

附录一 QT 第三方库文件的编译

解压光盘资料里面的 linux 目录下的 3rdpart-lib-for-Qtopia2.2.0.tar.gz 后，该目录有如下的几个文件：

arminclude：编译 Qtopia2.2.0 需要的头文件（迅为制作）

armlib：编译 Qtopia2.2.0 需要的库文件（迅为制作）

jpeg-6b：一个 jpeg 图形编码解码程序库

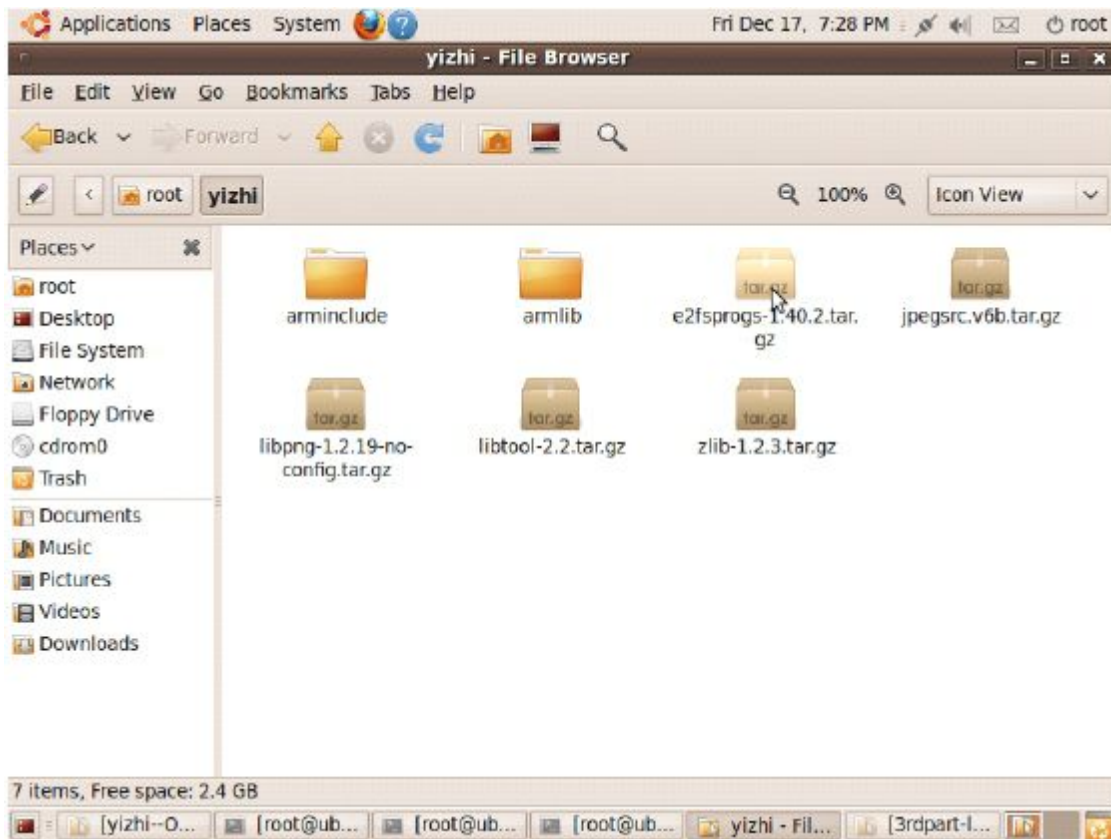
libpng-1.2.19：一个 png 图形编码解码程序库

zlib-1.2.3：一个压缩解压程序库

e2fsprogs-1.40.2：提供 uuid 支持

libtool-2.2.tar.gz 工具包，编译 jpeg-6b 过程中需要该工具

把 jpeg-6b，libpng-1.2.19，zlib-1.2.3，e2fsprogs-1.40.2 压缩包拷贝到 Ubuntu 的 root/yizhi 文件夹下面。如果您的 PC 没有 yizhi 目录，自己先建立一个，同时需要在 yizhi 文件夹下面建立 arminclude 和 armlib 文件夹，用来存放您自己制作的头文件和库文件。



交叉编译 jpeg 库

解压 jpegsrc.v6b.tar.gz 到/root/yizhi

```
#cd jpeg-6b
```

设置环境变量

```
#export PATH=/usr/local/arm/4.4.1/bin:$PATH
```

```
#export CC=/usr/local/arm/4.4.1/bin/arm-linux-gcc
```

```
#./configure --enable-shared
```

修改生成的 Makefile 文件

```
CC=/usr/local/arm/4.4.1/bin/arm-linux-gcc
```

```
AR=/usr/local/arm/4.4.1/bin/arm-linux-ar rc
```

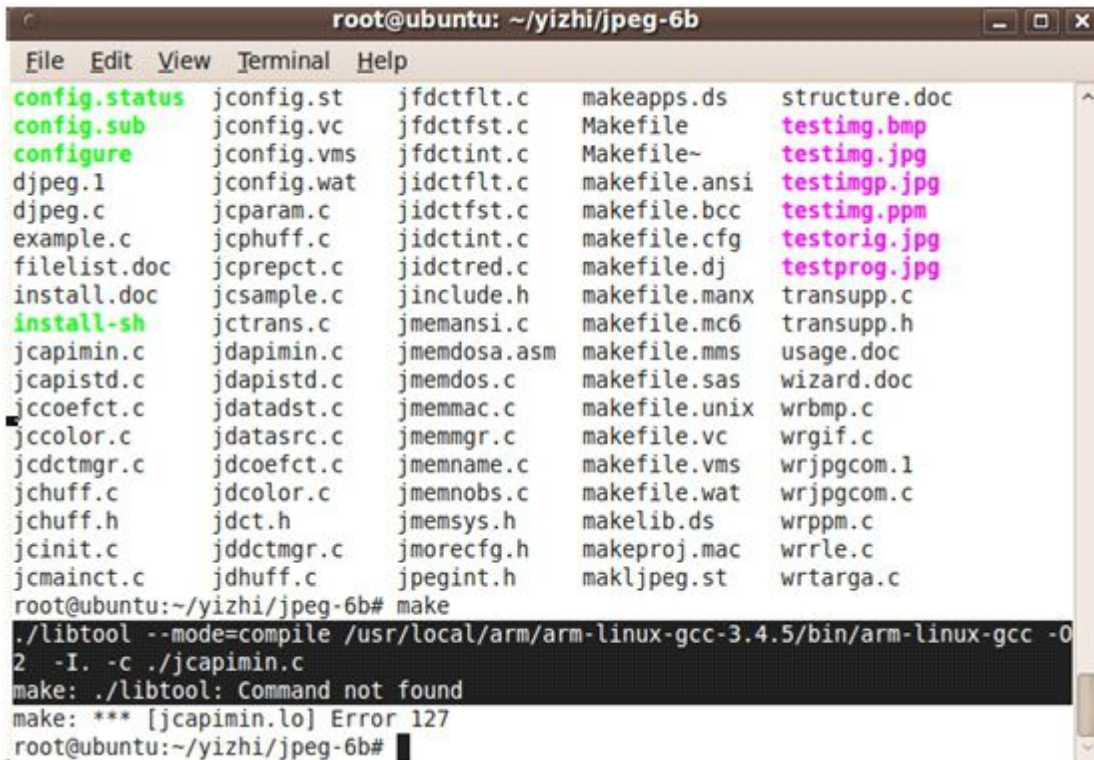
```
AR2=/usr/local/arm/4.4.1/bin/arm-linux-ranlib
```

保存

```
#cp jconfig.doc jconfig.h
```

```
#make
```

如果出现错误，提示找不到 libtool，如下图：



```

root@ubuntu: ~/yizhi/jpeg-6b
File Edit View Terminal Help
config.status  jconfig.st  jfdctflt.c  makeapps.ds  structure.doc
config.sub     jconfig.vc  jfdctfst.c  Makefile      testing.bmp
configure      jconfig.vms jfdctint.c  Makefile~     testing.jpg
djpeg.l        jconfig.wat jidctflt.c  makefile.ansi testingp.jpg
djpeg.c        jcparam.c   jidctfst.c  makefile.bcc  testing.ppm
example.c      jcphuff.c   jidctint.c  makefile.cfg  testorig.jpg
filelist.doc   jcprepct.c  jidctred.c  makefile.dj   testprog.jpg
install.doc    jcsample.c  jinclude.h  makefile.manx transupp.c
install-sh     jctrans.c   jmemansi.c  makefile.mc6  transupp.h
jcapimin.c     jdapimin.c  jmemdosa.asm makefile.mms  usage.doc
jcapistd.c     jdapistd.c  jmemdos.c   makefile.sas  wizard.doc
jccoeffct.c    jdatadst.c  jmemmac.c   makefile.unix wrbmp.c
jccolor.c      jdatasrc.c  jmemmgr.c   makefile.vc   wrgif.c
jcdctmgr.c     jdcoeft.c   jmemname.c  makefile.vms  wrjpgcom.1
jchuff.c       jdcclor.c   jmemnobs.c  makefile.wat  wrjpgcom.c
jchuff.h       jdct.h      jmemsys.h   makelib.ds    wrppm.c
jcinit.c       jddctmgr.c  jmorecfg.h  makeproj.mac  wrtle.c
jcmainct.c     jdhuft.c    jpegint.h   makljpeg.st   wrtarga.c
root@ubuntu:~/yizhi/jpeg-6b# make
./libtool --mode=compile /usr/local/arm/arm-linux-gcc-3.4.5/bin/arm-linux-gcc -O
2 -I. -c ./jcapimin.c
make: ./libtool: Command not found
make: *** [jcapimin.lo] Error 127
root@ubuntu:~/yizhi/jpeg-6b#

```

解决办法：

Ubuntu 上需要安装 libtool 工具，重新打开一个终端窗口，注意不要关闭编译 jpeg 时的窗口，在新窗口中执行以下命令：

```
#cd /root/yizhi/
```

```
#tar zxvf libtool-2.2.tar.gz
```

```
#cd /root/yizhi/libtool-2.2
```

```
#./configure --prefix=/usr
```

```
#make
```

```
#make install
```

这样 libtool 工具即可完成。

Libtool 工具安装完成。切换到编译 jpeg-6b 时的窗口，然后执行如下的命令：

```
#cp /usr/share/libtool/config/config.guess /root/yizhi/jpeg-6b
```

```
#cp /usr/share/libtool/config/config.sub /root.yizhi/jpeg-6b
```

```
#cd /root/yizhi/jpeg-6b
```

```
#./configure -enable-shared -enable-static
```

修改 Makefile 文件中的 CC , AR , AR2 变量 , 修改为:

```
CC=/usr/local/arm/4.4.1/bin/arm-linux-gcc
```

```
AR=/usr/local/arm/4.4.1/bin/arm-linux-ar rc
```

```
AR2=/usr/local/arm/4.4.1/bin/arm-linux-ranlib
```

保存 , 退出。

```
#make
```

编译通过。

执行以下命令即可完成 jpeg 头文件和库文件的制作工作。

```
#cp jpeglib.h jconfig.h jmorecfg.h /root/yizhi/arminclude
```

```
#cp .libs/libjpeg.so* /root/yizhi/armlib
```

交叉编译 e2fsprogs-1.40.2

编译 e2fsprogs-1.40.2 是为了得到二个文件 , uuid.h 及 libuuid.so , 编译 Qtopia2.2.0 需要这两个文件。

首先解压 e2fsprogs-1.40.2 , 解压后执行下面的命令 :

```
cd /root/yizhi/e2fsprogs-1.40.2
```

置环境变量

```
#export PATH=/usr/local/arm/4.4.1/bin:$PATH
```

```
#export CC=/usr/local/arm/4.4.1/bin/arm-linux-gcc
```

以上的两个环境变量如果已经设置 , 就不需要重新设置了 , 执行下面的命令来配置和编译 e2fsprogs-1.40.2:

```
# ./configure --enable-elf-shlibs --host=arm-linux
```

```
--with-cc=/usr/local/arm/4.4.1/bin/arm-linux-gcc
```

```
--with-linker=/usr/local/arm/4.4.1/bin/arm-linux-ld
```

```
# make
```

也许会提示 “makeinfo 命令没有找到” , 不过没关系。

将 e2fsprogs-1.40.2/lib/录下 uuid 的文件夹复制到 yizhi/arminclude 下

```
# cp -r lib/uuid /root/yizhi/arminclude
```

将 e2fsprogs-1.40.2/lib/录下面的库文件复制到/root/yizhi/armlib 下

```
# cp lib/libuuid.so* /root/yizhi/armlib/
```

e2fsprogs-1.40.2 相关工作完成。

交叉编译 libpng 库

将 libpng-1.2.19.tar.bz2 解压到/root/yizhi 录下

```
# cd /root/yizhi/libpng-1.2.1
```

```
# cp scripts/makefile.linux./Makefile
```

注意, 这里的 makefile 不是用./configure 生成的, 而是直接从 scripts 文件夹里面复制的 (见上面的 cp 命令), 修改 Makefile 文件:

```
# gedit Makefile
```

修改:

```
AR_RC=/usr/local/arm/4.4.1/bin/arm-linux-ar rc
```

```
CC=/usr/local/arm/4.4.1/bin/arm-linux-gcc
```

```
RANLIB=/usr/local/arm/4.4.1/bin/arm-linux-ranlib
```

保存

```
# make
```

```
# cp libpng12.so* /root/yizhi/armlib
```

```
# cp *.h /root/yizhi/arminclude
```

完成 libpng 头文件和库文件的制作。

交叉编译 zlib

编译前解压 zlib-1.2.3, 解压后

```
# cd /root/yizhi/zlib-1.2.3
```

设置环境变量

```
# export PATH=/usr/local/arm/4.4.1/bin:$PATH
```

```
# export LD_LIBRARY_PATH=/usr/local/arm/4.4.1/arm-
```

```
linux/lib:$LD_LIBRARY_PATH
```

```
# export CC= /usr/local/arm/4.4.1/bin/arm-linux-gcc
```

如果上面已经设置了 PATH 和 LD_LIBRARY_PATH 环境变量，这里不需要重新设置，

执行以下命令生成 Makefile 文件：

```
# ./configure --shared
```

修改一下：

```
# gedit Makefile
```

修改为：

```
CC=/usr/local/arm/4.4.1/bin/arm-linux-gcc
```

```
LDSHARED=/usr/local/arm/4.4.1/bin/arm-linux-gcc -shared-Wl, -soname ,
```

```
libz.so.1
```

```
CPP=/usr/local/arm/4.4.1/bin/arm-linux-gcc-E
```

```
AR=/usr/local/arm/4.4.1/bin/arm-linux-ar rc
```

```
RANLIB=/usr/local/arm/4.4.1/bin/arm-linux-ranlib
```

保存

```
#make
```

将编译生成的动态库及相关的头文件分别复制到 armlib 和 arminclude 录下

```
# cp libz.so* /root/yizhi/armlib
```

```
# cp *.h /root/yizhi/arminclude
```

经过以上步骤的操作，编译 Qtopia2.2.0 时所需的头文件和库文件制作完成。