Product Name: Volari XP10
Document Type:
General Reference Datasheet
Ver. No: 1.1

XGI Technology Inc.
Product Marketing
February 23, 2006

Volari XP10 Datasheet

NOTICE

The information in this document is subject to change, as the Company may make changes to product in order to improve reliability, design, or function, without prior written notice. No part of this manual may be reproduced or transmitted in any form or by any means without the written permission of the company.
IN NO EVENT WILL THE COMPANY BE LIABLE FOR SPECIAL INCIDENTAL OR CONSEQUENTIAL DAMAGES, WHETHER ARISING DIRECTLY OR INDIRECTLY, SUCH AS LOSS OF PROFIT OR GOOD WILL, THAT MAY BE SUFFERED IN CONNECTION WITH THE PURCHASE OF THIS PRODUCT OR FROM THE BREACH OF ANY REPRESENTATION OR WARRANTY.
January, 2005

LIMITED WARRANTY

This product is warranted against defects in materials and workmanship for a period of one year from the date of purchase. During the warranty period, product which the Company determines fails to meet warrant will be repaired or, at the Company's option, replaced at no charge. To be eligible for warranty service, product must be returned to the Company or to a Company authorized service center, costs of shipping prepaid. This warranty does not cover results of accident, abuse, neglect, use contrary to specifications or instructions, or repair or modification by anyone other than the Company.
THE COMPANY SPECIFICALLY DISCLAIMS ALL OTHER EXPRESS, IMPLIED OR STATUTORY WARRANTIES, INCLUDING THE IMPLIED WARRANTIES OF FITNESS FOR A PARTICULAR PURPOSE AND MERCHANTABILITY.
IF SUCH DISCLAIMER OF ANY IMPLIED WARRANTY IS NOT PERMITTED BY LAW, THE DURATION OF ANY SUCH IMPLIED WARRANTIES IS LIMITED TO 90 DAYS FROM THE DATE OF DELIVERY. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OF IMPLIED WARRANTIES OR LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY MAY LAST, OR THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO SUCH LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU. THIS WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS AND YOU MAY ALSO HAVE OTHER RIGHTS, WHICH VARY FROM JURISDICTION TO JURISDICTION.

TRADEMARK ACKNOWLEDGMENTS

1. Introduction

The information contained in this manual is intended to give technical assistance to design engineers in the development of PCI Express high-speed interconnect system boards utilizing the XGI Volari XP10® eXtreme Cache Technology® Multimedia 3D graphics processor.

This document includes a reference guide to all aspects of the design specifications and considerations in a typical development cycle.

The information regarding the Volari XP10 is presented in the following sections:

- Product Information
- Architectural and functional descriptions
- Electrical Specification
- Ball list, diagrams and descriptions

Note: For board level design, feature options and/or schematic samples, please review the Volari XP10 hardware design guide.

1.1 Purpose

This manual is a reference to guide design engineers in applications development. The material provided in this document provides the engineer with information for designing and laying out PCI Express bus interface cards using the Volari XP10, configuring the device, detailed pin signal information, and all other technical tasks required for success with the Volari XP10.

The information in this manual is subject to change without notice.

1.2 General Information

The Volari XP10 incorporates a native high-speed x16 PCI Express (PCIE) interface along with a highly integrated quad-pixel, single-cycle 3D/2D graphic engine and video accelerator with advanced DVD video de-interlacing and integrated TV output capability. A unique feature of the Volari XP10 is the eXtreme Cache Technology® (Discrete Unified Memory Architecture) which utilizes the high speed and bandwidth of the PCIE bus to access system memory to use as an additional frame buffer. The unique architecture of the Volari XP10 allows several memory configurations using onboard local DDR memory, system memory, or a combination of local and system memories. By exploiting the benefits of eXtreme Cache Technology, the amount of frame buffer available to the graphics processor (GPU) can be expanded beyond the physical limitations of the local DDR memories. The larger frame buffer (local + system memory) will give users a more cinematic and lifelike experience as more detailed information will be stored per pixel. The GPU can then render the realistic graphics scenes to give the user the ultimate visual experience.

The Volari XP10 also supports a video capture port to capture live MPEG 1/ MPEG 2 video streams, or DVD decompressed video streams to be overlaid with a graphics stream on multiplexed color depth displays. In supporting dual live videos, the Volari XP10 offers independent dual video windows with bi-cubical scaling capability ready for videoconferencing. The Volari XP10 contains one internal TMDS digital interface to connect to an LCD panel, a TV encoder, and two CRT interfaces. In addition to the display features above, the Volari XP10 also contains an advanced flat panel controller which supports two internal LVDS digital interface to directly connect UXGA panel(s) Integrating the programmable phase lock loop with high speed LUT DACs. Volari XP10 is a true price/performance solution for the modern multimedia based system.

2. Features

This chapter describes the features of the Volari XP10.

2.1 Overview

The Volari XP10 is a state-of-the-art graphics processor unit (GPU) developed by XGI. It delivers unparalleled price, performance, and image quality in 3D/2D graphics and DVD playback. The Volari XP10 incorporates Microsoft DirectX 9.0 programmable pixel and vertex shaders in hardware. The Volari XP10 has a flexible frame buffer interface utilizing a 64-bit bus interface to onboard (local) DDR SDRAM memory, an optional interface to use system memory as a frame buffer via the PCIE bus, or a combination of both onboard (local) and system memory interfaces. The Volari XP10 also integrates two high speed LVDS transmitters, one TMDS with HDCP interface, and a TV encoder for LCD panel, DVI and TV support. The x16 PCI Express bus interface enables the highest level of 3D/2D graphics performance. In addition, the Volari XP10 incorporates XGI's latest technologies (patented and patent-pending) to deliver the "best-of-breed" features in all categories of 3D/2D rendering, MPEG2/1 acceleration, video processing, dual display and power management.

Highly Integrated Design

- eXtreme Cache Technology® architecture provides multiple frame buffer interface options:
- Advanced Quad-Pixel, Single-pass 3D and 2D rendering engine
- Low power 0.13u CMOS process
- Support of Microsoft DirectX 9.0 API
- Support I²S input for sound capture
- Two internal LVDS digital interfaces, one internal TMDS with HDCP digital interface for LCD panel support, and two 10-bit DACs for CRT interfaces.
- Integrated TV encoder
- Digital interface to external TV encoder
- Digital interface to external DVI transmitter
- DVD hardware assists with Motion Compensation (MC) and Inverse Discrete Cosine Transform (IDCT)
- TrueVideo® with motion/edged adaptive Video De-interlacing
- Linear display memory addressing up to 4GB memory
- 256 Raster Operations (ROPs) up to 32-bit True Color
- 350 MHz built-in RAMDAC™and frequency synthesizer
- Two 32-bit hardware alpha cursors
- Two four-color hardware cursor and pop-up icons
- Two level Texture Cache
- On-die thermal sensor
- PC2001 Compliant

Performance

- Up to 250 MHz graphics engine clock
- Pixel fill rate: 1 billion pixels/sec
- Texel Processing rate: 4.8 billion texels/sec
- Up to 250 MHz DDR memory clock
- Up to 4 GBytes/sec memory bandwidth

BrightPixel™ Graphics Engine

- Up to 250 MHz engine clock
- Hierarchical pixel tiling for page base rendering
- 2x2 pixel pipelines and 4 independent Pixel-Shaders
- 2 pixels / clock with 8 texel / pixel
- Supports all DirectX 9.0 feature set and vertex format
- Texture size up to 8K x 8K and supports non-power of 2 textures
- Supports high order texture filtering up to 8x8 kernel
- Supports DirectX 9.0 sRGB linear color format

- Bi-linear, tri-linear and 16:1 anisotropic, flat cubic, and mipmap nearest/linear texture filtering
- Perspective correct mip-mapped texturing
- Texture compression and tiling
- Point sampled or bi-linear filtered texture maps are perspective corrected
- Full anti-aliasing support for texts, lines and scenes
- Special multi-resolution depth buffer
- Special bandwidth reduction hardware via compression
- Multi-level caches
- Fully Open-GL compliant blending including fog & depth cueing
- Special hardware support for DirectX 9.0 cubic mapping, bump mapping, and Alpha Blending
- Optimized 32-bit IEEE floating point setup engine
- 1024-bit with 9KB texture cache
- Single 64-bit frame buffer ports
- Flexible Pixel Shader
- Hardware accelerated TnL/Vertex Shader
- Simultaneous diffuse, Specula lighting, Gouraud Shading, Z-buffering and fog with no performance penalty
- Chroma key, Alpha blending and color key operations
- OpenGL 1.2.1/1.3/1.4 compliant

eXtreme Cache Technology® Architecture

- Optional 64-bit bus interface to onboard (local) DDR memory
- Optional frame buffer interface to system memory utilizing the high-speed and high-bandwidth features of the PCIE bus
- Both local and system memory can be used in combination to increase frame buffer size for realistic picture quality

SmartTile™ Memory Bus Architecture

- Optional 64-bit Double Data Rate (DDR) memory at up to 250 MHz
- Up to 256MBytes of local frame buffer
- Supports 8Mx32, 4Mx32, 2Mx32, 16Mx16, 8Mx16, 4Mx16 memory organizations
- Supports hierarchical pixel tiling
- DMA mastering with Scatter Gather
- Execute mode for Direct Textures, Video and DVD

CoolPower™ Management

- Unified software and hardware architecture for power management
- Dynamic clock gating, frequency scaling, work load balancing, and memory power-down control
- Supports Intel® defined Device Performance States (DPS)
- Optional battery optimizer under user control

In addition, the Volari XP10 provides flexible and extensive Mobile power management capabilities. All major functional blocks like RAMDAC, Video Engine can be powered down independently through register controls or pins. Mini-driver function calls are used to support ACPI.

Enhanced Bus Interface

- x16 PCI Express bus interface
- Low overhead latency communications to maximize bandwidth and throughput (2.5Gb/sec/lane)
- Hardware configuration using existing PCI implementations

THAMA™ Architecture for MPEG2 H/W Assisted DVD Playback

- THAMA™ architecture enables full DVD player support with AC3 and sub picture support
- Dedicated hardware for both Motion Compensation and Inverse Discrete Cosine Transform (IDCT), and brings advanced digital media capability such as MPEG-2/Hardware DVD playback technologies to the mass market.
- 30fps playback of 9.8Mbps VBR MPEG-2 video with 85% CPU headroom for other applications
- Hardware Alpha blending for sub-picture
- Advanced error recovery & concealment for handling poor quality video clips
- Programmable multi-tap filtering
- Supports DVD v1.0 and VCD v2.0
- Pan and Scan support

TrueVideo® Accelerator

The Volari XP10, with an integrated TrueVideo® processor and capture engine, supports dual apertures on the PCIE bus, enabling independent graphic and video data to be transported simultaneously without software involvement. TrueVideo® incorporates an edge recovery/enhancement algorithm for sharper line visibility, de-interlacing, anti-tearing, multi-tap filtering, dithering, gamma correction, and scaling operations with bilinear interpolation. The Color Space Converter (CSC) accelerates conversion for YUV 4:2:2 and 4:2:0 pixels into linear, true color RGB pixels on the fly. The additional X and Y minifiers are capable of shrinking video images, saving bus bandwidth and memory space.

- TrueVideo® provides horizontal and vertical interpolation with proprietary edge recovery scaling
- 3rd generation 5-field based motion and edge adaptive de-interlacing (beyond Bob and Weave)
- 7x2 pixel window based edge enhancement
- 3:2 pull-down with advanced film mode detection and recovery with bad edit detection
- Dual apertures for simultaneous access to graphics and video display memory areas
- Dual color space converters (CSC)
- Field rendering for interlace support on NI display
- Accelerates YUV planar format

Motion Video Capture Port

- ZV port accepting RGB, YUV 4:2:2, and YUV planar 4:2:0 formats (single 16-bit or dual 8-bit)
- CCIR 656/CCIR 601
- VBI (Intercast) interface with vertical blank interval data for transmission to CPU using PCIE Bus Mastering
- I^2S sound input capability for video capture

Advanced Mobile Power Management

- Dynamic core power rail control
- Dynamic engine clock control
- Dynamic memory clock control
- Low power cell design
- 12 GPIOs, suspend and standby modes

- Internal clock gating on each functional block
- PCIPM (H/W PCI initiated)
- ACPI 2.0 and DPMS support

Multiple/Simultaneous Display

- Different/same images with independent refresh rates on separate displays (panels, DVI, CRT or TV)
- Vertical/horizontal virtual desktop
- Microsoft® Windows Multi-Head and Multi-Monitor support
- Supports up to 4 displays

DVI Flat Panel Support

Digital Video Interface (DVI) is a low-cost, industry standard, high-speed digital link between a video source and display. High-bandwidth Digital Content Protection (HDCP) is a method of protecting the DVI data link by preventing the unauthorized copying of the downstream DVI output. The security keys for the HDCP/DVI interface are stored in an external DVI ROM, which contains an AKSV and 40 keys. A second configurable/scaleable I2C serial bus interface is incorporated into the Volari XP10 to support the DVI/HDCP/DVI ROM interfaces. The Volari XP10 supports both DVI and HDCP internally.

- One internal 24-bit single-channel TMDS with HDCP digital interface supporting up to 165MHz
- Spatial dithering for increased color depth
- Auto expansion and centering
- Auto horizontal/vertical bilinear scaling

TFT Flat Panel Support

- Dual channel 24-bit LVDS digital interfaces supporting up to WUXGA (1920x1200) resolutions or 224MHz
- Internal/external spread spectrum support
- LCD overdrive capability with frame-based adaptive algorithm
- Gamma correction for color enhancement
- Auto expansion and centering with bi-cubical filtering
- Auto horizontal/vertical bi-cubical scaling

CRT Support

- Two 10-bit DACs support high quality CRT display up to 2048x1536
- VESA™ DDC2B compliance
- Dual CRT support

TV

Volari XP10's integrated TV encoder is a TV-Out filter and digital TV encoder. It supports video devices, such as hardware, DVD, or Video CD decoders, requiring high quality TV output. The TV encoder features include:

- Supports NTSC-M, NTSC-4.43, NTSC-EIA (Japan)
- Supports PAL-B, D, G, H, I, N, M standards
- Supports Composite, S-Video, and SCART interface
- Supports Square and rectangular pixel modes
- 3 independent 10-bit DACs support simultaneous S-Video+CVBS+CRT
- Special support for DVD and Graphics resolutions: 1024x728, 848x480 (16:9 TV), 800x600, 720x576, 720x480, 640x480, 640x400, 320x240 (Mode X), and 320x200
- Interlaced/Non-interlaced operating modes
- Supports "TV-Out Filter Bypass" for interlaced video input to maintain incoming video quality
- VBI-bypass mode compatible with close captioned and Intercast broadcast
- On-chip PLL generating TV color subcarrier frequency
- MacroVision 7.01™ Copy Protection for both PAL and NTSC
- Crystal clear text and graphics display with programmable multi-tap TV deflicker filter
- Programmable multi-tap vertical UV (chroma) filter virtually eliminates dot crawl and color bleeding
- High quality horizontal/vertical Underscan
- Patent-pending auto-detection scheme of TV connection/disconnection for both Composite and S-Video
- Optional auto-switch between CRT and TV through auto-detection
- Auto power down when no TV is connected

3 Architectural & Functional Description

3.1 3D Graphic Drawing Engine

The Volari XP10 features an optimized 2x2 Pixel Pipeline, Single-Pass 3D/2D graphics incorporated with Microsoft DirectX Shader Model 2.0. The vertex and pixel shaders give graphics programmers the ability to go beyond the set graphics APIs to create never before seen custom effects and scenes. The list below further describes the other features of the XGI Volari XP10 3D graphics engine:

- 25M triangles per second (at 250 MHz)
- 30M vectors per second
- Four Pixel Shaders in parallel
- 4800M texels per second
- 600M Tri-linear/two bi-linear texture mapped pixels per second
- DirectX 9.0 Cubic mapping and bump mapping
- Floating point setup engine
- Reduced triangle setup time with user plane clipping and back face culling
- Support with 9KB on chip Cache
- Multiple primitives support: points, point sprites, lines, triangles, trapezoids, quadrilaterals, and other multiple-sided polygons
- Traditional shading and lighting, Gouraud Shading and and Specular lighting, alone or with texture mapping
- Instruction based per pixel shading with Shader Model 2.0
- Texture mapping with multiple filtering, lighting, palletting, and perspective correction choices
- Support anisotropic, and high order (up to 8x8 kernel) texture filtering
- Sub-pixel precision
- True Alpha Blending
- Up to sixteen textures in a single pass, such as traditional 2D, Cube, and 3D volume textures
- Four 4D / eight 2D textures in a single pass
- Per-pixel Fog (Vertex-based, linear, exponential, squared exponential, and ranged)
- Wide range of texture formats
- Support integer and floating (up to 32-bit) texture formats
- Support linear color format sRGB
- Support MET and MRT up to four targets
- 16/24/32-bit Z-buffer and W-buffer (with advanced bandwidth reduction technology)
- OpenGL compliant blending operations with constant fog effects & depth-cueing for photo-realistic images.

3.1.1 Shader Model 2.0

DirectX 9.0 requires Shader Model 2.0. It is a floating point processing unit to produce great quality 3D image using instruction based per pixel shading operations. Volari XP10 Pixel Shader supports both 32-bit full precision mode and 16-bit partial precision mode. Partial precision mode is running two times fast as full precision mode. This gives programmers the choice to boost the performance while retain good quality.

3.1.2 Pixel Shader

The pixel shader in the Graphic Engine has task of rendering, shading, and coloring all of the pixels that make up the virtual surfaces in a computer graphics image. The Pixel shader takes the output from the Vertex Shader and complements it by modifying the individual pixels. The Volari XP10 integrated pixel shader can perform functions like all texture blending functions, per pixel quality lighting, and fog blending.

3.1.3 Floating Point Setup Engine

The setup stage of the Graphic Engine pipeline takes the input data associated with each 3D primitive and computes the various parameters required for scan conversion. The resource intensive conversion offloads the CPU from performing the complex task and since the conversion is performed by the Setup Engine, the 3D performance is more than doubled.

3.1.4 Memory Access Converter

The Volari XP10 display engine supports both linear and non-linear (banded) memory accesses utilizing a converter. Non-linear memory accesses in the graphic engine improve performance by partitioning the color buffer into small units called tiles. The tile sizes range from 128 pixels by 32 pixels, 64 pixels by 64 pixels, and 32 pixels by 128 pixels. The converter will convert the linear memory address to the more efficient banded non-linear memory address for faster memory accesses. Once the graphic engine utilizes banded addressing mode, then all display engines reading from the color buffer will have to support banded mode. The converter is implemented at surface 0 (primary surface) and surface 1 (video overlay or Windows® based 3D.

3.1.5 Scaling and Blending

The Volari XP10's display engine combines a high performance overlay platform with high quality display images. The Scaling engine brings high expansion quality without extra bandwidth requirements. The Volari XP10 supports both bilinear horizontal and vertical interpolated scaling. Its mechanism is the same as the Video engine described in section 3.5.2 and 3.5.3.

3.1.6.1 Surface Overlay

Volari XP10 supports three surface overlays: surface 0 (primary surface), surface 1 (video overlay/windows based 3D, and surface 2 (video overlay and sub-picture/Alpha cursor). This allows for higher performance because of the decreased memory bandwidth (versus BitBLT), and per-pixel/per-surface alpha blending. The display engine divides the surface overlay platform into a Formatter and Blender. The Formatter prepares the surface data to be input into the blender using functions such as data scrambling, overlay combining (source key/destination key/window key), and data pipe timing control. The Blender blends the surfaces together (surface 0 blends with surface 1, then the resulting surface is blended with surface 2) for a more realistic computer graphics image.

3.1.6.2 Source/Destination/Window Key Overlay

The Volari XP10 can support source key (Chroma key), destination key (Color key), and window keys. Both source and destination color keying are used to mask out pixels that will not be overwritten. Envision, for example, that a destination is an on-screen background scene (on-screen area in the frame buffer) which is being displayed, and a source, which is a graphics figure being stored in the off-screen buffer area, is to be overlaid on this background scene. First, the source or graphics figure is stored inside a rectangular area using a background color key (the color of all the rectangular area except for the graphics figure itself). During the BLT process, the graphics figure (including the color-keyed rectangle) is retrieved from the off-screen buffer area by the graphics engine and then sent to the on-screen buffer area (background scene). The color key register is programmed to be the background scene color. Any on-screen pixel that matches the color key register will not be overwritten, except for the graphics figure area itself. Thus, after the BLT process, only the graphics figure area is overlaid on the background scene.

3.1.6.3 Alpha Blending

The Volari XP10 supports per-pixel or per-surface alpha blending. Alpha pre-multiplied data is also supported. The sequence of blending is surface 0 blends with surface 1, and then the resulting surface gets blended with surface 2. The formula is as follows:

RGB01 = RGB[0] x (~Alpha[1]) + RGB[1] x Alpha[1]

Result = RGB01 x (~Alpha[2]) + RGB[2] x Alpha[2]

3.1.7 Reduced Triangle Setup Time

One of the most intensive tasks in the 3D graphics pipeline is setting up the edge parameters and delta values for Z, ARGB, U, V, and perspective parameters. The Volari XP10 puts these tasks into hardware where they can be performed without needing CPU intervention and minimizes the amount of data that must be passed over the bus. The Volari XP10 optionally can detect and reject back facing polygons (back face culling); this potentially saves software a significant amount of time, especially in strip or mesh environments.

3.1.8 Gouraud Shading

The Volari XP10 can use Gouraud (smooth) shading for shading polygons for rendering lighted scenes. In Gouraud shading, colors are assigned to each vertex, and then they are blended across the face of the polygon. Since each vertex is typically associated with at least three distinct polygons, this makes the object look natural and smooth instead of faceted. This shading method can also be combined with texturing for heightened realism.

3.1.9 Specular Lighting

The Volari XP10 can add Specular Lighting in single pass to add shininess or reflections to texels emanated from other light sources. This adds a more realistic and environment responsive look to 3D objects.

3.1.10 Multiple Render Target (MRT)

Multiple render target (MRT) is an advanced rendering technology that enables graphics applications to do deferred shading. With this technique, lighting can be applied after all the geometry is rendered. This eliminates the necessity of multiple passes through the scene. In today's graphics applications, it is quite often that the underlying geometry is extremely complex and multiple light sources are used to achieve photorealistic visual effects. MRT allows users to save per-pixel data, such as depth, normal, colors, and materials, etc, in multiple buffers. For example, an application can render a color map, a normal map, and a depth map in the first pass. In the second pass, per-pixel information such as the color, the normal and the depth is fetched from those buffers. Then lighting is calculated with lighting conditions and the whole scene is lit with the color map. Note that the lighting calculation is only for visible pixels in the scene. Since MRT can effectively remove pixel shader processing for invisible pixels, it can greatly improve the GPU performance. Volari XP10 fully supports MRT as defined by Microsoft's DirectX 9.0, with some hardware limitations as follow.

All surfaces of multiple render targets have the same bit depth

All surfaces of multiple render targets have the same width and height.
Post-pixel shader operations, such as alpha blending, are only available to the first render target.

3.1.11 Floating Texture and Color Buffer

Floating texture and color buffer allow users to store floating-point data in textures and color buffers. If an application requires high precision on texture data or color values, this feature enables it to render images in high quality. For instance, in a multiple passes rendering, unclamped intermediate results can be stored in floating-point format to avoid precision lost. Another example is when using high dynamic-range images as textures. Volari XP10 supports the following floating-point texture formats and floating-point color buffer formats:

D3DFMT_R16F
D3DFMT_G16R16F
D3DFMT_R32F

3.1.12 Texture Mapping

Texture mapping is an effective tool for adding realism to a computer-generated scene. Textures on the Volari XP10 can be point-sampled, MIP-mapped nearest/linear, Linear, Bilinear, Trilinear, 16:1 Anisotropic, or flat cubic filtered, up to 8K x 8K.
In MIP-mapping, the Volari XP10 chooses from source textures of sizes decreasing by powers of two. This allows the retrieved texel to be matched with the area being textured in terms of the level of detail that is appropriate. The Volari XP10 can hold up to 8K x 8K to 1K x 1K levels of a MIP-mapped texture. For situations where point-sampling or MIP-mapping would result in excessive blockiness on the textured area due to the source texture being too low in detail, bilinear interpolation is often the best alternative.
Bilinear Interpolation (or Filtering) involves sampling 4 texels surrounding the source texel and combining their values with it to derive the pixel's final value.
Trilinear Interpolation (or Filtering) offers smoother filtering compared to Bilinear filtering eliminating sparkling effects almost completely. Trilinear filtering involves blending 2 nearest MIP Map then bilinear filtering by choosing the 4 nearest pixels from the blended MIP Map.
The Volari XP10 supports multiple (1,2,4, & 8bpp) palletized texture formats for reduced storage requirements. If look-up tables of less than 8bpp are used, multiple palettes can be stored in the Volari XP10 's on-board Texture CLUT.
Textured primitives can be either flat or smooth (Gouraud) shaded. Any type of texture (or color) can be perspective corrected. The Volari XP10 applies a division to every pixel to correct the effects of perspective projection on texture detail. This allows more realistic and solid textures without the curving or breathing sometimes seen on uncorrected texturing. This is especially important for large polygons or polygons that are particularly elongated along the Z-axis.
Filtering and lighting (shading) of textures can be combined in a single operation to provide highly realistic renderings. While each by itself is an appropriate means of rendering objects, realistic scenes require both the lighting effects provided by shading and the higher level of detail with a lower polygon count that texturing provides.

3.1.13 Anisotropic Filtering

Anisotropic filtering goes beyond Trilinear filtering. Textures are sampled along the U and V axis independently of each other as compared to Trilinear filtering which have the same sampling in the UV axis. This results in preserving texture detail, most noticeable when the texture is oblique as opposed to parallel to the view plane. Anisotropic filtering provides higher quality filtering and can be rendered in a single pass.

3.1.14 High-Order Filtering

Texture filtering is a convolution of filter kernel and texture image. Different filter kernels represent different sampling mode. For example, the point sampling's filter kernel is an impulse function, and bilinear filtering's filter kernel is a triangle (or a pyramid in 2D point of view) with width as 2 and height as 1. Unlike bilinear texture filter, which is basically a one-order, triangular low-pass filter, High-Order Filter can be any filter such as Gaussian filter, Raised Cosine filter, Lanczos filter, etc.

The Volari XP10 supports 8x8 filter kernel, and this kernel can be designed as any filter as you can. The special filter kernel not only can be set through register-setting, but can be set through a texture. By using special filter kernel, developers can do a lot of processing on textures, such as edge enhancement, special anti-aliasing, etc.

3.1.15 Multiple-Element Textures (MET)

Multiple element texture is a texture which has more than one element for each texel. Example of these elements are color, normal, and depth. Such texture can be created by the feature of Multiple Render Target in one pass. Application can use this feature to implement advanced rendering technique like deferred redenrng, and to achieve higher bpp floating textures. The Volari XP10 support this feature and is fully compatible to MS directx9.0.

3.1.16 Gamma Correction and sRGB color format

There is a special texture format: sRGB. With this kind of texture, all colors of texel are gamma-corrected. So after it is read, the color should be first do ungamma correction and then do operations in Pixel Shader. When a color is written out, it also needs do gamma correction to achieve a sRGB format. Volari XP10 supports this feature is fully compatible to MS directx9.0.

3.1.17 3D Volume Textures

Three-dimensional volume textures are three-dimensional collections of texels that can be used to paint a two-dimensional primitive. A 3D volume texture can be thought as layers of two-dimensional textures, so high resolution 3D texture consumes more memory resources than 2D texture in the same resolution, however, it can achieve effects that are either awkward or impossible with 1D or 2D texture.

One of the most often use of 3D volume textures is in medical application. 3D texture is a naturally representation of layered computed tomography (CT) or magnetic resonance imaging(MRI). Three-element texture coordinates (u, v, w) are required for each vertex of a primitive that is to be textured with a volume. By using 3D volume texture, it can make polygons just like "carve" through the volume, and give you the real feeling of 3D object, while 2D texture can only map 2D images on surface of objects. 3D volume textures are also used in special effects such as patchy fog, explosions, and so on.

The Volari XP10 fully support 3D volume texture and bilinear/trilinear filtering mode on 3D volume texture with high efficiency

3.1.18 DirectX 9.0 Cubic Mapping

The Volari XP10 is an advanced 3D accelerator that provides high-quality implementation of cubic texture maps as defined by Microsoft's DirectX 9.0 graphics standard.

Like standard circular or spherical maps, cubic maps capture views of full 360-degree environment surrounding an object. Environment maps are most commonly used to render high-quality photo-realistic specular reflections on the metallic surfaces, for instance, reflection of the showroom on the shining body of the new car.

Standard environment maps store surrounding images in a single texture, distorted at the edges to cover all sampling directions. These distortions are especially noticeable if viewing direction becomes different from the one used to create the original map. To maintain good quality, these maps have to be re-computed even for small changes in view direction, slowing down rendering performance.

One of the advantages of the cubic maps is their view-independence. Surrounding images are stored as 6 separate textures, representing 6 faces of the cube; if an object remains static, the camera can look at it from any direction using the same set of 6 faces to render reflections. Another advantage is that the process of creating each cubic map is very simple: just take six snapshots of the environment, two for each X, Y and Z-axis.

To sample cubic maps for the current pixel, an interpolated 3D vector (usually normal or reflection) is traced from the center of the cube until it intersects one of its faces. Pixels in the same triangle may sample different faces, which require resolving face addressing in real time. Fortunately, most computational blocks used for texture mapping can be used to compute vector-cube intersections, making hardware implementation less expensive.

Cubic maps are fully supported by DirectX 9.0 and can be accessed through OpenGL extensions. Currently, over 40 game titles in development rely on them to achieve photo-realistic effects. It produces good results when rendering sharp reflections on the faces of the cube at high resolutions (256*256 and above).

The Volari XP10 support for cubic maps as defined by DirectX 9.0 provides developers with new ways to increases speed of lighting calculations and to experiment with new lighting algorithms without penalizing performance.

3.1.19 Texture Compression

Textures can be stored compressed in texture memory, resulting in less PCIE bus or local frame buffer bandwidth requirements. By compressing textures, the amount of data transferred across the PCIE bus or local frame buffer can be reduced by up to 6 times. The Volari XP10 has on-chip logic that decompresses textures on the fly.

3.1.20 Sub-pixel Precision

In the Volari XP10, slopes are specified and calculated with 1/16th sub-pixel accuracy. This prevents texture coordinates from jumping unnaturally as they cross from one whole integer value to the next. Sub-pixel positioning adds 4-bits precision to the rendering process and allows higher quality natural looking images.

3.1.21 True Alpha Blending (2D/3D)

True Alpha Blending means that the Volari XP10 can determine the ARGB value of a pixel. The pixel to be written has its source pixel (in 32bpp mode) or a fixed value from the SRC Alpha register (in 16bpp mode) compared with the ARGB value already in the frame buffer. On the most basic level, this allows applications to use an effect such as transparency. It can also be used for more complex blending effects, especially using the alpha values accumulated by previous operations. The Volari XP10 supports all of the OpenGL Alpha Blending functions.

The Volari XP10 can also perform stippled alpha or *screen door transparency*, where an object is drawn on a pattern where its density corresponds to its opacity. This type of alpha is often used in games because it requires no dedicated alpha channel and no reads from the frame buffer.

3.1.22 Per-pixel Fog

The Volari XP10 can automatically produce "fog" or a "depth-cueing" effect by blending pixels on the basis of their Z-values. This type of fog is often used to hide the disconcerting effect of objects appearing abruptly as they penetrate the yon clipping plane. It is also useful for outdoor type scenes where the intention is for objects to grow less distinct in the distance. Vertex-based, linear, exponential, squared exponential, and ranged fog are all supported.

3.1.23 16/24/32-bit Z-buffer

For pixel-perfect hidden surface removal, the Volari XP10 implements the Z-buffer algorithm. It compares the incoming Z-value of individual pixels with the Z-value stored in a dedicated buffer. The Volari XP10 has a range of comparison modes for the Z value comparison. The result of the comparison determines whether or not to update the display (and the Z-buffer) or to simply discard the pixel. The Volari XP10 implementation of the Z-buffer algorithm has been further optimized to make as little impact on rasterization performance as possible. In addition, the newly developed Z bandwidth reduction technology has allowed a Z-buffer savings of greater than 50%, and more than 20% savings in total frame buffer bandwidth.

3.1.24 Texture Formats

Texture formats determine the type of data contained in a texture image. The Volari XP10 can support a wide variety of formats such as ARGB888, ARGB565/1555/4444, A8L8, UV88, UVL556, DXT0~DXT5, UYVY, YUY2, UVLX8888, UVQW, and 1/2/4/8 bpp palettized textures. All other texture formats are software preprocessed.

3.2 2D GUI Functions

The 2D engine is essential for today's operating environments such as Windows, Mac, and other GUIs. The list below gives an outline of some 2D acceleration features.

- BitBLTs
- Text Transfer
- Solid Area Fills and Patterned Area Fills
- 256 RasterOPs
- Trapezoidal Fill, solid or patterned
- Polygon Fill, solid or patterned
- Line Drawing
- Hardware Clipping
- Hardware Cursor
- Hardware Pop-up Icon
- Merge/blending sources from different windows on the fly

3.2.1 BitBLTs

The BitBLT is the single most important acceleration function for windowed GUI environments. A BitBLT is simply the movement of a block of data from one place to another, taking into account the special requirements and arrangements of the graphics memory. This function is utilized every time a window is moved, in which case the BitBLT is a simple Pixel Block Transfer. More complicated cases occur where some transformation of the source data is to occur, such as in a Color Expanded Block Transfer, where each monochromatic bit in the source is expanded to the color in the foreground or background register before being written to the display. Various operations and functions may also be used during and combined with BLTs, such as RasterOPs and Patterns (see below). Also, the source can simply be ignored and replaced with a value from the foreground (or background) register or expanded from the pattern register to the foreground (or background) color to cover a large area.

3.2.2 Text Transfer

Text Transfers are a special case of Color Expanded BLTs, specifically tuned for high-speed transfers of 8-bit wide monochromatic data (i.e., text) from system memory or the frame buffer to the display. For this special case, the alignment is to the byte boundaries, ensuring that no separate bits locate in a different byte. Each bit of the source is then expanded to the foreground or background color.

3.2.3 Solid Area Fills

When a solid rectangular background is required, such as the white area in a window, the Volari XP10 can fill it with a color from the foreground color register.

3.2.4 Patterned Area Fills

The Volari XP10 is able to fill a rectangular area with a repeated pattern from the pattern register for patterned desktops or window backgrounds.

3.2.5 256 RasterOPs

Raster Operations are logical bit wise operations involving a source, a destination, and a pattern. The source and destination are normally the source and destination rectangles for a BitBLT, while the pattern is held in a register. The Volari XP10 has a large 64x32 pattern register to avoid excessive and repeated loading.

3.2.6 Trapezoidal Fills

The trapezoidal fill is extremely useful for filling irregular areas by successive approximations. The Volari XP10 allows trapezoidal areas to be filled with either a solid color (from the foreground register) or pattern.

3.2.7 Polygon Fills

When areas can be described conveniently as polygons (objects with multiple sides), the Volari XP10 graphics engine can fill them with a solid color or a pattern.

3.2.8 Line Draws

When an application needs to draw a line between two points, the Volari XP10 will accelerate this function. The line can be solid or patterned and can have RasterOPs applied to it while it is being drawn. Aside from graphical interface applications, line drawing is also important to GUI elements, such as fine window borders and menu dividers. The Volari XP10 follows the DDA line drawing algorithm similar to the diamond exit rule, and the method identical to those used by MS and OpenGL.

3.2.9 Hardware Clipping

It is sometimes very expensive computationally for an application to determine whether or not an object to be drawn will be entirely inside a rectangular area. The clipping functions are useful to prevent the computer from calculating portions of an image that extend outside the viewable window. When an image is panned or moved in a window, the display controller is able to generate the lines or images that come into view and delete or clip those that move out of view. Thus, the display controller is updated quickly and efficiently by the clipping functions. The Volari XP10 uses a primitive rejection method for images that move outside the frustum or viewport, fast clipping in the rendering engine, and real 3D clipping against the guardband/frustum and eight user planes.

3.3 PCIE Host Interface

The host interface handles the PCIE interface protocol, performs proper signal synchronization from the CPU clock domain to the MCLK domain, and provides post write data buffering. The Volari XP10 also translates either linear addressing or bank addressing to the embedded DDR SDRAM addressing. In addition, the Volari XP10 decodes memory mapped I/O or I/O space into its internal locations. Through I/O, the host interface can communicate with video or other devices that support I^2C protocol. When the BIOS is enabled in the PCIE configuration, the host interface will deal with the ROM timing. The Volari XP10 fully supports the PCIE configuration space within the host interface block.

The Volari XP10 can translate either linear or bank addressing when addressing the embedded DDR SDRAM. The host interface can communicate with video or other devices that support I^2C protocol. The Volari XP10 supports the scatter and gathers mechanism. This feature is useful for videoconferencing applications that require two live video windows. Dual read/write apertures allow simultaneous access to display memory graphics and video areas that improve the data transfer flow dramatically.

The base address for the linear addressing windows may be set anywhere up to 4GB of linear memory space. The Volari XP10 allows a linear addressing window and the standard VGA memory space to be active at the same time. This allows greater flexibility in adapting non-linear addressed drivers to the linear addressing environment.

With PCIE Bus Mastering capability in the Volari XP10, a device driver can use either function to send graphics or decompressed video images to local display memory or to receive live video images from external video sources such as a TV-tuner, VCR, or camera. The Bus Master can improve the whole system operation parallelism by off-loading the CPU on the data transfer to or from the Volari XP10 subsystem, letting the CPU do other tasks, such as video compression or decompression and game logic for PC games.

PCIE palette snooping is handled within the host interface. Two possible snooping mechanisms are supported. One is due to a PCIE retry mechanism and the other is due to a bridge master abort mechanism. Mechanism selection is automatically determined by the video BIOS during power up.

A high speed data return buffer in the Volari XP10 is used to buffer 3D texture data to an internal texture cache for bilinear or tri-linear texture mapping or a command list stored in system memory to improve the whole 3D rendering parallelism.

Write data output FIFO in the high performance Volari XP10 can bridge asynchronous transfer from video capture data stored in local display memory to main system memory by the PCIE bus.

3.4 Hardware DVD Video Support

With an embedded hardware block for DVD video playback, the Volari XP10 combines software decoding with hardware decoding technologies for MPEG-2 video playback. Special hardware assist includes Motion Compensation (MC) and IDCT which are designed to reduce the CPU bandwidth required for processing of MPEG-2 video decoding. This partial software and partial hardware decoding technology improves MPEG-2 video quality over pure software decoding without increased cost. With Motion Compensation and IDCT, the Volari XP10 is able to play a 9.8Mbps MPEG-2 video at a frame rate of 30fps in real time.

Volari XP10 also supports Hardware Alpha-blending for sub picture. This feature allows an image to be rendered on top of another image, with a blend of images showing without any CPU utilization. This process allows the sub picture media like text for closed captioning, subtitles, buttons, menus, and credits to be brighter and vibrant without any black borders or blocks caused by the absence of alpha blending.

To handle corrupted MPEG-2 frames, Volari XP10 uses an advanced error recovery and concealment feature to skip such images.

Video streams can be displayed smoothly in different window sizes since Volari XP10 uses programmable Multi-tap Filtering to scale the Video image.

Volari XP10 DVD video is Microsoft Direct X Video Acceleration (DXVA) compatible.

3.5 Advanced Video Deinterlace and TrueVideo® Processor

Volari XP10 integrates a state of art video processing unit with XGI's proprietary deinterlacing technology and the TrueVideo Processor.

Volari XP10 uses a 5 field-based motion and edge adaptive algorithm to achieve the best deinterlace of interlaced video source. It also has movie detection and 3:2 pull down unit for seamless movie recovery.

The TrueVideo® Processor is the essential part of the Volari XP10 It accepts the data stream with YUV 4:2:2/4:1:1 or RGB formats. The block fetches the data from the off-screen memory and performs bilinearly interpolated scaling in both horizontal and vertical directions with arbitrary scaling. In addition, the TrueVideo® Processor converts the YUV pixels into linear RGB pixels on the fly. Video overlay with different color depths of graphic data is supported seamlessly. This allows the highest quality overlay of graphic data on top of video data as in pull-down menus. The DirectDraw™ acceleration increases the performance of the software playback.

The de-interlace block automatically detect the video source. If the source is interlaced video, Volari XP10 will do de-interlace. If the source is movie, Volari XP10 will do the 3:2 pull down process to recover correct movie sequence.

Video de-interlacing is the method of displaying traditional TV interlaced video on VGA-type progressive computer monitors. All existing 3D graphics controllers use the standard Bob & Weave algorithm for video de-interlacing due to its low cost of implementation, at the expense of lower picture quality.

Volari XP10 is the first 3D graphics controller that delivers superior video de-interlacing (as compared to Bob & Weave) with no increase in cost through XGI's proprietary 3rd generation 5-field based motion and edge adaptive de-interlacing algorithm. The algorithm incorporates motion detection hardware as the basis for de-interlacing.

The movie source commonly seen in DVD titles is a 30 frame per second converted through a 3:2 pull down process from the 24 frame per second movie. To correctly play this 3:2 pull down movie source on the 60 Hz progressive display devices like CRT or LCD panels, Volari XP10 hardware will dynamically detect the movie sequence and do inverse telecine or re-interlace.

3.5.1 Minifier™

The programmable input formatter accepts YUV 4:2:2 and integrates a programmable luminance interpolating filter and an independent linear X and Y minifier. The video image can be smoothed through a programmable multi-tap filter to reduce the jig-jag effect after minification. The video data can be minified to any linear fraction to save bus bandwidth or memory space and is written into the off-screen memory. Video information can also be minified before sending it to the frame buffer. Minify can be performed in both horizontal and vertical directions. The horizontal and vertical minify factor, HMF and VMF, are described as follows:

$$HMF = Hf/Hs \times 1024 - 1$$

$$VMF = Vf/Vs \times 1024 - 1$$

HMF and VMF are described by registers. In order to perform minification, the frame buffer horizontal and vertical destination counts need to be defined by the registers. Filtering is used in conjunction with minification. When minify is disabled, filtering is not recommended (It can still be enabled, but it will generate a fuzzy video image.). During the minify process, some pixels or lines will be dropped. In this case, filtering is needed to send the dropped information to the undropped pixels to preserve a better video image.

3.5.2 Bilinear H/V Interpolated Scaling

There are three types of vertical scaling: duplication, interpolation, and edge recovery interpolation. The method of vertical interpolation is to interpolate the current line pixels and previous line pixels that are stored in the line buffer. Similarly, when horizontal interpolation is enabled, each pixel is interpolated by averaging it with its adjacent pixels.

3.5.3 Zoom Block

Zooming is performed on the display path to save memory bandwidth, as compared with other methods that perform zooming before the data is sent to memory. Zooming can be performed in both horizontal and vertical directions. The horizontal and vertical zoom factor, HZF and VZF, are described as follows:

$$HZF = [SHR - 1]/[DHR - 1] \times 1024$$

$$VZF = [SVR - 1]/[DVR - 1] \times 1024$$

SHR: Source Horizontal Resolution

SVR: Source Vertical Resolution

DHR: Destination Horizontal Resolution

DVR: Destination Vertical Resolution

A bi-cubical interpolation can be done in the Zoom block to get smooth scaled up or scaled down video image. The maximum amount of zooming in each direction is 16. Interpolation can only be performed for zooming factors less than or equal to 8. Zooming factors greater than four can only be duplicated.

3.5.4 Edge Recovery

The hardware edge recovery mechanism reduces the zigzag effect due to the scaling up of oblique lines. For each pixel, the mechanism analyzes which interpolation pixel pairs to sharpen, then smoothes those edges.

3.5.5 Anti-Tearing

The hardware anti-tear mechanism prevents the tearing effect due to the frame buffer update and eases the burden of the software to flip the page. It is accomplished by removing the visual artifact that occurs when the display (read) data line and the capture (write) data line cross. The display would then show data from frame “N” on one part and from “N+1” on the next. To prevent this, the Volari XP10 allocates two capture surfaces (off-screen memory areas). While frame “N+1” is being written to one surface, it reads frame “N” from the other surface.

3.5.6 DirectDraw™

The Volari XP10 supports DirectDraw™, an interface for the Microsoft® Windows® operating system, which provides direct access to display devices while maintaining compatibility with Windows® GDI and enables world-class graphics on a Windows class PC. It provides access to the following display device-dependent benefits:

- Support for double-buffered and page flipping graphics
- Access to and control of the video cards BLTer
- Improved video playback quality through access to YUV color formats
- Improved graphics and video quality through access to image-stretching hardware
- Simultaneous access to standard and enhanced display device memory areas

The following sections describe the Volari XP10 featured functions.

3.5.6.1 Flipping Surfaces

DirectDraw™ has extended flipping surfaces to encompass more than page flipping and more than visible surface flipping. Any surface can now be constructed as a flipping surface. This has many advantages over the traditional, limited scope of page flipping.

3.5.7 Horizontal Filter

Video pixel data, after being converted to the internal format, is sent to the horizontal filtering block. The horizontal filtering can be one of several formats: bypass, 2, 3, 5, and 9 taps. The combined effect on the video pixel is to weigh that particular pixel data according to the adjacent video pixel data.

3.5.8 Video H/V Scale and Color Space Conversion

The Video Display Engine fetches the YUV 4:2:2 or RGB data from the off-screen memory and scales up with bi-linear interpolation in both horizontal and vertical directions. The on-chip hardware Color Space Conversion (CSC) accelerates conversions for the 16-bit YUV 4:2:2 data stream into the True Color RGB24 data stream and mixed with the graphic data. The graphic data and video data can be handled smoothly in different color depths with the color key support.

Color Space Conversion hardware accepts YUV422 or YUV411 as input and converts them to RGB888 for further processing. The input range of YUV data can be either CCIR601 compatible (16-235) or normalized (0-255), depending on the source format. For normalized YUV input, the conversion equations are:

$$R = Y + 1.402 * U$$
$$G = Y - 0.71414 * U - 0.34414 * V$$
$$B = Y + 1.772 * V$$

3.5.10 Dual Video Window for Videoconferencing

The Volari XP10 allows remote and local video images to be displayed simultaneously on the same screen. The different video sources can be accepted through the PCIE bus and the capture port at the same time. Performing YUV Planar, the Volari XP10 supports YUV 420 that removes the redundant video stream decoding procedures and reduces the load for the CPU while processing SW MPEGs or SW videoconferencing.

3.7 Quad Clock Synthesizer

Within the Volari XP10 core, there are four clock synthesizers. All four clocks are fully programmable and all of them have spread spectrum support.

The master clock is dedicated to the Volari XP10 engine core (MCLK1). It runs at a maximum clock rate of 250 MHz. One clock (MCLK2) is for DDR memory interface which can run up to 250 Mhz.

There are two video clocks (VCLKs) to support dual displays or multiple displays. The first video clock synthesizer is the primary pixel output clock (PCLK) which can run up to 350

MHz (recommended maximum). The second video clock (LCLK) is used to support the dual video display image with independent resolution and color depths. In addition the VCLKs also have three preset frequencies for VGA compatibility.

The Quad Clock Synthesizers are analog circuits fabricated on the Volari XP10 die. Power for the two functional circuits should be supplied from an isolated external 3.3V and GND on pins AVDD[3],[1] and AVSS[3],[1], respectively.

The Quad-Clock Synthesizer requires low-pass filters on MLF1, MLF2, VLF and LLF, and either a 14.318MHz TTL signal on XTLI or a 14.318MHz crystal on XTLI and XTLO for proper operation.

The PCLK, which is the frequency at which pixel data are sent out from the DAC to the display, is programmable. The maximum frequency of the PCLK is 350 MHz for CRT output only. The VCLK, which is the frequency at which pixel data is moved internally from the display queue, can be divided down from the PCLK. The maximum VCLK rating is 350 MHz.

3.8 VGA Controller

The high performance VGA Controller provides full hardware level VGA compatibility. At reset, the Volari XP10 is in hardware level VGA mode and all of the standard VGA subsections (Display Controller, VGA Graphics Controller, and Attribute Controller) are enabled.

3.8.1 Display Controller

The Display Controller generates the horizontal and vertical synchronization signals for flat panels and CRTs. It provides the address interface between the video memory and display screen, cursor, and underline timing of the text modes and generates refresh requests for the SDRAM refresh cycle. Auto-contrast, text-expansion, and auto-centering for image enhancement are part of the functions of this block.

3.8.2 VGA Graphics Controller

The Graphics Controller controls the graphic and the text modes for VGA compatibility. It is the interface between the video memory and the CPU during the video memory read/write operations. During system access of video memory, the Graphic Controller can perform logical operations on the memory data before reaching video memory or the system data bus. These logical operations are composed of four logical write modes and two logical read modes. The logical unit allows enhanced operations such as color compares in the read mode, individual bit masking during write modes, internal 32-bit writes in a single memory cycle, and writing to the display buffer on non-byte aligned boundaries. The Graphics Controller can also perform logic operations on memory data before it reaches the display memory or system data bus.

3.8.3 Attribute Controller

The Attribute Controller receives data from video memory and formats it for output on the display monitor, and also controls blinking, underlining, cursor insertion, and PEL panning. In text mode, 16 bits of code are divided into 8 bits of character code and 8 bits of attribute code. The character code is used as a look-up into a font table and the attribute code is used to determine character color, blinking, bold, the non-interlaced underline, etc. In graphics mode, the Attribute Controller serializes memory bits. Each output color is translated through the internal color palettes and then sent to the display DAC. It is then used as an address in the 18- or 24-bit color look-up table. The value read from the color look-up table is converted into three analog signals (R, G, and B) for driving an analog display.

For super VGA modes, the serialized data bypasses the Attribute Controller. When the video function is disabled, the Attribute Controller can also support the hardware cursor with the cursor pattern stored in the off-screen area.

3.8.4 Hardware Pop-up Icon

A four-color pop-up icon supports a user-defined pattern of 128x128x2, 64x64x2 or 32x32x2 pixel image with simple hot-key operations. The hardware pop-up icon pattern is stored in a non-visible portion of display memory and its area is separated from hardware cursor area without driver conflict.

3.8.5 Hardware Cursors

There are two hardware cursors in Volari XP10. Both of cursors support 2-10-10-10, 8-8-8-8 or 8-3-3-2 alpha cursor mode, as well as traditional 2 bit indexed mode. The size of the hardware cursor can be programmed as 64x64 or 128x128. The hardware cursor exists on a bit-plane independent of the graphics display. This allows the cursor to be used without regard to the action on the screen and eliminates the flickering caused by the alternating erasure and redrawing of a software cursor on the same plane as the graphics data. In alpha cursor modes, the cursors are alpha blending with graphics display. In the traditional 2-bpp format, the cursor can be programmed to allowing pixels of four different values to comprise the cursor: white, black, screen color, and inverted screen color.

3.9 Display Interface

The Volari XP10 provides an interface to all modern flat panels, DVI CRT, and/or CRTs without additional glue logic. Integrated LUT/DAC supports True Color display. The Volari XP10 supports Digital TV Out for input into either a XGI TVXpress™ digital encoder or a Chrontel® 700x. The standard NTSC/PAL and digital TV encoders are also supported and be displayed via an S-video connector or SCART I/F.

3.9.1 LUT/DAC

The integrated 10-bit True Color LUT/DAC operates up to 350 MHz, allowing 2048x1536 non-interlaced displays at 100Hz. For High Color or True Color, the LUT/DAC works in a bypass mode where the pixel data is the color displayed. For other color modes, the DAC works in an index mode where pixel data is the index to the color palette. The DAC module is compatible with the RS-343A and RS-170. It outputs RGB analog signals to directly drive an analog CRT monitor with a dual terminated 75 ohm coaxial cable. Composite sync and sub-carrier clock logic to drive a standard TV encoder is incorporated in the LUT/DAC.

3.9.2 LCD Flat Panel

The Volari XP10 can directly drive a variety of panels, including color active-matrix TFT panels (18- / 24- / 24+24-bit). Volari XP10 supports (24+24) double pixel per clock capable of up to 1920x1200 LCD panel resolution.
The Volari XP10 contains three LCD panel interfaces: two LVDS transmitter interfaces and one TMDS interface. The Volari 8300 contains one TMDS interface. Both the TMDS and LVDS transmitters can provide low-voltage, high-speed, DC-balanced differential data to the LCD panels.
The flat panel interface provides or supports the following functions for various panels:

- High quality scaler with cubic interpolation to either expand or expand with centering CRT display into fixed LCD panel size.
- Generates different video data formats to directly drive different types of panels
- Panel power sequence

3.9.3 Dual Display/Multiple Display

The Volari XP10 has the capability of displaying images on two different displays simultaneously exclusively for Windows®. In addition, the two displays may have different resolutions, color depths, and refresh rates. The two display devices can be LCD and CRT, LCD and DVI, or LCD and TV. An example of dual display: a user can be giving a demonstration on one display while manipulating data on another display.
Dual Display is achieved by using two separate display streams to two separate display devices (LCD-CRT, LCD-DVI, LCD-TV). One or two VCLKs can clock two display streams. In addition, the display images can be assigned to the same or different memory areas.
In addition to dual display, triple views with LCD, CRT, and TV can be achieved by utilizing three separate display streams.

3.9.4 High-bandwidth Digital Content Protection Support

Volari XP10 follows the High-bandwidth Digital Content Protection (HDCP) protocol designed by Digital Content Protection LLC. It protects the video data transmission between DVI ports of Volari XP10 and a DVI video receiver. By exchanging the HDCP authentication protocol over I2C Interface, Volari XP10 and a DVI video receiver establish a secure data transmission, which will protect the digital video data, in Down-Stream link. Via the protocol exchange, Volari XP10 and a DVI video receiver generate a shared secret value, which can not be determined by eavesdropper. The shared secret information then can be used as a symmetric key to encrypt video content transmitted over DVI/T.M.D.S. interface. An external EPROM is needed to store the key selection vector (KSV) and the confidential HDCP keys issued by Digital Content Protection LLC.

3.10 Advanced Power Management

The Volari XP10 provides flexible and extensive Advanced Power Management (APM) capabilities. Power down modes may be activated by hardware pins, hardware timers, or software control bits. DPMS is provided in either software control mode or hardware timer mode. All APM functions are easily controlled using registers for MCLK, VCLK, Oscillator, LUT/DAC, Panel, Standby, and Suspend. The entire Volari XP10 graphics system, including controller and memory, may be shut down and restored because of the complete read/write capability of all registers.

3.10.1 Power States

The Volari XP10 provides multiple states of power management. The power states, or modes, are defined as ready, standby, suspend, and off (see Tables 3-2 and 3-3). Through dedicated pins, register programming, and/or activity timers, power states can be set as follows:

3.10.1.1 Ready Mode

Ready mode is the state where the Volari XP10 is in normal operation. Functional blocks, such as DAC, can be disabled in this mode through register programming to save power.

3.10.1.2 Standby Mode

During this mode, the panel power off sequence is activated, the video clock is stopped and the video display is inactive. The internal memory timing sequencer responds only to DRAM refreshes and CPU accesses. The CPU can still access display memory, I/O registers, and the LUT/DAC. The standby state can be entered or exited through control pins, register programming, auto detection of the display memory access, and/or keyboard access. When in simultaneous display mode, the chip activates DPMS off mode with Standby Mode.

3.10.1.3 Suspend Mode

This mode is the lowest power consumption state in the Volari XP10 before losing main power. During this state, the panel power off sequence will be activated and both display memory and video clock are shut off. The CPU can no longer access the display memory and the LUT/DAC is in power-down mode. This mode can be activated through a pin or register. When activated by a register, the software will continue to have access to all internal registers. When activated by a pin, the host interface is also powered down. Only the pin can bring it out of the pin-suspend mode. When deactivated, the suspend pin is not delayed by the suspend timer.
When in simultaneous display mode, the chip will activate DPMS off mode with the suspend mode.

3.10.1.4 Off Mode (0V Suspend without SDRAM Refresh)

Also known as zero-volt suspend mode or machine powered off mode, this mode allows maximum power savings for long periods. The system can save the complete state of the video subsystem and restore the state later during warm up. The Volari XP10 allows all registers to be read and written to support this mode.

3.10.2 Activating Power Modes

This section describes the major activating and deactivating power modes.

3.10.2.1 Controlling Standby Mode

Standby mode is activated by any combination of pin, register bit, and timer settings, depending on which triggering mechanisms are enabled. Deactivating the standby mode is achieved by deactivating all sources that activate standby.

3.10.2.2 Controlling Suspend Mode

Suspend mode is activated by any combination of pin or register bit settings, depending on which triggering mechanisms are software enabled. If suspend mode was caused by the pin, a deactivated suspend pin will deactivate suspend mode or the software can disable use of the suspend pin by registers to deactivate suspend mode. If suspend mode was caused by the suspend register bit, only clearing this bit will deactivate suspend mode.

Table 3-2. Power State Summary

State	Display	I/O Access	Mem Access	DRAM Refresh	Chip VCC	VCLK	MCLK	DAC
Ready	on	on	on	on	on	on	on	on
Standby	off	on	on	on	on	on	on	off
Software Suspend	off	on	off	on	on	off	off	off
Hardware Suspend	off	off	off	on	on	off	off	off
Off	off	off	off	off	off	off	off	off

3.10.3 Panel Power Sequencing

The Volari XP10 starts the panel power sequencing. ENBKLT timing is included in the sequence to control either the Backlighting or Panel Enable (on some panels). The active polarity of ENBKLT is controlled by a register bit. The power up and power down sequencing is as follows:

3.10.3.1 Power-Up Sequence

ENPVDD active after one vertical timing period, the panel interface signal is active.

After 3 vertical timing periods, ENPVEE active.
After one vertical timing period, ENBKLT active.

3.10.3.2 Power-Down-Sequence

ENBKLT inactive.
After one vertical timing period, ENPVEE inactive.
After 3 vertical timing periods, panel interface signals are tri-stated.
After one vertical timing period, ENPVDD inactive.

State	HSYNC	VSYNC	DAC
Ready	on	on	on
Standby	off	on	off
Suspend	on	off	off
Off	off	off	off

Table 3-3. DPMS State Summary with CRT Only and Hardware Timers

3.11 Power-up Configuration

During power-on and system reset, the following signals are latched on the rising edge of the reset signal: $CAPD_0$ –$CAPD_{15}$, CAPVS, and $GPIO_0$-$GPIO_{11}$. If a logic HIGH is required depending on external configuration, the signal must not be pulled high. This would effect normal operation. Instead the signal is allowed to float high. If a logic LOW is required, the signal must be tied to ground through a 4.7K resistor. This resistor will not effect normal operation due to its high resistance.

3.12 Timer Modes

The Volari XP10 offers a flexible timer for use in Standby, and DPMS. The timer may be programmed to monitor a programmable inactivity period based on any combination of pin, memory accesses, and keyboard accesses.

The timer period is programmable to count ½ minute or 1 to 15 minutes in 1-minute intervals. If there is no activity on the monitored logic for the selected period, the time-out status is set and reflected on a readable bit.

The time-out state is reset in different ways, depending on the use of the timer. In all configurations, selecting no monitored activity (setting the Timer Activity Monitor select bits to 000) will reset the time-out state. In addition, if the timer is used for Standby or DPMS, activity on the monitored logic can reset the time-out state.

3.13 EEPROM/DDC Support

The Volari XP10 has twelve (12) GPIO pins, any two of which may be used to interface with any two-wire 3V or 5V serial interface, such as the VESA™ DDC standard or the SST™ M25P05A™. Monitors equipped with DDC or systems with this EEPROM can be used to store system information, such as monitor type, timing, and serial number.
The DDC INTERFACE/EEPROM requires two pins to communicate with the Volari XP10 controller: SCL (serial clock) and SDA (serial data). To ensure proper operation, the timing specified by the DDC INTERFACE/EEPROM must be met; the programmer controls this. The functional timing for the DDC INTERFACE/EEPROM interface is shown in Figure 3-6.

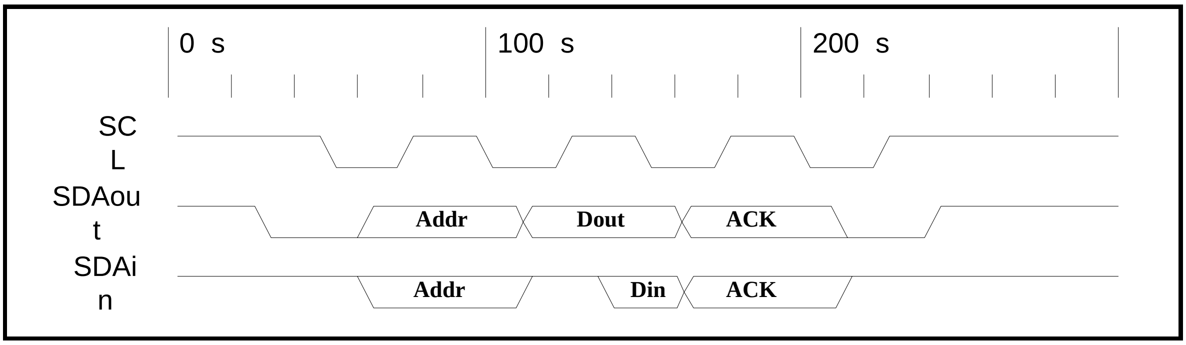

Figure 3-6 DDC INTERFACE/EEPROM Timing

3.13.1 ROM Pin Description and Connection

The pin definitions of the serial flash ROMs that Volari XP10 can support are as follows:

Pin C is the serial clock input for the serial flash ROM. Pin D is the serial data input to the ROM. Pin Q is the serial data output from the ROM. Pin S is the ROM chip select. Pin W is for the ROM write protect function. Pin Hold is used to hold the operation. Vcc is the supply voltage pin. Vss is the ground pin. Chip select (S), write protect (W) and Hold are all active low signals. For Volari XP10, write protect and hold will not used so both pins will be tied to a high logical level. The connection of the serial flash ROM is shown in Table 3-4.

C	GPIO5 (Volari XP10 pin Y3, a 33 Ohm series resistor from Volari XP10 is needed)
D	GPIO4 (Volari XP10 pin Y4)
Q	GPIO3 (Volari XP10 pin AA1)
S	ROMCS# (Volari XP10 pin Y2, a 4.7K Ohm pull up resistor tied to 3.3V is recommended)
W	High logical level (Needs a 4.7K Ohm pull up resistor tied to 3.3V)
HOLD	High logical level (Needs a 4.7K Ohm pull up resistor tied to 3.3V)
Vcc	Power(3.0-3.6V)
Vss	Ground

μ μ

Table 3-4 Pin connection of serial flash ROM

At power-up and power-down, the serial flash ROM must not be selected (that is chip select must follow the voltage applied on Vcc). A pull up resistor on chip select is recommended to insure safe and proper power up and power down.

3.13.2 Address Allocation

A 256K byte address is reserved by Volari XP10 for the serial flash ROM. As a result, the maximum serial flash ROM is a 256K byte serial flash ROM. Let us use Base to mean the base address for serial flash ROM. In the case that only a 64K byte serial flash ROM is used. The address Base+Offset, Base+10000h+Offset, Base+20000h+Offset, Base+30000h+Offset have the same effect to the serial flash ROM. However, it is highly recommended that you use only address Base - Base+0FFFFh when only a 64K-byte serial flash ROM is used. As a result, the address allocation for serial flash ROM with different size is shown in Table 3-6.

Size (Byte)	Address
64K	Base – Base + 0FFFFh
128K	Base – Base + 1FFFFh
256K	Base – Base + 3FFFFh

Table 3-6 Address allocation

3.14 ZV/Capture Port

The function of the video capture is to receive digital video and sound data streams from video sources such as TV decoders, VCR, or MPEG 2, and store them in the frame buffer. The stored video/sound can either be displayed together with a graphics image or fetched by the CPU to perform further processing such as video editing or hard disk storage. For video capture, there are two possible data paths. One path is from the capture port and the other one is from the PCIE host interface. They are merged together before any video data processing, buffered in the video FIFO, and then sent to the frame buffer. The major function blocks are the Video Capture Processing and the dual overlay window Video Display Processing. The Video inputs can either be from the PCIE bus or the Video Capture Port. This video engine accepts 8-bit and 16-bit video data bus width of standard video source input formats (CCIR656/CCIR601), providing a glueless interface with various decoders. The programmable input formatter accepts various YUV/RGB formats, integrates a programmable luminance interpolating filter and an independent linear X and Y minifier.

The Volari XP10 allows a PC card to write video data directly into the frame buffer. Three different controllable sizes and shapes of images can be overlaid with a mixed color depth up to the true color through the adjustments of the window key, color key and chroma key. The video image is stored in off-screen memory and the Video Display Processing block fetches the data from the off-screen memory and performs scaling operations with linear interpolation in both X and Y directions. In addition, it converts the YUV pixels into linear RGB888 pixels on the fly. The video image is overlaid with different color depths of the graphic data according to the color key. The hardware scaling and color space conversion accelerates the DirectDraw™ applications and accelerates the frame rate of the software playback.

The Volari XP10 is able to handle Intercast (VBI, Vertical Blanking Interval) supporting TV broadcasting with embedded web-site information.

3.15 External Memory Configuration

Volari XP10 can support the following specific types of market production DDR memory on both 64-bit port (Table 3-10).

a. 2MX32 b. 4MX32 c. 8MX32 d. 4MX16 e. 8MX16 f. 16MX16

DDR	16Mx16	2Mx32	4Mx32	8Mx32	4Mx16	8Mx16
A12 ~ A0	CS1_,MA11 ~ MA0	MA10 ~ MA0	MA11 ~ MA0	MA11 ~ MA0	MA11 ~ MA0	MA11 ~ MA0
DQ31 ~ DQ0	MD63 ~ MD48 MD47 ~ MD32 MD31 ~ MD16 MD15 ~ MD0	MD63 ~ MD32 MD31 ~ MD0	MD63 ~ MD32 MD31 ~ MD0	MD63 ~ MD32 MD31 ~ MD0	MD63 ~ MD48 MD47 ~ MD32 MD31 ~ MD16 MD15 ~ MD0	MD63 ~ MD48 MD47 ~ MD32 MD31 ~ MD16 MD15 ~ MD0
DM3 ~ DM0	DQM7 ~ DQM6 DQM5 ~ DQM4 DQM3 ~ DQM2 DQM1 ~ DQM0	DQM7 ~ DQM4 DQM3 ~ DQM0	DQM7 ~ DQM4 DQM3 ~ DQM0	DQM7 ~ DQM4 DQM3 ~ DQM0	DQM7 ~ DQM6 DQM5 ~ DQM4 DQM3 ~ DQM2 DQM1 ~ DQM0	DQM7 ~ DQM6 DQM5 ~ DQM4 DQM3 ~ DQM2 DQM1 ~ DQM0
RAS#	RAS#	RAS#	RAS#	RAS#	RAS#	RAS#
CAS#	CAS#	CAS#	CAS#	CAS#	CAS#	CAS#
WE#	WE#	WE#	WE#	WE#	WE#	WE#
CKE	CKE	CKE	CKE	CKE	CKE	CKE
BA1	BA1	MA11	BA1	BA1	BA1	BA1
BA0	BA0	BA0	BA0	BA0	BA0	BA0
DQS or DQS3 ~ DQS0 (BGA)	DQS7 ~ DQS4 DQS3 ~ DQS0	DQS5/DQS2 or DQS7 ~ DQS4 DQS3 ~ DQS0	DQS5/DQS2 or DQS7 ~ DQS4 DQS3 ~ DQS0	DQS5/DQS2 or DQS7 ~ DQS4 DQS3 ~ DQS0	DQS7 ~ DQS4 DQS3 ~ DQS0	DQS7 ~ DQS4 DQS3 ~ DQS0

Table 3-10 DDR Memory Configurations

Important notes:

In DDR connection topology, special care needs to be taken to guarantee similar trace length between the Volari XP10 chip and external memory modules on all signals.

In DDR connection topology, skew between all DQs should be well controlled. DDR I/F are very sensitive to these skews. Practically we need strict control to keep all the trace lengths of DQs, DMs and DQSs to be as equal as possible. Also, they should be grouped together when routed.

In DDR connection topology, DQS signals should be routed in the middle of a group of signals together with DQs. Skew needs to be well controlled between different DQS signals and is recommended to be less than 100ps.

3.16 TV Out

Volari XP10 has an integrated TV encoder that provides a flexible digital video interface, which supports a wide range of video formats and resolutions for various graphics, DVD, and VideoCD applications. It incorporates a high quality, programmable multi-tap TV Deflicker and UV Chroma filter. It supports both the NTSC and the PAL formats. It supports S-Video and Composite video (RCA jack) output interfaces. It supports various Master and Slave mode configurations to work with video sources such as TV decoder, DVD/VideoCD decoder or VGA controller.

The TV encoder supports a wide range of resolutions for graphics, DVD, and VideoCD decoders output to display on TV. It supports 1024x768, 800x600, 720x576, 720x480, 640x480, 640x200, 320x240 (Mode X), and 320x200 and enables 16:9 TV output from special decoders to display on regular TV. It also supports 848x480 resolution. It features an enhanced version of XGI's renowned ClearTV® technology with programmable multi-tap TV Deflicker filter and UV Chroma filter to display superb text, graphics, and video, while substantially reducing dot crawl and color bleeding, which can be found in typical TV encoders.

The high quality scaler supports arbitrary and independent, Horizontal and Vertical scaling factors. It fully supports Horizontal and Vertical Underscan for both video playback and Windows operation on TV. The high quality arbitrary horizontal and vertical scaling enables various resolutions to fit nicely into TV resolutions.

3.16.1 TV Connect/Disconnect Auto-Detection

The encoder provides a sophisticated auto-detection mechanism for TV connection / disconnection which meets or exceeds PC98 specification. The mechanism utilizes the load voltage difference, 75 ohms versus 37.5 ohms, between TV disconnected and connected. The detection is done by sending out known output voltages through each DAC. The voltage is set to 0.31V (if TV is disconnected) and 0.15V (if TV is connected) and the corresponding sensor for each DAC will compare the actual voltage with the internal voltage reference

3.16.2 System boot up Auto-Detection

During boot up, software will set the SSDATA register to a value equivalent to 0.31V (if TV is disconnected). It will then pulse the SSPULSE register bit high momentarily (set to H,

3.16.3 Normal TV output operation Auto-Detection

During normal operation, the typical detection scheme will cause TV screen noise. The encoder incorporates a patent-pending scheme to automatically detect a TV connection on every vertical blanking period without causing any TV screen noise.