

入门文章：教你学会编写Linux设备驱动之一 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/144/2021_2022__E5_85_A5_E9_97_A8_E6_96_87_E7_c103_144256.htm 内核版本: 2.4.22
阅读此文的目的: 学会编写Linux设备驱动。阅读此文的方法: 阅读以下2个文件: hello.c,asdf.c。此文假设读者:已经能用C语言编写Linux应用程序,理解"字符设备文件,块设备文件,主设备号,次设备号",会写简单的Shell脚本和Makefile。 1.

```
"hello.c"-----/** 这是我们的第一个源文件，* 它是一个可以加载的内核模块，* 加载时显示"Hello,World!"，* 卸载时显示"Bye!"。* 需要说明一点，写内核或内核模块不能用写应用程序时的系统调用或函数库，* 因为我们写的就是为应用程序提供系统调用的代码。* 内核有专用的函数库，如,，等,* 现在还没必要了解得很详细，* 这里用到的printk的功能类似于printf。* "/usr/src/linux"是你实际的内核源码目录的一个符号链接，* 如果没有现在就创建一个,因为下面和以后都会用到。* 编译它用"gcc -c -I/usr/src/linux/include hello.c"，* 如果正常会生成文件hello.o,* 加载它用"insmod hello.o",* 只有在文本终端下才能看到输出。* 卸载它用"rmmod hello"*/** 小技巧: 在用户目录的.bashrc里加上一行:* alias mkmod=gcc -c -I/usr/src/linux/include* 然后重新登陆Shell,* 以后就可以用"mkmod hello.c"的方式来编译内核模块了。*/** 开始例行公事 */#ifndef __KERNEL__# define __KERNEL__#endif#endif# include# include 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
```