Research Journal of Applied Sciences, Engineering and Technology 8(15): 1766-1771, 2014
DOI:10.19026/rjaset.8.1162
ISSN: 2040-7459; e-ISSN: 2040-7467

Submitted: September 13, 2014 Accepted: October 11, 2014 Published: October 15, 2014

Research Article

Advanced Investigation and Comparative Study of Graphics Processing Unit-queries Countered

A. Baskar, Shriram K. Vasudevan and P. Prakash
Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham University, Coimbatore, India

Abstract: GPU, Graphics Processing Unit, is the buzz word ruling the market these days. What is that and how has it gained that much importance is what to be answered in this research work. The study has been constructed with full attention paid towards answering the following question. What is a GPU? How is it different from a CPU? How good/bad it is computationally when comparing to CPU? Can GPU replace CPU, or it is a day dream? How significant is arrival of APU (Accelerated Processing Unit) in market? What tools are needed to make GPU work? What are the improvement/focus areas for GPU to stand in the market? All the above questions are discussed and answered well in this study with relevant explanations.

Keywords: APU, CPU, GPU, graphics, processing, performance, parallel programming and architecture, speed

INTRODUCTION

What is GPU (Graphics Processing Unit)? Expanded as Graphics Processing Unit, the name itself is self-revealing what it is meant for. It is specially used for increasing the graphics handling ability of a computer. But, this brief is not sufficient. It is also referred to be as VPU (Visual Processing Unit) which is intended to swiftly direct and alter memory to speed up the creation of images to be displayed. Experts say this way, GPU (Fatahalian and Houston, 2008; Brodtkorb *et al*., 2010; Davidson and Owens, 2010) is a solo chip processor used to administer and enhance the performance of video and graphics. Usage of GPU is not only restricted to PC, but also be extended to mobile phones, display adaptors', work stations, Medical imaging, Gaming etc. It is inevitable to check the utility areas of GPU. GPUs has got a wide range of application areas which include, Enterprise applications, Medicinal imaging, Gaming, Augmented reality applications, Virtual reality applications and many more. These days' people who use PCs, Televisions, Laptops, Mobile phones want a rich experience of visualizing things on the screen. It is made possible through GPU. The next question to be answered is how GPU is different from the CPU? The following image could be familiar to the readers; the NFS games would be looking and feeling awesome with GPUs in place. Figure 1 is gist of the rich graphics supported by GPU. Main aim of drafting this study is to get a clear cut idea on GPUs.

MATERIALS AND METHODS

How is it different from a CPU? NVIDIA makes this interesting claim. CPU is always regarded as brain of PC, whereas we would say GPU enhances the CPU. So GPU can be definitely certified as Soul of PC. This statement uncovers the importance paid to GPU.

As shown above in the Fig. 2, CPU and GPU both are composed of millions of transistors. The architectural details of the GPU are maintained as trade secret, where we could understand the architecture to a decent extent with the articles available online. CPU devotes most of its transistors towards control, DRAM and cache while assigning a limited amount of transistors to computational tasks and hence limiting the computational ability. GPUs are different in this aspect. GPUs dedicate most of the transistors towards the computational part i.e., ALU which will ultimately increase the computational ability of the processor. And yes, this makes GPU the unanimous choice for applications with undemanding control flow and high numerical intensity. Technically speaking the above claimed statements can be little better re-phrased. CPU is actually built with few cores with lots of cache memory that would be capable of handling only a few software threads at a time. But, GPU on the other hand is built with hundreds of cores which could handle thousands of threads at an instance. The highly admirable view of GPU is about its ability to achieve high acceleration with being more power and cost efficient compared to CPU. CPU can be seen as small group of very smart people who can swiftly do any task

Corresponding Author: A. Baskar, Department of Computer Science and Engineering, Amrita School of Engineering, Amrita Vishwa Vidyapeetham University, Coimbatore, India

Fig. 1: Rich graphics supported by GPUs.

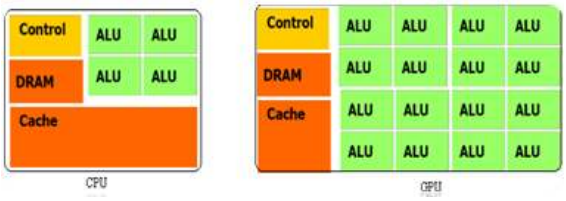

Fig. 2: CPU vs. GPU

given to them at a specified short duration of time. But, GPU could be a large group of not smart people, who are competent individually, but when made to work in a team, they outperform CPU.

History of GPUs: GPU has been evolving consistently over a last and half decades (Harris and Göddeke, 2002). The first and foremost 3D graphics all emerged with early display regulators referred to be as Video Shifters or VAG (Video Address Generators). They were designed to act as a pass-through amongst the core processor and the display unit. The output would be serial bit mapped and synchronized properly based on color, luminance, horizontal and vertical composition etc.

Motorola, the genius in the company revealed the MC6845 VAG (Wikipedia, 2010a). This was one of the most frequently used adapters by IBM, Apple and other giants who wanted the display to be having higher clarity. Since, the business was grooming up well in the VAG sector, Motorola revealed one more addition to the VAG sector, MC6847. It was used in the first generation PCs which did not exclude TRS-80. Figure 3 shows the IBM monochrome display adapter.

Intel was observing Motorola's move into the market and they have understood it is time for Intel to jump in to the market of Graphics adaptors. In 1978, iSBX275 VGCMB (Video Graphics Controller Multimodule Board) was unveiled. Drawing of character bitmaps, arcs, lines along with rectangles are all well augmented with the help of Direct Memory Access (DMA).

In 1980s, VGA (Video Graphics Array) is the one which was used before the advent of GPUs to handle graphics in the PCs. Since VLSI techniques and ability to handle transistors were both developing, VGA's quality and coverage has been enhanced to a greater extent in handling the images and video.

During the year 1985, there came the change which changed the way things looked in the past. Three researchers from Hong Kong Kwok Yuan Ho, Lee Lau and Benny Lau, formed Array Technology Inc. which is referred as ATI Technologies later. The product released by ATI immediately after its inception is OEM Color Emulation Card. It was useful to get the monochrome green, amber or white text in contradiction of black background to a Monitor. This was the path breaking invention in this field.

Following 1985, 1986 was is a very notable year where Texas Instruments unveiled TMS34010 which is a processor with having on chip graphical abilities. Though it had a good graphics processing ability and an instruction set supporting it, it also was used to handle general purpose tasks. TIGA-Texas Instruments Graphics Architecture, which later ruled the graphics world has TMS 34010 as base. In short this decade of 1980s had been wonderful for the graphics world.

Next came the most notable years of invention in the field of Graphics cards. 2D GUIs acceleration has started booming throughout this decade. APIs started finding its place in the market. 3D graphics with support from CPU has really started working out well and it was most common within gamers. Many organizations started working towards the low cost 3D graphics card. As ever, the first few of them failed to attain the glory. Few to mention are S3's ViRGE, ATI's Rage etc. Though they failed, they obviously mean to be the forefathers of the current day GPUs. As the days progressed and brains worked faster, things started happening. Video, 2D GUI, 3D features etc., are all integrated in under one roof of silicon.

In 1990's, came the name which has made revolution in the market of Graphics, OpenGL. It was not a great performer in the early stages like any other invention, with performance flaws. Then the issues were fixed and it started contributing well. There arisen among topographies offered in hardware and those

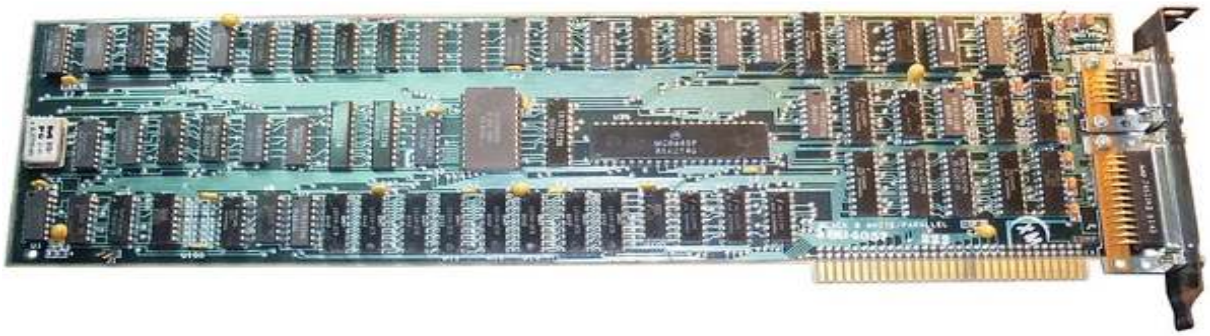

Fig. 3: IBM PC's monochrome display adapter

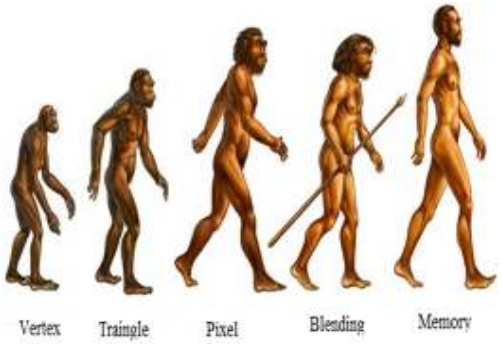

Fig. 4: Evolution of graphics pipeline

Generation - 1 (1993)
- Pipeline stared emerging
- One pixel per clock cycle
- Multiple pixel processing started.

Generation - 2 (1996)
- 3D game cards evolved.

Generation - 3 (1999)
- Graphics pipline implemented
- Accelerated Graphics Port (AGP) replaced PCI
- NVDIA introduced first GPU to the world.

Generation - 4 (2001)
- Fixed pipeline
- Programmable GPU

Generation - 5 (2002)
- Fully Programmable GPU
- Dedicated hardware for pixel shader

Generation - 6(2006)
- GPU as parallel processor
-GPGPU(general purpose computing on GPU)

Fig. 5: Generation wise lookup of GPU

which are mostly offered in OpenGL. Microsoft insisted on DirectX which was most commonly used by Windows gamers.

Graphics pipeline: Figure 4 represents the graphics pipeline, stage by stage starting with Vertex and ending with Memory. Figure 4 shows an idea of evolution of graphics pipeline.

The GPUs which have come to existence in the market today is based on Graphics pipeline and the same has been presented above in diagrammatic format. It actually models the stages that the graphical data really passes through. The stages is actually composed of a combination of blend of hardware and software where hardware is GPU Cores and the software can be something we have already referred to in this study as OpenGL or DirectX. There can be a question now asked that is this pipeline commonly followed by all the GPU manufacturers? Yes, it is precisely followed by all the manufacturers of GPU as NVIDIA (2010) and Owens *et al*. (2008) and ATI. The transformation is simply admirable as the coordinates from the 3D space gets transformed to 2D space in the screen. The generation wise look up is given in Fig. 5.

Can GPU be a CPU? First, GPGPU (Li *et al*., 2009) is the term coming to picture now. To answer this question, GPGPU is general purpose GPU. Indeed, one can simply see it as a GPU which will/can act as a CPU.

As we all know, the fabrication methods and technologies are advancing every now and then. This will make the major changes in the performance and cost. CPUs generally are constructed with having only performance in mind. Many or most of the transistors are moved towards providing non mathematical or computational tasks as caching. But, the GPU (Buck, 2010) is different in this aspect. It is parallel in nature and that permits the GPUs to have more transistors used for computation which will help in achieving high mathematically dense operations, without having need to increase the number of transistors.
To summarize quickly, GPUs are:

- **Powerful:** NVIDIA GeForce 6800 (NVIDIA, 2011, 2012) Ultra is capable of reaching 35.2 GB/sec of memory bandwidth.
- **Less expensive:** Above said processor is just $417, so definitely it is cheaper and cost effective.

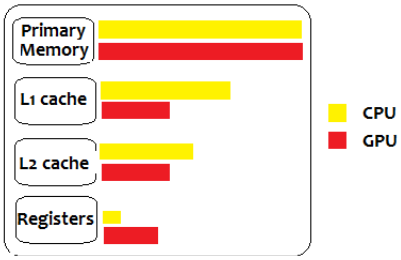

Fig. 6: Memory structure of CPU and GPU

- **Flexible:** Yes, it is really flexible without any much of operating system and other similar dependencies.
- **Programmable:** This is the highlight. It is expandable. It makes the life and creativity live.

How good/bad it is computationally when comparing to CPU? CPU is seen to be computationally less powered comparing GPU. It is because of the following reasons:

- CPU is restricted with having one thread on CPU core whereas in GPU there could be as many as several hundred threads on one GPU core.
- Most importantly, in CPU one thread has to take a pause for the next thread to run. In case of GPU it is different. It can queue at least thousands of threads and has to ability to hang some threads to run others.
- GPU has a policy which lets all threads to do the same thing which would fetch an efficient execution.
- Data parallelism remains the key factor for the success of GPU's which provide very high performance.
- High performance is provided with having above conveyed is unique and it has a programming model which stays different from the legacy CPU (Micikevicius, 2010a) programming model.
- GPUs are able to deliver because of the number of transistors present in it. There is a difference when comparing it with CPU. GPUs use smaller transistors when compared to the CPU. And this makes an obvious difference in the computation and processing ability.
- GPU is normally designed and expected to compute operations of the order of millions which is not in the case of CPU. Also GPUs (Wikipedia, 2010b) are capable of handling large textures as feed. Another point to observe which could seem to be quiet contradictory while reading is the latency related. Latency is for GPU is very high when having the comparison done with the CPU. Reason is being very simple, latency which normally would not exceed milli seconds are certainly not traceable by the human eyes. They will appear as such there is no latency at all.
- The memory structure is not the same and it is overt that they can't really be. A simple diagrammatic explanation would reveal this fact. The cache memory distribution is the area there is a significant difference GPU has given similar emphasize on the L1 and L2 wherein for CPU it is not the case. The cache hit rates are appearing to be comparatively much lesser for GPUs than the case of CPUs. Figure 6 shows the memory of CPU and GPU.
- Comparing instruction set is the next task. Earlier, the life was tedious where each instruction is limited to perform one operation. Means, every operation needs one instruction for the destiny to be reached. It was referred to be as CISC and later, the trend changed where one instruction can be used at different occasions to perform/achieve different tasks. It is termed as RISC. Since the navigation happens towards more of general purpose computing with the GPU the following supports are also made available and it is included:
  - Increased processing ability with included floating point calculations.
  - Uninformed reading option from memory is also provided.
  - Options to use Loops and Conditional constructs made the GPUs more GP.
- These above points are all taking the drift towards GPU becoming more like CPU with inclusion of CPU like elements as loops, conditional constructs, floating point capabilities etc.

Programming support for GPU: CUDA and OpenCL are the two wings of graphics programming. Expanded as Compute Unified Device Architecture, CUDA is a parallel computing platform created and innovated by NVIDIA. It has been used by the GPUs manufactured by NVIDIA. With CUDA in place, GPUs can be made more flexible and can be drifted towards general purpose computing. CUDA has been well supported by a strong set of libraries, Compiler directives etc. Most important point to be mentioned is the tremendous support it has for the programming languages. It supports C and C++ (Owens *et al*., 2008) which would be sufficient for the developers to be comfortable with. CUDA can't simply be referred as name for the API; instead it should be seen as architecture all together. SDK tool kits are available for variety of platforms starting from most used Linux, Windows and more sophisticated MAC OS for Apple.Well, is there a competitor from any other source? Yes, there is a competitor. It is named as OpenCL and it is open

Table 1: CUDA and OpenCL API comparison

	CUDA	OPENCL
Development models	Data parallel kernels support Efficient syntax used for deep host and device program integration support	Data and task parallel kernels support only separate compilation and kernel invocation with API calls for deep host and device program integration support
Tool chains	Mixed device/host code or custom kernel invocation syntax is used for host program	No Custom tool chain needed for host program
Kernel programming	Access to work-item indices through built-in variables Address space qualification needed for kernel pointer arguments: No defaults to global memory used First class built in vector types: Just vector types defined, no operators or functions Voting functions: Yes (CC 1.2 or greater) Asynchronous memory copying and prefetch functions: No	Access to work item indices through built-in functions Address space qualification needed for kernel pointer arguments: yes First class built in vector types: Yes: vector types, literals, built in operators and functions Voting functions: Only as extension Asynchronous memory copying and prefetch functions: Yes
Execution model	Grid Thread block Thread Thread ID Block index Thread index	NDRange Work group Work item Global ID Block ID Local ID
Memory model	Host memory Global or device memory Local memory Constant memory Texture memory Shared memory Registers	Host memory Global memory Global memory Constant memory Global memory Local memory Private memory

Table 2: GPU programming paradigms

	Pre-GPGPU	GPGPU	GPU Computing
Architectures	Fixed function Pipeline	Programmable Pipeline	Programmable Parallel computing
Year and language development	1992-OpenGL 1995-DirectX	2003-Cg, HLSL,GLSL 2004-BrookGPU	2006-CUDA 2008-OpenCL

Table 3: The AMD Accelerated Processing Unit - Architectural Details

	Year	Heterogeneous Systems Architecture (HSA) feature
Optimized platform	2012 Trinity APUs	C++ Support for GPU Shared Power Management (Priority goes to the processor most suited to the current tasks) GPU can access the entire system memory through the translation services and page fault management of the HSA MMU.
Architectural integration	2014 PlayStation 4, Kaveri APUs	Unified Address Space for CPU and GPU-Pointers can now be freely passed between CPU and GPU, hence enabling zero-copy. Fully coherent memory between CPU and GPU-Cache coherency is maintained. GPU uses pageable system memory via CPU pointers-GPU can take advantage of the shared virtual memory between CPU and GPU and pageable system memory can now be referenced directly by the GPU, instead of being copied or pinned before accessing.
System integration	2015 Carrizo APU	GPU compute context switch-allowing a multi-tasking environment and also faster interpretation between applications, compute and graphics. GPU graphics pre-emption-Long-running graphics tasks can be pre-empted so processes have low latency access to the GPU. Quality of Service-In addition to context switch and pre-emption, hardware resources can be either Equalized or prioritized among multiple users and applications.

source software. It is equally good computationally when compared to other options available in the market. Support is rendered to CPUs, GPUs and DSPs which is collar lifting fact on the OpenCL. A comparison is always nice and the below table provides the same between the OpenCL and CUDA with respect to API (Table 1 and 2).

The AMD Accelerated Processing Unit, formerly known as Fusion, is a series of 64-bit microprocessors from AMD designed to act as a CPU and graphics accelerator (GPU) on a single chip (Table 3).

RESULTS AND DISCUSSION

GPUs presently are not interacting much with cloud kind of infrastructure. But, in future it could be the need of the hour to make GPU (Micikevicius, 2010b) work in coherence with cloud. It is definitely

possible to have a better bottleneck reduced application support in the cloud with GPU. Technology grows towards the innovation and following are simple examples which we are eyeing in front of us in the recent past:

- CASS is one of the leading players coming up with Embedded GPU platforms, Hoopoe GPU cloud computing and CLIPP.
- NVIDIA is focusing towards the deep learning algorithms which definitely would benefit in producing better artificial intelligent systems.
- AMD has started to work on next generation processors named Carrizo which is expected to be a boom in the market of graphics.

CONCLUSION

We the authors have made efforts in understanding of the trends in GPU and the same has been presented in the study. The paper has clarified on what a GPU is and how is it different from a CPU. Also is it a solid replacement for legacy CPU or it is just hype in the modern era? How is the market reacting towards this innovation and what are the companies working to make the revolution happen? All these have been discussed and clarity has been given to the same. Also where can the GPU era head towards is projected in the paper. The same paper can be further crafted with giving more details on all the individual GPUs available in the market.

REFERENCES

Brodtkorb, A.R., C. Dyken, T.R. Hagen, J.M. Hjelmervik and O.O. Storaasli, 2010. State-of-the-art in heterogeneous computing. Sci. Prog., 18(1): 1-33.

Buck, I., 2010. The Evolution of GPUs for General Purpose Computing. GTC 2010. Retrieved from: https://docs.google.com/viewer?url=http://www.nvidia.com/content/GTC-2010/pdfs/2275_GTC2010.pdf.

Davidson, A. and J.D. Owens, 2010. Toward techniques for auto-tuning GPU algorithms. Proceeding of the Para 2010: State of the Art in Scientific and Parallel Computing.

Fatahalian, K. and M. Houston, 2008. A closer look at GPUs. Commun. ACM, 51: 50-57.

Harris, M. and D. Göddeke, 2002. General-purpose computation on graphics hardware. Retrieved from: http://gpgpu.org.

Li, Y., J. Dongarra and S. Tomov, 2009. A note on auto-tuning gemm for GPUs. Proceeding of the 9th International Conference on Computational Science: Part I.

Micikevicius, P., 2010a. Analysis-driven performance optimization. Proceedings of the GPU Technology Conference, Session 2012.

Micikevicius, P., 2010b. Fundamental performance optimizations for GPUs. Proceedings of the GPU Technology Conference.

NVIDIA, 2010. NVIDIA's Next Generation CUDA Compute Architecture: Fermi.

NVIDIA, 2011. NVIDIA CUDA Programming Guide 4.1.

NVIDIA, 2012. NVIDIA GeForce GTX 680. Technical Report, NVIDIA Corporation.

Owens, J., M. Houston, D. Luebke, S. Green, J. Stone and J. Phillips, 2008. GPU computing. Proc. IEEE, 96(5): 879-899.

Wikipedia, 2010a. Video Card. Retrieved from: http://en.wikipedia.org/wiki/Video_card. (Accessed on: November, 2010a)

Wikipedia, 2010b. Graphics Processing Unit. Retrieved from: http://en.wikipedia.org/wiki/Graphicsprocessingunit. (Accessed on: November, 2010b)