Graphics processing unit

A graphics processing unit (GPU), also occasionally called visual processing unit (VPU), is a specialized electronic circuit designed to rapidly manipulate and alter memory to accelerate the creation of images in a frame buffer intended for output to a display. GPUs are used in embedded systems, mobile phones, personal computers, workstations, and game consoles. Modern GPUs are very efficient at manipulating computer graphics, and their highly 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 personal computer, a GPU can be present on a video card, or it can be on the motherboard or—in certain CPUs—on the CPU die.

The term GPU was popularized by Nvidia in 1999, who marketed the GeForce 256 as "the world's first 'GPU', or Graphics Processing Unit, a single-chip processor with integrated transform, lighting, triangle setup/clipping, and rendering engines that are capable of processing a minimum of 10 million polygons per second". Rival ATI Technologies coined the term visual processing unit or VPU with the release of the Radeon 9700 in 2002.

History

1980s

In 1983, Intel made the iSBX 275 Video Graphics Controller Multimodule Board, for industrial systems based on the Multibus standard.[1] The card was based on the 82720 Graphics Display Controller, 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.

Released in 1985, the Commodore Amiga was one of the first personal computers to come standard with a GPU. The GPU supported line draw, area fill, and included a type of stream processor called a blitter 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 personal computer systems instead used 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 first 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 first video cards for IBM PC compatibles to implement fixed-function 2D primitives in electronic hardware.
In 1991, S3 Graphics introduced the S3 86C911, which its designers 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 games within Windows 95 and later.

In the early- and mid-1990s, CPU-assisted real-time 3D graphics were becoming increasingly common in computer and console games, which led to an increasing public demand for hardware-accelerated 3D graphics. Early examples of mass-marketed 3D graphics hardware can be found in fifth generation video game consoles such as PlayStation and Nintendo 64. In the PC world, notable failed first tries for low-cost 3D graphics chips were the S3 VirGE, ATI Rage, and Matrox Mystique. These chips were essentially previous-generation 2D accelerators with 3D features bolted on. Many were even pin-compatible with the earlier-generation chips for ease of implementation and minimal cost. Initially, performance 3D graphics were possible only with discrete boards dedicated to accelerating 3D functions (and lacking 2D GUI acceleration entirely) such as the 3dfx Voodoo. However, as manufacturing technology continued to progress, video, 2D GUI acceleration, and 3D functionality were all integrated into one chip. Rendition’s Verite chipsets were the first to do this well enough to be worthy of note.

OpenGL appeared in the early ’90s as a professional graphics API, but originally suffered from performance issues which allowed the Glide API to step in and become a dominant force on the PC in the late ’90s. 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 game 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 GPUs 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 Direct3D.)

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. Direct3D 5.0 was the first 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. Direct3D 7.0 introduced support for hardware-accelerated transform and lighting (T&L) for Direct3D, while OpenGL had this
capability already exposed from its inception. 3D accelerator cards moved beyond being just simple rasterizers to add another significant hardware stage to the 3D rendering pipeline. The Nvidia *GeForce 256* (also known as NV10) was the first consumer-level card on the market with hardware-accelerated T&L, while professional 3D 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 to 2005

With the advent of the OpenGL API and similar functionality in DirectX, GPUs added programmable 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 was first 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 first Direct3D 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 used 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, and then new generic stream processing unit GPUs became a more generalized computing device. Today, parallel GPUs have begun making computational inroads against the CPU, and a subfield of research, dubbed GPU Computing or GPGPU for *General Purpose Computing on GPU*, has found its way into fields as diverse as machine learning, oil exploration, scientific image processing, linear algebra,\[^3\] statistics,\[^4\] 3D reconstruction and even stock options pricing determination. Nvidia's CUDA platform was the earliest widely adopted programming model for GPU computing. More recently OpenCL has become broadly supported. OpenCL is an open standard defined by the Khronos Group which allows for the development of code for both GPUs and CPUs with an emphasis on portability.\[^5\] OpenCL solutions are supported by Intel, AMD, Nvidia, and ARM, and according to a recent report by Evan's data OpenCL is the GPGPU development platform most widely used by developers in both the US and Asia Pacific.

### GPU companies

Many companies have produced GPUs under a number of brand names. In 2009, Intel, Nvidia 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 integrated graphics solutions as GPUs. Not counting those numbers, Nvidia and ATI control nearly 100% of the market as of 2008.\[^6\] In addition, S3 Graphics (owned by VIA Technologies) and Matrox produce GPUs.
Graphics processing unit

**Computational functions**

Modern GPUs use most of their transistors to do calculations related to 3D computer graphics. They were initially used to accelerate the memory-intensive work of texture mapping and rendering polygons, later adding units to accelerate geometric calculations such as the rotation and translation of vertices into different coordinate systems. Recent developments in GPUs include support for programmable shaders which can manipulate vertices and textures with many of the same operations supported by CPUs, oversampling and interpolation techniques to reduce aliasing, and very high-precision color spaces. Because most of these computations involve matrix and vector operations, engineers and scientists have increasingly studied the use of GPUs for non-graphical calculations. An example of GPUs being used non-graphically is the generation of Bitcoins, where the graphical processing unit is used to solve hash functions.

In addition to the 3D hardware, today's GPUs include basic 2D acceleration and framebuffer capabilities (usually with a VGA compatibility mode). Newer cards like AMD/ATI HD5000-HD7000 even lack 2D acceleration; it has to be emulated by 3D hardware.

**GPU accelerated video decoding**

Most GPUs made since 1995 support the YUV color space and hardware overlays, important for digital video playback, and many GPUs made since 2000 also support MPEG primitives such as motion compensation and iDCT. This process of hardware accelerated video decoding, where portions of the video decoding process and video post-processing are offloaded to the GPU hardware, is commonly referred to as "GPU accelerated video decoding", "GPU assisted video decoding", "GPU hardware accelerated video decoding" or "GPU hardware assisted video decoding".

More recent graphics cards even decode high-definition video on the card, offloading the central processing unit. The most common APIs for GPU accelerated video decoding are DxVA for Microsoft Windows operating system and VDPAU, VA-API, XvMC, and XvBA for Linux and UNIX-based operating systems. All except XvMC are capable of decoding videos encoded with MPEG-1, MPEG-2, MPEG-4 ASP (MPEG-4 Part 2), MPEG-4 AVC (H.264 / DivX 6), VC-1, WMV3/WMV9, Xvid / OpenDivX (DivX 4), and DivX 5 codecs, while XvMC is only capable of decoding MPEG-1 and MPEG-2.

**Video decoding processes that can be accelerated**

The video decoding processes that can be accelerated by today's modern GPU hardware are:

- Motion compensation (mocomp)
- Inverse discrete cosine transform (iDCT)
  - Inverse telecine 3:2 and 2:2 pull-down correction
  - Inverse modified discrete cosine transform (iMDCT)
- In-loop deblocking filter
- Intra-frame prediction
- Inverse quantization (IQ)
- Variable-length decoding (VLD), more commonly known as slice-level acceleration
- Spatial-temporal deinterlacing and automatic interlace/progressive source detection
- Bitstream processing (Context-adaptive variable-length coding/Context-adaptive binary arithmetic coding) and perfect pixel positioning.
GPU forms

Dedicated graphics cards

The GPUs 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 used only when a PCIe or AGP slot is not available.

A dedicated GPU is not necessarily removable, nor does it necessarily interface with the motherboard in a standard 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 GPUs are removable. Dedicated GPUs for portable computers are most commonly interfaced through a non-standard 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 and CrossFire by ATI allow multiple GPUs to be used to draw a single image, increasing the processing power available for graphics.

Integrated graphics solutions

Integrated graphics solutions, shared graphics solutions, or Integrated graphics processors (IGP) utilize a portion of a computer's system RAM rather than dedicated graphics memory. Most are integrated into the motherboard, though exceptions include AMD's IGPs that use dedicated sideport memory on certain motherboards, and APUs, where they are integrated with the CPU die. Computers with integrated graphics account for 90% of all PC shipments. These solutions are less costly to implement than dedicated graphics solutions, but tend to be less capable. Historically, integrated solutions were often considered unfit to play 3D games 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. However, modern integrated graphics processors such as AMD Accelerated Processing Unit and Intel HD Graphics are more than capable of handling 2D graphics or low stress 3D graphics, having improved performance that they can match or exceed that of older dedicated graphic cards, although they are still far less capable than current generation dedicated GPUs.

As a GPU is extremely memory intensive, an integrated 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. Wikipedia:Please clarify Older integrated graphics chipsets lacked hardware transform and lighting, but newer ones include it. Wikipedia:Please clarify
Hybrid solutions

This newer class of GPUs competes with integrated graphics in the low-end desktop and notebook markets. The most common implementations of this are ATI's HyperMemory and Nvidia's TurboCache. Hybrid graphics cards are somewhat more expensive than integrated 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.

Stream Processing and General Purpose GPUs (GPGPU)

It is becoming increasingly common to use a general purpose graphics processing unit as a modified 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 certain 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) GPU designers, ATI and Nvidia, are beginning to pursue this approach with an array of applications. Both Nvidia and ATI have teamed with Stanford University to create a GPU-based client for the Folding@home distributed computing project, for protein folding calculations. In certain circumstances the GPU calculates forty times faster than the conventional CPUs traditionally used by such applications.

GPGPU can be used 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 GPU.

Furthermore, GPU-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 GPU acceleration.

NVIDIA cards support API extensions to the C programming language such as CUDA ("Compute Unified Device Architecture") and OpenCL. CUDA is specifically for NVIDIA GPUs whilst OpenCL is designed to work across a multitude of architectures including GPU, CPU and DSP (using vendor specific SDKs). These technologies allow specified functions (kernels) from a normal C program to run on the GPU's stream processors. This makes C programs capable of taking advantage of a GPU's ability to operate on large matrices in parallel, while still making use of the CPU when appropriate. CUDA is also the first API to allow CPU-based applications to directly access the resources of a GPU 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 GPUs 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 GPU 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 GPU’s SIMD architecture. However, substantial acceleration can also be obtained by not compiling the programs, and instead transferring them to the GPU, to be interpreted there.\cite{8} Acceleration can then be obtained by either interpreting multiple programs simultaneously, simultaneously running multiple example problems, or combinations of both. A modern GPU (e.g. 8800 GTX or later) can readily simultaneously interpret hundreds of thousands of very small programs.
References

[2] 3dfx Glide API

External links

- - a Graphics Hardware History (http://titancity.com/articles/gfxcards.html)
- General-Purpose Computation Using Graphics Hardware (http://www.gpgpu.org/)
- GPU Caps Viewer - Video card information utility (http://www.ozone3d.net/gpu_caps_viewer/)
- ARM Mali GPUs Overview (http://malideveloper.arm.com/learn-about-mali/about-mali/arm-mali-gpus/)