

## 01、根文件系统的制作(busybox)

### 准备工具:

busybox-1.21.1:或选用其他版本的 busybox,大同小异

下载地址: <http://www.busybox.net/>

交叉编译器: 4.3.2

---

### STEP 1: 构建目录结构

创建根文件系统目录, 主要包括以下目录

/dev /etc /lib /usr /var /proc /tmp /home /root /mnt /bin /sbin /opt  
/sys /media

执行:

```
#mkdir $HOME/rootfs
```

```
#cd $HOME/rootfs
```

```
#mkdir dev etc lib usr var proc tmp home root mnt bin sbin opt sys media
```

---

### STEP 2: 使用 busybox 构建/bin /sbin linuxrc:

解压 busubox

进入 busybox 目录, 执行

```
# make defconfig //使用默认配置 busybox,大部分的命令都会被选中
```

```
# make menuconfig
```

Busybox Setting ----->

Build Options ----->

//选择将 busybox 进行静态编译，这样编译出来的 busybox 可以不依赖动态库

//也可以不选，拷贝文件夹的同时也必须拷贝动态库到嵌入式系统

### **[\*]Build BusyBox as a static binary (no shared libs)**

//指定交叉编译器为，否则默认编译为 X86架构

### **(/usr/local/arm/4.3.2/bin/arm-linux-)Cross Compiler prefix**

注：交叉编译器输入错误 **ctrl+Backspace** 删除,在 xshell 下支持复制粘贴

/编译出的 busybox 的 shell 命令解释器支持显示当前路径及主机信息

### **Busybox Library Tuning--->**

#### **[\*]Username completion**

#### **[\*]Fancy shell prompts**

#### **[\*]Query cursor position from terminal**

保存退出

**# make**

**# make install**

在 busybox 目录下会看见 \_install 目录，里面有 bin sbin usr linuxrc 四个文件

将这三个目录或文件拷到第一步所建的 rootfs 文件夹下。

**#cp bin sbin usr linuxrc \$HOME/rootfs -rfa**

注：-a 的参数必不可少，因为 bin sbin 里面的命令大多是到 busybox 的动态链接,如果不加-a 则拷贝过程中会自动将每个命令复制一份

---

### **STEP3 构建 lib 目录:**

进入 /usr/local/arm/4.3.2/arm-none-linux-gnueabi/libc/armv4t/lib 目录

将以下动态库拷贝到\$HOME/rootfs/lib 下

```
#cp *so* $HOME/rootfs/lib -a
```

进入以下目录/usr/local/arm/4.3.2/arm-none-linux-gnueabi/libc/armv4t/usr/lib

将以下动态库拷贝到\$HOME/rootfs/lib 下

```
#cp ./libstdc++.so.* $HOME/rootfs/lib -a
```

---

## STEP 4 构建 etc 目录:

### 1)etc 目录下最重要的四个文件

inittab: 用来作为 linuxrc 的配置脚本

init.d/rcS: inittab 启动的第一个脚本

一般用来挂载系统必需的文件系统、必要的设备连接、设置 IP 地址、启动其他脚本等  
默认仅有 mount -a

fstab: 执行 mount -a 时, 按照此文件挂载文件系统

profile: 登陆完 shell 后自动执行此脚本, 一般用来配置用户的登录环境

### 2)进入根文件系统 rootfs 的 etc 目录, 执行如下操作:

拷贝 Busybox 目录下 examples/bootfloopy/etc/\* 到当前目录下

以此为模板构建我们的根文件系统

```
# cp examples/bootfloopy/etc/* $HOME/rootfs/etc -r
```

### 3)修改 inittab

inittab 的作用及格式请参考其他说明

删除第三行代码: tty2::askfirst:-/bin/sh

#### 4)修改 etc/init.d/rcS 文件

init 进程通过 inittab 启动的第一个进程为此脚本，我们需要做以下修改

```
#!/bin/sh
```

#1.根据 fstab 配置文件挂载相应的文件系统

```
/bin/mount -a
```

#2.配置 telnetd 服务器的时候需要以下文件夹及文件系统

```
/bin/mkdir -p /dev/pts
```

```
/bin/mkdir -p /dev/shm
```

```
/bin/mount -t devpts devpts /dev/pts
```

```
/bin/mount -t tmpfs tmpfs /dev/shm
```

#3.启动需要以下文件夹

```
/bin/mkdir /var/tmp
```

```
/bin/mkdir /var/modules
```

```
/bin/mkdir /var/run
```

```
/bin/mkdir /var/log
```

#4.在 lib 下创建相应文件夹（与当前内核版本相同），为驱动模块加载作准备

```
if ! [ -d /lib/modules ]; then
```

```
    mkdir /lib/modules
```

```
fi
```

```
if ! [ -d /lib/modules/$(uname -r) ]; then
```

```
    mkdir /lib/modules/$(uname -r)
```

```
fi
```

#5.自动在/dev 下创建设备节点

```
/sbin/mdev -s
```

5)修改 `etc/fstab` 文件，填充以下内容

#device	mount-point	type	options	dump	fask order
proc	/proc	proc	defaults	0	0
ramfs	/var	ramfs	defaults	0	0
ramfs	/tmp	ramfs	defaults	0	0
none	/sys	sysfs	defaults	0	0
none	/dev	ramfs	defaults	0	0

6)修改 `profile`

#1.初始化 `PATH` 环境变量及动态链接库路径

```
export PATH=$PATH:/bin:/sbin:/usr/bin:/usr/sbin
```

```
export LD_LIBRARY_PATH=$LD_LIBRARY_PATH:/lib:/usr/lib
```

#2.初始化主机名以及当前工作路径

```
/bin/hostname sunplusedu
```

```
USER="`id -un`"
```

```
LOGNAME=$USER
```

```
HOSTNAME='/bin/hostname'
```

```
PS1='[\u@\h \W]# '
```

7)用户名、密码文件构建(这步可以不做，一般开发板都不带密码)

拷贝虚拟机上的`/etc/passwd`, `/etc/group`, `/etc/shadow` 到 `rootfs/etc` 下。

```
#cp /etc/passwd $HOME/rootfs/etc
```

```
#cp /etc/group $HOME/rootfs/etc
```

```
#cp /etc/shadow $HOME/rootfs/etc
```

对以下三个文件修改，只保存与 root 相关的项(每个文件的第一行)

只保留 passwd 第一行，为：

```
root:x:0:0:root:/root:/bin/sh
```

（而且最后改成/bin/ash，PC 使用 bash 作为默认 shell，嵌入式系统不支持 bash，这里改为 sh 或 ash）

只保留 group 第一行，为：

```
root:x:0:root
```

只保留 shadow 第一行，为：

```
root:$1$x9yv1WIB$abJ2v9jOIoc9xW/y0QwPs.:14034:0:99999:7:::
```

（注意这个值不一定一样，各人密码不一样，这个值也不一样，这个的密码是六个 1：111111）这个文件在本地登录或者 ftp、telnet 登录时起作用

## 打开/etc/inittab

把第二行：::respawn:-/bin/sh

改为::respawn:-/bin/login //登录时需要用户名、密码验证

这样，以后登陆开发板时需输入用户名密码，同虚拟机相同

登陆后可以通过 passwd 命令修改密码或通过 adduser 增加新用户

## STEP 5 其它目录：

其他目录保持默认即可

至此，根文件系统制作完毕