

# ***TVP4010 USERS GUIDE***

## ***Architecture Overview***

SLAU009  
February 1997



## **IMPORTANT NOTICE**

Texas Instruments (TI) reserves the right to make changes to its products or to discontinue any semiconductor product or service without notice, and advises its customers to obtain the latest version of relevant information to verify, before placing orders, that the information being relied on is current.

TI warrants performance of its semiconductor products and related software to the specifications applicable at the time of sale in accordance with TI's standard warranty. Testing and other quality control techniques are utilized to the extent TI deems necessary to support this warranty. Specific testing of all parameters of each device is not necessarily performed, except those mandated by government requirements.

Certain applications using semiconductor products may involve potential risks of death, personal injury, or severe property or environmental damage ("Critical Applications").

**TI SEMICONDUCTOR PRODUCTS ARE NOT DESIGNED, INTENDED, AUTHORIZED, OR WARRANTED TO BE SUITABLE FOR USE IN LIFE-SUPPORT APPLICATIONS, DEVICES OR SYSTEMS OR OTHER CRITICAL APPLICATIONS.**

Inclusion of TI products in such applications is understood to be fully at the risk of the customer. Use of TI products in such applications requires the written approval of an appropriate TI officer. Questions concerning potential risk applications should be directed to TI through a local SC sales office.

In order to minimize risks associated with the customer's applications, adequate design and operating safeguards should be provided by the customer to minimize inherent or procedural hazards.

TI assumes no liability for applications assistance, customer product design, software performance, or infringement of patents or services described herein. Nor does TI warrant or represent that any license, either express or implied, is granted under any patent right, copyright, mask work right, or other intellectual property right of TI covering or relating to any combination, machine, or process in which such semiconductor products or services might be or are used.

## Preface

# Read This First

---

---

---

### ***About This Manual***

This manual describes the architecture of the TVP4010 graphics processor. The TVP4010 balances high quality 3-D texturing and graphics performance with leading edge windows, video and SVGA acceleration.

Only a basic understanding of computer graphics is required.

### ***How to Use This Manual***

This document contains the following chapters:

Chapter 1 gives an overview of the TVP4010.

Chapter 2 lists performance characteristics for the chip.

Chapter 3 describes typical board designs.

Chapter 4 describes feature highlights.

Chapter 5 describes 3-d features of the TVP4010.

Chapter 6 describes TVP4010 architecture.

Chapter 7 discusses external interfaces.

Chapter 8 describes display configurations.

Chapter 9 describes software drivers.

Chapter 10 describes manufacturing kits.

Chapter 11 provides a programming model.

A glossary of technical terms follows chapter 11.

***Related Documentation From Texas Instruments***

The following books describe the TVP4010 and related support tools. To obtain a copy of any of these TI documents, call the Texas Instruments Literature Response Center at (800) 477–8924. When ordering, please identify the book by its title and literature number.

TVP4010 Programmer Reference Guide, Literature No. SLAU006

TVP4010 Installation Manual, Literature No. SLAU008

TVP4010 Data Manual, Literature No. SLAS155

**If You Need Assistance . . .**

<b>If you want to . . .</b>	<b>Contact Texas Instruments at . . .</b>
Visit TI online	World Wide Web: <a href="http://www.ti.com">http://www.ti.com</a>
Receive general information or assistance	World Wide Web: <a href="http://www.ti.com/sc/docs/pic/home.htm">http://www.ti.com/sc/docs/pic/home.htm</a>
	North America, South America: (214) 644-5580
	Europe, Middle East, Africa
	Dutch: 33-1-3070-1166
	English: 33-1-3070-1165
	French: 33-1-3070-1164
	Italian: 33-1-3070-1167
	German: 33-1-3070-1168
	Japan (Japanese or English)
	Domestic toll-free: 0120-81-0026
International: 81-3-3457-0972 or 81-3-3457-0976	
	Korea (Korean or English): 82-2-551-2804
	Taiwan (Chinese or English): 886-2-3771450
Order Texas Instruments documentation (see Note 1)	Literature Response Center: (800) 477-8924
Make suggestions about or report errors in documentation (see Note 2)	Email: <a href="mailto:comments@books.sc.ti.com">comments@books.sc.ti.com</a>
	Mail: Texas Instruments Incorporated Technical Publications Manager, MS 702 P.O. Box 1443 Houston, Texas 77251-1443

- Notes:**
- 1) The literature number for the book is required; see the lower-right corner on the back cover.
  - 2) Please mention the full title of the book, the literature number from the lower-right corner of the back cover, and the publication date from the spine or front cover.

***Trademarks***

3Dlabs, GLINT Delta, and TVP4010 are registered trademarks of 3Dlabs, Inc.

CGL is a trademark of Creative Labs, Inc.

Heidi and 3D Studio MAX are trademarks of Autodesk, Inc.

Macintosh, QuickDraw, QuickDraw3D, and QuickDraw RAVE are trademarks of Apple Computer, Inc.

Microsoft, Windows, Win32, Direct3D, Windows95 and Windows NT are trademarks of Microsoft Corp.

OpenGL is a trademark of Silicon Graphics Inc.

RAMDAC is a trademark of Brooktree Corp.

TI is a trademark of Texas Instruments Incorporated.

# Contents

---

---

---

---

<b>1</b>	<b>Introduction</b>	<b>1-1</b>
1.1	What is the TVP4010?	1-2
1.1.1	The TVP4010 for Windows95t and Direct3Dt	1-2
1.1.2	The TVP4010 for Windows NTt and OpenGLt	1-2
1.1.3	The TVP4010 for QuickDraw3-Dt and QuickDraw RAVEt	1-2
1.2	TVP4010 Key Features	1-4
<b>2</b>	<b>Performance Characteristics</b>	<b>2-1</b>
2.1	3D Performance	2-2
2.2	Video Playback	2-2
2.3	2-D Performance	2-2
2.4	Host Interface Performance	2-3
<b>3</b>	<b>Typical Board Designs</b>	<b>3-1</b>
3.1	A Reference Board for All	3-2
3.2	A Reference Board for Power Users	3-2
3.3	Early Access Program	3-3
<b>4</b>	<b>Feature Highlights</b>	<b>4-1</b>
4.1	Feature Highlights	4-2
<b>5</b>	<b>3-D Features</b>	<b>5-1</b>
5.1	Benefits for Developers	5-2
<b>6</b>	<b>Architecture</b>	<b>6-1</b>
6.1	Internal Architecture	6-2
6.2	Graphics Core Units	6-3
6.2.1	Rasterizer	6-3
6.2.2	Scissor and Stipple	6-3
6.2.3	Depth and Stencil Test	6-3
6.2.4	YUV–RGB Converter	6-3
6.2.5	Color Interpolator	6-4
6.2.6	Texture, Fog, and Blend	6-4
6.2.7	Dither	6-4
6.2.8	Logic Ops	6-5
<b>7</b>	<b>External Interfaces</b>	<b>7-1</b>
7.1	Host Interface	7-2

7.1.1	PCI Bus .....	7-2
7.1.2	DMA Controller .....	7-2
7.1.3	Input FIFO .....	7-3
7.1.4	Interrupt Controller .....	7-3
7.1.5	Memory Bypass .....	7-3
7.1.6	GLINT Delta Interface .....	7-3
7.2	Memory Interface .....	7-4
7.2.1	SGRAM Overview .....	7-4
7.2.2	Memory Use .....	7-5
7.2.3	ROM Interface .....	7-5
7.3	Video Output .....	7-6
<b>8</b>	<b>Display Configurations .....</b>	<b>8-1</b>
8.1	Framebuffer Color Formats .....	8-2
8.2	Texture Color Formats .....	8-2
8.3	Depth and Stencil Formats .....	8-2
8.4	Typical Display Resolutions .....	8-3
8.5	Typical 3D Game Formats .....	8-3
8.6	Double Buffering .....	8-4
8.7	Overlays .....	8-4
8.8	Fast Screen and Depth Clearing .....	8-5
<b>9</b>	<b>Software Drivers .....</b>	<b>9-1</b>
9.1	3-D Drivers .....	9-2
9.2	Register-Level Access .....	9-2
9.3	2-D Drivers .....	9-2
9.4	Game Developer and Bundling Programs .....	9-3
<b>10</b>	<b>Manufacturing Kits .....</b>	<b>10-1</b>
<b>11</b>	<b>Programming Model .....</b>	<b>11-1</b>
11.1	State Update .....	11-2
11.2	Interrupts .....	11-2
11.3	Rendering a Primitive .....	11-2

# Figures

---

---

---

---

1-1	The 3D Market .....	1-3
3-1	Reference Board for All .....	3-2
3-2	Reference Board for Power Users .....	3-2
6-1	TVP4010 Internal Architecture .....	6-2
7-1	TVP4010 PCI Interface .....	7-2
7-2	TVP4010 Memory Interface .....	7-4

x

# Introduction

---

---

---

---

This user's guide provides an overview of the TVP4010 graphics processor architecture for design engineers and project managers. Hardware and software engineers can use this document as an introduction to TVP4010 architecture before proceeding to detailed information in other TVP4010 data and design documents.

## 1.1 What is the TVP4010?

The TVP4010 is a high performance graphics processor that balances high quality 3-D textured graphics acceleration, windows acceleration, and state-of-the-art video playback with a fast super video graphics adapter (SVGA).

The TVP4010 satisfies the need for high-quality, low-cost 3-D graphics and multimedia acceleration. Based on a proven scalable architecture, the TVP4010 accelerates a broad range of applications, including games, multimedia, animations, presentations, authoring, 3-D internet browsers, personal computer aided design (CAD), and visualization. Figure 1-1 shows the TVP4010 applications market.

### 1.1.1 The TVP4010 for Windows95™ and Direct3D™

A 2- to 4-Mbyte graphics board based on the TVP4010 processor provides a 3-D accelerator for Windows95 applications based on Direct3D. This solution delivers 3-D and multimedia acceleration for 3-D applications such as web browsers, games, and personal design packages.

### 1.1.2 The TVP4010 for Windows NT™ and OpenGL™

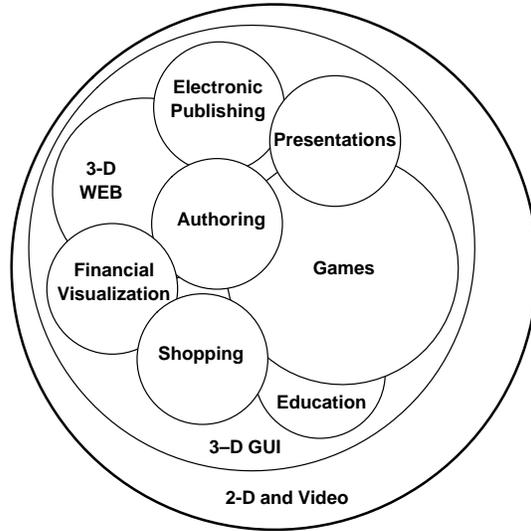
A 4- or 8- Mbyte graphics board using the TVP4010 and GLINT Delta™ combination provides a professional 3-D accelerator for Windows NT applications based on OpenGL. This solution combines the cost-effective rendering power of TVP4010 with the geometry set-up performance of GLINT Delta to provide acceleration for the most complex OpenGL applications. TVP4010 for Windows NT also offers 3D Studio MAX™ acceleration.

Entry level systems without GLINT Delta can also be configured for OpenGL and 3D Studio MAX, and TI™ drivers support Windows95.

### 1.1.3 The TVP4010 for QuickDraw 3D™ and QuickDraw RAVE™

For peripheral component interconnect (PCI) based Macintosh™ systems, the TVP4010 or TVP4010 plus GLINT Delta boards offer solutions that meet the price/performance requirements of the professional user. With built-in QuickDraw 3D specific features, bi-endian support, and high quality driver optimizations, TVP4010 is an ideal Macintosh 3-D and multimedia accelerator.

Figure 1-1. The 3-D Market



## 1.2 TVP4010 Key Features

- 42M textured bilinear-filtered pixels/s (TVP4010–80)
- 800K polygons/s (TVP4010–80)
- Polygon-based with optional depth (z) buffer
- Video playback acceleration
- Windows acceleration
- Fast on-chip SVGA
- PCI interface
- Low-cost synchronous graphics RAM (SGRAM)
- Single multifunction memory storage
- Optimized 3-D and 2-D drivers
- Reference board designs and manufacturing kits
- Supported by leading software developers

The TVP4010 is TI's second-generation low-cost graphics processor, combining proven hardware acceleration with extensive software support and design-in assistance to achieve fast time-to-market.

The TVP4010 turns the PC into a 3-D and multimedia platform.

# Performance Characteristics

---

---

---

---

The following paragraphs describe peak performance for the TVP4010 at 80 MHz.

## 2.1 3D Performance

- 42M textured pixels/s
- 800K textured polygons/s
- 120 Mbyte/s texture download rate
- Perspective correct textures
- Smooth shaded
- Depth-buffered
- Bilinear-filtered
- 4-bit palletized textures
- 50 displayed pixels per polygon
- 16-bit red-green-blue-alpha (RGBA) color at  $640 \times 480$  resolution

## 2.2 Video Playback

30 fps       $320 \times 200$  YUV (luminance, chrominance) source zoomed  
and filtered to  $640 \times 480$  16-bit RGB

## 2.3 2-D Performance

2 Gbytes/s	Fill rate using SGRAM block fill
120 Mbytes/s	Copy rate
2 Gbytes/s	Monochrome expansion

## **2.4 Host Interface Performance**

- 120 Mbytes/s PCI slave access to first-in, first-out (FIFO) storage
- 110 Mbytes/s PCI direct memory access (DMA) mastered read to FIFO storage
- 10 Mbytes/s PCI slave bypass read to memory
- 25 Mbytes/s PCI slave bypass write to memory
- 20 Mbytes/s PCI control register readback



# Typical Board Designs

---

---

---

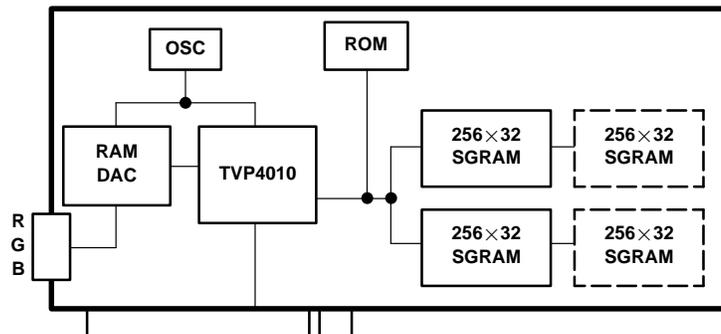
---

TVP4010-based boards can meet the requirements of particular markets and price/performance criteria. TI produces reference designs for common configurations. Figures 3–1 and 3–2 show two examples.

### 3.1 A Reference Board for All

The basic TVP4010 reference board provides low-cost, high-volume 3-D and multimedia acceleration. It is based on a TVP4010 and 2 Mbytes of memory with the option for an additional 2 Mbytes on the main board. Figure 3–1 shows the board components.

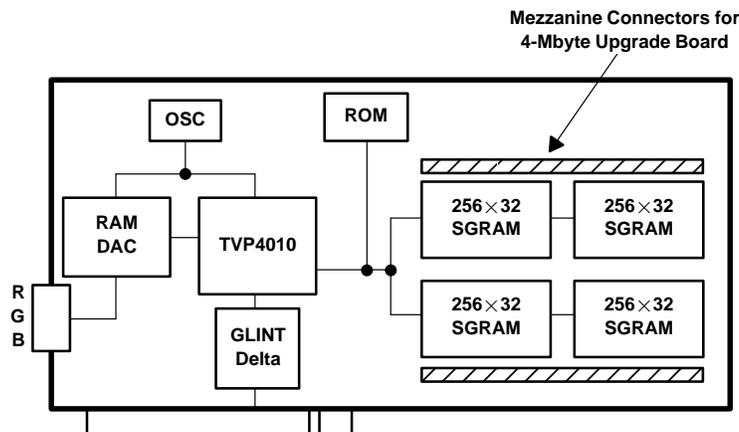
Figure 3–1. Reference Board for All



### 3.2 A Reference Board for Power Users

The TVP4010 reference board for power users adds a GLINT Delta and additional memory to the design. The GLINT Delta can double the performance of polygon-intensive applications by offloading floating-point calculations from the host and reducing the PCI bus traffic. For more information on the GLINT Delta see section 7.1.6. Figure 3–2 shows the board components.

Figure 3–2. Reference Board for Power Users



### **3.3 Early Access Program**

The TVP4010 Early Access Program (EAP) is for independent hardware vendors (IHVs) who wish to work closely with TI in bringing TVP4010-based designs to market quickly and efficiently. Supporting a close technical and marketing collaboration, the program is open to IHVs committed to developing TVP4010-based boards, and includes the following:

- Close technical support and joint marketing and press programs
- Early access to design engineers, design guides, and application notes
- Priority supply of sample parts and access to reference board schematics
- Participation in driver beta programs



# Feature Highlights

---

---

---

---

This chapter lists features that make the TVP4010 a high-performance graphics processor.

## 4.1 Feature Highlights

- Texture mapping
  - True perspective correction
  - Bilinear filtering
  - Palletized and RGB textures
  - Local texture buffer
  - Specular highlights
  - Fast texture loading
- 3-D rendering
  - Points, lines, triangles and bitmaps
  - Gouraud and flat shading
  - 8- or 16-bit RGBA or color indexed
  - Depth (z) buffering
  - Fogging and depth cueing
  - Alpha blending
  - Full screen antialiasing
  - Dithering
  - Area stippling
  - Stencil test and stencil buffer
  - Scissor test and logic operations
- Display features
  - 8-, 16-, and 32-bit RGB
  - Double and triple buffering
  - Hardware dithering
  - Hardware pan
  - Per-window double buffering
  - Overlays
- Programming interfaces
  - Direct3D and OpenGL

- Windows95 and Windows NT
- Creative Labs CGL™
- QuickDraw3D and QuickDraw RAVE
- Heidi™ for 3D Studio MAX
- Fast video playback
  - Moving Pictures Experts Group (MPEG) compliant
  - YUV color space conversion
  - Scaling (bilinear filtered)
  - Dithering
  - Chroma keying (blue-screen)
- Graphical user interface (GUI) acceleration
  - Bit block transfer (bitblt)
  - Points, lines, and polygons
  - Fills and text primitives
  - True color (16.7M)
  - Fast linear frame buffer
  - On-chip SVGA
  - Windows™ and QuickDraw™
- PCI interface
  - 32-bit glueless PCI Revision 2.1
  - Target and master support
  - DMA mastering
  - 32-entry command FIFO
  - Bi-endian apertures on bus
  - Interrupts
- Memory architecture
  - 64-bit SGRAM interface
  - Single multifunction memory
  - Frame buffer, back buffer, z buffer, and textures

## Feature Highlights

---

- Optimal memory usage
  - 2- to 8-Mbytes
- Display resolutions 320×200 to 1600×1200
- Video output
  - Internal video timing generator (VTG)
  - RAMDAC™ interface
- Green PC and plug and play
  - Video Electronics Standards Association (VESA) display power management signaling (DPMS)
  - VESA display data channel (DDC) support
- Industry standard package
  - Ball grid array (BGA)
  - 3.3 V (5-V tolerant I/O)
  - 3 W at 3.3 V

# 3-D Features

---

---

---

---

In addition to its 2-D and video features, the TVP4010 provides acceleration of the rendering features and special effects used by developers in 3-D-based authoring, professional, entertainment, and multimedia titles. These features include:

- Improved visual quality and performance
- Highly optimized software drivers for:
  - Windows95
  - Windows NT
  - QuickDraw
  - OpenGL
  - Direct3D
  - RAVE
  - Heidi
- New and improved special effects
- Increased color depth, screen resolutions, and frame rates

---

## 5.1 Benefits for Developers

The TVP4010 offers the following benefits for developers:

- Perspective-correct, bilinear-filtered texture mapping
  - More realism, reduced polygon count, non-blocky texturing, no shimmering
  - Palletized textures with internal look-up table (LUT), bilinear filtering for quality
- Smooth shading for highly realistic pixel-perfect 3D surface rendering
- Fog and depth cueing for realistic atmospheric effects and visual depth cues for landscapes
- Filtered zoom and dithering for high-quality visual output and image zoom even with reduced color depths
- Blending/transparency operations (with depth cueing) for high-quality translucent objects, explosions, and day/night effects
- Depth (z) buffering for fast hidden-surface elimination and reduced software/host overhead
- Sprite animation with perspective correction and scaling for depth-buffered overlays
- Video acceleration with real-time YUV video sources used as textures, or fast-scaled playback
- Chroma keying (blue screen) to overlay 3D data on top of images or real-time video
- Area stippling and stencil buffering for shadow and transparency effects and arbitrary-shaped 3D cutout definition
- Hardware picking and extent checking for reduced screen updating and target selection
- Full screen antialiasing
- Multifunction memory allows use of rendered images as textures
- Specular highlights for realistic lighting effects

# Architecture

---

---

---

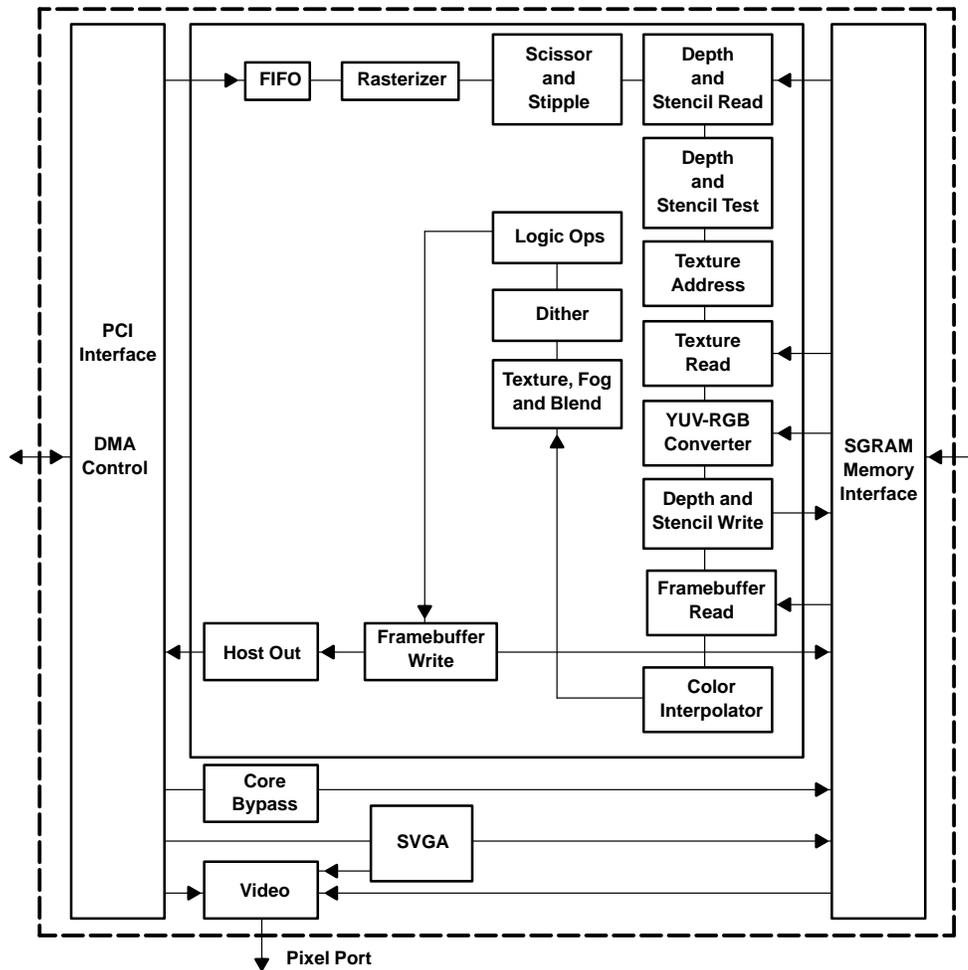
---

The TVP4010 internal architecture consists of a graphics core augmented by the I/O and memory interfaces described in this chapter. The video graphics adapter (VGA) is an independent unit that shares the memory controller to access the framebuffer.

## 6.1 Internal Architecture

Figure 6–1 shows the TVP4010 internal architecture.

Figure 6–1. TVP4010 Internal Architecture



## 6.2 Graphics Core Units

The TVP4010 graphics core uses the rendering algorithms and comprises a series of pipelined units. Any unit may be bypassed if it is not required.

### 6.2.1 Rasterizer

The rasterizer converts points, lines, bitmaps, and trapezoids into a list of x and y pixel coordinates, with subpixel precision when required. Images and textures are downloaded and uploaded by rasterizing trapezoids. The rasterizer also performs bitmask testing (e.g. for patterned lines) and byte swapping. The rasterizer supports SGRAM block fills.

### 6.2.2 Scissor and Stipple

The TVP4010 supports two scissor tests for rejecting pixels that lie outside a specified rectangle: the user scissor test, which can be used for window clipping, and the screen scissor test to reject off-screen pixels. A pixel must pass both tests to proceed for further processing by the graphics core.

Stippling tests each pixel against a bit in a predefined pattern. The pixel can be rejected or accepted for further processing depending on the result. Area stipple patterns (8×8) are supported and used for transparency effects and shadows.

### 6.2.3 Depth and Stencil Test

The depth (z) test compares an incoming-pixel depth value against the corresponding depth value stored in the localbuffer. The pixel can be rejected or accepted depending on the result of the depth test. This is the standard mechanism for removing hidden surfaces and lines, offering a simplified programming model and reduced load on the host. Eight conditional tests are supported.

The stencil test rejects (and conditionally modifies) pixels by comparing a reference value against the stencil value stored in a pixel destination address. This test masks out irregularly shaped portions of the screen (for example, a windshield) to prevent drawing.

### 6.2.4 YUV–RGB Converter

The YUV–RGB converter converts YUV (luminance, chrominance) data to RGBA (red green blue alpha). It can also do a chroma key test against upper and lower bounds for YUV or RGB data, with the test being applied independently against all four components.

### 6.2.5 Color Interpolator

The color interpolator unit generates color information for each pixel to be displayed. Two shading modes are supported: Gouraud (smooth) and flat. With Gouraud shading, given start and increment values, the unit performs linear interpolation. In flat shading mode, a constant color is associated with each pixel.

Two color modes are supported: RGBA and color index. With RGBA, red, green, blue and alpha components are generated for each pixel. In color index mode, a 4- or 8-bit index is generated.

### 6.2.6 Texture, Fog, and Blend

Texture-element (texel) data in RGBA, YUV, or color-indexed format is read and cached in the core from the localbuffer with perspective correction and bilinear filtering applied. Textures can be clamped, repeated, or mirrored at the boundaries of texture maps. The perspective correction and bilinear filtering can be turned on and off per primitive. Operations are performed in the order of texture, fog, and then blend.

Texture application can be done in one of three modes: copy, decal, or modulate. In copy mode the texture color replaces the current pixel color. In decal mode the texture color is blended with the current pixel color using the texture alpha value. In modulate mode the pixels current color and texture color are multiplied together.

Fog (or depth cueing) fades the current pixel color to a constant color based on the depth value. The degree of blending is proportional to the distance from the eye point and is performed on a per-pixel basis. Specular highlighting of textures is supported.

The alpha blend operation combines the current pixel color (after depth cueing) with the current value in the framebuffer. The blend value is constant across a primitive. Blending is supported in RGB mode.

Textures can be between 0 and 1536 pixels wide in multiples of 32, and between 0 and 1024 pixels high. They can be stored in both linear and patched formats in the localbuffer.

### 6.2.7 Dither

The TVP4010 applies a dither operation to reduce the internal color data from 8-bits per color component to the color format of the framebuffer. An ordered dither algorithm dithers polygons and a proprietary algorithm for lines.

### **6.2.8 Logic Ops**

The logic ops unit performs logical and software plane mask operations between the pixel and the original pixel color (e.g. XOR). The standard 16 logic operations are supported.



# External Interfaces

---

---

---

---

The TVP4010 has three primary external interfaces:

- Host interface (PCI bus)
- Memory interface
- Video port

The host interface over the PCI bus supplies control and data to the chip from the host processor. The memory interface accesses color, depth, stencil, and texture data. The memory interface also supports an external ROM. The video pixel port supplies video and timing data to the RAMDAC.

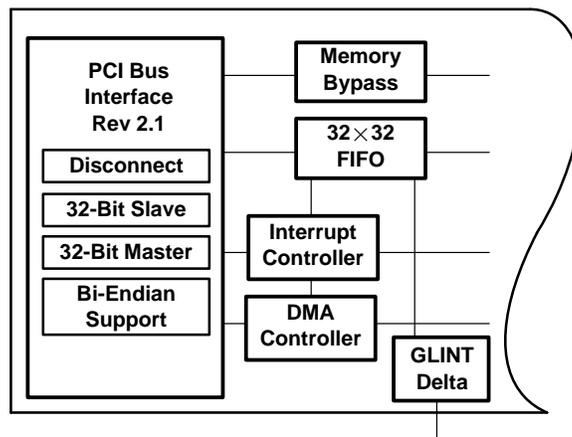
The color data (ready for display) is called the framebuffer, including any back buffers. The depth, stencil, and texture data are collectively called the localbuffer. The framebuffer and localbuffer reside in the same physical memory and can be placed anywhere in memory to minimize memory waste and offer a simplified programming model.

## 7.1 Host Interface

The host interface on the TVP4010 is PCI-Revision-2.1 compliant and contains a first-in, first-out (FIFO) memory and direct memory access (DMA) controller. Control registers for the host interface are memory mapped onto the PCI bus. The host can read control and state information from the programmable registers.

The host can communicate with the TVP4010 either directly to the FIFO, in which case the TVP4010 acts as a PCI slave, or, alternately, the TVP4010 can be programmed as a PCI master using the internal DMA controller to fetch commands into the FIFO. Figure 7–1 shows the host interface.

Figure 7–1. TVP4010 PCI Interface



### 7.1.1 PCI Bus

The PCI bus includes these features:

- Glueless interface – simple and low-cost design-in
- 32-bit master/slave – maximum speed
- Bi-endian – avoids byte swapping on PowerMacs™
- Revision 2.1 compliant – plug and play

### 7.1.2 DMA Controller

The DMA controller includes these features:

- Autonomous – setup/fetch parallelism
- No wait state – maximum transfer rate
- Programmable block size – large DMA buffers

### 7.1.3 Input FIFO

The input FIFO includes these features:

- 32 entries – fetch/draw parallelism
- Burst mode – bursts for programmed I/O
- PCI disconnect-on-full – avoids polling FIFO

### 7.1.4 Interrupt Controller

The interrupt controller includes these features:

- End-of-DMA – allows DMA chaining
- VSYNC – efficient double buffering
- Scanline – special effects
- Sync – indicates graphics core is idle
- Error – e.g. writing to a full FIFO

### 7.1.5 Memory Bypass

The memory bypass provides fast access to memory for software rendering

### 7.1.6 GLINT Delta Interface

The GLINT Delta is an 80-million-floating-point-operations-per-second (MFLOPS) OpenGL- and Direct3D-compliant set-up processor, designed to break the 3-D bottleneck on PCs that are unable to saturate the TVP4010 rendering capabilities. GLINT Delta calculates the slope and set-up information, and performs high precision floating-to-fixed point conversion. GLINT Delta reduces the load on the CPU and the PCI bus and supports any 3-D application program interface (API).

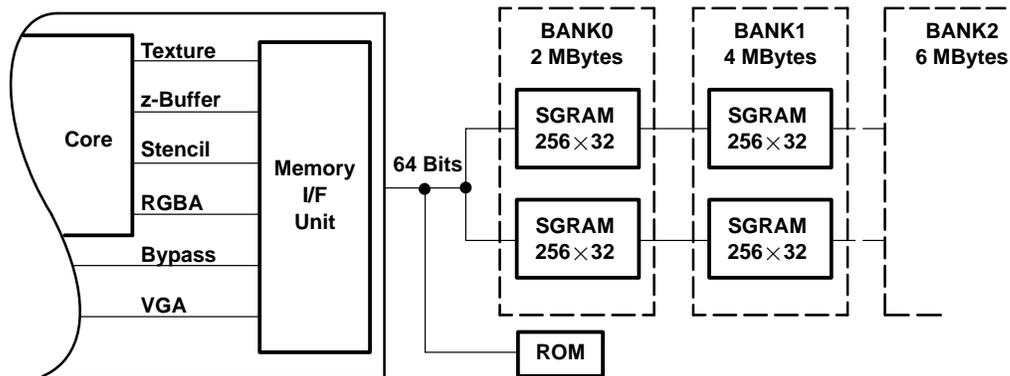
A separate architecture overview for the GLINT Delta is available from TI.

## 7.2 Memory Interface

The TVP4010 memory subsystem uses synchronous graphics RAM (SGRAM) to supply the 600 Mbytes/s needed for 3-D operations and display updates. The memory interface accesses color, depth, stencil, and texture data. The color data (ready for display) is called the framebuffer, and the depth, stencil, and texture data are collectively called the localbuffer.

The multi-function memory area allows the framebuffer and localbuffer to reside anywhere in the same physical memory, minimizing memory wastage and offering a simplified programming model. The VGA is an independent unit that shares the memory controller to access the framebuffer. Figure 7–2 shows the memory interface.

Figure 7–2. TVP4010 Memory Interface



### 7.2.1 SGRAM Overview

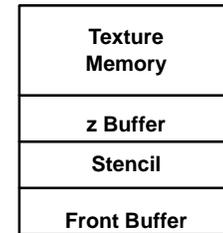
The SGRAM provides the bandwidth for screen refresh and 3D graphics, avoiding the need for expensive video RAM (VRAM) while retaining the following benefits, such as block writes and masked writes.

- 64-bit SGRAM Interface
- Two 256K × 32 parts for every 2 Mbytes
- 2, 4, 6, or 8 Mbytes
- 2, 4, or 8 pixels packed per 64-bit word
- Memory operates up to 100 MHz
- High-speed block fill
- Masked writes
- Typically 3.3 V in 100-pin QFP (JEDEC defined)

## 7.2.2 Memory Use

The TVP4010 can store a variety of data and color formats in memory at the same time, such as:

- Mixed RGB, color index, and YUV
- Mixed 4-, 8-, 16- or 24-bit textures
- Hardware double buffering
- Optional z buffer and stencil
- Linear or patched accessing



To minimize page breaks, the TVP4010 can store and access data in memory as 2D patches. This is particularly useful for texture maps, where texture accesses can go in any direction through memory. By storing the data in a 2D format, the chances of a page break are reduced. The TVP4010 also supports a proprietary and complementary patching system to increase the probability of a cache hit on texel reads.

TI drivers allow configurable texture compression on download, which saves memory and increases performance.

## 7.2.3 ROM Interface

The read-only memory (ROM) interface allows the host to access an expansion ROM through the PCI interface, allowing initialization and configuration data to be stored. Using the SGRAM data lines, the ROM interface supports 16 address and 8 data lines to access the ROM.

### **7.3 Video Output**

The TVP4010 has an internal video unit that permits a complete framebuffer to be installed with the addition of SGRAM, a clock generator, and suitable RAMDAC. (Some RAMDACs do not need an external clock generator.)

The video unit reads pixels for display from memory and stores them in a FIFO. The data is clocked out of the FIFO and sent to a RAMDAC through the high-speed pixel port, along with the timing control signals. The video unit and timing generator are controlled through a series of registers accessible from the PCI bus.

# Display Configurations

---

---

---

---

The TVP4010 supports a number of display formats and screen resolutions. For 3-D graphics the maximum color resolution is 16-bit, and for 2-D it is 24-bit. The TVP4010 enables 16-bit 3-D graphics to be displayed while running a 24-bit 2-D desktop.

## 8.1 Framebuffer Color Formats

The TVP4010 supports RGBA, color indexed (CI) and YUV (specifically YCbCr for MPEG compatibility) data being stored in the framebuffer. Both RGBA and BGRA ordering of pixels is supported. YUV is converted to the framebuffer format before display.

8-bit: 2:3:2:1 or 3:3:2:-  
16-bit: 5:5:5:1 or 5:6:5:- or 4:4:4:-  
32-bit: 8:8:8:8  
CI: 8:-:-  
YUV: 4:1:1 or 4:2:2

## 8.2 Texture Color Formats

Textures can be stored in the localbuffer in any combination of the color formats described in the framebuffer color formats.

- RGBA and color index
- Palletized
- YUV

If the texture format is different from the framebuffer format, the graphics core performs the conversion into the internal color format. If the texture map is 4 bits deep (palletized), the user-defined internal look-up table converts the data into the internal format.

## 8.3 Depth and Stencil Formats

The use of depth and stencil buffers is optional. Not using depth or stencil buffers increases the memory available to support higher display resolutions and larger local texture storage.

Depth: 0, 15, or 16 bits  
Stencil: 0 or 1 (if 1 then the depth must be set to 15)

## 8.4 Typical Display Resolutions

The TVP4010 supports the following display resolutions:

Screen widths: 320, 384, 512, 640, 800, 1024, 1152 and 1280 (or any multiple of 32 pixels between 0 and 1280)

Screen heights: Any height from 0 to 1024 pixels

## 8.5 Typical 3-D Game Formats

The following table shows the color and display resolutions and texture storage supported for some typical memory configurations for double-buffered 3-D games. Supported resolutions for 2-D graphics are higher, and are not shown in this table.

The texture column indicates the amount of texture memory available after the framebuffer (including back buffer) and depth buffer have been allocated.

Width	Height	Color	z	Texture Space	
				2Mb Card	4Mb Card
512	384	16-bit	0-bit	1.2 Mb	3.2 Mb
512	384	16-bit	16-bit	0.8 Mb	2.8 Mb
512	384	8-bit	0-bit	1.6 Mb	3.6 Mb
512	384	8-bit	16-bit	1.2 Mb	3.2 Mb
640	400	16-bit	0-bit	1 Mb	3 Mb
640	400	16-bit	16-bit	0.5 Mb	2.5 Mb
640	400	8-bit	0-bit	1.5 Mb	3.5 Mb
640	400	8-bit	16-bit	1 Mb	3 Mb
800	600	16-bit	0-bit	–	2.1 Mb
800	600	16-bit	16-bit	–	1.1 Mb
800	600	8-bit	0-bit	1 Mb	3 Mb
800	600	8-bit	16-bit	–	2.1 Mb

If there is not enough memory to support 3-D graphics at a particular resolution it is possible to render images at a lower resolution and zoom the image, with filtering, as it is copied to the front buffer.

## 8.6 Double Buffering

Double buffering allows new frames to be rendered off screen without disturbing the displayed picture; this is important for smooth animation. The TVP4010 supports three primary double-buffering mechanisms.

- Full screen
- Bit block transfer (bitblt) from off-screen memory
- Per window

In full-screen mode, the framebuffer holds two (or more) display buffers. The RAMDAC displays one buffer while the next frame is drawn into the other buffer. When the drawing is complete the buffers are flipped.

In bitblt mode an off-screen area of the framebuffer is used for the back buffer. When a new frame is ready the back buffer is copied to the on-screen front buffer. This mechanism supports multiple, independent, double-buffered windows and can render at a low resolution before zooming and filtering the image to the final display resolution.

In per-window mode a suitable RAMDAC is able to display a 16-bit pixel from either the upper or lower half of a 32-bit word. Hardware-write masks control which half is drawn to, with the buffers being swapped by changing the state of bit 31 (the alpha bit in 5:5:5:1 mode) using a combination of hardware-write masks and block fills.

## 8.7 Overlays

Overlays are useful for game and window systems that move sprites across a static background. In conjunction with a suitable RAMDAC the TVP4010 supports hardware overlays. Using the 5:5:5:1 color mode, it is possible to display a 16-bit pixel from either the upper or lower half of a 32-bit word; the alpha value and hardware-write masks control which half is drawn to, with the option of blending the overlay with the background image.

## **8.8 Fast Screen and Depth Clearing**

The TVP4010 supports two complementary methods for the fast clearing of the screen and depth buffer.

- Using hardware block fills supported by the SGRAM
- Using hardware extent checking

With the extent-checking unit, the area of the screen that has been updated is recorded and only this area needs to be cleared, thereby minimizing the number of pixels that need clearing.



# Software Drivers

---

---

---

---

---

TI delivers high-performance, high-quality, ready-to-ship software drivers that extract the maximum performance from the TVP4010 processor and the entire system.

## 9.1 3-D Drivers

The TVP4010 accelerates consumer-focused 3-D application programmers interfaces (API) and drivers including:

- Microsoft Direct3D
- Creative Labs CGL
- Apple QuickDraw3D and QuickDraw RAVE
- Autodesk Heidi for 3D Studio MAX support
- Custom 3-D engines and drivers
- Silicon Graphics OpenGL

Other drivers are available on request.

## 9.2 Register-Level Access

Programmers are given the option to access the hardware directly through the register-level interface. The precise programming model for the TVP4010 is described in the TVP4010 Programmer Reference (literature number SLAU006). Source code for a sample register-level driver is part of the Programmer Reference Guide.

## 9.3 2-D Drivers

The TVP4010 is a high-performance 2-D and video engine with optimized drivers for:

- Windows95
- Windows NT
- QuickDraw
- Windows 3.1™

Other drivers are available on request.

## **9.4 Game Developer and Bundling Programs**

An extensive game-developer support program puts the TVP4010-based development boards into the hands of the world's leading game developers. TI works closely with developers, providing advice and putting together game bundles for original equipment manufacturers (OEMs) as required.



## **Manufacturing Kits**

---

---

---

---

TI provides the TVP4010 Manufacturing Kit/Reference Designs to minimize development time and time-to-market. This kit contains a TVP4010-based PCI reference board along with extensive hardware design documentation, board schematics, ORCAD and Gerber files, design guides, application notes, and a full suite of 3-D and 2-D device drivers.



# Programming Model

---

---

---

---

The simplest way to view the programming interface to the TVP4010 is to view it as a flat block of memory-mapped registers (i.e., a register file). There are approximately 200 registers, and many of these registers are split into bit fields.

When a TVP4010 host software driver is initialized, it can map the register file into its address space. Each register has an associated address tag, giving its offset from the base of the register file. The most straightforward way to load a value into a register is to write the data to the mapped address. In reality the chip interface comprises a 32-entry-deep FIFO, and each write to a register causes the written value and the register address tag to be written as a new entry in the FIFO.

## 11.1 State Update

The TVP4010 uses a retained-state model. Some control registers, such as the StartX and StartY registers, have to be updated for almost every primitive, whereas control registers such as the scissor, logic op or dither, can be updated less frequently. Preloading the appropriate control registers can significantly reduce the amount of data that has to be loaded into the chip for a given primitive, thus improving efficiency. In addition, the final values in internal registers can sometimes be used for subsequent drawing operations.

## 11.2 Interrupts

The TVP4010 provides interrupt facilities as described in subsection 7.1.4.

## 11.3 Rendering a Primitive

To render a primitive, the required parameters are loaded into the TVP4010; the render command is then issued to draw the primitive. The exact parameters sent depend on the primitive type and what components (i.e., RGBA, depth, texture, etc.) are to be interpolated.

# Glossary

---

---

---

---

## A

**accumulation buffer:** A color buffer of higher resolution than the displayed buffer (typically 16bits per component for an 8bit per component display). Typically used to sum the result of rendering several frames from slightly different viewpoints to achieve motion blur effects or eliminate aliasing effects.

**activefragment:** A fragment which passes all the various culling tests, such as scissor, depth(Z), alpha, etc., is written to/combined with the corresponding pixel in the framebuffer. See also “fragment” and “passive fragment”.

**aliasing:** A phenomena resulting from a rendering style which ignores the fact that a pixel may not be wholly covered by a primitive, leading to jagged edges on primitives.

**alpha blending:** The ability to combine supplied Red, Green and Blue color values with those that exist in the framebuffer according to the supplied alpha value. Alpha blending forms the basis for techniques such as transparency and painting.

**alpha buffer:** A memory buffer containing the fourth component of a pixel's color in addition to Red, Green and Blue. This component is not displayed, but may be used for instance to control color blending.

**area stipple:** A two dimensional binary pattern which is used to cull fragments from being drawn.

## B

**bitblt:** Bit aligned block transfer. Copy of a rectangular array of pixels in a bitmap from one location to another.

**bitblt double buffering:** A technique to provide independent windowed double buffering by blting an area from one buffer to the other.

**bitplane double buffering:** A technique whereby fast independent windowed double buffering can be achieved by using a single bitplane bit.

**block write:** A feature provided in some memory devices such as VRAM and SGRAM which allows multiple pixels to be set to a given value by a single write. Fast fill is an alternative name for this feature.

## C

**chroma keying:** Also known as bluescreening, this is the practice of excluding color from an image allowing an underlying image to show through.

**chroma test:** The means by which chroma keying can be achieved.

**color index:** The mode in which the color information is stored for each pixel as a single number, the color index rather than as separate Red, Green, Blue and optionally Alpha values (RGBA mode). Each color index references an entry in a color look up table that contains a particular set of Red, Green and Blue values.

**command register:** A register which when loaded triggers activity in TVP4010. For instance the Render command register when loaded will cause TVP4010 to start rendering the specified primitive with the parameters currently set up in the control registers.

**context:** The state information associated with a particular task. Typically in a system more than one task will be using TVP4010 to render primitives. Software on the host must save away the current contents of the TVP4010 control registers when suspending one task to allow another to run, and must restore the state when that task is next scheduled to run.

**control register:** A register which contains state that dictates how TVP4010 will execute a command.

**culling:** The process of eliminating a fragment, object face, or primitive, so that it is not drawn.

## D

**DDA:** Digital Differential Analyser. An algorithm for determining the pixels to draw along a line or polygon edge. Also used to interpolate linearly varying values such as color and depth.

**delta:** A gradient of color, fog, depth etc. in the X or Y directions for a primitive.

**depth (Z) buffer:** A memory buffer containing the depth component of a pixel. Used to, for example, eliminate hidden surfaces.

**depth-cueing:** A technique that determines the color of a pixel based on its depth. Used, for instance, to fade far away objects into the background. Also known as fogging.

**dithering:** A rendering style that increases the perceived range of displayed colors at the cost of spatial resolution. The technique is similar to the use of stippled patterns of black and white pixels, to achieve shades of grey on a black and white display.

**dominant edge:** The side of a primitive such as a triangle, which has the greatest range of Y values.

**double-buffering:** A technique for achieving smooth animation, by rendering only to an undisplayed back buffer, and then swapping the back buffer to the front once drawing is complete.

## E

**extent checking:** A technique that determines the rectangular bounds of the area which has been rendered to.

## F

**fast fill:** A feature provided in some memory devices such as VRAM and SGRAM that allows multiple pixels to be set to a given value by a single write. Block write is an alternative name for this feature.

**flat shading:** The constant color shading or area filling of a primitive.

**fogging:** A technique that determines the color of a pixel based on its depth. Used, for instance, to fade far away objects into the background. Also known as depth-cueing.

**fragment:** A fragment is an object generated as a result of the rasterization of a primitive. It corresponds to and contains all the components of a single pixel. If a fragment passes all the various culling tests, such as scissor, depth(Z), stencil, etc., it is written to/combined with the corresponding pixel in the framebuffer.

**framebuffer:** An area of memory containing the displayable color buffers (front, back, left, right, overlay, underlay), their (optional) associated alpha components, and any associated (optional) window control information. This memory is typically separate from the localbuffer.

## G

**Gouraud shading:** The technique of variable color shading or area filling of a primitive using interpolation to gradually vary the color between vertices. Often known as smooth shading.

## H

**hardware writemask:** A bitmask implemented in memory devices such as VRAM and SGRAM to enable or inhibit the writing of the corresponding bits of a fragment's color into the framebuffer.

**host:** The processor which controls TVP4010.

## L

**localbuffer:** An area of memory that may be used to store textures and/or non-displayable depth(Z) and/or stencil pixel information. This memory is typically separate from the framebuffer.

**logic ops:** The technique of applying logical operations such as OR, XOR or AND to the fragment color values and/or those in the framebuffer.

**LUT:** A look-up-table. This normally contains color values to allow mapping from an index value to the desired red, green and blue value.

## O

**overlays:** The technique of ensuring certain drawn objects always remain foremost in view and not obscured by others. Historically this was one method of providing a cursor and was usually achieved by providing extra bit planes.

## P

**packed data:** The arrangement of data in a buffer which allows multiple pixels to be read or written in a single access.

**passive fragment:** A fragment which fails one or more of the various culling tests, such as scissor, depth (Z), stencil, etc., is not written to/combined with the corresponding pixel in the framebuffer. See also "fragment" and "active fragment".

**patched addressing:** A technique whereby data is organized in memory such that there is improved performance for accesses to adjacent scanlines in a buffer. For TVP4010, this is available for depth and/or stencil buffer accesses. For textures a special form, subpatch addressing is provided.

**picking:** A means of selecting drawn objects or primitives.

**preMult:** A method of alpha blending, also known as Ramp blend mode, used by QuickDraw3D.

**pixel:** Picture element. A pixel comprises the bits in all the buffers (whether stored in the localbuffer or framebuffer), corresponding to a particular location in the framebuffer.

**primitive:** A geometric object to be rendered. The TVP4010 primitives are points, lines, trapezoids (including triangles as a subset), and bitmaps.

## R

**Ramp blend mode:** A method of alpha blending, also known as preMult, used by QuickDraw3D.

**rasterization:** The act of converting a point, line, polygon, or bitmap, in device coordinates, into fragments.

**rendering:** Conversion of primitives in object coordinates into an image.

## S

**scissor test:** A means of culling fragments which lie outside the defined scissor rectangle. The scissor rectangle is defined in device coordinates.

**software witemasking:** A means of simulating hardware witemasking by performing a read-modify-write operation on framebuffer data.

**stencil buffer:** A buffer used to store information about a pixel which controls how subsequent stenciled fragments at the same location may be combined with its current value. Typically used to mask complex two-dimensional shapes.

**stipple:** A one or two dimensional binary pattern which is used to cull fragments from being drawn.

**subordinate edge:** The sides of a primitive such as a triangle, which do not have the greatest range of Y values.

**subpatch addressing:** A technique whereby data is organized in memory such that there is improved performance for accesses to adjacent scanlines in a buffer. For TVP4010, this particular form of patched addressing is available for accessing texture maps. See also Patch Addressing.

**subpixel correction:** A means of ensuring that all interpolated parameters associated with a fragment (color, depth, fog, texture) are correctly sampled at the fragment's center. This is required, for example, to ensure correct color shading of objects comprised of many primitives.

## T

**tag:** The data item that uniquely identifies a Graphics Core register.

**task:** A process, or thread on the host that uses the TVP4010 co-processor. Typically tasks assume that they have sole use of TVP4010 and rely on a device driver to save and restore their TVP4010 context, when they are swapped out.

**texel:** Texture element. An element of an image stored in texture memory which represents the color of the texture to be applied (fully or in part) to a corresponding fragment.

**texture:** An image used to modify the color of fragments during processing. Often used to achieve high realism in a scene, with relatively few primitives.

**texture mapping:** The process of applying a two-dimensional image to a primitive.

## W

**writemask:** A bit pattern used to enable or inhibit the writing of the corresponding bits of a fragment color into the framebuffer. See also Software Writemask and Hardware Writemask.

## Y

**YUV:** An alternative color format to RGB, also known as YCbCr. Color format used by MPEG.

## Z

**Z buffer:** An alternative name for the depth buffer.