Accelerator Architectures—A Ten-Year Retrospective

Wen-mei Hwu and Sanjay Patel
University of Illinois at Urbana-Champaign

This article is a ten-year retrospective of the rise of the accelerators since the authors coedited a 2008 IEEE MICRO special issue on Accelerator Architectures. It identifies the most prominent applications using the accelerators to date: high-performance computing, crypto currencies, and machine learning. For the two most popular types of accelerators, GPUs and FPGAs, the article gives a concise overview of the important trends in their compute throughput, memory bandwidth, and system interconnect. The article also articulates the importance of education for growing the adoption of accelerators. It concludes by identifying emerging types of accelerators and making a few predictions for the coming decade.

It has been an exciting ten years since we coedited an IEEE MICRO special issue titled "Accelerator Architectures" in 2008, shortly after NVIDIA launched the CUDA architecture for GPU Computing. In the Guest Editor's Introduction, we stated:

> *As with any emerging area, the technical delimitations of the field are not well established. The challenges and problems associated with acceleration are still formative. The commercial potential is not generally accepted. There is still the question, "If we build it,* ***who will come****?"*

We define accelerator as a separate architectural substructure (on the same chip, on a different die on the same package, on a different chip on the same board) that is architected using a different set of objectives than the base processor, where these objectives are derived from the needs of a special class of applications. Through this manner of design, the accelerator is tuned to provide higher performance, lower cost, lower power, less development effort, or combinations of the above than with the general-purpose base hardware.

Accelerators are generally attached to the base processor via a system-level interconnect, such as PCI Express (PCIe) for accelerators that are discrete chips. At the time of the 2008 special issue, high-end

Published by the IEEE Computer Society
0272-1732/18/$33.00 ©2018 IEEE

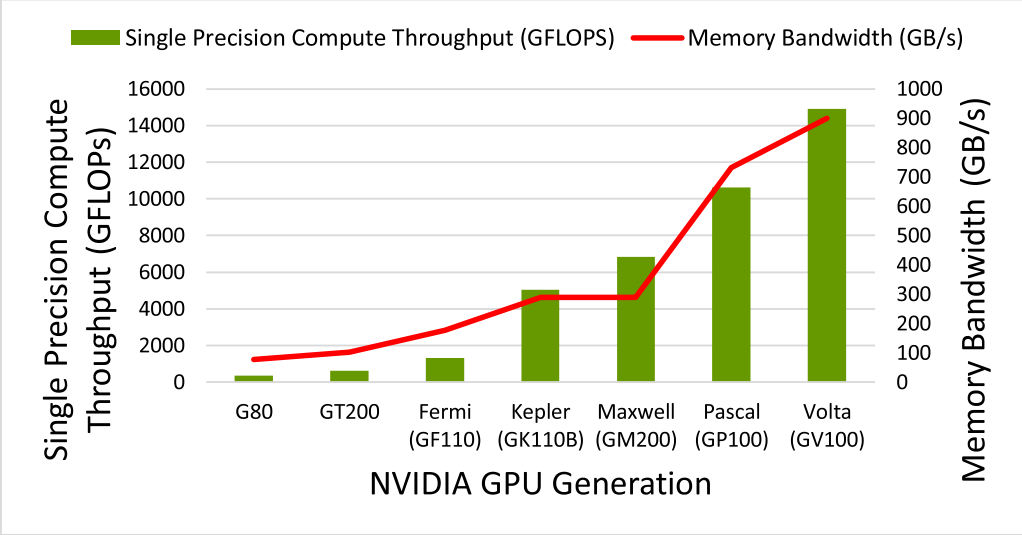

Figure 1. The progression of single-precision compute throughput and memory bandwidth of NVIDIA GPU generations since 2007.

accelerators such as GPUs were attached to the base processor via PCIe Gen-2 ×16 interconnect, which provided up to 4 GB/s of data transfer bandwidth in each direction. The high-end CUDA-enabled GPU at the time was GT280, which provided about 1 TFLOPS of single precision execution throughput and 150 GB/s memory bandwidth.

WHO CAME FOR THE GPUs?

Since 2008, GPUs have gained tremendous adoption in the computing market. According to NVIDIA, the number of CUDA-enabled GPUs sold in the market surpassed one billion in 2016. The most visible application areas for GPUs have been machine learning, cryptocurrency, and high-performance computing (HPC).

In HPC, GPUs have become popular in large computing clusters due to their high compute throughput, compute throughput per Watt ratio, memory bandwidth, and memory bandwidth per Watt ratio. Figure 1 shows the growth of peak computing throughput and memory bandwidth of generations of CUDA-enabled GPUs. As of 2018, the Volta GPUs provide about 15.7 TFLOPS single precision computing throughput and 900 GB/s memory bandwidth with a power rating at about 250 W, resulting in about 63 GFLOPS/W for computing throughput and 3.6 GB/s/W for memory bandwidth. (The double precision numbers are approximately half of the single precision numbers.) The throughput per watt and bandwidth per watt numbers make the GPUs very attractive for large-scale clusters. As a result, 98 of the Top 500 [1] and 98 of the Green 500 [2] supercomputers contain GPUs. In fact, the trend is to have multiple GPUs per cluster node to boost these ratios for the nodes. For example, each node in the recent Summit machine at the Oak Ridge National Lab contain four Volta GPUs. The most visible HPC applications that take advantage of GPUs include those that perform quantum chromo-dynamics, molecular dynamics, computational fluid dynamics, computational financing, and seismic analysis.

In cryptocurrency, GPUs have been the preferred vehicle for mining the various forms of cryptocurrencies, such as bitcoins.[3] Due to the computationally intensive nature of cryptocurrency mining, profit from mining is directly limited by the compute throughput per watt ratios of the computing devices being used. It is of little surprise that GPUs are the preferred computing devices. Since 2013 the Bitcoin market has grown from $1.6B to $109B.[4] An estimated 3 million GPUs have been sold into the Bitcoin market just in 2017 [4]! Note that Bitcoin is just one of the cryptocurrencies.

Since 2012, the industry has experienced an explosive growth in artificial intelligence and deep neural network-based machine learning applications. In 2012, Prof. Geoff Hinton's team from the University of Toronto won the ImageNet competition by using CUDA GPUs to train a neural network using 1.2 M images. This approach, commonly referred to as deep learning (DL) today, outperformed the best traditional computer vision approaches by more than 8% points, broke the 80% accuracy barrier, and sent a shock wave through the computer vision community.

By 2015, virtually all entries to the ImageNet competition use the DL approach. Since then, the use of DL has energized the development of self-driving cars. DL has also been spreading through many fields of natural data processing—speech, text, music, language translation, interactive dialogue systems, medical treatments, etc. At the center of this revolution is the need for training neural networks with a vast amount of data. GPUs have become the technology of choice. This has stimulated the very fast advancement of the training speed of CUDA GPUs. For example, for GoogleNet Training, the speed has improved by more than 80× form Kepler (K80) to Volta (V100) from 2015 to 2018. During the same period, the NVIDIA Stock price has also increased by more than 14×!

STATE OF GPU COMPUTING IN 2018

As we have shown in Figure 1, the computation throughput of GPUs has been on a climb through the expected path of technological scaling, microarchitecture innovations, and circuit innovations. The Volta GPUs are architected with peak computation throughputs of 7.4 TFLOPS for double precision (e.g., NVIDIA Quadro GV100), 13.8 TFLOPS for single precision (e.g., NVIDIA Titan V), and 27.6 TFLOPS for half precision (NVIDIA Titan V). If we consider the TensorCores, the Titan V achieves 125 TFLOPS for half precision. On the other hand, the growth in memory bandwidth has taken a bumpier trajectory.

Memory Bandwidth

In many application domains, high memory bandwidth is an important factor in GPU adoption. It is therefore very important that GPUs continue to scale their memory bandwidth in each generation. However, memory bandwidth scaling has been traditionally limited by the number of pins that can be supported by processor packaging. As a result, the memory bandwidth has not scaled as steeply as computation throughput for most generations of CUDA GPUs. For example, from Fermi to Kepler, the double precision compute throughput was increased from 515.2 GFLOPS (C2050) to 1175 GFLOPS (K20), or 2.3×, whereas the memory bandwidth was increased from 144 GB/s to 208 GB/s, or 1.4×. For Fermi GPUs to achieve peak compute throughput, an application has to reuse each operand at least 28.5 (515.2/(144/8)) times once it is fetched from the DRAM system. Otherwise, the application will be limited by memory bandwidth (memory bound) for the Fermi. For Kepler, each operand has to be reused 45.2 (1175/(208/8) times, making it even more difficult for applications to fully utilize the compute throughput of the GPU. Such limitation of the traditional packaging DRAM interface motivated the move into high bandwidth memory (HBM) for the following generations of GPUs.

Starting with Pascal GPUs, the packaging and DRAM interface switched to HBM, where three-dimensional (3-D) stacked DRAM chips are packaged on the same chip carrier as the GPU. This allows more wire connections between the GPU and the DRAM chips. As a result, there was a 3.65× jump of memory bandwidth from Kepler (208 GB/s) to Pascal (760 GB/s), as shown in Figure 1. This jump greatly alleviated the pressure of operand reuse for the Pascal generation. However, this is a one-time jump. Future generations will likely have only incremental improvement. For example, the Volta GPUs have 900 GB/s, 20% higher than Pascal.

System Interconnect

More recent GPUs use PCI-E Gen3 or Gen4, which supports 8-16 GB/s in each direction. The Pascal family of GPUs also supports NVLINK, a CPU-GPU and GPU-GPU interconnect that allows transfers of up to 40 GB/s per channel for NVLINK 2.0.[6] While ×86 processors have stayed with PCIe as their

interconnect to CUDA GPUs, the IBM Power 8 and Power 9 servers have adopted NVLINK 1.0 and 2.0 respectively as their interconnect to the GPUs. The significantly higher data exchange bandwidth between multiple GPUs and between CPU and GPU make these systems valuable for HPC applications and DL model training.

I/O and Network Architecture

The use of GPUs in large-scale HPC clusters has motivated the development of GPU Direct and RDMA support.[7] In GPU Direct, the network packets can go directly from the Network Interface Card (NiC) to the GPU memory and vice versa. This eliminates the need for the data to be copied from the NiC to the CPU memory and then to the GPU memory. The elimination of the intermediate data copy improves the network data throughput by about 2×. This, however, requires that NiC vendor support GPU Direct and RDMA.

WHO CAME FOR THE FGPAS?

One of the biggest trends of FPGA computing is that more and more FPGA solutions are becoming available in the cloud. For example, according to Microsoft,[9] there are more than one million servers with FPGA accelerators in the Azure Cloud data centers. These FPGA accelerators have been used to accelerate network host functions. Offloading the network stack functions to FPGAs allows these functions to be performed in a more energy efficient manner and allows more of the host computing to be available to applications and improve the cost-effectiveness of cloud server provisioning. For another example, an Amazon EC2 F1 instance includes up to 8 FPGAs in an instance. The hardware resources include Xilinx UltraScale+ FPGAs, each of which has 64 GB DDR4 memory and dedicated PCIe ×16 interconnect.

The FPGA market was valued at $5.83B in 2017 and is expected to reach $9.50B by 2023, at a CAGR of 8.5% between 2017 and 2023. To date, the use of FPGAs to accelerate other compute functions such as DL has been limited by the cost of the high-end FPGA, the complexity of their usage, and their

Table 1. Specs of example FPGAs from (a) Altera/Intel and (b) Xilinx.

	Stratix 10 GX	Arria 10 GX	Cyclone V GX
Logic Elements (K)	378 ~ 5,510	160 ~ 1,150	35.5 ~ 301
ALM Registers (K)	512 ~ 7,470	246 ~ 1,708	53 ~ 454
Memory Blocks	1,537 ~ 11,721 (M20K)	440 ~ 2,713 (M20K)	135 ~ 1,220 (M10K)
DSP Blocks	648 ~ 5,760	156 ~ 1,688	57 ~ 342
Peak GFLOPS	1000 ~ 9,200	140 ~ 1,519	N/A
Process	14nm	20nm	28nm

(a) Altera FPGAs.

	Virtex UltraScale+	Virtex UltraScale	Virtex 7	Spartan 6
Logic Cells (K)	862 ~ 2,852	783 ~ 5,541	326 ~ 1,954	3.84 ~ 147.4
Memory (Mb)	115.3 ~ 65,913	44.3 ~ 132.9	27.0 ~ 67.7	0.22 ~ 4.82
DSP Slices	2,280 ~ 12,288	600 ~ 2,880	1,120 ~ 3,600	8 ~ 180
Process	16nm	20nm	28nm	45nm

(b) Xilinx FPGAs

programming latency. However, recent works such as[8] show that FPGAs can be extremely competitive against embedded GPUs for inference in the edge computing domains.

STATE OF FPGA COMPUTING IN 2018

The latest FPGAs and SoCs provide a combination of high computation throughput, low latency, and good power efficiency. As more and more compute intensive and latency sensitive applications come into the picture, FPGA vendors have been pushing their products to higher and higher levels in terms of capabilities, system integration, application scenarios, FPGA design methodologies, etc.

Compute Throughput and Memory Bandwidth

FPGAs in 2018 have much higher computing capabilities compared to FPGAs back to five years ago. Table 1 shows the specs of representative Intel FPGAs and Xilinx FPGAs. One example of the latest FPGAs that have highest computing capabilities is the Intel Stratix 10 FPGA. Stratix 10 FPGA delivers 2× more performance gains over previous generation Stratix V FPGAs and saves up to 70% power. Stratix 10 FPGA is built on Intel 14 nm Tri-Gate process, featuring Intel Hyperflex FPGA architecture. Stratix 10 FPGA is the current largest monolithic FPGA device with 5.5 million logic elements. It delivers up to 10 TFPLOS of single-precision floating-point DSP performance. In terms of memory bandwidth, it provides over 2.5 TB/s bandwidth for serial memory interface for Hybrid Memory Cube, and 2.3 TB/s bandwidth for the parallel memory interface with support for DDR4 at 2666 Mb/s.

System Interconnect

Most of the FPGA computing solutions are based on PCIe interconnect between the FPGA accelerator board and the host system. In recent years, more and more integrated systems emerge and show promising potential in accelerating cloud applications. One typical example is the Intel HARP systems.[10] In Intel HARP v2 systems, the processor package contains an Intel Broadwell Xeon CPU and an Intel Arria 10 GX FPGA. The CPU-FPGA interconnection is based on Intel QPI, which delivers 6.4 GT/s throughput per lane.

EMERGING ACCELERATORS

The accelerator market is a dynamic marketplace, with ample appetite for investment as opportunities emerge. Ten years ago, it was unclear whether GPU computing would take hold, and the FPGA market was yet to see explosive growth. Today, we see the emergence of new types of accelerators, designed for different, more specific purposes that have the potential for strong impact as GPU computing did ten years ago.

One major new driver for accelerator architectures is machine learning. The impact of machine learning is already commercially significant and expected to continue growing. Machine learning is also compute intensive, requiring additional computing power beyond what is available on most platforms today. These prevailing conditions are ideal for new opportunities for accelerator architectures. As such, several new types of accelerators are emerging specifically around machine learning such as Google's Tensor Processing Unit,[12] Apple's A11 Neural Engine and,[13] and a slew of smaller initiatives. These architectures are programmable but specifically designed around the computational flow of dense matrix multiply and data types found in DL-based algorithms. Similar to machine learning accelerators, computer vision accelerators are emerging specifically for imaging and vision-tasks for autonomous vehicles. Examples include Movidius, MobileEye, and others.

IMPORTANCE OF EDUCATION

Designing an accelerator architecture and throwing it into the market is not enough. No one will come. To win adoptions, and win market share, a vibrant software ecosystem is required. Language

extensions, tools, accelerated libraries, frameworks need to be available to enable software developers to take advantage of the benefits of the accelerator. The GPU Computing movement was fueled by the rapid availability of tools and infrastructure from Nvidia, AMD, Intel, IBM, and other vendors, including CUDA and OpenCL which catalyzed the movement. But for GPUs, more was required. Since the programming model was substantially different than the models used by multicore CPUs, a new generation of programmer needed to be trained with a new parallel programming mindset.

In 2007, Wen-mei Hwu and NVIDIA fellow (was Chief Scientist then) David Kirk took the first step of the parallel programming revolution. They created a senior-level course at the University of Illinois at Urbana-Champaign called ECE498AL - *Programming Massively Parallel Processors*, and was targeted for students from a variety of disciplines. The course is now called ECE408Applied Parallel Programming. With an enrollment currently over 500 students each year. The course material has also been used by numerous universities including MIT, Stanford, Berkeley, CMU, Georgia Tech, and Toronto. The course material has been so popular that it has been translated into Chinese, Japanese, and other languages. In 2016, the University of Illinois partnered with NVIDIA to release the course material as GPU Teaching Kit. More than *300 universities* worldwide have participated in this program. In 2012, 2013, and 2014, Hwu offered a MOOC version of the course through Coursera, with a total of more than 70 000 students registered.

In 2010, Kirk and Hwu published a textbook on programming massively parallel processors. The book has been extremely popular, with more than 30 000 copies sold to date. International editions and translations are available in Chinese, Japanese, Russian, Spanish, Portuguese, Greek, and Korean. The second edition came out in 2012 and the third edition in 2016. As an evidence of impact on researchers in many domains, the book has more than 2970 citations according to Google Scholar.

According to NVIDIA, there are 800 000 registered CUDA Developers worldwide today. An amazing growth rate considering that there were zero about ten years prior. In 2017, there were 3.5 million downloads of the CUDA Software Development Kit.

The work by Hwu *et al.* have stimulated new applications for GPU computing. For example, in 2010, a University of Toronto graduate student Alex Krizhevsky from Geoff Hinton's group took a CUDA GPU class that is taught by Prof. Andreas Moshovas and modeled after the University of Illinois at Urbana-Champaign. Alex Krizhevsky did a GPU implementation of a convolution layer as his class project. Afterward, Alex Krizhevsky, Ilya Sutskever, and Geoff Hinton developed a network, now known as the AlexNet, and trained the network with 1.2 million images using CUDA GPUs and went on to win the 2012 ImageNet competition.

OUTLOOK

We would like to leave our readers with a few predictions. First, there will likely be better programming interfaces to GPUs, FGPAs, and other accelerators. Second, some accelerators will likely be placed near memory to further relieve the pressure of operand reuse.[11] Third, accelerators will be increasingly targeting the latency of AI inferences. For example, Microsoft's Project Brainwave aims to deliver real-time AI solution with FPGAs in the cloud. Fourth, FPGAs will likely find wide adoption in IoT devices and applications that typically have strict power consumption constraints and processing latency constraints. Fifth, FPGAs and GPUs are starting to play a role in autonomous vehicles in recent years. In autonomous cars, there is a huge amount of time series information that need to be processed by the advanced driver-assistance systems. Sixth, emerging accelerator architectures exemplified by TPU and A11 will likely become prominent in the coming ten years as part of the AI revolution. Finally, education will continue to play a key role in the adoption of any new accelerator architectures.

REFERENCES

1. Top 500 List, June 2018; https://www.top500.org/statistics/list/
2. Green 500 List, June 2018; https://www.top500.org/green500/lists/2018/06/
3. https://www.zdnet.com/article/cryptocurrency-miners-bought-3-million-gpus-in-2017/, 2018.
4. https://coinmarketcap.com/currencies/bitcoin/

5. Jon Peddie, "GPU market declined seasonally in Q4; cryptocurrency provides smaller offset as AIB prices rise"; https://www.jonpeddie.com/press-releases/gpu-market-declined-seasonally-in-q4-cryptocurrency-provides-smaller-offset
6. NVLink, Wikipedia; https://en.wikipedia.org/wiki/NVLink, 2018.
7. NVIDIA, NVIDIA GPU Direct technology; http://developer.download.nvidia.com/devzone/devcenter/cuda/docs/GPUDirect_Technology_Overview.pdf, 2018.
8. X. Zhang, *et al*, "DNNBuilder: An automated tool for building high-performance DNN hardware accelerators for FPGAs, in *Proc. of 37th Int. Conf. Comput. Aided Design*, San Diego, CA, USA, Nov. 2018.
9. D. Firestone, *et al*, "Azure accelerated networking: SmartNICs in the public cloud" in *Proc. NSDI*, 2018; https://www.usenix.org/sites/default/files/nsdi18_full_proceedings.pdf
10. Intel, Hardware Accelerator Research Program; https://software.intel.com/en-us/hardware-accelerator-research-program
11. Hwu *et al.*, "Rebooting data access hierarchy of computing systems," *IEEE Int. Conf. Rebooting Comput.*, 2017.
12. "Tensor Processing Unit," Wikipedia; https://en.wikipedia.org/wiki/Tensor_processing_unit, 2018.
13. "Apple A11," Wikipedia; https://en.wikipedia.org/wiki/Apple_A11, 2018.

ABOUT THE AUTHORS

Wen-mei Hwu is a Professor, Acting Department Head and holds the Sanders-AMD Endowed Chair in the Department of Electrical and Computer Engineering. He is the chief scientist of Parallel Computing Institute and director of the IMPACT research group (www.impact.crhc.illinois.edu). He received the ACM SigArch Maurice Wilkes Award, the ACM Grace Murray Hopper Award, the IEEE Computer Society Charles Babbage Award, the ISCA Influential Paper Award, the MICRO Test-of-Time Paper Award, the IEEE Computer Society B. R. Rau Award and the Distinguished Alumni Award in Computer Science of the University of California, Berkeley. He is a fellow of IEEE and ACM.

Sanjay J. Patel is a professor of electrical and computer engineering at the University of Illinois at Urbana-Champaign and serial technology entrepreneur. His research activities include work in computer architecture, computer vision, machine learning, and 3-D imaging. He served as Chief Architect and Chief Technology Officer at AGEIA Technologies, prior to its acquisition by NVIDIA. In 2010, he co-founded Personify, which is a pioneer in immersive and holographic video technologies, and is currently its CEO. He received the Ph.D. degree in computer science and engineering from the University of Michigan, Ann Arbor, in 1999.