3D*labs*®

Software Release Note

Windows 95 Display Driver for *PERMEDIA*® *& GLINT*® *500TX*

Introduction

This note describes the Windows 95 Display Driver for the PERMEDIA and GLINT 500TX reference boards, and how to install. The display driver includes support for DirectDraw, Direct3D, OpenGL and Heidi.

Release 4.03.00.2101-0076   Issue 9

Release Note 121.04.04

Change History

Issue	Date	Change
9	30-Jan-97	Updated for Release 4.03.00.2101-0076

3Dlabs Inc.
181 Metro Drive, Suite 520,

San Jose, CA 95110
United States

Tel: (408) 436 3455
Fax: (408) 436 3458
WWW: http://www.3Dlabs.com
Email: info@3Dlabs.com

3Dlabs Ltd.
Meadlake Place
Thorpe Lea Road.   Egham
Surrey, TW20 8HE
United Kingdom

Tel: +44 (0) 1784 470555
Fax: +44 (0) 1784 470699

Introduction

This document describes the Windows 95 Display Driver for the PERMEDIA and GLINT 500TX reference boards. The driver includes support for Direct3D, DirectDraw, OpenGL and Heidi.

Release Identity

Once the driver has been installed and the machine rebooted, the display driver release number can be determined by inspecting the Driver page of the Device Manager property sheets for the installed board. Note you need to click on the glint.drv file to display the version.

Prerequisites

- Windows 95 (Final Release) or OEMSR2
- Pentium processor or later.
- PERMEDIA, PERMEDIA NT or GLINT 500TX reference board.

Important Information

This driver provides support for PERMEDIA and GLINT 500TX and is compatible with DirectX 2 and 3. Further releases of this driver can also be obtained from the 3Dlabs web site:

http://www.3dlabs.com

Technical Support Contacts:

- Email: support@3Dlabs.com
- Tel: 408 436 3455, +44 (0) 1784 470 555
- Fax: 408 436 3458, +44 (0) 1784 470 699

Installation

NOTE: If you are upgrading from the 4.03.00.2101-0071 Release Candidate 2 then you will be warned that three files have a for a different language. This was a bug in RC2 and you should reply NO when asked if you want to keep the old file in all three cases, so that Windows 95 DOES install the files from the final release.

Fitting a new card with Windows 95 already installed:

1. Please refer to the original instructions provided with your PC for information on how to replace the graphics board. Remember to use anti-static precautions at all times to protect the hardware from static electricity. If in any doubt about the installation consult a qualified PC engineer.

2. Once your original graphics board has been removed, replace it with the 3Dlabs card. Ensure that it is firmly seated and secured to the machine using the appropriate fastenings. Note that the PERMEDIA, PERMEDIA NT and GLINT 500TX boards are VGA enabled, and there should be no other VGA in the machine once the card has been fitted. If your PC motherboard has built-in VGA it should be disabled. Again refer to the original equipment manufacturer's manuals for details on how to do this.

3. Fit the PC casing, switch on and boot Windows 95.

4. After the login procedure, your new card should be detected and Windows will ask for the correct drivers. Insert the floppy disk containing the 3Dlabs drivers and install them. Upon completion, Windows will reboot using the new card. PERMEDIA NT and GLINT 500TX boards have more than one PCI device on the board, which means there may be some extra reboots as Windows 95 detects the other devices present. You should start in at least a 16Bit mode for Direct3D acceleration.

Installing Windows 95 with a 3Dlabs board in the PC:

1. Install Windows 95 in the usual way. When installation has completed, Windows 95 should be booted on the VGA. It will not have recognised the 3Dlabs graphics card on installation.

2. Boot Windows 95 into safe mode. This can be done by pressing "F8" just after the machine has rebooted, when the message "Starting Windows 95…" appears.

3. Open the device manager. To do this right-click on the "My computer" icon and choose "Properties". Once the device manager is open, select the "Device Manager" tab at the top of the property sheet.

4. Look at the list of devices on your machine. If there are any listed under "3D Accelerators", such as "Delta Geometry Processor" or "GLINT 500TX", remove them. Next look at the "Display Adapters" section. All the devices listed in this section should be removed. Finally look in the "Other" section, and remove all the devices in this section.

5. Close the device manager and reboot the machine.

6. After the login procedure, your new card should be detected and Windows will ask for the correct drivers. Insert the floppy disk containing the 3Dlabs drivers and install them. Upon completion, Windows will reboot using the new card. PERMEDIA NT and GLINT 500TX boards have more than one PCI device on the board, which means there may be some extra reboots as Windows 95 detects the other devices present. You should start in at least a 16Bit mode for Direct3D acceleration.

Updating existing drivers:

1. Whilst running Windows95, open the control panel and choose "Display".

2. Navigate to the "Settings" tab at the top of the panel, and choose "Change Display Type" or "Advanced Properties". From here you can change the driver. Select "Change" and "Have Disk". Windows will now prompt for the installation floppy. Follow the usual procedure to install the drivers.

3. Windows will now reboot your machine using the new drivers.

Updating your monitor type:

The 3Dlabs drivers will try to use the optimal refresh rate that the monitor can handle. For this reason it is important to correctly set up the monitor type to match your display. If the monitor is over-specified in the Windows95 control panel settings, it is possible that the drivers will attempt to use a refresh rate that is beyond the capabilities of your monitor. To correct this please ensure that you have correctly specified your monitor type. The monitor settings can be found in the "Display" settings in the control panel.

The above procedures should work correctly for the 3Dlabs reference boards. If you experience problems with installing the drivers, and you have previously installed 3Dlabs drivers on the machine, there may be a conflict. Follow the steps below to resolve it:

1. Boot Windows 95 into safe mode. This can be done by pressing "F8" just after the machine has rebooted, when the message "Starting Windows 95…" appears.

2. Open the device manager. To do this right-click on the "My computer" icon and choose "Properties". Once the device manager is open, select the "Device Manager" tab at the top of the property sheet.

3. Look at the list of devices on your machine. If there are any listed under "3D Accelerators", such as "Delta Geometry Processor" or "GLINT 500TX", remove them. Next look at the "Display Adapters" section. If any 3Dlabs devices, e.g. "PERMEDIA" or "GLINT Compatible VGA", are listed remove them.

4. Close the device manager.

5. Open explorer and navigate to the "Windows" directory. In here you will find a hidden directory called "Inf". Some releases of Windows 95 use a different structure to others inside this directory, so some of the following files may not be present on your release. Find any file names in this directory that begin with "oem". Examine each of the OEM files, if you have them, in turn by double clicking to load them into Notepad. If any of the files contain references to 3Dlabs, PERMEDIA, or GLINT, delete them. You also need to delete the file called 3dlabs.inf from this directory. Look in the "Other" directory, if you have one, and delete any files whose name starts with 3Dlabs.

6. Windows is now clean of all references to 3Dlabs hardware. Reboot the machine normally and auto-detection of the hardware will begin again. Remember to supply the correct driver disk when it is requested.

Resolutions and Color Depths

The following modes are supported by this display driver release. This driver does not accelerate Direct3D in 256 color modes, although it does accelerate DirectDraw in these modes. For hardware acceleration, please ensure you are running in a High Color or True Color display mode. If a Direct3D sample application is started in 256 color mode, it will attempt to change color depth to High Color to work with the card. Note that Direct3D rendering is not supported at resolutions greater than 1280x1024.

The resolutions supported for Windows display are shown below. Note that the DirectDraw driver will support resolutions different to the Windows desktop (i.e. 512 x 384 on GLINT 500TX, 512 x 384, 320 x 240, 320 x 200 on the PERMEDIA and PERMEDIA NT). The memory requirements will differ depending upon the applications you run. If the Direct3D/DirectDraw applications complain that there is not enough memory, try reducing screen resolution.

For most modes, all refresh rates between 60Hz and 100Hz are supported by the driver, although the actual maximum resolution and refresh rate that you can achieve will depend on your monitor setting. The following tables show what resolutions are available and the Maximum refresh rate achievable in each mode.

PERMEDIA / PERMEDIA NT

Resolution	256	High Color (32768 colors)	True Color
640x400	100Hz	100Hz	100Hz
640x480	100Hz	100Hz	100Hz
800x600	100Hz	100Hz	100Hz
1024x768	100Hz	100Hz	90Hz
1152x870	100Hz	100Hz	69Hz
1280x1024	100Hz	100Hz	60Hz
1600x1200	100Hz	Not Supported	Not Supported

GLINT 500TX:

Resolution	256	High Color (32768 colors)	True Color
640x400	100Hz	100Hz	100Hz
640x480	100Hz	100Hz	100Hz
800x600	100Hz	100Hz	100Hz
1024x768	100Hz	100Hz	100Hz
1152x870	100Hz	100Hz	100Hz
1280x1024	100Hz	100Hz	Not Supported
1600x1200	100Hz	100Hz	Not Supported

Direct Draw Support

The driver contains support for hardware optimized DirectDraw functionality, which is used by DirectDraw games as well as by the Direct3D HAL. The driver is compatible with DirectX 2, and 3. Please obtain a release of this software if you require DirectDraw / Direct3D acceleration.

Direct3D

The driver contains a Direct3D HAL driver. This enables hardware acceleration of Direct3D applications and is compatible with DirectX 2 and DirectX 3. Some features are not available in this release and these are documented below:

Direct3D HAL - Current limitations.

- Spans and Single Points are not supported in this release of the drivers.

- Alpha Tests through Direct3D are not supported in this release. Note that Chroma tests are supported.

- YUV texturing is not supported in this release.

Known Anomalies and Restrictions

The following errors have been noted during testing of this software release:

- Monochrome VGA boards cannot be used in conjunction with PERMEDIA NT and GLINT 500TX reference boards

- Occasionally when switching from Windows 95 to a full screen DOS box and back, Offscreen Bitmaps get corrupted. When this happens, changing mode often cures the problem. If it doesn't, you will need to reboot the machine. To avoid this problem altogether you can disable Offscreen Bitmaps in the PERMEDIA Setup tab of the Display Control Panel. However, this reduces the performance of the Display Driver.

- Laplink does not function correctly using the remote control option, due to application bugs. Old versions of the application will not run remotely at all, where newer versions run with many black icons and bitmaps.

New Driver Features

This release of the Windows95 DirectX driver has several features to accelerate various aspects of DirectDraw and Direct3D, particularly in the PERMEDIA case.

New PERMEDIA Features:

320 x 240, 320 x 200 Resolutions:

Ultra-Low resolution formats for legacy games originating on console based systems.

Video Acceleration:

The PERMEDIA driver can now take a YUV format DirectDraw surface and blit it to the desktop whilst doing the color conversion to RGB. This gives high quality, fast decoding of the YUV format data that is generated by MPEG movie decoders.

Stretch Blits / Mirror Blits:

Stretching is now enabled for all blitting operations and can be combined with mirroring.

ColorKey Blits / Tests:

The driver can now apply ColorKey tests during DirectDraw blits and can also apply ColorKey tests to textured surfaces. This greatly accelerates applications that require ColorKey blits using DirectDraw surfaces and allows transparent texture effects.

Texture Compression:

PERMEDIA supports a proprietary texture compression scheme when using the Direct3D ALLOC_ON_LOAD mechanism. The scheme, known as 'Patching', greatly accelerates textured rendering with no loss in quality. Three different modes of operation are supported in the Direct3D driver - see below in "Control Panel Applets".

System Memory -> Video Memory Blits:

DirectX 2 onwards allows acceleration of Blits from System memory to video memory. This is now accelerated by the PERMEDIA driver for standard Blits.

New PERMEDIA & GLINT 500TX Features:

512 x 384 Resolution:

Many games prefer this mode with its good quality/resolution trade-off. It also saves on valuable card memory.

Alpha Blending / Stippled Alpha Blending:

Both drivers have support for the two methods of blending supported in Direct3D. Alpha Blending gives true-transparency effects and Stippled Alpha Blending can be used for screen door transparency and shadow effects.

Mono Color Model support:

Both drivers now support the mono color model as well as the RGB model. This mode can be used to light textures using monochromatic lights. Note that this will give a slight speed improvement on boards without a Delta chip.

Z Visibility Testing:

Both drivers now provide Z Visibility tests. This allows objects with many thousands of polygons to be quickly tested for visibility. Note that retained mode will automatically use this feature to accelerate rendering.

OpenGL

The driver provides support for OpenGL version 1.1. This library is a part of the OEM SR2 Windows95 release (the self-extracting executable opengl95.exe available from Microsoft and included with the 3Dlabs' drivers provides all the required libraries for those using the original retail release of Windows95). This enables hardware acceleration of OpenGL applications and requires installation of DirectX 2 (or later) on which it depends for pixel buffer allocation (back buffer, z-buffer, texture storage etc). A dialog box warns the user if there is insufficient memory to allocate a full-screen back buffer (for double-buffering) and z-buffer which are the minimum requirements for hardware acceleration (in this case the generic Microsoft software renderer is invoked). Some features are not available in this release and are documented below.

OpenGL Known Anomalies and Restrictions

- The DirectX surfaces upon which OpenGL apps depend will be lost during a display resolution mode change, although the application may continue to render incorrectly (or possibly crash). Any running applications should be shutdown before changing mode.

- True colour acceleration at 1024 x 768 on a PERMEDIA 8MB board requires the 'Rectangular Allocator' check box in the PERMEDIA Setup display tab to be unticked. This informs DirectDraw to use the more memory efficient linear memory allocator rather than the default rectangular allocator. Note that the GLINT 500TX ignores this setting (it always uses a rectangular allocator).

- The depth clear conformance test will fail unless the 'Disable Fast Clear Planes' check box in the GLINT OpenGL display tab is checked. The problem is with the readback of the Depth Buffer, not its clearing. If this variable is set then this disables the use of 3Dlabs proprietary fast clear mechanism that allows the depth(Z) buffer to be cleared up to 16 times more quickly than normal. Typically this becomes significant for animation rates of 10Hz or higher in large windows.

- The conformance tests that involve mip-map texture filtering (miplin.c, mipsel.c and texbc.c) will fail on GLINT 500 TX if the DisableMipMap check box in the GLINT control panel applet is set. The default state is to have this set because this provides the best trade-off between image quality and performance for the majority of applications.

- When using the glaux library supplied by Microsoft, specifying that you require alpha planes in the visual is not satisfied by requesting a visual type of AUX_RGBA as opposed to AUX_RGB when calling auxInitDisplayMode(type). In these instances the hardware accelerated visual that will be returned in some modes may not have alpha planes. This is because the display driver exports visuals without alpha planes before those that do. This problem can be resolved in two ways: Firstly, if you have the source code, then when specifying the visual type you can OR in AUX_ALPHA, along with AUX_RGB (for Example auxInitDisplayMode(AUX_RGB | AUX_ALPHA)). Secondly, if source is not available, the following registry variable can be set to 1, which enables the visuals with alpha planes to be selected first.

      3DExtensions.ExportAlpha

  in:

      HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\glint\Device0

  Setting this variable will result in a decrease in the performance of some applications as the driver must perform additional setup calculations for GLINT to cater for the Alpha value as well as R, G, and B.

- In some cases, there is confusion over the meaning of the PFD_SUPPORT_GDI bit in the dwFlags field of the PIXELFORMAT descriptor. 3Dlabs have seen applications (for instance Open Inventor) which incorrectly assume that if this flag is set, rendering to bitmaps is supported by the visual. The Installable Client Driver does not support bitmap rendering so these applications fail. To enable these applications to work the exporting of PFD_SUPPORT_GDI can be disabled by clearing the relevant settings in the OpenGL tab. The applications will then choose a Generic pixel format so using unaccelerated software rendering to draw to bitmaps.

  By default PFD_SUPPORT_GDI is set to TRUE for single buffer formats and FALSE for double buffered formats.

Note under Windows, Generic pixel formats that support double buffering and rendering via GDI are mutually exclusive. This is because GDI does not have the ability to render to the backbuffer. 3Dlabs have therefore chosen to set the default for double buffering, so as to be in line with the Microsoft implementation. However with care GDI rendering and double buffering may be mixed, so the latter registry variable will cause PFD_SUPPORT_GDI to be exported by double buffer formats, should an application benefit from this added functionality.

- When running multi-threaded applications it may be necessary to disable the use of the fast clear planes by setting the environment variable GLINT_DONT_USE_FCP to TRUE, or by checking the corresponding box in the GLINT control panel applet. This issue arises when more than one context is being used to render to the same window (e.g. Windows NT 4.0 OpenGL pipes screen-saver with multiple option selected). If this variable is set then this disables the use of 3Dlabs proprietary fast clear mechanism that allows the depth(Z) buffer to be cleared up to 16 times more quickly than normal. Typically this becomes significant for animation rates of 10Hz or higher in large windows.

- The following OpenGL version 1.1 API calls are not currently implemented for this release:

  glIndexub

   glIndexubv

  glPolygonOffset

  glCopyTexImage1D

  glCopyTexImage2D

  glCopyTexSubImage1D

  glCopyTexSubImage2D

  glTexSubImage1D

  glTexSubImage2D

- Support for texture memory caching is not available in this release.

- Support for the Overlay planes extension is not provided in this release.

OpenGL Texturing & Extensions

Efficient use of multiple textures

OpenGL applications that wish to render primitives with multiple texture maps will achieve much higher performance by avoiding the invoking of the different textures in immediate mode. There are two alternative options for efficient switching between multiple textures.

The first and much preferred option is to use the OpenGL texture object functionality. This is available with 3Dlabs Installable Client Driver as part of the standard API for OpenGL version 1.1 on Windows 95. Texture objects are fully editable and may have their images and parameters altered at any time (unlike the use of textures in display lists). Details on texture object functionality is available in the OpenGL 1.1 specification (Appendix E) and also in Appendix D of the 3Dlabs OpenGL Extensions document. The performance gain using this approach will benefit performance for both the GLINT 500TX and PERMEDIA.

The second option is to define each texture (or array of mip map resolutions) within a display list. Switching between different textures is then achieved by referencing the appropriate display list. Since display lists are not editable in OpenGL, the OpenGL implementation is able to cache texture data defined within a display list. This caching cannot be performed when a texture is invoked in immediate mode since the application in this case is at liberty to have changed the texture data since any previous reference.

Texture Memory Usage

For the GLINT 500TX and PERMEDIA, texture data is stored in the local buffer memory on the graphics card. The memory available for textures is therefore constrained by the local buffer memory available. It is also constrained by the amount of local buffer memory already consumed for the depth buffer, stencil buffer, etc., which varies according to the current display resolution in use. I.e. there is more memory available for textures when the display resolution (and therefore the size of the depth buffer, stencil buffer) is lowered.

In the current Windows 95 3Dlabs OpenGL release (ver 1.1.16), if the condition is reached where there is insufficient local buffer memory to load a new texture then the OpenGL texture download will not succeed and will set the error code GL_OUT_OF_MEMORY. Textured primitives that expected to use this texture will not be rendered correctly.

Texture Filter Modes

The default texture minification filtering for OpenGL involves mip-map filtering. This gives good textured rendering quality but at the cost of low performance. Much higher performance can be obtained by changing the default texture filtering such that the minification and magnification filtering modes are the SAME. Setting them to GL_LINEAR gives good quality bilinear filtering and improved performance. Setting both modes to GL_NEAREST will give nearest neighbour filtering and the fastest possible performance.

Only linear and nearest neighbour texture filtering modes are supported on PERMEDIA (mip-map filtering is not supported).

BGRA Extension

This extension provides an additional pixel color format for compatibility with the blue, green, red component ordering of Microsoft Windows DIB's (device independent bitmaps). Refer to Appendix C in the 3Dlabs OpenGL Extensions document for further details.

Palette Texture Extension

The GLINT 500TX and PERMEDIA both provide direct support for palette textures, where each texel represents an index into an on-chip 16 entry RGBA (8-bits per component) lookup-table. An OpenGL palette texture extension has been defined by Microsoft which is supported by 3Dlabs OpenGL ICD from release 1.0.11. The GLINT 500TX supports 1, 2 & 4bit texel depths. The PERMEDIA supports 4bit texel depths. Besides improving texture performance and reducing the memory requirements for storing textures in the local buffer, by repeatedly updating the texture LUT, animation effects such as real-time colour cycling are also possible. Refer to Appendices B and C for further details.

3Dlabs Driver Extension

In addition to the extensions mentioned above, the 3Dlabs_DriverState extension has been added and is detailed at Appendix F in the OpenGL Extensions document. This extension is simply a mechanism for adding extra state to the Client Driver and add extra control to the currently selected context.

Heidi

This driver release supports the Heidi graphics API used by 3D Studio Max. The Heidi driver has the same system requirements as OpenGL. These are that the 1.1 versions of Micrsoft's OpenGL binaries be installed and that Direct X3 is installed. The 1.1 OpenGL binaries are provided as standard on the OEM SR2 release of Windows 95 and are also provided in the self-extracting archive opengl95.exe. Information on how to download the Microsoft OpenGL binaries can be found at the 3Dlabs web site or at:

> http://www.microsoft.com/kb/articles/q154/8/77.htm

Installation

If GLINT hardware is already in use for 3D Studio Max simply copy the wglint.hdi file to your 3dsmax/drivers directory to upgrade to this driver. If 3D Studio is not currently using GLINT hardware then first copy the the wglint.hdi file to your 3dsmax/drivers directory and then start up 3D Studio Max. Under the File menu choose Preferences and in the resulting dialog click on the Viewports tab. Click on the button labelled Choose Driver and then select GLINT Hardware. You will then need to restart Max to enable GLINT hardware acceleration.

Performance Optimization

On PERMEDIA based boards best performance will be obtained when running with 32k colors. On 500TX based boards best performance will be provided when running in TrueColor display modes. On both types of board performance will be optimized by setting the number of DMA buffers allocated to 4, this can be done from the GLINT control applet.

GLINT Control Applets

PERMEDIA/GLINT specific variables can be changed in three tabs on the desktop display settings panel. This can be found by right-clicking on the desktop and choosing "Properties".

PERMEDIA/GLINT Setup Page

The hardware specific settings such as Interrupts, etc. available are:

Max Refresh:	Controls the maximum refresh rate that the drivers will allow.
DMA Buffers:	Controls the number of DMA buffers that the drivers will allocate. Setting 0 here disables DMA.
DMA Buffer Size:	Controls the size of each DMA buffer.
Font Cache:	Controls the amount of memory allocated for the font cache. Setting 0 here disables font caching.
Interrupt Enable:	Enables interrupts for DMA operations.
Offscreen Bitmaps:	Enables the use of Offscreen bitmaps for faster display driver performance.
Disable Delta setup	Checking this tick box will prevent OpenGL using the GLINT DELTA triangle chip on Racer and PERMEDIA NT based boards. This is useful for performance comparisons.

PERMEDIA/GLINT Direct3D Page

Note that only those settings which are appropriate for a GLINT 500TX based board as indicated will be available on this page.

Selective Patching:	This scheme will patch textures if they are loaded onto the card using the Direct3D ALLOC_ON_LOAD scheme. In general patching textures improves performance (not relevant to GLINT 500TX).
Always Patch:	This scheme will always patch textures, whether or not they are stored using the ALLOC_ON_LOAD scheme. Note that this can potentially break applications that need to render to texture surfaces without compression. (not relevant to GLINT 500TX)
Never Patch:	Never patches textures (not relevant to GLINT 500TX).
8Bit RGB Textures:	Direct3D Retained mode chooses the texture format that best matches the original image file. This can cause applications to use 8Bit RGB textures when a higher resolution texture format would be better. For this reason 8Bit RGB textures are disabled by default. This check box enables 8Bit RGB textures.
True Color Textures:	Enables the use of True Color textures in the Direct3D HAL driver. This option will use a lot of video memory if it is picked by a Direct3D application.
Don't Force CopyMode:	Allows the Direct3D driver to use texture modulation. When turned off it allows the original color of the texture map to be seen (not relevant to GLINT 500TX).

Disable Direct3D HAL: Disables hardware 3D acceleration.

No Vblank Wait: Stops the DirectDraw HAL driver from waiting for Video Blanking signals before it page flips. This stops full screen applications waiting for the Video Blanking interval and allows them to run at full speed. This may result in some "tearing" of the images on the screen.

Filter Stretching: Filters all stretched blit operations. This should be off by default as it will slow stretch blit performance. Useful for real time anti-aliasing applications that wish to stretch large images down with filtering (not relevant to GLINT 500TX).

PERMEDIA/GLINT OpenGL Page

Dynamic Buffer Option - Number of Sub-Buffers

Each DMA buffer is sub-divided into sub-buffers which are used in conjunction with an DMA mechanism to reduce latency in the system.. This is to allow the 3D application to fill one buffer and GLINT/ PERMEDIA to be performing DMA on another buffer. Setting this variable to zero disables DMA and forces FIFOs to be used. Interrupt driven DMA is currently not supported in this release (the case when the number of sub-buffers is greater than 2 which allows a DMA interrupt handler to process the remaining queue of buffers).

Disable Fast Clear Planes

Checking this box will disable the use of the fast Depth Clear planes and is equivalent to setting the environment variable GLINT_DONT_USE_FCP to TRUE. This option should be used in cases where the Depth buffer needs to be read back into the application. This option is not relevant to   PERMEDIA.

Draw Line Endpoints

This option when set can improve the legibility of text rendered by some applications using stroke fonts.

Force Nearest Neighbour Texturing

Setting this registry value will ensure that OpenGL only performs nearest neighbor texturing operations. In some applications this can improve performance, at the cost of some visual quality. Note that textures will still be rendered with perspective correction.

Enable Texture Compression

Setting this registry value will force OpenGL to shrink 2D texture maps as they are loaded to reduce the memory needed to store them. Texture maps are halved in both x and y dimensions so that they require a quarter of the original memory. The setting has no effect on 1D or paletted texture maps. This setting applies to all hardware configurations.

Disable MipMapping

This variable is only relevant to the GLINT 500TX . Checking this tick box will prevent OpenGL from allowing mip-map texture filtering to be enabled. This option will override mip-map filtering settings of an OpenGL application, forcing the allowed filtering to be either GL_LINEAR or GL_NEAREST. These two filtering options are rendered much faster than the mip-map filtering option on the GLINT 500TX.

Perspective Correction

This variable is only relevant to PERMEDIA. The accuracy of the perspective-correction divide performed during textured rendering can be varied on PERMEDIA. The default option, Force Nicest, enforces that the most accurate divide always operates, resulting in the best image quality. The Force Fastest option switches to a reduced accuracy divide when Delta hardware is present and no perspective-correction at all if Delta is not present. This results in best performance at the cost of lower image quality. The third option gives control of the divide accuracy to OpenGL applications through the API function glHint. A smart application can vary the divide accuracy used on a per primitive basis, since primitives that are heavily perspected require higher accuracy for good image quality than those that are not.