

浙江大学

《嵌入式系统》课程报告



题目 从 0 开始制作 U 盘 Linux

姓名

学号

指导老师

教学班

目录

一、项目简介.....	3
二、准备工具.....	3
三、安装步骤.....	3
1、准备开发主机.....	3
2、准备 Linux 内核代码	3
3、编译配置内核源代码.....	4
4、准备 BusyBox 工具.....	4
5、准备 U 盘并配置文件系统	5
6、安装 grub，建立启动文件	5
7、完成系统搭建并测试.....	6
四、遇到的问题及解决方法.....	7
1、内核编译失败并强制退出终端	7
2、grub 无法识别 U 盘内的系统，启动时报“kernel panic”错误。	7
3、VMWare 不支持添加物理磁盘作为虚拟机磁盘	7
五、总结和感悟.....	8
参考资料.....	8

一、项目简介

本项目实现在 U 盘上构建 Linux 操作系统。主要包含以下步骤：

- (1) 准备开发主机；
- (2) 准备 Linux 内核代码；
- (3) 配置编译内核源代码；
- (4) 准备 BusyBox 工具；
- (5) 准备 U 盘并配置文件系统；
- (6) 安装 grub，建立启动文件；
- (7) 完成系统搭建并测试。

二、准备工具

- (1) Windows 11 主机
- (2) Ubuntu 23.10.1
- (3) 128GiB USB3.0 U 盘
- (4) VMware 16.0 PRO
- (5) DisksGenius 软件

三、安装步骤

1、准备开发主机

在 VMWare 16.0 Pro 上安装 Ubuntu 23.10.1 虚拟机。

2、准备 Linux 内核代码

- (1) 进入终端，`sudo su` 授予超级用户权限。
- (2) 安装必要的包和库：

```
apt-get update
apt-get upgrade
apt-get install libncurses5-dev libncursesw5-dev
apt-get install ncurses-devel
apt-get install gcc
apt-get install vim
apt-get install wget
```

以上包中的部分内容，虚拟机系统可能自带，根据实际情况安装即可。

- (3) 在普通用户模式下输入 `cd ~`，进入用户目录，输入：

```
wget https://cdn.kernel.org/pub/linux/kernel/v6.x/linux-6.2.7.tar.xz
```

等待下载结束。

- (4) 输入：

```
xz -d linux-6.2.7.tar.xz
tar xvf linux-6.2.7.tar
cd linux-6.2.7
```

这样一个解压好的源码包就放在了用户目录下。

3、编译配置内核源代码

(1) 进入之前解压源代码的目录，输入 `make x86_64_defconfig` 创建 config 文件。此处可能提示安装未安装的包，依照提示安装即可。

(2) `make menuconfig` 进入图形界面配置选项。

(3) 在图形界面中按如下要求选择：

- 取消 Enable Loadable module support 和 Networking support
- 选择 Device Devices -> Block devices -> loopback device support 和 RAMblock device support
- 选择 Device Devices -> SCSI Support -> SCSI device support -> SCSI disk Support 和 SCSI low-level drivers -> Buslogic SCSI support
- 选择 Device Devices->USB Support 下的 Support for Host-side USB 和 USB Mass Storage support。另外，由于 U 盘为 USB 3.0 接口，还需选择 xHCI HCD (USB 3.0) support。如果需要其他 USB 2.0 设备，如键鼠输入等，还需要选 EHCI HCD (USB 2.0) support。保险起见可以都选上。
- 选择 File System -> ext3（部分旧版内核可能需要 ext2 系统，按需选择即可）
- 保存，退出

(4) `make bzImage` 编译内核。此过程较长，需耐心等待。若要重新修改编译配置选项，需在编译前执行 `make clean` 清除旧文件。

编译成功后提示 Kernel: arch/x86/boot/bzImage is ready

4、准备 BusyBox 工具

(1) 在用户目录下输入：

```
wget https://busybox.net/downloads/busybox-1.36.0.tar.bz2
```

等待 BusyBox 安装包下载完成。下载完成后输入：

```
tar -jxvf busybox-1.36.0.tar.bz2
cd busybox-1.36.0/
```

这样得到解压后的 BusyBox 文件。

(2) 输入：

```
make defconfig
make menuconfig
```

进入图形配置界面。

(3) 在图形界面中选择：

- BusyBox Setting -> Build Options -> Build static binary (no shared libs)
- BusyBox Setting -> Shells -> choose which shell is aliased to 'sh' name (ash)
- BusyBox Setting -> Installation Options -> Don't use /usr（否则 Busybox 会安装到 Ubuntu 的/usr 下，会覆盖原系统原有的命令）

(4) 输入 `make` 编译。

- (5) `make install` 安装。
- (6) `cd _install/` 后执行 `ls` , 可以看到生成了如下文件: `bin linuxrc/sbin` 。

5、准备 U 盘并配置文件系统

- (1) 在用户目录下输入 `mkdir rootfs`, 建立 `rootfs` 文件夹。
- (2) `cp -r {./busybox-1.36.0/_install/*} ./rootfs/` 将上一步 `_install` 目录中的文件复制到 `rootfs` 中。中括号 `{}` 不需要输入, 但其中的内容需根据文件实际所在位置输入。
- (3) `cd rootfs/` 进入 `rootfs`。
- (4) `mkdir usr proc mnt var tmp dev sys etc` 创建文件系统。
- (5) `ln -s bin/sh init` 创建 `init` 文件, 将它做成指向 `bin/sh` 的软链接。
- (6) `cd dev/` 在 `/dev` 目录下添加设备:

```
mknod console c 5 1
mknod null c 1 3
mknod ram b 1 0
mknod tty c 5 0
mknod tty1 c 4 1
mknod tty2 c 4 2
mknod tty3 c 4 3
mknod tty4 c 4 4
chmod 666 tty console null
chmod 666 tty1 tty2
chmod 600 ram
```

- (7) 制作镜像文件。

```
cd .. // 回到 rootfs 目录下
find . | cpio -H newc -o > ../rootfs.img
cd .. //回到用户目录
gzip rootfs.img -f //压缩镜像
```

- (8) 在 U 盘上建立 Linux 分区和 `ext3` 文件格式

在主机 (Windows) 中使用 DiskGenius 软件, 将 U 盘设置为 MBR 格式, 使用 Legacy 模式引导系统启动。

- 分区-> 删除当前分区。有几个分区就执行几次。
- 分区-> 建立新分区-> 选择“主磁盘分区”和“`ext3`”文件系统 (如果上一步内核配置时选择了 `ext2` 文件系统, 此时也需要选择 `ext2` 文件系统)
- 自定义分区大小, 可以使用 512MB 或 1GB

6、安装 grub, 建立启动文件

- (1) 将 U 盘连接到虚拟机中, 执行 `fdisk -l` 可以看到有一个 `/dev/sdb1` 的设备, 大小为前一步分区大小, 类型为 Linux。实际情况可能不为 `/dev/sdb1`, 需根据分区大小和类型找出对应的 U 盘设备, 一般在执行 `fdisk` 指令后终端输出的最后, 后续操作均以 `/dev/sdb1` 为例说明。
- (2) 在用户目录下执行 `mkdir /mnt/usb` 新建一个 `usb` 文件夹, 用于挂载 U 盘。
- (3) `mount /dev/sdb1 /mnt/usb`, 将 U 盘挂载到 `usb` 文件夹。
- (4) 安装 grub: `grub-install --root-directory=/mnt/usb /dev/sdb` //这里的 `sdb`

需根据实际情况填写

(5) 把编译完的内核 bzImage 和压缩好的镜像文件 rootfs.img.gz 复制到 U 盘/boot 目录:

```
cp linux-source-6.2.7/arch/x86/boot/bzImage /mnt/usb/boot/
```

```
cp rootfs.img.gz /mnt/usb/boot/
```

(6) vim /mnt/usb/boot/grub/grub.cfg 编辑 grub.cfg。在 vim 中输入 i, 进入输入模式, 在其中添加如下代码:

```
menuentry "user-linux" {  
  set root='hd0,msdos1'  
  linux /boot/bzImage root=/dev/ram  
  initrd /boot/rootfs.img.gz  
}
```

“user-linux”为 gurb 引导启动时显示的 U 盘 Linux 系统名, 可按需修改。修改完后, 按 ESC 回到常规模式, 使用 :wq 指令保存并退出 vim 编辑器。

(7) umount /mnt/usb 卸载 U 盘。

7、完成系统搭建并测试

(1) 以管理员模式启动 VMWare, 在虚拟机设置中添加硬件设备 (通常为 PhysicalDrive2), 选择使用整个磁盘。

(2) 虚拟机-> 电源-> 打开电源时进入固件, 进入虚拟机 BIOS。

(3) 在 Boot 页面将 U 盘 (通常后置标识为 (0: 1)) 移动到首位。

(4) F10 保存并退出, 启动虚拟机, 进入 gurb 引导界面, 选择制作的 U 盘 Linux 系统, 成功进入。

测试结果如下:

```
Usage: ping [OPTIONS] HOST  
Send ICMP ECHO_REQUESTs to HOST  
  
    -4,-6      Force IP or IPv6 name resolution  
    -c CNT     Send only CNT pings  
    -s SIZE    Send SIZE data bytes in packets (default 56)  
    -i SECS    Interval  
    -A        Ping as soon as reply is received  
    -t TTL     Set TTL  
    -I IFACE/IP Source interface or IP address  
    -W SEC     Seconds to wait for the first response (default 10)  
                (after all -c CNT packets are sent)  
    -w SEC     Seconds until ping exits (default:infinite)  
                (can exit earlier with -c CNT)  
    -q        Quiet, only display output at start/finish  
    -p HEXBYTE Payload pattern  
  
~ # cd  
~ # /usr  
~/bin/sh: /usr: Permission denied  
~ # ls  
bin      etc      linuxrc  proc     sbin     tmp      var  
dev      init     mnt     root     sys      usr  
~ #
```

图 1 测试结果之指令运行结果



图 2 测试结果之指令列表

图 2 中，使用 `/ping` 指令测试系统连通性，`/cd` 指令切换目录，`/ls` 指令列出当前目录下的文件列表。图 3 中，列出了 `/bin` 下支持的操作。可见常见的 Linux 指令均可以正常运行，该系统可以正常使用。

四、遇到的问题及解决方法

1、内核编译失败并强制退出终端

原因：主机内存或进程占用过多，虚拟机配置为双核 4GiB 内存，虚拟机进程被强制关闭。
解决方法：重启主机，释放内存即可。

2、grub 无法识别 U 盘内的系统，启动时报 “kernel panic” 错误。

原因：配置内核时未选择 ext3 文件系统或格式化 U 盘时未设置未 MBR 格式，导致 U 盘内的文件不能被正确识别。
解决方法：按要求重新配置并编译即可。

3、VMWare 不支持添加物理磁盘作为虚拟机磁盘

原因：添加物理磁盘作为虚拟机磁盘需要管理员权限。
解决方法：退出 VMWare，以管理员模式重新启动即可。

五、总结和感悟

本次从 0 开始构建 Linux 系统的作业让我们获益良多。

首先，由于我们都没有 Linux 系统使用的经验，本次作业可以说是第一次接触 Linux 系统，刚开始时由于不习惯 Linux 终端的操作，对 Linux 指令也不熟悉，导致进度缓慢。不过经过了一两天的操作和学习后，结合其他相关课程的学习经验，我们也已经能比较熟练地使用 Linux 系统了。这不仅有利于这次作业的完成，在今后个人的学习和工作生涯中也是一项非常有用的技能，这也是这次作业的意义之一。

其次，由于教材上使用的虚拟机、Linux 内核、Busybox 工具等版本都过于老旧，此次作业选择的内核版本等都比较新，操作流程和教材上有些许区别，这于我们来说意味着需要参考更多的教程和资料来解决。本次实验我们主要参考了 CC98 的经验贴和知乎的教程，也参考了老师的常见报错解决方案，最后成功达到作业要求。通过不断解决问题，我们在实践中深刻理解了计算机组成、系统内核等相关知识，也知道了系统启动时是如何在加载程序的引导下初始化并启动系统的。

最后，本次作业也提升了我们的团队合作能力，小组成员之间及时沟通，相互解答问题，也进一步加深了对这次作业的理解。

参考资料

[1] **【嵌入式作业，无图形界面】U 盘 Linux 教程**, <https://www.cc98.org/topic/5048562>

[2] **制作 U 盘 Linux**, <https://zhuanlan.zhihu.com/p/27009845>

[3] **从 0 开始构建自己的 U 盘 Linux (一)**,
https://blog.csdn.net/weixin_43124455/article/details/89667940