

DV10 VFS 接口说明

函数接口说明:

函数原型	Void init_VFS(int max_file_no)	
功能描述	VFS 相关数据结构的初始化等	
参数说明	max_file_no	允许同时打开的文件个数
返回	无	

函数原型	u32 scan_fs(DEV_HANDLE dev_io, u32 opt)	
功能描述	返回存储设备上分区个数;	
参数说明	Dev_io	存储设备句柄
	opt	扩展用
返回	设备上分区个数，若没有注册该设备的文件系统，则返回-1	

函数原型	FRESULT mount_partition(DEV_HANDLE dev_io , u32 partition_index, char* drv)	
功能描述	挂载磁盘某分区	
参数	dev_io	存储设备句柄

说明	partition_index	指定挂载的存储设备上的分区
	drv	挂载成功后，保存分区挂载后的盘符
返回	0: 成功；非 0: 不成功 挂载成功后，会将默认路径切换至此盘根目录	

函数原型	FRESULT unmount_partition (const char drv)	
功能描述	卸载指定盘符的分区	
参数说明	drv	盘符
返回	0: 成功；非 0: 不成功	

函数原型	mkfs_partition(DEV_HANDLE dev_io, u32 partition_index)	
功能描述	格式化指定分区	
参数说明	dev_io	存储设备句柄
	partition_index	指定挂载的存储设备上的分区
返回	0: 成功；非 0: 不成功	

函数原型	FRESULT mkfs_drv(char drv)	
功能	格式化指定盘符	

描述		
参数说明	drv	盘符
返回	0: 成功; 非 0: 不成功	

函数原型	u32 reg_fs(fs_adapter* the_fs_adapter)	
功能描述	注册一个 FS 到 VFS	
参数说明	the_fs_adapter	一个具体 FS 适配器
返回	成功返回 0, 否则不成功	

数据结构	fs_adapter	
描述	具体 FS 适配器	
字段说明	char *devName;	存储设备名 注意: 通用文件系统时空, 专用文件系统时指定设备名
	void (*init_cur_fat)();	FS 初始化, 注册时自动调用
	u32 (*get_partition_nums)();	获取设备上分区个数 注意: 通用文件系统时没有该功能, 专用文件系统时, 调用此接口扫描并返回存储设备上的分区个数
	u32 (*mount_partition)(u32 partition_index, device_info *the_device);	挂载指定设备上的指定分区

u32 (*fopen)(void *filehandle, u8* path, u32 opt, driver_info *the_driver);	打开文件
u32 (*fread)(void *fp, u8 *ptr, u32 len, driver_info *the_driver);	读文件
u32 (*fwrite)(void *fp, u8* ptr, u32 len, driver_info *the_driver);	写文件
u32 (*fseek)(void *fp, u32 len, u32 opt, driver_info *the_driver);	移动文件游标
u32 (*ftell)(void *fp, driver_info *the_driver);	返回当前文件游标
void (*fclose)(void *fp, driver_info *the_driver);	关闭文件
u32 (*fdelete)(u8* path, driver_info *the_driver);	删除文件
u32 (*get_file_name)(void *fp, u8 * ptr, u32 len, u32 opt, driver_info *the_driver);	返回文件名（有长文件名时返回长文件名）
u32 (*fopen_byindex)(u32 index, driver_info *the_driver, void *fp);	根据索引号打开文件 注：某些专用文件系统有该功能
u32 (*f_emu)(u8* filter, void *pram2callback, driver_info *the_driver, s32 (*callback)(void *fileparam, void *pram2callback));	枚举指定路径下的所有文件目录
u32 (*f_chdir)(u8* path, driver_info *the_driver);	切换目录
u32 (*f_mkdir)(u8* path, u32	创建目录

	opt, driver_info *the_driver);	
	u32 (*get_free_space)(driver_info *the_driver);	返回分区可用空间
	u32 (*get_file_length)(u8* path, driver_info *the_driver);	根据路径文件名返回文件长度
	u8 (*get_file_attr)(u8* path, driver_info *the_driver);	根据路径文件名返回文件目录属性
	u32 (*f_rename)(u8* oldname, u8* newname, driver_info *the_driver);	修改文件名
	u32 (*fsize)(void *fp, driver_info *the_driver);	根据句柄返回文件长度

说明 0: 以上是指具体 FS 的接口，有别于 VFS 的接口

说明 1: 若 FS 没有某接口时，注册的时候，在相应字段配成 NULL.

函数原型	file_t * fopen(u8* path ,u32 opt)	
功能描述	打开指定文件	
参数说明	path	路径
	opt	打开文件参数: FA_READ 读方式打开 FA_WRITE 写方式打开 FA_CREATE_NEW 文件不存的时候则创建新文件

		FA_CREATE_ALWAYS 始终创建新文件
返回	成功：返回文件句柄; 不成功：NULL	

函数原型	u32 fread(file_t *fp,u8 *ptr,u32 len) ;	
功能描述	打开指定文件	
参数说明	fp	文件句柄
	ptr	数据 buf
	len	数据 buf 长度
返回	读取的数据长度，0 表示没有数据或读错误	

函数原型	u32 fwrite(file_t *fp,u8 *ptr,u32 len) ;	
功能描述	打开指定文件	
参数说明	fp	文件句柄
	ptr	数据 buf
	len	数据 buf 长度
返回	写成功的数据长度，<len 表示有读错误	

函数原型	u32 fseek(file_t *fp, u32 opt , u32 len)	
功能描述	移动文件游标 说明：当前所用的 gen_fs 中，往文件后移时，若移动的范围超出文件大小，则会自动相应加长文件，此功能主要用于创建大文件使用，平时使用时需注意此特点	
参数说明	fp	文件句柄
	len	移动长度
	opt	移动方式： SEEK_CUR:从当前位置开始移 SEEK_END:从文件末尾开始移 SEEK_SET: 从文件头开始移
返回	0: 成功；非 0：不成功	

函数原型	U32 ftell(file_t *fp)	
功能描述	返回文件游标	
参数说明	Fp	文件句柄
返回	-1: 出错; 否则返回值为文件当前游标	

函数原型	u32 fclose(file_t *fp) ;	
功能描述	关闭文件	

参数说明	Fp	文件句柄
返回	0: 成功; 非 0: 不成功	

函数原型	u32 fdelete(u8* path)	
功能描述	删除文件	
参数说明	path	文件名
返回	0: 成功; 非 0: 不成功	

函数原型	u32 frename(u8* oldpath ,u8* newpath)	
功能描述	更换文件名	
参数说明	oldpath	旧文件名
	newpath	新文件名
返回	0: 不成功; 非 0: 成功	

函数原型	u32 get_file_name(file_t *fp ,u8 * ptr ,u32 len)	
功能描述	获取文件名，有长文件名时，获取的是长文件名	

参数说明	fp	文件句柄
	ptr	Buf, 用以保存取得的文件名
	len	Buf 长度
返回	0: 不成功; 非 0: 文件名长度	

函数原型	file_t * fopen_byindex(u32 index, u32 opt)	
功能描述	在当前盘根据索引号打开文件	
参数说明	index	文件索引号
	opt	同 fopen 中相应参数
返回	成功返回文件句柄; 不成功返回 NULL	

函数原型	u32 f_emu(u8* path, void *pram2callback, s32 (*callback)(void *fileparam, void *pram2callback))	
功能描述	枚举指定路径下的文件	
参数说明	path	枚举路径
	pram2callback	传给回调函数的用户指定参数指针, 若不需要时, 用 NULL 即可

	callback	<p>回调函数，非空时，每搜到一个文件均会回调一次，以文件名为参数</p> <p>注 1: 当 callback 返回 FALSE 时，枚举继续; 返回 TRUE 时，中止枚举;</p> <p>注 2: callback 的参数 fileparam 是个结构体指定，可定义如下：</p> <pre>typedef struct { u32 fsize; u8 fattrib; u8 fname[15]; u32 filename_len; u8 filename[256]; }FILE_PARAM;</pre>
返回	枚举到的文件目录个数	

函数原型	u32 f_chdir(u8* path)	
功能描述	切换当前目录	
参数说明	path	新目录
返回	0: 成功；非 0: 不成功	

函数原型	u32 f_mkdir (u8* path)	
功能描述	创建目录	

参数说明	path	目录名
返回	0: 成功; 非 0: 不成功	

函数原型	f_getdir(u8* buf, u32 len)	
功能描述	返回当前目录	
参数说明	buf	保存目录的 buf
	len	Buf 长度
返回	-1: 不成功; 否则为目录长度	

函数原型	u32 get_free_space(u8 drv);	
功能描述	返回指定盘的空闲空间	
参数说明	drv	盘符
返回	空闲空间 (以 512 为单位)	

函数原型	u32 get_logic_drvs(char * drvs,int count);	
功能描述	获取当前逻辑盘符数	
参数说明	drvs	用来保存逻辑盘符的 buf

	count	drvs 长度
返回	当前逻辑盘符的数量 注意：如果此数大于 count，则说明事先传入的 drvs 太小	

其它:

- 1、同时支持的设备、挂载的分区及文件系统种类个数受限;
- 2、分区挂载后的盘符为 A~Z。
- 3、同时打开的文件个数受限;
- 4、标准绝对路径示例: "C:/test/aaa.mp3";
- 5、不支持 NTFS;
- 6、磁盘分区上, 扩展分区及其内的逻辑分区不支持;
- 7、暂不支持以文本方式打开及操作文件, 故如 `fprintf` 这一类函数暂时不支持。