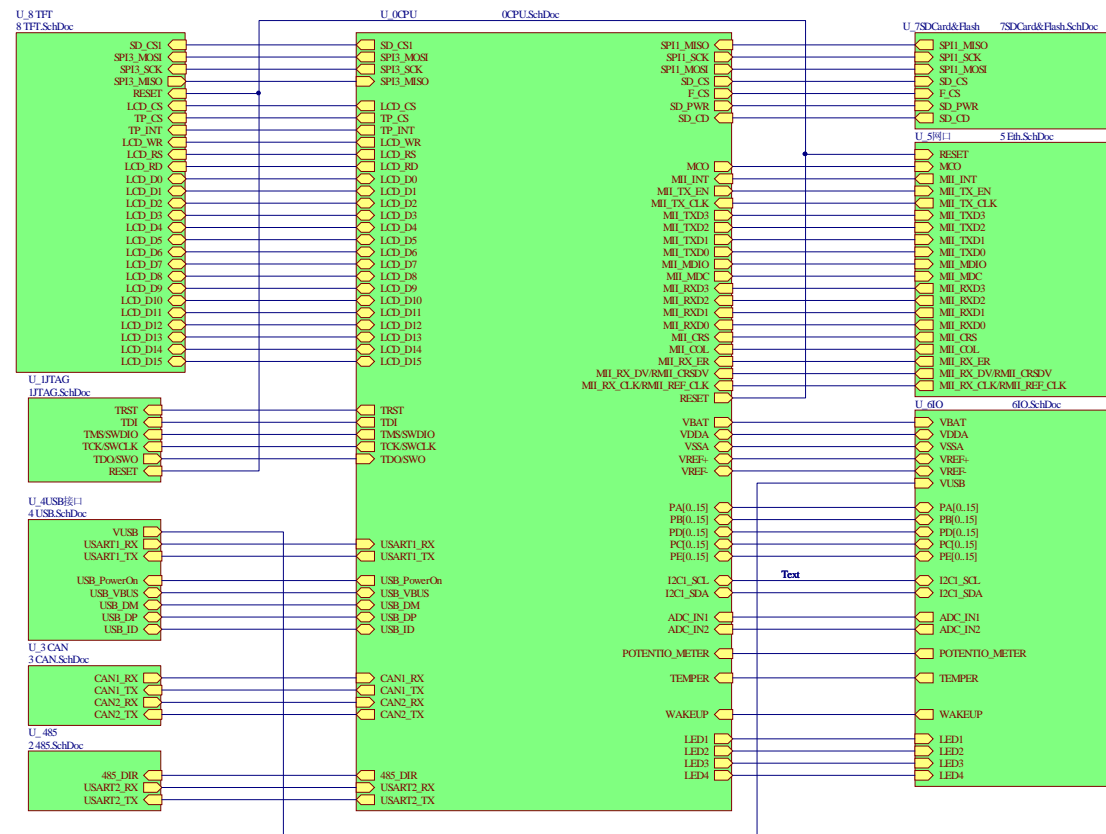
DP83848

VP230

板上接口、开关、灯:

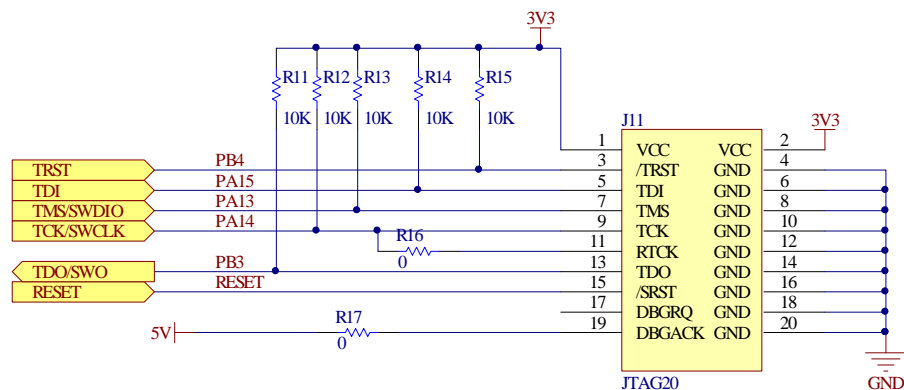
DC 电源, USB, SD 卡, JTAG, CR1220, RJ45, 按键 (复位)、按键*2、485 接口, 电源指示灯, LED×4

硬件系统结构:



3.1 下载模块

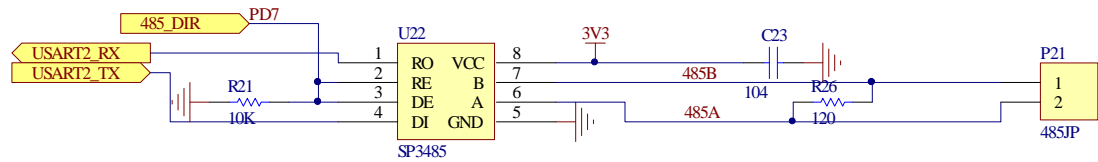
J11 为 JTAG 下载调试接口。



3.2 485 接口

P21：1 路 485 接口，采用芯片 3485 转接。

方向控制口： PD7



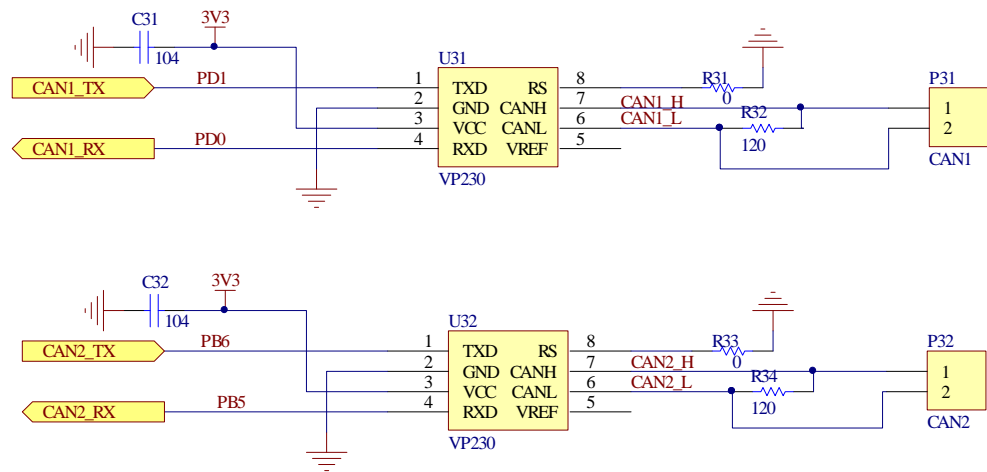
关断：RE=1 DE=0

接收：RE=0 DE=0

发送：RE=1 DE=1

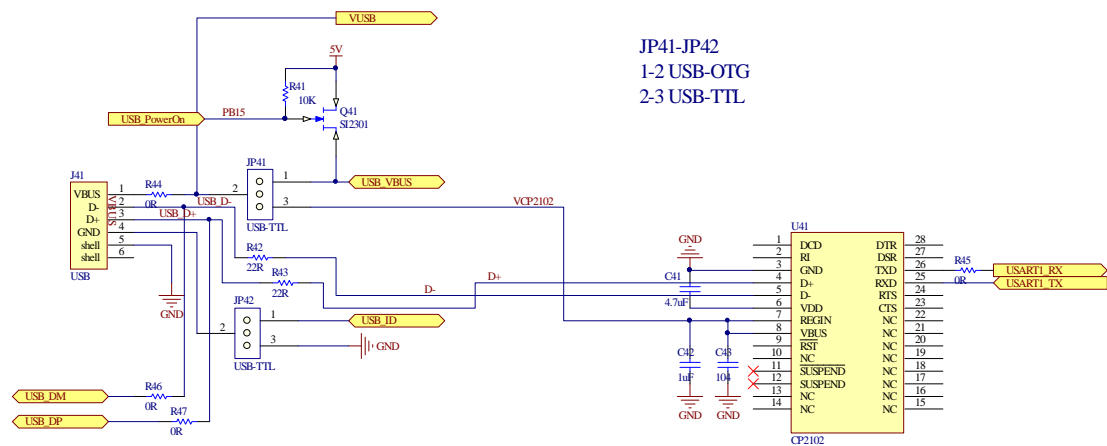
连接引脚	方向	功 能 描 述
PD7	OUT	485 通道 1 读写控制， 0:读； 1： 写
PD6	IN	USART2_RX 接收
PD5	OUT	USART2_TX 发送

3.3 CAN 总线接口



连接引脚	方向	功 能 描 述
PD0	IN	CAN1 接收
PD1	OUT	CAN1 发送
PB6	IN	CAN2 接收
PB5	OUT	CAN2 发送

3.4 USB 模块



JP41-JP42
1-2 USB-OTG
2-3 USB-TTL

JP41-JP41

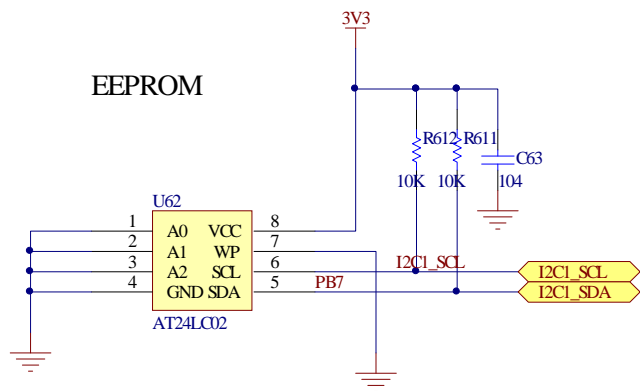
1-2 连接使用 USBOTG

2-3 连接使用 CP2102 USB 转串口的数据通信

连接引脚	方向	功能描述
VBUS		USB 口取电
PA9	OUT	UART1 TX/USB_VBUS
PA10	IN	UART1 RX/USB_ID
PA11	IN OUT	USB_DM
PA12	IN OUT	USB_DP

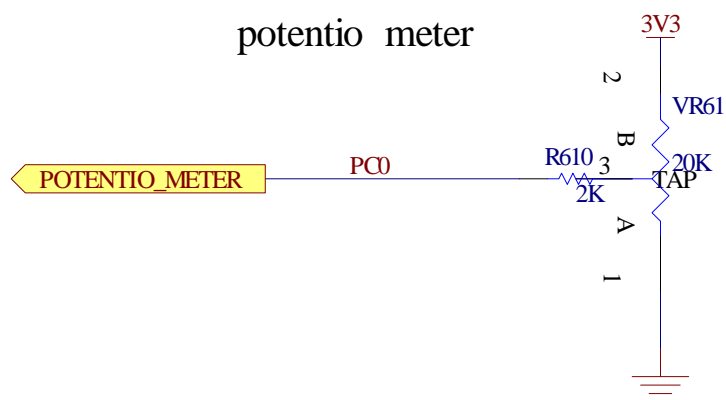
3.5 网口模块

J51: RJ45，带网络变压器的网络接口。



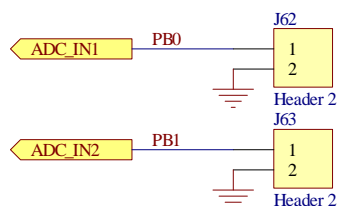
连接引脚	方向	功 能 描 述
PB6	OUT	I2C 总线 CLK 需要上拉
PB7	INOUT	I2C 总线 数据线 需要上拉

电位器: 用于模块输入



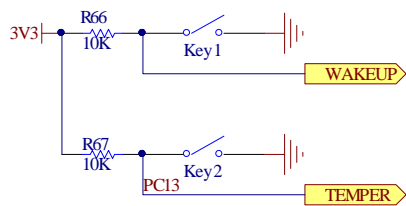
连接引脚	方向	功 能 描 述
PC0	IN	输入模拟量接口

模拟输入接口:



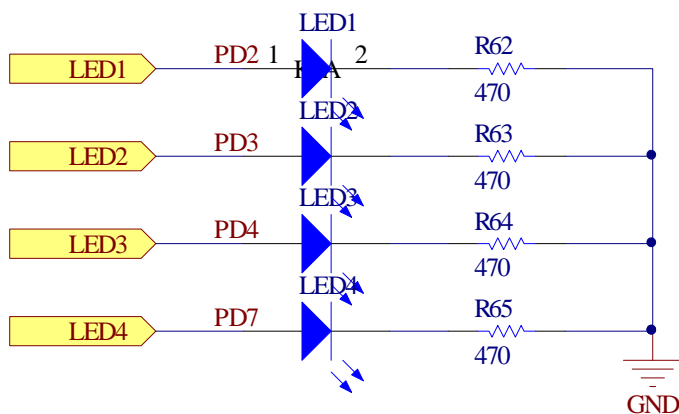
连接引脚	方向	功 能 描 述
PB0	IN	输入模拟量接口
PB1	IN	输入模拟量接口

按键: 配置成上拉输入



连接引脚	方向	功 能 描 述
PA0	IN	输入数字量接口
PC13	IN	输入数字量接口

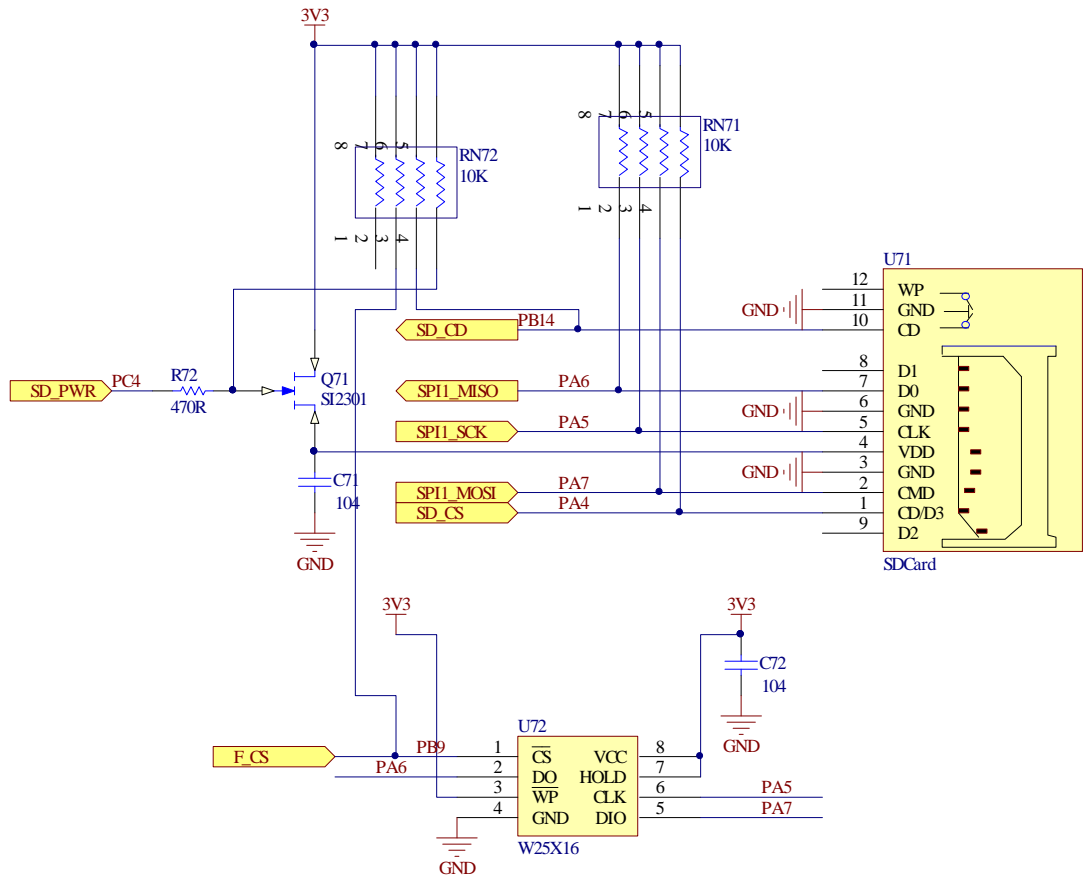
LED 指示: 必须配置与推挽输出



连接引脚	方向	功 能 描 述
PD2	OUT	输出数字量接口
PD3	OUT	输出数字量接口
PD4	OUT	输出数字量接口
PD7	OUT	输出数字量接口

3.7 SD 及 Flash 模块

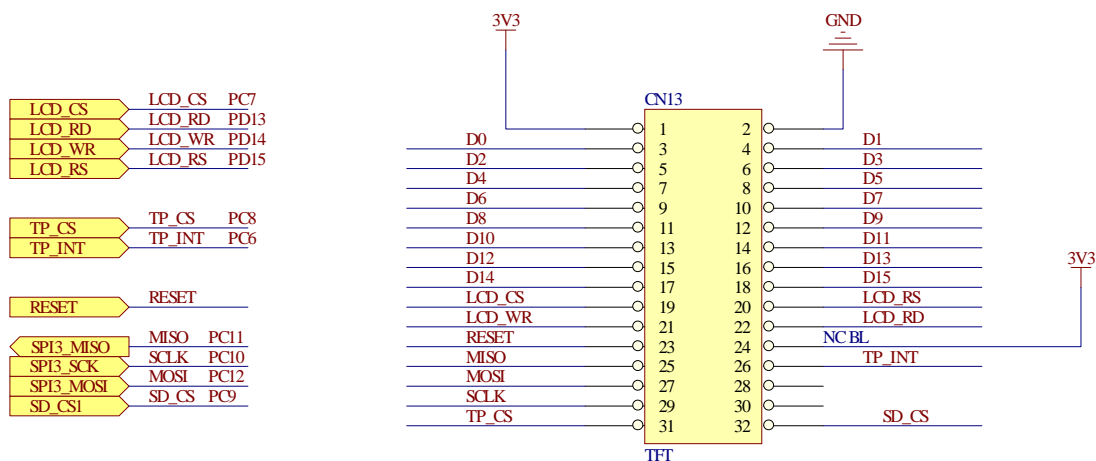
U71: 为 SD 卡座。



连接引脚	方向	功 能 描 述
PC4	OUT	SD 卡座电源 0：上电 1：下电
PB14	IN	SD 卡检测 1：无卡 0：有卡
PA6	IN	SPI 总线 MISO
PA5	OUT	SPI 总线 CLK
PA7	OUT	SPI 总线 MOSI
PA4	OUT	SD 卡片选信号 0 有效
PB9	OUT	W25X16 片选信号 0 有效

3.8 TFT 接口

TFT 采用 SSD1289 控制器



3.板上接口跳线说明(阴影为缺省状态)

网口工作模式 MII/RMII

JP51	JP52	JP53	Mode
OPEN	1-2	1-2	MI
Close	2-3	2-3	RMII

系统启动模式

BOOT1	BOOT0	BOOT MODE
ANY	2-3	User Boot
2-3	1-2	System Boot
1-2	1-2	SRAM_Boot

系统内部时钟供电方式:

JP04	供电方式
1-2	板上电源供电
2-3	CR1220 电池供电

USB 接口工作模式

JP05	JP07	JP41	JP42	USB MODE
1-2	1-2	1-2	1-2	USB-OTG
2-3	2-3	2-3	2-3	USB-TTL（缺省）

PA0 功能:

JP02	供电方式
1-2	按键 wakeup
2-3	可用网络的 MII 工作模式（缺省）

PB6 功能:

JP06	PB6 管脚用
短接	I2C1_SCL
断开	CAN2_TX（缺省）

4.开发板运行说明

4.1 正常运行说明

（1）上电

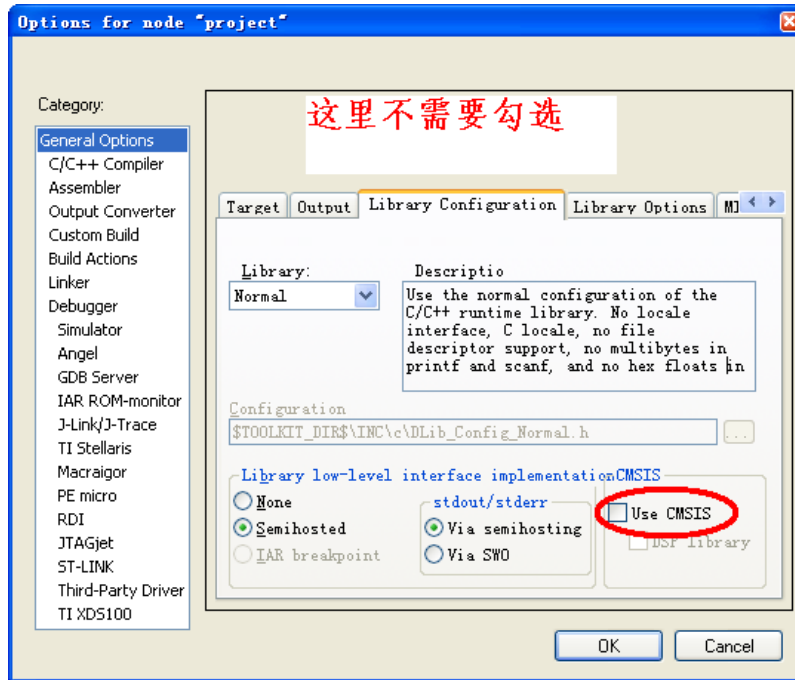
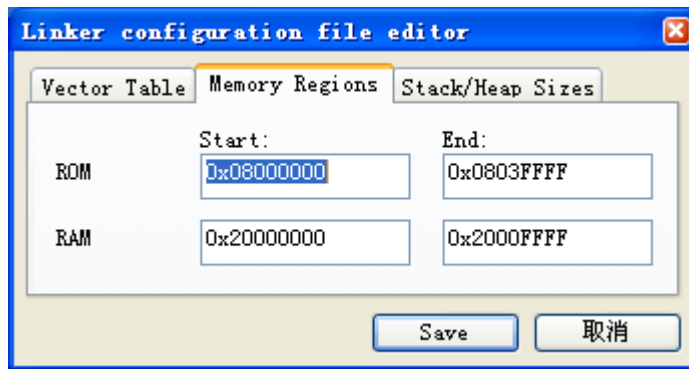
上电后,5V 电源指示灯亮

（2）下载程序,采用以下模式，程序下载到 flash 中运行

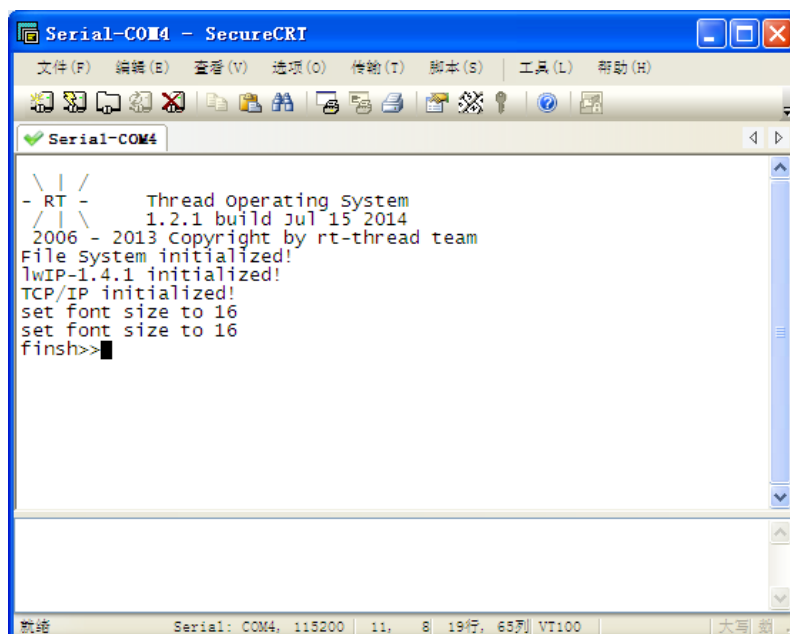
系统启动模式

BOOT1	BOOT0	BOOT MODE
ANY	2-3	User-Boot（缺省）

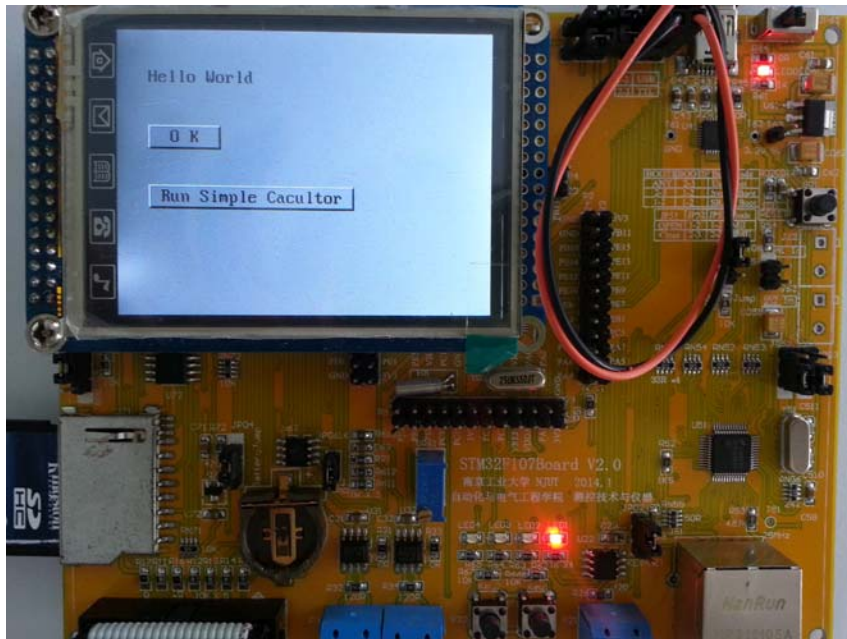
（3）工程配置



(4) 将开发板固件.hex 下载至开发板后，打开串口监视助手，115200 波特率，连接成功后上电。



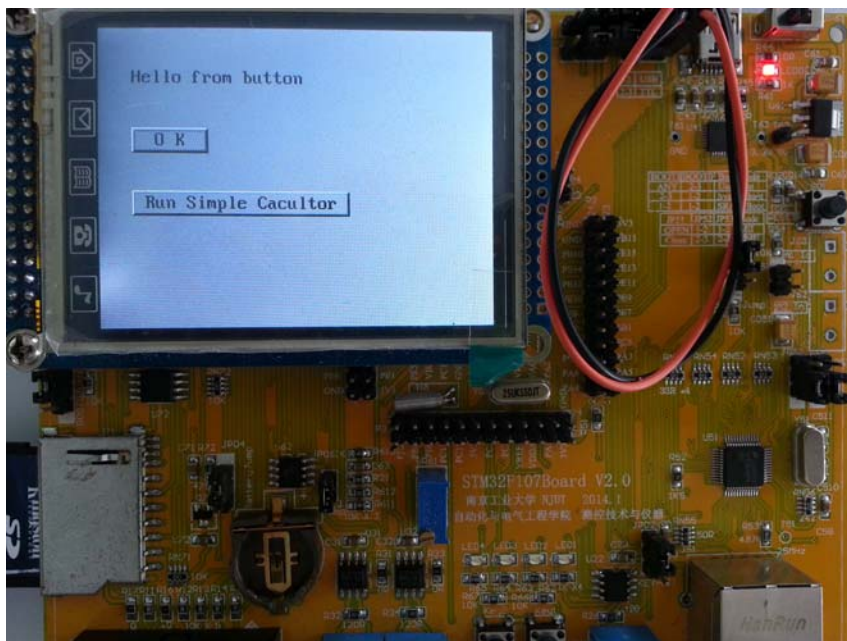
1) 显示 button 界面



点击 **OK** 后，

串口助手显示：

```
finsh>>touch->x:1571 touch->y:833  
touch down: (57, 94)  
touch->x:1570 touch->y:833  
touch->x:1572 touch->y:832  
touch->x:1574 touch->y:829  
touch up: (56, 94)  
hello, button!
```



点击 **Run Simple Cacultor** 后，

```
touch up: (80, 143)
```

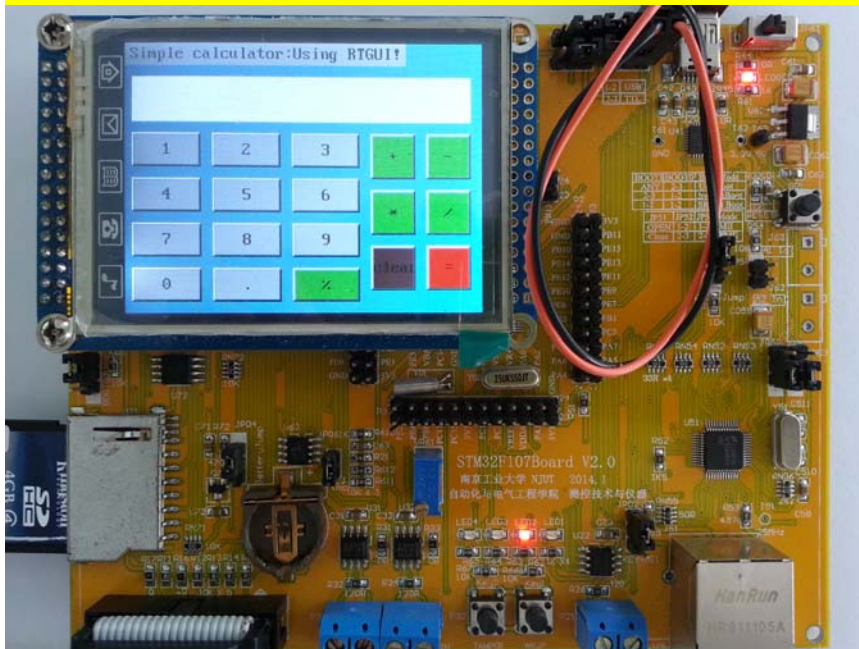
Oh! Run caculator !

event->type:18

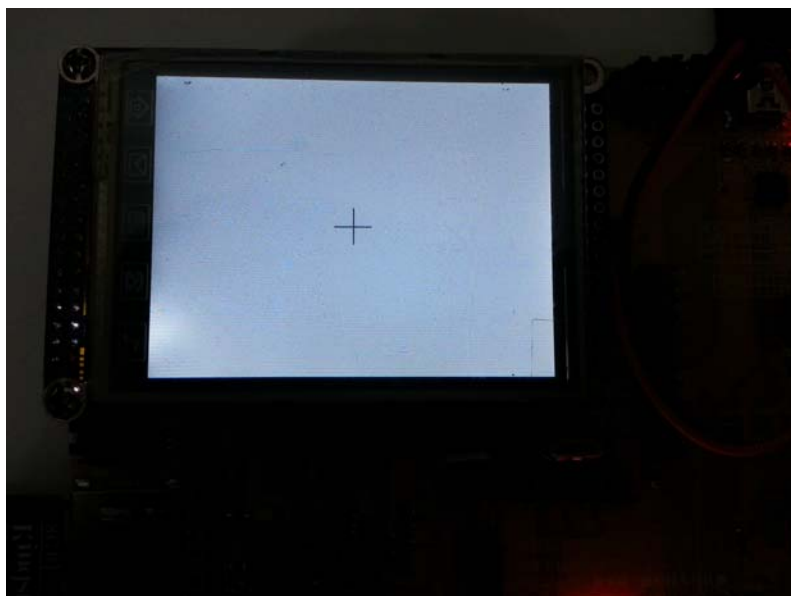
event->type:23

event->type:7

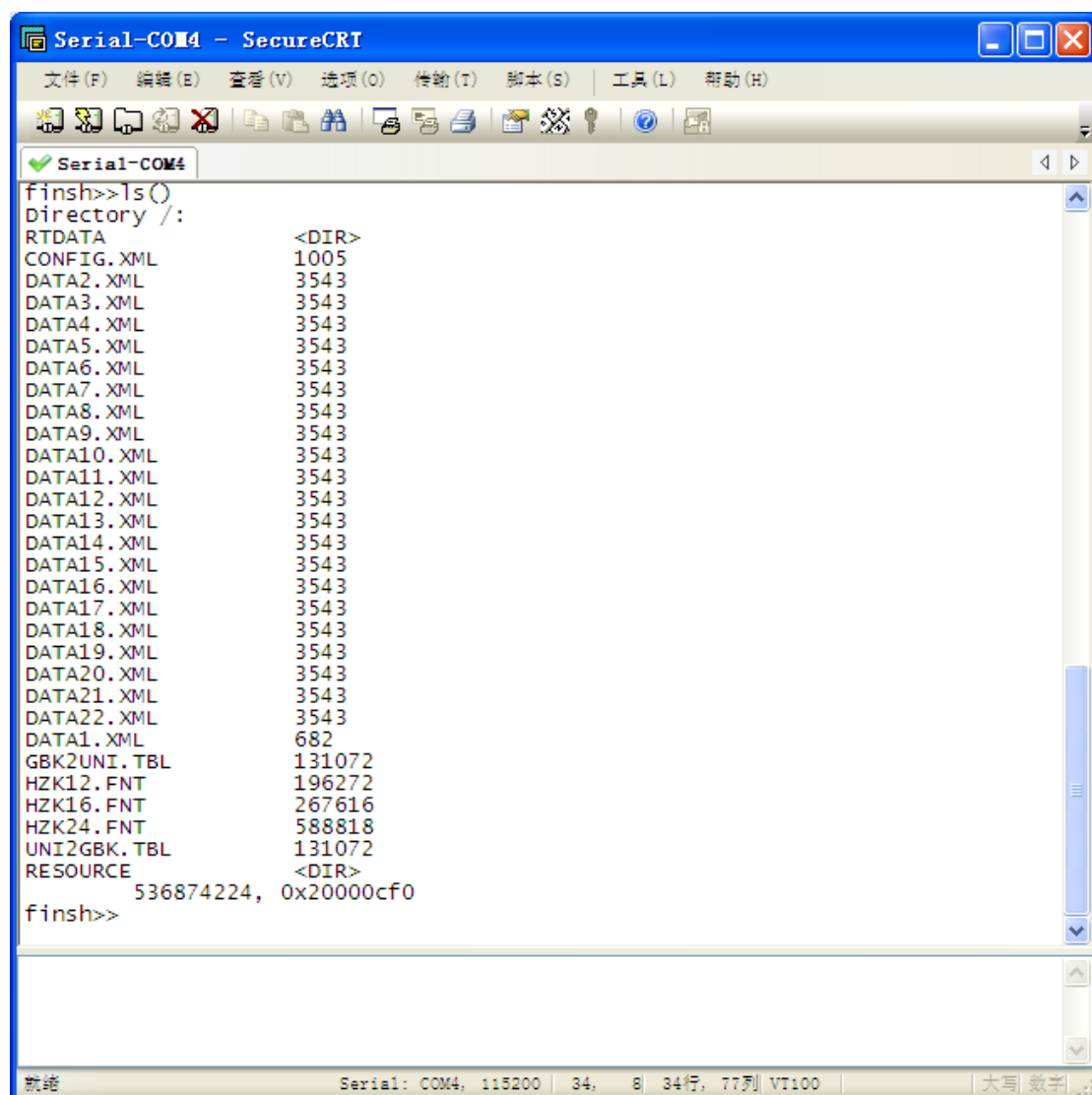
event->type:20



在串口助手中输入命令，`calibration()`后，可进行触摸校准。



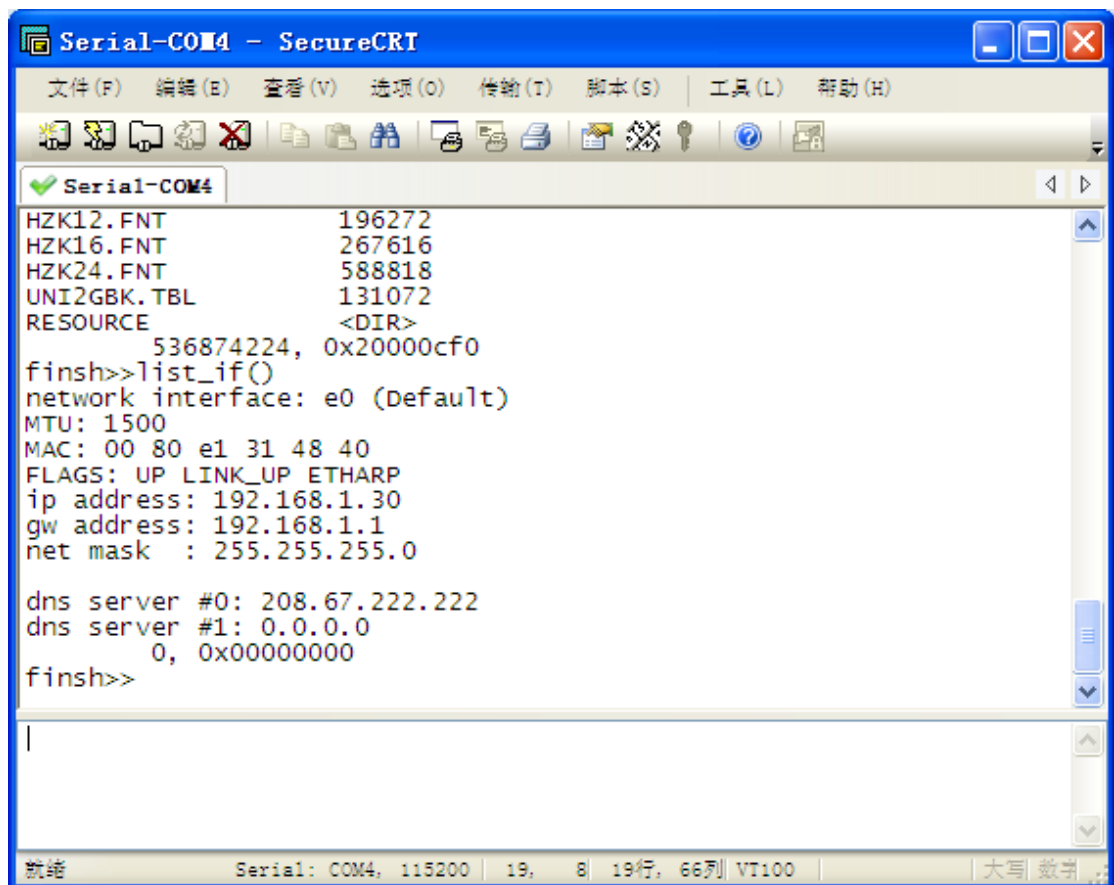
在串口助手中输入 `ls()`,可显示当前 SD 卡中文件



```
Serial-COM4 - SecureCRT
文件(F) 编辑(E) 查看(V) 选项(O) 传输(T) 脚本(S) 工具(L) 帮助(H)
Serial-COM4
finsh>>ls()
Directory /:
RTDATA                <DIR>
CONFIG.XML            1005
DATA2.XML              3543
DATA3.XML              3543
DATA4.XML              3543
DATA5.XML              3543
DATA6.XML              3543
DATA7.XML              3543
DATA8.XML              3543
DATA9.XML              3543
DATA10.XML             3543
DATA11.XML             3543
DATA12.XML             3543
DATA13.XML             3543
DATA14.XML             3543
DATA15.XML             3543
DATA16.XML             3543
DATA17.XML             3543
DATA18.XML             3543
DATA19.XML             3543
DATA20.XML             3543
DATA21.XML             3543
DATA22.XML             3543
DATA1.XML              682
GBK2UNI.TBL            131072
HZK12.FNT              196272
HZK16.FNT              267616
HZK24.FNT              588818
UNI2GBK.TBL            131072
RESOURCE               <DIR>
536874224, 0x20000cf0
finsh>>
```

就绪 Serial: COM4, 115200 34, 8 34行, 77列 VT100 大写 数字

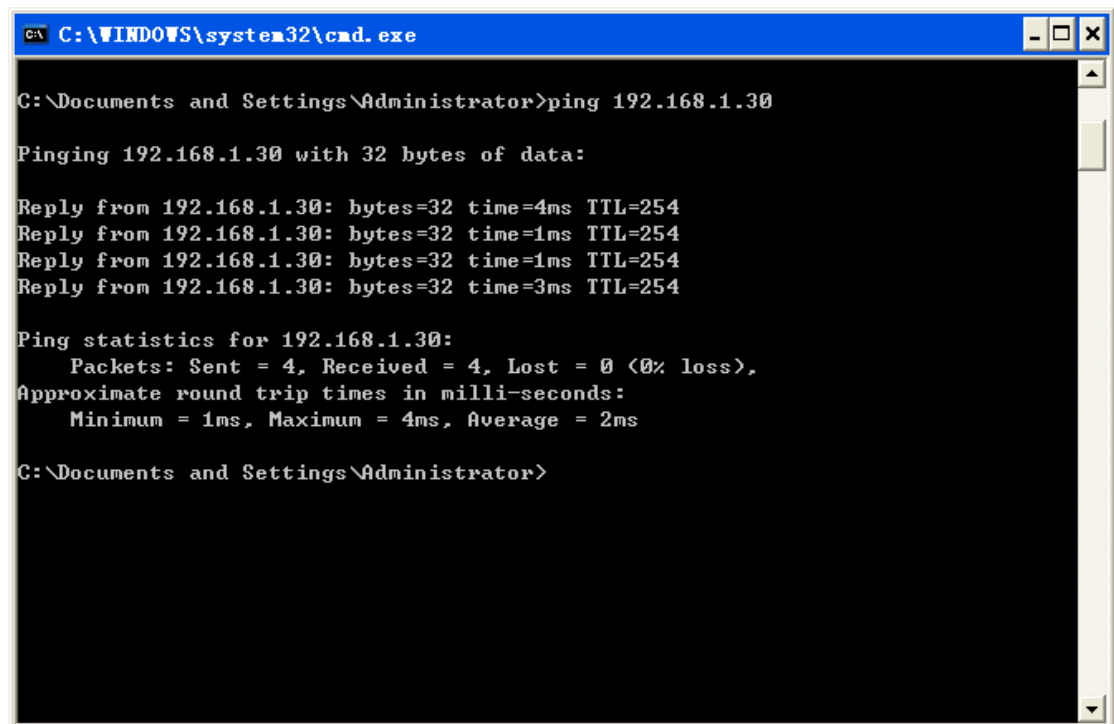
在串口助手中输入命令，显示当前网口信息：



```
Serial-COM4 - SecureCRT
文件(F) 编辑(E) 查看(V) 选项(O) 传输(T) 脚本(S) 工具(L) 帮助(H)
Serial-COM4
HZK12.FNT          196272
HZK16.FNT          267616
HZK24.FNT          588818
UNI2GBK.TBL        131072
RESOURCE            <DIR>
                    536874224, 0x20000cf0
finsh>>list_if()
network interface: e0 (Default)
MTU: 1500
MAC: 00 80 e1 31 48 40
FLAGS: UP LINK_UP ETHARP
ip address: 192.168.1.30
gw address: 192.168.1.1
net mask  : 255.255.255.0

dns server #0: 208.67.222.222
dns server #1: 0.0.0.0
              0, 0x00000000
finsh>>
```

连接网线至 PC 机，将 PC 机有线网口端口设置成 192.168.1.0-255 中任何一个（除去 30），后打开 cmd，用 ping 命令去查找开发板。能 ping 通，则正常。



```
C:\WINDOWS\system32\cmd.exe
C:\Documents and Settings\Administrator>ping 192.168.1.30

Pinging 192.168.1.30 with 32 bytes of data:

Reply from 192.168.1.30: bytes=32 time=4ms TTL=254
Reply from 192.168.1.30: bytes=32 time=1ms TTL=254
Reply from 192.168.1.30: bytes=32 time=1ms TTL=254
Reply from 192.168.1.30: bytes=32 time=3ms TTL=254

Ping statistics for 192.168.1.30:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 1ms, Maximum = 4ms, Average = 2ms

C:\Documents and Settings\Administrator>
```

关于 finsh ()的命令还有好多，自己去琢磨吧 ☺!!!

附录:

Finsh 的由来:

现有 F 语言，然后才有应用到 shell 中，所以是 F in shell，简称 finsh.