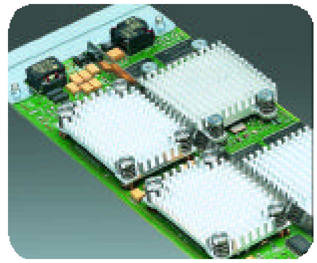
hp IA-32 visualize $fx^5$ and $fx^{10}$ Windows graphics accelerators



white paper

## Leadership Graphics Technology

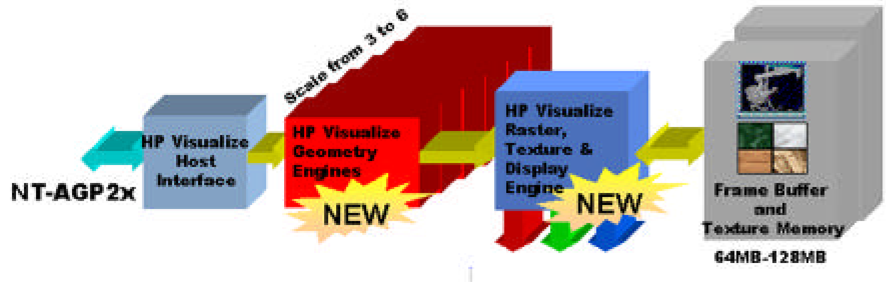
HP set the standard for Windows NT® 3D graphics performance and functionality with the introduction of the **hp** visualize $fx^4$ graphics accelerator in 1997. The subsequent introduction of the **hp** visualize $fx^+$ family for Windows workstations provided further gains in performance at a wide range of price points.

Now, HP continues to evolve its **hp** visualize family of graphics adapters with the introduction of the **hp** visualize $fx^5$ and **hp** visualize $fx^{10}$ graphics accelerators. The **hp** $fx^5$ and $fx^{10}$ feature industry-leading application performance, with a feature set typically found on much more expensive high-end graphics workstations. The hardware is designed for Windows NT® and Windows 2000 Professional®, providing full support for OpenGL 1.1, GDI, DirectDraw, and Direct3D.

Significant features of the **hp** $fx^5$ and $fx^{10}$:

- 3D and depth texture mapping for volume visualization and real-time shadows.
- Texture maps up to 16384 × 16384 in size.
- A hardware-accelerated accumulation buffer for full scene antialiasing and motion blur effects.
- Parallel visibility testing of bounding boxes for fast occlusion culling.
- Support for multiple display syncs for "cave" and "cove" displays.
- Up to 64MB of fully configurable shared framebuffer/texture memory. The flexible shared memory design allows the user to balance texture map storage requirements with pixel depth and desktop size.
- Identical software interfaces and device drivers for both the **hp** $fx^5$ and $fx^{10}$ graphics accelerators to reduce ISV certification expenses.
- Designed with the entire computer system in mind to maximize high-end 3D application performance.

# A Detailed Look at the Architecture and Features




| | **hp** $fx^{5}$ | **hp** $fx^{10}$ |
|---|---|---|
| **Host interface chip** | 1 | 1 |
| **Geometry engines** | 3 | 6 |
| **Rasterizer/texture/display chip** | 1 | 1 |
| **Shared framebuffer/texture memory** | 64MB SDR | 64MB SDR |
| **Software interface/device driver** | Identical for both devices | |

**Architectural Summary**

Both the **hp** $fx^{5}$ and **hp** $fx^{10}$ graphics accelerators contain sufficient frame-buffer memory to support an identical list of pixel formats. The **hp** $fx^{10}$ provides twice the geometry performance of the **hp** $fx^{5}$.

A detailed look at the individual components of the **hp** $fx^{5}$ and $fx^{10}$ follows.

## Host Interface Chip

Communication between the host computer system and the graphics device is via a host interface chip residing on the **hp** $fx^{5}$ and $fx^{10}$.

In order to operate at peak performance levels, the **hp** $fx^{5}$ and $fx^{10}$ support 133MHz AGP 2X DMA (Direct Memory Access) to transfer geometric, pixel, and texture data from the application to the graphics device. Unlike other data transfer methods, DMA is the only method which utilizes 100% of the available AGP bus cycles.

Using AGP 2X DMA, data is transferred to the graphics device at 400MB/sec. Assuming an average triangle size of 29 bytes[1], this is sufficient bandwidth to transmit over 14 million triangles/sec to the graphics device.

AGP 2X DMA is in perfect balance with the hp visualize personal workstation. The state of the art 133MHz front side bus provides a gross bandwidth of 1.06GB/sec. Since real applications simultaneously access their own internal data structures while generating graphics data, the maximum sustainable geometry data bandwidth is only half of that, or 530MB/sec. Inherent front side bus latencies and application overhead further limit the net geometry bandwidth to 400MB/sec or less. Any excess bandwidth to the graphics device would be wasted, and any less would create a bottleneck.

The **hp** $fx^{5}$ and $fx^{10}$ graphics accelerators are designed to provide peak performance for real applications, which can not afford to sit idle while data is being transmitted to the graphics device. Once the device driver initiates a DMA data transfer, the host interface chip retrieves the data from main memory asynchronously, freeing the host CPU for other tasks. Compared to PIO (Programmed I/O) and AGP 4X with Fast Writes, this results in vastly improved application performance.

*[1] A triangle strip primitive containing 10 triangles will have four bytes of overhead, followed by 12 24-byte vertices, for a total of 292 bytes. This is an average of 29 bytes/triangle.*

When comparing AGP 2X DMA to AGP 4X with Fast Writes, keep the following in mind:

- AGP 4X with Fast Writes occupies the CPU while writing data to the graphics device, preventing your application from doing other useful work. AGP 2X DMA frees the host CPU while writing data to the graphics device, resulting in superior application performance.
- Claims that 900MB/sec bandwidth is required for today's graphics applications is simply wrong. In fact, 400MB/sec is sufficient to transmit over 14 million triangles/sec to the graphics device. Furthermore, a 900MB/sec data transfer rate is unattainable by real applications on systems with a 133MHz front side bus. AGP 4X with Fast Writes does nothing to alleviate this bottleneck.
- AGP 2X DMA is a reliable data transfer mechanism that is known to produce excellent application performance. AGP 4X with Fast Writes provides no performance benefit over AGP 2X DMA, and its complexity compromises system reliability.

An additional DMA engine in the host interface chip uses 66MHz PCI protocol to transfer data from the graphics device to main memory. This is especially useful for reading the contents of the framebuffer, a critical operation for many Digital Content Creation, Video Editing, and Visualization applications.

The host interface chip also supports fast hardware state switching for acceleration of multiple concurrent rendering applications. Applications that use multiple OpenGL rendering contexts will also benefit from this feature. An application that caches state for different rendering scenarios in multiple OpenGL contexts will be able to rapidly switch between them.

## Geometry Engines

The geometry engines perform geometric transformations, lighting, model clipping, and other vertex operations on incoming geometric data. This frees the host CPU, leaving more processing power available for application work.

The geometry engines use floating point units based on **hp** PA-RISC processor technology to achieve maximum floating-point performance.

There are three geometry engines per geometry accelerator chip. The **hp** *fx*$^5$ has a single geometry accelerator chip containing three full geometry engines, while the **hp** *fx*$^{10}$ has two chips for a total of six full geometry engines.

Each hardware geometry engine supports a rich geometry feature set, including:

- Lighting and shading for up to eight separate OpenGL light sources
- All OpenGL primitive types
- Transformations
- View volume and model space clipping
- Material properties for accelerated rendering of lit surfaces
- Texture coordinate generation, useful in Scientific Visualization applications
- Environment mapping for fast realistic surface reflections
- Texture coordinate generation, useful in Scientific Visualization applications
- Environment mapping for fast realistic surface reflections
- Second generation hardware occlusion culling implementing faster rejection of invisible geometry based on its bounding volume.

## *Raster, Texture, and Display Engine Chip*

A single chip combines the raster, texture and display processors, maximizing the performance and efficiency of these operations.

The Raster Processor

The raster processor converts incoming geometric primitives into pixel data for storage in the framebuffer. It supports all RGBA OpenGL and Direct3D per-pixel operations, including:

- Depth, alpha, and stencil tests for hidden line and hidden surface removal (HLR/HSR), billboarding, Composite Solid Geometry (CSG) and other effects
- Linear and exponential fog for depth cueing and atmospheric effects
- Blending and logical operations for transparency effects and image processing applications
- Antialiased lines and points

Antialiased polygons are supported through hardware accelerated full scene antialiasing.

*The Texture Processor*

The texture processor contains built-in support for OpenGL 1.1 and Direct3D texture mapping. In addition to standard features such as mipmapping and bilinear and trilinear filtering, the following features are supported:

- 3D texture maps, for volume visualization
- Depth textures, for creating shadows
- Single pass multi-texturing, up to two textures per primitive
- 16, 12, and 8 bit indexed textures
- Pre-specular texture lighting for better realism

Since textures are stored in the same block of memory as the framebuffer, desktop size, pixel format, and texture format determine the maximum texture size. The texture processor supports texture sizes of up to 16384 x 16384.

*The Display Processor*

The video display processor combines the contents of the framebuffer to produce a displayed image. Features provided by the Video Display Processor include:

- Three color Look Up Tables (LUTs), allowing individual windows to maintain their own set of colors
- Gamma correction of 3D windows
- 8 bit/pixel or 16 bit/pixel double buffered or blended video overlay
- Synchronized stereo display, with support for industry standard stereo glasses or head mounted displays
- A hardware accelerated asynchronous mouse cursor for improved system responsiveness
- Intelligent buffer swaps synchronized to the display refresh rate (may be disabled to achieve maximum performance at the expense of image quality)
- Analog (DB15) and digital (DVI) video output connections
- Support for multiple syncs for "cave" and "cove" displays

| Resolution | 65536 Colors (16-bit) Double Buffered | True Color (32-bit) Double Buffered | Stereo Double Buffered | Refresh Rate (Hz) |
|---|---|---|---|---|
| 640 × 480 | ✔ | ✔ | ✔ | 60, 85, 120 |
| 800 × 600 | ✔ | ✔ | ✔ | 60, 85, 120 |
| 1024 × 768 | ✔ | ✔ | ✔ | 60, 85, 120 |
| 1280 × 1024 | ✔ | ✔ | ✔ | 60, 75, 85 |
| 1600 × 1200 | ✔ | ✔ | | 60, 75 |
| 1920 × 1080 | ✔ | ✔ | | 72 |
| 1920 × 1080 | ✔ | ✔ | | 84 |
| 1920 × 1200 | ✔ | ✔ | | 66 |
| 1920 × 1200 | ✔ | | | 76 |

**Supported Display Configurations**

In addition to the video formats listed above, the **hp** *fx*$^5$ and *fx*$^{10}$ support user-defined video formats and timings via the display properties dialog box.

## *Memory Architecture*

The **hp** *fx*$^5$ and *fx*$^{10}$ feature fully configurable framebuffer and texture memory. The device driver manages the memory to satisfy a broad range of framebuffer configurations and store memory-intensive texture maps.

Supported framebuffer configurations include:

- 32 bit/pixel RGBA, 24 bit/pixel RGB, or 16 bit/pixel RGB color buffers, single or double buffered, mono or stereo
- A 24 bit or 16 bit depth buffer
- A 4 bit stencil buffer
- 8 bits of single or double buffered overlay planes
- 16 bit video overlay
- A hardware accelerated accumulation buffer for fast full scene antialiasing
- A single clip plane for accelerated window clipping
- Additional bit planes to support per-window attributes such as fast buffer swaps

Texture maps are stored in framebuffer memory at up to 32 bits/pixel RGBA.

The **hp** *fx*$^5$ contains 64MB total shared framebuffer/texture SDR memory running at 166MHz. The **hp** *fx*$^{10}$ supports 64MB of fast SDR memory also running 166MHz.

## *OpenGL Support*

The **hp** *fx*$^5$ and *fx*$^{10}$ provide industry leading OpenGL performance, featuring an optimized display list execution path and enhanced state change architecture.

Both the **hp** *fx*$^5$ and *fx*$^{10}$ meet the conformance requirements for the OpenGL 1.1 industry standard. In addition, the **hp** *fx*$^5$ and *fx*$^{10}$ support several OpenGL extensions, so applications can access hardware features that are not exposed through the OpenGL 1.1 API. The extensions include:

- Industry standard OpenGL 1.1 texture mapping extensions, such as generate mipmap, texture border clamp, shadow, and depth texture.
- Many features which are part of the OpenGL 1.2 standard are supported through extensions, including: BGRA pixel formats; three dimensional texture maps; normal rescaling; texture coordinate edge clamping; and texture lighting.

- The **hp** texture color table extension.
- The **hp** draw array set extension, which allows rendering of multiple individual primitives through the vertex array feature.
- HP extensions for visibility testing, which can be used directly by an OpenGL application, or indirectly via the DirectModel or Fahrenheit APIs.
- The **hp** supersample extension, which provides support for full scene antialiasing.
- The vertex array, polygon offset, and subtexture features, which were only avail able as extensions under OpenGL 1.0, are supported via both the OpenGL 1.1 interface as well as the OpenGL 1.0 extension interface for backwards compatibility.
- Support for standard Windows NT® and Windows 2000 Professional® extensions, including paletted textures and swap hint.

The **hp** *fx*$^{5}$ and *fx*$^{10}$ are identical in terms of OpenGL feature support. An OpenGL application that runs on one device will run on the other.

## DirectDraw and Direct3D Support

In addition to their industry leading OpenGL support, the **hp** *fx*$^{5}$ and **fx**$^{10}$ also fully support Microsoft's® DirectDraw and Direct3D API. Features include:

- Standard DirectDraw and Direct3D features, such as vertex and table fog, multi-texturing, 32 and 16 bit textures, and direct framebuffer access, supported in hardware.
- 24 bit color and 16 bit Z buffers.
- Hardware support for 16 bit/pixel video overlay at all resolutions.
- Hardware support for scalable YUV video overlay.
- Hardware blits with color keying.
- Multi buffering.
- DirectX v7.0 Hardware Access Layer (HAL).

## 2D Support

The **hp** *fx*$^{5}$ and **fx**$^{10}$ provide exceptional 2D performance for operations such as area fill, hardware blit, hardware cursor, text display, and line rendering.

## Software

Several hardware features of the **hp** *fx*$^{5}$ and *fx*$^{10}$ may be controlled by the display properties dialog.

The wide range of different graphics devices available for MS Windows® has resulted in graphics applications having made different assumptions about how the graphics device works. The Options tab of the display properties dialog allows the user to easily customize the **hp** *fx*$^{5}$ and *fx*$^{10}$'s behavior to match the assumptions made by the application. Many pre-defined feature combinations are available for easy selection. These feature combinations provide maximum performance and compatibility for a large variety of key 3D graphics applications, including PTC Pro/ENGINEER, Dassault Systems Catia, SDRC I-DEAS Master Series, Autodesk Inventor, Alias/Wavefront Maya, Discreet 3D Studio MAX, and SoftImage. Additionally, individual hardware performance and compatibility features may be enabled or disabled separately, including buffer swap synchronization, fast swaps and clears, display list and lighting optimizations, stereo sync, and the hardware accumulation buffer.

The Gamma Correction tab controls the gamma correction of 3D windows. Adjusting the gamma value allows for correct color ramp brightness values on a wide range if different display types.

In addition to the selection of video formats and display frequencies available from the Settings tab, the Customize Video Formats tab allows the creation of user-defined video formats and frequencies.

## *Comparing the hp $fx^5$ and $fx^{10}$ to the hp $fx^{4+}$ and $fx^{6+}$*

The **hp** *$fx^5$* and **$fx^{10}$** represent an evolution of the previous **hp** *$fx^{4+}$* and *$fx^{6+}$* graphics accelerators. The following table illustrates the difference in features and functionality between the two sets of devices.

| **Feature** | **hp $fx^{4+}$ / $fx^{6+}$ Graphics Accelerators** | **hp $fx^5$ / $fx^{10}$ Graphics Accelerators** |
|---|---|---|
| Available video memory | 18MB SGRAM | 64MB SDR /64MB SDR |
| Maximum display resolution | 1600 × 1200 | 1920 × 1200 |
| Visibility testing and occlusion culling | ✔ | ✔ |
| Multiple visibility test results in parallel | | ✔ |
| Visibility statistics | | ✔ |
| Hardware accumulation buffer | ✔ | ✔ |
| Antialiased points and lines | ✔ | ✔ |
| Software-assisted Full scene antialiasing | ✔ | ✔ |
| Hardware full scene antialiasing | | ✔ |
| 8-bit destination alpha planes | | ✔ |
| Hardware Direct3D support | | ✔ |
| Hardware direct framebuffer access | ✔ | ✔ |
| 16-bit 565 RGB format | | ✔ |
| Double buffered overlay | software | hardware |
| Flat panel display | | ✔ |
| Extended video support | | ✔ |
| 16-bit YUV video overlay | | ✔ |
| Texture map hardware | optional | integrated |
| Texture memory | 16MBdedicated | 64MB / 64MB integrated with FB* |
| On-chip texture cache | | ✔ |
| Hardware environment mapping | ✔ | ✔ |
| Texture LOD | | ✔ |
| Direct 3D single pass multitexturing | | ✔ |
| Paletted textures, texture color tables | | ✔ |

**** 48MB available at 1280 x1024, True Color double-buffered resolution and 24 bit Z-buffer.***