NVIDIA VIDEO CODEC SDK APPLICATION NOTE - ENCODER

NVENC_DA-06209-001| April 2017

DOCUMENT CHANGE HISTORY

NVENC_DA-06209-001

Version	Date	Authors	Description of Change
01	Jan 30,2012	AP/CC	Initial release
02	Sept 24, 2012	AP	Updated for NVENC SDK release 2.0
03	April 10, 2013	AP	Updated for Monterey SDK 2.0.0 update
04	Aug 4, 2013	AP	Updated for NVENC SDK release 3.0
05	June 17, 2014	SM/AP	Updated for NVENC SDK release 4.0
06	Nov 14, 2014	SM	Updated for NVENC SDK release 5.0
07	Oct 10, 2015	SM	Updated for Video Codec SDK Release 6.0
08	June 10, 2016	SM	Updated for Video Codec SDK Release 7.0
09	Nov 15, 2016	SM	Updated for Video Codec SDK Release 7.1
10	Apr 11, 2017	SM/AP	Updated for Video Codec SDK Release 8.0

TABLE OF CONTENTS

LIST OF TABLES

NVIDIA HARDWARE VIDEO ENCODER

1. INTRODUCTION

NVIDIA GPUs - beginning with the Kepler generation - contain a hardware-based encoder (referred to as NVENC in this document) which provides fully-accelerated hardware-based video encoding and is independent of graphics performance. With complete encoding (which is computationally complex) offloaded to NVENC, the graphics engine and the CPU are free for other operations. For example, in a game recording scenario, encoding being completely offloaded to NVENC makes the graphics engine fully available for game rendering.

The hardware capabilities available in NVENC are exposed through APIs herein referred to as NVENCODE APIs in the document. This document provides information about the capabilities of the hardware encoder and features exposed through NVENCODE APIs.

2. NVENC CAPABILITIES

NVENC can perform end-to-end encoding for H.264, HEVC 8-bit and HEVC 10-bit. This includes, motion estimation and mode decision, motion compensation and residual coding, and entropy coding. These operations are accelerated in the hardware which is a dedicated block on GPU silicon die. Although the rate control algorithm is implemented in the GPU's firmware, from the application's perspective, rate control is a hardware function controlled via the parameters exposed in the NVENCODE APIs. The hardware is fully programmable and NVENCODE APIs provide the necessary knobs to program the hardware appropriately.

Table 1 summarizes the capabilities of the NVENC hardware exposed through NVENCODE APIs. Table 2 summarizes new encoder features available in Video SDK 8.0.

Table 1. NVENC hardware capabilities

Feature	Description	Kepler GPUs	First generation Maxwell GPUs	Second generation Maxwell GPUs	Pascal GPUs
H.264 baseline, main and high profiles	Capability to encode YUV 4:2:0 sequence and generate a H.264-bit stream.	✓	✓	✓	✓
H.264 4:4:4 encoding	Capability to encode YUV 4:4:4 sequence and generate a H.264-bit stream.	×	✓	✓	✓
H.264 lossless encoding	Lossless encoding.	×	✓	✓	✓
H.264 motion estimation (ME) only mode	Capability to provide macro-block level motion vectors and intra/inter modes.	×	✓	✓	✓
H.264/HEVC weighted prediction	Support for weighted prediction.	×	×	×	✓
Encoding support for H.264 ARGB content	Capability to encode RGB input.	✓	✓	✓	✓
HEVC main profile	Capability to encode YUV 4:2:0 sequence and generate a HEVC bit stream.	×	×	✓	✓
HEVC main10 profile	Support for encoding 10-bit content generate a HEVC bit stream.	×	×	×	✓
HEVC lossless encoding	Lossless encoding.	×	×	×	✓
HEVC sample adaptive offset(SAO)	Improves encoded video quality.	×	×	×	✓

Feature	Description	Kepler GPUs	First generation Maxwell GPUs	Second generation Maxwell GPUs	Pascal GPUs
HEVC 4:4:4 encoding	Capability to encode YUV 4:4:4 sequence and generate a HEVC bit stream.	×	×	×	✓
HEVC motion estimation (ME) only mode	Capability to provide CTB level motion vectors and intra/inter modes.	×	×	×	✓
HEVC 8K encoding*	Support for encoding 8192 × 8192 Content.	×	×	×	✓

*: Supported in select Pascal generation GPUs

Table 2. What's new in SDK 8.0

Feature	Description
Encoding support for OGL surfaces	NVENCODE API can accept OGL surfaces as input directly. The support is availabe only on Linux.
Improved quality for HEVC spatial adaptive quantization	Encoded quality for HEVC spatial adaptive quantization is improved.
Weighted prediction (WP)	WP support is added for H.264, HEVC and HEVC main10. The feature can be enabled through a flag exposed in the NVENCODE API. WP gives significant quality improvement for contents having illumination changes.
External motion hints for H.264 motion estimation (ME) only mode	User can pass external motion hints for H.264 ME only mode.
Fractional constant quality (CQ) support	User can specify fractional values for CQ rate control mode

3. NVENC LICENSING POLICY

There is no change in licensing policy in the current SDK in comparison to the earlier SDK. The licensing policy is as follows:

As far as NVENC hardware encoding is concerned, NVIDIA GPUs are classified into two categories: "qualified" and "non-qualified". On qualified GPUs, the number of concurrent encode sessions is limited by available system resources (encoder capacity, system memory, video memory etc.). On non-qualified GPUs, the number of concurrent encode sessions is limited to 2 per system. This limit of 2 concurrent sessions per system applies to the combined number of encoding sessions executed on all non-qualified cards present in the system.

For a complete list of qualified and non-qualified GPUs, refer to https://developer.nvidia.com/nvidia-video-codec-sdk.

For example, on a system with one Quadro K4000 card (which is a qualified GPU) and three GeForce cards (which are non-qualified GPUs), the application can run *N* simultaneous encode sessions on Quadro K4000 card (where *N* is defined by the encoder/memory/hardware limitations) and two sessions on all the three GeForce cards combined. Thus, the limit on the number of simultaneous encode sessions for such a system is $N + 2$.

4. NVENC PERFORMANCE

With every generation of NVIDIA GPUs (Kepler, Maxwell 1st/2nd gen, Pascal), NVENC performance has increased steadily. Table 3 provides *indicative*[1] NVENC performance on Kepler, Maxwell and Pascal GPUs for different presets and rate control modes (these two factors play major role in determining the performance and quality). Note that performance numbers in Table 3 are measured on GeForce hardware with assumptions listed under the table. The performance varies across GPU classes (e.g. Quadro, Tesla), and scales (almost) linearly with the clock speeds for each hardware.

While Kepler and first generation Maxwell GPUs had one NVENC engine per chip, certain variants of the second-generation Maxwell GPUs and Pascal generation GPUs have two/three NVENC engines per chip. This increases the aggregate encoder performance of the GPU. NVIDIA driver takes care of load balancing among multiple NVENC engines on the chip, so that applications don't require any special code to take advantage of multiple encoders and automatically benefit from higher encoder capacity on higher-end GPU hardware. The encode performance listed in Table 3 is given *per NVENC engine*. Thus, if the GPU has 2 NVENCs (e.g. GP104, GM204), multiply the corresponding number in Table 3 by the number of NVENCs per chip to get aggregate maximum performance (applicable only when running multiple simultaneous encode sessions). Note that performance with single encoding session cannot exceed performance per NVENC, regardless of the number of NVENCs present on the GPU.

[1] Encoder performance depends on many factors, including but not limited to: Encoder settings, GPU clocks, GPU type, video content type etc. Performance reported in SDK 7.1 and earlier SDK versions was measured using content which typically yields higher fps. Starting SDK 8.0, we are reporting average performance between best & worst case content.

NVENC hardware natively supports multiple hardware encoding contexts with negligible context-switching penalty. As a result, subject to the hardware performance limit and available memory, an application can encode multiple videos simultaneously. The hardware and software maintain the context for each encoding session, allowing many simultaneous encoding sessions to run in parallel.

NVENCODE API exposes several presets, rate control modes and other parameters for programming the hardware. A combination of these parameters enables video encoding at varying quality and performance levels. In general, one can trade performance for quality and vice versa.

Table 3. NVENC encoding performance

			H.264 (frames/second)				HEVC (frames/second)	
Preset	**Rate Control Mode**		**Kepler** 540 MHz	**First Gen. Maxwell** 1032 MHz	**Second Gen. Maxwell** 1366 MHz	**Pascal** 1911 MHz	**Second Gen. Maxwell** 1366 MHz	**Pascal** 1911 MHz
High Performance	Constant QP		219	361	396	633	214	388
	Single Pass		218	360	393	639	214	391
	Dual Pass		114	246	266	450	152	285
High Quality	Constant QP		76	216	270	384	152	243
	Single Pass		77	216	267	384	156	252
	Dual Pass	min	57	133	185	261	91	154
		max	57	182	221	348	114	194
Low latency High Performance	Constant QP		142	244	343	529	214	389
	Single Pass		142	244	344	524	216	392
	Dual Pass		91	201	244	414	152	284
Low latency High Quality	Constant QP		76	230	269	380	216	389
	Single Pass		76	230	271	380	217	392
	Dual Pass	min	57	146	191	362	118	216
		max	57	198	224	362	150	284
Lossless				277	327	532		257

Assumptions:

- Resolution/Input Format/Bit depth: 1920 × 1080/YUV 4:2:0/8-bit
- Hardware: Various GeForce GPU hardware with clocks held at P0, Intel Core i7-6700 CPU @ 3.40 GHz,
- GPU Clocks: GPU core clock reported by GPU-Z, as specified in the table
- Software: Windows 10, Video SDK 8.0, NVIDIA display driver: 378.99
- Dual pass performance varies depending upon other settings such as look-ahead, B-frames, VBV buffer size etc. Hence max and min performance is specified

5. PROGRAMMING NVENC

Video Codec SDK 8.0 is supported on R378 drivers and above. Please refer to the SDK release notes for information regarding the required driver version.

Please refer to the documents and the sample applications included in the SDK package for details on how to program NVENC.

6. FFMPEG AND LIBAV SUPPORT

FFmpeg and Libav are the most popular multimedia transcoding tools used extensively for video and audio transcoding.

The video hardware accelerators in NVIDIA GPUs can be effectively used with FFmpeg and Libav to significantly speed up the video decoding, encoding and end-to-end transcoding at very high performance.

Note that FFmpeg and Libav are open-source projects and their usage is governed by specific licenses and terms and conditions for each of these projects.

Notice

Trademarks

Copyright