

## W12: 设备驱动程序中的文件操作结构

Linux 中采用 VFS 文件系统为各类文件系统提供了一个统一接口，其中有 file、inode、super\_block、dentry 四个结构，对于每个结构，又抽象出了对应的 operations 结构，用于操作该结构。

Linux2.4 中，VFS 文件系统相关的结构定义在 Linux/include/linux/fs.h 中，`struct file_operations` 是文件操作结构。

```
858 /*
859  * NOTE:
860  * read, write, poll, fsync, readv, writev can be called
861  * without the big kernel lock held in all filesystems.
862  */
863 struct file_operations {
864     struct module *owner;
865     loff_t (*llseek) (struct file *, loff_t, int);
866     ssize_t (*read) (struct file *, char *, size_t, loff_t *);
867     ssize_t (*write) (struct file *, const char *, size_t, loff_t *);
868     int (*readdir) (struct file *, void *, filldir_t);
869     unsigned int (*poll) (struct file *, struct poll_table_struct *);
870     int (*ioctl) (struct inode *, struct file *, unsigned int,
unsigned long);
871     int (*mmap) (struct file *, struct vm_area_struct *);
872     int (*open) (struct inode *, struct file *);
873     int (*flush) (struct file *);
874     int (*release) (struct inode *, struct file *);
875     int (*fsync) (struct file *, struct dentry *, int datasync);
876     int (*fasync) (int, struct file *, int);
877     int (*lock) (struct file *, int, struct file_lock *);
878     ssize_t (*readv) (struct file *, const struct iovec *, unsigned
long, loff_t *);
879     ssize_t (*writev) (struct file *, const struct iovec *, unsigned
long, loff_t *);
880     ssize_t (*sendpage) (struct file *, struct page *, int, size_t,
loff_t *, int);
881     unsigned long (*get_unmapped_area)(struct file *, unsigned long,
unsigned long, unsigned long, unsigned long);
882 };
```