开发环境：Red Hat Enterprise Linux 5
交叉编译工具：arm-linux-gcc-4.5.1
开发板内核版本：linux-3.5
libtool-1.5.6.tar.gz
jpegsrc.v6b.tar.gz
mjpg-streamer-r63.tar.gz

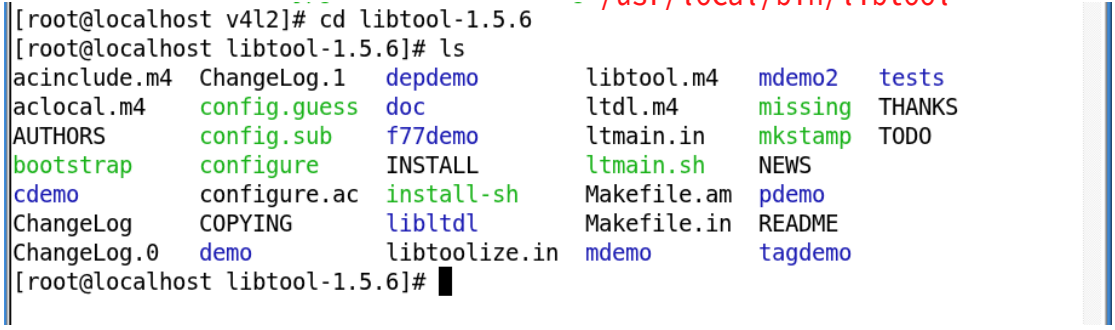
1、添加内核支持
1.1、Linux 内核目录
make menuconfig

Device Drivers --->
<*> Multimedia support --->
<*> Video For Linux
[*] Enable Video For Linux API 1 (DEPRECATED) (NEW)
[*] Video capture adapters (NEW) --->
[*] V4L USB devices (NEW) --->
<*> USB Video Class (UVC)
[*] UVC input events device support (NEW)

1.2、重新编译内核烧写
make uImage

2、移植 jpegsrc.v6b.tar.gz
2.1、安装 libtool-1.5.6.tar.gz（如果编译过，可以直接用）
2.1.1、解压源码包
tar xvzf libtool-1.5.6.tar.gz

2.1.2、进入解压目录进行配置安装
cd libtools-1.5.6



/usr/local/share/libtool/config.guess
/usr/local/share/libtool/config.sub
/usr/local/bin/libtool

默认配置安装：

./configure

#make && make install

2.2、编译 jpeg

2.2.1、解压 jpegsrc.v6b.tar.gz 进入解压目录

tar xvzf jpegsrc.v6b.tar.gz

```
[root@localhost v4l2]# ls
jpeg-6b             libtool-1.5.6         mjpg-streamer-r63.tar.gz
jpegsrc.v6b.tar.gz  libtool-1.5.6.tar.gz
[root@localhost v4l2]# cd jpeg-6b/
[root@localhost jpeg-6b]# ls
```

2.2.2、在 jpeg-6b 新建编译生成安装目录：

mkdir _install

cd _install

在指定安装目录下新建以下文件夹：

```
root@localhost:/xydzz/v4l2/jpeg-6b/_install - Shell - Konsole
会话  编辑  查看  书签  设置  帮助
[root@localhost _install]# ls
[root@localhost _install]# mkdir bin include lib man/man1 -p
[root@localhost _install]# ls
bin  include  lib  man
[root@localhost _install]#
```

2.2.3、返回 jpeg-6b 目录拷贝 libtools 到当前文件夹

/usr/local/share/libtool/config.guess
/usr/local/share/libtool/config.sub
/usr/local/bin/libtool

```
[root@localhost jpeg-6b]# cp /xydzz/v4l2/libtool-1.5.6/config.sub ./
cp：是否覆盖“./config.sub”? y
[root@localhost jpeg-6b]# cp /xydzz/v4l2/libtool-1.5.6/config.guess ./
cp：是否覆盖“./config.guess”? y
[root@localhost jpeg-6b]# cp /xydzz/v4l2/libtool-1.5.6/libtool ./
[root@localhost jpeg-6b]#
```

2.2.4、配置编译选项

./configure --prefix=/test/jpeg6 --host=arm-unknown-linux --prefix=/xydzz/v4l2/jpeg-6b/_install --enable-shared --enable-static

2.2.5、修改生成的 Makefile 的编译链

CC = arm-none-linux-gnueabi-gcc

```
16 includedir = $(prefix)/include
17 binprefix =
18 manprefix =
19 manext = 1
20 mandir = $(prefix)/man/man$(manext)
21
22 # The name of your C compiler:
23 CC= arm-none-linux-gnueabi-gcc
24
25 # You may need to adjust these cc options:
                                                    25,1          2%
```

2.2.6、编译安装

make && make install

2.2.7、安装目录拷贝动态库到开发板根文件系统

```
[root@localhost jpeg-6b]# cd _install/
[root@localhost _install]# ls
bin  include  lib  man
[root@localhost _install]# cd lib/
[root@localhost lib]# ls
libjpeg.a  libjpeg.la  libjpeg.so  libjpeg.so.62  libjpeg.so.62.0.0
[root@localhost lib]# cp *so* /xydzz/rootfs/lib/
[root@localhost lib]#
```

3、移植 mjpg-streamer

3.1、解压源码包进入解压目录

tar xvzf mjpg-streamer-r63.tar.gz

```
[root@localhost v4l2]# ls
jpeg-6b             libtool-1.5.6         mjpg-streamer-r63
jpegsrc.v6b.tar.gz  libtool-1.5.6.tar.gz  mjpg-streamer-r63.tar.gz
[root@localhost v4l2]# cd mjpg-streamer-r63
[root@localhost mjpg-streamer-r63]# ls
CHANGELOG  Makefile         mjpg_streamer.h  README    utils.c  www
LICENSE    mjpg_streamer.c  plugins          start.sh  utils.h
[root@localhost mjpg-streamer-r63]#
```

3.2、修改顶层 Makefile，指定编译工具为 arm-linux-gcc

[root@localhost mjpg-streamer-r63]# vim Makefile

```
 1 ###############################################################
 2 #
 3 # Purpose: Makefile for "M-JPEG Streamer"
 4 # Author.: Tom Stoeveken (TST)
 5 # Version: 0.3
 6 # License: GPL
 7 #
 8 ###############################################################
 9
10 CC = arm-linux-gcc
11
12 CFLAGS += -O2 -DLINUX -D_GNU_SOURCE -Wall
13 #CFLAGS += -O2 -DDEBUG -DLINUX -D_GNU_SOURCE -Wall
14 LFLAGS += -lpthread -ldl
```

3.3、修改 plugins/input_uvc/Makfile，指定编译工具为 arm-linux-gcc 和添加编译依赖库的路径

10 CC = arm-linux-gcc

14 CFLAGS += -O2 -DLINUX -D_GNU_SOURCE -Wall -shared -fPIC -I/xydzz/v4l2/jpeg-6b/_install/include

（注意-I 指定的头文件路径为 2.2 节编译 jpeg-6b 安装目录的 include）

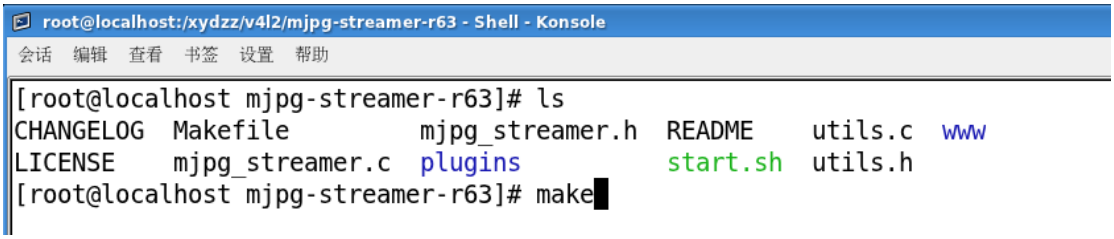
```
10 CC = arm-linux-gcc
11
12 OTHER_HEADERS = ../../mjpg_streamer.h ../../utils.h ../output.h ../input.h
13
14 CFLAGS += -O2 -DLINUX -D_GNU_SOURCE -Wall -shared -fPIC -I/xydzz/v4l2/jpeg-6
   b/_install/include
```

24 $(CC) $(CFLAGS) -ljpeg -L/xydzz/v4l2/jpeg-6b/_install/lib -o $@ input_uvc.c v4l2uvc.lo jpeg_utils.lo dynctrl.lo

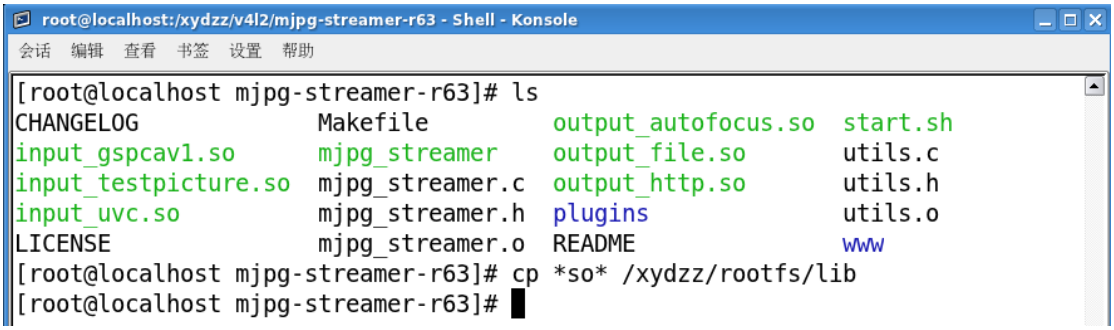
（注意-L 指定的头文件路径为 2.2 节编译 jpeg-6b 安装目录的 lib）

```
24          $(CC) $(CFLAGS) -ljpeg -L/xydzz/v4l2/jpeg-6b/_install/lib -o $@ input_uvc.c v4l2uvc.
    lo jpeg_utils.lo dynctrl.lo
```

3.4、编译



3.5、将生成的库拷贝到开发板



4、测试

将 linux 摄像头项目\v4l2_sample_v1.0_ok\v4l2_sample 拷贝到开发板运行测试