APPS042-Codensity T408™ Encoder Latency Measurement Procedure Application Note

V2.0

2022-10-28

Table of Contents

1 Table of Abbreviations

Abbreviation:	Full Form:
T408	Codensity T408 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:

Specifications:

3 Background

This document demonstrates the collection and calculation of encoder latency for T408 product.

3.1 Intended Audience

This document is intended to help engineers/technicians/developers operating NETINT transcoders better understand the encoder latency performance.

3.2 Compatibility

Software Compatibility

This guide is intended to be used with NETINT Codensity T408 Video Transcoder software Release 2.5.0 or newer.

Hardware Compatibility

Release 2.5.0 supports NETINT Codensity T408 Video Transcoder hardware.

Operation system

All OS supporting T408 Video Transcoder.

4 Encoder Latency Measurement

4.1 Scope

This section will demonstrate the collection of latency measurement data for T408.

The T408 and FFmpeg environment configuration and usage is not in the scope of this document. The document assumes a Linux host installed with a T408 card and with libxcoder and FFmpeg successfully compiling and T408 demonstrated for video transcoding operation.

4.2 Using libxcoder latency logs

This method includes enabling print outs for encoder latency time and then parsing the log data to determine the latency measurement.

4.2.1 Build libxcoder with -p flag

Run the libxcoder build with the -p flag for latency patch.

```
sh build.sh -p
```

```
sudo make install
```

Build FFmpeg-n4.2.1, for example, as usual.

4.2.2 Collect eLAT data

Run the following ffmpeg encode command with a yuv file input. It will generate an output log file.

```
ffmpeg -y -nostdin -hide_banner -vsync 0 -stream_loop 10 -f
rawvideo -pix_fmt yuv420p -s:v 1920x1080 -i
/mnt/ramdisk/dinner_1920x1080p30_300_br7500_b3__B265d.yuv -c:v
h265_ni_enc -enc 0 -xcoder-params
```

```
lowDelay=1:RCEnable=1:bitrate=6000000:gopPresetIdx=2 -f null -
2>&1 |ts '%.s' > output-y2h-0-0-0.log
```

The output log file will have extra log messages as follows.

```
1631827540.836110 DTS:8,DELTA:9275118,eLAT:7598042;
1631827540.845717 DTS:9,DELTA:9617610,eLAT:7951976;
1631827540.855171 DTS:10,DELTA:9453970,eLAT:7739550;
1631827540.864634 DTS:11,DELTA:9457717,eLAT:7736013;
1631827540.873962 DTS:12,DELTA:9324093,eLAT:7616187;
1631827540.883386 DTS:13,DELTA:9420464,eLAT:7724501;
1631827540.892662 DTS:14,DELTA:9277582,eLAT:7618391;
1631827540.902085 DTS:15,DELTA:9423821,eLAT:7742656;
1631827540.911502 DTS:16,DELTA:9419621,eLAT:7737806;
1631827540.920939 DTS:17,DELTA:9426686,eLAT:7767986;
1631827540.930462 DTS:18,DELTA:9529067,eLAT:7755662;
1631827540.939892 DTS:19,DELTA:9433107,eLAT:7738477;
1631827540.949159 DTS:20,DELTA:9269487,eLAT:7609123;
1631827540.958445 DTS:21,DELTA:9286921,eLAT:7595817;
1631827540.967911 DTS:22,DELTA:9463759,eLAT:7757906;
1631827540.977354 DTS:23,DELTA:9444901,eLAT:7757094;
1631827540.986672 DTS:24,DELTA:9318303,eLAT:7635857;
```

```
where,
DTS         Decoding time stamp of frame in the timebase
transmitted to libXcoder
DELTA       Frame period between this frame and previous frame in
nanoseconds
```

```
eLAT        HW encoder latency for frame of this DTS in
nanoseconds
```

4.2.3 Measure Latency

Parse the output log file to collect the eLAT time stamps.

```
cat output-y2h-0-0-0.log | sed 's/;/ /g' | grep "eLAT:" | cut -
d"," -f3 | sed 's/eLAT:/ /g'> encoder-out_time-y2h-0-0-0.log
```

Collate the parsed eLAT data to a csv file.

Calculate the eLAT / 1000000 to get the latency msec.

4.3 Using ffmpeg -debug_ts

This method can be used for collecting latency measurement for T408 and any 3rd party GPU like Nvidia or AMD.

1] Run ffmpeg command to perform encoding operation. Use -c:v parameter to specify video encoder to use like h264_nvenc for Nvidia GPU and h264_ni_enc for NETINT encoder.

T408 example:

```
ffmpeg -y -nostdin -hide_banner -vsync 0 -debug_ts -stream_loop 3
-f rawvideo -pix_fmt yuv420p -s:v 1920x1080 -i
dinner_1920x1080p30_300_br7500_b3__B265d.yuv -c:v h265_ni_enc -
enc 0 -xcoder-params lowDelay=1:RCEnable=1:
bitrate=6000000:gopPresetIdx=2 -f null - 2>&1 |ts '%.s' > output-
y2h-0-0-0.log
```

Nvidia GPU example:

```
ffmpeg -y -nostdin -hide_banner -vsync 0 -debug_ts -stream_loop 3
-f rawvideo -pix_fmt yuv420p -s:v 1920x1080 -i
dinner_1920x1080p30_300_br7500_b3__B265d.yuv -c:v hevc_nvenc -
preset ll -b:v 6M -g 99999 -bf 0 -f null - 2>&1 |ts '%.s' >
output-y2h-0-0-0.log
```

The output log file, for T408 as example, will have extra log messages as follows.

```
1631904294.347905 muxer <- type:video pkt_pts:0 pkt_pts_time:0
pkt_dts:0 pkt_dts_time:0 size:57176

1631904294.348972 demuxer -> ist_index:0 type:video
next_dts:120000 next_dts_time:0.12 next_pts:120000
next_pts_time:0.12 pkt_pts:3 pkt_pts_time:0.12 pkt_dts:3
pkt_dts_time:0.12 off:0 off_time:0

1631904294.349079 demuxer+ffmpeg -> ist_index:0 type:video
pkt_pts:3 pkt_pts_time:0.12 pkt_dts:3 pkt_dts_time:0.12 off:0
off_time:0

1631904294.349118 decoder -> ist_index:0 type:video frame_pts:3
frame_pts_time:0.12 best_effort_ts:3 best_effort_ts_time:0.12
keyframe:1 frame_type:1 time_base:1/25

1631904294.349149 filter -> pts:3 pts_time:0.12 exact:3.000008
time_base:1/25
```

```
1631904294.349183 encoder <- type:video frame_pts:3
frame_pts_time:0.12 time_base:1/25

1631904294.350352 encoder -> type:video pkt_pts:1
pkt_pts_time:0.04 pkt_dts:1 pkt_dts_time:0.04

1631904294.350462     Last message repeated 1 times
```

The output log file, for Nvidia as example, will have extra log messages as follows.

```
1631904530.324457 muxer <- type:video pkt_pts:2 pkt_pts_time:0.08
pkt_dts:2 pkt_dts_time:0.08 size:9606

1631904530.325000 demuxer -> ist_index:0 type:video
next_dts:200000 next_dts_time:0.2 next_pts:200000
next_pts_time:0.2 pkt_pts:5 pkt_pts_time:0.2 pkt_dts:5
pkt_dts_time:0.2 off:0 off_time:0

1631904530.325039 demuxer+ffmpeg -> ist_index:0 type:video
pkt_pts:5 pkt_pts_time:0.2 pkt_dts:5 pkt_dts_time:0.2 off:0
off_time:0

1631904530.325071 decoder -> ist_index:0 type:video frame_pts:5
frame_pts_time:0.2 best_effort_ts:5 best_effort_ts_time:0.2
keyframe:1 frame_type:1 time_base:1/25

1631904530.325103 filter -> pts:5 pts_time:0.2 exact:5.000008
time_base:1/25

1631904530.325129 encoder <- type:video frame_pts:5
frame_pts_time:0.2 time_base:1/25

1631904530.327419 encoder -> type:video pkt_pts:3
pkt_pts_time:0.12 pkt_dts:3 pkt_dts_time:0.12

1631904530.327582     Last message repeated 1 times
```

2] Filter and record latency data from output log

```
cat output-y2h-0-0-0.log | grep "encoder <-" | cut -d" " -f1 >
encoder-in_time-0-0.log && cat output-$tag.log | grep "encoder
->" | cut -d" " -f1 > encoder-out_time-0-0-0.log
```

3] Collate the encoder in time and encoder out time stamps to a csv file.

4] Calculate the output log encoder out and encoder in time stamps diff * 1000 to get the latency msec.

5 Revision History

Version	Change	Create Date	Author
0.1	Document Created	September 15, 2021	Balwinder Sidhu
1.0	Issued version with review comments from Daniel and Achraf.	September 16, 2021	Balwinder Sidhu
1.1	Issued version. Add sample logs.	September 17, 2021	Balwinder Sidhu
1.2	Issued version. Add libxcoder latency patch details.	September 21, 2021	Balwinder Sidhu
1.3	Correction to Section 3.2.3 for latency as eLAT/1000000	October 22, 2021	Balwinder Sidhu
2.0	Update as per new app note template	December 21, 2021	Balwinder Sidhu

6 Legal Notice