

# Game Consoles: A Look Ahead

By Brian Neal – December 2003

## Introduction

With the emergence of Microsoft and its Xbox, the console gaming market has become increasingly competitive, especially with Microsoft and Nintendo both angling for the No. 2 spot in the market and a shot for the top. While the present battle rages, though, the stage is being set for the next-generation of gaming machines. All three competitors have formed alliances with various partners and although Sony's Playstation 3 and the "CELL" project have been the subject of much discussion and countless rumors, information about the other future consoles has only begun to trickle out relatively recently. In this article, we'll take a look at the current generation as well as what's been revealed thus far for all three upcoming gaming machines, in an effort provide some speculation as to what we think the future may hold.

## The Status Quo

Before peering into the future, let's take a look at the current situation and the three major gaming consoles: the Sony Playstation 2, Nintendo Gamecube, and Microsoft Xbox.

## Sony Playstation 2

The Playstation 2 was introduced in 2000 as the successor to the well-received and popular Playstation. Under the hood of the PS2 is a 300 MHz Toshiba Emotion Engine and 150 MHz Sony Graphics Synthesizer. The Emotion Engine tackles the roles of both a general purpose CPU and vector processor/vertex shader for the GPU. The main CPU is a MIPS R5900 core -- an in-order 2-issue superscalar processor implementing the MIPS IV instruction set. It includes a 16 KB 2-way set-associative I-CACHE and an 8 KB 2-way set-associative D-CACHE, along with a 16 KB data scratchpad RAM. It also features an FPU and a set of 128-bit multimedia extensions akin to MMX on the x86 platform.

On the same piece of silicon as the R5900 there are two 128-bit VLIW vector units capable of functioning as independent processors. The first vector unit, VU0, can optionally function independently (micro mode) or as a co-processor to the MIPS CPU core (macro mode). It features 4 FMACs (Floating-Point Multiply Accumulate) and 1 FDIV (Floating-Point Divide). The VU register file consists of 32 128-bit floating-point registers that can be divided into four 32-bit single-precision fields. Each FMAC unit in the VU, FMACx, FMACy, FMACz, FMACw, can operate on a single-precision floating-point value in parallel with a latency of 4 cycles and a throughput of 1 operation per cycle.

The second vector unit, VU1, is similar to VU0, but is more specialized in that it is an independent unit targeted at geometric transformation for the Graphics Synthesizer (GS). To this end, VU1 connects to the GIF, a 150 MHz 64-bit interface to the GS. The GIF handles the transfer of display lists and textures to the GS.



*Virtua Fighter 4: Evolution (PS2)*

The VU1 core also features additional functional units, for a total of 5 FMACs and 2 FDIVs, and a larger integrated RAM (16 KB microcode RAM and 16 KB working RAM versus 4 KB microcode RAM and 4 KB working RAM for VU0).

Between the CPU, FPU, VU0, and VU1, the Emotion Engine is capable of a theoretical 21 floating-point operations per clock-cycle for a total of 6.2 GFLOPS. Being in-order and lacking a L2 cache (or a larger L1 cache), the performance of the R5900 CPU core can suffer heavily from instruction and particularly data-cache misses, while VU0 micro-mode is generally under-utilized. However, in discussing the PS2 with developers who work on the platform, we have learned that 60% of this theoretical 6.2 GFLOPS is quite achievable in sustained performance.



*Playstation 2 System Board - Graphics Synthesizer (labeled), Emotion Engine (below/left of GS)  
Courtesy [this article](#)*

One of the key differences between the Playstation 2 and the Xbox (as well as most PC-based GPUs) is where geometry transformation occurs. In the case of the PS2, geometry is transformed by the Emotion Engine's VU1 and sent to the Graphics Synthesizer. In the case of the Xbox, this job is handled by the NV2A's vertex shaders. For the Gamecube, fixed transformations and lighting occur on the GPU as well, though the CPU is required to handle dynamic tasks.

Also found on-board the Emotion Engine is a 10-channel programmable DMA controller, a dual-channel Rambus DRAM memory interface, and an image data processor with support for JPEG/MPEG decoding/quantization. The memory subsystem consists of two 16-bit RDRAM channels, each of which connects to a 16 MB PC800 RDRAM device for a total of 32 MB of main memory at 3.2 GB/s.

The PS2 also implements an input/output processor, or IOP, that is essentially a Playstation on a chip. This provides backwards compatibility with existing PS1 games.

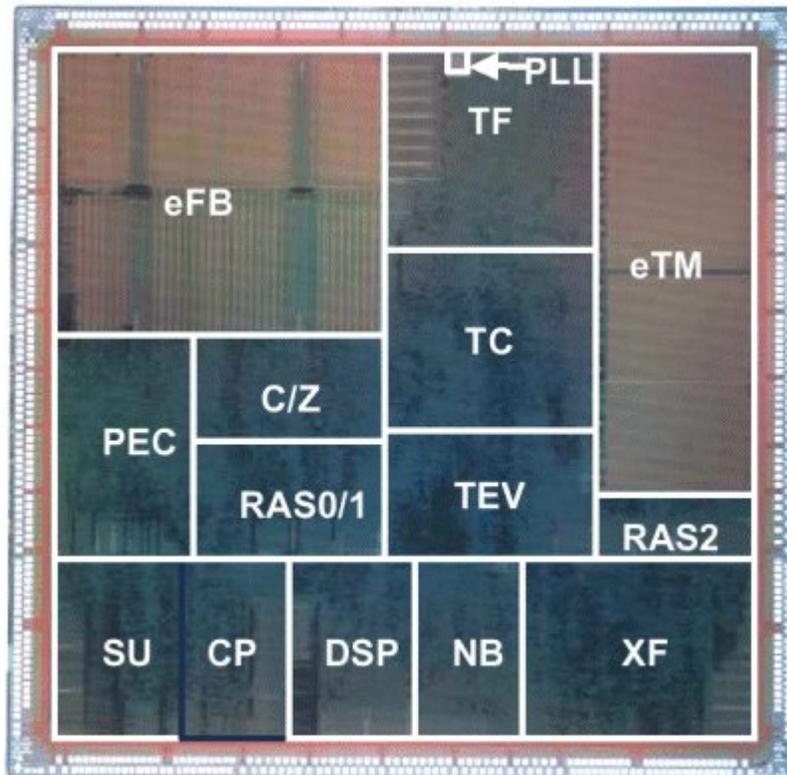
## Graphics Synthesizer

The Graphics Synthesizer serves as the PS2's GPU or rasterizer. At 150 MHz, the GS possesses a formidable fillrate with 16 pixel pipelines, arranged in a 4 x 4 configuration. This gives the GS an untextured fillrate of 2.4 GPixel/s and, with its 8 texture units, a textured fillrate of 1.2 GPixel/s. Backing up this fillrate is 4 MB of eDRAM connected to a 2560-bit bus (1024-bit read, 1024-bit write, and 512-bit texture) with a peak bandwidth of 48 GB/s. The GS lacks multitexturing and therefore relies on its massive fillrate for multipass rendering. This also means that the Emotion Engine has to resend the transformed geometry for each pass. The geometry can be cached to avoid the transform overhead, though bandwidth is still consumed on the GIF. Vector data compression between the Emotion Engine and the Graphics Synthesizer (on the order of 1:2 or 1:4) is supported along with CLUTs (texture look up tables) to conserve bandwidth.

With its massive bandwidth and fillrate, the GS is capable of handling quite a bit of fillrate-intensive alpha blending for transparencies as well as various full-screen effects. However, due to the lack of more modern features like multitexturing, the GS must spend much of its massive fillrate and bandwidth in multipass rendering for multiple texture layers (color + lightmap, etc.). In the end, it's a tradeoff that is discussed below.

## Embedded vs Discrete DRAM

The Playstation 2 and, as you will see below, the Gamecube, both rely on embedded memories to provide massive bandwidth to their respective GPUs. This is one of the more significant contrasts between these consoles and your typical desktop PC in that nearly all of today's modern GPUs from ATI and NVIDIA use discrete memory. The rather infamous **Bitboys** attempted to produce a high-performance desktop graphics accelerator, but none of their designs ever made it to market. The tradeoff is between bandwidth and space, with on-chip memories enabling very high-bandwidth and low-latencies, but at a cost to overall size.



Gamecube Flipper Die Photo - eFB: Embedded Frame Buffer, eTB: Embedded Texture Buffer  
Photo courtesy a Nintendo *presentation* by Howard Cheng

To contrast the differences, let's consider the high-end GPUs on the market at the time of the Playstation 2's introduction in 2000 or later in 2001, since that is the year of the Gamecube's launch. In the case of the **ATI Radeon** and **NVIDIA GeForce 3**, the cards have at most 64 MB of memory with a bandwidth of 5.85 GB/s (Radeon) to 7.36 GB/s. The Graphics Synthesizer, by comparison offers 48 GB/s of memory bandwidth (6.5 times the GeForce 3, 8.2 times the Radeon), but to only 4 MB of memory (16 times less space).

Obviously, the GS's theoretical memory bandwidth is impressive and remains unmatched by discrete solutions in use by GPUs three years later, but the size of that memory is clearly much smaller than what the PC-based GPUs of the time had available, let alone those on the market today. To understand the design choice, we have to look to the console environment and its applications. All PS2 games have a target NTSC or PAL televisions as the primary display with a resolution of around 640x480. Given this resolution, the frame and Z buffers can be stored entirely on-chip, consuming roughly 2-3 MB, leaving 1 MB free for use as a texture buffer depending upon the arrangement. The following slide from an SCEE presentation by Mark Breugelmanns describes an example VRAM configuration:

- For <60hz frame-rates

Display	$640 \times 448 \times 24 \text{bit} = 860\text{k}$
Draw	$640 \times 448 \times 24 \text{bit} = 860\text{k}$
Z Buffer 0.5meg	$640 \times 448 \times 16 \text{bit} = 573\text{k}$
Texture 1.3meg	Static textures = 1.3meg
Texture cache 0.4meg	Texture cache = $6 * 256 \times 256$ (8bit)

On the other hand, a PC GPU might be rendering to a much higher resolution, typically 1024x768 or 1280x1024. At such resolutions, a 4 MB buffer would be unsuitable without even considering space for a texture buffer. But at TV resolutions, it is more reasonable. The Gamecube's Flipper has a similar configuration with a 2 MB draw and Z buffers and 1 MB for textures, and it targets the same resolution.

In fact, the **Bitboys GPU** briefly mentioned above operated on a similar concept. It featured 12 MB of embedded DRAM for frame and Z buffers exclusively, while textures were to be stored in a separate pool of SDRAM. The idea was for the eDRAM to provide the needed bandwidth for the Z and frame buffers, thereby permitting textures to be stored in slower and less-expensive SDRAM.

In the case of both the Flipper and the Graphics Synthesizer, textures are streamed in and out of VRAM as needed. This is handled on the GS through a 1.2 GB/s (150 MHz 64-bit) interface dubbed the GIF. The Gamecube's Flipper incorporates a 2.6 GB interface to main memory and also supports S3TC texture compression. In fact, having been introduced a year later and on a smaller process (0.18μ instead of 0.25μ), the Flipper has a great deal more logic to support capabilities like multitexturing, texture compression, color combiners, fixed-function geometry transformation, and more.

Veteran console developer and 3D graphics expert Miha Peternel recounts texture streaming on the PS2:

We were targeting 50:50 textures:polygons. At 60 FPS that gives you only 10 megs for textures. Of course the way around is to render two frames in one pass and then you have 20 megs of textures and even better locality. You cannot even imagine the skills of artists and directors required to make quality art that fits the specs.

Clearly, there were some costly tradeoffs for using eDRAM at 0.25μ, but what about 0.09μ or 0.065μ?

## Nintendo Gamecube

Nintendo's successor to the N64 arrived late in 2001 with the Gamecube. For the Gamecube, Nintendo's partners consisted of IBM and ATI, formerly ArtX. IBM provided a 485 MHz **PowerPC 750** CPU codenamed *Gekko* while the 162 MHz ATI/ArtX GPU and I/O chip was dubbed *Flipper*. The CPU features a 64 KB L1 cache (32 KB Instruction, 32 KB Data), a 256 KB 2-way set-associative L2 cache, and vector extensions consisting of roughly 50 new instructions. There have been further modifications beyond the instruction extensions, however, as the FPU has been modified to process two single-precision values per cycle instead of one double-precision value. Correspondingly, each floating-point register can store either one 64-bit value or two 32-bit values. The FPU has a peak throughput of 1.9 GFLOPS with two multiply-add operations per cycle. At 485 MHz, the CPU has a typical power dissipation of 4.9W on an 0.18μ Cu process.



*Metroid Prime (Gamecube)*

The Gekko CPU is actually something of a hybrid 750CXe/750FX processor, boasting features from both the CXe and the FX. Specifically, it has the same cache structure as the CXe and the enhanced 60x bus of the FX. Half of the L1 data cache (16 KB) can be locked as well. Another trait of the FX that can be found in Gekko is a dual-reservation station FPU pipe, allowing for full pipelining in addition to the packed single-precision capability mentioned above.

It also offers instructions that can load 8/16-bit integers into the FP registers, converting them to FP on the fly. According to Gabriele Svelto, a long-time Ace's Hardware reader who has researched the Gamecube and Gekko CPU, this is a significant advantage for gaming applications compared to other non-AltiVec PowerPCs as integer-to-float and float-to-integer conversions can be expensive due to the fact that they incur load-from-store penalties.

The 51 million transistor Flipper serves not only as the Gamecube's GPU, but also as an I/O controller and sound processor. The chip also incorporates 3 MB of on-chip 1T-SRAM. 2 MB is reserved for frame (back) and z-buffers while the other 1 MB is used for texture memory. The on-chip memory provides low-latency access, on the order of 6.2 ns, as well as high bandwidth (10.4 GB/s texture bandwidth, 7.6 GB frame buffer bandwidth). S3TC texture compression is used to minimize the impact of the relatively small 1 MB texture buffer.



*Gamecube System Board - Gekko (top), Flipper (center), Splash (bottom)  
Courtesy [this Hotchips presentation](#) by Peter Sandon*

The GPU itself has 4 pixel pipelines with one TMU each, providing a peak fillrate of 648 MPixel/s. It is capable of multitexturing and can apply as many as 8 layers in a single pass.

The system's memory configuration is unique in that it uses 24 MB of MoSys 1T-SRAM (codenamed Splash) running at 324 MHz for a peak bandwidth of 2.6 GB/s across a 64-bit interface. It also features 16 MB of DRAM on an 8-bit interface running at 81 MHz. This 81 MB/s memory is used for audio, though creative developers may also find other uses like caching program code or data.

## Microsoft Xbox

For starters, the Xbox is based around a 733 MHz Pentium III with a 32 KB L1 cache (16 KB I + 16 KB D) and an 8-way set-associative 128 KB L2 cache. While the smaller cache is more reminiscent of Intel's budget Celeron processor, the Xbox's CPU does feature a 133 MHz FSB (1 GB/s) just like the Coppermine Pentium III. The CPU also incorporates MMX and SSE extensions for integer and floating-point SIMD operations (Single Instruction, Multiple Data). From an architectural standpoint, the Xbox very closely resembles a PC and therefore many of our readers will be quite familiar with its internals. For specific details of the P6 core (Pentium III, Celeron), see our series "The Secrets of High Performance CPUs" parts [one](#) and [two](#). For more information on the basis of the Xbox's NV2A GPU, see [this article](#) and [its follow-up](#).

In terms of memory, the Xbox features four 16MB DDR SDRAM devices (32-bits each), for a total of 64 MB. The memory is produced by Samsung and it runs at 200 MHz for a peak bandwidth of 6.4 GB/s. Unlike the other two systems, the Xbox uses a UMA approach whereby all of a game's program code, sound data, textures, and geometry are all stored in the same 64 MB pool, along with the framebuffer. While this degree of sharing between various components of the system may incur some bottlenecks in terms of bandwidth, it also allows for efficient utilization of all the system's memory without any issues like replication of data between pools that might pressure the system's memory size. Nevertheless, the real-world performance of this memory system is a topic for another article.



*Ninja Gaiden (Xbox)*

The Xbox's NV2A chipset is similar to that of NVIDIA's nForce for the Athlon, but it makes use of a much more powerful GeForce 4 level GPU than the nForce's integrated GeForce 2 MX. Beyond graphics, the NVIDIA chipset also handles the typical functions of the north and south-bridges of a modern PC, such as memory read/write, disk I/O, peripheral support, and so forth. The 233 MHz NV2A has a peak fillrate of slightly under 1 GPixel/s (932 MPixel/s). Each of the four pixel pipelines incorporates two TMUs, allowing the NV2A to apply two texture layers in a single pass with no fillrate hit (though there is a bandwidth hit).



*Xbox System Board - Pentium III/Celeron CPU (far-right, under heatsink), NV2A GPU (left of CPU, small heatsink /w fan)*

Like the GeForce 4, the NV2A supports vertex and pixel shading and features two programmable vertex pipelines. This contrasts to the Flipper's fixed-function T&L pipeline and the Emotion Engine's very flexible vector units in that the NV2A can progress dynamic geometry on the GPU itself.

Essentially a PC in a black and green box, the Xbox carries forth both advantages and disadvantages as a result of its design. By leveraging the PC architecture, along with Windows and DirectX, the Xbox has arguably the most robust development environment and toolset console developers have seen in years. It also proves to be very familiar to veteran PC game developers, easing the porting process for games from those developers. Other advantages include a featureful GPU, a powerful main CPU, and a large (relatively) pool of memory. Its only disadvantage from a technical standpoint might be the lack of specialization in the hardware. It lacks features like the Gamecube's mini-disc media that helps cut down on piracy or the flexibility found in the PS2 hardware. From a physical standpoint, it's also the largest of the three systems, a fact that may have contributed to the Xbox's continued lack of success in the Japanese market.

The system's PC roots are easily understandable given its collective parents, Microsoft, Intel, and NVIDIA. However, there are indications that Microsoft's next-generation system isn't going to be just another "PC in a box." We'll get to that next.

## Looking to the Future

Now that we've taken a look inside today's game consoles, we'll look to the next generation of systems. As a caution, there's very little concrete information about any of the next generation systems at this time, and so this section is highly speculative.

## Xbox Next / Xbox 2

At this point in time, there's very little solid information regarding Microsoft's next-generation console system. However, we do know who's involved in making it, and this time it's IBM, ATI, and SiS. Last month's announcement of IBM as a partner came as quite a surprise since it was generally expected that the next Xbox would use an x86 processor -- most likely from Intel or AMD. It's still possible that Microsoft could license an x86 core that would then be manufactured by IBM, but statements in the **press release** seem to indicate IBM will be responsible for the technology as well:

According to Bernie Meyerson, IBM Fellow and chief technologist for IBM's Technology Group, the new Xbox technologies will be based on the latest in IBM's family of state-of-the-art processors.

"IBM's advanced chip technologies are in demand across a wide range of industries and applications," Meyerson said. "We're excited to be working on a project of this magnitude and that Microsoft has chosen IBM to provide technologies that will power future consumer devices and expand the boundaries of what's possible in entertainment."

Could the next Xbox have a PowerPC inside? It's been heavily speculated that **this may be the case**, but what would be the implications of such a move? Until the IBM announcement, backwards compatibility has almost been taken for granted with an Xbox successor, but a shift in instruction sets challenges that. Microsoft may still be able to provide compatibility through emulation, though, perhaps through technology acquired from Connectix Virtual PC.

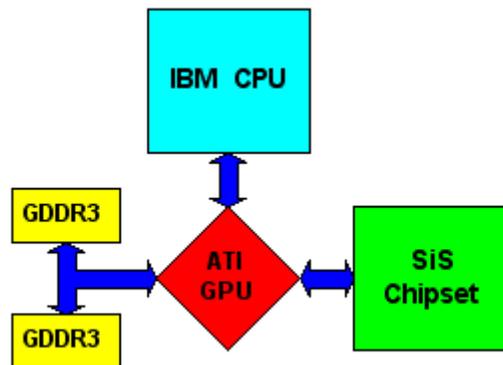
While most of the speculation has been about a mainstream PowerPC derivative, like the **PowerPC 970** (aka Apple G5), the **Sony-Toshiba-IBM CELL processor** also comes to mind. Could IBM supply Microsoft with a CELL variant for the next-generation Xbox? If so, it would be quite a coup, given the hype and \$400 million investment from the three companies in this project.

## Exit NVIDIA, Enter ATI

While the IBM agreement came as a surprise, the **announcement** of ATI as Microsoft's graphics partner was not. Last year, Microsoft and NVIDIA had a highly publicized **dispute** over the pricing of the Xbox's NVIDIA supplied NV2A GPU and system chipset. Statements made by NVIDIA CEO Jen-Hsun Huang gave the impression that NVIDIA was feeling **lukewarm** about involvement in a second Xbox. It's speculated that NVIDIA's Xbox development efforts caused the company to falter in its core PC graphics market.

Conversely, ATI has been on something of a surge since the introduction of the **R300 / Radeon 9700** core in 2002. The company has also found itself in the console market with the acquisition of ArtX (the company behind the Gamecube's Flipper) and **wanting more**.

The next Xbox's GPU is being developed at ATI's east coast design center and it is expected that the chip will be a derivative of the company's R500 core. The R500 will target DirectX 10, the next major version of the graphics API that is expected to ship with Longhorn.



*A Speculative Diagram of the Next Xbox*

## SiS Inside Xbox Next

The third partnership for the next Xbox is with Silicon Integrated Systems (SiS), as **announced** in November. The chipset maker has a deal with Microsoft to provide "media I/O technologies" for an Xbox successor. From this, we speculate that SiS will provide Microsoft with core logic for DVD, hard disk, and ethernet interfaces. The company could also be involved in the development of an audio chipset, but the press release is too vague to confirm SiS' exact role. A possible configuration for the complete system might involve an ATI GPU with memory interface, an IBM CPU, and a SiS southbridge.

## Playstation 3

The news of an **alliance** between Sony, Toshiba, and IBM arrived in 2001, shortly after the introduction of the Playstation 2 in North America. With investments of over \$400 million, three companies, and 300 engineers, the goal is an ambitious one. Codenamed "Cell," the Broadband Engine is a highly parallel design consisting of a number of Processor Elements (PE). Each PE includes a number of Attached Processing Units (APUs), and within each APU there are a variety of integer and floating-point units. According to the **patent**, a "preferred embodiment" of the APU features 4 integer units, 4 floating-point units, 128 KB "Local Storage" memory, a direct memory access controller (DMAC), and 128 registers (128-bit wide). The PE is described in the patent as follows:

[0013] The basic processing module is a processor element (PE). A PE preferably comprises a processing unit (PU), a direct memory access controller (DMAC) and a plurality of attached processing units (APUs). In a preferred embodiment, a PE comprises eight APUs. The PU and the APUs interact with a shared dynamic random access memory (DRAM) preferably having a cross-bar architecture. The PU schedules and orchestrates the processing of data and applications by the APUs. The APUs perform this processing in a parallel and independent manner. The DMAC controls accesses by the PU and the APUs to the data and applications stored in the shared DRAM.

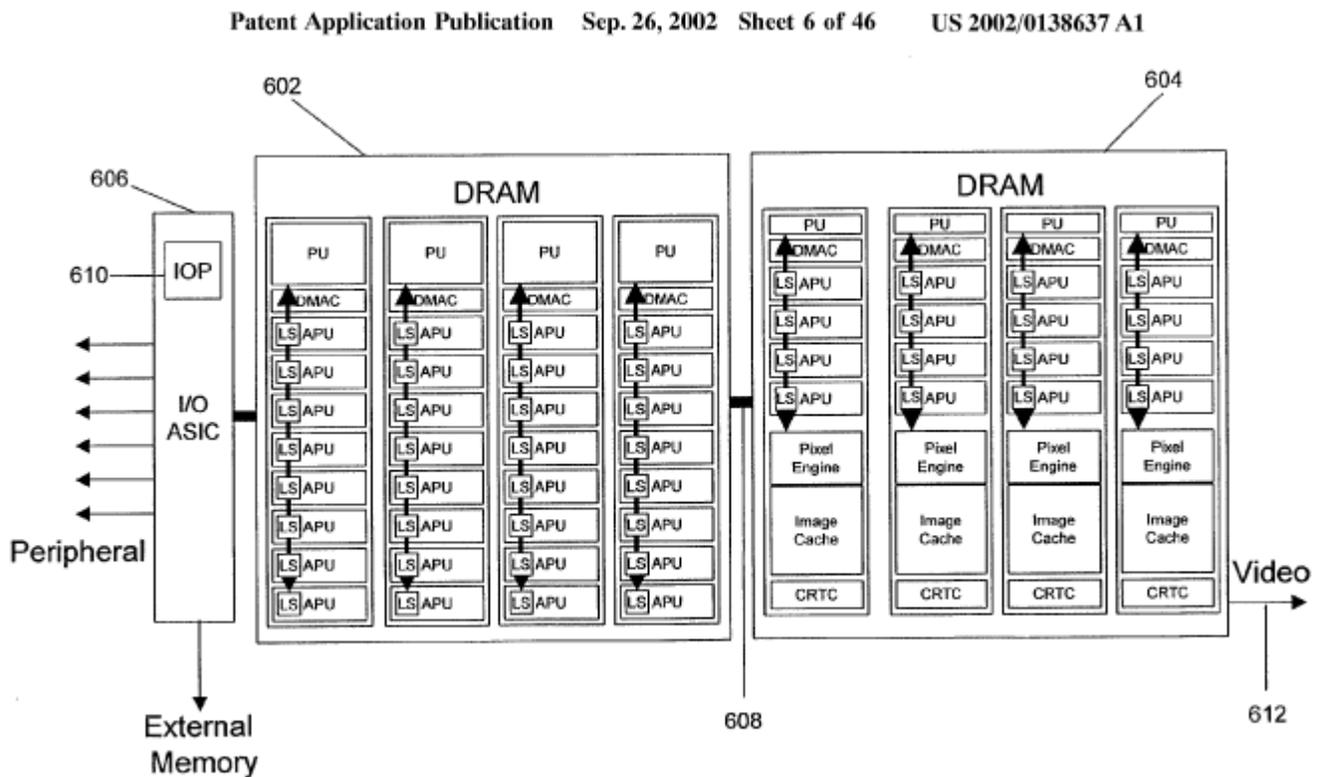


Figure 6 from **US Patent 2002/0138637** - Suzuki, Masakazu and Yamazaki, Takeshi  
*Broadband Engine and Visualizer, both leveraging "CELL"*

The diagram above from the patent demonstrates a possible system configuration with a Broadband Engine (left) and Visualizer (right). The Broadband Engine shown here includes 4 Processor Elements, each with 8 APUs, for a theoretical total of 128 floating-point units. To the right is the Visualizer, essentially a GPU with half of a Broadband Engine. The Visualizer incorporates 4 PEs, much like the Broadband Engine, but each element has half as many APUs. The other half of each element is replaced by a Pixel Engine, Image Cache, and CRT Controller (CRTC).

This arrangement bears some resemblance to the current Playstation 2's Emotion Engine and Graphics Synthesizer in terms of general functionality, with the exception that the GPU presented here features a great deal of logic (16 PEs - 64 FPU) presumably to handle vertex and pixel operations.

One interesting element of this diagram is the reference both to DRAM within the processor block diagrams as well as a link to external memory. This seems to indicate that the chips may use embedded DRAM in addition to high-speed discrete memory. In 2001, Sony and Toshiba announced a joint R&D effort for a **65nm eDRAM process** and a **recent announcement** indicates the two companies will begin trial production of 65nm parts in March 2004, with commercial production commencing sometime in Q2 or Q3 of 2005. The patent describes the various Processing Elements ideally sharing a 64 MB DRAM:

[0015] In a preferred embodiment, a plurality of PEs are associated with a shared DRAM. The DRAM preferably is segregated into a plurality of sections, and each of these sections is segregated into a plurality of memory banks. In a particularly preferred embodiment, the DRAM comprises sixty-four memory banks, and each bank has one megabyte of storage capacity. Each section of the DRAM preferably is controlled by a bank controller, and each DMAC of a PE preferably accesses each bank controller. The DMAC of each PE in this embodiment, therefore, can access any portion of the shared DRAM.

As to the issue of external memory, a **license agreement** was announced between Rambus, Sony, and Toshiba in January of this year for Rambus' high-speed **XDR DRAM** (codenamed "Yellowstone") and the Redwood parallel chip-to-chip interface. XDR DRAM supports datarates ranging from 3.2 GHz to 6.4 GHz with support for 64 or 128-bit interfaces supplying between 25 GB/s and 100 GB/s depending upon the implementation.

As a chip-to-chip interconnect, it follows that Redwood could be utilized between the Broadband Engine and the Visualizer and/or between the Broadband Engine and the I/O ASIC. Below, Figure 8 from the patent describes a single-chip version of the above system:

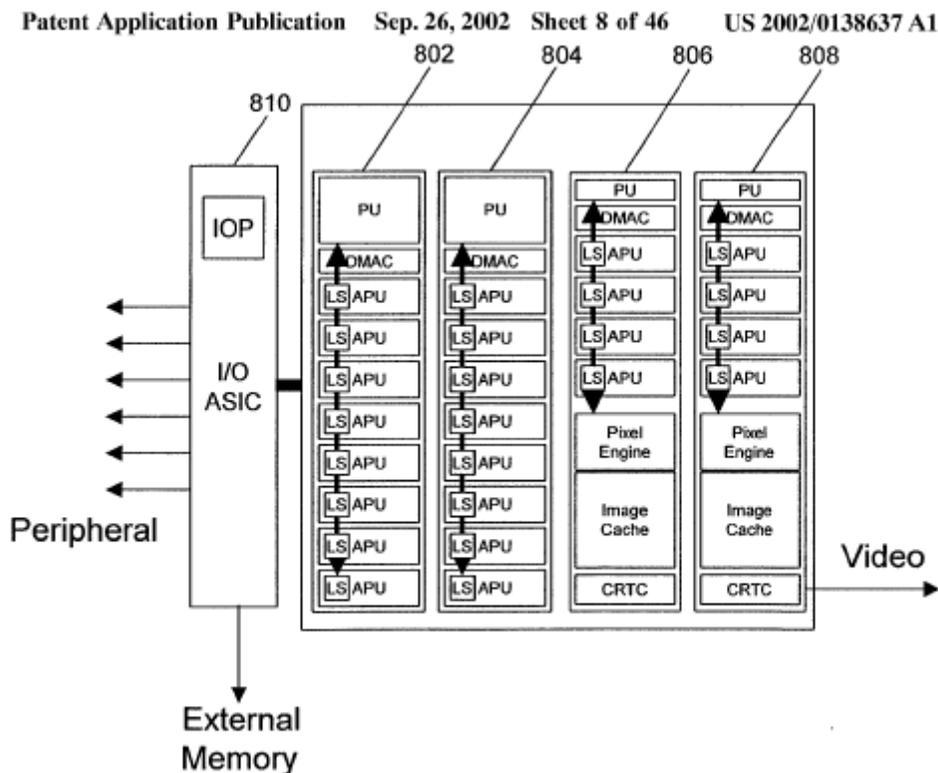


Figure 8 from *US Patent 2002/0138637* - Suzuki, Masakazu and Yamazaki, Takeshi  
Single Chip Broadband Engine and Visualizer

Here we see a combined Broadband Engine and Visualizer using half the elements from each. There are two "full" Processing Elements and another two for handling graphics operations. This would seem to indicate that Sony can pursue either a multi-chip or single-chip solution depending upon process geometries, costs, or performance targets.

## Gamecube 2

While announcements regarding the other two next-generation systems have been few and far between, solid information on Nintendo's fifth home game console is even harder to come by. Beyond statements made by the company that it is indeed working on a new system, a **technology agreement** was announced between Nintendo and ATI regarding the development of technologies for use in Nintendo products. The announcement does not specifically indicate any involvement in a future home console, though it's very likely given ATI's interest in supplying graphics chipsets for **both Nintendo and Microsoft**. Additionally, through the acquisition of ArtX, both companies have a long relationship that extends as far back as the Nintendo 64.

On the CPU front, there's even less to go on, but if IBM were to again supply Nintendo with a processor, Big Blue would have its hand in three different console kettles.

## Final Thoughts

From the perspective of a technology enthusiast, the development of a game console is a very interesting process. Given their specialized application (i.e. gaming), their architectures present us with some unique solutions to specific problems that might not otherwise be found in more general purpose systems. Unlike the PC market, backwards compatibility is not such a stringent requirement that it must be adhered to at all costs. Resultantly, it's not at all uncommon to see vastly different architectures from one generation to the next and from one company to another, as we have demonstrated in this article. We'll have more details on these next-generation systems as the information becomes available and perhaps we'll also delve deeper into current systems in a future installment. Until then, I hope this article has piqued your interest.

## Special Thanks

Special thanks to Miha Peternel and Gabriele Svelto for providing their valuable insight to help make this article possible.