Whitepaper

NVIDIA® Tegra® X1

NVIDIA'S New Mobile Superchip

V1.0

Table of Contents

Introduction

As mobile devices, automobiles, and numerous embedded applications continue to demand more and more visual computing capabilities, highly capable mobile processors incorporating advanced GPUs have become essential to enable all these new features.

NVIDIA Tegra® X1 is NVIDIA's newest mobile processor, and includes NVIDIA's highest performing, and power efficient **Maxwell™** GPU architecture. Utilizing a **256 CUDA Core Maxwell GPU**, Tegra X1 delivers class-leading performance and incredible energy efficiency, while supporting all the modern graphics and compute APIs.

Tegra X1 will significantly improve mobile gaming realism, deliver the highest quality 4K mobile video experiences, and enable important new embedded, automotive, robotics, and computer vision applications. In particular, Tegra X1 will help deliver breakthroughs in automotive systems for advanced driver assistance (ADAS), computer vision, deep learning, instrument clusters, and infotainment.

This paper will explain the architectural features and capabilities of Tegra X1, and focus in detail on mobile graphics and automotive technologies made possible by Tegra X1 and NVIDIA's strengths in visual computing, cloud computing, and supercomputing applied to automotive applications.

The Maxwell GPU in Tegra X1 delivers desktop-class visual quality and graphics richness to mobile games by supporting many advanced graphics features. In addition to Tessellation, Compute Shaders, Dynamic Lighting, and Bindless Textures that were also supported in Tegra K1, Tegra X1 includes new graphics features such as Third Generation Delta Color Compression for lower power consumption, two Maxwell-class Polymorph Engines for better tessellation performance, and Programmable Sampling which enables new anti-aliasing techniques such as Multi-Frame Anti-Aliasing (MFAA) for better visual quality at a lower performance cost.

The Maxwell GPU in Tegra X1 provides exceptional compute performance for deep learning, computer vision, and other compute-based applications. Tegra X1-based automotive systems working in concert with NVIDIA's cloud-based supercomputer technologies will enable many new advanced and intelligent driver assistance technologies, accelerating the path to the ultimate goal — self-driving vehicles.

Applications such as computer vision-based robots and multi-camera-based driverless cars need tremendous compute power to analyze multiple live video streams in real-time to deliver immediate, accurate, and meaningful results. The Maxwell GPU core in Tegra X1 includes native hardware support for 16-bit Floating Point calculations enabling higher compute performance that is particularly important for computer vision-based automotive and embedded applications. Tegra X1 also includes two high performance Image Signal Processors (ISP) each capable of processing 650 Mpixels/s for a total of 1.3 Gpixels/s of image data from multi-camera-based advanced driver assistance systems that can be processed in real-time

With 4K televisions and 4K media content becoming more widely available, Tegra X1 is designed to deliver a premium 4K experience. Tegra X1 supports hardware decode of both H.265 (HEVC) and VP9 4K

video streams at 60fps, delivering a silky smooth viewing experience for many types of high quality 4K video streams, including movies and fast action sports. Tegra X1 devices connected to cloud gaming services or local game streaming servers that employ H.265 video encoding will also benefit from H.265 hardware decode. Tegra X1 supports decode of 10-bit H.265 video streams enabling Tegra X1 powered devices to playback 4K content from services such as Netflix

Automotive Visual Computing

The number of digital display panels in automobiles has steadily increased over the years, and more car models are starting to use high-resolution display panels for navigation information, driver cockpit controls, and passenger infotainment content. Car manufacturers are also increasing the number of cameras in automobiles for improved driver assistance, and are already designing next generation cars that use up to twelve cameras. To deploy these advanced features, carmakers are investing heavily in new visual computing system hardware and software development. To that end, NVIDIA introduces the NVIDIA DRIVE™ line of car computers.

The **NVIDIA DRIVE™ CX** is a complete cockpit visualization platform powered by Tegra X1 that delivers advanced graphics and computer vision capabilities, along with a fully integrated and road-tested software stack. The NVIDIA DRIVE CX system includes standard input/output interfaces for cameras, modems, Bluetooth, and other ports to interface with the rest of the car. Car manufacturers can take the NVIDIA DRIVE CX platform as-is, easily integrate it into their car designs and quickly bring to market advanced visualization and driver assistance features at a fraction of their current costs.

The use of driver assistance systems in automobiles is increasing rapidly, and many cars include features such as top view, collision detection, and collision avoidance features. The performance, power efficiency, and programmability of Tegra X1 enables it to be used in advanced driver assistance systems that can deliver contextual driver assistance, deep learning-based, continually evolving collision avoidance capability, driverless Auto-Valet parking, and many other features.

The **NVIDIA DRIVE™ PX** Auto-Pilot computer, powered by dual Tegra X1 processors, is a complete platform that supports up to 12 camera inputs and is capable of running multi-layer neural network-based algorithms to deliver advanced real-time contextual driver assistance features. The NVIDIA DRIVE PX system is also designed to communicate with NVIDIA Tesla GPU-based supercomputers in the cloud, uploading real-time data for analysis, and periodically downloading newer, refined neural network models that help deliver continually improving driver assistance performance.

NVIDIA Tegra® X1

NVIDIA's **Tegra® K1** created a discontinuity in the state of mobile graphics by bringing the 192 core **NVIDIA Kepler™ GPU** architecture to mobile, and delivering tremendous visual computing capabilities, breakthrough power efficiency, and advanced desktop-class graphics features to mobile.

NVIDIA raises the bar again for mobile visual computing with its **Tegra® X1** mobile processor based on the **NVIDIA Maxwell™** GPU architecture. In addition to supporting advanced graphics and compute

features such **as OpenGL 4.5**, **AEP**, the **DirectX 12 API**, and **CUDA 6.0**, with over 1000 **GFLOPS** of GPU processing power for 16-bit workloads(fp16 operations), and over 500 GFLOPS for 32-bit workloads(fp32 operations), **Tegra X1** delivers **2x the raw performance** and **power efficiency** of Tegra K1. Tegra X1 is the world's first **TeraFLOPS**[1] mobile processor delivering both the performance and power efficiency needed by the next generation of visual computing applications in automotive, machine learning, embedded computing, and mobile devices.

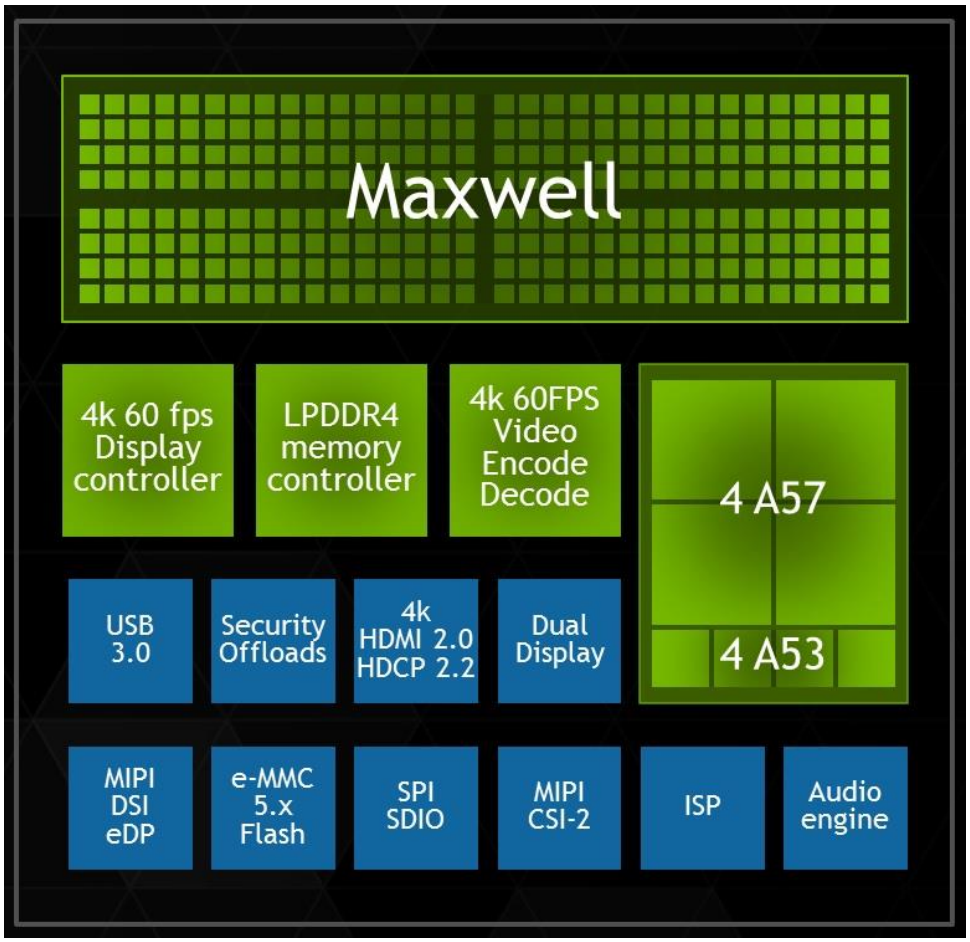

Figure 1 NVIDIA Tegra X1 Mobile Processor

Some of the key features of the Tegra X1 SoC (System-on-a-Chip) architecture are:

- **ARM® Cortex® A57/ A53 64/32-bit** CPU architecture that delivers high performance and power efficiency.
- **Maxwell GPU architecture** that utilizes 256 cores to deliver class-leading performance and power efficiency for the next generation of visual computing applications

[1] References the native FP16 (16-bit floating-point) processing capability of NVIDIA Tegra X1. FP16 precision is well suited for image processing applications, which are key application areas for automotive and embedded.

- **End-to-end 4K 60 fps pipeline** that delivers a premium 4K experience with support for 4K, 60 fps decode of H.265 (HEVC) and VP9 streams.
- Built on the TSMC **20nm** (20SoC) process to deliver excellent performance and power efficiency.

NVIDIA Tegra X1 CPU Architecture

The NVIDIA Tegra X1 CPU architecture uses four high performance ARM Cortex A57 cores in conjunction with four power-efficient ARM Cortex A53 cores. The Cortex A57 CPU complex on Tegra X1 shares a common **2MB L2 cache,** and each of the four CPU cores has a 48KB L1 instruction cache and a 32KB L1 data cache. The lower performance, more power-efficient Cortex A53 CPU complex share a common 512KB L2 cache, and each of its four CPU cores has its own 32KB L1 instruction cache and 32KB L1 data cache. Workloads that require high performance are processed by the A57 CPU cores, and lower performance workloads are processed by the energy-efficient A53 CPU cores. Intelligent algorithms analyze workloads presented by the operating system to dynamically switch between the high performance and low performance cores to deliver optimal performance and power efficiency.

Because of NVIDIA's learning and experience with its **4-PLUS-1** CPU architecture first introduced on NVIDIA Tegra 3, and expertise in creating high power, efficient silicon layout designs, Tegra X1 delivers higher performance and power efficiency than other **SoCs** (System-on-a-Chip) that are based on the A57/A53 CPU implementation. Tegra X1 provides almost 2x the power efficiency for the same CPU performance. And for the same power consumed, Tegra X1 delivers almost 1.4x higher CPU performance.

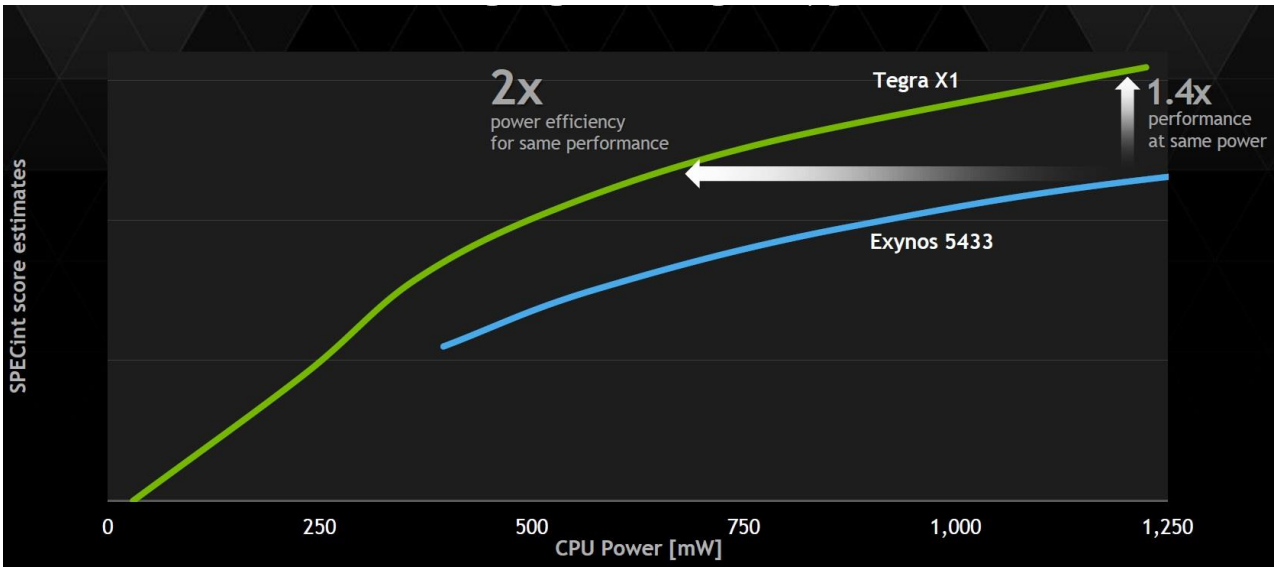

Figure 2 Tegra K1 Delivers higher CPU Performance and Power Efficiency[2]

[2] CPU power and performance measured on Tegra X1 development platform and the Exynos 5433 based Galaxy Note 4

High Performance End-to-End 4K 60 fps Pipeline

4K displays and televisions are becoming mainstream thanks to the rapidly dropping prices of 4K panels and the increasing availability of 4K content. To meet this rising use of 4K panels and content, NVIDIA Tegra X1 is architected with a high performance, end-to-end 4K 60 fps pipeline that delivers a premium 4K experience for use cases such as YouTube® videos, Netflix® streaming, Google Hangouts, 4K Gamestreaming, and 4K Chromecast. The I/O interfaces and processing cores of Tegra X1 including its high speed storage controller, memory controllers, image signal processor, video decoder, 4K compositor, graphics processor, and display controllers are all optimized to deliver 4K at sixty frames per second (60fps).

Tegra X1 supports 4K H.265 (HEVC) and VP9 video streams at 60 fps. Other processors support 4K at 30 fps, and deliver sub optimal experiences while viewing fast action sports, movies, and video games. Tegra X1 also supports decode of 10-bit color-depth 4K H.265 60 fps video streams. This enables Tegra X1 products to stream a wide selection 4K content from services such as Netflix. Tegra X1 supports 4K 60 fps local and external displays with support for HDMI 2.0 interfaces and HDCP 2.2 copy protection. On the encode side, Tegra X1 supports encode of 4K video at 30 fps in H.264, H.265 and VP8 formats.

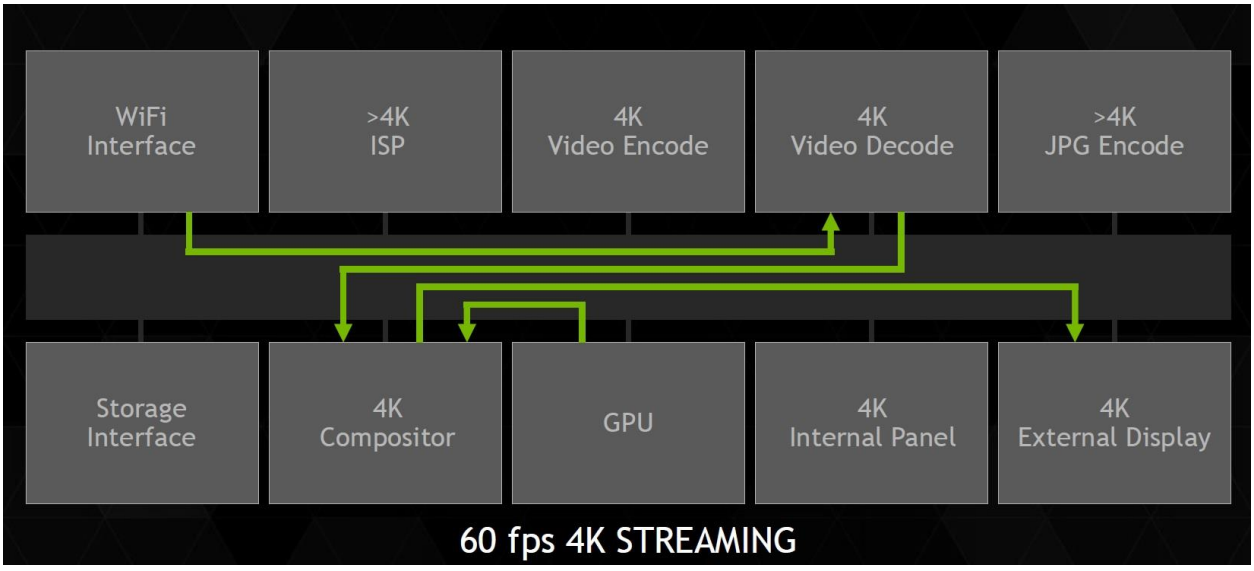

Figure 3 Tegra X1 is optimized end-to-end to support 4K, 60 fps decode of H.265 and VP9 streams

NVIDIA Maxwell

One of the most complex processors ever created, the GPU is the engine behind state-of-the-art computer graphics and energy efficient computing. NVIDIA's latest GPU architecture, codenamed Maxwell, delivers unprecedented performance and power efficiency. Maxwell-based GPUs such as the GeForce® GTX™ 980 and GTX 980M are the engines behind some of the world's highest performing gaming desktop and laptop PCs, respectively.

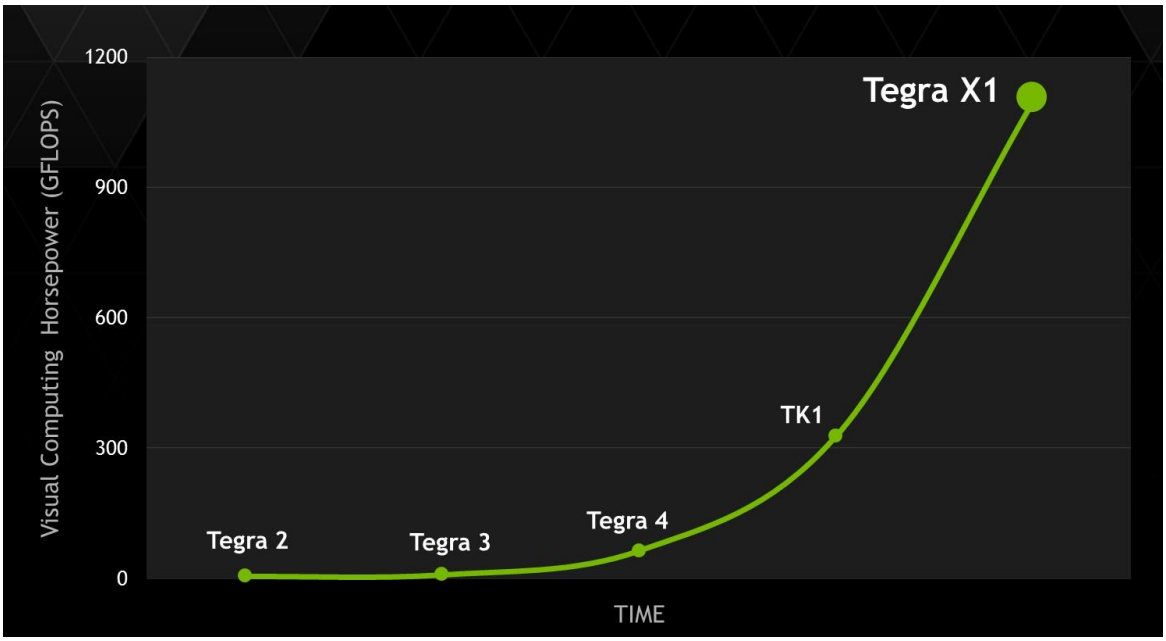

Figure 4 Maxwell in Tegra X1 delivers over one TeraFLOPS of FP16 performance

The Maxwell GPU architecture was designed to deliver an extraordinary leap in power efficiency and unrivaled performance. The Maxwell architecture at a high level is similar to its predecessor, the Kepler GPU architecture in the sense that it is based on fundamental compute cores called CUDA cores, Streaming Multiprocessors (SMs), Polymorph Engines, Warp Schedulers, Texture Caches, and other hardware elements. But each hardware block on Maxwell has been optimized and upgraded with an intensive focus on power efficiency.

By making the fundamental building blocks of the GPU more power efficient, the performance of the GPU as a whole can then be scaled for various use cases without compromising power consumption. Thanks to Maxwell's significant improvements in energy efficiency, the Maxwell-based Tegra X1 processor delivers 2x the power efficiency of Tegra K1 and over one TeraFLOPS of peak FP16 performance and over 500 GFLOPS of peak FP32 performance. As noted in our detailed architecture section below, a single Maxwell SM with 128 cores is roughly the same performance as a 192 core Kepler SM when operating at same clock rates due to the Maxwell SM's improvements in execution efficiency, so the two Maxwell SMs in Tegra X1 have roughly 2x the overall GPU horsepower of the single SM in Tegra K1, while Tegra X1 also includes native support for FP16 Fused Multiple-Add (FMA) operations in addition to FP32 and FP64. To provide double rate FP16 throughput, Tegra X1 supports 2-wide vector FP16 operations, for example a 2-wide vector FMA instruction would read three 32-bit source registers A, B and C, each containing one 16b element in the upper half of the register and a second in the lower half, and then compute two 16b results (A*B+C), pack the results into the high and low halves of a 32 bit output which is then written into a 32-bit output register. In addition to vector FMA, vector ADD and MUL are also supported.

Similar to Tegra K1, Tegra X1 with its Maxwell GPU core continues to stand apart from the competition by supporting the latest features of all the key graphics and compute APIs such as **OpenGL ES 3.1**,

OpenGL 4.5, **Android Extension Pack (AEP)**, **DirectX 12.0**, and **CUDA 6**. In addition to advanced graphics features such as tessellation, Bindless textures, and PhysX, the Maxwell GPU core in Tegra X1 brings next generation graphics features such as Voxel based Global Illumination (VXGI), MFAA™ (Multi-Frame Anti-aliasing) , improved memory compression, and faster path rendering.

Extraordinary Graphics Performance

While Tegra K1 with its 192-core Kepler GPU is one of the highest performing mobile processors, Tegra X1 with its 256-core Maxwell GPU raises performance of mobile processors to a new level. On graphics performance, based on the popular GFXBench 3.0 graphics benchmark, Tegra X1 delivers 2x the performance of Tegra K1.

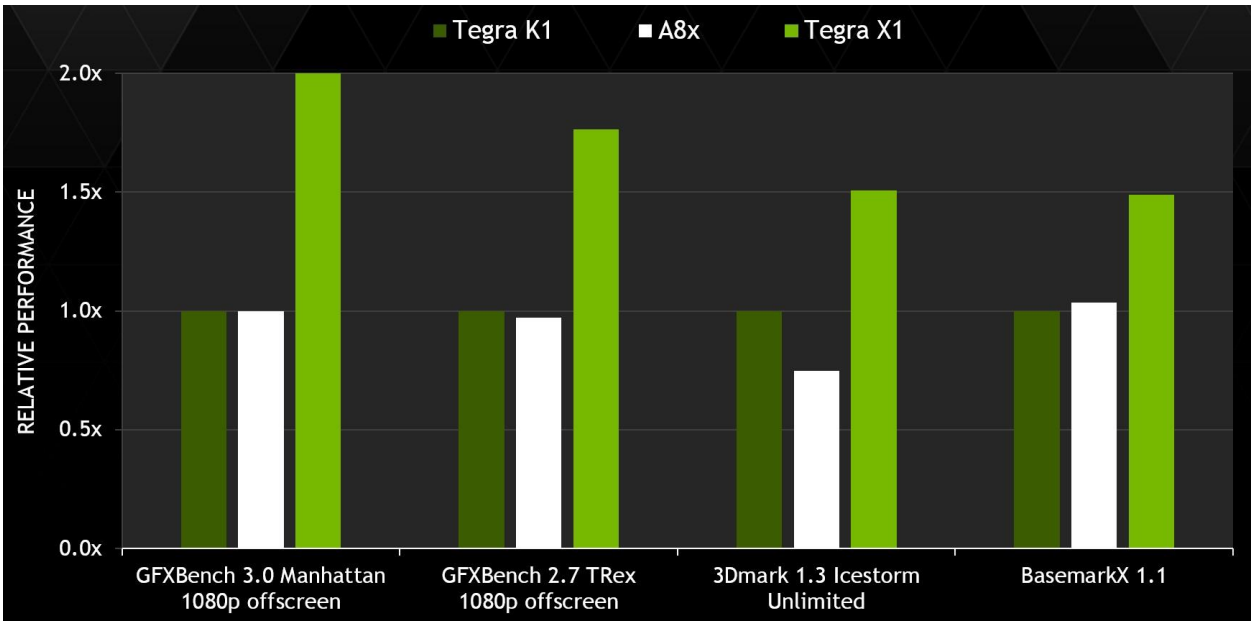

Figure 5 Tegra X1 graphics performance is up to 2x higher than other mobile processors

Looking beyond raw graphics performance, just like Tegra K1, Tegra X1 supports modern graphics APIs. While other mobile processors have claimed support for these features, Tegra X1 and Tegra K1 have successfully demonstrated applications such as the Epic Rivalry demo that use Android Extension Pack, OpenGL ES3.1, and the Unreal Engine 4 game engine. While Tegra K1 delivers an amazing visual experience on this technology demo, Tegra X1 takes it even further by delivering more than 2x the frames per second generated by Tegra K1.

Along with the extraordinary lead in graphics performance, due to its support for the full desktop graphics APIs, Tegra X1 will deliver a great experience on advanced PC games that are ported to mobile.

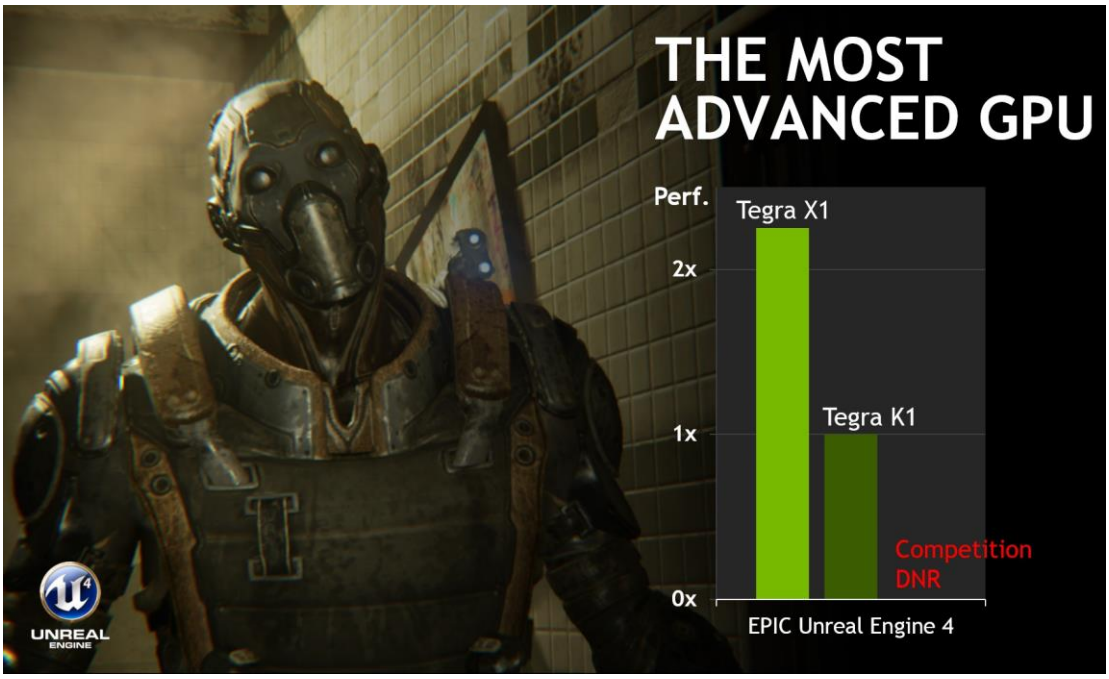

Figure 6 Tegra X1 delivers more than 2x the performance of Tegra K1 on AEP and OpenGL ES3.1 based EPIC Rivalry demo

Incredible Energy Efficiency

The Maxwell GPU architecture was designed to provide an extraordinary leap in power efficiency and deliver unrivaled performance, while simultaneously reducing power consumption from the previous generation. With a combination of advances originally developed for Tegra K1, and other architectural innovations, Tegra X1 with its Maxwell GPU core delivers up to 2x the performance per watt of Tegra K1.

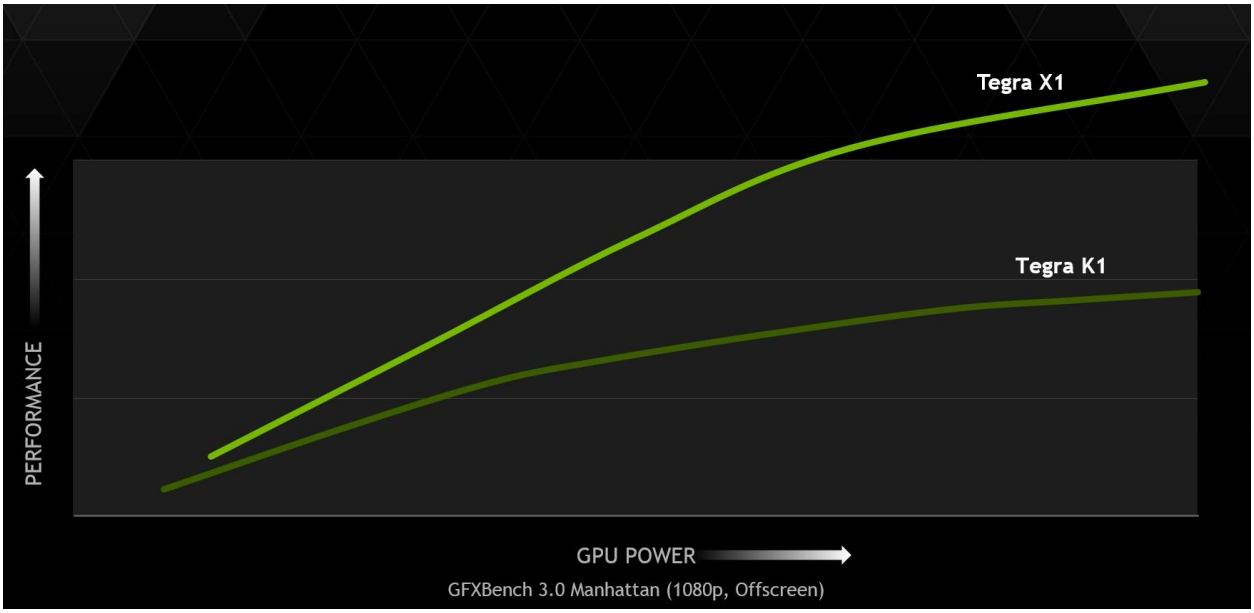

Figure 7 Tegra X1 delivers twice the power efficiency of Tegra K1

This leap in power efficiency is due to the optimizations implemented in the Maxwell compute core, architectural reorganizations in the Maxwell SMM, and improved memory compression (described in a later section). Several key features first implemented in the Kepler GPU core are also found on the Maxwell core and help enable higher power efficiency. Features such as hierarchical on-chip Z cull, primitive culling, Early Z culling, Texture, Z, and color compression, and a large unified L2 cache significantly decrease accesses to power hungry off-chip memory.

Maxwell Graphics Architecture in Tegra X1

Similar to the Kepler GPU, the Maxwell GPU architecture is organized in Graphics Processing Clusters (GPC), Streaming Multiprocessors (SM), and memory controllers (if you are not well versed in these structures, we suggest you first read the Kepler and Fermi whitepapers). The Maxwell GPU in Tegra X1 contains two SMs; each SM consists of fundamental compute cores called CUDA Cores, texture units, and a Polymorph engine. Each SM in the Kepler GPU (called SMX) architecture consists of 192 CUDA cores, while each Maxwell SM (called SMM) includes 128 CUDA cores. However, a Maxwell CUDA core is a significant upgrade over a Kepler CUDA core, and each Maxwell core delivers almost forty percent higher performance than a Kepler core.

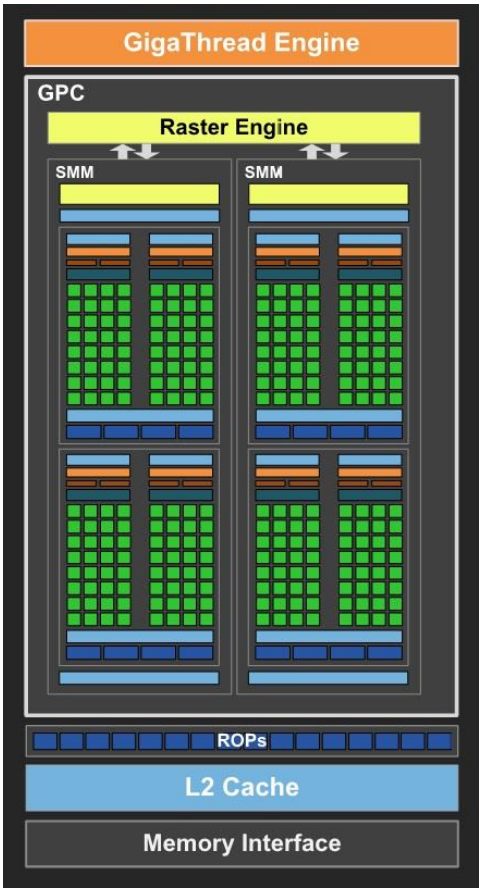

Figure 8 Maxwell GPU in Tegra X1

The fundamental architecture of the Maxwell GPU used in Tegra X1 is virtually identical to that found in the high end Maxwell based GPU (GM204) used in GTX 980 desktop graphics cards, but differs primarily in scale and memory architecture. The GM204 consists of four GPCs with each GPC having 4 SMM blocks, while the Tegra X1 configuration includes one GPC that has two SMM blocks. Thus while a high end Maxwell based GTX980 graphics card with a GM204 GPU includes a total of 2048 CUDA cores and 4GB of frame buffer memory, consuming approximately 165 Watts of power, the Maxwell GPU in Tegra X1 consists of 256 CUDA cores, shares DRAM with the Cortex A57/A53 CPU complexes, and consumes only a few watts.

The Maxwell GPU in Tegra X1 also includes 16 ROPs, 2 Geometry units, 16 Texture units, and has a 256KB L2 cache between the ROPs and the 64-bit LPDDR4 memory interface. The following table provides a high-level comparison of the Maxwell GPU core in Tegra X1 and the Kepler GPU core in Tegra K1

GPU	Tegra K1 (Kepler GPU)	Tegra X1 (Maxwell GPU)
SMs	1	2
CUDA Cores	192	256
GFLOPs (FP32) Peak	365	512
GFLOPs (FP16) Peak	365	1024
Texture Units	8	16
Texel fill-rate	7.6 Gigatexels/sec	16 Gigatexels/sec
Memory Clock	930 MHz	1.6GHz MHz
Memory Bandwidth	14.9 GB/s	25.6 GB/s
ROPs	4	16
L2 Cache Size	128KB	256KB
Manufacturing Process	28-nm	20-nm
Z-cull	256 pixels/clock	256 pixels/clock
Raster	4 pixels/clock	16 pixels/clock
Texture	8 bilinear filters/clock	16 bilinear filters/clock
ZROP	64 samples/clock	128 samples/clock

Table 1 Comparing Kepler GPU in Tegra K1 and Maxwell GPU in Tegra X1

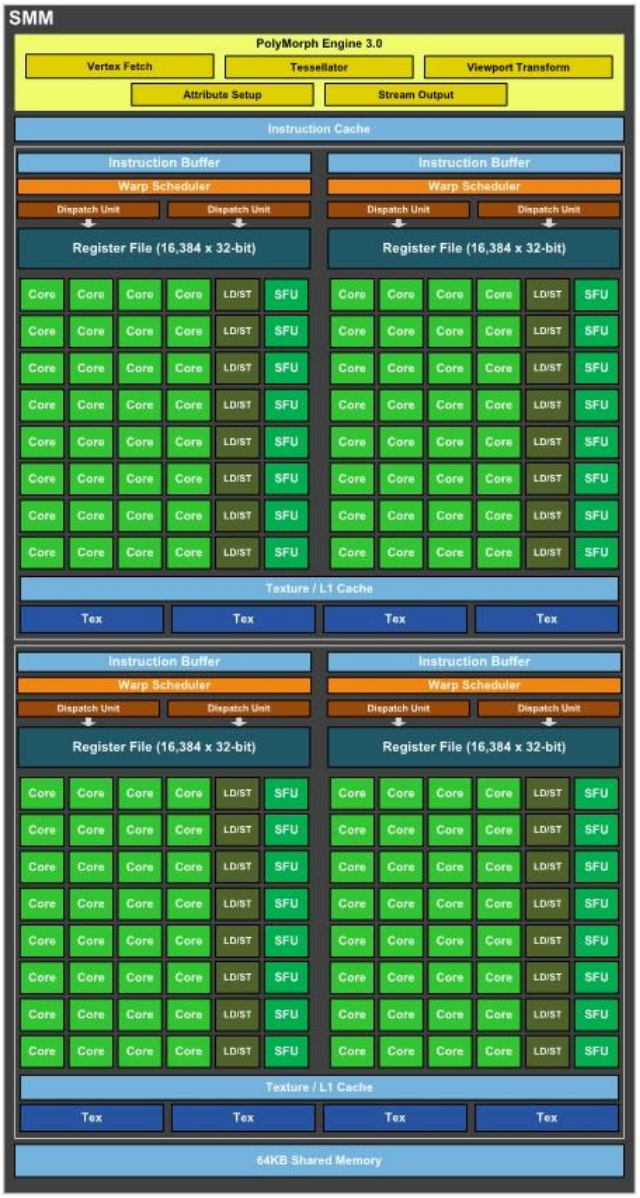

Figure 9: Maxwell SMM Diagram

Maxwell Streaming Multiprocessor

The SM is the heart of our GPUs. Almost every operation flows through the SM at some point in the rendering pipeline. Maxwell GPUs feature a new SM that's been designed to provide dramatically improved performance per watt than prior GeForce GPUs.

Compared to GPUs based on our Kepler architecture, Maxwell's new SMM design has been reconfigured to improve efficiency. Each SMM contains four warp schedulers, and each warp scheduler is capable of dispatching two instructions per warp every clock. Compared to Kepler's scheduling logic, we've integrated a number of improvements in the scheduler to further reduce redundant re-computation of scheduling decisions, improving energy efficiency. We've also integrated a completely new datapath organization. Whereas Kepler's SM shipped with 192 CUDA Cores—a non-power-of-two organization—the Maxwell SMM is partitioned into four distinct 32-CUDA core processing blocks (128 CUDA cores total per SM), each with its own dedicated resources for scheduling and instruction buffering. A pair of processing blocks shares a texture and L1 cache, while the PolyMorph Engine and shared memory are a common resource for all the cores in the SM. This new configuration in Maxwell aligns with warp size, making it easier to utilize efficiently and saving area and power that had to be spent to manage data transfer in the more complex datapath organization used by Kepler.

Compared to Kepler, the SMM's memory hierarchy has also changed. Rather than implementing a combined shared memory/L1 cache block as in Kepler SMX, Maxwell SMM units in Tegra X1 feature a 64KB dedicated shared memory, while the L1 caching function has been moved to be shared with the texture caching function.

As a result of these changes, each Maxwell CUDA core is able to deliver roughly 1.4x more performance per core compared to a Kepler CUDA core, and 2x the performance per watt. At the SM level, with 33% fewer total cores per SM, but 1.4x performance per core, each Maxwell SMM can deliver total per-SM

performance similar to Kepler's SMX at the same clock rates, and the area savings from this more efficient architecture enabled us to then double up the total SM count, compared to Tegra K1. Tegra X1 also has area benefits from using a 20nm manufacturing process.

Polymorph Engine 3.0

Tessellation is one of OpenGL 4.x's and AEP's key features and will play a bigger role in the future as the next generation of games are designed to use more tessellation. Due to the doubling of the number of SMMs on Tegra X1 allows it to benefit from 2x the Polymorph Engines (PE) compared to Tegra K1. As a result, performance on geometry heavy workloads is roughly doubled, and due to architectural improvements within the PE, even higher performance improvement can be achieved with high tessellation expansion factors.

Improved Memory Compression

Maxwell GPU architecture has significantly enhanced memory compression to reduce memory bandwidth and thus power consumption.

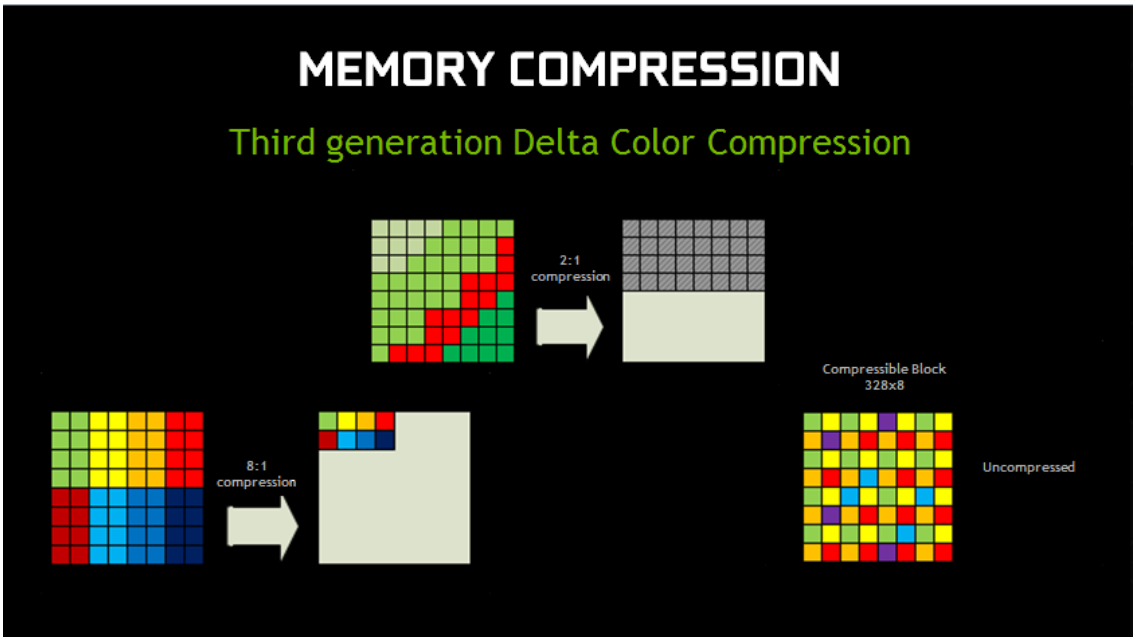

Figure 10 Third generation Delta Color compression in Maxwell GPU

To reduce DRAM bandwidth demands, NVIDIA GPUs make use of lossless compression techniques as data is written out to memory. The bandwidth savings from this compression is realized a second time when clients such as the Texture Unit later read the data. As illustrated in the preceding figure, our compression engine has multiple layers of compression algorithms. Any block going out to memory will first be examined to see if 4x2 pixel regions within the block are constant, in which case the data will be compressed 8:1 (i.e., from 256B to 32B of data, for 32b color). If that fails, but 2x2 pixel regions are constant, we will compress the data 4:1.

These modes are very effective for AA surfaces, but less so for 1xAA rendering. Therefore, starting in Fermi we also implemented support for a “delta color compression” mode. In this mode, we calculate the difference between each pixel in the block and its neighbor, and then try to pack these different values together using the minimum number of bits. For example if pixel A’s red value is 253 (8 bits) and pixel B’s red value is 250 (also 8 bits), the difference is 3, which can be represented in only 2 bits.

Finally, if the block cannot be compressed in any of these modes, then the GPU will write out data uncompressed, preserving the lossless rendering requirement.

However, the effectiveness of delta color compression depends on the specifics of which pixel ordering is chosen for the delta color calculation. Maxwell contains our third generation of delta color compression, which improves effectiveness by offering more choices of delta calculation to the compressor.

End-to-End Memory Compression

The Maxwell GPU core also supports end-to-end memory compression that helps reduce traffic to external system memory chip and thus reducing power compression. Display buffer data is compressed by the Maxwell GPU and sent to system memory. The Maxwell display controller and compositor are designed to read compressed data from system memory and un-compress the data on the fly before pushing it out to the local and/or external displays.

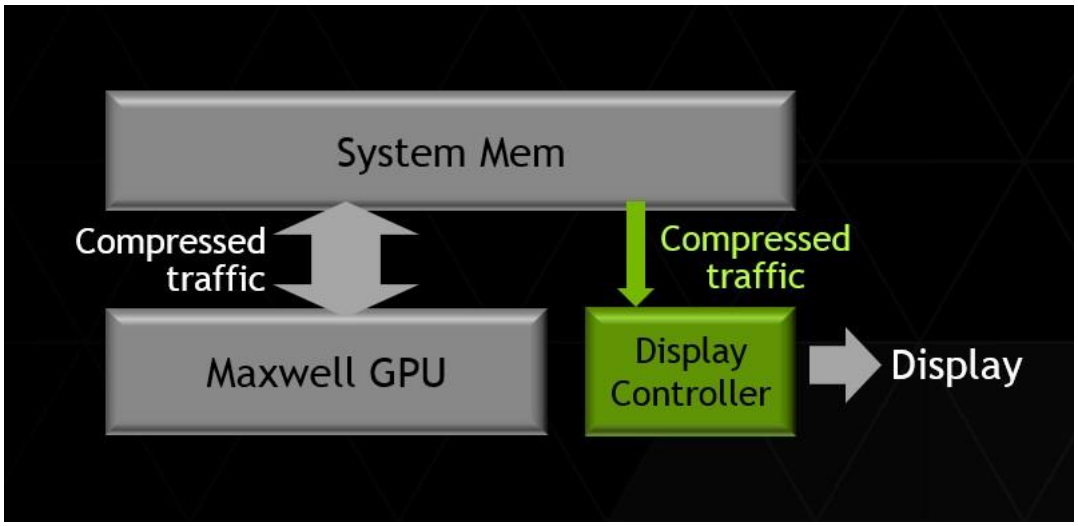

Figure 11 End-to-End compression reduces memory traffic and power consumption.

Thanks to the improvements in caching and compression in Maxwell, the GPU is able to significantly reduce the number of bytes that have to be fetched from memory per frame. In tests with a variety of games, Maxwell uses roughly 30% to 45% lower memory bandwidth compared to Tegra K1.

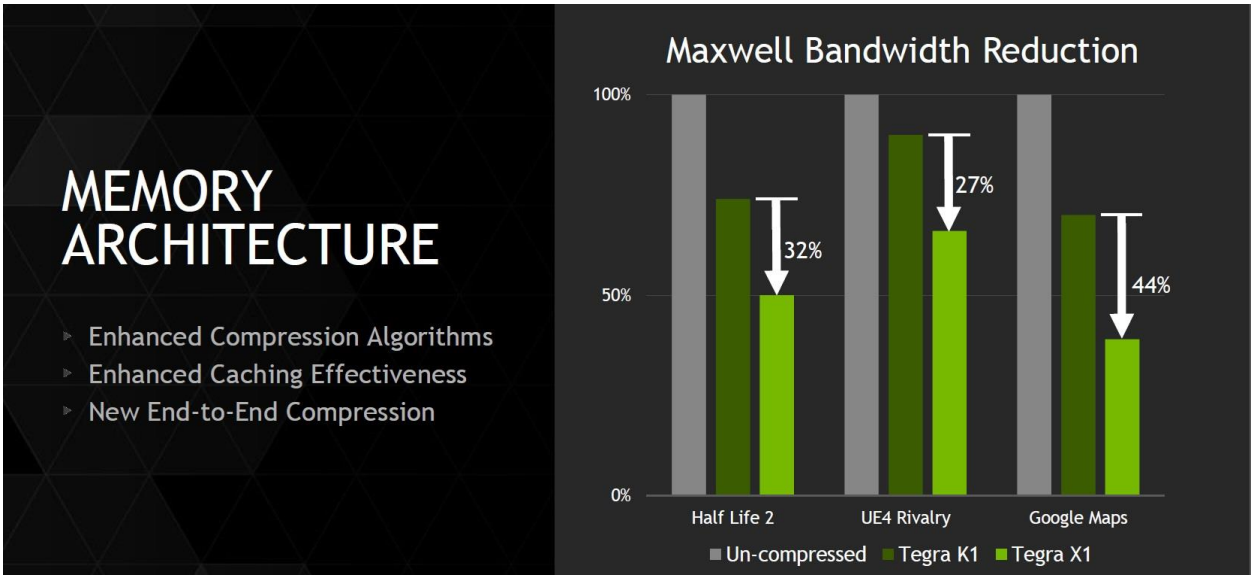

Figure 12 Memory bandwidth savings on Tegra X1

Raising the Bar on Mobile Graphics Quality

The Maxwell GPU core in Tegra X1 supports a host of features that enable a whole new level of mobile graphics quality. In addition to supporting all the advanced features first introduced in the Kepler GPU core, Maxwell brings a new set of features and capabilities that deliver impressive visual realism to mobile games. Some of these key features are briefly described below. For a deeper understanding of these features, please refer to the desktop Kepler and Maxwell whitepapers.

Tessellation

Tessellation is one of the key features of OpenGL 4.x and DirectX 11.x that has profoundly influenced 3D graphics for PC gaming, and has increased the level of visual realism in PC games to being almost film-like. The Kepler GPU core in Tegra K1 is the first to offer support for this feature in mobile. The Maxwell GPU in Tegra X1 delivers even higher tessellation performance for more detailed geometries at much higher frame rates and lower power consumption. More details on how tessellation works can be found here. Tessellation delivers more detailed terrains, character models, and environments.

Bindless Textures

In traditional GPU architectures, for the GPU to reference a texture, the texture had to be assigned a "slot" in a fixed-size binding table. The number of slots in that table ultimately limits how many unique textures a shader can read from at run time.

With bindless textures in Maxwell, the shader can reference textures directly in memory, making binding tables obsolete. This effectively eliminates any limits on the number of unique textures that can be used to render a scene. As a result, many more different texture materials can be used to increase

the texture detail in a game. Another benefit of bindless textures is the reduced driver and application overhead and lower CPU utilization.

Voxel Global Illumination

Voxel based Global Illumination (VXGI) is a technology that simulates light inside of a game, delivering incredibly realistic lighting, shading and reflections to next-generation games and game engines. This means that shadows look better, colors diffuse and mix based on light and the scene is much more realistic.

VXGI employs a combination of advanced software algorithm and specialized hardware in the Maxwell GPU and employs innovative new approach to computing a fast, approximate form of global illumination dynamically in real-time on the GPU. This new GI technology uses a voxel grid to store scene and lighting information, and a novel voxel cone tracing process to gather indirect lighting from the voxel grid. More details on how VXGI works can be found in here.

Multi-Projection Acceleration and Conservative Raster

To understand how voxel global illumination works, it is helpful to first understand voxels. The term "voxel" is related to "pixel." Whereas a pixel represents a 2D point in space, a voxel represents a small cube (a volume) of 3D space. To perform global illumination, we need to understand the light emitting from all of the objects in the scene, not just the direct lights. To accomplish this, we dice the entire 3D space of the scene in all three dimensions, into small cubes called voxels. "Voxelization" is the process of determining the content of the scene at every voxel, analogous to "rasterization" which is the process of determining the value of a scene at a given 2D coordinate.

To enable VXGI as a real-time dynamic lighting technique, the voxelization process needs to be extremely fast. Multi-Projection Acceleration and Conservative Raster are two new hardware features of Maxwell that were specifically designed for this purpose. During the voxelization process, the same scene geometry needs to be analyzed from many views –from every face of the voxel cube-to determine coverage and lighting. We call this property of rendering the same scene from multiple views "**multi-projection**." The specific capability that we added to speed up multi-projection is called "**Viewport Multicast**." With this feature, Maxwell can use dedicated hardware to automatically broadcast input geometry to any number of desired render targets, avoiding geometry shader overhead. In addition, we added some hardware support for certain kinds of per viewport processing that are important to this application.

Conservative Raster" is the second hardware feature in Maxwell that accelerates the voxelization process. Hardware support for conservative raster is very helpful for the coverage phase of voxelization. In this phase, fractional coverage of each voxel needs to be determined with high accuracy to ensure the voxelized 3D grid represents the original 3D triangle data properly. Conservative raster helps the hardware to perform this calculation efficiently; without conservative raster there are workarounds that can be used to achieve the same result, but they are much more expensive.

For more details on these features please refer to the desktop Maxwell Whitepaper.

Tiled Resources

DirectX 11.2 introduced a feature called Tiled Resources that could be accelerated with an NVIDIA Kepler and Maxwell hardware feature called **Sparse Texture**. With Tiled Resources, only the portions of the textures required for rendering are stored in the GPU's memory. Tiled Resources works by breaking textures down into tiles (pages), and the application determines which tiles might be needed and loads them into video memory. It is also possible to use the same texture tile in multiple textures without any additional texture memory cost; this is referred to as aliasing. In the implementation of voxel grids, aliasing can be used to avoid redundant storage of voxel data, saving significant amounts of memory. You can read more about Tiled Resources at this link.

Raster Ordered View

The next generation DX API introduces the concept of a "**Raster Ordered View**," which supports the same guaranteed processing order that has traditionally been supported by Z and Color ROP units. Specifically, given two shaders A and B, each associated with the same raster X and Y, hardware must guarantee that shader A completes all of its accesses to the ROV before shader B makes an access.

To support Raster Ordered View, Maxwell adds a new interlock unit in the shader with similar functionality to the unit in ROP. When shaders run with access to a ROV enabled, the interlock unit is responsible for tracking the XY of all active pixel shaders and blocking conflicting shaders from running simultaneously.

One potential application for Raster Ordered View is order independent transparency rendering algorithms, which handle the case of an application that is unable to pre-sort its transparent geometry by instead having the pixel shader maintain a sorted list of transparent fragments per pixel.

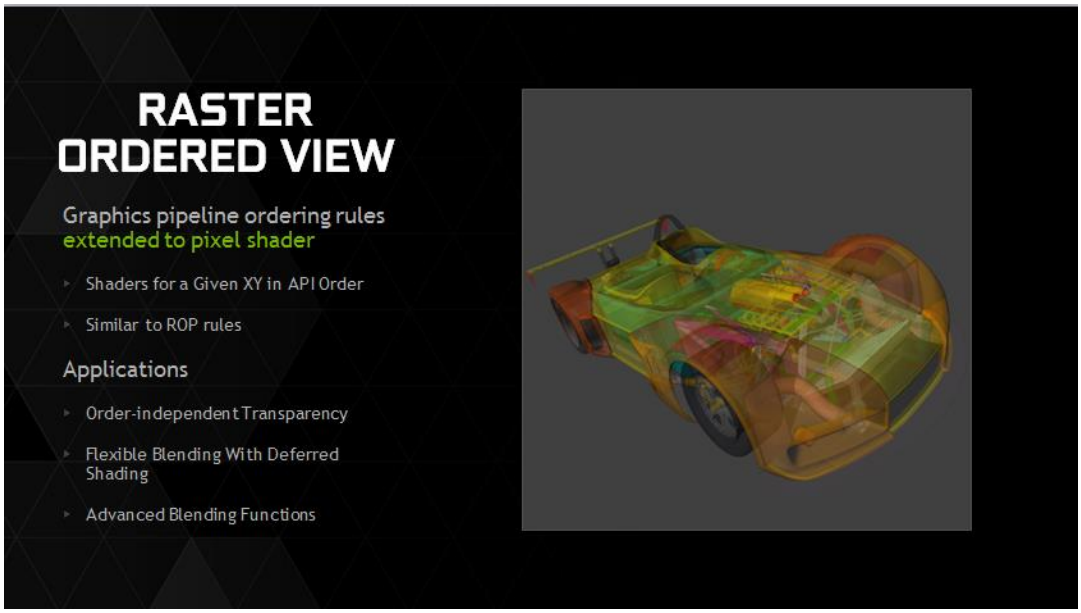

Figure 13: Raster Ordered View

Tegra X1 in Automotive

GPU-accelerated computing is rapidly increasing the velocity of innovation in the fields of science, medicine, finance, and engineering. CUDA has become the world's leading GPU computing platform used by millions of users for high-performance computing across a range of industries and sciences, including usage in many of the top supercomputers in the world. GPU computing delivers unprecedented levels of performance speedups by parallelizing application workloads and running them on the GPU.

The immense compute performance, easy general purpose GPU programmability and outstanding energy efficiency of Tegra X1 makes it very suitable for GPU compute-intensive automotive applications such as advanced surround-view based safety systems, computer vision based Auto-Valet parking systems, deep machine learning based contextual object recognizing driver assistance systems, and visually rich digital instrument clusters and infotainment systems.

NVIDIA DRIVE™ CX Cockpit Computer

The number of display panels used in automobiles is increasing rapidly. Just a few years ago, low-resolution panels displaying navigation information were offered in a handful of premium car models. But today most mainstream midrange car models come with built in display panels for navigation. Some high-end automobiles such as Tesla and Audi are even delivering navigation information and digital instrument clusters through high definition display panels in addition to delivering passenger infotainment through high definition rear-seat display panels.

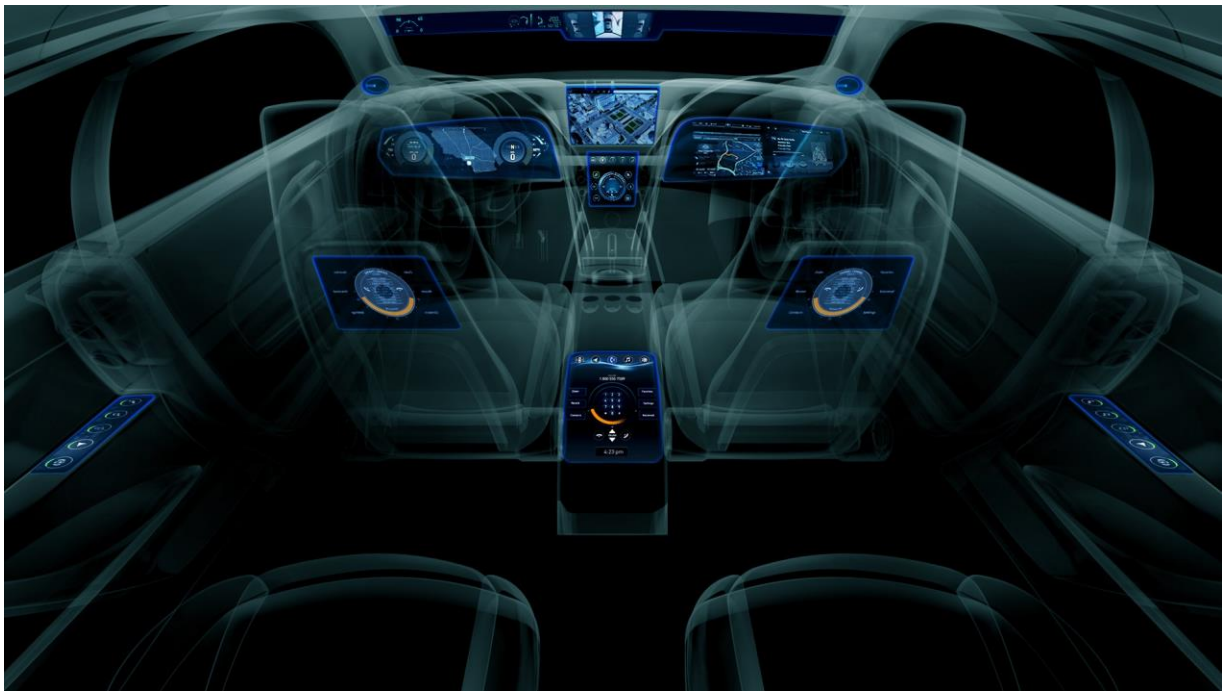

Figure 14 Cars in the future will use multiple display panels for various functions

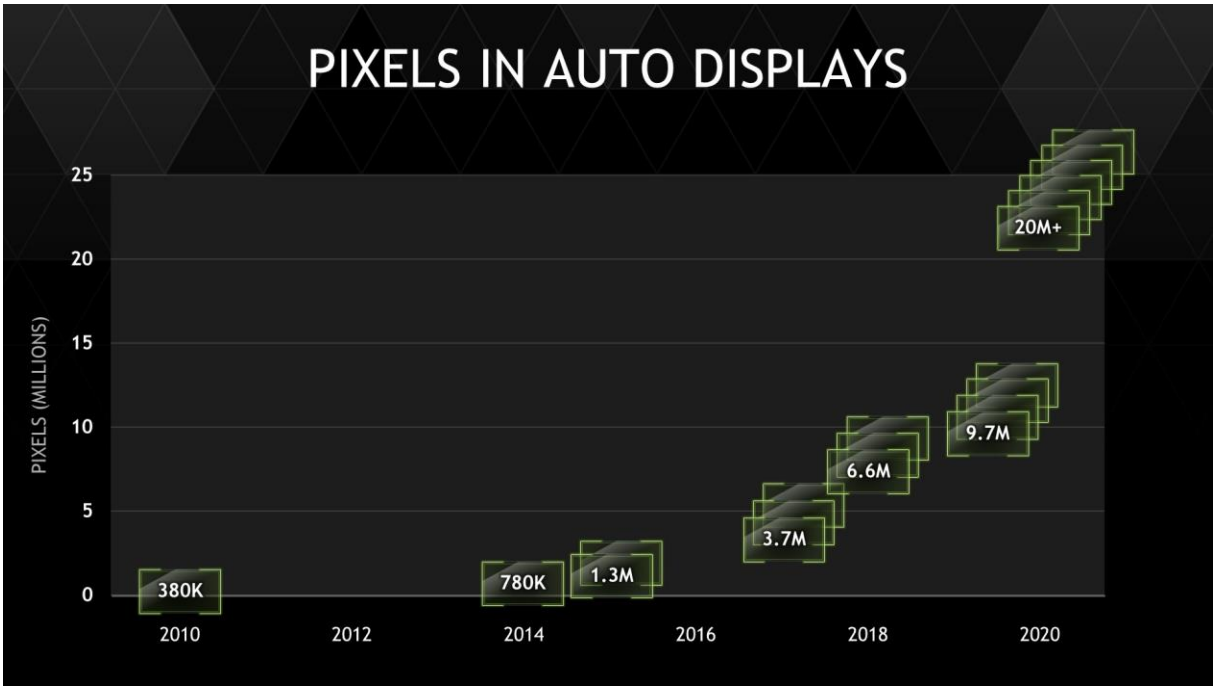

Figure 15 The number of display pixels car computers will have to drive is increasing rapidly

Newer car models coming out in the next few years will sport even more display panels for features such as camera-based virtual side view mirrors to eliminate drag caused by physical side view mirrors, virtual rear-view mirrors, passenger infotainment control, and even touch enabled digital displays for controlling windows and doors. A car computer may have to drive more than six high definition displays.

The number of display panels used in automobiles will continue to grow over the years, and as the above chart shows, the total number of pixels that needs to be rendered and displayed by the GPU in a car computer will continue to grow. For example, the car computer had to just drive 380 Kilo-pixels for a basic 800x480 resolution navigation display in 2010, and in 2014 it has to render 1.3 Mega-pixels to drive a couple of high resolution panels for navigation and digital clusters. By the year 2020 it will be possible to see cars using multiple HD display panels with a total display resolution exceeding 20 megapixels.

Alongside rendering and driving multiple HD displays, car computers also have to handle the processing of navigation, infotainment, camera views and digital cluster workloads. The design and implementation of car computers is becoming increasing complex, requiring GPUs and robust software stacks to handle these advanced features. The cost of development of these complex features is also increasing rapidly and is estimated to be in the range of $30M to $100M today.

The **NVIDIA DRIVE™ CX** cockpit computer is designed to deliver the graphics and compute performance needed to enable these advanced features, while significantly reducing the software development effort and costs for car manufacturers. NVIDIA DRIVE CX is a complete car computer built around the Tegra X1 processor that includes all the standard I/O interfaces such as Bluetooth, modems, audio, camera, and the standard interfaces required to connect the DRIVE CX system to the rest of the car control systems.

Figure 16 NVIDIA DRIVE CX Cockpit Computer

The NVIDIA DRIVE CX cockpit computer is capable of driving a total of 16.8 megapixels of display resolution, and capable of rendering advanced graphics to up to three displays. The system delivers the graphics performance required for 3D navigation rendering with realistic lighting effects, beautiful rendering of digital clusters, realistic rendering and compositing of a surround view based on multiple camera inputs, and enhancing video feeds from cameras to compensate for environmental conditions.

The system also comes with a complete road-tested software stack with software modules for features such as Photo-real Graphics, Surround Vision rendering, Advanced Navigation and more. The DRIVE CX hardware and software stack can be easily integrated into car designs, significantly accelerating a car maker’s or Tier 1 supplier's time-to-market and reducing software development costs.

The next two images show a few examples of how Tegra X1 and the DRIVE CX system can deliver a premium visual experience in cars. In Figure 17, the traditional two-dimensional navigation information shown on the left has uniform brightness across the map, and a lot of distracting features and text. Working with car manufacturers, NVIDIA has developed a new scheme to deliver navigation information that is graphically rich, three dimensional, and uses lighting effects to focus driver attention on only on

relevant regions of the map. The lighting effects brighten the area of the map that is relevant to the current path being navigated, and dims out the surrounding regions to reduce driver distraction while driving.

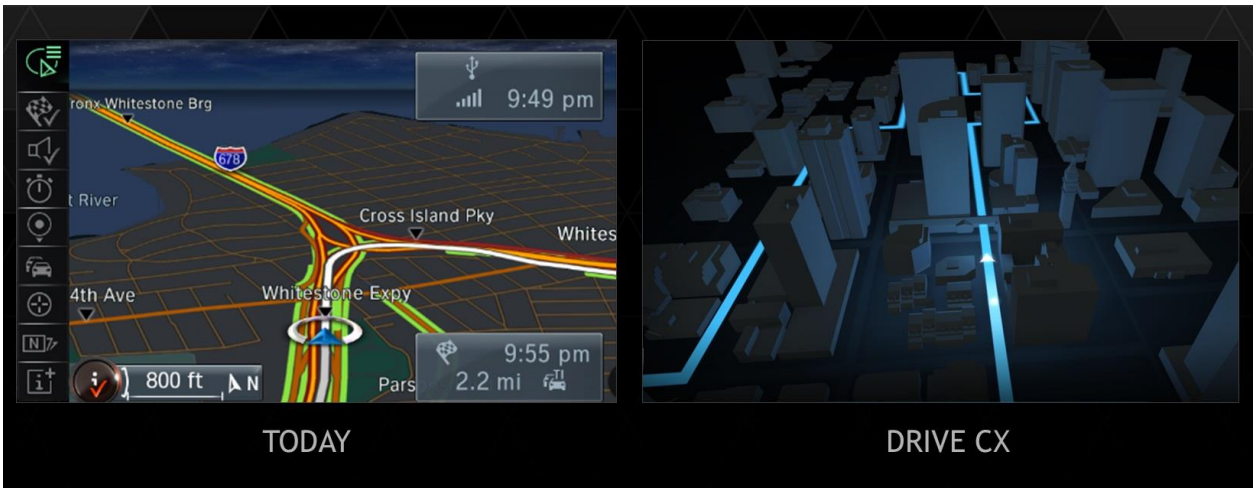

Figure 17 DRIVE CX delivers 3D navigation with advanced lighting effects to reduce driver distraction

Figure 18 illustrates an example of the beautiful three-dimensional rendering of speedometers, tachometers and other meters that are possible with the DRIVE CX platform. Feedback from car enthusiasts and automakers is that digital instrument clusters in cars today do not match the beauty and visual design philosophy of the rest of the car. Leveraging the immense graphics processing power of Tegra X1, carmakers can now deliver high quality, visually rich rendering of digital clusters that run at silky smooth frame rates and matches the premium physical designs of their cars.

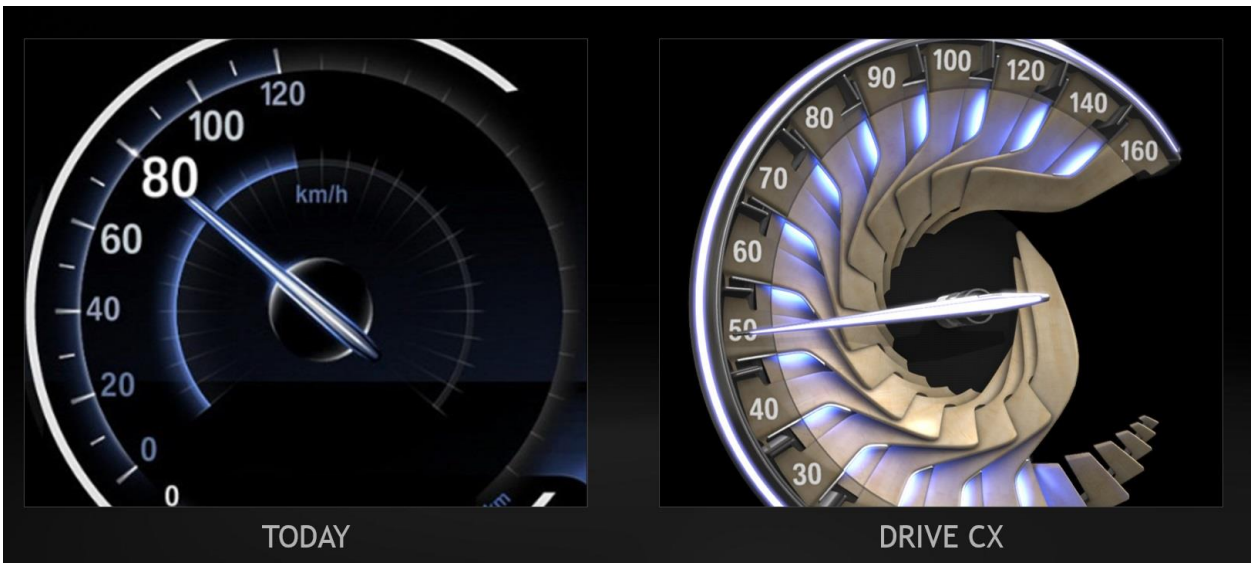

Figure 18 NVIDIA DRIVE CX delivers beautiful 3D rendering of digital clusters

NVIDIA DRIVE™ PX Auto-Pilot Platform

Along with including multiple display panels, newer cars are also designed with multiple onboard cameras to enable driver assistance, virtual mirrors, and surround view capabilities. Many car manufacturers are considering car designs that include multiple onboard cameras, and we can expect cars coming out in the next few years to include as many as twelve cameras, as shown in the image below. Cars will include cameras in the front, rear, and sides for surround vision, cameras to implement mirror-less side and rear view virtual mirrors, long and short range cameras in the front for driver assistance, corner cameras for cross-traffic detection, and a camera inside the car for driver monitoring.

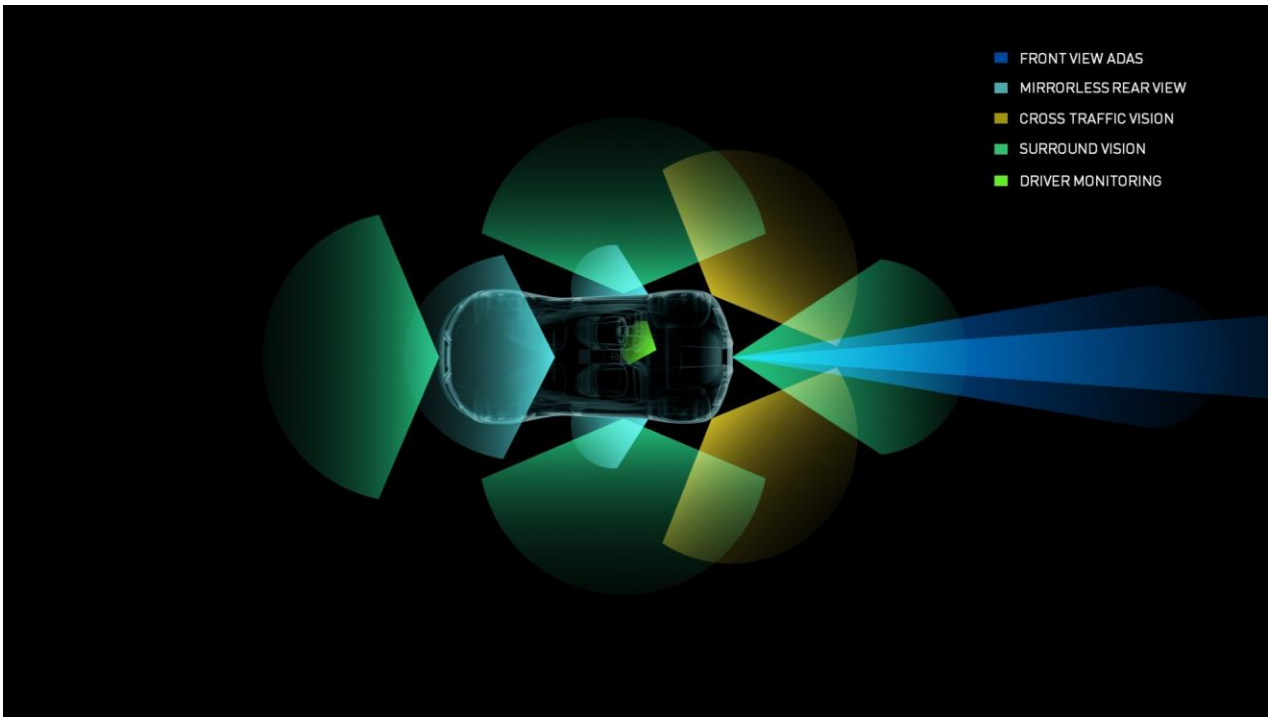

Figure 19 Next generation cars could have up to 12 cameras

Tremendous amounts of visual computing performance and image processing performance is required process, analyze, and act on the live high definition video streams being output by multiple cameras to deliver features such as surround vision, Auto-Valet parking, and contextual driver assistance. Many of these applications require hundreds of GFLOPS of compute power, and when the car computer is simultaneously working on several of these problems, the total workload could require TeraFLOPS of compute power.

Building highly accurate, reliable, and intelligent ADAS requires deep knowledge of parallel computing algorithms, neural computing, graphics, and expertise in graphics and writing complex software stacks that harness the power of the GPU. While car makers over the years have become experts in manufacturing cars that are better, cheaper, and more efficient, due to a lack of expertise in the areas of graphics processing, parallel processing and software development, car makers spend millions of dollars in the development of ADAS systems.

NVIDIA is the world leader in visual computing and has the expertise, experience, and knowledge to solve complex graphics and visual computing problems. The **NVIDIA DRIVE PX Auto-Pilot Platform** is a complete auto-pilot computing development platform that comes with software modules for advanced features such as surround vision, Auto-Valet parking, contextual driver assistance and more, at fraction of the cost to car makers. The DRIVE PX platform is designed to be a complete end-to-end solution for carmakers and Tier 1 suppliers, or elements can be licensed. The system is powered by two Tegra X1 processors that can be utilized to either work together, or with one of the processors serving as a redundant processor for critical applications.

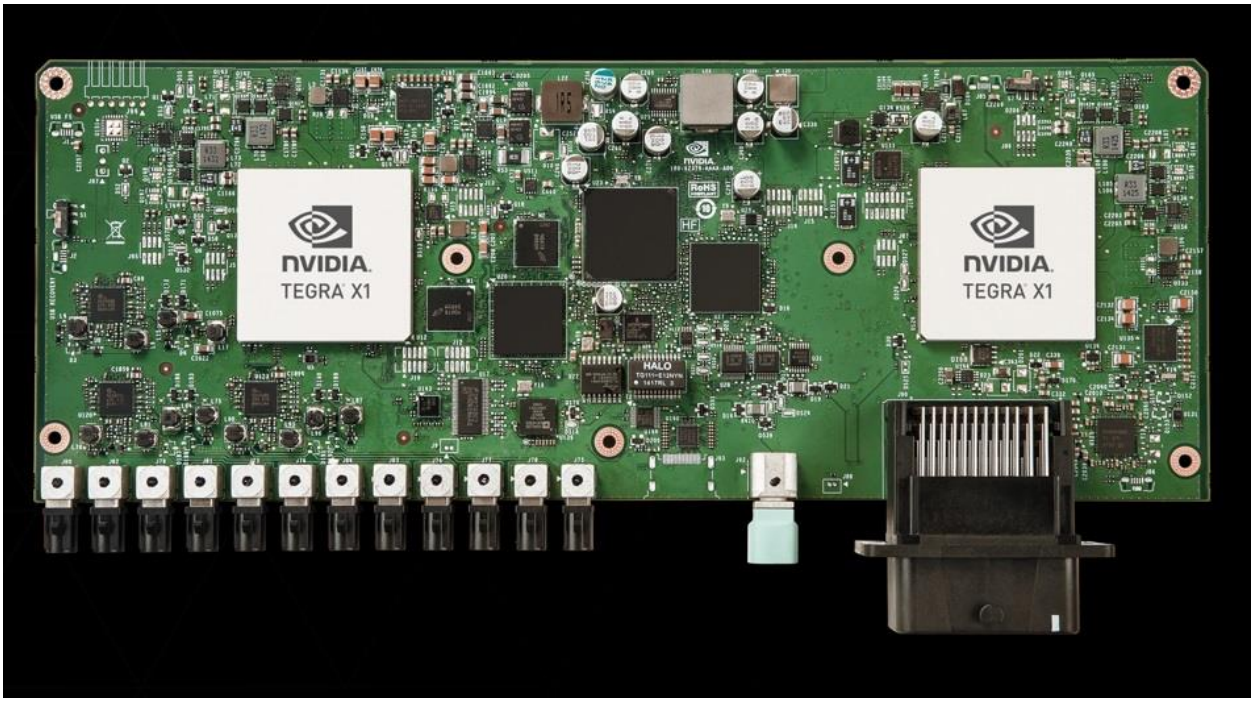

Figure 20 NVIDIA DRIVE PX Auto-Pilot Development platform

The DRIVE PX platform supports up to twelve camera inputs, and each Tegra X1 processor on the platform can access the data from the twelve onboard cameras and process it in real-time. Assuming each of these twelve cameras is a 1 Megapixel (1280x800) HD camera outputting at 30 fps, DRIVE PX will have to process 360 Mega-pixels per second of total video data. Since DRIVE PX has the ability to process 1.3 Gigapixels per second, it is capable of handling even higher resolution cameras outputting at higher frame rates. For computer vision-based applications, having higher resolution camera data at higher frame rates allows for faster and more accurate detection of objects in the incoming video streams.

DRIVE PX comes with a robust software stack and software modules that deliver accurate and visually rich surround vision capabilities, Auto-Valet parking capabilities, and a deep machine learning-based, increasingly smarter contextual driver assistance system.

Best in Class Surround Vision

The DRIVE PX platform comes with a Surround Vision software module that enables car manufacturers to deliver best in class surround vision experience. Current solutions in the market are limited in a number of ways. For example, in many surround view implementations, one will see lines extending out from the corners of the car displayed in the composited surround view image, and this is being done to hide the imperfect photo-stitching of the images being output by the four surround view cameras. The lines hide the anomalies and distortions introduced by the poor photo-stitching. Also traditional surround view solutions frequently suffer from ghosting and double vision when they are enabled in parking lots, and the parking lines in the stitched surround view image often appears twice, with other visual anomalies.

The DRIVE PX platform leverages the graphics and compute capabilities of Tegra X1 and sophisticated algorithms to deliver a premium surround vision experience. For example, as shown in the left hand side of Figure 21, when portions of the parking lot are sloping away from the car, DRIVE PX uses techniques such as ground-planing to detect slopes, and display a compensated stitched image that does not have any ghosting or visual anomalies. It also detects varying lighting conditions around the car and compensates for these differences when presenting the final stitched surround vision image. If for example, the camera on one side of the car is in bright sunlight, and the camera on the other side is in a shadow, DRIVE PX will compensate and display a stitched image that is evenly lit. Drive PX also beautifully renders the image of the car with advanced lighting, shadows, and high resolution textures to deliver a premium best-in-class surround vision experience as shown in Figure 21.

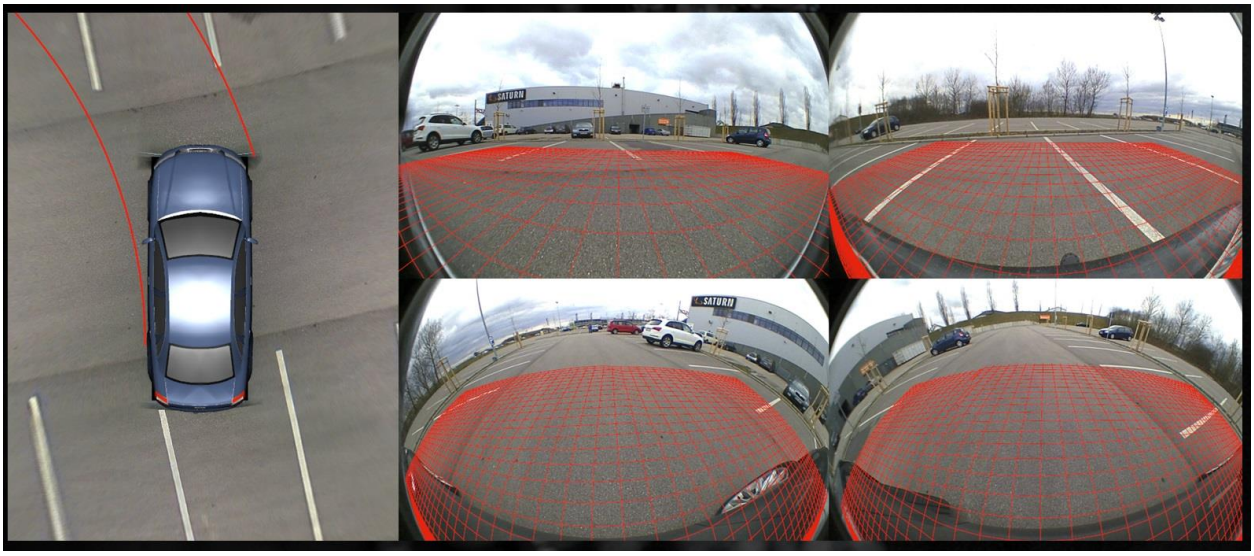

Figure 21 NVIDIA DRIVE PX delivers a best-in-class surround view solution

Autonomous Self-Parking

NVIDIA has developed a sophisticated Auto-Valet self-parking software module that runs on DRIVE PX and enables a car to autonomously drive through a parking lot, identify an empty parking spot, and safely park itself without any human intervention. The Auto-Valet software module uses advanced

algorithms to analyze and process the video streams from the front, rear, and side cameras of the car. First, it scans the videos delivered by the four cameras and uses a Structure From Motion (SFM) algorithm to create a point cloud 3D representation of the parking lot by detecting cars, empty spots, pillars, and other objects in the parking lot. This 3D representation of the parking garage is built in real-time as the car drives itself through the garage searching for an open spot and requires large amounts of compute performance.

Figure 22 NVIDIA DRIVE PX delivers computer vision based driverless self-parking capabilities

Second, after the Auto-Valet module identifies an empty spot, it needs to process the video feed to identify whether the empty spot is a valid parking spot. For example, if the spot does not have any parking lines, it may conclude that it is just an open space in the lot, or if it detects a handicap sign in a spot that has parking lines, it would mark it appropriately.

The Auto-Valet module also runs a path-planning algorithm to control the steering, brakes, and accelerator of the car while it is driving around the garage looking for a valid spot and while parking the car in the spot. After a valid spot has been identified, a parking algorithm detects obstacles and other cars around the parking spot and directs the car to either pull directly into the spot or execute the appropriate 3-point turn to park the car in the spot.

Tremendous amounts of compute performance and image processing power are required to implement the Auto-Valet feature, combined with a deep understanding of graphics, image processing and parallel computing to implement the algorithms and software stacks to deliver a safe, reliable and efficient self-parking capability. NVIDIA is able to deliver this sophisticated feature thanks to its expertise in all these

areas and its years of experience in researching, inventing and innovating in the areas of GPU compute, graphics processing, and computer vision. The NVIDIA DRIVE PX platform and software modules such as Surround Vision and Auto-Valet deliver tremendous software development cost savings to car manufacturers and enables them to quickly add advanced features to their cars.

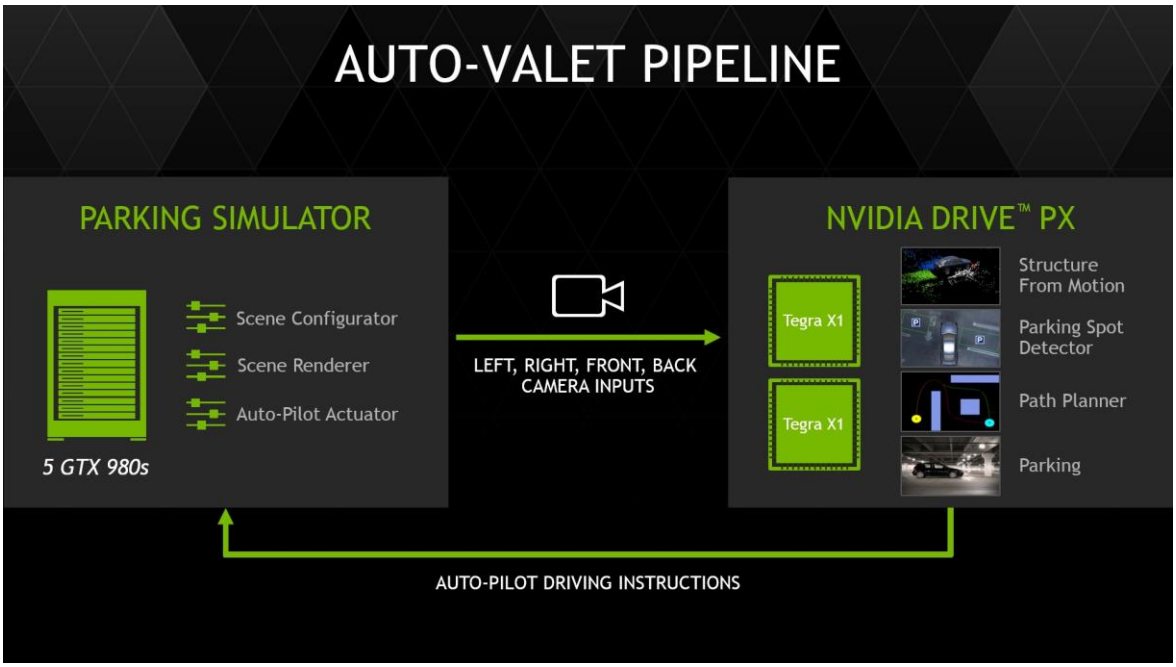

Figure 23 NVIDIA DRIVE PX Parking Simulator System

To enable even faster development and optimization of self-parking capabilities in cars, NVIDIA has created a complete Auto-Valet parking simulator system that allows car manufacturers to test and refine their self-parking algorithms under various physical and environmental situations. The parking simulator as shown in the image below has a scene configurator that can create various configurations of parking garages, add or remove cars in parking spots, or even create obstacles and other interesting features such as handicap zones and no-parking zones.

The scene renderer of the simulator uses a given scene configuration and then the piloted car driving instructions are fed back from the DRIVE PX module under test to render the videos that simulate the outputs of the front, rear, and side cameras of the car. These videos are then fed to the DRIVE PX module that runs the Auto-Valet algorithm and outputs the driving commands to control the brakes, steering, and accelerator. These driving commands can be analyzed to further optimize the various self-parking algorithms.

Contextual Advanced Driver Assistance Systems

Advanced Driver Assistance Systems (ADAS) in cars as the name suggests detects objects on the road such as other cars, pedestrians, and traffic signs to either warn the driver or control the car for collision avoidance, lane deviations, and traffic conformance. However, current generation ADAS have several limitations and do not deliver a satisfactory assistance to drivers.

First, due to limited compute power, these systems use videos from low resolution, low frame rate cameras to process and provide driver assistance. Lower resolution images often result in inaccurate identification and classification of objects in the scene, and slow responses to hazards on the road. Second, these systems are able to identify only a basic set of objects such pedestrians, cars, and traffic signals that are clearly visible and appear as familiar structures or outlines to the ADAS. But, they are unable to identify objects that look different when the viewpoint changes as shown in the below figure.

Figure 24 Current ADAS struggle to identify the same vehicle when it viewed from a different viewpoint

They also often fail in cases where objects are partially occluded from view. For example, they may fail to identify and classify pedestrians that are only partially visible to the camera, or pedestrians who are pushing a bicycle along with them. Third, many of these assistance features frequently fail under adverse environmental or lighting conditions such as rain, night time, and bright reflections in the screen.

Figure 25 Current ADAS often fail to identify pedestrians that are occluded by other objects in the scene[3]

[3] Image source: *Partially Occluded Pedestrian Dataset, Computer Vision Center, Universitat Autonoma de Barcelona*

Finally and most importantly, current generation ADAS do not understand the context of the captured scene. For example, as humans we would be more wary and drive more carefully when we see a distracted pedestrian who is crossing the road while talking on the phone, as compared to when we see a pedestrian who is aware of an approaching car and crosses the road more carefully. Also, current ADAS are not able to differentiate between objects that are similarly shaped, but require differing responses.

Figure 26 Current ADAS often struggle to differentiate between Ambulances and similar looking trucks

For example, since many trucks look like ambulances as shown in the above figure, the ADAS may identify an ambulance on the road as a truck and not reduce the speed of the car. Therefore to deliver an system that has full auto-piloting capabilities, we need an solution that overcomes all the above limitations.

The dual Tegra X1 powered DRIVE PX system has the compute performance and software sophistication to solve the above limitations, and is capable of handling higher resolution, high frame rate video streams, compensate for adverse lighting conditions, and identify more objects in a scene at a much faster pace. But in order to solve the problem of occluded objects, and to contextually classify objects, the ADAS needs to have the ability to recognize several millions of shapes and objects for correct classification in real-time, and would require Teraflops of compute power that just cannot be implemented locally on a small local system running traditional computer vision algorithms.

Deep Learning and Neural Networks

To address the complex problems detailed above, NVIDIA is bringing Deep Learning capabilities to the automotive industry through the DRIVE PX platform. Deep Learning in a nutshell is a technique that models the neural learning process of the human brain that continuously learns, and continuously gets smarter over time to deliver more accurate and faster results. Just as a child is initially taught by an adult to correctly identify and classify various shapes, eventually being able to identify shapes without any coaching, a deep learning or neural learning system has to be trained in object recognition and classification for it get smarter and more efficient at identifying basic objects and occluded objects, while also assigning context to objects.

As mentioned previously, deep learning systems are modeled on how the human brain works. At the simplest level, neurons in the human brain look at various inputs fed into it, then assigns to each of

these inputs appropriate importance levels that have been learned over time, and delivers an output that is passed on to other neurons to act on.

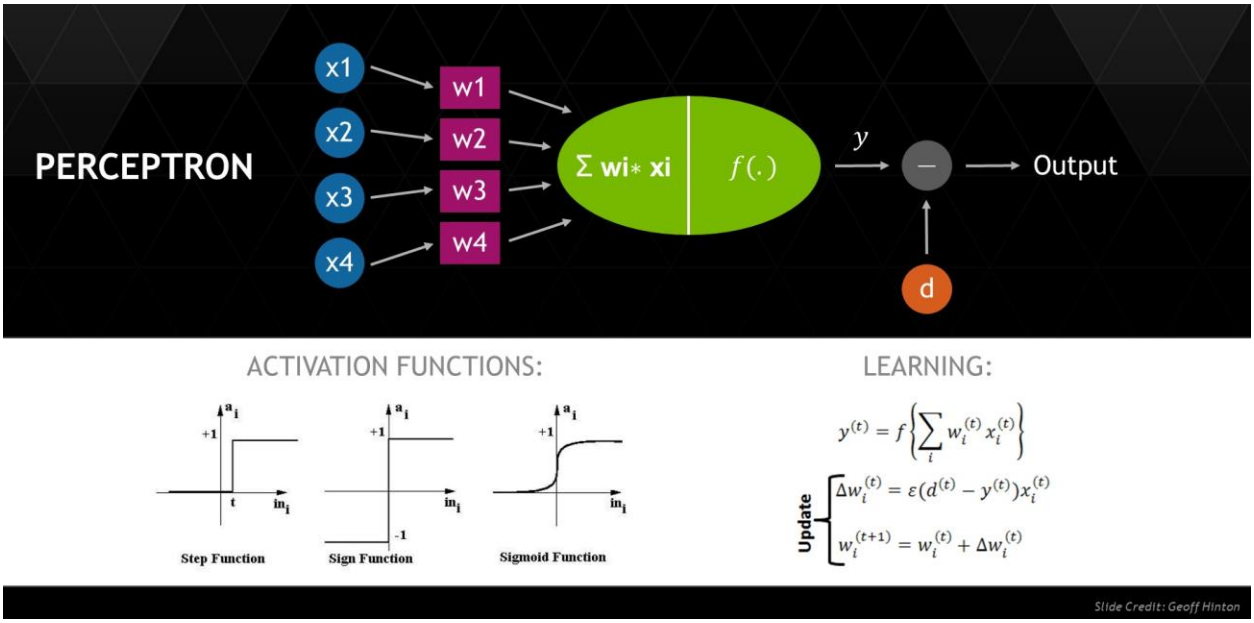

Figure 27 The Perceptron is the simplest model of a neural network

The Perceptron as shown in the above figure is the most basic model of a neural network. As seen in the image, the Perceptron has several inputs that represent various features of an object that the Perceptron is being trained to recognize and classify, and each of these features is assigned a certain weight based on the importance of that feature in defining the shape of an object.

For example, consider a Perceptron that is being trained to identify the number zero that is handwritten. Obviously, the number zero can be written in many different ways based on different handwriting styles. The Perceptron will take the image of the number zero, decompose it into various sections and assign these sections to features x1 through x4. The upper right hand curve in the number zero may be assigned to x1, the lower bottom curve to x2, and so on. The weight associated with a particular feature determines how important that feature is in correctly determining whether the handwritten number is a zero. The green blob at the center of the image is where the Perceptron is calculating the weighted sum of all the features in the image to determine whether the number is a zero. A function is then applied on this result to output a true or false value on whether the number is a zero.

The key aspect of a neural network is in training the network to make better predictions. The above Perceptron model to detect handwritten zeros is trained by initially assigning a set of weights to each of the features that define the number zero. The Perceptron is then provided with the number zero to check whether it correctly identifies the number, if it does not correctly identify the number, then the reason for the incorrect identification needs to be understood, and the weights need to adjusted for each feature until the perceptron correctly identifies a zero, and the weights have to be further adjusted until it correctly identifies zeros written in various handwriting styles. The equations shown in the

diagram look complex, but are basically mathematical representations of the above described training process.

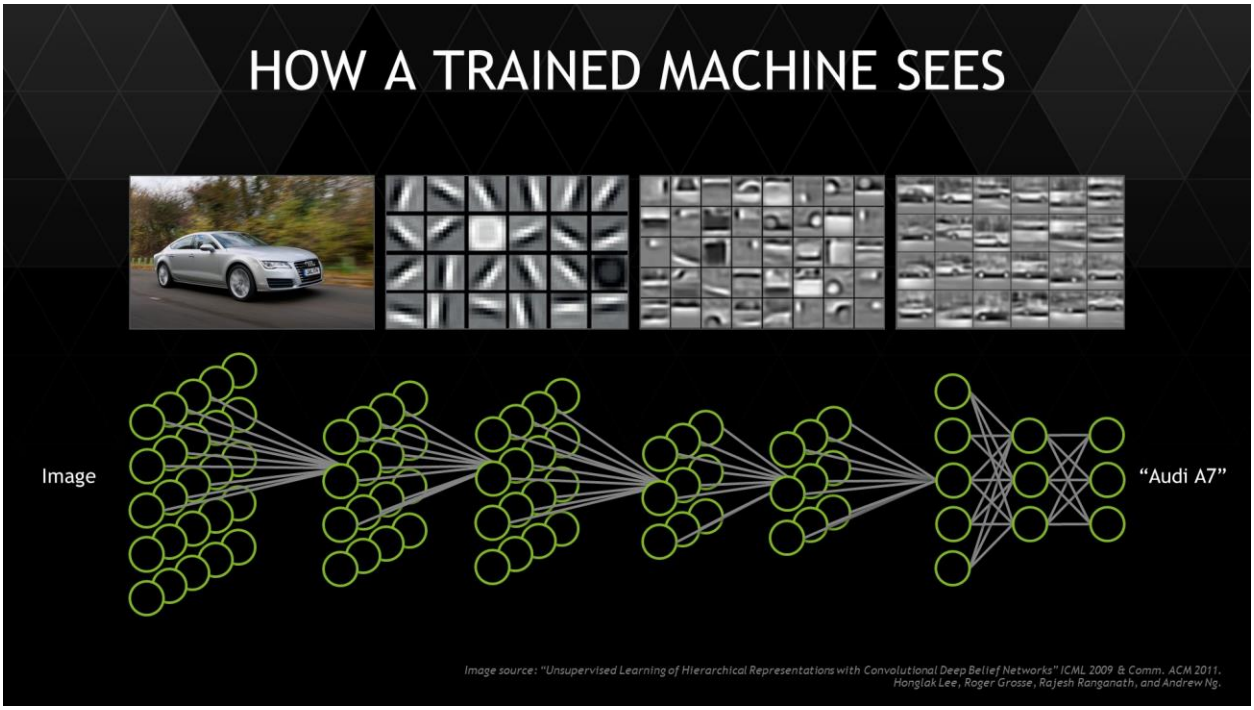

Figure 28 Complex multi-layer neural network models require lots of compute power

Though the Perceptron is a very simple model of a neural network, advanced multi-layered neural networks based on similar concepts are widely used today to identify handwritten numbers on checks deposited into ATM machines, identify images of friends in Facebook photos, deliver movie recommendations to over fifty million Netflix users, and in many more use cases. A multi-layered neural network model as shown in figure 28 may consist of multiple interconnected complex Perceptron-like nodes, with each node looking at number of input features and feeding its output to several other nodes.

In the model above, the first layer of the neural model breaks down the image into various sections and looks for basic patterns such as lines and angles, the second layer assembles these lines to look for higher level patterns such as wheels, windshields, and mirrors, the next layer identifies the type of vehicle, and the final few layers of the neural model identify the model of a specific brand (which in this case is an Audi A7).

GPU Acceleration of Neural Network Models

Neural networks and deep learning ideas have existed for decades, and only recently a group of scientists at Stanford University discovered that the training and learning process for a neural network model can be significantly accelerated if the model is written to be parallelized and run on GPUs that have thousands of parallel processing cores.

The three major breakthroughs that are currently revolutionizing the use of neural network-based computing in various industries are the use of GPUs as parallel processing supercomputers to train complex neural network models, the availability of Big Data methods, and the rapid advances in neural network algorithms.

ImageNet is a popular worldwide contest to evaluate effectiveness and efficiency of image processing neural networks. Due to the tremendous performance acceleration delivered by GPUs, the number of GPU based neural models submitted at the ImageNet competition has exponentially increased over the last few years, and in 2014 more than 95% of the entries were optimized to run on GPUs.

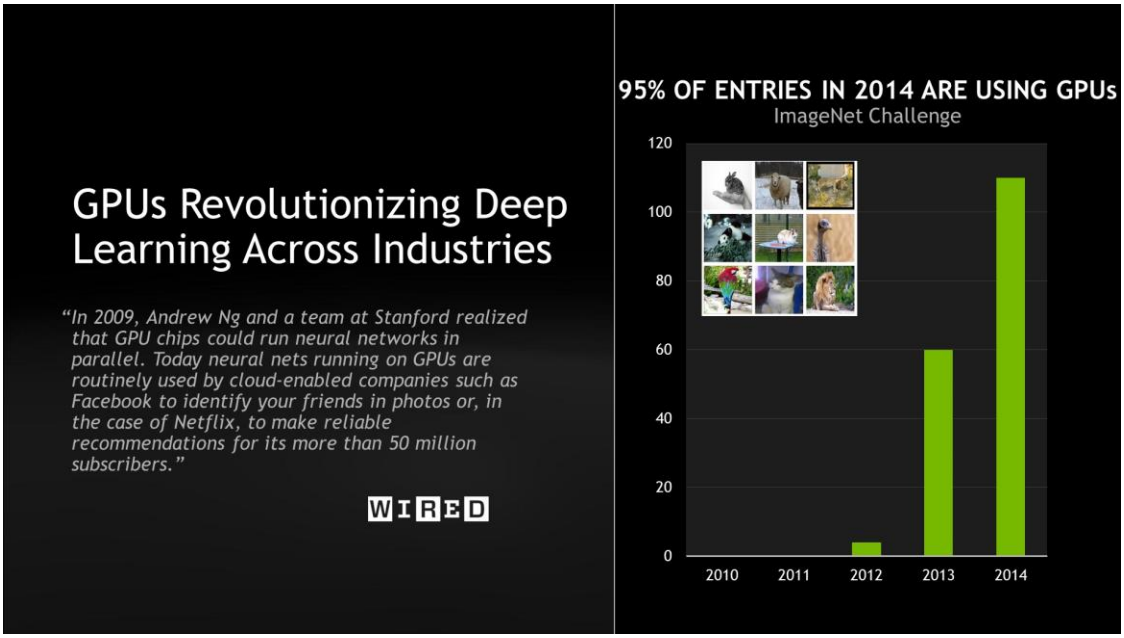

Figure 29 Majority of neural network models are optimized to run on GPUs

Tegra X1 with its 256-core Maxwell GPU is designed to deliver the compute performance required to run both traditional computer vision algorithms and also neural network algorithms. The charts below illustrate the performance of Tegra X1 on commonly used computer vision and neural network algorithms. Figure 28 shows the performance of Tegra X1 on image filtering algorithms and edge detection algorithms, with Tegra X1 delivering 1.5x to 2x the performance of Tegra K1. The next two figures show that Tegra X1 delivers around two times the performance of Tegra K1 on popular neural network models used for object classification.

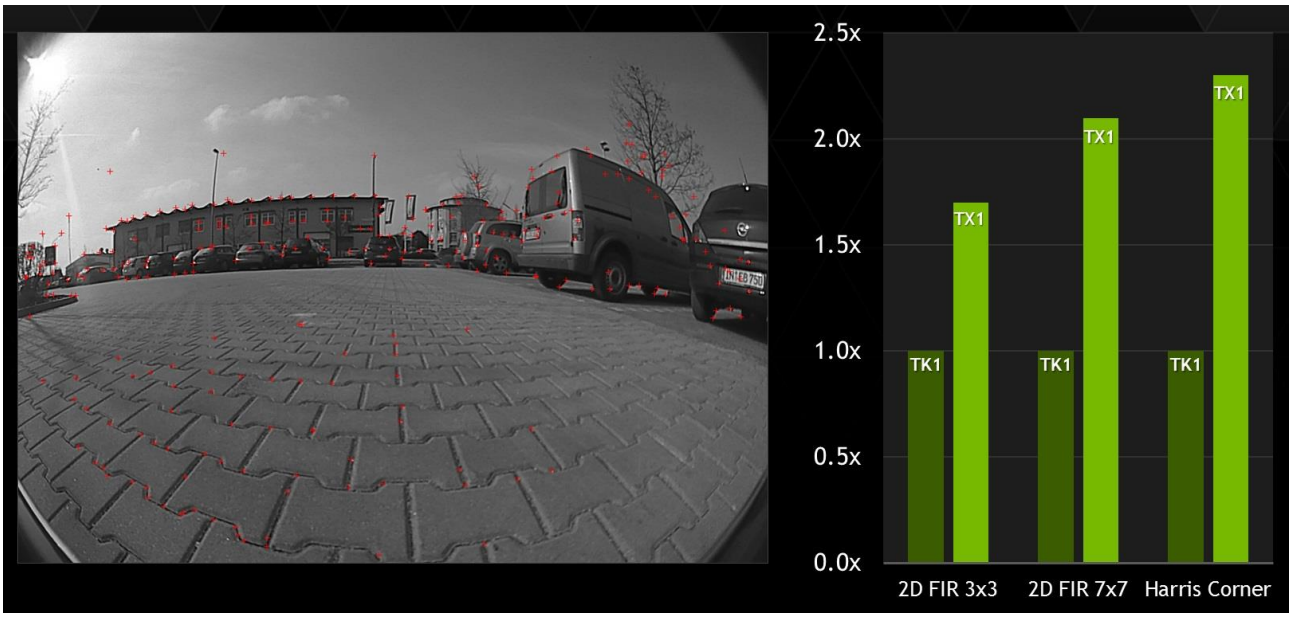

Figure 30 Tegra X1 delivers 2x the performance of Tegra K1 for computer vision workloads

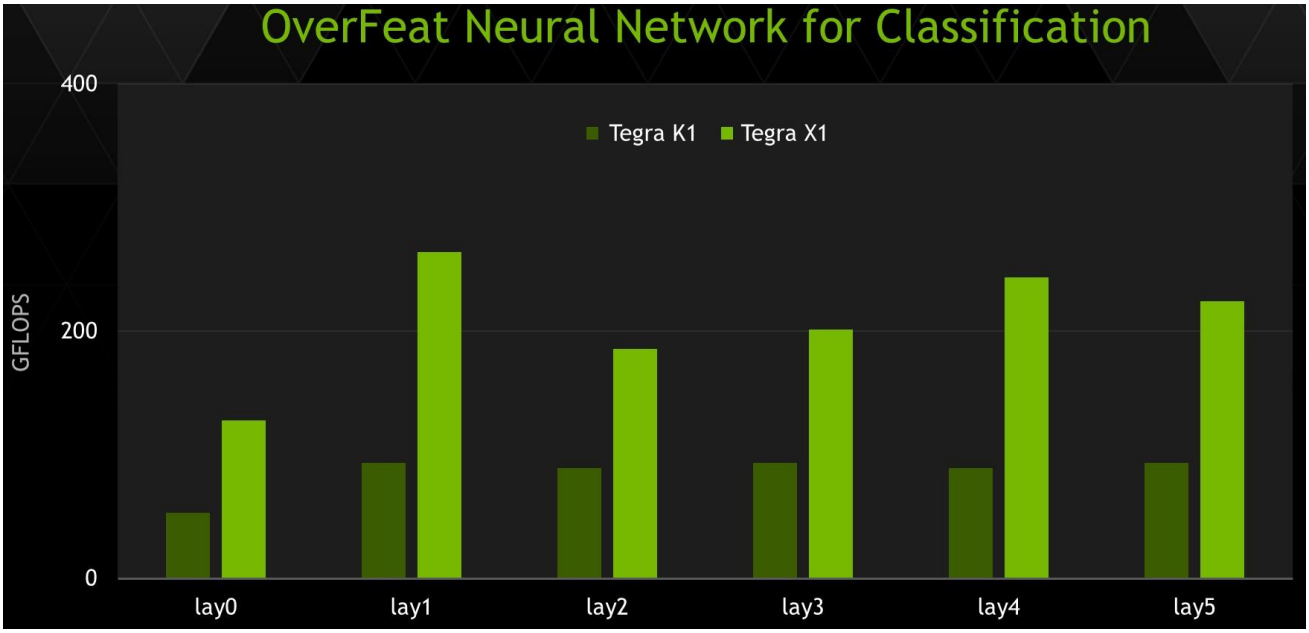

Figure 31 Performance of Tegra X1 on the OverFeat Neural Model used for Object classification

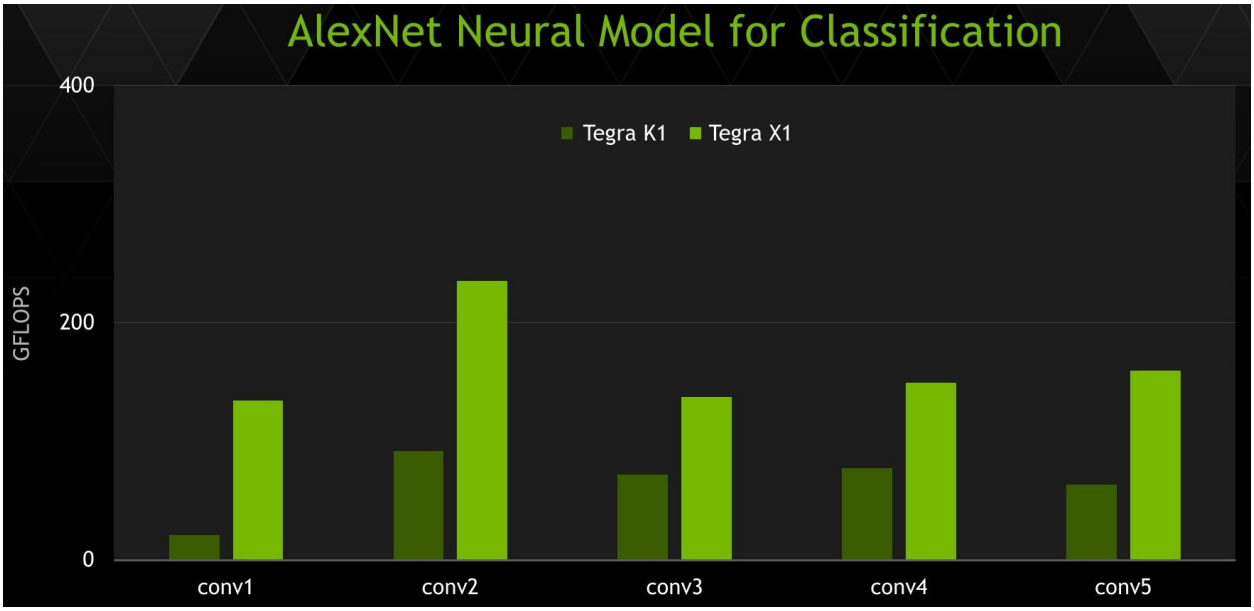

Figure 32 Performance of Tegra X1 on the Alexnet Neural Model used for Object classification

NVIDIA DRIVE PX Brings Deep Learning to Automobiles

Having understood how neural network models work and how GPUs are delivering tremendous performance boost in running these neural models, it is easy to see why NVIDIA is uniquely positioned to bring this exciting new technology to automobiles and deliver full auto-piloting capabilities to the next generation of cars.

The neural network-based deep learning capabilities as applied to computer vision and auto-piloting would require the model to correctly and contextually identify the various objects in a complex scene of a road as shown in the image below. The model has to correctly identify distracted pedestrians, occluded pedestrians and vehicles, traffic cones, construction vehicles, closed lanes, traffic signals, and other features in the scene. Training a neural network-based model to accurately identify these objects in a complex scene requires hundreds of TeraFLOPS and certainly cannot be done locally on the DRIVE PX system in an automobile. But once the model is sufficiently trained to identify and react to these complex road conditions, the trained, complex multi-layer neural network model can easily run in real-time on the highly parallel 256-core Tegra X1 processors in the DRIVE PX platform.

The development and training of the neural network model required to deliver object identification and classification is done on high performance NVIDIA Tesla GPU-based supercomputers. Typically data scientists would develop millions of images of various objects a car would encounter on a road and tweak the model until it can accurately detect all these objects. Once the car manufacturer is satisfied with the performance, accuracy, safety, and reliability of the neural model, it is installed on the NVIDIA DRIVE PX based ADAS and deployed in cars in the real world.

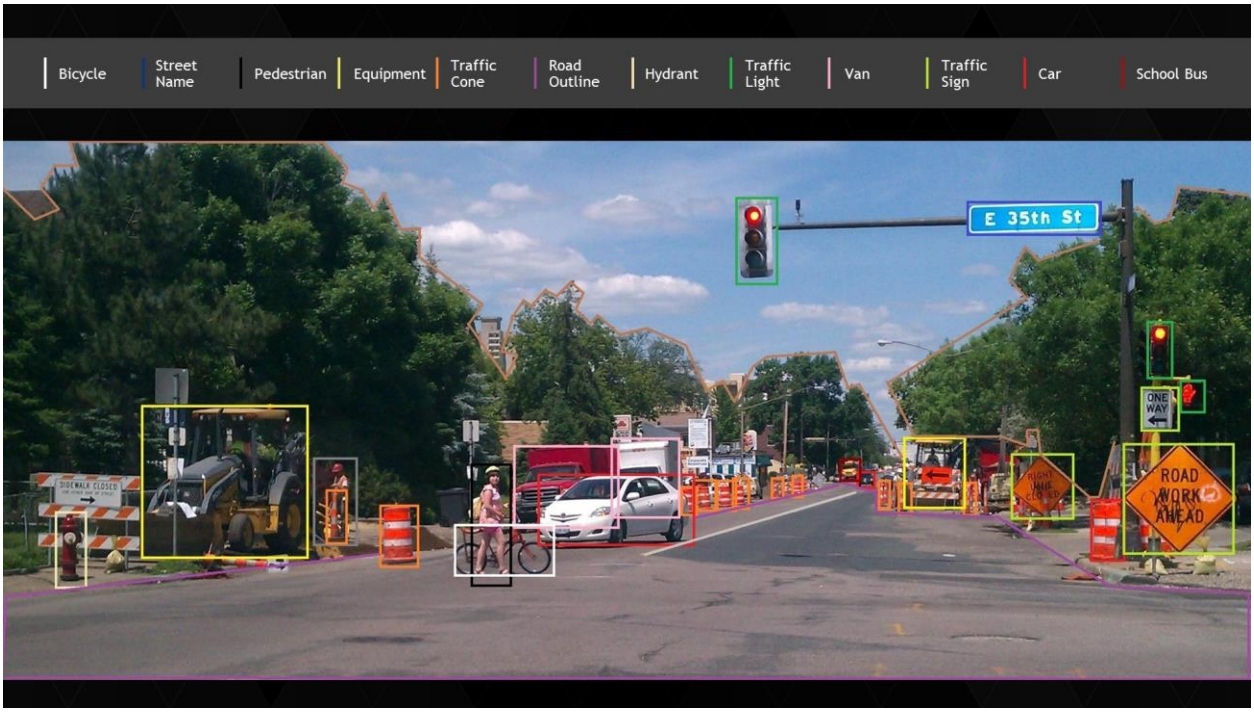

Figure 33 Complex scenes require Deep Learning-based object identification and classification

The cool part of this unique NVIDIA Tesla GPU supercomputer-based training in the cloud, and NVIDIA Tegra X1-based identification and classification on the road, is that the accuracy and scope of the neural model can be continually enhanced even after it is deployed on the road.

For example, a DRIVE PX-based auto-pilot system running in a car may not be able to identify a particular type of object it encounters, or other car system data may conflict with its identification and classification. In such cases, the system can be programmed to store such events along with videos and images captured by the cameras on the car, and then have the system upload this data to the cloud over the air, or a service technician can download it during service checkups and later upload it to the cloud. The data scientists working on refining the model can then observe these uploaded corner cases where the model failed and appropriately tune the neural model. The tuned and smarter neural model is then pushed out to the DRIVE PX-based fleet via an OTA (Over the Air) update, or installed during a service checkup.

With potentially hundreds of thousands of cars on the road running the neural model and submitting each of their failed identification or classifications, the neural network model can very quickly learn to identify objects in even more complex scenes and the DRIVE PX based ADAS will be able to deliver extremely reliable, accurate, and true auto-piloting capability.

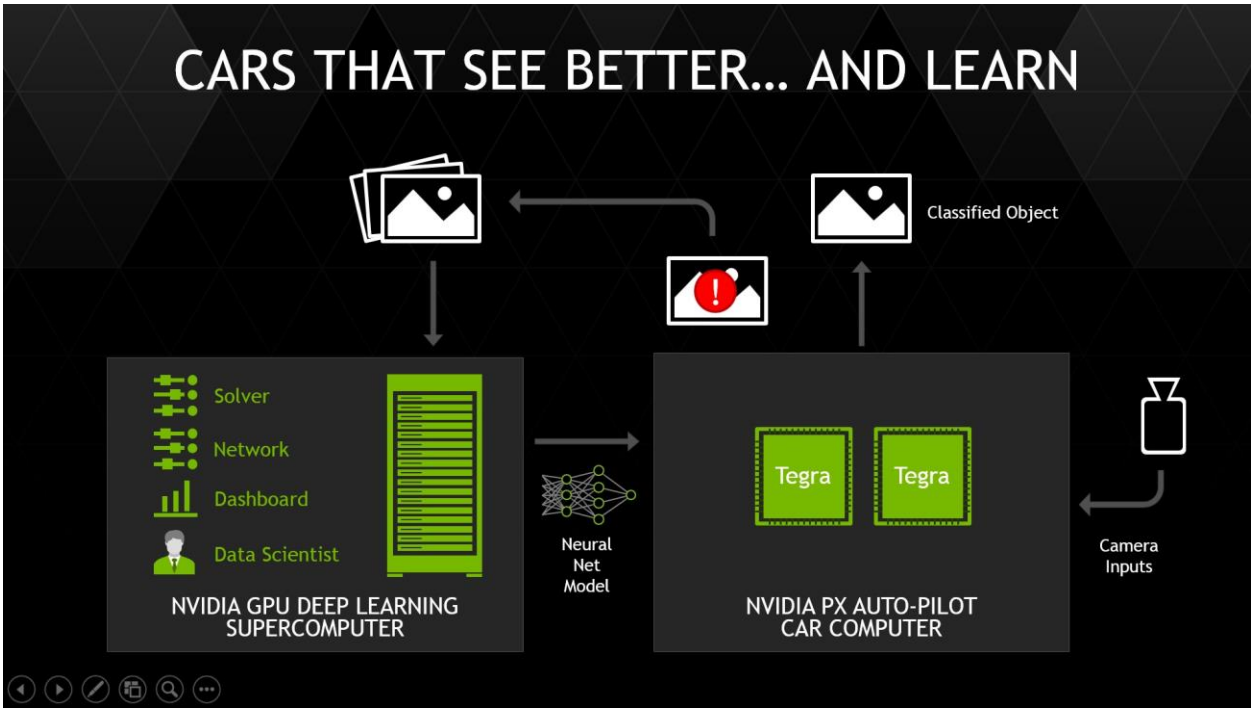

Figure 34 Training neural net models on Tesla supercomputers and running them in cars using DRIVE PX based ADAS

The NVIDIA DRIVE PX platform delivers a significant breakthrough for advanced driver assistance systems and brings deep learning technologies to automobiles, while the NVIDIA DRIVE CX cockpit computer delivers advanced cockpit visualization features such as graphically rich digital clusters, virtual mirrors, and high resolution infotainment panels. Both platforms are powered by Tegra X1 processors that deliver class-leading compute performance and incredible energy efficiency. Both the DRIVE PX and DRIVE CX platforms will be available to car manufacturers in the second quarter of 2015 and production ready systems will be available in 2016.

Conclusion

Last year NVIDIA Tegra K1 revolutionized the mobile industry by bring the power of NVIDIA's Kepler GPU architecture to mobile. Tegra K1 delivered unmatched performance, energy efficiency and a rich set of desktop graphics and compute APIs. Tegra X1 is NVIDIA's latest superchip, and has raised the bar again for mobile visual computing and energy-efficiency. Tegra X1 with its 256-core Maxwell GPU delivers 2x the performance and 2x the energy efficiency of Tegra K1. Tegra X1 supports the latest graphics APIs and further blurs the lines separating mobile, console and PC gaming. Recognizing the rapid growth of 4K TVs and content, Tegra X1 is optimized end-to-end to deliver a premium 4K experience and is the only mobile processor that delivers 4K at 60fps and supports 4K 10 bit color.

NVIDIA is the world leader in visual computing and is uniquely positioned to deliver new solutions and technologies that leverage our deep expertise in the areas of graphics, computer vision, parallel computing and software development. Recognizing the challenges faced by the auto industry in

bringing advanced visualization and driver assistance features to the next generation of cars, NVIDIA has developed the DRIVE line of car computers: DRIVE CX Cockpit visualization platform and the DRIVE PX Auto-Pilot development platform.

Powered by NVIDIA Tegra X1, the NVIDIA DRIVE CX cockpit computer delivers advanced cockpit visualization features such as graphically rich instrument clusters, virtual mirrors and support for multiple high definition display panels. The DRIVE CX is a complete car computer that comes with all the necessary hardware interfaces and software stacks necessary for a car maker to easily integrate into their car designs at a fraction of their current development costs. The NVIDIA DRIVE PX Auto-Pilot Development platform brings exciting new capabilities for car ADAS systems such as advanced Surround Vision, self-parking and a never before seen Deep Learning based auto-piloting technology that is set to revolutionize the auto industry. Both the DRIVE CX and DRIVE PX platforms come with road-tested software stacks along with software modules for advanced features such as Surround Vision, self-parking and auto-pilot, saving car manufacturers significant software development costs and accelerating the time-to-market of their next generation cars.

The energy efficiency, general purpose programmability and the TeraFLOPS compute performance of Tegra X1 paves the way for embedded, automotive, robotics and computer vision industries to bring exciting new applications to their respective fields.

NVIDIA Tegra X1 SoC Specifications

Processor	
CPU	Quad 64-bit A57 cores + Quad 64-bit A53 cores
Cache	Cortex A57 cluster: 2 MB Shared L2 Cache, 48KB /32KB (I/D) L1 Cache per core Cortex A53 cluster: 512KB shared L2 Cache, 32KB/32KB (I/D) L1 Cache per core
Memory	
Frequency	LPDDR3, LPDDR4-1600, 64-bit (25.6 GB/s)
Memory Size	Up to 4 GB
GPU	
Cores	256-core Maxwell GPU with support for FP16
API Support	OpenGL ES 3.1, OpenGL4.5, DirectX 12.0, AEP, CUDA 6.0
Video	
Decode	VP9, H.265, H.264 4K 60 fps; H.265 4K 60fps 10-bit color; VP8 1080p 60fps;
Encode	H.264, H.265 4K 30 fps; VP8 1080p 60 fps;
Imaging	
Image Processing	Dual ISP, 1.3 GigaPixels/s, 4096 focus points, 100 MP Sensor support, up to 6 camera inputs
JPEG Decode/Encode	600 MPixels/s
Display	
Display Controllers	2 Simultaneous
HDMI	HDMI 2.0, HDCP 2.2, 4K 60 fps
Local Display	4K 60 fps VESA DSC compression
Storage	
Storage interface	e-MMC 5.1 (HS533), CMD Queuing

Document Revision History

- Initial release 1.0

Notice

ALL INFORMATION PROVIDED IN THIS WHITE PAPER, INCLUDING COMMENTARY, OPINION, 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 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 or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

NVIDIA, the NVIDIA logo, Chimera, Tegra, TegraZone are trademarks or registered trademarks of NVIDIA Corporation in the United States and other countries. Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright