# 《奔跑吧Linux内核\*入门篇》重要实验说明 – 必读

**注意：**

**1. 本书的所有实验都是基于优麒麟Linux 18.04.2，若您安装了其他发行版或者版本，遇到问题，请自行解决。**

**2. 建议在尽量物理机上安装优麒麟，实在不行，使用vmware player虚拟机也可以。**

**3. 笨叔会提供安装和配置好整套环境的 vmware 虚拟机镜像给有需要的小伙伴下载。请留意奔跑吧Linux社区微信公众号和异步社区官网。**

**4. 《奔跑吧Linux内核\*入门篇》是基于ARM32架构来表述，大部分实验基于arm32架构。**

**5. 实验代码是在：**[**https://github.com/figozhang/runninglinuxkernel\_4.0**](https://github.com/figozhang/runninglinuxkernel_4.0)

**分支：rlk\_basic**

**实验代码目录：rlk\_lab/rlk\_basic**

**6. 图书勘误：**[**https://github.com/figozhang/Running-LinuxKernel**](https://github.com/figozhang/Running-LinuxKernel)

**7. 本实验采用的源代码版本：**

**\* Linux内核： 4.0**

**\* arm-linux-gnueabi-gcc编译器：5.5**

**\* aarch64-linux-gnu-gcc编译器：5.5**

**在ubuntu里请使用update-alternatives命令来切换gcc编译到5.x版本。**

**请不要使用高版本的gcc编译器，否则会编译有问题。**

## 1. 配套资源下载

1. 配备全球最丰富的教学和学习资源

配套约600页课件 课件已经上传到异步社区和github。

下载地址：

<https://www.epubit.com/book/detail/39321>

或者：

https://github.com/figozhang/Running-LinuxKernel

1. 配套全套实验代码： <https://github.com/figozhang/runninglinuxkernel_4.0>

**分支：rlk\_basic**

实验代码目录：rlk\_lab/rlk\_basic

1. 配套全套实验环境的vmware虚拟机镜像 （基于优麒麟Linux 18.04）

下载地址：

建议大家到百度网盘上下载最新版本。（2019年6月更新到v3.0版本）

百度网盘：<https://pan.baidu.com/s/1XlJzdUBH_V7IwNRH1R4jrg>

异步社区：<https://www.epubit.com/book/detail/39321> （只有v1.0版本）

谷歌网盘：<https://drive.google.com/open?id=1cTRQ73f0yzhTBoNt6rhHbkbbPTI8j_3P>

1. 配套全套实验的docker镜像

<https://hub.docker.com/r/benshushu/runninglinuxkernel>

1. 配套补充视频，10小时高清，33集真情播放 从3月25号起，33集奔跑吧高清视频在B站首发，每天一集，天天精彩。 请到B站 （https://www.bilibili.com/）搜索：奔跑吧linux内核
2. 配套视频下载。

《奔跑吧》入门篇免费视频通过GPL v2授权，大家可以自由下载。

链接：https://pan.baidu.com/s/1AvTkeOyyJUSBJ\_K0pxip0Q

提取码：le2s

1. 配套音频节目 从3月25号起，33集配套音频节目上线喜马拉雅

<https://www.ximalaya.com/keji/22283701/>

1. 微信交流群，请加奔跑吧客服微信号：runninglinuxkernel
2. 免费知识星球

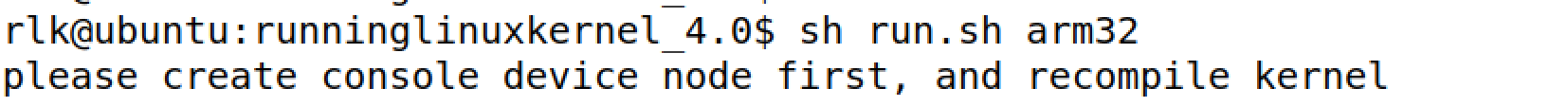


## 2. 编译运行busybox小系统

假设runninglinuxkernel\_4.0目录是在/home/rlk/rlk\_basic/runninglinuxkernel\_4.0

|  |
| --- |
| # cd /home/rlk/rlk\_basic/runninglinuxkernel\_4.0  # export ARCH=arm  # export CROSS\_COMPILE=arm-linux-gnueabi-  # make vexpress\_defconfig |

**如果是第一次运行Qemu虚拟机，那需要手工创建一个console节点。否则运行会提示错误。如下图所示。**



|  |
| --- |
| # cd \_install\_arm32  # mkdir -p dev  #cd dev  # sudo mknod console c 5 1 |

然后编译内核。

|  |
| --- |
| # cd /home/rlk/rlk\_basic/runninglinuxkernel\_4.0  # make -j8 //这里假设你的Linux机器有8个CPU |

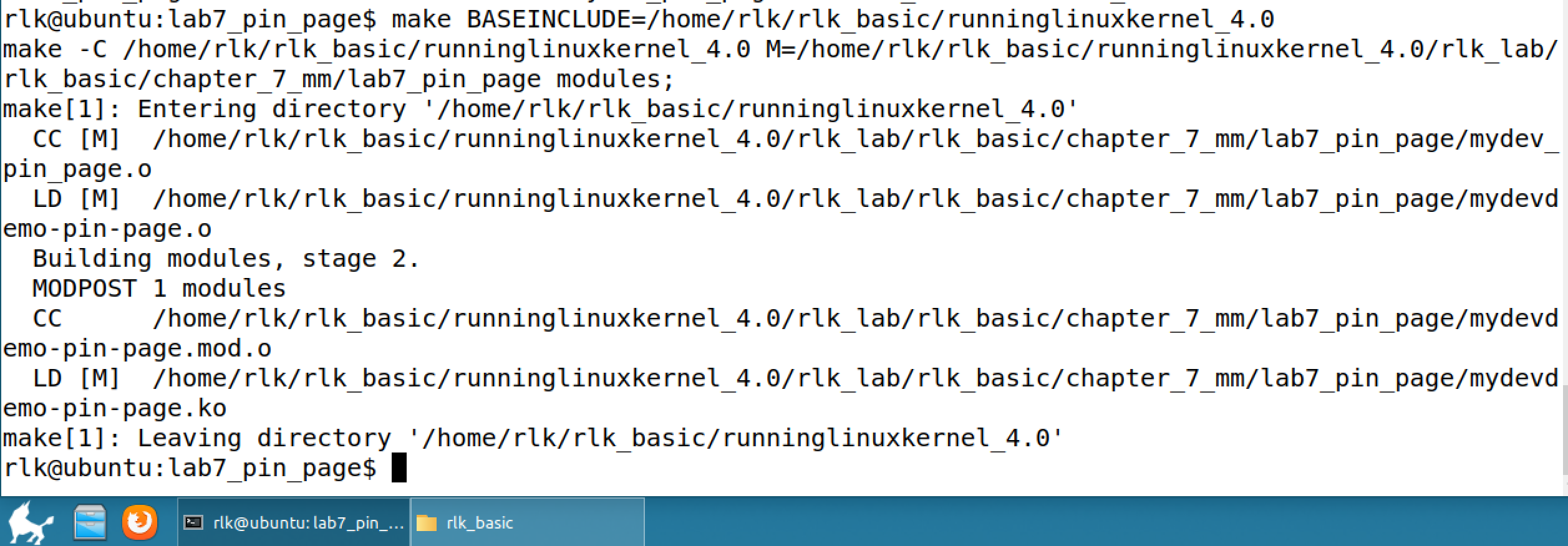
## 3. 内核模块编译和运行举例说明

下面以第7章，lab7为例：

1. 进入runninglinuxkernel\_4.0/rlk\_lab/rlk\_basic/chapter\_7\_mm/lab7\_pin\_page目录。

|  |
| --- |
| # cd /home/rlk/rlk\_basic/runninglinuxkernel\_4.0  # cd rlk\_lab/rlk\_basic/chapter\_7\_mm/lab7\_pin\_page  # export ARCH=arm  # export CROSS\_COMPILE=arm-linux-gnueabi-  # make BASEINCLUDE=你机器上runninglinuxkernel\_4.0的绝对路径 |

如图所示：



注意：这里BASEINCLUDE指定你当前runninglinuxkernel\_4.0的目录路径，请确保路径是对的，这里要使用绝对路径。

另外也可以在把BASEINCLUDE设置到系统的环境变量里。比如修改用户目录下面的.bashrc文件。

|  |
| --- |
| $ vim ~/.bashrc  在这个文件里添加  export BASEINCLUDE=你机器上runninglinuxkernel\_4.0的绝对路径 |

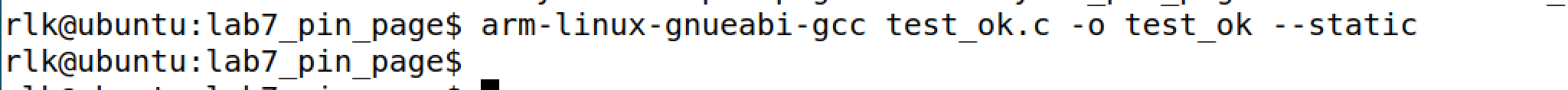
然后在让环境变量生效。

|  |
| --- |
| $ source ~/.bashrc |

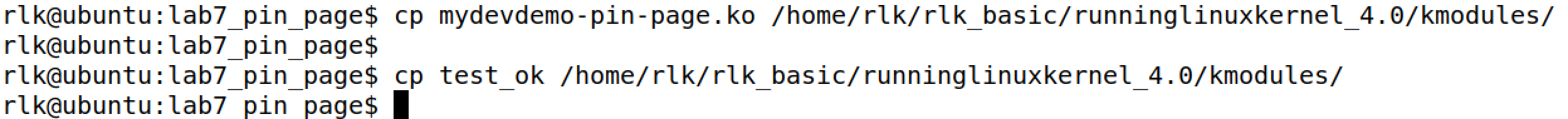
编译test测试app。

|  |
| --- |
| # arm-linux-gnueabi-gcc test\_ok.c -o test\_ok --static |

如图所示：



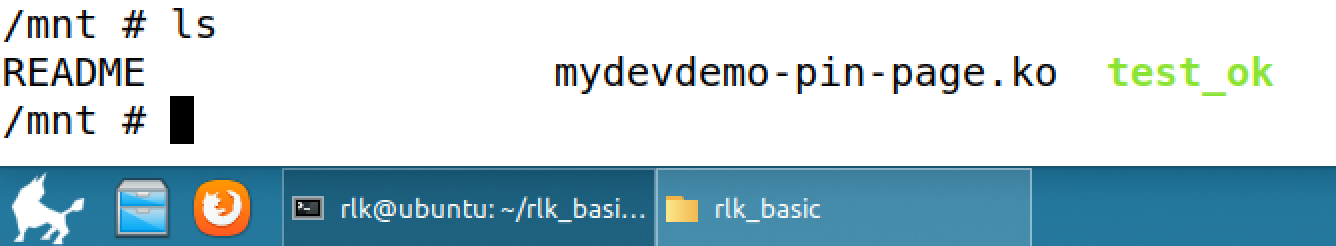
然后把内核模块mydevdemo-pin-page.ko和测试程序test\_ok拷贝到runninglinuxkernel\_4.0/kmodules目录下面。



可以在另一个终端上运行如下脚本启动Qemu。

|  |
| --- |
| #cd runninglinuxkernel\_4.0  # sh run.sh arm32 #启动虚拟机 |

在Qemu虚拟机，在虚拟机的mnt目录就可以看到刚才拷贝的内核模块和应用程序了。



|  |
| --- |
| # 在Qemu虚拟机里面  #cd /mnt  # insmod mydevdemo-pin-page.ko  #./test\_ok |

关闭Qemu虚拟机，可以输入组合键：Ctrl + A，然后快速按下“x”键

## 4.编译运行debian rootfs系统

奔跑吧linux内核支持运行debian系统，这样可以通过apt命令很方便的安装arm32或者arm64的软件，而不需要麻烦的交叉编译了。

1. 编译arm32的内核

|  |
| --- |
| $ cd runninglinuxkernel\_4.0  $ ./run\_debian\_arm32.sh build\_kernel |

1. 编译arm32的debian的rootfs系统

|  |
| --- |
| $sudo ./run\_debian\_arm32.sh build\_rootfs |

1. 运行

|  |
| --- |
| $ ./run\_debian\_arm32.sh run |

如读者需要跑arm64的debian系统，可以使用如下方法来编译和运行。

编译arm64的内核

|  |
| --- |
| $ cd runninglinuxkernel\_4.0  $ ./run\_debian\_arm64.sh build\_kernel |

编译arm64的debian的rootfs系统

|  |
| --- |
| $sudo ./run\_debian\_arm64.sh build\_rootfs |

运行

|  |
| --- |
| $ ./run\_debian\_arm64.sh run |

**debian系统的用户名为root，密码为：123**

系统登录后，先运行apt update命令来更新源，然后就可以使用apt install命令来安装软件包。

若apt update命令遇到问题，可以先ifconfig命令查看eth0网络设备是否在？

另外有可能是系统时间比较旧了，可以使用date命令来设置日期。

root@benshushu:~# date -s 2019-04-25 #假设最新日期是2019年4月25日

Thu Apr 25 00:00:00 UTC 2019

在debian系统中运行《奔跑吧linux内核\*入门篇》的实验，和之前的busybox小系统是类似的，都需要交叉编译内核模块，然后在拷贝到QEMU系统中运行。

请关注奔跑吧Linux社区微信公众号，里面有更多更有料更好玩的Linux干货：



奔跑吧Linux社区的旗舰篇视频节目，帮您打通Linux和职场任督二脉！





笨叔最新力作，全程约5小时高清，140多页ppt，8大实验，基于x86\_64的Centos 7.6和arm64，提供全套实验素材和环境。全面介绍kdump+crash在死机黑屏方面的实战应用，全部案例源自线上云服务器和嵌入式产品开发实际案例！

不能为客户和老板解决死机黑屏问题的都不是合格的linux运维或者研发攻城狮！





订阅视频请到淘宝店：<https://shop115683645.taobao.com>

