

嵌入式技术

GNU/Linux 文件系统

邢超

<EC.1>

1 文件系统

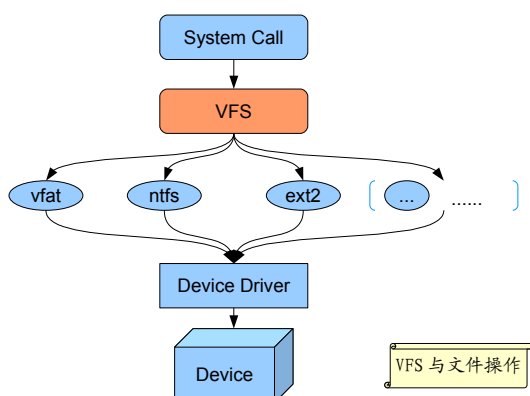
Linux 文件系统种类

- romfs
- ramdisk
- tmpfs
- JFFS/JFFS2 (Journaling Flash File System)
- YAFFS (Yet Another Flash File System)
- cramfs
- SquashFS
- NFS

<EC.2>

1.1 VFS

VFS



<EC.3>

1.2 EXT2 文件系统

EXT2 文件系统

- 引导块：在文件系统的开头，通常为一个扇区，存放引导程序，用于读入并启动操作系统
- 超级块 (superblock)：用于记录文件系统的管理信息。特定的文件系统定义了特定超级块
- inode 区 (索引节点)：一个文件 (或目录) 占据一个索引节点。利用根节点 (首个索引节点)，可以把一个文件系统挂在另一个文件系统的非叶节点上
- 数据区：用于存放文件数据或者管理数据 (如一级间址块、二级间址块等)

<EC.4>

文件存储方式

- EXT2 中文件由逻辑块的序列组成。数据块的长度相同。不同的 EXT2 系统长度可以不同。文件总是整块存储，不足一块的部分也占用一个数据块
- EXT2 中的每个文件都用一个单独的 inode (即 struct ext2_inode 结构) 来描述，而每个 inode 都有一个唯一的标志号。通过使用 inode 来定义文件系统的结构以及描述系统中每个文件的管理信息

<EC.5>

1.3 romfs File System

romfs 文件系统

- romfs (rom file system) 是一种只读文件系统
- 管理代码占用的空间比较小
- 使用 genromfs 工具创建
- 未完全实现文件访问权限
- 如果要进行写操作：
 - 在编译的时候加上写访问功能
 - 或者在运行时另外生成一个 RAMdisk 送暂存数据。

<EC.6>

- 文件系统结构
- 文件结构

<EC.7>

文件系统结构

include/linux/binfmts.h:

```
struct romfs_super_block {
    __u32 word0; /* -rom
    */
    __u32 word1; /* 1fs -
    */
    __u32 size; /* size of
    the file system */
    __u32 checksum; /* checksum
    of first 512k */
    char name[0]; /* volume
    name */
};
```

<EC.8>

文件头结构

include/linux/binfmts.h:

```
struct romfs_inode {
    __u32 next; /* low 4 bits for
    file type*/
    __u32 spec; /* spec.info */
    __u32 size; /* file size */
    __u32 checksum; /* checksum of
    file */
    char name[0]; /* file name */
};
```

<EC.9>

spec.info

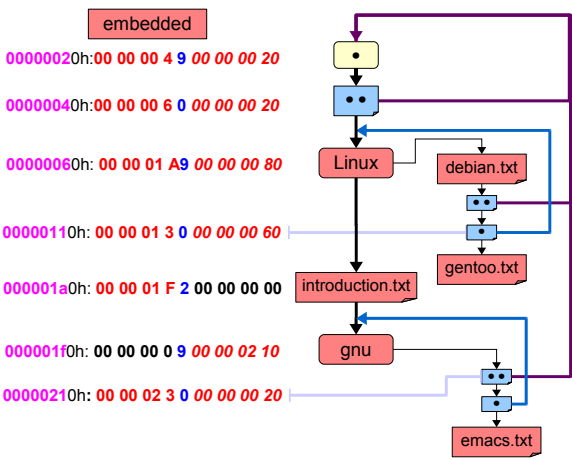
- 0: 硬链接, spec.info 域的内容用于链接的目标文件
- 1: 目录, spec.info 域的内容为第一个文件的文件头
- 2: 普通文件, spec.info 域的内容无效, 应设置为 0
- 3: 符号链接, spec.info 域的内容无效, 应设置为 0
- 4: 块设备, spec.info 域内容为各 16bit 的主、从设备号
- 5: 字符设备, spec.info 域的内容无效, 设置为 0
- 6: 网络 socket 套接字 spec.info 域的内容无效, 设置为 0
- 7: fifo 管道文件, spec.info 域的内容无效, 设置为 0

<EC.10>

```
00000000h: 2D 72 6F 6D 31 66 73 2D 00 00 02 B0 B9 4C BD 1D; -rom1fs... * L?
00000010h: 65 6D 62 65 64 64 65 64 00 00 00 00 00 00 00 00; embedded.....
00000020h: 00 00 00 49 00 00 00 20 00 00 00 00 D1 FF FF 97; .....? ?
00000030h: 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00; .....
00000040h: 00 00 00 60 00 00 00 20 00 00 00 00 D1 D1 FF 80; ..... 蛭 * e
00000050h: 2E 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00; .....
00000060h: 00 00 01 A9 00 00 00 80 00 00 00 00 1B 96 8F 62; ...?..€... 柁 b
00000070h: 6C 69 6E 75 78 00 00 00 00 00 00 00 00 00 00 00; linux.....
00000080h: 00 00 00 F2 00 00 00 00 00 00 00 00 43 C1 B8 6D EE; ...?..C 粮 m?
00000090h: 64 65 62 69 61 6E 2E 74 78 74 00 00 00 00 00 00; debian.txt.....
000000A0h: 00 00 01 10 00 00 00 20 00 00 00 00 D1 D1 FE D0; ..... 蛭 *
000000B0h: 2E 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00; .....
000000C0h: 00 00 01 30 00 00 00 60 00 00 00 00 D1 FF FE 70; ...0...? ?
000000D0h: 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00; .....
000000E0h: 00 00 00 02 00 00 00 00 00 00 00 00 4B B0 B7 62 CB; ..... K 按 b?
000000F0h: 67 65 6E 74 6F 6F 2E 74 78 74 00 00 00 00 00 00; gentoo.txt.....
0000010h: 00 00 01 F2 00 00 00 00 00 00 00 00 1A 84 4F 2C 3D; ...?.. 虱, =
00000110h: 69 6E 74 72 6F 64 75 63 74 69 6F 6E 2E 74 78 74; introduction.txt
00000120h: 00 00 00 09 00 00 02 10 00 00 00 00 98 91 88 E7; ..... *
00000130h: 67 6E 75 00 00 00 00 00 00 00 00 00 00 00 00 00; gnu.....
00000140h: 00 00 02 30 00 00 00 20 00 00 00 00 D1 D1 FD B0; ...0... 蛭 *
00000150h: 2E 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00; .....
00000160h: 00 00 02 50 00 00 01 F0 00 00 00 00 D1 FF FB C0; ...P...? ?
00000170h: 2E 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00; .....
00000180h: 00 00 00 02 00 00 00 00 00 00 00 00 35 B3 64 29 EE; ..... 5 * )?
00000190h: 65 6D 61 63 73 2E 74 78 74 00 00 00 00 00 00 00; emacs.txt.....
00000200h: 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00; .....
```

<EC.11>

romfs 示例



<EC.12>

2 文件系统使用

2.1 mount

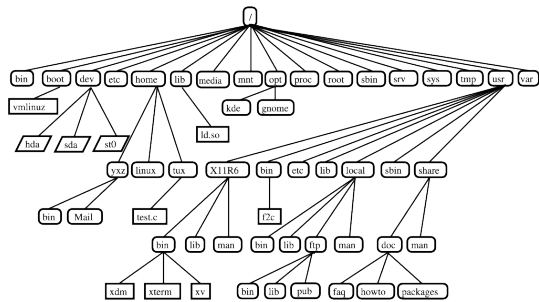
文件系统使用

- etc/fstab
- mount/umount

<EC.13>

2.2 FHS

FHS: Filesystem Hierarchy Standard



<EC.14>

3 思考

思考

- GNU/Linux 常用哪些文件系统？
- romfs 的优缺点？

<EC.15>