

全志 H3 硬件编解码库

一、需求来源

公司硬件产品使用了 [NanoPi NEO Core](#) 这样一款基于 [全志 H3](#) 的核心板。由于项目中涉及到视频直播、视频录制，且要求添加 OSD 文字信息，故需要视频编解码操作。而使用软件解码仅能满足 1280×720@25fps 解码性能，且解码时间随图像复杂度波动，再使用软件编码就跑不动 25 帧了，只能寻求视频硬件编解码解决方案。

二、调试经历

1. 内核移植及 Ubuntu Core 移植

由于友善之臂为 NanoPi NEO Core 提供的为 [Linux 4.X](#) 的内核并不支持硬件编解码，遂从 [原厂 BSP](#) 并参考官方文档及 [友善之臂提供的硬件资料](#) 移植了 3.4.39 内核。

```
root@AGS:~# uname -a
Linux AGS 3.4.39-h3 #1 SMP PREEMPT Fri Nov 9 08:19:44 UTC 2018 armv7l armv7l armv7l GNU/Linux
root@AGS:~#
```

www.go2aaron.com

移植原厂 BSP 的根本原因主要是需要内核驱动支持：/dev/ion、/dev/cedar_dev

```
root@AGS:~# ls /dev/
android_adb      loop5          tty0           tty42          urandom
apm_bios         loop6          tty1           tty43          usb_accessory
autofs           loop7          tty10          tty44          usbdev1.1
block           loop-control   tty11          tty45          usbdev2.1
bus             mapper         tty12          tty46          usbdev3.1
cachefiles       media0         tty13          tty47          usbdev3.2
cedar_dev        media1         tty14          tty48          usbdev4.1
cna             media2         tty15          tty49          usbdev4.2
console          mem            tty16          tty5           usbdev4.3
cpu_dma_latency mmcblk0        tty17          tty50          usbdev4.4
cuse             mmcblk0boot0   tty18          tty51          usbdev5.1
disk            mmcblk0boot1   tty19          tty52          usbdev5.2
disp            mmcblk0p1      tty2           tty53          v4l
fb0             mmcblk0p2      tty20          tty54          vcs
fb1             mtp_usb        tty21          tty55          vcs1
fb2            net             tty22          tty56          vcs2
fb3            network_latency tty23          tty57          vcs3
fb4            network_throughput tty24          tty58          vcs4
fb5            null           tty25          tty59          vcs5
fb6            ppp            tty26          tty6           vcs6
fb7            psaux          tty27          tty60          vcsa
fd             ptmx           tty28          tty61          vcsa1
full           pts            tty29          tty62          vcsa2
fuse           random         tty3           tty63          vcsa3
hdm            rfkill         tty30          tty7           vcsa4
i2c-0          rtc            tty31          tty8           vcsa5
i2c-2          rtc0           tty32          tty9           vcsa6
initctl        shm            tty33          ttyGS0         video0
input          snd            tty34          ttyGS1         video1
ion            spidev0.0      tty35          ttyGS2         video2
kmsg           stderr         tty36          ttyGS3         video3
log            stdin          tty37          ttyS0          video4
loop0          stdout         tty38          ttyS1          video5
loop1          sunxi_pwm      tty39          ttyS2          vmouse
loop2          sunxi-reg      tty4           tv             watchdog
loop3          sunxi_soc_info tty40          uhid           xt_qtaguid
loop4          tty            tty41          uinput         zero
```

在调试过程中发现提供的编解码库会引起应用程序段错误，最终确认是 gcc 库版本不兼容导致。

重新移植最新 Ubuntu Core 18.04 并使用 gcc 4.9 版本工具链后解决了该问题。

2、编解码库调试

尝试了不同的编解码库,建立了多个工程进行调试(多得自己已经记不清哪些版本😭,一张图足以说明问题)。

<https://github.com/linux-sunxi/libvdpau-sunxi>

坎坷的硬件编解码器调试之路...

<https://github.com/rosimildo/videoenc>

https://github.com/uboborov/ffmpeg_h264_H3

输入为NV12, 有缺陷, 不可解复用,
码率不可控(限制不住)。

仅支持H264编码

<https://github.com/ebutera/cedar-h264enc>

<https://github.com/fsebentley/CedarX-12.06.2015>

https://github.com/uboborov/h264_encoder_H3

<https://github.com/jemk/cedrus>

<https://github.com/huceke/libcedar>

实测: 输入为NV12, 有缺陷, 不可解复用。
两种编码模式, 动态码率不可控, 码率较大;
固定码率容易出现马赛克。

仅支持H264编码

<https://github.com/sailfish009/libvdpau-sunxi>

<https://github.com/free-electrons/libva-cedrus>

<https://github.com/jonsmirl/cam>

www.go2aaron.com

<https://github.com/allwinner-zh/media-codec>

<https://github.com/rosimildo/videoenc>

<https://github.com/hochiang/peephole-surveillance-camera>

OK, 编码质量较优。亲测解码可解复用。

实测: 支持4×720P@30fps解码/

https://github.com/FREEWING-JP/OrangePi_CedarX

1×1080p@60fps解码;

1x1080P@30fps编码。

https://github.com/mikhailk_2012/sunxi-cedarx2

实测支持MJPEG/H264编码
, MJPEG/H264解码。

<https://github.com/jernejsk/OpenELEC-OPi2/tree/openelec-7.0/packages/multimedia>

三、解决方案

全志 H3 硬件编解码库代码仓库地址:

https://github.com/FREEWING-JP/OrangePi_CedarX

实测支持 4×720P@30fps MJPEG 解码(解复用), 1×1080P@60fps MJPEG/H264 解码。

实测支持 1×1080P@30fps MJPEG/H264 编码, 多路低分辨率编码未测试验证。

实测解码输出支持指定图像格式：YV12、NV21、全志定义的 MB32_420 格式，并支持图像缩放。

```
61 enum EPIXELFORMAT
62 {
63     PIXEL_FORMAT_DEFAULT           = 0, //默认4
64
65     PIXEL_FORMAT_YUV_PLANER_420   = 1, //FAIL->7
66     PIXEL_FORMAT_YUV_PLANER_422   = 2, //FAIL->7
67     PIXEL_FORMAT_YUV_PLANER_444   = 3, //FAIL->7
68     解码支持的输出YUV图像格式列表: YV12、NV21、MB32_420
69     PIXEL_FORMAT_YV12              = 4, //OK
70     PIXEL_FORMAT_NV21              = 5, //OK
71     PIXEL_FORMAT_NV12              = 6, //FAIL->7
72     PIXEL_FORMAT_YUV_MB32_420     = 7, //OK
73     PIXEL_FORMAT_YUV_MB32_422     = 8, //FAIL->7
74     PIXEL_FORMAT_YUV_MB32_444     = 9, //FAIL->7
75
76     PIXEL_FORMAT_RGBA              = 10, //FAIL->7
77     PIXEL_FORMAT_ARGB              = 11, //FAIL->7
78     PIXEL_FORMAT_ABGR              = 12, //FAIL->7
79     PIXEL_FORMAT_BGRA              = 13, //FAIL->7
80     设置失败，均输出为MB32_420图像格式
81     PIXEL_FORMAT_YUYV              = 14, //FAIL->7
82     PIXEL_FORMAT_YVYU              = 15, //FAIL->7
83     PIXEL_FORMAT_UYVY              = 16, //FAIL->7
84     PIXEL_FORMAT_VYUY              = 17, //FAIL->7
85     www.go2aaron.com
86     PIXEL_FORMAT_PLANARUV_422     = 18, //FAIL->7
87     PIXEL_FORMAT_PLANARVU_422     = 19, //FAIL->7
88     PIXEL_FORMAT_PLANARUV_444     = 20, //FAIL->7
89     PIXEL_FORMAT_PLANARVU_444     = 21, //FAIL->7
90
91     PIXEL_FORMAT_MIN = PIXEL_FORMAT_DEFAULT,
92     PIXEL_FORMAT_MAX = PIXEL_FORMAT_PLANARVU_444,
93 };
94
```

实测编码输入支持图像格式：YUV420P、YUYV422（部分未测试或未记录）。

```

174 typedef enum VENC_PIXEL_FMT {
175     VENC_PIXEL_YUV420SP,
176     VENC_PIXEL_YVU420SP,
177     VENC_PIXEL_YUV420P,
178     VENC_PIXEL_YVU420P,
179     VENC_PIXEL_YUV422SP,
180     VENC_PIXEL_YVU422SP,
181     VENC_PIXEL_YUV422P,
182     VENC_PIXEL_YVU422P,
183     VENC_PIXEL_YUYV422,
184     VENC_PIXEL_UYVY422,
185     VENC_PIXEL_YVYU422,
186     VENC_PIXEL_VYUY422,
187     VENC_PIXEL_ARGB,
188     VENC_PIXEL_RGBA,
189     VENC_PIXEL_ABGR,
190     VENC_PIXEL_BGRA,
191     VENC_PIXEL_TILE_32X32,
192     VENC_PIXEL_TILE_128X32,
193 } VENC_PIXEL_FMT;
194

```

编码支持的输入图像格式

www.go2aaron.com

编码例程请参考[项目目录 /cedarx/demo/demoVencoder/EncoderTest.c](#)

解码例程请参考[项目目录 /cedarx/demo/demoVdecoder/demoVdecoder.c](#)

同编同解例程请参考[项目目录 /cedarx/demo/demoVencVdec/EncoderDecoderTest.c](#)

四、作品秀

早期单摄像头版本画面。



改进后三摄像头合成画面。



欢迎有河道水质监测及排污口扫描需求的客户来我司参观、交流，寻求合作。

有相关嵌入式软件开发项目亦可联系本人外包。 😊

最后修改：2019 年 02 月 25 日 11 : 42 PM

[上一篇](#) 16 条评论

[下一篇](#)



庄生梦

June 9th, 2020 at 03:19 pm

CdcMemOpen 是定义在什么地方呢？是内核3.4版本独有的吗？我的是全志的3.10。已经有现成的驱动节点 /dev/ion 和/dev/cedar_dev。到

还是交叉编译的差异。使用的全志SDK中提供的 arm-buildroot-linux-gnueabi-hf-gcc。希望得到你的帮助。

[回复](#)



zhuhuan

June 2nd, 2020 at 12:08 pm

您好，我遇到一个问题运行jpegdemo 是没有问题的 运行其他例子就会包这个错误error : ionAlloc : ION_IOC_ALLOC error

```
error : cedarc : pSbmBuf == NULL.
error : cedarc : h264Context->pBSManager == NULL
error : cedarc : BitStreamManager == NULL.
```

error : cedarc : BitStreamManager == NULL.

[回复](#)



sshenn

March 25th, 2020 at 11:13 pm

求教下是否可以在H5上做呢

[回复](#)



sshenn

March 25th, 2020 at 11:15 pm

[@sshenn](#) 求教下老板，用了硬编码，芯片发热量是否有降低？相对于用x264库来做的话

[回复](#)



何亚斌 博主

June 1st, 2020 at 07:31 pm

[@sshenn](#) 硬件编解码产热也比较大。我没做过测试，这个应该和编解码的分辨率，图像复杂度都相关的，也不能一概而论。

[回复](#)



okshall

February 29th, 2020 at 02:08 pm

你好，参考博主帖子，实现了硬件编解码，但是想请问下多图层显示怎么实现的啊，谢谢！

[回复](#)



潘晓宇

January 5th, 2020 at 11:31 pm

您好，请教一下，我在运行jpegdemo的时候会提示open /dev/ion failed 和 /dev/cedar_dev failed. 请问是什么原因呢。我用的是nano pi 给的交叉编译工具（arm-linux-gnueabi），用的libcedarx是gnueabi_3.10，谢谢

[回复](#)



何亚斌 博主

June 1st, 2020 at 07:29 pm

[@潘晓宇](#) 确认一下/dev/ion和/dev/cedar_dev设备节点是否存在，如果不存在则说明使用的内核里面没有开启/集成ion和cedar驱动，可以使用官方的BSP内核。

[回复](#)



小陈

December 4th, 2019 at 06:36 pm

该评论仅登录用户及评论双方可见

[回复](#)



刘根淼

August 8th, 2019 at 04:53 pm

该评论仅登录用户及评论双方可见

[回复](#)

danile



[回复](#)



```
error : cedarc : format '101' support!  
error : cedarc : unsupported format MJPEG  
error : cedarc : create video engine fail.
```

[回复](#)

```
AddVDPlugin();  
pVideo = CreateVideoDecoder();
```

回复



@何亚斌 实测还是会出现以上错误，请问是怎么回事呢

[回复](#)

@vector 确认一下你的库（libcdc_vd_xxx.so的那些库）有没有加到系统目录。调用AddVDPlug in();时会看到加载解码库的信息。将库文件拷贝到/usr/local/lib/cedarx目录下面。在/etc/ld.so.conf.d目录下面创建cedarx.conf文件，里面写入内容为你的库路径，即/usr/local/lib/cedarx。然后再运行ldconfig命令。

回复

[illegible]