Graphical Process Unit A New Era

Santosh Kumar, Shashi Bhushan Jha, Rupesh Kumar Singh
Students
Computer Science and Engineering
Dronacharya College of Engineering, Greater Noida, India

Abstract **- Now in present days every computer is come with G.P.U (graphical process unit). The graphics processing unit (G.P.U) has become an essential part of today's mainstream computing systems. Over the past 6 years, there has been a marked increase in the performance and potentiality of G.P.U. The modern G.P.Us is not only a powerful graphics engine but also a deeply parallel programmable processor showing peak arithmetic and memory bandwidth that substantially outpaces its CPU counterpart. The G.P.U's speedy increase in both programmability and capability has spawned a research community that has successfully mapped a broad area of computationally demanding, mixed problems to the G.P.U. This effort in general-purpose computing on the G.P.Us, also known as G.P.U computing, has positioned the G.P.U as a compelling alternative to traditional microprocessors in high-performance computer systems of the future. We illustrate the history, hardware, and programming model for G.P.U computing, abstract the state of the art in tools and techniques, and present 4 G.P.U computing successes in games physics and computational physics that deliver order-of-magnitude performance gains over optimized CPU applications.**

Index Terms **- G.P.U, History of G.P.U, Future of G.P.U, Problems in G.P.U, eG.P.U, Integrated graphics**

I. INTRODUCTION

A graphics processing unit (G.P.U), also known as visual processing unit (VPU), is a specialized electronic circuit structured to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display. G.P.Us are utilised in embedded system, mobile phones, PCs, workstations, and games consoles. Modern G.P.Us are very efficient at manipulating computer graphics and image processing, and their deeply parallel structure makes them more effective than general-purpose CPUs for algorithms where processing of large blocks of data is done in parallel. In a PC, a G.P.U can be present on a video card, or it can be on the motherboard or in some CPUs on the CPU die.

The term G.P.U was popularized by NVidia company in 1999, who marketed the GeForce 256 as "the world's 1st 'G.P.U', or Graphics Processing Unit, a single-chip processor with intersperse transform, lighting, triangle setup or clipping, and rendering engines that are capable of processing a minimum of 10 million polygons / second". Rival ATI Technologies coined the term visual processing unit or VPU with the release of the Radeon 9700 Graphics card in mid of 2002. G.P.U is structured specifically to perform the many floating-point calculations essential to 3 Dimensional graphics rendering. Modern G.P.U processors are massively parallel, and are fully programmable. The parallel floating point computing power found in a modern G.P.U is orders of magnitude higher than a CPU. G.P.Us can be found in a wide range of systems, from PC and laptops to mobile phones and super computers. With their parallel structure, G.P.Us implement a number of 2D and 3 Dimensional graphics primitives processing in hardware, making them much faster than a general purpose CPU at these operations.

II. HISTORY

Part 1: (1976 - 1995) The Early Days of 3 Dimensional Consumer Graphics
Part 2: (1995 - 1999) 3 Dimensional fx Voodoo: The Games-changer
Part 3: (2000 - 2006) The Nvidia company vs. ATI Era Begins
Part 4: (2006 - 2013) The Modern G.P.U: Stream processing units a.k.a. GPG.P.U

Arcade system boards have been using specialized graphics chips since the 1970s. Fujitsu's MB14241 video shifter was utilised to accelerate the drawing of sprite graphics for various 1970s arcade gamess from Taito and Midway, such as Gun Fight (1975), Sea Wolf (1976) and Space Invaders (1978). The Namco Galaxian arcade system in 1979 utilised specialized graphics hardware supporting RGB colors, multi-colored sprites and tile map backgrounds. The Galaxian hardware was widely utilized during the golden age of arcade video games, by games companies such as Namco, Centuri, Gremlin, Midway, Nichibutsu, Sega and Taito. In the home video games console market, theAtari 2600 in 1977 utilised a video shifter known as the Television Interface Adaptor.

1980s

The NEC µPD7220 GDC (Graphics Display Controller), developed during 1979-1981, is a video interface controller capable of drawing lines, circles, arcs, and character graphics to a bit mapped display. It was one of the 1st implementations of a graphics display controller as a single Large Scale Integration (LSI) intersperse circuit chip, potentially viable the structure of low-cost, high-performance video graphics cards such as those from Number Nine Visual Technology. It became one of the best known of what became known as graphics processing units in the 1980s. It came norm with the NEC PC-9801, APC III, Tulip System-1 and Epson QX-10.

In 1982, Intel made the iSBX 275 Video Graphics Controller Multimodule Board, for industrial systems based on the Multibus norm. The card was based on the NEC 82720 GDC, and accelerated the drawing of lines, arcs, rectangles, and character bitmaps. The framebuffer was also accelerated through loading via DMA. The board was intended for use with Intel's line of Multibus industrial single-board computer plugin cards.

Some of the 1st PCs to come norm with a G.P.U in the early 1980s include the µPD7220 based computers mentioned above, as well as TMS9918 based computers such as the TI-99/4A, MSX and Sega SC-3000. Video games consoles also began using graphics co-processors in the early 1980s, such as the ColecoVision and Sega SG-1000's TMS9918, the Atari 5200's ANTIC, and the Nintendo Entertainment System's Picture Processing Unit.

In the arcades, specialized video hardware for sprite-based pseudo-3 Dimensional graphics began appearing in the early 1980s. In 1981, the Sega VCO Object hardware introduced sprite-scaling with full-color graphics. In 1982, the Sega Zaxxon hardware produced scrolling graphics in an isometric perspective.] The Namco Pole Position system in 1982 utilised several custom graphics chips to produce colorful sprite-scaling background and foreground objects. This culminated in Sega's Super Scaler graphics hardware, which produced the most advanced pseudo-3 Dimensional graphics of the 1980s, for systems such as the Sega Space Harrier (1985), Sega OutRun (1986) and X Board (1987).

In 1985, the Commodore Amiga featured a G.P.U advanced for a PC at the time. It supported line draw, area fill, and included a type of stream processor known as ablitter which accelerated the movement, manipulation and combination of multiple arbitrary bitmaps. Also included was a coprocessor with its own (primitive) instruction set capable of directly invoking a sequence of graphics operations without CPU intervention. Prior to this and for quite some time after, many other PC systems instead utilised their main, general-purpose CPU to handle almost every aspect of drawing the display, short of generating the final video signal.

In 1986, Texas Instruments released the TMS34010, the 1st microprocessor with on-chip graphics capabilities. It could run general-purpose code, but it had a very graphics-oriented instruction set. In 1990-1991, this chip would become the basis of the Texas Instruments Graphics Architecture ("TIGA") Windows accelerator cards.

In 1987, the IBM 8514 graphics system was released as one of the 1st video cards for IBM PC compatibles to implement fixed-function 2D primitives in electronic hardware. The same year, Sharp released the X68000, which utilised a custom graphics chipset that was powerful for a home computer at the time, with a 65,536 color palette and hardware support for sprites, scrolling and multiple playfields, eventually serving as a development machine for CaPCom's CP System arcade board. Fujitsu later competed with the FM Towns computer, released in 1989 with support for a full 16,777,216 color palette.

In 1988, the 1st dedicated polygonal 3 Dimensional graphics boards were introduced in arcades with the Namco System 21 and Taito Air System.

1990s

In 1991, S3 Graphics introduced the S3 86C911, which its structureers named after the Porsche 911 as an implication of the performance increase it promised. The 86C911 spawned a host of imitators: by 1995, all major PC graphics chip makers had added 2D acceleration support to their chips. By this time, fixed-function Windows accelerators had surpassed expensive general-purpose graphics coprocessors in Windows performance, and these coprocessors faded away from the PC market.

Throughout the 1990s, 2D GUI acceleration continued to evolve. As manufacturing capabilities improved, so did the level of integration of graphics chips. Additional application programming interfaces (APIs) arrived for a variety of tasks, such as Microsoft's WinG graphics library for Windows 3.x, and their later DirectDraw interface for hardware acceleration of 2D gamess within Windows 95 and later.

In the early- and mid-1990s, CPU-assisted real-time 3 Dimensional graphics were becoming increasingly common in arcade, computer and console gamess, which led to an increasing public demand for hardware-accelerated 3 Dimensional graphics. Early examples of mass-marketed 3 Dimensional graphics hardware can be found in arcade system boards such as the Sega Model 1, Namco System 22, and Sega Model 2, and the fifth-generation video games consoles such as the Saturn, PlayStation and Nintendo 64. Arcade systems such as the Sega Model 2 and Namco Magic Edge Hornet Simulator were capable of hardware T&L (transform, clipping, and lighting) years before appearing in consumer graphics cards. Fujitsu, which worked on the Sega Model 2 arcade system, began working on integrating T&L into a single LSIsolution for use in home computers in 1995.

In the PC world, notable failed 1st tries for low-cost 3 Dimensional graphics chips were the S3 ViRGE, ATI Rage, and Matrox Mystique. These chips were essentially previous-generation 2D accelerators with 3 Dimensional features bolted on. Many were even pin-compatible with the earlier-generation chips for ease of implementation and minimal cost. Initially, performance 3 Dimensional graphics were possible only with discrete boards dedicated to accelerating 3 Dimensional functions (and lacking 2D GUI acceleration entirely) such as the PowerVR and the 3 Dimensionalfx Voodoo. However, as manufacturing technology continued to progress, video, 2D GUI acceleration and 3 Dimensional functionality were all intersperse into one chip.Rendition's Verite chipsets were among the 1st to do this well enough to be worthy of note. In 1997, Rendition went a step further by collaborating with Hercules and Fujitsu on a "Thriller Conspiracy" project which combined a Fujitsu FXG-1 Pinolite geometry processor with a Vérité V2200 core to create a graphics card with a full T&L engine years before Nvidia company's GeForce 256. This card, structureed to reduce the load placed upon the system's CPU, never made it to market.

OpenGL appeared in the early '90s as a professional graphics API, but originally suffered from performance issues which allowed theGlide API to step in and become a dominant force on the PC in the late '90s.[27] However, these issues were quickly overcome and the Glide API fell by the wayside. Software implementations of OpenGL were common during this time, although the influence of OpenGL eventually led to widespread hardware support. Over time, a parity emerged between features offered in hardware and those offered in OpenGL. DirectX became popular among Windows games developers during the late 90s. Unlike OpenGL, Microsoft insisted on providing strict one-to-one support of hardware. The approach made DirectX less popular as a

standalone graphics API initially, since many G.P.Us provided their own specific features, which existing OpenGL applications were already able to benefit from, leaving DirectX often one generation behind. (See: Comparison of OpenGL and Direct3 Dimensional.)

Over time, Microsoft began to work more closely with hardware developers, and started to target the releases of DirectX to coincide with those of the supporting graphics hardware. Direct3 Dimensional 5.0 was the 1st version of the burgeoning API to gain widespread adoption in the gaming market, and it competed directly with many more-hardware-specific, often proprietary graphics libraries, while OpenGL maintained a strong following. Direct3 Dimensional 7.0 introduced support for hardware-accelerated transform and lighting (T&L) for Direct3 Dimensional, while OpenGL had this capability already exposed from its inception. 3 Dimensional accelerator cards moved beyond being just simple rasterizers to add another significant hardware stage to the 3 Dimensional rendering pipeline. The Nvidia company GeForce 256 (also known as NV10) was the 1st consumer-level card released on the market with hardware-accelerated T&L, while professional 3 Dimensional cards already had this capability. Hardware transform and lighting, both already existing features of OpenGL, came to consumer-level hardware in the '90s and set the precedent for later pixel shader and vertex shader units which were far more flexible and programmable.

2000 – 2005

With the advent of the OpenGL API and similar functionality in DirectX, G.P.Us added shading to their capabilities. Each pixel could now be processed by a short program that could include additional image textures as inputs, and each geometric vertex could likewise be processed by a short program before it was projected onto the screen. Nvidia company was 1st to produce a chip capable of programmable shading, the GeForce 3 (code named NV20). By October 2002, with the introduction of the ATI Radeon 9700 (also known as R300), the world's 1st Direct3 Dimensional 9.0 accelerator, pixel and vertex shaders could implement looping and lengthy floating point math, and in general were quickly becoming as flexible as CPUs, and orders of magnitude faster for image-array operations. Pixel shading is often utilised for things like bump mapping, which adds texture, to make an object look shiny, dull, rough, or even round or extruded.

2006 to present

With the introduction of the GeForce 8 series, which was produced by Nvidia company, and then new generic stream processing unit G.P.Us became a more generalized computing device. Today, parallel G.P.Us have begun making computational inroads against the CPU, and a subfield of research, dubbed G.P.U Computing or GPG.P.U for General Purpose Computing on G.P.U, has found its way into fields as diverse as machine learning,] oil exploration, scientific image processing, linear algebra, statistics, 3 Dimensional reconstruction and even stock options pricing determination. Nvidia company's CUDA platform was the earliest widely adopted programming model for G.P.U computing. More recently OpenCL has become broadly supported. OpenCL is an open norm defined by the Khronos Group which allows for the development of code for both G.P.Us and CPUs with an emphasis on portability. OpenCL solutions are supported by Intel, AMD, Nvidia company, and ARM, and according to a recent report by Evan's data OpenCL is the GPG.P.U development platform most widely utilised by developers in both the US and Asia Pacific.

III. THE FUTURE

. So what next for these two giants of computer graphics? AMD's recent $36 million loss hasn't exactly had investors lining up to buy shares in the company, and the X86 side of the business means that it's fighting against not just Nvidia company, but the market leading Intel too. But when it comes to its G.P.U business, despite some excellent products at tempting price points, the company still faces a perception problem. Jump onto any article about AMD or Nvidia company and you'll likely find a comment about AMD's poor drivers, or shoddy hardware support. Overcoming that now ingrained perception of the company is a tall order, even for one with millions of marketing dollars to spend.

“The fact is, it takes an awful amount of time to fix reputation for drivers. It stays around. It's the ‘halo effect’ in reverse. – AMD

"I think we've been suffering from the year 2000 to be honest," says AMD's Huddy. "This is a legacy of opinion which is not justified. Before you arrived, we were talking together about what we are doing about dispelling that. It's a myth; it's not the case. The fact is, it takes an awful amount of time to fix reputation for drivers. It stays around. It's the ‘halo effect’ in reverse. I believe there's a lot of independent research which shows AMD drivers are really solid, and I believe we will be able to come up with research that proves AMD drivers are better, and I believe we'll be able to do that."

"There is this oddity," continued Huddy. "Nvidia company can produce an driver on a GamesWorks title which comes out on the day it is released, and it will raise the performance, and you'll think, 'hey, nice one Nvidia company, good job.' Interestingly it can happen because they've optimised a piece of broken code that they supplied to a gamess developer. Slightly worrisome again. They get the credit, for fixing the broken code they supplied, and we get criticised for our failure to fix the broken code that was supplied to us at very short notice. It's a real problem going on here, so let's name names and get it sorted."

As for Nvidia company, it's making further inroads into ARM. Just recently, the company launched its 32-bit Tegra K1 ARM SOC, with a custom 64-bit version featuring its Denver CPU arriving later this year. Nvidia company has also branched out into hardware, with the company hoping that its Shield Portable and the recently released Shield Tablet do for mobile gaming what its desktop G.P.Us have done for PC gaming. Its desktop G.P.Us aren't going away anytime soon (with a new range of flagships rumoured to be announced very soon), but Nvidia company is confident that mobile is going to play a bigger part of its business in the future.

"At some point, you could envisage a future where the two markets [PC and Mobile] converge, or at least the lines blur between the two, and maybe really the only difference is power consumption," says Nvidia company's Tamasi. "In fact, architecturally we've

been working that way. If you look at K1 that's Kepler, and that's what a Titan is built on. The fundamental difference there is power. From a developer perspective, we're trying to put the infrastructure in place that lets them do that."

"Do I think that a phone games and a PC games are going to be the same games?" continued Tamasi. "Hard to say. If nothing else, the way you interact with the device is different, and the way people games on phones today tends to be different, or there's a different playing time. Who's to say the future isn't that you sit down, you put your phone on the table, it wirelessly pairs with a keyboard and mouse and display, and you're playing the next-gen version of your favourite games in that way. Is the market going to evolve that way? Don't know, but we're doing everything we can to make that happen, because if it does, that's great for Nvidia company."

IV. PROBLEMS IN G.P.U

At Eurographics 2005, the authors presented their list of top ten problems in GPG.P.U. At the time we hoped that these problems would help focus the efforts of the GPG.P.U community on broad problems of general interest. In the intervening two years, substantial progress has been made on many of these problems, yet the problems themselves continue to be worthy of further study.

The killer app: Perhaps the most important question facing the community is finding an application that will drive the purchase of millions of G.P.Us. The number of G.P.Us sold today for computation is minuscule compared to the overall G.P.U market of half a billion units per year; a mass-market application that spurred millions of G.P.U sales, potentially viable a task that was not previously possible, would mark a major milestone in G.P.U computing.

Programming models and tools: With the new program-ming systems in Section IV, the state of the art over the past year has substantially improved. Much of the difficulty of early GPG.P.U programming has dissipated with the new capabilities of these programming systems, though support for debugging and profiling on the hardware is still primitive. One concern going forward, however, is the proprietary nature of the tools. Norm languages, tools, and APIs that work across G.P.Us from multiple vendors would advance the field, but it is as yet unclear whether those solutions will come from academia, the G.P.U vendors, or third-party software companies, large or small.

G.P.U in tomorrow's computer?: The fate of coprocessors in commodity computers (such as floating-point copro-cessors) has been to move into the chipset or onto the microprocessor. The G.P.U has resisted that trend with continued improvements in performance and functionality and by becoming an increasingly important part of today's computing environmentsVunlike with CPUs, the demand for continued G.P.U performance increases has been consistently large. However, economics and potential performance are motivating the migration of powerful G.P.U functionality onto the chipset or onto the processor die itself. While it is fairly clear that graphics capability is a vital part of future computing systems, it is wholly unclear which part of a future computer will provide that capability, or even if an increasingly important G.P.U with parallel computing capabilities could absorb a CPU.

Structure tradeoffs and their impact on the programming model: G.P.U vendors are constantly weighing decisions regarding flexibility and features for their programmers: how do those decisions impact the programming model and their ability to build hardware capable of top performance? An illustrative example is data conditionals. Today, G.P.Us support conditionals on a thread granularity, but conditionals are not free; any G.P.U today pays a performance penalty for incoherent branches. Program-mers want a small branch granularity so each thread can be independent; architects want a large branch granularity to build more efficient hardware. Another important structure decision is thread granularity: less powerful but many lightweight threads versus fewer, more powerful heavy-weight threads. As the programmable aspects of the hardware increasingly take center stage, and the G.P.U continues to mature as a general-purpose platform, these tradeoffs are only increasing in importance.

V. G.P.U COMPANIES

Many companies have produced G.P.Us under a number of brand names. In 2009, Intel, Nvidia company and AMD/ATI were the market share leaders, with 49.4%, 27.8% and 20.6% market share respectively. However, those numbers include Intel's intersperse graphics solutions as G.P.Us. Not counting those numbers, Nvidia company and ATI control nearly 100% of the market as of 2008. In addition, S3 Graphics (owned by VIA Technologies) and Matrox produce G.P.Us.

VI. G.P.U FIRMS

1. Dedicated graphics cards

The G.P.Us of the most powerful class typically interface with the motherboard by means of an expansion slot such as PCI Express (PCIe) or Accelerated Graphics Port (AGP) and can usually be replaced or upgraded with relative ease, assuming the motherboard is capable of supporting the upgrade. A few graphics cards still use Peripheral Component Interconnect (PCI) slots, but their bandwidth is so limited that they are generally utilised only when a PCIe or AGP slot is not available.

A dedicated G.P.U is not necessarily removable, nor does it necessarily interface with the motherboard in a norm fashion. The term "dedicated" refers to the fact that dedicated graphics cards have RAM that is dedicated to the card's use, not to the fact that most dedicated G.P.Us are removable. Dedicated G.P.Us for portable computers are most commonly interfaced through a non-norm and often proprietary slot due to size and weight constraints. Such ports may still be considered PCIe or AGP in terms of their logical host interface, even if they are not physically interchangeable with their counterparts.

Technologies such as SLI by Nvidia company and CrossFire by AMD allow multiple G.P.Us to draw images simultaneously for a single screen, increasing the processing power available for graphics.

2. Intersperse graphics solutions

Intersperse graphics solutions, shared graphics solutions, or intersperse graphics processors (IGP) utilize a portion of a computer's system RAM rather than dedicated graphics memory. IGPs can be intersperse onto the motherboard as part of the chipset, or within the same die as CPU (like AMD APUor Intel HD Graphics). Some of AMD's IGPs use dedicated sideport memory on some motherboards[clarification needed]. Computers with intersperse graphics account for 90% of all PC shipments.[36][needs update] These solutions are less costly to implement than dedicated graphics solutions, but tend to be less capable. Historically, intersperse solutions were often considered unfit to play 3 Dimensional gamess or run graphically intensive programs but could run less intensive programs such as Adobe Flash. Examples of such IGPs would be offerings from SiS and VIA circa 2004.[37] However, modern intersperse graphics processors such as AMD Accelerated Processing Unit and Intel HD Graphics are more than capable of handling 2D graphics or low stress 3 Dimensional graphics.

As a G.P.U is extremely memory intensive, an intersperse solution may find itself competing for the already relatively slow system RAM with the CPU, as it has minimal or no dedicated video memory. IGPs can have up to 29.856 GB/s of memory bandwidth from system RAM, however graphics cards can enjoy up to 264 GB/s of bandwidth over its memory-bus. Older intersperse graphics chipsets lacked hardware transform and lighting, but newer ones include it.

3. Hybrid solutions

This newer class of G.P.Us competes with intersperse graphics in the low-end desktop and notebook markets. The most common implementations of this are ATI's HyperMemory and Nvidia company's TurboCache.

Hybrid graphics cards are somewhat more expensive than intersperse graphics, but much less expensive than dedicated graphics cards. These share memory with the system and have a small dedicated memory cache, to make up for the high latency of the system RAM. Technologies within PCI Express can make this possible. While these solutions are sometimes advertised as having as much as 768MB of RAM, this refers to how much can be shared with the system memory.

4. Stream Processing and General Purpose G.P.Us (GPG.P.U)

It is becoming increasingly common to use a general purpose graphics processing unit as a improved form of stream processor. This concept turns the massive computational power of a modern graphics accelerator's shader pipeline into general-purpose computing power, as opposed to being hard wired solely to do graphical operations. In some applications requiring massive vector operations, this can yield several orders of magnitude higher performance than a conventional CPU. The two largest discrete (see "Dedicated graphics cards" above) G.P.U structureers, ATI and Nvidia company, are beginning to pursue this approach with an array of applications. Both Nvidia company and ATI have teamed withStanford University to create a G.P.U-based client for the Folding@home distributed computing project, for protein folding calculations. In some circumstances the G.P.U calculates forty times faster than the conventional CPUs traditionally utilised by such applications.

GPG.P.U can be utilised for many types of embarrassingly parallel tasks including ray tracing. They are generally suited to high-throughput type computations that exhibit data-parallelism to exploit the wide vector width SIMD architecture of the G.P.U.

Furthermore, G.P.U-based high performance computers are starting to play a significant role in large-scale modelling. Three of the 10 most powerful supercomputers in the world take advantage of G.P.U acceleration.

NVIDIA COMPANY cards support API extensions to the C programming language such as CUDA ("Compute Unified Device Architecture") and OpenCL. CUDA is specifically for NVIDIA COMPANY G.P.Us whilst OpenCL is structureed to work across a multitude of architectures including G.P.U, CPU and DSP (using vendor specific SDKs). These technologies allow specified functions (kernels) from a normal C program to run on the G.P.U's stream processors. This makes C programs capable of taking advantage of a G.P.U's ability to operate on large matrices in parallel, while still making use of the CPU when appropriate. CUDA is also the 1st API to allow CPU-based applications to directly access the resources of a G.P.U for more general purpose computing without the limitations of using a graphics API.

Since 2005 there has been interest in using the performance offered by G.P.Us for evolutionary computation in general, and for accelerating the fitness evaluation in genetic programming in particular. Most approaches compile linear or tree programs on the host PC and transfer the executable to the G.P.U to be run. Typically the performance advantage is only obtained by running the single active program simultaneously on many example problems in parallel, using the G.P.U's SIMD architecture. However, substantial acceleration can also be obtained by not compiling the programs, and instead transferring them to the G.P.U, to be interpreted there. Acceleration can then be obtained by either interpreting multiple programs simultaneously, simultaneously running multiple example problems, or combinations of both. A modern G.P.U (e.g. 8800 GTX or later) can readily simultaneously interpret hundreds of thousands of very small programs.

5. *External G.P.U (eG.P.U)*

An external G.P.U is utilised on for example laptops. Laptops might have a lot of RAM and a lot of processing power (CPU), but often lack a powerful graphics card (and instead have an on-board graphics chip). On-board graphics chips are often not powerful enough for playing the latest gamess, or for other tasks (video editing, ...).

Therefore it is desirable to be able to attach to some external PCIe bus of a notebook. That may be an x1 2.0 5Gbit/s expresscard or mPCIe (wifi) port or a 10Gbit/s/16Gbit/s Thunderbolt1/Thunderbolt2 port. Those ports being only available on some candidate notebook systems.

External G.P.U's have had little official vendor support. Promising solutions such as Silverstone T004 (aka ASUS XG2) and MSI GUS-II were never released to the general public. MSI's Gamesdock promising to deliver a full x16 external PCIe bus to a purpose built compact 13" MSI GS30 notebook. Lenovo and Magma partnering in Sep-2014 to deliver official Thunderbolt eG.P.U support.

This has not stopped enthusiasts from creating their own DIY eG.P.U solutions. expresscard/mPCIe eG.P.U adapters/enclosures are usually acquired from BPlus (PE4C, PE4L, PE4C), or EXP GDC. native Thunderbolt eG.P.U adaptere/enclosures acquired from One Stop Systems, AKiTiO, Sonnet (often rebadge as Other World Computing - OWC) and FirmTek.

VII. CONCLUSION

With the rising importance of G.P.U computing, G.P.U hardware and software are changing at a remarkable pace. In the uPComing years, we expect to see several changes to allow more flexibility and performance from future G.P.U computing systems:

- At Supercomputing 2006, both AMD and NVIDIA COMPANY announced future support for double-precision floating-point hardware by the end of 2007. The addition of double-precision support removes one of the major obstacles for the adoption of the G.P.U in many scientific computing applications.
- Another uPComing trend is a higher bandwidth path between CPU and G.P.U. The PCI Express bus between CPU and G.P.U is a bottleneck in many applications, so future support for PCI Express 2, HyperTransport, or other high-bandwidth connec-tions is a welcome trend. Sony's PlayStation 3 and Microsoft's XBox 360 both feature CPU– G.P.U connections with substantially greater band-width than PCI Express, and this additional bandwidth has been welcomed by developers. We expect the highest CPU–G.P.U bandwidth will be delivered by . . .
- future systems, such as AMD's Fusion, that place both the CPU and G.P.U on the same die. Fusion is initially targeted at portable, not high-performance, systems, but the lessons learned from developing this hardware and its heteroge-neous APIs will surely be applicable to future single-chip systems built for performance. One open question is the fate of the G.P.U's dedicated high-bandwidth memory system in a computer with a more tightly coupled CPU and G.P.U.
- Pharr notes that while individual stages of the graphics pipeline are programmable, the structure of the pipeline as a whole is not , and proposes future architectures that support not just program-mable shading but also a programmable pipeline. Such flexibility would lead to not only a greater variety of viable rendering approaches but also more flexible general-purpose processing.

Systems such as NVIDIA COMPANY's 4-G.P.U Quadroplex are well suited for placing multiple coarse-grained G.P.Us in a graphics system. On the G.P.U computing side, however, fine-grained cooperation between G.P.Us is still an unsolved problem. Future API support such as Microsoft's Windows Display Driver Model 2.1 will help multiple G.P.Us to collaborate on complex tasks, just as clusters of CPUs do today

VIII. ACKNOWLEDGMENT

1st of all I would like to thank my teachers for giving me an opportunity to write a project on this topic. I not only enjoyed doing this project, but also learned a lot of new things. Secondly, I would like to thank my friends for motivating me and appreciating my work. I would also like to thank my parents for helping me and encouraging me to go my own way. Last, but not the least, I would like to thank God, who made all the things possible.

REFRECNCE

[1] J. D. Owens, D. Luebke, N. Govindaraju,M. Harris, J. Kru¨ger, A. E. Lefohn, andT. Purcell, BA survey of general-purposecomputation on graphics hardware,[Comput.Graph. Forum, vol. 26, no. 1, pp. 80–113,2007.
[2] D. Blythe, BThe Direct3 Dimensional 10 system,[ACM Trans. Graph., vol. 25, no. 3,pp. 724–734, Aug. 2006
[3] M. Harris, BMapping computational conceptsto G.P.Us,[in G.P.U Gems 2, M. Pharr, Ed.Reading, MA: Addison-Wesley, Mar. 2005,pp. 493–508.
[4] Buck, T. Foley, D. Horn, J. Sugerman,K. Fatahalian, M. Houston, and P. Hanrahan,BBrook for G.P.Us: Stream computing on graphics hardware,[ACM Trans. Graph.,vol. 23, no. 3, pp. 777–786, Aug. 2004.
[5] M. McCool, S. Du Toit, T. Popa, B. Chan, andK. Moule, BShader algebra,[ACM Trans.Graph., vol. 23, no. 3, pp. 787–795,Aug. 2004.
[6] D. Tarditi, S. Puri, and J. Oglesby,BAccelerator: Using data-parallelism toprogram G.P.Us for general-purpose uses,[inProc. 12th Int. Conf. Architect. SupportProgram. Lang. Oper. Syst., Oct. 2006,pp. 325–335.
[7] M. McCool, BData-parallel programmingon the cell BE and the G.P.U using theRapidMind development platform,[inProc. GSPx Multicore Applicat. Conf.,Oct.–Nov. 2006.
[8] PeakStream, The PeakStream platform:High productivity software development formulti-core processors. [Online]. Available:http://www.peakstreaminc.com/reference/peakstream_platform_technote.pdf
[9] G. Blelloch, Vector Models for Data-ParallelComputing. Cambridge, MA: MIT Press,1990.
[10] M. Harris, S. Sengupta, and J. D. Owens,BParallel prefix sum (scan) with CUDA,[inG.P.U Gems 3, H. Nguyen, Ed. Reading, MA:Addison-Wesley, Aug. 2007, pp. 851–876.
[11] K. E. Batcher, BSorting networks and theirapplications,[in Proc. AFIPS Spring JointComput. Conf., Apr. 1968, vol. 32,pp. 307–314.
[12] N. K. Govindaraju, M. Henson, M. C. Lin, andD. Manocha, BInteractive visibility ordering
[13] http://en.wikipedia.org/wiki/Graphics_processing_unit
[14] http://cs.utsa.edu/~qitian/seminar/Spring11/03_04_11/G.P.U.pdf
[15] Article given by Nvidia company on site http://www.gamesspot.com/articles/the-past-present-and-future-of-the-G.P.U-according-t/1100-6421893/
[16] http://disi.unal.edu.co/~gjhernandezp/HeterParallComp/G.P.U/G.P.U-hist-paper.pdf