

# How to capture video using v4l2 and streaming video using ffmpeg

Reference:

Method adapted from boneCV:

boneCV: <https://github.com/derekmolloy/boneCV>

<http://derekmolloy.ie/beaglebone/beaglebone-video-capture-and-image-processing-on-embedded-linux-using-opencv/>

<http://derekmolloy.ie/udp-video-streaming-beaglebone-black/>

Useful previous cmpt433 how-to guides:

1. <http://www.cs.sfu.ca/CourseCentral/433/bfraser/other/2015-student-howtos/RecordingWebcamVideos.pdf>

2. <http://www.cs.sfu.ca/CourseCentral/433/bfraser/other/2014-student-howtos/WebCam.pdf>

3. <http://www.cs.sfu.ca/CourseCentral/433/bfraser/other/2016-student-howtos/WebCamVideoOpenCV.pdf>

## How to capture video

1. BBG comes with v4l2, to check

```
# whereis v4l2-ctl
```

```
v4l2-ctl: /usr/bin/v4l2-ctl
```

Some v4l2 commands(after connect camera to BBG):

```
# v4l2-ctl --list-formats
```

```
...
```

```
# v4l2-ctl --list-device
```

```
UVC Camera (046d:0825) (usb-musb-hdrc.1.auto-1):  
    /dev/video0
```

```
# v4l2-ctl --list-formats
```

```
ioctl: VIDIOC_ENUM_FMT
```

```
    Index    : 0
```

```
    Type     : Video Capture
```

```
    Pixel Format: 'YUYV'
```

```
    Name      : YUYV 4:2:2
```

```
    Index    : 1
```

```
    Type     : Video Capture
```

```
    Pixel Format: 'MJPG' (compressed)
```

```
    Name      : Motion-JPEG
```

```
# v4l2-ctl --get-priority
```

```
Priority: 2
```

```
# v4l2-ctl -D
```

```
Driver Info (not using libv4l2):
```

```
Driver name   : uvcvideo
Card type    : UVC Camera (046d:0825)
Bus info     : usb-musb-hdrc.1.auto-1
Driver version: 4.4.9
Capabilities  : 0x84200001
               Video Capture
               Streaming
               Extended Pix Format
               Device Capabilities
Device Caps   : 0x04200001
               Video Capture
               Streaming
               Extended Pix Format
```

## 2. how to capture video

1. # apt-get install libv4l-dev

2. store revised capture.c to NFS server

3. enter the NFS directory

```
# cd /mnt/remote/...
```

compile capture.c on BBG

```
# make
```

or

```
# gcc capture.c -lv4l2 -o capture
```

capture video and store it in output.raw

```
# ./captureVideo
```

or

```
# ./capture -F -c 300 -o > output.raw
```

-F force format to YUYV (after revision)

-c | --count Number of frames to grab [100] - use 0 for infinite

-o | --output Outputs stream to stdout

-h | --help

4. after capturing, a output.raw file will be produced. Convert raw file to mp4

```
# ./raw2mpg4
```

or

```
# ffmpeg -f rawvideo -vcodec rawvideo -s 320x240 -r 25 -pix_fmt yuv420p -i output.raw -c:v libx264 -
preset ultrafast -qp 0 output.mp4
```

## How to stream video using udp:

1. install ffmpeg on BBG

First, we need to add backports to our sources.list because we are running debian 8.4 on BBG

```
# cd /etc/apt/sources.list.d/
```

modify sources.list and add the following line to the end of the file  
deb http://ftp.debian.org/debian jessie-backports main

then ffmpeg can be installed using apt-get

```
# apt-get update
```

```
# apt-get install ffmpeg
```

2. install VLC media player on host

```
$ sudo apt update
```

```
$ sudo apt install vlc
```

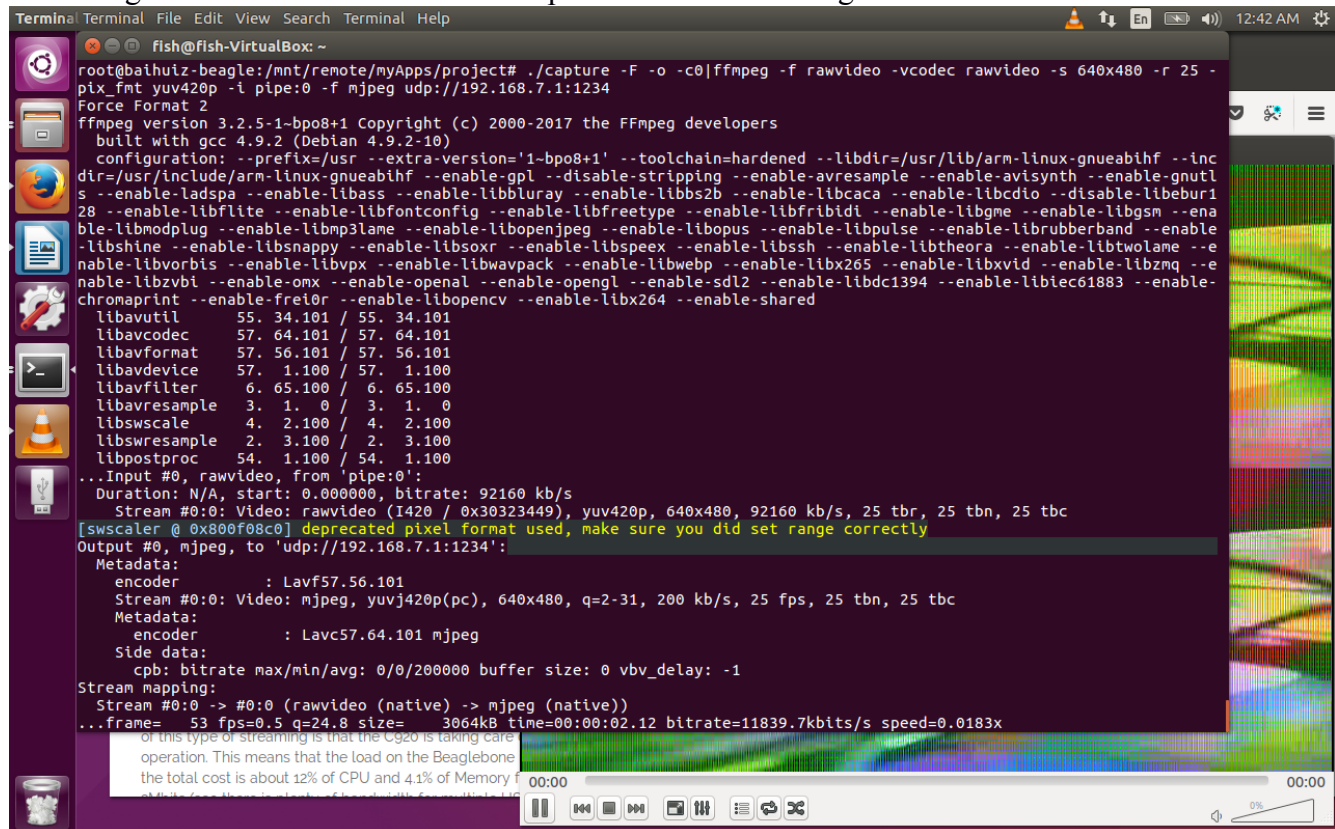
3. on BBG

```
# ./streamVideoUDP
```

or

```
# ./capture -F -o -c0|ffmpeg -f rawvideo -vcodec rawvideo -s 640x480 -r 25 -pix_fmt yuv420p -i pipe:0  
-f mjpeg udp://192.168.7.1:1234
```

Streaming is successful if the terminal output is like the following

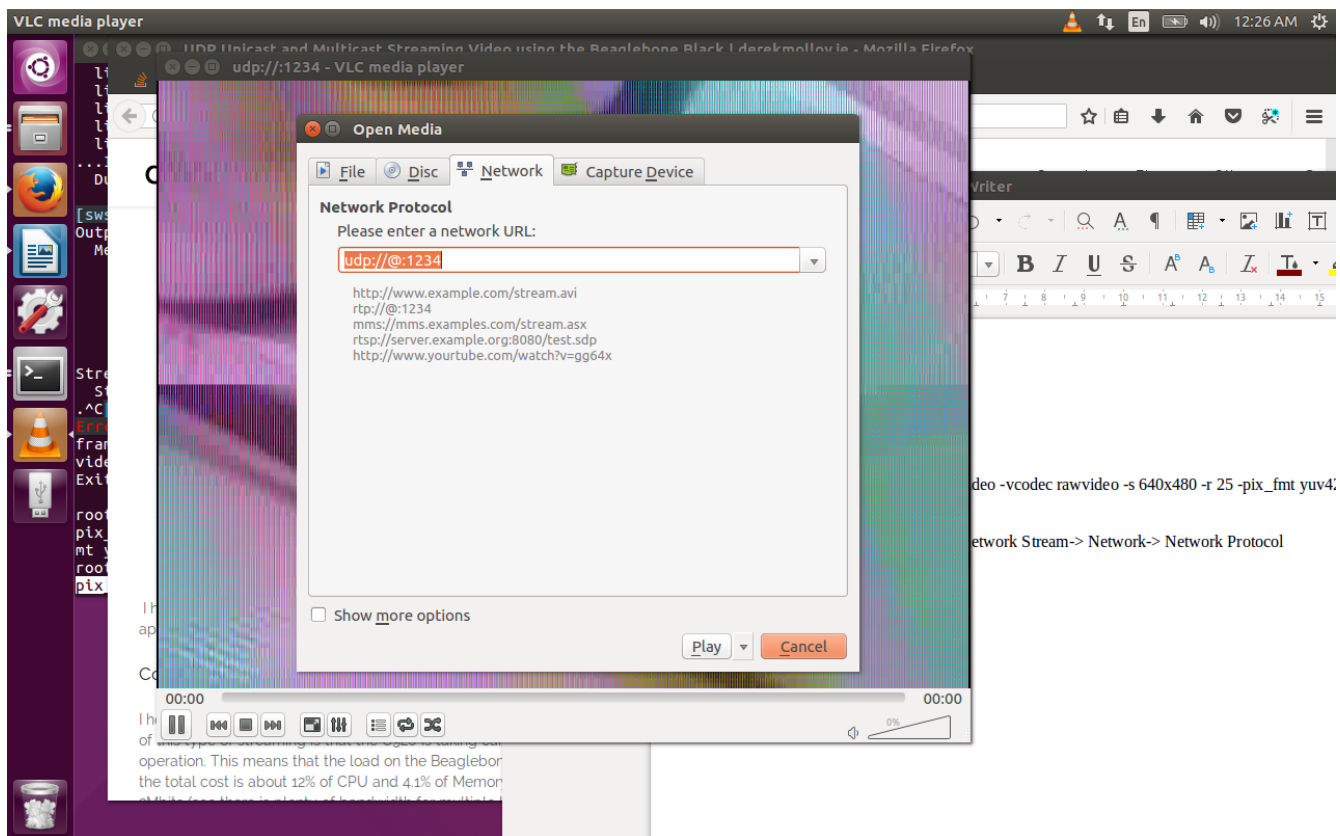


```
Terminal Terminal File Edit View Search Terminal Help
fish@fish-VirtualBox: ~
root@baituiz-beagle:/mnt/remote/myApps/project# ./capture -F -o -c0|ffmpeg -f rawvideo -vcodec rawvideo -s 640x480 -r 25 -
pix_fmt yuv420p -i pipe:0 -f mjpeg udp://192.168.7.1:1234
Force Format 2
ffmpeg version 3.2.5-1-bpo8+1 Copyright (c) 2000-2017 the FFmpeg developers
  built with gcc 4.9.2 (Debian 4.9.2-10)
  configuration: --prefix=/usr --extra-version='1-bpo8+1' --toolchain=hardened --libdir=/usr/lib/arm-linux-gnueabihf --inc
dir=/usr/include/arm-linux-gnueabihf --enable-gpl --disable-stripping --enable-avresample --enable-avisynth --enable-gnutl
s --enable-ladspa --enable-libass --enable-libbluray --enable-libbs2b --enable-libcaca --enable-libcdio --disable-libebur1
28 --enable-libflite --enable-libfontconfig --enable-libfreetype --enable-libfribidi --enable-libgme --enable-libgsm --ena
ble-libmodplug --enable-libmp3lame --enable-libopenjpeg --enable-libopus --enable-libpulse --enable-librubberband --enable
-libshine --enable-libsnappy --enable-libsoxr --enable-libspeex --enable-libssh --enable-libtheora --enable-libtwolame --e
nable-libvorbis --enable-libvpx --enable-libwavpack --enable-libwebp --enable-libx265 --enable-libxvid --enable-libzmq --e
nable-libzvt --enable-omx --enable-opengl --enable-sdl2 --enable-libdc1394 --enable-libiec61883 --enable-
chromaprint --enable-frei0r --enable-libopencv --enable-libx264 --enable-shared
libavutil      55. 34.101 / 55. 34.101
libavcodec     57. 64.101 / 57. 64.101
libavformat    57. 56.101 / 57. 56.101
libavdevice    57.  1.100 / 57.  1.100
libavfilter    6. 65.100 /  6. 65.100
libavresample  3.  1.  0 /  3.  1.  0
libswscale     4.  2.100 /  4.  2.100
libswresample  2.  3.100 /  2.  3.100
libpostproc   54.  1.100 / 54.  1.100
...Input #0, rawvideo, from 'pipe:0':
  Duration: N/A, start: 0.000000, bitrate: 92160 kb/s
  Stream #0:0: Video: rawvideo (I420 / 0x30323449), yuv420p, 640x480, 92160 kb/s, 25 tbr, 25 tbc
[swscaler @ 0x800f08c0] deprecated pixel format used, make sure you did set range correctly
Output #0, mjpeg, to 'udp://192.168.7.1:1234':
  Metadata:
    encoder      : Lavf57.56.101
  Stream #0:0: Video: mjpeg, yuvj420p(pc), 640x480, q=2-31, 200 kb/s, 25 fps, 25 tbn, 25 tbc
  Metadata:
    encoder      : Lavc57.64.101 mjpeg
  Side data:
    cpb: bitrate max/min/avg: 0/0/200000 buffer size: 0 vbv_delay: -1
Stream mapping:
  Stream #0:0 -> #0:0 (rawvideo (native) -> mjpeg (native))
...frame= 53 fps=0.5 q=24.8 size= 3064kB time=00:00:02.12 bitrate=11839.7kbits/s speed=0.0183x
or this type of streaming is that the Cg20 is taking care
operation. This means that the load on the Beaglebone
the total cost is about 12% of CPU and 4.1% of Memory
```

on host, open VLC, Media->open Network Stream-> Network-> Network Protocol->Please enter a  
network URL:

udp://@:1234

then click play



video -vcodec rawvideo -s 640x480 -r 25 -pix\_fmt yuv4;

network Stream-> Network-> Network Protocol