

零死角玩转STM32



串行FLASH文件系统 FatFs

淘宝：firestm32.taobao.com

论坛：www.firebbs.cn



扫描进入淘宝店铺

01

文件系统简介

02

FatFs文件系统简介

03

FatFs文件系统移植实验

04

FatFs功能使用实验

参考资料:《零死角玩转STM32》

“SPI—串行FLASH文件系统FatFs” 章节

串行FLASH文件系统FatFs



C语言中的文件操作

文件的打开操作 **fopen** 打开一个文件

文件的关闭操作 **fclose** 关闭一个文件

文件的读写操作 **fgetc** 从文件中读取一个字符

fputc 写一个字符到文件中

fgets 从文件中读取一个字符串

fputs 写一个字符串到文件中

fprintf 往文件中写格式化数据

fscanf 格式化读取文件中数据

fread 以二进制形式读取文件中的数据

fwrite 以二进制形式写数据到文件中

getw 以二进制形式读取一个整数

putw 以二进制形式存贮一个整数

文件状态检查函数 **feof** 文件结束

ferror 文件读/写出错

clearerr 清除文件错误标志

ftell 了解文件指针的当前位置

文件定位函数 **rewind** 反绕

fseek 随机定位

串行FLASH文件系统FatFs



FATFS文件系统简介

FatFs是面向小型嵌入式系统的一种通用的FAT文件系统。它完全是由AISI C语言编写并且完全独立于底层的I/O介质。因此它可以很容易地不加修改地移植到其他的处理器当中，如8051、PIC、AVR、SH、Z80、H8、ARM等。FatFs支持FAT12、FAT16、FAT32等格式。

利用前面写好的SPI Flash芯片驱动，把FatFs文件系统代码移植到工程之中，就可以利用文件系统的各种函数，对SPI Flash芯片以“文件”格式进行读写操作了。

FatFs文件系统的源码可以从fatfs官网下载：
http://elm-chan.org/fsw/ff/00index_e.html

FATFS与底层的IO口是独立的

FATFS不能直接操作IO进行读写，而是

FATFS先连接你写好的驱动函数，再通过驱动函数对IO口从而对FLASH操作

流程如下：

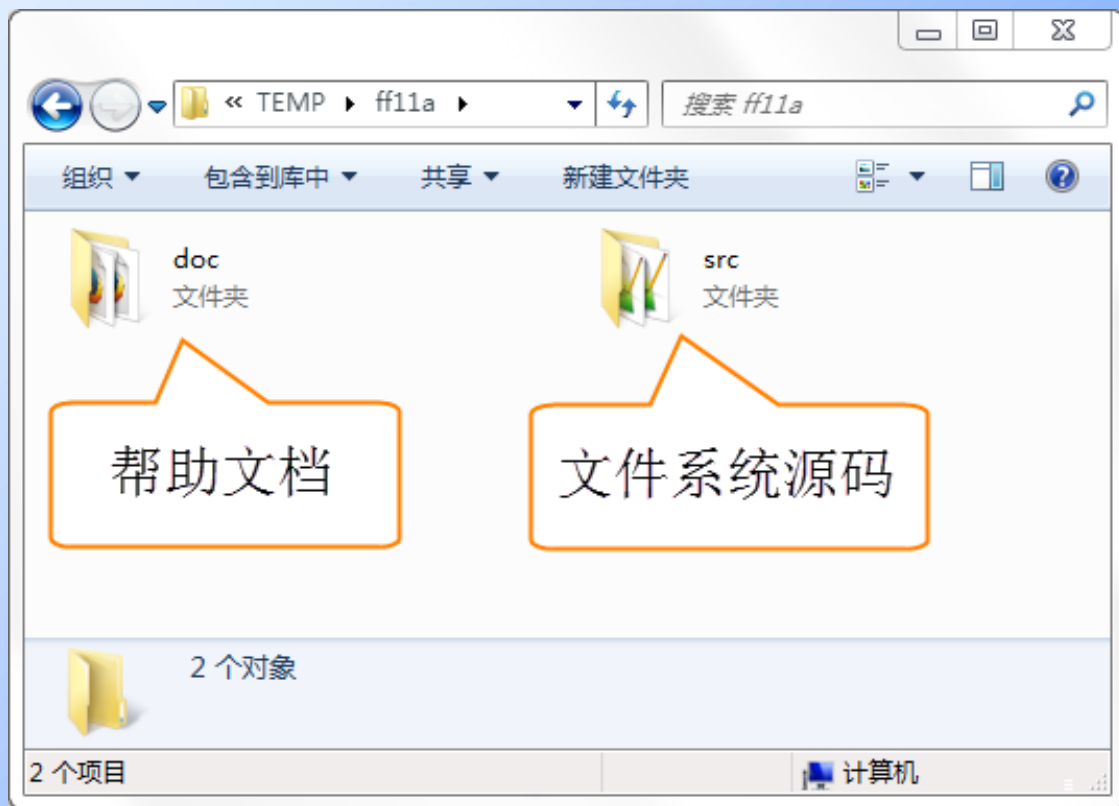
FATFS+驱动函数(flash/eeprom/sd等的驱动函数单字节写/读等)+外接设备(flash/eeprom/sd)

串行FLASH文件系统FatFs



FatFs的目录结构

下载FatFs源码包解压后，在doc 文件夹里面是一些使用帮助文档；
在src 是FatFs文件系统的源代码。

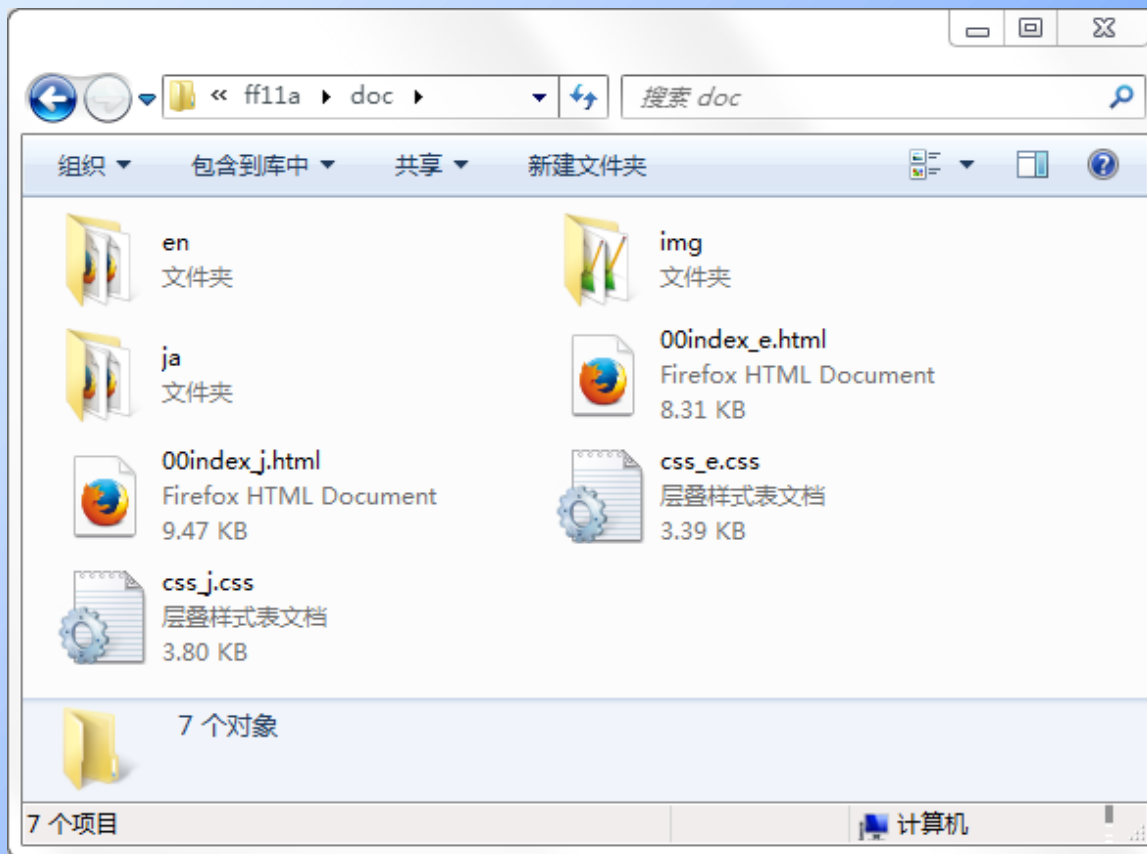


串行FLASH文件系统FatFs



FatFs帮助文档

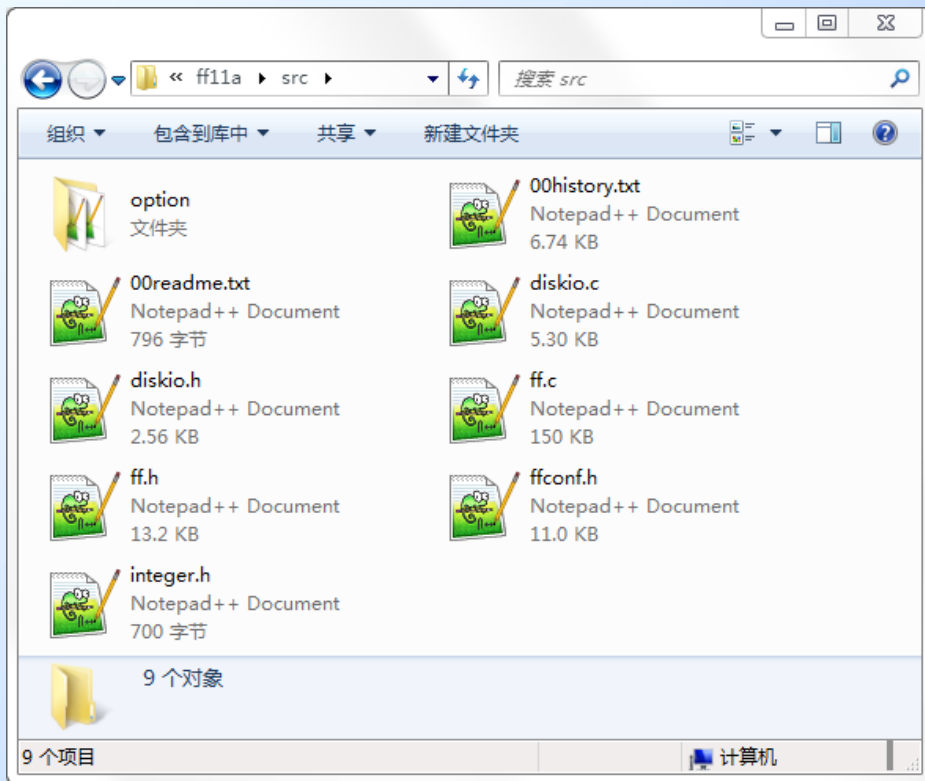
其中 en 和 ja 这两个文件夹里面是编译好的html文档，讲的是FATFS里面各个函数的使用方法。00index_e.html和00index_j.html是一些关于FATFS的简介。



串行FLASH文件系统FatFs



FATFS源码

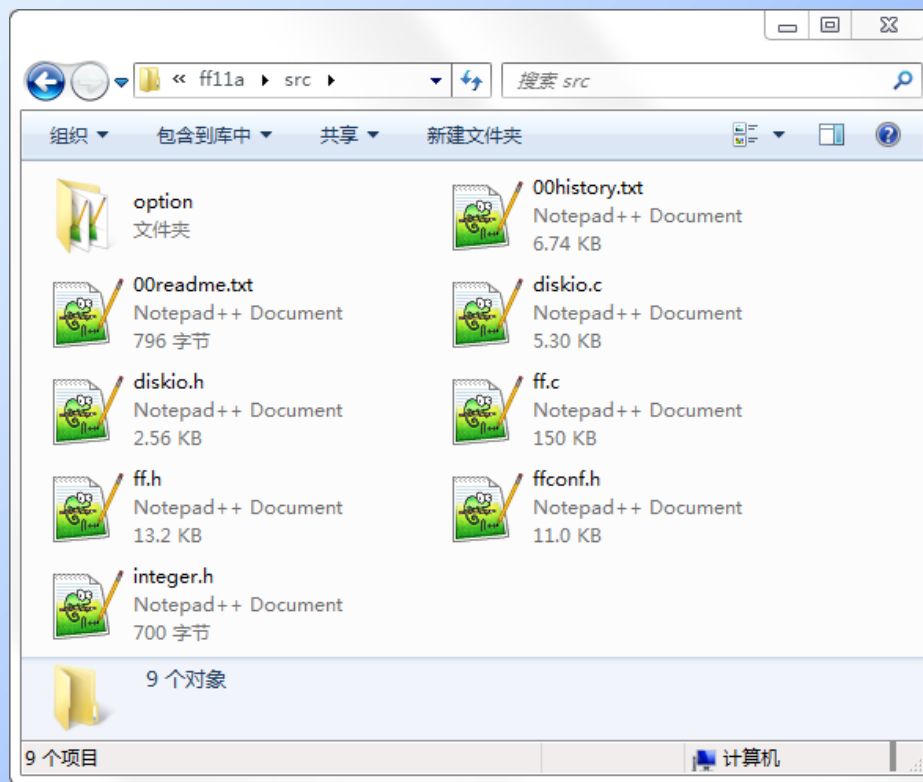


- **integer.h**: 文件中包含了一些数值类型定义。
- **diskio.c**: 包含底层存储介质的操作函数，这些函数需要用户自己实现，主要添加底层驱动函数。
- **ff.c**: **FatFs**核心文件，文件管理的实现方法。该文件独立于底层介质操作文件的函数，利用这些函数实现文件的读写。
- **cc936.c**: 本文件在**option**目录下，是简体中文支持所需要添加的文件，包含了简体中文的**GBK**和**Unicode**相互转换功能函数。
- **ffconf.h**: 这个头文件包含了对**FatFs**功能配置的宏定义，通过修改这些宏定义就可以裁剪**FatFs**的功能。如需要支持简体中文，需要把**ffconf.h**中的 **_CODE_PAGE** 的宏改成**936**并把上面的**cc936.c**文件加入到工程之中。

串行FLASH文件系统FatFs



FATFS源码



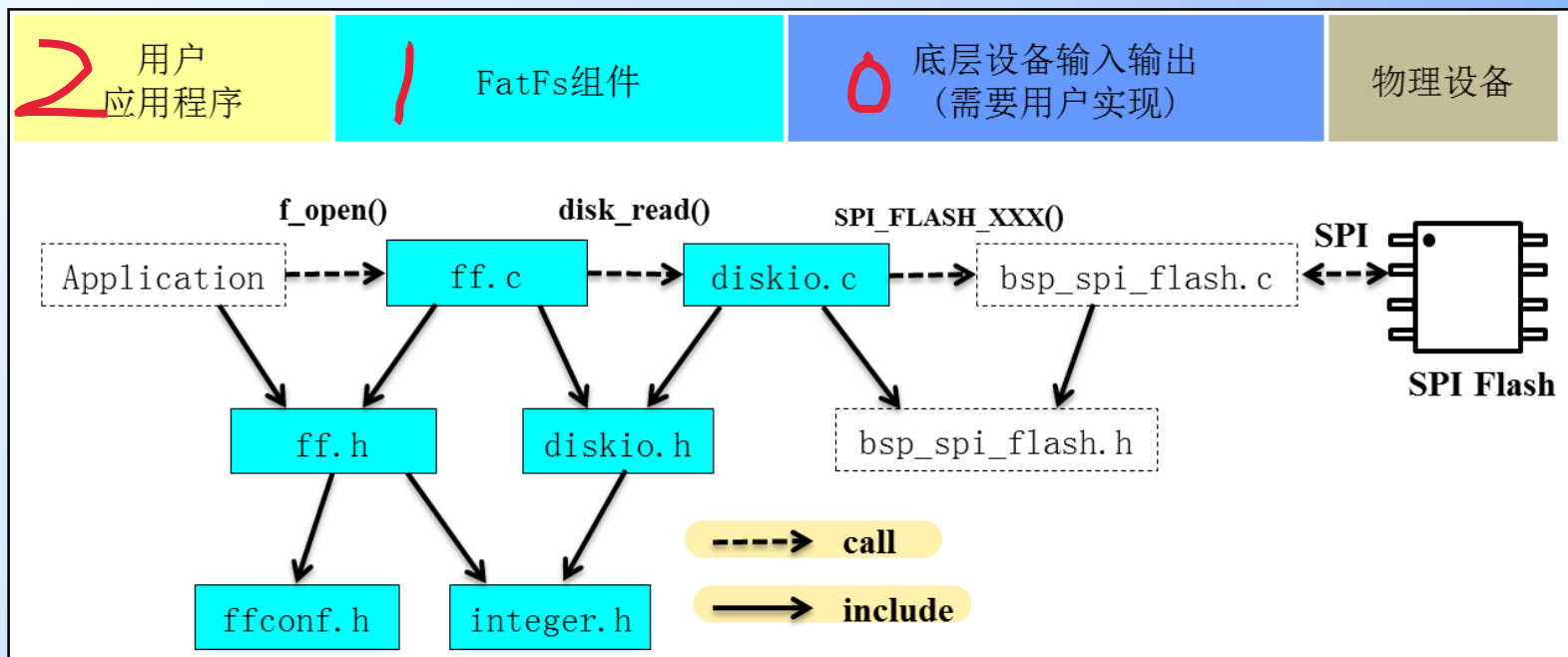
建议阅读这些源码的顺序为：`integer.h` --> `diskio.c` --> `ff.c`。

阅读文件系统源码`ff.c`文件需要一定的功底，建议先阅读FAT32的文件格式，再去分析`ff.c`文件。若仅为了使用文件系统，则只需要理解`integer.h`及`diskio.c`文件并会调用`ff.c`文件中的函数就可以了。

串行FLASH文件系统FatFs



FatFs在程序中的关系网络



- 用户应用程序需要由用户编写，想实现什么功能就编写什么的程序，一般我们只用到 `f_mount()`、`f_open()`、`f_write()`、`f_read()` 就可以实现文件的读写操作。这些应用层函数使用方法与标准C的文件操作函数类似。
- FatFs组件是FatFs的主体，文件都在源码src文件夹中，其中 `ff.c`、`ff.h`、`integer.h` 以及 `diskio.h` 四个文件我们不需要改动，只需要修改 `ffconf.h` 和 `diskio.c` 两个文件。
- 底层设备输入输出要求实现存储设备的读写操作函数、存储设备信息获取函数等等。

串行FLASH文件系统FatFs



FatFs移植需要用户支持函数

函数	条件(ffconf.h)	备注
disk_status disk_initialize disk_read	总是需要	通过都FFCONF.H函数的宏定义来配置ff.c diskio.c文件要哪个不要哪个函数
disk_write get_fattime 得到flash修改时间 disk_ioctl (CTRL_SYNC) 写缓存命令, flash没有RAM因此用不到	_FS_READONLY == 0	底层设备驱动函数
disk_ioctl (GET_SECTOR_COUNT) 得到flash的扇区数 disk_ioctl (GET_BLOCK_SIZE) 得到每次擦除区块block的大小	USE_MKFS == 1	
disk_ioctl (GET_SECTOR_SIZE) 得到扇区的大小	_MAX_SS != _MIN_SS MIN_SS是标准值512byte 而flash扇区是4096byte	
disk_ioctl (CTRL_TRIM)	_USE_TRIM == 1	
ff_convert ff_wtoupper	_USE_LFN != 0	Unicode支持, 为支持简体中文, 添加cc936.c到工程即可
ff_cre_syncobj ff_del_syncobj ff_req_grant ff_rel_grant	_FS_REENTRANT == 1	FatFs可重入配置, 需要多任务系统支持(一般不需要)
ff_mem_alloc ff_mem_free	_USE_LFN == 3	长文件名支持, 缓冲区设置在堆空间(一般设置_USE_LFN = 2)

这些命令
需要自己
编底层函
数实现

零死角玩转STM32



THANKS

论坛：www.firebbs.cn

淘宝：firestm32.taobao.com



扫描进入淘宝店铺