APPS049-Build FFmpeg Java API Library using JavaCPP Application Note

V2.0

2021-12-10

Table of Contents

1 Table of Abbreviations

Abbreviation:	Full Form:
T408	Codensity T408 Video Transcoder
T432	Codensity T432 Video Transcoder
AI	Artificial Intelligence
API	Application Programming Interface
AUD	Access Unit Delimiters
AV	Audio Video
AV1	Alliance Open Media Video 1 Codec
CABAC	Context Adaptive Binary Arithmetic Coding
CBR	Constant Bit Rate
CPU	Central Processing Unit
CRF	Constant Rate Factor
DSP	Digital Signal Processing
EOF	End Of File
FIXQB	Fixed Quantization Parameter
FF	Form Factor
FPS	Frames Per Second
GOP	Group of pictures
HDR	High Dynamic Range
HHHL	Half Height, Half Length
HHFL	Half Height, Full Length
HLG	Hybrid Log Gamma
HRD	Hypothetical Reference Decoder
IE	Inference Engine
Mbps	Mega Bit per second
MBps	Mega Byte per second
NLP	Natural Language Processing
NPU	Neural Processing Unit
NVMe	Non-Volatile Memory express
PCIe	Peripheral Component Interconnect express
PPS	Picture Parameter Set
QP	Quoted Printable
RGB	Red Green Blue
RGBA	Red Green Blue Alpha
RDO	Optimization
ROI	Region of interest
SDK	Software Development Kit

SEI	Supplemental Enhancement Information
SRIOV	Single Root I/O Virtualization
SPS	Sequence Parameter Set
VBR	Variable Bit Rat
VCL	Video Coding Layer
VPS	Video Parameter Set
VUI	Video Usability Information

2 References

Technical Notes:
- Codensity T408 & T432 Massif Quickstart Guide

Specifications:

3 Background

This document details how to build FFmpeg Java API library integrating NETINT Transcoders using JavaCPP.

3.1 Intended Audience

This document is intended for engineers/developers/technicians operating NETINT transcoders.

3.2 Compatibility

T408 / T432 Product	**Version**
Minimum Hardware Revision	2.5.F
Minimum Firmware Revision	2.5.F
Minimum FFmpeg Revision	Only support: 4.3.2、4.3.1、4.2.1、4.1.3、3.4.2、2.7.1

4 Setup software

Run the following command to install prerequisite packages:

```
   $ sudo apt-get install -y unrar rar javac* openjdk-8-jdk curl
maven cmake zip unzip
```

Check version of maven

```
      $ mvn -v
Maven version: 3.0 or later
```

5 Build NETINT Libxcoder

Follow 《QuickStartGuideT408_T432_FW2_5.pdf》 3.3 Operating Systems and Software.

Use NETINT firmware v2.5.F and libxcoder v2.5.K for demo test.

6 Build FFmpeg Java API library

6.1 Download javacpp-presets

The javacpp-presets version is strongly correlated with FFmpeg version. Their relationship is as follows：

javacpp-presets	-->	ffmpeg
1.5.5	-->	ffmpeg 4.3.2
1.5.4	-->	ffmpeg 4.3.1
1.5.2	-->	ffmpeg 4.2.1
1.5.1	-->	ffmpeg 4.1.3
1.4.1	-->	ffmpeg 3.4.2
1.0	-->	ffmpeg 2.7.1

We use javacpp-presets 1.5.2 and ffmpeg 4.2.1 for demo test.

Download command:

```
      $ sudo mkdir ./javacpp-presets
```

```
$ git clone -b 1.5.2 --depth=1
https://github.com/bytedeco/javacpp-presets.git ./javacpp-presets-1.5.2
```

If the host is in Chinese mainland, use this

```
$ git clone -b 1.5.2 --depth=1
https://gitee.com/mirrors/javacpp-presets.git ./javacpp-presets-1.5.2
```

6.2 Edit file cppbuild.sh

```
$ cd ./javacpp-presets-1.5.2/
$ vim ./ffmpeg/cppbuild.sh
```

Edit file ./ffmpeg/cppbuild.sh

Modify line 11:

ENABLE="--enable-shared --enable-gpl --enable-version3 --enable-nonfree --enable-runtime-cpudetect --enable-zlib --enable-libmp3lame --enable-libspeex --enable-libopencore-amrnb --enable-libopencore-amrwb --enable-libvo-amrwbenc --enable-openssl --enable-libopenh264 --enable-libvpx --enable-libfreetype --enable-libopus --enable-gpl --enable-nonfree --enable-libxcoder --enable-encoders --enable-decoders --enable-avfilter --enable-muxers --enable-demuxers --enable-parsers"

```
DISABLE="--disable-iconv --disable-opencl --disable-sdl2 --disable-bzlib --disable-lzma --disable-linux-perf"
ENABLE="--enable-shared --enable-gpl --enable-version3 --enable-nonfree --enable-runtime-cpudetect --enable-zlib --enable-libmp3lame --enable-libspeex --enable-
libopencore-amrnb --enable-libopencore-amrwb --enable-libvo-amrwbenc --enable-openssl --enable-libopenh264 --enable-libvpx --enable-libfreetype --enable-libopus
 --enable-gpl --enable-nonfree --enable-libxcoder --enable-encoders --enable-decoders --enable-avfilter --enable-muxers --enable-demuxers --enable-parsers"

# minimal configuration to support MPEG-4 streams with H.264 and AAC as well as Motion JPEG
# DISABLE="--disable-iconv --disable-libxcb --disable-opencl --disable-sdl2 --disable-bzlib --disable-lzma --disable-linux-perf --disable-everything"
# ENABLE="--enable-shared --enable-runtime-cpudetect --enable-libopenh264 --enable-encoder=libopenh264 --enable-encoder=aac --enable-encoder=mjpeg --enable-deco
der=h264 --enable-decoder=aac --enable-decoder=mjpeg --enable-parser=h264 --enable-parser=aac --enable-parser=mjpeg --enable-muxer=mp4 --enable-muxer=rtsp --ena
ble-muxer=mjpeg --enable-demuxer=mov --enable-demuxer=rtsp --enable-demuxer=mjpeg --enable-protocol=file --enable-protocol=http --enable-protocol=rtp --enable-p
rotocol=rtmp"
```

Line 54 add:

sudo rm -rf ./ffmpeg-${FFMPEG_VERSION}/

```
mkdir -p $PLATFORM
cd $PLATFORM
INSTALL_PATH=`pwd`
sudo rm -rf ./ffmpeg-${FFMPEG_VERSION}/
echo "Decompressing archives..."
tar --totals -xzf ../nasm-$NASM_VERSION.tar.gz
tar --totals -xzf ../$ZLIB.tar.gz
```

Line 750 add:

```
        cp ../../../../../javacpp-presets-ffmpeg-n${FFMPEG_VERSION}_t4xx_patch ./
        patch -t -p 1 < javacpp-presets-ffmpeg-n${FFMPEG_VERSION}_t4xx_patch
        cp ../../../../../javacpp-presets-ffmpeg-n${FFMPEG_VERSION}-configure_patch ./
        patch -t -p 1 < javacpp-presets-ffmpeg-n${FFMPEG_VERSION}-configure_patch
```

```
        fi
        cd ../nv-codec-headers-n$NVCODEC_VERSION
        make install PREFIX=$INSTALL_PATH
        cd ../ffmpeg-$FFMPEG_VERSION
        cp ../../../../../javacpp-presets-ffmpeg-n${FFMPEG_VERSION}_t4xx_patch ./
        patch -t -p 1 < javacpp-presets-ffmpeg-n${FFMPEG_VERSION}_t4xx_patch
        cp ../../../../../javacpp-presets-ffmpeg-n${FFMPEG_VERSION}-configure_patch ./
        patch -t -p 1 < javacpp-presets-ffmpeg-n${FFMPEG_VERSION}-configure_patch
        LDEXEFLAGS='-Wl,-rpath,\$$ORIGIN/' PKG_CONFIG_PATH=../lib/pkgconfig/ ./configure --prefix=.. $DISAB
c --enable-pthreads --enable-libxcb --cc="gcc -m64" --extra-cflags="-I../include/" --extra-ldflags="-L../li
BS"
```

Save and exit edit file ./ffmpeg/cppbuild.sh

6.3 Copy patch file

Copy patch file to dir ./javacpp-presets/

javacpp-presets-ffmpeg-n4.2.1-configure_patch

javacpp-presets-ffmpeg-n4.2.1_t4xx_patch

These two patch files are very important patch. This patch integrates NETINT transcoders to ffmpeg lib.

6.4 Make and install ffmpeg lib

Make lib

```
$ cd ./javacpp-presets-1.5.2/
$ sudo mvn install --projects ffmpeg -Djavacpp.platform=linux-x86_64
```

Install lib

```
$ cd ./javacpp-presets-1.5.2/platform/

$ sudo mvn install --projects ../ffmpeg/platform -
Djavacpp.platform.host
```

6.5 Test ffmpeg java lib

Unzip test.zip

test.zip

```
$ cd test/demo/

$ sudo mvn clean compile exec:java -
Dexec.mainClass=Transcoding
```

```
[INFO] Scanning for projects...
[INFO]
[INFO] ----------------------< org.bytedeco.ffmpeg:demo >----------------------
[INFO] Building demo 1.5.2
[INFO] --------------------------------[ jar ]---------------------------------
[INFO]
[INFO] --- maven-clean-plugin:2.5:clean (default-clean) @ demo ---
[INFO] Deleting /home/ni/test/demo/target
[INFO]
[INFO] --- maven-resources-plugin:2.6:resources (default-resources) @ demo ---
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory /home/ni/test/demo/src/main/resources
[INFO]
[INFO] --- maven-compiler-plugin:3.1:compile (default-compile) @ demo ---
[INFO] Changes detected - recompiling the module!
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 1 source file to /home/ni/test/demo/target/classes
[INFO]
[INFO] --- exec-maven-plugin:3.0.0:java (default-cli) @ demo ---
start transcoding......
Input #0, mov,mp4,m4a,3gp,3g2,mj2, from '/home/ni/test/output_9.mp4':
  Metadata:
    major_brand     : isom
    minor_version   : 512
    compatible_brands: isomiso2avc1mp41
    encoder         : Lavf58.45.100
  Duration: 00:00:20.88, start: 0.000000, bitrate: 3957 kb/s
    Stream #0:0(und): Video: h264 (High) (avc1 / 0x31637661), yuv420p(tv), 1280x720, 3954 kb/s, 24 fps, 24 tbr, 12288 tbn, 48 tbc (default)
    Metadata:
      handler_name    : VideoHandler
///////----null CODEC_ID = 173
[h265_ni_enc @ 0x7f62d478cd00] Session state: 0 allocate frame fifo.
[h265_ni_enc @ 0x7f62d478cd00] dts offset: 2, gop_offset_count: 0
Output #0, mp4, to '/home/ni/test/output_9_recode.mp4':
  Metadata:
    encoder         : Lavf58.29.100
    Stream #0:0: Video: hevc (hev1 / 0x31766568), yuv420p, 1280x720, q=2-31, 3954 kb/s, 24 fps, 12288 tbn, 24 tbc
[h264_ni_dec @ 0x7f62d478d380] The deprecated avcodec_decode_* API cannot return all the frames for this decoder. Some frames will be dropp
to the new decoding API to fix this.
Encoder HW[0] INST[127]-average usage:29%
Decoder HW[0] INST[127]-average usage:24%
/home/ni/test/output_9_recode.mp4
[INFO] ------------------------------------------------------------------------
[INFO] BUILD SUCCESS
[INFO] ------------------------------------------------------------------------
[INFO] Total time:  4.650 s
[INFO] Finished at: 2021-10-09T11:55:23+08:00
[INFO] ------------------------------------------------------------------------
```

7 How to make the file javacpp-presets-ffmpeg-n4.2.1_t4xx_patch

Proceed by modifying FFmpeg-n4.2.1_t4xx_patch files in the Codensity T4XX Software Release package.

- Delete the whole content about 'build_ffmpeg.sh' and 'configure' in FFmpeg-n4.2.1_t4xx_patch.
- Modify the content about 'allcodecs' in FFmpeg-n4.2.1_t4xx_patch.

```
diff -crBN -x .git -x .gitattributes -x .gitignore
base_ffmpeg_n4.2.1/libavcodec/allcodecs.c FFmpeg-n4.2.1/libavcodec/allcodecs.c
*** base_ffmpeg_n4.2.1/libavcodec/allcodecs.c      2020-12-09 02:15:25.172928356 -
0800
--- FFmpeg-n4.2.1/libavcodec/allcodecs.c            2020-03-05 14:37:18.361803289 -
0800
***************
*** 136,141 ****
--- 136,145 ----
 extern AVCodec ff_h263p_encoder;
 extern AVCodec ff_h263p_decoder;
 extern AVCodec ff_h263_v4l2m2m_decoder;
+extern AVCodec ff_h264_ni_decoder;
+extern AVCodec ff_h265_ni_decoder;
+extern AVCodec ff_h264_ni_encoder;
+extern AVCodec ff_h265_ni_encoder;
 extern AVCodec ff_h264_decoder;
 extern AVCodec ff_h264_crystalhd_decoder;
 extern AVCodec ff_h264_v4l2m2m_decoder;
```

```
    { "aframes",        OPT_AUDIO | HAS_ARG  | OPT_PERFILE | OPT_OUTPUT,           { .func_arg = opt_audio_frames },
diff -crBN -x .git -x .gitattributes -x .gitignore base_ffmpeg_n4.2.1/libavcodec/allcodecs.c FFmpeg-n4.2.1/libavcodec/allcodecs.c
*** base_ffmpeg_n4.2.1/libavcodec/allcodecs.c   2020-12-09 02:15:25.172928356 -0800
--- FFmpeg-n4.2.1/libavcodec/allcodecs.c        2020-03-05 14:37:18.361803289 -0800
***************
*** 136,141 ****
--- 136,145 ----
 extern AVCodec ff_h263p_encoder;
 extern AVCodec ff_h263p_decoder;
 extern AVCodec ff_h263_v4l2m2m_decoder;
+extern AVCodec ff_h264_ni_decoder;
+extern AVCodec ff_h265_ni_decoder;
+extern AVCodec ff_h264_ni_encoder;
+extern AVCodec ff_h265_ni_encoder;
 extern AVCodec ff_h264_decoder;
 extern AVCodec ff_h264_crystalhd_decoder;
 extern AVCodec ff_h264_v4l2m2m_decoder;
diff -crBN -x .git -x .gitattributes -x .gitignore base_ffmpeg_n4.2.1/libavcodec/hevc_mp4toannexb_bsf.c FFmpeg-n4.2.1/libavcodec/hevc_
```

8 Revision History

Version	Change	Create Date	Author
1.0	Document Created	October 09, 2021	Long Han
1.1	Minor update to template	October 19, 2021	Balwinder Sidhu
2.0	Update as per new app note template	December 10, 2021	Balwinder Sidhu

9 Legal Notice