**1.1 什么是混杂设备**

混杂设备（Miscellaneous device）是 Linux 字符设备驱动的一种简化机制，适用于功能较简单、无需自己管理主设备号的场景。

常见设备如 /dev/rtc、/dev/mem、/dev/input/eventX 等，很多都是通过混杂设备注册的。

**1.2 混杂设备结构体**

c

复制代码

struct miscdevice {

int minor; // 次设备号，MISC\_DYNAMIC\_MINOR 表示自动分配

const char \*name; // 设备名称，会在 /dev 下创建相应文件

const struct file\_operations \*fops; // 文件操作集

struct device \*parent; // 父设备

struct list\_head list; // 链表，内核使用

struct device \*this\_device;// 指向设备结构体

};

**1.3 注册与注销**

c

复制代码

// 注册设备

int misc\_register(struct miscdevice \*misc);

// 注销设备

int misc\_deregister(struct miscdevice \*misc);

🧠 你无需自己申请主设备号，也不用 cdev 注册，只要填好结构体即可。

**1.4 示例代码片段**

c

复制代码

static struct miscdevice myled\_misc = {

.minor = MISC\_DYNAMIC\_MINOR,

.name = "myled",

.fops = &myled\_fops,

};

misc\_register(&myled\_misc); // 注册

misc\_deregister(&myled\_misc); // 注销

**1.5 常见混杂设备面试题**

| **题目** | **要点** |
| --- | --- |
| 混杂设备和普通字符设备的区别？ | 混杂设备简化了主设备号管理，适合功能简单的设备 |
| 如何在驱动中创建 /dev/myled 设备文件？ | 使用 misc\_register() 并设置 .name="myled" |
| 混杂设备能否注册多个？ | 可以，多数通过动态次设备号分配 |