

VimVentures
MEDIEVAL 0.0.3

Generated by Doxygen 1.10.0

1 Welcome to VimVentures!	1
1.1 What's VimVentures?	1
1.2 Why VimVentures?	1
1.3 Installation	2
1.4 Contributing	2
2 Todo List	3
3 Hierarchical Index	5
3.1 Class Hierarchy	5
4 Class Index	7
4.1 Class List	7
5 File Index	9
5.1 File List	9
6 Class Documentation	11
6.1 ShaderProgram::_passKey_t Struct Reference	11
6.2 App Class Reference	11
6.2.1 Constructor & Destructor Documentation	12
6.2.1.1 App()	12
6.2.1.2 ~App()	12
6.2.2 Member Function Documentation	13
6.2.2.1 AddRenderable()	13
6.2.2.2 ClearRenderables()	13
6.2.2.3 CreateGraphicsPipeline()	13
6.2.2.4 DeltaTime()	13
6.2.2.5 GenerateFonts()	13
6.2.2.6 GetFPS()	14
6.2.2.7 GetHeight()	14
6.2.2.8 GetRenderer()	14
6.2.2.9 GetWidth()	14
6.2.2.10 Render()	14
6.2.2.11 Run()	14
6.2.2.12 SetState()	14
6.2.2.13 Stop()	15
6.2.3 Member Data Documentation	15
6.2.3.1 m_state	15
6.2.3.2 m_width	15
6.2.3.3 SECOND_MS	15
6.3 Error Struct Reference	15
6.3.1 Constructor & Destructor Documentation	15
6.3.1.1 Error()	15

6.3.2 Member Function Documentation	16
6.3.2.1 toString()	16
6.4 gladGLversionStruct Struct Reference	16
6.5 GraphicsController Class Reference	16
6.5.1 Member Function Documentation	17
6.5.1.1 CheckGLObjectStatus()	17
6.5.1.2 debugMessage()	17
6.5.1.3 enableDebug()	18
6.5.1.4 initGL()	18
6.5.2 Member Data Documentation	18
6.5.2.1 s_shaderPrograms	18
6.6 IntroState Class Reference	18
6.6.1 Constructor & Destructor Documentation	19
6.6.1.1 IntroState()	19
6.6.1.2 ~IntroState()	19
6.6.2 Member Function Documentation	20
6.6.2.1 NextSlide()	20
6.6.2.2 Run()	20
6.6.2.3 SendEvent()	20
6.7 IRender Class Reference	20
6.7.1 Member Function Documentation	21
6.7.1.1 GenBindBufferGL()	21
6.7.1.2 Render()	21
6.7.1.3 UpdateGL()	21
6.8 IState Class Reference	21
6.8.1 Member Function Documentation	22
6.8.1.1 Run()	22
6.8.1.2 SendEvent()	22
6.9 MenuState Class Reference	22
6.9.1 Constructor & Destructor Documentation	23
6.9.1.1 MenuState()	23
6.9.1.2 ~MenuState()	23
6.9.2 Member Function Documentation	24
6.9.2.1 Run()	24
6.9.2.2 Select()	24
6.9.2.3 SelectorNext()	24
6.9.2.4 SelectorPrev()	24
6.9.2.5 SendEvent()	24
6.10 MissionState Class Reference	24
6.10.1 Constructor & Destructor Documentation	25
6.10.1.1 MissionState()	25
6.10.1.2 ~MissionState()	25

6.10.2 Member Function Documentation	25
6.10.2.1 Run()	25
6.10.2.2 SendEvent()	26
6.11 Rect2D Class Reference	26
6.11.1 Constructor & Destructor Documentation	27
6.11.1.1 Rect2D()	27
6.11.1.2 ~Rect2D()	27
6.11.2 Member Function Documentation	28
6.11.2.1 GenBindBufferGL()	28
6.11.2.2 GetHeight()	28
6.11.2.3 GetVertexData()	28
6.11.2.4 GetWidth()	28
6.11.2.5 GetX()	28
6.11.2.6 GetY()	29
6.11.2.7 Render()	29
6.11.2.8 SetPosition()	29
6.11.2.9 SetShaderProgram()	29
6.11.2.10 SetSize()	29
6.11.2.11 UpdateGL()	30
6.11.2.12 UpdateVertexData()	30
6.12 SDLToX11Keymap Class Reference	30
6.12.1 Constructor & Destructor Documentation	31
6.12.1.1 SDLToX11Keymap()	31
6.12.2 Member Function Documentation	31
6.12.2.1 convert()	31
6.13 ShaderProgram Class Reference	31
6.13.1 Constructor & Destructor Documentation	32
6.13.1.1 ~ShaderProgram()	32
6.13.1.2 ShaderProgram()	32
6.13.2 Member Function Documentation	32
6.13.2.1 GetProgramId()	32
6.14 ShaderProgramBuilder Class Reference	33
6.14.1 Constructor & Destructor Documentation	33
6.14.1.1 ~ShaderProgramBuilder()	33
6.14.2 Member Function Documentation	33
6.14.2.1 GenerateShaderProgram()	33
6.14.2.2 LoadShader()	33
6.14.2.3 LoadShaderFile()	34
6.15 IntroState::Slide Struct Reference	34
6.16 TexturedRect2D Class Reference	35
6.16.1 Constructor & Destructor Documentation	36
6.16.1.1 TexturedRect2D()	36

6.16.1.2 ~TexturedRect2D()	37
6.16.2 Member Function Documentation	37
6.16.2.1 DisableTextureBlend()	37
6.16.2.2 EnableTextureBlend()	37
6.16.2.3 Render()	37
6.16.2.4 SetInternalTextureFormat()	37
6.16.2.5 SetTexture()	37
6.16.2.6 SetTextureFormat()	38
6.16.2.7 UpdateGL()	38
6.16.2.8 UpdateVertexData()	38
6.17 VimEmulator Class Reference	38
6.17.1 Constructor & Destructor Documentation	41
6.17.1.1 VimEmulator()	41
6.17.1.2 ~VimEmulator()	41
6.17.2 Member Function Documentation	41
6.17.2.1 findWindowByName()	41
6.17.2.2 GetFrameAsSurface()	41
6.17.2.3 MatchResizeEvent()	42
6.17.2.4 QueueFrame()	42
6.17.2.5 QueueFrameThread()	42
6.17.2.6 RegisterWindow()	42
6.17.2.7 RegisterWindowThread()	43
6.17.2.8 Render()	43
6.17.2.9 ResizeWindow()	43
6.17.2.10 ResizeWindowThread()	43
6.17.2.11 SendSDLKey()	44
6.17.2.12 SetSDLMod()	44
6.17.3 Member Data Documentation	44
6.17.3.1 m_modmask	44
6.17.3.2 R_MASK	44
7 File Documentation	45
7.1 App.hpp	45
7.2 constants.hpp	46
7.3 Error.hpp	46
7.4 glad.h	46
7.5 GraphicsController.hpp	243
7.6 IntroState.hpp	244
7.7 IRender.hpp	244
7.8 IState.hpp	245
7.9 khrplatform.h	245
7.10 MenuState.hpp	249

7.11 MissionState.hpp	249
7.12 Rect2D.hpp	250
7.13 SDLToX11Keymap.hpp	250
7.14 ShaderProgram.hpp	251
7.15 ShaderProgramBuilder.hpp	251
7.16 TexturedRect2D.hpp	251
7.17 VimEmulator.hpp	252
Index	255

Chapter 1

Welcome to VimVentures!

An open-source game, VimVentures, is a space-themed odyssey to mastering vim! Currently in development, this game invites you to a unique adventure across the cosmos.

1.1 What's VimVentures?

Imagine piloting a spaceship through the vast universe, but there's a twist: solar flares are tampering with your ship's source code! Your mission is to wield your vim skills to overcome these cosmic challenges and keep your ship on path to it's next destination.

1.2 Why VimVentures?

I began my journey learning Vim at the end of 2022. I have learned much from the vim community and want to give back!

My goal is to create a fun, and accessible way to learn vim that will be useful for users of any proficiency. Here are some of the planned features:

Fun and Functionality Combined: This isn't just any game; it's a vim-learning experience set against the backdrop of an interstellar voyage. There will be upgrades to your ship at the various stations you stop by, and a story that will keep you engaged throughout your journey.

Thrilling Time-Sensitive Missions: Tackle urgent code repairs to keep your spaceship on course, adding a thrilling aspect to your learning.

VentureTutor (Name is WIP): As you progress through missions, the game tracks actions you might have missed, aiming to enhance your vim efficiency for subsequent voyages. Whether you complete your mission or not, you'll receive a list of your most missed actions, complete with demonstrations of where an alternative action could have been more efficient. For example:

You pressed 'x' the most efficient command here would be 'diw'

Under Construction: Alongside this being a contribution to the vim community, this is also a learning project for me!

If you want to learn more about VimVentures, all resources and information for the project are available at the [VimVentures Wiki](#).

1.3 Installation

Before installing, VimVentures relies on the following dependencies:

- SDL2
- X11
- Alacritty
- Vim

To install, perform the following inside of the root directory:

```
cmake .  
make
```

Then, the game will be compiled inside of the `bin/` directory.

```
./bin/VimVentures
```

1.4 Contributing

The game is in its early stages. Contribution guidelines will evolve as the project matures towards a phase where building a collaborative community becomes feasible.

Source Code: Source code is under the AGPL license, please keep this in mind if you wish to contribute.

Assets: All assets, including images, music, and sound effects, should adhere to the Creative Commons Attribution-NonCommercial-ShareAlike (CC BY-NC-SA) License. Contributions will be credited to the game's documentation and credits.

Chapter 2

Todo List

Member [App::CreateGraphicsPipeline](#) ()

Incorporate other shaders, perhaps a constexpr list

Member [App::GenerateFonts](#) ()

May be better to grab pre-serialised, though this matches with OpenGL philosophy.

Member [TexturedRect2D::Render](#) () **override**

Gives texture rects a default shader program and method to select others.

Chapter 3

Hierarchical Index

3.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

ShaderProgram::_passKey_t	11
App	11
Error	15
gladGLversionStruct	16
GraphicsController	16
IRender	20
Rect2D	26
TexturedRect2D	35
VimEmulator	38
IState	21
IntroState	18
MenuState	22
MissionState	24
SDLToX11Keymap	30
ShaderProgram	31
ShaderProgramBuilder	33
IntroState::Slide	34

Chapter 4

Class Index

4.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

ShaderProgram::_passKey_t	11
App	11
Error	15
gladGLversionStruct	16
GraphicsController	16
IntroState	18
IRender	20
IState	21
MenuState	22
MissionState	24
Rect2D	26
SDLToX11Keymap	30
ShaderProgram	31
ShaderProgramBuilder	33
IntroState::Slide	34
TexturedRect2D	35
VimEmulator	38

Chapter 5

File Index

5.1 File List

Here is a list of all documented files with brief descriptions:

include/App.hpp	45
include/constants.hpp	46
include/Error.hpp	46
include/GraphicsController.hpp	243
include/IntroState.hpp	244
include/IRender.hpp	244
include/IState.hpp	245
include/MenuState.hpp	249
include/MissionState.hpp	249
include/Rect2D.hpp	250
include/SDLToX11Keymap.hpp	250
include/ShaderProgram.hpp	251
include/ShaderProgramBuilder.hpp	251
include/TexturedRect2D.hpp	251
include/VimEmulator.hpp	252
include/glad/glad.h	46
include/KHR/khrplatform.h	245

Chapter 6

Class Documentation

6.1 ShaderProgram::_passKey_t Struct Reference

The documentation for this struct was generated from the following file:

- include/ShaderProgram.hpp

6.2 App Class Reference

Public Member Functions

- [App](#) (UInt32 ssFlags, int x, int y, int w, int h)
- [~App](#) ()
- SDL_Renderer * [GetRenderer](#) ()
- void [SetState](#) (IState *state)
- void [Run](#) ()
- void [Stop](#) ()
- void [Render](#) ()
- void [AddRenderable](#) (std::shared_ptr< [IRender](#) > const &renderable)
- void [ClearRenderables](#) ()
- double [GetFPS](#) () const
- double [DeltaTime](#) () const
- int [GetWidth](#) () const
- int [GetHeight](#) () const

Private Member Functions

- void [PreDraw](#) () const

Static Private Member Functions

- static void [debugMessage](#) (GLenum source, GLenum type, GLuint debug_id, GLenum severity, GLsizei length, const GLchar *message, const void *userParam)
- static void [CreateGraphicsPipeline](#) ()
- static void [GenerateFonts](#) ()

Private Attributes

- `SDL_Window * m_window`
- `SDL_Renderer * m_renderer`
- `std::vector< std::shared_ptr< IRender > > m_renderables`
- `int m_width`
- `int m_height`
- `bool m_running`
- `IState * m_state`
- `Uint32 m_startTick`
- `Uint32 m_endTick`
- `double m_deltaTime`

Static Private Attributes

- `static constexpr double SECOND_MS = 1000.0`
- `static constexpr int APP_GL_DEPTH_SIZE = 24`
- `static constexpr int FONT_RENDER_SIZE = 128`

6.2.1 Constructor & Destructor Documentation

6.2.1.1 App()

```
App::App (
    Uint32 ssFlags,
    int x,
    int y,
    int w,
    int h )
```

VimVentures Game.

Parameters

<i>ssFlags</i>	Subsystem Flags
<i>x</i>	Window X Position
<i>y</i>	Window Y Position
<i>w</i>	Window Width
<i>h</i>	Window Height

6.2.1.2 ~App()

```
App::~App ( )
```

Destroy Game Window and Instance.

6.2.2 Member Function Documentation

6.2.2.1 AddRenderable()

```
void App::AddRenderable (
    std::shared_ptr< IRender > const & renderable )
```

Add a new renderable to renderables.

Parameters

<i>Renderable</i>	
-------------------	--

6.2.2.2 ClearRenderables()

```
void App::ClearRenderables ( )
```

Clears renderables.

6.2.2.3 CreateGraphicsPipeline()

```
void App::CreateGraphicsPipeline ( ) [static], [private]
```

Builds and loads the canode shader into the shader map.

Todo Incorporate other shaders, perhaps a constexpr list

6.2.2.4 DeltaTime()

```
double App::DeltaTime ( ) const
```

Returns

Delta Time

6.2.2.5 GenerateFonts()

```
void App::GenerateFonts ( ) [static], [private]
```

Generate Fonts.

Todo May be better to grab pre-serialised, though this matches with OpenGL philosophy.

6.2.2.6 GetFPS()

```
double App::GetFPS ( ) const
```

Getters

Returns

Game FPS

6.2.2.7 GetHeight()

```
int App::GetHeight ( ) const
```

Returns

Window Height

6.2.2.8 GetRenderer()

```
SDL_Renderer * App::GetRenderer ( )
```

Returns

SDL_Renderer context

6.2.2.9 GetWidth()

```
int App::GetWidth ( ) const
```

Returns

Window Width

6.2.2.10 Render()

```
void App::Render ( )
```

Render cycle, calls all renderables.

6.2.2.11 Run()

```
void App::Run ( )
```

Begin game cycle.

6.2.2.12 SetState()

```
void App::SetState (
    IState * state )
```

State Management

Sets the game state.

Parameters

<i>Game</i>	state
-------------	-------

6.2.2.13 Stop()

```
void App::Stop ( )
```

Stop Game Cycle.

6.2.3 Member Data Documentation**6.2.3.1 m_state**

```
IState* App::m_state [private]
```

State

6.2.3.2 m_width

```
int App::m_width [private]
```

Program Details

6.2.3.3 SECOND_MS

```
constexpr double App::SECOND_MS = 1000.0 [static], [constexpr], [private]
```

FPS

The documentation for this class was generated from the following files:

- include/App.hpp
- src/App.cpp

6.3 Error Struct Reference**Public Member Functions**

- [Error](#) (int infoLogLen, const char *infoLog)
- std::string [toString](#) () const

Public Attributes

- const int **m_infoLogLen**
- const char * **m_infoLog**

6.3.1 Constructor & Destructor Documentation**6.3.1.1 Error()**

```
Error::Error (
    int infoLogLen,
    const char * infoLog )
```

[Error](#) type for OpenGL errors, stored as a c-string.

Parameters

<i>infoLogLen</i>	Length of the string
<i>infoLog</i>	C-string log

6.3.2 Member Function Documentation

6.3.2.1 toString()

```
std::string Error::toString ( ) const
```

Converts error log to std::string.

Returns

[Error](#) log as std::string

The documentation for this struct was generated from the following files:

- include/Error.hpp
- src/Error.cpp

6.4 gladGLversionStruct Struct Reference

Public Attributes

- int **major**
- int **minor**

The documentation for this struct was generated from the following file:

- include/glad/glad.h

6.5 GraphicsController Class Reference

Static Public Member Functions

- static void [debugMessage](#) (GLenum source, GLenum type, GLuint debug_id, GLenum severity, GLsizei length, const GLchar *message, const void *userParam)
- static void [enableDebug](#) ()
- static std::optional< [Error](#) > [initGL](#) (SDL_Window *sdlWindow)
- static std::optional< [Error](#) > [CheckGLObjectStatus](#) (GLuint object, GLenum parameter, const std::function< void(GLuint, GLenum, GLint *)> &getiv, const std::function< void(GLuint, GLsizei, GLsizei *, GLchar *)> &getInfoLog)

Static Public Attributes

- static std::unordered_map< std::string, std::unique_ptr< [ShaderProgram](#) > > [s_shaderPrograms](#)
- static std::unordered_map< std::string, std::unique_ptr< TTF_Font, std::function< void(TTF_Font *)> > > [s_fonts](#))

Static Private Attributes

- static SDL_GLContext [s_glContext](#)

6.5.1 Member Function Documentation

6.5.1.1 CheckGLObjectStatus()

```
std::optional< Error > GraphicsController::CheckGLObjectStatus (
    GLuint object,
    GLenum parameter,
    const std::function< void(GLuint, GLenum, GLint *)> & getiv,
    const std::function< void(GLuint, GLsizei, GLsizei *, GLchar *)> & getInfoLog )
[static]
```

Checks the status of an OpenGL object and retrieves info logs if there are any errors.

Parameters

<i>object</i>	The ID of the OpenGL object (e.g. shaders, program)
<i>parameter</i>	The specific status parameter to query (e.g. GL_COMPILE_STATUS, GL_LINK_STATUS)
<i>getiv</i>	Parameter getter for the object type (e.g. glGetShaderiv)
<i>getInfoLog</i>	Info log getter for the object type (e.g. glGetShaderInfoLog)

Returns

std::optional<Error> Returns infolog if status results in an error

6.5.1.2 debugMessage()

```
void GraphicsController::debugMessage (
    GLenum source,
    GLenum type,
    GLuint debug_id,
    GLenum severity,
    GLsizei length,
    const GLchar * message,
    const void * userParam ) [static]
```

Debug message hook for OpenGL.

Parameters

<i>source</i>	
---------------	--

Parameters

<i>type</i>	
<i>debug_id</i>	
<i>severity</i>	
<i>length</i>	
<i>message</i>	
<i>userParam</i>	

6.5.1.3 enableDebug()

```
void GraphicsController::enableDebug ( ) [static]
```

Enables debug message callbacks on [GraphicsController::debugMessage](#).

6.5.1.4 initGL()

```
std::optional< Error > GraphicsController::initGL (
    SDL_Window * sdlWindow ) [static]
```

Initializes the OpenGL context.

Returns

std::optional<Error> Returns error if initialization fails

6.5.2 Member Data Documentation**6.5.2.1 s_shaderPrograms**

```
std::unordered_map< std::string, std::unique_ptr< ShaderProgram > > GraphicsController::s_↵
shaderPrograms [static]
```

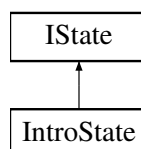
Static map storing shader programs, with strings as keys.

The documentation for this class was generated from the following files:

- include/GraphicsController.hpp
- src/GraphicsController.cpp

6.6 IntroState Class Reference

Inheritance diagram for IntroState:



Classes

- struct [Slide](#)

Public Member Functions

- [IntroState](#) ([App](#) &app)
- [~IntroState](#) () override
- void [Run](#) () override
- void [SendEvent](#) (SDL_Event &event) override

Private Member Functions

- void [NextSlide](#) ()

Private Attributes

- [App](#) & **app**
- std::shared_ptr< [TexturedRect2D](#) > **m_slideshow**
- [Slide](#) **m_slides** [3]
- int **m_currSlide**

Static Private Attributes

- static constexpr int **SLIDE_COUNT** = 3
- static constexpr float **SLIDE_RIGHT** = -1.4 / 2
- static constexpr float **SLIDE_LEFT** = -1.0 / 2
- static constexpr float **SLIDE_TOP** = -1.4 / 2
- static constexpr float **SLIDE_BOTTOM** = -1.0 / 2
- static constexpr float **SLIDE_MIDDLE** = -1.2 / 2
- static constexpr float **SLIDE_ZOOM** = 1.2
- static constexpr float **SECOND_MS** = 1000.0

6.6.1 Constructor & Destructor Documentation

6.6.1.1 IntroState()

```
IntroState::IntroState (
    App & app )
```

Mission Level State.

6.6.1.2 ~IntroState()

```
IntroState::~~IntroState ( ) [override], [default]
```

Default desctructor.

6.6.2 Member Function Documentation

6.6.2.1 NextSlide()

```
void IntroState::NextSlide ( ) [private]
```

Switches to the next slide if one is available.

Updates the [TexturedRect2D](#) slideshow.

6.6.2.2 Run()

```
void IntroState::Run ( ) [override], [virtual]
```

Runs the state loop.

Implements [IState](#).

6.6.2.3 SendEvent()

```
void IntroState::SendEvent (
    SDL_Event & event ) [override], [virtual]
```

Handles SDL Input.

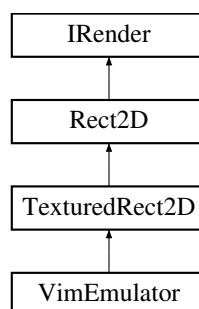
Implements [IState](#).

The documentation for this class was generated from the following files:

- include/IntroState.hpp
- src/IntroState.cpp

6.7 IRender Class Reference

Inheritance diagram for IRender:



Public Member Functions

- virtual void [Render](#) ()=0
- virtual void [UpdateGL](#) ()=0

Protected Member Functions

- virtual void [GenBindBufferGL](#) ()=0

Protected Attributes

- GLuint [m_vertexArrayObject](#)
- GLuint [m_vertexBufferObject](#)
- GLuint [m_indexBufferObject](#)
- std::string [m_shaderProgram](#)

6.7.1 Member Function Documentation

6.7.1.1 GenBindBufferGL()

```
virtual void IRender::GenBindBufferGL ( ) [protected], [pure virtual]
```

Implemented in [Rect2D](#).

6.7.1.2 Render()

```
virtual void IRender::Render ( ) [pure virtual]
```

Implemented in [Rect2D](#), [TexturedRect2D](#), and [VimEmulator](#).

6.7.1.3 UpdateGL()

```
virtual void IRender::UpdateGL ( ) [pure virtual]
```

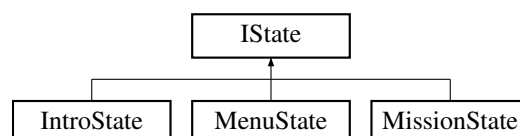
Implemented in [Rect2D](#), and [TexturedRect2D](#).

The documentation for this class was generated from the following file:

- include/IRender.hpp

6.8 IState Class Reference

Inheritance diagram for IState:



Public Member Functions

- virtual void [Run](#) ()=0
- virtual void [SendEvent](#) (SDL_Event &event)=0

6.8.1 Member Function Documentation

6.8.1.1 Run()

```
virtual void IState::Run ( ) [pure virtual]
```

Implemented in [IntroState](#), [MenuState](#), and [MissionState](#).

6.8.1.2 SendEvent()

```
virtual void IState::SendEvent (
    SDL_Event & event ) [pure virtual]
```

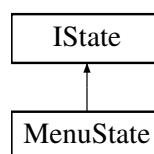
Implemented in [IntroState](#), [MenuState](#), and [MissionState](#).

The documentation for this class was generated from the following file:

- include/IState.hpp

6.9 MenuState Class Reference

Inheritance diagram for MenuState:



Public Member Functions

- [MenuState](#) (App &app)
- [~MenuState](#) () override
- void [Run](#) () override
- void [SendEvent](#) (SDL_Event &event) override

Private Types

- enum **MenuOption** { **START** = 0 , **QUIT** = 1 }

Private Member Functions

- void [SelectorNext](#) ()
- void [SelectorPrev](#) ()
- void [Select](#) ()

Private Attributes

- [App](#) & [app](#)
- std::shared_ptr< [TexturedRect2D](#) > [m_titleText](#)
- std::shared_ptr< [TexturedRect2D](#) > [m_startText](#)
- std::shared_ptr< [TexturedRect2D](#) > [m_quitText](#)
- std::shared_ptr< [TexturedRect2D](#) > [m_selector](#)
- int [selector_pos](#)
- std::shared_ptr< [TexturedRect2D](#) > [m_instructionText](#)

Static Private Attributes

- static constexpr [SDL_Color](#) [TEXT_COLOR](#) = {255, 255, 255}
- static constexpr float [TITLE_W_SCALE](#) = 0.001
- static constexpr float [TITLE_H_SCALE](#) = 0.001
- static constexpr float [TITLE_X_OFFSET](#) = 0.0
- static constexpr float [TITLE_Y_OFFSET](#) = 1.0 / 4.0
- static constexpr float [MENU_W_SCALE](#) = 0.0004
- static constexpr float [MENU_H_SCALE](#) = 0.0004
- static constexpr float [START_X_OFFSET](#) = 0.0
- static constexpr float [START_Y_OFFSET](#) = 0.0
- static constexpr float [QUIT_X_OFFSET](#) = 0.0
- static constexpr float [QUIT_Y_OFFSET](#) = -1.0 / 8.0
- static constexpr int [MENU_COUNT](#) = 2
- static constexpr float [SELECTOR_PADDING](#) = 1.0 / 32.0
- static constexpr float [INSTRUCTION_W_SCALE](#) = 0.0002
- static constexpr float [INSTRUCTION_H_SCALE](#) = 0.0002
- static constexpr float [INSTRUCTION_X_OFFSET](#) = 0.3
- static constexpr float [INSTRUCTION_Y_OFFSET](#) = -0.4

6.9.1 Constructor & Destructor Documentation

6.9.1.1 MenuState()

```
MenuState::MenuState (
    App & app )
```

Mission Level State.

6.9.1.2 ~MenuState()

```
MenuState::~MenuState ( ) [override], [default]
```

Default desctructor.

6.9.2 Member Function Documentation

6.9.2.1 Run()

```
void MenuState::Run ( ) [override], [virtual]
```

Runs the state loop.

Implements [IState](#).

6.9.2.2 Select()

```
void MenuState::Select ( ) [private]
```

Executes menu item logic based on the currently selected item.

6.9.2.3 SelectorNext()

```
void MenuState::SelectorNext ( ) [private]
```

Moves the selector to the next menu item.

6.9.2.4 SelectorPrev()

```
void MenuState::SelectorPrev ( ) [private]
```

Moves the selector to the previous menu item.

6.9.2.5 SendEvent()

```
void MenuState::SendEvent (
    SDL_Event & event ) [override], [virtual]
```

Handles SDL Input.

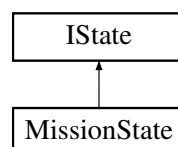
Implements [IState](#).

The documentation for this class was generated from the following files:

- include/MenuState.hpp
- src/MenuState.cpp

6.10 MissionState Class Reference

Inheritance diagram for MissionState:



Public Member Functions

- [MissionState](#) ([App](#) &app)
- [~MissionState](#) () override
- void [Run](#) () override
- void [SendEvent](#) (SDL_Event &event) override

Private Attributes

- [App](#) & app
- std::shared_ptr< [VimEmulator](#) > m_vimEmulator

Static Private Attributes

- static constexpr float **WIDTH_RATIO** = 2 / 3.0
- static constexpr int **DRAW_R** = 131
- static constexpr int **DRAW_G** = 50
- static constexpr int **DRAW_B** = 168

6.10.1 Constructor & Destructor Documentation

6.10.1.1 MissionState()

```
MissionState::MissionState (
    App & app )
```

Mission Level State.

6.10.1.2 ~MissionState()

```
MissionState::~MissionState ( ) [override], [default]
```

Destroy the terminal emulator on deletion.

6.10.2 Member Function Documentation

6.10.2.1 Run()

```
void MissionState::Run ( ) [override], [virtual]
```

Runs the state loop.

Implements [IState](#).

6.10.2.2 SendEvent()

```
void MissionState::SendEvent (
    SDL_Event & event ) [override], [virtual]
```

Handles SDL Input.

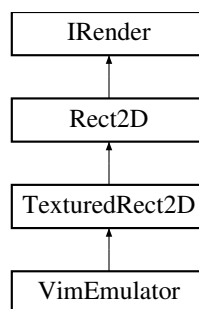
Implements [IState](#).

The documentation for this class was generated from the following files:

- include/MissionState.hpp
- src/MissionState.cpp

6.11 Rect2D Class Reference

Inheritance diagram for Rect2D:



Public Member Functions

- [Rect2D](#) (GLfloat xPos=0, GLfloat yPos=0, GLfloat width=0, GLfloat height=0)
- virtual void [UpdateVertexData](#) ()
- void [SetPosition](#) (GLfloat xPos, GLfloat yPos)
- void [SetSize](#) (GLfloat width, GLfloat height)
- GLfloat [GetX](#) () const
- GLfloat [GetY](#) () const
- GLfloat [GetWidth](#) () const
- GLfloat [GetHeight](#) () const
- std::vector< GLfloat > [GetVertexData](#) () const
- [~Rect2D](#) ()
- void [Render](#) () override
- void [UpdateGL](#) () override
- void [SetShaderProgram](#) (std::string shaderProgram)

Protected Member Functions

- void [GenBindBufferGL](#) () override

Protected Attributes

- `std::vector< GLfloat > m_vertexData`
- `std::vector< GLuint > m_indexData`
- `GLfloat m_x`
- `GLfloat m_y`
- `GLfloat m_width`
- `GLfloat m_height`
- `GLuint m_texture_width`
- `GLuint m_texture_height`

Protected Attributes inherited from [IRender](#)

- `GLuint m_vertexArrayObject`
- `GLuint m_vertexBufferObject`
- `GLuint m_indexBufferObject`
- `std::string m_shaderProgram`

Static Private Attributes

- `static constexpr int RECT_ELEMENTS = 6`

6.11.1 Constructor & Destructor Documentation**6.11.1.1 Rect2D()**

```
Rect2D::Rect2D (
    GLfloat xPos = 0,
    GLfloat yPos = 0,
    GLfloat width = 0,
    GLfloat height = 0 )
```

Representation a 2D quad, an encapsulation of two GL Triangles.

Parameters

<i>xPos</i>	X position
<i>yPos</i>	Y position
<i>width</i>	Width of the quad
<i>height</i>	Height of the quad

6.11.1.2 ~Rect2D()

```
Rect2D::~~Rect2D ( )
```

Cleans up OpenGL vertex arrays and buffers on destruction.

6.11.2 Member Function Documentation

6.11.2.1 GenBindBufferGL()

```
void Rect2D::GenBindBufferGL ( ) [override], [protected], [virtual]
```

Generates vertex arrays and buffers. Binds vertex arrays and buffers. Then initializes data stores for the vertex and index data.

Implements [IRender](#).

6.11.2.2 GetHeight()

```
GLfloat Rect2D::GetHeight ( ) const
```

Get the height of the quad.

Returns

Height of the quad

6.11.2.3 GetVertexData()

```
std::vector< GLfloat > Rect2D::GetVertexData ( ) const
```

Get the raw vertex data.

Returns

Vertex data as vector

6.11.2.4 GetWidth()

```
GLfloat Rect2D::GetWidth ( ) const
```

Get the width of the quad.

Returns

Width of the quad

6.11.2.5 GetX()

```
GLfloat Rect2D::GetX ( ) const
```

Get the X position of the quad.

Returns

X position of the quad

6.11.2.6 GetY()

```
GLfloat Rect2D::GetY ( ) const
```

Get the Y position of the quad.

Returns

Y position of the quad

6.11.2.7 Render()

```
void Rect2D::Render ( ) [override], [virtual]
```

Renders the quad.

Implements [IRender](#).

Reimplemented in [TexturedRect2D](#), and [VimEmulator](#).

6.11.2.8 SetPosition()

```
void Rect2D::SetPosition (
    GLfloat xPos,
    GLfloat yPos )
```

Set the position and update the vertex data.

Parameters

<i>xPos</i>	New x Position
<i>yPos</i>	New y Position

6.11.2.9 SetShaderProgram()

```
void Rect2D::SetShaderProgram (
    std::string shaderProgramName )
```

Set the shaderProgram to render this rectangle by name.

Parameters

<i>shaderProgram</i>	ShaderProgram name
----------------------	------------------------------------

6.11.2.10 SetSize()

```
void Rect2D::SetSize (
```

```
GLfloat width,
GLfloat height )
```

Set the width and height of the quad and update the vertex data.

Parameters

<i>width</i>	New quad width
<i>height</i>	New quad height

6.11.2.11 UpdateGL()

```
void Rect2D::UpdateGL ( ) [override], [virtual]
```

Updates the vertex arrays in the OpenGL FSM.

Implements [IRender](#).

Reimplemented in [TexturedRect2D](#).

6.11.2.12 UpdateVertexData()

```
void Rect2D::UpdateVertexData ( ) [virtual]
```

Updates the Vertex Data.

Note

This should be overridden if you want to provide more data in strides.

There are four groups, describing the four points of the quad.

```
void Rect2D::UpdateVertexData() {
    m_vertexData = {TL, BR, BL, TR}
    \\ \\ Always UpdateGL after changing vertex data.
    UpdateGL();
}
```

Reimplemented in [TexturedRect2D](#).

The documentation for this class was generated from the following files:

- include/Rect2D.hpp
- src/Rect2D.cpp

6.12 SDLToX11Keymap Class Reference

Public Member Functions

- [SDLToX11Keymap](#) ()
- KeySym [convert](#) (SDL_Keycode sdlKey) const

Private Attributes

- `std::unordered_map< SDL_Keycode, KeySym > m_keyMap`

6.12.1 Constructor & Destructor Documentation**6.12.1.1 SDLToX11Keymap()**

```
SDLToX11Keymap::SDLToX11Keymap ( )
```

SDL To X11 Key Hashmap.

6.12.2 Member Function Documentation**6.12.2.1 convert()**

```
KeySym SDLToX11Keymap::convert (
    SDL_Keycode sdlKey ) const
```

Converts an SDL_Keycode to an X11 KeySym.

Parameters

<i>SDL_Keycode</i>	
--------------------	--

Returns

X11 KeySym

The documentation for this class was generated from the following files:

- `include/SDLToX11Keymap.hpp`
- `src/SDLToX11Keymap.cpp`

6.13 ShaderProgram Class Reference**Classes**

- `struct _passKey_t`

Public Member Functions

- `~ShaderProgram ()`
- `GLuint GetProgramId () const`
- `ShaderProgram (_passKey_t, GLuint program_id)`

Private Attributes

- const GLuint **m_program_id** = 0

Friends

- class **ShaderProgramBuilder**

6.13.1 Constructor & Destructor Documentation

6.13.1.1 ~ShaderProgram()

```
ShaderProgram::~ShaderProgram ( )
```

Cleans up the program on destruction.

6.13.1.2 ShaderProgram()

```
ShaderProgram::ShaderProgram (
    _passKey_t _passKey,
    GLuint program_id )
```

Encapsulation of an OpenGL program.

Parameters

<i>passkey</i>	An instance of the _passKey_t struct. This restricts the constructor usage.
<i>program_id</i>	The OpenGL shader program ID.

The passkey is used to restrict instantiation of [ShaderProgram](#) objects. Only classes that have access to the [_passKey_t](#) struct can create [ShaderProgram](#) objects.

6.13.2 Member Function Documentation

6.13.2.1 GetProgramId()

```
GLuint ShaderProgram::GetProgramId ( ) const
```

Get the programID.

Returns

OpenGL Program ID as GLuint.

The documentation for this class was generated from the following files:

- include/ShaderProgram.hpp
- src/ShaderProgram.cpp

6.14 ShaderProgramBuilder Class Reference

Public Member Functions

- [~ShaderProgramBuilder](#) ()
- [ShaderProgramBuilder](#) & [LoadShaderFile](#) (GLenum type, std::string file)
- [ShaderProgramBuilder](#) & [LoadShader](#) (GLenum type, std::string shaderCode)
- std::variant< [Error](#), std::unique_ptr< [ShaderProgram](#) > > [GenerateShaderProgram](#) ()

Private Attributes

- std::vector< GLuint > **m_shaders**
- std::optional< [Error](#) > **m_error** = std::nullopt

6.14.1 Constructor & Destructor Documentation

6.14.1.1 ~ShaderProgramBuilder()

```
ShaderProgramBuilder::~~ShaderProgramBuilder ( )
```

Cleans up shaders on destruction.

6.14.2 Member Function Documentation

6.14.2.1 GenerateShaderProgram()

```
std::variant< Error, std::unique_ptr< ShaderProgram > > ShaderProgramBuilder::GenerateShaderProgram ( )
```

Links Shaders and compiles the shader program. Cleans up shaders after building.

Returns

A std::variant<Error, std::unique_ptr<ShaderProgram>> containing the shader.

The return type can be one of the following:

- [Error](#) if generation fails, contains error message generated by OpenGL.
- std::unique_ptr<[ShaderProgram](#)> A unique pointer to the generated OpenGL program. Encapsulated in the [ShaderProgram](#) class.

6.14.2.2 LoadShader()

```
ShaderProgramBuilder & ShaderProgramBuilder::LoadShader (
    GLenum type,
    std::string shaderCode )
```

Loads shader by string, ready for binding when the program is generated.

Parameters

<i>type</i>	Shader type
<i>shaderCode</i>	Shader code

Returns

A reference to the current instance of [ShaderProgramBuilder](#)

6.14.2.3 LoadShaderFile()

```
ShaderProgramBuilder & ShaderProgramBuilder::LoadShaderFile (
    GLenum type,
    std::string filePath )
```

Loads shader by file, ready for binding when the program is generated.

Parameters

<i>type</i>	Shader type
<i>filePath</i>	Path to shader file (*.frag, *.vert)

Returns

A reference to the current instance of [ShaderProgramBuilder](#)

The documentation for this class was generated from the following files:

- include/ShaderProgramBuilder.hpp
- src/ShaderProgramBuilder.cpp

6.15 IntroState::Slide Struct Reference

Public Attributes

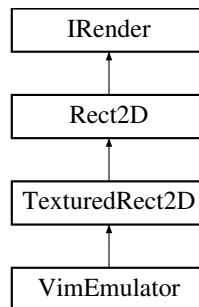
- SDL_Surface * **slideSurface**
- float **startX**
- float **stopX**
- float **startY**
- float **stopY**
- float **panSpeedMS**

The documentation for this struct was generated from the following file:

- include/IntroState.hpp

6.16 TexturedRect2D Class Reference

Inheritance diagram for TexturedRect2D:



Public Member Functions

- [TexturedRect2D](#) (GLfloat xPos=0, GLfloat yPos=0, GLfloat width=0, GLfloat height=0)
- void [UpdateVertexData](#) () override
- void [SetTexture](#) (void *data, GLuint width, GLuint height, bool force=false)
- void [SetTextureFormat](#) (GLenum format)
- void [SetInternalTextureFormat](#) (GLenum format)
- void [EnableTextureBlend](#) ()
- void [DisableTextureBlend](#) ()
- [~TexturedRect2D](#) ()
- void [Render](#) () override
- void [UpdateGL](#) () override

Public Member Functions inherited from [Rect2D](#)

- [Rect2D](#) (GLfloat xPos=0, GLfloat yPos=0, GLfloat width=0, GLfloat height=0)
- void [SetPosition](#) (GLfloat xPos, GLfloat yPos)
- void [SetSize](#) (GLfloat width, GLfloat height)
- GLfloat [GetX](#) () const
- GLfloat [GetY](#) () const
- GLfloat [GetWidth](#) () const
- GLfloat [GetHeight](#) () const
- std::vector< GLfloat > [GetVertexData](#) () const
- [~Rect2D](#) ()
- void [SetShaderProgram](#) (std::string shaderProgram)

Protected Attributes

- GLuint **m_texture**
- GLuint **m_texture_width**
- GLuint **m_texture_height**

Protected Attributes inherited from [Rect2D](#)

- `std::vector< GLfloat > m_vertexData`
- `std::vector< GLuint > m_indexData`
- `GLfloat m_x`
- `GLfloat m_y`
- `GLfloat m_width`
- `GLfloat m_height`
- `GLuint m_texture_width`
- `GLuint m_texture_height`

Protected Attributes inherited from [IRender](#)

- `GLuint m_vertexArrayObject`
- `GLuint m_vertexBufferObject`
- `GLuint m_indexBufferObject`
- `std::string m_shaderProgram`

Private Attributes

- `GLenum m_textureFormat`
- `GLenum m_internalTextureFormat`
- `bool m_enableBlend`

Additional Inherited Members

Protected Member Functions inherited from [Rect2D](#)

- `void GenBindBufferGL ()` override

6.16.1 Constructor & Destructor Documentation

6.16.1.1 `TexturedRect2D()`

```
TexturedRect2D::TexturedRect2D (
    GLfloat xPos = 0,
    GLfloat yPos = 0,
    GLfloat width = 0,
    GLfloat height = 0 )
```

An extension of [Rect2D](#), representing a 2D quad with a texture.

Parameters

<i>xPos</i>	X position
<i>yPos</i>	Y position
<i>width</i>	Width of the quad
<i>height</i>	Height of the quad

6.16.1.2 ~TexturedRect2D()

```
TexturedRect2D::~~TexturedRect2D ( )
```

Deletes the texture on destruction.

6.16.2 Member Function Documentation

6.16.2.1 DisableTextureBlend()

```
void TexturedRect2D::DisableTextureBlend ( )
```

Disables texture blending.

6.16.2.2 EnableTextureBlend()

```
void TexturedRect2D::EnableTextureBlend ( )
```

Enabled texture blending.

6.16.2.3 Render()

```
void TexturedRect2D::Render ( ) [override], [virtual]
```

Renders the Textured Quad. Currently uses the sp_cathode shader.

Todo Gives texture rects a default shader program and method to select others.

Reimplemented from [Rect2D](#).

Reimplemented in [VimEmulator](#).

6.16.2.4 SetInternalTextureFormat()

```
void TexturedRect2D::SetInternalTextureFormat (
    GLenum format )
```

Sets the internal texture format for the glTexImage.

6.16.2.5 SetTexture()

```
void TexturedRect2D::SetTexture (
    void * data,
    GLuint width,
    GLuint height,
    bool force = false )
```

Sets the texture with raw pixel data. Unless forced or the new texture has a different width and/or height will sub-image the texture rather than generate a new one. This helps for texture streaming such as in [VimEmulator](#).

Parameters

<i>data</i>	Pointer to pixel data
<i>width</i>	Width of the texture in pixels
<i>height</i>	Height of the texture in pixels
<i>force</i>	If set to true forces the texture to be re-generated

6.16.2.6 SetTextureFormat()

```
void TexturedRect2D::SetTextureFormat (
    GLenum format )
```

Sets the format for the pixel data given to SetTexture.

6.16.2.7 UpdateGL()

```
void TexturedRect2D::UpdateGL ( ) [override], [virtual]
```

Updates the vertex arrays in the OpenGL FSM. Uses strides to get texture data as well as coordinate data.

Reimplemented from [Rect2D](#).

6.16.2.8 UpdateVertexData()

```
void TexturedRect2D::UpdateVertexData ( ) [override], [virtual]
```

Updates the vertex data. Adds two more data points to the strides defining the texture coordinates.

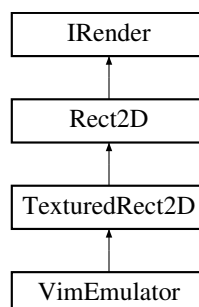
Reimplemented from [Rect2D](#).

The documentation for this class was generated from the following files:

- include/TexturedRect2D.hpp
- src/TexturedRect2D.cpp

6.17 VimEmulator Class Reference

Inheritance diagram for VimEmulator:



Public Member Functions

- [VimEmulator](#) (std::string terminal, std::string nArg)
- virtual [~VimEmulator](#) ()
- void [RegisterWindow](#) ()
- void [ResizeWindow](#) (int w, int h)
- void [QueueFrame](#) ()
- void [SendSDLKey](#) (SDL_Keycode key)
- void [SetSDLMod](#) (SDL_Keymod mod)
- void [Render](#) () override

Render cycle for a new frame.

Public Member Functions inherited from [TexturedRect2D](#)

- [TexturedRect2D](#) (GLfloat xPos=0, GLfloat yPos=0, GLfloat width=0, GLfloat height=0)
- void [UpdateVertexData](#) () override
- void [SetTexture](#) (void *data, GLuint width, GLuint height, bool force=false)
- void [SetTextureFormat](#) (GLenum format)
- void [SetInternalTextureFormat](#) (GLenum format)
- void [EnableTextureBlend](#) ()
- void [DisableTextureBlend](#) ()
- [~TexturedRect2D](#) ()
- void [UpdateGL](#) () override

Public Member Functions inherited from [Rect2D](#)

- [Rect2D](#) (GLfloat xPos=0, GLfloat yPos=0, GLfloat width=0, GLfloat height=0)
- void [SetPosition](#) (GLfloat xPos, GLfloat yPos)
- void [SetSize](#) (GLfloat width, GLfloat height)
- GLfloat [GetX](#) () const
- GLfloat [GetY](#) () const
- GLfloat [GetWidth](#) () const
- GLfloat [GetHeight](#) () const
- std::vector< GLfloat > [GetVertexData](#) () const
- [~Rect2D](#) ()
- void [SetShaderProgram](#) (std::string shaderProgram)

Private Member Functions

- SDL_Surface * [GetFrameAsSurface](#) ()
- Window * [findWindowByName](#) (Window window)
- void [RegisterWindowThread](#) ()
- void [ResizeWindowThread](#) (int w, int h)
- bool [MatchResizeEvent](#) (int w, int h, XEvent *event)
- void [QueueFrameThread](#) ()

Private Attributes

- Display * **m_display**
- int **m_screen**
- Window **m_rootWindow**
- Window * **m_window**
- std::string **m_windowName**
- int **m_width**
- int **m_height**
- std::mutex **m_mutex**
- pid_t **m_pid**
- XImage * **m_xImage**
- SDL_Surface * **m_surface**
- bool **m_frameReady**
- unsigned int * **m_modmask**

Static Private Attributes

- static constexpr int **REFRESH_MS** = 100
- static constexpr Uint32 **R_MASK** = 0xFF0000
- static constexpr Uint32 **G_MASK** = 0x00FF00
- static constexpr Uint32 **B_MASK** = 0x0000FF
- static constexpr Uint32 **A_MASK** = 0

Additional Inherited Members

Protected Member Functions inherited from [Rect2D](#)

- void [GenBindBufferGL](#) () override

Protected Attributes inherited from [TexturedRect2D](#)

- GLuint **m_texture**
- GLuint **m_texture_width**
- GLuint **m_texture_height**

Protected Attributes inherited from [Rect2D](#)

- std::vector< GLfloat > **m_vertexData**
- std::vector< GLuint > **m_indexData**
- GLfloat **m_x**
- GLfloat **m_y**
- GLfloat **m_width**
- GLfloat **m_height**
- GLuint **m_texture_width**
- GLuint **m_texture_height**

Protected Attributes inherited from [IRender](#)

- GLuint **m_vertexArrayObject**
- GLuint **m_vertexBufferObject**
- GLuint **m_indexBufferObject**
- std::string **m_shaderProgram**

6.17.1 Constructor & Destructor Documentation

6.17.1.1 VimEmulator()

```
VimEmulator::VimEmulator (
    std::string terminal,
    std::string nArg )
```

A Vim Terminal Instance.

Parameters

<i>terminal</i>	The name of the terminal program, E.g. 'Alacrity'
<i>nArg</i>	Arguments for launching vim

6.17.1.2 ~VimEmulator()

```
VimEmulator::~~VimEmulator ( ) [virtual]
```

Destroys the terminal and pointer vars.

6.17.2 Member Function Documentation

6.17.2.1 findWindowByName()

```
Window * VimEmulator::findWindowByName (
    Window window ) [private]
```

Finds and returns a window on the current system. Returns nullptr if not found.

Parameters

<i>window</i>	X11 Window to find.
---------------	---------------------

Returns

Found Window or nullptr

6.17.2.2 GetFrameAsSurface()

```
SDL_Surface * VimEmulator::GetFrameAsSurface ( ) [private]
```

Private Methods

Private Methods Returns the current surface and marks the current frame as not ready.

Returns

SDL_Surface

6.17.2.3 MatchResizeEvent()

```
bool VimEmulator::MatchResizeEvent (
    int w,
    int h,
    XEvent * event ) [private]
```

Looks for an X11 ConfigureNotify event matching the given width and height.

Parameters

<i>w</i>	Width
<i>h</i>	Height
<i>event</i>	X11 Event

Returns

True if match is found

6.17.2.4 QueueFrame()

```
void VimEmulator::QueueFrame ( )
```

Calls QueueFrame Logic in a new thread if the previous frame hasn't been used.

6.17.2.5 QueueFrameThread()

```
void VimEmulator::QueueFrameThread ( ) [private]
```

Requests the current terminal frame as an image.

6.17.2.6 RegisterWindow()

```
void VimEmulator::RegisterWindow ( )
```

Calls RegisterWindow Logic in a new thread.

6.17.2.7 RegisterWindowThread()

```
void VimEmulator::RegisterWindowThread ( ) [private]
```

Assigns the terminal window and window image to variables.

6.17.2.8 Render()

```
void VimEmulator::Render ( ) [override], [virtual]
```

Render cycle for a new frame.

[IRender](#)

[IRender](#)

If a frame is not ready it will request a new frame from the terminal. It will render the previous frame until a new one is found.

Parameters

<i>renderer</i>	SDL_Renderer context
-----------------	----------------------

Reimplemented from [TexturedRect2D](#).

6.17.2.9 ResizeWindow()

```
void VimEmulator::ResizeWindow (
    int w,
    int h )
```

Calls ResizeWindow Logic in a new thread.

Parameters

<i>w</i>	Width
<i>h</i>	Height

6.17.2.10 ResizeWindowThread()

```
void VimEmulator::ResizeWindowThread (
    int w,
    int h ) [private]
```

Performs a resize on the terminal window and re-registers it.

Parameters

<i>w</i>	Width
<i>h</i>	Height

6.17.2.11 SendSDLKey()

```
void VimEmulator::SendSDLKey (
    SDL_Keycode key )
```

X11 Keyboard

Translates an SDL_Keycode and sends it to the terminal.

Parameters

<i>key</i>	An SDL_Keycode
------------	----------------

6.17.2.12 SetSDLMod()

```
void VimEmulator::SetSDLMod (
    SDL_Keymod mod )
```

Sets the modmask using an SDL_Keymod.

Parameters

<i>mod</i>	An SDL_Keymod
------------	---------------

6.17.3 Member Data Documentation

6.17.3.1 m_modmask

```
unsigned int* VimEmulator::m_modmask [private]
```

XKeyboard

6.17.3.2 R_MASK

```
constexpr Uint32 VimEmulator::R_MASK = 0xFF0000 [static], [constexpr], [private]
```

Frames

The documentation for this class was generated from the following files:

- include/VimEmulator.hpp
- src/VimEmulator.cpp

Chapter 7

File Documentation

7.1 App.hpp

```
00001 #ifndef APP_HPP
00002 #define APP_HPP
00003
00004 #include "IRender.hpp"
00005 #include "IState.hpp"
00006 #include <SDL2/SDL.h>
00007 #include <SDL2/SDL_ttf.h>
00008 #include <glad/glad.h>
00009 #include <memory>
00010 #include <vector>
00011
00012 class App {
00013 public:
00014     App(Uint32 ssFlags, int x, int y, int w, int h);
00015
00016     ~App();
00017
00018     SDL_Renderer *GetRenderer();
00019
00020     void SetState(IState *state);
00021
00022     void Run();
00023
00024     void Stop();
00025
00026     void Render();
00027
00028     void AddRenderable(std::shared_ptr<IRender> const &renderable);
00029
00030     void ClearRenderables();
00031
00032     [[nodiscard]] double GetFPS() const;
00033
00034     [[nodiscard]] double DeltaTime() const;
00035
00036     [[nodiscard]] int GetWidth() const;
00037
00038     [[nodiscard]] int GetHeight() const;
00039
00040 private:
00041     SDL_Window *m_window;
00042     SDL_Renderer *m_renderer;
00043     std::vector<std::shared_ptr<IRender>> m_renderables;
00044
00045     int m_width;
00046     int m_height;
00047     bool m_running;
00048
00049     IState *m_state;
00050
00051     static constexpr double SECOND_MS = 1000.0;
00052     Uint32 m_startTick;
00053     Uint32 m_endTick;
00054     double m_deltaTime;
00055
00056     // Shaders
00057     static constexpr int APP_GL_DEPTH_SIZE = 24;
00058     static void debugMessage(GLenum source, GLenum type, GLuint debug_id,
```

```

00064             GLenum severity, GLsizei length,
00065             const GLchar *message, const void *userParam);
00066     static void CreateGraphicsPipeline();
00067
00068     // Fonts
00069     static constexpr int FONT_RENDER_SIZE = 128;
00070     static void GenerateFonts();
00071
00072     // Draw
00073     void PreDraw() const;
00074 };
00075
00076 #endif

```

7.2 constants.hpp

```

00001 #ifndef CONSTANTS_HPP
00002 #define CONSTANTS_HPP
00003
00004 #include "SDLToX11Keymap.hpp"
00005
00006 constexpr char APP_TITLE[] = "VimVentures";
00007 constexpr char APP_VERSION[] = "MEDIEVAL 0.0.3";
00008
00009 extern const SDLToX11Keymap &SDLX11KeymapRef;
00010
00011 #endif

```

7.3 Error.hpp

```

00001 #ifndef ERROR_HPP
00002 #define ERROR_HPP
00003
00004 #include <string>
00005
00006 struct Error {
00007     const int m_infoLogLen;
00008     const char *m_infoLog;
00009     Error(int infoLogLen, const char *infoLog);
00010     [[nodiscard]] std::string toString() const;
00011 };
00012
00013 #endif

```

7.4 glad.h

```

00001 /*
00002
00003     OpenGL loader generated by glad 0.1.36 on Fri Aug  2 12:24:15 2024.
00004
00005     Language/Generator: C/C++
00006     Specification: gl
00007     APIs: gl=4.1
00008     Profile: core
00009     Extensions:
00010         GL_3DFX_multisample,
00011         GL_3DFX_tbuffer,
00012         GL_3DFX_texture_compression_FXT1,
00013         GL_AMD_blend_minmax_factor,
00014         GL_AMD_conservative_depth,
00015         GL_AMD_debug_output,
00016         GL_AMD_depth_clamp_separate,
00017         GL_AMD_draw_buffers_blend,
00018         GL_AMD_framebuffer_multisample_advanced,
00019         GL_AMD_framebuffer_sample_positions,
00020         GL_AMD_gcn_shader,
00021         GL_AMD_gpu_shader_half_float,
00022         GL_AMD_gpu_shader_int16,
00023         GL_AMD_gpu_shader_int64,
00024         GL_AMD_interleaved_elements,
00025         GL_AMD_multi_draw_indirect,
00026         GL_AMD_name_gen_delete,
00027         GL_AMD_occlusion_query_event,
00028         GL_AMD_performance_monitor,
00029         GL_AMD_pinned_memory,

```

```
00030     GL_AMD_query_buffer_object,
00031     GL_AMD_sample_positions,
00032     GL_AMD_seamless_cubemap_per_texture,
00033     GL_AMD_shader_atomic_counter_ops,
00034     GL_AMD_shader_ballot,
00035     GL_AMD_shader_explicit_vertex_parameter,
00036     GL_AMD_shader_gpu_shader_half_float_fetch,
00037     GL_AMD_shader_image_load_store_lod,
00038     GL_AMD_shader_stencil_export,
00039     GL_AMD_shader_trinary_minmax,
00040     GL_AMD_sparse_texture,
00041     GL_AMD_stencil_operation_extended,
00042     GL_AMD_texture_gather_bias_lod,
00043     GL_AMD_texture_texture4,
00044     GL_AMD_transform_feedback3_lines_triangles,
00045     GL_AMD_transform_feedback4,
00046     GL_AMD_vertex_shader_layer,
00047     GL_AMD_vertex_shader_tessellator,
00048     GL_AMD_vertex_shader_viewport_index,
00049     GL_APPLE_aux_depth_stencil,
00050     GL_APPLE_client_storage,
00051     GL_APPLE_element_array,
00052     GL_APPLE_fence,
00053     GL_APPLE_float_pixels,
00054     GL_APPLE_flush_buffer_range,
00055     GL_APPLE_object_purgeable,
00056     GL_APPLE_rgb_422,
00057     GL_APPLE_row_bytes,
00058     GL_APPLE_specular_vector,
00059     GL_APPLE_texture_range,
00060     GL_APPLE_transform_hint,
00061     GL_APPLE_vertex_array_object,
00062     GL_APPLE_vertex_array_range,
00063     GL_APPLE_vertex_program_evaluators,
00064     GL_APPLE_ycbcr_422,
00065     GL_ARB_ES2_compatibility,
00066     GL_ARB_ES3_1_compatibility,
00067     GL_ARB_ES3_2_compatibility,
00068     GL_ARB_ES3_compatibility,
00069     GL_ARB_arrays_of_arrays,
00070     GL_ARB_base_instance,
00071     GL_ARB_bindless_texture,
00072     GL_ARB_blend_func_extended,
00073     GL_ARB_buffer_storage,
00074     GL_ARB_cl_event,
00075     GL_ARB_clear_buffer_object,
00076     GL_ARB_clear_texture,
00077     GL_ARB_clip_control,
00078     GL_ARB_color_buffer_float,
00079     GL_ARB_compatibility,
00080     GL_ARB_compressed_texture_pixel_storage,
00081     GL_ARB_compute_shader,
00082     GL_ARB_compute_variable_group_size,
00083     GL_ARB_conditional_render_inverted,
00084     GL_ARB_conservative_depth,
00085     GL_ARB_copy_buffer,
00086     GL_ARB_copy_image,
00087     GL_ARB_cull_distance,
00088     GL_ARB_debug_output,
00089     GL_ARB_depth_buffer_float,
00090     GL_ARB_depth_clamp,
00091     GL_ARB_depth_texture,
00092     GL_ARB_derivative_control,
00093     GL_ARB_direct_state_access,
00094     GL_ARB_draw_buffers,
00095     GL_ARB_draw_buffers_blend,
00096     GL_ARB_draw_elements_base_vertex,
00097     GL_ARB_draw_indirect,
00098     GL_ARB_draw_instanced,
00099     GL_ARB_enhanced_layouts,
00100     GL_ARB_explicit_attrib_location,
00101     GL_ARB_explicit_uniform_location,
00102     GL_ARB_fragment_coord_conventions,
00103     GL_ARB_fragment_layer_viewport,
00104     GL_ARB_fragment_program,
00105     GL_ARB_fragment_program_shadow,
00106     GL_ARB_fragment_shader,
00107     GL_ARB_fragment_shader_interlock,
00108     GL_ARB_framebuffer_no_attachments,
00109     GL_ARB_framebuffer_object,
00110     GL_ARB_framebuffer_sRGB,
00111     GL_ARB_geometry_shader4,
00112     GL_ARB_get_program_binary,
00113     GL_ARB_get_texture_sub_image,
00114     GL_ARB_gl_spirv,
00115     GL_ARB_gpu_shader5,
00116     GL_ARB_gpu_shader_fp64,
```

```
00117     GL_ARB_gpu_shader_int64,
00118     GL_ARB_half_float_pixel,
00119     GL_ARB_half_float_vertex,
00120     GL_ARB_imaging,
00121     GL_ARB_indirect_parameters,
00122     GL_ARB_instanced_arrays,
00123     GL_ARB_internalformat_query,
00124     GL_ARB_internalformat_query2,
00125     GL_ARB_invalidate_subdata,
00126     GL_ARB_map_buffer_alignment,
00127     GL_ARB_map_buffer_range,
00128     GL_ARB_matrix_palette,
00129     GL_ARB_multi_bind,
00130     GL_ARB_multi_draw_indirect,
00131     GL_ARB_multisample,
00132     GL_ARB_multitexture,
00133     GL_ARB_occlusion_query,
00134     GL_ARB_occlusion_query2,
00135     GL_ARB_parallel_shader_compile,
00136     GL_ARB_pipeline_statistics_query,
00137     GL_ARB_pixel_buffer_object,
00138     GL_ARB_point_parameters,
00139     GL_ARB_point_sprite,
00140     GL_ARB_polygon_offset_clamp,
00141     GL_ARB_post_depth_coverage,
00142     GL_ARB_program_interface_query,
00143     GL_ARB_provoking_vertex,
00144     GL_ARB_query_buffer_object,
00145     GL_ARB_robust_buffer_access_behavior,
00146     GL_ARB_robustness,
00147     GL_ARB_robustness_isolation,
00148     GL_ARB_sample_locations,
00149     GL_ARB_sample_shading,
00150     GL_ARB_sampler_objects,
00151     GL_ARB_seamless_cube_map,
00152     GL_ARB_seamless_cubemap_per_texture,
00153     GL_ARB_separate_shader_objects,
00154     GL_ARB_shader_atomic_counter_ops,
00155     GL_ARB_shader_atomic_counters,
00156     GL_ARB_shader_ballot,
00157     GL_ARB_shader_bit_encoding,
00158     GL_ARB_shader_clock,
00159     GL_ARB_shader_draw_parameters,
00160     GL_ARB_shader_group_vote,
00161     GL_ARB_shader_image_load_store,
00162     GL_ARB_shader_image_size,
00163     GL_ARB_shader_objects,
00164     GL_ARB_shader_precision,
00165     GL_ARB_shader_stencil_export,
00166     GL_ARB_shader_storage_buffer_object,
00167     GL_ARB_shader_subroutine,
00168     GL_ARB_shader_texture_image_samples,
00169     GL_ARB_shader_texture_lod,
00170     GL_ARB_shader_viewport_layer_array,
00171     GL_ARB_shading_language_100,
00172     GL_ARB_shading_language_420pack,
00173     GL_ARB_shading_language_include,
00174     GL_ARB_shading_language_packing,
00175     GL_ARB_shadow,
00176     GL_ARB_shadow_ambient,
00177     GL_ARB_sparse_buffer,
00178     GL_ARB_sparse_texture,
00179     GL_ARB_sparse_texture2,
00180     GL_ARB_sparse_texture_clamp,
00181     GL_ARB_spirv_extensions,
00182     GL_ARB_stencil_texturing,
00183     GL_ARB_sync,
00184     GL_ARB_tessellation_shader,
00185     GL_ARB_texture_barrier,
00186     GL_ARB_texture_border_clamp,
00187     GL_ARB_texture_buffer_object,
00188     GL_ARB_texture_buffer_object_rgb32,
00189     GL_ARB_texture_buffer_range,
00190     GL_ARB_texture_compression,
00191     GL_ARB_texture_compression_bptc,
00192     GL_ARB_texture_compression_rgtc,
00193     GL_ARB_texture_cube_map,
00194     GL_ARB_texture_cube_map_array,
00195     GL_ARB_texture_env_add,
00196     GL_ARB_texture_env_combine,
00197     GL_ARB_texture_env_crossbar,
00198     GL_ARB_texture_env_dot3,
00199     GL_ARB_texture_filter_anisotropic,
00200     GL_ARB_texture_filter_minmax,
00201     GL_ARB_texture_float,
00202     GL_ARB_texture_gather,
00203     GL_ARB_texture_mirror_clamp_to_edge,
```



```
00204     GL_ARB_texture_mirrored_repeat,
00205     GL_ARB_texture_multisample,
00206     GL_ARB_texture_non_power_of_two,
00207     GL_ARB_texture_query_levels,
00208     GL_ARB_texture_query_lod,
00209     GL_ARB_texture_rectangle,
00210     GL_ARB_texture_rg,
00211     GL_ARB_texture_rgb10_a2ui,
00212     GL_ARB_texture_stencil8,
00213     GL_ARB_texture_storage,
00214     GL_ARB_texture_storage_multisample,
00215     GL_ARB_texture_swizzle,
00216     GL_ARB_texture_view,
00217     GL_ARB_timer_query,
00218     GL_ARB_transform_feedback2,
00219     GL_ARB_transform_feedback3,
00220     GL_ARB_transform_feedback_instanced,
00221     GL_ARB_transform_feedback_overflow_query,
00222     GL_ARB_transpose_matrix,
00223     GL_ARB_uniform_buffer_object,
00224     GL_ARB_vertex_array_bgra,
00225     GL_ARB_vertex_array_object,
00226     GL_ARB_vertex_attrib_64bit,
00227     GL_ARB_vertex_attrib_binding,
00228     GL_ARB_vertex_blend,
00229     GL_ARB_vertex_buffer_object,
00230     GL_ARB_vertex_program,
00231     GL_ARB_vertex_shader,
00232     GL_ARB_vertex_type_10f_11f_11f_rev,
00233     GL_ARB_vertex_type_2_10_10_10_rev,
00234     GL_ARB_viewport_array,
00235     GL_ARB_window_pos,
00236     GL_ATI_draw_buffers,
00237     GL_ATI_element_array,
00238     GL_ATI_envmap_bumpmap,
00239     GL_ATI_fragment_shader,
00240     GL_ATI_map_object_buffer,
00241     GL_ATI_meminfo,
00242     GL_ATI_pixel_format_float,
00243     GL_ATI_pn_triangles,
00244     GL_ATI_separate_stencil,
00245     GL_ATI_text_fragment_shader,
00246     GL_ATI_texture_env_combine3,
00247     GL_ATI_texture_float,
00248     GL_ATI_texture_mirror_once,
00249     GL_ATI_vertex_array_object,
00250     GL_ATI_vertex_attrib_array_object,
00251     GL_ATI_vertex_streams,
00252     GL_EXT_422_pixels,
00253     GL_EXT_EGL_image_storage,
00254     GL_EXT_EGL_sync,
00255     GL_EXT_abgr,
00256     GL_EXT_bgra,
00257     GL_EXT_bindable_uniform,
00258     GL_EXT_blend_color,
00259     GL_EXT_blend_equation_separate,
00260     GL_EXT_blend_func_separate,
00261     GL_EXT_blend_logic_op,
00262     GL_EXT_blend_minmax,
00263     GL_EXT_blend_subtract,
00264     GL_EXT_clip_volume_hint,
00265     GL_EXT_cmyka,
00266     GL_EXT_color_subtable,
00267     GL_EXT_compiled_vertex_array,
00268     GL_EXT_convolution,
00269     GL_EXT_coordinate_frame,
00270     GL_EXT_copy_texture,
00271     GL_EXT_cull_vertex,
00272     GL_EXT_debug_label,
00273     GL_EXT_debug_marker,
00274     GL_EXT_depth_bounds_test,
00275     GL_EXT_direct_state_access,
00276     GL_EXT_draw_buffers2,
00277     GL_EXT_draw_instanced,
00278     GL_EXT_draw_range_elements,
00279     GL_EXT_external_buffer,
00280     GL_EXT_fog_coord,
00281     GL_EXT_framebuffer_blit,
00282     GL_EXT_framebuffer_blit_layers,
00283     GL_EXT_framebuffer_multisample,
00284     GL_EXT_framebuffer_multisample_blit_scaled,
00285     GL_EXT_framebuffer_object,
00286     GL_EXT_framebuffer_sRGB,
00287     GL_EXT_geometry_shader4,
00288     GL_EXT_gpu_program_parameters,
00289     GL_EXT_gpu_shader4,
00290     GL_EXT_histogram,
```

```
00291     GL_EXT_index_array_formats,
00292     GL_EXT_index_func,
00293     GL_EXT_index_material,
00294     GL_EXT_index_texture,
00295     GL_EXT_light_texture,
00296     GL_EXT_memory_object,
00297     GL_EXT_memory_object_fd,
00298     GL_EXT_memory_object_win32,
00299     GL_EXT_misc_attribute,
00300     GL_EXT_multi_draw_arrays,
00301     GL_EXT_multisample,
00302     GL_EXT_multiview_tessellation_geometry_shader,
00303     GL_EXT_multiview_texture_multisample,
00304     GL_EXT_multiview_timer_query,
00305     GL_EXT_packed_depth_stencil,
00306     GL_EXT_packed_float,
00307     GL_EXT_packed_pixels,
00308     GL_EXT_paletted_texture,
00309     GL_EXT_pixel_buffer_object,
00310     GL_EXT_pixel_transform,
00311     GL_EXT_pixel_transform_color_table,
00312     GL_EXT_point_parameters,
00313     GL_EXT_polygon_offset,
00314     GL_EXT_polygon_offset_clamp,
00315     GL_EXT_post_depth_coverage,
00316     GL_EXT_provoking_vertex,
00317     GL_EXT_raster_multisample,
00318     GL_EXT_rescale_normal,
00319     GL_EXT_secondary_color,
00320     GL_EXT_semaphore,
00321     GL_EXT_semaphore_fd,
00322     GL_EXT_semaphore_win32,
00323     GL_EXT_separate_shader_objects,
00324     GL_EXT_separate_specular_color,
00325     GL_EXT_shader_framebuffer_fetch,
00326     GL_EXT_shader_framebuffer_fetch_non_coherent,
00327     GL_EXT_shader_image_load_formatted,
00328     GL_EXT_shader_image_load_store,
00329     GL_EXT_shader_integer_mix,
00330     GL_EXT_shader_samples_identical,
00331     GL_EXT_shadow_funcs,
00332     GL_EXT_shared_texture_palette,
00333     GL_EXT_sparse_texture2,
00334     GL_EXT_stencil_clear_tag,
00335     GL_EXT_stencil_two_side,
00336     GL_EXT_stencil_wrap,
00337     GL_EXT_subtexture,
00338     GL_EXT_texture,
00339     GL_EXT_texture3D,
00340     GL_EXT_texture_array,
00341     GL_EXT_texture_buffer_object,
00342     GL_EXT_texture_compression_latc,
00343     GL_EXT_texture_compression_rgtc,
00344     GL_EXT_texture_compression_s3tc,
00345     GL_EXT_texture_cube_map,
00346     GL_EXT_texture_env_add,
00347     GL_EXT_texture_env_combine,
00348     GL_EXT_texture_env_dot3,
00349     GL_EXT_texture_filter_anisotropic,
00350     GL_EXT_texture_filter_minmax,
00351     GL_EXT_texture_integer,
00352     GL_EXT_texture_lod_bias,
00353     GL_EXT_texture_mirror_clamp,
00354     GL_EXT_texture_object,
00355     GL_EXT_texture_perturb_normal,
00356     GL_EXT_texture_sRGB,
00357     GL_EXT_texture_sRGB_R8,
00358     GL_EXT_texture_sRGB_RG8,
00359     GL_EXT_texture_sRGB_decode,
00360     GL_EXT_texture_shadow_lod,
00361     GL_EXT_texture_shared_exponent,
00362     GL_EXT_texture_snorm,
00363     GL_EXT_texture_storage,
00364     GL_EXT_texture_swizzle,
00365     GL_EXT_timer_query,
00366     GL_EXT_transform_feedback,
00367     GL_EXT_vertex_array,
00368     GL_EXT_vertex_array_bgra,
00369     GL_EXT_vertex_attrib_64bit,
00370     GL_EXT_vertex_shader,
00371     GL_EXT_vertex_weighting,
00372     GL_EXT_win32_keyed_mutex,
00373     GL_EXT_window_rectangles,
00374     GL_EXT_x11_sync_object,
00375     GL_GREMEDY_frame_terminator,
00376     GL_GREMEDY_string_marker,
00377     GL_HP_convolution_border_modes,
```

```
00378     GL_HP_image_transform,
00379     GL_HP_occlusion_test,
00380     GL_HP_texture_lighting,
00381     GL_IBM_cull_vertex,
00382     GL_IBM_multimode_draw_arrays,
00383     GL_IBM_rasterpos_clip,
00384     GL_IBM_static_data,
00385     GL_IBM_texture_mirrored_repeat,
00386     GL_IBM_vertex_array_lists,
00387     GL_INGR_blend_func_separate,
00388     GL_INGR_color_clamp,
00389     GL_INGR_interlace_read,
00390     GL_INTEL_blackhole_render,
00391     GL_INTEL_conservative_rasterization,
00392     GL_INTEL_fragment_shader_ordering,
00393     GL_INTEL_framebuffer_CMAA,
00394     GL_INTEL_map_texture,
00395     GL_INTEL_parallel_arrays,
00396     GL_INTEL_performance_query,
00397     GL_KHR_blend_equation_advanced,
00398     GL_KHR_blend_equation_advanced_coherent,
00399     GL_KHR_context_flush_control,
00400     GL_KHR_debug,
00401     GL_KHR_no_error,
00402     GL_KHR_parallel_shader_compile,
00403     GL_KHR_robust_buffer_access_behavior,
00404     GL_KHR_robustness,
00405     GL_KHR_shader_subgroup,
00406     GL_KHR_texture_compression_astc_hdr,
00407     GL_KHR_texture_compression_astc_ldr,
00408     GL_KHR_texture_compression_astc_sliced_3d,
00409     GL_MESA_texture_stack,
00410     GL_MESA_framebuffer_flip_x,
00411     GL_MESA_framebuffer_flip_y,
00412     GL_MESA_framebuffer_swap_xy,
00413     GL_MESA_pack_invert,
00414     GL_MESA_program_binary_formats,
00415     GL_MESA_resize_buffers,
00416     GL_MESA_shader_integer_functions,
00417     GL_MESA_tile_raster_order,
00418     GL_MESA_window_pos,
00419     GL_MESA_ycbcr_texture,
00420     GL_NVX_blend_equation_advanced_multi_draw_buffers,
00421     GL_NVX_conditional_render,
00422     GL_NVX_gpu_memory_info,
00423     GL_NVX_gpu_multicast2,
00424     GL_NVX_linked_gpu_multicast,
00425     GL_NVX_progress_fence,
00426     GL_NV_alpha_to_coverage_dither_control,
00427     GL_NV_bindless_multi_draw_indirect,
00428     GL_NV_bindless_multi_draw_indirect_count,
00429     GL_NV_bindless_texture,
00430     GL_NV_blend_equation_advanced,
00431     GL_NV_blend_equation_advanced_coherent,
00432     GL_NV_blend_minmax_factor,
00433     GL_NV_blend_square,
00434     GL_NV_clip_space_w_scaling,
00435     GL_NV_command_list,
00436     GL_NV_compute_program5,
00437     GL_NV_compute_shader_derivatives,
00438     GL_NV_conditional_render,
00439     GL_NV_conservative_raster,
00440     GL_NV_conservative_raster_dilate,
00441     GL_NV_conservative_raster_pre_snap,
00442     GL_NV_conservative_raster_pre_snap_triangles,
00443     GL_NV_conservative_raster_underestimation,
00444     GL_NV_copy_depth_to_color,
00445     GL_NV_copy_image,
00446     GL_NV_deep_texture3D,
00447     GL_NV_depth_buffer_float,
00448     GL_NV_depth_clamp,
00449     GL_NV_draw_texture,
00450     GL_NV_draw_vulkan_image,
00451     GL_NV_evaluators,
00452     GL_NV_explicit_multisample,
00453     GL_NV_fence,
00454     GL_NV_fill_rectangle,
00455     GL_NV_float_buffer,
00456     GL_NV_fog_distance,
00457     GL_NV_fragment_coverage_to_color,
00458     GL_NV_fragment_program,
00459     GL_NV_fragment_program2,
00460     GL_NV_fragment_program4,
00461     GL_NV_fragment_program_option,
00462     GL_NV_fragment_shader_barycentric,
00463     GL_NV_fragment_shader_interlock,
00464     GL_NV_framebuffer_mixed_samples,
```

```
00465     GL_NV_framebuffer_multisample_coverage,
00466     GL_NV_geometry_program4,
00467     GL_NV_geometry_shader4,
00468     GL_NV_geometry_shader_passthrough,
00469     GL_NV_gpu_multicast,
00470     GL_NV_gpu_program4,
00471     GL_NV_gpu_program5,
00472     GL_NV_gpu_program5_mem_extended,
00473     GL_NV_gpu_shader5,
00474     GL_NV_half_float,
00475     GL_NV_internalformat_sample_query,
00476     GL_NV_light_max_exponent,
00477     GL_NV_memory_attachment,
00478     GL_NV_memory_object_sparse,
00479     GL_NV_mesh_shader,
00480     GL_NV_multisample_coverage,
00481     GL_NV_multisample_filter_hint,
00482     GL_NV_occlusion_query,
00483     GL_NV_packed_depth_stencil,
00484     GL_NV_parameter_buffer_object,
00485     GL_NV_parameter_buffer_object2,
00486     GL_NV_path_rendering,
00487     GL_NV_path_rendering_shared_edge,
00488     GL_NV_pixel_data_range,
00489     GL_NV_point_sprite,
00490     GL_NV_present_video,
00491     GL_NV_primitive_restart,
00492     GL_NV_primitive_shading_rate,
00493     GL_NV_query_resource,
00494     GL_NV_query_resource_tag,
00495     GL_NV_register_combiners,
00496     GL_NV_register_combiners2,
00497     GL_NV_representative_fragment_test,
00498     GL_NV_robustness_video_memory_purge,
00499     GL_NV_sample_locations,
00500     GL_NV_sample_mask_override_coverage,
00501     GL_NV_scissor_exclusive,
00502     GL_NV_shader_atomic_counters,
00503     GL_NV_shader_atomic_float,
00504     GL_NV_shader_atomic_float64,
00505     GL_NV_shader_atomic_fp16_vector,
00506     GL_NV_shader_atomic_int64,
00507     GL_NV_shader_buffer_load,
00508     GL_NV_shader_buffer_store,
00509     GL_NV_shader_storage_buffer_object,
00510     GL_NV_shader_subgroup_partitioned,
00511     GL_NV_shader_texture_footprint,
00512     GL_NV_shader_thread_group,
00513     GL_NV_shader_thread_shuffle,
00514     GL_NV_shading_rate_image,
00515     GL_NV_stereo_view_rendering,
00516     GL_NV_tessellation_program5,
00517     GL_NV_texgen_emboss,
00518     GL_NV_texgen_reflection,
00519     GL_NV_texture_barrier,
00520     GL_NV_texture_compression_vtc,
00521     GL_NV_texture_env_combine4,
00522     GL_NV_texture_expand_normal,
00523     GL_NV_texture_multisample,
00524     GL_NV_texture_rectangle,
00525     GL_NV_texture_rectangle_compressed,
00526     GL_NV_texture_shader,
00527     GL_NV_texture_shader2,
00528     GL_NV_texture_shader3,
00529     GL_NV_timeline_semaphore,
00530     GL_NV_transform_feedback,
00531     GL_NV_transform_feedback2,
00532     GL_NV_uniform_buffer_std430_layout,
00533     GL_NV_uniform_buffer_unified_memory,
00534     GL_NV_vdpau_interop,
00535     GL_NV_vdpau_interop2,
00536     GL_NV_vertex_array_range,
00537     GL_NV_vertex_array_range2,
00538     GL_NV_vertex_attrib_integer_64bit,
00539     GL_NV_vertex_buffer_unified_memory,
00540     GL_NV_vertex_program,
00541     GL_NV_vertex_program1_1,
00542     GL_NV_vertex_program2,
00543     GL_NV_vertex_program2_option,
00544     GL_NV_vertex_program3,
00545     GL_NV_vertex_program4,
00546     GL_NV_video_capture,
00547     GL_NV_viewport_array2,
00548     GL_NV_viewport_swizzle,
00549     GL_OES_byte_coordinates,
00550     GL_OES_compressed_paletted_texture,
00551     GL_OES_fixed_point,
```

```

00552     GL_OES_query_matrix,
00553     GL_OES_read_format,
00554     GL_OES_single_precision,
00555     GL_OML_interlace,
00556     GL_OML_resample,
00557     GL_OML_subsample,
00558     GL_OVR_multiview,
00559     GL_OVR_multiview2,
00560     GL_PGI_misc_hints,
00561     GL_PGI_vertex_hints,
00562     GL_REND_screen_coordinates,
00563     GL_S3_s3tc,
00564     GL_SGIS_detail_texture,
00565     GL_SGIS_fog_function,
00566     GL_SGIS_generate_mipmap,
00567     GL_SGIS_multisample,
00568     GL_SGIS_pixel_texture,
00569     GL_SGIS_point_line_texgen,
00570     GL_SGIS_point_parameters,
00571     GL_SGIS_sharpen_texture,
00572     GL_SGIS_texture4D,
00573     GL_SGIS_texture_border_clamp,
00574     GL_SGIS_texture_color_mask,
00575     GL_SGIS_texture_edge_clamp,
00576     GL_SGIS_texture_filter4,
00577     GL_SGIS_texture_lod,
00578     GL_SGIS_texture_select,
00579     GL_SGIX_async,
00580     GL_SGIX_async_histogram,
00581     GL_SGIX_async_pixel,
00582     GL_SGIX_blend_alpha_minmax,
00583     GL_SGIX_calligraphic_fragment,
00584     GL_SGIX_clipmap,
00585     GL_SGIX_convolution_accuracy,
00586     GL_SGIX_depth_pass_instrument,
00587     GL_SGIX_depth_texture,
00588     GL_SGIX_flush_raster,
00589     GL_SGIX_fog_offset,
00590     GL_SGIX_fragment_lighting,
00591     GL_SGIX_framezoom,
00592     GL_SGIX_igloo_interface,
00593     GL_SGIX_instruments,
00594     GL_SGIX_interlace,
00595     GL_SGIX_ir_instrument1,
00596     GL_SGIX_list_priority,
00597     GL_SGIX_pixel_texture,
00598     GL_SGIX_pixel_tiles,
00599     GL_SGIX_polynomial_ffd,
00600     GL_SGIX_reference_plane,
00601     GL_SGIX_resample,
00602     GL_SGIX_scalebias_hint,
00603     GL_SGIX_shadow,
00604     GL_SGIX_shadow_ambient,
00605     GL_SGIX_sprite,
00606     GL_SGIX_subsample,
00607     GL_SGIX_tag_sample_buffer,
00608     GL_SGIX_texture_add_env,
00609     GL_SGIX_texture_coordinate_clamp,
00610     GL_SGIX_texture_lod_bias,
00611     GL_SGIX_texture_multi_buffer,
00612     GL_SGIX_texture_scale_bias,
00613     GL_SGIX_vertex_preclip,
00614     GL_SGIX_ycrcb,
00615     GL_SGIX_ycrcb_subsample,
00616     GL_SGIX_ycrcba,
00617     GL_SGI_color_matrix,
00618     GL_SGI_color_table,
00619     GL_SGI_texture_color_table,
00620     GL_SUNX_constant_data,
00621     GL_SUN_convolution_border_modes,
00622     GL_SUN_global_alpha,
00623     GL_SUN_mesh_array,
00624     GL_SUN_slice_accum,
00625     GL_SUN_triangle_list,
00626     GL_SUN_vertex,
00627     GL_WIN_phong_shading,
00628     GL_WIN_specular_fog
00629     Loader: True
00630     Local files: False
00631     Omit khrplatform: False
00632     Reproducible: False
00633
00634     Commandline:
00635         --profile="core" --api="gl=4.1" --generator="c" --spec="gl"
00636         --extensions="GL_3DFX_multisample, GL_3DFX_tbuffer, GL_3DFX_texture_compression_FXT1, GL_AMD_blend_minmax_factor, GL_AMD_com
00637         Online:
00638         Too many extensions

```

```

00638 */
00639
00640
00641 #ifndef __glad_h_
00642 #define __glad_h_
00643
00644 #ifdef __gl_h_
00645 #error OpenGL header already included, remove this include, glad already provides it
00646 #endif
00647 #define __gl_h_
00648
00649 #if defined(_WIN32) && !defined(APIENTRY) && !defined(__CYGWIN__) && !defined(__SCITECH_SNAP__)
00650 #define APIENTRY __stdcall
00651 #endif
00652
00653 #ifndef APIENTRY
00654 #define APIENTRY
00655 #endif
00656 #ifndef APIENTRY
00657 #define APIENTRY
00658 #endif
00659
00660 #ifndef GLAPIENTRY
00661 #define GLAPIENTRY APIENTRY
00662 #endif
00663
00664 #ifdef __cplusplus
00665 extern "C" {
00666 #endif
00667
00668 struct gladGLVersionStruct {
00669     int major;
00670     int minor;
00671 };
00672
00673 typedef void* (* GLADloadproc)(const char *name);
00674
00675 #ifndef GLAPI
00676 # if defined(GLAD_GLAPI_EXPORT)
00677 #   if defined(_WIN32) || defined(__CYGWIN__)
00678 #     if defined(GLAD_GLAPI_EXPORT_BUILD)
00679 #       if defined(__GNUC__)
00680 #         define GLAPI __attribute__ ((dllexport)) extern
00681 #       else
00682 #         define GLAPI __declspec(dllexport) extern
00683 #       endif
00684 #     else
00685 #       if defined(__GNUC__)
00686 #         define GLAPI __attribute__ ((dllimport)) extern
00687 #       else
00688 #         define GLAPI __declspec(dllimport) extern
00689 #       endif
00690 #     endif
00691 #   elif defined(__GNUC__) && defined(GLAD_GLAPI_EXPORT_BUILD)
00692 #     define GLAPI __attribute__ ((visibility ("default"))) extern
00693 #   else
00694 #     define GLAPI extern
00695 #   endif
00696 # else
00697 #   define GLAPI extern
00698 # endif
00699 #endif
00700
00701 GLAPI struct gladGLVersionStruct GLVersion;
00702
00703 GLAPI int gladLoadGL(void);
00704
00705 GLAPI int gladLoadGLLoader(GLADloadproc);
00706
00707 #include <KHR/khrplatform.h>
00708 typedef unsigned int GLenum;
00709 typedef unsigned char GLboolean;
00710 typedef unsigned int GLbitfield;
00711 typedef void GLvoid;
00712 typedef khronos_int8_t GLbyte;
00713 typedef khronos_uint8_t GLubyte;
00714 typedef khronos_int16_t GLshort;
00715 typedef khronos_uint16_t GLushort;
00716 typedef int GLint;
00717 typedef unsigned int GLuint;
00718 typedef khronos_int32_t GLclampx;
00719 typedef int GLsizei;
00720 typedef khronos_float_t GLfloat;
00721 typedef khronos_float_t GLclampf;
00722 typedef double GLdouble;
00723 typedef double GLclampd;
00724 typedef void *GLEglClientBufferEXT;

```

```
00725 typedef void *GLEglImageOES;
00726 typedef char GLchar;
00727 typedef char GLcharARB;
00728 #ifdef __APPLE__
00729 typedef void *GLhandleARB;
00730 #else
00731 typedef unsigned int GLhandleARB;
00732 #endif
00733 typedef khronos_uint16_t GLhalf;
00734 typedef khronos_uint16_t GLhalfARB;
00735 typedef khronos_int32_t GLfixed;
00736 typedef khronos_intptr_t GLintptr;
00737 typedef khronos_intptr_t GLintptrARB;
00738 typedef khronos_ssize_t GLsizeiptr;
00739 typedef khronos_ssize_t GLsizeiptrARB;
00740 typedef khronos_int64_t GLint64;
00741 typedef khronos_int64_t GLint64EXT;
00742 typedef khronos_uint64_t GLuint64;
00743 typedef khronos_uint64_t GLuint64EXT;
00744 typedef struct __GLsync *GLsync;
00745 struct _cl_context;
00746 struct _cl_event;
00747 typedef void (APIENTRY *GLDEBUGPROC) (GLenum source, GLenum type, GLuint id, GLenum severity, GLsizei
length, const GLchar *message, const void *userParam);
00748 typedef void (APIENTRY *GLDEBUGPROCARB) (GLenum source, GLenum type, GLuint id, GLenum severity, GLsizei
length, const GLchar *message, const void *userParam);
00749 typedef void (APIENTRY *GLDEBUGPROCKHR) (GLenum source, GLenum type, GLuint id, GLenum severity, GLsizei
length, const GLchar *message, const void *userParam);
00750 typedef void (APIENTRY *GLDEBUGPROCAMD) (GLuint id, GLenum category, GLenum severity, GLsizei length, const
GLchar *message, void *userParam);
00751 typedef unsigned short GLhalfNV;
00752 typedef GLintptr GLvdpauSurfaceNV;
00753 typedef void (APIENTRY *GLVULKANPROCNV) (void);
00754 #define GL_DEPTH_BUFFER_BIT 0x00000100
00755 #define GL_STENCIL_BUFFER_BIT 0x00000400
00756 #define GL_COLOR_BUFFER_BIT 0x00004000
00757 #define GL_FALSE 0
00758 #define GL_TRUE 1
00759 #define GL_POINTS 0x0000
00760 #define GL_LINES 0x0001
00761 #define GL_LINE_LOOP 0x0002
00762 #define GL_LINE_STRIP 0x0003
00763 #define GL_TRIANGLES 0x0004
00764 #define GL_TRIANGLE_STRIP 0x0005
00765 #define GL_TRIANGLE_FAN 0x0006
00766 #define GL_NEVER 0x0200
00767 #define GL_LESS 0x0201
00768 #define GL_EQUAL 0x0202
00769 #define GL_LEQUAL 0x0203
00770 #define GL_GREATER 0x0204
00771 #define GL_NOTEQUAL 0x0205
00772 #define GL_GEQUAL 0x0206
00773 #define GL_ALWAYS 0x0207
00774 #define GL_ZERO 0
00775 #define GL_ONE 1
00776 #define GL_SRC_COLOR 0x0300
00777 #define GL_ONE_MINUS_SRC_COLOR 0x0301
00778 #define GL_SRC_ALPHA 0x0302
00779 #define GL_ONE_MINUS_SRC_ALPHA 0x0303
00780 #define GL_DST_ALPHA 0x0304
00781 #define GL_ONE_MINUS_DST_ALPHA 0x0305
00782 #define GL_DST_COLOR 0x0306
00783 #define GL_ONE_MINUS_DST_COLOR 0x0307
00784 #define GL_SRC_ALPHA_SATURATE 0x0308
00785 #define GL_NONE 0
00786 #define GL_FRONT_LEFT 0x0400
00787 #define GL_FRONT_RIGHT 0x0401
00788 #define GL_BACK_LEFT 0x0402
00789 #define GL_BACK_RIGHT 0x0403
00790 #define GL_FRONT 0x0404
00791 #define GL_BACK 0x0405
00792 #define GL_LEFT 0x0406
00793 #define GL_RIGHT 0x0407
00794 #define GL_FRONT_AND_BACK 0x0408
00795 #define GL_NO_ERROR 0
00796 #define GL_INVALID_ENUM 0x0500
00797 #define GL_INVALID_VALUE 0x0501
00798 #define GL_INVALID_OPERATION 0x0502
00799 #define GL_OUT_OF_MEMORY 0x0505
00800 #define GL_CW 0x0900
00801 #define GL_CCW 0x0901
00802 #define GL_POINT_SIZE 0x0B11
00803 #define GL_POINT_SIZE_RANGE 0x0B12
00804 #define GL_POINT_SIZE_GRANULARITY 0x0B13
00805 #define GL_LINE_SMOOTH 0x0B20
00806 #define GL_LINE_WIDTH 0x0B21
00807 #define GL_LINE_WIDTH_RANGE 0x0B22
```

```
00808 #define GL_LINE_WIDTH_GRANULARITY 0x0B23
00809 #define GL_POLYGON_MODE 0x0B40
00810 #define GL_POLYGON_SMOOTH 0x0B41
00811 #define GL_CULL_FACE 0x0B44
00812 #define GL_CULL_FACE_MODE 0x0B45
00813 #define GL_FRONT_FACE 0x0B46
00814 #define GL_DEPTH_RANGE 0x0B70
00815 #define GL_DEPTH_TEST 0x0B71
00816 #define GL_DEPTH_WRITEMASK 0x0B72
00817 #define GL_DEPTH_CLEAR_VALUE 0x0B73
00818 #define GL_DEPTH_FUNC 0x0B74
00819 #define GL_STENCIL_TEST 0x0B90
00820 #define GL_STENCIL_CLEAR_VALUE 0x0B91
00821 #define GL_STENCIL_FUNC 0x0B92
00822 #define GL_STENCIL_VALUE_MASK 0x0B93
00823 #define GL_STENCIL_FAIL 0x0B94
00824 #define GL_STENCIL_PASS_DEPTH_FAIL 0x0B95
00825 #define GL_STENCIL_PASS_DEPTH_PASS 0x0B96
00826 #define GL_STENCIL_REF 0x0B97
00827 #define GL_STENCIL_WRITEMASK 0x0B98
00828 #define GL_VIEWPORT 0x0BA2
00829 #define GL_DITHER 0x0BD0
00830 #define GL_BLEND_DST 0x0BE0
00831 #define GL_BLEND_SRC 0x0BE1
00832 #define GL_BLEND 0x0BE2
00833 #define GL_LOGIC_OP_MODE 0x0BF0
00834 #define GL_DRAW_BUFFER 0x0C01
00835 #define GL_READ_BUFFER 0x0C02
00836 #define GL_SCISSOR_BOX 0x0C10
00837 #define GL_SCISSOR_TEST 0x0C11
00838 #define GL_COLOR_CLEAR_VALUE 0x0C22
00839 #define GL_COLOR_WRITEMASK 0x0C23
00840 #define GL_DOUBLEBUFFER 0x0C32
00841 #define GL_STEREO 0x0C33
00842 #define GL_LINE_SMOOTH_HINT 0x0C52
00843 #define GL_POLYGON_SMOOTH_HINT 0x0C53
00844 #define GL_UNPACK_SWAP_BYTES 0x0CF0
00845 #define GL_UNPACK_LSB_FIRST 0x0CF1
00846 #define GL_UNPACK_ROW_LENGTH 0x0CF2
00847 #define GL_UNPACK_SKIP_ROWS 0x0CF3
00848 #define GL_UNPACK_SKIP_PIXELS 0x0CF4
00849 #define GL_UNPACK_ALIGNMENT 0x0CF5
00850 #define GL_PACK_SWAP_BYTES 0x0D00
00851 #define GL_PACK_LSB_FIRST 0x0D01
00852 #define GL_PACK_ROW_LENGTH 0x0D02
00853 #define GL_PACK_SKIP_ROWS 0x0D03
00854 #define GL_PACK_SKIP_PIXELS 0x0D04
00855 #define GL_PACK_ALIGNMENT 0x0D05
00856 #define GL_MAX_TEXTURE_SIZE 0x0D33
00857 #define GL_MAX_VIEWPORT_DIMS 0x0D3A
00858 #define GL_SUBPIXEL_BITS 0x0D50
00859 #define GL_TEXTURE_1D 0x0DE0
00860 #define GL_TEXTURE_2D 0x0DE1
00861 #define GL_TEXTURE_WIDTH 0x1000
00862 #define GL_TEXTURE_HEIGHT 0x1001
00863 #define GL_TEXTURE_BORDER_COLOR 0x1004
00864 #define GL_DONT_CARE 0x1100
00865 #define GL_FASTEST 0x1101
00866 #define GL_NICEST 0x1102
00867 #define GL_BYTE 0x1400
00868 #define GL_UNSIGNED_BYTE 0x1401
00869 #define GL_SHORT 0x1402
00870 #define GL_UNSIGNED_SHORT 0x1403
00871 #define GL_INT 0x1404
00872 #define GL_UNSIGNED_INT 0x1405
00873 #define GL_FLOAT 0x1406
00874 #define GL_CLEAR 0x1500
00875 #define GL_AND 0x1501
00876 #define GL_AND_REVERSE 0x1502
00877 #define GL_COPY 0x1503
00878 #define GL_AND_INVERTED 0x1504
00879 #define GL_NOOP 0x1505
00880 #define GL_XOR 0x1506
00881 #define GL_OR 0x1507
00882 #define GL_NOR 0x1508
00883 #define GL_EQUIV 0x1509
00884 #define GL_INVERT 0x150A
00885 #define GL_OR_REVERSE 0x150B
00886 #define GL_COPY_INVERTED 0x150C
00887 #define GL_OR_INVERTED 0x150D
00888 #define GL_NAND 0x150E
00889 #define GL_SET 0x150F
00890 #define GL_TEXTURE 0x1702
00891 #define GL_COLOR 0x1800
00892 #define GL_DEPTH 0x1801
00893 #define GL_STENCIL 0x1802
00894 #define GL_STENCIL_INDEX 0x1901
```



```
00895 #define GL_DEPTH_COMPONENT 0x1902
00896 #define GL_RED 0x1903
00897 #define GL_GREEN 0x1904
00898 #define GL_BLUE 0x1905
00899 #define GL_ALPHA 0x1906
00900 #define GL_RGB 0x1907
00901 #define GL_RGBA 0x1908
00902 #define GL_POINT 0x1B00
00903 #define GL_LINE 0x1B01
00904 #define GL_FILL 0x1B02
00905 #define GL_KEEPP 0x1E00
00906 #define GL_REPLACE 0x1E01
00907 #define GL_INCR 0x1E02
00908 #define GL_DECR 0x1E03
00909 #define GL_VENDOR 0x1F00
00910 #define GL_RENDERER 0x1F01
00911 #define GL_VERSION 0x1F02
00912 #define GL_EXTENSIONS 0x1F03
00913 #define GL_NEAREST 0x2600
00914 #define GL_LINEAR 0x2601
00915 #define GL_NEAREST_MIPMAP_NEAREST 0x2700
00916 #define GL_LINEAR_MIPMAP_NEAREST 0x2701
00917 #define GL_NEAREST_MIPMAP_LINEAR 0x2702
00918 #define GL_LINEAR_MIPMAP_LINEAR 0x2703
00919 #define GL_TEXTURE_MAG_FILTER 0x2800
00920 #define GL_TEXTURE_MIN_FILTER 0x2801
00921 #define GL_TEXTURE_WRAP_S 0x2802
00922 #define GL_TEXTURE_WRAP_T 0x2803
00923 #define GL_REPEAT 0x2901
00924 #define GL_COLOR_LOGIC_OP 0x0BF2
00925 #define GL_POLYGON_OFFSET_UNITS 0x2A00
00926 #define GL_POLYGON_OFFSET_POINT 0x2A01
00927 #define GL_POLYGON_OFFSET_LINE 0x2A02
00928 #define GL_POLYGON_OFFSET_FILL 0x8037
00929 #define GL_POLYGON_OFFSET_FACTOR 0x8038
00930 #define GL_TEXTURE_BINDING_1D 0x8068
00931 #define GL_TEXTURE_BINDING_2D 0x8069
00932 #define GL_TEXTURE_INTERNAL_FORMAT 0x1003
00933 #define GL_TEXTURE_RED_SIZE 0x805C
00934 #define GL_TEXTURE_GREEN_SIZE 0x805D
00935 #define GL_TEXTURE_BLUE_SIZE 0x805E
00936 #define GL_TEXTURE_ALPHA_SIZE 0x805F
00937 #define GL_DOUBLE 0x140A
00938 #define GL_PROXY_TEXTURE_1D 0x8063
00939 #define GL_PROXY_TEXTURE_2D 0x8064
00940 #define GL_R3_G3_B2 0x2A10
00941 #define GL_RGB4 0x804F
00942 #define GL_RGB5 0x8050
00943 #define GL_RGB8 0x8051
00944 #define GL_RGB10 0x8052
00945 #define GL_RGB12 0x8053
00946 #define GL_RGB16 0x8054
00947 #define GL_RGBA2 0x8055
00948 #define GL_RGBA4 0x8056
00949 #define GL_RGB5_A1 0x8057
00950 #define GL_RGBA8 0x8058
00951 #define GL_RGB10_A2 0x8059
00952 #define GL_RGBA12 0x805A
00953 #define GL_RGBA16 0x805B
00954 #define GL_UNSIGNED_BYTE_3_3_2 0x8032
00955 #define GL_UNSIGNED_SHORT_4_4_4_4 0x8033
00956 #define GL_UNSIGNED_SHORT_5_5_5_1 0x8034
00957 #define GL_UNSIGNED_INT_8_8_8_8 0x8035
00958 #define GL_UNSIGNED_INT_10_10_10_2 0x8036
00959 #define GL_TEXTURE_BINDING_3D 0x806A
00960 #define GL_PACK_SKIP_IMAGES 0x806B
00961 #define GL_PACK_IMAGE_HEIGHT 0x806C
00962 #define GL_UNPACK_SKIP_IMAGES 0x806D
00963 #define GL_UNPACK_IMAGE_HEIGHT 0x806E
00964 #define GL_TEXTURE_3D 0x806F
00965 #define GL_PROXY_TEXTURE_3D 0x8070
00966 #define GL_TEXTURE_DEPTH 0x8071
00967 #define GL_TEXTURE_WRAP_R 0x8072
00968 #define GL_MAX_3D_TEXTURE_SIZE 0x8073
00969 #define GL_UNSIGNED_BYTE_2_3_3_REV 0x8362
00970 #define GL_UNSIGNED_SHORT_5_6_5 0x8363
00971 #define GL_UNSIGNED_SHORT_5_6_5_REV 0x8364
00972 #define GL_UNSIGNED_SHORT_4_4_4_4_REV 0x8365
00973 #define GL_UNSIGNED_SHORT_1_5_5_5_REV 0x8366
00974 #define GL_UNSIGNED_INT_8_8_8_8_REV 0x8367
00975 #define GL_UNSIGNED_INT_2_10_10_10_REV 0x8368
00976 #define GL_BGR 0x80E0
00977 #define GL_BGRA 0x80E1
00978 #define GL_MAX_ELEMENTS_VERTICES 0x80E8
00979 #define GL_MAX_ELEMENTS_INDICES 0x80E9
00980 #define GL_CLAMP_TO_EDGE 0x812F
00981 #define GL_TEXTURE_MIN_LOD 0x813A
```

```
00982 #define GL_TEXTURE_MAX_LOD 0x813B
00983 #define GL_TEXTURE_BASE_LEVEL 0x813C
00984 #define GL_TEXTURE_MAX_LEVEL 0x813D
00985 #define GL_SMOOTH_POINT_SIZE_RANGE 0x0B12
00986 #define GL_SMOOTH_POINT_SIZE_GRANULARITY 0x0B13
00987 #define GL_SMOOTH_LINE_WIDTH_RANGE 0x0B22
00988 #define GL_SMOOTH_LINE_WIDTH_GRANULARITY 0x0B23
00989 #define GL_ALIASED_LINE_WIDTH_RANGE 0x846E
00990 #define GL_TEXTURE0 0x84C0
00991 #define GL_TEXTURE1 0x84C1
00992 #define GL_TEXTURE2 0x84C2
00993 #define GL_TEXTURE3 0x84C3
00994 #define GL_TEXTURE4 0x84C4
00995 #define GL_TEXTURE5 0x84C5
00996 #define GL_TEXTURE6 0x84C6
00997 #define GL_TEXTURE7 0x84C7
00998 #define GL_TEXTURE8 0x84C8
00999 #define GL_TEXTURE9 0x84C9
01000 #define GL_TEXTURE10 0x84CA
01001 #define GL_TEXTURE11 0x84CB
01002 #define GL_TEXTURE12 0x84CC
01003 #define GL_TEXTURE13 0x84CD
01004 #define GL_TEXTURE14 0x84CE
01005 #define GL_TEXTURE15 0x84CF
01006 #define GL_TEXTURE16 0x84D0
01007 #define GL_TEXTURE17 0x84D1
01008 #define GL_TEXTURE18 0x84D2
01009 #define GL_TEXTURE19 0x84D3
01010 #define GL_TEXTURE20 0x84D4
01011 #define GL_TEXTURE21 0x84D5
01012 #define GL_TEXTURE22 0x84D6
01013 #define GL_TEXTURE23 0x84D7
01014 #define GL_TEXTURE24 0x84D8
01015 #define GL_TEXTURE25 0x84D9
01016 #define GL_TEXTURE26 0x84DA
01017 #define GL_TEXTURE27 0x84DB
01018 #define GL_TEXTURE28 0x84DC
01019 #define GL_TEXTURE29 0x84DD
01020 #define GL_TEXTURE30 0x84DE
01021 #define GL_TEXTURE31 0x84DF
01022 #define GL_ACTIVE_TEXTURE 0x84E0
01023 #define GL_MULTISAMPLE 0x809D
01024 #define GL_SAMPLE_ALPHA_TO_COVERAGE 0x809E
01025 #define GL_SAMPLE_ALPHA_TO_ONE 0x809F
01026 #define GL_SAMPLE_COVERAGE 0x80A0
01027 #define GL_SAMPLE_BUFFERS 0x80A8
01028 #define GL_SAMPLES 0x80A9
01029 #define GL_SAMPLE_COVERAGE_VALUE 0x80AA
01030 #define GL_SAMPLE_COVERAGE_INVERT 0x80AB
01031 #define GL_TEXTURE_CUBE_MAP 0x8513
01032 #define GL_TEXTURE_BINDING_CUBE_MAP 0x8514
01033 #define GL_TEXTURE_CUBE_MAP_POSITIVE_X 0x8515
01034 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_X 0x8516
01035 #define GL_TEXTURE_CUBE_MAP_POSITIVE_Y 0x8517
01036 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y 0x8518
01037 #define GL_TEXTURE_CUBE_MAP_POSITIVE_Z 0x8519
01038 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z 0x851A
01039 #define GL_PROXY_TEXTURE_CUBE_MAP 0x851B
01040 #define GL_MAX_CUBE_MAP_TEXTURE_SIZE 0x851C
01041 #define GL_COMPRESSED_RGB 0x84ED
01042 #define GL_COMPRESSED_RGBA 0x84EE
01043 #define GL_TEXTURE_COMPRESSION_HINT 0x84EF
01044 #define GL_TEXTURE_COMPRESSED_IMAGE_SIZE 0x86A0
01045 #define GL_TEXTURE_COMPRESSED 0x86A1
01046 #define GL_NUM_COMPRESSED_TEXTURE_FORMATS 0x86A2
01047 #define GL_COMPRESSED_TEXTURE_FORMATS 0x86A3
01048 #define GL_CLAMP_TO_BORDER 0x812D
01049 #define GL_BLEND_DST_RGB 0x80C8
01050 #define GL_BLEND_SRC_RGB 0x80C9
01051 #define GL_BLEND_DST_ALPHA 0x80CA
01052 #define GL_BLEND_SRC_ALPHA 0x80CB
01053 #define GL_POINT_FADE_THRESHOLD_SIZE 0x8128
01054 #define GL_DEPTH_COMPONENT16 0x81A5
01055 #define GL_DEPTH_COMPONENT24 0x81A6
01056 #define GL_DEPTH_COMPONENT32 0x81A7
01057 #define GL_MIRRORED_REPEAT 0x8370
01058 #define GL_MAX_TEXTURE_LOD_BIAS 0x84FD
01059 #define GL_TEXTURE_LOD_BIAS 0x8501
01060 #define GL_INCR_WRAP 0x8507
01061 #define GL DECR_WRAP 0x8508
01062 #define GL_TEXTURE_DEPTH_SIZE 0x884A
01063 #define GL_TEXTURE_COMPARE_MODE 0x884C
01064 #define GL_TEXTURE_COMPARE_FUNC 0x884D
01065 #define GL_BLEND_COLOR 0x8005
01066 #define GL_BLEND_EQUATION 0x8009
01067 #define GL_CONSTANT_COLOR 0x8001
01068 #define GL_ONE_MINUS_CONSTANT_COLOR 0x8002
```

```
01069 #define GL_CONSTANT_ALPHA 0x8003
01070 #define GL_ONE_MINUS_CONSTANT_ALPHA 0x8004
01071 #define GL_FUNC_ADD 0x8006
01072 #define GL_FUNC_REVERSE_SUBTRACT 0x800B
01073 #define GL_FUNC_SUBTRACT 0x800A
01074 #define GL_MIN 0x8007
01075 #define GL_MAX 0x8008
01076 #define GL_BUFFER_SIZE 0x8764
01077 #define GL_BUFFER_USAGE 0x8765
01078 #define GL_QUERY_COUNTER_BITS 0x8864
01079 #define GL_CURRENT_QUERY 0x8865
01080 #define GL_QUERY_RESULT 0x8866
01081 #define GL_QUERY_RESULT_AVAILABLE 0x8867
01082 #define GL_ARRAY_BUFFER 0x8892
01083 #define GL_ELEMENT_ARRAY_BUFFER 0x8893
01084 #define GL_ARRAY_BUFFER_BINDING 0x8894
01085 #define GL_ELEMENT_ARRAY_BUFFER_BINDING 0x8895
01086 #define GL_VERTEX_ATTRIB_ARRAY_BUFFER_BINDING 0x889F
01087 #define GL_READ_ONLY 0x88B8
01088 #define GL_WRITE_ONLY 0x88B9
01089 #define GL_READ_WRITE 0x88BA
01090 #define GL_BUFFER_ACCESS 0x88BB
01091 #define GL_BUFFER_MAPPED 0x88BC
01092 #define GL_BUFFER_MAP_POINTER 0x88BD
01093 #define GL_STREAM_DRAW 0x88E0
01094 #define GL_STREAM_READ 0x88E1
01095 #define GL_STREAM_COPY 0x88E2
01096 #define GL_STATIC_DRAW 0x88E4
01097 #define GL_STATIC_READ 0x88E5
01098 #define GL_STATIC_COPY 0x88E6
01099 #define GL_DYNAMIC_DRAW 0x88E8
01100 #define GL_DYNAMIC_READ 0x88E9
01101 #define GL_DYNAMIC_COPY 0x88EA
01102 #define GL_SAMPLES_PASSED 0x8914
01103 #define GL_SRC1_ALPHA 0x8589
01104 #define GL_BLEND_EQUATION_RGB 0x8009
01105 #define GL_VERTEX_ATTRIB_ARRAY_ENABLED 0x8622
01106 #define GL_VERTEX_ATTRIB_ARRAY_SIZE 0x8623
01107 #define GL_VERTEX_ATTRIB_ARRAY_STRIDE 0x8624
01108 #define GL_VERTEX_ATTRIB_ARRAY_TYPE 0x8625
01109 #define GL_CURRENT_VERTEX_ATTRIB 0x8626
01110 #define GL_VERTEX_PROGRAM_POINT_SIZE 0x8642
01111 #define GL_VERTEX_ATTRIB_ARRAY_POINTER 0x8645
01112 #define GL_STENCIL_BACK_FUNC 0x8800
01113 #define GL_STENCIL_BACK_FAIL 0x8801
01114 #define GL_STENCIL_BACK_PASS_DEPTH_FAIL 0x8802
01115 #define GL_STENCIL_BACK_PASS_DEPTH_PASS 0x8803
01116 #define GL_MAX_DRAW_BUFFERS 0x8824
01117 #define GL_DRAW_BUFFER0 0x8825
01118 #define GL_DRAW_BUFFER1 0x8826
01119 #define GL_DRAW_BUFFER2 0x8827
01120 #define GL_DRAW_BUFFER3 0x8828
01121 #define GL_DRAW_BUFFER4 0x8829
01122 #define GL_DRAW_BUFFER5 0x882A
01123 #define GL_DRAW_BUFFER6 0x882B
01124 #define GL_DRAW_BUFFER7 0x882C
01125 #define GL_DRAW_BUFFER8 0x882D
01126 #define GL_DRAW_BUFFER9 0x882E
01127 #define GL_DRAW_BUFFER10 0x882F
01128 #define GL_DRAW_BUFFER11 0x8830
01129 #define GL_DRAW_BUFFER12 0x8831
01130 #define GL_DRAW_BUFFER13 0x8832
01131 #define GL_DRAW_BUFFER14 0x8833
01132 #define GL_DRAW_BUFFER15 0x8834
01133 #define GL_BLEND_EQUATION_ALPHA 0x883D
01134 #define GL_MAX_VERTEX_ATTRIBS 0x8869
01135 #define GL_VERTEX_ATTRIB_ARRAY_NORMALIZED 0x886A
01136 #define GL_MAX_TEXTURE_IMAGE_UNITS 0x8872
01137 #define GL_FRAGMENT_SHADER 0x8B30
01138 #define GL_VERTEX_SHADER 0x8B31
01139 #define GL_MAX_FRAGMENT_UNIFORM_COMPONENTS 0x8B49
01140 #define GL_MAX_VERTEX_UNIFORM_COMPONENTS 0x8B4A
01141 #define GL_MAX_VARYING_FLOATS 0x8B4B
01142 #define GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS 0x8B4C
01143 #define GL_MAX_COMBINED_TEXTURE_IMAGE_UNITS 0x8B4D
01144 #define GL_SHADER_TYPE 0x8B4F
01145 #define GL_FLOAT_VEC2 0x8B50
01146 #define GL_FLOAT_VEC3 0x8B51
01147 #define GL_FLOAT_VEC4 0x8B52
01148 #define GL_INT_VEC2 0x8B53
01149 #define GL_INT_VEC3 0x8B54
01150 #define GL_INT_VEC4 0x8B55
01151 #define GL_BOOL 0x8B56
01152 #define GL_BOOL_VEC2 0x8B57
01153 #define GL_BOOL_VEC3 0x8B58
01154 #define GL_BOOL_VEC4 0x8B59
01155 #define GL_FLOAT_MAT2 0x8B5A
```

```
01156 #define GL_FLOAT_MAT3 0x8B5B
01157 #define GL_FLOAT_MAT4 0x8B5C
01158 #define GL_SAMPLER_1D 0x8B5D
01159 #define GL_SAMPLER_2D 0x8B5E
01160 #define GL_SAMPLER_3D 0x8B5F
01161 #define GL_SAMPLER_CUBE 0x8B60
01162 #define GL_SAMPLER_1D_SHADOW 0x8B61
01163 #define GL_SAMPLER_2D_SHADOW 0x8B62
01164 #define GL_DELETE_STATUS 0x8B80
01165 #define GL_COMPILE_STATUS 0x8B81
01166 #define GL_LINK_STATUS 0x8B82
01167 #define GL_VALIDATE_STATUS 0x8B83
01168 #define GL_INFO_LOG_LENGTH 0x8B84
01169 #define GL_ATTACHED_SHADERS 0x8B85
01170 #define GL_ACTIVE_UNIFORMS 0x8B86
01171 #define GL_ACTIVE_UNIFORM_MAX_LENGTH 0x8B87
01172 #define GL_SHADER_SOURCE_LENGTH 0x8B88
01173 #define GL_ACTIVE_ATTRIBUTES 0x8B89
01174 #define GL_ACTIVE_ATTRIBUTE_MAX_LENGTH 0x8B8A
01175 #define GL_FRAGMENT_SHADER_DERIVATIVE_HINT 0x8B8B
01176 #define GL_SHADING_LANGUAGE_VERSION 0x8B8C
01177 #define GL_CURRENT_PROGRAM 0x8B8D
01178 #define GL_POINT_SPRITE_COORD_ORIGIN 0x8CA0
01179 #define GL_LOWER_LEFT 0x8CA1
01180 #define GL_UPPER_LEFT 0x8CA2
01181 #define GL_STENCIL_BACK_REF 0x8CA3
01182 #define GL_STENCIL_BACK_VALUE_MASK 0x8CA4
01183 #define GL_STENCIL_BACK_WRITEMASK 0x8CA5
01184 #define GL_PIXEL_PACK_BUFFER 0x88EB
01185 #define GL_PIXEL_UNPACK_BUFFER 0x88EC
01186 #define GL_PIXEL_PACK_BUFFER_BINDING 0x88ED
01187 #define GL_PIXEL_UNPACK_BUFFER_BINDING 0x88EF
01188 #define GL_FLOAT_MAT2x3 0x8B65
01189 #define GL_FLOAT_MAT2x4 0x8B66
01190 #define GL_FLOAT_MAT3x2 0x8B67
01191 #define GL_FLOAT_MAT3x4 0x8B68
01192 #define GL_FLOAT_MAT4x2 0x8B69
01193 #define GL_FLOAT_MAT4x3 0x8B6A
01194 #define GL_SRGB 0x8C40
01195 #define GL_SRGB8 0x8C41
01196 #define GL_SRGB_ALPHA 0x8C42
01197 #define GL_SRGB8_ALPHA8 0x8C43
01198 #define GL_COMPRESSED_SRGB 0x8C48
01199 #define GL_COMPRESSED_SRGB_ALPHA 0x8C49
01200 #define GL_COMPARE_REF_TO_TEXTURE 0x884E
01201 #define GL_CLIP_DISTANCE0 0x3000
01202 #define GL_CLIP_DISTANCE1 0x3001
01203 #define GL_CLIP_DISTANCE2 0x3002
01204 #define GL_CLIP_DISTANCE3 0x3003
01205 #define GL_CLIP_DISTANCE4 0x3004
01206 #define GL_CLIP_DISTANCE5 0x3005
01207 #define GL_CLIP_DISTANCE6 0x3006
01208 #define GL_CLIP_DISTANCE7 0x3007
01209 #define GL_MAX_CLIP_DISTANCES 0x0D32
01210 #define GL_MAJOR_VERSION 0x821B
01211 #define GL_MINOR_VERSION 0x821C
01212 #define GL_NUM_EXTENSIONS 0x821D
01213 #define GL_CONTEXT_FLAGS 0x821E
01214 #define GL_COMPRESSED_RED 0x8225
01215 #define GL_COMPRESSED_RG 0x8226
01216 #define GL_CONTEXT_FLAG_FORWARD_COMPATIBLE_BIT 0x00000001
01217 #define GL_RGBA32F 0x8814
01218 #define GL_RGB32F 0x8815
01219 #define GL_RGBA16F 0x881A
01220 #define GL_RGB16F 0x881B
01221 #define GL_VERTEX_ATTRIB_ARRAY_INTEGER 0x88FD
01222 #define GL_MAX_ARRAY_TEXTURE_LAYERS 0x88FF
01223 #define GL_MIN_PROGRAM_TEXEL_OFFSET 0x8904
01224 #define GL_MAX_PROGRAM_TEXEL_OFFSET 0x8905
01225 #define GL_CLAMP_READ_COLOR 0x891C
01226 #define GL_FIXED_ONLY 0x891D
01227 #define GL_MAX_VARYING_COMPONENTS 0x8B4B
01228 #define GL_TEXTURE_1D_ARRAY 0x8C18
01229 #define GL_PROXY_TEXTURE_1D_ARRAY 0x8C19
01230 #define GL_TEXTURE_2D_ARRAY 0x8C1A
01231 #define GL_PROXY_TEXTURE_2D_ARRAY 0x8C1B
01232 #define GL_TEXTURE_BINDING_1D_ARRAY 0x8C1C
01233 #define GL_TEXTURE_BINDING_2D_ARRAY 0x8C1D
01234 #define GL_R11F_G11F_B10F 0x8C3A
01235 #define GL_UNSIGNED_INT_10F_11F_11F_REV 0x8C3B
01236 #define GL_RGB9_E5 0x8C3D
01237 #define GL_UNSIGNED_INT_5_9_9_9_REV 0x8C3E
01238 #define GL_TEXTURE_SHARED_SIZE 0x8C3F
01239 #define GL_TRANSFORM_FEEDBACK_VARYING_MAX_LENGTH 0x8C76
01240 #define GL_TRANSFORM_FEEDBACK_BUFFER_MODE 0x8C7F
01241 #define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_COMPONENTS 0x8C80
01242 #define GL_TRANSFORM_FEEDBACK_VARYINGS 0x8C83
```

```
01243 #define GL_TRANSFORM_FEEDBACK_BUFFER_START 0x8C84
01244 #define GL_TRANSFORM_FEEDBACK_BUFFER_SIZE 0x8C85
01245 #define GL_PRIMITIVES_GENERATED 0x8C87
01246 #define GL_TRANSFORM_FEEDBACK_PRIMITIVES_WRITTEN 0x8C88
01247 #define GL_RASTERIZER_DISCARD 0x8C89
01248 #define GL_MAX_TRANSFORM_FEEDBACK_INTERLEAVED_COMPONENTS 0x8C8A
01249 #define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_ATTRIBS 0x8C8B
01250 #define GL_INTERLEAVED_ATTRIBS 0x8C8C
01251 #define GL_SEPARATE_ATTRIBS 0x8C8D
01252 #define GL_TRANSFORM_FEEDBACK_BUFFER 0x8C8E
01253 #define GL_TRANSFORM_FEEDBACK_BUFFER_BINDING 0x8C8F
01254 #define GL_RGBA32UI 0x8D70
01255 #define GL_RGB32UI 0x8D71
01256 #define GL_RGBA16UI 0x8D76
01257 #define GL_RGB16UI 0x8D77
01258 #define GL_RGBA8UI 0x8D7C
01259 #define GL_RGB8UI 0x8D7D
01260 #define GL_RGBA32I 0x8D82
01261 #define GL_RGB32I 0x8D83
01262 #define GL_RGBA16I 0x8D88
01263 #define GL_RGB16I 0x8D89
01264 #define GL_RGBA8I 0x8D8E
01265 #define GL_RGB8I 0x8D8F
01266 #define GL_RED_INTEGER 0x8D94
01267 #define GL_GREEN_INTEGER 0x8D95
01268 #define GL_BLUE_INTEGER 0x8D96
01269 #define GL_RGB_INTEGER 0x8D98
01270 #define GL_RGBA_INTEGER 0x8D99
01271 #define GL_BGR_INTEGER 0x8D9A
01272 #define GL_BGRA_INTEGER 0x8D9B
01273 #define GL_SAMPLER_1D_ARRAY 0x8DC0
01274 #define GL_SAMPLER_2D_ARRAY 0x8DC1
01275 #define GL_SAMPLER_1D_ARRAY_SHADOW 0x8DC3
01276 #define GL_SAMPLER_2D_ARRAY_SHADOW 0x8DC4
01277 #define GL_SAMPLER_CUBE_SHADOW 0x8DC5
01278 #define GL_UNSIGNED_INT_VEC2 0x8DC6
01279 #define GL_UNSIGNED_INT_VEC3 0x8DC7
01280 #define GL_UNSIGNED_INT_VEC4 0x8DC8
01281 #define GL_INT_SAMPLER_1D 0x8DC9
01282 #define GL_INT_SAMPLER_2D 0x8DCA
01283 #define GL_INT_SAMPLER_3D 0x8DCB
01284 #define GL_INT_SAMPLER_CUBE 0x8DCC
01285 #define GL_INT_SAMPLER_1D_ARRAY 0x8DCE
01286 #define GL_INT_SAMPLER_2D_ARRAY 0x8DCF
01287 #define GL_UNSIGNED_INT_SAMPLER_1D 0x8DD1
01288 #define GL_UNSIGNED_INT_SAMPLER_2D 0x8DD2
01289 #define GL_UNSIGNED_INT_SAMPLER_3D 0x8DD3
01290 #define GL_UNSIGNED_INT_SAMPLER_CUBE 0x8DD4
01291 #define GL_UNSIGNED_INT_SAMPLER_1D_ARRAY 0x8DD6
01292 #define GL_UNSIGNED_INT_SAMPLER_2D_ARRAY 0x8DD7
01293 #define GL_QUERY_WAIT 0x8E13
01294 #define GL_QUERY_NO_WAIT 0x8E14
01295 #define GL_QUERY_BY_REGION_WAIT 0x8E15
01296 #define GL_QUERY_BY_REGION_NO_WAIT 0x8E16
01297 #define GL_BUFFER_ACCESS_FLAGS 0x911F
01298 #define GL_BUFFER_MAP_LENGTH 0x9120
01299 #define GL_BUFFER_MAP_OFFSET 0x9121
01300 #define GL_DEPTH_COMPONENT32F 0x8CAC
01301 #define GL_DEPTH32F_STENCIL8 0x8CAD
01302 #define GL_FLOAT_32_UNSIGNED_INT_24_8_REV 0x8DAD
01303 #define GL_INVALID_FRAMEBUFFER_OPERATION 0x0506
01304 #define GL_FRAMEBUFFER_ATTACHMENT_COLOR_ENCODING 0x8210
01305 #define GL_FRAMEBUFFER_ATTACHMENT_COMPONENT_TYPE 0x8211
01306 #define GL_FRAMEBUFFER_ATTACHMENT_RED_SIZE 0x8212
01307 #define GL_FRAMEBUFFER_ATTACHMENT_GREEN_SIZE 0x8213
01308 #define GL_FRAMEBUFFER_ATTACHMENT_BLUE_SIZE 0x8214
01309 #define GL_FRAMEBUFFER_ATTACHMENT_ALPHA_SIZE 0x8215
01310 #define GL_FRAMEBUFFER_ATTACHMENT_DEPTH_SIZE 0x8216
01311 #define GL_FRAMEBUFFER_ATTACHMENT_STENCIL_SIZE 0x8217
01312 #define GL_FRAMEBUFFER_DEFAULT 0x8218
01313 #define GL_FRAMEBUFFER_UNDEFINED 0x8219
01314 #define GL_DEPTH_STENCIL_ATTACHMENT 0x821A
01315 #define GL_MAX_RENDERBUFFER_SIZE 0x84E8
01316 #define GL_DEPTH_STENCIL 0x84F9
01317 #define GL_UNSIGNED_INT_24_8 0x84FA
01318 #define GL_DEPTH24_STENCIL8 0x88F0
01319 #define GL_TEXTURE_STENCIL_SIZE 0x88F1
01320 #define GL_TEXTURE_RED_TYPE 0x8C10
01321 #define GL_TEXTURE_GREEN_TYPE 0x8C11
01322 #define GL_TEXTURE_BLUE_TYPE 0x8C12
01323 #define GL_TEXTURE_ALPHA_TYPE 0x8C13
01324 #define GL_TEXTURE_DEPTH_TYPE 0x8C16
01325 #define GL_UNSIGNED_NORMALIZED 0x8C17
01326 #define GL_FRAMEBUFFER_BINDING 0x8CA6
01327 #define GL_DRAW_FRAMEBUFFER_BINDING 0x8CA7
01328 #define GL_RENDERBUFFER_BINDING 0x8CA7
01329 #define GL_READ_FRAMEBUFFER 0x8CA8
```

```
01330 #define GL_DRAW_FRAMEBUFFER 0x8CA9
01331 #define GL_READ_FRAMEBUFFER_BINDING 0x8CAA
01332 #define GL_RENDERBUFFER_SAMPLES 0x8CAB
01333 #define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE 0x8CD0
01334 #define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_NAME 0x8CD1
01335 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL 0x8CD2
01336 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE 0x8CD3
01337 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LAYER 0x8CD4
01338 #define GL_FRAMEBUFFER_COMPLETE 0x8CD5
01339 #define GL_FRAMEBUFFER_INCOMPLETE_ATTACHMENT 0x8CD6
01340 #define GL_FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT 0x8CD7
01341 #define GL_FRAMEBUFFER_INCOMPLETE_DRAW_BUFFER 0x8CDB
01342 #define GL_FRAMEBUFFER_INCOMPLETE_READ_BUFFER 0x8CDC
01343 #define GL_FRAMEBUFFER_UNSUPPORTED 0x8CDD
01344 #define GL_MAX_COLOR_ATTACHMENTS 0x8CDF
01345 #define GL_COLOR_ATTACHMENT0 0x8CE0
01346 #define GL_COLOR_ATTACHMENT1 0x8CE1
01347 #define GL_COLOR_ATTACHMENT2 0x8CE2
01348 #define GL_COLOR_ATTACHMENT3 0x8CE3
01349 #define GL_COLOR_ATTACHMENT4 0x8CE4
01350 #define GL_COLOR_ATTACHMENT5 0x8CE5
01351 #define GL_COLOR_ATTACHMENT6 0x8CE6
01352 #define GL_COLOR_ATTACHMENT7 0x8CE7
01353 #define GL_COLOR_ATTACHMENT8 0x8CE8
01354 #define GL_COLOR_ATTACHMENT9 0x8CE9
01355 #define GL_COLOR_ATTACHMENT10 0x8CEA
01356 #define GL_COLOR_ATTACHMENT11 0x8CEB
01357 #define GL_COLOR_ATTACHMENT12 0x8CEC
01358 #define GL_COLOR_ATTACHMENT13 0x8CED
01359 #define GL_COLOR_ATTACHMENT14 0x8CEE
01360 #define GL_COLOR_ATTACHMENT15 0x8CEF
01361 #define GL_COLOR_ATTACHMENT16 0x8CF0
01362 #define GL_COLOR_ATTACHMENT17 0x8CF1
01363 #define GL_COLOR_ATTACHMENT18 0x8CF2
01364 #define GL_COLOR_ATTACHMENT19 0x8CF3
01365 #define GL_COLOR_ATTACHMENT20 0x8CF4
01366 #define GL_COLOR_ATTACHMENT21 0x8CF5
01367 #define GL_COLOR_ATTACHMENT22 0x8CF6
01368 #define GL_COLOR_ATTACHMENT23 0x8CF7
01369 #define GL_COLOR_ATTACHMENT24 0x8CF8
01370 #define GL_COLOR_ATTACHMENT25 0x8CF9
01371 #define GL_COLOR_ATTACHMENT26 0x8CFA
01372 #define GL_COLOR_ATTACHMENT27 0x8CFB
01373 #define GL_COLOR_ATTACHMENT28 0x8CFC
01374 #define GL_COLOR_ATTACHMENT29 0x8CFD
01375 #define GL_COLOR_ATTACHMENT30 0x8CFE
01376 #define GL_COLOR_ATTACHMENT31 0x8CFF
01377 #define GL_DEPTH_ATTACHMENT 0x8D00
01378 #define GL_STENCIL_ATTACHMENT 0x8D20
01379 #define GL_FRAMEBUFFER 0x8D40
01380 #define GL_RENDERBUFFER 0x8D41
01381 #define GL_RENDERBUFFER_WIDTH 0x8D42
01382 #define GL_RENDERBUFFER_HEIGHT 0x8D43
01383 #define GL_RENDERBUFFER_INTERNAL_FORMAT 0x8D44
01384 #define GL_STENCIL_INDEX1 0x8D46
01385 #define GL_STENCIL_INDEX4 0x8D47
01386 #define GL_STENCIL_INDEX8 0x8D48
01387 #define GL_STENCIL_INDEX16 0x8D49
01388 #define GL_RENDERBUFFER_RED_SIZE 0x8D50
01389 #define GL_RENDERBUFFER_GREEN_SIZE 0x8D51
01390 #define GL_RENDERBUFFER_BLUE_SIZE 0x8D52
01391 #define GL_RENDERBUFFER_ALPHA_SIZE 0x8D53
01392 #define GL_RENDERBUFFER_DEPTH_SIZE 0x8D54
01393 #define GL_RENDERBUFFER_STENCIL_SIZE 0x8D55
01394 #define GL_FRAMEBUFFER_INCOMPLETE_MULTISAMPLE 0x8D56
01395 #define GL_MAX_SAMPLES 0x8D57
01396 #define GL_FRAMEBUFFER_SRGB 0x8DB9
01397 #define GL_HALF_FLOAT 0x140B
01398 #define GL_MAP_READ_BIT 0x0001
01399 #define GL_MAP_WRITE_BIT 0x0002
01400 #define GL_MAP_INVALIDATE_RANGE_BIT 0x0004
01401 #define GL_MAP_INVALIDATE_BUFFER_BIT 0x0008
01402 #define GL_MAP_FLUSH_EXPLICIT_BIT 0x0010
01403 #define GL_MAP_UNSYNCHRONIZED_BIT 0x0020
01404 #define GL_COMPRESSED_RED_RGTC1 0x8DBB
01405 #define GL_COMPRESSED_SIGNED_RED_RGTC1 0x8DBC
01406 #define GL_COMPRESSED_RG_RGTC2 0x8DBD
01407 #define GL_COMPRESSED_SIGNED_RG_RGTC2 0x8DBE
01408 #define GL_RG 0x8227
01409 #define GL_RG_INTEGER 0x8228
01410 #define GL_R8 0x8229
01411 #define GL_R16 0x822A
01412 #define GL_RG8 0x822B
01413 #define GL_RG16 0x822C
01414 #define GL_R16F 0x822D
01415 #define GL_R32F 0x822E
01416 #define GL_RG16F 0x822F
```



```
01417 #define GL_RG32F 0x8230
01418 #define GL_R8I 0x8231
01419 #define GL_R8UI 0x8232
01420 #define GL_R16I 0x8233
01421 #define GL_R16UI 0x8234
01422 #define GL_R32I 0x8235
01423 #define GL_R32UI 0x8236
01424 #define GL_RG8I 0x8237
01425 #define GL_RG8UI 0x8238
01426 #define GL_RG16I 0x8239
01427 #define GL_RG16UI 0x823A
01428 #define GL_RG32I 0x823B
01429 #define GL_RG32UI 0x823C
01430 #define GL_VERTEX_ARRAY_BINDING 0x85B5
01431 #define GL_SAMPLER_2D_RECT 0x8B63
01432 #define GL_SAMPLER_2D_RECT_SHADOW 0x8B64
01433 #define GL_SAMPLER_BUFFER 0x8DC2
01434 #define GL_INT_SAMPLER_2D_RECT 0x8DCD
01435 #define GL_INT_SAMPLER_BUFFER 0x8DD0
01436 #define GL_UNSIGNED_INT_SAMPLER_2D_RECT 0x8DD5
01437 #define GL_UNSIGNED_INT_SAMPLER_BUFFER 0x8DD8
01438 #define GL_TEXTURE_BUFFER 0x8C2A
01439 #define GL_MAX_TEXTURE_BUFFER_SIZE 0x8C2B
01440 #define GL_TEXTURE_BINDING_BUFFER 0x8C2C
01441 #define GL_TEXTURE_BUFFER_DATA_STORE_BINDING 0x8C2D
01442 #define GL_TEXTURE_RECTANGLE 0x84F5
01443 #define GL_TEXTURE_BINDING_RECTANGLE 0x84F6
01444 #define GL_PROXY_TEXTURE_RECTANGLE 0x84F7
01445 #define GL_MAX_RECTANGLE_TEXTURE_SIZE 0x84F8
01446 #define GL_R8_SNORM 0x8F94
01447 #define GL_RG8_SNORM 0x8F95
01448 #define GL_RGB8_SNORM 0x8F96
01449 #define GL_RGBA8_SNORM 0x8F97
01450 #define GL_R16_SNORM 0x8F98
01451 #define GL_RG16_SNORM 0x8F99
01452 #define GL_RGB16_SNORM 0x8F9A
01453 #define GL_RGBA16_SNORM 0x8F9B
01454 #define GL_SIGNED_NORMALIZED 0x8F9C
01455 #define GL_PRIMITIVE_RESTART 0x8F9D
01456 #define GL_PRIMITIVE_RESTART_INDEX 0x8F9E
01457 #define GL_COPY_READ_BUFFER 0x8F36
01458 #define GL_COPY_WRITE_BUFFER 0x8F37
01459 #define GL_UNIFORM_BUFFER 0x8A11
01460 #define GL_UNIFORM_BUFFER_BINDING 0x8A28
01461 #define GL_UNIFORM_BUFFER_START 0x8A29
01462 #define GL_UNIFORM_BUFFER_SIZE 0x8A2A
01463 #define GL_MAX_VERTEX_UNIFORM_BLOCKS 0x8A2B
01464 #define GL_MAX_GEOMETRY_UNIFORM_BLOCKS 0x8A2C
01465 #define GL_MAX_FRAGMENT_UNIFORM_BLOCKS 0x8A2D
01466 #define GL_MAX_COMBINED_UNIFORM_BLOCKS 0x8A2E
01467 #define GL_MAX_UNIFORM_BUFFER_BINDINGS 0x8A2F
01468 #define GL_MAX_UNIFORM_BLOCK_SIZE 0x8A30
01469 #define GL_MAX_COMBINED_VERTEX_UNIFORM_COMPONENTS 0x8A31
01470 #define GL_MAX_COMBINED_GEOMETRY_UNIFORM_COMPONENTS 0x8A32
01471 #define GL_MAX_COMBINED_FRAGMENT_UNIFORM_COMPONENTS 0x8A33
01472 #define GL_UNIFORM_BUFFER_OFFSET_ALIGNMENT 0x8A34
01473 #define GL_ACTIVE_UNIFORM_BLOCK_MAX_NAME_LENGTH 0x8A35
01474 #define GL_ACTIVE_UNIFORM_BLOCKS 0x8A36
01475 #define GL_UNIFORM_TYPE 0x8A37
01476 #define GL_UNIFORM_SIZE 0x8A38
01477 #define GL_UNIFORM_NAME_LENGTH 0x8A39
01478 #define GL_UNIFORM_BLOCK_INDEX 0x8A3A
01479 #define GL_UNIFORM_OFFSET 0x8A3B
01480 #define GL_UNIFORM_ARRAY_STRIDE 0x8A3C
01481 #define GL_UNIFORM_MATRIX_STRIDE 0x8A3D
01482 #define GL_UNIFORM_IS_ROW_MAJOR 0x8A3E
01483 #define GL_UNIFORM_BLOCK_BINDING 0x8A3F
01484 #define GL_UNIFORM_BLOCK_DATA_SIZE 0x8A40
01485 #define GL_UNIFORM_BLOCK_NAME_LENGTH 0x8A41
01486 #define GL_UNIFORM_BLOCK_ACTIVE_UNIFORMS 0x8A42
01487 #define GL_UNIFORM_BLOCK_ACTIVE_UNIFORM_INDICES 0x8A43
01488 #define GL_UNIFORM_BLOCK_REFERENCED_BY_VERTEX_SHADER 0x8A44
01489 #define GL_UNIFORM_BLOCK_REFERENCED_BY_GEOMETRY_SHADER 0x8A45
01490 #define GL_UNIFORM_BLOCK_REFERENCED_BY_FRAGMENT_SHADER 0x8A46
01491 #define GL_INVALID_INDEX 0xFFFFFFFF
01492 #define GL_CONTEXT_CORE_PROFILE_BIT 0x00000001
01493 #define GL_CONTEXT_COMPATIBILITY_PROFILE_BIT 0x00000002
01494 #define GL_LINES_ADJACENCY 0x000A
01495 #define GL_LINE_STRIP_ADJACENCY 0x000B
01496 #define GL_TRIANGLES_ADJACENCY 0x000C
01497 #define GL_TRIANGLE_STRIP_ADJACENCY 0x000D
01498 #define GL_PROGRAM_POINT_SIZE 0x8642
01499 #define GL_MAX_GEOMETRY_TEXTURE_IMAGE_UNITS 0x8C29
01500 #define GL_FRAMEBUFFER_ATTACHMENT_LAYERED 0x8DA7
01501 #define GL_FRAMEBUFFER_INCOMPLETE_LAYER_TARGETS 0x8DA8
01502 #define GL_GEOMETRY_SHADER 0x8DD9
01503 #define GL_GEOMETRY_VERTICES_OUT 0x8916
```

```
01504 #define GL_GEOMETRY_INPUT_TYPE 0x8917
01505 #define GL_GEOMETRY_OUTPUT_TYPE 0x8918
01506 #define GL_MAX_GEOMETRY_UNIFORM_COMPONENTS 0x8DDF
01507 #define GL_MAX_GEOMETRY_OUTPUT_VERTICES 0x8DE0
01508 #define GL_MAX_GEOMETRY_TOTAL_OUTPUT_COMPONENTS 0x8DE1
01509 #define GL_MAX_VERTEX_OUTPUT_COMPONENTS 0x9122
01510 #define GL_MAX_GEOMETRY_INPUT_COMPONENTS 0x9123
01511 #define GL_MAX_GEOMETRY_OUTPUT_COMPONENTS 0x9124
01512 #define GL_MAX_FRAGMENT_INPUT_COMPONENTS 0x9125
01513 #define GL_CONTEXT_PROFILE_MASK 0x9126
01514 #define GL_DEPTH_CLAMP 0x864F
01515 #define GL_QUADS_FOLLOW_PROVOKING_VERTEX_CONVENTION 0x8E4C
01516 #define GL_FIRST_VERTEX_CONVENTION 0x8E4D
01517 #define GL_LAST_VERTEX_CONVENTION 0x8E4E
01518 #define GL_PROVOKING_VERTEX 0x8E4F
01519 #define GL_TEXTURE_CUBE_MAP_SEAMLESS 0x884F
01520 #define GL_MAX_SERVER_WAIT_TIMEOUT 0x9111
01521 #define GL_OBJECT_TYPE 0x9112
01522 #define GL_SYNC_CONDITION 0x9113
01523 #define GL_SYNC_STATUS 0x9114
01524 #define GL_SYNC_FLAGS 0x9115
01525 #define GL_SYNC_FENCE 0x9116
01526 #define GL_SYNC_GPU_COMMANDS_COMPLETE 0x9117
01527 #define GL_UNSIGNALED 0x9118
01528 #define GL_SIGNED 0x9119
01529 #define GL_ALREADY_SIGNED 0x911A
01530 #define GL_TIMEOUT_EXPIRED 0x911B
01531 #define GL_CONDITION_SATISFIED 0x911C
01532 #define GL_WAIT_FAILED 0x911D
01533 #define GL_TIMEOUT_IGNORED 0xFFFFFFFFFFFFFFFF
01534 #define GL_SYNC_FLUSH_COMMANDS_BIT 0x00000001
01535 #define GL_SAMPLE_POSITION 0x8E50
01536 #define GL_SAMPLE_MASK 0x8E51
01537 #define GL_SAMPLE_MASK_VALUE 0x8E52
01538 #define GL_MAX_SAMPLE_MASK_WORDS 0x8E59
01539 #define GL_TEXTURE_2D_MULTISAMPLE 0x9100
01540 #define GL_PROXY_TEXTURE_2D_MULTISAMPLE 0x9101
01541 #define GL_TEXTURE_2D_MULTISAMPLE_ARRAY 0x9102
01542 #define GL_PROXY_TEXTURE_2D_MULTISAMPLE_ARRAY 0x9103
01543 #define GL_TEXTURE_BINDING_2D_MULTISAMPLE 0x9104
01544 #define GL_TEXTURE_BINDING_2D_MULTISAMPLE_ARRAY 0x9105
01545 #define GL_TEXTURE_SAMPLES 0x9106
01546 #define GL_TEXTURE_FIXED_SAMPLE_LOCATIONS 0x9107
01547 #define GL_SAMPLER_2D_MULTISAMPLE 0x9108
01548 #define GL_INT_SAMPLER_2D_MULTISAMPLE 0x9109
01549 #define GL_UNSIGNED_INT_SAMPLER_2D_MULTISAMPLE 0x910A
01550 #define GL_SAMPLER_2D_MULTISAMPLE_ARRAY 0x910B
01551 #define GL_INT_SAMPLER_2D_MULTISAMPLE_ARRAY 0x910C
01552 #define GL_UNSIGNED_INT_SAMPLER_2D_MULTISAMPLE_ARRAY 0x910D
01553 #define GL_MAX_COLOR_TEXTURE_SAMPLES 0x910E
01554 #define GL_MAX_DEPTH_TEXTURE_SAMPLES 0x910F
01555 #define GL_MAX_INTEGER_SAMPLES 0x9110
01556 #define GL_VERTEX_ATTRIB_ARRAY_DIVISOR 0x88FE
01557 #define GL_SRC1_COLOR 0x88F9
01558 #define GL_ONE_MINUS_SRC1_COLOR 0x88FA
01559 #define GL_ONE_MINUS_SRC1_ALPHA 0x88FB
01560 #define GL_MAX_DUAL_SOURCE_DRAW_BUFFERS 0x88FC
01561 #define GL_ANY_SAMPLES_PASSED 0x8C2F
01562 #define GL_SAMPLER_BINDING 0x8919
01563 #define GL_RGB10_A2UI 0x906F
01564 #define GL_TEXTURE_SWIZZLE_R 0x8E42
01565 #define GL_TEXTURE_SWIZZLE_G 0x8E43
01566 #define GL_TEXTURE_SWIZZLE_B 0x8E44
01567 #define GL_TEXTURE_SWIZZLE_A 0x8E45
01568 #define GL_TEXTURE_SWIZZLE_RGBA 0x8E46
01569 #define GL_TIME_ELAPSED 0x88BF
01570 #define GL_TIMESTAMP 0x8E28
01571 #define GL_INT_2_10_10_10_REV 0x8D9F
01572 #define GL_SAMPLE_SHADING 0x8C36
01573 #define GL_MIN_SAMPLE_SHADING_VALUE 0x8C37
01574 #define GL_MIN_PROGRAM_TEXTURE_GATHER_OFFSET 0x8E5E
01575 #define GL_MAX_PROGRAM_TEXTURE_GATHER_OFFSET 0x8E5F
01576 #define GL_TEXTURE_CUBE_MAP_ARRAY 0x9009
01577 #define GL_TEXTURE_BINDING_CUBE_MAP_ARRAY 0x900A
01578 #define GL_PROXY_TEXTURE_CUBE_MAP_ARRAY 0x900B
01579 #define GL_SAMPLER_CUBE_MAP_ARRAY 0x900C
01580 #define GL_SAMPLER_CUBE_MAP_ARRAY_SHADOW 0x900D
01581 #define GL_INT_SAMPLER_CUBE_MAP_ARRAY 0x900E
01582 #define GL_UNSIGNED_INT_SAMPLER_CUBE_MAP_ARRAY 0x900F
01583 #define GL_DRAW_INDIRECT_BUFFER 0x8F3F
01584 #define GL_DRAW_INDIRECT_BUFFER_BINDING 0x8F43
01585 #define GL_GEOMETRY_SHADER_INVOCATIONS 0x887F
01586 #define GL_MAX_GEOMETRY_SHADER_INVOCATIONS 0x8E5A
01587 #define GL_MIN_FRAGMENT_INTERPOLATION_OFFSET 0x8E5B
01588 #define GL_MAX_FRAGMENT_INTERPOLATION_OFFSET 0x8E5C
01589 #define GL_FRAGMENT_INTERPOLATION_OFFSET_BITS 0x8E5D
01590 #define GL_MAX_VERTEX_STREAMS 0x8E71
```



```
01591 #define GL_DOUBLE_VEC2 0x8FFC
01592 #define GL_DOUBLE_VEC3 0x8FFD
01593 #define GL_DOUBLE_VEC4 0x8FFE
01594 #define GL_DOUBLE_MAT2 0x8F46
01595 #define GL_DOUBLE_MAT3 0x8F47
01596 #define GL_DOUBLE_MAT4 0x8F48
01597 #define GL_DOUBLE_MAT2x3 0x8F49
01598 #define GL_DOUBLE_MAT2x4 0x8F4A
01599 #define GL_DOUBLE_MAT3x2 0x8F4B
01600 #define GL_DOUBLE_MAT3x4 0x8F4C
01601 #define GL_DOUBLE_MAT4x2 0x8F4D
01602 #define GL_DOUBLE_MAT4x3 0x8F4E
01603 #define GL_ACTIVE_SUBROUTINES 0x8DE5
01604 #define GL_ACTIVE_SUBROUTINE_UNIFORMS 0x8DE6
01605 #define GL_ACTIVE_SUBROUTINE_UNIFORM_LOCATIONS 0x8E47
01606 #define GL_ACTIVE_SUBROUTINE_MAX_LENGTH 0x8E48
01607 #define GL_ACTIVE_SUBROUTINE_UNIFORM_MAX_LENGTH 0x8E49
01608 #define GL_MAX_SUBROUTINES 0x8DE7
01609 #define GL_MAX_SUBROUTINE_UNIFORM_LOCATIONS 0x8DE8
01610 #define GL_NUM_COMPATIBLE_SUBROUTINES 0x8E4A
01611 #define GL_COMPATIBLE_SUBROUTINES 0x8E4B
01612 #define GL_PATCHES 0x000E
01613 #define GL_PATCH_VERTICES 0x8E72
01614 #define GL_PATCH_DEFAULT_INNER_LEVEL 0x8E73
01615 #define GL_PATCH_DEFAULT_OUTER_LEVEL 0x8E74
01616 #define GL_TESS_CONTROL_OUTPUT_VERTICES 0x8E75
01617 #define GL_TESS_GEN_MODE 0x8E76
01618 #define GL_TESS_GEN_SPACING 0x8E77
01619 #define GL_TESS_GEN_VERTEX_ORDER 0x8E78
01620 #define GL_TESS_GEN_POINT_MODE 0x8E79
01621 #define GL_ISOLINES 0x8E7A
01622 #define GL_QUADS 0x0007
01623 #define GL_FRACTIONAL_ODD 0x8E7B
01624 #define GL_FRACTIONAL_EVEN 0x8E7C
01625 #define GL_MAX_PATCH_VERTICES 0x8E7D
01626 #define GL_MAX_TESS_GEN_LEVEL 0x8E7E
01627 #define GL_MAX_TESS_CONTROL_UNIFORM_COMPONENTS 0x8E7F
01628 #define GL_MAX_TESS_EVALUATION_UNIFORM_COMPONENTS 0x8E80
01629 #define GL_MAX_TESS_CONTROL_TEXTURE_IMAGE_UNITS 0x8E81
01630 #define GL_MAX_TESS_EVALUATION_TEXTURE_IMAGE_UNITS 0x8E82
01631 #define GL_MAX_TESS_CONTROL_OUTPUT_COMPONENTS 0x8E83
01632 #define GL_MAX_TESS_PATCH_COMPONENTS 0x8E84
01633 #define GL_MAX_TESS_CONTROL_TOTAL_OUTPUT_COMPONENTS 0x8E85
01634 #define GL_MAX_TESS_EVALUATION_OUTPUT_COMPONENTS 0x8E86
01635 #define GL_MAX_TESS_CONTROL_UNIFORM_BLOCKS 0x8E89
01636 #define GL_MAX_TESS_EVALUATION_UNIFORM_BLOCKS 0x8E8A
01637 #define GL_MAX_TESS_CONTROL_INPUT_COMPONENTS 0x886C
01638 #define GL_MAX_TESS_EVALUATION_INPUT_COMPONENTS 0x886D
01639 #define GL_MAX_COMBINED_TESS_CONTROL_UNIFORM_COMPONENTS 0x8E1E
01640 #define GL_MAX_COMBINED_TESS_EVALUATION_UNIFORM_COMPONENTS 0x8E1F
01641 #define GL_UNIFORM_BLOCK_REFERENCED_BY_TESS_CONTROL_SHADER 0x84F0
01642 #define GL_UNIFORM_BLOCK_REFERENCED_BY_TESS_EVALUATION_SHADER 0x84F1
01643 #define GL_TESS_EVALUATION_SHADER 0x8E87
01644 #define GL_TESS_CONTROL_SHADER 0x8E88
01645 #define GL_TRANSFORM_FEEDBACK 0x8E22
01646 #define GL_TRANSFORM_FEEDBACK_BUFFER_PAUSED 0x8E23
01647 #define GL_TRANSFORM_FEEDBACK_BUFFER_ACTIVE 0x8E24
01648 #define GL_TRANSFORM_FEEDBACK_BINDING 0x8E25
01649 #define GL_MAX_TRANSFORM_FEEDBACK_BUFFERS 0x8E70
01650 #define GL_FIXED 0x140C
01651 #define GL_IMPLEMENTATION_COLOR_READ_TYPE 0x8B9A
01652 #define GL_IMPLEMENTATION_COLOR_READ_FORMAT 0x8B9B
01653 #define GL_LOW_FLOAT 0x8DF0
01654 #define GL_MEDIUM_FLOAT 0x8DF1
01655 #define GL_HIGH_FLOAT 0x8DF2
01656 #define GL_LOW_INT 0x8DF3
01657 #define GL_MEDIUM_INT 0x8DF4
01658 #define GL_HIGH_INT 0x8DF5
01659 #define GL_SHADER_COMPILER 0x8DFA
01660 #define GL_SHADER_BINARY_FORMATS 0x8DF8
01661 #define GL_NUM_SHADER_BINARY_FORMATS 0x8DF9
01662 #define GL_MAX_VERTEX_UNIFORM_VECTORS 0x8DFB
01663 #define GL_MAX_VARYING_VECTORS 0x8DFC
01664 #define GL_MAX_FRAGMENT_UNIFORM_VECTORS 0x8DFD
01665 #define GL_RGB565 0x8D62
01666 #define GL_PROGRAM_BINARY_RETRIEVABLE_HINT 0x8257
01667 #define GL_PROGRAM_BINARY_LENGTH 0x8741
01668 #define GL_NUM_PROGRAM_BINARY_FORMATS 0x87FE
01669 #define GL_PROGRAM_BINARY_FORMATS 0x87FF
01670 #define GL_VERTEX_SHADER_BIT 0x00000001
01671 #define GL_FRAGMENT_SHADER_BIT 0x00000002
01672 #define GL_GEOMETRY_SHADER_BIT 0x00000004
01673 #define GL_TESS_CONTROL_SHADER_BIT 0x00000008
01674 #define GL_TESS_EVALUATION_SHADER_BIT 0x00000010
01675 #define GL_ALL_SHADER_BITS 0xFFFFFFFF
01676 #define GL_PROGRAM_SEPARABLE 0x8258
01677 #define GL_ACTIVE_PROGRAM 0x8259
```

```

01678 #define GL_PROGRAM_PIPELINE_BINDING 0x825A
01679 #define GL_MAX_VIEWPORTS 0x825B
01680 #define GL_VIEWPORT_SUBPIXEL_BITS 0x825C
01681 #define GL_VIEWPORT_BOUNDS_RANGE 0x825D
01682 #define GL_LAYER_PROVOKING_VERTEX 0x825E
01683 #define GL_VIEWPORT_INDEX_PROVOKING_VERTEX 0x825F
01684 #define GL_UNDEFINED_VERTEX 0x8260
01685 #ifndef GL_VERSION_1_0
01686 #define GL_VERSION_1_0 1
01687 GLAPI int GLAD_GL_VERSION_1_0;
01688 typedef void (APIENTRY PFNGLCULLFACEPROC) (GLenum mode);
01689 GLAPI PFNGLCULLFACEPROC glad_glCullFace;
01690 #define glCullFace glad_glCullFace
01691 typedef void (APIENTRY PFNGLFRONTFACEPROC) (GLenum mode);
01692 GLAPI PFNGLFRONTFACEPROC glad_glFrontFace;
01693 #define glFrontFace glad_glFrontFace
01694 typedef void (APIENTRY PFNGLHINTPROC) (GLenum target, GLenum mode);
01695 GLAPI PFNGLHINTPROC glad_glHint;
01696 #define glHint glad_glHint
01697 typedef void (APIENTRY PFNGLLINEWIDTHPROC) (GLfloat width);
01698 GLAPI PFNGLLINEWIDTHPROC glad_glLineWidth;
01699 #define glLineWidth glad_glLineWidth
01700 typedef void (APIENTRY PFNGLPOINTSIZEPROC) (GLfloat size);
01701 GLAPI PFNGLPOINTSIZEPROC glad_glPointSize;
01702 #define glPointSize glad_glPointSize
01703 typedef void (APIENTRY PFNGLPOLYGONMODEPROC) (GLenum face, GLenum mode);
01704 GLAPI PFNGLPOLYGONMODEPROC glad_glPolygonMode;
01705 #define glPolygonMode glad_glPolygonMode
01706 typedef void (APIENTRY PFNGLSCISSORPROC) (GLint x, GLint y, GLsizei width, GLsizei height);
01707 GLAPI PFNGLSCISSORPROC glad_glScissor;
01708 #define glScissor glad_glScissor
01709 typedef void (APIENTRY PFNGLTEXPARAMETERFPROC) (GLenum target, GLenum pname, GLfloat param);
01710 GLAPI PFNGLTEXPARAMETERFPROC glad_glTexParameterf;
01711 #define glTexParameterf glad_glTexParameterf
01712 typedef void (APIENTRY PFNGLTEXPARAMETERFVPROC) (GLenum target, GLenum pname, const GLfloat *params);
01713 GLAPI PFNGLTEXPARAMETERFVPROC glad_glTexParameterfv;
01714 #define glTexParameterfv glad_glTexParameterfv
01715 typedef void (APIENTRY PFNGLTEXPARAMETERIPROC) (GLenum target, GLenum pname, GLint param);
01716 GLAPI PFNGLTEXPARAMETERIPROC glad_glTexParameteri;
01717 #define glTexParameteri glad_glTexParameteri
01718 typedef void (APIENTRY PFNGLTEXPARAMETERIVPROC) (GLenum target, GLenum pname, const GLint *params);
01719 GLAPI PFNGLTEXPARAMETERIVPROC glad_glTexParameteriv;
01720 #define glTexParameteriv glad_glTexParameteriv
01721 typedef void (APIENTRY PFNGLTEXIMAGE1DPROC) (GLenum target, GLint level, GLint internalformat, GLsizei width, GLint border, GLenum format, GLenum type, const void *pixels);
01722 GLAPI PFNGLTEXIMAGE1DPROC glad_glTexImage1D;
01723 #define glTexImage1D glad_glTexImage1D
01724 typedef void (APIENTRY PFNGLTEXIMAGE2DPROC) (GLenum target, GLint level, GLint internalformat, GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const void *pixels);
01725 GLAPI PFNGLTEXIMAGE2DPROC glad_glTexImage2D;
01726 #define glTexImage2D glad_glTexImage2D
01727 typedef void (APIENTRY PFNGLDRAWBUFFERPROC) (GLenum buf);
01728 GLAPI PFNGLDRAWBUFFERPROC glad_glDrawBuffer;
01729 #define glDrawBuffer glad_glDrawBuffer
01730 typedef void (APIENTRY PFNGLCLEARPROC) (GLbitfield mask);
01731 GLAPI PFNGLCLEARPROC glad_glClear;
01732 #define glClear glad_glClear
01733 typedef void (APIENTRY PFNGLCLEARCOLORPROC) (GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha);
01734 GLAPI PFNGLCLEARCOLORPROC glad_glClearColor;
01735 #define glClearColor glad_glClearColor
01736 typedef void (APIENTRY PFNGLCLEARSTENCILPROC) (GLint s);
01737 GLAPI PFNGLCLEARSTENCILPROC glad_glClearStencil;
01738 #define glClearStencil glad_glClearStencil
01739 typedef void (APIENTRY PFNGLCLEARDEPTHPROC) (GLdouble depth);
01740 GLAPI PFNGLCLEARDEPTHPROC glad_glClearDepth;
01741 #define glClearDepth glad_glClearDepth
01742 typedef void (APIENTRY PFNGLSTENCILMASKPROC) (GLuint mask);
01743 GLAPI PFNGLSTENCILMASKPROC glad_glStencilMask;
01744 #define glStencilMask glad_glStencilMask
01745 typedef void (APIENTRY PFNGLCOLORMASKPROC) (GLboolean red, GLboolean green, GLboolean blue, GLboolean alpha);
01746 GLAPI PFNGLCOLORMASKPROC glad_glColorMask;
01747 #define glColorMask glad_glColorMask
01748 typedef void (APIENTRY PFNGLDEPTHMASKPROC) (GLboolean flag);
01749 GLAPI PFNGLDEPTHMASKPROC glad_glDepthMask;
01750 #define glDepthMask glad_glDepthMask
01751 typedef void (APIENTRY PFNGLDISABLEPROC) (GLenum cap);
01752 GLAPI PFNGLDISABLEPROC glad_glDisable;
01753 #define glDisable glad_glDisable
01754 typedef void (APIENTRY PFNGLENABLEPROC) (GLenum cap);
01755 GLAPI PFNGLENABLEPROC glad_glEnable;
01756 #define glEnable glad_glEnable
01757 typedef void (APIENTRY PFNGLFINISHPROC) (void);
01758 GLAPI PFNGLFINISHPROC glad_glFinish;
01759 #define glFinish glad_glFinish
01760 typedef void (APIENTRY PFNGLFLUSHPROC) (void);
01761 GLAPI PFNGLFLUSHPROC glad_glFlush;

```

```

01762 #define glFlush glad_glFlush
01763 typedef void (APIENTRY PFNGLBLENDFUNCPROC) (GLenum sfactor, GLenum dfactor);
01764 GLAPI PFNGLBLENDFUNCPROC glad_glBlendFunc;
01765 #define glBlendFunc glad_glBlendFunc
01766 typedef void (APIENTRY PFNGLLOGICOPPROC) (GLenum opcode);
01767 GLAPI PFNGLLOGICOPPROC glad_glLogicOp;
01768 #define glLogicOp glad_glLogicOp
01769 typedef void (APIENTRY PFNGLSTENCILFUNCPROC) (GLenum func, GLint ref, GLuint mask);
01770 GLAPI PFNGLSTENCILFUNCPROC glad_glStencilFunc;
01771 #define glStencilFunc glad_glStencilFunc
01772 typedef void (APIENTRY PFNGLSTENCILOPPROC) (GLenum fail, GLenum zfail, GLenum zpass);
01773 GLAPI PFNGLSTENCILOPPROC glad_glStencilOp;
01774 #define glStencilOp glad_glStencilOp
01775 typedef void (APIENTRY PFNGLDEPTHFUNCPROC) (GLenum func);
01776 GLAPI PFNGLDEPTHFUNCPROC glad_glDepthFunc;
01777 #define glDepthFunc glad_glDepthFunc
01778 typedef void (APIENTRY PFNGLPIXELSTOREFPROC) (GLenum pname, GLfloat param);
01779 GLAPI PFNGLPIXELSTOREFPROC glad_glPixelStoref;
01780 #define glPixelStoref glad_glPixelStoref
01781 typedef void (APIENTRY PFNGLPIXELSTOREIPROC) (GLenum pname, GLint param);
01782 GLAPI PFNGLPIXELSTOREIPROC glad_glPixelStorei;
01783 #define glPixelStorei glad_glPixelStorei
01784 typedef void (APIENTRY PFNGLREADBUFFERPROC) (GLenum src);
01785 GLAPI PFNGLREADBUFFERPROC glad_glReadBuffer;
01786 #define glReadBuffer glad_glReadBuffer
01787 typedef void (APIENTRY PFNGLREADPIXELSPROC) (GLint x, GLint y, GLsizei width, GLsizei height, GLenum format, GLenum type, void *pixels);
01788 GLAPI PFNGLREADPIXELSPROC glad_glReadPixels;
01789 #define glReadPixels glad_glReadPixels
01790 typedef void (APIENTRY PFNGLGETBOOLEANVPROC) (GLenum pname, GLboolean *data);
01791 GLAPI PFNGLGETBOOLEANVPROC glad_glGetBooleanv;
01792 #define glGetBooleanv glad_glGetBooleanv
01793 typedef void (APIENTRY PFNGLGETDOUBLEVPROC) (GLenum pname, GLdouble *data);
01794 GLAPI PFNGLGETDOUBLEVPROC glad_glGetDoublev;
01795 #define glGetDoublev glad_glGetDoublev
01796 typedef GLenum (APIENTRY PFNGLGETERRORPROC) (void);
01797 GLAPI PFNGLGETERRORPROC glad_glGetError;
01798 #define glGetError glad_glGetError
01799 typedef void (APIENTRY PFNGLGETFLOATVPROC) (GLenum pname, GLfloat *data);
01800 GLAPI PFNGLGETFLOATVPROC glad_glGetFloatv;
01801 #define glGetFloatv glad_glGetFloatv
01802 typedef void (APIENTRY PFNGLGETINTEGERVPROC) (GLenum pname, GLint *data);
01803 GLAPI PFNGLGETINTEGERVPROC glad_glGetIntegerv;
01804 #define glGetIntegerv glad_glGetIntegerv
01805 typedef const GLubyte * (APIENTRY PFNGLGETSTRINGPROC) (GLenum name);
01806 GLAPI PFNGLGETSTRINGPROC glad_glGetString;
01807 #define glGetString glad_glGetString
01808 typedef void (APIENTRY PFNGLGETTEXIMAGEPROC) (GLenum target, GLint level, GLenum format, GLenum type, void *pixels);
01809 GLAPI PFNGLGETTEXIMAGEPROC glad_glGetTexImage;
01810 #define glGetTexImage glad_glGetTexImage
01811 typedef void (APIENTRY PFNGLGETTEXPARAMETERFVPROC) (GLenum target, GLenum pname, GLfloat *params);
01812 GLAPI PFNGLGETTEXPARAMETERFVPROC glad_glGetTexParameterfv;
01813 #define glGetTexParameterfv glad_glGetTexParameterfv
01814 typedef void (APIENTRY PFNGLGETTEXPARAMETERIVPROC) (GLenum target, GLenum pname, GLint *params);
01815 GLAPI PFNGLGETTEXPARAMETERIVPROC glad_glGetTexParameteriv;
01816 #define glGetTexParameteriv glad_glGetTexParameteriv
01817 typedef void (APIENTRY PFNGLGETTEXLEVELPARAMETERFVPROC) (GLenum target, GLint level, GLenum pname, GLfloat *params);
01818 GLAPI PFNGLGETTEXLEVELPARAMETERFVPROC glad_glGetTexLevelParameterfv;
01819 #define glGetTexLevelParameterfv glad_glGetTexLevelParameterfv
01820 typedef void (APIENTRY PFNGLGETTEXLEVELPARAMETERIVPROC) (GLenum target, GLint level, GLenum pname, GLint *params);
01821 GLAPI PFNGLGETTEXLEVELPARAMETERIVPROC glad_glGetTexLevelParameteriv;
01822 #define glGetTexLevelParameteriv glad_glGetTexLevelParameteriv
01823 typedef GLboolean (APIENTRY PFNGLISENABLEDPROC) (GLenum cap);
01824 GLAPI PFNGLISENABLEDPROC glad_glIsEnabled;
01825 #define glIsEnabled glad_glIsEnabled
01826 typedef void (APIENTRY PFNGLDEPTHRANGEPROC) (GLdouble n, GLdouble f);
01827 GLAPI PFNGLDEPTHRANGEPROC glad_glDepthRange;
01828 #define glDepthRange glad_glDepthRange
01829 typedef void (APIENTRY PFNGLVIEWPORTPROC) (GLint x, GLint y, GLsizei width, GLsizei height);
01830 GLAPI PFNGLVIEWPORTPROC glad_glViewport;
01831 #define glViewport glad_glViewport
01832 #endif
01833 #ifndef GL_VERSION_1_1
01834 #define GL_VERSION_1_1 1
01835 GLAPI int GLAD_GL_VERSION_1_1;
01836 typedef void (APIENTRY PFNGLDRAWARRAYSPROC) (GLenum mode, GLint first, GLsizei count);
01837 GLAPI PFNGLDRAWARRAYSPROC glad_glDrawArrays;
01838 #define glDrawArrays glad_glDrawArrays
01839 typedef void (APIENTRY PFNGLDRAWELEMENTSPROC) (GLenum mode, GLsizei count, GLenum type, const void *indices);
01840 GLAPI PFNGLDRAWELEMENTSPROC glad_glDrawElements;
01841 #define glDrawElements glad_glDrawElements
01842 typedef void (APIENTRY PFNGLPOLYGONOFFSETPROC) (GLfloat factor, GLfloat units);
01843 GLAPI PFNGLPOLYGONOFFSETPROC glad_glPolygonOffset;

```

```

01844 #define glPolygonOffset glad_glPolygonOffset
01845 typedef void (APIENTRY PFNGLCOPYTEXIMAGE1DPROC) (GLenum target, GLint level, GLenum internalformat,
01846   GLint x, GLint y, GLsizei width, GLint border);
01846 GLAPI PFNGLCOPYTEXIMAGE1DPROC glad_glCopyTexImage1D;
01847 #define glCopyTexImage1D glad_glCopyTexImage1D
01848 typedef void (APIENTRY PFNGLCOPYTEXIMAGE2DPROC) (GLenum target, GLint level, GLenum internalformat,
01849   GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
01849 GLAPI PFNGLCOPYTEXIMAGE2DPROC glad_glCopyTexImage2D;
01850 #define glCopyTexImage2D glad_glCopyTexImage2D
01851 typedef void (APIENTRY PFNGLCOPYTEXSUBIMAGE1DPROC) (GLenum target, GLint level, GLint xoffset, GLint
01852   x, GLint y, GLsizei width);
01852 GLAPI PFNGLCOPYTEXSUBIMAGE1DPROC glad_glCopyTexSubImage1D;
01853 #define glCopyTexSubImage1D glad_glCopyTexSubImage1D
01854 typedef void (APIENTRY PFNGLCOPYTEXSUBIMAGE2DPROC) (GLenum target, GLint level, GLint xoffset, GLint
01855   yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
01855 GLAPI PFNGLCOPYTEXSUBIMAGE2DPROC glad_glCopyTexSubImage2D;
01856 #define glCopyTexSubImage2D glad_glCopyTexSubImage2D
01857 typedef void (APIENTRY PFNGLTEXSUBIMAGE1DPROC) (GLenum target, GLint level, GLint xoffset, GLsizei
01858   width, GLenum format, GLenum type, const void *pixels);
01858 GLAPI PFNGLTEXSUBIMAGE1DPROC glad_glTexSubImage1D;
01859 #define glTexSubImage1D glad_glTexSubImage1D
01860 typedef void (APIENTRY PFNGLTEXSUBIMAGE2DPROC) (GLenum target, GLint level, GLint xoffset, GLint
01861   yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *pixels);
01861 GLAPI PFNGLTEXSUBIMAGE2DPROC glad_glTexSubImage2D;
01862 #define glTexSubImage2D glad_glTexSubImage2D
01863 typedef void (APIENTRY PFNGLBINDTEXTUREPROC) (GLenum target, GLuint texture);
01864 GLAPI PFNGLBINDTEXTUREPROC glad_glBindTexture;
01865 #define glBindTexture glad_glBindTexture
01866 typedef void (APIENTRY PFNGLDELETETEXTURESPROC) (GLsizei n, const GLuint *textures);
01867 GLAPI PFNGLDELETETEXTURESPROC glad_glDeleteTextures;
01868 #define glDeleteTextures glad_glDeleteTextures
01869 typedef void (APIENTRY PFNGLGENTEXTURESPROC) (GLsizei n, GLuint *textures);
01870 GLAPI PFNGLGENTEXTURESPROC glad_glGenTextures;
01871 #define glGenTextures glad_glGenTextures
01872 typedef GLboolean (APIENTRY PFNGLISTEXTUREPROC) (GLuint texture);
01873 GLAPI PFNGLISTEXTUREPROC glad_glIsTexture;
01874 #define glIsTexture glad_glIsTexture
01875 #endif
01876 #ifndef GL_VERSION_1_2
01877 #define GL_VERSION_1_2 1
01878 GLAPI int GLAD_GL_VERSION_1_2;
01879 typedef void (APIENTRY PFNGLDRAWRANGEELEMENTSPROC) (GLenum mode, GLuint start, GLuint end, GLsizei
01880   count, GLenum type, const void *indices);
01880 GLAPI PFNGLDRAWRANGEELEMENTSPROC glad_glDrawRangeElements;
01881 #define glDrawRangeElements glad_glDrawRangeElements
01882 typedef void (APIENTRY PFNGLTEXIMAGE3DPROC) (GLenum target, GLint level, GLint internalformat, GLsizei
01883   width, GLsizei height, GLsizei depth, GLint border, GLenum format, GLenum type, const void *pixels);
01883 GLAPI PFNGLTEXIMAGE3DPROC glad_glTexImage3D;
01884 #define glTexImage3D glad_glTexImage3D
01885 typedef void (APIENTRY PFNGLTEXSUBIMAGE3DPROC) (GLenum target, GLint level, GLint xoffset, GLint
01886   yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type,
01887   const void *pixels);
01886 GLAPI PFNGLTEXSUBIMAGE3DPROC glad_glTexSubImage3D;
01887 #define glTexSubImage3D glad_glTexSubImage3D
01888 typedef void (APIENTRY PFNGLCOPYTEXSUBIMAGE3DPROC) (GLenum target, GLint level, GLint xoffset, GLint
01889   yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
01889 GLAPI PFNGLCOPYTEXSUBIMAGE3DPROC glad_glCopyTexSubImage3D;
01890 #define glCopyTexSubImage3D glad_glCopyTexSubImage3D
01891 #endif
01892 #ifndef GL_VERSION_1_3
01893 #define GL_VERSION_1_3 1
01894 GLAPI int GLAD_GL_VERSION_1_3;
01895 typedef void (APIENTRY PFNGLACTIVETEXTUREPROC) (GLenum texture);
01896 GLAPI PFNGLACTIVETEXTUREPROC glad_glActiveTexture;
01897 #define glActiveTexture glad_glActiveTexture
01898 typedef void (APIENTRY PFNGLSAMPLECOVERAGEPROC) (GLfloat value, GLboolean invert);
01899 GLAPI PFNGLSAMPLECOVERAGEPROC glad_glSampleCoverage;
01900 #define glSampleCoverage glad_glSampleCoverage
01901 typedef void (APIENTRY PFNGLCOMPRESSEDTEXIMAGE3DPROC) (GLenum target, GLint level, GLenum
01902   internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei imageSize, const
01903   void *data);
01902 GLAPI PFNGLCOMPRESSEDTEXIMAGE3DPROC glad_glCompressedTexImage3D;
01903 #define glCompressedTexImage3D glad_glCompressedTexImage3D
01904 typedef void (APIENTRY PFNGLCOMPRESSEDTEXIMAGE2DPROC) (GLenum target, GLint level, GLenum
01905   internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const void *data);
01905 GLAPI PFNGLCOMPRESSEDTEXIMAGE2DPROC glad_glCompressedTexImage2D;
01906 #define glCompressedTexImage2D glad_glCompressedTexImage2D
01907 typedef void (APIENTRY PFNGLCOMPRESSEDTEXIMAGE1DPROC) (GLenum target, GLint level, GLenum
01908   internalformat, GLsizei width, GLint border, GLsizei imageSize, const void *data);
01908 GLAPI PFNGLCOMPRESSEDTEXIMAGE1DPROC glad_glCompressedTexImage1D;
01909 #define glCompressedTexImage1D glad_glCompressedTexImage1D
01910 typedef void (APIENTRY PFNGLCOMPRESSEDTEXSUBIMAGE3DPROC) (GLenum target, GLint level, GLint xoffset,
01911   GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLsizei
01912   imageSize, const void *data);
01911 GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE3DPROC glad_glCompressedTexSubImage3D;
01912 #define glCompressedTexSubImage3D glad_glCompressedTexSubImage3D
01913 typedef void (APIENTRY PFNGLCOMPRESSEDTEXSUBIMAGE2DPROC) (GLenum target, GLint level, GLint xoffset,

```

```

    GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void *data);
01914 GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE2DPROC glad_glCompressedTexSubImage2D;
01915 #define glCompressedTexSubImage2D glad_glCompressedTexSubImage2D
01916 typedef void (APIENTRY PFNGLCOMPRESSEDTEXSUBIMAGE1DPROC) (GLenum target, GLint level, GLint xoffset,
    GLsizei width, GLenum format, GLsizei imageSize, const void *data);
01917 GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE1DPROC glad_glCompressedTexSubImage1D;
01918 #define glCompressedTexSubImage1D glad_glCompressedTexSubImage1D
01919 typedef void (APIENTRY PFNGLGETCOMPRESSEDTEXIMAGEPROC) (GLenum target, GLint level, void *img);
01920 GLAPI PFNGLGETCOMPRESSEDTEXIMAGEPROC glad_glGetCompressedTexImage;
01921 #define glGetCompressedTexImage glad_glGetCompressedTexImage
01922 #endif
01923 #ifndef GL_VERSION_1_4
01924 #define GL_VERSION_1_4 1
01925 GLAPI int GLAD_GL_VERSION_1_4;
01926 typedef void (APIENTRY PFNGLBLENDFUNCSEPARATEPROC) (GLenum sfactorRGB, GLenum dfactorRGB, GLenum
    sfactorAlpha, GLenum dfactorAlpha);
01927 GLAPI PFNGLBLENDFUNCSEPARATEPROC glad_glBlendFuncSeparate;
01928 #define glBlendFuncSeparate glad_glBlendFuncSeparate
01929 typedef void (APIENTRY PFNGLMULTIDRAWARRAYSPROC) (GLenum mode, const GLint *first, const GLsizei
    *count, GLsizei drawcount);
01930 GLAPI PFNGLMULTIDRAWARRAYSPROC glad_glMultiDrawArrays;
01931 #define glMultiDrawArrays glad_glMultiDrawArrays
01932 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSPROC) (GLenum mode, const GLsizei *count, GLenum type,
    const void *const*indices, GLsizei drawcount);
01933 GLAPI PFNGLMULTIDRAWELEMENTSPROC glad_glMultiDrawElements;
01934 #define glMultiDrawElements glad_glMultiDrawElements
01935 typedef void (APIENTRY PFNGLPOINTPARAMETERFPROC) (GLenum pname, GLfloat param);
01936 GLAPI PFNGLPOINTPARAMETERFPROC glad_glPointParameterf;
01937 #define glPointParameterf glad_glPointParameterf
01938 typedef void (APIENTRY PFNGLPOINTPARAMETERFVPROC) (GLenum pname, const GLfloat *params);
01939 GLAPI PFNGLPOINTPARAMETERFVPROC glad_glPointParameterfv;
01940 #define glPointParameterfv glad_glPointParameterfv
01941 typedef void (APIENTRY PFNGLPOINTPARAMETERIPROC) (GLenum pname, GLint param);
01942 GLAPI PFNGLPOINTPARAMETERIPROC glad_glPointParameteri;
01943 #define glPointParameteri glad_glPointParameteri
01944 typedef void (APIENTRY PFNGLPOINTPARAMETERIVPROC) (GLenum pname, const GLint *params);
01945 GLAPI PFNGLPOINTPARAMETERIVPROC glad_glPointParameteriv;
01946 #define glPointParameteriv glad_glPointParameteriv
01947 typedef void (APIENTRY PFNGLBLENDCOLORPROC) (GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha);
01948 GLAPI PFNGLBLENDCOLORPROC glad_glBlendColor;
01949 #define glBlendColor glad_glBlendColor
01950 typedef void (APIENTRY PFNGLBLENDEQUATIONPROC) (GLenum mode);
01951 GLAPI PFNGLBLENDEQUATIONPROC glad_glBlendEquation;
01952 #define glBlendEquation glad_glBlendEquation
01953 #endif
01954 #ifndef GL_VERSION_1_5
01955 #define GL_VERSION_1_5 1
01956 GLAPI int GLAD_GL_VERSION_1_5;
01957 typedef void (APIENTRY PFNGLGENQUERIESPROC) (GLsizei n, GLuint *ids);
01958 GLAPI PFNGLGENQUERIESPROC glad_glGenQueries;
01959 #define glGenQueries glad_glGenQueries
01960 typedef void (APIENTRY PFNGLDELETEQUERIESPROC) (GLsizei n, const GLuint *ids);
01961 GLAPI PFNGLDELETEQUERIESPROC glad_glDeleteQueries;
01962 #define glDeleteQueries glad_glDeleteQueries
01963 typedef GLboolean (APIENTRY PFNGLISQUERYPROC) (GLuint id);
01964 GLAPI PFNGLISQUERYPROC glad_glIsQuery;
01965 #define glIsQuery glad_glIsQuery
01966 typedef void (APIENTRY PFNGLBEGINQUERYPROC) (GLenum target, GLuint id);
01967 GLAPI PFNGLBEGINQUERYPROC glad_glBeginQuery;
01968 #define glBeginQuery glad_glBeginQuery
01969 typedef void (APIENTRY PFNGLENDQUERYPROC) (GLenum target);
01970 GLAPI PFNGLENDQUERYPROC glad_glEndQuery;
01971 #define glEndQuery glad_glEndQuery
01972 typedef void (APIENTRY PFNGLGETQUERYIVPROC) (GLenum target, GLenum pname, GLint *params);
01973 GLAPI PFNGLGETQUERYIVPROC glad_glGetQueryiv;
01974 #define glGetQueryiv glad_glGetQueryiv
01975 typedef void (APIENTRY PFNGLGETQUERYOBJECTIVPROC) (GLuint id, GLenum pname, GLint *params);
01976 GLAPI PFNGLGETQUERYOBJECTIVPROC glad_glGetQueryObjectiv;
01977 #define glGetQueryObjectiv glad_glGetQueryObjectiv
01978 typedef void (APIENTRY PFNGLGETQUERYOBJECTUIVPROC) (GLuint id, GLenum pname, GLuint *params);
01979 GLAPI PFNGLGETQUERYOBJECTUIVPROC glad_glGetQueryObjectuiv;
01980 #define glGetQueryObjectuiv glad_glGetQueryObjectuiv
01981 typedef void (APIENTRY PFNGLBINDBUFFERPROC) (GLenum target, GLuint buffer);
01982 GLAPI PFNGLBINDBUFFERPROC glad_glBindBuffer;
01983 #define glBindBuffer glad_glBindBuffer
01984 typedef void (APIENTRY PFNGLDELETEBUFFERSPROC) (GLsizei n, const GLuint *buffers);
01985 GLAPI PFNGLDELETEBUFFERSPROC glad_glDeleteBuffers;
01986 #define glDeleteBuffers glad_glDeleteBuffers
01987 typedef void (APIENTRY PFNGLGENBUFFERSPROC) (GLsizei n, GLuint *buffers);
01988 GLAPI PFNGLGENBUFFERSPROC glad_glGenBuffers;
01989 #define glGenBuffers glad_glGenBuffers
01990 typedef GLboolean (APIENTRY PFNGLISBUFFERPROC) (GLuint buffer);
01991 GLAPI PFNGLISBUFFERPROC glad_glIsBuffer;
01992 #define glIsBuffer glad_glIsBuffer
01993 typedef void (APIENTRY PFNGLBUFFERDATAPROC) (GLenum target, GLsizeiptr size, const void *data, GLenum
    usage);
01994 GLAPI PFNGLBUFFERDATAPROC glad_glBufferData;

```



```

01995 #define glBufferData glad_glBufferData
01996 typedef void (APIENTRY PFNGLBUFFERSUBDATAPROC) (GLenum target, GLintptr offset, GLsizeiptr size, const
void *data);
01997 GLAPI PFNGLBUFFERSUBDATAPROC glad_glBufferSubData;
01998 #define glBufferSubData glad_glBufferSubData
01999 typedef void (APIENTRY PFNGLGETBUFFERSUBDATAPROC) (GLenum target, GLintptr offset, GLsizeiptr size,
void *data);
02000 GLAPI PFNGLGETBUFFERSUBDATAPROC glad_glGetBufferSubData;
02001 #define glGetBufferSubData glad_glGetBufferSubData
02002 typedef void * (APIENTRY PFNGLMAPBUFFERPROC) (GLenum target, GLenum access);
02003 GLAPI PFNGLMAPBUFFERPROC glad_glMapBuffer;
02004 #define glMapBuffer glad_glMapBuffer
02005 typedef GLboolean (APIENTRY PFNGLUNMAPBUFFERPROC) (GLenum target);
02006 GLAPI PFNGLUNMAPBUFFERPROC glad_glUnmapBuffer;
02007 #define glUnmapBuffer glad_glUnmapBuffer
02008 typedef void (APIENTRY PFNGLGETBUFFERPARAMETERIVPROC) (GLenum target, GLenum pname, GLint *params);
02009 GLAPI PFNGLGETBUFFERPARAMETERIVPROC glad_glGetBufferParameteriv;
02010 #define glGetBufferParameteriv glad_glGetBufferParameteriv
02011 typedef void (APIENTRY PFNGLGETBUFFERPOINTERVPROC) (GLenum target, GLenum pname, void **params);
02012 GLAPI PFNGLGETBUFFERPOINTERVPROC glad_glGetBufferPointerv;
02013 #define glGetBufferPointerv glad_glGetBufferPointerv
02014 #endif
02015 #ifndef GL_VERSION_2_0
02016 #define GL_VERSION_2_0 1
02017 GLAPI int GLAD_GL_VERSION_2_0;
02018 typedef void (APIENTRY PFNGLBLENDEQUATIONSEPARATEPROC) (GLenum modeRGB, GLenum modeAlpha);
02019 GLAPI PFNGLBLENDEQUATIONSEPARATEPROC glad_glBlendEquationSeparate;
02020 #define glBlendEquationSeparate glad_glBlendEquationSeparate
02021 typedef void (APIENTRY PFNGLDRAWBUFFERSPROC) (GLsizei n, const GLenum *bufs);
02022 GLAPI PFNGLDRAWBUFFERSPROC glad_glDrawBuffers;
02023 #define glDrawBuffers glad_glDrawBuffers
02024 typedef void (APIENTRY PFNGLSTENCILOPSEPARATEPROC) (GLenum face, GLenum sfail, GLenum dpfail, GLenum
dppass);
02025 GLAPI PFNGLSTENCILOPSEPARATEPROC glad_glStencilOpSeparate;
02026 #define glStencilOpSeparate glad_glStencilOpSeparate
02027 typedef void (APIENTRY PFNGLSTENCILFUNCSEPARATEPROC) (GLenum face, GLenum func, GLint ref, GLuint
mask);
02028 GLAPI PFNGLSTENCILFUNCSEPARATEPROC glad_glStencilFuncSeparate;
02029 #define glStencilFuncSeparate glad_glStencilFuncSeparate
02030 typedef void (APIENTRY PFNGLSTENCILMASKSEPARATEPROC) (GLenum face, GLuint mask);
02031 GLAPI PFNGLSTENCILMASKSEPARATEPROC glad_glStencilMaskSeparate;
02032 #define glStencilMaskSeparate glad_glStencilMaskSeparate
02033 typedef void (APIENTRY PFNGLATTACHSHADERPROC) (GLuint program, GLuint shader);
02034 GLAPI PFNGLATTACHSHADERPROC glad_glAttachShader;
02035 #define glAttachShader glad_glAttachShader
02036 typedef void (APIENTRY PFNGLBINDATTRIBLOCATIONPROC) (GLuint program, GLuint index, const GLchar
*name);
02037 GLAPI PFNGLBINDATTRIBLOCATIONPROC glad_glBindAttribLocation;
02038 #define glBindAttribLocation glad_glBindAttribLocation
02039 typedef void (APIENTRY PFNGLCOMPILESHADERPROC) (GLuint shader);
02040 GLAPI PFNGLCOMPILESHADERPROC glad_glCompileShader;
02041 #define glCompileShader glad_glCompileShader
02042 typedef GLuint (APIENTRY PFNGLCREATEPROGRAMPROC) (void);
02043 GLAPI PFNGLCREATEPROGRAMPROC glad_glCreateProgram;
02044 #define glCreateProgram glad_glCreateProgram
02045 typedef GLuint (APIENTRY PFNGLCREATESHADERPROC) (GLenum type);
02046 GLAPI PFNGLCREATESHADERPROC glad_glCreateShader;
02047 #define glCreateShader glad_glCreateShader
02048 typedef void (APIENTRY PFNGLDELETEPROGRAMPROC) (GLuint program);
02049 GLAPI PFNGLDELETEPROGRAMPROC glad_glDeleteProgram;
02050 #define glDeleteProgram glad_glDeleteProgram
02051 typedef void (APIENTRY PFNGLDELETESHADERPROC) (GLuint shader);
02052 GLAPI PFNGLDELETESHADERPROC glad_glDeleteShader;
02053 #define glDeleteShader glad_glDeleteShader
02054 typedef void (APIENTRY PFNGLDETACHSHADERPROC) (GLuint program, GLuint shader);
02055 GLAPI PFNGLDETACHSHADERPROC glad_glDetachShader;
02056 #define glDetachShader glad_glDetachShader
02057 typedef void (APIENTRY PFNGLDISABLEVERTEXATTRIBARRAYPROC) (GLuint index);
02058 GLAPI PFNGLDISABLEVERTEXATTRIBARRAYPROC glad_glDisableVertexAttribArray;
02059 #define glDisableVertexAttribArray glad_glDisableVertexAttribArray
02060 typedef void (APIENTRY PFNGLENABLEVERTEXATTRIBARRAYPROC) (GLuint index);
02061 GLAPI PFNGLENABLEVERTEXATTRIBARRAYPROC glad_glEnableVertexAttribArray;
02062 #define glEnableVertexAttribArray glad_glEnableVertexAttribArray
02063 typedef void (APIENTRY PFNGLGETACTIVEATTRIBPROC) (GLuint program, GLuint index, GLsizei bufSize,
GLsizei *length, GLint *size, GLenum *type, GLchar *name);
02064 GLAPI PFNGLGETACTIVEATTRIBPROC glad_glGetActiveAttrib;
02065 #define glGetActiveAttrib glad_glGetActiveAttrib
02066 typedef void (APIENTRY PFNGLGETACTIVEUNIFORMPROC) (GLuint program, GLuint index, GLsizei bufSize,
GLsizei *length, GLint *size, GLenum *type, GLchar *name);
02067 GLAPI PFNGLGETACTIVEUNIFORMPROC glad_glGetActiveUniform;
02068 #define glGetActiveUniform glad_glGetActiveUniform
02069 typedef void (APIENTRY PFNGLGETATTACHEDSHADERSPROC) (GLuint program, GLsizei maxCount, GLsizei *count,
GLuint *shaders);
02070 GLAPI PFNGLGETATTACHEDSHADERSPROC glad_glGetAttachedShaders;
02071 #define glGetAttachedShaders glad_glGetAttachedShaders
02072 typedef GLint (APIENTRY PFNGLGETATTRIBLOCATIONPROC) (GLuint program, const GLchar *name);
02073 GLAPI PFNGLGETATTRIBLOCATIONPROC glad_glGetAttribLocation;

```

```

02074 #define glGetAttribLocation glad_glGetAttribLocation
02075 typedef void (APIENTRY PFNGLGETPROGRAMIVPROC) (GLuint program, GLenum pname, GLint *params);
02076 GLAPI PFNGLGETPROGRAMIVPROC glad_glGetProgramiv;
02077 #define glGetProgramiv glad_glGetProgramiv
02078 typedef void (APIENTRY PFNGLGETPROGRAMINFOLOGPROC) (GLuint program, GLsizei bufSize, GLsizei *length,
GLchar *infoLog);
02079 GLAPI PFNGLGETPROGRAMINFOLOGPROC glad_glGetProgramInfoLog;
02080 #define glGetProgramInfoLog glad_glGetProgramInfoLog
02081 typedef void (APIENTRY PFNGLGETSHADERIVPROC) (GLuint shader, GLenum pname, GLint *params);
02082 GLAPI PFNGLGETSHADERIVPROC glad_glGetShaderiv;
02083 #define glGetShaderiv glad_glGetShaderiv
02084 typedef void (APIENTRY PFNGLGETSHADERINFOLOGPROC) (GLuint shader, GLsizei bufSize, GLsizei *length,
GLchar *infoLog);
02085 GLAPI PFNGLGETSHADERINFOLOGPROC glad_glGetShaderInfoLog;
02086 #define glGetShaderInfoLog glad_glGetShaderInfoLog
02087 typedef void (APIENTRY PFNGLGETSHADERSOURCEPROC) (GLuint shader, GLsizei bufSize, GLsizei *length,
GLchar *source);
02088 GLAPI PFNGLGETSHADERSOURCEPROC glad_glGetShaderSource;
02089 #define glGetShaderSource glad_glGetShaderSource
02090 typedef GLint (APIENTRY PFNGLGETUNIFORMLOCATIONPROC) (GLuint program, const GLchar *name);
02091 GLAPI PFNGLGETUNIFORMLOCATIONPROC glad_glGetUniformLocation;
02092 #define glGetUniformLocation glad_glGetUniformLocation
02093 typedef void (APIENTRY PFNGLGETUNIFORMFVPROC) (GLuint program, GLint location, GLfloat *params);
02094 GLAPI PFNGLGETUNIFORMFVPROC glad_glGetUniformfv;
02095 #define glGetUniformfv glad_glGetUniformfv
02096 typedef void (APIENTRY PFNGLGETUNIFORMIVPROC) (GLuint program, GLint location, GLint *params);
02097 GLAPI PFNGLGETUNIFORMIVPROC glad_glGetUniformiv;
02098 #define glGetUniformiv glad_glGetUniformiv
02099 typedef void (APIENTRY PFNGLGETVERTEXATTRIBDVPROC) (GLuint index, GLenum pname, GLdouble *params);
02100 GLAPI PFNGLGETVERTEXATTRIBDVPROC glad_glGetVertexAttribdv;
02101 #define glGetVertexAttribdv glad_glGetVertexAttribdv
02102 typedef void (APIENTRY PFNGLGETVERTEXATTRIBFVPROC) (GLuint index, GLenum pname, GLfloat *params);
02103 GLAPI PFNGLGETVERTEXATTRIBFVPROC glad_glGetVertexAttribfv;
02104 #define glGetVertexAttribfv glad_glGetVertexAttribfv
02105 typedef void (APIENTRY PFNGLGETVERTEXATTRIBIVPROC) (GLuint index, GLenum pname, GLint *params);
02106 GLAPI PFNGLGETVERTEXATTRIBIVPROC glad_glGetVertexAttribiv;
02107 #define glGetVertexAttribiv glad_glGetVertexAttribiv
02108 typedef void (APIENTRY PFNGLGETVERTEXATTRIBPOINTERVPROC) (GLuint index, GLenum pname, void **pointer);
02109 GLAPI PFNGLGETVERTEXATTRIBPOINTERVPROC glad_glGetVertexAttribPointerv;
02110 #define glGetVertexAttribPointerv glad_glGetVertexAttribPointerv
02111 typedef GLboolean (APIENTRY PFNGLISPROGRAMPROC) (GLuint program);
02112 GLAPI PFNGLISPROGRAMPROC glad_glIsProgram;
02113 #define glIsProgram glad_glIsProgram
02114 typedef GLboolean (APIENTRY PFNGLISSHADERPROC) (GLuint shader);
02115 GLAPI PFNGLISSHADERPROC glad_glIsShader;
02116 #define glIsShader glad_glIsShader
02117 typedef void (APIENTRY PFNGLLINKPROGRAMPROC) (GLuint program);
02118 GLAPI PFNGLLINKPROGRAMPROC glad_glLinkProgram;
02119 #define glLinkProgram glad_glLinkProgram
02120 typedef void (APIENTRY PFNGLSHADERSOURCEPROC) (GLuint shader, GLsizei count, const GLchar
*const*string, const GLint *length);
02121 GLAPI PFNGLSHADERSOURCEPROC glad_glShaderSource;
02122 #define glShaderSource glad_glShaderSource
02123 typedef void (APIENTRY PFNGLUSEPROGRAMPROC) (GLuint program);
02124 GLAPI PFNGLUSEPROGRAMPROC glad_glUseProgram;
02125 #define glUseProgram glad_glUseProgram
02126 typedef void (APIENTRY PFNGLUNIFORM1FPROC) (GLint location, GLfloat v0);
02127 GLAPI PFNGLUNIFORM1FPROC glad_glUniform1f;
02128 #define glUniform1f glad_glUniform1f
02129 typedef void (APIENTRY PFNGLUNIFORM2FPROC) (GLint location, GLfloat v0, GLfloat v1);
02130 GLAPI PFNGLUNIFORM2FPROC glad_glUniform2f;
02131 #define glUniform2f glad_glUniform2f
02132 typedef void (APIENTRY PFNGLUNIFORM3FPROC) (GLint location, GLfloat v0, GLfloat v1, GLfloat v2);
02133 GLAPI PFNGLUNIFORM3FPROC glad_glUniform3f;
02134 #define glUniform3f glad_glUniform3f
02135 typedef void (APIENTRY PFNGLUNIFORM4FPROC) (GLint location, GLfloat v0, GLfloat v1, GLfloat v2,
GLfloat v3);
02136 GLAPI PFNGLUNIFORM4FPROC glad_glUniform4f;
02137 #define glUniform4f glad_glUniform4f
02138 typedef void (APIENTRY PFNGLUNIFORM1IPROC) (GLint location, GLint v0);
02139 GLAPI PFNGLUNIFORM1IPROC glad_glUniform1i;
02140 #define glUniform1i glad_glUniform1i
02141 typedef void (APIENTRY PFNGLUNIFORM2IPROC) (GLint location, GLint v0, GLint v1);
02142 GLAPI PFNGLUNIFORM2IPROC glad_glUniform2i;
02143 #define glUniform2i glad_glUniform2i
02144 typedef void (APIENTRY PFNGLUNIFORM3IPROC) (GLint location, GLint v0, GLint v1, GLint v2);
02145 GLAPI PFNGLUNIFORM3IPROC glad_glUniform3i;
02146 #define glUniform3i glad_glUniform3i
02147 typedef void (APIENTRY PFNGLUNIFORM4IPROC) (GLint location, GLint v0, GLint v1, GLint v2, GLint v3);
02148 GLAPI PFNGLUNIFORM4IPROC glad_glUniform4i;
02149 #define glUniform4i glad_glUniform4i
02150 typedef void (APIENTRY PFNGLUNIFORM1FVPROC) (GLint location, GLsizei count, const GLfloat *value);
02151 GLAPI PFNGLUNIFORM1FVPROC glad_glUniform1fv;
02152 #define glUniform1fv glad_glUniform1fv
02153 typedef void (APIENTRY PFNGLUNIFORM2FVPROC) (GLint location, GLsizei count, const GLfloat *value);
02154 GLAPI PFNGLUNIFORM2FVPROC glad_glUniform2fv;
02155 #define glUniform2fv glad_glUniform2fv

```

```
02156 typedef void (APIENTRY PFNGLUNIFORM3FVPROC) (GLint location, GLsizei count, const GLfloat *value);
02157 GLAPI PFNGLUNIFORM3FVPROC glad_glUniform3fv;
02158 #define glUniform3fv glad_glUniform3fv
02159 typedef void (APIENTRY PFNGLUNIFORM4FVPROC) (GLint location, GLsizei count, const GLfloat *value);
02160 GLAPI PFNGLUNIFORM4FVPROC glad_glUniform4fv;
02161 #define glUniform4fv glad_glUniform4fv
02162 typedef void (APIENTRY PFNGLUNIFORM1IVPROC) (GLint location, GLsizei count, const GLint *value);
02163 GLAPI PFNGLUNIFORM1IVPROC glad_glUniform1iv;
02164 #define glUniform1iv glad_glUniform1iv
02165 typedef void (APIENTRY PFNGLUNIFORM2IVPROC) (GLint location, GLsizei count, const GLint *value);
02166 GLAPI PFNGLUNIFORM2IVPROC glad_glUniform2iv;
02167 #define glUniform2iv glad_glUniform2iv
02168 typedef void (APIENTRY PFNGLUNIFORM3IVPROC) (GLint location, GLsizei count, const GLint *value);
02169 GLAPI PFNGLUNIFORM3IVPROC glad_glUniform3iv;
02170 #define glUniform3iv glad_glUniform3iv
02171 typedef void (APIENTRY PFNGLUNIFORM4IVPROC) (GLint location, GLsizei count, const GLint *value);
02172 GLAPI PFNGLUNIFORM4IVPROC glad_glUniform4iv;
02173 #define glUniform4iv glad_glUniform4iv
02174 typedef void (APIENTRY PFNGLUNIFORMMATRIX2FVPROC) (GLint location, GLsizei count, GLboolean transpose,
const GLfloat *value);
02175 GLAPI PFNGLUNIFORMMATRIX2FVPROC glad_glUniformMatrix2fv;
02176 #define glUniformMatrix2fv glad_glUniformMatrix2fv
02177 typedef void (APIENTRY PFNGLUNIFORMMATRIX3FVPROC) (GLint location, GLsizei count, GLboolean transpose,
const GLfloat *value);
02178 GLAPI PFNGLUNIFORMMATRIX3FVPROC glad_glUniformMatrix3fv;
02179 #define glUniformMatrix3fv glad_glUniformMatrix3fv
02180 typedef void (APIENTRY PFNGLUNIFORMMATRIX4FVPROC) (GLint location, GLsizei count, GLboolean transpose,
const GLfloat *value);
02181 GLAPI PFNGLUNIFORMMATRIX4FVPROC glad_glUniformMatrix4fv;
02182 #define glUniformMatrix4fv glad_glUniformMatrix4fv
02183 typedef void (APIENTRY PFNGLVALIDATEPROGRAMPROC) (GLuint program);
02184 GLAPI PFNGLVALIDATEPROGRAMPROC glad_glValidateProgram;
02185 #define glValidateProgram glad_glValidateProgram
02186 typedef void (APIENTRY PFNGLVERTEXATTRIB1DPROC) (GLuint index, GLdouble x);
02187 GLAPI PFNGLVERTEXATTRIB1DPROC glad_glVertexAttrib1d;
02188 #define glVertexAttrib1d glad_glVertexAttrib1d
02189 typedef void (APIENTRY PFNGLVERTEXATTRIB1DVPROC) (GLuint index, const GLdouble *v);
02190 GLAPI PFNGLVERTEXATTRIB1DVPROC glad_glVertexAttrib1dv;
02191 #define glVertexAttrib1dv glad_glVertexAttrib1dv
02192 typedef void (APIENTRY PFNGLVERTEXATTRIB1FPROC) (GLuint index, GLfloat x);
02193 GLAPI PFNGLVERTEXATTRIB1FPROC glad_glVertexAttrib1f;
02194 #define glVertexAttrib1f glad_glVertexAttrib1f
02195 typedef void (APIENTRY PFNGLVERTEXATTRIB1FVPROC) (GLuint index, const GLfloat *v);
02196 GLAPI PFNGLVERTEXATTRIB1FVPROC glad_glVertexAttrib1fv;
02197 #define glVertexAttrib1fv glad_glVertexAttrib1fv
02198 typedef void (APIENTRY PFNGLVERTEXATTRIB1SPROC) (GLuint index, GLshort x);
02199 GLAPI PFNGLVERTEXATTRIB1SPROC glad_glVertexAttrib1s;
02200 #define glVertexAttrib1s glad_glVertexAttrib1s
02201 typedef void (APIENTRY PFNGLVERTEXATTRIB1SVPROC) (GLuint index, const GLshort *v);
02202 GLAPI PFNGLVERTEXATTRIB1SVPROC glad_glVertexAttrib1sv;
02203 #define glVertexAttrib1sv glad_glVertexAttrib1sv
02204 typedef void (APIENTRY PFNGLVERTEXATTRIB2DPROC) (GLuint index, GLdouble x, GLdouble y);
02205 GLAPI PFNGLVERTEXATTRIB2DPROC glad_glVertexAttrib2d;
02206 #define glVertexAttrib2d glad_glVertexAttrib2d
02207 typedef void (APIENTRY PFNGLVERTEXATTRIB2DVPROC) (GLuint index, const GLdouble *v);
02208 GLAPI PFNGLVERTEXATTRIB2DVPROC glad_glVertexAttrib2dv;
02209 #define glVertexAttrib2dv glad_glVertexAttrib2dv
02210 typedef void (APIENTRY PFNGLVERTEXATTRIB2FPROC) (GLuint index, GLfloat x, GLfloat y);
02211 GLAPI PFNGLVERTEXATTRIB2FPROC glad_glVertexAttrib2f;
02212 #define glVertexAttrib2f glad_glVertexAttrib2f
02213 typedef void (APIENTRY PFNGLVERTEXATTRIB2FVPROC) (GLuint index, const GLfloat *v);
02214 GLAPI PFNGLVERTEXATTRIB2FVPROC glad_glVertexAttrib2fv;
02215 #define glVertexAttrib2fv glad_glVertexAttrib2fv
02216 typedef void (APIENTRY PFNGLVERTEXATTRIB2SPROC) (GLuint index, GLshort x, GLshort y);
02217 GLAPI PFNGLVERTEXATTRIB2SPROC glad_glVertexAttrib2s;
02218 #define glVertexAttrib2s glad_glVertexAttrib2s
02219 typedef void (APIENTRY PFNGLVERTEXATTRIB2SVPROC) (GLuint index, const GLshort *v);
02220 GLAPI PFNGLVERTEXATTRIB2SVPROC glad_glVertexAttrib2sv;
02221 #define glVertexAttrib2sv glad_glVertexAttrib2sv
02222 typedef void (APIENTRY PFNGLVERTEXATTRIB3DPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble z);
02223 GLAPI PFNGLVERTEXATTRIB3DPROC glad_glVertexAttrib3d;
02224 #define glVertexAttrib3d glad_glVertexAttrib3d
02225 typedef void (APIENTRY PFNGLVERTEXATTRIB3DVPROC) (GLuint index, const GLdouble *v);
02226 GLAPI PFNGLVERTEXATTRIB3DVPROC glad_glVertexAttrib3dv;
02227 #define glVertexAttrib3dv glad_glVertexAttrib3dv
02228 typedef void (APIENTRY PFNGLVERTEXATTRIB3FPROC) (GLuint index, GLfloat x, GLfloat y, GLfloat z);
02229 GLAPI PFNGLVERTEXATTRIB3FPROC glad_glVertexAttrib3f;
02230 #define glVertexAttrib3f glad_glVertexAttrib3f
02231 typedef void (APIENTRY PFNGLVERTEXATTRIB3FVPROC) (GLuint index, const GLfloat *v);
02232 GLAPI PFNGLVERTEXATTRIB3FVPROC glad_glVertexAttrib3fv;
02233 #define glVertexAttrib3fv glad_glVertexAttrib3fv
02234 typedef void (APIENTRY PFNGLVERTEXATTRIB3SPROC) (GLuint index, GLshort x, GLshort y, GLshort z);
02235 GLAPI PFNGLVERTEXATTRIB3SPROC glad_glVertexAttrib3s;
02236 #define glVertexAttrib3s glad_glVertexAttrib3s
02237 typedef void (APIENTRY PFNGLVERTEXATTRIB3SVPROC) (GLuint index, const GLshort *v);
02238 GLAPI PFNGLVERTEXATTRIB3SVPROC glad_glVertexAttrib3sv;
02239 #define glVertexAttrib3sv glad_glVertexAttrib3sv
```



```

02240 typedef void (APIENTRY PFNGLVERTEXATTRIB4NBVPROC) (GLuint index, const GLbyte *v);
02241 GLAPI PFNGLVERTEXATTRIB4NBVPROC glad_glVertexAttrib4Nbv;
02242 #define glVertexAttrib4Nbv glad_glVertexAttrib4Nbv
02243 typedef void (APIENTRY PFNGLVERTEXATTRIB4NIVPROC) (GLuint index, const GLint *v);
02244 GLAPI PFNGLVERTEXATTRIB4NIVPROC glad_glVertexAttrib4Niv;
02245 #define glVertexAttrib4Niv glad_glVertexAttrib4Niv
02246 typedef void (APIENTRY PFNGLVERTEXATTRIB4NSVPROC) (GLuint index, const GLshort *v);
02247 GLAPI PFNGLVERTEXATTRIB4NSVPROC glad_glVertexAttrib4Nsv;
02248 #define glVertexAttrib4Nsv glad_glVertexAttrib4Nsv
02249 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUBPROC) (GLuint index, GLubyte x, GLubyte y, GLubyte z,
GLubyte w);
02250 GLAPI PFNGLVERTEXATTRIB4NUBPROC glad_glVertexAttrib4Nub;
02251 #define glVertexAttrib4Nub glad_glVertexAttrib4Nub
02252 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUBPROC) (GLuint index, const GLubyte *v);
02253 GLAPI PFNGLVERTEXATTRIB4NUBPROC glad_glVertexAttrib4Nubv;
02254 #define glVertexAttrib4Nubv glad_glVertexAttrib4Nubv
02255 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUIVPROC) (GLuint index, const GLuint *v);
02256 GLAPI PFNGLVERTEXATTRIB4NUIVPROC glad_glVertexAttrib4Nuiv;
02257 #define glVertexAttrib4Nuiv glad_glVertexAttrib4Nuiv
02258 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUSVPROC) (GLuint index, const GLushort *v);
02259 GLAPI PFNGLVERTEXATTRIB4NUSVPROC glad_glVertexAttrib4Nusv;
02260 #define glVertexAttrib4Nusv glad_glVertexAttrib4Nusv
02261 typedef void (APIENTRY PFNGLVERTEXATTRIB4BVPROC) (GLuint index, const GLbyte *v);
02262 GLAPI PFNGLVERTEXATTRIB4BVPROC glad_glVertexAttrib4bv;
02263 #define glVertexAttrib4bv glad_glVertexAttrib4bv
02264 typedef void (APIENTRY PFNGLVERTEXATTRIB4DPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble z,
GLdouble w);
02265 GLAPI PFNGLVERTEXATTRIB4DPROC glad_glVertexAttrib4d;
02266 #define glVertexAttrib4d glad_glVertexAttrib4d
02267 typedef void (APIENTRY PFNGLVERTEXATTRIB4DVPROC) (GLuint index, const GLdouble *v);
02268 GLAPI PFNGLVERTEXATTRIB4DVPROC glad_glVertexAttrib4dv;
02269 #define glVertexAttrib4dv glad_glVertexAttrib4dv
02270 typedef void (APIENTRY PFNGLVERTEXATTRIB4FPROC) (GLuint index, GLfloat x, GLfloat y, GLfloat z,
GLfloat w);
02271 GLAPI PFNGLVERTEXATTRIB4FPROC glad_glVertexAttrib4f;
02272 #define glVertexAttrib4f glad_glVertexAttrib4f
02273 typedef void (APIENTRY PFNGLVERTEXATTRIB4FVPROC) (GLuint index, const GLfloat *v);
02274 GLAPI PFNGLVERTEXATTRIB4FVPROC glad_glVertexAttrib4fv;
02275 #define glVertexAttrib4fv glad_glVertexAttrib4fv
02276 typedef void (APIENTRY PFNGLVERTEXATTRIB4IVPROC) (GLuint index, const GLint *v);
02277 GLAPI PFNGLVERTEXATTRIB4IVPROC glad_glVertexAttrib4iv;
02278 #define glVertexAttrib4iv glad_glVertexAttrib4iv
02279 typedef void (APIENTRY PFNGLVERTEXATTRIB4SPROC) (GLuint index, GLshort x, GLshort y, GLshort z,
GLshort w);
02280 GLAPI PFNGLVERTEXATTRIB4SPROC glad_glVertexAttrib4s;
02281 #define glVertexAttrib4s glad_glVertexAttrib4s
02282 typedef void (APIENTRY PFNGLVERTEXATTRIB4SVPROC) (GLuint index, const GLshort *v);
02283 GLAPI PFNGLVERTEXATTRIB4SVPROC glad_glVertexAttrib4sv;
02284 #define glVertexAttrib4sv glad_glVertexAttrib4sv
02285 typedef void (APIENTRY PFNGLVERTEXATTRIB4UBVPROC) (GLuint index, const GLubyte *v);
02286 GLAPI PFNGLVERTEXATTRIB4UBVPROC glad_glVertexAttrib4ubv;
02287 #define glVertexAttrib4ubv glad_glVertexAttrib4ubv
02288 typedef void (APIENTRY PFNGLVERTEXATTRIB4UIVPROC) (GLuint index, const GLuint *v);
02289 GLAPI PFNGLVERTEXATTRIB4UIVPROC glad_glVertexAttrib4uiv;
02290 #define glVertexAttrib4uiv glad_glVertexAttrib4uiv
02291 typedef void (APIENTRY PFNGLVERTEXATTRIB4USVPROC) (GLuint index, const GLushort *v);
02292 GLAPI PFNGLVERTEXATTRIB4USVPROC glad_glVertexAttrib4usv;
02293 #define glVertexAttrib4usv glad_glVertexAttrib4usv
02294 typedef void (APIENTRY PFNGLVERTEXATTRIBPOINTERPROC) (GLuint index, GLint size, GLenum type, GLboolean
normalized, GLsizei stride, const void *pointer);
02295 GLAPI PFNGLVERTEXATTRIBPOINTERPROC glad_glVertexAttribPointer;
02296 #define glVertexAttribPointer glad_glVertexAttribPointer
02297 #endif
02298 #ifndef GL_VERSION_2_1
02299 #define GL_VERSION_2_1 1
02300 GLAPI int GLAD_GL_VERSION_2_1;
02301 typedef void (APIENTRY PFNGLUNIFORMMATRIX2X3FVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
02302 GLAPI PFNGLUNIFORMMATRIX2X3FVPROC glad_glUniformMatrix2x3fv;
02303 #define glUniformMatrix2x3fv glad_glUniformMatrix2x3fv
02304 typedef void (APIENTRY PFNGLUNIFORMMATRIX3X2FVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
02305 GLAPI PFNGLUNIFORMMATRIX3X2FVPROC glad_glUniformMatrix3x2fv;
02306 #define glUniformMatrix3x2fv glad_glUniformMatrix3x2fv
02307 typedef void (APIENTRY PFNGLUNIFORMMATRIX2X4FVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
02308 GLAPI PFNGLUNIFORMMATRIX2X4FVPROC glad_glUniformMatrix2x4fv;
02309 #define glUniformMatrix2x4fv glad_glUniformMatrix2x4fv
02310 typedef void (APIENTRY PFNGLUNIFORMMATRIX4X2FVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
02311 GLAPI PFNGLUNIFORMMATRIX4X2FVPROC glad_glUniformMatrix4x2fv;
02312 #define glUniformMatrix4x2fv glad_glUniformMatrix4x2fv
02313 typedef void (APIENTRY PFNGLUNIFORMMATRIX3X4FVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
02314 GLAPI PFNGLUNIFORMMATRIX3X4FVPROC glad_glUniformMatrix3x4fv;
02315 #define glUniformMatrix3x4fv glad_glUniformMatrix3x4fv
02316 typedef void (APIENTRY PFNGLUNIFORMMATRIX4X3FVPROC) (GLint location, GLsizei count, GLboolean

```

```

        transpose, const GLfloat *value);
02317 GLAPI PFNGLUNIFORMMATRIX4X3FVPROC glad_glUniformMatrix4x3fv;
02318 #define glUniformMatrix4x3fv glad_glUniformMatrix4x3fv
02319 #endif
02320 #ifndef GL_VERSION_3_0
02321 #define GL_VERSION_3_0 1
02322 GLAPI int GLAD_GL_VERSION_3_0;
02323 typedef void (APIENTRY PFNGLCOLORMASKIPROC) (GLuint index, GLboolean r, GLboolean g, GLboolean b,
GLboolean a);
02324 GLAPI PFNGLCOLORMASKIPROC glad_glColorMaski;
02325 #define glColorMaski glad_glColorMaski
02326 typedef void (APIENTRY PFNGLGETBOOLEANI_VPROC) (GLenum target, GLuint index, GLboolean *data);
02327 GLAPI PFNGLGETBOOLEANI_VPROC glad_glGetBooleani_v;
02328 #define glGetBooleani_v glad_glGetBooleani_v
02329 typedef void (APIENTRY PFNGLGETINTEGERI_VPROC) (GLenum target, GLuint index, GLint *data);
02330 GLAPI PFNGLGETINTEGERI_VPROC glad_glGetIntegeri_v;
02331 #define glGetIntegeri_v glad_glGetIntegeri_v
02332 typedef void (APIENTRY PFNGLENABLEIPROC) (GLenum target, GLuint index);
02333 GLAPI PFNGLENABLEIPROC glad_glEnablei;
02334 #define glEnablei glad_glEnablei
02335 typedef void (APIENTRY PFNGLDISABLEIPROC) (GLenum target, GLuint index);
02336 GLAPI PFNGLDISABLEIPROC glad_glDisablei;
02337 #define glDisablei glad_glDisablei
02338 typedef GLboolean (APIENTRY PFNGLISENABLEDIPROC) (GLenum target, GLuint index);
02339 GLAPI PFNGLISENABLEDIPROC glad_glIsEnabledi;
02340 #define glIsEnabledi glad_glIsEnabledi
02341 typedef void (APIENTRY PFNGLBEGINTRANSFORMFEEDBACKPROC) (GLenum primitiveMode);
02342 GLAPI PFNGLBEGINTRANSFORMFEEDBACKPROC glad_glBeginTransformFeedback;
02343 #define glBeginTransformFeedback glad_glBeginTransformFeedback
02344 typedef void (APIENTRY PFNGLENDTRANSFORMFEEDBACKPROC) (void);
02345 GLAPI PFNGLENDTRANSFORMFEEDBACKPROC glad_glEndTransformFeedback;
02346 #define glEndTransformFeedback glad_glEndTransformFeedback
02347 typedef void (APIENTRY PFNGLBINDBUFFERRANGEPROC) (GLenum target, GLuint index, GLuint buffer, GLintptr
offset, GLsizeiptr size);
02348 GLAPI PFNGLBINDBUFFERRANGEPROC glad_glBindBufferRange;
02349 #define glBindBufferRange glad_glBindBufferRange
02350 typedef void (APIENTRY PFNGLBINDBUFFERBASEPROC) (GLenum target, GLuint index, GLuint buffer);
02351 GLAPI PFNGLBINDBUFFERBASEPROC glad_glBindBufferBase;
02352 #define glBindBufferBase glad_glBindBufferBase
02353 typedef void (APIENTRY PFNGLTRANSFORMFEEDBACKVARYINGSPROC) (GLuint program, GLsizei count, const
GLchar *const*varyings, GLenum bufferMode);
02354 GLAPI PFNGLTRANSFORMFEEDBACKVARYINGSPROC glad_glTransformFeedbackVaryings;
02355 #define glTransformFeedbackVaryings glad_glTransformFeedbackVaryings
02356 typedef void (APIENTRY PFNGLGETTRANSFORMFEEDBACKVARYINGPROC) (GLuint program, GLuint index, GLsizei
bufSize, GLsizei *length, GLsizei *size, GLenum *type, GLchar *name);
02357 GLAPI PFNGLGETTRANSFORMFEEDBACKVARYINGPROC glad_glGetTransformFeedbackVarying;
02358 #define glGetTransformFeedbackVarying glad_glGetTransformFeedbackVarying
02359 typedef void (APIENTRY PFNGLCLAMPCOLORPROC) (GLenum target, GLenum clamp);
02360 GLAPI PFNGLCLAMPCOLORPROC glad_glClampColor;
02361 #define glClampColor glad_glClampColor
02362 typedef void (APIENTRY PFNGLBEGINCONDITIONALRENDERPROC) (GLuint id, GLenum mode);
02363 GLAPI PFNGLBEGINCONDITIONALRENDERPROC glad_glBeginConditionalRender;
02364 #define glBeginConditionalRender glad_glBeginConditionalRender
02365 typedef void (APIENTRY PFNGLENDCONDITIONALRENDERPROC) (void);
02366 GLAPI PFNGLENDCONDITIONALRENDERPROC glad_glEndConditionalRender;
02367 #define glEndConditionalRender glad_glEndConditionalRender
02368 typedef void (APIENTRY PFNGLVERTEXATTRIBIPOINTERPROC) (GLuint index, GLint size, GLenum type, GLsizei
stride, const void *pointer);
02369 GLAPI PFNGLVERTEXATTRIBIPOINTERPROC glad_glVertexAttribIPointer;
02370 #define glVertexAttribIPointer glad_glVertexAttribIPointer
02371 typedef void (APIENTRY PFNGLGETVERTEXATTRIBI_VPROC) (GLuint index, GLenum pname, GLint *params);
02372 GLAPI PFNGLGETVERTEXATTRIBI_VPROC glad_glGetVertexAttribIiv;
02373 #define glGetVertexAttribIiv glad_glGetVertexAttribIiv
02374 typedef void (APIENTRY PFNGLGETVERTEXATTRIBIUIVPROC) (GLuint index, GLenum pname, GLuint *params);
02375 GLAPI PFNGLGETVERTEXATTRIBIUIVPROC glad_glGetVertexAttribIuiv;
02376 #define glGetVertexAttribIuiv glad_glGetVertexAttribIuiv
02377 typedef void (APIENTRY PFNGLVERTEXATTRIBI1PROC) (GLuint index, GLint x);
02378 GLAPI PFNGLVERTEXATTRIBI1PROC glad_glVertexAttribI1i;
02379 #define glVertexAttribI1i glad_glVertexAttribI1i
02380 typedef void (APIENTRY PFNGLVERTEXATTRIBI2IPROC) (GLuint index, GLint x, GLint y);
02381 GLAPI PFNGLVERTEXATTRIBI2IPROC glad_glVertexAttribI2i;
02382 #define glVertexAttribI2i glad_glVertexAttribI2i
02383 typedef void (APIENTRY PFNGLVERTEXATTRIBI3IPROC) (GLuint index, GLint x, GLint y, GLint z);
02384 GLAPI PFNGLVERTEXATTRIBI3IPROC glad_glVertexAttribI3i;
02385 #define glVertexAttribI3i glad_glVertexAttribI3i
02386 typedef void (APIENTRY PFNGLVERTEXATTRIBI4IPROC) (GLuint index, GLint x, GLint y, GLint z, GLint w);
02387 GLAPI PFNGLVERTEXATTRIBI4IPROC glad_glVertexAttribI4i;
02388 #define glVertexAttribI4i glad_glVertexAttribI4i
02389 typedef void (APIENTRY PFNGLVERTEXATTRIBI1UIPROC) (GLuint index, GLuint x);
02390 GLAPI PFNGLVERTEXATTRIBI1UIPROC glad_glVertexAttribI1ui;
02391 #define glVertexAttribI1ui glad_glVertexAttribI1ui
02392 typedef void (APIENTRY PFNGLVERTEXATTRIBI2UIPROC) (GLuint index, GLuint x, GLuint y);
02393 GLAPI PFNGLVERTEXATTRIBI2UIPROC glad_glVertexAttribI2ui;
02394 #define glVertexAttribI2ui glad_glVertexAttribI2ui
02395 typedef void (APIENTRY PFNGLVERTEXATTRIBI3UIPROC) (GLuint index, GLuint x, GLuint y, GLuint z);
02396 GLAPI PFNGLVERTEXATTRIBI3UIPROC glad_glVertexAttribI3ui;
02397 #define glVertexAttribI3ui glad_glVertexAttribI3ui

```

```

02398 typedef void (APIENTRY PFNGLVERTEXATTRIBI4UIPROC) (GLuint index, GLuint x, GLuint y, GLuint z, GLuint
    w);
02399 GLAPI PFNGLVERTEXATTRIBI4UIPROC glad_glVertexAttribI4ui;
02400 #define glVertexAttribI4ui glad_glVertexAttribI4ui
02401 typedef void (APIENTRY PFNGLVERTEXATTRIBI1IIVPROC) (GLuint index, const GLint *v);
02402 GLAPI PFNGLVERTEXATTRIBI1IIVPROC glad_glVertexAttribI1iv;
02403 #define glVertexAttribI1iv glad_glVertexAttribI1iv
02404 typedef void (APIENTRY PFNGLVERTEXATTRIBI2IIVPROC) (GLuint index, const GLint *v);
02405 GLAPI PFNGLVERTEXATTRIBI2IIVPROC glad_glVertexAttribI2iv;
02406 #define glVertexAttribI2iv glad_glVertexAttribI2iv
02407 typedef void (APIENTRY PFNGLVERTEXATTRIBI3IIVPROC) (GLuint index, const GLint *v);
02408 GLAPI PFNGLVERTEXATTRIBI3IIVPROC glad_glVertexAttribI3iv;
02409 #define glVertexAttribI3iv glad_glVertexAttribI3iv
02410 typedef void (APIENTRY PFNGLVERTEXATTRIBI4IIVPROC) (GLuint index, const GLint *v);
02411 GLAPI PFNGLVERTEXATTRIBI4IIVPROC glad_glVertexAttribI4iv;
02412 #define glVertexAttribI4iv glad_glVertexAttribI4iv
02413 typedef void (APIENTRY PFNGLVERTEXATTRIBI1UIVPROC) (GLuint index, const GLuint *v);
02414 GLAPI PFNGLVERTEXATTRIBI1UIVPROC glad_glVertexAttribI1uiv;
02415 #define glVertexAttribI1uiv glad_glVertexAttribI1uiv
02416 typedef void (APIENTRY PFNGLVERTEXATTRIBI2UIVPROC) (GLuint index, const GLuint *v);
02417 GLAPI PFNGLVERTEXATTRIBI2UIVPROC glad_glVertexAttribI2uiv;
02418 #define glVertexAttribI2uiv glad_glVertexAttribI2uiv
02419 typedef void (APIENTRY PFNGLVERTEXATTRIBI3UIVPROC) (GLuint index, const GLuint *v);
02420 GLAPI PFNGLVERTEXATTRIBI3UIVPROC glad_glVertexAttribI3uiv;
02421 #define glVertexAttribI3uiv glad_glVertexAttribI3uiv
02422 typedef void (APIENTRY PFNGLVERTEXATTRIBI4UIVPROC) (GLuint index, const GLuint *v);
02423 GLAPI PFNGLVERTEXATTRIBI4UIVPROC glad_glVertexAttribI4uiv;
02424 #define glVertexAttribI4uiv glad_glVertexAttribI4uiv
02425 typedef void (APIENTRY PFNGLVERTEXATTRIBI4BIVPROC) (GLuint index, const GLbyte *v);
02426 GLAPI PFNGLVERTEXATTRIBI4BIVPROC glad_glVertexAttribI4bv;
02427 #define glVertexAttribI4bv glad_glVertexAttribI4bv
02428 typedef void (APIENTRY PFNGLVERTEXATTRIBI4SVPROC) (GLuint index, const GLshort *v);
02429 GLAPI PFNGLVERTEXATTRIBI4SVPROC glad_glVertexAttribI4sv;
02430 #define glVertexAttribI4sv glad_glVertexAttribI4sv
02431 typedef void (APIENTRY PFNGLVERTEXATTRIBI4UBVPROC) (GLuint index, const GLubyte *v);
02432 GLAPI PFNGLVERTEXATTRIBI4UBVPROC glad_glVertexAttribI4ubv;
02433 #define glVertexAttribI4ubv glad_glVertexAttribI4ubv
02434 typedef void (APIENTRY PFNGLVERTEXATTRIBI4USVPROC) (GLuint index, const GLushort *v);
02435 GLAPI PFNGLVERTEXATTRIBI4USVPROC glad_glVertexAttribI4usv;
02436 #define glVertexAttribI4usv glad_glVertexAttribI4usv
02437 typedef void (APIENTRY PFNGLGETUNIFORMUIVPROC) (GLuint program, GLint location, GLuint *params);
02438 GLAPI PFNGLGETUNIFORMUIVPROC glad_glGetUniformuiv;
02439 #define glGetUniformuiv glad_glGetUniformuiv
02440 typedef void (APIENTRY PFNGLBINDFRAGDATALOCATIONPROC) (GLuint program, GLuint color, const GLchar
    *name);
02441 GLAPI PFNGLBINDFRAGDATALOCATIONPROC glad_glBindFragDataLocation;
02442 #define glBindFragDataLocation glad_glBindFragDataLocation
02443 typedef GLint (APIENTRY PFNGLGETFRAGDATALOCATIONPROC) (GLuint program, const GLchar *name);
02444 GLAPI PFNGLGETFRAGDATALOCATIONPROC glad_glGetFragDataLocation;
02445 #define glGetFragDataLocation glad_glGetFragDataLocation
02446 typedef void (APIENTRY PFNGLUNIFORM1UIPROC) (GLint location, GLuint v0);
02447 GLAPI PFNGLUNIFORM1UIPROC glad_glUniform1ui;
02448 #define glUniform1ui glad_glUniform1ui
02449 typedef void (APIENTRY PFNGLUNIFORM2UIPROC) (GLint location, GLuint v0, GLuint v1);
02450 GLAPI PFNGLUNIFORM2UIPROC glad_glUniform2ui;
02451 #define glUniform2ui glad_glUniform2ui
02452 typedef void (APIENTRY PFNGLUNIFORM3UIPROC) (GLint location, GLuint v0, GLuint v1, GLuint v2);
02453 GLAPI PFNGLUNIFORM3UIPROC glad_glUniform3ui;
02454 #define glUniform3ui glad_glUniform3ui
02455 typedef void (APIENTRY PFNGLUNIFORM4UIPROC) (GLint location, GLuint v0, GLuint v1, GLuint v2, GLuint
    v3);
02456 GLAPI PFNGLUNIFORM4UIPROC glad_glUniform4ui;
02457 #define glUniform4ui glad_glUniform4ui
02458 typedef void (APIENTRY PFNGLUNIFORM1UIVPROC) (GLint location, GLsizei count, const GLuint *value);
02459 GLAPI PFNGLUNIFORM1UIVPROC glad_glUniform1uiv;
02460 #define glUniform1uiv glad_glUniform1uiv
02461 typedef void (APIENTRY PFNGLUNIFORM2UIVPROC) (GLint location, GLsizei count, const GLuint *value);
02462 GLAPI PFNGLUNIFORM2UIVPROC glad_glUniform2uiv;
02463 #define glUniform2uiv glad_glUniform2uiv
02464 typedef void (APIENTRY PFNGLUNIFORM3UIVPROC) (GLint location, GLsizei count, const GLuint *value);
02465 GLAPI PFNGLUNIFORM3UIVPROC glad_glUniform3uiv;
02466 #define glUniform3uiv glad_glUniform3uiv
02467 typedef void (APIENTRY PFNGLUNIFORM4UIVPROC) (GLint location, GLsizei count, const GLuint *value);
02468 GLAPI PFNGLUNIFORM4UIVPROC glad_glUniform4uiv;
02469 #define glUniform4uiv glad_glUniform4uiv
02470 typedef void (APIENTRY PFNGLTEXPARAMETERIIVPROC) (GLenum target, GLenum pname, const GLint *params);
02471 GLAPI PFNGLTEXPARAMETERIIVPROC glad_glTexParameterIiv;
02472 #define glTexParameterIiv glad_glTexParameterIiv
02473 typedef void (APIENTRY PFNGLTEXPARAMETERIUIVPROC) (GLenum target, GLenum pname, const GLuint *params);
02474 GLAPI PFNGLTEXPARAMETERIUIVPROC glad_glTexParameterIuiv;
02475 #define glTexParameterIuiv glad_glTexParameterIuiv
02476 typedef void (APIENTRY PFNGLGETTEXPARAMETERIIVPROC) (GLenum target, GLenum pname, GLint *params);
02477 GLAPI PFNGLGETTEXPARAMETERIIVPROC glad_glGetTexParameterIiv;
02478 #define glGetTexParameterIiv glad_glGetTexParameterIiv
02479 typedef void (APIENTRY PFNGLGETTEXPARAMETERIUIVPROC) (GLenum target, GLenum pname, GLuint *params);
02480 GLAPI PFNGLGETTEXPARAMETERIUIVPROC glad_glGetTexParameterIuiv;
02481 #define glGetTexParameterIuiv glad_glGetTexParameterIuiv

```

```

02482 typedef void (APIENTRY PFNGLCLEARBUFFERIVPROC)(GLenum buffer, GLint drawbuffer, const GLint *value);
02483 GLAPI PFNGLCLEARBUFFERIVPROC glad_glClearBufferiv;
02484 #define glClearBufferiv glad_glClearBufferiv
02485 typedef void (APIENTRY PFNGLCLEARBUFFERUIVPROC)(GLenum buffer, GLint drawbuffer, const GLuint
    *value);
02486 GLAPI PFNGLCLEARBUFFERUIVPROC glad_glClearBufferuiv;
02487 #define glClearBufferuiv glad_glClearBufferuiv
02488 typedef void (APIENTRY PFNGLCLEARBUFFERFVPROC)(GLenum buffer, GLint drawbuffer, const GLfloat
    *value);
02489 GLAPI PFNGLCLEARBUFFERFVPROC glad_glClearBufferfv;
02490 #define glClearBufferfv glad_glClearBufferfv
02491 typedef void (APIENTRY PFNGLCLEARBUFFERFIPROC)(GLenum buffer, GLint drawbuffer, GLfloat depth, GLint
    stencil);
02492 GLAPI PFNGLCLEARBUFFERFIPROC glad_glClearBufferfi;
02493 #define glClearBufferfi glad_glClearBufferfi
02494 typedef const GLubyte * (APIENTRY PFNGLGETSTRINGIPROC)(GLenum name, GLuint index);
02495 GLAPI PFNGLGETSTRINGIPROC glad_glGetStringi;
02496 #define glGetStringi glad_glGetStringi
02497 typedef GLboolean (APIENTRY PFNGLISRENDERBUFFERPROC)(GLuint renderbuffer);
02498 GLAPI PFNGLISRENDERBUFFERPROC glad_glIsRenderbuffer;
02499 #define glIsRenderbuffer glad_glIsRenderbuffer
02500 typedef void (APIENTRY PFNGLBINDRENDERBUFFERPROC)(GLenum target, GLuint renderbuffer);
02501 GLAPI PFNGLBINDRENDERBUFFERPROC glad_glBindRenderbuffer;
02502 #define glBindRenderbuffer glad_glBindRenderbuffer
02503 typedef void (APIENTRY PFNGLDELETERENDERBUFFERSPROC)(GLsizei n, const GLuint *renderbuffers);
02504 GLAPI PFNGLDELETERENDERBUFFERSPROC glad_glDeleteRenderbuffers;
02505 #define glDeleteRenderbuffers glad_glDeleteRenderbuffers
02506 typedef void (APIENTRY PFNGLGENRENDERBUFFERSPROC)(GLsizei n, GLuint *renderbuffers);
02507 GLAPI PFNGLGENRENDERBUFFERSPROC glad_glGenRenderbuffers;
02508 #define glGenRenderbuffers glad_glGenRenderbuffers
02509 typedef void (APIENTRY PFNGLRENDERBUFFERSTORAGEPROC)(GLenum target, GLenum internalformat, GLsizei
    width, GLsizei height);
02510 GLAPI PFNGLRENDERBUFFERSTORAGEPROC glad_glRenderbufferStorage;
02511 #define glRenderbufferStorage glad_glRenderbufferStorage
02512 typedef void (APIENTRY PFNGLGETRENDERBUFFERPARAMETERIVPROC)(GLenum target, GLenum pname, GLint
    *params);
02513 GLAPI PFNGLGETRENDERBUFFERPARAMETERIVPROC glad_glGetRenderbufferParameteriv;
02514 #define glGetRenderbufferParameteriv glad_glGetRenderbufferParameteriv
02515 typedef GLboolean (APIENTRY PFNGLISFRAMEBUFFERPROC)(GLuint framebuffer);
02516 GLAPI PFNGLISFRAMEBUFFERPROC glad_glIsFramebuffer;
02517 #define glIsFramebuffer glad_glIsFramebuffer
02518 typedef void (APIENTRY PFNGLBINDFRAMEBUFFERPROC)(GLenum target, GLuint framebuffer);
02519 GLAPI PFNGLBINDFRAMEBUFFERPROC glad_glBindFramebuffer;
02520 #define glBindFramebuffer glad_glBindFramebuffer
02521 typedef void (APIENTRY PFNGLDELETEFRAMEBUFFERSPROC)(GLsizei n, const GLuint *framebuffers);
02522 GLAPI PFNGLDELETEFRAMEBUFFERSPROC glad_glDeleteFramebuffers;
02523 #define glDeleteFramebuffers glad_glDeleteFramebuffers
02524 typedef void (APIENTRY PFNGLGENFRAMEBUFFERSPROC)(GLsizei n, GLuint *framebuffers);
02525 GLAPI PFNGLGENFRAMEBUFFERSPROC glad_glGenFramebuffers;
02526 #define glGenFramebuffers glad_glGenFramebuffers
02527 typedef GLenum (APIENTRY PFNGLCHECKFRAMEBUFFERSTATUSPROC)(GLenum target);
02528 GLAPI PFNGLCHECKFRAMEBUFFERSTATUSPROC glad_glCheckFramebufferStatus;
02529 #define glCheckFramebufferStatus glad_glCheckFramebufferStatus
02530 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURE1DPROC)(GLenum target, GLenum attachment, GLenum
    textarget, GLuint texture, GLint level);
02531 GLAPI PFNGLFRAMEBUFFERTEXTURE1DPROC glad_glFramebufferTexture1D;
02532 #define glFramebufferTexture1D glad_glFramebufferTexture1D
02533 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURE2DPROC)(GLenum target, GLenum attachment, GLenum
    textarget, GLuint texture, GLint level);
02534 GLAPI PFNGLFRAMEBUFFERTEXTURE2DPROC glad_glFramebufferTexture2D;
02535 #define glFramebufferTexture2D glad_glFramebufferTexture2D
02536 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURE3DPROC)(GLenum target, GLenum attachment, GLenum
    textarget, GLuint texture, GLint level, GLint zoffset);
02537 GLAPI PFNGLFRAMEBUFFERTEXTURE3DPROC glad_glFramebufferTexture3D;
02538 #define glFramebufferTexture3D glad_glFramebufferTexture3D
02539 typedef void (APIENTRY PFNGLFRAMEBUFFERRENDERBUFFERPROC)(GLenum target, GLenum attachment, GLenum
    renderbuffertarget, GLuint renderbuffer);
02540 GLAPI PFNGLFRAMEBUFFERRENDERBUFFERPROC glad_glFramebufferRenderbuffer;
02541 #define glFramebufferRenderbuffer glad_glFramebufferRenderbuffer
02542 typedef void (APIENTRY PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVPROC)(GLenum target, GLenum
    attachment, GLenum pname, GLint *params);
02543 GLAPI PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVPROC glad_glGetFramebufferAttachmentParameteriv;
02544 #define glGetFramebufferAttachmentParameteriv glad_glGetFramebufferAttachmentParameteriv
02545 typedef void (APIENTRY PFNGLGENERATEMIPMAPPROC)(GLenum target);
02546 GLAPI PFNGLGENERATEMIPMAPPROC glad_glGenerateMipmap;
02547 #define glGenerateMipmap glad_glGenerateMipmap
02548 typedef void (APIENTRY PFNGLBLITFRAMEBUFFERPROC)(GLint srcX0, GLint srcY0, GLint srcX1, GLint srcY1,
    GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLbitfield mask, GLenum filter);
02549 GLAPI PFNGLBLITFRAMEBUFFERPROC glad_glBlitFramebuffer;
02550 #define glBlitFramebuffer glad_glBlitFramebuffer
02551 typedef void (APIENTRY PFNGLRENDERBUFFERSTAGEMULTISAMPLEPROC)(GLenum target, GLsizei samples,
    GLenum internalformat, GLsizei width, GLsizei height);
02552 GLAPI PFNGLRENDERBUFFERSTAGEMULTISAMPLEPROC glad_glRenderbufferStorageMultisample;
02553 #define glRenderbufferStorageMultisample glad_glRenderbufferStorageMultisample
02554 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURELAYERPROC)(GLenum target, GLenum attachment, GLuint
    texture, GLint level, GLint layer);
02555 GLAPI PFNGLFRAMEBUFFERTEXTURELAYERPROC glad_glFramebufferTextureLayer;

```



```

02556 #define glFramebufferTextureLayer glad_glFramebufferTextureLayer
02557 typedef void * (APIENTRY PFNGLMAPBUFFERRANGEPROC) (GLenum target, GLintptr offset, GLsizeiptr length,
GLbitfield access);
02558 GLAPI PFNGLMAPBUFFERRANGEPROC glad_glMapBufferRange;
02559 #define glMapBufferRange glad_glMapBufferRange
02560 typedef void (APIENTRY PFNGLFLUSHMAPPEDBUFFERRANGEPROC) (GLenum target, GLintptr offset, GLsizeiptr
length);
02561 GLAPI PFNGLFLUSHMAPPEDBUFFERRANGEPROC glad_glFlushMappedBufferRange;
02562 #define glFlushMappedBufferRange glad_glFlushMappedBufferRange
02563 typedef void (APIENTRY PFNGLBINDVERTEXARRAYPROC) (GLuint array);
02564 GLAPI PFNGLBINDVERTEXARRAYPROC glad_glBindVertexArray;
02565 #define glBindVertexArray glad_glBindVertexArray
02566 typedef void (APIENTRY PFNGLDELETEVERTEXARRAYSPROC) (GLsizei n, const GLuint *arrays);
02567 GLAPI PFNGLDELETEVERTEXARRAYSPROC glad_glDeleteVertexArrays;
02568 #define glDeleteVertexArrays glad_glDeleteVertexArrays
02569 typedef void (APIENTRY PFNGLGENVERTEXARRAYSPROC) (GLsizei n, GLuint *arrays);
02570 GLAPI PFNGLGENVERTEXARRAYSPROC glad_glGenVertexArrays;
02571 #define glGenVertexArrays glad_glGenVertexArrays
02572 typedef GLboolean (APIENTRY PFNGLISVERTEXARRAYPROC) (GLuint array);
02573 GLAPI PFNGLISVERTEXARRAYPROC glad_glIsVertexArray;
02574 #define glIsVertexArray glad_glIsVertexArray
02575 #endif
02576 #ifndef GL_VERSION_3_1
02577 #define GL_VERSION_3_1 1
02578 GLAPI int GLAD_GL_VERSION_3_1;
02579 typedef void (APIENTRY PFNGLDRAWARRAYSINSTANCEDPROC) (GLenum mode, GLint first, GLsizei count, GLsizei
instancecount);
02580 GLAPI PFNGLDRAWARRAYSINSTANCEDPROC glad_glDrawArraysInstanced;
02581 #define glDrawArraysInstanced glad_glDrawArraysInstanced
02582 typedef void (APIENTRY PFNGLDRAWELEMENTSINSTANCEDPROC) (GLenum mode, GLsizei count, GLenum type, const
void *indices, GLsizei instancecount);
02583 GLAPI PFNGLDRAWELEMENTSINSTANCEDPROC glad_glDrawElementsInstanced;
02584 #define glDrawElementsInstanced glad_glDrawElementsInstanced
02585 typedef void (APIENTRY PFNGLTEXBUFFERPROC) (GLenum target, GLenum internalformat, GLuint buffer);
02586 GLAPI PFNGLTEXBUFFERPROC glad_glTexBuffer;
02587 #define glTexBuffer glad_glTexBuffer
02588 typedef void (APIENTRY PFNGLPRIMITIVEVERESTARTINDEXPROC) (GLuint index);
02589 GLAPI PFNGLPRIMITIVEVERESTARTINDEXPROC glad_glPrimitiveRestartIndex;
02590 #define glPrimitiveRestartIndex glad_glPrimitiveRestartIndex
02591 typedef void (APIENTRY PFNGLCOPYBUFFERSUBDATAPROC) (GLenum readTarget, GLenum writeTarget, GLintptr
readOffset, GLintptr writeOffset, GLsizeiptr size);
02592 GLAPI PFNGLCOPYBUFFERSUBDATAPROC glad_glCopyBufferSubData;
02593 #define glCopyBufferSubData glad_glCopyBufferSubData
02594 typedef void (APIENTRY PFNGLGETUNIFORMINDICESPROC) (GLuint program, GLsizei uniformCount, const GLchar
*const*uniformNames, GLuint *uniformIndices);
02595 GLAPI PFNGLGETUNIFORMINDICESPROC glad_glGetUniformIndices;
02596 #define glGetUniformIndices glad_glGetUniformIndices
02597 typedef void (APIENTRY PFNGLGETACTIVEUNIFORMSIVPROC) (GLuint program, GLsizei uniformCount, const
GLuint *uniformIndices, GLenum pname, GLint *params);
02598 GLAPI PFNGLGETACTIVEUNIFORMSIVPROC glad_glGetActiveUniformsiv;
02599 #define glGetActiveUniformsiv glad_glGetActiveUniformsiv
02600 typedef void (APIENTRY PFNGLGETACTIVEUNIFORMNAMEPROC) (GLuint program, GLuint uniformIndex, GLsizei
bufSize, GLsizei *length, GLchar *uniformName);
02601 GLAPI PFNGLGETACTIVEUNIFORMNAMEPROC glad_glGetActiveUniformName;
02602 #define glGetActiveUniformName glad_glGetActiveUniformName
02603 typedef GLuint (APIENTRY PFNGLGETUNIFORMBLOCKINDEXPROC) (GLuint program, const GLchar
*uniformBlockName);
02604 GLAPI PFNGLGETUNIFORMBLOCKINDEXPROC glad_glGetUniformBlockIndex;
02605 #define glGetUniformBlockIndex glad_glGetUniformBlockIndex
02606 typedef void (APIENTRY PFNGLGETACTIVEUNIFORMBLOCKIVPROC) (GLuint program, GLuint uniformBlockIndex,
GLenum pname, GLint *params);
02607 GLAPI PFNGLGETACTIVEUNIFORMBLOCKIVPROC glad_glGetActiveUniformBlockiv;
02608 #define glGetActiveUniformBlockiv glad_glGetActiveUniformBlockiv
02609 typedef void (APIENTRY PFNGLGETACTIVEUNIFORMBLOCKNAMEPROC) (GLuint program, GLuint uniformBlockIndex,
GLsizei bufSize, GLsizei *length, GLchar *uniformBlockName);
02610 GLAPI PFNGLGETACTIVEUNIFORMBLOCKNAMEPROC glad_glGetActiveUniformBlockName;
02611 #define glGetActiveUniformBlockName glad_glGetActiveUniformBlockName
02612 typedef void (APIENTRY PFNGLUNIFORMBLOCKBINDINGPROC) (GLuint program, GLuint uniformBlockIndex, GLuint
uniformBlockBinding);
02613 GLAPI PFNGLUNIFORMBLOCKBINDINGPROC glad_glUniformBlockBinding;
02614 #define glUniformBlockBinding glad_glUniformBlockBinding
02615 #endif
02616 #ifndef GL_VERSION_3_2
02617 #define GL_VERSION_3_2 1
02618 GLAPI int GLAD_GL_VERSION_3_2;
02619 typedef void (APIENTRY PFNGLDRAWELEMENTSBASEVERTEXPROC) (GLenum mode, GLsizei count, GLenum type,
const void *indices, GLint basevertex);
02620 GLAPI PFNGLDRAWELEMENTSBASEVERTEXPROC glad_glDrawElementsBaseVertex;
02621 #define glDrawElementsBaseVertex glad_glDrawElementsBaseVertex
02622 typedef void (APIENTRY PFNGLDRAWRANGEELEMENTSBASEVERTEXPROC) (GLenum mode, GLuint start, GLuint end,
GLsizei count, GLenum type, const void *indices, GLint basevertex);
02623 GLAPI PFNGLDRAWRANGEELEMENTSBASEVERTEXPROC glad_glDrawRangeElementsBaseVertex;
02624 #define glDrawRangeElementsBaseVertex glad_glDrawRangeElementsBaseVertex
02625 typedef void (APIENTRY PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXPROC) (GLenum mode, GLsizei count, GLenum
type, const void *indices, GLsizei instancecount, GLint basevertex);
02626 GLAPI PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXPROC glad_glDrawElementsInstancedBaseVertex;
02627 #define glDrawElementsInstancedBaseVertex glad_glDrawElementsInstancedBaseVertex

```

```

02628 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSBASEVERTEXPROC)(GLenum mode, const GLsizei *count,
    GLenum type, const void *const*indices, GLsizei drawcount, const GLint *basevertex);
02629 GLAPI PFNGLMULTIDRAWELEMENTSBASEVERTEXPROC glad_glMultiDrawElementsBaseVertex;
02630 #define glMultiDrawElementsBaseVertex glad_glMultiDrawElementsBaseVertex
02631 typedef void (APIENTRY PFNGLPROVOKINGVERTEXPROC)(GLenum mode);
02632 GLAPI PFNGLPROVOKINGVERTEXPROC glad_glProvokingVertex;
02633 #define glProvokingVertex glad_glProvokingVertex
02634 typedef GLsync (APIENTRY PFNGLFENCESYNCPROC)(GLenum condition, GLbitfield flags);
02635 GLAPI PFNGLFENCESYNCPROC glad_glFenceSync;
02636 #define glFenceSync glad_glFenceSync
02637 typedef GLboolean (APIENTRY PFNGLISSYNCPROC)(GLsync sync);
02638 GLAPI PFNGLISSYNCPROC glad_glIsSync;
02639 #define glIsSync glad_glIsSync
02640 typedef void (APIENTRY PFNGLDELETESYNCPROC)(GLsync sync);
02641 GLAPI PFNGLDELETESYNCPROC glad_glDeleteSync;
02642 #define glDeleteSync glad_glDeleteSync
02643 typedef GLenum (APIENTRY PFNGLCLIENTWAITSYNCPROC)(GLsync sync, GLbitfield flags, GLuint64 timeout);
02644 GLAPI PFNGLCLIENTWAITSYNCPROC glad_glClientWaitSync;
02645 #define glClientWaitSync glad_glClientWaitSync
02646 typedef void (APIENTRY PFNGLWAITSYNCPROC)(GLsync sync, GLbitfield flags, GLuint64 timeout);
02647 GLAPI PFNGLWAITSYNCPROC glad_glWaitSync;
02648 #define glWaitSync glad_glWaitSync
02649 typedef void (APIENTRY PFNGLGETINTEGER64VPROC)(GLenum pname, GLint64 *data);
02650 GLAPI PFNGLGETINTEGER64VPROC glad_glGetInteger64v;
02651 #define glGetInteger64v glad_glGetInteger64v
02652 typedef void (APIENTRY PFNGLGETSYNCPROC)(GLsync sync, GLenum pname, GLsizei count, GLsizei *length,
    GLint *values);
02653 GLAPI PFNGLGETSYNCPROC glad_glGetSynciv;
02654 #define glGetSynciv glad_glGetSynciv
02655 typedef void (APIENTRY PFNGLGETINTEGER64I_VPROC)(GLenum target, GLuint index, GLint64 *data);
02656 GLAPI PFNGLGETINTEGER64I_VPROC glad_glGetInteger64i_v;
02657 #define glGetInteger64i_v glad_glGetInteger64i_v
02658 typedef void (APIENTRY PFNGLGETBUFFERPARAMETERI64VPROC)(GLenum target, GLenum pname, GLint64
    *params);
02659 GLAPI PFNGLGETBUFFERPARAMETERI64VPROC glad_glGetBufferParameteri64v;
02660 #define glGetBufferParameteri64v glad_glGetBufferParameteri64v
02661 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTUREPROC)(GLenum target, GLenum attachment, GLuint texture,
    GLint level);
02662 GLAPI PFNGLFRAMEBUFFERTEXTUREPROC glad_glFramebufferTexture;
02663 #define glFramebufferTexture glad_glFramebufferTexture
02664 typedef void (APIENTRY PFNGLTEXIMAGE2DMULTISAMPLEPROC)(GLenum target, GLsizei samples, GLenum
    internalformat, GLsizei width, GLsizei height, GLboolean fixedsamplelocations);
02665 GLAPI PFNGLTEXIMAGE2DMULTISAMPLEPROC glad_glTexImage2DMultisample;
02666 #define glTexImage2DMultisample glad_glTexImage2DMultisample
02667 typedef void (APIENTRY PFNGLTEXIMAGE3DMULTISAMPLEPROC)(GLenum target, GLsizei samples, GLenum
    internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedsamplelocations);
02668 GLAPI PFNGLTEXIMAGE3DMULTISAMPLEPROC glad_glTexImage3DMultisample;
02669 #define glTexImage3DMultisample glad_glTexImage3DMultisample
02670 typedef void (APIENTRY PFNGLGETMULTISAMPLEFVPROC)(GLenum pname, GLuint index, GLfloat *val);
02671 GLAPI PFNGLGETMULTISAMPLEFVPROC glad_glGetMultisamplefv;
02672 #define glGetMultisamplefv glad_glGetMultisamplefv
02673 typedef void (APIENTRY PFNGLSAMPLERMASKIPROC)(GLuint maskNumber, GLbitfield mask);
02674 GLAPI PFNGLSAMPLERMASKIPROC glad_glSampleMaski;
02675 #define glSampleMaski glad_glSampleMaski
02676 #endif
02677 #ifndef GL_VERSION_3_3
02678 #define GL_VERSION_3_3 1
02679 GLAPI int GLAD_GL_VERSION_3_3;
02680 typedef void (APIENTRY PFNGLBINDFRAGDATALOCATIONINDEXEDPROC)(GLuint program, GLuint colorNumber,
    GLuint index, const GLchar *name);
02681 GLAPI PFNGLBINDFRAGDATALOCATIONINDEXEDPROC glad_glBindFragDataLocationIndexed;
02682 #define glBindFragDataLocationIndexed glad_glBindFragDataLocationIndexed
02683 typedef GLint (APIENTRY PFNGLGETFRAGDATAINDEXPROC)(GLuint program, const GLchar *name);
02684 GLAPI PFNGLGETFRAGDATAINDEXPROC glad_glGetFragDataIndex;
02685 #define glGetFragDataIndex glad_glGetFragDataIndex
02686 typedef void (APIENTRY PFNGLGENSAMPLERSPROC)(GLsizei count, GLuint *samplers);
02687 GLAPI PFNGLGENSAMPLERSPROC glad_glGenSamplers;
02688 #define glGenSamplers glad_glGenSamplers
02689 typedef void (APIENTRY PFNGLDELETESAMPLERSPROC)(GLsizei count, const GLuint *samplers);
02690 GLAPI PFNGLDELETESAMPLERSPROC glad_glDeleteSamplers;
02691 #define glDeleteSamplers glad_glDeleteSamplers
02692 typedef GLboolean (APIENTRY PFNGLISSAMPLERPROC)(GLuint sampler);
02693 GLAPI PFNGLISSAMPLERPROC glad_glIsSampler;
02694 #define glIsSampler glad_glIsSampler
02695 typedef void (APIENTRY PFNGLBINDSAMPLERPROC)(GLuint unit, GLuint sampler);
02696 GLAPI PFNGLBINDSAMPLERPROC glad_glBindSampler;
02697 #define glBindSampler glad_glBindSampler
02698 typedef void (APIENTRY PFNGLSAMPLERPARAMETERIPROC)(GLuint sampler, GLenum pname, GLint param);
02699 GLAPI PFNGLSAMPLERPARAMETERIPROC glad_glSamplerParameteri;
02700 #define glSamplerParameteri glad_glSamplerParameteri
02701 typedef void (APIENTRY PFNGLSAMPLERPARAMETERIVPROC)(GLuint sampler, GLenum pname, const GLint
    *param);
02702 GLAPI PFNGLSAMPLERPARAMETERIVPROC glad_glSamplerParameteriv;
02703 #define glSamplerParameteriv glad_glSamplerParameteriv
02704 typedef void (APIENTRY PFNGLSAMPLERPARAMETERFPROC)(GLuint sampler, GLenum pname, GLfloat param);
02705 GLAPI PFNGLSAMPLERPARAMETERFPROC glad_glSamplerParameterf;
02706 #define glSamplerParameterf glad_glSamplerParameterf

```

```

02707 typedef void (APIENTRYP PFNGLSAMPLERPARAMETERFVPROC) (GLuint sampler, GLenum pname, const GLfloat
    *param);
02708 GLAPI PFNGLSAMPLERPARAMETERFVPROC glad_glSamplerParameterfv;
02709 #define glSamplerParameterfv glad_glSamplerParameterfv
02710 typedef void (APIENTRYP PFNGLSAMPLERPARAMETERIIVPROC) (GLuint sampler, GLenum pname, const GLint
    *param);
02711 GLAPI PFNGLSAMPLERPARAMETERIIVPROC glad_glSamplerParameterIiv;
02712 #define glSamplerParameterIiv glad_glSamplerParameterIiv
02713 typedef void (APIENTRYP PFNGLSAMPLERPARAMETERUIVPROC) (GLuint sampler, GLenum pname, const GLuint
    *param);
02714 GLAPI PFNGLSAMPLERPARAMETERUIVPROC glad_glSamplerParameterIuiv;
02715 #define glSamplerParameterIuiv glad_glSamplerParameterIuiv
02716 typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERIVPROC) (GLuint sampler, GLenum pname, GLint *params);
02717 GLAPI PFNGLGETSAMPLERPARAMETERIVPROC glad_glGetSamplerParameteriv;
02718 #define glGetSamplerParameteriv glad_glGetSamplerParameteriv
02719 typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERIIVPROC) (GLuint sampler, GLenum pname, GLint *params);
02720 GLAPI PFNGLGETSAMPLERPARAMETERIIVPROC glad_glGetSamplerParameterIiv;
02721 #define glGetSamplerParameterIiv glad_glGetSamplerParameterIiv
02722 typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERFVPROC) (GLuint sampler, GLenum pname, GLfloat
    *params);
02723 GLAPI PFNGLGETSAMPLERPARAMETERFVPROC glad_glGetSamplerParameterfv;
02724 #define glGetSamplerParameterfv glad_glGetSamplerParameterfv
02725 typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERUIVPROC) (GLuint sampler, GLenum pname, GLuint
    *params);
02726 GLAPI PFNGLGETSAMPLERPARAMETERUIVPROC glad_glGetSamplerParameterIuiv;
02727 #define glGetSamplerParameterIuiv glad_glGetSamplerParameterIuiv
02728 typedef void (APIENTRYP PFNGLQUERYCOUNTERPROC) (GLuint id, GLenum target);
02729 GLAPI PFNGLQUERYCOUNTERPROC glad_glQueryCounter;
02730 #define glQueryCounter glad_glQueryCounter
02731 typedef void (APIENTRYP PFNGLGETQUERYOBJECTI64VPROC) (GLuint id, GLenum pname, GLint64 *params);
02732 GLAPI PFNGLGETQUERYOBJECTI64VPROC glad_glGetQueryObjecti64v;
02733 #define glGetQueryObjecti64v glad_glGetQueryObjecti64v
02734 typedef void (APIENTRYP PFNGLGETQUERYOBJECTUI64VPROC) (GLuint id, GLenum pname, GLuint64 *params);
02735 GLAPI PFNGLGETQUERYOBJECTUI64VPROC glad_glGetQueryObjectui64v;
02736 #define glGetQueryObjectui64v glad_glGetQueryObjectui64v
02737 typedef void (APIENTRYP PFNGLVERTEXATTRIBDIVISORPROC) (GLuint index, GLuint divisor);
02738 GLAPI PFNGLVERTEXATTRIBDIVISORPROC glad_glVertexAttribDivisor;
02739 #define glVertexAttribDivisor glad_glVertexAttribDivisor
02740 typedef void (APIENTRYP PFNGLVERTEXATTRIBP1UIPROC) (GLuint index, GLenum type, GLboolean normalized,
    GLuint value);
02741 GLAPI PFNGLVERTEXATTRIBP1UIPROC glad_glVertexAttribP1ui;
02742 #define glVertexAttribP1ui glad_glVertexAttribP1ui
02743 typedef void (APIENTRYP PFNGLVERTEXATTRIBP1UIVPROC) (GLuint index, GLenum type, GLboolean normalized,
    const GLuint *value);
02744 GLAPI PFNGLVERTEXATTRIBP1UIVPROC glad_glVertexAttribP1uiv;
02745 #define glVertexAttribP1uiv glad_glVertexAttribP1uiv
02746 typedef void (APIENTRYP PFNGLVERTEXATTRIBP2UIPROC) (GLuint index, GLenum type, GLboolean normalized,
    GLuint value);
02747 GLAPI PFNGLVERTEXATTRIBP2UIPROC glad_glVertexAttribP2ui;
02748 #define glVertexAttribP2ui glad_glVertexAttribP2ui
02749 typedef void (APIENTRYP PFNGLVERTEXATTRIBP2UIVPROC) (GLuint index, GLenum type, GLboolean normalized,
    const GLuint *value);
02750 GLAPI PFNGLVERTEXATTRIBP2UIVPROC glad_glVertexAttribP2uiv;
02751 #define glVertexAttribP2uiv glad_glVertexAttribP2uiv
02752 typedef void (APIENTRYP PFNGLVERTEXATTRIBP3UIPROC) (GLuint index, GLenum type, GLboolean normalized,
    GLuint value);
02753 GLAPI PFNGLVERTEXATTRIBP3UIPROC glad_glVertexAttribP3ui;
02754 #define glVertexAttribP3ui glad_glVertexAttribP3ui
02755 typedef void (APIENTRYP PFNGLVERTEXATTRIBP3UIVPROC) (GLuint index, GLenum type, GLboolean normalized,
    const GLuint *value);
02756 GLAPI PFNGLVERTEXATTRIBP3UIVPROC glad_glVertexAttribP3uiv;
02757 #define glVertexAttribP3uiv glad_glVertexAttribP3uiv
02758 typedef void (APIENTRYP PFNGLVERTEXATTRIBP4UIPROC) (GLuint index, GLenum type, GLboolean normalized,
    GLuint value);
02759 GLAPI PFNGLVERTEXATTRIBP4UIPROC glad_glVertexAttribP4ui;
02760 #define glVertexAttribP4ui glad_glVertexAttribP4ui
02761 typedef void (APIENTRYP PFNGLVERTEXATTRIBP4UIVPROC) (GLuint index, GLenum type, GLboolean normalized,
    const GLuint *value);
02762 GLAPI PFNGLVERTEXATTRIBP4UIVPROC glad_glVertexAttribP4uiv;
02763 #define glVertexAttribP4uiv glad_glVertexAttribP4uiv
02764 typedef void (APIENTRYP PFNGLVERTEXP2UIPROC) (GLenum type, GLuint value);
02765 GLAPI PFNGLVERTEXP2UIPROC glad_glVertexP2ui;
02766 #define glVertexP2ui glad_glVertexP2ui
02767 typedef void (APIENTRYP PFNGLVERTEXP2UIVPROC) (GLenum type, const GLuint *value);
02768 GLAPI PFNGLVERTEXP2UIVPROC glad_glVertexP2uiv;
02769 #define glVertexP2uiv glad_glVertexP2uiv
02770 typedef void (APIENTRYP PFNGLVERTEXP3UIPROC) (GLenum type, GLuint value);
02771 GLAPI PFNGLVERTEXP3UIPROC glad_glVertexP3ui;
02772 #define glVertexP3ui glad_glVertexP3ui
02773 typedef void (APIENTRYP PFNGLVERTEXP3UIVPROC) (GLenum type, const GLuint *value);
02774 GLAPI PFNGLVERTEXP3UIVPROC glad_glVertexP3uiv;
02775 #define glVertexP3uiv glad_glVertexP3uiv
02776 typedef void (APIENTRYP PFNGLVERTEXP4UIPROC) (GLenum type, GLuint value);
02777 GLAPI PFNGLVERTEXP4UIPROC glad_glVertexP4ui;
02778 #define glVertexP4ui glad_glVertexP4ui
02779 typedef void (APIENTRYP PFNGLVERTEXP4UIVPROC) (GLenum type, const GLuint *value);
02780 GLAPI PFNGLVERTEXP4UIVPROC glad_glVertexP4uiv;

```

```

02781 #define glVertexP4uiv glad_glVertexP4uiv
02782 typedef void (APIENTRY PFNGLTEXCOORDP1UIPROC) (GLenum type, GLuint coords);
02783 GLAPI PFNGLTEXCOORDP1UIPROC glad_glTexCoordP1ui;
02784 #define glTexCoordP1ui glad_glTexCoordP1ui
02785 typedef void (APIENTRY PFNGLTEXCOORDP1UIVPROC) (GLenum type, const GLuint *coords);
02786 GLAPI PFNGLTEXCOORDP1UIVPROC glad_glTexCoordP1uiv;
02787 #define glTexCoordP1uiv glad_glTexCoordP1uiv
02788 typedef void (APIENTRY PFNGLTEXCOORDP2UIPROC) (GLenum type, GLuint coords);
02789 GLAPI PFNGLTEXCOORDP2UIPROC glad_glTexCoordP2ui;
02790 #define glTexCoordP2ui glad_glTexCoordP2ui
02791 typedef void (APIENTRY PFNGLTEXCOORDP2UIVPROC) (GLenum type, const GLuint *coords);
02792 GLAPI PFNGLTEXCOORDP2UIVPROC glad_glTexCoordP2uiv;
02793 #define glTexCoordP2uiv glad_glTexCoordP2uiv
02794 typedef void (APIENTRY PFNGLTEXCOORDP3UIPROC) (GLenum type, GLuint coords);
02795 GLAPI PFNGLTEXCOORDP3UIPROC glad_glTexCoordP3ui;
02796 #define glTexCoordP3ui glad_glTexCoordP3ui
02797 typedef void (APIENTRY PFNGLTEXCOORDP3UIVPROC) (GLenum type, const GLuint *coords);
02798 GLAPI PFNGLTEXCOORDP3UIVPROC glad_glTexCoordP3uiv;
02799 #define glTexCoordP3uiv glad_glTexCoordP3uiv
02800 typedef void (APIENTRY PFNGLTEXCOORDP4UIPROC) (GLenum type, GLuint coords);
02801 GLAPI PFNGLTEXCOORDP4UIPROC glad_glTexCoordP4ui;
02802 #define glTexCoordP4ui glad_glTexCoordP4ui
02803 typedef void (APIENTRY PFNGLTEXCOORDP4UIVPROC) (GLenum type, const GLuint *coords);
02804 GLAPI PFNGLTEXCOORDP4UIVPROC glad_glTexCoordP4uiv;
02805 #define glTexCoordP4uiv glad_glTexCoordP4uiv
02806 typedef void (APIENTRY PFNGLMULTITEXCOORDP1UIPROC) (GLenum texture, GLenum type, GLuint coords);
02807 GLAPI PFNGLMULTITEXCOORDP1UIPROC glad_glMultiTexCoordP1ui;
02808 #define glMultiTexCoordP1ui glad_glMultiTexCoordP1ui
02809 typedef void (APIENTRY PFNGLMULTITEXCOORDP1UIVPROC) (GLenum texture, GLenum type, const GLuint
*coords);
02810 GLAPI PFNGLMULTITEXCOORDP1UIVPROC glad_glMultiTexCoordP1uiv;
02811 #define glMultiTexCoordP1uiv glad_glMultiTexCoordP1uiv
02812 typedef void (APIENTRY PFNGLMULTITEXCOORDP2UIPROC) (GLenum texture, GLenum type, GLuint coords);
02813 GLAPI PFNGLMULTITEXCOORDP2UIPROC glad_glMultiTexCoordP2ui;
02814 #define glMultiTexCoordP2ui glad_glMultiTexCoordP2ui
02815 typedef void (APIENTRY PFNGLMULTITEXCOORDP2UIVPROC) (GLenum texture, GLenum type, const GLuint
*coords);
02816 GLAPI PFNGLMULTITEXCOORDP2UIVPROC glad_glMultiTexCoordP2uiv;
02817 #define glMultiTexCoordP2uiv glad_glMultiTexCoordP2uiv
02818 typedef void (APIENTRY PFNGLMULTITEXCOORDP3UIPROC) (GLenum texture, GLenum type, GLuint coords);
02819 GLAPI PFNGLMULTITEXCOORDP3UIPROC glad_glMultiTexCoordP3ui;
02820 #define glMultiTexCoordP3ui glad_glMultiTexCoordP3ui
02821 typedef void (APIENTRY PFNGLMULTITEXCOORDP3UIVPROC) (GLenum texture, GLenum type, const GLuint
*coords);
02822 GLAPI PFNGLMULTITEXCOORDP3UIVPROC glad_glMultiTexCoordP3uiv;
02823 #define glMultiTexCoordP3uiv glad_glMultiTexCoordP3uiv
02824 typedef void (APIENTRY PFNGLMULTITEXCOORDP4UIPROC) (GLenum texture, GLenum type, GLuint coords);
02825 GLAPI PFNGLMULTITEXCOORDP4UIPROC glad_glMultiTexCoordP4ui;
02826 #define glMultiTexCoordP4ui glad_glMultiTexCoordP4ui
02827 typedef void (APIENTRY PFNGLMULTITEXCOORDP4UIVPROC) (GLenum texture, GLenum type, const GLuint
*coords);
02828 GLAPI PFNGLMULTITEXCOORDP4UIVPROC glad_glMultiTexCoordP4uiv;
02829 #define glMultiTexCoordP4uiv glad_glMultiTexCoordP4uiv
02830 typedef void (APIENTRY PFNGLNORMALP3UIPROC) (GLenum type, GLuint coords);
02831 GLAPI PFNGLNORMALP3UIPROC glad_glNormalP3ui;
02832 #define glNormalP3ui glad_glNormalP3ui
02833 typedef void (APIENTRY PFNGLNORMALP3UIVPROC) (GLenum type, const GLuint *coords);
02834 GLAPI PFNGLNORMALP3UIVPROC glad_glNormalP3uiv;
02835 #define glNormalP3uiv glad_glNormalP3uiv
02836 typedef void (APIENTRY PFNGLCOLORP3UIPROC) (GLenum type, GLuint color);
02837 GLAPI PFNGLCOLORP3UIPROC glad_glColorP3ui;
02838 #define glColorP3ui glad_glColorP3ui
02839 typedef void (APIENTRY PFNGLCOLORP3UIVPROC) (GLenum type, const GLuint *color);
02840 GLAPI PFNGLCOLORP3UIVPROC glad_glColorP3uiv;
02841 #define glColorP3uiv glad_glColorP3uiv
02842 typedef void (APIENTRY PFNGLCOLORP4UIPROC) (GLenum type, GLuint color);
02843 GLAPI PFNGLCOLORP4UIPROC glad_glColorP4ui;
02844 #define glColorP4ui glad_glColorP4ui
02845 typedef void (APIENTRY PFNGLCOLORP4UIVPROC) (GLenum type, const GLuint *color);
02846 GLAPI PFNGLCOLORP4UIVPROC glad_glColorP4uiv;
02847 #define glColorP4uiv glad_glColorP4uiv
02848 typedef void (APIENTRY PFNGLSECONDARYCOLORP3UIPROC) (GLenum type, GLuint color);
02849 GLAPI PFNGLSECONDARYCOLORP3UIPROC glad_glSecondaryColorP3ui;
02850 #define glSecondaryColorP3ui glad_glSecondaryColorP3ui
02851 typedef void (APIENTRY PFNGLSECONDARYCOLORP3UIVPROC) (GLenum type, const GLuint *color);
02852 GLAPI PFNGLSECONDARYCOLORP3UIVPROC glad_glSecondaryColorP3uiv;
02853 #define glSecondaryColorP3uiv glad_glSecondaryColorP3uiv
02854 #endif
02855 #ifndef GL_VERSION_4_0
02856 #define GL_VERSION_4_0 1
02857 GLAPI int GLAD_GL_VERSION_4_0;
02858 typedef void (APIENTRY PFNGLMINSAMPLESHADINGPROC) (GLfloat value);
02859 GLAPI PFNGLMINSAMPLESHADINGPROC glad_glMinSampleShading;
02860 #define glMinSampleShading glad_glMinSampleShading
02861 typedef void (APIENTRY PFNGLBLENDEQUATIONIPROC) (GLuint buf, GLenum mode);
02862 GLAPI PFNGLBLENDEQUATIONIPROC glad_glBlendEquationi;
02863 #define glBlendEquationi glad_glBlendEquationi

```



```

02864 typedef void (APIENTRY PFNGLBLENDSEPARATEPROC) (GLuint buf, GLenum modeRGB, GLenum
modeAlpha);
02865 GLAPI PFNGLBLENDSEPARATEPROC glad_glBlendEquationSeparatei;
02866 #define glBlendEquationSeparatei glad_glBlendEquationSeparatei
02867 typedef void (APIENTRY PFNGLBLENDFUNCIPROC) (GLuint buf, GLenum src, GLenum dst);
02868 GLAPI PFNGLBLENDFUNCIPROC glad_glBlendFunci;
02869 #define glBlendFunci glad_glBlendFunci
02870 typedef void (APIENTRY PFNGLBLENDFUNCSEPARATEPROC) (GLuint buf, GLenum srcRGB, GLenum dstRGB, GLenum
srcAlpha, GLenum dstAlpha);
02871 GLAPI PFNGLBLENDFUNCSEPARATEPROC glad_glBlendFuncSeparatei;
02872 #define glBlendFuncSeparatei glad_glBlendFuncSeparatei
02873 typedef void (APIENTRY PFNGLDRAWARRAYSINDIRECTPROC) (GLenum mode, const void *indirect);
02874 GLAPI PFNGLDRAWARRAYSINDIRECTPROC glad_glDrawArraysIndirect;
02875 #define glDrawArraysIndirect glad_glDrawArraysIndirect
02876 typedef void (APIENTRY PFNGLDRAWELEMENTSINDIRECTPROC) (GLenum mode, GLenum type, const void
*indirect);
02877 GLAPI PFNGLDRAWELEMENTSINDIRECTPROC glad_glDrawElementsIndirect;
02878 #define glDrawElementsIndirect glad_glDrawElementsIndirect
02879 typedef void (APIENTRY PFNGLUNIFORM1DPROC) (GLint location, GLdouble x);
02880 GLAPI PFNGLUNIFORM1DPROC glad_glUniform1d;
02881 #define glUniform1d glad_glUniform1d
02882 typedef void (APIENTRY PFNGLUNIFORM2DPROC) (GLint location, GLdouble x, GLdouble y);
02883 GLAPI PFNGLUNIFORM2DPROC glad_glUniform2d;
02884 #define glUniform2d glad_glUniform2d
02885 typedef void (APIENTRY PFNGLUNIFORM3DPROC) (GLint location, GLdouble x, GLdouble y, GLdouble z);
02886 GLAPI PFNGLUNIFORM3DPROC glad_glUniform3d;
02887 #define glUniform3d glad_glUniform3d
02888 typedef void (APIENTRY PFNGLUNIFORM4DPROC) (GLint location, GLdouble x, GLdouble y, GLdouble z,
GLdouble w);
02889 GLAPI PFNGLUNIFORM4DPROC glad_glUniform4d;
02890 #define glUniform4d glad_glUniform4d
02891 typedef void (APIENTRY PFNGLUNIFORM1DVPROC) (GLint location, GLsizei count, const GLdouble *value);
02892 GLAPI PFNGLUNIFORM1DVPROC glad_glUniform1dv;
02893 #define glUniform1dv glad_glUniform1dv
02894 typedef void (APIENTRY PFNGLUNIFORM2DVPROC) (GLint location, GLsizei count, const GLdouble *value);
02895 GLAPI PFNGLUNIFORM2DVPROC glad_glUniform2dv;
02896 #define glUniform2dv glad_glUniform2dv
02897 typedef void (APIENTRY PFNGLUNIFORM3DVPROC) (GLint location, GLsizei count, const GLdouble *value);
02898 GLAPI PFNGLUNIFORM3DVPROC glad_glUniform3dv;
02899 #define glUniform3dv glad_glUniform3dv
02900 typedef void (APIENTRY PFNGLUNIFORM4DVPROC) (GLint location, GLsizei count, const GLdouble *value);
02901 GLAPI PFNGLUNIFORM4DVPROC glad_glUniform4dv;
02902 #define glUniform4dv glad_glUniform4dv
02903 typedef void (APIENTRY PFNGLUNIFORMMATRIX2DVPROC) (GLint location, GLsizei count, GLboolean transpose,
const GLdouble *value);
02904 GLAPI PFNGLUNIFORMMATRIX2DVPROC glad_glUniformMatrix2dv;
02905 #define glUniformMatrix2dv glad_glUniformMatrix2dv
02906 typedef void (APIENTRY PFNGLUNIFORMMATRIX3DVPROC) (GLint location, GLsizei count, GLboolean transpose,
const GLdouble *value);
02907 GLAPI PFNGLUNIFORMMATRIX3DVPROC glad_glUniformMatrix3dv;
02908 #define glUniformMatrix3dv glad_glUniformMatrix3dv
02909 typedef void (APIENTRY PFNGLUNIFORMMATRIX4DVPROC) (GLint location, GLsizei count, GLboolean transpose,
const GLdouble *value);
02910 GLAPI PFNGLUNIFORMMATRIX4DVPROC glad_glUniformMatrix4dv;
02911 #define glUniformMatrix4dv glad_glUniformMatrix4dv
02912 typedef void (APIENTRY PFNGLUNIFORMMATRIX2X3DVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLdouble *value);
02913 GLAPI PFNGLUNIFORMMATRIX2X3DVPROC glad_glUniformMatrix2x3dv;
02914 #define glUniformMatrix2x3dv glad_glUniformMatrix2x3dv
02915 typedef void (APIENTRY PFNGLUNIFORMMATRIX2X4DVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLdouble *value);
02916 GLAPI PFNGLUNIFORMMATRIX2X4DVPROC glad_glUniformMatrix2x4dv;
02917 #define glUniformMatrix2x4dv glad_glUniformMatrix2x4dv
02918 typedef void (APIENTRY PFNGLUNIFORMMATRIX3X2DVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLdouble *value);
02919 GLAPI PFNGLUNIFORMMATRIX3X2DVPROC glad_glUniformMatrix3x2dv;
02920 #define glUniformMatrix3x2dv glad_glUniformMatrix3x2dv
02921 typedef void (APIENTRY PFNGLUNIFORMMATRIX3X4DVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLdouble *value);
02922 GLAPI PFNGLUNIFORMMATRIX3X4DVPROC glad_glUniformMatrix3x4dv;
02923 #define glUniformMatrix3x4dv glad_glUniformMatrix3x4dv
02924 typedef void (APIENTRY PFNGLUNIFORMMATRIX4X2DVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLdouble *value);
02925 GLAPI PFNGLUNIFORMMATRIX4X2DVPROC glad_glUniformMatrix4x2dv;
02926 #define glUniformMatrix4x2dv glad_glUniformMatrix4x2dv
02927 typedef void (APIENTRY PFNGLUNIFORMMATRIX4X3DVPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLdouble *value);
02928 GLAPI PFNGLUNIFORMMATRIX4X3DVPROC glad_glUniformMatrix4x3dv;
02929 #define glUniformMatrix4x3dv glad_glUniformMatrix4x3dv
02930 typedef void (APIENTRY PFNGLGETUNIFORMDVPROC) (GLuint program, GLint location, GLdouble *params);
02931 GLAPI PFNGLGETUNIFORMDVPROC glad_glGetUniformdv;
02932 #define glGetUniformdv glad_glGetUniformdv
02933 typedef GLint (APIENTRY PFNGLGETSUBROUTINEUNIFORMLOCATIONPROC) (GLuint program, GLenum shadertype,
const GLchar *name);
02934 GLAPI PFNGLGETSUBROUTINEUNIFORMLOCATIONPROC glad_glGetSubroutineUniformLocation;
02935 #define glGetSubroutineUniformLocation glad_glGetSubroutineUniformLocation
02936 typedef GLuint (APIENTRY PFNGLGETSUBROUTINEINDEXPROC) (GLuint program, GLenum shadertype, const GLchar

```

```

    *name);
02937 GLAPI PFNGLGETSUBROUTINEINDEXPROC glad_glGetSubroutineIndex;
02938 #define glGetSubroutineIndex glad_glGetSubroutineIndex
02939 typedef void (APIENTRY PFNGLGETACTIVESUBROUTINEUNIFORMIVPROC) (GLuint program, GLenum shadertype,
    GLuint index, GLenum pname, GLint *values);
02940 GLAPI PFNGLGETACTIVESUBROUTINEUNIFORMIVPROC glad_glGetActiveSubroutineUniformiv;
02941 #define glGetActiveSubroutineUniformiv glad_glGetActiveSubroutineUniformiv
02942 typedef void (APIENTRY PFNGLGETACTIVESUBROUTINEUNIFORMNAMEPROC) (GLuint program, GLenum shadertype,
    GLuint index, GLsizei bufSize, GLsizei *length, GLchar *name);
02943 GLAPI PFNGLGETACTIVESUBROUTINEUNIFORMNAMEPROC glad_glGetActiveSubroutineUniformName;
02944 #define glGetActiveSubroutineUniformName glad_glGetActiveSubroutineUniformName
02945 typedef void (APIENTRY PFNGLGETACTIVESUBROUTINENAMEPROC) (GLuint program, GLenum shadertype, GLuint
    index, GLsizei bufSize, GLsizei *length, GLchar *name);
02946 GLAPI PFNGLGETACTIVESUBROUTINENAMEPROC glad_glGetActiveSubroutineName;
02947 #define glGetActiveSubroutineName glad_glGetActiveSubroutineName
02948 typedef void (APIENTRY PFNGLUNIFORMSUBROUTINESUIVPROC) (GLenum shadertype, GLsizei count, const GLuint
    *indices);
02949 GLAPI PFNGLUNIFORMSUBROUTINESUIVPROC glad_glUniformSubroutinesuiv;
02950 #define glUniformSubroutinesuiv glad_glUniformSubroutinesuiv
02951 typedef void (APIENTRY PFNGLGETUNIFORMSUBROUTINEUIVPROC) (GLenum shadertype, GLint location, GLuint
    *params);
02952 GLAPI PFNGLGETUNIFORMSUBROUTINEUIVPROC glad_glGetUniformSubroutineuiv;
02953 #define glGetUniformSubroutineuiv glad_glGetUniformSubroutineuiv
02954 typedef void (APIENTRY PFNGLGETPROGRAMSTAGEIVPROC) (GLuint program, GLenum shadertype, GLenum pname,
    GLint *values);
02955 GLAPI PFNGLGETPROGRAMSTAGEIVPROC glad_glGetProgramStageiv;
02956 #define glGetProgramStageiv glad_glGetProgramStageiv
02957 typedef void (APIENTRY PFNGLPATCHPARAMETERIPROC) (GLenum pname, GLint value);
02958 GLAPI PFNGLPATCHPARAMETERIPROC glad_glPatchParameteri;
02959 #define glPatchParameteri glad_glPatchParameteri
02960 typedef void (APIENTRY PFNGLPATCHPARAMETERFVPROC) (GLenum pname, const GLfloat *values);
02961 GLAPI PFNGLPATCHPARAMETERFVPROC glad_glPatchParameterfv;
02962 #define glPatchParameterfv glad_glPatchParameterfv
02963 typedef void (APIENTRY PFNGLBINDTRANSFORMFEEDBACKPROC) (GLenum target, GLuint id);
02964 GLAPI PFNGLBINDTRANSFORMFEEDBACKPROC glad_glBindTransformFeedback;
02965 #define glBindTransformFeedback glad_glBindTransformFeedback
02966 typedef void (APIENTRY PFNGLDELETETRANSFORMFEEDBACKSPROC) (GLsizei n, const GLuint *ids);
02967 GLAPI PFNGLDELETETRANSFORMFEEDBACKSPROC glad_glDeleteTransformFeedbacks;
02968 #define glDeleteTransformFeedbacks glad_glDeleteTransformFeedbacks
02969 typedef void (APIENTRY PFNGLGENTRANSFORMFEEDBACKSPROC) (GLsizei n, GLuint *ids);
02970 GLAPI PFNGLGENTRANSFORMFEEDBACKSPROC glad_glGenTransformFeedbacks;
02971 #define glGenTransformFeedbacks glad_glGenTransformFeedbacks
02972 typedef void (APIENTRY PFNGLISTRANSFORMFEEDBACKPROC) (GLuint id);
02973 GLAPI PFNGLISTRANSFORMFEEDBACKPROC glad_glIsTransformFeedback;
02974 #define glIsTransformFeedback glad_glIsTransformFeedback
02975 typedef void (APIENTRY PFNGLPAUSETRANSFORMFEEDBACKPROC) (void);
02976 GLAPI PFNGLPAUSETRANSFORMFEEDBACKPROC glad_glPauseTransformFeedback;
02977 #define glPauseTransformFeedback glad_glPauseTransformFeedback
02978 typedef void (APIENTRY PFNGLRESUMETRANSFORMFEEDBACKPROC) (void);
02979 GLAPI PFNGLRESUMETRANSFORMFEEDBACKPROC glad_glResumeTransformFeedback;
02980 #define glResumeTransformFeedback glad_glResumeTransformFeedback
02981 typedef void (APIENTRY PFNGLDRAWTRANSFORMFEEDBACKPROC) (GLenum mode, GLuint id);
02982 GLAPI PFNGLDRAWTRANSFORMFEEDBACKPROC glad_glDrawTransformFeedback;
02983 #define glDrawTransformFeedback glad_glDrawTransformFeedback
02984 typedef void (APIENTRY PFNGLDRAWTRANSFORMFEEDBACKSTREAMPROC) (GLenum mode, GLuint id, GLuint stream);
02985 GLAPI PFNGLDRAWTRANSFORMFEEDBACKSTREAMPROC glad_glDrawTransformFeedbackStream;
02986 #define glDrawTransformFeedbackStream glad_glDrawTransformFeedbackStream
02987 typedef void (APIENTRY PFNGBEGINQUERYINDEXEDPROC) (GLenum target, GLuint index, GLuint id);
02988 GLAPI PFNGBEGINQUERYINDEXEDPROC glad_glBeginQueryIndexed;
02989 #define glBeginQueryIndexed glad_glBeginQueryIndexed
02990 typedef void (APIENTRY PFNGLENDQUERYINDEXEDPROC) (GLenum target, GLuint index);
02991 GLAPI PFNGLENDQUERYINDEXEDPROC glad_glEndQueryIndexed;
02992 #define glEndQueryIndexed glad_glEndQueryIndexed
02993 typedef void (APIENTRY PFNGLGETQUERYINDEXEDIVPROC) (GLenum target, GLuint index, GLenum pname, GLint
    *params);
02994 GLAPI PFNGLGETQUERYINDEXEDIVPROC glad_glGetQueryIndexediv;
02995 #define glGetQueryIndexediv glad_glGetQueryIndexediv
02996 #endif
02997 #ifndef GL_VERSION_4_1
02998 #define GL_VERSION_4_1 1
02999 GLAPI int GLAD_GL_VERSION_4_1;
03000 typedef void (APIENTRY PFNGLRELEASESHADERCOMPILERPROC) (void);
03001 GLAPI PFNGLRELEASESHADERCOMPILERPROC glad_glReleaseShaderCompiler;
03002 #define glReleaseShaderCompiler glad_glReleaseShaderCompiler
03003 typedef void (APIENTRY PFNGLSHADERBINARYPROC) (GLsizei count, const GLuint *shaders, GLenum
    binaryFormat, const void *binary, GLsizei length);
03004 GLAPI PFNGLSHADERBINARYPROC glad_glShaderBinary;
03005 #define glShaderBinary glad_glShaderBinary
03006 typedef void (APIENTRY PFNGLGETSHADERPRECISIONFORMATPROC) (GLenum shadertype, GLenum precisiontype,
    GLint *range, GLint *precision);
03007 GLAPI PFNGLGETSHADERPRECISIONFORMATPROC glad_glGetShaderPrecisionFormat;
03008 #define glGetShaderPrecisionFormat glad_glGetShaderPrecisionFormat
03009 typedef void (APIENTRY PFNGLDEPTHRANGEFPROC) (GLfloat n, GLfloat f);
03010 GLAPI PFNGLDEPTHRANGEFPROC glad_glDepthRangef;
03011 #define glDepthRangef glad_glDepthRangef
03012 typedef void (APIENTRY PFNGLCLEARDEPTHPROC) (GLfloat d);
03013 GLAPI PFNGLCLEARDEPTHPROC glad_glClearDepthf;

```

```

03014 #define glClearDepthf glad_glClearDepthf
03015 typedef void (APIENTRY PFNGLGETPROGRAMBINARYPROC) (GLuint program, GLsizei bufSize, GLsizei *length,
03016             GLenum *binaryFormat, void *binary);
03016 GLAPI PFNGLGETPROGRAMBINARYPROC glad_glGetProgramBinary;
03017 #define glGetProgramBinary glad_glGetProgramBinary
03018 typedef void (APIENTRY PFNGLPROGRAMBINARYPROC) (GLuint program, GLenum binaryFormat, const void
03019             *binary, GLsizei length);
03019 GLAPI PFNGLPROGRAMBINARYPROC glad_glProgramBinary;
03020 #define glProgramBinary glad_glProgramBinary
03021 typedef void (APIENTRY PFNGLPROGRAMPARAMETERIPROC) (GLuint program, GLenum pname, GLint value);
03022 GLAPI PFNGLPROGRAMPARAMETERIPROC glad_glProgramParameteri;
03023 #define glProgramParameteri glad_glProgramParameteri
03024 typedef void (APIENTRY PFNGLUSEPROGRAMSTAGESPROC) (GLuint pipeline, GLbitfield stages, GLuint
03025             program);
03025 GLAPI PFNGLUSEPROGRAMSTAGESPROC glad_glUseProgramStages;
03026 #define glUseProgramStages glad_glUseProgramStages
03027 typedef void (APIENTRY PFNGLACTIVESHADERPROGRAMPROC) (GLuint pipeline, GLuint program);
03028 GLAPI PFNGLACTIVESHADERPROGRAMPROC glad_glActiveShaderProgram;
03029 #define glActiveShaderProgram glad_glActiveShaderProgram
03030 typedef GLuint (APIENTRY PFNGLCREATESHADERPROGRAMVPROC) (GLenum type, GLsizei count, const GLchar
03031             *const*strings);
03031 GLAPI PFNGLCREATESHADERPROGRAMVPROC glad_glCreateShaderProgramv;
03032 #define glCreateShaderProgramv glad_glCreateShaderProgramv
03033 typedef void (APIENTRY PFNGLBINDPROGRAMPIPELINEPROC) (GLuint pipeline);
03034 GLAPI PFNGLBINDPROGRAMPIPELINEPROC glad_glBindProgramPipeline;
03035 #define glBindProgramPipeline glad_glBindProgramPipeline
03036 typedef void (APIENTRY PFNGLDELETEPROGRAMPIPELINESPROC) (GLsizei n, const GLuint *pipelines);
03037 GLAPI PFNGLDELETEPROGRAMPIPELINESPROC glad_glDeleteProgramPipelines;
03038 #define glDeleteProgramPipelines glad_glDeleteProgramPipelines
03039 typedef void (APIENTRY PFNGLGENPROGRAMPIPELINESPROC) (GLsizei n, GLuint *pipelines);
03040 GLAPI PFNGLGENPROGRAMPIPELINESPROC glad_glGenProgramPipelines;
03041 #define glGenProgramPipelines glad_glGenProgramPipelines
03042 typedef GLboolean (APIENTRY PFNGLISPROGRAMPIPELINEPROC) (GLuint pipeline);
03043 GLAPI PFNGLISPROGRAMPIPELINEPROC glad_glIsProgramPipeline;
03044 #define glIsProgramPipeline glad_glIsProgramPipeline
03045 typedef void (APIENTRY PFNGLGETPROGRAMPIPELINEIVPROC) (GLuint pipeline, GLenum pname, GLint *params);
03046 GLAPI PFNGLGETPROGRAMPIPELINEIVPROC glad_glGetProgramPipelineiv;
03047 #define glGetProgramPipelineiv glad_glGetProgramPipelineiv
03048 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1IPROC) (GLuint program, GLint location, GLint v0);
03049 GLAPI PFNGLPROGRAMUNIFORM1IPROC glad_glProgramUniform1i;
03050 #define glProgramUniform1i glad_glProgramUniform1i
03051 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1IVPROC) (GLuint program, GLint location, GLsizei count,
03052             const GLint *value);
03052 GLAPI PFNGLPROGRAMUNIFORM1IVPROC glad_glProgramUniform1iv;
03053 #define glProgramUniform1iv glad_glProgramUniform1iv
03054 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1FPROC) (GLuint program, GLint location, GLfloat v0);
03055 GLAPI PFNGLPROGRAMUNIFORM1FPROC glad_glProgramUniform1f;
03056 #define glProgramUniform1f glad_glProgramUniform1f
03057 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1FVPROC) (GLuint program, GLint location, GLsizei count,
03058             const GLfloat *value);
03058 GLAPI PFNGLPROGRAMUNIFORM1FVPROC glad_glProgramUniform1fv;
03059 #define glProgramUniform1fv glad_glProgramUniform1fv
03060 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1DPROC) (GLuint program, GLint location, GLdouble v0);
03061 GLAPI PFNGLPROGRAMUNIFORM1DPROC glad_glProgramUniform1d;
03062 #define glProgramUniform1d glad_glProgramUniform1d
03063 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1DVPROC) (GLuint program, GLint location, GLsizei count,
03064             const GLdouble *value);
03064 GLAPI PFNGLPROGRAMUNIFORM1DVPROC glad_glProgramUniform1dv;
03065 #define glProgramUniform1dv glad_glProgramUniform1dv
03066 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UIPROC) (GLuint program, GLint location, GLuint v0);
03067 GLAPI PFNGLPROGRAMUNIFORM1UIPROC glad_glProgramUniform1ui;
03068 #define glProgramUniform1ui glad_glProgramUniform1ui
03069 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UIVPROC) (GLuint program, GLint location, GLsizei count,
03070             const GLuint *value);
03070 GLAPI PFNGLPROGRAMUNIFORM1UIVPROC glad_glProgramUniform1uiv;
03071 #define glProgramUniform1uiv glad_glProgramUniform1uiv
03072 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2IPROC) (GLuint program, GLint location, GLint v0, GLint
03073             v1);
03073 GLAPI PFNGLPROGRAMUNIFORM2IPROC glad_glProgramUniform2i;
03074 #define glProgramUniform2i glad_glProgramUniform2i
03075 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2IVPROC) (GLuint program, GLint location, GLsizei count,
03076             const GLint *value);
03076 GLAPI PFNGLPROGRAMUNIFORM2IVPROC glad_glProgramUniform2iv;
03077 #define glProgramUniform2iv glad_glProgramUniform2iv
03078 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2FPROC) (GLuint program, GLint location, GLfloat v0, GLfloat
03079             v1);
03079 GLAPI PFNGLPROGRAMUNIFORM2FPROC glad_glProgramUniform2f;
03080 #define glProgramUniform2f glad_glProgramUniform2f
03081 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2FVPROC) (GLuint program, GLint location, GLsizei count,
03082             const GLfloat *value);
03082 GLAPI PFNGLPROGRAMUNIFORM2FVPROC glad_glProgramUniform2fv;
03083 #define glProgramUniform2fv glad_glProgramUniform2fv
03084 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2DPROC) (GLuint program, GLint location, GLdouble v0,
03085             GLdouble v1);
03085 GLAPI PFNGLPROGRAMUNIFORM2DPROC glad_glProgramUniform2d;
03086 #define glProgramUniform2d glad_glProgramUniform2d
03087 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2DVPROC) (GLuint program, GLint location, GLsizei count,

```

```

    const GLdouble *value);
03088 GLAPI PFNGLPROGRAMUNIFORM2DVPROC glad_glProgramUniform2dv;
03089 #define glProgramUniform2dv glad_glProgramUniform2dv
03090 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UIPROC) (GLuint program, GLint location, GLuint v0, GLuint
    v1);
03091 GLAPI PFNGLPROGRAMUNIFORM2UIPROC glad_glProgramUniform2ui;
03092 #define glProgramUniform2ui glad_glProgramUniform2ui
03093 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UIVPROC) (GLuint program, GLint location, GLsizei count,
    const GLuint *value);
03094 GLAPI PFNGLPROGRAMUNIFORM2UIVPROC glad_glProgramUniform2uiv;
03095 #define glProgramUniform2uiv glad_glProgramUniform2uiv
03096 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3IPROC) (GLuint program, GLint location, GLint v0, GLint v1,
    GLint v2);
03097 GLAPI PFNGLPROGRAMUNIFORM3IPROC glad_glProgramUniform3i;
03098 #define glProgramUniform3i glad_glProgramUniform3i
03099 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3IVPROC) (GLuint program, GLint location, GLsizei count,
    const GLint *value);
03100 GLAPI PFNGLPROGRAMUNIFORM3IVPROC glad_glProgramUniform3iv;
03101 #define glProgramUniform3iv glad_glProgramUniform3iv
03102 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3FPROC) (GLuint program, GLint location, GLfloat v0, GLfloat
    v1, GLfloat v2);
03103 GLAPI PFNGLPROGRAMUNIFORM3FPROC glad_glProgramUniform3f;
03104 #define glProgramUniform3f glad_glProgramUniform3f
03105 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3FVPROC) (GLuint program, GLint location, GLsizei count,
    const GLfloat *value);
03106 GLAPI PFNGLPROGRAMUNIFORM3FVPROC glad_glProgramUniform3fv;
03107 #define glProgramUniform3fv glad_glProgramUniform3fv
03108 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3DPROC) (GLuint program, GLint location, GLdouble v0,
    GLdouble v1, GLdouble v2);
03109 GLAPI PFNGLPROGRAMUNIFORM3DPROC glad_glProgramUniform3d;
03110 #define glProgramUniform3d glad_glProgramUniform3d
03111 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3DVPROC) (GLuint program, GLint location, GLsizei count,
    const GLdouble *value);
03112 GLAPI PFNGLPROGRAMUNIFORM3DVPROC glad_glProgramUniform3dv;
03113 #define glProgramUniform3dv glad_glProgramUniform3dv
03114 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UIPROC) (GLuint program, GLint location, GLuint v0, GLuint
    v1, GLuint v2);
03115 GLAPI PFNGLPROGRAMUNIFORM3UIPROC glad_glProgramUniform3ui;
03116 #define glProgramUniform3ui glad_glProgramUniform3ui
03117 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UIVPROC) (GLuint program, GLint location, GLsizei count,
    const GLuint *value);
03118 GLAPI PFNGLPROGRAMUNIFORM3UIVPROC glad_glProgramUniform3uiv;
03119 #define glProgramUniform3uiv glad_glProgramUniform3uiv
03120 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4IPROC) (GLuint program, GLint location, GLint v0, GLint v1,
    GLint v2, GLint v3);
03121 GLAPI PFNGLPROGRAMUNIFORM4IPROC glad_glProgramUniform4i;
03122 #define glProgramUniform4i glad_glProgramUniform4i
03123 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4IVPROC) (GLuint program, GLint location, GLsizei count,
    const GLint *value);
03124 GLAPI PFNGLPROGRAMUNIFORM4IVPROC glad_glProgramUniform4iv;
03125 #define glProgramUniform4iv glad_glProgramUniform4iv
03126 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4FPROC) (GLuint program, GLint location, GLfloat v0, GLfloat
    v1, GLfloat v2, GLfloat v3);
03127 GLAPI PFNGLPROGRAMUNIFORM4FPROC glad_glProgramUniform4f;
03128 #define glProgramUniform4f glad_glProgramUniform4f
03129 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4FVPROC) (GLuint program, GLint location, GLsizei count,
    const GLfloat *value);
03130 GLAPI PFNGLPROGRAMUNIFORM4FVPROC glad_glProgramUniform4fv;
03131 #define glProgramUniform4fv glad_glProgramUniform4fv
03132 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4DPROC) (GLuint program, GLint location, GLdouble v0,
    GLdouble v1, GLdouble v2, GLdouble v3);
03133 GLAPI PFNGLPROGRAMUNIFORM4DPROC glad_glProgramUniform4d;
03134 #define glProgramUniform4d glad_glProgramUniform4d
03135 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4DVPROC) (GLuint program, GLint location, GLsizei count,
    const GLdouble *value);
03136 GLAPI PFNGLPROGRAMUNIFORM4DVPROC glad_glProgramUniform4dv;
03137 #define glProgramUniform4dv glad_glProgramUniform4dv
03138 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UIPROC) (GLuint program, GLint location, GLuint v0, GLuint
    v1, GLuint v2, GLuint v3);
03139 GLAPI PFNGLPROGRAMUNIFORM4UIPROC glad_glProgramUniform4ui;
03140 #define glProgramUniform4ui glad_glProgramUniform4ui
03141 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UIVPROC) (GLuint program, GLint location, GLsizei count,
    const GLuint *value);
03142 GLAPI PFNGLPROGRAMUNIFORM4UIVPROC glad_glProgramUniform4uiv;
03143 #define glProgramUniform4uiv glad_glProgramUniform4uiv
03144 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2FVPROC) (GLuint program, GLint location, GLsizei
    count, GLboolean transpose, const GLfloat *value);
03145 GLAPI PFNGLPROGRAMUNIFORMMATRIX2FVPROC glad_glProgramUniformMatrix2fv;
03146 #define glProgramUniformMatrix2fv glad_glProgramUniformMatrix2fv
03147 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3FVPROC) (GLuint program, GLint location, GLsizei
    count, GLboolean transpose, const GLfloat *value);
03148 GLAPI PFNGLPROGRAMUNIFORMMATRIX3FVPROC glad_glProgramUniformMatrix3fv;
03149 #define glProgramUniformMatrix3fv glad_glProgramUniformMatrix3fv
03150 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4FVPROC) (GLuint program, GLint location, GLsizei
    count, GLboolean transpose, const GLfloat *value);
03151 GLAPI PFNGLPROGRAMUNIFORMMATRIX4FVPROC glad_glProgramUniformMatrix4fv;
03152 #define glProgramUniformMatrix4fv glad_glProgramUniformMatrix4fv

```



```

03153 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03154 GLAPI PFNGLPROGRAMUNIFORMMATRIX2DVPROC glad_glProgramUniformMatrix2dv;
03155 #define glProgramUniformMatrix2dv glad_glProgramUniformMatrix2dv
03156 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03157 GLAPI PFNGLPROGRAMUNIFORMMATRIX3DVPROC glad_glProgramUniformMatrix3dv;
03158 #define glProgramUniformMatrix3dv glad_glProgramUniformMatrix3dv
03159 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03160 GLAPI PFNGLPROGRAMUNIFORMMATRIX4DVPROC glad_glProgramUniformMatrix4dv;
03161 #define glProgramUniformMatrix4dv glad_glProgramUniformMatrix4dv
03162 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X3FVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLfloat *value);
03163 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3FVPROC glad_glProgramUniformMatrix2x3fv;
03164 #define glProgramUniformMatrix2x3fv glad_glProgramUniformMatrix2x3fv
03165 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X2FVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLfloat *value);
03166 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2FVPROC glad_glProgramUniformMatrix3x2fv;
03167 #define glProgramUniformMatrix3x2fv glad_glProgramUniformMatrix3x2fv
03168 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X4FVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLfloat *value);
03169 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4FVPROC glad_glProgramUniformMatrix2x4fv;
03170 #define glProgramUniformMatrix2x4fv glad_glProgramUniformMatrix2x4fv
03171 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X2FVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLfloat *value);
03172 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2FVPROC glad_glProgramUniformMatrix4x2fv;
03173 #define glProgramUniformMatrix4x2fv glad_glProgramUniformMatrix4x2fv
03174 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X4FVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLfloat *value);
03175 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4FVPROC glad_glProgramUniformMatrix3x4fv;
03176 #define glProgramUniformMatrix3x4fv glad_glProgramUniformMatrix3x4fv
03177 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X3FVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLfloat *value);
03178 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3FVPROC glad_glProgramUniformMatrix4x3fv;
03179 #define glProgramUniformMatrix4x3fv glad_glProgramUniformMatrix4x3fv
03180 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X3DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03181 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3DVPROC glad_glProgramUniformMatrix2x3dv;
03182 #define glProgramUniformMatrix2x3dv glad_glProgramUniformMatrix2x3dv
03183 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X2DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03184 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2DVPROC glad_glProgramUniformMatrix3x2dv;
03185 #define glProgramUniformMatrix3x2dv glad_glProgramUniformMatrix3x2dv
03186 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X4DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03187 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4DVPROC glad_glProgramUniformMatrix2x4dv;
03188 #define glProgramUniformMatrix2x4dv glad_glProgramUniformMatrix2x4dv
03189 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X2DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03190 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2DVPROC glad_glProgramUniformMatrix4x2dv;
03191 #define glProgramUniformMatrix4x2dv glad_glProgramUniformMatrix4x2dv
03192 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X4DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03193 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4DVPROC glad_glProgramUniformMatrix3x4dv;
03194 #define glProgramUniformMatrix3x4dv glad_glProgramUniformMatrix3x4dv
03195 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X3DVPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
03196 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3DVPROC glad_glProgramUniformMatrix4x3dv;
03197 #define glProgramUniformMatrix4x3dv glad_glProgramUniformMatrix4x3dv
03198 typedef void (APIENTRY PFNGLVALIDATEPROGRAMPIPELINEPROC) (GLuint pipeline);
03199 GLAPI PFNGLVALIDATEPROGRAMPIPELINEPROC glad_glValidateProgramPipeline;
03200 #define glValidateProgramPipeline glad_glValidateProgramPipeline
03201 typedef void (APIENTRY PFNGLGETPROGRAMPIPELINEINFOLOGPROC) (GLuint pipeline, GLsizei bufSize, GLsizei
*length, GLchar *infoLog);
03202 GLAPI PFNGLGETPROGRAMPIPELINEINFOLOGPROC glad_glGetProgramPipelineInfoLog;
03203 #define glGetProgramPipelineInfoLog glad_glGetProgramPipelineInfoLog
03204 typedef void (APIENTRY PFNGLVERTEXATTRIB1DPROC) (GLuint index, GLdouble x);
03205 GLAPI PFNGLVERTEXATTRIB1DPROC glad_glVertexAttrib1d;
03206 #define glVertexAttrib1d glad_glVertexAttrib1d
03207 typedef void (APIENTRY PFNGLVERTEXATTRIB2DPROC) (GLuint index, GLdouble x, GLdouble y);
03208 GLAPI PFNGLVERTEXATTRIB2DPROC glad_glVertexAttrib2d;
03209 #define glVertexAttrib2d glad_glVertexAttrib2d
03210 typedef void (APIENTRY PFNGLVERTEXATTRIB3DPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble z);
03211 GLAPI PFNGLVERTEXATTRIB3DPROC glad_glVertexAttrib3d;
03212 #define glVertexAttrib3d glad_glVertexAttrib3d
03213 typedef void (APIENTRY PFNGLVERTEXATTRIB4DPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble z,
GLdouble w);
03214 GLAPI PFNGLVERTEXATTRIB4DPROC glad_glVertexAttrib4d;
03215 #define glVertexAttrib4d glad_glVertexAttrib4d
03216 typedef void (APIENTRY PFNGLVERTEXATTRIB1DVPROC) (GLuint index, const GLdouble *v);
03217 GLAPI PFNGLVERTEXATTRIB1DVPROC glad_glVertexAttrib1dv;
03218 #define glVertexAttrib1dv glad_glVertexAttrib1dv
03219 typedef void (APIENTRY PFNGLVERTEXATTRIB2DVPROC) (GLuint index, const GLdouble *v);
03220 GLAPI PFNGLVERTEXATTRIB2DVPROC glad_glVertexAttrib2dv;
03221 #define glVertexAttrib2dv glad_glVertexAttrib2dv
03222 typedef void (APIENTRY PFNGLVERTEXATTRIB3DVPROC) (GLuint index, const GLdouble *v);

```

```

03223 GLAPI PFNGLVERTEXATTRIBL3DVPROC glad_glVertexAttribL3dv;
03224 #define glVertexAttribL3dv glad_glVertexAttribL3dv
03225 typedef void (APIENTRY PFNGLVERTEXATTRIBL4DVPROC) (GLuint index, const GLdouble *v);
03226 GLAPI PFNGLVERTEXATTRIBL4DVPROC glad_glVertexAttribL4dv;
03227 #define glVertexAttribL4dv glad_glVertexAttribL4dv
03228 typedef void (APIENTRY PFNGLVERTEXATTRIBLPOINTERPROC) (GLuint index, GLint size, GLenum type, GLsizei
stride, const void *pointer);
03229 GLAPI PFNGLVERTEXATTRIBLPOINTERPROC glad_glVertexAttribLPointer;
03230 #define glVertexAttribLPointer glad_glVertexAttribLPointer
03231 typedef void (APIENTRY PFNGLGETVERTEXATTRIBLDVPROC) (GLuint index, GLenum pname, GLdouble *params);
03232 GLAPI PFNGLGETVERTEXATTRIBLDVPROC glad_glGetVertexAttribLdv;
03233 #define glGetVertexAttribLdv glad_glGetVertexAttribLdv
03234 typedef void (APIENTRY PFNGLVIEWPORTARRAYVPROC) (GLuint first, GLsizei count, const GLfloat *v);
03235 GLAPI PFNGLVIEWPORTARRAYVPROC glad_glViewportArrayv;
03236 #define glGetViewportArrayv glad_glViewportArrayv
03237 typedef void (APIENTRY PFNGLVIEWPORTINDEXEDFPROC) (GLuint index, GLfloat x, GLfloat y, GLfloat w,
GLfloat h);
03238 GLAPI PFNGLVIEWPORTINDEXEDFPROC glad_glViewportIndexedf;
03239 #define glGetViewportIndexedf glad_glViewportIndexedf
03240 typedef void (APIENTRY PFNGLVIEWPORTINDEXEDFVPROC) (GLuint index, const GLfloat *v);
03241 GLAPI PFNGLVIEWPORTINDEXEDFVPROC glad_glViewportIndexedfv;
03242 #define glGetViewportIndexedfv glad_glViewportIndexedfv
03243 typedef void (APIENTRY PFNGLSCISSORARRAYVPROC) (GLuint first, GLsizei count, const GLint *v);
03244 GLAPI PFNGLSCISSORARRAYVPROC glad_glScissorArrayv;
03245 #define glGetScissorArrayv glad_glScissorArrayv
03246 typedef void (APIENTRY PFNGLSCISSORINDEXEDPROC) (GLuint index, GLint left, GLint bottom, GLsizei
width, GLsizei height);
03247 GLAPI PFNGLSCISSORINDEXEDPROC glad_glScissorIndexed;
03248 #define glGetScissorIndexed glad_glScissorIndexed
03249 typedef void (APIENTRY PFNGLSCISSORINDEXEDVPROC) (GLuint index, const GLint *v);
03250 GLAPI PFNGLSCISSORINDEXEDVPROC glad_glScissorIndexedv;
03251 #define glGetScissorIndexedv glad_glScissorIndexedv
03252 typedef void (APIENTRY PFNGLDEPTHRANGEARRAYVPROC) (GLuint first, GLsizei count, const GLdouble *v);
03253 GLAPI PFNGLDEPTHRANGEARRAYVPROC glad_glDepthRangeArrayv;
03254 #define glGetDepthRangeArrayv glad_glDepthRangeArrayv
03255 typedef void (APIENTRY PFNGLDEPTHRANGEINDEXEDPROC) (GLuint index, GLdouble n, GLdouble f);
03256 GLAPI PFNGLDEPTHRANGEINDEXEDPROC glad_glDepthRangeIndexed;
03257 #define glGetDepthRangeIndexed glad_glDepthRangeIndexed
03258 typedef void (APIENTRY PFNGLGETFLOATI_VPROC) (GLenum target, GLuint index, GLfloat *data);
03259 GLAPI PFNGLGETFLOATI_VPROC glad_glGetFloati_v;
03260 #define glGetFloati_v glad_glGetFloati_v
03261 typedef void (APIENTRY PFNGLGETDOUBLEI_VPROC) (GLenum target, GLuint index, GLdouble *data);
03262 GLAPI PFNGLGETDOUBLEI_VPROC glad_glGetDoublei_v;
03263 #define glGetDoublei_v glad_glGetDoublei_v
03264 #endif
03265 #define GL_MULTISAMPLE_3DFX 0x86B2
03266 #define GL_SAMPLE_BUFFERS_3DFX 0x86B3
03267 #define GL_SAMPLES_3DFX 0x86B4
03268 #define GL_MULTISAMPLE_BIT_3DFX 0x20000000
03269 #define GL_COMPRESSED_RGB_FXT1_3DFX 0x86B0
03270 #define GL_COMPRESSED_RGBA_FXT1_3DFX 0x86B1
03271 #define GL_FACTOR_MIN_AMD 0x901C
03272 #define GL_FACTOR_MAX_AMD 0x901D
03273 #define GL_MAX_DEBUG_MESSAGE_LENGTH_AMD 0x9143
03274 #define GL_MAX_DEBUG_LOGGED_MESSAGES_AMD 0x9144
03275 #define GL_DEBUG_LOGGED_MESSAGES_AMD 0x9145
03276 #define GL_DEBUG_SEVERITY_HIGH_AMD 0x9146
03277 #define GL_DEBUG_SEVERITY_MEDIUM_AMD 0x9147
03278 #define GL_DEBUG_SEVERITY_LOW_AMD 0x9148
03279 #define GL_DEBUG_CATEGORY_API_ERROR_AMD 0x9149
03280 #define GL_DEBUG_CATEGORY_WINDOW_SYSTEM_AMD 0x914A
03281 #define GL_DEBUG_CATEGORY_DEPRECATION_AMD 0x914B
03282 #define GL_DEBUG_CATEGORY_UNDEFINED_BEHAVIOR_AMD 0x914C
03283 #define GL_DEBUG_CATEGORY_PERFORMANCE_AMD 0x914D
03284 #define GL_DEBUG_CATEGORY_SHADER_COMPILER_AMD 0x914E
03285 #define GL_DEBUG_CATEGORY_APPLICATION_AMD 0x914F
03286 #define GL_DEBUG_CATEGORY_OTHER_AMD 0x9150
03287 #define GL_DEPTH_CLAMP_NEAR_AMD 0x901E
03288 #define GL_DEPTH_CLAMP_FAR_AMD 0x901F
03289 #define GL_RENDERBUFFER_STORAGE_SAMPLES_AMD 0x91B2
03290 #define GL_MAX_COLOR_FRAMEBUFFER_SAMPLES_AMD 0x91B3
03291 #define GL_MAX_COLOR_FRAMEBUFFER_STORAGE_SAMPLES_AMD 0x91B4
03292 #define GL_MAX_DEPTH_STENCIL_FRAMEBUFFER_SAMPLES_AMD 0x91B5
03293 #define GL_NUM_SUPPORTED_MULTISAMPLE_MODES_AMD 0x91B6
03294 #define GL_SUPPORTED_MULTISAMPLE_MODES_AMD 0x91B7
03295 #define GL_SUBSAMPLE_DISTANCE_AMD 0x883F
03296 #define GL_PIXELS_PER_SAMPLE_PATTERN_X_AMD 0x91AE
03297 #define GL_PIXELS_PER_SAMPLE_PATTERN_Y_AMD 0x91AF
03298 #define GL_ALL_PIXELS_AMD 0xFFFFFFFF
03299 #define GL_FLOAT16_NV 0x8FF8
03300 #define GL_FLOAT16_VEC2_NV 0x8FF9
03301 #define GL_FLOAT16_VEC3_NV 0x8FFA
03302 #define GL_FLOAT16_VEC4_NV 0x8FFB
03303 #define GL_FLOAT16_MAT2_AMD 0x91C5
03304 #define GL_FLOAT16_MAT3_AMD 0x91C6
03305 #define GL_FLOAT16_MAT4_AMD 0x91C7
03306 #define GL_FLOAT16_MAT2x3_AMD 0x91C8

```

```
03307 #define GL_FLOAT16_MAT2x4_AMD 0x91C9
03308 #define GL_FLOAT16_MAT3x2_AMD 0x91CA
03309 #define GL_FLOAT16_MAT3x4_AMD 0x91CB
03310 #define GL_FLOAT16_MAT4x2_AMD 0x91CC
03311 #define GL_FLOAT16_MAT4x3_AMD 0x91CD
03312 #define GL_INT64_NV 0x140E
03313 #define GL_UNSIGNED_INT64_NV 0x140F
03314 #define GL_INT8_NV 0x8FE0
03315 #define GL_INT8_VEC2_NV 0x8FE1
03316 #define GL_INT8_VEC3_NV 0x8FE2
03317 #define GL_INT8_VEC4_NV 0x8FE3
03318 #define GL_INT16_NV 0x8FE4
03319 #define GL_INT16_VEC2_NV 0x8FE5
03320 #define GL_INT16_VEC3_NV 0x8FE6
03321 #define GL_INT16_VEC4_NV 0x8FE7
03322 #define GL_INT64_VEC2_NV 0x8FE9
03323 #define GL_INT64_VEC3_NV 0x8FEA
03324 #define GL_INT64_VEC4_NV 0x8FEB
03325 #define GL_UNSIGNED_INT8_NV 0x8FEC
03326 #define GL_UNSIGNED_INT8_VEC2_NV 0x8FED
03327 #define GL_UNSIGNED_INT8_VEC3_NV 0x8FEE
03328 #define GL_UNSIGNED_INT8_VEC4_NV 0x8FEF
03329 #define GL_UNSIGNED_INT16_NV 0x8FF0
03330 #define GL_UNSIGNED_INT16_VEC2_NV 0x8FF1
03331 #define GL_UNSIGNED_INT16_VEC3_NV 0x8FF2
03332 #define GL_UNSIGNED_INT16_VEC4_NV 0x8FF3
03333 #define GL_UNSIGNED_INT64_VEC2_NV 0x8FF5
03334 #define GL_UNSIGNED_INT64_VEC3_NV 0x8FF6
03335 #define GL_UNSIGNED_INT64_VEC4_NV 0x8FF7
03336 #define GL_VERTEX_ELEMENT_SWIZZLE_AMD 0x91A4
03337 #define GL_VERTEX_ID_SWIZZLE_AMD 0x91A5
03338 #define GL_DATA_BUFFER_AMD 0x9151
03339 #define GL_PERFORMANCE_MONITOR_AMD 0x9152
03340 #define GL_QUERY_OBJECT_AMD 0x9153
03341 #define GL_VERTEX_ARRAY_OBJECT_AMD 0x9154
03342 #define GL_SAMPLER_OBJECT_AMD 0x9155
03343 #define GL_OCCLUSION_QUERY_EVENT_MASK_AMD 0x874F
03344 #define GL_QUERY_DEPTH_PASS_EVENT_BIT_AMD 0x00000001
03345 #define GL_QUERY_DEPTH_FAIL_EVENT_BIT_AMD 0x00000002
03346 #define GL_QUERY_STENCIL_FAIL_EVENT_BIT_AMD 0x00000004
03347 #define GL_QUERY_DEPTH_BOUNDS_FAIL_EVENT_BIT_AMD 0x00000008
03348 #define GL_QUERY_ALL_EVENT_BITS_AMD 0xFFFFFFFF
03349 #define GL_COUNTER_TYPE_AMD 0x8BC0
03350 #define GL_COUNTER_RANGE_AMD 0x8BC1
03351 #define GL_UNSIGNED_INT64_AMD 0x8BC2
03352 #define GL_PERCENTAGE_AMD 0x8BC3
03353 #define GL_PERFMON_RESULT_AVAILABLE_AMD 0x8BC4
03354 #define GL_PERFMON_RESULT_SIZE_AMD 0x8BC5
03355 #define GL_PERFMON_RESULT_AMD 0x8BC6
03356 #define GL_EXTERNAL_VIRTUAL_MEMORY_BUFFER_AMD 0x9160
03357 #define GL_QUERY_BUFFER_AMD 0x9192
03358 #define GL_QUERY_BUFFER_BINDING_AMD 0x9193
03359 #define GL_QUERY_RESULT_NO_WAIT_AMD 0x9194
03360 #define GL_VIRTUAL_PAGE_SIZE_X_AMD 0x9195
03361 #define GL_VIRTUAL_PAGE_SIZE_Y_AMD 0x9196
03362 #define GL_VIRTUAL_PAGE_SIZE_Z_AMD 0x9197
03363 #define GL_MAX_SPARSE_TEXTURE_SIZE_AMD 0x9198
03364 #define GL_MAX_SPARSE_3D_TEXTURE_SIZE_AMD 0x9199
03365 #define GL_MAX_SPARSE_ARRAY_TEXTURE_LAYERS 0x919A
03366 #define GL_MIN_SPARSE_LEVEL_AMD 0x919B
03367 #define GL_MIN_LOD_WARNING_AMD 0x919C
03368 #define GL_TEXTURE_STORAGE_SPARSE_BIT_AMD 0x00000001
03369 #define GL_SET_AMD 0x874A
03370 #define GL_REPLACE_VALUE_AMD 0x874B
03371 #define GL_STENCIL_OP_VALUE_AMD 0x874C
03372 #define GL_STENCIL_BACK_OP_VALUE_AMD 0x874D
03373 #define GL_STREAM_RASTERIZATION_AMD 0x91A0
03374 #define GL_SAMPLER_BUFFER_AMD 0x9001
03375 #define GL_INT_SAMPLER_BUFFER_AMD 0x9002
03376 #define GL_UNSIGNED_INT_SAMPLER_BUFFER_AMD 0x9003
03377 #define GL_TESSELLATION_MODE_AMD 0x9004
03378 #define GL_TESSELLATION_FACTOR_AMD 0x9005
03379 #define GL_DISCRETE_AMD 0x9006
03380 #define GL_CONTINUOUS_AMD 0x9007
03381 #define GL_AUX_DEPTH_STENCIL_APPLE 0x8A14
03382 #define GL_UNPACK_CLIENT_STORAGE_APPLE 0x85B2
03383 #define GL_ELEMENT_ARRAY_APPLE 0x8A0C
03384 #define GL_ELEMENT_ARRAY_TYPE_APPLE 0x8A0D
03385 #define GL_ELEMENT_ARRAY_POINTER_APPLE 0x8A0E
03386 #define GL_DRAW_PIXELS_APPLE 0x8A0A
03387 #define GL_FENCE_APPLE 0x8A0B
03388 #define GL_HALF_APPLE 0x140B
03389 #define GL_RGBA_FLOAT32_APPLE 0x8814
03390 #define GL_RGB_FLOAT32_APPLE 0x8815
03391 #define GL_ALPHA_FLOAT32_APPLE 0x8816
03392 #define GL_INTENSITY_FLOAT32_APPLE 0x8817
03393 #define GL_LUMINANCE_FLOAT32_APPLE 0x8818
```

```
03394 #define GL_LUMINANCE_ALPHA_FLOAT32_APPLE 0x8819
03395 #define GL_RGBA_FLOAT16_APPLE 0x881A
03396 #define GL_RGB_FLOAT16_APPLE 0x881B
03397 #define GL_ALPHA_FLOAT16_APPLE 0x881C
03398 #define GL_INTENSITY_FLOAT16_APPLE 0x881D
03399 #define GL_LUMINANCE_FLOAT16_APPLE 0x881E
03400 #define GL_LUMINANCE_ALPHA_FLOAT16_APPLE 0x881F
03401 #define GL_COLOR_FLOAT_APPLE 0x8A0F
03402 #define GL_BUFFER_SERIALIZED_MODIFY_APPLE 0x8A12
03403 #define GL_BUFFER_FLUSHING_UNMAP_APPLE 0x8A13
03404 #define GL_BUFFER_OBJECT_APPLE 0x85B3
03405 #define GL_RELEASED_APPLE 0x8A19
03406 #define GL_VOLATILE_APPLE 0x8A1A
03407 #define GL_RETAINED_APPLE 0x8A1B
03408 #define GL_UNDEFINED_APPLE 0x8A1C
03409 #define GL_PURGEABLE_APPLE 0x8A1D
03410 #define GL_RGB_422_APPLE 0x8A1F
03411 #define GL_UNSIGNED_SHORT_8_8_APPLE 0x85BA
03412 #define GL_UNSIGNED_SHORT_8_8_REV_APPLE 0x85BB
03413 #define GL_RGB_RAW_422_APPLE 0x8A51
03414 #define GL_PACK_ROW_BYTES_APPLE 0x8A15
03415 #define GL_UNPACK_ROW_BYTES_APPLE 0x8A16
03416 #define GL_LIGHT_MODEL_SPECULAR_VECTOR_APPLE 0x85B0
03417 #define GL_TEXTURE_RANGE_LENGTH_APPLE 0x85B7
03418 #define GL_TEXTURE_RANGE_POINTER_APPLE 0x85B8
03419 #define GL_TEXTURE_STORAGE_HINT_APPLE 0x85BC
03420 #define GL_STORAGE_PRIVATE_APPLE 0x85BD
03421 #define GL_STORAGE_CACHED_APPLE 0x85BE
03422 #define GL_STORAGE_SHARED_APPLE 0x85BF
03423 #define GL_TRANSFORM_HINT_APPLE 0x85B1
03424 #define GL_VERTEX_ARRAY_BINDING_APPLE 0x85B5
03425 #define GL_VERTEX_ARRAY_RANGE_APPLE 0x851D
03426 #define GL_VERTEX_ARRAY_RANGE_LENGTH_APPLE 0x851E
03427 #define GL_VERTEX_ARRAY_STORAGE_HINT_APPLE 0x851F
03428 #define GL_VERTEX_ARRAY_RANGE_POINTER_APPLE 0x8521
03429 #define GL_STORAGE_CLIENT_APPLE 0x85B4
03430 #define GL_VERTEX_ATTRIB_MAP1_APPLE 0x8A00
03431 #define GL_VERTEX_ATTRIB_MAP2_APPLE 0x8A01
03432 #define GL_VERTEX_ATTRIB_MAP1_SIZE_APPLE 0x8A02
03433 #define GL_VERTEX_ATTRIB_MAP1_COEFF_APPLE 0x8A03
03434 #define GL_VERTEX_ATTRIB_MAP1_ORDER_APPLE 0x8A04
03435 #define GL_VERTEX_ATTRIB_MAP1_DOMAIN_APPLE 0x8A05
03436 #define GL_VERTEX_ATTRIB_MAP2_SIZE_APPLE 0x8A06
03437 #define GL_VERTEX_ATTRIB_MAP2_COEFF_APPLE 0x8A07
03438 #define GL_VERTEX_ATTRIB_MAP2_ORDER_APPLE 0x8A08
03439 #define GL_VERTEX_ATTRIB_MAP2_DOMAIN_APPLE 0x8A09
03440 #define GL_YCBCR_422_APPLE 0x85B9
03441 #define GL_PRIMITIVE_BOUNDING_BOX_ARB 0x92BE
03442 #define GL_MULTISAMPLE_LINE_WIDTH_RANGE_ARB 0x9381
03443 #define GL_MULTISAMPLE_LINE_WIDTH_GRANULARITY_ARB 0x9382
03444 #define GL_COMPRESSED_RGB8_ETC2 0x9274
03445 #define GL_COMPRESSED_SRGB8_ETC2 0x9275
03446 #define GL_COMPRESSED_RGB8_PUNCHTHROUGH_ALPHA1_ETC2 0x9276
03447 #define GL_COMPRESSED_SRGB8_PUNCHTHROUGH_ALPHA1_ETC2 0x9277
03448 #define GL_COMPRESSED_RGBA8_ETC2_EAC 0x9278
03449 #define GL_COMPRESSED_SRGB8_ALPHA8_ETC2_EAC 0x9279
03450 #define GL_COMPRESSED_R11_EAC 0x9270
03451 #define GL_COMPRESSED_SIGNED_R11_EAC 0x9271
03452 #define GL_COMPRESSED_RG11_EAC 0x9272
03453 #define GL_COMPRESSED_SIGNED_RG11_EAC 0x9273
03454 #define GL_PRIMITIVE_RESTART_FIXED_INDEX 0x8D69
03455 #define GL_ANY_SAMPLES_PASSED_CONSERVATIVE 0x8D6A
03456 #define GL_MAX_ELEMENT_INDEX 0x8D6B
03457 #define GL_UNSIGNED_INT64_ARB 0x140F
03458 #define GL_MAP_PERSISTENT_BIT 0x0040
03459 #define GL_MAP_COHERENT_BIT 0x0080
03460 #define GL_DYNAMIC_STORAGE_BIT 0x0100
03461 #define GL_CLIENT_STORAGE_BIT 0x0200
03462 #define GL_CLIENT_MAPPED_BUFFER_BARRIER_BIT 0x00004000
03463 #define GL_BUFFER_IMMUTABLE_STORAGE 0x821F
03464 #define GL_BUFFER_STORAGE_FLAGS 0x8220
03465 #define GL_SYNC_CL_EVENT_ARB 0x8240
03466 #define GL_SYNC_CL_EVENT_COMPLETE_ARB 0x8241
03467 #define GL_CLEAR_TEXTURE 0x9365
03468 #define GL_NEGATIVE_ONE_TO_ONE 0x935E
03469 #define GL_ZERO_TO_ONE 0x935F
03470 #define GL_CLIP_ORIGIN 0x935C
03471 #define GL_CLIP_DEPTH_MODE 0x935D
03472 #define GL_RGBA_FLOAT_MODE_ARB 0x8820
03473 #define GL_CLAMP_VERTEX_COLOR_ARB 0x891A
03474 #define GL_CLAMP_FRAGMENT_COLOR_ARB 0x891B
03475 #define GL_CLAMP_READ_COLOR_ARB 0x891C
03476 #define GL_FIXED_ONLY_ARB 0x891D
03477 #define GL_UNPACK_COMPRESSED_BLOCK_WIDTH 0x9127
03478 #define GL_UNPACK_COMPRESSED_BLOCK_HEIGHT 0x9128
03479 #define GL_UNPACK_COMPRESSED_BLOCK_DEPTH 0x9129
03480 #define GL_UNPACK_COMPRESSED_BLOCK_SIZE 0x912A
```



```
03481 #define GL_PACK_COMPRESSED_BLOCK_WIDTH 0x912B
03482 #define GL_PACK_COMPRESSED_BLOCK_HEIGHT 0x912C
03483 #define GL_PACK_COMPRESSED_BLOCK_DEPTH 0x912D
03484 #define GL_PACK_COMPRESSED_BLOCK_SIZE 0x912E
03485 #define GL_COMPUTE_SHADER 0x91B9
03486 #define GL_MAX_COMPUTE_UNIFORM_BLOCKS 0x91BB
03487 #define GL_MAX_COMPUTE_TEXTURE_IMAGE_UNITS 0x91BC
03488 #define GL_MAX_COMPUTE_IMAGE_UNIFORMS 0x91BD
03489 #define GL_MAX_COMPUTE_SHARED_MEMORY_SIZE 0x8262
03490 #define GL_MAX_COMPUTE_UNIFORM_COMPONENTS 0x8263
03491 #define GL_MAX_COMPUTE_ATOMIC_COUNTER_BUFFERS 0x8264
03492 #define GL_MAX_COMPUTE_ATOMIC_COUNTERS 0x8265
03493 #define GL_MAX_COMBINED_COMPUTE_UNIFORM_COMPONENTS 0x8266
03494 #define GL_MAX_COMPUTE_WORK_GROUP_INVOCATIONS 0x90EB
03495 #define GL_MAX_COMPUTE_WORK_GROUP_COUNT 0x91BE
03496 #define GL_MAX_COMPUTE_WORK_GROUP_SIZE 0x91BF
03497 #define GL_COMPUTE_WORK_GROUP_SIZE 0x8267
03498 #define GL_UNIFORM_BLOCK_REFERENCED_BY_COMPUTE_SHADER 0x90EC
03499 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_COMPUTE_SHADER 0x90ED
03500 #define GL_DISPATCH_INDIRECT_BUFFER 0x90EE
03501 #define GL_DISPATCH_INDIRECT_BUFFER_BINDING 0x90EF
03502 #define GL_COMPUTE_SHADER_BIT 0x00000020
03503 #define GL_MAX_COMPUTE_VARIABLE_GROUP_INVOCATIONS_ARB 0x9344
03504 #define GL_MAX_COMPUTE_FIXED_GROUP_INVOCATIONS_ARB 0x90EB
03505 #define GL_MAX_COMPUTE_VARIABLE_GROUP_SIZE_ARB 0x9345
03506 #define GL_MAX_COMPUTE_FIXED_GROUP_SIZE_ARB 0x91BF
03507 #define GL_QUERY_WAIT_INVERTED 0x8E17
03508 #define GL_QUERY_NO_WAIT_INVERTED 0x8E18
03509 #define GL_QUERY_BY_REGION_WAIT_INVERTED 0x8E19
03510 #define GL_QUERY_BY_REGION_NO_WAIT_INVERTED 0x8E1A
03511 #define GL_MAX_CULL_DISTANCES 0x82F9
03512 #define GL_MAX_COMBINED_CLIP_AND_CULL_DISTANCES 0x82FA
03513 #define GL_DEBUG_OUTPUT_SYNCHRONOUS_ARB 0x8242
03514 #define GL_DEBUG_NEXT_LOGGED_MESSAGE_LENGTH_ARB 0x8243
03515 #define GL_DEBUG_CALLBACK_FUNCTION_ARB 0x8244
03516 #define GL_DEBUG_CALLBACK_USER_PARAM_ARB 0x8245
03517 #define GL_DEBUG_SOURCE_API_ARB 0x8246
03518 #define GL_DEBUG_SOURCE_WINDOW_SYSTEM_ARB 0x8247
03519 #define GL_DEBUG_SOURCE_SHADER_COMPILER_ARB 0x8248
03520 #define GL_DEBUG_SOURCE_THIRD_PARTY_ARB 0x8249
03521 #define GL_DEBUG_SOURCE_APPLICATION_ARB 0x824A
03522 #define GL_DEBUG_SOURCE_OTHER_ARB 0x824B
03523 #define GL_DEBUG_TYPE_ERROR_ARB 0x824C
03524 #define GL_DEBUG_TYPE_DEPRECATED_BEHAVIOR_ARB 0x824D
03525 #define GL_DEBUG_TYPE_UNDEFINED_BEHAVIOR_ARB 0x824E
03526 #define GL_DEBUG_TYPE_PORTABILITY_ARB 0x824F
03527 #define GL_DEBUG_TYPE_PERFORMANCE_ARB 0x8250
03528 #define GL_DEBUG_TYPE_OTHER_ARB 0x8251
03529 #define GL_MAX_DEBUG_MESSAGE_LENGTH_ARB 0x9143
03530 #define GL_MAX_DEBUG_LOGGED_MESSAGES_ARB 0x9144
03531 #define GL_DEBUG_LOGGED_MESSAGES_ARB 0x9145
03532 #define GL_DEBUG_SEVERITY_HIGH_ARB 0x9146
03533 #define GL_DEBUG_SEVERITY_MEDIUM_ARB 0x9147
03534 #define GL_DEBUG_SEVERITY_LOW_ARB 0x9148
03535 #define GL_DEPTH_COMPONENT16_ARB 0x81A5
03536 #define GL_DEPTH_COMPONENT24_ARB 0x81A6
03537 #define GL_DEPTH_COMPONENT32_ARB 0x81A7
03538 #define GL_TEXTURE_DEPTH_SIZE_ARB 0x884A
03539 #define GL_DEPTH_TEXTURE_MODE_ARB 0x884B
03540 #define GL_TEXTURE_TARGET 0x1006
03541 #define GL_QUERY_TARGET 0x82EA
03542 #define GL_MAX_DRAW_BUFFERS_ARB 0x8824
03543 #define GL_DRAW_BUFFER0_ARB 0x8825
03544 #define GL_DRAW_BUFFER1_ARB 0x8826
03545 #define GL_DRAW_BUFFER2_ARB 0x8827
03546 #define GL_DRAW_BUFFER3_ARB 0x8828
03547 #define GL_DRAW_BUFFER4_ARB 0x8829
03548 #define GL_DRAW_BUFFERS5_ARB 0x882A
03549 #define GL_DRAW_BUFFER6_ARB 0x882B
03550 #define GL_DRAW_BUFFER7_ARB 0x882C
03551 #define GL_DRAW_BUFFER8_ARB 0x882D
03552 #define GL_DRAW_BUFFER9_ARB 0x882E
03553 #define GL_DRAW_BUFFER10_ARB 0x882F
03554 #define GL_DRAW_BUFFER11_ARB 0x8830
03555 #define GL_DRAW_BUFFER12_ARB 0x8831
03556 #define GL_DRAW_BUFFER13_ARB 0x8832
03557 #define GL_DRAW_BUFFER14_ARB 0x8833
03558 #define GL_DRAW_BUFFER15_ARB 0x8834
03559 #define GL_LOCATION_COMPONENT 0x934A
03560 #define GL_TRANSFORM_FEEDBACK_BUFFER_INDEX 0x934B
03561 #define GL_TRANSFORM_FEEDBACK_BUFFER_STRIDE 0x934C
03562 #define GL_MAX_UNIFORM_LOCATIONS 0x826E
03563 #define GL_FRAGMENT_PROGRAM_ARB 0x8804
03564 #define GL_PROGRAM_FORMAT_ASCII_ARB 0x8875
03565 #define GL_PROGRAM_LENGTH_ARB 0x8627
03566 #define GL_PROGRAM_FORMAT_ARB 0x8876
03567 #define GL_PROGRAM_BINDING_ARB 0x8677
```

```
03568 #define GL_PROGRAM_INSTRUCTIONS_ARB 0x88A0
03569 #define GL_MAX_PROGRAM_INSTRUCTIONS_ARB 0x88A1
03570 #define GL_PROGRAM_NATIVE_INSTRUCTIONS_ARB 0x88A2
03571 #define GL_MAX_PROGRAM_NATIVE_INSTRUCTIONS_ARB 0x88A3
03572 #define GL_PROGRAM_TEMPORARIES_ARB 0x88A4
03573 #define GL_MAX_PROGRAM_TEMPORARIES_ARB 0x88A5
03574 #define GL_PROGRAM_NATIVE_TEMPORARIES_ARB 0x88A6
03575 #define GL_MAX_PROGRAM_NATIVE_TEMPORARIES_ARB 0x88A7
03576 #define GL_PROGRAM_PARAMETERS_ARB 0x88A8
03577 #define GL_MAX_PROGRAM_PARAMETERS_ARB 0x88A9
03578 #define GL_PROGRAM_NATIVE_PARAMETERS_ARB 0x88AA
03579 #define GL_MAX_PROGRAM_NATIVE_PARAMETERS_ARB 0x88AB
03580 #define GL_PROGRAM_ATTRIBS_ARB 0x88AC
03581 #define GL_MAX_PROGRAM_ATTRIBS_ARB 0x88AD
03582 #define GL_PROGRAM_NATIVE_ATTRIBS_ARB 0x88AE
03583 #define GL_MAX_PROGRAM_NATIVE_ATTRIBS_ARB 0x88AF
03584 #define GL_MAX_PROGRAM_LOCAL_PARAMETERS_ARB 0x88B4
03585 #define GL_MAX_PROGRAM_ENV_PARAMETERS_ARB 0x88B5
03586 #define GL_PROGRAM_UNDER_NATIVE_LIMITS_ARB 0x88B6
03587 #define GL_PROGRAM_ALU_INSTRUCTIONS_ARB 0x8805
03588 #define GL_PROGRAM_TEX_INSTRUCTIONS_ARB 0x8806
03589 #define GL_PROGRAM_TEX_INDIRECTIONS_ARB 0x8807
03590 #define GL_PROGRAM_NATIVE_ALU_INSTRUCTIONS_ARB 0x8808
03591 #define GL_PROGRAM_NATIVE_TEX_INSTRUCTIONS_ARB 0x8809
03592 #define GL_PROGRAM_NATIVE_TEX_INDIRECTIONS_ARB 0x880A
03593 #define GL_MAX_PROGRAM_ALU_INSTRUCTIONS_ARB 0x880B
03594 #define GL_MAX_PROGRAM_TEX_INSTRUCTIONS_ARB 0x880C
03595 #define GL_MAX_PROGRAM_TEX_INDIRECTIONS_ARB 0x880D
03596 #define GL_MAX_PROGRAM_NATIVE_ALU_INSTRUCTIONS_ARB 0x880E
03597 #define GL_MAX_PROGRAM_NATIVE_TEX_INSTRUCTIONS_ARB 0x880F
03598 #define GL_MAX_PROGRAM_NATIVE_TEX_INDIRECTIONS_ARB 0x8810
03599 #define GL_PROGRAM_STRING_ARB 0x8628
03600 #define GL_PROGRAM_ERROR_POSITION_ARB 0x864B
03601 #define GL_CURRENT_MATRIX_ARB 0x8641
03602 #define GL_TRANSPOSE_CURRENT_MATRIX_ARB 0x88B7
03603 #define GL_CURRENT_MATRIX_STACK_DEPTH_ARB 0x8640
03604 #define GL_MAX_PROGRAM_MATRICES_ARB 0x862F
03605 #define GL_MAX_PROGRAM_MATRIX_STACK_DEPTH_ARB 0x862E
03606 #define GL_MAX_TEXTURE_COORDS_ARB 0x8871
03607 #define GL_MAX_TEXTURE_IMAGE_UNITS_ARB 0x8872
03608 #define GL_PROGRAM_ERROR_STRING_ARB 0x8874
03609 #define GL_MATRIX0_ARB 0x88C0
03610 #define GL_MATRIX1_ARB 0x88C1
03611 #define GL_MATRIX2_ARB 0x88C2
03612 #define GL_MATRIX3_ARB 0x88C3
03613 #define GL_MATRIX4_ARB 0x88C4
03614 #define GL_MATRIX5_ARB 0x88C5
03615 #define GL_MATRIX6_ARB 0x88C6
03616 #define GL_MATRIX7_ARB 0x88C7
03617 #define GL_MATRIX8_ARB 0x88C8
03618 #define GL_MATRIX9_ARB 0x88C9
03619 #define GL_MATRIX10_ARB 0x88CA
03620 #define GL_MATRIX11_ARB 0x88CB
03621 #define GL_MATRIX12_ARB 0x88CC
03622 #define GL_MATRIX13_ARB 0x88CD
03623 #define GL_MATRIX14_ARB 0x88CE
03624 #define GL_MATRIX15_ARB 0x88CF
03625 #define GL_MATRIX16_ARB 0x88D0
03626 #define GL_MATRIX17_ARB 0x88D1
03627 #define GL_MATRIX18_ARB 0x88D2
03628 #define GL_MATRIX19_ARB 0x88D3
03629 #define GL_MATRIX20_ARB 0x88D4
03630 #define GL_MATRIX21_ARB 0x88D5
03631 #define GL_MATRIX22_ARB 0x88D6
03632 #define GL_MATRIX23_ARB 0x88D7
03633 #define GL_MATRIX24_ARB 0x88D8
03634 #define GL_MATRIX25_ARB 0x88D9
03635 #define GL_MATRIX26_ARB 0x88DA
03636 #define GL_MATRIX27_ARB 0x88DB
03637 #define GL_MATRIX28_ARB 0x88DC
03638 #define GL_MATRIX29_ARB 0x88DD
03639 #define GL_MATRIX30_ARB 0x88DE
03640 #define GL_MATRIX31_ARB 0x88DF
03641 #define GL_FRAGMENT_SHADER_ARB 0x8B30
03642 #define GL_MAX_FRAGMENT_UNIFORM_COMPONENTS_ARB 0x8B49
03643 #define GL_FRAGMENT_SHADER_DERIVATIVE_HINT_ARB 0x8B8B
03644 #define GL_FRAMEBUFFER_DEFAULT_WIDTH 0x9310
03645 #define GL_FRAMEBUFFER_DEFAULT_HEIGHT 0x9311
03646 #define GL_FRAMEBUFFER_DEFAULT_LAYERS 0x9312
03647 #define GL_FRAMEBUFFER_DEFAULT_SAMPLES 0x9313
03648 #define GL_FRAMEBUFFER_DEFAULT_FIXED_SAMPLE_LOCATIONS 0x9314
03649 #define GL_MAX_FRAMEBUFFER_WIDTH 0x9315
03650 #define GL_MAX_FRAMEBUFFER_HEIGHT 0x9316
03651 #define GL_MAX_FRAMEBUFFER_LAYERS 0x9317
03652 #define GL_MAX_FRAMEBUFFER_SAMPLES 0x9318
03653 #define GL_INDEX 0x8222
03654 #define GL_LINES_ADJACENCY_ARB 0x000A
```

```
03655 #define GL_LINE_STRIP_ADJACENCY_ARB 0x000B
03656 #define GL_TRIANGLES_ADJACENCY_ARB 0x000C
03657 #define GL_TRIANGLE_STRIP_ADJACENCY_ARB 0x000D
03658 #define GL_PROGRAM_POINT_SIZE_ARB 0x8642
03659 #define GL_MAX_GEOMETRY_TEXTURE_IMAGE_UNITS_ARB 0x8C29
03660 #define GL_FRAMEBUFFER_ATTACHMENT_LAYERED_ARB 0x8DA7
03661 #define GL_FRAMEBUFFER_INCOMPLETE_LAYER_TARGETS_ARB 0x8DA8
03662 #define GL_FRAMEBUFFER_INCOMPLETE_LAYER_COUNT_ARB 0x8DA9
03663 #define GL_GEOMETRY_SHADER_ARB 0x8DD9
03664 #define GL_GEOMETRY_VERTICES_OUT_ARB 0x8DDA
03665 #define GL_GEOMETRY_INPUT_TYPE_ARB 0x8DDB
03666 #define GL_GEOMETRY_OUTPUT_TYPE_ARB 0x8DDC
03667 #define GL_MAX_GEOMETRY_VARYING_COMPONENTS_ARB 0x8DDD
03668 #define GL_MAX_VERTEX_VARYING_COMPONENTS_ARB 0x8DDE
03669 #define GL_MAX_GEOMETRY_UNIFORM_COMPONENTS_ARB 0x8DDF
03670 #define GL_MAX_GEOMETRY_OUTPUT_VERTICES_ARB 0x8DE0
03671 #define GL_MAX_GEOMETRY_TOTAL_OUTPUT_COMPONENTS_ARB 0x8DE1
03672 #define GL_SHADER_BINARY_FORMAT_SPIR_V_ARB 0x9551
03673 #define GL_SPIR_V_BINARY_ARB 0x9552
03674 #define GL_INT64_ARB 0x140E
03675 #define GL_INT64_VEC2_ARB 0x8FE9
03676 #define GL_INT64_VEC3_ARB 0x8FEA
03677 #define GL_INT64_VEC4_ARB 0x8FEB
03678 #define GL_UNSIGNED_INT64_VEC2_ARB 0x8FF5
03679 #define GL_UNSIGNED_INT64_VEC3_ARB 0x8FF6
03680 #define GL_UNSIGNED_INT64_VEC4_ARB 0x8FF7
03681 #define GL_HALF_FLOAT_ARB 0x140B
03682 #define GL_CONVOLUTION_1D 0x8010
03683 #define GL_CONVOLUTION_2D 0x8011
03684 #define GL_SEPARABLE_2D 0x8012
03685 #define GL_CONVOLUTION_BORDER_MODE 0x8013
03686 #define GL_CONVOLUTION_FILTER_SCALE 0x8014
03687 #define GL_CONVOLUTION_FILTER_BIAS 0x8015
03688 #define GL_REDUCE 0x8016
03689 #define GL_CONVOLUTION_FORMAT 0x8017
03690 #define GL_CONVOLUTION_WIDTH 0x8018
03691 #define GL_CONVOLUTION_HEIGHT 0x8019
03692 #define GL_MAX_CONVOLUTION_WIDTH 0x801A
03693 #define GL_MAX_CONVOLUTION_HEIGHT 0x801B
03694 #define GL_POST_CONVOLUTION_RED_SCALE 0x801C
03695 #define GL_POST_CONVOLUTION_GREEN_SCALE 0x801D
03696 #define GL_POST_CONVOLUTION_BLUE_SCALE 0x801E
03697 #define GL_POST_CONVOLUTION_ALPHA_SCALE 0x801F
03698 #define GL_POST_CONVOLUTION_RED_BIAS 0x8020
03699 #define GL_POST_CONVOLUTION_GREEN_BIAS 0x8021
03700 #define GL_POST_CONVOLUTION_BLUE_BIAS 0x8022
03701 #define GL_POST_CONVOLUTION_ALPHA_BIAS 0x8023
03702 #define GL_HISTOGRAM 0x8024
03703 #define GL_PROXY_HISTOGRAM 0x8025
03704 #define GL_HISTOGRAM_WIDTH 0x8026
03705 #define GL_HISTOGRAM_FORMAT 0x8027
03706 #define GL_HISTOGRAM_RED_SIZE 0x8028
03707 #define GL_HISTOGRAM_GREEN_SIZE 0x8029
03708 #define GL_HISTOGRAM_BLUE_SIZE 0x802A
03709 #define GL_HISTOGRAM_ALPHA_SIZE 0x802B
03710 #define GL_HISTOGRAM_LUMINANCE_SIZE 0x802C
03711 #define GL_HISTOGRAM_SINK 0x802D
03712 #define GL_MINMAX 0x802E
03713 #define GL_MINMAX_FORMAT 0x802F
03714 #define GL_MINMAX_SINK 0x8030
03715 #define GL_TABLE_TOO_LARGE 0x8031
03716 #define GL_COLOR_MATRIX 0x80B1
03717 #define GL_COLOR_MATRIX_STACK_DEPTH 0x80B2
03718 #define GL_MAX_COLOR_MATRIX_STACK_DEPTH 0x80B3
03719 #define GL_POST_COLOR_MATRIX_RED_SCALE 0x80B4
03720 #define GL_POST_COLOR_MATRIX_GREEN_SCALE 0x80B5
03721 #define GL_POST_COLOR_MATRIX_BLUE_SCALE 0x80B6
03722 #define GL_POST_COLOR_MATRIX_ALPHA_SCALE 0x80B7
03723 #define GL_POST_COLOR_MATRIX_RED_BIAS 0x80B8
03724 #define GL_POST_COLOR_MATRIX_GREEN_BIAS 0x80B9
03725 #define GL_POST_COLOR_MATRIX_BLUE_BIAS 0x80BA
03726 #define GL_POST_COLOR_MATRIX_ALPHA_BIAS 0x80BB
03727 #define GL_COLOR_TABLE 0x80D0
03728 #define GL_POST_CONVOLUTION_COLOR_TABLE 0x80D1
03729 #define GL_POST_COLOR_MATRIX_COLOR_TABLE 0x80D2
03730 #define GL_PROXY_COLOR_TABLE 0x80D3
03731 #define GL_PROXY_POST_CONVOLUTION_COLOR_TABLE 0x80D4
03732 #define GL_PROXY_POST_COLOR_MATRIX_COLOR_TABLE 0x80D5
03733 #define GL_COLOR_TABLE_SCALE 0x80D6
03734 #define GL_COLOR_TABLE_BIAS 0x80D7
03735 #define GL_COLOR_TABLE_FORMAT 0x80D8
03736 #define GL_COLOR_TABLE_WIDTH 0x80D9
03737 #define GL_COLOR_TABLE_RED_SIZE 0x80DA
03738 #define GL_COLOR_TABLE_GREEN_SIZE 0x80DB
03739 #define GL_COLOR_TABLE_BLUE_SIZE 0x80DC
03740 #define GL_COLOR_TABLE_ALPHA_SIZE 0x80DD
03741 #define GL_COLOR_TABLE_LUMINANCE_SIZE 0x80DE
```

```
03742 #define GL_COLOR_TABLE_INTENSITY_SIZE 0x80DF
03743 #define GL_CONSTANT_BORDER 0x8151
03744 #define GL_REPLICATE_BORDER 0x8153
03745 #define GL_CONVOLUTION_BORDER_COLOR 0x8154
03746 #define GL_PARAMETER_BUFFER_ARB 0x80EE
03747 #define GL_PARAMETER_BUFFER_BINDING_ARB 0x80EF
03748 #define GL_VERTEX_ATTRIB_ARRAY_DIVISOR_ARB 0x88FE
03749 #define GL_NUM_SAMPLE_COUNTS 0x9380
03750 #define GL_IMAGE_FORMAT_COMPATIBILITY_TYPE 0x90C7
03751 #define GL_INTERNALFORMAT_SUPPORTED 0x826F
03752 #define GL_INTERNALFORMAT_PREFERRED 0x8270
03753 #define GL_INTERNALFORMAT_RED_SIZE 0x8271
03754 #define GL_INTERNALFORMAT_GREEN_SIZE 0x8272
03755 #define GL_INTERNALFORMAT_BLUE_SIZE 0x8273
03756 #define GL_INTERNALFORMAT_ALPHA_SIZE 0x8274
03757 #define GL_INTERNALFORMAT_DEPTH_SIZE 0x8275
03758 #define GL_INTERNALFORMAT_STENCIL_SIZE 0x8276
03759 #define GL_INTERNALFORMAT_SHARED_SIZE 0x8277
03760 #define GL_INTERNALFORMAT_RED_TYPE 0x8278
03761 #define GL_INTERNALFORMAT_GREEN_TYPE 0x8279
03762 #define GL_INTERNALFORMAT_BLUE_TYPE 0x827A
03763 #define GL_INTERNALFORMAT_ALPHA_TYPE 0x827B
03764 #define GL_INTERNALFORMAT_DEPTH_TYPE 0x827C
03765 #define GL_INTERNALFORMAT_STENCIL_TYPE 0x827D
03766 #define GL_MAX_WIDTH 0x827E
03767 #define GL_MAX_HEIGHT 0x827F
03768 #define GL_MAX_DEPTH 0x8280
03769 #define GL_MAX_LAYERS 0x8281
03770 #define GL_MAX_COMBINED_DIMENSIONS 0x8282
03771 #define GL_COLOR_COMPONENTS 0x8283
03772 #define GL_DEPTH_COMPONENTS 0x8284
03773 #define GL_STENCIL_COMPONENTS 0x8285
03774 #define GL_COLOR_RENDERABLE 0x8286
03775 #define GL_DEPTH_RENDERABLE 0x8287
03776 #define GL_STENCIL_RENDERABLE 0x8288
03777 #define GL_FRAMEBUFFER_RENDERABLE 0x8289
03778 #define GL_FRAMEBUFFER_RENDERABLE_LAYERED 0x828A
03779 #define GL_FRAMEBUFFER_BLEND 0x828B
03780 #define GL_READ_PIXELS 0x828C
03781 #define GL_READ_PIXELS_FORMAT 0x828D
03782 #define GL_READ_PIXELS_TYPE 0x828E
03783 #define GL_TEXTURE_IMAGE_FORMAT 0x828F
03784 #define GL_TEXTURE_IMAGE_TYPE 0x8290
03785 #define GL_GET_TEXTURE_IMAGE_FORMAT 0x8291
03786 #define GL_GET_TEXTURE_IMAGE_TYPE 0x8292
03787 #define GL_MIPMAP 0x8293
03788 #define GL_MANUAL_GENERATE_MIPMAP 0x8294
03789 #define GL_AUTO_GENERATE_MIPMAP 0x8295
03790 #define GL_COLOR_ENCODING 0x8296
03791 #define GL_SRGB_READ 0x8297
03792 #define GL_SRGB_WRITE 0x8298
03793 #define GL_SRGB_DECODE_ARB 0x8299
03794 #define GL_FILTER 0x829A
03795 #define GL_VERTEX_TEXTURE 0x829B
03796 #define GL_TESS_CONTROL_TEXTURE 0x829C
03797 #define GL_TESS_EVALUATION_TEXTURE 0x829D
03798 #define GL_GEOMETRY_TEXTURE 0x829E
03799 #define GL_FRAGMENT_TEXTURE 0x829F
03800 #define GL_COMPUTE_TEXTURE 0x82A0
03801 #define GL_TEXTURE_SHADOW 0x82A1
03802 #define GL_TEXTURE_GATHER 0x82A2
03803 #define GL_TEXTURE_GATHER_SHADOW 0x82A3
03804 #define GL_SHADER_IMAGE_LOAD 0x82A4
03805 #define GL_SHADER_IMAGE_STORE 0x82A5
03806 #define GL_SHADER_IMAGE_ATOMIC 0x82A6
03807 #define GL_IMAGE_TEXEL_SIZE 0x82A7
03808 #define GL_IMAGE_COMPATIBILITY_CLASS 0x82A8
03809 #define GL_IMAGE_PIXEL_FORMAT 0x82A9
03810 #define GL_IMAGE_PIXEL_TYPE 0x82AA
03811 #define GL_SIMULTANEOUS_TEXTURE_AND_DEPTH_TEST 0x82AC
03812 #define GL_SIMULTANEOUS_TEXTURE_AND_STENCIL_TEST 0x82AD
03813 #define GL_SIMULTANEOUS_TEXTURE_AND_DEPTH_WRITE 0x82AE
03814 #define GL_SIMULTANEOUS_TEXTURE_AND_STENCIL_WRITE 0x82AF
03815 #define GL_TEXTURE_COMPRESSED_BLOCK_WIDTH 0x82B1
03816 #define GL_TEXTURE_COMPRESSED_BLOCK_HEIGHT 0x82B2
03817 #define GL_TEXTURE_COMPRESSED_BLOCK_SIZE 0x82B3
03818 #define GL_CLEAR_BUFFER 0x82B4
03819 #define GL_TEXTURE_VIEW 0x82B5
03820 #define GL_VIEW_COMPATIBILITY_CLASS 0x82B6
03821 #define GL_FULL_SUPPORT 0x82B7
03822 #define GL_CAVEAT_SUPPORT 0x82B8
03823 #define GL_IMAGE_CLASS_4_X_32 0x82B9
03824 #define GL_IMAGE_CLASS_2_X_32 0x82BA
03825 #define GL_IMAGE_CLASS_1_X_32 0x82BB
03826 #define GL_IMAGE_CLASS_4_X_16 0x82BC
03827 #define GL_IMAGE_CLASS_2_X_16 0x82BD
03828 #define GL_IMAGE_CLASS_1_X_16 0x82BE
```

```
03829 #define GL_IMAGE_CLASS_4_X_8 0x82BF
03830 #define GL_IMAGE_CLASS_2_X_8 0x82C0
03831 #define GL_IMAGE_CLASS_1_X_8 0x82C1
03832 #define GL_IMAGE_CLASS_11_11_10 0x82C2
03833 #define GL_IMAGE_CLASS_10_10_10_2 0x82C3
03834 #define GL_VIEW_CLASS_128_BITS 0x82C4
03835 #define GL_VIEW_CLASS_96_BITS 0x82C5
03836 #define GL_VIEW_CLASS_64_BITS 0x82C6
03837 #define GL_VIEW_CLASS_48_BITS 0x82C7
03838 #define GL_VIEW_CLASS_32_BITS 0x82C8
03839 #define GL_VIEW_CLASS_24_BITS 0x82C9
03840 #define GL_VIEW_CLASS_16_BITS 0x82CA
03841 #define GL_VIEW_CLASS_8_BITS 0x82CB
03842 #define GL_VIEW_CLASS_S3TC_DXT1_RGB 0x82CC
03843 #define GL_VIEW_CLASS_S3TC_DXT1_RGBA 0x82CD
03844 #define GL_VIEW_CLASS_S3TC_DXT3_RGBA 0x82CE
03845 #define GL_VIEW_CLASS_S3TC_DXT5_RGBA 0x82CF
03846 #define GL_VIEW_CLASS_RGTC1_RED 0x82D0
03847 #define GL_VIEW_CLASS_RGTC2_RG 0x82D1
03848 #define GL_VIEW_CLASS_BPTC_UNORM 0x82D2
03849 #define GL_VIEW_CLASS_BPTC_FLOAT 0x82D3
03850 #define GL_VIEW_CLASS_EAC_R11 0x9383
03851 #define GL_VIEW_CLASS_EAC_RG11 0x9384
03852 #define GL_VIEW_CLASS_ETC2_RGB 0x9385
03853 #define GL_VIEW_CLASS_ETC2_RGBA 0x9386
03854 #define GL_VIEW_CLASS_ETC2_EAC_RGBA 0x9387
03855 #define GL_VIEW_CLASS_ASTC_4x4_RGBA 0x9388
03856 #define GL_VIEW_CLASS_ASTC_5x4_RGBA 0x9389
03857 #define GL_VIEW_CLASS_ASTC_5x5_RGBA 0x938A
03858 #define GL_VIEW_CLASS_ASTC_6x5_RGBA 0x938B
03859 #define GL_VIEW_CLASS_ASTC_6x6_RGBA 0x938C
03860 #define GL_VIEW_CLASS_ASTC_8x5_RGBA 0x938D
03861 #define GL_VIEW_CLASS_ASTC_8x6_RGBA 0x938E
03862 #define GL_VIEW_CLASS_ASTC_8x8_RGBA 0x938F
03863 #define GL_VIEW_CLASS_ASTC_10x5_RGBA 0x9390
03864 #define GL_VIEW_CLASS_ASTC_10x6_RGBA 0x9391
03865 #define GL_VIEW_CLASS_ASTC_10x8_RGBA 0x9392
03866 #define GL_VIEW_CLASS_ASTC_10x10_RGBA 0x9393
03867 #define GL_VIEW_CLASS_ASTC_12x10_RGBA 0x9394
03868 #define GL_VIEW_CLASS_ASTC_12x12_RGBA 0x9395
03869 #define GL_MIN_MAP_BUFFER_ALIGNMENT 0x90BC
03870 #define GL_MATRIX_PALETTE_ARB 0x8840
03871 #define GL_MAX_MATRIX_PALETTE_STACK_DEPTH_ARB 0x8841
03872 #define GL_MAX_PALETTE_MATRICES_ARB 0x8842
03873 #define GL_CURRENT_PALETTE_MATRIX_ARB 0x8843
03874 #define GL_MATRIX_INDEX_ARRAY_ARB 0x8844
03875 #define GL_CURRENT_MATRIX_INDEX_ARB 0x8845
03876 #define GL_MATRIX_INDEX_ARRAY_SIZE_ARB 0x8846
03877 #define GL_MATRIX_INDEX_ARRAY_TYPE_ARB 0x8847
03878 #define GL_MATRIX_INDEX_ARRAY_STRIDE_ARB 0x8848
03879 #define GL_MATRIX_INDEX_ARRAY_POINTER_ARB 0x8849
03880 #define GL_MULTISAMPLE_ARB 0x809D
03881 #define GL_SAMPLE_ALPHA_TO_COVERAGE_ARB 0x809E
03882 #define GL_SAMPLE_ALPHA_TO_ONE_ARB 0x809F
03883 #define GL_SAMPLE_COVERAGE_ARB 0x80A0
03884 #define GL_SAMPLE_BUFFERS_ARB 0x80A8
03885 #define GL_SAMPLES_ARB 0x80A9
03886 #define GL_SAMPLE_COVERAGE_VALUE_ARB 0x80AA
03887 #define GL_SAMPLE_COVERAGE_INVERT_ARB 0x80AB
03888 #define GL_MULTISAMPLE_BIT_ARB 0x20000000
03889 #define GL_TEXTURE0_ARB 0x84C0
03890 #define GL_TEXTURE1_ARB 0x84C1
03891 #define GL_TEXTURE2_ARB 0x84C2
03892 #define GL_TEXTURE3_ARB 0x84C3
03893 #define GL_TEXTURE4_ARB 0x84C4
03894 #define GL_TEXTURE5_ARB 0x84C5
03895 #define GL_TEXTURE6_ARB 0x84C6
03896 #define GL_TEXTURE7_ARB 0x84C7
03897 #define GL_TEXTURE8_ARB 0x84C8
03898 #define GL_TEXTURE9_ARB 0x84C9
03899 #define GL_TEXTURE10_ARB 0x84CA
03900 #define GL_TEXTURE11_ARB 0x84CB
03901 #define GL_TEXTURE12_ARB 0x84CC
03902 #define GL_TEXTURE13_ARB 0x84CD
03903 #define GL_TEXTURE14_ARB 0x84CE
03904 #define GL_TEXTURE15_ARB 0x84CF
03905 #define GL_TEXTURE16_ARB 0x84D0
03906 #define GL_TEXTURE17_ARB 0x84D1
03907 #define GL_TEXTURE18_ARB 0x84D2
03908 #define GL_TEXTURE19_ARB 0x84D3
03909 #define GL_TEXTURE20_ARB 0x84D4
03910 #define GL_TEXTURE21_ARB 0x84D5
03911 #define GL_TEXTURE22_ARB 0x84D6
03912 #define GL_TEXTURE23_ARB 0x84D7
03913 #define GL_TEXTURE24_ARB 0x84D8
03914 #define GL_TEXTURE25_ARB 0x84D9
03915 #define GL_TEXTURE26_ARB 0x84DA
```



```
03916 #define GL_TEXTURE27_ARB 0x84DB
03917 #define GL_TEXTURE28_ARB 0x84DC
03918 #define GL_TEXTURE29_ARB 0x84DD
03919 #define GL_TEXTURE30_ARB 0x84DE
03920 #define GL_TEXTURE31_ARB 0x84DF
03921 #define GL_ACTIVE_TEXTURE_ARB 0x84E0
03922 #define GL_CLIENT_ACTIVE_TEXTURE_ARB 0x84E1
03923 #define GL_MAX_TEXTURE_UNITS_ARB 0x84E2
03924 #define GL_QUERY_COUNTER_BITS_ARB 0x8864
03925 #define GL_CURRENT_QUERY_ARB 0x8865
03926 #define GL_QUERY_RESULT_ARB 0x8866
03927 #define GL_QUERY_RESULT_AVAILABLE_ARB 0x8867
03928 #define GL_SAMPLES_PASSED_ARB 0x8914
03929 #define GL_MAX_SHADER_COMPILER_THREADS_ARB 0x91B0
03930 #define GL_COMPLETION_STATUS_ARB 0x91B1
03931 #define GL_VERTICES_SUBMITTED_ARB 0x82EE
03932 #define GL_PRIMITIVES_SUBMITTED_ARB 0x82EF
03933 #define GL_VERTEX_SHADER_INVOCATIONS_ARB 0x82F0
03934 #define GL_TESS_CONTROL_SHADER_PATCHES_ARB 0x82F1
03935 #define GL_TESS_EVALUATION_SHADER_INVOCATIONS_ARB 0x82F2
03936 #define GL_GEOMETRY_SHADER_PRIMITIVES_EMITTED_ARB 0x82F3
03937 #define GL_FRAGMENT_SHADER_INVOCATIONS_ARB 0x82F4
03938 #define GL_COMPUTE_SHADER_INVOCATIONS_ARB 0x82F5
03939 #define GL_CLIPPING_INPUT_PRIMITIVES_ARB 0x82F6
03940 #define GL_CLIPPING_OUTPUT_PRIMITIVES_ARB 0x82F7
03941 #define GL_PIXEL_PACK_BUFFER_ARB 0x88EB
03942 #define GL_PIXEL_UNPACK_BUFFER_ARB 0x88EC
03943 #define GL_PIXEL_PACK_BUFFER_BINDING_ARB 0x88ED
03944 #define GL_PIXEL_UNPACK_BUFFER_BINDING_ARB 0x88EF
03945 #define GL_POINT_SIZE_MIN_ARB 0x8126
03946 #define GL_POINT_SIZE_MAX_ARB 0x8127
03947 #define GL_POINT_FADE_THRESHOLD_SIZE_ARB 0x8128
03948 #define GL_POINT_DISTANCE_ATTENUATION_ARB 0x8129
03949 #define GL_POINT_SPRITE_ARB 0x8861
03950 #define GL_COORD_REPLACE_ARB 0x8862
03951 #define GL_POLYGON_OFFSET_CLAMP 0x8E1B
03952 #define GL_UNIFORM 0x92E1
03953 #define GL_UNIFORM_BLOCK 0x92E2
03954 #define GL_PROGRAM_INPUT 0x92E3
03955 #define GL_PROGRAM_OUTPUT 0x92E4
03956 #define GL_BUFFER_VARIABLE 0x92E5
03957 #define GL_SHADER_STORAGE_BLOCK 0x92E6
03958 #define GL_ATOMIC_COUNTER_BUFFER 0x92C0
03959 #define GL_VERTEX_SUBROUTINE 0x92E8
03960 #define GL_TESS_CONTROL_SUBROUTINE 0x92E9
03961 #define GL_TESS_EVALUATION_SUBROUTINE 0x92EA
03962 #define GL_GEOMETRY_SUBROUTINE 0x92EB
03963 #define GL_FRAGMENT_SUBROUTINE 0x92EC
03964 #define GL_COMPUTE_SUBROUTINE 0x92ED
03965 #define GL_VERTEX_SUBROUTINE_UNIFORM 0x92EE
03966 #define GL_TESS_CONTROL_SUBROUTINE_UNIFORM 0x92EF
03967 #define GL_TESS_EVALUATION_SUBROUTINE_UNIFORM 0x92F0
03968 #define GL_GEOMETRY_SUBROUTINE_UNIFORM 0x92F1
03969 #define GL_FRAGMENT_SUBROUTINE_UNIFORM 0x92F2
03970 #define GL_COMPUTE_SUBROUTINE_UNIFORM 0x92F3
03971 #define GL_TRANSFORM_FEEDBACK_VARYING 0x92F4
03972 #define GL_ACTIVE_RESOURCES 0x92F5
03973 #define GL_MAX_NAME_LENGTH 0x92F6
03974 #define GL_MAX_NUM_ACTIVE_VARIABLES 0x92F7
03975 #define GL_MAX_NUM_COMPATIBLE_SUBROUTINES 0x92F8
03976 #define GL_NAME_LENGTH 0x92F9
03977 #define GL_TYPE 0x92FA
03978 #define GL_ARRAY_SIZE 0x92FB
03979 #define GL_OFFSET 0x92FC
03980 #define GL_BLOCK_INDEX 0x92FD
03981 #define GL_ARRAY_STRIDE 0x92FE
03982 #define GL_MATRIX_STRIDE 0x92FF
03983 #define GL_IS_ROW_MAJOR 0x9300
03984 #define GL_ATOMIC_COUNTER_BUFFER_INDEX 0x9301
03985 #define GL_BUFFER_BINDING 0x9302
03986 #define GL_BUFFER_DATA_SIZE 0x9303
03987 #define GL_NUM_ACTIVE_VARIABLES 0x9304
03988 #define GL_ACTIVE_VARIABLES 0x9305
03989 #define GL_REFERENCED_BY_VERTEX_SHADER 0x9306
03990 #define GL_REFERENCED_BY_TESS_CONTROL_SHADER 0x9307
03991 #define GL_REFERENCED_BY_TESS_EVALUATION_SHADER 0x9308
03992 #define GL_REFERENCED_BY_GEOMETRY_SHADER 0x9309
03993 #define GL_REFERENCED_BY_FRAGMENT_SHADER 0x930A
03994 #define GL_REFERENCED_BY_COMPUTE_SHADER 0x930B
03995 #define GL_TOP_LEVEL_ARRAY_SIZE 0x930C
03996 #define GL_TOP_LEVEL_ARRAY_STRIDE 0x930D
03997 #define GL_LOCATION 0x930E
03998 #define GL_LOCATION_INDEX 0x930F
03999 #define GL_IS_PER_PATCH 0x92E7
04000 #define GL_QUERY_BUFFER 0x9192
04001 #define GL_QUERY_BUFFER_BARRIER_BIT 0x00008000
04002 #define GL_QUERY_BUFFER_BINDING 0x9193
```

```
04003 #define GL_QUERY_RESULT_NO_WAIT 0x9194
04004 #define GL_CONTEXT_FLAG_ROBUST_ACCESS_BIT_ARB 0x00000004
04005 #define GL_LOSE_CONTEXT_ON_RESET_ARB 0x8252
04006 #define GL_GUILTY_CONTEXT_RESET_ARB 0x8253
04007 #define GL_INNOCENT_CONTEXT_RESET_ARB 0x8254
04008 #define GL_UNKNOWN_CONTEXT_RESET_ARB 0x8255
04009 #define GL_RESET_NOTIFICATION_STRATEGY_ARB 0x8256
04010 #define GL_NO_RESET_NOTIFICATION_ARB 0x8261
04011 #define GL_SAMPLE_LOCATION_SUBPIXEL_BITS_ARB 0x933D
04012 #define GL_SAMPLE_LOCATION_PIXEL_GRID_WIDTH_ARB 0x933E
04013 #define GL_SAMPLE_LOCATION_PIXEL_GRID_HEIGHT_ARB 0x933F
04014 #define GL_PROGRAMMABLE_SAMPLE_LOCATION_TABLE_SIZE_ARB 0x9340
04015 #define GL_SAMPLE_LOCATION_ARB 0x8E50
04016 #define GL_PROGRAMMABLE_SAMPLE_LOCATION_ARB 0x9341
04017 #define GL_FRAMEBUFFER_PROGRAMMABLE_SAMPLE_LOCATIONS_ARB 0x9342
04018 #define GL_FRAMEBUFFER_SAMPLE_LOCATION_PIXEL_GRID_ARB 0x9343
04019 #define GL_SAMPLE_SHADING_ARB 0x8C36
04020 #define GL_MIN_SAMPLE_SHADING_VALUE_ARB 0x8C37
04021 #define GL_ATOMIC_COUNTER_BUFFER_BINDING 0x92C1
04022 #define GL_ATOMIC_COUNTER_BUFFER_START 0x92C2
04023 #define GL_ATOMIC_COUNTER_BUFFER_SIZE 0x92C3
04024 #define GL_ATOMIC_COUNTER_BUFFER_DATA_SIZE 0x92C4
04025 #define GL_ATOMIC_COUNTER_BUFFER_ACTIVE_ATOMIC_COUNTERS 0x92C5
04026 #define GL_ATOMIC_COUNTER_BUFFER_ACTIVE_ATOMIC_COUNTER_INDICES 0x92C6
04027 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_VERTEX_SHADER 0x92C7
04028 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_TESS_CONTROL_SHADER 0x92C8
04029 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_TESS_EVALUATION_SHADER 0x92C9
04030 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_GEOMETRY_SHADER 0x92CA
04031 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_FRAGMENT_SHADER 0x92CB
04032 #define GL_MAX_VERTEX_ATOMIC_COUNTER_BUFFERS 0x92CC
04033 #define GL_MAX_TESS_CONTROL_ATOMIC_COUNTER_BUFFERS 0x92CD
04034 #define GL_MAX_TESS_EVALUATION_ATOMIC_COUNTER_BUFFERS 0x92CE
04035 #define GL_MAX_GEOMETRY_ATOMIC_COUNTER_BUFFERS 0x92CF
04036 #define GL_MAX_FRAGMENT_ATOMIC_COUNTER_BUFFERS 0x92D0
04037 #define GL_MAX_COMBINED_ATOMIC_COUNTER_BUFFERS 0x92D1
04038 #define GL_MAX_VERTEX_ATOMIC_COUNTERS 0x92D2
04039 #define GL_MAX_TESS_CONTROL_ATOMIC_COUNTERS 0x92D3
04040 #define GL_MAX_TESS_EVALUATION_ATOMIC_COUNTERS 0x92D4
04041 #define GL_MAX_GEOMETRY_ATOMIC_COUNTERS 0x92D5
04042 #define GL_MAX_FRAGMENT_ATOMIC_COUNTERS 0x92D6
04043 #define GL_MAX_COMBINED_ATOMIC_COUNTERS 0x92D7
04044 #define GL_MAX_ATOMIC_COUNTER_BUFFER_SIZE 0x92D8
04045 #define GL_MAX_ATOMIC_COUNTER_BUFFER_BINDINGS 0x92DC
04046 #define GL_ACTIVE_ATOMIC_COUNTER_BUFFERS 0x92D9
04047 #define GL_UNIFORM_ATOMIC_COUNTER_BUFFER_INDEX 0x92DA
04048 #define GL_UNSIGNED_INT_ATOMIC_COUNTER 0x92DB
04049 #define GL_VERTEX_ATTRIB_ARRAY_BARRIER_BIT 0x00000001
04050 #define GL_ELEMENT_ARRAY_BARRIER_BIT 0x00000002
04051 #define GL_UNIFORM_BARRIER_BIT 0x00000004
04052 #define GL_TEXTURE_FETCH_BARRIER_BIT 0x00000008
04053 #define GL_SHADER_IMAGE_ACCESS_BARRIER_BIT 0x00000020
04054 #define GL_COMMAND_BARRIER_BIT 0x00000040
04055 #define GL_PIXEL_BUFFER_BARRIER_BIT 0x00000080
04056 #define GL_TEXTURE_UPDATE_BARRIER_BIT 0x00000100
04057 #define GL_BUFFER_UPDATE_BARRIER_BIT 0x00000200
04058 #define GL_FRAMEBUFFER_BARRIER_BIT 0x00000400
04059 #define GL_TRANSFORM_FEEDBACK_BARRIER_BIT 0x00000800
04060 #define GL_ATOMIC_COUNTER_BARRIER_BIT 0x00001000
04061 #define GL_ALL_BARRIER_BITS 0xFFFFFFFF
04062 #define GL_MAX_IMAGE_UNITS 0x8F38
04063 #define GL_MAX_COMBINED_IMAGE_UNITS_AND_FRAGMENT_OUTPUTS 0x8F39
04064 #define GL_IMAGE_BINDING_NAME 0x8F3A
04065 #define GL_IMAGE_BINDING_LEVEL 0x8F3B
04066 #define GL_IMAGE_BINDING_LAYERED 0x8F3C
04067 #define GL_IMAGE_BINDING_LAYER 0x8F3D
04068 #define GL_IMAGE_BINDING_ACCESS 0x8F3E
04069 #define GL_IMAGE_1D 0x904C
04070 #define GL_IMAGE_2D 0x904D
04071 #define GL_IMAGE_3D 0x904E
04072 #define GL_IMAGE_2D_RECT 0x904F
04073 #define GL_IMAGE_CUBE 0x9050
04074 #define GL_IMAGE_BUFFER 0x9051
04075 #define GL_IMAGE_1D_ARRAY 0x9052
04076 #define GL_IMAGE_2D_ARRAY 0x9053
04077 #define GL_IMAGE_CUBE_MAP_ARRAY 0x9054
04078 #define GL_IMAGE_2D_MULTISAMPLE 0x9055
04079 #define GL_IMAGE_2D_MULTISAMPLE_ARRAY 0x9056
04080 #define GL_INT_IMAGE_1D 0x9057
04081 #define GL_INT_IMAGE_2D 0x9058
04082 #define GL_INT_IMAGE_3D 0x9059
04083 #define GL_INT_IMAGE_2D_RECT 0x905A
04084 #define GL_INT_IMAGE_CUBE 0x905B
04085 #define GL_INT_IMAGE_BUFFER 0x905C
04086 #define GL_INT_IMAGE_1D_ARRAY 0x905D
04087 #define GL_INT_IMAGE_2D_ARRAY 0x905E
04088 #define GL_INT_IMAGE_CUBE_MAP_ARRAY 0x905F
04089 #define GL_INT_IMAGE_2D_MULTISAMPLE 0x9060
```

```
04090 #define GL_INT_IMAGE_2D_MULTISAMPLE_ARRAY 0x9061
04091 #define GL_UNSIGNED_INT_IMAGE_1D 0x9062
04092 #define GL_UNSIGNED_INT_IMAGE_2D 0x9063
04093 #define GL_UNSIGNED_INT_IMAGE_3D 0x9064
04094 #define GL_UNSIGNED_INT_IMAGE_2D_RECT 0x9065
04095 #define GL_UNSIGNED_INT_IMAGE_CUBE 0x9066
04096 #define GL_UNSIGNED_INT_IMAGE_BUFFER 0x9067
04097 #define GL_UNSIGNED_INT_IMAGE_1D_ARRAY 0x9068
04098 #define GL_UNSIGNED_INT_IMAGE_2D_ARRAY 0x9069
04099 #define GL_UNSIGNED_INT_IMAGE_CUBE_MAP_ARRAY 0x906A
04100 #define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE 0x906B
04101 #define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE_ARRAY 0x906C
04102 #define GL_MAX_IMAGE_SAMPLES 0x906D
04103 #define GL_IMAGE_BINDING_FORMAT 0x906E
04104 #define GL_IMAGE_FORMAT_COMPATIBILITY_BY_SIZE 0x90C8
04105 #define GL_IMAGE_FORMAT_COMPATIBILITY_BY_CLASS 0x90C9
04106 #define GL_MAX_VERTEX_IMAGE_UNIFORMS 0x90CA
04107 #define GL_MAX_TESS_CONTROL_IMAGE_UNIFORMS 0x90CB
04108 #define GL_MAX_TESS_EVALUATION_IMAGE_UNIFORMS 0x90CC
04109 #define GL_MAX_GEOMETRY_IMAGE_UNIFORMS 0x90CD
04110 #define GL_MAX_FRAGMENT_IMAGE_UNIFORMS 0x90CE
04111 #define GL_MAX_COMBINED_IMAGE_UNIFORMS 0x90CF
04112 #define GL_PROGRAM_OBJECT_ARB 0x8B40
04113 #define GL_SHADER_OBJECT_ARB 0x8B48
04114 #define GL_OBJECT_TYPE_ARB 0x8B4E
04115 #define GL_OBJECT_SUBTYPE_ARB 0x8B4F
04116 #define GL_FLOAT_VEC2_ARB 0x8B50
04117 #define GL_FLOAT_VEC3_ARB 0x8B51
04118 #define GL_FLOAT_VEC4_ARB 0x8B52
04119 #define GL_INT_VEC2_ARB 0x8B53
04120 #define GL_INT_VEC3_ARB 0x8B54
04121 #define GL_INT_VEC4_ARB 0x8B55
04122 #define GL_BOOL_ARB 0x8B56
04123 #define GL_BOOL_VEC2_ARB 0x8B57
04124 #define GL_BOOL_VEC3_ARB 0x8B58
04125 #define GL_BOOL_VEC4_ARB 0x8B59
04126 #define GL_FLOAT_MAT2_ARB 0x8B5A
04127 #define GL_FLOAT_MAT3_ARB 0x8B5B
04128 #define GL_FLOAT_MAT4_ARB 0x8B5C
04129 #define GL_SAMPLER_1D_ARB 0x8B5D
04130 #define GL_SAMPLER_2D_ARB 0x8B5E
04131 #define GL_SAMPLER_3D_ARB 0x8B5F
04132 #define GL_SAMPLER_CUBE_ARB 0x8B60
04133 #define GL_SAMPLER_1D_SHADOW_ARB 0x8B61
04134 #define GL_SAMPLER_2D_SHADOW_ARB 0x8B62
04135 #define GL_SAMPLER_2D_RECT_ARB 0x8B63
04136 #define GL_SAMPLER_2D_RECT_SHADOW_ARB 0x8B64
04137 #define GL_OBJECT_DELETE_STATUS_ARB 0x8B80
04138 #define GL_OBJECT_COMPILE_STATUS_ARB 0x8B81
04139 #define GL_OBJECT_LINK_STATUS_ARB 0x8B82
04140 #define GL_OBJECT_VALIDATE_STATUS_ARB 0x8B83
04141 #define GL_OBJECT_INFO_LOG_LENGTH_ARB 0x8B84
04142 #define GL_OBJECT_ATTACHED_OBJECTS_ARB 0x8B85
04143 #define GL_OBJECT_ACTIVE_UNIFORMS_ARB 0x8B86
04144 #define GL_OBJECT_ACTIVE_UNIFORM_MAX_LENGTH_ARB 0x8B87
04145 #define GL_OBJECT_SHADER_SOURCE_LENGTH_ARB 0x8B88
04146 #define GL_SHADER_STORAGE_BUFFER 0x90D2
04147 #define GL_SHADER_STORAGE_BUFFER_BINDING 0x90D3
04148 #define GL_SHADER_STORAGE_BUFFER_START 0x90D4
04149 #define GL_SHADER_STORAGE_BUFFER_SIZE 0x90D5
04150 #define GL_MAX_VERTEX_SHADER_STORAGE_BLOCKS 0x90D6
04151 #define GL_MAX_GEOMETRY_SHADER_STORAGE_BLOCKS 0x90D7
04152 #define GL_MAX_TESS_CONTROL_SHADER_STORAGE_BLOCKS 0x90D8
04153 #define GL_MAX_TESS_EVALUATION_SHADER_STORAGE_BLOCKS 0x90D9
04154 #define GL_MAX_FRAGMENT_SHADER_STORAGE_BLOCKS 0x90DA
04155 #define GL_MAX_COMPUTE_SHADER_STORAGE_BLOCKS 0x90DB
04156 #define GL_MAX_COMBINED_SHADER_STORAGE_BLOCKS 0x90DC
04157 #define GL_MAX_SHADER_STORAGE_BUFFER_BINDINGS 0x90DD
04158 #define GL_MAX_SHADER_STORAGE_BLOCK_SIZE 0x90DE
04159 #define GL_SHADER_STORAGE_BUFFER_OFFSET_ALIGNMENT 0x90DF
04160 #define GL_SHADER_STORAGE_BARRIER_BIT 0x00002000
04161 #define GL_MAX_COMBINED_SHADER_OUTPUT_RESOURCES 0x8F39
04162 #define GL_SHADING_LANGUAGE_VERSION_ARB 0x8B8C
04163 #define GL_SHADER_INCLUDE_ARB 0x8DAE
04164 #define GL_NAMED_STRING_LENGTH_ARB 0x8DE9
04165 #define GL_NAMED_STRING_TYPE_ARB 0x8DEA
04166 #define GL_TEXTURE_COMPARE_MODE_ARB 0x884C
04167 #define GL_TEXTURE_COMPARE_FUNC_ARB 0x884D
04168 #define GL_COMPARE_R_TO_TEXTURE_ARB 0x884E
04169 #define GL_TEXTURE_COMPARE_FAIL_VALUE_ARB 0x80BF
04170 #define GL_SPARSE_STORAGE_BIT_ARB 0x0400
04171 #define GL_SPARSE_BUFFER_PAGE_SIZE_ARB 0x82F8
04172 #define GL_TEXTURE_SPARSE_ARB 0x91A6
04173 #define GL_VIRTUAL_PAGE_SIZE_INDEX_ARB 0x91A7
04174 #define GL_NUM_SPARSE_LEVELS_ARB 0x91AA
04175 #define GL_NUM_VIRTUAL_PAGE_SIZES_ARB 0x91A8
04176 #define GL_VIRTUAL_PAGE_SIZE_X_ARB 0x9195
```



```
04177 #define GL_VIRTUAL_PAGE_SIZE_Y_ARB 0x9196
04178 #define GL_VIRTUAL_PAGE_SIZE_Z_ARB 0x9197
04179 #define GL_MAX_SPARSE_TEXTURE_SIZE_ARB 0x9198
04180 #define GL_MAX_SPARSE_3D_TEXTURE_SIZE_ARB 0x9199
04181 #define GL_MAX_SPARSE_ARRAY_TEXTURE_LAYERS_ARB 0x919A
04182 #define GL_SPARSE_TEXTURE_FULL_ARRAY_CUBE_MIPMAPS_ARB 0x91A9
04183 #define GL_SPIR_V_EXTENSIONS 0x9553
04184 #define GL_NUM_SPIR_V_EXTENSIONS 0x9554
04185 #define GL_DEPTH_STENCIL_TEXTURE_MODE 0x90EA
04186 #define GL_CLAMP_TO_BORDER_ARB 0x812D
04187 #define GL_TEXTURE_BUFFER_ARB 0x8C2A
04188 #define GL_MAX_TEXTURE_BUFFER_SIZE_ARB 0x8C2B
04189 #define GL_TEXTURE_BINDING_BUFFER_ARB 0x8C2C
04190 #define GL_TEXTURE_BUFFER_DATA_STORE_BINDING_ARB 0x8C2D
04191 #define GL_TEXTURE_BUFFER_FORMAT_ARB 0x8C2E
04192 #define GL_TEXTURE_BUFFER_OFFSET 0x919D
04193 #define GL_TEXTURE_BUFFER_SIZE 0x919E
04194 #define GL_TEXTURE_BUFFER_OFFSET_ALIGNMENT 0x919F
04195 #define GL_COMPRESSED_ALPHA_ARB 0x84E9
04196 #define GL_COMPRESSED_LUMINANCE_ARB 0x84EA
04197 #define GL_COMPRESSED_LUMINANCE_ALPHA_ARB 0x84EB
04198 #define GL_COMPRESSED_INTENSITY_ARB 0x84EC
04199 #define GL_COMPRESSED_RGB_ARB 0x84ED
04200 #define GL_COMPRESSED_RGBA_ARB 0x84EE
04201 #define GL_TEXTURE_COMPRESSION_HINT_ARB 0x84EF
04202 #define GL_TEXTURE_COMPRESSED_IMAGE_SIZE_ARB 0x86A0
04203 #define GL_TEXTURE_COMPRESSED_ARB 0x86A1
04204 #define GL_NUM_COMPRESSED_TEXTURE_FORMATS_ARB 0x86A2
04205 #define GL_COMPRESSED_TEXTURE_FORMATS_ARB 0x86A3
04206 #define GL_COMPRESSED_RGBA_BPTC_UNORM_ARB 0x8E8C
04207 #define GL_COMPRESSED_SRGB_ALPHA_BPTC_UNORM_ARB 0x8E8D
04208 #define GL_COMPRESSED_RGB_BPTC_SIGNED_FLOAT_ARB 0x8E8E
04209 #define GL_COMPRESSED_RGB_BPTC_UNSIGNED_FLOAT_ARB 0x8E8F
04210 #define GL_NORMAL_MAP_ARB 0x8511
04211 #define GL_REFLECTION_MAP_ARB 0x8512
04212 #define GL_TEXTURE_CUBE_MAP_ARB 0x8513
04213 #define GL_TEXTURE_BINDING_CUBE_MAP_ARB 0x8514
04214 #define GL_TEXTURE_CUBE_MAP_POSITIVE_X_ARB 0x8515
04215 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_X_ARB 0x8516
04216 #define GL_TEXTURE_CUBE_MAP_POSITIVE_Y_ARB 0x8517
04217 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y_ARB 0x8518
04218 #define GL_TEXTURE_CUBE_MAP_POSITIVE_Z_ARB 0x8519
04219 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z_ARB 0x851A
04220 #define GL_PROXY_TEXTURE_CUBE_MAP_ARB 0x851B
04221 #define GL_MAX_CUBE_MAP_TEXTURE_SIZE_ARB 0x851C
04222 #define GL_TEXTURE_CUBE_MAP_ARRAY_ARB 0x9009
04223 #define GL_TEXTURE_BINDING_CUBE_MAP_ARRAY_ARB 0x900A
04224 #define GL_PROXY_TEXTURE_CUBE_MAP_ARRAY_ARB 0x900B
04225 #define GL_SAMPLER_CUBE_MAP_ARRAY_ARB 0x900C
04226 #define GL_SAMPLER_CUBE_MAP_ARRAY_SHADOW_ARB 0x900D
04227 #define GL_INT_SAMPLER_CUBE_MAP_ARRAY_ARB 0x900E
04228 #define GL_UNSIGNED_INT_SAMPLER_CUBE_MAP_ARRAY_ARB 0x900F
04229 #define GL_COMBINE_ARB 0x8570
04230 #define GL_COMBINE_RGB_ARB 0x8571
04231 #define GL_COMBINE_ALPHA_ARB 0x8572
04232 #define GL_SOURCE0_RGB_ARB 0x8580
04233 #define GL_SOURCE1_RGB_ARB 0x8581
04234 #define GL_SOURCE2_RGB_ARB 0x8582
04235 #define GL_SOURCE0_ALPHA_ARB 0x8588
04236 #define GL_SOURCE1_ALPHA_ARB 0x8589
04237 #define GL_SOURCE2_ALPHA_ARB 0x858A
04238 #define GL_OPERAND0_RGB_ARB 0x8590
04239 #define GL_OPERAND1_RGB_ARB 0x8591
04240 #define GL_OPERAND2_RGB_ARB 0x8592
04241 #define GL_OPERAND0_ALPHA_ARB 0x8598
04242 #define GL_OPERAND1_ALPHA_ARB 0x8599
04243 #define GL_OPERAND2_ALPHA_ARB 0x859A
04244 #define GL_RGB_SCALE_ARB 0x8573
04245 #define GL_ADD_SIGNED_ARB 0x8574
04246 #define GL_INTERPOLATE_ARB 0x8575
04247 #define GL_SUBTRACT_ARB 0x84E7
04248 #define GL_CONSTANT_ARB 0x8576
04249 #define GL_PRIMARY_COLOR_ARB 0x8577
04250 #define GL_PREVIOUS_ARB 0x8578
04251 #define GL_DOT3_RGB_ARB 0x86AE
04252 #define GL_DOT3_RGBA_ARB 0x86AF
04253 #define GL_TEXTURE_MAX_ANISOTROPY 0x84FE
04254 #define GL_MAX_TEXTURE_MAX_ANISOTROPY 0x84FF
04255 #define GL_TEXTURE_REDUCTION_MODE_ARB 0x9366
04256 #define GL_WEIGHTED_AVERAGE_ARB 0x9367
04257 #define GL_TEXTURE_RED_TYPE_ARB 0x8C10
04258 #define GL_TEXTURE_GREEN_TYPE_ARB 0x8C11
04259 #define GL_TEXTURE_BLUE_TYPE_ARB 0x8C12
04260 #define GL_TEXTURE_ALPHA_TYPE_ARB 0x8C13
04261 #define GL_TEXTURE_LUMINANCE_TYPE_ARB 0x8C14
04262 #define GL_TEXTURE_INTENSITY_TYPE_ARB 0x8C15
04263 #define GL_TEXTURE_DEPTH_TYPE_ARB 0x8C16
```

```
04264 #define GL_UNSIGNED_NORMALIZED_ARB 0x8C17
04265 #define GL_RGBA32F_ARB 0x8814
04266 #define GL_RGB32F_ARB 0x8815
04267 #define GL_ALPHA32F_ARB 0x8816
04268 #define GL_INTENSITY32F_ARB 0x8817
04269 #define GL_LUMINANCE32F_ARB 0x8818
04270 #define GL_LUMINANCE_ALPHA32F_ARB 0x8819
04271 #define GL_RGBA16F_ARB 0x881A
04272 #define GL_RGB16F_ARB 0x881B
04273 #define GL_ALPHA16F_ARB 0x881C
04274 #define GL_INTENSITY16F_ARB 0x881D
04275 #define GL_LUMINANCE16F_ARB 0x881E
04276 #define GL_LUMINANCE_ALPHA16F_ARB 0x881F
04277 #define GL_MIN_PROGRAM_TEXTURE_GATHER_OFFSET_ARB 0x8E5E
04278 #define GL_MAX_PROGRAM_TEXTURE_GATHER_OFFSET_ARB 0x8E5F
04279 #define GL_MAX_PROGRAM_TEXTURE_GATHER_COMPONENTS_ARB 0x8F9F
04280 #define GL_MIRROR_CLAMP_TO_EDGE 0x8743
04281 #define GL_MIRRORED_REPEAT_ARB 0x8370
04282 #define GL_TEXTURE_RECTANGLE_ARB 0x84F5
04283 #define GL_TEXTURE_BINDING_RECTANGLE_ARB 0x84F6
04284 #define GL_PROXY_TEXTURE_RECTANGLE_ARB 0x84F7
04285 #define GL_MAX_RECTANGLE_TEXTURE_SIZE_ARB 0x84F8
04286 #define GL_TEXTURE_IMMUTABLE_FORMAT 0x912F
04287 #define GL_TEXTURE_VIEW_MIN_LEVEL 0x82DB
04288 #define GL_TEXTURE_VIEW_NUM_LEVELS 0x82DC
04289 #define GL_TEXTURE_VIEW_MIN_LAYER 0x82DD
04290 #define GL_TEXTURE_VIEW_NUM_LAYERS 0x82DE
04291 #define GL_TEXTURE_IMMUTABLE_LEVELS 0x82DF
04292 #define GL_TRANSFORM_FEEDBACK_OVERFLOW_ARB 0x82EC
04293 #define GL_TRANSFORM_FEEDBACK_STREAM_OVERFLOW_ARB 0x82ED
04294 #define GL_TRANSPOSE_MODELVIEW_MATRIX_ARB 0x84E3
04295 #define GL_TRANSPOSE_PROJECTION_MATRIX_ARB 0x84E4
04296 #define GL_TRANSPOSE_TEXTURE_MATRIX_ARB 0x84E5
04297 #define GL_TRANSPOSE_COLOR_MATRIX_ARB 0x84E6
04298 #define GL_VERTEX_ATTRIB_BINDING 0x82D4
04299 #define GL_VERTEX_ATTRIB_RELATIVE_OFFSET 0x82D5
04300 #define GL_VERTEX_BINDING_DIVISOR 0x82D6
04301 #define GL_VERTEX_BINDING_OFFSET 0x82D7
04302 #define GL_VERTEX_BINDING_STRIDE 0x82D8
04303 #define GL_MAX_VERTEX_ATTRIB_RELATIVE_OFFSET 0x82D9
04304 #define GL_MAX_VERTEX_ATTRIB_BINDINGS 0x82DA
04305 #define GL_MAX_VERTEX_UNITS_ARB 0x86A4
04306 #define GL_ACTIVE_VERTEX_UNITS_ARB 0x86A5
04307 #define GL_WEIGHT_SUM_UNITY_ARB 0x86A6
04308 #define GL_VERTEX_BLEND_ARB 0x86A7
04309 #define GL_CURRENT_WEIGHT_ARB 0x86A8
04310 #define GL_WEIGHT_ARRAY_TYPE_ARB 0x86A9
04311 #define GL_WEIGHT_ARRAY_STRIDE_ARB 0x86AA
04312 #define GL_WEIGHT_ARRAY_SIZE_ARB 0x86AB
04313 #define GL_WEIGHT_ARRAY_POINTER_ARB 0x86AC
04314 #define GL_WEIGHT_ARRAY_ARB 0x86AD
04315 #define GL_MODELVIEW0_ARB 0x1700
04316 #define GL_MODELVIEW1_ARB 0x850A
04317 #define GL_MODELVIEW2_ARB 0x8722
04318 #define GL_MODELVIEW3_ARB 0x8723
04319 #define GL_MODELVIEW4_ARB 0x8724
04320 #define GL_MODELVIEW5_ARB 0x8725
04321 #define GL_MODELVIEW6_ARB 0x8726
04322 #define GL_MODELVIEW7_ARB 0x8727
04323 #define GL_MODELVIEW8_ARB 0x8728
04324 #define GL_MODELVIEW9_ARB 0x8729
04325 #define GL_MODELVIEW10_ARB 0x872A
04326 #define GL_MODELVIEW11_ARB 0x872B
04327 #define GL_MODELVIEW12_ARB 0x872C
04328 #define GL_MODELVIEW13_ARB 0x872D
04329 #define GL_MODELVIEW14_ARB 0x872E
04330 #define GL_MODELVIEW15_ARB 0x872F
04331 #define GL_MODELVIEW16_ARB 0x8730
04332 #define GL_MODELVIEW17_ARB 0x8731
04333 #define GL_MODELVIEW18_ARB 0x8732
04334 #define GL_MODELVIEW19_ARB 0x8733
04335 #define GL_MODELVIEW20_ARB 0x8734
04336 #define GL_MODELVIEW21_ARB 0x8735
04337 #define GL_MODELVIEW22_ARB 0x8736
04338 #define GL_MODELVIEW23_ARB 0x8737
04339 #define GL_MODELVIEW24_ARB 0x8738
04340 #define GL_MODELVIEW25_ARB 0x8739
04341 #define GL_MODELVIEW26_ARB 0x873A
04342 #define GL_MODELVIEW27_ARB 0x873B
04343 #define GL_MODELVIEW28_ARB 0x873C
04344 #define GL_MODELVIEW29_ARB 0x873D
04345 #define GL_MODELVIEW30_ARB 0x873E
04346 #define GL_MODELVIEW31_ARB 0x873F
04347 #define GL_BUFFER_SIZE_ARB 0x8764
04348 #define GL_BUFFER_USAGE_ARB 0x8765
04349 #define GL_ARRAY_BUFFER_ARB 0x8892
04350 #define GL_ELEMENT_ARRAY_BUFFER_ARB 0x8893
```

```
04351 #define GL_ARRAY_BUFFER_BINDING_ARB 0x8894
04352 #define GL_ELEMENT_ARRAY_BUFFER_BINDING_ARB 0x8895
04353 #define GL_VERTEX_ARRAY_BUFFER_BINDING_ARB 0x8896
04354 #define GL_NORMAL_ARRAY_BUFFER_BINDING_ARB 0x8897
04355 #define GL_COLOR_ARRAY_BUFFER_BINDING_ARB 0x8898
04356 #define GL_INDEX_ARRAY_BUFFER_BINDING_ARB 0x8899
04357 #define GL_TEXTURE_COORD_ARRAY_BUFFER_BINDING_ARB 0x889A
04358 #define GL_EDGE_FLAG_ARRAY_BUFFER_BINDING_ARB 0x889B
04359 #define GL_SECONDARY_COLOR_ARRAY_BUFFER_BINDING_ARB 0x889C
04360 #define GL_FOG_COORDINATE_ARRAY_BUFFER_BINDING_ARB 0x889D
04361 #define GL_WEIGHT_ARRAY_BUFFER_BINDING_ARB 0x889E
04362 #define GL_VERTEX_ATTRIB_ARRAY_BUFFER_BINDING_ARB 0x889F
04363 #define GL_READ_ONLY_ARB 0x88B8
04364 #define GL_WRITE_ONLY_ARB 0x88B9
04365 #define GL_READ_WRITE_ARB 0x88BA
04366 #define GL_BUFFER_ACCESS_ARB 0x88BB
04367 #define GL_BUFFER_MAPPED_ARB 0x88BC
04368 #define GL_BUFFER_MAP_POINTER_ARB 0x88BD
04369 #define GL_STREAM_DRAW_ARB 0x88E0
04370 #define GL_STREAM_READ_ARB 0x88E1
04371 #define GL_STREAM_COPY_ARB 0x88E2
04372 #define GL_STATIC_DRAW_ARB 0x88E4
04373 #define GL_STATIC_READ_ARB 0x88E5
04374 #define GL_STATIC_COPY_ARB 0x88E6
04375 #define GL_DYNAMIC_DRAW_ARB 0x88E8
04376 #define GL_DYNAMIC_READ_ARB 0x88E9
04377 #define GL_DYNAMIC_COPY_ARB 0x88EA
04378 #define GL_COLOR_SUM_ARB 0x8458
04379 #define GL_VERTEX_PROGRAM_ARB 0x8620
04380 #define GL_VERTEX_ATTRIB_ARRAY_ENABLED_ARB 0x8622
04381 #define GL_VERTEX_ATTRIB_ARRAY_SIZE_ARB 0x8623
04382 #define GL_VERTEX_ATTRIB_ARRAY_STRIDE_ARB 0x8624
04383 #define GL_VERTEX_ATTRIB_ARRAY_TYPE_ARB 0x8625
04384 #define GL_CURRENT_VERTEX_ATTRIB_ARB 0x8626
04385 #define GL_VERTEX_PROGRAM_POINT_SIZE_ARB 0x8642
04386 #define GL_VERTEX_PROGRAM_TWO_SIDE_ARB 0x8643
04387 #define GL_VERTEX_ATTRIB_ARRAY_POINTER_ARB 0x8645
04388 #define GL_MAX_VERTEX_ATTRIBS_ARB 0x8869
04389 #define GL_VERTEX_ATTRIB_ARRAY_NORMALIZED_ARB 0x886A
04390 #define GL_PROGRAM_ADDRESS_REGISTERS_ARB 0x88B0
04391 #define GL_MAX_PROGRAM_ADDRESS_REGISTERS_ARB 0x88B1
04392 #define GL_PROGRAM_NATIVE_ADDRESS_REGISTERS_ARB 0x88B2
04393 #define GL_MAX_PROGRAM_NATIVE_ADDRESS_REGISTERS_ARB 0x88B3
04394 #define GL_VERTEX_SHADER_ARB 0x8B31
04395 #define GL_MAX_VERTEX_UNIFORM_COMPONENTS_ARB 0x8B4A
04396 #define GL_MAX_VARYING_FLOATS_ARB 0x8B4B
04397 #define GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS_ARB 0x8B4C
04398 #define GL_MAX_COMBINED_TEXTURE_IMAGE_UNITS_ARB 0x8B4D
04399 #define GL_OBJECT_ACTIVE_ATTRIBUTES_ARB 0x8B89
04400 #define GL_OBJECT_ACTIVE_ATTRIBUTE_MAX_LENGTH_ARB 0x8B8A
04401 #define GL_MAX_DRAW_BUFFERS_ATI 0x8824
04402 #define GL_DRAW_BUFFER0_ATI 0x8825
04403 #define GL_DRAW_BUFFER1_ATI 0x8826
04404 #define GL_DRAW_BUFFER2_ATI 0x8827
04405 #define GL_DRAW_BUFFER3_ATI 0x8828
04406 #define GL_DRAW_BUFFER4_ATI 0x8829
04407 #define GL_DRAW_BUFFER5_ATI 0x882A
04408 #define GL_DRAW_BUFFER6_ATI 0x882B
04409 #define GL_DRAW_BUFFER7_ATI 0x882C
04410 #define GL_DRAW_BUFFER8_ATI 0x882D
04411 #define GL_DRAW_BUFFER9_ATI 0x882E
04412 #define GL_DRAW_BUFFER10_ATI 0x882F
04413 #define GL_DRAW_BUFFER11_ATI 0x8830
04414 #define GL_DRAW_BUFFER12_ATI 0x8831
04415 #define GL_DRAW_BUFFER13_ATI 0x8832
04416 #define GL_DRAW_BUFFER14_ATI 0x8833
04417 #define GL_DRAW_BUFFER15_ATI 0x8834
04418 #define GL_ELEMENT_ARRAY_ATI 0x8768
04419 #define GL_ELEMENT_ARRAY_TYPE_ATI 0x8769
04420 #define GL_ELEMENT_ARRAY_POINTER_ATI 0x876A
04421 #define GL_BUMP_ROT_MATRIX_ATI 0x8775
04422 #define GL_BUMP_ROT_MATRIX_SIZE_ATI 0x8776
04423 #define GL_BUMP_NUM_TEX_UNITS_ATI 0x8777
04424 #define GL_BUMP_TEX_UNITS_ATI 0x8778
04425 #define GL_DUDV_ATI 0x8779
04426 #define GL_DU8DV8_ATI 0x877A
04427 #define GL_BUMP_ENVMAP_ATI 0x877B
04428 #define GL_BUMP_TARGET_ATI 0x877C
04429 #define GL_FRAGMENT_SHADER_ATI 0x8920
04430 #define GL_REG_0_ATI 0x8921
04431 #define GL_REG_1_ATI 0x8922
04432 #define GL_REG_2_ATI 0x8923
04433 #define GL_REG_3_ATI 0x8924
04434 #define GL_REG_4_ATI 0x8925
04435 #define GL_REG_5_ATI 0x8926
04436 #define GL_REG_6_ATI 0x8927
04437 #define GL_REG_7_ATI 0x8928
```

```
04438 #define GL_REG_8_ATI 0x8929
04439 #define GL_REG_9_ATI 0x892A
04440 #define GL_REG_10_ATI 0x892B
04441 #define GL_REG_11_ATI 0x892C
04442 #define GL_REG_12_ATI 0x892D
04443 #define GL_REG_13_ATI 0x892E
04444 #define GL_REG_14_ATI 0x892F
04445 #define GL_REG_15_ATI 0x8930
04446 #define GL_REG_16_ATI 0x8931
04447 #define GL_REG_17_ATI 0x8932
04448 #define GL_REG_18_ATI 0x8933
04449 #define GL_REG_19_ATI 0x8934
04450 #define GL_REG_20_ATI 0x8935
04451 #define GL_REG_21_ATI 0x8936
04452 #define GL_REG_22_ATI 0x8937
04453 #define GL_REG_23_ATI 0x8938
04454 #define GL_REG_24_ATI 0x8939
04455 #define GL_REG_25_ATI 0x893A
04456 #define GL_REG_26_ATI 0x893B
04457 #define GL_REG_27_ATI 0x893C
04458 #define GL_REG_28_ATI 0x893D
04459 #define GL_REG_29_ATI 0x893E
04460 #define GL_REG_30_ATI 0x893F
04461 #define GL_REG_31_ATI 0x8940
04462 #define GL_CON_0_ATI 0x8941
04463 #define GL_CON_1_ATI 0x8942
04464 #define GL_CON_2_ATI 0x8943
04465 #define GL_CON_3_ATI 0x8944
04466 #define GL_CON_4_ATI 0x8945
04467 #define GL_CON_5_ATI 0x8946
04468 #define GL_CON_6_ATI 0x8947
04469 #define GL_CON_7_ATI 0x8948
04470 #define GL_CON_8_ATI 0x8949
04471 #define GL_CON_9_ATI 0x894A
04472 #define GL_CON_10_ATI 0x894B
04473 #define GL_CON_11_ATI 0x894C
04474 #define GL_CON_12_ATI 0x894D
04475 #define GL_CON_13_ATI 0x894E
04476 #define GL_CON_14_ATI 0x894F
04477 #define GL_CON_15_ATI 0x8950
04478 #define GL_CON_16_ATI 0x8951
04479 #define GL_CON_17_ATI 0x8952
04480 #define GL_CON_18_ATI 0x8953
04481 #define GL_CON_19_ATI 0x8954
04482 #define GL_CON_20_ATI 0x8955
04483 #define GL_CON_21_ATI 0x8956
04484 #define GL_CON_22_ATI 0x8957
04485 #define GL_CON_23_ATI 0x8958
04486 #define GL_CON_24_ATI 0x8959
04487 #define GL_CON_25_ATI 0x895A
04488 #define GL_CON_26_ATI 0x895B
04489 #define GL_CON_27_ATI 0x895C
04490 #define GL_CON_28_ATI 0x895D
04491 #define GL_CON_29_ATI 0x895E
04492 #define GL_CON_30_ATI 0x895F
04493 #define GL_CON_31_ATI 0x8960
04494 #define GL_MOV_ATI 0x8961
04495 #define GL_ADD_ATI 0x8963
04496 #define GL_MUL_ATI 0x8964
04497 #define GL_SUB_ATI 0x8965
04498 #define GL_DOT3_ATI 0x8966
04499 #define GL_DOT4_ATI 0x8967
04500 #define GL_MAD_ATI 0x8968
04501 #define GL_LERP_ATI 0x8969
04502 #define GL_CND_ATI 0x896A
04503 #define GL_CND0_ATI 0x896B
04504 #define GL_DOT2_ADD_ATI 0x896C
04505 #define GL_SECONDARY_INTERPOLATOR_ATI 0x896D
04506 #define GL_NUM_FRAGMENT_REGISTERS_ATI 0x896E
04507 #define GL_NUM_FRAGMENT_CONSTANTS_ATI 0x896F
04508 #define GL_NUM_PASSES_ATI 0x8970
04509 #define GL_NUM_INSTRUCTIONS_PER_PASS_ATI 0x8971
04510 #define GL_NUM_INSTRUCTIONS_TOTAL_ATI 0x8972
04511 #define GL_NUM_INPUT_INTERPOLATOR_COMPONENTS_ATI 0x8973
04512 #define GL_NUM_LOOPBACK_COMPONENTS_ATI 0x8974
04513 #define GL_COLOR_ALPHA_PAIRING_ATI 0x8975
04514 #define GL_SWIZZLE_STR_ATI 0x8976
04515 #define GL_SWIZZLE_STQ_ATI 0x8977
04516 #define GL_SWIZZLE_STR_DR_ATI 0x8978
04517 #define GL_SWIZZLE_STQ_DQ_ATI 0x8979
04518 #define GL_SWIZZLE_STRQ_ATI 0x897A
04519 #define GL_SWIZZLE_STRQ_DQ_ATI 0x897B
04520 #define GL_RED_BIT_ATI 0x00000001
04521 #define GL_GREEN_BIT_ATI 0x00000002
04522 #define GL_BLUE_BIT_ATI 0x00000004
04523 #define GL_2X_BIT_ATI 0x00000001
04524 #define GL_4X_BIT_ATI 0x00000002
```

```
04525 #define GL_8X_BIT_ATI 0x00000004
04526 #define GL_HALF_BIT_ATI 0x00000008
04527 #define GL_QUARTER_BIT_ATI 0x00000010
04528 #define GL_EIGHTH_BIT_ATI 0x00000020
04529 #define GL_SATURATE_BIT_ATI 0x00000040
04530 #define GL_COMP_BIT_ATI 0x00000002
04531 #define GL_NEGATE_BIT_ATI 0x00000004
04532 #define GL_BIAS_BIT_ATI 0x00000008
04533 #define GL_VBO_FREE_MEMORY_ATI 0x87FB
04534 #define GL_TEXTURE_FREE_MEMORY_ATI 0x87FC
04535 #define GL_RENDERBUFFER_FREE_MEMORY_ATI 0x87FD
04536 #define GL_RGBA_FLOAT_MODE_ATI 0x8820
04537 #define GL_COLOR_CLEAR_UNCLAMPED_VALUE_ATI 0x8835
04538 #define GL_PN_TRIANGLES_ATI 0x87F0
04539 #define GL_MAX_PN_TRIANGLES_TESSELATION_LEVEL_ATI 0x87F1
04540 #define GL_PN_TRIANGLES_POINT_MODE_ATI 0x87F2
04541 #define GL_PN_TRIANGLES_NORMAL_MODE_ATI 0x87F3
04542 #define GL_PN_TRIANGLES_TESSELATION_LEVEL_ATI 0x87F4
04543 #define GL_PN_TRIANGLES_POINT_MODE_LINEAR_ATI 0x87F5
04544 #define GL_PN_TRIANGLES_POINT_MODE_CUBIC_ATI 0x87F6
04545 #define GL_PN_TRIANGLES_NORMAL_MODE_LINEAR_ATI 0x87F7
04546 #define GL_PN_TRIANGLES_NORMAL_MODE_QUADRATIC_ATI 0x87F8
04547 #define GL_STENCIL_BACK_FUNC_ATI 0x8800
04548 #define GL_STENCIL_BACK_FAIL_ATI 0x8801
04549 #define GL_STENCIL_BACK_PASS_DEPTH_FAIL_ATI 0x8802
04550 #define GL_STENCIL_BACK_PASS_DEPTH_PASS_ATI 0x8803
04551 #define GL_TEXT_FRAGMENT_SHADER_ATI 0x8200
04552 #define GL_MODULATE_ADD_ATI 0x8744
04553 #define GL_MODULATE_SIGNED_ADD_ATI 0x8745
04554 #define GL_MODULATE_SUBTRACT_ATI 0x8746
04555 #define GL_RGBA_FLOAT32_ATI 0x8814
04556 #define GL_RGB_FLOAT32_ATI 0x8815
04557 #define GL_ALPHA_FLOAT32_ATI 0x8816
04558 #define GL_INTENSITY_FLOAT32_ATI 0x8817
04559 #define GL_LUMINANCE_FLOAT32_ATI 0x8818
04560 #define GL_LUMINANCE_ALPHA_FLOAT32_ATI 0x8819
04561 #define GL_RGBA_FLOAT16_ATI 0x881A
04562 #define GL_RGB_FLOAT16_ATI 0x881B
04563 #define GL_ALPHA_FLOAT16_ATI 0x881C
04564 #define GL_INTENSITY_FLOAT16_ATI 0x881D
04565 #define GL_LUMINANCE_FLOAT16_ATI 0x881E
04566 #define GL_LUMINANCE_ALPHA_FLOAT16_ATI 0x881F
04567 #define GL_MIRROR_CLAMP_ATI 0x8742
04568 #define GL_MIRROR_CLAMP_TO_EDGE_ATI 0x8743
04569 #define GL_STATIC_ATI 0x8760
04570 #define GL_DYNAMIC_ATI 0x8761
04571 #define GL_PRESERVE_ATI 0x8762
04572 #define GL_DISCARD_ATI 0x8763
04573 #define GL_OBJECT_BUFFER_SIZE_ATI 0x8764
04574 #define GL_OBJECT_BUFFER_USAGE_ATI 0x8765
04575 #define GL_ARRAY_OBJECT_BUFFER_ATI 0x8766
04576 #define GL_ARRAY_OBJECT_OFFSET_ATI 0x8767
04577 #define GL_MAX_VERTEX_STREAMS_ATI 0x876B
04578 #define GL_VERTEX_STREAM0_ATI 0x876C
04579 #define GL_VERTEX_STREAM1_ATI 0x876D
04580 #define GL_VERTEX_STREAM2_ATI 0x876E
04581 #define GL_VERTEX_STREAM3_ATI 0x876F
04582 #define GL_VERTEX_STREAM4_ATI 0x8770
04583 #define GL_VERTEX_STREAM5_ATI 0x8771
04584 #define GL_VERTEX_STREAM6_ATI 0x8772
04585 #define GL_VERTEX_STREAM7_ATI 0x8773
04586 #define GL_VERTEX_SOURCE_ATI 0x8774
04587 #define GL_422_EXT 0x80CC
04588 #define GL_422_REV_EXT 0x80CD
04589 #define GL_422_AVERAGE_EXT 0x80CE
04590 #define GL_422_REV_AVERAGE_EXT 0x80CF
04591 #define GL_ABGR_EXT 0x8000
04592 #define GL_BGR_EXT 0x80E0
04593 #define GL_BGRA_EXT 0x80E1
04594 #define GL_MAX_VERTEX_BINDABLE_UNIFORMS_EXT 0x8DE2
04595 #define GL_MAX_FRAGMENT_BINDABLE_UNIFORMS_EXT 0x8DE3
04596 #define GL_MAX_GEOMETRY_BINDABLE_UNIFORMS_EXT 0x8DE4
04597 #define GL_MAX_BINDABLE_UNIFORM_SIZE_EXT 0x8DED
04598 #define GL_UNIFORM_BUFFER_EXT 0x8DEE
04599 #define GL_UNIFORM_BUFFER_BINDING_EXT 0x8DEF
04600 #define GL_CONSTANT_COLOR_EXT 0x8001
04601 #define GL_ONE_MINUS_CONSTANT_COLOR_EXT 0x8002
04602 #define GL_CONSTANT_ALPHA_EXT 0x8003
04603 #define GL_ONE_MINUS_CONSTANT_ALPHA_EXT 0x8004
04604 #define GL_BLEND_COLOR_EXT 0x8005
04605 #define GL_BLEND_EQUATION_RGB_EXT 0x8009
04606 #define GL_BLEND_EQUATION_ALPHA_EXT 0x883D
04607 #define GL_BLEND_DST_RGB_EXT 0x80C8
04608 #define GL_BLEND_SRC_RGB_EXT 0x80C9
04609 #define GL_BLEND_DST_ALPHA_EXT 0x80CA
04610 #define GL_BLEND_SRC_ALPHA_EXT 0x80CB
04611 #define GL_MIN_EXT 0x8007
```

```
04612 #define GL_MAX_EXT 0x8008
04613 #define GL_FUNC_ADD_EXT 0x8006
04614 #define GL_BLEND_EQUATION_EXT 0x8009
04615 #define GL_FUNC_SUBTRACT_EXT 0x800A
04616 #define GL_FUNC_REVERSE_SUBTRACT_EXT 0x800B
04617 #define GL_CLIP_VOLUME_CLIPPING_HINT_EXT 0x80F0
04618 #define GL_CMYK_EXT 0x800C
04619 #define GL_CMYKA_EXT 0x800D
04620 #define GL_PACK_CMYK_HINT_EXT 0x800E
04621 #define GL_UNPACK_CMYK_HINT_EXT 0x800F
04622 #define GL_ARRAY_ELEMENT_LOCK_FIRST_EXT 0x81A8
04623 #define GL_ARRAY_ELEMENT_LOCK_COUNT_EXT 0x81A9
04624 #define GL_CONVOLUTION_1D_EXT 0x8010
04625 #define GL_CONVOLUTION_2D_EXT 0x8011
04626 #define GL_SEPARABLE_2D_EXT 0x8012
04627 #define GL_CONVOLUTION_BORDER_MODE_EXT 0x8013
04628 #define GL_CONVOLUTION_FILTER_SCALE_EXT 0x8014
04629 #define GL_CONVOLUTION_FILTER_BIAS_EXT 0x8015
04630 #define GL_REDUCE_EXT 0x8016
04631 #define GL_CONVOLUTION_FORMAT_EXT 0x8017
04632 #define GL_CONVOLUTION_WIDTH_EXT 0x8018
04633 #define GL_CONVOLUTION_HEIGHT_EXT 0x8019
04634 #define GL_MAX_CONVOLUTION_WIDTH_EXT 0x801A
04635 #define GL_MAX_CONVOLUTION_HEIGHT_EXT 0x801B
04636 #define GL_POST_CONVOLUTION_RED_SCALE_EXT 0x801C
04637 #define GL_POST_CONVOLUTION_GREEN_SCALE_EXT 0x801D
04638 #define GL_POST_CONVOLUTION_BLUE_SCALE_EXT 0x801E
04639 #define GL_POST_CONVOLUTION_ALPHA_SCALE_EXT 0x801F
04640 #define GL_POST_CONVOLUTION_RED_BIAS_EXT 0x8020
04641 #define GL_POST_CONVOLUTION_GREEN_BIAS_EXT 0x8021
04642 #define GL_POST_CONVOLUTION_BLUE_BIAS_EXT 0x8022
04643 #define GL_POST_CONVOLUTION_ALPHA_BIAS_EXT 0x8023
04644 #define GL_TANGENT_ARRAY_EXT 0x8439
04645 #define GL_BINORMAL_ARRAY_EXT 0x843A
04646 #define GL_CURRENT_TANGENT_EXT 0x843B
04647 #define GL_CURRENT_BINORMAL_EXT 0x843C
04648 #define GL_TANGENT_ARRAY_TYPE_EXT 0x843E
04649 #define GL_TANGENT_ARRAY_STRIDE_EXT 0x843F
04650 #define GL_BINORMAL_ARRAY_TYPE_EXT 0x8440
04651 #define GL_BINORMAL_ARRAY_STRIDE_EXT 0x8441
04652 #define GL_TANGENT_ARRAY_POINTER_EXT 0x8442
04653 #define GL_BINORMAL_ARRAY_POINTER_EXT 0x8443
04654 #define GL_MAP1_TANGENT_EXT 0x8444
04655 #define GL_MAP2_TANGENT_EXT 0x8445
04656 #define GL_MAP1_BINORMAL_EXT 0x8446
04657 #define GL_MAP2_BINORMAL_EXT 0x8447
04658 #define GL_CULL_VERTEX_EXT 0x81AA
04659 #define GL_CULL_VERTEX_EYE_POSITION_EXT 0x81AB
04660 #define GL_CULL_VERTEX_OBJECT_POSITION_EXT 0x81AC
04661 #define GL_PROGRAM_PIPELINE_OBJECT_EXT 0x8A4F
04662 #define GL_PROGRAM_OBJECT_EXT 0x8B40
04663 #define GL_SHADER_OBJECT_EXT 0x8B48
04664 #define GL_BUFFER_OBJECT_EXT 0x9151
04665 #define GL_QUERY_OBJECT_EXT 0x9153
04666 #define GL_VERTEX_ARRAY_OBJECT_EXT 0x9154
04667 #define GL_SAMPLER 0x82E6
04668 #define GL_DEPTH_BOUNDS_TEST_EXT 0x8890
04669 #define GL_DEPTH_BOUNDS_EXT 0x8891
04670 #define GL_PROGRAM_MATRIX_EXT 0x8E2D
04671 #define GL_TRANSPOSE_PROGRAM_MATRIX_EXT 0x8E2E
04672 #define GL_PROGRAM_MATRIX_STACK_DEPTH_EXT 0x8E2F
04673 #define GL_MAX_ELEMENTS_VERTICES_EXT 0x80E8
04674 #define GL_MAX_ELEMENTS_INDICES_EXT 0x80E9
04675 #define GL_FOG_COORDINATE_SOURCE_EXT 0x8450
04676 #define GL_FOG_COORDINATE_EXT 0x8451
04677 #define GL_FRAGMENT_DEPTH_EXT 0x8452
04678 #define GL_CURRENT_FOG_COORDINATE_EXT 0x8453
04679 #define GL_FOG_COORDINATE_ARRAY_TYPE_EXT 0x8454
04680 #define GL_FOG_COORDINATE_ARRAY_STRIDE_EXT 0x8455
04681 #define GL_FOG_COORDINATE_ARRAY_POINTER_EXT 0x8456
04682 #define GL_FOG_COORDINATE_ARRAY_EXT 0x8457
04683 #define GL_READ_FRAMEBUFFER_EXT 0x8CA8
04684 #define GL_DRAW_FRAMEBUFFER_EXT 0x8CA9
04685 #define GL_DRAW_FRAMEBUFFER_BINDING_EXT 0x8CA6
04686 #define GL_READ_FRAMEBUFFER_BINDING_EXT 0x8CAA
04687 #define GL_RENDERBUFFER_SAMPLES_EXT 0x8CAB
04688 #define GL_FRAMEBUFFER_INCOMPLETE_MULTISAMPLE_EXT 0x8D56
04689 #define GL_MAX_SAMPLES_EXT 0x8D57
04690 #define GL_SCALED_RESOLVE_FASTEST_EXT 0x90BA
04691 #define GL_SCALED_RESOLVE_NICEST_EXT 0x90BB
04692 #define GL_INVALID_FRAMEBUFFER_OPERATION_EXT 0x0506
04693 #define GL_MAX_RENDERBUFFER_SIZE_EXT 0x84E8
04694 #define GL_FRAMEBUFFER_BINDING_EXT 0x8CA6
04695 #define GL_RENDERBUFFER_BINDING_EXT 0x8CA7
04696 #define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE_EXT 0x8CD0
04697 #define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_NAME_EXT 0x8CD1
04698 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL_EXT 0x8CD2
```



```
04699 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE_EXT 0x8CD3
04700 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_3D_ZOFFSET_EXT 0x8CD4
04701 #define GL_FRAMEBUFFER_COMPLETE_EXT 0x8CD5
04702 #define GL_FRAMEBUFFER_INCOMPLETE_ATTACHMENT_EXT 0x8CD6
04703 #define GL_FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT_EXT 0x8CD7
04704 #define GL_FRAMEBUFFER_INCOMPLETE_DIMENSIONS_EXT 0x8CD9
04705 #define GL_FRAMEBUFFER_INCOMPLETE_FORMATS_EXT 0x8CDA
04706 #define GL_FRAMEBUFFER_INCOMPLETE_DRAW_BUFFER_EXT 0x8CDB
04707 #define GL_FRAMEBUFFER_INCOMPLETE_READ_BUFFER_EXT 0x8CDC
04708 #define GL_FRAMEBUFFER_UNSUPPORTED_EXT 0x8CDD
04709 #define GL_MAX_COLOR_ATTACHMENTS_EXT 0x8CDF
04710 #define GL_COLOR_ATTACHMENT0_EXT 0x8CE0
04711 #define GL_COLOR_ATTACHMENT1_EXT 0x8CE1
04712 #define GL_COLOR_ATTACHMENT2_EXT 0x8CE2
04713 #define GL_COLOR_ATTACHMENT3_EXT 0x8CE3
04714 #define GL_COLOR_ATTACHMENT4_EXT 0x8CE4
04715 #define GL_COLOR_ATTACHMENT5_EXT 0x8CE5
04716 #define GL_COLOR_ATTACHMENT6_EXT 0x8CE6
04717 #define GL_COLOR_ATTACHMENT7_EXT 0x8CE7
04718 #define GL_COLOR_ATTACHMENT8_EXT 0x8CE8
04719 #define GL_COLOR_ATTACHMENT9_EXT 0x8CE9
04720 #define GL_COLOR_ATTACHMENT10_EXT 0x8CEA
04721 #define GL_COLOR_ATTACHMENT11_EXT 0x8CEB
04722 #define GL_COLOR_ATTACHMENT12_EXT 0x8CEC
04723 #define GL_COLOR_ATTACHMENT13_EXT 0x8CED
04724 #define GL_COLOR_ATTACHMENT14_EXT 0x8CEE
04725 #define GL_COLOR_ATTACHMENT15_EXT 0x8CEF
04726 #define GL_DEPTH_ATTACHMENT_EXT 0x8D00
04727 #define GL_STENCIL_ATTACHMENT_EXT 0x8D20
04728 #define GL_FRAMEBUFFER_EXT 0x8D40
04729 #define GL_RENDERBUFFER_EXT 0x8D41
04730 #define GL_RENDERBUFFER_WIDTH_EXT 0x8D42
04731 #define GL_RENDERBUFFER_HEIGHT_EXT 0x8D43
04732 #define GL_RENDERBUFFER_INTERNAL_FORMAT_EXT 0x8D44
04733 #define GL_STENCIL_INDEX1_EXT 0x8D46
04734 #define GL_STENCIL_INDEX4_EXT 0x8D47
04735 #define GL_STENCIL_INDEX8_EXT 0x8D48
04736 #define GL_STENCIL_INDEX16_EXT 0x8D49
04737 #define GL_RENDERBUFFER_RED_SIZE_EXT 0x8D50
04738 #define GL_RENDERBUFFER_GREEN_SIZE_EXT 0x8D51
04739 #define GL_RENDERBUFFER_BLUE_SIZE_EXT 0x8D52
04740 #define GL_RENDERBUFFER_ALPHA_SIZE_EXT 0x8D53
04741 #define GL_RENDERBUFFER_DEPTH_SIZE_EXT 0x8D54
04742 #define GL_RENDERBUFFER_STENCIL_SIZE_EXT 0x8D55
04743 #define GL_FRAMEBUFFER_SRGB_EXT 0x8DB9
04744 #define GL_FRAMEBUFFER_SRGB_CAPABLE_EXT 0x8DBA
04745 #define GL_GEOMETRY_SHADER_EXT 0x8DD9
04746 #define GL_GEOMETRY_VERTICES_OUT_EXT 0x8DDA
04747 #define GL_GEOMETRY_INPUT_TYPE_EXT 0x8ddb
04748 #define GL_GEOMETRY_OUTPUT_TYPE_EXT 0x8DDC
04749 #define GL_MAX_GEOMETRY_TEXTURE_IMAGE_UNITS_EXT 0x8C29
04750 #define GL_MAX_GEOMETRY_VARYING_COMPONENTS_EXT 0x8DDD
04751 #define GL_MAX_VERTEX_VARYING_COMPONENTS_EXT 0x8DDE
04752 #define GL_MAX_VARYING_COMPONENTS_EXT 0x8B4B
04753 #define GL_MAX_GEOMETRY_UNIFORM_COMPONENTS_EXT 0x8DDF
04754 #define GL_MAX_GEOMETRY_OUTPUT_VERTICES_EXT 0x8DE0
04755 #define GL_MAX_GEOMETRY_TOTAL_OUTPUT_COMPONENTS_EXT 0x8DE1
04756 #define GL_LINES_ADJACENCY_EXT 0x000A
04757 #define GL_LINE_STRIP_ADJACENCY_EXT 0x000B
04758 #define GL_TRIANGLES_ADJACENCY_EXT 0x000C
04759 #define GL_TRIANGLE_STRIP_ADJACENCY_EXT 0x000D
04760 #define GL_FRAMEBUFFER_INCOMPLETE_LAYER_TARGETS_EXT 0x8DA8
04761 #define GL_FRAMEBUFFER_INCOMPLETE_LAYER_COUNT_EXT 0x8DA9
04762 #define GL_FRAMEBUFFER_ATTACHMENT_LAYERED_EXT 0x8DA7
04763 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LAYER_EXT 0x8CD4
04764 #define GL_PROGRAM_POINT_SIZE_EXT 0x8642
04765 #define GL_SAMPLER_1D_ARRAY_EXT 0x8DC0
04766 #define GL_SAMPLER_2D_ARRAY_EXT 0x8DC1
04767 #define GL_SAMPLER_BUFFER_EXT 0x8DC2
04768 #define GL_SAMPLER_1D_ARRAY_SHADOW_EXT 0x8DC3
04769 #define GL_SAMPLER_2D_ARRAY_SHADOW_EXT 0x8DC4
04770 #define GL_SAMPLER_CUBE_SHADOW_EXT 0x8DC5
04771 #define GL_UNSIGNED_INT_VEC2_EXT 0x8DC6
04772 #define GL_UNSIGNED_INT_VEC3_EXT 0x8DC7
04773 #define GL_UNSIGNED_INT_VEC4_EXT 0x8DC8
04774 #define GL_INT_SAMPLER_1D_EXT 0x8DC9
04775 #define GL_INT_SAMPLER_2D_EXT 0x8DCA
04776 #define GL_INT_SAMPLER_3D_EXT 0x8DCB
04777 #define GL_INT_SAMPLER_CUBE_EXT 0x8DCC
04778 #define GL_INT_SAMPLER_2D_RECT_EXT 0x8DCD
04779 #define GL_INT_SAMPLER_1D_ARRAY_EXT 0x8DCE
04780 #define GL_INT_SAMPLER_2D_ARRAY_EXT 0x8DCF
04781 #define GL_INT_SAMPLER_BUFFER_EXT 0x8DD0
04782 #define GL_UNSIGNED_INT_SAMPLER_1D_EXT 0x8DD1
04783 #define GL_UNSIGNED_INT_SAMPLER_2D_EXT 0x8DD2
04784 #define GL_UNSIGNED_INT_SAMPLER_3D_EXT 0x8DD3
04785 #define GL_UNSIGNED_INT_SAMPLER_CUBE_EXT 0x8DD4
```

```
04786 #define GL_UNSIGNED_INT_SAMPLER_2D_RECT_EXT 0x8DD5
04787 #define GL_UNSIGNED_INT_SAMPLER_1D_ARRAY_EXT 0x8DD6
04788 #define GL_UNSIGNED_INT_SAMPLER_2D_ARRAY_EXT 0x8DD7
04789 #define GL_UNSIGNED_INT_SAMPLER_BUFFER_EXT 0x8DD8
04790 #define GL_MIN_PROGRAM_TEXEL_OFFSET_EXT 0x8904
04791 #define GL_MAX_PROGRAM_TEXEL_OFFSET_EXT 0x8905
04792 #define GL_VERTEX_ATTRIB_ARRAY_INTEGER_EXT 0x88FD
04793 #define GL_HISTOGRAM_EXT 0x8024
04794 #define GL_PROXY_HISTOGRAM_EXT 0x8025
04795 #define GL_HISTOGRAM_WIDTH_EXT 0x8026
04796 #define GL_HISTOGRAM_FORMAT_EXT 0x8027
04797 #define GL_HISTOGRAM_RED_SIZE_EXT 0x8028
04798 #define GL_HISTOGRAM_GREEN_SIZE_EXT 0x8029
04799 #define GL_HISTOGRAM_BLUE_SIZE_EXT 0x802A
04800 #define GL_HISTOGRAM_ALPHA_SIZE_EXT 0x802B
04801 #define GL_HISTOGRAM_LUMINANCE_SIZE_EXT 0x802C
04802 #define GL_HISTOGRAM_SINK_EXT 0x802D
04803 #define GL_MINMAX_EXT 0x802E
04804 #define GL_MINMAX_FORMAT_EXT 0x802F
04805 #define GL_MINMAX_SINK_EXT 0x8030
04806 #define GL_TABLE_TOO_LARGE_EXT 0x8031
04807 #define GL_IUI_V2F_EXT 0x81AD
04808 #define GL_IUI_V3F_EXT 0x81AE
04809 #define GL_IUI_N3F_V2F_EXT 0x81AF
04810 #define GL_IUI_N3F_V3F_EXT 0x81B0
04811 #define GL_T2F_IUI_V2F_EXT 0x81B1
04812 #define GL_T2F_IUI_V3F_EXT 0x81B2
04813 #define GL_T2F_IUI_N3F_V2F_EXT 0x81B3
04814 #define GL_T2F_IUI_N3F_V3F_EXT 0x81B4
04815 #define GL_INDEX_TEST_EXT 0x81B5
04816 #define GL_INDEX_TEST_FUNC_EXT 0x81B6
04817 #define GL_INDEX_TEST_REF_EXT 0x81B7
04818 #define GL_INDEX_MATERIAL_EXT 0x81B8
04819 #define GL_INDEX_MATERIAL_PARAMETER_EXT 0x81B9
04820 #define GL_INDEX_MATERIAL_FACE_EXT 0x81BA
04821 #define GL_FRAGMENT_MATERIAL_EXT 0x8349
04822 #define GL_FRAGMENT_NORMAL_EXT 0x834A
04823 #define GL_FRAGMENT_COLOR_EXT 0x834C
04824 #define GL_ATTENUATION_EXT 0x834D
04825 #define GL_SHADOW_ATTENUATION_EXT 0x834E
04826 #define GL_TEXTURE_APPLICATION_MODE_EXT 0x834F
04827 #define GL_TEXTURE_LIGHT_EXT 0x8350
04828 #define GL_TEXTURE_MATERIAL_FACE_EXT 0x8351
04829 #define GL_TEXTURE_MATERIAL_PARAMETER_EXT 0x8352
04830 #define GL_TEXTURE_TILING_EXT 0x9580
04831 #define GL_DEDICATED_MEMORY_OBJECT_EXT 0x9581
04832 #define GL_PROTECTED_MEMORY_OBJECT_EXT 0x959B
04833 #define GL_NUM_TILING_TYPES_EXT 0x9582
04834 #define GL_TILING_TYPES_EXT 0x9583
04835 #define GL_OPTIMAL_TILING_EXT 0x9584
04836 #define GL_LINEAR_TILING_EXT 0x9585
04837 #define GL_NUM_DEVICE_UUIDS_EXT 0x9596
04838 #define GL_DEVICE_UUID_EXT 0x9597
04839 #define GL_DRIVER_UUID_EXT 0x9598
04840 #define GL_UUID_SIZE_EXT 16
04841 #define GL_HANDLE_TYPE_OPAQUE_FD_EXT 0x9586
04842 #define GL_HANDLE_TYPE_OPAQUE_WIN32_EXT 0x9587
04843 #define GL_HANDLE_TYPE_OPAQUE_WIN32_KMT_EXT 0x9588
04844 #define GL_DEVICE_LUID_EXT 0x9599
04845 #define GL_DEVICE_NODE_MASK_EXT 0x959A
04846 #define GL_LUID_SIZE_EXT 8
04847 #define GL_HANDLE_TYPE_D3D12_TILEPOOL_EXT 0x9589
04848 #define GL_HANDLE_TYPE_D3D12_RESOURCE_EXT 0x958A
04849 #define GL_HANDLE_TYPE_D3D11_IMAGE_EXT 0x958B
04850 #define GL_HANDLE_TYPE_D3D11_IMAGE_KMT_EXT 0x958C
04851 #define GL_MULTISAMPLE_EXT 0x809D
04852 #define GL_SAMPLE_ALPHA_TO_MASK_EXT 0x809E
04853 #define GL_SAMPLE_ALPHA_TO_ONE_EXT 0x809F
04854 #define GL_SAMPLE_MASK_EXT 0x80A0
04855 #define GL_1PASS_EXT 0x80A1
04856 #define GL_2PASS_0_EXT 0x80A2
04857 #define GL_2PASS_1_EXT 0x80A3
04858 #define GL_4PASS_0_EXT 0x80A4
04859 #define GL_4PASS_1_EXT 0x80A5
04860 #define GL_4PASS_2_EXT 0x80A6
04861 #define GL_4PASS_3_EXT 0x80A7
04862 #define GL_SAMPLE_BUFFERS_EXT 0x80A8
04863 #define GL_SAMPLES_EXT 0x80A9
04864 #define GL_SAMPLE_MASK_VALUE_EXT 0x80AA
04865 #define GL_SAMPLE_MASK_INVERT_EXT 0x80AB
04866 #define GL_SAMPLE_PATTERN_EXT 0x80AC
04867 #define GL_MULTISAMPLE_BIT_EXT 0x20000000
04868 #define GL_DEPTH_STENCIL_EXT 0x84F9
04869 #define GL_UNSIGNED_INT_24_8_EXT 0x84FA
04870 #define GL_DEPTH24_STENCIL8_EXT 0x88F0
04871 #define GL_TEXTURE_STENCIL_SIZE_EXT 0x88F1
04872 #define GL_R11F_G11F_B10F_EXT 0x8C3A
```



```
04873 #define GL_UNSIGNED_INT_10F_11F_11F_REV_EXT 0x8C3B
04874 #define GL_RGBA_SIGNED_COMPONENTS_EXT 0x8C3C
04875 #define GL_UNSIGNED_BYTE_3_3_2_EXT 0x8032
04876 #define GL_UNSIGNED_SHORT_4_4_4_4_EXT 0x8033
04877 #define GL_UNSIGNED_SHORT_5_5_5_1_EXT 0x8034
04878 #define GL_UNSIGNED_INT_8_8_8_8_EXT 0x8035
04879 #define GL_UNSIGNED_INT_10_10_10_2_EXT 0x8036
04880 #define GL_COLOR_INDEX1_EXT 0x80E2
04881 #define GL_COLOR_INDEX2_EXT 0x80E3
04882 #define GL_COLOR_INDEX4_EXT 0x80E4
04883 #define GL_COLOR_INDEX8_EXT 0x80E5
04884 #define GL_COLOR_INDEX12_EXT 0x80E6
04885 #define GL_COLOR_INDEX16_EXT 0x80E7
04886 #define GL_TEXTURE_INDEX_SIZE_EXT 0x80ED
04887 #define GL_PIXEL_PACK_BUFFER_EXT 0x88EB
04888 #define GL_PIXEL_UNPACK_BUFFER_EXT 0x88EC
04889 #define GL_PIXEL_PACK_BUFFER_BINDING_EXT 0x88ED
04890 #define GL_PIXEL_UNPACK_BUFFER_BINDING_EXT 0x88EF
04891 #define GL_PIXEL_TRANSFORM_2D_EXT 0x8330
04892 #define GL_PIXEL_MAG_FILTER_EXT 0x8331
04893 #define GL_PIXEL_MIN_FILTER_EXT 0x8332
04894 #define GL_PIXEL_CUBIC_WEIGHT_EXT 0x8333
04895 #define GL_CUBIC_EXT 0x8334
04896 #define GL_AVERAGE_EXT 0x8335
04897 #define GL_PIXEL_TRANSFORM_2D_STACK_DEPTH_EXT 0x8336
04898 #define GL_MAX_PIXEL_TRANSFORM_2D_STACK_DEPTH_EXT 0x8337
04899 #define GL_PIXEL_TRANSFORM_2D_MATRIX_EXT 0x8338
04900 #define GL_POINT_SIZE_MIN_EXT 0x8126
04901 #define GL_POINT_SIZE_MAX_EXT 0x8127
04902 #define GL_POINT_FADE_THRESHOLD_SIZE_EXT 0x8128
04903 #define GL_DISTANCE_ATTENUATION_EXT 0x8129
04904 #define GL_POLYGON_OFFSET_EXT 0x8037
04905 #define GL_POLYGON_OFFSET_FACTOR_EXT 0x8038
04906 #define GL_POLYGON_OFFSET_BIAS_EXT 0x8039
04907 #define GL_POLYGON_OFFSET_CLAMP_EXT 0x8E1B
04908 #define GL_QUADS_FOLLOW_PROVOKING_VERTEX_CONVENTION_EXT 0x8E4C
04909 #define GL_FIRST_VERTEX_CONVENTION_EXT 0x8E4D
04910 #define GL_LAST_VERTEX_CONVENTION_EXT 0x8E4E
04911 #define GL_PROVOKING_VERTEX_EXT 0x8E4F
04912 #define GL_RASTER_MULTISAMPLE_EXT 0x9327
04913 #define GL_RASTER_SAMPLES_EXT 0x9328
04914 #define GL_MAX_RASTER_SAMPLES_EXT 0x9329
04915 #define GL_RASTER_FIXED_SAMPLE_LOCATIONS_EXT 0x932A
04916 #define GL_MULTISAMPLE_RASTERIZATION_ALLOWED_EXT 0x932B
04917 #define GL_EFFECTIVE_RASTER_SAMPLES_EXT 0x932C
04918 #define GL_RESCALE_NORMAL_EXT 0x803A
04919 #define GL_COLOR_SUM_EXT 0x8458
04920 #define GL_CURRENT_SECONDARY_COLOR_EXT 0x8459
04921 #define GL_SECONDARY_COLOR_ARRAY_SIZE_EXT 0x845A
04922 #define GL_SECONDARY_COLOR_ARRAY_TYPE_EXT 0x845B
04923 #define GL_SECONDARY_COLOR_ARRAY_STRIDE_EXT 0x845C
04924 #define GL_SECONDARY_COLOR_ARRAY_POINTER_EXT 0x845D
04925 #define GL_SECONDARY_COLOR_ARRAY_EXT 0x845E
04926 #define GL_LAYOUT_GENERAL_EXT 0x958D
04927 #define GL_LAYOUT_COLOR_ATTACHMENT_EXT 0x958E
04928 #define GL_LAYOUT_DEPTH_STENCIL_ATTACHMENT_EXT 0x958F
04929 #define GL_LAYOUT_DEPTH_STENCIL_READ_ONLY_EXT 0x9590
04930 #define GL_LAYOUT_SHADER_READ_ONLY_EXT 0x9591
04931 #define GL_LAYOUT_TRANSFER_SRC_EXT 0x9592
04932 #define GL_LAYOUT_TRANSFER_DST_EXT 0x9593
04933 #define GL_LAYOUT_DEPTH_READ_ONLY_STENCIL_ATTACHMENT_EXT 0x9530
04934 #define GL_LAYOUT_DEPTH_ATTACHMENT_STENCIL_READ_ONLY_EXT 0x9531
04935 #define GL_HANDLE_TYPE_D3D12_FENCE_EXT 0x9594
04936 #define GL_D3D12_FENCE_VALUE_EXT 0x9595
04937 #define GL_ACTIVE_PROGRAM_EXT 0x8B8D
04938 #define GL_VERTEX_SHADER_BIT_EXT 0x00000001
04939 #define GL_FRAGMENT_SHADER_BIT_EXT 0x00000002
04940 #define GL_ALL_SHADER_BITS_EXT 0xFFFFFFFF
04941 #define GL_PROGRAM_SEPARABLE_EXT 0x8258
04942 #define GL_PROGRAM_PIPELINE_BINDING_EXT 0x825A
04943 #define GL_LIGHT_MODEL_COLOR_CONTROL_EXT 0x81F8
04944 #define GL_SINGLE_COLOR_EXT 0x81F9
04945 #define GL_SEPARATE_SPECULAR_COLOR_EXT 0x81FA
04946 #define GL_FRAGMENT_SHADER_DISCARDS_SAMPLES_EXT 0x8A52
04947 #define GL_MAX_IMAGE_UNITS_EXT 0x8F38
04948 #define GL_MAX_COMBINED_IMAGE_UNITS_AND_FRAGMENT_OUTPUTS_EXT 0x8F39
04949 #define GL_IMAGE_BINDING_NAME_EXT 0x8F3A
04950 #define GL_IMAGE_BINDING_LEVEL_EXT 0x8F3B
04951 #define GL_IMAGE_BINDING_LAYERED_EXT 0x8F3C
04952 #define GL_IMAGE_BINDING_LAYER_EXT 0x8F3D
04953 #define GL_IMAGE_BINDING_ACCESS_EXT 0x8F3E
04954 #define GL_IMAGE_1D_EXT 0x904C
04955 #define GL_IMAGE_2D_EXT 0x904D
04956 #define GL_IMAGE_3D_EXT 0x904E
04957 #define GL_IMAGE_2D_RECT_EXT 0x904F
04958 #define GL_IMAGE_CUBE_EXT 0x9050
04959 #define GL_IMAGE_BUFFER_EXT 0x9051
```

```
04960 #define GL_IMAGE_1D_ARRAY_EXT 0x9052
04961 #define GL_IMAGE_2D_ARRAY_EXT 0x9053
04962 #define GL_IMAGE_CUBE_MAP_ARRAY_EXT 0x9054
04963 #define GL_IMAGE_2D_MULTISAMPLE_EXT 0x9055
04964 #define GL_IMAGE_2D_MULTISAMPLE_ARRAY_EXT 0x9056
04965 #define GL_INT_IMAGE_1D_EXT 0x9057
04966 #define GL_INT_IMAGE_2D_EXT 0x9058
04967 #define GL_INT_IMAGE_3D_EXT 0x9059
04968 #define GL_INT_IMAGE_2D_RECT_EXT 0x905A
04969 #define GL_INT_IMAGE_CUBE_EXT 0x905B
04970 #define GL_INT_IMAGE_BUFFER_EXT 0x905C
04971 #define GL_INT_IMAGE_1D_ARRAY_EXT 0x905D
04972 #define GL_INT_IMAGE_2D_ARRAY_EXT 0x905E
04973 #define GL_INT_IMAGE_CUBE_MAP_ARRAY_EXT 0x905F
04974 #define GL_INT_IMAGE_2D_MULTISAMPLE_EXT 0x9060
04975 #define GL_INT_IMAGE_2D_MULTISAMPLE_ARRAY_EXT 0x9061
04976 #define GL_UNSIGNED_INT_IMAGE_1D_EXT 0x9062
04977 #define GL_UNSIGNED_INT_IMAGE_2D_EXT 0x9063
04978 #define GL_UNSIGNED_INT_IMAGE_3D_EXT 0x9064
04979 #define GL_UNSIGNED_INT_IMAGE_2D_RECT_EXT 0x9065
04980 #define GL_UNSIGNED_INT_IMAGE_CUBE_EXT 0x9066
04981 #define GL_UNSIGNED_INT_IMAGE_BUFFER_EXT 0x9067
04982 #define GL_UNSIGNED_INT_IMAGE_1D_ARRAY_EXT 0x9068
04983 #define GL_UNSIGNED_INT_IMAGE_2D_ARRAY_EXT 0x9069
04984 #define GL_UNSIGNED_INT_IMAGE_CUBE_MAP_ARRAY_EXT 0x906A
04985 #define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE_EXT 0x906B
04986 #define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE_ARRAY_EXT 0x906C
04987 #define GL_MAX_IMAGE_SAMPLES_EXT 0x906D
04988 #define GL_IMAGE_BINDING_FORMAT_EXT 0x906E
04989 #define GL_VERTEX_ATTRIB_ARRAY_BARRIER_BIT_EXT 0x00000001
04990 #define GL_ELEMENT_ARRAY_BARRIER_BIT_EXT 0x00000002
04991 #define GL_UNIFORM_BARRIER_BIT_EXT 0x00000004
04992 #define GL_TEXTURE_FETCH_BARRIER_BIT_EXT 0x00000008
04993 #define GL_SHADER_IMAGE_ACCESS_BARRIER_BIT_EXT 0x00000020
04994 #define GL_COMMAND_BARRIER_BIT_EXT 0x00000040
04995 #define GL_PIXEL_BUFFER_BARRIER_BIT_EXT 0x00000080
04996 #define GL_TEXTURE_UPDATE_BARRIER_BIT_EXT 0x00000100
04997 #define GL_BUFFER_UPDATE_BARRIER_BIT_EXT 0x00000200
04998 #define GL_FRAMEBUFFER_BARRIER_BIT_EXT 0x00000400
04999 #define GL_TRANSFORM_FEEDBACK_BARRIER_BIT_EXT 0x00000800
05000 #define GL_ATOMIC_COUNTER_BARRIER_BIT_EXT 0x00001000
05001 #define GL_ALL_BARRIER_BITS_EXT 0xFFFFFFFF
05002 #define GL_SHARED_TEXTURE_PALETTE_EXT 0x81FB
05003 #define GL_STENCIL_TAG_BITS_EXT 0x88F2
05004 #define GL_STENCIL_CLEAR_TAG_VALUE_EXT 0x88F3
05005 #define GL_STENCIL_TEST_TWO_SIDE_EXT 0x8910
05006 #define GL_ACTIVE_STENCIL_FACE_EXT 0x8911
05007 #define GL_INCR_WRAP_EXT 0x8507
05008 #define GL DECR_WRAP_EXT 0x8508
05009 #define GL_ALPHA4_EXT 0x803B
05010 #define GL_ALPHA8_EXT 0x803C
05011 #define GL_ALPHA12_EXT 0x803D
05012 #define GL_ALPHA16_EXT 0x803E
05013 #define GL_LUMINANCE4_EXT 0x803F
05014 #define GL_LUMINANCE8_EXT 0x8040
05015 #define GL_LUMINANCE12_EXT 0x8041
05016 #define GL_LUMINANCE16_EXT 0x8042
05017 #define GL_LUMINANCE4_ALPHA4_EXT 0x8043
05018 #define GL_LUMINANCE6_ALPHA2_EXT 0x8044
05019 #define GL_LUMINANCE8_ALPHA8_EXT 0x8045
05020 #define GL_LUMINANCE12_ALPHA4_EXT 0x8046
05021 #define GL_LUMINANCE12_ALPHA12_EXT 0x8047
05022 #define GL_LUMINANCE16_ALPHA16_EXT 0x8048
05023 #define GL_INTENSITY_EXT 0x8049
05024 #define GL_INTENSITY4_EXT 0x804A
05025 #define GL_INTENSITY8_EXT 0x804B
05026 #define GL_INTENSITY12_EXT 0x804C
05027 #define GL_INTENSITY16_EXT 0x804D
05028 #define GL_RGB2_EXT 0x804E
05029 #define GL_RGB4_EXT 0x804F
05030 #define GL_RGB5_EXT 0x8050
05031 #define GL_RGB8_EXT 0x8051
05032 #define GL_RGB10_EXT 0x8052
05033 #define GL_RGB12_EXT 0x8053
05034 #define GL_RGB16_EXT 0x8054
05035 #define GL_RGBA2_EXT 0x8055
05036 #define GL_RGBA4_EXT 0x8056
05037 #define GL_RGB5_A1_EXT 0x8057
05038 #define GL_RGBA8_EXT 0x8058
05039 #define GL_RGB10_A2_EXT 0x8059
05040 #define GL_RGBA12_EXT 0x805A
05041 #define GL_RGBA16_EXT 0x805B
05042 #define GL_TEXTURE_RED_SIZE_EXT 0x805C
05043 #define GL_TEXTURE_GREEN_SIZE_EXT 0x805D
05044 #define GL_TEXTURE_BLUE_SIZE_EXT 0x805E
05045 #define GL_TEXTURE_ALPHA_SIZE_EXT 0x805F
05046 #define GL_TEXTURE_LUMINANCE_SIZE_EXT 0x8060
```

```
05047 #define GL_TEXTURE_INTENSITY_SIZE_EXT 0x8061
05048 #define GL_REPLACE_EXT 0x8062
05049 #define GL_PROXY_TEXTURE_1D_EXT 0x8063
05050 #define GL_PROXY_TEXTURE_2D_EXT 0x8064
05051 #define GL_TEXTURE_TOO_LARGE_EXT 0x8065
05052 #define GL_PACK_SKIP_IMAGES_EXT 0x806B
05053 #define GL_PACK_IMAGE_HEIGHT_EXT 0x806C
05054 #define GL_UNPACK_SKIP_IMAGES_EXT 0x806D
05055 #define GL_UNPACK_IMAGE_HEIGHT_EXT 0x806E
05056 #define GL_TEXTURE_3D_EXT 0x806F
05057 #define GL_PROXY_TEXTURE_3D_EXT 0x8070
05058 #define GL_TEXTURE_DEPTH_EXT 0x8071
05059 #define GL_TEXTURE_WRAP_R_EXT 0x8072
05060 #define GL_MAX_3D_TEXTURE_SIZE_EXT 0x8073
05061 #define GL_TEXTURE_1D_ARRAY_EXT 0x8C18
05062 #define GL_PROXY_TEXTURE_1D_ARRAY_EXT 0x8C19
05063 #define GL_TEXTURE_2D_ARRAY_EXT 0x8C1A
05064 #define GL_PROXY_TEXTURE_2D_ARRAY_EXT 0x8C1B
05065 #define GL_TEXTURE_BINDING_1D_ARRAY_EXT 0x8C1C
05066 #define GL_TEXTURE_BINDING_2D_ARRAY_EXT 0x8C1D
05067 #define GL_MAX_ARRAY_TEXTURE_LAYERS_EXT 0x88FF
05068 #define GL_COMPARE_REF_DEPTH_TO_TEXTURE_EXT 0x884E
05069 #define GL_TEXTURE_BUFFER_EXT 0x8C2A
05070 #define GL_MAX_TEXTURE_BUFFER_SIZE_EXT 0x8C2B
05071 #define GL_TEXTURE_BINDING_BUFFER_EXT 0x8C2C
05072 #define GL_TEXTURE_BUFFER_DATA_STORE_BINDING_EXT 0x8C2D
05073 #define GL_TEXTURE_BUFFER_FORMAT_EXT 0x8C2E
05074 #define GL_COMPRESSED_LUMINANCE_LATC1_EXT 0x8C70
05075 #define GL_COMPRESSED_SIGNED_LUMINANCE_LATC1_EXT 0x8C71
05076 #define GL_COMPRESSED_LUMINANCE_ALPHA_LATC2_EXT 0x8C72
05077 #define GL_COMPRESSED_SIGNED_LUMINANCE_ALPHA_LATC2_EXT 0x8C73
05078 #define GL_COMPRESSED_RED_RGTC1_EXT 0x8DBB
05079 #define GL_COMPRESSED_SIGNED_RED_RGTC1_EXT 0x8DBC
05080 #define GL_COMPRESSED_RED_GREEN_RGTC2_EXT 0x8DBD
05081 #define GL_COMPRESSED_SIGNED_RED_GREEN_RGTC2_EXT 0x8DBE
05082 #define GL_COMPRESSED_RGB_S3TC_DXT1_EXT 0x83F0
05083 #define GL_COMPRESSED_RGBA_S3TC_DXT1_EXT 0x83F1
05084 #define GL_COMPRESSED_RGBA_S3TC_DXT3_EXT 0x83F2
05085 #define GL_COMPRESSED_RGBA_S3TC_DXT5_EXT 0x83F3
05086 #define GL_NORMAL_MAP_EXT 0x8511
05087 #define GL_REFLECTION_MAP_EXT 0x8512
05088 #define GL_TEXTURE_CUBE_MAP_EXT 0x8513
05089 #define GL_TEXTURE_BINDING_CUBE_MAP_EXT 0x8514
05090 #define GL_TEXTURE_CUBE_MAP_POSITIVE_X_EXT 0x8515
05091 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_X_EXT 0x8516
05092 #define GL_TEXTURE_CUBE_MAP_POSITIVE_Y_EXT 0x8517
05093 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y_EXT 0x8518
05094 #define GL_TEXTURE_CUBE_MAP_POSITIVE_Z_EXT 0x8519
05095 #define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z_EXT 0x851A
05096 #define GL_PROXY_TEXTURE_CUBE_MAP_EXT 0x851B
05097 #define GL_MAX_CUBE_MAP_TEXTURE_SIZE_EXT 0x851C
05098 #define GL_COMBINE_EXT 0x8570
05099 #define GL_COMBINE_RGB_EXT 0x8571
05100 #define GL_COMBINE_ALPHA_EXT 0x8572
05101 #define GL_RGB_SCALE_EXT 0x8573
05102 #define GL_ADD_SIGNED_EXT 0x8574
05103 #define GL_INTERPOLATE_EXT 0x8575
05104 #define GL_CONSTANT_EXT 0x8576
05105 #define GL_PRIMARY_COLOR_EXT 0x8577
05106 #define GL_PREVIOUS_EXT 0x8578
05107 #define GL_SOURCE0_RGB_EXT 0x8580
05108 #define GL_SOURCE1_RGB_EXT 0x8581
05109 #define GL_SOURCE2_RGB_EXT 0x8582
05110 #define GL_SOURCE0_ALPHA_EXT 0x8588
05111 #define GL_SOURCE1_ALPHA_EXT 0x8589
05112 #define GL_SOURCE2_ALPHA_EXT 0x858A
05113 #define GL_OPERAND0_RGB_EXT 0x8590
05114 #define GL_OPERAND1_RGB_EXT 0x8591
05115 #define GL_OPERAND2_RGB_EXT 0x8592
05116 #define GL_OPERAND0_ALPHA_EXT 0x8598
05117 #define GL_OPERAND1_ALPHA_EXT 0x8599
05118 #define GL_OPERAND2_ALPHA_EXT 0x859A
05119 #define GL_DOT3_RGB_EXT 0x8740
05120 #define GL_DOT3_RGBA_EXT 0x8741
05121 #define GL_TEXTURE_MAX_ANISOTROPY_EXT 0x84FE
05122 #define GL_MAX_TEXTURE_MAX_ANISOTROPY_EXT 0x84FF
05123 #define GL_TEXTURE_REDUCTION_MODE_EXT 0x9366
05124 #define GL_WEIGHTED_AVERAGE_EXT 0x9367
05125 #define GL_RGBA32UI_EXT 0x8D70
05126 #define GL_RGB32UI_EXT 0x8D71
05127 #define GL_ALPHA32UI_EXT 0x8D72
05128 #define GL_INTENSITY32UI_EXT 0x8D73
05129 #define GL_LUMINANCE32UI_EXT 0x8D74
05130 #define GL_LUMINANCE_ALPHA32UI_EXT 0x8D75
05131 #define GL_RGBA16UI_EXT 0x8D76
05132 #define GL_RGB16UI_EXT 0x8D77
05133 #define GL_ALPHA16UI_EXT 0x8D78
```

```
05134 #define GL_INTENSITY16UI_EXT 0x8D79
05135 #define GL_LUMINANCE16UI_EXT 0x8D7A
05136 #define GL_LUMINANCE_ALPHA16UI_EXT 0x8D7B
05137 #define GL_RGBA8UI_EXT 0x8D7C
05138 #define GL_RGB8UI_EXT 0x8D7D
05139 #define GL_ALPHA8UI_EXT 0x8D7E
05140 #define GL_INTENSITY8UI_EXT 0x8D7F
05141 #define GL_LUMINANCE8UI_EXT 0x8D80
05142 #define GL_LUMINANCE_ALPHA8UI_EXT 0x8D81
05143 #define GL_RGBA32I_EXT 0x8D82
05144 #define GL_RGB32I_EXT 0x8D83
05145 #define GL_ALPHA32I_EXT 0x8D84
05146 #define GL_INTENSITY32I_EXT 0x8D85
05147 #define GL_LUMINANCE32I_EXT 0x8D86
05148 #define GL_LUMINANCE_ALPHA32I_EXT 0x8D87
05149 #define GL_RGBA16I_EXT 0x8D88
05150 #define GL_RGB16I_EXT 0x8D89
05151 #define GL_ALPHA16I_EXT 0x8D8A
05152 #define GL_INTENSITY16I_EXT 0x8D8B
05153 #define GL_LUMINANCE16I_EXT 0x8D8C
05154 #define GL_LUMINANCE_ALPHA16I_EXT 0x8D8D
05155 #define GL_RGBA8I_EXT 0x8D8E
05156 #define GL_RGB8I_EXT 0x8D8F
05157 #define GL_ALPHA8I_EXT 0x8D90
05158 #define GL_INTENSITY8I_EXT 0x8D91
05159 #define GL_LUMINANCE8I_EXT 0x8D92
05160 #define GL_LUMINANCE_ALPHA8I_EXT 0x8D93
05161 #define GL_RED_INTEGER_EXT 0x8D94
05162 #define GL_GREEN_INTEGER_EXT 0x8D95
05163 #define GL_BLUE_INTEGER_EXT 0x8D96
05164 #define GL_ALPHA_INTEGER_EXT 0x8D97
05165 #define GL_RGB_INTEGER_EXT 0x8D98
05166 #define GL_RGBA_INTEGER_EXT 0x8D99
05167 #define GL_BGR_INTEGER_EXT 0x8D9A
05168 #define GL_BGRA_INTEGER_EXT 0x8D9B
05169 #define GL_LUMINANCE_INTEGER_EXT 0x8D9C
05170 #define GL_LUMINANCE_ALPHA_INTEGER_EXT 0x8D9D
05171 #define GL_RGBA_INTEGER_MODE_EXT 0x8D9E
05172 #define GL_MAX_TEXTURE_LOD_BIAS_EXT 0x84FD
05173 #define GL_TEXTURE_FILTER_CONTROL_EXT 0x8500
05174 #define GL_TEXTURE_LOD_BIAS_EXT 0x8501
05175 #define GL_MIRROR_CLAMP_EXT 0x8742
05176 #define GL_MIRROR_CLAMP_TO_EDGE_EXT 0x8743
05177 #define GL_MIRROR_CLAMP_TO_BORDER_EXT 0x8912
05178 #define GL_TEXTURE_PRIORITY_EXT 0x8066
05179 #define GL_TEXTURE_RESIDENT_EXT 0x8067
05180 #define GL_TEXTURE_1D_BINDING_EXT 0x8068
05181 #define GL_TEXTURE_2D_BINDING_EXT 0x8069
05182 #define GL_TEXTURE_3D_BINDING_EXT 0x806A
05183 #define GL_PERTURB_EXT 0x85AE
05184 #define GL_TEXTURE_NORMAL_EXT 0x85AF
05185 #define GL_SRGB_EXT 0x8C40
05186 #define GL_SRGB8_EXT 0x8C41
05187 #define GL_SRGB_ALPHA_EXT 0x8C42
05188 #define GL_SRGB8_ALPHA8_EXT 0x8C43
05189 #define GL_SLUMINANCE_ALPHA_EXT 0x8C44
05190 #define GL_SLUMINANCE8_ALPHA8_EXT 0x8C45
05191 #define GL_SLUMINANCE_EXT 0x8C46
05192 #define GL_SLUMINANCE8_EXT 0x8C47
05193 #define GL_COMPRESSED_SRGB_EXT 0x8C48
05194 #define GL_COMPRESSED_SRGB_ALPHA_EXT 0x8C49
05195 #define GL_COMPRESSED_SLUMINANCE_EXT 0x8C4A
05196 #define GL_COMPRESSED_SLUMINANCE_ALPHA_EXT 0x8C4B
05197 #define GL_COMPRESSED_SRGB_S3TC_DXT1_EXT 0x8C4C
05198 #define GL_COMPRESSED_SRGB_ALPHA_S3TC_DXT1_EXT 0x8C4D
05199 #define GL_COMPRESSED_SRGB_ALPHA_S3TC_DXT3_EXT 0x8C4E
05200 #define GL_COMPRESSED_SRGB_ALPHA_S3TC_DXT5_EXT 0x8C4F
05201 #define GL_SR8_EXT 0x8FBD
05202 #define GL_SRG8_EXT 0x8FBE
05203 #define GL_TEXTURE_SRGB_DECODE_EXT 0x8A48
05204 #define GL_DECODE_EXT 0x8A49
05205 #define GL_SKIP_DECODE_EXT 0x8A4A
05206 #define GL_RGB9_E5_EXT 0x8C3D
05207 #define GL_UNSIGNED_INT_5_9_9_REV_EXT 0x8C3E
05208 #define GL_TEXTURE_SHARED_SIZE_EXT 0x8C3F
05209 #define GL_ALPHA_SNORM 0x9010
05210 #define GL_LUMINANCE_SNORM 0x9011
05211 #define GL_LUMINANCE_ALPHA_SNORM 0x9012
05212 #define GL_INTENSITY_SNORM 0x9013
05213 #define GL_ALPHA8_SNORM 0x9014
05214 #define GL_LUMINANCE8_SNORM 0x9015
05215 #define GL_LUMINANCE8_ALPHA8_SNORM 0x9016
05216 #define GL_INTENSITY8_SNORM 0x9017
05217 #define GL_ALPHA16_SNORM 0x9018
05218 #define GL_LUMINANCE16_SNORM 0x9019
05219 #define GL_LUMINANCE16_ALPHA16_SNORM 0x901A
05220 #define GL_INTENSITY16_SNORM 0x901B
```

```
05221 #define GL_RED_SNORM 0x8F90
05222 #define GL_RG_SNORM 0x8F91
05223 #define GL_RGB_SNORM 0x8F92
05224 #define GL_RGBA_SNORM 0x8F93
05225 #define GL_TEXTURE_IMMUTABLE_FORMAT_EXT 0x912F
05226 #define GL_RGBA32F_EXT 0x8814
05227 #define GL_RGB32F_EXT 0x8815
05228 #define GL_ALPHA32F_EXT 0x8816
05229 #define GL_LUMINANCE32F_EXT 0x8818
05230 #define GL_LUMINANCE_ALPHA32F_EXT 0x8819
05231 #define GL_RGBA16F_EXT 0x881A
05232 #define GL_RGB16F_EXT 0x881B
05233 #define GL_ALPHA16F_EXT 0x881C
05234 #define GL_LUMINANCE16F_EXT 0x881E
05235 #define GL_LUMINANCE_ALPHA16F_EXT 0x881F
05236 #define GL_BGRA8_EXT 0x93A1
05237 #define GL_R8_EXT 0x8229
05238 #define GL_RG8_EXT 0x822B
05239 #define GL_R32F_EXT 0x822E
05240 #define GL_RG32F_EXT 0x8230
05241 #define GL_R16F_EXT 0x822D
05242 #define GL_RG16F_EXT 0x822F
05243 #define GL_TEXTURE_SWIZZLE_R_EXT 0x8E42
05244 #define GL_TEXTURE_SWIZZLE_G_EXT 0x8E43
05245 #define GL_TEXTURE_SWIZZLE_B_EXT 0x8E44
05246 #define GL_TEXTURE_SWIZZLE_A_EXT 0x8E45
05247 #define GL_TEXTURE_SWIZZLE_RGBA_EXT 0x8E46
05248 #define GL_TIME_ELAPSED_EXT 0x88BF
05249 #define GL_TRANSFORM_FEEDBACK_BUFFER_EXT 0x8C8E
05250 #define GL_TRANSFORM_FEEDBACK_BUFFER_START_EXT 0x8C84
05251 #define GL_TRANSFORM_FEEDBACK_BUFFER_SIZE_EXT 0x8C85
05252 #define GL_TRANSFORM_FEEDBACK_BUFFER_BINDING_EXT 0x8C8F
05253 #define GL_INTERLEAVED_ATTRIBS_EXT 0x8C8C
05254 #define GL_SEPARATE_ATTRIBS_EXT 0x8C8D
05255 #define GL_PRIMITIVES_GENERATED_EXT 0x8C87
05256 #define GL_TRANSFORM_FEEDBACK_PRIMITIVES_WRITTEN_EXT 0x8C88
05257 #define GL_RASTERIZER_DISCARD_EXT 0x8C89
05258 #define GL_MAX_TRANSFORM_FEEDBACK_INTERLEAVED_COMPONENTS_EXT 0x8C8A
05259 #define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_ATTRIBS_EXT 0x8C8B
05260 #define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_COMPONENTS_EXT 0x8C80
05261 #define GL_TRANSFORM_FEEDBACK_VARYINGS_EXT 0x8C83
05262 #define GL_TRANSFORM_FEEDBACK_BUFFER_MODE_EXT 0x8C7F
05263 #define GL_TRANSFORM_FEEDBACK_VARYING_MAX_LENGTH_EXT 0x8C76
05264 #define GL_VERTEX_ARRAY_EXT 0x8074
05265 #define GL_NORMAL_ARRAY_EXT 0x8075
05266 #define GL_COLOR_ARRAY_EXT 0x8076
05267 #define GL_INDEX_ARRAY_EXT 0x8077
05268 #define GL_TEXTURE_COORD_ARRAY_EXT 0x8078
05269 #define GL_EDGE_FLAG_ARRAY_EXT 0x8079
05270 #define GL_VERTEX_ARRAY_SIZE_EXT 0x807A
05271 #define GL_VERTEX_ARRAY_TYPE_EXT 0x807B
05272 #define GL_VERTEX_ARRAY_STRIDE_EXT 0x807C
05273 #define GL_VERTEX_ARRAY_COUNT_EXT 0x807D
05274 #define GL_NORMAL_ARRAY_TYPE_EXT 0x807E
05275 #define GL_NORMAL_ARRAY_STRIDE_EXT 0x807F
05276 #define GL_NORMAL_ARRAY_COUNT_EXT 0x8080
05277 #define GL_COLOR_ARRAY_SIZE_EXT 0x8081
05278 #define GL_COLOR_ARRAY_TYPE_EXT 0x8082
05279 #define GL_COLOR_ARRAY_STRIDE_EXT 0x8083
05280 #define GL_COLOR_ARRAY_COUNT_EXT 0x8084
05281 #define GL_INDEX_ARRAY_TYPE_EXT 0x8085
05282 #define GL_INDEX_ARRAY_STRIDE_EXT 0x8086
05283 #define GL_INDEX_ARRAY_COUNT_EXT 0x8087
05284 #define GL_TEXTURE_COORD_ARRAY_SIZE_EXT 0x8088
05285 #define GL_TEXTURE_COORD_ARRAY_TYPE_EXT 0x8089
05286 #define GL_TEXTURE_COORD_ARRAY_STRIDE_EXT 0x808A
05287 #define GL_TEXTURE_COORD_ARRAY_COUNT_EXT 0x808B
05288 #define GL_EDGE_FLAG_ARRAY_STRIDE_EXT 0x808C
05289 #define GL_EDGE_FLAG_ARRAY_COUNT_EXT 0x808D
05290 #define GL_VERTEX_ARRAY_POINTER_EXT 0x808E
05291 #define GL_NORMAL_ARRAY_POINTER_EXT 0x808F
05292 #define GL_COLOR_ARRAY_POINTER_EXT 0x8090
05293 #define GL_INDEX_ARRAY_POINTER_EXT 0x8091
05294 #define GL_TEXTURE_COORD_ARRAY_POINTER_EXT 0x8092
05295 #define GL_EDGE_FLAG_ARRAY_POINTER_EXT 0x8093
05296 #define GL_DOUBLE_VEC2_EXT 0x8FFC
05297 #define GL_DOUBLE_VEC3_EXT 0x8FFD
05298 #define GL_DOUBLE_VEC4_EXT 0x8FFE
05299 #define GL_DOUBLE_MAT2_EXT 0x8F46
05300 #define GL_DOUBLE_MAT3_EXT 0x8F47
05301 #define GL_DOUBLE_MAT4_EXT 0x8F48
05302 #define GL_DOUBLE_MAT2x3_EXT 0x8F49
05303 #define GL_DOUBLE_MAT2x4_EXT 0x8F4A
05304 #define GL_DOUBLE_MAT3x2_EXT 0x8F4B
05305 #define GL_DOUBLE_MAT3x4_EXT 0x8F4C
05306 #define GL_DOUBLE_MAT4x2_EXT 0x8F4D
05307 #define GL_DOUBLE_MAT4x3_EXT 0x8F4E
```



```
05308 #define GL_VERTEX_SHADER_EXT 0x8780
05309 #define GL_VERTEX_SHADER_BINDING_EXT 0x8781
05310 #define GL_OP_INDEX_EXT 0x8782
05311 #define GL_OP_NEGATE_EXT 0x8783
05312 #define GL_OP_DOT3_EXT 0x8784
05313 #define GL_OP_DOT4_EXT 0x8785
05314 #define GL_OP_MUL_EXT 0x8786
05315 #define GL_OP_ADD_EXT 0x8787
05316 #define GL_OP_MADD_EXT 0x8788
05317 #define GL_OP_FRAC_EXT 0x8789
05318 #define GL_OP_MAX_EXT 0x878A
05319 #define GL_OP_MIN_EXT 0x878B
05320 #define GL_OP_SET_GE_EXT 0x878C
05321 #define GL_OP_SET_LT_EXT 0x878D
05322 #define GL_OP_CLAMP_EXT 0x878E
05323 #define GL_OP_FLOOR_EXT 0x878F
05324 #define GL_OP_ROUND_EXT 0x8790
05325 #define GL_OP_EXP_BASE_2_EXT 0x8791
05326 #define GL_OP_LOG_BASE_2_EXT 0x8792
05327 #define GL_OP_POWER_EXT 0x8793
05328 #define GL_OP_RECIP_EXT 0x8794
05329 #define GL_OP_RECIP_SQRT_EXT 0x8795
05330 #define GL_OP_SUB_EXT 0x8796
05331 #define GL_OP_CROSS_PRODUCT_EXT 0x8797
05332 #define GL_OP_MULTIPLY_MATRIX_EXT 0x8798
05333 #define GL_OP_MOV_EXT 0x8799
05334 #define GL_OUTPUT_VERTEX_EXT 0x879A
05335 #define GL_OUTPUT_COLOR0_EXT 0x879B
05336 #define GL_OUTPUT_COLOR1_EXT 0x879C
05337 #define GL_OUTPUT_TEXTURE_COORD0_EXT 0x879D
05338 #define GL_OUTPUT_TEXTURE_COORD1_EXT 0x879E
05339 #define GL_OUTPUT_TEXTURE_COORD2_EXT 0x879F
05340 #define GL_OUTPUT_TEXTURE_COORD3_EXT 0x87A0
05341 #define GL_OUTPUT_TEXTURE_COORD4_EXT 0x87A1
05342 #define GL_OUTPUT_TEXTURE_COORD5_EXT 0x87A2
05343 #define GL_OUTPUT_TEXTURE_COORD6_EXT 0x87A3
05344 #define GL_OUTPUT_TEXTURE_COORD7_EXT 0x87A4
05345 #define GL_OUTPUT_TEXTURE_COORD8_EXT 0x87A5
05346 #define GL_OUTPUT_TEXTURE_COORD9_EXT 0x87A6
05347 #define GL_OUTPUT_TEXTURE_COORD10_EXT 0x87A7
05348 #define GL_OUTPUT_TEXTURE_COORD11_EXT 0x87A8
05349 #define GL_OUTPUT_TEXTURE_COORD12_EXT 0x87A9
05350 #define GL_OUTPUT_TEXTURE_COORD13_EXT 0x87AA
05351 #define GL_OUTPUT_TEXTURE_COORD14_EXT 0x87AB
05352 #define GL_OUTPUT_TEXTURE_COORD15_EXT 0x87AC
05353 #define GL_OUTPUT_TEXTURE_COORD16_EXT 0x87AD
05354 #define GL_OUTPUT_TEXTURE_COORD17_EXT 0x87AE
05355 #define GL_OUTPUT_TEXTURE_COORD18_EXT 0x87AF
05356 #define GL_OUTPUT_TEXTURE_COORD19_EXT 0x87B0
05357 #define GL_OUTPUT_TEXTURE_COORD20_EXT 0x87B1
05358 #define GL_OUTPUT_TEXTURE_COORD21_EXT 0x87B2
05359 #define GL_OUTPUT_TEXTURE_COORD22_EXT 0x87B3
05360 #define GL_OUTPUT_TEXTURE_COORD23_EXT 0x87B4
05361 #define GL_OUTPUT_TEXTURE_COORD24_EXT 0x87B5
05362 #define GL_OUTPUT_TEXTURE_COORD25_EXT 0x87B6
05363 #define GL_OUTPUT_TEXTURE_COORD26_EXT 0x87B7
05364 #define GL_OUTPUT_TEXTURE_COORD27_EXT 0x87B8
05365 #define GL_OUTPUT_TEXTURE_COORD28_EXT 0x87B9
05366 #define GL_OUTPUT_TEXTURE_COORD29_EXT 0x87BA
05367 #define GL_OUTPUT_TEXTURE_COORD30_EXT 0x87BB
05368 #define GL_OUTPUT_TEXTURE_COORD31_EXT 0x87BC
05369 #define GL_OUTPUT_FOG_EXT 0x87BD
05370 #define GL_SCALAR_EXT 0x87BE
05371 #define GL_VECTOR_EXT 0x87BF
05372 #define GL_MATRIX_EXT 0x87C0
05373 #define GL_VARIANT_EXT 0x87C1
05374 #define GL_INVARIANT_EXT 0x87C2
05375 #define GL_LOCAL_CONSTANT_EXT 0x87C3
05376 #define GL_LOCAL_EXT 0x87C4
05377 #define GL_MAX_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87C5
05378 #define GL_MAX_VERTEX_SHADER_VARIANTS_EXT 0x87C6
05379 #define GL_MAX_VERTEX_SHADER_INVARIANTS_EXT 0x87C7
05380 #define GL_MAX_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87C8
05381 #define GL_MAX_VERTEX_SHADER_LOCALS_EXT 0x87C9
05382 #define GL_MAX_OPTIMIZED_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87CA
05383 #define GL_MAX_OPTIMIZED_VERTEX_SHADER_VARIANTS_EXT 0x87CB
05384 #define GL_MAX_OPTIMIZED_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87CC
05385 #define GL_MAX_OPTIMIZED_VERTEX_SHADER_INVARIANTS_EXT 0x87CD
05386 #define GL_MAX_OPTIMIZED_VERTEX_SHADER_LOCALS_EXT 0x87CE
05387 #define GL_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87CF
05388 #define GL_VERTEX_SHADER_VARIANTS_EXT 0x87D0
05389 #define GL_VERTEX_SHADER_INVARIANTS_EXT 0x87D1
05390 #define GL_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87D2
05391 #define GL_VERTEX_SHADER_LOCALS_EXT 0x87D3
05392 #define GL_VERTEX_SHADER_OPTIMIZED_EXT 0x87D4
05393 #define GL_X_EXT 0x87D5
05394 #define GL_Y_EXT 0x87D6
```

```
05395 #define GL_Z_EXT 0x87D7
05396 #define GL_W_EXT 0x87D8
05397 #define GL_NEGATIVE_X_EXT 0x87D9
05398 #define GL_NEGATIVE_Y_EXT 0x87DA
05399 #define GL_NEGATIVE_Z_EXT 0x87DB
05400 #define GL_NEGATIVE_W_EXT 0x87DC
05401 #define GL_ZERO_EXT 0x87DD
05402 #define GL_ONE_EXT 0x87DE
05403 #define GL_NEGATIVE_ONE_EXT 0x87DF
05404 #define GL_NORMALIZED_RANGE_EXT 0x87E0
05405 #define GL_FULL_RANGE_EXT 0x87E1
05406 #define GL_CURRENT_VERTEX_EXT 0x87E2
05407 #define GL_MVP_MATRIX_EXT 0x87E3
05408 #define GL_VARIANT_VALUE_EXT 0x87E4
05409 #define GL_VARIANT_DATATYPE_EXT 0x87E5
05410 #define GL_VARIANT_ARRAY_STRIDE_EXT 0x87E6
05411 #define GL_VARIANT_ARRAY_TYPE_EXT 0x87E7
05412 #define GL_VARIANT_ARRAY_EXT 0x87E8
05413 #define GL_VARIANT_ARRAY_POINTER_EXT 0x87E9
05414 #define GL_INVARIANT_VALUE_EXT 0x87EA
05415 #define GL_INVARIANT_DATATYPE_EXT 0x87EB
05416 #define GL_LOCAL_CONSTANT_VALUE_EXT 0x87EC
05417 #define GL_LOCAL_CONSTANT_DATATYPE_EXT 0x87ED
05418 #define GL_MODELVIEW0_STACK_DEPTH_EXT 0x0BA3
05419 #define GL_MODELVIEW1_STACK_DEPTH_EXT 0x8502
05420 #define GL_MODELVIEW0_MATRIX_EXT 0x0BA6
05421 #define GL_MODELVIEW1_MATRIX_EXT 0x8506
05422 #define GL_VERTEX_WEIGHTING_EXT 0x8509
05423 #define GL_MODELVIEW0_EXT 0x1700
05424 #define GL_MODELVIEW1_EXT 0x850A
05425 #define GL_CURRENT_VERTEX_WEIGHT_EXT 0x850B
05426 #define GL_VERTEX_WEIGHT_ARRAY_EXT 0x850C
05427 #define GL_VERTEX_WEIGHT_ARRAY_SIZE_EXT 0x850D
05428 #define GL_VERTEX_WEIGHT_ARRAY_TYPE_EXT 0x850E
05429 #define GL_VERTEX_WEIGHT_ARRAY_STRIDE_EXT 0x850F
05430 #define GL_VERTEX_WEIGHT_ARRAY_POINTER_EXT 0x8510
05431 #define GL_INCLUSIVE_EXT 0x8F10
05432 #define GL_EXCLUSIVE_EXT 0x8F11
05433 #define GL_WINDOW_RECTANGLE_EXT 0x8F12
05434 #define GL_WINDOW_RECTANGLE_MODE_EXT 0x8F13
05435 #define GL_MAX_WINDOW_RECTANGLES_EXT 0x8F14
05436 #define GL_NUM_WINDOW_RECTANGLES_EXT 0x8F15
05437 #define GL_SYNC_X11_FENCE_EXT 0x90E1
05438 #define GL_IGNORE_BORDER_HP 0x8150
05439 #define GL_CONSTANT_BORDER_HP 0x8151
05440 #define GL_REPLICATE_BORDER_HP 0x8153
05441 #define GL_CONVOLUTION_BORDER_COLOR_HP 0x8154
05442 #define GL_IMAGE_SCALE_X_HP 0x8155
05443 #define GL_IMAGE_SCALE_Y_HP 0x8156
05444 #define GL_IMAGE_TRANSLATE_X_HP 0x8157
05445 #define GL_IMAGE_TRANSLATE_Y_HP 0x8158
05446 #define GL_IMAGE_ROTATE_ANGLE_HP 0x8159
05447 #define GL_IMAGE_ROTATE_ORIGIN_X_HP 0x815A
05448 #define GL_IMAGE_ROTATE_ORIGIN_Y_HP 0x815B
05449 #define GL_IMAGE_MAG_FILTER_HP 0x815C
05450 #define GL_IMAGE_MIN_FILTER_HP 0x815D
05451 #define GL_IMAGE_CUBIC_WEIGHT_HP 0x815E
05452 #define GL_CUBIC_HP 0x815F
05453 #define GL_AVERAGE_HP 0x8160
05454 #define GL_IMAGE_TRANSFORM_2D_HP 0x8161
05455 #define GL_POST_IMAGE_TRANSFORM_COLOR_TABLE_HP 0x8162
05456 #define GL_PROXY_POST_IMAGE_TRANSFORM_COLOR_TABLE_HP 0x8163
05457 #define GL_OCCLUSION_TEST_HP 0x8165
05458 #define GL_OCCLUSION_TEST_RESULT_HP 0x8166
05459 #define GL_TEXTURE_LIGHTING_MODE_HP 0x8167
05460 #define GL_TEXTURE_POST_SPECULAR_HP 0x8168
05461 #define GL_TEXTURE_PRE_SPECULAR_HP 0x8169
05462 #define GL_CULL_VERTEX_IBM 103050
05463 #define GL_RASTER_POSITION_UNCLIPPED_IBM 0x19262
05464 #define GL_ALL_STATIC_DATA_IBM 103060
05465 #define GL_STATIC_VERTEX_ARRAY_IBM 103061
05466 #define GL_MIRRORED_REPEAT_IBM 0x8370
05467 #define GL_VERTEX_ARRAY_LIST_IBM 103070
05468 #define GL_NORMAL_ARRAY_LIST_IBM 103071
05469 #define GL_COLOR_ARRAY_LIST_IBM 103072
05470 #define GL_INDEX_ARRAY_LIST_IBM 103073
05471 #define GL_TEXTURE_COORD_ARRAY_LIST_IBM 103074
05472 #define GL_EDGE_FLAG_ARRAY_LIST_IBM 103075
05473 #define GL_FOG_COORDINATE_ARRAY_LIST_IBM 103076
05474 #define GL_SECONDARY_COLOR_ARRAY_LIST_IBM 103077
05475 #define GL_VERTEX_ARRAY_LIST_STRIDE_IBM 103080
05476 #define GL_NORMAL_ARRAY_LIST_STRIDE_IBM 103081
05477 #define GL_COLOR_ARRAY_LIST_STRIDE_IBM 103082
05478 #define GL_INDEX_ARRAY_LIST_STRIDE_IBM 103083
05479 #define GL_TEXTURE_COORD_ARRAY_LIST_STRIDE_IBM 103084
05480 #define GL_EDGE_FLAG_ARRAY_LIST_STRIDE_IBM 103085
05481 #define GL_FOG_COORDINATE_ARRAY_LIST_STRIDE_IBM 103086
```

```
05482 #define GL_SECONDARY_COLOR_ARRAY_LIST_STRIDE_IBM 103087
05483 #define GL_RED_MIN_CLAMP_INGR 0x8560
05484 #define GL_GREEN_MIN_CLAMP_INGR 0x8561
05485 #define GL_BLUE_MIN_CLAMP_INGR 0x8562
05486 #define GL_ALPHA_MIN_CLAMP_INGR 0x8563
05487 #define GL_RED_MAX_CLAMP_INGR 0x8564
05488 #define GL_GREEN_MAX_CLAMP_INGR 0x8565
05489 #define GL_BLUE_MAX_CLAMP_INGR 0x8566
05490 #define GL_ALPHA_MAX_CLAMP_INGR 0x8567
05491 #define GL_INTERLACE_READ_INGR 0x8568
05492 #define GL_BLACKHOLE_RENDER_INTEL 0x83FC
05493 #define GL_CONSERVATIVE_RASTERIZATION_INTEL 0x83FE
05494 #define GL_TEXTURE_MEMORY_LAYOUT_INTEL 0x83FF
05495 #define GL_LAYOUT_DEFAULT_INTEL 0
05496 #define GL_LAYOUT_LINEAR_INTEL 1
05497 #define GL_LAYOUT_LINEAR_CPU_CACHED_INTEL 2
05498 #define GL_PARALLEL_ARRAYS_INTEL 0x83F4
05499 #define GL_VERTEX_ARRAY_PARALLEL_POINTERS_INTEL 0x83F5
05500 #define GL_NORMAL_ARRAY_PARALLEL_POINTERS_INTEL 0x83F6
05501 #define GL_COLOR_ARRAY_PARALLEL_POINTERS_INTEL 0x83F7
05502 #define GL_TEXTURE_COORD_ARRAY_PARALLEL_POINTERS_INTEL 0x83F8
05503 #define GL_PERFQUERY_SINGLE_CONTEXT_INTEL 0x00000000
05504 #define GL_PERFQUERY_GLOBAL_CONTEXT_INTEL 0x00000001
05505 #define GL_PERFQUERY_WAIT_INTEL 0x83FB
05506 #define GL_PERFQUERY_FLUSH_INTEL 0x83FA
05507 #define GL_PERFQUERY_DONOT_FLUSH_INTEL 0x83F9
05508 #define GL_PERFQUERY_COUNTER_EVENT_INTEL 0x94F0
05509 #define GL_PERFQUERY_COUNTER_DURATION_NORM_INTEL 0x94F1
05510 #define GL_PERFQUERY_COUNTER_DURATION_RAW_INTEL 0x94F2
05511 #define GL_PERFQUERY_COUNTER_THROUGHPUT_INTEL 0x94F3
05512 #define GL_PERFQUERY_COUNTER_RAW_INTEL 0x94F4
05513 #define GL_PERFQUERY_COUNTER_TIMESTAMP_INTEL 0x94F5
05514 #define GL_PERFQUERY_COUNTER_DATA_UINT32_INTEL 0x94F8
05515 #define GL_PERFQUERY_COUNTER_DATA_UINT64_INTEL 0x94F9
05516 #define GL_PERFQUERY_COUNTER_DATA_FLOAT_INTEL 0x94FA
05517 #define GL_PERFQUERY_COUNTER_DATA_DOUBLE_INTEL 0x94FB
05518 #define GL_PERFQUERY_COUNTER_DATA_BOOL32_INTEL 0x94FC
05519 #define GL_PERFQUERY_QUERY_NAME_LENGTH_MAX_INTEL 0x94FD
05520 #define GL_PERFQUERY_COUNTER_NAME_LENGTH_MAX_INTEL 0x94FE
05521 #define GL_PERFQUERY_COUNTER_DESC_LENGTH_MAX_INTEL 0x94FF
05522 #define GL_PERFQUERY_GPA_EXTENDED_COUNTERS_INTEL 0x9500
05523 #define GL_MULTIPLY_KHR 0x9294
05524 #define GL_SCREEN_KHR 0x9295
05525 #define GL_OVERLAY_KHR 0x9296
05526 #define GL_DARKEN_KHR 0x9297
05527 #define GL_LIGHTEN_KHR 0x9298
05528 #define GL_COLOR Dodge_KHR 0x9299
05529 #define GL_COLORBURN_KHR 0x929A
05530 #define GL_HARDLIGHT_KHR 0x929B
05531 #define GL_SOFTLIGHT_KHR 0x929C
05532 #define GL_DIFFERENCE_KHR 0x929E
05533 #define GL_EXCLUSION_KHR 0x92A0
05534 #define GL_HSL_HUE_KHR 0x92AD
05535 #define GL_HSL_SATURATION_KHR 0x92AE
05536 #define GL_HSL_COLOR_KHR 0x92AF
05537 #define GL_HSL_LUMINOSITY_KHR 0x92B0
05538 #define GL_BLEND_ADVANCED_COHERENT_KHR 0x9285
05539 #define GL_CONTEXT_RELEASE_BEHAVIOR 0x82FB
05540 #define GL_CONTEXT_RELEASE_BEHAVIOR_FLUSH 0x82FC
05541 #define GL_CONTEXT_RELEASE_BEHAVIOR_KHR 0x82FB
05542 #define GL_CONTEXT_RELEASE_BEHAVIOR_FLUSH_KHR 0x82FC
05543 #define GL_DEBUG_OUTPUT_SYNCHRONOUS 0x8242
05544 #define GL_DEBUG_NEXT_LOGGED_MESSAGE_LENGTH 0x8243
05545 #define GL_DEBUG_CALLBACK_FUNCTION 0x8244
05546 #define GL_DEBUG_CALLBACK_USER_PARAM 0x8245
05547 #define GL_DEBUG_SOURCE_API 0x8246
05548 #define GL_DEBUG_SOURCE_WINDOW_SYSTEM 0x8247
05549 #define GL_DEBUG_SOURCE_SHADER_COMPILER 0x8248
05550 #define GL_DEBUG_SOURCE_THIRD_PARTY 0x8249
05551 #define GL_DEBUG_SOURCE_APPLICATION 0x824A
05552 #define GL_DEBUG_SOURCE_OTHER 0x824B
05553 #define GL_DEBUG_TYPE_ERROR 0x824C
05554 #define GL_DEBUG_TYPE_DEPRECATED_BEHAVIOR 0x824D
05555 #define GL_DEBUG_TYPE_UNDEFINED_BEHAVIOR 0x824E
05556 #define GL_DEBUG_TYPE_PORTABILITY 0x824F
05557 #define GL_DEBUG_TYPE_PERFORMANCE 0x8250
05558 #define GL_DEBUG_TYPE_OTHER 0x8251
05559 #define GL_DEBUG_TYPE_MARKER 0x8268
05560 #define GL_DEBUG_TYPE_PUSH_GROUP 0x8269
05561 #define GL_DEBUG_TYPE_POP_GROUP 0x826A
05562 #define GL_DEBUG_SEVERITY_NOTIFICATION 0x826B
05563 #define GL_MAX_DEBUG_GROUP_STACK_DEPTH 0x826C
05564 #define GL_DEBUG_GROUP_STACK_DEPTH 0x826D
05565 #define GL_BUFFER 0x82E0
05566 #define GL_SHADER 0x82E1
05567 #define GL_PROGRAM 0x82E2
05568 #define GL_VERTEX_ARRAY 0x8074
```



```
05569 #define GL_QUERY 0x82E3
05570 #define GL_PROGRAM_PIPELINE 0x82E4
05571 #define GL_MAX_LABEL_LENGTH 0x82E8
05572 #define GL_MAX_DEBUG_MESSAGE_LENGTH 0x9143
05573 #define GL_MAX_DEBUG_LOGGED_MESSAGES 0x9144
05574 #define GL_DEBUG_LOGGED_MESSAGES 0x9145
05575 #define GL_DEBUG_SEVERITY_HIGH 0x9146
05576 #define GL_DEBUG_SEVERITY_MEDIUM 0x9147
05577 #define GL_DEBUG_SEVERITY_LOW 0x9148
05578 #define GL_DEBUG_OUTPUT 0x92E0
05579 #define GL_CONTEXT_FLAG_DEBUG_BIT 0x00000002
05580 #define GL_STACK_OVERFLOW 0x0503
05581 #define GL_STACK_UNDERFLOW 0x0504
05582 #define GL_DEBUG_OUTPUT_SYNCHRONOUS_KHR 0x8242
05583 #define GL_DEBUG_NEXT_LOGGED_MESSAGE_LENGTH_KHR 0x8243
05584 #define GL_DEBUG_CALLBACK_FUNCTION_KHR 0x8244
05585 #define GL_DEBUG_CALLBACK_USER_PARAM_KHR 0x8245
05586 #define GL_DEBUG_SOURCE_API_KHR 0x8246
05587 #define GL_DEBUG_SOURCE_WINDOW_SYSTEM_KHR 0x8247
05588 #define GL_DEBUG_SOURCE_SHADER_COMPILER_KHR 0x8248
05589 #define GL_DEBUG_SOURCE_THIRD_PARTY_KHR 0x8249
05590 #define GL_DEBUG_SOURCE_APPLICATION_KHR 0x824A
05591 #define GL_DEBUG_SOURCE_OTHER_KHR 0x824B
05592 #define GL_DEBUG_TYPE_ERROR_KHR 0x824C
05593 #define GL_DEBUG_TYPE_DEPRECATED_BEHAVIOR_KHR 0x824D
05594 #define GL_DEBUG_TYPE_UNDEFINED_BEHAVIOR_KHR 0x824E
05595 #define GL_DEBUG_TYPE_PORTABILITY_KHR 0x824F
05596 #define GL_DEBUG_TYPE_PERFORMANCE_KHR 0x8250
05597 #define GL_DEBUG_TYPE_OTHER_KHR 0x8251
05598 #define GL_DEBUG_TYPE_MARKER_KHR 0x8268
05599 #define GL_DEBUG_TYPE_PUSH_GROUP_KHR 0x8269
05600 #define GL_DEBUG_TYPE_POP_GROUP_KHR 0x826A
05601 #define GL_DEBUG_SEVERITY_NOTIFICATION_KHR 0x826B
05602 #define GL_MAX_DEBUG_GROUP_STACK_DEPTH_KHR 0x826C
05603 #define GL_DEBUG_GROUP_STACK_DEPTH_KHR 0x826D
05604 #define GL_BUFFER_KHR 0x82E0
05605 #define GL_SHADER_KHR 0x82E1
05606 #define GL_PROGRAM_KHR 0x82E2
05607 #define GL_VERTEX_ARRAY_KHR 0x8074
05608 #define GL_QUERY_KHR 0x82E3
05609 #define GL_PROGRAM_PIPELINE_KHR 0x82E4
05610 #define GL_SAMPLER_KHR 0x82E6
05611 #define GL_MAX_LABEL_LENGTH_KHR 0x82E8
05612 #define GL_MAX_DEBUG_MESSAGE_LENGTH_KHR 0x9143
05613 #define GL_MAX_DEBUG_LOGGED_MESSAGES_KHR 0x9144
05614 #define GL_DEBUG_LOGGED_MESSAGES_KHR 0x9145
05615 #define GL_DEBUG_SEVERITY_HIGH_KHR 0x9146
05616 #define GL_DEBUG_SEVERITY_MEDIUM_KHR 0x9147
05617 #define GL_DEBUG_SEVERITY_LOW_KHR 0x9148
05618 #define GL_DEBUG_OUTPUT_KHR 0x92E0
05619 #define GL_CONTEXT_FLAG_DEBUG_BIT_KHR 0x00000002
05620 #define GL_STACK_OVERFLOW_KHR 0x0503
05621 #define GL_STACK_UNDERFLOW_KHR 0x0504
05622 #define GL_DISPLAY_LIST 0x82E7
05623 #define GL_CONTEXT_FLAG_NO_ERROR_BIT_KHR 0x00000008
05624 #define GL_MAX_SHADER_COMPILER_THREADS_KHR 0x91B0
05625 #define GL_COMPLETION_STATUS_KHR 0x91B1
05626 #define GL_CONTEXT_ROBUST_ACCESS 0x90F3
05627 #define GL_LOSE_CONTEXT_ON_RESET 0x8252
05628 #define GL_GUILTY_CONTEXT_RESET 0x8253
05629 #define GL_INNOCENT_CONTEXT_RESET 0x8254
05630 #define GL_UNKNOWN_CONTEXT_RESET 0x8255
05631 #define GL_RESET_NOTIFICATION_STRATEGY 0x8256
05632 #define GL_NO_RESET_NOTIFICATION 0x8261
05633 #define GL_CONTEXT_LOST 0x0507
05634 #define GL_CONTEXT_ROBUST_ACCESS_KHR 0x90F3
05635 #define GL_LOSE_CONTEXT_ON_RESET_KHR 0x8252
05636 #define GL_GUILTY_CONTEXT_RESET_KHR 0x8253
05637 #define GL_INNOCENT_CONTEXT_RESET_KHR 0x8254
05638 #define GL_UNKNOWN_CONTEXT_RESET_KHR 0x8255
05639 #define GL_RESET_NOTIFICATION_STRATEGY_KHR 0x8256
05640 #define GL_NO_RESET_NOTIFICATION_KHR 0x8261
05641 #define GL_CONTEXT_LOST_KHR 0x0507
05642 #define GL_SUBGROUP_SIZE_KHR 0x9532
05643 #define GL_SUBGROUP_SUPPORTED_STAGES_KHR 0x9533
05644 #define GL_SUBGROUP_SUPPORTED_FEATURES_KHR 0x9534
05645 #define GL_SUBGROUP_QUAD_ALL_STAGES_KHR 0x9535
05646 #define GL_SUBGROUP_FEATURE_BASIC_BIT_KHR 0x00000001
05647 #define GL_SUBGROUP_FEATURE_VOTE_BIT_KHR 0x00000002
05648 #define GL_SUBGROUP_FEATURE_ARITHMETIC_BIT_KHR 0x00000004
05649 #define GL_SUBGROUP_FEATURE_BALLOT_BIT_KHR 0x00000008
05650 #define GL_SUBGROUP_FEATURE_SHUFFLE_BIT_KHR 0x00000010
05651 #define GL_SUBGROUP_FEATURE_SHUFFLE_RELATIVE_BIT_KHR 0x00000020
05652 #define GL_SUBGROUP_FEATURE_CLUSTERED_BIT_KHR 0x00000040
05653 #define GL_SUBGROUP_FEATURE_QUAD_BIT_KHR 0x00000080
05654 #define GL_COMPRESSED_RGBA_ASTC_4x4_KHR 0x93B0
05655 #define GL_COMPRESSED_RGBA_ASTC_5x4_KHR 0x93B1
```

```
05656 #define GL_COMPRESSED_RGBA_ASTC_5x5_KHR 0x93B2
05657 #define GL_COMPRESSED_RGBA_ASTC_6x5_KHR 0x93B3
05658 #define GL_COMPRESSED_RGBA_ASTC_6x6_KHR 0x93B4
05659 #define GL_COMPRESSED_RGBA_ASTC_8x5_KHR 0x93B5
05660 #define GL_COMPRESSED_RGBA_ASTC_8x6_KHR 0x93B6
05661 #define GL_COMPRESSED_RGBA_ASTC_8x8_KHR 0x93B7
05662 #define GL_COMPRESSED_RGBA_ASTC_10x5_KHR 0x93B8
05663 #define GL_COMPRESSED_RGBA_ASTC_10x6_KHR 0x93B9
05664 #define GL_COMPRESSED_RGBA_ASTC_10x8_KHR 0x93BA
05665 #define GL_COMPRESSED_RGBA_ASTC_10x10_KHR 0x93BB
05666 #define GL_COMPRESSED_RGBA_ASTC_12x10_KHR 0x93BC
05667 #define GL_COMPRESSED_RGBA_ASTC_12x12_KHR 0x93BD
05668 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_4x4_KHR 0x93D0
05669 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_5x4_KHR 0x93D1
05670 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_5x5_KHR 0x93D2
05671 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_6x5_KHR 0x93D3
05672 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_6x6_KHR 0x93D4
05673 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_8x5_KHR 0x93D5
05674 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_8x6_KHR 0x93D6
05675 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_8x8_KHR 0x93D7
05676 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x5_KHR 0x93D8
05677 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x6_KHR 0x93D9
05678 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x8_KHR 0x93DA
05679 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x10_KHR 0x93DB
05680 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_12x10_KHR 0x93DC
05681 #define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_12x12_KHR 0x93DD
05682 #define GL_TEXTURE_1D_STACK_MESAX 0x8759
05683 #define GL_TEXTURE_2D_STACK_MESAX 0x875A
05684 #define GL_PROXY_TEXTURE_1D_STACK_MESAX 0x875B
05685 #define GL_PROXY_TEXTURE_2D_STACK_MESAX 0x875C
05686 #define GL_TEXTURE_1D_STACK_BINDING_MESAX 0x875D
05687 #define GL_TEXTURE_2D_STACK_BINDING_MESAX 0x875E
05688 #define GL_FRAMEBUFFER_FLIP_X_MESA 0x8BBB
05689 #define GL_FRAMEBUFFER_FLIP_Y_MESA 0x8BBB
05690 #define GL_FRAMEBUFFER_SWAP_XY_MESA 0x8BBB
05691 #define GL_PACK_INVERT_MESA 0x8758
05692 #define GL_PROGRAM_BINARY_FORMAT_MESA 0x875F
05693 #define GL_TILE_RASTER_ORDER_FIXED_MESA 0x8BB8
05694 #define GL_TILE_RASTER_ORDER_INCREASING_X_MESA 0x8BB9
05695 #define GL_TILE_RASTER_ORDER_INCREASING_Y_MESA 0x8BBA
05696 #define GL_UNSIGNED_SHORT_8_8_MESA 0x85BA
05697 #define GL_UNSIGNED_SHORT_8_8_REV_MESA 0x85BB
05698 #define GL_YCBCR_MESA 0x8757
05699 #define GL_GPU_MEMORY_INFO_DEDICATED_VIDMEM_NVX 0x9047
05700 #define GL_GPU_MEMORY_INFO_TOTAL_AVAILABLE_MEMORY_NVX 0x9048
05701 #define GL_GPU_MEMORY_INFO_CURRENT_AVAILABLE_VIDMEM_NVX 0x9049
05702 #define GL_GPU_MEMORY_INFO_EVICTION_COUNT_NVX 0x904A
05703 #define GL_GPU_MEMORY_INFO_EVICTED_MEMORY_NVX 0x904B
05704 #define GL_UPLOAD_GPU_MASK_NVX 0x954A
05705 #define GL_LGPU_SEPARATE_STORAGE_BIT_NVX 0x0800
05706 #define GL_MAX_LGPU_GPUS_NVX 0x92BA
05707 #define GL_ALPHA_TO_COVERAGE_DITHER_DEFAULT_NV 0x934D
05708 #define GL_ALPHA_TO_COVERAGE_DITHER_ENABLE_NV 0x934E
05709 #define GL_ALPHA_TO_COVERAGE_DITHER_DISABLE_NV 0x934F
05710 #define GL_ALPHA_TO_COVERAGE_DITHER_MODE_NV 0x92BF
05711 #define GL_BLEND_OVERLAP_NV 0x9281
05712 #define GL_BLEND_PREMULTIPLIED_SRC_NV 0x9280
05713 #define GL_BLUE_NV 0x1905
05714 #define GL_COLORBURN_NV 0x929A
05715 #define GL_COLORDODGE_NV 0x9299
05716 #define GL_CONJOINT_NV 0x9284
05717 #define GL_CONTRAST_NV 0x92A1
05718 #define GL_DARKEN_NV 0x9297
05719 #define GL_DIFFERENCE_NV 0x929E
05720 #define GL_DISJOINT_NV 0x9283
05721 #define GL_DST_ATOP_NV 0x928F
05722 #define GL_DST_IN_NV 0x928B
05723 #define GL_DST_NV 0x9287
05724 #define GL_DST_OUT_NV 0x928D
05725 #define GL_DST_OVER_NV 0x9289
05726 #define GL_EXCLUSION_NV 0x92A0
05727 #define GL_GREEN_NV 0x1904
05728 #define GL_HARDLIGHT_NV 0x929B
05729 #define GL_HARDMIX_NV 0x92A9
05730 #define GL_HSL_COLOR_NV 0x92AF
05731 #define GL_HSL_HUE_NV 0x92AD
05732 #define GL_HSL_LUMINOSITY_NV 0x92B0
05733 #define GL_HSL_SATURATION_NV 0x92AE
05734 #define GL_INVERT_OVG_NV 0x92B4
05735 #define GL_INVERT_RGB_NV 0x92A3
05736 #define GL_LIGHTEN_NV 0x9298
05737 #define GL_LINEARBURN_NV 0x92A5
05738 #define GL_LINEARDODGE_NV 0x92A4
05739 #define GL_LINEARLIGHT_NV 0x92A7
05740 #define GL_MINUS_CLAMPED_NV 0x92B3
05741 #define GL_MINUS_NV 0x929F
05742 #define GL_MULTIPLY_NV 0x9294
```

```
05743 #define GL_OVERLAY_NV 0x9296
05744 #define GL_PINLIGHT_NV 0x92A8
05745 #define GL_PLUS_CLAMPED_ALPHA_NV 0x92B2
05746 #define GL_PLUS_CLAMPED_NV 0x92B1
05747 #define GL_PLUS_DARKER_NV 0x9292
05748 #define GL_PLUS_NV 0x9291
05749 #define GL_RED_NV 0x1903
05750 #define GL_SCREEN_NV 0x9295
05751 #define GL_SOFTLIGHT_NV 0x929C
05752 #define GL_SRC_ATOP_NV 0x928E
05753 #define GL_SRC_IN_NV 0x928A
05754 #define GL_SRC_NV 0x9286
05755 #define GL_SRC_OUT_NV 0x928C
05756 #define GL_SRC_OVER_NV 0x9288
05757 #define GL_UNCORRELATED_NV 0x9282
05758 #define GL_VIVIDLIGHT_NV 0x92A6
05759 #define GL_XOR_NV 0x1506
05760 #define GL_BLEND_ADVANCED_COHERENT_NV 0x9285
05761 #define GL_VIEWPORT_POSITION_W_SCALE_NV 0x937C
05762 #define GL_VIEWPORT_POSITION_W_SCALE_X_COEFF_NV 0x937D
05763 #define GL_VIEWPORT_POSITION_W_SCALE_Y_COEFF_NV 0x937E
05764 #define GL_TERMINATE_SEQUENCE_COMMAND_NV 0x0000
05765 #define GL_NOP_COMMAND_NV 0x0001
05766 #define GL_DRAW_ELEMENTS_COMMAND_NV 0x0002
05767 #define GL_DRAW_ARRAYS_COMMAND_NV 0x0003
05768 #define GL_DRAW_ELEMENTS_STRIP_COMMAND_NV 0x0004
05769 #define GL_DRAW_ARRAYS_STRIP_COMMAND_NV 0x0005
05770 #define GL_DRAW_ELEMENTS_INSTANCED_COMMAND_NV 0x0006
05771 #define GL_DRAW_ARRAYS_INSTANCED_COMMAND_NV 0x0007
05772 #define GL_ELEMENT_ADDRESS_COMMAND_NV 0x0008
05773 #define GL_ATTRIBUTE_ADDRESS_COMMAND_NV 0x0009
05774 #define GL_UNIFORM_ADDRESS_COMMAND_NV 0x000A
05775 #define GL_BLEND_COLOR_COMMAND_NV 0x000B
05776 #define GL_STENCIL_REF_COMMAND_NV 0x000C
05777 #define GL_LINE_WIDTH_COMMAND_NV 0x000D
05778 #define GL_POLYGON_OFFSET_COMMAND_NV 0x000E
05779 #define GL_ALPHA_REF_COMMAND_NV 0x000F
05780 #define GL_VIEWPORT_COMMAND_NV 0x0010
05781 #define GL_SCISSOR_COMMAND_NV 0x0011
05782 #define GL_FRONT_FACE_COMMAND_NV 0x0012
05783 #define GL_COMPUTE_PROGRAM_NV 0x90FB
05784 #define GL_COMPUTE_PROGRAM_PARAMETER_BUFFER_NV 0x90FC
05785 #define GL_QUERY_WAIT_NV 0x8E13
05786 #define GL_QUERY_NO_WAIT_NV 0x8E14
05787 #define GL_QUERY_BY_REGION_WAIT_NV 0x8E15
05788 #define GL_QUERY_BY_REGION_NO_WAIT_NV 0x8E16
05789 #define GL_CONSERVATIVE_RASTERIZATION_NV 0x9346
05790 #define GL_SUBPIXEL_PRECISION_BIAS_X_BITS_NV 0x9347
05791 #define GL_SUBPIXEL_PRECISION_BIAS_Y_BITS_NV 0x9348
05792 #define GL_MAX_SUBPIXEL_PRECISION_BIAS_BITS_NV 0x9349
05793 #define GL_CONSERVATIVE_RASTER_DILATE_NV 0x9379
05794 #define GL_CONSERVATIVE_RASTER_DILATE_RANGE_NV 0x937A
05795 #define GL_CONSERVATIVE_RASTER_DILATE_GRANULARITY_NV 0x937B
05796 #define GL_CONSERVATIVE_RASTER_MODE_PRE_SNAP_NV 0x9550
05797 #define GL_CONSERVATIVE_RASTER_MODE_NV 0x954D
05798 #define GL_CONSERVATIVE_RASTER_MODE_POST_SNAP_NV 0x954E
05799 #define GL_CONSERVATIVE_RASTER_MODE_PRE_SNAP_TRIANGLES_NV 0x954F
05800 #define GL_DEPTH_STENCIL_TO_RGBA_NV 0x886E
05801 #define GL_DEPTH_STENCIL_TO_BGRA_NV 0x886F
05802 #define GL_MAX_DEEP_3D_TEXTURE_WIDTH_HEIGHT_NV 0x90D0
05803 #define GL_MAX_DEEP_3D_TEXTURE_DEPTH_NV 0x90D1
05804 #define GL_DEPTH_COMPONENT32F_NV 0x8DAB
05805 #define GL_DEPTH32F_STENCIL8_NV 0x8DAC
05806 #define GL_FLOAT_32_UNSIGNED_INT_24_8_REV_NV 0x8DAD
05807 #define GL_DEPTH_BUFFER_FLOAT_MODE_NV 0x8DAF
05808 #define GL_DEPTH_CLAMP_NV 0x864F
05809 #define GL_EVAL_2D_NV 0x86C0
05810 #define GL_EVAL_TRIANGULAR_2D_NV 0x86C1
05811 #define GL_MAP_TESSELLATION_NV 0x86C2
05812 #define GL_MAP_ATTRIB_U_ORDER_NV 0x86C3
05813 #define GL_MAP_ATTRIB_V_ORDER_NV 0x86C4
05814 #define GL_EVAL_FRACTIONAL_TESSELLATION_NV 0x86C5
05815 #define GL_EVAL_VERTEX_ATTRIB0_NV 0x86C6
05816 #define GL_EVAL_VERTEX_ATTRIB1_NV 0x86C7
05817 #define GL_EVAL_VERTEX_ATTRIB2_NV 0x86C8
05818 #define GL_EVAL_VERTEX_ATTRIB3_NV 0x86C9
05819 #define GL_EVAL_VERTEX_ATTRIB4_NV 0x86CA
05820 #define GL_EVAL_VERTEX_ATTRIB5_NV 0x86CB
05821 #define GL_EVAL_VERTEX_ATTRIB6_NV 0x86CC
05822 #define GL_EVAL_VERTEX_ATTRIB7_NV 0x86CD
05823 #define GL_EVAL_VERTEX_ATTRIB8_NV 0x86CE
05824 #define GL_EVAL_VERTEX_ATTRIB9_NV 0x86CF
05825 #define GL_EVAL_VERTEX_ATTRIB10_NV 0x86D0
05826 #define GL_EVAL_VERTEX_ATTRIB11_NV 0x86D1
05827 #define GL_EVAL_VERTEX_ATTRIB12_NV 0x86D2
05828 #define GL_EVAL_VERTEX_ATTRIB13_NV 0x86D3
05829 #define GL_EVAL_VERTEX_ATTRIB14_NV 0x86D4
```

```
05830 #define GL_EVAL_VERTEX_ATTRIB15_NV 0x86D5
05831 #define GL_MAX_MAP_TESSELLATION_NV 0x86D6
05832 #define GL_MAX_RATIONAL_EVAL_ORDER_NV 0x86D7
05833 #define GL_SAMPLE_POSITION_NV 0x8E50
05834 #define GL_SAMPLE_MASK_NV 0x8E51
05835 #define GL_SAMPLE_MASK_VALUE_NV 0x8E52
05836 #define GL_TEXTURE_BINDING_RENDERBUFFER_NV 0x8E53
05837 #define GL_TEXTURE_RENDERBUFFER_DATA_STORE_BINDING_NV 0x8E54
05838 #define GL_TEXTURE_RENDERBUFFER_NV 0x8E55
05839 #define GL_SAMPLER_RENDERBUFFER_NV 0x8E56
05840 #define GL_INT_SAMPLER_RENDERBUFFER_NV 0x8E57
05841 #define GL_UNSIGNED_INT_SAMPLER_RENDERBUFFER_NV 0x8E58
05842 #define GL_MAX_SAMPLE_MASK_WORDS_NV 0x8E59
05843 #define GL_ALL_COMPLETED_NV 0x84F2
05844 #define GL_FENCE_STATUS_NV 0x84F3
05845 #define GL_FENCE_CONDITION_NV 0x84F4
05846 #define GL_FILL_RECTANGLE_NV 0x933C
05847 #define GL_FLOAT_R_NV 0x8880
05848 #define GL_FLOAT_RG_NV 0x8881
05849 #define GL_FLOAT_RGB_NV 0x8882
05850 #define GL_FLOAT_RGBA_NV 0x8883
05851 #define GL_FLOAT_R16_NV 0x8884
05852 #define GL_FLOAT_R32_NV 0x8885
05853 #define GL_FLOAT_RG16_NV 0x8886
05854 #define GL_FLOAT_RG32_NV 0x8887
05855 #define GL_FLOAT_RGB16_NV 0x8888
05856 #define GL_FLOAT_RGB32_NV 0x8889
05857 #define GL_FLOAT_RGBA16_NV 0x888A
05858 #define GL_FLOAT_RGBA32_NV 0x888B
05859 #define GL_TEXTURE_FLOAT_COMPONENTS_NV 0x888C
05860 #define GL_FLOAT_CLEAR_COLOR_VALUE_NV 0x888D
05861 #define GL_FLOAT_RGBA_MODE_NV 0x888E
05862 #define GL_FOG_DISTANCE_MODE_NV 0x855A
05863 #define GL_EYE_RADIAL_NV 0x855B
05864 #define GL_EYE_PLANE_ABSOLUTE_NV 0x855C
05865 #define GL_EYE_PLANE 0x2502
05866 #define GL_FRAGMENT_COVERAGE_TO_COLOR_NV 0x92DD
05867 #define GL_FRAGMENT_COVERAGE_COLOR_NV 0x92DE
05868 #define GL_MAX_FRAGMENT_PROGRAM_LOCAL_PARAMETERS_NV 0x8868
05869 #define GL_FRAGMENT_PROGRAM_NV 0x8870
05870 #define GL_MAX_TEXTURE_COORDS_NV 0x8871
05871 #define GL_MAX_TEXTURE_IMAGE_UNITS_NV 0x8872
05872 #define GL_FRAGMENT_PROGRAM_BINDING_NV 0x8873
05873 #define GL_PROGRAM_ERROR_STRING_NV 0x8874
05874 #define GL_MAX_PROGRAM_EXEC_INSTRUCTIONS_NV 0x88F4
05875 #define GL_MAX_PROGRAM_CALL_DEPTH_NV 0x88F5
05876 #define GL_MAX_PROGRAM_IF_DEPTH_NV 0x88F6
05877 #define GL_MAX_PROGRAM_LOOP_DEPTH_NV 0x88F7
05878 #define GL_MAX_PROGRAM_LOOP_COUNT_NV 0x88F8
05879 #define GL_COVERAGE_MODULATION_TABLE_NV 0x9331
05880 #define GL_COLOR_SAMPLES_NV 0x8E20
05881 #define GL_DEPTH_SAMPLES_NV 0x932D
05882 #define GL_STENCIL_SAMPLES_NV 0x932E
05883 #define GL_MIXED_DEPTH_SAMPLES_SUPPORTED_NV 0x932F
05884 #define GL_MIXED_STENCIL_SAMPLES_SUPPORTED_NV 0x9330
05885 #define GL_COVERAGE_MODULATION_NV 0x9332
05886 #define GL_COVERAGE_MODULATION_TABLE_SIZE_NV 0x9333
05887 #define GL_RENDERBUFFER_COVERAGE_SAMPLES_NV 0x8CAB
05888 #define GL_RENDERBUFFER_COLOR_SAMPLES_NV 0x8E10
05889 #define GL_MAX_MULTISAMPLE_COVERAGE_MODES_NV 0x8E11
05890 #define GL_MULTISAMPLE_COVERAGE_MODES_NV 0x8E12
05891 #define GL_GEOMETRY_PROGRAM_NV 0x8C26
05892 #define GL_MAX_PROGRAM_OUTPUT_VERTICES_NV 0x8C27
05893 #define GL_MAX_PROGRAM_TOTAL_OUTPUT_COMPONENTS_NV 0x8C28
05894 #define GL_PER_GPU_STORAGE_BIT_NV 0x0800
05895 #define GL_MULTICAST_GPUS_NV 0x92BA
05896 #define GL_RENDER_GPU_MASK_NV 0x9558
05897 #define GL_PER_GPU_STORAGE_NV 0x9548
05898 #define GL_MULTICAST_PROGRAMMABLE_SAMPLE_LOCATION_NV 0x9549
05899 #define GL_MIN_PROGRAM_TEXEL_OFFSET_NV 0x8904
05900 #define GL_MAX_PROGRAM_TEXEL_OFFSET_NV 0x8905
05901 #define GL_PROGRAM_ATTRIB_COMPONENTS_NV 0x8906
05902 #define GL_PROGRAM_RESULT_COMPONENTS_NV 0x8907
05903 #define GL_MAX_PROGRAM_ATTRIB_COMPONENTS_NV 0x8908
05904 #define GL_MAX_PROGRAM_RESULT_COMPONENTS_NV 0x8909
05905 #define GL_MAX_PROGRAM_GENERIC_ATTRIBS_NV 0x8DA5
05906 #define GL_MAX_PROGRAM_GENERIC_RESULTS_NV 0x8DA6
05907 #define GL_MAX_GEOMETRY_PROGRAM_INVOCATIONS_NV 0x8E5A
05908 #define GL_MIN_FRAGMENT_INTERPOLATION_OFFSET_NV 0x8E5B
05909 #define GL_MAX_FRAGMENT_INTERPOLATION_OFFSET_NV 0x8E5C
05910 #define GL_FRAGMENT_PROGRAM_INTERPOLATION_OFFSET_BITS_NV 0x8E5D
05911 #define GL_MIN_PROGRAM_TEXTURE_GATHER_OFFSET_NV 0x8E5E
05912 #define GL_MAX_PROGRAM_TEXTURE_GATHER_OFFSET_NV 0x8E5F
05913 #define GL_MAX_PROGRAM_SUBROUTINE_PARAMETERS_NV 0x8F44
05914 #define GL_MAX_PROGRAM_SUBROUTINE_NUM_NV 0x8F45
05915 #define GL_HALF_FLOAT_NV 0x140B
05916 #define GL_MULTISAMPLES_NV 0x9371
```

```
05917 #define GL_SUPERSAMPLE_SCALE_X_NV 0x9372
05918 #define GL_SUPERSAMPLE_SCALE_Y_NV 0x9373
05919 #define GL_CONFORMANT_NV 0x9374
05920 #define GL_MAX_SHININESS_NV 0x8504
05921 #define GL_MAX_SPOT_EXPONENT_NV 0x8505
05922 #define GL_ATTACHED_MEMORY_OBJECT_NV 0x95A4
05923 #define GL_ATTACHED_MEMORY_OFFSET_NV 0x95A5
05924 #define GL_MEMORY_ATTACHABLE_ALIGNMENT_NV 0x95A6
05925 #define GL_MEMORY_ATTACHABLE_SIZE_NV 0x95A7
05926 #define GL_MEMORY_ATTACHABLE_NV 0x95A8
05927 #define GL_DETACHED_MEMORY_INCARNATION_NV 0x95A9
05928 #define GL_DETACHED_TEXTURES_NV 0x95AA
05929 #define GL_DETACHED_BUFFERS_NV 0x95AB
05930 #define GL_MAX_DETACHED_TEXTURES_NV 0x95AC
05931 #define GL_MAX_DETACHED_BUFFERS_NV 0x95AD
05932 #define GL_MESH_SHADER_NV 0x9559
05933 #define GL_TASK_SHADER_NV 0x955A
05934 #define GL_MAX_MESH_UNIFORM_BLOCKS_NV 0x8E60
05935 #define GL_MAX_MESH_TEXTURE_IMAGE_UNITS_NV 0x8E61
05936 #define GL_MAX_MESH_IMAGE_UNIFORMS_NV 0x8E62
05937 #define GL_MAX_MESH_UNIFORM_COMPONENTS_NV 0x8E63
05938 #define GL_MAX_MESH_ATOMIC_COUNTER_BUFFERS_NV 0x8E64
05939 #define GL_MAX_MESH_ATOMIC_COUNTERS_NV 0x8E65
05940 #define GL_MAX_MESH_SHADER_STORAGE_BLOCKS_NV 0x8E66
05941 #define GL_MAX_COMBINED_MESH_UNIFORM_COMPONENTS_NV 0x8E67
05942 #define GL_MAX_TASK_UNIFORM_BLOCKS_NV 0x8E68
05943 #define GL_MAX_TASK_TEXTURE_IMAGE_UNITS_NV 0x8E69
05944 #define GL_MAX_TASK_IMAGE_UNIFORMS_NV 0x8E6A
05945 #define GL_MAX_TASK_UNIFORM_COMPONENTS_NV 0x8E6B
05946 #define GL_MAX_TASK_ATOMIC_COUNTER_BUFFERS_NV 0x8E6C
05947 #define GL_MAX_TASK_ATOMIC_COUNTERS_NV 0x8E6D
05948 #define GL_MAX_TASK_SHADER_STORAGE_BLOCKS_NV 0x8E6E
05949 #define GL_MAX_COMBINED_TASK_UNIFORM_COMPONENTS_NV 0x8E6F
05950 #define GL_MAX_MESH_WORK_GROUP_INVOCATIONS_NV 0x95A2
05951 #define GL_MAX_TASK_WORK_GROUP_INVOCATIONS_NV 0x95A3
05952 #define GL_MAX_MESH_TOTAL_MEMORY_SIZE_NV 0x9536
05953 #define GL_MAX_TASK_TOTAL_MEMORY_SIZE_NV 0x9537
05954 #define GL_MAX_MESH_OUTPUT_VERTICES_NV 0x9538
05955 #define GL_MAX_MESH_OUTPUT_PRIMITIVES_NV 0x9539
05956 #define GL_MAX_TASK_OUTPUT_COUNT_NV 0x953A
05957 #define GL_MAX_DRAW_MESH_TASKS_COUNT_NV 0x953D
05958 #define GL_MAX_MESH_VIEWS_NV 0x9557
05959 #define GL_MESH_OUTPUT_PER_VERTEX_GRANULARITY_NV 0x92DF
05960 #define GL_MESH_OUTPUT_PER_PRIMITIVE_GRANULARITY_NV 0x9543
05961 #define GL_MAX_MESH_WORK_GROUP_SIZE_NV 0x953B
05962 #define GL_MAX_TASK_WORK_GROUP_SIZE_NV 0x953C
05963 #define GL_MESH_WORK_GROUP_SIZE_NV 0x953E
05964 #define GL_TASK_WORK_GROUP_SIZE_NV 0x953F
05965 #define GL_MESH_VERTICES_OUT_NV 0x9579
05966 #define GL_MESH_PRIMITIVES_OUT_NV 0x957A
05967 #define GL_MESH_OUTPUT_TYPE_NV 0x957B
05968 #define GL_UNIFORM_BLOCK_REFERENCED_BY_MESH_SHADER_NV 0x959C
05969 #define GL_UNIFORM_BLOCK_REFERENCED_BY_TASK_SHADER_NV 0x959D
05970 #define GL_REFERENCED_BY_MESH_SHADER_NV 0x95A0
05971 #define GL_REFERENCED_BY_TASK_SHADER_NV 0x95A1
05972 #define GL_MESH_SHADER_BIT_NV 0x00000040
05973 #define GL_TASK_SHADER_BIT_NV 0x00000080
05974 #define GL_MESH_SUBROUTINE_NV 0x957C
05975 #define GL_TASK_SUBROUTINE_NV 0x957D
05976 #define GL_MESH_SUBROUTINE_UNIFORM_NV 0x957E
05977 #define GL_TASK_SUBROUTINE_UNIFORM_NV 0x957F
05978 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_MESH_SHADER_NV 0x959E
05979 #define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_TASK_SHADER_NV 0x959F
05980 #define GL_MULTISAMPLE_FILTER_HINT_NV 0x8534
05981 #define GL_PIXEL_COUNTER_BITS_NV 0x8864
05982 #define GL_CURRENT_OCCLUSION_QUERY_ID_NV 0x8865
05983 #define GL_PIXEL_COUNT_NV 0x8866
05984 #define GL_PIXEL_COUNT_AVAILABLE_NV 0x8867
05985 #define GL_DEPTH_STENCIL_NV 0x84F9
05986 #define GL_UNSIGNED_INT_24_8_NV 0x84FA
05987 #define GL_MAX_PROGRAM_PARAMETER_BUFFER_BINDINGS_NV 0x8DA0
05988 #define GL_MAX_PROGRAM_PARAMETER_BUFFER_SIZE_NV 0x8DA1
05989 #define GL_VERTEX_PROGRAM_PARAMETER_BUFFER_NV 0x8DA2
05990 #define GL_GEOMETRY_PROGRAM_PARAMETER_BUFFER_NV 0x8DA3
05991 #define GL_FRAGMENT_PROGRAM_PARAMETER_BUFFER_NV 0x8DA4
05992 #define GL_PATH_FORMAT_SVG_NV 0x9070
05993 #define GL_PATH_FORMAT_PS_NV 0x9071
05994 #define GL_STANDARD_FONT_NAME_NV 0x9072
05995 #define GL_SYSTEM_FONT_NAME_NV 0x9073
05996 #define GL_FILE_NAME_NV 0x9074
05997 #define GL_PATH_STROKE_WIDTH_NV 0x9075
05998 #define GL_PATH_END_CAPS_NV 0x9076
05999 #define GL_PATH_INITIAL_END_CAP_NV 0x9077
06000 #define GL_PATH_TERMINAL_END_CAP_NV 0x9078
06001 #define GL_PATH_JOIN_STYLE_NV 0x9079
06002 #define GL_PATH_MITER_LIMIT_NV 0x907A
06003 #define GL_PATH_DASH_CAPS_NV 0x907B
```



```
06004 #define GL_PATH_INITIAL_DASH_CAP_NV 0x907C
06005 #define GL_PATH_TERMINAL_DASH_CAP_NV 0x907D
06006 #define GL_PATH_DASH_OFFSET_NV 0x907E
06007 #define GL_PATH_CLIENT_LENGTH_NV 0x907F
06008 #define GL_PATH_FILL_MODE_NV 0x9080
06009 #define GL_PATH_FILL_MASK_NV 0x9081
06010 #define GL_PATH_FILL_COVER_MODE_NV 0x9082
06011 #define GL_PATH_STROKE_COVER_MODE_NV 0x9083
06012 #define GL_PATH_STROKE_MASK_NV 0x9084
06013 #define GL_COUNT_UP_NV 0x9088
06014 #define GL_COUNT_DOWN_NV 0x9089
06015 #define GL_PATH_OBJECT_BOUNDING_BOX_NV 0x908A
06016 #define GL_CONVEX_HULL_NV 0x908B
06017 #define GL_BOUNDING_BOX_NV 0x908D
06018 #define GL_TRANSLATE_X_NV 0x908E
06019 #define GL_TRANSLATE_Y_NV 0x908F
06020 #define GL_TRANSLATE_2D_NV 0x9090
06021 #define GL_TRANSLATE_3D_NV 0x9091
06022 #define GL_AFFINE_2D_NV 0x9092
06023 #define GL_AFFINE_3D_NV 0x9094
06024 #define GL_TRANSPOSE_AFFINE_2D_NV 0x9096
06025 #define GL_TRANSPOSE_AFFINE_3D_NV 0x9098
06026 #define GL_UTF8_NV 0x909A
06027 #define GL_UTF16_NV 0x909B
06028 #define GL_BOUNDING_BOX_OF_BOUNDING_BOXES_NV 0x909C
06029 #define GL_PATH_COMMAND_COUNT_NV 0x909D
06030 #define GL_PATH_COORD_COUNT_NV 0x909E
06031 #define GL_PATH_DASH_ARRAY_COUNT_NV 0x909F
06032 #define GL_PATH_COMPUTED_LENGTH_NV 0x90A0
06033 #define GL_PATH_FILL_BOUNDING_BOX_NV 0x90A1
06034 #define GL_PATH_STROKE_BOUNDING_BOX_NV 0x90A2
06035 #define GL_SQUARE_NV 0x90A3
06036 #define GL_ROUND_NV 0x90A4
06037 #define GL_TRIANGULAR_NV 0x90A5
06038 #define GL_BEVEL_NV 0x90A6
06039 #define GL_MITER_REVERT_NV 0x90A7
06040 #define GL_MITER_TRUNCATE_NV 0x90A8
06041 #define GL_SKIP_MISSING_GLYPH_NV 0x90A9
06042 #define GL_USE_MISSING_GLYPH_NV 0x90AA
06043 #define GL_PATH_ERROR_POSITION_NV 0x90AB
06044 #define GL_ACCUM_ADJACENT_PAIRS_NV 0x90AD
06045 #define GL_ADJACENT_PAIRS_NV 0x90AE
06046 #define GL_FIRST_TO_REST_NV 0x90AF
06047 #define GL_PATH_GEN_MODE_NV 0x90B0
06048 #define GL_PATH_GEN_COEFF_NV 0x90B1
06049 #define GL_PATH_GEN_COMPONENTS_NV 0x90B3
06050 #define GL_PATH_STENCIL_FUNC_NV 0x90B7
06051 #define GL_PATH_STENCIL_REF_NV 0x90B8
06052 #define GL_PATH_STENCIL_VALUE_MASK_NV 0x90B9
06053 #define GL_PATH_STENCIL_DEPTH_OFFSET_FACTOR_NV 0x90BD
06054 #define GL_PATH_STENCIL_DEPTH_OFFSET_UNITS_NV 0x90BE
06055 #define GL_PATH_COVER_DEPTH_FUNC_NV 0x90BF
06056 #define GL_PATH_DASH_OFFSET_RESET_NV 0x90B4
06057 #define GL_MOVE_TO_RESETS_NV 0x90B5
06058 #define GL_MOVE_TO_CONTINUES_NV 0x90B6
06059 #define GL_CLOSE_PATH_NV 0x00
06060 #define GL_MOVE_TO_NV 0x02
06061 #define GL_RELATIVE_MOVE_TO_NV 0x03
06062 #define GL_LINE_TO_NV 0x04
06063 #define GL_RELATIVE_LINE_TO_NV 0x05
06064 #define GL_HORIZONTAL_LINE_TO_NV 0x06
06065 #define GL_RELATIVE_HORIZONTAL_LINE_TO_NV 0x07
06066 #define GL_VERTICAL_LINE_TO_NV 0x08
06067 #define GL_RELATIVE_VERTICAL_LINE_TO_NV 0x09
06068 #define GL_QUADRATIC_CURVE_TO_NV 0x0A
06069 #define GL_RELATIVE_QUADRATIC_CURVE_TO_NV 0x0B
06070 #define GL_CUBIC_CURVE_TO_NV 0x0C
06071 #define GL_RELATIVE_CUBIC_CURVE_TO_NV 0x0D
06072 #define GL_SMOOTH_QUADRATIC_CURVE_TO_NV 0x0E
06073 #define GL_RELATIVE_SMOOTH_QUADRATIC_CURVE_TO_NV 0x0F
06074 #define GL_SMOOTH_CUBIC_CURVE_TO_NV 0x10
06075 #define GL_RELATIVE_SMOOTH_CUBIC_CURVE_TO_NV 0x11
06076 #define GL_SMALL_CCW_ARC_TO_NV 0x12
06077 #define GL_RELATIVE_SMALL_CCW_ARC_TO_NV 0x13
06078 #define GL_SMALL_CW_ARC_TO_NV 0x14
06079 #define GL_RELATIVE_SMALL_CW_ARC_TO_NV 0x15
06080 #define GL_LARGE_CCW_ARC_TO_NV 0x16
06081 #define GL_RELATIVE_LARGE_CCW_ARC_TO_NV 0x17
06082 #define GL_LARGE_CW_ARC_TO_NV 0x18
06083 #define GL_RELATIVE_LARGE_CW_ARC_TO_NV 0x19
06084 #define GL_RESTART_PATH_NV 0xF0
06085 #define GL_DUP_FIRST_CUBIC_CURVE_TO_NV 0xF2
06086 #define GL_DUP_LAST_CUBIC_CURVE_TO_NV 0xF4
06087 #define GL_RECT_NV 0xF6
06088 #define GL_CIRCULAR_CCW_ARC_TO_NV 0xF8
06089 #define GL_CIRCULAR_CW_ARC_TO_NV 0xFA
06090 #define GL_CIRCULAR_TANGENT_ARC_TO_NV 0xFC
```

```
06091 #define GL_ARC_TO_NV 0xFE
06092 #define GL_RELATIVE_ARC_TO_NV 0xFF
06093 #define GL_BOLD_BIT_NV 0x01
06094 #define GL_ITALIC_BIT_NV 0x02
06095 #define GL_GLYPH_WIDTH_BIT_NV 0x01
06096 #define GL_GLYPH_HEIGHT_BIT_NV 0x02
06097 #define GL_GLYPH_HORIZONTAL_BEARING_X_BIT_NV 0x04
06098 #define GL_GLYPH_HORIZONTAL_BEARING_Y_BIT_NV 0x08
06099 #define GL_GLYPH_HORIZONTAL_BEARING_ADVANCE_BIT_NV 0x10
06100 #define GL_GLYPH_VERTICAL_BEARING_X_BIT_NV 0x20
06101 #define GL_GLYPH_VERTICAL_BEARING_Y_BIT_NV 0x40
06102 #define GL_GLYPH_VERTICAL_BEARING_ADVANCE_BIT_NV 0x80
06103 #define GL_GLYPH_HAS_KERNING_BIT_NV 0x100
06104 #define GL_FONT_X_MIN_BOUNDS_BIT_NV 0x00010000
06105 #define GL_FONT_Y_MIN_BOUNDS_BIT_NV 0x00020000
06106 #define GL_FONT_X_MAX_BOUNDS_BIT_NV 0x00040000
06107 #define GL_FONT_Y_MAX_BOUNDS_BIT_NV 0x00080000
06108 #define GL_FONT_UNITS_PER_EM_BIT_NV 0x00100000
06109 #define GL_FONT_ASCENDER_BIT_NV 0x00200000
06110 #define GL_FONT_DESCENDER_BIT_NV 0x00400000
06111 #define GL_FONT_HEIGHT_BIT_NV 0x00800000
06112 #define GL_FONT_MAX_ADVANCE_WIDTH_BIT_NV 0x01000000
06113 #define GL_FONT_MAX_ADVANCE_HEIGHT_BIT_NV 0x02000000
06114 #define GL_FONT_UNDERLINE_POSITION_BIT_NV 0x04000000
06115 #define GL_FONT_UNDERLINE_THICKNESS_BIT_NV 0x08000000
06116 #define GL_FONT_HAS_KERNING_BIT_NV 0x10000000
06117 #define GL_ROUNDED_RECT_NV 0xE8
06118 #define GL_RELATIVE_ROUNDED_RECT_NV 0xE9
06119 #define GL_ROUNDED_RECT2_NV 0xEA
06120 #define GL_RELATIVE_ROUNDED_RECT2_NV 0xEB
06121 #define GL_ROUNDED_RECT4_NV 0xEC
06122 #define GL_RELATIVE_ROUNDED_RECT4_NV 0xED
06123 #define GL_ROUNDED_RECT8_NV 0xEE
06124 #define GL_RELATIVE_ROUNDED_RECT8_NV 0xEF
06125 #define GL_RELATIVE_RECT_NV 0xF7
06126 #define GL_FONT_GLYPHS_AVAILABLE_NV 0x9368
06127 #define GL_FONT_TARGET_UNAVAILABLE_NV 0x9369
06128 #define GL_FONT_UNAVAILABLE_NV 0x936A
06129 #define GL_FONT_UNINTELLIGIBLE_NV 0x936B
06130 #define GL_CONIC_CURVE_TO_NV 0x1A
06131 #define GL_RELATIVE_CONIC_CURVE_TO_NV 0x1B
06132 #define GL_FONT_NUM_GLYPH_INDICES_BIT_NV 0x20000000
06133 #define GL_STANDARD_FONT_FORMAT_NV 0x936C
06134 #define GL_2_BYTES_NV 0x1407
06135 #define GL_3_BYTES_NV 0x1408
06136 #define GL_4_BYTES_NV 0x1409
06137 #define GL_EYE_LINEAR_NV 0x2400
06138 #define GL_OBJECT_LINEAR_NV 0x2401
06139 #define GL_CONSTANT_NV 0x8576
06140 #define GL_PATH_FOG_GEN_MODE_NV 0x90AC
06141 #define GL_PRIMARY_COLOR 0x8577
06142 #define GL_PRIMARY_COLOR_NV 0x852C
06143 #define GL_SECONDARY_COLOR_NV 0x852D
06144 #define GL_PATH_GEN_COLOR_FORMAT_NV 0x90B2
06145 #define GL_PATH_PROJECTION_NV 0x1701
06146 #define GL_PATH_MODELVIEW_NV 0x1700
06147 #define GL_PATH_MODELVIEW_STACK_DEPTH_NV 0x0BA3
06148 #define GL_PATH_MODELVIEW_MATRIX_NV 0x0BA6
06149 #define GL_PATH_MAX_MODELVIEW_STACK_DEPTH_NV 0x0D36
06150 #define GL_PATH_TRANSPOSE_MODELVIEW_MATRIX_NV 0x84E3
06151 #define GL_PATH_PROJECTION_STACK_DEPTH_NV 0x0BA4
06152 #define GL_PATH_PROJECTION_MATRIX_NV 0x0BA7
06153 #define GL_PATH_MAX_PROJECTION_STACK_DEPTH_NV 0x0D38
06154 #define GL_PATH_TRANSPOSE_PROJECTION_MATRIX_NV 0x84E4
06155 #define GL_FRAGMENT_INPUT_NV 0x936D
06156 #define GL_SHARED_EDGE_NV 0xC0
06157 #define GL_WRITE_PIXEL_DATA_RANGE_NV 0x8878
06158 #define GL_READ_PIXEL_DATA_RANGE_NV 0x8879
06159 #define GL_WRITE_PIXEL_DATA_RANGE_LENGTH_NV 0x887A
06160 #define GL_READ_PIXEL_DATA_RANGE_LENGTH_NV 0x887B
06161 #define GL_WRITE_PIXEL_DATA_RANGE_POINTER_NV 0x887C
06162 #define GL_READ_PIXEL_DATA_RANGE_POINTER_NV 0x887D
06163 #define GL_POINT_SPRITE_NV 0x8861
06164 #define GL_COORD_REPLACE_NV 0x8862
06165 #define GL_POINT_SPRITE_R_MODE_NV 0x8863
06166 #define GL_FRAME_NV 0x8E26
06167 #define GL_FIELDS_NV 0x8E27
06168 #define GL_CURRENT_TIME_NV 0x8E28
06169 #define GL_NUM_FILL_STREAMS_NV 0x8E29
06170 #define GL_PRESENT_TIME_NV 0x8E2A
06171 #define GL_PRESENT_DURATION_NV 0x8E2B
06172 #define GL_PRIMITIVE_RESTART_NV 0x8558
06173 #define GL_PRIMITIVE_RESTART_INDEX_NV 0x8559
06174 #define GL_SHADING_RATE_IMAGE_PER_PRIMITIVE_NV 0x95B1
06175 #define GL_SHADING_RATE_IMAGE_PALETTE_COUNT_NV 0x95B2
06176 #define GL_QUERY_RESOURCE_TYPE_VIDMEM_ALLOC_NV 0x9540
06177 #define GL_QUERY_RESOURCE_MEMTYPE_VIDMEM_NV 0x9542
```

```
06178 #define GL_QUERY_RESOURCE_SYS_RESERVED_NV 0x9544
06179 #define GL_QUERY_RESOURCE_TEXTURE_NV 0x9545
06180 #define GL_QUERY_RESOURCE_RENDERBUFFER_NV 0x9546
06181 #define GL_QUERY_RESOURCE_BUFFEROBJECT_NV 0x9547
06182 #define GL_REGISTER_COMBINERS_NV 0x8522
06183 #define GL_VARIABLE_A_NV 0x8523
06184 #define GL_VARIABLE_B_NV 0x8524
06185 #define GL_VARIABLE_C_NV 0x8525
06186 #define GL_VARIABLE_D_NV 0x8526
06187 #define GL_VARIABLE_E_NV 0x8527
06188 #define GL_VARIABLE_F_NV 0x8528
06189 #define GL_VARIABLE_G_NV 0x8529
06190 #define GL_CONSTANT_COLOR0_NV 0x852A
06191 #define GL_CONSTANT_COLOR1_NV 0x852B
06192 #define GL_SPARE0_NV 0x852E
06193 #define GL_SPARE1_NV 0x852F
06194 #define GL_DISCARD_NV 0x8530
06195 #define GL_E_TIMES_F_NV 0x8531
06196 #define GL_SPARE0_PLUS_SECONDARY_COLOR_NV 0x8532
06197 #define GL_UNSIGNED_IDENTITY_NV 0x8536
06198 #define GL_UNSIGNED_INVERT_NV 0x8537
06199 #define GL_EXPAND_NORMAL_NV 0x8538
06200 #define GL_EXPAND_NEGATE_NV 0x8539
06201 #define GL_HALF_BIAS_NORMAL_NV 0x853A
06202 #define GL_HALF_BIAS_NEGATE_NV 0x853B
06203 #define GL_SIGNED_IDENTITY_NV 0x853C
06204 #define GL_SIGNED_NEGATE_NV 0x853D
06205 #define GL_SCALE_BY_TWO_NV 0x853E
06206 #define GL_SCALE_BY_FOUR_NV 0x853F
06207 #define GL_SCALE_BY_ONE_HALF_NV 0x8540
06208 #define GL_BIAS_BY_NEGATIVE_ONE_HALF_NV 0x8541
06209 #define GL_COMBINER_INPUT_NV 0x8542
06210 #define GL_COMBINER_MAPPING_NV 0x8543
06211 #define GL_COMBINER_COMPONENT_USAGE_NV 0x8544
06212 #define GL_COMBINER_AB_DOT_PRODUCT_NV 0x8545
06213 #define GL_COMBINER_CD_DOT_PRODUCT_NV 0x8546
06214 #define GL_COMBINER_MUX_SUM_NV 0x8547
06215 #define GL_COMBINER_SCALE_NV 0x8548
06216 #define GL_COMBINER_BIAS_NV 0x8549
06217 #define GL_COMBINER_AB_OUTPUT_NV 0x854A
06218 #define GL_COMBINER_CD_OUTPUT_NV 0x854B
06219 #define GL_COMBINER_SUM_OUTPUT_NV 0x854C
06220 #define GL_MAX_GENERAL_COMBINERS_NV 0x854D
06221 #define GL_NUM_GENERAL_COMBINERS_NV 0x854E
06222 #define GL_COLOR_SUM_CLAMP_NV 0x854F
06223 #define GL_COMBINER0_NV 0x8550
06224 #define GL_COMBINER1_NV 0x8551
06225 #define GL_COMBINER2_NV 0x8552
06226 #define GL_COMBINER3_NV 0x8553
06227 #define GL_COMBINER4_NV 0x8554
06228 #define GL_COMBINER5_NV 0x8555
06229 #define GL_COMBINER6_NV 0x8556
06230 #define GL_COMBINER7_NV 0x8557
06231 #define GL_FOG 0x0B60
06232 #define GL_PER_STAGE_CONSTANTS_NV 0x8535
06233 #define GL_REPRESENTATIVE_FRAGMENT_TEST_NV 0x937F
06234 #define GL_PURGED_CONTEXT_RESET_NV 0x92BB
06235 #define GL_SAMPLE_LOCATION_SUBPIXEL_BITS_NV 0x933D
06236 #define GL_SAMPLE_LOCATION_PIXEL_GRID_WIDTH_NV 0x933E
06237 #define GL_SAMPLE_LOCATION_PIXEL_GRID_HEIGHT_NV 0x933F
06238 #define GL_PROGRAMMABLE_SAMPLE_LOCATION_TABLE_SIZE_NV 0x9340
06239 #define GL_SAMPLE_LOCATION_NV 0x8E50
06240 #define GL_PROGRAMMABLE_SAMPLE_LOCATION_NV 0x9341
06241 #define GL_FRAMEBUFFER_PROGRAMMABLE_SAMPLE_LOCATIONS_NV 0x9342
06242 #define GL_FRAMEBUFFER_SAMPLE_LOCATION_PIXEL_GRID_NV 0x9343
06243 #define GL_SCISSOR_TEST_EXCLUSIVE_NV 0x9555
06244 #define GL_SCISSOR_BOX_EXCLUSIVE_NV 0x9556
06245 #define GL_BUFFER_GPU_ADDRESS_NV 0x8F1D
06246 #define GL_GPU_ADDRESS_NV 0x8F34
06247 #define GL_MAX_SHADER_BUFFER_ADDRESS_NV 0x8F35
06248 #define GL_SHADER_GLOBAL_ACCESS_BARRIER_BIT_NV 0x00000010
06249 #define GL_SUBGROUP_FEATURE_PARTITIONED_BIT_NV 0x00000100
06250 #define GL_WARP_SIZE_NV 0x9339
06251 #define GL WARPS_PER_SM_NV 0x933A
06252 #define GL_SM_COUNT_NV 0x933B
06253 #define GL_SHADING_RATE_IMAGE_NV 0x9563
06254 #define GL_SHADING_RATE_NO_INVOCATIONS_NV 0x9564
06255 #define GL_SHADING_RATE_1_INVOCATION_PER_PIXEL_NV 0x9565
06256 #define GL_SHADING_RATE_1_INVOCATION_PER_1X2_PIXELS_NV 0x9566
06257 #define GL_SHADING_RATE_1_INVOCATION_PER_2X1_PIXELS_NV 0x9567
06258 #define GL_SHADING_RATE_1_INVOCATION_PER_2X2_PIXELS_NV 0x9568
06259 #define GL_SHADING_RATE_1_INVOCATION_PER_2X4_PIXELS_NV 0x9569
06260 #define GL_SHADING_RATE_1_INVOCATION_PER_4X2_PIXELS_NV 0x956A
06261 #define GL_SHADING_RATE_1_INVOCATION_PER_4X4_PIXELS_NV 0x956B
06262 #define GL_SHADING_RATE_2_INVOCATIONS_PER_PIXEL_NV 0x956C
06263 #define GL_SHADING_RATE_4_INVOCATIONS_PER_PIXEL_NV 0x956D
06264 #define GL_SHADING_RATE_8_INVOCATIONS_PER_PIXEL_NV 0x956E
```



```
06265 #define GL_SHADING_RATE_16_INVOCATIONS_PER_PIXEL_NV 0x956F
06266 #define GL_SHADING_RATE_IMAGE_BINDING_NV 0x955B
06267 #define GL_SHADING_RATE_IMAGE_TEXEL_WIDTH_NV 0x955C
06268 #define GL_SHADING_RATE_IMAGE_TEXEL_HEIGHT_NV 0x955D
06269 #define GL_SHADING_RATE_IMAGE_PALETTE_SIZE_NV 0x955E
06270 #define GL_MAX_COARSE_FRAGMENT_SAMPLES_NV 0x955F
06271 #define GL_SHADING_RATE_SAMPLE_ORDER_DEFAULT_NV 0x95AE
06272 #define GL_SHADING_RATE_SAMPLE_ORDER_PIXEL_MAJOR_NV 0x95AF
06273 #define GL_SHADING_RATE_SAMPLE_ORDER_SAMPLE_MAJOR_NV 0x95B0
06274 #define GL_MAX_PROGRAM_PATCH_ATTRIBS_NV 0x86D8
06275 #define GL_TESS_CONTROL_PROGRAM_NV 0x891E
06276 #define GL_TESS_EVALUATION_PROGRAM_NV 0x891F
06277 #define GL_TESS_CONTROL_PROGRAM_PARAMETER_BUFFER_NV 0x8C74
06278 #define GL_TESS_EVALUATION_PROGRAM_PARAMETER_BUFFER_NV 0x8C75
06279 #define GL_EMBOSS_LIGHT_NV 0x855D
06280 #define GL_EMBOSS_CONSTANT_NV 0x855E
06281 #define GL_EMBOSS_MAP_NV 0x855F
06282 #define GL_NORMAL_MAP_NV 0x8511
06283 #define GL_REFLECTION_MAP_NV 0x8512
06284 #define GL_COMBINE4_NV 0x8503
06285 #define GL_SOURCE3_RGB_NV 0x8583
06286 #define GL_SOURCE3_ALPHA_NV 0x858B
06287 #define GL_OPERAND3_RGB_NV 0x8593
06288 #define GL_OPERAND3_ALPHA_NV 0x859B
06289 #define GL_TEXTURE_UNSIGNED_REMAP_MODE_NV 0x888F
06290 #define GL_TEXTURE_COVERAGE_SAMPLES_NV 0x9045
06291 #define GL_TEXTURE_COLOR_SAMPLES_NV 0x9046
06292 #define GL_TEXTURE_RECTANGLE_NV 0x84F5
06293 #define GL_TEXTURE_BINDING_RECTANGLE_NV 0x84F6
06294 #define GL_PROXY_TEXTURE_RECTANGLE_NV 0x84F7
06295 #define GL_MAX_RECTANGLE_TEXTURE_SIZE_NV 0x84F8
06296 #define GL_OFFSET_TEXTURE_RECTANGLE_NV 0x864C
06297 #define GL_OFFSET_TEXTURE_RECTANGLE_SCALE_NV 0x864D
06298 #define GL_DOT_PRODUCT_TEXTURE_RECTANGLE_NV 0x864E
06299 #define GL_RGBA_UNSIGNED_DOT_PRODUCT_MAPPING_NV 0x86D9
06300 #define GL_UNSIGNED_INT_S8_S8_8_8_NV 0x86DA
06301 #define GL_UNSIGNED_INT_8_8_S8_S8_REV_NV 0x86DB
06302 #define GL_DSDT_MAG_INTENSITY_NV 0x86DC
06303 #define GL_SHADER_CONSISTENT_NV 0x86DD
06304 #define GL_TEXTURE_SHADER_NV 0x86DE
06305 #define GL_SHADER_OPERATION_NV 0x86DF
06306 #define GL_CULL_MODES_NV 0x86E0
06307 #define GL_OFFSET_TEXTURE_MATRIX_NV 0x86E1
06308 #define GL_OFFSET_TEXTURE_SCALE_NV 0x86E2
06309 #define GL_OFFSET_TEXTURE_BIAS_NV 0x86E3
06310 #define GL_OFFSET_TEXTURE_2D_MATRIX_NV 0x86E1
06311 #define GL_OFFSET_TEXTURE_2D_SCALE_NV 0x86E2
06312 #define GL_OFFSET_TEXTURE_2D_BIAS_NV 0x86E3
06313 #define GL_PREVIOUS_TEXTURE_INPUT_NV 0x86E4
06314 #define GL_CONST_EYE_NV 0x86E5
06315 #define GL_PASS_THROUGH_NV 0x86E6
06316 #define GL_CULL_FRAGMENT_NV 0x86E7
06317 #define GL_OFFSET_TEXTURE_2D_NV 0x86E8
06318 #define GL_DEPENDENT_AR_TEXTURE_2D_NV 0x86E9
06319 #define GL_DEPENDENT_GB_TEXTURE_2D_NV 0x86EA
06320 #define GL_DOT_PRODUCT_NV 0x86EC
06321 #define GL_DOT_PRODUCT_DEPTH_REPLACE_NV 0x86ED
06322 #define GL_DOT_PRODUCT_TEXTURE_2D_NV 0x86EE
06323 #define GL_DOT_PRODUCT_TEXTURE_CUBE_MAP_NV 0x86F0
06324 #define GL_DOT_PRODUCT_DIFFUSE_CUBE_MAP_NV 0x86F1
06325 #define GL_DOT_PRODUCT_REFLECT_CUBE_MAP_NV 0x86F2
06326 #define GL_DOT_PRODUCT_CONST_EYE_REFLECT_CUBE_MAP_NV 0x86F3
06327 #define GL_HILO_NV 0x86F4
06328 #define GL_DSDT_NV 0x86F5
06329 #define GL_DSDT_MAG_NV 0x86F6
06330 #define GL_DSDT_MAG_VIB_NV 0x86F7
06331 #define GL_HILO16_NV 0x86F8
06332 #define GL_SIGNED_HILO_NV 0x86F9
06333 #define GL_SIGNED_HILO16_NV 0x86FA
06334 #define GL_SIGNED_RGBA_NV 0x86FB
06335 #define GL_SIGNED_RGBA8_NV 0x86FC
06336 #define GL_SIGNED_RGB_NV 0x86FE
06337 #define GL_SIGNED_RGB8_NV 0x86FF
06338 #define GL_SIGNED_LUMINANCE_NV 0x8701
06339 #define GL_SIGNED_LUMINANCE8_NV 0x8702
06340 #define GL_SIGNED_LUMINANCE_ALPHA_NV 0x8703
06341 #define GL_SIGNED_LUMINANCE8_ALPHA8_NV 0x8704
06342 #define GL_SIGNED_ALPHA_NV 0x8705
06343 #define GL_SIGNED_ALPHA8_NV 0x8706
06344 #define GL_SIGNED_INTENSITY_NV 0x8707
06345 #define GL_SIGNED_INTENSITY8_NV 0x8708
06346 #define GL_DSDT8_NV 0x8709
06347 #define GL_DSDT8_MAG8_NV 0x870A
06348 #define GL_DSDT8_MAG8_INTENSITY8_NV 0x870B
06349 #define GL_SIGNED_RGB_UNSIGNED_ALPHA_NV 0x870C
06350 #define GL_SIGNED_RGB8_UNSIGNED_ALPHA8_NV 0x870D
06351 #define GL_HI_SCALE_NV 0x870E
```

```
06352 #define GL_LO_SCALE_NV 0x870F
06353 #define GL_DS_SCALE_NV 0x8710
06354 #define GL_DT_SCALE_NV 0x8711
06355 #define GL_MAGNITUDE_SCALE_NV 0x8712
06356 #define GL_VIBRANCE_SCALE_NV 0x8713
06357 #define GL_HI_BIAS_NV 0x8714
06358 #define GL_LO_BIAS_NV 0x8715
06359 #define GL_DS_BIAS_NV 0x8716
06360 #define GL_DT_BIAS_NV 0x8717
06361 #define GL_MAGNITUDE_BIAS_NV 0x8718
06362 #define GL_VIBRANCE_BIAS_NV 0x8719
06363 #define GL_TEXTURE_BORDER_VALUES_NV 0x871A
06364 #define GL_TEXTURE_HI_SIZE_NV 0x871B
06365 #define GL_TEXTURE_LO_SIZE_NV 0x871C
06366 #define GL_TEXTURE_DS_SIZE_NV 0x871D
06367 #define GL_TEXTURE_DT_SIZE_NV 0x871E
06368 #define GL_TEXTURE_MAG_SIZE_NV 0x871F
06369 #define GL_DOT_PRODUCT_TEXTURE_3D_NV 0x86EF
06370 #define GL_OFFSET_PROJECTIVE_TEXTURE_2D_NV 0x8850
06371 #define GL_OFFSET_PROJECTIVE_TEXTURE_2D_SCALE_NV 0x8851
06372 #define GL_OFFSET_PROJECTIVE_TEXTURE_RECTANGLE_NV 0x8852
06373 #define GL_OFFSET_PROJECTIVE_TEXTURE_RECTANGLE_SCALE_NV 0x8853
06374 #define GL_OFFSET_HILO_TEXTURE_2D_NV 0x8854
06375 #define GL_OFFSET_HILO_TEXTURE_RECTANGLE_NV 0x8855
06376 #define GL_OFFSET_HILO_PROJECTIVE_TEXTURE_2D_NV 0x8856
06377 #define GL_OFFSET_HILO_PROJECTIVE_TEXTURE_RECTANGLE_NV 0x8857
06378 #define GL_DEPENDENT_HILO_TEXTURE_2D_NV 0x8858
06379 #define GL_DEPENDENT_RGB_TEXTURE_3D_NV 0x8859
06380 #define GL_DEPENDENT_RGB_TEXTURE_CUBE_MAP_NV 0x885A
06381 #define GL_DOT_PRODUCT_PASS_THROUGH_NV 0x885B
06382 #define GL_DOT_PRODUCT_TEXTURE_1D_NV 0x885C
06383 #define GL_DOT_PRODUCT_AFFINE_DEPTH_REPLACE_NV 0x885D
06384 #define GL_HILO8_NV 0x885E
06385 #define GL_SIGNED_HILO8_NV 0x885F
06386 #define GL_FORCE_BLUE_TO_ONE_NV 0x8860
06387 #define GL_TIMELINE_SEMAPHORE_VALUE_NV 0x9595
06388 #define GL_SEMAPHORE_TYPE_NV 0x95B3
06389 #define GL_SEMAPHORE_TYPE_BINARY_NV 0x95B4
06390 #define GL_SEMAPHORE_TYPE_TIMELINE_NV 0x95B5
06391 #define GL_MAX_TIMELINE_SEMAPHORE_VALUE_DIFFERENCE_NV 0x95B6
06392 #define GL_BACK_PRIMARY_COLOR_NV 0x8C77
06393 #define GL_BACK_SECONDARY_COLOR_NV 0x8C78
06394 #define GL_TEXTURE_COORD_NV 0x8C79
06395 #define GL_CLIP_DISTANCE_NV 0x8C7A
06396 #define GL_VERTEX_ID_NV 0x8C7B
06397 #define GL_PRIMITIVE_ID_NV 0x8C7C
06398 #define GL_GENERIC_ATTRIB_NV 0x8C7D
06399 #define GL_TRANSFORM_FEEDBACK_ATTRIBS_NV 0x8C7E
06400 #define GL_TRANSFORM_FEEDBACK_BUFFER_MODE_NV 0x8C7F
06401 #define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_COMPONENTS_NV 0x8C80
06402 #define GL_ACTIVE_VARYINGS_NV 0x8C81
06403 #define GL_ACTIVE_VARYING_MAX_LENGTH_NV 0x8C82
06404 #define GL_TRANSFORM_FEEDBACK_VARYINGS_NV 0x8C83
06405 #define GL_TRANSFORM_FEEDBACK_BUFFER_START_NV 0x8C84
06406 #define GL_TRANSFORM_FEEDBACK_BUFFER_SIZE_NV 0x8C85
06407 #define GL_TRANSFORM_FEEDBACK_RECORD_NV 0x8C86
06408 #define GL_PRIMITIVES_GENERATED_NV 0x8C87
06409 #define GL_TRANSFORM_FEEDBACK_PRIMITIVES_WRITTEN_NV 0x8C88
06410 #define GL_RASTERIZER_DISCARD_NV 0x8C89
06411 #define GL_MAX_TRANSFORM_FEEDBACK_INTERLEAVED_COMPONENTS_NV 0x8C8A
06412 #define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_ATTRIBS_NV 0x8C8B
06413 #define GL_INTERLEAVED_ATTRIBS_NV 0x8C8C
06414 #define GL_SEPARATE_ATTRIBS_NV 0x8C8D
06415 #define GL_TRANSFORM_FEEDBACK_BUFFER_NV 0x8C8E
06416 #define GL_TRANSFORM_FEEDBACK_BUFFER_BINDING_NV 0x8C8F
06417 #define GL_LAYER_NV 0x8DAA
06418 #define GL_NEXT_BUFFER_NV -2
06419 #define GL_SKIP_COMPONENTS4_NV -3
06420 #define GL_SKIP_COMPONENTS3_NV -4
06421 #define GL_SKIP_COMPONENTS2_NV -5
06422 #define GL_SKIP_COMPONENTS1_NV -6
06423 #define GL_TRANSFORM_FEEDBACK_NV 0x8E22
06424 #define GL_TRANSFORM_FEEDBACK_BUFFER_PAUSED_NV 0x8E23
06425 #define GL_TRANSFORM_FEEDBACK_BUFFER_ACTIVE_NV 0x8E24
06426 #define GL_TRANSFORM_FEEDBACK_BINDING_NV 0x8E25
06427 #define GL_UNIFORM_BUFFER_UNIFIED_NV 0x936E
06428 #define GL_UNIFORM_BUFFER_ADDRESS_NV 0x936F
06429 #define GL_UNIFORM_BUFFER_LENGTH_NV 0x9370
06430 #define GL_SURFACE_STATE_NV 0x86EB
06431 #define GL_SURFACE_REGISTERED_NV 0x86FD
06432 #define GL_SURFACE_MAPPED_NV 0x8700
06433 #define GL_WRITE_DISCARD_NV 0x88BE
06434 #define GL_VERTEX_ARRAY_RANGE_NV 0x851D
06435 #define GL_VERTEX_ARRAY_RANGE_LENGTH_NV 0x851E
06436 #define GL_VERTEX_ARRAY_RANGE_VALID_NV 0x851F
06437 #define GL_MAX_VERTEX_ARRAY_RANGE_ELEMENT_NV 0x8520
06438 #define GL_VERTEX_ARRAY_RANGE_POINTER_NV 0x8521
```

```
06439 #define GL_VERTEX_ARRAY_RANGE_WITHOUT_FLUSH_NV 0x8533
06440 #define GL_VERTEX_ATTRIB_ARRAY_UNIFIED_NV 0x8F1E
06441 #define GL_ELEMENT_ARRAY_UNIFIED_NV 0x8F1F
06442 #define GL_VERTEX_ATTRIB_ARRAY_ADDRESS_NV 0x8F20
06443 #define GL_VERTEX_ARRAY_ADDRESS_NV 0x8F21
06444 #define GL_NORMAL_ARRAY_ADDRESS_NV 0x8F22
06445 #define GL_COLOR_ARRAY_ADDRESS_NV 0x8F23
06446 #define GL_INDEX_ARRAY_ADDRESS_NV 0x8F24
06447 #define GL_TEXTURE_COORD_ARRAY_ADDRESS_NV 0x8F25
06448 #define GL_EDGE_FLAG_ARRAY_ADDRESS_NV 0x8F26
06449 #define GL_SECONDARY_COLOR_ARRAY_ADDRESS_NV 0x8F27
06450 #define GL_FOG_COORD_ARRAY_ADDRESS_NV 0x8F28
06451 #define GL_ELEMENT_ARRAY_ADDRESS_NV 0x8F29
06452 #define GL_VERTEX_ATTRIB_ARRAY_LENGTH_NV 0x8F2A
06453 #define GL_VERTEX_ARRAY_LENGTH_NV 0x8F2B
06454 #define GL_NORMAL_ARRAY_LENGTH_NV 0x8F2C
06455 #define GL_COLOR_ARRAY_LENGTH_NV 0x8F2D
06456 #define GL_INDEX_ARRAY_LENGTH_NV 0x8F2E
06457 #define GL_TEXTURE_COORD_ARRAY_LENGTH_NV 0x8F2F
06458 #define GL_EDGE_FLAG_ARRAY_LENGTH_NV 0x8F30
06459 #define GL_SECONDARY_COLOR_ARRAY_LENGTH_NV 0x8F31
06460 #define GL_FOG_COORD_ARRAY_LENGTH_NV 0x8F32
06461 #define GL_ELEMENT_ARRAY_LENGTH_NV 0x8F33
06462 #define GL_DRAW_INDIRECT_UNIFIED_NV 0x8F40
06463 #define GL_DRAW_INDIRECT_ADDRESS_NV 0x8F41
06464 #define GL_DRAW_INDIRECT_LENGTH_NV 0x8F42
06465 #define GL_VERTEX_PROGRAM_NV 0x8620
06466 #define GL_VERTEX_STATE_PROGRAM_NV 0x8621
06467 #define GL_ATTRIB_ARRAY_SIZE_NV 0x8623
06468 #define GL_ATTRIB_ARRAY_STRIDE_NV 0x8624
06469 #define GL_ATTRIB_ARRAY_TYPE_NV 0x8625
06470 #define GL_CURRENT_ATTRIB_NV 0x8626
06471 #define GL_PROGRAM_LENGTH_NV 0x8627
06472 #define GL_PROGRAM_STRING_NV 0x8628
06473 #define GL_MODELVIEW_PROJECTION_NV 0x8629
06474 #define GL_IDENTITY_NV 0x862A
06475 #define GL_INVERSE_NV 0x862B
06476 #define GL_TRANSPOSE_NV 0x862C
06477 #define GL_INVERSE_TRANSPOSE_NV 0x862D
06478 #define GL_MAX_TRACK_MATRIX_STACK_DEPTH_NV 0x862E
06479 #define GL_MAX_TRACK_MATRICES_NV 0x862F
06480 #define GL_MATRIX0_NV 0x8630
06481 #define GL_MATRIX1_NV 0x8631
06482 #define GL_MATRIX2_NV 0x8632
06483 #define GL_MATRIX3_NV 0x8633
06484 #define GL_MATRIX4_NV 0x8634
06485 #define GL_MATRIX5_NV 0x8635
06486 #define GL_MATRIX6_NV 0x8636
06487 #define GL_MATRIX7_NV 0x8637
06488 #define GL_CURRENT_MATRIX_STACK_DEPTH_NV 0x8640
06489 #define GL_CURRENT_MATRIX_NV 0x8641
06490 #define GL_VERTEX_PROGRAM_POINT_SIZE_NV 0x8642
06491 #define GL_VERTEX_PROGRAM_TWO_SIDE_NV 0x8643
06492 #define GL_PROGRAM_PARAMETER_NV 0x8644
06493 #define GL_ATTRIB_ARRAY_POINTER_NV 0x8645
06494 #define GL_PROGRAM_TARGET_NV 0x8646
06495 #define GL_PROGRAM_RESIDENT_NV 0x8647
06496 #define GL_TRACK_MATRIX_NV 0x8648
06497 #define GL_TRACK_MATRIX_TRANSFORM_NV 0x8649
06498 #define GL_VERTEX_PROGRAM_BINDING_NV 0x864A
06499 #define GL_PROGRAM_ERROR_POSITION_NV 0x864B
06500 #define GL_VERTEX_ATTRIB_ARRAY0_NV 0x8650
06501 #define GL_VERTEX_ATTRIB_ARRAY1_NV 0x8651
06502 #define GL_VERTEX_ATTRIB_ARRAY2_NV 0x8652
06503 #define GL_VERTEX_ATTRIB_ARRAY3_NV 0x8653
06504 #define GL_VERTEX_ATTRIB_ARRAY4_NV 0x8654
06505 #define GL_VERTEX_ATTRIB_ARRAY5_NV 0x8655
06506 #define GL_VERTEX_ATTRIB_ARRAY6_NV 0x8656
06507 #define GL_VERTEX_ATTRIB_ARRAY7_NV 0x8657
06508 #define GL_VERTEX_ATTRIB_ARRAY8_NV 0x8658
06509 #define GL_VERTEX_ATTRIB_ARRAY9_NV 0x8659
06510 #define GL_VERTEX_ATTRIB_ARRAY10_NV 0x865A
06511 #define GL_VERTEX_ATTRIB_ARRAY11_NV 0x865B
06512 #define GL_VERTEX_ATTRIB_ARRAY12_NV 0x865C
06513 #define GL_VERTEX_ATTRIB_ARRAY13_NV 0x865D
06514 #define GL_VERTEX_ATTRIB_ARRAY14_NV 0x865E
06515 #define GL_VERTEX_ATTRIB_ARRAY15_NV 0x865F
06516 #define GL_MAP1_VERTEX_ATTRIB0_4_NV 0x8660
06517 #define GL_MAP1_VERTEX_ATTRIB1_4_NV 0x8661
06518 #define GL_MAP1_VERTEX_ATTRIB2_4_NV 0x8662
06519 #define GL_MAP1_VERTEX_ATTRIB3_4_NV 0x8663
06520 #define GL_MAP1_VERTEX_ATTRIB4_4_NV 0x8664
06521 #define GL_MAP1_VERTEX_ATTRIB5_4_NV 0x8665
06522 #define GL_MAP1_VERTEX_ATTRIB6_4_NV 0x8666
06523 #define GL_MAP1_VERTEX_ATTRIB7_4_NV 0x8667
06524 #define GL_MAP1_VERTEX_ATTRIB8_4_NV 0x8668
06525 #define GL_MAP1_VERTEX_ATTRIB9_4_NV 0x8669
```

```
06526 #define GL_MAP1_VERTEX_ATTRIB10_4_NV 0x866A
06527 #define GL_MAP1_VERTEX_ATTRIB11_4_NV 0x866B
06528 #define GL_MAP1_VERTEX_ATTRIB12_4_NV 0x866C
06529 #define GL_MAP1_VERTEX_ATTRIB13_4_NV 0x866D
06530 #define GL_MAP1_VERTEX_ATTRIB14_4_NV 0x866E
06531 #define GL_MAP1_VERTEX_ATTRIB15_4_NV 0x866F
06532 #define GL_MAP2_VERTEX_ATTRIB0_4_NV 0x8670
06533 #define GL_MAP2_VERTEX_ATTRIB1_4_NV 0x8671
06534 #define GL_MAP2_VERTEX_ATTRIB2_4_NV 0x8672
06535 #define GL_MAP2_VERTEX_ATTRIB3_4_NV 0x8673
06536 #define GL_MAP2_VERTEX_ATTRIB4_4_NV 0x8674
06537 #define GL_MAP2_VERTEX_ATTRIB5_4_NV 0x8675
06538 #define GL_MAP2_VERTEX_ATTRIB6_4_NV 0x8676
06539 #define GL_MAP2_VERTEX_ATTRIB7_4_NV 0x8677
06540 #define GL_MAP2_VERTEX_ATTRIB8_4_NV 0x8678
06541 #define GL_MAP2_VERTEX_ATTRIB9_4_NV 0x8679
06542 #define GL_MAP2_VERTEX_ATTRIB10_4_NV 0x867A
06543 #define GL_MAP2_VERTEX_ATTRIB11_4_NV 0x867B
06544 #define GL_MAP2_VERTEX_ATTRIB12_4_NV 0x867C
06545 #define GL_MAP2_VERTEX_ATTRIB13_4_NV 0x867D
06546 #define GL_MAP2_VERTEX_ATTRIB14_4_NV 0x867E
06547 #define GL_MAP2_VERTEX_ATTRIB15_4_NV 0x867F
06548 #define GL_VERTEX_ATTRIB_ARRAY_INTEGER_NV 0x88FD
06549 #define GL_VIDEO_BUFFER_NV 0x9020
06550 #define GL_VIDEO_BUFFER_BINDING_NV 0x9021
06551 #define GL_FIELD_UPPER_NV 0x9022
06552 #define GL_FIELD_LOWER_NV 0x9023
06553 #define GL_NUM_VIDEO_CAPTURE_STREAMS_NV 0x9024
06554 #define GL_NEXT_VIDEO_CAPTURE_BUFFER_STATUS_NV 0x9025
06555 #define GL_VIDEO_CAPTURE_TO_422_SUPPORTED_NV 0x9026
06556 #define GL_LAST_VIDEO_CAPTURE_STATUS_NV 0x9027
06557 #define GL_VIDEO_BUFFER_PITCH_NV 0x9028
06558 #define GL_VIDEO_COLOR_CONVERSION_MATRIX_NV 0x9029
06559 #define GL_VIDEO_COLOR_CONVERSION_MAX_NV 0x902A
06560 #define GL_VIDEO_COLOR_CONVERSION_MIN_NV 0x902B
06561 #define GL_VIDEO_COLOR_CONVERSION_OFFSET_NV 0x902C
06562 #define GL_VIDEO_BUFFER_INTERNAL_FORMAT_NV 0x902D
06563 #define GL_PARTIAL_SUCCESS_NV 0x902E
06564 #define GL_SUCCESS_NV 0x902F
06565 #define GL_FAILURE_NV 0x9030
06566 #define GL_YCBYCR8_422_NV 0x9031
06567 #define GL_YCBAYCR8A_4224_NV 0x9032
06568 #define GL_Z6Y10Z6CB10Z6Y10Z6CR10_422_NV 0x9033
06569 #define GL_Z6Y10Z6CB10Z6A10Z6Y10Z6CR10Z6A10_4224_NV 0x9034
06570 #define GL_Z4Y12Z4CB12Z4Y12Z4CR12_422_NV 0x9035
06571 #define GL_Z4Y12Z4CB12Z4A12Z4Y12Z4CR12Z4A12_4224_NV 0x9036
06572 #define GL_Z4Y12Z4CB12Z4CR12_444_NV 0x9037
06573 #define GL_VIDEO_CAPTURE_FRAME_WIDTH_NV 0x9038
06574 #define GL_VIDEO_CAPTURE_FRAME_HEIGHT_NV 0x9039
06575 #define GL_VIDEO_CAPTURE_FIELD_UPPER_HEIGHT_NV 0x903A
06576 #define GL_VIDEO_CAPTURE_FIELD_LOWER_HEIGHT_NV 0x903B
06577 #define GL_VIDEO_CAPTURE_SURFACE_ORIGIN_NV 0x903C
06578 #define GL_VIEWPORT_SWIZZLE_POSITIVE_X_NV 0x9350
06579 #define GL_VIEWPORT_SWIZZLE_NEGATIVE_X_NV 0x9351
06580 #define GL_VIEWPORT_SWIZZLE_POSITIVE_Y_NV 0x9352
06581 #define GL_VIEWPORT_SWIZZLE_NEGATIVE_Y_NV 0x9353
06582 #define GL_VIEWPORT_SWIZZLE_POSITIVE_Z_NV 0x9354
06583 #define GL_VIEWPORT_SWIZZLE_NEGATIVE_Z_NV 0x9355
06584 #define GL_VIEWPORT_SWIZZLE_POSITIVE_W_NV 0x9356
06585 #define GL_VIEWPORT_SWIZZLE_NEGATIVE_W_NV 0x9357
06586 #define GL_VIEWPORT_SWIZZLE_X_NV 0x9358
06587 #define GL_VIEWPORT_SWIZZLE_Y_NV 0x9359
06588 #define GL_VIEWPORT_SWIZZLE_Z_NV 0x935A
06589 #define GL_VIEWPORT_SWIZZLE_W_NV 0x935B
06590 #define GL_PALETTE4_RGB8_OES 0x8B90
06591 #define GL_PALETTE4_RGBA8_OES 0x8B91
06592 #define GL_PALETTE4_R5_G6_B5_OES 0x8B92
06593 #define GL_PALETTE4_RGBA4_OES 0x8B93
06594 #define GL_PALETTE4_RGB5_A1_OES 0x8B94
06595 #define GL_PALETTE8_RGB8_OES 0x8B95
06596 #define GL_PALETTE8_RGBA8_OES 0x8B96
06597 #define GL_PALETTE8_R5_G6_B5_OES 0x8B97
06598 #define GL_PALETTE8_RGBA4_OES 0x8B98
06599 #define GL_PALETTE8_RGB5_A1_OES 0x8B99
06600 #define GL_FIXED_OES 0x140C
06601 #define GL_IMPLEMENTATION_COLOR_READ_TYPE_OES 0x8B9A
06602 #define GL_IMPLEMENTATION_COLOR_READ_FORMAT_OES 0x8B9B
06603 #define GL_INTERLACE_OML 0x8980
06604 #define GL_INTERLACE_READ_OML 0x8981
06605 #define GL_PACK_RESAMPLE_OML 0x8984
06606 #define GL_UNPACK_RESAMPLE_OML 0x8985
06607 #define GL_RESAMPLE_REPLICATE_OML 0x8986
06608 #define GL_RESAMPLE_ZERO_FILL_OML 0x8987
06609 #define GL_RESAMPLE_AVERAGE_OML 0x8988
06610 #define GL_RESAMPLE_DECIMATE_OML 0x8989
06611 #define GL_FORMAT_SUBSAMPLE_24_24_OML 0x8982
06612 #define GL_FORMAT_SUBSAMPLE_244_244_OML 0x8983
```

```
06613 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_NUM_VIEWS_OVR 0x9630
06614 #define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_BASE_VIEW_INDEX_OVR 0x9632
06615 #define GL_MAX_VIEWS_OVR 0x9631
06616 #define GL_FRAMEBUFFER_INCOMPLETE_VIEW_TARGETS_OVR 0x9633
06617 #define GL_PREFER_DOUBLEBUFFER_HINT_PGI 0x1A1F8
06618 #define GL_CONSERVE_MEMORY_HINT_PGI 0x1A1FD
06619 #define GL_RECLAIM_MEMORY_HINT_PGI 0x1A1FE
06620 #define GL_NATIVE_GRAPHICS_HANDLE_PGI 0x1A202
06621 #define GL_NATIVE_GRAPHICS_BEGIN_HINT_PGI 0x1A203
06622 #define GL_NATIVE_GRAPHICS_END_HINT_PGI 0x1A204
06623 #define GL_ALWAYS_FAST_HINT_PGI 0x1A20C
06624 #define GL_ALWAYS_SOFT_HINT_PGI 0x1A20D
06625 #define GL_ALLOW_DRAW_OBJ_HINT_PGI 0x1A20E
06626 #define GL_ALLOW_DRAW_WIN_HINT_PGI 0x1A20F
06627 #define GL_ALLOW_DRAW_FRG_HINT_PGI 0x1A210
06628 #define GL_ALLOW_DRAW_MEM_HINT_PGI 0x1A211
06629 #define GL_STRICT_DEPTHFUNC_HINT_PGI 0x1A216
06630 #define GL_STRICT_LIGHTING_HINT_PGI 0x1A217
06631 #define GL_STRICT_SCISSOR_HINT_PGI 0x1A218
06632 #define GL_FULL_STIPPLE_HINT_PGI 0x1A219
06633 #define GL_CLIP_NEAR_HINT_PGI 0x1A220
06634 #define GL_CLIP_FAR_HINT_PGI 0x1A221
06635 #define GL_WIDE_LINE_HINT_PGI 0x1A222
06636 #define GL_BACK_NORMALS_HINT_PGI 0x1A223
06637 #define GL_VERTEX_DATA_HINT_PGI 0x1A22A
06638 #define GL_VERTEX_CONSISTENT_HINT_PGI 0x1A22B
06639 #define GL_MATERIAL_SIDE_HINT_PGI 0x1A22C
06640 #define GL_MAX_VERTEX_HINT_PGI 0x1A22D
06641 #define GL_COLOR3_BIT_PGI 0x00010000
06642 #define GL_COLOR4_BIT_PGI 0x00020000
06643 #define GL_EDGEFLAG_BIT_PGI 0x00040000
06644 #define GL_INDEX_BIT_PGI 0x00080000
06645 #define GL_MAT_AMBIENT_BIT_PGI 0x00100000
06646 #define GL_MAT_AMBIENT_AND_DIFFUSE_BIT_PGI 0x00200000
06647 #define GL_MAT_DIFFUSE_BIT_PGI 0x00400000
06648 #define GL_MAT_EMISSION_BIT_PGI 0x00800000
06649 #define GL_MAT_COLOR_INDEXES_BIT_PGI 0x01000000
06650 #define GL_MAT_SHININESS_BIT_PGI 0x02000000
06651 #define GL_MAT_SPECULAR_BIT_PGI 0x04000000
06652 #define GL_NORMAL_BIT_PGI 0x08000000
06653 #define GL_TEXCOORD1_BIT_PGI 0x10000000
06654 #define GL_TEXCOORD2_BIT_PGI 0x20000000
06655 #define GL_TEXCOORD3_BIT_PGI 0x40000000
06656 #define GL_TEXCOORD4_BIT_PGI 0x80000000
06657 #define GL_VERTEX23_BIT_PGI 0x00000004
06658 #define GL_VERTEX4_BIT_PGI 0x00000008
06659 #define GL_SCREEN_COORDINATES_REND 0x8490
06660 #define GL_INVERTED_SCREEN_W_REND 0x8491
06661 #define GL_RGB_S3TC 0x83A0
06662 #define GL_RGBA_S3TC 0x83A1
06663 #define GL_RGBA_S3TC 0x83A2
06664 #define GL_RGBA4_S3TC 0x83A3
06665 #define GL_RGBA_DXT5_S3TC 0x83A4
06666 #define GL_RGBA4_DXT5_S3TC 0x83A5
06667 #define GL_DETAIL_TEXTURE_2D_SGIS 0x8095
06668 #define GL_DETAIL_TEXTURE_2D_BINDING_SGIS 0x8096
06669 #define GL_LINEAR_DETAIL_SGIS 0x8097
06670 #define GL_LINEAR_DETAIL_ALPHA_SGIS 0x8098
06671 #define GL_LINEAR_DETAIL_COLOR_SGIS 0x8099
06672 #define GL_DETAIL_TEXTURE_LEVEL_SGIS 0x809A
06673 #define GL_DETAIL_TEXTURE_MODE_SGIS 0x809B
06674 #define GL_DETAIL_TEXTURE_FUNC_POINTS_SGIS 0x809C
06675 #define GL_FOG_FUNC_SGIS 0x812A
06676 #define GL_FOG_FUNC_POINTS_SGIS 0x812B
06677 #define GL_MAX_FOG_FUNC_POINTS_SGIS 0x812C
06678 #define GL_GENERATE_MIPMAP_SGIS 0x8191
06679 #define GL_GENERATE_MIPMAP_HINT_SGIS 0x8192
06680 #define GL_MULTISAMPLE_SGIS 0x809D
06681 #define GL_SAMPLE_ALPHA_TO_MASK_SGIS 0x809E
06682 #define GL_SAMPLE_ALPHA_TO_ONE_SGIS 0x809F
06683 #define GL_SAMPLE_MASK_SGIS 0x80A0
06684 #define GL_1PASS_SGIS 0x80A1
06685 #define GL_2PASS_0_SGIS 0x80A2
06686 #define GL_2PASS_1_SGIS 0x80A3
06687 #define GL_4PASS_0_SGIS 0x80A4
06688 #define GL_4PASS_1_SGIS 0x80A5
06689 #define GL_4PASS_2_SGIS 0x80A6
06690 #define GL_4PASS_3_SGIS 0x80A7
06691 #define GL_SAMPLE_BUFFERS_SGIS 0x80A8
06692 #define GL_SAMPLES_SGIS 0x80A9
06693 #define GL_SAMPLE_MASK_VALUE_SGIS 0x80AA
06694 #define GL_SAMPLE_MASK_INVERT_SGIS 0x80AB
06695 #define GL_SAMPLE_PATTERN_SGIS 0x80AC
06696 #define GL_PIXEL_TEXTURE_SGIS 0x8353
06697 #define GL_PIXEL_FRAGMENT_RGB_SOURCE_SGIS 0x8354
06698 #define GL_PIXEL_FRAGMENT_ALPHA_SOURCE_SGIS 0x8355
06699 #define GL_PIXEL_GROUP_COLOR_SGIS 0x8356
```



```
06700 #define GL_EYE_DISTANCE_TO_POINT_SGIS 0x81F0
06701 #define GL_OBJECT_DISTANCE_TO_POINT_SGIS 0x81F1
06702 #define GL_EYE_DISTANCE_TO_LINE_SGIS 0x81F2
06703 #define GL_OBJECT_DISTANCE_TO_LINE_SGIS 0x81F3
06704 #define GL_EYE_POINT_SGIS 0x81F4
06705 #define GL_OBJECT_POINT_SGIS 0x81F5
06706 #define GL_EYE_LINE_SGIS 0x81F6
06707 #define GL_OBJECT_LINE_SGIS 0x81F7
06708 #define GL_POINT_SIZE_MIN_SGIS 0x8126
06709 #define GL_POINT_SIZE_MAX_SGIS 0x8127
06710 #define GL_POINT_FADE_THRESHOLD_SIZE_SGIS 0x8128
06711 #define GL_DISTANCE_ATTENUATION_SGIS 0x8129
06712 #define GL_LINEAR_SHARPEN_SGIS 0x80AD
06713 #define GL_LINEAR_SHARPEN_ALPHA_SGIS 0x80AE
06714 #define GL_LINEAR_SHARPEN_COLOR_SGIS 0x80AF
06715 #define GL_SHARPEN_TEXTURE_FUNC_POINTS_SGIS 0x80B0
06716 #define GL_PACK_SKIP_VOLUMES_SGIS 0x8130
06717 #define GL_PACK_IMAGE_DEPTH_SGIS 0x8131
06718 #define GL_UNPACK_SKIP_VOLUMES_SGIS 0x8132
06719 #define GL_UNPACK_IMAGE_DEPTH_SGIS 0x8133
06720 #define GL_TEXTURE_4D_SGIS 0x8134
06721 #define GL_PROXY_TEXTURE_4D_SGIS 0x8135
06722 #define GL_TEXTURE_4DSIZE_SGIS 0x8136
06723 #define GL_TEXTURE_WRAP_Q_SGIS 0x8137
06724 #define GL_MAX_4D_TEXTURE_SIZE_SGIS 0x8138
06725 #define GL_TEXTURE_4D_BINDING_SGIS 0x814F
06726 #define GL_CLAMP_TO_BORDER_SGIS 0x812D
06727 #define GL_TEXTURE_COLOR_WRITEMASK_SGIS 0x81EF
06728 #define GL_CLAMP_TO_EDGE_SGIS 0x812F
06729 #define GL_FILTER4_SGIS 0x8146
06730 #define GL_TEXTURE_FILTER4_SIZE_SGIS 0x8147
06731 #define GL_TEXTURE_MIN_LOD_SGIS 0x813A
06732 #define GL_TEXTURE_MAX_LOD_SGIS 0x813B
06733 #define GL_TEXTURE_BASE_LEVEL_SGIS 0x813C
06734 #define GL_TEXTURE_MAX_LEVEL_SGIS 0x813D
06735 #define GL_DUAL_ALPHA4_SGIS 0x8110
06736 #define GL_DUAL_ALPHA8_SGIS 0x8111
06737 #define GL_DUAL_ALPHA12_SGIS 0x8112
06738 #define GL_DUAL_ALPHA16_SGIS 0x8113
06739 #define GL_DUAL_LUMINANCE4_SGIS 0x8114
06740 #define GL_DUAL_LUMINANCE8_SGIS 0x8115
06741 #define GL_DUAL_LUMINANCE12_SGIS 0x8116
06742 #define GL_DUAL_LUMINANCE16_SGIS 0x8117
06743 #define GL_DUAL_INTENSITY4_SGIS 0x8118
06744 #define GL_DUAL_INTENSITY8_SGIS 0x8119
06745 #define GL_DUAL_INTENSITY12_SGIS 0x811A
06746 #define GL_DUAL_INTENSITY16_SGIS 0x811B
06747 #define GL_DUAL_LUMINANCE_ALPHA4_SGIS 0x811C
06748 #define GL_DUAL_LUMINANCE_ALPHA8_SGIS 0x811D
06749 #define GL_QUAD_ALPHA4_SGIS 0x811E
06750 #define GL_QUAD_ALPHA8_SGIS 0x811F
06751 #define GL_QUAD_LUMINANCE4_SGIS 0x8120
06752 #define GL_QUAD_LUMINANCE8_SGIS 0x8121
06753 #define GL_QUAD_INTENSITY4_SGIS 0x8122
06754 #define GL_QUAD_INTENSITY8_SGIS 0x8123
06755 #define GL_DUAL_TEXTURE_SELECT_SGIS 0x8124
06756 #define GL_QUAD_TEXTURE_SELECT_SGIS 0x8125
06757 #define GL_ASYNC_MARKER_SGIX 0x8329
06758 #define GL_ASYNC_HISTOGRAM_SGIX 0x832C
06759 #define GL_MAX_ASYNC_HISTOGRAM_SGIX 0x832D
06760 #define GL_ASYNC_TEX_IMAGE_SGIX 0x835C
06761 #define GL_ASYNC_DRAW_PIXELS_SGIX 0x835D
06762 #define GL_ASYNC_READ_PIXELS_SGIX 0x835E
06763 #define GL_MAX_ASYNC_TEX_IMAGE_SGIX 0x835F
06764 #define GL_MAX_ASYNC_DRAW_PIXELS_SGIX 0x8360
06765 #define GL_MAX_ASYNC_READ_PIXELS_SGIX 0x8361
06766 #define GL_ALPHA_MIN_SGIX 0x8320
06767 #define GL_ALPHA_MAX_SGIX 0x8321
06768 #define GL_CALLIGRAPHIC_FRAGMENT_SGIX 0x8183
06769 #define GL_LINEAR_CLIPMAP_LINEAR_SGIX 0x8170
06770 #define GL_TEXTURE_CLIPMAP_CENTER_SGIX 0x8171
06771 #define GL_TEXTURE_CLIPMAP_FRAME_SGIX 0x8172
06772 #define GL_TEXTURE_CLIPMAP_OFFSET_SGIX 0x8173
06773 #define GL_TEXTURE_CLIPMAP_VIRTUAL_DEPTH_SGIX 0x8174
06774 #define GL_TEXTURE_CLIPMAP_LOD_OFFSET_SGIX 0x8175
06775 #define GL_TEXTURE_CLIPMAP_DEPTH_SGIX 0x8176
06776 #define GL_MAX_CLIPMAP_DEPTH_SGIX 0x8177
06777 #define GL_MAX_CLIPMAP_VIRTUAL_DEPTH_SGIX 0x8178
06778 #define GL_NEAREST_CLIPMAP_NEAREST_SGIX 0x844D
06779 #define GL_NEAREST_CLIPMAP_LINEAR_SGIX 0x844E
06780 #define GL_LINEAR_CLIPMAP_NEAREST_SGIX 0x844F
06781 #define GL_CONVOLUTION_HINT_SGIX 0x8316
06782 #define GL_DEPTH_COMPONENT16_SGIX 0x81A5
06783 #define GL_DEPTH_COMPONENT24_SGIX 0x81A6
06784 #define GL_DEPTH_COMPONENT32_SGIX 0x81A7
06785 #define GL_FOG_OFFSET_SGIX 0x8198
06786 #define GL_FOG_OFFSET_VALUE_SGIX 0x8199
```

```
06787 #define GL_FRAGMENT_LIGHTING_SGIX 0x8400
06788 #define GL_FRAGMENT_COLOR_MATERIAL_SGIX 0x8401
06789 #define GL_FRAGMENT_COLOR_MATERIAL_FACE_SGIX 0x8402
06790 #define GL_FRAGMENT_COLOR_MATERIAL_PARAMETER_SGIX 0x8403
06791 #define GL_MAX_FRAGMENT_LIGHTS_SGIX 0x8404
06792 #define GL_MAX_ACTIVE_LIGHTS_SGIX 0x8405
06793 #define GL_CURRENT_RASTER_NORMAL_SGIX 0x8406
06794 #define GL_LIGHT_ENV_MODE_SGIX 0x8407
06795 #define GL_FRAGMENT_LIGHT_MODEL_LOCAL_VIEWER_SGIX 0x8408
06796 #define GL_FRAGMENT_LIGHT_MODEL_TWO_SIDE_SGIX 0x8409
06797 #define GL_FRAGMENT_LIGHT_MODEL_AMBIENT_SGIX 0x840A
06798 #define GL_FRAGMENT_LIGHT_MODEL_NORMAL_INTERPOLATION_SGIX 0x840B
06799 #define GL_FRAGMENT_LIGHT0_SGIX 0x840C
06800 #define GL_FRAGMENT_LIGHT1_SGIX 0x840D
06801 #define GL_FRAGMENT_LIGHT2_SGIX 0x840E
06802 #define GL_FRAGMENT_LIGHT3_SGIX 0x840F
06803 #define GL_FRAGMENT_LIGHT4_SGIX 0x8410
06804 #define GL_FRAGMENT_LIGHT5_SGIX 0x8411
06805 #define GL_FRAGMENT_LIGHT6_SGIX 0x8412
06806 #define GL_FRAGMENT_LIGHT7_SGIX 0x8413
06807 #define GL_FRAMEZOOM_SGIX 0x818B
06808 #define GL_FRAMEZOOM_FACTOR_SGIX 0x818C
06809 #define GL_MAX_FRAMEZOOM_FACTOR_SGIX 0x818D
06810 #define GL_INSTRUMENT_BUFFER_POINTER_SGIX 0x8180
06811 #define GL_INSTRUMENT_MEASUREMENTS_SGIX 0x8181
06812 #define GL_INTERLACE_SGIX 0x8094
06813 #define GL_IR_INSTRUMENT1_SGIX 0x817F
06814 #define GL_LIST_PRIORITY_SGIX 0x8182
06815 #define GL_PIXEL_TEX_GEN_SGIX 0x8139
06816 #define GL_PIXEL_TEX_GEN_MODE_SGIX 0x832B
06817 #define GL_PIXEL_TILE_BEST_ALIGNMENT_SGIX 0x813E
06818 #define GL_PIXEL_TILE_CACHE_INCREMENT_SGIX 0x813F
06819 #define GL_PIXEL_TILE_WIDTH_SGIX 0x8140
06820 #define GL_PIXEL_TILE_HEIGHT_SGIX 0x8141
06821 #define GL_PIXEL_TILE_GRID_WIDTH_SGIX 0x8142
06822 #define GL_PIXEL_TILE_GRID_HEIGHT_SGIX 0x8143
06823 #define GL_PIXEL_TILE_GRID_DEPTH_SGIX 0x8144
06824 #define GL_PIXEL_TILE_CACHE_SIZE_SGIX 0x8145
06825 #define GL_TEXTURE_DEFORMATION_BIT_SGIX 0x00000001
06826 #define GL_GEOMETRY_DEFORMATION_BIT_SGIX 0x00000002
06827 #define GL_GEOMETRY_DEFORMATION_SGIX 0x8194
06828 #define GL_TEXTURE_DEFORMATION_SGIX 0x8195
06829 #define GL_DEFORMATIONS_MASK_SGIX 0x8196
06830 #define GL_MAX_DEFORMATION_ORDER_SGIX 0x8197
06831 #define GL_REFERENCE_PLANE_SGIX 0x817D
06832 #define GL_REFERENCE_PLANE_EQUATION_SGIX 0x817E
06833 #define GL_PACK_RESAMPLE_SGIX 0x842E
06834 #define GL_UNPACK_RESAMPLE_SGIX 0x842F
06835 #define GL_RESAMPLE_REPLICATE_SGIX 0x8433
06836 #define GL_RESAMPLE_ZERO_FILL_SGIX 0x8434
06837 #define GL_RESAMPLE_DECIMATE_SGIX 0x8430
06838 #define GL_SCALEBIAS_HINT_SGIX 0x8322
06839 #define GL_TEXTURE_COMPARE_SGIX 0x819A
06840 #define GL_TEXTURE_COMPARE_OPERATOR_SGIX 0x819B
06841 #define GL_TEXTURE_LEQUAL_R_SGIX 0x819C
06842 #define GL_TEXTURE_GEQUAL_R_SGIX 0x819D
06843 #define GL_SHADOW_AMBIENT_SGIX 0x80BF
06844 #define GL_SPRITE_SGIX 0x8148
06845 #define GL_SPRITE_MODE_SGIX 0x8149
06846 #define GL_SPRITE_AXIS_SGIX 0x814A
06847 #define GL_SPRITE_TRANSLATION_SGIX 0x814B
06848 #define GL_SPRITE_AXIAL_SGIX 0x814C
06849 #define GL_SPRITE_OBJECT_ALIGNED_SGIX 0x814D
06850 #define GL_SPRITE_EYE_ALIGNED_SGIX 0x814E
06851 #define GL_PACK_SUBSAMPLE_RATE_SGIX 0x85A0
06852 #define GL_UNPACK_SUBSAMPLE_RATE_SGIX 0x85A1
06853 #define GL_PIXEL_SUBSAMPLE_4444_SGIX 0x85A2
06854 #define GL_PIXEL_SUBSAMPLE_2424_SGIX 0x85A3
06855 #define GL_PIXEL_SUBSAMPLE_4242_SGIX 0x85A4
06856 #define GL_TEXTURE_ENV_BIAS_SGIX 0x80BE
06857 #define GL_TEXTURE_MAX_CLAMP_S_SGIX 0x8369
06858 #define GL_TEXTURE_MAX_CLAMP_T_SGIX 0x836A
06859 #define GL_TEXTURE_MAX_CLAMP_R_SGIX 0x836B
06860 #define GL_TEXTURE_LOD_BIAS_S_SGIX 0x818E
06861 #define GL_TEXTURE_LOD_BIAS_T_SGIX 0x818F
06862 #define GL_TEXTURE_LOD_BIAS_R_SGIX 0x8190
06863 #define GL_TEXTURE_MULTI_BUFFER_HINT_SGIX 0x812E
06864 #define GL_POST_TEXTURE_FILTER_BIAS_SGIX 0x8179
06865 #define GL_POST_TEXTURE_FILTER_SCALE_SGIX 0x817A
06866 #define GL_POST_TEXTURE_FILTER_BIAS_RANGE_SGIX 0x817B
06867 #define GL_POST_TEXTURE_FILTER_SCALE_RANGE_SGIX 0x817C
06868 #define GL_VERTEX_PRECLIP_SGIX 0x83EE
06869 #define GL_VERTEX_PRECLIP_HINT_SGIX 0x83EF
06870 #define GL_YCRCB_422_SGIX 0x81BB
06871 #define GL_YCRCB_444_SGIX 0x81BC
06872 #define GL_YCRCB_SGIX 0x8318
06873 #define GL_YCRCBA_SGIX 0x8319
```

```

06874 #define GL_COLOR_MATRIX_SGI 0x80B1
06875 #define GL_COLOR_MATRIX_STACK_DEPTH_SGI 0x80B2
06876 #define GL_MAX_COLOR_MATRIX_STACK_DEPTH_SGI 0x80B3
06877 #define GL_POST_COLOR_MATRIX_RED_SCALE_SGI 0x80B4
06878 #define GL_POST_COLOR_MATRIX_GREEN_SCALE_SGI 0x80B5
06879 #define GL_POST_COLOR_MATRIX_BLUE_SCALE_SGI 0x80B6
06880 #define GL_POST_COLOR_MATRIX_ALPHA_SCALE_SGI 0x80B7
06881 #define GL_POST_COLOR_MATRIX_RED_BIAS_SGI 0x80B8
06882 #define GL_POST_COLOR_MATRIX_GREEN_BIAS_SGI 0x80B9
06883 #define GL_POST_COLOR_MATRIX_BLUE_BIAS_SGI 0x80BA
06884 #define GL_POST_COLOR_MATRIX_ALPHA_BIAS_SGI 0x80BB
06885 #define GL_COLOR_TABLE_SGI 0x80D0
06886 #define GL_POST_CONVOLUTION_COLOR_TABLE_SGI 0x80D1
06887 #define GL_POST_COLOR_MATRIX_COLOR_TABLE_SGI 0x80D2
06888 #define GL_PROXY_COLOR_TABLE_SGI 0x80D3
06889 #define GL_PROXY_POST_CONVOLUTION_COLOR_TABLE_SGI 0x80D4
06890 #define GL_PROXY_POST_COLOR_MATRIX_COLOR_TABLE_SGI 0x80D5
06891 #define GL_COLOR_TABLE_SCALE_SGI 0x80D6
06892 #define GL_COLOR_TABLE_BIAS_SGI 0x80D7
06893 #define GL_COLOR_TABLE_FORMAT_SGI 0x80D8
06894 #define GL_COLOR_TABLE_WIDTH_SGI 0x80D9
06895 #define GL_COLOR_TABLE_RED_SIZE_SGI 0x80DA
06896 #define GL_COLOR_TABLE_GREEN_SIZE_SGI 0x80DB
06897 #define GL_COLOR_TABLE_BLUE_SIZE_SGI 0x80DC
06898 #define GL_COLOR_TABLE_ALPHA_SIZE_SGI 0x80DD
06899 #define GL_COLOR_TABLE_LUMINANCE_SIZE_SGI 0x80DE
06900 #define GL_COLOR_TABLE_INTENSITY_SIZE_SGI 0x80DF
06901 #define GL_TEXTURE_COLOR_TABLE_SGI 0x80BC
06902 #define GL_PROXY_TEXTURE_COLOR_TABLE_SGI 0x80BD
06903 #define GL_UNPACK_CONSTANT_DATA_SUNX 0x81D5
06904 #define GL_TEXTURE_CONSTANT_DATA_SUNX 0x81D6
06905 #define GL_WRAP_BORDER_SUN 0x81D4
06906 #define GL_GLOBAL_ALPHA_SUN 0x81D9
06907 #define GL_GLOBAL_ALPHA_FACTOR_SUN 0x81DA
06908 #define GL_QUAD_MESH_SUN 0x8614
06909 #define GL_TRIANGLE_MESH_SUN 0x8615
06910 #define GL_SLICE_ACCUM_SUN 0x85CC
06911 #define GL_RESTART_SUN 0x0001
06912 #define GL_REPLACE_MIDDLE_SUN 0x0002
06913 #define GL_REPLACE_OLDEST_SUN 0x0003
06914 #define GL_TRIANGLE_LIST_SUN 0x81D7
06915 #define GL_REPLACEMENT_CODE_SUN 0x81D8
06916 #define GL_REPLACEMENT_CODE_ARRAY_SUN 0x85C0
06917 #define GL_REPLACEMENT_CODE_ARRAY_TYPE_SUN 0x85C1
06918 #define GL_REPLACEMENT_CODE_ARRAY_STRIDE_SUN 0x85C2
06919 #define GL_REPLACEMENT_CODE_ARRAY_POINTER_SUN 0x85C3
06920 #define GL_R1UI_V3F_SUN 0x85C4
06921 #define GL_R1UI_C4UB_V3F_SUN 0x85C5
06922 #define GL_R1UI_C3F_V3F_SUN 0x85C6
06923 #define GL_R1UI_N3F_V3F_SUN 0x85C7
06924 #define GL_R1UI_C4F_N3F_V3F_SUN 0x85C8
06925 #define GL_R1UI_T2F_V3F_SUN 0x85C9
06926 #define GL_R1UI_T2F_N3F_V3F_SUN 0x85CA
06927 #define GL_R1UI_T2F_C4F_N3F_V3F_SUN 0x85CB
06928 #define GL_PHONG_WIN 0x80EA
06929 #define GL_PHONG_HINT_WIN 0x80EB
06930 #define GL_FOG_SPECULAR_TEXTURE_WIN 0x80EC
06931 #ifndef GL_3DFX_multisample
06932 #define GL_3DFX_multisample 1
06933 GLAPI int GLAD_GL_3DFX_multisample;
06934 #endif
06935 #ifndef GL_3DFX_tbuffer
06936 #define GL_3DFX_tbuffer 1
06937 GLAPI int GLAD_GL_3DFX_tbuffer;
06938 typedef void (APIENTRY PFNGLTBUFFERMASK3DFXPROC) (GLuint mask);
06939 GLAPI PFNGLTBUFFERMASK3DFXPROC glad_glTbufferMask3DFX;
06940 #define glTbufferMask3DFX glad_glTbufferMask3DFX
06941 #endif
06942 #ifndef GL_3DFX_texture_compression_FXT1
06943 #define GL_3DFX_texture_compression_FXT1 1
06944 GLAPI int GLAD_GL_3DFX_texture_compression_FXT1;
06945 #endif
06946 #ifndef GL_AMD_blend_minmax_factor
06947 #define GL_AMD_blend_minmax_factor 1
06948 GLAPI int GLAD_GL_AMD_blend_minmax_factor;
06949 #endif
06950 #ifndef GL_AMD_conservative_depth
06951 #define GL_AMD_conservative_depth 1
06952 GLAPI int GLAD_GL_AMD_conservative_depth;
06953 #endif
06954 #ifndef GL_AMD_debug_output
06955 #define GL_AMD_debug_output 1
06956 GLAPI int GLAD_GL_AMD_debug_output;
06957 typedef void (APIENTRY PFNGLDEBUGMESSAGEENABLEAMDPROC) (GLenum category, GLenum severity, GLsizei
count, const GLuint *ids, GLboolean enabled);
06958 GLAPI PFNGLDEBUGMESSAGEENABLEAMDPROC glad_glDebugMessageEnableAMD;
06959 #define glDebugMessageEnableAMD glad_glDebugMessageEnableAMD

```



```

06960 typedef void (APIENTRY PFNGLDEBUGMESSAGEINSERTAMDPROC)(GLenum category, GLenum severity, GLuint id,
06961 GLsizei length, const GLchar *buf);
06962 GLAPI PFNGLDEBUGMESSAGEINSERTAMDPROC glad_glDebugMessageInsertAMD;
06963 #define glDebugMessageInsertAMD glad_glDebugMessageInsertAMD
06964 typedef void (APIENTRY PFNGLDEBUGMESSAGECALLBACKAMDPROC)(GLDEBUGPROCAMD callback, void *userParam);
06965 GLAPI PFNGLDEBUGMESSAGECALLBACKAMDPROC glad_glDebugMessageCallbackAMD;
06966 #define glDebugMessageCallbackAMD glad_glDebugMessageCallbackAMD
06967 typedef GLuint (APIENTRY PFNGLGETDEBUGMESSAGELOGAMDPROC)(GLuint count, GLsizei bufSize, GLenum
06968 *categories, GLenum *severities, GLuint *ids, GLsizei *lengths, GLchar *message);
06969 GLAPI PFNGLGETDEBUGMESSAGELOGAMDPROC glad_glGetDebugMessageLogAMD;
06970 #define glGetDebugMessageLogAMD glad_glGetDebugMessageLogAMD
06971 #endif
06972 #ifndef GL_AMD_depth_clamp_separate
06973 #define GL_AMD_depth_clamp_separate 1
06974 GLAPI int GLAD_GL_AMD_depth_clamp_separate;
06975 #endif
06976 #ifndef GL_AMD_draw_buffers_blend
06977 #define GL_AMD_draw_buffers_blend 1
06978 GLAPI int GLAD_GL_AMD_draw_buffers_blend;
06979 typedef void (APIENTRY PFNGLBLENDFUNCINDEXEDAMDPROC)(GLuint buf, GLenum src, GLenum dst);
06980 GLAPI PFNGLBLENDFUNCINDEXEDAMDPROC glad_glBlendFuncIndexedAMD;
06981 #define glBlendFuncIndexedAMD glad_glBlendFuncIndexedAMD
06982 typedef void (APIENTRY PFNGLBLENDFUNCSEPARATEINDEXEDAMDPROC)(GLuint buf, GLenum srcRGB, GLenum
06983 dstRGB, GLenum srcAlpha, GLenum dstAlpha);
06984 GLAPI PFNGLBLENDFUNCSEPARATEINDEXEDAMDPROC glad_glBlendFuncSeparateIndexedAMD;
06985 #define glBlendFuncSeparateIndexedAMD glad_glBlendFuncSeparateIndexedAMD
06986 typedef void (APIENTRY PFNGLBLENDEQUATIONINDEXEDAMDPROC)(GLuint buf, GLenum mode);
06987 GLAPI PFNGLBLENDEQUATIONINDEXEDAMDPROC glad_glBlendEquationIndexedAMD;
06988 #define glBlendEquationIndexedAMD glad_glBlendEquationIndexedAMD
06989 typedef void (APIENTRY PFNGLBLENDEQUATIONSEPARATEINDEXEDAMDPROC)(GLuint buf, GLenum modeRGB, GLenum
06990 modeAlpha);
06991 GLAPI PFNGLBLENDEQUATIONSEPARATEINDEXEDAMDPROC glad_glBlendEquationSeparateIndexedAMD;
06992 #define glBlendEquationSeparateIndexedAMD glad_glBlendEquationSeparateIndexedAMD
06993 #endif
06994 #ifndef GL_AMD_framebuffer_multisample_advanced
06995 #define GL_AMD_framebuffer_multisample_advanced 1
06996 GLAPI int GLAD_GL_AMD_framebuffer_multisample_advanced;
06997 typedef void (APIENTRY PFNGLRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC)(GLenum target, GLsizei
06998 samples, GLsizei storageSamples, GLenum internalformat, GLsizei width, GLsizei height);
06999 GLAPI PFNGLRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC
07000 glad_glRenderbufferStorageMultisampleAdvancedAMD;
07001 #define glRenderbufferStorageMultisampleAdvancedAMD glad_glRenderbufferStorageMultisampleAdvancedAMD
07002 typedef void (APIENTRY PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC)(GLuint renderbuffer,
07003 GLsizei samples, GLsizei storageSamples, GLenum internalformat, GLsizei width, GLsizei height);
07004 GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC
07005 glad_glNamedRenderbufferStorageMultisampleAdvancedAMD;
07006 #define glNamedRenderbufferStorageMultisampleAdvancedAMD
07007 glad_glNamedRenderbufferStorageMultisampleAdvancedAMD
07008 #endif
07009 #ifndef GL_AMD_framebuffer_sample_positions
07010 #define GL_AMD_framebuffer_sample_positions 1
07011 GLAPI int GLAD_GL_AMD_framebuffer_sample_positions;
07012 typedef void (APIENTRY PFNGLFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC)(GLenum target, GLuint numsamples,
07013 GLuint pixelindex, const GLfloat *values);
07014 GLAPI PFNGLFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC glad_glFramebufferSamplePositionsfvAMD;
07015 #define glFramebufferSamplePositionsfvAMD glad_glFramebufferSamplePositionsfvAMD
07016 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC)(GLuint framebuffer, GLuint
07017 numsamples, GLuint pixelindex, const GLfloat *values);
07018 GLAPI PFNGLNAMEDFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC glad_glNamedFramebufferSamplePositionsfvAMD;
07019 #define glNamedFramebufferSamplePositionsfvAMD glad_glNamedFramebufferSamplePositionsfvAMD
07020 typedef void (APIENTRY PFNGLGETFRAMEBUFFERPARAMETERFVAMDPROC)(GLenum target, GLenum pname, GLuint
07021 numsamples, GLuint pixelindex, GLsizei size, GLfloat *values);
07022 GLAPI PFNGLGETFRAMEBUFFERPARAMETERFVAMDPROC glad_glGetFramebufferParameterfvAMD;
07023 #define glGetFramebufferParameterfvAMD glad_glGetFramebufferParameterfvAMD
07024 typedef void (APIENTRY PFNGLGETNAMEDFRAMEBUFFERPARAMETERFVAMDPROC)(GLuint framebuffer, GLenum pname,
07025 GLuint numsamples, GLuint pixelindex, GLsizei size, GLfloat *values);
07026 GLAPI PFNGLGETNAMEDFRAMEBUFFERPARAMETERFVAMDPROC glad_glGetNamedFramebufferParameterfvAMD;
07027 #define glGetNamedFramebufferParameterfvAMD glad_glGetNamedFramebufferParameterfvAMD
07028 #endif
07029 #ifndef GL_AMD_gcn_shader
07030 #define GL_AMD_gcn_shader 1
07031 GLAPI int GLAD_GL_AMD_gcn_shader;
07032 #endif
07033 #ifndef GL_AMD_gpu_shader_half_float
07034 #define GL_AMD_gpu_shader_half_float 1
07035 GLAPI int GLAD_GL_AMD_gpu_shader_half_float;
07036 #endif
07037 #ifndef GL_AMD_gpu_shader_int16
07038 #define GL_AMD_gpu_shader_int16 1
07039 GLAPI int GLAD_GL_AMD_gpu_shader_int16;
07040 #endif
07041 #ifndef GL_AMD_gpu_shader_int64
07042 #define GL_AMD_gpu_shader_int64 1
07043 GLAPI int GLAD_GL_AMD_gpu_shader_int64;
07044 typedef void (APIENTRY PFNGLUNIFORM1I64NVPROC)(GLint location, GLint64EXT x);
07045 GLAPI PFNGLUNIFORM1I64NVPROC glad_glUniform1i64NV;
07046 #define glUniform1i64NV glad_glUniform1i64NV

```

```
07034 typedef void (APIENTRY PFNGLUNIFORM2I64NVPROC)(GLint location, GLint64EXT x, GLint64EXT y);
07035 GLAPI PFNGLUNIFORM2I64NVPROC glad_glUniform2i64NV;
07036 #define glUniform2i64NV glad_glUniform2i64NV
07037 typedef void (APIENTRY PFNGLUNIFORM3I64NVPROC)(GLint location, GLint64EXT x, GLint64EXT y, GLint64EXT
z);
07038 GLAPI PFNGLUNIFORM3I64NVPROC glad_glUniform3i64NV;
07039 #define glUniform3i64NV glad_glUniform3i64NV
07040 typedef void (APIENTRY PFNGLUNIFORM4I64NVPROC)(GLint location, GLint64EXT x, GLint64EXT y, GLint64EXT
z, GLint64EXT w);
07041 GLAPI PFNGLUNIFORM4I64NVPROC glad_glUniform4i64NV;
07042 #define glUniform4i64NV glad_glUniform4i64NV
07043 typedef void (APIENTRY PFNGLUNIFORM1I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT
*value);
07044 GLAPI PFNGLUNIFORM1I64VNVPROC glad_glUniform1i64vNV;
07045 #define glUniform1i64vNV glad_glUniform1i64vNV
07046 typedef void (APIENTRY PFNGLUNIFORM2I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT
*value);
07047 GLAPI PFNGLUNIFORM2I64VNVPROC glad_glUniform2i64vNV;
07048 #define glUniform2i64vNV glad_glUniform2i64vNV
07049 typedef void (APIENTRY PFNGLUNIFORM3I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT
*value);
07050 GLAPI PFNGLUNIFORM3I64VNVPROC glad_glUniform3i64vNV;
07051 #define glUniform3i64vNV glad_glUniform3i64vNV
07052 typedef void (APIENTRY PFNGLUNIFORM4I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT
*value);
07053 GLAPI PFNGLUNIFORM4I64VNVPROC glad_glUniform4i64vNV;
07054 #define glUniform4i64vNV glad_glUniform4i64vNV
07055 typedef void (APIENTRY PFNGLUNIFORM1UI64NVPROC)(GLint location, GLuint64EXT x);
07056 GLAPI PFNGLUNIFORM1UI64NVPROC glad_glUniform1ui64NV;
07057 #define glUniform1ui64NV glad_glUniform1ui64NV
07058 typedef void (APIENTRY PFNGLUNIFORM2UI64NVPROC)(GLint location, GLuint64EXT x, GLuint64EXT y);
07059 GLAPI PFNGLUNIFORM2UI64NVPROC glad_glUniform2ui64NV;
07060 #define glUniform2ui64NV glad_glUniform2ui64NV
07061 typedef void (APIENTRY PFNGLUNIFORM3UI64NVPROC)(GLint location, GLuint64EXT x, GLuint64EXT y,
GLuint64EXT z);
07062 GLAPI PFNGLUNIFORM3UI64NVPROC glad_glUniform3ui64NV;
07063 #define glUniform3ui64NV glad_glUniform3ui64NV
07064 typedef void (APIENTRY PFNGLUNIFORM4UI64NVPROC)(GLint location, GLuint64EXT x, GLuint64EXT y,
GLuint64EXT z, GLuint64EXT w);
07065 GLAPI PFNGLUNIFORM4UI64NVPROC glad_glUniform4ui64NV;
07066 #define glUniform4ui64NV glad_glUniform4ui64NV
07067 typedef void (APIENTRY PFNGLUNIFORM1UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT
*value);
07068 GLAPI PFNGLUNIFORM1UI64VNVPROC glad_glUniform1ui64vNV;
07069 #define glUniform1ui64vNV glad_glUniform1ui64vNV
07070 typedef void (APIENTRY PFNGLUNIFORM2UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT
*value);
07071 GLAPI PFNGLUNIFORM2UI64VNVPROC glad_glUniform2ui64vNV;
07072 #define glUniform2ui64vNV glad_glUniform2ui64vNV
07073 typedef void (APIENTRY PFNGLUNIFORM3UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT
*value);
07074 GLAPI PFNGLUNIFORM3UI64VNVPROC glad_glUniform3ui64vNV;
07075 #define glUniform3ui64vNV glad_glUniform3ui64vNV
07076 typedef void (APIENTRY PFNGLUNIFORM4UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT
*value);
07077 GLAPI PFNGLUNIFORM4UI64VNVPROC glad_glUniform4ui64vNV;
07078 #define glUniform4ui64vNV glad_glUniform4ui64vNV
07079 typedef void (APIENTRY PFNGLGETUNIFORMI64VNVPROC)(GLuint program, GLint location, GLint64EXT
*params);
07080 GLAPI PFNGLGETUNIFORMI64VNVPROC glad_glGetUniformi64vNV;
07081 #define glGetUniformi64vNV glad_glGetUniformi64vNV
07082 typedef void (APIENTRY PFNGLGETUNIFORMUI64VNVPROC)(GLuint program, GLint location, GLuint64EXT
*params);
07083 GLAPI PFNGLGETUNIFORMUI64VNVPROC glad_glGetUniformui64vNV;
07084 #define glGetUniformui64vNV glad_glGetUniformui64vNV
07085 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1I64NVPROC)(GLuint program, GLint location, GLint64EXT x);
07086 GLAPI PFNGLPROGRAMUNIFORM1I64NVPROC glad_glProgramUniform1i64NV;
07087 #define glProgramUniform1i64NV glad_glProgramUniform1i64NV
07088 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2I64NVPROC)(GLuint program, GLint location, GLint64EXT x,
GLint64EXT y);
07089 GLAPI PFNGLPROGRAMUNIFORM2I64NVPROC glad_glProgramUniform2i64NV;
07090 #define glProgramUniform2i64NV glad_glProgramUniform2i64NV
07091 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3I64NVPROC)(GLuint program, GLint location, GLint64EXT x,
GLint64EXT y, GLint64EXT z);
07092 GLAPI PFNGLPROGRAMUNIFORM3I64NVPROC glad_glProgramUniform3i64NV;
07093 #define glProgramUniform3i64NV glad_glProgramUniform3i64NV
07094 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4I64NVPROC)(GLuint program, GLint location, GLint64EXT x,
GLint64EXT y, GLint64EXT z, GLint64EXT w);
07095 GLAPI PFNGLPROGRAMUNIFORM4I64NVPROC glad_glProgramUniform4i64NV;
07096 #define glProgramUniform4i64NV glad_glProgramUniform4i64NV
07097 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1I64VNVPROC)(GLuint program, GLint location, GLsizei count,
const GLint64EXT *value);
07098 GLAPI PFNGLPROGRAMUNIFORM1I64VNVPROC glad_glProgramUniform1i64vNV;
07099 #define glProgramUniform1i64vNV glad_glProgramUniform1i64vNV
07100 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2I64VNVPROC)(GLuint program, GLint location, GLsizei count,
const GLint64EXT *value);
07101 GLAPI PFNGLPROGRAMUNIFORM2I64VNVPROC glad_glProgramUniform2i64vNV;
```

```

07102 #define glProgramUniform2i64vNV glad_glProgramUniform2i64vNV
07103 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3I64VNVPROC) (GLuint program, GLint location, GLsizei count,
07104 const GLint64EXT *value);
07104 GLAPI PFNGLPROGRAMUNIFORM3I64VNVPROC glad_glProgramUniform3i64vNV;
07105 #define glProgramUniform3i64vNV glad_glProgramUniform3i64vNV
07106 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4I64VNVPROC) (GLuint program, GLint location, GLsizei count,
07107 const GLint64EXT *value);
07107 GLAPI PFNGLPROGRAMUNIFORM4I64VNVPROC glad_glProgramUniform4i64vNV;
07108 #define glProgramUniform4i64vNV glad_glProgramUniform4i64vNV
07109 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UI64NVPROC) (GLuint program, GLint location, GLuint64EXT
07110 x);
07110 GLAPI PFNGLPROGRAMUNIFORM1UI64NVPROC glad_glProgramUniform1ui64NV;
07111 #define glProgramUniform1ui64NV glad_glProgramUniform1ui64NV
07112 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UI64NVPROC) (GLuint program, GLint location, GLuint64EXT x,
07113 GLuint64EXT y);
07113 GLAPI PFNGLPROGRAMUNIFORM2UI64NVPROC glad_glProgramUniform2ui64NV;
07114 #define glProgramUniform2ui64NV glad_glProgramUniform2ui64NV
07115 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UI64NVPROC) (GLuint program, GLint location, GLuint64EXT x,
07116 GLuint64EXT y, GLuint64EXT z);
07116 GLAPI PFNGLPROGRAMUNIFORM3UI64NVPROC glad_glProgramUniform3ui64NV;
07117 #define glProgramUniform3ui64NV glad_glProgramUniform3ui64NV
07118 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UI64NVPROC) (GLuint program, GLint location, GLuint64EXT x,
07119 GLuint64EXT y, GLuint64EXT z, GLuint64EXT w);
07119 GLAPI PFNGLPROGRAMUNIFORM4UI64NVPROC glad_glProgramUniform4ui64NV;
07120 #define glProgramUniform4ui64NV glad_glProgramUniform4ui64NV
07121 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UI64VNVPROC) (GLuint program, GLint location, GLsizei
07122 count, const GLuint64EXT *value);
07122 GLAPI PFNGLPROGRAMUNIFORM1UI64VNVPROC glad_glProgramUniform1ui64vNV;
07123 #define glProgramUniform1ui64vNV glad_glProgramUniform1ui64vNV
07124 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UI64VNVPROC) (GLuint program, GLint location, GLsizei
07125 count, const GLuint64EXT *value);
07125 GLAPI PFNGLPROGRAMUNIFORM2UI64VNVPROC glad_glProgramUniform2ui64vNV;
07126 #define glProgramUniform2ui64vNV glad_glProgramUniform2ui64vNV
07127 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UI64VNVPROC) (GLuint program, GLint location, GLsizei
07128 count, const GLuint64EXT *value);
07128 GLAPI PFNGLPROGRAMUNIFORM3UI64VNVPROC glad_glProgramUniform3ui64vNV;
07129 #define glProgramUniform3ui64vNV glad_glProgramUniform3ui64vNV
07130 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UI64VNVPROC) (GLuint program, GLint location, GLsizei
07131 count, const GLuint64EXT *value);
07131 GLAPI PFNGLPROGRAMUNIFORM4UI64VNVPROC glad_glProgramUniform4ui64vNV;
07132 #define glProgramUniform4ui64vNV glad_glProgramUniform4ui64vNV
07133 #endif
07134 #ifndef GL_AMD_interleaved_elements
07135 #define GL_AMD_interleaved_elements 1
07136 GLAPI int GLAD_GL_AMD_interleaved_elements;
07137 typedef void (APIENTRY PFNGLVERTEXATTRIBPARAMETERIAMDPROC) (GLuint index, GLenum pname, GLint param);
07138 GLAPI PFNGLVERTEXATTRIBPARAMETERIAMDPROC glad_glVertexAttribParameteriAMD;
07139 #define glVertexAttribParameteriAMD glad_glVertexAttribParameteriAMD
07140 #endif
07141 #ifndef GL_AMD_multi_draw_indirect
07142 #define GL_AMD_multi_draw_indirect 1
07143 GLAPI int GLAD_GL_AMD_multi_draw_indirect;
07144 typedef void (APIENTRY PFNGLMULTIDRAWARRAYSINDIRECTAMDPROC) (GLenum mode, const void *indirect,
07145 GLsizei primcount, GLsizei stride);
07145 GLAPI PFNGLMULTIDRAWARRAYSINDIRECTAMDPROC glad_glMultiDrawArraysIndirectAMD;
07146 #define glMultiDrawArraysIndirectAMD glad_glMultiDrawArraysIndirectAMD
07147 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSINDIRECTAMDPROC) (GLenum mode, GLenum type, const void
07148 *indirect, GLsizei primcount, GLsizei stride);
07148 GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTAMDPROC glad_glMultiDrawElementsIndirectAMD;
07149 #define glMultiDrawElementsIndirectAMD glad_glMultiDrawElementsIndirectAMD
07150 #endif
07151 #ifndef GL_AMD_name_gen_delete
07152 #define GL_AMD_name_gen_delete 1
07153 GLAPI int GLAD_GL_AMD_name_gen_delete;
07154 typedef void (APIENTRY PFNGLGENNAMESAMDPROC) (GLenum identifier, GLuint num, GLuint *names);
07155 GLAPI PFNGLGENNAMESAMDPROC glad_glGenNamesAMD;
07156 #define glGenNamesAMD glad_glGenNamesAMD
07157 typedef void (APIENTRY PFNGLDELETENAMESAMDPROC) (GLenum identifier, GLuint num, const GLuint *names);
07158 GLAPI PFNGLDELETENAMESAMDPROC glad_glDeleteNamesAMD;
07159 #define glDeleteNamesAMD glad_glDeleteNamesAMD
07160 typedef GLboolean (APIENTRY PFNGLISNAMEAMDPROC) (GLenum identifier, GLuint name);
07161 GLAPI PFNGLISNAMEAMDPROC glad_glIsNameAMD;
07162 #define glIsNameAMD glad_glIsNameAMD
07163 #endif
07164 #ifndef GL_AMD_occlusion_query_event
07165 #define GL_AMD_occlusion_query_event 1
07166 GLAPI int GLAD_GL_AMD_occlusion_query_event;
07167 typedef void (APIENTRY PFNGLQUERYOBJECTPARAMETERUIAMDPROC) (GLenum target, GLuint id, GLenum pname,
07168 GLuint param);
07168 GLAPI PFNGLQUERYOBJECTPARAMETERUIAMDPROC glad_glQueryObjectParameteruiAMD;
07169 #define glQueryObjectParameteruiAMD glad_glQueryObjectParameteruiAMD
07170 #endif
07171 #ifndef GL_AMD_performance_monitor
07172 #define GL_AMD_performance_monitor 1
07173 GLAPI int GLAD_GL_AMD_performance_monitor;
07174 typedef void (APIENTRY PFNGLGETPERFMONITORGROUPSAMDPROC) (GLint *numGroups, GLsizei groupsSize, GLuint
07175 *groups);

```

```

07175 GLAPI PFNGLGETPERFMONITORGROUPSAMDPROC glad_glGetPerfMonitorGroupsAMD;
07176 #define glGetPerfMonitorGroupsAMD glad_glGetPerfMonitorGroupsAMD
07177 typedef void (APIENTRY PFNGLGETPERFMONITORCOUNTERSAMDPROC) (GLuint group, GLint *numCounters, GLint
    *maxActiveCounters, GLsizei counterSize, GLuint *counters);
07178 GLAPI PFNGLGETPERFMONITORCOUNTERSAMDPROC glad_glGetPerfMonitorCountersAMD;
07179 #define glGetPerfMonitorCountersAMD glad_glGetPerfMonitorCountersAMD
07180 typedef void (APIENTRY PFNGLGETPERFMONITORGROUPSTRINGAMDPROC) (GLuint group, GLsizei bufSize, GLsizei
    *length, GLchar *groupString);
07181 GLAPI PFNGLGETPERFMONITORGROUPSTRINGAMDPROC glad_glGetPerfMonitorGroupStringAMD;
07182 #define glGetPerfMonitorGroupStringAMD glad_glGetPerfMonitorGroupStringAMD
07183 typedef void (APIENTRY PFNGLGETPERFMONITORCOUNTERSTRINGAMDPROC) (GLuint group, GLuint counter, GLsizei
    bufSize, GLsizei *length, GLchar *counterString);
07184 GLAPI PFNGLGETPERFMONITORCOUNTERSTRINGAMDPROC glad_glGetPerfMonitorCounterStringAMD;
07185 #define glGetPerfMonitorCounterStringAMD glad_glGetPerfMonitorCounterStringAMD
07186 typedef void (APIENTRY PFNGLGETPERFMONITORCOUNTERINFOAMDPROC) (GLuint group, GLuint counter, GLenum
    pname, void *data);
07187 GLAPI PFNGLGETPERFMONITORCOUNTERINFOAMDPROC glad_glGetPerfMonitorCounterInfoAMD;
07188 #define glGetPerfMonitorCounterInfoAMD glad_glGetPerfMonitorCounterInfoAMD
07189 typedef void (APIENTRY PFNGLGENPERFMONITORSAMDPROC) (GLsizei n, GLuint *monitors);
07190 GLAPI PFNGLGENPERFMONITORSAMDPROC glad_glGenPerfMonitorsAMD;
07191 #define glGenPerfMonitorsAMD glad_glGenPerfMonitorsAMD
07192 typedef void (APIENTRY PFNGLDELETEPERFMONITORSAMDPROC) (GLsizei n, GLuint *monitors);
07193 GLAPI PFNGLDELETEPERFMONITORSAMDPROC glad_glDeletePerfMonitorsAMD;
07194 #define glDeletePerfMonitorsAMD glad_glDeletePerfMonitorsAMD
07195 typedef void (APIENTRY PFNGLSELECTPERFMONITORCOUNTERSAMDPROC) (GLuint monitor, GLboolean enable,
    GLuint group, GLint numCounters, GLuint *counterList);
07196 GLAPI PFNGLSELECTPERFMONITORCOUNTERSAMDPROC glad_glSelectPerfMonitorCountersAMD;
07197 #define glSelectPerfMonitorCountersAMD glad_glSelectPerfMonitorCountersAMD
07198 typedef void (APIENTRY PFNGLBEGINPERFMONITORAMDPROC) (GLuint monitor);
07199 GLAPI PFNGLBEGINPERFMONITORAMDPROC glad_glBeginPerfMonitorAMD;
07200 #define glBeginPerfMonitorAMD glad_glBeginPerfMonitorAMD
07201 typedef void (APIENTRY PFNGLENDPERFMONITORAMDPROC) (GLuint monitor);
07202 GLAPI PFNGLENDPERFMONITORAMDPROC glad_glEndPerfMonitorAMD;
07203 #define glEndPerfMonitorAMD glad_glEndPerfMonitorAMD
07204 typedef void (APIENTRY PFNGLGETPERFMONITORCOUNTERDATAAMDPROC) (GLuint monitor, GLenum pname, GLsizei
    dataSize, GLuint *data, GLint *bytesWritten);
07205 GLAPI PFNGLGETPERFMONITORCOUNTERDATAAMDPROC glad_glGetPerfMonitorCounterDataAMD;
07206 #define glGetPerfMonitorCounterDataAMD glad_glGetPerfMonitorCounterDataAMD
07207 #endif
07208 #ifndef GL_AMD_pinned_memory
07209 #define GL_AMD_pinned_memory 1
07210 GLAPI int GLAD_GL_AMD_pinned_memory;
07211 #endif
07212 #ifndef GL_AMD_query_buffer_object
07213 #define GL_AMD_query_buffer_object 1
07214 GLAPI int GLAD_GL_AMD_query_buffer_object;
07215 #endif
07216 #ifndef GL_AMD_sample_positions
07217 #define GL_AMD_sample_positions 1
07218 GLAPI int GLAD_GL_AMD_sample_positions;
07219 typedef void (APIENTRY PFNGLSETMULTISAMPLEFVAMDPROC) (GLenum pname, GLuint index, const GLfloat *val);
07220 GLAPI PFNGLSETMULTISAMPLEFVAMDPROC glad_glSetMultisamplefvAMD;
07221 #define glSetMultisamplefvAMD glad_glSetMultisamplefvAMD
07222 #endif
07223 #ifndef GL_AMD_seamless_cubemap_per_texture
07224 #define GL_AMD_seamless_cubemap_per_texture 1
07225 GLAPI int GLAD_GL_AMD_seamless_cubemap_per_texture;
07226 #endif
07227 #ifndef GL_AMD_shader_atomic_counter_ops
07228 #define GL_AMD_shader_atomic_counter_ops 1
07229 GLAPI int GLAD_GL_AMD_shader_atomic_counter_ops;
07230 #endif
07231 #ifndef GL_AMD_shader_ballot
07232 #define GL_AMD_shader_ballot 1
07233 GLAPI int GLAD_GL_AMD_shader_ballot;
07234 #endif
07235 #ifndef GL_AMD_shader_explicit_vertex_parameter
07236 #define GL_AMD_shader_explicit_vertex_parameter 1
07237 GLAPI int GLAD_GL_AMD_shader_explicit_vertex_parameter;
07238 #endif
07239 #ifndef GL_AMD_shader_gpu_shader_half_float_fetch
07240 #define GL_AMD_shader_gpu_shader_half_float_fetch 1
07241 GLAPI int GLAD_GL_AMD_shader_gpu_shader_half_float_fetch;
07242 #endif
07243 #ifndef GL_AMD_shader_image_load_store_lod
07244 #define GL_AMD_shader_image_load_store_lod 1
07245 GLAPI int GLAD_GL_AMD_shader_image_load_store_lod;
07246 #endif
07247 #ifndef GL_AMD_shader_stencil_export
07248 #define GL_AMD_shader_stencil_export 1
07249 GLAPI int GLAD_GL_AMD_shader_stencil_export;
07250 #endif
07251 #ifndef GL_AMD_shader_trinary_minmax
07252 #define GL_AMD_shader_trinary_minmax 1
07253 GLAPI int GLAD_GL_AMD_shader_trinary_minmax;
07254 #endif
07255 #ifndef GL_AMD_sparse_texture

```

```

07256 #define GL_AMD_sparse_texture 1
07257 GLAPI int GLAD_GL_AMD_sparse_texture;
07258 typedef void (APIENTRY PFNGLTEXTURESTORAGEPARSEAMDPROC)(GLenum target, GLenum internalFormat, GLsizei
width, GLsizei height, GLsizei depth, GLsizei layers, GLbitfield flags);
07259 GLAPI PFNGLTEXTURESTORAGEPARSEAMDPROC glad_glTexStorageSparseAMD;
07260 #define glTexStorageSparseAMD glad_glTexStorageSparseAMD
07261 typedef void (APIENTRY PFNGLTEXTURESTORAGEPARSEAMDPROC)(GLuint texture, GLenum target, GLenum
internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLsizei layers, GLbitfield flags);
07262 GLAPI PFNGLTEXTURESTORAGEPARSEAMDPROC glad_glTextureStorageSparseAMD;
07263 #define glTextureStorageSparseAMD glad_glTextureStorageSparseAMD
07264 #endif
07265 #ifndef GL_AMD_stencil_operation_extended
07266 #define GL_AMD_stencil_operation_extended 1
07267 GLAPI int GLAD_GL_AMD_stencil_operation_extended;
07268 typedef void (APIENTRY PFNGLSTENCILOPVALUEAMDPROC)(GLenum face, GLuint value);
07269 GLAPI PFNGLSTENCILOPVALUEAMDPROC glad_glStencilOpValueAMD;
07270 #define glStencilOpValueAMD glad_glStencilOpValueAMD
07271 #endif
07272 #ifndef GL_AMD_texture_gather_bias_lod
07273 #define GL_AMD_texture_gather_bias_lod 1
07274 GLAPI int GLAD_GL_AMD_texture_gather_bias_lod;
07275 #endif
07276 #ifndef GL_AMD_texture_texture4
07277 #define GL_AMD_texture_texture4 1
07278 GLAPI int GLAD_GL_AMD_texture_texture4;
07279 #endif
07280 #ifndef GL_AMD_transform_feedback3_lines_triangles
07281 #define GL_AMD_transform_feedback3_lines_triangles 1
07282 GLAPI int GLAD_GL_AMD_transform_feedback3_lines_triangles;
07283 #endif
07284 #ifndef GL_AMD_transform_feedback4
07285 #define GL_AMD_transform_feedback4 1
07286 GLAPI int GLAD_GL_AMD_transform_feedback4;
07287 #endif
07288 #ifndef GL_AMD_vertex_shader_layer
07289 #define GL_AMD_vertex_shader_layer 1
07290 GLAPI int GLAD_GL_AMD_vertex_shader_layer;
07291 #endif
07292 #ifndef GL_AMD_vertex_shader_tessellator
07293 #define GL_AMD_vertex_shader_tessellator 1
07294 GLAPI int GLAD_GL_AMD_vertex_shader_tessellator;
07295 typedef void (APIENTRY PFNGLTESSELLATIONFACTORAMDPROC)(GLfloat factor);
07296 GLAPI PFNGLTESSELLATIONFACTORAMDPROC glad_glTessellationFactorAMD;
07297 #define glTessellationFactorAMD glad_glTessellationFactorAMD
07298 typedef void (APIENTRY PFNGLTESSELLATIONMODEAMDPROC)(GLenum mode);
07299 GLAPI PFNGLTESSELLATIONMODEAMDPROC glad_glTessellationModeAMD;
07300 #define glTessellationModeAMD glad_glTessellationModeAMD
07301 #endif
07302 #ifndef GL_AMD_vertex_shader_viewport_index
07303 #define GL_AMD_vertex_shader_viewport_index 1
07304 GLAPI int GLAD_GL_AMD_vertex_shader_viewport_index;
07305 #endif
07306 #ifndef GL_APPLE_aux_depth_stencil
07307 #define GL_APPLE_aux_depth_stencil 1
07308 GLAPI int GLAD_GL_APPLE_aux_depth_stencil;
07309 #endif
07310 #ifndef GL_APPLE_client_storage
07311 #define GL_APPLE_client_storage 1
07312 GLAPI int GLAD_GL_APPLE_client_storage;
07313 #endif
07314 #ifndef GL_APPLE_element_array
07315 #define GL_APPLE_element_array 1
07316 GLAPI int GLAD_GL_APPLE_element_array;
07317 typedef void (APIENTRY PFNGLELEMENTPOINTERAPPLEPROC)(GLenum type, const void *pointer);
07318 GLAPI PFNGLELEMENTPOINTERAPPLEPROC glad_glElementPointerAPPLE;
07319 #define glElementPointerAPPLE glad_glElementPointerAPPLE
07320 typedef void (APIENTRY PFNGLDRAWELEMENTARRAYAPPLEPROC)(GLenum mode, GLint first, GLsizei count);
07321 GLAPI PFNGLDRAWELEMENTARRAYAPPLEPROC glad_glDrawElementArrayAPPLE;
07322 #define glDrawElementArrayAPPLE glad_glDrawElementArrayAPPLE
07323 typedef void (APIENTRY PFNGLDRAWRANGEELEMENTARRAYAPPLEPROC)(GLenum mode, GLuint start, GLuint end,
GLint first, GLsizei count);
07324 GLAPI PFNGLDRAWRANGEELEMENTARRAYAPPLEPROC glad_glDrawRangeElementArrayAPPLE;
07325 #define glDrawRangeElementArrayAPPLE glad_glDrawRangeElementArrayAPPLE
07326 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTARRAYAPPLEPROC)(GLenum mode, const GLint *first, const
GLsizei *count, GLsizei primcount);
07327 GLAPI PFNGLMULTIDRAWELEMENTARRAYAPPLEPROC glad_glMultiDrawElementArrayAPPLE;
07328 #define glMultiDrawElementArrayAPPLE glad_glMultiDrawElementArrayAPPLE
07329 typedef void (APIENTRY PFNGLMULTIDRAWRANGEELEMENTARRAYAPPLEPROC)(GLenum mode, GLuint start, GLuint
end, const GLint *first, const GLsizei *count, GLsizei primcount);
07330 GLAPI PFNGLMULTIDRAWRANGEELEMENTARRAYAPPLEPROC glad_glMultiDrawRangeElementArrayAPPLE;
07331 #define glMultiDrawRangeElementArrayAPPLE glad_glMultiDrawRangeElementArrayAPPLE
07332 #endif
07333 #ifndef GL_APPLE_fence
07334 #define GL_APPLE_fence 1
07335 GLAPI int GLAD_GL_APPLE_fence;
07336 typedef void (APIENTRY PFNGLGENFENCESAPPLEPROC)(GLsizei n, GLuint *fences);
07337 GLAPI PFNGLGENFENCESAPPLEPROC glad_glGenFencesAPPLE;

```



```

07338 #define glGenFencesAPPLE glad_glGenFencesAPPLE
07339 typedef void (APIENTRY PFNGLDELETEFENCESAPPLEPROC) (GLsizei n, const GLuint *fences);
07340 GLAPI PFNGLDELETEFENCESAPPLEPROC glad_glDeleteFencesAPPLE;
07341 #define glDeleteFencesAPPLE glad_glDeleteFencesAPPLE
07342 typedef void (APIENTRY PFNGLSETFENCEAPPLEPROC) (GLuint fence);
07343 GLAPI PFNGLSETFENCEAPPLEPROC glad_glSetFenceAPPLE;
07344 #define glSetFenceAPPLE glad_glSetFenceAPPLE
07345 typedef GLboolean (APIENTRY PFNGLISFENCEAPPLEPROC) (GLuint fence);
07346 GLAPI PFNGLISFENCEAPPLEPROC glad_glIsFenceAPPLE;
07347 #define glIsFenceAPPLE glad_glIsFenceAPPLE
07348 typedef GLboolean (APIENTRY PFNGLTESTFENCEAPPLEPROC) (GLuint fence);
07349 GLAPI PFNGLTESTFENCEAPPLEPROC glad_glTestFenceAPPLE;
07350 #define glTestFenceAPPLE glad_glTestFenceAPPLE
07351 typedef void (APIENTRY PFNGLFINISHFENCEAPPLEPROC) (GLuint fence);
07352 GLAPI PFNGLFINISHFENCEAPPLEPROC glad_glFinishFenceAPPLE;
07353 #define glFinishFenceAPPLE glad_glFinishFenceAPPLE
07354 typedef GLboolean (APIENTRY PFNGLTESTOBJECTAPPLEPROC) (GLenum object, GLuint name);
07355 GLAPI PFNGLTESTOBJECTAPPLEPROC glad_glTestObjectAPPLE;
07356 #define glTestObjectAPPLE glad_glTestObjectAPPLE
07357 typedef void (APIENTRY PFNGLFINISHOBJECTAPPLEPROC) (GLenum object, GLint name);
07358 GLAPI PFNGLFINISHOBJECTAPPLEPROC glad_glFinishObjectAPPLE;
07359 #define glFinishObjectAPPLE glad_glFinishObjectAPPLE
07360 #endif
07361 #ifndef GL_APPLE_float_pixels
07362 #define GL_APPLE_float_pixels 1
07363 GLAPI int GLAD_GL_APPLE_float_pixels;
07364 #endif
07365 #ifndef GL_APPLE_flush_buffer_range
07366 #define GL_APPLE_flush_buffer_range 1
07367 GLAPI int GLAD_GL_APPLE_flush_buffer_range;
07368 typedef void (APIENTRY PFNGLBUFFERPARAMETERIAPPLEPROC) (GLenum target, GLenum pname, GLint param);
07369 GLAPI PFNGLBUFFERPARAMETERIAPPLEPROC glad_glBufferParameteriAPPLE;
07370 #define glBufferParameteriAPPLE glad_glBufferParameteriAPPLE
07371 typedef void (APIENTRY PFNGLFLUSHMAPPEDBUFFERRANGEAPPLEPROC) (GLenum target, GLintptr offset,
GLsizei ptr size);
07372 GLAPI PFNGLFLUSHMAPPEDBUFFERRANGEAPPLEPROC glad_glFlushMappedBufferRangeAPPLE;
07373 #define glFlushMappedBufferRangeAPPLE glad_glFlushMappedBufferRangeAPPLE
07374 #endif
07375 #ifndef GL_APPLE_object_purgeable
07376 #define GL_APPLE_object_purgeable 1
07377 GLAPI int GLAD_GL_APPLE_object_purgeable;
07378 typedef GLenum (APIENTRY PFNGLOBJECTPURGEABLEAPPLEPROC) (GLenum objectType, GLuint name, GLenum
option);
07379 GLAPI PFNGLOBJECTPURGEABLEAPPLEPROC glad_glObjectPurgeableAPPLE;
07380 #define glObjectPurgeableAPPLE glad_glObjectPurgeableAPPLE
07381 typedef GLenum (APIENTRY PFNGLOBJECTUNPURGEABLEAPPLEPROC) (GLenum objectType, GLuint name, GLenum
option);
07382 GLAPI PFNGLOBJECTUNPURGEABLEAPPLEPROC glad_glObjectUnpurgeableAPPLE;
07383 #define glObjectUnpurgeableAPPLE glad_glObjectUnpurgeableAPPLE
07384 typedef void (APIENTRY PFNGLGETOBJECTPARAMETERIVAPPLEPROC) (GLenum objectType, GLuint name, GLenum
pname, GLint *params);
07385 GLAPI PFNGLGETOBJECTPARAMETERIVAPPLEPROC glad_glGetObjectParameterivAPPLE;
07386 #define glGetObjectParameterivAPPLE glad_glGetObjectParameterivAPPLE
07387 #endif
07388 #ifndef GL_APPLE_rgb_422
07389 #define GL_APPLE_rgb_422 1
07390 GLAPI int GLAD_GL_APPLE_rgb_422;
07391 #endif
07392 #ifndef GL_APPLE_row_bytes
07393 #define GL_APPLE_row_bytes 1
07394 GLAPI int GLAD_GL_APPLE_row_bytes;
07395 #endif
07396 #ifndef GL_APPLE_specular_vector
07397 #define GL_APPLE_specular_vector 1
07398 GLAPI int GLAD_GL_APPLE_specular_vector;
07399 #endif
07400 #ifndef GL_APPLE_texture_range
07401 #define GL_APPLE_texture_range 1
07402 GLAPI int GLAD_GL_APPLE_texture_range;
07403 typedef void (APIENTRY PFNGLTEXTURERANGEAPPLEPROC) (GLenum target, GLsizei length, const void
*pointer);
07404 GLAPI PFNGLTEXTURERANGEAPPLEPROC glad_glTextureRangeAPPLE;
07405 #define glTextureRangeAPPLE glad_glTextureRangeAPPLE
07406 typedef void (APIENTRY PFNGLGETTEXPARAMETERPOINTERVAPPLEPROC) (GLenum target, GLenum pname, void
**params);
07407 GLAPI PFNGLGETTEXPARAMETERPOINTERVAPPLEPROC glad_glGetTexParameterPointervAPPLE;
07408 #define glGetTexParameterPointervAPPLE glad_glGetTexParameterPointervAPPLE
07409 #endif
07410 #ifndef GL_APPLE_transform_hint
07411 #define GL_APPLE_transform_hint 1
07412 GLAPI int GLAD_GL_APPLE_transform_hint;
07413 #endif
07414 #ifndef GL_APPLE_vertex_array_object
07415 #define GL_APPLE_vertex_array_object 1
07416 GLAPI int GLAD_GL_APPLE_vertex_array_object;
07417 typedef void (APIENTRY PFNGLBINDVERTEXARRAYAPPLEPROC) (GLuint array);
07418 GLAPI PFNGLBINDVERTEXARRAYAPPLEPROC glad_glBindVertexArrayAPPLE;

```

```

07419 #define glBindVertexArrayAPPLE glad_glBindVertexArrayAPPLE
07420 typedef void (APIENTRY PFNGLDELETEVERTEXARRAYSAPPLEPROC) (GLsizei n, const GLuint *arrays);
07421 GLAPI PFNGLDELETEVERTEXARRAYSAPPLEPROC glad_glDeleteVertexArraysAPPLE;
07422 #define glDeleteVertexArraysAPPLE glad_glDeleteVertexArraysAPPLE
07423 typedef void (APIENTRY PFNGLGENVERTEXARRAYSAPPLEPROC) (GLsizei n, GLuint *arrays);
07424 GLAPI PFNGLGENVERTEXARRAYSAPPLEPROC glad_glGenVertexArraysAPPLE;
07425 #define glGenVertexArraysAPPLE glad_glGenVertexArraysAPPLE
07426 typedef GLboolean (APIENTRY PFNGLISVERTEXARRAYAPPLEPROC) (GLuint array);
07427 GLAPI PFNGLISVERTEXARRAYAPPLEPROC glad_glIsVertexArrayAPPLE;
07428 #define glIsVertexArrayAPPLE glad_glIsVertexArrayAPPLE
07429 #endif
07430 #ifndef GL_APPLE_vertex_array_range
07431 #define GL_APPLE_vertex_array_range 1
07432 GLAPI int GLAD_GL_APPLE_vertex_array_range;
07433 typedef void (APIENTRY PFNGLVERTEXARRAYRANGEAPPLEPROC) (GLsizei length, void *pointer);
07434 GLAPI PFNGLVERTEXARRAYRANGEAPPLEPROC glad_glVertexArrayRangeAPPLE;
07435 #define glVertexArrayRangeAPPLE glad_glVertexArrayRangeAPPLE
07436 typedef void (APIENTRY PFNGLFLUSHVERTEXARRAYRANGEAPPLEPROC) (GLsizei length, void *pointer);
07437 GLAPI PFNGLFLUSHVERTEXARRAYRANGEAPPLEPROC glad_glFlushVertexArrayRangeAPPLE;
07438 #define glFlushVertexArrayRangeAPPLE glad_glFlushVertexArrayRangeAPPLE
07439 typedef void (APIENTRY PFNGLVERTEXARRAYPARAMETERIAPPLEPROC) (GLenum pname, GLint param);
07440 GLAPI PFNGLVERTEXARRAYPARAMETERIAPPLEPROC glad_glVertexArrayParameteriAPPLE;
07441 #define glVertexArrayParameteriAPPLE glad_glVertexArrayParameteriAPPLE
07442 #endif
07443 #ifndef GL_APPLE_vertex_program_evaluators
07444 #define GL_APPLE_vertex_program_evaluators 1
07445 GLAPI int GLAD_GL_APPLE_vertex_program_evaluators;
07446 typedef void (APIENTRY PFNGLENABLEVERTEXATTRIBAPPLEPROC) (GLuint index, GLenum pname);
07447 GLAPI PFNGLENABLEVERTEXATTRIBAPPLEPROC glad_glEnableVertexAttribAPPLE;
07448 #define glEnableVertexAttribAPPLE glad_glEnableVertexAttribAPPLE
07449 typedef void (APIENTRY PFNGLDISABLEVERTEXATTRIBAPPLEPROC) (GLuint index, GLenum pname);
07450 GLAPI PFNGLDISABLEVERTEXATTRIBAPPLEPROC glad_glDisableVertexAttribAPPLE;
07451 #define glDisableVertexAttribAPPLE glad_glDisableVertexAttribAPPLE
07452 typedef GLboolean (APIENTRY PFNGLISVERTEXATTRIBENABLEDAPPLEPROC) (GLuint index, GLenum pname);
07453 GLAPI PFNGLISVERTEXATTRIBENABLEDAPPLEPROC glad_glIsVertexAttribEnabledAPPLE;
07454 #define glIsVertexAttribEnabledAPPLE glad_glIsVertexAttribEnabledAPPLE
07455 typedef void (APIENTRY PFNGLMAPVERTEXATTRIB1DAPPLEPROC) (GLuint index, GLuint size, GLdouble u1,
GLdouble u2, GLint stride, GLint order, const GLdouble *points);
07456 GLAPI PFNGLMAPVERTEXATTRIB1DAPPLEPROC glad_glMapVertexAttrib1dAPPLE;
07457 #define glMapVertexAttrib1dAPPLE glad_glMapVertexAttrib1dAPPLE
07458 typedef void (APIENTRY PFNGLMAPVERTEXATTRIB1FAPPLEPROC) (GLuint index, GLuint size, GLfloat u1,
GLfloat u2, GLint stride, GLint order, const GLfloat *points);
07459 GLAPI PFNGLMAPVERTEXATTRIB1FAPPLEPROC glad_glMapVertexAttrib1fAPPLE;
07460 #define glMapVertexAttrib1fAPPLE glad_glMapVertexAttrib1fAPPLE
07461 typedef void (APIENTRY PFNGLMAPVERTEXATTRIB2DAPPLEPROC) (GLuint index, GLuint size, GLdouble u1,
GLdouble u2, GLint ustride, GLint uorder, GLdouble v1, GLdouble v2, GLint vstride, GLint vorder, const
GLdouble *points);
07462 GLAPI PFNGLMAPVERTEXATTRIB2DAPPLEPROC glad_glMapVertexAttrib2dAPPLE;
07463 #define glMapVertexAttrib2dAPPLE glad_glMapVertexAttrib2dAPPLE
07464 typedef void (APIENTRY PFNGLMAPVERTEXATTRIB2FAPPLEPROC) (GLuint index, GLuint size, GLfloat u1,
GLfloat u2, GLint ustride, GLint uorder, GLfloat v1, GLfloat v2, GLint vstride, GLint vorder, const
GLfloat *points);
07465 GLAPI PFNGLMAPVERTEXATTRIB2FAPPLEPROC glad_glMapVertexAttrib2fAPPLE;
07466 #define glMapVertexAttrib2fAPPLE glad_glMapVertexAttrib2fAPPLE
07467 #endif
07468 #ifndef GL_APPLE_ycbcr_422
07469 #define GL_APPLE_ycbcr_422 1
07470 GLAPI int GLAD_GL_APPLE_ycbcr_422;
07471 #endif
07472 #ifndef GL_ARB_ES2_compatibility
07473 #define GL_ARB_ES2_compatibility 1
07474 GLAPI int GLAD_GL_ARB_ES2_compatibility;
07475 #endif
07476 #ifndef GL_ARB_ES3_1_compatibility
07477 #define GL_ARB_ES3_1_compatibility 1
07478 GLAPI int GLAD_GL_ARB_ES3_1_compatibility;
07479 typedef void (APIENTRY PFNGLMEMORYBARRIERBYREGIONPROC) (GLbitfield barriers);
07480 GLAPI PFNGLMEMORYBARRIERBYREGIONPROC glad_glMemoryBarrierByRegion;
07481 #define glMemoryBarrierByRegion glad_glMemoryBarrierByRegion
07482 #endif
07483 #ifndef GL_ARB_ES3_2_compatibility
07484 #define GL_ARB_ES3_2_compatibility 1
07485 GLAPI int GLAD_GL_ARB_ES3_2_compatibility;
07486 typedef void (APIENTRY PFNGLPRIMITIVEBOUNDINGBOXARBPROC) (GLfloat minX, GLfloat minY, GLfloat minZ,
GLfloat minW, GLfloat maxX, GLfloat maxY, GLfloat maxZ, GLfloat maxW);
07487 GLAPI PFNGLPRIMITIVEBOUNDINGBOXARBPROC glad_glPrimitiveBoundingBoxARB;
07488 #define glPrimitiveBoundingBoxARB glad_glPrimitiveBoundingBoxARB
07489 #endif
07490 #ifndef GL_ARB_ES3_compatibility
07491 #define GL_ARB_ES3_compatibility 1
07492 GLAPI int GLAD_GL_ARB_ES3_compatibility;
07493 #endif
07494 #ifndef GL_ARB_arrays_of_arrays
07495 #define GL_ARB_arrays_of_arrays 1
07496 GLAPI int GLAD_GL_ARB_arrays_of_arrays;
07497 #endif
07498 #ifndef GL_ARB_base_instance

```

```

07499 #define GL_ARB_base_instance 1
07500 GLAPI int GLAD_GL_ARB_base_instance;
07501 typedef void (APIENTRY PFNGLDRAWARRAYSINSTANCEDBASEINSTANCEPROC)(GLenum mode, GLint first, GLsizei
count, GLsizei instancecount, GLuint baseinstance);
07502 GLAPI PFNGLDRAWARRAYSINSTANCEDBASEINSTANCEPROC glad_glDrawArraysInstancedBaseInstance;
07503 #define glDrawArraysInstancedBaseInstance glad_glDrawArraysInstancedBaseInstance
07504 typedef void (APIENTRY PFNGLDRAWELEMENTSINSTANCEDBASEINSTANCEPROC)(GLenum mode, GLsizei count, GLenum
type, const void *indices, GLsizei instancecount, GLuint baseinstance);
07505 GLAPI PFNGLDRAWELEMENTSINSTANCEDBASEINSTANCEPROC glad_glDrawElementsInstancedBaseInstance;
07506 #define glDrawElementsInstancedBaseInstance glad_glDrawElementsInstancedBaseInstance
07507 typedef void (APIENTRY PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXBASEINSTANCEPROC)(GLenum mode, GLsizei
count, GLenum type, const void *indices, GLsizei instancecount, GLint basevertex, GLuint
baseinstance);
07508 GLAPI PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXBASEINSTANCEPROC
glad_glDrawElementsInstancedBaseVertexBaseInstance;
07509 #define glDrawElementsInstancedBaseVertexBaseInstance
glad_glDrawElementsInstancedBaseVertexBaseInstance
07510 #endif
07511 #ifndef GL_ARB_bindless_texture
07512 #define GL_ARB_bindless_texture 1
07513 GLAPI int GLAD_GL_ARB_bindless_texture;
07514 typedef GLuint64 (APIENTRY PFNGLGETTEXTUREHANDLEARBPROC)(GLuint texture);
07515 GLAPI PFNGLGETTEXTUREHANDLEARBPROC glad_glGetTextureHandleARB;
07516 #define glGetTextureHandleARB glad_glGetTextureHandleARB
07517 typedef GLuint64 (APIENTRY PFNGLGETTEXTURESAMPLERHANDLEARBPROC)(GLuint texture, GLuint sampler);
07518 GLAPI PFNGLGETTEXTURESAMPLERHANDLEARBPROC glad_glGetTextureSamplerHandleARB;
07519 #define glGetTextureSamplerHandleARB glad_glGetTextureSamplerHandleARB
07520 typedef void (APIENTRY PFNGLMAKETEXTUREHANDLERESIDENTARBPROC)(GLuint64 handle);
07521 GLAPI PFNGLMAKETEXTUREHANDLERESIDENTARBPROC glad_glMakeTextureHandleResidentARB;
07522 #define glMakeTextureHandleResidentARB glad_glMakeTextureHandleResidentARB
07523 typedef void (APIENTRY PFNGLMAKETEXTUREHANDLENONRESIDENTARBPROC)(GLuint64 handle);
07524 GLAPI PFNGLMAKETEXTUREHANDLENONRESIDENTARBPROC glad_glMakeTextureHandleNonResidentARB;
07525 #define glMakeTextureHandleNonResidentARB glad_glMakeTextureHandleNonResidentARB
07526 typedef GLuint64 (APIENTRY PFNGLGETIMAGEHANDLEARBPROC)(GLuint texture, GLint level, GLboolean
layered, GLint layer, GLenum format);
07527 GLAPI PFNGLGETIMAGEHANDLEARBPROC glad_glGetImageHandleARB;
07528 #define glGetImageHandleARB glad_glGetImageHandleARB
07529 typedef void (APIENTRY PFNGLMAKEIMAGEHANDLERESIDENTARBPROC)(GLuint64 handle, GLenum access);
07530 GLAPI PFNGLMAKEIMAGEHANDLERESIDENTARBPROC glad_glMakeImageHandleResidentARB;
07531 #define glMakeImageHandleResidentARB glad_glMakeImageHandleResidentARB
07532 typedef void (APIENTRY PFNGLMAKEIMAGEHANDLENONRESIDENTARBPROC)(GLuint64 handle);
07533 GLAPI PFNGLMAKEIMAGEHANDLENONRESIDENTARBPROC glad_glMakeImageHandleNonResidentARB;
07534 #define glMakeImageHandleNonResidentARB glad_glMakeImageHandleNonResidentARB
07535 typedef void (APIENTRY PFNGLUNIFORMHANDLEUI64ARBPROC)(GLint location, GLuint64 value);
07536 GLAPI PFNGLUNIFORMHANDLEUI64ARBPROC glad_glUniformHandleui64ARB;
07537 #define glUniformHandleui64ARB glad_glUniformHandleui64ARB
07538 typedef void (APIENTRY PFNGLUNIFORMHANDLEUI64VARBPROC)(GLint location, GLsizei count, const GLuint64
*value);
07539 GLAPI PFNGLUNIFORMHANDLEUI64VARBPROC glad_glUniformHandleui64vARB;
07540 #define glUniformHandleui64vARB glad_glUniformHandleui64vARB
07541 typedef void (APIENTRY PFNGLPROGRAMUNIFORMHANDLEUI64ARBPROC)(GLuint program, GLint location, GLuint64
value);
07542 GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64ARBPROC glad_glProgramUniformHandleui64ARB;
07543 #define glProgramUniformHandleui64ARB glad_glProgramUniformHandleui64ARB
07544 typedef void (APIENTRY PFNGLPROGRAMUNIFORMHANDLEUI64VARBPROC)(GLuint program, GLint location, GLsizei
count, const GLuint64 *values);
07545 GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64VARBPROC glad_glProgramUniformHandleui64vARB;
07546 #define glProgramUniformHandleui64vARB glad_glProgramUniformHandleui64vARB
07547 typedef GLboolean (APIENTRY PFNGLISTEXTUREHANDLERESIDENTARBPROC)(GLuint64 handle);
07548 GLAPI PFNGLISTEXTUREHANDLERESIDENTARBPROC glad_glIsTextureHandleResidentARB;
07549 #define glIsTextureHandleResidentARB glad_glIsTextureHandleResidentARB
07550 typedef GLboolean (APIENTRY PFNGLISIMAGEHANDLERESIDENTARBPROC)(GLuint64 handle);
07551 GLAPI PFNGLISIMAGEHANDLERESIDENTARBPROC glad_glIsImageHandleResidentARB;
07552 #define glIsImageHandleResidentARB glad_glIsImageHandleResidentARB
07553 typedef void (APIENTRY PFNGLVERTEXATTRIBLUI64ARBPROC)(GLuint index, GLuint64EXT x);
07554 GLAPI PFNGLVERTEXATTRIBLUI64ARBPROC glad_glVertexAttribLui64ARB;
07555 #define glVertexAttribLui64ARB glad_glVertexAttribLui64ARB
07556 typedef void (APIENTRY PFNGLVERTEXATTRIBLUI64VARBPROC)(GLuint index, const GLuint64EXT *v);
07557 GLAPI PFNGLVERTEXATTRIBLUI64VARBPROC glad_glVertexAttribLui64vARB;
07558 #define glVertexAttribLui64vARB glad_glVertexAttribLui64vARB
07559 typedef void (APIENTRY PFNGLGETVERTEXATTRIBLUI64VARBPROC)(GLuint index, GLenum pname, GLuint64EXT
*params);
07560 GLAPI PFNGLGETVERTEXATTRIBLUI64VARBPROC glad_glGetVertexAttribLui64vARB;
07561 #define glGetVertexAttribLui64vARB glad_glGetVertexAttribLui64vARB
07562 #endif
07563 #ifndef GL_ARB_blend_func_extended
07564 #define GL_ARB_blend_func_extended 1
07565 GLAPI int GLAD_GL_ARB_blend_func_extended;
07566 #endif
07567 #ifndef GL_ARB_buffer_storage
07568 #define GL_ARB_buffer_storage 1
07569 GLAPI int GLAD_GL_ARB_buffer_storage;
07570 typedef void (APIENTRY PFNGLBUFFERSTORAGEPROC)(GLenum target, GLsizeiptr size, const void *data,
GLbitfield flags);
07571 GLAPI PFNGLBUFFERSTORAGEPROC glad_glBufferStorage;
07572 #define glBufferStorage glad_glBufferStorage
07573 #endif

```



```

07574 #ifndef GL_ARB_cl_event
07575 #define GL_ARB_cl_event 1
07576 GLAPI int GLAD_GL_ARB_cl_event;
07577 typedef GLsync (APIENTRY PFNGLCREATESYNCFROMCLEVENTARBPROC) (struct _cl_context *context, struct
    _cl_event *event, GLbitfield flags);
07578 GLAPI PFNGLCREATESYNCFROMCLEVENTARBPROC glad_glCreateSyncFromCLeventARB;
07579 #define glCreateSyncFromCLeventARB glad_glCreateSyncFromCLeventARB
07580 #endif
07581 #ifndef GL_ARB_clear_buffer_object
07582 #define GL_ARB_clear_buffer_object 1
07583 GLAPI int GLAD_GL_ARB_clear_buffer_object;
07584 typedef void (APIENTRY PFNGLCLEARBUFFERDATAPROC) (GLenum target, GLenum internalformat, GLenum format,
    GLenum type, const void *data);
07585 GLAPI PFNGLCLEARBUFFERDATAPROC glad_glClearBufferData;
07586 #define glClearBufferData glad_glClearBufferData
07587 typedef void (APIENTRY PFNGLCLEARBUFFERSUBDATAPROC) (GLenum target, GLenum internalformat, GLintptr
    offset, GLsizeiptr size, GLenum format, GLenum type, const void *data);
07588 GLAPI PFNGLCLEARBUFFERSUBDATAPROC glad_glClearBufferSubData;
07589 #define glClearBufferSubData glad_glClearBufferSubData
07590 #endif
07591 #ifndef GL_ARB_clear_texture
07592 #define GL_ARB_clear_texture 1
07593 GLAPI int GLAD_GL_ARB_clear_texture;
07594 typedef void (APIENTRY PFNGLCLEARTEXIMAGEPROC) (GLuint texture, GLint level, GLenum format, GLenum
    type, const void *data);
07595 GLAPI PFNGLCLEARTEXIMAGEPROC glad_glClearTexImage;
07596 #define glClearTexImage glad_glClearTexImage
07597 typedef void (APIENTRY PFNGLCLEARTEXSUBIMAGEPROC) (GLuint texture, GLint level, GLint xoffset, GLint
    yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type,
    const void *data);
07598 GLAPI PFNGLCLEARTEXSUBIMAGEPROC glad_glClearTexSubImage;
07599 #define glClearTexSubImage glad_glClearTexSubImage
07600 #endif
07601 #ifndef GL_ARB_clip_control
07602 #define GL_ARB_clip_control 1
07603 GLAPI int GLAD_GL_ARB_clip_control;
07604 typedef void (APIENTRY PFNGLCLIPCONTROLPROC) (GLenum origin, GLenum depth);
07605 GLAPI PFNGLCLIPCONTROLPROC glad_glClipControl;
07606 #define glClipControl glad_glClipControl
07607 #endif
07608 #ifndef GL_ARB_color_buffer_float
07609 #define GL_ARB_color_buffer_float 1
07610 GLAPI int GLAD_GL_ARB_color_buffer_float;
07611 typedef void (APIENTRY PFNGLCLAMPCOLORARBPROC) (GLenum target, GLenum clamp);
07612 GLAPI PFNGLCLAMPCOLORARBPROC glad_glClampColorARB;
07613 #define glClampColorARB glad_glClampColorARB
07614 #endif
07615 #ifndef GL_ARB_compatibility
07616 #define GL_ARB_compatibility 1
07617 GLAPI int GLAD_GL_ARB_compatibility;
07618 #endif
07619 #ifndef GL_ARB_compressed_texture_pixel_storage
07620 #define GL_ARB_compressed_texture_pixel_storage 1
07621 GLAPI int GLAD_GL_ARB_compressed_texture_pixel_storage;
07622 #endif
07623 #ifndef GL_ARB_compute_shader
07624 #define GL_ARB_compute_shader 1
07625 GLAPI int GLAD_GL_ARB_compute_shader;
07626 typedef void (APIENTRY PFNGLDISPATCHCOMPUTEPROC) (GLuint num_groups_x, GLuint num_groups_y, GLuint
    num_groups_z);
07627 GLAPI PFNGLDISPATCHCOMPUTEPROC glad_glDispatchCompute;
07628 #define glDispatchCompute glad_glDispatchCompute
07629 typedef void (APIENTRY PFNGLDISPATCHCOMPUTEINDIRECTPROC) (GLintptr indirect);
07630 GLAPI PFNGLDISPATCHCOMPUTEINDIRECTPROC glad_glDispatchComputeIndirect;
07631 #define glDispatchComputeIndirect glad_glDispatchComputeIndirect
07632 #endif
07633 #ifndef GL_ARB_compute_variable_group_size
07634 #define GL_ARB_compute_variable_group_size 1
07635 GLAPI int GLAD_GL_ARB_compute_variable_group_size;
07636 typedef void (APIENTRY PFNGLDISPATCHCOMPUTEGROUPSIZEARBPROC) (GLuint num_groups_x, GLuint
    num_groups_y, GLuint num_groups_z, GLuint group_size_x, GLuint group_size_y, GLuint group_size_z);
07637 GLAPI PFNGLDISPATCHCOMPUTEGROUPSIZEARBPROC glad_glDispatchComputeGroupSizeARB;
07638 #define glDispatchComputeGroupSizeARB glad_glDispatchComputeGroupSizeARB
07639 #endif
07640 #ifndef GL_ARB_conditional_render_inverted
07641 #define GL_ARB_conditional_render_inverted 1
07642 GLAPI int GLAD_GL_ARB_conditional_render_inverted;
07643 #endif
07644 #ifndef GL_ARB_conservative_depth
07645 #define GL_ARB_conservative_depth 1
07646 GLAPI int GLAD_GL_ARB_conservative_depth;
07647 #endif
07648 #ifndef GL_ARB_copy_buffer
07649 #define GL_ARB_copy_buffer 1
07650 GLAPI int GLAD_GL_ARB_copy_buffer;
07651 #endif
07652 #ifndef GL_ARB_copy_image

```

```

07653 #define GL_ARB_copy_image 1
07654 GLAPI int GLAD_GL_ARB_copy_image;
07655 typedef void (APIENTRY PFNGLCOPYIMAGESUBDATAPROC) (GLuint srcName, GLenum srcTarget, GLint srcLevel,
GLint srcX, GLint srcY, GLint srcZ, GLuint dstName, GLenum dstTarget, GLint dstLevel, GLint dstX,
GLint dstY, GLint dstZ, GLsizei srcWidth, GLsizei srcHeight, GLsizei srcDepth);
07656 GLAPI PFNGLCOPYIMAGESUBDATAPROC glad_glCopyImageSubData;
07657 #define glCopyImageSubData glad_glCopyImageSubData
07658 #endif
07659 #ifndef GL_ARB_cull_distance
07660 #define GL_ARB_cull_distance 1
07661 GLAPI int GLAD_GL_ARB_cull_distance;
07662 #endif
07663 #ifndef GL_ARB_debug_output
07664 #define GL_ARB_debug_output 1
07665 GLAPI int GLAD_GL_ARB_debug_output;
07666 typedef void (APIENTRY PFNGLDEBUGMESSAGECONTROLARBPROC) (GLenum source, GLenum type, GLenum severity,
GLsizei count, const GLuint *ids, GLboolean enabled);
07667 GLAPI PFNGLDEBUGMESSAGECONTROLARBPROC glad_glDebugMessageControlARB;
07668 #define glDebugMessageControlARB glad_glDebugMessageControlARB
07669 typedef void (APIENTRY PFNGLDEBUGMESSAGEINSERTARBPROC) (GLenum source, GLenum type, GLuint id, GLenum
severity, GLsizei length, const GLchar *buf);
07670 GLAPI PFNGLDEBUGMESSAGEINSERTARBPROC glad_glDebugMessageInsertARB;
07671 #define glDebugMessageInsertARB glad_glDebugMessageInsertARB
07672 typedef void (APIENTRY PFNGLDEBUGMESSAGECALLBACKARBPROC) (GLDEBUGPROCARB callback, const void
*userParam);
07673 GLAPI PFNGLDEBUGMESSAGECALLBACKARBPROC glad_glDebugMessageCallbackARB;
07674 #define glDebugMessageCallbackARB glad_glDebugMessageCallbackARB
07675 typedef GLuint (APIENTRY PFNGLGETDEBUGMESSAGELOGARBPROC) (GLuint count, GLsizei bufSize, GLenum
*sources, GLenum *types, GLuint *ids, GLenum *severities, GLsizei *lengths, GLchar *messageLog);
07676 GLAPI PFNGLGETDEBUGMESSAGELOGARBPROC glad_glGetDebugMessageLogARB;
07677 #define glGetDebugMessageLogARB glad_glGetDebugMessageLogARB
07678 #endif
07679 #ifndef GL_ARB_depth_buffer_float
07680 #define GL_ARB_depth_buffer_float 1
07681 GLAPI int GLAD_GL_ARB_depth_buffer_float;
07682 #endif
07683 #ifndef GL_ARB_depth_clamp
07684 #define GL_ARB_depth_clamp 1
07685 GLAPI int GLAD_GL_ARB_depth_clamp;
07686 #endif
07687 #ifndef GL_ARB_depth_texture
07688 #define GL_ARB_depth_texture 1
07689 GLAPI int GLAD_GL_ARB_depth_texture;
07690 #endif
07691 #ifndef GL_ARB_derivative_control
07692 #define GL_ARB_derivative_control 1
07693 GLAPI int GLAD_GL_ARB_derivative_control;
07694 #endif
07695 #ifndef GL_ARB_direct_state_access
07696 #define GL_ARB_direct_state_access 1
07697 GLAPI int GLAD_GL_ARB_direct_state_access;
07698 typedef void (APIENTRY PFNGLCREATETRANSFORMFEEDBACKSPROC) (GLsizei n, GLuint *ids);
07699 GLAPI PFNGLCREATETRANSFORMFEEDBACKSPROC glad_glCreateTransformFeedbacks;
07700 #define glCreateTransformFeedbacks glad_glCreateTransformFeedbacks
07701 typedef void (APIENTRY PFNGLTRANSFORMFEEDBACKBUFFERBASEPROC) (GLuint xfb, GLuint index, GLuint
buffer);
07702 GLAPI PFNGLTRANSFORMFEEDBACKBUFFERBASEPROC glad_glTransformFeedbackBufferBase;
07703 #define glTransformFeedbackBufferBase glad_glTransformFeedbackBufferBase
07704 typedef void (APIENTRY PFNGLTRANSFORMFEEDBACKBUFFERRANGEPROC) (GLuint xfb, GLuint index, GLuint
buffer, GLintptr offset, GLsizeiptr size);
07705 GLAPI PFNGLTRANSFORMFEEDBACKBUFFERRANGEPROC glad_glTransformFeedbackBufferRange;
07706 #define glTransformFeedbackBufferRange glad_glTransformFeedbackBufferRange
07707 typedef void (APIENTRY PFNGLGETTRANSFORMFEEDBACKIVPROC) (GLuint xfb, GLenum pname, GLint *param);
07708 GLAPI PFNGLGETTRANSFORMFEEDBACKIVPROC glad_glGetTransformFeedbackiv;
07709 #define glGetTransformFeedbackiv glad_glGetTransformFeedbackiv
07710 typedef void (APIENTRY PFNGLGETTRANSFORMFEEDBACKI_VPROC) (GLuint xfb, GLenum pname, GLuint index,
GLint *param);
07711 GLAPI PFNGLGETTRANSFORMFEEDBACKI_VPROC glad_glGetTransformFeedbacki_v;
07712 #define glGetTransformFeedbacki_v glad_glGetTransformFeedbacki_v
07713 typedef void (APIENTRY PFNGLGETTRANSFORMFEEDBACKI64_VPROC) (GLuint xfb, GLenum pname, GLuint index,
GLint64 *param);
07714 GLAPI PFNGLGETTRANSFORMFEEDBACKI64_VPROC glad_glGetTransformFeedbacki64_v;
07715 #define glGetTransformFeedbacki64_v glad_glGetTransformFeedbacki64_v
07716 typedef void (APIENTRY PFNGLCREATEBUFFERSPROC) (GLsizei n, GLuint *buffers);
07717 GLAPI PFNGLCREATEBUFFERSPROC glad_glCreateBuffers;
07718 #define glCreateBuffers glad_glCreateBuffers
07719 typedef void (APIENTRY PFNGLNAMEDBUFFERSTORAGEPROC) (GLuint buffer, GLsizeiptr size, const void *data,
GLbitfield flags);
07720 GLAPI PFNGLNAMEDBUFFERSTORAGEPROC glad_glNamedBufferStorage;
07721 #define glNamedBufferStorage glad_glNamedBufferStorage
07722 typedef void (APIENTRY PFNGLNAMEDBUFFERDATAPROC) (GLuint buffer, GLsizeiptr size, const void *data,
GLenum usage);
07723 GLAPI PFNGLNAMEDBUFFERDATAPROC glad_glNamedBufferData;
07724 #define glNamedBufferData glad_glNamedBufferData
07725 typedef void (APIENTRY PFNGLNAMEDBUFFERSUBDATAPROC) (GLuint buffer, GLintptr offset, GLsizeiptr size,
const void *data);
07726 GLAPI PFNGLNAMEDBUFFERSUBDATAPROC glad_glNamedBufferSubData;

```

```

07727 #define glNamedBufferSubData glad_glNamedBufferSubData
07728 typedef void (APIENTRY PFNGLCOPYNAMEDBUFFERSUBDATAPROC) (GLuint readBuffer, GLuint writeBuffer,
07729 GLintptr readOffset, GLintptr writeOffset, GLsizeiptr size);
07729 GLAPI PFNGLCOPYNAMEDBUFFERSUBDATAPROC glad_glCopyNamedBufferSubData;
07730 #define glCopyNamedBufferSubData glad_glCopyNamedBufferSubData
07731 typedef void (APIENTRY PFNGLCLEARNAMEDBUFFERDATAPROC) (GLuint buffer, GLenum internalformat, GLenum
07732 format, GLenum type, const void *data);
07732 GLAPI PFNGLCLEARNAMEDBUFFERDATAPROC glad_glClearNamedBufferData;
07733 #define glClearNamedBufferData glad_glClearNamedBufferData
07734 typedef void (APIENTRY PFNGLCLEARNAMEDBUFFERSUBDATAPROC) (GLuint buffer, GLenum internalformat,
07735 GLintptr offset, GLsizeiptr size, GLenum format, GLenum type, const void *data);
07735 GLAPI PFNGLCLEARNAMEDBUFFERSUBDATAPROC glad_glClearNamedBufferSubData;
07736 #define glClearNamedBufferSubData glad_glClearNamedBufferSubData
07737 typedef void * (APIENTRY PFNGLMAPNAMEDBUFFERPROC) (GLuint buffer, GLenum access);
07738 GLAPI PFNGLMAPNAMEDBUFFERPROC glad_glMapNamedBuffer;
07739 #define glMapNamedBuffer glad_glMapNamedBuffer
07740 typedef void * (APIENTRY PFNGLMAPNAMEDBUFFERRANGEPROC) (GLuint buffer, GLintptr offset, GLsizeiptr
07741 length, GLbitfield access);
07741 GLAPI PFNGLMAPNAMEDBUFFERRANGEPROC glad_glMapNamedBufferRange;
07742 #define glMapNamedBufferRange glad_glMapNamedBufferRange
07743 typedef GLboolean (APIENTRY PFNGLUNMAPNAMEDBUFFERPROC) (GLuint buffer);
07744 GLAPI PFNGLUNMAPNAMEDBUFFERPROC glad_glUnmapNamedBuffer;
07745 #define glUnmapNamedBuffer glad_glUnmapNamedBuffer
07746 typedef void (APIENTRY PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEPROC) (GLuint buffer, GLintptr offset,
07747 GLsizeiptr length);
07747 GLAPI PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEPROC glad_glFlushMappedNamedBufferRange;
07748 #define glFlushMappedNamedBufferRange glad_glFlushMappedNamedBufferRange
07749 typedef void (APIENTRY PFNGLGETNAMEDBUFFERPARAMETERIVPROC) (GLuint buffer, GLenum pname, GLint
07750 *params);
07750 GLAPI PFNGLGETNAMEDBUFFERPARAMETERIVPROC glad_glGetNamedBufferParameteriv;
07751 #define glGetNamedBufferParameteriv glad_glGetNamedBufferParameteriv
07752 typedef void (APIENTRY PFNGLGETNAMEDBUFFERPARAMETERI64VPROC) (GLuint buffer, GLenum pname, GLint64
07753 *params);
07753 GLAPI PFNGLGETNAMEDBUFFERPARAMETERI64VPROC glad_glGetNamedBufferParameteri64v;
07754 #define glGetNamedBufferParameteri64v glad_glGetNamedBufferParameteri64v
07755 typedef void (APIENTRY PFNGLGETNAMEDBUFFERPOINTERVPROC) (GLuint buffer, GLenum pname, void **params);
07756 GLAPI PFNGLGETNAMEDBUFFERPOINTERVPROC glad_glGetNamedBufferPointerv;
07757 #define glGetNamedBufferPointerv glad_glGetNamedBufferPointerv
07758 typedef void (APIENTRY PFNGLGETNAMEDBUFFERSUBDATAPROC) (GLuint buffer, GLintptr offset, GLsizeiptr
07759 size, void *data);
07759 GLAPI PFNGLGETNAMEDBUFFERSUBDATAPROC glad_glGetNamedBufferSubData;
07760 #define glGetNamedBufferSubData glad_glGetNamedBufferSubData
07761 typedef void (APIENTRY PFNGLCREATEFRAMEBUFFERSPROC) (GLsizei n, GLuint *framebuffers);
07762 GLAPI PFNGLCREATEFRAMEBUFFERSPROC glad_glCreateFramebuffers;
07763 #define glCreateFramebuffers glad_glCreateFramebuffers
07764 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERRENDERBUFFERPROC) (GLuint framebuffer, GLenum attachment,
07765 GLenum renderbuffertarget, GLuint renderbuffer);
07765 GLAPI PFNGLNAMEDFRAMEBUFFERRENDERBUFFERPROC glad_glNamedFramebufferRenderbuffer;
07766 #define glNamedFramebufferRenderbuffer glad_glNamedFramebufferRenderbuffer
07767 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERPARAMETERIPROC) (GLuint framebuffer, GLenum pname, GLint
07768 param);
07768 GLAPI PFNGLNAMEDFRAMEBUFFERPARAMETERIPROC glad_glNamedFramebufferParameteri;
07769 #define glNamedFramebufferParameteri glad_glNamedFramebufferParameteri
07770 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTUREPROC) (GLuint framebuffer, GLenum attachment,
07771 GLuint texture, GLint level);
07771 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREPROC glad_glNamedFramebufferTexture;
07772 #define glNamedFramebufferTexture glad_glNamedFramebufferTexture
07773 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTURELAYERPROC) (GLuint framebuffer, GLenum attachment,
07774 GLuint texture, GLint level, GLint layer);
07774 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURELAYERPROC glad_glNamedFramebufferTextureLayer;
07775 #define glNamedFramebufferTextureLayer glad_glNamedFramebufferTextureLayer
07776 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERDRAWBUFFERPROC) (GLuint framebuffer, GLenum buf);
07777 GLAPI PFNGLNAMEDFRAMEBUFFERDRAWBUFFERPROC glad_glNamedFramebufferDrawBuffer;
07778 #define glNamedFramebufferDrawBuffer glad_glNamedFramebufferDrawBuffer
07779 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERDRAWBUFFERSPROC) (GLuint framebuffer, GLsizei n, const
07780 GLenum *bufs);
07780 GLAPI PFNGLNAMEDFRAMEBUFFERDRAWBUFFERSPROC glad_glNamedFramebufferDrawBuffers;
07781 #define glNamedFramebufferDrawBuffers glad_glNamedFramebufferDrawBuffers
07782 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERREADBUFFERPROC) (GLuint framebuffer, GLenum src);
07783 GLAPI PFNGLNAMEDFRAMEBUFFERREADBUFFERPROC glad_glNamedFramebufferReadBuffer;
07784 #define glNamedFramebufferReadBuffer glad_glNamedFramebufferReadBuffer
07785 typedef void (APIENTRY PFNGLINVALIDATENAMEDFRAMEBUFFERDATAPROC) (GLuint framebuffer, GLsizei
07786 numAttachments, const GLenum *attachments);
07786 GLAPI PFNGLINVALIDATENAMEDFRAMEBUFFERDATAPROC glad_glInvalidateNamedFramebufferData;
07787 #define glInvalidateNamedFramebufferData glad_glInvalidateNamedFramebufferData
07788 typedef void (APIENTRY PFNGLINVALIDATENAMEDFRAMEBUFFERSUBDATAPROC) (GLuint framebuffer, GLsizei
07789 numAttachments, const GLenum *attachments, GLint x, GLint y, GLsizei width, GLsizei height);
07789 GLAPI PFNGLINVALIDATENAMEDFRAMEBUFFERSUBDATAPROC glad_glInvalidateNamedFramebufferSubData;
07790 #define glInvalidateNamedFramebufferSubData glad_glInvalidateNamedFramebufferSubData
07791 typedef void (APIENTRY PFNGLCLEARNAMEDFRAMEBUFFERIVPROC) (GLuint framebuffer, GLenum buffer, GLint
07792 drawbuffer, const GLint *value);
07792 GLAPI PFNGLCLEARNAMEDFRAMEBUFFERIVPROC glad_glClearNamedFramebufferiv;
07793 #define glClearNamedFramebufferiv glad_glClearNamedFramebufferiv
07794 typedef void (APIENTRY PFNGLCLEARNAMEDFRAMEBUFFERUIVPROC) (GLuint framebuffer, GLenum buffer, GLint
07795 drawbuffer, const GLuint *value);
07795 GLAPI PFNGLCLEARNAMEDFRAMEBUFFERUIVPROC glad_glClearNamedFramebufferui;
07796 #define glClearNamedFramebufferui glad_glClearNamedFramebufferui

```

```

07797 typedef void (APIENTRY PFNGLCLEARNAMEFRAMEBUFFERFVPROC) (GLuint framebuffer, GLenum buffer, GLint
drawbuffer, const GLfloat *value);
07798 GLAPI PFNGLCLEARNAMEFRAMEBUFFERFVPROC glad_glClearNamedFramebufferfv;
07799 #define glClearNamedFramebufferfv glad_glClearNamedFramebufferfv
07800 typedef void (APIENTRY PFNGLCLEARNAMEFRAMEBUFFERFIPROC) (GLuint framebuffer, GLenum buffer, GLint
drawbuffer, GLfloat depth, GLint stencil);
07801 GLAPI PFNGLCLEARNAMEFRAMEBUFFERFIPROC glad_glClearNamedFramebufferfi;
07802 #define glClearNamedFramebufferfi glad_glClearNamedFramebufferfi
07803 typedef void (APIENTRY PFNGLBLITNAMEFRAMEBUFFERPROC) (GLuint readFramebuffer, GLuint drawFramebuffer,
GLint srcX0, GLint srcY0, GLint srcX1, GLint srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint
dstY1, GLbitfield mask, GLenum filter);
07804 GLAPI PFNGLBLITNAMEFRAMEBUFFERPROC glad_glBlitNamedFramebuffer;
07805 #define glBlitNamedFramebuffer glad_glBlitNamedFramebuffer
07806 typedef GLenum (APIENTRY PFNGLCHECKNAMEFRAMEBUFFERSTATUSPROC) (GLuint framebuffer, GLenum target);
07807 GLAPI PFNGLCHECKNAMEFRAMEBUFFERSTATUSPROC glad_glCheckNamedFramebufferStatus;
07808 #define glCheckNamedFramebufferStatus glad_glCheckNamedFramebufferStatus
07809 typedef void (APIENTRY PFNGLGETNAMEFRAMEBUFFERPARAMETERIVPROC) (GLuint framebuffer, GLenum pname,
GLint *param);
07810 GLAPI PFNGLGETNAMEFRAMEBUFFERPARAMETERIVPROC glad_glGetNamedFramebufferParameteriv;
07811 #define glGetNamedFramebufferParameteriv glad_glGetNamedFramebufferParameteriv
07812 typedef void (APIENTRY PFNGLGETNAMEFRAMEBUFFERATTACHMENTPARAMETERIVPROC) (GLuint framebuffer, GLenum
attachment, GLenum pname, GLint *params);
07813 GLAPI PFNGLGETNAMEFRAMEBUFFERATTACHMENTPARAMETERIVPROC
glad_glGetNamedFramebufferAttachmentParameteriv;
07814 #define glGetNamedFramebufferAttachmentParameteriv glad_glGetNamedFramebufferAttachmentParameteriv
07815 typedef void (APIENTRY PFNGLCREATERENDERBUFFERSPROC) (GLsizei n, GLuint *renderbuffers);
07816 GLAPI PFNGLCREATERENDERBUFFERSPROC glad_glCreateRenderbuffers;
07817 #define glCreateRenderbuffers glad_glCreateRenderbuffers
07818 typedef void (APIENTRY PFNGLNAMEDRENDERBUFFERSTORAGEPROC) (GLuint renderbuffer, GLenum internalformat,
GLsizei width, GLsizei height);
07819 GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEPROC glad_glNamedRenderbufferStorage;
07820 #define glNamedRenderbufferStorage glad_glNamedRenderbufferStorage
07821 typedef void (APIENTRY PFNGLNAMEDRENDERBUFFERSTAGEMULTISAMPLEPROC) (GLuint renderbuffer, GLsizei
samples, GLenum internalformat, GLsizei width, GLsizei height);
07822 GLAPI PFNGLNAMEDRENDERBUFFERSTAGEMULTISAMPLEPROC glad_glNamedRenderbufferStorageMultisample;
07823 #define glNamedRenderbufferStorageMultisample glad_glNamedRenderbufferStorageMultisample
07824 typedef void (APIENTRY PFNGLGETNAMEDENDERBUFFERPARAMETERIVPROC) (GLuint renderbuffer, GLenum pname,
GLint *params);
07825 GLAPI PFNGLGETNAMEDENDERBUFFERPARAMETERIVPROC glad_glGetNamedRenderbufferParameteriv;
07826 #define glGetNamedRenderbufferParameteriv glad_glGetNamedRenderbufferParameteriv
07827 typedef void (APIENTRY PFNGLCREATETEXTURESPROC) (GLenum target, GLsizei n, GLuint *textures);
07828 GLAPI PFNGLCREATETEXTURESPROC glad_glCreateTextures;
07829 #define glCreateTextures glad_glCreateTextures
07830 typedef void (APIENTRY PFNGLTEXTUREBUFFERPROC) (GLuint texture, GLenum internalformat, GLuint buffer);
07831 GLAPI PFNGLTEXTUREBUFFERPROC glad_glTextureBuffer;
07832 #define glTextureBuffer glad_glTextureBuffer
07833 typedef void (APIENTRY PFNGLTEXTUREBUFFERRANGEPROC) (GLuint texture, GLenum internalformat, GLuint
buffer, GLintptr offset, GLsizeiptr size);
07834 GLAPI PFNGLTEXTUREBUFFERRANGEPROC glad_glTextureBufferRange;
07835 #define glTextureBufferRange glad_glTextureBufferRange
07836 typedef void (APIENTRY PFNGLTEXTURESTORAGE1DPROC) (GLuint texture, GLsizei levels, GLenum
internalformat, GLsizei width);
07837 GLAPI PFNGLTEXTURESTORAGE1DPROC glad_glTextureStorage1D;
07838 #define glTextureStorage1D glad_glTextureStorage1D
07839 typedef void (APIENTRY PFNGLTEXTURESTORAGE2DPROC) (GLuint texture, GLsizei levels, GLenum
internalformat, GLsizei width, GLsizei height);
07840 GLAPI PFNGLTEXTURESTORAGE2DPROC glad_glTextureStorage2D;
07841 #define glTextureStorage2D glad_glTextureStorage2D
07842 typedef void (APIENTRY PFNGLTEXTURESTORAGE3DPROC) (GLuint texture, GLsizei levels, GLenum
internalformat, GLsizei width, GLsizei height, GLsizei depth);
07843 GLAPI PFNGLTEXTURESTORAGE3DPROC glad_glTextureStorage3D;
07844 #define glTextureStorage3D glad_glTextureStorage3D
07845 typedef void (APIENTRY PFNGLTEXTURESTORAGE2DMULTISAMPLEPROC) (GLuint texture, GLsizei samples, GLenum
internalformat, GLsizei width, GLsizei height, GLboolean fixedsamplelocations);
07846 GLAPI PFNGLTEXTURESTORAGE2DMULTISAMPLEPROC glad_glTextureStorage2DMultisample;
07847 #define glTextureStorage2DMultisample glad_glTextureStorage2DMultisample
07848 typedef void (APIENTRY PFNGLTEXTURESTORAGE3DMULTISAMPLEPROC) (GLuint texture, GLsizei samples, GLenum
internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedsamplelocations);
07849 GLAPI PFNGLTEXTURESTORAGE3DMULTISAMPLEPROC glad_glTextureStorage3DMultisample;
07850 #define glTextureStorage3DMultisample glad_glTextureStorage3DMultisample
07851 typedef void (APIENTRY PFNGLTEXTURESUBIMAGE1DPROC) (GLuint texture, GLint level, GLint xoffset,
GLsizei width, GLenum format, GLenum type, const void *pixels);
07852 GLAPI PFNGLTEXTURESUBIMAGE1DPROC glad_glTextureSubImage1D;
07853 #define glTextureSubImage1D glad_glTextureSubImage1D
07854 typedef void (APIENTRY PFNGLTEXTURESUBIMAGE2DPROC) (GLuint texture, GLint level, GLint xoffset, GLint
yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *pixels);
07855 GLAPI PFNGLTEXTURESUBIMAGE2DPROC glad_glTextureSubImage2D;
07856 #define glTextureSubImage2D glad_glTextureSubImage2D
07857 typedef void (APIENTRY PFNGLTEXTURESUBIMAGE3DPROC) (GLuint texture, GLint level, GLint xoffset, GLint
yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type,
const void *pixels);
07858 GLAPI PFNGLTEXTURESUBIMAGE3DPROC glad_glTextureSubImage3D;
07859 #define glTextureSubImage3D glad_glTextureSubImage3D
07860 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTURESUBIMAGE1DPROC) (GLuint texture, GLint level, GLint
xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *data);
07861 GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE1DPROC glad_glCompressedTextureSubImage1D;
07862 #define glCompressedTextureSubImage1D glad_glCompressedTextureSubImage1D

```

```

07863 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTURESUBIMAGE2DPROC)(GLuint texture, GLint level, GLint
xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void
*data);
07864 GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE2DPROC glad_glCompressedTextureSubImage2D;
07865 #define glCompressedTextureSubImage2D glad_glCompressedTextureSubImage2D
07866 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTURESUBIMAGE3DPROC)(GLuint texture, GLint level, GLint
xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format,
GLsizei imageSize, const void *data);
07867 GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE3DPROC glad_glCompressedTextureSubImage3D;
07868 #define glCompressedTextureSubImage3D glad_glCompressedTextureSubImage3D
07869 typedef void (APIENTRY PFNGLCOPYTEXTURESUBIMAGE1DPROC)(GLuint texture, GLint level, GLint xoffset,
GLint x, GLint y, GLsizei width);
07870 GLAPI PFNGLCOPYTEXTURESUBIMAGE1DPROC glad_glCopyTextureSubImage1D;
07871 #define glCopyTextureSubImage1D glad_glCopyTextureSubImage1D
07872 typedef void (APIENTRY PFNGLCOPYTEXTURESUBIMAGE2DPROC)(GLuint texture, GLint level, GLint xoffset,
GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
07873 GLAPI PFNGLCOPYTEXTURESUBIMAGE2DPROC glad_glCopyTextureSubImage2D;
07874 #define glCopyTextureSubImage2D glad_glCopyTextureSubImage2D
07875 typedef void (APIENTRY PFNGLCOPYTEXTURESUBIMAGE3DPROC)(GLuint texture, GLint level, GLint xoffset,
GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
07876 GLAPI PFNGLCOPYTEXTURESUBIMAGE3DPROC glad_glCopyTextureSubImage3D;
07877 #define glCopyTextureSubImage3D glad_glCopyTextureSubImage3D
07878 typedef void (APIENTRY PFNGLTEXTUREPARAMETERFPROC)(GLuint texture, GLenum pname, GLfloat param);
07879 GLAPI PFNGLTEXTUREPARAMETERFPROC glad_glTextureParameterf;
07880 #define glTextureParameterf glad_glTextureParameterf
07881 typedef void (APIENTRY PFNGLTEXTUREPARAMETERFVPROC)(GLuint texture, GLenum pname, const GLfloat
*param);
07882 GLAPI PFNGLTEXTUREPARAMETERFVPROC glad_glTextureParameterfv;
07883 #define glTextureParameterfv glad_glTextureParameterfv
07884 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIPROC)(GLuint texture, GLenum pname, GLint param);
07885 GLAPI PFNGLTEXTUREPARAMETERIPROC glad_glTextureParameteri;
07886 #define glTextureParameteri glad_glTextureParameteri
07887 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIIVPROC)(GLuint texture, GLenum pname, const GLint
*params);
07888 GLAPI PFNGLTEXTUREPARAMETERIIVPROC glad_glTextureParameteriiv;
07889 #define glTextureParameteriiv glad_glTextureParameteriiv
07890 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIUIVPROC)(GLuint texture, GLenum pname, const GLuint
*params);
07891 GLAPI PFNGLTEXTUREPARAMETERIUIVPROC glad_glTextureParameteriui;
07892 #define glTextureParameteriui glad_glTextureParameteriui
07893 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIVPROC)(GLuint texture, GLenum pname, const GLint
*param);
07894 GLAPI PFNGLTEXTUREPARAMETERIVPROC glad_glTextureParameteriv;
07895 #define glTextureParameteriv glad_glTextureParameteriv
07896 typedef void (APIENTRY PFNGLGENERATETEXTUREMIPMAPPROC)(GLuint texture);
07897 GLAPI PFNGLGENERATETEXTUREMIPMAPPROC glad_glGenerateTextureMipmap;
07898 #define glGenerateTextureMipmap glad_glGenerateTextureMipmap
07899 typedef void (APIENTRY PFNGLBINDTEXTUREUNITPROC)(GLuint unit, GLuint texture);
07900 GLAPI PFNGLBINDTEXTUREUNITPROC glad_glBindTextureUnit;
07901 #define glBindTextureUnit glad_glBindTextureUnit
07902 typedef void (APIENTRY PFNGLGETTEXTUREIMAGEPROC)(GLuint texture, GLint level, GLenum format, GLenum
type, GLsizei bufSize, void *pixels);
07903 GLAPI PFNGLGETTEXTUREIMAGEPROC glad_glGetTextureImage;
07904 #define glGetTextureImage glad_glGetTextureImage
07905 typedef void (APIENTRY PFNGLGETCOMPRESSEDTEXTUREIMAGEPROC)(GLuint texture, GLint level, GLsizei
bufSize, void *pixels);
07906 GLAPI PFNGLGETCOMPRESSEDTEXTUREIMAGEPROC glad_glGetCompressedTextureImage;
07907 #define glGetCompressedTextureImage glad_glGetCompressedTextureImage
07908 typedef void (APIENTRY PFNGLGETTEXTURELEVELPARAMETERFVPROC)(GLuint texture, GLint level, GLenum
pname, GLfloat *params);
07909 GLAPI PFNGLGETTEXTURELEVELPARAMETERFVPROC glad_glGetTextureLevelParameterfv;
07910 #define glGetTextureLevelParameterfv glad_glGetTextureLevelParameterfv
07911 typedef void (APIENTRY PFNGLGETTEXTURELEVELPARAMETERIVPROC)(GLuint texture, GLint level, GLenum
pname, GLint *params);
07912 GLAPI PFNGLGETTEXTURELEVELPARAMETERIVPROC glad_glGetTextureLevelParameteriv;
07913 #define glGetTextureLevelParameteriv glad_glGetTextureLevelParameteriv
07914 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERFVPROC)(GLuint texture, GLenum pname, GLfloat
*params);
07915 GLAPI PFNGLGETTEXTUREPARAMETERFVPROC glad_glGetTextureParameterfv;
07916 #define glGetTextureParameterfv glad_glGetTextureParameterfv
07917 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERIIVPROC)(GLuint texture, GLenum pname, GLint *params);
07918 GLAPI PFNGLGETTEXTUREPARAMETERIIVPROC glad_glGetTextureParameteriiv;
07919 #define glGetTextureParameteriiv glad_glGetTextureParameteriiv
07920 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERIUIVPROC)(GLuint texture, GLenum pname, GLuint
*params);
07921 GLAPI PFNGLGETTEXTUREPARAMETERIUIVPROC glad_glGetTextureParameteriui;
07922 #define glGetTextureParameteriui glad_glGetTextureParameteriui
07923 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERIVPROC)(GLuint texture, GLenum pname, GLint *params);
07924 GLAPI PFNGLGETTEXTUREPARAMETERIVPROC glad_glGetTextureParameteriv;
07925 #define glGetTextureParameteriv glad_glGetTextureParameteriv
07926 typedef void (APIENTRY PFNGLCREATEVERTEXARRAYSPROC)(GLsizei n, GLuint *arrays);
07927 GLAPI PFNGLCREATEVERTEXARRAYSPROC glad_glCreateVertexArrays;
07928 #define glCreateVertexArrays glad_glCreateVertexArrays
07929 typedef void (APIENTRY PFNGLDISABLEVERTEXARRAYATTRIBPROC)(GLuint vaobj, GLuint index);
07930 GLAPI PFNGLDISABLEVERTEXARRAYATTRIBPROC glad_glDisableVertexArrayAttrib;
07931 #define glDisableVertexArrayAttrib glad_glDisableVertexArrayAttrib
07932 typedef void (APIENTRY PFNGLENABLEVERTEXARRAYATTRIBPROC)(GLuint vaobj, GLuint index);

```



```

07933 GLAPI PFNGLENABLEVERTEXARRAYATTRIBPROC glad_glEnableVertexArrayAttrib;
07934 #define glEnableVertexArrayAttrib glad_glEnableVertexArrayAttrib
07935 typedef void (APIENTRY PFNGLVERTEXARRAYELEMENTBUFFERPROC) (GLuint vaobj, GLuint buffer);
07936 GLAPI PFNGLVERTEXARRAYELEMENTBUFFERPROC glad_glVertexArrayElementBuffer;
07937 #define glVertexArrayElementBuffer glad_glVertexArrayElementBuffer
07938 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXBUFFERPROC) (GLuint vaobj, GLuint bindingindex, GLuint
buffer, GLintptr offset, GLsizei stride);
07939 GLAPI PFNGLVERTEXARRAYVERTEXBUFFERPROC glad_glVertexArrayVertexBuffer;
07940 #define glVertexArrayVertexBuffer glad_glVertexArrayVertexBuffer
07941 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXBUFFERSPROC) (GLuint vaobj, GLuint first, GLsizei count,
const GLuint *buffers, const GLintptr *offsets, const GLsizei *strides);
07942 GLAPI PFNGLVERTEXARRAYVERTEXBUFFERSPROC glad_glVertexArrayVertexBuffers;
07943 #define glVertexArrayVertexBuffers glad_glVertexArrayVertexBuffers
07944 typedef void (APIENTRY PFNGLVERTEXARRAYATTRIBBINDINGPROC) (GLuint vaobj, GLuint attribindex, GLuint
bindingindex);
07945 GLAPI PFNGLVERTEXARRAYATTRIBBINDINGPROC glad_glVertexArrayAttribBinding;
07946 #define glVertexArrayAttribBinding glad_glVertexArrayAttribBinding
07947 typedef void (APIENTRY PFNGLVERTEXARRAYATTRIBIFORMATPROC) (GLuint vaobj, GLuint attribindex, GLint
size, GLenum type, GLboolean normalized, GLuint relativeoffset);
07948 GLAPI PFNGLVERTEXARRAYATTRIBIFORMATPROC glad_glVertexArrayAttribFormat;
07949 #define glVertexArrayAttribFormat glad_glVertexArrayAttribFormat
07950 typedef void (APIENTRY PFNGLVERTEXARRAYATTRIBIFORMATPROC) (GLuint vaobj, GLuint attribindex, GLint
size, GLenum type, GLuint relativeoffset);
07951 GLAPI PFNGLVERTEXARRAYATTRIBIFORMATPROC glad_glVertexArrayAttribIFormat;
07952 #define glVertexArrayAttribIFormat glad_glVertexArrayAttribIFormat
07953 typedef void (APIENTRY PFNGLVERTEXARRAYATTRIBIFORMATPROC) (GLuint vaobj, GLuint attribindex, GLint
size, GLenum type, GLuint relativeoffset);
07954 GLAPI PFNGLVERTEXARRAYATTRIBIFORMATPROC glad_glVertexArrayAttribLFormat;
07955 #define glVertexArrayAttribLFormat glad_glVertexArrayAttribLFormat
07956 typedef void (APIENTRY PFNGLVERTEXARRAYBINDINGDIVISORPROC) (GLuint vaobj, GLuint bindingindex, GLuint
divisor);
07957 GLAPI PFNGLVERTEXARRAYBINDINGDIVISORPROC glad_glVertexArrayBindingDivisor;
07958 #define glVertexArrayBindingDivisor glad_glVertexArrayBindingDivisor
07959 typedef void (APIENTRY PFNGLGETVERTEXARRAYIVPROC) (GLuint vaobj, GLenum pname, GLint *param);
07960 GLAPI PFNGLGETVERTEXARRAYIVPROC glad_glGetVertexArrayiv;
07961 #define glGetVertexArrayiv glad_glGetVertexArrayiv
07962 typedef void (APIENTRY PFNGLGETVERTEXARRAYINDEXEDIVPROC) (GLuint vaobj, GLuint index, GLenum pname,
GLint *param);
07963 GLAPI PFNGLGETVERTEXARRAYINDEXEDIVPROC glad_glGetVertexArrayIndexediv;
07964 #define glGetVertexArrayIndexediv glad_glGetVertexArrayIndexediv
07965 typedef void (APIENTRY PFNGLGETVERTEXARRAYINDEXED64IVPROC) (GLuint vaobj, GLuint index, GLenum pname,
GLint64 *param);
07966 GLAPI PFNGLGETVERTEXARRAYINDEXED64IVPROC glad_glGetVertexArrayIndexed64iv;
07967 #define glGetVertexArrayIndexed64iv glad_glGetVertexArrayIndexed64iv
07968 typedef void (APIENTRY PFNGLCREATESAMPLERSPROC) (GLsizei n, GLuint *samplers);
07969 GLAPI PFNGLCREATESAMPLERSPROC glad_glCreateSamplers;
07970 #define glCreateSamplers glad_glCreateSamplers
07971 typedef void (APIENTRY PFNGLCREATEPROGRAMPIPELINESPROC) (GLsizei n, GLuint *pipelines);
07972 GLAPI PFNGLCREATEPROGRAMPIPELINESPROC glad_glCreateProgramPipelines;
07973 #define glCreateProgramPipelines glad_glCreateProgramPipelines
07974 typedef void (APIENTRY PFNGLCREATEQUERIESPROC) (GLenum target, GLsizei n, GLuint *ids);
07975 GLAPI PFNGLCREATEQUERIESPROC glad_glCreateQueries;
07976 #define glCreateQueries glad_glCreateQueries
07977 typedef void (APIENTRY PFNGLGETQUERYBUFFEROBJECTI64VPROC) (GLuint id, GLuint buffer, GLenum pname,
GLintptr offset);
07978 GLAPI PFNGLGETQUERYBUFFEROBJECTI64VPROC glad_glGetQueryBufferObjecti64v;
07979 #define glGetQueryBufferObjecti64v glad_glGetQueryBufferObjecti64v
07980 typedef void (APIENTRY PFNGLGETQUERYBUFFEROBJECTIVPROC) (GLuint id, GLuint buffer, GLenum pname,
GLintptr offset);
07981 GLAPI PFNGLGETQUERYBUFFEROBJECTIVPROC glad_glGetQueryBufferObjectiv;
07982 #define glGetQueryBufferObjectiv glad_glGetQueryBufferObjectiv
07983 typedef void (APIENTRY PFNGLGETQUERYBUFFEROBJECTUI64VPROC) (GLuint id, GLuint buffer, GLenum pname,
GLintptr offset);
07984 GLAPI PFNGLGETQUERYBUFFEROBJECTUI64VPROC glad_glGetQueryBufferObjectui64v;
07985 #define glGetQueryBufferObjectui64v glad_glGetQueryBufferObjectui64v
07986 typedef void (APIENTRY PFNGLGETQUERYBUFFEROBJECTUIVPROC) (GLuint id, GLuint buffer, GLenum pname,
GLintptr offset);
07987 GLAPI PFNGLGETQUERYBUFFEROBJECTUIVPROC glad_glGetQueryBufferObjectuiv;
07988 #define glGetQueryBufferObjectuiv glad_glGetQueryBufferObjectuiv
07989 #endif
07990 #ifndef GL_ARB_draw_buffers
07991 #define GL_ARB_draw_buffers 1
07992 GLAPI int GLAD_GL_ARB_draw_buffers;
07993 typedef void (APIENTRY PFNGLDRAWBUFFERSARBPROC) (GLsizei n, const GLenum *bufs);
07994 GLAPI PFNGLDRAWBUFFERSARBPROC glad_glDrawBuffersARB;
07995 #define glDrawBuffersARB glad_glDrawBuffersARB
07996 #endif
07997 #ifndef GL_ARB_draw_buffers_blend
07998 #define GL_ARB_draw_buffers_blend 1
07999 GLAPI int GLAD_GL_ARB_draw_buffers_blend;
08000 typedef void (APIENTRY PFNGLBLENDEQUATIONIARBPROC) (GLuint buf, GLenum mode);
08001 GLAPI PFNGLBLENDEQUATIONIARBPROC glad_glBlendEquationiARB;
08002 #define glBlendEquationiARB glad_glBlendEquationiARB
08003 typedef void (APIENTRY PFNGLBLENDEQUATIONSEPARATEIARBPROC) (GLuint buf, GLenum modeRGB, GLenum
modeAlpha);
08004 GLAPI PFNGLBLENDEQUATIONSEPARATEIARBPROC glad_glBlendEquationSeparateiARB;
08005 #define glBlendEquationSeparateiARB glad_glBlendEquationSeparateiARB

```

```

08006 typedef void (APIENTRY PFNGLBLENDFUNCARBPROC)(GLuint buf, GLenum src, GLenum dst);
08007 GLAPI PFNGLBLENDFUNCARBPROC glad_glBlendFuncARB;
08008 #define glBlendFuncARB glad_glBlendFuncARB
08009 typedef void (APIENTRY PFNGLBLENDFUNCSEPARATEARBPROC)(GLuint buf, GLenum srcRGB, GLenum dstRGB,
    GLenum srcAlpha, GLenum dstAlpha);
08010 GLAPI PFNGLBLENDFUNCSEPARATEARBPROC glad_glBlendFuncSeparateARB;
08011 #define glBlendFuncSeparateARB glad_glBlendFuncSeparateARB
08012 #endif
08013 #ifndef GL_ARB_draw_elements_base_vertex
08014 #define GL_ARB_draw_elements_base_vertex 1
08015 GLAPI int GLAD_GL_ARB_draw_elements_base_vertex;
08016 #endif
08017 #ifndef GL_ARB_draw_indirect
08018 #define GL_ARB_draw_indirect 1
08019 GLAPI int GLAD_GL_ARB_draw_indirect;
08020 #endif
08021 #ifndef GL_ARB_draw_instanced
08022 #define GL_ARB_draw_instanced 1
08023 GLAPI int GLAD_GL_ARB_draw_instanced;
08024 typedef void (APIENTRY PFNGLDRAWARRAYSINSTANCEDARBPROC)(GLenum mode, GLint first, GLsizei count,
    GLsizei primcount);
08025 GLAPI PFNGLDRAWARRAYSINSTANCEDARBPROC glad_glDrawArraysInstancedARB;
08026 #define glDrawArraysInstancedARB glad_glDrawArraysInstancedARB
08027 typedef void (APIENTRY PFNGLDRAWELEMENTSINSTANCEDARBPROC)(GLenum mode, GLsizei count, GLenum type,
    const void *indices, GLsizei primcount);
08028 GLAPI PFNGLDRAWELEMENTSINSTANCEDARBPROC glad_glDrawElementsInstancedARB;
08029 #define glDrawElementsInstancedARB glad_glDrawElementsInstancedARB
08030 #endif
08031 #ifndef GL_ARB_enhanced_layouts
08032 #define GL_ARB_enhanced_layouts 1
08033 GLAPI int GLAD_GL_ARB_enhanced_layouts;
08034 #endif
08035 #ifndef GL_ARB_explicit_attrib_location
08036 #define GL_ARB_explicit_attrib_location 1
08037 GLAPI int GLAD_GL_ARB_explicit_attrib_location;
08038 #endif
08039 #ifndef GL_ARB_explicit_uniform_location
08040 #define GL_ARB_explicit_uniform_location 1
08041 GLAPI int GLAD_GL_ARB_explicit_uniform_location;
08042 #endif
08043 #ifndef GL_ARB_fragment_coord_conventions
08044 #define GL_ARB_fragment_coord_conventions 1
08045 GLAPI int GLAD_GL_ARB_fragment_coord_conventions;
08046 #endif
08047 #ifndef GL_ARB_fragment_layer_viewport
08048 #define GL_ARB_fragment_layer_viewport 1
08049 GLAPI int GLAD_GL_ARB_fragment_layer_viewport;
08050 #endif
08051 #ifndef GL_ARB_fragment_program
08052 #define GL_ARB_fragment_program 1
08053 GLAPI int GLAD_GL_ARB_fragment_program;
08054 typedef void (APIENTRY PFNGLPROGRAMSTRINGARBPROC)(GLenum target, GLenum format, GLsizei len, const
    void *string);
08055 GLAPI PFNGLPROGRAMSTRINGARBPROC glad_glProgramStringARB;
08056 #define glProgramStringARB glad_glProgramStringARB
08057 typedef void (APIENTRY PFNGLBINDPROGRAMARBPROC)(GLenum target, GLuint program);
08058 GLAPI PFNGLBINDPROGRAMARBPROC glad_glBindProgramARB;
08059 #define glBindProgramARB glad_glBindProgramARB
08060 typedef void (APIENTRY PFNGLDELETEPROGRAMSARBPROC)(GLsizei n, const GLuint *programs);
08061 GLAPI PFNGLDELETEPROGRAMSARBPROC glad_glDeleteProgramsARB;
08062 #define glDeleteProgramsARB glad_glDeleteProgramsARB
08063 typedef void (APIENTRY PFNGLGENPROGRAMSARBPROC)(GLsizei n, GLuint *programs);
08064 GLAPI PFNGLGENPROGRAMSARBPROC glad_glGenProgramsARB;
08065 #define glGenProgramsARB glad_glGenProgramsARB
08066 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETER4DARBPROC)(GLenum target, GLuint index, GLdouble x,
    GLdouble y, GLdouble z, GLdouble w);
08067 GLAPI PFNGLPROGRAMENVPARAMETER4DARBPROC glad_glProgramEnvParameter4dARB;
08068 #define glProgramEnvParameter4dARB glad_glProgramEnvParameter4dARB
08069 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETER4DVARBPROC)(GLenum target, GLuint index, const
    GLdouble *params);
08070 GLAPI PFNGLPROGRAMENVPARAMETER4DVARBPROC glad_glProgramEnvParameter4dvARB;
08071 #define glProgramEnvParameter4dvARB glad_glProgramEnvParameter4dvARB
08072 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETER4FARBPROC)(GLenum target, GLuint index, GLfloat x,
    GLfloat y, GLfloat z, GLfloat w);
08073 GLAPI PFNGLPROGRAMENVPARAMETER4FARBPROC glad_glProgramEnvParameter4fARB;
08074 #define glProgramEnvParameter4fARB glad_glProgramEnvParameter4fARB
08075 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETER4FVARBPROC)(GLenum target, GLuint index, const
    GLfloat *params);
08076 GLAPI PFNGLPROGRAMENVPARAMETER4FVARBPROC glad_glProgramEnvParameter4fvARB;
08077 #define glProgramEnvParameter4fvARB glad_glProgramEnvParameter4fvARB
08078 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETER4DARBPROC)(GLenum target, GLuint index, GLdouble x,
    GLdouble y, GLdouble z, GLdouble w);
08079 GLAPI PFNGLPROGRAMLOCALPARAMETER4DARBPROC glad_glProgramLocalParameter4dARB;
08080 #define glProgramLocalParameter4dARB glad_glProgramLocalParameter4dARB
08081 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETER4DVARBPROC)(GLenum target, GLuint index, const
    GLdouble *params);
08082 GLAPI PFNGLPROGRAMLOCALPARAMETER4DVARBPROC glad_glProgramLocalParameter4dvARB;

```

```

08083 #define glProgramLocalParameter4dvARB glad_glProgramLocalParameter4dvARB
08084 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETER4FARBPROC) (GLenum target, GLuint index, GLfloat x,
08085   GLfloat y, GLfloat z, GLfloat w);
08085 GLAPI PFNGLPROGRAMLOCALPARAMETER4FARBPROC glad_glProgramLocalParameter4fARB;
08086 #define glProgramLocalParameter4fARB glad_glProgramLocalParameter4fARB
08087 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETER4FVARBPROC) (GLenum target, GLuint index, const
08088   GLfloat *params);
08088 GLAPI PFNGLPROGRAMLOCALPARAMETER4FVARBPROC glad_glProgramLocalParameter4fvARB;
08089 #define glProgramLocalParameter4fvARB glad_glProgramLocalParameter4fvARB
08090 typedef void (APIENTRY PFNGLGETPROGRAMENVPARAMETERDVARBPROC) (GLenum target, GLuint index, GLdouble
08091   *params);
08091 GLAPI PFNGLGETPROGRAMENVPARAMETERDVARBPROC glad_glGetProgramEnvParameterdvARB;
08092 #define glGetProgramEnvParameterdvARB glad_glGetProgramEnvParameterdvARB
08093 typedef void (APIENTRY PFNGLGETPROGRAMENVPARAMETERFVARBPROC) (GLenum target, GLuint index, GLfloat
08094   *params);
08094 GLAPI PFNGLGETPROGRAMENVPARAMETERFVARBPROC glad_glGetProgramEnvParameterfvARB;
08095 #define glGetProgramEnvParameterfvARB glad_glGetProgramEnvParameterfvARB
08096 typedef void (APIENTRY PFNGLGETPROGRAMLOCALPARAMETERDVARBPROC) (GLenum target, GLuint index, GLdouble
08097   *params);
08097 GLAPI PFNGLGETPROGRAMLOCALPARAMETERDVARBPROC glad_glGetProgramLocalParameterdvARB;
08098 #define glGetProgramLocalParameterdvARB glad_glGetProgramLocalParameterdvARB
08099 typedef void (APIENTRY PFNGLGETPROGRAMLOCALPARAMETERFVARBPROC) (GLenum target, GLuint index, GLfloat
08100   *params);
08100 GLAPI PFNGLGETPROGRAMLOCALPARAMETERFVARBPROC glad_glGetProgramLocalParameterfvARB;
08101 #define glGetProgramLocalParameterfvARB glad_glGetProgramLocalParameterfvARB
08102 typedef void (APIENTRY PFNGLGETPROGRAMIVARBPROC) (GLenum target, GLenum pname, GLint *params);
08103 GLAPI PFNGLGETPROGRAMIVARBPROC glad_glGetProgramivARB;
08104 #define glGetProgramivARB glad_glGetProgramivARB
08105 typedef void (APIENTRY PFNGLGETPROGRAMSTRINGARBPROC) (GLenum target, GLenum pname, void *string);
08106 GLAPI PFNGLGETPROGRAMSTRINGARBPROC glad_glGetProgramStringARB;
08107 #define glGetProgramStringARB glad_glGetProgramStringARB
08108 typedef GLboolean (APIENTRY PFNGLISPROGRAMARBPROC) (GLuint program);
08109 GLAPI PFNGLISPROGRAMARBPROC glad_glIsProgramARB;
08110 #define glIsProgramARB glad_glIsProgramARB
08111 #endif
08112 #ifndef GL_ARB_fragment_program_shadow
08113 #define GL_ARB_fragment_program_shadow 1
08114 GLAPI int GLAD_GL_ARB_fragment_program_shadow;
08115 #endif
08116 #ifndef GL_ARB_fragment_shader
08117 #define GL_ARB_fragment_shader 1
08118 GLAPI int GLAD_GL_ARB_fragment_shader;
08119 #endif
08120 #ifndef GL_ARB_fragment_shader_interlock
08121 #define GL_ARB_fragment_shader_interlock 1
08122 GLAPI int GLAD_GL_ARB_fragment_shader_interlock;
08123 #endif
08124 #ifndef GL_ARB_framebuffer_no_attachments
08125 #define GL_ARB_framebuffer_no_attachments 1
08126 GLAPI int GLAD_GL_ARB_framebuffer_no_attachments;
08127 typedef void (APIENTRY PFNGLFRAMEBUFFERPARAMETERIPROC) (GLenum target, GLenum pname, GLint param);
08128 GLAPI PFNGLFRAMEBUFFERPARAMETERIPROC glad_glFramebufferParameteri;
08129 #define glFramebufferParameteri glad_glFramebufferParameteri
08130 typedef void (APIENTRY PFNGLGETFRAMEBUFFERPARAMETERIVPROC) (GLenum target, GLenum pname, GLint
08131   *params);
08131 GLAPI PFNGLGETFRAMEBUFFERPARAMETERIVPROC glad_glGetFramebufferParameteriv;
08132 #define glGetFramebufferParameteriv glad_glGetFramebufferParameteriv
08133 #endif
08134 #ifndef GL_ARB_framebuffer_object
08135 #define GL_ARB_framebuffer_object 1
08136 GLAPI int GLAD_GL_ARB_framebuffer_object;
08137 #endif
08138 #ifndef GL_ARB_framebuffer_sRGB
08139 #define GL_ARB_framebuffer_sRGB 1
08140 GLAPI int GLAD_GL_ARB_framebuffer_sRGB;
08141 #endif
08142 #ifndef GL_ARB_geometry_shader4
08143 #define GL_ARB_geometry_shader4 1
08144 GLAPI int GLAD_GL_ARB_geometry_shader4;
08145 typedef void (APIENTRY PFNGLPROGRAMPARAMETERIARBPROC) (GLuint program, GLenum pname, GLint value);
08146 GLAPI PFNGLPROGRAMPARAMETERIARBPROC glad_glProgramParameteriARB;
08147 #define glProgramParameteriARB glad_glProgramParameteriARB
08148 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTUREARBPROC) (GLenum target, GLenum attachment, GLuint
08149   texture, GLint level);
08149 GLAPI PFNGLFRAMEBUFFERTEXTUREARBPROC glad_glFramebufferTextureARB;
08150 #define glFramebufferTextureARB glad_glFramebufferTextureARB
08151 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURELAYERARBPROC) (GLenum target, GLenum attachment, GLuint
08152   texture, GLint level, GLint layer);
08152 GLAPI PFNGLFRAMEBUFFERTEXTURELAYERARBPROC glad_glFramebufferTextureLayerARB;
08153 #define glFramebufferTextureLayerARB glad_glFramebufferTextureLayerARB
08154 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTUREFACEARBPROC) (GLenum target, GLenum attachment, GLuint
08155   texture, GLint level, GLenum face);
08155 GLAPI PFNGLFRAMEBUFFERTEXTUREFACEARBPROC glad_glFramebufferTextureFaceARB;
08156 #define glFramebufferTextureFaceARB glad_glFramebufferTextureFaceARB
08157 #endif
08158 #ifndef GL_ARB_get_program_binary
08159 #define GL_ARB_get_program_binary 1

```



```

08160 GLAPI int GLAD_GL_ARB_get_program_binary;
08161 #endif
08162 #ifndef GL_ARB_get_texture_sub_image
08163 #define GL_ARB_get_texture_sub_image 1
08164 GLAPI int GLAD_GL_ARB_get_texture_sub_image;
08165 typedef void (APIENTRY PFNGLGETTEXTURESUBIMAGEPROC) (GLuint texture, GLint level, GLint xoffset, GLint
yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type,
GLsizei bufSize, void *pixels);
08166 GLAPI PFNGLGETTEXTURESUBIMAGEPROC glad_glGetTextureSubImage;
08167 #define glGetTextureSubImage glad_glGetTextureSubImage
08168 typedef void (APIENTRY PFNGLGETCOMPRESSEDTEXTURESUBIMAGEPROC) (GLuint texture, GLint level, GLint
xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLsizei bufSize,
void *pixels);
08169 GLAPI PFNGLGETCOMPRESSEDTEXTURESUBIMAGEPROC glad_glGetCompressedTextureSubImage;
08170 #define glGetCompressedTextureSubImage glad_glGetCompressedTextureSubImage
08171 #endif
08172 #ifndef GL_ARB_gl_spirv
08173 #define GL_ARB_gl_spirv 1
08174 GLAPI int GLAD_GL_ARB_gl_spirv;
08175 typedef void (APIENTRY PFNGLSPECIALIZESHADERARBPROC) (GLuint shader, const GLchar *pEntryPoint, GLuint
numSpecializationConstants, const GLuint *pConstantIndex, const GLuint *pConstantValue);
08176 GLAPI PFNGLSPECIALIZESHADERARBPROC glad_glSpecializeShaderARB;
08177 #define glSpecializeShaderARB glad_glSpecializeShaderARB
08178 #endif
08179 #ifndef GL_ARB_gpu_shader5
08180 #define GL_ARB_gpu_shader5 1
08181 GLAPI int GLAD_GL_ARB_gpu_shader5;
08182 #endif
08183 #ifndef GL_ARB_gpu_shader_fp64
08184 #define GL_ARB_gpu_shader_fp64 1
08185 GLAPI int GLAD_GL_ARB_gpu_shader_fp64;
08186 #endif
08187 #ifndef GL_ARB_gpu_shader_int64
08188 #define GL_ARB_gpu_shader_int64 1
08189 GLAPI int GLAD_GL_ARB_gpu_shader_int64;
08190 typedef void (APIENTRY PFNGLUNIFORM1I64ARBPROC) (GLint location, GLint64 x);
08191 GLAPI PFNGLUNIFORM1I64ARBPROC glad_glUniform1i64ARB;
08192 #define glUniform1i64ARB glad_glUniform1i64ARB
08193 typedef void (APIENTRY PFNGLUNIFORM2I64ARBPROC) (GLint location, GLint64 x, GLint64 y);
08194 GLAPI PFNGLUNIFORM2I64ARBPROC glad_glUniform2i64ARB;
08195 #define glUniform2i64ARB glad_glUniform2i64ARB
08196 typedef void (APIENTRY PFNGLUNIFORM3I64ARBPROC) (GLint location, GLint64 x, GLint64 y, GLint64 z);
08197 GLAPI PFNGLUNIFORM3I64ARBPROC glad_glUniform3i64ARB;
08198 #define glUniform3i64ARB glad_glUniform3i64ARB
08199 typedef void (APIENTRY PFNGLUNIFORM4I64ARBPROC) (GLint location, GLint64 x, GLint64 y, GLint64 z,
GLint64 w);
08200 GLAPI PFNGLUNIFORM4I64ARBPROC glad_glUniform4i64ARB;
08201 #define glUniform4i64ARB glad_glUniform4i64ARB
08202 typedef void (APIENTRY PFNGLUNIFORM1I64VARBPROC) (GLint location, GLsizei count, const GLint64
*value);
08203 GLAPI PFNGLUNIFORM1I64VARBPROC glad_glUniform1i64vARB;
08204 #define glUniform1i64vARB glad_glUniform1i64vARB
08205 typedef void (APIENTRY PFNGLUNIFORM2I64VARBPROC) (GLint location, GLsizei count, const GLint64
*value);
08206 GLAPI PFNGLUNIFORM2I64VARBPROC glad_glUniform2i64vARB;
08207 #define glUniform2i64vARB glad_glUniform2i64vARB
08208 typedef void (APIENTRY PFNGLUNIFORM3I64VARBPROC) (GLint location, GLsizei count, const GLint64
*value);
08209 GLAPI PFNGLUNIFORM3I64VARBPROC glad_glUniform3i64vARB;
08210 #define glUniform3i64vARB glad_glUniform3i64vARB
08211 typedef void (APIENTRY PFNGLUNIFORM4I64VARBPROC) (GLint location, GLsizei count, const GLint64
*value);
08212 GLAPI PFNGLUNIFORM4I64VARBPROC glad_glUniform4i64vARB;
08213 #define glUniform4i64vARB glad_glUniform4i64vARB
08214 typedef void (APIENTRY PFNGLUNIFORM1UI64ARBPROC) (GLint location, GLuint64 x);
08215 GLAPI PFNGLUNIFORM1UI64ARBPROC glad_glUniform1ui64ARB;
08216 #define glUniform1ui64ARB glad_glUniform1ui64ARB
08217 typedef void (APIENTRY PFNGLUNIFORM2UI64ARBPROC) (GLint location, GLuint64 x, GLuint64 y);
08218 GLAPI PFNGLUNIFORM2UI64ARBPROC glad_glUniform2ui64ARB;
08219 #define glUniform2ui64ARB glad_glUniform2ui64ARB
08220 typedef void (APIENTRY PFNGLUNIFORM3UI64ARBPROC) (GLint location, GLuint64 x, GLuint64 y, GLuint64 z);
08221 GLAPI PFNGLUNIFORM3UI64ARBPROC glad_glUniform3ui64ARB;
08222 #define glUniform3ui64ARB glad_glUniform3ui64ARB
08223 typedef void (APIENTRY PFNGLUNIFORM4UI64ARBPROC) (GLint location, GLuint64 x, GLuint64 y, GLuint64 z,
GLuint64 w);
08224 GLAPI PFNGLUNIFORM4UI64ARBPROC glad_glUniform4ui64ARB;
08225 #define glUniform4ui64ARB glad_glUniform4ui64ARB
08226 typedef void (APIENTRY PFNGLUNIFORM1UI64VARBPROC) (GLint location, GLsizei count, const GLuint64
*value);
08227 GLAPI PFNGLUNIFORM1UI64VARBPROC glad_glUniform1ui64vARB;
08228 #define glUniform1ui64vARB glad_glUniform1ui64vARB
08229 typedef void (APIENTRY PFNGLUNIFORM2UI64VARBPROC) (GLint location, GLsizei count, const GLuint64
*value);
08230 GLAPI PFNGLUNIFORM2UI64VARBPROC glad_glUniform2ui64vARB;
08231 #define glUniform2ui64vARB glad_glUniform2ui64vARB
08232 typedef void (APIENTRY PFNGLUNIFORM3UI64VARBPROC) (GLint location, GLsizei count, const GLuint64
*value);

```

```

08233 GLAPI PFNGLUNIFORM3UI64VARBPROC glad_glUniform3ui64vARB;
08234 #define glUniform3ui64vARB glad_glUniform3ui64vARB
08235 typedef void (APIENTRY PFNGLUNIFORM4UI64VARBPROC) (GLint location, GLsizei count, const GLuint64
    *value);
08236 GLAPI PFNGLUNIFORM4UI64VARBPROC glad_glUniform4ui64vARB;
08237 #define glUniform4ui64vARB glad_glUniform4ui64vARB
08238 typedef void (APIENTRY PFNGLGETUNIFORMI64VARBPROC) (GLuint program, GLint location, GLint64 *params);
08239 GLAPI PFNGLGETUNIFORMI64VARBPROC glad_glGetUniformi64vARB;
08240 #define glGetUniformi64vARB glad_glGetUniformi64vARB
08241 typedef void (APIENTRY PFNGLGETUNIFORMUI64VARBPROC) (GLuint program, GLint location, GLuint64
    *params);
08242 GLAPI PFNGLGETUNIFORMUI64VARBPROC glad_glGetUniformui64vARB;
08243 #define glGetUniformui64vARB glad_glGetUniformui64vARB
08244 typedef void (APIENTRY PFNGLGETUNIFORMI64VARBPROC) (GLuint program, GLint location, GLsizei bufSize,
    GLint64 *params);
08245 GLAPI PFNGLGETUNIFORMI64VARBPROC glad_glGetnUniformi64vARB;
08246 #define glGetnUniformi64vARB glad_glGetnUniformi64vARB
08247 typedef void (APIENTRY PFNGLGETUNIFORMUI64VARBPROC) (GLuint program, GLint location, GLsizei bufSize,
    GLuint64 *params);
08248 GLAPI PFNGLGETUNIFORMUI64VARBPROC glad_glGetnUniformui64vARB;
08249 #define glGetnUniformui64vARB glad_glGetnUniformui64vARB
08250 typedef void (APIENTRY PFNGLPROGRAMUNIFORMI64ARBPROC) (GLuint program, GLint location, GLint64 x);
08251 GLAPI PFNGLPROGRAMUNIFORMI64ARBPROC glad_glProgramUniformi64ARB;
08252 #define glProgramUniformi64ARB glad_glProgramUniformi64ARB
08253 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2I64ARBPROC) (GLuint program, GLint location, GLint64 x,
    GLint64 y);
08254 GLAPI PFNGLPROGRAMUNIFORM2I64ARBPROC glad_glProgramUniform2i64ARB;
08255 #define glProgramUniform2i64ARB glad_glProgramUniform2i64ARB
08256 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3I64ARBPROC) (GLuint program, GLint location, GLint64 x,
    GLint64 y, GLint64 z);
08257 GLAPI PFNGLPROGRAMUNIFORM3I64ARBPROC glad_glProgramUniform3i64ARB;
08258 #define glProgramUniform3i64ARB glad_glProgramUniform3i64ARB
08259 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4I64ARBPROC) (GLuint program, GLint location, GLint64 x,
    GLint64 y, GLint64 z, GLint64 w);
08260 GLAPI PFNGLPROGRAMUNIFORM4I64ARBPROC glad_glProgramUniform4i64ARB;
08261 #define glProgramUniform4i64ARB glad_glProgramUniform4i64ARB
08262 typedef void (APIENTRY PFNGLPROGRAMUNIFORMI64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLint64 *value);
08263 GLAPI PFNGLPROGRAMUNIFORMI64VARBPROC glad_glProgramUniformi64vARB;
08264 #define glProgramUniformi64vARB glad_glProgramUniformi64vARB
08265 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2I64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLint64 *value);
08266 GLAPI PFNGLPROGRAMUNIFORM2I64VARBPROC glad_glProgramUniform2i64vARB;
08267 #define glProgramUniform2i64vARB glad_glProgramUniform2i64vARB
08268 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3I64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLint64 *value);
08269 GLAPI PFNGLPROGRAMUNIFORM3I64VARBPROC glad_glProgramUniform3i64vARB;
08270 #define glProgramUniform3i64vARB glad_glProgramUniform3i64vARB
08271 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4I64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLint64 *value);
08272 GLAPI PFNGLPROGRAMUNIFORM4I64VARBPROC glad_glProgramUniform4i64vARB;
08273 #define glProgramUniform4i64vARB glad_glProgramUniform4i64vARB
08274 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UI64ARBPROC) (GLuint program, GLint location, GLuint64 x);
08275 GLAPI PFNGLPROGRAMUNIFORM1UI64ARBPROC glad_glProgramUniform1ui64ARB;
08276 #define glProgramUniform1ui64ARB glad_glProgramUniform1ui64ARB
08277 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UI64ARBPROC) (GLuint program, GLint location, GLuint64 x,
    GLuint64 y);
08278 GLAPI PFNGLPROGRAMUNIFORM2UI64ARBPROC glad_glProgramUniform2ui64ARB;
08279 #define glProgramUniform2ui64ARB glad_glProgramUniform2ui64ARB
08280 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UI64ARBPROC) (GLuint program, GLint location, GLuint64 x,
    GLuint64 y, GLuint64 z);
08281 GLAPI PFNGLPROGRAMUNIFORM3UI64ARBPROC glad_glProgramUniform3ui64ARB;
08282 #define glProgramUniform3ui64ARB glad_glProgramUniform3ui64ARB
08283 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UI64ARBPROC) (GLuint program, GLint location, GLuint64 x,
    GLuint64 y, GLuint64 z, GLuint64 w);
08284 GLAPI PFNGLPROGRAMUNIFORM4UI64ARBPROC glad_glProgramUniform4ui64ARB;
08285 #define glProgramUniform4ui64ARB glad_glProgramUniform4ui64ARB
08286 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UI64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLuint64 *value);
08287 GLAPI PFNGLPROGRAMUNIFORM1UI64VARBPROC glad_glProgramUniform1ui64vARB;
08288 #define glProgramUniform1ui64vARB glad_glProgramUniform1ui64vARB
08289 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UI64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLuint64 *value);
08290 GLAPI PFNGLPROGRAMUNIFORM2UI64VARBPROC glad_glProgramUniform2ui64vARB;
08291 #define glProgramUniform2ui64vARB glad_glProgramUniform2ui64vARB
08292 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UI64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLuint64 *value);
08293 GLAPI PFNGLPROGRAMUNIFORM3UI64VARBPROC glad_glProgramUniform3ui64vARB;
08294 #define glProgramUniform3ui64vARB glad_glProgramUniform3ui64vARB
08295 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UI64VARBPROC) (GLuint program, GLint location, GLsizei
    count, const GLuint64 *value);
08296 GLAPI PFNGLPROGRAMUNIFORM4UI64VARBPROC glad_glProgramUniform4ui64vARB;
08297 #define glProgramUniform4ui64vARB glad_glProgramUniform4ui64vARB
08298 #endif
08299 #ifndef GL_ARB_half_float_pixel
08300 #define GL_ARB_half_float_pixel 1
08301 GLAPI int GLAD_GL_ARB_half_float_pixel;

```

```

08302 #endif
08303 #ifndef GL_ARB_half_float_vertex
08304 #define GL_ARB_half_float_vertex 1
08305 GLAPI int GLAD_GL_ARB_half_float_vertex;
08306 #endif
08307 #ifndef GL_ARB_imaging
08308 #define GL_ARB_imaging 1
08309 GLAPI int GLAD_GL_ARB_imaging;
08310 typedef void (APIENTRY PFNGLCOLORTABLEPROC) (GLenum target, GLenum internalformat, GLsizei width,
    GLenum format, GLenum type, const void *table);
08311 GLAPI PFNGLCOLORTABLEPROC glad_glColorTable;
08312 #define glColorTable glad_glColorTable
08313 typedef void (APIENTRY PFNGLCOLORTABLEPARAMETERFVPROC) (GLenum target, GLenum pname, const GLfloat
    *params);
08314 GLAPI PFNGLCOLORTABLEPARAMETERFVPROC glad_glColorTableParameterfv;
08315 #define glColorTableParameterfv glad_glColorTableParameterfv
08316 typedef void (APIENTRY PFNGLCOLORTABLEPARAMETERIVPROC) (GLenum target, GLenum pname, const GLint
    *params);
08317 GLAPI PFNGLCOLORTABLEPARAMETERIVPROC glad_glColorTableParameteriv;
08318 #define glColorTableParameteriv glad_glColorTableParameteriv
08319 typedef void (APIENTRY PFNGLCOPYCOLORTABLEPROC) (GLenum target, GLenum internalformat, GLint x, GLint
    y, GLsizei width);
08320 GLAPI PFNGLCOPYCOLORTABLEPROC glad_glCopyColorTable;
08321 #define glCopyColorTable glad_glCopyColorTable
08322 typedef void (APIENTRY PFNGLGETCOLORTABLEPROC) (GLenum target, GLenum format, GLenum type, void
    *table);
08323 GLAPI PFNGLGETCOLORTABLEPROC glad_glGetColorTable;
08324 #define glGetColorTable glad_glGetColorTable
08325 typedef void (APIENTRY PFNGLGETCOLORTABLEPARAMETERFVPROC) (GLenum target, GLenum pname, GLfloat
    *params);
08326 GLAPI PFNGLGETCOLORTABLEPARAMETERFVPROC glad_glGetColorTableParameterfv;
08327 #define glGetColorTableParameterfv glad_glGetColorTableParameterfv
08328 typedef void (APIENTRY PFNGLGETCOLORTABLEPARAMETERIVPROC) (GLenum target, GLenum pname, GLint
    *params);
08329 GLAPI PFNGLGETCOLORTABLEPARAMETERIVPROC glad_glGetColorTableParameteriv;
08330 #define glGetColorTableParameteriv glad_glGetColorTableParameteriv
08331 typedef void (APIENTRY PFNGLCOLORSUBTABLEPROC) (GLenum target, GLsizei start, GLsizei count, GLenum
    format, GLenum type, const void *data);
08332 GLAPI PFNGLCOLORSUBTABLEPROC glad_glColorSubTable;
08333 #define glColorSubTable glad_glColorSubTable
08334 typedef void (APIENTRY PFNGLCOPYCOLORSUBTABLEPROC) (GLenum target, GLsizei start, GLint x, GLint y,
    GLsizei width);
08335 GLAPI PFNGLCOPYCOLORSUBTABLEPROC glad_glCopyColorSubTable;
08336 #define glCopyColorSubTable glad_glCopyColorSubTable
08337 typedef void (APIENTRY PFNGLCONVOLUTIONFILTER1DPROC) (GLenum target, GLenum internalformat, GLsizei
    width, GLenum format, GLenum type, const void *image);
08338 GLAPI PFNGLCONVOLUTIONFILTER1DPROC glad_glConvolutionFilter1D;
08339 #define glConvolutionFilter1D glad_glConvolutionFilter1D
08340 typedef void (APIENTRY PFNGLCONVOLUTIONFILTER2DPROC) (GLenum target, GLenum internalformat, GLsizei
    width, GLsizei height, GLenum format, GLenum type, const void *image);
08341 GLAPI PFNGLCONVOLUTIONFILTER2DPROC glad_glConvolutionFilter2D;
08342 #define glConvolutionFilter2D glad_glConvolutionFilter2D
08343 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERFPROC) (GLenum target, GLenum pname, GLfloat params);
08344 GLAPI PFNGLCONVOLUTIONPARAMETERFPROC glad_glConvolutionParameterf;
08345 #define glConvolutionParameterf glad_glConvolutionParameterf
08346 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERFVPROC) (GLenum target, GLenum pname, const GLfloat
    *params);
08347 GLAPI PFNGLCONVOLUTIONPARAMETERFVPROC glad_glConvolutionParameterfv;
08348 #define glConvolutionParameterfv glad_glConvolutionParameterfv
08349 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERIPROC) (GLenum target, GLenum pname, GLint params);
08350 GLAPI PFNGLCONVOLUTIONPARAMETERIPROC glad_glConvolutionParameteri;
08351 #define glConvolutionParameteri glad_glConvolutionParameteri
08352 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERIVPROC) (GLenum target, GLenum pname, const GLint
    *params);
08353 GLAPI PFNGLCONVOLUTIONPARAMETERIVPROC glad_glConvolutionParameteriv;
08354 #define glConvolutionParameteriv glad_glConvolutionParameteriv
08355 typedef void (APIENTRY PFNGLCOPYCONVOLUTIONFILTER1DPROC) (GLenum target, GLenum internalformat, GLint
    x, GLint y, GLsizei width);
08356 GLAPI PFNGLCOPYCONVOLUTIONFILTER1DPROC glad_glCopyConvolutionFilter1D;
08357 #define glCopyConvolutionFilter1D glad_glCopyConvolutionFilter1D
08358 typedef void (APIENTRY PFNGLCOPYCONVOLUTIONFILTER2DPROC) (GLenum target, GLenum internalformat, GLint
    x, GLint y, GLsizei width, GLsizei height);
08359 GLAPI PFNGLCOPYCONVOLUTIONFILTER2DPROC glad_glCopyConvolutionFilter2D;
08360 #define glCopyConvolutionFilter2D glad_glCopyConvolutionFilter2D
08361 typedef void (APIENTRY PFNGLGETCONVOLUTIONFILTERPROC) (GLenum target, GLenum format, GLenum type, void
    *image);
08362 GLAPI PFNGLGETCONVOLUTIONFILTERPROC glad_glGetConvolutionFilter;
08363 #define glGetConvolutionFilter glad_glGetConvolutionFilter
08364 typedef void (APIENTRY PFNGLGETCONVOLUTIONPARAMETERFVPROC) (GLenum target, GLenum pname, GLfloat
    *params);
08365 GLAPI PFNGLGETCONVOLUTIONPARAMETERFVPROC glad_glGetConvolutionParameterfv;
08366 #define glGetConvolutionParameterfv glad_glGetConvolutionParameterfv
08367 typedef void (APIENTRY PFNGLGETCONVOLUTIONPARAMETERIVPROC) (GLenum target, GLenum pname, GLint
    *params);
08368 GLAPI PFNGLGETCONVOLUTIONPARAMETERIVPROC glad_glGetConvolutionParameteriv;
08369 #define glGetConvolutionParameteriv glad_glGetConvolutionParameteriv
08370 typedef void (APIENTRY PFNGLGETSEPARABLEFILTERPROC) (GLenum target, GLenum format, GLenum type, void

```

```

        *row, void *column, void *span);
08371 GLAPI PFNGLGETSEPARABLEFILTERPROC glad_glGetSeparableFilter;
08372 #define glGetSeparableFilter glad_glGetSeparableFilter
08373 typedef void (APIENTRY PFNGLSEPARABLEFILTER2DPROC) (GLenum target, GLenum internalformat, GLsizei
width, GLsizei height, GLenum format, GLenum type, const void *row, const void *column);
08374 GLAPI PFNGLSEPARABLEFILTER2DPROC glad_glSeparableFilter2D;
08375 #define glSeparableFilter2D glad_glSeparableFilter2D
08376 typedef void (APIENTRY PFNGLGETHISTOGRAMPROC) (GLenum target, GLboolean reset, GLenum format, GLenum
type, void *values);
08377 GLAPI PFNGLGETHISTOGRAMPROC glad_glGetHistogram;
08378 #define glGetHistogram glad_glGetHistogram
08379 typedef void (APIENTRY PFNGLGETHISTOGRAMPARAMETERFVPROC) (GLenum target, GLenum pname, GLfloat
*params);
08380 GLAPI PFNGLGETHISTOGRAMPARAMETERFVPROC glad_glGetHistogramParameterfv;
08381 #define glGetHistogramParameterfv glad_glGetHistogramParameterfv
08382 typedef void (APIENTRY PFNGLGETHISTOGRAMPARAMETERIVPROC) (GLenum target, GLenum pname, GLint *params);
08383 GLAPI PFNGLGETHISTOGRAMPARAMETERIVPROC glad_glGetHistogramParameteriv;
08384 #define glGetHistogramParameteriv glad_glGetHistogramParameteriv
08385 typedef void (APIENTRY PFNGLGETMINMAXPROC) (GLenum target, GLboolean reset, GLenum format, GLenum
type, void *values);
08386 GLAPI PFNGLGETMINMAXPROC glad_glGetMinmax;
08387 #define glGetMinmax glad_glGetMinmax
08388 typedef void (APIENTRY PFNGLGETMINMAXPARAMETERFVPROC) (GLenum target, GLenum pname, GLfloat *params);
08389 GLAPI PFNGLGETMINMAXPARAMETERFVPROC glad_glGetMinmaxParameterfv;
08390 #define glGetMinmaxParameterfv glad_glGetMinmaxParameterfv
08391 typedef void (APIENTRY PFNGLGETMINMAXPARAMETERIVPROC) (GLenum target, GLenum pname, GLint *params);
08392 GLAPI PFNGLGETMINMAXPARAMETERIVPROC glad_glGetMinmaxParameteriv;
08393 #define glGetMinmaxParameteriv glad_glGetMinmaxParameteriv
08394 typedef void (APIENTRY PFNGLHISTOGRAMPROC) (GLenum target, GLsizei width, GLenum internalformat,
GLboolean sink);
08395 GLAPI PFNGLHISTOGRAMPROC glad_glHistogram;
08396 #define glGetHistogram glad_glHistogram
08397 typedef void (APIENTRY PFNGLMINMAXPROC) (GLenum target, GLenum internalformat, GLboolean sink);
08398 GLAPI PFNGLMINMAXPROC glad_glMinmax;
08399 #define glGetMinmax glad_glMinmax
08400 typedef void (APIENTRY PFNGLRESETHISTOGRAMPROC) (GLenum target);
08401 GLAPI PFNGLRESETHISTOGRAMPROC glad_glResetHistogram;
08402 #define glResetHistogram glad_glResetHistogram
08403 typedef void (APIENTRY PFNGLRESETMINMAXPROC) (GLenum target);
08404 GLAPI PFNGLRESETMINMAXPROC glad_glResetMinmax;
08405 #define glResetMinmax glad_glResetMinmax
08406 #endif
08407 #ifndef GL_ARB_indirect_parameters
08408 #define GL_ARB_indirect_parameters 1
08409 GLAPI int GLAD_GL_ARB_indirect_parameters;
08410 typedef void (APIENTRY PFNGLMULTIDRAWARRAYSINDIRECTCOUNTPROC) (GLenum mode, const void *indirect,
GLintptr drawcount, GLsizei maxdrawcount, GLsizei stride);
08411 GLAPI PFNGLMULTIDRAWARRAYSINDIRECTCOUNTPROC glad_glMultiDrawArraysIndirectCountARB;
08412 #define glMultiDrawArraysIndirectCountARB glad_glMultiDrawArraysIndirectCountARB
08413 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSINDIRECTCOUNTPROC) (GLenum mode, GLenum type, const
void *indirect, GLintptr drawcount, GLsizei maxdrawcount, GLsizei stride);
08414 GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTCOUNTPROC glad_glMultiDrawElementsIndirectCountARB;
08415 #define glMultiDrawElementsIndirectCountARB glad_glMultiDrawElementsIndirectCountARB
08416 #endif
08417 #ifndef GL_ARB_instanced_arrays
08418 #define GL_ARB_instanced_arrays 1
08419 GLAPI int GLAD_GL_ARB_instanced_arrays;
08420 typedef void (APIENTRY PFNGLVERTEXATTRIBDIVISORARBPROC) (GLuint index, GLuint divisor);
08421 GLAPI PFNGLVERTEXATTRIBDIVISORARBPROC glad_glVertexAttribDivisorARB;
08422 #define glVertexAttribDivisorARB glad_glVertexAttribDivisorARB
08423 #endif
08424 #ifndef GL_ARB_internalformat_query
08425 #define GL_ARB_internalformat_query 1
08426 GLAPI int GLAD_GL_ARB_internalformat_query;
08427 typedef void (APIENTRY PFNGLGETINTERNALFORMATIVPROC) (GLenum target, GLenum internalformat, GLenum
pname, GLsizei count, GLint *params);
08428 GLAPI PFNGLGETINTERNALFORMATIVPROC glad_glGetInternalformativ;
08429 #define glGetInternalformativ glad_glGetInternalformativ
08430 #endif
08431 #ifndef GL_ARB_internalformat_query2
08432 #define GL_ARB_internalformat_query2 1
08433 GLAPI int GLAD_GL_ARB_internalformat_query2;
08434 typedef void (APIENTRY PFNGLGETINTERNALFORMATI64VPROC) (GLenum target, GLenum internalformat, GLenum
pname, GLsizei count, GLint64 *params);
08435 GLAPI PFNGLGETINTERNALFORMATI64VPROC glad_glGetInternalformati64v;
08436 #define glGetInternalformati64v glad_glGetInternalformati64v
08437 #endif
08438 #ifndef GL_ARB_invalidate_subdata
08439 #define GL_ARB_invalidate_subdata 1
08440 GLAPI int GLAD_GL_ARB_invalidate_subdata;
08441 typedef void (APIENTRY PFNGLINVALIDATETEXSUBIMAGEPROC) (GLuint texture, GLint level, GLint xoffset,
GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth);
08442 GLAPI PFNGLINVALIDATETEXSUBIMAGEPROC glad_glInvalidateTexSubImage;
08443 #define glInvalidateTexSubImage glad_glInvalidateTexSubImage
08444 typedef void (APIENTRY PFNGLINVALIDATETEXIMAGEPROC) (GLuint texture, GLint level);
08445 GLAPI PFNGLINVALIDATETEXIMAGEPROC glad_glInvalidateTexImage;
08446 #define glInvalidateTexImage glad_glInvalidateTexImage

```

```

08447 typedef void (APIENTRY PFNGLINVALIDATEBUFFERSUBDATAPROC) (GLuint buffer, GLintptr offset, GLsizeiptr
length);
08448 GLAPI PFNGLINVALIDATEBUFFERSUBDATAPROC glad_glInvalidateBufferSubData;
08449 #define glInvalidateBufferSubData glad_glInvalidateBufferSubData
08450 typedef void (APIENTRY PFNGLINVALIDATEBUFFERDATAPROC) (GLuint buffer);
08451 GLAPI PFNGLINVALIDATEBUFFERDATAPROC glad_glInvalidateBufferData;
08452 #define glInvalidateBufferData glad_glInvalidateBufferData
08453 typedef void (APIENTRY PFNGLINVALIDATEFRAMEBUFFERPROC) (GLenum target, GLsizei numAttachments, const
GLenum *attachments);
08454 GLAPI PFNGLINVALIDATEFRAMEBUFFERPROC glad_glInvalidateFramebuffer;
08455 #define glInvalidateFramebuffer glad_glInvalidateFramebuffer
08456 typedef void (APIENTRY PFNGLINVALIDATESUBFRAMEBUFFERPROC) (GLenum target, GLsizei numAttachments,
const GLenum *attachments, GLint x, GLint y, GLsizei width, GLsizei height);
08457 GLAPI PFNGLINVALIDATESUBFRAMEBUFFERPROC glad_glInvalidateSubFramebuffer;
08458 #define glInvalidateSubFramebuffer glad_glInvalidateSubFramebuffer
08459 #endif
08460 #ifndef GL_ARB_map_buffer_alignment
08461 #define GL_ARB_map_buffer_alignment 1
08462 GLAPI int GLAD_GL_ARB_map_buffer_alignment;
08463 #endif
08464 #ifndef GL_ARB_matrix_palette
08465 #define GL_ARB_matrix_palette 1
08466 GLAPI int GLAD_GL_ARB_matrix_palette;
08467 #endif
08468 #ifndef GL_ARB_matrix_palette
08469 #define GL_ARB_matrix_palette 1
08470 GLAPI int GLAD_GL_ARB_matrix_palette;
08471 typedef void (APIENTRY PFNGLCURRENTPALETTEMATRIXARBPROC) (GLint index);
08472 GLAPI PFNGLCURRENTPALETTEMATRIXARBPROC glad_glCurrentPaletteMatrixARB;
08473 #define glCurrentPaletteMatrixARB glad_glCurrentPaletteMatrixARB
08474 typedef void (APIENTRY PFNGLMATRIXINDEXUBVARBPROC) (GLint size, const GLubyte *indices);
08475 GLAPI PFNGLMATRIXINDEXUBVARBPROC glad_glMatrixIndexubVARB;
08476 #define glMatrixIndexubVARB glad_glMatrixIndexubVARB
08477 typedef void (APIENTRY PFNGLMATRIXINDEXUSVARBPROC) (GLint size, const GLushort *indices);
08478 GLAPI PFNGLMATRIXINDEXUSVARBPROC glad_glMatrixIndexusVARB;
08479 #define glMatrixIndexusVARB glad_glMatrixIndexusVARB
08480 typedef void (APIENTRY PFNGLMATRIXINDEXUIVARBPROC) (GLint size, const GLuint *indices);
08481 GLAPI PFNGLMATRIXINDEXUIVARBPROC glad_glMatrixIndexuiVARB;
08482 #define glMatrixIndexuiVARB glad_glMatrixIndexuiVARB
08483 typedef void (APIENTRY PFNGLMATRIXINDEXPOINTERARBPROC) (GLint size, GLenum type, GLsizei stride, const
void *pointer);
08484 GLAPI PFNGLMATRIXINDEXPOINTERARBPROC glad_glMatrixIndexPointerARB;
08485 #define glMatrixIndexPointerARB glad_glMatrixIndexPointerARB
08486 #endif
08487 #ifndef GL_ARB_multi_bind
08488 #define GL_ARB_multi_bind 1
08489 GLAPI int GLAD_GL_ARB_multi_bind;
08490 typedef void (APIENTRY PFNGLBINDBUFFERSBASEPROC) (GLenum target, GLuint first, GLsizei count, const
GLuint *buffers);
08491 GLAPI PFNGLBINDBUFFERSBASEPROC glad_glBindBuffersBase;
08492 #define glBindBuffersBase glad_glBindBuffersBase
08493 typedef void (APIENTRY PFNGLBINDBUFFERSRANGEPROC) (GLenum target, GLuint first, GLsizei count, const
GLuint *buffers, const GLintptr *offsets, const GLsizeiptr *sizes);
08494 GLAPI PFNGLBINDBUFFERSRANGEPROC glad_glBindBuffersRange;
08495 #define glBindBuffersRange glad_glBindBuffersRange
08496 typedef void (APIENTRY PFNGLBINDTEXTURESPROC) (GLuint first, GLsizei count, const GLuint *textures);
08497 GLAPI PFNGLBINDTEXTURESPROC glad_glBindTextures;
08498 #define glBindTextures glad_glBindTextures
08499 typedef void (APIENTRY PFNGLBINDSAMPLERSPROC) (GLuint first, GLsizei count, const GLuint *samplers);
08500 GLAPI PFNGLBINDSAMPLERSPROC glad_glBindSamplers;
08501 #define glBindSamplers glad_glBindSamplers
08502 typedef void (APIENTRY PFNGLBINDIMAGETEXTURESPROC) (GLuint first, GLsizei count, const GLuint
*textures);
08503 GLAPI PFNGLBINDIMAGETEXTURESPROC glad_glBindImageTextures;
08504 #define glBindImageTextures glad_glBindImageTextures
08505 typedef void (APIENTRY PFNGLBINDVERTEXBUFFERSPROC) (GLuint first, GLsizei count, const GLuint
*buffers, const GLintptr *offsets, const GLsizei *strides);
08506 GLAPI PFNGLBINDVERTEXBUFFERSPROC glad_glBindVertexBuffers;
08507 #define glBindVertexBuffers glad_glBindVertexBuffers
08508 #endif
08509 #ifndef GL_ARB_multi_draw_indirect
08510 #define GL_ARB_multi_draw_indirect 1
08511 GLAPI int GLAD_GL_ARB_multi_draw_indirect;
08512 typedef void (APIENTRY PFNGLMULTIDRAWARRAYSINDIRECTPROC) (GLenum mode, const void *indirect, GLsizei
drawcount, GLsizei stride);
08513 GLAPI PFNGLMULTIDRAWARRAYSINDIRECTPROC glad_glMultiDrawArraysIndirect;
08514 #define glMultiDrawArraysIndirect glad_glMultiDrawArraysIndirect
08515 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSINDIRECTPROC) (GLenum mode, GLenum type, const void
*indirect, GLsizei drawcount, GLsizei stride);
08516 GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTPROC glad_glMultiDrawElementsIndirect;
08517 #define glMultiDrawElementsIndirect glad_glMultiDrawElementsIndirect
08518 #endif
08519 #ifndef GL_ARB_multisample
08520 #define GL_ARB_multisample 1
08521 GLAPI int GLAD_GL_ARB_multisample;
08522 typedef void (APIENTRY PFNGLSAMPLECOVERGEARBPROC) (GLfloat value, GLboolean invert);
08523 GLAPI PFNGLSAMPLECOVERGEARBPROC glad_glSampleCoverageARB;

```



```

08524 #define glSampleCoverageARB glad_glSampleCoverageARB
08525 #endif
08526 #ifndef GL_ARB_multitexture
08527 #define GL_ARB_multitexture 1
08528 GLAPI int GLAD_GL_ARB_multitexture;
08529 typedef void (APIENTRY PFNGLACTIVETEXTUREARBPROC) (GLenum texture);
08530 GLAPI PFNGLACTIVETEXTUREARBPROC glad_glActiveTextureARB;
08531 #define glActiveTextureARB glad_glActiveTextureARB
08532 typedef void (APIENTRY PFNGLCLIENTACTIVETEXTUREARBPROC) (GLenum texture);
08533 GLAPI PFNGLCLIENTACTIVETEXTUREARBPROC glad_glClientActiveTextureARB;
08534 #define glClientActiveTextureARB glad_glClientActiveTextureARB
08535 typedef void (APIENTRY PFNGLMULTITEXCOORD1DARBPROC) (GLenum target, GLdouble s);
08536 GLAPI PFNGLMULTITEXCOORD1DARBPROC glad_glMultiTexCoord1dARB;
08537 #define glMultiTexCoord1dARB glad_glMultiTexCoord1dARB
08538 typedef void (APIENTRY PFNGLMULTITEXCOORD1DVARBPROC) (GLenum target, const GLdouble *v);
08539 GLAPI PFNGLMULTITEXCOORD1DVARBPROC glad_glMultiTexCoord1dvARB;
08540 #define glMultiTexCoord1dvARB glad_glMultiTexCoord1dvARB
08541 typedef void (APIENTRY PFNGLMULTITEXCOORD1FARBPROC) (GLenum target, GLfloat s);
08542 GLAPI PFNGLMULTITEXCOORD1FARBPROC glad_glMultiTexCoord1fARB;
08543 #define glMultiTexCoord1fARB glad_glMultiTexCoord1fARB
08544 typedef void (APIENTRY PFNGLMULTITEXCOORD1FVARBPROC) (GLenum target, const GLfloat *v);
08545 GLAPI PFNGLMULTITEXCOORD1FVARBPROC glad_glMultiTexCoord1fvARB;
08546 #define glMultiTexCoord1fvARB glad_glMultiTexCoord1fvARB
08547 typedef void (APIENTRY PFNGLMULTITEXCOORD1IARBPROC) (GLenum target, GLint s);
08548 GLAPI PFNGLMULTITEXCOORD1IARBPROC glad_glMultiTexCoord1iARB;
08549 #define glMultiTexCoord1iARB glad_glMultiTexCoord1iARB
08550 typedef void (APIENTRY PFNGLMULTITEXCOORD1IVARBPROC) (GLenum target, const GLint *v);
08551 GLAPI PFNGLMULTITEXCOORD1IVARBPROC glad_glMultiTexCoord1ivARB;
08552 #define glMultiTexCoord1ivARB glad_glMultiTexCoord1ivARB
08553 typedef void (APIENTRY PFNGLMULTITEXCOORD1SARBPROC) (GLenum target, GLshort s);
08554 GLAPI PFNGLMULTITEXCOORD1SARBPROC glad_glMultiTexCoord1sARB;
08555 #define glMultiTexCoord1sARB glad_glMultiTexCoord1sARB
08556 typedef void (APIENTRY PFNGLMULTITEXCOORD1SVARBPROC) (GLenum target, const GLshort *v);
08557 GLAPI PFNGLMULTITEXCOORD1SVARBPROC glad_glMultiTexCoord1svARB;
08558 #define glMultiTexCoord1svARB glad_glMultiTexCoord1svARB
08559 typedef void (APIENTRY PFNGLMULTITEXCOORD2DARBPROC) (GLenum target, GLdouble s, GLdouble t);
08560 GLAPI PFNGLMULTITEXCOORD2DARBPROC glad_glMultiTexCoord2dARB;
08561 #define glMultiTexCoord2dARB glad_glMultiTexCoord2dARB
08562 typedef void (APIENTRY PFNGLMULTITEXCOORD2DVARBPROC) (GLenum target, const GLdouble *v);
08563 GLAPI PFNGLMULTITEXCOORD2DVARBPROC glad_glMultiTexCoord2dvARB;
08564 #define glMultiTexCoord2dvARB glad_glMultiTexCoord2dvARB
08565 typedef void (APIENTRY PFNGLMULTITEXCOORD2FARBPROC) (GLenum target, GLfloat s, GLfloat t);
08566 GLAPI PFNGLMULTITEXCOORD2FARBPROC glad_glMultiTexCoord2fARB;
08567 #define glMultiTexCoord2fARB glad_glMultiTexCoord2fARB
08568 typedef void (APIENTRY PFNGLMULTITEXCOORD2FVARBPROC) (GLenum target, const GLfloat *v);
08569 GLAPI PFNGLMULTITEXCOORD2FVARBPROC glad_glMultiTexCoord2fvARB;
08570 #define glMultiTexCoord2fvARB glad_glMultiTexCoord2fvARB
08571 typedef void (APIENTRY PFNGLMULTITEXCOORD2IARBPROC) (GLenum target, GLint s, GLint t);
08572 GLAPI PFNGLMULTITEXCOORD2IARBPROC glad_glMultiTexCoord2iARB;
08573 #define glMultiTexCoord2iARB glad_glMultiTexCoord2iARB
08574 typedef void (APIENTRY PFNGLMULTITEXCOORD2IVARBPROC) (GLenum target, const GLint *v);
08575 GLAPI PFNGLMULTITEXCOORD2IVARBPROC glad_glMultiTexCoord2ivARB;
08576 #define glMultiTexCoord2ivARB glad_glMultiTexCoord2ivARB
08577 typedef void (APIENTRY PFNGLMULTITEXCOORD2SARBPROC) (GLenum target, GLshort s, GLshort t);
08578 GLAPI PFNGLMULTITEXCOORD2SARBPROC glad_glMultiTexCoord2sARB;
08579 #define glMultiTexCoord2sARB glad_glMultiTexCoord2sARB
08580 typedef void (APIENTRY PFNGLMULTITEXCOORD2SVARBPROC) (GLenum target, const GLshort *v);
08581 GLAPI PFNGLMULTITEXCOORD2SVARBPROC glad_glMultiTexCoord2svARB;
08582 #define glMultiTexCoord2svARB glad_glMultiTexCoord2svARB
08583 typedef void (APIENTRY PFNGLMULTITEXCOORD3DARBPROC) (GLenum target, GLdouble s, GLdouble t, GLdouble
r);
08584 GLAPI PFNGLMULTITEXCOORD3DARBPROC glad_glMultiTexCoord3dARB;
08585 #define glMultiTexCoord3dARB glad_glMultiTexCoord3dARB
08586 typedef void (APIENTRY PFNGLMULTITEXCOORD3DVARBPROC) (GLenum target, const GLdouble *v);
08587 GLAPI PFNGLMULTITEXCOORD3DVARBPROC glad_glMultiTexCoord3dvARB;
08588 #define glMultiTexCoord3dvARB glad_glMultiTexCoord3dvARB
08589 typedef void (APIENTRY PFNGLMULTITEXCOORD3FARBPROC) (GLenum target, GLfloat s, GLfloat t, GLfloat r);
08590 GLAPI PFNGLMULTITEXCOORD3FARBPROC glad_glMultiTexCoord3fARB;
08591 #define glMultiTexCoord3fARB glad_glMultiTexCoord3fARB
08592 typedef void (APIENTRY PFNGLMULTITEXCOORD3FVARBPROC) (GLenum target, const GLfloat *v);
08593 GLAPI PFNGLMULTITEXCOORD3FVARBPROC glad_glMultiTexCoord3fvARB;
08594 #define glMultiTexCoord3fvARB glad_glMultiTexCoord3fvARB
08595 typedef void (APIENTRY PFNGLMULTITEXCOORD3IARBPROC) (GLenum target, GLint s, GLint t, GLint r);
08596 GLAPI PFNGLMULTITEXCOORD3IARBPROC glad_glMultiTexCoord3iARB;
08597 #define glMultiTexCoord3iARB glad_glMultiTexCoord3iARB
08598 typedef void (APIENTRY PFNGLMULTITEXCOORD3IVARBPROC) (GLenum target, const GLint *v);
08599 GLAPI PFNGLMULTITEXCOORD3IVARBPROC glad_glMultiTexCoord3ivARB;
08600 #define glMultiTexCoord3ivARB glad_glMultiTexCoord3ivARB
08601 typedef void (APIENTRY PFNGLMULTITEXCOORD3SARBPROC) (GLenum target, GLshort s, GLshort t, GLshort r);
08602 GLAPI PFNGLMULTITEXCOORD3SARBPROC glad_glMultiTexCoord3sARB;
08603 #define glMultiTexCoord3sARB glad_glMultiTexCoord3sARB
08604 typedef void (APIENTRY PFNGLMULTITEXCOORD3SVARBPROC) (GLenum target, const GLshort *v);
08605 GLAPI PFNGLMULTITEXCOORD3SVARBPROC glad_glMultiTexCoord3svARB;
08606 #define glMultiTexCoord3svARB glad_glMultiTexCoord3svARB
08607 typedef void (APIENTRY PFNGLMULTITEXCOORD4DARBPROC) (GLenum target, GLdouble s, GLdouble t, GLdouble
r, GLdouble q);
08608 GLAPI PFNGLMULTITEXCOORD4DARBPROC glad_glMultiTexCoord4dARB;

```

```

08609 #define glMultiTexCoord4dARB glad_glMultiTexCoord4dARB
08610 typedef void (APIENTRY PFNGLMULTITEXCOORD4DVARBPROC) (GLenum target, const GLdouble *v);
08611 GLAPI PFNGLMULTITEXCOORD4DVARBPROC glad_glMultiTexCoord4dvARB;
08612 #define glMultiTexCoord4dvARB glad_glMultiTexCoord4dvARB
08613 typedef void (APIENTRY PFNGLMULTITEXCOORD4FARBPROC) (GLenum target, GLfloat s, GLfloat t, GLfloat r,
GLfloat q);
08614 GLAPI PFNGLMULTITEXCOORD4FARBPROC glad_glMultiTexCoord4fARB;
08615 #define glMultiTexCoord4fARB glad_glMultiTexCoord4fARB
08616 typedef void (APIENTRY PFNGLMULTITEXCOORD4FVARBPROC) (GLenum target, const GLfloat *v);
08617 GLAPI PFNGLMULTITEXCOORD4FVARBPROC glad_glMultiTexCoord4fvARB;
08618 #define glMultiTexCoord4fvARB glad_glMultiTexCoord4fvARB
08619 typedef void (APIENTRY PFNGLMULTITEXCOORD4IARBPROC) (GLenum target, GLint s, GLint t, GLint r, GLint
q);
08620 GLAPI PFNGLMULTITEXCOORD4IARBPROC glad_glMultiTexCoord4iARB;
08621 #define glMultiTexCoord4iARB glad_glMultiTexCoord4iARB
08622 typedef void (APIENTRY PFNGLMULTITEXCOORD4IVARBPROC) (GLenum target, const GLint *v);
08623 GLAPI PFNGLMULTITEXCOORD4IVARBPROC glad_glMultiTexCoord4ivARB;
08624 #define glMultiTexCoord4ivARB glad_glMultiTexCoord4ivARB
08625 typedef void (APIENTRY PFNGLMULTITEXCOORD4SARBPROC) (GLenum target, GLshort s, GLshort t, GLshort r,
GLshort q);
08626 GLAPI PFNGLMULTITEXCOORD4SARBPROC glad_glMultiTexCoord4sARB;
08627 #define glMultiTexCoord4sARB glad_glMultiTexCoord4sARB
08628 typedef void (APIENTRY PFNGLMULTITEXCOORD4SVARBPROC) (GLenum target, const GLshort *v);
08629 GLAPI PFNGLMULTITEXCOORD4SVARBPROC glad_glMultiTexCoord4svARB;
08630 #define glMultiTexCoord4svARB glad_glMultiTexCoord4svARB
08631 #endif
08632 #ifndef GL_ARB_occlusion_query
08633 #define GL_ARB_occlusion_query 1
08634 GLAPI int GLAD_GL_ARB_occlusion_query;
08635 typedef void (APIENTRY PFNGLGENQUERIESARBPROC) (GLsizei n, GLuint *ids);
08636 GLAPI PFNGLGENQUERIESARBPROC glad_glGenQueriesARB;
08637 #define glGenQueriesARB glad_glGenQueriesARB
08638 typedef void (APIENTRY PFNGLDELETEQUERIESARBPROC) (GLsizei n, const GLuint *ids);
08639 GLAPI PFNGLDELETEQUERIESARBPROC glad_glDeleteQueriesARB;
08640 #define glDeleteQueriesARB glad_glDeleteQueriesARB
08641 typedef GLboolean (APIENTRY PFNGLISQUERYARBPROC) (GLuint id);
08642 GLAPI PFNGLISQUERYARBPROC glad_glIsQueryARB;
08643 #define glIsQueryARB glad_glIsQueryARB
08644 typedef void (APIENTRY PFNGLBEGINQUERYARBPROC) (GLenum target, GLuint id);
08645 GLAPI PFNGLBEGINQUERYARBPROC glad_glBeginQueryARB;
08646 #define glBeginQueryARB glad_glBeginQueryARB
08647 typedef void (APIENTRY PFNGLENDQUERYARBPROC) (GLenum target);
08648 GLAPI PFNGLENDQUERYARBPROC glad_glEndQueryARB;
08649 #define glEndQueryARB glad_glEndQueryARB
08650 typedef void (APIENTRY PFNGLGETQUERYIVARBPROC) (GLenum target, GLenum pname, GLint *params);
08651 GLAPI PFNGLGETQUERYIVARBPROC glad_glGetQueryivARB;
08652 #define glGetQueryivARB glad_glGetQueryivARB
08653 typedef void (APIENTRY PFNGLGETQUERYOBJECTIVARBPROC) (GLuint id, GLenum pname, GLint *params);
08654 GLAPI PFNGLGETQUERYOBJECTIVARBPROC glad_glGetQueryObjectivARB;
08655 #define glGetQueryObjectivARB glad_glGetQueryObjectivARB
08656 typedef void (APIENTRY PFNGLGETQUERYOBJECTUIVARBPROC) (GLuint id, GLenum pname, GLuint *params);
08657 GLAPI PFNGLGETQUERYOBJECTUIVARBPROC glad_glGetQueryObjectuivARB;
08658 #define glGetQueryObjectuivARB glad_glGetQueryObjectuivARB
08659 #endif
08660 #ifndef GL_ARB_occlusion_query2
08661 #define GL_ARB_occlusion_query2 1
08662 GLAPI int GLAD_GL_ARB_occlusion_query2;
08663 #endif
08664 #ifndef GL_ARB_parallel_shader_compile
08665 #define GL_ARB_parallel_shader_compile 1
08666 GLAPI int GLAD_GL_ARB_parallel_shader_compile;
08667 typedef void (APIENTRY PFNGLMAXSHADERCOMPILERTHREADSARBPROC) (GLuint count);
08668 GLAPI PFNGLMAXSHADERCOMPILERTHREADSARBPROC glad_glMaxShaderCompilerThreadsARB;
08669 #define glMaxShaderCompilerThreadsARB glad_glMaxShaderCompilerThreadsARB
08670 #endif
08671 #ifndef GL_ARB_pipeline_statistics_query
08672 #define GL_ARB_pipeline_statistics_query 1
08673 GLAPI int GLAD_GL_ARB_pipeline_statistics_query;
08674 #endif
08675 #ifndef GL_ARB_pixel_buffer_object
08676 #define GL_ARB_pixel_buffer_object 1
08677 GLAPI int GLAD_GL_ARB_pixel_buffer_object;
08678 #endif
08679 #ifndef GL_ARB_point_parameters
08680 #define GL_ARB_point_parameters 1
08681 GLAPI int GLAD_GL_ARB_point_parameters;
08682 typedef void (APIENTRY PFNGLPOINTPARAMETERFARBPROC) (GLenum pname, GLfloat param);
08683 GLAPI PFNGLPOINTPARAMETERFARBPROC glad_glPointParameterfARB;
08684 #define glPointParameterfARB glad_glPointParameterfARB
08685 typedef void (APIENTRY PFNGLPOINTPARAMETERFVARBPROC) (GLenum pname, const GLfloat *params);
08686 GLAPI PFNGLPOINTPARAMETERFVARBPROC glad_glPointParameterfvARB;
08687 #define glPointParameterfvARB glad_glPointParameterfvARB
08688 #endif
08689 #ifndef GL_ARB_point_sprite
08690 #define GL_ARB_point_sprite 1
08691 GLAPI int GLAD_GL_ARB_point_sprite;
08692 #endif

```

```

08693 #ifndef GL_ARB_polygon_offset_clamp
08694 #define GL_ARB_polygon_offset_clamp 1
08695 GLAPI int GLAD_GL_ARB_polygon_offset_clamp;
08696 typedef void (APIENTRY PFNGLPOLYGONOFFSETCLAMPPROC) (GLfloat factor, GLfloat units, GLfloat clamp);
08697 GLAPI PFNGLPOLYGONOFFSETCLAMPPROC glad_glPolygonOffsetClamp;
08698 #define glPolygonOffsetClamp glad_glPolygonOffsetClamp
08699 #endif
08700 #ifndef GL_ARB_post_depth_coverage
08701 #define GL_ARB_post_depth_coverage 1
08702 GLAPI int GLAD_GL_ARB_post_depth_coverage;
08703 #endif
08704 #ifndef GL_ARB_program_interface_query
08705 #define GL_ARB_program_interface_query 1
08706 GLAPI int GLAD_GL_ARB_program_interface_query;
08707 typedef void (APIENTRY PFNGLGETPROGRAMINTERFACEIVPROC) (GLuint program, GLenum programInterface,
08708   GLenum pname, GLint *params);
08709 GLAPI PFNGLGETPROGRAMINTERFACEIVPROC glad_glGetProgramInterfaceiv;
08710 #define glGetProgramInterfaceiv glad_glGetProgramInterfaceiv
08711 typedef GLuint (APIENTRY PFNGLGETPROGRAMRESOURCEINDEXPROC) (GLuint program, GLenum programInterface,
08712   const GLchar *name);
08713 GLAPI PFNGLGETPROGRAMRESOURCEINDEXPROC glad_glGetProgramResourceIndex;
08714 #define glGetProgramResourceIndex glad_glGetProgramResourceIndex
08715 typedef void (APIENTRY PFNGLGETPROGRAMRESOURCECNAMEPROC) (GLuint program, GLenum programInterface,
08716   GLuint index, GLsizei bufSize, GLsizei *length, GLchar *name);
08717 GLAPI PFNGLGETPROGRAMRESOURCECNAMEPROC glad_glGetProgramResourceName;
08718 #define glGetProgramResourceName glad_glGetProgramResourceName
08719 typedef void (APIENTRY PFNGLGETPROGRAMRESOURCEIVPROC) (GLuint program, GLenum programInterface, GLuint
08720   index, GLsizei propCount, const GLenum *props, GLsizei count, GLsizei *length, GLint *params);
08721 GLAPI PFNGLGETPROGRAMRESOURCEIVPROC glad_glGetProgramResourceiv;
08722 #define glGetProgramResourceiv glad_glGetProgramResourceiv
08723 typedef GLint (APIENTRY PFNGLGETPROGRAMRESOURCELOCATIONPROC) (GLuint program, GLenum programInterface,
08724   const GLchar *name);
08725 GLAPI PFNGLGETPROGRAMRESOURCELOCATIONPROC glad_glGetProgramResourceLocation;
08726 #define glGetProgramResourceLocation glad_glGetProgramResourceLocation
08727 typedef GLint (APIENTRY PFNGLGETPROGRAMRESOURCELOCATIONINDEXPROC) (GLuint program, GLenum
08728   programInterface, const GLchar *name);
08729 GLAPI PFNGLGETPROGRAMRESOURCELOCATIONINDEXPROC glad_glGetProgramResourceLocationIndex;
08730 #define glGetProgramResourceLocationIndex glad_glGetProgramResourceLocationIndex
08731 #endif
08732 #ifndef GL_ARB_provoking_vertex
08733 #define GL_ARB_provoking_vertex 1
08734 GLAPI int GLAD_GL_ARB_provoking_vertex;
08735 #endif
08736 #ifndef GL_ARB_query_buffer_object
08737 #define GL_ARB_query_buffer_object 1
08738 GLAPI int GLAD_GL_ARB_query_buffer_object;
08739 #endif
08740 #ifndef GL_ARB_robust_buffer_access_behavior
08741 #define GL_ARB_robust_buffer_access_behavior 1
08742 GLAPI int GLAD_GL_ARB_robust_buffer_access_behavior;
08743 #endif
08744 #ifndef GL_ARB_robustness
08745 #define GL_ARB_robustness 1
08746 GLAPI int GLAD_GL_ARB_robustness;
08747 typedef GLenum (APIENTRY PFNGLGETGRAPHICSRESETSTATUSARBPROC) (void);
08748 GLAPI PFNGLGETGRAPHICSRESETSTATUSARBPROC glad_glGetGraphicsResetStatusARB;
08749 #define glGetGraphicsResetStatusARB glad_glGetGraphicsResetStatusARB
08750 typedef void (APIENTRY PFNGLGETNTEXIMAGEARBPROC) (GLenum target, GLint level, GLenum format, GLenum
08751   type, GLsizei bufSize, void *img);
08752 GLAPI PFNGLGETNTEXIMAGEARBPROC glad_glGetnTexImageARB;
08753 #define glGetnTexImageARB glad_glGetnTexImageARB
08754 typedef void (APIENTRY PFNGLREADNPIXELSARBPROC) (GLint x, GLint y, GLsizei width, GLsizei height,
08755   GLenum format, GLenum type, GLsizei bufSize, void *data);
08756 GLAPI PFNGLREADNPIXELSARBPROC glad_glReadnPixelsARB;
08757 #define glGetnPixelsARB glad_glReadnPixelsARB
08758 typedef void (APIENTRY PFNGLGETNCOMPRESSEDTEXIMAGEARBPROC) (GLenum target, GLint lod, GLsizei bufSize,
08759   void *img);
08760 GLAPI PFNGLGETNCOMPRESSEDTEXIMAGEARBPROC glad_glGetnCompressedTexImageARB;
08761 #define glGetnCompressedTexImageARB glad_glGetnCompressedTexImageARB
08762 typedef void (APIENTRY PFNGLGETNUNIFORMFVARBPROC) (GLuint program, GLint location, GLsizei bufSize,
08763   GLfloat *params);
08764 GLAPI PFNGLGETNUNIFORMFVARBPROC glad_glGetnUniformfvARB;
08765 #define glGetnUniformfvARB glad_glGetnUniformfvARB
08766 typedef void (APIENTRY PFNGLGETNUNIFORMIVARBPROC) (GLuint program, GLint location, GLsizei bufSize,
08767   GLint *params);
08768 GLAPI PFNGLGETNUNIFORMIVARBPROC glad_glGetnUniformivARB;
08769 #define glGetnUniformivARB glad_glGetnUniformivARB
08770 typedef void (APIENTRY PFNGLGETNUNIFORMUIVARBPROC) (GLuint program, GLint location, GLsizei bufSize,
08771   GLuint *params);
08772 GLAPI PFNGLGETNUNIFORMUIVARBPROC glad_glGetnUniformuivARB;
08773 #define glGetnUniformuivARB glad_glGetnUniformuivARB
08774 typedef void (APIENTRY PFNGLGETNUNIFORMDVARBPROC) (GLuint program, GLint location, GLsizei bufSize,
08775   GLdouble *params);
08776 GLAPI PFNGLGETNUNIFORMDVARBPROC glad_glGetnUniformdvARB;
08777 #define glGetnUniformdvARB glad_glGetnUniformdvARB
08778 typedef void (APIENTRY PFNGLGETNMAPPDVARBPROC) (GLenum target, GLenum query, GLsizei bufSize, GLdouble
08779   *v);

```



```

08766 GLAPI PFNGLGETNMAPDVARBPROC glad_glGetnMapdvARB;
08767 #define glGetnMapdvARB glad_glGetnMapdvARB
08768 typedef void (APIENTRY PFNGLGETNMAPFVARBPROC) (GLenum target, GLenum query, GLsizei bufSize, GLfloat
    *v);
08769 GLAPI PFNGLGETNMAPFVARBPROC glad_glGetnMapfvARB;
08770 #define glGetnMapfvARB glad_glGetnMapfvARB
08771 typedef void (APIENTRY PFNGLGETNMAPIVARBPROC) (GLenum target, GLenum query, GLsizei bufSize, GLint
    *v);
08772 GLAPI PFNGLGETNMAPIVARBPROC glad_glGetnMapivARB;
08773 #define glGetnMapivARB glad_glGetnMapivARB
08774 typedef void (APIENTRY PFNGLGETNPIXELMAPFVARBPROC) (GLenum map, GLsizei bufSize, GLfloat *values);
08775 GLAPI PFNGLGETNPIXELMAPFVARBPROC glad_glGetnPixelMapfvARB;
08776 #define glGetnPixelMapfvARB glad_glGetnPixelMapfvARB
08777 typedef void (APIENTRY PFNGLGETNPIXELMAPIVARBPROC) (GLenum map, GLsizei bufSize, GLuint *values);
08778 GLAPI PFNGLGETNPIXELMAPIVARBPROC glad_glGetnPixelMapuivARB;
08779 #define glGetnPixelMapuivARB glad_glGetnPixelMapuivARB
08780 typedef void (APIENTRY PFNGLGETNPIXELMAPUSVARBPROC) (GLenum map, GLsizei bufSize, GLushort *values);
08781 GLAPI PFNGLGETNPIXELMAPUSVARBPROC glad_glGetnPixelMapusvARB;
08782 #define glGetnPixelMapusvARB glad_glGetnPixelMapusvARB
08783 typedef void (APIENTRY PFNGLGETNPOLYGONSTIPPLEARBPROC) (GLsizei bufSize, GLubyte *pattern);
08784 GLAPI PFNGLGETNPOLYGONSTIPPLEARBPROC glad_glGetnPolygonStippleARB;
08785 #define glGetnPolygonStippleARB glad_glGetnPolygonStippleARB
08786 typedef void (APIENTRY PFNGLGETNCOLORTABLEARBPROC) (GLenum target, GLenum format, GLenum type, GLsizei
    bufSize, void *table);
08787 GLAPI PFNGLGETNCOLORTABLEARBPROC glad_glGetnColorTableARB;
08788 #define glGetnColorTableARB glad_glGetnColorTableARB
08789 typedef void (APIENTRY PFNGLGETNCONVOLUTIONFILTERARBPROC) (GLenum target, GLenum format, GLenum type,
    GLsizei bufSize, void *image);
08790 GLAPI PFNGLGETNCONVOLUTIONFILTERARBPROC glad_glGetnConvolutionFilterARB;
08791 #define glGetnConvolutionFilterARB glad_glGetnConvolutionFilterARB
08792 typedef void (APIENTRY PFNGLGETNSEPARABLEFILTERARBPROC) (GLenum target, GLenum format, GLenum type,
    GLsizei rowBufSize, void *row, GLsizei columnBufSize, void *column, void *span);
08793 GLAPI PFNGLGETNSEPARABLEFILTERARBPROC glad_glGetnSeparableFilterARB;
08794 #define glGetnSeparableFilterARB glad_glGetnSeparableFilterARB
08795 typedef void (APIENTRY PFNGLGETNHISTOGRAMARBPROC) (GLenum target, GLboolean reset, GLenum format,
    GLenum type, GLsizei bufSize, void *values);
08796 GLAPI PFNGLGETNHISTOGRAMARBPROC glad_glGetnHistogramARB;
08797 #define glGetnHistogramARB glad_glGetnHistogramARB
08798 typedef void (APIENTRY PFNGLGETNMINMAXARBPROC) (GLenum target, GLboolean reset, GLenum format, GLenum
    type, GLsizei bufSize, void *values);
08799 GLAPI PFNGLGETNMINMAXARBPROC glad_glGetnMinmaxARB;
08800 #define glGetnMinmaxARB glad_glGetnMinmaxARB
08801 #endif
08802 #ifndef GL_ARB_robustness_isolation
08803 #define GL_ARB_robustness_isolation 1
08804 GLAPI int GLAD_GL_ARB_robustness_isolation;
08805 #endif
08806 #ifndef GL_ARB_sample_locations
08807 #define GL_ARB_sample_locations 1
08808 GLAPI int GLAD_GL_ARB_sample_locations;
08809 typedef void (APIENTRY PFNGLFRAMEBUFFERSAMPLELOCATIONSFVARBPROC) (GLenum target, GLuint start, GLsizei
    count, const GLfloat *v);
08810 GLAPI PFNGLFRAMEBUFFERSAMPLELOCATIONSFVARBPROC glad_glFramebufferSampleLocationsfvARB;
08811 #define glFramebufferSampleLocationsfvARB glad_glFramebufferSampleLocationsfvARB
08812 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVARBPROC) (GLuint framebuffer, GLuint
    start, GLsizei count, const GLfloat *v);
08813 GLAPI PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVARBPROC glad_glNamedFramebufferSampleLocationsfvARB;
08814 #define glNamedFramebufferSampleLocationsfvARB glad_glNamedFramebufferSampleLocationsfvARB
08815 typedef void (APIENTRY PFNGLEVALUATEDEPTHVALUESARBPROC) (void);
08816 GLAPI PFNGLEVALUATEDEPTHVALUESARBPROC glad_glEvaluateDepthValuesARB;
08817 #define glEvaluateDepthValuesARB glad_glEvaluateDepthValuesARB
08818 #endif
08819 #ifndef GL_ARB_sample_shading
08820 #define GL_ARB_sample_shading 1
08821 GLAPI int GLAD_GL_ARB_sample_shading;
08822 typedef void (APIENTRY PFNGLMINSAMPLESHADINGARBPROC) (GLfloat value);
08823 GLAPI PFNGLMINSAMPLESHADINGARBPROC glad_glMinSampleShadingARB;
08824 #define glMinSampleShadingARB glad_glMinSampleShadingARB
08825 #endif
08826 #ifndef GL_ARB_sampler_objects
08827 #define GL_ARB_sampler_objects 1
08828 GLAPI int GLAD_GL_ARB_sampler_objects;
08829 #endif
08830 #ifndef GL_ARB_seamless_cube_map
08831 #define GL_ARB_seamless_cube_map 1
08832 GLAPI int GLAD_GL_ARB_seamless_cube_map;
08833 #endif
08834 #ifndef GL_ARB_seamless_cubemap_per_texture
08835 #define GL_ARB_seamless_cubemap_per_texture 1
08836 GLAPI int GLAD_GL_ARB_seamless_cubemap_per_texture;
08837 #endif
08838 #ifndef GL_ARB_separate_shader_objects
08839 #define GL_ARB_separate_shader_objects 1
08840 GLAPI int GLAD_GL_ARB_separate_shader_objects;
08841 #endif
08842 #ifndef GL_ARB_shader_atomic_counter_ops
08843 #define GL_ARB_shader_atomic_counter_ops 1

```

```

08844 GLAPI int GLAD_GL_ARB_shader_atomic_counter_ops;
08845 #endif
08846 #ifndef GL_ARB_shader_atomic_counters
08847 #define GL_ARB_shader_atomic_counters 1
08848 GLAPI int GLAD_GL_ARB_shader_atomic_counters;
08849 typedef void (APIENTRY PFNGLGETACTIVEATOMICCOUNTERBUFFERIVPROC) (GLuint program, GLuint bufferIndex,
    GLenum pname, GLint *params);
08850 GLAPI PFNGLGETACTIVEATOMICCOUNTERBUFFERIVPROC glad_glGetActiveAtomicCounterBufferiv;
08851 #define glGetActiveAtomicCounterBufferiv glad_glGetActiveAtomicCounterBufferiv
08852 #endif
08853 #ifndef GL_ARB_shader_ballot
08854 #define GL_ARB_shader_ballot 1
08855 GLAPI int GLAD_GL_ARB_shader_ballot;
08856 #endif
08857 #ifndef GL_ARB_shader_bit_encoding
08858 #define GL_ARB_shader_bit_encoding 1
08859 GLAPI int GLAD_GL_ARB_shader_bit_encoding;
08860 #endif
08861 #ifndef GL_ARB_shader_clock
08862 #define GL_ARB_shader_clock 1
08863 GLAPI int GLAD_GL_ARB_shader_clock;
08864 #endif
08865 #ifndef GL_ARB_shader_draw_parameters
08866 #define GL_ARB_shader_draw_parameters 1
08867 GLAPI int GLAD_GL_ARB_shader_draw_parameters;
08868 #endif
08869 #ifndef GL_ARB_shader_group_vote
08870 #define GL_ARB_shader_group_vote 1
08871 GLAPI int GLAD_GL_ARB_shader_group_vote;
08872 #endif
08873 #ifndef GL_ARB_shader_image_load_store
08874 #define GL_ARB_shader_image_load_store 1
08875 GLAPI int GLAD_GL_ARB_shader_image_load_store;
08876 typedef void (APIENTRY PFNGLBINDIMAGETEXTUREPROC) (GLuint unit, GLuint texture, GLint level, GLboolean
    layered, GLint layer, GLenum access, GLenum format);
08877 GLAPI PFNGLBINDIMAGETEXTUREPROC glad_glBindImageTexture;
08878 #define glBindImageTexture glad_glBindImageTexture
08879 typedef void (APIENTRY PFNGLMEMORYBARRIERPROC) (GLbitfield barriers);
08880 GLAPI PFNGLMEMORYBARRIERPROC glad_glMemoryBarrier;
08881 #define glMemoryBarrier glad_glMemoryBarrier
08882 #endif
08883 #ifndef GL_ARB_shader_image_size
08884 #define GL_ARB_shader_image_size 1
08885 GLAPI int GLAD_GL_ARB_shader_image_size;
08886 #endif
08887 #ifndef GL_ARB_shader_objects
08888 #define GL_ARB_shader_objects 1
08889 GLAPI int GLAD_GL_ARB_shader_objects;
08890 typedef void (APIENTRY PFNGLDELETEOBJECTARBPROC) (GLhandleARB obj);
08891 GLAPI PFNGLDELETEOBJECTARBPROC glad_glDeleteObjectARB;
08892 #define glDeleteObjectARB glad_glDeleteObjectARB
08893 typedef GLhandleARB (APIENTRY PFNGLGETHANDLEARBPROC) (GLenum pname);
08894 GLAPI PFNGLGETHANDLEARBPROC glad_glGetHandleARB;
08895 #define glGetHandleARB glad_glGetHandleARB
08896 typedef void (APIENTRY PFNGLDETACHOBJECTARBPROC) (GLhandleARB containerObj, GLhandleARB attachedObj);
08897 GLAPI PFNGLDETACHOBJECTARBPROC glad_glDetachObjectARB;
08898 #define glDetachObjectARB glad_glDetachObjectARB
08899 typedef GLhandleARB (APIENTRY PFNGLCREATESHADEROBJECTARBPROC) (GLenum shaderType);
08900 GLAPI PFNGLCREATESHADEROBJECTARBPROC glad_glCreateShaderObjectARB;
08901 #define glCreateShaderObjectARB glad_glCreateShaderObjectARB
08902 typedef void (APIENTRY PFNGLSHADERSOURCEARBPROC) (GLhandleARB shaderObj, GLsizei count, const
    GLcharARB **string, const GLint *length);
08903 GLAPI PFNGLSHADERSOURCEARBPROC glad_glShaderSourceARB;
08904 #define glShaderSourceARB glad_glShaderSourceARB
08905 typedef void (APIENTRY PFNGLCOMPILESHADERARBPROC) (GLhandleARB shaderObj);
08906 GLAPI PFNGLCOMPILESHADERARBPROC glad_glCompileShaderARB;
08907 #define glCompileShaderARB glad_glCompileShaderARB
08908 typedef GLhandleARB (APIENTRY PFNGLCREATEPROGRAMOBJECTARBPROC) (void);
08909 GLAPI PFNGLCREATEPROGRAMOBJECTARBPROC glad_glCreateProgramObjectARB;
08910 #define glCreateProgramObjectARB glad_glCreateProgramObjectARB
08911 typedef void (APIENTRY PFNGLATTACHOBJECTARBPROC) (GLhandleARB containerObj, GLhandleARB obj);
08912 GLAPI PFNGLATTACHOBJECTARBPROC glad_glAttachObjectARB;
08913 #define glAttachObjectARB glad_glAttachObjectARB
08914 typedef void (APIENTRY PFNGLLINKPROGRAMARBPROC) (GLhandleARB programObj);
08915 GLAPI PFNGLLINKPROGRAMARBPROC glad_glLinkProgramARB;
08916 #define glLinkProgramARB glad_glLinkProgramARB
08917 typedef void (APIENTRY PFNGLUSEPROGRAMOBJECTARBPROC) (GLhandleARB programObj);
08918 GLAPI PFNGLUSEPROGRAMOBJECTARBPROC glad_glUseProgramObjectARB;
08919 #define glUseProgramObjectARB glad_glUseProgramObjectARB
08920 typedef void (APIENTRY PFNGLVALIDATEPROGRAMARBPROC) (GLhandleARB programObj);
08921 GLAPI PFNGLVALIDATEPROGRAMARBPROC glad_glValidateProgramARB;
08922 #define glValidateProgramARB glad_glValidateProgramARB
08923 typedef void (APIENTRY PFNGLUNIFORM1FARBPROC) (GLint location, GLfloat v0);
08924 GLAPI PFNGLUNIFORM1FARBPROC glad_glUniform1fARB;
08925 #define glUniform1fARB glad_glUniform1fARB
08926 typedef void (APIENTRY PFNGLUNIFORM2FARBPROC) (GLint location, GLfloat v0, GLfloat v1);
08927 GLAPI PFNGLUNIFORM2FARBPROC glad_glUniform2fARB;

```

```

08928 #define glUniform2fARB glad_glUniform2fARB
08929 typedef void (APIENTRY PFNGLUNIFORM3FARBPROC) (GLint location, GLfloat v0, GLfloat v1, GLfloat v2);
08930 GLAPI PFNGLUNIFORM3FARBPROC glad_glUniform3fARB;
08931 #define glUniform3fARB glad_glUniform3fARB
08932 typedef void (APIENTRY PFNGLUNIFORM4FARBPROC) (GLint location, GLfloat v0, GLfloat v1, GLfloat v2,
GLfloat v3);
08933 GLAPI PFNGLUNIFORM4FARBPROC glad_glUniform4fARB;
08934 #define glUniform4fARB glad_glUniform4fARB
08935 typedef void (APIENTRY PFNGLUNIFORM1IARBPROC) (GLint location, GLint v0);
08936 GLAPI PFNGLUNIFORM1IARBPROC glad_glUniform1iARB;
08937 #define glUniform1iARB glad_glUniform1iARB
08938 typedef void (APIENTRY PFNGLUNIFORM2IARBPROC) (GLint location, GLint v0, GLint v1);
08939 GLAPI PFNGLUNIFORM2IARBPROC glad_glUniform2iARB;
08940 #define glUniform2iARB glad_glUniform2iARB
08941 typedef void (APIENTRY PFNGLUNIFORM3IARBPROC) (GLint location, GLint v0, GLint v1, GLint v2);
08942 GLAPI PFNGLUNIFORM3IARBPROC glad_glUniform3iARB;
08943 #define glUniform3iARB glad_glUniform3iARB
08944 typedef void (APIENTRY PFNGLUNIFORM4IARBPROC) (GLint location, GLint v0, GLint v1, GLint v2, GLint
v3);
08945 GLAPI PFNGLUNIFORM4IARBPROC glad_glUniform4iARB;
08946 #define glUniform4iARB glad_glUniform4iARB
08947 typedef void (APIENTRY PFNGLUNIFORM1FVARBPROC) (GLint location, GLsizei count, const GLfloat *value);
08948 GLAPI PFNGLUNIFORM1FVARBPROC glad_glUniform1fvARB;
08949 #define glUniform1fvARB glad_glUniform1fvARB
08950 typedef void (APIENTRY PFNGLUNIFORM2FVARBPROC) (GLint location, GLsizei count, const GLfloat *value);
08951 GLAPI PFNGLUNIFORM2FVARBPROC glad_glUniform2fvARB;
08952 #define glUniform2fvARB glad_glUniform2fvARB
08953 typedef void (APIENTRY PFNGLUNIFORM3FVARBPROC) (GLint location, GLsizei count, const GLfloat *value);
08954 GLAPI PFNGLUNIFORM3FVARBPROC glad_glUniform3fvARB;
08955 #define glUniform3fvARB glad_glUniform3fvARB
08956 typedef void (APIENTRY PFNGLUNIFORM4FVARBPROC) (GLint location, GLsizei count, const GLfloat *value);
08957 GLAPI PFNGLUNIFORM4FVARBPROC glad_glUniform4fvARB;
08958 #define glUniform4fvARB glad_glUniform4fvARB
08959 typedef void (APIENTRY PFNGLUNIFORM1IVARBPROC) (GLint location, GLsizei count, const GLint *value);
08960 GLAPI PFNGLUNIFORM1IVARBPROC glad_glUniform1ivARB;
08961 #define glUniform1ivARB glad_glUniform1ivARB
08962 typedef void (APIENTRY PFNGLUNIFORM2IVARBPROC) (GLint location, GLsizei count, const GLint *value);
08963 GLAPI PFNGLUNIFORM2IVARBPROC glad_glUniform2ivARB;
08964 #define glUniform2ivARB glad_glUniform2ivARB
08965 typedef void (APIENTRY PFNGLUNIFORM3IVARBPROC) (GLint location, GLsizei count, const GLint *value);
08966 GLAPI PFNGLUNIFORM3IVARBPROC glad_glUniform3ivARB;
08967 #define glUniform3ivARB glad_glUniform3ivARB
08968 typedef void (APIENTRY PFNGLUNIFORM4IVARBPROC) (GLint location, GLsizei count, const GLint *value);
08969 GLAPI PFNGLUNIFORM4IVARBPROC glad_glUniform4ivARB;
08970 #define glUniform4ivARB glad_glUniform4ivARB
08971 typedef void (APIENTRY PFNGLUNIFORMMATRIX2FVARBPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
08972 GLAPI PFNGLUNIFORMMATRIX2FVARBPROC glad_glUniformMatrix2fvARB;
08973 #define glUniformMatrix2fvARB glad_glUniformMatrix2fvARB
08974 typedef void (APIENTRY PFNGLUNIFORMMATRIX3FVARBPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
08975 GLAPI PFNGLUNIFORMMATRIX3FVARBPROC glad_glUniformMatrix3fvARB;
08976 #define glUniformMatrix3fvARB glad_glUniformMatrix3fvARB
08977 typedef void (APIENTRY PFNGLUNIFORMMATRIX4FVARBPROC) (GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
08978 GLAPI PFNGLUNIFORMMATRIX4FVARBPROC glad_glUniformMatrix4fvARB;
08979 #define glUniformMatrix4fvARB glad_glUniformMatrix4fvARB
08980 typedef void (APIENTRY PFNGLGETOBJECTPARAMETERFVARBPROC) (GLhandleARB obj, GLenum pname, GLfloat
*params);
08981 GLAPI PFNGLGETOBJECTPARAMETERFVARBPROC glad_glGetObjectParameterfvARB;
08982 #define glGetObjectParameterfvARB glad_glGetObjectParameterfvARB
08983 typedef void (APIENTRY PFNGLGETOBJECTPARAMETERIVARBPROC) (GLhandleARB obj, GLenum pname, GLint
*params);
08984 GLAPI PFNGLGETOBJECTPARAMETERIVARBPROC glad_glGetObjectParameterivARB;
08985 #define glGetObjectParameterivARB glad_glGetObjectParameterivARB
08986 typedef void (APIENTRY PFNGLGETINFOLOGARBPROC) (GLhandleARB obj, GLsizei maxLength, GLsizei *length,
GLcharARB *infoLog);
08987 GLAPI PFNGLGETINFOLOGARBPROC glad_glGetInfoLogARB;
08988 #define glGetInfoLogARB glad_glGetInfoLogARB
08989 typedef void (APIENTRY PFNGLGETATTACHEDOBJECTSARBPROC) (GLhandleARB containerObj, GLsizei maxCount,
GLsizei *count, GLhandleARB *obj);
08990 GLAPI PFNGLGETATTACHEDOBJECTSARBPROC glad_glGetAttachedObjectsARB;
08991 #define glGetAttachedObjectsARB glad_glGetAttachedObjectsARB
08992 typedef void (APIENTRY PFNGLGETUNIFORMLOCATIONARBPROC) (GLhandleARB programObj, const GLcharARB
*name);
08993 GLAPI PFNGLGETUNIFORMLOCATIONARBPROC glad_glGetUniformLocationARB;
08994 #define glGetUniformLocationARB glad_glGetUniformLocationARB
08995 typedef void (APIENTRY PFNGLGETACTIVEUNIFORMARBPROC) (GLhandleARB programObj, GLuint index, GLsizei
maxLength, GLsizei *length, GLint *size, GLenum *type, GLcharARB *name);
08996 GLAPI PFNGLGETACTIVEUNIFORMARBPROC glad_glGetActiveUniformARB;
08997 #define glGetActiveUniformARB glad_glGetActiveUniformARB
08998 typedef void (APIENTRY PFNGLGETUNIFORMFVARBPROC) (GLhandleARB programObj, GLint location, GLfloat
*params);
08999 GLAPI PFNGLGETUNIFORMFVARBPROC glad_glGetUniformfvARB;
09000 #define glGetUniformfvARB glad_glGetUniformfvARB
09001 typedef void (APIENTRY PFNGLGETUNIFORMIVARBPROC) (GLhandleARB programObj, GLint location, GLint
*params);

```

```

09002 GLAPI PFNGLGETUNIFORMIVARBPROC glad_glGetUniformivARB;
09003 #define glGetUniformivARB glad_glGetUniformivARB
09004 typedef void (APIENTRY PFNGLGETSHADERSOURCEARBPROC) (GLhandleARB obj, GLsizei maxLength, GLsizei
    *length, GLcharARB *source);
09005 GLAPI PFNGLGETSHADERSOURCEARBPROC glad_glGetShaderSourceARB;
09006 #define glGetShaderSourceARB glad_glGetShaderSourceARB
09007 #endif
09008 #ifndef GL_ARB_shader_precision
09009 #define GL_ARB_shader_precision 1
09010 GLAPI int GLAD_GL_ARB_shader_precision;
09011 #endif
09012 #ifndef GL_ARB_shader_stencil_export
09013 #define GL_ARB_shader_stencil_export 1
09014 GLAPI int GLAD_GL_ARB_shader_stencil_export;
09015 #endif
09016 #ifndef GL_ARB_shader_storage_buffer_object
09017 #define GL_ARB_shader_storage_buffer_object 1
09018 GLAPI int GLAD_GL_ARB_shader_storage_buffer_object;
09019 typedef void (APIENTRY PFNGLSHADERSTORAGEBLOCKBINDINGPROC) (GLuint program, GLuint storageBlockIndex,
    GLuint storageBlockBinding);
09020 GLAPI PFNGLSHADERSTORAGEBLOCKBINDINGPROC glad_glShaderStorageBlockBinding;
09021 #define glShaderStorageBlockBinding glad_glShaderStorageBlockBinding
09022 #endif
09023 #ifndef GL_ARB_shader_subroutine
09024 #define GL_ARB_shader_subroutine 1
09025 GLAPI int GLAD_GL_ARB_shader_subroutine;
09026 #endif
09027 #ifndef GL_ARB_shader_texture_image_samples
09028 #define GL_ARB_shader_texture_image_samples 1
09029 GLAPI int GLAD_GL_ARB_shader_texture_image_samples;
09030 #endif
09031 #ifndef GL_ARB_shader_texture_lod
09032 #define GL_ARB_shader_texture_lod 1
09033 GLAPI int GLAD_GL_ARB_shader_texture_lod;
09034 #endif
09035 #ifndef GL_ARB_shader_viewport_layer_array
09036 #define GL_ARB_shader_viewport_layer_array 1
09037 GLAPI int GLAD_GL_ARB_shader_viewport_layer_array;
09038 #endif
09039 #ifndef GL_ARB_shading_language_100
09040 #define GL_ARB_shading_language_100 1
09041 GLAPI int GLAD_GL_ARB_shading_language_100;
09042 #endif
09043 #ifndef GL_ARB_shading_language_420pack
09044 #define GL_ARB_shading_language_420pack 1
09045 GLAPI int GLAD_GL_ARB_shading_language_420pack;
09046 #endif
09047 #ifndef GL_ARB_shading_language_include
09048 #define GL_ARB_shading_language_include 1
09049 GLAPI int GLAD_GL_ARB_shading_language_include;
09050 typedef void (APIENTRY PFNGLNAMEDSTRINGARBPROC) (GLenum type, GLint namelen, const GLchar *name, GLint
    stringlen, const GLchar *string);
09051 GLAPI PFNGLNAMEDSTRINGARBPROC glad_glNamedStringARB;
09052 #define glNamedStringARB glad_glNamedStringARB
09053 typedef void (APIENTRY PFNGLDELETENAMEDSTRINGARBPROC) (GLint namelen, const GLchar *name);
09054 GLAPI PFNGLDELETENAMEDSTRINGARBPROC glad_glDeleteNamedStringARB;
09055 #define glDeleteNamedStringARB glad_glDeleteNamedStringARB
09056 typedef void (APIENTRY PFNGLCOMPILESHADERINCLUDEARBPROC) (GLuint shader, GLsizei count, const GLchar
    *const*path, const GLint *length);
09057 GLAPI PFNGLCOMPILESHADERINCLUDEARBPROC glad_glCompileShaderIncludeARB;
09058 #define glCompileShaderIncludeARB glad_glCompileShaderIncludeARB
09059 typedef GLboolean (APIENTRY PFNGLISNAMEDSTRINGARBPROC) (GLint namelen, const GLchar *name);
09060 GLAPI PFNGLISNAMEDSTRINGARBPROC glad_glIsNamedStringARB;
09061 #define glIsNamedStringARB glad_glIsNamedStringARB
09062 typedef void (APIENTRY PFNGLGETNAMEDSTRINGARBPROC) (GLint namelen, const GLchar *name, GLsizei
    bufSize, GLint *stringlen, GLchar *string);
09063 GLAPI PFNGLGETNAMEDSTRINGARBPROC glad_glGetNamedStringARB;
09064 #define glGetNamedStringARB glad_glGetNamedStringARB
09065 typedef void (APIENTRY PFNGLGETNAMEDSTRINGIVARBPROC) (GLint namelen, const GLchar *name, GLenum pname,
    GLint *params);
09066 GLAPI PFNGLGETNAMEDSTRINGIVARBPROC glad_glGetNamedStringivARB;
09067 #define glGetNamedStringivARB glad_glGetNamedStringivARB
09068 #endif
09069 #ifndef GL_ARB_shading_language_packing
09070 #define GL_ARB_shading_language_packing 1
09071 GLAPI int GLAD_GL_ARB_shading_language_packing;
09072 #endif
09073 #ifndef GL_ARB_shadow
09074 #define GL_ARB_shadow 1
09075 GLAPI int GLAD_GL_ARB_shadow;
09076 #endif
09077 #ifndef GL_ARB_shadow_ambient
09078 #define GL_ARB_shadow_ambient 1
09079 GLAPI int GLAD_GL_ARB_shadow_ambient;
09080 #endif
09081 #ifndef GL_ARB_sparse_buffer
09082 #define GL_ARB_sparse_buffer 1

```

```

09083 GLAPI int GLAD_GL_ARB_sparse_buffer;
09084 typedef void (APIENTRY PFNGLBUFFERPAGECOMMITMENTARBPROC) (GLenum target, GLintptr offset, GLsizeiptr
size, GLboolean commit);
09085 GLAPI PFNGLBUFFERPAGECOMMITMENTARBPROC glad_glBufferPageCommitmentARB;
09086 #define glBufferPageCommitmentARB glad_glBufferPageCommitmentARB
09087 typedef void (APIENTRY PFNGLNAMEDBUFFERPAGECOMMITMENTEXTPROC) (GLuint buffer, GLintptr offset,
GLsizeiptr size, GLboolean commit);
09088 GLAPI PFNGLNAMEDBUFFERPAGECOMMITMENTEXTPROC glad_glNamedBufferPageCommitmentEXT;
09089 #define glNamedBufferPageCommitmentEXT glad_glNamedBufferPageCommitmentEXT
09090 typedef void (APIENTRY PFNGLNAMEDBUFFERPAGECOMMITMENTARBPROC) (GLuint buffer, GLintptr offset,
GLsizeiptr size, GLboolean commit);
09091 GLAPI PFNGLNAMEDBUFFERPAGECOMMITMENTARBPROC glad_glNamedBufferPageCommitmentARB;
09092 #define glNamedBufferPageCommitmentARB glad_glNamedBufferPageCommitmentARB
09093 #endif
09094 #ifndef GL_ARB_sparse_texture
09095 #define GL_ARB_sparse_texture 1
09096 GLAPI int GLAD_GL_ARB_sparse_texture;
09097 typedef void (APIENTRY PFNGLTEXPAGECOMMITMENTARBPROC) (GLenum target, GLint level, GLint xoffset,
GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLboolean commit);
09098 GLAPI PFNGLTEXPAGECOMMITMENTARBPROC glad_glTexPageCommitmentARB;
09099 #define glTexPageCommitmentARB glad_glTexPageCommitmentARB
09100 #endif
09101 #ifndef GL_ARB_sparse_texture2
09102 #define GL_ARB_sparse_texture2 1
09103 GLAPI int GLAD_GL_ARB_sparse_texture2;
09104 #endif
09105 #ifndef GL_ARB_sparse_texture_clamp
09106 #define GL_ARB_sparse_texture_clamp 1
09107 GLAPI int GLAD_GL_ARB_sparse_texture_clamp;
09108 #endif
09109 #ifndef GL_ARB_spirv_extensions
09110 #define GL_ARB_spirv_extensions 1
09111 GLAPI int GLAD_GL_ARB_spirv_extensions;
09112 #endif
09113 #ifndef GL_ARB_stencil_texturing
09114 #define GL_ARB_stencil_texturing 1
09115 GLAPI int GLAD_GL_ARB_stencil_texturing;
09116 #endif
09117 #ifndef GL_ARB_sync
09118 #define GL_ARB_sync 1
09119 GLAPI int GLAD_GL_ARB_sync;
09120 #endif
09121 #ifndef GL_ARB_tessellation_shader
09122 #define GL_ARB_tessellation_shader 1
09123 GLAPI int GLAD_GL_ARB_tessellation_shader;
09124 #endif
09125 #ifndef GL_ARB_texture_barrier
09126 #define GL_ARB_texture_barrier 1
09127 GLAPI int GLAD_GL_ARB_texture_barrier;
09128 typedef void (APIENTRY PFNGLTEXTUREBARRIERPROC) (void);
09129 GLAPI PFNGLTEXTUREBARRIERPROC glad_glTextureBarrier;
09130 #define glTextureBarrier glad_glTextureBarrier
09131 #endif
09132 #ifndef GL_ARB_texture_border_clamp
09133 #define GL_ARB_texture_border_clamp 1
09134 GLAPI int GLAD_GL_ARB_texture_border_clamp;
09135 #endif
09136 #ifndef GL_ARB_texture_buffer_object
09137 #define GL_ARB_texture_buffer_object 1
09138 GLAPI int GLAD_GL_ARB_texture_buffer_object;
09139 typedef void (APIENTRY PFNGLTEXBUFFERARBPROC) (GLenum target, GLenum internalformat, GLuint buffer);
09140 GLAPI PFNGLTEXBUFFERARBPROC glad_glTexBufferARB;
09141 #define glTexBufferARB glad_glTexBufferARB
09142 #endif
09143 #ifndef GL_ARB_texture_buffer_object_rgb32
09144 #define GL_ARB_texture_buffer_object_rgb32 1
09145 GLAPI int GLAD_GL_ARB_texture_buffer_object_rgb32;
09146 #endif
09147 #ifndef GL_ARB_texture_buffer_range
09148 #define GL_ARB_texture_buffer_range 1
09149 GLAPI int GLAD_GL_ARB_texture_buffer_range;
09150 typedef void (APIENTRY PFNGLTEXBUFFERRANGEPROC) (GLenum target, GLenum internalformat, GLuint buffer,
GLintptr offset, GLsizeiptr size);
09151 GLAPI PFNGLTEXBUFFERRANGEPROC glad_glTexBufferRange;
09152 #define glTexBufferRange glad_glTexBufferRange
09153 #endif
09154 #ifndef GL_ARB_texture_compression
09155 #define GL_ARB_texture_compression 1
09156 GLAPI int GLAD_GL_ARB_texture_compression;
09157 typedef void (APIENTRY PFNGLCOMPRESSEDTEXIMAGE3DARBPROC) (GLenum target, GLint level, GLenum
internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei imageSize, const
void *data);
09158 GLAPI PFNGLCOMPRESSEDTEXIMAGE3DARBPROC glad_glCompressedTexImage3DARB;
09159 #define glCompressedTexImage3DARB glad_glCompressedTexImage3DARB
09160 typedef void (APIENTRY PFNGLCOMPRESSEDTEXIMAGE2DARBPROC) (GLenum target, GLint level, GLenum
internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const void *data);
09161 GLAPI PFNGLCOMPRESSEDTEXIMAGE2DARBPROC glad_glCompressedTexImage2DARB;

```



```

09162 #define glCompressedTexImage2DARB glad_glCompressedTexImage2DARB
09163 typedef void (APIENTRY PFNGLCOMPRESSEDTEXIMAGE1DARBPROC) (GLenum target, GLint level, GLenum
internalFormat, GLsizei width, GLint border, GLsizei imageSize, const void *data);
09164 GLAPI PFNGLCOMPRESSEDTEXIMAGE1DARBPROC glad_glCompressedTexImage1DARB;
09165 #define glCompressedTexImage1DARB glad_glCompressedTexImage1DARB
09166 typedef void (APIENTRY PFNGLCOMPRESSEDTEXSUBIMAGE3DARBPROC) (GLenum target, GLint level, GLint
xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format,
GLsizei imageSize, const void *data);
09167 GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE3DARBPROC glad_glCompressedTexSubImage3DARB;
09168 #define glCompressedTexSubImage3DARB glad_glCompressedTexSubImage3DARB
09169 typedef void (APIENTRY PFNGLCOMPRESSEDTEXSUBIMAGE2DARBPROC) (GLenum target, GLint level, GLint
xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void
*data);
09170 GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE2DARBPROC glad_glCompressedTexSubImage2DARB;
09171 #define glCompressedTexSubImage2DARB glad_glCompressedTexSubImage2DARB
09172 typedef void (APIENTRY PFNGLCOMPRESSEDTEXSUBIMAGE1DARBPROC) (GLenum target, GLint level, GLint
xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *data);
09173 GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE1DARBPROC glad_glCompressedTexSubImage1DARB;
09174 #define glCompressedTexSubImage1DARB glad_glCompressedTexSubImage1DARB
09175 typedef void (APIENTRY PFNGLGETCOMPRESSEDTEXIMAGEARBPROC) (GLenum target, GLint level, void *img);
09176 GLAPI PFNGLGETCOMPRESSEDTEXIMAGEARBPROC glad_glGetCompressedTexImageARB;
09177 #define glGetCompressedTexImageARB glad_glGetCompressedTexImageARB
09178 #endif
09179 #ifndef GL_ARB_texture_compression_bptc
09180 #define GL_ARB_texture_compression_bptc 1
09181 GLAPI int GLAD_GL_ARB_texture_compression_bptc;
09182 #endif
09183 #ifndef GL_ARB_texture_compression_rgtc
09184 #define GL_ARB_texture_compression_rgtc 1
09185 GLAPI int GLAD_GL_ARB_texture_compression_rgtc;
09186 #endif
09187 #ifndef GL_ARB_texture_cube_map
09188 #define GL_ARB_texture_cube_map 1
09189 GLAPI int GLAD_GL_ARB_texture_cube_map;
09190 #endif
09191 #ifndef GL_ARB_texture_cube_map_array
09192 #define GL_ARB_texture_cube_map_array 1
09193 GLAPI int GLAD_GL_ARB_texture_cube_map_array;
09194 #endif
09195 #ifndef GL_ARB_texture_env_add
09196 #define GL_ARB_texture_env_add 1
09197 GLAPI int GLAD_GL_ARB_texture_env_add;
09198 #endif
09199 #ifndef GL_ARB_texture_env_combine
09200 #define GL_ARB_texture_env_combine 1
09201 GLAPI int GLAD_GL_ARB_texture_env_combine;
09202 #endif
09203 #ifndef GL_ARB_texture_env_crossbar
09204 #define GL_ARB_texture_env_crossbar 1
09205 GLAPI int GLAD_GL_ARB_texture_env_crossbar;
09206 #endif
09207 #ifndef GL_ARB_texture_env_dot3
09208 #define GL_ARB_texture_env_dot3 1
09209 GLAPI int GLAD_GL_ARB_texture_env_dot3;
09210 #endif
09211 #ifndef GL_ARB_texture_filter_anisotropic
09212 #define GL_ARB_texture_filter_anisotropic 1
09213 GLAPI int GLAD_GL_ARB_texture_filter_anisotropic;
09214 #endif
09215 #ifndef GL_ARB_texture_filter_minmax
09216 #define GL_ARB_texture_filter_minmax 1
09217 GLAPI int GLAD_GL_ARB_texture_filter_minmax;
09218 #endif
09219 #ifndef GL_ARB_texture_float
09220 #define GL_ARB_texture_float 1
09221 GLAPI int GLAD_GL_ARB_texture_float;
09222 #endif
09223 #ifndef GL_ARB_texture_gather
09224 #define GL_ARB_texture_gather 1
09225 GLAPI int GLAD_GL_ARB_texture_gather;
09226 #endif
09227 #ifndef GL_ARB_texture_mirror_clamp_to_edge
09228 #define GL_ARB_texture_mirror_clamp_to_edge 1
09229 GLAPI int GLAD_GL_ARB_texture_mirror_clamp_to_edge;
09230 #endif
09231 #ifndef GL_ARB_texture_mirrored_repeat
09232 #define GL_ARB_texture_mirrored_repeat 1
09233 GLAPI int GLAD_GL_ARB_texture_mirrored_repeat;
09234 #endif
09235 #ifndef GL_ARB_texture_multisample
09236 #define GL_ARB_texture_multisample 1
09237 GLAPI int GLAD_GL_ARB_texture_multisample;
09238 #endif
09239 #ifndef GL_ARB_texture_non_power_of_two
09240 #define GL_ARB_texture_non_power_of_two 1
09241 GLAPI int GLAD_GL_ARB_texture_non_power_of_two;
09242 #endif

```

```

09243 #ifndef GL_ARB_texture_query_levels
09244 #define GL_ARB_texture_query_levels 1
09245 GLAPI int GLAD_GL_ARB_texture_query_levels;
09246 #endif
09247 #ifndef GL_ARB_texture_query_lod
09248 #define GL_ARB_texture_query_lod 1
09249 GLAPI int GLAD_GL_ARB_texture_query_lod;
09250 #endif
09251 #ifndef GL_ARB_texture_rectangle
09252 #define GL_ARB_texture_rectangle 1
09253 GLAPI int GLAD_GL_ARB_texture_rectangle;
09254 #endif
09255 #ifndef GL_ARB_texture_rg
09256 #define GL_ARB_texture_rg 1
09257 GLAPI int GLAD_GL_ARB_texture_rg;
09258 #endif
09259 #ifndef GL_ARB_texture_rgb10_a2ui
09260 #define GL_ARB_texture_rgb10_a2ui 1
09261 GLAPI int GLAD_GL_ARB_texture_rgb10_a2ui;
09262 #endif
09263 #ifndef GL_ARB_texture_stencil8
09264 #define GL_ARB_texture_stencil8 1
09265 GLAPI int GLAD_GL_ARB_texture_stencil8;
09266 #endif
09267 #ifndef GL_ARB_texture_storage
09268 #define GL_ARB_texture_storage 1
09269 GLAPI int GLAD_GL_ARB_texture_storage;
09270 typedef void (APIENTRY PFNGLTEXSTORAGE1DPROC) (GLenum target, GLsizei levels, GLenum internalformat,
GLsizei width);
09271 GLAPI PFNGLTEXSTORAGE1DPROC glad_glTexStorage1D;
09272 #define glTexStorage1D glad_glTexStorage1D
09273 typedef void (APIENTRY PFNGLTEXSTORAGE2DPROC) (GLenum target, GLsizei levels, GLenum internalformat,
GLsizei width, GLsizei height);
09274 GLAPI PFNGLTEXSTORAGE2DPROC glad_glTexStorage2D;
09275 #define glTexStorage2D glad_glTexStorage2D
09276 typedef void (APIENTRY PFNGLTEXSTORAGE3DPROC) (GLenum target, GLsizei levels, GLenum internalformat,
GLsizei width, GLsizei height, GLsizei depth);
09277 GLAPI PFNGLTEXSTORAGE3DPROC glad_glTexStorage3D;
09278 #define glTexStorage3D glad_glTexStorage3D
09279 #endif
09280 #ifndef GL_ARB_texture_storage_multisample
09281 #define GL_ARB_texture_storage_multisample 1
09282 GLAPI int GLAD_GL_ARB_texture_storage_multisample;
09283 typedef void (APIENTRY PFNGLTEXSTORAGE2DMULTISAMPLEPROC) (GLenum target, GLsizei samples, GLenum
internalformat, GLsizei width, GLsizei height, GLboolean fixedsamplelocations);
09284 GLAPI PFNGLTEXSTORAGE2DMULTISAMPLEPROC glad_glTexStorage2DMultisample;
09285 #define glTexStorage2DMultisample glad_glTexStorage2DMultisample
09286 typedef void (APIENTRY PFNGLTEXSTORAGE3DMULTISAMPLEPROC) (GLenum target, GLsizei samples, GLenum
internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedsamplelocations);
09287 GLAPI PFNGLTEXSTORAGE3DMULTISAMPLEPROC glad_glTexStorage3DMultisample;
09288 #define glTexStorage3DMultisample glad_glTexStorage3DMultisample
09289 #endif
09290 #ifndef GL_ARB_texture_swizzle
09291 #define GL_ARB_texture_swizzle 1
09292 GLAPI int GLAD_GL_ARB_texture_swizzle;
09293 #endif
09294 #ifndef GL_ARB_texture_view
09295 #define GL_ARB_texture_view 1
09296 GLAPI int GLAD_GL_ARB_texture_view;
09297 typedef void (APIENTRY PFNGLTEXTUREVIEWPROC) (GLuint texture, GLenum target, GLuint origtexture,
GLenum internalformat, GLuint minlevel, GLuint numlevels, GLuint minlayer, GLuint numlayers);
09298 GLAPI PFNGLTEXTUREVIEWPROC glad_glTextureView;
09299 #define glTextureView glad_glTextureView
09300 #endif
09301 #ifndef GL_ARB_timer_query
09302 #define GL_ARB_timer_query 1
09303 GLAPI int GLAD_GL_ARB_timer_query;
09304 #endif
09305 #ifndef GL_ARB_transform_feedback2
09306 #define GL_ARB_transform_feedback2 1
09307 GLAPI int GLAD_GL_ARB_transform_feedback2;
09308 #endif
09309 #ifndef GL_ARB_transform_feedback3
09310 #define GL_ARB_transform_feedback3 1
09311 GLAPI int GLAD_GL_ARB_transform_feedback3;
09312 #endif
09313 #ifndef GL_ARB_transform_feedback_instanced
09314 #define GL_ARB_transform_feedback_instanced 1
09315 GLAPI int GLAD_GL_ARB_transform_feedback_instanced;
09316 typedef void (APIENTRY PFNGLDRAWTRANSFORMFEEDBACKINSTANCEDPROC) (GLenum mode, GLuint id, GLsizei
instancecount);
09317 GLAPI PFNGLDRAWTRANSFORMFEEDBACKINSTANCEDPROC glad_glDrawTransformFeedbackInstanced;
09318 #define glDrawTransformFeedbackInstanced glad_glDrawTransformFeedbackInstanced
09319 typedef void (APIENTRY PFNGLDRAWTRANSFORMFEEDBACKSTREAMINSTANCEDPROC) (GLenum mode, GLuint id, GLuint
stream, GLsizei instancecount);
09320 GLAPI PFNGLDRAWTRANSFORMFEEDBACKSTREAMINSTANCEDPROC glad_glDrawTransformFeedbackStreamInstanced;
09321 #define glDrawTransformFeedbackStreamInstanced glad_glDrawTransformFeedbackStreamInstanced

```

```

09322 #endif
09323 #ifndef GL_ARB_transform_feedback_overflow_query
09324 #define GL_ARB_transform_feedback_overflow_query 1
09325 GLAPI int GLAD_GL_ARB_transform_feedback_overflow_query;
09326 #endif
09327 #ifndef GL_ARB_transpose_matrix
09328 #define GL_ARB_transpose_matrix 1
09329 GLAPI int GLAD_GL_ARB_transpose_matrix;
09330 typedef void (APIENTRY PFNGLLOADTRANSPOSEMATRIXFARBPROC) (const GLfloat *m);
09331 GLAPI PFNGLLOADTRANSPOSEMATRIXFARBPROC glad_glLoadTransposeMatrixfARB;
09332 #define glLoadTransposeMatrixfARB glad_glLoadTransposeMatrixfARB
09333 typedef void (APIENTRY PFNGLLOADTRANSPOSEMATRIXDARBPROC) (const GLdouble *m);
09334 GLAPI PFNGLLOADTRANSPOSEMATRIXDARBPROC glad_glLoadTransposeMatrixdARB;
09335 #define glLoadTransposeMatrixdARB glad_glLoadTransposeMatrixdARB
09336 typedef void (APIENTRY PFNGLMULTTRANSPOSEMATRIXFARBPROC) (const GLfloat *m);
09337 GLAPI PFNGLMULTTRANSPOSEMATRIXFARBPROC glad_glMultTransposeMatrixfARB;
09338 #define glMultTransposeMatrixfARB glad_glMultTransposeMatrixfARB
09339 typedef void (APIENTRY PFNGLMULTTRANSPOSEMATRIXDARBPROC) (const GLdouble *m);
09340 GLAPI PFNGLMULTTRANSPOSEMATRIXDARBPROC glad_glMultTransposeMatrixdARB;
09341 #define glMultTransposeMatrixdARB glad_glMultTransposeMatrixdARB
09342 #endif
09343 #ifndef GL_ARB_uniform_buffer_object
09344 #define GL_ARB_uniform_buffer_object 1
09345 GLAPI int GLAD_GL_ARB_uniform_buffer_object;
09346 #endif
09347 #ifndef GL_ARB_vertex_array_bgra
09348 #define GL_ARB_vertex_array_bgra 1
09349 GLAPI int GLAD_GL_ARB_vertex_array_bgra;
09350 #endif
09351 #ifndef GL_ARB_vertex_array_object
09352 #define GL_ARB_vertex_array_object 1
09353 GLAPI int GLAD_GL_ARB_vertex_array_object;
09354 #endif
09355 #ifndef GL_ARB_vertex_attrib_64bit
09356 #define GL_ARB_vertex_attrib_64bit 1
09357 GLAPI int GLAD_GL_ARB_vertex_attrib_64bit;
09358 #endif
09359 #ifndef GL_ARB_vertex_attrib_binding
09360 #define GL_ARB_vertex_attrib_binding 1
09361 GLAPI int GLAD_GL_ARB_vertex_attrib_binding;
09362 typedef void (APIENTRY PFNGLBINDVERTEXBUFFERPROC) (GLuint bindingindex, GLuint buffer, GLintptr
offset, GLsizei stride);
09363 GLAPI PFNGLBINDVERTEXBUFFERPROC glad_glBindVertexBuffer;
09364 #define glBindVertexBuffer glad_glBindVertexBuffer
09365 typedef void (APIENTRY PFNGLVERTEXATTRIBFORMATPROC) (GLuint attribindex, GLint size, GLenum type,
GLboolean normalized, GLuint relativeoffset);
09366 GLAPI PFNGLVERTEXATTRIBFORMATPROC glad_glVertexAttribFormat;
09367 #define glVertexAttribFormat glad_glVertexAttribFormat
09368 typedef void (APIENTRY PFNGLVERTEXATTRIBIFORMATPROC) (GLuint attribindex, GLint size, GLenum type,
GLuint relativeoffset);
09369 GLAPI PFNGLVERTEXATTRIBIFORMATPROC glad_glVertexAttribIFormat;
09370 #define glVertexAttribIFormat glad_glVertexAttribIFormat
09371 typedef void (APIENTRY PFNGLVERTEXATTRIBLFORMATPROC) (GLuint attribindex, GLint size, GLenum type,
GLuint relativeoffset);
09372 GLAPI PFNGLVERTEXATTRIBLFORMATPROC glad_glVertexAttribLFormat;
09373 #define glVertexAttribLFormat glad_glVertexAttribLFormat
09374 typedef void (APIENTRY PFNGLVERTEXATTRIBBINDINGPROC) (GLuint attribindex, GLuint bindingindex);
09375 GLAPI PFNGLVERTEXATTRIBBINDINGPROC glad_glVertexAttribBinding;
09376 #define glVertexAttribBinding glad_glVertexAttribBinding
09377 typedef void (APIENTRY PFNGLVERTEXBINDINGDIVISORPROC) (GLuint bindingindex, GLuint divisor);
09378 GLAPI PFNGLVERTEXBINDINGDIVISORPROC glad_glVertexBindingDivisor;
09379 #define glVertexBindingDivisor glad_glVertexBindingDivisor
09380 #endif
09381 #ifndef GL_ARB_vertex_blend
09382 #define GL_ARB_vertex_blend 1
09383 GLAPI int GLAD_GL_ARB_vertex_blend;
09384 typedef void (APIENTRY PFNGLWEIGHTBVARBPROC) (GLint size, const GLbyte *weights);
09385 GLAPI PFNGLWEIGHTBVARBPROC glad_glWeightbvARB;
09386 #define glWeightbvARB glad_glWeightbvARB
09387 typedef void (APIENTRY PFNGLWEIGHTSVARBPROC) (GLint size, const GLshort *weights);
09388 GLAPI PFNGLWEIGHTSVARBPROC glad_glWeightsvARB;
09389 #define glWeightsvARB glad_glWeightsvARB
09390 typedef void (APIENTRY PFNGLWEIGHTIVARBPROC) (GLint size, const GLint *weights);
09391 GLAPI PFNGLWEIGHTIVARBPROC glad_glWeightivARB;
09392 #define glWeightivARB glad_glWeightivARB
09393 typedef void (APIENTRY PFNGLWEIGHTFVARBPROC) (GLint size, const GLfloat *weights);
09394 GLAPI PFNGLWEIGHTFVARBPROC glad_glWeightfvARB;
09395 #define glWeightfvARB glad_glWeightfvARB
09396 typedef void (APIENTRY PFNGLWEIGHTDVARBPROC) (GLint size, const GLdouble *weights);
09397 GLAPI PFNGLWEIGHTDVARBPROC glad_glWeightdvARB;
09398 #define glWeightdvARB glad_glWeightdvARB
09399 typedef void (APIENTRY PFNGLWEIGHTUBVARBPROC) (GLint size, const GLubyte *weights);
09400 GLAPI PFNGLWEIGHTUBVARBPROC glad_glWeightubvARB;
09401 #define glWeightubvARB glad_glWeightubvARB
09402 typedef void (APIENTRY PFNGLWEIGHTUSVARBPROC) (GLint size, const GLushort *weights);
09403 GLAPI PFNGLWEIGHTUSVARBPROC glad_glWeightusvARB;
09404 #define glWeightusvARB glad_glWeightusvARB

```



```

09405 typedef void (APIENTRY PFNGLWEIGHTUIVARBPROC) (GLint size, const GLuint *weights);
09406 GLAPI PFNGLWEIGHTUIVARBPROC glad_glWeightuiVARB;
09407 #define glWeightuiVARB glad_glWeightuiVARB
09408 typedef void (APIENTRY PFNGLWEIGHTPOINTERARBPROC) (GLint size, GLenum type, GLsizei stride, const void
*pointer);
09409 GLAPI PFNGLWEIGHTPOINTERARBPROC glad_glWeightPointerARB;
09410 #define glWeightPointerARB glad_glWeightPointerARB
09411 typedef void (APIENTRY PFNGLVERTEXBLENDARBPROC) (GLint count);
09412 GLAPI PFNGLVERTEXBLENDARBPROC glad_glVertexBlendARB;
09413 #define glVertexBlendARB glad_glVertexBlendARB
09414 #endif
09415 #ifndef GL_ARB_vertex_buffer_object
09416 #define GL_ARB_vertex_buffer_object 1
09417 GLAPI int GLAD_GL_ARB_vertex_buffer_object;
09418 typedef void (APIENTRY PFNGLBINDBUFFERARBPROC) (GLenum target, GLuint buffer);
09419 GLAPI PFNGLBINDBUFFERARBPROC glad_glBindBufferARB;
09420 #define glBindBufferARB glad_glBindBufferARB
09421 typedef void (APIENTRY PFNGLDELETEBUFFERSARBPROC) (GLsizei n, const GLuint *buffers);
09422 GLAPI PFNGLDELETEBUFFERSARBPROC glad_glDeleteBuffersARB;
09423 #define glDeleteBuffersARB glad_glDeleteBuffersARB
09424 typedef void (APIENTRY PFNGLGENBUFFERSARBPROC) (GLsizei n, GLuint *buffers);
09425 GLAPI PFNGLGENBUFFERSARBPROC glad_glGenBuffersARB;
09426 #define glGenBuffersARB glad_glGenBuffersARB
09427 typedef GLboolean (APIENTRY PFNGLISBUFFERARBPROC) (GLuint buffer);
09428 GLAPI PFNGLISBUFFERARBPROC glad_glIsBufferARB;
09429 #define glIsBufferARB glad_glIsBufferARB
09430 typedef void (APIENTRY PFNGLBUFFERDATAARBPROC) (GLenum target, GLsizeiptrARB size, const void *data,
GLenum usage);
09431 GLAPI PFNGLBUFFERDATAARBPROC glad_glBufferDataARB;
09432 #define glBufferDataARB glad_glBufferDataARB
09433 typedef void (APIENTRY PFNGLBUFFERSUBDATAARBPROC) (GLenum target, GLintptrARB offset, GLsizeiptrARB
size, const void *data);
09434 GLAPI PFNGLBUFFERSUBDATAARBPROC glad_glBufferSubDataARB;
09435 #define glBufferSubDataARB glad_glBufferSubDataARB
09436 typedef void (APIENTRY PFNGLGETBUFFERSUBDATAARBPROC) (GLenum target, GLintptrARB offset, GLsizeiptrARB
size, void *data);
09437 GLAPI PFNGLGETBUFFERSUBDATAARBPROC glad_glGetBufferSubDataARB;
09438 #define glGetBufferSubDataARB glad_glGetBufferSubDataARB
09439 typedef void * (APIENTRY PFNGLMAPBUFFERARBPROC) (GLenum target, GLenum access);
09440 GLAPI PFNGLMAPBUFFERARBPROC glad_glMapBufferARB;
09441 #define glMapBufferARB glad_glMapBufferARB
09442 typedef GLboolean (APIENTRY PFNGLUNMAPBUFFERARBPROC) (GLenum target);
09443 GLAPI PFNGLUNMAPBUFFERARBPROC glad_glUnmapBufferARB;
09444 #define glUnmapBufferARB glad_glUnmapBufferARB
09445 typedef void (APIENTRY PFNGLGETBUFFERPARAMETERIVARBPROC) (GLenum target, GLenum pname, GLint *params);
09446 GLAPI PFNGLGETBUFFERPARAMETERIVARBPROC glad_glGetBufferParameterivARB;
09447 #define glGetBufferParameterivARB glad_glGetBufferParameterivARB
09448 typedef void (APIENTRY PFNGLGETBUFFERPOINTERVARBPROC) (GLenum target, GLenum pname, void **params);
09449 GLAPI PFNGLGETBUFFERPOINTERVARBPROC glad_glGetBufferPointervARB;
09450 #define glGetBufferPointervARB glad_glGetBufferPointervARB
09451 #endif
09452 #ifndef GL_ARB_vertex_program
09453 #define GL_ARB_vertex_program 1
09454 GLAPI int GLAD_GL_ARB_vertex_program;
09455 typedef void (APIENTRY PFNGLVERTEXATTRIB1DARBPROC) (GLuint index, GLdouble x);
09456 GLAPI PFNGLVERTEXATTRIB1DARBPROC glad_glVertexAttrib1dARB;
09457 #define glVertexAttrib1dARB glad_glVertexAttrib1dARB
09458 typedef void (APIENTRY PFNGLVERTEXATTRIB1DVARBPROC) (GLuint index, const GLdouble *v);
09459 GLAPI PFNGLVERTEXATTRIB1DVARBPROC glad_glVertexAttrib1dvarb;
09460 #define glVertexAttrib1dvarb glad_glVertexAttrib1dvarb
09461 typedef void (APIENTRY PFNGLVERTEXATTRIB1FARBPROC) (GLuint index, GLfloat x);
09462 GLAPI PFNGLVERTEXATTRIB1FARBPROC glad_glVertexAttrib1fARB;
09463 #define glVertexAttrib1fARB glad_glVertexAttrib1fARB
09464 typedef void (APIENTRY PFNGLVERTEXATTRIB1FVARBPROC) (GLuint index, const GLfloat *v);
09465 GLAPI PFNGLVERTEXATTRIB1FVARBPROC glad_glVertexAttrib1fvARB;
09466 #define glVertexAttrib1fvARB glad_glVertexAttrib1fvARB
09467 typedef void (APIENTRY PFNGLVERTEXATTRIB1SARBPROC) (GLuint index, GLshort x);
09468 GLAPI PFNGLVERTEXATTRIB1SARBPROC glad_glVertexAttrib1sARB;
09469 #define glVertexAttrib1sARB glad_glVertexAttrib1sARB
09470 typedef void (APIENTRY PFNGLVERTEXATTRIB1SVARBPROC) (GLuint index, const GLshort *v);
09471 GLAPI PFNGLVERTEXATTRIB1SVARBPROC glad_glVertexAttrib1svARB;
09472 #define glVertexAttrib1svARB glad_glVertexAttrib1svARB
09473 typedef void (APIENTRY PFNGLVERTEXATTRIB2DARBPROC) (GLuint index, GLdouble x, GLdouble y);
09474 GLAPI PFNGLVERTEXATTRIB2DARBPROC glad_glVertexAttrib2dARB;
09475 #define glVertexAttrib2dARB glad_glVertexAttrib2dARB
09476 typedef void (APIENTRY PFNGLVERTEXATTRIB2DVARBPROC) (GLuint index, const GLdouble *v);
09477 GLAPI PFNGLVERTEXATTRIB2DVARBPROC glad_glVertexAttrib2dvarb;
09478 #define glVertexAttrib2dvarb glad_glVertexAttrib2dvarb
09479 typedef void (APIENTRY PFNGLVERTEXATTRIB2FARBPROC) (GLuint index, GLfloat x, GLfloat y);
09480 GLAPI PFNGLVERTEXATTRIB2FARBPROC glad_glVertexAttrib2fARB;
09481 #define glVertexAttrib2fARB glad_glVertexAttrib2fARB
09482 typedef void (APIENTRY PFNGLVERTEXATTRIB2FVARBPROC) (GLuint index, const GLfloat *v);
09483 GLAPI PFNGLVERTEXATTRIB2FVARBPROC glad_glVertexAttrib2fvARB;
09484 #define glVertexAttrib2fvARB glad_glVertexAttrib2fvARB
09485 typedef void (APIENTRY PFNGLVERTEXATTRIB2SARBPROC) (GLuint index, GLshort x, GLshort y);
09486 GLAPI PFNGLVERTEXATTRIB2SARBPROC glad_glVertexAttrib2sARB;
09487 #define glVertexAttrib2sARB glad_glVertexAttrib2sARB

```

```

09488 typedef void (APIENTRY PFNGLVERTEXATTRIB2SVARBPROC) (GLuint index, const GLshort *v);
09489 GLAPI PFNGLVERTEXATTRIB2SVARBPROC glad_glVertexAttrib2svARB;
09490 #define glVertexAttrib2svARB glad_glVertexAttrib2svARB
09491 typedef void (APIENTRY PFNGLVERTEXATTRIB3DARBPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble z);
09492 GLAPI PFNGLVERTEXATTRIB3DARBPROC glad_glVertexAttrib3dARB;
09493 #define glVertexAttrib3dARB glad_glVertexAttrib3dARB
09494 typedef void (APIENTRY PFNGLVERTEXATTRIB3DVARBPROC) (GLuint index, const GLdouble *v);
09495 GLAPI PFNGLVERTEXATTRIB3DVARBPROC glad_glVertexAttrib3dvARB;
09496 #define glVertexAttrib3dvARB glad_glVertexAttrib3dvARB
09497 typedef void (APIENTRY PFNGLVERTEXATTRIB3FARBPROC) (GLuint index, GLfloat x, GLfloat y, GLfloat z);
09498 GLAPI PFNGLVERTEXATTRIB3FARBPROC glad_glVertexAttrib3fARB;
09499 #define glVertexAttrib3fARB glad_glVertexAttrib3fARB
09500 typedef void (APIENTRY PFNGLVERTEXATTRIB3FVARBPROC) (GLuint index, const GLfloat *v);
09501 GLAPI PFNGLVERTEXATTRIB3FVARBPROC glad_glVertexAttrib3fvARB;
09502 #define glVertexAttrib3fvARB glad_glVertexAttrib3fvARB
09503 typedef void (APIENTRY PFNGLVERTEXATTRIB3SARBPROC) (GLuint index, GLshort x, GLshort y, GLshort z);
09504 GLAPI PFNGLVERTEXATTRIB3SARBPROC glad_glVertexAttrib3sARB;
09505 #define glVertexAttrib3sARB glad_glVertexAttrib3sARB
09506 typedef void (APIENTRY PFNGLVERTEXATTRIB3SVARBPROC) (GLuint index, const GLshort *v);
09507 GLAPI PFNGLVERTEXATTRIB3SVARBPROC glad_glVertexAttrib3svARB;
09508 #define glVertexAttrib3svARB glad_glVertexAttrib3svARB
09509 typedef void (APIENTRY PFNGLVERTEXATTRIB4NBVARBPROC) (GLuint index, const GLbyte *v);
09510 GLAPI PFNGLVERTEXATTRIB4NBVARBPROC glad_glVertexAttrib4NbvARB;
09511 #define glVertexAttrib4NbvARB glad_glVertexAttrib4NbvARB
09512 typedef void (APIENTRY PFNGLVERTEXATTRIB4NIVARBPROC) (GLuint index, const GLint *v);
09513 GLAPI PFNGLVERTEXATTRIB4NIVARBPROC glad_glVertexAttrib4NivARB;
09514 #define glVertexAttrib4NivARB glad_glVertexAttrib4NivARB
09515 typedef void (APIENTRY PFNGLVERTEXATTRIB4NSVARBPROC) (GLuint index, const GLshort *v);
09516 GLAPI PFNGLVERTEXATTRIB4NSVARBPROC glad_glVertexAttrib4NsvARB;
09517 #define glVertexAttrib4NsvARB glad_glVertexAttrib4NsvARB
09518 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUBVARBPROC) (GLuint index, GLubyte x, GLubyte y, GLubyte z,
GLubyte w);
09519 GLAPI PFNGLVERTEXATTRIB4NUBVARBPROC glad_glVertexAttrib4NubARB;
09520 #define glVertexAttrib4NubARB glad_glVertexAttrib4NubARB
09521 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUBVARBPROC) (GLuint index, const GLubyte *v);
09522 GLAPI PFNGLVERTEXATTRIB4NUBVARBPROC glad_glVertexAttrib4NubvARB;
09523 #define glVertexAttrib4NubvARB glad_glVertexAttrib4NubvARB
09524 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUIVARBPROC) (GLuint index, const GLuint *v);
09525 GLAPI PFNGLVERTEXATTRIB4NUIVARBPROC glad_glVertexAttrib4NuivARB;
09526 #define glVertexAttrib4NuivARB glad_glVertexAttrib4NuivARB
09527 typedef void (APIENTRY PFNGLVERTEXATTRIB4NUSVARBPROC) (GLuint index, const GLushort *v);
09528 GLAPI PFNGLVERTEXATTRIB4NUSVARBPROC glad_glVertexAttrib4NusvARB;
09529 #define glVertexAttrib4NusvARB glad_glVertexAttrib4NusvARB
09530 typedef void (APIENTRY PFNGLVERTEXATTRIB4BVARBPROC) (GLuint index, const GLbyte *v);
09531 GLAPI PFNGLVERTEXATTRIB4BVARBPROC glad_glVertexAttrib4bvARB;
09532 #define glVertexAttrib4bvARB glad_glVertexAttrib4bvARB
09533 typedef void (APIENTRY PFNGLVERTEXATTRIB4DARBPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble z,
GLdouble w);
09534 GLAPI PFNGLVERTEXATTRIB4DARBPROC glad_glVertexAttrib4dARB;
09535 #define glVertexAttrib4dARB glad_glVertexAttrib4dARB
09536 typedef void (APIENTRY PFNGLVERTEXATTRIB4DVARBPROC) (GLuint index, const GLdouble *v);
09537 GLAPI PFNGLVERTEXATTRIB4DVARBPROC glad_glVertexAttrib4dvARB;
09538 #define glVertexAttrib4dvARB glad_glVertexAttrib4dvARB
09539 typedef void (APIENTRY PFNGLVERTEXATTRIB4FARBPROC) (GLuint index, GLfloat x, GLfloat y, GLfloat z,
GLfloat w);
09540 GLAPI PFNGLVERTEXATTRIB4FARBPROC glad_glVertexAttrib4fARB;
09541 #define glVertexAttrib4fARB glad_glVertexAttrib4fARB
09542 typedef void (APIENTRY PFNGLVERTEXATTRIB4FVARBPROC) (GLuint index, const GLfloat *v);
09543 GLAPI PFNGLVERTEXATTRIB4FVARBPROC glad_glVertexAttrib4fvARB;
09544 #define glVertexAttrib4fvARB glad_glVertexAttrib4fvARB
09545 typedef void (APIENTRY PFNGLVERTEXATTRIB4IVARBPROC) (GLuint index, const GLint *v);
09546 GLAPI PFNGLVERTEXATTRIB4IVARBPROC glad_glVertexAttrib4ivARB;
09547 #define glVertexAttrib4ivARB glad_glVertexAttrib4ivARB
09548 typedef void (APIENTRY PFNGLVERTEXATTRIB4SARBPROC) (GLuint index, GLshort x, GLshort y, GLshort z,
GLshort w);
09549 GLAPI PFNGLVERTEXATTRIB4SARBPROC glad_glVertexAttrib4sARB;
09550 #define glVertexAttrib4sARB glad_glVertexAttrib4sARB
09551 typedef void (APIENTRY PFNGLVERTEXATTRIB4SVARBPROC) (GLuint index, const GLshort *v);
09552 GLAPI PFNGLVERTEXATTRIB4SVARBPROC glad_glVertexAttrib4svARB;
09553 #define glVertexAttrib4svARB glad_glVertexAttrib4svARB
09554 typedef void (APIENTRY PFNGLVERTEXATTRIB4UBVARBPROC) (GLuint index, const GLubyte *v);
09555 GLAPI PFNGLVERTEXATTRIB4UBVARBPROC glad_glVertexAttrib4ubvARB;
09556 #define glVertexAttrib4ubvARB glad_glVertexAttrib4ubvARB
09557 typedef void (APIENTRY PFNGLVERTEXATTRIB4UIVARBPROC) (GLuint index, const GLuint *v);
09558 GLAPI PFNGLVERTEXATTRIB4UIVARBPROC glad_glVertexAttrib4uivARB;
09559 #define glVertexAttrib4uivARB glad_glVertexAttrib4uivARB
09560 typedef void (APIENTRY PFNGLVERTEXATTRIB4USVARBPROC) (GLuint index, const GLushort *v);
09561 GLAPI PFNGLVERTEXATTRIB4USVARBPROC glad_glVertexAttrib4usvARB;
09562 #define glVertexAttrib4usvARB glad_glVertexAttrib4usvARB
09563 typedef void (APIENTRY PFNGLVERTEXATTRIBPOINTERARBPROC) (GLuint index, GLint size, GLenum type,
GLboolean normalized, GLsizei stride, const void *pointer);
09564 GLAPI PFNGLVERTEXATTRIBPOINTERARBPROC glad_glVertexAttribPointerARB;
09565 #define glVertexAttribPointerARB glad_glVertexAttribPointerARB
09566 typedef void (APIENTRY PFNGLVERTEXATTRIBARRAYARBPROC) (GLuint index);
09567 GLAPI PFNGLVERTEXATTRIBARRAYARBPROC glad_glEnableVertexAttribArrayARB;
09568 #define glEnableVertexAttribArrayARB glad_glEnableVertexAttribArrayARB
09569 typedef void (APIENTRY PFNGLDISABLEVERTEXATTRIBARRAYARBPROC) (GLuint index);

```

```

09570 GLAPI PFNGLDISABLEVERTEXATTRIBARRAYARBPROC glad_glDisableVertexArrayARB;
09571 #define glDisableVertexArrayARB glad_glDisableVertexArrayARB
09572 typedef void (APIENTRY PFNGLGETVERTEXATTRIBDVARBPROC) (GLuint index, GLenum pname, GLdouble *params);
09573 GLAPI PFNGLGETVERTEXATTRIBDVARBPROC glad_glGetVertexAttribdvARB;
09574 #define glGetVertexAttribdvARB glad_glGetVertexAttribdvARB
09575 typedef void (APIENTRY PFNGLGETVERTEXATTRIBFVARBPROC) (GLuint index, GLenum pname, GLfloat *params);
09576 GLAPI PFNGLGETVERTEXATTRIBFVARBPROC glad_glGetVertexAttribfvARB;
09577 #define glGetVertexAttribfvARB glad_glGetVertexAttribfvARB
09578 typedef void (APIENTRY PFNGLGETVERTEXATTRIBIVARBPROC) (GLuint index, GLenum pname, GLint *params);
09579 GLAPI PFNGLGETVERTEXATTRIBIVARBPROC glad_glGetVertexAttribivARB;
09580 #define glGetVertexAttribivARB glad_glGetVertexAttribivARB
09581 typedef void (APIENTRY PFNGLGETVERTEXATTRIBPOINTERVARBPROC) (GLuint index, GLenum pname, void
**pointer);
09582 GLAPI PFNGLGETVERTEXATTRIBPOINTERVARBPROC glad_glGetVertexAttribPointervARB;
09583 #define glGetVertexAttribPointervARB glad_glGetVertexAttribPointervARB
09584 #endif
09585 #ifndef GL_ARB_vertex_shader
09586 #define GL_ARB_vertex_shader 1
09587 GLAPI int GLAD_GL_ARB_vertex_shader;
09588 typedef void (APIENTRY PFNGLBINDATTRIBLOCATIONARBPROC) (GLhandleARB programObj, GLuint index, const
GLcharARB *name);
09589 GLAPI PFNGLBINDATTRIBLOCATIONARBPROC glad_glBindAttribLocationARB;
09590 #define glBindAttribLocationARB glad_glBindAttribLocationARB
09591 typedef void (APIENTRY PFNGLGETACTIVEATTRIBARBPROC) (GLhandleARB programObj, GLuint index, GLsizei
maxLength, GLsizei *length, GLint *size, GLenum *type, GLcharARB *name);
09592 GLAPI PFNGLGETACTIVEATTRIBARBPROC glad_glGetActiveAttribARB;
09593 #define glGetActiveAttribARB glad_glGetActiveAttribARB
09594 typedef GLint (APIENTRY PFNGLGETATTRIBLOCATIONARBPROC) (GLhandleARB programObj, const GLcharARB
*name);
09595 GLAPI PFNGLGETATTRIBLOCATIONARBPROC glad_glGetAttribLocationARB;
09596 #define glGetAttribLocationARB glad_glGetAttribLocationARB
09597 #endif
09598 #ifndef GL_ARB_vertex_type_10f_11f_11f_rev
09599 #define GL_ARB_vertex_type_10f_11f_11f_rev 1
09600 GLAPI int GLAD_GL_ARB_vertex_type_10f_11f_11f_rev;
09601 #endif
09602 #ifndef GL_ARB_vertex_type_2_10_10_10_rev
09603 #define GL_ARB_vertex_type_2_10_10_10_rev 1
09604 GLAPI int GLAD_GL_ARB_vertex_type_2_10_10_10_rev;
09605 #endif
09606 #ifndef GL_ARB_viewport_array
09607 #define GL_ARB_viewport_array 1
09608 GLAPI int GLAD_GL_ARB_viewport_array;
09609 typedef void (APIENTRY PFNGLDEPTHRANGEARRAYDVNPVPROC) (GLuint first, GLsizei count, const GLdouble *v);
09610 GLAPI PFNGLDEPTHRANGEARRAYDVNPVPROC glad_glDepthRangeArraydvNV;
09611 #define glDepthRangeArraydvNV glad_glDepthRangeArraydvNV
09612 typedef void (APIENTRY PFNGLDEPTHRANGEINDEXEDDVNPVPROC) (GLuint index, GLdouble n, GLdouble f);
09613 GLAPI PFNGLDEPTHRANGEINDEXEDDVNPVPROC glad_glDepthRangeIndexeddNV;
09614 #define glDepthRangeIndexeddNV glad_glDepthRangeIndexeddNV
09615 #endif
09616 #ifndef GL_ARB_window_pos
09617 #define GL_ARB_window_pos 1
09618 GLAPI int GLAD_GL_ARB_window_pos;
09619 typedef void (APIENTRY PFNGLWINDOWPOS2DARBPROC) (GLdouble x, GLdouble y);
09620 GLAPI PFNGLWINDOWPOS2DARBPROC glad_glWindowPos2dARB;
09621 #define glWindowPos2dARB glad_glWindowPos2dARB
09622 typedef void (APIENTRY PFNGLWINDOWPOS2DVARBPROC) (const GLdouble *v);
09623 GLAPI PFNGLWINDOWPOS2DVARBPROC glad_glWindowPos2dvARB;
09624 #define glWindowPos2dvARB glad_glWindowPos2dvARB
09625 typedef void (APIENTRY PFNGLWINDOWPOS2FARBPROC) (GLfloat x, GLfloat y);
09626 GLAPI PFNGLWINDOWPOS2FARBPROC glad_glWindowPos2fARB;
09627 #define glWindowPos2fARB glad_glWindowPos2fARB
09628 typedef void (APIENTRY PFNGLWINDOWPOS2FVARBPROC) (const GLfloat *v);
09629 GLAPI PFNGLWINDOWPOS2FVARBPROC glad_glWindowPos2fvARB;
09630 #define glWindowPos2fvARB glad_glWindowPos2fvARB
09631 typedef void (APIENTRY PFNGLWINDOWPOS2IARBPROC) (GLint x, GLint y);
09632 GLAPI PFNGLWINDOWPOS2IARBPROC glad_glWindowPos2iARB;
09633 #define glWindowPos2iARB glad_glWindowPos2iARB
09634 typedef void (APIENTRY PFNGLWINDOWPOS2IVARBPROC) (const GLint *v);
09635 GLAPI PFNGLWINDOWPOS2IVARBPROC glad_glWindowPos2ivARB;
09636 #define glWindowPos2ivARB glad_glWindowPos2ivARB
09637 typedef void (APIENTRY PFNGLWINDOWPOS2SARBPROC) (GLshort x, GLshort y);
09638 GLAPI PFNGLWINDOWPOS2SARBPROC glad_glWindowPos2sARB;
09639 #define glWindowPos2sARB glad_glWindowPos2sARB
09640 typedef void (APIENTRY PFNGLWINDOWPOS2SVARBPROC) (const GLshort *v);
09641 GLAPI PFNGLWINDOWPOS2SVARBPROC glad_glWindowPos2svARB;
09642 #define glWindowPos2svARB glad_glWindowPos2svARB
09643 typedef void (APIENTRY PFNGLWINDOWPOS3DARBPROC) (GLdouble x, GLdouble y, GLdouble z);
09644 GLAPI PFNGLWINDOWPOS3DARBPROC glad_glWindowPos3dARB;
09645 #define glWindowPos3dARB glad_glWindowPos3dARB
09646 typedef void (APIENTRY PFNGLWINDOWPOS3DVARBPROC) (const GLdouble *v);
09647 GLAPI PFNGLWINDOWPOS3DVARBPROC glad_glWindowPos3dvARB;
09648 #define glWindowPos3dvARB glad_glWindowPos3dvARB
09649 typedef void (APIENTRY PFNGLWINDOWPOS3FARBPROC) (GLfloat x, GLfloat y, GLfloat z);
09650 GLAPI PFNGLWINDOWPOS3FARBPROC glad_glWindowPos3fARB;
09651 #define glWindowPos3fARB glad_glWindowPos3fARB
09652 typedef void (APIENTRY PFNGLWINDOWPOS3FVARBPROC) (const GLfloat *v);

```

```

09653 GLAPI PFNGLWINDOWPOS3FVARBPROC glad_glWindowPos3fvARB;
09654 #define glWindowPos3fvARB glad_glWindowPos3fvARB
09655 typedef void (APIENTRY PFNGLWINDOWPOS3IARBPROC)(GLint x, GLint y, GLint z);
09656 GLAPI PFNGLWINDOWPOS3IARBPROC glad_glWindowPos3iARB;
09657 #define glWindowPos3iARB glad_glWindowPos3iARB
09658 typedef void (APIENTRY PFNGLWINDOWPOS3IVARBPROC)(const GLint *v);
09659 GLAPI PFNGLWINDOWPOS3IVARBPROC glad_glWindowPos3ivARB;
09660 #define glWindowPos3ivARB glad_glWindowPos3ivARB
09661 typedef void (APIENTRY PFNGLWINDOWPOS3SARBPROC)(GLshort x, GLshort y, GLshort z);
09662 GLAPI PFNGLWINDOWPOS3SARBPROC glad_glWindowPos3sARB;
09663 #define glWindowPos3sARB glad_glWindowPos3sARB
09664 typedef void (APIENTRY PFNGLWINDOWPOS3SVARBPROC)(const GLshort *v);
09665 GLAPI PFNGLWINDOWPOS3SVARBPROC glad_glWindowPos3svARB;
09666 #define glWindowPos3svARB glad_glWindowPos3svARB
09667 #endif
09668 #ifndef GL_ATI_draw_buffers
09669 #define GL_ATI_draw_buffers 1
09670 GLAPI int GLAD_GL_ATI_draw_buffers;
09671 typedef void (APIENTRY PFNGLDRAWBUFFERSATIPROC)(GLsizei n, const GLenum *bufs);
09672 GLAPI PFNGLDRAWBUFFERSATIPROC glad_glDrawBuffersATI;
09673 #define glDrawBuffersATI glad_glDrawBuffersATI
09674 #endif
09675 #ifndef GL_ATI_element_array
09676 #define GL_ATI_element_array 1
09677 GLAPI int GLAD_GL_ATI_element_array;
09678 typedef void (APIENTRY PFNGLELEMENTPOINTERATIPROC)(GLenum type, const void *pointer);
09679 GLAPI PFNGLELEMENTPOINTERATIPROC glad_glElementPointerATI;
09680 #define glElementPointerATI glad_glElementPointerATI
09681 typedef void (APIENTRY PFNGLDRAWELEMENTARRAYATIPROC)(GLenum mode, GLsizei count);
09682 GLAPI PFNGLDRAWELEMENTARRAYATIPROC glad_glDrawElementArrayATI;
09683 #define glDrawElementArrayATI glad_glDrawElementArrayATI
09684 typedef void (APIENTRY PFNGLDRAWRANGEELEMENTARRAYATIPROC)(GLenum mode, GLuint start, GLuint end,
GLsizei count);
09685 GLAPI PFNGLDRAWRANGEELEMENTARRAYATIPROC glad_glDrawRangeElementArrayATI;
09686 #define glDrawRangeElementArrayATI glad_glDrawRangeElementArrayATI
09687 #endif
09688 #ifndef GL_ATI_envmap_bumpmap
09689 #define GL_ATI_envmap_bumpmap 1
09690 GLAPI int GLAD_GL_ATI_envmap_bumpmap;
09691 typedef void (APIENTRY PFNGLTEXBUMPPARAMETERIVATIPROC)(GLenum pname, const GLint *param);
09692 GLAPI PFNGLTEXBUMPPARAMETERIVATIPROC glad_glTexBumpParameterivATI;
09693 #define glTexBumpParameterivATI glad_glTexBumpParameterivATI
09694 typedef void (APIENTRY PFNGLTEXBUMPPARAMETERFVATIPROC)(GLenum pname, const GLfloat *param);
09695 GLAPI PFNGLTEXBUMPPARAMETERFVATIPROC glad_glTexBumpParameterfvATI;
09696 #define glTexBumpParameterfvATI glad_glTexBumpParameterfvATI
09697 typedef void (APIENTRY PFNGLGETTEXBUMPPARAMETERIVATIPROC)(GLenum pname, GLint *param);
09698 GLAPI PFNGLGETTEXBUMPPARAMETERIVATIPROC glad_glGetTexBumpParameterivATI;
09699 #define glGetTexBumpParameterivATI glad_glGetTexBumpParameterivATI
09700 typedef void (APIENTRY PFNGLGETTEXBUMPPARAMETERFVATIPROC)(GLenum pname, GLfloat *param);
09701 GLAPI PFNGLGETTEXBUMPPARAMETERFVATIPROC glad_glGetTexBumpParameterfvATI;
09702 #define glGetTexBumpParameterfvATI glad_glGetTexBumpParameterfvATI
09703 #endif
09704 #ifndef GL_ATI_fragment_shader
09705 #define GL_ATI_fragment_shader 1
09706 GLAPI int GLAD_GL_ATI_fragment_shader;
09707 typedef GLuint (APIENTRY PFNGLGENFRAGMENTSHADERSATIPROC)(GLuint range);
09708 GLAPI PFNGLGENFRAGMENTSHADERSATIPROC glad_glGenFragmentShadersATI;
09709 #define glGenFragmentShadersATI glad_glGenFragmentShadersATI
09710 typedef void (APIENTRY PFNGLBINDFRAGMENTSHADERATIPROC)(GLuint id);
09711 GLAPI PFNGLBINDFRAGMENTSHADERATIPROC glad_glBindFragmentShaderATI;
09712 #define glBindFragmentShaderATI glad_glBindFragmentShaderATI
09713 typedef void (APIENTRY PFNGLDELETEFRAGMENTSHADERATIPROC)(GLuint id);
09714 GLAPI PFNGLDELETEFRAGMENTSHADERATIPROC glad_glDeleteFragmentShaderATI;
09715 #define glDeleteFragmentShaderATI glad_glDeleteFragmentShaderATI
09716 typedef void (APIENTRY PFNGLBEGINFRAGMENTSHADERATIPROC)(void);
09717 GLAPI PFNGLBEGINFRAGMENTSHADERATIPROC glad_glBeginFragmentShaderATI;
09718 #define glBeginFragmentShaderATI glad_glBeginFragmentShaderATI
09719 typedef void (APIENTRY PFNGLENDFRAGMENTSHADERATIPROC)(void);
09720 GLAPI PFNGLENDFRAGMENTSHADERATIPROC glad_glEndFragmentShaderATI;
09721 #define glEndFragmentShaderATI glad_glEndFragmentShaderATI
09722 typedef void (APIENTRY PFNGLPASSTEXCOORDATIPROC)(GLuint dst, GLuint coord, GLenum swizzle);
09723 GLAPI PFNGLPASSTEXCOORDATIPROC glad_glPassTexCoordATI;
09724 #define glPassTexCoordATI glad_glPassTexCoordATI
09725 typedef void (APIENTRY PFNGLSAMPLEMAPATIPROC)(GLuint dst, GLuint interp, GLenum swizzle);
09726 GLAPI PFNGLSAMPLEMAPATIPROC glad_glSampleMapATI;
09727 #define glSampleMapATI glad_glSampleMapATI
09728 typedef void (APIENTRY PFNGLCOLORFRAGMENTOP1ATIPROC)(GLenum op, GLuint dst, GLuint dstMask, GLuint
dstMod, GLuint arg1, GLuint arg1Rep, GLuint arg1Mod);
09729 GLAPI PFNGLCOLORFRAGMENTOP1ATIPROC glad_glColorFragmentOp1ATI;
09730 #define glColorFragmentOp1ATI glad_glColorFragmentOp1ATI
09731 typedef void (APIENTRY PFNGLCOLORFRAGMENTOP2ATIPROC)(GLenum op, GLuint dst, GLuint dstMask, GLuint
dstMod, GLuint arg1, GLuint arg1Rep, GLuint arg1Mod, GLuint arg2, GLuint arg2Rep, GLuint arg2Mod);
09732 GLAPI PFNGLCOLORFRAGMENTOP2ATIPROC glad_glColorFragmentOp2ATI;
09733 #define glColorFragmentOp2ATI glad_glColorFragmentOp2ATI
09734 typedef void (APIENTRY PFNGLCOLORFRAGMENTOP3ATIPROC)(GLenum op, GLuint dst, GLuint dstMask, GLuint
dstMod, GLuint arg1, GLuint arg1Rep, GLuint arg1Mod, GLuint arg2, GLuint arg2Rep, GLuint arg2Mod,
GLuint arg3, GLuint arg3Rep, GLuint arg3Mod);

```



```

09735 GLAPI PFNGLCOLORFRAGMENTOP3ATIPROC glad_glColorFragmentOp3ATI;
09736 #define glColorFragmentOp3ATI glad_glColorFragmentOp3ATI
09737 typedef void (APIENTRY PFNGLALPHAFRAGMENTOP1ATIPROC) (GLenum op, GLuint dst, GLuint dstMod, GLuint
    arg1, GLuint arg1Rep, GLuint arg1Mod);
09738 GLAPI PFNGLALPHAFRAGMENTOP1ATIPROC glad_glAlphaFragmentOp1ATI;
09739 #define glAlphaFragmentOp1ATI glad_glAlphaFragmentOp1ATI
09740 typedef void (APIENTRY PFNGLALPHAFRAGMENTOP2ATIPROC) (GLenum op, GLuint dst, GLuint dstMod, GLuint
    arg1, GLuint arg1Rep, GLuint arg1Mod, GLuint arg2, GLuint arg2Rep, GLuint arg2Mod);
09741 GLAPI PFNGLALPHAFRAGMENTOP2ATIPROC glad_glAlphaFragmentOp2ATI;
09742 #define glAlphaFragmentOp2ATI glad_glAlphaFragmentOp2ATI
09743 typedef void (APIENTRY PFNGLALPHAFRAGMENTOP3ATIPROC) (GLenum op, GLuint dst, GLuint dstMod, GLuint
    arg1, GLuint arg1Rep, GLuint arg1Mod, GLuint arg2, GLuint arg2Rep, GLuint arg2Mod, GLuint arg3, GLuint
    arg3Rep, GLuint arg3Mod);
09744 GLAPI PFNGLALPHAFRAGMENTOP3ATIPROC glad_glAlphaFragmentOp3ATI;
09745 #define glAlphaFragmentOp3ATI glad_glAlphaFragmentOp3ATI
09746 typedef void (APIENTRY PFNGLSETFRAGMENTSHADERCONSTANTATIPROC) (GLuint dst, const GLfloat *value);
09747 GLAPI PFNGLSETFRAGMENTSHADERCONSTANTATIPROC glad_glSetFragmentShaderConstantATI;
09748 #define glSetFragmentShaderConstantATI glad_glSetFragmentShaderConstantATI
09749 #endif
09750 #ifndef GL_ATI_map_object_buffer
09751 #define GL_ATI_map_object_buffer 1
09752 GLAPI int GLAD_GL_ATI_map_object_buffer;
09753 typedef void * (APIENTRY PFNGLMAPOBJECTBUFFERATIPROC) (GLuint buffer);
09754 GLAPI PFNGLMAPOBJECTBUFFERATIPROC glad_glMapObjectBufferATI;
09755 #define glMapObjectBufferATI glad_glMapObjectBufferATI
09756 typedef void (APIENTRY PFNGLUNMAPOBJECTBUFFERATIPROC) (GLuint buffer);
09757 GLAPI PFNGLUNMAPOBJECTBUFFERATIPROC glad_glUnmapObjectBufferATI;
09758 #define glUnmapObjectBufferATI glad_glUnmapObjectBufferATI
09759 #endif
09760 #ifndef GL_ATI_meminfo
09761 #define GL_ATI_meminfo 1
09762 GLAPI int GLAD_GL_ATI_meminfo;
09763 #endif
09764 #ifndef GL_ATI_pixel_format_float
09765 #define GL_ATI_pixel_format_float 1
09766 GLAPI int GLAD_GL_ATI_pixel_format_float;
09767 #endif
09768 #ifndef GL_ATI_pn_triangles
09769 #define GL_ATI_pn_triangles 1
09770 GLAPI int GLAD_GL_ATI_pn_triangles;
09771 typedef void (APIENTRY PFNGLPNTRIANGLESIATIPROC) (GLenum pname, GLint param);
09772 GLAPI PFNGLPNTRIANGLESIATIPROC glad_glPNTrianglesiATI;
09773 #define glPNTrianglesiATI glad_glPNTrianglesiATI
09774 typedef void (APIENTRY PFNGLPNTRIANGLESFATIPROC) (GLenum pname, GLfloat param);
09775 GLAPI PFNGLPNTRIANGLESFATIPROC glad_glPNTrianglesfATI;
09776 #define glPNTrianglesfATI glad_glPNTrianglesfATI
09777 #endif
09778 #ifndef GL_ATI_separate_stencil
09779 #define GL_ATI_separate_stencil 1
09780 GLAPI int GLAD_GL_ATI_separate_stencil;
09781 typedef void (APIENTRY PFNGLSTENCILOPSEPARATEATIPROC) (GLenum face, GLenum sfail, GLenum dpfail,
    GLenum dppass);
09782 GLAPI PFNGLSTENCILOPSEPARATEATIPROC glad_glStencilOpSeparateATI;
09783 #define glStencilOpSeparateATI glad_glStencilOpSeparateATI
09784 typedef void (APIENTRY PFNGLSTENCILFUNCSEPARATEATIPROC) (GLenum frontfunc, GLenum backfunc, GLint ref,
    GLuint mask);
09785 GLAPI PFNGLSTENCILFUNCSEPARATEATIPROC glad_glStencilFuncSeparateATI;
09786 #define glStencilFuncSeparateATI glad_glStencilFuncSeparateATI
09787 #endif
09788 #ifndef GL_ATI_text_fragment_shader
09789 #define GL_ATI_text_fragment_shader 1
09790 GLAPI int GLAD_GL_ATI_text_fragment_shader;
09791 #endif
09792 #ifndef GL_ATI_texture_env_combine3
09793 #define GL_ATI_texture_env_combine3 1
09794 GLAPI int GLAD_GL_ATI_texture_env_combine3;
09795 #endif
09796 #ifndef GL_ATI_texture_float
09797 #define GL_ATI_texture_float 1
09798 GLAPI int GLAD_GL_ATI_texture_float;
09799 #endif
09800 #ifndef GL_ATI_texture_mirror_once
09801 #define GL_ATI_texture_mirror_once 1
09802 GLAPI int GLAD_GL_ATI_texture_mirror_once;
09803 #endif
09804 #ifndef GL_ATI_vertex_array_object
09805 #define GL_ATI_vertex_array_object 1
09806 GLAPI int GLAD_GL_ATI_vertex_array_object;
09807 typedef GLuint (APIENTRY PFNGLNEWOBJECTBUFFERATIPROC) (GLsizei size, const void *pointer, GLenum
    usage);
09808 GLAPI PFNGLNEWOBJECTBUFFERATIPROC glad_glNewObjectBufferATI;
09809 #define glNewObjectBufferATI glad_glNewObjectBufferATI
09810 typedef GLboolean (APIENTRY PFNGLISOBJECTBUFFERATIPROC) (GLuint buffer);
09811 GLAPI PFNGLISOBJECTBUFFERATIPROC glad_glIsObjectBufferATI;
09812 #define glIsObjectBufferATI glad_glIsObjectBufferATI
09813 typedef void (APIENTRY PFNGLUPDATEOBJECTBUFFERATIPROC) (GLuint buffer, GLuint offset, GLsizei size,
    const void *pointer, GLenum preserve);

```

```

09814 GLAPI PFNGLUPDATEOBJECTBUFFERATIPROC glad_glUpdateObjectBufferATI;
09815 #define glUpdateObjectBufferATI glad_glUpdateObjectBufferATI
09816 typedef void (APIENTRY PFNGLGETOBJECTBUFFERFVATIPROC) (GLuint buffer, GLenum pname, GLfloat *params);
09817 GLAPI PFNGLGETOBJECTBUFFERFVATIPROC glad_glGetObjectBufferfvATI;
09818 #define glGetObjectBufferfvATI glad_glGetObjectBufferfvATI
09819 typedef void (APIENTRY PFNGLGETOBJECTBUFFERIVATIPROC) (GLuint buffer, GLenum pname, GLint *params);
09820 GLAPI PFNGLGETOBJECTBUFFERIVATIPROC glad_glGetObjectBufferivATI;
09821 #define glGetObjectBufferivATI glad_glGetObjectBufferivATI
09822 typedef void (APIENTRY PFNGLFREEOBJECTBUFFERATIPROC) (GLuint buffer);
09823 GLAPI PFNGLFREEOBJECTBUFFERATIPROC glad_glFreeObjectBufferATI;
09824 #define glFreeObjectBufferATI glad_glFreeObjectBufferATI
09825 typedef void (APIENTRY PFNGLARRAYOBJECTATIPROC) (GLenum array, GLint size, GLenum type, GLsizei
stride, GLuint buffer, GLuint offset);
09826 GLAPI PFNGLARRAYOBJECTATIPROC glad_glArrayObjectATI;
09827 #define glArrayObjectATI glad_glArrayObjectATI
09828 typedef void (APIENTRY PFNGLGETARRAYOBJECTFVATIPROC) (GLenum array, GLenum pname, GLfloat *params);
09829 GLAPI PFNGLGETARRAYOBJECTFVATIPROC glad_glGetArrayObjectfvATI;
09830 #define glGetArrayObjectfvATI glad_glGetArrayObjectfvATI
09831 typedef void (APIENTRY PFNGLGETARRAYOBJECTIVATIPROC) (GLenum array, GLenum pname, GLint *params);
09832 GLAPI PFNGLGETARRAYOBJECTIVATIPROC glad_glGetArrayObjectivATI;
09833 #define glGetArrayObjectivATI glad_glGetArrayObjectivATI
09834 typedef void (APIENTRY PFNGLVARIANTARRAYOBJECTATIPROC) (GLuint id, GLenum type, GLsizei stride, GLuint
buffer, GLuint offset);
09835 GLAPI PFNGLVARIANTARRAYOBJECTATIPROC glad_glVariantArrayObjectATI;
09836 #define glVariantArrayObjectATI glad_glVariantArrayObjectATI
09837 typedef void (APIENTRY PFNGLGETVARIANTARRAYOBJECTFVATIPROC) (GLuint id, GLenum pname, GLfloat
*params);
09838 GLAPI PFNGLGETVARIANTARRAYOBJECTFVATIPROC glad_glGetVariantArrayObjectfvATI;
09839 #define glGetVariantArrayObjectfvATI glad_glGetVariantArrayObjectfvATI
09840 typedef void (APIENTRY PFNGLGETVARIANTARRAYOBJECTIVATIPROC) (GLuint id, GLenum pname, GLint *params);
09841 GLAPI PFNGLGETVARIANTARRAYOBJECTIVATIPROC glad_glGetVariantArrayObjectivATI;
09842 #define glGetVariantArrayObjectivATI glad_glGetVariantArrayObjectivATI
09843 #endif
09844 #ifndef GL_ATI_vertex_attrib_array_object
09845 #define GL_ATI_vertex_attrib_array_object 1
09846 GLAPI int GLAD_GL_ATI_vertex_attrib_array_object;
09847 typedef void (APIENTRY PFNGLVERTEXATTRIBARRAYOBJECTATIPROC) (GLuint index, GLint size, GLenum type,
GLboolean normalized, GLsizei stride, GLuint buffer, GLuint offset);
09848 GLAPI PFNGLVERTEXATTRIBARRAYOBJECTATIPROC glad_glVertexAttribArrayObjectATI;
09849 #define glVertexAttribArrayObjectATI glad_glVertexAttribArrayObjectATI
09850 typedef void (APIENTRY PFNGLGETVERTEXATTRIBARRAYOBJECTFVATIPROC) (GLuint index, GLenum pname, GLfloat
*params);
09851 GLAPI PFNGLGETVERTEXATTRIBARRAYOBJECTFVATIPROC glad_glGetVertexAttribArrayObjectfvATI;
09852 #define glGetVertexAttribArrayObjectfvATI glad_glGetVertexAttribArrayObjectfvATI
09853 typedef void (APIENTRY PFNGLGETVERTEXATTRIBARRAYOBJECTIVATIPROC) (GLuint index, GLenum pname, GLint
*params);
09854 GLAPI PFNGLGETVERTEXATTRIBARRAYOBJECTIVATIPROC glad_glGetVertexAttribArrayObjectivATI;
09855 #define glGetVertexAttribArrayObjectivATI glad_glGetVertexAttribArrayObjectivATI
09856 #endif
09857 #ifndef GL_ATI_vertex_streams
09858 #define GL_ATI_vertex_streams 1
09859 GLAPI int GLAD_GL_ATI_vertex_streams;
09860 typedef void (APIENTRY PFNGLVERTEXSTREAM1SATIPROC) (GLenum stream, GLshort x);
09861 GLAPI PFNGLVERTEXSTREAM1SATIPROC glad_glVertexStream1sATI;
09862 #define glVertexStream1sATI glad_glVertexStream1sATI
09863 typedef void (APIENTRY PFNGLVERTEXSTREAM1SVATIPROC) (GLenum stream, const GLshort *coords);
09864 GLAPI PFNGLVERTEXSTREAM1SVATIPROC glad_glVertexStream1svATI;
09865 #define glVertexStream1svATI glad_glVertexStream1svATI
09866 typedef void (APIENTRY PFNGLVERTEXSTREAM1IATIPROC) (GLenum stream, GLint x);
09867 GLAPI PFNGLVERTEXSTREAM1IATIPROC glad_glVertexStream1iATI;
09868 #define glVertexStream1iATI glad_glVertexStream1iATI
09869 typedef void (APIENTRY PFNGLVERTEXSTREAM1IVATIPROC) (GLenum stream, const GLint *coords);
09870 GLAPI PFNGLVERTEXSTREAM1IVATIPROC glad_glVertexStream1ivATI;
09871 #define glVertexStream1ivATI glad_glVertexStream1ivATI
09872 typedef void (APIENTRY PFNGLVERTEXSTREAM1FATIPROC) (GLenum stream, GLfloat x);
09873 GLAPI PFNGLVERTEXSTREAM1FATIPROC glad_glVertexStream1fATI;
09874 #define glVertexStream1fATI glad_glVertexStream1fATI
09875 typedef void (APIENTRY PFNGLVERTEXSTREAM1FVATIPROC) (GLenum stream, const GLfloat *coords);
09876 GLAPI PFNGLVERTEXSTREAM1FVATIPROC glad_glVertexStream1fvATI;
09877 #define glVertexStream1fvATI glad_glVertexStream1fvATI
09878 typedef void (APIENTRY PFNGLVERTEXSTREAM1DATIPROC) (GLenum stream, GLdouble x);
09879 GLAPI PFNGLVERTEXSTREAM1DATIPROC glad_glVertexStream1dATI;
09880 #define glVertexStream1dATI glad_glVertexStream1dATI
09881 typedef void (APIENTRY PFNGLVERTEXSTREAM1DVATIPROC) (GLenum stream, const GLdouble *coords);
09882 GLAPI PFNGLVERTEXSTREAM1DVATIPROC glad_glVertexStream1dvATI;
09883 #define glVertexStream1dvATI glad_glVertexStream1dvATI
09884 typedef void (APIENTRY PFNGLVERTEXSTREAM2SATIPROC) (GLenum stream, GLshort x, GLshort y);
09885 GLAPI PFNGLVERTEXSTREAM2SATIPROC glad_glVertexStream2sATI;
09886 #define glVertexStream2sATI glad_glVertexStream2sATI
09887 typedef void (APIENTRY PFNGLVERTEXSTREAM2SVATIPROC) (GLenum stream, const GLshort *coords);
09888 GLAPI PFNGLVERTEXSTREAM2SVATIPROC glad_glVertexStream2svATI;
09889 #define glVertexStream2svATI glad_glVertexStream2svATI
09890 typedef void (APIENTRY PFNGLVERTEXSTREAM2IATIPROC) (GLenum stream, GLint x, GLint y);
09891 GLAPI PFNGLVERTEXSTREAM2IATIPROC glad_glVertexStream2iATI;
09892 #define glVertexStream2iATI glad_glVertexStream2iATI
09893 typedef void (APIENTRY PFNGLVERTEXSTREAM2IVATIPROC) (GLenum stream, const GLint *coords);
09894 GLAPI PFNGLVERTEXSTREAM2IVATIPROC glad_glVertexStream2ivATI;

```

```

09895 #define glVertexStream2ivATI glad_glVertexStream2ivATI
09896 typedef void (APIENTRY PFNGLVERTEXSTREAM2FATIPROC) (GLenum stream, GLfloat x, GLfloat y);
09897 GLAPI PFNGLVERTEXSTREAM2FATIPROC glad_glVertexStream2fATI;
09898 #define glVertexStream2fATI glad_glVertexStream2fATI
09899 typedef void (APIENTRY PFNGLVERTEXSTREAM2FVATIPROC) (GLenum stream, const GLfloat *coords);
09900 GLAPI PFNGLVERTEXSTREAM2FVATIPROC glad_glVertexStream2fvATI;
09901 #define glVertexStream2dvATI glad_glVertexStream2dvATI
09902 typedef void (APIENTRY PFNGLVERTEXSTREAM2DATIPROC) (GLenum stream, GLdouble x, GLdouble y);
09903 GLAPI PFNGLVERTEXSTREAM2DATIPROC glad_glVertexStream2dATI;
09904 #define glVertexStream2dATI glad_glVertexStream2dATI
09905 typedef void (APIENTRY PFNGLVERTEXSTREAM2DVATIPROC) (GLenum stream, const GLdouble *coords);
09906 GLAPI PFNGLVERTEXSTREAM2DVATIPROC glad_glVertexStream2dvATI;
09907 #define glVertexStream2dvATI glad_glVertexStream2dvATI
09908 typedef void (APIENTRY PFNGLVERTEXSTREAM3SATIPROC) (GLenum stream, GLshort x, GLshort y, GLshort z);
09909 GLAPI PFNGLVERTEXSTREAM3SATIPROC glad_glVertexStream3sATI;
09910 #define glVertexStream3sATI glad_glVertexStream3sATI
09911 typedef void (APIENTRY PFNGLVERTEXSTREAM3SVATIPROC) (GLenum stream, const GLshort *coords);
09912 GLAPI PFNGLVERTEXSTREAM3SVATIPROC glad_glVertexStream3svATI;
09913 #define glVertexStream3svATI glad_glVertexStream3svATI
09914 typedef void (APIENTRY PFNGLVERTEXSTREAM3IATIPROC) (GLenum stream, GLint x, GLint y, GLint z);
09915 GLAPI PFNGLVERTEXSTREAM3IATIPROC glad_glVertexStream3iATI;
09916 #define glVertexStream3iATI glad_glVertexStream3iATI
09917 typedef void (APIENTRY PFNGLVERTEXSTREAM3IVATIPROC) (GLenum stream, const GLint *coords);
09918 GLAPI PFNGLVERTEXSTREAM3IVATIPROC glad_glVertexStream3ivATI;
09919 #define glVertexStream3ivATI glad_glVertexStream3ivATI
09920 typedef void (APIENTRY PFNGLVERTEXSTREAM3FATIPROC) (GLenum stream, GLfloat x, GLfloat y, GLfloat z);
09921 GLAPI PFNGLVERTEXSTREAM3FATIPROC glad_glVertexStream3fATI;
09922 #define glVertexStream3fATI glad_glVertexStream3fATI
09923 typedef void (APIENTRY PFNGLVERTEXSTREAM3FVATIPROC) (GLenum stream, const GLfloat *coords);
09924 GLAPI PFNGLVERTEXSTREAM3FVATIPROC glad_glVertexStream3fvATI;
09925 #define glVertexStream3fvATI glad_glVertexStream3fvATI
09926 typedef void (APIENTRY PFNGLVERTEXSTREAM3DATIPROC) (GLenum stream, GLdouble x, GLdouble y, GLdouble
z);
09927 GLAPI PFNGLVERTEXSTREAM3DATIPROC glad_glVertexStream3dATI;
09928 #define glVertexStream3dATI glad_glVertexStream3dATI
09929 typedef void (APIENTRY PFNGLVERTEXSTREAM3DVATIPROC) (GLenum stream, const GLdouble *coords);
09930 GLAPI PFNGLVERTEXSTREAM3DVATIPROC glad_glVertexStream3dvATI;
09931 #define glVertexStream3dvATI glad_glVertexStream3dvATI
09932 typedef void (APIENTRY PFNGLVERTEXSTREAM4SATIPROC) (GLenum stream, GLshort x, GLshort y, GLshort z,
GLshort w);
09933 GLAPI PFNGLVERTEXSTREAM4SATIPROC glad_glVertexStream4sATI;
09934 #define glVertexStream4sATI glad_glVertexStream4sATI
09935 typedef void (APIENTRY PFNGLVERTEXSTREAM4SVATIPROC) (GLenum stream, const GLshort *coords);
09936 GLAPI PFNGLVERTEXSTREAM4SVATIPROC glad_glVertexStream4svATI;
09937 #define glVertexStream4svATI glad_glVertexStream4svATI
09938 typedef void (APIENTRY PFNGLVERTEXSTREAM4IATIPROC) (GLenum stream, GLint x, GLint y, GLint z, GLint
w);
09939 GLAPI PFNGLVERTEXSTREAM4IATIPROC glad_glVertexStream4iATI;
09940 #define glVertexStream4iATI glad_glVertexStream4iATI
09941 typedef void (APIENTRY PFNGLVERTEXSTREAM4IVATIPROC) (GLenum stream, const GLint *coords);
09942 GLAPI PFNGLVERTEXSTREAM4IVATIPROC glad_glVertexStream4ivATI;
09943 #define glVertexStream4ivATI glad_glVertexStream4ivATI
09944 typedef void (APIENTRY PFNGLVERTEXSTREAM4FATIPROC) (GLenum stream, GLfloat x, GLfloat y, GLfloat z,
GLfloat w);
09945 GLAPI PFNGLVERTEXSTREAM4FATIPROC glad_glVertexStream4fATI;
09946 #define glVertexStream4fATI glad_glVertexStream4fATI
09947 typedef void (APIENTRY PFNGLVERTEXSTREAM4FVATIPROC) (GLenum stream, const GLfloat *coords);
09948 GLAPI PFNGLVERTEXSTREAM4FVATIPROC glad_glVertexStream4fvATI;
09949 #define glVertexStream4fvATI glad_glVertexStream4fvATI
09950 typedef void (APIENTRY PFNGLVERTEXSTREAM4DATIPROC) (GLenum stream, GLdouble x, GLdouble y, GLdouble z,
GLdouble w);
09951 GLAPI PFNGLVERTEXSTREAM4DATIPROC glad_glVertexStream4dATI;
09952 #define glVertexStream4dATI glad_glVertexStream4dATI
09953 typedef void (APIENTRY PFNGLVERTEXSTREAM4DVATIPROC) (GLenum stream, const GLdouble *coords);
09954 GLAPI PFNGLVERTEXSTREAM4DVATIPROC glad_glVertexStream4dvATI;
09955 #define glVertexStream4dvATI glad_glVertexStream4dvATI
09956 typedef void (APIENTRY PFNGLNORMALSTREAM3BATIPROC) (GLenum stream, GLbyte nx, GLbyte ny, GLbyte nz);
09957 GLAPI PFNGLNORMALSTREAM3BATIPROC glad_glNormalStream3bATI;
09958 #define glNormalStream3bATI glad_glNormalStream3bATI
09959 typedef void (APIENTRY PFNGLNORMALSTREAM3BVATIPROC) (GLenum stream, const GLbyte *coords);
09960 GLAPI PFNGLNORMALSTREAM3BVATIPROC glad_glNormalStream3bvATI;
09961 #define glNormalStream3bvATI glad_glNormalStream3bvATI
09962 typedef void (APIENTRY PFNGLNORMALSTREAM3SATIPROC) (GLenum stream, GLshort nx, GLshort ny, GLshort
nz);
09963 GLAPI PFNGLNORMALSTREAM3SATIPROC glad_glNormalStream3sATI;
09964 #define glNormalStream3sATI glad_glNormalStream3sATI
09965 typedef void (APIENTRY PFNGLNORMALSTREAM3SVATIPROC) (GLenum stream, const GLshort *coords);
09966 GLAPI PFNGLNORMALSTREAM3SVATIPROC glad_glNormalStream3svATI;
09967 #define glNormalStream3svATI glad_glNormalStream3svATI
09968 typedef void (APIENTRY PFNGLNORMALSTREAM3IATIPROC) (GLenum stream, GLint nx, GLint ny, GLint nz);
09969 GLAPI PFNGLNORMALSTREAM3IATIPROC glad_glNormalStream3iATI;
09970 #define glNormalStream3iATI glad_glNormalStream3iATI
09971 typedef void (APIENTRY PFNGLNORMALSTREAM3IVATIPROC) (GLenum stream, const GLint *coords);
09972 GLAPI PFNGLNORMALSTREAM3IVATIPROC glad_glNormalStream3ivATI;
09973 #define glNormalStream3ivATI glad_glNormalStream3ivATI
09974 typedef void (APIENTRY PFNGLNORMALSTREAM3FATIPROC) (GLenum stream, GLfloat nx, GLfloat ny, GLfloat
nz);

```



```

09975 GLAPI PFNGLNORMALSTREAM3FATIPROC glad_glNormalStream3fATI;
09976 #define glNormalStream3fATI glad_glNormalStream3fATI
09977 typedef void (APIENTRY PFNGLNORMALSTREAM3FVATIPROC) (GLenum stream, const GLfloat *coords);
09978 GLAPI PFNGLNORMALSTREAM3FVATIPROC glad_glNormalStream3fvATI;
09979 #define glNormalStream3fvATI glad_glNormalStream3fvATI
09980 typedef void (APIENTRY PFNGLNORMALSTREAM3DATIPROC) (GLenum stream, GLdouble nx, GLdouble ny, GLdouble
nz);
09981 GLAPI PFNGLNORMALSTREAM3DATIPROC glad_glNormalStream3dATI;
09982 #define glNormalStream3dATI glad_glNormalStream3dATI
09983 typedef void (APIENTRY PFNGLNORMALSTREAM3DVATIPROC) (GLenum stream, const GLdouble *coords);
09984 GLAPI PFNGLNORMALSTREAM3DVATIPROC glad_glNormalStream3dvATI;
09985 #define glNormalStream3dvATI glad_glNormalStream3dvATI
09986 typedef void (APIENTRY PFNGLCLIENTACTIVEVERTEXSTREAMATIPROC) (GLenum stream);
09987 GLAPI PFNGLCLIENTACTIVEVERTEXSTREAMATIPROC glad_glClientActiveVertexStreamATI;
09988 #define glClientActiveVertexStreamATI glad_glClientActiveVertexStreamATI
09989 typedef void (APIENTRY PFNGLVERTEXBLENDENVIATIPROC) (GLenum pname, GLint param);
09990 GLAPI PFNGLVERTEXBLENDENVIATIPROC glad_glVertexBlendEnviATI;
09991 #define glVertexBlendEnviATI glad_glVertexBlendEnviATI
09992 typedef void (APIENTRY PFNGLVERTEXBLENDENVFATIPROC) (GLenum pname, GLfloat param);
09993 GLAPI PFNGLVERTEXBLENDENVFATIPROC glad_glVertexBlendEnvfATI;
09994 #define glVertexBlendEnvfATI glad_glVertexBlendEnvfATI
09995 #endif
09996 #ifndef GL_EXT_422_pixels
09997 #define GL_EXT_422_pixels 1
09998 GLAPI int GLAD_GL_EXT_422_pixels;
09999 #endif
10000 #ifndef GL_EXT_EGL_image_storage
10001 #define GL_EXT_EGL_image_storage 1
10002 GLAPI int GLAD_GL_EXT_EGL_image_storage;
10003 typedef void (APIENTRY PFNGLEGLIMAGETARGETTEXTSTORAGEEXTPROC) (GLenum target, GLeglImageOES image,
const GLint* attrib_list);
10004 GLAPI PFNGLEGLIMAGETARGETTEXTSTORAGEEXTPROC glad_glEGLImageTargetTexStorageEXT;
10005 #define glEGLImageTargetTexStorageEXT glad_glEGLImageTargetTexStorageEXT
10006 typedef void (APIENTRY PFNGLEGLIMAGETARGETTEXTURESTORAGEEXTPROC) (GLuint texture, GLeglImageOES image,
const GLint* attrib_list);
10007 GLAPI PFNGLEGLIMAGETARGETTEXTURESTORAGEEXTPROC glad_glEGLImageTargetTextureStorageEXT;
10008 #define glEGLImageTargetTextureStorageEXT glad_glEGLImageTargetTextureStorageEXT
10009 #endif
10010 #ifndef GL_EXT_EGL_sync
10011 #define GL_EXT_EGL_sync 1
10012 GLAPI int GLAD_GL_EXT_EGL_sync;
10013 #endif
10014 #ifndef GL_EXT_abgr
10015 #define GL_EXT_abgr 1
10016 GLAPI int GLAD_GL_EXT_abgr;
10017 #endif
10018 #ifndef GL_EXT_bgra
10019 #define GL_EXT_bgra 1
10020 GLAPI int GLAD_GL_EXT_bgra;
10021 #endif
10022 #ifndef GL_EXT_bindable_uniform
10023 #define GL_EXT_bindable_uniform 1
10024 GLAPI int GLAD_GL_EXT_bindable_uniform;
10025 typedef void (APIENTRY PFNGLUNIFORMBUFFEREXTPROC) (GLuint program, GLint location, GLuint buffer);
10026 GLAPI PFNGLUNIFORMBUFFEREXTPROC glad_glUniformBufferEXT;
10027 #define glUniformBufferEXT glad_glUniformBufferEXT
10028 typedef GLint (APIENTRY PFNGLGETUNIFORMBUFFERSIZEEXTPROC) (GLuint program, GLint location);
10029 GLAPI PFNGLGETUNIFORMBUFFERSIZEEXTPROC glad_glGetUniformBufferSizeEXT;
10030 #define glGetUniformBufferSizeEXT glad_glGetUniformBufferSizeEXT
10031 typedef GLintptr (APIENTRY PFNGLGETUNIFORMOFFSETEXTPROC) (GLuint program, GLint location);
10032 GLAPI PFNGLGETUNIFORMOFFSETEXTPROC glad_glGetUniformOffsetEXT;
10033 #define glGetUniformOffsetEXT glad_glGetUniformOffsetEXT
10034 #endif
10035 #ifndef GL_EXT_blend_color
10036 #define GL_EXT_blend_color 1
10037 GLAPI int GLAD_GL_EXT_blend_color;
10038 typedef void (APIENTRY PFNGLBLENDCOLOREXTPROC) (GLfloat red, GLfloat green, GLfloat blue, GLfloat
alpha);
10039 GLAPI PFNGLBLENDCOLOREXTPROC glad_glBlendColorEXT;
10040 #define glBlendColorEXT glad_glBlendColorEXT
10041 #endif
10042 #ifndef GL_EXT_blend_equation_separate
10043 #define GL_EXT_blend_equation_separate 1
10044 GLAPI int GLAD_GL_EXT_blend_equation_separate;
10045 typedef void (APIENTRY PFNGLBLENDSEPARATEEXTPROC) (GLenum modeRGB, GLenum modeAlpha);
10046 GLAPI PFNGLBLENDSEPARATEEXTPROC glad_glBlendEquationSeparateEXT;
10047 #define glBlendEquationSeparateEXT glad_glBlendEquationSeparateEXT
10048 #endif
10049 #ifndef GL_EXT_blend_func_separate
10050 #define GL_EXT_blend_func_separate 1
10051 GLAPI int GLAD_GL_EXT_blend_func_separate;
10052 typedef void (APIENTRY PFNGLBLENDFUNCSEPARATEEXTPROC) (GLenum sfactorRGB, GLenum dfactorRGB, GLenum
sfactorAlpha, GLenum dfactorAlpha);
10053 GLAPI PFNGLBLENDFUNCSEPARATEEXTPROC glad_glBlendFuncSeparateEXT;
10054 #define glBlendFuncSeparateEXT glad_glBlendFuncSeparateEXT
10055 #endif
10056 #ifndef GL_EXT_blend_logic_op

```

```

10057 #define GL_EXT_blend_logic_op 1
10058 GLAPI int GLAD_GL_EXT_blend_logic_op;
10059 #endif
10060 #ifndef GL_EXT_blend_minmax
10061 #define GL_EXT_blend_minmax 1
10062 GLAPI int GLAD_GL_EXT_blend_minmax;
10063 typedef void (APIENTRY PFNGLBLENDEQUATIONEXTPROC) (GLenum mode);
10064 GLAPI PFNGLBLENDEQUATIONEXTPROC glad_glBlendEquationEXT;
10065 #define glBlendEquationEXT glad_glBlendEquationEXT
10066 #endif
10067 #ifndef GL_EXT_blend_subtract
10068 #define GL_EXT_blend_subtract 1
10069 GLAPI int GLAD_GL_EXT_blend_subtract;
10070 #endif
10071 #ifndef GL_EXT_clip_volume_hint
10072 #define GL_EXT_clip_volume_hint 1
10073 GLAPI int GLAD_GL_EXT_clip_volume_hint;
10074 #endif
10075 #ifndef GL_EXT_cmyka
10076 #define GL_EXT_cmyka 1
10077 GLAPI int GLAD_GL_EXT_cmyka;
10078 #endif
10079 #ifndef GL_EXT_color_subtable
10080 #define GL_EXT_color_subtable 1
10081 GLAPI int GLAD_GL_EXT_color_subtable;
10082 typedef void (APIENTRY PFNGLCOLORSUBTABLEEXTPROC) (GLenum target, GLsizei start, GLsizei count, GLenum
format, GLenum type, const void *data);
10083 GLAPI PFNGLCOLORSUBTABLEEXTPROC glad_glColorSubTableEXT;
10084 #define glColorSubTableEXT glad_glColorSubTableEXT
10085 typedef void (APIENTRY PFNGLCOPYCOLORSUBTABLEEXTPROC) (GLenum target, GLsizei start, GLint x, GLint y,
GLsizei width);
10086 GLAPI PFNGLCOPYCOLORSUBTABLEEXTPROC glad_glCopyColorSubTableEXT;
10087 #define glCopyColorSubTableEXT glad_glCopyColorSubTableEXT
10088 #endif
10089 #ifndef GL_EXT_compiled_vertex_array
10090 #define GL_EXT_compiled_vertex_array 1
10091 GLAPI int GLAD_GL_EXT_compiled_vertex_array;
10092 typedef void (APIENTRY PFNGLLOCKARRAYSEXTPROC) (GLint first, GLsizei count);
10093 GLAPI PFNGLLOCKARRAYSEXTPROC glad_glLockArraysEXT;
10094 #define glLockArraysEXT glad_glLockArraysEXT
10095 typedef void (APIENTRY PFNGLUNLOCKARRAYSEXTPROC) (void);
10096 GLAPI PFNGLUNLOCKARRAYSEXTPROC glad_glUnlockArraysEXT;
10097 #define glUnlockArraysEXT glad_glUnlockArraysEXT
10098 #endif
10099 #ifndef GL_EXT_convolution
10100 #define GL_EXT_convolution 1
10101 GLAPI int GLAD_GL_EXT_convolution;
10102 typedef void (APIENTRY PFNGLCONVOLUTIONFILTER1DEXTPROC) (GLenum target, GLenum internalformat, GLsizei
width, GLenum format, GLenum type, const void *image);
10103 GLAPI PFNGLCONVOLUTIONFILTER1DEXTPROC glad_glConvolutionFilter1DEXT;
10104 #define glConvolutionFilter1DEXT glad_glConvolutionFilter1DEXT
10105 typedef void (APIENTRY PFNGLCONVOLUTIONFILTER2DEXTPROC) (GLenum target, GLenum internalformat, GLsizei
width, GLsizei height, GLenum format, GLenum type, const void *image);
10106 GLAPI PFNGLCONVOLUTIONFILTER2DEXTPROC glad_glConvolutionFilter2DEXT;
10107 #define glConvolutionFilter2DEXT glad_glConvolutionFilter2DEXT
10108 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERFEXTPROC) (GLenum target, GLenum pname, GLfloat
params);
10109 GLAPI PFNGLCONVOLUTIONPARAMETERFEXTPROC glad_glConvolutionParameterfEXT;
10110 #define glConvolutionParameterfEXT glad_glConvolutionParameterfEXT
10111 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERFVEXTPROC) (GLenum target, GLenum pname, const GLfloat
*params);
10112 GLAPI PFNGLCONVOLUTIONPARAMETERFVEXTPROC glad_glConvolutionParameterfvEXT;
10113 #define glConvolutionParameterfvEXT glad_glConvolutionParameterfvEXT
10114 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERIEXTPROC) (GLenum target, GLenum pname, GLint params);
10115 GLAPI PFNGLCONVOLUTIONPARAMETERIEXTPROC glad_glConvolutionParameteriEXT;
10116 #define glConvolutionParameteriEXT glad_glConvolutionParameteriEXT
10117 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERIVEXTPROC) (GLenum target, GLenum pname, const GLint
*params);
10118 GLAPI PFNGLCONVOLUTIONPARAMETERIVEXTPROC glad_glConvolutionParameterivEXT;
10119 #define glConvolutionParameterivEXT glad_glConvolutionParameterivEXT
10120 typedef void (APIENTRY PFNGLCOPYCONVOLUTIONFILTER1DEXTPROC) (GLenum target, GLenum internalformat,
GLint x, GLint y, GLsizei width);
10121 GLAPI PFNGLCOPYCONVOLUTIONFILTER1DEXTPROC glad_glCopyConvolutionFilter1DEXT;
10122 #define glCopyConvolutionFilter1DEXT glad_glCopyConvolutionFilter1DEXT
10123 typedef void (APIENTRY PFNGLCOPYCONVOLUTIONFILTER2DEXTPROC) (GLenum target, GLenum internalformat,
GLint x, GLint y, GLsizei width, GLsizei height);
10124 GLAPI PFNGLCOPYCONVOLUTIONFILTER2DEXTPROC glad_glCopyConvolutionFilter2DEXT;
10125 #define glCopyConvolutionFilter2DEXT glad_glCopyConvolutionFilter2DEXT
10126 typedef void (APIENTRY PFNGLGETCONVOLUTIONFILTEREXTPROC) (GLenum target, GLenum format, GLenum type,
void *image);
10127 GLAPI PFNGLGETCONVOLUTIONFILTEREXTPROC glad_glGetConvolutionFilterEXT;
10128 #define glGetConvolutionFilterEXT glad_glGetConvolutionFilterEXT
10129 typedef void (APIENTRY PFNGLGETCONVOLUTIONPARAMETERFVEXTPROC) (GLenum target, GLenum pname, GLfloat
*params);
10130 GLAPI PFNGLGETCONVOLUTIONPARAMETERFVEXTPROC glad_glGetConvolutionParameterfvEXT;
10131 #define glGetConvolutionParameterfvEXT glad_glGetConvolutionParameterfvEXT
10132 typedef void (APIENTRY PFNGLGETCONVOLUTIONPARAMETERIVEXTPROC) (GLenum target, GLenum pname, GLint

```

```

        *params);
10133 GLAPI PFNGLGETCONVOLUTIONPARAMETERIVEXTPROC glad_glGetConvolutionParameterivEXT;
10134 #define glGetConvolutionParameterivEXT glad_glGetConvolutionParameterivEXT
10135 typedef void (APIENTRYP PFNGLGETSEPARABLEFILTEREXTPROC) (GLenum target, GLenum format, GLenum type,
        void *row, void *column, void *span);
10136 GLAPI PFNGLGETSEPARABLEFILTEREXTPROC glad_glGetSeparableFilterEXT;
10137 #define glGetSeparableFilterEXT glad_glGetSeparableFilterEXT
10138 typedef void (APIENTRYP PFNGLSEPARABLEFILTER2DEXTPROC) (GLenum target, GLenum internalformat, GLsizei
        width, GLsizei height, GLenum format, GLenum type, const void *row, const void *column);
10139 GLAPI PFNGLSEPARABLEFILTER2DEXTPROC glad_glSeparableFilter2DEXT;
10140 #define glSeparableFilter2DEXT glad_glSeparableFilter2DEXT
10141 #endif
10142 #ifndef GL_EXT_coordinate_frame
10143 #define GL_EXT_coordinate_frame 1
10144 GLAPI int GLAD_GL_EXT_coordinate_frame;
10145 typedef void (APIENTRYP PFNGLTANGENT3BEXTPROC) (GLbyte tx, GLbyte ty, GLbyte tz);
10146 GLAPI PFNGLTANGENT3BEXTPROC glad_glTangent3bEXT;
10147 #define glTangent3bEXT glad_glTangent3bEXT
10148 typedef void (APIENTRYP PFNGLTANGENT3BVEXTPROC) (const GLbyte *v);
10149 GLAPI PFNGLTANGENT3BVEXTPROC glad_glTangent3bvEXT;
10150 #define glTangent3bvEXT glad_glTangent3bvEXT
10151 typedef void (APIENTRYP PFNGLTANGENT3DEXTPROC) (GLdouble tx, GLdouble ty, GLdouble tz);
10152 GLAPI PFNGLTANGENT3DEXTPROC glad_glTangent3dEXT;
10153 #define glTangent3dEXT glad_glTangent3dEXT
10154 typedef void (APIENTRYP PFNGLTANGENT3DVEXTPROC) (const GLdouble *v);
10155 GLAPI PFNGLTANGENT3DVEXTPROC glad_glTangent3dvEXT;
10156 #define glTangent3dvEXT glad_glTangent3dvEXT
10157 typedef void (APIENTRYP PFNGLTANGENT3FEXTPROC) (GLfloat tx, GLfloat ty, GLfloat tz);
10158 GLAPI PFNGLTANGENT3FEXTPROC glad_glTangent3fEXT;
10159 #define glTangent3fEXT glad_glTangent3fEXT
10160 typedef void (APIENTRYP PFNGLTANGENT3FVEXTPROC) (const GLfloat *v);
10161 GLAPI PFNGLTANGENT3FVEXTPROC glad_glTangent3fvEXT;
10162 #define glTangent3fvEXT glad_glTangent3fvEXT
10163 typedef void (APIENTRYP PFNGLTANGENT3IEXTPROC) (GLint tx, GLint ty, GLint tz);
10164 GLAPI PFNGLTANGENT3IEXTPROC glad_glTangent3iEXT;
10165 #define glTangent3iEXT glad_glTangent3iEXT
10166 typedef void (APIENTRYP PFNGLTANGENT3IVEXTPROC) (const GLint *v);
10167 GLAPI PFNGLTANGENT3IVEXTPROC glad_glTangent3ivEXT;
10168 #define glTangent3ivEXT glad_glTangent3ivEXT
10169 typedef void (APIENTRYP PFNGLTANGENT3SEXTPROC) (GLshort tx, GLshort ty, GLshort tz);
10170 GLAPI PFNGLTANGENT3SEXTPROC glad_glTangent3sEXT;
10171 #define glTangent3sEXT glad_glTangent3sEXT
10172 typedef void (APIENTRYP PFNGLTANGENT3SVEXTPROC) (const GLshort *v);
10173 GLAPI PFNGLTANGENT3SVEXTPROC glad_glTangent3svEXT;
10174 #define glTangent3svEXT glad_glTangent3svEXT
10175 typedef void (APIENTRYP PFNGLBINORMAL3BEXTPROC) (GLbyte bx, GLbyte by, GLbyte bz);
10176 GLAPI PFNGLBINORMAL3BEXTPROC glad_glBinormal3bEXT;
10177 #define glBinormal3bEXT glad_glBinormal3bEXT
10178 typedef void (APIENTRYP PFNGLBINORMAL3BVEXTPROC) (const GLbyte *v);
10179 GLAPI PFNGLBINORMAL3BVEXTPROC glad_glBinormal3bvEXT;
10180 #define glBinormal3bvEXT glad_glBinormal3bvEXT
10181 typedef void (APIENTRYP PFNGLBINORMAL3DEXTPROC) (GLdouble bx, GLdouble by, GLdouble bz);
10182 GLAPI PFNGLBINORMAL3DEXTPROC glad_glBinormal3dEXT;
10183 #define glBinormal3dEXT glad_glBinormal3dEXT
10184 typedef void (APIENTRYP PFNGLBINORMAL3DVEXTPROC) (const GLdouble *v);
10185 GLAPI PFNGLBINORMAL3DVEXTPROC glad_glBinormal3dvEXT;
10186 #define glBinormal3dvEXT glad_glBinormal3dvEXT
10187 typedef void (APIENTRYP PFNGLBINORMAL3FEXTPROC) (GLfloat bx, GLfloat by, GLfloat bz);
10188 GLAPI PFNGLBINORMAL3FEXTPROC glad_glBinormal3fEXT;
10189 #define glBinormal3fEXT glad_glBinormal3fEXT
10190 typedef void (APIENTRYP PFNGLBINORMAL3FVEXTPROC) (const GLfloat *v);
10191 GLAPI PFNGLBINORMAL3FVEXTPROC glad_glBinormal3fvEXT;
10192 #define glBinormal3fvEXT glad_glBinormal3fvEXT
10193 typedef void (APIENTRYP PFNGLBINORMAL3IEXTPROC) (GLint bx, GLint by, GLint bz);
10194 GLAPI PFNGLBINORMAL3IEXTPROC glad_glBinormal3iEXT;
10195 #define glBinormal3iEXT glad_glBinormal3iEXT
10196 typedef void (APIENTRYP PFNGLBINORMAL3IVEXTPROC) (const GLint *v);
10197 GLAPI PFNGLBINORMAL3IVEXTPROC glad_glBinormal3ivEXT;
10198 #define glBinormal3ivEXT glad_glBinormal3ivEXT
10199 typedef void (APIENTRYP PFNGLBINORMAL3SEXTPROC) (GLshort bx, GLshort by, GLshort bz);
10200 GLAPI PFNGLBINORMAL3SEXTPROC glad_glBinormal3sEXT;
10201 #define glBinormal3sEXT glad_glBinormal3sEXT
10202 typedef void (APIENTRYP PFNGLBINORMAL3SVEXTPROC) (const GLshort *v);
10203 GLAPI PFNGLBINORMAL3SVEXTPROC glad_glBinormal3svEXT;
10204 #define glBinormal3svEXT glad_glBinormal3svEXT
10205 typedef void (APIENTRYP PFNGLTANGENTPOINTEREXTPROC) (GLenum type, GLsizei stride, const void *pointer);
10206 GLAPI PFNGLTANGENTPOINTEREXTPROC glad_glTangentPointerEXT;
10207 #define glTangentPointerEXT glad_glTangentPointerEXT
10208 typedef void (APIENTRYP PFNGLBINORMALPOINTEREXTPROC) (GLenum type, GLsizei stride, const void
        *pointer);
10209 GLAPI PFNGLBINORMALPOINTEREXTPROC glad_glBinormalPointerEXT;
10210 #define glBinormalPointerEXT glad_glBinormalPointerEXT
10211 #endif
10212 #ifndef GL_EXT_copy_texture
10213 #define GL_EXT_copy_texture 1
10214 GLAPI int GLAD_GL_EXT_copy_texture;
10215 typedef void (APIENTRYP PFNGLCOPYTEXIMAGE1DEXTPROC) (GLenum target, GLint level, GLenum internalformat,

```

```

    GLint x, GLint y, GLsizei width, GLint border);
10216 GLAPI PFNGLCOPYTEXIMAGE1DEXTPROC glad_glCopyTexImage1DEXT;
10217 #define glCopyTexImage1DEXT glad_glCopyTexImage1DEXT
10218 typedef void (APIENTRY PFNGLCOPYTEXIMAGE2DEXTPROC) (GLenum target, GLint level, GLenum internalformat,
    GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
10219 GLAPI PFNGLCOPYTEXIMAGE2DEXTPROC glad_glCopyTexImage2DEXT;
10220 #define glCopyTexImage2DEXT glad_glCopyTexImage2DEXT
10221 typedef void (APIENTRY PFNGLCOPYTEXSUBIMAGE1DEXTPROC) (GLenum target, GLint level, GLint xoffset,
    GLint x, GLint y, GLsizei width);
10222 GLAPI PFNGLCOPYTEXSUBIMAGE1DEXTPROC glad_glCopyTexSubImage1DEXT;
10223 #define glCopyTexSubImage1DEXT glad_glCopyTexSubImage1DEXT
10224 typedef void (APIENTRY PFNGLCOPYTEXSUBIMAGE2DEXTPROC) (GLenum target, GLint level, GLint xoffset,
    GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
10225 GLAPI PFNGLCOPYTEXSUBIMAGE2DEXTPROC glad_glCopyTexSubImage2DEXT;
10226 #define glCopyTexSubImage2DEXT glad_glCopyTexSubImage2DEXT
10227 typedef void (APIENTRY PFNGLCOPYTEXSUBIMAGE3DEXTPROC) (GLenum target, GLint level, GLint xoffset,
    GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
10228 GLAPI PFNGLCOPYTEXSUBIMAGE3DEXTPROC glad_glCopyTexSubImage3DEXT;
10229 #define glCopyTexSubImage3DEXT glad_glCopyTexSubImage3DEXT
10230 #endif
10231 #ifndef GL_EXT_cull_vertex
10232 #define GL_EXT_cull_vertex 1
10233 GLAPI int GLAD_GL_EXT_cull_vertex;
10234 typedef void (APIENTRY PFNGLCULLPARAMETERDVEXTPROC) (GLenum pname, GLdouble *params);
10235 GLAPI PFNGLCULLPARAMETERDVEXTPROC glad_glCullParameterdvEXT;
10236 #define glCullParameterdvEXT glad_glCullParameterdvEXT
10237 typedef void (APIENTRY PFNGLCULLPARAMETERFVEXTPROC) (GLenum pname, GLfloat *params);
10238 GLAPI PFNGLCULLPARAMETERFVEXTPROC glad_glCullParameterfvEXT;
10239 #define glCullParameterfvEXT glad_glCullParameterfvEXT
10240 #endif
10241 #ifndef GL_EXT_debug_label
10242 #define GL_EXT_debug_label 1
10243 GLAPI int GLAD_GL_EXT_debug_label;
10244 typedef void (APIENTRY PFNGLLABELOBJECTEXTPROC) (GLenum type, GLuint object, GLsizei length, const
    GLchar *label);
10245 GLAPI PFNGLLABELOBJECTEXTPROC glad_glLabelObjectEXT;
10246 #define glLabelObjectEXT glad_glLabelObjectEXT
10247 typedef void (APIENTRY PFNGLGETOBJECTLABELEXTPROC) (GLenum type, GLuint object, GLsizei bufSize,
    GLsizei *length, GLchar *label);
10248 GLAPI PFNGLGETOBJECTLABELEXTPROC glad_glGetObjectLabelEXT;
10249 #define glGetObjectLabelEXT glad_glGetObjectLabelEXT
10250 #endif
10251 #ifndef GL_EXT_debug_marker
10252 #define GL_EXT_debug_marker 1
10253 GLAPI int GLAD_GL_EXT_debug_marker;
10254 typedef void (APIENTRY PFNGLINSERTEVENTMARKEREXTPROC) (GLsizei length, const GLchar *marker);
10255 GLAPI PFNGLINSERTEVENTMARKEREXTPROC glad_glInsertEventMarkerEXT;
10256 #define glInsertEventMarkerEXT glad_glInsertEventMarkerEXT
10257 typedef void (APIENTRY PFNGLPUSHGROUPMARKEREXTPROC) (GLsizei length, const GLchar *marker);
10258 GLAPI PFNGLPUSHGROUPMARKEREXTPROC glad_glPushGroupMarkerEXT;
10259 #define glPushGroupMarkerEXT glad_glPushGroupMarkerEXT
10260 typedef void (APIENTRY PFNGLPOPGROUPMARKEREXTPROC) (void);
10261 GLAPI PFNGLPOPGROUPMARKEREXTPROC glad_glPopGroupMarkerEXT;
10262 #define glPopGroupMarkerEXT glad_glPopGroupMarkerEXT
10263 #endif
10264 #ifndef GL_EXT_depth_bounds_test
10265 #define GL_EXT_depth_bounds_test 1
10266 GLAPI int GLAD_GL_EXT_depth_bounds_test;
10267 typedef void (APIENTRY PFNGLDEPTHBOUNDSEXTPROC) (GLclampd zmin, GLclampd zmax);
10268 GLAPI PFNGLDEPTHBOUNDSEXTPROC glad_glDepthBoundsEXT;
10269 #define glDepthBoundsEXT glad_glDepthBoundsEXT
10270 #endif
10271 #ifndef GL_EXT_direct_state_access
10272 #define GL_EXT_direct_state_access 1
10273 GLAPI int GLAD_GL_EXT_direct_state_access;
10274 typedef void (APIENTRY PFNGLMATRIXLOADFEXTPROC) (GLenum mode, const GLfloat *m);
10275 GLAPI PFNGLMATRIXLOADFEXTPROC glad_glMatrixLoadfEXT;
10276 #define glMatrixLoadfEXT glad_glMatrixLoadfEXT
10277 typedef void (APIENTRY PFNGLMATRIXLOADDEXTPROC) (GLenum mode, const GLdouble *m);
10278 GLAPI PFNGLMATRIXLOADDEXTPROC glad_glMatrixLoaddEXT;
10279 #define glMatrixLoaddEXT glad_glMatrixLoaddEXT
10280 typedef void (APIENTRY PFNGLMATRIXMULTFEXTPROC) (GLenum mode, const GLfloat *m);
10281 GLAPI PFNGLMATRIXMULTFEXTPROC glad_glMatrixMultfEXT;
10282 #define glMatrixMultfEXT glad_glMatrixMultfEXT
10283 typedef void (APIENTRY PFNGLMATRIXMULTDEXTPROC) (GLenum mode, const GLdouble *m);
10284 GLAPI PFNGLMATRIXMULTDEXTPROC glad_glMatrixMultdEXT;
10285 #define glMatrixMultdEXT glad_glMatrixMultdEXT
10286 typedef void (APIENTRY PFNGLMATRIXLOADIDENTITYEXTPROC) (GLenum mode);
10287 GLAPI PFNGLMATRIXLOADIDENTITYEXTPROC glad_glMatrixLoadIdentityEXT;
10288 #define glMatrixLoadIdentityEXT glad_glMatrixLoadIdentityEXT
10289 typedef void (APIENTRY PFNGLMATRIXROTATEFEXTPROC) (GLenum mode, GLfloat angle, GLfloat x, GLfloat y,
    GLfloat z);
10290 GLAPI PFNGLMATRIXROTATEFEXTPROC glad_glMatrixRotatfEXT;
10291 #define glMatrixRotatfEXT glad_glMatrixRotatfEXT
10292 typedef void (APIENTRY PFNGLMATRIXROTATEDEXTPROC) (GLenum mode, GLdouble angle, GLdouble x, GLdouble
    y, GLdouble z);
10293 GLAPI PFNGLMATRIXROTATEDEXTPROC glad_glMatrixRotatedEXT;

```

```

10294 #define glMatrixRotatedEXT glad_glMatrixRotatedEXT
10295 typedef void (APIENTRY PFNGLMATRIXSCALEFEXTPROC) (GLenum mode, GLfloat x, GLfloat y, GLfloat z);
10296 GLAPI PFNGLMATRIXSCALEFEXTPROC glad_glMatrixScalefEXT;
10297 #define glMatrixScalefEXT glad_glMatrixScalefEXT
10298 typedef void (APIENTRY PFNGLMATRIXSCALEDEXTPROC) (GLenum mode, GLdouble x, GLdouble y, GLdouble z);
10299 GLAPI PFNGLMATRIXSCALEDEXTPROC glad_glMatrixScaledEXT;
10300 #define glMatrixScaledEXT glad_glMatrixScaledEXT
10301 typedef void (APIENTRY PFNGLMATRIXTRANSLATEFEXTPROC) (GLenum mode, GLfloat x, GLfloat y, GLfloat z);
10302 GLAPI PFNGLMATRIXTRANSLATEFEXTPROC glad_glMatrixTranslatefEXT;
10303 #define glMatrixTranslatefEXT glad_glMatrixTranslatefEXT
10304 typedef void (APIENTRY PFNGLMATRIXTRANSLATEDEXTPROC) (GLenum mode, GLdouble x, GLdouble y, GLdouble
z);
10305 GLAPI PFNGLMATRIXTRANSLATEDEXTPROC glad_glMatrixTranslatedEXT;
10306 #define glMatrixTranslatedEXT glad_glMatrixTranslatedEXT
10307 typedef void (APIENTRY PFNGLMATRIXFRUSTUMEXTPROC) (GLenum mode, GLdouble left, GLdouble right,
GLdouble bottom, GLdouble top, GLdouble zNear, GLdouble zFar);
10308 GLAPI PFNGLMATRIXFRUSTUMEXTPROC glad_glMatrixFrustumEXT;
10309 #define glMatrixFrustumEXT glad_glMatrixFrustumEXT
10310 typedef void (APIENTRY PFNGLMATRIXORTHOEXTPROC) (GLenum mode, GLdouble left, GLdouble right, GLdouble
bottom, GLdouble top, GLdouble zNear, GLdouble zFar);
10311 GLAPI PFNGLMATRIXORTHOEXTPROC glad_glMatrixOrthoEXT;
10312 #define glMatrixOrthoEXT glad_glMatrixOrthoEXT
10313 typedef void (APIENTRY PFNGLMATRIXPOPEXTPROC) (GLenum mode);
10314 GLAPI PFNGLMATRIXPOPEXTPROC glad_glMatrixPopEXT;
10315 #define glMatrixPopEXT glad_glMatrixPopEXT
10316 typedef void (APIENTRY PFNGLMATRIXPUSHEXTPROC) (GLenum mode);
10317 GLAPI PFNGLMATRIXPUSHEXTPROC glad_glMatrixPushEXT;
10318 #define glMatrixPushEXT glad_glMatrixPushEXT
10319 typedef void (APIENTRY PFNGLCLIENTATTRIBDEFAULTEXTPROC) (GLbitfield mask);
10320 GLAPI PFNGLCLIENTATTRIBDEFAULTEXTPROC glad_glClientAttribDefaultEXT;
10321 #define glClientAttribDefaultEXT glad_glClientAttribDefaultEXT
10322 typedef void (APIENTRY PFNGLPUSHCLIENTATTRIBDEFAULTEXTPROC) (GLbitfield mask);
10323 GLAPI PFNGLPUSHCLIENTATTRIBDEFAULTEXTPROC glad_glPushClientAttribDefaultEXT;
10324 #define glPushClientAttribDefaultEXT glad_glPushClientAttribDefaultEXT
10325 typedef void (APIENTRY PFNGLTEXTUREPARAMETERFEXTPROC) (GLuint texture, GLenum target, GLenum pname,
GLfloat param);
10326 GLAPI PFNGLTEXTUREPARAMETERFEXTPROC glad_glTextureParameterfEXT;
10327 #define glTextureParameterfEXT glad_glTextureParameterfEXT
10328 typedef void (APIENTRY PFNGLTEXTUREPARAMETERFVEXTPROC) (GLuint texture, GLenum target, GLenum pname,
const GLfloat *params);
10329 GLAPI PFNGLTEXTUREPARAMETERFVEXTPROC glad_glTextureParameterfvEXT;
10330 #define glTextureParameterfvEXT glad_glTextureParameterfvEXT
10331 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIEXTPROC) (GLuint texture, GLenum target, GLenum pname,
GLint param);
10332 GLAPI PFNGLTEXTUREPARAMETERIEXTPROC glad_glTextureParameteriEXT;
10333 #define glTextureParameteriEXT glad_glTextureParameteriEXT
10334 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIVEXTPROC) (GLuint texture, GLenum target, GLenum pname,
const GLint *params);
10335 GLAPI PFNGLTEXTUREPARAMETERIVEXTPROC glad_glTextureParameterivEXT;
10336 #define glTextureParameterivEXT glad_glTextureParameterivEXT
10337 typedef void (APIENTRY PFNGLTEXTUREIMAGE1DEXTPROC) (GLuint texture, GLenum target, GLint level, GLint
internalformat, GLsizei width, GLint border, GLenum format, GLenum type, const void *pixels);
10338 GLAPI PFNGLTEXTUREIMAGE1DEXTPROC glad_glTextureImage1DEXT;
10339 #define glTextureImage1DEXT glad_glTextureImage1DEXT
10340 typedef void (APIENTRY PFNGLTEXTUREIMAGE2DEXTPROC) (GLuint texture, GLenum target, GLint level, GLint
internalformat, GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const void
*pixels);
10341 GLAPI PFNGLTEXTUREIMAGE2DEXTPROC glad_glTextureImage2DEXT;
10342 #define glTextureImage2DEXT glad_glTextureImage2DEXT
10343 typedef void (APIENTRY PFNGLTEXTURESUBIMAGE1DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLint xoffset, GLsizei width, GLenum format, GLenum type, const void *pixels);
10344 GLAPI PFNGLTEXTURESUBIMAGE1DEXTPROC glad_glTextureSubImage1DEXT;
10345 #define glTextureSubImage1DEXT glad_glTextureSubImage1DEXT
10346 typedef void (APIENTRY PFNGLTEXTURESUBIMAGE2DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void
*pixels);
10347 GLAPI PFNGLTEXTURESUBIMAGE2DEXTPROC glad_glTextureSubImage2DEXT;
10348 #define glTextureSubImage2DEXT glad_glTextureSubImage2DEXT
10349 typedef void (APIENTRY PFNGLCOPYTEXTUREIMAGE1DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLenum internalformat, GLint x, GLint y, GLsizei width, GLint border);
10350 GLAPI PFNGLCOPYTEXTUREIMAGE1DEXTPROC glad_glCopyTextureImage1DEXT;
10351 #define glCopyTextureImage1DEXT glad_glCopyTextureImage1DEXT
10352 typedef void (APIENTRY PFNGLCOPYTEXTUREIMAGE2DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
10353 GLAPI PFNGLCOPYTEXTUREIMAGE2DEXTPROC glad_glCopyTextureImage2DEXT;
10354 #define glCopyTextureImage2DEXT glad_glCopyTextureImage2DEXT
10355 typedef void (APIENTRY PFNGLCOPYTEXTURESUBIMAGE1DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLint xoffset, GLint x, GLint y, GLsizei width);
10356 GLAPI PFNGLCOPYTEXTURESUBIMAGE1DEXTPROC glad_glCopyTextureSubImage1DEXT;
10357 #define glCopyTextureSubImage1DEXT glad_glCopyTextureSubImage1DEXT
10358 typedef void (APIENTRY PFNGLCOPYTEXTURESUBIMAGE2DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
10359 GLAPI PFNGLCOPYTEXTURESUBIMAGE2DEXTPROC glad_glCopyTextureSubImage2DEXT;
10360 #define glCopyTextureSubImage2DEXT glad_glCopyTextureSubImage2DEXT
10361 typedef void (APIENTRY PFNGLGETTEXTUREIMAGEEXTPROC) (GLuint texture, GLenum target, GLint level,
GLenum format, GLenum type, void *pixels);
10362 GLAPI PFNGLGETTEXTUREIMAGEEXTPROC glad_glGetTextureImageEXT;

```



```

10363 #define glGetTextureImageEXT glad_glGetTextureImageEXT
10364 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERFVEXTPROC) (GLuint texture, GLenum target, GLenum
pname, GLfloat *params);
10365 GLAPI PFNGLGETTEXTUREPARAMETERFVEXTPROC glad_glGetTextureParameterfvEXT;
10366 #define glGetTextureParameterfvEXT glad_glGetTextureParameterfvEXT
10367 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERIVEXTPROC) (GLuint texture, GLenum target, GLenum
pname, GLint *params);
10368 GLAPI PFNGLGETTEXTUREPARAMETERIVEXTPROC glad_glGetTextureParameterivEXT;
10369 #define glGetTextureParameterivEXT glad_glGetTextureParameterivEXT
10370 typedef void (APIENTRY PFNGLGETTEXTURELEVELPARAMETERFVEXTPROC) (GLuint texture, GLenum target, GLint
level, GLenum pname, GLfloat *params);
10371 GLAPI PFNGLGETTEXTURELEVELPARAMETERFVEXTPROC glad_glGetTextureLevelParameterfvEXT;
10372 #define glGetTextureLevelParameterfvEXT glad_glGetTextureLevelParameterfvEXT
10373 typedef void (APIENTRY PFNGLGETTEXTURELEVELPARAMETERIVEXTPROC) (GLuint texture, GLenum target, GLint
level, GLenum pname, GLint *params);
10374 GLAPI PFNGLGETTEXTURELEVELPARAMETERIVEXTPROC glad_glGetTextureLevelParameterivEXT;
10375 #define glGetTextureLevelParameterivEXT glad_glGetTextureLevelParameterivEXT
10376 typedef void (APIENTRY PFNGLTEXTUREIMAGE3DEXTPROC) (GLuint texture, GLenum target, GLint level, GLint
internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLenum format, GLenum
type, const void *pixels);
10377 GLAPI PFNGLTEXTUREIMAGE3DEXTPROC glad_glTextureImage3DEXT;
10378 #define glTextureImage3DEXT glad_glTextureImage3DEXT
10379 typedef void (APIENTRY PFNGLTEXTURESUBIMAGE3DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum
format, GLenum type, const void *pixels);
10380 GLAPI PFNGLTEXTURESUBIMAGE3DEXTPROC glad_glTextureSubImage3DEXT;
10381 #define glTextureSubImage3DEXT glad_glTextureSubImage3DEXT
10382 typedef void (APIENTRY PFNGLCOPYTEXTURESUBIMAGE3DEXTPROC) (GLuint texture, GLenum target, GLint level,
GLint xoffset, GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
10383 GLAPI PFNGLCOPYTEXTURESUBIMAGE3DEXTPROC glad_glCopyTextureSubImage3DEXT;
10384 #define glCopyTextureSubImage3DEXT glad_glCopyTextureSubImage3DEXT
10385 typedef void (APIENTRY PFNGLBINDMULTITEXTUREEXTPROC) (GLenum texunit, GLenum target, GLuint texture);
10386 GLAPI PFNGLBINDMULTITEXTUREEXTPROC glad_glBindMultiTextureEXT;
10387 #define glBindMultiTextureEXT glad_glBindMultiTextureEXT
10388 typedef void (APIENTRY PFNGLMULTITEXCOORDPOINTEREXTPROC) (GLenum texunit, GLint size, GLenum type,
GLsizei stride, const void *pointer);
10389 GLAPI PFNGLMULTITEXCOORDPOINTEREXTPROC glad_glMultiTexCoordPointerEXT;
10390 #define glMultiTexCoordPointerEXT glad_glMultiTexCoordPointerEXT
10391 typedef void (APIENTRY PFNGLMULTITEXENVFEXTPROC) (GLenum texunit, GLenum target, GLenum pname, GLfloat
param);
10392 GLAPI PFNGLMULTITEXENVFEXTPROC glad_glMultiTexEnvfEXT;
10393 #define glMultiTexEnvfEXT glad_glMultiTexEnvfEXT
10394 typedef void (APIENTRY PFNGLMULTITEXENVFVEXTPROC) (GLenum texunit, GLenum target, GLenum pname, const
GLfloat *params);
10395 GLAPI PFNGLMULTITEXENVFVEXTPROC glad_glMultiTexEnvfvEXT;
10396 #define glMultiTexEnvfvEXT glad_glMultiTexEnvfvEXT
10397 typedef void (APIENTRY PFNGLMULTITEXENVIEVEXTPROC) (GLenum texunit, GLenum target, GLenum pname, GLint
param);
10398 GLAPI PFNGLMULTITEXENVIEVEXTPROC glad_glMultiTexEnvievEXT;
10399 #define glMultiTexEnvievEXT glad_glMultiTexEnvievEXT
10400 typedef void (APIENTRY PFNGLMULTITEXENVIVEXTPROC) (GLenum texunit, GLenum target, GLenum pname, const
GLint *params);
10401 GLAPI PFNGLMULTITEXENVIVEXTPROC glad_glMultiTexEnvivEXT;
10402 #define glMultiTexEnvivEXT glad_glMultiTexEnvivEXT
10403 typedef void (APIENTRY PFNGLMULTITEXGENDEXTPROC) (GLenum texunit, GLenum coord, GLenum pname, GLdouble
param);
10404 GLAPI PFNGLMULTITEXGENDEXTPROC glad_glMultiTexGendEXT;
10405 #define glMultiTexGendEXT glad_glMultiTexGendEXT
10406 typedef void (APIENTRY PFNGLMULTITEXGENDVEXTPROC) (GLenum texunit, GLenum coord, GLenum pname, const
GLdouble *params);
10407 GLAPI PFNGLMULTITEXGENDVEXTPROC glad_glMultiTexGendvEXT;
10408 #define glMultiTexGendvEXT glad_glMultiTexGendvEXT
10409 typedef void (APIENTRY PFNGLMULTITEXGENFEXTPROC) (GLenum texunit, GLenum coord, GLenum pname, GLfloat
param);
10410 GLAPI PFNGLMULTITEXGENFEXTPROC glad_glMultiTexGenfEXT;
10411 #define glMultiTexGenfEXT glad_glMultiTexGenfEXT
10412 typedef void (APIENTRY PFNGLMULTITEXGENFVEXTPROC) (GLenum texunit, GLenum coord, GLenum pname, const
GLfloat *params);
10413 GLAPI PFNGLMULTITEXGENFVEXTPROC glad_glMultiTexGenfvEXT;
10414 #define glMultiTexGenfvEXT glad_glMultiTexGenfvEXT
10415 typedef void (APIENTRY PFNGLMULTITEXGENIEVEXTPROC) (GLenum texunit, GLenum coord, GLenum pname, GLint
param);
10416 GLAPI PFNGLMULTITEXGENIEVEXTPROC glad_glMultiTexGenievEXT;
10417 #define glMultiTexGenievEXT glad_glMultiTexGenievEXT
10418 typedef void (APIENTRY PFNGLMULTITEXGENIVEXTPROC) (GLenum texunit, GLenum coord, GLenum pname, const
GLint *params);
10419 GLAPI PFNGLMULTITEXGENIVEXTPROC glad_glMultiTexGenivEXT;
10420 #define glMultiTexGenivEXT glad_glMultiTexGenivEXT
10421 typedef void (APIENTRY PFNGLGETMULTITEXENVFVEXTPROC) (GLenum texunit, GLenum target, GLenum pname,
GLfloat *params);
10422 GLAPI PFNGLGETMULTITEXENVFVEXTPROC glad_glGetMultiTexEnvfvEXT;
10423 #define glGetMultiTexEnvfvEXT glad_glGetMultiTexEnvfvEXT
10424 typedef void (APIENTRY PFNGLGETMULTITEXENVIVEXTPROC) (GLenum texunit, GLenum target, GLenum pname,
GLint *params);
10425 GLAPI PFNGLGETMULTITEXENVIVEXTPROC glad_glGetMultiTexEnvivEXT;
10426 #define glGetMultiTexEnvivEXT glad_glGetMultiTexEnvivEXT
10427 typedef void (APIENTRY PFNGLGETMULTITEXGENDVEXTPROC) (GLenum texunit, GLenum coord, GLenum pname,

```

```

    GLdouble *params);
10428 GLAPI PFNGLGETMULTITEXGENDVEXTPROC glad_glGetMultiTexGendvEXT;
10429 #define glGetMultiTexGendvEXT glad_glGetMultiTexGendvEXT
10430 typedef void (APIENTRYP PFNGLGETMULTITEXGENFVEXTPROC) (GLenum texunit, GLenum coord, GLenum pname,
    GLfloat *params);
10431 GLAPI PFNGLGETMULTITEXGENFVEXTPROC glad_glGetMultiTexGenfvEXT;
10432 #define glGetMultiTexGenfvEXT glad_glGetMultiTexGenfvEXT
10433 typedef void (APIENTRYP PFNGLGETMULTITEXGENIVEXTPROC) (GLenum texunit, GLenum coord, GLenum pname,
    GLint *params);
10434 GLAPI PFNGLGETMULTITEXGENIVEXTPROC glad_glGetMultiTexGenivEXT;
10435 #define glGetMultiTexGenivEXT glad_glGetMultiTexGenivEXT
10436 typedef void (APIENTRYP PFNGLMULTITEXPARAMETERIEXTPROC) (GLenum texunit, GLenum target, GLenum pname,
    GLint param);
10437 GLAPI PFNGLMULTITEXPARAMETERIEXTPROC glad_glMultiTexParameteriEXT;
10438 #define glMultiTexParameteriEXT glad_glMultiTexParameteriEXT
10439 typedef void (APIENTRYP PFNGLMULTITEXPARAMETERIVEXTPROC) (GLenum texunit, GLenum target, GLenum pname,
    const GLint *params);
10440 GLAPI PFNGLMULTITEXPARAMETERIVEXTPROC glad_glMultiTexParameterivEXT;
10441 #define glMultiTexParameterivEXT glad_glMultiTexParameterivEXT
10442 typedef void (APIENTRYP PFNGLMULTITEXPARAMETERFEXTPROC) (GLenum texunit, GLenum target, GLenum pname,
    GLfloat param);
10443 GLAPI PFNGLMULTITEXPARAMETERFEXTPROC glad_glMultiTexParameterfEXT;
10444 #define glMultiTexParameterfEXT glad_glMultiTexParameterfEXT
10445 typedef void (APIENTRYP PFNGLMULTITEXPARAMETERFVEXTPROC) (GLenum texunit, GLenum target, GLenum pname,
    const GLfloat *params);
10446 GLAPI PFNGLMULTITEXPARAMETERFVEXTPROC glad_glMultiTexParameterfvEXT;
10447 #define glMultiTexParameterfvEXT glad_glMultiTexParameterfvEXT
10448 typedef void (APIENTRYP PFNGLMULTITEXIMAGE1DEXTPROC) (GLenum texunit, GLenum target, GLint level, GLint
    internalformat, GLsizei width, GLint border, GLenum format, GLenum type, const void *pixels);
10449 GLAPI PFNGLMULTITEXIMAGE1DEXTPROC glad_glMultiTexImage1DEXT;
10450 #define glMultiTexImage1DEXT glad_glMultiTexImage1DEXT
10451 typedef void (APIENTRYP PFNGLMULTITEXIMAGE2DEXTPROC) (GLenum texunit, GLenum target, GLint level, GLint
    internalformat, GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const void
    *pixels);
10452 GLAPI PFNGLMULTITEXIMAGE2DEXTPROC glad_glMultiTexImage2DEXT;
10453 #define glMultiTexImage2DEXT glad_glMultiTexImage2DEXT
10454 typedef void (APIENTRYP PFNGLMULTITEXSUBIMAGE1DEXTPROC) (GLenum texunit, GLenum target, GLint level,
    GLint xoffset, GLsizei width, GLenum format, GLenum type, const void *pixels);
10455 GLAPI PFNGLMULTITEXSUBIMAGE1DEXTPROC glad_glMultiTexSubImage1DEXT;
10456 #define glMultiTexSubImage1DEXT glad_glMultiTexSubImage1DEXT
10457 typedef void (APIENTRYP PFNGLMULTITEXSUBIMAGE2DEXTPROC) (GLenum texunit, GLenum target, GLint level,
    GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void
    *pixels);
10458 GLAPI PFNGLMULTITEXSUBIMAGE2DEXTPROC glad_glMultiTexSubImage2DEXT;
10459 #define glMultiTexSubImage2DEXT glad_glMultiTexSubImage2DEXT
10460 typedef void (APIENTRYP PFNGLCOPYMULTITEXIMAGE1DEXTPROC) (GLenum texunit, GLenum target, GLint level,
    GLenum internalformat, GLint x, GLint y, GLsizei width, GLint border);
10461 GLAPI PFNGLCOPYMULTITEXIMAGE1DEXTPROC glad_glCopyMultiTexImage1DEXT;
10462 #define glCopyMultiTexImage1DEXT glad_glCopyMultiTexImage1DEXT
10463 typedef void (APIENTRYP PFNGLCOPYMULTITEXIMAGE2DEXTPROC) (GLenum texunit, GLenum target, GLint level,
    GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
10464 GLAPI PFNGLCOPYMULTITEXIMAGE2DEXTPROC glad_glCopyMultiTexImage2DEXT;
10465 #define glCopyMultiTexImage2DEXT glad_glCopyMultiTexImage2DEXT
10466 typedef void (APIENTRYP PFNGLCOPYMULTITEXSUBIMAGE1DEXTPROC) (GLenum texunit, GLenum target, GLint
    level, GLint xoffset, GLint x, GLint y, GLsizei width);
10467 GLAPI PFNGLCOPYMULTITEXSUBIMAGE1DEXTPROC glad_glCopyMultiTexSubImage1DEXT;
10468 #define glCopyMultiTexSubImage1DEXT glad_glCopyMultiTexSubImage1DEXT
10469 typedef void (APIENTRYP PFNGLCOPYMULTITEXSUBIMAGE2DEXTPROC) (GLenum texunit, GLenum target, GLint
    level, GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
10470 GLAPI PFNGLCOPYMULTITEXSUBIMAGE2DEXTPROC glad_glCopyMultiTexSubImage2DEXT;
10471 #define glCopyMultiTexSubImage2DEXT glad_glCopyMultiTexSubImage2DEXT
10472 typedef void (APIENTRYP PFNGLGETMULTITEXIMAGEEXTPROC) (GLenum texunit, GLenum target, GLint level,
    GLenum format, GLenum type, void *pixels);
10473 GLAPI PFNGLGETMULTITEXIMAGEEXTPROC glad_glGetMultiTexImageEXT;
10474 #define glGetMultiTexImageEXT glad_glGetMultiTexImageEXT
10475 typedef void (APIENTRYP PFNGLGETMULTITEXPARAMETERFVEXTPROC) (GLenum texunit, GLenum target, GLenum
    pname, GLfloat *params);
10476 GLAPI PFNGLGETMULTITEXPARAMETERFVEXTPROC glad_glGetMultiTexParameterfvEXT;
10477 #define glGetMultiTexParameterfvEXT glad_glGetMultiTexParameterfvEXT
10478 typedef void (APIENTRYP PFNGLGETMULTITEXPARAMETERIVEXTPROC) (GLenum texunit, GLenum target, GLenum
    pname, GLint *params);
10479 GLAPI PFNGLGETMULTITEXPARAMETERIVEXTPROC glad_glGetMultiTexParameterivEXT;
10480 #define glGetMultiTexParameterivEXT glad_glGetMultiTexParameterivEXT
10481 typedef void (APIENTRYP PFNGLGETMULTITEXLEVELPARAMETERFVEXTPROC) (GLenum texunit, GLenum target, GLint
    level, GLenum pname, GLfloat *params);
10482 GLAPI PFNGLGETMULTITEXLEVELPARAMETERFVEXTPROC glad_glGetMultiTexLevelParameterfvEXT;
10483 #define glGetMultiTexLevelParameterfvEXT glad_glGetMultiTexLevelParameterfvEXT
10484 typedef void (APIENTRYP PFNGLGETMULTITEXLEVELPARAMETERIVEXTPROC) (GLenum texunit, GLenum target, GLint
    level, GLenum pname, GLint *params);
10485 GLAPI PFNGLGETMULTITEXLEVELPARAMETERIVEXTPROC glad_glGetMultiTexLevelParameterivEXT;
10486 #define glGetMultiTexLevelParameterivEXT glad_glGetMultiTexLevelParameterivEXT
10487 typedef void (APIENTRYP PFNGLMULTITEXIMAGE3DEXTPROC) (GLenum texunit, GLenum target, GLint level, GLint
    internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLenum format, GLenum
    type, const void *pixels);
10488 GLAPI PFNGLMULTITEXIMAGE3DEXTPROC glad_glMultiTexImage3DEXT;
10489 #define glMultiTexImage3DEXT glad_glMultiTexImage3DEXT
10490 typedef void (APIENTRYP PFNGLMULTITEXSUBIMAGE3DEXTPROC) (GLenum texunit, GLenum target, GLint level,

```



```

        GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum
        format, GLenum type, const void *pixels);
10491 GLAPI PFNGLMULTITEXSUBIMAGE3DEXTPROC glad_glMultiTexSubImage3DEXT;
10492 #define glMultiTexSubImage3DEXT glad_glMultiTexSubImage3DEXT
10493 typedef void (APIENTRY PFNGLCOPYMULTITEXSUBIMAGE3DEXTPROC)(GLenum texunit, GLenum target, GLint
        level, GLint xoffset, GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
10494 GLAPI PFNGLCOPYMULTITEXSUBIMAGE3DEXTPROC glad_glCopyMultiTexSubImage3DEXT;
10495 #define glCopyMultiTexSubImage3DEXT glad_glCopyMultiTexSubImage3DEXT
10496 typedef void (APIENTRY PFNGLENABLECLIENTSTATEINDEXEDEXTPROC)(GLenum array, GLuint index);
10497 GLAPI PFNGLENABLECLIENTSTATEINDEXEDEXTPROC glad_glEnableClientStateIndexedEXT;
10498 #define glEnableClientStateIndexedEXT glad_glEnableClientStateIndexedEXT
10499 typedef void (APIENTRY PFNGLDISABLECLIENTSTATEINDEXEDEXTPROC)(GLenum array, GLuint index);
10500 GLAPI PFNGLDISABLECLIENTSTATEINDEXEDEXTPROC glad_glDisableClientStateIndexedEXT;
10501 #define glDisableClientStateIndexedEXT glad_glDisableClientStateIndexedEXT
10502 typedef void (APIENTRY PFNGLGETFLOATINDEXEDVEXTPROC)(GLenum target, GLuint index, GLfloat *data);
10503 GLAPI PFNGLGETFLOATINDEXEDVEXTPROC glad_glGetFloatIndexedvEXT;
10504 #define glGetFloatIndexedvEXT glad_glGetFloatIndexedvEXT
10505 typedef void (APIENTRY PFNGLGETDOUBLEINDEXEDVEXTPROC)(GLenum target, GLuint index, GLdouble *data);
10506 GLAPI PFNGLGETDOUBLEINDEXEDVEXTPROC glad_glGetDoubleIndexedvEXT;
10507 #define glGetDoubleIndexedvEXT glad_glGetDoubleIndexedvEXT
10508 typedef void (APIENTRY PFNGLGETPOINTERINDEXEDVEXTPROC)(GLenum target, GLuint index, void **data);
10509 GLAPI PFNGLGETPOINTERINDEXEDVEXTPROC glad_glGetPointerIndexedvEXT;
10510 #define glGetPointerIndexedvEXT glad_glGetPointerIndexedvEXT
10511 typedef void (APIENTRY PFNGLENABLEINDEXEDEXTPROC)(GLenum target, GLuint index);
10512 GLAPI PFNGLENABLEINDEXEDEXTPROC glad_glEnableIndexedEXT;
10513 #define glEnableIndexedEXT glad_glEnableIndexedEXT
10514 typedef void (APIENTRY PFNGLDISABLEINDEXEDEXTPROC)(GLenum target, GLuint index);
10515 GLAPI PFNGLDISABLEINDEXEDEXTPROC glad_glDisableIndexedEXT;
10516 #define glDisableIndexedEXT glad_glDisableIndexedEXT
10517 typedef GLboolean (APIENTRY PFNGLISENABLEDINDEXEDEXTPROC)(GLenum target, GLuint index);
10518 GLAPI PFNGLISENABLEDINDEXEDEXTPROC glad_glIsEnabledIndexedEXT;
10519 #define glIsEnabledIndexedEXT glad_glIsEnabledIndexedEXT
10520 typedef void (APIENTRY PFNGLGETINTEGERINDEXEDVEXTPROC)(GLenum target, GLuint index, GLint *data);
10521 GLAPI PFNGLGETINTEGERINDEXEDVEXTPROC glad_glGetIntegerIndexedvEXT;
10522 #define glGetIntegerIndexedvEXT glad_glGetIntegerIndexedvEXT
10523 typedef void (APIENTRY PFNGLGETBOOLEANINDEXEDVEXTPROC)(GLenum target, GLuint index, GLboolean *data);
10524 GLAPI PFNGLGETBOOLEANINDEXEDVEXTPROC glad_glGetBooleanIndexedvEXT;
10525 #define glGetBooleanIndexedvEXT glad_glGetBooleanIndexedvEXT
10526 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTUREIMAGE3DEXTPROC)(GLuint texture, GLenum target, GLint
        level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei
        imageSize, const void *bits);
10527 GLAPI PFNGLCOMPRESSEDTEXTUREIMAGE3DEXTPROC glad_glCompressedTextureImage3DEXT;
10528 #define glCompressedTextureImage3DEXT glad_glCompressedTextureImage3DEXT
10529 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTUREIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint
        level, GLenum internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const
        void *bits);
10530 GLAPI PFNGLCOMPRESSEDTEXTUREIMAGE2DEXTPROC glad_glCompressedTextureImage2DEXT;
10531 #define glCompressedTextureImage2DEXT glad_glCompressedTextureImage2DEXT
10532 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTUREIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint
        level, GLenum internalformat, GLsizei width, GLint border, GLsizei imageSize, const void *bits);
10533 GLAPI PFNGLCOMPRESSEDTEXTUREIMAGE1DEXTPROC glad_glCompressedTextureImage1DEXT;
10534 #define glCompressedTextureImage1DEXT glad_glCompressedTextureImage1DEXT
10535 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTURESUBIMAGE3DEXTPROC)(GLuint texture, GLenum target, GLint
        level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth,
        GLenum format, GLsizei imageSize, const void *bits);
10536 GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE3DEXTPROC glad_glCompressedTextureSubImage3DEXT;
10537 #define glCompressedTextureSubImage3DEXT glad_glCompressedTextureSubImage3DEXT
10538 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTURESUBIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint
        level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize,
        const void *bits);
10539 GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE2DEXTPROC glad_glCompressedTextureSubImage2DEXT;
10540 #define glCompressedTextureSubImage2DEXT glad_glCompressedTextureSubImage2DEXT
10541 typedef void (APIENTRY PFNGLCOMPRESSEDTEXTURESUBIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint
        level, GLint xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *bits);
10542 GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE1DEXTPROC glad_glCompressedTextureSubImage1DEXT;
10543 #define glCompressedTextureSubImage1DEXT glad_glCompressedTextureSubImage1DEXT
10544 typedef void (APIENTRY PFNGLGETCOMPRESSEDTEXTUREIMAGEEXTPROC)(GLuint texture, GLenum target, GLint
        lod, void *img);
10545 GLAPI PFNGLGETCOMPRESSEDTEXTUREIMAGEEXTPROC glad_glGetCompressedTextureImageEXT;
10546 #define glGetCompressedTextureImageEXT glad_glGetCompressedTextureImageEXT
10547 typedef void (APIENTRY PFNGLCOMPRESSEDMULTITEXIMAGE3DEXTPROC)(GLenum texunit, GLenum target, GLint
        level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei
        imageSize, const void *bits);
10548 GLAPI PFNGLCOMPRESSEDMULTITEXIMAGE3DEXTPROC glad_glCompressedMultiTexImage3DEXT;
10549 #define glCompressedMultiTexImage3DEXT glad_glCompressedMultiTexImage3DEXT
10550 typedef void (APIENTRY PFNGLCOMPRESSEDMULTITEXIMAGE2DEXTPROC)(GLenum texunit, GLenum target, GLint
        level, GLenum internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const
        void *bits);
10551 GLAPI PFNGLCOMPRESSEDMULTITEXIMAGE2DEXTPROC glad_glCompressedMultiTexImage2DEXT;
10552 #define glCompressedMultiTexImage2DEXT glad_glCompressedMultiTexImage2DEXT
10553 typedef void (APIENTRY PFNGLCOMPRESSEDMULTITEXIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint
        level, GLenum internalformat, GLsizei width, GLint border, GLsizei imageSize, const void *bits);
10554 GLAPI PFNGLCOMPRESSEDMULTITEXIMAGE1DEXTPROC glad_glCompressedMultiTexImage1DEXT;
10555 #define glCompressedMultiTexImage1DEXT glad_glCompressedMultiTexImage1DEXT
10556 typedef void (APIENTRY PFNGLCOMPRESSEDMULTITEXSUBIMAGE3DEXTPROC)(GLenum texunit, GLenum target, GLint
        level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth,
        GLenum format, GLsizei imageSize, const void *bits);

```

```

10557 GLAPI PFNGLCOMPRESSEDMULTITEXSUBIMAGE3DEXTPROC glad_glCompressedMultiTexSubImage3DEXT;
10558 #define glCompressedMultiTexSubImage3DEXT glad_glCompressedMultiTexSubImage3DEXT
10559 typedef void (APIENTRY PFNGLCOMPRESSEDMULTITEXSUBIMAGE2DEXTPROC)(GLenum texunit, GLenum target, GLint
level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize,
const void *bits);
10560 GLAPI PFNGLCOMPRESSEDMULTITEXSUBIMAGE2DEXTPROC glad_glCompressedMultiTexSubImage2DEXT;
10561 #define glCompressedMultiTexSubImage2DEXT glad_glCompressedMultiTexSubImage2DEXT
10562 typedef void (APIENTRY PFNGLCOMPRESSEDMULTITEXSUBIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint
level, GLint xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *bits);
10563 GLAPI PFNGLCOMPRESSEDMULTITEXSUBIMAGE1DEXTPROC glad_glCompressedMultiTexSubImage1DEXT;
10564 #define glCompressedMultiTexSubImage1DEXT glad_glCompressedMultiTexSubImage1DEXT
10565 typedef void (APIENTRY PFNGLGETCOMPRESSEDMULTITEXIMAGEEXTPROC)(GLenum texunit, GLenum target, GLint
lod, void *img);
10566 GLAPI PFNGLGETCOMPRESSEDMULTITEXIMAGEEXTPROC glad_glGetCompressedMultiTexImageEXT;
10567 #define glGetCompressedMultiTexImageEXT glad_glGetCompressedMultiTexImageEXT
10568 typedef void (APIENTRY PFNGLMATRIXLOADTRANSPOSEFEXTPROC)(GLenum mode, const GLfloat *m);
10569 GLAPI PFNGLMATRIXLOADTRANSPOSEFEXTPROC glad_glMatrixLoadTransposeEXT;
10570 #define glMatrixLoadTransposeEXT glad_glMatrixLoadTransposeEXT
10571 typedef void (APIENTRY PFNGLMATRIXLOADTRANSPOSEDEXTPROC)(GLenum mode, const GLdouble *m);
10572 GLAPI PFNGLMATRIXLOADTRANSPOSEDEXTPROC glad_glMatrixLoadTransposedEXT;
10573 #define glMatrixLoadTransposedEXT glad_glMatrixLoadTransposedEXT
10574 typedef void (APIENTRY PFNGLMATRIXMULTTRANSPOSEFEXTPROC)(GLenum mode, const GLfloat *m);
10575 GLAPI PFNGLMATRIXMULTTRANSPOSEFEXTPROC glad_glMatrixMultTransposeEXT;
10576 #define glMatrixMultTransposeEXT glad_glMatrixMultTransposeEXT
10577 typedef void (APIENTRY PFNGLMATRIXMULTTRANSPOSEDEXTPROC)(GLenum mode, const GLdouble *m);
10578 GLAPI PFNGLMATRIXMULTTRANSPOSEDEXTPROC glad_glMatrixMultTransposedEXT;
10579 #define glMatrixMultTransposedEXT glad_glMatrixMultTransposedEXT
10580 typedef void (APIENTRY PFNGLNAMEDBUFFERDATAEXTPROC)(GLuint buffer, GLsizeiptr size, const void *data,
GLenum usage);
10581 GLAPI PFNGLNAMEDBUFFERDATAEXTPROC glad_glNamedBufferDataEXT;
10582 #define glNamedBufferDataEXT glad_glNamedBufferDataEXT
10583 typedef void (APIENTRY PFNGLNAMEDBUFFERSUBDATAEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr
size, const void *data);
10584 GLAPI PFNGLNAMEDBUFFERSUBDATAEXTPROC glad_glNamedBufferSubDataEXT;
10585 #define glNamedBufferSubDataEXT glad_glNamedBufferSubDataEXT
10586 typedef void * (APIENTRY PFNGLMAPNAMEDBUFFEREXTPROC)(GLuint buffer, GLenum access);
10587 GLAPI PFNGLMAPNAMEDBUFFEREXTPROC glad_glMapNamedBufferEXT;
10588 #define glMapNamedBufferEXT glad_glMapNamedBufferEXT
10589 typedef GLboolean (APIENTRY PFNGLUNMAPNAMEDBUFFEREXTPROC)(GLuint buffer);
10590 GLAPI PFNGLUNMAPNAMEDBUFFEREXTPROC glad_glUnmapNamedBufferEXT;
10591 #define glUnmapNamedBufferEXT glad_glUnmapNamedBufferEXT
10592 typedef void (APIENTRY PFNGLGETNAMEDBUFFERPARAMETERIVEXTPROC)(GLuint buffer, GLenum pname, GLint
*params);
10593 GLAPI PFNGLGETNAMEDBUFFERPARAMETERIVEXTPROC glad_glGetNamedBufferParameterivEXT;
10594 #define glGetNamedBufferParameterivEXT glad_glGetNamedBufferParameterivEXT
10595 typedef void (APIENTRY PFNGLGETNAMEDBUFFERPOINTERVEXTPROC)(GLuint buffer, GLenum pname, void
**params);
10596 GLAPI PFNGLGETNAMEDBUFFERPOINTERVEXTPROC glad_glGetNamedBufferPointervEXT;
10597 #define glGetNamedBufferPointervEXT glad_glGetNamedBufferPointervEXT
10598 typedef void (APIENTRY PFNGLGETNAMEDBUFFERSUBDATAEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr
size, void *data);
10599 GLAPI PFNGLGETNAMEDBUFFERSUBDATAEXTPROC glad_glGetNamedBufferSubDataEXT;
10600 #define glGetNamedBufferSubDataEXT glad_glGetNamedBufferSubDataEXT
10601 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1FEXTPROC)(GLuint program, GLint location, GLfloat v0);
10602 GLAPI PFNGLPROGRAMUNIFORM1FEXTPROC glad_glProgramUniform1fEXT;
10603 #define glProgramUniform1fEXT glad_glProgramUniform1fEXT
10604 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2FEXTPROC)(GLuint program, GLint location, GLfloat v0,
GLfloat v1);
10605 GLAPI PFNGLPROGRAMUNIFORM2FEXTPROC glad_glProgramUniform2fEXT;
10606 #define glProgramUniform2fEXT glad_glProgramUniform2fEXT
10607 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3FEXTPROC)(GLuint program, GLint location, GLfloat v0,
GLfloat v1, GLfloat v2);
10608 GLAPI PFNGLPROGRAMUNIFORM3FEXTPROC glad_glProgramUniform3fEXT;
10609 #define glProgramUniform3fEXT glad_glProgramUniform3fEXT
10610 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4FEXTPROC)(GLuint program, GLint location, GLfloat v0,
GLfloat v1, GLfloat v2, GLfloat v3);
10611 GLAPI PFNGLPROGRAMUNIFORM4FEXTPROC glad_glProgramUniform4fEXT;
10612 #define glProgramUniform4fEXT glad_glProgramUniform4fEXT
10613 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1IEXTPROC)(GLuint program, GLint location, GLint v0);
10614 GLAPI PFNGLPROGRAMUNIFORM1IEXTPROC glad_glProgramUniform1iEXT;
10615 #define glProgramUniform1iEXT glad_glProgramUniform1iEXT
10616 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2IEXTPROC)(GLuint program, GLint location, GLint v0, GLint
v1);
10617 GLAPI PFNGLPROGRAMUNIFORM2IEXTPROC glad_glProgramUniform2iEXT;
10618 #define glProgramUniform2iEXT glad_glProgramUniform2iEXT
10619 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3IEXTPROC)(GLuint program, GLint location, GLint v0, GLint
v1, GLint v2);
10620 GLAPI PFNGLPROGRAMUNIFORM3IEXTPROC glad_glProgramUniform3iEXT;
10621 #define glProgramUniform3iEXT glad_glProgramUniform3iEXT
10622 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4IEXTPROC)(GLuint program, GLint location, GLint v0, GLint
v1, GLint v2, GLint v3);
10623 GLAPI PFNGLPROGRAMUNIFORM4IEXTPROC glad_glProgramUniform4iEXT;
10624 #define glProgramUniform4iEXT glad_glProgramUniform4iEXT
10625 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1FVEXTPROC)(GLuint program, GLint location, GLsizei count,
const GLfloat *value);
10626 GLAPI PFNGLPROGRAMUNIFORM1FVEXTPROC glad_glProgramUniform1fvEXT;
10627 #define glProgramUniform1fvEXT glad_glProgramUniform1fvEXT

```

```

10628 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2FVEXTPROC) (GLuint program, GLint location, GLsizei count,
10629 const GLfloat *value);
10629 GLAPI PFNGLPROGRAMUNIFORM2FVEXTPROC glad_glProgramUniform2fvEXT;
10630 #define glProgramUniform2fvEXT glad_glProgramUniform2fvEXT
10631 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3FVEXTPROC) (GLuint program, GLint location, GLsizei count,
10632 const GLfloat *value);
10632 GLAPI PFNGLPROGRAMUNIFORM3FVEXTPROC glad_glProgramUniform3fvEXT;
10633 #define glProgramUniform3fvEXT glad_glProgramUniform3fvEXT
10634 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4FVEXTPROC) (GLuint program, GLint location, GLsizei count,
10635 const GLfloat *value);
10635 GLAPI PFNGLPROGRAMUNIFORM4FVEXTPROC glad_glProgramUniform4fvEXT;
10636 #define glProgramUniform4fvEXT glad_glProgramUniform4fvEXT
10637 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1IVEXTPROC) (GLuint program, GLint location, GLsizei count,
10638 const GLint *value);
10638 GLAPI PFNGLPROGRAMUNIFORM1IVEXTPROC glad_glProgramUniform1ivEXT;
10639 #define glProgramUniform1ivEXT glad_glProgramUniform1ivEXT
10640 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2IVEXTPROC) (GLuint program, GLint location, GLsizei count,
10641 const GLint *value);
10641 GLAPI PFNGLPROGRAMUNIFORM2IVEXTPROC glad_glProgramUniform2ivEXT;
10642 #define glProgramUniform2ivEXT glad_glProgramUniform2ivEXT
10643 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3IVEXTPROC) (GLuint program, GLint location, GLsizei count,
10644 const GLint *value);
10644 GLAPI PFNGLPROGRAMUNIFORM3IVEXTPROC glad_glProgramUniform3ivEXT;
10645 #define glProgramUniform3ivEXT glad_glProgramUniform3ivEXT
10646 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4IVEXTPROC) (GLuint program, GLint location, GLsizei count,
10647 const GLint *value);
10647 GLAPI PFNGLPROGRAMUNIFORM4IVEXTPROC glad_glProgramUniform4ivEXT;
10648 #define glProgramUniform4ivEXT glad_glProgramUniform4ivEXT
10649 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2FVEXTPROC) (GLuint program, GLint location, GLsizei
10650 count, GLboolean transpose, const GLfloat *value);
10650 GLAPI PFNGLPROGRAMUNIFORMMATRIX2FVEXTPROC glad_glProgramUniformMatrix2fvEXT;
10651 #define glProgramUniformMatrix2fvEXT glad_glProgramUniformMatrix2fvEXT
10652 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3FVEXTPROC) (GLuint program, GLint location, GLsizei
10653 count, GLboolean transpose, const GLfloat *value);
10653 GLAPI PFNGLPROGRAMUNIFORMMATRIX3FVEXTPROC glad_glProgramUniformMatrix3fvEXT;
10654 #define glProgramUniformMatrix3fvEXT glad_glProgramUniformMatrix3fvEXT
10655 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4FVEXTPROC) (GLuint program, GLint location, GLsizei
10656 count, GLboolean transpose, const GLfloat *value);
10656 GLAPI PFNGLPROGRAMUNIFORMMATRIX4FVEXTPROC glad_glProgramUniformMatrix4fvEXT;
10657 #define glProgramUniformMatrix4fvEXT glad_glProgramUniformMatrix4fvEXT
10658 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X3FVEXTPROC) (GLuint program, GLint location, GLsizei
10659 count, GLboolean transpose, const GLfloat *value);
10659 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3FVEXTPROC glad_glProgramUniformMatrix2x3fvEXT;
10660 #define glProgramUniformMatrix2x3fvEXT glad_glProgramUniformMatrix2x3fvEXT
10661 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X2FVEXTPROC) (GLuint program, GLint location, GLsizei
10662 count, GLboolean transpose, const GLfloat *value);
10662 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2FVEXTPROC glad_glProgramUniformMatrix3x2fvEXT;
10663 #define glProgramUniformMatrix3x2fvEXT glad_glProgramUniformMatrix3x2fvEXT
10664 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X4FVEXTPROC) (GLuint program, GLint location, GLsizei
10665 count, GLboolean transpose, const GLfloat *value);
10665 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4FVEXTPROC glad_glProgramUniformMatrix2x4fvEXT;
10666 #define glProgramUniformMatrix2x4fvEXT glad_glProgramUniformMatrix2x4fvEXT
10667 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X2FVEXTPROC) (GLuint program, GLint location, GLsizei
10668 count, GLboolean transpose, const GLfloat *value);
10668 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2FVEXTPROC glad_glProgramUniformMatrix4x2fvEXT;
10669 #define glProgramUniformMatrix4x2fvEXT glad_glProgramUniformMatrix4x2fvEXT
10670 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X4FVEXTPROC) (GLuint program, GLint location, GLsizei
10671 count, GLboolean transpose, const GLfloat *value);
10671 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4FVEXTPROC glad_glProgramUniformMatrix3x4fvEXT;
10672 #define glProgramUniformMatrix3x4fvEXT glad_glProgramUniformMatrix3x4fvEXT
10673 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X3FVEXTPROC) (GLuint program, GLint location, GLsizei
10674 count, GLboolean transpose, const GLfloat *value);
10674 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3FVEXTPROC glad_glProgramUniformMatrix4x3fvEXT;
10675 #define glProgramUniformMatrix4x3fvEXT glad_glProgramUniformMatrix4x3fvEXT
10676 typedef void (APIENTRY PFNGLTEXTUREBUFFEREXTPROC) (GLuint texture, GLenum target, GLenum
10677 internalformat, GLuint buffer);
10677 GLAPI PFNGLTEXTUREBUFFEREXTPROC glad_glTextureBufferEXT;
10678 #define glTextureBufferEXT glad_glTextureBufferEXT
10679 typedef void (APIENTRY PFNGLMULTITEXBUFFEREXTPROC) (GLenum texunit, GLenum target, GLenum
10680 internalformat, GLuint buffer);
10680 GLAPI PFNGLMULTITEXBUFFEREXTPROC glad_glMultiTexBufferEXT;
10681 #define glMultiTexBufferEXT glad_glMultiTexBufferEXT
10682 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIIVEXTPROC) (GLuint texture, GLenum target, GLenum pname,
10683 const GLint *params);
10683 GLAPI PFNGLTEXTUREPARAMETERIIVEXTPROC glad_glTextureParameterIivEXT;
10684 #define glTextureParameterIivEXT glad_glTextureParameterIivEXT
10685 typedef void (APIENTRY PFNGLTEXTUREPARAMETERIUIVEXTPROC) (GLuint texture, GLenum target, GLenum pname,
10686 const GLuint *params);
10686 GLAPI PFNGLTEXTUREPARAMETERIUIVEXTPROC glad_glTextureParameterIuivEXT;
10687 #define glTextureParameterIuivEXT glad_glTextureParameterIuivEXT
10688 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERIIVEXTPROC) (GLuint texture, GLenum target, GLenum
10689 pname, GLint *params);
10689 GLAPI PFNGLGETTEXTUREPARAMETERIIVEXTPROC glad_glGetTextureParameterIivEXT;
10690 #define glGetTextureParameterIivEXT glad_glGetTextureParameterIivEXT
10691 typedef void (APIENTRY PFNGLGETTEXTUREPARAMETERIUIVEXTPROC) (GLuint texture, GLenum target, GLenum
10692 pname, GLuint *params);
10692 GLAPI PFNGLGETTEXTUREPARAMETERIUIVEXTPROC glad_glGetTextureParameterIuivEXT;

```

```

10693 #define glGetTextureParameterIuivEXT glad_glGetTextureParameterIuivEXT
10694 typedef void (APIENTRY PFNGLMULTITEXPARAMETERIIVEXTPROC) (GLenum texunit, GLenum target, GLenum pname,
10695 const GLint *params);
10696 GLAPI PFNGLMULTITEXPARAMETERIIVEXTPROC glad_glMultiTexParameterIivEXT;
10697 #define glGetMultiTexParameterIivEXT glad_glMultiTexParameterIivEXT
10698 typedef void (APIENTRY PFNGLMULTITEXPARAMETERIUIVEXTPROC) (GLenum texunit, GLenum target, GLenum
10699 pname, const GLuint *params);
10700 GLAPI PFNGLMULTITEXPARAMETERIUIVEXTPROC glad_glMultiTexParameterIuivEXT;
10701 #define glGetMultiTexParameterIuivEXT glad_glGetMultiTexParameterIuivEXT
10702 typedef void (APIENTRY PFNGLMULTITEXPARAMETERIUIVEXTPROC) (GLenum texunit, GLenum target, GLenum
10703 pname, GLuint *params);
10704 GLAPI PFNGLMULTITEXPARAMETERIUIVEXTPROC glad_glGetMultiTexParameterIuivEXT;
10705 #define glGetMultiTexParameterIuivEXT glad_glGetMultiTexParameterIuivEXT
10706 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UIEXTPROC) (GLuint program, GLint location, GLuint v0);
10707 GLAPI PFNGLPROGRAMUNIFORM1UIEXTPROC glad_glProgramUniform1uiEXT;
10708 #define glGetProgramUniform1uiEXT glad_glProgramUniform1uiEXT
10709 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UIEXTPROC) (GLuint program, GLint location, GLuint v0,
10710 GLuint v1);
10711 GLAPI PFNGLPROGRAMUNIFORM2UIEXTPROC glad_glProgramUniform2uiEXT;
10712 #define glGetProgramUniform2uiEXT glad_glProgramUniform2uiEXT
10713 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UIEXTPROC) (GLuint program, GLint location, GLuint v0,
10714 GLuint v1, GLuint v2);
10715 GLAPI PFNGLPROGRAMUNIFORM3UIEXTPROC glad_glProgramUniform3uiEXT;
10716 #define glGetProgramUniform3uiEXT glad_glProgramUniform3uiEXT
10717 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UIEXTPROC) (GLuint program, GLint location, GLuint v0,
10718 GLuint v1, GLuint v2, GLuint v3);
10719 GLAPI PFNGLPROGRAMUNIFORM4UIEXTPROC glad_glProgramUniform4uiEXT;
10720 #define glGetProgramUniform4uiEXT glad_glProgramUniform4uiEXT
10721 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1UIVEXTPROC) (GLuint program, GLint location, GLsizei count,
10722 const GLuint *value);
10723 GLAPI PFNGLPROGRAMUNIFORM1UIVEXTPROC glad_glProgramUniform1uivEXT;
10724 #define glGetProgramUniform1uivEXT glad_glProgramUniform1uivEXT
10725 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2UIVEXTPROC) (GLuint program, GLint location, GLsizei count,
10726 const GLuint *value);
10727 GLAPI PFNGLPROGRAMUNIFORM2UIVEXTPROC glad_glProgramUniform2uivEXT;
10728 #define glGetProgramUniform2uivEXT glad_glProgramUniform2uivEXT
10729 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3UIVEXTPROC) (GLuint program, GLint location, GLsizei count,
10730 const GLuint *value);
10731 GLAPI PFNGLPROGRAMUNIFORM3UIVEXTPROC glad_glProgramUniform3uivEXT;
10732 #define glGetProgramUniform3uivEXT glad_glProgramUniform3uivEXT
10733 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4UIVEXTPROC) (GLuint program, GLint location, GLsizei count,
10734 const GLuint *value);
10735 GLAPI PFNGLPROGRAMUNIFORM4UIVEXTPROC glad_glProgramUniform4uivEXT;
10736 #define glGetProgramUniform4uivEXT glad_glProgramUniform4uivEXT
10737 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETERS4FVEXTPROC) (GLuint program, GLenum target,
10738 GLuint index, GLsizei count, const GLfloat *params);
10739 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERS4FVEXTPROC glad_glNamedProgramLocalParameters4fvEXT;
10740 #define glGetNamedProgramLocalParameters4fvEXT glad_glNamedProgramLocalParameters4fvEXT
10741 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETERI4IEXTPROC) (GLuint program, GLenum target,
10742 GLuint index, GLint x, GLint y, GLint z, GLint w);
10743 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERI4IEXTPROC glad_glNamedProgramLocalParameterI4iEXT;
10744 #define glGetNamedProgramLocalParameterI4iEXT glad_glNamedProgramLocalParameterI4iEXT
10745 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETERI4IVEXTPROC) (GLuint program, GLenum target,
10746 GLuint index, const GLint *params);
10747 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERI4IVEXTPROC glad_glNamedProgramLocalParameterI4ivEXT;
10748 #define glGetNamedProgramLocalParameterI4ivEXT glad_glNamedProgramLocalParameterI4ivEXT
10749 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETERSI4IVEXTPROC) (GLuint program, GLenum target,
10750 GLuint index, GLsizei count, const GLuint *params);
10751 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERSI4IVEXTPROC glad_glNamedProgramLocalParametersI4ivEXT;
10752 #define glGetNamedProgramLocalParametersI4ivEXT glad_glNamedProgramLocalParametersI4ivEXT
10753 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETERI4UIEXTPROC) (GLuint program, GLenum target,
10754 GLuint index, GLuint x, GLuint y, GLuint z, GLuint w);
10755 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERI4UIEXTPROC glad_glNamedProgramLocalParameterI4uiEXT;
10756 #define glGetNamedProgramLocalParameterI4uiEXT glad_glNamedProgramLocalParameterI4uiEXT
10757 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETERSI4UIVEXTPROC) (GLuint program, GLenum target,
10758 GLuint index, GLsizei count, const GLuint *params);
10759 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERSI4UIVEXTPROC glad_glNamedProgramLocalParametersI4uivEXT;
10760 #define glGetNamedProgramLocalParametersI4uivEXT glad_glNamedProgramLocalParametersI4uivEXT
10761 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMLOCALPARAMETERIIVEXTPROC) (GLuint program, GLenum target,
10762 GLuint index, GLint *params);
10763 GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERIIVEXTPROC glad_glGetNamedProgramLocalParameterIivEXT;
10764 #define glGetNamedProgramLocalParameterIivEXT glad_glGetNamedProgramLocalParameterIivEXT
10765 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMLOCALPARAMETERIUIVEXTPROC) (GLuint program, GLenum target,
10766 GLuint index, GLuint *params);
10767 GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERIUIVEXTPROC glad_glGetNamedProgramLocalParameterIuivEXT;
10768 #define glGetNamedProgramLocalParameterIuivEXT glad_glGetNamedProgramLocalParameterIuivEXT
10769 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMLOCALPARAMETERSIIVEXTPROC) (GLuint program, GLenum target,
10770 GLuint index, GLint *params);
10771 GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERSIIVEXTPROC glad_glGetNamedProgramLocalParametersIivEXT;
10772 #define glGetNamedProgramLocalParametersIivEXT glad_glGetNamedProgramLocalParametersIivEXT
10773 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMLOCALPARAMETERSIUIVEXTPROC) (GLuint program, GLenum target,
10774 GLuint index, GLuint *params);
10775 GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERSIUIVEXTPROC glad_glGetNamedProgramLocalParametersIuivEXT;
10776 #define glGetNamedProgramLocalParametersIuivEXT glad_glGetNamedProgramLocalParametersIuivEXT
10777 typedef void (APIENTRY PFNGLNABLECLIENTSTATEIEXTPROC) (GLenum array, GLuint index);
10778 GLAPI PFNGLNABLECLIENTSTATEIEXTPROC glad_glEnableClientStateiEXT;
10779 #define glEnableClientStateiEXT glad_glEnableClientStateiEXT

```



```

10760 typedef void (APIENTRY PFNGLDISABLECLIENTSTATEIEXTPROC) (GLenum array, GLuint index);
10761 GLAPI PFNGLDISABLECLIENTSTATEIEXTPROC glad_glDisableClientStateiEXT;
10762 #define glDisableClientStateiEXT glad_glDisableClientStateiEXT
10763 typedef void (APIENTRY PFNGLGETFLOATI_VEXTPROC) (GLenum pname, GLuint index, GLfloat *params);
10764 GLAPI PFNGLGETFLOATI_VEXTPROC glad_glGetFloati_vEXT;
10765 #define glGetFloati_vEXT glad_glGetFloati_vEXT
10766 typedef void (APIENTRY PFNGLGETDOUBLEI_VEXTPROC) (GLenum pname, GLuint index, GLdouble *params);
10767 GLAPI PFNGLGETDOUBLEI_VEXTPROC glad_glGetDoublei_vEXT;
10768 #define glGetDoublei_vEXT glad_glGetDoublei_vEXT
10769 typedef void (APIENTRY PFNGLGETPOINTERI_VEXTPROC) (GLenum pname, GLuint index, void **params);
10770 GLAPI PFNGLGETPOINTERI_VEXTPROC glad_glGetPointeri_vEXT;
10771 #define glGetPointerI_vEXT glad_glGetPointerI_vEXT
10772 typedef void (APIENTRY PFNGLNAMEDPROGRAMSTRINGEXTPROC) (GLuint program, GLenum target, GLenum format,
GLsizei len, const void *string);
10773 GLAPI PFNGLNAMEDPROGRAMSTRINGEXTPROC glad_glNamedProgramStringEXT;
10774 #define glNamedProgramStringEXT glad_glNamedProgramStringEXT
10775 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETER4DEXTPROC) (GLuint program, GLenum target,
GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
10776 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4DEXTPROC glad_glNamedProgramLocalParameter4dEXT;
10777 #define glNamedProgramLocalParameter4dEXT glad_glNamedProgramLocalParameter4dEXT
10778 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETER4DVEXTPROC) (GLuint program, GLenum target,
GLuint index, const GLdouble *params);
10779 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4DVEXTPROC glad_glNamedProgramLocalParameter4dvEXT;
10780 #define glNamedProgramLocalParameter4dvEXT glad_glNamedProgramLocalParameter4dvEXT
10781 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETER4FEXTPROC) (GLuint program, GLenum target,
GLuint index, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
10782 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4FEXTPROC glad_glNamedProgramLocalParameter4fEXT;
10783 #define glNamedProgramLocalParameter4fEXT glad_glNamedProgramLocalParameter4fEXT
10784 typedef void (APIENTRY PFNGLNAMEDPROGRAMLOCALPARAMETER4FVEXTPROC) (GLuint program, GLenum target,
GLuint index, const GLfloat *params);
10785 GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4FVEXTPROC glad_glNamedProgramLocalParameter4fvEXT;
10786 #define glNamedProgramLocalParameter4fvEXT glad_glNamedProgramLocalParameter4fvEXT
10787 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMLOCALPARAMETERDVEXTPROC) (GLuint program, GLenum target,
GLuint index, GLdouble *params);
10788 GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERDVEXTPROC glad_glGetNamedProgramLocalParameterdvEXT;
10789 #define glGetNamedProgramLocalParameterdvEXT glad_glGetNamedProgramLocalParameterdvEXT
10790 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMLOCALPARAMETERFVEXTPROC) (GLuint program, GLenum target,
GLuint index, GLfloat *params);
10791 GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERFVEXTPROC glad_glGetNamedProgramLocalParameterfvEXT;
10792 #define glGetNamedProgramLocalParameterfvEXT glad_glGetNamedProgramLocalParameterfvEXT
10793 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMIVEXTPROC) (GLuint program, GLenum target, GLenum pname,
GLint *params);
10794 GLAPI PFNGLGETNAMEDPROGRAMIVEXTPROC glad_glGetNamedProgramivEXT;
10795 #define glGetNamedProgramivEXT glad_glGetNamedProgramivEXT
10796 typedef void (APIENTRY PFNGLGETNAMEDPROGRAMSTRINGEXTPROC) (GLuint program, GLenum target, GLenum
pname, void *string);
10797 GLAPI PFNGLGETNAMEDPROGRAMSTRINGEXTPROC glad_glGetNamedProgramStringEXT;
10798 #define glGetNamedProgramStringEXT glad_glGetNamedProgramStringEXT
10799 typedef void (APIENTRY PFNGLNAMEDRENDERBUFFERSTORAGEEXTPROC) (GLuint renderbuffer, GLenum
internalformat, GLsizei width, GLsizei height);
10800 GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEEXTPROC glad_glNamedRenderbufferStorageEXT;
10801 #define glNamedRenderbufferStorageEXT glad_glNamedRenderbufferStorageEXT
10802 typedef void (APIENTRY PFNGLGETNAMEDRENDERBUFFERPARAMETERIVEXTPROC) (GLuint renderbuffer, GLenum
pname, GLint *params);
10803 GLAPI PFNGLGETNAMEDRENDERBUFFERPARAMETERIVEXTPROC glad_glGetNamedRenderbufferParameterivEXT;
10804 #define glGetNamedRenderbufferParameterivEXT glad_glGetNamedRenderbufferParameterivEXT
10805 typedef void (APIENTRY PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEEXTPROC) (GLuint renderbuffer, GLsizei
samples, GLenum internalformat, GLsizei width, GLsizei height);
10806 GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEEXTPROC glad_glNamedRenderbufferStorageMultisampleEXT;
10807 #define glNamedRenderbufferStorageMultisampleEXT glad_glNamedRenderbufferStorageMultisampleEXT
10808 typedef void (APIENTRY PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLECOVERAGEEXTPROC) (GLuint renderbuffer,
GLsizei coverageSamples, GLsizei colorSamples, GLenum internalformat, GLsizei width, GLsizei height);
10809 GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLECOVERAGEEXTPROC glad_glNamedRenderbufferStorageMultisampleCoverageEXT;
10810 #define glNamedRenderbufferStorageMultisampleCoverageEXT glad_glNamedRenderbufferStorageMultisampleCoverageEXT
10811 typedef GLenum (APIENTRY PFNGLCHECKNAMEDFRAMEBUFFERSTATUSEXTPROC) (GLuint framebuffer, GLenum target);
10812 GLAPI PFNGLCHECKNAMEDFRAMEBUFFERSTATUSEXTPROC glad_glCheckNamedFramebufferStatusEXT;
10813 #define glCheckNamedFramebufferStatusEXT glad_glCheckNamedFramebufferStatusEXT
10814 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTURE1DEXTPROC) (GLuint framebuffer, GLenum attachment,
GLenum texture, GLuint level);
10815 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURE1DEXTPROC glad_glNamedFramebufferTexture1DEXT;
10816 #define glNamedFramebufferTexture1DEXT glad_glNamedFramebufferTexture1DEXT
10817 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTURE2DEXTPROC) (GLuint framebuffer, GLenum attachment,
GLenum texture, GLuint level);
10818 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURE2DEXTPROC glad_glNamedFramebufferTexture2DEXT;
10819 #define glNamedFramebufferTexture2DEXT glad_glNamedFramebufferTexture2DEXT
10820 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTURE3DEXTPROC) (GLuint framebuffer, GLenum attachment,
GLenum texture, GLuint level, GLint zoffset);
10821 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURE3DEXTPROC glad_glNamedFramebufferTexture3DEXT;
10822 #define glNamedFramebufferTexture3DEXT glad_glNamedFramebufferTexture3DEXT
10823 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERRENDERBUFFEREXTPROC) (GLuint framebuffer, GLenum
attachment, GLenum renderbuffertarget, GLuint renderbuffer);
10824 GLAPI PFNGLNAMEDFRAMEBUFFERRENDERBUFFEREXTPROC glad_glNamedFramebufferRenderbufferEXT;
10825 #define glNamedFramebufferRenderbufferEXT glad_glNamedFramebufferRenderbufferEXT
10826 typedef void (APIENTRY PFNGLGETNAMEDFRAMEBUFFERATTACHMENTPARAMETERIVEXTPROC) (GLuint framebuffer,
GLenum attachment, GLenum pname, GLint *params);

```

```
10827 GLAPI PFNGLGETNAMEDFRAMEBUFFERATTACHMENTPARAMETERIVEXTPROC
glad_glGetNamedFramebufferAttachmentParameterivEXT;
10828 #define glGetNamedFramebufferAttachmentParameterivEXT
glad_glGetNamedFramebufferAttachmentParameterivEXT
10829 typedef void (APIENTRY PFNGLGENERATEMULTITEXMIPMAPEXTPROC)(GLuint texture, GLenum target);
10830 GLAPI PFNGLGENERATEMULTITEXMIPMAPEXTPROC glad_glGenerateTextureMipmapEXT;
10831 #define glGenerateTextureMipmapEXT glad_glGenerateTextureMipmapEXT
10832 typedef void (APIENTRY PFNGLGENERATEMULTITEXMIPMAPEXTPROC)(GLenum texunit, GLenum target);
10833 GLAPI PFNGLGENERATEMULTITEXMIPMAPEXTPROC glad_glGenerateMultiTexMipmapEXT;
10834 #define glGenerateMultiTexMipmapEXT glad_glGenerateMultiTexMipmapEXT
10835 typedef void (APIENTRY PFNGLFRAMEBUFFERDRAWBUFFEREXTPROC)(GLuint framebuffer, GLenum mode);
10836 GLAPI PFNGLFRAMEBUFFERDRAWBUFFEREXTPROC glad_glFramebufferDrawBufferEXT;
10837 #define glFramebufferDrawBufferEXT glad_glFramebufferDrawBufferEXT
10838 typedef void (APIENTRY PFNGLFRAMEBUFFERDRAWBUFFERSEXTPROC)(GLuint framebuffer, GLsizei n, const
GLenum *bufs);
10839 GLAPI PFNGLFRAMEBUFFERDRAWBUFFERSEXTPROC glad_glFramebufferDrawBuffersEXT;
10840 #define glFramebufferDrawBuffersEXT glad_glFramebufferDrawBuffersEXT
10841 typedef void (APIENTRY PFNGLFRAMEBUFFERREADBUFFEREXTPROC)(GLuint framebuffer, GLenum mode);
10842 GLAPI PFNGLFRAMEBUFFERREADBUFFEREXTPROC glad_glFramebufferReadBufferEXT;
10843 #define glFramebufferReadBufferEXT glad_glFramebufferReadBufferEXT
10844 typedef void (APIENTRY PFNGLGETFRAMEBUFFERPARAMETERIVEXTPROC)(GLuint framebuffer, GLenum pname, GLint
*params);
10845 GLAPI PFNGLGETFRAMEBUFFERPARAMETERIVEXTPROC glad_glGetFramebufferParameterivEXT;
10846 #define glGetFramebufferParameterivEXT glad_glGetFramebufferParameterivEXT
10847 typedef void (APIENTRY PFNGLNAMEDCOPYBUFFERSUBDATAEXTPROC)(GLuint readBuffer, GLuint writeBuffer,
GLintptr readOffset, GLintptr writeOffset, GLsizeiptr size);
10848 GLAPI PFNGLNAMEDCOPYBUFFERSUBDATAEXTPROC glad_glNamedCopyBufferSubDataEXT;
10849 #define glNamedCopyBufferSubDataEXT glad_glNamedCopyBufferSubDataEXT
10850 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTUREEXTPROC)(GLuint framebuffer, GLenum attachment,
GLuint texture, GLint level);
10851 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREEXTPROC glad_glNamedFramebufferTextureEXT;
10852 #define glNamedFramebufferTextureEXT glad_glNamedFramebufferTextureEXT
10853 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTURELAYEREXTPROC)(GLuint framebuffer, GLenum
attachment, GLuint texture, GLint level, GLint layer);
10854 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURELAYEREXTPROC glad_glNamedFramebufferTextureLayerEXT;
10855 #define glNamedFramebufferTextureLayerEXT glad_glNamedFramebufferTextureLayerEXT
10856 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTUREFACEEXTPROC)(GLuint framebuffer, GLenum
attachment, GLuint texture, GLint level, GLenum face);
10857 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREFACEEXTPROC glad_glNamedFramebufferTextureFaceEXT;
10858 #define glNamedFramebufferTextureFaceEXT glad_glNamedFramebufferTextureFaceEXT
10859 typedef void (APIENTRY PFNGLTEXTURERENDERBUFFEREXTPROC)(GLuint texture, GLenum target, GLuint
renderbuffer);
10860 GLAPI PFNGLTEXTURERENDERBUFFEREXTPROC glad_glTextureRenderbufferEXT;
10861 #define glTextureRenderbufferEXT glad_glTextureRenderbufferEXT
10862 typedef void (APIENTRY PFNGLMULTITEXRENDERBUFFEREXTPROC)(GLenum texunit, GLenum target, GLuint
renderbuffer);
10863 GLAPI PFNGLMULTITEXRENDERBUFFEREXTPROC glad_glMultiTexRenderbufferEXT;
10864 #define glMultiTexRenderbufferEXT glad_glMultiTexRenderbufferEXT
10865 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLint size,
GLenum type, GLsizei stride, GLintptr offset);
10866 GLAPI PFNGLVERTEXARRAYVERTEXOFFSETEXTPROC glad_glVertexArrayVertexOffsetEXT;
10867 #define glVertexArrayVertexOffsetEXT glad_glVertexArrayVertexOffsetEXT
10868 typedef void (APIENTRY PFNGLVERTEXARRAYCOLOROFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLint size,
GLenum type, GLsizei stride, GLintptr offset);
10869 GLAPI PFNGLVERTEXARRAYCOLOROFFSETEXTPROC glad_glVertexArrayColorOffsetEXT;
10870 #define glVertexArrayColorOffsetEXT glad_glVertexArrayColorOffsetEXT
10871 typedef void (APIENTRY PFNGLVERTEXARRAYEDGEFLAGOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLsizei
stride, GLintptr offset);
10872 GLAPI PFNGLVERTEXARRAYEDGEFLAGOFFSETEXTPROC glad_glVertexArrayEdgeFlagOffsetEXT;
10873 #define glVertexArrayEdgeFlagOffsetEXT glad_glVertexArrayEdgeFlagOffsetEXT
10874 typedef void (APIENTRY PFNGLVERTEXARRAYINDEXOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLenum type,
GLsizei stride, GLintptr offset);
10875 GLAPI PFNGLVERTEXARRAYINDEXOFFSETEXTPROC glad_glVertexArrayIndexOffsetEXT;
10876 #define glVertexArrayIndexOffsetEXT glad_glVertexArrayIndexOffsetEXT
10877 typedef void (APIENTRY PFNGLVERTEXARRAYNORMALOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLenum type,
GLsizei stride, GLintptr offset);
10878 GLAPI PFNGLVERTEXARRAYNORMALOFFSETEXTPROC glad_glVertexArrayNormalOffsetEXT;
10879 #define glVertexArrayNormalOffsetEXT glad_glVertexArrayNormalOffsetEXT
10880 typedef void (APIENTRY PFNGLVERTEXARRAYTEXCOORDOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLint
size, GLenum type, GLsizei stride, GLintptr offset);
10881 GLAPI PFNGLVERTEXARRAYTEXCOORDOFFSETEXTPROC glad_glVertexArrayTexCoordOffsetEXT;
10882 #define glVertexArrayTexCoordOffsetEXT glad_glVertexArrayTexCoordOffsetEXT
10883 typedef void (APIENTRY PFNGLVERTEXARRAYMULTITEXCOORDOFFSETEXTPROC)(GLuint vaobj, GLuint buffer,
GLenum texunit, GLint size, GLenum type, GLsizei stride, GLintptr offset);
10884 GLAPI PFNGLVERTEXARRAYMULTITEXCOORDOFFSETEXTPROC glad_glVertexArrayMultiTexCoordOffsetEXT;
10885 #define glVertexArrayMultiTexCoordOffsetEXT glad_glVertexArrayMultiTexCoordOffsetEXT
10886 typedef void (APIENTRY PFNGLVERTEXARRAYFOGCOORDOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLenum
type, GLsizei stride, GLintptr offset);
10887 GLAPI PFNGLVERTEXARRAYFOGCOORDOFFSETEXTPROC glad_glVertexArrayFogCoordOffsetEXT;
10888 #define glVertexArrayFogCoordOffsetEXT glad_glVertexArrayFogCoordOffsetEXT
10889 typedef void (APIENTRY PFNGLVERTEXARRAYSECONDARYCOLOROFFSETEXTPROC)(GLuint vaobj, GLuint buffer,
GLint size, GLenum type, GLsizei stride, GLintptr offset);
10890 GLAPI PFNGLVERTEXARRAYSECONDARYCOLOROFFSETEXTPROC glad_glVertexArraySecondaryColorOffsetEXT;
10891 #define glVertexArraySecondaryColorOffsetEXT glad_glVertexArraySecondaryColorOffsetEXT
10892 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLuint
index, GLint size, GLenum type, GLboolean normalized, GLsizei stride, GLintptr offset);
10893 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBOFFSETEXTPROC glad_glVertexArrayVertexAttribOffsetEXT;
```

```

10894 #define glVertexArrayVertexAttribOffsetEXT glad_glVertexArrayVertexAttribOffsetEXT
10895 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBOFFSETPROC) (GLuint vaobj, GLuint buffer,
10896   GLuint index, GLint size, GLenum type, GLsizei stride, GLintptr offset);
10897 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBOFFSETPROC glad_glVertexArrayVertexAttribOffsetEXT;
10898 #define glVertexArrayVertexAttribIOffsetEXT glad_glVertexArrayVertexAttribIOffsetEXT
10899 typedef void (APIENTRY PFNGLENABLEVERTEXARRAYEXTPROC) (GLuint vaobj, GLenum array);
10900 GLAPI PFNGLENABLEVERTEXARRAYEXTPROC glad_glEnableVertexArrayEXT;
10901 #define glEnableVertexArrayEXT glad_glEnableVertexArrayEXT
10902 typedef void (APIENTRY PFNGLDISABLEVERTEXARRAYEXTPROC) (GLuint vaobj, GLenum array);
10903 GLAPI PFNGLDISABLEVERTEXARRAYEXTPROC glad_glDisableVertexArrayEXT;
10904 #define glDisableVertexArrayEXT glad_glDisableVertexArrayEXT
10905 typedef void (APIENTRY PFNGLENABLEVERTEXARRAYATTRIBEXTPROC) (GLuint vaobj, GLuint index);
10906 GLAPI PFNGLENABLEVERTEXARRAYATTRIBEXTPROC glad_glEnableVertexArrayAttribEXT;
10907 #define glEnableVertexArrayAttribEXT glad_glEnableVertexArrayAttribEXT
10908 typedef void (APIENTRY PFNGLDISABLEVERTEXARRAYATTRIBEXTPROC) (GLuint vaobj, GLuint index);
10909 GLAPI PFNGLDISABLEVERTEXARRAYATTRIBEXTPROC glad_glDisableVertexArrayAttribEXT;
10910 #define glDisableVertexArrayAttribEXT glad_glDisableVertexArrayAttribEXT
10911 typedef void (APIENTRY PFNGLGETVERTEXARRAYINTEGERVEXTPROC) (GLuint vaobj, GLenum pname, GLint *param);
10912 GLAPI PFNGLGETVERTEXARRAYINTEGERVEXTPROC glad_glGetVertexArrayIntegervEXT;
10913 #define glGetVertexArrayIntegervEXT glad_glGetVertexArrayIntegervEXT
10914 typedef void (APIENTRY PFNGLGETVERTEXARRAYPOINTERVEXTPROC) (GLuint vaobj, GLenum pname, void **param);
10915 GLAPI PFNGLGETVERTEXARRAYPOINTERVEXTPROC glad_glGetVertexArrayPointervEXT;
10916 #define glGetVertexArrayPointervEXT glad_glGetVertexArrayPointervEXT
10917 typedef void (APIENTRY PFNGLGETVERTEXARRAYINTEGERI_VEXTPROC) (GLuint vaobj, GLuint index, GLenum
10918   pname, GLint *param);
10919 GLAPI PFNGLGETVERTEXARRAYINTEGERI_VEXTPROC glad_glGetVertexArrayIntegeri_vEXT;
10920 #define glGetVertexArrayIntegeri_vEXT glad_glGetVertexArrayIntegeri_vEXT
10921 typedef void (APIENTRY PFNGLGETVERTEXARRAYPOINTERI_VEXTPROC) (GLuint vaobj, GLuint index, GLenum
10922   pname, void **param);
10923 GLAPI PFNGLGETVERTEXARRAYPOINTERI_VEXTPROC glad_glGetVertexArrayPointeri_vEXT;
10924 #define glGetVertexArrayPointeri_vEXT glad_glGetVertexArrayPointeri_vEXT
10925 typedef void * (APIENTRY PFNGLMAPNAMEDBUFFERRANGEEXTPROC) (GLuint buffer, GLintptr offset, GLsizeiptr
10926   length, GLbitfield access);
10927 GLAPI PFNGLMAPNAMEDBUFFERRANGEEXTPROC glad_glMapNamedBufferRangeEXT;
10928 #define glMapNamedBufferRangeEXT glad_glMapNamedBufferRangeEXT
10929 typedef void (APIENTRY PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEEXTPROC) (GLuint buffer, GLintptr offset,
10930   GLsizeiptr length);
10931 GLAPI PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEEXTPROC glad_glFlushMappedNamedBufferRangeEXT;
10932 #define glFlushMappedNamedBufferRangeEXT glad_glFlushMappedNamedBufferRangeEXT
10933 typedef void (APIENTRY PFNGLNAMEDBUFFERSTORAGEEXTPROC) (GLuint buffer, GLsizeiptr size, const void
10934   *data, GLbitfield flags);
10935 GLAPI PFNGLNAMEDBUFFERSTORAGEEXTPROC glad_glNamedBufferStorageEXT;
10936 #define glNamedBufferStorageEXT glad_glNamedBufferStorageEXT
10937 typedef void (APIENTRY PFNGLCLEARNAMEDBUFFERDATAEXTPROC) (GLuint buffer, GLenum internalformat, GLenum
10938   format, GLenum type, const void *data);
10939 GLAPI PFNGLCLEARNAMEDBUFFERDATAEXTPROC glad_glClearNamedBufferDataEXT;
10940 #define glClearNamedBufferDataEXT glad_glClearNamedBufferDataEXT
10941 typedef void (APIENTRY PFNGLCLEARNAMEDBUFFERSUBDATAEXTPROC) (GLuint buffer, GLenum internalformat,
10942   GLsizeiptr offset, GLsizeiptr size, GLenum format, GLenum type, const void *data);
10943 GLAPI PFNGLCLEARNAMEDBUFFERSUBDATAEXTPROC glad_glClearNamedBufferSubDataEXT;
10944 #define glClearNamedBufferSubDataEXT glad_glClearNamedBufferSubDataEXT
10945 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERPARAMETERIEXTPROC) (GLuint framebuffer, GLenum pname,
10946   GLint param);
10947 GLAPI PFNGLNAMEDFRAMEBUFFERPARAMETERIEXTPROC glad_glNamedFramebufferParameteriEXT;
10948 #define glNamedFramebufferParameteriEXT glad_glNamedFramebufferParameteriEXT
10949 typedef void (APIENTRY PFNGLGETNAMEDFRAMEBUFFERPARAMETERIVEXTPROC) (GLuint framebuffer, GLenum pname,
10950   GLint *params);
10951 GLAPI PFNGLGETNAMEDFRAMEBUFFERPARAMETERIVEXTPROC glad_glGetNamedFramebufferParameterivEXT;
10952 #define glGetNamedFramebufferParameterivEXT glad_glGetNamedFramebufferParameterivEXT
10953 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1DEXTPROC) (GLuint program, GLint location, GLdouble x);
10954 GLAPI PFNGLPROGRAMUNIFORM1DEXTPROC glad_glProgramUniform1dEXT;
10955 #define glProgramUniform1dEXT glad_glProgramUniform1dEXT
10956 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2DEXTPROC) (GLuint program, GLint location, GLdouble x,
10957   GLdouble y);
10958 GLAPI PFNGLPROGRAMUNIFORM2DEXTPROC glad_glProgramUniform2dEXT;
10959 #define glProgramUniform2dEXT glad_glProgramUniform2dEXT
10960 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3DEXTPROC) (GLuint program, GLint location, GLdouble x,
10961   GLdouble y, GLdouble z);
10962 GLAPI PFNGLPROGRAMUNIFORM3DEXTPROC glad_glProgramUniform3dEXT;
10963 #define glProgramUniform3dEXT glad_glProgramUniform3dEXT
10964 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4DEXTPROC) (GLuint program, GLint location, GLdouble x,
10965   GLdouble y, GLdouble z, GLdouble w);
10966 GLAPI PFNGLPROGRAMUNIFORM4DEXTPROC glad_glProgramUniform4dEXT;
10967 #define glProgramUniform4dEXT glad_glProgramUniform4dEXT
10968 typedef void (APIENTRY PFNGLPROGRAMUNIFORM1DVEXTPROC) (GLuint program, GLint location, GLsizei count,
10969   const GLdouble *value);
10970 GLAPI PFNGLPROGRAMUNIFORM1DVEXTPROC glad_glProgramUniform1dvEXT;
10971 #define glProgramUniform1dvEXT glad_glProgramUniform1dvEXT
10972 typedef void (APIENTRY PFNGLPROGRAMUNIFORM2DVEXTPROC) (GLuint program, GLint location, GLsizei count,
10973   const GLdouble *value);
10974 GLAPI PFNGLPROGRAMUNIFORM2DVEXTPROC glad_glProgramUniform2dvEXT;
10975 #define glProgramUniform2dvEXT glad_glProgramUniform2dvEXT
10976 typedef void (APIENTRY PFNGLPROGRAMUNIFORM3DVEXTPROC) (GLuint program, GLint location, GLsizei count,
10977   const GLdouble *value);
10978 GLAPI PFNGLPROGRAMUNIFORM3DVEXTPROC glad_glProgramUniform3dvEXT;
10979 #define glProgramUniform3dvEXT glad_glProgramUniform3dvEXT
10980 typedef void (APIENTRY PFNGLPROGRAMUNIFORM4DVEXTPROC) (GLuint program, GLint location, GLsizei count,
10981   const GLdouble *value);
10982 GLAPI PFNGLPROGRAMUNIFORM4DVEXTPROC glad_glProgramUniform4dvEXT;
10983 #define glProgramUniform4dvEXT glad_glProgramUniform4dvEXT

```



```

    const GLdouble *value);
10965 GLAPI PFNGLPROGRAMUNIFORM4DVECTPROC glad_glProgramUniform4dvEXT;
10966 #define glProgramUniform4dvEXT glad_glProgramUniform4dvEXT
10967 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10968 GLAPI PFNGLPROGRAMUNIFORMMATRIX2DVECTPROC glad_glProgramUniformMatrix2dvEXT;
10969 #define glProgramUniformMatrix2dvEXT glad_glProgramUniformMatrix2dvEXT
10970 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10971 GLAPI PFNGLPROGRAMUNIFORMMATRIX3DVECTPROC glad_glProgramUniformMatrix3dvEXT;
10972 #define glProgramUniformMatrix3dvEXT glad_glProgramUniformMatrix3dvEXT
10973 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10974 GLAPI PFNGLPROGRAMUNIFORMMATRIX4DVECTPROC glad_glProgramUniformMatrix4dvEXT;
10975 #define glProgramUniformMatrix4dvEXT glad_glProgramUniformMatrix4dvEXT
10976 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X3DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10977 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3DVECTPROC glad_glProgramUniformMatrix2x3dvEXT;
10978 #define glProgramUniformMatrix2x3dvEXT glad_glProgramUniformMatrix2x3dvEXT
10979 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX2X4DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10980 GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4DVECTPROC glad_glProgramUniformMatrix2x4dvEXT;
10981 #define glProgramUniformMatrix2x4dvEXT glad_glProgramUniformMatrix2x4dvEXT
10982 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X2DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10983 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2DVECTPROC glad_glProgramUniformMatrix3x2dvEXT;
10984 #define glProgramUniformMatrix3x2dvEXT glad_glProgramUniformMatrix3x2dvEXT
10985 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX3X4DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10986 GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4DVECTPROC glad_glProgramUniformMatrix3x4dvEXT;
10987 #define glProgramUniformMatrix3x4dvEXT glad_glProgramUniformMatrix3x4dvEXT
10988 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X2DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10989 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2DVECTPROC glad_glProgramUniformMatrix4x2dvEXT;
10990 #define glProgramUniformMatrix4x2dvEXT glad_glProgramUniformMatrix4x2dvEXT
10991 typedef void (APIENTRY PFNGLPROGRAMUNIFORMMATRIX4X3DVECTPROC) (GLuint program, GLint location, GLsizei
count, GLboolean transpose, const GLdouble *value);
10992 GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3DVECTPROC glad_glProgramUniformMatrix4x3dvEXT;
10993 #define glProgramUniformMatrix4x3dvEXT glad_glProgramUniformMatrix4x3dvEXT
10994 typedef void (APIENTRY PFNGLTEXTUREBUFFERRANGEEXTPROC) (GLuint texture, GLenum target, GLenum
internalformat, GLuint buffer, GLintptr offset, GLsizeiptr size);
10995 GLAPI PFNGLTEXTUREBUFFERRANGEEXTPROC glad_glTextureBufferRangeEXT;
10996 #define glTextureBufferRangeEXT glad_glTextureBufferRangeEXT
10997 typedef void (APIENTRY PFNGLTEXTURESTORAGE1DEXTPROC) (GLuint texture, GLenum target, GLsizei levels,
GLenum internalformat, GLsizei width);
10998 GLAPI PFNGLTEXTURESTORAGE1DEXTPROC glad_glTextureStorage1DEXT;
10999 #define glTextureStorage1DEXT glad_glTextureStorage1DEXT
11000 typedef void (APIENTRY PFNGLTEXTURESTORAGE2DEXTPROC) (GLuint texture, GLenum target, GLsizei levels,
GLenum internalformat, GLsizei width, GLsizei height);
11001 GLAPI PFNGLTEXTURESTORAGE2DEXTPROC glad_glTextureStorage2DEXT;
11002 #define glTextureStorage2DEXT glad_glTextureStorage2DEXT
11003 typedef void (APIENTRY PFNGLTEXTURESTORAGE3DEXTPROC) (GLuint texture, GLenum target, GLsizei levels,
GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth);
11004 GLAPI PFNGLTEXTURESTORAGE3DEXTPROC glad_glTextureStorage3DEXT;
11005 #define glTextureStorage3DEXT glad_glTextureStorage3DEXT
11006 typedef void (APIENTRY PFNGLTEXTURESTORAGE2DMULTISAMPLEEXTPROC) (GLuint texture, GLenum target,
GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean
fixedsamplelocations);
11007 GLAPI PFNGLTEXTURESTORAGE2DMULTISAMPLEEXTPROC glad_glTextureStorage2DMultisampleEXT;
11008 #define glTextureStorage2DMultisampleEXT glad_glTextureStorage2DMultisampleEXT
11009 typedef void (APIENTRY PFNGLTEXTURESTORAGE3DMULTISAMPLEEXTPROC) (GLuint texture, GLenum target,
GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean
fixedsamplelocations);
11010 GLAPI PFNGLTEXTURESTORAGE3DMULTISAMPLEEXTPROC glad_glTextureStorage3DMultisampleEXT;
11011 #define glTextureStorage3DMultisampleEXT glad_glTextureStorage3DMultisampleEXT
11012 typedef void (APIENTRY PFNGLVERTEXARRAYBINDVERTEXBUFFEREXTPROC) (GLuint vaobj, GLuint bindingindex,
GLuint buffer, GLintptr offset, GLsizei stride);
11013 GLAPI PFNGLVERTEXARRAYBINDVERTEXBUFFEREXTPROC glad_glVertexArrayBindVertexBufferEXT;
11014 #define glVertexArrayBindVertexBufferEXT glad_glVertexArrayBindVertexBufferEXT
11015 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBFORMATEXTPROC) (GLuint vaobj, GLuint attribindex,
GLint size, GLenum type, GLboolean normalized, GLuint relativeoffset);
11016 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBFORMATEXTPROC glad_glVertexArrayVertexAttribFormatEXT;
11017 #define glVertexArrayVertexAttribFormatEXT glad_glVertexArrayVertexAttribFormatEXT
11018 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBIFORMATEXTPROC) (GLuint vaobj, GLuint attribindex,
GLint size, GLenum type, GLuint relativeoffset);
11019 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBIFORMATEXTPROC glad_glVertexArrayVertexAttribIFormatEXT;
11020 #define glVertexArrayVertexAttribIFormatEXT glad_glVertexArrayVertexAttribIFormatEXT
11021 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBLFORMATEXTPROC) (GLuint vaobj, GLuint attribindex,
GLint size, GLenum type, GLuint relativeoffset);
11022 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBLFORMATEXTPROC glad_glVertexArrayVertexAttribLFormatEXT;
11023 #define glVertexArrayVertexAttribLFormatEXT glad_glVertexArrayVertexAttribLFormatEXT
11024 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBBINDINGEXTPROC) (GLuint vaobj, GLuint attribindex,
GLuint bindingindex);
11025 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBBINDINGEXTPROC glad_glVertexArrayVertexAttribBindingEXT;
11026 #define glVertexArrayVertexAttribBindingEXT glad_glVertexArrayVertexAttribBindingEXT
11027 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXBINDINGDIVISOREXTPROC) (GLuint vaobj, GLuint
bindingindex, GLuint divisor);

```

```

11028 GLAPI PFNGLVERTEXARRAYVERTEXBINDINGDIVISOREXTPROC glad_glVertexArrayVertexBindingDivisorEXT;
11029 #define glVertexArrayVertexBindingDivisorEXT glad_glVertexArrayVertexBindingDivisorEXT
11030 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBLOFFSETEXTPROC) (GLuint vaobj, GLuint buffer,
    GLuint index, GLint size, GLenum type, GLsizei stride, GLintptr offset);
11031 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBLOFFSETEXTPROC glad_glVertexArrayVertexAttribOffsetEXT;
11032 #define glVertexArrayVertexAttribOffsetEXT glad_glVertexArrayVertexAttribOffsetEXT
11033 typedef void (APIENTRY PFNGLTEXTUREPAGECOMMITMENTEXTPROC) (GLuint texture, GLint level, GLint xoffset,
    GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLboolean commit);
11034 GLAPI PFNGLTEXTUREPAGECOMMITMENTEXTPROC glad_glTexturePageCommitmentEXT;
11035 #define glTexturePageCommitmentEXT glad_glTexturePageCommitmentEXT
11036 typedef void (APIENTRY PFNGLVERTEXARRAYVERTEXATTRIBDIVISOREXTPROC) (GLuint vaobj, GLuint index, GLuint
    divisor);
11037 GLAPI PFNGLVERTEXARRAYVERTEXATTRIBDIVISOREXTPROC glad_glVertexArrayVertexAttribDivisorEXT;
11038 #define glVertexArrayVertexAttribDivisorEXT glad_glVertexArrayVertexAttribDivisorEXT
11039 #endif
11040 #ifndef GL_EXT_draw_buffers2
11041 #define GL_EXT_draw_buffers2 1
11042 GLAPI int GLAD_GL_EXT_draw_buffers2;
11043 typedef void (APIENTRY PFNGLCOLORMASKINDEXEDEXTPROC) (GLuint index, GLboolean r, GLboolean g,
    GLboolean b, GLboolean a);
11044 GLAPI PFNGLCOLORMASKINDEXEDEXTPROC glad_glColorMaskIndexedEXT;
11045 #define glColorMaskIndexedEXT glad_glColorMaskIndexedEXT
11046 #endif
11047 #ifndef GL_EXT_draw_instanced
11048 #define GL_EXT_draw_instanced 1
11049 GLAPI int GLAD_GL_EXT_draw_instanced;
11050 typedef void (APIENTRY PFNGLDRAWARRAYSINSTANCEDEXTPROC) (GLenum mode, GLint start, GLsizei count,
    GLsizei primcount);
11051 GLAPI PFNGLDRAWARRAYSINSTANCEDEXTPROC glad_glDrawArraysInstancedEXT;
11052 #define glDrawArraysInstancedEXT glad_glDrawArraysInstancedEXT
11053 typedef void (APIENTRY PFNGLDRAWELEMENTSINSTANCEDEXTPROC) (GLenum mode, GLsizei count, GLenum type,
    const void *indices, GLsizei primcount);
11054 GLAPI PFNGLDRAWELEMENTSINSTANCEDEXTPROC glad_glDrawElementsInstancedEXT;
11055 #define glDrawElementsInstancedEXT glad_glDrawElementsInstancedEXT
11056 #endif
11057 #ifndef GL_EXT_draw_range_elements
11058 #define GL_EXT_draw_range_elements 1
11059 GLAPI int GLAD_GL_EXT_draw_range_elements;
11060 typedef void (APIENTRY PFNGLDRAWRANGEELEMENTSEXTPROC) (GLenum mode, GLuint start, GLuint end, GLsizei
    count, GLenum type, const void *indices);
11061 GLAPI PFNGLDRAWRANGEELEMENTSEXTPROC glad_glDrawRangeElementsEXT;
11062 #define glDrawRangeElementsEXT glad_glDrawRangeElementsEXT
11063 #endif
11064 #ifndef GL_EXT_external_buffer
11065 #define GL_EXT_external_buffer 1
11066 GLAPI int GLAD_GL_EXT_external_buffer;
11067 typedef void (APIENTRY PFNGLBUFFERSTORAGEEXTERNALEXTPROC) (GLenum target, GLintptr offset, GLsizeiptr
    size, GLeglClientBufferEXT clientBuffer, GLbitfield flags);
11068 GLAPI PFNGLBUFFERSTORAGEEXTERNALEXTPROC glad_glBufferStorageExternalEXT;
11069 #define glBufferStorageExternalEXT glad_glBufferStorageExternalEXT
11070 typedef void (APIENTRY PFNGLNAMEDBUFFERSTORAGEEXTERNALEXTPROC) (GLuint buffer, GLintptr offset,
    GLsizeiptr size, GLeglClientBufferEXT clientBuffer, GLbitfield flags);
11071 GLAPI PFNGLNAMEDBUFFERSTORAGEEXTERNALEXTPROC glad_glNamedBufferStorageExternalEXT;
11072 #define glNamedBufferStorageExternalEXT glad_glNamedBufferStorageExternalEXT
11073 #endif
11074 #ifndef GL_EXT_fog_coord
11075 #define GL_EXT_fog_coord 1
11076 GLAPI int GLAD_GL_EXT_fog_coord;
11077 typedef void (APIENTRY PFNGLFOGCOORDFEXTPROC) (GLfloat coord);
11078 GLAPI PFNGLFOGCOORDFEXTPROC glad_glFogCoordfEXT;
11079 #define glFogCoordfEXT glad_glFogCoordfEXT
11080 typedef void (APIENTRY PFNGLFOGCOORDFVEXTPROC) (const GLfloat *coord);
11081 GLAPI PFNGLFOGCOORDFVEXTPROC glad_glFogCoordfvEXT;
11082 #define glFogCoordfvEXT glad_glFogCoordfvEXT
11083 typedef void (APIENTRY PFNGLFOGCOORDDEXTPROC) (GLdouble coord);
11084 GLAPI PFNGLFOGCOORDDEXTPROC glad_glFogCoorddEXT;
11085 #define glFogCoorddEXT glad_glFogCoorddEXT
11086 typedef void (APIENTRY PFNGLFOGCOORDDVEXTPROC) (const GLdouble *coord);
11087 GLAPI PFNGLFOGCOORDDVEXTPROC glad_glFogCoorddvEXT;
11088 #define glFogCoorddvEXT glad_glFogCoorddvEXT
11089 typedef void (APIENTRY PFNGLFOGCOORDPOINTEREXTPROC) (GLenum type, GLsizei stride, const void
    *pointer);
11090 GLAPI PFNGLFOGCOORDPOINTEREXTPROC glad_glFogCoordPointerEXT;
11091 #define glFogCoordPointerEXT glad_glFogCoordPointerEXT
11092 #endif
11093 #ifndef GL_EXT_framebuffer_blit
11094 #define GL_EXT_framebuffer_blit 1
11095 GLAPI int GLAD_GL_EXT_framebuffer_blit;
11096 typedef void (APIENTRY PFNGLBLITFRAMEBUFFEREXTPROC) (GLint srcX0, GLint srcY0, GLint srcX1, GLint
    srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLbitfield mask, GLenum filter);
11097 GLAPI PFNGLBLITFRAMEBUFFEREXTPROC glad_glBlitFramebufferEXT;
11098 #define glBlitFramebufferEXT glad_glBlitFramebufferEXT
11099 #endif
11100 #ifndef GL_EXT_framebuffer_blit_layers
11101 #define GL_EXT_framebuffer_blit_layers 1
11102 GLAPI int GLAD_GL_EXT_framebuffer_blit_layers;
11103 typedef void (APIENTRY PFNGLBLITFRAMEBUFFERLAYERSEXTPROC) (GLint srcX0, GLint srcY0, GLint srcX1,

```

```

GLint srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLbitfield mask, GLenum filter);
11104 GLAPI PFNGLBLITFRAMEBUFFERLAYERSEXTPROC glad_glBlitFramebufferLayersEXT;
11105 #define glBlitFramebufferLayersEXT glad_glBlitFramebufferLayersEXT
11106 typedef void (APIENTRY PFNGLBLITFRAMEBUFFERLAYERSEXTPROC) (GLint srcX0, GLint srcY0, GLint srcX1, GLint
srcY1, GLint srcLayer, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLint dstLayer, GLbitfield
mask, GLenum filter);
11107 GLAPI PFNGLBLITFRAMEBUFFERLAYERSEXTPROC glad_glBlitFramebufferLayerEXT;
11108 #define glBlitFramebufferLayerEXT glad_glBlitFramebufferLayerEXT
11109 #endif
11110 #ifndef GL_EXT_framebuffer_multisample
11111 #define GL_EXT_framebuffer_multisample 1
11112 GLAPI int GLAD_GL_EXT_framebuffer_multisample;
11113 typedef void (APIENTRY PFNGLRENDERBUFFERSTORAGEMULTISAMPLESEXTPROC) (GLenum target, GLsizei samples,
GLenum internalformat, GLsizei width, GLsizei height);
11114 GLAPI PFNGLRENDERBUFFERSTORAGEMULTISAMPLESEXTPROC glad_glRenderbufferStorageMultisampleEXT;
11115 #define glRenderbufferStorageMultisampleEXT glad_glRenderbufferStorageMultisampleEXT
11116 #endif
11117 #ifndef GL_EXT_framebuffer_multisample_blit_scaled
11118 #define GL_EXT_framebuffer_multisample_blit_scaled 1
11119 GLAPI int GLAD_GL_EXT_framebuffer_multisample_blit_scaled;
11120 #endif
11121 #ifndef GL_EXT_framebuffer_object
11122 #define GL_EXT_framebuffer_object 1
11123 GLAPI int GLAD_GL_EXT_framebuffer_object;
11124 typedef GLboolean (APIENTRY PFNGLISRENDERBUFFERSEXTPROC) (GLuint renderbuffer);
11125 GLAPI PFNGLISRENDERBUFFERSEXTPROC glad_glIsRenderbufferEXT;
11126 #define glIsRenderbufferEXT glad_glIsRenderbufferEXT
11127 typedef void (APIENTRY PFNGLBINDRENDERBUFFERSEXTPROC) (GLenum target, GLuint renderbuffer);
11128 GLAPI PFNGLBINDRENDERBUFFERSEXTPROC glad_glBindRenderbufferEXT;
11129 #define glBindRenderbufferEXT glad_glBindRenderbufferEXT
11130 typedef void (APIENTRY PFNGLDELETERENDERBUFFERSEXTPROC) (GLsizei n, const GLuint *renderbuffers);
11131 GLAPI PFNGLDELETERENDERBUFFERSEXTPROC glad_glDeleteRenderbuffersEXT;
11132 #define glDeleteRenderbuffersEXT glad_glDeleteRenderbuffersEXT
11133 typedef void (APIENTRY PFNGLGENRENDERBUFFERSEXTPROC) (GLsizei n, GLuint *renderbuffers);
11134 GLAPI PFNGLGENRENDERBUFFERSEXTPROC glad_glGenRenderbuffersEXT;
11135 #define glGenRenderbuffersEXT glad_glGenRenderbuffersEXT
11136 typedef void (APIENTRY PFNGLRENDERBUFFERSTORAGESEXTPROC) (GLenum target, GLenum internalformat, GLsizei
width, GLsizei height);
11137 GLAPI PFNGLRENDERBUFFERSTORAGESEXTPROC glad_glRenderbufferStorageEXT;
11138 #define glRenderbufferStorageEXT glad_glRenderbufferStorageEXT
11139 typedef void (APIENTRY PFNGLGETRENDERBUFFERPARAMETERIVSEXTPROC) (GLenum target, GLenum pname, GLint
*params);
11140 GLAPI PFNGLGETRENDERBUFFERPARAMETERIVSEXTPROC glad_glGetRenderbufferParameterivEXT;
11141 #define glGetRenderbufferParameterivEXT glad_glGetRenderbufferParameterivEXT
11142 typedef GLboolean (APIENTRY PFNGLISFRAMEBUFFERSEXTPROC) (GLuint framebuffer);
11143 GLAPI PFNGLISFRAMEBUFFERSEXTPROC glad_glIsFramebufferEXT;
11144 #define glIsFramebufferEXT glad_glIsFramebufferEXT
11145 typedef void (APIENTRY PFNGLBINDFRAMEBUFFERSEXTPROC) (GLenum target, GLuint framebuffer);
11146 GLAPI PFNGLBINDFRAMEBUFFERSEXTPROC glad_glBindFramebufferEXT;
11147 #define glBindFramebufferEXT glad_glBindFramebufferEXT
11148 typedef void (APIENTRY PFNGLDELETEFRAMEBUFFERSEXTPROC) (GLsizei n, const GLuint *framebuffers);
11149 GLAPI PFNGLDELETEFRAMEBUFFERSEXTPROC glad_glDeleteFramebuffersEXT;
11150 #define glDeleteFramebuffersEXT glad_glDeleteFramebuffersEXT
11151 typedef void (APIENTRY PFNGLGENFRAMEBUFFERSEXTPROC) (GLsizei n, GLuint *framebuffers);
11152 GLAPI PFNGLGENFRAMEBUFFERSEXTPROC glad_glGenFramebuffersEXT;
11153 #define glGenFramebuffersEXT glad_glGenFramebuffersEXT
11154 typedef GLenum (APIENTRY PFNGLCHECKFRAMEBUFFERSTATUSSEXTPROC) (GLenum target);
11155 GLAPI PFNGLCHECKFRAMEBUFFERSTATUSSEXTPROC glad_glCheckFramebufferStatusEXT;
11156 #define glCheckFramebufferStatusEXT glad_glCheckFramebufferStatusEXT
11157 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURE1DSEXTPROC) (GLenum target, GLenum attachment, GLenum
textarget, GLuint texture, GLint level);
11158 GLAPI PFNGLFRAMEBUFFERTEXTURE1DSEXTPROC glad_glFramebufferTexture1DEXT;
11159 #define glFramebufferTexture1DEXT glad_glFramebufferTexture1DEXT
11160 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURE2DSEXTPROC) (GLenum target, GLenum attachment, GLenum
textarget, GLuint texture, GLint level);
11161 GLAPI PFNGLFRAMEBUFFERTEXTURE2DSEXTPROC glad_glFramebufferTexture2DEXT;
11162 #define glFramebufferTexture2DEXT glad_glFramebufferTexture2DEXT
11163 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURE3DSEXTPROC) (GLenum target, GLenum attachment, GLenum
textarget, GLuint texture, GLint level, GLint zoffset);
11164 GLAPI PFNGLFRAMEBUFFERTEXTURE3DSEXTPROC glad_glFramebufferTexture3DEXT;
11165 #define glFramebufferTexture3DEXT glad_glFramebufferTexture3DEXT
11166 typedef void (APIENTRY PFNGLFRAMEBUFFERRENDERBUFFERSEXTPROC) (GLenum target, GLenum attachment, GLenum
renderbuffertarget, GLuint renderbuffer);
11167 GLAPI PFNGLFRAMEBUFFERRENDERBUFFERSEXTPROC glad_glFramebufferRenderbufferEXT;
11168 #define glFramebufferRenderbufferEXT glad_glFramebufferRenderbufferEXT
11169 typedef void (APIENTRY PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVSEXTPROC) (GLenum target, GLenum
attachment, GLenum pname, GLint *params);
11170 GLAPI PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVSEXTPROC glad_glGetFramebufferAttachmentParameterivEXT;
11171 #define glGetFramebufferAttachmentParameterivEXT glad_glGetFramebufferAttachmentParameterivEXT
11172 typedef void (APIENTRY PFNGLGENERATEMIPMAPEXTPROC) (GLenum target);
11173 GLAPI PFNGLGENERATEMIPMAPEXTPROC glad_glGenerateMipmapEXT;
11174 #define glGenerateMipmapEXT glad_glGenerateMipmapEXT
11175 #endif
11176 #ifndef GL_EXT_framebuffer_sRGB
11177 #define GL_EXT_framebuffer_sRGB 1
11178 GLAPI int GLAD_GL_EXT_framebuffer_sRGB;
11179 #endif

```

```

11180 #ifndef GL_EXT_geometry_shader4
11181 #define GL_EXT_geometry_shader4 1
11182 GLAPI int GLAD_GL_EXT_geometry_shader4;
11183 typedef void (APIENTRY PFNGLPROGRAMPARAMETERIEXTPROC) (GLuint program, GLenum pname, GLint value);
11184 GLAPI PFNGLPROGRAMPARAMETERIEXTPROC glad_glProgramParameteriEXT;
11185 #define glProgramParameteriEXT glad_glProgramParameteriEXT
11186 #endif
11187 #ifndef GL_EXT_gpu_program_parameters
11188 #define GL_EXT_gpu_program_parameters 1
11189 GLAPI int GLAD_GL_EXT_gpu_program_parameters;
11190 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETERS4FVEXTPROC) (GLenum target, GLuint index, GLsizei
count, const GLfloat *params);
11191 GLAPI PFNGLPROGRAMENVPARAMETERS4FVEXTPROC glad_glProgramEnvParameters4fvEXT;
11192 #define glProgramEnvParameters4fvEXT glad_glProgramEnvParameters4fvEXT
11193 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETERS4FVEXTPROC) (GLenum target, GLuint index, GLsizei
count, const GLfloat *params);
11194 GLAPI PFNGLPROGRAMLOCALPARAMETERS4FVEXTPROC glad_glProgramLocalParameters4fvEXT;
11195 #define glProgramLocalParameters4fvEXT glad_glProgramLocalParameters4fvEXT
11196 #endif
11197 #ifndef GL_EXT_gpu_shader4
11198 #define GL_EXT_gpu_shader4 1
11199 GLAPI int GLAD_GL_EXT_gpu_shader4;
11200 typedef void (APIENTRY PFNGLGETUNIFORMUIVEXTPROC) (GLuint program, GLint location, GLuint *params);
11201 GLAPI PFNGLGETUNIFORMUIVEXTPROC glad_glGetUniformivEXT;
11202 #define glGetUniformivEXT glad_glGetUniformivEXT
11203 typedef void (APIENTRY PFNGLBINDFRAGDATALOCATIONEXTPROC) (GLuint program, GLuint color, const GLchar
*name);
11204 GLAPI PFNGLBINDFRAGDATALOCATIONEXTPROC glad_glBindFragDataLocationEXT;
11205 #define glBindFragDataLocationEXT glad_glBindFragDataLocationEXT
11206 typedef GLint (APIENTRY PFNGLGETFRAGDATALOCATIONEXTPROC) (GLuint program, const GLchar *name);
11207 GLAPI PFNGLGETFRAGDATALOCATIONEXTPROC glad_glGetFragDataLocationEXT;
11208 #define glGetFragDataLocationEXT glad_glGetFragDataLocationEXT
11209 typedef void (APIENTRY PFNGLUNIFORM1UIEXTPROC) (GLint location, GLuint v0);
11210 GLAPI PFNGLUNIFORM1UIEXTPROC glad_glUniform1uiEXT;
11211 #define glUniform1uiEXT glad_glUniform1uiEXT
11212 typedef void (APIENTRY PFNGLUNIFORM2UIEXTPROC) (GLint location, GLuint v0, GLuint v1);
11213 GLAPI PFNGLUNIFORM2UIEXTPROC glad_glUniform2uiEXT;
11214 #define glUniform2uiEXT glad_glUniform2uiEXT
11215 typedef void (APIENTRY PFNGLUNIFORM3UIEXTPROC) (GLint location, GLuint v0, GLuint v1, GLuint v2);
11216 GLAPI PFNGLUNIFORM3UIEXTPROC glad_glUniform3uiEXT;
11217 #define glUniform3uiEXT glad_glUniform3uiEXT
11218 typedef void (APIENTRY PFNGLUNIFORM4UIEXTPROC) (GLint location, GLuint v0, GLuint v1, GLuint v2,
GLuint v3);
11219 GLAPI PFNGLUNIFORM4UIEXTPROC glad_glUniform4uiEXT;
11220 #define glUniform4uiEXT glad_glUniform4uiEXT
11221 typedef void (APIENTRY PFNGLUNIFORM1UIVEXTPROC) (GLint location, GLsizei count, const GLuint *value);
11222 GLAPI PFNGLUNIFORM1UIVEXTPROC glad_glUniform1uivEXT;
11223 #define glUniform1uivEXT glad_glUniform1uivEXT
11224 typedef void (APIENTRY PFNGLUNIFORM2UIVEXTPROC) (GLint location, GLsizei count, const GLuint *value);
11225 GLAPI PFNGLUNIFORM2UIVEXTPROC glad_glUniform2uivEXT;
11226 #define glUniform2uivEXT glad_glUniform2uivEXT
11227 typedef void (APIENTRY PFNGLUNIFORM3UIVEXTPROC) (GLint location, GLsizei count, const GLuint *value);
11228 GLAPI PFNGLUNIFORM3UIVEXTPROC glad_glUniform3uivEXT;
11229 #define glUniform3uivEXT glad_glUniform3uivEXT
11230 typedef void (APIENTRY PFNGLUNIFORM4UIVEXTPROC) (GLint location, GLsizei count, const GLuint *value);
11231 GLAPI PFNGLUNIFORM4UIVEXTPROC glad_glUniform4uivEXT;
11232 #define glUniform4uivEXT glad_glUniform4uivEXT
11233 typedef void (APIENTRY PFNGLVERTEXATTRIBI1EXTPROC) (GLuint index, GLint x);
11234 GLAPI PFNGLVERTEXATTRIBI1EXTPROC glad_glVertexAttribI1iEXT;
11235 #define glVertexAttribI1iEXT glad_glVertexAttribI1iEXT
11236 typedef void (APIENTRY PFNGLVERTEXATTRIBI2EXTPROC) (GLuint index, GLint x, GLint y);
11237 GLAPI PFNGLVERTEXATTRIBI2EXTPROC glad_glVertexAttribI2iEXT;
11238 #define glVertexAttribI2iEXT glad_glVertexAttribI2iEXT
11239 typedef void (APIENTRY PFNGLVERTEXATTRIBI3EXTPROC) (GLuint index, GLint x, GLint y, GLint z);
11240 GLAPI PFNGLVERTEXATTRIBI3EXTPROC glad_glVertexAttribI3iEXT;
11241 #define glVertexAttribI3iEXT glad_glVertexAttribI3iEXT
11242 typedef void (APIENTRY PFNGLVERTEXATTRIBI4EXTPROC) (GLuint index, GLint x, GLint y, GLint z, GLint
w);
11243 GLAPI PFNGLVERTEXATTRIBI4EXTPROC glad_glVertexAttribI4iEXT;
11244 #define glVertexAttribI4iEXT glad_glVertexAttribI4iEXT
11245 typedef void (APIENTRY PFNGLVERTEXATTRIBI1UIEXTPROC) (GLuint index, GLuint x);
11246 GLAPI PFNGLVERTEXATTRIBI1UIEXTPROC glad_glVertexAttribI1uiEXT;
11247 #define glVertexAttribI1uiEXT glad_glVertexAttribI1uiEXT
11248 typedef void (APIENTRY PFNGLVERTEXATTRIBI2UIEXTPROC) (GLuint index, GLuint x, GLuint y);
11249 GLAPI PFNGLVERTEXATTRIBI2UIEXTPROC glad_glVertexAttribI2uiEXT;
11250 #define glVertexAttribI2uiEXT glad_glVertexAttribI2uiEXT
11251 typedef void (APIENTRY PFNGLVERTEXATTRIBI3UIEXTPROC) (GLuint index, GLuint x, GLuint y, GLuint z);
11252 GLAPI PFNGLVERTEXATTRIBI3UIEXTPROC glad_glVertexAttribI3uiEXT;
11253 #define glVertexAttribI3uiEXT glad_glVertexAttribI3uiEXT
11254 typedef void (APIENTRY PFNGLVERTEXATTRIBI4UIEXTPROC) (GLuint index, GLuint x, GLuint y, GLuint z,
GLuint w);
11255 GLAPI PFNGLVERTEXATTRIBI4UIEXTPROC glad_glVertexAttribI4uiEXT;
11256 #define glVertexAttribI4uiEXT glad_glVertexAttribI4uiEXT
11257 typedef void (APIENTRY PFNGLVERTEXATTRIBI1VEXTPROC) (GLuint index, const GLint *v);
11258 GLAPI PFNGLVERTEXATTRIBI1VEXTPROC glad_glVertexAttribI1ivEXT;
11259 #define glVertexAttribI1ivEXT glad_glVertexAttribI1ivEXT
11260 typedef void (APIENTRY PFNGLVERTEXATTRIBI2VEXTPROC) (GLuint index, const GLint *v);

```

```

11261 GLAPI PFNGLVERTEXATTRIBI2IVEXTPROC glad_glVertexAttribI2ivEXT;
11262 #define glVertexAttribI2ivEXT glad_glVertexAttribI2ivEXT
11263 typedef void (APIENTRY PFNGLVERTEXATTRIBI3IVEXTPROC) (GLuint index, const GLint *v);
11264 GLAPI PFNGLVERTEXATTRIBI3IVEXTPROC glad_glVertexAttribI3ivEXT;
11265 #define glVertexAttribI3ivEXT glad_glVertexAttribI3ivEXT
11266 typedef void (APIENTRY PFNGLVERTEXATTRIBI4IVEXTPROC) (GLuint index, const GLint *v);
11267 GLAPI PFNGLVERTEXATTRIBI4IVEXTPROC glad_glVertexAttribI4ivEXT;
11268 #define glVertexAttribI4ivEXT glad_glVertexAttribI4ivEXT
11269 typedef void (APIENTRY PFNGLVERTEXATTRIBI1UIVEXTPROC) (GLuint index, const GLuint *v);
11270 GLAPI PFNGLVERTEXATTRIBI1UIVEXTPROC glad_glVertexAttribI1uivEXT;
11271 #define glVertexAttribI1uivEXT glad_glVertexAttribI1uivEXT
11272 typedef void (APIENTRY PFNGLVERTEXATTRIBI2UIVEXTPROC) (GLuint index, const GLuint *v);
11273 GLAPI PFNGLVERTEXATTRIBI2UIVEXTPROC glad_glVertexAttribI2uivEXT;
11274 #define glVertexAttribI2uivEXT glad_glVertexAttribI2uivEXT
11275 typedef void (APIENTRY PFNGLVERTEXATTRIBI3UIVEXTPROC) (GLuint index, const GLuint *v);
11276 GLAPI PFNGLVERTEXATTRIBI3UIVEXTPROC glad_glVertexAttribI3uivEXT;
11277 #define glVertexAttribI3uivEXT glad_glVertexAttribI3uivEXT
11278 typedef void (APIENTRY PFNGLVERTEXATTRIBI4UIVEXTPROC) (GLuint index, const GLuint *v);
11279 GLAPI PFNGLVERTEXATTRIBI4UIVEXTPROC glad_glVertexAttribI4uivEXT;
11280 #define glVertexAttribI4uivEXT glad_glVertexAttribI4uivEXT
11281 typedef void (APIENTRY PFNGLVERTEXATTRIBI4BVEXTPROC) (GLuint index, const GLbyte *v);
11282 GLAPI PFNGLVERTEXATTRIBI4BVEXTPROC glad_glVertexAttribI4bvEXT;
11283 #define glVertexAttribI4bvEXT glad_glVertexAttribI4bvEXT
11284 typedef void (APIENTRY PFNGLVERTEXATTRIBI4SVEXTPROC) (GLuint index, const GLshort *v);
11285 GLAPI PFNGLVERTEXATTRIBI4SVEXTPROC glad_glVertexAttribI4svEXT;
11286 #define glVertexAttribI4svEXT glad_glVertexAttribI4svEXT
11287 typedef void (APIENTRY PFNGLVERTEXATTRIBI4UBVEXTPROC) (GLuint index, const GLubyte *v);
11288 GLAPI PFNGLVERTEXATTRIBI4UBVEXTPROC glad_glVertexAttribI4ubvEXT;
11289 #define glVertexAttribI4ubvEXT glad_glVertexAttribI4ubvEXT
11290 typedef void (APIENTRY PFNGLVERTEXATTRIBI4USVEXTPROC) (GLuint index, const GLushort *v);
11291 GLAPI PFNGLVERTEXATTRIBI4USVEXTPROC glad_glVertexAttribI4usvEXT;
11292 #define glVertexAttribI4usvEXT glad_glVertexAttribI4usvEXT
11293 typedef void (APIENTRY PFNGLVERTEXATTRIBIPOINTEREXTPROC) (GLuint index, GLint size, GLenum type,
GLsizei stride, const void *pointer);
11294 GLAPI PFNGLVERTEXATTRIBIPOINTEREXTPROC glad_glVertexAttribIPointerEXT;
11295 #define glVertexAttribIPointerEXT glad_glVertexAttribIPointerEXT
11296 typedef void (APIENTRY PFNGLGETVERTEXATTRIBIIVEXTPROC) (GLuint index, GLenum pname, GLint *params);
11297 GLAPI PFNGLGETVERTEXATTRIBIIVEXTPROC glad_glGetVertexAttribIivEXT;
11298 #define glGetVertexAttribIivEXT glad_glGetVertexAttribIivEXT
11299 typedef void (APIENTRY PFNGLGETVERTEXATTRIBIUIVEXTPROC) (GLuint index, GLenum pname, GLuint *params);
11300 GLAPI PFNGLGETVERTEXATTRIBIUIVEXTPROC glad_glGetVertexAttribIuivEXT;
11301 #define glGetVertexAttribIuivEXT glad_glGetVertexAttribIuivEXT
11302 #endif
11303 #ifndef GL_EXT_histogram
11304 #define GL_EXT_histogram 1
11305 GLAPI int GLAD_GL_EXT_histogram;
11306 typedef void (APIENTRY PFNGLGETHISTOGRAMEXTPROC) (GLenum target, GLboolean reset, GLenum format,
GLenum type, void *values);
11307 GLAPI PFNGLGETHISTOGRAMEXTPROC glad_glGetHistogramEXT;
11308 #define glGetHistogramEXT glad_glGetHistogramEXT
11309 typedef void (APIENTRY PFNGLGETHISTOGRAMPARAMETERFVEXTPROC) (GLenum target, GLenum pname, GLfloat
*params);
11310 GLAPI PFNGLGETHISTOGRAMPARAMETERFVEXTPROC glad_glGetHistogramParameterfvEXT;
11311 #define glGetHistogramParameterfvEXT glad_glGetHistogramParameterfvEXT
11312 typedef void (APIENTRY PFNGLGETHISTOGRAMPARAMETERIVEXTPROC) (GLenum target, GLenum pname, GLint
*params);
11313 GLAPI PFNGLGETHISTOGRAMPARAMETERIVEXTPROC glad_glGetHistogramParameterivEXT;
11314 #define glGetHistogramParameterivEXT glad_glGetHistogramParameterivEXT
11315 typedef void (APIENTRY PFNGLGETMINMAXEXTPROC) (GLenum target, GLboolean reset, GLenum format, GLenum
type, void *values);
11316 GLAPI PFNGLGETMINMAXEXTPROC glad_glGetMinmaxEXT;
11317 #define glGetMinmaxEXT glad_glGetMinmaxEXT
11318 typedef void (APIENTRY PFNGLGETMINMAXPARAMETERFVEXTPROC) (GLenum target, GLenum pname, GLfloat
*params);
11319 GLAPI PFNGLGETMINMAXPARAMETERFVEXTPROC glad_glGetMinmaxParameterfvEXT;
11320 #define glGetMinmaxParameterfvEXT glad_glGetMinmaxParameterfvEXT
11321 typedef void (APIENTRY PFNGLGETMINMAXPARAMETERIVEXTPROC) (GLenum target, GLenum pname, GLint *params);
11322 GLAPI PFNGLGETMINMAXPARAMETERIVEXTPROC glad_glGetMinmaxParameterivEXT;
11323 #define glGetMinmaxParameterivEXT glad_glGetMinmaxParameterivEXT
11324 typedef void (APIENTRY PFNGLHISTOGRAMEXTPROC) (GLenum target, GLsizei width, GLenum internalformat,
GLboolean sink);
11325 GLAPI PFNGLHISTOGRAMEXTPROC glad_glHistogramEXT;
11326 #define glGetHistogramEXT glad_glHistogramEXT
11327 typedef void (APIENTRY PFNGLMINMAXEXTPROC) (GLenum target, GLenum internalformat, GLboolean sink);
11328 GLAPI PFNGLMINMAXEXTPROC glad_glMinmaxEXT;
11329 #define glGetMinmaxEXT glad_glMinmaxEXT
11330 typedef void (APIENTRY PFNGLRESETHISTOGRAMEXTPROC) (GLenum target);
11331 GLAPI PFNGLRESETHISTOGRAMEXTPROC glad_glResetHistogramEXT;
11332 #define glGetResetHistogramEXT glad_glResetHistogramEXT
11333 typedef void (APIENTRY PFNGLRESETMINMAXEXTPROC) (GLenum target);
11334 GLAPI PFNGLRESETMINMAXEXTPROC glad_glResetMinmaxEXT;
11335 #define glGetResetMinmaxEXT glad_glResetMinmaxEXT
11336 #endif
11337 #ifndef GL_EXT_index_array_formats
11338 #define GL_EXT_index_array_formats 1
11339 GLAPI int GLAD_GL_EXT_index_array_formats;
11340 #endif

```



```

11341 #ifndef GL_EXT_index_func
11342 #define GL_EXT_index_func 1
11343 GLAPI int GLAD_GL_EXT_index_func;
11344 typedef void (APIENTRY PFNGLINDEXFUNCEXTPROC) (GLenum func, GLclampf ref);
11345 GLAPI PFNGLINDEXFUNCEXTPROC glad_glIndexFuncEXT;
11346 #define glIndexFuncEXT glad_glIndexFuncEXT
11347 #endif
11348 #ifndef GL_EXT_index_material
11349 #define GL_EXT_index_material 1
11350 GLAPI int GLAD_GL_EXT_index_material;
11351 typedef void (APIENTRY PFNGLINDEXMATERIALEXTPROC) (GLenum face, GLenum mode);
11352 GLAPI PFNGLINDEXMATERIALEXTPROC glad_glIndexMaterialEXT;
11353 #define glIndexMaterialEXT glad_glIndexMaterialEXT
11354 #endif
11355 #ifndef GL_EXT_index_texture
11356 #define GL_EXT_index_texture 1
11357 GLAPI int GLAD_GL_EXT_index_texture;
11358 #endif
11359 #ifndef GL_EXT_light_texture
11360 #define GL_EXT_light_texture 1
11361 GLAPI int GLAD_GL_EXT_light_texture;
11362 typedef void (APIENTRY PFNGLAPPLYTEXTUREEXTPROC) (GLenum mode);
11363 GLAPI PFNGLAPPLYTEXTUREEXTPROC glad_glApplyTextureEXT;
11364 #define glApplyTextureEXT glad_glApplyTextureEXT
11365 typedef void (APIENTRY PFNGLTEXTURELIGHTEXTPROC) (GLenum pname);
11366 GLAPI PFNGLTEXTURELIGHTEXTPROC glad_glTextureLightEXT;
11367 #define glTextureLightEXT glad_glTextureLightEXT
11368 typedef void (APIENTRY PFNGLTEXTUREMATERIALEXTPROC) (GLenum face, GLenum mode);
11369 GLAPI PFNGLTEXTUREMATERIALEXTPROC glad_glTextureMaterialEXT;
11370 #define glTextureMaterialEXT glad_glTextureMaterialEXT
11371 #endif
11372 #ifndef GL_EXT_memory_object
11373 #define GL_EXT_memory_object 1
11374 GLAPI int GLAD_GL_EXT_memory_object;
11375 typedef void (APIENTRY PFNGLGETUNSIGNEDBYTEVEXTPROC) (GLenum pname, GLubyte *data);
11376 GLAPI PFNGLGETUNSIGNEDBYTEVEXTPROC glad_glGetUnsignedBytevEXT;
11377 #define glGetUnsignedBytevEXT glad_glGetUnsignedBytevEXT
11378 typedef void (APIENTRY PFNGLGETUNSIGNEDBYTEI_VEXTPROC) (GLenum target, GLuint index, GLubyte *data);
11379 GLAPI PFNGLGETUNSIGNEDBYTEI_VEXTPROC glad_glGetUnsignedBytei_vEXT;
11380 #define glGetUnsignedBytei_vEXT glad_glGetUnsignedBytei_vEXT
11381 typedef void (APIENTRY PFNGLDELETEMEMORYOBJECTSEXTPROC) (GLsizei n, const GLuint *memoryObjects);
11382 GLAPI PFNGLDELETEMEMORYOBJECTSEXTPROC glad_glDeleteMemoryObjectsEXT;
11383 #define glDeleteMemoryObjectsEXT glad_glDeleteMemoryObjectsEXT
11384 typedef GLboolean (APIENTRY PFNGLISMEMORYOBJECTSEXTPROC) (GLuint memoryObject);
11385 GLAPI PFNGLISMEMORYOBJECTSEXTPROC glad_glIsMemoryObjectEXT;
11386 #define glIsMemoryObjectEXT glad_glIsMemoryObjectEXT
11387 typedef void (APIENTRY PFNGLCREATEMEMORYOBJECTSEXTPROC) (GLsizei n, GLuint *memoryObjects);
11388 GLAPI PFNGLCREATEMEMORYOBJECTSEXTPROC glad_glCreateMemoryObjectsEXT;
11389 #define glCreateMemoryObjectsEXT glad_glCreateMemoryObjectsEXT
11390 typedef void (APIENTRY PFNGLMEMORYOBJECTPARAMETERIVEXTPROC) (GLuint memoryObject, GLenum pname, const GLint *params);
11391 GLAPI PFNGLMEMORYOBJECTPARAMETERIVEXTPROC glad_glMemoryObjectParameterivEXT;
11392 #define glMemoryObjectParameterivEXT glad_glMemoryObjectParameterivEXT
11393 typedef void (APIENTRY PFNGLGETMEMORYOBJECTPARAMETERIVEXTPROC) (GLuint memoryObject, GLenum pname, GLint *params);
11394 GLAPI PFNGLGETMEMORYOBJECTPARAMETERIVEXTPROC glad_glGetMemoryObjectParameterivEXT;
11395 #define glGetMemoryObjectParameterivEXT glad_glGetMemoryObjectParameterivEXT
11396 typedef void (APIENTRY PFNGLTEXSTORAGEMEM2DEXTPROC) (GLenum target, GLsizei levels, GLenum internalFormat, GLsizei width, GLsizei height, GLuint memory, GLuint64 offset);
11397 GLAPI PFNGLTEXSTORAGEMEM2DEXTPROC glad_glTexStorageMem2DEXT;
11398 #define glTexStorageMem2DEXT glad_glTexStorageMem2DEXT
11399 typedef void (APIENTRY PFNGLTEXSTORAGEMEM2DMULTISAMPLEEXTPROC) (GLenum target, GLsizei samples, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations, GLuint memory, GLuint64 offset);
11400 GLAPI PFNGLTEXSTORAGEMEM2DMULTISAMPLEEXTPROC glad_glTexStorageMem2DMultisampleEXT;
11401 #define glTexStorageMem2DMultisampleEXT glad_glTexStorageMem2DMultisampleEXT
11402 typedef void (APIENTRY PFNGLTEXSTORAGEMEM3DEXTPROC) (GLenum target, GLsizei levels, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLuint memory, GLuint64 offset);
11403 GLAPI PFNGLTEXSTORAGEMEM3DEXTPROC glad_glTexStorageMem3DEXT;
11404 #define glTexStorageMem3DEXT glad_glTexStorageMem3DEXT
11405 typedef void (APIENTRY PFNGLTEXSTORAGEMEM3DMULTISAMPLEEXTPROC) (GLenum target, GLsizei samples, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations, GLuint memory, GLuint64 offset);
11406 GLAPI PFNGLTEXSTORAGEMEM3DMULTISAMPLEEXTPROC glad_glTexStorageMem3DMultisampleEXT;
11407 #define glTexStorageMem3DMultisampleEXT glad_glTexStorageMem3DMultisampleEXT
11408 typedef void (APIENTRY PFNGLBUFFERSTORAGEMEMEXTPROC) (GLenum target, GLsizeiptr size, GLuint memory, GLuint64 offset);
11409 GLAPI PFNGLBUFFERSTORAGEMEMEXTPROC glad_glBufferStorageMemEXT;
11410 #define glBufferStorageMemEXT glad_glBufferStorageMemEXT
11411 typedef void (APIENTRY PFNGLTEXTURESTORAGEMEM2DEXTPROC) (GLuint texture, GLsizei levels, GLenum internalFormat, GLsizei width, GLsizei height, GLuint memory, GLuint64 offset);
11412 GLAPI PFNGLTEXTURESTORAGEMEM2DEXTPROC glad_glTextureStorageMem2DEXT;
11413 #define glTextureStorageMem2DEXT glad_glTextureStorageMem2DEXT
11414 typedef void (APIENTRY PFNGLTEXTURESTORAGEMEM2DMULTISAMPLEEXTPROC) (GLuint texture, GLsizei samples, GLenum internalFormat, GLsizei width, GLsizei height, GLboolean fixedSampleLocations, GLuint memory, GLuint64 offset);
11415 GLAPI PFNGLTEXTURESTORAGEMEM2DMULTISAMPLEEXTPROC glad_glTextureStorageMem2DMultisampleEXT;

```

```

11416 #define glTextureStorageMem2DMultisampleEXT glad_glTextureStorageMem2DMultisampleEXT
11417 typedef void (APIENTRY PFNGLTEXTURESTORAGEMEM3DEXTPROC) (GLuint texture, GLsizei levels, GLenum
internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLuint memory, GLuint64 offset);
11418 GLAPI PFNGLTEXTURESTORAGEMEM3DEXTPROC glad_glTextureStorageMem3DEXT;
11419 #define glTextureStorageMem3DEXT glad_glTextureStorageMem3DEXT
11420 typedef void (APIENTRY PFNGLTEXTURESTORAGEMEM3DMULTISAMPLEEXTPROC) (GLuint texture, GLsizei samples,
GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations,
GLuint memory, GLuint64 offset);
11421 GLAPI PFNGLTEXTURESTORAGEMEM3DMULTISAMPLEEXTPROC glad_glTextureStorageMem3DMultisampleEXT;
11422 #define glTextureStorageMem3DMultisampleEXT glad_glTextureStorageMem3DMultisampleEXT
11423 typedef void (APIENTRY PFNGLNAMEDBUFFERSTORAGEMEMEXTPROC) (GLuint buffer, GLsizeiptr size, GLuint
memory, GLuint64 offset);
11424 GLAPI PFNGLNAMEDBUFFERSTORAGEMEMEXTPROC glad_glNamedBufferStorageMemEXT;
11425 #define glNamedBufferStorageMemEXT glad_glNamedBufferStorageMemEXT
11426 typedef void (APIENTRY PFNGLTEXTSTORAGEMEM1DEXTPROC) (GLenum target, GLsizei levels, GLenum
internalFormat, GLsizei width, GLuint memory, GLuint64 offset);
11427 GLAPI PFNGLTEXTSTORAGEMEM1DEXTPROC glad_glTexStorageMem1DEXT;
11428 #define glTexStorageMem1DEXT glad_glTexStorageMem1DEXT
11429 typedef void (APIENTRY PFNGLTEXTURESTORAGEMEM1DEXTPROC) (GLuint texture, GLsizei levels, GLenum
internalFormat, GLsizei width, GLuint memory, GLuint64 offset);
11430 GLAPI PFNGLTEXTURESTORAGEMEM1DEXTPROC glad_glTextureStorageMem1DEXT;
11431 #define glTextureStorageMem1DEXT glad_glTextureStorageMem1DEXT
11432 #endif
11433 #ifndef GL_EXT_memory_object_fd
11434 #define GL_EXT_memory_object_fd 1
11435 GLAPI int GLAD_GL_EXT_memory_object_fd;
11436 typedef void (APIENTRY PFNGLIMPORTMEMORYFDEXTPROC) (GLuint memory, GLuint64 size, GLenum handleType,
GLint fd);
11437 GLAPI PFNGLIMPORTMEMORYFDEXTPROC glad_glImportMemoryFdEXT;
11438 #define glImportMemoryFdEXT glad_glImportMemoryFdEXT
11439 #endif
11440 #ifndef GL_EXT_memory_object_win32
11441 #define GL_EXT_memory_object_win32 1
11442 GLAPI int GLAD_GL_EXT_memory_object_win32;
11443 typedef void (APIENTRY PFNGLIMPORTMEMORYWIN32HANDLEEXTPROC) (GLuint memory, GLuint64 size, GLenum
handleType, void *handle);
11444 GLAPI PFNGLIMPORTMEMORYWIN32HANDLEEXTPROC glad_glImportMemoryWin32HandleEXT;
11445 #define glImportMemoryWin32HandleEXT glad_glImportMemoryWin32HandleEXT
11446 typedef void (APIENTRY PFNGLIMPORTMEMORYWIN32NAMEEXTPROC) (GLuint memory, GLuint64 size, GLenum
handleType, const void *name);
11447 GLAPI PFNGLIMPORTMEMORYWIN32NAMEEXTPROC glad_glImportMemoryWin32NameEXT;
11448 #define glImportMemoryWin32NameEXT glad_glImportMemoryWin32NameEXT
11449 #endif
11450 #ifndef GL_EXT_misc_attribute
11451 #define GL_EXT_misc_attribute 1
11452 GLAPI int GLAD_GL_EXT_misc_attribute;
11453 #endif
11454 #ifndef GL_EXT_multi_draw_arrays
11455 #define GL_EXT_multi_draw_arrays 1
11456 GLAPI int GLAD_GL_EXT_multi_draw_arrays;
11457 typedef void (APIENTRY PFNGLMULTIDRAWARRAYSEXTPROC) (GLenum mode, const GLint *first, const GLsizei
*count, GLsizei primcount);
11458 GLAPI PFNGLMULTIDRAWARRAYSEXTPROC glad_glMultiDrawArraysEXT;
11459 #define glMultiDrawArraysEXT glad_glMultiDrawArraysEXT
11460 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSEXTPROC) (GLenum mode, const GLsizei *count, GLenum type,
const void *const*indices, GLsizei primcount);
11461 GLAPI PFNGLMULTIDRAWELEMENTSEXTPROC glad_glMultiDrawElementsEXT;
11462 #define glMultiDrawElementsEXT glad_glMultiDrawElementsEXT
11463 #endif
11464 #ifndef GL_EXT_multisample
11465 #define GL_EXT_multisample 1
11466 GLAPI int GLAD_GL_EXT_multisample;
11467 typedef void (APIENTRY PFNGLSAMPLEMASKEXTPROC) (GLclampf value, GLboolean invert);
11468 GLAPI PFNGLSAMPLEMASKEXTPROC glad_glSampleMaskEXT;
11469 #define glSampleMaskEXT glad_glSampleMaskEXT
11470 typedef void (APIENTRY PFNGLSAMPLEPATTERNEXTPROC) (GLenum pattern);
11471 GLAPI PFNGLSAMPLEPATTERNEXTPROC glad_glSamplePatternEXT;
11472 #define glSamplePatternEXT glad_glSamplePatternEXT
11473 #endif
11474 #ifndef GL_EXT_multiview_tessellation_geometry_shader
11475 #define GL_EXT_multiview_tessellation_geometry_shader 1
11476 GLAPI int GLAD_GL_EXT_multiview_tessellation_geometry_shader;
11477 #endif
11478 #ifndef GL_EXT_multiview_texture_multisample
11479 #define GL_EXT_multiview_texture_multisample 1
11480 GLAPI int GLAD_GL_EXT_multiview_texture_multisample;
11481 #endif
11482 #ifndef GL_EXT_multiview_timer_query
11483 #define GL_EXT_multiview_timer_query 1
11484 GLAPI int GLAD_GL_EXT_multiview_timer_query;
11485 #endif
11486 #ifndef GL_EXT_packed_depth_stencil
11487 #define GL_EXT_packed_depth_stencil 1
11488 GLAPI int GLAD_GL_EXT_packed_depth_stencil;
11489 #endif
11490 #ifndef GL_EXT_packed_float
11491 #define GL_EXT_packed_float 1

```



```

11492 GLAPI int GLAD_GL_EXT_packed_float;
11493 #endif
11494 #ifndef GL_EXT_packed_pixels
11495 #define GL_EXT_packed_pixels 1
11496 GLAPI int GLAD_GL_EXT_packed_pixels;
11497 #endif
11498 #ifndef GL_EXT_paletted_texture
11499 #define GL_EXT_paletted_texture 1
11500 GLAPI int GLAD_GL_EXT_paletted_texture;
11501 typedef void (APIENTRY PFNGLCOLORTABLEEXTPROC)(GLenum target, GLenum internalFormat, GLsizei width,
    GLenum format, GLenum type, const void *table);
11502 GLAPI PFNGLCOLORTABLEEXTPROC glad_glColorTableEXT;
11503 #define glColorTableEXT glad_glColorTableEXT
11504 typedef void (APIENTRY PFNGLGETCOLORTABLEEXTPROC)(GLenum target, GLenum format, GLenum type, void
    *data);
11505 GLAPI PFNGLGETCOLORTABLEEXTPROC glad_glGetColorTableEXT;
11506 #define glGetColorTableEXT glad_glGetColorTableEXT
11507 typedef void (APIENTRY PFNGLGETCOLORTABLEPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint
    *params);
11508 GLAPI PFNGLGETCOLORTABLEPARAMETERIVEXTPROC glad_glGetColorTableParameterivEXT;
11509 #define glGetColorTableParameterivEXT glad_glGetColorTableParameterivEXT
11510 typedef void (APIENTRY PFNGLGETCOLORTABLEPARAMETERFVEXTPROC)(GLenum target, GLenum pname, GLfloat
    *params);
11511 GLAPI PFNGLGETCOLORTABLEPARAMETERFVEXTPROC glad_glGetColorTableParameterfvEXT;
11512 #define glGetColorTableParameterfvEXT glad_glGetColorTableParameterfvEXT
11513 #endif
11514 #ifndef GL_EXT_pixel_buffer_object
11515 #define GL_EXT_pixel_buffer_object 1
11516 GLAPI int GLAD_GL_EXT_pixel_buffer_object;
11517 #endif
11518 #ifndef GL_EXT_pixel_transform
11519 #define GL_EXT_pixel_transform 1
11520 GLAPI int GLAD_GL_EXT_pixel_transform;
11521 typedef void (APIENTRY PFNGLPIXELTRANSFORMPARAMETERIEXTPROC)(GLenum target, GLenum pname, GLint
    param);
11522 GLAPI PFNGLPIXELTRANSFORMPARAMETERIEXTPROC glad_glPixelTransformParameteriEXT;
11523 #define glPixelTransformParameteriEXT glad_glPixelTransformParameteriEXT
11524 typedef void (APIENTRY PFNGLPIXELTRANSFORMPARAMETERFEXTPROC)(GLenum target, GLenum pname, GLfloat
    param);
11525 GLAPI PFNGLPIXELTRANSFORMPARAMETERFEXTPROC glad_glPixelTransformParameterfEXT;
11526 #define glPixelTransformParameterfEXT glad_glPixelTransformParameterfEXT
11527 typedef void (APIENTRY PFNGLPIXELTRANSFORMPARAMETERIVEXTPROC)(GLenum target, GLenum pname, const
    GLint *params);
11528 GLAPI PFNGLPIXELTRANSFORMPARAMETERIVEXTPROC glad_glPixelTransformParameterivEXT;
11529 #define glPixelTransformParameterivEXT glad_glPixelTransformParameterivEXT
11530 typedef void (APIENTRY PFNGLPIXELTRANSFORMPARAMETERFVEXTPROC)(GLenum target, GLenum pname, const
    GLfloat *params);
11531 GLAPI PFNGLPIXELTRANSFORMPARAMETERFVEXTPROC glad_glPixelTransformParameterfvEXT;
11532 #define glPixelTransformParameterfvEXT glad_glPixelTransformParameterfvEXT
11533 typedef void (APIENTRY PFNGLGETPIXELTRANSFORMPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint
    *params);
11534 GLAPI PFNGLGETPIXELTRANSFORMPARAMETERIVEXTPROC glad_glGetPixelTransformParameterivEXT;
11535 #define glGetPixelTransformParameterivEXT glad_glGetPixelTransformParameterivEXT
11536 typedef void (APIENTRY PFNGLGETPIXELTRANSFORMPARAMETERFVEXTPROC)(GLenum target, GLenum pname, GLfloat
    *params);
11537 GLAPI PFNGLGETPIXELTRANSFORMPARAMETERFVEXTPROC glad_glGetPixelTransformParameterfvEXT;
11538 #define glGetPixelTransformParameterfvEXT glad_glGetPixelTransformParameterfvEXT
11539 #endif
11540 #ifndef GL_EXT_pixel_transform_color_table
11541 #define GL_EXT_pixel_transform_color_table 1
11542 GLAPI int GLAD_GL_EXT_pixel_transform_color_table;
11543 #endif
11544 #ifndef GL_EXT_point_parameters
11545 #define GL_EXT_point_parameters 1
11546 GLAPI int GLAD_GL_EXT_point_parameters;
11547 typedef void (APIENTRY PFNGLPOINTPARAMETERFEXTPROC)(GLenum pname, GLfloat param);
11548 GLAPI PFNGLPOINTPARAMETERFEXTPROC glad_glPointParameterfEXT;
11549 #define glPointParameterfEXT glad_glPointParameterfEXT
11550 typedef void (APIENTRY PFNGLPOINTPARAMETERFVEXTPROC)(GLenum pname, const GLfloat *params);
11551 GLAPI PFNGLPOINTPARAMETERFVEXTPROC glad_glPointParameterfvEXT;
11552 #define glPointParameterfvEXT glad_glPointParameterfvEXT
11553 #endif
11554 #ifndef GL_EXT_polygon_offset
11555 #define GL_EXT_polygon_offset 1
11556 GLAPI int GLAD_GL_EXT_polygon_offset;
11557 typedef void (APIENTRY PFNGLPOLYGONOFFSETEXTPROC)(GLfloat factor, GLfloat bias);
11558 GLAPI PFNGLPOLYGONOFFSETEXTPROC glad_glPolygonOffsetEXT;
11559 #define glPolygonOffsetEXT glad_glPolygonOffsetEXT
11560 #endif
11561 #ifndef GL_EXT_polygon_offset_clamp
11562 #define GL_EXT_polygon_offset_clamp 1
11563 GLAPI int GLAD_GL_EXT_polygon_offset_clamp;
11564 typedef void (APIENTRY PFNGLPOLYGONOFFSETCLAMPEXTPROC)(GLfloat factor, GLfloat units, GLfloat clamp);
11565 GLAPI PFNGLPOLYGONOFFSETCLAMPEXTPROC glad_glPolygonOffsetClampEXT;
11566 #define glPolygonOffsetClampEXT glad_glPolygonOffsetClampEXT
11567 #endif
11568 #ifndef GL_EXT_post_depth_coverage

```

```

11569 #define GL_EXT_post_depth_coverage 1
11570 GLAPI int GLAD_GL_EXT_post_depth_coverage;
11571 #endif
11572 #ifndef GL_EXT_provoking_vertex
11573 #define GL_EXT_provoking_vertex 1
11574 GLAPI int GLAD_GL_EXT_provoking_vertex;
11575 typedef void (APIENTRY PFNGLPROVOKINGVERTEXEXTPROC) (GLenum mode);
11576 GLAPI PFNGLPROVOKINGVERTEXEXTPROC glad_glProvokingVertexEXT;
11577 #define glProvokingVertexEXT glad_glProvokingVertexEXT
11578 #endif
11579 #ifndef GL_EXT_raster_multisample
11580 #define GL_EXT_raster_multisample 1
11581 GLAPI int GLAD_GL_EXT_raster_multisample;
11582 typedef void (APIENTRY PFNGLRASTERSAMPLESEXTPROC) (GLuint samples, GLboolean fixedsamplelocations);
11583 GLAPI PFNGLRASTERSAMPLESEXTPROC glad_glRasterSamplesEXT;
11584 #define glRasterSamplesEXT glad_glRasterSamplesEXT
11585 #endif
11586 #ifndef GL_EXT_rescale_normal
11587 #define GL_EXT_rescale_normal 1
11588 GLAPI int GLAD_GL_EXT_rescale_normal;
11589 #endif
11590 #ifndef GL_EXT_secondary_color
11591 #define GL_EXT_secondary_color 1
11592 GLAPI int GLAD_GL_EXT_secondary_color;
11593 typedef void (APIENTRY PFNGLSECONDARYCOLOR3BEXTPROC) (GLbyte red, GLbyte green, GLbyte blue);
11594 GLAPI PFNGLSECONDARYCOLOR3BEXTPROC glad_glSecondaryColor3bEXT;
11595 #define glSecondaryColor3bEXT glad_glSecondaryColor3bEXT
11596 typedef void (APIENTRY PFNGLSECONDARYCOLOR3BSEXTPROC) (const GLbyte *v);
11597 GLAPI PFNGLSECONDARYCOLOR3BSEXTPROC glad_glSecondaryColor3bvEXT;
11598 #define glSecondaryColor3bvEXT glad_glSecondaryColor3bvEXT
11599 typedef void (APIENTRY PFNGLSECONDARYCOLOR3DEXTPROC) (GLdouble red, GLdouble green, GLdouble blue);
11600 GLAPI PFNGLSECONDARYCOLOR3DEXTPROC glad_glSecondaryColor3dEXT;
11601 #define glSecondaryColor3dEXT glad_glSecondaryColor3dEXT
11602 typedef void (APIENTRY PFNGLSECONDARYCOLOR3DSEXTPROC) (const GLdouble *v);
11603 GLAPI PFNGLSECONDARYCOLOR3DSEXTPROC glad_glSecondaryColor3dvEXT;
11604 #define glSecondaryColor3dvEXT glad_glSecondaryColor3dvEXT
11605 typedef void (APIENTRY PFNGLSECONDARYCOLOR3FEXTPROC) (GLfloat red, GLfloat green, GLfloat blue);
11606 GLAPI PFNGLSECONDARYCOLOR3FEXTPROC glad_glSecondaryColor3fEXT;
11607 #define glSecondaryColor3fEXT glad_glSecondaryColor3fEXT
11608 typedef void (APIENTRY PFNGLSECONDARYCOLOR3FSEXTPROC) (const GLfloat *v);
11609 GLAPI PFNGLSECONDARYCOLOR3FSEXTPROC glad_glSecondaryColor3fvEXT;
11610 #define glSecondaryColor3fvEXT glad_glSecondaryColor3fvEXT
11611 typedef void (APIENTRY PFNGLSECONDARYCOLOR3IEXTPROC) (GLint red, GLint green, GLint blue);
11612 GLAPI PFNGLSECONDARYCOLOR3IEXTPROC glad_glSecondaryColor3iEXT;
11613 #define glSecondaryColor3iEXT glad_glSecondaryColor3iEXT
11614 typedef void (APIENTRY PFNGLSECONDARYCOLOR3ISEXTPROC) (const GLint *v);
11615 GLAPI PFNGLSECONDARYCOLOR3ISEXTPROC glad_glSecondaryColor3ivEXT;
11616 #define glSecondaryColor3ivEXT glad_glSecondaryColor3ivEXT
11617 typedef void (APIENTRY PFNGLSECONDARYCOLOR3SEXTPROC) (GLshort red, GLshort green, GLshort blue);
11618 GLAPI PFNGLSECONDARYCOLOR3SEXTPROC glad_glSecondaryColor3sEXT;
11619 #define glSecondaryColor3sEXT glad_glSecondaryColor3sEXT
11620 typedef void (APIENTRY PFNGLSECONDARYCOLOR3SEXTPROC) (const GLshort *v);
11621 GLAPI PFNGLSECONDARYCOLOR3SEXTPROC glad_glSecondaryColor3svEXT;
11622 #define glSecondaryColor3svEXT glad_glSecondaryColor3svEXT
11623 typedef void (APIENTRY PFNGLSECONDARYCOLOR3UBEXTPROC) (GLubyte red, GLubyte green, GLubyte blue);
11624 GLAPI PFNGLSECONDARYCOLOR3UBEXTPROC glad_glSecondaryColor3ubEXT;
11625 #define glSecondaryColor3ubEXT glad_glSecondaryColor3ubEXT
11626 typedef void (APIENTRY PFNGLSECONDARYCOLOR3UBSEXTPROC) (const GLubyte *v);
11627 GLAPI PFNGLSECONDARYCOLOR3UBSEXTPROC glad_glSecondaryColor3ubvEXT;
11628 #define glSecondaryColor3ubvEXT glad_glSecondaryColor3ubvEXT
11629 typedef void (APIENTRY PFNGLSECONDARYCOLOR3UIEXTPROC) (GLuint red, GLuint green, GLuint blue);
11630 GLAPI PFNGLSECONDARYCOLOR3UIEXTPROC glad_glSecondaryColor3uiEXT;
11631 #define glSecondaryColor3uiEXT glad_glSecondaryColor3uiEXT
11632 typedef void (APIENTRY PFNGLSECONDARYCOLOR3UISEXTPROC) (const GLuint *v);
11633 GLAPI PFNGLSECONDARYCOLOR3UISEXTPROC glad_glSecondaryColor3uivEXT;
11634 #define glSecondaryColor3uivEXT glad_glSecondaryColor3uivEXT
11635 typedef void (APIENTRY PFNGLSECONDARYCOLOR3USEXTPROC) (GLushort red, GLushort green, GLushort blue);
11636 GLAPI PFNGLSECONDARYCOLOR3USEXTPROC glad_glSecondaryColor3usEXT;
11637 #define glSecondaryColor3usEXT glad_glSecondaryColor3usEXT
11638 typedef void (APIENTRY PFNGLSECONDARYCOLOR3USSEXTPROC) (const GLushort *v);
11639 GLAPI PFNGLSECONDARYCOLOR3USSEXTPROC glad_glSecondaryColor3usvEXT;
11640 #define glSecondaryColor3usvEXT glad_glSecondaryColor3usvEXT
11641 typedef void (APIENTRY PFNGLSECONDARYCOLORPOINTEREXTPROC) (GLint size, GLenum type, GLsizei stride,
const void *pointer);
11642 GLAPI PFNGLSECONDARYCOLORPOINTEREXTPROC glad_glSecondaryColorPointerEXT;
11643 #define glSecondaryColorPointerEXT glad_glSecondaryColorPointerEXT
11644 #endif
11645 #ifndef GL_EXT_semaphore
11646 #define GL_EXT_semaphore 1
11647 GLAPI int GLAD_GL_EXT_semaphore;
11648 typedef void (APIENTRY PFNGLGENSEMAPHORESEXTPROC) (GLsizei n, GLuint *semaphores);
11649 GLAPI PFNGLGENSEMAPHORESEXTPROC glad_glGenSemaphoresEXT;
11650 #define glGenSemaphoresEXT glad_glGenSemaphoresEXT
11651 typedef void (APIENTRY PFNGLDELETESEMAPHORESEXTPROC) (GLsizei n, const GLuint *semaphores);
11652 GLAPI PFNGLDELETESEMAPHORESEXTPROC glad_glDeleteSemaphoresEXT;
11653 #define glDeleteSemaphoresEXT glad_glDeleteSemaphoresEXT
11654 typedef GLboolean (APIENTRY PFNGLISSEMAPHOREEXTPROC) (GLuint semaphore);

```

```

11655 GLAPI PFNGLISSEMAPHOREEXTPROC glad_glIsSemaphoreEXT;
11656 #define glIsSemaphoreEXT glad_glIsSemaphoreEXT
11657 typedef void (APIENTRY PFNGLSEMAPHOREPARAMETERUI64VEXTPROC) (GLuint semaphore, GLenum pname, const
    GLuint64 *params);
11658 GLAPI PFNGLSEMAPHOREPARAMETERUI64VEXTPROC glad_glSemaphoreParameterui64vEXT;
11659 #define glSemaphoreParameterui64vEXT glad_glSemaphoreParameterui64vEXT
11660 typedef void (APIENTRY PFNGLGETSEMAPHOREPARAMETERUI64VEXTPROC) (GLuint semaphore, GLenum pname,
    GLuint64 *params);
11661 GLAPI PFNGLGETSEMAPHOREPARAMETERUI64VEXTPROC glad_glGetSemaphoreParameterui64vEXT;
11662 #define glGetSemaphoreParameterui64vEXT glad_glGetSemaphoreParameterui64vEXT
11663 typedef void (APIENTRY PFNGLWAITSEMAPHOREEXTPROC) (GLuint semaphore, GLuint numBufferBarriers, const
    GLuint *buffers, GLuint numTextureBarriers, const GLuint *textures, const GLenum *srcLayouts);
11664 GLAPI PFNGLWAITSEMAPHOREEXTPROC glad_glWaitSemaphoreEXT;
11665 #define glWaitSemaphoreEXT glad_glWaitSemaphoreEXT
11666 typedef void (APIENTRY PFNGLSIGNALSEMAPHOREEXTPROC) (GLuint semaphore, GLuint numBufferBarriers, const
    GLuint *buffers, GLuint numTextureBarriers, const GLuint *textures, const GLenum *dstLayouts);
11667 GLAPI PFNGLSIGNALSEMAPHOREEXTPROC glad_glSignalSemaphoreEXT;
11668 #define glSignalSemaphoreEXT glad_glSignalSemaphoreEXT
11669 #endif
11670 #ifndef GL_EXT_semaphore_fd
11671 #define GL_EXT_semaphore_fd 1
11672 GLAPI int GLAD_GL_EXT_semaphore_fd;
11673 typedef void (APIENTRY PFNGLIMPORTSEMAPHOREFDEXTPROC) (GLuint semaphore, GLenum handleType, GLint fd);
11674 GLAPI PFNGLIMPORTSEMAPHOREFDEXTPROC glad_glImportSemaphoreFdEXT;
11675 #define glImportSemaphoreFdEXT glad_glImportSemaphoreFdEXT
11676 #endif
11677 #ifndef GL_EXT_semaphore_win32
11678 #define GL_EXT_semaphore_win32 1
11679 GLAPI int GLAD_GL_EXT_semaphore_win32;
11680 typedef void (APIENTRY PFNGLIMPORTSEMAPHOREWIN32HANDLEEXTPROC) (GLuint semaphore, GLenum handleType,
    void *handle);
11681 GLAPI PFNGLIMPORTSEMAPHOREWIN32HANDLEEXTPROC glad_glImportSemaphoreWin32HandleEXT;
11682 #define glImportSemaphoreWin32HandleEXT glad_glImportSemaphoreWin32HandleEXT
11683 typedef void (APIENTRY PFNGLIMPORTSEMAPHOREWIN32NAMEEXTPROC) (GLuint semaphore, GLenum handleType,
    const void *name);
11684 GLAPI PFNGLIMPORTSEMAPHOREWIN32NAMEEXTPROC glad_glImportSemaphoreWin32NameEXT;
11685 #define glImportSemaphoreWin32NameEXT glad_glImportSemaphoreWin32NameEXT
11686 #endif
11687 #ifndef GL_EXT_separate_shader_objects
11688 #define GL_EXT_separate_shader_objects 1
11689 GLAPI int GLAD_GL_EXT_separate_shader_objects;
11690 typedef void (APIENTRY PFNGLUSESHADERPROGRAMEXTPROC) (GLenum type, GLuint program);
11691 GLAPI PFNGLUSESHADERPROGRAMEXTPROC glad_glUseProgramEXT;
11692 #define glUseProgramEXT glad_glUseProgramEXT
11693 typedef void (APIENTRY PFNGLACTIVEPROGRAMEXTPROC) (GLuint program);
11694 GLAPI PFNGLACTIVEPROGRAMEXTPROC glad_glActiveProgramEXT;
11695 #define glActiveProgramEXT glad_glActiveProgramEXT
11696 typedef GLuint (APIENTRY PFNGLCREATESHADERPROGRAMEXTPROC) (GLenum type, const GLchar *string);
11697 GLAPI PFNGLCREATESHADERPROGRAMEXTPROC glad_glCreateShaderProgramEXT;
11698 #define glCreateShaderProgramEXT glad_glCreateShaderProgramEXT
11699 typedef void (APIENTRY PFNGLACTIVESHADERPROGRAMEXTPROC) (GLuint pipeline, GLuint program);
11700 GLAPI PFNGLACTIVESHADERPROGRAMEXTPROC glad_glActiveShaderProgramEXT;
11701 #define glActiveShaderProgramEXT glad_glActiveShaderProgramEXT
11702 typedef void (APIENTRY PFNGLBINDPROGRAMPIPELINEEXTPROC) (GLuint pipeline);
11703 GLAPI PFNGLBINDPROGRAMPIPELINEEXTPROC glad_glBindProgramPipelineEXT;
11704 #define glBindProgramPipelineEXT glad_glBindProgramPipelineEXT
11705 typedef GLuint (APIENTRY PFNGLCREATESHADERPROGRAMVEXTPROC) (GLenum type, GLsizei count, const GLchar
    *const*strings);
11706 GLAPI PFNGLCREATESHADERPROGRAMVEXTPROC glad_glCreateShaderProgramvEXT;
11707 #define glCreateShaderProgramvEXT glad_glCreateShaderProgramvEXT
11708 typedef void (APIENTRY PFNGLDELETEPROGRAMPIPELINESEXTPROC) (GLsizei n, const GLuint *pipelines);
11709 GLAPI PFNGLDELETEPROGRAMPIPELINESEXTPROC glad_glDeleteProgramPipelinesEXT;
11710 #define glDeleteProgramPipelinesEXT glad_glDeleteProgramPipelinesEXT
11711 typedef void (APIENTRY PFNGLGENPROGRAMPIPELINESEXTPROC) (GLsizei n, GLuint *pipelines);
11712 GLAPI PFNGLGENPROGRAMPIPELINESEXTPROC glad_glGenProgramPipelinesEXT;
11713 #define glGenProgramPipelinesEXT glad_glGenProgramPipelinesEXT
11714 typedef void (APIENTRY PFNGLGETPROGRAMPIPELINEINFOLOGEXTPROC) (GLuint pipeline, GLsizei bufSize,
    GLsizei *length, GLchar *infoLog);
11715 GLAPI PFNGLGETPROGRAMPIPELINEINFOLOGEXTPROC glad_glGetProgramPipelineInfoLogEXT;
11716 #define glGetProgramPipelineInfoLogEXT glad_glGetProgramPipelineInfoLogEXT
11717 typedef void (APIENTRY PFNGLGETPROGRAMPIPELINEIVEXTPROC) (GLuint pipeline, GLenum pname, GLint
    *params);
11718 GLAPI PFNGLGETPROGRAMPIPELINEIVEXTPROC glad_glGetProgramPipelineivEXT;
11719 #define glGetProgramPipelineivEXT glad_glGetProgramPipelineivEXT
11720 typedef GLboolean (APIENTRY PFNGLISPROGRAMPIPELINEEXTPROC) (GLuint pipeline);
11721 GLAPI PFNGLISPROGRAMPIPELINEEXTPROC glad_glIsProgramPipelineEXT;
11722 #define glIsProgramPipelineEXT glad_glIsProgramPipelineEXT
11723 typedef void (APIENTRY PFNGLUSEPROGRAMSTAGESEXTPROC) (GLuint pipeline, GLbitfield stages, GLuint
    program);
11724 GLAPI PFNGLUSEPROGRAMSTAGESEXTPROC glad_glUseProgramStagesEXT;
11725 #define glUseProgramStagesEXT glad_glUseProgramStagesEXT
11726 typedef void (APIENTRY PFNGLVALIDATEPROGRAMPIPELINEEXTPROC) (GLuint pipeline);
11727 GLAPI PFNGLVALIDATEPROGRAMPIPELINEEXTPROC glad_glValidateProgramPipelineEXT;
11728 #define glValidateProgramPipelineEXT glad_glValidateProgramPipelineEXT
11729 #endif
11730 #ifndef GL_EXT_separate_specular_color
11731 #define GL_EXT_separate_specular_color 1

```

```

11732 GLAPI int GLAD_GL_EXT_separate_specular_color;
11733 #endif
11734 #ifndef GL_EXT_shader_framebuffer_fetch
11735 #define GL_EXT_shader_framebuffer_fetch 1
11736 GLAPI int GLAD_GL_EXT_shader_framebuffer_fetch;
11737 #endif
11738 #ifndef GL_EXT_shader_framebuffer_fetch_non_coherent
11739 #define GL_EXT_shader_framebuffer_fetch_non_coherent 1
11740 GLAPI int GLAD_GL_EXT_shader_framebuffer_fetch_non_coherent;
11741 typedef void (APIENTRY PFNGLFRAMEBUFFERFETCHBARRIEREXTPROC)(void);
11742 GLAPI PFNGLFRAMEBUFFERFETCHBARRIEREXTPROC glad_glFramebufferFetchBarrierEXT;
11743 #define glFramebufferFetchBarrierEXT glad_glFramebufferFetchBarrierEXT
11744 #endif
11745 #ifndef GL_EXT_shader_image_load_formatted
11746 #define GL_EXT_shader_image_load_formatted 1
11747 GLAPI int GLAD_GL_EXT_shader_image_load_formatted;
11748 #endif
11749 #ifndef GL_EXT_shader_image_load_store
11750 #define GL_EXT_shader_image_load_store 1
11751 GLAPI int GLAD_GL_EXT_shader_image_load_store;
11752 typedef void (APIENTRY PFNGLBINDIMAGETEXTUREEXTPROC)(GLuint index, GLuint texture, GLint level,
GLboolean layered, GLint layer, GLenum access, GLint format);
11753 GLAPI PFNGLBINDIMAGETEXTUREEXTPROC glad_glBindImageTextureEXT;
11754 #define glBindImageTextureEXT glad_glBindImageTextureEXT
11755 typedef void (APIENTRY PFNGLMEMORYBARRIEREXTPROC)(GLbitfield barriers);
11756 GLAPI PFNGLMEMORYBARRIEREXTPROC glad_glMemoryBarrierEXT;
11757 #define glMemoryBarrierEXT glad_glMemoryBarrierEXT
11758 #endif
11759 #ifndef GL_EXT_shader_integer_mix
11760 #define GL_EXT_shader_integer_mix 1
11761 GLAPI int GLAD_GL_EXT_shader_integer_mix;
11762 #endif
11763 #ifndef GL_EXT_shader_samples_identical
11764 #define GL_EXT_shader_samples_identical 1
11765 GLAPI int GLAD_GL_EXT_shader_samples_identical;
11766 #endif
11767 #ifndef GL_EXT_shadow_funcs
11768 #define GL_EXT_shadow_funcs 1
11769 GLAPI int GLAD_GL_EXT_shadow_funcs;
11770 #endif
11771 #ifndef GL_EXT_shared_texture_palette
11772 #define GL_EXT_shared_texture_palette 1
11773 GLAPI int GLAD_GL_EXT_shared_texture_palette;
11774 #endif
11775 #ifndef GL_EXT_sparse_texture2
11776 #define GL_EXT_sparse_texture2 1
11777 GLAPI int GLAD_GL_EXT_sparse_texture2;
11778 #endif
11779 #ifndef GL_EXT_stencil_clear_tag
11780 #define GL_EXT_stencil_clear_tag 1
11781 GLAPI int GLAD_GL_EXT_stencil_clear_tag;
11782 typedef void (APIENTRY PFNGLSTENCILCLEARTAGEXTPROC)(GLsizei stencilTagBits, GLuint stencilClearTag);
11783 GLAPI PFNGLSTENCILCLEARTAGEXTPROC glad_glStencilClearTagEXT;
11784 #define glStencilClearTagEXT glad_glStencilClearTagEXT
11785 #endif
11786 #ifndef GL_EXT_stencil_two_side
11787 #define GL_EXT_stencil_two_side 1
11788 GLAPI int GLAD_GL_EXT_stencil_two_side;
11789 typedef void (APIENTRY PFNGLACTIVESTENCILFACEEXTPROC)(GLenum face);
11790 GLAPI PFNGLACTIVESTENCILFACEEXTPROC glad_glActiveStencilFaceEXT;
11791 #define glActiveStencilFaceEXT glad_glActiveStencilFaceEXT
11792 #endif
11793 #ifndef GL_EXT_stencil_wrap
11794 #define GL_EXT_stencil_wrap 1
11795 GLAPI int GLAD_GL_EXT_stencil_wrap;
11796 #endif
11797 #ifndef GL_EXT_subtexture
11798 #define GL_EXT_subtexture 1
11799 GLAPI int GLAD_GL_EXT_subtexture;
11800 typedef void (APIENTRY PFNGLTEXSUBIMAGE1DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLsizei
width, GLenum format, GLenum type, const void *pixels);
11801 GLAPI PFNGLTEXSUBIMAGE1DEXTPROC glad_glTexSubImage1DEXT;
11802 #define glTexSubImage1DEXT glad_glTexSubImage1DEXT
11803 typedef void (APIENTRY PFNGLTEXSUBIMAGE2DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLint
yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *pixels);
11804 GLAPI PFNGLTEXSUBIMAGE2DEXTPROC glad_glTexSubImage2DEXT;
11805 #define glTexSubImage2DEXT glad_glTexSubImage2DEXT
11806 #endif
11807 #ifndef GL_EXT_texture
11808 #define GL_EXT_texture 1
11809 GLAPI int GLAD_GL_EXT_texture;
11810 #endif
11811 #ifndef GL_EXT_texture3D
11812 #define GL_EXT_texture3D 1
11813 GLAPI int GLAD_GL_EXT_texture3D;
11814 typedef void (APIENTRY PFNGLTEXIMAGE3DEXTPROC)(GLenum target, GLint level, GLenum internalformat,
GLsizei width, GLsizei height, GLsizei depth, GLint border, GLenum format, GLenum type, const void

```

```

        *pixels);
11815 GLAPI PFNGLTEXIMAGE3DEXTPROC glad_glTexImage3DEXT;
11816 #define glTexImage3DEXT glad_glTexImage3DEXT
11817 typedef void (APIENTRY PFNGLTEXSUBIMAGE3DEXTPROC) (GLenum target, GLint level, GLint xoffset, GLint
        yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type,
        const void *pixels);
11818 GLAPI PFNGLTEXSUBIMAGE3DEXTPROC glad_glTexSubImage3DEXT;
11819 #define glTexSubImage3DEXT glad_glTexSubImage3DEXT
11820 #endif
11821 #ifndef GL_EXT_texture_array
11822 #define GL_EXT_texture_array 1
11823 GLAPI int GLAD_GL_EXT_texture_array;
11824 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTURELAYEREXTPROC) (GLenum target, GLenum attachment, GLuint
        texture, GLint level, GLint layer);
11825 GLAPI PFNGLFRAMEBUFFERTEXTURELAYEREXTPROC glad_glFramebufferTextureLayerEXT;
11826 #define glFramebufferTextureLayerEXT glad_glFramebufferTextureLayerEXT
11827 #endif
11828 #ifndef GL_EXT_texture_buffer_object
11829 #define GL_EXT_texture_buffer_object 1
11830 GLAPI int GLAD_GL_EXT_texture_buffer_object;
11831 typedef void (APIENTRY PFNGLTEXBUFFEREXTPROC) (GLenum target, GLenum internalformat, GLuint buffer);
11832 GLAPI PFNGLTEXBUFFEREXTPROC glad_glTexBufferEXT;
11833 #define glTexBufferEXT glad_glTexBufferEXT
11834 #endif
11835 #ifndef GL_EXT_texture_compression_latc
11836 #define GL_EXT_texture_compression_latc 1
11837 GLAPI int GLAD_GL_EXT_texture_compression_latc;
11838 #endif
11839 #ifndef GL_EXT_texture_compression_rgtc
11840 #define GL_EXT_texture_compression_rgtc 1
11841 GLAPI int GLAD_GL_EXT_texture_compression_rgtc;
11842 #endif
11843 #ifndef GL_EXT_texture_compression_s3tc
11844 #define GL_EXT_texture_compression_s3tc 1
11845 GLAPI int GLAD_GL_EXT_texture_compression_s3tc;
11846 #endif
11847 #ifndef GL_EXT_texture_cube_map
11848 #define GL_EXT_texture_cube_map 1
11849 GLAPI int GLAD_GL_EXT_texture_cube_map;
11850 #endif
11851 #ifndef GL_EXT_texture_env_add
11852 #define GL_EXT_texture_env_add 1
11853 GLAPI int GLAD_GL_EXT_texture_env_add;
11854 #endif
11855 #ifndef GL_EXT_texture_env_combine
11856 #define GL_EXT_texture_env_combine 1
11857 GLAPI int GLAD_GL_EXT_texture_env_combine;
11858 #endif
11859 #ifndef GL_EXT_texture_env_dot3
11860 #define GL_EXT_texture_env_dot3 1
11861 GLAPI int GLAD_GL_EXT_texture_env_dot3;
11862 #endif
11863 #ifndef GL_EXT_texture_filter_anisotropic
11864 #define GL_EXT_texture_filter_anisotropic 1
11865 GLAPI int GLAD_GL_EXT_texture_filter_anisotropic;
11866 #endif
11867 #ifndef GL_EXT_texture_filter_minmax
11868 #define GL_EXT_texture_filter_minmax 1
11869 GLAPI int GLAD_GL_EXT_texture_filter_minmax;
11870 #endif
11871 #ifndef GL_EXT_texture_integer
11872 #define GL_EXT_texture_integer 1
11873 GLAPI int GLAD_GL_EXT_texture_integer;
11874 typedef void (APIENTRY PFNGLTEXPARAMETERIIVEXTPROC) (GLenum target, GLenum pname, const GLint
        *params);
11875 GLAPI PFNGLTEXPARAMETERIIVEXTPROC glad_glTexParameterIivEXT;
11876 #define glTexParameterIivEXT glad_glTexParameterIivEXT
11877 typedef void (APIENTRY PFNGLTEXPARAMETERIUIVEXTPROC) (GLenum target, GLenum pname, const GLuint
        *params);
11878 GLAPI PFNGLTEXPARAMETERIUIVEXTPROC glad_glTexParameterIuiEXT;
11879 #define glTexParameterIuiEXT glad_glTexParameterIuiEXT
11880 typedef void (APIENTRY PFNGLGETTEXPARAMETERIIVEXTPROC) (GLenum target, GLenum pname, GLint *params);
11881 GLAPI PFNGLGETTEXPARAMETERIIVEXTPROC glad_glGetTexParameterIivEXT;
11882 #define glGetTexParameterIivEXT glad_glGetTexParameterIivEXT
11883 typedef void (APIENTRY PFNGLGETTEXPARAMETERIUIVEXTPROC) (GLenum target, GLenum pname, GLuint *params);
11884 GLAPI PFNGLGETTEXPARAMETERIUIVEXTPROC glad_glGetTexParameterIuiEXT;
11885 #define glGetTexParameterIuiEXT glad_glGetTexParameterIuiEXT
11886 typedef void (APIENTRY PFNGLCLEARCOLORIIECTPROC) (GLint red, GLint green, GLint blue, GLint alpha);
11887 GLAPI PFNGLCLEARCOLORIIECTPROC glad_glClearColorIiEXT;
11888 #define glClearColorIiEXT glad_glClearColorIiEXT
11889 typedef void (APIENTRY PFNGLCLEARCOLORIUIEXTPROC) (GLuint red, GLuint green, GLuint blue, GLuint
        alpha);
11890 GLAPI PFNGLCLEARCOLORIUIEXTPROC glad_glClearColorIuiEXT;
11891 #define glClearColorIuiEXT glad_glClearColorIuiEXT
11892 #endif
11893 #ifndef GL_EXT_texture_lod_bias
11894 #define GL_EXT_texture_lod_bias 1

```



```

11895 GLAPI int GLAD_GL_EXT_texture_lod_bias;
11896 #endif
11897 #ifndef GL_EXT_texture_mirror_clamp
11898 #define GL_EXT_texture_mirror_clamp 1
11899 GLAPI int GLAD_GL_EXT_texture_mirror_clamp;
11900 #endif
11901 #ifndef GL_EXT_texture_object
11902 #define GL_EXT_texture_object 1
11903 GLAPI int GLAD_GL_EXT_texture_object;
11904 typedef GLboolean (APIENTRY PFNGLARETEXTURESRESIDENTEXTPROC) (GLsizei n, const GLuint *textures,
GLboolean *residences);
11905 GLAPI PFNGLARETEXTURESRESIDENTEXTPROC glad_glAreTexturesResidentEXT;
11906 #define glAreTexturesResidentEXT glad_glAreTexturesResidentEXT
11907 typedef void (APIENTRY PFNGLBINDTEXTUREEXTPROC) (GLenum target, GLuint texture);
11908 GLAPI PFNGLBINDTEXTUREEXTPROC glad_glBindTextureEXT;
11909 #define glBindTextureEXT glad_glBindTextureEXT
11910 typedef void (APIENTRY PFNGLDELETETEXTURESEXTPROC) (GLsizei n, const GLuint *textures);
11911 GLAPI PFNGLDELETETEXTURESEXTPROC glad_glDeleteTexturesEXT;
11912 #define glDeleteTexturesEXT glad_glDeleteTexturesEXT
11913 typedef void (APIENTRY PFNGLGENTEXTURESEXTPROC) (GLsizei n, GLuint *textures);
11914 GLAPI PFNGLGENTEXTURESEXTPROC glad_glGenTexturesEXT;
11915 #define glGenTexturesEXT glad_glGenTexturesEXT
11916 typedef GLboolean (APIENTRY PFNGLISTEXTUREEXTPROC) (GLuint texture);
11917 GLAPI PFNGLISTEXTUREEXTPROC glad_glIsTextureEXT;
11918 #define glIsTextureEXT glad_glIsTextureEXT
11919 typedef void (APIENTRY PFNGLPRIORITIZETEXTURESEXTPROC) (GLsizei n, const GLuint *textures, const
GLclampf *priorities);
11920 GLAPI PFNGLPRIORITIZETEXTURESEXTPROC glad_glPrioritizeTexturesEXT;
11921 #define glPrioritizeTexturesEXT glad_glPrioritizeTexturesEXT
11922 #endif
11923 #ifndef GL_EXT_texture_perturb_normal
11924 #define GL_EXT_texture_perturb_normal 1
11925 GLAPI int GLAD_GL_EXT_texture_perturb_normal;
11926 typedef void (APIENTRY PFNGLTEXTURENORMALEXTPROC) (GLenum mode);
11927 GLAPI PFNGLTEXTURENORMALEXTPROC glad_glTextureNormalEXT;
11928 #define glTextureNormalEXT glad_glTextureNormalEXT
11929 #endif
11930 #ifndef GL_EXT_texture_sRGB
11931 #define GL_EXT_texture_sRGB 1
11932 GLAPI int GLAD_GL_EXT_texture_sRGB;
11933 #endif
11934 #ifndef GL_EXT_texture_sRGB_R8
11935 #define GL_EXT_texture_sRGB_R8 1
11936 GLAPI int GLAD_GL_EXT_texture_sRGB_R8;
11937 #endif
11938 #ifndef GL_EXT_texture_sRGB_RG8
11939 #define GL_EXT_texture_sRGB_RG8 1
11940 GLAPI int GLAD_GL_EXT_texture_sRGB_RG8;
11941 #endif
11942 #ifndef GL_EXT_texture_sRGB_decode
11943 #define GL_EXT_texture_sRGB_decode 1
11944 GLAPI int GLAD_GL_EXT_texture_sRGB_decode;
11945 #endif
11946 #ifndef GL_EXT_texture_shadow_lod
11947 #define GL_EXT_texture_shadow_lod 1
11948 GLAPI int GLAD_GL_EXT_texture_shadow_lod;
11949 #endif
11950 #ifndef GL_EXT_texture_shared_exponent
11951 #define GL_EXT_texture_shared_exponent 1
11952 GLAPI int GLAD_GL_EXT_texture_shared_exponent;
11953 #endif
11954 #ifndef GL_EXT_texture_snorm
11955 #define GL_EXT_texture_snorm 1
11956 GLAPI int GLAD_GL_EXT_texture_snorm;
11957 #endif
11958 #ifndef GL_EXT_texture_storage
11959 #define GL_EXT_texture_storage 1
11960 GLAPI int GLAD_GL_EXT_texture_storage;
11961 typedef void (APIENTRY PFNGLTEXTSTORAGE1DEXTPROC) (GLenum target, GLsizei levels, GLenum
internalformat, GLsizei width);
11962 GLAPI PFNGLTEXTSTORAGE1DEXTPROC glad_glTexStorage1DEXT;
11963 #define glTexStorage1DEXT glad_glTexStorage1DEXT
11964 typedef void (APIENTRY PFNGLTEXTSTORAGE2DEXTPROC) (GLenum target, GLsizei levels, GLenum
internalformat, GLsizei width, GLsizei height);
11965 GLAPI PFNGLTEXTSTORAGE2DEXTPROC glad_glTexStorage2DEXT;
11966 #define glTexStorage2DEXT glad_glTexStorage2DEXT
11967 typedef void (APIENTRY PFNGLTEXTSTORAGE3DEXTPROC) (GLenum target, GLsizei levels, GLenum
internalformat, GLsizei width, GLsizei height, GLsizei depth);
11968 GLAPI PFNGLTEXTSTORAGE3DEXTPROC glad_glTexStorage3DEXT;
11969 #define glTexStorage3DEXT glad_glTexStorage3DEXT
11970 #endif
11971 #ifndef GL_EXT_texture_swizzle
11972 #define GL_EXT_texture_swizzle 1
11973 GLAPI int GLAD_GL_EXT_texture_swizzle;
11974 #endif
11975 #ifndef GL_EXT_timer_query
11976 #define GL_EXT_timer_query 1

```

```

11977 GLAPI int GLAD_GL_EXT_timer_query;
11978 typedef void (APIENTRY PFNGLGETQUERYOBJECTI64VEXTPROC) (GLuint id, GLenum pname, GLint64 *params);
11979 GLAPI PFNGLGETQUERYOBJECTI64VEXTPROC glad_glGetQueryObjecti64vEXT;
11980 #define glGetQueryObjecti64vEXT glad_glGetQueryObjecti64vEXT
11981 typedef void (APIENTRY PFNGLGETQUERYOBJECTUI64VEXTPROC) (GLuint id, GLenum pname, GLuint64 *params);
11982 GLAPI PFNGLGETQUERYOBJECTUI64VEXTPROC glad_glGetQueryObjectui64vEXT;
11983 #define glGetQueryObjectui64vEXT glad_glGetQueryObjectui64vEXT
11984 #endif
11985 #ifndef GL_EXT_transform_feedback
11986 #define GL_EXT_transform_feedback 1
11987 GLAPI int GLAD_GL_EXT_transform_feedback;
11988 typedef void (APIENTRY PFNGLBEGINTRANSFORMFEEDBACKEXTPROC) (GLenum primitiveMode);
11989 GLAPI PFNGLBEGINTRANSFORMFEEDBACKEXTPROC glad_glBeginTransformFeedbackEXT;
11990 #define glBeginTransformFeedbackEXT glad_glBeginTransformFeedbackEXT
11991 typedef void (APIENTRY PFNGLENDTRANSFORMFEEDBACKEXTPROC) (void);
11992 GLAPI PFNGLENDTRANSFORMFEEDBACKEXTPROC glad_glEndTransformFeedbackEXT;
11993 #define glEndTransformFeedbackEXT glad_glEndTransformFeedbackEXT
11994 typedef void (APIENTRY PFNGLBINDBUFFERRANGEEXTPROC) (GLenum target, GLuint index, GLuint buffer,
GLintptr offset, GLsizeiptr size);
11995 GLAPI PFNGLBINDBUFFERRANGEEXTPROC glad_glBindBufferRangeEXT;
11996 #define glBindBufferRangeEXT glad_glBindBufferRangeEXT
11997 typedef void (APIENTRY PFNGLBINDBUFFEROFFSETEXTPROC) (GLenum target, GLuint index, GLuint buffer,
GLintptr offset);
11998 GLAPI PFNGLBINDBUFFEROFFSETEXTPROC glad_glBindBufferOffsetEXT;
11999 #define glBindBufferOffsetEXT glad_glBindBufferOffsetEXT
12000 typedef void (APIENTRY PFNGLBINDBUFFERBASEEXTPROC) (GLenum target, GLuint index, GLuint buffer);
12001 GLAPI PFNGLBINDBUFFERBASEEXTPROC glad_glBindBufferBaseEXT;
12002 #define glBindBufferBaseEXT glad_glBindBufferBaseEXT
12003 typedef void (APIENTRY PFNGLTRANSFORMFEEDBACKVARYINGSEXTPROC) (GLuint program, GLsizei count, const
GLchar *const*varyings, GLenum bufferMode);
12004 GLAPI PFNGLTRANSFORMFEEDBACKVARYINGSEXTPROC glad_glTransformFeedbackVaryingsEXT;
12005 #define glTransformFeedbackVaryingsEXT glad_glTransformFeedbackVaryingsEXT
12006 typedef void (APIENTRY PFNGLGETTRANSFORMFEEDBACKVARYINGEXTPROC) (GLuint program, GLuint index, GLsizei
bufSize, GLsizei *length, GLsizei *size, GLenum *type, GLchar *name);
12007 GLAPI PFNGLGETTRANSFORMFEEDBACKVARYINGEXTPROC glad_glGetTransformFeedbackVaryingEXT;
12008 #define glGetTransformFeedbackVaryingEXT glad_glGetTransformFeedbackVaryingEXT
12009 #endif
12010 #ifndef GL_EXT_vertex_array
12011 #define GL_EXT_vertex_array 1
12012 GLAPI int GLAD_GL_EXT_vertex_array;
12013 typedef void (APIENTRY PFNGLARRAYELEMENTEXTPROC) (GLint i);
12014 GLAPI PFNGLARRAYELEMENTEXTPROC glad_glArrayElementEXT;
12015 #define glArrayElementEXT glad_glArrayElementEXT
12016 typedef void (APIENTRY PFNGLCOLORPOINTEREXTPROC) (GLint size, GLenum type, GLsizei stride, GLsizei
count, const void *pointer);
12017 GLAPI PFNGLCOLORPOINTEREXTPROC glad_glColorPointerEXT;
12018 #define glColorPointerEXT glad_glColorPointerEXT
12019 typedef void (APIENTRY PFNGLDRAWARRAYSEXTPROC) (GLenum mode, GLint first, GLsizei count);
12020 GLAPI PFNGLDRAWARRAYSEXTPROC glad_glDrawArraysEXT;
12021 #define glDrawArraysEXT glad_glDrawArraysEXT
12022 typedef void (APIENTRY PFNGLEDGEFLAGPOINTEREXTPROC) (GLsizei stride, GLsizei count, const GLboolean
*pointer);
12023 GLAPI PFNGLEDGEFLAGPOINTEREXTPROC glad_glEdgeFlagPointerEXT;
12024 #define glEdgeFlagPointerEXT glad_glEdgeFlagPointerEXT
12025 typedef void (APIENTRY PFNGLGETPOINTERVEXTPROC) (GLenum pname, void **params);
12026 GLAPI PFNGLGETPOINTERVEXTPROC glad_glGetPointervEXT;
12027 #define glGetPointervEXT glad_glGetPointervEXT
12028 typedef void (APIENTRY PFNGLINDEXPOINTEREXTPROC) (GLenum type, GLsizei stride, GLsizei count, const
void *pointer);
12029 GLAPI PFNGLINDEXPOINTEREXTPROC glad_glIndexPointerEXT;
12030 #define glIndexPointerEXT glad_glIndexPointerEXT
12031 typedef void (APIENTRY PFNGLNORMALPOINTEREXTPROC) (GLenum type, GLsizei stride, GLsizei count, const
void *pointer);
12032 GLAPI PFNGLNORMALPOINTEREXTPROC glad_glNormalPointerEXT;
12033 #define glNormalPointerEXT glad_glNormalPointerEXT
12034 typedef void (APIENTRY PFNGLTEXCOORDPOINTEREXTPROC) (GLint size, GLenum type, GLsizei stride, GLsizei
count, const void *pointer);
12035 GLAPI PFNGLTEXCOORDPOINTEREXTPROC glad_glTexCoordPointerEXT;
12036 #define glTexCoordPointerEXT glad_glTexCoordPointerEXT
12037 typedef void (APIENTRY PFNGLVERTEXPOINTEREXTPROC) (GLint size, GLenum type, GLsizei stride, GLsizei
count, const void *pointer);
12038 GLAPI PFNGLVERTEXPOINTEREXTPROC glad_glVertexPointerEXT;
12039 #define glVertexPointerEXT glad_glVertexPointerEXT
12040 #endif
12041 #ifndef GL_EXT_vertex_array_bgra
12042 #define GL_EXT_vertex_array_bgra 1
12043 GLAPI int GLAD_GL_EXT_vertex_array_bgra;
12044 #endif
12045 #ifndef GL_EXT_vertex_attrib_64bit
12046 #define GL_EXT_vertex_attrib_64bit 1
12047 GLAPI int GLAD_GL_EXT_vertex_attrib_64bit;
12048 typedef void (APIENTRY PFNGLVERTEXATTRIBL1DEXTPROC) (GLuint index, GLdouble x);
12049 GLAPI PFNGLVERTEXATTRIBL1DEXTPROC glad_glVertexAttribL1dEXT;
12050 #define glVertexAttribL1dEXT glad_glVertexAttribL1dEXT
12051 typedef void (APIENTRY PFNGLVERTEXATTRIBL2DEXTPROC) (GLuint index, GLdouble x, GLdouble y);
12052 GLAPI PFNGLVERTEXATTRIBL2DEXTPROC glad_glVertexAttribL2dEXT;
12053 #define glVertexAttribL2dEXT glad_glVertexAttribL2dEXT

```



```

12054 typedef void (APIENTRY PFNGLVERTEXATTRIBL3DEXTPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble
z);
12055 GLAPI PFNGLVERTEXATTRIBL3DEXTPROC glad_glVertexAttribL3dEXT;
12056 #define glVertexAttribL3dEXT glad_glVertexAttribL3dEXT
12057 typedef void (APIENTRY PFNGLVERTEXATTRIBL4DEXTPROC) (GLuint index, GLdouble x, GLdouble y, GLdouble z,
GLdouble w);
12058 GLAPI PFNGLVERTEXATTRIBL4DEXTPROC glad_glVertexAttribL4dEXT;
12059 #define glVertexAttribL4dEXT glad_glVertexAttribL4dEXT
12060 typedef void (APIENTRY PFNGLVERTEXATTRIBL1DVEXTPROC) (GLuint index, const GLdouble *v);
12061 GLAPI PFNGLVERTEXATTRIBL1DVEXTPROC glad_glVertexAttribL1dvEXT;
12062 #define glVertexAttribL1dvEXT glad_glVertexAttribL1dvEXT
12063 typedef void (APIENTRY PFNGLVERTEXATTRIBL2DVEXTPROC) (GLuint index, const GLdouble *v);
12064 GLAPI PFNGLVERTEXATTRIBL2DVEXTPROC glad_glVertexAttribL2dvEXT;
12065 #define glVertexAttribL2dvEXT glad_glVertexAttribL2dvEXT
12066 typedef void (APIENTRY PFNGLVERTEXATTRIBL3DVEXTPROC) (GLuint index, const GLdouble *v);
12067 GLAPI PFNGLVERTEXATTRIBL3DVEXTPROC glad_glVertexAttribL3dvEXT;
12068 #define glVertexAttribL3dvEXT glad_glVertexAttribL3dvEXT
12069 typedef void (APIENTRY PFNGLVERTEXATTRIBL4DVEXTPROC) (GLuint index, const GLdouble *v);
12070 GLAPI PFNGLVERTEXATTRIBL4DVEXTPROC glad_glVertexAttribL4dvEXT;
12071 #define glVertexAttribL4dvEXT glad_glVertexAttribL4dvEXT
12072 typedef void (APIENTRY PFNGLVERTEXATTRIBLPOINTEREXTPROC) (GLuint index, GLint size, GLenum type,
GLsizei stride, const void *pointer);
12073 GLAPI PFNGLVERTEXATTRIBLPOINTEREXTPROC glad_glVertexAttribLPointerEXT;
12074 #define glVertexAttribLPointerEXT glad_glVertexAttribLPointerEXT
12075 typedef void (APIENTRY PFNGLGETVERTEXATTRIBLDVEXTPROC) (GLuint index, GLenum pname, GLdouble *params);
12076 GLAPI PFNGLGETVERTEXATTRIBLDVEXTPROC glad_glGetVertexAttribLdvEXT;
12077 #define glGetVertexAttribLdvEXT glad_glGetVertexAttribLdvEXT
12078 #endif
12079 #ifndef GL_EXT_vertex_shader
12080 #define GL_EXT_vertex_shader 1
12081 GLAPI int GLAD_GL_EXT_vertex_shader;
12082 typedef void (APIENTRY PFNGLBEGINVERTEXSHADEREXTPROC) (void);
12083 GLAPI PFNGLBEGINVERTEXSHADEREXTPROC glad_glBeginVertexShaderEXT;
12084 #define glBeginVertexShaderEXT glad_glBeginVertexShaderEXT
12085 typedef void (APIENTRY PFNGLENDVERTEXSHADEREXTPROC) (void);
12086 GLAPI PFNGLENDVERTEXSHADEREXTPROC glad_glEndVertexShaderEXT;
12087 #define glEndVertexShaderEXT glad_glEndVertexShaderEXT
12088 typedef void (APIENTRY PFNGLBINDVERTEXSHADEREXTPROC) (GLuint id);
12089 GLAPI PFNGLBINDVERTEXSHADEREXTPROC glad_glBindVertexShaderEXT;
12090 #define glBindVertexShaderEXT glad_glBindVertexShaderEXT
12091 typedef GLuint (APIENTRY PFNGLGENVERTEXSHADERSEXTPROC) (GLuint range);
12092 GLAPI PFNGLGENVERTEXSHADERSEXTPROC glad_glGenVertexShadersEXT;
12093 #define glGenVertexShadersEXT glad_glGenVertexShadersEXT
12094 typedef void (APIENTRY PFNGLDELETEVERTEXSHADEREXTPROC) (GLuint id);
12095 GLAPI PFNGLDELETEVERTEXSHADEREXTPROC glad_glDeleteVertexShaderEXT;
12096 #define glDeleteVertexShaderEXT glad_glDeleteVertexShaderEXT
12097 typedef void (APIENTRY PFNGLSHADEROP1EXTPROC) (GLenum op, GLuint res, GLuint arg1);
12098 GLAPI PFNGLSHADEROP1EXTPROC glad_glShaderOp1EXT;
12099 #define glShaderOp1EXT glad_glShaderOp1EXT
12100 typedef void (APIENTRY PFNGLSHADEROP2EXTPROC) (GLenum op, GLuint res, GLuint arg1, GLuint arg2);
12101 GLAPI PFNGLSHADEROP2EXTPROC glad_glShaderOp2EXT;
12102 #define glShaderOp2EXT glad_glShaderOp2EXT
12103 typedef void (APIENTRY PFNGLSHADEROP3EXTPROC) (GLenum op, GLuint res, GLuint arg1, GLuint arg2, GLuint
arg3);
12104 GLAPI PFNGLSHADEROP3EXTPROC glad_glShaderOp3EXT;
12105 #define glShaderOp3EXT glad_glShaderOp3EXT
12106 typedef void (APIENTRY PFNGLSWIZZLEEXTPROC) (GLuint res, GLuint in, GLenum outX, GLenum outY, GLenum
outZ, GLenum outW);
12107 GLAPI PFNGLSWIZZLEEXTPROC glad_glSwizzleEXT;
12108 #define glSwizzleEXT glad_glSwizzleEXT
12109 typedef void (APIENTRY PFNGLWRITEMASKEXTPROC) (GLuint res, GLuint in, GLenum outX, GLenum outY, GLenum
outZ, GLenum outW);
12110 GLAPI PFNGLWRITEMASKEXTPROC glad_glWriteMaskEXT;
12111 #define glWriteMaskEXT glad_glWriteMaskEXT
12112 typedef void (APIENTRY PFNGLINSERTCOMPONENTEXTPROC) (GLuint res, GLuint src, GLuint num);
12113 GLAPI PFNGLINSERTCOMPONENTEXTPROC glad_glInsertComponentEXT;
12114 #define glInsertComponentEXT glad_glInsertComponentEXT
12115 typedef void (APIENTRY PFNGLEXTRACTCOMPONENTEXTPROC) (GLuint res, GLuint src, GLuint num);
12116 GLAPI PFNGLEXTRACTCOMPONENTEXTPROC glad_glExtractComponentEXT;
12117 #define glExtractComponentEXT glad_glExtractComponentEXT
12118 typedef GLuint (APIENTRY PFNGLGENSYMBOLSEXTPROC) (GLenum datatype, GLenum storagetype, GLenum range,
GLuint components);
12119 GLAPI PFNGLGENSYMBOLSEXTPROC glad_glGenSymbolsEXT;
12120 #define glGenSymbolsEXT glad_glGenSymbolsEXT
12121 typedef void (APIENTRY PFNGLSETINVARIANTEXTPROC) (GLuint id, GLenum type, const void *addr);
12122 GLAPI PFNGLSETINVARIANTEXTPROC glad_glSetInvariantEXT;
12123 #define glSetInvariantEXT glad_glSetInvariantEXT
12124 typedef void (APIENTRY PFNGLSETLOCALCONSTANTEXTPROC) (GLuint id, GLenum type, const void *addr);
12125 GLAPI PFNGLSETLOCALCONSTANTEXTPROC glad_glSetLocalConstantEXT;
12126 #define glSetLocalConstantEXT glad_glSetLocalConstantEXT
12127 typedef void (APIENTRY PFNGLVARIANTBVEXTPROC) (GLuint id, const GLbyte *addr);
12128 GLAPI PFNGLVARIANTBVEXTPROC glad_glVariantbvEXT;
12129 #define glVariantbvEXT glad_glVariantbvEXT
12130 typedef void (APIENTRY PFNGLVARIANTSVEXTPROC) (GLuint id, const GLshort *addr);
12131 GLAPI PFNGLVARIANTSVEXTPROC glad_glVariantsvEXT;
12132 #define glVariantsvEXT glad_glVariantsvEXT
12133 typedef void (APIENTRY PFNGLVARIANTIVEXTPROC) (GLuint id, const GLint *addr);

```

```

12134 GLAPI PFNGLVARIANTIVEXTPROC glad_glVariantivEXT;
12135 #define glVariantivEXT glad_glVariantivEXT
12136 typedef void (APIENTRY PFNGLVARIANTFVEXTPROC) (GLuint id, const GLfloat *addr);
12137 GLAPI PFNGLVARIANTFVEXTPROC glad_glVariantfvEXT;
12138 #define glVariantfvEXT glad_glVariantfvEXT
12139 typedef void (APIENTRY PFNGLVARIANTDVEXTPROC) (GLuint id, const GLdouble *addr);
12140 GLAPI PFNGLVARIANTDVEXTPROC glad_glVariantdvEXT;
12141 #define glVariantdvEXT glad_glVariantdvEXT
12142 typedef void (APIENTRY PFNGLVARIANTUBVEXTPROC) (GLuint id, const GLubyte *addr);
12143 GLAPI PFNGLVARIANTUBVEXTPROC glad_glVariantubvEXT;
12144 #define glVariantubvEXT glad_glVariantubvEXT
12145 typedef void (APIENTRY PFNGLVARIANTUSVEXTPROC) (GLuint id, const GLushort *addr);
12146 GLAPI PFNGLVARIANTUSVEXTPROC glad_glVariantusvEXT;
12147 #define glVariantusvEXT glad_glVariantusvEXT
12148 typedef void (APIENTRY PFNGLVARIANTUIVEXTPROC) (GLuint id, const GLuint *addr);
12149 GLAPI PFNGLVARIANTUIVEXTPROC glad_glVariantuivEXT;
12150 #define glVariantuivEXT glad_glVariantuivEXT
12151 typedef void (APIENTRY PFNGLVARIANTPOINTEREXTPROC) (GLuint id, GLenum type, GLuint stride, const void
*addr);
12152 GLAPI PFNGLVARIANTPOINTEREXTPROC glad_glVariantPointerEXT;
12153 #define glVariantPointerEXT glad_glVariantPointerEXT
12154 typedef void (APIENTRY PFNGLENABLEVARIANTCLIENTSTATEEXTPROC) (GLuint id);
12155 GLAPI PFNGLENABLEVARIANTCLIENTSTATEEXTPROC glad_glEnableVariantClientStateEXT;
12156 #define glDisableVariantClientStateEXT glad_glDisableVariantClientStateEXT
12157 typedef void (APIENTRY PFNGLDISABLEVARIANTCLIENTSTATEEXTPROC) (GLuint id);
12158 GLAPI PFNGLDISABLEVARIANTCLIENTSTATEEXTPROC glad_glDisableVariantClientStateEXT;
12159 #define glDisableVariantClientStateEXT glad_glDisableVariantClientStateEXT
12160 typedef GLuint (APIENTRY PFNGLBINDLIGHTPARAMETEREXTPROC) (GLenum light, GLenum value);
12161 GLAPI PFNGLBINDLIGHTPARAMETEREXTPROC glad_glBindLightParameterEXT;
12162 #define glBindLightParameterEXT glad_glBindLightParameterEXT
12163 typedef GLuint (APIENTRY PFNGLBINDMATERIALPARAMETEREXTPROC) (GLenum face, GLenum value);
12164 GLAPI PFNGLBINDMATERIALPARAMETEREXTPROC glad_glBindMaterialParameterEXT;
12165 #define glBindMaterialParameterEXT glad_glBindMaterialParameterEXT
12166 typedef GLuint (APIENTRY PFNGLBINDTEXGENPARAMETEREXTPROC) (GLenum unit, GLenum coord, GLenum value);
12167 GLAPI PFNGLBINDTEXGENPARAMETEREXTPROC glad_glBindTexGenParameterEXT;
12168 #define glBindTexGenParameterEXT glad_glBindTexGenParameterEXT
12169 typedef GLuint (APIENTRY PFNGLBINDTEXTUREUNITPARAMETEREXTPROC) (GLenum unit, GLenum value);
12170 GLAPI PFNGLBINDTEXTUREUNITPARAMETEREXTPROC glad_glBindTextureUnitParameterEXT;
12171 #define glBindTextureUnitParameterEXT glad_glBindTextureUnitParameterEXT
12172 typedef GLuint (APIENTRY PFNGLBINDPARAMETEREXTPROC) (GLenum value);
12173 GLAPI PFNGLBINDPARAMETEREXTPROC glad_glBindParameterEXT;
12174 #define glBindParameterEXT glad_glBindParameterEXT
12175 typedef GLboolean (APIENTRY PFNGLISVARIANTENABLEDEXTPROC) (GLuint id, GLenum cap);
12176 GLAPI PFNGLISVARIANTENABLEDEXTPROC glad_glIsVariantEnabledEXT;
12177 #define glIsVariantEnabledEXT glad_glIsVariantEnabledEXT
12178 typedef void (APIENTRY PFNGLGETVARIANTBOOLEANVEXTPROC) (GLuint id, GLenum value, GLboolean *data);
12179 GLAPI PFNGLGETVARIANTBOOLEANVEXTPROC glad_glGetVariantBooleanvEXT;
12180 #define glGetVariantBooleanvEXT glad_glGetVariantBooleanvEXT
12181 typedef void (APIENTRY PFNGLGETVARIANTINTEGERVEXTPROC) (GLuint id, GLenum value, GLint *data);
12182 GLAPI PFNGLGETVARIANTINTEGERVEXTPROC glad_glGetVariantIntegervEXT;
12183 #define glGetVariantIntegervEXT glad_glGetVariantIntegervEXT
12184 typedef void (APIENTRY PFNGLGETVARIANTFLOATVEXTPROC) (GLuint id, GLenum value, GLfloat *data);
12185 GLAPI PFNGLGETVARIANTFLOATVEXTPROC glad_glGetVariantFloatvEXT;
12186 #define glGetVariantFloatvEXT glad_glGetVariantFloatvEXT
12187 typedef void (APIENTRY PFNGLGETVARIANTPOINTERVEXTPROC) (GLuint id, GLenum value, void **data);
12188 GLAPI PFNGLGETVARIANTPOINTERVEXTPROC glad_glGetVariantPointervEXT;
12189 #define glGetVariantPointervEXT glad_glGetVariantPointervEXT
12190 typedef void (APIENTRY PFNGLGETINVARIANTBOOLEANVEXTPROC) (GLuint id, GLenum value, GLboolean *data);
12191 GLAPI PFNGLGETINVARIANTBOOLEANVEXTPROC glad_glGetInvariantBooleanvEXT;
12192 #define glGetInvariantBooleanvEXT glad_glGetInvariantBooleanvEXT
12193 typedef void (APIENTRY PFNGLGETINVARIANTINTEGERVEXTPROC) (GLuint id, GLenum value, GLint *data);
12194 GLAPI PFNGLGETINVARIANTINTEGERVEXTPROC glad_glGetInvariantIntegervEXT;
12195 #define glGetInvariantIntegervEXT glad_glGetInvariantIntegervEXT
12196 typedef void (APIENTRY PFNGLGETINVARIANTFLOATVEXTPROC) (GLuint id, GLenum value, GLfloat *data);
12197 GLAPI PFNGLGETINVARIANTFLOATVEXTPROC glad_glGetInvariantFloatvEXT;
12198 #define glGetInvariantFloatvEXT glad_glGetInvariantFloatvEXT
12199 typedef void (APIENTRY PFNGLGETLOCALCONSTANTBOOLEANVEXTPROC) (GLuint id, GLenum value, GLboolean
*data);
12200 GLAPI PFNGLGETLOCALCONSTANTBOOLEANVEXTPROC glad_glGetLocalConstantBooleanvEXT;
12201 #define glGetLocalConstantBooleanvEXT glad_glGetLocalConstantBooleanvEXT
12202 typedef void (APIENTRY PFNGLGETLOCALCONSTANTINTEGERVEXTPROC) (GLuint id, GLenum value, GLint *data);
12203 GLAPI PFNGLGETLOCALCONSTANTINTEGERVEXTPROC glad_glGetLocalConstantIntegervEXT;
12204 #define glGetLocalConstantIntegervEXT glad_glGetLocalConstantIntegervEXT
12205 typedef void (APIENTRY PFNGLGETLOCALCONSTANTFLOATVEXTPROC) (GLuint id, GLenum value, GLfloat *data);
12206 GLAPI PFNGLGETLOCALCONSTANTFLOATVEXTPROC glad_glGetLocalConstantFloatvEXT;
12207 #define glGetLocalConstantFloatvEXT glad_glGetLocalConstantFloatvEXT
12208 #endif
12209 #ifndef GL_EXT_vertex_weighting
12210 #define GL_EXT_vertex_weighting 1
12211 GLAPI int GLAD_GL_EXT_vertex_weighting;
12212 typedef void (APIENTRY PFNGLVERTEXWEIGHTFEXTPROC) (GLfloat weight);
12213 GLAPI PFNGLVERTEXWEIGHTFEXTPROC glad_glVertexWeightfEXT;
12214 #define glVertexWeightfEXT glad_glVertexWeightfEXT
12215 typedef void (APIENTRY PFNGLVERTEXWEIGHTFVEXTPROC) (const GLfloat *weight);
12216 GLAPI PFNGLVERTEXWEIGHTFVEXTPROC glad_glVertexWeightfvEXT;
12217 #define glVertexWeightfvEXT glad_glVertexWeightfvEXT
12218 typedef void (APIENTRY PFNGLVERTEXWEIGHTPOINTEREXTPROC) (GLint size, GLenum type, GLsizei stride,

```

```

    const void *pointer);
12219 GLAPI PFNGLVERTEXWEIGHTPOINTEREXTPROC glad_glVertexWeightPointerEXT;
12220 #define glVertexWeightPointerEXT glad_glVertexWeightPointerEXT
12221 #endif
12222 #ifndef GL_EXT_win32_keyed_mutex
12223 #define GL_EXT_win32_keyed_mutex 1
12224 GLAPI int GLAD_GL_EXT_win32_keyed_mutex;
12225 typedef GLboolean (APIENTRY PFNGLACQUIREKEYEDMUTEXWIN32EXTPROC) (GLuint memory, GLuint64 key, GLuint
    timeout);
12226 GLAPI PFNGLACQUIREKEYEDMUTEXWIN32EXTPROC glad_glAcquireKeyedMutexWin32EXT;
12227 #define glAcquireKeyedMutexWin32EXT glad_glAcquireKeyedMutexWin32EXT
12228 typedef GLboolean (APIENTRY PFNGLRELEASEKEYEDMUTEXWIN32EXTPROC) (GLuint memory, GLuint64 key);
12229 GLAPI PFNGLRELEASEKEYEDMUTEXWIN32EXTPROC glad_glReleaseKeyedMutexWin32EXT;
12230 #define glReleaseKeyedMutexWin32EXT glad_glReleaseKeyedMutexWin32EXT
12231 #endif
12232 #ifndef GL_EXT_window_rectangles
12233 #define GL_EXT_window_rectangles 1
12234 GLAPI int GLAD_GL_EXT_window_rectangles;
12235 typedef void (APIENTRY PFNGLWINDOWRECTANGLESEXTPROC) (GLenum mode, GLsizei count, const GLint *box);
12236 GLAPI PFNGLWINDOWRECTANGLESEXTPROC glad_glWindowRectanglesEXT;
12237 #define glWindowRectanglesEXT glad_glWindowRectanglesEXT
12238 #endif
12239 #ifndef GL_EXT_x11_sync_object
12240 #define GL_EXT_x11_sync_object 1
12241 GLAPI int GLAD_GL_EXT_x11_sync_object;
12242 typedef GLsync (APIENTRY PFNGLIMPORTSYNCEXTPROC) (GLenum external_sync_type, GLintptr external_sync,
    GLbitfield flags);
12243 GLAPI PFNGLIMPORTSYNCEXTPROC glad_glImportSyncEXT;
12244 #define glImportSyncEXT glad_glImportSyncEXT
12245 #endif
12246 #ifndef GL_GREMEDY_frame_terminator
12247 #define GL_GREMEDY_frame_terminator 1
12248 GLAPI int GLAD_GL_GREMEDY_frame_terminator;
12249 typedef void (APIENTRY PFNGLFRAMETERMINATORGREMEDYPROC) (void);
12250 GLAPI PFNGLFRAMETERMINATORGREMEDYPROC glad_glFrameTerminatorGREMEDY;
12251 #define glFrameTerminatorGREMEDY glad_glFrameTerminatorGREMEDY
12252 #endif
12253 #ifndef GL_GREMEDY_string_marker
12254 #define GL_GREMEDY_string_marker 1
12255 GLAPI int GLAD_GL_GREMEDY_string_marker;
12256 typedef void (APIENTRY PFNGLSTRINGMARKERGREMEDYPROC) (GLsizei len, const void *string);
12257 GLAPI PFNGLSTRINGMARKERGREMEDYPROC glad_glStringMarkerGREMEDY;
12258 #define glStringMarkerGREMEDY glad_glStringMarkerGREMEDY
12259 #endif
12260 #ifndef GL_HP_convolution_border_modes
12261 #define GL_HP_convolution_border_modes 1
12262 GLAPI int GLAD_GL_HP_convolution_border_modes;
12263 #endif
12264 #ifndef GL_HP_image_transform
12265 #define GL_HP_image_transform 1
12266 GLAPI int GLAD_GL_HP_image_transform;
12267 typedef void (APIENTRY PFNGLIMAGETRANSFORMPARAMETERIHPPROC) (GLenum target, GLenum pname, GLint
    param);
12268 GLAPI PFNGLIMAGETRANSFORMPARAMETERIHPPROC glad_glImageTransformParameteriHP;
12269 #define glImageTransformParameteriHP glad_glImageTransformParameteriHP
12270 typedef void (APIENTRY PFNGLIMAGETRANSFORMPARAMETERFHPPROC) (GLenum target, GLenum pname, GLfloat
    param);
12271 GLAPI PFNGLIMAGETRANSFORMPARAMETERFHPPROC glad_glImageTransformParameterfHP;
12272 #define glImageTransformParameterfHP glad_glImageTransformParameterfHP
12273 typedef void (APIENTRY PFNGLIMAGETRANSFORMPARAMETERIVHPPROC) (GLenum target, GLenum pname, const GLint
    *params);
12274 GLAPI PFNGLIMAGETRANSFORMPARAMETERIVHPPROC glad_glImageTransformParameterivHP;
12275 #define glImageTransformParameterivHP glad_glImageTransformParameterivHP
12276 typedef void (APIENTRY PFNGLIMAGETRANSFORMPARAMETERFVHPPROC) (GLenum target, GLenum pname, const
    GLfloat *params);
12277 GLAPI PFNGLIMAGETRANSFORMPARAMETERFVHPPROC glad_glImageTransformParameterfvHP;
12278 #define glImageTransformParameterfvHP glad_glImageTransformParameterfvHP
12279 typedef void (APIENTRY PFNGLGETIMAGETRANSFORMPARAMETERIVHPPROC) (GLenum target, GLenum pname, GLint
    *params);
12280 GLAPI PFNGLGETIMAGETRANSFORMPARAMETERIVHPPROC glad_glGetImageTransformParameterivHP;
12281 #define glGetImageTransformParameterivHP glad_glGetImageTransformParameterivHP
12282 typedef void (APIENTRY PFNGLGETIMAGETRANSFORMPARAMETERFVHPPROC) (GLenum target, GLenum pname, GLfloat
    *params);
12283 GLAPI PFNGLGETIMAGETRANSFORMPARAMETERFVHPPROC glad_glGetImageTransformParameterfvHP;
12284 #define glGetImageTransformParameterfvHP glad_glGetImageTransformParameterfvHP
12285 #endif
12286 #ifndef GL_HP_occlusion_test
12287 #define GL_HP_occlusion_test 1
12288 GLAPI int GLAD_GL_HP_occlusion_test;
12289 #endif
12290 #ifndef GL_HP_texture_lighting
12291 #define GL_HP_texture_lighting 1
12292 GLAPI int GLAD_GL_HP_texture_lighting;
12293 #endif
12294 #ifndef GL_IBM_cull_vertex
12295 #define GL_IBM_cull_vertex 1
12296 GLAPI int GLAD_GL_IBM_cull_vertex;

```

```

12297 #endif
12298 #ifndef GL_IBM_multimode_draw_arrays
12299 #define GL_IBM_multimode_draw_arrays 1
12300 GLAPI int GLAD_GL_IBM_multimode_draw_arrays;
12301 typedef void (APIENTRY PFNGLMULTIMODEDRAWARRAYSIBMPROC)(const GLenum *mode, const GLint *first, const
GLsizei *count, GLsizei primcount, GLint modestride);
12302 GLAPI PFNGLMULTIMODEDRAWARRAYSIBMPROC glad_glMultiModeDrawArraysIBM;
12303 #define glMultiModeDrawArraysIBM glad_glMultiModeDrawArraysIBM
12304 typedef void (APIENTRY PFNGLMULTIMODEDRAWELEMENTSIBMPROC)(const GLenum *mode, const GLsizei *count,
GLenum type, const void *const*indices, GLsizei primcount, GLint modestride);
12305 GLAPI PFNGLMULTIMODEDRAWELEMENTSIBMPROC glad_glMultiModeDrawElementsIBM;
12306 #define glMultiModeDrawElementsIBM glad_glMultiModeDrawElementsIBM
12307 #endif
12308 #ifndef GL_IBM_rasterpos_clip
12309 #define GL_IBM_rasterpos_clip 1
12310 GLAPI int GLAD_GL_IBM_rasterpos_clip;
12311 #endif
12312 #ifndef GL_IBM_static_data
12313 #define GL_IBM_static_data 1
12314 GLAPI int GLAD_GL_IBM_static_data;
12315 typedef void (APIENTRY PFNGLFLUSHSTATICDATAIBMPROC)(GLenum target);
12316 GLAPI PFNGLFLUSHSTATICDATAIBMPROC glad_glFlushStaticDataIBM;
12317 #define glFlushStaticDataIBM glad_glFlushStaticDataIBM
12318 #endif
12319 #ifndef GL_IBM_texture_mirrored_repeat
12320 #define GL_IBM_texture_mirrored_repeat 1
12321 GLAPI int GLAD_GL_IBM_texture_mirrored_repeat;
12322 #endif
12323 #ifndef GL_IBM_vertex_array_lists
12324 #define GL_IBM_vertex_array_lists 1
12325 GLAPI int GLAD_GL_IBM_vertex_array_lists;
12326 typedef void (APIENTRY PFNGLCOLORPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride, const
void **pointer, GLint ptrstride);
12327 GLAPI PFNGLCOLORPOINTERLISTIBMPROC glad_glColorPointerListIBM;
12328 #define glColorPointerListIBM glad_glColorPointerListIBM
12329 typedef void (APIENTRY PFNGLSECONDARYCOLORPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride,
const void **pointer, GLint ptrstride);
12330 GLAPI PFNGLSECONDARYCOLORPOINTERLISTIBMPROC glad_glSecondaryColorPointerListIBM;
12331 #define glSecondaryColorPointerListIBM glad_glSecondaryColorPointerListIBM
12332 typedef void (APIENTRY PFNGLEDGEFLAGPOINTERLISTIBMPROC)(GLint stride, const GLboolean **pointer,
GLint ptrstride);
12333 GLAPI PFNGLEDGEFLAGPOINTERLISTIBMPROC glad_glEdgeFlagPointerListIBM;
12334 #define glEdgeFlagPointerListIBM glad_glEdgeFlagPointerListIBM
12335 typedef void (APIENTRY PFNGLFOGCOORDPOINTERLISTIBMPROC)(GLenum type, GLint stride, const void
**pointer, GLint ptrstride);
12336 GLAPI PFNGLFOGCOORDPOINTERLISTIBMPROC glad_glFogCoordPointerListIBM;
12337 #define glFogCoordPointerListIBM glad_glFogCoordPointerListIBM
12338 typedef void (APIENTRY PFNGLINDEXPOINTERLISTIBMPROC)(GLenum type, GLint stride, const void **pointer,
GLint ptrstride);
12339 GLAPI PFNGLINDEXPOINTERLISTIBMPROC glad_glIndexPointerListIBM;
12340 #define glIndexPointerListIBM glad_glIndexPointerListIBM
12341 typedef void (APIENTRY PFNGLNORMALPOINTERLISTIBMPROC)(GLenum type, GLint stride, const void
**pointer, GLint ptrstride);
12342 GLAPI PFNGLNORMALPOINTERLISTIBMPROC glad_glNormalPointerListIBM;
12343 #define glNormalPointerListIBM glad_glNormalPointerListIBM
12344 typedef void (APIENTRY PFNGLTEXCOORDPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride, const
void **pointer, GLint ptrstride);
12345 GLAPI PFNGLTEXCOORDPOINTERLISTIBMPROC glad_glTexCoordPointerListIBM;
12346 #define glTexCoordPointerListIBM glad_glTexCoordPointerListIBM
12347 typedef void (APIENTRY PFNGLVERTEXPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride, const
void **pointer, GLint ptrstride);
12348 GLAPI PFNGLVERTEXPOINTERLISTIBMPROC glad_glVertexPointerListIBM;
12349 #define glVertexPointerListIBM glad_glVertexPointerListIBM
12350 #endif
12351 #ifndef GL_INGR_blend_func_separate
12352 #define GL_INGR_blend_func_separate 1
12353 GLAPI int GLAD_GL_INGR_blend_func_separate;
12354 typedef void (APIENTRY PFNGLBLENDFUNCSEPARATEINGRPROC)(GLenum sfactorRGB, GLenum dfactorRGB, GLenum
sfactorAlpha, GLenum dfactorAlpha);
12355 GLAPI PFNGLBLENDFUNCSEPARATEINGRPROC glad_glBlendFuncSeparateINGR;
12356 #define glBlendFuncSeparateINGR glad_glBlendFuncSeparateINGR
12357 #endif
12358 #ifndef GL_INGR_color_clamp
12359 #define GL_INGR_color_clamp 1
12360 GLAPI int GLAD_GL_INGR_color_clamp;
12361 #endif
12362 #ifndef GL_INGR_interlace_read
12363 #define GL_INGR_interlace_read 1
12364 GLAPI int GLAD_GL_INGR_interlace_read;
12365 #endif
12366 #ifndef GL_INTEL_blackhole_render
12367 #define GL_INTEL_blackhole_render 1
12368 GLAPI int GLAD_GL_INTEL_blackhole_render;
12369 #endif
12370 #ifndef GL_INTEL_conservative_rasterization
12371 #define GL_INTEL_conservative_rasterization 1
12372 GLAPI int GLAD_GL_INTEL_conservative_rasterization;

```

```

12373 #endif
12374 #ifndef GL_INTEL_fragment_shader_ordering
12375 #define GL_INTEL_fragment_shader_ordering 1
12376 GLAPI int GLAD_GL_INTEL_fragment_shader_ordering;
12377 #endif
12378 #ifndef GL_INTEL_framebuffer_CMAA
12379 #define GL_INTEL_framebuffer_CMAA 1
12380 GLAPI int GLAD_GL_INTEL_framebuffer_CMAA;
12381 typedef void (APIENTRY PFNGLAPPLYFRAMEBUFFERATTACHMENTCMAAINTELPROC)(void);
12382 GLAPI PFNGLAPPLYFRAMEBUFFERATTACHMENTCMAAINTELPROC glad_glApplyFramebufferAttachmentCMAAINTEL;
12383 #define glApplyFramebufferAttachmentCMAAINTEL glad_glApplyFramebufferAttachmentCMAAINTEL
12384 #endif
12385 #ifndef GL_INTEL_map_texture
12386 #define GL_INTEL_map_texture 1
12387 GLAPI int GLAD_GL_INTEL_map_texture;
12388 typedef void (APIENTRY PFNGLSYNCTEXTUREINTELPROC)(GLuint texture);
12389 GLAPI PFNGLSYNCTEXTUREINTELPROC glad_glSyncTextureINTEL;
12390 #define glSyncTextureINTEL glad_glSyncTextureINTEL
12391 typedef void (APIENTRY PFNGLUNMAPTEXTURE2DINTELPROC)(GLuint texture, GLint level);
12392 GLAPI PFNGLUNMAPTEXTURE2DINTELPROC glad_glUnmapTexture2DINTEL;
12393 #define glUnmapTexture2DINTEL glad_glUnmapTexture2DINTEL
12394 typedef void * (APIENTRY PFNGLMAPTEXTURE2DINTELPROC)(GLuint texture, GLint level, GLbitfield access,
12395   GLint *stride, GLenum *layout);
12396 GLAPI PFNGLMAPTEXTURE2DINTELPROC glad_glMapTexture2DINTEL;
12397 #define glMapTexture2DINTEL glad_glMapTexture2DINTEL
12398 #endif
12399 #ifndef GL_INTEL_parallel_arrays
12400 #define GL_INTEL_parallel_arrays 1
12401 GLAPI int GLAD_GL_INTEL_parallel_arrays;
12402 typedef void (APIENTRY PFNGLVERTEXPOINTERVINTELPROC)(GLint size, GLenum type, const void **pointer);
12403 GLAPI PFNGLVERTEXPOINTERVINTELPROC glad_glVertexPointervINTEL;
12404 #define glVertexPointervINTEL glad_glVertexPointervINTEL
12405 typedef void (APIENTRY PFNGLNORMALPOINTERVINTELPROC)(GLenum type, const void **pointer);
12406 GLAPI PFNGLNORMALPOINTERVINTELPROC glad_glNormalPointervINTEL;
12407 #define glNormalPointervINTEL glad_glNormalPointervINTEL
12408 typedef void (APIENTRY PFNGLCOLORPOINTERVINTELPROC)(GLint size, GLenum type, const void **pointer);
12409 GLAPI PFNGLCOLORPOINTERVINTELPROC glad_glColorPointervINTEL;
12410 #define glColorPointervINTEL glad_glColorPointervINTEL
12411 typedef void (APIENTRY PFNGLTEXCOORDPOINTERVINTELPROC)(GLint size, GLenum type, const void
12412   **pointer);
12413 GLAPI PFNGLTEXCOORDPOINTERVINTELPROC glad_glTexCoordPointervINTEL;
12414 #define glTexCoordPointervINTEL glad_glTexCoordPointervINTEL
12415 #endif
12416 #ifndef GL_INTEL_performance_query
12417 #define GL_INTEL_performance_query 1
12418 GLAPI int GLAD_GL_INTEL_performance_query;
12419 typedef void (APIENTRY PFNGLBEGINPERFQUERYINTELPROC)(GLuint queryHandle);
12420 GLAPI PFNGLBEGINPERFQUERYINTELPROC glad_glBeginPerfQueryINTEL;
12421 #define glBeginPerfQueryINTEL glad_glBeginPerfQueryINTEL
12422 typedef void (APIENTRY PFNGLCREATEPERFQUERYINTELPROC)(GLuint queryId, GLuint *queryHandle);
12423 GLAPI PFNGLCREATEPERFQUERYINTELPROC glad_glCreatePerfQueryINTEL;
12424 #define glCreatePerfQueryINTEL glad_glCreatePerfQueryINTEL
12425 typedef void (APIENTRY PFNGLDELETEPERFQUERYINTELPROC)(GLuint queryHandle);
12426 GLAPI PFNGLDELETEPERFQUERYINTELPROC glad_glDeletePerfQueryINTEL;
12427 #define glDeletePerfQueryINTEL glad_glDeletePerfQueryINTEL
12428 typedef void (APIENTRY PFNGLENDPERFQUERYINTELPROC)(GLuint queryHandle);
12429 GLAPI PFNGLENDPERFQUERYINTELPROC glad_glEndPerfQueryINTEL;
12430 #define glEndPerfQueryINTEL glad_glEndPerfQueryINTEL
12431 typedef void (APIENTRY PFNGLGETFIRSTPERFQUERYIDINTELPROC)(GLuint *queryId);
12432 GLAPI PFNGLGETFIRSTPERFQUERYIDINTELPROC glad_glGetFirstPerfQueryIdINTEL;
12433 #define glGetFirstPerfQueryIdINTEL glad_glGetFirstPerfQueryIdINTEL
12434 typedef void (APIENTRY PFNGLGETNEXTPERFQUERYIDINTELPROC)(GLuint queryId, GLuint *nextQueryId);
12435 GLAPI PFNGLGETNEXTPERFQUERYIDINTELPROC glad_glGetNextPerfQueryIdINTEL;
12436 #define glGetNextPerfQueryIdINTEL glad_glGetNextPerfQueryIdINTEL
12437 typedef void (APIENTRY PFNGLGETPERFCOUNTERINFOINTELPROC)(GLuint queryId, GLuint counterId, GLuint
12438   counterNameLength, GLchar *counterName, GLuint counterDescLength, GLchar *counterDesc, GLuint
12439   *counterOffset, GLuint *counterDataSize, GLuint *counterTypeEnum, GLuint *counterDataTypeEnum,
12440   GLuint64 *rowCounterMaxValue);
12441 GLAPI PFNGLGETPERFCOUNTERINFOINTELPROC glad_glGetPerfCounterInfoINTEL;
12442 #define glGetPerfCounterInfoINTEL glad_glGetPerfCounterInfoINTEL
12443 typedef void (APIENTRY PFNGLGETPERFQUERYDATAINTELPROC)(GLuint queryHandle, GLuint flags, GLsizei
12444   dataSize, void *data, GLuint *bytesWritten);
12445 GLAPI PFNGLGETPERFQUERYDATAINTELPROC glad_glGetPerfQueryDataINTEL;
12446 #define glGetPerfQueryDataINTEL glad_glGetPerfQueryDataINTEL
12447 typedef void (APIENTRY PFNGLGETPERFQUERYIDBYNAMEINTELPROC)(GLchar *queryName, GLuint *queryId);
12448 GLAPI PFNGLGETPERFQUERYIDBYNAMEINTELPROC glad_glGetPerfQueryIdByNameINTEL;
12449 #define glGetPerfQueryIdByNameINTEL glad_glGetPerfQueryIdByNameINTEL
12450 typedef void (APIENTRY PFNGLGETPERFQUERYINFOINTELPROC)(GLuint queryId, GLuint queryNameLength, GLchar
12451   *queryName, GLuint *dataSize, GLuint *noCounters, GLuint *noInstances, GLuint *capsMask);
12452 GLAPI PFNGLGETPERFQUERYINFOINTELPROC glad_glGetPerfQueryInfoINTEL;
12453 #define glGetPerfQueryInfoINTEL glad_glGetPerfQueryInfoINTEL
12454 #endif
12455 #ifndef GL_KHR_blend_equation_advanced
12456 #define GL_KHR_blend_equation_advanced 1
12457 GLAPI int GLAD_GL_KHR_blend_equation_advanced;
12458 typedef void (APIENTRY PFNGLBLENDBARRIERKHRPROC)(void);
12459 GLAPI PFNGLBLENDBARRIERKHRPROC glad_glBlendBarrierKHR;

```



```

12453 #define glBlendBarrierKHR glad_glBlendBarrierKHR
12454 #endif
12455 #ifndef GL_KHR_blend_equation_advanced_coherent
12456 #define GL_KHR_blend_equation_advanced_coherent 1
12457 GLAPI int GLAD_GL_KHR_blend_equation_advanced_coherent;
12458 #endif
12459 #ifndef GL_KHR_context_flush_control
12460 #define GL_KHR_context_flush_control 1
12461 GLAPI int GLAD_GL_KHR_context_flush_control;
12462 #endif
12463 #ifndef GL_KHR_debug
12464 #define GL_KHR_debug 1
12465 GLAPI int GLAD_GL_KHR_debug;
12466 typedef void (APIENTRY PFNGLDEBUGMESSAGECONTROLPROC) (GLenum source, GLenum type, GLenum severity,
GLsizei count, const GLuint *ids, GLboolean enabled);
12467 GLAPI PFNGLDEBUGMESSAGECONTROLPROC glad_glDebugMessageControl;
12468 #define glDebugMessageControl glad_glDebugMessageControl
12469 typedef void (APIENTRY PFNGLDEBUGMESSAGEINSERTPROC) (GLenum source, GLenum type, GLuint id, GLenum
severity, GLsizei length, const GLchar *buf);
12470 GLAPI PFNGLDEBUGMESSAGEINSERTPROC glad_glDebugMessageInsert;
12471 #define glDebugMessageInsert glad_glDebugMessageInsert
12472 typedef void (APIENTRY PFNGLDEBUGMESSAGECALLBACKPROC) (GLDEBUGPROC callback, const void *userParam);
12473 GLAPI PFNGLDEBUGMESSAGECALLBACKPROC glad_glDebugMessageCallback;
12474 #define glDebugMessageCallback glad_glDebugMessageCallback
12475 typedef GLuint (APIENTRY PFNGLGETDEBUGMESSAGELOGPROC) (GLuint count, GLsizei bufSize, GLenum *sources,
GLenum *types, GLuint *ids, GLenum *severities, GLsizei *lengths, GLchar *messageLog);
12476 GLAPI PFNGLGETDEBUGMESSAGELOGPROC glad_glGetDebugMessageLog;
12477 #define glGetDebugMessageLog glad_glGetDebugMessageLog
12478 typedef void (APIENTRY PFNGLPUSHDEBUGGROUPPROC) (GLenum source, GLuint id, GLsizei length, const
GLchar *message);
12479 GLAPI PFNGLPUSHDEBUGGROUPPROC glad_glPushDebugGroup;
12480 #define glPushDebugGroup glad_glPushDebugGroup
12481 typedef void (APIENTRY PFNGLPOPDEBUGGROUPPROC) (void);
12482 GLAPI PFNGLPOPDEBUGGROUPPROC glad_glPopDebugGroup;
12483 #define glPopDebugGroup glad_glPopDebugGroup
12484 typedef void (APIENTRY PFNGLOBJECTLABELPROC) (GLenum identifier, GLuint name, GLsizei length, const
GLchar *label);
12485 GLAPI PFNGLOBJECTLABELPROC glad_glObjectLabel;
12486 #define glObjectLabel glad_glObjectLabel
12487 typedef void (APIENTRY PFNGLGETOBJECTLABELPROC) (GLenum identifier, GLuint name, GLsizei bufSize,
GLsizei *length, GLchar *label);
12488 GLAPI PFNGLGETOBJECTLABELPROC glad_glGetObjectLabel;
12489 #define glGetObjectLabel glad_glGetObjectLabel
12490 typedef void (APIENTRY PFNGLOBJECTPTRLABELPROC) (const void *ptr, GLsizei length, const GLchar
*label);
12491 GLAPI PFNGLOBJECTPTRLABELPROC glad_glObjectPtrLabel;
12492 #define glObjectPtrLabel glad_glObjectPtrLabel
12493 typedef void (APIENTRY PFNGLGETOBJECTPTRLABELPROC) (const void *ptr, GLsizei bufSize, GLsizei *length,
GLchar *label);
12494 GLAPI PFNGLGETOBJECTPTRLABELPROC glad_glGetObjectPtrLabel;
12495 #define glGetObjectPtrLabel glad_glGetObjectPtrLabel
12496 typedef void (APIENTRY PFNGLGETPOINTERVPROC) (GLenum pname, void **params);
12497 GLAPI PFNGLGETPOINTERVPROC glad_glGetPointerv;
12498 #define glGetPointerv glad_glGetPointerv
12499 typedef void (APIENTRY PFNGLDEBUGMESSAGECONTROLKHRPROC) (GLenum source, GLenum type, GLenum severity,
GLsizei count, const GLuint *ids, GLboolean enabled);
12500 GLAPI PFNGLDEBUGMESSAGECONTROLKHRPROC glad_glDebugMessageControlKHR;
12501 #define glDebugMessageControlKHR glad_glDebugMessageControlKHR
12502 typedef void (APIENTRY PFNGLDEBUGMESSAGEINSERTKHRPROC) (GLenum source, GLenum type, GLuint id, GLenum
severity, GLsizei length, const GLchar *buf);
12503 GLAPI PFNGLDEBUGMESSAGEINSERTKHRPROC glad_glDebugMessageInsertKHR;
12504 #define glDebugMessageInsertKHR glad_glDebugMessageInsertKHR
12505 typedef void (APIENTRY PFNGLDEBUGMESSAGECALLBACKKHRPROC) (GLDEBUGPROCKHR callback, const void
*userParam);
12506 GLAPI PFNGLDEBUGMESSAGECALLBACKKHRPROC glad_glDebugMessageCallbackKHR;
12507 #define glDebugMessageCallbackKHR glad_glDebugMessageCallbackKHR
12508 typedef GLuint (APIENTRY PFNGLGETDEBUGMESSAGELOGKHRPROC) (GLuint count, GLsizei bufSize, GLenum
*sources, GLenum *types, GLuint *ids, GLenum *severities, GLsizei *lengths, GLchar *messageLog);
12509 GLAPI PFNGLGETDEBUGMESSAGELOGKHRPROC glad_glGetDebugMessageLogKHR;
12510 #define glGetDebugMessageLogKHR glad_glGetDebugMessageLogKHR
12511 typedef void (APIENTRY PFNGLPUSHDEBUGGROUPKHRPROC) (GLenum source, GLuint id, GLsizei length, const
GLchar *message);
12512 GLAPI PFNGLPUSHDEBUGGROUPKHRPROC glad_glPushDebugGroupKHR;
12513 #define glPushDebugGroupKHR glad_glPushDebugGroupKHR
12514 typedef void (APIENTRY PFNGLPOPDEBUGGROUPKHRPROC) (void);
12515 GLAPI PFNGLPOPDEBUGGROUPKHRPROC glad_glPopDebugGroupKHR;
12516 #define glPopDebugGroupKHR glad_glPopDebugGroupKHR
12517 typedef void (APIENTRY PFNGLOBJECTLABELKHRPROC) (GLenum identifier, GLuint name, GLsizei length, const
GLchar *label);
12518 GLAPI PFNGLOBJECTLABELKHRPROC glad_glObjectLabelKHR;
12519 #define glObjectLabelKHR glad_glObjectLabelKHR
12520 typedef void (APIENTRY PFNGLGETOBJECTLABELKHRPROC) (GLenum identifier, GLuint name, GLsizei bufSize,
GLsizei *length, GLchar *label);
12521 GLAPI PFNGLGETOBJECTLABELKHRPROC glad_glGetObjectLabelKHR;
12522 #define glGetObjectLabelKHR glad_glGetObjectLabelKHR
12523 typedef void (APIENTRY PFNGLOBJECTPTRLABELKHRPROC) (const void *ptr, GLsizei length, const GLchar
*label);

```

```

12524 GLAPI PFNGLOBJECTPTRLABELKHRPROC glad_glObjectPtrLabelKHR;
12525 #define glObjectPtrLabelKHR glad_glObjectPtrLabelKHR
12526 typedef void (APIENTRY PFNGLGETOBJECTPTRLABELKHRPROC) (const void *ptr, GLsizei bufSize, GLsizei
    *length, GLchar *label);
12527 GLAPI PFNGLGETOBJECTPTRLABELKHRPROC glad_glGetObjectPtrLabelKHR;
12528 #define glGetObjectPtrLabelKHR glad_glGetObjectPtrLabelKHR
12529 typedef void (APIENTRY PFNGLGETPOINTERVKHRPROC) (GLenum pname, void **params);
12530 GLAPI PFNGLGETPOINTERVKHRPROC glad_glGetPointervKHR;
12531 #define glGetPointervKHR glad_glGetPointervKHR
12532 #endif
12533 #ifndef GL_KHR_no_error
12534 #define GL_KHR_no_error 1
12535 GLAPI int GLAD_GL_KHR_no_error;
12536 #endif
12537 #ifndef GL_KHR_parallel_shader_compile
12538 #define GL_KHR_parallel_shader_compile 1
12539 GLAPI int GLAD_GL_KHR_parallel_shader_compile;
12540 typedef void (APIENTRY PFNGLMAXSHADERCOMPILERTHREADSKHRPROC) (GLuint count);
12541 GLAPI PFNGLMAXSHADERCOMPILERTHREADSKHRPROC glad_glMaxShaderCompilerThreadsKHR;
12542 #define glMaxShaderCompilerThreadsKHR glad_glMaxShaderCompilerThreadsKHR
12543 #endif
12544 #ifndef GL_KHR_robust_buffer_access_behavior
12545 #define GL_KHR_robust_buffer_access_behavior 1
12546 GLAPI int GLAD_GL_KHR_robust_buffer_access_behavior;
12547 #endif
12548 #ifndef GL_KHR_robustness
12549 #define GL_KHR_robustness 1
12550 GLAPI int GLAD_GL_KHR_robustness;
12551 typedef GLenum (APIENTRY PFNGLGETGRAPHICSRESETSTATUSPROC) (void);
12552 GLAPI PFNGLGETGRAPHICSRESETSTATUSPROC glad_glGetGraphicsResetStatus;
12553 #define glGetGraphicsResetStatus glad_glGetGraphicsResetStatus
12554 typedef void (APIENTRY PFNGLREADNPIXELSPROC) (GLint x, GLint y, GLsizei width, GLsizei height, GLenum
    format, GLenum type, GLsizei bufSize, void *data);
12555 GLAPI PFNGLREADNPIXELSPROC glad_glReadnPixels;
12556 #define glReadnPixels glad_glReadnPixels
12557 typedef void (APIENTRY PFNGLGETNUNIFORMFVPROC) (GLuint program, GLint location, GLsizei bufSize,
    GLfloat *params);
12558 GLAPI PFNGLGETNUNIFORMFVPROC glad_glGetnUniformfv;
12559 #define glGetnUniformfv glad_glGetnUniformfv
12560 typedef void (APIENTRY PFNGLGETNUNIFORMIVPROC) (GLuint program, GLint location, GLsizei bufSize, GLint
    *params);
12561 GLAPI PFNGLGETNUNIFORMIVPROC glad_glGetnUniformiv;
12562 #define glGetnUniformiv glad_glGetnUniformiv
12563 typedef void (APIENTRY PFNGLGETNUNIFORMUIVPROC) (GLuint program, GLint location, GLsizei bufSize,
    GLuint *params);
12564 GLAPI PFNGLGETNUNIFORMUIVPROC glad_glGetnUniformuiv;
12565 #define glGetnUniformuiv glad_glGetnUniformuiv
12566 typedef void (APIENTRY PFNGLGETGRAPHICSRESETSTATUSKHRPROC) (void);
12567 GLAPI PFNGLGETGRAPHICSRESETSTATUSKHRPROC glad_glGetGraphicsResetStatusKHR;
12568 #define glGetGraphicsResetStatusKHR glad_glGetGraphicsResetStatusKHR
12569 typedef void (APIENTRY PFNGLREADNPIXELSKHRPROC) (GLint x, GLint y, GLsizei width, GLsizei height,
    GLenum format, GLenum type, GLsizei bufSize, void *data);
12570 GLAPI PFNGLREADNPIXELSKHRPROC glad_glReadnPixelsKHR;
12571 #define glReadnPixelsKHR glad_glReadnPixelsKHR
12572 typedef void (APIENTRY PFNGLGETNUNIFORMFVKHRPROC) (GLuint program, GLint location, GLsizei bufSize,
    GLfloat *params);
12573 GLAPI PFNGLGETNUNIFORMFVKHRPROC glad_glGetnUniformfvKHR;
12574 #define glGetnUniformfvKHR glad_glGetnUniformfvKHR
12575 typedef void (APIENTRY PFNGLGETNUNIFORMIVKHRPROC) (GLuint program, GLint location, GLsizei bufSize,
    GLint *params);
12576 GLAPI PFNGLGETNUNIFORMIVKHRPROC glad_glGetnUniformivKHR;
12577 #define glGetnUniformivKHR glad_glGetnUniformivKHR
12578 typedef void (APIENTRY PFNGLGETNUNIFORMUIVKHRPROC) (GLuint program, GLint location, GLsizei bufSize,
    GLuint *params);
12579 GLAPI PFNGLGETNUNIFORMUIVKHRPROC glad_glGetnUniformuivKHR;
12580 #define glGetnUniformuivKHR glad_glGetnUniformuivKHR
12581 #endif
12582 #ifndef GL_KHR_shader_subgroup
12583 #define GL_KHR_shader_subgroup 1
12584 GLAPI int GLAD_GL_KHR_shader_subgroup;
12585 #endif
12586 #ifndef GL_KHR_texture_compression_astc_hdr
12587 #define GL_KHR_texture_compression_astc_hdr 1
12588 GLAPI int GLAD_GL_KHR_texture_compression_astc_hdr;
12589 #endif
12590 #ifndef GL_KHR_texture_compression_astc_ldr
12591 #define GL_KHR_texture_compression_astc_ldr 1
12592 GLAPI int GLAD_GL_KHR_texture_compression_astc_ldr;
12593 #endif
12594 #ifndef GL_KHR_texture_compression_astc_sliced_3d
12595 #define GL_KHR_texture_compression_astc_sliced_3d 1
12596 GLAPI int GLAD_GL_KHR_texture_compression_astc_sliced_3d;
12597 #endif
12598 #ifndef GL_MESAX_texture_stack
12599 #define GL_MESAX_texture_stack 1
12600 GLAPI int GLAD_GL_MESAX_texture_stack;
12601 #endif

```



```

12602 #ifndef GL_MESA_framebuffer_flip_x
12603 #define GL_MESA_framebuffer_flip_x 1
12604 GLAPI int GLAD_GL_MESA_framebuffer_flip_x;
12605 #endif
12606 #ifndef GL_MESA_framebuffer_flip_y
12607 #define GL_MESA_framebuffer_flip_y 1
12608 GLAPI int GLAD_GL_MESA_framebuffer_flip_y;
12609 typedef void (APIENTRY PFNGLFRAMEBUFFERPARAMETERIMESAPROC)(GLenum target, GLenum pname, GLint param);
12610 GLAPI PFNGLFRAMEBUFFERPARAMETERIMESAPROC glad_glFramebufferParameteriMESA;
12611 #define glFramebufferParameteriMESA glad_glFramebufferParameteriMESA
12612 typedef void (APIENTRY PFNGLGETFRAMEBUFFERPARAMETERIVMESAPROC)(GLenum target, GLenum pname, GLint
    *params);
12613 GLAPI PFNGLGETFRAMEBUFFERPARAMETERIVMESAPROC glad_glGetFramebufferParameterivMESA;
12614 #define glGetFramebufferParameterivMESA glad_glGetFramebufferParameterivMESA
12615 #endif
12616 #ifndef GL_MESA_framebuffer_swap_xy
12617 #define GL_MESA_framebuffer_swap_xy 1
12618 GLAPI int GLAD_GL_MESA_framebuffer_swap_xy;
12619 #endif
12620 #ifndef GL_MESA_pack_invert
12621 #define GL_MESA_pack_invert 1
12622 GLAPI int GLAD_GL_MESA_pack_invert;
12623 #endif
12624 #ifndef GL_MESA_program_binary_formats
12625 #define GL_MESA_program_binary_formats 1
12626 GLAPI int GLAD_GL_MESA_program_binary_formats;
12627 #endif
12628 #ifndef GL_MESA_resize_buffers
12629 #define GL_MESA_resize_buffers 1
12630 GLAPI int GLAD_GL_MESA_resize_buffers;
12631 typedef void (APIENTRY PFNGLRESIZEBUFFERSMESAPROC)(void);
12632 GLAPI PFNGLRESIZEBUFFERSMESAPROC glad_glResizeBuffersMESA;
12633 #define glResizeBuffersMESA glad_glResizeBuffersMESA
12634 #endif
12635 #ifndef GL_MESA_shader_integer_functions
12636 #define GL_MESA_shader_integer_functions 1
12637 GLAPI int GLAD_GL_MESA_shader_integer_functions;
12638 #endif
12639 #ifndef GL_MESA_tile_raster_order
12640 #define GL_MESA_tile_raster_order 1
12641 GLAPI int GLAD_GL_MESA_tile_raster_order;
12642 #endif
12643 #ifndef GL_MESA_window_pos
12644 #define GL_MESA_window_pos 1
12645 GLAPI int GLAD_GL_MESA_window_pos;
12646 typedef void (APIENTRY PFNGLWINDOWPOS2DMESAPROC)(GLdouble x, GLdouble y);
12647 GLAPI PFNGLWINDOWPOS2DMESAPROC glad_glWindowPos2dMESA;
12648 #define glWindowPos2dMESA glad_glWindowPos2dMESA
12649 typedef void (APIENTRY PFNGLWINDOWPOS2DVESAPROC)(const GLdouble *v);
12650 GLAPI PFNGLWINDOWPOS2DVESAPROC glad_glWindowPos2dvMESA;
12651 #define glWindowPos2dvMESA glad_glWindowPos2dvMESA
12652 typedef void (APIENTRY PFNGLWINDOWPOS2FMESAPROC)(GLfloat x, GLfloat y);
12653 GLAPI PFNGLWINDOWPOS2FMESAPROC glad_glWindowPos2fMESA;
12654 #define glWindowPos2fMESA glad_glWindowPos2fMESA
12655 typedef void (APIENTRY PFNGLWINDOWPOS2FVMESAPROC)(const GLfloat *v);
12656 GLAPI PFNGLWINDOWPOS2FVMESAPROC glad_glWindowPos2fvMESA;
12657 #define glWindowPos2fvMESA glad_glWindowPos2fvMESA
12658 typedef void (APIENTRY PFNGLWINDOWPOS2IMESAPROC)(GLint x, GLint y);
12659 GLAPI PFNGLWINDOWPOS2IMESAPROC glad_glWindowPos2iMESA;
12660 #define glWindowPos2iMESA glad_glWindowPos2iMESA
12661 typedef void (APIENTRY PFNGLWINDOWPOS2IVMESAPROC)(const GLint *v);
12662 GLAPI PFNGLWINDOWPOS2IVMESAPROC glad_glWindowPos2ivMESA;
12663 #define glWindowPos2ivMESA glad_glWindowPos2ivMESA
12664 typedef void (APIENTRY PFNGLWINDOWPOS2SMESAPROC)(GLshort x, GLshort y);
12665 GLAPI PFNGLWINDOWPOS2SMESAPROC glad_glWindowPos2sMESA;
12666 #define glWindowPos2sMESA glad_glWindowPos2sMESA
12667 typedef void (APIENTRY PFNGLWINDOWPOS2SVMESAPROC)(const GLshort *v);
12668 GLAPI PFNGLWINDOWPOS2SVMESAPROC glad_glWindowPos2svMESA;
12669 #define glWindowPos2svMESA glad_glWindowPos2svMESA
12670 typedef void (APIENTRY PFNGLWINDOWPOS3DMESAPROC)(GLdouble x, GLdouble y, GLdouble z);
12671 GLAPI PFNGLWINDOWPOS3DMESAPROC glad_glWindowPos3dMESA;
12672 #define glWindowPos3dMESA glad_glWindowPos3dMESA
12673 typedef void (APIENTRY PFNGLWINDOWPOS3DVESAPROC)(const GLdouble *v);
12674 GLAPI PFNGLWINDOWPOS3DVESAPROC glad_glWindowPos3dvMESA;
12675 #define glWindowPos3dvMESA glad_glWindowPos3dvMESA
12676 typedef void (APIENTRY PFNGLWINDOWPOS3FMESAPROC)(GLfloat x, GLfloat y, GLfloat z);
12677 GLAPI PFNGLWINDOWPOS3FMESAPROC glad_glWindowPos3fMESA;
12678 #define glWindowPos3fMESA glad_glWindowPos3fMESA
12679 typedef void (APIENTRY PFNGLWINDOWPOS3FVMESAPROC)(const GLfloat *v);
12680 GLAPI PFNGLWINDOWPOS3FVMESAPROC glad_glWindowPos3fvMESA;
12681 #define glWindowPos3fvMESA glad_glWindowPos3fvMESA
12682 typedef void (APIENTRY PFNGLWINDOWPOS3IMESAPROC)(GLint x, GLint y, GLint z);
12683 GLAPI PFNGLWINDOWPOS3IMESAPROC glad_glWindowPos3iMESA;
12684 #define glWindowPos3iMESA glad_glWindowPos3iMESA
12685 typedef void (APIENTRY PFNGLWINDOWPOS3IVMESAPROC)(const GLint *v);
12686 GLAPI PFNGLWINDOWPOS3IVMESAPROC glad_glWindowPos3ivMESA;
12687 #define glWindowPos3ivMESA glad_glWindowPos3ivMESA

```

```

12688 typedef void (APIENTRY PFNGLWINDOWPOS3SMESAPROC) (GLshort x, GLshort y, GLshort z);
12689 GLAPI PFNGLWINDOWPOS3SMESAPROC glad_glWindowPos3sMESA;
12690 #define glWindowPos3sMESA glad_glWindowPos3sMESA
12691 typedef void (APIENTRY PFNGLWINDOWPOS3SVMESAPROC) (const GLshort *v);
12692 GLAPI PFNGLWINDOWPOS3SVMESAPROC glad_glWindowPos3svMESA;
12693 #define glWindowPos3svMESA glad_glWindowPos3svMESA
12694 typedef void (APIENTRY PFNGLWINDOWPOS4DMESAPROC) (GLdouble x, GLdouble y, GLdouble z, GLdouble w);
12695 GLAPI PFNGLWINDOWPOS4DMESAPROC glad_glWindowPos4dMESA;
12696 #define glWindowPos4dMESA glad_glWindowPos4dMESA
12697 typedef void (APIENTRY PFNGLWINDOWPOS4DVMESAPROC) (const GLdouble *v);
12698 GLAPI PFNGLWINDOWPOS4DVMESAPROC glad_glWindowPos4dvMESA;
12699 #define glWindowPos4dvMESA glad_glWindowPos4dvMESA
12700 typedef void (APIENTRY PFNGLWINDOWPOS4FMESAPROC) (GLfloat x, GLfloat y, GLfloat z, GLfloat w);
12701 GLAPI PFNGLWINDOWPOS4FMESAPROC glad_glWindowPos4fMESA;
12702 #define glWindowPos4fMESA glad_glWindowPos4fMESA
12703 typedef void (APIENTRY PFNGLWINDOWPOS4FVMESAPROC) (const GLfloat *v);
12704 GLAPI PFNGLWINDOWPOS4FVMESAPROC glad_glWindowPos4fvMESA;
12705 #define glWindowPos4fvMESA glad_glWindowPos4fvMESA
12706 typedef void (APIENTRY PFNGLWINDOWPOS4IMESAPROC) (GLint x, GLint y, GLint z, GLint w);
12707 GLAPI PFNGLWINDOWPOS4IMESAPROC glad_glWindowPos4iMESA;
12708 #define glWindowPos4iMESA glad_glWindowPos4iMESA
12709 typedef void (APIENTRY PFNGLWINDOWPOS4IVMESAPROC) (const GLint *v);
12710 GLAPI PFNGLWINDOWPOS4IVMESAPROC glad_glWindowPos4ivMESA;
12711 #define glWindowPos4ivMESA glad_glWindowPos4ivMESA
12712 typedef void (APIENTRY PFNGLWINDOWPOS4SMESAPROC) (GLshort x, GLshort y, GLshort z, GLshort w);
12713 GLAPI PFNGLWINDOWPOS4SMESAPROC glad_glWindowPos4sMESA;
12714 #define glWindowPos4sMESA glad_glWindowPos4sMESA
12715 typedef void (APIENTRY PFNGLWINDOWPOS4SVMESAPROC) (const GLshort *v);
12716 GLAPI PFNGLWINDOWPOS4SVMESAPROC glad_glWindowPos4svMESA;
12717 #define glWindowPos4svMESA glad_glWindowPos4svMESA
12718 #endif
12719 #ifndef GL_MESA_ycbcr_texture
12720 #define GL_MESA_ycbcr_texture 1
12721 GLAPI int GLAD_GL_MESA_ycbcr_texture;
12722 #endif
12723 #ifndef GL_NVX_blend_equation_advanced_multi_draw_buffers
12724 #define GL_NVX_blend_equation_advanced_multi_draw_buffers 1
12725 GLAPI int GLAD_GL_NVX_blend_equation_advanced_multi_draw_buffers;
12726 #endif
12727 #ifndef GL_NVX_conditional_render
12728 #define GL_NVX_conditional_render 1
12729 GLAPI int GLAD_GL_NVX_conditional_render;
12730 typedef void (APIENTRY PFNGLBEGINCONDITIONALRENDERNVXPROC) (GLuint id);
12731 GLAPI PFNGLBEGINCONDITIONALRENDERNVXPROC glad_glBeginConditionalRenderNVX;
12732 #define glBeginConditionalRenderNVX glad_glBeginConditionalRenderNVX
12733 typedef void (APIENTRY PFNGLENDCONDITIONALRENDERNVXPROC) (void);
12734 GLAPI PFNGLENDCONDITIONALRENDERNVXPROC glad_glEndConditionalRenderNVX;
12735 #define glEndConditionalRenderNVX glad_glEndConditionalRenderNVX
12736 #endif
12737 #ifndef GL_NVX_gpu_memory_info
12738 #define GL_NVX_gpu_memory_info 1
12739 GLAPI int GLAD_GL_NVX_gpu_memory_info;
12740 #endif
12741 #ifndef GL_NVX_gpu_multicast2
12742 #define GL_NVX_gpu_multicast2 1
12743 GLAPI int GLAD_GL_NVX_gpu_multicast2;
12744 typedef void (APIENTRY PFNGLUPLOADGPUMASKNVXPROC) (GLbitfield mask);
12745 GLAPI PFNGLUPLOADGPUMASKNVXPROC glad_glUploadGpuMaskNVX;
12746 #define glUploadGpuMaskNVX glad_glUploadGpuMaskNVX
12747 typedef void (APIENTRY PFNGLMULTICASTVIEWPORTARRAYVNVXPROC) (GLuint gpu, GLuint first, GLsizei count,
const GLfloat *v);
12748 GLAPI PFNGLMULTICASTVIEWPORTARRAYVNVXPROC glad_glMulticastViewportArrayvNVX;
12749 #define glMulticastViewportArrayvNVX glad_glMulticastViewportArrayvNVX
12750 typedef void (APIENTRY PFNGLMULTICASTVIEWPORTPOSITIONWSCALENVXPROC) (GLuint gpu, GLuint index, GLfloat
xcoeff, GLfloat ycoeff);
12751 GLAPI PFNGLMULTICASTVIEWPORTPOSITIONWSCALENVXPROC glad_glMulticastViewportPositionWScaleNVX;
12752 #define glMulticastViewportPositionWScaleNVX glad_glMulticastViewportPositionWScaleNVX
12753 typedef void (APIENTRY PFNGLMULTICASTSCISSORARRAYVNVXPROC) (GLuint gpu, GLuint first, GLsizei count,
const GLint *v);
12754 GLAPI PFNGLMULTICASTSCISSORARRAYVNVXPROC glad_glMulticastScissorArrayvNVX;
12755 #define glMulticastScissorArrayvNVX glad_glMulticastScissorArrayvNVX
12756 typedef GLuint (APIENTRY PFNGLASYNCCOPYBUFFERSUBDATANVXPROC) (GLsizei waitSemaphoreCount, const GLuint
*waitSemaphoreArray, const GLuint64 *fenceValueArray, GLuint readGpu, GLbitfield writeGpuMask, GLuint
readBuffer, GLuint writeBuffer, GLintptr readOffset, GLintptr writeOffset, GLsizeiptr size, GLsizei
signalSemaphoreCount, const GLuint *signalSemaphoreArray, const GLuint64 *signalValueArray);
12757 GLAPI PFNGLASYNCCOPYBUFFERSUBDATANVXPROC glad_glAsyncCopyBufferSubDataNVX;
12758 #define glAsyncCopyBufferSubDataNVX glad_glAsyncCopyBufferSubDataNVX
12759 typedef void (APIENTRY PFNGLASYNCCOPYIMAGESUBDATANVXPROC) (GLsizei waitSemaphoreCount, const GLuint
*waitSemaphoreArray, const GLuint64 *waitValueArray, GLuint srcGpu, GLbitfield dstGpuMask, GLuint
srcName, GLenum srcTarget, GLint srcLevel, GLint srcX, GLint srcY, GLint srcZ, GLuint dstName, GLenum
dstTarget, GLint dstLevel, GLint dstX, GLint dstY, GLint dstZ, GLsizei srcWidth, GLsizei srcHeight,
GLsizei srcDepth, GLsizei signalSemaphoreCount, const GLuint *signalSemaphoreArray, const GLuint64
*signalValueArray);
12760 GLAPI PFNGLASYNCCOPYIMAGESUBDATANVXPROC glad_glAsyncCopyImageSubDataNVX;
12761 #define glAsyncCopyImageSubDataNVX glad_glAsyncCopyImageSubDataNVX
12762 #endif
12763 #ifndef GL_NVX_linked_gpu_multicast

```

```

12764 #define GL_NVX_linked_gpu_multicast 1
12765 GLAPI int GLAD_GL_NVX_linked_gpu_multicast;
12766 typedef void (APIENTRY PFNGLLGPUNAMEDBUFFERSUBDATANVXPROC) (GLbitfield gpuMask, GLuint buffer,
12767 GLintptr offset, GLsizei ptr size, const void *data);
12767 GLAPI PFNGLLGPUNAMEDBUFFERSUBDATANVXPROC glad_gllGPUNamedBufferSubDataNVX;
12768 #define gllGPUNamedBufferSubDataNVX glad_gllGPUNamedBufferSubDataNVX
12769 typedef void (APIENTRY PFNGLLGPUCOPYIMAGESUBDATANVXPROC) (GLuint sourceGpu, GLbitfield
12770 destinationGpuMask, GLuint srcName, GLenum srcTarget, GLint srcLevel, GLint srcX, GLint srcY, GLint
12771 srcZ, GLuint dstName, GLenum dstTarget, GLint dstLevel, GLint dstX, GLint dstY, GLint dstZ, GLsizei
12772 width, GLsizei height, GLsizei depth);
12770 GLAPI PFNGLLGPUCOPYIMAGESUBDATANVXPROC glad_gllGPUCopyImageSubDataNVX;
12771 #define gllGPUCopyImageSubDataNVX glad_gllGPUCopyImageSubDataNVX
12772 typedef void (APIENTRY PFNGLLGPUINTERLOCKNVXPROC) (void);
12773 GLAPI PFNGLLGPUINTERLOCKNVXPROC glad_gllGPUInterlockNVX;
12774 #define gllGPUInterlockNVX glad_gllGPUInterlockNVX
12775 #endif
12776 #ifndef GL_NVX_progress_fence
12777 #define GL_NVX_progress_fence 1
12778 GLAPI int GLAD_GL_NVX_progress_fence;
12779 typedef GLuint (APIENTRY PFNGLCREATEPROGRESSFENCENVXPROC) (void);
12780 GLAPI PFNGLCREATEPROGRESSFENCENVXPROC glad_glCreateProgressFenceNVX;
12781 #define glCreateProgressFenceNVX glad_glCreateProgressFenceNVX
12782 typedef void (APIENTRY PFNGLSIGNALSEMAPHOREUI64NVXPROC) (GLuint signalGpu, GLsizei fenceObjectCount,
12783 const GLuint *semaphoreArray, const GLuint64 *fenceValueArray);
12783 GLAPI PFNGLSIGNALSEMAPHOREUI64NVXPROC glad_glSignalSemaphoreui64NVX;
12784 #define glSignalSemaphoreui64NVX glad_glSignalSemaphoreui64NVX
12785 typedef void (APIENTRY PFNGLWAITSEMAPHOREUI64NVXPROC) (GLuint waitGpu, GLsizei fenceObjectCount, const
12786 GLuint *semaphoreArray, const GLuint64 *fenceValueArray);
12786 GLAPI PFNGLWAITSEMAPHOREUI64NVXPROC glad_glWaitSemaphoreui64NVX;
12787 #define glWaitSemaphoreui64NVX glad_glWaitSemaphoreui64NVX
12788 typedef void (APIENTRY PFNGLCLIENTWAITSEMAPHOREUI64NVXPROC) (GLsizei fenceObjectCount, const GLuint
12789 *semaphoreArray, const GLuint64 *fenceValueArray);
12789 GLAPI PFNGLCLIENTWAITSEMAPHOREUI64NVXPROC glad_glClientWaitSemaphoreui64NVX;
12790 #define glClientWaitSemaphoreui64NVX glad_glClientWaitSemaphoreui64NVX
12791 #endif
12792 #ifndef GL_NV_alpha_to_coverage_dither_control
12793 #define GL_NV_alpha_to_coverage_dither_control 1
12794 GLAPI int GLAD_GL_NV_alpha_to_coverage_dither_control;
12795 typedef void (APIENTRY PFNGLALPHATOCOVERAGEDITHERCONTROLNVPROC) (GLenum mode);
12796 GLAPI PFNGLALPHATOCOVERAGEDITHERCONTROLNVPROC glad_glAlphaToCoverageDitherControlNV;
12797 #define glAlphaToCoverageDitherControlNV glad_glAlphaToCoverageDitherControlNV
12798 #endif
12799 #ifndef GL_NV_bindless_multi_draw_indirect
12800 #define GL_NV_bindless_multi_draw_indirect 1
12801 GLAPI int GLAD_GL_NV_bindless_multi_draw_indirect;
12802 typedef void (APIENTRY PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSNVPROC) (GLenum mode, const void *indirect,
12803 GLsizei drawCount, GLsizei stride, GLint vertexBufferCount);
12803 GLAPI PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSNVPROC glad_glMultiDrawArraysIndirectBindlessNV;
12804 #define glMultiDrawArraysIndirectBindlessNV glad_glMultiDrawArraysIndirectBindlessNV
12805 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSNVPROC) (GLenum mode, GLenum type, const
12806 void *indirect, GLsizei drawCount, GLsizei stride, GLint vertexBufferCount);
12806 GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSNVPROC glad_glMultiDrawElementsIndirectBindlessNV;
12807 #define glMultiDrawElementsIndirectBindlessNV glad_glMultiDrawElementsIndirectBindlessNV
12808 #endif
12809 #ifndef GL_NV_bindless_multi_draw_indirect_count
12810 #define GL_NV_bindless_multi_draw_indirect_count 1
12811 GLAPI int GLAD_GL_NV_bindless_multi_draw_indirect_count;
12812 typedef void (APIENTRY PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSCOUNTNVPROC) (GLenum mode, const void
12813 *indirect, GLsizei drawCount, GLsizei maxDrawCount, GLsizei stride, GLint vertexBufferCount);
12813 GLAPI PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSCOUNTNVPROC glad_glMultiDrawArraysIndirectBindlessCountNV;
12814 #define glMultiDrawArraysIndirectBindlessCountNV glad_glMultiDrawArraysIndirectBindlessCountNV
12815 typedef void (APIENTRY PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSCOUNTNVPROC) (GLenum mode, GLenum type,
12816 const void *indirect, GLsizei drawCount, GLsizei maxDrawCount, GLsizei stride, GLint
12817 vertexBufferCount);
12816 GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSCOUNTNVPROC
12817 glad_glMultiDrawElementsIndirectBindlessCountNV;
12817 #define glMultiDrawElementsIndirectBindlessCountNV glad_glMultiDrawElementsIndirectBindlessCountNV
12818 #endif
12819 #ifndef GL_NV_bindless_texture
12820 #define GL_NV_bindless_texture 1
12821 GLAPI int GLAD_GL_NV_bindless_texture;
12822 typedef GLuint64 (APIENTRY PFNGLGETTEXTUREHANDLENVPROC) (GLuint texture);
12823 GLAPI PFNGLGETTEXTUREHANDLENVPROC glad_glGetTextureHandleNV;
12824 #define glGetTextureHandleNV glad_glGetTextureHandleNV
12825 typedef GLuint64 (APIENTRY PFNGLGETTEXTURESAMPLERHANDLENVPROC) (GLuint texture, GLuint sampler);
12826 GLAPI PFNGLGETTEXTURESAMPLERHANDLENVPROC glad_glGetTextureSamplerHandleNV;
12827 #define glGetTextureSamplerHandleNV glad_glGetTextureSamplerHandleNV
12828 typedef void (APIENTRY PFNGLMAKETEXTUREHANDLERESIDENTNVPROC) (GLuint64 handle);
12829 GLAPI PFNGLMAKETEXTUREHANDLERESIDENTNVPROC glad_glMakeTextureHandleResidentNV;
12830 #define glMakeTextureHandleResidentNV glad_glMakeTextureHandleResidentNV
12831 typedef void (APIENTRY PFNGLMAKETEXTUREHANDLENONRESIDENTNVPROC) (GLuint64 handle);
12832 GLAPI PFNGLMAKETEXTUREHANDLENONRESIDENTNVPROC glad_glMakeTextureHandleNonResidentNV;
12833 #define glMakeTextureHandleNonResidentNV glad_glMakeTextureHandleNonResidentNV
12834 typedef GLuint64 (APIENTRY PFNGLGETIMAGEHANDLENVPROC) (GLuint texture, GLint level, GLboolean layered,
12835 GLint layer, GLenum format);
12835 GLAPI PFNGLGETIMAGEHANDLENVPROC glad_glGetImageHandleNV;
12836 #define glGetImageHandleNV glad_glGetImageHandleNV

```

```

12837 typedef void (APIENTRY PFNGLMAKEIMAGEHANDLERESIDENTNVPROC) (GLuint64 handle, GLenum access);
12838 GLAPI PFNGLMAKEIMAGEHANDLERESIDENTNVPROC glad_glMakeImageHandleResidentNV;
12839 #define glMakeImageHandleResidentNV glad_glMakeImageHandleResidentNV
12840 typedef void (APIENTRY PFNGLMAKEIMAGEHANDLENONRESIDENTNVPROC) (GLuint64 handle);
12841 GLAPI PFNGLMAKEIMAGEHANDLENONRESIDENTNVPROC glad_glMakeImageHandleNonResidentNV;
12842 #define glMakeImageHandleNonResidentNV glad_glMakeImageHandleNonResidentNV
12843 typedef void (APIENTRY PFNGLUNIFORMHANDLEUI64NVPROC) (GLint location, GLuint64 value);
12844 GLAPI PFNGLUNIFORMHANDLEUI64NVPROC glad_glUniformHandleui64NV;
12845 #define glUniformHandleui64NV glad_glUniformHandleui64NV
12846 typedef void (APIENTRY PFNGLUNIFORMHANDLEUI64VNVPROC) (GLint location, GLsizei count, const GLuint64
    *value);
12847 GLAPI PFNGLUNIFORMHANDLEUI64VNVPROC glad_glUniformHandleui64vNV;
12848 #define glUniformHandleui64vNV glad_glUniformHandleui64vNV
12849 typedef void (APIENTRY PFNGLPROGRAMUNIFORMHANDLEUI64NVPROC) (GLuint program, GLint location, GLuint64
    value);
12850 GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64NVPROC glad_glProgramUniformHandleui64NV;
12851 #define glProgramUniformHandleui64NV glad_glProgramUniformHandleui64NV
12852 typedef void (APIENTRY PFNGLPROGRAMUNIFORMHANDLEUI64VNVPROC) (GLuint program, GLint location, GLsizei
    count, const GLuint64 *values);
12853 GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64VNVPROC glad_glProgramUniformHandleui64vNV;
12854 #define glProgramUniformHandleui64vNV glad_glProgramUniformHandleui64vNV
12855 typedef GLboolean (APIENTRY PFNGLISTEXTUREHANDLERESIDENTNVPROC) (GLuint64 handle);
12856 GLAPI PFNGLISTEXTUREHANDLERESIDENTNVPROC glad_glIsTextureHandleResidentNV;
12857 #define glIsTextureHandleResidentNV glad_glIsTextureHandleResidentNV
12858 typedef GLboolean (APIENTRY PFNGLISIMAGEHANDLERESIDENTNVPROC) (GLuint64 handle);
12859 GLAPI PFNGLISIMAGEHANDLERESIDENTNVPROC glad_glIsImageHandleResidentNV;
12860 #define glIsImageHandleResidentNV glad_glIsImageHandleResidentNV
12861 #endif
12862 #ifndef GL_NV_blend_equation_advanced
12863 #define GL_NV_blend_equation_advanced 1
12864 GLAPI int GLAD_GL_NV_blend_equation_advanced;
12865 typedef void (APIENTRY PFNGLBLENDPARAMETERINVPROC) (GLenum pname, GLint value);
12866 GLAPI PFNGLBLENDPARAMETERINVPROC glad_glBlendParameteriNV;
12867 #define glBlendParameteriNV glad_glBlendParameteriNV
12868 typedef void (APIENTRY PFNGLBLENDBARRIERNVPROC) (void);
12869 GLAPI PFNGLBLENDBARRIERNVPROC glad_glBlendBarrierNV;
12870 #define glBlendBarrierNV glad_glBlendBarrierNV
12871 #endif
12872 #ifndef GL_NV_blend_equation_advanced_coherent
12873 #define GL_NV_blend_equation_advanced_coherent 1
12874 GLAPI int GLAD_GL_NV_blend_equation_advanced_coherent;
12875 #endif
12876 #ifndef GL_NV_blend_minmax_factor
12877 #define GL_NV_blend_minmax_factor 1
12878 GLAPI int GLAD_GL_NV_blend_minmax_factor;
12879 #endif
12880 #ifndef GL_NV_blend_square
12881 #define GL_NV_blend_square 1
12882 GLAPI int GLAD_GL_NV_blend_square;
12883 #endif
12884 #ifndef GL_NV_clip_space_w_scaling
12885 #define GL_NV_clip_space_w_scaling 1
12886 GLAPI int GLAD_GL_NV_clip_space_w_scaling;
12887 typedef void (APIENTRY PFNGLVIEWPORTPOSITIONWSCALENVPROC) (GLuint index, GLfloat xcoeff, GLfloat
    ycoeff);
12888 GLAPI PFNGLVIEWPORTPOSITIONWSCALENVPROC glad_glViewportPositionWScaleNV;
12889 #define glViewportPositionWScaleNV glad_glViewportPositionWScaleNV
12890 #endif
12891 #ifndef GL_NV_command_list
12892 #define GL_NV_command_list 1
12893 GLAPI int GLAD_GL_NV_command_list;
12894 typedef void (APIENTRY PFNGLCREATESTATESNVPROC) (GLsizei n, GLuint *states);
12895 GLAPI PFNGLCREATESTATESNVPROC glad_glCreateStatesNV;
12896 #define glCreateStatesNV glad_glCreateStatesNV
12897 typedef void (APIENTRY PFNGLDELETESTATESNVPROC) (GLsizei n, const GLuint *states);
12898 GLAPI PFNGLDELETESTATESNVPROC glad_glDeleteStatesNV;
12899 #define glDeleteStatesNV glad_glDeleteStatesNV
12900 typedef GLboolean (APIENTRY PFNGLISSTATENVPROC) (GLuint state);
12901 GLAPI PFNGLISSTATENVPROC glad_glIsStateNV;
12902 #define glIsStateNV glad_glIsStateNV
12903 typedef void (APIENTRY PFNGLSTATECAPTURENVPROC) (GLuint state, GLenum mode);
12904 GLAPI PFNGLSTATECAPTURENVPROC glad_glStateCaptureNV;
12905 #define glStateCaptureNV glad_glStateCaptureNV
12906 typedef GLuint (APIENTRY PFNGLGETCOMMANDHEADERNVPROC) (GLenum tokenID, GLuint size);
12907 GLAPI PFNGLGETCOMMANDHEADERNVPROC glad_glGetCommandHeaderNV;
12908 #define glGetCommandHeaderNV glad_glGetCommandHeaderNV
12909 typedef GLushort (APIENTRY PFNGLGETSTAGEINDEXNVPROC) (GLenum shadertype);
12910 GLAPI PFNGLGETSTAGEINDEXNVPROC glad_glGetStageIndexNV;
12911 #define glGetStageIndexNV glad_glGetStageIndexNV
12912 typedef void (APIENTRY PFNGLDRAWCOMMANDSNVPROC) (GLenum primitiveMode, GLuint buffer, const GLintptr
    *indirects, const GLsizei *sizes, GLuint count);
12913 GLAPI PFNGLDRAWCOMMANDSNVPROC glad_glDrawCommandsNV;
12914 #define glDrawCommandsNV glad_glDrawCommandsNV
12915 typedef void (APIENTRY PFNGLDRAWCOMMANDSADDRESSNVPROC) (GLenum primitiveMode, const GLuint64
    *indirects, const GLsizei *sizes, GLuint count);
12916 GLAPI PFNGLDRAWCOMMANDSADDRESSNVPROC glad_glDrawCommandsAddressNV;
12917 #define glDrawCommandsAddressNV glad_glDrawCommandsAddressNV

```

```

12918 typedef void (APIENTRY PFNGLDRAWCOMMANDSSTATESNVPROC)(GLuint buffer, const GLintptr *indirects, const
GLsizei *sizes, const GLuint *states, const GLuint *fbos, GLuint count);
12919 GLAPI PFNGLDRAWCOMMANDSSTATESNVPROC glad_glDrawCommandsStatesNV;
12920 #define glDrawCommandsStatesNV glad_glDrawCommandsStatesNV
12921 typedef void (APIENTRY PFNGLDRAWCOMMANDSSTATESADDRESSNVPROC)(const GLuint64 *indirects, const GLsizei
*sizes, const GLuint *states, const GLuint *fbos, GLuint count);
12922 GLAPI PFNGLDRAWCOMMANDSSTATESADDRESSNVPROC glad_glDrawCommandsStatesAddressNV;
12923 #define glDrawCommandsStatesAddressNV glad_glDrawCommandsStatesAddressNV
12924 typedef void (APIENTRY PFNGLCREATECOMMANDLISTSNVPROC)(GLsizei n, GLuint *lists);
12925 GLAPI PFNGLCREATECOMMANDLISTSNVPROC glad_glCreateCommandListsNV;
12926 #define glCreateCommandListsNV glad_glCreateCommandListsNV
12927 typedef void (APIENTRY PFNGLDELETECOMMANDLISTSNVPROC)(GLsizei n, const GLuint *lists);
12928 GLAPI PFNGLDELETECOMMANDLISTSNVPROC glad_glDeleteCommandListsNV;
12929 #define glDeleteCommandListsNV glad_glDeleteCommandListsNV
12930 typedef GLboolean (APIENTRY PFNGLISCOMMANDLISTNVPROC)(GLuint list);
12931 GLAPI PFNGLISCOMMANDLISTNVPROC glad_glIsCommandListNV;
12932 #define glIsCommandListNV glad_glIsCommandListNV
12933 typedef void (APIENTRY PFNGLLISTDRAWCOMMANDSSTATESCLIENTNVPROC)(GLuint list, GLuint segment, const
void **indirects, const GLsizei *sizes, const GLuint *states, const GLuint *fbos, GLuint count);
12934 GLAPI PFNGLLISTDRAWCOMMANDSSTATESCLIENTNVPROC glad_glListDrawCommandsStatesClientNV;
12935 #define glListDrawCommandsStatesClientNV glad_glListDrawCommandsStatesClientNV
12936 typedef void (APIENTRY PFNGLCOMMANDLISTSEGMENTSNVPROC)(GLuint list, GLuint segments);
12937 GLAPI PFNGLCOMMANDLISTSEGMENTSNVPROC glad_glCommandListSegmentsNV;
12938 #define glCommandListSegmentsNV glad_glCommandListSegmentsNV
12939 typedef void (APIENTRY PFNGLCOMPILECOMMANDLISTNVPROC)(GLuint list);
12940 GLAPI PFNGLCOMPILECOMMANDLISTNVPROC glad_glCompileCommandListNV;
12941 #define glCompileCommandListNV glad_glCompileCommandListNV
12942 typedef void (APIENTRY PFNGLCALLCOMMANDLISTNVPROC)(GLuint list);
12943 GLAPI PFNGLCALLCOMMANDLISTNVPROC glad_glCallCommandListNV;
12944 #define glCallCommandListNV glad_glCallCommandListNV
12945 #endif
12946 #ifndef GL_NV_compute_program5
12947 #define GL_NV_compute_program5 1
12948 GLAPI int GLAD_GL_NV_compute_program5;
12949 #endif
12950 #ifndef GL_NV_compute_shader_derivatives
12951 #define GL_NV_compute_shader_derivatives 1
12952 GLAPI int GLAD_GL_NV_compute_shader_derivatives;
12953 #endif
12954 #ifndef GL_NV_conditional_render
12955 #define GL_NV_conditional_render 1
12956 GLAPI int GLAD_GL_NV_conditional_render;
12957 typedef void (APIENTRY PFNGBEGINCONDITIONALRENDERNVPROC)(GLuint id, GLenum mode);
12958 GLAPI PFNGBEGINCONDITIONALRENDERNVPROC glad_glBeginConditionalRenderNV;
12959 #define glBeginConditionalRenderNV glad_glBeginConditionalRenderNV
12960 typedef void (APIENTRY PFNGLENDCONDITIONALRENDERNVPROC)(void);
12961 GLAPI PFNGLENDCONDITIONALRENDERNVPROC glad_glEndConditionalRenderNV;
12962 #define glEndConditionalRenderNV glad_glEndConditionalRenderNV
12963 #endif
12964 #ifndef GL_NV_conservative_raster
12965 #define GL_NV_conservative_raster 1
12966 GLAPI int GLAD_GL_NV_conservative_raster;
12967 typedef void (APIENTRY PFNGLSUBPIXELPRECISIONBIASNVPROC)(GLuint xbits, GLuint ybits);
12968 GLAPI PFNGLSUBPIXELPRECISIONBIASNVPROC glad_glSubpixelPrecisionBiasNV;
12969 #define glSubpixelPrecisionBiasNV glad_glSubpixelPrecisionBiasNV
12970 #endif
12971 #ifndef GL_NV_conservative_raster_dilate
12972 #define GL_NV_conservative_raster_dilate 1
12973 GLAPI int GLAD_GL_NV_conservative_raster_dilate;
12974 typedef void (APIENTRY PFNGLCONSERVATIVERASTERPARAMETERFNVPROC)(GLenum pname, GLfloat value);
12975 GLAPI PFNGLCONSERVATIVERASTERPARAMETERFNVPROC glad_glConservativeRasterParameterfNV;
12976 #define glConservativeRasterParameterfNV glad_glConservativeRasterParameterfNV
12977 #endif
12978 #ifndef GL_NV_conservative_raster_pre_snap
12979 #define GL_NV_conservative_raster_pre_snap 1
12980 GLAPI int GLAD_GL_NV_conservative_raster_pre_snap;
12981 #endif
12982 #ifndef GL_NV_conservative_raster_pre_snap_triangles
12983 #define GL_NV_conservative_raster_pre_snap_triangles 1
12984 GLAPI int GLAD_GL_NV_conservative_raster_pre_snap_triangles;
12985 typedef void (APIENTRY PFNGLCONSERVATIVERASTERPARAMETERINNVPROC)(GLenum pname, GLint param);
12986 GLAPI PFNGLCONSERVATIVERASTERPARAMETERINNVPROC glad_glConservativeRasterParameteriNV;
12987 #define glConservativeRasterParameteriNV glad_glConservativeRasterParameteriNV
12988 #endif
12989 #ifndef GL_NV_conservative_raster_underestimation
12990 #define GL_NV_conservative_raster_underestimation 1
12991 GLAPI int GLAD_GL_NV_conservative_raster_underestimation;
12992 #endif
12993 #ifndef GL_NV_copy_depth_to_color
12994 #define GL_NV_copy_depth_to_color 1
12995 GLAPI int GLAD_GL_NV_copy_depth_to_color;
12996 #endif
12997 #ifndef GL_NV_copy_image
12998 #define GL_NV_copy_image 1
12999 GLAPI int GLAD_GL_NV_copy_image;
13000 typedef void (APIENTRY PFNGLCOPYIMAGESUBDATANVPROC)(GLuint srcName, GLenum srcTarget, GLint srcLevel,
GLint srcX, GLint srcY, GLint srcZ, GLuint dstName, GLenum dstTarget, GLint dstLevel, GLint dstX,

```



```

        GLint dstY, GLint dstZ, GLsizei width, GLsizei height, GLsizei depth);
13001 GLAPI PFNGLCOPYIMAGESUBDATANVPROC glad_glCopyImageSubDataNV;
13002 #define glCopyImageSubDataNV glad_glCopyImageSubDataNV
13003 #endif
13004 #ifndef GL_NV_deep_texture3D
13005 #define GL_NV_deep_texture3D 1
13006 GLAPI int GLAD_GL_NV_deep_texture3D;
13007 #endif
13008 #ifndef GL_NV_depth_buffer_float
13009 #define GL_NV_depth_buffer_float 1
13010 GLAPI int GLAD_GL_NV_depth_buffer_float;
13011 typedef void (APIENTRY PFNGLDEPTHFRANGEDNVPROC) (GLdouble zNear, GLdouble zFar);
13012 GLAPI PFNGLDEPTHFRANGEDNVPROC glad_glDepthRangedNV;
13013 #define glDepthRangedNV glad_glDepthRangedNV
13014 typedef void (APIENTRY PFNGLCLEARDEPTHDNVPROC) (GLdouble depth);
13015 GLAPI PFNGLCLEARDEPTHDNVPROC glad_glClearDepthdNV;
13016 #define glClearDepthdNV glad_glClearDepthdNV
13017 typedef void (APIENTRY PFNGLDEPTHBOUNDSDNVPROC) (GLdouble zmin, GLdouble zmax);
13018 GLAPI PFNGLDEPTHBOUNDSDNVPROC glad_glDepthBoundsdNV;
13019 #define glDepthBoundsdNV glad_glDepthBoundsdNV
13020 #endif
13021 #ifndef GL_NV_depth_clamp
13022 #define GL_NV_depth_clamp 1
13023 GLAPI int GLAD_GL_NV_depth_clamp;
13024 #endif
13025 #ifndef GL_NV_draw_texture
13026 #define GL_NV_draw_texture 1
13027 GLAPI int GLAD_GL_NV_draw_texture;
13028 typedef void (APIENTRY PFNGLDRAWTEXTURENVPROC) (GLuint texture, GLuint sampler, GLfloat x0, GLfloat
y0, GLfloat x1, GLfloat y1, GLfloat z, GLfloat s0, GLfloat s1, GLfloat t1);
13029 GLAPI PFNGLDRAWTEXTURENVPROC glad_glDrawTextureNV;
13030 #define glDrawTextureNV glad_glDrawTextureNV
13031 #endif
13032 #ifndef GL_NV_draw_vulkan_image
13033 #define GL_NV_draw_vulkan_image 1
13034 GLAPI int GLAD_GL_NV_draw_vulkan_image;
13035 typedef void (APIENTRY PFNGLDRAWVKIMAGENVPROC) (GLuint64 vkImage, GLuint sampler, GLfloat x0, GLfloat
y0, GLfloat x1, GLfloat y1, GLfloat z, GLfloat s0, GLfloat s1, GLfloat t1);
13036 GLAPI PFNGLDRAWVKIMAGENVPROC glad_glDrawVkImageNV;
13037 #define glDrawVkImageNV glad_glDrawVkImageNV
13038 typedef GLVULKANPROCNV (APIENTRY PFNGLGETVKPROCADDRNVPROC) (const GLchar *name);
13039 GLAPI PFNGLGETVKPROCADDRNVPROC glad_glGetVkProcAddrNV;
13040 #define glGetVkProcAddrNV glad_glGetVkProcAddrNV
13041 typedef void (APIENTRY PFNGLWAITVKSEMAPHORENVPROC) (GLuint64 vkSemaphore);
13042 GLAPI PFNGLWAITVKSEMAPHORENVPROC glad_glWaitVkSemaphoreNV;
13043 #define glWaitVkSemaphoreNV glad_glWaitVkSemaphoreNV
13044 typedef void (APIENTRY PFNGLSIGNALVKSEMAPHORENVPROC) (GLuint64 vkSemaphore);
13045 GLAPI PFNGLSIGNALVKSEMAPHORENVPROC glad_glSignalVkSemaphoreNV;
13046 #define glSignalVkSemaphoreNV glad_glSignalVkSemaphoreNV
13047 typedef void (APIENTRY PFNGLSIGNALVKFENCENVPROC) (GLuint64 vkFence);
13048 GLAPI PFNGLSIGNALVKFENCENVPROC glad_glSignalVkFenceNV;
13049 #define glSignalVkFenceNV glad_glSignalVkFenceNV
13050 #endif
13051 #ifndef GL_NV_evaluators
13052 #define GL_NV_evaluators 1
13053 GLAPI int GLAD_GL_NV_evaluators;
13054 typedef void (APIENTRY PFNGLMAPCONTROLPOINTSNVPROC) (GLenum target, GLuint index, GLenum type, GLsizei
ustride, GLsizei vstride, GLint uorder, GLint vorder, GLboolean packed, const void *points);
13055 GLAPI PFNGLMAPCONTROLPOINTSNVPROC glad_glMapControlPointsNV;
13056 #define glMapControlPointsNV glad_glMapControlPointsNV
13057 typedef void (APIENTRY PFNGLMAPPARAMETERIVNVPROC) (GLenum target, GLenum pname, const GLint *params);
13058 GLAPI PFNGLMAPPARAMETERIVNVPROC glad_glMapParameterivNV;
13059 #define glMapParameterivNV glad_glMapParameterivNV
13060 typedef void (APIENTRY PFNGLMAPPARAMETERFVNVPROC) (GLenum target, GLenum pname, const GLfloat
*params);
13061 GLAPI PFNGLMAPPARAMETERFVNVPROC glad_glMapParameterfvNV;
13062 #define glMapParameterfvNV glad_glMapParameterfvNV
13063 typedef void (APIENTRY PFNGLGETMAPCONTROLPOINTSNVPROC) (GLenum target, GLuint index, GLenum type,
GLsizei ustride, GLsizei vstride, GLboolean packed, void *points);
13064 GLAPI PFNGLGETMAPCONTROLPOINTSNVPROC glad_glGetMapControlPointsNV;
13065 #define glGetMapControlPointsNV glad_glGetMapControlPointsNV
13066 typedef void (APIENTRY PFNGLGETMAPPARAMETERIVNVPROC) (GLenum target, GLenum pname, GLint *params);
13067 GLAPI PFNGLGETMAPPARAMETERIVNVPROC glad_glGetMapParameterivNV;
13068 #define glGetMapParameterivNV glad_glGetMapParameterivNV
13069 typedef void (APIENTRY PFNGLGETMAPPARAMETERFVNVPROC) (GLenum target, GLenum pname, GLfloat *params);
13070 GLAPI PFNGLGETMAPPARAMETERFVNVPROC glad_glGetMapParameterfvNV;
13071 #define glGetMapParameterfvNV glad_glGetMapParameterfvNV
13072 typedef void (APIENTRY PFNGLGETMAPATTRIBPARAMETERIVNVPROC) (GLenum target, GLuint index, GLenum pname,
GLint *params);
13073 GLAPI PFNGLGETMAPATTRIBPARAMETERIVNVPROC glad_glGetMapAttribParameterivNV;
13074 #define glGetMapAttribParameterivNV glad_glGetMapAttribParameterivNV
13075 typedef void (APIENTRY PFNGLGETMAPATTRIBPARAMETERFVNVPROC) (GLenum target, GLuint index, GLenum pname,
GLfloat *params);
13076 GLAPI PFNGLGETMAPATTRIBPARAMETERFVNVPROC glad_glGetMapAttribParameterfvNV;
13077 #define glGetMapAttribParameterfvNV glad_glGetMapAttribParameterfvNV
13078 typedef void (APIENTRY PFNGLVALMAPSNVPROC) (GLenum target, GLenum mode);
13079 GLAPI PFNGLVALMAPSNVPROC glad_glEvalMapsNV;

```

```

13080 #define glEvalMapsNV glad_glEvalMapsNV
13081 #endif
13082 #ifndef GL_NV_explicit_multisample
13083 #define GL_NV_explicit_multisample 1
13084 GLAPI int GLAD_GL_NV_explicit_multisample;
13085 typedef void (APIENTRY PFNGLGETMULTISAMPLEFVNVPROC) (GLenum pname, GLuint index, GLfloat *val);
13086 GLAPI PFNGLGETMULTISAMPLEFVNVPROC glad_glGetMultisamplefvNV;
13087 #define glGetMultisamplefvNV glad_glGetMultisamplefvNV
13088 typedef void (APIENTRY PFNGLSAMPLEMASKINDEXEDNVPROC) (GLuint index, GLbitfield mask);
13089 GLAPI PFNGLSAMPLEMASKINDEXEDNVPROC glad_glSampleMaskIndexedNV;
13090 #define glSampleMaskIndexedNV glad_glSampleMaskIndexedNV
13091 typedef void (APIENTRY PFNGLTEXRENDERBUFFERNVPROC) (GLenum target, GLuint renderbuffer);
13092 GLAPI PFNGLTEXRENDERBUFFERNVPROC glad_glTexRenderbufferNV;
13093 #define glTexRenderbufferNV glad_glTexRenderbufferNV
13094 #endif
13095 #ifndef GL_NV_fence
13096 #define GL_NV_fence 1
13097 GLAPI int GLAD_GL_NV_fence;
13098 typedef void (APIENTRY PFNGLDELETEFENCESNVPROC) (GLsizei n, const GLuint *fences);
13099 GLAPI PFNGLDELETEFENCESNVPROC glad_glDeleteFencesNV;
13100 #define glDeleteFencesNV glad_glDeleteFencesNV
13101 typedef void (APIENTRY PFNGLGENFENCESNVPROC) (GLsizei n, GLuint *fences);
13102 GLAPI PFNGLGENFENCESNVPROC glad_glGenFencesNV;
13103 #define glGenFencesNV glad_glGenFencesNV
13104 typedef GLboolean (APIENTRY PFNGLISFENCENVPROC) (GLuint fence);
13105 GLAPI PFNGLISFENCENVPROC glad_glIsFenceNV;
13106 #define glIsFenceNV glad_glIsFenceNV
13107 typedef GLboolean (APIENTRY PFNGLTESTFENCENVPROC) (GLuint fence);
13108 GLAPI PFNGLTESTFENCENVPROC glad_glTestFenceNV;
13109 #define glTestFenceNV glad_glTestFenceNV
13110 typedef void (APIENTRY PFNGLGETFENCEIVNVPROC) (GLuint fence, GLenum pname, GLint *params);
13111 GLAPI PFNGLGETFENCEIVNVPROC glad_glGetFenceivNV;
13112 #define glGetFenceivNV glad_glGetFenceivNV
13113 typedef void (APIENTRY PFNGLFINISHFENCENVPROC) (GLuint fence);
13114 GLAPI PFNGLFINISHFENCENVPROC glad_glFinishFenceNV;
13115 #define glFinishFenceNV glad_glFinishFenceNV
13116 typedef void (APIENTRY PFNGLSETFENCENVPROC) (GLuint fence, GLenum condition);
13117 GLAPI PFNGLSETFENCENVPROC glad_glSetFenceNV;
13118 #define glSetFenceNV glad_glSetFenceNV
13119 #endif
13120 #ifndef GL_NV_fill_rectangle
13121 #define GL_NV_fill_rectangle 1
13122 GLAPI int GLAD_GL_NV_fill_rectangle;
13123 #endif
13124 #ifndef GL_NV_float_buffer
13125 #define GL_NV_float_buffer 1
13126 GLAPI int GLAD_GL_NV_float_buffer;
13127 #endif
13128 #ifndef GL_NV_fog_distance
13129 #define GL_NV_fog_distance 1
13130 GLAPI int GLAD_GL_NV_fog_distance;
13131 #endif
13132 #ifndef GL_NV_fragment_coverage_to_color
13133 #define GL_NV_fragment_coverage_to_color 1
13134 GLAPI int GLAD_GL_NV_fragment_coverage_to_color;
13135 typedef void (APIENTRY PFNGLFRAGMENTCOVERAGECOLORNVPROC) (GLuint color);
13136 GLAPI PFNGLFRAGMENTCOVERAGECOLORNVPROC glad_glFragmentCoverageColorNV;
13137 #define glFragmentCoverageColorNV glad_glFragmentCoverageColorNV
13138 #endif
13139 #ifndef GL_NV_fragment_program
13140 #define GL_NV_fragment_program 1
13141 GLAPI int GLAD_GL_NV_fragment_program;
13142 typedef void (APIENTRY PFNGLPROGRAMNAMEDPARAMETER4FNVPROC) (GLuint id, GLsizei len, const GLubyte
    *name, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
13143 GLAPI PFNGLPROGRAMNAMEDPARAMETER4FNVPROC glad_glProgramNamedParameter4fvNV;
13144 #define glProgramNamedParameter4fvNV glad_glProgramNamedParameter4fvNV
13145 typedef void (APIENTRY PFNGLPROGRAMNAMEDPARAMETER4FVNVPROC) (GLuint id, GLsizei len, const GLubyte
    *name, const GLfloat *v);
13146 GLAPI PFNGLPROGRAMNAMEDPARAMETER4FVNVPROC glad_glProgramNamedParameter4fvNV;
13147 #define glProgramNamedParameter4fvNV glad_glProgramNamedParameter4fvNV
13148 typedef void (APIENTRY PFNGLPROGRAMNAMEDPARAMETER4DNVPROC) (GLuint id, GLsizei len, const GLubyte
    *name, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
13149 GLAPI PFNGLPROGRAMNAMEDPARAMETER4DNVPROC glad_glProgramNamedParameter4dvNV;
13150 #define glProgramNamedParameter4dvNV glad_glProgramNamedParameter4dvNV
13151 typedef void (APIENTRY PFNGLPROGRAMNAMEDPARAMETER4DVNVPROC) (GLuint id, GLsizei len, const GLubyte
    *name, const GLdouble *v);
13152 GLAPI PFNGLPROGRAMNAMEDPARAMETER4DVNVPROC glad_glProgramNamedParameter4dvNV;
13153 #define glProgramNamedParameter4dvNV glad_glProgramNamedParameter4dvNV
13154 typedef void (APIENTRY PFNGLGETPROGRAMNAMEDPARAMETERFVNVPROC) (GLuint id, GLsizei len, const GLubyte
    *name, GLfloat *params);
13155 GLAPI PFNGLGETPROGRAMNAMEDPARAMETERFVNVPROC glad_glGetProgramNamedParameterfvNV;
13156 #define glGetProgramNamedParameterfvNV glad_glGetProgramNamedParameterfvNV
13157 typedef void (APIENTRY PFNGLGETPROGRAMNAMEDPARAMETERDVNVPROC) (GLuint id, GLsizei len, const GLubyte
    *name, GLdouble *params);
13158 GLAPI PFNGLGETPROGRAMNAMEDPARAMETERDVNVPROC glad_glGetProgramNamedParameterdvNV;
13159 #define glGetProgramNamedParameterdvNV glad_glGetProgramNamedParameterdvNV
13160 #endif

```



```

13161 #ifndef GL_NV_fragment_program2
13162 #define GL_NV_fragment_program2 1
13163 GLAPI int GLAD_GL_NV_fragment_program2;
13164 #endif
13165 #ifndef GL_NV_fragment_program4
13166 #define GL_NV_fragment_program4 1
13167 GLAPI int GLAD_GL_NV_fragment_program4;
13168 #endif
13169 #ifndef GL_NV_fragment_program_option
13170 #define GL_NV_fragment_program_option 1
13171 GLAPI int GLAD_GL_NV_fragment_program_option;
13172 #endif
13173 #ifndef GL_NV_fragment_shader_barycentric
13174 #define GL_NV_fragment_shader_barycentric 1
13175 GLAPI int GLAD_GL_NV_fragment_shader_barycentric;
13176 #endif
13177 #ifndef GL_NV_fragment_shader_interlock
13178 #define GL_NV_fragment_shader_interlock 1
13179 GLAPI int GLAD_GL_NV_fragment_shader_interlock;
13180 #endif
13181 #ifndef GL_NV_framebuffer_mixed_samples
13182 #define GL_NV_framebuffer_mixed_samples 1
13183 GLAPI int GLAD_GL_NV_framebuffer_mixed_samples;
13184 typedef void (APIENTRY PFNGLCOVERAGEMODULATIONTABLENVPROC) (GLsizei n, const GLfloat *v);
13185 GLAPI PFNGLCOVERAGEMODULATIONTABLENVPROC glad_glCoverageModulationTableNV;
13186 #define glCoverageModulationTableNV glad_glCoverageModulationTableNV
13187 typedef void (APIENTRY PFNGLGETCOVERAGEMODULATIONTABLENVPROC) (GLsizei bufSize, GLfloat *v);
13188 GLAPI PFNGLGETCOVERAGEMODULATIONTABLENVPROC glad_glGetCoverageModulationTableNV;
13189 #define glGetCoverageModulationTableNV glad_glGetCoverageModulationTableNV
13190 typedef void (APIENTRY PFNGLCOVERAGEMODULATIONNVPROC) (GLenum components);
13191 GLAPI PFNGLCOVERAGEMODULATIONNVPROC glad_glCoverageModulationNV;
13192 #define glCoverageModulationNV glad_glCoverageModulationNV
13193 #endif
13194 #ifndef GL_NV_framebuffer_multisample_coverage
13195 #define GL_NV_framebuffer_multisample_coverage 1
13196 GLAPI int GLAD_GL_NV_framebuffer_multisample_coverage;
13197 typedef void (APIENTRY PFNGLRENDERBUFFERSTORAGEMULTISAMPLECOVERGENVPROC) (GLenum target, GLsizei
coverageSamples, GLsizei colorSamples, GLenum internalformat, GLsizei width, GLsizei height);
13198 GLAPI PFNGLRENDERBUFFERSTORAGEMULTISAMPLECOVERGENVPROC glad_glRenderbufferStorageMultisampleCoverageNV;
13199 #define glRenderbufferStorageMultisampleCoverageNV glad_glRenderbufferStorageMultisampleCoverageNV
13200 #endif
13201 #ifndef GL_NV_geometry_program4
13202 #define GL_NV_geometry_program4 1
13203 GLAPI int GLAD_GL_NV_geometry_program4;
13204 typedef void (APIENTRY PFNGLPROGRAMVERTEXLIMITNVPROC) (GLenum target, GLint limit);
13205 GLAPI PFNGLPROGRAMVERTEXLIMITNVPROC glad_glProgramVertexLimitNV;
13206 #define glProgramVertexLimitNV glad_glProgramVertexLimitNV
13207 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTUREEXTPROC) (GLenum target, GLenum attachment, GLuint
texture, GLint level);
13208 GLAPI PFNGLFRAMEBUFFERTEXTUREEXTPROC glad_glFramebufferTextureEXT;
13209 #define glFramebufferTextureEXT glad_glFramebufferTextureEXT
13210 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTUREFACEEXTPROC) (GLenum target, GLenum attachment, GLuint
texture, GLint level, GLenum face);
13211 GLAPI PFNGLFRAMEBUFFERTEXTUREFACEEXTPROC glad_glFramebufferTextureFaceEXT;
13212 #define glFramebufferTextureFaceEXT glad_glFramebufferTextureFaceEXT
13213 #endif
13214 #ifndef GL_NV_geometry_shader4
13215 #define GL_NV_geometry_shader4 1
13216 GLAPI int GLAD_GL_NV_geometry_shader4;
13217 #endif
13218 #ifndef GL_NV_geometry_shader_passthrough
13219 #define GL_NV_geometry_shader_passthrough 1
13220 GLAPI int GLAD_GL_NV_geometry_shader_passthrough;
13221 #endif
13222 #ifndef GL_NV_gpu_multicast
13223 #define GL_NV_gpu_multicast 1
13224 GLAPI int GLAD_GL_NV_gpu_multicast;
13225 typedef void (APIENTRY PFNGLRENDERGPUMASKNVPROC) (GLbitfield mask);
13226 GLAPI PFNGLRENDERGPUMASKNVPROC glad_glRenderGpuMaskNV;
13227 #define glRenderGpuMaskNV glad_glRenderGpuMaskNV
13228 typedef void (APIENTRY PFNGLMULTICASTBUFFERSUBDATANVPROC) (GLbitfield gpuMask, GLuint buffer, GLintptr
offset, GLsizeiptr size, const void *data);
13229 GLAPI PFNGLMULTICASTBUFFERSUBDATANVPROC glad_glMulticastBufferSubDataNV;
13230 #define glMulticastBufferSubDataNV glad_glMulticastBufferSubDataNV
13231 typedef void (APIENTRY PFNGLMULTICASTCOPYBUFFERSUBDATANVPROC) (GLuint readGpu, GLbitfield
writeGpuMask, GLuint readBuffer, GLuint writeBuffer, GLintptr readOffset, GLintptr writeOffset,
GLsizeiptr size);
13232 GLAPI PFNGLMULTICASTCOPYBUFFERSUBDATANVPROC glad_glMulticastCopyBufferSubDataNV;
13233 #define glMulticastCopyBufferSubDataNV glad_glMulticastCopyBufferSubDataNV
13234 typedef void (APIENTRY PFNGLMULTICASTCOPYIMAGESUBDATANVPROC) (GLuint srcGpu, GLbitfield dstGpuMask,
GLuint srcName, GLenum srcTarget, GLint srcLevel, GLint srcX, GLint srcY, GLint srcZ, GLuint dstName,
GLenum dstTarget, GLint dstLevel, GLint dstX, GLint dstY, GLint dstZ, GLsizei srcWidth, GLsizei
srcHeight, GLsizei srcDepth);
13235 GLAPI PFNGLMULTICASTCOPYIMAGESUBDATANVPROC glad_glMulticastCopyImageSubDataNV;
13236 #define glMulticastCopyImageSubDataNV glad_glMulticastCopyImageSubDataNV
13237 typedef void (APIENTRY PFNGLMULTICASTBLITFRAMEBUFFERNVPROC) (GLuint srcGpu, GLuint dstGpu, GLint

```

```

        srcX0, GLint srcY0, GLint srcX1, GLint srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1,
        GLbitfield mask, GLenum filter);
13238 GLAPI PFNGLMULTICASTBLITFRAMEBUFFERNVPROC glad_glMulticastBlitFramebufferNV;
13239 #define glMulticastBlitFramebufferNV glad_glMulticastBlitFramebufferNV
13240 typedef void (APIENTRY PFNGLMULTICASTFRAMEBUFFERSAMPLELOCATIONSFVNVPROC) (GLuint gpu, GLuint
        framebuffer, GLuint start, GLsizei count, const GLfloat *v);
13241 GLAPI PFNGLMULTICASTFRAMEBUFFERSAMPLELOCATIONSFVNVPROC glad_glMulticastFramebufferSampleLocationsfvNV;
13242 #define glMulticastFramebufferSampleLocationsfvNV glad_glMulticastFramebufferSampleLocationsfvNV
13243 typedef void (APIENTRY PFNGLMULTICASTBARRIERNVPROC) (void);
13244 GLAPI PFNGLMULTICASTBARRIERNVPROC glad_glMulticastBarrierNV;
13245 #define glMulticastBarrierNV glad_glMulticastBarrierNV
13246 typedef void (APIENTRY PFNGLMULTICASTWAITSYNCPROC) (GLuint signalGpu, GLbitfield waitGpuMask);
13247 GLAPI PFNGLMULTICASTWAITSYNCPROC glad_glMulticastWaitSyncNV;
13248 #define glMulticastWaitSyncNV glad_glMulticastWaitSyncNV
13249 typedef void (APIENTRY PFNGLMULTICASTGETQUERYOBJECTIVNVPROC) (GLuint gpu, GLuint id, GLenum pname,
        GLint *params);
13250 GLAPI PFNGLMULTICASTGETQUERYOBJECTIVNVPROC glad_glMulticastGetQueryObjectivNV;
13251 #define glMulticastGetQueryObjectivNV glad_glMulticastGetQueryObjectivNV
13252 typedef void (APIENTRY PFNGLMULTICASTGETQUERYOBJECTUIVNVPROC) (GLuint gpu, GLuint id, GLenum pname,
        GLuint *params);
13253 GLAPI PFNGLMULTICASTGETQUERYOBJECTUIVNVPROC glad_glMulticastGetQueryObjectuivNV;
13254 #define glMulticastGetQueryObjectuivNV glad_glMulticastGetQueryObjectuivNV
13255 typedef void (APIENTRY PFNGLMULTICASTGETQUERYOBJECTI64VNVPROC) (GLuint gpu, GLuint id, GLenum pname,
        GLint64 *params);
13256 GLAPI PFNGLMULTICASTGETQUERYOBJECTI64VNVPROC glad_glMulticastGetQueryObjecti64vNV;
13257 #define glMulticastGetQueryObjecti64vNV glad_glMulticastGetQueryObjecti64vNV
13258 typedef void (APIENTRY PFNGLMULTICASTGETQUERYOBJECTUI64VNVPROC) (GLuint gpu, GLuint id, GLenum pname,
        GLuint64 *params);
13259 GLAPI PFNGLMULTICASTGETQUERYOBJECTUI64VNVPROC glad_glMulticastGetQueryObjectui64vNV;
13260 #define glMulticastGetQueryObjectui64vNV glad_glMulticastGetQueryObjectui64vNV
13261 #endif
13262 #ifndef GL_NV_gpu_program4
13263 #define GL_NV_gpu_program4 1
13264 GLAPI int GLAD_GL_NV_gpu_program4;
13265 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETERI4INVPROC) (GLenum target, GLuint index, GLint x,
        GLint y, GLint z, GLint w);
13266 GLAPI PFNGLPROGRAMLOCALPARAMETERI4INVPROC glad_glProgramLocalParameterI4iNV;
13267 #define glProgramLocalParameterI4iNV glad_glProgramLocalParameterI4iNV
13268 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETERI4IVNVPROC) (GLenum target, GLuint index, const GLint
        *params);
13269 GLAPI PFNGLPROGRAMLOCALPARAMETERI4IVNVPROC glad_glProgramLocalParameterI4ivNV;
13270 #define glProgramLocalParameterI4ivNV glad_glProgramLocalParameterI4ivNV
13271 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETERSI4IVNVPROC) (GLenum target, GLuint index, GLsizei
        count, const GLint *params);
13272 GLAPI PFNGLPROGRAMLOCALPARAMETERSI4IVNVPROC glad_glProgramLocalParametersI4ivNV;
13273 #define glProgramLocalParametersI4ivNV glad_glProgramLocalParametersI4ivNV
13274 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETERI4UIVNVPROC) (GLenum target, GLuint index, GLuint x,
        GLuint y, GLuint z, GLuint w);
13275 GLAPI PFNGLPROGRAMLOCALPARAMETERI4UIVNVPROC glad_glProgramLocalParameterI4uiNV;
13276 #define glProgramLocalParameterI4uiNV glad_glProgramLocalParameterI4uiNV
13277 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETERI4UIVNVPROC) (GLenum target, GLuint index, const
        GLuint *params);
13278 GLAPI PFNGLPROGRAMLOCALPARAMETERI4UIVNVPROC glad_glProgramLocalParameterI4uiNV;
13279 #define glProgramLocalParameterI4uiNV glad_glProgramLocalParameterI4uiNV
13280 typedef void (APIENTRY PFNGLPROGRAMLOCALPARAMETERSI4UIVNVPROC) (GLenum target, GLuint index, GLsizei
        count, const GLuint *params);
13281 GLAPI PFNGLPROGRAMLOCALPARAMETERSI4UIVNVPROC glad_glProgramLocalParametersI4uiNV;
13282 #define glProgramLocalParametersI4uiNV glad_glProgramLocalParametersI4uiNV
13283 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETERI4INVPROC) (GLenum target, GLuint index, GLint x, GLint
        y, GLint z, GLint w);
13284 GLAPI PFNGLPROGRAMENVPARAMETERI4INVPROC glad_glProgramEnvParameterI4iNV;
13285 #define glProgramEnvParameterI4iNV glad_glProgramEnvParameterI4iNV
13286 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETERI4IVNVPROC) (GLenum target, GLuint index, const GLint
        *params);
13287 GLAPI PFNGLPROGRAMENVPARAMETERI4IVNVPROC glad_glProgramEnvParameterI4ivNV;
13288 #define glProgramEnvParameterI4ivNV glad_glProgramEnvParameterI4ivNV
13289 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETERSI4IVNVPROC) (GLenum target, GLuint index, GLsizei
        count, const GLint *params);
13290 GLAPI PFNGLPROGRAMENVPARAMETERSI4IVNVPROC glad_glProgramEnvParametersI4ivNV;
13291 #define glProgramEnvParametersI4ivNV glad_glProgramEnvParametersI4ivNV
13292 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETERI4UIVNVPROC) (GLenum target, GLuint index, GLuint x,
        GLuint y, GLuint z, GLuint w);
13293 GLAPI PFNGLPROGRAMENVPARAMETERI4UIVNVPROC glad_glProgramEnvParameterI4uiNV;
13294 #define glProgramEnvParameterI4uiNV glad_glProgramEnvParameterI4uiNV
13295 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETERI4UIVNVPROC) (GLenum target, GLuint index, const GLuint
        *params);
13296 GLAPI PFNGLPROGRAMENVPARAMETERI4UIVNVPROC glad_glProgramEnvParameterI4uiNV;
13297 #define glProgramEnvParameterI4uiNV glad_glProgramEnvParameterI4uiNV
13298 typedef void (APIENTRY PFNGLPROGRAMENVPARAMETERSI4UIVNVPROC) (GLenum target, GLuint index, GLsizei
        count, const GLuint *params);
13299 GLAPI PFNGLPROGRAMENVPARAMETERSI4UIVNVPROC glad_glProgramEnvParametersI4uiNV;
13300 #define glProgramEnvParametersI4uiNV glad_glProgramEnvParametersI4uiNV
13301 typedef void (APIENTRY PFNGLGETPROGRAMLOCALPARAMETERIIVNVPROC) (GLenum target, GLuint index, GLint
        *params);
13302 GLAPI PFNGLGETPROGRAMLOCALPARAMETERIIVNVPROC glad_glGetProgramLocalParameterIivNV;
13303 #define glGetProgramLocalParameterIivNV glad_glGetProgramLocalParameterIivNV
13304 typedef void (APIENTRY PFNGLGETPROGRAMLOCALPARAMETERIUIVNVPROC) (GLenum target, GLuint index, GLuint

```

```

        *params);
13305 GLAPI PFNGLGETPROGRAMLOCALPARAMETERIUIVNVPROC glad_glGetProgramLocalParameterIuiNV;
13306 #define glGetProgramLocalParameterIuiNV glad_glGetProgramLocalParameterIuiNV
13307 typedef void (APIENTRY PFNGLGETPROGRAMENVPARAMETERIUIVNVPROC) (GLenum target, GLuint index, GLint
        *params);
13308 GLAPI PFNGLGETPROGRAMENVPARAMETERIUIVNVPROC glad_glGetProgramEnvParameterIuiNV;
13309 #define glGetProgramEnvParameterIuiNV glad_glGetProgramEnvParameterIuiNV
13310 typedef void (APIENTRY PFNGLGETPROGRAMENVPARAMETERIUIVNVPROC) (GLenum target, GLuint index, GLuint
        *params);
13311 GLAPI PFNGLGETPROGRAMENVPARAMETERIUIVNVPROC glad_glGetProgramEnvParameterIuiNV;
13312 #define glGetProgramEnvParameterIuiNV glad_glGetProgramEnvParameterIuiNV
13313 #endif
13314 #ifndef GL_NV_gpu_program5
13315 #define GL_NV_gpu_program5 1
13316 GLAPI int GLAD_GL_NV_gpu_program5;
13317 typedef void (APIENTRY PFNGLPROGRAMSUBROUTINEPARAMETERSUIVNVPROC) (GLenum target, GLsizei count, const
        GLuint *params);
13318 GLAPI PFNGLPROGRAMSUBROUTINEPARAMETERSUIVNVPROC glad_glProgramSubroutineParametersuiNV;
13319 #define glProgramSubroutineParametersuiNV glad_glProgramSubroutineParametersuiNV
13320 typedef void (APIENTRY PFNGLGETPROGRAMSUBROUTINEPARAMETERIUIVNVPROC) (GLenum target, GLuint index,
        GLuint *param);
13321 GLAPI PFNGLGETPROGRAMSUBROUTINEPARAMETERIUIVNVPROC glad_glGetProgramSubroutineParameteruiNV;
13322 #define glGetProgramSubroutineParameteruiNV glad_glGetProgramSubroutineParameteruiNV
13323 #endif
13324 #ifndef GL_NV_gpu_program5_mem_extended
13325 #define GL_NV_gpu_program5_mem_extended 1
13326 GLAPI int GLAD_GL_NV_gpu_program5_mem_extended;
13327 #endif
13328 #ifndef GL_NV_gpu_shader5
13329 #define GL_NV_gpu_shader5 1
13330 GLAPI int GLAD_GL_NV_gpu_shader5;
13331 #endif
13332 #ifndef GL_NV_half_float
13333 #define GL_NV_half_float 1
13334 GLAPI int GLAD_GL_NV_half_float;
13335 typedef void (APIENTRY PFNGLVERTEX2HNVPROC) (GLhalfNV x, GLhalfNV y);
13336 GLAPI PFNGLVERTEX2HNVPROC glad_glVertex2hNV;
13337 #define glVertex2hNV glad_glVertex2hNV
13338 typedef void (APIENTRY PFNGLVERTEX2HNVNVPROC) (const GLhalfNV *v);
13339 GLAPI PFNGLVERTEX2HNVNVPROC glad_glVertex2hNV;
13340 #define glVertex2hNVNV glad_glVertex2hNV
13341 typedef void (APIENTRY PFNGLVERTEX3HNVPROC) (GLhalfNV x, GLhalfNV y, GLhalfNV z);
13342 GLAPI PFNGLVERTEX3HNVPROC glad_glVertex3hNV;
13343 #define glVertex3hNV glad_glVertex3hNV
13344 typedef void (APIENTRY PFNGLVERTEX3HNVNVPROC) (const GLhalfNV *v);
13345 GLAPI PFNGLVERTEX3HNVNVPROC glad_glVertex3hNV;
13346 #define glVertex3hNVNV glad_glVertex3hNV
13347 typedef void (APIENTRY PFNGLVERTEX4HNVPROC) (GLhalfNV x, GLhalfNV y, GLhalfNV z, GLhalfNV w);
13348 GLAPI PFNGLVERTEX4HNVPROC glad_glVertex4hNV;
13349 #define glVertex4hNV glad_glVertex4hNV
13350 typedef void (APIENTRY PFNGLVERTEX4HNVNVPROC) (const GLhalfNV *v);
13351 GLAPI PFNGLVERTEX4HNVNVPROC glad_glVertex4hNV;
13352 #define glVertex4hNVNV glad_glVertex4hNV
13353 typedef void (APIENTRY PFNGLNORMAL3HNVPROC) (GLhalfNV nx, GLhalfNV ny, GLhalfNV nz);
13354 GLAPI PFNGLNORMAL3HNVPROC glad_glNormal3hNV;
13355 #define glNormal3hNV glad_glNormal3hNV
13356 typedef void (APIENTRY PFNGLNORMAL3HNVNVPROC) (const GLhalfNV *v);
13357 GLAPI PFNGLNORMAL3HNVNVPROC glad_glNormal3hNV;
13358 #define glNormal3hNVNV glad_glNormal3hNV
13359 typedef void (APIENTRY PFNGLCOLOR3HNVPROC) (GLhalfNV red, GLhalfNV green, GLhalfNV blue);
13360 GLAPI PFNGLCOLOR3HNVPROC glad_glColor3hNV;
13361 #define glColor3hNV glad_glColor3hNV
13362 typedef void (APIENTRY PFNGLCOLOR3HNVNVPROC) (const GLhalfNV *v);
13363 GLAPI PFNGLCOLOR3HNVNVPROC glad_glColor3hNV;
13364 #define glColor3hNVNV glad_glColor3hNV
13365 typedef void (APIENTRY PFNGLCOLOR4HNVPROC) (GLhalfNV red, GLhalfNV green, GLhalfNV blue, GLhalfNV
        alpha);
13366 GLAPI PFNGLCOLOR4HNVPROC glad_glColor4hNV;
13367 #define glColor4hNV glad_glColor4hNV
13368 typedef void (APIENTRY PFNGLCOLOR4HNVNVPROC) (const GLhalfNV *v);
13369 GLAPI PFNGLCOLOR4HNVNVPROC glad_glColor4hNV;
13370 #define glColor4hNVNV glad_glColor4hNV
13371 typedef void (APIENTRY PFNGLTEXCOORD1HNVPROC) (GLhalfNV s);
13372 GLAPI PFNGLTEXCOORD1HNVPROC glad_glTexCoord1hNV;
13373 #define glTexCoord1hNV glad_glTexCoord1hNV
13374 typedef void (APIENTRY PFNGLTEXCOORD1HNVNVPROC) (const GLhalfNV *v);
13375 GLAPI PFNGLTEXCOORD1HNVNVPROC glad_glTexCoord1hNV;
13376 #define glTexCoord1hNVNV glad_glTexCoord1hNV
13377 typedef void (APIENTRY PFNGLTEXCOORD2HNVPROC) (GLhalfNV s, GLhalfNV t);
13378 GLAPI PFNGLTEXCOORD2HNVPROC glad_glTexCoord2hNV;
13379 #define glTexCoord2hNV glad_glTexCoord2hNV
13380 typedef void (APIENTRY PFNGLTEXCOORD2HNVNVPROC) (const GLhalfNV *v);
13381 GLAPI PFNGLTEXCOORD2HNVNVPROC glad_glTexCoord2hNV;
13382 #define glTexCoord2hNVNV glad_glTexCoord2hNV
13383 typedef void (APIENTRY PFNGLTEXCOORD3HNVPROC) (GLhalfNV s, GLhalfNV t, GLhalfNV r);
13384 GLAPI PFNGLTEXCOORD3HNVPROC glad_glTexCoord3hNV;
13385 #define glTexCoord3hNV glad_glTexCoord3hNV

```

```

13386 typedef void (APIENTRY PFNGLTEXCOORD3HVNVPROC)(const GLhalfNV *v);
13387 GLAPI PFNGLTEXCOORD3HVNVPROC glad_glTexCoord3hvnv;
13388 #define glTexCoord3hvnv glad_glTexCoord3hvnv
13389 typedef void (APIENTRY PFNGLTEXCOORD4HVNVPROC)(GLhalfNV s, GLhalfNV t, GLhalfNV r, GLhalfNV q);
13390 GLAPI PFNGLTEXCOORD4HVNVPROC glad_glTexCoord4hvnv;
13391 #define glTexCoord4hvnv glad_glTexCoord4hvnv
13392 typedef void (APIENTRY PFNGLTEXCOORD4HVNVPROC)(const GLhalfNV *v);
13393 GLAPI PFNGLTEXCOORD4HVNVPROC glad_glTexCoord4hvnv;
13394 #define glTexCoord4hvnv glad_glTexCoord4hvnv
13395 typedef void (APIENTRY PFNGLMULTITEXCOORD1HVNVPROC)(GLenum target, GLhalfNV s);
13396 GLAPI PFNGLMULTITEXCOORD1HVNVPROC glad_glMultiTexCoord1hvnv;
13397 #define glMultiTexCoord1hvnv glad_glMultiTexCoord1hvnv
13398 typedef void (APIENTRY PFNGLMULTITEXCOORD1HVNVPROC)(GLenum target, const GLhalfNV *v);
13399 GLAPI PFNGLMULTITEXCOORD1HVNVPROC glad_glMultiTexCoord1hvnv;
13400 #define glMultiTexCoord1hvnv glad_glMultiTexCoord1hvnv
13401 typedef void (APIENTRY PFNGLMULTITEXCOORD2HVNVPROC)(GLenum target, GLhalfNV s, GLhalfNV t);
13402 GLAPI PFNGLMULTITEXCOORD2HVNVPROC glad_glMultiTexCoord2hvnv;
13403 #define glMultiTexCoord2hvnv glad_glMultiTexCoord2hvnv
13404 typedef void (APIENTRY PFNGLMULTITEXCOORD2HVNVPROC)(GLenum target, const GLhalfNV *v);
13405 GLAPI PFNGLMULTITEXCOORD2HVNVPROC glad_glMultiTexCoord2hvnv;
13406 #define glMultiTexCoord2hvnv glad_glMultiTexCoord2hvnv
13407 typedef void (APIENTRY PFNGLMULTITEXCOORD3HVNVPROC)(GLenum target, GLhalfNV s, GLhalfNV t, GLhalfNV
r);
13408 GLAPI PFNGLMULTITEXCOORD3HVNVPROC glad_glMultiTexCoord3hvnv;
13409 #define glMultiTexCoord3hvnv glad_glMultiTexCoord3hvnv
13410 typedef void (APIENTRY PFNGLMULTITEXCOORD3HVNVPROC)(GLenum target, const GLhalfNV *v);
13411 GLAPI PFNGLMULTITEXCOORD3HVNVPROC glad_glMultiTexCoord3hvnv;
13412 #define glMultiTexCoord3hvnv glad_glMultiTexCoord3hvnv
13413 typedef void (APIENTRY PFNGLMULTITEXCOORD4HVNVPROC)(GLenum target, GLhalfNV s, GLhalfNV t, GLhalfNV r,
GLhalfNV q);
13414 GLAPI PFNGLMULTITEXCOORD4HVNVPROC glad_glMultiTexCoord4hvnv;
13415 #define glMultiTexCoord4hvnv glad_glMultiTexCoord4hvnv
13416 typedef void (APIENTRY PFNGLMULTITEXCOORD4HVNVPROC)(GLenum target, const GLhalfNV *v);
13417 GLAPI PFNGLMULTITEXCOORD4HVNVPROC glad_glMultiTexCoord4hvnv;
13418 #define glMultiTexCoord4hvnv glad_glMultiTexCoord4hvnv
13419 typedef void (APIENTRY PFNGLVERTEXATTRIB1HVNVPROC)(GLuint index, GLhalfNV x);
13420 GLAPI PFNGLVERTEXATTRIB1HVNVPROC glad_glVertexAttrib1hvnv;
13421 #define glVertexAttrib1hvnv glad_glVertexAttrib1hvnv
13422 typedef void (APIENTRY PFNGLVERTEXATTRIB1HVNVPROC)(GLuint index, const GLhalfNV *v);
13423 GLAPI PFNGLVERTEXATTRIB1HVNVPROC glad_glVertexAttrib1hvnv;
13424 #define glVertexAttrib1hvnv glad_glVertexAttrib1hvnv
13425 typedef void (APIENTRY PFNGLVERTEXATTRIB2HVNVPROC)(GLuint index, GLhalfNV x, GLhalfNV y);
13426 GLAPI PFNGLVERTEXATTRIB2HVNVPROC glad_glVertexAttrib2hvnv;
13427 #define glVertexAttrib2hvnv glad_glVertexAttrib2hvnv
13428 typedef void (APIENTRY PFNGLVERTEXATTRIB2HVNVPROC)(GLuint index, const GLhalfNV *v);
13429 GLAPI PFNGLVERTEXATTRIB2HVNVPROC glad_glVertexAttrib2hvnv;
13430 #define glVertexAttrib2hvnv glad_glVertexAttrib2hvnv
13431 typedef void (APIENTRY PFNGLVERTEXATTRIB3HVNVPROC)(GLuint index, GLhalfNV x, GLhalfNV y, GLhalfNV z);
13432 GLAPI PFNGLVERTEXATTRIB3HVNVPROC glad_glVertexAttrib3hvnv;
13433 #define glVertexAttrib3hvnv glad_glVertexAttrib3hvnv
13434 typedef void (APIENTRY PFNGLVERTEXATTRIB3HVNVPROC)(GLuint index, const GLhalfNV *v);
13435 GLAPI PFNGLVERTEXATTRIB3HVNVPROC glad_glVertexAttrib3hvnv;
13436 #define glVertexAttrib3hvnv glad_glVertexAttrib3hvnv
13437 typedef void (APIENTRY PFNGLVERTEXATTRIB4HVNVPROC)(GLuint index, GLhalfNV x, GLhalfNV y, GLhalfNV z,
GLhalfNV w);
13438 GLAPI PFNGLVERTEXATTRIB4HVNVPROC glad_glVertexAttrib4hvnv;
13439 #define glVertexAttrib4hvnv glad_glVertexAttrib4hvnv
13440 typedef void (APIENTRY PFNGLVERTEXATTRIB4HVNVPROC)(GLuint index, const GLhalfNV *v);
13441 GLAPI PFNGLVERTEXATTRIB4HVNVPROC glad_glVertexAttrib4hvnv;
13442 #define glVertexAttrib4hvnv glad_glVertexAttrib4hvnv
13443 typedef void (APIENTRY PFNGLVERTEXATTRIBS1HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
13444 GLAPI PFNGLVERTEXATTRIBS1HVNVPROC glad_glVertexAttribs1hvnv;
13445 #define glVertexAttribs1hvnv glad_glVertexAttribs1hvnv
13446 typedef void (APIENTRY PFNGLVERTEXATTRIBS2HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
13447 GLAPI PFNGLVERTEXATTRIBS2HVNVPROC glad_glVertexAttribs2hvnv;
13448 #define glVertexAttribs2hvnv glad_glVertexAttribs2hvnv
13449 typedef void (APIENTRY PFNGLVERTEXATTRIBS3HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
13450 GLAPI PFNGLVERTEXATTRIBS3HVNVPROC glad_glVertexAttribs3hvnv;
13451 #define glVertexAttribs3hvnv glad_glVertexAttribs3hvnv
13452 typedef void (APIENTRY PFNGLVERTEXATTRIBS4HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
13453 GLAPI PFNGLVERTEXATTRIBS4HVNVPROC glad_glVertexAttribs4hvnv;
13454 #define glVertexAttribs4hvnv glad_glVertexAttribs4hvnv
13455 typedef void (APIENTRY PFNGLFOGCOORDHVNVPROC)(GLhalfNV fog);
13456 GLAPI PFNGLFOGCOORDHVNVPROC glad_glFogCoordhvnv;
13457 #define glFogCoordhvnv glad_glFogCoordhvnv
13458 typedef void (APIENTRY PFNGLFOGCOORDHVNVPROC)(const GLhalfNV *fog);
13459 GLAPI PFNGLFOGCOORDHVNVPROC glad_glFogCoordhvnv;
13460 #define glFogCoordhvnv glad_glFogCoordhvnv
13461 typedef void (APIENTRY PFNGLSECONDARYCOLOR3HVNVPROC)(GLhalfNV red, GLhalfNV green, GLhalfNV blue);
13462 GLAPI PFNGLSECONDARYCOLOR3HVNVPROC glad_glSecondaryColor3hvnv;
13463 #define glSecondaryColor3hvnv glad_glSecondaryColor3hvnv
13464 typedef void (APIENTRY PFNGLSECONDARYCOLOR3HVNVPROC)(const GLhalfNV *v);
13465 GLAPI PFNGLSECONDARYCOLOR3HVNVPROC glad_glSecondaryColor3hvnv;
13466 #define glSecondaryColor3hvnv glad_glSecondaryColor3hvnv
13467 typedef void (APIENTRY PFNGLVERTEXWEIGHTHVNVPROC)(GLhalfNV weight);
13468 GLAPI PFNGLVERTEXWEIGHTHVNVPROC glad_glVertexWeighthvnv;
13469 #define glVertexWeighthvnv glad_glVertexWeighthvnv

```

```

13470 typedef void (APIENTRY PFNGLVERTEXWEIGHTHVNVPROC)(const GLhalfNV *weight);
13471 GLAPI PFNGLVERTEXWEIGHTHVNVPROC glad_glVertexWeightvNV;
13472 #define glVertexWeightvNV glad_glVertexWeightvNV
13473 #endif
13474 #ifndef GL_NV_internalformat_sample_query
13475 #define GL_NV_internalformat_sample_query 1
13476 GLAPI int GLAD_GL_NV_internalformat_sample_query;
13477 typedef void (APIENTRY PFNGLGETINTERNALFORMATSAMPLEIVNVPROC)(GLenum target, GLenum internalformat,
13478 GLsizei samples, GLenum pname, GLsizei count, GLint *params);
13479 GLAPI PFNGLGETINTERNALFORMATSAMPLEIVNVPROC glad_glGetInternalFormatSampleivNV;
13480 #define glGetInternalFormatSampleivNV glad_glGetInternalFormatSampleivNV
13481 #endif
13482 #ifndef GL_NV_light_max_exponent
13483 #define GL_NV_light_max_exponent 1
13484 GLAPI int GLAD_GL_NV_light_max_exponent;
13485 #endif
13486 #ifndef GL_NV_memory_attachment
13487 #define GL_NV_memory_attachment 1
13488 GLAPI int GLAD_GL_NV_memory_attachment;
13489 typedef void (APIENTRY PFNGLGETMEMORYOBJECTDETACHEDRESOURCESUIVNVPROC)(GLuint memory, GLenum pname,
13490 GLint first, GLsizei count, GLuint *params);
13491 GLAPI PFNGLGETMEMORYOBJECTDETACHEDRESOURCESUIVNVPROC glad_glGetMemoryObjectDetachedResourcesuivNV;
13492 #define glGetMemoryObjectDetachedResourcesuivNV glad_glGetMemoryObjectDetachedResourcesuivNV
13493 typedef void (APIENTRY PFNGLRESETMEMORYOBJECTPARAMETERNVPROC)(GLuint memory, GLenum pname);
13494 GLAPI PFNGLRESETMEMORYOBJECTPARAMETERNVPROC glad_glResetMemoryObjectParameterNV;
13495 #define glResetMemoryObjectParameterNV glad_glResetMemoryObjectParameterNV
13496 typedef void (APIENTRY PFNGLTEXATTACHMEMORYNVPROC)(GLenum target, GLuint memory, GLuint64 offset);
13497 GLAPI PFNGLTEXATTACHMEMORYNVPROC glad_glTexAttachMemoryNV;
13498 #define glTexAttachMemoryNV glad_glTexAttachMemoryNV
13499 typedef void (APIENTRY PFNGLBUFFERATTACHMEMORYNVPROC)(GLenum target, GLuint memory, GLuint64 offset);
13500 GLAPI PFNGLBUFFERATTACHMEMORYNVPROC glad_glBufferAttachMemoryNV;
13501 #define glBufferAttachMemoryNV glad_glBufferAttachMemoryNV
13502 typedef void (APIENTRY PFNGLTEXTUREATTACHMEMORYNVPROC)(GLuint texture, GLuint memory, GLuint64
13503 offset);
13504 GLAPI PFNGLTEXTUREATTACHMEMORYNVPROC glad_glTextureAttachMemoryNV;
13505 #define glTextureAttachMemoryNV glad_glTextureAttachMemoryNV
13506 typedef void (APIENTRY PFNGLNAMEDBUFFERATTACHMEMORYNVPROC)(GLuint buffer, GLuint memory, GLuint64
13507 offset);
13508 GLAPI PFNGLNAMEDBUFFERATTACHMEMORYNVPROC glad_glNamedBufferAttachMemoryNV;
13509 #define glNamedBufferAttachMemoryNV glad_glNamedBufferAttachMemoryNV
13510 #endif
13511 #ifndef GL_NV_memory_object_sparse
13512 #define GL_NV_memory_object_sparse 1
13513 GLAPI int GLAD_GL_NV_memory_object_sparse;
13514 typedef void (APIENTRY PFNGLBUFFERPAGECOMMITMENTMEMNVPROC)(GLenum target, GLintptr offset, GLsizeiptr
13515 size, GLuint memory, GLuint64 memOffset, GLboolean commit);
13516 GLAPI PFNGLBUFFERPAGECOMMITMENTMEMNVPROC glad_glBufferPageCommitmentMemNV;
13517 #define glBufferPageCommitmentMemNV glad_glBufferPageCommitmentMemNV
13518 typedef void (APIENTRY PFNGLTEXPAGECOMMITMENTMEMNVPROC)(GLenum target, GLint layer, GLint level,
13519 GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLuint
13520 memory, GLuint64 offset, GLboolean commit);
13521 GLAPI PFNGLTEXPAGECOMMITMENTMEMNVPROC glad_glTexPageCommitmentMemNV;
13522 #define glTexPageCommitmentMemNV glad_glTexPageCommitmentMemNV
13523 typedef void (APIENTRY PFNGLNAMEDBUFFERPAGECOMMITMENTMEMNVPROC)(GLuint buffer, GLintptr offset,
13524 GLsizeiptr size, GLuint memory, GLuint64 memOffset, GLboolean commit);
13525 GLAPI PFNGLNAMEDBUFFERPAGECOMMITMENTMEMNVPROC glad_glNamedBufferPageCommitmentMemNV;
13526 #define glNamedBufferPageCommitmentMemNV glad_glNamedBufferPageCommitmentMemNV
13527 typedef void (APIENTRY PFNGLTEXTUREPAGECOMMITMENTMEMNVPROC)(GLuint texture, GLint layer, GLint level,
13528 GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLuint
13529 memory, GLuint64 offset, GLboolean commit);
13530 GLAPI PFNGLTEXTUREPAGECOMMITMENTMEMNVPROC glad_glTexturePageCommitmentMemNV;
13531 #define glTexturePageCommitmentMemNV glad_glTexturePageCommitmentMemNV
13532 #endif
13533 #ifndef GL_NV_mesh_shader
13534 #define GL_NV_mesh_shader 1
13535 GLAPI int GLAD_GL_NV_mesh_shader;
13536 typedef void (APIENTRY PFNGLDRAWMESHTASKSNVPROC)(GLuint first, GLuint count);
13537 GLAPI PFNGLDRAWMESHTASKSNVPROC glad_glDrawMeshTasksNV;
13538 #define glDrawMeshTasksNV glad_glDrawMeshTasksNV
13539 typedef void (APIENTRY PFNGLDRAWMESHTASKSINDIRECTNVPROC)(GLintptr indirect);
13540 GLAPI PFNGLDRAWMESHTASKSINDIRECTNVPROC glad_glDrawMeshTasksIndirectNV;
13541 #define glDrawMeshTasksIndirectNV glad_glDrawMeshTasksIndirectNV
13542 typedef void (APIENTRY PFNGLMULTIDRAWMESHTASKSINDIRECTNVPROC)(GLintptr indirect, GLsizei drawcount,
13543 GLsizei stride);
13544 GLAPI PFNGLMULTIDRAWMESHTASKSINDIRECTNVPROC glad_glMultiDrawMeshTasksIndirectNV;
13545 #define glMultiDrawMeshTasksIndirectNV glad_glMultiDrawMeshTasksIndirectNV
13546 typedef void (APIENTRY PFNGLMULTIDRAWMESHTASKSINDIRECTCOUNTNVPROC)(GLintptr indirect, GLintptr
13547 drawcount, GLsizei maxdrawcount, GLsizei stride);
13548 GLAPI PFNGLMULTIDRAWMESHTASKSINDIRECTCOUNTNVPROC glad_glMultiDrawMeshTasksIndirectCountNV;
13549 #define glMultiDrawMeshTasksIndirectCountNV glad_glMultiDrawMeshTasksIndirectCountNV
13550 #endif
13551 #ifndef GL_NV_multisample_coverage
13552 #define GL_NV_multisample_coverage 1
13553 GLAPI int GLAD_GL_NV_multisample_coverage;
13554 #endif
13555 #ifndef GL_NV_multisample_filter_hint
13556 #define GL_NV_multisample_filter_hint 1

```



```

13545 GLAPI int GLAD_GL_NV_multisample_filter_hint;
13546 #endif
13547 #ifndef GL_NV_occlusion_query
13548 #define GL_NV_occlusion_query 1
13549 GLAPI int GLAD_GL_NV_occlusion_query;
13550 typedef void (APIENTRY PFNGLGENOCCLUSIONQUERIESNVPROC) (GLsizei n, GLuint *ids);
13551 GLAPI PFNGLGENOCCLUSIONQUERIESNVPROC glad_glGenOcclusionQueriesNV;
13552 #define glGenOcclusionQueriesNV glad_glGenOcclusionQueriesNV
13553 typedef void (APIENTRY PFNGLDELETEOCCLUSIONQUERIESNVPROC) (GLsizei n, const GLuint *ids);
13554 GLAPI PFNGLDELETEOCCLUSIONQUERIESNVPROC glad_glDeleteOcclusionQueriesNV;
13555 #define glDeleteOcclusionQueriesNV glad_glDeleteOcclusionQueriesNV
13556 typedef GLboolean (APIENTRY PFNGLISOCCLUSIONQUERYNVPROC) (GLuint id);
13557 GLAPI PFNGLISOCCLUSIONQUERYNVPROC glad_glIsOcclusionQueryNV;
13558 #define glIsOcclusionQueryNV glad_glIsOcclusionQueryNV
13559 typedef void (APIENTRY PFNGLBEGINOCCLUSIONQUERYNVPROC) (GLuint id);
13560 GLAPI PFNGLBEGINOCCLUSIONQUERYNVPROC glad_glBeginOcclusionQueryNV;
13561 #define glBeginOcclusionQueryNV glad_glBeginOcclusionQueryNV
13562 typedef void (APIENTRY PFNGLENDOCCLUSIONQUERYNVPROC) (void);
13563 GLAPI PFNGLENDOCCLUSIONQUERYNVPROC glad_glEndOcclusionQueryNV;
13564 #define glEndOcclusionQueryNV glad_glEndOcclusionQueryNV
13565 typedef void (APIENTRY PFNGLGETOCCLUSIONQUERYIIVNVPROC) (GLuint id, GLenum pname, GLint *params);
13566 GLAPI PFNGLGETOCCLUSIONQUERYIIVNVPROC glad_glGetOcclusionQueryiivNV;
13567 #define glGetOcclusionQueryiivNV glad_glGetOcclusionQueryiivNV
13568 typedef void (APIENTRY PFNGLGETOCCLUSIONQUERYUIIVNVPROC) (GLuint id, GLenum pname, GLuint *params);
13569 GLAPI PFNGLGETOCCLUSIONQUERYUIIVNVPROC glad_glGetOcclusionQueryuiivNV;
13570 #define glGetOcclusionQueryuiivNV glad_glGetOcclusionQueryuiivNV
13571 #endif
13572 #ifndef GL_NV_packed_depth_stencil
13573 #define GL_NV_packed_depth_stencil 1
13574 GLAPI int GLAD_GL_NV_packed_depth_stencil;
13575 #endif
13576 #ifndef GL_NV_parameter_buffer_object
13577 #define GL_NV_parameter_buffer_object 1
13578 GLAPI int GLAD_GL_NV_parameter_buffer_object;
13579 typedef void (APIENTRY PFNGLPROGRAMBUFFERPARAMETERSFVNVPROC) (GLenum target, GLuint bindingIndex,
    GLuint wordIndex, GLsizei count, const GLfloat *params);
13580 GLAPI PFNGLPROGRAMBUFFERPARAMETERSFVNVPROC glad_glProgramBufferParametersfvNV;
13581 #define glProgramBufferParametersfvNV glad_glProgramBufferParametersfvNV
13582 typedef void (APIENTRY PFNGLPROGRAMBUFFERPARAMETERSIIVNVPROC) (GLenum target, GLuint bindingIndex,
    GLuint wordIndex, GLsizei count, const GLint *params);
13583 GLAPI PFNGLPROGRAMBUFFERPARAMETERSIIVNVPROC glad_glProgramBufferParametersiivNV;
13584 #define glProgramBufferParametersiivNV glad_glProgramBufferParametersiivNV
13585 typedef void (APIENTRY PFNGLPROGRAMBUFFERPARAMETERSUIIVNVPROC) (GLenum target, GLuint bindingIndex,
    GLuint wordIndex, GLsizei count, const GLuint *params);
13586 GLAPI PFNGLPROGRAMBUFFERPARAMETERSUIIVNVPROC glad_glProgramBufferParametersuiivNV;
13587 #define glProgramBufferParametersuiivNV glad_glProgramBufferParametersuiivNV
13588 #endif
13589 #ifndef GL_NV_parameter_buffer_object2
13590 #define GL_NV_parameter_buffer_object2 1
13591 GLAPI int GLAD_GL_NV_parameter_buffer_object2;
13592 #endif
13593 #ifndef GL_NV_path_rendering
13594 #define GL_NV_path_rendering 1
13595 GLAPI int GLAD_GL_NV_path_rendering;
13596 typedef GLuint (APIENTRY PFNGLGENPATHSNVPROC) (GLsizei range);
13597 GLAPI PFNGLGENPATHSNVPROC glad_glGenPathsNV;
13598 #define glGenPathsNV glad_glGenPathsNV
13599 typedef void (APIENTRY PFNGLDELETEPATHSNVPROC) (GLuint path, GLsizei range);
13600 GLAPI PFNGLDELETEPATHSNVPROC glad_glDeletePathsNV;
13601 #define glDeletePathsNV glad_glDeletePathsNV
13602 typedef GLboolean (APIENTRY PFNGLISPATHNVPROC) (GLuint path);
13603 GLAPI PFNGLISPATHNVPROC glad_glIsPathNV;
13604 #define glIsPathNV glad_glIsPathNV
13605 typedef void (APIENTRY PFNGLPATHCOMMANDSNVPROC) (GLuint path, GLsizei numCommands, const GLubyte
    *commands, GLsizei numCoords, GLenum coordType, const void *coords);
13606 GLAPI PFNGLPATHCOMMANDSNVPROC glad_glPathCommandsNV;
13607 #define glPathCommandsNV glad_glPathCommandsNV
13608 typedef void (APIENTRY PFNGLPATHCOORDSNVPROC) (GLuint path, GLsizei numCoords, GLenum coordType, const
    void *coords);
13609 GLAPI PFNGLPATHCOORDSNVPROC glad_glPathCoordsNV;
13610 #define glPathCoordsNV glad_glPathCoordsNV
13611 typedef void (APIENTRY PFNGLPATHSUBCOMMANDSNVPROC) (GLuint path, GLsizei commandStart, GLsizei
    commandsToDelete, GLsizei numCommands, const GLubyte *commands, GLsizei numCoords, GLenum coordType,
    const void *coords);
13612 GLAPI PFNGLPATHSUBCOMMANDSNVPROC glad_glPathSubCommandsNV;
13613 #define glPathSubCommandsNV glad_glPathSubCommandsNV
13614 typedef void (APIENTRY PFNGLPATHSUBCOORDSNVPROC) (GLuint path, GLsizei coordStart, GLsizei numCoords,
    GLenum coordType, const void *coords);
13615 GLAPI PFNGLPATHSUBCOORDSNVPROC glad_glPathSubCoordsNV;
13616 #define glPathSubCoordsNV glad_glPathSubCoordsNV
13617 typedef void (APIENTRY PFNGLPATHSTRINGNVPROC) (GLuint path, GLenum format, GLsizei length, const void
    *pathString);
13618 GLAPI PFNGLPATHSTRINGNVPROC glad_glPathStringNV;
13619 #define glPathStringNV glad_glPathStringNV
13620 typedef void (APIENTRY PFNGLPATHGLYPHSNVPROC) (GLuint firstPathName, GLenum fontTarget, const void
    *fontName, GLbitfield fontStyle, GLsizei numGlyphs, GLenum type, const void *charcodes, GLenum
    handleMissingGlyphs, GLuint pathParameterTemplate, GLfloat emScale);

```

```

13621 GLAPI PFNGLPATHGLYPHSNVPROC glad_glPathGlyphsNV;
13622 #define glPathGlyphsNV glad_glPathGlyphsNV
13623 typedef void (APIENTRY PFNGLPATHGLYPHRANGENVPROC) (GLuint firstPathName, GLenum fontTarget, const void
    *fontName, GLbitfield fontStyle, GLuint firstGlyph, GLsizei numGlyphs, GLenum handleMissingGlyphs,
    GLuint pathParameterTemplate, GLfloat emScale);
13624 GLAPI PFNGLPATHGLYPHRANGENVPROC glad_glPathGlyphRangeNV;
13625 #define glPathGlyphRangeNV glad_glPathGlyphRangeNV
13626 typedef void (APIENTRY PFNGLWEIGHTPATHSNVPROC) (GLuint resultPath, GLsizei numPaths, const GLuint
    *paths, const GLfloat *weights);
13627 GLAPI PFNGLWEIGHTPATHSNVPROC glad_glWeightPathsNV;
13628 #define glWeightPathsNV glad_glWeightPathsNV
13629 typedef void (APIENTRY PFNGLCOPYPATHNVPROC) (GLuint resultPath, GLuint srcPath);
13630 GLAPI PFNGLCOPYPATHNVPROC glad_glCopyPathNV;
13631 #define glCopyPathNV glad_glCopyPathNV
13632 typedef void (APIENTRY PFNGLINTERPOLATEPATHSNVPROC) (GLuint resultPath, GLuint pathA, GLuint pathB,
    GLfloat weight);
13633 GLAPI PFNGLINTERPOLATEPATHSNVPROC glad_glInterpolatePathsNV;
13634 #define glInterpolatePathsNV glad_glInterpolatePathsNV
13635 typedef void (APIENTRY PFNGLTRANSFORMPATHNVPROC) (GLuint resultPath, GLuint srcPath, GLenum
    transformType, const GLfloat *transformValues);
13636 GLAPI PFNGLTRANSFORMPATHNVPROC glad_glTransformPathNV;
13637 #define glTransformPathNV glad_glTransformPathNV
13638 typedef void (APIENTRY PFNGLPATHPARAMETERIVNVPROC) (GLuint path, GLenum pname, const GLint *value);
13639 GLAPI PFNGLPATHPARAMETERIVNVPROC glad_glPathParameterivNV;
13640 #define glPathParameterivNV glad_glPathParameterivNV
13641 typedef void (APIENTRY PFNGLPATHPARAMETERINVPROC) (GLuint path, GLenum pname, GLint value);
13642 GLAPI PFNGLPATHPARAMETERINVPROC glad_glPathParameteriNV;
13643 #define glPathParameteriNV glad_glPathParameteriNV
13644 typedef void (APIENTRY PFNGLPATHPARAMETERFVNVPROC) (GLuint path, GLenum pname, const GLfloat *value);
13645 GLAPI PFNGLPATHPARAMETERFVNVPROC glad_glPathParameterfvNV;
13646 #define glPathParameterfvNV glad_glPathParameterfvNV
13647 typedef void (APIENTRY PFNGLPATHPARAMETERFNVPROC) (GLuint path, GLenum pname, GLfloat value);
13648 GLAPI PFNGLPATHPARAMETERFNVPROC glad_glPathParameterfNV;
13649 #define glPathParameterfNV glad_glPathParameterfNV
13650 typedef void (APIENTRY PFNGLPATHDASHARRAYNVPROC) (GLuint path, GLsizei dashCount, const GLfloat
    *dashArray);
13651 GLAPI PFNGLPATHDASHARRAYNVPROC glad_glPathDashArrayNV;
13652 #define glPathDashArrayNV glad_glPathDashArrayNV
13653 typedef void (APIENTRY PFNGLPATHSTENCILFUNCNVPROC) (GLenum func, GLint ref, GLuint mask);
13654 GLAPI PFNGLPATHSTENCILFUNCNVPROC glad_glPathStencilFuncNV;
13655 #define glPathStencilFuncNV glad_glPathStencilFuncNV
13656 typedef void (APIENTRY PFNGLPATHSTENCILDEPTHOFFSETNVPROC) (GLfloat factor, GLfloat units);
13657 GLAPI PFNGLPATHSTENCILDEPTHOFFSETNVPROC glad_glPathStencilDepthOffsetNV;
13658 #define glPathStencilDepthOffsetNV glad_glPathStencilDepthOffsetNV
13659 typedef void (APIENTRY PFNGLSTENCILFILLPATHNVPROC) (GLuint path, GLenum fillMode, GLuint mask);
13660 GLAPI PFNGLSTENCILFILLPATHNVPROC glad_glStencilFillPathNV;
13661 #define glStencilFillPathNV glad_glStencilFillPathNV
13662 typedef void (APIENTRY PFNGLSTENCILSTROKEPATHNVPROC) (GLuint path, GLint reference, GLuint mask);
13663 GLAPI PFNGLSTENCILSTROKEPATHNVPROC glad_glStencilStrokePathNV;
13664 #define glStencilStrokePathNV glad_glStencilStrokePathNV
13665 typedef void (APIENTRY PFNGLSTENCILFILLPATHINSTANCEDNVPROC) (GLsizei numPaths, GLenum pathNameType,
    const void *paths, GLuint pathBase, GLenum fillMode, GLuint mask, GLenum transformType, const GLfloat
    *transformValues);
13666 GLAPI PFNGLSTENCILFILLPATHINSTANCEDNVPROC glad_glStencilFillPathInstancedNV;
13667 #define glStencilFillPathInstancedNV glad_glStencilFillPathInstancedNV
13668 typedef void (APIENTRY PFNGLSTENCILSTROKEPATHINSTANCEDNVPROC) (GLsizei numPaths, GLenum pathNameType,
    const void *paths, GLuint pathBase, GLint reference, GLuint mask, GLenum transformType, const GLfloat
    *transformValues);
13669 GLAPI PFNGLSTENCILSTROKEPATHINSTANCEDNVPROC glad_glStencilStrokePathInstancedNV;
13670 #define glStencilStrokePathInstancedNV glad_glStencilStrokePathInstancedNV
13671 typedef void (APIENTRY PFNGLPATHCOVERDEPTHFUNCNVPROC) (GLenum func);
13672 GLAPI PFNGLPATHCOVERDEPTHFUNCNVPROC glad_glPathCoverDepthFuncNV;
13673 #define glPathCoverDepthFuncNV glad_glPathCoverDepthFuncNV
13674 typedef void (APIENTRY PFNGLCOVERFILLPATHNVPROC) (GLuint path, GLenum coverMode);
13675 GLAPI PFNGLCOVERFILLPATHNVPROC glad_glCoverFillPathNV;
13676 #define glCoverFillPathNV glad_glCoverFillPathNV
13677 typedef void (APIENTRY PFNGLCOVERSTROKEPATHNVPROC) (GLuint path, GLenum coverMode);
13678 GLAPI PFNGLCOVERSTROKEPATHNVPROC glad_glCoverStrokePathNV;
13679 #define glCoverStrokePathNV glad_glCoverStrokePathNV
13680 typedef void (APIENTRY PFNGLCOVERFILLPATHINSTANCEDNVPROC) (GLsizei numPaths, GLenum pathNameType,
    const void *paths, GLuint pathBase, GLenum coverMode, GLenum transformType, const GLfloat
    *transformValues);
13681 GLAPI PFNGLCOVERFILLPATHINSTANCEDNVPROC glad_glCoverFillPathInstancedNV;
13682 #define glCoverFillPathInstancedNV glad_glCoverFillPathInstancedNV
13683 typedef void (APIENTRY PFNGLCOVERSTROKEPATHINSTANCEDNVPROC) (GLsizei numPaths, GLenum pathNameType,
    const void *paths, GLuint pathBase, GLenum coverMode, GLenum transformType, const GLfloat
    *transformValues);
13684 GLAPI PFNGLCOVERSTROKEPATHINSTANCEDNVPROC glad_glCoverStrokePathInstancedNV;
13685 #define glCoverStrokePathInstancedNV glad_glCoverStrokePathInstancedNV
13686 typedef void (APIENTRY PFNGLGETPATHPARAMETERIVNVPROC) (GLuint path, GLenum pname, GLint *value);
13687 GLAPI PFNGLGETPATHPARAMETERIVNVPROC glad_glGetPathParameterivNV;
13688 #define glGetPathParameterivNV glad_glGetPathParameterivNV
13689 typedef void (APIENTRY PFNGLGETPATHPARAMETERFVNVPROC) (GLuint path, GLenum pname, GLfloat *value);
13690 GLAPI PFNGLGETPATHPARAMETERFVNVPROC glad_glGetPathParameterfvNV;
13691 #define glGetPathParameterfvNV glad_glGetPathParameterfvNV
13692 typedef void (APIENTRY PFNGLGETPATHCOMMANDSNVPROC) (GLuint path, GLubyte *commands);
13693 GLAPI PFNGLGETPATHCOMMANDSNVPROC glad_glGetPathCommandsNV;

```



```

13694 #define glGetPathCommandsNV glad_glGetPathCommandsNV
13695 typedef void (APIENTRY PFNGLGETPATHCOORDSNVPROC) (GLuint path, GLfloat *coords);
13696 GLAPI PFNGLGETPATHCOORDSNVPROC glad_glGetPathCoordsNV;
13697 #define glGetPathCoordsNV glad_glGetPathCoordsNV
13698 typedef void (APIENTRY PFNGLGETPATHDASHARRAYNVPROC) (GLuint path, GLfloat *dashArray);
13699 GLAPI PFNGLGETPATHDASHARRAYNVPROC glad_glGetPathDashArrayNV;
13700 #define glGetPathDashArrayNV glad_glGetPathDashArrayNV
13701 typedef void (APIENTRY PFNGLGETPATHMETRICSNVPROC) (GLbitfield metricQueryMask, GLsizei numPaths,
13702 GLenum pathNameType, const void *paths, GLuint pathBase, GLsizei stride, GLfloat *metrics);
13703 GLAPI PFNGLGETPATHMETRICSNVPROC glad_glGetPathMetricsNV;
13704 #define glGetPathMetricsNV glad_glGetPathMetricsNV
13705 typedef void (APIENTRY PFNGLGETPATHMETRICRANGENVPROC) (GLbitfield metricQueryMask, GLuint
13706 firstPathName, GLsizei numPaths, GLsizei stride, GLfloat *metrics);
13707 GLAPI PFNGLGETPATHMETRICRANGENVPROC glad_glGetPathMetricRangeNV;
13708 #define glGetPathMetricRangeNV glad_glGetPathMetricRangeNV
13709 typedef void (APIENTRY PFNGLGETPATHSPACINGNVPROC) (GLenum pathListMode, GLsizei numPaths, GLenum
13710 pathNameType, const void *paths, GLuint pathBase, GLfloat advanceScale, GLfloat kerningScale, GLenum
13711 transformType, GLfloat *returnedSpacing);
13712 GLAPI PFNGLGETPATHSPACINGNVPROC glad_glGetPathSpacingNV;
13713 #define glGetPathSpacingNV glad_glGetPathSpacingNV
13714 typedef GLboolean (APIENTRY PFNGLISPOINTINFILLPATHNVPROC) (GLuint path, GLuint mask, GLfloat x,
13715 GLfloat y);
13716 GLAPI PFNGLISPOINTINFILLPATHNVPROC glad_glIsPointInFillPathNV;
13717 #define glGetIsPointInFillPathNV glad_glIsPointInFillPathNV
13718 typedef GLboolean (APIENTRY PFNGLISPOINTINSTROKEPATHNVPROC) (GLuint path, GLfloat x, GLfloat y);
13719 GLAPI PFNGLISPOINTINSTROKEPATHNVPROC glad_glIsPointInStrokePathNV;
13720 #define glGetIsPointInStrokePathNV glad_glIsPointInStrokePathNV
13721 typedef GLfloat (APIENTRY PFNGLGETPATHLENGTHNVPROC) (GLuint path, GLsizei startSegment, GLsizei
13722 numSegments);
13723 GLAPI PFNGLGETPATHLENGTHNVPROC glad_glGetPathLengthNV;
13724 #define glGetPathLengthNV glad_glGetPathLengthNV
13725 typedef GLboolean (APIENTRY PFNGLPOINTALONGPATHNVPROC) (GLuint path, GLsizei startSegment, GLsizei
13726 numSegments, GLfloat distance, GLfloat *x, GLfloat *y, GLfloat *tangentX, GLfloat *tangentY);
13727 GLAPI PFNGLPOINTALONGPATHNVPROC glad_glPointAlongPathNV;
13728 #define glGetPointAlongPathNV glad_glPointAlongPathNV
13729 typedef void (APIENTRY PFNGLMATRIXLOAD3X2FNVPROC) (GLenum matrixMode, const GLfloat *m);
13730 GLAPI PFNGLMATRIXLOAD3X2FNVPROC glad_glMatrixLoad3x2fNV;
13731 #define glGetMatrixLoad3x2fNV glad_glMatrixLoad3x2fNV
13732 typedef void (APIENTRY PFNGLMATRIXLOAD3X3FNVPROC) (GLenum matrixMode, const GLfloat *m);
13733 GLAPI PFNGLMATRIXLOAD3X3FNVPROC glad_glMatrixLoad3x3fNV;
13734 #define glGetMatrixLoad3x3fNV glad_glMatrixLoad3x3fNV
13735 typedef void (APIENTRY PFNGLMATRIXLOADTRANSPOSE3X3FNVPROC) (GLenum matrixMode, const GLfloat *m);
13736 GLAPI PFNGLMATRIXLOADTRANSPOSE3X3FNVPROC glad_glMatrixLoadTranspose3x3fNV;
13737 #define glGetMatrixLoadTranspose3x3fNV glad_glMatrixLoadTranspose3x3fNV
13738 typedef void (APIENTRY PFNGLMATRIXMULT3X2FNVPROC) (GLenum matrixMode, const GLfloat *m);
13739 GLAPI PFNGLMATRIXMULT3X2FNVPROC glad_glMatrixMult3x2fNV;
13740 #define glGetMatrixMult3x2fNV glad_glMatrixMult3x2fNV
13741 typedef void (APIENTRY PFNGLMATRIXMULT3X3FNVPROC) (GLenum matrixMode, const GLfloat *m);
13742 GLAPI PFNGLMATRIXMULT3X3FNVPROC glad_glMatrixMult3x3fNV;
13743 #define glGetMatrixMult3x3fNV glad_glMatrixMult3x3fNV
13744 typedef void (APIENTRY PFNGLMATRIXMULTTRANSPOSE3X3FNVPROC) (GLenum matrixMode, const GLfloat *m);
13745 GLAPI PFNGLMATRIXMULTTRANSPOSE3X3FNVPROC glad_glMatrixMultTranspose3x3fNV;
13746 #define glGetMatrixMultTranspose3x3fNV glad_glMatrixMultTranspose3x3fNV
13747 typedef void (APIENTRY PFNGLSTENCILTHENCOVERFILLPATHNVPROC) (GLuint path, GLenum fillMode, GLuint
13748 mask, GLenum coverMode);
13749 GLAPI PFNGLSTENCILTHENCOVERFILLPATHNVPROC glad_glStencilThenCoverFillPathNV;
13750 #define glGetStencilThenCoverFillPathNV glad_glStencilThenCoverFillPathNV
13751 typedef void (APIENTRY PFNGLSTENCILTHENCOVERSTROKEPATHNVPROC) (GLuint path, GLint reference, GLuint
13752 mask, GLenum coverMode);
13753 GLAPI PFNGLSTENCILTHENCOVERSTROKEPATHNVPROC glad_glStencilThenCoverStrokePathNV;
13754 #define glGetStencilThenCoverStrokePathNV glad_glStencilThenCoverStrokePathNV
13755 typedef void (APIENTRY PFNGLSTENCILTHENCOVERFILLPATHINSTANCEDNVPROC) (GLsizei numPaths, GLenum
13756 pathNameType, const void *paths, GLuint pathBase, GLenum fillMode, GLuint mask, GLenum coverMode,
13757 GLenum transformType, const GLfloat *transformValues);
13758 GLAPI PFNGLSTENCILTHENCOVERFILLPATHINSTANCEDNVPROC glad_glStencilThenCoverFillPathInstancedNV;
13759 #define glGetStencilThenCoverFillPathInstancedNV glad_glStencilThenCoverFillPathInstancedNV
13760 typedef void (APIENTRY PFNGLSTENCILTHENCOVERSTROKEPATHINSTANCEDNVPROC) (GLsizei numPaths, GLenum
13761 pathNameType, const void *paths, GLuint pathBase, GLint reference, GLuint mask, GLenum coverMode,
13762 GLenum transformType, const GLfloat *transformValues);
13763 GLAPI PFNGLSTENCILTHENCOVERSTROKEPATHINSTANCEDNVPROC glad_glStencilThenCoverStrokePathInstancedNV;
13764 #define glGetStencilThenCoverStrokePathInstancedNV glad_glStencilThenCoverStrokePathInstancedNV
13765 typedef void (APIENTRY PFNGLPATHGLYPHINDEXRANGENVPROC) (GLenum fontTarget, const void *fontName,
13766 GLbitfield fontStyle, GLuint pathParameterTemplate, GLfloat emScale, GLuint *baseAndCount);
13767 GLAPI PFNGLPATHGLYPHINDEXRANGENVPROC glad_glPathGlyphIndexRangeNV;
13768 #define glGetPathGlyphIndexRangeNV glad_glPathGlyphIndexRangeNV
13769 typedef GLenum (APIENTRY PFNGLPATHGLYPHINDEXARRAYNVPROC) (GLuint firstPathName, GLenum fontTarget,
13770 const void *fontName, GLbitfield fontStyle, GLuint firstGlyphIndex, GLsizei numGlyphs, GLuint
13771 pathParameterTemplate, GLfloat emScale);
13772 GLAPI PFNGLPATHGLYPHINDEXARRAYNVPROC glad_glPathGlyphIndexArrayNV;
13773 #define glGetPathGlyphIndexArrayNV glad_glPathGlyphIndexArrayNV
13774 typedef void (APIENTRY PFNGLPATHMEMORYGLYPHINDEXARRAYNVPROC) (GLuint firstPathName, GLenum
13775 fontTarget, GLsizei ptr fontSize, const void *fontData, GLsizei faceIndex, GLuint firstGlyphIndex,
13776 GLsizei numGlyphs, GLuint pathParameterTemplate, GLfloat emScale);
13777 GLAPI PFNGLPATHMEMORYGLYPHINDEXARRAYNVPROC glad_glPathMemoryGlyphIndexArrayNV;
13778 #define glGetPathMemoryGlyphIndexArrayNV glad_glPathMemoryGlyphIndexArrayNV
13779 typedef void (APIENTRY PFNGLPROGRAMPATHFRAGMENTINPUTGENNVPROC) (GLuint program, GLint location, GLenum
13780 genMode, GLint components, const GLfloat *coeffs);

```

```

13762 GLAPI PFNGLPROGRAMPATHFRAGMENTINPUTGENVPROC glad_glProgramPathFragmentInputGenNV;
13763 #define glProgramPathFragmentInputGenNV glad_glProgramPathFragmentInputGenNV
13764 typedef void (APIENTRY PFNGLPROGRAMRESOURCEFVNVPROC) (GLuint program, GLenum programInterface,
    GLuint index, GLsizei propCount, const GLenum *props, GLsizei count, GLsizei *length, GLfloat
    *params);
13765 GLAPI PFNGLGETPROGRAMRESOURCEFVNVPROC glad_glGetProgramResourcefvNV;
13766 #define glGetProgramResourcefvNV glad_glGetProgramResourcefvNV
13767 typedef void (APIENTRY PFNGLPATHCOLORGENVPROC) (GLenum color, GLenum genMode, GLenum colorFormat,
    const GLfloat *coeffs);
13768 GLAPI PFNGLPATHCOLORGENVPROC glad_glPathColorGenNV;
13769 #define glPathColorGenNV glad_glPathColorGenNV
13770 typedef void (APIENTRY PFNGLPATHTEXGENVPROC) (GLenum texCoordSet, GLenum genMode, GLint components,
    const GLfloat *coeffs);
13771 GLAPI PFNGLPATHTEXGENVPROC glad_glPathTexGenNV;
13772 #define glPathTexGenNV glad_glPathTexGenNV
13773 typedef void (APIENTRY PFNGLPATHFOGGENVPROC) (GLenum genMode);
13774 GLAPI PFNGLPATHFOGGENVPROC glad_glPathFogGenNV;
13775 #define glPathFogGenNV glad_glPathFogGenNV
13776 typedef void (APIENTRY PFNGLGETPATHCOLORGENIVNVPROC) (GLenum color, GLenum pname, GLint *value);
13777 GLAPI PFNGLGETPATHCOLORGENIVNVPROC glad_glGetPathColorGenivNV;
13778 #define glGetPathColorGenivNV glad_glGetPathColorGenivNV
13779 typedef void (APIENTRY PFNGLGETPATHCOLORGENFVNVPROC) (GLenum color, GLenum pname, GLfloat *value);
13780 GLAPI PFNGLGETPATHCOLORGENFVNVPROC glad_glGetPathColorGenfvNV;
13781 #define glGetPathColorGenfvNV glad_glGetPathColorGenfvNV
13782 typedef void (APIENTRY PFNGLGETPATHTEXGENIVNVPROC) (GLenum texCoordSet, GLenum pname, GLint *value);
13783 GLAPI PFNGLGETPATHTEXGENIVNVPROC glad_glGetPathTexGenivNV;
13784 #define glGetPathTexGenivNV glad_glGetPathTexGenivNV
13785 typedef void (APIENTRY PFNGLGETPATHTEXGENFVNVPROC) (GLenum texCoordSet, GLenum pname, GLfloat *value);
13786 GLAPI PFNGLGETPATHTEXGENFVNVPROC glad_glGetPathTexGenfvNV;
13787 #define glGetPathTexGenfvNV glad_glGetPathTexGenfvNV
13788 #endif
13789 #ifndef GL_NV_path_rendering_shared_edge
13790 #define GL_NV_path_rendering_shared_edge 1
13791 GLAPI int GLAD_GL_NV_path_rendering_shared_edge;
13792 #endif
13793 #ifndef GL_NV_pixel_data_range
13794 #define GL_NV_pixel_data_range 1
13795 GLAPI int GLAD_GL_NV_pixel_data_range;
13796 typedef void (APIENTRY PFNGLPIXELDATARANGENVPROC) (GLenum target, GLsizei length, const void
    *pointer);
13797 GLAPI PFNGLPIXELDATARANGENVPROC glad_glPixelDataRangeNV;
13798 #define glPixelDataRangeNV glad_glPixelDataRangeNV
13799 typedef void (APIENTRY PFNGLFLUSHPIXELDATARANGENVPROC) (GLenum target);
13800 GLAPI PFNGLFLUSHPIXELDATARANGENVPROC glad_glFlushPixelDataRangeNV;
13801 #define glFlushPixelDataRangeNV glad_glFlushPixelDataRangeNV
13802 #endif
13803 #ifndef GL_NV_point_sprite
13804 #define GL_NV_point_sprite 1
13805 GLAPI int GLAD_GL_NV_point_sprite;
13806 typedef void (APIENTRY PFNGLPOINTPARAMETERINVPROC) (GLenum pname, GLint param);
13807 GLAPI PFNGLPOINTPARAMETERINVPROC glad_glPointParameteriNV;
13808 #define glPointParameteriNV glad_glPointParameteriNV
13809 typedef void (APIENTRY PFNGLPOINTPARAMETERIVNVPROC) (GLenum pname, const GLint *params);
13810 GLAPI PFNGLPOINTPARAMETERIVNVPROC glad_glPointParameterivNV;
13811 #define glPointParameterivNV glad_glPointParameterivNV
13812 #endif
13813 #ifndef GL_NV_present_video
13814 #define GL_NV_present_video 1
13815 GLAPI int GLAD_GL_NV_present_video;
13816 typedef void (APIENTRY PFNGLPRESENTFRAMEKEYEDNVPROC) (GLuint video_slot, GLuint64EXT minPresentTime,
    GLuint beginPresentTimeId, GLuint presentDurationId, GLenum type, GLenum target0, GLuint fill0, GLuint
    key0, GLenum target1, GLuint fill1, GLuint key1);
13817 GLAPI PFNGLPRESENTFRAMEKEYEDNVPROC glad_glPresentFrameKeyedNV;
13818 #define glPresentFrameKeyedNV glad_glPresentFrameKeyedNV
13819 typedef void (APIENTRY PFNGLPRESENTFRAMEDUALFILLNVPROC) (GLuint video_slot, GLuint64EXT
    minPresentTime, GLuint beginPresentTimeId, GLuint presentDurationId, GLenum type, GLenum target0,
    GLuint fill0, GLenum target1, GLuint fill1, GLenum target2, GLuint fill2, GLenum target3, GLuint
    fill3);
13820 GLAPI PFNGLPRESENTFRAMEDUALFILLNVPROC glad_glPresentFrameDualFillNV;
13821 #define glPresentFrameDualFillNV glad_glPresentFrameDualFillNV
13822 typedef void (APIENTRY PFNGLGETVIDEOIIVNVPROC) (GLuint video_slot, GLenum pname, GLint *params);
13823 GLAPI PFNGLGETVIDEOIIVNVPROC glad_glGetVideoiivNV;
13824 #define glGetVideoiivNV glad_glGetVideoiivNV
13825 typedef void (APIENTRY PFNGLGETVIDEOUIVNVPROC) (GLuint video_slot, GLenum pname, GLuint *params);
13826 GLAPI PFNGLGETVIDEOUIVNVPROC glad_glGetVideoiivNV;
13827 #define glGetVideoiivNV glad_glGetVideoiivNV
13828 typedef void (APIENTRY PFNGLGETVIDEOI64VNVPROC) (GLuint video_slot, GLenum pname, GLint64EXT *params);
13829 GLAPI PFNGLGETVIDEOI64VNVPROC glad_glGetVideoi64vNV;
13830 #define glGetVideoi64vNV glad_glGetVideoi64vNV
13831 typedef void (APIENTRY PFNGLGETVIDEOUI64VNVPROC) (GLuint video_slot, GLenum pname, GLuint64EXT
    *params);
13832 GLAPI PFNGLGETVIDEOUI64VNVPROC glad_glGetVideoi64vNV;
13833 #define glGetVideoi64vNV glad_glGetVideoi64vNV
13834 #endif
13835 #ifndef GL_NV_primitive_restart
13836 #define GL_NV_primitive_restart 1
13837 GLAPI int GLAD_GL_NV_primitive_restart;

```

```

13838 typedef void (APIENTRY PFNGLPRIMITIVERESTARTNVPROC)(void);
13839 GLAPI PFNGLPRIMITIVERESTARTNVPROC glad_glPrimitiveRestartNV;
13840 #define glPrimitiveRestartNV glad_glPrimitiveRestartNV
13841 typedef void (APIENTRY PFNGLPRIMITIVERESTARTINDEXNVPROC)(GLuint index);
13842 GLAPI PFNGLPRIMITIVERESTARTINDEXNVPROC glad_glPrimitiveRestartIndexNV;
13843 #define glPrimitiveRestartIndexNV glad_glPrimitiveRestartIndexNV
13844 #endif
13845 #ifndef GL_NV_primitive_shading_rate
13846 #define GL_NV_primitive_shading_rate 1
13847 GLAPI int GLAD_GL_NV_primitive_shading_rate;
13848 #endif
13849 #ifndef GL_NV_query_resource
13850 #define GL_NV_query_resource 1
13851 GLAPI int GLAD_GL_NV_query_resource;
13852 typedef GLint (APIENTRY PFNGLQUERYRESOURCENVPROC)(GLenum queryType, GLint tagId, GLuint count, GLint
*buffer);
13853 GLAPI PFNGLQUERYRESOURCENVPROC glad_glQueryResourceNV;
13854 #define glQueryResourceNV glad_glQueryResourceNV
13855 #endif
13856 #ifndef GL_NV_query_resource_tag
13857 #define GL_NV_query_resource_tag 1
13858 GLAPI int GLAD_GL_NV_query_resource_tag;
13859 typedef void (APIENTRY PFNGLGENQUERYRESOURCETAGNVPROC)(GLsizei n, GLint *tagIds);
13860 GLAPI PFNGLGENQUERYRESOURCETAGNVPROC glad_glGenQueryResourceTagNV;
13861 #define glGenQueryResourceTagNV glad_glGenQueryResourceTagNV
13862 typedef void (APIENTRY PFNGLDELETEQUERYRESOURCETAGNVPROC)(GLsizei n, const GLint *tagIds);
13863 GLAPI PFNGLDELETEQUERYRESOURCETAGNVPROC glad_glDeleteQueryResourceTagNV;
13864 #define glDeleteQueryResourceTagNV glad_glDeleteQueryResourceTagNV
13865 typedef void (APIENTRY PFNGLQUERYRESOURCETAGNVPROC)(GLint tagId, const GLchar *tagString);
13866 GLAPI PFNGLQUERYRESOURCETAGNVPROC glad_glQueryResourceTagNV;
13867 #define glQueryResourceTagNV glad_glQueryResourceTagNV
13868 #endif
13869 #ifndef GL_NV_register_combiners
13870 #define GL_NV_register_combiners 1
13871 GLAPI int GLAD_GL_NV_register_combiners;
13872 typedef void (APIENTRY PFNGLCOMBINERPARAMETERFNVPROC)(GLenum pname, const GLfloat *params);
13873 GLAPI PFNGLCOMBINERPARAMETERFNVPROC glad_glCombinerParameterfvNV;
13874 #define glCombinerParameterfvNV glad_glCombinerParameterfvNV
13875 typedef void (APIENTRY PFNGLCOMBINERPARAMETERFNVPROC)(GLenum pname, GLfloat param);
13876 GLAPI PFNGLCOMBINERPARAMETERFNVPROC glad_glCombinerParameterfvNV;
13877 #define glCombinerParameterfvNV glad_glCombinerParameterfvNV
13878 typedef void (APIENTRY PFNGLCOMBINERPARAMETERIVNVPROC)(GLenum pname, const GLint *params);
13879 GLAPI PFNGLCOMBINERPARAMETERIVNVPROC glad_glCombinerParameterivNV;
13880 #define glCombinerParameterivNV glad_glCombinerParameterivNV
13881 typedef void (APIENTRY PFNGLCOMBINERPARAMETERINNVPROC)(GLenum pname, GLint param);
13882 GLAPI PFNGLCOMBINERPARAMETERINNVPROC glad_glCombinerParameterivNV;
13883 #define glCombinerParameterivNV glad_glCombinerParameterivNV
13884 typedef void (APIENTRY PFNGLCOMBINERINPUTNVPROC)(GLenum stage, GLenum portion, GLenum variable,
GLenum input, GLenum mapping, GLenum componentUsage);
13885 GLAPI PFNGLCOMBINERINPUTNVPROC glad_glCombinerInputNV;
13886 #define glCombinerInputNV glad_glCombinerInputNV
13887 typedef void (APIENTRY PFNGLCOMBINEROUTPUTNVPROC)(GLenum stage, GLenum portion, GLenum abOutput,
GLenum cdOutput, GLenum sumOutput, GLenum scale, GLenum bias, GLboolean abDotProduct, GLboolean
cdDotProduct, GLboolean muxSum);
13888 GLAPI PFNGLCOMBINEROUTPUTNVPROC glad_glCombinerOutputNV;
13889 #define glCombinerOutputNV glad_glCombinerOutputNV
13890 typedef void (APIENTRY PFNGLFINALCOMBINERINPUTNVPROC)(GLenum variable, GLenum input, GLenum mapping,
GLenum componentUsage);
13891 GLAPI PFNGLFINALCOMBINERINPUTNVPROC glad_glFinalCombinerInputNV;
13892 #define glFinalCombinerInputNV glad_glFinalCombinerInputNV
13893 typedef void (APIENTRY PFNGLGETCOMBINERINPUTPARAMETERFNVPROC)(GLenum stage, GLenum portion, GLenum
variable, GLenum pname, GLfloat *params);
13894 GLAPI PFNGLGETCOMBINERINPUTPARAMETERFNVPROC glad_glGetCombinerInputParameterfvNV;
13895 #define glGetCombinerInputParameterfvNV glad_glGetCombinerInputParameterfvNV
13896 typedef void (APIENTRY PFNGLGETCOMBINERINPUTPARAMETERIVNVPROC)(GLenum stage, GLenum portion, GLenum
variable, GLenum pname, GLint *params);
13897 GLAPI PFNGLGETCOMBINERINPUTPARAMETERIVNVPROC glad_glGetCombinerInputParameterivNV;
13898 #define glGetCombinerInputParameterivNV glad_glGetCombinerInputParameterivNV
13899 typedef void (APIENTRY PFNGLGETCOMBINEROUTPUTPARAMETERFNVPROC)(GLenum stage, GLenum portion, GLenum
pname, GLfloat *params);
13900 GLAPI PFNGLGETCOMBINEROUTPUTPARAMETERFNVPROC glad_glGetCombinerOutputParameterfvNV;
13901 #define glGetCombinerOutputParameterfvNV glad_glGetCombinerOutputParameterfvNV
13902 typedef void (APIENTRY PFNGLGETCOMBINEROUTPUTPARAMETERIVNVPROC)(GLenum stage, GLenum portion, GLenum
pname, GLint *params);
13903 GLAPI PFNGLGETCOMBINEROUTPUTPARAMETERIVNVPROC glad_glGetCombinerOutputParameterivNV;
13904 #define glGetCombinerOutputParameterivNV glad_glGetCombinerOutputParameterivNV
13905 typedef void (APIENTRY PFNGLGETFINALCOMBINERINPUTPARAMETERFNVPROC)(GLenum variable, GLenum pname,
GLfloat *params);
13906 GLAPI PFNGLGETFINALCOMBINERINPUTPARAMETERFNVPROC glad_glGetFinalCombinerInputParameterfvNV;
13907 #define glGetFinalCombinerInputParameterfvNV glad_glGetFinalCombinerInputParameterfvNV
13908 typedef void (APIENTRY PFNGLGETFINALCOMBINERINPUTPARAMETERIVNVPROC)(GLenum variable, GLenum pname,
GLint *params);
13909 GLAPI PFNGLGETFINALCOMBINERINPUTPARAMETERIVNVPROC glad_glGetFinalCombinerInputParameterivNV;
13910 #define glGetFinalCombinerInputParameterivNV glad_glGetFinalCombinerInputParameterivNV
13911 #endif
13912 #ifndef GL_NV_register_combiners2
13913 #define GL_NV_register_combiners2 1

```

```

13914 GLAPI int GLAD_GL_NV_register_combiners2;
13915 typedef void (APIENTRY PFNGLCOMBINERSTAGEPARAMETERFVNVPROC) (GLenum stage, GLenum pname, const GLfloat
    *params);
13916 GLAPI PFNGLCOMBINERSTAGEPARAMETERFVNVPROC glad_glCombinerStageParameterfvNV;
13917 #define glCombinerStageParameterfvNV glad_glCombinerStageParameterfvNV
13918 typedef void (APIENTRY PFNGLGETCOMBINERSTAGEPARAMETERFVNVPROC) (GLenum stage, GLenum pname, GLfloat
    *params);
13919 GLAPI PFNGLGETCOMBINERSTAGEPARAMETERFVNVPROC glad_glGetCombinerStageParameterfvNV;
13920 #define glGetCombinerStageParameterfvNV glad_glGetCombinerStageParameterfvNV
13921 #endif
13922 #ifndef GL_NV_representative_fragment_test
13923 #define GL_NV_representative_fragment_test 1
13924 GLAPI int GLAD_GL_NV_representative_fragment_test;
13925 #endif
13926 #ifndef GL_NV_robustness_video_memory_purge
13927 #define GL_NV_robustness_video_memory_purge 1
13928 GLAPI int GLAD_GL_NV_robustness_video_memory_purge;
13929 #endif
13930 #ifndef GL_NV_sample_locations
13931 #define GL_NV_sample_locations 1
13932 GLAPI int GLAD_GL_NV_sample_locations;
13933 typedef void (APIENTRY PFNGLFRAMEBUFFERSAMPLELOCATIONSFVNVPROC) (GLenum target, GLuint start, GLsizei
    count, const GLfloat *v);
13934 GLAPI PFNGLFRAMEBUFFERSAMPLELOCATIONSFVNVPROC glad_glFramebufferSampleLocationsfvNV;
13935 #define glFramebufferSampleLocationsfvNV glad_glFramebufferSampleLocationsfvNV
13936 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVNVPROC) (GLuint framebuffer, GLuint
    start, GLsizei count, const GLfloat *v);
13937 GLAPI PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVNVPROC glad_glNamedFramebufferSampleLocationsfvNV;
13938 #define glNamedFramebufferSampleLocationsfvNV glad_glNamedFramebufferSampleLocationsfvNV
13939 typedef void (APIENTRY PFNGLRESOLVEDEPTHVALUESNVPROC) (void);
13940 GLAPI PFNGLRESOLVEDEPTHVALUESNVPROC glad_glResolveDepthValuesNV;
13941 #define glResolveDepthValuesNV glad_glResolveDepthValuesNV
13942 #endif
13943 #ifndef GL_NV_sample_mask_override_coverage
13944 #define GL_NV_sample_mask_override_coverage 1
13945 GLAPI int GLAD_GL_NV_sample_mask_override_coverage;
13946 #endif
13947 #ifndef GL_NV_scissor_exclusive
13948 #define GL_NV_scissor_exclusive 1
13949 GLAPI int GLAD_GL_NV_scissor_exclusive;
13950 typedef void (APIENTRY PFNGLSCISSOREXCLUSIVENVPROC) (GLint x, GLint y, GLsizei width, GLsizei height);
13951 GLAPI PFNGLSCISSOREXCLUSIVENVPROC glad_glScissorExclusiveNV;
13952 #define glScissorExclusiveNV glad_glScissorExclusiveNV
13953 typedef void (APIENTRY PFNGLSCISSOREXCLUSIVEARRAYVNVPROC) (GLuint first, GLsizei count, const GLint
    *v);
13954 GLAPI PFNGLSCISSOREXCLUSIVEARRAYVNVPROC glad_glScissorExclusiveArrayvNV;
13955 #define glScissorExclusiveArrayvNV glad_glScissorExclusiveArrayvNV
13956 #endif
13957 #ifndef GL_NV_shader_atomic_counters
13958 #define GL_NV_shader_atomic_counters 1
13959 GLAPI int GLAD_GL_NV_shader_atomic_counters;
13960 #endif
13961 #ifndef GL_NV_shader_atomic_float
13962 #define GL_NV_shader_atomic_float 1
13963 GLAPI int GLAD_GL_NV_shader_atomic_float;
13964 #endif
13965 #ifndef GL_NV_shader_atomic_float64
13966 #define GL_NV_shader_atomic_float64 1
13967 GLAPI int GLAD_GL_NV_shader_atomic_float64;
13968 #endif
13969 #ifndef GL_NV_shader_atomic_fp16_vector
13970 #define GL_NV_shader_atomic_fp16_vector 1
13971 GLAPI int GLAD_GL_NV_shader_atomic_fp16_vector;
13972 #endif
13973 #ifndef GL_NV_shader_atomic_int64
13974 #define GL_NV_shader_atomic_int64 1
13975 GLAPI int GLAD_GL_NV_shader_atomic_int64;
13976 #endif
13977 #ifndef GL_NV_shader_buffer_load
13978 #define GL_NV_shader_buffer_load 1
13979 GLAPI int GLAD_GL_NV_shader_buffer_load;
13980 typedef void (APIENTRY PFNGLMAKEBUFFERRESIDENTNVPROC) (GLenum target, GLenum access);
13981 GLAPI PFNGLMAKEBUFFERRESIDENTNVPROC glad_glMakeBufferResidentNV;
13982 #define glMakeBufferResidentNV glad_glMakeBufferResidentNV
13983 typedef void (APIENTRY PFNGLMAKEBUFFERNONRESIDENTNVPROC) (GLenum target);
13984 GLAPI PFNGLMAKEBUFFERNONRESIDENTNVPROC glad_glMakeBufferNonResidentNV;
13985 #define glMakeBufferNonResidentNV glad_glMakeBufferNonResidentNV
13986 typedef GLboolean (APIENTRY PFNGLISBUFFERRESIDENTNVPROC) (GLenum target);
13987 GLAPI PFNGLISBUFFERRESIDENTNVPROC glad_glIsBufferResidentNV;
13988 #define glIsBufferResidentNV glad_glIsBufferResidentNV
13989 typedef void (APIENTRY PFNGLMAKENAMEDBUFFERRESIDENTNVPROC) (GLuint buffer, GLenum access);
13990 GLAPI PFNGLMAKENAMEDBUFFERRESIDENTNVPROC glad_glMakeNamedBufferResidentNV;
13991 #define glMakeNamedBufferResidentNV glad_glMakeNamedBufferResidentNV
13992 typedef void (APIENTRY PFNGLMAKENAMEDBUFFERNONRESIDENTNVPROC) (GLuint buffer);
13993 GLAPI PFNGLMAKENAMEDBUFFERNONRESIDENTNVPROC glad_glMakeNamedBufferNonResidentNV;
13994 #define glMakeNamedBufferNonResidentNV glad_glMakeNamedBufferNonResidentNV
13995 typedef GLboolean (APIENTRY PFNGLISNAMEDBUFFERRESIDENTNVPROC) (GLuint buffer);

```

```

13996 GLAPI PFNGLISNAMEDBUFFERRESIDENTNVPROC glad_glIsNamedBufferResidentNV;
13997 #define glIsNamedBufferResidentNV glad_glIsNamedBufferResidentNV
13998 typedef void (APIENTRY PFNGLGETBUFFERPARAMETERUI64VNVPROC) (GLenum target, GLenum pname, GLuint64EXT
    *params);
13999 GLAPI PFNGLGETBUFFERPARAMETERUI64VNVPROC glad_glGetBufferParameterui64vNV;
14000 #define glGetBufferParameterui64vNV glad_glGetBufferParameterui64vNV
14001 typedef void (APIENTRY PFNGLGETNAMEDBUFFERPARAMETERUI64VNVPROC) (GLuint buffer, GLenum pname,
    GLuint64EXT *params);
14002 GLAPI PFNGLGETNAMEDBUFFERPARAMETERUI64VNVPROC glad_glGetNamedBufferParameterui64vNV;
14003 #define glGetNamedBufferParameterui64vNV glad_glGetNamedBufferParameterui64vNV
14004 typedef void (APIENTRY PFNGLGETINTEGERUI64VNVPROC) (GLenum value, GLuint64EXT *result);
14005 GLAPI PFNGLGETINTEGERUI64VNVPROC glad_glGetIntegerui64vNV;
14006 #define glGetIntegerui64vNV glad_glGetIntegerui64vNV
14007 typedef void (APIENTRY PFNGLUNIFORMUI64VNVPROC) (GLint location, GLuint64EXT value);
14008 GLAPI PFNGLUNIFORMUI64VNVPROC glad_glUniformui64NV;
14009 #define glUniformui64NV glad_glUniformui64NV
14010 typedef void (APIENTRY PFNGLUNIFORMUI64VNVPROC) (GLint location, GLsizei count, const GLuint64EXT
    *value);
14011 GLAPI PFNGLUNIFORMUI64VNVPROC glad_glUniformui64vNV;
14012 #define glUniformui64vNV glad_glUniformui64vNV
14013 typedef void (APIENTRY PFNGLPROGRAMUNIFORMUI64VNVPROC) (GLuint program, GLint location, GLuint64EXT
    value);
14014 GLAPI PFNGLPROGRAMUNIFORMUI64VNVPROC glad_glProgramUniformui64NV;
14015 #define glProgramUniformui64NV glad_glProgramUniformui64NV
14016 typedef void (APIENTRY PFNGLPROGRAMUNIFORMUI64VNVPROC) (GLuint program, GLint location, GLsizei count,
    const GLuint64EXT *value);
14017 GLAPI PFNGLPROGRAMUNIFORMUI64VNVPROC glad_glProgramUniformui64vNV;
14018 #define glProgramUniformui64vNV glad_glProgramUniformui64vNV
14019 #endif
14020 #ifndef GL_NV_shader_buffer_store
14021 #define GL_NV_shader_buffer_store 1
14022 GLAPI int GLAD_GL_NV_shader_buffer_store;
14023 #endif
14024 #ifndef GL_NV_shader_storage_buffer_object
14025 #define GL_NV_shader_storage_buffer_object 1
14026 GLAPI int GLAD_GL_NV_shader_storage_buffer_object;
14027 #endif
14028 #ifndef GL_NV_shader_subgroup_partitioned
14029 #define GL_NV_shader_subgroup_partitioned 1
14030 GLAPI int GLAD_GL_NV_shader_subgroup_partitioned;
14031 #endif
14032 #ifndef GL_NV_shader_texture_footprint
14033 #define GL_NV_shader_texture_footprint 1
14034 GLAPI int GLAD_GL_NV_shader_texture_footprint;
14035 #endif
14036 #ifndef GL_NV_shader_thread_group
14037 #define GL_NV_shader_thread_group 1
14038 GLAPI int GLAD_GL_NV_shader_thread_group;
14039 #endif
14040 #ifndef GL_NV_shader_thread_shuffle
14041 #define GL_NV_shader_thread_shuffle 1
14042 GLAPI int GLAD_GL_NV_shader_thread_shuffle;
14043 #endif
14044 #ifndef GL_NV_shading_rate_image
14045 #define GL_NV_shading_rate_image 1
14046 GLAPI int GLAD_GL_NV_shading_rate_image;
14047 typedef void (APIENTRY PFNGLBINDSHADINGRATEIMAGENVPROC) (GLuint texture);
14048 GLAPI PFNGLBINDSHADINGRATEIMAGENVPROC glad_glBindShadingRateImageNV;
14049 #define glBindShadingRateImageNV glad_glBindShadingRateImageNV
14050 typedef void (APIENTRY PFNGLGETSHADINGRATEIMAGEPALETTEVNVPROC) (GLuint viewport, GLuint entry, GLenum
    *rate);
14051 GLAPI PFNGLGETSHADINGRATEIMAGEPALETTEVNVPROC glad_glGetShadingRateImagePaletteNV;
14052 #define glGetShadingRateImagePaletteNV glad_glGetShadingRateImagePaletteNV
14053 typedef void (APIENTRY PFNGLGETSHADINGRATESAMPLELOCATIONIVNVPROC) (GLenum rate, GLuint samples, GLuint
    index, GLint *location);
14054 GLAPI PFNGLGETSHADINGRATESAMPLELOCATIONIVNVPROC glad_glGetShadingRateSampleLocationivNV;
14055 #define glGetShadingRateSampleLocationivNV glad_glGetShadingRateSampleLocationivNV
14056 typedef void (APIENTRY PFNGLSHADINGRATEIMAGEBARRIERNVPROC) (GLboolean synchronize);
14057 GLAPI PFNGLSHADINGRATEIMAGEBARRIERNVPROC glad_glShadingRateImageBarrierNV;
14058 #define glShadingRateImageBarrierNV glad_glShadingRateImageBarrierNV
14059 typedef void (APIENTRY PFNGLSHADINGRATEIMAGEPALETTEVNVPROC) (GLuint viewport, GLuint first, GLsizei
    count, const GLenum *rates);
14060 GLAPI PFNGLSHADINGRATEIMAGEPALETTEVNVPROC glad_glShadingRateImagePaletteNV;
14061 #define glShadingRateImagePaletteNV glad_glShadingRateImagePaletteNV
14062 typedef void (APIENTRY PFNGLSHADINGRATESAMPLEORDERNVPROC) (GLenum order);
14063 GLAPI PFNGLSHADINGRATESAMPLEORDERNVPROC glad_glShadingRateSampleOrderNV;
14064 #define glShadingRateSampleOrderNV glad_glShadingRateSampleOrderNV
14065 typedef void (APIENTRY PFNGLSHADINGRATESAMPLEORDERCUSTOMNVPROC) (GLenum rate, GLuint samples, const
    GLint *locations);
14066 GLAPI PFNGLSHADINGRATESAMPLEORDERCUSTOMNVPROC glad_glShadingRateSampleOrderCustomNV;
14067 #define glShadingRateSampleOrderCustomNV glad_glShadingRateSampleOrderCustomNV
14068 #endif
14069 #ifndef GL_NV_stereo_view_rendering
14070 #define GL_NV_stereo_view_rendering 1
14071 GLAPI int GLAD_GL_NV_stereo_view_rendering;
14072 #endif
14073 #ifndef GL_NV_tessellation_program5

```



```

14074 #define GL_NV_tessellation_program5 1
14075 GLAPI int GLAD_GL_NV_tessellation_program5;
14076 #endif
14077 #ifndef GL_NV_texgen_emboss
14078 #define GL_NV_texgen_emboss 1
14079 GLAPI int GLAD_GL_NV_texgen_emboss;
14080 #endif
14081 #ifndef GL_NV_texgen_reflection
14082 #define GL_NV_texgen_reflection 1
14083 GLAPI int GLAD_GL_NV_texgen_reflection;
14084 #endif
14085 #ifndef GL_NV_texture_barrier
14086 #define GL_NV_texture_barrier 1
14087 GLAPI int GLAD_GL_NV_texture_barrier;
14088 typedef void (APIENTRY PFNGLTEXTUREBARRIERNVPROC) (void);
14089 GLAPI PFNGLTEXTUREBARRIERNVPROC glad_glTextureBarrierNV;
14090 #define glTextureBarrierNV glad_glTextureBarrierNV
14091 #endif
14092 #ifndef GL_NV_texture_compression_vtc
14093 #define GL_NV_texture_compression_vtc 1
14094 GLAPI int GLAD_GL_NV_texture_compression_vtc;
14095 #endif
14096 #ifndef GL_NV_texture_env_combine4
14097 #define GL_NV_texture_env_combine4 1
14098 GLAPI int GLAD_GL_NV_texture_env_combine4;
14099 #endif
14100 #ifndef GL_NV_texture_expand_normal
14101 #define GL_NV_texture_expand_normal 1
14102 GLAPI int GLAD_GL_NV_texture_expand_normal;
14103 #endif
14104 #ifndef GL_NV_texture_multisample
14105 #define GL_NV_texture_multisample 1
14106 GLAPI int GLAD_GL_NV_texture_multisample;
14107 typedef void (APIENTRY PFNGLTEXIMAGE2DMULTISAMPLECOVERAGENVPROC) (GLenum target, GLsizei
    coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height, GLboolean
    fixedSampleLocations);
14108 GLAPI PFNGLTEXIMAGE2DMULTISAMPLECOVERAGENVPROC glad_glTexImage2DMultisampleCoverageNV;
14109 #define glTexImage2DMultisampleCoverageNV glad_glTexImage2DMultisampleCoverageNV
14110 typedef void (APIENTRY PFNGLTEXIMAGE3DMULTISAMPLECOVERAGENVPROC) (GLenum target, GLsizei
    coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height, GLsizei
    depth, GLboolean fixedSampleLocations);
14111 GLAPI PFNGLTEXIMAGE3DMULTISAMPLECOVERAGENVPROC glad_glTexImage3DMultisampleCoverageNV;
14112 #define glTexImage3DMultisampleCoverageNV glad_glTexImage3DMultisampleCoverageNV
14113 typedef void (APIENTRY PFNGLTEXTUREIMAGE2DMULTISAMPLENVPROC) (GLuint texture, GLenum target, GLsizei
    samples, GLint internalFormat, GLsizei width, GLsizei height, GLboolean fixedSampleLocations);
14114 GLAPI PFNGLTEXTUREIMAGE2DMULTISAMPLENVPROC glad_glTextureImage2DMultisampleNV;
14115 #define glTextureImage2DMultisampleNV glad_glTextureImage2DMultisampleNV
14116 typedef void (APIENTRY PFNGLTEXTUREIMAGE3DMULTISAMPLENVPROC) (GLuint texture, GLenum target, GLsizei
    samples, GLint internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean
    fixedSampleLocations);
14117 GLAPI PFNGLTEXTUREIMAGE3DMULTISAMPLENVPROC glad_glTextureImage3DMultisampleNV;
14118 #define glTextureImage3DMultisampleNV glad_glTextureImage3DMultisampleNV
14119 typedef void (APIENTRY PFNGLTEXTUREIMAGE2DMULTISAMPLECOVERAGENVPROC) (GLuint texture, GLenum target,
    GLsizei coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height,
    GLboolean fixedSampleLocations);
14120 GLAPI PFNGLTEXTUREIMAGE2DMULTISAMPLECOVERAGENVPROC glad_glTextureImage2DMultisampleCoverageNV;
14121 #define glTextureImage2DMultisampleCoverageNV glad_glTextureImage2DMultisampleCoverageNV
14122 typedef void (APIENTRY PFNGLTEXTUREIMAGE3DMULTISAMPLECOVERAGENVPROC) (GLuint texture, GLenum target,
    GLsizei coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height,
    GLsizei depth, GLboolean fixedSampleLocations);
14123 GLAPI PFNGLTEXTUREIMAGE3DMULTISAMPLECOVERAGENVPROC glad_glTextureImage3DMultisampleCoverageNV;
14124 #define glTextureImage3DMultisampleCoverageNV glad_glTextureImage3DMultisampleCoverageNV
14125 #endif
14126 #ifndef GL_NV_texture_rectangle
14127 #define GL_NV_texture_rectangle 1
14128 GLAPI int GLAD_GL_NV_texture_rectangle;
14129 #endif
14130 #ifndef GL_NV_texture_rectangle_compressed
14131 #define GL_NV_texture_rectangle_compressed 1
14132 GLAPI int GLAD_GL_NV_texture_rectangle_compressed;
14133 #endif
14134 #ifndef GL_NV_texture_shader
14135 #define GL_NV_texture_shader 1
14136 GLAPI int GLAD_GL_NV_texture_shader;
14137 #endif
14138 #ifndef GL_NV_texture_shader2
14139 #define GL_NV_texture_shader2 1
14140 GLAPI int GLAD_GL_NV_texture_shader2;
14141 #endif
14142 #ifndef GL_NV_texture_shader3
14143 #define GL_NV_texture_shader3 1
14144 GLAPI int GLAD_GL_NV_texture_shader3;
14145 #endif
14146 #ifndef GL_NV_timeline_semaphore
14147 #define GL_NV_timeline_semaphore 1
14148 GLAPI int GLAD_GL_NV_timeline_semaphore;
14149 typedef void (APIENTRY PFNGLCREATESEMAPHORESNVPROC) (GLsizei n, GLuint *semaphores);

```



```

14150 GLAPI PFNGLCREATESEMAPHORESNVPROC glad_glCreateSemaphoresNV;
14151 #define glCreateSemaphoresNV glad_glCreateSemaphoresNV
14152 typedef void (APIENTRY PFNGLSEMAPHOREPARAMETERIVNVPROC) (GLuint semaphore, GLenum pname, const GLint
    *params);
14153 GLAPI PFNGLSEMAPHOREPARAMETERIVNVPROC glad_glSemaphoreParameterivNV;
14154 #define glSemaphoreParameterivNV glad_glSemaphoreParameterivNV
14155 typedef void (APIENTRY PFNGLGETSEMAPHOREPARAMETERIVNVPROC) (GLuint semaphore, GLenum pname, GLint
    *params);
14156 GLAPI PFNGLGETSEMAPHOREPARAMETERIVNVPROC glad_glGetSemaphoreParameterivNV;
14157 #define glGetSemaphoreParameterivNV glad_glGetSemaphoreParameterivNV
14158 #endif
14159 #ifndef GL_NV_transform_feedback
14160 #define GL_NV_transform_feedback 1
14161 GLAPI int GLAD_GL_NV_transform_feedback;
14162 typedef void (APIENTRY PFNGLBEGINTRANSFORMFEEDBACKNVPROC) (GLenum primitiveMode);
14163 GLAPI PFNGLBEGINTRANSFORMFEEDBACKNVPROC glad_glBeginTransformFeedbackNV;
14164 #define glBeginTransformFeedbackNV glad_glBeginTransformFeedbackNV
14165 typedef void (APIENTRY PFNGLENDTRANSFORMFEEDBACKNVPROC) (void);
14166 GLAPI PFNGLENDTRANSFORMFEEDBACKNVPROC glad_glEndTransformFeedbackNV;
14167 #define glEndTransformFeedbackNV glad_glEndTransformFeedbackNV
14168 typedef void (APIENTRY PFNGLTRANSFORMFEEDBACKATTRIBSNVPROC) (GLsizei count, const GLint *attribs,
    GLenum bufferMode);
14169 GLAPI PFNGLTRANSFORMFEEDBACKATTRIBSNVPROC glad_glTransformFeedbackAttribsNV;
14170 #define glTransformFeedbackAttribsNV glad_glTransformFeedbackAttribsNV
14171 typedef void (APIENTRY PFNGLBINDBUFFERRANGENVPROC) (GLenum target, GLuint index, GLuint buffer,
    GLintptr offset, GLsizeiptr size);
14172 GLAPI PFNGLBINDBUFFERRANGENVPROC glad_glBindBufferRangeNV;
14173 #define glBindBufferRangeNV glad_glBindBufferRangeNV
14174 typedef void (APIENTRY PFNGLBINDBUFFEROFFSETNVPROC) (GLenum target, GLuint index, GLuint buffer,
    GLintptr offset);
14175 GLAPI PFNGLBINDBUFFEROFFSETNVPROC glad_glBindBufferOffsetNV;
14176 #define glBindBufferOffsetNV glad_glBindBufferOffsetNV
14177 typedef void (APIENTRY PFNGLBINDBUFFERBASENVPROC) (GLenum target, GLuint index, GLuint buffer);
14178 GLAPI PFNGLBINDBUFFERBASENVPROC glad_glBindBufferBaseNV;
14179 #define glBindBufferBaseNV glad_glBindBufferBaseNV
14180 typedef void (APIENTRY PFNGLTRANSFORMFEEDBACKVARYINGSNVPROC) (GLuint program, GLsizei count, const
    GLint *locations, GLenum bufferMode);
14181 GLAPI PFNGLTRANSFORMFEEDBACKVARYINGSNVPROC glad_glTransformFeedbackVaryingsNV;
14182 #define glTransformFeedbackVaryingsNV glad_glTransformFeedbackVaryingsNV
14183 typedef void (APIENTRY PFNGLACTIVEVARYINGNVPROC) (GLuint program, const GLchar *name);
14184 GLAPI PFNGLACTIVEVARYINGNVPROC glad_glActiveVaryingNV;
14185 #define glActiveVaryingNV glad_glActiveVaryingNV
14186 typedef GLint (APIENTRY PFNGLGETVARYINGLOCATIONNVPROC) (GLuint program, const GLchar *name);
14187 GLAPI PFNGLGETVARYINGLOCATIONNVPROC glad_glGetVaryingLocationNV;
14188 #define glGetVaryingLocationNV glad_glGetVaryingLocationNV
14189 typedef void (APIENTRY PFNGLGETACTIVEVARYINGNVPROC) (GLuint program, GLuint index, GLsizei bufSize,
    GLsizei *length, GLsizei *size, GLenum *type, GLchar *name);
14190 GLAPI PFNGLGETACTIVEVARYINGNVPROC glad_glGetActiveVaryingNV;
14191 #define glGetActiveVaryingNV glad_glGetActiveVaryingNV
14192 typedef void (APIENTRY PFNGLGETTRANSFORMFEEDBACKVARYINGNVPROC) (GLuint program, GLuint index, GLint
    *location);
14193 GLAPI PFNGLGETTRANSFORMFEEDBACKVARYINGNVPROC glad_glGetTransformFeedbackVaryingNV;
14194 #define glGetTransformFeedbackVaryingNV glad_glGetTransformFeedbackVaryingNV
14195 typedef void (APIENTRY PFNGLTRANSFORMFEEDBACKSTREAMATTRIBSNVPROC) (GLsizei count, const GLint
    *attribs, GLsizei nbuffers, const GLint *bufstreams, GLenum bufferMode);
14196 GLAPI PFNGLTRANSFORMFEEDBACKSTREAMATTRIBSNVPROC glad_glTransformFeedbackStreamAttribsNV;
14197 #define glTransformFeedbackStreamAttribsNV glad_glTransformFeedbackStreamAttribsNV
14198 #endif
14199 #ifndef GL_NV_transform_feedback2
14200 #define GL_NV_transform_feedback2 1
14201 GLAPI int GLAD_GL_NV_transform_feedback2;
14202 typedef void (APIENTRY PFNGLBINDTRANSFORMFEEDBACKNVPROC) (GLenum target, GLuint id);
14203 GLAPI PFNGLBINDTRANSFORMFEEDBACKNVPROC glad_glBindTransformFeedbackNV;
14204 #define glBindTransformFeedbackNV glad_glBindTransformFeedbackNV
14205 typedef void (APIENTRY PFNGLDELETETRANSFORMFEEDBACKSNVPROC) (GLsizei n, const GLuint *ids);
14206 GLAPI PFNGLDELETETRANSFORMFEEDBACKSNVPROC glad_glDeleteTransformFeedbacksNV;
14207 #define glDeleteTransformFeedbacksNV glad_glDeleteTransformFeedbacksNV
14208 typedef void (APIENTRY PFNGLGENTRANSFORMFEEDBACKSNVPROC) (GLsizei n, GLuint *ids);
14209 GLAPI PFNGLGENTRANSFORMFEEDBACKSNVPROC glad_glGenTransformFeedbacksNV;
14210 #define glGenTransformFeedbacksNV glad_glGenTransformFeedbacksNV
14211 typedef GLboolean (APIENTRY PFNGLISTRANSFORMFEEDBACKNVPROC) (GLuint id);
14212 GLAPI PFNGLISTRANSFORMFEEDBACKNVPROC glad_glIsTransformFeedbackNV;
14213 #define glIsTransformFeedbackNV glad_glIsTransformFeedbackNV
14214 typedef void (APIENTRY PFNGLPAUSETRANSFORMFEEDBACKNVPROC) (void);
14215 GLAPI PFNGLPAUSETRANSFORMFEEDBACKNVPROC glad_glPauseTransformFeedbackNV;
14216 #define glPauseTransformFeedbackNV glad_glPauseTransformFeedbackNV
14217 typedef void (APIENTRY PFNGLRESUMETRANSFORMFEEDBACKNVPROC) (void);
14218 GLAPI PFNGLRESUMETRANSFORMFEEDBACKNVPROC glad_glResumeTransformFeedbackNV;
14219 #define glResumeTransformFeedbackNV glad_glResumeTransformFeedbackNV
14220 typedef void (APIENTRY PFNGLDRAWTRANSFORMFEEDBACKNVPROC) (GLenum mode, GLuint id);
14221 GLAPI PFNGLDRAWTRANSFORMFEEDBACKNVPROC glad_glDrawTransformFeedbackNV;
14222 #define glDrawTransformFeedbackNV glad_glDrawTransformFeedbackNV
14223 #endif
14224 #ifndef GL_NV_uniform_buffer_std430_layout
14225 #define GL_NV_uniform_buffer_std430_layout 1
14226 GLAPI int GLAD_GL_NV_uniform_buffer_std430_layout;
14227 #endif

```

```

14228 #ifndef GL_NV_uniform_buffer_unified_memory
14229 #define GL_NV_uniform_buffer_unified_memory 1
14230 GLAPI int GLAD_GL_NV_uniform_buffer_unified_memory;
14231 #endif
14232 #ifndef GL_NV_vdpau_interop
14233 #define GL_NV_vdpau_interop 1
14234 GLAPI int GLAD_GL_NV_vdpau_interop;
14235 typedef void (APIENTRY PFNGLVDPAAUINITNVPROC)(const void *vdpDevice, const void *getProcAddress);
14236 GLAPI PFNGLVDPAAUINITNVPROC glad_glVDPAAUinitNV;
14237 #define glVDPAAUinitNV glad_glVDPAAUinitNV
14238 typedef void (APIENTRY PFNGLVDPAAUFININVPROC)(void);
14239 GLAPI PFNGLVDPAAUFININVPROC glad_glVDPAAUfiniNV;
14240 #define glVDPAAUfiniNV glad_glVDPAAUfiniNV
14241 typedef GLvdpauSurfaceNV (APIENTRY PFNGLVDPAAUREGISTERVIDEOSURFACENVPROC)(const void *vdpSurface,
    GLenum target, GLsizei numTextureNames, const GLuint *textureNames);
14242 GLAPI PFNGLVDPAAUREGISTERVIDEOSURFACENVPROC glad_glVDPAAUregisterVideoSurfaceNV;
14243 #define glVDPAAUregisterVideoSurfaceNV glad_glVDPAAUregisterVideoSurfaceNV
14244 typedef GLvdpauSurfaceNV (APIENTRY PFNGLVDPAAUREGISTEROUTPUTSURFACENVPROC)(const void *vdpSurface,
    GLenum target, GLsizei numTextureNames, const GLuint *textureNames);
14245 GLAPI PFNGLVDPAAUREGISTEROUTPUTSURFACENVPROC glad_glVDPAAUregisterOutputSurfaceNV;
14246 #define glVDPAAUregisterOutputSurfaceNV glad_glVDPAAUregisterOutputSurfaceNV
14247 typedef GLboolean (APIENTRY PFNGLVDPAAUISSURFACENVPROC)(GLvdpauSurfaceNV surface);
14248 GLAPI PFNGLVDPAAUISSURFACENVPROC glad_glVDPAAUIsSurfaceNV;
14249 #define glVDPAAUIsSurfaceNV glad_glVDPAAUIsSurfaceNV
14250 typedef void (APIENTRY PFNGLVDPAAUUNREGISTERSURFACENVPROC)(GLvdpauSurfaceNV surface);
14251 GLAPI PFNGLVDPAAUUNREGISTERSURFACENVPROC glad_glVDPAAUunregisterSurfaceNV;
14252 #define glVDPAAUunregisterSurfaceNV glad_glVDPAAUunregisterSurfaceNV
14253 typedef void (APIENTRY PFNGLVDPAAUGETSURFACEIVNVPROC)(GLvdpauSurfaceNV surface, GLenum pname, GLsizei
    count, GLsizei *length, GLint *values);
14254 GLAPI PFNGLVDPAAUGETSURFACEIVNVPROC glad_glVDPAAUGetSurfaceivNV;
14255 #define glVDPAAUGetSurfaceivNV glad_glVDPAAUGetSurfaceivNV
14256 typedef void (APIENTRY PFNGLVDPAAUSURFACEACCESSNVPROC)(GLvdpauSurfaceNV surface, GLenum access);
14257 GLAPI PFNGLVDPAAUSURFACEACCESSNVPROC glad_glVDPAAUSurfaceAccessNV;
14258 #define glVDPAAUSurfaceAccessNV glad_glVDPAAUSurfaceAccessNV
14259 typedef void (APIENTRY PFNGLVDPAAUMAPSURFACESNVPROC)(GLsizei numSurfaces, const GLvdpauSurfaceNV
    *surfaces);
14260 GLAPI PFNGLVDPAAUMAPSURFACESNVPROC glad_glVDPAAUmapSurfacesNV;
14261 #define glVDPAAUmapSurfacesNV glad_glVDPAAUmapSurfacesNV
14262 typedef void (APIENTRY PFNGLVDPAAUUNMAPSURFACESNVPROC)(GLsizei numSurface, const GLvdpauSurfaceNV
    *surfaces);
14263 GLAPI PFNGLVDPAAUUNMAPSURFACESNVPROC glad_glVDPAAUunmapSurfacesNV;
14264 #define glVDPAAUunmapSurfacesNV glad_glVDPAAUunmapSurfacesNV
14265 #endif
14266 #ifndef GL_NV_vdpau_interop2
14267 #define GL_NV_vdpau_interop2 1
14268 GLAPI int GLAD_GL_NV_vdpau_interop2;
14269 typedef GLvdpauSurfaceNV (APIENTRY PFNGLVDPAAUREGISTERVIDEOSURFACEWITHPICTURESTRUCTURENVPROC)(const
    void *vdpSurface, GLenum target, GLsizei numTextureNames, const GLuint *textureNames, GLboolean
    isFrameStructure);
14270 GLAPI PFNGLVDPAAUREGISTERVIDEOSURFACEWITHPICTURESTRUCTURENVPROC
    glad_glVDPAAUregisterVideoSurfaceWithPictureStructureNV;
14271 #define glVDPAAUregisterVideoSurfaceWithPictureStructureNV
    glad_glVDPAAUregisterVideoSurfaceWithPictureStructureNV
14272 #endif
14273 #ifndef GL_NV_vertex_array_range
14274 #define GL_NV_vertex_array_range 1
14275 GLAPI int GLAD_GL_NV_vertex_array_range;
14276 typedef void (APIENTRY PFNGLFLUSHVERTEXARRAYRANGENVPROC)(void);
14277 GLAPI PFNGLFLUSHVERTEXARRAYRANGENVPROC glad_glFlushVertexArrayRangeNV;
14278 #define glFlushVertexArrayRangeNV glad_glFlushVertexArrayRangeNV
14279 typedef void (APIENTRY PFNGLVERTEXARRAYRANGENVPROC)(GLsizei length, const void *pointer);
14280 GLAPI PFNGLVERTEXARRAYRANGENVPROC glad_glVertexArrayRangeNV;
14281 #define glVertexArrayRangeNV glad_glVertexArrayRangeNV
14282 #endif
14283 #ifndef GL_NV_vertex_array_range2
14284 #define GL_NV_vertex_array_range2 1
14285 GLAPI int GLAD_GL_NV_vertex_array_range2;
14286 #endif
14287 #ifndef GL_NV_vertex_attrib_integer_64bit
14288 #define GL_NV_vertex_attrib_integer_64bit 1
14289 GLAPI int GLAD_GL_NV_vertex_attrib_integer_64bit;
14290 typedef void (APIENTRY PFNGLVERTEXATTRIBL1I64NVPROC)(GLuint index, GLint64EXT x);
14291 GLAPI PFNGLVERTEXATTRIBL1I64NVPROC glad_glVertexAttribL1i64NV;
14292 #define glVertexAttribL1i64NV glad_glVertexAttribL1i64NV
14293 typedef void (APIENTRY PFNGLVERTEXATTRIBL2I64NVPROC)(GLuint index, GLint64EXT x, GLint64EXT y);
14294 GLAPI PFNGLVERTEXATTRIBL2I64NVPROC glad_glVertexAttribL2i64NV;
14295 #define glVertexAttribL2i64NV glad_glVertexAttribL2i64NV
14296 typedef void (APIENTRY PFNGLVERTEXATTRIBL3I64NVPROC)(GLuint index, GLint64EXT x, GLint64EXT y,
    GLint64EXT z);
14297 GLAPI PFNGLVERTEXATTRIBL3I64NVPROC glad_glVertexAttribL3i64NV;
14298 #define glVertexAttribL3i64NV glad_glVertexAttribL3i64NV
14299 typedef void (APIENTRY PFNGLVERTEXATTRIBL4I64NVPROC)(GLuint index, GLint64EXT x, GLint64EXT y,
    GLint64EXT z, GLint64EXT w);
14300 GLAPI PFNGLVERTEXATTRIBL4I64NVPROC glad_glVertexAttribL4i64NV;
14301 #define glVertexAttribL4i64NV glad_glVertexAttribL4i64NV
14302 typedef void (APIENTRY PFNGLVERTEXATTRIBL1I64VNVPROC)(GLuint index, const GLint64EXT *v);
14303 GLAPI PFNGLVERTEXATTRIBL1I64VNVPROC glad_glVertexAttribL1i64vNV;

```

```

14304 #define glVertexAttribL1i64vNV glad_glVertexAttribL1i64vNV
14305 typedef void (APIENTRY PFNGLVERTEXATTRIBL2i64VNVPROC) (GLuint index, const GLint64EXT *v);
14306 GLAPI PFNGLVERTEXATTRIBL2i64VNVPROC glad_glVertexAttribL2i64vNV;
14307 #define glVertexAttribL2i64vNV glad_glVertexAttribL2i64vNV
14308 typedef void (APIENTRY PFNGLVERTEXATTRIBL3i64VNVPROC) (GLuint index, const GLint64EXT *v);
14309 GLAPI PFNGLVERTEXATTRIBL3i64VNVPROC glad_glVertexAttribL3i64vNV;
14310 #define glVertexAttribL3i64vNV glad_glVertexAttribL3i64vNV
14311 typedef void (APIENTRY PFNGLVERTEXATTRIBL4i64VNVPROC) (GLuint index, const GLint64EXT *v);
14312 GLAPI PFNGLVERTEXATTRIBL4i64VNVPROC glad_glVertexAttribL4i64vNV;
14313 #define glVertexAttribL4i64vNV glad_glVertexAttribL4i64vNV
14314 typedef void (APIENTRY PFNGLVERTEXATTRIBL1UI64VNVPROC) (GLuint index, GLuint64EXT x);
14315 GLAPI PFNGLVERTEXATTRIBL1UI64VNVPROC glad_glVertexAttribL1ui64NV;
14316 #define glVertexAttribL1ui64NV glad_glVertexAttribL1ui64NV
14317 typedef void (APIENTRY PFNGLVERTEXATTRIBL2UI64VNVPROC) (GLuint index, GLuint64EXT x, GLuint64EXT y);
14318 GLAPI PFNGLVERTEXATTRIBL2UI64VNVPROC glad_glVertexAttribL2ui64NV;
14319 #define glVertexAttribL2ui64NV glad_glVertexAttribL2ui64NV
14320 typedef void (APIENTRY PFNGLVERTEXATTRIBL3UI64VNVPROC) (GLuint index, GLuint64EXT x, GLuint64EXT y,
    GLuint64EXT z);
14321 GLAPI PFNGLVERTEXATTRIBL3UI64VNVPROC glad_glVertexAttribL3ui64NV;
14322 #define glVertexAttribL3ui64NV glad_glVertexAttribL3ui64NV
14323 typedef void (APIENTRY PFNGLVERTEXATTRIBL4UI64VNVPROC) (GLuint index, GLuint64EXT x, GLuint64EXT y,
    GLuint64EXT z, GLuint64EXT w);
14324 GLAPI PFNGLVERTEXATTRIBL4UI64VNVPROC glad_glVertexAttribL4ui64NV;
14325 #define glVertexAttribL4ui64NV glad_glVertexAttribL4ui64NV
14326 typedef void (APIENTRY PFNGLVERTEXATTRIBL1UI64VNVPROC) (GLuint index, const GLuint64EXT *v);
14327 GLAPI PFNGLVERTEXATTRIBL1UI64VNVPROC glad_glVertexAttribL1ui64vNV;
14328 #define glVertexAttribL1ui64vNV glad_glVertexAttribL1ui64vNV
14329 typedef void (APIENTRY PFNGLVERTEXATTRIBL2UI64VNVPROC) (GLuint index, const GLuint64EXT *v);
14330 GLAPI PFNGLVERTEXATTRIBL2UI64VNVPROC glad_glVertexAttribL2ui64vNV;
14331 #define glVertexAttribL2ui64vNV glad_glVertexAttribL2ui64vNV
14332 typedef void (APIENTRY PFNGLVERTEXATTRIBL3UI64VNVPROC) (GLuint index, const GLuint64EXT *v);
14333 GLAPI PFNGLVERTEXATTRIBL3UI64VNVPROC glad_glVertexAttribL3ui64vNV;
14334 #define glVertexAttribL3ui64vNV glad_glVertexAttribL3ui64vNV
14335 typedef void (APIENTRY PFNGLVERTEXATTRIBL4UI64VNVPROC) (GLuint index, const GLuint64EXT *v);
14336 GLAPI PFNGLVERTEXATTRIBL4UI64VNVPROC glad_glVertexAttribL4ui64vNV;
14337 #define glVertexAttribL4ui64vNV glad_glVertexAttribL4ui64vNV
14338 typedef void (APIENTRY PFNGLGETVERTEXATTRIBLi64VNVPROC) (GLuint index, GLenum pname, GLint64EXT
    *params);
14339 GLAPI PFNGLGETVERTEXATTRIBLi64VNVPROC glad_glGetVertexAttribLi64vNV;
14340 #define glGetVertexAttribLi64vNV glad_glGetVertexAttribLi64vNV
14341 typedef void (APIENTRY PFNGLGETVERTEXATTRIBLUI64VNVPROC) (GLuint index, GLenum pname, GLuint64EXT
    *params);
14342 GLAPI PFNGLGETVERTEXATTRIBLUI64VNVPROC glad_glGetVertexAttribLui64vNV;
14343 #define glGetVertexAttribLui64vNV glad_glGetVertexAttribLui64vNV
14344 typedef void (APIENTRY PFNGLVERTEXATTRIBLFORMATNVPROC) (GLuint index, GLint size, GLenum type, GLsizei
    stride);
14345 GLAPI PFNGLVERTEXATTRIBLFORMATNVPROC glad_glVertexAttribLFormatNV;
14346 #define glVertexAttribLFormatNV glad_glVertexAttribLFormatNV
14347 #endif
14348 #ifndef GL_NV_vertex_buffer_unified_memory
14349 #define GL_NV_vertex_buffer_unified_memory 1
14350 GLAPI int GLAD_GL_NV_vertex_buffer_unified_memory;
14351 typedef void (APIENTRY PFNGLBUFFERADDRESSRANGENVPROC) (GLenum pname, GLuint index, GLuint64EXT
    address, GLsizeiptr length);
14352 GLAPI PFNGLBUFFERADDRESSRANGENVPROC glad_glBufferAddressRangeNV;
14353 #define glBufferAddressRangeNV glad_glBufferAddressRangeNV
14354 typedef void (APIENTRY PFNGLVERTEXFORMATNVPROC) (GLint size, GLenum type, GLsizei stride);
14355 GLAPI PFNGLVERTEXFORMATNVPROC glad_glVertexFormatNV;
14356 #define glVertexFormatNV glad_glVertexFormatNV
14357 typedef void (APIENTRY PFNGLNORMALFORMATNVPROC) (GLenum type, GLsizei stride);
14358 GLAPI PFNGLNORMALFORMATNVPROC glad_glNormalFormatNV;
14359 #define glNormalFormatNV glad_glNormalFormatNV
14360 typedef void (APIENTRY PFNGLCOLORFORMATNVPROC) (GLint size, GLenum type, GLsizei stride);
14361 GLAPI PFNGLCOLORFORMATNVPROC glad_glColorFormatNV;
14362 #define glColorFormatNV glad_glColorFormatNV
14363 typedef void (APIENTRY PFNGLINDEXFORMATNVPROC) (GLenum type, GLsizei stride);
14364 GLAPI PFNGLINDEXFORMATNVPROC glad_glIndexFormatNV;
14365 #define glIndexFormatNV glad_glIndexFormatNV
14366 typedef void (APIENTRY PFNGLTEXCOORDFORMATNVPROC) (GLint size, GLenum type, GLsizei stride);
14367 GLAPI PFNGLTEXCOORDFORMATNVPROC glad_glTexCoordFormatNV;
14368 #define glTexCoordFormatNV glad_glTexCoordFormatNV
14369 typedef void (APIENTRY PFNGLEDGEFLAGFORMATNVPROC) (GLsizei stride);
14370 GLAPI PFNGLEDGEFLAGFORMATNVPROC glad_glEdgeFlagFormatNV;
14371 #define glEdgeFlagFormatNV glad_glEdgeFlagFormatNV
14372 typedef void (APIENTRY PFNGLSECONDARYCOLORFORMATNVPROC) (GLint size, GLenum type, GLsizei stride);
14373 GLAPI PFNGLSECONDARYCOLORFORMATNVPROC glad_glSecondaryColorFormatNV;
14374 #define glSecondaryColorFormatNV glad_glSecondaryColorFormatNV
14375 typedef void (APIENTRY PFNGLFOGCOORDFORMATNVPROC) (GLenum type, GLsizei stride);
14376 GLAPI PFNGLFOGCOORDFORMATNVPROC glad_glFogCoordFormatNV;
14377 #define glFogCoordFormatNV glad_glFogCoordFormatNV
14378 typedef void (APIENTRY PFNGLVERTEXATTRIBFORMATNVPROC) (GLuint index, GLint size, GLenum type,
    GLboolean normalized, GLsizei stride);
14379 GLAPI PFNGLVERTEXATTRIBFORMATNVPROC glad_glVertexAttribFormatNV;
14380 #define glVertexAttribFormatNV glad_glVertexAttribFormatNV
14381 typedef void (APIENTRY PFNGLVERTEXATTRIBIFORMATNVPROC) (GLuint index, GLint size, GLenum type, GLsizei
    stride);
14382 GLAPI PFNGLVERTEXATTRIBIFORMATNVPROC glad_glVertexAttribIFormatNV;

```

```

14383 #define glVertexAttribIFormatNV glad_glVertexAttribIFormatNV
14384 typedef void (APIENTRY PFNGLGETINTEGERUI64I_VNVPROC) (GLenum value, GLuint index, GLuint64EXT
    *result);
14385 GLAPI PFNGLGETINTEGERUI64I_VNVPROC glad_glGetIntegerui64i_vNV;
14386 #define glGetIntegerui64i_vNV glad_glGetIntegerui64i_vNV
14387 #endif
14388 #ifndef GL_NV_vertex_program
14389 #define GL_NV_vertex_program 1
14390 GLAPI int GLAD_GL_NV_vertex_program;
14391 typedef GLboolean (APIENTRY PFNGLAREPROGRAMSRESIDENTNVPROC) (GLsizei n, const GLuint *programs,
    GLboolean *residences);
14392 GLAPI PFNGLAREPROGRAMSRESIDENTNVPROC glad_glAreProgramsResidentNV;
14393 #define glAreProgramsResidentNV glad_glAreProgramsResidentNV
14394 typedef void (APIENTRY PFNGLBINDPROGRAMNVPROC) (GLenum target, GLuint id);
14395 GLAPI PFNGLBINDPROGRAMNVPROC glad_glBindProgramNV;
14396 #define glBindProgramNV glad_glBindProgramNV
14397 typedef void (APIENTRY PFNGLDELETEPROGRAMSNVPROC) (GLsizei n, const GLuint *programs);
14398 GLAPI PFNGLDELETEPROGRAMSNVPROC glad_glDeleteProgramsNV;
14399 #define glDeleteProgramsNV glad_glDeleteProgramsNV
14400 typedef void (APIENTRY PFNGLEXECUTEPROGRAMNVPROC) (GLenum target, GLuint id, const GLfloat *params);
14401 GLAPI PFNGLEXECUTEPROGRAMNVPROC glad_glExecuteProgramNV;
14402 #define glExecuteProgramNV glad_glExecuteProgramNV
14403 typedef void (APIENTRY PFNGLGENPROGRAMSNVPROC) (GLsizei n, GLuint *programs);
14404 GLAPI PFNGLGENPROGRAMSNVPROC glad_glGenProgramsNV;
14405 #define glGenProgramsNV glad_glGenProgramsNV
14406 typedef void (APIENTRY PFNGLGETPROGRAMPARAMETERDVNVPROC) (GLenum target, GLuint index, GLenum pname,
    GLdouble *params);
14407 GLAPI PFNGLGETPROGRAMPARAMETERDVNVPROC glad_glGetProgramParameterdvNV;
14408 #define glGetProgramParameterdvNV glad_glGetProgramParameterdvNV
14409 typedef void (APIENTRY PFNGLGETPROGRAMPARAMETERFVNVPROC) (GLenum target, GLuint index, GLenum pname,
    GLfloat *params);
14410 GLAPI PFNGLGETPROGRAMPARAMETERFVNVPROC glad_glGetProgramParameterfvNV;
14411 #define glGetProgramParameterfvNV glad_glGetProgramParameterfvNV
14412 typedef void (APIENTRY PFNGLGETPROGRAMIVNVPROC) (GLuint id, GLenum pname, GLint *params);
14413 GLAPI PFNGLGETPROGRAMIVNVPROC glad_glGetProgramivNV;
14414 #define glGetProgramivNV glad_glGetProgramivNV
14415 typedef void (APIENTRY PFNGLGETPROGRAMSTRINGNVPROC) (GLuint id, GLenum pname, GLubyte *program);
14416 GLAPI PFNGLGETPROGRAMSTRINGNVPROC glad_glGetProgramStringNV;
14417 #define glGetProgramStringNV glad_glGetProgramStringNV
14418 typedef void (APIENTRY PFNGLGETTRACKMATRIXIVNVPROC) (GLenum target, GLuint address, GLenum pname,
    GLint *params);
14419 GLAPI PFNGLGETTRACKMATRIXIVNVPROC glad_glGetTrackMatrixivNV;
14420 #define glGetTrackMatrixivNV glad_glGetTrackMatrixivNV
14421 typedef void (APIENTRY PFNGLGETVERTEXATTRIBDVNVPROC) (GLuint index, GLenum pname, GLdouble *params);
14422 GLAPI PFNGLGETVERTEXATTRIBDVNVPROC glad_glGetVertexAttribdvNV;
14423 #define glGetVertexAttribdvNV glad_glGetVertexAttribdvNV
14424 typedef void (APIENTRY PFNGLGETVERTEXATTRIBFVNVPROC) (GLuint index, GLenum pname, GLfloat *params);
14425 GLAPI PFNGLGETVERTEXATTRIBFVNVPROC glad_glGetVertexAttribfvNV;
14426 #define glGetVertexAttribfvNV glad_glGetVertexAttribfvNV
14427 typedef void (APIENTRY PFNGLGETVERTEXATTRIBIVNVPROC) (GLuint index, GLenum pname, GLint *params);
14428 GLAPI PFNGLGETVERTEXATTRIBIVNVPROC glad_glGetVertexAttribivNV;
14429 #define glGetVertexAttribivNV glad_glGetVertexAttribivNV
14430 typedef void (APIENTRY PFNGLGETVERTEXATTRIBPOINTERVNVPROC) (GLuint index, GLenum pname, void
    **pointer);
14431 GLAPI PFNGLGETVERTEXATTRIBPOINTERVNVPROC glad_glGetVertexAttribPointervNV;
14432 #define glGetVertexAttribPointervNV glad_glGetVertexAttribPointervNV
14433 typedef GLboolean (APIENTRY PFNGLISPROGRAMNVPROC) (GLuint id);
14434 GLAPI PFNGLISPROGRAMNVPROC glad_glIsProgramNV;
14435 #define glIsProgramNV glad_glIsProgramNV
14436 typedef void (APIENTRY PFNGLLOADPROGRAMNVPROC) (GLenum target, GLuint id, GLsizei len, const GLubyte
    *program);
14437 GLAPI PFNGLLOADPROGRAMNVPROC glad_glLoadProgramNV;
14438 #define glLoadProgramNV glad_glLoadProgramNV
14439 typedef void (APIENTRY PFNGLPROGRAMPARAMETER4DNVPROC) (GLenum target, GLuint index, GLdouble x,
    GLdouble y, GLdouble z, GLdouble w);
14440 GLAPI PFNGLPROGRAMPARAMETER4DNVPROC glad_glProgramParameter4dNV;
14441 #define glProgramParameter4dNV glad_glProgramParameter4dNV
14442 typedef void (APIENTRY PFNGLPROGRAMPARAMETER4DVNVPROC) (GLenum target, GLuint index, const GLdouble
    *v);
14443 GLAPI PFNGLPROGRAMPARAMETER4DVNVPROC glad_glProgramParameter4dvNV;
14444 #define glProgramParameter4dvNV glad_glProgramParameter4dvNV
14445 typedef void (APIENTRY PFNGLPROGRAMPARAMETER4FNVPROC) (GLenum target, GLuint index, GLfloat x, GLfloat
    y, GLfloat z, GLfloat w);
14446 GLAPI PFNGLPROGRAMPARAMETER4FNVPROC glad_glProgramParameter4fvNV;
14447 #define glProgramParameter4fvNV glad_glProgramParameter4fvNV
14448 typedef void (APIENTRY PFNGLPROGRAMPARAMETER4FVNVPROC) (GLenum target, GLuint index, const GLfloat
    *v);
14449 GLAPI PFNGLPROGRAMPARAMETER4FVNVPROC glad_glProgramParameter4fvNV;
14450 #define glProgramParameter4fvNV glad_glProgramParameter4fvNV
14451 typedef void (APIENTRY PFNGLPROGRAMPARAMETERS4DVNVPROC) (GLenum target, GLuint index, GLsizei count,
    const GLdouble *v);
14452 GLAPI PFNGLPROGRAMPARAMETERS4DVNVPROC glad_glProgramParameters4dvNV;
14453 #define glProgramParameters4dvNV glad_glProgramParameters4dvNV
14454 typedef void (APIENTRY PFNGLPROGRAMPARAMETERS4FVNVPROC) (GLenum target, GLuint index, GLsizei count,
    const GLfloat *v);
14455 GLAPI PFNGLPROGRAMPARAMETERS4FVNVPROC glad_glProgramParameters4fvNV;
14456 #define glProgramParameters4fvNV glad_glProgramParameters4fvNV

```



```
14457 typedef void (APIENTRY PFNGLREQUESTRESIDENTPROGRAMSNVPROC)(GLsizei n, const GLuint *programs);
14458 GLAPI PFNGLREQUESTRESIDENTPROGRAMSNVPROC glad_glRequestResidentProgramsNV;
14459 #define glRequestResidentProgramsNV glad_glRequestResidentProgramsNV
14460 typedef void (APIENTRY PFNGLTRACKMATRIXNVPROC)(GLenum target, GLuint address, GLenum matrix, GLenum
transform);
14461 GLAPI PFNGLTRACKMATRIXNVPROC glad_glTrackMatrixNV;
14462 #define glTrackMatrixNV glad_glTrackMatrixNV
14463 typedef void (APIENTRY PFNGLVERTEXATTRIBPOINTERNVPROC)(GLuint index, GLint fsize, GLenum type,
GLsizei stride, const void *pointer);
14464 GLAPI PFNGLVERTEXATTRIBPOINTERNVPROC glad_glVertexAttribPointerNV;
14465 #define glVertexAttribPointerNV glad_glVertexAttribPointerNV
14466 typedef void (APIENTRY PFNGLVERTEXATTRIB1DNVPROC)(GLuint index, GLdouble x);
14467 GLAPI PFNGLVERTEXATTRIB1DNVPROC glad_glVertexAttrib1dNV;
14468 #define glVertexAttrib1dNV glad_glVertexAttrib1dNV
14469 typedef void (APIENTRY PFNGLVERTEXATTRIB1DVNVPROC)(GLuint index, const GLdouble *v);
14470 GLAPI PFNGLVERTEXATTRIB1DVNVPROC glad_glVertexAttrib1dvNV;
14471 #define glVertexAttrib1dvNV glad_glVertexAttrib1dvNV
14472 typedef void (APIENTRY PFNGLVERTEXATTRIB1FNVPROC)(GLuint index, GLfloat x);
14473 GLAPI PFNGLVERTEXATTRIB1FNVPROC glad_glVertexAttrib1fNV;
14474 #define glVertexAttrib1fNV glad_glVertexAttrib1fNV
14475 typedef void (APIENTRY PFNGLVERTEXATTRIB1FVNVPROC)(GLuint index, const GLfloat *v);
14476 GLAPI PFNGLVERTEXATTRIB1FVNVPROC glad_glVertexAttrib1fvNV;
14477 #define glVertexAttrib1fvNV glad_glVertexAttrib1fvNV
14478 typedef void (APIENTRY PFNGLVERTEXATTRIB1SNVPROC)(GLuint index, GLshort x);
14479 GLAPI PFNGLVERTEXATTRIB1SNVPROC glad_glVertexAttrib1sNV;
14480 #define glVertexAttrib1sNV glad_glVertexAttrib1sNV
14481 typedef void (APIENTRY PFNGLVERTEXATTRIB1SVNVPROC)(GLuint index, const GLshort *v);
14482 GLAPI PFNGLVERTEXATTRIB1SVNVPROC glad_glVertexAttrib1svNV;
14483 #define glVertexAttrib1svNV glad_glVertexAttrib1svNV
14484 typedef void (APIENTRY PFNGLVERTEXATTRIB2DNVPROC)(GLuint index, GLdouble x, GLdouble y);
14485 GLAPI PFNGLVERTEXATTRIB2DNVPROC glad_glVertexAttrib2dNV;
14486 #define glVertexAttrib2dNV glad_glVertexAttrib2dNV
14487 typedef void (APIENTRY PFNGLVERTEXATTRIB2DVNVPROC)(GLuint index, const GLdouble *v);
14488 GLAPI PFNGLVERTEXATTRIB2DVNVPROC glad_glVertexAttrib2dvNV;
14489 #define glVertexAttrib2dvNV glad_glVertexAttrib2dvNV
14490 typedef void (APIENTRY PFNGLVERTEXATTRIB2FNVPROC)(GLuint index, GLfloat x, GLfloat y);
14491 GLAPI PFNGLVERTEXATTRIB2FNVPROC glad_glVertexAttrib2fNV;
14492 #define glVertexAttrib2fNV glad_glVertexAttrib2fNV
14493 typedef void (APIENTRY PFNGLVERTEXATTRIB2FVNVPROC)(GLuint index, const GLfloat *v);
14494 GLAPI PFNGLVERTEXATTRIB2FVNVPROC glad_glVertexAttrib2fvNV;
14495 #define glVertexAttrib2fvNV glad_glVertexAttrib2fvNV
14496 typedef void (APIENTRY PFNGLVERTEXATTRIB2SNVPROC)(GLuint index, GLshort x, GLshort y);
14497 GLAPI PFNGLVERTEXATTRIB2SNVPROC glad_glVertexAttrib2sNV;
14498 #define glVertexAttrib2sNV glad_glVertexAttrib2sNV
14499 typedef void (APIENTRY PFNGLVERTEXATTRIB2SVNVPROC)(GLuint index, const GLshort *v);
14500 GLAPI PFNGLVERTEXATTRIB2SVNVPROC glad_glVertexAttrib2svNV;
14501 #define glVertexAttrib2svNV glad_glVertexAttrib2svNV
14502 typedef void (APIENTRY PFNGLVERTEXATTRIB3DNVPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z);
14503 GLAPI PFNGLVERTEXATTRIB3DNVPROC glad_glVertexAttrib3dNV;
14504 #define glVertexAttrib3dNV glad_glVertexAttrib3dNV
14505 typedef void (APIENTRY PFNGLVERTEXATTRIB3DVNVPROC)(GLuint index, const GLdouble *v);
14506 GLAPI PFNGLVERTEXATTRIB3DVNVPROC glad_glVertexAttrib3dvNV;
14507 #define glVertexAttrib3dvNV glad_glVertexAttrib3dvNV
14508 typedef void (APIENTRY PFNGLVERTEXATTRIB3FNVPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z);
14509 GLAPI PFNGLVERTEXATTRIB3FNVPROC glad_glVertexAttrib3fNV;
14510 #define glVertexAttrib3fNV glad_glVertexAttrib3fNV
14511 typedef void (APIENTRY PFNGLVERTEXATTRIB3FVNVPROC)(GLuint index, const GLfloat *v);
14512 GLAPI PFNGLVERTEXATTRIB3FVNVPROC glad_glVertexAttrib3fvNV;
14513 #define glVertexAttrib3fvNV glad_glVertexAttrib3fvNV
14514 typedef void (APIENTRY PFNGLVERTEXATTRIB3SNVPROC)(GLuint index, GLshort x, GLshort y, GLshort z);
14515 GLAPI PFNGLVERTEXATTRIB3SNVPROC glad_glVertexAttrib3sNV;
14516 #define glVertexAttrib3sNV glad_glVertexAttrib3sNV
14517 typedef void (APIENTRY PFNGLVERTEXATTRIB3SVNVPROC)(GLuint index, const GLshort *v);
14518 GLAPI PFNGLVERTEXATTRIB3SVNVPROC glad_glVertexAttrib3svNV;
14519 #define glVertexAttrib3svNV glad_glVertexAttrib3svNV
14520 typedef void (APIENTRY PFNGLVERTEXATTRIB4DNVPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z,
GLdouble w);
14521 GLAPI PFNGLVERTEXATTRIB4DNVPROC glad_glVertexAttrib4dNV;
14522 #define glVertexAttrib4dNV glad_glVertexAttrib4dNV
14523 typedef void (APIENTRY PFNGLVERTEXATTRIB4DVNVPROC)(GLuint index, const GLdouble *v);
14524 GLAPI PFNGLVERTEXATTRIB4DVNVPROC glad_glVertexAttrib4dvNV;
14525 #define glVertexAttrib4dvNV glad_glVertexAttrib4dvNV
14526 typedef void (APIENTRY PFNGLVERTEXATTRIB4FNVPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z,
GLfloat w);
14527 GLAPI PFNGLVERTEXATTRIB4FNVPROC glad_glVertexAttrib4fNV;
14528 #define glVertexAttrib4fNV glad_glVertexAttrib4fNV
14529 typedef void (APIENTRY PFNGLVERTEXATTRIB4FVNVPROC)(GLuint index, const GLfloat *v);
14530 GLAPI PFNGLVERTEXATTRIB4FVNVPROC glad_glVertexAttrib4fvNV;
14531 #define glVertexAttrib4fvNV glad_glVertexAttrib4fvNV
14532 typedef void (APIENTRY PFNGLVERTEXATTRIB4SNVPROC)(GLuint index, GLshort x, GLshort y, GLshort z,
GLshort w);
14533 GLAPI PFNGLVERTEXATTRIB4SNVPROC glad_glVertexAttrib4sNV;
14534 #define glVertexAttrib4sNV glad_glVertexAttrib4sNV
14535 typedef void (APIENTRY PFNGLVERTEXATTRIB4SVNVPROC)(GLuint index, const GLshort *v);
14536 GLAPI PFNGLVERTEXATTRIB4SVNVPROC glad_glVertexAttrib4svNV;
14537 #define glVertexAttrib4svNV glad_glVertexAttrib4svNV
14538 typedef void (APIENTRY PFNGLVERTEXATTRIB4UBNVPROC)(GLuint index, GLubyte x, GLubyte y, GLubyte z,
```

```

    GLubyte w);
14539 GLAPI PFNGLVERTEXATTRIB4UBNVPROC glad_glVertexAttrib4ubNV;
14540 #define glVertexAttrib4ubNV glad_glVertexAttrib4ubNV
14541 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIB4UBNVPROC) (GLuint index, const GLubyte *v);
14542 GLAPI PFNGLVERTEXATTRIB4UBNVPROC glad_glVertexAttrib4ubNV;
14543 #define glVertexAttrib4ubNV glad_glVertexAttrib4ubNV
14544 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS1DVNVPROC) (GLuint index, GLsizei count, const GLdouble *v);
14545 GLAPI PFNGLVERTEXATTRIBS1DVNVPROC glad_glVertexAttribs1dvNV;
14546 #define glVertexAttribs1dvNV glad_glVertexAttribs1dvNV
14547 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS1FVNVPROC) (GLuint index, GLsizei count, const GLfloat *v);
14548 GLAPI PFNGLVERTEXATTRIBS1FVNVPROC glad_glVertexAttribs1fvNV;
14549 #define glVertexAttribs1fvNV glad_glVertexAttribs1fvNV
14550 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS1SVNVPROC) (GLuint index, GLsizei count, const GLshort *v);
14551 GLAPI PFNGLVERTEXATTRIBS1SVNVPROC glad_glVertexAttribs1svNV;
14552 #define glVertexAttribs1svNV glad_glVertexAttribs1svNV
14553 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS2DVNVPROC) (GLuint index, GLsizei count, const GLdouble *v);
14554 GLAPI PFNGLVERTEXATTRIBS2DVNVPROC glad_glVertexAttribs2dvNV;
14555 #define glVertexAttribs2dvNV glad_glVertexAttribs2dvNV
14556 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS2FVNVPROC) (GLuint index, GLsizei count, const GLfloat *v);
14557 GLAPI PFNGLVERTEXATTRIBS2FVNVPROC glad_glVertexAttribs2fvNV;
14558 #define glVertexAttribs2fvNV glad_glVertexAttribs2fvNV
14559 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS2SVNVPROC) (GLuint index, GLsizei count, const GLshort *v);
14560 GLAPI PFNGLVERTEXATTRIBS2SVNVPROC glad_glVertexAttribs2svNV;
14561 #define glVertexAttribs2svNV glad_glVertexAttribs2svNV
14562 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS3DVNVPROC) (GLuint index, GLsizei count, const GLdouble *v);
14563 GLAPI PFNGLVERTEXATTRIBS3DVNVPROC glad_glVertexAttribs3dvNV;
14564 #define glVertexAttribs3dvNV glad_glVertexAttribs3dvNV
14565 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS3FVNVPROC) (GLuint index, GLsizei count, const GLfloat *v);
14566 GLAPI PFNGLVERTEXATTRIBS3FVNVPROC glad_glVertexAttribs3fvNV;
14567 #define glVertexAttribs3fvNV glad_glVertexAttribs3fvNV
14568 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS3SVNVPROC) (GLuint index, GLsizei count, const GLshort *v);
14569 GLAPI PFNGLVERTEXATTRIBS3SVNVPROC glad_glVertexAttribs3svNV;
14570 #define glVertexAttribs3svNV glad_glVertexAttribs3svNV
14571 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS4DVNVPROC) (GLuint index, GLsizei count, const GLdouble *v);
14572 GLAPI PFNGLVERTEXATTRIBS4DVNVPROC glad_glVertexAttribs4dvNV;
14573 #define glVertexAttribs4dvNV glad_glVertexAttribs4dvNV
14574 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS4FVNVPROC) (GLuint index, GLsizei count, const GLfloat *v);
14575 GLAPI PFNGLVERTEXATTRIBS4FVNVPROC glad_glVertexAttribs4fvNV;
14576 #define glVertexAttribs4fvNV glad_glVertexAttribs4fvNV
14577 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS4SVNVPROC) (GLuint index, GLsizei count, const GLshort *v);
14578 GLAPI PFNGLVERTEXATTRIBS4SVNVPROC glad_glVertexAttribs4svNV;
14579 #define glVertexAttribs4svNV glad_glVertexAttribs4svNV
14580 typedef void (APIENTRYTYPE PFNGLVERTEXATTRIBS4UBNVPROC) (GLuint index, GLsizei count, const GLubyte *v);
14581 GLAPI PFNGLVERTEXATTRIBS4UBNVPROC glad_glVertexAttribs4ubNV;
14582 #define glVertexAttribs4ubNV glad_glVertexAttribs4ubNV
14583 #endif
14584 #ifndef GL_NV_vertex_program1_1
14585 #define GL_NV_vertex_program1_1 1
14586 GLAPI int GLAD_GL_NV_vertex_program1_1;
14587 #endif
14588 #ifndef GL_NV_vertex_program2
14589 #define GL_NV_vertex_program2 1
14590 GLAPI int GLAD_GL_NV_vertex_program2;
14591 #endif
14592 #ifndef GL_NV_vertex_program2_option
14593 #define GL_NV_vertex_program2_option 1
14594 GLAPI int GLAD_GL_NV_vertex_program2_option;
14595 #endif
14596 #ifndef GL_NV_vertex_program3
14597 #define GL_NV_vertex_program3 1
14598 GLAPI int GLAD_GL_NV_vertex_program3;
14599 #endif
14600 #ifndef GL_NV_vertex_program4
14601 #define GL_NV_vertex_program4 1
14602 GLAPI int GLAD_GL_NV_vertex_program4;
14603 #endif
14604 #ifndef GL_NV_video_capture
14605 #define GL_NV_video_capture 1
14606 GLAPI int GLAD_GL_NV_video_capture;
14607 typedef void (APIENTRYTYPE PFNGLBEGINVIDEOCAPTURENVPROC) (GLuint video_capture_slot);
14608 GLAPI PFNGLBEGINVIDEOCAPTURENVPROC glad_glBeginVideoCaptureNV;
14609 #define glBeginVideoCaptureNV glad_glBeginVideoCaptureNV
14610 typedef void (APIENTRYTYPE PFNGLBINDVIDEOCAPTURESTREAMBUFFERNVPROC) (GLuint video_capture_slot, GLuint
stream, GLenum frame_region, GLintptrARB offset);
14611 GLAPI PFNGLBINDVIDEOCAPTURESTREAMBUFFERNVPROC glad_glBindVideoCaptureStreamBufferNV;
14612 #define glBindVideoCaptureStreamBufferNV glad_glBindVideoCaptureStreamBufferNV
14613 typedef void (APIENTRYTYPE PFNGLBINDVIDEOCAPTURESTREAMTEXTURENVPROC) (GLuint video_capture_slot, GLuint
stream, GLenum frame_target, GLenum target, GLuint texture);
14614 GLAPI PFNGLBINDVIDEOCAPTURESTREAMTEXTURENVPROC glad_glBindVideoCaptureStreamTextureNV;
14615 #define glBindVideoCaptureStreamTextureNV glad_glBindVideoCaptureStreamTextureNV
14616 typedef void (APIENTRYTYPE PFNGLENDVIDEOCAPTURENVPROC) (GLuint video_capture_slot);
14617 GLAPI PFNGLENDVIDEOCAPTURENVPROC glad_glEndVideoCaptureNV;
14618 #define glEndVideoCaptureNV glad_glEndVideoCaptureNV
14619 typedef void (APIENTRYTYPE PFNGLGETVIDEOCAPTUREIVNVPROC) (GLuint video_capture_slot, GLenum pname, GLint
*params);
14620 GLAPI PFNGLGETVIDEOCAPTUREIVNVPROC glad_glGetVideoCaptureivNV;
14621 #define glGetVideoCaptureivNV glad_glGetVideoCaptureivNV

```



```

14622 typedef void (APIENTRY PFNGLGETVIDEOCAPTURESTREAMIVNVPROC)(GLuint video_capture_slot, GLuint stream,
14623 GLenum pname, GLint *params);
14624 GLAPI PFNGLGETVIDEOCAPTURESTREAMIVNVPROC glad_glGetVideoCaptureStreamivNV;
14625 #define glGetVideoCaptureStreamivNV glad_glGetVideoCaptureStreamivNV
14626 typedef void (APIENTRY PFNGLGETVIDEOCAPTURESTREAMFVNVPROC)(GLuint video_capture_slot, GLuint stream,
14627 GLenum pname, GLfloat *params);
14628 GLAPI PFNGLGETVIDEOCAPTURESTREAMFVNVPROC glad_glGetVideoCaptureStreamfvNV;
14629 #define glGetVideoCaptureStreamfvNV glad_glGetVideoCaptureStreamfvNV
14630 typedef void (APIENTRY PFNGLGETVIDEOCAPTURESTREAMDVNVPROC)(GLuint video_capture_slot, GLuint stream,
14631 GLenum pname, GLdouble *params);
14632 GLAPI PFNGLGETVIDEOCAPTURESTREAMDVNVPROC glad_glGetVideoCaptureStreamdvNV;
14633 #define glGetVideoCaptureStreamdvNV glad_glGetVideoCaptureStreamdvNV
14634 typedef void (APIENTRY PFNGLVIDEOCAPTURENV)(GLuint video_capture_slot, GLuint *sequence_num,
14635 GLuint64EXT *capture_time);
14636 GLAPI PFNGLVIDEOCAPTURENVPROC glad_glVideoCaptureNV;
14637 #define glVideoCaptureNV glad_glVideoCaptureNV
14638 typedef void (APIENTRY PFNGLVIDEOCAPTURESTREAMPARAMETERIVNVPROC)(GLuint video_capture_slot, GLuint
14639 stream, GLenum pname, const GLint *params);
14640 GLAPI PFNGLVIDEOCAPTURESTREAMPARAMETERIVNVPROC glad_glVideoCaptureStreamParameterivNV;
14641 #define glVideoCaptureStreamParameterivNV glad_glVideoCaptureStreamParameterivNV
14642 typedef void (APIENTRY PFNGLVIDEOCAPTURESTREAMPARAMETERFVNVPROC)(GLuint video_capture_slot, GLuint
14643 stream, GLenum pname, const GLfloat *params);
14644 GLAPI PFNGLVIDEOCAPTURESTREAMPARAMETERFVNVPROC glad_glVideoCaptureStreamParameterfvNV;
14645 #define glVideoCaptureStreamParameterfvNV glad_glVideoCaptureStreamParameterfvNV
14646 typedef void (APIENTRY PFNGLVIDEOCAPTURESTREAMPARAMETERDVNVPROC)(GLuint video_capture_slot, GLuint
14647 stream, GLenum pname, const GLdouble *params);
14648 GLAPI PFNGLVIDEOCAPTURESTREAMPARAMETERDVNVPROC glad_glVideoCaptureStreamParameterdvNV;
14649 #define glVideoCaptureStreamParameterdvNV glad_glVideoCaptureStreamParameterdvNV
14650 #endif
14651 #ifndef GL_NV_viewport_array2
14652 #define GL_NV_viewport_array2 1
14653 GLAPI int GLAD_GL_NV_viewport_array2;
14654 #endif
14655 #ifndef GL_NV_viewport_swizzle
14656 #define GL_NV_viewport_swizzle 1
14657 GLAPI int GLAD_GL_NV_viewport_swizzle;
14658 #endif
14659 typedef void (APIENTRY PFNGLVIEWPORTSWIZZLENVPROC)(GLuint index, GLenum swizzle, GLenum swizzley,
14660 GLenum swizzlez, GLenum swizzlew);
14661 GLAPI PFNGLVIEWPORTSWIZZLENVPROC glad_glViewportSwizzleNV;
14662 #define glViewportSwizzleNV glad_glViewportSwizzleNV
14663 #endif
14664 #ifndef GL_OES_byte_coordinates
14665 #define GL_OES_byte_coordinates 1
14666 GLAPI int GLAD_GL_OES_byte_coordinates;
14667 #endif
14668 typedef void (APIENTRY PFNGLMULTITEXCOORD1BOESPROC)(GLenum texture, GLbyte s);
14669 GLAPI PFNGLMULTITEXCOORD1BOESPROC glad_glMultiTexCoord1bOES;
14670 #define glMultiTexCoord1bOES glad_glMultiTexCoord1bOES
14671 typedef void (APIENTRY PFNGLMULTITEXCOORD1BVOESPROC)(GLenum texture, const GLbyte *coords);
14672 GLAPI PFNGLMULTITEXCOORD1BVOESPROC glad_glMultiTexCoord1bvOES;
14673 #define glMultiTexCoord1bvOES glad_glMultiTexCoord1bvOES
14674 typedef void (APIENTRY PFNGLMULTITEXCOORD2BOESPROC)(GLenum texture, GLbyte s, GLbyte t);
14675 GLAPI PFNGLMULTITEXCOORD2BOESPROC glad_glMultiTexCoord2bOES;
14676 #define glMultiTexCoord2bOES glad_glMultiTexCoord2bOES
14677 typedef void (APIENTRY PFNGLMULTITEXCOORD2BVOESPROC)(GLenum texture, const GLbyte *coords);
14678 GLAPI PFNGLMULTITEXCOORD2BVOESPROC glad_glMultiTexCoord2bvOES;
14679 #define glMultiTexCoord2bvOES glad_glMultiTexCoord2bvOES
14680 typedef void (APIENTRY PFNGLMULTITEXCOORD3BOESPROC)(GLenum texture, GLbyte s, GLbyte t, GLbyte r);
14681 GLAPI PFNGLMULTITEXCOORD3BOESPROC glad_glMultiTexCoord3bOES;
14682 #define glMultiTexCoord3bOES glad_glMultiTexCoord3bOES
14683 typedef void (APIENTRY PFNGLMULTITEXCOORD3BVOESPROC)(GLenum texture, const GLbyte *coords);
14684 GLAPI PFNGLMULTITEXCOORD3BVOESPROC glad_glMultiTexCoord3bvOES;
14685 #define glMultiTexCoord3bvOES glad_glMultiTexCoord3bvOES
14686 typedef void (APIENTRY PFNGLMULTITEXCOORD4BOESPROC)(GLenum texture, GLbyte s, GLbyte t, GLbyte r,
14687 GLbyte q);
14688 GLAPI PFNGLMULTITEXCOORD4BOESPROC glad_glMultiTexCoord4bOES;
14689 #define glMultiTexCoord4bOES glad_glMultiTexCoord4bOES
14690 typedef void (APIENTRY PFNGLMULTITEXCOORD4BVOESPROC)(GLenum texture, const GLbyte *coords);
14691 GLAPI PFNGLMULTITEXCOORD4BVOESPROC glad_glMultiTexCoord4bvOES;
14692 #define glMultiTexCoord4bvOES glad_glMultiTexCoord4bvOES
14693 typedef void (APIENTRY PFNGLTEXCOORD1BOESPROC)(GLbyte s);
14694 GLAPI PFNGLTEXCOORD1BOESPROC glad_glTexCoord1bOES;
14695 #define glTexCoord1bOES glad_glTexCoord1bOES
14696 typedef void (APIENTRY PFNGLTEXCOORD1BVOESPROC)(const GLbyte *coords);
14697 GLAPI PFNGLTEXCOORD1BVOESPROC glad_glTexCoord1bvOES;
14698 #define glTexCoord1bvOES glad_glTexCoord1bvOES
14699 typedef void (APIENTRY PFNGLTEXCOORD2BOESPROC)(GLbyte s, GLbyte t);
14700 GLAPI PFNGLTEXCOORD2BOESPROC glad_glTexCoord2bOES;
14701 #define glTexCoord2bOES glad_glTexCoord2bOES
14702 typedef void (APIENTRY PFNGLTEXCOORD2BVOESPROC)(const GLbyte *coords);
14703 GLAPI PFNGLTEXCOORD2BVOESPROC glad_glTexCoord2bvOES;
14704 #define glTexCoord2bvOES glad_glTexCoord2bvOES
14705 typedef void (APIENTRY PFNGLTEXCOORD3BOESPROC)(GLbyte s, GLbyte t, GLbyte r);
14706 GLAPI PFNGLTEXCOORD3BOESPROC glad_glTexCoord3bOES;
14707 #define glTexCoord3bOES glad_glTexCoord3bOES
14708 typedef void (APIENTRY PFNGLTEXCOORD3BVOESPROC)(const GLbyte *coords);
14709 GLAPI PFNGLTEXCOORD3BVOESPROC glad_glTexCoord3bvOES;
14710 #define glTexCoord3bvOES glad_glTexCoord3bvOES

```

```

14700 typedef void (APIENTRY PFNGLTEXCOORD4BOESPROC)(GLbyte s, GLbyte t, GLbyte r, GLbyte q);
14701 GLAPI PFNGLTEXCOORD4BOESPROC glad_glTexCoord4bOES;
14702 #define glTexCoord4bOES glad_glTexCoord4bOES
14703 typedef void (APIENTRY PFNGLTEXCOORD4BVOESPROC)(const GLbyte *coords);
14704 GLAPI PFNGLTEXCOORD4BVOESPROC glad_glTexCoord4bvOES;
14705 #define glTexCoord4bvOES glad_glTexCoord4bvOES
14706 typedef void (APIENTRY PFNGLVERTEX2BOESPROC)(GLbyte x, GLbyte y);
14707 GLAPI PFNGLVERTEX2BOESPROC glad_glVertex2bOES;
14708 #define glVertex2bOES glad_glVertex2bOES
14709 typedef void (APIENTRY PFNGLVERTEX2BVOESPROC)(const GLbyte *coords);
14710 GLAPI PFNGLVERTEX2BVOESPROC glad_glVertex2bvOES;
14711 #define glVertex2bvOES glad_glVertex2bvOES
14712 typedef void (APIENTRY PFNGLVERTEX3BOESPROC)(GLbyte x, GLbyte y, GLbyte z);
14713 GLAPI PFNGLVERTEX3BOESPROC glad_glVertex3bOES;
14714 #define glVertex3bOES glad_glVertex3bOES
14715 typedef void (APIENTRY PFNGLVERTEX3BVOESPROC)(const GLbyte *coords);
14716 GLAPI PFNGLVERTEX3BVOESPROC glad_glVertex3bvOES;
14717 #define glVertex3bvOES glad_glVertex3bvOES
14718 typedef void (APIENTRY PFNGLVERTEX4BOESPROC)(GLbyte x, GLbyte y, GLbyte z, GLbyte w);
14719 GLAPI PFNGLVERTEX4BOESPROC glad_glVertex4bOES;
14720 #define glVertex4bOES glad_glVertex4bOES
14721 typedef void (APIENTRY PFNGLVERTEX4BVOESPROC)(const GLbyte *coords);
14722 GLAPI PFNGLVERTEX4BVOESPROC glad_glVertex4bvOES;
14723 #define glVertex4bvOES glad_glVertex4bvOES
14724 #endif
14725 #ifndef GL_OES_compressed_paletted_texture
14726 #define GL_OES_compressed_paletted_texture 1
14727 GLAPI int GLAD_GL_OES_compressed_paletted_texture;
14728 #endif
14729 #ifndef GL_OES_fixed_point
14730 #define GL_OES_fixed_point 1
14731 GLAPI int GLAD_GL_OES_fixed_point;
14732 typedef void (APIENTRY PFNGLALPHAFUNCXOESPROC)(GLenum func, GLfixed ref);
14733 GLAPI PFNGLALPHAFUNCXOESPROC glad_glAlphaFuncxOES;
14734 #define glAlphaFuncxOES glad_glAlphaFuncxOES
14735 typedef void (APIENTRY PFNGLCLEARCOLORXOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed
alpha);
14736 GLAPI PFNGLCLEARCOLORXOESPROC glad_glClearColorxOES;
14737 #define glClearColorxOES glad_glClearColorxOES
14738 typedef void (APIENTRY PFNGLCLEARDEPTHXOESPROC)(GLfixed depth);
14739 GLAPI PFNGLCLEARDEPTHXOESPROC glad_glClearDepthxOES;
14740 #define glClearDepthxOES glad_glClearDepthxOES
14741 typedef void (APIENTRY PFNGLCLIPPLANEXOESPROC)(GLenum plane, const GLfixed *equation);
14742 GLAPI PFNGLCLIPPLANEXOESPROC glad_glClipPlanexOES;
14743 #define glClipPlanexOES glad_glClipPlanexOES
14744 typedef void (APIENTRY PFNGLCOLOR4XOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed alpha);
14745 GLAPI PFNGLCOLOR4XOESPROC glad_glColor4xOES;
14746 #define glColor4xOES glad_glColor4xOES
14747 typedef void (APIENTRY PFNGLDEPTHRANGEXOESPROC)(GLfixed n, GLfixed f);
14748 GLAPI PFNGLDEPTHRANGEXOESPROC glad_glDepthRangexOES;
14749 #define glDepthRangexOES glad_glDepthRangexOES
14750 typedef void (APIENTRY PFNGLFOGXOESPROC)(GLenum pname, GLfixed param);
14751 GLAPI PFNGLFOGXOESPROC glad_glFogxOES;
14752 #define glFogxOES glad_glFogxOES
14753 typedef void (APIENTRY PFNGLFOGXVOESPROC)(GLenum pname, const GLfixed *param);
14754 GLAPI PFNGLFOGXVOESPROC glad_glFogxvOES;
14755 #define glFogxvOES glad_glFogxvOES
14756 typedef void (APIENTRY PFNGLFRUSTUMXOESPROC)(GLfixed l, GLfixed r, GLfixed b, GLfixed t, GLfixed n,
GLfixed f);
14757 GLAPI PFNGLFRUSTUMXOESPROC glad_glFrustumxOES;
14758 #define glFrustumxOES glad_glFrustumxOES
14759 typedef void (APIENTRY PFNGLGETCLIPPLANEXOESPROC)(GLenum plane, GLfixed *equation);
14760 GLAPI PFNGLGETCLIPPLANEXOESPROC glad_glGetClipPlanexOES;
14761 #define glGetClipPlanexOES glad_glGetClipPlanexOES
14762 typedef void (APIENTRY PFNGLGETFIXEDVOESPROC)(GLenum pname, GLfixed *params);
14763 GLAPI PFNGLGETFIXEDVOESPROC glad_glGetFixedvOES;
14764 #define glGetFixedvOES glad_glGetFixedvOES
14765 typedef void (APIENTRY PFNGLGETTEXENVXVOESPROC)(GLenum target, GLenum pname, GLfixed *params);
14766 GLAPI PFNGLGETTEXENVXVOESPROC glad_glGetTexEnvxvOES;
14767 #define glGetTexEnvxvOES glad_glGetTexEnvxvOES
14768 typedef void (APIENTRY PFNGLGETTEXPARAMETERXVOESPROC)(GLenum target, GLenum pname, GLfixed *params);
14769 GLAPI PFNGLGETTEXPARAMETERXVOESPROC glad_glGetTexParameterxvOES;
14770 #define glGetTexParameterxvOES glad_glGetTexParameterxvOES
14771 typedef void (APIENTRY PFNGLLIGHTMODELXOESPROC)(GLenum pname, GLfixed param);
14772 GLAPI PFNGLLIGHTMODELXOESPROC glad_glLightModelxOES;
14773 #define glLightModelxOES glad_glLightModelxOES
14774 typedef void (APIENTRY PFNGLLIGHTMODELXVOESPROC)(GLenum pname, const GLfixed *param);
14775 GLAPI PFNGLLIGHTMODELXVOESPROC glad_glLightModelxvOES;
14776 #define glLightModelxvOES glad_glLightModelxvOES
14777 typedef void (APIENTRY PFNGLLIGHTXOESPROC)(GLenum light, GLenum pname, GLfixed param);
14778 GLAPI PFNGLLIGHTXOESPROC glad_glLightxOES;
14779 #define glLightxOES glad_glLightxOES
14780 typedef void (APIENTRY PFNGLLIGHTXVOESPROC)(GLenum light, GLenum pname, const GLfixed *params);
14781 GLAPI PFNGLLIGHTXVOESPROC glad_glLightxvOES;
14782 #define glLightxvOES glad_glLightxvOES
14783 typedef void (APIENTRY PFNGLLINEWIDTHXOESPROC)(GLfixed width);
14784 GLAPI PFNGLLINEWIDTHXOESPROC glad_glLineWidthxOES;

```

```

14785 #define glLineWidthxOES glad_glLineWidthxOES
14786 typedef void (APIENTRY PFNGLLOADMATRIXVOESPROC)(const GLfixed *m);
14787 GLAPI PFNGLLOADMATRIXVOESPROC glad_glLoadMatrixxOES;
14788 #define glLoadMatrixxOES glad_glLoadMatrixxOES
14789 typedef void (APIENTRY PFNGLMATERIALXVOESPROC)(GLenum face, GLenum pname, GLfixed param);
14790 GLAPI PFNGLMATERIALXVOESPROC glad_glMaterialxOES;
14791 #define glMaterialxOES glad_glMaterialxOES
14792 typedef void (APIENTRY PFNGLMATERIALXVOESPROC)(GLenum face, GLenum pname, const GLfixed *param);
14793 GLAPI PFNGLMATERIALXVOESPROC glad_glMaterialxvOES;
14794 #define glMaterialxvOES glad_glMaterialxvOES
14795 typedef void (APIENTRY PFNGLMULTMATRIXXVOESPROC)(const GLfixed *m);
14796 GLAPI PFNGLMULTMATRIXXVOESPROC glad_glMultMatrixxOES;
14797 #define glMultMatrixxOES glad_glMultMatrixxOES
14798 typedef void (APIENTRY PFNGLMULTITEXCOORD4XVOESPROC)(GLenum texture, GLfixed s, GLfixed t, GLfixed r,
GLfixed q);
14799 GLAPI PFNGLMULTITEXCOORD4XVOESPROC glad_glMultiTexCoord4xOES;
14800 #define glMultiTexCoord4xOES glad_glMultiTexCoord4xOES
14801 typedef void (APIENTRY PFNGLNORMAL3XVOESPROC)(GLfixed nx, GLfixed ny, GLfixed nz);
14802 GLAPI PFNGLNORMAL3XVOESPROC glad_glNormal3xOES;
14803 #define glNormal3xOES glad_glNormal3xOES
14804 typedef void (APIENTRY PFNGLORTHOGONEXVOESPROC)(GLfixed l, GLfixed r, GLfixed b, GLfixed t, GLfixed n,
GLfixed f);
14805 GLAPI PFNGLORTHOGONEXVOESPROC glad_glOrthoxOES;
14806 #define glOrthoxOES glad_glOrthoxOES
14807 typedef void (APIENTRY PFNGLPOINTPARAMETERXVOESPROC)(GLenum pname, const GLfixed *params);
14808 GLAPI PFNGLPOINTPARAMETERXVOESPROC glad_glPointParameterxvOES;
14809 #define glPointParameterxvOES glad_glPointParameterxvOES
14810 typedef void (APIENTRY PFNGLPOINTSIZEXVOESPROC)(GLfixed size);
14811 GLAPI PFNGLPOINTSIZEXVOESPROC glad_glPointSizexOES;
14812 #define glPointSizexOES glad_glPointSizexOES
14813 typedef void (APIENTRY PFNGLPOLYGONOFFSETXVOESPROC)(GLfixed factor, GLfixed units);
14814 GLAPI PFNGLPOLYGONOFFSETXVOESPROC glad_glPolygonOffsetxOES;
14815 #define glPolygonOffsetxOES glad_glPolygonOffsetxOES
14816 typedef void (APIENTRY PFNGLROTATEXVOESPROC)(GLfixed angle, GLfixed x, GLfixed y, GLfixed z);
14817 GLAPI PFNGLROTATEXVOESPROC glad_glRotatexOES;
14818 #define glRotatexOES glad_glRotatexOES
14819 typedef void (APIENTRY PFNGLSCALEXVOESPROC)(GLfixed x, GLfixed y, GLfixed z);
14820 GLAPI PFNGLSCALEXVOESPROC glad_glScalexOES;
14821 #define glScalexOES glad_glScalexOES
14822 typedef void (APIENTRY PFNGLTEXENVXVOESPROC)(GLenum target, GLenum pname, GLfixed param);
14823 GLAPI PFNGLTEXENVXVOESPROC glad_glTexEnvxOES;
14824 #define glTexEnvxOES glad_glTexEnvxOES
14825 typedef void (APIENTRY PFNGLTEXENVXVOESPROC)(GLenum target, GLenum pname, const GLfixed *params);
14826 GLAPI PFNGLTEXENVXVOESPROC glad_glTexEnvxvOES;
14827 #define glTexEnvxvOES glad_glTexEnvxvOES
14828 typedef void (APIENTRY PFNGLTEXPARAMETERXVOESPROC)(GLenum target, GLenum pname, GLfixed param);
14829 GLAPI PFNGLTEXPARAMETERXVOESPROC glad_glTexParameterxOES;
14830 #define glTexParameterxOES glad_glTexParameterxOES
14831 typedef void (APIENTRY PFNGLTEXPARAMETERXVOESPROC)(GLenum target, GLenum pname, const GLfixed
*params);
14832 GLAPI PFNGLTEXPARAMETERXVOESPROC glad_glTexParameterxvOES;
14833 #define glTexParameterxvOES glad_glTexParameterxvOES
14834 typedef void (APIENTRY PFNGLTRANSLATEXVOESPROC)(GLfixed x, GLfixed y, GLfixed z);
14835 GLAPI PFNGLTRANSLATEXVOESPROC glad_glTranslatexOES;
14836 #define glTranslatexOES glad_glTranslatexOES
14837 typedef void (APIENTRY PFNGLGETLIGHTXVOESPROC)(GLenum light, GLenum pname, GLfixed *params);
14838 GLAPI PFNGLGETLIGHTXVOESPROC glad_glGetLightxvOES;
14839 #define glGetLightxvOES glad_glGetLightxvOES
14840 typedef void (APIENTRY PFNGLGETMATERIALXVOESPROC)(GLenum face, GLenum pname, GLfixed *params);
14841 GLAPI PFNGLGETMATERIALXVOESPROC glad_glGetMaterialxvOES;
14842 #define glGetMaterialxvOES glad_glGetMaterialxvOES
14843 typedef void (APIENTRY PFNGLPOINTPARAMETERXVOESPROC)(GLenum pname, GLfixed param);
14844 GLAPI PFNGLPOINTPARAMETERXVOESPROC glad_glPointParameterxOES;
14845 #define glPointParameterxOES glad_glPointParameterxOES
14846 typedef void (APIENTRY PFNGLSAMPLECOVERAGEXVOESPROC)(GLclampx value, GLboolean invert);
14847 GLAPI PFNGLSAMPLECOVERAGEXVOESPROC glad_glSampleCoveragexOES;
14848 #define glSampleCoveragexOES glad_glSampleCoveragexOES
14849 typedef void (APIENTRY PFNGLACCUMXVOESPROC)(GLenum op, GLfixed value);
14850 GLAPI PFNGLACCUMXVOESPROC glad_glAccumxOES;
14851 #define glAccumxOES glad_glAccumxOES
14852 typedef void (APIENTRY PFNGLBITMAPXVOESPROC)(GLsizei width, GLsizei height, GLfixed xorig, GLfixed
yorig, GLfixed xmove, GLfixed ymove, const GLubyte *bitmap);
14853 GLAPI PFNGLBITMAPXVOESPROC glad_glBitmapxOES;
14854 #define glBitmapxOES glad_glBitmapxOES
14855 typedef void (APIENTRY PFNGLBLENDCOLORXVOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed
alpha);
14856 GLAPI PFNGLBLENDCOLORXVOESPROC glad_glBlendColorxOES;
14857 #define glBlendColorxOES glad_glBlendColorxOES
14858 typedef void (APIENTRY PFNGLCLEARACCUMXVOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed
alpha);
14859 GLAPI PFNGLCLEARACCUMXVOESPROC glad_glClearAccumxOES;
14860 #define glClearAccumxOES glad_glClearAccumxOES
14861 typedef void (APIENTRY PFNGLCOLOR3XVOESPROC)(GLfixed red, GLfixed green, GLfixed blue);
14862 GLAPI PFNGLCOLOR3XVOESPROC glad_glColor3xOES;
14863 #define glColor3xOES glad_glColor3xOES
14864 typedef void (APIENTRY PFNGLCOLOR3XVOESPROC)(const GLfixed *components);
14865 GLAPI PFNGLCOLOR3XVOESPROC glad_glColor3xvOES;

```

```

14866 #define glColor3xvOES glad_glColor3xvOES
14867 typedef void (APIENTRY PFNGLCOLOR4XVOESPROC)(const GLfixed *components);
14868 GLAPI PFNGLCOLOR4XVOESPROC glad_glColor4xvOES;
14869 #define glColor4xvOES glad_glColor4xvOES
14870 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERXOESPROC)(GLenum target, GLenum pname, GLfixed
    param);
14871 GLAPI PFNGLCONVOLUTIONPARAMETERXOESPROC glad_glConvolutionParameterxvOES;
14872 #define glConvolutionParameterxvOES glad_glConvolutionParameterxvOES
14873 typedef void (APIENTRY PFNGLCONVOLUTIONPARAMETERXVOESPROC)(GLenum target, GLenum pname, const GLfixed
    *params);
14874 GLAPI PFNGLCONVOLUTIONPARAMETERXVOESPROC glad_glConvolutionParameterxvOES;
14875 #define glConvolutionParameterxvOES glad_glConvolutionParameterxvOES
14876 typedef void (APIENTRY PFNGLEVALCOORD1XOESPROC)(GLfixed u);
14877 GLAPI PFNGLEVALCOORD1XOESPROC glad_glEvalCoord1xvOES;
14878 #define glEvalCoord1xvOES glad_glEvalCoord1xvOES
14879 typedef void (APIENTRY PFNGLEVALCOORD1XVOESPROC)(const GLfixed *coords);
14880 GLAPI PFNGLEVALCOORD1XVOESPROC glad_glEvalCoord1xvOES;
14881 #define glEvalCoord1xvOES glad_glEvalCoord1xvOES
14882 typedef void (APIENTRY PFNGLEVALCOORD2XOESPROC)(GLfixed u, GLfixed v);
14883 GLAPI PFNGLEVALCOORD2XOESPROC glad_glEvalCoord2xvOES;
14884 #define glEvalCoord2xvOES glad_glEvalCoord2xvOES
14885 typedef void (APIENTRY PFNGLEVALCOORD2XVOESPROC)(const GLfixed *coords);
14886 GLAPI PFNGLEVALCOORD2XVOESPROC glad_glEvalCoord2xvOES;
14887 #define glEvalCoord2xvOES glad_glEvalCoord2xvOES
14888 typedef void (APIENTRY PFNGLFEEDBACKBUFFERXOESPROC)(GLsizei n, GLenum type, const GLfixed *buffer);
14889 GLAPI PFNGLFEEDBACKBUFFERXOESPROC glad_glFeedbackBufferxvOES;
14890 #define glFeedbackBufferxvOES glad_glFeedbackBufferxvOES
14891 typedef void (APIENTRY PFNGLGETCONVOLUTIONPARAMETERXVOESPROC)(GLenum target, GLenum pname, GLfixed
    *params);
14892 GLAPI PFNGLGETCONVOLUTIONPARAMETERXVOESPROC glad_glGetConvolutionParameterxvOES;
14893 #define glGetConvolutionParameterxvOES glad_glGetConvolutionParameterxvOES
14894 typedef void (APIENTRY PFNGLGETHISTOGRAMPARAMETERXVOESPROC)(GLenum target, GLenum pname, GLfixed
    *params);
14895 GLAPI PFNGLGETHISTOGRAMPARAMETERXVOESPROC glad_glGetHistogramParameterxvOES;
14896 #define glGetHistogramParameterxvOES glad_glGetHistogramParameterxvOES
14897 typedef void (APIENTRY PFNGLGETLIGHTXOESPROC)(GLenum light, GLenum pname, GLfixed *params);
14898 GLAPI PFNGLGETLIGHTXOESPROC glad_glGetLightxvOES;
14899 #define glGetLightxvOES glad_glGetLightxvOES
14900 typedef void (APIENTRY PFNGLGETMAPXVOESPROC)(GLenum target, GLenum query, GLfixed *v);
14901 GLAPI PFNGLGETMAPXVOESPROC glad_glGetMapxvOES;
14902 #define glGetMapxvOES glad_glGetMapxvOES
14903 typedef void (APIENTRY PFNGLGETMATERIALXOESPROC)(GLenum face, GLenum pname, GLfixed param);
14904 GLAPI PFNGLGETMATERIALXOESPROC glad_glGetMaterialxvOES;
14905 #define glGetMaterialxvOES glad_glGetMaterialxvOES
14906 typedef void (APIENTRY PFNGLGETPIXELMAPXVPROC)(GLenum map, GLint size, GLfixed *values);
14907 GLAPI PFNGLGETPIXELMAPXVPROC glad_glGetPixelMapxv;
14908 #define glGetPixelMapxv glad_glGetPixelMapxv
14909 typedef void (APIENTRY PFNGLGETTEXGENXVOESPROC)(GLenum coord, GLenum pname, GLfixed *params);
14910 GLAPI PFNGLGETTEXGENXVOESPROC glad_glGetTexGenxvOES;
14911 #define glGetTexGenxvOES glad_glGetTexGenxvOES
14912 typedef void (APIENTRY PFNGLGETTEXLEVELPARAMETERXVOESPROC)(GLenum target, GLint level, GLenum pname,
    GLfixed *params);
14913 GLAPI PFNGLGETTEXLEVELPARAMETERXVOESPROC glad_glGetTexLevelParameterxvOES;
14914 #define glGetTexLevelParameterxvOES glad_glGetTexLevelParameterxvOES
14915 typedef void (APIENTRY PFNGLINDEXXOESPROC)(GLfixed component);
14916 GLAPI PFNGLINDEXXOESPROC glad_glIndexxvOES;
14917 #define glIndexxvOES glad_glIndexxvOES
14918 typedef void (APIENTRY PFNGLINDEXXVOESPROC)(const GLfixed *component);
14919 GLAPI PFNGLINDEXXVOESPROC glad_glIndexxvOES;
14920 #define glIndexxvOES glad_glIndexxvOES
14921 typedef void (APIENTRY PFNGLLOADTRANSPOSEMATRIXXOESPROC)(const GLfixed *m);
14922 GLAPI PFNGLLOADTRANSPOSEMATRIXXOESPROC glad_glLoadTransposeMatrixxvOES;
14923 #define glLoadTransposeMatrixxvOES glad_glLoadTransposeMatrixxvOES
14924 typedef void (APIENTRY PFNGLMAP1XOESPROC)(GLenum target, GLfixed u1, GLfixed u2, GLint stride, GLint
    order, GLfixed points);
14925 GLAPI PFNGLMAP1XOESPROC glad_glMap1xvOES;
14926 #define glMap1xvOES glad_glMap1xvOES
14927 typedef void (APIENTRY PFNGLMAP2XOESPROC)(GLenum target, GLfixed u1, GLfixed u2, GLint ustride, GLint
    uorder, GLfixed v1, GLfixed v2, GLint vstride, GLint vorder, GLfixed points);
14928 GLAPI PFNGLMAP2XOESPROC glad_glMap2xvOES;
14929 #define glMap2xvOES glad_glMap2xvOES
14930 typedef void (APIENTRY PFNGLMAPGRIDLXOESPROC)(GLint n, GLfixed u1, GLfixed u2);
14931 GLAPI PFNGLMAPGRIDLXOESPROC glad_glMapGrid1xvOES;
14932 #define glMapGrid1xvOES glad_glMapGrid1xvOES
14933 typedef void (APIENTRY PFNGLMAPGRID2XOESPROC)(GLint n, GLfixed u1, GLfixed u2, GLfixed v1, GLfixed
    v2);
14934 GLAPI PFNGLMAPGRID2XOESPROC glad_glMapGrid2xvOES;
14935 #define glMapGrid2xvOES glad_glMapGrid2xvOES
14936 typedef void (APIENTRY PFNGLMULTTRANSPOSEMATRIXXOESPROC)(const GLfixed *m);
14937 GLAPI PFNGLMULTTRANSPOSEMATRIXXOESPROC glad_glMultTransposeMatrixxvOES;
14938 #define glMultTransposeMatrixxvOES glad_glMultTransposeMatrixxvOES
14939 typedef void (APIENTRY PFNGLMULTITEXCOORD1XOESPROC)(GLenum texture, GLfixed s);
14940 GLAPI PFNGLMULTITEXCOORD1XOESPROC glad_glMultiTexCoord1xvOES;
14941 #define glMultiTexCoord1xvOES glad_glMultiTexCoord1xvOES
14942 typedef void (APIENTRY PFNGLMULTITEXCOORD1XVOESPROC)(GLenum texture, const GLfixed *coords);
14943 GLAPI PFNGLMULTITEXCOORD1XVOESPROC glad_glMultiTexCoord1xvOES;
14944 #define glMultiTexCoord1xvOES glad_glMultiTexCoord1xvOES

```



```

14945 typedef void (APIENTRY PFNGLMULTITEXCOORD2XOESPROC) (GLenum texture, GLfixed s, GLfixed t);
14946 GLAPI PFNGLMULTITEXCOORD2XOESPROC glad_glMultiTexCoord2xOES;
14947 #define glMultiTexCoord2xOES glad_glMultiTexCoord2xOES
14948 typedef void (APIENTRY PFNGLMULTITEXCOORD2XVOESPROC) (GLenum texture, const GLfixed *coords);
14949 GLAPI PFNGLMULTITEXCOORD2XVOESPROC glad_glMultiTexCoord2xVOES;
14950 #define glMultiTexCoord2xVOES glad_glMultiTexCoord2xVOES
14951 typedef void (APIENTRY PFNGLMULTITEXCOORD3XOESPROC) (GLenum texture, GLfixed s, GLfixed t, GLfixed r);
14952 GLAPI PFNGLMULTITEXCOORD3XOESPROC glad_glMultiTexCoord3xOES;
14953 #define glMultiTexCoord3xOES glad_glMultiTexCoord3xOES
14954 typedef void (APIENTRY PFNGLMULTITEXCOORD3XVOESPROC) (GLenum texture, const GLfixed *coords);
14955 GLAPI PFNGLMULTITEXCOORD3XVOESPROC glad_glMultiTexCoord3xVOES;
14956 #define glMultiTexCoord3xVOES glad_glMultiTexCoord3xVOES
14957 typedef void (APIENTRY PFNGLMULTITEXCOORD4XOESPROC) (GLenum texture, const GLfixed *coords);
14958 GLAPI PFNGLMULTITEXCOORD4XOESPROC glad_glMultiTexCoord4xVOES;
14959 #define glMultiTexCoord4xVOES glad_glMultiTexCoord4xVOES
14960 typedef void (APIENTRY PFNGLNORMAL3XVOESPROC) (const GLfixed *coords);
14961 GLAPI PFNGLNORMAL3XVOESPROC glad_glNormal3xVOES;
14962 #define glNormal3xVOES glad_glNormal3xVOES
14963 typedef void (APIENTRY PFNGLPASSTHROUGHXOESPROC) (GLfixed token);
14964 GLAPI PFNGLPASSTHROUGHXOESPROC glad_glPassThroughxOES;
14965 #define glPassThroughxOES glad_glPassThroughxOES
14966 typedef void (APIENTRY PFNGLPIXELMAPXPROC) (GLenum map, GLint size, const GLfixed *values);
14967 GLAPI PFNGLPIXELMAPXPROC glad_glPixelMapx;
14968 #define glPixelMapx glad_glPixelMapx
14969 typedef void (APIENTRY PFNGLPIXELSTOREXPROC) (GLenum pname, GLfixed param);
14970 GLAPI PFNGLPIXELSTOREXPROC glad_glPixelStorex;
14971 #define glPixelStorex glad_glPixelStorex
14972 typedef void (APIENTRY PFNGLPIXELTRANSFERXOESPROC) (GLenum pname, GLfixed param);
14973 GLAPI PFNGLPIXELTRANSFERXOESPROC glad_glPixelTransferxOES;
14974 #define glPixelTransferxOES glad_glPixelTransferxOES
14975 typedef void (APIENTRY PFNGLPIXELZOOMXOESPROC) (GLfixed xfactor, GLfixed yfactor);
14976 GLAPI PFNGLPIXELZOOMXOESPROC glad_glPixelZoomxOES;
14977 #define glPixelZoomxOES glad_glPixelZoomxOES
14978 typedef void (APIENTRY PFNGLPRIORITIZETEXTURESXOESPROC) (GLsizei n, const GLuint *textures, const
    GLfixed *priorities);
14979 GLAPI PFNGLPRIORITIZETEXTURESXOESPROC glad_glPrioritizeTexturesxOES;
14980 #define glPrioritizeTexturesxOES glad_glPrioritizeTexturesxOES
14981 typedef void (APIENTRY PFNGLRASTERPOS2XOESPROC) (GLfixed x, GLfixed y);
14982 GLAPI PFNGLRASTERPOS2XOESPROC glad_glRasterPos2xOES;
14983 #define glRasterPos2xOES glad_glRasterPos2xOES
14984 typedef void (APIENTRY PFNGLRASTERPOS2XVOESPROC) (const GLfixed *coords);
14985 GLAPI PFNGLRASTERPOS2XVOESPROC glad_glRasterPos2xVOES;
14986 #define glRasterPos2xVOES glad_glRasterPos2xVOES
14987 typedef void (APIENTRY PFNGLRASTERPOS3XOESPROC) (GLfixed x, GLfixed y, GLfixed z);
14988 GLAPI PFNGLRASTERPOS3XOESPROC glad_glRasterPos3xOES;
14989 #define glRasterPos3xOES glad_glRasterPos3xOES
14990 typedef void (APIENTRY PFNGLRASTERPOS3XVOESPROC) (const GLfixed *coords);
14991 GLAPI PFNGLRASTERPOS3XVOESPROC glad_glRasterPos3xVOES;
14992 #define glRasterPos3xVOES glad_glRasterPos3xVOES
14993 typedef void (APIENTRY PFNGLRASTERPOS4XOESPROC) (GLfixed x, GLfixed y, GLfixed z, GLfixed w);
14994 GLAPI PFNGLRASTERPOS4XOESPROC glad_glRasterPos4xOES;
14995 #define glRasterPos4xOES glad_glRasterPos4xOES
14996 typedef void (APIENTRY PFNGLRASTERPOS4XVOESPROC) (const GLfixed *coords);
14997 GLAPI PFNGLRASTERPOS4XVOESPROC glad_glRasterPos4xVOES;
14998 #define glRasterPos4xVOES glad_glRasterPos4xVOES
14999 typedef void (APIENTRY PFNGLRECTXOESPROC) (GLfixed x1, GLfixed y1, GLfixed x2, GLfixed y2);
15000 GLAPI PFNGLRECTXOESPROC glad_glRectxOES;
15001 #define glRectxOES glad_glRectxOES
15002 typedef void (APIENTRY PFNGLRECTXVOESPROC) (const GLfixed *v1, const GLfixed *v2);
15003 GLAPI PFNGLRECTXVOESPROC glad_glRectxVOES;
15004 #define glRectxVOES glad_glRectxVOES
15005 typedef void (APIENTRY PFNGLTEXCOORD1XOESPROC) (GLfixed s);
15006 GLAPI PFNGLTEXCOORD1XOESPROC glad_glTexCoord1xOES;
15007 #define glTexCoord1xOES glad_glTexCoord1xOES
15008 typedef void (APIENTRY PFNGLTEXCOORD1XVOESPROC) (const GLfixed *coords);
15009 GLAPI PFNGLTEXCOORD1XVOESPROC glad_glTexCoord1xVOES;
15010 #define glTexCoord1xVOES glad_glTexCoord1xVOES
15011 typedef void (APIENTRY PFNGLTEXCOORD2XOESPROC) (GLfixed s, GLfixed t);
15012 GLAPI PFNGLTEXCOORD2XOESPROC glad_glTexCoord2xOES;
15013 #define glTexCoord2xOES glad_glTexCoord2xOES
15014 typedef void (APIENTRY PFNGLTEXCOORD2XVOESPROC) (const GLfixed *coords);
15015 GLAPI PFNGLTEXCOORD2XVOESPROC glad_glTexCoord2xVOES;
15016 #define glTexCoord2xVOES glad_glTexCoord2xVOES
15017 typedef void (APIENTRY PFNGLTEXCOORD3XOESPROC) (GLfixed s, GLfixed t, GLfixed r);
15018 GLAPI PFNGLTEXCOORD3XOESPROC glad_glTexCoord3xOES;
15019 #define glTexCoord3xOES glad_glTexCoord3xOES
15020 typedef void (APIENTRY PFNGLTEXCOORD3XVOESPROC) (const GLfixed *coords);
15021 GLAPI PFNGLTEXCOORD3XVOESPROC glad_glTexCoord3xVOES;
15022 #define glTexCoord3xVOES glad_glTexCoord3xVOES
15023 typedef void (APIENTRY PFNGLTEXCOORD4XOESPROC) (GLfixed s, GLfixed t, GLfixed r, GLfixed q);
15024 GLAPI PFNGLTEXCOORD4XOESPROC glad_glTexCoord4xOES;
15025 #define glTexCoord4xOES glad_glTexCoord4xOES
15026 typedef void (APIENTRY PFNGLTEXCOORD4XVOESPROC) (const GLfixed *coords);
15027 GLAPI PFNGLTEXCOORD4XVOESPROC glad_glTexCoord4xVOES;
15028 #define glTexCoord4xVOES glad_glTexCoord4xVOES
15029 typedef void (APIENTRY PFNGLTEXGENXOESPROC) (GLenum coord, GLenum pname, GLfixed param);
15030 GLAPI PFNGLTEXGENXOESPROC glad_glTexGenxOES;

```

```

15031 #define glTexGenxOES glad_glTexGenxOES
15032 typedef void (APIENTRY PFNGLTEXGENXVOESPROC) (GLenum coord, GLenum pname, const GLfixed *params);
15033 GLAPI PFNGLTEXGENXVOESPROC glad_glTexGenxvOES;
15034 #define glTexGenxvOES glad_glTexGenxvOES
15035 typedef void (APIENTRY PFNGLVERTEX2XOESPROC) (GLfixed x);
15036 GLAPI PFNGLVERTEX2XOESPROC glad_glVertex2xOES;
15037 #define glVertex2xOES glad_glVertex2xOES
15038 typedef void (APIENTRY PFNGLVERTEX2XVOESPROC) (const GLfixed *coords);
15039 GLAPI PFNGLVERTEX2XVOESPROC glad_glVertex2xvOES;
15040 #define glVertex2xvOES glad_glVertex2xvOES
15041 typedef void (APIENTRY PFNGLVERTEX3XOESPROC) (GLfixed x, GLfixed y);
15042 GLAPI PFNGLVERTEX3XOESPROC glad_glVertex3xOES;
15043 #define glVertex3xOES glad_glVertex3xOES
15044 typedef void (APIENTRY PFNGLVERTEX3XVOESPROC) (const GLfixed *coords);
15045 GLAPI PFNGLVERTEX3XVOESPROC glad_glVertex3xvOES;
15046 #define glVertex3xvOES glad_glVertex3xvOES
15047 typedef void (APIENTRY PFNGLVERTEX4XOESPROC) (GLfixed x, GLfixed y, GLfixed z);
15048 GLAPI PFNGLVERTEX4XOESPROC glad_glVertex4xOES;
15049 #define glVertex4xOES glad_glVertex4xOES
15050 typedef void (APIENTRY PFNGLVERTEX4XVOESPROC) (const GLfixed *coords);
15051 GLAPI PFNGLVERTEX4XVOESPROC glad_glVertex4xvOES;
15052 #define glVertex4xvOES glad_glVertex4xvOES
15053 #endif
15054 #ifndef GL_OES_query_matrix
15055 #define GL_OES_query_matrix 1
15056 GLAPI int GLAD_GL_OES_query_matrix;
15057 typedef GLbitfield (APIENTRY PFNGLQUERYMATRIXXOESPROC) (GLfixed *mantissa, GLint *exponent);
15058 GLAPI PFNGLQUERYMATRIXXOESPROC glad_glQueryMatrixxOES;
15059 #define glQueryMatrixxOES glad_glQueryMatrixxOES
15060 #endif
15061 #ifndef GL_OES_read_format
15062 #define GL_OES_read_format 1
15063 GLAPI int GLAD_GL_OES_read_format;
15064 #endif
15065 #ifndef GL_OES_single_precision
15066 #define GL_OES_single_precision 1
15067 GLAPI int GLAD_GL_OES_single_precision;
15068 typedef void (APIENTRY PFNGLCLEARDEPTHFOESPROC) (GLclampf depth);
15069 GLAPI PFNGLCLEARDEPTHFOESPROC glad_glClearDepthfOES;
15070 #define glClearDepthfOES glad_glClearDepthfOES
15071 typedef void (APIENTRY PFNGLCLIPPLANEFOESPROC) (GLenum plane, const GLfloat *equation);
15072 GLAPI PFNGLCLIPPLANEFOESPROC glad_glClipPlanefOES;
15073 #define glClipPlanefOES glad_glClipPlanefOES
15074 typedef void (APIENTRY PFNGLDEPTHRANGEFOESPROC) (GLclampf n, GLclampf f);
15075 GLAPI PFNGLDEPTHRANGEFOESPROC glad_glDepthRangefOES;
15076 #define glDepthRangefOES glad_glDepthRangefOES
15077 typedef void (APIENTRY PFNGLFRUSTUMFOESPROC) (GLfloat l, GLfloat r, GLfloat b, GLfloat t, GLfloat n,
    GLfloat f);
15078 GLAPI PFNGLFRUSTUMFOESPROC glad_glFrustumfOES;
15079 #define glFrustumfOES glad_glFrustumfOES
15080 typedef void (APIENTRY PFNGLGETCLIPPLANEFOESPROC) (GLenum plane, GLfloat *equation);
15081 GLAPI PFNGLGETCLIPPLANEFOESPROC glad_glGetClipPlanefOES;
15082 #define glGetClipPlanefOES glad_glGetClipPlanefOES
15083 typedef void (APIENTRY PFNGLORTHOFOESPROC) (GLfloat l, GLfloat r, GLfloat b, GLfloat t, GLfloat n,
    GLfloat f);
15084 GLAPI PFNGLORTHOFOESPROC glad_glOrthofOES;
15085 #define glOrthofOES glad_glOrthofOES
15086 #endif
15087 #ifndef GL_OML_interlace
15088 #define GL_OML_interlace 1
15089 GLAPI int GLAD_GL_OML_interlace;
15090 #endif
15091 #ifndef GL_OML_resample
15092 #define GL_OML_resample 1
15093 GLAPI int GLAD_GL_OML_resample;
15094 #endif
15095 #ifndef GL_OML_subsample
15096 #define GL_OML_subsample 1
15097 GLAPI int GLAD_GL_OML_subsample;
15098 #endif
15099 #ifndef GL_OVR_multiview
15100 #define GL_OVR_multiview 1
15101 GLAPI int GLAD_GL_OVR_multiview;
15102 typedef void (APIENTRY PFNGLFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC) (GLenum target, GLenum attachment,
    GLuint texture, GLint level, GLint baseViewIndex, GLsizei numViews);
15103 GLAPI PFNGLFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC glad_glFramebufferTextureMultiviewOVR;
15104 #define glFramebufferTextureMultiviewOVR glad_glFramebufferTextureMultiviewOVR
15105 typedef void (APIENTRY PFNGLNAMEDFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC) (GLuint framebuffer, GLenum
    attachment, GLuint texture, GLint level, GLint baseViewIndex, GLsizei numViews);
15106 GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC glad_glNamedFramebufferTextureMultiviewOVR;
15107 #define glNamedFramebufferTextureMultiviewOVR glad_glNamedFramebufferTextureMultiviewOVR
15108 #endif
15109 #ifndef GL_OVR_multiview2
15110 #define GL_OVR_multiview2 1
15111 GLAPI int GLAD_GL_OVR_multiview2;
15112 #endif
15113 #ifndef GL_PGI_misc_hints

```



```

15114 #define GL_PGI_misc_hints 1
15115 GLAPI int GLAD_GL_PGI_misc_hints;
15116 typedef void (APIENTRY PFNGLHINTPGIPROC) (GLenum target, GLint mode);
15117 GLAPI PFNGLHINTPGIPROC glad_glHintPGI;
15118 #define glHintPGI glad_glHintPGI
15119 #endif
15120 #ifndef GL_PGI_vertex_hints
15121 #define GL_PGI_vertex_hints 1
15122 GLAPI int GLAD_GL_PGI_vertex_hints;
15123 #endif
15124 #ifndef GL_REND_screen_coordinates
15125 #define GL_REND_screen_coordinates 1
15126 GLAPI int GLAD_GL_REND_screen_coordinates;
15127 #endif
15128 #ifndef GL_S3_s3tc
15129 #define GL_S3_s3tc 1
15130 GLAPI int GLAD_GL_S3_s3tc;
15131 #endif
15132 #ifndef GL_SGIS_detail_texture
15133 #define GL_SGIS_detail_texture 1
15134 GLAPI int GLAD_GL_SGIS_detail_texture;
15135 typedef void (APIENTRY PFNGLDETAILTEXFUNCSPROC) (GLenum target, GLsizei n, const GLfloat *points);
15136 GLAPI PFNGLDETAILTEXFUNCSPROC glad_glDetailTexFuncSGIS;
15137 #define glDetailTexFuncSGIS glad_glDetailTexFuncSGIS
15138 typedef void (APIENTRY PFNGLGETDETAILTEXFUNCSPROC) (GLenum target, GLfloat *points);
15139 GLAPI PFNGLGETDETAILTEXFUNCSPROC glad_glGetDetailTexFuncSGIS;
15140 #define glGetDetailTexFuncSGIS glad_glGetDetailTexFuncSGIS
15141 #endif
15142 #ifndef GL_SGIS_fog_function
15143 #define GL_SGIS_fog_function 1
15144 GLAPI int GLAD_GL_SGIS_fog_function;
15145 typedef void (APIENTRY PFNGLFOGFUNCSPROC) (GLsizei n, const GLfloat *points);
15146 GLAPI PFNGLFOGFUNCSPROC glad_glFogFuncSGIS;
15147 #define glFogFuncSGIS glad_glFogFuncSGIS
15148 typedef void (APIENTRY PFNGLGETFOGFUNCSPROC) (GLfloat *points);
15149 GLAPI PFNGLGETFOGFUNCSPROC glad_glGetFogFuncSGIS;
15150 #define glGetFogFuncSGIS glad_glGetFogFuncSGIS
15151 #endif
15152 #ifndef GL_SGIS_generate_mipmap
15153 #define GL_SGIS_generate_mipmap 1
15154 GLAPI int GLAD_GL_SGIS_generate_mipmap;
15155 #endif
15156 #ifndef GL_SGIS_multisample
15157 #define GL_SGIS_multisample 1
15158 GLAPI int GLAD_GL_SGIS_multisample;
15159 typedef void (APIENTRY PFNGLSAMPLEMASKSGIPROC) (GLclampf value, GLboolean invert);
15160 GLAPI PFNGLSAMPLEMASKSGIPROC glad_glSampleMaskSGIS;
15161 #define glSampleMaskSGIS glad_glSampleMaskSGIS
15162 typedef void (APIENTRY PFNGLSAMPLEPATTERNSGIPROC) (GLenum pattern);
15163 GLAPI PFNGLSAMPLEPATTERNSGIPROC glad_glSamplePatternSGIS;
15164 #define glSamplePatternSGIS glad_glSamplePatternSGIS
15165 #endif
15166 #ifndef GL_SGIS_pixel_texture
15167 #define GL_SGIS_pixel_texture 1
15168 GLAPI int GLAD_GL_SGIS_pixel_texture;
15169 typedef void (APIENTRY PFNGLPIXELTEXGENPARAMETERISGIPROC) (GLenum pname, GLint param);
15170 GLAPI PFNGLPIXELTEXGENPARAMETERISGIPROC glad_glPixelTexGenParameteriSGIS;
15171 #define glPixelTexGenParameteriSGIS glad_glPixelTexGenParameteriSGIS
15172 typedef void (APIENTRY PFNGLPIXELTEXGENPARAMETERIVSGIPROC) (GLenum pname, const GLint *params);
15173 GLAPI PFNGLPIXELTEXGENPARAMETERIVSGIPROC glad_glPixelTexGenParameterivSGIS;
15174 #define glPixelTexGenParameterivSGIS glad_glPixelTexGenParameterivSGIS
15175 typedef void (APIENTRY PFNGLPIXELTEXGENPARAMETERFVSGIPROC) (GLenum pname, GLfloat param);
15176 GLAPI PFNGLPIXELTEXGENPARAMETERFVSGIPROC glad_glPixelTexGenParameterfvSGIS;
15177 #define glPixelTexGenParameterfvSGIS glad_glPixelTexGenParameterfvSGIS
15178 typedef void (APIENTRY PFNGLPIXELTEXGENPARAMETERFVSGIPROC) (GLenum pname, const GLfloat *params);
15179 GLAPI PFNGLPIXELTEXGENPARAMETERFVSGIPROC glad_glPixelTexGenParameterfvSGIS;
15180 #define glGetPixelTexGenParameterfvSGIS glad_glGetPixelTexGenParameterfvSGIS
15181 #endif
15182 #ifndef GL_SGIS_point_line_texgen
15183 #define GL_SGIS_point_line_texgen 1
15184 GLAPI int GLAD_GL_SGIS_point_line_texgen;
15185 #endif
15186 #ifndef GL_SGIS_point_parameters
15187 #define GL_SGIS_point_parameters 1
15188 GLAPI int GLAD_GL_SGIS_point_parameters;
15189 typedef void (APIENTRY PFNGLPOINTPARAMETERFVSGIPROC) (GLenum pname, GLfloat param);
15190 GLAPI PFNGLPOINTPARAMETERFVSGIPROC glad_glPointParameterfvSGIS;
15191 #define glPointParameterfvSGIS glad_glPointParameterfvSGIS
15192 #endif
15193 #ifndef GL_SGIS_point_parameters
15194 #define GL_SGIS_point_parameters 1
15195 GLAPI int GLAD_GL_SGIS_point_parameters;
15196 typedef void (APIENTRY PFNGLPOINTPARAMETERFVSGIPROC) (GLenum pname, GLfloat param);
15197 GLAPI PFNGLPOINTPARAMETERFVSGIPROC glad_glPointParameterfvSGIS;
15198 #define glPointParameterfvSGIS glad_glPointParameterfvSGIS
15199 #endif
15200 #define glPointParameterfvSGIS glad_glPointParameterfvSGIS

```

```

15201 #endif
15202 #ifndef GL_SGIS_sharpen_texture
15203 #define GL_SGIS_sharpen_texture 1
15204 GLAPI int GLAD_GL_SGIS_sharpen_texture;
15205 typedef void (APIENTRY PFNGLSHARPENTEXFUNCSPROC) (GLenum target, GLsizei n, const GLfloat *points);
15206 GLAPI PFNGLSHARPENTEXFUNCSPROC glad_glSharpenTexFuncSGIS;
15207 #define glSharpenTexFuncSGIS glad_glSharpenTexFuncSGIS
15208 typedef void (APIENTRY PFNGLGETSHARPENTEXFUNCSPROC) (GLenum target, GLfloat *points);
15209 GLAPI PFNGLGETSHARPENTEXFUNCSPROC glad_glGetSharpenTexFuncSGIS;
15210 #define glGetSharpenTexFuncSGIS glad_glGetSharpenTexFuncSGIS
15211 #endif
15212 #ifndef GL_SGIS_texture4D
15213 #define GL_SGIS_texture4D 1
15214 GLAPI int GLAD_GL_SGIS_texture4D;
15215 typedef void (APIENTRY PFNGLTEXIMAGE4DPROC) (GLenum target, GLint level, GLenum internalformat,
GLsizei width, GLsizei height, GLsizei depth, GLsizei size4d, GLint border, GLenum format, GLenum
type, const void *pixels);
15216 GLAPI PFNGLTEXIMAGE4DPROC glad_glTexImage4DSGIS;
15217 #define glTexImage4DSGIS glad_glTexImage4DSGIS
15218 typedef void (APIENTRY PFNGLTEXSUBIMAGE4DPROC) (GLenum target, GLint level, GLint xoffset, GLint
yoffset, GLint zoffset, GLint woffset, GLsizei width, GLsizei height, GLsizei depth, GLsizei size4d,
GLenum format, GLenum type, const void *pixels);
15219 GLAPI PFNGLTEXSUBIMAGE4DPROC glad_glTexSubImage4DSGIS;
15220 #define glTexSubImage4DSGIS glad_glTexSubImage4DSGIS
15221 #endif
15222 #ifndef GL_SGIS_texture_border_clamp
15223 #define GL_SGIS_texture_border_clamp 1
15224 GLAPI int GLAD_GL_SGIS_texture_border_clamp;
15225 #endif
15226 #ifndef GL_SGIS_texture_color_mask
15227 #define GL_SGIS_texture_color_mask 1
15228 GLAPI int GLAD_GL_SGIS_texture_color_mask;
15229 typedef void (APIENTRY PFNGLTEXTURECOLORMASKSPROC) (GLboolean red, GLboolean green, GLboolean blue,
GLboolean alpha);
15230 GLAPI PFNGLTEXTURECOLORMASKSPROC glad_glTextureColorMaskSGIS;
15231 #define glTextureColorMaskSGIS glad_glTextureColorMaskSGIS
15232 #endif
15233 #ifndef GL_SGIS_texture_edge_clamp
15234 #define GL_SGIS_texture_edge_clamp 1
15235 GLAPI int GLAD_GL_SGIS_texture_edge_clamp;
15236 #endif
15237 #ifndef GL_SGIS_texture_filter4
15238 #define GL_SGIS_texture_filter4 1
15239 GLAPI int GLAD_GL_SGIS_texture_filter4;
15240 typedef void (APIENTRY PFNGLGETTEXFILTERFUNCSPROC) (GLenum target, GLenum filter, GLfloat
*weights);
15241 GLAPI PFNGLGETTEXFILTERFUNCSPROC glad_glGetTexFilterFuncSGIS;
15242 #define glGetTexFilterFuncSGIS glad_glGetTexFilterFuncSGIS
15243 typedef void (APIENTRY PFNGLTEXFILTERFUNCSPROC) (GLenum target, GLenum filter, GLsizei n, const
GLfloat *weights);
15244 GLAPI PFNGLTEXFILTERFUNCSPROC glad_glTexFilterFuncSGIS;
15245 #define glTexFilterFuncSGIS glad_glTexFilterFuncSGIS
15246 #endif
15247 #ifndef GL_SGIS_texture_lod
15248 #define GL_SGIS_texture_lod 1
15249 GLAPI int GLAD_GL_SGIS_texture_lod;
15250 #endif
15251 #ifndef GL_SGIS_texture_select
15252 #define GL_SGIS_texture_select 1
15253 GLAPI int GLAD_GL_SGIS_texture_select;
15254 #endif
15255 #ifndef GL_SGIX_async
15256 #define GL_SGIX_async 1
15257 GLAPI int GLAD_GL_SGIX_async;
15258 typedef void (APIENTRY PFNGLASYNCMARKERSGIXPROC) (GLuint marker);
15259 GLAPI PFNGLASYNCMARKERSGIXPROC glad_glAsyncMarkerSGIX;
15260 #define glAsyncMarkerSGIX glad_glAsyncMarkerSGIX
15261 typedef GLint (APIENTRY PFNGLFINISHASYNCSGIXPROC) (GLuint *markerp);
15262 GLAPI PFNGLFINISHASYNCSGIXPROC glad_glFinishAsyncSGIX;
15263 #define glFinishAsyncSGIX glad_glFinishAsyncSGIX
15264 typedef GLint (APIENTRY PFNGLPOLLASYNCSGIXPROC) (GLuint *markerp);
15265 GLAPI PFNGLPOLLASYNCSGIXPROC glad_glPollAsyncSGIX;
15266 #define glPollAsyncSGIX glad_glPollAsyncSGIX
15267 typedef GLuint (APIENTRY PFNGLGENASYNCMARKERSGIXPROC) (GLsizei range);
15268 GLAPI PFNGLGENASYNCMARKERSGIXPROC glad_glGenAsyncMarkersSGIX;
15269 #define glGenAsyncMarkersSGIX glad_glGenAsyncMarkersSGIX
15270 typedef void (APIENTRY PFNGLDELETEASYNCMARKERSGIXPROC) (GLuint marker, GLsizei range);
15271 GLAPI PFNGLDELETEASYNCMARKERSGIXPROC glad_glDeleteAsyncMarkersSGIX;
15272 #define glDeleteAsyncMarkersSGIX glad_glDeleteAsyncMarkersSGIX
15273 typedef GLboolean (APIENTRY PFNGLISASYNCMARKERSGIXPROC) (GLuint marker);
15274 GLAPI PFNGLISASYNCMARKERSGIXPROC glad_glIsAsyncMarkerSGIX;
15275 #define glIsAsyncMarkerSGIX glad_glIsAsyncMarkerSGIX
15276 #endif
15277 #ifndef GL_SGIX_async_histogram
15278 #define GL_SGIX_async_histogram 1
15279 GLAPI int GLAD_GL_SGIX_async_histogram;
15280 #endif

```

```

15281 #ifndef GL_SGIX_async_pixel
15282 #define GL_SGIX_async_pixel 1
15283 GLAPI int GLAD_GL_SGIX_async_pixel;
15284 #endif
15285 #ifndef GL_SGIX_blend_alpha_minmax
15286 #define GL_SGIX_blend_alpha_minmax 1
15287 GLAPI int GLAD_GL_SGIX_blend_alpha_minmax;
15288 #endif
15289 #ifndef GL_SGIX_calligraphic_fragment
15290 #define GL_SGIX_calligraphic_fragment 1
15291 GLAPI int GLAD_GL_SGIX_calligraphic_fragment;
15292 #endif
15293 #ifndef GL_SGIX_clipmap
15294 #define GL_SGIX_clipmap 1
15295 GLAPI int GLAD_GL_SGIX_clipmap;
15296 #endif
15297 #ifndef GL_SGIX_convolution_accuracy
15298 #define GL_SGIX_convolution_accuracy 1
15299 GLAPI int GLAD_GL_SGIX_convolution_accuracy;
15300 #endif
15301 #ifndef GL_SGIX_depth_pass_instrument
15302 #define GL_SGIX_depth_pass_instrument 1
15303 GLAPI int GLAD_GL_SGIX_depth_pass_instrument;
15304 #endif
15305 #ifndef GL_SGIX_depth_texture
15306 #define GL_SGIX_depth_texture 1
15307 GLAPI int GLAD_GL_SGIX_depth_texture;
15308 #endif
15309 #ifndef GL_SGIX_flush_raster
15310 #define GL_SGIX_flush_raster 1
15311 GLAPI int GLAD_GL_SGIX_flush_raster;
15312 typedef void (APIENTRY PFNGLFLUSHRASTERSGIXPROC)(void);
15313 GLAPI PFNGLFLUSHRASTERSGIXPROC glad_glFlushRasterSGIX;
15314 #define glFlushRasterSGIX glad_glFlushRasterSGIX
15315 #endif
15316 #ifndef GL_SGIX_fog_offset
15317 #define GL_SGIX_fog_offset 1
15318 GLAPI int GLAD_GL_SGIX_fog_offset;
15319 #endif
15320 #ifndef GL_SGIX_fragment_lighting
15321 #define GL_SGIX_fragment_lighting 1
15322 GLAPI int GLAD_GL_SGIX_fragment_lighting;
15323 typedef void (APIENTRY PFNGLFRAGMENTCOLORMATERIALSGIXPROC)(GLenum face, GLenum mode);
15324 GLAPI PFNGLFRAGMENTCOLORMATERIALSGIXPROC glad_glFragmentColorMaterialSGIX;
15325 #define glFragmentColorMaterialSGIX glad_glFragmentColorMaterialSGIX
15326 typedef void (APIENTRY PFNGLFRAGMENTLIGHTFSGIXPROC)(GLenum light, GLenum pname, GLfloat param);
15327 GLAPI PFNGLFRAGMENTLIGHTFSGIXPROC glad_glFragmentLightfSGIX;
15328 #define glFragmentLightfSGIX glad_glFragmentLightfSGIX
15329 typedef void (APIENTRY PFNGLFRAGMENTLIGHTFVSGIXPROC)(GLenum light, GLenum pname, const GLfloat
    *params);
15330 GLAPI PFNGLFRAGMENTLIGHTFVSGIXPROC glad_glFragmentLightfvSGIX;
15331 #define glFragmentLightfvSGIX glad_glFragmentLightfvSGIX
15332 typedef void (APIENTRY PFNGLFRAGMENTLIGHTISGIXPROC)(GLenum light, GLenum pname, GLint param);
15333 GLAPI PFNGLFRAGMENTLIGHTISGIXPROC glad_glFragmentLightiSGIX;
15334 #define glFragmentLightiSGIX glad_glFragmentLightiSGIX
15335 typedef void (APIENTRY PFNGLFRAGMENTLIGHTIVSGIXPROC)(GLenum light, GLenum pname, const GLint
    *params);
15336 GLAPI PFNGLFRAGMENTLIGHTIVSGIXPROC glad_glFragmentLightivSGIX;
15337 #define glFragmentLightivSGIX glad_glFragmentLightivSGIX
15338 typedef void (APIENTRY PFNGLFRAGMENTLIGHTMODELFSGIXPROC)(GLenum pname, GLfloat param);
15339 GLAPI PFNGLFRAGMENTLIGHTMODELFSGIXPROC glad_glFragmentLightModelfSGIX;
15340 #define glFragmentLightModelfSGIX glad_glFragmentLightModelfSGIX
15341 typedef void (APIENTRY PFNGLFRAGMENTLIGHTMODELFSVSGIXPROC)(GLenum pname, const GLfloat *params);
15342 GLAPI PFNGLFRAGMENTLIGHTMODELFSVSGIXPROC glad_glFragmentLightModelfvSGIX;
15343 #define glFragmentLightModelfvSGIX glad_glFragmentLightModelfvSGIX
15344 typedef void (APIENTRY PFNGLFRAGMENTLIGHTMODELISGIXPROC)(GLenum pname, GLint param);
15345 GLAPI PFNGLFRAGMENTLIGHTMODELISGIXPROC glad_glFragmentLightModeliSGIX;
15346 #define glFragmentLightModeliSGIX glad_glFragmentLightModeliSGIX
15347 typedef void (APIENTRY PFNGLFRAGMENTLIGHTMODELIVSGIXPROC)(GLenum pname, const GLint *params);
15348 GLAPI PFNGLFRAGMENTLIGHTMODELIVSGIXPROC glad_glFragmentLightModelivSGIX;
15349 #define glFragmentLightModelivSGIX glad_glFragmentLightModelivSGIX
15350 typedef void (APIENTRY PFNGLFRAGMENTMATERIALFSGIXPROC)(GLenum face, GLenum pname, GLfloat param);
15351 GLAPI PFNGLFRAGMENTMATERIALFSGIXPROC glad_glFragmentMaterialfSGIX;
15352 #define glFragmentMaterialfSGIX glad_glFragmentMaterialfSGIX
15353 typedef void (APIENTRY PFNGLFRAGMENTMATERIALFVSGIXPROC)(GLenum face, GLenum pname, const GLfloat
    *params);
15354 GLAPI PFNGLFRAGMENTMATERIALFVSGIXPROC glad_glFragmentMaterialfvSGIX;
15355 #define glFragmentMaterialfvSGIX glad_glFragmentMaterialfvSGIX
15356 typedef void (APIENTRY PFNGLFRAGMENTMATERIALISGIXPROC)(GLenum face, GLenum pname, GLint param);
15357 GLAPI PFNGLFRAGMENTMATERIALISGIXPROC glad_glFragmentMaterialiSGIX;
15358 #define glFragmentMaterialiSGIX glad_glFragmentMaterialiSGIX
15359 typedef void (APIENTRY PFNGLFRAGMENTMATERIALIVSGIXPROC)(GLenum face, GLenum pname, const GLint
    *params);
15360 GLAPI PFNGLFRAGMENTMATERIALIVSGIXPROC glad_glFragmentMaterialivSGIX;
15361 #define glFragmentMaterialivSGIX glad_glFragmentMaterialivSGIX
15362 typedef void (APIENTRY PFNGLGETFRAGMENTLIGHTFVSGIXPROC)(GLenum light, GLenum pname, GLfloat *params);
15363 GLAPI PFNGLGETFRAGMENTLIGHTFVSGIXPROC glad_glGetFragmentLightfvSGIX;

```

```

15364 #define glGetFragmentLightfvSGIX glad_glGetFragmentLightfvSGIX
15365 typedef void (APIENTRY PFNGLGETFRAGMENTLIGHTIVSGIXPROC) (GLenum light, GLenum pname, GLint *params);
15366 GLAPI PFNGLGETFRAGMENTLIGHTIVSGIXPROC glad_glGetFragmentLightivSGIX;
15367 #define glGetFragmentLightivSGIX glad_glGetFragmentLightivSGIX
15368 typedef void (APIENTRY PFNGLGETFRAGMENTMATERIALFVSGIXPROC) (GLenum face, GLenum pname, GLfloat
    *params);
15369 GLAPI PFNGLGETFRAGMENTMATERIALFVSGIXPROC glad_glGetFragmentMaterialfvSGIX;
15370 #define glGetFragmentMaterialfvSGIX glad_glGetFragmentMaterialfvSGIX
15371 typedef void (APIENTRY PFNGLGETFRAGMENTMATERIALIVSGIXPROC) (GLenum face, GLenum pname, GLint *params);
15372 GLAPI PFNGLGETFRAGMENTMATERIALIVSGIXPROC glad_glGetFragmentMaterialivSGIX;
15373 #define glGetFragmentMaterialivSGIX glad_glGetFragmentMaterialivSGIX
15374 typedef void (APIENTRY PFNGLLIGHTENVISGXPROC) (GLenum pname, GLint param);
15375 GLAPI PFNGLLIGHTENVISGXPROC glad_glLightEnvivSGIX;
15376 #define glLightEnvivSGIX glad_glLightEnvivSGIX
15377 #endif
15378 #ifndef GL_SGIX_framezoom
15379 #define GL_SGIX_framezoom 1
15380 GLAPI int GLAD_GL_SGIX_framezoom;
15381 typedef void (APIENTRY PFNGLFRAMEZOOMSGIXPROC) (GLint factor);
15382 GLAPI PFNGLFRAMEZOOMSGIXPROC glad_glFrameZoomSGIX;
15383 #define glFrameZoomSGIX glad_glFrameZoomSGIX
15384 #endif
15385 #ifndef GL_SGIX_igloo_interface
15386 #define GL_SGIX_igloo_interface 1
15387 GLAPI int GLAD_GL_SGIX_igloo_interface;
15388 typedef void (APIENTRY PFNGLIGLOOINTERFACESGXPROC) (GLenum pname, const void *params);
15389 GLAPI PFNGLIGLOOINTERFACESGXPROC glad_glIgllooInterfaceSGIX;
15390 #define glIgllooInterfaceSGIX glad_glIgllooInterfaceSGIX
15391 #endif
15392 #ifndef GL_SGIX_instruments
15393 #define GL_SGIX_instruments 1
15394 GLAPI int GLAD_GL_SGIX_instruments;
15395 typedef GLint (APIENTRY PFNGLGETINSTRUMENTSSGXPROC) (void);
15396 GLAPI PFNGLGETINSTRUMENTSSGXPROC glad_glGetInstrumentsSGIX;
15397 #define glGetInstrumentsSGIX glad_glGetInstrumentsSGIX
15398 typedef void (APIENTRY PFNGLINSTRUMENTSBUFFERSGIXPROC) (GLsizei size, GLint *buffer);
15399 GLAPI PFNGLINSTRUMENTSBUFFERSGIXPROC glad_glInstrumentsBufferSGIX;
15400 #define glInstrumentsBufferSGIX glad_glInstrumentsBufferSGIX
15401 typedef GLint (APIENTRY PFNGLPOLLINSTRUMENTSSGXPROC) (GLint *marker_p);
15402 GLAPI PFNGLPOLLINSTRUMENTSSGXPROC glad_glPollInstrumentsSGIX;
15403 #define glPollInstrumentsSGIX glad_glPollInstrumentsSGIX
15404 typedef void (APIENTRY PFNGLREADINSTRUMENTSSGXPROC) (GLint marker);
15405 GLAPI PFNGLREADINSTRUMENTSSGXPROC glad_glReadInstrumentsSGIX;
15406 #define glReadInstrumentsSGIX glad_glReadInstrumentsSGIX
15407 typedef void (APIENTRY PFNGLSTARTINSTRUMENTSSGXPROC) (void);
15408 GLAPI PFNGLSTARTINSTRUMENTSSGXPROC glad_glStartInstrumentsSGIX;
15409 #define glStartInstrumentsSGIX glad_glStartInstrumentsSGIX
15410 typedef void (APIENTRY PFNGLSTOPINSTRUMENTSSGXPROC) (GLint marker);
15411 GLAPI PFNGLSTOPINSTRUMENTSSGXPROC glad_glStopInstrumentsSGIX;
15412 #define glStopInstrumentsSGIX glad_glStopInstrumentsSGIX
15413 #endif
15414 #ifndef GL_SGIX_interlace
15415 #define GL_SGIX_interlace 1
15416 GLAPI int GLAD_GL_SGIX_interlace;
15417 #endif
15418 #ifndef GL_SGIX_ir_instrument1
15419 #define GL_SGIX_ir_instrument1 1
15420 GLAPI int GLAD_GL_SGIX_ir_instrument1;
15421 #endif
15422 #ifndef GL_SGIX_list_priority
15423 #define GL_SGIX_list_priority 1
15424 GLAPI int GLAD_GL_SGIX_list_priority;
15425 typedef void (APIENTRY PFNGLGETLISTPARAMETERFVSGIXPROC) (GLuint list, GLenum pname, GLfloat *params);
15426 GLAPI PFNGLGETLISTPARAMETERFVSGIXPROC glad_glGetListParameterfvSGIX;
15427 #define glGetListParameterfvSGIX glad_glGetListParameterfvSGIX
15428 typedef void (APIENTRY PFNGLGETLISTPARAMETERIVSGIXPROC) (GLuint list, GLenum pname, GLint *params);
15429 GLAPI PFNGLGETLISTPARAMETERIVSGIXPROC glad_glGetListParameterivSGIX;
15430 #define glGetListParameterivSGIX glad_glGetListParameterivSGIX
15431 typedef void (APIENTRY PFNGLLISTPARAMETERFSGIXPROC) (GLuint list, GLenum pname, GLfloat param);
15432 GLAPI PFNGLLISTPARAMETERFSGIXPROC glad_glListParameterfSGIX;
15433 #define glListParameterfSGIX glad_glListParameterfSGIX
15434 typedef void (APIENTRY PFNGLLISTPARAMETERFVSGIXPROC) (GLuint list, GLenum pname, const GLfloat
    *params);
15435 GLAPI PFNGLLISTPARAMETERFVSGIXPROC glad_glListParameterfvSGIX;
15436 #define glListParameterfvSGIX glad_glListParameterfvSGIX
15437 typedef void (APIENTRY PFNGLLISTPARAMETERISGXPROC) (GLuint list, GLenum pname, GLint param);
15438 GLAPI PFNGLLISTPARAMETERISGXPROC glad_glListParameterisGX;
15439 #define glListParameterisGX glad_glListParameterisGX
15440 typedef void (APIENTRY PFNGLLISTPARAMETERIVSGIXPROC) (GLuint list, GLenum pname, const GLint *params);
15441 GLAPI PFNGLLISTPARAMETERIVSGIXPROC glad_glListParameterivSGIX;
15442 #define glListParameterivSGIX glad_glListParameterivSGIX
15443 #endif
15444 #ifndef GL_SGIX_pixel_texture
15445 #define GL_SGIX_pixel_texture 1
15446 GLAPI int GLAD_GL_SGIX_pixel_texture;
15447 typedef void (APIENTRY PFNGLPIXELTEXGENSGIXPROC) (GLenum mode);
15448 GLAPI PFNGLPIXELTEXGENSGIXPROC glad_glPixelTexGenSGIX;

```

```

15449 #define glPixelTexGenSGIX glad_glPixelTexGenSGIX
15450 #endif
15451 #ifndef GL_SGIX_pixel_tiles
15452 #define GL_SGIX_pixel_tiles 1
15453 GLAPI int GLAD_GL_SGIX_pixel_tiles;
15454 #endif
15455 #ifndef GL_SGIX_polynomial_ffd
15456 #define GL_SGIX_polynomial_ffd 1
15457 GLAPI int GLAD_GL_SGIX_polynomial_ffd;
15458 typedef void (APIENTRY PFNGLDEFORMATIONMAP3DSGIXPROC) (GLenum target, GLdouble u1, GLdouble u2, GLint
    ustride, GLint uorder, GLdouble v1, GLdouble v2, GLint vstride, GLint vorder, GLdouble w1, GLdouble
    w2, GLint wstride, GLint worder, const GLdouble *points);
15459 GLAPI PFNGLDEFORMATIONMAP3DSGIXPROC glad_glDeformationMap3dSGIX;
15460 #define glDeformationMap3dSGIX glad_glDeformationMap3dSGIX
15461 typedef void (APIENTRY PFNGLDEFORMATIONMAP3FSGIXPROC) (GLenum target, GLfloat u1, GLfloat u2, GLint
    ustride, GLint uorder, GLfloat v1, GLfloat v2, GLint vstride, GLint vorder, GLfloat w1, GLfloat w2,
    GLint wstride, GLint worder, const GLfloat *points);
15462 GLAPI PFNGLDEFORMATIONMAP3FSGIXPROC glad_glDeformationMap3fSGIX;
15463 #define glDeformationMap3fSGIX glad_glDeformationMap3fSGIX
15464 typedef void (APIENTRY PFNGLDEFORMSGIXPROC) (GLbitfield mask);
15465 GLAPI PFNGLDEFORMSGIXPROC glad_glDeformSGIX;
15466 #define glDeformSGIX glad_glDeformSGIX
15467 typedef void (APIENTRY PFNGLLOADIDENTITYDEFORMATIONMAPSGIXPROC) (GLbitfield mask);
15468 GLAPI PFNGLLOADIDENTITYDEFORMATIONMAPSGIXPROC glad_glLoadIdentityDeformationMapSGIX;
15469 #define glLoadIdentityDeformationMapSGIX glad_glLoadIdentityDeformationMapSGIX
15470 #endif
15471 #ifndef GL_SGIX_reference_plane
15472 #define GL_SGIX_reference_plane 1
15473 GLAPI int GLAD_GL_SGIX_reference_plane;
15474 typedef void (APIENTRY PFNGLREFERENCEPLANESGIXPROC) (const GLdouble *equation);
15475 GLAPI PFNGLREFERENCEPLANESGIXPROC glad_glReferencePlaneSGIX;
15476 #define glReferencePlaneSGIX glad_glReferencePlaneSGIX
15477 #endif
15478 #ifndef GL_SGIX_resample
15479 #define GL_SGIX_resample 1
15480 GLAPI int GLAD_GL_SGIX_resample;
15481 #endif
15482 #ifndef GL_SGIX_scalebias_hint
15483 #define GL_SGIX_scalebias_hint 1
15484 GLAPI int GLAD_GL_SGIX_scalebias_hint;
15485 #endif
15486 #ifndef GL_SGIX_shadow
15487 #define GL_SGIX_shadow 1
15488 GLAPI int GLAD_GL_SGIX_shadow;
15489 #endif
15490 #ifndef GL_SGIX_shadow_ambient
15491 #define GL_SGIX_shadow_ambient 1
15492 GLAPI int GLAD_GL_SGIX_shadow_ambient;
15493 #endif
15494 #ifndef GL_SGIX_sprite
15495 #define GL_SGIX_sprite 1
15496 GLAPI int GLAD_GL_SGIX_sprite;
15497 typedef void (APIENTRY PFNGLSPRITEPARAMETERFSGIXPROC) (GLenum pname, GLfloat param);
15498 GLAPI PFNGLSPRITEPARAMETERFSGIXPROC glad_glSpriteParameterfSGIX;
15499 #define glSpriteParameterfSGIX glad_glSpriteParameterfSGIX
15500 typedef void (APIENTRY PFNGLSPRITEPARAMETERFVSGIXPROC) (GLenum pname, const GLfloat *params);
15501 GLAPI PFNGLSPRITEPARAMETERFVSGIXPROC glad_glSpriteParameterfvSGIX;
15502 #define glSpriteParameterfvSGIX glad_glSpriteParameterfvSGIX
15503 typedef void (APIENTRY PFNGLSPRITEPARAMETERISGIXPROC) (GLenum pname, GLint param);
15504 GLAPI PFNGLSPRITEPARAMETERISGIXPROC glad_glSpriteParameteriSGIX;
15505 #define glSpriteParameteriSGIX glad_glSpriteParameteriSGIX
15506 typedef void (APIENTRY PFNGLSPRITEPARAMETERIVSGIXPROC) (GLenum pname, const GLint *params);
15507 GLAPI PFNGLSPRITEPARAMETERIVSGIXPROC glad_glSpriteParameterivSGIX;
15508 #define glSpriteParameterivSGIX glad_glSpriteParameterivSGIX
15509 #endif
15510 #ifndef GL_SGIX_subsample
15511 #define GL_SGIX_subsample 1
15512 GLAPI int GLAD_GL_SGIX_subsample;
15513 #endif
15514 #ifndef GL_SGIX_tag_sample_buffer
15515 #define GL_SGIX_tag_sample_buffer 1
15516 GLAPI int GLAD_GL_SGIX_tag_sample_buffer;
15517 typedef void (APIENTRY PFNGLTAGSAMPLEBUFFERSGIXPROC) (void);
15518 GLAPI PFNGLTAGSAMPLEBUFFERSGIXPROC glad_glTagSampleBufferSGIX;
15519 #define glTagSampleBufferSGIX glad_glTagSampleBufferSGIX
15520 #endif
15521 #ifndef GL_SGIX_texture_add_env
15522 #define GL_SGIX_texture_add_env 1
15523 GLAPI int GLAD_GL_SGIX_texture_add_env;
15524 #endif
15525 #ifndef GL_SGIX_texture_coordinate_clamp
15526 #define GL_SGIX_texture_coordinate_clamp 1
15527 GLAPI int GLAD_GL_SGIX_texture_coordinate_clamp;
15528 #endif
15529 #ifndef GL_SGIX_texture_lod_bias
15530 #define GL_SGIX_texture_lod_bias 1
15531 GLAPI int GLAD_GL_SGIX_texture_lod_bias;

```

```

15532 #endif
15533 #ifndef GL_SGIX_texture_multi_buffer
15534 #define GL_SGIX_texture_multi_buffer 1
15535 GLAPI int GLAD_GL_SGIX_texture_multi_buffer;
15536 #endif
15537 #ifndef GL_SGIX_texture_scale_bias
15538 #define GL_SGIX_texture_scale_bias 1
15539 GLAPI int GLAD_GL_SGIX_texture_scale_bias;
15540 #endif
15541 #ifndef GL_SGIX_vertex_preclip
15542 #define GL_SGIX_vertex_preclip 1
15543 GLAPI int GLAD_GL_SGIX_vertex_preclip;
15544 #endif
15545 #ifndef GL_SGIX_ycrCb
15546 #define GL_SGIX_ycrCb 1
15547 GLAPI int GLAD_GL_SGIX_ycrCb;
15548 #endif
15549 #ifndef GL_SGIX_ycrCb_subsample
15550 #define GL_SGIX_ycrCb_subsample 1
15551 GLAPI int GLAD_GL_SGIX_ycrCb_subsample;
15552 #endif
15553 #ifndef GL_SGIX_ycrcb
15554 #define GL_SGIX_ycrcb 1
15555 GLAPI int GLAD_GL_SGIX_ycrcb;
15556 #endif
15557 #ifndef GL_SGI_color_matrix
15558 #define GL_SGI_color_matrix 1
15559 GLAPI int GLAD_GL_SGI_color_matrix;
15560 #endif
15561 #ifndef GL_SGI_color_table
15562 #define GL_SGI_color_table 1
15563 GLAPI int GLAD_GL_SGI_color_table;
15564 typedef void (APIENTRY PFNGLCOLORTABLESGIPROC) (GLenum target, GLenum internalformat, GLsizei width,
15565   GLenum format, GLenum type, const void *table);
15566 GLAPI PFNGLCOLORTABLESGIPROC glad_glColorTableSGI;
15567 typedef void (APIENTRY PFNGLCOLORTABLEPARAMETERFVSGIPROC) (GLenum target, GLenum pname, const GLfloat
15568   *params);
15569 GLAPI PFNGLCOLORTABLEPARAMETERFVSGIPROC glad_glColorTableParameterfvSGI;
15570 #define glColorTableParameterfvSGI glad_glColorTableParameterfvSGI
15571 typedef void (APIENTRY PFNGLCOLORTABLEPARAMETERIVSGIPROC) (GLenum target, GLenum pname, const GLint
15572   *params);
15573 GLAPI PFNGLCOLORTABLEPARAMETERIVSGIPROC glad_glColorTableParameterivSGI;
15574 #define glColorTableParameterivSGI glad_glColorTableParameterivSGI
15575 typedef void (APIENTRY PFNGLCOPYCOLORTABLESGIPROC) (GLenum target, GLenum internalformat, GLint x,
15576   GLint y, GLsizei width);
15577 GLAPI PFNGLCOPYCOLORTABLESGIPROC glad_glCopyColorTableSGI;
15578 #define glCopyColorTableSGI glad_glCopyColorTableSGI
15579 typedef void (APIENTRY PFNGLGETCOLORTABLESGIPROC) (GLenum target, GLenum format, GLenum type, void
15580   *table);
15581 GLAPI PFNGLGETCOLORTABLESGIPROC glad_glGetColorTableSGI;
15582 #define glGetColorTableSGI glad_glGetColorTableSGI
15583 typedef void (APIENTRY PFNGLGETCOLORTABLEPARAMETERFVSGIPROC) (GLenum target, GLenum pname, GLfloat
15584   *params);
15585 GLAPI PFNGLGETCOLORTABLEPARAMETERFVSGIPROC glad_glGetColorTableParameterfvSGI;
15586 #define glGetColorTableParameterfvSGI glad_glGetColorTableParameterfvSGI
15587 typedef void (APIENTRY PFNGLGETCOLORTABLEPARAMETERIVSGIPROC) (GLenum target, GLenum pname, GLint
15588   *params);
15589 GLAPI PFNGLGETCOLORTABLEPARAMETERIVSGIPROC glad_glGetColorTableParameterivSGI;
15590 #define glGetColorTableParameterivSGI glad_glGetColorTableParameterivSGI
15591 #endif
15592 #ifndef GL_SGI_texture_color_table
15593 #define GL_SGI_texture_color_table 1
15594 GLAPI int GLAD_GL_SGI_texture_color_table;
15595 #endif
15596 #ifndef GL_SUNX_constant_data
15597 #define GL_SUNX_constant_data 1
15598 GLAPI int GLAD_GL_SUNX_constant_data;
15599 typedef void (APIENTRY PFNGLFINISHTEXTURESUNXPROC) (void);
15600 GLAPI PFNGLFINISHTEXTURESUNXPROC glad_glFinishTextureSUNX;
15601 #define glFinishTextureSUNX glad_glFinishTextureSUNX
15602 #endif
15603 #ifndef GL_SUN_convolution_border_modes
15604 #define GL_SUN_convolution_border_modes 1
15605 GLAPI int GLAD_GL_SUN_convolution_border_modes;
15606 #endif
15607 #ifndef GL_SUN_global_alpha
15608 #define GL_SUN_global_alpha 1
15609 GLAPI int GLAD_GL_SUN_global_alpha;
15610 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORBSUNPROC) (GLbyte factor);
15611 GLAPI PFNGLGLOBALALPHAFACTORBSUNPROC glad_glGlobalAlphaFactorbSUN;
15612 #define glGlobalAlphaFactorbSUN glad_glGlobalAlphaFactorbSUN
15613 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORSSUNPROC) (GLshort factor);
15614 GLAPI PFNGLGLOBALALPHAFACTORSSUNPROC glad_glGlobalAlphaFactorsSUN;
15615 #define glGlobalAlphaFactorsSUN glad_glGlobalAlphaFactorsSUN
15616 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORISUNPROC) (GLint factor);
15617 GLAPI PFNGLGLOBALALPHAFACTORISUNPROC glad_glGlobalAlphaFactoriSUN;

```



```

15612 #define glGlobalAlphaFactoriSUN glad_glGlobalAlphaFactoriSUN
15613 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORFSUNPROC) (GLfloat factor);
15614 GLAPI PFNGLGLOBALALPHAFACTORFSUNPROC glad_glGlobalAlphaFactorfSUN;
15615 #define glGlobalAlphaFactorfSUN glad_glGlobalAlphaFactorfSUN
15616 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORDSUNPROC) (GLdouble factor);
15617 GLAPI PFNGLGLOBALALPHAFACTORDSUNPROC glad_glGlobalAlphaFactordSUN;
15618 #define glGlobalAlphaFactordSUN glad_glGlobalAlphaFactordSUN
15619 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORUBSUNPROC) (GLubyte factor);
15620 GLAPI PFNGLGLOBALALPHAFACTORUBSUNPROC glad_glGlobalAlphaFactorubSUN;
15621 #define glGlobalAlphaFactorubSUN glad_glGlobalAlphaFactorubSUN
15622 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORUSSUNPROC) (GLushort factor);
15623 GLAPI PFNGLGLOBALALPHAFACTORUSSUNPROC glad_glGlobalAlphaFactorusSUN;
15624 #define glGlobalAlphaFactorusSUN glad_glGlobalAlphaFactorusSUN
15625 typedef void (APIENTRY PFNGLGLOBALALPHAFACTORUISUNPROC) (GLuint factor);
15626 GLAPI PFNGLGLOBALALPHAFACTORUISUNPROC glad_glGlobalAlphaFactoruiSUN;
15627 #define glGlobalAlphaFactoruiSUN glad_glGlobalAlphaFactoruiSUN
15628 #endif
15629 #ifndef GL_SUN_mesh_array
15630 #define GL_SUN_mesh_array 1
15631 GLAPI int GLAD_GL_SUN_mesh_array;
15632 typedef void (APIENTRY PFNGLDRAWMESHHARRAYSSUNPROC) (GLenum mode, GLint first, GLsizei count, GLsizei
width);
15633 GLAPI PFNGLDRAWMESHHARRAYSSUNPROC glad_glDrawMeshArraysSUN;
15634 #define glDrawMeshArraysSUN glad_glDrawMeshArraysSUN
15635 #endif
15636 #ifndef GL_SUN_slice_accum
15637 #define GL_SUN_slice_accum 1
15638 GLAPI int GLAD_GL_SUN_slice_accum;
15639 #endif
15640 #ifndef GL_SUN_triangle_list
15641 #define GL_SUN_triangle_list 1
15642 GLAPI int GLAD_GL_SUN_triangle_list;
15643 typedef void (APIENTRY PFNGLREPLACEMENTCODEUISUNPROC) (GLuint code);
15644 GLAPI PFNGLREPLACEMENTCODEUISUNPROC glad_glReplacementCodeuiSUN;
15645 #define glReplacementCodeuiSUN glad_glReplacementCodeuiSUN
15646 typedef void (APIENTRY PFNGLREPLACEMENTCODEUSSUNPROC) (GLushort code);
15647 GLAPI PFNGLREPLACEMENTCODEUSSUNPROC glad_glReplacementCodeusSUN;
15648 #define glReplacementCodeusSUN glad_glReplacementCodeusSUN
15649 typedef void (APIENTRY PFNGLREPLACEMENTCODEUBSUNPROC) (GLubyte code);
15650 GLAPI PFNGLREPLACEMENTCODEUBSUNPROC glad_glReplacementCodeubSUN;
15651 #define glReplacementCodeubSUN glad_glReplacementCodeubSUN
15652 typedef void (APIENTRY PFNGLREPLACEMENTCODEUIVSUNPROC) (const GLuint *code);
15653 GLAPI PFNGLREPLACEMENTCODEUIVSUNPROC glad_glReplacementCodeuivSUN;
15654 #define glReplacementCodeuivSUN glad_glReplacementCodeuivSUN
15655 typedef void (APIENTRY PFNGLREPLACEMENTCODEUSVSUNPROC) (const GLushort *code);
15656 GLAPI PFNGLREPLACEMENTCODEUSVSUNPROC glad_glReplacementCodeusvSUN;
15657 #define glReplacementCodeusvSUN glad_glReplacementCodeusvSUN
15658 typedef void (APIENTRY PFNGLREPLACEMENTCODEUBVSUNPROC) (const GLubyte *code);
15659 GLAPI PFNGLREPLACEMENTCODEUBVSUNPROC glad_glReplacementCodeubvSUN;
15660 #define glReplacementCodeubvSUN glad_glReplacementCodeubvSUN
15661 typedef void (APIENTRY PFNGLREPLACEMENTCODEPOINTERSUNPROC) (GLenum type, GLsizei stride, const void
**pointer);
15662 GLAPI PFNGLREPLACEMENTCODEPOINTERSUNPROC glad_glReplacementCodePointerSUN;
15663 #define glReplacementCodePointerSUN glad_glReplacementCodePointerSUN
15664 #endif
15665 #ifndef GL_SUN_vertex
15666 #define GL_SUN_vertex 1
15667 GLAPI int GLAD_GL_SUN_vertex;
15668 typedef void (APIENTRY PFNGLCOLOR4UBVERTEX2FSUNPROC) (GLubyte r, GLubyte g, GLubyte b, GLubyte a,
GLfloat x, GLfloat y);
15669 GLAPI PFNGLCOLOR4UBVERTEX2FSUNPROC glad_glColor4ubVertex2fSUN;
15670 #define glColor4ubVertex2fSUN glad_glColor4ubVertex2fSUN
15671 typedef void (APIENTRY PFNGLCOLOR4UBVERTEX2FVSUNPROC) (const GLubyte *c, const GLfloat *v);
15672 GLAPI PFNGLCOLOR4UBVERTEX2FVSUNPROC glad_glColor4ubVertex2fvSUN;
15673 #define glColor4ubVertex2fvSUN glad_glColor4ubVertex2fvSUN
15674 typedef void (APIENTRY PFNGLCOLOR4UBVERTEX3FSUNPROC) (GLubyte r, GLubyte g, GLubyte b, GLubyte a,
GLfloat x, GLfloat y, GLfloat z);
15675 GLAPI PFNGLCOLOR4UBVERTEX3FSUNPROC glad_glColor4ubVertex3fSUN;
15676 #define glColor4ubVertex3fSUN glad_glColor4ubVertex3fSUN
15677 typedef void (APIENTRY PFNGLCOLOR4UBVERTEX3FVSUNPROC) (const GLubyte *c, const GLfloat *v);
15678 GLAPI PFNGLCOLOR4UBVERTEX3FVSUNPROC glad_glColor4ubVertex3fvSUN;
15679 #define glColor4ubVertex3fvSUN glad_glColor4ubVertex3fvSUN
15680 typedef void (APIENTRY PFNGLCOLOR3FVERTEX3FSUNPROC) (GLfloat r, GLfloat g, GLfloat b, GLfloat x,
GLfloat y, GLfloat z);
15681 GLAPI PFNGLCOLOR3FVERTEX3FSUNPROC glad_glColor3fVertex3fSUN;
15682 #define glColor3fVertex3fSUN glad_glColor3fVertex3fSUN
15683 typedef void (APIENTRY PFNGLCOLOR3FVERTEX3FVSUNPROC) (const GLfloat *c, const GLfloat *v);
15684 GLAPI PFNGLCOLOR3FVERTEX3FVSUNPROC glad_glColor3fVertex3fvSUN;
15685 #define glColor3fVertex3fvSUN glad_glColor3fVertex3fvSUN
15686 typedef void (APIENTRY PFNGLNORMAL3FVERTEX3FSUNPROC) (GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x,
GLfloat y, GLfloat z);
15687 GLAPI PFNGLNORMAL3FVERTEX3FSUNPROC glad_glNormal3fVertex3fSUN;
15688 #define glNormal3fVertex3fSUN glad_glNormal3fVertex3fSUN
15689 typedef void (APIENTRY PFNGLNORMAL3FVERTEX3FVSUNPROC) (const GLfloat *n, const GLfloat *v);
15690 GLAPI PFNGLNORMAL3FVERTEX3FVSUNPROC glad_glNormal3fVertex3fvSUN;
15691 #define glNormal3fVertex3fvSUN glad_glNormal3fVertex3fvSUN
15692 typedef void (APIENTRY PFNGLCOLOR4FNORMAL3FVERTEX3FSUNPROC) (GLfloat r, GLfloat g, GLfloat b, GLfloat

```

```
a, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
15693 GLAPI PFNGLCOLOR4FNORMAL3FVERTEX3FSUNPROC glad_glColor4fNormal3fVertex3fSUN;
15694 #define glColor4fNormal3fVertex3fSUN glad_glColor4fNormal3fVertex3fSUN
15695 typedef void (APIENTRY PFNGLCOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLfloat *c, const GLfloat *n,
const GLfloat *v);
15696 GLAPI PFNGLCOLOR4FNORMAL3FVERTEX3FVSUNPROC glad_glColor4fNormal3fVertex3fvSUN;
15697 #define glColor4fNormal3fVertex3fvSUN glad_glColor4fNormal3fVertex3fvSUN
15698 typedef void (APIENTRY PFNGLTEXCOORD2FVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLfloat x, GLfloat y,
GLfloat z);
15699 GLAPI PFNGLTEXCOORD2FVERTEX3FSUNPROC glad_glTexCoord2fVertex3fSUN;
15700 #define glTexCoord2fVertex3fSUN glad_glTexCoord2fVertex3fSUN
15701 typedef void (APIENTRY PFNGLTEXCOORD2FVERTEX3FVSUNPROC)(const GLfloat *tc, const GLfloat *v);
15702 GLAPI PFNGLTEXCOORD2FVERTEX3FVSUNPROC glad_glTexCoord2fVertex3fvSUN;
15703 #define glTexCoord2fVertex3fvSUN glad_glTexCoord2fVertex3fvSUN
15704 typedef void (APIENTRY PFNGLTEXCOORD4FVERTEX4FSUNPROC)(GLfloat s, GLfloat t, GLfloat p, GLfloat q,
GLfloat x, GLfloat y, GLfloat z, GLfloat w);
15705 GLAPI PFNGLTEXCOORD4FVERTEX4FSUNPROC glad_glTexCoord4fVertex4fSUN;
15706 #define glTexCoord4fVertex4fSUN glad_glTexCoord4fVertex4fSUN
15707 typedef void (APIENTRY PFNGLTEXCOORD4FVERTEX4FVSUNPROC)(const GLfloat *tc, const GLfloat *v);
15708 GLAPI PFNGLTEXCOORD4FVERTEX4FVSUNPROC glad_glTexCoord4fVertex4fvSUN;
15709 #define glTexCoord4fVertex4fvSUN glad_glTexCoord4fVertex4fvSUN
15710 typedef void (APIENTRY PFNGLTEXCOORD2FCOLOR4UBVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLubyte r,
GLubyte g, GLubyte b, GLubyte a, GLfloat x, GLfloat y, GLfloat z);
15711 GLAPI PFNGLTEXCOORD2FCOLOR4UBVERTEX3FSUNPROC glad_glTexCoord2fColor4ubVertex3fSUN;
15712 #define glTexCoord2fColor4ubVertex3fSUN glad_glTexCoord2fColor4ubVertex3fSUN
15713 typedef void (APIENTRY PFNGLTEXCOORD2FCOLOR4UBVERTEX3FVSUNPROC)(const GLfloat *tc, const GLubyte *c,
const GLfloat *v);
15714 GLAPI PFNGLTEXCOORD2FCOLOR4UBVERTEX3FVSUNPROC glad_glTexCoord2fColor4ubVertex3fvSUN;
15715 #define glTexCoord2fColor4ubVertex3fvSUN glad_glTexCoord2fColor4ubVertex3fvSUN
15716 typedef void (APIENTRY PFNGLTEXCOORD2FCOLOR3FVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLfloat r,
GLfloat g, GLfloat b, GLfloat x, GLfloat y, GLfloat z);
15717 GLAPI PFNGLTEXCOORD2FCOLOR3FVERTEX3FSUNPROC glad_glTexCoord2fColor3fVertex3fSUN;
15718 #define glTexCoord2fColor3fVertex3fSUN glad_glTexCoord2fColor3fVertex3fSUN
15719 typedef void (APIENTRY PFNGLTEXCOORD2FCOLOR3FVERTEX3FVSUNPROC)(const GLfloat *tc, const GLfloat *c,
const GLfloat *v);
15720 GLAPI PFNGLTEXCOORD2FCOLOR3FVERTEX3FVSUNPROC glad_glTexCoord2fColor3fVertex3fvSUN;
15721 #define glTexCoord2fColor3fVertex3fvSUN glad_glTexCoord2fColor3fVertex3fvSUN
15722 typedef void (APIENTRY PFNGLTEXCOORD2FNORMAL3FVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLfloat nx,
GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
15723 GLAPI PFNGLTEXCOORD2FNORMAL3FVERTEX3FSUNPROC glad_glTexCoord2fNormal3fVertex3fSUN;
15724 #define glTexCoord2fNormal3fVertex3fSUN glad_glTexCoord2fNormal3fVertex3fSUN
15725 typedef void (APIENTRY PFNGLTEXCOORD2FNORMAL3FVERTEX3FVSUNPROC)(const GLfloat *tc, const GLfloat *n,
const GLfloat *v);
15726 GLAPI PFNGLTEXCOORD2FNORMAL3FVERTEX3FVSUNPROC glad_glTexCoord2fNormal3fVertex3fvSUN;
15727 #define glTexCoord2fNormal3fVertex3fvSUN glad_glTexCoord2fNormal3fVertex3fvSUN
15728 typedef void (APIENTRY PFNGLTEXCOORD2FCOLOR4FNORMAL3FVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLfloat
r, GLfloat g, GLfloat b, GLfloat a, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat
z);
15729 GLAPI PFNGLTEXCOORD2FCOLOR4FNORMAL3FVERTEX3FSUNPROC glad_glTexCoord2fColor4fNormal3fVertex3fSUN;
15730 #define glTexCoord2fColor4fNormal3fVertex3fSUN glad_glTexCoord2fColor4fNormal3fVertex3fSUN
15731 typedef void (APIENTRY PFNGLTEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLfloat *tc, const
GLfloat *c, const GLfloat *n, const GLfloat *v);
15732 GLAPI PFNGLTEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC glad_glTexCoord2fColor4fNormal3fVertex3fvSUN;
15733 #define glTexCoord2fColor4fNormal3fVertex3fvSUN glad_glTexCoord2fColor4fNormal3fVertex3fvSUN
15734 typedef void (APIENTRY PFNGLTEXCOORD4FCOLOR4FNORMAL3FVERTEX4FSUNPROC)(GLfloat s, GLfloat t, GLfloat
p, GLfloat q, GLfloat r, GLfloat g, GLfloat b, GLfloat a, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat
x, GLfloat y, GLfloat z, GLfloat w);
15735 GLAPI PFNGLTEXCOORD4FCOLOR4FNORMAL3FVERTEX4FSUNPROC glad_glTexCoord4fColor4fNormal3fVertex4fSUN;
15736 #define glTexCoord4fColor4fNormal3fVertex4fSUN glad_glTexCoord4fColor4fNormal3fVertex4fSUN
15737 typedef void (APIENTRY PFNGLTEXCOORD4FCOLOR4FNORMAL3FVERTEX4FVSUNPROC)(const GLfloat *tc, const
GLfloat *c, const GLfloat *n, const GLfloat *v);
15738 GLAPI PFNGLTEXCOORD4FCOLOR4FNORMAL3FVERTEX4FVSUNPROC glad_glTexCoord4fColor4fNormal3fVertex4fvSUN;
15739 #define glTexCoord4fColor4fNormal3fVertex4fvSUN glad_glTexCoord4fColor4fNormal3fVertex4fvSUN
15740 typedef void (APIENTRY PFNGLREPLACEMENTCODEUIVERTEX3FSUNPROC)(GLuint rc, GLfloat x, GLfloat y,
GLfloat z);
15741 GLAPI PFNGLREPLACEMENTCODEUIVERTEX3FSUNPROC glad_glReplacementCodeuiVertex3fSUN;
15742 #define glReplacementCodeuiVertex3fSUN glad_glReplacementCodeuiVertex3fSUN
15743 typedef void (APIENTRY PFNGLREPLACEMENTCODEUIVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *v);
15744 GLAPI PFNGLREPLACEMENTCODEUIVERTEX3FVSUNPROC glad_glReplacementCodeuiVertex3fvSUN;
15745 #define glReplacementCodeuiVertex3fvSUN glad_glReplacementCodeuiVertex3fvSUN
15746 typedef void (APIENTRY PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FSUNPROC)(GLuint rc, GLubyte r, GLubyte
g, GLubyte b, GLubyte a, GLfloat x, GLfloat y, GLfloat z);
15747 GLAPI PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FSUNPROC glad_glReplacementCodeuiColor4ubVertex3fSUN;
15748 #define glReplacementCodeuiColor4ubVertex3fSUN glad_glReplacementCodeuiColor4ubVertex3fSUN
15749 typedef void (APIENTRY PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FVSUNPROC)(const GLuint *rc, const
GLubyte *c, const GLfloat *v);
15750 GLAPI PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FVSUNPROC glad_glReplacementCodeuiColor4ubVertex3fvSUN;
15751 #define glReplacementCodeuiColor4ubVertex3fvSUN glad_glReplacementCodeuiColor4ubVertex3fvSUN
15752 typedef void (APIENTRY PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FSUNPROC)(GLuint rc, GLfloat r, GLfloat g,
GLfloat b, GLfloat x, GLfloat y, GLfloat z);
15753 GLAPI PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FSUNPROC glad_glReplacementCodeuiColor3fVertex3fSUN;
15754 #define glReplacementCodeuiColor3fVertex3fSUN glad_glReplacementCodeuiColor3fVertex3fSUN
15755 typedef void (APIENTRY PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat
*c, const GLfloat *v);
15756 GLAPI PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FVSUNPROC glad_glReplacementCodeuiColor3fVertex3fvSUN;
15757 #define glReplacementCodeuiColor3fVertex3fvSUN glad_glReplacementCodeuiColor3fVertex3fvSUN
15758 typedef void (APIENTRY PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FSUNPROC)(GLuint rc, GLfloat nx, GLfloat
```

```

ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
15759 GLAPI PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FSUNPROC glad_glReplacementCodeuiNormal3fVertex3fSUN;
15760 #define glReplacementCodeuiNormal3fVertex3fSUN glad_glReplacementCodeuiNormal3fVertex3fSUN
15761 typedef void (APIENTRY PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FSUNPROC)(const GLuint *rc, const
    GLfloat *n, const GLfloat *v);
15762 GLAPI PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FVSUNPROC glad_glReplacementCodeuiNormal3fVertex3fvSUN;
15763 #define glReplacementCodeuiNormal3fVertex3fvSUN glad_glReplacementCodeuiNormal3fVertex3fvSUN
15764 typedef void (APIENTRY PFNGLREPLACEMENTCODEUCOLOR4FNORMAL3FVERTEX3FSUNPROC)(GLuint rc, GLfloat r,
    GLfloat g, GLfloat b, GLfloat a, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
15765 GLAPI PFNGLREPLACEMENTCODEUCOLOR4FNORMAL3FVERTEX3FSUNPROC
    glad_glReplacementCodeuiColor4fNormal3fVertex3fSUN;
15766 #define glReplacementCodeuiColor4fNormal3fVertex3fSUN
    glad_glReplacementCodeuiColor4fNormal3fVertex3fSUN
15767 typedef void (APIENTRY PFNGLREPLACEMENTCODEUCOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLuint *rc, const
    GLfloat *c, const GLfloat *n, const GLfloat *v);
15768 GLAPI PFNGLREPLACEMENTCODEUCOLOR4FNORMAL3FVERTEX3FVSUNPROC
    glad_glReplacementCodeuiColor4fNormal3fVertex3fvSUN;
15769 #define glReplacementCodeuiColor4fNormal3fVertex3fvSUN
    glad_glReplacementCodeuiColor4fNormal3fVertex3fvSUN
15770 typedef void (APIENTRY PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FSUNPROC)(GLuint rc, GLfloat s, GLfloat
    t, GLfloat x, GLfloat y, GLfloat z);
15771 GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FSUNPROC glad_glReplacementCodeuiTexCoord2fVertex3fSUN;
15772 #define glReplacementCodeuiTexCoord2fVertex3fSUN glad_glReplacementCodeuiTexCoord2fVertex3fSUN
15773 typedef void (APIENTRY PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FVSUNPROC)(const GLuint *rc, const
    GLfloat *tc, const GLfloat *v);
15774 GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FVSUNPROC glad_glReplacementCodeuiTexCoord2fVertex3fvSUN;
15775 #define glReplacementCodeuiTexCoord2fVertex3fvSUN glad_glReplacementCodeuiTexCoord2fVertex3fvSUN
15776 typedef void (APIENTRY PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FSUNPROC)(GLuint rc, GLfloat s,
    GLfloat t, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
15777 GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FSUNPROC
    glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fSUN;
15778 #define glReplacementCodeuiTexCoord2fNormal3fVertex3fSUN
    glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fSUN
15779 typedef void (APIENTRY PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FVSUNPROC)(const GLuint *rc,
    const GLfloat *tc, const GLfloat *n, const GLfloat *v);
15780 GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FVSUNPROC
    glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fvSUN;
15781 #define glReplacementCodeuiTexCoord2fNormal3fVertex3fvSUN
    glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fvSUN
15782 typedef void (APIENTRY PFNGLREPLACEMENTCODEUITEXCOORD2FCOLOR4FNORMAL3FVERTEX3FSUNPROC)(GLuint rc,
    GLfloat s, GLfloat t, GLfloat r, GLfloat g, GLfloat b, GLfloat a, GLfloat nx, GLfloat ny, GLfloat nz,
    GLfloat x, GLfloat y, GLfloat z);
15783 GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FCOLOR4FNORMAL3FVERTEX3FSUNPROC
    glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fSUN;
15784 #define glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fSUN
    glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fSUN
15785 typedef void (APIENTRY PFNGLREPLACEMENTCODEUITEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLuint
    *rc, const GLfloat *tc, const GLfloat *c, const GLfloat *n, const GLfloat *v);
15786 GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC
    glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fvSUN;
15787 #define glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fvSUN
    glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fvSUN
15788 #endif
15789 #ifndef GL_WIN_phong_shading
15790 #define GL_WIN_phong_shading 1
15791 GLAPI int GLAD_GL_WIN_phong_shading;
15792 #endif
15793 #ifndef GL_WIN_specular_fog
15794 #define GL_WIN_specular_fog 1
15795 GLAPI int GLAD_GL_WIN_specular_fog;
15796 #endif
15797
15798 #ifdef __cplusplus
15799 }
15800 #endif
15801
15802 #endif

```

7.5 GraphicsController.hpp

```

00001 #ifndef GRAPHICSCONTROLLER_HPP
00002 #define GRAPHICSCONTROLLER_HPP
00003
00004 #include <Error.hpp>
00005 #include <SDL2/SDL.h>
00006 #include <SDL2/SDL_ttf.h>
00007 #include <ShaderProgram.hpp>
00008 #include <functional>
00009 #include <glad/glad.h>
00010 #include <memory>
00011 #include <optional>
00012 #include <string>
00013 #include <unordered_map>

```

```

00014
00015 class GraphicsController {
00016     public:
00017         static void debugMessage(GLenum source, GLenum type, GLuint debug_id,
00018                                 GLenum severity, GLsizei length,
00019                                 const GLchar *message, const void *userParam);
00020         static void enableDebug();
00021         static std::optional<Error> initGL(SDL_Window *sdlWindow);
00022         static std::optional<Error> CheckGLObjectStatus(
00023             GLuint object, GLenum parameter,
00024             const std::function<void(GLuint, GLenum, GLint *)> &getiv,
00025             const std::function<void(GLuint, GLsizei, GLsizei *, GLchar *)>
00026             &getInfoLog);
00027         static std::unordered_map<std::string, std::unique_ptr<ShaderProgram>
00028             s_shaderPrograms;
00029         static std::unordered_map<
00030             std::string, std::unique_ptr<TTF_Font, std::function<void(TTF_Font *)>>
00031             s_fonts;
00032     private:
00033         static SDL_GLContext s_glContext;
00034 };
00035 #endif
00036

```

7.6 IntroState.hpp

```

00001 #ifndef INTROSTATE_HPP
00002 #define INTROSTATE_HPP
00003
00004 #include "App.hpp"
00005 #include "IState.hpp"
00006 #include <TexturedRect2D.hpp>
00007 #include <memory>
00008
00009 class IntroState : public IState {
00010     App &app;
00011
00012     public:
00013         IntroState(App &app);
00014         ~IntroState() override;
00015         void Run() override;
00016         void SendEvent(SDL_Event &event) override;
00017
00018     private:
00019         struct Slide {
00020             SDL_Surface *slideSurface;
00021             float startX;
00022             float stopX;
00023             float startY;
00024             float stopY;
00025             float panSpeedMS;
00026         };
00027
00028         std::shared_ptr<TexturedRect2D> m_slideshow;
00029         Slide m_slides[3];
00030         int m_currSlide;
00031
00032         void NextSlide();
00033
00034         static constexpr int SLIDE_COUNT = 3;
00035         static constexpr float SLIDE_RIGHT = -1.4 / 2;
00036         static constexpr float SLIDE_LEFT = -1.0 / 2;
00037         static constexpr float SLIDE_TOP = -1.4 / 2;
00038         static constexpr float SLIDE_BOTTOM = -1.0 / 2;
00039         static constexpr float SLIDE_MIDDLE = -1.2 / 2;
00040         static constexpr float SLIDE_ZOOM = 1.2;
00041         static constexpr float SECOND_MS = 1000.0;
00042     };
00043 #endif
00044

```

7.7 IRender.hpp

```

00001 #ifndef IRENDER_HPP
00002 #define IRENDER_HPP
00003
00004 #include <SDL2/SDL.h>
00005 #include <glad/glad.h>
00006 #include <string>

```

```

00007
00008 class IRender {
00009     protected:
00010         GLuint m_vertexArrayObject;
00011         GLuint m_vertexBufferObject;
00012         GLuint m_indexBufferObject;
00013         std::string m_shaderProgram;
00014         virtual void GenBindBufferGL() = 0;
00015
00016     public:
00017         virtual void Render() = 0;
00018         virtual void UpdateGL() = 0;
00019 };
00020
00021 #endif

```

7.8 IState.hpp

```

00001 #ifndef ISTATE_HPP
00002 #define ISTATE_HPP
00003
00004 #include <SDL2/SDL.h>
00005
00006 class IState {
00007     public:
00008         virtual void Run() = 0;
00009         virtual void SendEvent(SDL_Event &event) = 0;
00010         virtual ~IState() = default;
00011 };
00012
00013 #endif

```

7.9 khrplatform.h

```

00001 #ifndef __khrplatform_h_
00002 #define __khrplatform_h_
00003
00004 /*
00005 ** Copyright (c) 2008-2018 The Khronos Group Inc.
00006 **
00007 ** Permission is hereby granted, free of charge, to any person obtaining a
00008 ** copy of this software and/or associated documentation files (the
00009 ** "Materials"), to deal in the Materials without restriction, including
00010 ** without limitation the rights to use, copy, modify, merge, publish,
00011 ** distribute, sublicense, and/or sell copies of the Materials, and to
00012 ** permit persons to whom the Materials are furnished to do so, subject to
00013 ** the following conditions:
00014 **
00015 ** The above copyright notice and this permission notice shall be included
00016 ** in all copies or substantial portions of the Materials.
00017 **
00018 ** THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
00019 ** EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
00020 ** MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
00021 ** IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
00022 ** CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
00023 ** TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
00024 ** MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.
00025 */
00026
00027 /* Khronos platform-specific types and definitions.
00028 *
00029 * The master copy of khrplatform.h is maintained in the Khronos EGL
00030 * Registry repository at https://github.com/KhronosGroup/EGL-Registry
00031 * The last semantic modification to khrplatform.h was at commit ID:
00032 * 67a3e0864c2d75ea5287b9f3d2eb74a745936692
00033 *
00034 * Adopters may modify this file to suit their platform. Adopters are
00035 * encouraged to submit platform specific modifications to the Khronos
00036 * group so that they can be included in future versions of this file.
00037 * Please submit changes by filing pull requests or issues on
00038 * the EGL Registry repository linked above.
00039 *
00040 *
00041 * See the Implementer's Guidelines for information about where this file
00042 * should be located on your system and for more details of its use:
00043 * http://www.khronos.org/registry/implementers\_guide.pdf
00044 *
00045 * This file should be included as

```

```

00046 *      #include <KHR/khrplatform.h>
00047 * by Khronos client API header files that use its types and defines.
00048 *
00049 * The types in khrplatform.h should only be used to define API-specific types.
00050 *
00051 * Types defined in khrplatform.h:
00052 *   khronos_int8_t          signed   8  bit
00053 *   khronos_uint8_t         unsigned 8  bit
00054 *   khronos_int16_t         signed  16 bit
00055 *   khronos_uint16_t        unsigned 16 bit
00056 *   khronos_int32_t         signed  32 bit
00057 *   khronos_uint32_t        unsigned 32 bit
00058 *   khronos_int64_t         signed  64 bit
00059 *   khronos_uint64_t        unsigned 64 bit
00060 *   khronos_intptr_t        signed   same number of bits as a pointer
00061 *   khronos_uintptr_t       unsigned same number of bits as a pointer
00062 *   khronos_ssize_t         signed   size
00063 *   khronos_usize_t         unsigned size
00064 *   khronos_float_t         signed   32 bit floating point
00065 *   khronos_time_ns_t       unsigned 64 bit time in nanoseconds
00066 *   khronos_etime_nanoseconds_t unsigned time interval or absolute time in
00067 *                               nanoseconds
00068 *   khronos_stime_nanoseconds_t signed time interval in nanoseconds
00069 *   khronos_boolean_enum_t   enumerated boolean type. This should
00070 *   only be used as a base type when a client API's boolean type is
00071 *   an enum. Client APIs which use an integer or other type for
00072 *   booleans cannot use this as the base type for their boolean.
00073 *
00074 * Tokens defined in khrplatform.h:
00075 *
00076 *   KHRONOS_FALSE, KHRONOS_TRUE Enumerated boolean false/true values.
00077 *
00078 *   KHRONOS_SUPPORT_INT64 is 1 if 64 bit integers are supported; otherwise 0.
00079 *   KHRONOS_SUPPORT_FLOAT is 1 if floats are supported; otherwise 0.
00080 *
00081 * Calling convention macros defined in this file:
00082 *   KHRONOS_APICALL
00083 *   KHRONOS_APIENTRY
00084 *   KHRONOS_APIATTRIBUTES
00085 *
00086 * These may be used in function prototypes as:
00087 *
00088 *   KHRONOS_APICALL void KHRONOS_APIENTRY funcname(
00089 *       int arg1,
00090 *       int arg2) KHRONOS_APIATTRIBUTES;
00091 */
00092
00093 #if defined(__SCITECH_SNAP__) && !defined(KHRONOS_STATIC)
00094 #   define KHRONOS_STATIC 1
00095 #endif
00096
00097 /*-----
00098 * Definition of KHRONOS_APICALL
00099 *-----
00100 * This precedes the return type of the function in the function prototype.
00101 */
00102 #if defined(KHRONOS_STATIC)
00103     /* If the preprocessor constant KHRONOS_STATIC is defined, make the
00104      * header compatible with static linking. */
00105     #define KHRONOS_APICALL
00106 #elif defined(_WIN32)
00107     #define KHRONOS_APICALL __declspec(dllimport)
00108 #elif defined(__SYMBIAN32__)
00109     #define KHRONOS_APICALL IMPORT_C
00110 #elif defined(__ANDROID__)
00111     #define KHRONOS_APICALL __attribute__((visibility("default")))
00112 #else
00113     #define KHRONOS_APICALL
00114 #endif
00115
00116 /*-----
00117 * Definition of KHRONOS_APIENTRY
00118 *-----
00119 * This follows the return type of the function and precedes the function
00120 * name in the function prototype.
00121 */
00122 #if defined(_WIN32) && !defined(_WIN32_WCE) && !defined(__SCITECH_SNAP__)
00123     /* Win32 but not WinCE */
00124     #define KHRONOS_APIENTRY __stdcall
00125 #else
00126     #define KHRONOS_APIENTRY
00127 #endif
00128
00129 /*-----
00130 * Definition of KHRONOS_APIATTRIBUTES
00131 *-----
00132 * This follows the closing parenthesis of the function prototype arguments.

```



```

00133  */
00134  #if defined (__ARMCC_2__)
00135  #define KHRONOS_APIATTRIBUTES __softfp
00136  #else
00137  #define KHRONOS_APIATTRIBUTES
00138  #endif
00139
00140  /*-----
00141  * basic type definitions
00142  *-----*/
00143  #if (defined(__STDC_VERSION__) && __STDC_VERSION__ >= 199901L) || defined(__GNUC__) ||
    defined(__SCO__) || defined(__USLC__)
00144
00145
00146  /*
00147  * Using <stdint.h>
00148  */
00149  #include <stdint.h>
00150  typedef int32_t          khronos_int32_t;
00151  typedef uint32_t         khronos_uint32_t;
00152  typedef int64_t          khronos_int64_t;
00153  typedef uint64_t         khronos_uint64_t;
00154  #define KHRONOS_SUPPORT_INT64 1
00155  #define KHRONOS_SUPPORT_FLOAT 1
00156  /*
00157  * To support platform where unsigned long cannot be used interchangeably with
00158  * intptr_t (e.g. CHERI-extended ISAs), we can use the stdint.h intptr_t.
00159  * Ideally, we could just use (u)intptr_t everywhere, but this could result in
00160  * ABI breakage if khronos_uintptr_t is changed from unsigned long to
00161  * unsigned long long or similar (this results in different C++ name mangling).
00162  * To avoid changes for existing platforms, we restrict usage of intptr_t to
00163  * platforms where the size of a pointer is larger than the size of long.
00164  */
00165  #if defined(__SIZEOF_LONG__) && defined(__SIZEOF_POINTER__)
00166  #if __SIZEOF_POINTER__ > __SIZEOF_LONG__
00167  #define KHRONOS_USE_INTPTR_T
00168  #endif
00169  #endif
00170
00171  #elif defined(__VMS ) || defined(__sgi)
00172
00173  /*
00174  * Using <inttypes.h>
00175  */
00176  #include <inttypes.h>
00177  typedef int32_t          khronos_int32_t;
00178  typedef uint32_t         khronos_uint32_t;
00179  typedef int64_t          khronos_int64_t;
00180  typedef uint64_t         khronos_uint64_t;
00181  #define KHRONOS_SUPPORT_INT64 1
00182  #define KHRONOS_SUPPORT_FLOAT 1
00183
00184  #elif defined(_WIN32) && !defined(__SCITECH_SNAP__)
00185
00186  /*
00187  * Win32
00188  */
00189  typedef __int32          khronos_int32_t;
00190  typedef unsigned __int32 khronos_uint32_t;
00191  typedef __int64          khronos_int64_t;
00192  typedef unsigned __int64 khronos_uint64_t;
00193  #define KHRONOS_SUPPORT_INT64 1
00194  #define KHRONOS_SUPPORT_FLOAT 1
00195
00196  #elif defined(__sun__) || defined(__digital__)
00197
00198  /*
00199  * Sun or Digital
00200  */
00201  typedef int              khronos_int32_t;
00202  typedef unsigned int     khronos_uint32_t;
00203  #if defined(__arch64__) || defined(__LP64)
00204  typedef long int         khronos_int64_t;
00205  typedef unsigned long int khronos_uint64_t;
00206  #else
00207  typedef long long int    khronos_int64_t;
00208  typedef unsigned long long int khronos_uint64_t;
00209  #endif /* __arch64__ */
00210  #define KHRONOS_SUPPORT_INT64 1
00211  #define KHRONOS_SUPPORT_FLOAT 1
00212
00213  #elif 0
00214
00215  /*
00216  * Hypothetical platform with no float or int64 support
00217  */
00218  typedef int              khronos_int32_t;

```

```

00219 typedef unsigned int          khronos_uint32_t;
00220 #define KHRONOS_SUPPORT_INT64  0
00221 #define KHRONOS_SUPPORT_FLOAT  0
00222
00223 #else
00224
00225 /*
00226  * Generic fallback
00227  */
00228 #include <stdint.h>
00229 typedef int32_t          khronos_int32_t;
00230 typedef uint32_t         khronos_uint32_t;
00231 typedef int64_t          khronos_int64_t;
00232 typedef uint64_t         khronos_uint64_t;
00233 #define KHRONOS_SUPPORT_INT64  1
00234 #define KHRONOS_SUPPORT_FLOAT  1
00235
00236 #endif
00237
00238 /*
00239  * Types that are (so far) the same on all platforms
00240  */
00241
00242 typedef signed   char          khronos_int8_t;
00243 typedef unsigned char          khronos_uint8_t;
00244 typedef signed   short int     khronos_int16_t;
00245 typedef unsigned short int     khronos_uint16_t;
00246
00247 /*
00248  * Types that differ between LLP64 and LP64 architectures - in LLP64,
00249  * pointers are 64 bits, but 'long' is still 32 bits. Win64 appears
00250  * to be the only LLP64 architecture in current use.
00251  */
00252 #ifndef KHRONOS_USE_INTPTR_T
00253 typedef intptr_t          khronos_intptr_t;
00254 typedef uintptr_t         khronos_uintptr_t;
00255 #elif defined(_WIN64)
00256 typedef signed   long long int khronos_intptr_t;
00257 typedef unsigned long long int khronos_uintptr_t;
00258 #else
00259 typedef signed   long int     khronos_intptr_t;
00260 typedef unsigned long int     khronos_uintptr_t;
00261 #endif
00262
00263 #if defined(_WIN64)
00264 typedef signed   long long int khronos_ssize_t;
00265 typedef unsigned long long int khronos_usize_t;
00266 #else
00267 typedef signed   long int     khronos_ssize_t;
00268 typedef unsigned long int     khronos_usize_t;
00269 #endif
00270
00271 #if KHRONOS_SUPPORT_FLOAT
00272 /*
00273  * Float type
00274  */
00275 typedef float          khronos_float_t;
00276 #endif
00277
00278 #if KHRONOS_SUPPORT_INT64
00279 /* Time types
00280  *
00281  * These types can be used to represent a time interval in nanoseconds or
00282  * an absolute Unadjusted System Time. Unadjusted System Time is the number
00283  * of nanoseconds since some arbitrary system event (e.g. since the last
00284  * time the system booted). The Unadjusted System Time is an unsigned
00285  * 64 bit value that wraps back to 0 every 584 years. Time intervals
00286  * may be either signed or unsigned.
00287  */
00288 typedef khronos_uint64_t          khronos_utime_nanoseconds_t;
00289 typedef khronos_int64_t           khronos_stime_nanoseconds_t;
00290 #endif
00291
00292 /*
00293  * Dummy value used to pad enum types to 32 bits.
00294  */
00295 #ifndef KHRONOS_MAX_ENUM
00296 #define KHRONOS_MAX_ENUM 0x7FFFFFFF
00297 #endif
00298
00299 /*
00300  * Enumerated boolean type
00301  *
00302  * Values other than zero should be considered to be true. Therefore
00303  * comparisons should not be made against KHRONOS_TRUE.
00304  */
00305 typedef enum {

```

```

00306     KHRONOS_FALSE = 0,
00307     KHRONOS_TRUE  = 1,
00308     KHRONOS_BOOLEAN_ENUM_FORCE_SIZE = KHRONOS_MAX_ENUM
00309 } khronos_boolean_enum_t;
00310
00311 #endif /* __khrplatform_h */

```

7.10 MenuState.hpp

```

00001 #ifndef MENUSTATE_HPP
00002 #define MENUSTATE_HPP
00003
00004 #include "App.hpp"
00005 #include "IState.hpp"
00006 #include "TexturedRect2D.hpp"
00007 #include <memory>
00008
00009 class MenuState : public IState {
00010     App &app;
00011
00012 public:
00013     MenuState(App &app);
00014     ~MenuState() override;
00015     void Run() override;
00016     void SendEvent(SDL_Event &event) override;
00017
00018 private:
00019     static constexpr SDL_Color TEXT_COLOR = {255, 255, 255};
00020
00021     // Title Variables
00022     std::shared_ptr<TexturedRect2D> m_titleText;
00023     static constexpr float TITLE_W_SCALE = 0.001;
00024     static constexpr float TITLE_H_SCALE = 0.001;
00025     static constexpr float TITLE_X_OFFSET = 0.0;
00026     static constexpr float TITLE_Y_OFFSET = 1.0 / 4.0;
00027
00028     // Menu Variables
00029     static constexpr float MENU_W_SCALE = 0.0004;
00030     static constexpr float MENU_H_SCALE = 0.0004;
00031
00032     std::shared_ptr<TexturedRect2D> m_startText;
00033     static constexpr float START_X_OFFSET = 0.0;
00034     static constexpr float START_Y_OFFSET = 0.0;
00035
00036     std::shared_ptr<TexturedRect2D> m_quitText;
00037     static constexpr float QUIT_X_OFFSET = 0.0;
00038     static constexpr float QUIT_Y_OFFSET = -1.0 / 8.0;
00039
00040     static constexpr int MENU_COUNT = 2;
00041
00042     // Selector
00043     std::shared_ptr<TexturedRect2D> m_selector;
00044     int selector_pos;
00045     static constexpr float SELECTOR_PADDING = 1.0 / 32.0;
00046     void SelectorNext();
00047     void SelectorPrev();
00048     void Select();
00049
00050     enum MenuOption { START = 0, QUIT = 1 };
00051
00052     // Instructions
00053     std::shared_ptr<TexturedRect2D> m_instructionText;
00054     static constexpr float INSTRUCTION_W_SCALE = 0.0002;
00055     static constexpr float INSTRUCTION_H_SCALE = 0.0002;
00056     static constexpr float INSTRUCTION_X_OFFSET = 0.3;
00057     static constexpr float INSTRUCTION_Y_OFFSET = -0.4;
00058 };
00059
00060 #endif

```

7.11 MissionState.hpp

```

00001 #ifndef MISSION_HPP
00002 #define MISSION_HPP
00003
00004 #include "App.hpp"
00005 #include "IState.hpp"
00006 #include "VimEmulator.hpp"
00007

```

```

00008 class MissionState : public IState {
00009     App &app;
00010
00011 public:
00012     MissionState(App &app);
00013     ~MissionState() override;
00014     void Run() override;
00015     void SendEvent(SDL_Event &event) override;
00016
00017 private:
00018     static constexpr float WIDTH_RATIO = 2 / 3.0;
00019     static constexpr int DRAW_R = 131;
00020     static constexpr int DRAW_G = 50;
00021     static constexpr int DRAW_B = 168;
00022
00023     std::shared_ptr<VimEmulator> m_vimEmulator;
00024 };
00025
00026 #endif

```

7.12 Rect2D.hpp

```

00001 #ifndef RECT2D_HPP
00002 #define RECT2D_HPP
00003
00004 #include <GraphicsController.hpp>
00005 #include <IRenderer.hpp>
00006 #include <SDL2/SDL.h>
00007 #include <ShaderProgram.hpp>
00008 #include <glad/glad.h>
00009 #include <vector>
00010
00011 class Rect2D : public IRender {
00012 protected:
00013     std::vector<GLfloat> m_vertexData;
00014     std::vector<GLuint> m_indexData;
00015     GLfloat m_x;
00016     GLfloat m_y;
00017     GLfloat m_width;
00018     GLfloat m_height;
00019
00020     GLuint m_texture_width;
00021     GLuint m_texture_height;
00022
00023     void GenBindBufferGL() override;
00024
00025 public:
00026     Rect2D(GLfloat xPos = 0, GLfloat yPos = 0, GLfloat width = 0,
00027           GLfloat height = 0);
00028
00029     virtual void UpdateVertexData();
00030
00031     void SetPosition(GLfloat xPos, GLfloat yPos);
00032
00033     void SetSize(GLfloat width, GLfloat height);
00034
00035     [[nodiscard]] GLfloat GetX() const;
00036     [[nodiscard]] GLfloat GetY() const;
00037     [[nodiscard]] GLfloat GetWidth() const;
00038     [[nodiscard]] GLfloat GetHeight() const;
00039
00040     [[nodiscard]] std::vector<GLfloat> GetVertexData() const;
00041
00042     ~Rect2D();
00043
00044     void Render() override;
00045
00046     void UpdateGL() override;
00047
00048     void SetShaderProgram(std::string shaderProgram);
00049
00050 private:
00051     static constexpr int RECT_ELEMENTS = 6;
00052 };
00053
00054 #endif

```

7.13 SDLToX11Keymap.hpp

```

00001 #ifndef SDLTUX11KEYMAP_HPP

```

```

00002 #define SDLTOX11KEYMAP_HPP
00003
00004 #include <SDL2/SDL.h>
00005 #include <X11/Xlib.h>
00006 #include <unordered_map>
00007
00008 class SDLToX11Keymap {
00009 public:
00010     SDLToX11Keymap();
00011     KeySym convert(SDL_Keycode sdlKey) const;
00012
00013 private:
00014     std::unordered_map<SDL_Keycode, KeySym> m_keyMap;
00015 };
00016
00017 #endif

```

7.14 ShaderProgram.hpp

```

00001 #ifndef SHADERPROGRAM_HPP
00002 #define SHADERPROGRAM_HPP
00003
00004 #include <glad/glad.h>
00005
00006 class ShaderProgram {
00007 private:
00008     friend class ShaderProgramBuilder;
00009     struct _passKey_t {
00010         _passKey_t() = default;
00011     };
00012     const GLuint m_program_id = 0;
00013
00014 public:
00015     ~ShaderProgram();
00016     [[nodiscard]] GLuint GetProgramId() const;
00017     ShaderProgram(_passKey_t, GLuint program_id);
00018 };
00019 #endif

```

7.15 ShaderProgramBuilder.hpp

```

00001 #ifndef SHADERPROGRAMBUILDER_HPP
00002 #define SHADERPROGRAMBUILDER_HPP
00003
00004 #include <Error.hpp>
00005 #include <ShaderProgram.hpp>
00006 #include <glad/glad.h>
00007 #include <memory>
00008 #include <optional>
00009 #include <string>
00010 #include <variant>
00011 #include <vector>
00012
00013 class ShaderProgramBuilder {
00014 public:
00015     ~ShaderProgramBuilder();
00016     ShaderProgramBuilder &LoadShaderFile(GLenum type, std::string file);
00017     ShaderProgramBuilder &LoadShader(GLenum type, std::string shaderCode);
00018     std::variant<Error, std::unique_ptr<ShaderProgram>> GenerateShaderProgram();
00019
00020 private:
00021     std::vector<GLuint> m_shaders;
00022     std::optional<Error> m_error = std::nullopt;
00023 };
00024
00025 #endif

```

7.16 TexturedRect2D.hpp

```

00001 #ifndef TEXTUREDRECT2D_HPP
00002 #define TEXTUREDRECT2D_HPP
00003
00004 #include <GraphicsController.hpp>
00005 #include <Rect2D.hpp>
00006 #include <SDL2/SDL.h>

```

```

00007 #include <glad/glad.h>
00008 #include <iostream>
00009 #include <vector>
00010
00011 class TexturedRect2D : public Rect2D {
00012     protected:
00013         GLuint m_texture;
00014         GLuint m_texture_width;
00015         GLuint m_texture_height;
00016
00017     public:
00018         TexturedRect2D(GLfloat xPos = 0, GLfloat yPos = 0, GLfloat width = 0,
00019             GLfloat height = 0);
00020
00021         void UpdateVertexData() override;
00022
00023         void SetTexture(void *data, GLuint width, GLuint height,
00024             bool force = false);
00025
00026         void SetTextureFormat(GLenum format);
00027
00028         void SetInternalTextureFormat(GLenum format);
00029
00030         void EnableTextureBlend();
00031
00032         void DisableTextureBlend();
00033
00034         ~TexturedRect2D();
00035
00036         void Render() override;
00037
00038         void UpdateGL() override;
00039
00040     private:
00041         GLenum m_textureFormat;
00042         GLenum m_internalTextureFormat;
00043         bool m_enableBlend;
00044 };
00045
00046 #endif

```

7.17 VimEmulator.hpp

```

00001 #ifndef VIMEMULATOR_HPP
00002 #define VIMEMULATOR_HPP
00003 #include <SDL2/SDL.h>
00004 #include <X11/Xlib.h>
00005 #include <mutex>
00006 #include <string>
00007
00008 #include "TexturedRect2D.hpp"
00009
00010 class VimEmulator : public TexturedRect2D {
00011     public:
00012         VimEmulator(std::string terminal, std::string nArg);
00013         virtual ~VimEmulator();
00014         void RegisterWindow();
00015         void ResizeWindow(int w, int h);
00016         void QueueFrame();
00017
00018         void SendSDLKey(SDL_Keycode key);
00019         void SetSDLMod(SDL_Keymod mod);
00020
00021         void Render() override;
00022
00023     private:
00024         static constexpr int REFRESH_MS = 100;
00025         Display *m_display;
00026         int m_screen;
00027         Window m_rootWindow;
00028         Window *m_window;
00029         std::string m_windowName;
00030         int m_width;
00031         int m_height;
00032         std::mutex m_mutex;
00033         pid_t m_pid;
00034
00035         static constexpr Uint32 R_MASK = 0xFF0000;
00036         static constexpr Uint32 G_MASK = 0x00FF00;
00037         static constexpr Uint32 B_MASK = 0x0000FF;
00038         static constexpr Uint32 A_MASK = 0;
00039         XImage *m_xImage;
00040         SDL_Surface *m_surface;
00041
00042
00043

```



```
00044     bool m_frameReady;
00045
00047     unsigned int *m_modmask;
00048
00050     SDL_Surface *GetFrameAsSurface();
00051     Window *findWindowByName(Window window);
00052     void RegisterWindowThread();
00053     void ResizeWindowThread(int w, int h);
00054     // Replace this with a generic event handler if needed
00055     bool MatchResizeEvent(int w, int h, XEvent *event);
00056     void QueueFrameThread();
00057 };
00058
00059 #endif
```


Index

- ~App
 - App, [12](#)
- ~IntroState
 - IntroState, [19](#)
- ~MenuState
 - MenuState, [23](#)
- ~MissionState
 - MissionState, [25](#)
- ~Rect2D
 - Rect2D, [27](#)
- ~ShaderProgram
 - ShaderProgram, [32](#)
- ~ShaderProgramBuilder
 - ShaderProgramBuilder, [33](#)
- ~TexturedRect2D
 - TexturedRect2D, [37](#)
- ~VimEmulator
 - VimEmulator, [41](#)
- AddRenderable
 - App, [13](#)
- App, [11](#)
 - ~App, [12](#)
 - AddRenderable, [13](#)
 - App, [12](#)
 - ClearRenderables, [13](#)
 - CreateGraphicsPipeline, [13](#)
 - DeltaTime, [13](#)
 - GenerateFonts, [13](#)
 - GetFPS, [13](#)
 - GetHeight, [14](#)
 - GetRenderer, [14](#)
 - GetWidth, [14](#)
 - m_state, [15](#)
 - m_width, [15](#)
 - Render, [14](#)
 - Run, [14](#)
 - SECOND_MS, [15](#)
 - SetState, [14](#)
 - Stop, [15](#)
- CheckGLObjectStatus
 - GraphicsController, [17](#)
- ClearRenderables
 - App, [13](#)
- convert
 - SDLToX11Keymap, [31](#)
- CreateGraphicsPipeline
 - App, [13](#)
- debugMessage
 - GraphicsController, [17](#)
- DeltaTime
 - App, [13](#)
- DisableTextureBlend
 - TexturedRect2D, [37](#)
- enableDebug
 - GraphicsController, [18](#)
- EnableTextureBlend
 - TexturedRect2D, [37](#)
- Error, [15](#)
 - Error, [15](#)
 - toString, [16](#)
- findWindowByName
 - VimEmulator, [41](#)
- GenBindBufferGL
 - IRender, [21](#)
 - Rect2D, [28](#)
- GenerateFonts
 - App, [13](#)
- GenerateShaderProgram
 - ShaderProgramBuilder, [33](#)
- GetFPS
 - App, [13](#)
- GetFrameAsSurface
 - VimEmulator, [41](#)
- GetHeight
 - App, [14](#)
 - Rect2D, [28](#)
- GetProgramId
 - ShaderProgram, [32](#)
- GetRenderer
 - App, [14](#)
- GetVertexData
 - Rect2D, [28](#)
- GetWidth
 - App, [14](#)
 - Rect2D, [28](#)
- GetX
 - Rect2D, [28](#)
- GetY
 - Rect2D, [28](#)
- gladGLversionStruct, [16](#)
- GraphicsController, [16](#)
 - CheckGLObjectStatus, [17](#)
 - debugMessage, [17](#)
 - enableDebug, [18](#)

- initGL, 18
- s_shaderPrograms, 18
- include/App.hpp, 45
- include/constants.hpp, 46
- include/Error.hpp, 46
- include/glad/glad.h, 46
- include/GraphicsController.hpp, 243
- include/IntroState.hpp, 244
- include/IRender.hpp, 244
- include/IState.hpp, 245
- include/KHR/khrplatform.h, 245
- include/MenuState.hpp, 249
- include/MissionState.hpp, 249
- include/Rect2D.hpp, 250
- include/SDLToX11Keymap.hpp, 250
- include/ShaderProgram.hpp, 251
- include/ShaderProgramBuilder.hpp, 251
- include/TexturedRect2D.hpp, 251
- include/VimEmulator.hpp, 252
- initGL
 - GraphicsController, 18
- IntroState, 18
 - ~IntroState, 19
 - IntroState, 19
 - NextSlide, 20
 - Run, 20
 - SendEvent, 20
- IntroState::Slide, 34
- IRender, 20
 - GenBindBufferGL, 21
 - Render, 21
 - UpdateGL, 21
- IState, 21
 - Run, 22
 - SendEvent, 22
- LoadShader
 - ShaderProgramBuilder, 33
- LoadShaderFile
 - ShaderProgramBuilder, 34
- m_modmask
 - VimEmulator, 44
- m_state
 - App, 15
- m_width
 - App, 15
- MatchResizeEvent
 - VimEmulator, 42
- MenuState, 22
 - ~MenuState, 23
 - MenuState, 23
 - Run, 24
 - Select, 24
 - SelectorNext, 24
 - SelectorPrev, 24
 - SendEvent, 24
- MissionState, 24
 - ~MissionState, 25
 - MissionState, 25
 - Run, 25
 - SendEvent, 25
- NextSlide
 - IntroState, 20
- QueueFrame
 - VimEmulator, 42
- QueueFrameThread
 - VimEmulator, 42
- R_MASK
 - VimEmulator, 44
- Rect2D, 26
 - ~Rect2D, 27
 - GenBindBufferGL, 28
 - GetHeight, 28
 - GetVertexData, 28
 - GetWidth, 28
 - GetX, 28
 - GetY, 28
 - Rect2D, 27
 - Render, 29
 - SetPosition, 29
 - SetShaderProgram, 29
 - SetSize, 29
 - UpdateGL, 30
 - UpdateVertexData, 30
- RegisterWindow
 - VimEmulator, 42
- RegisterWindowThread
 - VimEmulator, 42
- Render
 - App, 14
 - IRender, 21
 - Rect2D, 29
 - TexturedRect2D, 37
 - VimEmulator, 43
- ResizeWindow
 - VimEmulator, 43
- ResizeWindowThread
 - VimEmulator, 43
- Run
 - App, 14
 - IntroState, 20
 - IState, 22
 - MenuState, 24
 - MissionState, 25
- s_shaderPrograms
 - GraphicsController, 18
- SDLToX11Keymap, 30
 - convert, 31
 - SDLToX11Keymap, 31
- SECOND_MS
 - App, 15
- Select

- MenuState, [24](#)
- SelectorNext
 - MenuState, [24](#)
- SelectorPrev
 - MenuState, [24](#)
- SendEvent
 - IntroState, [20](#)
 - IState, [22](#)
 - MenuState, [24](#)
 - MissionState, [25](#)
- SendSDLKey
 - VimEmulator, [44](#)
- SetInternalTextureFormat
 - TexturedRect2D, [37](#)
- SetPosition
 - Rect2D, [29](#)
- SetSDLMod
 - VimEmulator, [44](#)
- SetShaderProgram
 - Rect2D, [29](#)
- SetSize
 - Rect2D, [29](#)
- SetState
 - App, [14](#)
- SetTexture
 - TexturedRect2D, [37](#)
- SetTextureFormat
 - TexturedRect2D, [38](#)
- ShaderProgram, [31](#)
 - ~ShaderProgram, [32](#)
 - GetProgramId, [32](#)
 - ShaderProgram, [32](#)
- ShaderProgram::_passKey_t, [11](#)
- ShaderProgramBuilder, [33](#)
 - ~ShaderProgramBuilder, [33](#)
 - GenerateShaderProgram, [33](#)
 - LoadShader, [33](#)
 - LoadShaderFile, [34](#)
- Stop
 - App, [15](#)
- TexturedRect2D, [35](#)
 - ~TexturedRect2D, [37](#)
 - DisableTextureBlend, [37](#)
 - EnableTextureBlend, [37](#)
 - Render, [37](#)
 - SetInternalTextureFormat, [37](#)
 - SetTexture, [37](#)
 - SetTextureFormat, [38](#)
 - TexturedRect2D, [36](#)
 - UpdateGL, [38](#)
 - UpdateVertexData, [38](#)
- Todo List, [3](#)
- toString
 - Error, [16](#)
- UpdateGL
 - IRender, [21](#)
 - Rect2D, [30](#)
 - TexturedRect2D, [38](#)
- UpdateVertexData
 - Rect2D, [30](#)
 - TexturedRect2D, [38](#)
- VimEmulator, [38](#)
 - ~VimEmulator, [41](#)
 - findWindowByName, [41](#)
 - GetFrameAsSurface, [41](#)
 - m_modmask, [44](#)
 - MatchResizeEvent, [42](#)
 - QueueFrame, [42](#)
 - QueueFrameThread, [42](#)
 - R_MASK, [44](#)
 - RegisterWindow, [42](#)
 - RegisterWindowThread, [42](#)
 - Render, [43](#)
 - ResizeWindow, [43](#)
 - ResizeWindowThread, [43](#)
 - SendSDLKey, [44](#)
 - SetSDLMod, [44](#)
 - VimEmulator, [41](#)
- Welcome to VimVentures!, [1](#)