

h264_qsv 性能测试

OS : ubuntu 12.04 lts 64bit

Cpu: i3-4330T

内存 4G

主板设置: 显示输出接口 IGFX

显示内存设置: 448M

Intel media sdk 2014 R2

Ffmpeg 2.2

H264_qsv

输入的文件信息:

格式: Mkv

编码: h264

大小: 3.2G

视频总时间: 1 小时 26 分

```
'HDJ.Bounty.Lady.Ep19.20131220.HDTV.720p.x264-CHDTV.mkv':
  Metadata:
    encoder      : libebml v1.3.0 + libmatroska v1.4.1
    creation_time : 2013-12-20 19:08:40
  Duration: 01:26:24.44, start: 0.000000, bitrate: 4887 kb/s
  Stream #0:0(chi): Video: h264 (High), yuv420p, 1280x720 [SAR 1:1 DAR 16:9], 25
  fps, 25 tbr, 1k tbn, 50 tbc (default)
  Stream #0:1(chi): Audio: ac3, 48000 Hz, 5.1(side), fltp, 384 kb/s (default)
  Metadata:
    title      : Cantonese DD 5.1
  Stream #0:2(chi): Subtitle: subrip (default)
  Metadata:
    title      : CHS
  Stream #0:3(chi): Subtitle: subrip
  Metadata:
    title      : CHT
  Stream #0:4(eng): Subtitle: dvd_subtitle, 720x576
  Metadata:
    title      : ENG
```

一、单路视频将源转为 rate 2000k

开始转码,转换用时 7分27秒

```
time echo "gotron" | sudo -S $FFMPEG -i $INFILE -sn -acodec copy -vcodec h264_qsv -b:v 2000k -minrate 1000k -maxrate 4000k -f mp4 -y $OUTFILE

Output #0, mp4, to 'out.mp4':
  Metadata:
    encoder           : Lavf55.33.100
    Stream #0:0(chi): Video: h264 (h264_qsv) ([33][0][0][0] / 0x0021), nv12, 1280x720 [SAR 1:1 DAR 16:9], q=2-31, 2000 kb/s, 12800 tbn, 25 tbc (default)
    Stream #0:1(chi): Audio: ac3 ([165][0][0][0] / 0x00A5), 48000 Hz, 5.1(side), 384 kb/s (default)
  Metadata:
    title             : Cantonese DD 5.1
Stream mapping:
  Stream #0:0 -> #0:0 (h264 -> h264_qsv)
  Stream #0:1 -> #0:1 (copy)
Press [q] to stop, [?] for help
frame= 163 fps=0.0 q=-1.0 size= 1627kB time=00:00:06.94 bitrate=1919.6kbits/s
frame= 342 fps=341 q=-1.0 size= 3569kB time=00:00:14.17 bitrate=2062.5kbits/s
frame= 504 fps=335 q=-1.0 size= 5538kB time=00:00:20.51 bitrate=2211.8kbits/s
frame=129512 fps=290 q=-1.0 size= 151010kB time=01:26:16.80 bitrate=2390.5kbits/s
frame=129537 fps=290 q=-1.0 size= 1512107kB time=01:26:21.82 bitrate=2390.5kbits/s
frame=129611 fps=290 q=-1.0 size= 1516152kB time=01:26:24.40 bitrate=2395.7kbits/s
video:1270050kB audio:243016kB subtitle:0 data:0 global headers:0kB muxing overhead 0.203858%

real 7m27.007s
```

转码后的文件信息

```
gotron@gotron-Z87M-D3H:~/Desktop$ ffmpeg -i out.mp4
ffmpeg version 2.2 Copyright (c) 2000-2014 the FFmpeg developers
  built on Dec  4 2014 11:16:01 with gcc 4.6 (Ubuntu/Linaro 4.6.3-1ubuntu5)
  configuration: --prefix=/home/gotron/gotron/ffmpeg/qsv-ffmpeg-codec-master/ffmpeg-2.2/install --enable-gpl --enable-shared --enable-pic --enable-ffplay --extra-libs='-lmfxhw64 -lsupc++ -lstdc++ -ldl -lva -lva-drm' --extra-ldflags=-L/opt/intel/mediasdk/lib64/8086/0412 --extra-cflags=-I/opt/intel/mediasdk/include
  libavutil      52. 66.100 / 52. 66.100
  libavcodec     55. 52.102 / 55. 52.102
  libavformat    55. 33.100 / 55. 33.100
  libavdevice    55. 10.100 / 55. 10.100
  libavfilter     4.  2.100 /  4.  2.100
  libswscale     2.  5.102 /  2.  5.102
  libswresample  0. 18.100 /  0. 18.100
  libpostproc   52.  3.100 / 52.  3.100
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from 'out.mp4':
Metadata:
  major_brand      : isom
  minor_version    : 512
  compatible_brands: isomiso2avc1mp41
  encoder          : Lavf55.33.100
Duration: 01:26:24.44, start: 0.000000, bitrate: 2395 kb/s
  Stream #0:0(chi): Video: h264 (Main) (avc1 / 0x31637661), yuv420p, 1280x720 [SAR 1:1 DAR 16:9], 2007 kb/s, 25 fps, 25 tbr, 12800 tbn, 50 tbc (default)
Metadata:
  handler_name     : VideoHandler
  Stream #0:1(chi): Audio: ac3 (ac-3 / 0x332D6361), 48000 Hz, 5.1(side), fltp, 384 kb/s (default)
Metadata:
  handler_name     : SoundHandler
```

二、同时转换 2 路视频 一路为 6000k 一路为 2000k

用时 11 分 41 秒

三、10 个视频流同时转换

转码命令脚本:

```
FFMPEG=/home/gotron/gotron/ffmpeg/qsv-ffmpeg-codec-master/ffmpeg-2.2/install/bin/ffmpeg
INFILE="HDJ.Bounty.Lady.Ep19.20131220.HDTV.720p.x264-CHDTV.mkv"
COUNT=0
function filename()
{
```

```
COUNT=`expr $COUNT + 1`  
echo $COUNT  
}
```

```
RATE1="2000k -minrate 1500k -maxrate 7000k"  
RATE2="2500k -minrate 2000k -maxrate 7000k"  
RATE3="3000k -minrate 2000k -maxrate 7000k"  
RATE4="3500k -minrate 3000k -maxrate 7000k"  
RATE5="4000k -minrate 3000k -maxrate 7000k"  
RATE6="4500k -minrate 4000k -maxrate 7000k"  
RATE7="5000k -minrate 4000k -maxrate 7000k"  
RATE8="5500k -minrate 4000k -maxrate 7000k"  
RATE9="6000k -minrate 5000k -maxrate 7000k"  
RATE10="6500k -minrate 5000k -maxrate 7000k"
```

filename

```
TR1="-sn -acodec copy -vcodec h264_qsv -b:v $RATE1 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR2="-sn -acodec copy -vcodec h264_qsv -b:v $RATE2 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR3="-sn -acodec copy -vcodec h264_qsv -b:v $RATE3 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR4="-sn -acodec copy -vcodec h264_qsv -b:v $RATE4 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR5="-sn -acodec copy -vcodec h264_qsv -b:v $RATE5 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR6="-sn -acodec copy -vcodec h264_qsv -b:v $RATE6 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR7="-sn -acodec copy -vcodec h264_qsv -b:v $RATE7 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR8="-sn -acodec copy -vcodec h264_qsv -b:v $RATE8 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR9="-sn -acodec copy -vcodec h264_qsv -b:v $RATE9 -f mp4 -y  
$COUNT.mp4"
```

filename

```
TR10="-sn -acodec copy -vcodec h264_qsv -b:v $RATE10 -f mp4  
-y $COUNT.mp4"
```

```
#single in single out
```

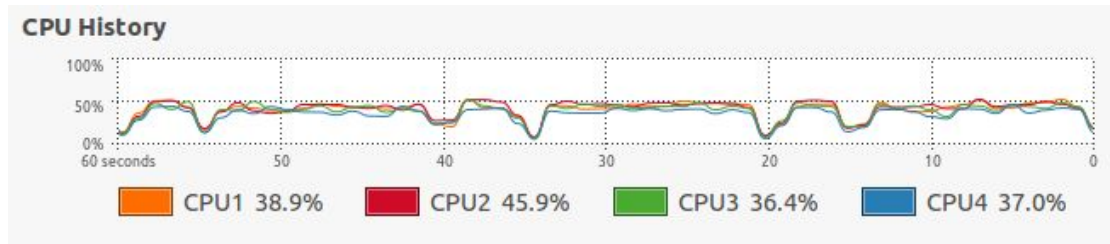
```
#time echo "gotron" | sudo -S $FFMPEG -i $INFILE -sn -acodec  
copy -vcodec h264_qsv -b:v 6000k -minrate 4000k -maxrate 7000k  
-f mp4 -y $OUTFILE
```

```
time echo "gotron" | sudo -S $FFMPEG -i $INFILE $STR1 $STR2  
$STR3 $STR4 $STR5 $STR6 $STR7 $STR8 $STR9 $STR10
```

开始转码:

```
title : Cantonese DD 5.1  
Stream mapping:  
Stream #0:0 -> #0:0 (h264 -> h264_qsv)  
Stream #0:1 -> #0:1 (copy)  
Stream #0:0 -> #1:0 (h264 -> h264_qsv)  
Stream #0:1 -> #1:1 (copy)  
Stream #0:0 -> #2:0 (h264 -> h264_qsv)  
Stream #0:1 -> #2:1 (copy)  
Stream #0:0 -> #3:0 (h264 -> h264_qsv)  
Stream #0:1 -> #3:1 (copy)  
Stream #0:0 -> #4:0 (h264 -> h264_qsv)  
Stream #0:1 -> #4:1 (copy)  
Stream #0:0 -> #5:0 (h264 -> h264_qsv)  
Stream #0:1 -> #5:1 (copy)  
Stream #0:0 -> #6:0 (h264 -> h264_qsv)  
Stream #0:1 -> #6:1 (copy)  
Stream #0:0 -> #7:0 (h264 -> h264_qsv)  
Stream #0:1 -> #7:1 (copy)  
Stream #0:0 -> #8:0 (h264 -> h264_qsv)  
Stream #0:1 -> #8:1 (copy)  
Stream #0:0 -> #9:0 (h264 -> h264_qsv)  
Stream #0:1 -> #9:1 (copy)
```

整个转码过程中 Cpu 使用情况



HDJ...mkv 为输入文件，源文件时间 1 小时 26 分，转换总用时 54 分 16 秒

```

frame=129567 fps= 40 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
frame=129590 fps= 40 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
frame=129611 fps= 40 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
frame=129611 fps= 40 q=-1.0 Lq=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
0 q=-1.0 size= 1491145kB time=01:26:24.40 bitrate=2356.2kb/s
video:26747923kB audio:2430165kB subtitle:0 data:0 global headers:5kB muxing overhead
-94.889503%

real    54m16.634s
user    55m18.832s
sys     12m9.940s
  
```

各文件大小

```

gotron@gotron-Z87M-D3H:~/Desktop$ ls -sh
total 31G
4.2G 10.mp4  3.0G 6.mp4
1.5G 1.mp4    3.3G 7.mp4
1.8G 2.mp4    3.6G 8.mp4
2.1G 3.mp4    3.9G 9.mp4
2.4G 4.mp4    3.0G HDJ.Bounty.Lady.Ep19.20131220.HDTV.720p.x264-CHDTV.mkv
2.7G 5.mp4    4.0K test2.sh
  
```

源文件二：

源码率 20347kb/s
 编码 h264
 分辨率 1920x1080
 时长 58 秒

```

Input #0, mov,mp4,m4a,3gp,3g2,mj2, from '1080h264.mp4':
  Metadata:
    major_brand      : mp42
    minor_version    : 1
    compatible_brands: mp41mp42isom
    creation_time    : 2014-12-06 02:28:17
  Duration: 00:00:58.56, start: 0.000000, bitrate: 20347 kb/s
    Stream #0:0(zho): Audio: aac (mp4a / 0x6134706D), 48000 Hz, stereo, flt
0 kb/s (default)
    Metadata:
      creation_time  : 2014-12-06 02:28:17
      handler_name    : Core Media Audio
    Stream #0:1(und): Video: h264 (High) (avc1 / 0x31637661), yuv420p(tv,
, 1920x1080, 20079 kb/s, SAR 1:1 DAR 16:9, 50 fps, 50 tbr, 50 tbn, 50 tbc
lt)
    Metadata:
      creation_time  : 2014-12-06 02:28:17
      handler_name    : Core Media Video

```

1.如果用 libx264 转一路成为 2000kb/s 要 2分 57秒, 使用 h264_qvs 要 39秒

2.转 10路:

修改上面测试脚本:

RATE1="2000k -minrate 1500k -maxrate 7000k "

RATE2="2500k -minrate 2000k -maxrate 7000k "

RATE3="3000k -minrate 2000k -maxrate 7000k "

RATE4="3500k -minrate 3000k -maxrate 7000k "

RATE5="4000k -minrate 3000k -maxrate 7000k "

RATE6="4500k -minrate 4000k -maxrate 7000k "

RATE7="5000k -minrate 4000k -maxrate 7000k "

RATE8="5500k -minrate 4000k -maxrate 7000k "

RATE9="22000k -minrate 9000k -maxrate 25000k "

RATE10="24000k -minrate 9000k -maxrate 25000k "

```

frame= 2886 fps= 22 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
frame= 2896 fps= 22 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
frame= 2908 fps= 22 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
frame= 2920 fps= 22 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
frame= 2928 fps= 22 q=-1.0 Lq=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0
1.0 q=-1.0 size= 16334kB time=00:00:58.62 bitrate=2282.5kbits/s
video:554220kB audio:18618kB subtitle:0 data:0 global headers:5kB muxing ov
d -97.148537%

real    2m13.970s
user    4m13.036s
sys     0m33.352s

```

转换用时 2分 13秒

3.转换 TR1-TR5 五路 用时 1分 04秒


```

frame= 2923 fps= 46 q=-1.0 q=-1.0 q=-1.0 q=-1.0 q=-1.0 size= 16216kB time=
frame= 2928 fps= 46 q=-1.0 Lq=-1.0 q=-1.0 q=-1.0 q=-1.0 size= 16340kB time=
00:58.62 bitrate=2283.4kbits/s
video:109320kB audio:9309kB subtitle:0 data:0 global headers:2kB muxing over
-86.225990%

real    1m4.430s
user    3m42.096s
sys     0m6.792s

```

4.转换 TR1-TR4 四路 用时 58 秒

```

frame= 2819 fps= 51 q=-1.0 q=-1.0 q=-1.0 q=-1.0 size= 15722kB time=00:00:
frame= 2845 fps= 51 q=-1.0 q=-1.0 q=-1.0 q=-1.0 size= 15851kB time=00:00:
frame= 2872 fps= 51 q=-1.0 q=-1.0 q=-1.0 q=-1.0 size= 16023kB time=00:00:
frame= 2902 fps= 51 q=-1.0 q=-1.0 q=-1.0 q=-1.0 size= 16173kB time=00:00:
frame= 2928 fps= 51 q=-1.0 Lq=-1.0 q=-1.0 q=-1.0 size= 16422kB time=00:00:
2 bitrate=2294.8kbits/s
video:79997kB audio:7447kB subtitle:0 data:0 global headers:2kB muxing over
-81.220352%

real    0m58.042s
user    3m31.100s
sys     0m5.664s

```

源文件三:

这个是用罗生给的文件转换后得来的，可能有些 mpeg2video

h264_qsv 不能支持,或给来

那个文件有问题。

罗生给的文件信息:

```

[mpeg2video @ 0x1724660] Invalid frame dimensions 0x0.
  Last message repeated 10 times
Input #0, mpegts, from '[空中危机].Flightplan.2005.Hdtv.1080i.sShadow.dts.fai
ilu.disk2.ts':
  Duration: 00:31:03.70, start: 4018.929689, bitrate: 9617 kb/s
  Program 2
    Stream #0:0[0x21]: Video: mpeg2video (Main) ([2][0][0][0] / 0x0002), yu
(tv, bt709), 1920x1080 [SAR 1:1 DAR 16:9], max. 14000 kb/s, 28.75 fps, 59.9
, 90k tbn, 59.94 tbc
    Stream #0:1[0x24](ENG): Audio: dts (DTS) (DTS1 / 0x31535444), 48000 Hz,
side), fltp, 768 kb/s
    Stream #0:2[0x25](ENG): Audio: ac3 ([129][0][0][0] / 0x0081), 48000 Hz,
side), fltp, 448 kb/s

```

转换命令: \$FFMPEG -i \$INFILE -map 0:1 -c:a copy -map 0:2 -c:a copy -map 0:0 -b:v 9617k

-minrate 6000k -maxrate 20000k -vcodec mpeg2video -y my.ts
转换后文件（以下称源文件，mpeg2video --> h264_qsv），
源码率 13719kb/s
编码 mpeg2video
分辨率 1920x1080
时长 31 秒

```
Input #0, mpegts, from 'my.ts':
  Duration: 00:31:03.70, start: 1.433367, bitrate: 13719 kb/s
  Program 1
    Metadata:
      service_name      : Service01
      service_provider : FFmpeg
      Stream #0:0[0x100](ENG): Audio: dts (DTS) ([6][0][0][0] / 0x0006), 48000 Hz (side), fltp, 768 kb/s
      Stream #0:1[0x101](ENG): Audio: ac3 ([129][0][0][0] / 0x0081), 48000 Hz (side), fltp, 448 kb/s
      Stream #0:2[0x102]: Video: mpeg2video (Main) ([2][0][0][0] / 0x0002), yuv420p(tv), 1920x1080 [SAR 1:1 DAR 16:9], max. 20000 kb/s, 29.97 fps, 29.97 tbr, 13.5 tbn, 59.94 tbc
```

设置转换条件

```
RATE1="6000k -minrate 1500k -maxrate 20000k "  
RATE2="2500k -minrate 2000k -maxrate 7000k "  
RATE3="3000k -minrate 2000k -maxrate 7000k "  
RATE4="3500k -minrate 3000k -maxrate 7000k "  
RATE5="4000k -minrate 3000k -maxrate 7000k "  
RATE6="4500k -minrate 4000k -maxrate 7000k "  
RATE7="5000k -minrate 4000k -maxrate 7000k "  
RATE8="5500k -minrate 4000k -maxrate 7000k "  
RATE9="11000k -minrate 9000k -maxrate 25000k "  
RATE10="13000k -minrate 9000k -maxrate 25000k "
```

转换 1 路用时 mpeg2video --> h264_qsv : 7 分 37 秒

```

Output #0, avi, to '1.avi':
  Metadata:
    ISFT                : Lavf55.33.100
    Stream #0:0: Video: h264 (h264_qsv) (H264 / 0x34363248), nv12, 1920x1080, 29.97 tbn, 29.97 tbc
    Stream #0:1(ENG): Audio: dts ([1] [0][0] / 0x2001), 48000 Hz, 5.1(side)
Stream mapping:
  Stream #0:2 -> #0:0 (mpeg2video -> h264_qsv)
  Stream #0:0 -> #0:1 (copy)
Press [q] to stop, [?] for help
frame=55816 fps=123 q=-1.0 Lsize= 5313570kB time=00:31:03.71 bitrate=23356.
video:1339074kB audio:171653kB subtitle:0 data:0 global headers:0kB muxing

real    7m37.183s
user    10m42.800s
sys     0m24.084s

```

转换 4 路 mpeg2video --> h264_qsv (RATE1 - RATE5):

```

Stream #3:0: Video: h264 (h264_qsv) (H264 / 0x34363248), nv12, 1920x1080, 29.97 tbn, 29.97 tbc
R 1:1 DAR 16:9], q=2-31, 3500 kb/s, 29.97 tbn, 29.97 tbc
  Stream #3:1(ENG): Audio: dts ([1] [0][0] / 0x2001), 48000 Hz, 5.1(side)
kb/s
Stream mapping:
  Stream #0:2 -> #0:0 (mpeg2video -> h264_qsv)
  Stream #0:0 -> #0:1 (copy)
  Stream #0:2 -> #1:0 (mpeg2video -> h264_qsv)
  Stream #0:0 -> #1:1 (copy)
  Stream #0:2 -> #2:0 (mpeg2video -> h264_qsv)
  Stream #0:0 -> #2:1 (copy)
  Stream #0:2 -> #3:0 (mpeg2video -> h264_qsv)
  Stream #0:0 -> #3:1 (copy)
Press [q] to stop, [?] for help

```

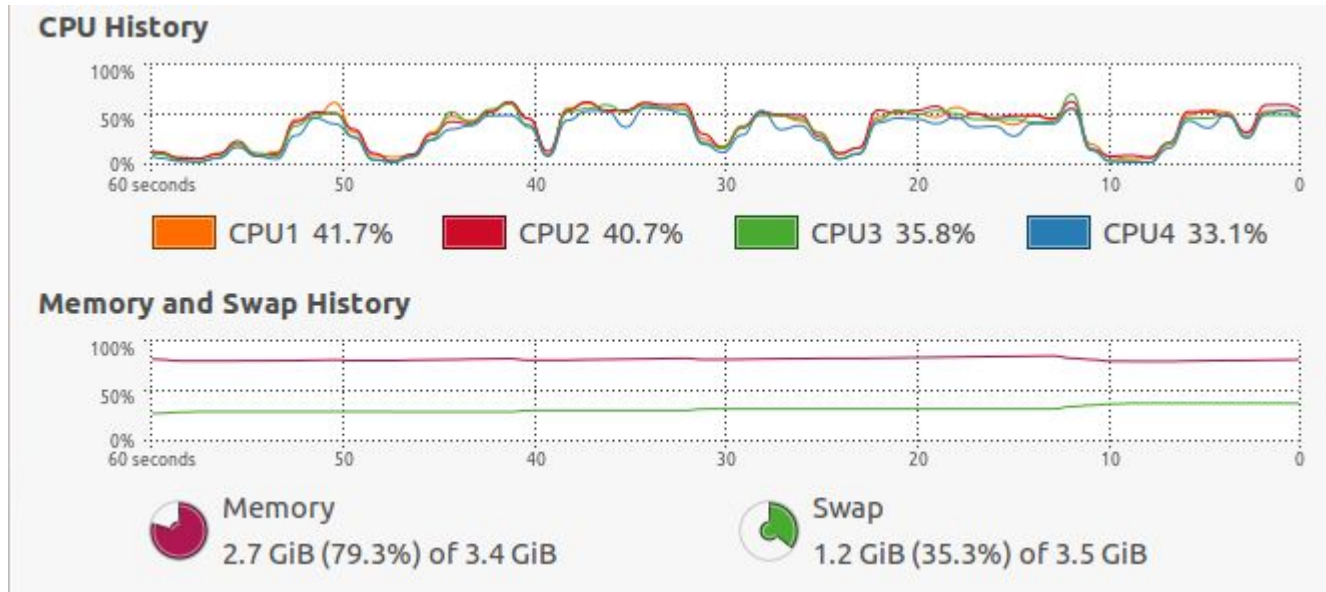
转换多路，内存消耗很大，使用到了交换分区，根本无法进行下去，系统变慢,也许内存会成为瓶颈。参考下面的讨论

<http://forum.videohelp.com/threads/356970-Does-FFMpeg-have-an-option-to-limit-the-memory-usage>

下面这个有说内存吃紧，可能是源文件有问题：

<http://forum.videohelp.com/threads/242265-ffmpeg-exe-EATING-memory>

下周换个文件再测。



Ffmpeg 编译配置信息:

C compiler	gcc
C library	glibc
ARCH	x86 (generic)
big-endian	no
runtime cpu detection	yes
yasm	no
MMX enabled	yes
MMXEXT enabled	yes
3DNow! enabled	yes
3DNow! extended enabled	yes
SSE enabled	yes
SSSE3 enabled	yes
AVX enabled	yes
XOP enabled	yes
FMA3 enabled	yes
FMA4 enabled	yes
i686 features enabled	yes
CMOV is fast	yes
EBX available	yes
EBP available	yes
debug symbols	yes
strip symbols	yes
optimize for size	no
optimizations	yes
static	yes
shared	yes

postprocessing support	yes
new filter support	yes
network support	yes
threading support	pthread
safe bitstream reader	yes
SDL support	no
opencl enabled	no
libzvbi enabled	no
texi2html enabled	no
perl enabled	yes
pod2man enabled	yes
makeinfo enabled	no

External libraries:

iconv	libvorbis	libx264
libfaac	libvpx	libxvid
libmp3lame		

Enabled decoders:

aac	bintext	iff_byterun1
aac_latm	bmp	iff_ilbm
aasc	bmvideo	imc
ac3	bmvideo	indeo2
adpcm_4xm	brender_pix	indeo3
adpcm_adx	c93	indeo4
adpcm_afc	cavs	indeo5
adpcm_ct	cdgraphics	interplay_dpcm
adpcm_dtk	cdxl	interplay_video
adpcm_ea	cinepak	jacsub
adpcm_ea_maxis_xa	cljr	jpeg2000
adpcm_ea_r1	cllc	jpegls
adpcm_ea_r2	comfortnoise	jv
adpcm_ea_r3	cook	kgv1
adpcm_ea_xas	cpia	kmvc
adpcm_g722	cscd	lagarith
adpcm_g726	cyuv	libvorbis
adpcm_g726le	dca	libvpx_vp8
adpcm_ima_amv	dfa	loco
adpcm_ima_apc	dirac	mace3
adpcm_ima_dk3	dnxhd	mace6
adpcm_ima_dk4	dpx	mdec
adpcm_ima_ea_eacs	dsicinaudio	metasound
adpcm_ima_ea_sead	dsicinvideo	microvd
adpcm_ima_iss	dvbsub	mimic

adpcm_ima_oki	dvdsub	mjpeg
adpcm_ima_qt	dvvideo	mjpegb
adpcm_ima_rad	dxtory	mlp
adpcm_ima_smjpeg	eac3	mmvideo
adpcm_ima_wav	eacmv	motionpixels
adpcm_ima_ws	eamad	movtext
adpcm_ms	eatgq	mp1
adpcm_sbpro_2	eatgv	mp1float
adpcm_sbpro_3	eatqi	mp2
adpcm_sbpro_4	eightbps	mp2float
adpcm_swf	eightsvx_exp	mp3
adpcm_thp	eightsvx_fib	mp3adu
adpcm_xa	escape124	mp3adufloat
adpcm_yamaha	escape130	mp3float
aic	evrc	mp3on4
alac	ffv1	mp3on4float
als	ffvhuff	mpc7
amrnb	ffwavesynth	mpc8
amrwb	fic	mpeg1video
amv	flac	mpeg2video
anm	flic	mpeg4
ansi	flv	mpegvideo
ape	fourxm	mpl2
ass	fraps	msa1
asv1	frwu	msmpeg4v1
asv2	g723_1	msmpeg4v2
atrac1	g729	msmpeg4v3
atrac3	gif	msrle
atrac3p	gsm	mss1
aura	gsm_ms	mss2
aura2	h261	msvideo1
avrn	h263	mszh
avrp	h263i	mts2
avs	h263p	mvc1
avui	h264	mvc2
ayuv	hevc	mxpeg
bethsoftvid	hnm4_video	nellymoser
bfi	huffyuv	nuv
bink	iac	paf_audio
binkaudio_dct	idcin	paf_video
binkaudio_rdft	idf	pam
pbm	realtxt	v210
pcm_alaw	rl2	v210x
pcm_bluray	roq	v308

pcm_dvd	roq_dpcm	v408
pcm_f32be	rpza	v410
pcm_f32le	rv10	vb
pcm_f64be	rv20	vble
pcm_f64le	rv30	vc1
pcm_lxf	rv40	vc1image
pcm_mulaw	s302m	vcr1
pcm_s16be	sami	vima
pcm_s16be_planar	sanm	vmdaudio
pcm_s16le	sgi	vmdvideo
pcm_s16le_planar	sgirle	vmnc
pcm_s24be	shorten	vorbis
pcm_s24daud	sipr	vp3
pcm_s24le	smackaud	vp5
pcm_s24le_planar	smacker	vp6
pcm_s32be	smc	vp6a
pcm_s32le	smvjpeg	vp6f
pcm_s32le_planar	snow	vp8
pcm_s8	sol_dpcm	vp9
pcm_s8_planar	sonic	vplayer
pcm_u16be	sp5x	vqa
pcm_u16le	srt	wavpack
pcm_u24be	ssa	webp
pcm_u24le	subrip	webvtt
pcm_u32be	subviewer	wmalossless
pcm_u32le	subviewer1	wmapro
pcm_u8	sunrast	wmav1
pcm_zork	svq1	wmav2
pcx	svq3	wmavoiced
pgm	tak	wmv1
pgmyuv	targa	wmv2
pgssub	targa_y216	wmv3
pictor	text	wmv3image
pjs	theora	wmv1
ppm	thp	ws_snd1
prores	tiertexseqvideo	xan_dpcm
prores_lgpl	tiff	xan_wc3
ptx	tmv	xan_wc4
qcelp	truehd	xbin
qdm2	truemotion1	xbm
qdraw	truemotion2	xface
qpeg	truespeech	xl
qtrle	tsc2	xsub
r10k	tta	xwd

r210	twinvq	y41p
ra_144	txd	yop
ra_288	ulti	yuv4
ralf	utvideo	zero12v
rawvideo		

Enabled encoders:

a64multi	libmp3lame	pcx
a64multi5	libvorbis	pgm
aac	libvpx_vp8	pgmyuv
ac3	libx264	ppm
ac3_fixed	libx264rgb	prores
adpcm_adx	libxvid	prores_aw
adpcm_g722	ljpeg	prores_ks
adpcm_g726	mjpeg	qtrle
adpcm_ima_qt	movtext	r10k
adpcm_ima_wav	mp2	r210
adpcm_ms	mp2fixed	ra_144
adpcm_swf	mpeg1video	rawvideo
adpcm_yamaha	mpeg2video	roq
alac	mpeg4	roq_dpcm
amv	msmpeg4v2	rv10
ass	msmpeg4v3	rv20
asv1	msvideo1	s302m
asv2	nellymoser	sgi
avrp	pam	snow
avui	pbm	sonic
ayuv	pcm_alaw	sonic_ls
bmp	pcm_f32be	srt
cinepak	pcm_f32le	ssa
cljr	pcm_f64be	subrip
comfortnoise	pcm_f64le	sunrast
dca	pcm_mulaw	svq1
dnxhd	pcm_s16be	targa
dpx	pcm_s16be_planar	tiff
dvbsub	pcm_s16le	tta
dvdsup	pcm_s16le_planar	utvideo
dvvideo	pcm_s24be	v210
eac3	pcm_s24daud	v308
ffv1	pcm_s24le	v408
ffvhuff	pcm_s24le_planar	v410
flac	pcm_s32be	vorbis
flv	pcm_s32le	wavpack
g723_1	pcm_s32le_planar	wmav1

gif	pcm_s8	wmav2
h261	pcm_s8_planar	wmv1
h263	pcm_u16be	wmv2
h263p	pcm_u16le	xbm
h264_qsv	pcm_u24be	xface
huffyuv	pcm_u24le	xsub
jpeg2000	pcm_u32be	xwd
jpegls	pcm_u32le	y41p
libfaac	pcm_u8	yuv4

Enabled hwaccels:

h263_vaapi	mpeg2_vaapi	vc1_vaapi
h264_vaapi	mpeg4_vaapi	wmv3_vaapi

Enabled parsers:

aac	dvd_nav	mpegvideo
aac_latm	dvdsup	png
ac3	flac	pnm
adx	gsm	rv30
bmp	h261	rv40
cavsvideo	h263	tak
cook	h264	vc1
dca	hevc	vorbis
dirac	mjpeg	vp3
dnxhd	mlp	vp8
dpx	mpeg4video	vp9
dvbsub	mpegaudio	

Enabled demuxers:

aac	hevc	pcm_s32le
ac3	hls	pcm_s8
act	hnm	pcm_u16be
adf	ico	pcm_u16le
adp	idcin	pcm_u24be
adx	idf	pcm_u24le
aea	iff	pcm_u32be
afc	ilbc	pcm_u32le
aiff	image2	pcm_u8
amr	image2pipe	pjs
anm	ingenient	pmp
apc	ipmovie	pva
ape	ircam	pvf
aqtitle	iss	qcp
asf	iv8	r3d

ass	ivf	rawvideo
ast	jacosub	realtext
au	jv	redspark
avi	latm	rl2
avr	lmlm4	rm
avs	loas	roq
bethsoftvid	lvf	rpl
bfi	lxf	rsd
bink	m4v	rso
bintext	matroska	rtp
bit	mgsts	rtsp
bmv	microdvd	sami
boa	mjpeg	sap
brstm	mlp	sbg
c93	mm	sdp
caf	mmf	sdr2
cavsvideo	mov	segafilm
cdg	mp3	shorten
cdxl	mpc	siff
concat	mpc8	smacker
data	mpegps	smjpeg
daud	mpegs	smush
dfa	mpegsraw	sol
dirac	mpegvideo	sox
dnxhd	mpl2	spdif
dsicin	mbsub	srt
dts	msnwc_tcp	str
dtshd	mtv	subviewer
dv	mv	subviewer1
dxa	mvi	swf
ea	mxf	tak
ea_cdata	mxg	tedcaptions
eac3	nc	thp
epaf	nistsphere	tiertexseq
ffm	nsv	tmv
ffmetadata	nut	truehd
filmstrip	nuv	tta
flac	ogg	tty
flic	oma	txd
flv	paf	vc1
fourxm	pcm_alaw	vc1t
frm	pcm_f32be	vivo
g722	pcm_f32le	vmd
g723_1	pcm_f64be	vobsub

g729	pcm_f64le	voc
gif	pcm_mulaw	vplayer
gsm	pcm_s16be	vqf
gxf	pcm_s16le	w64
h261	pcm_s24be	wav
h263	pcm_s24le	wc3
h264	pcm_s32be	webvtt
wsaud	xa	xwma
wsvqa	xbin	yop
wtv	xmv	yuv4mpegpipe
wv		

Enabled muxers:

a64	image2pipe	pcm_s24be
ac3	ipod	pcm_s24le
adts	ircam	pcm_s32be
adx	ismv	pcm_s32le
aiff	ivf	pcm_s8
amr	jacosub	pcm_u16be
asf	latm	pcm_u16le
asf_stream	m4v	pcm_u24be
ass	matroska	pcm_u24le
ast	matroska_audio	pcm_u32be
au	md5	pcm_u32le
avi	microdvd	pcm_u8
avm2	mjpeg	psp
bit	mkvtimestamp_v2	rawvideo
caf	mlp	rm
cavsvideo	mmf	roq
crc	mov	rso
data	mp2	rtp
daud	mp3	rtsp
dirac	mp4	sap
dnxhd	mpeg1system	segment
dtc	mpeg1vcd	smjpeg
dv	mpeg1video	smoothstreaming
eac3	mpeg2dvd	sox
f4v	mpeg2svcd	spdif
ffm	mpeg2video	speex
ffmetadata	mpeg2vob	srt
filmstrip	mpegts	stream_segment
flac	mpjpeg	swf
flv	mxfl	tee
framecrc	mxfl_d10	tg2

framemd5	null	tgp
g722	nut	truehd
g723_1	ogg	uncodedframecrc
gif	oma	vc1
gxf	opus	vc1t
h261	pcm_alaw	voc
h263	pcm_f32be	w64
h264	pcm_f32le	wav
hds	pcm_f64be	webm
hevc	pcm_f64le	webvtt
hls	pcm_mulaw	wtv
ico	pcm_s16be	wv
ilbc	pcm_s16le	yuv4mpegpipe
image2		

Enabled protocols:

cache	hls	rtmp
concat	http	rtmpt
crypto	httpproxy	rtp
data	md5	srtplib
ffrtmphttp	mms	tcp
file	mmst	udp
ftp	pipe	unix
gopher		

Enabled filters:

aconvert	curves	nullsink
adelay	dctdnoiz	nullsrc
aecho	decimate	overlay
aeval	dejudger	owdenoise
aevalsrc	delogo	pad
afade	deshake	pan
aformat	drawbox	perms
ainterleave	drawgrid	perspective
allpass	earwax	phase
alphaextract	ebur128	pixdesctest
alphamerge	edgedetect	pp
amerge	elbg	psnr
amix	equalizer	pullup
amovie	extractplanes	removelogo
anull	fade	replaygain
anullsink	field	rgbttestsrc
anullsrc	fieldmatch	rotate
apad	fieldorder	sab

aperms	format	scale
aphaser	fps	select
aresample	framepack	sendcmd
aselect	framestep	separatefields
asendcmd	geq	setdar
asetnsamples	gradfun	setfield
asetpts	haldclut	setpts
asetrate	haldclutsrc	setsar
asettb	hflip	settb
ashowinfo	highpass	showinfo
asplit	histeq	showspectrum
astats	histogram	showwaves
astreamsync	hqdn3d	silencedetect
atempo	hue	sine
atrim	idet	smartblur
avectorscope	il	smptebars
bandpass	interlace	smptehdbars
bandreject	interleave	split
bass	join	spp
bbox	kerndeint	stereo3d
biquad	life	super2xsai
blackdetect	lowpass	swapuv
blackframe	lut	telecine
blend	lut3d	testsrc
boxblur	lutrgb	thumbnail
cellauto	lutyuv	tile
channelmap	mandelbrot	tinterlace
channelsplit	mcdeint	transpose
color	mergeplanes	treble
colorbalance	movie	trim
colorchannelmixer	mp	unsharp
colormatrix	mpdecimate	vflip
compand	mptestsrc	vignette
concat	negate	volume
copy	noformat	volumedetect
crop	noise	w3fdif
cropdetect	null	yadif

Enabled bsfs:

aac_adtstoasc	imx_dump_header	mp3_header_decompress
chomp	mjpeg2jpeg	noise
dump_extradata	mjpega_dump_header	remove_extradata
h264_mp4toannexbmov2textsub		text2movsub

Enabled indevs:

dv1394	lavfi	v412
fbdev	oss	

Enabled outdevs:

fbdev	oss	v412
-------	-----	------