

**头文件：**

#include <linux/module.h>

#include <linux/module.h>

所有的Linux 代码必须遵循GPL 协议

如果不声明GPL 协议，模块将无法在Linux 中使用的

MODULE\_LICENSE(\_license)添加遵循GPL协议

MODULE\_AUTHOR(\_author)代码作者

#include <linux/init.h>

包含初始化宏定义的头文件,代码中的函数module\_init和module\_exit在此文件中

入口函数module\_init(x)

出口函数module\_exit(x)

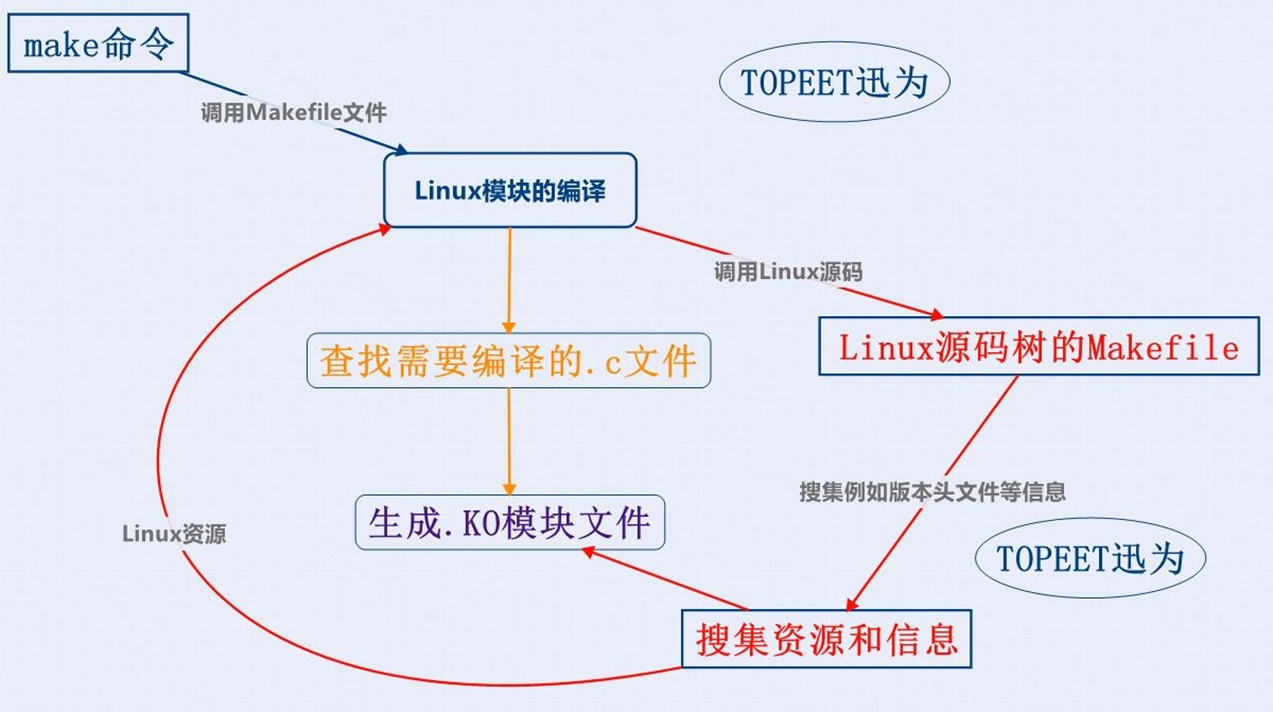
**代码：**

1. #include <linux/init.h>
2. #include <linux/module.h>
4. MODULE\_LICENSE("Dual BSD/GPL");
5. MODULE\_AUTHOR("LX");
7. **static** **int** hello\_init(**void**)
8. {
9. printk(KERN\_EMERG "HELLO WORLD !\n");
10. //KERN\_EMERG表示紧急信息
11. **return** 0;
12. }
14. **static** **void** hello\_exit(**void**)
15. {
16. printk(KERN\_EMERG "HELLO WORLD exit!\n");
18. }
20. module\_init(hello\_init);
21. module\_exit(hello\_exit);

**Makefile文件：**

1. #!/bin/bash
2. #通知编译器我们要编译模块的哪些源码
3. #这里是编译itop4412\_hello.c这个文件编译成中间文件itop4412\_hello.o
4. obj-m += mini\_linux\_module.o
6. #源码目录变量，这里用户需要根据实际情况选择路径
7. #作者是将Linux的源码拷贝到目录/home/topeet/android4.0下并解压的
8. KDIR := /home/topeet/android4.0/iTop4412\_Kernel\_3.0
10. #当前目录变量
11. PWD ?= $(shell pwd)
13. #make命名默认寻找第一个目标
14. #make -C就是指调用执行的路径
15. #$(KDIR)Linux源码目录，作者这里指的是/home/topeet/android4.0/iTop4412\_Kernel\_3.0
16. #$(PWD)当前目录变量
17. #modules要执行的操作
18. all:
19. make -C $(KDIR) M=$(PWD) modules
21. #make clean执行的操作是删除后缀为o的文件
22. clean:
23. rm -rf \*.o

**编译流程分析：**



**常见问题：**

**无法卸载模块的解决办法**

是用rmmod卸载模块的时候会弹出错误，如下

– rmmod: can't change directory to '/lib/modules': No such file or directory

提示没有'/lib/modules'目录

– 使用命令“#mkdir /lib/modules”，新建一个目录

是用rmmod命令仍然会出现如下错误：

– rmmod: can't change directory to '3.0.15': No such file or directory

提示没有目录'3.0.15'

– 使用命令“#mkdir /lib/modules/3.0.15”,继续建目录

**Ubuntu系统下的Vim编辑器无法显示中文字符**

解决办法，添加编码格式“utf-8”

打开文件“/etc/vim/vimrc”

在最底部添加代码“set fencs=utf-8,GB18030,ucs-bom,default,latin1”