

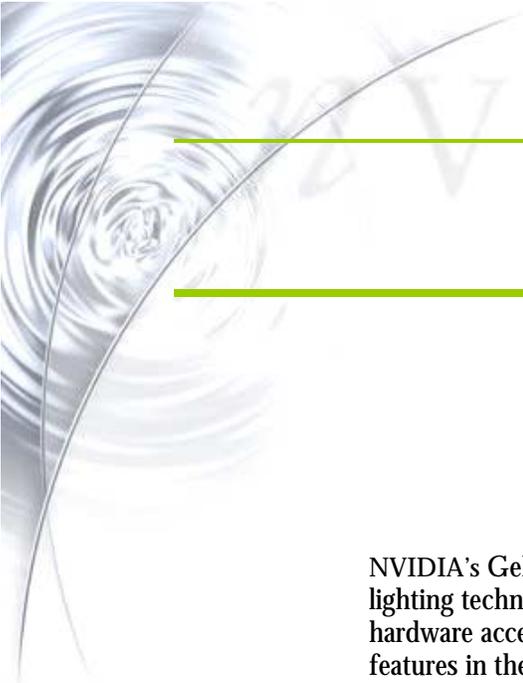


Shadow Buffer Technical Brief

Realistic Shadows Using
NVIDIA Shadow Buffer Technology



***n*VIDIA®**



Shadow Buffer Technology

NVIDIA's GeForce3 Ti Graphics Processing Units (GPU) are changing the face of lighting techniques for PC graphics. NVIDIA's new GPUs are the first to bring hardware acceleration for realistic shadows to consumer PCs. Special hardware features in the texture unit of the GeForce3 Ti family enable these GPUs to create and use shadow buffers for real-time shadows. The technique of using shadow buffers has been field proven for such animated films as Columbia Pictures computer generated (CG) motion picture *Final Fantasy: The Sprits Within*. Until now, shadow buffering was only available on professional rendering equipment outside of the consumers' reach (like SGI's Octane workstation).

NVIDIA shadow buffer technology closes the gap between PC gaming and cinematic style special effects for realistic, real-time shadows. This Technical Brief outlines the techniques used to create and utilize shadow buffers. It also describes the unique hardware features in the GeForce3 Ti GPUs that enable this groundbreaking capability and its benefits for end users.

Realistic Shadows Captivate Users

The fundamental goal of computer graphics is to create the perception of an alternate reality. Graphic imagery becomes more compelling as it gets closer to replicating a real world visual experience.

Until now, lighting techniques for real-time computer graphics were missing a key element: realistic shadows. Artists and photographers have always known the power of lighting and shadows. In fact, black and white film is still used for artistic photography because of the powerful effects that lighting and shadows, even without color, have on people. These same effects can be used to captivate users and draw them into real-time graphics environments.



Figure 1. Lighting Graphic Images

© 2001 FFFP. All Rights Reserved

Creating Real-Time Shadows

The NVIDIA shadow buffer technique used by the GeForce3 Ti processors involves creating a “map” of which objects are lighted in the scene. This map is stored in a shadow buffer so that it can be used later and accessed like a texture through the special shadow buffer hardware located in the texture engine within the GeForce3 Ti GPUs. Figure 2 is a block diagram that shows where the shadow buffer hardware is located in the GeForce3 Ti GPUs.

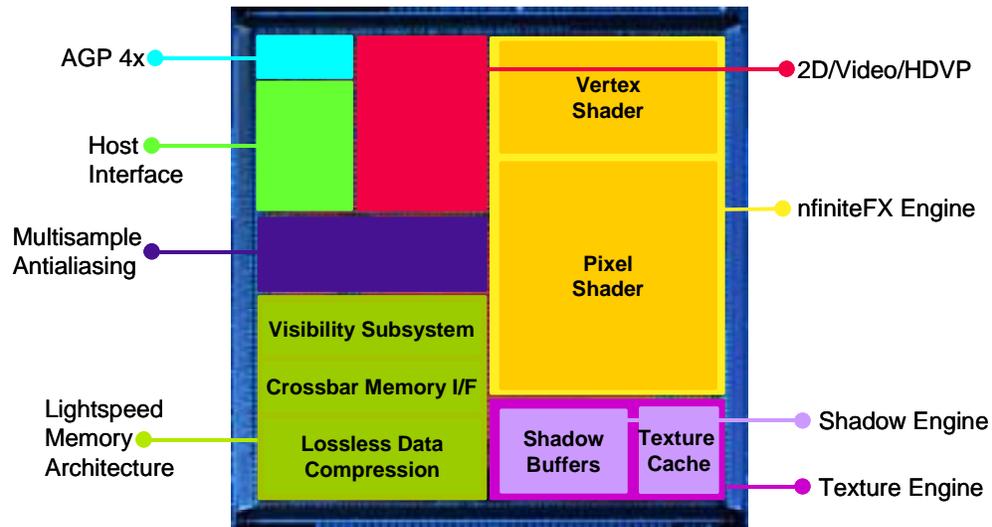


Figure 2. Location of Shadow Buffer Hardware on GeForce3 Ti GPU

While mapping the lighted areas instead of the shadowed areas may seem odd, this technique is faster and simplifies the hardware because, while there is only one lit pixel per XY location, there may be many shadowed pixels behind it. Shadow buffer support requires a number of new hardware features in the GeForce3 Ti GPU, but also leverages the general render-to-texture and texture-mapping capabilities previously implemented in the NVIDIA texture engines. As a result, shadow buffers enable the GeForce3 Ti to use texture-filtering techniques to create soft edges on shadows.

None of this would be possible however, if the GeForce3 Ti GPU did not have the special hardware features required to create and use a shadow buffer. These special hardware features include the ability to:

- ❑ Designate an area of graphics memory as a shadow buffer.
- ❑ Render the shadow map into the shadow buffer.
- ❑ Read data from the shadow buffer for the mathematical calculation to determine if a specific pixel in the frame buffer is completely shadowed, partially shadowed or not shadowed at all.

To create a shadow map, the GeForce3 GPU renders the scene from the point of view of the light, as if the viewer were located at the same position and looking in

the same direction as the light. As the GeForce3 renders the scene, it stores the depth information, or Z-value, for each pixel in the shadow buffer. The hardware has a special algorithm to bias this Z-value to avoid undesirable artifacts from precision errors. All of the objects in the scene are rendered, but only the smallest biased Z-value for each pixel is kept to ensure that the shadow buffer represents only the objects that are closest to the light.

Once a shadow map is created and stored in the shadow buffer, it is ready to be used. When the GeForce3 Ti GPU renders the scene, it performs the following steps:

1. Transform the pixel into the coordinate system defined by the light source.
2. Use the texture filtering techniques to read multiple Z-values stored in the shadow buffer from locations that surround the XY location of the pixel. The multiple Z-values are used for two important calculations:
 - a) Using a proprietary shadow algorithm, the multiple Z-values are compared to the single Z-value for the pixel to determine what fraction of the pixel is in shadow. This enables up to 256 discrete levels of shadow to create soft edges for shadows.
 - b) Using texture-filtering algorithms, the multiple Z-values are filtered down to a single Z value.
3. The results of steps 2a and 2b are used by the nfiniteFX engine as input data for pixel shading operations. If the Z value of the pixel is equal to (or very close to equal to) the Z value stored in the shadow buffer, then the pixel is lit. Otherwise, the pixel is shadowed with the degree of shadow determined by the result of step 2a. The nfiniteFX engine can also turn off specular highlights for any pixel that is in shadow, enhancing the realism of the image.

The images shown in Figure 3 and 4 illustrate this process visually.

To render the scene in Figure 3, it is necessary to first create the shadow map shown in Figure 4. Note the direction of the light source in Figure 3 and you'll see how the image in Figure 4 is rendered from the light's point of view.

Note that Figure 4 simply looks like a gray-scale image of the scene from the point of view of the light source.

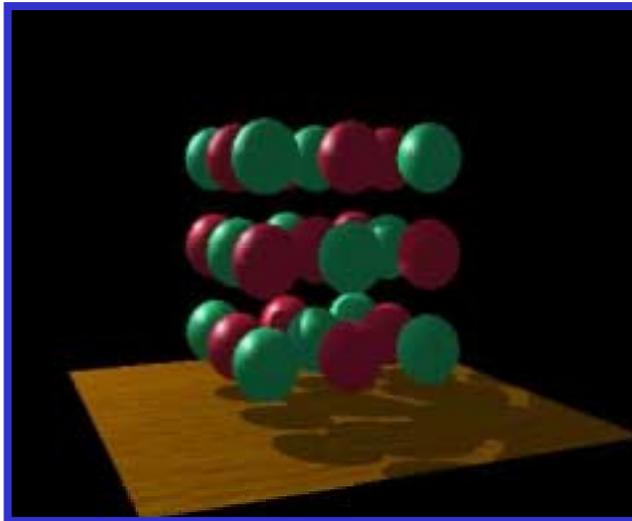


Figure 3. Scene Rendered from Shadow Map

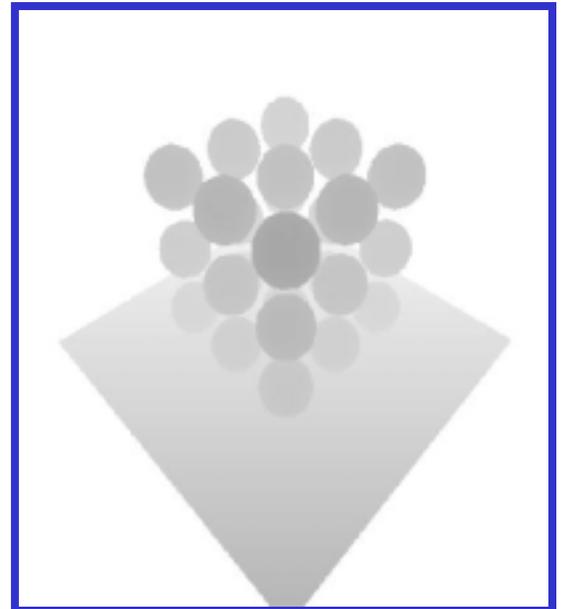


Figure 4. Shadow Map

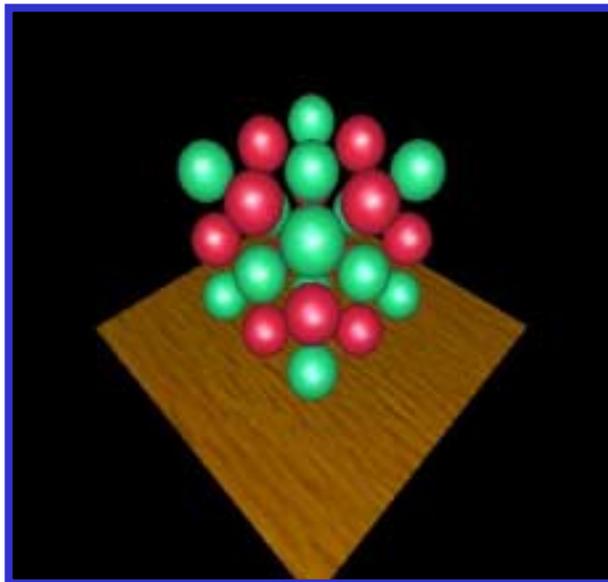


Figure 5. Scene Rendered in Full Color

Figure 5 shows the shadow buffer image rendered in full color to highlight the fact that everything that is “in view” of the light, will not be shadowed; everything else is shadowed.

For more examples of this process, see Appendix A.

NVIDIA Shadow Mapping Techniques

There are many techniques for rendering shadows and each of them have their strengths. However, shadow mapping with NVIDIA shadow buffers has a number of key advantages over alternative shadow rendering techniques:

- **Complex Scenes**
Works equally well for arbitrarily complex scenes (many objects or complex shapes). See Figure 6.

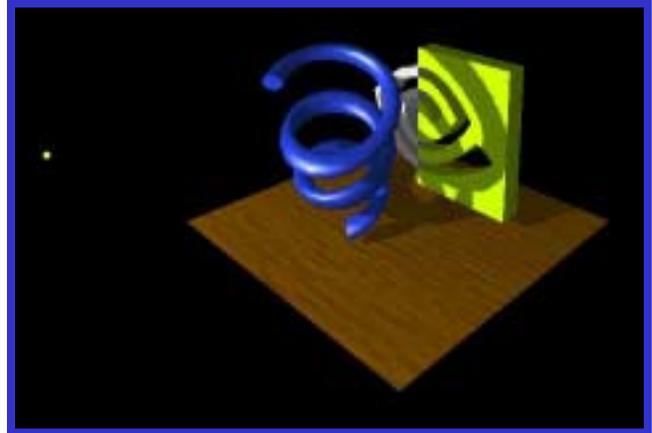
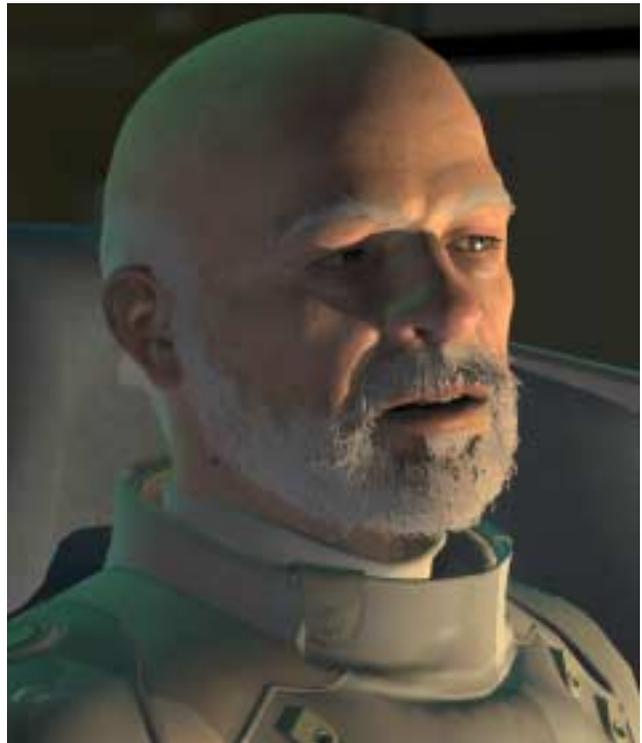


Figure 6. Rendering complex scenes

- **Soften shadow edges**
Shadow mapping can soften shadow edges using the proprietary shadow-filtering algorithm with up to 256 discrete levels of shadowing based on the percentage coverage of the pixel, if the pixel area is partially shadowed and partially lighted.



© 2001 FFFP. All Rights Reserved

Figure 7. Soften Shadow Edges

- **Self Shadow**

Objects can “self shadow” (a persons arm can cast a shadow on their chest.) Note the wings on the spacecraft in Figure 8 cast a shadow on other parts of the craft



Figure 8. Object Self-Shadow

- **Compatible with multi-texture rendering**

NVIDIA shadow buffers are compatible with multi-texture rendering. Stenciled shadow volumes can force multi-pass rendering, reducing performance if the application is either fill rate limited or memory bandwidth limited.



Figure 9. Multi-texture Rendering

Conclusion

NVIDIA shadow buffers represent a giant leap forward for the PC graphics industry. The unique hardware features of the GeForce3 Ti GPUs make realistic, real-time shadows available for the first time ever on consumer PCs and bridge the gap between cinema style shadow effects and the state-of-the-art for PC graphics.

Appendix A

How Shadow Buffers Work

Figure 10 uses a heavy black line to show which values get stored in the shadow buffer, leaving other locations 'in shadow' because they are 'behind' some other object or part of an object.

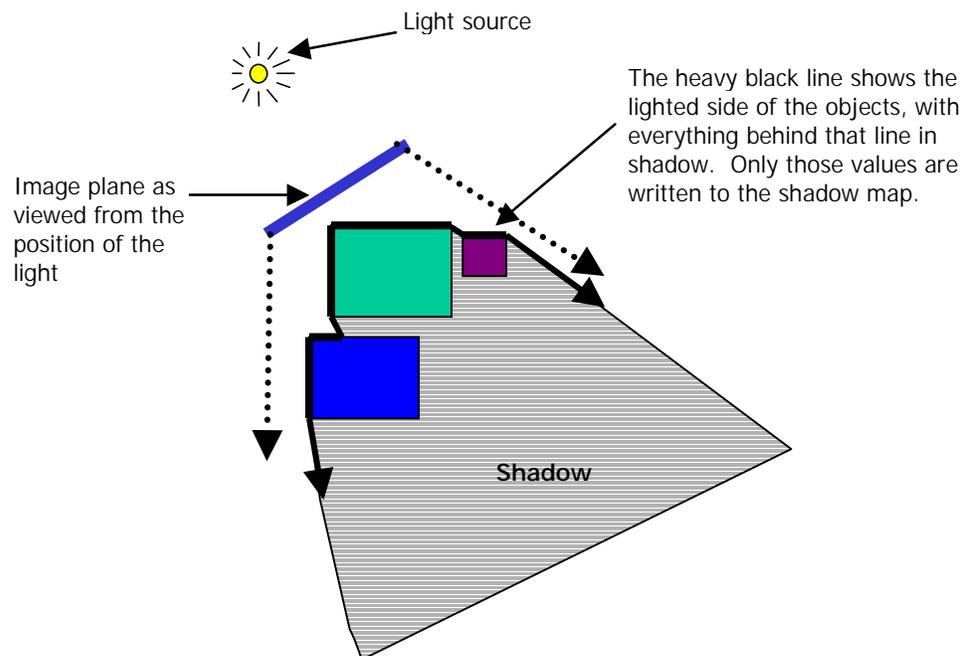


Figure 10. Stored Shadow Buffer Values

Figure 11 shows how objects that are hidden from the light (i.e. shadowed) can be visible to the eye when the scene is rendered from the eye's perspective. Note that the yellow circle is visible to the eye, but is hidden from the light. That simply means that it is 'in shadow.'

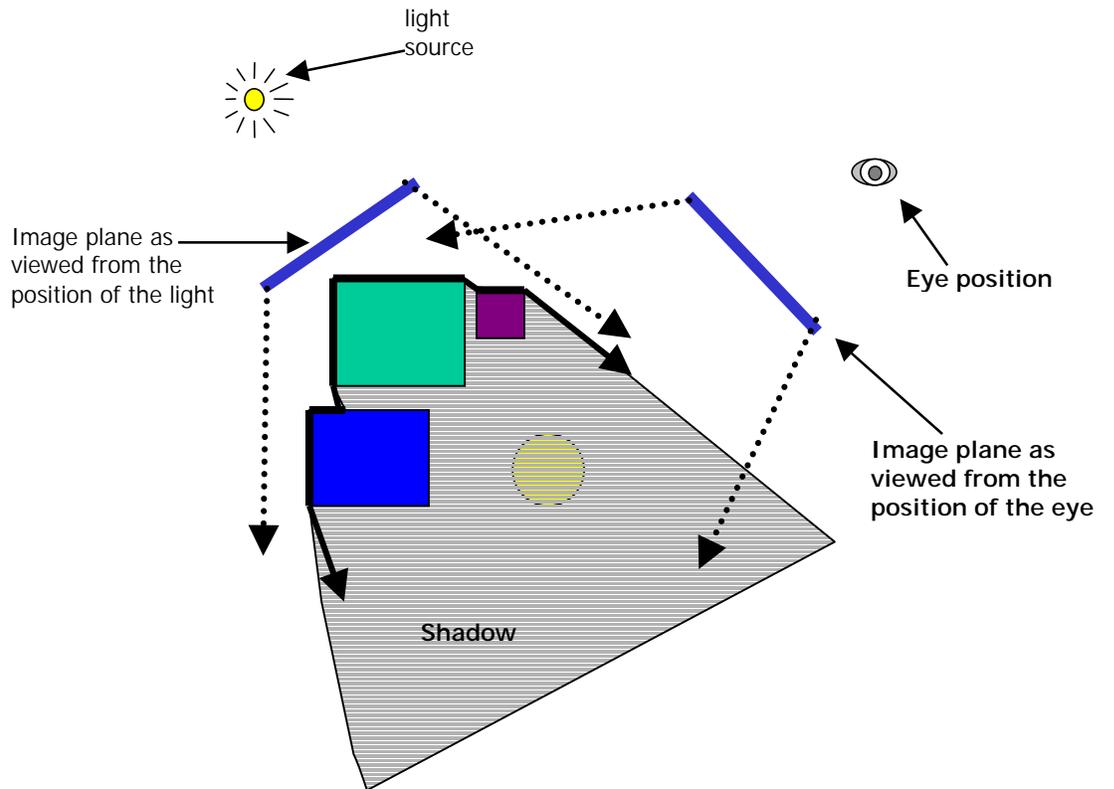


Figure 11. Objects Hidden from the Light

The specific test to determine whether the pixel is shadowed or lighted is the comparison of the distance Z_{light} to the filtered Z value from the shadow buffer. Figure 12 explicitly shows these two values via two arrows with the length of the arrows representing the distance of the object from the image plane, which has a value of $Z=0$.

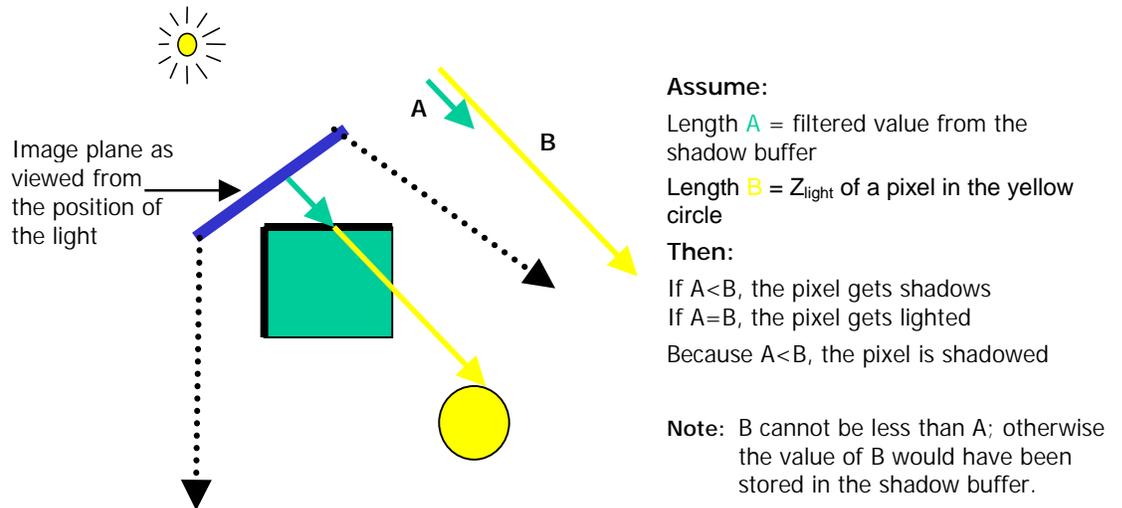


Figure 12. Determining When to Shadow or Light an Object

An alternate scenario would be where the yellow circle moves toward the light source until it is closer to the light than the green square. As shown in Figure 13, the yellow circle would cast a shadow on part of the green square, because $A \cong B$.

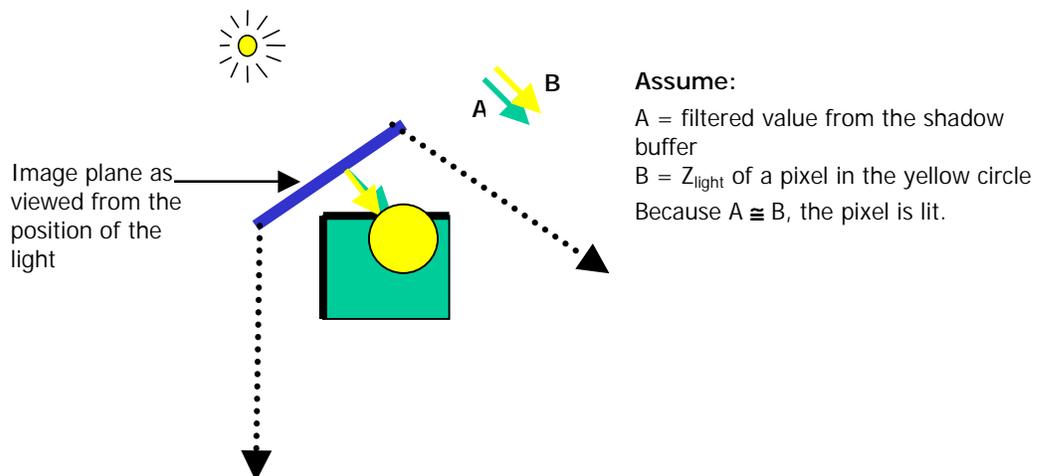
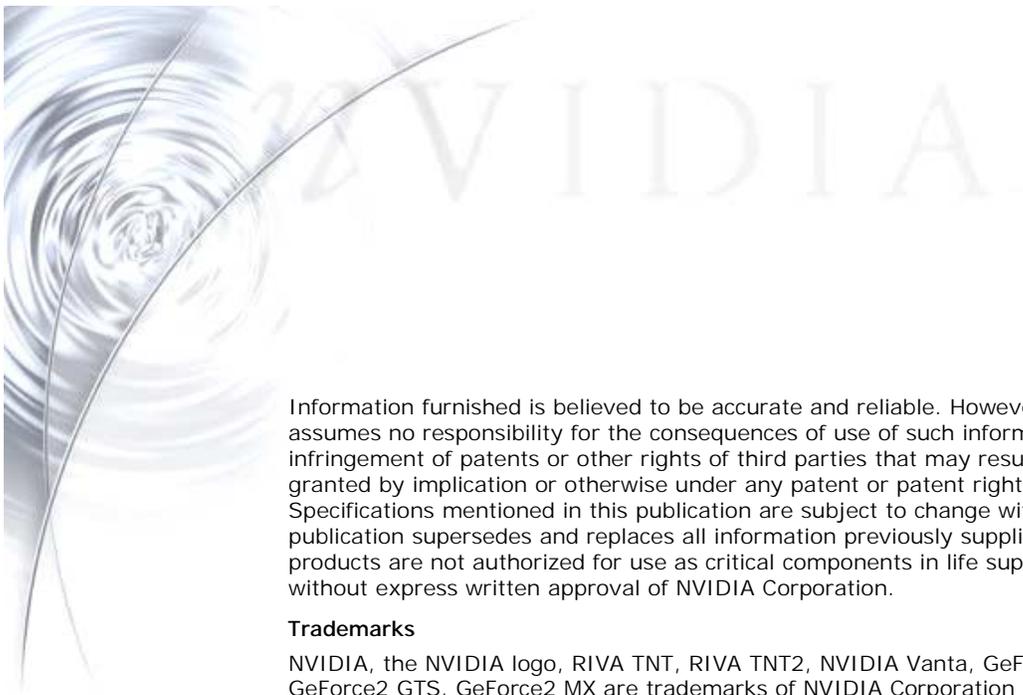


Figure 13. Self Shadows



Information furnished is believed to be accurate and reliable. However, NVIDIA Corporation assumes no responsibility for the consequences of use of such information or for any infringement of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent or patent rights of NVIDIA Corporation. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. NVIDIA Corporation products are not authorized for use as critical components in life support devices or systems without express written approval of NVIDIA Corporation.

Trademarks

NVIDIA, the NVIDIA logo, RIVA TNT, RIVA TNT2, NVIDIA Vanta, GeForce, GeForce2, and GeForce2 GTS, GeForce2 MX are trademarks of NVIDIA Corporation

Panel Link and TMDS are trademarks of Silicon Image, Inc.

Microsoft, Windows and the Windows logo are registered trademarks of Microsoft Corporation

RIVA, RIVA 128 and RIVA 128 ZX are trademarks of NVIDIA Corp. and STMicroelectronics.

Other company and product names may be trademarks of the respective companies with which they are associated.

Copyright

Copyright NVIDIA Corporation 2001



NVIDIA Corporation
2701 San Tomas Expressway
Santa Clara, CA 95050