
Subject: Re: NVIDIA Quadro FX 4000 Card (Any experience in using with IDL)
Posted by [Matt Feinstein](#) on Thu, 18 Nov 2004 15:38:31 GMT
[View Forum Message](#) <> [Reply to Message](#)

On 18 Nov 2004 06:13:31 -0800, gambler_1650@yahoo.com (Robert Gamble) wrote:

> I'm an IDL programmer at National Marine Fisheries and was wondering
> if anyone has had any experience using an NVIDIA Quadro FX 4000 card
> with IDL. We're looking for a workstation card due to the graphically
> intensive programs we hope to write down the road. It supports
> OpenGL, and we're hoping to also use it under linux so if anyone also
> happens to be using linux, IDL and said card or even earlier models of
> the Quadro FX line... (yeah, I know, probably pretty unlikely), any
> real world info would be helpful.

I have a Quadro FX 1000 card on my workstation, and it works with all applications. I also do OpenGL programming, and it works for that too.

As with just about any recent nVidia (or ATI) card, the only real requirement is to have the current video driver installed. For the Quadro cards, you've got to be somewhat careful about the power demands-- they are significantly higher than for, say, an off-the-shelf FX 5200 (which also works and is much cheaper).

FYI, here is the output of a program that queries the various extension strings and some framebuffer parameters:

```
alpha = 8  blue = 8  red = 8  green = 8
depth bits = 24 texture size = 4096 viewport size = (4096, 4096)
acc alpha = 16  acc red = 16  acc green = 16  acc blue = 16
max clip planes = 6  max vertices = 4096  max indices = 4096  stencil
bits = 0
Vendor = NVIDIA Corporation
Renderer = Quadro FX 1000/AGP/SSE2
OpenGL version = 1.5.1
GL extensions = GL_ARB_depth_texture GL_ARB_fragment_program
GL_ARB_fragment_program_shadow GL_ARB_fragment_shader GL_ARB_imaging
GL_ARB_multisample GL_ARB_multitexture GL_ARB_occlusion_query
GL_ARB_point_parameters GL_ARB_point_sprite GL_ARB_shadow
GL_ARB_shader_objects GL_ARB_shading_language_100
GL_ARB_texture_border_clamp GL_ARB_texture_compression
GL_ARB_texture_cube_map GL_ARB_texture_env_add
GL_ARB_texture_env_combine GL_ARB_texture_env_dot3
GL_ARB_texture_mirrored_repeat GL_ARB_transpose_matrix
GL_ARB_vertex_buffer_object GL_ARB_vertex_program GL_ARB_vertex_shader
```

GL_ARB_window_pos GL_S3_s3tc GL_EXT_texture_env_add GL_EXT_abgr
 GL_EXT_bgra GL_EXT_blend_color GL_EXT_blend_func_separate
 GL_EXT_blend_minmax GL_EXT_blend_subtract GL_EXT_compiled_vertex_array
 GL_EXT_Cg_shader GL_EXT_draw_range_elements GL_EXT_fog_coord
 GL_EXT_multi_draw_arrays GL_EXT_packed_pixels GL_EXT_paletted_texture
 GL_EXT_pixel_buffer_object GL_EXT_point_parameters
 GL_EXT_rescale_normal GL_EXT_secondary_color
 GL_EXT_separate_specular_color GL_EXT_shadow_funcs
 GL_EXT_shared_texture_palette GL_EXT_stencil_two_side
 GL_EXT_stencil_wrap GL_EXT_texture3D GL_EXT_texture_compression_s3tc
 GL_EXT_texture_cube_map GL_EXT_texture_edge_clamp
 GL_EXT_texture_env_combine GL_EXT_texture_env_dot3
 GL_EXT_texture_filter_anisotropic GL_EXT_texture_lod
 GL_EXT_texture_lod_bias GL_EXT_texture_object GL_EXT_vertex_array
 GL_HP_occlusion_test GL_IBM_rasterpos_clip
 GL_IBM_texture_mirrored_repeat GL_KTX_buffer_region GL_NV_blend_square
 GL_NV_copy_depth_to_color GL_NV_depth_clamp GL_NV_fence
 GL_NV_float_buffer GL_NV_fog_distance GL_NV_fragment_program
 GL_NV_fragment_program_option GL_NV_half_float
 GL_NV_light_max_exponent GL_NV_multisample_filter_hint
 GL_NV_occlusion_query GL_NV_packed_depth_stencil
 GL_NV_pixel_data_range GL_NV_point_sprite GL_NV_primitive_restart
 GL_NV_register_combiners GL_NV_register_combiners2
 GL_NV_texgen_reflection GL_NV_texture_compression_vtc
 GL_NV_texture_env_combine4 GL_NV_texture_expand_normal
 GL_NV_texture_rectangle GL_NV_texture_shader GL_NV_texture_shader2
 GL_NV_texture_shader3 GL_NV_vertex_array_range
 GL_NV_vertex_array_range2 GL_NV_vertex_program GL_NV_vertex_program1_1
 GL_NV_vertex_program2 GL_NV_vertex_program2_option
 GL_SGIS_generate_mipmap GL_SGIS_texture_lod GL_SGIX_depth_texture
 GL_SGIX_shadow GL_SUN_slice_accum GL_WIN_swap_hint
 WGL_EXT_swap_control GL_Autodesk_valid_back_buffer_hint
 Glu version = 1.2.2.0 Microsoft Corporation
 Glu extensions = GL_EXT_bgra
 wgl extensions = WGL_ARB_buffer_region WGL_ARB_extensions_string
 WGL_ARB_make_current_read WGL_ARB_multisample WGL_ARB_pbuffer
 WGL_ARB_pixel_format WGL_ARB_render_texture WGL_EXT_extensions_string
 WGL_EXT_swap_control WGL_NV_float_buffer WGL_NV_render_depth_texture
 WGL_NV_render_texture_rectangle WGL_I3D_genlock WGL_NV_swap_group
 Pbuffer extensions initialized:
 wglCreatePbufferARB
 wglGetPbufferDCARB
 wglReleasePbufferDCARB
 wglDestroyPbufferARB
 wglQueryPbufferARB
 wglGetPixelFormatAttribfvARB
 wglGetPixelFormatAttribivARB
 wglChoosePixelFormatARB

Matt Feinstein

--
There is no virtue in believing something that can be proved to be true.
