

Linux网络设备驱动编程Linux认证考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/644/2021_2022_Linux_E7_BD_91_E7_BB_c103_644791.htm 在此仅仅讨论网络设备驱动的一般写法，有关硬件部分的相关代码由于硬件规格不同，予以省略。有什么地方错误，或补充，欢迎大家提出。

1, 驱动模块的加载和卸载 如果网络设备（包括wireless）是PCI规范的，则先是向内核注册该PCI设备(pci_register_driver)，然后由pci_driver数据结构中的probe函数指针所指向的侦测函数来初始化该PCI设备，并且同时注册和初始化该网络设备。如果网络设备（包括wireless）是PCMCIA规范的，则先是向内核注册该PCMCIA设备(register_pccard_driver)，然后driver_info_t数据结构中的attach函数指针所指向的侦测函数来初始化该PCMCIA设备，并且同时注册和初始化该网络设备。

```
static int __init tg3_init(void) { //先注册成PCI设备，并初始化，如果是其他的ESIA，PCMCIA，用其他函数 return pci_module_init(amp.tg3_driver).//注销PCI设备 } module_init(tg3_init). //驱动模块的加载 module_exit(tg3_cleanup). //驱动模块的卸载 申明为PCI设备： static struct pci_driver tg3_driver = { .name = DRV_MODULE_NAME, .id_table = tg3_pci_tbl, //此驱动所支持的网卡系列， vendor_id, device_id .probe = tg3_init_one, //初始化网络设备的回调函数 .remove = __devexit_p(tg3_remove_one), //注销网络设备的回调函数 .suspend = tg3_suspend, //设备挂起函数 .resume = tg3_resume //设备恢复函数 }.
```

2，PCI设备探测函数probe，初始化网络设备

```
static int __devinit tg3_init_one(struct pci_dev *pdev, const struct
pci_device_id *ent) { //初始化设备，使I/O，memory可用，唤醒设备
pci_enable_device(pdev). //申请内存空间，配置网卡的I/O，memory资源
pci_request_regions(pdev, DRV_MODULE_NAME). pci_set_master(pdev). //设置DMA属性
pci_set_dma_mask(pdev, (u64) 0xffffffffffff). //网卡I/O,memory资源的起始地址
tg3reg_base = pci_resource_start(pdev, 0). //网卡I/O,memory资源的大小
tg3reg_len = pci_resource_len(pdev, 0). //分配并设置网络设备
dev = alloc_etherdev(sizeof(*tp)). //申明为内核设备模块
SET_MODULE_OWNER(dev). //初始化私有结构中的各成员值
tp = dev->pdev = pdev. tp->regs = (unsigned long) ioremap(tg3reg_base, tg3reg_len).
dev->irq. //网络设备回调函数赋值
dev->stop = tg3_close. dev->set_multicast_list = tg3_set_rx_mode.
dev->do_ioctl = tg3_ioctl. dev->hard_start_xmit= tg3_start_xmit. //网卡的MAC地址赋值
dev->addr tg3_get_device_address(tp). //注册网络设备
```

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com