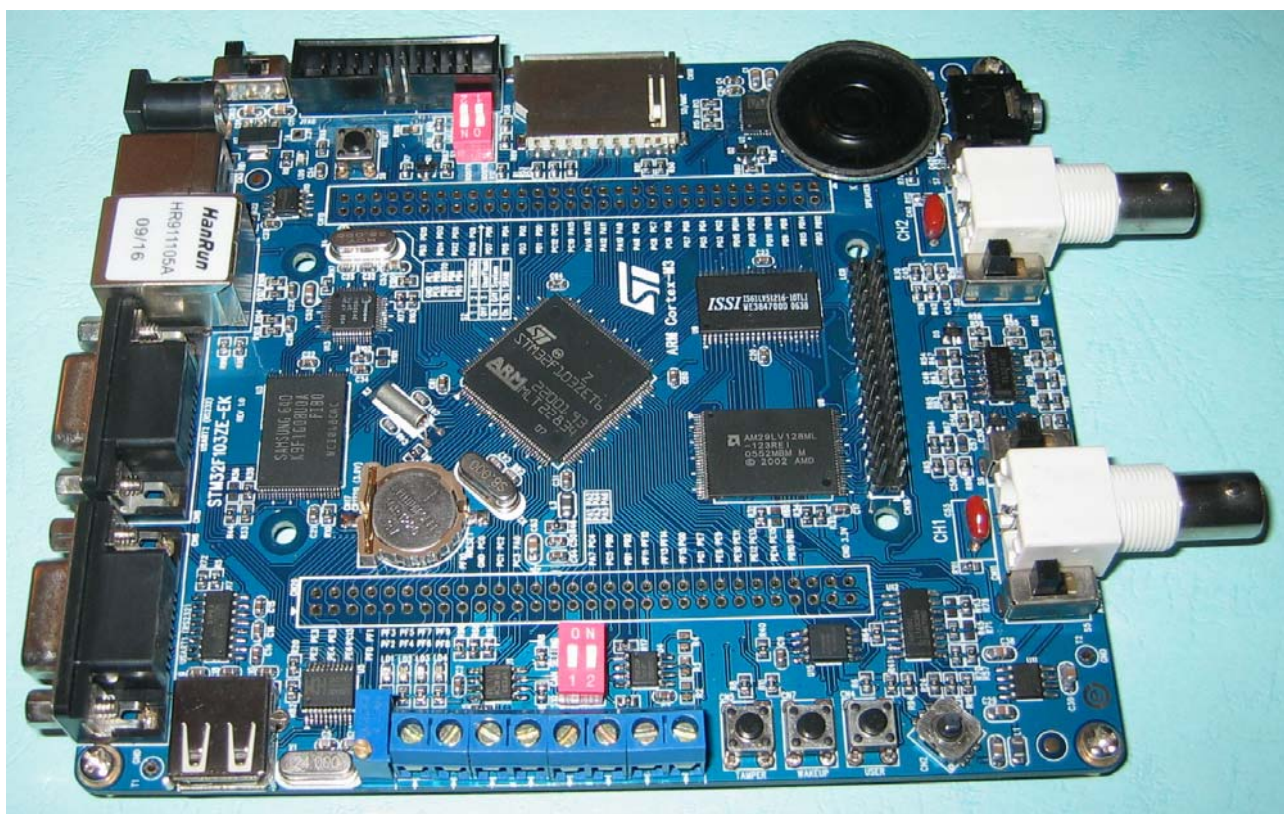


STM32F103ZE-EK开发板

用户手册

版本：V1.0



安富莱电子开发网

WWW.ARMFLY.COM

1. 产品规格

简介

STM32F103ZE-EK开发板以**STM32F103ZET6** (LQFP144) 为核心。

STM32F103ZE 是ST (意法半导体) 公司推出的ARM Cortex-M3产品线中功能最强大的一款CPU。片内集成512kB Flash、64kB RAM、1个USB、1个CAN、8个定时器、5个USART、3个ADC、2个DAC、3个SPI、2个I2C、2个I2S、1个SDIO、112个GPIO、FSMC总线 (支持NOR,NAND,SRAM) 。CPU主频72MHz,广泛适用于各种应用场合。

本开发板具备丰富的硬件资源,配套的试验例程均提供源代码,文档齐备,非常适合于学习和项目评估。

硬件资源

- 8M晶振作为MCU的时钟, 32768晶振用于RTC
- **1M字节SRAM, 16M字节NOR Flash, 128M字节NAND Flash**
- 2M字节串行Flash, 256字节串行EEPROM
- 1个SD/MMC卡座
- 1个CAN2.0A/B接口
- 2个RS232串口
- 1个RS485接口
- 1个USB2.0全速DEVICE接口
- **1个USB2.0全速HOST接口**
- **1个100M/10M以太网接口**
- **I2S音频DAC (24bit, 96kHz)**, 1个立体声耳机插座, 1个扬声器
- 3.0寸TFT真彩触摸LCD (**WQVGA, 400x240**)
- 1个5向摇杆, 1个Reset按钮、1个wakeUp按钮、1个自定义按钮

- 4个自定义LED, 1个电源LED, 1个音频LED
- 1个CR1220电池座
- 1个精密可调电阻连接到ADC输入
- 所有的GPIO引到2.54mm间距焊盘
- 1个DAC引出端子, 1个PWM引出端子
- 标准2.54mm间距JTAG插座
- **2个BNC输入端子, 集成双通道示波器电路, 具备AC/DC切换、输入增益切换开关**
- 3种供电方式: USB电缆、外接5V电源、JTAG调试接口 (J-LINK仿真器)
- 1个电源开关, 上下电时无需拔插电缆
- 3种启动方式: 用户Flash、系统存储器、SRAM
- **用拨码开关取代跳线帽, 避免跳线帽丢失**
- 板子规格: 14cm x 12cm

软件资源

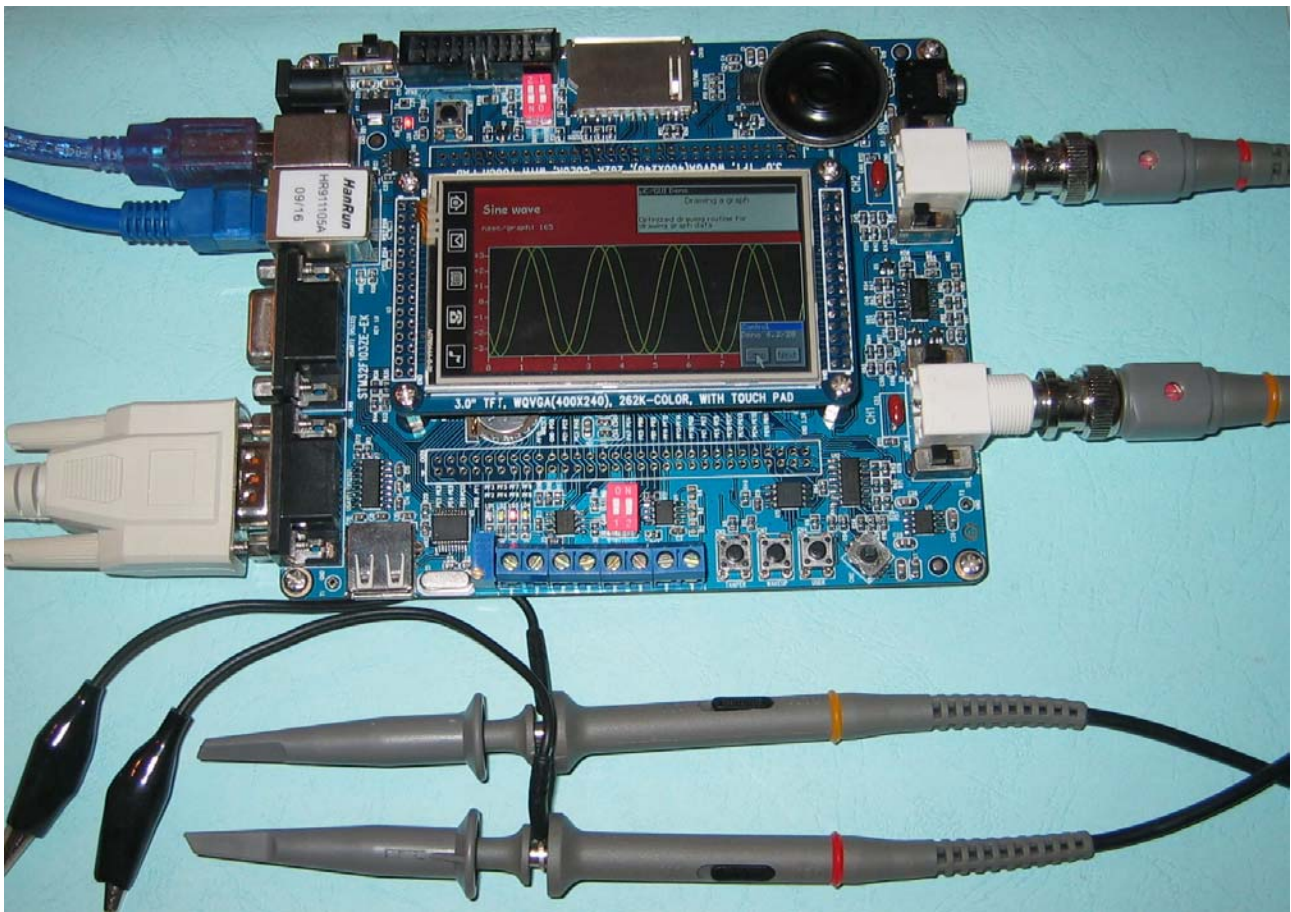
- 提供100多个试验例程
- **提供uCOS_II+ucGUI例程和文档**
- 即将展开USB虚拟示波器项目源码
- 即将移植ucLinux (硬件资源已满足要求)
- 更多的软件资源将在www.armfly.com发布

标配清单

- STM32F103ZE-EK开发板1块
- 3.0寸TFT触摸显示模块1块
- 1根串口线、1根网线、1根USB电缆
- 资料光盘1张

可选的配件:

- 60M示波器探头1对
- USB转串口线1根



2. 快速入门

2.1. 注意事项

- (1) 外接电源必须是 5.0V 的直流电源，插头有极性，内正外负。（一般情况下用 USB 电缆供电）
- (2) 板子放置的桌面区域不要有金属物品，以免造成短路。
- (3) 触摸屏属于易碎物品，请注意保护。
- (4) 扬声器纸盆容易被硬物戳伤，请注意保护。

2.2. 检查硬件功能

我们设计了一个测试程序专门用于测试板子的硬件功能。出厂的板子均预先烧录了这个测试程序。如果你拿到的板子没有预先烧写这个测试程序，请按照的“烧写说明”步骤进行操作。

这个测试程序存放光盘上：\05.例程镜像\ HardTest.bin

上电前的准备工作：

- (1) 插上串口线（使用USB转串口的线也行），连接开发板的COM1口至PC机的串口。
- (2) 插上网线（交叉网线和直连网线均可），连接开发板的网口至交换机端口或者电脑内置网卡。
- (3) 打开串口工具或超级终端（波特率115200，1个起始位，1个停止位，8个数据位，无校验，无硬件流控）。

确保开发板周围没有可能造成板子短路物品。插上USB电缆，打开开发板上的电源开关（S1），电源指示灯LD5将点亮。

开发板LCD上显示如下界面：（中间白色区域是320x240，两侧的紫色带是40像素宽）



PC机超级终端上显示如下内容（----红色字体是笔者添加的注释）：

```
*****
* STM32F103ZE-EK Evaluate Kit
* www.armfly.com co Copyright 2009
*****
CPU Reset
CPU : STM32F103ZET6, LQFP144, SN = FFFFFFF36 36393658 43094138 ----CPU序号，每个板子均不一样
Testing the SRAM, Address = 0x68000000, Size = 1M Bytes ----测试外部SRAM, 1MB空间全扫描
The SRAM is OK
LCD Detected (SPFD5420A) 3.0" TFT LCD, WQVGA 400x240, ----检测到LCD驱动芯片
DM9000AE Detect Ok, vid&pid = 0A469000 ----检测到网卡芯片
CH374T Detect Ok ----检测到USB Host驱动芯片
Nor Flash ID = 0001, 227E, 2212, 2200 ----检测到NOR Flash, 显示ID
Nor Flash Type : AM29LV128ML
Nand Flash ID = EC, F1, 80, 15 ----检测到NAND Flash, 显示ID
Nand Flash Type : K9F1G08U0A
Testing CAN ..... ----检测CAN芯片
```

CAN is OK	
Testing RS485	-----检测RS485芯片
RS485 is OK	
Display the STM32 introduction	-----TFT上显示ST公司的LOGO
Display the main menu	-----TFT上显示主菜单图标

测试项目列表：

序号	测试对象	说明
1	串口1	判据：观察串口1打印信息。 说明：很多测试信息将打印到串口1
2	串口2	判据：将PC机串口连接到开发板的COM2，然后按开发板复位按键。如果超级终端接收到“COM2”则表示COM2硬件正常。 说明：程序运行时会向COM2输出一串信息。
3	NAND Flash	判据：观察串口打印信息。 说明：读取芯片ID，如果正确则表示硬件无问题
4	NOR Flash	判据：观察串口打印信息。关注图标显示和音频播放。 说明：读取芯片ID，如果正确则表示硬件连接无问题。主界面显示的图片以及wav音频文件均存放在NOR Flash(约15M字节)
5	外部SRAM	判据：观察串口打印信息。 说明：对整个1M字节空间进行写读测试
6	串行Flash	判据：观察串口打印信息 说明：读取芯片ID，如果正确则表示硬件无问题
7	串行EEPROM	修改IP地址，IP地址将保存到EEPROM
8	RS485芯片	判据：观察串口打印信息 原理：同时使能RS485芯片的接收和发送，然后发送1个字节，如果接收到这个字节，表示硬件无问题。
9	CAN芯片	判据：观察串口打印信息 说明：同时使能RS485芯片的接收和发送，然后发送1个字节，如果接收到这个字节，表示硬件无问题。
10	SD卡测试	操作： 1. 在SD卡座插入一张2G以下的SD卡 2. 主菜单-USB Mass Storages – Start

		3. 在windows文件浏览器中可以看到SD卡设备，并可以读取或修改SD卡内文件 说明： (1) 目前仅支持小容量的SD卡，2G或2G以上的卡的支持还存在问题。 (2) NAND Flash的支持也存在问题。
11	网卡	操作： 4. 主菜单-Ethernet-IP Address，修改IP地址（缺省是192.168.1.10） 5. 主菜单-Ethernet-Web Server 6. 在IE浏览器输入http://192.168.1.10 7. 在windows CMD状态，执行ping 192.168.1.10 判据：IE浏览器显示web页面；观察ping命令执行结果。 说明：本程序移植了uIP协议栈
12	音频测试	播放一段WAV格式解说语音，检查耳机输出和扬声器输出
13	USB HOST测试	操作： 1、主菜单-USB Host – USB Disk 2、插入U盘，进行读写文件测试 3、通过串口打印的信息观察测试结果。 说明：这个例子支持FAT文件系统，插入U盘后，自动显示根目录下所有的文件名，然后将一个文本文件写到U盘根目录。
14	示波器测试	通过DAC_OUT1输出频率1KHz，1V的方波信号，用示波器探头去探测，然后在LCD上显示。 说明： 改功能还未实现
15	USB接口	按下USER 按键，然后复位开发板，windows发现一个USB硬件设备插入，这是DFU升级程序再运行。
16	JTAG接口	通过仿真器验证

2.3. 跳线配置

为了便于初学者使用，本开发尽可能地减少了配置跳线，仅有2个拨码开关。

拨码开关S2选择CPU的启动方式

位1状态	位2状态	启动模式
OFF	无关	用户闪存存储器启动
ON	OFF	从系统存储器启动
ON	ON	从内嵌SRAM启动

拨码开关S10用于设置CAN接口

位1状态	控制120欧负载
OFF	断开CAN总线并接的120欧负载电阻
ON	连接120欧负载电阻

位2状态	控制CAN芯片的RS口线状态
OFF	CAN芯片工作与高速模式
ON	CAN芯片工作与待机模式

2.4. 学习建议

本产品的售后和技术支持方法：

- (1) 到BBS上提问。 <http://www.armfly.com/bbs/index.php>
- (2) QQ邮件交流。 armfly@qq.com

本开发板的官方资料下载地址：

<http://www.armfly.com/bbs/forumdisplay.php?fid=2>

ST（意法半导体）官方资料下载地址：

<http://www.st.com/mcu/familiesdocs-110.html>

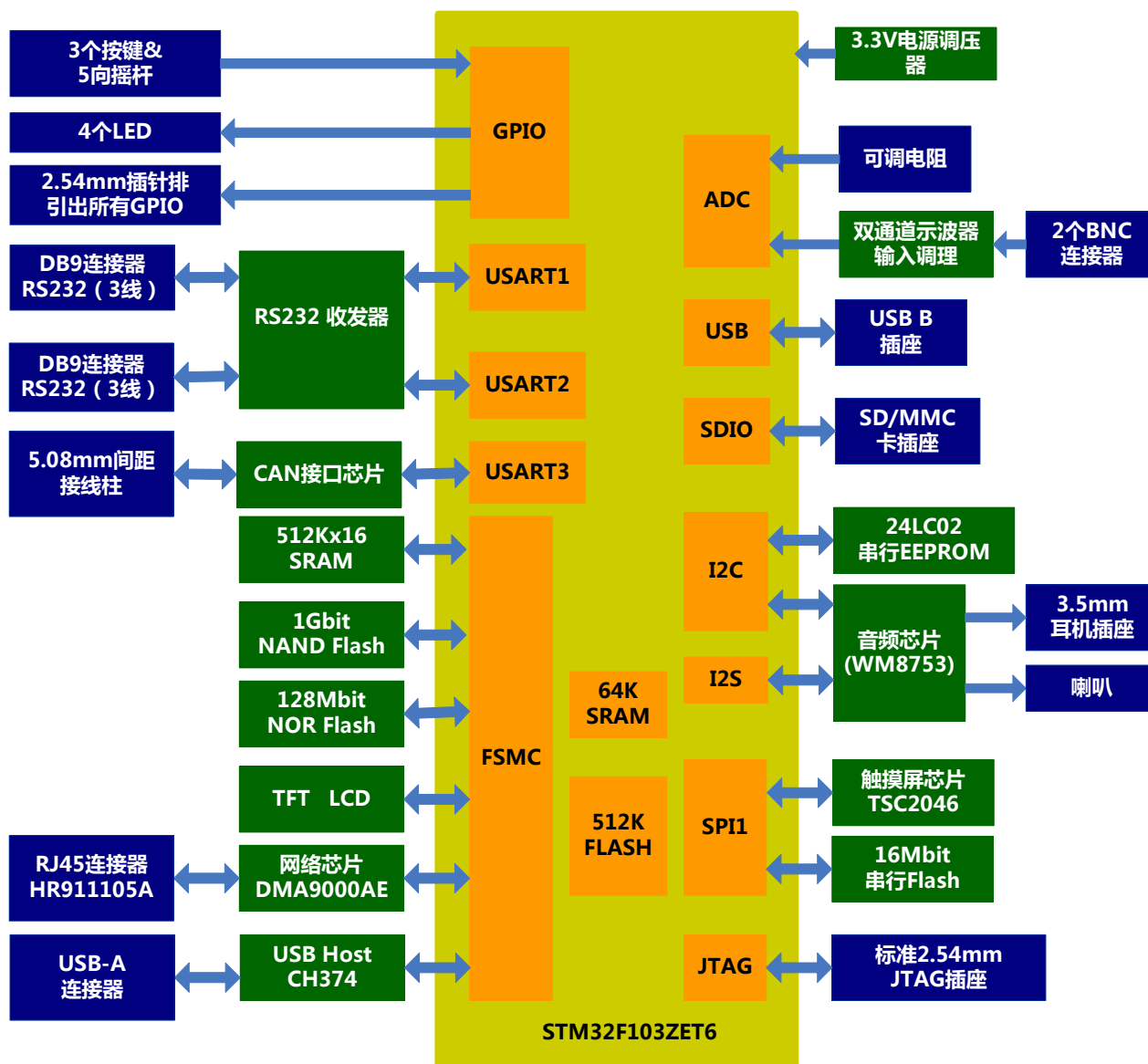
推荐的学习步骤：

- (1) 快速浏览CPU参考手册中的Memory and bus architecture , PWR , RCC , GPIO、 USART、 interrupts and events等章节。
- (2) 将本开发提供的基础例程都跑一遍，主要是熟悉ST软件库中常用函数的用法。顺序可以按 GPIO , USART , RTC , PWR , NVIC开始，然后熟悉ADC , CAN , SPI , USB等
- (3) 阅读并跟踪调试Demo程序，Demo程序几乎使用了板子的全部资源。包括LCD、网卡、USB Host、DfuSe固件升级。

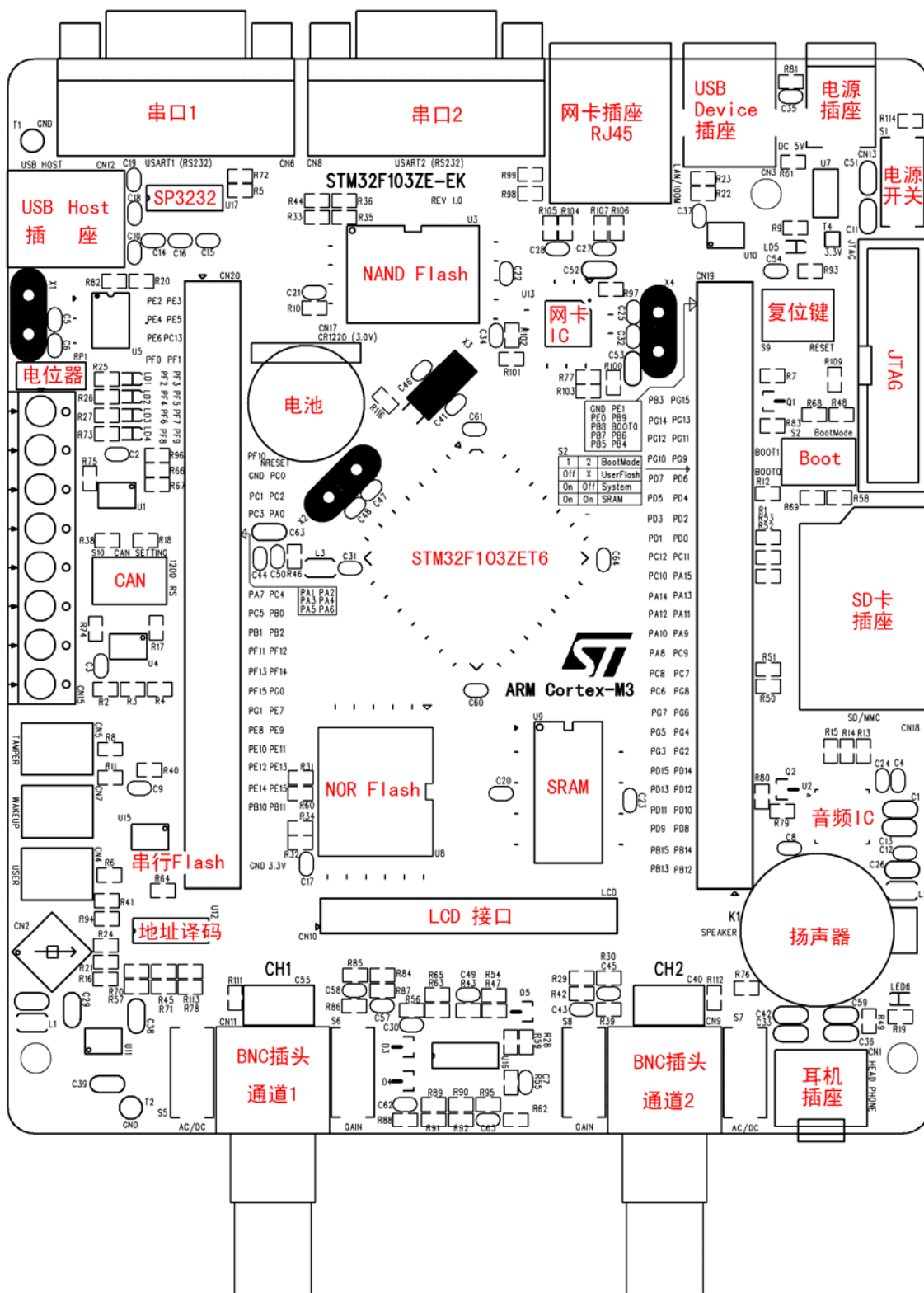
- (4) 阅读并跟踪调试uCOS + ucGUI的Demo程序。
- (5) 在学习的过程中，要逐步掌握开发工具的用法。主要是KEIL或IAR集成环境、J-Link仿真器。有一定基础后，可以自己设计或者在别人的基础上设计一个有使用价值的完整的程序，然后共享你的成果，和大家一起分享成功的喜悦。
- (6) 选择性地参与1至2个开源项目，进一步提高自己的编码能力。

3. 硬件介绍

3.1. 硬件模块框图



3.2. 丝印图



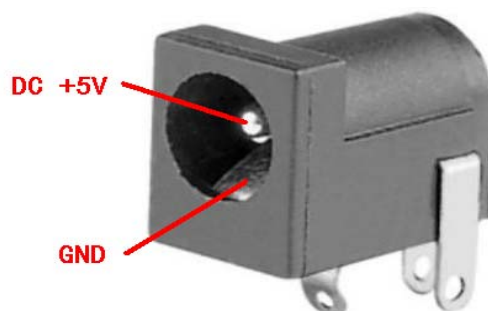
3.3. 电源供应

STM32F103ZE-EK开发板有三种供电方式，并且提供了电源开关S1进行切换。

- 1、通过外部5V直流电源供电（插座CN13）。
- 2、通过USB电缆供电（插座CN3）。
- 3、通过J-LINK调试器供电（插座CN16）。

对外接5V电源的要求：

- （1）主板的电源插座CN1为内正外负。
- （2）电源的电压不得高于5.5V。



J-LINK供电的方法：

- （1）将J-LINK的排线插入主板JTAG插座CN16，并将电源开关S1拨向板边。
- （2）启动windows开始菜单中的“J-Link Commander”，然后通过如下命令控制目标板上电、下电和复位。

命令	功能
power on	目标板上电。J-LINK掉电后再上电时，将不会对目标板供电
power on off	目标板下电
power on perm	目标板上电，并设置为缺省状态。以后每次连接上J-LINK时将自动给目标板上电
power off perm	目标板下电，并设置为缺省状态。
r0	使目标板处于复位状态
r1	使目标板退出复位状态，组合r0 r1 命令可实现目标板的复位

3.4. 启动模式

STM32F103ZE-EK开发板有以下三种启动方式：

- 从用户闪存存储器启动（正常运行时选择这种模式）
- 从系统存储器启动（做ISP下载时用）
- 从内嵌SRAM启动（调试用，一般很少使用）

通过设置子板上的拨码开关S2选择启动方式

位1状态	位2状态	启动模式
OFF	无关	用户闪存存储器启动
ON	OFF	从系统存储器启动
ON	ON	从内嵌SRAM启动



3.5. 时钟源

STM32F103ZE-EK开发板上有四种时钟源：

- 32KHz 晶振X3作为RTC的时钟源
- 8MHz晶振X2作为MCU的时钟源
- 25MHz晶振X4作为网卡芯片的时钟源
- 24MHz晶振X1作为USB Host芯片的时钟源

3.6. 复位

STM32F103ZE-EK开发板有两种复位方式：

- 通过开发板上的复位按键S9复位
- 通过JTAG调试口输入复位信号

3.7. 音频

STM32F103ZE-EK开发板集成了一个音频DAC芯片WM8753，具有立体声耳机驱动输出和扬声器驱动输出，支持耳机插入检测。

CPU和WM8753之间通过I2S和I2C接口连接，I2S通道用于传输音频数据流，I2C通道用于配置WM8753的工作状态（音量、外放和耳机切换、均衡控制等）。

CPU的DAC_OUT1输出连接到WM8753的线路输入通道LINE1，因此也可以通过CPU的DAC功能播放音频文件。CPU的DAC分辨率最大为12bit，WM8753支持的分辨率是24bit。

WM8753的空闲GPIO连接了1个LED指示灯，可以用于音频播放状态指示。

3.8. SRAM

STM32F103ZE-EK开发板提供1片512X16的SRAM（1M字节），连接到FSMC的BANK3，同时支持8bit和16bit访问模式。

芯片型号是IS61LV51216。

3.9. NAND Flash

STM32F103ZE-EK开发板提供1片128M字节的NAND Flash，连接到FSMC的NAND BANK2。NAND的READ/BUSY信号连接到FSMC_INT2引脚。

芯片型号是K9F1G08。

3.10. NOR Flash

STM32F103ZE-EK开发板提供1片16M字节的NOR Flash，连接到FSMC的NAND BANK1。最大支持32M字节。

芯片型号是AM29LV128ML。

3.11. 串行Flash

STM32F103ZE-EK开发板集成了一个串行Flash芯片SST25VF016B，容量为16Mbit（2M字节）。串行Flash连接到CPU的SPI1接口，其片选口线由CPU的PB2（也就是BOOT1）脚控制。

串行Flash的片选虽然和BOOT1（控制引导模式）复用，但是这个地方不存在冲突问题。因为在大多数情况下，BOOT1切换到高电平（从内部Flash启动或者系统存储器启动），只有在从内部RAM启动时，BOOT1才通过1个10K电阻接地。如果你的代码需要从内部RAM启动，那么CPU初始化外设时，应该立即将PB2口线设置为输出，并且设置为高电平状态。

3.12. 串行EEPROM

STM32F103ZE-EK开发板集成了一个串行EEPROM芯片24LC02，容量为256字节。串行EEPROM连接到CPU的I2C接口。

I2C接口上连接的设备还有音频芯片WM8753。两个器件的从地址不同，因此不存在冲突问题。

3.13. CAN

STM32F103ZE-EK开发板使用 SN65HVD230 (U10) 作为CAN 驱动器。CAN接口采用5.08mm间距的接线端子引出。

拨码开关S10用于设置CAN接口

位1状态	控制120欧负载
OFF	断开CAN总线并接的120欧负载电阻
ON	连接120欧负载电阻

位2状态	控制CAN芯片的RS口线状态
OFF	CAN芯片工作与高速模式
ON	CAN芯片工作与待机模式



如果需要将CANRX引脚用作它用，可以将R75电阻（0欧）去掉。

3.14. RS232 接口

STM32F103ZE-EK开发板提供2路RS232接口（DB9连接器），CPU的USART1连接到COM1端口，USART2连接到COM2端口。两个COM端口的信号均是标准的RS232电平。

DB9连接器脚位示意图如下：



COM1端口

DB9脚位	CPU引脚信号
2	USART1_TX (PA9)
3	USART1_RX (PA10)
5	GND
1、4、6、7、8、9	空脚

COM2端口

DB9脚位	CPU引脚信号
2	USART2_TX (PA2)
3	USART2_RX (PA3)
5	GND
1、4、6、7、8、9	空脚

STM32F103ZE-EK开发板和PC机连接使用直连的串口线（不是对传的串口线），市面出售的USB转串口线可以直接使用。

3.15. RS485 接口

STM32F103ZE-EK开发板提供1路RS485接口，RS485接口芯片为SP3485。485接口通过5.08mm间距的接线端子引出。

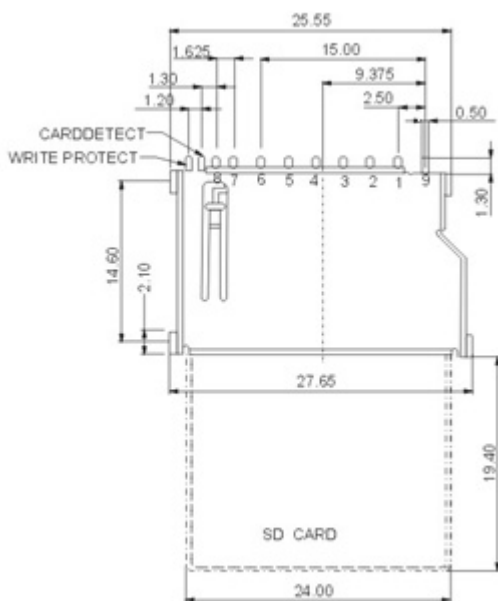
RS485芯片的数据端口连接到CPU的USART3接口，RS485芯片的接收使能和发送使能引脚通过两个独立的GPIO控制，因此可以做回环通信试验（自发自收模式）。

RS485总线上拉电阻、下拉电阻以及A-B间负载电阻由客户根据实际需要选贴。

如果用户需要将USRT3_RX用作它用，可以将R74电阻（0欧）去掉。

3.16. SD卡

STM32F103ZE-EK开发板具有SD卡接口，支持SD卡的读写。SD卡和STM32F103ZE-EK开发板接口的连接信号如下：



脚位	信号名称	对应的SMT32引脚
1	CD/DAT3	SD_DAT3 (PC11)
2	CMD	SD_CMD (PD2)
3	VSS1	GND
4	VDD	VDD3.3V
5	CLK	SD_CLK(PC12)
6	VSS2	GND
7	DAT0	SD_DAT0(PC8)
8	DAT1	SD_DAT1(PC9)
9	DAT2	SD_DAT2(PC10)
10	WP (写保护检测)	空
11	NCD (卡插入检测)	PC7

3.17. ADC输入

STM32F103ZE-EK开发板提供1个精密可调电阻，连接到引脚PC4/AIN12_14，可以做ADC采样试验。

3.18. 示波器电路

STM32F103ZE-EK开发板集成了1个双通道示波器前端调理电路，提供2个BNC接口，输入组阻抗为1M欧姆，支持负极信号输入，可直接使用标准的示波器探头。利用这个电路可以在开发板上进行USB虚拟示波器或者低端手持示波器项目。

通道1的模拟输出连接到CPU的AIN123_IN10、AIN123_IN12、AIN123_IN13，可以启用2个或者3

个ADC对信号进行分时采样，从而到达更高的采样速率。

通道2的模拟输出连接到CPU的AIN123_IN11，可以启用两个ADC对通道1和通道2进行同步采样。

每个通道配置有2个小型拨动开关，分别控制AC/DC切换和增益切换。

拨动开关外观：



开关标号	功能	值
S5	CH1通道AC/DC切换	拨向CPU方向选择DC，反之是AC
S6	CH1通道增益切换	拨向CPU方向选择高增益(-2V~+2V) 反之是低增益 (-10V~+10V)
S7	CH2通道AC/DC切换	拨向CPU方向选择DC，反之是AC
S8	CH2通道增益切换	拨向CPU方向选择高增益(-2V~+2V) 反之是低增益 (-10V~+10V)

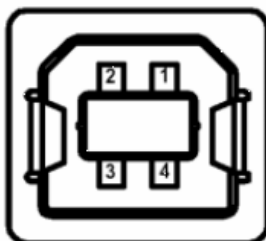
USB虚拟示波器性能指标：

序号	指标名称	值
1	模拟带宽	3MHz
2	CH1最高采样速率	3MHz
3	CH2最高采样速率	1MHz
4	采集深度	由内存分配决定
5	样本分辨率	12bit
6	AC/DC	双通道独立切换AC输入和DC输入
7	输入信号幅值范围	1档：-2V ~ +2V 2档：-10V ~ +10V 3档：-20V ~ +20V (探头X10) 4档：-100V ~ +100V (探头X10)

3.19. USB Device

STM32F103ZE-EK开发板提供了1个全速USB2.0 设备端口，通过标准的USB-B型连接器引出。通过该连接器，可以由PC机给目标供电（最大电流500mA限制）。

USB-D+信号线上的上拉电阻通过PB14引脚控制，高电平使能USB总线。



USB-B型连接器信号定义：

脚位	信号名称	对应的开发板信号
1	VBUS	USB_5V (5V电源)
2	D-	PA11/USBDM
3	D+	PA12/USBDP
4	GND	GND

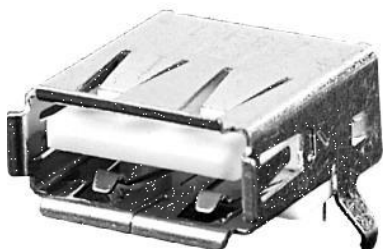
和 USB-B 插座配套的 USB 电缆又称为打印机电缆，电脑市场很多柜台有售。

3.20. USB Host

STM32F103ZE处理器内部并不带USB HOST功能，因此本开发板外扩了一个常用的USB专用芯片CH374T。

CH374T 是一个USB总线的通用接口芯片，支持USB-HOST主机方式，支持低速和全速的控制传输、批量传输、中断传输以及同步/等时传输。CH374 T具有8 位数据总线和读、写、片选控制线以及中断输出，可以方便地挂接到单片机/DSP/MCU/MPU等控制器的系统总线上。

STM32F103ZE-EK开发板提供一个USB-A型连接器，可以连接U盘、USB鼠标、USB键盘等设备。USB-A型连接器对外接的设备提供5V电源。



需要注意的是，如果这个5V电源是通过USB-DEVICE接口获得的，那么需要考虑一下外接设备的功耗问题。

3.21. 10M/100M网卡

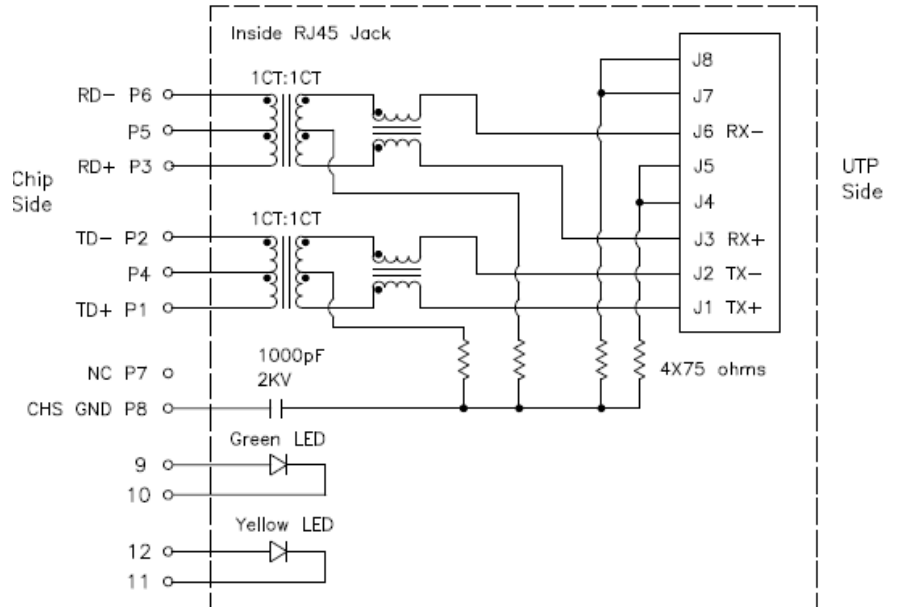
STM32F103ZE处理器内部没有集成 MAC电路，所以需要外接网络芯片。STM32F103ZE-EK开发板选择的网络芯片是DM9000A。

DM9000A是中国台湾DAVICOM公司推出的一款高速以太网接口芯片，其基本特征是：

- 48P LQFP封装
- 集成10/100M物理层接口
- 内部带有16K字节SRAM用作接收发送的FIFO缓存
- 遵循IEEE颁布的802.3以太网传输协议
- 支持8/16bit两种主机工作模式
- 支持唤醒帧、连接状态变化和magic包远程唤醒
- 通过HP认证的AUTO-Mdix(支持直接互连自动翻转)功能
- 支持TCP/IP加速（硬件实现IP/TCP/UDP包校验和的生成和校验）减轻CPU负担
- 支持从串行EEPROM自动装载制造商ID和产品ID(串行EEPROM是可选的)
- 支持唤醒帧、连接状态变化和magic包远程唤醒
- 具备接口速度和连接/激活状态指示灯控制信号
- 内建3.3V到2.5V的电压调节器
- 10ns I/O读写时间
- 具备待机和挂起睡眠模式



为了保证网络可靠运行，我们使用了集成网络变压器的 RJ45，HR911105A，带2个LED指示灯。



3.22. TFT显示屏和触摸屏接口

CPU没有内置触摸屏控制器，因此外扩一个触摸屏接口芯片ADS7843E（或TSC2046）。

ADS7843E和CPU间采用SPI接口通信，占用了CPU的SPI1接口，并且具有中断输出口线。当触摸屏被按压时PENIRQ口线输出低电平。

CPU没有内置LCD控制器，因此需要选用带控制器的LCD屏。STM32F103ZE-EK开发板标配的LCD屏为MP4、MP5中流行的3.0寸WQVGA（400x240）真彩屏，驱动芯片为SPFD5420A。该显示屏具有8bit和16bit两种总线模式，采用4个白色LED作为背光源，可以非常方便地和单片机接口。

触摸屏信号和LCD显示信号集通过一个2x15间距2.54mm的双排针引出。双排针的定义如下：

脚位	信号符号	中文含义	对应的CPU管脚
1	NCS	LCD控制器总线片选信号，低有效	FSMC_NCS
2	RS	LCD控制器寄存器地址线，用于选择命令寄存器或数据寄存器	FSMC_A0
3	NWE	LCD控制器总线写使能信号，低有效	FSMC_NWE
4	NOE	LCD控制器总线写使能信号	FSMC_NOE
5	NRESET	LCD控制器总线复位信号，低有效	NRESET
6	DB0	LCD控制器总线数据线（16bit）	FSMC_D0
~	~		~
21	DB15		FSMC_D15
22	TP_BUSY	触摸屏芯片输出的忙信号	PB5

23	TP_NCS	触摸屏芯片片选 (SPI)	PG11
24	TP_SCK	触摸屏芯片SPI接口时钟	SPI1_SCK
25	TP_MISO	触摸屏芯片SPI接口数据线 (触摸屏至CPU)	SPI1_MISO
26	TP_MOSI	触摸屏芯片SPI接口数据线 (CPU至触摸屏)	SPI1_MOSI
27	TP_NIRQ	触摸屏芯片输出的中断信号	PC5
28	BACK_LIGHT	LCD背光控制	PB1
29	GND	地	GND
30	VCC	电源	VDD3.3V

3.23. 按键和LED指示灯

STM32F103ZE-EK开发板总共具有6个LED指示灯，包括1个电源 (3.3V LDO) 指示灯、4个GPIO控制的指示灯、1个WM8753音频芯片控制的指示灯。

器件标号	用途说明	对应的CPU管脚
LD1	软件自由控制	PF6
LD2	软件自由控制	PF7
LD3	软件自由控制	PF8
LD4	软件自由控制	PF9
LD5	3.3V电源指示灯	VDD3.3V
LD6	WM8753音频芯片控制的指示灯,CPU通过I2C总线向WM8753发送命令,可以控制这个LED指示灯	WM8753的GP1

STM32F103ZE-EK开发板总共具有5个按键，1个USER按键、1个Tamper按键、1个WAKEUP按键、1个5向摇杆 (相当于5个独立的按键) 、1个复位按键。

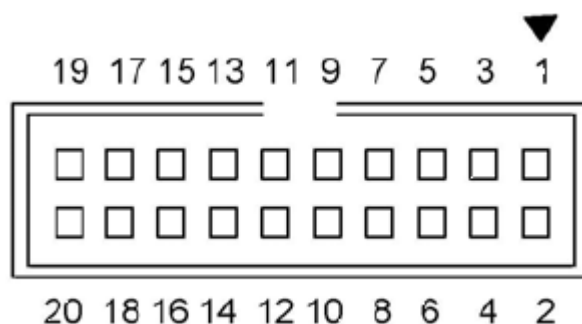
器件标号	用途说明	对应的CPU管脚
S9	复位按键	NRESET
CN4	用户自定义按键	PG8
CN5	TAMPER测试按键,也可以自定义功能	PC13/TAMPER
CN7	WAKEUP测试按键,也可以自定义功能	PA0/WKUP
CN2	5向摇杆,上键	PG15

CN2	5向摇杆，下键	PD3
CN2	5向摇杆，左键	PG14
CN2	5向摇杆，右键	PG13
CN2	5向摇杆，确认键	PG7

3.24. JTAG调试接口

STM32F103ZE-EK开发板提供了一个标准的2.54mm间距的20P插座，可以使用ARM仿真器烧写固件和调试程序。

JTAG接口定义如下：



器件标号	用途说明	对应的CPU管脚
S9	复位按键	NRESET
CN4	用户自定义按键	PG8
CN5	TAMPER测试按键，也可以自定义功能	PC13/TAMPER
CN7	WAKEUP测试按键，也可以自定义功能	PA0/WKUP
CN2	5向摇杆，上键	PG15
CN2	5向摇杆，下键	PD3
CN2	5向摇杆，左键	PG14
CN2	5向摇杆，右键	PG13
CN2	5向摇杆，确认键	PG7

3.25. 纽扣电池

主板上的电池座可以安装CR1220型的纽扣电池。该电池提供给CPU内部RTC使用。

当外部供电时，纽扣电池不对RTC电路供电，电流几乎为零。

长期不使用时，请卸下电池。

4. GPIO资源分配

I = 输入，O = 输出，X = 空闲

管脚	方向	用途	管脚	方向	用途
PA0	I	WKUP	PB0	X	空闲，引出到接线端子
PA1	I	DM9000AE中断	PB1	O	LCD背光
PA2	O	串口2，USART2_TX	PB2	O	BOOT1&串行Flash片选
PA3	I	串口2，USART2_RX	PB3	O	JTAG, TDO
PA4	O	DAC_OUT1	PB4	I	JTAG, TRST
PA5	O	串行Flash，SPI1_SCK	PB5	I	触摸屏，TP_BUSY
PA6	I	串行Flash，SPI1_MISO	PB6	I	串行EEPROM，I2C1_SCL
PA7	O	串行Flash，SPI1_MOSI	PB7	I/O	串行EEPROM，I2C1_SDA
PA8	X	空闲，引出到接线端子	PB8	I	CANRX
PA9	O	串口1，USART1_TX	PB9	O	CANTX
PA10	I	串口1，USART1_RX	PB10	O	RS485，USART3_TX
PA11	-	USBDM (D-)	PB11	I	RS485，USART3_TX
PA12	-	USBDP (D+)	PB12	O	WM8753，I2S2_WS
PA13	I	JTAG, TMS	PB13	O	WM8753，I2S2_CK
PA14	I	JTAG, TCK	PB14	O	USB上拉使能
PA15	I	JTAG, TDI	PB15	O	WM8753，I2S2_SD
PC0	I	WKUP	PD0	I/O	FSMC_D2
PC1	I	DSO，ADC123_IN10	PD1	I/O	FSMC_D3
PC2	I	DSO，ADC123_IN11	PD2	O	SD卡，SDIO_CMD
PC3	I	DSO，ADC123_IN12	PD3	I	摇杆，下
PC4	I	DSO，ADC123_IN13	PD4	O	FSMC_NOE
PC5	I	触摸屏中断	PD5	O	FSMC_NWE
PC6	O	WM8753，I2S2_MCK	PD6	I	FSMC_NWAIT
PC7	O	SD卡插入检测	PD7	O	FSMC_NCE2
PC8	I/O	SD卡，SDIO_D0	PD8	I/O	FSMC_D13

PC9	I/O	SD卡, SDIO_D1		PD9	I/O	FSMC_D14
PC10	I/O	SD卡, SDIO_D2		PD10	I/O	FSMC_D15
PC11	I/O	SD卡, SDIO_D3		PD11	O	FSMC_A16
PC12	O	SD卡, SDIO_CK		PD12	O	FSMC_A17
PC13	I	按键, TAMPER		PD13	O	FSMC_A18
PC14	I	32K晶振, OS32_IN		PD14	I/O	FSMC_D0
PC15	O	32K晶振, OS32_OUT		PD15	I/O	FSMC_D1
PE0	O	FSMC_NBL0		PF0	O	FSMC_A0
PE1	I	FSMC_NBL1		PF1	O	FSMC_A1
PE2	I	CH374中断信号		PF2	O	FSMC_A2
PE3	O	FSMC_A19		PF3	O	FSMC_A3
PE4	O	FSMC_A20		PF4	O	FSMC_A4
PE5	O	FSMC_A21		PF5	O	FSMC_A5
PE6	O	FSMC_A22		PF6	O	LED1
PE7	I/O	FSMC_D4		PF7	O	LED2
PE8	I/O	FSMC_D5		PF8	O	LED3
PE9	I/O	FSMC_D6		PF9	O	LED4
PE10	I/O	FSMC_D7		PF10	O	RS485, 发送使能 (1有效)
PE11	I/O	FSMC_D8		PF11	O	RS485, 接收使能 (0有效)
PE12	I/O	FSMC_D9		PF12	O	FSMC_A6
PE13	I/O	FSMC_D10		PF13	O	FSMC_A7
PE14	I/O	FSMC_D11		PF14	O	FSMC_A8
PE15	I/O	FSMC_D12		PF15	O	FSMC_A9
PG0	O	FSMC_A10				
PG1	O	FSMC_A11				
PG2	O	FSMC_A12				
PG3	O	FSMC_A13				
PG4	O	FSMC_A14				
PG5	O	FSMC_A15				
PG6	O	FSMC_INT2				
PG7	I	摇杆, 中间键				
PG8	I	按键, USER				

PG9	O	FSMC_NE2				
PG10	O	FSMC_NE3				
PG11	O	触摸屏SPI片选				
PG12	I/O	FSMC_NE4				
PG13	I/O	摇杆，右				
PG14	I/O	摇杆，左				
PG15	I/O	摇杆，上				

除了FSMC类的GPIO外，其它的输出型（方向=' O' ）的GPIO均可直接外接其它外设，可作输出口线也可做输入口线用，用户不必担心存在板载器件和外扩设备的冲突问题。

如果用户需要将输入型（方向=' I' ）的GPIO用于外设，那么需要将该GPIO上串联的0欧姆电阻去掉，否则可能会存在冲突问题。

5.FSMC资源分配

器件标号，芯片	地址范围	占用的FSMC管脚
U8，NOR Flash	0x6400 0000 ~ 0x64FF FFFF	A0-A22,D0-D15,NE2
U9，SRAM	0x6800 0000 ~ 0x680F FFFF	A0-A18,D0-D15,NE3
CN10，LCD	0x6C00 0000，0x6C00 0002	A0,D0-D15,NE4(外扩地址译码器)
U13，DM9000AE	0x6C40 0000，0x6C40 0008	A2,D0-D15,NE4(外扩地址译码器)
U5，CH374	0x6C80 0000，0x6C80 0008	A2,D0-D7,NE4(外扩地址译码器)

STM32的CPU同时只能使用4个片选，也就是只能外接4个总线型设备。NAND Flash已经占用一个，剩余的3个片选口线无法同时连接5个设备。

为了解决上述问题，本开发板外扩一个地址译码器电路，将NE3和NE4空间各划分出4等份。也就是将2根片选扩充为8根，本开发板占用其中的4根，剩余的4个片选也引到GPIO排针，用户还可以外接4个总线设备。

6. 基础例程（初级）

6.1. 基础例程介绍

基础例程是针对CPU某一个硬件功能的测试程序，这些测试程序功能单一，源代码代码比较简单，主要目的是帮助用户了解CPU的接口以及ST标准固件库的用法。

本章节介绍的基础例程全部来自于ST公司官方网站发布的源码，我们对每个例程添加了工程文件，并根据板子硬件的特点做了少量修改。

初学者通过运行基础例程，可以快速掌握开发环境和调试工具的基本用法，以及了解STM32系列CPU的基本特征。

STM32F103ZE-EK开发板配套的试验例程均**基于ST公司最新发布的V3.1版本的固件库**。我们精心收集并整理了每个例程的源码，**确保每个例程可在STM32F103ZE-EK开发板上调试并运行**。

所有的基础例程均提供单独的KEIL和IAR的工程文件。每个KEIL的工程中包含4套目标代码设置：

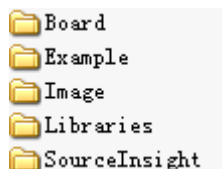
- 在CPU内部Flash运行，Target名字：flash
- 在CPU内部RAM运行，Target名字：ram（由于flash寿命有限，推荐大家在ram中调试）
- 在外部SRAM运行，Target名字：extram
- 在外部NOR运行，Target名字：nor（注：可以生成目标文件，但是通过JLINK在线烧写NOR Flash存在异常，估计是J-Link驱动的问题，待查）

我们采用源代码方式将固件源码嵌入工程，没有采用库文件形式。这样做的目的是方便开发者步单步跟踪调试库中的每个函数。

每个工程的编译优化选项设置为最低，目的也是为了便于单步调试。

我们修改过的源代码均进行了注释，注释中均包含有“armfly：”标识。

6.2. 源码目录结构



Board：存放跟开发板外围硬件相关的源代码文件。

Example：存放每个例程的源代码和工程文件。

Image：集中存放每个例程生成的目标代码（Hex格式），便于用户直接烧写到Flash进行测试。

Libraries : 存放 ST公司提供的固件库源代码。

SourceInsight : 存放Source insight的项目文件，强烈推荐大家使用该工具阅读代码。这个项目文件包含了所有的例程源码，包括st的lib源码。

6.3. 基础例程列表（86 个例程）

ADC_3ADCs_DMA	I2C_EEPROM	TIM_OCActive
ADC_ADC1_DMA	I2C_Interrupt	TIM_OCInactive
ADC_AnalogWatchdog	I2C_SMBus	TIM_OCToggle
ADC_ExtLinesTrigger	I2S_Interrupt	TIM_OnePulse
ADC_RegSimul_DualMode	I2S_SPI_I2S_Switch	TIM_Parallel_Synchro
ADC_TIMTrigger_AutoInjection	IWDG	TIM_TimeBase
Backup_Data	Lib_DEBUG	USART_DMA_Interrupt
Backup_Tamper	NVIC_DMA_WFIMode	USART_DMA_Polling
CAN_LoopBack	NVIC_IRQ_Channels	USART_HalfDuplex
CAN_Normal	NVIC_Priority	USART_HyperTerminal_HwFlowControl
CortexM3_BitBand	NVIC_VectorTable_Relocation	USART_HyperTerminal_Interrupt
CortexM3_Mode_Privilege	PWM_Input	USART_Interrupt
CRC	PWM_Output	USART_IrDA
DAC_DualModeDMA_SineWave	PWR_STANDBY	USART_MultiProcessor
DAC_OneChannel_NoiseWave	PWR_STOP	USART_Polling
DAC_OneChannelDMA_Escalator	RCC	USART_Printf
DAC_TwoChannels_TriangleWave	RTC_Calendar	USART_Smartcard
DMA_ADC_TIM1	RTC_LSI_Calib	USART_Synchronous
DMA_FLASH_RAM	SDIO	WWDG
DMA_FSMC	SPI_CRC	
DMA_I2C_RAM	SPI_DMA	
DMA_SPI_RAM	SPI_FullDuplex_SoftNSS	
Ex1_Touch	SPI_M25P64_FLASH	
EXTI	SPI_Simplex_Interrupt	
Flash_Program	SysTick	
Flash_Write_Protection	Test_NOR_Flash	
FSMC_NAND	TIM1_Synchro	
FSMC_NOR	TIM_6Steps	
FSMC_NOR_CodeExecute	TIM_7PWM_Output	
FSMC_SRAM	TIM_Cascade_Synchro	
GPIO_IOToggle	TIM_ComplementarySignals	
GPIO_JTAG_Remap	TIM_DMA	
I2C_10bitAddress	TIM_ExtTrigger_Synchro	
I2C_DualAddress	TIM_InputCapture	

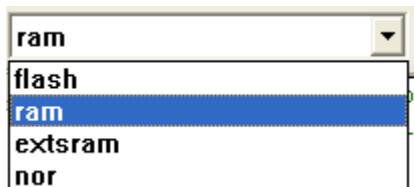
基础例程较多，初学者根据自己的需要选择性的了解即可。

6.4. 基础例程调试方法（KEIL）

（1） 安装好KEIL MDK集成环境，以及segger仿真器驱动。

- (2) 接好J-Link仿真器，给开发板上电。
- (3) 进入 Example\GPIO_IOToggle\RVMDK目录，双击Project.Uv2，启动KEIL开发环境。

点击下来列表框，可以选择目标类型，推荐用ram。



点击按钮  编译和连接整个工程的源代码，以后可以只用点击按钮  编译修改过的源文件。
编译和连接成功后，就可以点击按钮  进入Debug状态。

进入Debug界面后，可以进行全速、单步、跟踪、暂停等操作。

6.5. 基础例程调试方法（IAR）

目前，IAR的工程都是在Flash中运行中。

1. 安装好IAR集成环境，以及segger仿真器驱动。
2. 接好J-Link仿真器，给开发板上电。
3. 进入 \Example\GPIO_IOToggle\EWARMv5目录，双击Project.eww，启动IAR EWARM开发环境。

点击按钮  编译和连接整个工程的源代码。
编译和连接成功后，就可以点击按钮  开始下载程序并且进入Debug状态。

进入Debug界面后，可以进行全速、单步、跟踪、暂停等操作。

4.应用例程（中级）

本文所指的应用例程相对与基础例程而言，代码规模明显要大些，涉及面也比较广。

该章节的应用例程主要来源于ST官方网站，也有部分例程来源与网络。我们根据自己的板子硬件特性进行了必要的修改，以确保这些例程可以在我们板子上正常运行。

本开发板配套的应用例程有：

- (1) I2S音频播放器
- (2) 用游戏杆模拟USB 鼠标
- (3) 实现USB 虚拟串口
- (4) 使用DfuSe进行固件升级
- (5) USB Mass Storege项目，将开发板作为U盘，存储介质是SD卡。
- (6) CH374的U盘读写。
- (7) CH374连接鼠标、键盘。
- (8) DM9000AE+uIP实现web服务器。
- (9) uCOS + ucGUI的移植。

5.大型项目（高级）

计划开展的项目有：

- (1) ucLinux的移植
- (2) USB虚拟示波器（连续采集模拟信号，然后通过USB或网卡传输模拟信号至PC机，由PC机展示和分析波形）
- (3) USB手持示波表（采集波形，本地TFT显示器直接显示）
- (4) MP3播放器

6.烧写本开发板的演示Demo程序

第1步：烧写Dfu_BootLoader程序到CPU内部Flash。起始地址：0x00000

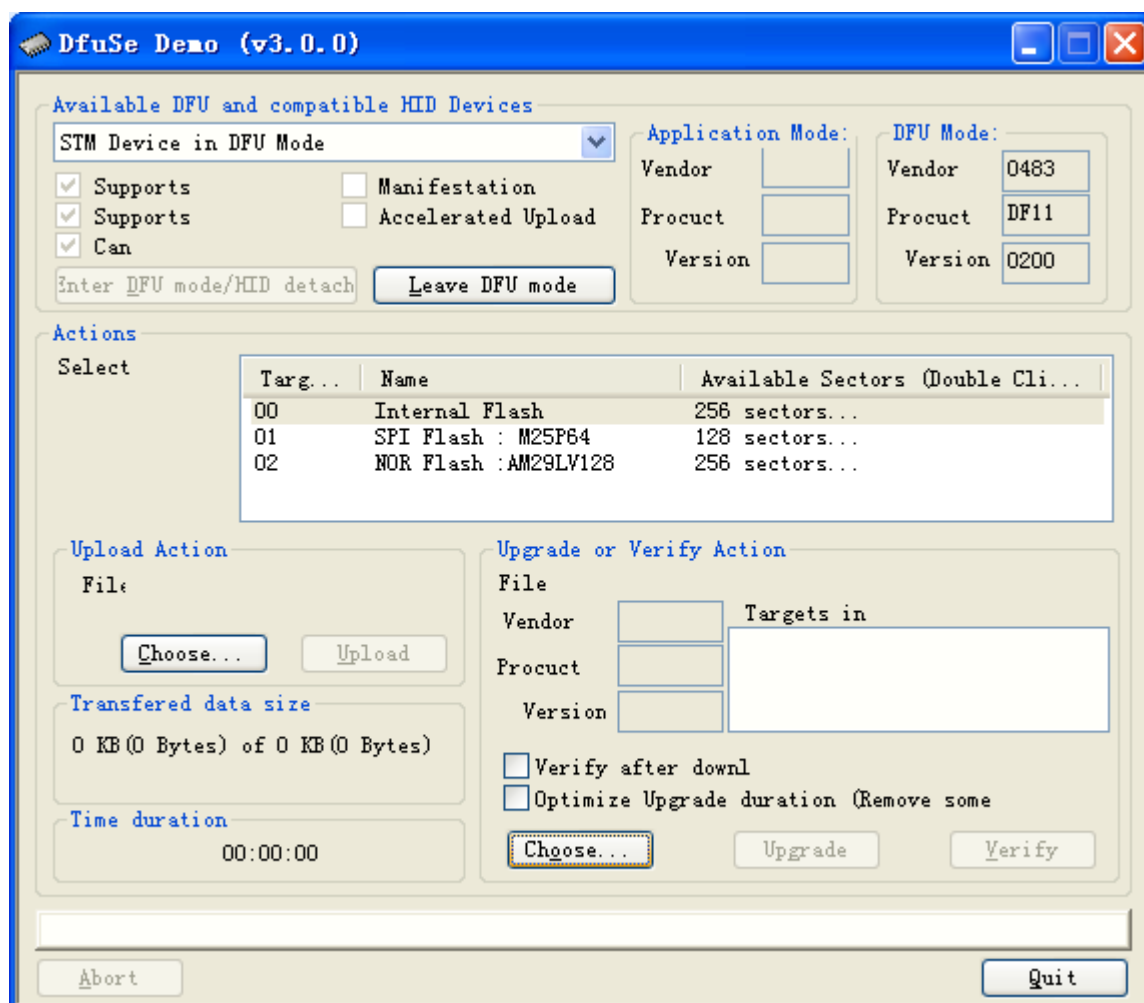
使用J-Link仿真器或者串口ISP烧写程序完成本操作。

第2步：烧写ucGuiDemo.hex程序到CPU内部Flash地址。起始地址：0x03000

使用J-Link仿真器或者串口ISP烧写程序完成本操作。

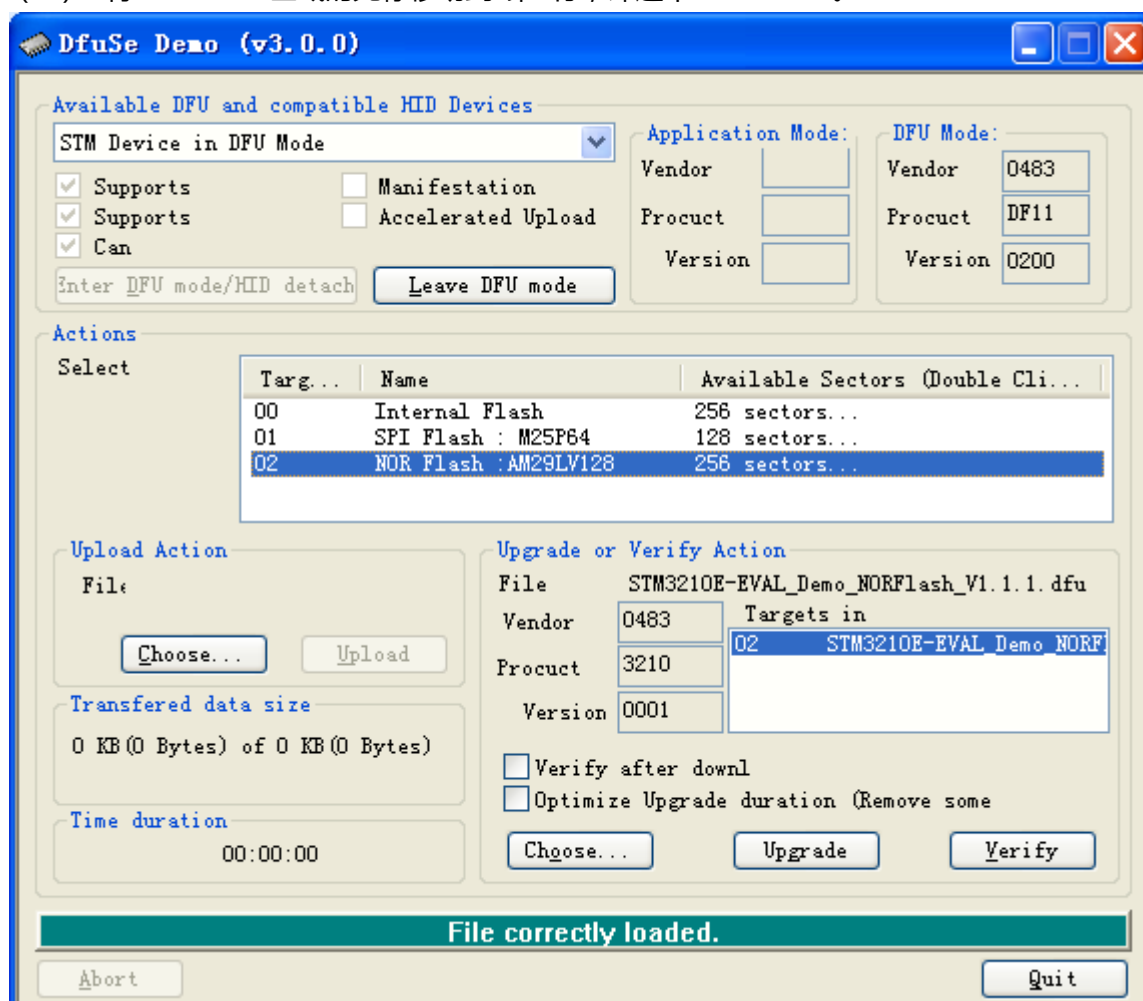
第3步：烧写STM3210E-EVAL_Demo_NORFlash_V1.1.1.dfu到NOR Flash。起始地址：0x64006000

- (1) 安装um0412中的 DfuSe_Demo_V3.0_Setup.exe
- (2) 执行windows开始菜单 “STMicroelectronics” - “DfuSe” - “DfuSe Demonstration”
- (3) 开发板上电，按住开发板上的“ USER” 键不放，再按复位键，此时Windows应该发现USB设备插入，USB驱动安装成功后，即可在DfuSe Demonstration界面看到开发板上的资源。



(4) 点击按钮“Choose”，选择“STM3210E-EVAL_Demo_NORFlash_V1.1.1.dfu”文件

(5) 将“Seletc”区域的光标移动到第3行,即选中NOR Flash。



(6) 点击按钮“Upgrade”,将开始下载文件到NOR Flash。此过程耗时很长,请耐心等待。如果不加校验,耗时13分钟,加上校验则耗时23分钟。

注:我做测试,单独写个程序完成NOR Flash整个16M空间擦写和重写只需要3分钟。看来大部分时间耗费在USB通信上了。这个Dfu Boot程序和PC机程序的通信机制还有待改进。我分析慢的原因:

- a) USB通信时每包数据长度过小,只有512字节,造成通信量很大,需要32768包数据。
- b) 一问一答式的通信协议也是效率低下的一个原因。

7.使用J-LINK仿真器烧写CPU内部Flash

(1) 启动J-Link ARM软件。

- (2) 点击主菜单 “Options” – “Project Settings”
- (3) 选择 “CPU” 页签，然后在 “Device” 下拉列表框中选择 “ STM32F103ZE ”
- (4) 点击 “确定” 后完成设置。你也可以再点击 “File ” - “Save Project As” 保存 *.prj文件，以便于下次进行重复设置。
- (5) 然后点击主菜单 “File ” – “ Open ”装入待烧写的文件。该软件会根据后缀自动识别文件格式。（支持hex，bin，s19等）。
- (6) 点击主菜单 “Target ”-“ Connect ”，测试一下目标板和J-Link之间的连接是否正常。
- (7) 点击主菜单 “Target ” - “Auto” ，将开始烧写Flash，之后会进行校验。也可以直接按 “F7” 键开始自动烧写。

8. 硬件缺陷说明

硬件版本识别：

产品型号：STM32F103ZE-EK

硬件版本：REV 1.0 （丝印在串口附近）

缺陷 1：示波器 BNC 插头的芯线和地线反了。

解决：BNC插头的两根引线交叉后再焊接，出厂的板子均已经做处理。

缺陷 2：示波器电路中的有 2 个运放单元的输入相位反了。

解决：将运算放大器芯片相关引脚进行交叉处理后再焊接，出厂板子均已经做处理。（呵呵，修补手术对外观的影响很小，不仔细看，很难看出来）

说明：光盘中提供的原理图已经对这 2 个缺陷进行了处理。



9.更新记录

版本	更改说明	作者	发布日期
1.0	初版	armfly	2009-10-01

目录

1. 产品规格	2
2. 快速入门	3
2.1. 注意事项	3
2.2. 检查硬件功能	4
2.3. 跳线配置	6
2.4. 学习建议	7
3. 硬件介绍	9
3.1. 硬件模块框图	9
3.2. 丝印图	10
3.3. 电源供应	11
3.4. 启动模式	12
3.5. 时钟源	12
3.6. 复位	12
3.7. 音频	12
3.8. SRAM	13
3.9. NAND Flash	13
3.10. NOR Flash	13
3.11. 串行 Flash	13
3.12. 串行 EEPROM	14
3.13. CAN	14



3.14. RS232 接口.....	14
3.15. RS485 接口.....	15
3.16. SD 卡	16
3.17. ADC 输入	16
3.18. 示波器电路.....	16
3.19. USB Device	18
3.20. USB Host.....	18
3.21. 10M/100M 网卡	19
3.22. TFT 显示屏和触摸屏接口	20
3.23. 按键和 LED 指示灯	21
3.24. JTAG 调试接口	22
3.25. 纽扣电池.....	22
4. GPIO 资源分配	23
5. FSMC 资源分配	25
6. 基础例程（初级）	26
6.1. 基础例程介绍.....	26
6.2. 源码目录结构.....	26
6.3. 基础例程列表（86 个例程）	27
6.4. 基础例程调试方法（KEIL）	27
6.5. 基础例程调试方法（IAR）	28
4. 应用例程（中级）	28
5. 大型项目（高级）	29



6. 烧写本开发板的演示 Demo 程序.....	29
7. 使用 J-LINK 仿真器烧写 CPU 内部 Flash	31
8. 硬件缺陷说明	33
9. 更新记录	34