

Fatfs Cube配置详解

http://elm-chan.org/fsw/ff/doc/config.html

配置选项

功能配置

FF_FS_READONLY

读/写（0）或只读（1）。只读配置还删除了编写 **API** 函数f_write, f_sync, f_unlink, f_mkdir, f_chmod, f_rename, f_truncate, f_getfree和可选的编写函数。

FF_FS_MINIMIZE

此选项定义最小化级别，以删除一些基本的API函数，如下所示：

值	描述
0	所有基本API函数均可用。
1个	f_stat, f_getfree, f_unlink, f_mkdir, f_chmod, f_utime, f_truncate和f_rename函数被删除。
2	除1之外，还删除了f_opendir, f_readdir和f_closedir函数。
3	除2之外，还将删除f_lseek函数。

FF_USE_STRFUNC

此选项切换字符串函数f_gets, f_putc, f_puts和f_printf。

值	描述
0	禁用字符串函数。
1个	启用不进行LF-CRLF转换的字符串函数。
2	使用LF-CRLF转换启用字符串功能。

FF_USE_FIND

禁用（0）或启用（1）过滤的目录读取功能f_findfirst和f_findnext。FF_FS_MINIMIZE也必须为0或1。

FF_USE_MKFS

禁用（0）或启用（1）f_mkfs功能。

FF_USE_FASTSEEK

禁用（0）或启用（1）快速查找功能可启用f_lseek, f_read和f_write功能的加速模式。有关更多信息，请阅读[此处](#)。

FF_USE_EXPAND

禁用（0）或启用（1）f_expand函数。

FF_USE_CHMOD

禁用（0）或启用（1）元数据控制功能f_chmod和f_utime。FF_FS_READONLY也必须为0。

FF_USE_LABEL

对卷标, f_getlabel和f_setlabel禁用（0）或启用（1）API函数。

FF_USE_FORWARD

禁用（0）或启用（1）f_forward函数。

命名空间和语言环境配置

FF_CODE_PAGE

此选项指定目标系统上使用的OEM代码页。代码页设置不正确会导致文件打开失败。如果未将任何非ASCII字符用作路径名，或者字符编码为Unicode，则任何代码页设置之间都没有区别。无论如何将其设置为437。

值	代码页
0	包括以下所有代码页，并由f_setcp（）设置
437	美国

720	阿拉伯
737	希腊语
771	KBL
775	波罗的海
850	拉丁1
852	拉丁2
855	西里尔
857	土耳其
860	葡萄牙语
861	冰岛的
862	希伯来语
863	加拿大法语
864	阿拉伯
865	北欧风
866	俄语
869	希腊文2
932	日语（DBCS）
936	简体中文（DBCS）
949	韩文（DBCS）
950	繁体中文（DBCS）

FF_USE_LFN

此选项切换对长文件名（LFN）的支持。启用LFN时，需要将Unicode支持模块ffunicode.c添加到项目中。当使用堆栈作为工作缓冲区时，请注意堆栈溢出。当使用堆内存作为工作缓冲区时，需要将内存管理功能（ff_memalloc和ff_memfree）添加到项目中。

值	描述
0	禁用LFN。只能使用8.3格式的路径名。
1个	在BSS上启用带有静态工作缓冲区的LFN。始终不是线程安全的。
2	在堆栈上启用具有动态工作缓冲区的LFN。
3	在HEAP上启用具有动态工作缓冲区的LFN。

FF_MAX_LFN

LFN函数需要某些内部工作缓冲区来获取文件名。此选项定义缓冲区的大小，该值的范围可以是LFN的UTF-16编码单位，范围为12到255。启用exFAT时，缓冲区占用（FF_MAX_LFN + 1）* 2个字节，另外占用（FF_MAX_LFN + 44）/ 15 * 32个字节。建议将其设置为255，以完全支持LFN规范。如果未启用LFN，则此选项无效。

FF_LFN_UNICODE

此选项切换API上文件名的字符编码。当选择Unicode时，FF_CODE_PAGE实际上没有任何意义，除了与旧系统（例如MS-DOS和不支持LFN的系统）兼容。FatFs支持最多U + 10FFFF的代码点。

值	字符编码	TCHAR
0	当前CP中的ANSI / OEM	烧焦
1个	UTF-16中的Unicode	WCHAR
2	UTF-8中的Unicode	烧焦
3	UTF-32中的Unicode	双字

此选项还会影响字符串I / O函数的行为（请参阅FF_STRF_ENCODE）。如果未启用LFN，则此选项无效，并且FatF在API上的ANSI / OEM代码下工作。有关更多信息，请阅读[此处](#)。

FF_LFN_BUF, FF_SFN_BUF

这组选项在**FILINFO**结构中定义文件名成员**fname []**和**altname []**的大小，该结构用于读取目录项。这些值应足以读取文件名。读取文件名的最大可能长度取决于API上的字符编码，如下所示：

编码方式	LFN长度	SFN长度
SBCS的ANSI / OEM	255项目	12项
DBCS的ANSI / OEM	510项目	12项
UTF-16 / 32中的Unicode	255项目	12项
UTF-8中的Unicode	765项目	34项目

如果名称成员的大小不足以使用LFN，则将该项目视为没有LFN。未启用LFN时，这些选项无效。

FF_STRF_ENCODE

当API上的字符编码为Unicode（**FF_LFN_UNICODE**> = 1）时，字符串I / O函数**f_gets**，**f_putc**，**f_puts**和**f_printf**转换其中的字符编码。该选项定义了通过这些功能读写**文件**的字符编码的假设。当未启用LFN或**FF_LFN_UNICODE** == 0时，字符串函数无需任何编码转换即可工作，并且此选项无效。

值	文件上的字符编码
0	当前代码页中的ANSI / OEM
1个	UTF-16LE中的Unicode
2	UTF-16BE中的Unicode
3	UTF-8中的Unicode

FF_FS_RPATH

此选项配置相对路径功能。有关更多信息，请阅读[此处](#)。

值	描述
0	禁用相对路径功能并删除相关功能。
1个	启用相对路径功能。 f_chdir 和 f_chdrive 功能可用。
2	f_getcwd 函数除了1外还可用

卷/驱动器配置

FF_VOLUMES

此选项配置要使用的卷数（逻辑驱动器最多10个）。

FF_STR_VOLUME_ID

此选项切换对字符串卷ID的支持。当为驱动器前缀启用卷ID的任意字符串时，**FF_VOLUME_STRS**的预定义字符串或用户定义的字符串也可以用作路径名中的驱动器前缀。不论此选项如何，数字驱动器号始终有效，并且该选项还可以启用驱动器前缀的任何一种格式。

值	描述	例
0	只能使用数字ID中的DOS / Windows样式驱动器前缀。	0: /文件名
1个	也可以使用字符串ID中的DOS / Windows样式驱动器前缀。	flash: /文件名
2	也可以使用字符串ID中的Unix样式驱动器前缀。	/ flash /文件名

FF_VOLUME_STRS

此选项定义每个逻辑驱动器的卷ID字符串。项数不能小于**FF_VOLUMES**。卷ID字符串的有效字符为AZ，az和0-9，但是，它们在不区分大小写的情况下进行比较。如果**FF_STR_VOLUME_ID** == 0，则此选项无效。如果**FF_STR_VOLUME_ID**> = 1并且未定义此选项，则需要定义用户定义的卷字符串表，如下所示。该表不应即时修改。

```
1 | / *用户定义的0: 1: 2: 3: 卷ID字符串* /
2 | const char * VolumeStr [FF_VOLUMES] = {" ram", " flash", " sdc", " usb"};
```

FF_MULTI_PARTITION

禁用（0）或启用（1）。此选项切换多分区功能。默认情况下（0），每个逻辑驱动器号都绑定到相同的物理驱动器号，并且仅安装物理驱动器中的一个卷。启用后，每个逻辑驱动器都绑定到用户定义的分区解析表**VolToPart []**中列出的物理驱动器上的分区。此外**f_fdisk**功能可按可用。有关更多信息，请阅读[此处](#)。

FF_MIN_SS, FF_MAX_SS

这组选项定义了用于低级磁盘I/O接口， disk_read和disk_write函数的扇区大小范围。有效值为512、1024、2048和4096。FF_MIN_SS定义最小扇区大小，FF_MAX_SS定义最大扇区大小。始终将存储卡和硬盘都设置为512。但是，板载闪存和某些类型的光学介质可能需要更大的值。当FF_MAX_SS> FF_MIN_SS时，启用对可变扇区大小的支持，并且需要对disk_ioctl函数实施GET_SECTOR_SIZE命令。

FF_LBA64

此选项将媒体访问接口切换到64位LBA，并启用GUID分区表（GPT）进行分区管理（已启用（1）或已禁用（0））。需要启用exFAT文件系统才能启用此功能。

FF_MIN_GPT

该选项指定在驱动器上使用f_mkfs和f_fdisk函数创建分区时确定分区格式的阈值。当驱动器上的扇区数等于或大于此值时，驱动器将在GPT中分区。当FF_LBA64 == 0时，此选项无效。

FF_USE_TRIM

禁用（0）或启用（1）。此选项切换ATA-TRIM功能。要启用Trim功能，还应该对disk_ioctl函数实施CTRL_TRIM命令。

系统配置

FF_FS_TINY

普通（0）或微小（1）。微小的配置减少了FIL结构，文件对象和FF_MAX_SS字节的大小。FATFS结构中的公用扇区缓冲区（文件系统对象）不是从文件对象中删除的专用扇区缓冲区，而是用于文件数据传输。

FF_FS_EXFAT

除FAT / FAT32文件系统外，此选项还支持exFAT文件系统，即启用（1）或禁用（0）。要启用exFAT，还必须启用LFN，并建议将FF_LFN_UNICODE> = 1和FF_MAX_LFN == 255配置为功能齐全的exFAT功能。请注意，启用exFAT会放弃ANSI C（C89）兼容性，并且由于需要64位整数类型而需要C99。

FF_FS_NORTC

使用RTC（0）或不使用RTC（1）。此选项控制时间戳功能。如果系统没有任何RTC功能，或者不需要有效的时间戳，请将FF_FS_NORTC设置为1可禁用时间戳功能。FatFs修改的每个对象都将具有由FF_NORTC_MON，FF_NORTC_MDAY和FF_NORTC_YEAR定义的固定时间戳。要使用时间戳功能，请设置FF_FS_NORTC == 0并将get_fattime函数添加到项目中以从RTC获取当前时间。此选项对只读配置无效。

FF_NORTC_MON, FF_NORTC_MDAY, FF_NORTC_YEAR

这组选项定义了没有RTC系统使用的时间。该选项对只读配置或FF_FS_NORTC == 0无效。

FF_FS_NOFSINFO

0到3。如果您需要知道FAT32卷上的正确可用空间，请设置此选项的位0，并且在卷装入后第一次启动f_getfree函数将强制执行完整的FAT扫描。位1控制最后分配的群集号的使用。

值	描述
位0 = 0	如果可用，请在FSINFO中使用可用的群集计数。
位0 = 1	不要信任FSINFO中的可用群集计数。
位1 = 0	使用FSINFO中最后分配的群集号查找可用的群集（如果有）。
位1 = 1	不要信任FSINFO中最后分配的集群号。

FF_FS_LOCK

此选项切换文件锁定功能以控制重复文件的打开和非法操作以打开对象。请注意，文件锁定功能与重新输入无关。在只读配置下，此选项必须为0。

值	描述
0	禁用文件锁定功能。为避免文件因错误的文件操作而崩溃，应用程序需要避免非法打开，删除并重命名为打开的对象。
> 0	启用文件锁定功能。该值定义在文件锁定控制下可以同时打开多少个文件/子目录。对打开对象的非法操作将被FR_LOCKED拒绝。

FF_FS_REENTRANT

禁用（0）或启用（1）。此选项可切换FatFs模块本身的重新进入（线程安全）。请注意，始终可以重新进入对不同卷的文件/目录访问权限，并且无论该选项如何，它都可以同时工作，但是，卷管理功能f_mount，f_mkfs和f_fdisk始终不可以重入。此功能控制着仅文件/目录访问同一卷，换句话说，每个文件系统对象的独占使用。为了启用此功能，用户还提供了同步处理程序ff_req_grant，ff_rel_grant，ff_del_syncobj和ff_cre_syncobj，需要添加到项目中。ffsystem.c中提供了示例代码。

FF_FS_TIMEOUT

等待时间过长时，使用FR_TIMEOUT中止文件功能的时间间隔。当FF_FS_REENTRANT == 0时，此选项无效。

FF_SYNC_t

此选项定义与O / S相关的同步对象类型。例如HANDLE，ID，OS_EVENT *，SemaphoreHandle_t等。用于O / S定义的头文件需要包含在ff.c范围内的某个位置。当FF_FS_REENTRANT == 0时，此选项无效。