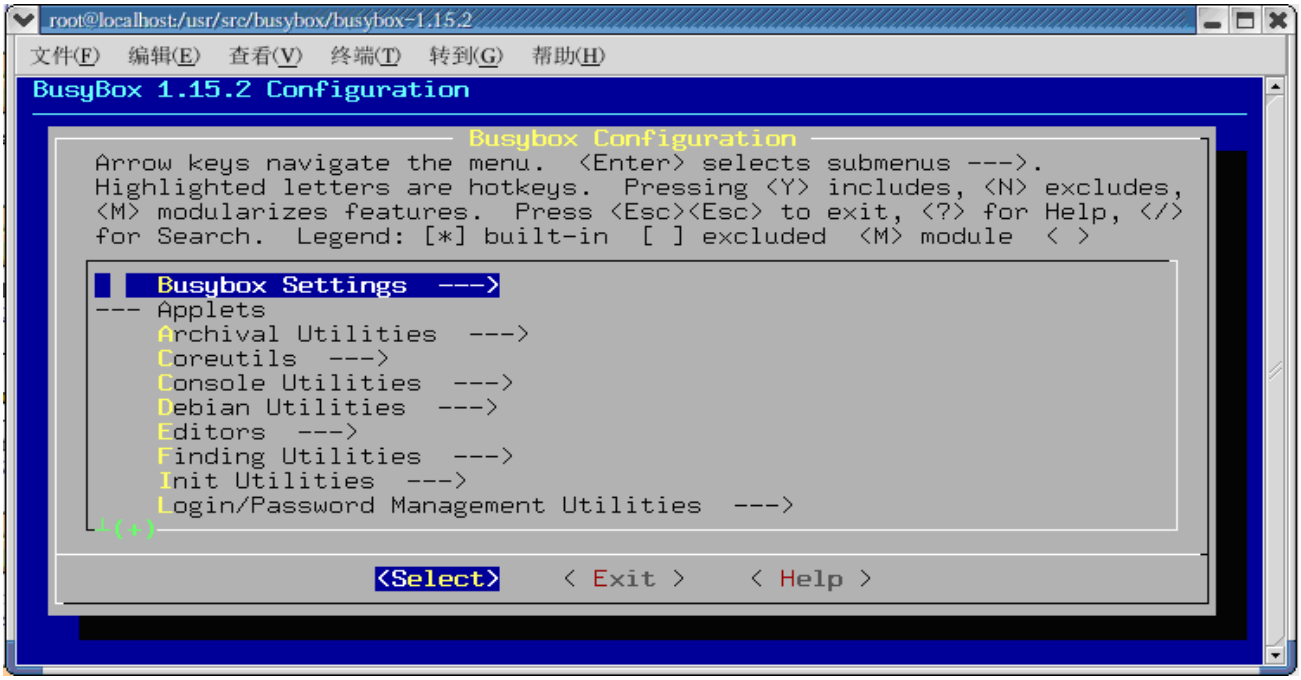
用 Busybox 制作根文件系统

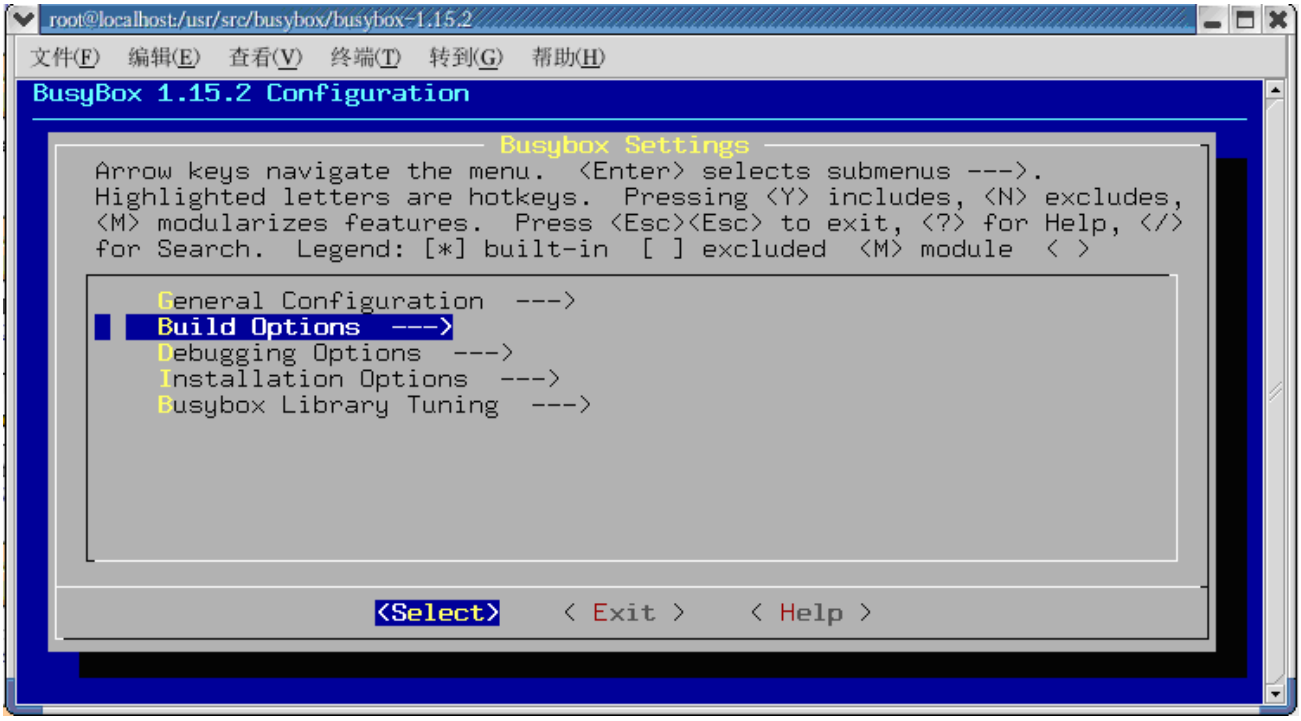
本文描述了如何利用用 Busybox 制作根文件系统
安装并设置交叉编译工具链 arm-none-linux-gnueabi
下载 busybox 压缩包 busybox-1.15.2.tar.bz2 以及 7 个附加补丁
busybox-1.15.2-ash.patch
busybox-1.15.2-awk.patch
busybox-1.15.2-buildsys.patch
busybox-1.15.2-flash.patch
busybox-1.15.2-grep.patch
busybox-1.15.2-ping.patch
busybox-1.15.2-split.patch
解压缩 busybox-1.15.2.tar.bz2 后，生成 busybox-1.15.2 文件夹，将 7 个补丁移动至目录下。
创建一个补丁脚本 patch-sh,内容如下：

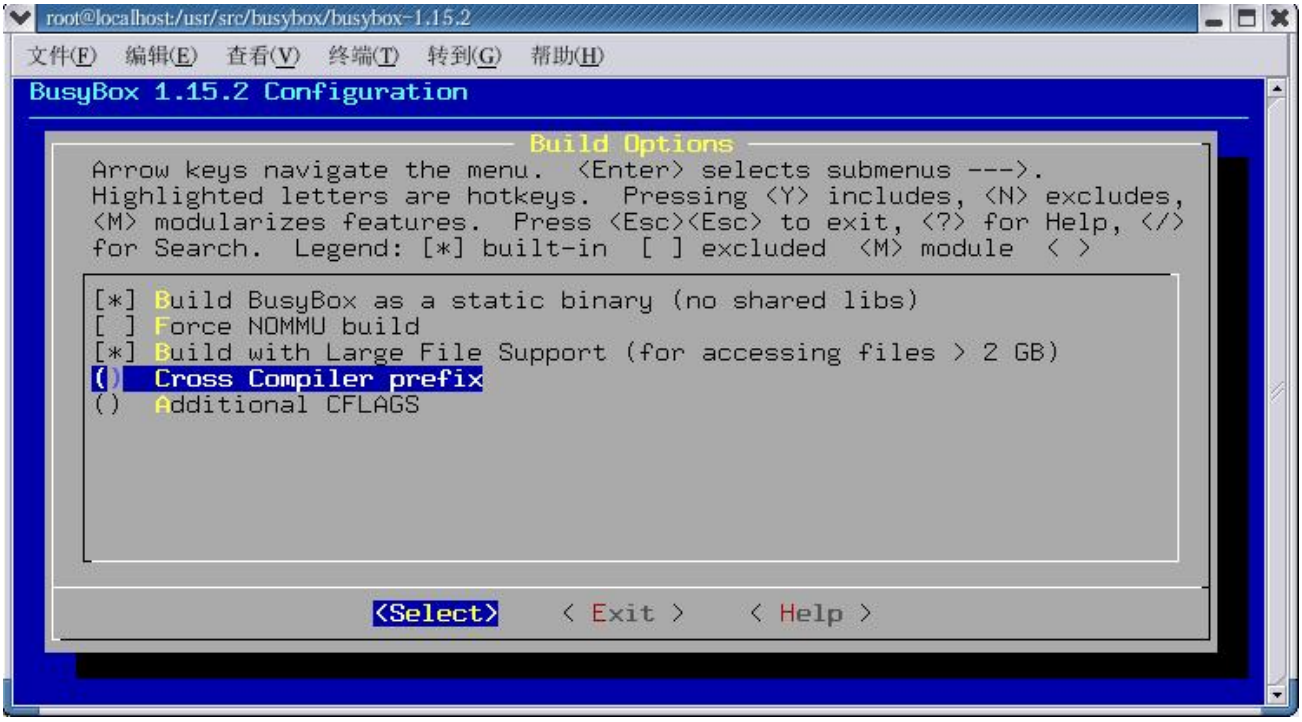
```
################################################
#!/bin/sh
cat busybox-1.15.2-ash.patch | patch –p1
cat busybox-1.15.2- awk.patch | patch –p1
cat busybox-1.15.2- buildsys.patch | patch –p1
cat busybox-1.15.2- flash.patch | patch –p1
cat busybox-1.15.2- grep.patch | patch –p1
cat busybox-1.15.2- ping.patch | patch –p1
cat busybox-1.15.2- split.patch | patch –p1
################################################
```

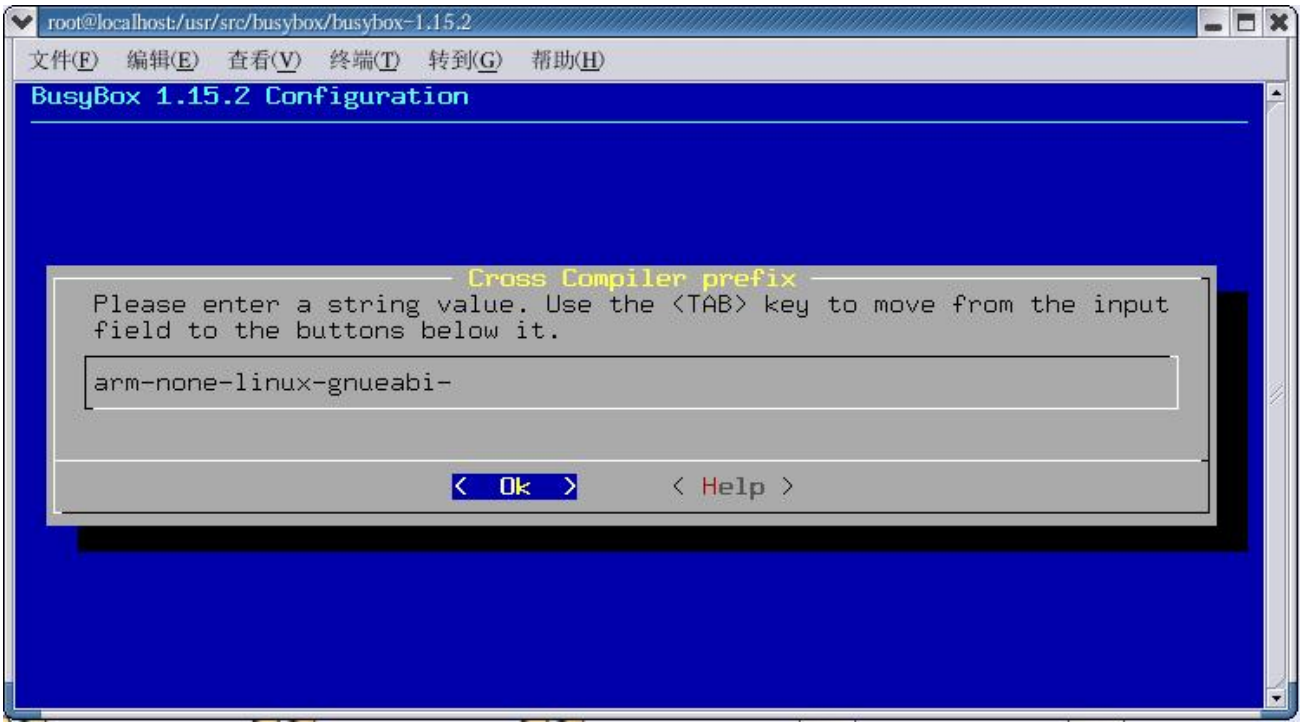
运行脚本文件给 busybox 打入补丁
>./patch-sh

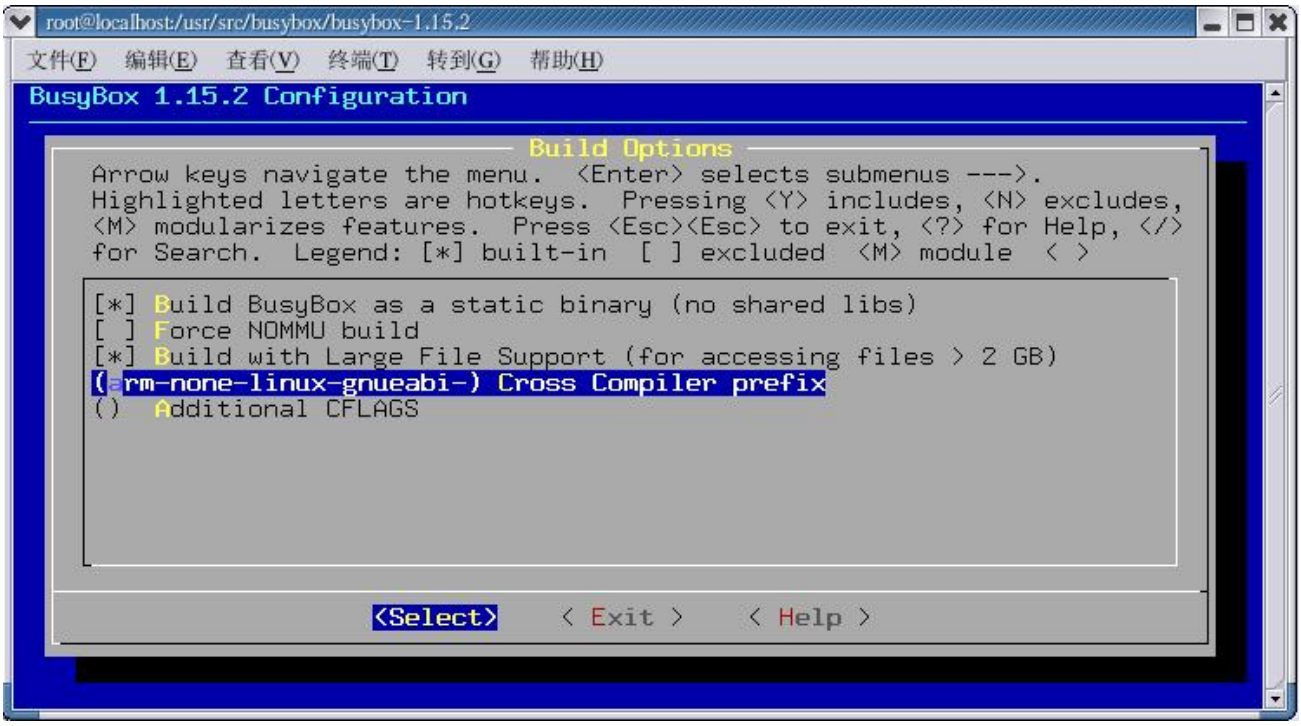
开始配置 Busybox
>make menuconfig
设置

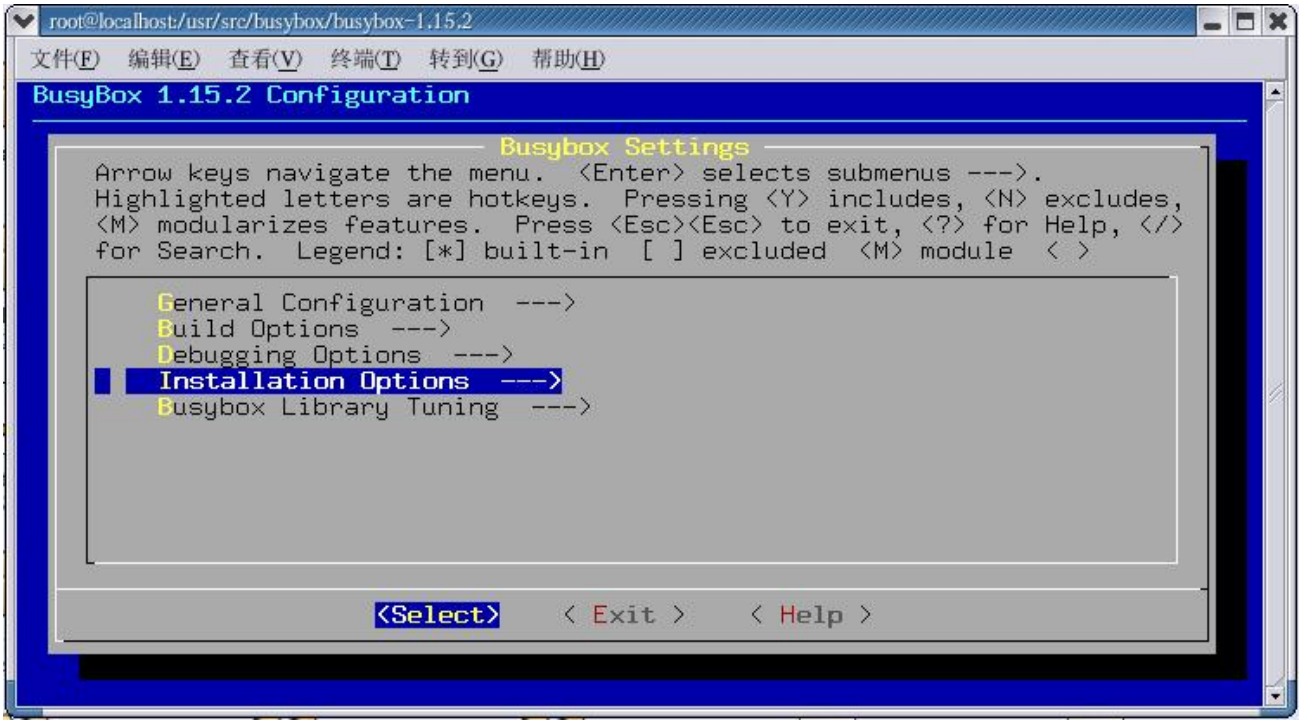

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Configuration
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
Busybox Settings --->
--- Applets
Archival Utilities --->
Coreutils --->
Console Utilities --->
Debian Utilities --->
Editors --->
Finding Utilities --->
Init Utilities --->
Login/Password Management Utilities --->
<Select> < Exit > < Help >

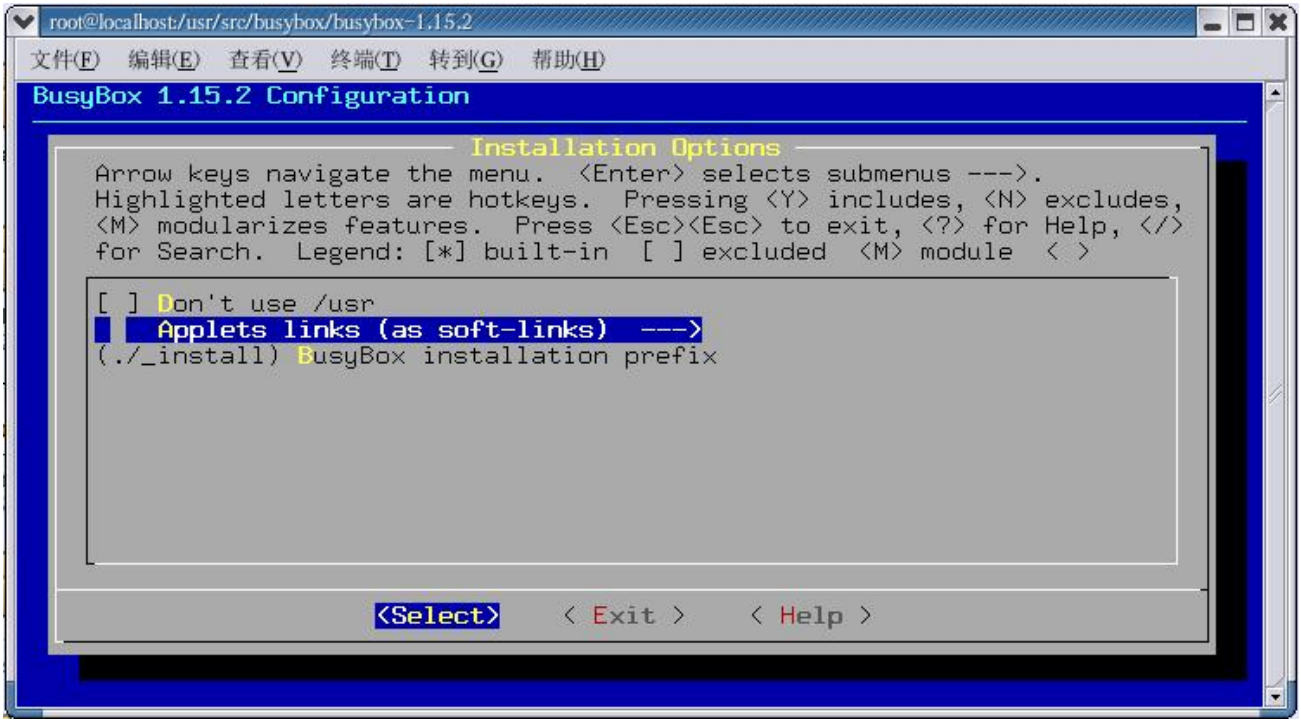

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Settings
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
General Configuration --->
Build Options --->
Debugging Options --->
Installation Options --->
Busybox Library Tuning --->
<Select> < Exit > < Help >

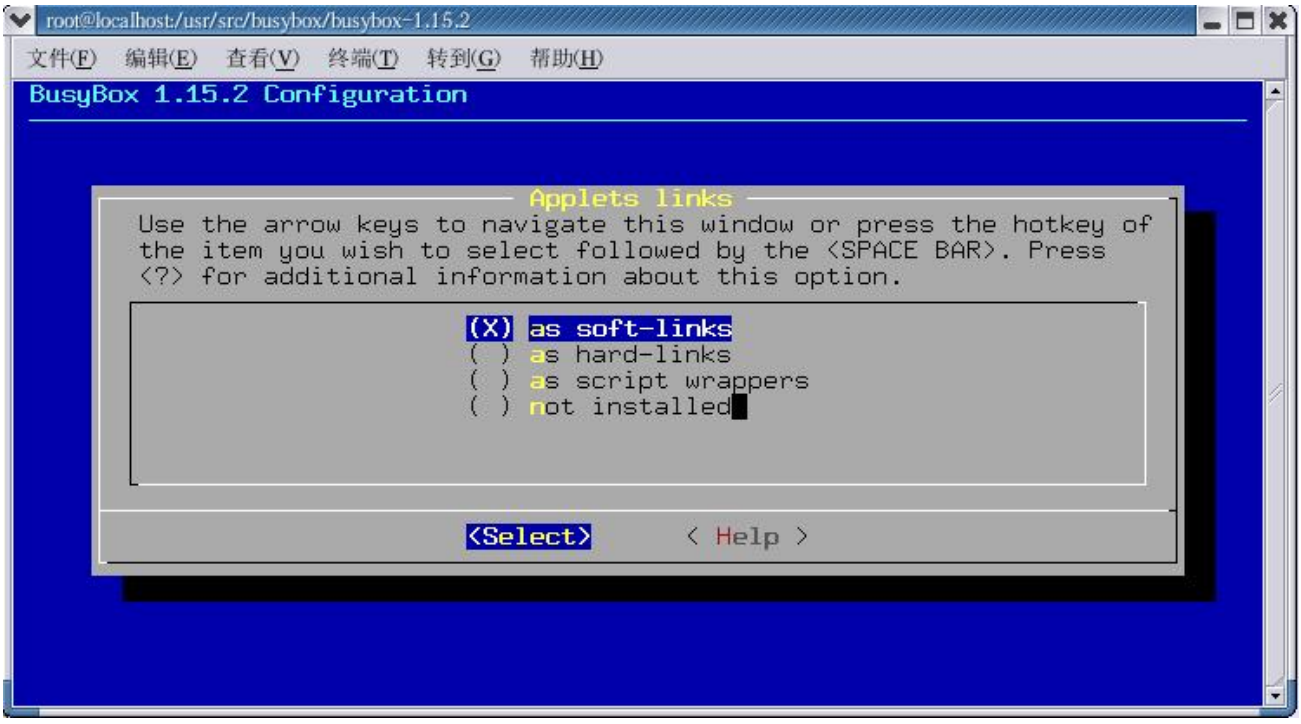

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Build Options
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
[*] Build BusyBox as a static binary (no shared libs)
[ ] Force NOMMU build
[*] Build with Large File Support (for accessing files > 2 GB)
() Cross Compiler prefix
() Additional CFLAGS
<Select> < Exit > < Help >

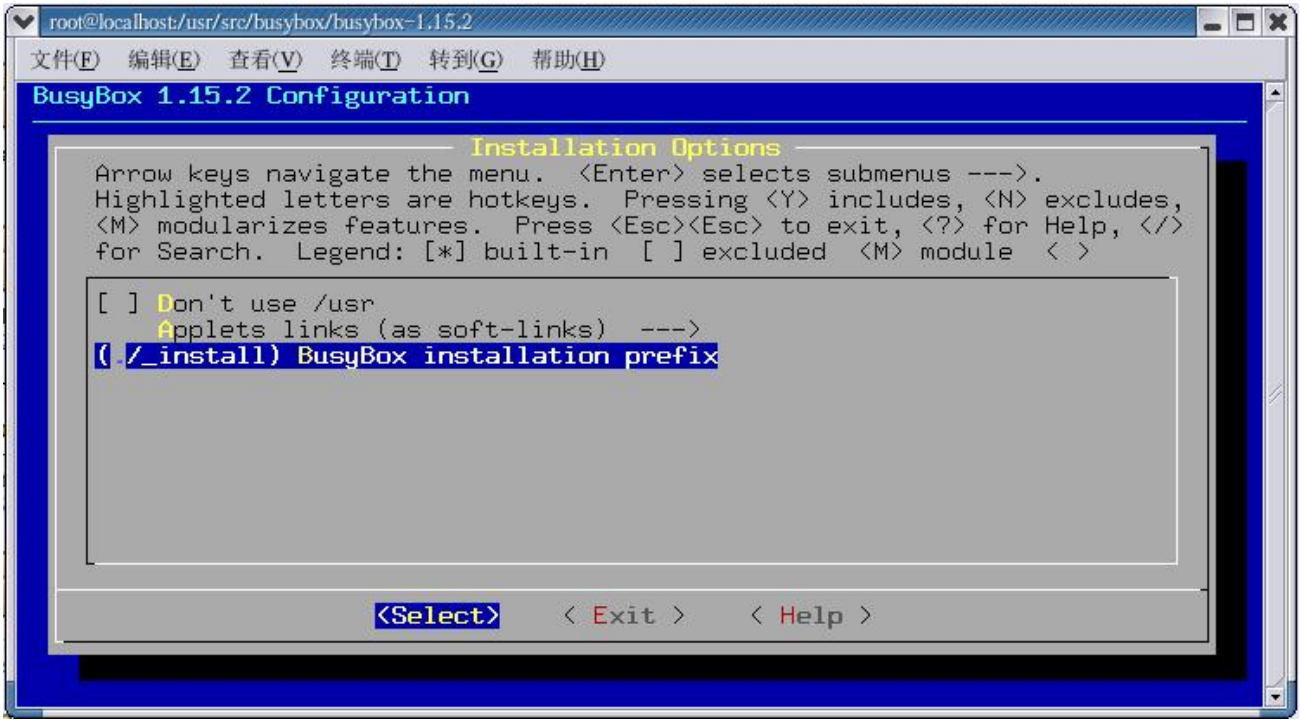

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Cross Compiler prefix
Please enter a string value. Use the <TAB> key to move from the input
field to the buttons below it.
arm-none-linux-gnueabi-
< Ok > < Help >

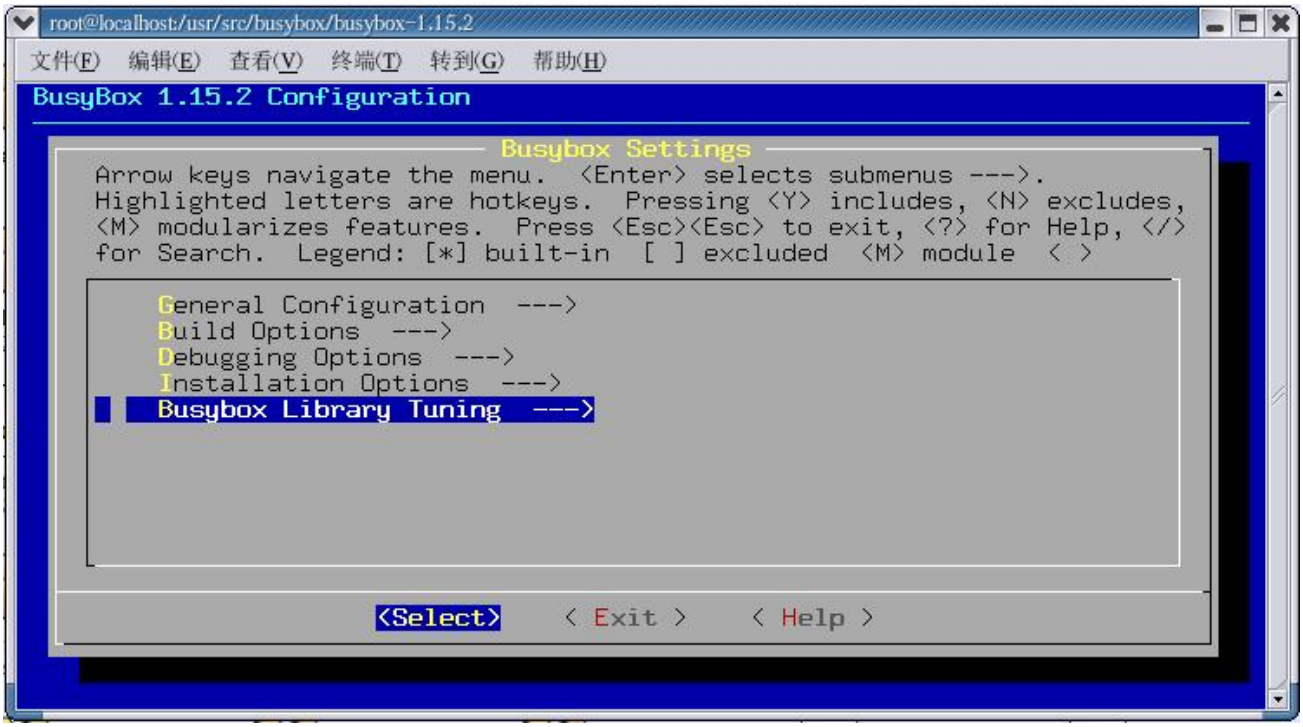

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Build Options
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
[*] Build BusyBox as a static binary (no shared libs)
[ ] Force NOMMU build
[*] Build with Large File Support (for accessing files > 2 GB)
(arm-none-linux-gnueabi-) Cross Compiler prefix
() Additional CFLAGS
<Select> < Exit > < Help >

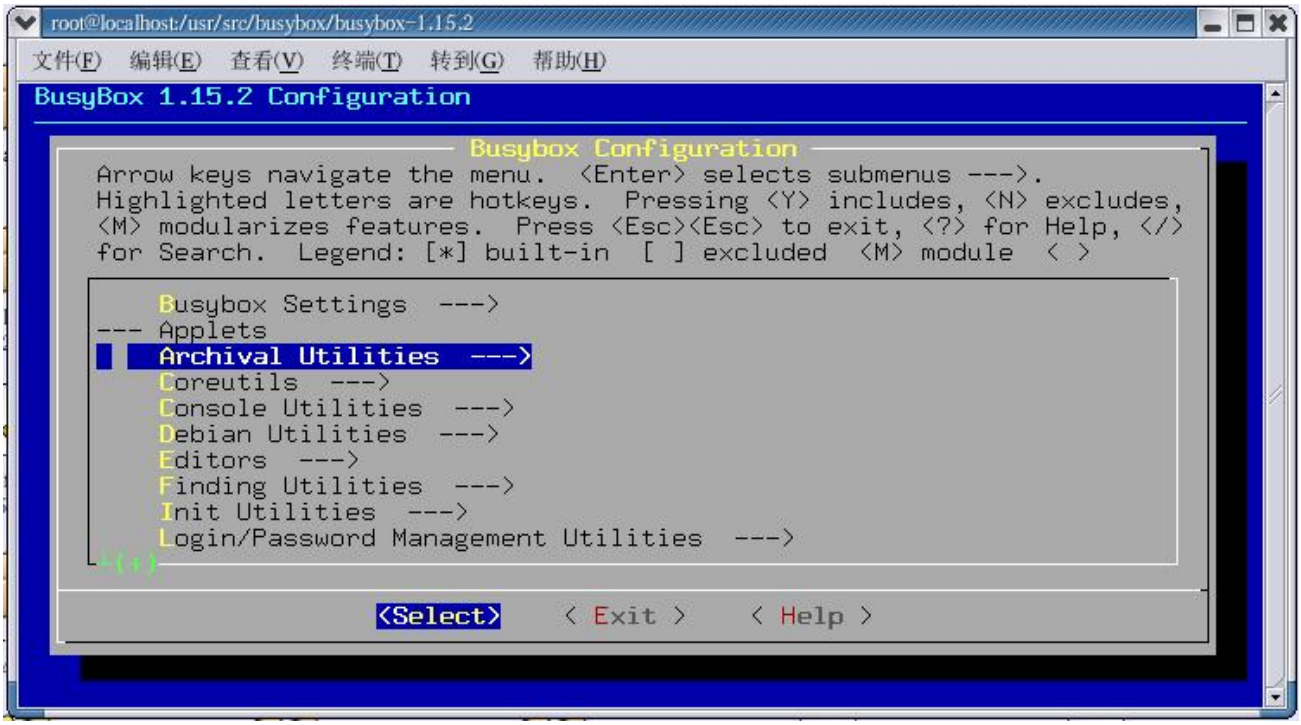

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Settings
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
General Configuration --->
Build Options --->
Debugging Options --->
Installation Options --->
Busybox Library Tuning --->
<Select> < Exit > < Help >

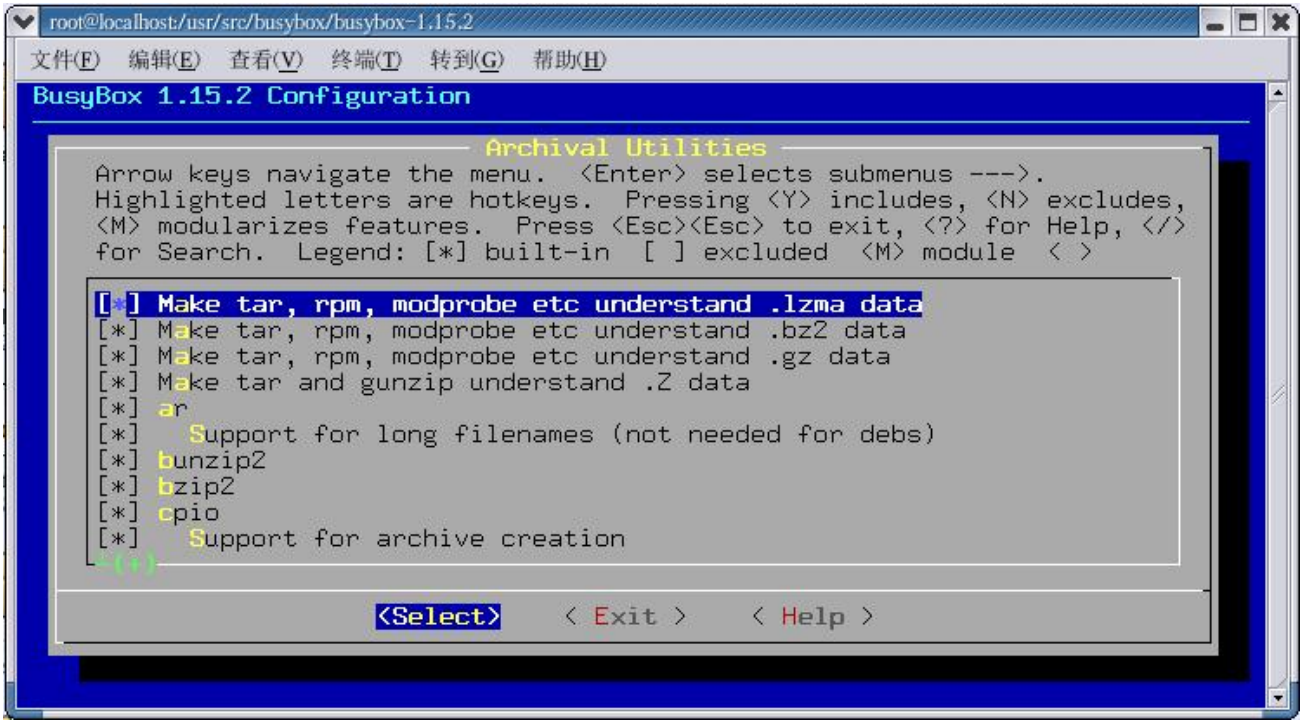

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Installation Options
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
[ ] Don't use /usr
Applets links (as soft-links) --->
(./_install) BusyBox installation prefix
<Select> < Exit > < Help >

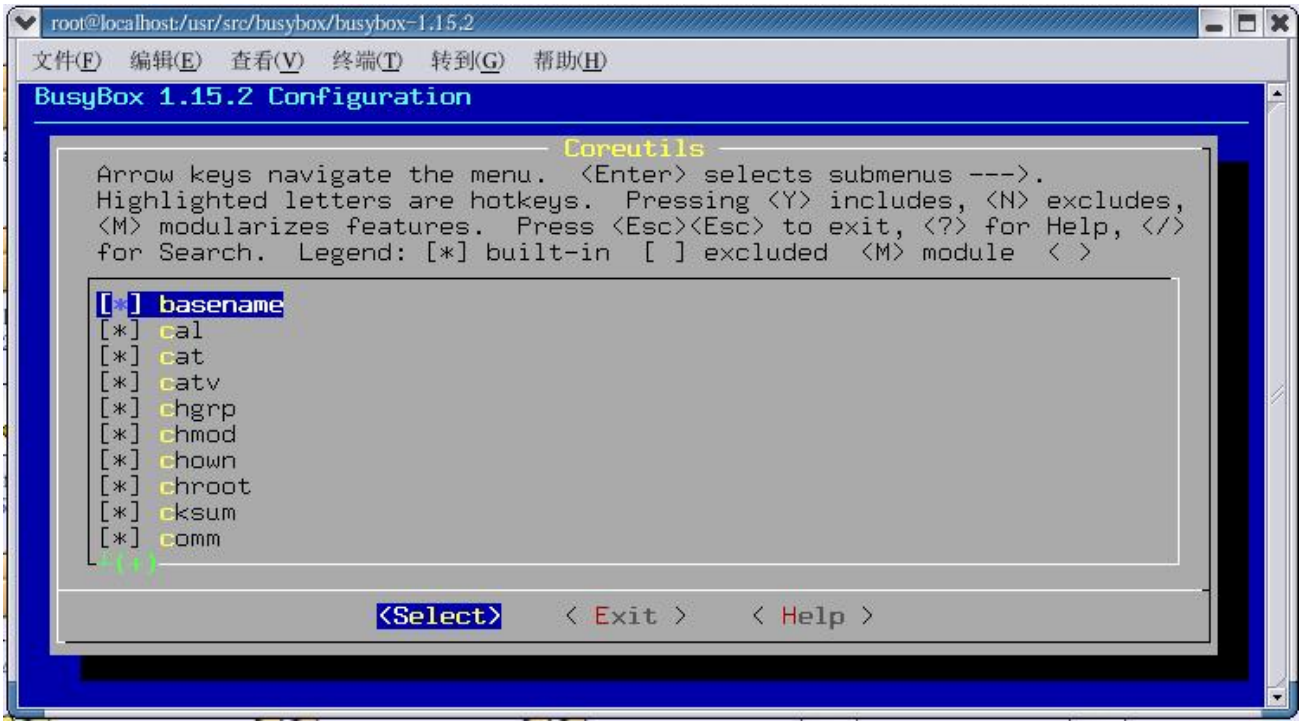

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Applets links
Use the arrow keys to navigate this window or press the hotkey of
the item you wish to select followed by the <SPACE BAR>. Press
<?> for additional information about this option.
(X) as soft-links
( ) as hard-links
( ) as script wrappers
( ) not installed
<Select> < Help >

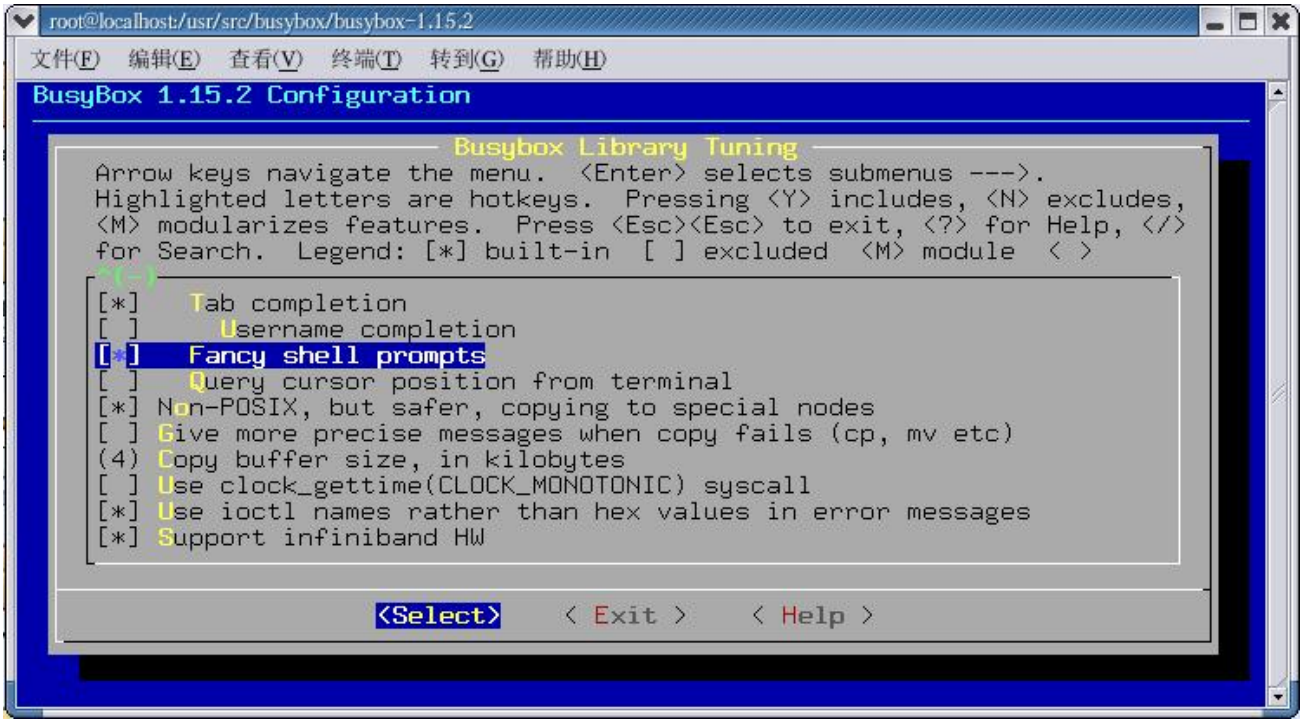

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Installation Options
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
[ ] Don't use /usr
Applets links (as soft-links) --->
(./_install) BusyBox installation prefix
<Select> < Exit > < Help >

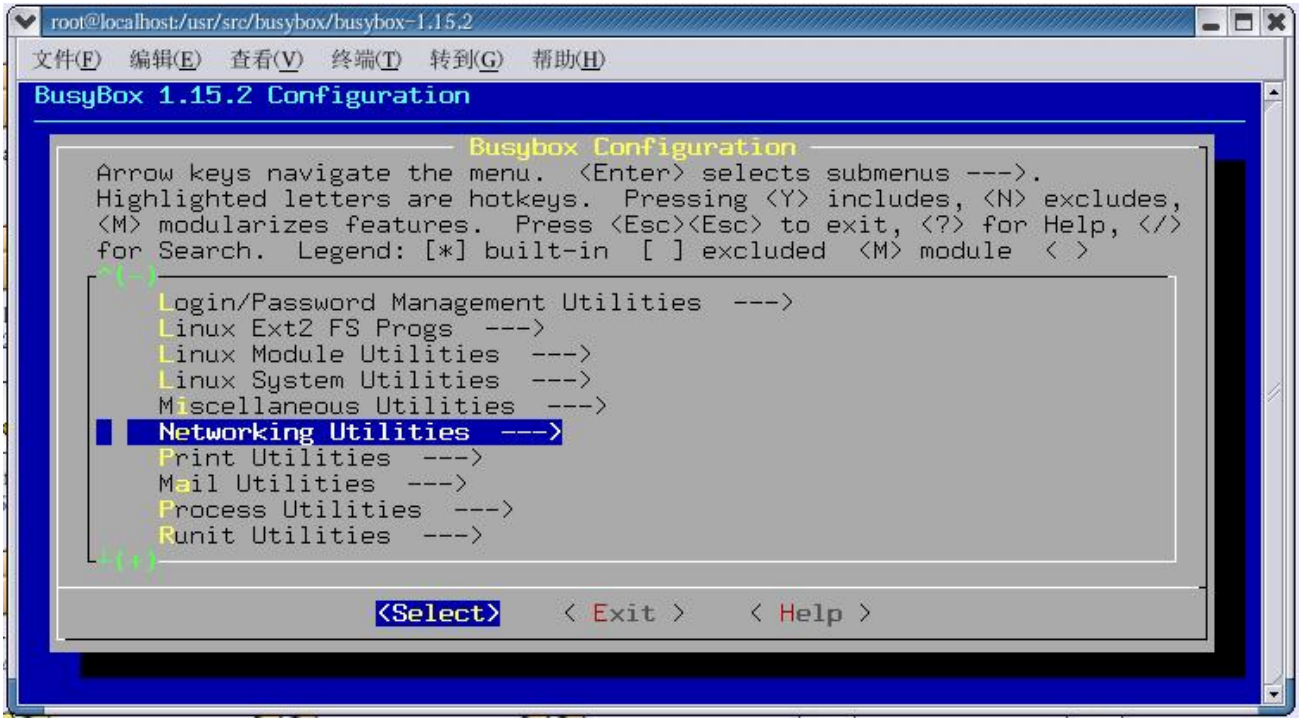

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Settings
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
General Configuration --->
Build Options --->
Debugging Options --->
Installation Options --->
Busybox Library Tuning --->
<Select> < Exit > < Help >

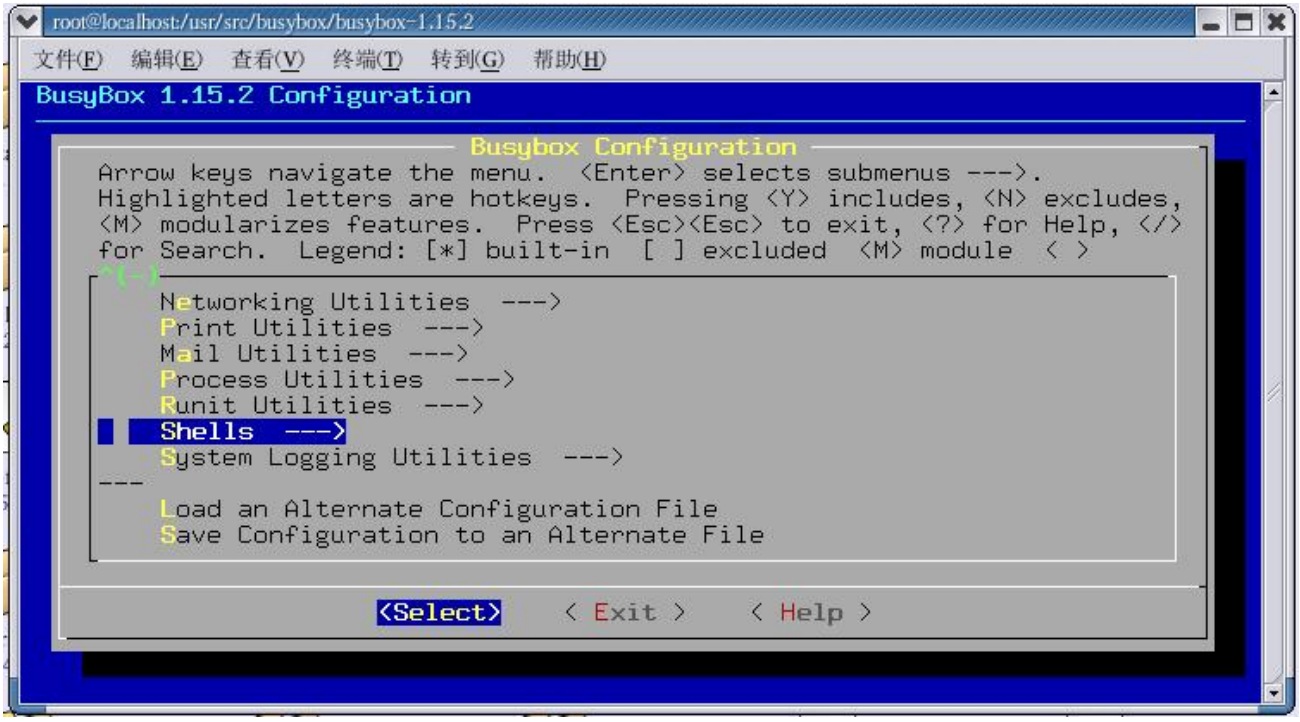

root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Configuration
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
Busybox Settings --->
--- Applets
Archival Utilities --->
Coreutils --->
Console Utilities --->
Debian Utilities --->
Editors --->
Finding Utilities --->
Init Utilities --->
Login/Password Management Utilities --->
<Select> < Exit > < Help >


root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Archival Utilities
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
[*] Make tar, rpm, modprobe etc understand .lzma data
[*] Make tar, rpm, modprobe etc understand .bz2 data
[*] Make tar, rpm, modprobe etc understand .gz data
[*] Make tar and gunzip understand .Z data
[*] ar
[*] Support for long filenames (not needed for debs)
[*] bunzip2
[*] bzip2
[*] cpio
[*] Support for archive creation
<Select> < Exit > < Help >


root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Coreutils
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
[*] basename
[*] cal
[*] cat
[*] catv
[*] chgrp
[*] chmod
[*] chown
[*] chroot
[*] cksum
[*] comm
<Select> < Exit > < Help >


root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Library Tuning
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
[*] Tab completion
[ ] Username completion
[*] Fancy shell prompts
[ ] Query cursor position from terminal
[*] Non-POSIX, but safer, copying to special nodes
[ ] Give more precise messages when copy fails (cp, mv etc)
(4) Copy buffer size, in kilobytes
[ ] Use clock_gettime(CLOCK_MONOTONIC) syscall
[*] Use ioctl names rather than hex values in error messages
[*] Support infiniband HW
<Select> < Exit > < Help >


root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Configuration
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
Login/Password Management Utilities --->
Linux Ext2 FS Progs --->
Linux Module Utilities --->
Linux System Utilities --->
Miscellaneous Utilities --->
Networking Utilities --->
Print Utilities --->
Mail Utilities --->
Process Utilities --->
Runit Utilities --->
<Select> < Exit > < Help >


root@localhost:/usr/src/busybox/busybox-1.15.2
文件(F) 编辑(E) 查看(V) 终端(T) 转到(G) 帮助(H)
BusyBox 1.15.2 Configuration
Busybox Configuration
Arrow keys navigate the menu. <Enter> selects submenus --->.
Highlighted letters are hotkeys. Pressing <Y> includes, <N> excludes,
<M> modularizes features. Press <Esc><Esc> to exit, <?> for Help, </>
for Search. Legend: [*] built-in [ ] excluded <M> module < >
Networking Utilities --->
Print Utilities --->
Mail Utilities --->
Process Utilities --->
Runit Utilities --->
Shells --->
System Logging Utilities --->

Load an Alternate Configuration File
Save Configuration to an Alternate File
<Select> < Exit > < Help >

配置好上述所有选项后就可以开始进行编译了，在命令行提示下输入：make
>make
编译结束后再执行安装命令。
>make install
安装结束后会在 busybox 目录下生成_install 文件夹，内部就是编译好的 busybox 二进制可执行文件以及命令软链接。

建立根文件系统的目录结构
在 linux 环境 NFS 服务目录中建立一个文件夹：
>mkdir /home/target
>cd /home/target
>mkdir bin dev etc lib home proc mnt sbin var sys usr
复制 busybox 生成的_install 文件夹下所有内容到 target 目录下
>cp –a _install/* /home/target
复制交叉编译器所带的动态库到 target/lib 目录下，这样做的目的是，在利用交叉编译器编译应用程序时可能会用动态链接方式进行编译，应用程序在运行的时候就必须要加载这些动态库，所以根文件系统 lib 目录中必须要存在这些库，使程序运行时能正确链接到这些动态库。
>cp –a /opt/arm-none-linux-gnueabi/libc/armv4t/lib/* /target/lib

建立相关文件
删除 target 目录下的 linuxrc 文件
创建为 shell 导入全局变量的/etc/profile 文件：

```
#############################################
#!/bin/sh
export LD_LIBRARY_PATH=/lib:/usr/lib

HOSTNAME=’/bin/hostname’
export HOSTNAME

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

alias ll=”ls –al”

USER=” ‘id –un’ “
LOGNAME = $ USER
PS1=’[\u@\h \W]\$’

echo “---End Of    Profile---”
#############################################
```

创建初始化文件/etc/inittab

```
##############################################
::sysinit:/etc/ini.d/rcS
#::respawn:-/bin/sh
#::respawn:/sbin/getty 115200 ttyS0 vt100 -n -l /bin/sh

::respawn:/sbin/getty 115200 ttyS0 vt100

::restart:/sbin/init
::ctrlaltdel:/sbin/reboot
::shutdown:/bin/umount -a -r
::shutdown:/sbin/swapoff –a
##############################################
```

创建/etc/fstab 文件

```
#############################################################################
devpts                  /dev/pts                devpts    gid=5,mode=620    0 0
proc                    /proc                   proc      defaults          0 0
usbfs                   /proc/bus/usb           usbfs     defaults          0 0
tmpfs                   /var/volatile           tmpfs     size=1M           0 0
#############################################################################
```

创建初始化脚本文件/etc/init.d/rcS 文件

```
##############################################
#!/bin/sh
runlevel=S
export runlevel
#host name
/bin/hostname taihumicro
[ -d "/proc/1" ] || mount /proc

echo "---mount all---
/bin/mount -av

#mdev setting
echo "---mdev setting---"
/bin/mount -t sysfs sysfs /sys
/bin/echo /sbin/mdev > /proc/sys/kernel/hotplug
mdev -s

#local server
/etc/init.d/rc.local
echo    "---welcome---"
##############################################
```

创建 mdev 配置文件/etc/mdev.conf。
>touch /etc/mdev.conf

创建 name server 配置文件/etc/resolv.conf
```
##########################################
nameserver 192.168.1.1
##########################################
```

创建/etc/host.conf
```
##########################################
order hosts,bind
##########################################
```

创建/etc/hosts
```
##########################################
127.0.0.1        localhost:localdomain        localhost
##########################################
```

创建网络应用程序用到的标准服务端口映射表/etc/services
```
##########################################
tcpmux 1/tcp
tcpmux 1/udp
ftp-data      20/tcp
ftp    21/tcp
ssh    22/tcp
ssh    22/udp
telnet 23/tcp
nameserver      42/tcp name
syslog    514/udp
##########################################
```

创建库相关文件/etc/ld.so.conf
```
##########################################
/lib
/usr/lib
##########################################
```

创建组和用户相关文件/etc/passwd
```
##########################################
root::0:0:root:/home/root:/bin/sh
nobody:*:65534:65534:nobody:/nonexisten:/bin/sh

#/etc/group
root:*:0:
```

nobody:*:65534:
##

为 root 用户建立目录
>cd target/home
>mkdir root

复制设备节点，将 Linux 虚拟机中/dev 目录下的一些节点复制到根文件系统 target/dev/目录下。
cp –a /dev/console /home/target/dev
cp –a /dev/null /home/target/dev

测试根文件系统
为了验证根文件系统创建是否成功，需要用 Linux 来启动该文件系统。为了方便起见可以先用 NFS 方式来挂载根文件系统。
在 Uboot 提示符下设置启动参数：
U-Boot>setenv bootargs mem=64M console=ttyS0 115200 root=/dev/nfs nfsroot=192.168.1.10:/home/target ip=192.168.1.2:192.168.1.10:192.168.1.10:255.255.255.0::eth0:off
然后保存参数:
U-Boot>saveenv

其中nfsroot=192.168.1.10:/home/target表示nfs 将从192.168.1.10:/home/target 文件夹内启动根文件系统
ip=192.168.0.2:192.168.0.10:192.168.0.10:255.255.255.0::eth0:off表示
ip=$(taget IP):$(servicer IP):$(netGate):$(netMask)::(device):off
重新启动目标板，就能从nfs 根文件系统启动了。