



# NVIDIA VIDEO CODEC SDK APPLICATION NOTE - ENCODER

NVENC\_DA-06209-001\_v08 | June 2016

## Application Note

## DOCUMENT CHANGE HISTORY

NVENC\_DA-06209-001\_v08

Version	Date	Authors	Description of Change	Highlights
01	Jan 30,2012	AP/CC	Initial release	Initial Support for Kepler NVENC
02	Sept 24, 2012	AP	Updated for NVENC SDK release 2.0	Additional features on Kepler NVENC
03	April 10, 2013	AP	Updated for Monterey SDK 2.0.0 update	Additional features on Kepler NVENC
04	Aug 4, 2013	AP	Updated for NVENC SDK release 3.0	Support for additional software features
05	June 17, 2014	SM/AP	Updated for NVENC SDK release 4.0	Software Support for First generation Maxwell GPUs
06	Nov 14, 2014	SM	Updated for NVENC SDK release 5.0	Software Support for Second generation Maxwell GPUs
07	Oct 10, 2015	SM	Updated for Video Codec SDK Release 6.0	Support for additional software features
08	June 10, 2016	SM	Updated for Video Codec SDK Release 7.0	Support for Pascal GPUs

# TABLE OF CONTENTS

**NVIDIA Hardware Video ENCODER ..... 4**

- 1. Introduction..... 4
- 2. NVENC Capabilities ..... 5
- 3. NVENC Licensing Policy ..... 7
- 4. NVENC Performance..... 8
- 5. Programming NVENC.....10

# LIST OF TABLES

Table 1. NVENC Hardware Capabilities..... 5

Table 2. Feature additions in the current SDK ..... 6

Table 3. NVENC Encoding Performance ..... 9

# 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 bandwidth fully available for game rendering.

In the current SDK, support for NVENC on Pascal generation GPUs along with support for several additional features in H.264 and HEVC have been added.

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. The current document *only* highlights the changes in the current Video Codec SDK package with respect to the previous SDK packages. In order to know about the features exposed in earlier SDKs please refer to the earlier SDK package(s). Any driver supporting SDK 7.0 is completely *backward compatible* with earlier SDKs, which means that applications compiled with earlier NVENC header version(s) can be expected to work “as-is” with the driver supporting SDK 7.0 and beyond.

## 2. NVENC CAPABILITIES

NVENC can perform all tasks that are needed for end-to-end H.264 and HEVC encoding. The rate control algorithm is implemented in the GPU's firmware and controlled via the driver. From the application's perspective, rate control is a hardware function controlled via the parameters exposed in the NVENCODE APIs. The hardware also provides the ability to use external motion estimation engine (to feed external Motion hints) and custom quantization parameter (QP) maps (for "Region of Interest" encoding). The region of interest encoding has been made available using the "QP delta map" where the quantization parameters derived from the rate control algorithm can be tweaked using the QP delta map.

At a high level, capabilities of the NVENC hardware exposed through NVENC APIs are summarized in Table 1 and the additional features exposed in the Software stack in the current SDK are explained in Table 2.

Table 1. NVENC Hardware Capabilities

Feature	Description	Kepler GPUs	First generation Maxwell GPUs	Second generation Maxwell GPUs	Pascal GPUs
H.264 Base, Main, 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.	×	✓	✓	✓
Support for ARGB Input	Capability to encode RGB input.	✓	✓	✓	✓
HEVC Main Profile	Capability to encode YUV 4:2:0 sequence and generate a HEVC bit stream.	×	×	✓	✓

Feature	Description	Kepler GPUs	First generation Maxwell GPUs	Second generation Maxwell GPUs	Pascal GPUs
HEVC Main10 Profile	Support for Encoding 10-bit content generate a HEVC bit stream.	×	×	×	✓
HEVC Lossless Encoding	Lossless Encoding.	×	×	×	✓
HEVC Sample adaptive Offset(SAO)	Significantly improves encoded video quality for HEVC.	×	×	×	✓
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 8192x8192 Content	×	×	×	✓

\* : Present in Select Pascal Generation GPUs

Table 2. Feature additions in the current SDK

Additional Software features	Description
Pascal Support	Underlying software support for the Pascal generation GPUs.
HEVC Main10 support	Software support to input 10 bit YUV 4:2:0, YUV 4:4:4 and RGB input has been added. The feature can be controlled from NV Encode API.
HEVC Sample Adaptive Offset(SAO)	SAO is always enabled from the underlying Software Stack. This provides significant quality improvement.
HEVC 4:4:4 encoding support	Software support for encoding a YUV 4:4:4 Content.
HEVC ME only mode	Motion Estimation only mode for HEVC.
HEVC Lossless encoding support	Software support for encoding a content as Lossless. This is supported for all combinations of YUV4:2:0 and YUV4:4:4 for bit depths of 8 and 10.
HEVC Long Term Reference Frame support	Long term reference frame for HEVC helps in error resilience while streaming videos across noisy channels.

Additional Software features	Description
HEVC 8K Encoding support	Select Pascal generation GPUs can encode 8192x8192 resolution videos.
H.264 Temporal Adaptive Quantization (TAQ)	Improves perceptual Video quality when enabled.
Look ahead	Improves encoded picture quality at the cost of increased latency. It also adaptively inserts B frames (for H.264 Only) and Intra frames.
Asynchronous ME only Mode (H.264 & HEVC)	Support for invoking the Motion Estimation (ME) only Mode asynchronously which improves overall throughput.
Constant Quality (CQ)	Enables Constant Quality Mode where in video quality can be chosen by a quality factor.
Bug fixes	There have been several bug fixes since the last SDK release which have improved the overall stability and robustness of the software stack.
Sample application support	Support for some of the newly exposed features have been added to the sample applications.

### 3. NVENC LICENSING POLICY

There is no change in licensing policy in the current SDK in comparison to the earlier SDK (Video Codec SDK 6.0). The licensing policy is explained as follows:

The underlying software puts a limit of “two” concurrent encoding sessions on the combined number of encoding sessions executed on all non-qualified cards present on the system.

For example, on a system with one Quadro K4000 card and three GeForce cards, 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  $N + 2$ .

For the purposes of this discussion, non-qualified hardware is defined as any GeForce GPUs or low-end Quadro GPUs (for a complete list, refer to <https://developer.nvidia.com/nvidia-video-codec-sdk>).

## 4. NVENC PERFORMANCE

The Pascal NVENC hardware improves standalone encoding performance compared to earlier NVENC engines. The application can trade performance for encoded picture quality.

While Kepler and first generation Maxwell GPUs had one NVENC engine, certain variants of the second generation Maxwell GPUs and Pascal Generation GPUs have two/three NVENC engines physically present on the silicon. That enables the clients to support a greater number of concurrent encoding sessions. The underlying software implementation takes care of the load balancing between the two/three engines so that applications don't require changes in their own software stack to take advantage of multiple engines.

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 a large number of simultaneous encoding sessions to run in parallel.

NVENC API exposes several presets, rate control modes and flags for programming the hardware. A combination of these parameters enables video encoding at varying quality and performance.

Note that the encoder performance is a function of several parameters. Table 3 provide an indicative data of NVENC performance on Kepler, Maxwell and Pascal GPUs for different presets and rate control modes.

Table 3. NVENC Encoding Performance

Preset	RC Mode*	H.264 (FPS)				HEVC(FPS)	
		Kepler	First Gen. Maxwell	Second Gen. Maxwell	Pascal	Second Gen. Maxwell	Pascal
High Performance	Constant QP	227	329	430	648	199	391
	Single Pass	220	345	432	631	200	395
	Dual Pass	114	247	302	470	150	286
High Quality	Constant QP	78	213	261	384	137	249
	Single Pass	78	211	261	388	142	259
	Dual Pass	57	180	240	355	79	151
Low latency High Performance	Constant QP	146	235	361	535	199	392
	Single Pass	143	234	357	531	200	396
	Dual Pass	93	202	277	398	149	279
Low latency High Quality	Constant QP	77	226	260	381	199	392
	Single Pass	77	223	259	379	200	396
	Dual Pass	57	200	242	364	104	215

Resolution/Format: 1920x1080/YUV4:2:0, 8 bit

FPS: Encoding speed in "Frames per second".

\*: Rate Control Mode

## 5. PROGRAMMING NVENC

Video Codec SDK 7.0 is supported on R367 drivers and above. Please refer to the SDK Release notes for information regarding the minimum R367 driver version which adds the support for the current SDK.

Various capabilities of NVENC are exposed to the application software via the NVIDIA proprietary application programming interface (NVENC API). Please refer to the Video Encoder NVENC Programming guide for details on using the APIs to accelerate video encoding with NVENC hardware.

For a complete list of GPUs supporting hardware accelerated encoding please refer to <https://developer.nvidia.com/nvidia-video-codec-sdk>.

## **Notice**

ALL NVIDIA DESIGN SPECIFICATIONS, REFERENCE BOARDS, FILES, DRAWINGS, DIAGNOSTICS, LISTS, AND OTHER DOCUMENTS (TOGETHER AND SEPARATELY, "MATERIALS") ARE BEING PROVIDED "AS IS." NVIDIA MAKES NO WARRANTIES, EXPRESSED, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE MATERIALS, AND EXPRESSLY DISCLAIMS ALL IMPLIED WARRANTIES OF NONINFRINGEMENT, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE.

Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication of otherwise under any patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all other information previously supplied. NVIDIA Corporation products are not authorized as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

## **Trademarks**

NVIDIA, the NVIDIA logo, GeForce, Quadro, Tesla, and NVIDIA GRID are trademarks and/or registered trademarks of NVIDIA Corporation in the U.S. and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

## **Copyright**

© 2011-2016 NVIDIA Corporation. All rights reserved.