International Journal of Electrical and
Electronics Engineering (IJEEE)
ISSN(P): 2278-9944; ISSN(E): 2278-9952
Vol. 8, Issue 1, Dec - Jan 2019, 1-20
© IASET

International Academy of Science,
Engineering and Technology
Connecting Researchers; Nurturing Innovations

STREAMING MULTIPROCESSOR ARCHITECTURE, TENSOR CORES AND NVIDIANVLINK OPTIMIZED FOR DEEP LEARNING AND HIGH PERFORMANCE COMPUTING (NVIDIA TESLA V100&NVIDIA VOLTA V100)

Omkar Rajesh Kachare

Research Scholar, Department of Electrical and Electronics Engineering,

California State University, Sacramento, CA, USA

ABSTRACT

AI is not defined by any one industry. It exists in the fields of supercomputing, healthcare, financial services, big data analytics and gaming. It is the future of every industry and market because every enterprise needs intelligence. Modern High Performance Computers (HPC) data centers are the key to solving some of the world's most important scientific and engineering challenges. NVidia Tesla V100 GPU accelerated computing platform powers these data centers, with industry leading applications to accelerate HPC and AI workloads.[8] The features of the V100 GPU are,

- *New Streaming Multiprocessor (SM) Architecture optimized for deep learning.*
- *NVidiaNVLink fabric.*
- *150 Teraflop per second.*
- *640 Tensor cores.*
- *Paired NVidia CUDA and Tensor cores to deliver high performance.*

The main objective of this paper is to study NVidiaNVLink fabric and Tensor cores, which are used for transfer of data from CPU and GPU, and Streaming Multiprocessors Architecture optimization for deep learning.

The rapid growth in deep learning workloads has driven the need for a faster and more scalable interconnect, as PCIe bandwidth increasingly becomes the bottleneck at the multi-GPU system level. NVidiaNVLink technology addresses the interconnect issues by providing higher bandwidth, more links and improvises stability for multi- GPU and multi GPU-CPU system configurations. A single NVidia tesla V100 GPU supports up to 6 NVLink connections with a total bandwidth of 300GB/sec, which is 10x the bandwidth of PCIe gen 3.[11]

KEYWORDS: *GPUs the best choice for computing deep neural network based applications and machine learning applications*

Article History

Received: *04 Nov 2018* **|** ***Revised:*** *14 Nov 2018* **|** ***Accepted:*** *27 Nov 2018*

INTRODUCTION

Gordon E. Moore, while working as the director of Research and Development at Fairchild Semiconductor, as asked to predict what could be the future of the semiconductor industry in next 10 years. A brief response by him in an article entitled, “Cramming more components onto integrated circuits”, gave rise to Moore’s Law.

Moore’s Law could be stated as “The number of transistors in a densely integrated circuit doubles every year”.[1] This projected rate of growth in the number of transistors would continue to double up every two years, for at least another decade. In 1957, he revised his prediction to stating “The Number of transistors will double up every two years”.

Moore’s Law held itself obsolete for several decades. It used in the semiconductor industry to guide the research and development to set their targets and plans. Moore’s law, not only limited to the number of transistors alone. It can be observed in digital electronics too, and rather in a very prominent way. Microprocessor’s prices, Digital memory capacity, sensors, screen pixel density and even the size of pixels in a digital camera.

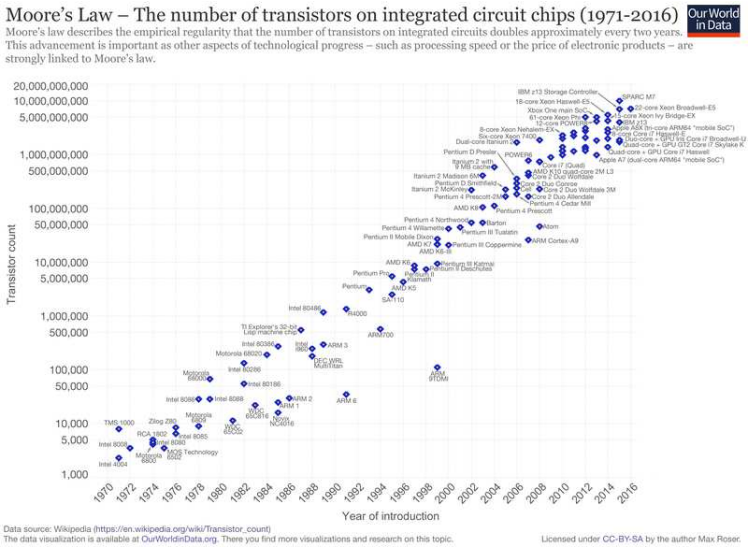


Figure 1: Moore’s Law.

Moore’s law is just an observation of a trend and not a law. The rate of doubling of transistors was held true until 2012. After which it was observed to be failing.

The Number of transistors could be doubled up in an integrated circuit because of scaling down of the sizes of transistors. In 2016, transistors of 10 nanometerswere designed by Intel. Intel’s current Ivy Bridge and Haswell processors are based on 22 nanometer process. Intel states that the Moore’s law could be carried forward for only five more years, as Intel is already working on the 7 nanometer process. This level of reduction in transistor size gives rise to new issues. The device physics at this scale does not work as expected. Hence, there is a plateauing in the Moore’s law prediction.[3]

GPU Computing

So, will this be an end of semiconductor industry? Not Likely. One of the emerging technologies to tackle this issue is using a Graphic Processing Unit accelerated CPU. Graphic Processor Unit (GPU) Computing is the use of a GPU as a co-processor to enhance and accelerate CPUs for scientific, engineering or even for general purpose.

The GPU augments the CPU acceleration by sharing or offloading the heavy, compute- intense, and time consuming tasks. The task, though, still runs on the CPU primarily. From the users view, the task is executed faster because of the massive parallel processing power provided by the GPU. This parallel processing boosts the CPU and GPU performance. This is called "heterogeneous" or "Hybrid" computing.

A CPU consists of few cores, usually four to eight. A GPU on the other hand consists of hundreds of smaller cores. In a conjunction, CPU and GPU crunch through the data in any application. This massive parallel executing architecture gives a GPU higher computing performance. This GPU accelerated CPUs paves a way for High Performance Computing (HPC).

History of GPU Computing

Initially, the sole purpose of the GPU was only to process the visual data. Over the time period, these graphic chips became extremely programmable. This made NVidia to launch the first GPU. During 2000-2001, scientists, along with the researches in the field of Electromagnetics and medical imaging started using GPUs to accelerate the range of their scientific applications. This gave a rise to General Purpose Graphical Processing Units, GPGPUs. The GPUs previously required only OpenGL and Cg to program them; this was a bottle neck as these languages were not that prominent. NVidia realized the potential that the GPU packed and invested in modifying GPU to make it fully programmable. Support for high level languages like C, C++ was added. This led to CUDA Parallel Computing Platform for GPU.

High Performance Computing:

The use of super computers and parallel processing techniques to solve complex computational problems is known as HPC. HPC focuses on developing parallel processing algorithms and systems by including both administration and parallel computing techniques.

HPC is usually used to solve advanced problems and performing research activities through computer modelling, simulation and analysis. High performance systems have the ability to deliver sustained high performance through the concurrent use of computing resources.

Parallel Computing

Parallel computing forms the base of High performance computing. Programs for HPC systems must be split up into many smaller "programs" called threads, corresponding to each core. The cores must be able communicate with each other efficiently, to piece the larger program together, and the system as a whole must be organized well.

Programs on HPC systems produce a huge quantity of information, which is very tedious for standard file systems and storage hardware process. HPC file systems should be in a position to morph to contain and instantly transfer giant amounts of information. In addition to information in use, researchers often keep previous data for comparison or as a starting point for future projects. Older data are stored in the archival storage systems.

The Core of Artificial Intelligence

"Artificial Intelligence is an era of computing science that emphasizes the creation of intelligent machines that work and react like humans". A typical AI perceives its environment and takes actions that maximize its chance of successfully achieving its goals. An AI's intended goal function can be simple ("1 if the AI wins a game of Go, 0

otherwise") or complex ("Do actions mathematically similar to the actions that got you rewards in the past"). Goals **may be expressly outlined**, or **may be elicited**. If the AI is programmed for "reinforcement learning", goals can be implicitly induced by rewarding some types of behaviour and punishing others. Alternatively, an evolutionary system can induce goals by using a "fitness function" to change and clone the high-scoring AI systems; this is just same as how the humans evolved. There are a certain desirable traits which were carried forward in the next generation and the undesirable were lost or improved. Nearest-neighbour AI systems are not generally given goals, except to the degree that goals are somehow implicit in their training data. These can still be benchmarked if the non-goal system is framed as a system whose "goal" is to successfully accomplish its narrow classification task.

Artificial Intelligence is revolutionizing every industry. It is already revolutionizing the field of supercomputing, health care, financial services, big data analytics and gaming.

Deep Learning

Deep learning is a part of the wider stream of machine learning. Deep learning has its roots embedded in machine learning methods based on learning data representation instead of task specific algorithms.

Deep learning is a class of machine learning algorithms that: [7]

- Use a cascade of multiple layers of nonlinear processing units for feature extraction and transformation. Each sequential layer uses the output from the previous layer as input.
- Learn in supervised (e.g., classification) and/or unsupervised (e.g., pattern analysis) manners.
- Learn multiple levels of representations that correspond to different levels of abstraction; the levels form a hierarchy of concepts.

GPU-Accelerated Deep Learning

NVidia is universally recognized, may it be in academics or research, for its high performing processors for training deep neural networks. They are used due to their speed and efficiency, which is better than the most of the chips available in the market. Neural networks are highly parallel in nature, as they are created froma large number of similar neurons. This is a boon for GPU, as GPU excels over any CPU for parallel processing.

Neural networks are based on heavy matrix maths operations, and complex multi-layered networks needan extremenumber of floating-point performance and bandwidth for both speed and efficiency. GPUs contain thousands of processes cores specially designed and optimized for matrix math operations, providing tens to hundreds of TFLOPS of performance. These features make GPUs the best choice for computing deep neural network based applications and machine learning applications.

Volta is designed in such a way that it runs deep learning workloads optimally, achieving an amazing increase in performance within the equal power budget than the previous generation architecture design.

Hardware Optimized for Artificial Intelligence

AI computer software code has been continuously received the lion’s share of attention, but as the computational resources needed to process this software soar exponentially, a new generation of AI chips is coming into being.

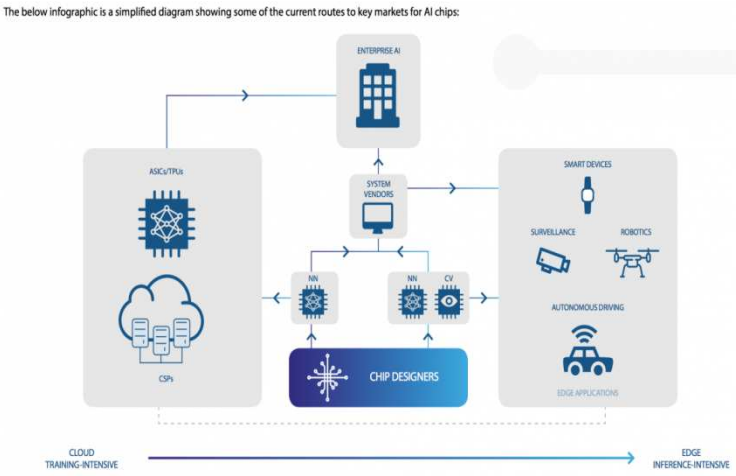


Figure 2: Key Markets for AI Chips

The AI computation load is different from the calculations most of our current computers are designed to do. AI implies prediction, inference, and intuition. But the most creative machine learning algorithms are limited by the machines that can't utilise their power. Hence, if we're to make great strides in AI, the hardware must support the ever-changing needs. Starting with GPUs, and then evolving to analog devices, and then to fault tolerant quantum computers. Deep learning has progressed 2.5 times per year since 2009, when the purpose of GPUs went from video game graphics accelerators to deep learning model trainers.[5]

Balance ratios are the key to understanding the abundance of AI hardware solutions that are being developed or are soon to become available. Future proofing procurements to support run-anywhere solutions—rather than hardware specific solutions—is the key

The fundamental idea behind balance ratios is to keep the working part and modify and improve the hardware properties whenever possible. Currently hard wares are limited by memory bandwidth. Thus, the flop/s per memory bandwidth balance ratio is most vital for deep learning training. So long as there is a sufficient arithmetic capability to support the memory (and cache bandwidth), the hardware will deliver the best performance possible by itself. Similarly, the flop/s per network performance is vital for scaling data preprocessing and training runs for deep learning applications. Storage IOP/s (IO Operations per Second) is vital for performing irregular accesses in storage when working with unstructured data.

Compute

The future of AI hardware includes CPUs, accelerators/ purpose-built hardware, FPGAs and future neuromorphic chips. Intel is also focused on developing the AI hardware, their efforts include:

- CPUs: including the Intel Xeon Scalable processor family for evolving AI workloads, as well as Intel Xeon Phi processors.
- Special purpose-built silicon for AI training such as the Intel Neural Network Processor family.
- Intel FPGAs, which can serve as programmable accelerators for inference.
- Neuromorphic chips such as Loihi neuromorphic chips.

- Intel 17-Qubit Superconducting Chip, Intel's next step in quantum computing.

Although an aborning technology, it is worth mentioning that machine learning can be mapped to the quantum computing, and if this happens in the near future, then such a hybrid machine can change and revolutionize many fields.

Intel Neural Network Processor

Intel has bought AI hardware Nervana Systems. As previously discussed, Intel Nervana Graph will act as a hardwarespecific software that will provide the best hardware optimization.

On October 17, 2017, Intel announced it will ship the industry's first silicon for neural network processing, the Intel Nervana Neural Network Processor (NNP), before the end of this year. Intel's CEO Brian Krzanich stated at the WSJDlive global technology conference, "We have multiple generations of Intel Nervana NNP products in the pipeline that will deliver higher performance and enable new levels of scalability for AI models. This puts us on track to exceed the goal we set last year of achieving 100 times greater AI performance by 2020."

Neuromorphic Chips

Intel has developed a self-learning neuromorphic chip—codenamed Loihi—it is inspired from the biological neurons in the human brain and the way they respond to the environment. These self-learning chips uses asynchronous spiking instead of the activation functions used in current machine and deep learning neural networks.

Figure 3: Intel's Loihi Neuromorphic Chip

Storage

As the case with main memory, storage performance is decidedby throughput and latency. Solid State storage (coupled with distributed file systems such as Lustre) is one of the biggest developments in unstructured data analysis.

Instead of being able to perform a few hundred IOP/s, an SSD device can perform over half a million random IO operations per second. This makes managing big data feasible. [6]

NVidia Tesla V100

The NVidia Tesla V100 accelerator is the most performing parallel computing processor in the world today. The V100has significant hardware innovations that provide extreme speedups for machine learning, deep learning algorithms and frameworks, in addition to providing extreme computational power for HPC Systems and applications.

In an attempt to overcome the hardware limitations, the current trend is to use CPU and GPU in a conjunction, or GPU Computing. GPU computing is a relatively a new field, where NVidia is leading the charge with its NVidia Tesla V100 and NVidia Volta V100.

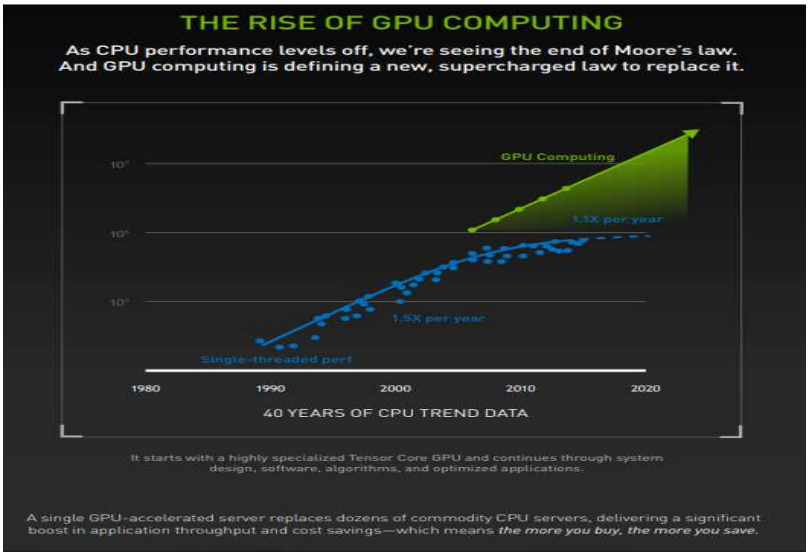


Figure 4: Rise of GPU Computing

Similar to the previous generation of GPUs following Pascal architecture, the V100 consists of multiple GPU Processing Clusters, Texture processing Clusters, Streaming Multiprocessors, and memory controllers. A full V100 consists of:

- Six GPCs

  Each GPC has:

  - Seven TPCs (each including two SMs)
  - 14 SMs
- 84 Volta SMs

  Each SM has:

  - 64 FP32 cores
  - 64 INT32 cores
  - 32 FP64 cores
  - 8 Tensor Cores
  - Four texture units
- Eight 512-bit memory controllers (4096 bits total)

With 84 SMs, a full GV100 GPU has a total of 5376 FP32 cores, 5376 INT32 cores, 2688 FP64 cores, 672 Tensor Cores, and 336 texture units. Each HBM2 DRAM stacks is controlled by a pair of memory controllers. The full GV100 GPU includes a complete of 6144 KB of L2 cache. Figure 4 shows a full GV100 GPU with 84 SMs (different products can use totally different configurations of GV100). The Tesla V100 accelerator uses 80 SMs.

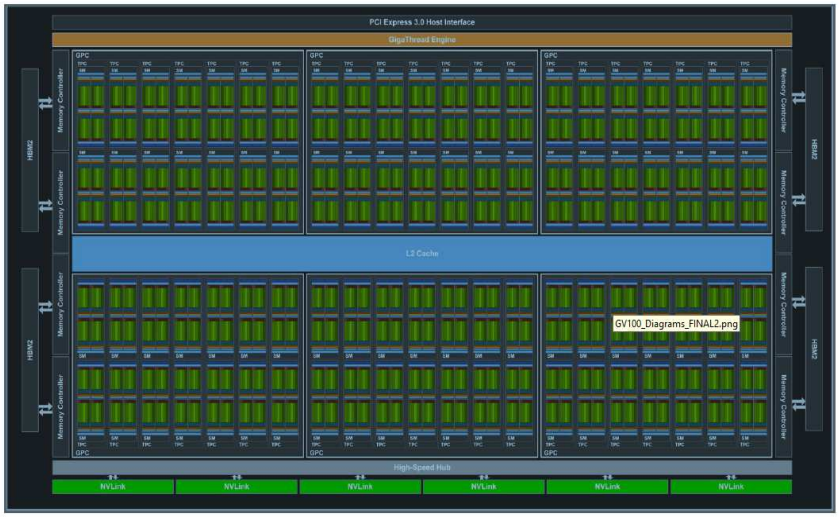


Figure 5: Tesla V100 Full GPU with 84 SM Units

Volta Streaming Multiprocessor

NVidia Volta features a new Architecture for Streaming Multiprocessor (SM)which delivers a huge improvement in performance, efficient in energy utilization and easily programmable.

The Highlight Feature Includes

- Specially designed mixed-precision Tensor Core, for deep learning matrix arithmetic. Efficient in energy usage by 50% in general computing workload.
- Enhanced high performance by L1 data cache.
- The limitations present in SMIT and SMID are removed by a new SMIT thread model

Compared with Pascal GP100, the GV100 SM has 64 FP32 cores and 32 FP64 cores per SM. but, the GV100 SM uses a brand new partitioning methodology to boostStreaming performance and overall efficiency. The V100 SM is partitioned into four blocks, each with 16 FP32 Cores, 8 FP64 Cores, 16 INT32 Cores, two of which arenew, mixed-precision Tensor Cores for deep learning, one warp scheduler, one dispatch unit, matrix arithmetic, a new L0 instruction cache, and a 64 KB Register File.

While the number of registers in V100 is same as in Pascal GP100 SM, the entire V100 GPU has more SMs, and thus much more registers overall. In all, V100 supports more threads, warps, and thread blocks as compared to previous GPU generations.

The union of shared memory and L1 resources enables allows a bump in shared memory capacity to 96 KB per Volta SM.

What is Streaming Multiprocessor?

The streaming multiprocessors (SMs) are a part of the GPU that runs our CUDA kernels. Each SM contains the following.

- Thousands of registers that can be partitioned among threads of execution

- Several caches:
- Fast data interchange using a shared memory
- Fast broadcasts of reading using a cache memory
- Collecting the bandwidth from the texture memory using the Texture cache
- Local and Global memory latency is reduced using L1 cache.

*Qui*ckly switch contexts between threads and issue instructions to warps that are ready to execute, using a wrap scheduler.

- Execution cores for integer and floating-point operations:
- Integer and single-precision floating point operations
- Special Function Units (SFUs) for single-precision floating-point transcendental functions
- Double-precision floating point

As there are several registers the hardware can context switch between threadsefficiently, which maximize the throughput of the hardware. The GPU is meant to possess to have enough state to cover up both the execution latency, memory latency of hundreds of clock cycles which it may take for information from the device memory to arrive once the read instruction is executed.

SMs are all-purpose processors, however, they are designed very differently when compared to the execution cores in CPUs: They target a very lower clock rate; they support instruction-level parallelism, however, they do not predict or speculate execution; and that they have less cache, if they have any cache at all. For appropriate workloads, the sheer computing grunt in a GPU makes up for these shortcomings.

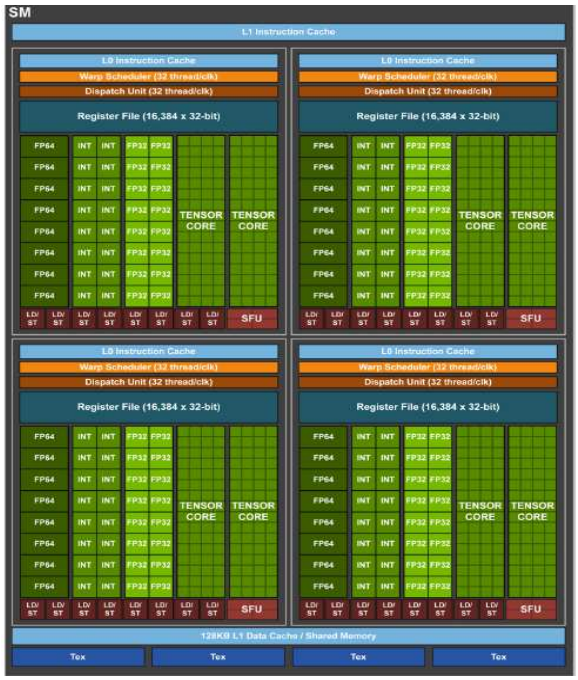


Figure 6: Volta V100 Streaming Multiprocessor (SM)

NVidiaNVLink

Before we dive into the working of NVidiaNVLink, we will first discuss, what NVidiaNVLink is?

NVidiaNVLink, a.k.a NVidiaNVLink fabric is an interconnect. NVidia introduced NVLink to connect multiple GPUs and GPUs to CPUs. It has about 10x the bandwidth to that of a PCIe interconnect. This high available bandwidth, in turn boosts the computing capacity.

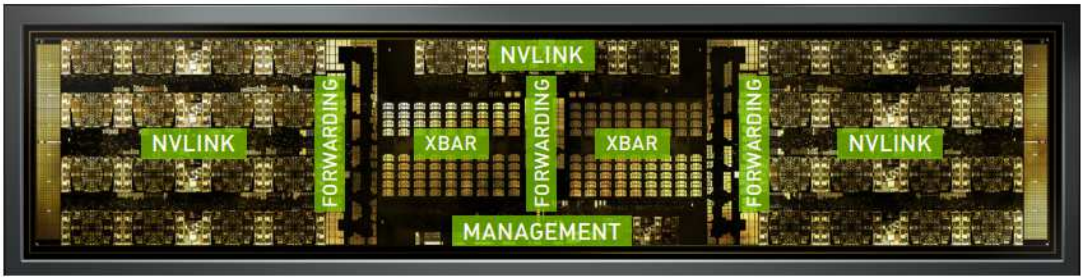


Figure 7: NVidiaNVLink Die Shot

Supercomputer systems with multi GPUs and multi CPUs are becoming common in the variety of industries. Industries based on Deep learning, Artificial Intelligence, molecular dynamics requires heavy processing. These applications rely on parallelism. Parallelism is a concept where multiple tasks are performed simultaneously. The systems working on these applications include 4-GPU and 8-GPU system configuration using PCI system interconnects to solve complex and lengthy problems. These systems initially were connected using PCIe interconnects. But PCIe interconnect starts to bottle neck the multi GPU or GPU to CPU systems, due to its limited availability of bandwidth.

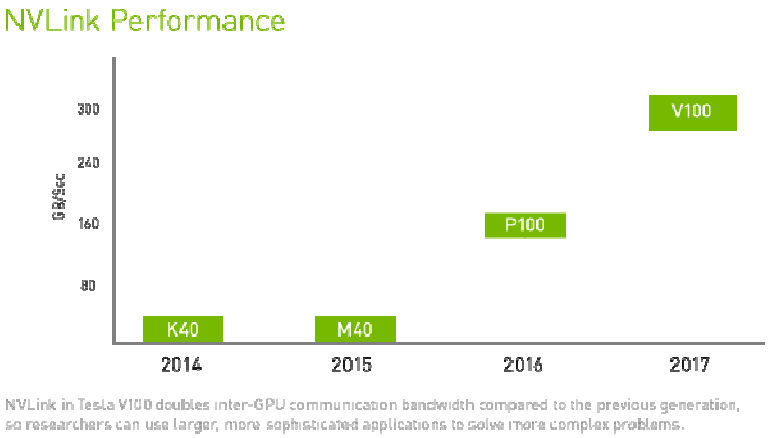


Figure 8: NVLink Performance in V100 GPU

NVidiaNVLink technology solves the issue of bottlenecking PCIe interconnects by giving higher bandwidth, higher number of linking, and an improved scalability for multi-GPU and multi GPU/CPU system configurations. NVidiaNVLink was a solution that enabled 8 GPUs to be connected together in a point to point network called a hybrid cube mesh.

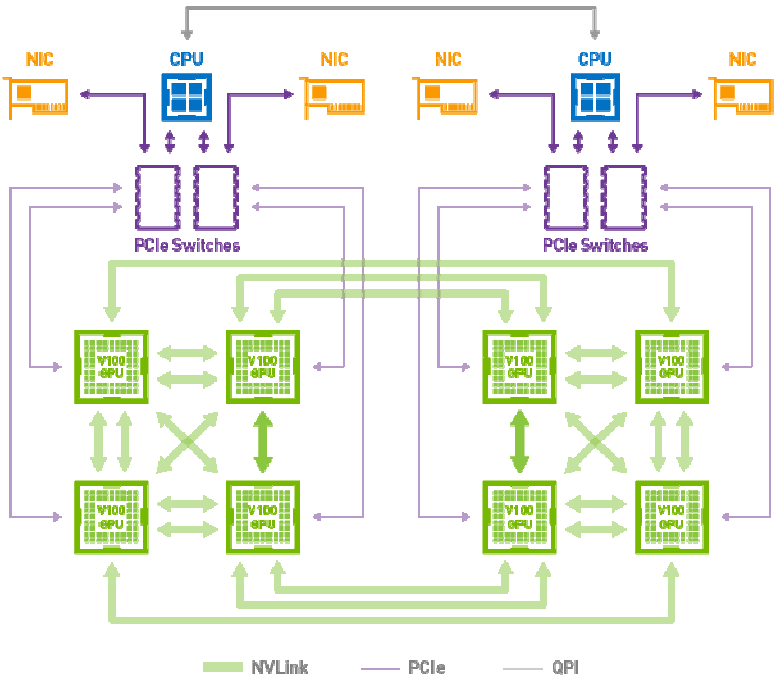


Figure 9: Hybrid Cube Mesh

NVLink is designed for high bandwidth GPU/GPU and GPU/COU interfacing. It stores frameworks that let the programmers directlyread and write into the local graphics memory (GMEM) and peer GMEM or the CPU's memory (SYSMEM) using the NVLink. Everything is stored in the shared address space. NVLink lets the user groups GPUs or GPUs/CPUs, and use them as a single large unit.

NVLink has a two-way(bidirectional)interface. Every NVLink comprises of eight differential pairs in each direction, for a total of 32 wires. Data transmission is sent up to 20 Gbits per second, giving it a bandwidth of40 GBps for one NVLink. V100 supports up to6 NVLinks, giving it a bidirectional bandwidth of 300gbps. A clock, embedded in system is used. Lane reversal and polarity inversion are supported which eases routing. The bit error specified is better than 1 in 1e12. A 25bit (CRC) cyclic redundancy clock is used to detect the error.

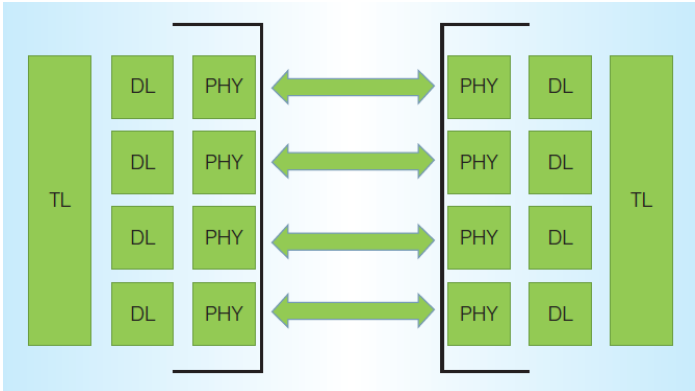


Figure 10: NVLink Physicals Showing the four NVLinks between Two GPUs and the Layering Protocol

The size of the NVLink packets ranges from a single 128 bit upto 18 128 bits, to support 256 byte transfer, An NVLink transaction is made up of atleast a request and a response and optionally an address extension (AE), a byte enables and between 0 to 16 data payloads.

A request header is made up of the 25bit CRC; an 83 bit transaction layer field with request type, address, flow control credits and tag identifier, and a 20 bit data link (DL), which also comprises of packet length information. Address extension (AE) consists of information that should be relatively static form request to request.

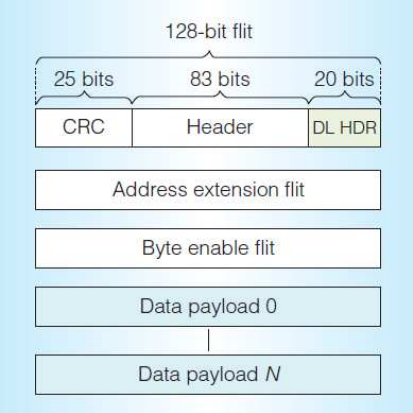


Figure 11: Write Transaction Packet Format with Header, Address Extension (AE), Byte Enable(BE) and Data. CRC(Cyclic Redundancy Check)

CRC

CRC enables the detection of 5 random bits from the 25 sequential transmitted bits. Packets are stored in a buffer. Every packet has its associated ID with it. When a packet arrives with good CRC, a positive acknowledgement is sent to the source. If a positive acknowledgement is not received in the stipulated amount of time, a replay sequence is initiated, and a packet in error and all subsequent packets are retransmitted.

Packet length is variable, and the information regarding its length is conveyed through its header, as DL Header. As the header consists of the packet length information and the protocols contains no framing symbols, the CRC on the header must be checked before to transmitting the packet.

Efficiency

A posted, write of the maximum size packet without Address Enable (AE) flit takes 17 flits. One for the header, and the rest 16 for the 16 data payload cycles. A link efficiency of 94.1% can be sustained.

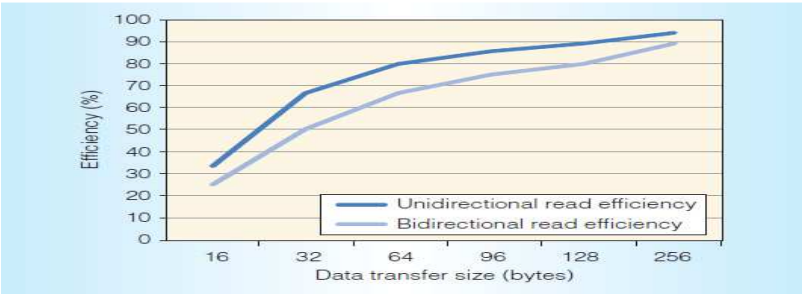


Figure 12: Unidirectional and Bidirectional Read Efficiency Showing the Roll Off in Efficiency as Payload Size Decreases

Reducing to a 128 byte packet drops the efficiency to 88.9%. Posted write takes up the bandwidth in one direction only. There is no response required.The upstream indication that a transaction has occurred can be carried by unrelated upstream packets and doesnot incur any additional overhead.

Tensor Cores

What is Tensor Cores

Tensor core or a Tensor Processing unit is an Application Specific Integrated Circuit (ASIC). It was developed by Google, used especially for AI acceleration and Neural Network Machine Learning.

The chip was specifically designed to be used by Google's TensorFlow Framework. TensorFlow frame work is a mathematics library, which finds its uses in machine learning applications, such as neural networks.

When compared to the Graphics Processing Unit, Tensor cores are used for high volume but low precision computations, with higher Input Outputs per Seconds (IOPS).

Functioning of a Tensor Core

To restate the above mentioned description, a Tensor Core unit is a new type of processing core that performs highly specialized matrix mathematics, which is desirable for the applications of Deep Learning and HPCs. The sole purpose of a tensor core is to perform a fused multiply, add, where two 4x4 FP16 (Half-Precision floating point) matrices are multiplied and then the result is added into a 4x4 FP16 or 4x4 FP32 Matrix. These results are referred as mixed precision maths, because the results are in full precision, whereas the inputs were half precision.

Tesla V100's Tensor core units can deliver upto 125 Tensor TFLOPS for training and inference operations. V100 contains 640 Tensor cores, 8 per SM(Streaming Multiprocessor). Tensor cores and their paths are custom designed to minimize the area ant the power cost. To maximize power savings, clock gating is used extensively.

Each tensor core provides a 4x4x4 matrix processing, which performs the operation D=A*B+C, where the variables A, B, C and D are nothing but 4x4 Matrices. The input matrix A and matrix B are FP16 matrices, which are multiplied and the C and D are accumulation matrices, which could be FP16 or a FP32.

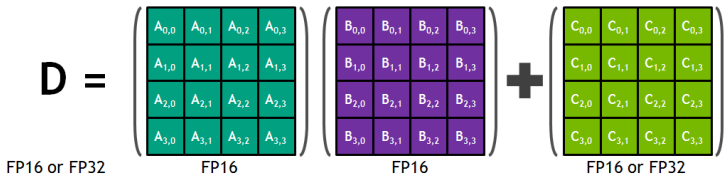


Figure 13: Tensor Core 4x4x4 Matrix Multiply and Accumulate

Each Tensor core performs a 64 floating point FMA mixed-precision operations per clock. The 8 tensor cores present in V100's SM performs a total of 1024 floating point operations in one clock cycle. This drastically increases thethroughput, for Deep learning applications per SM. It is about 8x when compared to previous generations of Pascal architecture.

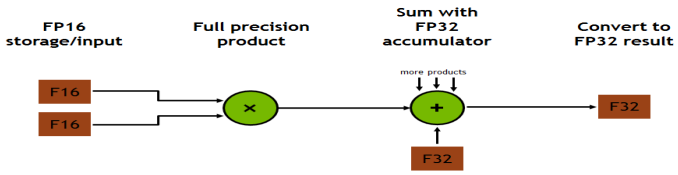


Figure 14: Tensor Core Operation

During the execution of the program, multiple Tensor cores are used inparallel by a full wrap of execution. The thread provides a larger 16x16x16 matrix operation to be performed and processed by the tensor cores. This can be understood in details bystudying wrap level operations in the CUDA C++ WMMA API. This IDE provides a specialized interface to load the matrices, multiply and accumulate them.

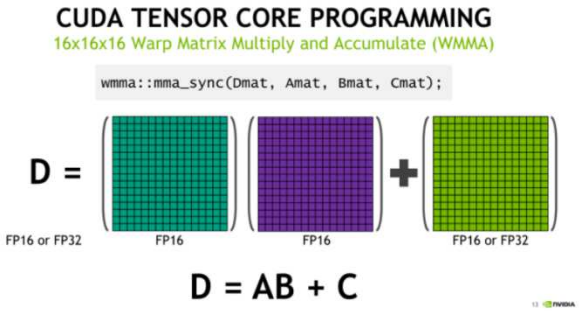


Figure 15: Tensor Core 16x16x16 Matrix Multiplication

Low level microbenching by a team at Citadel LCC studied these Multiplication and accumulation processes. Their study revealed a number of Volta microarchitecture details, including the tensor core operations and the fragments involved, both the location in the register and the identity compared to the input matrices. The team at Citadel observed the working pattern while calculating the input matrix, while all 32 threads in action. The tensor cores operate on 4x4 submatrice to calculate the larger 16x16 matrix. It makes the use of Volta's new scheduling model and cooperative groups.

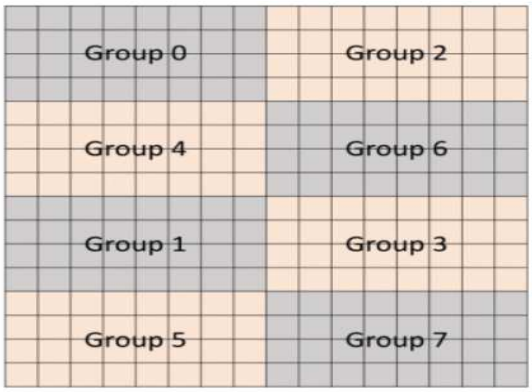


Figure 16: Mapping between Positions in Matrix c and Thread Group Indices

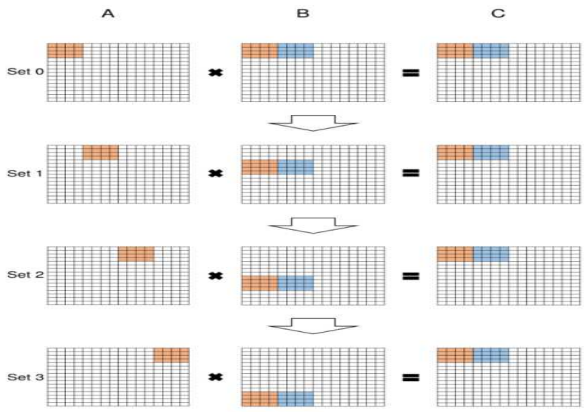


Figure 17: Four Sets of HMMA Instructions, Complete 4x8results in Matrix C within Thread Group 0. Different Sets us different Element in A&B. Instruction Executes in set 0 First, then 1,2 and 3

As the wrap texts are spread out into 8 threads groups of 8 threadseach, each group computes an 8x4 chunk in a serial order. It goes through above shown processes of 4 sets. So, each group deals with 1/8 of the resultant matrix.

Execution Model of Streaming Multiprocessor Architecture

Now that the basics are covered, let us now briefly study the execution of a Streaming Multiprocessor.

In any GPU, there are two main components:

- Global Memory.
- Streaming Multiprocessor.
- Global Memory:

Global memory is analogous to RAM in a CPU server. The Global memory is accessible by both the CPU and the GPU.

Currently, for V100 the size of global memory is 16 and 32Gb, depending on the versions of the GPU.

Streaming Multiprocessor

Streaming Multiprocessor is the component which performs the actual calculations and computations.

Every SM has its own Control Units, registers, execution pipelines and cache memories.

Execution Model

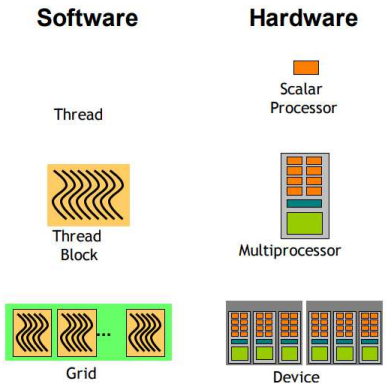


Figure 18: Execution Model of SM

To understand the execution of any Streaming Multiprocessor, Figure 6.1 gives an over view of its implementation. Threads are executed by the scalar processors. These thread blocks are then executed on the multiprocessor. It is important to note that the thread blocks do not migrate. Several concurrent threadblocks can reside on one multiprocessor. The numbers of threads depend on the resource availability of that particular multiprocessor. A kernel is then launched as a grid of thread blocks.

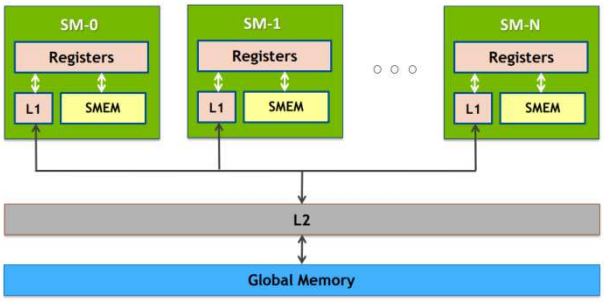


Figure 19: GPU Memory Hierarchy Review

Figure 6.2 gives a deeper insight into how the memory is accessed by the Streaming Multiprocessors. The memory elements in GPU are extremely fast, but the size is very small, of about 10s of Kb.

L1 is used as a cache memory;sometimes it is used as shared memory. The usage depends on the programmer.

L2 is a unified cache memory. It is fast and enables coherent data sharing across all the cores in the GPU. It is large in size as compared to the L1, usually in the range of 100s of kb.

Speed Vs. Throughput

When it comes to the Execution speed of a task and the Throughput of the task, the CPU architecture and the GPU architecture differ from each other in following ways:

- With each threading, the CPU architecture must minimize the latency.
- GPU architecture hideslatency with computation from another thread wraps.

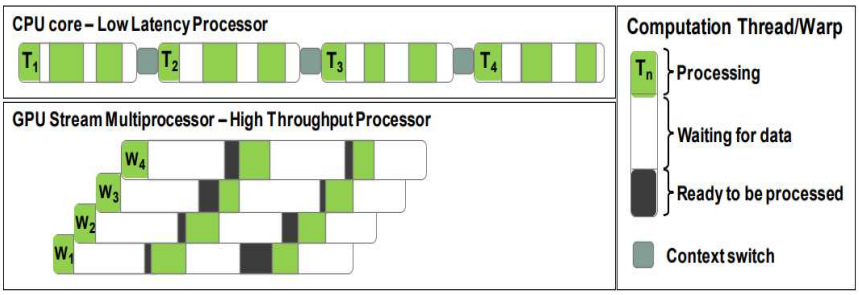
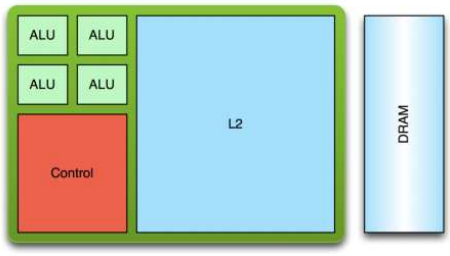
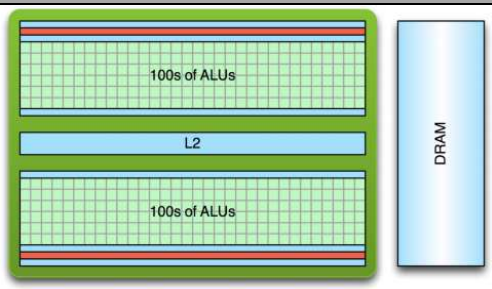


Figure 20: CPU Vs GPU Computation thread

Table 1: CPU vs. GPU Architecture

CPU	GPU
	
Optimized for low latency access to cache data sets.	Optimized for data parallel, throughput computation.
Control logic for out-of-order and	Tolerant of memory latency.

speculativeexecutions	More transistors are dedicated to computation.
10s of threads	10000s of threads

System Configuration for Deep Learning and HPC

NVidia NVLinks can be grouped to increase bandwidth between two ends or used solely to maximize the number of endpoints. This provides flexibility in what types of systems can be could be created using these links. A very simple configuration would be two GPUs communicating over a gang of four NVidia NVLinks.

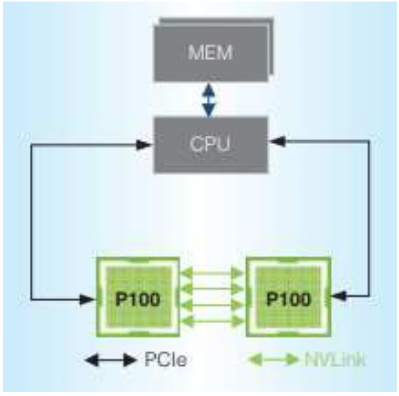


Figure 21: A Four-NVLink Group used to Connect GPUs

In the above example, the NVidia NVLink bidirectional bandwidth is 160 GBps—a significant increase when compared to the 32 GBps peak bidirectional bandwidth given by an x16 Gen3 PCIe peer-to-peer link through a switch. This configuration used for NVidia's DGX-1 Deep Learning Supercomputer is called as a Hybrid Cubed Mesh. Imagine a cube with a V100 SXM2 module at each corner. The four GPUs on the top face are completely connected with one link to each the other. The remaining link for each GPU is used to connect the top and bottom faces of the cube together, as shown in the next Figure 7.2.

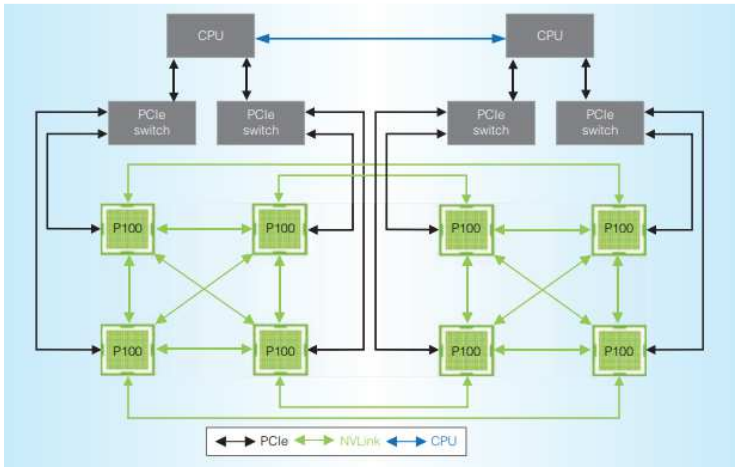


Figure 22: Hybrid Cubed Mesh. Each GPU has an NVLink is connected to four other GPUs

The CPUs in this configuration are not NVidiaNVLink enabled. Each GPU is connected to a CPU using a x16 PCIe Gen3 connection through a PCIe switch.A switch is shared between the two GPUs.Forming a standalone SMP (Symmetric Multiprocessor), are the two CPUs.NVLink can also be used for GPU to CPU connections, allowing for high-bandwidth direct load and store access to the large system memory.

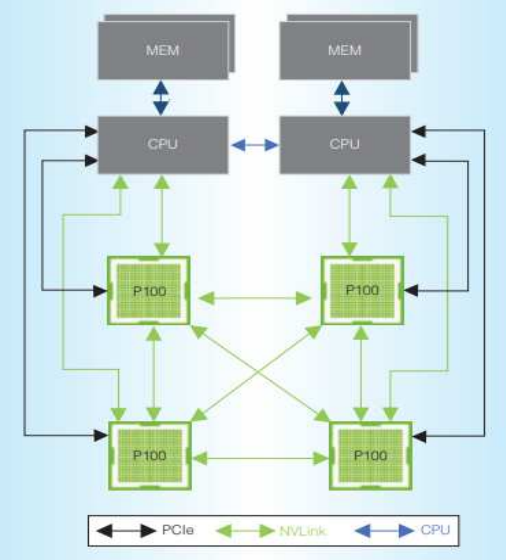


Figure 23: GPU to CPU connections using NVLink

Figure 23 shows a two-socket SMP configuration in which each of the CPUs supported by NVidia NVLinks. Each GPU has one link to each of its peers and a single link to one of the two CPUs. For this configuration, each GPU has up to 20 GBps of read and write bandwidth to and from system memory. It also has 20 GBps of read/ write bandwidth available to each of its peer.

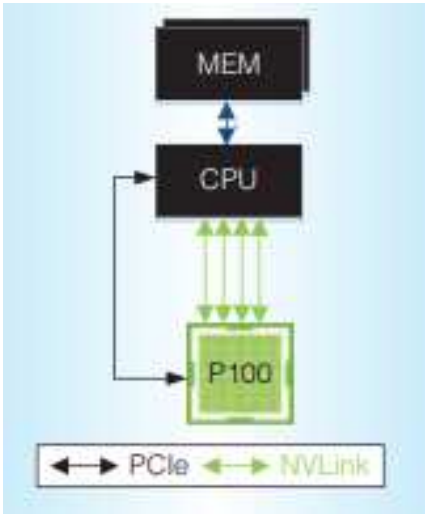


Figure 24: Four NVLinks ganged from the GPU and CPU

Figure 24 shows a configuration with four NVLinks grouped between a single GPU and a CPU. In this system, the GPU has up to 80 GBps of read and write bandwidth to system memory

V100 is NVidia'sbest performing GPU till today. The power efficiency, bandwidth, and density of the in package HBM accompanied along with unified memory, FP16 operations, and NVLink enablesNVidia to design a machine with unmatched deep learning abilities.

REFERENCES

1. *https://en.wikipedia.org/wiki/Moore%27s_law.*
2. *https://upload.wikimedia.org/wikipedia/commons/thumb/9/9d/Moore%27s_Law_Transistor_Count_1971-2016.png/800px-Moore%27s_Law_Transistor_Count_1971-2016.png.*
3. *https://ourworldindata.org/technological-progress.*
4. *Kisaco AI Chip Infographic, Source: Kisaco Research Conference Agenda.*
5. *https://www.ibm.com/blogs/research/2017/12/future-hardware-ai/.*
6. *https://insidehpc.com/2018/01/ai-hardware-support-software/.*
7. *CUDA Handbook: A Comprehensive Guide to GPU Programming.*
8. *Deng, L.; Yu, D. (2014). "Deep Learning: Methods and Applications" (PDF). Foundations and Trends in Signal Processing.* ***7*** *(3–4): 1–199. doi:10.1561/2000000039.*
9. *Denis Foley, John Danskin, Ultra-Performance Pascal GPU and NVLink Interconnect. IEEE Micro (Volume 32, Issue 2, March- April 2017) Electronic ISSN: 1937-4143.*
10. *https://www.NVidia.com/en-us/data-center/NVLink/.*
11. *http://www.icl.utk.edu/~luszczek/teaching/courses/fall2016/cosc462/pdf/GPU_Fundamentals.pdf.*
12. *https://images.NVidia.com/content/volta-architecture/pdf/volta-architecture-whitepaper.pdf.*
13. *https://images.NVidia.com/content/pdf/v100-application-performance-guide.pdf.*
14. *Ultra performance Pascal GPU and NVLink Interconnect, IEEE Micro, Volume 37, Issue2, March- April 2017. (ISSN 0272-1732) Pages 7-17. Publication date: 10 May 2017.*
15. *https://www.NVidia.com/en-us/data-center/NVLink/.*