# 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**

| | |
|---|---|
| *ssFlags* | Subsystem Flags |
| *x* | Window X Position |
| *y* | Window Y Position |
| *w* | Window Width |
| *h* | 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**

|  |  |
|---|---|
| *Renderable* |  |

### 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**

| | |
|---|---|
| *Game* | 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**

|  |  |
|---|---|
| *infoLogLen* | Length of the string |
| *infoLog* | 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**

| | |
|---|---|
| *object* | The ID of the OpenGL object (e.g. shaders, program) |
| *parameter* | The specific status parameter to query (e.g. GL_COMPILE_STATUS, GL_LINK_STATUS) |
| *getiv* | Parameter getter for the object type (e.g. glGetShaderiv) |
| *getInfoLog* | 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**

| | |
|---|---|
| *source* | |

**Parameters**

| | |
|---|---|
| *type* | |
| *debug_id* | |
| *severity* | |
| *length* | |
| *message* | |
| *userParam* | |

#### 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]
```

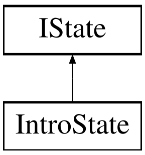
Static map storing shader programs, with strings as keys.

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.

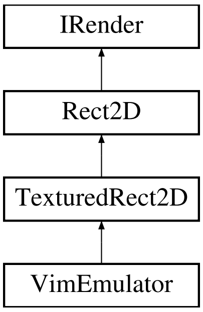
Implements IState.

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]
```

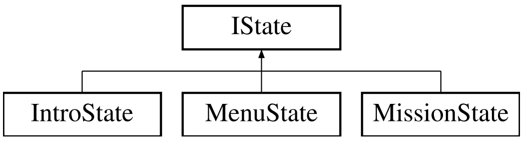
Implemented in Rect2D, and TexturedRect2D.

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]
```

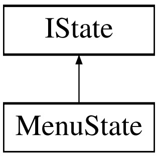
Implemented in IntroState, MenuState, and MissionState.

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.

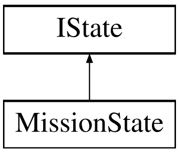
Implements IState.

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.

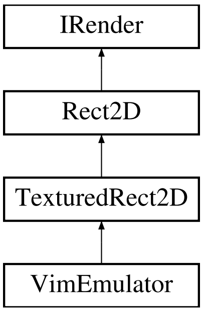
Implements IState.

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**

| | |
|---|---|
| *xPos* | X position |
| *yPos* | Y position |
| *width* | Width of the quad |
| *height* | 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**

| | |
|---|---|
| *xPos* | New x Position |
| *yPos* | New y Position |

#### 6.11.2.9 SetShaderProgram()

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

Set the shaderProgram to render this rectangle by name.

**Parameters**

| | |
|---|---|
| *shaderProgram* | 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**

| | |
|---|---|
| *width* | New quad width |
| *height* | 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**

| | |
|---|---|
| *SDL_Keycode* | |

**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**

| | |
|---|---|
| *passkey* | An instance of the _passKey_t struct. This restricts the constructor usage. |
| *program←_id* | 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::Generate↩
ShaderProgram ( )
```

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**

| | |
|---|---|
| *type* | Shader type |
| *shaderCode* | 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**

| | |
|---|---|
| *type* | Shader type |
| *filePath* | 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

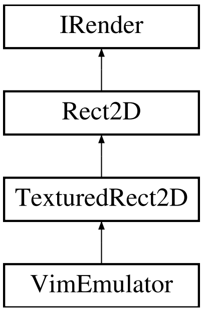
- SDL_Surface ∗ **slideSurface**
- float **startX**
- float **stopX**
- float **startY**
- float **stopY**
- float **panSpeedMS**

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**

| | |
|---|---|
| *xPos* | X position |
| *yPos* | Y position |
| *width* | Width of the quad |
| *height* | 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**

| | |
|---|---|
| *data* | Pointer to pixel data |
| *width* | Width of the texture in pixels |
| *height* | Height of the texture in pixels |
| *force* | 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.

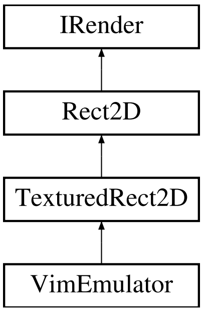
Reimplemented from Rect2D.

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**

| | |
|---|---|
| *terminal* | The name of the terminal program, E.g. 'Alacritty' |
| *nArg* | 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**

| | |
|---|---|
| *window* | 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**

| | |
|---|---|
| *w* | Width |
| *h* | Height |
| *event* | 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**

| | |
|---|---|
| *renderer* | 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**

| | |
|---|---|
| *w* | Width |
| *h* | 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**

| | |
|---|---|
| *w* | Width |
| *h* | 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**

| | |
|---|---|
| *key* | An SDL_Keycode |

#### 6.17.2.12 SetSDLMod()

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

Sets the modmask using an SDL_Keymod.

**Parameters**

| | |
|---|---|
| *mod* | 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

```
#ifndef APP_HPP
#define APP_HPP

#include "IRender.hpp"
#include "IState.hpp"
#include <SDL2/SDL.h>
#include <SDL2/SDL_ttf.h>
#include <glad/glad.h>
#include <memory>
#include <vector>

class App {
  public:
    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();

    [[nodiscard]] double GetFPS() const;

    [[nodiscard]] double DeltaTime() const;

    [[nodiscard]] int GetWidth() const;

    [[nodiscard]] int GetHeight() const;

  private:
    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;

    static constexpr double SECOND_MS = 1000.0;
    Uint32 m_startTick;
    Uint32 m_endTick;
    double m_deltaTime;

    // Shaders
    static constexpr int APP_GL_DEPTH_SIZE = 24;
    static void debugMessage(GLenum source, GLenum type, GLuint debug_id,
                             GLenum severity, GLsizei length,
                             const GLchar *message, const void *userParam);
    static void CreateGraphicsPipeline();

    // Fonts
    static constexpr int FONT_RENDER_SIZE = 128;
    static void GenerateFonts();

    // Draw
    void PreDraw() const;
};

#endif
```

## 7.2 constants.hpp

```
#ifndef CONSTANTS_HPP
#define CONSTANTS_HPP

#include "SDLToX11Keymap.hpp"

constexpr char APP_TITLE[] = "VimVentures";
constexpr char APP_VERSION[] = "MEDIEVAL 0.0.3";

extern const SDLToX11Keymap &SDLX11KeymapRef;

#endif
```

## 7.3 Error.hpp

```
#ifndef ERROR_HPP
#define ERROR_HPP

#include <string>

struct Error {
    const int m_infoLogLen;
    const char *m_infoLog;
    Error(int infoLogLen, const char *infoLog);
    [[nodiscard]] std::string toString() const;
};

#endif
```

## 7.4 glad.h

```
/*

    OpenGL loader generated by glad 0.1.36 on Fri Aug  2 12:24:15 2024.

    Language/Generator: C/C++
    Specification: gl
    APIs: gl=4.1
    Profile: core
    Extensions:
        GL_3DFX_multisample,
        GL_3DFX_tbuffer,
        GL_3DFX_texture_compression_FXT1,
        GL_AMD_blend_minmax_factor,
        GL_AMD_conservative_depth,
        GL_AMD_debug_output,
        GL_AMD_depth_clamp_separate,
        GL_AMD_draw_buffers_blend,
        GL_AMD_framebuffer_multisample_advanced,
        GL_AMD_framebuffer_sample_positions,
        GL_AMD_gcn_shader,
        GL_AMD_gpu_shader_half_float,
        GL_AMD_gpu_shader_int16,
        GL_AMD_gpu_shader_int64,
        GL_AMD_interleaved_elements,
        GL_AMD_multi_draw_indirect,
        GL_AMD_name_gen_delete,
        GL_AMD_occlusion_query_event,
        GL_AMD_performance_monitor,
        GL_AMD_pinned_memory,
        GL_AMD_query_buffer_object,
        GL_AMD_sample_positions,
        GL_AMD_seamless_cubemap_per_texture,
        GL_AMD_shader_atomic_counter_ops,
        GL_AMD_shader_ballot,
        GL_AMD_shader_explicit_vertex_parameter,
        GL_AMD_shader_gpu_shader_half_float_fetch,
        GL_AMD_shader_image_load_store_lod,
        GL_AMD_shader_stencil_export,
        GL_AMD_shader_trinary_minmax,
        GL_AMD_sparse_texture,
        GL_AMD_stencil_operation_extended,
        GL_AMD_texture_gather_bias_lod,
        GL_AMD_texture_texture4,
        GL_AMD_transform_feedback3_lines_triangles,
        GL_AMD_transform_feedback4,
        GL_AMD_vertex_shader_layer,
        GL_AMD_vertex_shader_tessellator,
        GL_AMD_vertex_shader_viewport_index,
        GL_APPLE_aux_depth_stencil,
        GL_APPLE_client_storage,
        GL_APPLE_element_array,
        GL_APPLE_fence,
        GL_APPLE_float_pixels,
        GL_APPLE_flush_buffer_range,
        GL_APPLE_object_purgeable,
        GL_APPLE_rgb_422,
        GL_APPLE_row_bytes,
        GL_APPLE_specular_vector,
        GL_APPLE_texture_range,
        GL_APPLE_transform_hint,
        GL_APPLE_vertex_array_object,
        GL_APPLE_vertex_array_range,
        GL_APPLE_vertex_program_evaluators,
        GL_APPLE_ycbcr_422,
        GL_ARB_ES2_compatibility,
        GL_ARB_ES3_1_compatibility,
        GL_ARB_ES3_2_compatibility,
        GL_ARB_ES3_compatibility,
        GL_ARB_arrays_of_arrays,
        GL_ARB_base_instance,
        GL_ARB_bindless_texture,
        GL_ARB_blend_func_extended,
        GL_ARB_buffer_storage,
        GL_ARB_cl_event,
        GL_ARB_clear_buffer_object,
        GL_ARB_clear_texture,
        GL_ARB_clip_control,
        GL_ARB_color_buffer_float,
        GL_ARB_compatibility,
        GL_ARB_compressed_texture_pixel_storage,
        GL_ARB_compute_shader,
        GL_ARB_compute_variable_group_size,
        GL_ARB_conditional_render_inverted,
        GL_ARB_conservative_depth,
        GL_ARB_copy_buffer,
        GL_ARB_copy_image,
        GL_ARB_cull_distance,
        GL_ARB_debug_output,
        GL_ARB_depth_buffer_float,
        GL_ARB_depth_clamp,
        GL_ARB_depth_texture,
        GL_ARB_derivative_control,
        GL_ARB_direct_state_access,
        GL_ARB_draw_buffers,
        GL_ARB_draw_buffers_blend,
        GL_ARB_draw_elements_base_vertex,
        GL_ARB_draw_indirect,
        GL_ARB_draw_instanced,
        GL_ARB_enhanced_layouts,
        GL_ARB_explicit_attrib_location,
        GL_ARB_explicit_uniform_location,
        GL_ARB_fragment_coord_conventions,
        GL_ARB_fragment_layer_viewport,
        GL_ARB_fragment_program,
        GL_ARB_fragment_program_shadow,
        GL_ARB_fragment_shader,
        GL_ARB_fragment_shader_interlock,
        GL_ARB_framebuffer_no_attachments,
        GL_ARB_framebuffer_object,
        GL_ARB_framebuffer_sRGB,
        GL_ARB_geometry_shader4,
        GL_ARB_get_program_binary,
        GL_ARB_get_texture_sub_image,
        GL_ARB_gl_spirv,
        GL_ARB_gpu_shader5,
        GL_ARB_gpu_shader_fp64,
        GL_ARB_gpu_shader_int64,
        GL_ARB_half_float_pixel,
        GL_ARB_half_float_vertex,
        GL_ARB_imaging,
        GL_ARB_indirect_parameters,
        GL_ARB_instanced_arrays,
        GL_ARB_internalformat_query,
        GL_ARB_internalformat_query2,
        GL_ARB_invalidate_subdata,
        GL_ARB_map_buffer_alignment,
        GL_ARB_map_buffer_range,
        GL_ARB_matrix_palette,
        GL_ARB_multi_bind,
        GL_ARB_multi_draw_indirect,
        GL_ARB_multisample,
        GL_ARB_multitexture,
        GL_ARB_occlusion_query,
        GL_ARB_occlusion_query2,
        GL_ARB_parallel_shader_compile,
        GL_ARB_pipeline_statistics_query,
        GL_ARB_pixel_buffer_object,
        GL_ARB_point_parameters,
        GL_ARB_point_sprite,
        GL_ARB_polygon_offset_clamp,
        GL_ARB_post_depth_coverage,
        GL_ARB_program_interface_query,
        GL_ARB_provoking_vertex,
        GL_ARB_query_buffer_object,
        GL_ARB_robust_buffer_access_behavior,
        GL_ARB_robustness,
        GL_ARB_robustness_isolation,
        GL_ARB_sample_locations,
        GL_ARB_sample_shading,
        GL_ARB_sampler_objects,
        GL_ARB_seamless_cube_map,
        GL_ARB_seamless_cubemap_per_texture,
        GL_ARB_separate_shader_objects,
        GL_ARB_shader_atomic_counter_ops,
        GL_ARB_shader_atomic_counters,
        GL_ARB_shader_ballot,
        GL_ARB_shader_bit_encoding,
        GL_ARB_shader_clock,
        GL_ARB_shader_draw_parameters,
        GL_ARB_shader_group_vote,
        GL_ARB_shader_image_load_store,
        GL_ARB_shader_image_size,
        GL_ARB_shader_objects,
        GL_ARB_shader_precision,
        GL_ARB_shader_stencil_export,
        GL_ARB_shader_storage_buffer_object,
        GL_ARB_shader_subroutine,
        GL_ARB_shader_texture_image_samples,
        GL_ARB_shader_texture_lod,
        GL_ARB_shader_viewport_layer_array,
        GL_ARB_shading_language_100,
        GL_ARB_shading_language_420pack,
        GL_ARB_shading_language_include,
        GL_ARB_shading_language_packing,
        GL_ARB_shadow,
        GL_ARB_shadow_ambient,
        GL_ARB_sparse_buffer,
        GL_ARB_sparse_texture,
        GL_ARB_sparse_texture2,
        GL_ARB_sparse_texture_clamp,
        GL_ARB_spirv_extensions,
        GL_ARB_stencil_texturing,
        GL_ARB_sync,
        GL_ARB_tessellation_shader,
        GL_ARB_texture_barrier,
        GL_ARB_texture_border_clamp,
        GL_ARB_texture_buffer_object,
        GL_ARB_texture_buffer_object_rgb32,
        GL_ARB_texture_buffer_range,
        GL_ARB_texture_compression,
        GL_ARB_texture_compression_bptc,
        GL_ARB_texture_compression_rgtc,
        GL_ARB_texture_cube_map,
        GL_ARB_texture_cube_map_array,
        GL_ARB_texture_env_add,
        GL_ARB_texture_env_combine,
        GL_ARB_texture_env_crossbar,
        GL_ARB_texture_env_dot3,
        GL_ARB_texture_filter_anisotropic,
        GL_ARB_texture_filter_minmax,
        GL_ARB_texture_float,
        GL_ARB_texture_gather,
        GL_ARB_texture_mirror_clamp_to_edge,
        GL_ARB_texture_mirrored_repeat,
        GL_ARB_texture_multisample,
        GL_ARB_texture_non_power_of_two,
        GL_ARB_texture_query_levels,
        GL_ARB_texture_query_lod,
        GL_ARB_texture_rectangle,
        GL_ARB_texture_rg,
        GL_ARB_texture_rgb10_a2ui,
        GL_ARB_texture_stencil8,
        GL_ARB_texture_storage,
        GL_ARB_texture_storage_multisample,
        GL_ARB_texture_swizzle,
        GL_ARB_texture_view,
        GL_ARB_timer_query,
        GL_ARB_transform_feedback2,
        GL_ARB_transform_feedback3,
        GL_ARB_transform_feedback_instanced,
        GL_ARB_transform_feedback_overflow_query,
        GL_ARB_transpose_matrix,
        GL_ARB_uniform_buffer_object,
        GL_ARB_vertex_array_bgra,
        GL_ARB_vertex_array_object,
        GL_ARB_vertex_attrib_64bit,
        GL_ARB_vertex_attrib_binding,
        GL_ARB_vertex_blend,
        GL_ARB_vertex_buffer_object,
        GL_ARB_vertex_program,
        GL_ARB_vertex_shader,
        GL_ARB_vertex_type_10f_11f_11f_rev,
        GL_ARB_vertex_type_2_10_10_10_rev,
        GL_ARB_viewport_array,
        GL_ARB_window_pos,
        GL_ATI_draw_buffers,
        GL_ATI_element_array,
        GL_ATI_envmap_bumpmap,
        GL_ATI_fragment_shader,
        GL_ATI_map_object_buffer,
        GL_ATI_meminfo,
        GL_ATI_pixel_format_float,
        GL_ATI_pn_triangles,
        GL_ATI_separate_stencil,
        GL_ATI_text_fragment_shader,
        GL_ATI_texture_env_combine3,
        GL_ATI_texture_float,
        GL_ATI_texture_mirror_once,
        GL_ATI_vertex_array_object,
        GL_ATI_vertex_attrib_array_object,
        GL_ATI_vertex_streams,
        GL_EXT_422_pixels,
        GL_EXT_EGL_image_storage,
        GL_EXT_EGL_sync,
        GL_EXT_abgr,
        GL_EXT_bgra,
        GL_EXT_bindable_uniform,
        GL_EXT_blend_color,
        GL_EXT_blend_equation_separate,
        GL_EXT_blend_func_separate,
        GL_EXT_blend_logic_op,
        GL_EXT_blend_minmax,
        GL_EXT_blend_subtract,
        GL_EXT_clip_volume_hint,
        GL_EXT_cmyka,
        GL_EXT_color_subtable,
        GL_EXT_compiled_vertex_array,
        GL_EXT_convolution,
        GL_EXT_coordinate_frame,
        GL_EXT_copy_texture,
        GL_EXT_cull_vertex,
        GL_EXT_debug_label,
        GL_EXT_debug_marker,
        GL_EXT_depth_bounds_test,
        GL_EXT_direct_state_access,
        GL_EXT_draw_buffers2,
        GL_EXT_draw_instanced,
        GL_EXT_draw_range_elements,
        GL_EXT_external_buffer,
        GL_EXT_fog_coord,
        GL_EXT_framebuffer_blit,
        GL_EXT_framebuffer_blit_layers,
        GL_EXT_framebuffer_multisample,
        GL_EXT_framebuffer_multisample_blit_scaled,
        GL_EXT_framebuffer_object,
        GL_EXT_framebuffer_sRGB,
        GL_EXT_geometry_shader4,
        GL_EXT_gpu_program_parameters,
        GL_EXT_gpu_shader4,
        GL_EXT_histogram,
        GL_EXT_index_array_formats,
        GL_EXT_index_func,
        GL_EXT_index_material,
        GL_EXT_index_texture,
        GL_EXT_light_texture,
        GL_EXT_memory_object,
        GL_EXT_memory_object_fd,
        GL_EXT_memory_object_win32,
        GL_EXT_misc_attribute,
        GL_EXT_multi_draw_arrays,
        GL_EXT_multisample,
        GL_EXT_multiview_tessellation_geometry_shader,
        GL_EXT_multiview_texture_multisample,
        GL_EXT_multiview_timer_query,
        GL_EXT_packed_depth_stencil,
        GL_EXT_packed_float,
        GL_EXT_packed_pixels,
        GL_EXT_paletted_texture,
        GL_EXT_pixel_buffer_object,
        GL_EXT_pixel_transform,
        GL_EXT_pixel_transform_color_table,
        GL_EXT_point_parameters,
        GL_EXT_polygon_offset,
        GL_EXT_polygon_offset_clamp,
        GL_EXT_post_depth_coverage,
        GL_EXT_provoking_vertex,
        GL_EXT_raster_multisample,
        GL_EXT_rescale_normal,
        GL_EXT_secondary_color,
        GL_EXT_semaphore,
        GL_EXT_semaphore_fd,
        GL_EXT_semaphore_win32,
        GL_EXT_separate_shader_objects,
        GL_EXT_separate_specular_color,
        GL_EXT_shader_framebuffer_fetch,
        GL_EXT_shader_framebuffer_fetch_non_coherent,
        GL_EXT_shader_image_load_formatted,
        GL_EXT_shader_image_load_store,
        GL_EXT_shader_integer_mix,
        GL_EXT_shader_samples_identical,
        GL_EXT_shadow_funcs,
        GL_EXT_shared_texture_palette,
        GL_EXT_sparse_texture2,
        GL_EXT_stencil_clear_tag,
        GL_EXT_stencil_two_side,
        GL_EXT_stencil_wrap,
        GL_EXT_subtexture,
        GL_EXT_texture,
        GL_EXT_texture3D,
        GL_EXT_texture_array,
        GL_EXT_texture_buffer_object,
        GL_EXT_texture_compression_latc,
        GL_EXT_texture_compression_rgtc,
        GL_EXT_texture_compression_s3tc,
        GL_EXT_texture_cube_map,
        GL_EXT_texture_env_add,
        GL_EXT_texture_env_combine,
        GL_EXT_texture_env_dot3,
        GL_EXT_texture_filter_anisotropic,
        GL_EXT_texture_filter_minmax,
        GL_EXT_texture_integer,
        GL_EXT_texture_lod_bias,
        GL_EXT_texture_mirror_clamp,
        GL_EXT_texture_object,
        GL_EXT_texture_perturb_normal,
        GL_EXT_texture_sRGB,
        GL_EXT_texture_sRGB_R8,
        GL_EXT_texture_sRGB_RG8,
        GL_EXT_texture_sRGB_decode,
        GL_EXT_texture_shadow_lod,
        GL_EXT_texture_shared_exponent,
        GL_EXT_texture_snorm,
        GL_EXT_texture_storage,
        GL_EXT_texture_swizzle,
        GL_EXT_timer_query,
        GL_EXT_transform_feedback,
        GL_EXT_vertex_array,
        GL_EXT_vertex_array_bgra,
        GL_EXT_vertex_attrib_64bit,
        GL_EXT_vertex_shader,
        GL_EXT_vertex_weighting,
        GL_EXT_win32_keyed_mutex,
        GL_EXT_window_rectangles,
        GL_EXT_x11_sync_object,
        GL_GREMEDY_frame_terminator,
        GL_GREMEDY_string_marker,
        GL_HP_convolution_border_modes,
        GL_HP_image_transform,
        GL_HP_occlusion_test,
        GL_HP_texture_lighting,
        GL_IBM_cull_vertex,
        GL_IBM_multimode_draw_arrays,
        GL_IBM_rasterpos_clip,
        GL_IBM_static_data,
        GL_IBM_texture_mirrored_repeat,
        GL_IBM_vertex_array_lists,
        GL_INGR_blend_func_separate,
        GL_INGR_color_clamp,
        GL_INGR_interlace_read,
        GL_INTEL_blackhole_render,
        GL_INTEL_conservative_rasterization,
        GL_INTEL_fragment_shader_ordering,
        GL_INTEL_framebuffer_CMAA,
        GL_INTEL_map_texture,
        GL_INTEL_parallel_arrays,
        GL_INTEL_performance_query,
        GL_KHR_blend_equation_advanced,
        GL_KHR_blend_equation_advanced_coherent,
        GL_KHR_context_flush_control,
        GL_KHR_debug,
        GL_KHR_no_error,
        GL_KHR_parallel_shader_compile,
        GL_KHR_robust_buffer_access_behavior,
        GL_KHR_robustness,
        GL_KHR_shader_subgroup,
        GL_KHR_texture_compression_astc_hdr,
        GL_KHR_texture_compression_astc_ldr,
        GL_KHR_texture_compression_astc_sliced_3d,
        GL_MESAX_texture_stack,
        GL_MESA_framebuffer_flip_x,
        GL_MESA_framebuffer_flip_y,
        GL_MESA_framebuffer_swap_xy,
        GL_MESA_pack_invert,
        GL_MESA_program_binary_formats,
        GL_MESA_resize_buffers,
        GL_MESA_shader_integer_functions,
        GL_MESA_tile_raster_order,
        GL_MESA_window_pos,
        GL_MESA_ycbcr_texture,
        GL_NVX_blend_equation_advanced_multi_draw_buffers,
        GL_NVX_conditional_render,
        GL_NVX_gpu_memory_info,
        GL_NVX_gpu_multicast2,
        GL_NVX_linked_gpu_multicast,
        GL_NVX_progress_fence,
        GL_NV_alpha_to_coverage_dither_control,
        GL_NV_bindless_multi_draw_indirect,
        GL_NV_bindless_multi_draw_indirect_count,
        GL_NV_bindless_texture,
        GL_NV_blend_equation_advanced,
        GL_NV_blend_equation_advanced_coherent,
        GL_NV_blend_minmax_factor,
        GL_NV_blend_square,
        GL_NV_clip_space_w_scaling,
        GL_NV_command_list,
        GL_NV_compute_program5,
        GL_NV_compute_shader_derivatives,
        GL_NV_conditional_render,
        GL_NV_conservative_raster,
        GL_NV_conservative_raster_dilate,
        GL_NV_conservative_raster_pre_snap,
        GL_NV_conservative_raster_pre_snap_triangles,
        GL_NV_conservative_raster_underestimation,
        GL_NV_copy_depth_to_color,
        GL_NV_copy_image,
        GL_NV_deep_texture3D,
        GL_NV_depth_buffer_float,
        GL_NV_depth_clamp,
        GL_NV_draw_texture,
        GL_NV_draw_vulkan_image,
        GL_NV_evaluators,
        GL_NV_explicit_multisample,
        GL_NV_fence,
        GL_NV_fill_rectangle,
        GL_NV_float_buffer,
        GL_NV_fog_distance,
        GL_NV_fragment_coverage_to_color,
        GL_NV_fragment_program,
        GL_NV_fragment_program2,
        GL_NV_fragment_program4,
        GL_NV_fragment_program_option,
        GL_NV_fragment_shader_barycentric,
        GL_NV_fragment_shader_interlock,
        GL_NV_framebuffer_mixed_samples,
        GL_NV_framebuffer_multisample_coverage,
        GL_NV_geometry_program4,
        GL_NV_geometry_shader4,
        GL_NV_geometry_shader_passthrough,
        GL_NV_gpu_multicast,
        GL_NV_gpu_program4,
        GL_NV_gpu_program5,
        GL_NV_gpu_program5_mem_extended,
        GL_NV_gpu_shader5,
        GL_NV_half_float,
        GL_NV_internalformat_sample_query,
        GL_NV_light_max_exponent,
        GL_NV_memory_attachment,
        GL_NV_memory_object_sparse,
        GL_NV_mesh_shader,
        GL_NV_multisample_coverage,
        GL_NV_multisample_filter_hint,
        GL_NV_occlusion_query,
        GL_NV_packed_depth_stencil,
        GL_NV_parameter_buffer_object,
        GL_NV_parameter_buffer_object2,
        GL_NV_path_rendering,
        GL_NV_path_rendering_shared_edge,
        GL_NV_pixel_data_range,
        GL_NV_point_sprite,
        GL_NV_present_video,
        GL_NV_primitive_restart,
        GL_NV_primitive_shading_rate,
        GL_NV_query_resource,
        GL_NV_query_resource_tag,
        GL_NV_register_combiners,
        GL_NV_register_combiners2,
        GL_NV_representative_fragment_test,
        GL_NV_robustness_video_memory_purge,
        GL_NV_sample_locations,
        GL_NV_sample_mask_override_coverage,
        GL_NV_scissor_exclusive,
        GL_NV_shader_atomic_counters,
        GL_NV_shader_atomic_float,
        GL_NV_shader_atomic_float64,
        GL_NV_shader_atomic_fp16_vector,
        GL_NV_shader_atomic_int64,
        GL_NV_shader_buffer_load,
        GL_NV_shader_buffer_store,
        GL_NV_shader_storage_buffer_object,
        GL_NV_shader_subgroup_partitioned,
        GL_NV_shader_texture_footprint,
        GL_NV_shader_thread_group,
        GL_NV_shader_thread_shuffle,
        GL_NV_shading_rate_image,
        GL_NV_stereo_view_rendering,
        GL_NV_tessellation_program5,
        GL_NV_texgen_emboss,
        GL_NV_texgen_reflection,
        GL_NV_texture_barrier,
        GL_NV_texture_compression_vtc,
        GL_NV_texture_env_combine4,
        GL_NV_texture_expand_normal,
        GL_NV_texture_multisample,
        GL_NV_texture_rectangle,
        GL_NV_texture_rectangle_compressed,
        GL_NV_texture_shader,
        GL_NV_texture_shader2,
        GL_NV_texture_shader3,
        GL_NV_timeline_semaphore,
        GL_NV_transform_feedback,
        GL_NV_transform_feedback2,
        GL_NV_uniform_buffer_std430_layout,
        GL_NV_uniform_buffer_unified_memory,
        GL_NV_vdpau_interop,
        GL_NV_vdpau_interop2,
        GL_NV_vertex_array_range,
        GL_NV_vertex_array_range2,
        GL_NV_vertex_attrib_integer_64bit,
        GL_NV_vertex_buffer_unified_memory,
        GL_NV_vertex_program,
        GL_NV_vertex_program1_1,
        GL_NV_vertex_program2,
        GL_NV_vertex_program2_option,
        GL_NV_vertex_program3,
        GL_NV_vertex_program4,
        GL_NV_video_capture,
        GL_NV_viewport_array2,
        GL_NV_viewport_swizzle,
        GL_OES_byte_coordinates,
        GL_OES_compressed_paletted_texture,
        GL_OES_fixed_point,
        GL_OES_query_matrix,
        GL_OES_read_format,
        GL_OES_single_precision,
        GL_OML_interlace,
        GL_OML_resample,
        GL_OML_subsample,
        GL_OVR_multiview,
        GL_OVR_multiview2,
        GL_PGI_misc_hints,
        GL_PGI_vertex_hints,
        GL_REND_screen_coordinates,
        GL_S3_s3tc,
        GL_SGIS_detail_texture,
        GL_SGIS_fog_function,
        GL_SGIS_generate_mipmap,
        GL_SGIS_multisample,
        GL_SGIS_pixel_texture,
        GL_SGIS_point_line_texgen,
        GL_SGIS_point_parameters,
        GL_SGIS_sharpen_texture,
        GL_SGIS_texture4D,
        GL_SGIS_texture_border_clamp,
        GL_SGIS_texture_color_mask,
        GL_SGIS_texture_edge_clamp,
        GL_SGIS_texture_filter4,
        GL_SGIS_texture_lod,
        GL_SGIS_texture_select,
        GL_SGIX_async,
        GL_SGIX_async_histogram,
        GL_SGIX_async_pixel,
        GL_SGIX_blend_alpha_minmax,
        GL_SGIX_calligraphic_fragment,
        GL_SGIX_clipmap,
        GL_SGIX_convolution_accuracy,
        GL_SGIX_depth_pass_instrument,
        GL_SGIX_depth_texture,
        GL_SGIX_flush_raster,
        GL_SGIX_fog_offset,
        GL_SGIX_fragment_lighting,
        GL_SGIX_framezoom,
        GL_SGIX_igloo_interface,
        GL_SGIX_instruments,
        GL_SGIX_interlace,
        GL_SGIX_ir_instrument1,
        GL_SGIX_list_priority,
        GL_SGIX_pixel_texture,
        GL_SGIX_pixel_tiles,
        GL_SGIX_polynomial_ffd,
        GL_SGIX_reference_plane,
        GL_SGIX_resample,
        GL_SGIX_scalebias_hint,
        GL_SGIX_shadow,
        GL_SGIX_shadow_ambient,
        GL_SGIX_sprite,
        GL_SGIX_subsample,
        GL_SGIX_tag_sample_buffer,
        GL_SGIX_texture_add_env,
        GL_SGIX_texture_coordinate_clamp,
        GL_SGIX_texture_lod_bias,
        GL_SGIX_texture_multi_buffer,
        GL_SGIX_texture_scale_bias,
        GL_SGIX_vertex_preclip,
        GL_SGIX_ycrcb,
        GL_SGIX_ycrcb_subsample,
        GL_SGIX_ycrcba,
        GL_SGI_color_matrix,
        GL_SGI_color_table,
        GL_SGI_texture_color_table,
        GL_SUNX_constant_data,
        GL_SUN_convolution_border_modes,
        GL_SUN_global_alpha,
        GL_SUN_mesh_array,
        GL_SUN_slice_accum,
        GL_SUN_triangle_list,
        GL_SUN_vertex,
        GL_WIN_phong_shading,
        GL_WIN_specular_fog
    Loader: True
    Local files: False
    Omit khrplatform: False
    Reproducible: False

    Commandline:
        --profile="core" --api="gl=4.1" --generator="c" --spec="gl"
--extensions="GL_3DFX_multisample,GL_3DFX_tbuffer,GL_3DFX_texture_compression_FXT1,GL_AMD_blend_minmax_factor,GL_AMD_co
    Online:
        Too many extensions
```

```
*/


#ifndef __glad_h_
#define __glad_h_

#ifdef __gl_h_
#error OpenGL header already included, remove this include, glad already provides it
#endif
#define __gl_h_

#if defined(_WIN32) && !defined(APIENTRY) && !defined(__CYGWIN__) && !defined(__SCITECH_SNAP__)
#define APIENTRY __stdcall
#endif

#ifndef APIENTRY
#define APIENTRY
#endif
#ifndef APIENTRYP
#define APIENTRYP APIENTRY *
#endif

#ifndef GLAPIENTRY
#define GLAPIENTRY APIENTRY
#endif

#ifdef __cplusplus
extern "C" {
#endif

struct gladGLversionStruct {
    int major;
    int minor;
};

typedef void* (* GLADloadproc)(const char *name);

#ifndef GLAPI
# if defined(GLAD_GLAPI_EXPORT)
#  if defined(_WIN32) || defined(__CYGWIN__)
#   if defined(GLAD_GLAPI_EXPORT_BUILD)
#    if defined(__GNUC__)
#     define GLAPI __attribute__ ((dllexport)) extern
#    else
#     define GLAPI __declspec(dllexport) extern
#    endif
#   else
#    if defined(__GNUC__)
#     define GLAPI __attribute__ ((dllimport)) extern
#    else
#     define GLAPI __declspec(dllimport) extern
#    endif
#   endif
#  elif defined(__GNUC__) && defined(GLAD_GLAPI_EXPORT_BUILD)
#   define GLAPI __attribute__ ((visibility ("default"))) extern
#  else
#   define GLAPI extern
#  endif
# else
#  define GLAPI extern
# endif
#endif

GLAPI struct gladGLversionStruct GLVersion;

GLAPI int gladLoadGL(void);

GLAPI int gladLoadGLLoader(GLADloadproc);

#include <KHR/khrplatform.h>
typedef unsigned int GLenum;
typedef unsigned char GLboolean;
typedef unsigned int GLbitfield;
typedef void GLvoid;
typedef khronos_int8_t GLbyte;
typedef khronos_uint8_t GLubyte;
typedef khronos_int16_t GLshort;
typedef khronos_uint16_t GLushort;
typedef int GLint;
typedef unsigned int GLuint;
typedef khronos_int32_t GLclampx;
typedef int GLsizei;
typedef khronos_float_t GLfloat;
typedef khronos_float_t GLclampf;
typedef double GLdouble;
typedef double GLclampd;
typedef void *GLeglClientBufferEXT;
typedef void *GLeglImageOES;
typedef char GLchar;
typedef char GLcharARB;
#ifdef __APPLE__
typedef void *GLhandleARB;
#else
typedef unsigned int GLhandleARB;
#endif
typedef khronos_uint16_t GLhalf;
typedef khronos_uint16_t GLhalfARB;
typedef khronos_int32_t GLfixed;
typedef khronos_intptr_t GLintptr;
typedef khronos_intptr_t GLintptrARB;
typedef khronos_ssize_t GLsizeiptr;
typedef khronos_ssize_t GLsizeiptrARB;
typedef khronos_int64_t GLint64;
typedef khronos_int64_t GLint64EXT;
typedef khronos_uint64_t GLuint64;
typedef khronos_uint64_t GLuint64EXT;
typedef struct __GLsync *GLsync;
struct _cl_context;
struct _cl_event;
typedef void (APIENTRY *GLDEBUGPROC)(GLenum source,GLenum type,GLuint id,GLenum severity,GLsizei length,const GLchar *message,const void *userParam);
typedef void (APIENTRY *GLDEBUGPROCARB)(GLenum source,GLenum type,GLuint id,GLenum severity,GLsizei length,const GLchar *message,const void *userParam);
typedef void (APIENTRY *GLDEBUGPROCKHR)(GLenum source,GLenum type,GLuint id,GLenum severity,GLsizei length,const GLchar *message,const void *userParam);
typedef void (APIENTRY *GLDEBUGPROCAMD)(GLuint id,GLenum category,GLenum severity,GLsizei length,const GLchar *message,void *userParam);
typedef unsigned short GLhalfNV;
typedef GLintptr GLvdpauSurfaceNV;
typedef void (APIENTRY *GLVULKANPROCNV)(void);
#define GL_DEPTH_BUFFER_BIT 0x00000100
#define GL_STENCIL_BUFFER_BIT 0x00000400
#define GL_COLOR_BUFFER_BIT 0x00004000
#define GL_FALSE 0
#define GL_TRUE 1
#define GL_POINTS 0x0000
#define GL_LINES 0x0001
#define GL_LINE_LOOP 0x0002
#define GL_LINE_STRIP 0x0003
#define GL_TRIANGLES 0x0004
#define GL_TRIANGLE_STRIP 0x0005
#define GL_TRIANGLE_FAN 0x0006
#define GL_NEVER 0x0200
#define GL_LESS 0x0201
#define GL_EQUAL 0x0202
#define GL_LEQUAL 0x0203
#define GL_GREATER 0x0204
#define GL_NOTEQUAL 0x0205
#define GL_GEQUAL 0x0206
#define GL_ALWAYS 0x0207
#define GL_ZERO 0
#define GL_ONE 1
#define GL_SRC_COLOR 0x0300
#define GL_ONE_MINUS_SRC_COLOR 0x0301
#define GL_SRC_ALPHA 0x0302
#define GL_ONE_MINUS_SRC_ALPHA 0x0303
#define GL_DST_ALPHA 0x0304
#define GL_ONE_MINUS_DST_ALPHA 0x0305
#define GL_DST_COLOR 0x0306
#define GL_ONE_MINUS_DST_COLOR 0x0307
#define GL_SRC_ALPHA_SATURATE 0x0308
#define GL_NONE 0
#define GL_FRONT_LEFT 0x0400
#define GL_FRONT_RIGHT 0x0401
#define GL_BACK_LEFT 0x0402
#define GL_BACK_RIGHT 0x0403
#define GL_FRONT 0x0404
#define GL_BACK 0x0405
#define GL_LEFT 0x0406
#define GL_RIGHT 0x0407
#define GL_FRONT_AND_BACK 0x0408
#define GL_NO_ERROR 0
#define GL_INVALID_ENUM 0x0500
#define GL_INVALID_VALUE 0x0501
#define GL_INVALID_OPERATION 0x0502
#define GL_OUT_OF_MEMORY 0x0505
#define GL_CW 0x0900
#define GL_CCW 0x0901
#define GL_POINT_SIZE 0x0B11
#define GL_POINT_SIZE_RANGE 0x0B12
#define GL_POINT_SIZE_GRANULARITY 0x0B13
#define GL_LINE_SMOOTH 0x0B20
#define GL_LINE_WIDTH 0x0B21
#define GL_LINE_WIDTH_RANGE 0x0B22
```

```
#define GL_LINE_WIDTH_GRANULARITY 0x0B23
#define GL_POLYGON_MODE 0x0B40
#define GL_POLYGON_SMOOTH 0x0B41
#define GL_CULL_FACE 0x0B44
#define GL_CULL_FACE_MODE 0x0B45
#define GL_FRONT_FACE 0x0B46
#define GL_DEPTH_RANGE 0x0B70
#define GL_DEPTH_TEST 0x0B71
#define GL_DEPTH_WRITEMASK 0x0B72
#define GL_DEPTH_CLEAR_VALUE 0x0B73
#define GL_DEPTH_FUNC 0x0B74
#define GL_STENCIL_TEST 0x0B90
#define GL_STENCIL_CLEAR_VALUE 0x0B91
#define GL_STENCIL_FUNC 0x0B92
#define GL_STENCIL_VALUE_MASK 0x0B93
#define GL_STENCIL_FAIL 0x0B94
#define GL_STENCIL_PASS_DEPTH_FAIL 0x0B95
#define GL_STENCIL_PASS_DEPTH_PASS 0x0B96
#define GL_STENCIL_REF 0x0B97
#define GL_STENCIL_WRITEMASK 0x0B98
#define GL_VIEWPORT 0x0BA2
#define GL_DITHER 0x0BD0
#define GL_BLEND_DST 0x0BE0
#define GL_BLEND_SRC 0x0BE1
#define GL_BLEND 0x0BE2
#define GL_LOGIC_OP_MODE 0x0BF0
#define GL_DRAW_BUFFER 0x0C01
#define GL_READ_BUFFER 0x0C02
#define GL_SCISSOR_BOX 0x0C10
#define GL_SCISSOR_TEST 0x0C11
#define GL_COLOR_CLEAR_VALUE 0x0C22
#define GL_COLOR_WRITEMASK 0x0C23
#define GL_DOUBLEBUFFER 0x0C32
#define GL_STEREO 0x0C33
#define GL_LINE_SMOOTH_HINT 0x0C52
#define GL_POLYGON_SMOOTH_HINT 0x0C53
#define GL_UNPACK_SWAP_BYTES 0x0CF0
#define GL_UNPACK_LSB_FIRST 0x0CF1
#define GL_UNPACK_ROW_LENGTH 0x0CF2
#define GL_UNPACK_SKIP_ROWS 0x0CF3
#define GL_UNPACK_SKIP_PIXELS 0x0CF4
#define GL_UNPACK_ALIGNMENT 0x0CF5
#define GL_PACK_SWAP_BYTES 0x0D00
#define GL_PACK_LSB_FIRST 0x0D01
#define GL_PACK_ROW_LENGTH 0x0D02
#define GL_PACK_SKIP_ROWS 0x0D03
#define GL_PACK_SKIP_PIXELS 0x0D04
#define GL_PACK_ALIGNMENT 0x0D05
#define GL_MAX_TEXTURE_SIZE 0x0D33
#define GL_MAX_VIEWPORT_DIMS 0x0D3A
#define GL_SUBPIXEL_BITS 0x0D50
#define GL_TEXTURE_1D 0x0DE0
#define GL_TEXTURE_2D 0x0DE1
#define GL_TEXTURE_WIDTH 0x1000
#define GL_TEXTURE_HEIGHT 0x1001
#define GL_TEXTURE_BORDER_COLOR 0x1004
#define GL_DONT_CARE 0x1100
#define GL_FASTEST 0x1101
#define GL_NICEST 0x1102
#define GL_BYTE 0x1400
#define GL_UNSIGNED_BYTE 0x1401
#define GL_SHORT 0x1402
#define GL_UNSIGNED_SHORT 0x1403
#define GL_INT 0x1404
#define GL_UNSIGNED_INT 0x1405
#define GL_FLOAT 0x1406
#define GL_CLEAR 0x1500
#define GL_AND 0x1501
#define GL_AND_REVERSE 0x1502
#define GL_COPY 0x1503
#define GL_AND_INVERTED 0x1504
#define GL_NOOP 0x1505
#define GL_XOR 0x1506
#define GL_OR 0x1507
#define GL_NOR 0x1508
#define GL_EQUIV 0x1509
#define GL_INVERT 0x150A
#define GL_OR_REVERSE 0x150B
#define GL_COPY_INVERTED 0x150C
#define GL_OR_INVERTED 0x150D
#define GL_NAND 0x150E
#define GL_SET 0x150F
#define GL_TEXTURE 0x1702
#define GL_COLOR 0x1800
#define GL_DEPTH 0x1801
#define GL_STENCIL 0x1802
#define GL_STENCIL_INDEX 0x1901
```

```
#define GL_DEPTH_COMPONENT 0x1902
#define GL_RED 0x1903
#define GL_GREEN 0x1904
#define GL_BLUE 0x1905
#define GL_ALPHA 0x1906
#define GL_RGB 0x1907
#define GL_RGBA 0x1908
#define GL_POINT 0x1B00
#define GL_LINE 0x1B01
#define GL_FILL 0x1B02
#define GL_KEEP 0x1E00
#define GL_REPLACE 0x1E01
#define GL_INCR 0x1E02
#define GL_DECR 0x1E03
#define GL_VENDOR 0x1F00
#define GL_RENDERER 0x1F01
#define GL_VERSION 0x1F02
#define GL_EXTENSIONS 0x1F03
#define GL_NEAREST 0x2600
#define GL_LINEAR 0x2601
#define GL_NEAREST_MIPMAP_NEAREST 0x2700
#define GL_LINEAR_MIPMAP_NEAREST 0x2701
#define GL_NEAREST_MIPMAP_LINEAR 0x2702
#define GL_LINEAR_MIPMAP_LINEAR 0x2703
#define GL_TEXTURE_MAG_FILTER 0x2800
#define GL_TEXTURE_MIN_FILTER 0x2801
#define GL_TEXTURE_WRAP_S 0x2802
#define GL_TEXTURE_WRAP_T 0x2803
#define GL_REPEAT 0x2901
#define GL_COLOR_LOGIC_OP 0x0BF2
#define GL_POLYGON_OFFSET_UNITS 0x2A00
#define GL_POLYGON_OFFSET_POINT 0x2A01
#define GL_POLYGON_OFFSET_LINE 0x2A02
#define GL_POLYGON_OFFSET_FILL 0x8037
#define GL_POLYGON_OFFSET_FACTOR 0x8038
#define GL_TEXTURE_BINDING_1D 0x8068
#define GL_TEXTURE_BINDING_2D 0x8069
#define GL_TEXTURE_INTERNAL_FORMAT 0x1003
#define GL_TEXTURE_RED_SIZE 0x805C
#define GL_TEXTURE_GREEN_SIZE 0x805D
#define GL_TEXTURE_BLUE_SIZE 0x805E
#define GL_TEXTURE_ALPHA_SIZE 0x805F
#define GL_DOUBLE 0x140A
#define GL_PROXY_TEXTURE_1D 0x8063
#define GL_PROXY_TEXTURE_2D 0x8064
#define GL_R3_G3_B2 0x2A10
#define GL_RGB4 0x804F
#define GL_RGB5 0x8050
#define GL_RGB8 0x8051
#define GL_RGB10 0x8052
#define GL_RGB12 0x8053
#define GL_RGB16 0x8054
#define GL_RGBA2 0x8055
#define GL_RGBA4 0x8056
#define GL_RGB5_A1 0x8057
#define GL_RGBA8 0x8058
#define GL_RGB10_A2 0x8059
#define GL_RGBA12 0x805A
#define GL_RGBA16 0x805B
#define GL_UNSIGNED_BYTE_3_3_2 0x8032
#define GL_UNSIGNED_SHORT_4_4_4_4 0x8033
#define GL_UNSIGNED_SHORT_5_5_5_1 0x8034
#define GL_UNSIGNED_INT_8_8_8_8 0x8035
#define GL_UNSIGNED_INT_10_10_10_2 0x8036
#define GL_TEXTURE_BINDING_3D 0x806A
#define GL_PACK_SKIP_IMAGES 0x806B
#define GL_PACK_IMAGE_HEIGHT 0x806C
#define GL_UNPACK_SKIP_IMAGES 0x806D
#define GL_UNPACK_IMAGE_HEIGHT 0x806E
#define GL_TEXTURE_3D 0x806F
#define GL_PROXY_TEXTURE_3D 0x8070
#define GL_TEXTURE_DEPTH 0x8071
#define GL_TEXTURE_WRAP_R 0x8072
#define GL_MAX_3D_TEXTURE_SIZE 0x8073
#define GL_UNSIGNED_BYTE_2_3_3_REV 0x8362
#define GL_UNSIGNED_SHORT_5_6_5 0x8363
#define GL_UNSIGNED_SHORT_5_6_5_REV 0x8364
#define GL_UNSIGNED_SHORT_4_4_4_4_REV 0x8365
#define GL_UNSIGNED_SHORT_1_5_5_5_REV 0x8366
#define GL_UNSIGNED_INT_8_8_8_8_REV 0x8367
#define GL_UNSIGNED_INT_2_10_10_10_REV 0x8368
#define GL_BGR 0x80E0
#define GL_BGRA 0x80E1
#define GL_MAX_ELEMENTS_VERTICES 0x80E8
#define GL_MAX_ELEMENTS_INDICES 0x80E9
#define GL_CLAMP_TO_EDGE 0x812F
#define GL_TEXTURE_MIN_LOD 0x813A
```

```
#define GL_TEXTURE_MAX_LOD 0x813B
#define GL_TEXTURE_BASE_LEVEL 0x813C
#define GL_TEXTURE_MAX_LEVEL 0x813D
#define GL_SMOOTH_POINT_SIZE_RANGE 0x0B12
#define GL_SMOOTH_POINT_SIZE_GRANULARITY 0x0B13
#define GL_SMOOTH_LINE_WIDTH_RANGE 0x0B22
#define GL_SMOOTH_LINE_WIDTH_GRANULARITY 0x0B23
#define GL_ALIASED_LINE_WIDTH_RANGE 0x846E
#define GL_TEXTURE0 0x84C0
#define GL_TEXTURE1 0x84C1
#define GL_TEXTURE2 0x84C2
#define GL_TEXTURE3 0x84C3
#define GL_TEXTURE4 0x84C4
#define GL_TEXTURE5 0x84C5
#define GL_TEXTURE6 0x84C6
#define GL_TEXTURE7 0x84C7
#define GL_TEXTURE8 0x84C8
#define GL_TEXTURE9 0x84C9
#define GL_TEXTURE10 0x84CA
#define GL_TEXTURE11 0x84CB
#define GL_TEXTURE12 0x84CC
#define GL_TEXTURE13 0x84CD
#define GL_TEXTURE14 0x84CE
#define GL_TEXTURE15 0x84CF
#define GL_TEXTURE16 0x84D0
#define GL_TEXTURE17 0x84D1
#define GL_TEXTURE18 0x84D2
#define GL_TEXTURE19 0x84D3
#define GL_TEXTURE20 0x84D4
#define GL_TEXTURE21 0x84D5
#define GL_TEXTURE22 0x84D6
#define GL_TEXTURE23 0x84D7
#define GL_TEXTURE24 0x84D8
#define GL_TEXTURE25 0x84D9
#define GL_TEXTURE26 0x84DA
#define GL_TEXTURE27 0x84DB
#define GL_TEXTURE28 0x84DC
#define GL_TEXTURE29 0x84DD
#define GL_TEXTURE30 0x84DE
#define GL_TEXTURE31 0x84DF
#define GL_ACTIVE_TEXTURE 0x84E0
#define GL_MULTISAMPLE 0x809D
#define GL_SAMPLE_ALPHA_TO_COVERAGE 0x809E
#define GL_SAMPLE_ALPHA_TO_ONE 0x809F
#define GL_SAMPLE_COVERAGE 0x80A0
#define GL_SAMPLE_BUFFERS 0x80A8
#define GL_SAMPLES 0x80A9
#define GL_SAMPLE_COVERAGE_VALUE 0x80AA
#define GL_SAMPLE_COVERAGE_INVERT 0x80AB
#define GL_TEXTURE_CUBE_MAP 0x8513
#define GL_TEXTURE_BINDING_CUBE_MAP 0x8514
#define GL_TEXTURE_CUBE_MAP_POSITIVE_X 0x8515
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_X 0x8516
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Y 0x8517
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y 0x8518
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Z 0x8519
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z 0x851A
#define GL_PROXY_TEXTURE_CUBE_MAP 0x851B
#define GL_MAX_CUBE_MAP_TEXTURE_SIZE 0x851C
#define GL_COMPRESSED_RGB 0x84ED
#define GL_COMPRESSED_RGBA 0x84EE
#define GL_TEXTURE_COMPRESSION_HINT 0x84EF
#define GL_TEXTURE_COMPRESSED_IMAGE_SIZE 0x86A0
#define GL_TEXTURE_COMPRESSED 0x86A1
#define GL_NUM_COMPRESSED_TEXTURE_FORMATS 0x86A2
#define GL_COMPRESSED_TEXTURE_FORMATS 0x86A3
#define GL_CLAMP_TO_BORDER 0x812D
#define GL_BLEND_DST_RGB 0x80C8
#define GL_BLEND_SRC_RGB 0x80C9
#define GL_BLEND_DST_ALPHA 0x80CA
#define GL_BLEND_SRC_ALPHA 0x80CB
#define GL_POINT_FADE_THRESHOLD_SIZE 0x8128
#define GL_DEPTH_COMPONENT16 0x81A5
#define GL_DEPTH_COMPONENT24 0x81A6
#define GL_DEPTH_COMPONENT32 0x81A7
#define GL_MIRRORED_REPEAT 0x8370
#define GL_MAX_TEXTURE_LOD_BIAS 0x84FD
#define GL_TEXTURE_LOD_BIAS 0x8501
#define GL_INCR_WRAP 0x8507
#define GL_DECR_WRAP 0x8508
#define GL_TEXTURE_DEPTH_SIZE 0x884A
#define GL_TEXTURE_COMPARE_MODE 0x884C
#define GL_TEXTURE_COMPARE_FUNC 0x884D
#define GL_BLEND_COLOR 0x8005
#define GL_BLEND_EQUATION 0x8009
#define GL_CONSTANT_COLOR 0x8001
#define GL_ONE_MINUS_CONSTANT_COLOR 0x8002
```

```
#define GL_CONSTANT_ALPHA 0x8003
#define GL_ONE_MINUS_CONSTANT_ALPHA 0x8004
#define GL_FUNC_ADD 0x8006
#define GL_FUNC_REVERSE_SUBTRACT 0x800B
#define GL_FUNC_SUBTRACT 0x800A
#define GL_MIN 0x8007
#define GL_MAX 0x8008
#define GL_BUFFER_SIZE 0x8764
#define GL_BUFFER_USAGE 0x8765
#define GL_QUERY_COUNTER_BITS 0x8864
#define GL_CURRENT_QUERY 0x8865
#define GL_QUERY_RESULT 0x8866
#define GL_QUERY_RESULT_AVAILABLE 0x8867
#define GL_ARRAY_BUFFER 0x8892
#define GL_ELEMENT_ARRAY_BUFFER 0x8893
#define GL_ARRAY_BUFFER_BINDING 0x8894
#define GL_ELEMENT_ARRAY_BUFFER_BINDING 0x8895
#define GL_VERTEX_ATTRIB_ARRAY_BUFFER_BINDING 0x889F
#define GL_READ_ONLY 0x88B8
#define GL_WRITE_ONLY 0x88B9
#define GL_READ_WRITE 0x88BA
#define GL_BUFFER_ACCESS 0x88BB
#define GL_BUFFER_MAPPED 0x88BC
#define GL_BUFFER_MAP_POINTER 0x88BD
#define GL_STREAM_DRAW 0x88E0
#define GL_STREAM_READ 0x88E1
#define GL_STREAM_COPY 0x88E2
#define GL_STATIC_DRAW 0x88E4
#define GL_STATIC_READ 0x88E5
#define GL_STATIC_COPY 0x88E6
#define GL_DYNAMIC_DRAW 0x88E8
#define GL_DYNAMIC_READ 0x88E9
#define GL_DYNAMIC_COPY 0x88EA
#define GL_SAMPLES_PASSED 0x8914
#define GL_SRC1_ALPHA 0x8589
#define GL_BLEND_EQUATION_RGB 0x8009
#define GL_VERTEX_ATTRIB_ARRAY_ENABLED 0x8622
#define GL_VERTEX_ATTRIB_ARRAY_SIZE 0x8623
#define GL_VERTEX_ATTRIB_ARRAY_STRIDE 0x8624
#define GL_VERTEX_ATTRIB_ARRAY_TYPE 0x8625
#define GL_CURRENT_VERTEX_ATTRIB 0x8626
#define GL_VERTEX_PROGRAM_POINT_SIZE 0x8642
#define GL_VERTEX_ATTRIB_ARRAY_POINTER 0x8645
#define GL_STENCIL_BACK_FUNC 0x8800
#define GL_STENCIL_BACK_FAIL 0x8801
#define GL_STENCIL_BACK_PASS_DEPTH_FAIL 0x8802
#define GL_STENCIL_BACK_PASS_DEPTH_PASS 0x8803
#define GL_MAX_DRAW_BUFFERS 0x8824
#define GL_DRAW_BUFFER0 0x8825
#define GL_DRAW_BUFFER1 0x8826
#define GL_DRAW_BUFFER2 0x8827
#define GL_DRAW_BUFFER3 0x8828
#define GL_DRAW_BUFFER4 0x8829
#define GL_DRAW_BUFFER5 0x882A
#define GL_DRAW_BUFFER6 0x882B
#define GL_DRAW_BUFFER7 0x882C
#define GL_DRAW_BUFFER8 0x882D
#define GL_DRAW_BUFFER9 0x882E
#define GL_DRAW_BUFFER10 0x882F
#define GL_DRAW_BUFFER11 0x8830
#define GL_DRAW_BUFFER12 0x8831
#define GL_DRAW_BUFFER13 0x8832
#define GL_DRAW_BUFFER14 0x8833
#define GL_DRAW_BUFFER15 0x8834
#define GL_BLEND_EQUATION_ALPHA 0x883D
#define GL_MAX_VERTEX_ATTRIBS 0x8869
#define GL_VERTEX_ATTRIB_ARRAY_NORMALIZED 0x886A
#define GL_MAX_TEXTURE_IMAGE_UNITS 0x8872
#define GL_FRAGMENT_SHADER 0x8B30
#define GL_VERTEX_SHADER 0x8B31
#define GL_MAX_FRAGMENT_UNIFORM_COMPONENTS 0x8B49
#define GL_MAX_VERTEX_UNIFORM_COMPONENTS 0x8B4A
#define GL_MAX_VARYING_FLOATS 0x8B4B
#define GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS 0x8B4C
#define GL_MAX_COMBINED_TEXTURE_IMAGE_UNITS 0x8B4D
#define GL_SHADER_TYPE 0x8B4F
#define GL_FLOAT_VEC2 0x8B50
#define GL_FLOAT_VEC3 0x8B51
#define GL_FLOAT_VEC4 0x8B52
#define GL_INT_VEC2 0x8B53
#define GL_INT_VEC3 0x8B54
#define GL_INT_VEC4 0x8B55
#define GL_BOOL 0x8B56
#define GL_BOOL_VEC2 0x8B57
#define GL_BOOL_VEC3 0x8B58
#define GL_BOOL_VEC4 0x8B59
#define GL_FLOAT_MAT2 0x8B5A
```

```
#define GL_FLOAT_MAT3 0x8B5B
#define GL_FLOAT_MAT4 0x8B5C
#define GL_SAMPLER_1D 0x8B5D
#define GL_SAMPLER_2D 0x8B5E
#define GL_SAMPLER_3D 0x8B5F
#define GL_SAMPLER_CUBE 0x8B60
#define GL_SAMPLER_1D_SHADOW 0x8B61
#define GL_SAMPLER_2D_SHADOW 0x8B62
#define GL_DELETE_STATUS 0x8B80
#define GL_COMPILE_STATUS 0x8B81
#define GL_LINK_STATUS 0x8B82
#define GL_VALIDATE_STATUS 0x8B83
#define GL_INFO_LOG_LENGTH 0x8B84
#define GL_ATTACHED_SHADERS 0x8B85
#define GL_ACTIVE_UNIFORMS 0x8B86
#define GL_ACTIVE_UNIFORM_MAX_LENGTH 0x8B87
#define GL_SHADER_SOURCE_LENGTH 0x8B88
#define GL_ACTIVE_ATTRIBUTES 0x8B89
#define GL_ACTIVE_ATTRIBUTE_MAX_LENGTH 0x8B8A
#define GL_FRAGMENT_SHADER_DERIVATIVE_HINT 0x8B8B
#define GL_SHADING_LANGUAGE_VERSION 0x8B8C
#define GL_CURRENT_PROGRAM 0x8B8D
#define GL_POINT_SPRITE_COORD_ORIGIN 0x8CA0
#define GL_LOWER_LEFT 0x8CA1
#define GL_UPPER_LEFT 0x8CA2
#define GL_STENCIL_BACK_REF 0x8CA3
#define GL_STENCIL_BACK_VALUE_MASK 0x8CA4
#define GL_STENCIL_BACK_WRITEMASK 0x8CA5
#define GL_PIXEL_PACK_BUFFER 0x88EB
#define GL_PIXEL_UNPACK_BUFFER 0x88EC
#define GL_PIXEL_PACK_BUFFER_BINDING 0x88ED
#define GL_PIXEL_UNPACK_BUFFER_BINDING 0x88EF
#define GL_FLOAT_MAT2x3 0x8B65
#define GL_FLOAT_MAT2x4 0x8B66
#define GL_FLOAT_MAT3x2 0x8B67
#define GL_FLOAT_MAT3x4 0x8B68
#define GL_FLOAT_MAT4x2 0x8B69
#define GL_FLOAT_MAT4x3 0x8B6A
#define GL_SRGB 0x8C40
#define GL_SRGB8 0x8C41
#define GL_SRGB_ALPHA 0x8C42
#define GL_SRGB8_ALPHA8 0x8C43
#define GL_COMPRESSED_SRGB 0x8C48
#define GL_COMPRESSED_SRGB_ALPHA 0x8C49
#define GL_COMPARE_REF_TO_TEXTURE 0x884E
#define GL_CLIP_DISTANCE0 0x3000
#define GL_CLIP_DISTANCE1 0x3001
#define GL_CLIP_DISTANCE2 0x3002
#define GL_CLIP_DISTANCE3 0x3003
#define GL_CLIP_DISTANCE4 0x3004
#define GL_CLIP_DISTANCE5 0x3005
#define GL_CLIP_DISTANCE6 0x3006
#define GL_CLIP_DISTANCE7 0x3007
#define GL_MAX_CLIP_DISTANCES 0x0D32
#define GL_MAJOR_VERSION 0x821B
#define GL_MINOR_VERSION 0x821C
#define GL_NUM_EXTENSIONS 0x821D
#define GL_CONTEXT_FLAGS 0x821E
#define GL_COMPRESSED_RED 0x8225
#define GL_COMPRESSED_RG 0x8226
#define GL_CONTEXT_FLAG_FORWARD_COMPATIBLE_BIT 0x00000001
#define GL_RGBA32F 0x8814
#define GL_RGB32F 0x8815
#define GL_RGBA16F 0x881A
#define GL_RGB16F 0x881B
#define GL_VERTEX_ATTRIB_ARRAY_INTEGER 0x88FD
#define GL_MAX_ARRAY_TEXTURE_LAYERS 0x88FF
#define GL_MIN_PROGRAM_TEXEL_OFFSET 0x8904
#define GL_MAX_PROGRAM_TEXEL_OFFSET 0x8905
#define GL_CLAMP_READ_COLOR 0x891C
#define GL_FIXED_ONLY 0x891D
#define GL_MAX_VARYING_COMPONENTS 0x8B4B
#define GL_TEXTURE_1D_ARRAY 0x8C18
#define GL_PROXY_TEXTURE_1D_ARRAY 0x8C19
#define GL_TEXTURE_2D_ARRAY 0x8C1A
#define GL_PROXY_TEXTURE_2D_ARRAY 0x8C1B
#define GL_TEXTURE_BINDING_1D_ARRAY 0x8C1C
#define GL_TEXTURE_BINDING_2D_ARRAY 0x8C1D
#define GL_R11F_G11F_B10F 0x8C3A
#define GL_UNSIGNED_INT_10F_11F_11F_REV 0x8C3B
#define GL_RGB9_E5 0x8C3D
#define GL_UNSIGNED_INT_5_9_9_9_REV 0x8C3E
#define GL_TEXTURE_SHARED_SIZE 0x8C3F
#define GL_TRANSFORM_FEEDBACK_VARYING_MAX_LENGTH 0x8C76
#define GL_TRANSFORM_FEEDBACK_BUFFER_MODE 0x8C7F
#define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_COMPONENTS 0x8C80
#define GL_TRANSFORM_FEEDBACK_VARYINGS 0x8C83
```

```
#define GL_TRANSFORM_FEEDBACK_BUFFER_START 0x8C84
#define GL_TRANSFORM_FEEDBACK_BUFFER_SIZE 0x8C85
#define GL_PRIMITIVES_GENERATED 0x8C87
#define GL_TRANSFORM_FEEDBACK_PRIMITIVES_WRITTEN 0x8C88
#define GL_RASTERIZER_DISCARD 0x8C89
#define GL_MAX_TRANSFORM_FEEDBACK_INTERLEAVED_COMPONENTS 0x8C8A
#define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_ATTRIBS 0x8C8B
#define GL_INTERLEAVED_ATTRIBS 0x8C8C
#define GL_SEPARATE_ATTRIBS 0x8C8D
#define GL_TRANSFORM_FEEDBACK_BUFFER 0x8C8E
#define GL_TRANSFORM_FEEDBACK_BUFFER_BINDING 0x8C8F
#define GL_RGBA32UI 0x8D70
#define GL_RGB32UI 0x8D71
#define GL_RGBA16UI 0x8D76
#define GL_RGB16UI 0x8D77
#define GL_RGBA8UI 0x8D7C
#define GL_RGB8UI 0x8D7D
#define GL_RGBA32I 0x8D82
#define GL_RGB32I 0x8D83
#define GL_RGBA16I 0x8D88
#define GL_RGB16I 0x8D89
#define GL_RGBA8I 0x8D8E
#define GL_RGB8I 0x8D8F
#define GL_RED_INTEGER 0x8D94
#define GL_GREEN_INTEGER 0x8D95
#define GL_BLUE_INTEGER 0x8D96
#define GL_RGB_INTEGER 0x8D98
#define GL_RGBA_INTEGER 0x8D99
#define GL_BGR_INTEGER 0x8D9A
#define GL_BGRA_INTEGER 0x8D9B
#define GL_SAMPLER_1D_ARRAY 0x8DC0
#define GL_SAMPLER_2D_ARRAY 0x8DC1
#define GL_SAMPLER_1D_ARRAY_SHADOW 0x8DC3
#define GL_SAMPLER_2D_ARRAY_SHADOW 0x8DC4
#define GL_SAMPLER_CUBE_SHADOW 0x8DC5
#define GL_UNSIGNED_INT_VEC2 0x8DC6
#define GL_UNSIGNED_INT_VEC3 0x8DC7
#define GL_UNSIGNED_INT_VEC4 0x8DC8
#define GL_INT_SAMPLER_1D 0x8DC9
#define GL_INT_SAMPLER_2D 0x8DCA
#define GL_INT_SAMPLER_3D 0x8DCB
#define GL_INT_SAMPLER_CUBE 0x8DCC
#define GL_INT_SAMPLER_1D_ARRAY 0x8DCE
#define GL_INT_SAMPLER_2D_ARRAY 0x8DCF
#define GL_UNSIGNED_INT_SAMPLER_1D 0x8DD1
#define GL_UNSIGNED_INT_SAMPLER_2D 0x8DD2
#define GL_UNSIGNED_INT_SAMPLER_3D 0x8DD3
#define GL_UNSIGNED_INT_SAMPLER_CUBE 0x8DD4
#define GL_UNSIGNED_INT_SAMPLER_1D_ARRAY 0x8DD6
#define GL_UNSIGNED_INT_SAMPLER_2D_ARRAY 0x8DD7
#define GL_QUERY_WAIT 0x8E13
#define GL_QUERY_NO_WAIT 0x8E14
#define GL_QUERY_BY_REGION_WAIT 0x8E15
#define GL_QUERY_BY_REGION_NO_WAIT 0x8E16
#define GL_BUFFER_ACCESS_FLAGS 0x911F
#define GL_BUFFER_MAP_LENGTH 0x9120
#define GL_BUFFER_MAP_OFFSET 0x9121
#define GL_DEPTH_COMPONENT32F 0x8CAC
#define GL_DEPTH32F_STENCIL8 0x8CAD
#define GL_FLOAT_32_UNSIGNED_INT_24_8_REV 0x8DAD
#define GL_INVALID_FRAMEBUFFER_OPERATION 0x0506
#define GL_FRAMEBUFFER_ATTACHMENT_COLOR_ENCODING 0x8210
#define GL_FRAMEBUFFER_ATTACHMENT_COMPONENT_TYPE 0x8211
#define GL_FRAMEBUFFER_ATTACHMENT_RED_SIZE 0x8212
#define GL_FRAMEBUFFER_ATTACHMENT_GREEN_SIZE 0x8213
#define GL_FRAMEBUFFER_ATTACHMENT_BLUE_SIZE 0x8214
#define GL_FRAMEBUFFER_ATTACHMENT_ALPHA_SIZE 0x8215
#define GL_FRAMEBUFFER_ATTACHMENT_DEPTH_SIZE 0x8216
#define GL_FRAMEBUFFER_ATTACHMENT_STENCIL_SIZE 0x8217
#define GL_FRAMEBUFFER_DEFAULT 0x8218
#define GL_FRAMEBUFFER_UNDEFINED 0x8219
#define GL_DEPTH_STENCIL_ATTACHMENT 0x821A
#define GL_MAX_RENDERBUFFER_SIZE 0x84E8
#define GL_DEPTH_STENCIL 0x84F9
#define GL_UNSIGNED_INT_24_8 0x84FA
#define GL_DEPTH24_STENCIL8 0x88F0
#define GL_TEXTURE_STENCIL_SIZE 0x88F1
#define GL_TEXTURE_RED_TYPE 0x8C10
#define GL_TEXTURE_GREEN_TYPE 0x8C11
#define GL_TEXTURE_BLUE_TYPE 0x8C12
#define GL_TEXTURE_ALPHA_TYPE 0x8C13
#define GL_TEXTURE_DEPTH_TYPE 0x8C16
#define GL_UNSIGNED_NORMALIZED 0x8C17
#define GL_FRAMEBUFFER_BINDING 0x8CA6
#define GL_DRAW_FRAMEBUFFER_BINDING 0x8CA6
#define GL_RENDERBUFFER_BINDING 0x8CA7
#define GL_READ_FRAMEBUFFER 0x8CA8
```

```
#define GL_DRAW_FRAMEBUFFER 0x8CA9
#define GL_READ_FRAMEBUFFER_BINDING 0x8CAA
#define GL_RENDERBUFFER_SAMPLES 0x8CAB
#define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE 0x8CD0
#define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_NAME 0x8CD1
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL 0x8CD2
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE 0x8CD3
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LAYER 0x8CD4
#define GL_FRAMEBUFFER_COMPLETE 0x8CD5
#define GL_FRAMEBUFFER_INCOMPLETE_ATTACHMENT 0x8CD6
#define GL_FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT 0x8CD7
#define GL_FRAMEBUFFER_INCOMPLETE_DRAW_BUFFER 0x8CDB
#define GL_FRAMEBUFFER_INCOMPLETE_READ_BUFFER 0x8CDC
#define GL_FRAMEBUFFER_UNSUPPORTED 0x8CDD
#define GL_MAX_COLOR_ATTACHMENTS 0x8CDF
#define GL_COLOR_ATTACHMENT0 0x8CE0
#define GL_COLOR_ATTACHMENT1 0x8CE1
#define GL_COLOR_ATTACHMENT2 0x8CE2
#define GL_COLOR_ATTACHMENT3 0x8CE3
#define GL_COLOR_ATTACHMENT4 0x8CE4
#define GL_COLOR_ATTACHMENT5 0x8CE5
#define GL_COLOR_ATTACHMENT6 0x8CE6
#define GL_COLOR_ATTACHMENT7 0x8CE7
#define GL_COLOR_ATTACHMENT8 0x8CE8
#define GL_COLOR_ATTACHMENT9 0x8CE9
#define GL_COLOR_ATTACHMENT10 0x8CEA
#define GL_COLOR_ATTACHMENT11 0x8CEB
#define GL_COLOR_ATTACHMENT12 0x8CEC
#define GL_COLOR_ATTACHMENT13 0x8CED
#define GL_COLOR_ATTACHMENT14 0x8CEE
#define GL_COLOR_ATTACHMENT15 0x8CEF
#define GL_COLOR_ATTACHMENT16 0x8CF0
#define GL_COLOR_ATTACHMENT17 0x8CF1
#define GL_COLOR_ATTACHMENT18 0x8CF2
#define GL_COLOR_ATTACHMENT19 0x8CF3
#define GL_COLOR_ATTACHMENT20 0x8CF4
#define GL_COLOR_ATTACHMENT21 0x8CF5
#define GL_COLOR_ATTACHMENT22 0x8CF6
#define GL_COLOR_ATTACHMENT23 0x8CF7
#define GL_COLOR_ATTACHMENT24 0x8CF8
#define GL_COLOR_ATTACHMENT25 0x8CF9
#define GL_COLOR_ATTACHMENT26 0x8CFA
#define GL_COLOR_ATTACHMENT27 0x8CFB
#define GL_COLOR_ATTACHMENT28 0x8CFC
#define GL_COLOR_ATTACHMENT29 0x8CFD
#define GL_COLOR_ATTACHMENT30 0x8CFE
#define GL_COLOR_ATTACHMENT31 0x8CFF
#define GL_DEPTH_ATTACHMENT 0x8D00
#define GL_STENCIL_ATTACHMENT 0x8D20
#define GL_FRAMEBUFFER 0x8D40
#define GL_RENDERBUFFER 0x8D41
#define GL_RENDERBUFFER_WIDTH 0x8D42
#define GL_RENDERBUFFER_HEIGHT 0x8D43
#define GL_RENDERBUFFER_INTERNAL_FORMAT 0x8D44
#define GL_STENCIL_INDEX1 0x8D46
#define GL_STENCIL_INDEX4 0x8D47
#define GL_STENCIL_INDEX8 0x8D48
#define GL_STENCIL_INDEX16 0x8D49
#define GL_RENDERBUFFER_RED_SIZE 0x8D50
#define GL_RENDERBUFFER_GREEN_SIZE 0x8D51
#define GL_RENDERBUFFER_BLUE_SIZE 0x8D52
#define GL_RENDERBUFFER_ALPHA_SIZE 0x8D53
#define GL_RENDERBUFFER_DEPTH_SIZE 0x8D54
#define GL_RENDERBUFFER_STENCIL_SIZE 0x8D55
#define GL_FRAMEBUFFER_INCOMPLETE_MULTISAMPLE 0x8D56
#define GL_MAX_SAMPLES 0x8D57
#define GL_FRAMEBUFFER_SRGB 0x8DB9
#define GL_HALF_FLOAT 0x140B
#define GL_MAP_READ_BIT 0x0001
#define GL_MAP_WRITE_BIT 0x0002
#define GL_MAP_INVALIDATE_RANGE_BIT 0x0004
#define GL_MAP_INVALIDATE_BUFFER_BIT 0x0008
#define GL_MAP_FLUSH_EXPLICIT_BIT 0x0010
#define GL_MAP_UNSYNCHRONIZED_BIT 0x0020
#define GL_COMPRESSED_RED_RGTC1 0x8DBB
#define GL_COMPRESSED_SIGNED_RED_RGTC1 0x8DBC
#define GL_COMPRESSED_RG_RGTC2 0x8DBD
#define GL_COMPRESSED_SIGNED_RG_RGTC2 0x8DBE
#define GL_RG 0x8227
#define GL_RG_INTEGER 0x8228
#define GL_R8 0x8229
#define GL_R16 0x822A
#define GL_RG8 0x822B
#define GL_RG16 0x822C
#define GL_R16F 0x822D
#define GL_R32F 0x822E
#define GL_RG16F 0x822F
```

```
#define GL_RG32F 0x8230
#define GL_R8I 0x8231
#define GL_R8UI 0x8232
#define GL_R16I 0x8233
#define GL_R16UI 0x8234
#define GL_R32I 0x8235
#define GL_R32UI 0x8236
#define GL_RG8I 0x8237
#define GL_RG8UI 0x8238
#define GL_RG16I 0x8239
#define GL_RG16UI 0x823A
#define GL_RG32I 0x823B
#define GL_RG32UI 0x823C
#define GL_VERTEX_ARRAY_BINDING 0x85B5
#define GL_SAMPLER_2D_RECT 0x8B63
#define GL_SAMPLER_2D_RECT_SHADOW 0x8B64
#define GL_SAMPLER_BUFFER 0x8DC2
#define GL_INT_SAMPLER_2D_RECT 0x8DCD
#define GL_INT_SAMPLER_BUFFER 0x8DD0
#define GL_UNSIGNED_INT_SAMPLER_2D_RECT 0x8DD5
#define GL_UNSIGNED_INT_SAMPLER_BUFFER 0x8DD8
#define GL_TEXTURE_BUFFER 0x8C2A
#define GL_MAX_TEXTURE_BUFFER_SIZE 0x8C2B
#define GL_TEXTURE_BINDING_BUFFER 0x8C2C
#define GL_TEXTURE_BUFFER_DATA_STORE_BINDING 0x8C2D
#define GL_TEXTURE_RECTANGLE 0x84F5
#define GL_TEXTURE_BINDING_RECTANGLE 0x84F6
#define GL_PROXY_TEXTURE_RECTANGLE 0x84F7
#define GL_MAX_RECTANGLE_TEXTURE_SIZE 0x84F8
#define GL_R8_SNORM 0x8F94
#define GL_RG8_SNORM 0x8F95
#define GL_RGB8_SNORM 0x8F96
#define GL_RGBA8_SNORM 0x8F97
#define GL_R16_SNORM 0x8F98
#define GL_RG16_SNORM 0x8F99
#define GL_RGB16_SNORM 0x8F9A
#define GL_RGBA16_SNORM 0x8F9B
#define GL_SIGNED_NORMALIZED 0x8F9C
#define GL_PRIMITIVE_RESTART 0x8F9D
#define GL_PRIMITIVE_RESTART_INDEX 0x8F9E
#define GL_COPY_READ_BUFFER 0x8F36
#define GL_COPY_WRITE_BUFFER 0x8F37
#define GL_UNIFORM_BUFFER 0x8A11
#define GL_UNIFORM_BUFFER_BINDING 0x8A28
#define GL_UNIFORM_BUFFER_START 0x8A29
#define GL_UNIFORM_BUFFER_SIZE 0x8A2A
#define GL_MAX_VERTEX_UNIFORM_BLOCKS 0x8A2B
#define GL_MAX_GEOMETRY_UNIFORM_BLOCKS 0x8A2C
#define GL_MAX_FRAGMENT_UNIFORM_BLOCKS 0x8A2D
#define GL_MAX_COMBINED_UNIFORM_BLOCKS 0x8A2E
#define GL_MAX_UNIFORM_BUFFER_BINDINGS 0x8A2F
#define GL_MAX_UNIFORM_BLOCK_SIZE 0x8A30
#define GL_MAX_COMBINED_VERTEX_UNIFORM_COMPONENTS 0x8A31
#define GL_MAX_COMBINED_GEOMETRY_UNIFORM_COMPONENTS 0x8A32
#define GL_MAX_COMBINED_FRAGMENT_UNIFORM_COMPONENTS 0x8A33
#define GL_UNIFORM_BUFFER_OFFSET_ALIGNMENT 0x8A34
#define GL_ACTIVE_UNIFORM_BLOCK_MAX_NAME_LENGTH 0x8A35
#define GL_ACTIVE_UNIFORM_BLOCKS 0x8A36
#define GL_UNIFORM_TYPE 0x8A37
#define GL_UNIFORM_SIZE 0x8A38
#define GL_UNIFORM_NAME_LENGTH 0x8A39
#define GL_UNIFORM_BLOCK_INDEX 0x8A3A
#define GL_UNIFORM_OFFSET 0x8A3B
#define GL_UNIFORM_ARRAY_STRIDE 0x8A3C
#define GL_UNIFORM_MATRIX_STRIDE 0x8A3D
#define GL_UNIFORM_IS_ROW_MAJOR 0x8A3E
#define GL_UNIFORM_BLOCK_BINDING 0x8A3F
#define GL_UNIFORM_BLOCK_DATA_SIZE 0x8A40
#define GL_UNIFORM_BLOCK_NAME_LENGTH 0x8A41
#define GL_UNIFORM_BLOCK_ACTIVE_UNIFORMS 0x8A42
#define GL_UNIFORM_BLOCK_ACTIVE_UNIFORM_INDICES 0x8A43
#define GL_UNIFORM_BLOCK_REFERENCED_BY_VERTEX_SHADER 0x8A44
#define GL_UNIFORM_BLOCK_REFERENCED_BY_GEOMETRY_SHADER 0x8A45
#define GL_UNIFORM_BLOCK_REFERENCED_BY_FRAGMENT_SHADER 0x8A46
#define GL_INVALID_INDEX 0xFFFFFFFF
#define GL_CONTEXT_CORE_PROFILE_BIT 0x00000001
#define GL_CONTEXT_COMPATIBILITY_PROFILE_BIT 0x00000002
#define GL_LINES_ADJACENCY 0x000A
#define GL_LINE_STRIP_ADJACENCY 0x000B
#define GL_TRIANGLES_ADJACENCY 0x000C
#define GL_TRIANGLE_STRIP_ADJACENCY 0x000D
#define GL_PROGRAM_POINT_SIZE 0x8642
#define GL_MAX_GEOMETRY_TEXTURE_IMAGE_UNITS 0x8C29
#define GL_FRAMEBUFFER_ATTACHMENT_LAYERED 0x8DA7
#define GL_FRAMEBUFFER_INCOMPLETE_LAYER_TARGETS 0x8DA8
#define GL_GEOMETRY_SHADER 0x8DD9
#define GL_GEOMETRY_VERTICES_OUT 0x8916
```

```
#define GL_GEOMETRY_INPUT_TYPE 0x8917
#define GL_GEOMETRY_OUTPUT_TYPE 0x8918
#define GL_MAX_GEOMETRY_UNIFORM_COMPONENTS 0x8DDF
#define GL_MAX_GEOMETRY_OUTPUT_VERTICES 0x8DE0
#define GL_MAX_GEOMETRY_TOTAL_OUTPUT_COMPONENTS 0x8DE1
#define GL_MAX_VERTEX_OUTPUT_COMPONENTS 0x9122
#define GL_MAX_GEOMETRY_INPUT_COMPONENTS 0x9123
#define GL_MAX_GEOMETRY_OUTPUT_COMPONENTS 0x9124
#define GL_MAX_FRAGMENT_INPUT_COMPONENTS 0x9125
#define GL_CONTEXT_PROFILE_MASK 0x9126
#define GL_DEPTH_CLAMP 0x864F
#define GL_QUADS_FOLLOW_PROVOKING_VERTEX_CONVENTION 0x8E4C
#define GL_FIRST_VERTEX_CONVENTION 0x8E4D
#define GL_LAST_VERTEX_CONVENTION 0x8E4E
#define GL_PROVOKING_VERTEX 0x8E4F
#define GL_TEXTURE_CUBE_MAP_SEAMLESS 0x884F
#define GL_MAX_SERVER_WAIT_TIMEOUT 0x9111
#define GL_OBJECT_TYPE 0x9112
#define GL_SYNC_CONDITION 0x9113
#define GL_SYNC_STATUS 0x9114
#define GL_SYNC_FLAGS 0x9115
#define GL_SYNC_FENCE 0x9116
#define GL_SYNC_GPU_COMMANDS_COMPLETE 0x9117
#define GL_UNSIGNALED 0x9118
#define GL_SIGNALED 0x9119
#define GL_ALREADY_SIGNALED 0x911A
#define GL_TIMEOUT_EXPIRED 0x911B
#define GL_CONDITION_SATISFIED 0x911C
#define GL_WAIT_FAILED 0x911D
#define GL_TIMEOUT_IGNORED 0xFFFFFFFFFFFFFFFF
#define GL_SYNC_FLUSH_COMMANDS_BIT 0x00000001
#define GL_SAMPLE_POSITION 0x8E50
#define GL_SAMPLE_MASK 0x8E51
#define GL_SAMPLE_MASK_VALUE 0x8E52
#define GL_MAX_SAMPLE_MASK_WORDS 0x8E59
#define GL_TEXTURE_2D_MULTISAMPLE 0x9100
#define GL_PROXY_TEXTURE_2D_MULTISAMPLE 0x9101
#define GL_TEXTURE_2D_MULTISAMPLE_ARRAY 0x9102
#define GL_PROXY_TEXTURE_2D_MULTISAMPLE_ARRAY 0x9103
#define GL_TEXTURE_BINDING_2D_MULTISAMPLE 0x9104
#define GL_TEXTURE_BINDING_2D_MULTISAMPLE_ARRAY 0x9105
#define GL_TEXTURE_SAMPLES 0x9106
#define GL_TEXTURE_FIXED_SAMPLE_LOCATIONS 0x9107
#define GL_SAMPLER_2D_MULTISAMPLE 0x9108
#define GL_INT_SAMPLER_2D_MULTISAMPLE 0x9109
#define GL_UNSIGNED_INT_SAMPLER_2D_MULTISAMPLE 0x910A
#define GL_SAMPLER_2D_MULTISAMPLE_ARRAY 0x910B
#define GL_INT_SAMPLER_2D_MULTISAMPLE_ARRAY 0x910C
#define GL_UNSIGNED_INT_SAMPLER_2D_MULTISAMPLE_ARRAY 0x910D
#define GL_MAX_COLOR_TEXTURE_SAMPLES 0x910E
#define GL_MAX_DEPTH_TEXTURE_SAMPLES 0x910F
#define GL_MAX_INTEGER_SAMPLES 0x9110
#define GL_VERTEX_ATTRIB_ARRAY_DIVISOR 0x88FE
#define GL_SRC1_COLOR 0x88F9
#define GL_ONE_MINUS_SRC1_COLOR 0x88FA
#define GL_ONE_MINUS_SRC1_ALPHA 0x88FB
#define GL_MAX_DUAL_SOURCE_DRAW_BUFFERS 0x88FC
#define GL_ANY_SAMPLES_PASSED 0x8C2F
#define GL_SAMPLER_BINDING 0x8919
#define GL_RGB10_A2UI 0x906F
#define GL_TEXTURE_SWIZZLE_R 0x8E42
#define GL_TEXTURE_SWIZZLE_G 0x8E43
#define GL_TEXTURE_SWIZZLE_B 0x8E44
#define GL_TEXTURE_SWIZZLE_A 0x8E45
#define GL_TEXTURE_SWIZZLE_RGBA 0x8E46
#define GL_TIME_ELAPSED 0x88BF
#define GL_TIMESTAMP 0x8E28
#define GL_INT_2_10_10_10_REV 0x8D9F
#define GL_SAMPLE_SHADING 0x8C36
#define GL_MIN_SAMPLE_SHADING_VALUE 0x8C37
#define GL_MIN_PROGRAM_TEXTURE_GATHER_OFFSET 0x8E5E
#define GL_MAX_PROGRAM_TEXTURE_GATHER_OFFSET 0x8E5F
#define GL_TEXTURE_CUBE_MAP_ARRAY 0x9009
#define GL_TEXTURE_BINDING_CUBE_MAP_ARRAY 0x900A
#define GL_PROXY_TEXTURE_CUBE_MAP_ARRAY 0x900B
#define GL_SAMPLER_CUBE_MAP_ARRAY 0x900C
#define GL_SAMPLER_CUBE_MAP_ARRAY_SHADOW 0x900D
#define GL_INT_SAMPLER_CUBE_MAP_ARRAY 0x900E
#define GL_UNSIGNED_INT_SAMPLER_CUBE_MAP_ARRAY 0x900F
#define GL_DRAW_INDIRECT_BUFFER 0x8F3F
#define GL_DRAW_INDIRECT_BUFFER_BINDING 0x8F43
#define GL_GEOMETRY_SHADER_INVOCATIONS 0x887F
#define GL_MAX_GEOMETRY_SHADER_INVOCATIONS 0x8E5A
#define GL_MIN_FRAGMENT_INTERPOLATION_OFFSET 0x8E5B
#define GL_MAX_FRAGMENT_INTERPOLATION_OFFSET 0x8E5C
#define GL_FRAGMENT_INTERPOLATION_OFFSET_BITS 0x8E5D
#define GL_MAX_VERTEX_STREAMS 0x8E71
```

```
#define GL_DOUBLE_VEC2 0x8FFC
#define GL_DOUBLE_VEC3 0x8FFD
#define GL_DOUBLE_VEC4 0x8FFE
#define GL_DOUBLE_MAT2 0x8F46
#define GL_DOUBLE_MAT3 0x8F47
#define GL_DOUBLE_MAT4 0x8F48
#define GL_DOUBLE_MAT2x3 0x8F49
#define GL_DOUBLE_MAT2x4 0x8F4A
#define GL_DOUBLE_MAT3x2 0x8F4B
#define GL_DOUBLE_MAT3x4 0x8F4C
#define GL_DOUBLE_MAT4x2 0x8F4D
#define GL_DOUBLE_MAT4x3 0x8F4E
#define GL_ACTIVE_SUBROUTINES 0x8DE5
#define GL_ACTIVE_SUBROUTINE_UNIFORMS 0x8DE6
#define GL_ACTIVE_SUBROUTINE_UNIFORM_LOCATIONS 0x8E47
#define GL_ACTIVE_SUBROUTINE_MAX_LENGTH 0x8E48
#define GL_ACTIVE_SUBROUTINE_UNIFORM_MAX_LENGTH 0x8E49
#define GL_MAX_SUBROUTINES 0x8DE7
#define GL_MAX_SUBROUTINE_UNIFORM_LOCATIONS 0x8DE8
#define GL_NUM_COMPATIBLE_SUBROUTINES 0x8E4A
#define GL_COMPATIBLE_SUBROUTINES 0x8E4B
#define GL_PATCHES 0x000E
#define GL_PATCH_VERTICES 0x8E72
#define GL_PATCH_DEFAULT_INNER_LEVEL 0x8E73
#define GL_PATCH_DEFAULT_OUTER_LEVEL 0x8E74
#define GL_TESS_CONTROL_OUTPUT_VERTICES 0x8E75
#define GL_TESS_GEN_MODE 0x8E76
#define GL_TESS_GEN_SPACING 0x8E77
#define GL_TESS_GEN_VERTEX_ORDER 0x8E78
#define GL_TESS_GEN_POINT_MODE 0x8E79
#define GL_ISOLINES 0x8E7A
#define GL_QUADS 0x0007
#define GL_FRACTIONAL_ODD 0x8E7B
#define GL_FRACTIONAL_EVEN 0x8E7C
#define GL_MAX_PATCH_VERTICES 0x8E7D
#define GL_MAX_TESS_GEN_LEVEL 0x8E7E
#define GL_MAX_TESS_CONTROL_UNIFORM_COMPONENTS 0x8E7F
#define GL_MAX_TESS_EVALUATION_UNIFORM_COMPONENTS 0x8E80
#define GL_MAX_TESS_CONTROL_TEXTURE_IMAGE_UNITS 0x8E81
#define GL_MAX_TESS_EVALUATION_TEXTURE_IMAGE_UNITS 0x8E82
#define GL_MAX_TESS_CONTROL_OUTPUT_COMPONENTS 0x8E83
#define GL_MAX_TESS_PATCH_COMPONENTS 0x8E84
#define GL_MAX_TESS_CONTROL_TOTAL_OUTPUT_COMPONENTS 0x8E85
#define GL_MAX_TESS_EVALUATION_OUTPUT_COMPONENTS 0x8E86
#define GL_MAX_TESS_CONTROL_UNIFORM_BLOCKS 0x8E89
#define GL_MAX_TESS_EVALUATION_UNIFORM_BLOCKS 0x8E8A
#define GL_MAX_TESS_CONTROL_INPUT_COMPONENTS 0x886C
#define GL_MAX_TESS_EVALUATION_INPUT_COMPONENTS 0x886D
#define GL_MAX_COMBINED_TESS_CONTROL_UNIFORM_COMPONENTS 0x8E1E
#define GL_MAX_COMBINED_TESS_EVALUATION_UNIFORM_COMPONENTS 0x8E1F
#define GL_UNIFORM_BLOCK_REFERENCED_BY_TESS_CONTROL_SHADER 0x84F0
#define GL_UNIFORM_BLOCK_REFERENCED_BY_TESS_EVALUATION_SHADER 0x84F1
#define GL_TESS_EVALUATION_SHADER 0x8E87
#define GL_TESS_CONTROL_SHADER 0x8E88
#define GL_TRANSFORM_FEEDBACK 0x8E22
#define GL_TRANSFORM_FEEDBACK_BUFFER_PAUSED 0x8E23
#define GL_TRANSFORM_FEEDBACK_BUFFER_ACTIVE 0x8E24
#define GL_TRANSFORM_FEEDBACK_BINDING 0x8E25
#define GL_MAX_TRANSFORM_FEEDBACK_BUFFERS 0x8E70
#define GL_FIXED 0x140C
#define GL_IMPLEMENTATION_COLOR_READ_TYPE 0x8B9A
#define GL_IMPLEMENTATION_COLOR_READ_FORMAT 0x8B9B
#define GL_LOW_FLOAT 0x8DF0
#define GL_MEDIUM_FLOAT 0x8DF1
#define GL_HIGH_FLOAT 0x8DF2
#define GL_LOW_INT 0x8DF3
#define GL_MEDIUM_INT 0x8DF4
#define GL_HIGH_INT 0x8DF5
#define GL_SHADER_COMPILER 0x8DFA
#define GL_SHADER_BINARY_FORMATS 0x8DF8
#define GL_NUM_SHADER_BINARY_FORMATS 0x8DF9
#define GL_MAX_VERTEX_UNIFORM_VECTORS 0x8DFB
#define GL_MAX_VARYING_VECTORS 0x8DFC
#define GL_MAX_FRAGMENT_UNIFORM_VECTORS 0x8DFD
#define GL_RGB565 0x8D62
#define GL_PROGRAM_BINARY_RETRIEVABLE_HINT 0x8257
#define GL_PROGRAM_BINARY_LENGTH 0x8741
#define GL_NUM_PROGRAM_BINARY_FORMATS 0x87FE
#define GL_PROGRAM_BINARY_FORMATS 0x87FF
#define GL_VERTEX_SHADER_BIT 0x00000001
#define GL_FRAGMENT_SHADER_BIT 0x00000002
#define GL_GEOMETRY_SHADER_BIT 0x00000004
#define GL_TESS_CONTROL_SHADER_BIT 0x00000008
#define GL_TESS_EVALUATION_SHADER_BIT 0x00000010
#define GL_ALL_SHADER_BITS 0xFFFFFFFF
#define GL_PROGRAM_SEPARABLE 0x8258
#define GL_ACTIVE_PROGRAM 0x8259
```

```
#define GL_PROGRAM_PIPELINE_BINDING 0x825A
#define GL_MAX_VIEWPORTS 0x825B
#define GL_VIEWPORT_SUBPIXEL_BITS 0x825C
#define GL_VIEWPORT_BOUNDS_RANGE 0x825D
#define GL_LAYER_PROVOKING_VERTEX 0x825E
#define GL_VIEWPORT_INDEX_PROVOKING_VERTEX 0x825F
#define GL_UNDEFINED_VERTEX 0x8260
#ifndef GL_VERSION_1_0
#define GL_VERSION_1_0 1
GLAPI int GLAD_GL_VERSION_1_0;
typedef void (APIENTRYP PFNGLCULLFACEPROC)(GLenum mode);
GLAPI PFNGLCULLFACEPROC glad_glCullFace;
#define glCullFace glad_glCullFace
typedef void (APIENTRYP PFNGLFRONTFACEPROC)(GLenum mode);
GLAPI PFNGLFRONTFACEPROC glad_glFrontFace;
#define glFrontFace glad_glFrontFace
typedef void (APIENTRYP PFNGLHINTPROC)(GLenum target, GLenum mode);
GLAPI PFNGLHINTPROC glad_glHint;
#define glHint glad_glHint
typedef void (APIENTRYP PFNGLLINEWIDTHPROC)(GLfloat width);
GLAPI PFNGLLINEWIDTHPROC glad_glLineWidth;
#define glLineWidth glad_glLineWidth
typedef void (APIENTRYP PFNGLPOINTSIZEPROC)(GLfloat size);
GLAPI PFNGLPOINTSIZEPROC glad_glPointSize;
#define glPointSize glad_glPointSize
typedef void (APIENTRYP PFNGLPOLYGONMODEPROC)(GLenum face, GLenum mode);
GLAPI PFNGLPOLYGONMODEPROC glad_glPolygonMode;
#define glPolygonMode glad_glPolygonMode
typedef void (APIENTRYP PFNGLSCISSORPROC)(GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLSCISSORPROC glad_glScissor;
#define glScissor glad_glScissor
typedef void (APIENTRYP PFNGLTEXPARAMETERFPROC)(GLenum target, GLenum pname, GLfloat param);
GLAPI PFNGLTEXPARAMETERFPROC glad_glTexParameterf;
#define glTexParameterf glad_glTexParameterf
typedef void (APIENTRYP PFNGLTEXPARAMETERFVPROC)(GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLTEXPARAMETERFVPROC glad_glTexParameterfv;
#define glTexParameterfv glad_glTexParameterfv
typedef void (APIENTRYP PFNGLTEXPARAMETERIPROC)(GLenum target, GLenum pname, GLint param);
GLAPI PFNGLTEXPARAMETERIPROC glad_glTexParameteri;
#define glTexParameteri glad_glTexParameteri
typedef void (APIENTRYP PFNGLTEXPARAMETERIVPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLTEXPARAMETERIVPROC glad_glTexParameteriv;
#define glTexParameteriv glad_glTexParameteriv
typedef void (APIENTRYP PFNGLTEXIMAGE1DPROC)(GLenum target, GLint level, GLint internalformat, GLsizei width, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXIMAGE1DPROC glad_glTexImage1D;
#define glTexImage1D glad_glTexImage1D
typedef void (APIENTRYP PFNGLTEXIMAGE2DPROC)(GLenum target, GLint level, GLint internalformat, GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXIMAGE2DPROC glad_glTexImage2D;
#define glTexImage2D glad_glTexImage2D
typedef void (APIENTRYP PFNGLDRAWBUFFERPROC)(GLenum buf);
GLAPI PFNGLDRAWBUFFERPROC glad_glDrawBuffer;
#define glDrawBuffer glad_glDrawBuffer
typedef void (APIENTRYP PFNGLCLEARPROC)(GLbitfield mask);
GLAPI PFNGLCLEARPROC glad_glClear;
#define glClear glad_glClear
typedef void (APIENTRYP PFNGLCLEARCOLORPROC)(GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha);
GLAPI PFNGLCLEARCOLORPROC glad_glClearColor;
#define glClearColor glad_glClearColor
typedef void (APIENTRYP PFNGLCLEARSTENCILPROC)(GLint s);
GLAPI PFNGLCLEARSTENCILPROC glad_glClearStencil;
#define glClearStencil glad_glClearStencil
typedef void (APIENTRYP PFNGLCLEARDEPTHPROC)(GLdouble depth);
GLAPI PFNGLCLEARDEPTHPROC glad_glClearDepth;
#define glClearDepth glad_glClearDepth
typedef void (APIENTRYP PFNGLSTENCILMASKPROC)(GLuint mask);
GLAPI PFNGLSTENCILMASKPROC glad_glStencilMask;
#define glStencilMask glad_glStencilMask
typedef void (APIENTRYP PFNGLCOLORMASKPROC)(GLboolean red, GLboolean green, GLboolean blue, GLboolean alpha);
GLAPI PFNGLCOLORMASKPROC glad_glColorMask;
#define glColorMask glad_glColorMask
typedef void (APIENTRYP PFNGLDEPTHMASKPROC)(GLboolean flag);
GLAPI PFNGLDEPTHMASKPROC glad_glDepthMask;
#define glDepthMask glad_glDepthMask
typedef void (APIENTRYP PFNGLDISABLEPROC)(GLenum cap);
GLAPI PFNGLDISABLEPROC glad_glDisable;
#define glDisable glad_glDisable
typedef void (APIENTRYP PFNGLENABLEPROC)(GLenum cap);
GLAPI PFNGLENABLEPROC glad_glEnable;
#define glEnable glad_glEnable
typedef void (APIENTRYP PFNGLFINISHPROC)(void);
GLAPI PFNGLFINISHPROC glad_glFinish;
#define glFinish glad_glFinish
typedef void (APIENTRYP PFNGLFLUSHPROC)(void);
GLAPI PFNGLFLUSHPROC glad_glFlush;
```

```
#define glFlush glad_glFlush
typedef void (APIENTRYP PFNGLBLENDFUNCPROC)(GLenum sfactor, GLenum dfactor);
GLAPI PFNGLBLENDFUNCPROC glad_glBlendFunc;
#define glBlendFunc glad_glBlendFunc
typedef void (APIENTRYP PFNGLLOGICOPPROC)(GLenum opcode);
GLAPI PFNGLLOGICOPPROC glad_glLogicOp;
#define glLogicOp glad_glLogicOp
typedef void (APIENTRYP PFNGLSTENCILFUNCPROC)(GLenum func, GLint ref, GLuint mask);
GLAPI PFNGLSTENCILFUNCPROC glad_glStencilFunc;
#define glStencilFunc glad_glStencilFunc
typedef void (APIENTRYP PFNGLSTENCILOPPROC)(GLenum fail, GLenum zfail, GLenum zpass);
GLAPI PFNGLSTENCILOPPROC glad_glStencilOp;
#define glStencilOp glad_glStencilOp
typedef void (APIENTRYP PFNGLDEPTHFUNCPROC)(GLenum func);
GLAPI PFNGLDEPTHFUNCPROC glad_glDepthFunc;
#define glDepthFunc glad_glDepthFunc
typedef void (APIENTRYP PFNGLPIXELSTOREFPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLPIXELSTOREFPROC glad_glPixelStoref;
#define glPixelStoref glad_glPixelStoref
typedef void (APIENTRYP PFNGLPIXELSTOREIPROC)(GLenum pname, GLint param);
GLAPI PFNGLPIXELSTOREIPROC glad_glPixelStorei;
#define glPixelStorei glad_glPixelStorei
typedef void (APIENTRYP PFNGLREADBUFFERPROC)(GLenum src);
GLAPI PFNGLREADBUFFERPROC glad_glReadBuffer;
#define glReadBuffer glad_glReadBuffer
typedef void (APIENTRYP PFNGLREADPIXELSPROC)(GLint x, GLint y, GLsizei width, GLsizei height, GLenum format, GLenum type, void *pixels);
GLAPI PFNGLREADPIXELSPROC glad_glReadPixels;
#define glReadPixels glad_glReadPixels
typedef void (APIENTRYP PFNGLGETBOOLEANVPROC)(GLenum pname, GLboolean *data);
GLAPI PFNGLGETBOOLEANVPROC glad_glGetBooleanv;
#define glGetBooleanv glad_glGetBooleanv
typedef void (APIENTRYP PFNGLGETDOUBLEVPROC)(GLenum pname, GLdouble *data);
GLAPI PFNGLGETDOUBLEVPROC glad_glGetDoublev;
#define glGetDoublev glad_glGetDoublev
typedef GLenum (APIENTRYP PFNGLGETERRORPROC)(void);
GLAPI PFNGLGETERRORPROC glad_glGetError;
#define glGetError glad_glGetError
typedef void (APIENTRYP PFNGLGETFLOATVPROC)(GLenum pname, GLfloat *data);
GLAPI PFNGLGETFLOATVPROC glad_glGetFloatv;
#define glGetFloatv glad_glGetFloatv
typedef void (APIENTRYP PFNGLGETINTEGERVPROC)(GLenum pname, GLint *data);
GLAPI PFNGLGETINTEGERVPROC glad_glGetIntegerv;
#define glGetIntegerv glad_glGetIntegerv
typedef const GLubyte * (APIENTRYP PFNGLGETSTRINGPROC)(GLenum name);
GLAPI PFNGLGETSTRINGPROC glad_glGetString;
#define glGetString glad_glGetString
typedef void (APIENTRYP PFNGLGETTEXIMAGEPROC)(GLenum target, GLint level, GLenum format, GLenum type, void *pixels);
GLAPI PFNGLGETTEXIMAGEPROC glad_glGetTexImage;
#define glGetTexImage glad_glGetTexImage
typedef void (APIENTRYP PFNGLGETTEXPARAMETERFVPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETTEXPARAMETERFVPROC glad_glGetTexParameterfv;
#define glGetTexParameterfv glad_glGetTexParameterfv
typedef void (APIENTRYP PFNGLGETTEXPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXPARAMETERIVPROC glad_glGetTexParameteriv;
#define glGetTexParameteriv glad_glGetTexParameteriv
typedef void (APIENTRYP PFNGLGETTEXLEVELPARAMETERFVPROC)(GLenum target, GLint level, GLenum pname, GLfloat *params);
GLAPI PFNGLGETTEXLEVELPARAMETERFVPROC glad_glGetTexLevelParameterfv;
#define glGetTexLevelParameterfv glad_glGetTexLevelParameterfv
typedef void (APIENTRYP PFNGLGETTEXLEVELPARAMETERIVPROC)(GLenum target, GLint level, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXLEVELPARAMETERIVPROC glad_glGetTexLevelParameteriv;
#define glGetTexLevelParameteriv glad_glGetTexLevelParameteriv
typedef GLboolean (APIENTRYP PFNGLISENABLEDPROC)(GLenum cap);
GLAPI PFNGLISENABLEDPROC glad_glIsEnabled;
#define glIsEnabled glad_glIsEnabled
typedef void (APIENTRYP PFNGLDEPTHRANGEPROC)(GLdouble n, GLdouble f);
GLAPI PFNGLDEPTHRANGEPROC glad_glDepthRange;
#define glDepthRange glad_glDepthRange
typedef void (APIENTRYP PFNGLVIEWPORTPROC)(GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLVIEWPORTPROC glad_glViewport;
#define glViewport glad_glViewport
#endif
#ifndef GL_VERSION_1_1
#define GL_VERSION_1_1 1
GLAPI int GLAD_GL_VERSION_1_1;
typedef void (APIENTRYP PFNGLDRAWARRAYSPROC)(GLenum mode, GLint first, GLsizei count);
GLAPI PFNGLDRAWARRAYSPROC glad_glDrawArrays;
#define glDrawArrays glad_glDrawArrays
typedef void (APIENTRYP PFNGLDRAWELEMENTSPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices);
GLAPI PFNGLDRAWELEMENTSPROC glad_glDrawElements;
#define glDrawElements glad_glDrawElements
typedef void (APIENTRYP PFNGLPOLYGONOFFSETPROC)(GLfloat factor, GLfloat units);
GLAPI PFNGLPOLYGONOFFSETPROC glad_glPolygonOffset;
```

```
#define glPolygonOffset glad_glPolygonOffset
typedef void (APIENTRYP PFNGLCOPYTEXIMAGE1DPROC)(GLenum target, GLint level, GLenum internalformat, GLint x, GLint y, GLsizei width, GLint border);
GLAPI PFNGLCOPYTEXIMAGE1DPROC glad_glCopyTexImage1D;
#define glCopyTexImage1D glad_glCopyTexImage1D
typedef void (APIENTRYP PFNGLCOPYTEXIMAGE2DPROC)(GLenum target, GLint level, GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
GLAPI PFNGLCOPYTEXIMAGE2DPROC glad_glCopyTexImage2D;
#define glCopyTexImage2D glad_glCopyTexImage2D
typedef void (APIENTRYP PFNGLCOPYTEXSUBIMAGE1DPROC)(GLenum target, GLint level, GLint xoffset, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYTEXSUBIMAGE1DPROC glad_glCopyTexSubImage1D;
#define glCopyTexSubImage1D glad_glCopyTexSubImage1D
typedef void (APIENTRYP PFNGLCOPYTEXSUBIMAGE2DPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXSUBIMAGE2DPROC glad_glCopyTexSubImage2D;
#define glCopyTexSubImage2D glad_glCopyTexSubImage2D
typedef void (APIENTRYP PFNGLTEXSUBIMAGE1DPROC)(GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXSUBIMAGE1DPROC glad_glTexSubImage1D;
#define glTexSubImage1D glad_glTexSubImage1D
typedef void (APIENTRYP PFNGLTEXSUBIMAGE2DPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXSUBIMAGE2DPROC glad_glTexSubImage2D;
#define glTexSubImage2D glad_glTexSubImage2D
typedef void (APIENTRYP PFNGLBINDTEXTUREPROC)(GLenum target, GLuint texture);
GLAPI PFNGLBINDTEXTUREPROC glad_glBindTexture;
#define glBindTexture glad_glBindTexture
typedef void (APIENTRYP PFNGLDELETETEXTURESPROC)(GLsizei n, const GLuint *textures);
GLAPI PFNGLDELETETEXTURESPROC glad_glDeleteTextures;
#define glDeleteTextures glad_glDeleteTextures
typedef void (APIENTRYP PFNGLGENTEXTURESPROC)(GLsizei n, GLuint *textures);
GLAPI PFNGLGENTEXTURESPROC glad_glGenTextures;
#define glGenTextures glad_glGenTextures
typedef GLboolean (APIENTRYP PFNGLISTEXTUREPROC)(GLuint texture);
GLAPI PFNGLISTEXTUREPROC glad_glIsTexture;
#define glIsTexture glad_glIsTexture
#endif
#ifndef GL_VERSION_1_2
#define GL_VERSION_1_2 1
GLAPI int GLAD_GL_VERSION_1_2;
typedef void (APIENTRYP PFNGLDRAWRANGEELEMENTSPROC)(GLenum mode, GLuint start, GLuint end, GLsizei count, GLenum type, const void *indices);
GLAPI PFNGLDRAWRANGEELEMENTSPROC glad_glDrawRangeElements;
#define glDrawRangeElements glad_glDrawRangeElements
typedef void (APIENTRYP PFNGLTEXIMAGE3DPROC)(GLenum target, GLint level, GLint internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXIMAGE3DPROC glad_glTexImage3D;
#define glTexImage3D glad_glTexImage3D
typedef void (APIENTRYP PFNGLTEXSUBIMAGE3DPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXSUBIMAGE3DPROC glad_glTexSubImage3D;
#define glTexSubImage3D glad_glTexSubImage3D
typedef void (APIENTRYP PFNGLCOPYTEXSUBIMAGE3DPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXSUBIMAGE3DPROC glad_glCopyTexSubImage3D;
#define glCopyTexSubImage3D glad_glCopyTexSubImage3D
#endif
#ifndef GL_VERSION_1_3
#define GL_VERSION_1_3 1
GLAPI int GLAD_GL_VERSION_1_3;
typedef void (APIENTRYP PFNGLACTIVETEXTUREPROC)(GLenum texture);
GLAPI PFNGLACTIVETEXTUREPROC glad_glActiveTexture;
#define glActiveTexture glad_glActiveTexture
typedef void (APIENTRYP PFNGLSAMPLECOVERAGEPROC)(GLfloat value, GLboolean invert);
GLAPI PFNGLSAMPLECOVERAGEPROC glad_glSampleCoverage;
#define glSampleCoverage glad_glSampleCoverage
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXIMAGE3DPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXIMAGE3DPROC glad_glCompressedTexImage3D;
#define glCompressedTexImage3D glad_glCompressedTexImage3D
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXIMAGE2DPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXIMAGE2DPROC glad_glCompressedTexImage2D;
#define glCompressedTexImage2D glad_glCompressedTexImage2D
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXIMAGE1DPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLint border, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXIMAGE1DPROC glad_glCompressedTexImage1D;
#define glCompressedTexImage1D glad_glCompressedTexImage1D
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXSUBIMAGE3DPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE3DPROC glad_glCompressedTexSubImage3D;
#define glCompressedTexSubImage3D glad_glCompressedTexSubImage3D
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXSUBIMAGE2DPROC)(GLenum target, GLint level, GLint xoffset,
GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE2DPROC glad_glCompressedTexSubImage2D;
#define glCompressedTexSubImage2D glad_glCompressedTexSubImage2D
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXSUBIMAGE1DPROC)(GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE1DPROC glad_glCompressedTexSubImage1D;
#define glCompressedTexSubImage1D glad_glCompressedTexSubImage1D
typedef void (APIENTRYP PFNGLGETCOMPRESSEDTEXIMAGEPROC)(GLenum target, GLint level, void *img);
GLAPI PFNGLGETCOMPRESSEDTEXIMAGEPROC glad_glGetCompressedTexImage;
#define glGetCompressedTexImage glad_glGetCompressedTexImage
#endif
#ifndef GL_VERSION_1_4
#define GL_VERSION_1_4 1
GLAPI int GLAD_GL_VERSION_1_4;
typedef void (APIENTRYP PFNGLBLENDFUNCSEPARATEPROC)(GLenum sfactorRGB, GLenum dfactorRGB, GLenum sfactorAlpha, GLenum dfactorAlpha);
GLAPI PFNGLBLENDFUNCSEPARATEPROC glad_glBlendFuncSeparate;
#define glBlendFuncSeparate glad_glBlendFuncSeparate
typedef void (APIENTRYP PFNGLMULTIDRAWARRAYSPROC)(GLenum mode, const GLint *first, const GLsizei *count, GLsizei drawcount);
GLAPI PFNGLMULTIDRAWARRAYSPROC glad_glMultiDrawArrays;
#define glMultiDrawArrays glad_glMultiDrawArrays
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSPROC)(GLenum mode, const GLsizei *count, GLenum type, const void *const*indices, GLsizei drawcount);
GLAPI PFNGLMULTIDRAWELEMENTSPROC glad_glMultiDrawElements;
#define glMultiDrawElements glad_glMultiDrawElements
typedef void (APIENTRYP PFNGLPOINTPARAMETERFPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLPOINTPARAMETERFPROC glad_glPointParameterf;
#define glPointParameterf glad_glPointParameterf
typedef void (APIENTRYP PFNGLPOINTPARAMETERFVPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLPOINTPARAMETERFVPROC glad_glPointParameterfv;
#define glPointParameterfv glad_glPointParameterfv
typedef void (APIENTRYP PFNGLPOINTPARAMETERIPROC)(GLenum pname, GLint param);
GLAPI PFNGLPOINTPARAMETERIPROC glad_glPointParameteri;
#define glPointParameteri glad_glPointParameteri
typedef void (APIENTRYP PFNGLPOINTPARAMETERIVPROC)(GLenum pname, const GLint *params);
GLAPI PFNGLPOINTPARAMETERIVPROC glad_glPointParameteriv;
#define glPointParameteriv glad_glPointParameteriv
typedef void (APIENTRYP PFNGLBLENDCOLORPROC)(GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha);
GLAPI PFNGLBLENDCOLORPROC glad_glBlendColor;
#define glBlendColor glad_glBlendColor
typedef void (APIENTRYP PFNGLBLENDEQUATIONPROC)(GLenum mode);
GLAPI PFNGLBLENDEQUATIONPROC glad_glBlendEquation;
#define glBlendEquation glad_glBlendEquation
#endif
#ifndef GL_VERSION_1_5
#define GL_VERSION_1_5 1
GLAPI int GLAD_GL_VERSION_1_5;
typedef void (APIENTRYP PFNGLGENQUERIESPROC)(GLsizei n, GLuint *ids);
GLAPI PFNGLGENQUERIESPROC glad_glGenQueries;
#define glGenQueries glad_glGenQueries
typedef void (APIENTRYP PFNGLDELETEQUERIESPROC)(GLsizei n, const GLuint *ids);
GLAPI PFNGLDELETEQUERIESPROC glad_glDeleteQueries;
#define glDeleteQueries glad_glDeleteQueries
typedef GLboolean (APIENTRYP PFNGLISQUERYPROC)(GLuint id);
GLAPI PFNGLISQUERYPROC glad_glIsQuery;
#define glIsQuery glad_glIsQuery
typedef void (APIENTRYP PFNGLBEGINQUERYPROC)(GLenum target, GLuint id);
GLAPI PFNGLBEGINQUERYPROC glad_glBeginQuery;
#define glBeginQuery glad_glBeginQuery
typedef void (APIENTRYP PFNGLENDQUERYPROC)(GLenum target);
GLAPI PFNGLENDQUERYPROC glad_glEndQuery;
#define glEndQuery glad_glEndQuery
typedef void (APIENTRYP PFNGLGETQUERYIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETQUERYIVPROC glad_glGetQueryiv;
#define glGetQueryiv glad_glGetQueryiv
typedef void (APIENTRYP PFNGLGETQUERYOBJECTIVPROC)(GLuint id, GLenum pname, GLint *params);
GLAPI PFNGLGETQUERYOBJECTIVPROC glad_glGetQueryObjectiv;
#define glGetQueryObjectiv glad_glGetQueryObjectiv
typedef void (APIENTRYP PFNGLGETQUERYOBJECTUIVPROC)(GLuint id, GLenum pname, GLuint *params);
GLAPI PFNGLGETQUERYOBJECTUIVPROC glad_glGetQueryObjectuiv;
#define glGetQueryObjectuiv glad_glGetQueryObjectuiv
typedef void (APIENTRYP PFNGLBINDBUFFERPROC)(GLenum target, GLuint buffer);
GLAPI PFNGLBINDBUFFERPROC glad_glBindBuffer;
#define glBindBuffer glad_glBindBuffer
typedef void (APIENTRYP PFNGLDELETEBUFFERSPROC)(GLsizei n, const GLuint *buffers);
GLAPI PFNGLDELETEBUFFERSPROC glad_glDeleteBuffers;
#define glDeleteBuffers glad_glDeleteBuffers
typedef void (APIENTRYP PFNGLGENBUFFERSPROC)(GLsizei n, GLuint *buffers);
GLAPI PFNGLGENBUFFERSPROC glad_glGenBuffers;
#define glGenBuffers glad_glGenBuffers
typedef GLboolean (APIENTRYP PFNGLISBUFFERPROC)(GLuint buffer);
GLAPI PFNGLISBUFFERPROC glad_glIsBuffer;
#define glIsBuffer glad_glIsBuffer
typedef void (APIENTRYP PFNGLBUFFERDATAPROC)(GLenum target, GLsizeiptr size, const void *data, GLenum usage);
GLAPI PFNGLBUFFERDATAPROC glad_glBufferData;
```

```
#define glBufferData glad_glBufferData
typedef void (APIENTRYP PFNGLBUFFERSUBDATAPROC)(GLenum target, GLintptr offset, GLsizeiptr size, const void *data);
GLAPI PFNGLBUFFERSUBDATAPROC glad_glBufferSubData;
#define glBufferSubData glad_glBufferSubData
typedef void (APIENTRYP PFNGLGETBUFFERSUBDATAPROC)(GLenum target, GLintptr offset, GLsizeiptr size, void *data);
GLAPI PFNGLGETBUFFERSUBDATAPROC glad_glGetBufferSubData;
#define glGetBufferSubData glad_glGetBufferSubData
typedef void * (APIENTRYP PFNGLMAPBUFFERPROC)(GLenum target, GLenum access);
GLAPI PFNGLMAPBUFFERPROC glad_glMapBuffer;
#define glMapBuffer glad_glMapBuffer
typedef GLboolean (APIENTRYP PFNGLUNMAPBUFFERPROC)(GLenum target);
GLAPI PFNGLUNMAPBUFFERPROC glad_glUnmapBuffer;
#define glUnmapBuffer glad_glUnmapBuffer
typedef void (APIENTRYP PFNGLGETBUFFERPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETBUFFERPARAMETERIVPROC glad_glGetBufferParameteriv;
#define glGetBufferParameteriv glad_glGetBufferParameteriv
typedef void (APIENTRYP PFNGLGETBUFFERPOINTERVPROC)(GLenum target, GLenum pname, void **params);
GLAPI PFNGLGETBUFFERPOINTERVPROC glad_glGetBufferPointerv;
#define glGetBufferPointerv glad_glGetBufferPointerv
#endif
#ifndef GL_VERSION_2_0
#define GL_VERSION_2_0 1
GLAPI int GLAD_GL_VERSION_2_0;
typedef void (APIENTRYP PFNGLBLENDEQUATIONSEPARATEPROC)(GLenum modeRGB, GLenum modeAlpha);
GLAPI PFNGLBLENDEQUATIONSEPARATEPROC glad_glBlendEquationSeparate;
#define glBlendEquationSeparate glad_glBlendEquationSeparate
typedef void (APIENTRYP PFNGLDRAWBUFFERSPROC)(GLsizei n, const GLenum *bufs);
GLAPI PFNGLDRAWBUFFERSPROC glad_glDrawBuffers;
#define glDrawBuffers glad_glDrawBuffers
typedef void (APIENTRYP PFNGLSTENCILOPSEPARATEPROC)(GLenum face, GLenum sfail, GLenum dpfail, GLenum dppass);
GLAPI PFNGLSTENCILOPSEPARATEPROC glad_glStencilOpSeparate;
#define glStencilOpSeparate glad_glStencilOpSeparate
typedef void (APIENTRYP PFNGLSTENCILFUNCSEPARATEPROC)(GLenum face, GLenum func, GLint ref, GLuint mask);
GLAPI PFNGLSTENCILFUNCSEPARATEPROC glad_glStencilFuncSeparate;
#define glStencilFuncSeparate glad_glStencilFuncSeparate
typedef void (APIENTRYP PFNGLSTENCILMASKSEPARATEPROC)(GLenum face, GLuint mask);
GLAPI PFNGLSTENCILMASKSEPARATEPROC glad_glStencilMaskSeparate;
#define glStencilMaskSeparate glad_glStencilMaskSeparate
typedef void (APIENTRYP PFNGLATTACHSHADERPROC)(GLuint program, GLuint shader);
GLAPI PFNGLATTACHSHADERPROC glad_glAttachShader;
#define glAttachShader glad_glAttachShader
typedef void (APIENTRYP PFNGLBINDATTRIBLOCATIONPROC)(GLuint program, GLuint index, const GLchar *name);
GLAPI PFNGLBINDATTRIBLOCATIONPROC glad_glBindAttribLocation;
#define glBindAttribLocation glad_glBindAttribLocation
typedef void (APIENTRYP PFNGLCOMPILESHADERPROC)(GLuint shader);
GLAPI PFNGLCOMPILESHADERPROC glad_glCompileShader;
#define glCompileShader glad_glCompileShader
typedef GLuint (APIENTRYP PFNGLCREATEPROGRAMPROC)(void);
GLAPI PFNGLCREATEPROGRAMPROC glad_glCreateProgram;
#define glCreateProgram glad_glCreateProgram
typedef GLuint (APIENTRYP PFNGLCREATESHADERPROC)(GLenum type);
GLAPI PFNGLCREATESHADERPROC glad_glCreateShader;
#define glCreateShader glad_glCreateShader
typedef void (APIENTRYP PFNGLDELETEPROGRAMPROC)(GLuint program);
GLAPI PFNGLDELETEPROGRAMPROC glad_glDeleteProgram;
#define glDeleteProgram glad_glDeleteProgram
typedef void (APIENTRYP PFNGLDELETESHADERPROC)(GLuint shader);
GLAPI PFNGLDELETESHADERPROC glad_glDeleteShader;
#define glDeleteShader glad_glDeleteShader
typedef void (APIENTRYP PFNGLDETACHSHADERPROC)(GLuint program, GLuint shader);
GLAPI PFNGLDETACHSHADERPROC glad_glDetachShader;
#define glDetachShader glad_glDetachShader
typedef void (APIENTRYP PFNGLDISABLEVERTEXATTRIBARRAYPROC)(GLuint index);
GLAPI PFNGLDISABLEVERTEXATTRIBARRAYPROC glad_glDisableVertexAttribArray;
#define glDisableVertexAttribArray glad_glDisableVertexAttribArray
typedef void (APIENTRYP PFNGLENABLEVERTEXATTRIBARRAYPROC)(GLuint index);
GLAPI PFNGLENABLEVERTEXATTRIBARRAYPROC glad_glEnableVertexAttribArray;
#define glEnableVertexAttribArray glad_glEnableVertexAttribArray
typedef void (APIENTRYP PFNGLGETACTIVEATTRIBPROC)(GLuint program, GLuint index, GLsizei bufSize, GLsizei *length, GLint *size, GLenum *type, GLchar *name);
GLAPI PFNGLGETACTIVEATTRIBPROC glad_glGetActiveAttrib;
#define glGetActiveAttrib glad_glGetActiveAttrib
typedef void (APIENTRYP PFNGLGETACTIVEUNIFORMPROC)(GLuint program, GLuint index, GLsizei bufSize, GLsizei *length, GLint *size, GLenum *type, GLchar *name);
GLAPI PFNGLGETACTIVEUNIFORMPROC glad_glGetActiveUniform;
#define glGetActiveUniform glad_glGetActiveUniform
typedef void (APIENTRYP PFNGLGETATTACHEDSHADERSPROC)(GLuint program, GLsizei maxCount, GLsizei *count, GLuint *shaders);
GLAPI PFNGLGETATTACHEDSHADERSPROC glad_glGetAttachedShaders;
#define glGetAttachedShaders glad_glGetAttachedShaders
typedef GLint (APIENTRYP PFNGLGETATTRIBLOCATIONPROC)(GLuint program, const GLchar *name);
GLAPI PFNGLGETATTRIBLOCATIONPROC glad_glGetAttribLocation;
```

```
#define glGetAttribLocation glad_glGetAttribLocation
typedef void (APIENTRYP PFNGLGETPROGRAMIVPROC)(GLuint program, GLenum pname, GLint *params);
GLAPI PFNGLGETPROGRAMIVPROC glad_glGetProgramiv;
#define glGetProgramiv glad_glGetProgramiv
typedef void (APIENTRYP PFNGLGETPROGRAMINFOLOGPROC)(GLuint program, GLsizei bufSize, GLsizei *length, GLchar *infoLog);
GLAPI PFNGLGETPROGRAMINFOLOGPROC glad_glGetProgramInfoLog;
#define glGetProgramInfoLog glad_glGetProgramInfoLog
typedef void (APIENTRYP PFNGLGETSHADERIVPROC)(GLuint shader, GLenum pname, GLint *params);
GLAPI PFNGLGETSHADERIVPROC glad_glGetShaderiv;
#define glGetShaderiv glad_glGetShaderiv
typedef void (APIENTRYP PFNGLGETSHADERINFOLOGPROC)(GLuint shader, GLsizei bufSize, GLsizei *length, GLchar *infoLog);
GLAPI PFNGLGETSHADERINFOLOGPROC glad_glGetShaderInfoLog;
#define glGetShaderInfoLog glad_glGetShaderInfoLog
typedef void (APIENTRYP PFNGLGETSHADERSOURCEPROC)(GLuint shader, GLsizei bufSize, GLsizei *length, GLchar *source);
GLAPI PFNGLGETSHADERSOURCEPROC glad_glGetShaderSource;
#define glGetShaderSource glad_glGetShaderSource
typedef GLint (APIENTRYP PFNGLGETUNIFORMLOCATIONPROC)(GLuint program, const GLchar *name);
GLAPI PFNGLGETUNIFORMLOCATIONPROC glad_glGetUniformLocation;
#define glGetUniformLocation glad_glGetUniformLocation
typedef void (APIENTRYP PFNGLGETUNIFORMFVPROC)(GLuint program, GLint location, GLfloat *params);
GLAPI PFNGLGETUNIFORMFVPROC glad_glGetUniformfv;
#define glGetUniformfv glad_glGetUniformfv
typedef void (APIENTRYP PFNGLGETUNIFORMIVPROC)(GLuint program, GLint location, GLint *params);
GLAPI PFNGLGETUNIFORMIVPROC glad_glGetUniformiv;
#define glGetUniformiv glad_glGetUniformiv
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBDVPROC)(GLuint index, GLenum pname, GLdouble *params);
GLAPI PFNGLGETVERTEXATTRIBDVPROC glad_glGetVertexAttribdv;
#define glGetVertexAttribdv glad_glGetVertexAttribdv
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBFVPROC)(GLuint index, GLenum pname, GLfloat *params);
GLAPI PFNGLGETVERTEXATTRIBFVPROC glad_glGetVertexAttribfv;
#define glGetVertexAttribfv glad_glGetVertexAttribfv
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBIVPROC)(GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETVERTEXATTRIBIVPROC glad_glGetVertexAttribiv;
#define glGetVertexAttribiv glad_glGetVertexAttribiv
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBPOINTERVPROC)(GLuint index, GLenum pname, void **pointer);
GLAPI PFNGLGETVERTEXATTRIBPOINTERVPROC glad_glGetVertexAttribPointerv;
#define glGetVertexAttribPointerv glad_glGetVertexAttribPointerv
typedef GLboolean (APIENTRYP PFNGLISPROGRAMPROC)(GLuint program);
GLAPI PFNGLISPROGRAMPROC glad_glIsProgram;
#define glIsProgram glad_glIsProgram
typedef GLboolean (APIENTRYP PFNGLISSHADERPROC)(GLuint shader);
GLAPI PFNGLISSHADERPROC glad_glIsShader;
#define glIsShader glad_glIsShader
typedef void (APIENTRYP PFNGLLINKPROGRAMPROC)(GLuint program);
GLAPI PFNGLLINKPROGRAMPROC glad_glLinkProgram;
#define glLinkProgram glad_glLinkProgram
typedef void (APIENTRYP PFNGLSHADERSOURCEPROC)(GLuint shader, GLsizei count, const GLchar *const*string, const GLint *length);
GLAPI PFNGLSHADERSOURCEPROC glad_glShaderSource;
#define glShaderSource glad_glShaderSource
typedef void (APIENTRYP PFNGLUSEPROGRAMPROC)(GLuint program);
GLAPI PFNGLUSEPROGRAMPROC glad_glUseProgram;
#define glUseProgram glad_glUseProgram
typedef void (APIENTRYP PFNGLUNIFORM1FPROC)(GLint location, GLfloat v0);
GLAPI PFNGLUNIFORM1FPROC glad_glUniform1f;
#define glUniform1f glad_glUniform1f
typedef void (APIENTRYP PFNGLUNIFORM2FPROC)(GLint location, GLfloat v0, GLfloat v1);
GLAPI PFNGLUNIFORM2FPROC glad_glUniform2f;
#define glUniform2f glad_glUniform2f
typedef void (APIENTRYP PFNGLUNIFORM3FPROC)(GLint location, GLfloat v0, GLfloat v1, GLfloat v2);
GLAPI PFNGLUNIFORM3FPROC glad_glUniform3f;
#define glUniform3f glad_glUniform3f
typedef void (APIENTRYP PFNGLUNIFORM4FPROC)(GLint location, GLfloat v0, GLfloat v1, GLfloat v2, GLfloat v3);
GLAPI PFNGLUNIFORM4FPROC glad_glUniform4f;
#define glUniform4f glad_glUniform4f
typedef void (APIENTRYP PFNGLUNIFORM1IPROC)(GLint location, GLint v0);
GLAPI PFNGLUNIFORM1IPROC glad_glUniform1i;
#define glUniform1i glad_glUniform1i
typedef void (APIENTRYP PFNGLUNIFORM2IPROC)(GLint location, GLint v0, GLint v1);
GLAPI PFNGLUNIFORM2IPROC glad_glUniform2i;
#define glUniform2i glad_glUniform2i
typedef void (APIENTRYP PFNGLUNIFORM3IPROC)(GLint location, GLint v0, GLint v1, GLint v2);
GLAPI PFNGLUNIFORM3IPROC glad_glUniform3i;
#define glUniform3i glad_glUniform3i
typedef void (APIENTRYP PFNGLUNIFORM4IPROC)(GLint location, GLint v0, GLint v1, GLint v2, GLint v3);
GLAPI PFNGLUNIFORM4IPROC glad_glUniform4i;
#define glUniform4i glad_glUniform4i
typedef void (APIENTRYP PFNGLUNIFORM1FVPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM1FVPROC glad_glUniform1fv;
#define glUniform1fv glad_glUniform1fv
typedef void (APIENTRYP PFNGLUNIFORM2FVPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM2FVPROC glad_glUniform2fv;
#define glUniform2fv glad_glUniform2fv
```

```
typedef void (APIENTRYP PFNGLUNIFORM3FVPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM3FVPROC glad_glUniform3fv;
#define glUniform3fv glad_glUniform3fv
typedef void (APIENTRYP PFNGLUNIFORM4FVPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM4FVPROC glad_glUniform4fv;
#define glUniform4fv glad_glUniform4fv
typedef void (APIENTRYP PFNGLUNIFORM1IVPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM1IVPROC glad_glUniform1iv;
#define glUniform1iv glad_glUniform1iv
typedef void (APIENTRYP PFNGLUNIFORM2IVPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM2IVPROC glad_glUniform2iv;
#define glUniform2iv glad_glUniform2iv
typedef void (APIENTRYP PFNGLUNIFORM3IVPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM3IVPROC glad_glUniform3iv;
#define glUniform3iv glad_glUniform3iv
typedef void (APIENTRYP PFNGLUNIFORM4IVPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM4IVPROC glad_glUniform4iv;
#define glUniform4iv glad_glUniform4iv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX2FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX2FVPROC glad_glUniformMatrix2fv;
#define glUniformMatrix2fv glad_glUniformMatrix2fv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX3FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX3FVPROC glad_glUniformMatrix3fv;
#define glUniformMatrix3fv glad_glUniformMatrix3fv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX4FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX4FVPROC glad_glUniformMatrix4fv;
#define glUniformMatrix4fv glad_glUniformMatrix4fv
typedef void (APIENTRYP PFNGLVALIDATEPROGRAMPROC)(GLuint program);
GLAPI PFNGLVALIDATEPROGRAMPROC glad_glValidateProgram;
#define glValidateProgram glad_glValidateProgram
typedef void (APIENTRYP PFNGLVERTEXATTRIB1DPROC)(GLuint index, GLdouble x);
GLAPI PFNGLVERTEXATTRIB1DPROC glad_glVertexAttrib1d;
#define glVertexAttrib1d glad_glVertexAttrib1d
typedef void (APIENTRYP PFNGLVERTEXATTRIB1DVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB1DVPROC glad_glVertexAttrib1dv;
#define glVertexAttrib1dv glad_glVertexAttrib1dv
typedef void (APIENTRYP PFNGLVERTEXATTRIB1FPROC)(GLuint index, GLfloat x);
GLAPI PFNGLVERTEXATTRIB1FPROC glad_glVertexAttrib1f;
#define glVertexAttrib1f glad_glVertexAttrib1f
typedef void (APIENTRYP PFNGLVERTEXATTRIB1FVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB1FVPROC glad_glVertexAttrib1fv;
#define glVertexAttrib1fv glad_glVertexAttrib1fv
typedef void (APIENTRYP PFNGLVERTEXATTRIB1SPROC)(GLuint index, GLshort x);
GLAPI PFNGLVERTEXATTRIB1SPROC glad_glVertexAttrib1s;
#define glVertexAttrib1s glad_glVertexAttrib1s
typedef void (APIENTRYP PFNGLVERTEXATTRIB1SVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB1SVPROC glad_glVertexAttrib1sv;
#define glVertexAttrib1sv glad_glVertexAttrib1sv
typedef void (APIENTRYP PFNGLVERTEXATTRIB2DPROC)(GLuint index, GLdouble x, GLdouble y);
GLAPI PFNGLVERTEXATTRIB2DPROC glad_glVertexAttrib2d;
#define glVertexAttrib2d glad_glVertexAttrib2d
typedef void (APIENTRYP PFNGLVERTEXATTRIB2DVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB2DVPROC glad_glVertexAttrib2dv;
#define glVertexAttrib2dv glad_glVertexAttrib2dv
typedef void (APIENTRYP PFNGLVERTEXATTRIB2FPROC)(GLuint index, GLfloat x, GLfloat y);
GLAPI PFNGLVERTEXATTRIB2FPROC glad_glVertexAttrib2f;
#define glVertexAttrib2f glad_glVertexAttrib2f
typedef void (APIENTRYP PFNGLVERTEXATTRIB2FVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB2FVPROC glad_glVertexAttrib2fv;
#define glVertexAttrib2fv glad_glVertexAttrib2fv
typedef void (APIENTRYP PFNGLVERTEXATTRIB2SPROC)(GLuint index, GLshort x, GLshort y);
GLAPI PFNGLVERTEXATTRIB2SPROC glad_glVertexAttrib2s;
#define glVertexAttrib2s glad_glVertexAttrib2s
typedef void (APIENTRYP PFNGLVERTEXATTRIB2SVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB2SVPROC glad_glVertexAttrib2sv;
#define glVertexAttrib2sv glad_glVertexAttrib2sv
typedef void (APIENTRYP PFNGLVERTEXATTRIB3DPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLVERTEXATTRIB3DPROC glad_glVertexAttrib3d;
#define glVertexAttrib3d glad_glVertexAttrib3d
typedef void (APIENTRYP PFNGLVERTEXATTRIB3DVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB3DVPROC glad_glVertexAttrib3dv;
#define glVertexAttrib3dv glad_glVertexAttrib3dv
typedef void (APIENTRYP PFNGLVERTEXATTRIB3FPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLVERTEXATTRIB3FPROC glad_glVertexAttrib3f;
#define glVertexAttrib3f glad_glVertexAttrib3f
typedef void (APIENTRYP PFNGLVERTEXATTRIB3FVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB3FVPROC glad_glVertexAttrib3fv;
#define glVertexAttrib3fv glad_glVertexAttrib3fv
typedef void (APIENTRYP PFNGLVERTEXATTRIB3SPROC)(GLuint index, GLshort x, GLshort y, GLshort z);
GLAPI PFNGLVERTEXATTRIB3SPROC glad_glVertexAttrib3s;
#define glVertexAttrib3s glad_glVertexAttrib3s
typedef void (APIENTRYP PFNGLVERTEXATTRIB3SVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB3SVPROC glad_glVertexAttrib3sv;
#define glVertexAttrib3sv glad_glVertexAttrib3sv
```

```
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NBVPROC)(GLuint index, const GLbyte *v);
GLAPI PFNGLVERTEXATTRIB4NBVPROC glad_glVertexAttrib4Nbv;
#define glVertexAttrib4Nbv glad_glVertexAttrib4Nbv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NIVPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIB4NIVPROC glad_glVertexAttrib4Niv;
#define glVertexAttrib4Niv glad_glVertexAttrib4Niv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NSVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB4NSVPROC glad_glVertexAttrib4Nsv;
#define glVertexAttrib4Nsv glad_glVertexAttrib4Nsv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUBPROC)(GLuint index, GLubyte x, GLubyte y, GLubyte z, GLubyte w);
GLAPI PFNGLVERTEXATTRIB4NUBPROC glad_glVertexAttrib4Nub;
#define glVertexAttrib4Nub glad_glVertexAttrib4Nub
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUBVPROC)(GLuint index, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIB4NUBVPROC glad_glVertexAttrib4Nubv;
#define glVertexAttrib4Nubv glad_glVertexAttrib4Nubv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUIVPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIB4NUIVPROC glad_glVertexAttrib4Nuiv;
#define glVertexAttrib4Nuiv glad_glVertexAttrib4Nuiv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUSVPROC)(GLuint index, const GLushort *v);
GLAPI PFNGLVERTEXATTRIB4NUSVPROC glad_glVertexAttrib4Nusv;
#define glVertexAttrib4Nusv glad_glVertexAttrib4Nusv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4BVPROC)(GLuint index, const GLbyte *v);
GLAPI PFNGLVERTEXATTRIB4BVPROC glad_glVertexAttrib4bv;
#define glVertexAttrib4bv glad_glVertexAttrib4bv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4DPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLVERTEXATTRIB4DPROC glad_glVertexAttrib4d;
#define glVertexAttrib4d glad_glVertexAttrib4d
typedef void (APIENTRYP PFNGLVERTEXATTRIB4DVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB4DVPROC glad_glVertexAttrib4dv;
#define glVertexAttrib4dv glad_glVertexAttrib4dv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4FPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLVERTEXATTRIB4FPROC glad_glVertexAttrib4f;
#define glVertexAttrib4f glad_glVertexAttrib4f
typedef void (APIENTRYP PFNGLVERTEXATTRIB4FVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB4FVPROC glad_glVertexAttrib4fv;
#define glVertexAttrib4fv glad_glVertexAttrib4fv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4IVPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIB4IVPROC glad_glVertexAttrib4iv;
#define glVertexAttrib4iv glad_glVertexAttrib4iv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4SPROC)(GLuint index, GLshort x, GLshort y, GLshort z, GLshort w);
GLAPI PFNGLVERTEXATTRIB4SPROC glad_glVertexAttrib4s;
#define glVertexAttrib4s glad_glVertexAttrib4s
typedef void (APIENTRYP PFNGLVERTEXATTRIB4SVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB4SVPROC glad_glVertexAttrib4sv;
#define glVertexAttrib4sv glad_glVertexAttrib4sv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4UBVPROC)(GLuint index, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIB4UBVPROC glad_glVertexAttrib4ubv;
#define glVertexAttrib4ubv glad_glVertexAttrib4ubv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4UIVPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIB4UIVPROC glad_glVertexAttrib4uiv;
#define glVertexAttrib4uiv glad_glVertexAttrib4uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIB4USVPROC)(GLuint index, const GLushort *v);
GLAPI PFNGLVERTEXATTRIB4USVPROC glad_glVertexAttrib4usv;
#define glVertexAttrib4usv glad_glVertexAttrib4usv
typedef void (APIENTRYP PFNGLVERTEXATTRIBPOINTERPROC)(GLuint index, GLint size, GLenum type, GLboolean normalized, GLsizei stride, const void *pointer);
GLAPI PFNGLVERTEXATTRIBPOINTERPROC glad_glVertexAttribPointer;
#define glVertexAttribPointer glad_glVertexAttribPointer
#endif
#ifndef GL_VERSION_2_1
#define GL_VERSION_2_1 1
GLAPI int GLAD_GL_VERSION_2_1;
typedef void (APIENTRYP PFNGLUNIFORMMATRIX2X3FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX2X3FVPROC glad_glUniformMatrix2x3fv;
#define glUniformMatrix2x3fv glad_glUniformMatrix2x3fv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX3X2FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX3X2FVPROC glad_glUniformMatrix3x2fv;
#define glUniformMatrix3x2fv glad_glUniformMatrix3x2fv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX2X4FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX2X4FVPROC glad_glUniformMatrix2x4fv;
#define glUniformMatrix2x4fv glad_glUniformMatrix2x4fv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX4X2FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX4X2FVPROC glad_glUniformMatrix4x2fv;
#define glUniformMatrix4x2fv glad_glUniformMatrix4x2fv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX3X4FVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX3X4FVPROC glad_glUniformMatrix3x4fv;
#define glUniformMatrix3x4fv glad_glUniformMatrix3x4fv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX4X3FVPROC)(GLint location, GLsizei count, GLboolean
transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX4X3FVPROC glad_glUniformMatrix4x3fv;
#define glUniformMatrix4x3fv glad_glUniformMatrix4x3fv
#endif
#ifndef GL_VERSION_3_0
#define GL_VERSION_3_0 1
GLAPI int GLAD_GL_VERSION_3_0;
typedef void (APIENTRYP PFNGLCOLORMASKIPROC)(GLuint index, GLboolean r, GLboolean g, GLboolean b, GLboolean a);
GLAPI PFNGLCOLORMASKIPROC glad_glColorMaski;
#define glColorMaski glad_glColorMaski
typedef void (APIENTRYP PFNGLGETBOOLEANI_VPROC)(GLenum target, GLuint index, GLboolean *data);
GLAPI PFNGLGETBOOLEANI_VPROC glad_glGetBooleani_v;
#define glGetBooleani_v glad_glGetBooleani_v
typedef void (APIENTRYP PFNGLGETINTEGERI_VPROC)(GLenum target, GLuint index, GLint *data);
GLAPI PFNGLGETINTEGERI_VPROC glad_glGetIntegeri_v;
#define glGetIntegeri_v glad_glGetIntegeri_v
typedef void (APIENTRYP PFNGLENABLEIPROC)(GLenum target, GLuint index);
GLAPI PFNGLENABLEIPROC glad_glEnablei;
#define glEnablei glad_glEnablei
typedef void (APIENTRYP PFNGLDISABLEIPROC)(GLenum target, GLuint index);
GLAPI PFNGLDISABLEIPROC glad_glDisablei;
#define glDisablei glad_glDisablei
typedef GLboolean (APIENTRYP PFNGLISENABLEDIPROC)(GLenum target, GLuint index);
GLAPI PFNGLISENABLEDIPROC glad_glIsEnabledi;
#define glIsEnabledi glad_glIsEnabledi
typedef void (APIENTRYP PFNGLBEGINTRANSFORMFEEDBACKPROC)(GLenum primitiveMode);
GLAPI PFNGLBEGINTRANSFORMFEEDBACKPROC glad_glBeginTransformFeedback;
#define glBeginTransformFeedback glad_glBeginTransformFeedback
typedef void (APIENTRYP PFNGLENDTRANSFORMFEEDBACKPROC)(void);
GLAPI PFNGLENDTRANSFORMFEEDBACKPROC glad_glEndTransformFeedback;
#define glEndTransformFeedback glad_glEndTransformFeedback
typedef void (APIENTRYP PFNGLBINDBUFFERRANGEPROC)(GLenum target, GLuint index, GLuint buffer, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLBINDBUFFERRANGEPROC glad_glBindBufferRange;
#define glBindBufferRange glad_glBindBufferRange
typedef void (APIENTRYP PFNGLBINDBUFFERBASEPROC)(GLenum target, GLuint index, GLuint buffer);
GLAPI PFNGLBINDBUFFERBASEPROC glad_glBindBufferBase;
#define glBindBufferBase glad_glBindBufferBase
typedef void (APIENTRYP PFNGLTRANSFORMFEEDBACKVARYINGSPROC)(GLuint program, GLsizei count, const GLchar *const*varyings, GLenum bufferMode);
GLAPI PFNGLTRANSFORMFEEDBACKVARYINGSPROC glad_glTransformFeedbackVaryings;
#define glTransformFeedbackVaryings glad_glTransformFeedbackVaryings
typedef void (APIENTRYP PFNGLGETTRANSFORMFEEDBACKVARYINGPROC)(GLuint program, GLuint index, GLsizei bufSize, GLsizei *length, GLsizei *size, GLenum *type, GLchar *name);
GLAPI PFNGLGETTRANSFORMFEEDBACKVARYINGPROC glad_glGetTransformFeedbackVarying;
#define glGetTransformFeedbackVarying glad_glGetTransformFeedbackVarying
typedef void (APIENTRYP PFNGLCLAMPCOLORPROC)(GLenum target, GLenum clamp);
GLAPI PFNGLCLAMPCOLORPROC glad_glClampColor;
#define glClampColor glad_glClampColor
typedef void (APIENTRYP PFNGLBEGINCONDITIONALRENDERPROC)(GLuint id, GLenum mode);
GLAPI PFNGLBEGINCONDITIONALRENDERPROC glad_glBeginConditionalRender;
#define glBeginConditionalRender glad_glBeginConditionalRender
typedef void (APIENTRYP PFNGLENDCONDITIONALRENDERPROC)(void);
GLAPI PFNGLENDCONDITIONALRENDERPROC glad_glEndConditionalRender;
#define glEndConditionalRender glad_glEndConditionalRender
typedef void (APIENTRYP PFNGLVERTEXATTRIBIPOINTERPROC)(GLuint index, GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLVERTEXATTRIBIPOINTERPROC glad_glVertexAttribIPointer;
#define glVertexAttribIPointer glad_glVertexAttribIPointer
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBIIVPROC)(GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETVERTEXATTRIBIIVPROC glad_glGetVertexAttribIiv;
#define glGetVertexAttribIiv glad_glGetVertexAttribIiv
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBIUIVPROC)(GLuint index, GLenum pname, GLuint *params);
GLAPI PFNGLGETVERTEXATTRIBIUIVPROC glad_glGetVertexAttribIuiv;
#define glGetVertexAttribIuiv glad_glGetVertexAttribIuiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1IPROC)(GLuint index, GLint x);
GLAPI PFNGLVERTEXATTRIBI1IPROC glad_glVertexAttribI1i;
#define glVertexAttribI1i glad_glVertexAttribI1i
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2IPROC)(GLuint index, GLint x, GLint y);
GLAPI PFNGLVERTEXATTRIBI2IPROC glad_glVertexAttribI2i;
#define glVertexAttribI2i glad_glVertexAttribI2i
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3IPROC)(GLuint index, GLint x, GLint y, GLint z);
GLAPI PFNGLVERTEXATTRIBI3IPROC glad_glVertexAttribI3i;
#define glVertexAttribI3i glad_glVertexAttribI3i
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4IPROC)(GLuint index, GLint x, GLint y, GLint z, GLint w);
GLAPI PFNGLVERTEXATTRIBI4IPROC glad_glVertexAttribI4i;
#define glVertexAttribI4i glad_glVertexAttribI4i
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1UIPROC)(GLuint index, GLuint x);
GLAPI PFNGLVERTEXATTRIBI1UIPROC glad_glVertexAttribI1ui;
#define glVertexAttribI1ui glad_glVertexAttribI1ui
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2UIPROC)(GLuint index, GLuint x, GLuint y);
GLAPI PFNGLVERTEXATTRIBI2UIPROC glad_glVertexAttribI2ui;
#define glVertexAttribI2ui glad_glVertexAttribI2ui
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3UIPROC)(GLuint index, GLuint x, GLuint y, GLuint z);
GLAPI PFNGLVERTEXATTRIBI3UIPROC glad_glVertexAttribI3ui;
#define glVertexAttribI3ui glad_glVertexAttribI3ui
```

```
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4UIPROC)(GLuint index, GLuint x, GLuint y, GLuint z, GLuint w);
GLAPI PFNGLVERTEXATTRIBI4UIPROC glad_glVertexAttribI4ui;
#define glVertexAttribI4ui glad_glVertexAttribI4ui
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1IVPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIBI1IVPROC glad_glVertexAttribI1iv;
#define glVertexAttribI1iv glad_glVertexAttribI1iv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2IVPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIBI2IVPROC glad_glVertexAttribI2iv;
#define glVertexAttribI2iv glad_glVertexAttribI2iv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3IVPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIBI3IVPROC glad_glVertexAttribI3iv;
#define glVertexAttribI3iv glad_glVertexAttribI3iv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4IVPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIBI4IVPROC glad_glVertexAttribI4iv;
#define glVertexAttribI4iv glad_glVertexAttribI4iv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1UIVPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI1UIVPROC glad_glVertexAttribI1uiv;
#define glVertexAttribI1uiv glad_glVertexAttribI1uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2UIVPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI2UIVPROC glad_glVertexAttribI2uiv;
#define glVertexAttribI2uiv glad_glVertexAttribI2uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3UIVPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI3UIVPROC glad_glVertexAttribI3uiv;
#define glVertexAttribI3uiv glad_glVertexAttribI3uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4UIVPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI4UIVPROC glad_glVertexAttribI4uiv;
#define glVertexAttribI4uiv glad_glVertexAttribI4uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4BVPROC)(GLuint index, const GLbyte *v);
GLAPI PFNGLVERTEXATTRIBI4BVPROC glad_glVertexAttribI4bv;
#define glVertexAttribI4bv glad_glVertexAttribI4bv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4SVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIBI4SVPROC glad_glVertexAttribI4sv;
#define glVertexAttribI4sv glad_glVertexAttribI4sv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4UBVPROC)(GLuint index, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIBI4UBVPROC glad_glVertexAttribI4ubv;
#define glVertexAttribI4ubv glad_glVertexAttribI4ubv
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4USVPROC)(GLuint index, const GLushort *v);
GLAPI PFNGLVERTEXATTRIBI4USVPROC glad_glVertexAttribI4usv;
#define glVertexAttribI4usv glad_glVertexAttribI4usv
typedef void (APIENTRYP PFNGLGETUNIFORMUIVPROC)(GLuint program, GLint location, GLuint *params);
GLAPI PFNGLGETUNIFORMUIVPROC glad_glGetUniformuiv;
#define glGetUniformuiv glad_glGetUniformuiv
typedef void (APIENTRYP PFNGLBINDFRAGDATALOCATIONPROC)(GLuint program, GLuint color, const GLchar *name);
GLAPI PFNGLBINDFRAGDATALOCATIONPROC glad_glBindFragDataLocation;
#define glBindFragDataLocation glad_glBindFragDataLocation
typedef GLint (APIENTRYP PFNGLGETFRAGDATALOCATIONPROC)(GLuint program, const GLchar *name);
GLAPI PFNGLGETFRAGDATALOCATIONPROC glad_glGetFragDataLocation;
#define glGetFragDataLocation glad_glGetFragDataLocation
typedef void (APIENTRYP PFNGLUNIFORM1UIPROC)(GLint location, GLuint v0);
GLAPI PFNGLUNIFORM1UIPROC glad_glUniform1ui;
#define glUniform1ui glad_glUniform1ui
typedef void (APIENTRYP PFNGLUNIFORM2UIPROC)(GLint location, GLuint v0, GLuint v1);
GLAPI PFNGLUNIFORM2UIPROC glad_glUniform2ui;
#define glUniform2ui glad_glUniform2ui
typedef void (APIENTRYP PFNGLUNIFORM3UIPROC)(GLint location, GLuint v0, GLuint v1, GLuint v2);
GLAPI PFNGLUNIFORM3UIPROC glad_glUniform3ui;
#define glUniform3ui glad_glUniform3ui
typedef void (APIENTRYP PFNGLUNIFORM4UIPROC)(GLint location, GLuint v0, GLuint v1, GLuint v2, GLuint v3);
GLAPI PFNGLUNIFORM4UIPROC glad_glUniform4ui;
#define glUniform4ui glad_glUniform4ui
typedef void (APIENTRYP PFNGLUNIFORM1UIVPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM1UIVPROC glad_glUniform1uiv;
#define glUniform1uiv glad_glUniform1uiv
typedef void (APIENTRYP PFNGLUNIFORM2UIVPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM2UIVPROC glad_glUniform2uiv;
#define glUniform2uiv glad_glUniform2uiv
typedef void (APIENTRYP PFNGLUNIFORM3UIVPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM3UIVPROC glad_glUniform3uiv;
#define glUniform3uiv glad_glUniform3uiv
typedef void (APIENTRYP PFNGLUNIFORM4UIVPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM4UIVPROC glad_glUniform4uiv;
#define glUniform4uiv glad_glUniform4uiv
typedef void (APIENTRYP PFNGLTEXPARAMETERIIVPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLTEXPARAMETERIIVPROC glad_glTexParameterIiv;
#define glTexParameterIiv glad_glTexParameterIiv
typedef void (APIENTRYP PFNGLTEXPARAMETERIUIVPROC)(GLenum target, GLenum pname, const GLuint *params);
GLAPI PFNGLTEXPARAMETERIUIVPROC glad_glTexParameterIuiv;
#define glTexParameterIuiv glad_glTexParameterIuiv
typedef void (APIENTRYP PFNGLGETTEXPARAMETERIIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXPARAMETERIIVPROC glad_glGetTexParameterIiv;
#define glGetTexParameterIiv glad_glGetTexParameterIiv
typedef void (APIENTRYP PFNGLGETTEXPARAMETERIUIVPROC)(GLenum target, GLenum pname, GLuint *params);
GLAPI PFNGLGETTEXPARAMETERIUIVPROC glad_glGetTexParameterIuiv;
#define glGetTexParameterIuiv glad_glGetTexParameterIuiv
```

```
typedef void (APIENTRYP PFNGLCLEARBUFFERIVPROC)(GLenum buffer, GLint drawbuffer, const GLint *value);
GLAPI PFNGLCLEARBUFFERIVPROC glad_glClearBufferiv;
#define glClearBufferiv glad_glClearBufferiv
typedef void (APIENTRYP PFNGLCLEARBUFFERUIVPROC)(GLenum buffer, GLint drawbuffer, const GLuint *value);
GLAPI PFNGLCLEARBUFFERUIVPROC glad_glClearBufferuiv;
#define glClearBufferuiv glad_glClearBufferuiv
typedef void (APIENTRYP PFNGLCLEARBUFFERFVPROC)(GLenum buffer, GLint drawbuffer, const GLfloat *value);
GLAPI PFNGLCLEARBUFFERFVPROC glad_glClearBufferfv;
#define glClearBufferfv glad_glClearBufferfv
typedef void (APIENTRYP PFNGLCLEARBUFFERFIPROC)(GLenum buffer, GLint drawbuffer, GLfloat depth, GLint stencil);
GLAPI PFNGLCLEARBUFFERFIPROC glad_glClearBufferfi;
#define glClearBufferfi glad_glClearBufferfi
typedef const GLubyte * (APIENTRYP PFNGLGETSTRINGIPROC)(GLenum name, GLuint index);
GLAPI PFNGLGETSTRINGIPROC glad_glGetStringi;
#define glGetStringi glad_glGetStringi
typedef GLboolean (APIENTRYP PFNGLISRENDERBUFFERPROC)(GLuint renderbuffer);
GLAPI PFNGLISRENDERBUFFERPROC glad_glIsRenderbuffer;
#define glIsRenderbuffer glad_glIsRenderbuffer
typedef void (APIENTRYP PFNGLBINDRENDERBUFFERPROC)(GLenum target, GLuint renderbuffer);
GLAPI PFNGLBINDRENDERBUFFERPROC glad_glBindRenderbuffer;
#define glBindRenderbuffer glad_glBindRenderbuffer
typedef void (APIENTRYP PFNGLDELETERENDERBUFFERSPROC)(GLsizei n, const GLuint *renderbuffers);
GLAPI PFNGLDELETERENDERBUFFERSPROC glad_glDeleteRenderbuffers;
#define glDeleteRenderbuffers glad_glDeleteRenderbuffers
typedef void (APIENTRYP PFNGLGENRENDERBUFFERSPROC)(GLsizei n, GLuint *renderbuffers);
GLAPI PFNGLGENRENDERBUFFERSPROC glad_glGenRenderbuffers;
#define glGenRenderbuffers glad_glGenRenderbuffers
typedef void (APIENTRYP PFNGLRENDERBUFFERSTORAGEPROC)(GLenum target, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLRENDERBUFFERSTORAGEPROC glad_glRenderbufferStorage;
#define glRenderbufferStorage glad_glRenderbufferStorage
typedef void (APIENTRYP PFNGLGETRENDERBUFFERPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETRENDERBUFFERPARAMETERIVPROC glad_glGetRenderbufferParameteriv;
#define glGetRenderbufferParameteriv glad_glGetRenderbufferParameteriv
typedef GLboolean (APIENTRYP PFNGLISFRAMEBUFFERPROC)(GLuint framebuffer);
GLAPI PFNGLISFRAMEBUFFERPROC glad_glIsFramebuffer;
#define glIsFramebuffer glad_glIsFramebuffer
typedef void (APIENTRYP PFNGLBINDFRAMEBUFFERPROC)(GLenum target, GLuint framebuffer);
GLAPI PFNGLBINDFRAMEBUFFERPROC glad_glBindFramebuffer;
#define glBindFramebuffer glad_glBindFramebuffer
typedef void (APIENTRYP PFNGLDELETEFRAMEBUFFERSPROC)(GLsizei n, const GLuint *framebuffers);
GLAPI PFNGLDELETEFRAMEBUFFERSPROC glad_glDeleteFramebuffers;
#define glDeleteFramebuffers glad_glDeleteFramebuffers
typedef void (APIENTRYP PFNGLGENFRAMEBUFFERSPROC)(GLsizei n, GLuint *framebuffers);
GLAPI PFNGLGENFRAMEBUFFERSPROC glad_glGenFramebuffers;
#define glGenFramebuffers glad_glGenFramebuffers
typedef GLenum (APIENTRYP PFNGLCHECKFRAMEBUFFERSTATUSPROC)(GLenum target);
GLAPI PFNGLCHECKFRAMEBUFFERSTATUSPROC glad_glCheckFramebufferStatus;
#define glCheckFramebufferStatus glad_glCheckFramebufferStatus
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURE1DPROC)(GLenum target, GLenum attachment, GLenum textarget, GLuint texture, GLint level);
GLAPI PFNGLFRAMEBUFFERTEXTURE1DPROC glad_glFramebufferTexture1D;
#define glFramebufferTexture1D glad_glFramebufferTexture1D
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURE2DPROC)(GLenum target, GLenum attachment, GLenum textarget, GLuint texture, GLint level);
GLAPI PFNGLFRAMEBUFFERTEXTURE2DPROC glad_glFramebufferTexture2D;
#define glFramebufferTexture2D glad_glFramebufferTexture2D
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURE3DPROC)(GLenum target, GLenum attachment, GLenum textarget, GLuint texture, GLint level, GLint zoffset);
GLAPI PFNGLFRAMEBUFFERTEXTURE3DPROC glad_glFramebufferTexture3D;
#define glFramebufferTexture3D glad_glFramebufferTexture3D
typedef void (APIENTRYP PFNGLFRAMEBUFFERRENDERBUFFERPROC)(GLenum target, GLenum attachment, GLenum renderbuffertarget, GLuint renderbuffer);
GLAPI PFNGLFRAMEBUFFERRENDERBUFFERPROC glad_glFramebufferRenderbuffer;
#define glFramebufferRenderbuffer glad_glFramebufferRenderbuffer
typedef void (APIENTRYP PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVPROC)(GLenum target, GLenum attachment, GLenum pname, GLint *params);
GLAPI PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVPROC glad_glGetFramebufferAttachmentParameteriv;
#define glGetFramebufferAttachmentParameteriv glad_glGetFramebufferAttachmentParameteriv
typedef void (APIENTRYP PFNGLGENERATEMIPMAPPROC)(GLenum target);
GLAPI PFNGLGENERATEMIPMAPPROC glad_glGenerateMipmap;
#define glGenerateMipmap glad_glGenerateMipmap
typedef void (APIENTRYP PFNGLBLITFRAMEBUFFERPROC)(GLint srcX0, GLint srcY0, GLint srcX1, GLint srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLbitfield mask, GLenum filter);
GLAPI PFNGLBLITFRAMEBUFFERPROC glad_glBlitFramebuffer;
#define glBlitFramebuffer glad_glBlitFramebuffer
typedef void (APIENTRYP PFNGLRENDERBUFFERSTORAGEMULTISAMPLEPROC)(GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLRENDERBUFFERSTORAGEMULTISAMPLEPROC glad_glRenderbufferStorageMultisample;
#define glRenderbufferStorageMultisample glad_glRenderbufferStorageMultisample
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURELAYERPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level, GLint layer);
GLAPI PFNGLFRAMEBUFFERTEXTURELAYERPROC glad_glFramebufferTextureLayer;
```

```
#define glFramebufferTextureLayer glad_glFramebufferTextureLayer
typedef void * (APIENTRYP PFNGLMAPBUFFERRANGEPROC)(GLenum target, GLintptr offset, GLsizeiptr length, GLbitfield access);
GLAPI PFNGLMAPBUFFERRANGEPROC glad_glMapBufferRange;
#define glMapBufferRange glad_glMapBufferRange
typedef void (APIENTRYP PFNGLFLUSHMAPPEDBUFFERRANGEPROC)(GLenum target, GLintptr offset, GLsizeiptr length);
GLAPI PFNGLFLUSHMAPPEDBUFFERRANGEPROC glad_glFlushMappedBufferRange;
#define glFlushMappedBufferRange glad_glFlushMappedBufferRange
typedef void (APIENTRYP PFNGLBINDVERTEXARRAYPROC)(GLuint array);
GLAPI PFNGLBINDVERTEXARRAYPROC glad_glBindVertexArray;
#define glBindVertexArray glad_glBindVertexArray
typedef void (APIENTRYP PFNGLDELETEVERTEXARRAYSPROC)(GLsizei n, const GLuint *arrays);
GLAPI PFNGLDELETEVERTEXARRAYSPROC glad_glDeleteVertexArrays;
#define glDeleteVertexArrays glad_glDeleteVertexArrays
typedef void (APIENTRYP PFNGLGENVERTEXARRAYSPROC)(GLsizei n, GLuint *arrays);
GLAPI PFNGLGENVERTEXARRAYSPROC glad_glGenVertexArrays;
#define glGenVertexArrays glad_glGenVertexArrays
typedef GLboolean (APIENTRYP PFNGLISVERTEXARRAYPROC)(GLuint array);
GLAPI PFNGLISVERTEXARRAYPROC glad_glIsVertexArray;
#define glIsVertexArray glad_glIsVertexArray
#endif
#ifndef GL_VERSION_3_1
#define GL_VERSION_3_1 1
GLAPI int GLAD_GL_VERSION_3_1;
typedef void (APIENTRYP PFNGLDRAWARRAYSINSTANCEDPROC)(GLenum mode, GLint first, GLsizei count, GLsizei instancecount);
GLAPI PFNGLDRAWARRAYSINSTANCEDPROC glad_glDrawArraysInstanced;
#define glDrawArraysInstanced glad_glDrawArraysInstanced
typedef void (APIENTRYP PFNGLDRAWELEMENTSINSTANCEDPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices, GLsizei instancecount);
GLAPI PFNGLDRAWELEMENTSINSTANCEDPROC glad_glDrawElementsInstanced;
#define glDrawElementsInstanced glad_glDrawElementsInstanced
typedef void (APIENTRYP PFNGLTEXBUFFERPROC)(GLenum target, GLenum internalformat, GLuint buffer);
GLAPI PFNGLTEXBUFFERPROC glad_glTexBuffer;
#define glTexBuffer glad_glTexBuffer
typedef void (APIENTRYP PFNGLPRIMITIVERESTARTINDEXPROC)(GLuint index);
GLAPI PFNGLPRIMITIVERESTARTINDEXPROC glad_glPrimitiveRestartIndex;
#define glPrimitiveRestartIndex glad_glPrimitiveRestartIndex
typedef void (APIENTRYP PFNGLCOPYBUFFERSUBDATAPROC)(GLenum readTarget, GLenum writeTarget, GLintptr readOffset, GLintptr writeOffset, GLsizeiptr size);
GLAPI PFNGLCOPYBUFFERSUBDATAPROC glad_glCopyBufferSubData;
#define glCopyBufferSubData glad_glCopyBufferSubData
typedef void (APIENTRYP PFNGLGETUNIFORMINDICESPROC)(GLuint program, GLsizei uniformCount, const GLchar *const*uniformNames, GLuint *uniformIndices);
GLAPI PFNGLGETUNIFORMINDICESPROC glad_glGetUniformIndices;
#define glGetUniformIndices glad_glGetUniformIndices
typedef void (APIENTRYP PFNGLGETACTIVEUNIFORMSIVPROC)(GLuint program, GLsizei uniformCount, const GLuint *uniformIndices, GLenum pname, GLint *params);
GLAPI PFNGLGETACTIVEUNIFORMSIVPROC glad_glGetActiveUniformsiv;
#define glGetActiveUniformsiv glad_glGetActiveUniformsiv
typedef void (APIENTRYP PFNGLGETACTIVEUNIFORMNAMEPROC)(GLuint program, GLuint uniformIndex, GLsizei bufSize, GLsizei *length, GLchar *uniformName);
GLAPI PFNGLGETACTIVEUNIFORMNAMEPROC glad_glGetActiveUniformName;
#define glGetActiveUniformName glad_glGetActiveUniformName
typedef GLuint (APIENTRYP PFNGLGETUNIFORMBLOCKINDEXPROC)(GLuint program, const GLchar *uniformBlockName);
GLAPI PFNGLGETUNIFORMBLOCKINDEXPROC glad_glGetUniformBlockIndex;
#define glGetUniformBlockIndex glad_glGetUniformBlockIndex
typedef void (APIENTRYP PFNGLGETACTIVEUNIFORMBLOCKIVPROC)(GLuint program, GLuint uniformBlockIndex, GLenum pname, GLint *params);
GLAPI PFNGLGETACTIVEUNIFORMBLOCKIVPROC glad_glGetActiveUniformBlockiv;
#define glGetActiveUniformBlockiv glad_glGetActiveUniformBlockiv
typedef void (APIENTRYP PFNGLGETACTIVEUNIFORMBLOCKNAMEPROC)(GLuint program, GLuint uniformBlockIndex, GLsizei bufSize, GLsizei *length, GLchar *uniformBlockName);
GLAPI PFNGLGETACTIVEUNIFORMBLOCKNAMEPROC glad_glGetActiveUniformBlockName;
#define glGetActiveUniformBlockName glad_glGetActiveUniformBlockName
typedef void (APIENTRYP PFNGLUNIFORMBLOCKBINDINGPROC)(GLuint program, GLuint uniformBlockIndex, GLuint uniformBlockBinding);
GLAPI PFNGLUNIFORMBLOCKBINDINGPROC glad_glUniformBlockBinding;
#define glUniformBlockBinding glad_glUniformBlockBinding
#endif
#ifndef GL_VERSION_3_2
#define GL_VERSION_3_2 1
GLAPI int GLAD_GL_VERSION_3_2;
typedef void (APIENTRYP PFNGLDRAWELEMENTSBASEVERTEXPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices, GLint basevertex);
GLAPI PFNGLDRAWELEMENTSBASEVERTEXPROC glad_glDrawElementsBaseVertex;
#define glDrawElementsBaseVertex glad_glDrawElementsBaseVertex
typedef void (APIENTRYP PFNGLDRAWRANGEELEMENTSBASEVERTEXPROC)(GLenum mode, GLuint start, GLuint end, GLsizei count, GLenum type, const void *indices, GLint basevertex);
GLAPI PFNGLDRAWRANGEELEMENTSBASEVERTEXPROC glad_glDrawRangeElementsBaseVertex;
#define glDrawRangeElementsBaseVertex glad_glDrawRangeElementsBaseVertex
typedef void (APIENTRYP PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices, GLsizei instancecount, GLint basevertex);
GLAPI PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXPROC glad_glDrawElementsInstancedBaseVertex;
#define glDrawElementsInstancedBaseVertex glad_glDrawElementsInstancedBaseVertex
```

```
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSBASEVERTEXPROC)(GLenum mode, const GLsizei *count, GLenum type, const void *const*indices, GLsizei drawcount, const GLint *basevertex);
GLAPI PFNGLMULTIDRAWELEMENTSBASEVERTEXPROC glad_glMultiDrawElementsBaseVertex;
#define glMultiDrawElementsBaseVertex glad_glMultiDrawElementsBaseVertex
typedef void (APIENTRYP PFNGLPROVOKINGVERTEXPROC)(GLenum mode);
GLAPI PFNGLPROVOKINGVERTEXPROC glad_glProvokingVertex;
#define glProvokingVertex glad_glProvokingVertex
typedef GLsync (APIENTRYP PFNGLFENCESYNCPROC)(GLenum condition, GLbitfield flags);
GLAPI PFNGLFENCESYNCPROC glad_glFenceSync;
#define glFenceSync glad_glFenceSync
typedef GLboolean (APIENTRYP PFNGLISSYNCPROC)(GLsync sync);
GLAPI PFNGLISSYNCPROC glad_glIsSync;
#define glIsSync glad_glIsSync
typedef void (APIENTRYP PFNGLDELETESYNCPROC)(GLsync sync);
GLAPI PFNGLDELETESYNCPROC glad_glDeleteSync;
#define glDeleteSync glad_glDeleteSync
typedef GLenum (APIENTRYP PFNGLCLIENTWAITSYNCPROC)(GLsync sync, GLbitfield flags, GLuint64 timeout);
GLAPI PFNGLCLIENTWAITSYNCPROC glad_glClientWaitSync;
#define glClientWaitSync glad_glClientWaitSync
typedef void (APIENTRYP PFNGLWAITSYNCPROC)(GLsync sync, GLbitfield flags, GLuint64 timeout);
GLAPI PFNGLWAITSYNCPROC glad_glWaitSync;
#define glWaitSync glad_glWaitSync
typedef void (APIENTRYP PFNGLGETINTEGER64VPROC)(GLenum pname, GLint64 *data);
GLAPI PFNGLGETINTEGER64VPROC glad_glGetInteger64v;
#define glGetInteger64v glad_glGetInteger64v
typedef void (APIENTRYP PFNGLGETSYNCIVPROC)(GLsync sync, GLenum pname, GLsizei count, GLsizei *length, GLint *values);
GLAPI PFNGLGETSYNCIVPROC glad_glGetSynciv;
#define glGetSynciv glad_glGetSynciv
typedef void (APIENTRYP PFNGLGETINTEGER64I_VPROC)(GLenum target, GLuint index, GLint64 *data);
GLAPI PFNGLGETINTEGER64I_VPROC glad_glGetInteger64i_v;
#define glGetInteger64i_v glad_glGetInteger64i_v
typedef void (APIENTRYP PFNGLGETBUFFERPARAMETERI64VPROC)(GLenum target, GLenum pname, GLint64 *params);
GLAPI PFNGLGETBUFFERPARAMETERI64VPROC glad_glGetBufferParameteri64v;
#define glGetBufferParameteri64v glad_glGetBufferParameteri64v
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTUREPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level);
GLAPI PFNGLFRAMEBUFFERTEXTUREPROC glad_glFramebufferTexture;
#define glFramebufferTexture glad_glFramebufferTexture
typedef void (APIENTRYP PFNGLTEXIMAGE2DMULTISAMPLEPROC)(GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXIMAGE2DMULTISAMPLEPROC glad_glTexImage2DMultisample;
#define glTexImage2DMultisample glad_glTexImage2DMultisample
typedef void (APIENTRYP PFNGLTEXIMAGE3DMULTISAMPLEPROC)(GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXIMAGE3DMULTISAMPLEPROC glad_glTexImage3DMultisample;
#define glTexImage3DMultisample glad_glTexImage3DMultisample
typedef void (APIENTRYP PFNGLGETMULTISAMPLEFVPROC)(GLenum pname, GLuint index, GLfloat *val);
GLAPI PFNGLGETMULTISAMPLEFVPROC glad_glGetMultisamplefv;
#define glGetMultisamplefv glad_glGetMultisamplefv
typedef void (APIENTRYP PFNGLSAMPLEMASKIPROC)(GLuint maskNumber, GLbitfield mask);
GLAPI PFNGLSAMPLEMASKIPROC glad_glSampleMaski;
#define glSampleMaski glad_glSampleMaski
#endif
#ifndef GL_VERSION_3_3
#define GL_VERSION_3_3 1
GLAPI int GLAD_GL_VERSION_3_3;
typedef void (APIENTRYP PFNGLBINDFRAGDATALOCATIONINDEXEDPROC)(GLuint program, GLuint colorNumber, GLuint index, const GLchar *name);
GLAPI PFNGLBINDFRAGDATALOCATIONINDEXEDPROC glad_glBindFragDataLocationIndexed;
#define glBindFragDataLocationIndexed glad_glBindFragDataLocationIndexed
typedef GLint (APIENTRYP PFNGLGETFRAGDATAINDEXPROC)(GLuint program, const GLchar *name);
GLAPI PFNGLGETFRAGDATAINDEXPROC glad_glGetFragDataIndex;
#define glGetFragDataIndex glad_glGetFragDataIndex
typedef void (APIENTRYP PFNGLGENSAMPLERSPROC)(GLsizei count, GLuint *samplers);
GLAPI PFNGLGENSAMPLERSPROC glad_glGenSamplers;
#define glGenSamplers glad_glGenSamplers
typedef void (APIENTRYP PFNGLDELETESAMPLERSPROC)(GLsizei count, const GLuint *samplers);
GLAPI PFNGLDELETESAMPLERSPROC glad_glDeleteSamplers;
#define glDeleteSamplers glad_glDeleteSamplers
typedef GLboolean (APIENTRYP PFNGLISSAMPLERPROC)(GLuint sampler);
GLAPI PFNGLISSAMPLERPROC glad_glIsSampler;
#define glIsSampler glad_glIsSampler
typedef void (APIENTRYP PFNGLBINDSAMPLERPROC)(GLuint unit, GLuint sampler);
GLAPI PFNGLBINDSAMPLERPROC glad_glBindSampler;
#define glBindSampler glad_glBindSampler
typedef void (APIENTRYP PFNGLSAMPLERPARAMETERIPROC)(GLuint sampler, GLenum pname, GLint param);
GLAPI PFNGLSAMPLERPARAMETERIPROC glad_glSamplerParameteri;
#define glSamplerParameteri glad_glSamplerParameteri
typedef void (APIENTRYP PFNGLSAMPLERPARAMETERIVPROC)(GLuint sampler, GLenum pname, const GLint *param);
GLAPI PFNGLSAMPLERPARAMETERIVPROC glad_glSamplerParameteriv;
#define glSamplerParameteriv glad_glSamplerParameteriv
typedef void (APIENTRYP PFNGLSAMPLERPARAMETERFPROC)(GLuint sampler, GLenum pname, GLfloat param);
GLAPI PFNGLSAMPLERPARAMETERFPROC glad_glSamplerParameterf;
#define glSamplerParameterf glad_glSamplerParameterf
```

```
typedef void (APIENTRYP PFNGLSAMPLERPARAMETERFVPROC)(GLuint sampler, GLenum pname, const GLfloat *param);
GLAPI PFNGLSAMPLERPARAMETERFVPROC glad_glSamplerParameterfv;
#define glSamplerParameterfv glad_glSamplerParameterfv
typedef void (APIENTRYP PFNGLSAMPLERPARAMETERIIVPROC)(GLuint sampler, GLenum pname, const GLint *param);
GLAPI PFNGLSAMPLERPARAMETERIIVPROC glad_glSamplerParameterIiv;
#define glSamplerParameterIiv glad_glSamplerParameterIiv
typedef void (APIENTRYP PFNGLSAMPLERPARAMETERIUIVPROC)(GLuint sampler, GLenum pname, const GLuint *param);
GLAPI PFNGLSAMPLERPARAMETERIUIVPROC glad_glSamplerParameterIuiv;
#define glSamplerParameterIuiv glad_glSamplerParameterIuiv
typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERIVPROC)(GLuint sampler, GLenum pname, GLint *params);
GLAPI PFNGLGETSAMPLERPARAMETERIVPROC glad_glGetSamplerParameteriv;
#define glGetSamplerParameteriv glad_glGetSamplerParameteriv
typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERIIVPROC)(GLuint sampler, GLenum pname, GLint *params);
GLAPI PFNGLGETSAMPLERPARAMETERIIVPROC glad_glGetSamplerParameterIiv;
#define glGetSamplerParameterIiv glad_glGetSamplerParameterIiv
typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERFVPROC)(GLuint sampler, GLenum pname, GLfloat *params);
GLAPI PFNGLGETSAMPLERPARAMETERFVPROC glad_glGetSamplerParameterfv;
#define glGetSamplerParameterfv glad_glGetSamplerParameterfv
typedef void (APIENTRYP PFNGLGETSAMPLERPARAMETERIUIVPROC)(GLuint sampler, GLenum pname, GLuint *params);
GLAPI PFNGLGETSAMPLERPARAMETERIUIVPROC glad_glGetSamplerParameterIuiv;
#define glGetSamplerParameterIuiv glad_glGetSamplerParameterIuiv
typedef void (APIENTRYP PFNGLQUERYCOUNTERPROC)(GLuint id, GLenum target);
GLAPI PFNGLQUERYCOUNTERPROC glad_glQueryCounter;
#define glQueryCounter glad_glQueryCounter
typedef void (APIENTRYP PFNGLGETQUERYOBJECTI64VPROC)(GLuint id, GLenum pname, GLint64 *params);
GLAPI PFNGLGETQUERYOBJECTI64VPROC glad_glGetQueryObjecti64v;
#define glGetQueryObjecti64v glad_glGetQueryObjecti64v
typedef void (APIENTRYP PFNGLGETQUERYOBJECTUI64VPROC)(GLuint id, GLenum pname, GLuint64 *params);
GLAPI PFNGLGETQUERYOBJECTUI64VPROC glad_glGetQueryObjectui64v;
#define glGetQueryObjectui64v glad_glGetQueryObjectui64v
typedef void (APIENTRYP PFNGLVERTEXATTRIBDIVISORPROC)(GLuint index, GLuint divisor);
GLAPI PFNGLVERTEXATTRIBDIVISORPROC glad_glVertexAttribDivisor;
#define glVertexAttribDivisor glad_glVertexAttribDivisor
typedef void (APIENTRYP PFNGLVERTEXATTRIBP1UIPROC)(GLuint index, GLenum type, GLboolean normalized, GLuint value);
GLAPI PFNGLVERTEXATTRIBP1UIPROC glad_glVertexAttribP1ui;
#define glVertexAttribP1ui glad_glVertexAttribP1ui
typedef void (APIENTRYP PFNGLVERTEXATTRIBP1UIVPROC)(GLuint index, GLenum type, GLboolean normalized, const GLuint *value);
GLAPI PFNGLVERTEXATTRIBP1UIVPROC glad_glVertexAttribP1uiv;
#define glVertexAttribP1uiv glad_glVertexAttribP1uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBP2UIPROC)(GLuint index, GLenum type, GLboolean normalized, GLuint value);
GLAPI PFNGLVERTEXATTRIBP2UIPROC glad_glVertexAttribP2ui;
#define glVertexAttribP2ui glad_glVertexAttribP2ui
typedef void (APIENTRYP PFNGLVERTEXATTRIBP2UIVPROC)(GLuint index, GLenum type, GLboolean normalized, const GLuint *value);
GLAPI PFNGLVERTEXATTRIBP2UIVPROC glad_glVertexAttribP2uiv;
#define glVertexAttribP2uiv glad_glVertexAttribP2uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBP3UIPROC)(GLuint index, GLenum type, GLboolean normalized, GLuint value);
GLAPI PFNGLVERTEXATTRIBP3UIPROC glad_glVertexAttribP3ui;
#define glVertexAttribP3ui glad_glVertexAttribP3ui
typedef void (APIENTRYP PFNGLVERTEXATTRIBP3UIVPROC)(GLuint index, GLenum type, GLboolean normalized, const GLuint *value);
GLAPI PFNGLVERTEXATTRIBP3UIVPROC glad_glVertexAttribP3uiv;
#define glVertexAttribP3uiv glad_glVertexAttribP3uiv
typedef void (APIENTRYP PFNGLVERTEXATTRIBP4UIPROC)(GLuint index, GLenum type, GLboolean normalized, GLuint value);
GLAPI PFNGLVERTEXATTRIBP4UIPROC glad_glVertexAttribP4ui;
#define glVertexAttribP4ui glad_glVertexAttribP4ui
typedef void (APIENTRYP PFNGLVERTEXATTRIBP4UIVPROC)(GLuint index, GLenum type, GLboolean normalized, const GLuint *value);
GLAPI PFNGLVERTEXATTRIBP4UIVPROC glad_glVertexAttribP4uiv;
#define glVertexAttribP4uiv glad_glVertexAttribP4uiv
typedef void (APIENTRYP PFNGLVERTEXP2UIPROC)(GLenum type, GLuint value);
GLAPI PFNGLVERTEXP2UIPROC glad_glVertexP2ui;
#define glVertexP2ui glad_glVertexP2ui
typedef void (APIENTRYP PFNGLVERTEXP2UIVPROC)(GLenum type, const GLuint *value);
GLAPI PFNGLVERTEXP2UIVPROC glad_glVertexP2uiv;
#define glVertexP2uiv glad_glVertexP2uiv
typedef void (APIENTRYP PFNGLVERTEXP3UIPROC)(GLenum type, GLuint value);
GLAPI PFNGLVERTEXP3UIPROC glad_glVertexP3ui;
#define glVertexP3ui glad_glVertexP3ui
typedef void (APIENTRYP PFNGLVERTEXP3UIVPROC)(GLenum type, const GLuint *value);
GLAPI PFNGLVERTEXP3UIVPROC glad_glVertexP3uiv;
#define glVertexP3uiv glad_glVertexP3uiv
typedef void (APIENTRYP PFNGLVERTEXP4UIPROC)(GLenum type, GLuint value);
GLAPI PFNGLVERTEXP4UIPROC glad_glVertexP4ui;
#define glVertexP4ui glad_glVertexP4ui
typedef void (APIENTRYP PFNGLVERTEXP4UIVPROC)(GLenum type, const GLuint *value);
GLAPI PFNGLVERTEXP4UIVPROC glad_glVertexP4uiv;
```

```
#define glVertexP4uiv glad_glVertexP4uiv
typedef void (APIENTRYP PFNGLTEXCOORDP1UIPROC)(GLenum type, GLuint coords);
GLAPI PFNGLTEXCOORDP1UIPROC glad_glTexCoordP1ui;
#define glTexCoordP1ui glad_glTexCoordP1ui
typedef void (APIENTRYP PFNGLTEXCOORDP1UIVPROC)(GLenum type, const GLuint *coords);
GLAPI PFNGLTEXCOORDP1UIVPROC glad_glTexCoordP1uiv;
#define glTexCoordP1uiv glad_glTexCoordP1uiv
typedef void (APIENTRYP PFNGLTEXCOORDP2UIPROC)(GLenum type, GLuint coords);
GLAPI PFNGLTEXCOORDP2UIPROC glad_glTexCoordP2ui;
#define glTexCoordP2ui glad_glTexCoordP2ui
typedef void (APIENTRYP PFNGLTEXCOORDP2UIVPROC)(GLenum type, const GLuint *coords);
GLAPI PFNGLTEXCOORDP2UIVPROC glad_glTexCoordP2uiv;
#define glTexCoordP2uiv glad_glTexCoordP2uiv
typedef void (APIENTRYP PFNGLTEXCOORDP3UIPROC)(GLenum type, GLuint coords);
GLAPI PFNGLTEXCOORDP3UIPROC glad_glTexCoordP3ui;
#define glTexCoordP3ui glad_glTexCoordP3ui
typedef void (APIENTRYP PFNGLTEXCOORDP3UIVPROC)(GLenum type, const GLuint *coords);
GLAPI PFNGLTEXCOORDP3UIVPROC glad_glTexCoordP3uiv;
#define glTexCoordP3uiv glad_glTexCoordP3uiv
typedef void (APIENTRYP PFNGLTEXCOORDP4UIPROC)(GLenum type, GLuint coords);
GLAPI PFNGLTEXCOORDP4UIPROC glad_glTexCoordP4ui;
#define glTexCoordP4ui glad_glTexCoordP4ui
typedef void (APIENTRYP PFNGLTEXCOORDP4UIVPROC)(GLenum type, const GLuint *coords);
GLAPI PFNGLTEXCOORDP4UIVPROC glad_glTexCoordP4uiv;
#define glTexCoordP4uiv glad_glTexCoordP4uiv
typedef void (APIENTRYP PFNGLMULTITEXCOORDP1UIPROC)(GLenum texture, GLenum type, GLuint coords);
GLAPI PFNGLMULTITEXCOORDP1UIPROC glad_glMultiTexCoordP1ui;
#define glMultiTexCoordP1ui glad_glMultiTexCoordP1ui
typedef void (APIENTRYP PFNGLMULTITEXCOORDP1UIVPROC)(GLenum texture, GLenum type, const GLuint *coords);
GLAPI PFNGLMULTITEXCOORDP1UIVPROC glad_glMultiTexCoordP1uiv;
#define glMultiTexCoordP1uiv glad_glMultiTexCoordP1uiv
typedef void (APIENTRYP PFNGLMULTITEXCOORDP2UIPROC)(GLenum texture, GLenum type, GLuint coords);
GLAPI PFNGLMULTITEXCOORDP2UIPROC glad_glMultiTexCoordP2ui;
#define glMultiTexCoordP2ui glad_glMultiTexCoordP2ui
typedef void (APIENTRYP PFNGLMULTITEXCOORDP2UIVPROC)(GLenum texture, GLenum type, const GLuint *coords);
GLAPI PFNGLMULTITEXCOORDP2UIVPROC glad_glMultiTexCoordP2uiv;
#define glMultiTexCoordP2uiv glad_glMultiTexCoordP2uiv
typedef void (APIENTRYP PFNGLMULTITEXCOORDP3UIPROC)(GLenum texture, GLenum type, GLuint coords);
GLAPI PFNGLMULTITEXCOORDP3UIPROC glad_glMultiTexCoordP3ui;
#define glMultiTexCoordP3ui glad_glMultiTexCoordP3ui
typedef void (APIENTRYP PFNGLMULTITEXCOORDP3UIVPROC)(GLenum texture, GLenum type, const GLuint *coords);
GLAPI PFNGLMULTITEXCOORDP3UIVPROC glad_glMultiTexCoordP3uiv;
#define glMultiTexCoordP3uiv glad_glMultiTexCoordP3uiv
typedef void (APIENTRYP PFNGLMULTITEXCOORDP4UIPROC)(GLenum texture, GLenum type, GLuint coords);
GLAPI PFNGLMULTITEXCOORDP4UIPROC glad_glMultiTexCoordP4ui;
#define glMultiTexCoordP4ui glad_glMultiTexCoordP4ui
typedef void (APIENTRYP PFNGLMULTITEXCOORDP4UIVPROC)(GLenum texture, GLenum type, const GLuint *coords);
GLAPI PFNGLMULTITEXCOORDP4UIVPROC glad_glMultiTexCoordP4uiv;
#define glMultiTexCoordP4uiv glad_glMultiTexCoordP4uiv
typedef void (APIENTRYP PFNGLNORMALP3UIPROC)(GLenum type, GLuint coords);
GLAPI PFNGLNORMALP3UIPROC glad_glNormalP3ui;
#define glNormalP3ui glad_glNormalP3ui
typedef void (APIENTRYP PFNGLNORMALP3UIVPROC)(GLenum type, const GLuint *coords);
GLAPI PFNGLNORMALP3UIVPROC glad_glNormalP3uiv;
#define glNormalP3uiv glad_glNormalP3uiv
typedef void (APIENTRYP PFNGLCOLORP3UIPROC)(GLenum type, GLuint color);
GLAPI PFNGLCOLORP3UIPROC glad_glColorP3ui;
#define glColorP3ui glad_glColorP3ui
typedef void (APIENTRYP PFNGLCOLORP3UIVPROC)(GLenum type, const GLuint *color);
GLAPI PFNGLCOLORP3UIVPROC glad_glColorP3uiv;
#define glColorP3uiv glad_glColorP3uiv
typedef void (APIENTRYP PFNGLCOLORP4UIPROC)(GLenum type, GLuint color);
GLAPI PFNGLCOLORP4UIPROC glad_glColorP4ui;
#define glColorP4ui glad_glColorP4ui
typedef void (APIENTRYP PFNGLCOLORP4UIVPROC)(GLenum type, const GLuint *color);
GLAPI PFNGLCOLORP4UIVPROC glad_glColorP4uiv;
#define glColorP4uiv glad_glColorP4uiv
typedef void (APIENTRYP PFNGLSECONDARYCOLORP3UIPROC)(GLenum type, GLuint color);
GLAPI PFNGLSECONDARYCOLORP3UIPROC glad_glSecondaryColorP3ui;
#define glSecondaryColorP3ui glad_glSecondaryColorP3ui
typedef void (APIENTRYP PFNGLSECONDARYCOLORP3UIVPROC)(GLenum type, const GLuint *color);
GLAPI PFNGLSECONDARYCOLORP3UIVPROC glad_glSecondaryColorP3uiv;
#define glSecondaryColorP3uiv glad_glSecondaryColorP3uiv
#endif
#ifndef GL_VERSION_4_0
#define GL_VERSION_4_0 1
GLAPI int GLAD_GL_VERSION_4_0;
typedef void (APIENTRYP PFNGLMINSAMPLESHADINGPROC)(GLfloat value);
GLAPI PFNGLMINSAMPLESHADINGPROC glad_glMinSampleShading;
#define glMinSampleShading glad_glMinSampleShading
typedef void (APIENTRYP PFNGLBLENDEQUATIONIPROC)(GLuint buf, GLenum mode);
GLAPI PFNGLBLENDEQUATIONIPROC glad_glBlendEquationi;
#define glBlendEquationi glad_glBlendEquationi
```

```
typedef void (APIENTRYP PFNGLBLENDEQUATIONSEPARATEIPROC)(GLuint buf, GLenum modeRGB, GLenum modeAlpha);
GLAPI PFNGLBLENDEQUATIONSEPARATEIPROC glad_glBlendEquationSeparatei;
#define glBlendEquationSeparatei glad_glBlendEquationSeparatei
typedef void (APIENTRYP PFNGLBLENDFUNCIPROC)(GLuint buf, GLenum src, GLenum dst);
GLAPI PFNGLBLENDFUNCIPROC glad_glBlendFunci;
#define glBlendFunci glad_glBlendFunci
typedef void (APIENTRYP PFNGLBLENDFUNCSEPARATEIPROC)(GLuint buf, GLenum srcRGB, GLenum dstRGB, GLenum srcAlpha, GLenum dstAlpha);
GLAPI PFNGLBLENDFUNCSEPARATEIPROC glad_glBlendFuncSeparatei;
#define glBlendFuncSeparatei glad_glBlendFuncSeparatei
typedef void (APIENTRYP PFNGLDRAWARRAYSINDIRECTPROC)(GLenum mode, const void *indirect);
GLAPI PFNGLDRAWARRAYSINDIRECTPROC glad_glDrawArraysIndirect;
#define glDrawArraysIndirect glad_glDrawArraysIndirect
typedef void (APIENTRYP PFNGLDRAWELEMENTSINDIRECTPROC)(GLenum mode, GLenum type, const void *indirect);
GLAPI PFNGLDRAWELEMENTSINDIRECTPROC glad_glDrawElementsIndirect;
#define glDrawElementsIndirect glad_glDrawElementsIndirect
typedef void (APIENTRYP PFNGLUNIFORM1DPROC)(GLint location, GLdouble x);
GLAPI PFNGLUNIFORM1DPROC glad_glUniform1d;
#define glUniform1d glad_glUniform1d
typedef void (APIENTRYP PFNGLUNIFORM2DPROC)(GLint location, GLdouble x, GLdouble y);
GLAPI PFNGLUNIFORM2DPROC glad_glUniform2d;
#define glUniform2d glad_glUniform2d
typedef void (APIENTRYP PFNGLUNIFORM3DPROC)(GLint location, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLUNIFORM3DPROC glad_glUniform3d;
#define glUniform3d glad_glUniform3d
typedef void (APIENTRYP PFNGLUNIFORM4DPROC)(GLint location, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLUNIFORM4DPROC glad_glUniform4d;
#define glUniform4d glad_glUniform4d
typedef void (APIENTRYP PFNGLUNIFORM1DVPROC)(GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLUNIFORM1DVPROC glad_glUniform1dv;
#define glUniform1dv glad_glUniform1dv
typedef void (APIENTRYP PFNGLUNIFORM2DVPROC)(GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLUNIFORM2DVPROC glad_glUniform2dv;
#define glUniform2dv glad_glUniform2dv
typedef void (APIENTRYP PFNGLUNIFORM3DVPROC)(GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLUNIFORM3DVPROC glad_glUniform3dv;
#define glUniform3dv glad_glUniform3dv
typedef void (APIENTRYP PFNGLUNIFORM4DVPROC)(GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLUNIFORM4DVPROC glad_glUniform4dv;
#define glUniform4dv glad_glUniform4dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX2DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX2DVPROC glad_glUniformMatrix2dv;
#define glUniformMatrix2dv glad_glUniformMatrix2dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX3DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX3DVPROC glad_glUniformMatrix3dv;
#define glUniformMatrix3dv glad_glUniformMatrix3dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX4DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX4DVPROC glad_glUniformMatrix4dv;
#define glUniformMatrix4dv glad_glUniformMatrix4dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX2X3DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX2X3DVPROC glad_glUniformMatrix2x3dv;
#define glUniformMatrix2x3dv glad_glUniformMatrix2x3dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX2X4DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX2X4DVPROC glad_glUniformMatrix2x4dv;
#define glUniformMatrix2x4dv glad_glUniformMatrix2x4dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX3X2DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX3X2DVPROC glad_glUniformMatrix3x2dv;
#define glUniformMatrix3x2dv glad_glUniformMatrix3x2dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX3X4DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX3X4DVPROC glad_glUniformMatrix3x4dv;
#define glUniformMatrix3x4dv glad_glUniformMatrix3x4dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX4X2DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX4X2DVPROC glad_glUniformMatrix4x2dv;
#define glUniformMatrix4x2dv glad_glUniformMatrix4x2dv
typedef void (APIENTRYP PFNGLUNIFORMMATRIX4X3DVPROC)(GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLUNIFORMMATRIX4X3DVPROC glad_glUniformMatrix4x3dv;
#define glUniformMatrix4x3dv glad_glUniformMatrix4x3dv
typedef void (APIENTRYP PFNGLGETUNIFORMDVPROC)(GLuint program, GLint location, GLdouble *params);
GLAPI PFNGLGETUNIFORMDVPROC glad_glGetUniformdv;
#define glGetUniformdv glad_glGetUniformdv
typedef GLint (APIENTRYP PFNGLGETSUBROUTINEUNIFORMLOCATIONPROC)(GLuint program, GLenum shadertype, const GLchar *name);
GLAPI PFNGLGETSUBROUTINEUNIFORMLOCATIONPROC glad_glGetSubroutineUniformLocation;
#define glGetSubroutineUniformLocation glad_glGetSubroutineUniformLocation
typedef GLuint (APIENTRYP PFNGLGETSUBROUTINEINDEXPROC)(GLuint program, GLenum shadertype, const GLchar
*name);
GLAPI PFNGLGETSUBROUTINEINDEXPROC glad_glGetSubroutineIndex;
#define glGetSubroutineIndex glad_glGetSubroutineIndex
typedef void (APIENTRYP PFNGLGETACTIVESUBROUTINEUNIFORMIVPROC)(GLuint program, GLenum shadertype, GLuint index, GLenum pname, GLint *values);
GLAPI PFNGLGETACTIVESUBROUTINEUNIFORMIVPROC glad_glGetActiveSubroutineUniformiv;
#define glGetActiveSubroutineUniformiv glad_glGetActiveSubroutineUniformiv
typedef void (APIENTRYP PFNGLGETACTIVESUBROUTINEUNIFORMNAMEPROC)(GLuint program, GLenum shadertype, GLuint index, GLsizei bufSize, GLsizei *length, GLchar *name);
GLAPI PFNGLGETACTIVESUBROUTINEUNIFORMNAMEPROC glad_glGetActiveSubroutineUniformName;
#define glGetActiveSubroutineUniformName glad_glGetActiveSubroutineUniformName
typedef void (APIENTRYP PFNGLGETACTIVESUBROUTINENAMEPROC)(GLuint program, GLenum shadertype, GLuint index, GLsizei bufSize, GLsizei *length, GLchar *name);
GLAPI PFNGLGETACTIVESUBROUTINENAMEPROC glad_glGetActiveSubroutineName;
#define glGetActiveSubroutineName glad_glGetActiveSubroutineName
typedef void (APIENTRYP PFNGLUNIFORMSUBROUTINESUIVPROC)(GLenum shadertype, GLsizei count, const GLuint *indices);
GLAPI PFNGLUNIFORMSUBROUTINESUIVPROC glad_glUniformSubroutinesuiv;
#define glUniformSubroutinesuiv glad_glUniformSubroutinesuiv
typedef void (APIENTRYP PFNGLGETUNIFORMSUBROUTINEUIVPROC)(GLenum shadertype, GLint location, GLuint *params);
GLAPI PFNGLGETUNIFORMSUBROUTINEUIVPROC glad_glGetUniformSubroutineuiv;
#define glGetUniformSubroutineuiv glad_glGetUniformSubroutineuiv
typedef void (APIENTRYP PFNGLGETPROGRAMSTAGEIVPROC)(GLuint program, GLenum shadertype, GLenum pname, GLint *values);
GLAPI PFNGLGETPROGRAMSTAGEIVPROC glad_glGetProgramStageiv;
#define glGetProgramStageiv glad_glGetProgramStageiv
typedef void (APIENTRYP PFNGLPATCHPARAMETERIPROC)(GLenum pname, GLint value);
GLAPI PFNGLPATCHPARAMETERIPROC glad_glPatchParameteri;
#define glPatchParameteri glad_glPatchParameteri
typedef void (APIENTRYP PFNGLPATCHPARAMETERFVPROC)(GLenum pname, const GLfloat *values);
GLAPI PFNGLPATCHPARAMETERFVPROC glad_glPatchParameterfv;
#define glPatchParameterfv glad_glPatchParameterfv
typedef void (APIENTRYP PFNGLBINDTRANSFORMFEEDBACKPROC)(GLenum target, GLuint id);
GLAPI PFNGLBINDTRANSFORMFEEDBACKPROC glad_glBindTransformFeedback;
#define glBindTransformFeedback glad_glBindTransformFeedback
typedef void (APIENTRYP PFNGLDELETETRANSFORMFEEDBACKSPROC)(GLsizei n, const GLuint *ids);
GLAPI PFNGLDELETETRANSFORMFEEDBACKSPROC glad_glDeleteTransformFeedbacks;
#define glDeleteTransformFeedbacks glad_glDeleteTransformFeedbacks
typedef void (APIENTRYP PFNGLGENTRANSFORMFEEDBACKSPROC)(GLsizei n, GLuint *ids);
GLAPI PFNGLGENTRANSFORMFEEDBACKSPROC glad_glGenTransformFeedbacks;
#define glGenTransformFeedbacks glad_glGenTransformFeedbacks
typedef GLboolean (APIENTRYP PFNGLISTRANSFORMFEEDBACKPROC)(GLuint id);
GLAPI PFNGLISTRANSFORMFEEDBACKPROC glad_glIsTransformFeedback;
#define glIsTransformFeedback glad_glIsTransformFeedback
typedef void (APIENTRYP PFNGLPAUSETRANSFORMFEEDBACKPROC)(void);
GLAPI PFNGLPAUSETRANSFORMFEEDBACKPROC glad_glPauseTransformFeedback;
#define glPauseTransformFeedback glad_glPauseTransformFeedback
typedef void (APIENTRYP PFNGLRESUMETRANSFORMFEEDBACKPROC)(void);
GLAPI PFNGLRESUMETRANSFORMFEEDBACKPROC glad_glResumeTransformFeedback;
#define glResumeTransformFeedback glad_glResumeTransformFeedback
typedef void (APIENTRYP PFNGLDRAWTRANSFORMFEEDBACKPROC)(GLenum mode, GLuint id);
GLAPI PFNGLDRAWTRANSFORMFEEDBACKPROC glad_glDrawTransformFeedback;
#define glDrawTransformFeedback glad_glDrawTransformFeedback
typedef void (APIENTRYP PFNGLDRAWTRANSFORMFEEDBACKSTREAMPROC)(GLenum mode, GLuint id, GLuint stream);
GLAPI PFNGLDRAWTRANSFORMFEEDBACKSTREAMPROC glad_glDrawTransformFeedbackStream;
#define glDrawTransformFeedbackStream glad_glDrawTransformFeedbackStream
typedef void (APIENTRYP PFNGLBEGINQUERYINDEXEDPROC)(GLenum target, GLuint index, GLuint id);
GLAPI PFNGLBEGINQUERYINDEXEDPROC glad_glBeginQueryIndexed;
#define glBeginQueryIndexed glad_glBeginQueryIndexed
typedef void (APIENTRYP PFNGLENDQUERYINDEXEDPROC)(GLenum target, GLuint index);
GLAPI PFNGLENDQUERYINDEXEDPROC glad_glEndQueryIndexed;
#define glEndQueryIndexed glad_glEndQueryIndexed
typedef void (APIENTRYP PFNGLGETQUERYINDEXEDIVPROC)(GLenum target, GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETQUERYINDEXEDIVPROC glad_glGetQueryIndexediv;
#define glGetQueryIndexediv glad_glGetQueryIndexediv
#endif
#ifndef GL_VERSION_4_1
#define GL_VERSION_4_1 1
GLAPI int GLAD_GL_VERSION_4_1;
typedef void (APIENTRYP PFNGLRELEASESHADERCOMPILERPROC)(void);
GLAPI PFNGLRELEASESHADERCOMPILERPROC glad_glReleaseShaderCompiler;
#define glReleaseShaderCompiler glad_glReleaseShaderCompiler
typedef void (APIENTRYP PFNGLSHADERBINARYPROC)(GLsizei count, const GLuint *shaders, GLenum binaryFormat, const void *binary, GLsizei length);
GLAPI PFNGLSHADERBINARYPROC glad_glShaderBinary;
#define glShaderBinary glad_glShaderBinary
typedef void (APIENTRYP PFNGLGETSHADERPRECISIONFORMATPROC)(GLenum shadertype, GLenum precisiontype, GLint *range, GLint *precision);
GLAPI PFNGLGETSHADERPRECISIONFORMATPROC glad_glGetShaderPrecisionFormat;
#define glGetShaderPrecisionFormat glad_glGetShaderPrecisionFormat
typedef void (APIENTRYP PFNGLDEPTHRANGEFPROC)(GLfloat n, GLfloat f);
GLAPI PFNGLDEPTHRANGEFPROC glad_glDepthRangef;
#define glDepthRangef glad_glDepthRangef
typedef void (APIENTRYP PFNGLCLEARDEPTHFPROC)(GLfloat d);
GLAPI PFNGLCLEARDEPTHFPROC glad_glClearDepthf;
```

```
#define glClearDepthf glad_glClearDepthf
typedef void (APIENTRYP PFNGLGETPROGRAMBINARYPROC)(GLuint program, GLsizei bufSize, GLsizei *length, GLenum *binaryFormat, void *binary);
GLAPI PFNGLGETPROGRAMBINARYPROC glad_glGetProgramBinary;
#define glGetProgramBinary glad_glGetProgramBinary
typedef void (APIENTRYP PFNGLPROGRAMBINARYPROC)(GLuint program, GLenum binaryFormat, const void *binary, GLsizei length);
GLAPI PFNGLPROGRAMBINARYPROC glad_glProgramBinary;
#define glProgramBinary glad_glProgramBinary
typedef void (APIENTRYP PFNGLPROGRAMPARAMETERIPROC)(GLuint program, GLenum pname, GLint value);
GLAPI PFNGLPROGRAMPARAMETERIPROC glad_glProgramParameteri;
#define glProgramParameteri glad_glProgramParameteri
typedef void (APIENTRYP PFNGLUSEPROGRAMSTAGESPROC)(GLuint pipeline, GLbitfield stages, GLuint program);
GLAPI PFNGLUSEPROGRAMSTAGESPROC glad_glUseProgramStages;
#define glUseProgramStages glad_glUseProgramStages
typedef void (APIENTRYP PFNGLACTIVESHADERPROGRAMPROC)(GLuint pipeline, GLuint program);
GLAPI PFNGLACTIVESHADERPROGRAMPROC glad_glActiveShaderProgram;
#define glActiveShaderProgram glad_glActiveShaderProgram
typedef GLuint (APIENTRYP PFNGLCREATESHADERPROGRAMVPROC)(GLenum type, GLsizei count, const GLchar *const*strings);
GLAPI PFNGLCREATESHADERPROGRAMVPROC glad_glCreateShaderProgramv;
#define glCreateShaderProgramv glad_glCreateShaderProgramv
typedef void (APIENTRYP PFNGLBINDPROGRAMPIPELINEPROC)(GLuint pipeline);
GLAPI PFNGLBINDPROGRAMPIPELINEPROC glad_glBindProgramPipeline;
#define glBindProgramPipeline glad_glBindProgramPipeline
typedef void (APIENTRYP PFNGLDELETEPROGRAMPIPELINESPROC)(GLsizei n, const GLuint *pipelines);
GLAPI PFNGLDELETEPROGRAMPIPELINESPROC glad_glDeleteProgramPipelines;
#define glDeleteProgramPipelines glad_glDeleteProgramPipelines
typedef void (APIENTRYP PFNGLGENPROGRAMPIPELINESPROC)(GLsizei n, GLuint *pipelines);
GLAPI PFNGLGENPROGRAMPIPELINESPROC glad_glGenProgramPipelines;
#define glGenProgramPipelines glad_glGenProgramPipelines
typedef GLboolean (APIENTRYP PFNGLISPROGRAMPIPELINEPROC)(GLuint pipeline);
GLAPI PFNGLISPROGRAMPIPELINEPROC glad_glIsProgramPipeline;
#define glIsProgramPipeline glad_glIsProgramPipeline
typedef void (APIENTRYP PFNGLGETPROGRAMPIPELINEIVPROC)(GLuint pipeline, GLenum pname, GLint *params);
GLAPI PFNGLGETPROGRAMPIPELINEIVPROC glad_glGetProgramPipelineiv;
#define glGetProgramPipelineiv glad_glGetProgramPipelineiv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1IPROC)(GLuint program, GLint location, GLint v0);
GLAPI PFNGLPROGRAMUNIFORM1IPROC glad_glProgramUniform1i;
#define glProgramUniform1i glad_glProgramUniform1i
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1IVPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM1IVPROC glad_glProgramUniform1iv;
#define glProgramUniform1iv glad_glProgramUniform1iv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1FPROC)(GLuint program, GLint location, GLfloat v0);
GLAPI PFNGLPROGRAMUNIFORM1FPROC glad_glProgramUniform1f;
#define glProgramUniform1f glad_glProgramUniform1f
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1FVPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM1FVPROC glad_glProgramUniform1fv;
#define glProgramUniform1fv glad_glProgramUniform1fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1DPROC)(GLuint program, GLint location, GLdouble v0);
GLAPI PFNGLPROGRAMUNIFORM1DPROC glad_glProgramUniform1d;
#define glProgramUniform1d glad_glProgramUniform1d
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1DVPROC)(GLuint program, GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM1DVPROC glad_glProgramUniform1dv;
#define glProgramUniform1dv glad_glProgramUniform1dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UIPROC)(GLuint program, GLint location, GLuint v0);
GLAPI PFNGLPROGRAMUNIFORM1UIPROC glad_glProgramUniform1ui;
#define glProgramUniform1ui glad_glProgramUniform1ui
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UIVPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM1UIVPROC glad_glProgramUniform1uiv;
#define glProgramUniform1uiv glad_glProgramUniform1uiv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2IPROC)(GLuint program, GLint location, GLint v0, GLint v1);
GLAPI PFNGLPROGRAMUNIFORM2IPROC glad_glProgramUniform2i;
#define glProgramUniform2i glad_glProgramUniform2i
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2IVPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM2IVPROC glad_glProgramUniform2iv;
#define glProgramUniform2iv glad_glProgramUniform2iv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2FPROC)(GLuint program, GLint location, GLfloat v0, GLfloat v1);
GLAPI PFNGLPROGRAMUNIFORM2FPROC glad_glProgramUniform2f;
#define glProgramUniform2f glad_glProgramUniform2f
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2FVPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM2FVPROC glad_glProgramUniform2fv;
#define glProgramUniform2fv glad_glProgramUniform2fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2DPROC)(GLuint program, GLint location, GLdouble v0, GLdouble v1);
GLAPI PFNGLPROGRAMUNIFORM2DPROC glad_glProgramUniform2d;
#define glProgramUniform2d glad_glProgramUniform2d
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2DVPROC)(GLuint program, GLint location, GLsizei count,
const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM2DVPROC glad_glProgramUniform2dv;
#define glProgramUniform2dv glad_glProgramUniform2dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UIPROC)(GLuint program, GLint location, GLuint v0, GLuint v1);
GLAPI PFNGLPROGRAMUNIFORM2UIPROC glad_glProgramUniform2ui;
#define glProgramUniform2ui glad_glProgramUniform2ui
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UIVPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM2UIVPROC glad_glProgramUniform2uiv;
#define glProgramUniform2uiv glad_glProgramUniform2uiv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3IPROC)(GLuint program, GLint location, GLint v0, GLint v1, GLint v2);
GLAPI PFNGLPROGRAMUNIFORM3IPROC glad_glProgramUniform3i;
#define glProgramUniform3i glad_glProgramUniform3i
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3IVPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM3IVPROC glad_glProgramUniform3iv;
#define glProgramUniform3iv glad_glProgramUniform3iv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3FPROC)(GLuint program, GLint location, GLfloat v0, GLfloat v1, GLfloat v2);
GLAPI PFNGLPROGRAMUNIFORM3FPROC glad_glProgramUniform3f;
#define glProgramUniform3f glad_glProgramUniform3f
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3FVPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM3FVPROC glad_glProgramUniform3fv;
#define glProgramUniform3fv glad_glProgramUniform3fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3DPROC)(GLuint program, GLint location, GLdouble v0, GLdouble v1, GLdouble v2);
GLAPI PFNGLPROGRAMUNIFORM3DPROC glad_glProgramUniform3d;
#define glProgramUniform3d glad_glProgramUniform3d
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3DVPROC)(GLuint program, GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM3DVPROC glad_glProgramUniform3dv;
#define glProgramUniform3dv glad_glProgramUniform3dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UIPROC)(GLuint program, GLint location, GLuint v0, GLuint v1, GLuint v2);
GLAPI PFNGLPROGRAMUNIFORM3UIPROC glad_glProgramUniform3ui;
#define glProgramUniform3ui glad_glProgramUniform3ui
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UIVPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM3UIVPROC glad_glProgramUniform3uiv;
#define glProgramUniform3uiv glad_glProgramUniform3uiv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4IPROC)(GLuint program, GLint location, GLint v0, GLint v1, GLint v2, GLint v3);
GLAPI PFNGLPROGRAMUNIFORM4IPROC glad_glProgramUniform4i;
#define glProgramUniform4i glad_glProgramUniform4i
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4IVPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM4IVPROC glad_glProgramUniform4iv;
#define glProgramUniform4iv glad_glProgramUniform4iv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4FPROC)(GLuint program, GLint location, GLfloat v0, GLfloat v1, GLfloat v2, GLfloat v3);
GLAPI PFNGLPROGRAMUNIFORM4FPROC glad_glProgramUniform4f;
#define glProgramUniform4f glad_glProgramUniform4f
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4FVPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM4FVPROC glad_glProgramUniform4fv;
#define glProgramUniform4fv glad_glProgramUniform4fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4DPROC)(GLuint program, GLint location, GLdouble v0, GLdouble v1, GLdouble v2, GLdouble v3);
GLAPI PFNGLPROGRAMUNIFORM4DPROC glad_glProgramUniform4d;
#define glProgramUniform4d glad_glProgramUniform4d
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4DVPROC)(GLuint program, GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM4DVPROC glad_glProgramUniform4dv;
#define glProgramUniform4dv glad_glProgramUniform4dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UIPROC)(GLuint program, GLint location, GLuint v0, GLuint v1, GLuint v2, GLuint v3);
GLAPI PFNGLPROGRAMUNIFORM4UIPROC glad_glProgramUniform4ui;
#define glProgramUniform4ui glad_glProgramUniform4ui
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UIVPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM4UIVPROC glad_glProgramUniform4uiv;
#define glProgramUniform4uiv glad_glProgramUniform4uiv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2FVPROC glad_glProgramUniformMatrix2fv;
#define glProgramUniformMatrix2fv glad_glProgramUniformMatrix2fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3FVPROC glad_glProgramUniformMatrix3fv;
#define glProgramUniformMatrix3fv glad_glProgramUniformMatrix3fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4FVPROC glad_glProgramUniformMatrix4fv;
#define glProgramUniformMatrix4fv glad_glProgramUniformMatrix4fv
```

```
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2DVPROC glad_glProgramUniformMatrix2dv;
#define glProgramUniformMatrix2dv glad_glProgramUniformMatrix2dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3DVPROC glad_glProgramUniformMatrix3dv;
#define glProgramUniformMatrix3dv glad_glProgramUniformMatrix3dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4DVPROC glad_glProgramUniformMatrix4dv;
#define glProgramUniformMatrix4dv glad_glProgramUniformMatrix4dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X3FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3FVPROC glad_glProgramUniformMatrix2x3fv;
#define glProgramUniformMatrix2x3fv glad_glProgramUniformMatrix2x3fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X2FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2FVPROC glad_glProgramUniformMatrix3x2fv;
#define glProgramUniformMatrix3x2fv glad_glProgramUniformMatrix3x2fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X4FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4FVPROC glad_glProgramUniformMatrix2x4fv;
#define glProgramUniformMatrix2x4fv glad_glProgramUniformMatrix2x4fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X2FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2FVPROC glad_glProgramUniformMatrix4x2fv;
#define glProgramUniformMatrix4x2fv glad_glProgramUniformMatrix4x2fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X4FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4FVPROC glad_glProgramUniformMatrix3x4fv;
#define glProgramUniformMatrix3x4fv glad_glProgramUniformMatrix3x4fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X3FVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3FVPROC glad_glProgramUniformMatrix4x3fv;
#define glProgramUniformMatrix4x3fv glad_glProgramUniformMatrix4x3fv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X3DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3DVPROC glad_glProgramUniformMatrix2x3dv;
#define glProgramUniformMatrix2x3dv glad_glProgramUniformMatrix2x3dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X2DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2DVPROC glad_glProgramUniformMatrix3x2dv;
#define glProgramUniformMatrix3x2dv glad_glProgramUniformMatrix3x2dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X4DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4DVPROC glad_glProgramUniformMatrix2x4dv;
#define glProgramUniformMatrix2x4dv glad_glProgramUniformMatrix2x4dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X2DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2DVPROC glad_glProgramUniformMatrix4x2dv;
#define glProgramUniformMatrix4x2dv glad_glProgramUniformMatrix4x2dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X4DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4DVPROC glad_glProgramUniformMatrix3x4dv;
#define glProgramUniformMatrix3x4dv glad_glProgramUniformMatrix3x4dv
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X3DVPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3DVPROC glad_glProgramUniformMatrix4x3dv;
#define glProgramUniformMatrix4x3dv glad_glProgramUniformMatrix4x3dv
typedef void (APIENTRYP PFNGLVALIDATEPROGRAMPIPELINEPROC)(GLuint pipeline);
GLAPI PFNGLVALIDATEPROGRAMPIPELINEPROC glad_glValidateProgramPipeline;
#define glValidateProgramPipeline glad_glValidateProgramPipeline
typedef void (APIENTRYP PFNGLGETPROGRAMPIPELINEINFOLOGPROC)(GLuint pipeline, GLsizei bufSize, GLsizei *length, GLchar *infoLog);
GLAPI PFNGLGETPROGRAMPIPELINEINFOLOGPROC glad_glGetProgramPipelineInfoLog;
#define glGetProgramPipelineInfoLog glad_glGetProgramPipelineInfoLog
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1DPROC)(GLuint index, GLdouble x);
GLAPI PFNGLVERTEXATTRIBL1DPROC glad_glVertexAttribL1d;
#define glVertexAttribL1d glad_glVertexAttribL1d
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2DPROC)(GLuint index, GLdouble x, GLdouble y);
GLAPI PFNGLVERTEXATTRIBL2DPROC glad_glVertexAttribL2d;
#define glVertexAttribL2d glad_glVertexAttribL2d
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3DPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLVERTEXATTRIBL3DPROC glad_glVertexAttribL3d;
#define glVertexAttribL3d glad_glVertexAttribL3d
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4DPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLVERTEXATTRIBL4DPROC glad_glVertexAttribL4d;
#define glVertexAttribL4d glad_glVertexAttribL4d
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1DVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBL1DVPROC glad_glVertexAttribL1dv;
#define glVertexAttribL1dv glad_glVertexAttribL1dv
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2DVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBL2DVPROC glad_glVertexAttribL2dv;
#define glVertexAttribL2dv glad_glVertexAttribL2dv
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3DVPROC)(GLuint index, const GLdouble *v);
```

```
GLAPI PFNGLVERTEXATTRIBL3DVPROC glad_glVertexAttribL3dv;
#define glVertexAttribL3dv glad_glVertexAttribL3dv
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4DVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBL4DVPROC glad_glVertexAttribL4dv;
#define glVertexAttribL4dv glad_glVertexAttribL4dv
typedef void (APIENTRYP PFNGLVERTEXATTRIBLPOINTERPROC)(GLuint index, GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLVERTEXATTRIBLPOINTERPROC glad_glVertexAttribLPointer;
#define glVertexAttribLPointer glad_glVertexAttribLPointer
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBLDVPROC)(GLuint index, GLenum pname, GLdouble *params);
GLAPI PFNGLGETVERTEXATTRIBLDVPROC glad_glGetVertexAttribLdv;
#define glGetVertexAttribLdv glad_glGetVertexAttribLdv
typedef void (APIENTRYP PFNGLVIEWPORTARRAYVPROC)(GLuint first, GLsizei count, const GLfloat *v);
GLAPI PFNGLVIEWPORTARRAYVPROC glad_glViewportArrayv;
#define glViewportArrayv glad_glViewportArrayv
typedef void (APIENTRYP PFNGLVIEWPORTINDEXEDFPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat w, GLfloat h);
GLAPI PFNGLVIEWPORTINDEXEDFPROC glad_glViewportIndexedf;
#define glViewportIndexedf glad_glViewportIndexedf
typedef void (APIENTRYP PFNGLVIEWPORTINDEXEDFVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVIEWPORTINDEXEDFVPROC glad_glViewportIndexedfv;
#define glViewportIndexedfv glad_glViewportIndexedfv
typedef void (APIENTRYP PFNGLSCISSORARRAYVPROC)(GLuint first, GLsizei count, const GLint *v);
GLAPI PFNGLSCISSORARRAYVPROC glad_glScissorArrayv;
#define glScissorArrayv glad_glScissorArrayv
typedef void (APIENTRYP PFNGLSCISSORINDEXEDPROC)(GLuint index, GLint left, GLint bottom, GLsizei width, GLsizei height);
GLAPI PFNGLSCISSORINDEXEDPROC glad_glScissorIndexed;
#define glScissorIndexed glad_glScissorIndexed
typedef void (APIENTRYP PFNGLSCISSORINDEXEDVPROC)(GLuint index, const GLint *v);
GLAPI PFNGLSCISSORINDEXEDVPROC glad_glScissorIndexedv;
#define glScissorIndexedv glad_glScissorIndexedv
typedef void (APIENTRYP PFNGLDEPTHRANGEARRAYVPROC)(GLuint first, GLsizei count, const GLdouble *v);
GLAPI PFNGLDEPTHRANGEARRAYVPROC glad_glDepthRangeArrayv;
#define glDepthRangeArrayv glad_glDepthRangeArrayv
typedef void (APIENTRYP PFNGLDEPTHRANGEINDEXEDPROC)(GLuint index, GLdouble n, GLdouble f);
GLAPI PFNGLDEPTHRANGEINDEXEDPROC glad_glDepthRangeIndexed;
#define glDepthRangeIndexed glad_glDepthRangeIndexed
typedef void (APIENTRYP PFNGLGETFLOATI_VPROC)(GLenum target, GLuint index, GLfloat *data);
GLAPI PFNGLGETFLOATI_VPROC glad_glGetFloati_v;
#define glGetFloati_v glad_glGetFloati_v
typedef void (APIENTRYP PFNGLGETDOUBLEI_VPROC)(GLenum target, GLuint index, GLdouble *data);
GLAPI PFNGLGETDOUBLEI_VPROC glad_glGetDoublei_v;
#define glGetDoublei_v glad_glGetDoublei_v
#endif
#define GL_MULTISAMPLE_3DFX 0x86B2
#define GL_SAMPLE_BUFFERS_3DFX 0x86B3
#define GL_SAMPLES_3DFX 0x86B4
#define GL_MULTISAMPLE_BIT_3DFX 0x20000000
#define GL_COMPRESSED_RGB_FXT1_3DFX 0x86B0
#define GL_COMPRESSED_RGBA_FXT1_3DFX 0x86B1
#define GL_FACTOR_MIN_AMD 0x901C
#define GL_FACTOR_MAX_AMD 0x901D
#define GL_MAX_DEBUG_MESSAGE_LENGTH_AMD 0x9143
#define GL_MAX_DEBUG_LOGGED_MESSAGES_AMD 0x9144
#define GL_DEBUG_LOGGED_MESSAGES_AMD 0x9145
#define GL_DEBUG_SEVERITY_HIGH_AMD 0x9146
#define GL_DEBUG_SEVERITY_MEDIUM_AMD 0x9147
#define GL_DEBUG_SEVERITY_LOW_AMD 0x9148
#define GL_DEBUG_CATEGORY_API_ERROR_AMD 0x9149
#define GL_DEBUG_CATEGORY_WINDOW_SYSTEM_AMD 0x914A
#define GL_DEBUG_CATEGORY_DEPRECATION_AMD 0x914B
#define GL_DEBUG_CATEGORY_UNDEFINED_BEHAVIOR_AMD 0x914C
#define GL_DEBUG_CATEGORY_PERFORMANCE_AMD 0x914D
#define GL_DEBUG_CATEGORY_SHADER_COMPILER_AMD 0x914E
#define GL_DEBUG_CATEGORY_APPLICATION_AMD 0x914F
#define GL_DEBUG_CATEGORY_OTHER_AMD 0x9150
#define GL_DEPTH_CLAMP_NEAR_AMD 0x901E
#define GL_DEPTH_CLAMP_FAR_AMD 0x901F
#define GL_RENDERBUFFER_STORAGE_SAMPLES_AMD 0x91B2
#define GL_MAX_COLOR_FRAMEBUFFER_SAMPLES_AMD 0x91B3
#define GL_MAX_COLOR_FRAMEBUFFER_STORAGE_SAMPLES_AMD 0x91B4
#define GL_MAX_DEPTH_STENCIL_FRAMEBUFFER_SAMPLES_AMD 0x91B5
#define GL_NUM_SUPPORTED_MULTISAMPLE_MODES_AMD 0x91B6
#define GL_SUPPORTED_MULTISAMPLE_MODES_AMD 0x91B7
#define GL_SUBSAMPLE_DISTANCE_AMD 0x883F
#define GL_PIXELS_PER_SAMPLE_PATTERN_X_AMD 0x91AE
#define GL_PIXELS_PER_SAMPLE_PATTERN_Y_AMD 0x91AF
#define GL_ALL_PIXELS_AMD 0xFFFFFFFF
#define GL_FLOAT16_NV 0x8FF8
#define GL_FLOAT16_VEC2_NV 0x8FF9
#define GL_FLOAT16_VEC3_NV 0x8FFA
#define GL_FLOAT16_VEC4_NV 0x8FFB
#define GL_FLOAT16_MAT2_AMD 0x91C5
#define GL_FLOAT16_MAT3_AMD 0x91C6
#define GL_FLOAT16_MAT4_AMD 0x91C7
#define GL_FLOAT16_MAT2x3_AMD 0x91C8
```

```
#define GL_FLOAT16_MAT2x4_AMD 0x91C9
#define GL_FLOAT16_MAT3x2_AMD 0x91CA
#define GL_FLOAT16_MAT3x4_AMD 0x91CB
#define GL_FLOAT16_MAT4x2_AMD 0x91CC
#define GL_FLOAT16_MAT4x3_AMD 0x91CD
#define GL_INT64_NV 0x140E
#define GL_UNSIGNED_INT64_NV 0x140F
#define GL_INT8_NV 0x8FE0
#define GL_INT8_VEC2_NV 0x8FE1
#define GL_INT8_VEC3_NV 0x8FE2
#define GL_INT8_VEC4_NV 0x8FE3
#define GL_INT16_NV 0x8FE4
#define GL_INT16_VEC2_NV 0x8FE5
#define GL_INT16_VEC3_NV 0x8FE6
#define GL_INT16_VEC4_NV 0x8FE7
#define GL_INT64_VEC2_NV 0x8FE9
#define GL_INT64_VEC3_NV 0x8FEA
#define GL_INT64_VEC4_NV 0x8FEB
#define GL_UNSIGNED_INT8_NV 0x8FEC
#define GL_UNSIGNED_INT8_VEC2_NV 0x8FED
#define GL_UNSIGNED_INT8_VEC3_NV 0x8FEE
#define GL_UNSIGNED_INT8_VEC4_NV 0x8FEF
#define GL_UNSIGNED_INT16_NV 0x8FF0
#define GL_UNSIGNED_INT16_VEC2_NV 0x8FF1
#define GL_UNSIGNED_INT16_VEC3_NV 0x8FF2
#define GL_UNSIGNED_INT16_VEC4_NV 0x8FF3
#define GL_UNSIGNED_INT64_VEC2_NV 0x8FF5
#define GL_UNSIGNED_INT64_VEC3_NV 0x8FF6
#define GL_UNSIGNED_INT64_VEC4_NV 0x8FF7
#define GL_VERTEX_ELEMENT_SWIZZLE_AMD 0x91A4
#define GL_VERTEX_ID_SWIZZLE_AMD 0x91A5
#define GL_DATA_BUFFER_AMD 0x9151
#define GL_PERFORMANCE_MONITOR_AMD 0x9152
#define GL_QUERY_OBJECT_AMD 0x9153
#define GL_VERTEX_ARRAY_OBJECT_AMD 0x9154
#define GL_SAMPLER_OBJECT_AMD 0x9155
#define GL_OCCLUSION_QUERY_EVENT_MASK_AMD 0x874F
#define GL_QUERY_DEPTH_PASS_EVENT_BIT_AMD 0x00000001
#define GL_QUERY_DEPTH_FAIL_EVENT_BIT_AMD 0x00000002
#define GL_QUERY_STENCIL_FAIL_EVENT_BIT_AMD 0x00000004
#define GL_QUERY_DEPTH_BOUNDS_FAIL_EVENT_BIT_AMD 0x00000008
#define GL_QUERY_ALL_EVENT_BITS_AMD 0xFFFFFFFF
#define GL_COUNTER_TYPE_AMD 0x8BC0
#define GL_COUNTER_RANGE_AMD 0x8BC1
#define GL_UNSIGNED_INT64_AMD 0x8BC2
#define GL_PERCENTAGE_AMD 0x8BC3
#define GL_PERFMON_RESULT_AVAILABLE_AMD 0x8BC4
#define GL_PERFMON_RESULT_SIZE_AMD 0x8BC5
#define GL_PERFMON_RESULT_AMD 0x8BC6
#define GL_EXTERNAL_VIRTUAL_MEMORY_BUFFER_AMD 0x9160
#define GL_QUERY_BUFFER_AMD 0x9192
#define GL_QUERY_BUFFER_BINDING_AMD 0x9193
#define GL_QUERY_RESULT_NO_WAIT_AMD 0x9194
#define GL_VIRTUAL_PAGE_SIZE_X_AMD 0x9195
#define GL_VIRTUAL_PAGE_SIZE_Y_AMD 0x9196
#define GL_VIRTUAL_PAGE_SIZE_Z_AMD 0x9197
#define GL_MAX_SPARSE_TEXTURE_SIZE_AMD 0x9198
#define GL_MAX_SPARSE_3D_TEXTURE_SIZE_AMD 0x9199
#define GL_MAX_SPARSE_ARRAY_TEXTURE_LAYERS 0x919A
#define GL_MIN_SPARSE_LEVEL_AMD 0x919B
#define GL_MIN_LOD_WARNING_AMD 0x919C
#define GL_TEXTURE_STORAGE_SPARSE_BIT_AMD 0x00000001
#define GL_SET_AMD 0x874A
#define GL_REPLACE_VALUE_AMD 0x874B
#define GL_STENCIL_OP_VALUE_AMD 0x874C
#define GL_STENCIL_BACK_OP_VALUE_AMD 0x874D
#define GL_STREAM_RASTERIZATION_AMD 0x91A0
#define GL_SAMPLER_BUFFER_AMD 0x9001
#define GL_INT_SAMPLER_BUFFER_AMD 0x9002
#define GL_UNSIGNED_INT_SAMPLER_BUFFER_AMD 0x9003
#define GL_TESSELLATION_MODE_AMD 0x9004
#define GL_TESSELLATION_FACTOR_AMD 0x9005
#define GL_DISCRETE_AMD 0x9006
#define GL_CONTINUOUS_AMD 0x9007
#define GL_AUX_DEPTH_STENCIL_APPLE 0x8A14
#define GL_UNPACK_CLIENT_STORAGE_APPLE 0x85B2
#define GL_ELEMENT_ARRAY_APPLE 0x8A0C
#define GL_ELEMENT_ARRAY_TYPE_APPLE 0x8A0D
#define GL_ELEMENT_ARRAY_POINTER_APPLE 0x8A0E
#define GL_DRAW_PIXELS_APPLE 0x8A0A
#define GL_FENCE_APPLE 0x8A0B
#define GL_HALF_APPLE 0x140B
#define GL_RGBA_FLOAT32_APPLE 0x8814
#define GL_RGB_FLOAT32_APPLE 0x8815
#define GL_ALPHA_FLOAT32_APPLE 0x8816
#define GL_INTENSITY_FLOAT32_APPLE 0x8817
#define GL_LUMINANCE_FLOAT32_APPLE 0x8818
```

```
#define GL_LUMINANCE_ALPHA_FLOAT32_APPLE 0x8819
#define GL_RGBA_FLOAT16_APPLE 0x881A
#define GL_RGB_FLOAT16_APPLE 0x881B
#define GL_ALPHA_FLOAT16_APPLE 0x881C
#define GL_INTENSITY_FLOAT16_APPLE 0x881D
#define GL_LUMINANCE_FLOAT16_APPLE 0x881E
#define GL_LUMINANCE_ALPHA_FLOAT16_APPLE 0x881F
#define GL_COLOR_FLOAT_APPLE 0x8A0F
#define GL_BUFFER_SERIALIZED_MODIFY_APPLE 0x8A12
#define GL_BUFFER_FLUSHING_UNMAP_APPLE 0x8A13
#define GL_BUFFER_OBJECT_APPLE 0x85B3
#define GL_RELEASED_APPLE 0x8A19
#define GL_VOLATILE_APPLE 0x8A1A
#define GL_RETAINED_APPLE 0x8A1B
#define GL_UNDEFINED_APPLE 0x8A1C
#define GL_PURGEABLE_APPLE 0x8A1D
#define GL_RGB_422_APPLE 0x8A1F
#define GL_UNSIGNED_SHORT_8_8_APPLE 0x85BA
#define GL_UNSIGNED_SHORT_8_8_REV_APPLE 0x85BB
#define GL_RGB_RAW_422_APPLE 0x8A51
#define GL_PACK_ROW_BYTES_APPLE 0x8A15
#define GL_UNPACK_ROW_BYTES_APPLE 0x8A16
#define GL_LIGHT_MODEL_SPECULAR_VECTOR_APPLE 0x85B0
#define GL_TEXTURE_RANGE_LENGTH_APPLE 0x85B7
#define GL_TEXTURE_RANGE_POINTER_APPLE 0x85B8
#define GL_TEXTURE_STORAGE_HINT_APPLE 0x85BC
#define GL_STORAGE_PRIVATE_APPLE 0x85BD
#define GL_STORAGE_CACHED_APPLE 0x85BE
#define GL_STORAGE_SHARED_APPLE 0x85BF
#define GL_TRANSFORM_HINT_APPLE 0x85B1
#define GL_VERTEX_ARRAY_BINDING_APPLE 0x85B5
#define GL_VERTEX_ARRAY_RANGE_APPLE 0x851D
#define GL_VERTEX_ARRAY_RANGE_LENGTH_APPLE 0x851E
#define GL_VERTEX_ARRAY_STORAGE_HINT_APPLE 0x851F
#define GL_VERTEX_ARRAY_RANGE_POINTER_APPLE 0x8521
#define GL_STORAGE_CLIENT_APPLE 0x85B4
#define GL_VERTEX_ATTRIB_MAP1_APPLE 0x8A00
#define GL_VERTEX_ATTRIB_MAP2_APPLE 0x8A01
#define GL_VERTEX_ATTRIB_MAP1_SIZE_APPLE 0x8A02
#define GL_VERTEX_ATTRIB_MAP1_COEFF_APPLE 0x8A03
#define GL_VERTEX_ATTRIB_MAP1_ORDER_APPLE 0x8A04
#define GL_VERTEX_ATTRIB_MAP1_DOMAIN_APPLE 0x8A05
#define GL_VERTEX_ATTRIB_MAP2_SIZE_APPLE 0x8A06
#define GL_VERTEX_ATTRIB_MAP2_COEFF_APPLE 0x8A07
#define GL_VERTEX_ATTRIB_MAP2_ORDER_APPLE 0x8A08
#define GL_VERTEX_ATTRIB_MAP2_DOMAIN_APPLE 0x8A09
#define GL_YCBCR_422_APPLE 0x85B9
#define GL_PRIMITIVE_BOUNDING_BOX_ARB 0x92BE
#define GL_MULTISAMPLE_LINE_WIDTH_RANGE_ARB 0x9381
#define GL_MULTISAMPLE_LINE_WIDTH_GRANULARITY_ARB 0x9382
#define GL_COMPRESSED_RGB8_ETC2 0x9274
#define GL_COMPRESSED_SRGB8_ETC2 0x9275
#define GL_COMPRESSED_RGB8_PUNCHTHROUGH_ALPHA1_ETC2 0x9276
#define GL_COMPRESSED_SRGB8_PUNCHTHROUGH_ALPHA1_ETC2 0x9277
#define GL_COMPRESSED_RGBA8_ETC2_EAC 0x9278
#define GL_COMPRESSED_SRGB8_ALPHA8_ETC2_EAC 0x9279
#define GL_COMPRESSED_R11_EAC 0x9270
#define GL_COMPRESSED_SIGNED_R11_EAC 0x9271
#define GL_COMPRESSED_RG11_EAC 0x9272
#define GL_COMPRESSED_SIGNED_RG11_EAC 0x9273
#define GL_PRIMITIVE_RESTART_FIXED_INDEX 0x8D69
#define GL_ANY_SAMPLES_PASSED_CONSERVATIVE 0x8D6A
#define GL_MAX_ELEMENT_INDEX 0x8D6B
#define GL_UNSIGNED_INT64_ARB 0x140F
#define GL_MAP_PERSISTENT_BIT 0x0040
#define GL_MAP_COHERENT_BIT 0x0080
#define GL_DYNAMIC_STORAGE_BIT 0x0100
#define GL_CLIENT_STORAGE_BIT 0x0200
#define GL_CLIENT_MAPPED_BUFFER_BARRIER_BIT 0x00004000
#define GL_BUFFER_IMMUTABLE_STORAGE 0x821F
#define GL_BUFFER_STORAGE_FLAGS 0x8220
#define GL_SYNC_CL_EVENT_ARB 0x8240
#define GL_SYNC_CL_EVENT_COMPLETE_ARB 0x8241
#define GL_CLEAR_TEXTURE 0x9365
#define GL_NEGATIVE_ONE_TO_ONE 0x935E
#define GL_ZERO_TO_ONE 0x935F
#define GL_CLIP_ORIGIN 0x935C
#define GL_CLIP_DEPTH_MODE 0x935D
#define GL_RGBA_FLOAT_MODE_ARB 0x8820
#define GL_CLAMP_VERTEX_COLOR_ARB 0x891A
#define GL_CLAMP_FRAGMENT_COLOR_ARB 0x891B
#define GL_CLAMP_READ_COLOR_ARB 0x891C
#define GL_FIXED_ONLY_ARB 0x891D
#define GL_UNPACK_COMPRESSED_BLOCK_WIDTH 0x9127
#define GL_UNPACK_COMPRESSED_BLOCK_HEIGHT 0x9128
#define GL_UNPACK_COMPRESSED_BLOCK_DEPTH 0x9129
#define GL_UNPACK_COMPRESSED_BLOCK_SIZE 0x912A
```

```
#define GL_PACK_COMPRESSED_BLOCK_WIDTH 0x912B
#define GL_PACK_COMPRESSED_BLOCK_HEIGHT 0x912C
#define GL_PACK_COMPRESSED_BLOCK_DEPTH 0x912D
#define GL_PACK_COMPRESSED_BLOCK_SIZE 0x912E
#define GL_COMPUTE_SHADER 0x91B9
#define GL_MAX_COMPUTE_UNIFORM_BLOCKS 0x91BB
#define GL_MAX_COMPUTE_TEXTURE_IMAGE_UNITS 0x91BC
#define GL_MAX_COMPUTE_IMAGE_UNIFORMS 0x91BD
#define GL_MAX_COMPUTE_SHARED_MEMORY_SIZE 0x8262
#define GL_MAX_COMPUTE_UNIFORM_COMPONENTS 0x8263
#define GL_MAX_COMPUTE_ATOMIC_COUNTER_BUFFERS 0x8264
#define GL_MAX_COMPUTE_ATOMIC_COUNTERS 0x8265
#define GL_MAX_COMBINED_COMPUTE_UNIFORM_COMPONENTS 0x8266
#define GL_MAX_COMPUTE_WORK_GROUP_INVOCATIONS 0x90EB
#define GL_MAX_COMPUTE_WORK_GROUP_COUNT 0x91BE
#define GL_MAX_COMPUTE_WORK_GROUP_SIZE 0x91BF
#define GL_COMPUTE_WORK_GROUP_SIZE 0x8267
#define GL_UNIFORM_BLOCK_REFERENCED_BY_COMPUTE_SHADER 0x90EC
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_COMPUTE_SHADER 0x90ED
#define GL_DISPATCH_INDIRECT_BUFFER 0x90EE
#define GL_DISPATCH_INDIRECT_BUFFER_BINDING 0x90EF
#define GL_COMPUTE_SHADER_BIT 0x00000020
#define GL_MAX_COMPUTE_VARIABLE_GROUP_INVOCATIONS_ARB 0x9344
#define GL_MAX_COMPUTE_FIXED_GROUP_INVOCATIONS_ARB 0x90EB
#define GL_MAX_COMPUTE_VARIABLE_GROUP_SIZE_ARB 0x9345
#define GL_MAX_COMPUTE_FIXED_GROUP_SIZE_ARB 0x91BF
#define GL_QUERY_WAIT_INVERTED 0x8E17
#define GL_QUERY_NO_WAIT_INVERTED 0x8E18
#define GL_QUERY_BY_REGION_WAIT_INVERTED 0x8E19
#define GL_QUERY_BY_REGION_NO_WAIT_INVERTED 0x8E1A
#define GL_MAX_CULL_DISTANCES 0x82F9
#define GL_MAX_COMBINED_CLIP_AND_CULL_DISTANCES 0x82FA
#define GL_DEBUG_OUTPUT_SYNCHRONOUS_ARB 0x8242
#define GL_DEBUG_NEXT_LOGGED_MESSAGE_LENGTH_ARB 0x8243
#define GL_DEBUG_CALLBACK_FUNCTION_ARB 0x8244
#define GL_DEBUG_CALLBACK_USER_PARAM_ARB 0x8245
#define GL_DEBUG_SOURCE_API_ARB 0x8246
#define GL_DEBUG_SOURCE_WINDOW_SYSTEM_ARB 0x8247
#define GL_DEBUG_SOURCE_SHADER_COMPILER_ARB 0x8248
#define GL_DEBUG_SOURCE_THIRD_PARTY_ARB 0x8249
#define GL_DEBUG_SOURCE_APPLICATION_ARB 0x824A
#define GL_DEBUG_SOURCE_OTHER_ARB 0x824B
#define GL_DEBUG_TYPE_ERROR_ARB 0x824C
#define GL_DEBUG_TYPE_DEPRECATED_BEHAVIOR_ARB 0x824D
#define GL_DEBUG_TYPE_UNDEFINED_BEHAVIOR_ARB 0x824E
#define GL_DEBUG_TYPE_PORTABILITY_ARB 0x824F
#define GL_DEBUG_TYPE_PERFORMANCE_ARB 0x8250
#define GL_DEBUG_TYPE_OTHER_ARB 0x8251
#define GL_MAX_DEBUG_MESSAGE_LENGTH_ARB 0x9143
#define GL_MAX_DEBUG_LOGGED_MESSAGES_ARB 0x9144
#define GL_DEBUG_LOGGED_MESSAGES_ARB 0x9145
#define GL_DEBUG_SEVERITY_HIGH_ARB 0x9146
#define GL_DEBUG_SEVERITY_MEDIUM_ARB 0x9147
#define GL_DEBUG_SEVERITY_LOW_ARB 0x9148
#define GL_DEPTH_COMPONENT16_ARB 0x81A5
#define GL_DEPTH_COMPONENT24_ARB 0x81A6
#define GL_DEPTH_COMPONENT32_ARB 0x81A7
#define GL_TEXTURE_DEPTH_SIZE_ARB 0x884A
#define GL_DEPTH_TEXTURE_MODE_ARB 0x884B
#define GL_TEXTURE_TARGET 0x1006
#define GL_QUERY_TARGET 0x82EA
#define GL_MAX_DRAW_BUFFERS_ARB 0x8824
#define GL_DRAW_BUFFER0_ARB 0x8825
#define GL_DRAW_BUFFER1_ARB 0x8826
#define GL_DRAW_BUFFER2_ARB 0x8827
#define GL_DRAW_BUFFER3_ARB 0x8828
#define GL_DRAW_BUFFER4_ARB 0x8829
#define GL_DRAW_BUFFER5_ARB 0x882A
#define GL_DRAW_BUFFER6_ARB 0x882B
#define GL_DRAW_BUFFER7_ARB 0x882C
#define GL_DRAW_BUFFER8_ARB 0x882D
#define GL_DRAW_BUFFER9_ARB 0x882E
#define GL_DRAW_BUFFER10_ARB 0x882F
#define GL_DRAW_BUFFER11_ARB 0x8830
#define GL_DRAW_BUFFER12_ARB 0x8831
#define GL_DRAW_BUFFER13_ARB 0x8832
#define GL_DRAW_BUFFER14_ARB 0x8833
#define GL_DRAW_BUFFER15_ARB 0x8834
#define GL_LOCATION_COMPONENT 0x934A
#define GL_TRANSFORM_FEEDBACK_BUFFER_INDEX 0x934B
#define GL_TRANSFORM_FEEDBACK_BUFFER_STRIDE 0x934C
#define GL_MAX_UNIFORM_LOCATIONS 0x826E
#define GL_FRAGMENT_PROGRAM_ARB 0x8804
#define GL_PROGRAM_FORMAT_ASCII_ARB 0x8875
#define GL_PROGRAM_LENGTH_ARB 0x8627
#define GL_PROGRAM_FORMAT_ARB 0x8876
#define GL_PROGRAM_BINDING_ARB 0x8677
```

```
#define GL_PROGRAM_INSTRUCTIONS_ARB 0x88A0
#define GL_MAX_PROGRAM_INSTRUCTIONS_ARB 0x88A1
#define GL_PROGRAM_NATIVE_INSTRUCTIONS_ARB 0x88A2
#define GL_MAX_PROGRAM_NATIVE_INSTRUCTIONS_ARB 0x88A3
#define GL_PROGRAM_TEMPORARIES_ARB 0x88A4
#define GL_MAX_PROGRAM_TEMPORARIES_ARB 0x88A5
#define GL_PROGRAM_NATIVE_TEMPORARIES_ARB 0x88A6
#define GL_MAX_PROGRAM_NATIVE_TEMPORARIES_ARB 0x88A7
#define GL_PROGRAM_PARAMETERS_ARB 0x88A8
#define GL_MAX_PROGRAM_PARAMETERS_ARB 0x88A9
#define GL_PROGRAM_NATIVE_PARAMETERS_ARB 0x88AA
#define GL_MAX_PROGRAM_NATIVE_PARAMETERS_ARB 0x88AB
#define GL_PROGRAM_ATTRIBS_ARB 0x88AC
#define GL_MAX_PROGRAM_ATTRIBS_ARB 0x88AD
#define GL_PROGRAM_NATIVE_ATTRIBS_ARB 0x88AE
#define GL_MAX_PROGRAM_NATIVE_ATTRIBS_ARB 0x88AF
#define GL_MAX_PROGRAM_LOCAL_PARAMETERS_ARB 0x88B4
#define GL_MAX_PROGRAM_ENV_PARAMETERS_ARB 0x88B5
#define GL_PROGRAM_UNDER_NATIVE_LIMITS_ARB 0x88B6
#define GL_PROGRAM_ALU_INSTRUCTIONS_ARB 0x8805
#define GL_PROGRAM_TEX_INSTRUCTIONS_ARB 0x8806
#define GL_PROGRAM_TEX_INDIRECTIONS_ARB 0x8807
#define GL_PROGRAM_NATIVE_ALU_INSTRUCTIONS_ARB 0x8808
#define GL_PROGRAM_NATIVE_TEX_INSTRUCTIONS_ARB 0x8809
#define GL_PROGRAM_NATIVE_TEX_INDIRECTIONS_ARB 0x880A
#define GL_MAX_PROGRAM_ALU_INSTRUCTIONS_ARB 0x880B
#define GL_MAX_PROGRAM_TEX_INSTRUCTIONS_ARB 0x880C
#define GL_MAX_PROGRAM_TEX_INDIRECTIONS_ARB 0x880D
#define GL_MAX_PROGRAM_NATIVE_ALU_INSTRUCTIONS_ARB 0x880E
#define GL_MAX_PROGRAM_NATIVE_TEX_INSTRUCTIONS_ARB 0x880F
#define GL_MAX_PROGRAM_NATIVE_TEX_INDIRECTIONS_ARB 0x8810
#define GL_PROGRAM_STRING_ARB 0x8628
#define GL_PROGRAM_ERROR_POSITION_ARB 0x864B
#define GL_CURRENT_MATRIX_ARB 0x8641
#define GL_TRANSPOSE_CURRENT_MATRIX_ARB 0x88B7
#define GL_CURRENT_MATRIX_STACK_DEPTH_ARB 0x8640
#define GL_MAX_PROGRAM_MATRICES_ARB 0x862F
#define GL_MAX_PROGRAM_MATRIX_STACK_DEPTH_ARB 0x862E
#define GL_MAX_TEXTURE_COORDS_ARB 0x8871
#define GL_MAX_TEXTURE_IMAGE_UNITS_ARB 0x8872
#define GL_PROGRAM_ERROR_STRING_ARB 0x8874
#define GL_MATRIX0_ARB 0x88C0
#define GL_MATRIX1_ARB 0x88C1
#define GL_MATRIX2_ARB 0x88C2
#define GL_MATRIX3_ARB 0x88C3
#define GL_MATRIX4_ARB 0x88C4
#define GL_MATRIX5_ARB 0x88C5
#define GL_MATRIX6_ARB 0x88C6
#define GL_MATRIX7_ARB 0x88C7
#define GL_MATRIX8_ARB 0x88C8
#define GL_MATRIX9_ARB 0x88C9
#define GL_MATRIX10_ARB 0x88CA
#define GL_MATRIX11_ARB 0x88CB
#define GL_MATRIX12_ARB 0x88CC
#define GL_MATRIX13_ARB 0x88CD
#define GL_MATRIX14_ARB 0x88CE
#define GL_MATRIX15_ARB 0x88CF
#define GL_MATRIX16_ARB 0x88D0
#define GL_MATRIX17_ARB 0x88D1
#define GL_MATRIX18_ARB 0x88D2
#define GL_MATRIX19_ARB 0x88D3
#define GL_MATRIX20_ARB 0x88D4
#define GL_MATRIX21_ARB 0x88D5
#define GL_MATRIX22_ARB 0x88D6
#define GL_MATRIX23_ARB 0x88D7
#define GL_MATRIX24_ARB 0x88D8
#define GL_MATRIX25_ARB 0x88D9
#define GL_MATRIX26_ARB 0x88DA
#define GL_MATRIX27_ARB 0x88DB
#define GL_MATRIX28_ARB 0x88DC
#define GL_MATRIX29_ARB 0x88DD
#define GL_MATRIX30_ARB 0x88DE
#define GL_MATRIX31_ARB 0x88DF
#define GL_FRAGMENT_SHADER_ARB 0x8B30
#define GL_MAX_FRAGMENT_UNIFORM_COMPONENTS_ARB 0x8B49
#define GL_FRAGMENT_SHADER_DERIVATIVE_HINT_ARB 0x8B8B
#define GL_FRAMEBUFFER_DEFAULT_WIDTH 0x9310
#define GL_FRAMEBUFFER_DEFAULT_HEIGHT 0x9311
#define GL_FRAMEBUFFER_DEFAULT_LAYERS 0x9312
#define GL_FRAMEBUFFER_DEFAULT_SAMPLES 0x9313
#define GL_FRAMEBUFFER_DEFAULT_FIXED_SAMPLE_LOCATIONS 0x9314
#define GL_MAX_FRAMEBUFFER_WIDTH 0x9315
#define GL_MAX_FRAMEBUFFER_HEIGHT 0x9316
#define GL_MAX_FRAMEBUFFER_LAYERS 0x9317
#define GL_MAX_FRAMEBUFFER_SAMPLES 0x9318
#define GL_INDEX 0x8222
#define GL_LINES_ADJACENCY_ARB 0x000A
```

```
#define GL_LINE_STRIP_ADJACENCY_ARB 0x000B
#define GL_TRIANGLES_ADJACENCY_ARB 0x000C
#define GL_TRIANGLE_STRIP_ADJACENCY_ARB 0x000D
#define GL_PROGRAM_POINT_SIZE_ARB 0x8642
#define GL_MAX_GEOMETRY_TEXTURE_IMAGE_UNITS_ARB 0x8C29
#define GL_FRAMEBUFFER_ATTACHMENT_LAYERED_ARB 0x8DA7
#define GL_FRAMEBUFFER_INCOMPLETE_LAYER_TARGETS_ARB 0x8DA8
#define GL_FRAMEBUFFER_INCOMPLETE_LAYER_COUNT_ARB 0x8DA9
#define GL_GEOMETRY_SHADER_ARB 0x8DD9
#define GL_GEOMETRY_VERTICES_OUT_ARB 0x8DDA
#define GL_GEOMETRY_INPUT_TYPE_ARB 0x8DDB
#define GL_GEOMETRY_OUTPUT_TYPE_ARB 0x8DDC
#define GL_MAX_GEOMETRY_VARYING_COMPONENTS_ARB 0x8DDD
#define GL_MAX_VERTEX_VARYING_COMPONENTS_ARB 0x8DDE
#define GL_MAX_GEOMETRY_UNIFORM_COMPONENTS_ARB 0x8DDF
#define GL_MAX_GEOMETRY_OUTPUT_VERTICES_ARB 0x8DE0
#define GL_MAX_GEOMETRY_TOTAL_OUTPUT_COMPONENTS_ARB 0x8DE1
#define GL_SHADER_BINARY_FORMAT_SPIR_V_ARB 0x9551
#define GL_SPIR_V_BINARY_ARB 0x9552
#define GL_INT64_ARB 0x140E
#define GL_INT64_VEC2_ARB 0x8FE9
#define GL_INT64_VEC3_ARB 0x8FEA
#define GL_INT64_VEC4_ARB 0x8FEB
#define GL_UNSIGNED_INT64_VEC2_ARB 0x8FF5
#define GL_UNSIGNED_INT64_VEC3_ARB 0x8FF6
#define GL_UNSIGNED_INT64_VEC4_ARB 0x8FF7
#define GL_HALF_FLOAT_ARB 0x140B
#define GL_CONVOLUTION_1D 0x8010
#define GL_CONVOLUTION_2D 0x8011
#define GL_SEPARABLE_2D 0x8012
#define GL_CONVOLUTION_BORDER_MODE 0x8013
#define GL_CONVOLUTION_FILTER_SCALE 0x8014
#define GL_CONVOLUTION_FILTER_BIAS 0x8015
#define GL_REDUCE 0x8016
#define GL_CONVOLUTION_FORMAT 0x8017
#define GL_CONVOLUTION_WIDTH 0x8018
#define GL_CONVOLUTION_HEIGHT 0x8019
#define GL_MAX_CONVOLUTION_WIDTH 0x801A
#define GL_MAX_CONVOLUTION_HEIGHT 0x801B
#define GL_POST_CONVOLUTION_RED_SCALE 0x801C
#define GL_POST_CONVOLUTION_GREEN_SCALE 0x801D
#define GL_POST_CONVOLUTION_BLUE_SCALE 0x801E
#define GL_POST_CONVOLUTION_ALPHA_SCALE 0x801F
#define GL_POST_CONVOLUTION_RED_BIAS 0x8020
#define GL_POST_CONVOLUTION_GREEN_BIAS 0x8021
#define GL_POST_CONVOLUTION_BLUE_BIAS 0x8022
#define GL_POST_CONVOLUTION_ALPHA_BIAS 0x8023
#define GL_HISTOGRAM 0x8024
#define GL_PROXY_HISTOGRAM 0x8025
#define GL_HISTOGRAM_WIDTH 0x8026
#define GL_HISTOGRAM_FORMAT 0x8027
#define GL_HISTOGRAM_RED_SIZE 0x8028
#define GL_HISTOGRAM_GREEN_SIZE 0x8029
#define GL_HISTOGRAM_BLUE_SIZE 0x802A
#define GL_HISTOGRAM_ALPHA_SIZE 0x802B
#define GL_HISTOGRAM_LUMINANCE_SIZE 0x802C
#define GL_HISTOGRAM_SINK 0x802D
#define GL_MINMAX 0x802E
#define GL_MINMAX_FORMAT 0x802F
#define GL_MINMAX_SINK 0x8030
#define GL_TABLE_TOO_LARGE 0x8031
#define GL_COLOR_MATRIX 0x80B1
#define GL_COLOR_MATRIX_STACK_DEPTH 0x80B2
#define GL_MAX_COLOR_MATRIX_STACK_DEPTH 0x80B3
#define GL_POST_COLOR_MATRIX_RED_SCALE 0x80B4
#define GL_POST_COLOR_MATRIX_GREEN_SCALE 0x80B5
#define GL_POST_COLOR_MATRIX_BLUE_SCALE 0x80B6
#define GL_POST_COLOR_MATRIX_ALPHA_SCALE 0x80B7
#define GL_POST_COLOR_MATRIX_RED_BIAS 0x80B8
#define GL_POST_COLOR_MATRIX_GREEN_BIAS 0x80B9
#define GL_POST_COLOR_MATRIX_BLUE_BIAS 0x80BA
#define GL_POST_COLOR_MATRIX_ALPHA_BIAS 0x80BB
#define GL_COLOR_TABLE 0x80D0
#define GL_POST_CONVOLUTION_COLOR_TABLE 0x80D1
#define GL_POST_COLOR_MATRIX_COLOR_TABLE 0x80D2
#define GL_PROXY_COLOR_TABLE 0x80D3
#define GL_PROXY_POST_CONVOLUTION_COLOR_TABLE 0x80D4
#define GL_PROXY_POST_COLOR_MATRIX_COLOR_TABLE 0x80D5
#define GL_COLOR_TABLE_SCALE 0x80D6
#define GL_COLOR_TABLE_BIAS 0x80D7
#define GL_COLOR_TABLE_FORMAT 0x80D8
#define GL_COLOR_TABLE_WIDTH 0x80D9
#define GL_COLOR_TABLE_RED_SIZE 0x80DA
#define GL_COLOR_TABLE_GREEN_SIZE 0x80DB
#define GL_COLOR_TABLE_BLUE_SIZE 0x80DC
#define GL_COLOR_TABLE_ALPHA_SIZE 0x80DD
#define GL_COLOR_TABLE_LUMINANCE_SIZE 0x80DE
```

```
#define GL_COLOR_TABLE_INTENSITY_SIZE 0x80DF
#define GL_CONSTANT_BORDER 0x8151
#define GL_REPLICATE_BORDER 0x8153
#define GL_CONVOLUTION_BORDER_COLOR 0x8154
#define GL_PARAMETER_BUFFER_ARB 0x80EE
#define GL_PARAMETER_BUFFER_BINDING_ARB 0x80EF
#define GL_VERTEX_ATTRIB_ARRAY_DIVISOR_ARB 0x88FE
#define GL_NUM_SAMPLE_COUNTS 0x9380
#define GL_IMAGE_FORMAT_COMPATIBILITY_TYPE 0x90C7
#define GL_INTERNALFORMAT_SUPPORTED 0x826F
#define GL_INTERNALFORMAT_PREFERRED 0x8270
#define GL_INTERNALFORMAT_RED_SIZE 0x8271
#define GL_INTERNALFORMAT_GREEN_SIZE 0x8272
#define GL_INTERNALFORMAT_BLUE_SIZE 0x8273
#define GL_INTERNALFORMAT_ALPHA_SIZE 0x8274
#define GL_INTERNALFORMAT_DEPTH_SIZE 0x8275
#define GL_INTERNALFORMAT_STENCIL_SIZE 0x8276
#define GL_INTERNALFORMAT_SHARED_SIZE 0x8277
#define GL_INTERNALFORMAT_RED_TYPE 0x8278
#define GL_INTERNALFORMAT_GREEN_TYPE 0x8279
#define GL_INTERNALFORMAT_BLUE_TYPE 0x827A
#define GL_INTERNALFORMAT_ALPHA_TYPE 0x827B
#define GL_INTERNALFORMAT_DEPTH_TYPE 0x827C
#define GL_INTERNALFORMAT_STENCIL_TYPE 0x827D
#define GL_MAX_WIDTH 0x827E
#define GL_MAX_HEIGHT 0x827F
#define GL_MAX_DEPTH 0x8280
#define GL_MAX_LAYERS 0x8281
#define GL_MAX_COMBINED_DIMENSIONS 0x8282
#define GL_COLOR_COMPONENTS 0x8283
#define GL_DEPTH_COMPONENTS 0x8284
#define GL_STENCIL_COMPONENTS 0x8285
#define GL_COLOR_RENDERABLE 0x8286
#define GL_DEPTH_RENDERABLE 0x8287
#define GL_STENCIL_RENDERABLE 0x8288
#define GL_FRAMEBUFFER_RENDERABLE 0x8289
#define GL_FRAMEBUFFER_RENDERABLE_LAYERED 0x828A
#define GL_FRAMEBUFFER_BLEND 0x828B
#define GL_READ_PIXELS 0x828C
#define GL_READ_PIXELS_FORMAT 0x828D
#define GL_READ_PIXELS_TYPE 0x828E
#define GL_TEXTURE_IMAGE_FORMAT 0x828F
#define GL_TEXTURE_IMAGE_TYPE 0x8290
#define GL_GET_TEXTURE_IMAGE_FORMAT 0x8291
#define GL_GET_TEXTURE_IMAGE_TYPE 0x8292
#define GL_MIPMAP 0x8293
#define GL_MANUAL_GENERATE_MIPMAP 0x8294
#define GL_AUTO_GENERATE_MIPMAP 0x8295
#define GL_COLOR_ENCODING 0x8296
#define GL_SRGB_READ 0x8297
#define GL_SRGB_WRITE 0x8298
#define GL_SRGB_DECODE_ARB 0x8299
#define GL_FILTER 0x829A
#define GL_VERTEX_TEXTURE 0x829B
#define GL_TESS_CONTROL_TEXTURE 0x829C
#define GL_TESS_EVALUATION_TEXTURE 0x829D
#define GL_GEOMETRY_TEXTURE 0x829E
#define GL_FRAGMENT_TEXTURE 0x829F
#define GL_COMPUTE_TEXTURE 0x82A0
#define GL_TEXTURE_SHADOW 0x82A1
#define GL_TEXTURE_GATHER 0x82A2
#define GL_TEXTURE_GATHER_SHADOW 0x82A3
#define GL_SHADER_IMAGE_LOAD 0x82A4
#define GL_SHADER_IMAGE_STORE 0x82A5
#define GL_SHADER_IMAGE_ATOMIC 0x82A6
#define GL_IMAGE_TEXEL_SIZE 0x82A7
#define GL_IMAGE_COMPATIBILITY_CLASS 0x82A8
#define GL_IMAGE_PIXEL_FORMAT 0x82A9
#define GL_IMAGE_PIXEL_TYPE 0x82AA
#define GL_SIMULTANEOUS_TEXTURE_AND_DEPTH_TEST 0x82AC
#define GL_SIMULTANEOUS_TEXTURE_AND_STENCIL_TEST 0x82AD
#define GL_SIMULTANEOUS_TEXTURE_AND_DEPTH_WRITE 0x82AE
#define GL_SIMULTANEOUS_TEXTURE_AND_STENCIL_WRITE 0x82AF
#define GL_TEXTURE_COMPRESSED_BLOCK_WIDTH 0x82B1
#define GL_TEXTURE_COMPRESSED_BLOCK_HEIGHT 0x82B2
#define GL_TEXTURE_COMPRESSED_BLOCK_SIZE 0x82B3
#define GL_CLEAR_BUFFER 0x82B4
#define GL_TEXTURE_VIEW 0x82B5
#define GL_VIEW_COMPATIBILITY_CLASS 0x82B6
#define GL_FULL_SUPPORT 0x82B7
#define GL_CAVEAT_SUPPORT 0x82B8
#define GL_IMAGE_CLASS_4_X_32 0x82B9
#define GL_IMAGE_CLASS_2_X_32 0x82BA
#define GL_IMAGE_CLASS_1_X_32 0x82BB
#define GL_IMAGE_CLASS_4_X_16 0x82BC
#define GL_IMAGE_CLASS_2_X_16 0x82BD
#define GL_IMAGE_CLASS_1_X_16 0x82BE
```

```
#define GL_IMAGE_CLASS_4_X_8 0x82BF
#define GL_IMAGE_CLASS_2_X_8 0x82C0
#define GL_IMAGE_CLASS_1_X_8 0x82C1
#define GL_IMAGE_CLASS_11_11_10 0x82C2
#define GL_IMAGE_CLASS_10_10_10_2 0x82C3
#define GL_VIEW_CLASS_128_BITS 0x82C4
#define GL_VIEW_CLASS_96_BITS 0x82C5
#define GL_VIEW_CLASS_64_BITS 0x82C6
#define GL_VIEW_CLASS_48_BITS 0x82C7
#define GL_VIEW_CLASS_32_BITS 0x82C8
#define GL_VIEW_CLASS_24_BITS 0x82C9
#define GL_VIEW_CLASS_16_BITS 0x82CA
#define GL_VIEW_CLASS_8_BITS 0x82CB
#define GL_VIEW_CLASS_S3TC_DXT1_RGB 0x82CC
#define GL_VIEW_CLASS_S3TC_DXT1_RGBA 0x82CD
#define GL_VIEW_CLASS_S3TC_DXT3_RGBA 0x82CE
#define GL_VIEW_CLASS_S3TC_DXT5_RGBA 0x82CF
#define GL_VIEW_CLASS_RGTC1_RED 0x82D0
#define GL_VIEW_CLASS_RGTC2_RG 0x82D1
#define GL_VIEW_CLASS_BPTC_UNORM 0x82D2
#define GL_VIEW_CLASS_BPTC_FLOAT 0x82D3
#define GL_VIEW_CLASS_EAC_R11 0x9383
#define GL_VIEW_CLASS_EAC_RG11 0x9384
#define GL_VIEW_CLASS_ETC2_RGB 0x9385
#define GL_VIEW_CLASS_ETC2_RGBA 0x9386
#define GL_VIEW_CLASS_ETC2_EAC_RGBA 0x9387
#define GL_VIEW_CLASS_ASTC_4x4_RGBA 0x9388
#define GL_VIEW_CLASS_ASTC_5x4_RGBA 0x9389
#define GL_VIEW_CLASS_ASTC_5x5_RGBA 0x938A
#define GL_VIEW_CLASS_ASTC_6x5_RGBA 0x938B
#define GL_VIEW_CLASS_ASTC_6x6_RGBA 0x938C
#define GL_VIEW_CLASS_ASTC_8x5_RGBA 0x938D
#define GL_VIEW_CLASS_ASTC_8x6_RGBA 0x938E
#define GL_VIEW_CLASS_ASTC_8x8_RGBA 0x938F
#define GL_VIEW_CLASS_ASTC_10x5_RGBA 0x9390
#define GL_VIEW_CLASS_ASTC_10x6_RGBA 0x9391
#define GL_VIEW_CLASS_ASTC_10x8_RGBA 0x9392
#define GL_VIEW_CLASS_ASTC_10x10_RGBA 0x9393
#define GL_VIEW_CLASS_ASTC_12x10_RGBA 0x9394
#define GL_VIEW_CLASS_ASTC_12x12_RGBA 0x9395
#define GL_MIN_MAP_BUFFER_ALIGNMENT 0x90BC
#define GL_MATRIX_PALETTE_ARB 0x8840
#define GL_MAX_MATRIX_PALETTE_STACK_DEPTH_ARB 0x8841
#define GL_MAX_PALETTE_MATRICES_ARB 0x8842
#define GL_CURRENT_PALETTE_MATRIX_ARB 0x8843
#define GL_MATRIX_INDEX_ARRAY_ARB 0x8844
#define GL_CURRENT_MATRIX_INDEX_ARB 0x8845
#define GL_MATRIX_INDEX_ARRAY_SIZE_ARB 0x8846
#define GL_MATRIX_INDEX_ARRAY_TYPE_ARB 0x8847
#define GL_MATRIX_INDEX_ARRAY_STRIDE_ARB 0x8848
#define GL_MATRIX_INDEX_ARRAY_POINTER_ARB 0x8849
#define GL_MULTISAMPLE_ARB 0x809D
#define GL_SAMPLE_ALPHA_TO_COVERAGE_ARB 0x809E
#define GL_SAMPLE_ALPHA_TO_ONE_ARB 0x809F
#define GL_SAMPLE_COVERAGE_ARB 0x80A0
#define GL_SAMPLE_BUFFERS_ARB 0x80A8
#define GL_SAMPLES_ARB 0x80A9
#define GL_SAMPLE_COVERAGE_VALUE_ARB 0x80AA
#define GL_SAMPLE_COVERAGE_INVERT_ARB 0x80AB
#define GL_MULTISAMPLE_BIT_ARB 0x20000000
#define GL_TEXTURE0_ARB 0x84C0
#define GL_TEXTURE1_ARB 0x84C1
#define GL_TEXTURE2_ARB 0x84C2
#define GL_TEXTURE3_ARB 0x84C3
#define GL_TEXTURE4_ARB 0x84C4
#define GL_TEXTURE5_ARB 0x84C5
#define GL_TEXTURE6_ARB 0x84C6
#define GL_TEXTURE7_ARB 0x84C7
#define GL_TEXTURE8_ARB 0x84C8
#define GL_TEXTURE9_ARB 0x84C9
#define GL_TEXTURE10_ARB 0x84CA
#define GL_TEXTURE11_ARB 0x84CB
#define GL_TEXTURE12_ARB 0x84CC
#define GL_TEXTURE13_ARB 0x84CD
#define GL_TEXTURE14_ARB 0x84CE
#define GL_TEXTURE15_ARB 0x84CF
#define GL_TEXTURE16_ARB 0x84D0
#define GL_TEXTURE17_ARB 0x84D1
#define GL_TEXTURE18_ARB 0x84D2
#define GL_TEXTURE19_ARB 0x84D3
#define GL_TEXTURE20_ARB 0x84D4
#define GL_TEXTURE21_ARB 0x84D5
#define GL_TEXTURE22_ARB 0x84D6
#define GL_TEXTURE23_ARB 0x84D7
#define GL_TEXTURE24_ARB 0x84D8
#define GL_TEXTURE25_ARB 0x84D9
#define GL_TEXTURE26_ARB 0x84DA
```

```
#define GL_TEXTURE27_ARB 0x84DB
#define GL_TEXTURE28_ARB 0x84DC
#define GL_TEXTURE29_ARB 0x84DD
#define GL_TEXTURE30_ARB 0x84DE
#define GL_TEXTURE31_ARB 0x84DF
#define GL_ACTIVE_TEXTURE_ARB 0x84E0
#define GL_CLIENT_ACTIVE_TEXTURE_ARB 0x84E1
#define GL_MAX_TEXTURE_UNITS_ARB 0x84E2
#define GL_QUERY_COUNTER_BITS_ARB 0x8864
#define GL_CURRENT_QUERY_ARB 0x8865
#define GL_QUERY_RESULT_ARB 0x8866
#define GL_QUERY_RESULT_AVAILABLE_ARB 0x8867
#define GL_SAMPLES_PASSED_ARB 0x8914
#define GL_MAX_SHADER_COMPILER_THREADS_ARB 0x91B0
#define GL_COMPLETION_STATUS_ARB 0x91B1
#define GL_VERTICES_SUBMITTED_ARB 0x82EE
#define GL_PRIMITIVES_SUBMITTED_ARB 0x82EF
#define GL_VERTEX_SHADER_INVOCATIONS_ARB 0x82F0
#define GL_TESS_CONTROL_SHADER_PATCHES_ARB 0x82F1
#define GL_TESS_EVALUATION_SHADER_INVOCATIONS_ARB 0x82F2
#define GL_GEOMETRY_SHADER_PRIMITIVES_EMITTED_ARB 0x82F3
#define GL_FRAGMENT_SHADER_INVOCATIONS_ARB 0x82F4
#define GL_COMPUTE_SHADER_INVOCATIONS_ARB 0x82F5
#define GL_CLIPPING_INPUT_PRIMITIVES_ARB 0x82F6
#define GL_CLIPPING_OUTPUT_PRIMITIVES_ARB 0x82F7
#define GL_PIXEL_PACK_BUFFER_ARB 0x88EB
#define GL_PIXEL_UNPACK_BUFFER_ARB 0x88EC
#define GL_PIXEL_PACK_BUFFER_BINDING_ARB 0x88ED
#define GL_PIXEL_UNPACK_BUFFER_BINDING_ARB 0x88EF
#define GL_POINT_SIZE_MIN_ARB 0x8126
#define GL_POINT_SIZE_MAX_ARB 0x8127
#define GL_POINT_FADE_THRESHOLD_SIZE_ARB 0x8128
#define GL_POINT_DISTANCE_ATTENUATION_ARB 0x8129
#define GL_POINT_SPRITE_ARB 0x8861
#define GL_COORD_REPLACE_ARB 0x8862
#define GL_POLYGON_OFFSET_CLAMP 0x8E1B
#define GL_UNIFORM 0x92E1
#define GL_UNIFORM_BLOCK 0x92E2
#define GL_PROGRAM_INPUT 0x92E3
#define GL_PROGRAM_OUTPUT 0x92E4
#define GL_BUFFER_VARIABLE 0x92E5
#define GL_SHADER_STORAGE_BLOCK 0x92E6
#define GL_ATOMIC_COUNTER_BUFFER 0x92C0
#define GL_VERTEX_SUBROUTINE 0x92E8
#define GL_TESS_CONTROL_SUBROUTINE 0x92E9
#define GL_TESS_EVALUATION_SUBROUTINE 0x92EA
#define GL_GEOMETRY_SUBROUTINE 0x92EB
#define GL_FRAGMENT_SUBROUTINE 0x92EC
#define GL_COMPUTE_SUBROUTINE 0x92ED
#define GL_VERTEX_SUBROUTINE_UNIFORM 0x92EE
#define GL_TESS_CONTROL_SUBROUTINE_UNIFORM 0x92EF
#define GL_TESS_EVALUATION_SUBROUTINE_UNIFORM 0x92F0
#define GL_GEOMETRY_SUBROUTINE_UNIFORM 0x92F1
#define GL_FRAGMENT_SUBROUTINE_UNIFORM 0x92F2
#define GL_COMPUTE_SUBROUTINE_UNIFORM 0x92F3
#define GL_TRANSFORM_FEEDBACK_VARYING 0x92F4
#define GL_ACTIVE_RESOURCES 0x92F5
#define GL_MAX_NAME_LENGTH 0x92F6
#define GL_MAX_NUM_ACTIVE_VARIABLES 0x92F7
#define GL_MAX_NUM_COMPATIBLE_SUBROUTINES 0x92F8
#define GL_NAME_LENGTH 0x92F9
#define GL_TYPE 0x92FA
#define GL_ARRAY_SIZE 0x92FB
#define GL_OFFSET 0x92FC
#define GL_BLOCK_INDEX 0x92FD
#define GL_ARRAY_STRIDE 0x92FE
#define GL_MATRIX_STRIDE 0x92FF
#define GL_IS_ROW_MAJOR 0x9300
#define GL_ATOMIC_COUNTER_BUFFER_INDEX 0x9301
#define GL_BUFFER_BINDING 0x9302
#define GL_BUFFER_DATA_SIZE 0x9303
#define GL_NUM_ACTIVE_VARIABLES 0x9304
#define GL_ACTIVE_VARIABLES 0x9305
#define GL_REFERENCED_BY_VERTEX_SHADER 0x9306
#define GL_REFERENCED_BY_TESS_CONTROL_SHADER 0x9307
#define GL_REFERENCED_BY_TESS_EVALUATION_SHADER 0x9308
#define GL_REFERENCED_BY_GEOMETRY_SHADER 0x9309
#define GL_REFERENCED_BY_FRAGMENT_SHADER 0x930A
#define GL_REFERENCED_BY_COMPUTE_SHADER 0x930B
#define GL_TOP_LEVEL_ARRAY_SIZE 0x930C
#define GL_TOP_LEVEL_ARRAY_STRIDE 0x930D
#define GL_LOCATION 0x930E
#define GL_LOCATION_INDEX 0x930F
#define GL_IS_PER_PATCH 0x92E7
#define GL_QUERY_BUFFER 0x9192
#define GL_QUERY_BUFFER_BARRIER_BIT 0x00008000
#define GL_QUERY_BUFFER_BINDING 0x9193
```

```
#define GL_QUERY_RESULT_NO_WAIT 0x9194
#define GL_CONTEXT_FLAG_ROBUST_ACCESS_BIT_ARB 0x00000004
#define GL_LOSE_CONTEXT_ON_RESET_ARB 0x8252
#define GL_GUILTY_CONTEXT_RESET_ARB 0x8253
#define GL_INNOCENT_CONTEXT_RESET_ARB 0x8254
#define GL_UNKNOWN_CONTEXT_RESET_ARB 0x8255
#define GL_RESET_NOTIFICATION_STRATEGY_ARB 0x8256
#define GL_NO_RESET_NOTIFICATION_ARB 0x8261
#define GL_SAMPLE_LOCATION_SUBPIXEL_BITS_ARB 0x933D
#define GL_SAMPLE_LOCATION_PIXEL_GRID_WIDTH_ARB 0x933E
#define GL_SAMPLE_LOCATION_PIXEL_GRID_HEIGHT_ARB 0x933F
#define GL_PROGRAMMABLE_SAMPLE_LOCATION_TABLE_SIZE_ARB 0x9340
#define GL_SAMPLE_LOCATION_ARB 0x8E50
#define GL_PROGRAMMABLE_SAMPLE_LOCATION_ARB 0x9341
#define GL_FRAMEBUFFER_PROGRAMMABLE_SAMPLE_LOCATIONS_ARB 0x9342
#define GL_FRAMEBUFFER_SAMPLE_LOCATION_PIXEL_GRID_ARB 0x9343
#define GL_SAMPLE_SHADING_ARB 0x8C36
#define GL_MIN_SAMPLE_SHADING_VALUE_ARB 0x8C37
#define GL_ATOMIC_COUNTER_BUFFER_BINDING 0x92C1
#define GL_ATOMIC_COUNTER_BUFFER_START 0x92C2
#define GL_ATOMIC_COUNTER_BUFFER_SIZE 0x92C3
#define GL_ATOMIC_COUNTER_BUFFER_DATA_SIZE 0x92C4
#define GL_ATOMIC_COUNTER_BUFFER_ACTIVE_ATOMIC_COUNTERS 0x92C5
#define GL_ATOMIC_COUNTER_BUFFER_ACTIVE_ATOMIC_COUNTER_INDICES 0x92C6
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_VERTEX_SHADER 0x92C7
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_TESS_CONTROL_SHADER 0x92C8
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_TESS_EVALUATION_SHADER 0x92C9
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_GEOMETRY_SHADER 0x92CA
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_FRAGMENT_SHADER 0x92CB
#define GL_MAX_VERTEX_ATOMIC_COUNTER_BUFFERS 0x92CC
#define GL_MAX_TESS_CONTROL_ATOMIC_COUNTER_BUFFERS 0x92CD
#define GL_MAX_TESS_EVALUATION_ATOMIC_COUNTER_BUFFERS 0x92CE
#define GL_MAX_GEOMETRY_ATOMIC_COUNTER_BUFFERS 0x92CF
#define GL_MAX_FRAGMENT_ATOMIC_COUNTER_BUFFERS 0x92D0
#define GL_MAX_COMBINED_ATOMIC_COUNTER_BUFFERS 0x92D1
#define GL_MAX_VERTEX_ATOMIC_COUNTERS 0x92D2
#define GL_MAX_TESS_CONTROL_ATOMIC_COUNTERS 0x92D3
#define GL_MAX_TESS_EVALUATION_ATOMIC_COUNTERS 0x92D4
#define GL_MAX_GEOMETRY_ATOMIC_COUNTERS 0x92D5
#define GL_MAX_FRAGMENT_ATOMIC_COUNTERS 0x92D6
#define GL_MAX_COMBINED_ATOMIC_COUNTERS 0x92D7
#define GL_MAX_ATOMIC_COUNTER_BUFFER_SIZE 0x92D8
#define GL_MAX_ATOMIC_COUNTER_BUFFER_BINDINGS 0x92DC
#define GL_ACTIVE_ATOMIC_COUNTER_BUFFERS 0x92D9
#define GL_UNIFORM_ATOMIC_COUNTER_BUFFER_INDEX 0x92DA
#define GL_UNSIGNED_INT_ATOMIC_COUNTER 0x92DB
#define GL_VERTEX_ATTRIB_ARRAY_BARRIER_BIT 0x00000001
#define GL_ELEMENT_ARRAY_BARRIER_BIT 0x00000002
#define GL_UNIFORM_BARRIER_BIT 0x00000004
#define GL_TEXTURE_FETCH_BARRIER_BIT 0x00000008
#define GL_SHADER_IMAGE_ACCESS_BARRIER_BIT 0x00000020
#define GL_COMMAND_BARRIER_BIT 0x00000040
#define GL_PIXEL_BUFFER_BARRIER_BIT 0x00000080
#define GL_TEXTURE_UPDATE_BARRIER_BIT 0x00000100
#define GL_BUFFER_UPDATE_BARRIER_BIT 0x00000200
#define GL_FRAMEBUFFER_BARRIER_BIT 0x00000400
#define GL_TRANSFORM_FEEDBACK_BARRIER_BIT 0x00000800
#define GL_ATOMIC_COUNTER_BARRIER_BIT 0x00001000
#define GL_ALL_BARRIER_BITS 0xFFFFFFFF
#define GL_MAX_IMAGE_UNITS 0x8F38
#define GL_MAX_COMBINED_IMAGE_UNITS_AND_FRAGMENT_OUTPUTS 0x8F39
#define GL_IMAGE_BINDING_NAME 0x8F3A
#define GL_IMAGE_BINDING_LEVEL 0x8F3B
#define GL_IMAGE_BINDING_LAYERED 0x8F3C
#define GL_IMAGE_BINDING_LAYER 0x8F3D
#define GL_IMAGE_BINDING_ACCESS 0x8F3E
#define GL_IMAGE_1D 0x904C
#define GL_IMAGE_2D 0x904D
#define GL_IMAGE_3D 0x904E
#define GL_IMAGE_2D_RECT 0x904F
#define GL_IMAGE_CUBE 0x9050
#define GL_IMAGE_BUFFER 0x9051
#define GL_IMAGE_1D_ARRAY 0x9052
#define GL_IMAGE_2D_ARRAY 0x9053
#define GL_IMAGE_CUBE_MAP_ARRAY 0x9054
#define GL_IMAGE_2D_MULTISAMPLE 0x9055
#define GL_IMAGE_2D_MULTISAMPLE_ARRAY 0x9056
#define GL_INT_IMAGE_1D 0x9057
#define GL_INT_IMAGE_2D 0x9058
#define GL_INT_IMAGE_3D 0x9059
#define GL_INT_IMAGE_2D_RECT 0x905A
#define GL_INT_IMAGE_CUBE 0x905B
#define GL_INT_IMAGE_BUFFER 0x905C
#define GL_INT_IMAGE_1D_ARRAY 0x905D
#define GL_INT_IMAGE_2D_ARRAY 0x905E
#define GL_INT_IMAGE_CUBE_MAP_ARRAY 0x905F
#define GL_INT_IMAGE_2D_MULTISAMPLE 0x9060
```

```
#define GL_INT_IMAGE_2D_MULTISAMPLE_ARRAY 0x9061
#define GL_UNSIGNED_INT_IMAGE_1D 0x9062
#define GL_UNSIGNED_INT_IMAGE_2D 0x9063
#define GL_UNSIGNED_INT_IMAGE_3D 0x9064
#define GL_UNSIGNED_INT_IMAGE_2D_RECT 0x9065
#define GL_UNSIGNED_INT_IMAGE_CUBE 0x9066
#define GL_UNSIGNED_INT_IMAGE_BUFFER 0x9067
#define GL_UNSIGNED_INT_IMAGE_1D_ARRAY 0x9068
#define GL_UNSIGNED_INT_IMAGE_2D_ARRAY 0x9069
#define GL_UNSIGNED_INT_IMAGE_CUBE_MAP_ARRAY 0x906A
#define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE 0x906B
#define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE_ARRAY 0x906C
#define GL_MAX_IMAGE_SAMPLES 0x906D
#define GL_IMAGE_BINDING_FORMAT 0x906E
#define GL_IMAGE_FORMAT_COMPATIBILITY_BY_SIZE 0x90C8
#define GL_IMAGE_FORMAT_COMPATIBILITY_BY_CLASS 0x90C9
#define GL_MAX_VERTEX_IMAGE_UNIFORMS 0x90CA
#define GL_MAX_TESS_CONTROL_IMAGE_UNIFORMS 0x90CB
#define GL_MAX_TESS_EVALUATION_IMAGE_UNIFORMS 0x90CC
#define GL_MAX_GEOMETRY_IMAGE_UNIFORMS 0x90CD
#define GL_MAX_FRAGMENT_IMAGE_UNIFORMS 0x90CE
#define GL_MAX_COMBINED_IMAGE_UNIFORMS 0x90CF
#define GL_PROGRAM_OBJECT_ARB 0x8B40
#define GL_SHADER_OBJECT_ARB 0x8B48
#define GL_OBJECT_TYPE_ARB 0x8B4E
#define GL_OBJECT_SUBTYPE_ARB 0x8B4F
#define GL_FLOAT_VEC2_ARB 0x8B50
#define GL_FLOAT_VEC3_ARB 0x8B51
#define GL_FLOAT_VEC4_ARB 0x8B52
#define GL_INT_VEC2_ARB 0x8B53
#define GL_INT_VEC3_ARB 0x8B54
#define GL_INT_VEC4_ARB 0x8B55
#define GL_BOOL_ARB 0x8B56
#define GL_BOOL_VEC2_ARB 0x8B57
#define GL_BOOL_VEC3_ARB 0x8B58
#define GL_BOOL_VEC4_ARB 0x8B59
#define GL_FLOAT_MAT2_ARB 0x8B5A
#define GL_FLOAT_MAT3_ARB 0x8B5B
#define GL_FLOAT_MAT4_ARB 0x8B5C
#define GL_SAMPLER_1D_ARB 0x8B5D
#define GL_SAMPLER_2D_ARB 0x8B5E
#define GL_SAMPLER_3D_ARB 0x8B5F
#define GL_SAMPLER_CUBE_ARB 0x8B60
#define GL_SAMPLER_1D_SHADOW_ARB 0x8B61
#define GL_SAMPLER_2D_SHADOW_ARB 0x8B62
#define GL_SAMPLER_2D_RECT_ARB 0x8B63
#define GL_SAMPLER_2D_RECT_SHADOW_ARB 0x8B64
#define GL_OBJECT_DELETE_STATUS_ARB 0x8B80
#define GL_OBJECT_COMPILE_STATUS_ARB 0x8B81
#define GL_OBJECT_LINK_STATUS_ARB 0x8B82
#define GL_OBJECT_VALIDATE_STATUS_ARB 0x8B83
#define GL_OBJECT_INFO_LOG_LENGTH_ARB 0x8B84
#define GL_OBJECT_ATTACHED_OBJECTS_ARB 0x8B85
#define GL_OBJECT_ACTIVE_UNIFORMS_ARB 0x8B86
#define GL_OBJECT_ACTIVE_UNIFORM_MAX_LENGTH_ARB 0x8B87
#define GL_OBJECT_SHADER_SOURCE_LENGTH_ARB 0x8B88
#define GL_SHADER_STORAGE_BUFFER 0x90D2
#define GL_SHADER_STORAGE_BUFFER_BINDING 0x90D3
#define GL_SHADER_STORAGE_BUFFER_START 0x90D4
#define GL_SHADER_STORAGE_BUFFER_SIZE 0x90D5
#define GL_MAX_VERTEX_SHADER_STORAGE_BLOCKS 0x90D6
#define GL_MAX_GEOMETRY_SHADER_STORAGE_BLOCKS 0x90D7
#define GL_MAX_TESS_CONTROL_SHADER_STORAGE_BLOCKS 0x90D8
#define GL_MAX_TESS_EVALUATION_SHADER_STORAGE_BLOCKS 0x90D9
#define GL_MAX_FRAGMENT_SHADER_STORAGE_BLOCKS 0x90DA
#define GL_MAX_COMPUTE_SHADER_STORAGE_BLOCKS 0x90DB
#define GL_MAX_COMBINED_SHADER_STORAGE_BLOCKS 0x90DC
#define GL_MAX_SHADER_STORAGE_BUFFER_BINDINGS 0x90DD
#define GL_MAX_SHADER_STORAGE_BLOCK_SIZE 0x90DE
#define GL_SHADER_STORAGE_BUFFER_OFFSET_ALIGNMENT 0x90DF
#define GL_SHADER_STORAGE_BARRIER_BIT 0x00002000
#define GL_MAX_COMBINED_SHADER_OUTPUT_RESOURCES 0x8F39
#define GL_SHADING_LANGUAGE_VERSION_ARB 0x8B8C
#define GL_SHADER_INCLUDE_ARB 0x8DAE
#define GL_NAMED_STRING_LENGTH_ARB 0x8DE9
#define GL_NAMED_STRING_TYPE_ARB 0x8DEA
#define GL_TEXTURE_COMPARE_MODE_ARB 0x884C
#define GL_TEXTURE_COMPARE_FUNC_ARB 0x884D
#define GL_COMPARE_R_TO_TEXTURE_ARB 0x884E
#define GL_TEXTURE_COMPARE_FAIL_VALUE_ARB 0x80BF
#define GL_SPARSE_STORAGE_BIT_ARB 0x0400
#define GL_SPARSE_BUFFER_PAGE_SIZE_ARB 0x82F8
#define GL_TEXTURE_SPARSE_ARB 0x91A6
#define GL_VIRTUAL_PAGE_SIZE_INDEX_ARB 0x91A7
#define GL_NUM_SPARSE_LEVELS_ARB 0x91AA
#define GL_NUM_VIRTUAL_PAGE_SIZES_ARB 0x91A8
#define GL_VIRTUAL_PAGE_SIZE_X_ARB 0x9195
```

```
#define GL_VIRTUAL_PAGE_SIZE_Y_ARB 0x9196
#define GL_VIRTUAL_PAGE_SIZE_Z_ARB 0x9197
#define GL_MAX_SPARSE_TEXTURE_SIZE_ARB 0x9198
#define GL_MAX_SPARSE_3D_TEXTURE_SIZE_ARB 0x9199
#define GL_MAX_SPARSE_ARRAY_TEXTURE_LAYERS_ARB 0x919A
#define GL_SPARSE_TEXTURE_FULL_ARRAY_CUBE_MIPMAPS_ARB 0x91A9
#define GL_SPIR_V_EXTENSIONS 0x9553
#define GL_NUM_SPIR_V_EXTENSIONS 0x9554
#define GL_DEPTH_STENCIL_TEXTURE_MODE 0x90EA
#define GL_CLAMP_TO_BORDER_ARB 0x812D
#define GL_TEXTURE_BUFFER_ARB 0x8C2A
#define GL_MAX_TEXTURE_BUFFER_SIZE_ARB 0x8C2B
#define GL_TEXTURE_BINDING_BUFFER_ARB 0x8C2C
#define GL_TEXTURE_BUFFER_DATA_STORE_BINDING_ARB 0x8C2D
#define GL_TEXTURE_BUFFER_FORMAT_ARB 0x8C2E
#define GL_TEXTURE_BUFFER_OFFSET 0x919D
#define GL_TEXTURE_BUFFER_SIZE 0x919E
#define GL_TEXTURE_BUFFER_OFFSET_ALIGNMENT 0x919F
#define GL_COMPRESSED_ALPHA_ARB 0x84E9
#define GL_COMPRESSED_LUMINANCE_ARB 0x84EA
#define GL_COMPRESSED_LUMINANCE_ALPHA_ARB 0x84EB
#define GL_COMPRESSED_INTENSITY_ARB 0x84EC
#define GL_COMPRESSED_RGB_ARB 0x84ED
#define GL_COMPRESSED_RGBA_ARB 0x84EE
#define GL_TEXTURE_COMPRESSION_HINT_ARB 0x84EF
#define GL_TEXTURE_COMPRESSED_IMAGE_SIZE_ARB 0x86A0
#define GL_TEXTURE_COMPRESSED_ARB 0x86A1
#define GL_NUM_COMPRESSED_TEXTURE_FORMATS_ARB 0x86A2
#define GL_COMPRESSED_TEXTURE_FORMATS_ARB 0x86A3
#define GL_COMPRESSED_RGBA_BPTC_UNORM_ARB 0x8E8C
#define GL_COMPRESSED_SRGB_ALPHA_BPTC_UNORM_ARB 0x8E8D
#define GL_COMPRESSED_RGB_BPTC_SIGNED_FLOAT_ARB 0x8E8E
#define GL_COMPRESSED_RGB_BPTC_UNSIGNED_FLOAT_ARB 0x8E8F
#define GL_NORMAL_MAP_ARB 0x8511
#define GL_REFLECTION_MAP_ARB 0x8512
#define GL_TEXTURE_CUBE_MAP_ARB 0x8513
#define GL_TEXTURE_BINDING_CUBE_MAP_ARB 0x8514
#define GL_TEXTURE_CUBE_MAP_POSITIVE_X_ARB 0x8515
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_X_ARB 0x8516
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Y_ARB 0x8517
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y_ARB 0x8518
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Z_ARB 0x8519
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z_ARB 0x851A
#define GL_PROXY_TEXTURE_CUBE_MAP_ARB 0x851B
#define GL_MAX_CUBE_MAP_TEXTURE_SIZE_ARB 0x851C
#define GL_TEXTURE_CUBE_MAP_ARRAY_ARB 0x9009
#define GL_TEXTURE_BINDING_CUBE_MAP_ARRAY_ARB 0x900A
#define GL_PROXY_TEXTURE_CUBE_MAP_ARRAY_ARB 0x900B
#define GL_SAMPLER_CUBE_MAP_ARRAY_ARB 0x900C
#define GL_SAMPLER_CUBE_MAP_ARRAY_SHADOW_ARB 0x900D
#define GL_INT_SAMPLER_CUBE_MAP_ARRAY_ARB 0x900E
#define GL_UNSIGNED_INT_SAMPLER_CUBE_MAP_ARRAY_ARB 0x900F
#define GL_COMBINE_ARB 0x8570
#define GL_COMBINE_RGB_ARB 0x8571
#define GL_COMBINE_ALPHA_ARB 0x8572
#define GL_SOURCE0_RGB_ARB 0x8580
#define GL_SOURCE1_RGB_ARB 0x8581
#define GL_SOURCE2_RGB_ARB 0x8582
#define GL_SOURCE0_ALPHA_ARB 0x8588
#define GL_SOURCE1_ALPHA_ARB 0x8589
#define GL_SOURCE2_ALPHA_ARB 0x858A
#define GL_OPERAND0_RGB_ARB 0x8590
#define GL_OPERAND1_RGB_ARB 0x8591
#define GL_OPERAND2_RGB_ARB 0x8592
#define GL_OPERAND0_ALPHA_ARB 0x8598
#define GL_OPERAND1_ALPHA_ARB 0x8599
#define GL_OPERAND2_ALPHA_ARB 0x859A
#define GL_RGB_SCALE_ARB 0x8573
#define GL_ADD_SIGNED_ARB 0x8574
#define GL_INTERPOLATE_ARB 0x8575
#define GL_SUBTRACT_ARB 0x84E7
#define GL_CONSTANT_ARB 0x8576
#define GL_PRIMARY_COLOR_ARB 0x8577
#define GL_PREVIOUS_ARB 0x8578
#define GL_DOT3_RGB_ARB 0x86AE
#define GL_DOT3_RGBA_ARB 0x86AF
#define GL_TEXTURE_MAX_ANISOTROPY 0x84FE
#define GL_MAX_TEXTURE_MAX_ANISOTROPY 0x84FF
#define GL_TEXTURE_REDUCTION_MODE_ARB 0x9366
#define GL_WEIGHTED_AVERAGE_ARB 0x9367
#define GL_TEXTURE_RED_TYPE_ARB 0x8C10
#define GL_TEXTURE_GREEN_TYPE_ARB 0x8C11
#define GL_TEXTURE_BLUE_TYPE_ARB 0x8C12
#define GL_TEXTURE_ALPHA_TYPE_ARB 0x8C13
#define GL_TEXTURE_LUMINANCE_TYPE_ARB 0x8C14
#define GL_TEXTURE_INTENSITY_TYPE_ARB 0x8C15
#define GL_TEXTURE_DEPTH_TYPE_ARB 0x8C16
```

```
#define GL_UNSIGNED_NORMALIZED_ARB 0x8C17
#define GL_RGBA32F_ARB 0x8814
#define GL_RGB32F_ARB 0x8815
#define GL_ALPHA32F_ARB 0x8816
#define GL_INTENSITY32F_ARB 0x8817
#define GL_LUMINANCE32F_ARB 0x8818
#define GL_LUMINANCE_ALPHA32F_ARB 0x8819
#define GL_RGBA16F_ARB 0x881A
#define GL_RGB16F_ARB 0x881B
#define GL_ALPHA16F_ARB 0x881C
#define GL_INTENSITY16F_ARB 0x881D
#define GL_LUMINANCE16F_ARB 0x881E
#define GL_LUMINANCE_ALPHA16F_ARB 0x881F
#define GL_MIN_PROGRAM_TEXTURE_GATHER_OFFSET_ARB 0x8E5E
#define GL_MAX_PROGRAM_TEXTURE_GATHER_OFFSET_ARB 0x8E5F
#define GL_MAX_PROGRAM_TEXTURE_GATHER_COMPONENTS_ARB 0x8F9F
#define GL_MIRROR_CLAMP_TO_EDGE 0x8743
#define GL_MIRRORED_REPEAT_ARB 0x8370
#define GL_TEXTURE_RECTANGLE_ARB 0x84F5
#define GL_TEXTURE_BINDING_RECTANGLE_ARB 0x84F6
#define GL_PROXY_TEXTURE_RECTANGLE_ARB 0x84F7
#define GL_MAX_RECTANGLE_TEXTURE_SIZE_ARB 0x84F8
#define GL_TEXTURE_IMMUTABLE_FORMAT 0x912F
#define GL_TEXTURE_VIEW_MIN_LEVEL 0x82DB
#define GL_TEXTURE_VIEW_NUM_LEVELS 0x82DC
#define GL_TEXTURE_VIEW_MIN_LAYER 0x82DD
#define GL_TEXTURE_VIEW_NUM_LAYERS 0x82DE
#define GL_TEXTURE_IMMUTABLE_LEVELS 0x82DF
#define GL_TRANSFORM_FEEDBACK_OVERFLOW_ARB 0x82EC
#define GL_TRANSFORM_FEEDBACK_STREAM_OVERFLOW_ARB 0x82ED
#define GL_TRANSPOSE_MODELVIEW_MATRIX_ARB 0x84E3
#define GL_TRANSPOSE_PROJECTION_MATRIX_ARB 0x84E4
#define GL_TRANSPOSE_TEXTURE_MATRIX_ARB 0x84E5
#define GL_TRANSPOSE_COLOR_MATRIX_ARB 0x84E6
#define GL_VERTEX_ATTRIB_BINDING 0x82D4
#define GL_VERTEX_ATTRIB_RELATIVE_OFFSET 0x82D5
#define GL_VERTEX_BINDING_DIVISOR 0x82D6
#define GL_VERTEX_BINDING_OFFSET 0x82D7
#define GL_VERTEX_BINDING_STRIDE 0x82D8
#define GL_MAX_VERTEX_ATTRIB_RELATIVE_OFFSET 0x82D9
#define GL_MAX_VERTEX_ATTRIB_BINDINGS 0x82DA
#define GL_MAX_VERTEX_UNITS_ARB 0x86A4
#define GL_ACTIVE_VERTEX_UNITS_ARB 0x86A5
#define GL_WEIGHT_SUM_UNITY_ARB 0x86A6
#define GL_VERTEX_BLEND_ARB 0x86A7
#define GL_CURRENT_WEIGHT_ARB 0x86A8
#define GL_WEIGHT_ARRAY_TYPE_ARB 0x86A9
#define GL_WEIGHT_ARRAY_STRIDE_ARB 0x86AA
#define GL_WEIGHT_ARRAY_SIZE_ARB 0x86AB
#define GL_WEIGHT_ARRAY_POINTER_ARB 0x86AC
#define GL_WEIGHT_ARRAY_ARB 0x86AD
#define GL_MODELVIEW0_ARB 0x1700
#define GL_MODELVIEW1_ARB 0x850A
#define GL_MODELVIEW2_ARB 0x8722
#define GL_MODELVIEW3_ARB 0x8723
#define GL_MODELVIEW4_ARB 0x8724
#define GL_MODELVIEW5_ARB 0x8725
#define GL_MODELVIEW6_ARB 0x8726
#define GL_MODELVIEW7_ARB 0x8727
#define GL_MODELVIEW8_ARB 0x8728
#define GL_MODELVIEW9_ARB 0x8729
#define GL_MODELVIEW10_ARB 0x872A
#define GL_MODELVIEW11_ARB 0x872B
#define GL_MODELVIEW12_ARB 0x872C
#define GL_MODELVIEW13_ARB 0x872D
#define GL_MODELVIEW14_ARB 0x872E
#define GL_MODELVIEW15_ARB 0x872F
#define GL_MODELVIEW16_ARB 0x8730
#define GL_MODELVIEW17_ARB 0x8731
#define GL_MODELVIEW18_ARB 0x8732
#define GL_MODELVIEW19_ARB 0x8733
#define GL_MODELVIEW20_ARB 0x8734
#define GL_MODELVIEW21_ARB 0x8735
#define GL_MODELVIEW22_ARB 0x8736
#define GL_MODELVIEW23_ARB 0x8737
#define GL_MODELVIEW24_ARB 0x8738
#define GL_MODELVIEW25_ARB 0x8739
#define GL_MODELVIEW26_ARB 0x873A
#define GL_MODELVIEW27_ARB 0x873B
#define GL_MODELVIEW28_ARB 0x873C
#define GL_MODELVIEW29_ARB 0x873D
#define GL_MODELVIEW30_ARB 0x873E
#define GL_MODELVIEW31_ARB 0x873F
#define GL_BUFFER_SIZE_ARB 0x8764
#define GL_BUFFER_USAGE_ARB 0x8765
#define GL_ARRAY_BUFFER_ARB 0x8892
#define GL_ELEMENT_ARRAY_BUFFER_ARB 0x8893
```

```
#define GL_ARRAY_BUFFER_BINDING_ARB 0x8894
#define GL_ELEMENT_ARRAY_BUFFER_BINDING_ARB 0x8895
#define GL_VERTEX_ARRAY_BUFFER_BINDING_ARB 0x8896
#define GL_NORMAL_ARRAY_BUFFER_BINDING_ARB 0x8897
#define GL_COLOR_ARRAY_BUFFER_BINDING_ARB 0x8898
#define GL_INDEX_ARRAY_BUFFER_BINDING_ARB 0x8899
#define GL_TEXTURE_COORD_ARRAY_BUFFER_BINDING_ARB 0x889A
#define GL_EDGE_FLAG_ARRAY_BUFFER_BINDING_ARB 0x889B
#define GL_SECONDARY_COLOR_ARRAY_BUFFER_BINDING_ARB 0x889C
#define GL_FOG_COORDINATE_ARRAY_BUFFER_BINDING_ARB 0x889D
#define GL_WEIGHT_ARRAY_BUFFER_BINDING_ARB 0x889E
#define GL_VERTEX_ATTRIB_ARRAY_BUFFER_BINDING_ARB 0x889F
#define GL_READ_ONLY_ARB 0x88B8
#define GL_WRITE_ONLY_ARB 0x88B9
#define GL_READ_WRITE_ARB 0x88BA
#define GL_BUFFER_ACCESS_ARB 0x88BB
#define GL_BUFFER_MAPPED_ARB 0x88BC
#define GL_BUFFER_MAP_POINTER_ARB 0x88BD
#define GL_STREAM_DRAW_ARB 0x88E0
#define GL_STREAM_READ_ARB 0x88E1
#define GL_STREAM_COPY_ARB 0x88E2
#define GL_STATIC_DRAW_ARB 0x88E4
#define GL_STATIC_READ_ARB 0x88E5
#define GL_STATIC_COPY_ARB 0x88E6
#define GL_DYNAMIC_DRAW_ARB 0x88E8
#define GL_DYNAMIC_READ_ARB 0x88E9
#define GL_DYNAMIC_COPY_ARB 0x88EA
#define GL_COLOR_SUM_ARB 0x8458
#define GL_VERTEX_PROGRAM_ARB 0x8620
#define GL_VERTEX_ATTRIB_ARRAY_ENABLED_ARB 0x8622
#define GL_VERTEX_ATTRIB_ARRAY_SIZE_ARB 0x8623
#define GL_VERTEX_ATTRIB_ARRAY_STRIDE_ARB 0x8624
#define GL_VERTEX_ATTRIB_ARRAY_TYPE_ARB 0x8625
#define GL_CURRENT_VERTEX_ATTRIB_ARB 0x8626
#define GL_VERTEX_PROGRAM_POINT_SIZE_ARB 0x8642
#define GL_VERTEX_PROGRAM_TWO_SIDE_ARB 0x8643
#define GL_VERTEX_ATTRIB_ARRAY_POINTER_ARB 0x8645
#define GL_MAX_VERTEX_ATTRIBS_ARB 0x8869
#define GL_VERTEX_ATTRIB_ARRAY_NORMALIZED_ARB 0x886A
#define GL_PROGRAM_ADDRESS_REGISTERS_ARB 0x88B0
#define GL_MAX_PROGRAM_ADDRESS_REGISTERS_ARB 0x88B1
#define GL_PROGRAM_NATIVE_ADDRESS_REGISTERS_ARB 0x88B2
#define GL_MAX_PROGRAM_NATIVE_ADDRESS_REGISTERS_ARB 0x88B3
#define GL_VERTEX_SHADER_ARB 0x8B31
#define GL_MAX_VERTEX_UNIFORM_COMPONENTS_ARB 0x8B4A
#define GL_MAX_VARYING_FLOATS_ARB 0x8B4B
#define GL_MAX_VERTEX_TEXTURE_IMAGE_UNITS_ARB 0x8B4C
#define GL_MAX_COMBINED_TEXTURE_IMAGE_UNITS_ARB 0x8B4D
#define GL_OBJECT_ACTIVE_ATTRIBUTES_ARB 0x8B89
#define GL_OBJECT_ACTIVE_ATTRIBUTE_MAX_LENGTH_ARB 0x8B8A
#define GL_MAX_DRAW_BUFFERS_ATI 0x8824
#define GL_DRAW_BUFFER0_ATI 0x8825
#define GL_DRAW_BUFFER1_ATI 0x8826
#define GL_DRAW_BUFFER2_ATI 0x8827
#define GL_DRAW_BUFFER3_ATI 0x8828
#define GL_DRAW_BUFFER4_ATI 0x8829
#define GL_DRAW_BUFFER5_ATI 0x882A
#define GL_DRAW_BUFFER6_ATI 0x882B
#define GL_DRAW_BUFFER7_ATI 0x882C
#define GL_DRAW_BUFFER8_ATI 0x882D
#define GL_DRAW_BUFFER9_ATI 0x882E
#define GL_DRAW_BUFFER10_ATI 0x882F
#define GL_DRAW_BUFFER11_ATI 0x8830
#define GL_DRAW_BUFFER12_ATI 0x8831
#define GL_DRAW_BUFFER13_ATI 0x8832
#define GL_DRAW_BUFFER14_ATI 0x8833
#define GL_DRAW_BUFFER15_ATI 0x8834
#define GL_ELEMENT_ARRAY_ATI 0x8768
#define GL_ELEMENT_ARRAY_TYPE_ATI 0x8769
#define GL_ELEMENT_ARRAY_POINTER_ATI 0x876A
#define GL_BUMP_ROT_MATRIX_ATI 0x8775
#define GL_BUMP_ROT_MATRIX_SIZE_ATI 0x8776
#define GL_BUMP_NUM_TEX_UNITS_ATI 0x8777
#define GL_BUMP_TEX_UNITS_ATI 0x8778
#define GL_DUDV_ATI 0x8779
#define GL_DU8DV8_ATI 0x877A
#define GL_BUMP_ENVMAP_ATI 0x877B
#define GL_BUMP_TARGET_ATI 0x877C
#define GL_FRAGMENT_SHADER_ATI 0x8920
#define GL_REG_0_ATI 0x8921
#define GL_REG_1_ATI 0x8922
#define GL_REG_2_ATI 0x8923
#define GL_REG_3_ATI 0x8924
#define GL_REG_4_ATI 0x8925
#define GL_REG_5_ATI 0x8926
#define GL_REG_6_ATI 0x8927
#define GL_REG_7_ATI 0x8928
```

```
#define GL_REG_8_ATI 0x8929
#define GL_REG_9_ATI 0x892A
#define GL_REG_10_ATI 0x892B
#define GL_REG_11_ATI 0x892C
#define GL_REG_12_ATI 0x892D
#define GL_REG_13_ATI 0x892E
#define GL_REG_14_ATI 0x892F
#define GL_REG_15_ATI 0x8930
#define GL_REG_16_ATI 0x8931
#define GL_REG_17_ATI 0x8932
#define GL_REG_18_ATI 0x8933
#define GL_REG_19_ATI 0x8934
#define GL_REG_20_ATI 0x8935
#define GL_REG_21_ATI 0x8936
#define GL_REG_22_ATI 0x8937
#define GL_REG_23_ATI 0x8938
#define GL_REG_24_ATI 0x8939
#define GL_REG_25_ATI 0x893A
#define GL_REG_26_ATI 0x893B
#define GL_REG_27_ATI 0x893C
#define GL_REG_28_ATI 0x893D
#define GL_REG_29_ATI 0x893E
#define GL_REG_30_ATI 0x893F
#define GL_REG_31_ATI 0x8940
#define GL_CON_0_ATI 0x8941
#define GL_CON_1_ATI 0x8942
#define GL_CON_2_ATI 0x8943
#define GL_CON_3_ATI 0x8944
#define GL_CON_4_ATI 0x8945
#define GL_CON_5_ATI 0x8946
#define GL_CON_6_ATI 0x8947
#define GL_CON_7_ATI 0x8948
#define GL_CON_8_ATI 0x8949
#define GL_CON_9_ATI 0x894A
#define GL_CON_10_ATI 0x894B
#define GL_CON_11_ATI 0x894C
#define GL_CON_12_ATI 0x894D
#define GL_CON_13_ATI 0x894E
#define GL_CON_14_ATI 0x894F
#define GL_CON_15_ATI 0x8950
#define GL_CON_16_ATI 0x8951
#define GL_CON_17_ATI 0x8952
#define GL_CON_18_ATI 0x8953
#define GL_CON_19_ATI 0x8954
#define GL_CON_20_ATI 0x8955
#define GL_CON_21_ATI 0x8956
#define GL_CON_22_ATI 0x8957
#define GL_CON_23_ATI 0x8958
#define GL_CON_24_ATI 0x8959
#define GL_CON_25_ATI 0x895A
#define GL_CON_26_ATI 0x895B
#define GL_CON_27_ATI 0x895C
#define GL_CON_28_ATI 0x895D
#define GL_CON_29_ATI 0x895E
#define GL_CON_30_ATI 0x895F
#define GL_CON_31_ATI 0x8960
#define GL_MOV_ATI 0x8961
#define GL_ADD_ATI 0x8963
#define GL_MUL_ATI 0x8964
#define GL_SUB_ATI 0x8965
#define GL_DOT3_ATI 0x8966
#define GL_DOT4_ATI 0x8967
#define GL_MAD_ATI 0x8968
#define GL_LERP_ATI 0x8969
#define GL_CND_ATI 0x896A
#define GL_CND0_ATI 0x896B
#define GL_DOT2_ADD_ATI 0x896C
#define GL_SECONDARY_INTERPOLATOR_ATI 0x896D
#define GL_NUM_FRAGMENT_REGISTERS_ATI 0x896E
#define GL_NUM_FRAGMENT_CONSTANTS_ATI 0x896F
#define GL_NUM_PASSES_ATI 0x8970
#define GL_NUM_INSTRUCTIONS_PER_PASS_ATI 0x8971
#define GL_NUM_INSTRUCTIONS_TOTAL_ATI 0x8972
#define GL_NUM_INPUT_INTERPOLATOR_COMPONENTS_ATI 0x8973
#define GL_NUM_LOOPBACK_COMPONENTS_ATI 0x8974
#define GL_COLOR_ALPHA_PAIRING_ATI 0x8975
#define GL_SWIZZLE_STR_ATI 0x8976
#define GL_SWIZZLE_STQ_ATI 0x8977
#define GL_SWIZZLE_STR_DR_ATI 0x8978
#define GL_SWIZZLE_STQ_DQ_ATI 0x8979
#define GL_SWIZZLE_STRQ_ATI 0x897A
#define GL_SWIZZLE_STRQ_DQ_ATI 0x897B
#define GL_RED_BIT_ATI 0x00000001
#define GL_GREEN_BIT_ATI 0x00000002
#define GL_BLUE_BIT_ATI 0x00000004
#define GL_2X_BIT_ATI 0x00000001
#define GL_4X_BIT_ATI 0x00000002
```

```
#define GL_8X_BIT_ATI 0x00000004
#define GL_HALF_BIT_ATI 0x00000008
#define GL_QUARTER_BIT_ATI 0x00000010
#define GL_EIGHTH_BIT_ATI 0x00000020
#define GL_SATURATE_BIT_ATI 0x00000040
#define GL_COMP_BIT_ATI 0x00000002
#define GL_NEGATE_BIT_ATI 0x00000004
#define GL_BIAS_BIT_ATI 0x00000008
#define GL_VBO_FREE_MEMORY_ATI 0x87FB
#define GL_TEXTURE_FREE_MEMORY_ATI 0x87FC
#define GL_RENDERBUFFER_FREE_MEMORY_ATI 0x87FD
#define GL_RGBA_FLOAT_MODE_ATI 0x8820
#define GL_COLOR_CLEAR_UNCLAMPED_VALUE_ATI 0x8835
#define GL_PN_TRIANGLES_ATI 0x87F0
#define GL_MAX_PN_TRIANGLES_TESSELATION_LEVEL_ATI 0x87F1
#define GL_PN_TRIANGLES_POINT_MODE_ATI 0x87F2
#define GL_PN_TRIANGLES_NORMAL_MODE_ATI 0x87F3
#define GL_PN_TRIANGLES_TESSELATION_LEVEL_ATI 0x87F4
#define GL_PN_TRIANGLES_POINT_MODE_LINEAR_ATI 0x87F5
#define GL_PN_TRIANGLES_POINT_MODE_CUBIC_ATI 0x87F6
#define GL_PN_TRIANGLES_NORMAL_MODE_LINEAR_ATI 0x87F7
#define GL_PN_TRIANGLES_NORMAL_MODE_QUADRATIC_ATI 0x87F8
#define GL_STENCIL_BACK_FUNC_ATI 0x8800
#define GL_STENCIL_BACK_FAIL_ATI 0x8801
#define GL_STENCIL_BACK_PASS_DEPTH_FAIL_ATI 0x8802
#define GL_STENCIL_BACK_PASS_DEPTH_PASS_ATI 0x8803
#define GL_TEXT_FRAGMENT_SHADER_ATI 0x8200
#define GL_MODULATE_ADD_ATI 0x8744
#define GL_MODULATE_SIGNED_ADD_ATI 0x8745
#define GL_MODULATE_SUBTRACT_ATI 0x8746
#define GL_RGBA_FLOAT32_ATI 0x8814
#define GL_RGB_FLOAT32_ATI 0x8815
#define GL_ALPHA_FLOAT32_ATI 0x8816
#define GL_INTENSITY_FLOAT32_ATI 0x8817
#define GL_LUMINANCE_FLOAT32_ATI 0x8818
#define GL_LUMINANCE_ALPHA_FLOAT32_ATI 0x8819
#define GL_RGBA_FLOAT16_ATI 0x881A
#define GL_RGB_FLOAT16_ATI 0x881B
#define GL_ALPHA_FLOAT16_ATI 0x881C
#define GL_INTENSITY_FLOAT16_ATI 0x881D
#define GL_LUMINANCE_FLOAT16_ATI 0x881E
#define GL_LUMINANCE_ALPHA_FLOAT16_ATI 0x881F
#define GL_MIRROR_CLAMP_ATI 0x8742
#define GL_MIRROR_CLAMP_TO_EDGE_ATI 0x8743
#define GL_STATIC_ATI 0x8760
#define GL_DYNAMIC_ATI 0x8761
#define GL_PRESERVE_ATI 0x8762
#define GL_DISCARD_ATI 0x8763
#define GL_OBJECT_BUFFER_SIZE_ATI 0x8764
#define GL_OBJECT_BUFFER_USAGE_ATI 0x8765
#define GL_ARRAY_OBJECT_BUFFER_ATI 0x8766
#define GL_ARRAY_OBJECT_OFFSET_ATI 0x8767
#define GL_MAX_VERTEX_STREAMS_ATI 0x876B
#define GL_VERTEX_STREAM0_ATI 0x876C
#define GL_VERTEX_STREAM1_ATI 0x876D
#define GL_VERTEX_STREAM2_ATI 0x876E
#define GL_VERTEX_STREAM3_ATI 0x876F
#define GL_VERTEX_STREAM4_ATI 0x8770
#define GL_VERTEX_STREAM5_ATI 0x8771
#define GL_VERTEX_STREAM6_ATI 0x8772
#define GL_VERTEX_STREAM7_ATI 0x8773
#define GL_VERTEX_SOURCE_ATI 0x8774
#define GL_422_EXT 0x80CC
#define GL_422_REV_EXT 0x80CD
#define GL_422_AVERAGE_EXT 0x80CE
#define GL_422_REV_AVERAGE_EXT 0x80CF
#define GL_ABGR_EXT 0x8000
#define GL_BGR_EXT 0x80E0
#define GL_BGRA_EXT 0x80E1
#define GL_MAX_VERTEX_BINDABLE_UNIFORMS_EXT 0x8DE2
#define GL_MAX_FRAGMENT_BINDABLE_UNIFORMS_EXT 0x8DE3
#define GL_MAX_GEOMETRY_BINDABLE_UNIFORMS_EXT 0x8DE4
#define GL_MAX_BINDABLE_UNIFORM_SIZE_EXT 0x8DED
#define GL_UNIFORM_BUFFER_EXT 0x8DEE
#define GL_UNIFORM_BUFFER_BINDING_EXT 0x8DEF
#define GL_CONSTANT_COLOR_EXT 0x8001
#define GL_ONE_MINUS_CONSTANT_COLOR_EXT 0x8002
#define GL_CONSTANT_ALPHA_EXT 0x8003
#define GL_ONE_MINUS_CONSTANT_ALPHA_EXT 0x8004
#define GL_BLEND_COLOR_EXT 0x8005
#define GL_BLEND_EQUATION_RGB_EXT 0x8009
#define GL_BLEND_EQUATION_ALPHA_EXT 0x883D
#define GL_BLEND_DST_RGB_EXT 0x80C8
#define GL_BLEND_SRC_RGB_EXT 0x80C9
#define GL_BLEND_DST_ALPHA_EXT 0x80CA
#define GL_BLEND_SRC_ALPHA_EXT 0x80CB
#define GL_MIN_EXT 0x8007
```

```
#define GL_MAX_EXT 0x8008
#define GL_FUNC_ADD_EXT 0x8006
#define GL_BLEND_EQUATION_EXT 0x8009
#define GL_FUNC_SUBTRACT_EXT 0x800A
#define GL_FUNC_REVERSE_SUBTRACT_EXT 0x800B
#define GL_CLIP_VOLUME_CLIPPING_HINT_EXT 0x80F0
#define GL_CMYK_EXT 0x800C
#define GL_CMYKA_EXT 0x800D
#define GL_PACK_CMYK_HINT_EXT 0x800E
#define GL_UNPACK_CMYK_HINT_EXT 0x800F
#define GL_ARRAY_ELEMENT_LOCK_FIRST_EXT 0x81A8
#define GL_ARRAY_ELEMENT_LOCK_COUNT_EXT 0x81A9
#define GL_CONVOLUTION_1D_EXT 0x8010
#define GL_CONVOLUTION_2D_EXT 0x8011
#define GL_SEPARABLE_2D_EXT 0x8012
#define GL_CONVOLUTION_BORDER_MODE_EXT 0x8013
#define GL_CONVOLUTION_FILTER_SCALE_EXT 0x8014
#define GL_CONVOLUTION_FILTER_BIAS_EXT 0x8015
#define GL_REDUCE_EXT 0x8016
#define GL_CONVOLUTION_FORMAT_EXT 0x8017
#define GL_CONVOLUTION_WIDTH_EXT 0x8018
#define GL_CONVOLUTION_HEIGHT_EXT 0x8019
#define GL_MAX_CONVOLUTION_WIDTH_EXT 0x801A
#define GL_MAX_CONVOLUTION_HEIGHT_EXT 0x801B
#define GL_POST_CONVOLUTION_RED_SCALE_EXT 0x801C
#define GL_POST_CONVOLUTION_GREEN_SCALE_EXT 0x801D
#define GL_POST_CONVOLUTION_BLUE_SCALE_EXT 0x801E
#define GL_POST_CONVOLUTION_ALPHA_SCALE_EXT 0x801F
#define GL_POST_CONVOLUTION_RED_BIAS_EXT 0x8020
#define GL_POST_CONVOLUTION_GREEN_BIAS_EXT 0x8021
#define GL_POST_CONVOLUTION_BLUE_BIAS_EXT 0x8022
#define GL_POST_CONVOLUTION_ALPHA_BIAS_EXT 0x8023
#define GL_TANGENT_ARRAY_EXT 0x8439
#define GL_BINORMAL_ARRAY_EXT 0x843A
#define GL_CURRENT_TANGENT_EXT 0x843B
#define GL_CURRENT_BINORMAL_EXT 0x843C
#define GL_TANGENT_ARRAY_TYPE_EXT 0x843E
#define GL_TANGENT_ARRAY_STRIDE_EXT 0x843F
#define GL_BINORMAL_ARRAY_TYPE_EXT 0x8440
#define GL_BINORMAL_ARRAY_STRIDE_EXT 0x8441
#define GL_TANGENT_ARRAY_POINTER_EXT 0x8442
#define GL_BINORMAL_ARRAY_POINTER_EXT 0x8443
#define GL_MAP1_TANGENT_EXT 0x8444
#define GL_MAP2_TANGENT_EXT 0x8445
#define GL_MAP1_BINORMAL_EXT 0x8446
#define GL_MAP2_BINORMAL_EXT 0x8447
#define GL_CULL_VERTEX_EXT 0x81AA
#define GL_CULL_VERTEX_EYE_POSITION_EXT 0x81AB
#define GL_CULL_VERTEX_OBJECT_POSITION_EXT 0x81AC
#define GL_PROGRAM_PIPELINE_OBJECT_EXT 0x8A4F
#define GL_PROGRAM_OBJECT_EXT 0x8B40
#define GL_SHADER_OBJECT_EXT 0x8B48
#define GL_BUFFER_OBJECT_EXT 0x9151
#define GL_QUERY_OBJECT_EXT 0x9153
#define GL_VERTEX_ARRAY_OBJECT_EXT 0x9154
#define GL_SAMPLER 0x82E6
#define GL_DEPTH_BOUNDS_TEST_EXT 0x8890
#define GL_DEPTH_BOUNDS_EXT 0x8891
#define GL_PROGRAM_MATRIX_EXT 0x8E2D
#define GL_TRANSPOSE_PROGRAM_MATRIX_EXT 0x8E2E
#define GL_PROGRAM_MATRIX_STACK_DEPTH_EXT 0x8E2F
#define GL_MAX_ELEMENTS_VERTICES_EXT 0x80E8
#define GL_MAX_ELEMENTS_INDICES_EXT 0x80E9
#define GL_FOG_COORDINATE_SOURCE_EXT 0x8450
#define GL_FOG_COORDINATE_EXT 0x8451
#define GL_FRAGMENT_DEPTH_EXT 0x8452
#define GL_CURRENT_FOG_COORDINATE_EXT 0x8453
#define GL_FOG_COORDINATE_ARRAY_TYPE_EXT 0x8454
#define GL_FOG_COORDINATE_ARRAY_STRIDE_EXT 0x8455
#define GL_FOG_COORDINATE_ARRAY_POINTER_EXT 0x8456
#define GL_FOG_COORDINATE_ARRAY_EXT 0x8457
#define GL_READ_FRAMEBUFFER_EXT 0x8CA8
#define GL_DRAW_FRAMEBUFFER_EXT 0x8CA9
#define GL_DRAW_FRAMEBUFFER_BINDING_EXT 0x8CA6
#define GL_READ_FRAMEBUFFER_BINDING_EXT 0x8CAA
#define GL_RENDERBUFFER_SAMPLES_EXT 0x8CAB
#define GL_FRAMEBUFFER_INCOMPLETE_MULTISAMPLE_EXT 0x8D56
#define GL_MAX_SAMPLES_EXT 0x8D57
#define GL_SCALED_RESOLVE_FASTEST_EXT 0x90BA
#define GL_SCALED_RESOLVE_NICEST_EXT 0x90BB
#define GL_INVALID_FRAMEBUFFER_OPERATION_EXT 0x0506
#define GL_MAX_RENDERBUFFER_SIZE_EXT 0x84E8
#define GL_FRAMEBUFFER_BINDING_EXT 0x8CA6
#define GL_RENDERBUFFER_BINDING_EXT 0x8CA7
#define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_TYPE_EXT 0x8CD0
#define GL_FRAMEBUFFER_ATTACHMENT_OBJECT_NAME_EXT 0x8CD1
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LEVEL_EXT 0x8CD2
```

```
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_CUBE_MAP_FACE_EXT 0x8CD3
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_3D_ZOFFSET_EXT 0x8CD4
#define GL_FRAMEBUFFER_COMPLETE_EXT 0x8CD5
#define GL_FRAMEBUFFER_INCOMPLETE_ATTACHMENT_EXT 0x8CD6
#define GL_FRAMEBUFFER_INCOMPLETE_MISSING_ATTACHMENT_EXT 0x8CD7
#define GL_FRAMEBUFFER_INCOMPLETE_DIMENSIONS_EXT 0x8CD9
#define GL_FRAMEBUFFER_INCOMPLETE_FORMATS_EXT 0x8CDA
#define GL_FRAMEBUFFER_INCOMPLETE_DRAW_BUFFER_EXT 0x8CDB
#define GL_FRAMEBUFFER_INCOMPLETE_READ_BUFFER_EXT 0x8CDC
#define GL_FRAMEBUFFER_UNSUPPORTED_EXT 0x8CDD
#define GL_MAX_COLOR_ATTACHMENTS_EXT 0x8CDF
#define GL_COLOR_ATTACHMENT0_EXT 0x8CE0
#define GL_COLOR_ATTACHMENT1_EXT 0x8CE1
#define GL_COLOR_ATTACHMENT2_EXT 0x8CE2
#define GL_COLOR_ATTACHMENT3_EXT 0x8CE3
#define GL_COLOR_ATTACHMENT4_EXT 0x8CE4
#define GL_COLOR_ATTACHMENT5_EXT 0x8CE5
#define GL_COLOR_ATTACHMENT6_EXT 0x8CE6
#define GL_COLOR_ATTACHMENT7_EXT 0x8CE7
#define GL_COLOR_ATTACHMENT8_EXT 0x8CE8
#define GL_COLOR_ATTACHMENT9_EXT 0x8CE9
#define GL_COLOR_ATTACHMENT10_EXT 0x8CEA
#define GL_COLOR_ATTACHMENT11_EXT 0x8CEB
#define GL_COLOR_ATTACHMENT12_EXT 0x8CEC
#define GL_COLOR_ATTACHMENT13_EXT 0x8CED
#define GL_COLOR_ATTACHMENT14_EXT 0x8CEE
#define GL_COLOR_ATTACHMENT15_EXT 0x8CEF
#define GL_DEPTH_ATTACHMENT_EXT 0x8D00
#define GL_STENCIL_ATTACHMENT_EXT 0x8D20
#define GL_FRAMEBUFFER_EXT 0x8D40
#define GL_RENDERBUFFER_EXT 0x8D41
#define GL_RENDERBUFFER_WIDTH_EXT 0x8D42
#define GL_RENDERBUFFER_HEIGHT_EXT 0x8D43
#define GL_RENDERBUFFER_INTERNAL_FORMAT_EXT 0x8D44
#define GL_STENCIL_INDEX1_EXT 0x8D46
#define GL_STENCIL_INDEX4_EXT 0x8D47
#define GL_STENCIL_INDEX8_EXT 0x8D48
#define GL_STENCIL_INDEX16_EXT 0x8D49
#define GL_RENDERBUFFER_RED_SIZE_EXT 0x8D50
#define GL_RENDERBUFFER_GREEN_SIZE_EXT 0x8D51
#define GL_RENDERBUFFER_BLUE_SIZE_EXT 0x8D52
#define GL_RENDERBUFFER_ALPHA_SIZE_EXT 0x8D53
#define GL_RENDERBUFFER_DEPTH_SIZE_EXT 0x8D54
#define GL_RENDERBUFFER_STENCIL_SIZE_EXT 0x8D55
#define GL_FRAMEBUFFER_SRGB_EXT 0x8DB9
#define GL_FRAMEBUFFER_SRGB_CAPABLE_EXT 0x8DBA
#define GL_GEOMETRY_SHADER_EXT 0x8DD9
#define GL_GEOMETRY_VERTICES_OUT_EXT 0x8DDA
#define GL_GEOMETRY_INPUT_TYPE_EXT 0x8DDB
#define GL_GEOMETRY_OUTPUT_TYPE_EXT 0x8DDC
#define GL_MAX_GEOMETRY_TEXTURE_IMAGE_UNITS_EXT 0x8C29
#define GL_MAX_GEOMETRY_VARYING_COMPONENTS_EXT 0x8DDD
#define GL_MAX_VERTEX_VARYING_COMPONENTS_EXT 0x8DDE
#define GL_MAX_VARYING_COMPONENTS_EXT 0x8B4B
#define GL_MAX_GEOMETRY_UNIFORM_COMPONENTS_EXT 0x8DDF
#define GL_MAX_GEOMETRY_OUTPUT_VERTICES_EXT 0x8DE0
#define GL_MAX_GEOMETRY_TOTAL_OUTPUT_COMPONENTS_EXT 0x8DE1
#define GL_LINES_ADJACENCY_EXT 0x000A
#define GL_LINE_STRIP_ADJACENCY_EXT 0x000B
#define GL_TRIANGLES_ADJACENCY_EXT 0x000C
#define GL_TRIANGLE_STRIP_ADJACENCY_EXT 0x000D
#define GL_FRAMEBUFFER_INCOMPLETE_LAYER_TARGETS_EXT 0x8DA8
#define GL_FRAMEBUFFER_INCOMPLETE_LAYER_COUNT_EXT 0x8DA9
#define GL_FRAMEBUFFER_ATTACHMENT_LAYERED_EXT 0x8DA7
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_LAYER_EXT 0x8CD4
#define GL_PROGRAM_POINT_SIZE_EXT 0x8642
#define GL_SAMPLER_1D_ARRAY_EXT 0x8DC0
#define GL_SAMPLER_2D_ARRAY_EXT 0x8DC1
#define GL_SAMPLER_BUFFER_EXT 0x8DC2
#define GL_SAMPLER_1D_ARRAY_SHADOW_EXT 0x8DC3
#define GL_SAMPLER_2D_ARRAY_SHADOW_EXT 0x8DC4
#define GL_SAMPLER_CUBE_SHADOW_EXT 0x8DC5
#define GL_UNSIGNED_INT_VEC2_EXT 0x8DC6
#define GL_UNSIGNED_INT_VEC3_EXT 0x8DC7
#define GL_UNSIGNED_INT_VEC4_EXT 0x8DC8
#define GL_INT_SAMPLER_1D_EXT 0x8DC9
#define GL_INT_SAMPLER_2D_EXT 0x8DCA
#define GL_INT_SAMPLER_3D_EXT 0x8DCB
#define GL_INT_SAMPLER_CUBE_EXT 0x8DCC
#define GL_INT_SAMPLER_2D_RECT_EXT 0x8DCD
#define GL_INT_SAMPLER_1D_ARRAY_EXT 0x8DCE
#define GL_INT_SAMPLER_2D_ARRAY_EXT 0x8DCF
#define GL_INT_SAMPLER_BUFFER_EXT 0x8DD0
#define GL_UNSIGNED_INT_SAMPLER_1D_EXT 0x8DD1
#define GL_UNSIGNED_INT_SAMPLER_2D_EXT 0x8DD2
#define GL_UNSIGNED_INT_SAMPLER_3D_EXT 0x8DD3
#define GL_UNSIGNED_INT_SAMPLER_CUBE_EXT 0x8DD4
```

```
#define GL_UNSIGNED_INT_SAMPLER_2D_RECT_EXT 0x8DD5
#define GL_UNSIGNED_INT_SAMPLER_1D_ARRAY_EXT 0x8DD6
#define GL_UNSIGNED_INT_SAMPLER_2D_ARRAY_EXT 0x8DD7
#define GL_UNSIGNED_INT_SAMPLER_BUFFER_EXT 0x8DD8
#define GL_MIN_PROGRAM_TEXEL_OFFSET_EXT 0x8904
#define GL_MAX_PROGRAM_TEXEL_OFFSET_EXT 0x8905
#define GL_VERTEX_ATTRIB_ARRAY_INTEGER_EXT 0x88FD
#define GL_HISTOGRAM_EXT 0x8024
#define GL_PROXY_HISTOGRAM_EXT 0x8025
#define GL_HISTOGRAM_WIDTH_EXT 0x8026
#define GL_HISTOGRAM_FORMAT_EXT 0x8027
#define GL_HISTOGRAM_RED_SIZE_EXT 0x8028
#define GL_HISTOGRAM_GREEN_SIZE_EXT 0x8029
#define GL_HISTOGRAM_BLUE_SIZE_EXT 0x802A
#define GL_HISTOGRAM_ALPHA_SIZE_EXT 0x802B
#define GL_HISTOGRAM_LUMINANCE_SIZE_EXT 0x802C
#define GL_HISTOGRAM_SINK_EXT 0x802D
#define GL_MINMAX_EXT 0x802E
#define GL_MINMAX_FORMAT_EXT 0x802F
#define GL_MINMAX_SINK_EXT 0x8030
#define GL_TABLE_TOO_LARGE_EXT 0x8031
#define GL_IUI_V2F_EXT 0x81AD
#define GL_IUI_V3F_EXT 0x81AE
#define GL_IUI_N3F_V2F_EXT 0x81AF
#define GL_IUI_N3F_V3F_EXT 0x81B0
#define GL_T2F_IUI_V2F_EXT 0x81B1
#define GL_T2F_IUI_V3F_EXT 0x81B2
#define GL_T2F_IUI_N3F_V2F_EXT 0x81B3
#define GL_T2F_IUI_N3F_V3F_EXT 0x81B4
#define GL_INDEX_TEST_EXT 0x81B5
#define GL_INDEX_TEST_FUNC_EXT 0x81B6
#define GL_INDEX_TEST_REF_EXT 0x81B7
#define GL_INDEX_MATERIAL_EXT 0x81B8
#define GL_INDEX_MATERIAL_PARAMETER_EXT 0x81B9
#define GL_INDEX_MATERIAL_FACE_EXT 0x81BA
#define GL_FRAGMENT_MATERIAL_EXT 0x8349
#define GL_FRAGMENT_NORMAL_EXT 0x834A
#define GL_FRAGMENT_COLOR_EXT 0x834C
#define GL_ATTENUATION_EXT 0x834D
#define GL_SHADOW_ATTENUATION_EXT 0x834E
#define GL_TEXTURE_APPLICATION_MODE_EXT 0x834F
#define GL_TEXTURE_LIGHT_EXT 0x8350
#define GL_TEXTURE_MATERIAL_FACE_EXT 0x8351
#define GL_TEXTURE_MATERIAL_PARAMETER_EXT 0x8352
#define GL_TEXTURE_TILING_EXT 0x9580
#define GL_DEDICATED_MEMORY_OBJECT_EXT 0x9581
#define GL_PROTECTED_MEMORY_OBJECT_EXT 0x959B
#define GL_NUM_TILING_TYPES_EXT 0x9582
#define GL_TILING_TYPES_EXT 0x9583
#define GL_OPTIMAL_TILING_EXT 0x9584
#define GL_LINEAR_TILING_EXT 0x9585
#define GL_NUM_DEVICE_UUIDS_EXT 0x9596
#define GL_DEVICE_UUID_EXT 0x9597
#define GL_DRIVER_UUID_EXT 0x9598
#define GL_UUID_SIZE_EXT 16
#define GL_HANDLE_TYPE_OPAQUE_FD_EXT 0x9586
#define GL_HANDLE_TYPE_OPAQUE_WIN32_EXT 0x9587
#define GL_HANDLE_TYPE_OPAQUE_WIN32_KMT_EXT 0x9588
#define GL_DEVICE_LUID_EXT 0x9599
#define GL_DEVICE_NODE_MASK_EXT 0x959A
#define GL_LUID_SIZE_EXT 8
#define GL_HANDLE_TYPE_D3D12_TILEPOOL_EXT 0x9589
#define GL_HANDLE_TYPE_D3D12_RESOURCE_EXT 0x958A
#define GL_HANDLE_TYPE_D3D11_IMAGE_EXT 0x958B
#define GL_HANDLE_TYPE_D3D11_IMAGE_KMT_EXT 0x958C
#define GL_MULTISAMPLE_EXT 0x809D
#define GL_SAMPLE_ALPHA_TO_MASK_EXT 0x809E
#define GL_SAMPLE_ALPHA_TO_ONE_EXT 0x809F
#define GL_SAMPLE_MASK_EXT 0x80A0
#define GL_1PASS_EXT 0x80A1
#define GL_2PASS_0_EXT 0x80A2
#define GL_2PASS_1_EXT 0x80A3
#define GL_4PASS_0_EXT 0x80A4
#define GL_4PASS_1_EXT 0x80A5
#define GL_4PASS_2_EXT 0x80A6
#define GL_4PASS_3_EXT 0x80A7
#define GL_SAMPLE_BUFFERS_EXT 0x80A8
#define GL_SAMPLES_EXT 0x80A9
#define GL_SAMPLE_MASK_VALUE_EXT 0x80AA
#define GL_SAMPLE_MASK_INVERT_EXT 0x80AB
#define GL_SAMPLE_PATTERN_EXT 0x80AC
#define GL_MULTISAMPLE_BIT_EXT 0x20000000
#define GL_DEPTH_STENCIL_EXT 0x84F9
#define GL_UNSIGNED_INT_24_8_EXT 0x84FA
#define GL_DEPTH24_STENCIL8_EXT 0x88F0
#define GL_TEXTURE_STENCIL_SIZE_EXT 0x88F1
#define GL_R11F_G11F_B10F_EXT 0x8C3A
```

```
#define GL_UNSIGNED_INT_10F_11F_11F_REV_EXT 0x8C3B
#define GL_RGBA_SIGNED_COMPONENTS_EXT 0x8C3C
#define GL_UNSIGNED_BYTE_3_3_2_EXT 0x8032
#define GL_UNSIGNED_SHORT_4_4_4_4_EXT 0x8033
#define GL_UNSIGNED_SHORT_5_5_5_1_EXT 0x8034
#define GL_UNSIGNED_INT_8_8_8_8_EXT 0x8035
#define GL_UNSIGNED_INT_10_10_10_2_EXT 0x8036
#define GL_COLOR_INDEX1_EXT 0x80E2
#define GL_COLOR_INDEX2_EXT 0x80E3
#define GL_COLOR_INDEX4_EXT 0x80E4
#define GL_COLOR_INDEX8_EXT 0x80E5
#define GL_COLOR_INDEX12_EXT 0x80E6
#define GL_COLOR_INDEX16_EXT 0x80E7
#define GL_TEXTURE_INDEX_SIZE_EXT 0x80ED
#define GL_PIXEL_PACK_BUFFER_EXT 0x88EB
#define GL_PIXEL_UNPACK_BUFFER_EXT 0x88EC
#define GL_PIXEL_PACK_BUFFER_BINDING_EXT 0x88ED
#define GL_PIXEL_UNPACK_BUFFER_BINDING_EXT 0x88EF
#define GL_PIXEL_TRANSFORM_2D_EXT 0x8330
#define GL_PIXEL_MAG_FILTER_EXT 0x8331
#define GL_PIXEL_MIN_FILTER_EXT 0x8332
#define GL_PIXEL_CUBIC_WEIGHT_EXT 0x8333
#define GL_CUBIC_EXT 0x8334
#define GL_AVERAGE_EXT 0x8335
#define GL_PIXEL_TRANSFORM_2D_STACK_DEPTH_EXT 0x8336
#define GL_MAX_PIXEL_TRANSFORM_2D_STACK_DEPTH_EXT 0x8337
#define GL_PIXEL_TRANSFORM_2D_MATRIX_EXT 0x8338
#define GL_POINT_SIZE_MIN_EXT 0x8126
#define GL_POINT_SIZE_MAX_EXT 0x8127
#define GL_POINT_FADE_THRESHOLD_SIZE_EXT 0x8128
#define GL_DISTANCE_ATTENUATION_EXT 0x8129
#define GL_POLYGON_OFFSET_EXT 0x8037
#define GL_POLYGON_OFFSET_FACTOR_EXT 0x8038
#define GL_POLYGON_OFFSET_BIAS_EXT 0x8039
#define GL_POLYGON_OFFSET_CLAMP_EXT 0x8E1B
#define GL_QUADS_FOLLOW_PROVOKING_VERTEX_CONVENTION_EXT 0x8E4C
#define GL_FIRST_VERTEX_CONVENTION_EXT 0x8E4D
#define GL_LAST_VERTEX_CONVENTION_EXT 0x8E4E
#define GL_PROVOKING_VERTEX_EXT 0x8E4F
#define GL_RASTER_MULTISAMPLE_EXT 0x9327
#define GL_RASTER_SAMPLES_EXT 0x9328
#define GL_MAX_RASTER_SAMPLES_EXT 0x9329
#define GL_RASTER_FIXED_SAMPLE_LOCATIONS_EXT 0x932A
#define GL_MULTISAMPLE_RASTERIZATION_ALLOWED_EXT 0x932B
#define GL_EFFECTIVE_RASTER_SAMPLES_EXT 0x932C
#define GL_RESCALE_NORMAL_EXT 0x803A
#define GL_COLOR_SUM_EXT 0x8458
#define GL_CURRENT_SECONDARY_COLOR_EXT 0x8459
#define GL_SECONDARY_COLOR_ARRAY_SIZE_EXT 0x845A
#define GL_SECONDARY_COLOR_ARRAY_TYPE_EXT 0x845B
#define GL_SECONDARY_COLOR_ARRAY_STRIDE_EXT 0x845C
#define GL_SECONDARY_COLOR_ARRAY_POINTER_EXT 0x845D
#define GL_SECONDARY_COLOR_ARRAY_EXT 0x845E
#define GL_LAYOUT_GENERAL_EXT 0x958D
#define GL_LAYOUT_COLOR_ATTACHMENT_EXT 0x958E
#define GL_LAYOUT_DEPTH_STENCIL_ATTACHMENT_EXT 0x958F
#define GL_LAYOUT_DEPTH_STENCIL_READ_ONLY_EXT 0x9590
#define GL_LAYOUT_SHADER_READ_ONLY_EXT 0x9591
#define GL_LAYOUT_TRANSFER_SRC_EXT 0x9592
#define GL_LAYOUT_TRANSFER_DST_EXT 0x9593
#define GL_LAYOUT_DEPTH_READ_ONLY_STENCIL_ATTACHMENT_EXT 0x9530
#define GL_LAYOUT_DEPTH_ATTACHMENT_STENCIL_READ_ONLY_EXT 0x9531
#define GL_HANDLE_TYPE_D3D12_FENCE_EXT 0x9594
#define GL_D3D12_FENCE_VALUE_EXT 0x9595
#define GL_ACTIVE_PROGRAM_EXT 0x8B8D
#define GL_VERTEX_SHADER_BIT_EXT 0x00000001
#define GL_FRAGMENT_SHADER_BIT_EXT 0x00000002
#define GL_ALL_SHADER_BITS_EXT 0xFFFFFFFF
#define GL_PROGRAM_SEPARABLE_EXT 0x8258
#define GL_PROGRAM_PIPELINE_BINDING_EXT 0x825A
#define GL_LIGHT_MODEL_COLOR_CONTROL_EXT 0x81F8
#define GL_SINGLE_COLOR_EXT 0x81F9
#define GL_SEPARATE_SPECULAR_COLOR_EXT 0x81FA
#define GL_FRAGMENT_SHADER_DISCARDS_SAMPLES_EXT 0x8A52
#define GL_MAX_IMAGE_UNITS_EXT 0x8F38
#define GL_MAX_COMBINED_IMAGE_UNITS_AND_FRAGMENT_OUTPUTS_EXT 0x8F39
#define GL_IMAGE_BINDING_NAME_EXT 0x8F3A
#define GL_IMAGE_BINDING_LEVEL_EXT 0x8F3B
#define GL_IMAGE_BINDING_LAYERED_EXT 0x8F3C
#define GL_IMAGE_BINDING_LAYER_EXT 0x8F3D
#define GL_IMAGE_BINDING_ACCESS_EXT 0x8F3E
#define GL_IMAGE_1D_EXT 0x904C
#define GL_IMAGE_2D_EXT 0x904D
#define GL_IMAGE_3D_EXT 0x904E
#define GL_IMAGE_2D_RECT_EXT 0x904F
#define GL_IMAGE_CUBE_EXT 0x9050
#define GL_IMAGE_BUFFER_EXT 0x9051
```

```
#define GL_IMAGE_1D_ARRAY_EXT 0x9052
#define GL_IMAGE_2D_ARRAY_EXT 0x9053
#define GL_IMAGE_CUBE_MAP_ARRAY_EXT 0x9054
#define GL_IMAGE_2D_MULTISAMPLE_EXT 0x9055
#define GL_IMAGE_2D_MULTISAMPLE_ARRAY_EXT 0x9056
#define GL_INT_IMAGE_1D_EXT 0x9057
#define GL_INT_IMAGE_2D_EXT 0x9058
#define GL_INT_IMAGE_3D_EXT 0x9059
#define GL_INT_IMAGE_2D_RECT_EXT 0x905A
#define GL_INT_IMAGE_CUBE_EXT 0x905B
#define GL_INT_IMAGE_BUFFER_EXT 0x905C
#define GL_INT_IMAGE_1D_ARRAY_EXT 0x905D
#define GL_INT_IMAGE_2D_ARRAY_EXT 0x905E
#define GL_INT_IMAGE_CUBE_MAP_ARRAY_EXT 0x905F
#define GL_INT_IMAGE_2D_MULTISAMPLE_EXT 0x9060
#define GL_INT_IMAGE_2D_MULTISAMPLE_ARRAY_EXT 0x9061
#define GL_UNSIGNED_INT_IMAGE_1D_EXT 0x9062
#define GL_UNSIGNED_INT_IMAGE_2D_EXT 0x9063
#define GL_UNSIGNED_INT_IMAGE_3D_EXT 0x9064
#define GL_UNSIGNED_INT_IMAGE_2D_RECT_EXT 0x9065
#define GL_UNSIGNED_INT_IMAGE_CUBE_EXT 0x9066
#define GL_UNSIGNED_INT_IMAGE_BUFFER_EXT 0x9067
#define GL_UNSIGNED_INT_IMAGE_1D_ARRAY_EXT 0x9068
#define GL_UNSIGNED_INT_IMAGE_2D_ARRAY_EXT 0x9069
#define GL_UNSIGNED_INT_IMAGE_CUBE_MAP_ARRAY_EXT 0x906A
#define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE_EXT 0x906B
#define GL_UNSIGNED_INT_IMAGE_2D_MULTISAMPLE_ARRAY_EXT 0x906C
#define GL_MAX_IMAGE_SAMPLES_EXT 0x906D
#define GL_IMAGE_BINDING_FORMAT_EXT 0x906E
#define GL_VERTEX_ATTRIB_ARRAY_BARRIER_BIT_EXT 0x00000001
#define GL_ELEMENT_ARRAY_BARRIER_BIT_EXT 0x00000002
#define GL_UNIFORM_BARRIER_BIT_EXT 0x00000004
#define GL_TEXTURE_FETCH_BARRIER_BIT_EXT 0x00000008
#define GL_SHADER_IMAGE_ACCESS_BARRIER_BIT_EXT 0x00000020
#define GL_COMMAND_BARRIER_BIT_EXT 0x00000040
#define GL_PIXEL_BUFFER_BARRIER_BIT_EXT 0x00000080
#define GL_TEXTURE_UPDATE_BARRIER_BIT_EXT 0x00000100
#define GL_BUFFER_UPDATE_BARRIER_BIT_EXT 0x00000200
#define GL_FRAMEBUFFER_BARRIER_BIT_EXT 0x00000400
#define GL_TRANSFORM_FEEDBACK_BARRIER_BIT_EXT 0x00000800
#define GL_ATOMIC_COUNTER_BARRIER_BIT_EXT 0x00001000
#define GL_ALL_BARRIER_BITS_EXT 0xFFFFFFFF
#define GL_SHARED_TEXTURE_PALETTE_EXT 0x81FB
#define GL_STENCIL_TAG_BITS_EXT 0x88F2
#define GL_STENCIL_CLEAR_TAG_VALUE_EXT 0x88F3
#define GL_STENCIL_TEST_TWO_SIDE_EXT 0x8910
#define GL_ACTIVE_STENCIL_FACE_EXT 0x8911
#define GL_INCR_WRAP_EXT 0x8507
#define GL_DECR_WRAP_EXT 0x8508
#define GL_ALPHA4_EXT 0x803B
#define GL_ALPHA8_EXT 0x803C
#define GL_ALPHA12_EXT 0x803D
#define GL_ALPHA16_EXT 0x803E
#define GL_LUMINANCE4_EXT 0x803F
#define GL_LUMINANCE8_EXT 0x8040
#define GL_LUMINANCE12_EXT 0x8041
#define GL_LUMINANCE16_EXT 0x8042
#define GL_LUMINANCE4_ALPHA4_EXT 0x8043
#define GL_LUMINANCE6_ALPHA2_EXT 0x8044
#define GL_LUMINANCE8_ALPHA8_EXT 0x8045
#define GL_LUMINANCE12_ALPHA4_EXT 0x8046
#define GL_LUMINANCE12_ALPHA12_EXT 0x8047
#define GL_LUMINANCE16_ALPHA16_EXT 0x8048
#define GL_INTENSITY_EXT 0x8049
#define GL_INTENSITY4_EXT 0x804A
#define GL_INTENSITY8_EXT 0x804B
#define GL_INTENSITY12_EXT 0x804C
#define GL_INTENSITY16_EXT 0x804D
#define GL_RGB2_EXT 0x804E
#define GL_RGB4_EXT 0x804F
#define GL_RGB5_EXT 0x8050
#define GL_RGB8_EXT 0x8051
#define GL_RGB10_EXT 0x8052
#define GL_RGB12_EXT 0x8053
#define GL_RGB16_EXT 0x8054
#define GL_RGBA2_EXT 0x8055
#define GL_RGBA4_EXT 0x8056
#define GL_RGB5_A1_EXT 0x8057
#define GL_RGBA8_EXT 0x8058
#define GL_RGB10_A2_EXT 0x8059
#define GL_RGBA12_EXT 0x805A
#define GL_RGBA16_EXT 0x805B
#define GL_TEXTURE_RED_SIZE_EXT 0x805C
#define GL_TEXTURE_GREEN_SIZE_EXT 0x805D
#define GL_TEXTURE_BLUE_SIZE_EXT 0x805E
#define GL_TEXTURE_ALPHA_SIZE_EXT 0x805F
#define GL_TEXTURE_LUMINANCE_SIZE_EXT 0x8060
```

```
#define GL_TEXTURE_INTENSITY_SIZE_EXT 0x8061
#define GL_REPLACE_EXT 0x8062
#define GL_PROXY_TEXTURE_1D_EXT 0x8063
#define GL_PROXY_TEXTURE_2D_EXT 0x8064
#define GL_TEXTURE_TOO_LARGE_EXT 0x8065
#define GL_PACK_SKIP_IMAGES_EXT 0x806B
#define GL_PACK_IMAGE_HEIGHT_EXT 0x806C
#define GL_UNPACK_SKIP_IMAGES_EXT 0x806D
#define GL_UNPACK_IMAGE_HEIGHT_EXT 0x806E
#define GL_TEXTURE_3D_EXT 0x806F
#define GL_PROXY_TEXTURE_3D_EXT 0x8070
#define GL_TEXTURE_DEPTH_EXT 0x8071
#define GL_TEXTURE_WRAP_R_EXT 0x8072
#define GL_MAX_3D_TEXTURE_SIZE_EXT 0x8073
#define GL_TEXTURE_1D_ARRAY_EXT 0x8C18
#define GL_PROXY_TEXTURE_1D_ARRAY_EXT 0x8C19
#define GL_TEXTURE_2D_ARRAY_EXT 0x8C1A
#define GL_PROXY_TEXTURE_2D_ARRAY_EXT 0x8C1B
#define GL_TEXTURE_BINDING_1D_ARRAY_EXT 0x8C1C
#define GL_TEXTURE_BINDING_2D_ARRAY_EXT 0x8C1D
#define GL_MAX_ARRAY_TEXTURE_LAYERS_EXT 0x88FF
#define GL_COMPARE_REF_DEPTH_TO_TEXTURE_EXT 0x884E
#define GL_TEXTURE_BUFFER_EXT 0x8C2A
#define GL_MAX_TEXTURE_BUFFER_SIZE_EXT 0x8C2B
#define GL_TEXTURE_BINDING_BUFFER_EXT 0x8C2C
#define GL_TEXTURE_BUFFER_DATA_STORE_BINDING_EXT 0x8C2D
#define GL_TEXTURE_BUFFER_FORMAT_EXT 0x8C2E
#define GL_COMPRESSED_LUMINANCE_LATC1_EXT 0x8C70
#define GL_COMPRESSED_SIGNED_LUMINANCE_LATC1_EXT 0x8C71
#define GL_COMPRESSED_LUMINANCE_ALPHA_LATC2_EXT 0x8C72
#define GL_COMPRESSED_SIGNED_LUMINANCE_ALPHA_LATC2_EXT 0x8C73
#define GL_COMPRESSED_RED_RGTC1_EXT 0x8DBB
#define GL_COMPRESSED_SIGNED_RED_RGTC1_EXT 0x8DBC
#define GL_COMPRESSED_RED_GREEN_RGTC2_EXT 0x8DBD
#define GL_COMPRESSED_SIGNED_RED_GREEN_RGTC2_EXT 0x8DBE
#define GL_COMPRESSED_RGB_S3TC_DXT1_EXT 0x83F0
#define GL_COMPRESSED_RGBA_S3TC_DXT1_EXT 0x83F1
#define GL_COMPRESSED_RGBA_S3TC_DXT3_EXT 0x83F2
#define GL_COMPRESSED_RGBA_S3TC_DXT5_EXT 0x83F3
#define GL_NORMAL_MAP_EXT 0x8511
#define GL_REFLECTION_MAP_EXT 0x8512
#define GL_TEXTURE_CUBE_MAP_EXT 0x8513
#define GL_TEXTURE_BINDING_CUBE_MAP_EXT 0x8514
#define GL_TEXTURE_CUBE_MAP_POSITIVE_X_EXT 0x8515
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_X_EXT 0x8516
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Y_EXT 0x8517
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Y_EXT 0x8518
#define GL_TEXTURE_CUBE_MAP_POSITIVE_Z_EXT 0x8519
#define GL_TEXTURE_CUBE_MAP_NEGATIVE_Z_EXT 0x851A
#define GL_PROXY_TEXTURE_CUBE_MAP_EXT 0x851B
#define GL_MAX_CUBE_MAP_TEXTURE_SIZE_EXT 0x851C
#define GL_COMBINE_EXT 0x8570
#define GL_COMBINE_RGB_EXT 0x8571
#define GL_COMBINE_ALPHA_EXT 0x8572
#define GL_RGB_SCALE_EXT 0x8573
#define GL_ADD_SIGNED_EXT 0x8574
#define GL_INTERPOLATE_EXT 0x8575
#define GL_CONSTANT_EXT 0x8576
#define GL_PRIMARY_COLOR_EXT 0x8577
#define GL_PREVIOUS_EXT 0x8578
#define GL_SOURCE0_RGB_EXT 0x8580
#define GL_SOURCE1_RGB_EXT 0x8581
#define GL_SOURCE2_RGB_EXT 0x8582
#define GL_SOURCE0_ALPHA_EXT 0x8588
#define GL_SOURCE1_ALPHA_EXT 0x8589
#define GL_SOURCE2_ALPHA_EXT 0x858A
#define GL_OPERAND0_RGB_EXT 0x8590
#define GL_OPERAND1_RGB_EXT 0x8591
#define GL_OPERAND2_RGB_EXT 0x8592
#define GL_OPERAND0_ALPHA_EXT 0x8598
#define GL_OPERAND1_ALPHA_EXT 0x8599
#define GL_OPERAND2_ALPHA_EXT 0x859A
#define GL_DOT3_RGB_EXT 0x8740
#define GL_DOT3_RGBA_EXT 0x8741
#define GL_TEXTURE_MAX_ANISOTROPY_EXT 0x84FE
#define GL_MAX_TEXTURE_MAX_ANISOTROPY_EXT 0x84FF
#define GL_TEXTURE_REDUCTION_MODE_EXT 0x9366
#define GL_WEIGHTED_AVERAGE_EXT 0x9367
#define GL_RGBA32UI_EXT 0x8D70
#define GL_RGB32UI_EXT 0x8D71
#define GL_ALPHA32UI_EXT 0x8D72
#define GL_INTENSITY32UI_EXT 0x8D73
#define GL_LUMINANCE32UI_EXT 0x8D74
#define GL_LUMINANCE_ALPHA32UI_EXT 0x8D75
#define GL_RGBA16UI_EXT 0x8D76
#define GL_RGB16UI_EXT 0x8D77
#define GL_ALPHA16UI_EXT 0x8D78
```

```
#define GL_INTENSITY16UI_EXT 0x8D79
#define GL_LUMINANCE16UI_EXT 0x8D7A
#define GL_LUMINANCE_ALPHA16UI_EXT 0x8D7B
#define GL_RGBA8UI_EXT 0x8D7C
#define GL_RGB8UI_EXT 0x8D7D
#define GL_ALPHA8UI_EXT 0x8D7E
#define GL_INTENSITY8UI_EXT 0x8D7F
#define GL_LUMINANCE8UI_EXT 0x8D80
#define GL_LUMINANCE_ALPHA8UI_EXT 0x8D81
#define GL_RGBA32I_EXT 0x8D82
#define GL_RGB32I_EXT 0x8D83
#define GL_ALPHA32I_EXT 0x8D84
#define GL_INTENSITY32I_EXT 0x8D85
#define GL_LUMINANCE32I_EXT 0x8D86
#define GL_LUMINANCE_ALPHA32I_EXT 0x8D87
#define GL_RGBA16I_EXT 0x8D88
#define GL_RGB16I_EXT 0x8D89
#define GL_ALPHA16I_EXT 0x8D8A
#define GL_INTENSITY16I_EXT 0x8D8B
#define GL_LUMINANCE16I_EXT 0x8D8C
#define GL_LUMINANCE_ALPHA16I_EXT 0x8D8D
#define GL_RGBA8I_EXT 0x8D8E
#define GL_RGB8I_EXT 0x8D8F
#define GL_ALPHA8I_EXT 0x8D90
#define GL_INTENSITY8I_EXT 0x8D91
#define GL_LUMINANCE8I_EXT 0x8D92
#define GL_LUMINANCE_ALPHA8I_EXT 0x8D93
#define GL_RED_INTEGER_EXT 0x8D94
#define GL_GREEN_INTEGER_EXT 0x8D95
#define GL_BLUE_INTEGER_EXT 0x8D96
#define GL_ALPHA_INTEGER_EXT 0x8D97
#define GL_RGB_INTEGER_EXT 0x8D98
#define GL_RGBA_INTEGER_EXT 0x8D99
#define GL_BGR_INTEGER_EXT 0x8D9A
#define GL_BGRA_INTEGER_EXT 0x8D9B
#define GL_LUMINANCE_INTEGER_EXT 0x8D9C
#define GL_LUMINANCE_ALPHA_INTEGER_EXT 0x8D9D
#define GL_RGBA_INTEGER_MODE_EXT 0x8D9E
#define GL_MAX_TEXTURE_LOD_BIAS_EXT 0x84FD
#define GL_TEXTURE_FILTER_CONTROL_EXT 0x8500
#define GL_TEXTURE_LOD_BIAS_EXT 0x8501
#define GL_MIRROR_CLAMP_EXT 0x8742
#define GL_MIRROR_CLAMP_TO_EDGE_EXT 0x8743
#define GL_MIRROR_CLAMP_TO_BORDER_EXT 0x8912
#define GL_TEXTURE_PRIORITY_EXT 0x8066
#define GL_TEXTURE_RESIDENT_EXT 0x8067
#define GL_TEXTURE_1D_BINDING_EXT 0x8068
#define GL_TEXTURE_2D_BINDING_EXT 0x8069
#define GL_TEXTURE_3D_BINDING_EXT 0x806A
#define GL_PERTURB_EXT 0x85AE
#define GL_TEXTURE_NORMAL_EXT 0x85AF
#define GL_SRGB_EXT 0x8C40
#define GL_SRGB8_EXT 0x8C41
#define GL_SRGB_ALPHA_EXT 0x8C42
#define GL_SRGB8_ALPHA8_EXT 0x8C43
#define GL_SLUMINANCE_ALPHA_EXT 0x8C44
#define GL_SLUMINANCE8_ALPHA8_EXT 0x8C45
#define GL_SLUMINANCE_EXT 0x8C46
#define GL_SLUMINANCE8_EXT 0x8C47
#define GL_COMPRESSED_SRGB_EXT 0x8C48
#define GL_COMPRESSED_SRGB_ALPHA_EXT 0x8C49
#define GL_COMPRESSED_SLUMINANCE_EXT 0x8C4A
#define GL_COMPRESSED_SLUMINANCE_ALPHA_EXT 0x8C4B
#define GL_COMPRESSED_SRGB_S3TC_DXT1_EXT 0x8C4C
#define GL_COMPRESSED_SRGB_ALPHA_S3TC_DXT1_EXT 0x8C4D
#define GL_COMPRESSED_SRGB_ALPHA_S3TC_DXT3_EXT 0x8C4E
#define GL_COMPRESSED_SRGB_ALPHA_S3TC_DXT5_EXT 0x8C4F
#define GL_SR8_EXT 0x8FBD
#define GL_SRG8_EXT 0x8FBE
#define GL_TEXTURE_SRGB_DECODE_EXT 0x8A48
#define GL_DECODE_EXT 0x8A49
#define GL_SKIP_DECODE_EXT 0x8A4A
#define GL_RGB9_E5_EXT 0x8C3D
#define GL_UNSIGNED_INT_5_9_9_9_REV_EXT 0x8C3E
#define GL_TEXTURE_SHARED_SIZE_EXT 0x8C3F
#define GL_ALPHA_SNORM 0x9010
#define GL_LUMINANCE_SNORM 0x9011
#define GL_LUMINANCE_ALPHA_SNORM 0x9012
#define GL_INTENSITY_SNORM 0x9013
#define GL_ALPHA8_SNORM 0x9014
#define GL_LUMINANCE8_SNORM 0x9015
#define GL_LUMINANCE8_ALPHA8_SNORM 0x9016
#define GL_INTENSITY8_SNORM 0x9017
#define GL_ALPHA16_SNORM 0x9018
#define GL_LUMINANCE16_SNORM 0x9019
#define GL_LUMINANCE16_ALPHA16_SNORM 0x901A
#define GL_INTENSITY16_SNORM 0x901B
```

```
#define GL_RED_SNORM 0x8F90
#define GL_RG_SNORM 0x8F91
#define GL_RGB_SNORM 0x8F92
#define GL_RGBA_SNORM 0x8F93
#define GL_TEXTURE_IMMUTABLE_FORMAT_EXT 0x912F
#define GL_RGBA32F_EXT 0x8814
#define GL_RGB32F_EXT 0x8815
#define GL_ALPHA32F_EXT 0x8816
#define GL_LUMINANCE32F_EXT 0x8818
#define GL_LUMINANCE_ALPHA32F_EXT 0x8819
#define GL_RGBA16F_EXT 0x881A
#define GL_RGB16F_EXT 0x881B
#define GL_ALPHA16F_EXT 0x881C
#define GL_LUMINANCE16F_EXT 0x881E
#define GL_LUMINANCE_ALPHA16F_EXT 0x881F
#define GL_BGRA8_EXT 0x93A1
#define GL_R8_EXT 0x8229
#define GL_RG8_EXT 0x822B
#define GL_R32F_EXT 0x822E
#define GL_RG32F_EXT 0x8230
#define GL_R16F_EXT 0x822D
#define GL_RG16F_EXT 0x822F
#define GL_TEXTURE_SWIZZLE_R_EXT 0x8E42
#define GL_TEXTURE_SWIZZLE_G_EXT 0x8E43
#define GL_TEXTURE_SWIZZLE_B_EXT 0x8E44
#define GL_TEXTURE_SWIZZLE_A_EXT 0x8E45
#define GL_TEXTURE_SWIZZLE_RGBA_EXT 0x8E46
#define GL_TIME_ELAPSED_EXT 0x88BF
#define GL_TRANSFORM_FEEDBACK_BUFFER_EXT 0x8C8E
#define GL_TRANSFORM_FEEDBACK_BUFFER_START_EXT 0x8C84
#define GL_TRANSFORM_FEEDBACK_BUFFER_SIZE_EXT 0x8C85
#define GL_TRANSFORM_FEEDBACK_BUFFER_BINDING_EXT 0x8C8F
#define GL_INTERLEAVED_ATTRIBS_EXT 0x8C8C
#define GL_SEPARATE_ATTRIBS_EXT 0x8C8D
#define GL_PRIMITIVES_GENERATED_EXT 0x8C87
#define GL_TRANSFORM_FEEDBACK_PRIMITIVES_WRITTEN_EXT 0x8C88
#define GL_RASTERIZER_DISCARD_EXT 0x8C89
#define GL_MAX_TRANSFORM_FEEDBACK_INTERLEAVED_COMPONENTS_EXT 0x8C8A
#define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_ATTRIBS_EXT 0x8C8B
#define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_COMPONENTS_EXT 0x8C80
#define GL_TRANSFORM_FEEDBACK_VARYINGS_EXT 0x8C83
#define GL_TRANSFORM_FEEDBACK_BUFFER_MODE_EXT 0x8C7F
#define GL_TRANSFORM_FEEDBACK_VARYING_MAX_LENGTH_EXT 0x8C76
#define GL_VERTEX_ARRAY_EXT 0x8074
#define GL_NORMAL_ARRAY_EXT 0x8075
#define GL_COLOR_ARRAY_EXT 0x8076
#define GL_INDEX_ARRAY_EXT 0x8077
#define GL_TEXTURE_COORD_ARRAY_EXT 0x8078
#define GL_EDGE_FLAG_ARRAY_EXT 0x8079
#define GL_VERTEX_ARRAY_SIZE_EXT 0x807A
#define GL_VERTEX_ARRAY_TYPE_EXT 0x807B
#define GL_VERTEX_ARRAY_STRIDE_EXT 0x807C
#define GL_VERTEX_ARRAY_COUNT_EXT 0x807D
#define GL_NORMAL_ARRAY_TYPE_EXT 0x807E
#define GL_NORMAL_ARRAY_STRIDE_EXT 0x807F
#define GL_NORMAL_ARRAY_COUNT_EXT 0x8080
#define GL_COLOR_ARRAY_SIZE_EXT 0x8081
#define GL_COLOR_ARRAY_TYPE_EXT 0x8082
#define GL_COLOR_ARRAY_STRIDE_EXT 0x8083
#define GL_COLOR_ARRAY_COUNT_EXT 0x8084
#define GL_INDEX_ARRAY_TYPE_EXT 0x8085
#define GL_INDEX_ARRAY_STRIDE_EXT 0x8086
#define GL_INDEX_ARRAY_COUNT_EXT 0x8087
#define GL_TEXTURE_COORD_ARRAY_SIZE_EXT 0x8088
#define GL_TEXTURE_COORD_ARRAY_TYPE_EXT 0x8089
#define GL_TEXTURE_COORD_ARRAY_STRIDE_EXT 0x808A
#define GL_TEXTURE_COORD_ARRAY_COUNT_EXT 0x808B
#define GL_EDGE_FLAG_ARRAY_STRIDE_EXT 0x808C
#define GL_EDGE_FLAG_ARRAY_COUNT_EXT 0x808D
#define GL_VERTEX_ARRAY_POINTER_EXT 0x808E
#define GL_NORMAL_ARRAY_POINTER_EXT 0x808F
#define GL_COLOR_ARRAY_POINTER_EXT 0x8090
#define GL_INDEX_ARRAY_POINTER_EXT 0x8091
#define GL_TEXTURE_COORD_ARRAY_POINTER_EXT 0x8092
#define GL_EDGE_FLAG_ARRAY_POINTER_EXT 0x8093
#define GL_DOUBLE_VEC2_EXT 0x8FFC
#define GL_DOUBLE_VEC3_EXT 0x8FFD
#define GL_DOUBLE_VEC4_EXT 0x8FFE
#define GL_DOUBLE_MAT2_EXT 0x8F46
#define GL_DOUBLE_MAT3_EXT 0x8F47
#define GL_DOUBLE_MAT4_EXT 0x8F48
#define GL_DOUBLE_MAT2x3_EXT 0x8F49
#define GL_DOUBLE_MAT2x4_EXT 0x8F4A
#define GL_DOUBLE_MAT3x2_EXT 0x8F4B
#define GL_DOUBLE_MAT3x4_EXT 0x8F4C
#define GL_DOUBLE_MAT4x2_EXT 0x8F4D
#define GL_DOUBLE_MAT4x3_EXT 0x8F4E
```

```
#define GL_VERTEX_SHADER_EXT 0x8780
#define GL_VERTEX_SHADER_BINDING_EXT 0x8781
#define GL_OP_INDEX_EXT 0x8782
#define GL_OP_NEGATE_EXT 0x8783
#define GL_OP_DOT3_EXT 0x8784
#define GL_OP_DOT4_EXT 0x8785
#define GL_OP_MUL_EXT 0x8786
#define GL_OP_ADD_EXT 0x8787
#define GL_OP_MADD_EXT 0x8788
#define GL_OP_FRAC_EXT 0x8789
#define GL_OP_MAX_EXT 0x878A
#define GL_OP_MIN_EXT 0x878B
#define GL_OP_SET_GE_EXT 0x878C
#define GL_OP_SET_LT_EXT 0x878D
#define GL_OP_CLAMP_EXT 0x878E
#define GL_OP_FLOOR_EXT 0x878F
#define GL_OP_ROUND_EXT 0x8790
#define GL_OP_EXP_BASE_2_EXT 0x8791
#define GL_OP_LOG_BASE_2_EXT 0x8792
#define GL_OP_POWER_EXT 0x8793
#define GL_OP_RECIP_EXT 0x8794
#define GL_OP_RECIP_SQRT_EXT 0x8795
#define GL_OP_SUB_EXT 0x8796
#define GL_OP_CROSS_PRODUCT_EXT 0x8797
#define GL_OP_MULTIPLY_MATRIX_EXT 0x8798
#define GL_OP_MOV_EXT 0x8799
#define GL_OUTPUT_VERTEX_EXT 0x879A
#define GL_OUTPUT_COLOR0_EXT 0x879B
#define GL_OUTPUT_COLOR1_EXT 0x879C
#define GL_OUTPUT_TEXTURE_COORD0_EXT 0x879D
#define GL_OUTPUT_TEXTURE_COORD1_EXT 0x879E
#define GL_OUTPUT_TEXTURE_COORD2_EXT 0x879F
#define GL_OUTPUT_TEXTURE_COORD3_EXT 0x87A0
#define GL_OUTPUT_TEXTURE_COORD4_EXT 0x87A1
#define GL_OUTPUT_TEXTURE_COORD5_EXT 0x87A2
#define GL_OUTPUT_TEXTURE_COORD6_EXT 0x87A3
#define GL_OUTPUT_TEXTURE_COORD7_EXT 0x87A4
#define GL_OUTPUT_TEXTURE_COORD8_EXT 0x87A5
#define GL_OUTPUT_TEXTURE_COORD9_EXT 0x87A6
#define GL_OUTPUT_TEXTURE_COORD10_EXT 0x87A7
#define GL_OUTPUT_TEXTURE_COORD11_EXT 0x87A8
#define GL_OUTPUT_TEXTURE_COORD12_EXT 0x87A9
#define GL_OUTPUT_TEXTURE_COORD13_EXT 0x87AA
#define GL_OUTPUT_TEXTURE_COORD14_EXT 0x87AB
#define GL_OUTPUT_TEXTURE_COORD15_EXT 0x87AC
#define GL_OUTPUT_TEXTURE_COORD16_EXT 0x87AD
#define GL_OUTPUT_TEXTURE_COORD17_EXT 0x87AE
#define GL_OUTPUT_TEXTURE_COORD18_EXT 0x87AF
#define GL_OUTPUT_TEXTURE_COORD19_EXT 0x87B0
#define GL_OUTPUT_TEXTURE_COORD20_EXT 0x87B1
#define GL_OUTPUT_TEXTURE_COORD21_EXT 0x87B2
#define GL_OUTPUT_TEXTURE_COORD22_EXT 0x87B3
#define GL_OUTPUT_TEXTURE_COORD23_EXT 0x87B4
#define GL_OUTPUT_TEXTURE_COORD24_EXT 0x87B5
#define GL_OUTPUT_TEXTURE_COORD25_EXT 0x87B6
#define GL_OUTPUT_TEXTURE_COORD26_EXT 0x87B7
#define GL_OUTPUT_TEXTURE_COORD27_EXT 0x87B8
#define GL_OUTPUT_TEXTURE_COORD28_EXT 0x87B9
#define GL_OUTPUT_TEXTURE_COORD29_EXT 0x87BA
#define GL_OUTPUT_TEXTURE_COORD30_EXT 0x87BB
#define GL_OUTPUT_TEXTURE_COORD31_EXT 0x87BC
#define GL_OUTPUT_FOG_EXT 0x87BD
#define GL_SCALAR_EXT 0x87BE
#define GL_VECTOR_EXT 0x87BF
#define GL_MATRIX_EXT 0x87C0
#define GL_VARIANT_EXT 0x87C1
#define GL_INVARIANT_EXT 0x87C2
#define GL_LOCAL_CONSTANT_EXT 0x87C3
#define GL_LOCAL_EXT 0x87C4
#define GL_MAX_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87C5
#define GL_MAX_VERTEX_SHADER_VARIANTS_EXT 0x87C6
#define GL_MAX_VERTEX_SHADER_INVARIANTS_EXT 0x87C7
#define GL_MAX_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87C8
#define GL_MAX_VERTEX_SHADER_LOCALS_EXT 0x87C9
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87CA
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_VARIANTS_EXT 0x87CB
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87CC
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_INVARIANTS_EXT 0x87CD
#define GL_MAX_OPTIMIZED_VERTEX_SHADER_LOCALS_EXT 0x87CE
#define GL_VERTEX_SHADER_INSTRUCTIONS_EXT 0x87CF
#define GL_VERTEX_SHADER_VARIANTS_EXT 0x87D0
#define GL_VERTEX_SHADER_INVARIANTS_EXT 0x87D1
#define GL_VERTEX_SHADER_LOCAL_CONSTANTS_EXT 0x87D2
#define GL_VERTEX_SHADER_LOCALS_EXT 0x87D3
#define GL_VERTEX_SHADER_OPTIMIZED_EXT 0x87D4
#define GL_X_EXT 0x87D5
#define GL_Y_EXT 0x87D6
```

```
#define GL_Z_EXT 0x87D7
#define GL_W_EXT 0x87D8
#define GL_NEGATIVE_X_EXT 0x87D9
#define GL_NEGATIVE_Y_EXT 0x87DA
#define GL_NEGATIVE_Z_EXT 0x87DB
#define GL_NEGATIVE_W_EXT 0x87DC
#define GL_ZERO_EXT 0x87DD
#define GL_ONE_EXT 0x87DE
#define GL_NEGATIVE_ONE_EXT 0x87DF
#define GL_NORMALIZED_RANGE_EXT 0x87E0
#define GL_FULL_RANGE_EXT 0x87E1
#define GL_CURRENT_VERTEX_EXT 0x87E2
#define GL_MVP_MATRIX_EXT 0x87E3
#define GL_VARIANT_VALUE_EXT 0x87E4
#define GL_VARIANT_DATATYPE_EXT 0x87E5
#define GL_VARIANT_ARRAY_STRIDE_EXT 0x87E6
#define GL_VARIANT_ARRAY_TYPE_EXT 0x87E7
#define GL_VARIANT_ARRAY_EXT 0x87E8
#define GL_VARIANT_ARRAY_POINTER_EXT 0x87E9
#define GL_INVARIANT_VALUE_EXT 0x87EA
#define GL_INVARIANT_DATATYPE_EXT 0x87EB
#define GL_LOCAL_CONSTANT_VALUE_EXT 0x87EC
#define GL_LOCAL_CONSTANT_DATATYPE_EXT 0x87ED
#define GL_MODELVIEW0_STACK_DEPTH_EXT 0x0BA3
#define GL_MODELVIEW1_STACK_DEPTH_EXT 0x8502
#define GL_MODELVIEW0_MATRIX_EXT 0x0BA6
#define GL_MODELVIEW1_MATRIX_EXT 0x8506
#define GL_VERTEX_WEIGHTING_EXT 0x8509
#define GL_MODELVIEW0_EXT 0x1700
#define GL_MODELVIEW1_EXT 0x850A
#define GL_CURRENT_VERTEX_WEIGHT_EXT 0x850B
#define GL_VERTEX_WEIGHT_ARRAY_EXT 0x850C
#define GL_VERTEX_WEIGHT_ARRAY_SIZE_EXT 0x850D
#define GL_VERTEX_WEIGHT_ARRAY_TYPE_EXT 0x850E
#define GL_VERTEX_WEIGHT_ARRAY_STRIDE_EXT 0x850F
#define GL_VERTEX_WEIGHT_ARRAY_POINTER_EXT 0x8510
#define GL_INCLUSIVE_EXT 0x8F10
#define GL_EXCLUSIVE_EXT 0x8F11
#define GL_WINDOW_RECTANGLE_EXT 0x8F12
#define GL_WINDOW_RECTANGLE_MODE_EXT 0x8F13
#define GL_MAX_WINDOW_RECTANGLES_EXT 0x8F14
#define GL_NUM_WINDOW_RECTANGLES_EXT 0x8F15
#define GL_SYNC_X11_FENCE_EXT 0x90E1
#define GL_IGNORE_BORDER_HP 0x8150
#define GL_CONSTANT_BORDER_HP 0x8151
#define GL_REPLICATE_BORDER_HP 0x8153
#define GL_CONVOLUTION_BORDER_COLOR_HP 0x8154
#define GL_IMAGE_SCALE_X_HP 0x8155
#define GL_IMAGE_SCALE_Y_HP 0x8156
#define GL_IMAGE_TRANSLATE_X_HP 0x8157
#define GL_IMAGE_TRANSLATE_Y_HP 0x8158
#define GL_IMAGE_ROTATE_ANGLE_HP 0x8159
#define GL_IMAGE_ROTATE_ORIGIN_X_HP 0x815A
#define GL_IMAGE_ROTATE_ORIGIN_Y_HP 0x815B
#define GL_IMAGE_MAG_FILTER_HP 0x815C
#define GL_IMAGE_MIN_FILTER_HP 0x815D
#define GL_IMAGE_CUBIC_WEIGHT_HP 0x815E
#define GL_CUBIC_HP 0x815F
#define GL_AVERAGE_HP 0x8160
#define GL_IMAGE_TRANSFORM_2D_HP 0x8161
#define GL_POST_IMAGE_TRANSFORM_COLOR_TABLE_HP 0x8162
#define GL_PROXY_POST_IMAGE_TRANSFORM_COLOR_TABLE_HP 0x8163
#define GL_OCCLUSION_TEST_HP 0x8165
#define GL_OCCLUSION_TEST_RESULT_HP 0x8166
#define GL_TEXTURE_LIGHTING_MODE_HP 0x8167
#define GL_TEXTURE_POST_SPECULAR_HP 0x8168
#define GL_TEXTURE_PRE_SPECULAR_HP 0x8169
#define GL_CULL_VERTEX_IBM 103050
#define GL_RASTER_POSITION_UNCLIPPED_IBM 0x19262
#define GL_ALL_STATIC_DATA_IBM 103060
#define GL_STATIC_VERTEX_ARRAY_IBM 103061
#define GL_MIRRORED_REPEAT_IBM 0x8370
#define GL_VERTEX_ARRAY_LIST_IBM 103070
#define GL_NORMAL_ARRAY_LIST_IBM 103071
#define GL_COLOR_ARRAY_LIST_IBM 103072
#define GL_INDEX_ARRAY_LIST_IBM 103073
#define GL_TEXTURE_COORD_ARRAY_LIST_IBM 103074
#define GL_EDGE_FLAG_ARRAY_LIST_IBM 103075
#define GL_FOG_COORDINATE_ARRAY_LIST_IBM 103076
#define GL_SECONDARY_COLOR_ARRAY_LIST_IBM 103077
#define GL_VERTEX_ARRAY_LIST_STRIDE_IBM 103080
#define GL_NORMAL_ARRAY_LIST_STRIDE_IBM 103081
#define GL_COLOR_ARRAY_LIST_STRIDE_IBM 103082
#define GL_INDEX_ARRAY_LIST_STRIDE_IBM 103083
#define GL_TEXTURE_COORD_ARRAY_LIST_STRIDE_IBM 103084
#define GL_EDGE_FLAG_ARRAY_LIST_STRIDE_IBM 103085
#define GL_FOG_COORDINATE_ARRAY_LIST_STRIDE_IBM 103086
```

```
#define GL_SECONDARY_COLOR_ARRAY_LIST_STRIDE_IBM 103087
#define GL_RED_MIN_CLAMP_INGR 0x8560
#define GL_GREEN_MIN_CLAMP_INGR 0x8561
#define GL_BLUE_MIN_CLAMP_INGR 0x8562
#define GL_ALPHA_MIN_CLAMP_INGR 0x8563
#define GL_RED_MAX_CLAMP_INGR 0x8564
#define GL_GREEN_MAX_CLAMP_INGR 0x8565
#define GL_BLUE_MAX_CLAMP_INGR 0x8566
#define GL_ALPHA_MAX_CLAMP_INGR 0x8567
#define GL_INTERLACE_READ_INGR 0x8568
#define GL_BLACKHOLE_RENDER_INTEL 0x83FC
#define GL_CONSERVATIVE_RASTERIZATION_INTEL 0x83FE
#define GL_TEXTURE_MEMORY_LAYOUT_INTEL 0x83FF
#define GL_LAYOUT_DEFAULT_INTEL 0
#define GL_LAYOUT_LINEAR_INTEL 1
#define GL_LAYOUT_LINEAR_CPU_CACHED_INTEL 2
#define GL_PARALLEL_ARRAYS_INTEL 0x83F4
#define GL_VERTEX_ARRAY_PARALLEL_POINTERS_INTEL 0x83F5
#define GL_NORMAL_ARRAY_PARALLEL_POINTERS_INTEL 0x83F6
#define GL_COLOR_ARRAY_PARALLEL_POINTERS_INTEL 0x83F7
#define GL_TEXTURE_COORD_ARRAY_PARALLEL_POINTERS_INTEL 0x83F8
#define GL_PERFQUERY_SINGLE_CONTEXT_INTEL 0x00000000
#define GL_PERFQUERY_GLOBAL_CONTEXT_INTEL 0x00000001
#define GL_PERFQUERY_WAIT_INTEL 0x83FB
#define GL_PERFQUERY_FLUSH_INTEL 0x83FA
#define GL_PERFQUERY_DONOT_FLUSH_INTEL 0x83F9
#define GL_PERFQUERY_COUNTER_EVENT_INTEL 0x94F0
#define GL_PERFQUERY_COUNTER_DURATION_NORM_INTEL 0x94F1
#define GL_PERFQUERY_COUNTER_DURATION_RAW_INTEL 0x94F2
#define GL_PERFQUERY_COUNTER_THROUGHPUT_INTEL 0x94F3
#define GL_PERFQUERY_COUNTER_RAW_INTEL 0x94F4
#define GL_PERFQUERY_COUNTER_TIMESTAMP_INTEL 0x94F5
#define GL_PERFQUERY_COUNTER_DATA_UINT32_INTEL 0x94F8
#define GL_PERFQUERY_COUNTER_DATA_UINT64_INTEL 0x94F9
#define GL_PERFQUERY_COUNTER_DATA_FLOAT_INTEL 0x94FA
#define GL_PERFQUERY_COUNTER_DATA_DOUBLE_INTEL 0x94FB
#define GL_PERFQUERY_COUNTER_DATA_BOOL32_INTEL 0x94FC
#define GL_PERFQUERY_QUERY_NAME_LENGTH_MAX_INTEL 0x94FD
#define GL_PERFQUERY_COUNTER_NAME_LENGTH_MAX_INTEL 0x94FE
#define GL_PERFQUERY_COUNTER_DESC_LENGTH_MAX_INTEL 0x94FF
#define GL_PERFQUERY_GPA_EXTENDED_COUNTERS_INTEL 0x9500
#define GL_MULTIPLY_KHR 0x9294
#define GL_SCREEN_KHR 0x9295
#define GL_OVERLAY_KHR 0x9296
#define GL_DARKEN_KHR 0x9297
#define GL_LIGHTEN_KHR 0x9298
#define GL_COLORDODGE_KHR 0x9299
#define GL_COLORBURN_KHR 0x929A
#define GL_HARDLIGHT_KHR 0x929B
#define GL_SOFTLIGHT_KHR 0x929C
#define GL_DIFFERENCE_KHR 0x929E
#define GL_EXCLUSION_KHR 0x92A0
#define GL_HSL_HUE_KHR 0x92AD
#define GL_HSL_SATURATION_KHR 0x92AE
#define GL_HSL_COLOR_KHR 0x92AF
#define GL_HSL_LUMINOSITY_KHR 0x92B0
#define GL_BLEND_ADVANCED_COHERENT_KHR 0x9285
#define GL_CONTEXT_RELEASE_BEHAVIOR 0x82FB
#define GL_CONTEXT_RELEASE_BEHAVIOR_FLUSH 0x82FC
#define GL_CONTEXT_RELEASE_BEHAVIOR_KHR 0x82FB
#define GL_CONTEXT_RELEASE_BEHAVIOR_FLUSH_KHR 0x82FC
#define GL_DEBUG_OUTPUT_SYNCHRONOUS 0x8242
#define GL_DEBUG_NEXT_LOGGED_MESSAGE_LENGTH 0x8243
#define GL_DEBUG_CALLBACK_FUNCTION 0x8244
#define GL_DEBUG_CALLBACK_USER_PARAM 0x8245
#define GL_DEBUG_SOURCE_API 0x8246
#define GL_DEBUG_SOURCE_WINDOW_SYSTEM 0x8247
#define GL_DEBUG_SOURCE_SHADER_COMPILER 0x8248
#define GL_DEBUG_SOURCE_THIRD_PARTY 0x8249
#define GL_DEBUG_SOURCE_APPLICATION 0x824A
#define GL_DEBUG_SOURCE_OTHER 0x824B
#define GL_DEBUG_TYPE_ERROR 0x824C
#define GL_DEBUG_TYPE_DEPRECATED_BEHAVIOR 0x824D
#define GL_DEBUG_TYPE_UNDEFINED_BEHAVIOR 0x824E
#define GL_DEBUG_TYPE_PORTABILITY 0x824F
#define GL_DEBUG_TYPE_PERFORMANCE 0x8250
#define GL_DEBUG_TYPE_OTHER 0x8251
#define GL_DEBUG_TYPE_MARKER 0x8268
#define GL_DEBUG_TYPE_PUSH_GROUP 0x8269
#define GL_DEBUG_TYPE_POP_GROUP 0x826A
#define GL_DEBUG_SEVERITY_NOTIFICATION 0x826B
#define GL_MAX_DEBUG_GROUP_STACK_DEPTH 0x826C
#define GL_DEBUG_GROUP_STACK_DEPTH 0x826D
#define GL_BUFFER 0x82E0
#define GL_SHADER 0x82E1
#define GL_PROGRAM 0x82E2
#define GL_VERTEX_ARRAY 0x8074
```

```
#define GL_QUERY 0x82E3
#define GL_PROGRAM_PIPELINE 0x82E4
#define GL_MAX_LABEL_LENGTH 0x82E8
#define GL_MAX_DEBUG_MESSAGE_LENGTH 0x9143
#define GL_MAX_DEBUG_LOGGED_MESSAGES 0x9144
#define GL_DEBUG_LOGGED_MESSAGES 0x9145
#define GL_DEBUG_SEVERITY_HIGH 0x9146
#define GL_DEBUG_SEVERITY_MEDIUM 0x9147
#define GL_DEBUG_SEVERITY_LOW 0x9148
#define GL_DEBUG_OUTPUT 0x92E0
#define GL_CONTEXT_FLAG_DEBUG_BIT 0x00000002
#define GL_STACK_OVERFLOW 0x0503
#define GL_STACK_UNDERFLOW 0x0504
#define GL_DEBUG_OUTPUT_SYNCHRONOUS_KHR 0x8242
#define GL_DEBUG_NEXT_LOGGED_MESSAGE_LENGTH_KHR 0x8243
#define GL_DEBUG_CALLBACK_FUNCTION_KHR 0x8244
#define GL_DEBUG_CALLBACK_USER_PARAM_KHR 0x8245
#define GL_DEBUG_SOURCE_API_KHR 0x8246
#define GL_DEBUG_SOURCE_WINDOW_SYSTEM_KHR 0x8247
#define GL_DEBUG_SOURCE_SHADER_COMPILER_KHR 0x8248
#define GL_DEBUG_SOURCE_THIRD_PARTY_KHR 0x8249
#define GL_DEBUG_SOURCE_APPLICATION_KHR 0x824A
#define GL_DEBUG_SOURCE_OTHER_KHR 0x824B
#define GL_DEBUG_TYPE_ERROR_KHR 0x824C
#define GL_DEBUG_TYPE_DEPRECATED_BEHAVIOR_KHR 0x824D
#define GL_DEBUG_TYPE_UNDEFINED_BEHAVIOR_KHR 0x824E
#define GL_DEBUG_TYPE_PORTABILITY_KHR 0x824F
#define GL_DEBUG_TYPE_PERFORMANCE_KHR 0x8250
#define GL_DEBUG_TYPE_OTHER_KHR 0x8251
#define GL_DEBUG_TYPE_MARKER_KHR 0x8268
#define GL_DEBUG_TYPE_PUSH_GROUP_KHR 0x8269
#define GL_DEBUG_TYPE_POP_GROUP_KHR 0x826A
#define GL_DEBUG_SEVERITY_NOTIFICATION_KHR 0x826B
#define GL_MAX_DEBUG_GROUP_STACK_DEPTH_KHR 0x826C
#define GL_DEBUG_GROUP_STACK_DEPTH_KHR 0x826D
#define GL_BUFFER_KHR 0x82E0
#define GL_SHADER_KHR 0x82E1
#define GL_PROGRAM_KHR 0x82E2
#define GL_VERTEX_ARRAY_KHR 0x8074
#define GL_QUERY_KHR 0x82E3
#define GL_PROGRAM_PIPELINE_KHR 0x82E4
#define GL_SAMPLER_KHR 0x82E6
#define GL_MAX_LABEL_LENGTH_KHR 0x82E8
#define GL_MAX_DEBUG_MESSAGE_LENGTH_KHR 0x9143
#define GL_MAX_DEBUG_LOGGED_MESSAGES_KHR 0x9144
#define GL_DEBUG_LOGGED_MESSAGES_KHR 0x9145
#define GL_DEBUG_SEVERITY_HIGH_KHR 0x9146
#define GL_DEBUG_SEVERITY_MEDIUM_KHR 0x9147
#define GL_DEBUG_SEVERITY_LOW_KHR 0x9148
#define GL_DEBUG_OUTPUT_KHR 0x92E0
#define GL_CONTEXT_FLAG_DEBUG_BIT_KHR 0x00000002
#define GL_STACK_OVERFLOW_KHR 0x0503
#define GL_STACK_UNDERFLOW_KHR 0x0504
#define GL_DISPLAY_LIST 0x82E7
#define GL_CONTEXT_FLAG_NO_ERROR_BIT_KHR 0x00000008
#define GL_MAX_SHADER_COMPILER_THREADS_KHR 0x91B0
#define GL_COMPLETION_STATUS_KHR 0x91B1
#define GL_CONTEXT_ROBUST_ACCESS 0x90F3
#define GL_LOSE_CONTEXT_ON_RESET 0x8252
#define GL_GUILTY_CONTEXT_RESET 0x8253
#define GL_INNOCENT_CONTEXT_RESET 0x8254
#define GL_UNKNOWN_CONTEXT_RESET 0x8255
#define GL_RESET_NOTIFICATION_STRATEGY 0x8256
#define GL_NO_RESET_NOTIFICATION 0x8261
#define GL_CONTEXT_LOST 0x0507
#define GL_CONTEXT_ROBUST_ACCESS_KHR 0x90F3
#define GL_LOSE_CONTEXT_ON_RESET_KHR 0x8252
#define GL_GUILTY_CONTEXT_RESET_KHR 0x8253
#define GL_INNOCENT_CONTEXT_RESET_KHR 0x8254
#define GL_UNKNOWN_CONTEXT_RESET_KHR 0x8255
#define GL_RESET_NOTIFICATION_STRATEGY_KHR 0x8256
#define GL_NO_RESET_NOTIFICATION_KHR 0x8261
#define GL_CONTEXT_LOST_KHR 0x0507
#define GL_SUBGROUP_SIZE_KHR 0x9532
#define GL_SUBGROUP_SUPPORTED_STAGES_KHR 0x9533
#define GL_SUBGROUP_SUPPORTED_FEATURES_KHR 0x9534
#define GL_SUBGROUP_QUAD_ALL_STAGES_KHR 0x9535
#define GL_SUBGROUP_FEATURE_BASIC_BIT_KHR 0x00000001
#define GL_SUBGROUP_FEATURE_VOTE_BIT_KHR 0x00000002
#define GL_SUBGROUP_FEATURE_ARITHMETIC_BIT_KHR 0x00000004
#define GL_SUBGROUP_FEATURE_BALLOT_BIT_KHR 0x00000008
#define GL_SUBGROUP_FEATURE_SHUFFLE_BIT_KHR 0x00000010
#define GL_SUBGROUP_FEATURE_SHUFFLE_RELATIVE_BIT_KHR 0x00000020
#define GL_SUBGROUP_FEATURE_CLUSTERED_BIT_KHR 0x00000040
#define GL_SUBGROUP_FEATURE_QUAD_BIT_KHR 0x00000080
#define GL_COMPRESSED_RGBA_ASTC_4x4_KHR 0x93B0
#define GL_COMPRESSED_RGBA_ASTC_5x4_KHR 0x93B1
```

```
#define GL_COMPRESSED_RGBA_ASTC_5x5_KHR 0x93B2
#define GL_COMPRESSED_RGBA_ASTC_6x5_KHR 0x93B3
#define GL_COMPRESSED_RGBA_ASTC_6x6_KHR 0x93B4
#define GL_COMPRESSED_RGBA_ASTC_8x5_KHR 0x93B5
#define GL_COMPRESSED_RGBA_ASTC_8x6_KHR 0x93B6
#define GL_COMPRESSED_RGBA_ASTC_8x8_KHR 0x93B7
#define GL_COMPRESSED_RGBA_ASTC_10x5_KHR 0x93B8
#define GL_COMPRESSED_RGBA_ASTC_10x6_KHR 0x93B9
#define GL_COMPRESSED_RGBA_ASTC_10x8_KHR 0x93BA
#define GL_COMPRESSED_RGBA_ASTC_10x10_KHR 0x93BB
#define GL_COMPRESSED_RGBA_ASTC_12x10_KHR 0x93BC
#define GL_COMPRESSED_RGBA_ASTC_12x12_KHR 0x93BD
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_4x4_KHR 0x93D0
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_5x4_KHR 0x93D1
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_5x5_KHR 0x93D2
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_6x5_KHR 0x93D3
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_6x6_KHR 0x93D4
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_8x5_KHR 0x93D5
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_8x6_KHR 0x93D6
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_8x8_KHR 0x93D7
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x5_KHR 0x93D8
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x6_KHR 0x93D9
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x8_KHR 0x93DA
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_10x10_KHR 0x93DB
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_12x10_KHR 0x93DC
#define GL_COMPRESSED_SRGB8_ALPHA8_ASTC_12x12_KHR 0x93DD
#define GL_TEXTURE_1D_STACK_MESAX 0x8759
#define GL_TEXTURE_2D_STACK_MESAX 0x875A
#define GL_PROXY_TEXTURE_1D_STACK_MESAX 0x875B
#define GL_PROXY_TEXTURE_2D_STACK_MESAX 0x875C
#define GL_TEXTURE_1D_STACK_BINDING_MESAX 0x875D
#define GL_TEXTURE_2D_STACK_BINDING_MESAX 0x875E
#define GL_FRAMEBUFFER_FLIP_X_MESA 0x8BBC
#define GL_FRAMEBUFFER_FLIP_Y_MESA 0x8BBB
#define GL_FRAMEBUFFER_SWAP_XY_MESA 0x8BBD
#define GL_PACK_INVERT_MESA 0x8758
#define GL_PROGRAM_BINARY_FORMAT_MESA 0x875F
#define GL_TILE_RASTER_ORDER_FIXED_MESA 0x8BB8
#define GL_TILE_RASTER_ORDER_INCREASING_X_MESA 0x8BB9
#define GL_TILE_RASTER_ORDER_INCREASING_Y_MESA 0x8BBA
#define GL_UNSIGNED_SHORT_8_8_MESA 0x85BA
#define GL_UNSIGNED_SHORT_8_8_REV_MESA 0x85BB
#define GL_YCBCR_MESA 0x8757
#define GL_GPU_MEMORY_INFO_DEDICATED_VIDMEM_NVX 0x9047
#define GL_GPU_MEMORY_INFO_TOTAL_AVAILABLE_MEMORY_NVX 0x9048
#define GL_GPU_MEMORY_INFO_CURRENT_AVAILABLE_VIDMEM_NVX 0x9049
#define GL_GPU_MEMORY_INFO_EVICTION_COUNT_NVX 0x904A
#define GL_GPU_MEMORY_INFO_EVICTED_MEMORY_NVX 0x904B
#define GL_UPLOAD_GPU_MASK_NVX 0x954A
#define GL_LGPU_SEPARATE_STORAGE_BIT_NVX 0x0800
#define GL_MAX_LGPU_GPUS_NVX 0x92BA
#define GL_ALPHA_TO_COVERAGE_DITHER_DEFAULT_NV 0x934D
#define GL_ALPHA_TO_COVERAGE_DITHER_ENABLE_NV 0x934E
#define GL_ALPHA_TO_COVERAGE_DITHER_DISABLE_NV 0x934F
#define GL_ALPHA_TO_COVERAGE_DITHER_MODE_NV 0x92BF
#define GL_BLEND_OVERLAP_NV 0x9281
#define GL_BLEND_PREMULTIPLIED_SRC_NV 0x9280
#define GL_BLUE_NV 0x1905
#define GL_COLORBURN_NV 0x929A
#define GL_COLORDODGE_NV 0x9299
#define GL_CONJOINT_NV 0x9284
#define GL_CONTRAST_NV 0x92A1
#define GL_DARKEN_NV 0x9297
#define GL_DIFFERENCE_NV 0x929E
#define GL_DISJOINT_NV 0x9283
#define GL_DST_ATOP_NV 0x928F
#define GL_DST_IN_NV 0x928B
#define GL_DST_NV 0x9287
#define GL_DST_OUT_NV 0x928D
#define GL_DST_OVER_NV 0x9289
#define GL_EXCLUSION_NV 0x92A0
#define GL_GREEN_NV 0x1904
#define GL_HARDLIGHT_NV 0x929B
#define GL_HARDMIX_NV 0x92A9
#define GL_HSL_COLOR_NV 0x92AF
#define GL_HSL_HUE_NV 0x92AD
#define GL_HSL_LUMINOSITY_NV 0x92B0
#define GL_HSL_SATURATION_NV 0x92AE
#define GL_INVERT_OVG_NV 0x92B4
#define GL_INVERT_RGB_NV 0x92A3
#define GL_LIGHTEN_NV 0x9298
#define GL_LINEARBURN_NV 0x92A5
#define GL_LINEARDODGE_NV 0x92A4
#define GL_LINEARLIGHT_NV 0x92A7
#define GL_MINUS_CLAMPED_NV 0x92B3
#define GL_MINUS_NV 0x929F
#define GL_MULTIPLY_NV 0x9294
```

```
#define GL_OVERLAY_NV 0x9296
#define GL_PINLIGHT_NV 0x92A8
#define GL_PLUS_CLAMPED_ALPHA_NV 0x92B2
#define GL_PLUS_CLAMPED_NV 0x92B1
#define GL_PLUS_DARKER_NV 0x9292
#define GL_PLUS_NV 0x9291
#define GL_RED_NV 0x1903
#define GL_SCREEN_NV 0x9295
#define GL_SOFTLIGHT_NV 0x929C
#define GL_SRC_ATOP_NV 0x928E
#define GL_SRC_IN_NV 0x928A
#define GL_SRC_NV 0x9286
#define GL_SRC_OUT_NV 0x928C
#define GL_SRC_OVER_NV 0x9288
#define GL_UNCORRELATED_NV 0x9282
#define GL_VIVIDLIGHT_NV 0x92A6
#define GL_XOR_NV 0x1506
#define GL_BLEND_ADVANCED_COHERENT_NV 0x9285
#define GL_VIEWPORT_POSITION_W_SCALE_NV 0x937C
#define GL_VIEWPORT_POSITION_W_SCALE_X_COEFF_NV 0x937D
#define GL_VIEWPORT_POSITION_W_SCALE_Y_COEFF_NV 0x937E
#define GL_TERMINATE_SEQUENCE_COMMAND_NV 0x0000
#define GL_NOP_COMMAND_NV 0x0001
#define GL_DRAW_ELEMENTS_COMMAND_NV 0x0002
#define GL_DRAW_ARRAYS_COMMAND_NV 0x0003
#define GL_DRAW_ELEMENTS_STRIP_COMMAND_NV 0x0004
#define GL_DRAW_ARRAYS_STRIP_COMMAND_NV 0x0005
#define GL_DRAW_ELEMENTS_INSTANCED_COMMAND_NV 0x0006
#define GL_DRAW_ARRAYS_INSTANCED_COMMAND_NV 0x0007
#define GL_ELEMENT_ADDRESS_COMMAND_NV 0x0008
#define GL_ATTRIBUTE_ADDRESS_COMMAND_NV 0x0009
#define GL_UNIFORM_ADDRESS_COMMAND_NV 0x000A
#define GL_BLEND_COLOR_COMMAND_NV 0x000B
#define GL_STENCIL_REF_COMMAND_NV 0x000C
#define GL_LINE_WIDTH_COMMAND_NV 0x000D
#define GL_POLYGON_OFFSET_COMMAND_NV 0x000E
#define GL_ALPHA_REF_COMMAND_NV 0x000F
#define GL_VIEWPORT_COMMAND_NV 0x0010
#define GL_SCISSOR_COMMAND_NV 0x0011
#define GL_FRONT_FACE_COMMAND_NV 0x0012
#define GL_COMPUTE_PROGRAM_NV 0x90FB
#define GL_COMPUTE_PROGRAM_PARAMETER_BUFFER_NV 0x90FC
#define GL_QUERY_WAIT_NV 0x8E13
#define GL_QUERY_NO_WAIT_NV 0x8E14
#define GL_QUERY_BY_REGION_WAIT_NV 0x8E15
#define GL_QUERY_BY_REGION_NO_WAIT_NV 0x8E16
#define GL_CONSERVATIVE_RASTERIZATION_NV 0x9346
#define GL_SUBPIXEL_PRECISION_BIAS_X_BITS_NV 0x9347
#define GL_SUBPIXEL_PRECISION_BIAS_Y_BITS_NV 0x9348
#define GL_MAX_SUBPIXEL_PRECISION_BIAS_BITS_NV 0x9349
#define GL_CONSERVATIVE_RASTER_DILATE_NV 0x9379
#define GL_CONSERVATIVE_RASTER_DILATE_RANGE_NV 0x937A
#define GL_CONSERVATIVE_RASTER_DILATE_GRANULARITY_NV 0x937B
#define GL_CONSERVATIVE_RASTER_MODE_PRE_SNAP_NV 0x9550
#define GL_CONSERVATIVE_RASTER_MODE_NV 0x954D
#define GL_CONSERVATIVE_RASTER_MODE_POST_SNAP_NV 0x954E
#define GL_CONSERVATIVE_RASTER_MODE_PRE_SNAP_TRIANGLES_NV 0x954F
#define GL_DEPTH_STENCIL_TO_RGBA_NV 0x886E
#define GL_DEPTH_STENCIL_TO_BGRA_NV 0x886F
#define GL_MAX_DEEP_3D_TEXTURE_WIDTH_HEIGHT_NV 0x90D0
#define GL_MAX_DEEP_3D_TEXTURE_DEPTH_NV 0x90D1
#define GL_DEPTH_COMPONENT32F_NV 0x8DAB
#define GL_DEPTH32F_STENCIL8_NV 0x8DAC
#define GL_FLOAT_32_UNSIGNED_INT_24_8_REV_NV 0x8DAD
#define GL_DEPTH_BUFFER_FLOAT_MODE_NV 0x8DAF
#define GL_DEPTH_CLAMP_NV 0x864F
#define GL_EVAL_2D_NV 0x86C0
#define GL_EVAL_TRIANGULAR_2D_NV 0x86C1
#define GL_MAP_TESSELLATION_NV 0x86C2
#define GL_MAP_ATTRIB_U_ORDER_NV 0x86C3
#define GL_MAP_ATTRIB_V_ORDER_NV 0x86C4
#define GL_EVAL_FRACTIONAL_TESSELLATION_NV 0x86C5
#define GL_EVAL_VERTEX_ATTRIB0_NV 0x86C6
#define GL_EVAL_VERTEX_ATTRIB1_NV 0x86C7
#define GL_EVAL_VERTEX_ATTRIB2_NV 0x86C8
#define GL_EVAL_VERTEX_ATTRIB3_NV 0x86C9
#define GL_EVAL_VERTEX_ATTRIB4_NV 0x86CA
#define GL_EVAL_VERTEX_ATTRIB5_NV 0x86CB
#define GL_EVAL_VERTEX_ATTRIB6_NV 0x86CC
#define GL_EVAL_VERTEX_ATTRIB7_NV 0x86CD
#define GL_EVAL_VERTEX_ATTRIB8_NV 0x86CE
#define GL_EVAL_VERTEX_ATTRIB9_NV 0x86CF
#define GL_EVAL_VERTEX_ATTRIB10_NV 0x86D0
#define GL_EVAL_VERTEX_ATTRIB11_NV 0x86D1
#define GL_EVAL_VERTEX_ATTRIB12_NV 0x86D2
#define GL_EVAL_VERTEX_ATTRIB13_NV 0x86D3
#define GL_EVAL_VERTEX_ATTRIB14_NV 0x86D4
```

```
#define GL_EVAL_VERTEX_ATTRIB15_NV 0x86D5
#define GL_MAX_MAP_TESSELLATION_NV 0x86D6
#define GL_MAX_RATIONAL_EVAL_ORDER_NV 0x86D7
#define GL_SAMPLE_POSITION_NV 0x8E50
#define GL_SAMPLE_MASK_NV 0x8E51
#define GL_SAMPLE_MASK_VALUE_NV 0x8E52
#define GL_TEXTURE_BINDING_RENDERBUFFER_NV 0x8E53
#define GL_TEXTURE_RENDERBUFFER_DATA_STORE_BINDING_NV 0x8E54
#define GL_TEXTURE_RENDERBUFFER_NV 0x8E55
#define GL_SAMPLER_RENDERBUFFER_NV 0x8E56
#define GL_INT_SAMPLER_RENDERBUFFER_NV 0x8E57
#define GL_UNSIGNED_INT_SAMPLER_RENDERBUFFER_NV 0x8E58
#define GL_MAX_SAMPLE_MASK_WORDS_NV 0x8E59
#define GL_ALL_COMPLETED_NV 0x84F2
#define GL_FENCE_STATUS_NV 0x84F3
#define GL_FENCE_CONDITION_NV 0x84F4
#define GL_FILL_RECTANGLE_NV 0x933C
#define GL_FLOAT_R_NV 0x8880
#define GL_FLOAT_RG_NV 0x8881
#define GL_FLOAT_RGB_NV 0x8882
#define GL_FLOAT_RGBA_NV 0x8883
#define GL_FLOAT_R16_NV 0x8884
#define GL_FLOAT_R32_NV 0x8885
#define GL_FLOAT_RG16_NV 0x8886
#define GL_FLOAT_RG32_NV 0x8887
#define GL_FLOAT_RGB16_NV 0x8888
#define GL_FLOAT_RGB32_NV 0x8889
#define GL_FLOAT_RGBA16_NV 0x888A
#define GL_FLOAT_RGBA32_NV 0x888B
#define GL_TEXTURE_FLOAT_COMPONENTS_NV 0x888C
#define GL_FLOAT_CLEAR_COLOR_VALUE_NV 0x888D
#define GL_FLOAT_RGBA_MODE_NV 0x888E
#define GL_FOG_DISTANCE_MODE_NV 0x855A
#define GL_EYE_RADIAL_NV 0x855B
#define GL_EYE_PLANE_ABSOLUTE_NV 0x855C
#define GL_EYE_PLANE 0x2502
#define GL_FRAGMENT_COVERAGE_TO_COLOR_NV 0x92DD
#define GL_FRAGMENT_COVERAGE_COLOR_NV 0x92DE
#define GL_MAX_FRAGMENT_PROGRAM_LOCAL_PARAMETERS_NV 0x8868
#define GL_FRAGMENT_PROGRAM_NV 0x8870
#define GL_MAX_TEXTURE_COORDS_NV 0x8871
#define GL_MAX_TEXTURE_IMAGE_UNITS_NV 0x8872
#define GL_FRAGMENT_PROGRAM_BINDING_NV 0x8873
#define GL_PROGRAM_ERROR_STRING_NV 0x8874
#define GL_MAX_PROGRAM_EXEC_INSTRUCTIONS_NV 0x88F4
#define GL_MAX_PROGRAM_CALL_DEPTH_NV 0x88F5
#define GL_MAX_PROGRAM_IF_DEPTH_NV 0x88F6
#define GL_MAX_PROGRAM_LOOP_DEPTH_NV 0x88F7
#define GL_MAX_PROGRAM_LOOP_COUNT_NV 0x88F8
#define GL_COVERAGE_MODULATION_TABLE_NV 0x9331
#define GL_COLOR_SAMPLES_NV 0x8E20
#define GL_DEPTH_SAMPLES_NV 0x932D
#define GL_STENCIL_SAMPLES_NV 0x932E
#define GL_MIXED_DEPTH_SAMPLES_SUPPORTED_NV 0x932F
#define GL_MIXED_STENCIL_SAMPLES_SUPPORTED_NV 0x9330
#define GL_COVERAGE_MODULATION_NV 0x9332
#define GL_COVERAGE_MODULATION_TABLE_SIZE_NV 0x9333
#define GL_RENDERBUFFER_COVERAGE_SAMPLES_NV 0x8CAB
#define GL_RENDERBUFFER_COLOR_SAMPLES_NV 0x8E10
#define GL_MAX_MULTISAMPLE_COVERAGE_MODES_NV 0x8E11
#define GL_MULTISAMPLE_COVERAGE_MODES_NV 0x8E12
#define GL_GEOMETRY_PROGRAM_NV 0x8C26
#define GL_MAX_PROGRAM_OUTPUT_VERTICES_NV 0x8C27
#define GL_MAX_PROGRAM_TOTAL_OUTPUT_COMPONENTS_NV 0x8C28
#define GL_PER_GPU_STORAGE_BIT_NV 0x0800
#define GL_MULTICAST_GPUS_NV 0x92BA
#define GL_RENDER_GPU_MASK_NV 0x9558
#define GL_PER_GPU_STORAGE_NV 0x9548
#define GL_MULTICAST_PROGRAMMABLE_SAMPLE_LOCATION_NV 0x9549
#define GL_MIN_PROGRAM_TEXEL_OFFSET_NV 0x8904
#define GL_MAX_PROGRAM_TEXEL_OFFSET_NV 0x8905
#define GL_PROGRAM_ATTRIB_COMPONENTS_NV 0x8906
#define GL_PROGRAM_RESULT_COMPONENTS_NV 0x8907
#define GL_MAX_PROGRAM_ATTRIB_COMPONENTS_NV 0x8908
#define GL_MAX_PROGRAM_RESULT_COMPONENTS_NV 0x8909
#define GL_MAX_PROGRAM_GENERIC_ATTRIBS_NV 0x8DA5
#define GL_MAX_PROGRAM_GENERIC_RESULTS_NV 0x8DA6
#define GL_MAX_GEOMETRY_PROGRAM_INVOCATIONS_NV 0x8E5A
#define GL_MIN_FRAGMENT_INTERPOLATION_OFFSET_NV 0x8E5B
#define GL_MAX_FRAGMENT_INTERPOLATION_OFFSET_NV 0x8E5C
#define GL_FRAGMENT_PROGRAM_INTERPOLATION_OFFSET_BITS_NV 0x8E5D
#define GL_MIN_PROGRAM_TEXTURE_GATHER_OFFSET_NV 0x8E5E
#define GL_MAX_PROGRAM_TEXTURE_GATHER_OFFSET_NV 0x8E5F
#define GL_MAX_PROGRAM_SUBROUTINE_PARAMETERS_NV 0x8F44
#define GL_MAX_PROGRAM_SUBROUTINE_NUM_NV 0x8F45
#define GL_HALF_FLOAT_NV 0x140B
#define GL_MULTISAMPLES_NV 0x9371
```

```
#define GL_SUPERSAMPLE_SCALE_X_NV 0x9372
#define GL_SUPERSAMPLE_SCALE_Y_NV 0x9373
#define GL_CONFORMANT_NV 0x9374
#define GL_MAX_SHININESS_NV 0x8504
#define GL_MAX_SPOT_EXPONENT_NV 0x8505
#define GL_ATTACHED_MEMORY_OBJECT_NV 0x95A4
#define GL_ATTACHED_MEMORY_OFFSET_NV 0x95A5
#define GL_MEMORY_ATTACHABLE_ALIGNMENT_NV 0x95A6
#define GL_MEMORY_ATTACHABLE_SIZE_NV 0x95A7
#define GL_MEMORY_ATTACHABLE_NV 0x95A8
#define GL_DETACHED_MEMORY_INCARNATION_NV 0x95A9
#define GL_DETACHED_TEXTURES_NV 0x95AA
#define GL_DETACHED_BUFFERS_NV 0x95AB
#define GL_MAX_DETACHED_TEXTURES_NV 0x95AC
#define GL_MAX_DETACHED_BUFFERS_NV 0x95AD
#define GL_MESH_SHADER_NV 0x9559
#define GL_TASK_SHADER_NV 0x955A
#define GL_MAX_MESH_UNIFORM_BLOCKS_NV 0x8E60
#define GL_MAX_MESH_TEXTURE_IMAGE_UNITS_NV 0x8E61
#define GL_MAX_MESH_IMAGE_UNIFORMS_NV 0x8E62
#define GL_MAX_MESH_UNIFORM_COMPONENTS_NV 0x8E63
#define GL_MAX_MESH_ATOMIC_COUNTER_BUFFERS_NV 0x8E64
#define GL_MAX_MESH_ATOMIC_COUNTERS_NV 0x8E65
#define GL_MAX_MESH_SHADER_STORAGE_BLOCKS_NV 0x8E66
#define GL_MAX_COMBINED_MESH_UNIFORM_COMPONENTS_NV 0x8E67
#define GL_MAX_TASK_UNIFORM_BLOCKS_NV 0x8E68
#define GL_MAX_TASK_TEXTURE_IMAGE_UNITS_NV 0x8E69
#define GL_MAX_TASK_IMAGE_UNIFORMS_NV 0x8E6A
#define GL_MAX_TASK_UNIFORM_COMPONENTS_NV 0x8E6B
#define GL_MAX_TASK_ATOMIC_COUNTER_BUFFERS_NV 0x8E6C
#define GL_MAX_TASK_ATOMIC_COUNTERS_NV 0x8E6D
#define GL_MAX_TASK_SHADER_STORAGE_BLOCKS_NV 0x8E6E
#define GL_MAX_COMBINED_TASK_UNIFORM_COMPONENTS_NV 0x8E6F
#define GL_MAX_MESH_WORK_GROUP_INVOCATIONS_NV 0x95A2
#define GL_MAX_TASK_WORK_GROUP_INVOCATIONS_NV 0x95A3
#define GL_MAX_MESH_TOTAL_MEMORY_SIZE_NV 0x9536
#define GL_MAX_TASK_TOTAL_MEMORY_SIZE_NV 0x9537
#define GL_MAX_MESH_OUTPUT_VERTICES_NV 0x9538
#define GL_MAX_MESH_OUTPUT_PRIMITIVES_NV 0x9539
#define GL_MAX_TASK_OUTPUT_COUNT_NV 0x953A
#define GL_MAX_DRAW_MESH_TASKS_COUNT_NV 0x953D
#define GL_MAX_MESH_VIEWS_NV 0x9557
#define GL_MESH_OUTPUT_PER_VERTEX_GRANULARITY_NV 0x92DF
#define GL_MESH_OUTPUT_PER_PRIMITIVE_GRANULARITY_NV 0x9543
#define GL_MAX_MESH_WORK_GROUP_SIZE_NV 0x953B
#define GL_MAX_TASK_WORK_GROUP_SIZE_NV 0x953C
#define GL_MESH_WORK_GROUP_SIZE_NV 0x953E
#define GL_TASK_WORK_GROUP_SIZE_NV 0x953F
#define GL_MESH_VERTICES_OUT_NV 0x9579
#define GL_MESH_PRIMITIVES_OUT_NV 0x957A
#define GL_MESH_OUTPUT_TYPE_NV 0x957B
#define GL_UNIFORM_BLOCK_REFERENCED_BY_MESH_SHADER_NV 0x959C
#define GL_UNIFORM_BLOCK_REFERENCED_BY_TASK_SHADER_NV 0x959D
#define GL_REFERENCED_BY_MESH_SHADER_NV 0x95A0
#define GL_REFERENCED_BY_TASK_SHADER_NV 0x95A1
#define GL_MESH_SHADER_BIT_NV 0x00000040
#define GL_TASK_SHADER_BIT_NV 0x00000080
#define GL_MESH_SUBROUTINE_NV 0x957C
#define GL_TASK_SUBROUTINE_NV 0x957D
#define GL_MESH_SUBROUTINE_UNIFORM_NV 0x957E
#define GL_TASK_SUBROUTINE_UNIFORM_NV 0x957F
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_MESH_SHADER_NV 0x959E
#define GL_ATOMIC_COUNTER_BUFFER_REFERENCED_BY_TASK_SHADER_NV 0x959F
#define GL_MULTISAMPLE_FILTER_HINT_NV 0x8534
#define GL_PIXEL_COUNTER_BITS_NV 0x8864
#define GL_CURRENT_OCCLUSION_QUERY_ID_NV 0x8865
#define GL_PIXEL_COUNT_NV 0x8866
#define GL_PIXEL_COUNT_AVAILABLE_NV 0x8867
#define GL_DEPTH_STENCIL_NV 0x84F9
#define GL_UNSIGNED_INT_24_8_NV 0x84FA
#define GL_MAX_PROGRAM_PARAMETER_BUFFER_BINDINGS_NV 0x8DA0
#define GL_MAX_PROGRAM_PARAMETER_BUFFER_SIZE_NV 0x8DA1
#define GL_VERTEX_PROGRAM_PARAMETER_BUFFER_NV 0x8DA2
#define GL_GEOMETRY_PROGRAM_PARAMETER_BUFFER_NV 0x8DA3
#define GL_FRAGMENT_PROGRAM_PARAMETER_BUFFER_NV 0x8DA4
#define GL_PATH_FORMAT_SVG_NV 0x9070
#define GL_PATH_FORMAT_PS_NV 0x9071
#define GL_STANDARD_FONT_NAME_NV 0x9072
#define GL_SYSTEM_FONT_NAME_NV 0x9073
#define GL_FILE_NAME_NV 0x9074
#define GL_PATH_STROKE_WIDTH_NV 0x9075
#define GL_PATH_END_CAPS_NV 0x9076
#define GL_PATH_INITIAL_END_CAP_NV 0x9077
#define GL_PATH_TERMINAL_END_CAP_NV 0x9078
#define GL_PATH_JOIN_STYLE_NV 0x9079
#define GL_PATH_MITER_LIMIT_NV 0x907A
#define GL_PATH_DASH_CAPS_NV 0x907B
```

```
#define GL_PATH_INITIAL_DASH_CAP_NV 0x907C
#define GL_PATH_TERMINAL_DASH_CAP_NV 0x907D
#define GL_PATH_DASH_OFFSET_NV 0x907E
#define GL_PATH_CLIENT_LENGTH_NV 0x907F
#define GL_PATH_FILL_MODE_NV 0x9080
#define GL_PATH_FILL_MASK_NV 0x9081
#define GL_PATH_FILL_COVER_MODE_NV 0x9082
#define GL_PATH_STROKE_COVER_MODE_NV 0x9083
#define GL_PATH_STROKE_MASK_NV 0x9084
#define GL_COUNT_UP_NV 0x9088
#define GL_COUNT_DOWN_NV 0x9089
#define GL_PATH_OBJECT_BOUNDING_BOX_NV 0x908A
#define GL_CONVEX_HULL_NV 0x908B
#define GL_BOUNDING_BOX_NV 0x908D
#define GL_TRANSLATE_X_NV 0x908E
#define GL_TRANSLATE_Y_NV 0x908F
#define GL_TRANSLATE_2D_NV 0x9090
#define GL_TRANSLATE_3D_NV 0x9091
#define GL_AFFINE_2D_NV 0x9092
#define GL_AFFINE_3D_NV 0x9094
#define GL_TRANSPOSE_AFFINE_2D_NV 0x9096
#define GL_TRANSPOSE_AFFINE_3D_NV 0x9098
#define GL_UTF8_NV 0x909A
#define GL_UTF16_NV 0x909B
#define GL_BOUNDING_BOX_OF_BOUNDING_BOXES_NV 0x909C
#define GL_PATH_COMMAND_COUNT_NV 0x909D
#define GL_PATH_COORD_COUNT_NV 0x909E
#define GL_PATH_DASH_ARRAY_COUNT_NV 0x909F
#define GL_PATH_COMPUTED_LENGTH_NV 0x90A0
#define GL_PATH_FILL_BOUNDING_BOX_NV 0x90A1
#define GL_PATH_STROKE_BOUNDING_BOX_NV 0x90A2
#define GL_SQUARE_NV 0x90A3
#define GL_ROUND_NV 0x90A4
#define GL_TRIANGULAR_NV 0x90A5
#define GL_BEVEL_NV 0x90A6
#define GL_MITER_REVERT_NV 0x90A7
#define GL_MITER_TRUNCATE_NV 0x90A8
#define GL_SKIP_MISSING_GLYPH_NV 0x90A9
#define GL_USE_MISSING_GLYPH_NV 0x90AA
#define GL_PATH_ERROR_POSITION_NV 0x90AB
#define GL_ACCUM_ADJACENT_PAIRS_NV 0x90AD
#define GL_ADJACENT_PAIRS_NV 0x90AE
#define GL_FIRST_TO_REST_NV 0x90AF
#define GL_PATH_GEN_MODE_NV 0x90B0
#define GL_PATH_GEN_COEFF_NV 0x90B1
#define GL_PATH_GEN_COMPONENTS_NV 0x90B3
#define GL_PATH_STENCIL_FUNC_NV 0x90B7
#define GL_PATH_STENCIL_REF_NV 0x90B8
#define GL_PATH_STENCIL_VALUE_MASK_NV 0x90B9
#define GL_PATH_STENCIL_DEPTH_OFFSET_FACTOR_NV 0x90BD
#define GL_PATH_STENCIL_DEPTH_OFFSET_UNITS_NV 0x90BE
#define GL_PATH_COVER_DEPTH_FUNC_NV 0x90BF
#define GL_PATH_DASH_OFFSET_RESET_NV 0x90B4
#define GL_MOVE_TO_RESETS_NV 0x90B5
#define GL_MOVE_TO_CONTINUES_NV 0x90B6
#define GL_CLOSE_PATH_NV 0x00
#define GL_MOVE_TO_NV 0x02
#define GL_RELATIVE_MOVE_TO_NV 0x03
#define GL_LINE_TO_NV 0x04
#define GL_RELATIVE_LINE_TO_NV 0x05
#define GL_HORIZONTAL_LINE_TO_NV 0x06
#define GL_RELATIVE_HORIZONTAL_LINE_TO_NV 0x07
#define GL_VERTICAL_LINE_TO_NV 0x08
#define GL_RELATIVE_VERTICAL_LINE_TO_NV 0x09
#define GL_QUADRATIC_CURVE_TO_NV 0x0A
#define GL_RELATIVE_QUADRATIC_CURVE_TO_NV 0x0B
#define GL_CUBIC_CURVE_TO_NV 0x0C
#define GL_RELATIVE_CUBIC_CURVE_TO_NV 0x0D
#define GL_SMOOTH_QUADRATIC_CURVE_TO_NV 0x0E
#define GL_RELATIVE_SMOOTH_QUADRATIC_CURVE_TO_NV 0x0F
#define GL_SMOOTH_CUBIC_CURVE_TO_NV 0x10
#define GL_RELATIVE_SMOOTH_CUBIC_CURVE_TO_NV 0x11
#define GL_SMALL_CCW_ARC_TO_NV 0x12
#define GL_RELATIVE_SMALL_CCW_ARC_TO_NV 0x13
#define GL_SMALL_CW_ARC_TO_NV 0x14
#define GL_RELATIVE_SMALL_CW_ARC_TO_NV 0x15
#define GL_LARGE_CCW_ARC_TO_NV 0x16
#define GL_RELATIVE_LARGE_CCW_ARC_TO_NV 0x17
#define GL_LARGE_CW_ARC_TO_NV 0x18
#define GL_RELATIVE_LARGE_CW_ARC_TO_NV 0x19
#define GL_RESTART_PATH_NV 0xF0
#define GL_DUP_FIRST_CUBIC_CURVE_TO_NV 0xF2
#define GL_DUP_LAST_CUBIC_CURVE_TO_NV 0xF4
#define GL_RECT_NV 0xF6
#define GL_CIRCULAR_CCW_ARC_TO_NV 0xF8
#define GL_CIRCULAR_CW_ARC_TO_NV 0xFA
#define GL_CIRCULAR_TANGENT_ARC_TO_NV 0xFC
```

```
#define GL_ARC_TO_NV 0xFE
#define GL_RELATIVE_ARC_TO_NV 0xFF
#define GL_BOLD_BIT_NV 0x01
#define GL_ITALIC_BIT_NV 0x02
#define GL_GLYPH_WIDTH_BIT_NV 0x01
#define GL_GLYPH_HEIGHT_BIT_NV 0x02
#define GL_GLYPH_HORIZONTAL_BEARING_X_BIT_NV 0x04
#define GL_GLYPH_HORIZONTAL_BEARING_Y_BIT_NV 0x08
#define GL_GLYPH_HORIZONTAL_BEARING_ADVANCE_BIT_NV 0x10
#define GL_GLYPH_VERTICAL_BEARING_X_BIT_NV 0x20
#define GL_GLYPH_VERTICAL_BEARING_Y_BIT_NV 0x40
#define GL_GLYPH_VERTICAL_BEARING_ADVANCE_BIT_NV 0x80
#define GL_GLYPH_HAS_KERNING_BIT_NV 0x100
#define GL_FONT_X_MIN_BOUNDS_BIT_NV 0x00010000
#define GL_FONT_Y_MIN_BOUNDS_BIT_NV 0x00020000
#define GL_FONT_X_MAX_BOUNDS_BIT_NV 0x00040000
#define GL_FONT_Y_MAX_BOUNDS_BIT_NV 0x00080000
#define GL_FONT_UNITS_PER_EM_BIT_NV 0x00100000
#define GL_FONT_ASCENDER_BIT_NV 0x00200000
#define GL_FONT_DESCENDER_BIT_NV 0x00400000
#define GL_FONT_HEIGHT_BIT_NV 0x00800000
#define GL_FONT_MAX_ADVANCE_WIDTH_BIT_NV 0x01000000
#define GL_FONT_MAX_ADVANCE_HEIGHT_BIT_NV 0x02000000
#define GL_FONT_UNDERLINE_POSITION_BIT_NV 0x04000000
#define GL_FONT_UNDERLINE_THICKNESS_BIT_NV 0x08000000
#define GL_FONT_HAS_KERNING_BIT_NV 0x10000000
#define GL_ROUNDED_RECT_NV 0xE8
#define GL_RELATIVE_ROUNDED_RECT_NV 0xE9
#define GL_ROUNDED_RECT2_NV 0xEA
#define GL_RELATIVE_ROUNDED_RECT2_NV 0xEB
#define GL_ROUNDED_RECT4_NV 0xEC
#define GL_RELATIVE_ROUNDED_RECT4_NV 0xED
#define GL_ROUNDED_RECT8_NV 0xEE
#define GL_RELATIVE_ROUNDED_RECT8_NV 0xEF
#define GL_RELATIVE_RECT_NV 0xF7
#define GL_FONT_GLYPHS_AVAILABLE_NV 0x9368
#define GL_FONT_TARGET_UNAVAILABLE_NV 0x9369
#define GL_FONT_UNAVAILABLE_NV 0x936A
#define GL_FONT_UNINTELLIGIBLE_NV 0x936B
#define GL_CONIC_CURVE_TO_NV 0x1A
#define GL_RELATIVE_CONIC_CURVE_TO_NV 0x1B
#define GL_FONT_NUM_GLYPH_INDICES_BIT_NV 0x20000000
#define GL_STANDARD_FONT_FORMAT_NV 0x936C
#define GL_2_BYTES_NV 0x1407
#define GL_3_BYTES_NV 0x1408
#define GL_4_BYTES_NV 0x1409
#define GL_EYE_LINEAR_NV 0x2400
#define GL_OBJECT_LINEAR_NV 0x2401
#define GL_CONSTANT_NV 0x8576
#define GL_PATH_FOG_GEN_MODE_NV 0x90AC
#define GL_PRIMARY_COLOR 0x8577
#define GL_PRIMARY_COLOR_NV 0x852C
#define GL_SECONDARY_COLOR_NV 0x852D
#define GL_PATH_GEN_COLOR_FORMAT_NV 0x90B2
#define GL_PATH_PROJECTION_NV 0x1701
#define GL_PATH_MODELVIEW_NV 0x1700
#define GL_PATH_MODELVIEW_STACK_DEPTH_NV 0x0BA3
#define GL_PATH_MODELVIEW_MATRIX_NV 0x0BA6
#define GL_PATH_MAX_MODELVIEW_STACK_DEPTH_NV 0x0D36
#define GL_PATH_TRANSPOSE_MODELVIEW_MATRIX_NV 0x84E3
#define GL_PATH_PROJECTION_STACK_DEPTH_NV 0x0BA4
#define GL_PATH_PROJECTION_MATRIX_NV 0x0BA7
#define GL_PATH_MAX_PROJECTION_STACK_DEPTH_NV 0x0D38
#define GL_PATH_TRANSPOSE_PROJECTION_MATRIX_NV 0x84E4
#define GL_FRAGMENT_INPUT_NV 0x936D
#define GL_SHARED_EDGE_NV 0xC0
#define GL_WRITE_PIXEL_DATA_RANGE_NV 0x8878
#define GL_READ_PIXEL_DATA_RANGE_NV 0x8879
#define GL_WRITE_PIXEL_DATA_RANGE_LENGTH_NV 0x887A
#define GL_READ_PIXEL_DATA_RANGE_LENGTH_NV 0x887B
#define GL_WRITE_PIXEL_DATA_RANGE_POINTER_NV 0x887C
#define GL_READ_PIXEL_DATA_RANGE_POINTER_NV 0x887D
#define GL_POINT_SPRITE_NV 0x8861
#define GL_COORD_REPLACE_NV 0x8862
#define GL_POINT_SPRITE_R_MODE_NV 0x8863
#define GL_FRAME_NV 0x8E26
#define GL_FIELDS_NV 0x8E27
#define GL_CURRENT_TIME_NV 0x8E28
#define GL_NUM_FILL_STREAMS_NV 0x8E29
#define GL_PRESENT_TIME_NV 0x8E2A
#define GL_PRESENT_DURATION_NV 0x8E2B
#define GL_PRIMITIVE_RESTART_NV 0x8558
#define GL_PRIMITIVE_RESTART_INDEX_NV 0x8559
#define GL_SHADING_RATE_IMAGE_PER_PRIMITIVE_NV 0x95B1
#define GL_SHADING_RATE_IMAGE_PALETTE_COUNT_NV 0x95B2
#define GL_QUERY_RESOURCE_TYPE_VIDMEM_ALLOC_NV 0x9540
#define GL_QUERY_RESOURCE_MEMTYPE_VIDMEM_NV 0x9542
```

```
#define GL_QUERY_RESOURCE_SYS_RESERVED_NV 0x9544
#define GL_QUERY_RESOURCE_TEXTURE_NV 0x9545
#define GL_QUERY_RESOURCE_RENDERBUFFER_NV 0x9546
#define GL_QUERY_RESOURCE_BUFFEROBJECT_NV 0x9547
#define GL_REGISTER_COMBINERS_NV 0x8522
#define GL_VARIABLE_A_NV 0x8523
#define GL_VARIABLE_B_NV 0x8524
#define GL_VARIABLE_C_NV 0x8525
#define GL_VARIABLE_D_NV 0x8526
#define GL_VARIABLE_E_NV 0x8527
#define GL_VARIABLE_F_NV 0x8528
#define GL_VARIABLE_G_NV 0x8529
#define GL_CONSTANT_COLOR0_NV 0x852A
#define GL_CONSTANT_COLOR1_NV 0x852B
#define GL_SPARE0_NV 0x852E
#define GL_SPARE1_NV 0x852F
#define GL_DISCARD_NV 0x8530
#define GL_E_TIMES_F_NV 0x8531
#define GL_SPARE0_PLUS_SECONDARY_COLOR_NV 0x8532
#define GL_UNSIGNED_IDENTITY_NV 0x8536
#define GL_UNSIGNED_INVERT_NV 0x8537
#define GL_EXPAND_NORMAL_NV 0x8538
#define GL_EXPAND_NEGATE_NV 0x8539
#define GL_HALF_BIAS_NORMAL_NV 0x853A
#define GL_HALF_BIAS_NEGATE_NV 0x853B
#define GL_SIGNED_IDENTITY_NV 0x853C
#define GL_SIGNED_NEGATE_NV 0x853D
#define GL_SCALE_BY_TWO_NV 0x853E
#define GL_SCALE_BY_FOUR_NV 0x853F
#define GL_SCALE_BY_ONE_HALF_NV 0x8540
#define GL_BIAS_BY_NEGATIVE_ONE_HALF_NV 0x8541
#define GL_COMBINER_INPUT_NV 0x8542
#define GL_COMBINER_MAPPING_NV 0x8543
#define GL_COMBINER_COMPONENT_USAGE_NV 0x8544
#define GL_COMBINER_AB_DOT_PRODUCT_NV 0x8545
#define GL_COMBINER_CD_DOT_PRODUCT_NV 0x8546
#define GL_COMBINER_MUX_SUM_NV 0x8547
#define GL_COMBINER_SCALE_NV 0x8548
#define GL_COMBINER_BIAS_NV 0x8549
#define GL_COMBINER_AB_OUTPUT_NV 0x854A
#define GL_COMBINER_CD_OUTPUT_NV 0x854B
#define GL_COMBINER_SUM_OUTPUT_NV 0x854C
#define GL_MAX_GENERAL_COMBINERS_NV 0x854D
#define GL_NUM_GENERAL_COMBINERS_NV 0x854E
#define GL_COLOR_SUM_CLAMP_NV 0x854F
#define GL_COMBINER0_NV 0x8550
#define GL_COMBINER1_NV 0x8551
#define GL_COMBINER2_NV 0x8552
#define GL_COMBINER3_NV 0x8553
#define GL_COMBINER4_NV 0x8554
#define GL_COMBINER5_NV 0x8555
#define GL_COMBINER6_NV 0x8556
#define GL_COMBINER7_NV 0x8557
#define GL_FOG 0x0B60
#define GL_PER_STAGE_CONSTANTS_NV 0x8535
#define GL_REPRESENTATIVE_FRAGMENT_TEST_NV 0x937F
#define GL_PURGED_CONTEXT_RESET_NV 0x92BB
#define GL_SAMPLE_LOCATION_SUBPIXEL_BITS_NV 0x933D
#define GL_SAMPLE_LOCATION_PIXEL_GRID_WIDTH_NV 0x933E
#define GL_SAMPLE_LOCATION_PIXEL_GRID_HEIGHT_NV 0x933F
#define GL_PROGRAMMABLE_SAMPLE_LOCATION_TABLE_SIZE_NV 0x9340
#define GL_SAMPLE_LOCATION_NV 0x8E50
#define GL_PROGRAMMABLE_SAMPLE_LOCATION_NV 0x9341
#define GL_FRAMEBUFFER_PROGRAMMABLE_SAMPLE_LOCATIONS_NV 0x9342
#define GL_FRAMEBUFFER_SAMPLE_LOCATION_PIXEL_GRID_NV 0x9343
#define GL_SCISSOR_TEST_EXCLUSIVE_NV 0x9555
#define GL_SCISSOR_BOX_EXCLUSIVE_NV 0x9556
#define GL_BUFFER_GPU_ADDRESS_NV 0x8F1D
#define GL_GPU_ADDRESS_NV 0x8F34
#define GL_MAX_SHADER_BUFFER_ADDRESS_NV 0x8F35
#define GL_SHADER_GLOBAL_ACCESS_BARRIER_BIT_NV 0x00000010
#define GL_SUBGROUP_FEATURE_PARTITIONED_BIT_NV 0x00000100
#define GL_WARP_SIZE_NV 0x9339
#define GL_WARPS_PER_SM_NV 0x933A
#define GL_SM_COUNT_NV 0x933B
#define GL_SHADING_RATE_IMAGE_NV 0x9563
#define GL_SHADING_RATE_NO_INVOCATIONS_NV 0x9564
#define GL_SHADING_RATE_1_INVOCATION_PER_PIXEL_NV 0x9565
#define GL_SHADING_RATE_1_INVOCATION_PER_1X2_PIXELS_NV 0x9566
#define GL_SHADING_RATE_1_INVOCATION_PER_2X1_PIXELS_NV 0x9567
#define GL_SHADING_RATE_1_INVOCATION_PER_2X2_PIXELS_NV 0x9568
#define GL_SHADING_RATE_1_INVOCATION_PER_2X4_PIXELS_NV 0x9569
#define GL_SHADING_RATE_1_INVOCATION_PER_4X2_PIXELS_NV 0x956A
#define GL_SHADING_RATE_1_INVOCATION_PER_4X4_PIXELS_NV 0x956B
#define GL_SHADING_RATE_2_INVOCATIONS_PER_PIXEL_NV 0x956C
#define GL_SHADING_RATE_4_INVOCATIONS_PER_PIXEL_NV 0x956D
#define GL_SHADING_RATE_8_INVOCATIONS_PER_PIXEL_NV 0x956E
```

```
#define GL_SHADING_RATE_16_INVOCATIONS_PER_PIXEL_NV 0x956F
#define GL_SHADING_RATE_IMAGE_BINDING_NV 0x955B
#define GL_SHADING_RATE_IMAGE_TEXEL_WIDTH_NV 0x955C
#define GL_SHADING_RATE_IMAGE_TEXEL_HEIGHT_NV 0x955D
#define GL_SHADING_RATE_IMAGE_PALETTE_SIZE_NV 0x955E
#define GL_MAX_COARSE_FRAGMENT_SAMPLES_NV 0x955F
#define GL_SHADING_RATE_SAMPLE_ORDER_DEFAULT_NV 0x95AE
#define GL_SHADING_RATE_SAMPLE_ORDER_PIXEL_MAJOR_NV 0x95AF
#define GL_SHADING_RATE_SAMPLE_ORDER_SAMPLE_MAJOR_NV 0x95B0
#define GL_MAX_PROGRAM_PATCH_ATTRIBS_NV 0x86D8
#define GL_TESS_CONTROL_PROGRAM_NV 0x891E
#define GL_TESS_EVALUATION_PROGRAM_NV 0x891F
#define GL_TESS_CONTROL_PROGRAM_PARAMETER_BUFFER_NV 0x8C74
#define GL_TESS_EVALUATION_PROGRAM_PARAMETER_BUFFER_NV 0x8C75
#define GL_EMBOSS_LIGHT_NV 0x855D
#define GL_EMBOSS_CONSTANT_NV 0x855E
#define GL_EMBOSS_MAP_NV 0x855F
#define GL_NORMAL_MAP_NV 0x8511
#define GL_REFLECTION_MAP_NV 0x8512
#define GL_COMBINE4_NV 0x8503
#define GL_SOURCE3_RGB_NV 0x8583
#define GL_SOURCE3_ALPHA_NV 0x858B
#define GL_OPERAND3_RGB_NV 0x8593
#define GL_OPERAND3_ALPHA_NV 0x859B
#define GL_TEXTURE_UNSIGNED_REMAP_MODE_NV 0x888F
#define GL_TEXTURE_COVERAGE_SAMPLES_NV 0x9045
#define GL_TEXTURE_COLOR_SAMPLES_NV 0x9046
#define GL_TEXTURE_RECTANGLE_NV 0x84F5
#define GL_TEXTURE_BINDING_RECTANGLE_NV 0x84F6
#define GL_PROXY_TEXTURE_RECTANGLE_NV 0x84F7
#define GL_MAX_RECTANGLE_TEXTURE_SIZE_NV 0x84F8
#define GL_OFFSET_TEXTURE_RECTANGLE_NV 0x864C
#define GL_OFFSET_TEXTURE_RECTANGLE_SCALE_NV 0x864D
#define GL_DOT_PRODUCT_TEXTURE_RECTANGLE_NV 0x864E
#define GL_RGBA_UNSIGNED_DOT_PRODUCT_MAPPING_NV 0x86D9
#define GL_UNSIGNED_INT_S8_S8_8_8_NV 0x86DA
#define GL_UNSIGNED_INT_8_8_S8_S8_REV_NV 0x86DB
#define GL_DSDT_MAG_INTENSITY_NV 0x86DC
#define GL_SHADER_CONSISTENT_NV 0x86DD
#define GL_TEXTURE_SHADER_NV 0x86DE
#define GL_SHADER_OPERATION_NV 0x86DF
#define GL_CULL_MODES_NV 0x86E0
#define GL_OFFSET_TEXTURE_MATRIX_NV 0x86E1
#define GL_OFFSET_TEXTURE_SCALE_NV 0x86E2
#define GL_OFFSET_TEXTURE_BIAS_NV 0x86E3
#define GL_OFFSET_TEXTURE_2D_MATRIX_NV 0x86E1
#define GL_OFFSET_TEXTURE_2D_SCALE_NV 0x86E2
#define GL_OFFSET_TEXTURE_2D_BIAS_NV 0x86E3
#define GL_PREVIOUS_TEXTURE_INPUT_NV 0x86E4
#define GL_CONST_EYE_NV 0x86E5
#define GL_PASS_THROUGH_NV 0x86E6
#define GL_CULL_FRAGMENT_NV 0x86E7
#define GL_OFFSET_TEXTURE_2D_NV 0x86E8
#define GL_DEPENDENT_AR_TEXTURE_2D_NV 0x86E9
#define GL_DEPENDENT_GB_TEXTURE_2D_NV 0x86EA
#define GL_DOT_PRODUCT_NV 0x86EC
#define GL_DOT_PRODUCT_DEPTH_REPLACE_NV 0x86ED
#define GL_DOT_PRODUCT_TEXTURE_2D_NV 0x86EE
#define GL_DOT_PRODUCT_TEXTURE_CUBE_MAP_NV 0x86F0
#define GL_DOT_PRODUCT_DIFFUSE_CUBE_MAP_NV 0x86F1
#define GL_DOT_PRODUCT_REFLECT_CUBE_MAP_NV 0x86F2
#define GL_DOT_PRODUCT_CONST_EYE_REFLECT_CUBE_MAP_NV 0x86F3
#define GL_HILO_NV 0x86F4
#define GL_DSDT_NV 0x86F5
#define GL_DSDT_MAG_NV 0x86F6
#define GL_DSDT_MAG_VIB_NV 0x86F7
#define GL_HILO16_NV 0x86F8
#define GL_SIGNED_HILO_NV 0x86F9
#define GL_SIGNED_HILO16_NV 0x86FA
#define GL_SIGNED_RGBA_NV 0x86FB
#define GL_SIGNED_RGBA8_NV 0x86FC
#define GL_SIGNED_RGB_NV 0x86FE
#define GL_SIGNED_RGB8_NV 0x86FF
#define GL_SIGNED_LUMINANCE_NV 0x8701
#define GL_SIGNED_LUMINANCE8_NV 0x8702
#define GL_SIGNED_LUMINANCE_ALPHA_NV 0x8703
#define GL_SIGNED_LUMINANCE8_ALPHA8_NV 0x8704
#define GL_SIGNED_ALPHA_NV 0x8705
#define GL_SIGNED_ALPHA8_NV 0x8706
#define GL_SIGNED_INTENSITY_NV 0x8707
#define GL_SIGNED_INTENSITY8_NV 0x8708
#define GL_DSDT8_NV 0x8709
#define GL_DSDT8_MAG8_NV 0x870A
#define GL_DSDT8_MAG8_INTENSITY8_NV 0x870B
#define GL_SIGNED_RGB_UNSIGNED_ALPHA_NV 0x870C
#define GL_SIGNED_RGB8_UNSIGNED_ALPHA8_NV 0x870D
#define GL_HI_SCALE_NV 0x870E
```

```
#define GL_LO_SCALE_NV 0x870F
#define GL_DS_SCALE_NV 0x8710
#define GL_DT_SCALE_NV 0x8711
#define GL_MAGNITUDE_SCALE_NV 0x8712
#define GL_VIBRANCE_SCALE_NV 0x8713
#define GL_HI_BIAS_NV 0x8714
#define GL_LO_BIAS_NV 0x8715
#define GL_DS_BIAS_NV 0x8716
#define GL_DT_BIAS_NV 0x8717
#define GL_MAGNITUDE_BIAS_NV 0x8718
#define GL_VIBRANCE_BIAS_NV 0x8719
#define GL_TEXTURE_BORDER_VALUES_NV 0x871A
#define GL_TEXTURE_HI_SIZE_NV 0x871B
#define GL_TEXTURE_LO_SIZE_NV 0x871C
#define GL_TEXTURE_DS_SIZE_NV 0x871D
#define GL_TEXTURE_DT_SIZE_NV 0x871E
#define GL_TEXTURE_MAG_SIZE_NV 0x871F
#define GL_DOT_PRODUCT_TEXTURE_3D_NV 0x86EF
#define GL_OFFSET_PROJECTIVE_TEXTURE_2D_NV 0x8850
#define GL_OFFSET_PROJECTIVE_TEXTURE_2D_SCALE_NV 0x8851
#define GL_OFFSET_PROJECTIVE_TEXTURE_RECTANGLE_NV 0x8852
#define GL_OFFSET_PROJECTIVE_TEXTURE_RECTANGLE_SCALE_NV 0x8853
#define GL_OFFSET_HILO_TEXTURE_2D_NV 0x8854
#define GL_OFFSET_HILO_TEXTURE_RECTANGLE_NV 0x8855
#define GL_OFFSET_HILO_PROJECTIVE_TEXTURE_2D_NV 0x8856
#define GL_OFFSET_HILO_PROJECTIVE_TEXTURE_RECTANGLE_NV 0x8857
#define GL_DEPENDENT_HILO_TEXTURE_2D_NV 0x8858
#define GL_DEPENDENT_RGB_TEXTURE_3D_NV 0x8859
#define GL_DEPENDENT_RGB_TEXTURE_CUBE_MAP_NV 0x885A
#define GL_DOT_PRODUCT_PASS_THROUGH_NV 0x885B
#define GL_DOT_PRODUCT_TEXTURE_1D_NV 0x885C
#define GL_DOT_PRODUCT_AFFINE_DEPTH_REPLACE_NV 0x885D
#define GL_HILO8_NV 0x885E
#define GL_SIGNED_HILO8_NV 0x885F
#define GL_FORCE_BLUE_TO_ONE_NV 0x8860
#define GL_TIMELINE_SEMAPHORE_VALUE_NV 0x9595
#define GL_SEMAPHORE_TYPE_NV 0x95B3
#define GL_SEMAPHORE_TYPE_BINARY_NV 0x95B4
#define GL_SEMAPHORE_TYPE_TIMELINE_NV 0x95B5
#define GL_MAX_TIMELINE_SEMAPHORE_VALUE_DIFFERENCE_NV 0x95B6
#define GL_BACK_PRIMARY_COLOR_NV 0x8C77
#define GL_BACK_SECONDARY_COLOR_NV 0x8C78
#define GL_TEXTURE_COORD_NV 0x8C79
#define GL_CLIP_DISTANCE_NV 0x8C7A
#define GL_VERTEX_ID_NV 0x8C7B
#define GL_PRIMITIVE_ID_NV 0x8C7C
#define GL_GENERIC_ATTRIB_NV 0x8C7D
#define GL_TRANSFORM_FEEDBACK_ATTRIBS_NV 0x8C7E
#define GL_TRANSFORM_FEEDBACK_BUFFER_MODE_NV 0x8C7F
#define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_COMPONENTS_NV 0x8C80
#define GL_ACTIVE_VARYINGS_NV 0x8C81
#define GL_ACTIVE_VARYING_MAX_LENGTH_NV 0x8C82
#define GL_TRANSFORM_FEEDBACK_VARYINGS_NV 0x8C83
#define GL_TRANSFORM_FEEDBACK_BUFFER_START_NV 0x8C84
#define GL_TRANSFORM_FEEDBACK_BUFFER_SIZE_NV 0x8C85
#define GL_TRANSFORM_FEEDBACK_RECORD_NV 0x8C86
#define GL_PRIMITIVES_GENERATED_NV 0x8C87
#define GL_TRANSFORM_FEEDBACK_PRIMITIVES_WRITTEN_NV 0x8C88
#define GL_RASTERIZER_DISCARD_NV 0x8C89
#define GL_MAX_TRANSFORM_FEEDBACK_INTERLEAVED_COMPONENTS_NV 0x8C8A
#define GL_MAX_TRANSFORM_FEEDBACK_SEPARATE_ATTRIBS_NV 0x8C8B
#define GL_INTERLEAVED_ATTRIBS_NV 0x8C8C
#define GL_SEPARATE_ATTRIBS_NV 0x8C8D
#define GL_TRANSFORM_FEEDBACK_BUFFER_NV 0x8C8E
#define GL_TRANSFORM_FEEDBACK_BUFFER_BINDING_NV 0x8C8F
#define GL_LAYER_NV 0x8DAA
#define GL_NEXT_BUFFER_NV -2
#define GL_SKIP_COMPONENTS4_NV -3
#define GL_SKIP_COMPONENTS3_NV -4
#define GL_SKIP_COMPONENTS2_NV -5
#define GL_SKIP_COMPONENTS1_NV -6
#define GL_TRANSFORM_FEEDBACK_NV 0x8E22
#define GL_TRANSFORM_FEEDBACK_BUFFER_PAUSED_NV 0x8E23
#define GL_TRANSFORM_FEEDBACK_BUFFER_ACTIVE_NV 0x8E24
#define GL_TRANSFORM_FEEDBACK_BINDING_NV 0x8E25
#define GL_UNIFORM_BUFFER_UNIFIED_NV 0x936E
#define GL_UNIFORM_BUFFER_ADDRESS_NV 0x936F
#define GL_UNIFORM_BUFFER_LENGTH_NV 0x9370
#define GL_SURFACE_STATE_NV 0x86EB
#define GL_SURFACE_REGISTERED_NV 0x86FD
#define GL_SURFACE_MAPPED_NV 0x8700
#define GL_WRITE_DISCARD_NV 0x88BE
#define GL_VERTEX_ARRAY_RANGE_NV 0x851D
#define GL_VERTEX_ARRAY_RANGE_LENGTH_NV 0x851E
#define GL_VERTEX_ARRAY_RANGE_VALID_NV 0x851F
#define GL_MAX_VERTEX_ARRAY_RANGE_ELEMENT_NV 0x8520
#define GL_VERTEX_ARRAY_RANGE_POINTER_NV 0x8521
```

```
#define GL_VERTEX_ARRAY_RANGE_WITHOUT_FLUSH_NV 0x8533
#define GL_VERTEX_ATTRIB_ARRAY_UNIFIED_NV 0x8F1E
#define GL_ELEMENT_ARRAY_UNIFIED_NV 0x8F1F
#define GL_VERTEX_ATTRIB_ARRAY_ADDRESS_NV 0x8F20
#define GL_VERTEX_ARRAY_ADDRESS_NV 0x8F21
#define GL_NORMAL_ARRAY_ADDRESS_NV 0x8F22
#define GL_COLOR_ARRAY_ADDRESS_NV 0x8F23
#define GL_INDEX_ARRAY_ADDRESS_NV 0x8F24
#define GL_TEXTURE_COORD_ARRAY_ADDRESS_NV 0x8F25
#define GL_EDGE_FLAG_ARRAY_ADDRESS_NV 0x8F26
#define GL_SECONDARY_COLOR_ARRAY_ADDRESS_NV 0x8F27
#define GL_FOG_COORD_ARRAY_ADDRESS_NV 0x8F28
#define GL_ELEMENT_ARRAY_ADDRESS_NV 0x8F29
#define GL_VERTEX_ATTRIB_ARRAY_LENGTH_NV 0x8F2A
#define GL_VERTEX_ARRAY_LENGTH_NV 0x8F2B
#define GL_NORMAL_ARRAY_LENGTH_NV 0x8F2C
#define GL_COLOR_ARRAY_LENGTH_NV 0x8F2D
#define GL_INDEX_ARRAY_LENGTH_NV 0x8F2E
#define GL_TEXTURE_COORD_ARRAY_LENGTH_NV 0x8F2F
#define GL_EDGE_FLAG_ARRAY_LENGTH_NV 0x8F30
#define GL_SECONDARY_COLOR_ARRAY_LENGTH_NV 0x8F31
#define GL_FOG_COORD_ARRAY_LENGTH_NV 0x8F32
#define GL_ELEMENT_ARRAY_LENGTH_NV 0x8F33
#define GL_DRAW_INDIRECT_UNIFIED_NV 0x8F40
#define GL_DRAW_INDIRECT_ADDRESS_NV 0x8F41
#define GL_DRAW_INDIRECT_LENGTH_NV 0x8F42
#define GL_VERTEX_PROGRAM_NV 0x8620
#define GL_VERTEX_STATE_PROGRAM_NV 0x8621
#define GL_ATTRIB_ARRAY_SIZE_NV 0x8623
#define GL_ATTRIB_ARRAY_STRIDE_NV 0x8624
#define GL_ATTRIB_ARRAY_TYPE_NV 0x8625
#define GL_CURRENT_ATTRIB_NV 0x8626
#define GL_PROGRAM_LENGTH_NV 0x8627
#define GL_PROGRAM_STRING_NV 0x8628
#define GL_MODELVIEW_PROJECTION_NV 0x8629
#define GL_IDENTITY_NV 0x862A
#define GL_INVERSE_NV 0x862B
#define GL_TRANSPOSE_NV 0x862C
#define GL_INVERSE_TRANSPOSE_NV 0x862D
#define GL_MAX_TRACK_MATRIX_STACK_DEPTH_NV 0x862E
#define GL_MAX_TRACK_MATRICES_NV 0x862F
#define GL_MATRIX0_NV 0x8630
#define GL_MATRIX1_NV 0x8631
#define GL_MATRIX2_NV 0x8632
#define GL_MATRIX3_NV 0x8633
#define GL_MATRIX4_NV 0x8634
#define GL_MATRIX5_NV 0x8635
#define GL_MATRIX6_NV 0x8636
#define GL_MATRIX7_NV 0x8637
#define GL_CURRENT_MATRIX_STACK_DEPTH_NV 0x8640
#define GL_CURRENT_MATRIX_NV 0x8641
#define GL_VERTEX_PROGRAM_POINT_SIZE_NV 0x8642
#define GL_VERTEX_PROGRAM_TWO_SIDE_NV 0x8643
#define GL_PROGRAM_PARAMETER_NV 0x8644
#define GL_ATTRIB_ARRAY_POINTER_NV 0x8645
#define GL_PROGRAM_TARGET_NV 0x8646
#define GL_PROGRAM_RESIDENT_NV 0x8647
#define GL_TRACK_MATRIX_NV 0x8648
#define GL_TRACK_MATRIX_TRANSFORM_NV 0x8649
#define GL_VERTEX_PROGRAM_BINDING_NV 0x864A
#define GL_PROGRAM_ERROR_POSITION_NV 0x864B
#define GL_VERTEX_ATTRIB_ARRAY0_NV 0x8650
#define GL_VERTEX_ATTRIB_ARRAY1_NV 0x8651
#define GL_VERTEX_ATTRIB_ARRAY2_NV 0x8652
#define GL_VERTEX_ATTRIB_ARRAY3_NV 0x8653
#define GL_VERTEX_ATTRIB_ARRAY4_NV 0x8654
#define GL_VERTEX_ATTRIB_ARRAY5_NV 0x8655
#define GL_VERTEX_ATTRIB_ARRAY6_NV 0x8656
#define GL_VERTEX_ATTRIB_ARRAY7_NV 0x8657
#define GL_VERTEX_ATTRIB_ARRAY8_NV 0x8658
#define GL_VERTEX_ATTRIB_ARRAY9_NV 0x8659
#define GL_VERTEX_ATTRIB_ARRAY10_NV 0x865A
#define GL_VERTEX_ATTRIB_ARRAY11_NV 0x865B
#define GL_VERTEX_ATTRIB_ARRAY12_NV 0x865C
#define GL_VERTEX_ATTRIB_ARRAY13_NV 0x865D
#define GL_VERTEX_ATTRIB_ARRAY14_NV 0x865E
#define GL_VERTEX_ATTRIB_ARRAY15_NV 0x865F
#define GL_MAP1_VERTEX_ATTRIB0_4_NV 0x8660
#define GL_MAP1_VERTEX_ATTRIB1_4_NV 0x8661
#define GL_MAP1_VERTEX_ATTRIB2_4_NV 0x8662
#define GL_MAP1_VERTEX_ATTRIB3_4_NV 0x8663
#define GL_MAP1_VERTEX_ATTRIB4_4_NV 0x8664
#define GL_MAP1_VERTEX_ATTRIB5_4_NV 0x8665
#define GL_MAP1_VERTEX_ATTRIB6_4_NV 0x8666
#define GL_MAP1_VERTEX_ATTRIB7_4_NV 0x8667
#define GL_MAP1_VERTEX_ATTRIB8_4_NV 0x8668
#define GL_MAP1_VERTEX_ATTRIB9_4_NV 0x8669
```

```
#define GL_MAP1_VERTEX_ATTRIB10_4_NV 0x866A
#define GL_MAP1_VERTEX_ATTRIB11_4_NV 0x866B
#define GL_MAP1_VERTEX_ATTRIB12_4_NV 0x866C
#define GL_MAP1_VERTEX_ATTRIB13_4_NV 0x866D
#define GL_MAP1_VERTEX_ATTRIB14_4_NV 0x866E
#define GL_MAP1_VERTEX_ATTRIB15_4_NV 0x866F
#define GL_MAP2_VERTEX_ATTRIB0_4_NV 0x8670
#define GL_MAP2_VERTEX_ATTRIB1_4_NV 0x8671
#define GL_MAP2_VERTEX_ATTRIB2_4_NV 0x8672
#define GL_MAP2_VERTEX_ATTRIB3_4_NV 0x8673
#define GL_MAP2_VERTEX_ATTRIB4_4_NV 0x8674
#define GL_MAP2_VERTEX_ATTRIB5_4_NV 0x8675
#define GL_MAP2_VERTEX_ATTRIB6_4_NV 0x8676
#define GL_MAP2_VERTEX_ATTRIB7_4_NV 0x8677
#define GL_MAP2_VERTEX_ATTRIB8_4_NV 0x8678
#define GL_MAP2_VERTEX_ATTRIB9_4_NV 0x8679
#define GL_MAP2_VERTEX_ATTRIB10_4_NV 0x867A
#define GL_MAP2_VERTEX_ATTRIB11_4_NV 0x867B
#define GL_MAP2_VERTEX_ATTRIB12_4_NV 0x867C
#define GL_MAP2_VERTEX_ATTRIB13_4_NV 0x867D
#define GL_MAP2_VERTEX_ATTRIB14_4_NV 0x867E
#define GL_MAP2_VERTEX_ATTRIB15_4_NV 0x867F
#define GL_VERTEX_ATTRIB_ARRAY_INTEGER_NV 0x88FD
#define GL_VIDEO_BUFFER_NV 0x9020
#define GL_VIDEO_BUFFER_BINDING_NV 0x9021
#define GL_FIELD_UPPER_NV 0x9022
#define GL_FIELD_LOWER_NV 0x9023
#define GL_NUM_VIDEO_CAPTURE_STREAMS_NV 0x9024
#define GL_NEXT_VIDEO_CAPTURE_BUFFER_STATUS_NV 0x9025
#define GL_VIDEO_CAPTURE_TO_422_SUPPORTED_NV 0x9026
#define GL_LAST_VIDEO_CAPTURE_STATUS_NV 0x9027
#define GL_VIDEO_BUFFER_PITCH_NV 0x9028
#define GL_VIDEO_COLOR_CONVERSION_MATRIX_NV 0x9029
#define GL_VIDEO_COLOR_CONVERSION_MAX_NV 0x902A
#define GL_VIDEO_COLOR_CONVERSION_MIN_NV 0x902B
#define GL_VIDEO_COLOR_CONVERSION_OFFSET_NV 0x902C
#define GL_VIDEO_BUFFER_INTERNAL_FORMAT_NV 0x902D
#define GL_PARTIAL_SUCCESS_NV 0x902E
#define GL_SUCCESS_NV 0x902F
#define GL_FAILURE_NV 0x9030
#define GL_YCBYCR8_422_NV 0x9031
#define GL_YCBAYCR8A_4224_NV 0x9032
#define GL_Z6Y10Z6CB10Z6Y10Z6CR10_422_NV 0x9033
#define GL_Z6Y10Z6CB10Z6A10Z6Y10Z6CR10Z6A10_4224_NV 0x9034
#define GL_Z4Y12Z4CB12Z4Y12Z4CR12_422_NV 0x9035
#define GL_Z4Y12Z4CB12Z4A12Z4Y12Z4CR12Z4A12_4224_NV 0x9036
#define GL_Z4Y12Z4CB12Z4CR12_444_NV 0x9037
#define GL_VIDEO_CAPTURE_FRAME_WIDTH_NV 0x9038
#define GL_VIDEO_CAPTURE_FRAME_HEIGHT_NV 0x9039
#define GL_VIDEO_CAPTURE_FIELD_UPPER_HEIGHT_NV 0x903A
#define GL_VIDEO_CAPTURE_FIELD_LOWER_HEIGHT_NV 0x903B
#define GL_VIDEO_CAPTURE_SURFACE_ORIGIN_NV 0x903C
#define GL_VIEWPORT_SWIZZLE_POSITIVE_X_NV 0x9350
#define GL_VIEWPORT_SWIZZLE_NEGATIVE_X_NV 0x9351
#define GL_VIEWPORT_SWIZZLE_POSITIVE_Y_NV 0x9352
#define GL_VIEWPORT_SWIZZLE_NEGATIVE_Y_NV 0x9353
#define GL_VIEWPORT_SWIZZLE_POSITIVE_Z_NV 0x9354
#define GL_VIEWPORT_SWIZZLE_NEGATIVE_Z_NV 0x9355
#define GL_VIEWPORT_SWIZZLE_POSITIVE_W_NV 0x9356
#define GL_VIEWPORT_SWIZZLE_NEGATIVE_W_NV 0x9357
#define GL_VIEWPORT_SWIZZLE_X_NV 0x9358
#define GL_VIEWPORT_SWIZZLE_Y_NV 0x9359
#define GL_VIEWPORT_SWIZZLE_Z_NV 0x935A
#define GL_VIEWPORT_SWIZZLE_W_NV 0x935B
#define GL_PALETTE4_RGB8_OES 0x8B90
#define GL_PALETTE4_RGBA8_OES 0x8B91
#define GL_PALETTE4_R5_G6_B5_OES 0x8B92
#define GL_PALETTE4_RGBA4_OES 0x8B93
#define GL_PALETTE4_RGB5_A1_OES 0x8B94
#define GL_PALETTE8_RGB8_OES 0x8B95
#define GL_PALETTE8_RGBA8_OES 0x8B96
#define GL_PALETTE8_R5_G6_B5_OES 0x8B97
#define GL_PALETTE8_RGBA4_OES 0x8B98
#define GL_PALETTE8_RGB5_A1_OES 0x8B99
#define GL_FIXED_OES 0x140C
#define GL_IMPLEMENTATION_COLOR_READ_TYPE_OES 0x8B9A
#define GL_IMPLEMENTATION_COLOR_READ_FORMAT_OES 0x8B9B
#define GL_INTERLACE_OML 0x8980
#define GL_INTERLACE_READ_OML 0x8981
#define GL_PACK_RESAMPLE_OML 0x8984
#define GL_UNPACK_RESAMPLE_OML 0x8985
#define GL_RESAMPLE_REPLICATE_OML 0x8986
#define GL_RESAMPLE_ZERO_FILL_OML 0x8987
#define GL_RESAMPLE_AVERAGE_OML 0x8988
#define GL_RESAMPLE_DECIMATE_OML 0x8989
#define GL_FORMAT_SUBSAMPLE_24_24_OML 0x8982
#define GL_FORMAT_SUBSAMPLE_244_244_OML 0x8983
```

```
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_NUM_VIEWS_OVR 0x9630
#define GL_FRAMEBUFFER_ATTACHMENT_TEXTURE_BASE_VIEW_INDEX_OVR 0x9632
#define GL_MAX_VIEWS_OVR 0x9631
#define GL_FRAMEBUFFER_INCOMPLETE_VIEW_TARGETS_OVR 0x9633
#define GL_PREFER_DOUBLEBUFFER_HINT_PGI 0x1A1F8
#define GL_CONSERVE_MEMORY_HINT_PGI 0x1A1FD
#define GL_RECLAIM_MEMORY_HINT_PGI 0x1A1FE
#define GL_NATIVE_GRAPHICS_HANDLE_PGI 0x1A202
#define GL_NATIVE_GRAPHICS_BEGIN_HINT_PGI 0x1A203
#define GL_NATIVE_GRAPHICS_END_HINT_PGI 0x1A204
#define GL_ALWAYS_FAST_HINT_PGI 0x1A20C
#define GL_ALWAYS_SOFT_HINT_PGI 0x1A20D
#define GL_ALLOW_DRAW_OBJ_HINT_PGI 0x1A20E
#define GL_ALLOW_DRAW_WIN_HINT_PGI 0x1A20F
#define GL_ALLOW_DRAW_FRG_HINT_PGI 0x1A210
#define GL_ALLOW_DRAW_MEM_HINT_PGI 0x1A211
#define GL_STRICT_DEPTHFUNC_HINT_PGI 0x1A216
#define GL_STRICT_LIGHTING_HINT_PGI 0x1A217
#define GL_STRICT_SCISSOR_HINT_PGI 0x1A218
#define GL_FULL_STIPPLE_HINT_PGI 0x1A219
#define GL_CLIP_NEAR_HINT_PGI 0x1A220
#define GL_CLIP_FAR_HINT_PGI 0x1A221
#define GL_WIDE_LINE_HINT_PGI 0x1A222
#define GL_BACK_NORMALS_HINT_PGI 0x1A223
#define GL_VERTEX_DATA_HINT_PGI 0x1A22A
#define GL_VERTEX_CONSISTENT_HINT_PGI 0x1A22B
#define GL_MATERIAL_SIDE_HINT_PGI 0x1A22C
#define GL_MAX_VERTEX_HINT_PGI 0x1A22D
#define GL_COLOR3_BIT_PGI 0x00010000
#define GL_COLOR4_BIT_PGI 0x00020000
#define GL_EDGEFLAG_BIT_PGI 0x00040000
#define GL_INDEX_BIT_PGI 0x00080000
#define GL_MAT_AMBIENT_BIT_PGI 0x00100000
#define GL_MAT_AMBIENT_AND_DIFFUSE_BIT_PGI 0x00200000
#define GL_MAT_DIFFUSE_BIT_PGI 0x00400000
#define GL_MAT_EMISSION_BIT_PGI 0x00800000
#define GL_MAT_COLOR_INDEXES_BIT_PGI 0x01000000
#define GL_MAT_SHININESS_BIT_PGI 0x02000000
#define GL_MAT_SPECULAR_BIT_PGI 0x04000000
#define GL_NORMAL_BIT_PGI 0x08000000
#define GL_TEXCOORD1_BIT_PGI 0x10000000
#define GL_TEXCOORD2_BIT_PGI 0x20000000
#define GL_TEXCOORD3_BIT_PGI 0x40000000
#define GL_TEXCOORD4_BIT_PGI 0x80000000
#define GL_VERTEX23_BIT_PGI 0x00000004
#define GL_VERTEX4_BIT_PGI 0x00000008
#define GL_SCREEN_COORDINATES_REND 0x8490
#define GL_INVERTED_SCREEN_W_REND 0x8491
#define GL_RGB_S3TC 0x83A0
#define GL_RGB4_S3TC 0x83A1
#define GL_RGBA_S3TC 0x83A2
#define GL_RGBA4_S3TC 0x83A3
#define GL_RGBA_DXT5_S3TC 0x83A4
#define GL_RGBA4_DXT5_S3TC 0x83A5
#define GL_DETAIL_TEXTURE_2D_SGIS 0x8095
#define GL_DETAIL_TEXTURE_2D_BINDING_SGIS 0x8096
#define GL_LINEAR_DETAIL_SGIS 0x8097
#define GL_LINEAR_DETAIL_ALPHA_SGIS 0x8098
#define GL_LINEAR_DETAIL_COLOR_SGIS 0x8099
#define GL_DETAIL_TEXTURE_LEVEL_SGIS 0x809A
#define GL_DETAIL_TEXTURE_MODE_SGIS 0x809B
#define GL_DETAIL_TEXTURE_FUNC_POINTS_SGIS 0x809C
#define GL_FOG_FUNC_SGIS 0x812A
#define GL_FOG_FUNC_POINTS_SGIS 0x812B
#define GL_MAX_FOG_FUNC_POINTS_SGIS 0x812C
#define GL_GENERATE_MIPMAP_SGIS 0x8191
#define GL_GENERATE_MIPMAP_HINT_SGIS 0x8192
#define GL_MULTISAMPLE_SGIS 0x809D
#define GL_SAMPLE_ALPHA_TO_MASK_SGIS 0x809E
#define GL_SAMPLE_ALPHA_TO_ONE_SGIS 0x809F
#define GL_SAMPLE_MASK_SGIS 0x80A0
#define GL_1PASS_SGIS 0x80A1
#define GL_2PASS_0_SGIS 0x80A2
#define GL_2PASS_1_SGIS 0x80A3
#define GL_4PASS_0_SGIS 0x80A4
#define GL_4PASS_1_SGIS 0x80A5
#define GL_4PASS_2_SGIS 0x80A6
#define GL_4PASS_3_SGIS 0x80A7
#define GL_SAMPLE_BUFFERS_SGIS 0x80A8
#define GL_SAMPLES_SGIS 0x80A9
#define GL_SAMPLE_MASK_VALUE_SGIS 0x80AA
#define GL_SAMPLE_MASK_INVERT_SGIS 0x80AB
#define GL_SAMPLE_PATTERN_SGIS 0x80AC
#define GL_PIXEL_TEXTURE_SGIS 0x8353
#define GL_PIXEL_FRAGMENT_RGB_SOURCE_SGIS 0x8354
#define GL_PIXEL_FRAGMENT_ALPHA_SOURCE_SGIS 0x8355
#define GL_PIXEL_GROUP_COLOR_SGIS 0x8356
```

```
#define GL_EYE_DISTANCE_TO_POINT_SGIS 0x81F0
#define GL_OBJECT_DISTANCE_TO_POINT_SGIS 0x81F1
#define GL_EYE_DISTANCE_TO_LINE_SGIS 0x81F2
#define GL_OBJECT_DISTANCE_TO_LINE_SGIS 0x81F3
#define GL_EYE_POINT_SGIS 0x81F4
#define GL_OBJECT_POINT_SGIS 0x81F5
#define GL_EYE_LINE_SGIS 0x81F6
#define GL_OBJECT_LINE_SGIS 0x81F7
#define GL_POINT_SIZE_MIN_SGIS 0x8126
#define GL_POINT_SIZE_MAX_SGIS 0x8127
#define GL_POINT_FADE_THRESHOLD_SIZE_SGIS 0x8128
#define GL_DISTANCE_ATTENUATION_SGIS 0x8129
#define GL_LINEAR_SHARPEN_SGIS 0x80AD
#define GL_LINEAR_SHARPEN_ALPHA_SGIS 0x80AE
#define GL_LINEAR_SHARPEN_COLOR_SGIS 0x80AF
#define GL_SHARPEN_TEXTURE_FUNC_POINTS_SGIS 0x80B0
#define GL_PACK_SKIP_VOLUMES_SGIS 0x8130
#define GL_PACK_IMAGE_DEPTH_SGIS 0x8131
#define GL_UNPACK_SKIP_VOLUMES_SGIS 0x8132
#define GL_UNPACK_IMAGE_DEPTH_SGIS 0x8133
#define GL_TEXTURE_4D_SGIS 0x8134
#define GL_PROXY_TEXTURE_4D_SGIS 0x8135
#define GL_TEXTURE_4DSIZE_SGIS 0x8136
#define GL_TEXTURE_WRAP_Q_SGIS 0x8137
#define GL_MAX_4D_TEXTURE_SIZE_SGIS 0x8138
#define GL_TEXTURE_4D_BINDING_SGIS 0x814F
#define GL_CLAMP_TO_BORDER_SGIS 0x812D
#define GL_TEXTURE_COLOR_WRITEMASK_SGIS 0x81EF
#define GL_CLAMP_TO_EDGE_SGIS 0x812F
#define GL_FILTER4_SGIS 0x8146
#define GL_TEXTURE_FILTER4_SIZE_SGIS 0x8147
#define GL_TEXTURE_MIN_LOD_SGIS 0x813A
#define GL_TEXTURE_MAX_LOD_SGIS 0x813B
#define GL_TEXTURE_BASE_LEVEL_SGIS 0x813C
#define GL_TEXTURE_MAX_LEVEL_SGIS 0x813D
#define GL_DUAL_ALPHA4_SGIS 0x8110
#define GL_DUAL_ALPHA8_SGIS 0x8111
#define GL_DUAL_ALPHA12_SGIS 0x8112
#define GL_DUAL_ALPHA16_SGIS 0x8113
#define GL_DUAL_LUMINANCE4_SGIS 0x8114
#define GL_DUAL_LUMINANCE8_SGIS 0x8115
#define GL_DUAL_LUMINANCE12_SGIS 0x8116
#define GL_DUAL_LUMINANCE16_SGIS 0x8117
#define GL_DUAL_INTENSITY4_SGIS 0x8118
#define GL_DUAL_INTENSITY8_SGIS 0x8119
#define GL_DUAL_INTENSITY12_SGIS 0x811A
#define GL_DUAL_INTENSITY16_SGIS 0x811B
#define GL_DUAL_LUMINANCE_ALPHA4_SGIS 0x811C
#define GL_DUAL_LUMINANCE_ALPHA8_SGIS 0x811D
#define GL_QUAD_ALPHA4_SGIS 0x811E
#define GL_QUAD_ALPHA8_SGIS 0x811F
#define GL_QUAD_LUMINANCE4_SGIS 0x8120
#define GL_QUAD_LUMINANCE8_SGIS 0x8121
#define GL_QUAD_INTENSITY4_SGIS 0x8122
#define GL_QUAD_INTENSITY8_SGIS 0x8123
#define GL_DUAL_TEXTURE_SELECT_SGIS 0x8124
#define GL_QUAD_TEXTURE_SELECT_SGIS 0x8125
#define GL_ASYNC_MARKER_SGIX 0x8329
#define GL_ASYNC_HISTOGRAM_SGIX 0x832C
#define GL_MAX_ASYNC_HISTOGRAM_SGIX 0x832D
#define GL_ASYNC_TEX_IMAGE_SGIX 0x835C
#define GL_ASYNC_DRAW_PIXELS_SGIX 0x835D
#define GL_ASYNC_READ_PIXELS_SGIX 0x835E
#define GL_MAX_ASYNC_TEX_IMAGE_SGIX 0x835F
#define GL_MAX_ASYNC_DRAW_PIXELS_SGIX 0x8360
#define GL_MAX_ASYNC_READ_PIXELS_SGIX 0x8361
#define GL_ALPHA_MIN_SGIX 0x8320
#define GL_ALPHA_MAX_SGIX 0x8321
#define GL_CALLIGRAPHIC_FRAGMENT_SGIX 0x8183
#define GL_LINEAR_CLIPMAP_LINEAR_SGIX 0x8170
#define GL_TEXTURE_CLIPMAP_CENTER_SGIX 0x8171
#define GL_TEXTURE_CLIPMAP_FRAME_SGIX 0x8172
#define GL_TEXTURE_CLIPMAP_OFFSET_SGIX 0x8173
#define GL_TEXTURE_CLIPMAP_VIRTUAL_DEPTH_SGIX 0x8174
#define GL_TEXTURE_CLIPMAP_LOD_OFFSET_SGIX 0x8175
#define GL_TEXTURE_CLIPMAP_DEPTH_SGIX 0x8176
#define GL_MAX_CLIPMAP_DEPTH_SGIX 0x8177
#define GL_MAX_CLIPMAP_VIRTUAL_DEPTH_SGIX 0x8178
#define GL_NEAREST_CLIPMAP_NEAREST_SGIX 0x844D
#define GL_NEAREST_CLIPMAP_LINEAR_SGIX 0x844E
#define GL_LINEAR_CLIPMAP_NEAREST_SGIX 0x844F
#define GL_CONVOLUTION_HINT_SGIX 0x8316
#define GL_DEPTH_COMPONENT16_SGIX 0x81A5
#define GL_DEPTH_COMPONENT24_SGIX 0x81A6
#define GL_DEPTH_COMPONENT32_SGIX 0x81A7
#define GL_FOG_OFFSET_SGIX 0x8198
#define GL_FOG_OFFSET_VALUE_SGIX 0x8199
```

```
#define GL_FRAGMENT_LIGHTING_SGIX 0x8400
#define GL_FRAGMENT_COLOR_MATERIAL_SGIX 0x8401
#define GL_FRAGMENT_COLOR_MATERIAL_FACE_SGIX 0x8402
#define GL_FRAGMENT_COLOR_MATERIAL_PARAMETER_SGIX 0x8403
#define GL_MAX_FRAGMENT_LIGHTS_SGIX 0x8404
#define GL_MAX_ACTIVE_LIGHTS_SGIX 0x8405
#define GL_CURRENT_RASTER_NORMAL_SGIX 0x8406
#define GL_LIGHT_ENV_MODE_SGIX 0x8407
#define GL_FRAGMENT_LIGHT_MODEL_LOCAL_VIEWER_SGIX 0x8408
#define GL_FRAGMENT_LIGHT_MODEL_TWO_SIDE_SGIX 0x8409
#define GL_FRAGMENT_LIGHT_MODEL_AMBIENT_SGIX 0x840A
#define GL_FRAGMENT_LIGHT_MODEL_NORMAL_INTERPOLATION_SGIX 0x840B
#define GL_FRAGMENT_LIGHT0_SGIX 0x840C
#define GL_FRAGMENT_LIGHT1_SGIX 0x840D
#define GL_FRAGMENT_LIGHT2_SGIX 0x840E
#define GL_FRAGMENT_LIGHT3_SGIX 0x840F
#define GL_FRAGMENT_LIGHT4_SGIX 0x8410
#define GL_FRAGMENT_LIGHT5_SGIX 0x8411
#define GL_FRAGMENT_LIGHT6_SGIX 0x8412
#define GL_FRAGMENT_LIGHT7_SGIX 0x8413
#define GL_FRAMEZOOM_SGIX 0x818B
#define GL_FRAMEZOOM_FACTOR_SGIX 0x818C
#define GL_MAX_FRAMEZOOM_FACTOR_SGIX 0x818D
#define GL_INSTRUMENT_BUFFER_POINTER_SGIX 0x8180
#define GL_INSTRUMENT_MEASUREMENTS_SGIX 0x8181
#define GL_INTERLACE_SGIX 0x8094
#define GL_IR_INSTRUMENT1_SGIX 0x817F
#define GL_LIST_PRIORITY_SGIX 0x8182
#define GL_PIXEL_TEX_GEN_SGIX 0x8139
#define GL_PIXEL_TEX_GEN_MODE_SGIX 0x832B
#define GL_PIXEL_TILE_BEST_ALIGNMENT_SGIX 0x813E
#define GL_PIXEL_TILE_CACHE_INCREMENT_SGIX 0x813F
#define GL_PIXEL_TILE_WIDTH_SGIX 0x8140
#define GL_PIXEL_TILE_HEIGHT_SGIX 0x8141
#define GL_PIXEL_TILE_GRID_WIDTH_SGIX 0x8142
#define GL_PIXEL_TILE_GRID_HEIGHT_SGIX 0x8143
#define GL_PIXEL_TILE_GRID_DEPTH_SGIX 0x8144
#define GL_PIXEL_TILE_CACHE_SIZE_SGIX 0x8145
#define GL_TEXTURE_DEFORMATION_BIT_SGIX 0x00000001
#define GL_GEOMETRY_DEFORMATION_BIT_SGIX 0x00000002
#define GL_GEOMETRY_DEFORMATION_SGIX 0x8194
#define GL_TEXTURE_DEFORMATION_SGIX 0x8195
#define GL_DEFORMATIONS_MASK_SGIX 0x8196
#define GL_MAX_DEFORMATION_ORDER_SGIX 0x8197
#define GL_REFERENCE_PLANE_SGIX 0x817D
#define GL_REFERENCE_PLANE_EQUATION_SGIX 0x817E
#define GL_PACK_RESAMPLE_SGIX 0x842E
#define GL_UNPACK_RESAMPLE_SGIX 0x842F
#define GL_RESAMPLE_REPLICATE_SGIX 0x8433
#define GL_RESAMPLE_ZERO_FILL_SGIX 0x8434
#define GL_RESAMPLE_DECIMATE_SGIX 0x8430
#define GL_SCALEBIAS_HINT_SGIX 0x8322
#define GL_TEXTURE_COMPARE_SGIX 0x819A
#define GL_TEXTURE_COMPARE_OPERATOR_SGIX 0x819B
#define GL_TEXTURE_LEQUAL_R_SGIX 0x819C
#define GL_TEXTURE_GEQUAL_R_SGIX 0x819D
#define GL_SHADOW_AMBIENT_SGIX 0x80BF
#define GL_SPRITE_SGIX 0x8148
#define GL_SPRITE_MODE_SGIX 0x8149
#define GL_SPRITE_AXIS_SGIX 0x814A
#define GL_SPRITE_TRANSLATION_SGIX 0x814B
#define GL_SPRITE_AXIAL_SGIX 0x814C
#define GL_SPRITE_OBJECT_ALIGNED_SGIX 0x814D
#define GL_SPRITE_EYE_ALIGNED_SGIX 0x814E
#define GL_PACK_SUBSAMPLE_RATE_SGIX 0x85A0
#define GL_UNPACK_SUBSAMPLE_RATE_SGIX 0x85A1
#define GL_PIXEL_SUBSAMPLE_4444_SGIX 0x85A2
#define GL_PIXEL_SUBSAMPLE_2424_SGIX 0x85A3
#define GL_PIXEL_SUBSAMPLE_4242_SGIX 0x85A4
#define GL_TEXTURE_ENV_BIAS_SGIX 0x80BE
#define GL_TEXTURE_MAX_CLAMP_S_SGIX 0x8369
#define GL_TEXTURE_MAX_CLAMP_T_SGIX 0x836A
#define GL_TEXTURE_MAX_CLAMP_R_SGIX 0x836B
#define GL_TEXTURE_LOD_BIAS_S_SGIX 0x818E
#define GL_TEXTURE_LOD_BIAS_T_SGIX 0x818F
#define GL_TEXTURE_LOD_BIAS_R_SGIX 0x8190
#define GL_TEXTURE_MULTI_BUFFER_HINT_SGIX 0x812E
#define GL_POST_TEXTURE_FILTER_BIAS_SGIX 0x8179
#define GL_POST_TEXTURE_FILTER_SCALE_SGIX 0x817A
#define GL_POST_TEXTURE_FILTER_BIAS_RANGE_SGIX 0x817B
#define GL_POST_TEXTURE_FILTER_SCALE_RANGE_SGIX 0x817C
#define GL_VERTEX_PRECLIP_SGIX 0x83EE
#define GL_VERTEX_PRECLIP_HINT_SGIX 0x83EF
#define GL_YCRCB_422_SGIX 0x81BB
#define GL_YCRCB_444_SGIX 0x81BC
#define GL_YCRCB_SGIX 0x8318
#define GL_YCRCBA_SGIX 0x8319
```

```
#define GL_COLOR_MATRIX_SGI 0x80B1
#define GL_COLOR_MATRIX_STACK_DEPTH_SGI 0x80B2
#define GL_MAX_COLOR_MATRIX_STACK_DEPTH_SGI 0x80B3
#define GL_POST_COLOR_MATRIX_RED_SCALE_SGI 0x80B4
#define GL_POST_COLOR_MATRIX_GREEN_SCALE_SGI 0x80B5
#define GL_POST_COLOR_MATRIX_BLUE_SCALE_SGI 0x80B6
#define GL_POST_COLOR_MATRIX_ALPHA_SCALE_SGI 0x80B7
#define GL_POST_COLOR_MATRIX_RED_BIAS_SGI 0x80B8
#define GL_POST_COLOR_MATRIX_GREEN_BIAS_SGI 0x80B9
#define GL_POST_COLOR_MATRIX_BLUE_BIAS_SGI 0x80BA
#define GL_POST_COLOR_MATRIX_ALPHA_BIAS_SGI 0x80BB
#define GL_COLOR_TABLE_SGI 0x80D0
#define GL_POST_CONVOLUTION_COLOR_TABLE_SGI 0x80D1
#define GL_POST_COLOR_MATRIX_COLOR_TABLE_SGI 0x80D2
#define GL_PROXY_COLOR_TABLE_SGI 0x80D3
#define GL_PROXY_POST_CONVOLUTION_COLOR_TABLE_SGI 0x80D4
#define GL_PROXY_POST_COLOR_MATRIX_COLOR_TABLE_SGI 0x80D5
#define GL_COLOR_TABLE_SCALE_SGI 0x80D6
#define GL_COLOR_TABLE_BIAS_SGI 0x80D7
#define GL_COLOR_TABLE_FORMAT_SGI 0x80D8
#define GL_COLOR_TABLE_WIDTH_SGI 0x80D9
#define GL_COLOR_TABLE_RED_SIZE_SGI 0x80DA
#define GL_COLOR_TABLE_GREEN_SIZE_SGI 0x80DB
#define GL_COLOR_TABLE_BLUE_SIZE_SGI 0x80DC
#define GL_COLOR_TABLE_ALPHA_SIZE_SGI 0x80DD
#define GL_COLOR_TABLE_LUMINANCE_SIZE_SGI 0x80DE
#define GL_COLOR_TABLE_INTENSITY_SIZE_SGI 0x80DF
#define GL_TEXTURE_COLOR_TABLE_SGI 0x80BC
#define GL_PROXY_TEXTURE_COLOR_TABLE_SGI 0x80BD
#define GL_UNPACK_CONSTANT_DATA_SUNX 0x81D5
#define GL_TEXTURE_CONSTANT_DATA_SUNX 0x81D6
#define GL_WRAP_BORDER_SUN 0x81D4
#define GL_GLOBAL_ALPHA_SUN 0x81D9
#define GL_GLOBAL_ALPHA_FACTOR_SUN 0x81DA
#define GL_QUAD_MESH_SUN 0x8614
#define GL_TRIANGLE_MESH_SUN 0x8615
#define GL_SLICE_ACCUM_SUN 0x85CC
#define GL_RESTART_SUN 0x0001
#define GL_REPLACE_MIDDLE_SUN 0x0002
#define GL_REPLACE_OLDEST_SUN 0x0003
#define GL_TRIANGLE_LIST_SUN 0x81D7
#define GL_REPLACEMENT_CODE_SUN 0x81D8
#define GL_REPLACEMENT_CODE_ARRAY_SUN 0x85C0
#define GL_REPLACEMENT_CODE_ARRAY_TYPE_SUN 0x85C1
#define GL_REPLACEMENT_CODE_ARRAY_STRIDE_SUN 0x85C2
#define GL_REPLACEMENT_CODE_ARRAY_POINTER_SUN 0x85C3
#define GL_R1UI_V3F_SUN 0x85C4
#define GL_R1UI_C4UB_V3F_SUN 0x85C5
#define GL_R1UI_C3F_V3F_SUN 0x85C6
#define GL_R1UI_N3F_V3F_SUN 0x85C7
#define GL_R1UI_C4F_N3F_V3F_SUN 0x85C8
#define GL_R1UI_T2F_V3F_SUN 0x85C9
#define GL_R1UI_T2F_N3F_V3F_SUN 0x85CA
#define GL_R1UI_T2F_C4F_N3F_V3F_SUN 0x85CB
#define GL_PHONG_WIN 0x80EA
#define GL_PHONG_HINT_WIN 0x80EB
#define GL_FOG_SPECULAR_TEXTURE_WIN 0x80EC
#ifndef GL_3DFX_multisample
#define GL_3DFX_multisample 1
GLAPI int GLAD_GL_3DFX_multisample;
#endif
#ifndef GL_3DFX_tbuffer
#define GL_3DFX_tbuffer 1
GLAPI int GLAD_GL_3DFX_tbuffer;
typedef void (APIENTRYP PFNGLTBUFFERMASK3DFXPROC)(GLuint mask);
GLAPI PFNGLTBUFFERMASK3DFXPROC glad_glTbufferMask3DFX;
#define glTbufferMask3DFX glad_glTbufferMask3DFX
#endif
#ifndef GL_3DFX_texture_compression_FXT1
#define GL_3DFX_texture_compression_FXT1 1
GLAPI int GLAD_GL_3DFX_texture_compression_FXT1;
#endif
#ifndef GL_AMD_blend_minmax_factor
#define GL_AMD_blend_minmax_factor 1
GLAPI int GLAD_GL_AMD_blend_minmax_factor;
#endif
#ifndef GL_AMD_conservative_depth
#define GL_AMD_conservative_depth 1
GLAPI int GLAD_GL_AMD_conservative_depth;
#endif
#ifndef GL_AMD_debug_output
#define GL_AMD_debug_output 1
GLAPI int GLAD_GL_AMD_debug_output;
typedef void (APIENTRYP PFNGLDEBUGMESSAGEENABLEAMDPROC)(GLenum category, GLenum severity, GLsizei
count, const GLuint *ids, GLboolean enabled);
GLAPI PFNGLDEBUGMESSAGEENABLEAMDPROC glad_glDebugMessageEnableAMD;
#define glDebugMessageEnableAMD glad_glDebugMessageEnableAMD
```

```
typedef void (APIENTRYP PFNGLDEBUGMESSAGEINSERTAMDPROC)(GLenum category, GLenum severity, GLuint id, GLsizei length, const GLchar *buf);
GLAPI PFNGLDEBUGMESSAGEINSERTAMDPROC glad_glDebugMessageInsertAMD;
#define glDebugMessageInsertAMD glad_glDebugMessageInsertAMD
typedef void (APIENTRYP PFNGLDEBUGMESSAGECALLBACKAMDPROC)(GLDEBUGPROCAMD callback, void *userParam);
GLAPI PFNGLDEBUGMESSAGECALLBACKAMDPROC glad_glDebugMessageCallbackAMD;
#define glDebugMessageCallbackAMD glad_glDebugMessageCallbackAMD
typedef GLuint (APIENTRYP PFNGLGETDEBUGMESSAGELOGAMDPROC)(GLuint count, GLsizei bufSize, GLenum *categories, GLenum *severities, GLuint *ids, GLsizei *lengths, GLchar *message);
GLAPI PFNGLGETDEBUGMESSAGELOGAMDPROC glad_glGetDebugMessageLogAMD;
#define glGetDebugMessageLogAMD glad_glGetDebugMessageLogAMD
#endif
#ifndef GL_AMD_depth_clamp_separate
#define GL_AMD_depth_clamp_separate 1
GLAPI int GLAD_GL_AMD_depth_clamp_separate;
#endif
#ifndef GL_AMD_draw_buffers_blend
#define GL_AMD_draw_buffers_blend 1
GLAPI int GLAD_GL_AMD_draw_buffers_blend;
typedef void (APIENTRYP PFNGLBLENDFUNCINDEXEDAMDPROC)(GLuint buf, GLenum src, GLenum dst);
GLAPI PFNGLBLENDFUNCINDEXEDAMDPROC glad_glBlendFuncIndexedAMD;
#define glBlendFuncIndexedAMD glad_glBlendFuncIndexedAMD
typedef void (APIENTRYP PFNGLBLENDFUNCSEPARATEINDEXEDAMDPROC)(GLuint buf, GLenum srcRGB, GLenum dstRGB, GLenum srcAlpha, GLenum dstAlpha);
GLAPI PFNGLBLENDFUNCSEPARATEINDEXEDAMDPROC glad_glBlendFuncSeparateIndexedAMD;
#define glBlendFuncSeparateIndexedAMD glad_glBlendFuncSeparateIndexedAMD
typedef void (APIENTRYP PFNGLBLENDEQUATIONINDEXEDAMDPROC)(GLuint buf, GLenum mode);
GLAPI PFNGLBLENDEQUATIONINDEXEDAMDPROC glad_glBlendEquationIndexedAMD;
#define glBlendEquationIndexedAMD glad_glBlendEquationIndexedAMD
typedef void (APIENTRYP PFNGLBLENDEQUATIONSEPARATEINDEXEDAMDPROC)(GLuint buf, GLenum modeRGB, GLenum modeAlpha);
GLAPI PFNGLBLENDEQUATIONSEPARATEINDEXEDAMDPROC glad_glBlendEquationSeparateIndexedAMD;
#define glBlendEquationSeparateIndexedAMD glad_glBlendEquationSeparateIndexedAMD
#endif
#ifndef GL_AMD_framebuffer_multisample_advanced
#define GL_AMD_framebuffer_multisample_advanced 1
GLAPI int GLAD_GL_AMD_framebuffer_multisample_advanced;
typedef void (APIENTRYP PFNGLRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC)(GLenum target, GLsizei samples, GLsizei storageSamples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC glad_glRenderbufferStorageMultisampleAdvancedAMD;
#define glRenderbufferStorageMultisampleAdvancedAMD glad_glRenderbufferStorageMultisampleAdvancedAMD
typedef void (APIENTRYP PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC)(GLuint renderbuffer, GLsizei samples, GLsizei storageSamples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEADVANCEDAMDPROC glad_glNamedRenderbufferStorageMultisampleAdvancedAMD;
#define glNamedRenderbufferStorageMultisampleAdvancedAMD glad_glNamedRenderbufferStorageMultisampleAdvancedAMD
#endif
#ifndef GL_AMD_framebuffer_sample_positions
#define GL_AMD_framebuffer_sample_positions 1
GLAPI int GLAD_GL_AMD_framebuffer_sample_positions;
typedef void (APIENTRYP PFNGLFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC)(GLenum target, GLuint numsamples, GLuint pixelindex, const GLfloat *values);
GLAPI PFNGLFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC glad_glFramebufferSamplePositionsfvAMD;
#define glFramebufferSamplePositionsfvAMD glad_glFramebufferSamplePositionsfvAMD
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC)(GLuint framebuffer, GLuint numsamples, GLuint pixelindex, const GLfloat *values);
GLAPI PFNGLNAMEDFRAMEBUFFERSAMPLEPOSITIONSFVAMDPROC glad_glNamedFramebufferSamplePositionsfvAMD;
#define glNamedFramebufferSamplePositionsfvAMD glad_glNamedFramebufferSamplePositionsfvAMD
typedef void (APIENTRYP PFNGLGETFRAMEBUFFERPARAMETERFVAMDPROC)(GLenum target, GLenum pname, GLuint numsamples, GLuint pixelindex, GLsizei size, GLfloat *values);
GLAPI PFNGLGETFRAMEBUFFERPARAMETERFVAMDPROC glad_glGetFramebufferParameterfvAMD;
#define glGetFramebufferParameterfvAMD glad_glGetFramebufferParameterfvAMD
typedef void (APIENTRYP PFNGLGETNAMEDFRAMEBUFFERPARAMETERFVAMDPROC)(GLuint framebuffer, GLenum pname, GLuint numsamples, GLuint pixelindex, GLsizei size, GLfloat *values);
GLAPI PFNGLGETNAMEDFRAMEBUFFERPARAMETERFVAMDPROC glad_glGetNamedFramebufferParameterfvAMD;
#define glGetNamedFramebufferParameterfvAMD glad_glGetNamedFramebufferParameterfvAMD
#endif
#ifndef GL_AMD_gcn_shader
#define GL_AMD_gcn_shader 1
GLAPI int GLAD_GL_AMD_gcn_shader;
#endif
#ifndef GL_AMD_gpu_shader_half_float
#define GL_AMD_gpu_shader_half_float 1
GLAPI int GLAD_GL_AMD_gpu_shader_half_float;
#endif
#ifndef GL_AMD_gpu_shader_int16
#define GL_AMD_gpu_shader_int16 1
GLAPI int GLAD_GL_AMD_gpu_shader_int16;
#endif
#ifndef GL_AMD_gpu_shader_int64
#define GL_AMD_gpu_shader_int64 1
GLAPI int GLAD_GL_AMD_gpu_shader_int64;
typedef void (APIENTRYP PFNGLUNIFORM1I64NVPROC)(GLint location, GLint64EXT x);
GLAPI PFNGLUNIFORM1I64NVPROC glad_glUniform1i64NV;
#define glUniform1i64NV glad_glUniform1i64NV
```

```
typedef void (APIENTRYP PFNGLUNIFORM2I64NVPROC)(GLint location, GLint64EXT x, GLint64EXT y);
GLAPI PFNGLUNIFORM2I64NVPROC glad_glUniform2i64NV;
#define glUniform2i64NV glad_glUniform2i64NV
typedef void (APIENTRYP PFNGLUNIFORM3I64NVPROC)(GLint location, GLint64EXT x, GLint64EXT y, GLint64EXT z);
GLAPI PFNGLUNIFORM3I64NVPROC glad_glUniform3i64NV;
#define glUniform3i64NV glad_glUniform3i64NV
typedef void (APIENTRYP PFNGLUNIFORM4I64NVPROC)(GLint location, GLint64EXT x, GLint64EXT y, GLint64EXT z, GLint64EXT w);
GLAPI PFNGLUNIFORM4I64NVPROC glad_glUniform4i64NV;
#define glUniform4i64NV glad_glUniform4i64NV
typedef void (APIENTRYP PFNGLUNIFORM1I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLUNIFORM1I64VNVPROC glad_glUniform1i64vNV;
#define glUniform1i64vNV glad_glUniform1i64vNV
typedef void (APIENTRYP PFNGLUNIFORM2I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLUNIFORM2I64VNVPROC glad_glUniform2i64vNV;
#define glUniform2i64vNV glad_glUniform2i64vNV
typedef void (APIENTRYP PFNGLUNIFORM3I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLUNIFORM3I64VNVPROC glad_glUniform3i64vNV;
#define glUniform3i64vNV glad_glUniform3i64vNV
typedef void (APIENTRYP PFNGLUNIFORM4I64VNVPROC)(GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLUNIFORM4I64VNVPROC glad_glUniform4i64vNV;
#define glUniform4i64vNV glad_glUniform4i64vNV
typedef void (APIENTRYP PFNGLUNIFORM1UI64NVPROC)(GLint location, GLuint64EXT x);
GLAPI PFNGLUNIFORM1UI64NVPROC glad_glUniform1ui64NV;
#define glUniform1ui64NV glad_glUniform1ui64NV
typedef void (APIENTRYP PFNGLUNIFORM2UI64NVPROC)(GLint location, GLuint64EXT x, GLuint64EXT y);
GLAPI PFNGLUNIFORM2UI64NVPROC glad_glUniform2ui64NV;
#define glUniform2ui64NV glad_glUniform2ui64NV
typedef void (APIENTRYP PFNGLUNIFORM3UI64NVPROC)(GLint location, GLuint64EXT x, GLuint64EXT y, GLuint64EXT z);
GLAPI PFNGLUNIFORM3UI64NVPROC glad_glUniform3ui64NV;
#define glUniform3ui64NV glad_glUniform3ui64NV
typedef void (APIENTRYP PFNGLUNIFORM4UI64NVPROC)(GLint location, GLuint64EXT x, GLuint64EXT y, GLuint64EXT z, GLuint64EXT w);
GLAPI PFNGLUNIFORM4UI64NVPROC glad_glUniform4ui64NV;
#define glUniform4ui64NV glad_glUniform4ui64NV
typedef void (APIENTRYP PFNGLUNIFORM1UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLUNIFORM1UI64VNVPROC glad_glUniform1ui64vNV;
#define glUniform1ui64vNV glad_glUniform1ui64vNV
typedef void (APIENTRYP PFNGLUNIFORM2UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLUNIFORM2UI64VNVPROC glad_glUniform2ui64vNV;
#define glUniform2ui64vNV glad_glUniform2ui64vNV
typedef void (APIENTRYP PFNGLUNIFORM3UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLUNIFORM3UI64VNVPROC glad_glUniform3ui64vNV;
#define glUniform3ui64vNV glad_glUniform3ui64vNV
typedef void (APIENTRYP PFNGLUNIFORM4UI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLUNIFORM4UI64VNVPROC glad_glUniform4ui64vNV;
#define glUniform4ui64vNV glad_glUniform4ui64vNV
typedef void (APIENTRYP PFNGLGETUNIFORMI64VNVPROC)(GLuint program, GLint location, GLint64EXT *params);
GLAPI PFNGLGETUNIFORMI64VNVPROC glad_glGetUniformi64vNV;
#define glGetUniformi64vNV glad_glGetUniformi64vNV
typedef void (APIENTRYP PFNGLGETUNIFORMUI64VNVPROC)(GLuint program, GLint location, GLuint64EXT *params);
GLAPI PFNGLGETUNIFORMUI64VNVPROC glad_glGetUniformui64vNV;
#define glGetUniformui64vNV glad_glGetUniformui64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1I64NVPROC)(GLuint program, GLint location, GLint64EXT x);
GLAPI PFNGLPROGRAMUNIFORM1I64NVPROC glad_glProgramUniform1i64NV;
#define glProgramUniform1i64NV glad_glProgramUniform1i64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2I64NVPROC)(GLuint program, GLint location, GLint64EXT x, GLint64EXT y);
GLAPI PFNGLPROGRAMUNIFORM2I64NVPROC glad_glProgramUniform2i64NV;
#define glProgramUniform2i64NV glad_glProgramUniform2i64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3I64NVPROC)(GLuint program, GLint location, GLint64EXT x, GLint64EXT y, GLint64EXT z);
GLAPI PFNGLPROGRAMUNIFORM3I64NVPROC glad_glProgramUniform3i64NV;
#define glProgramUniform3i64NV glad_glProgramUniform3i64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4I64NVPROC)(GLuint program, GLint location, GLint64EXT x, GLint64EXT y, GLint64EXT z, GLint64EXT w);
GLAPI PFNGLPROGRAMUNIFORM4I64NVPROC glad_glProgramUniform4i64NV;
#define glProgramUniform4i64NV glad_glProgramUniform4i64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1I64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM1I64VNVPROC glad_glProgramUniform1i64vNV;
#define glProgramUniform1i64vNV glad_glProgramUniform1i64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2I64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM2I64VNVPROC glad_glProgramUniform2i64vNV;
```

```
#define glProgramUniform2i64vNV glad_glProgramUniform2i64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3I64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM3I64VNVPROC glad_glProgramUniform3i64vNV;
#define glProgramUniform3i64vNV glad_glProgramUniform3i64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4I64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM4I64VNVPROC glad_glProgramUniform4i64vNV;
#define glProgramUniform4i64vNV glad_glProgramUniform4i64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UI64NVPROC)(GLuint program, GLint location, GLuint64EXT x);
GLAPI PFNGLPROGRAMUNIFORM1UI64NVPROC glad_glProgramUniform1ui64NV;
#define glProgramUniform1ui64NV glad_glProgramUniform1ui64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UI64NVPROC)(GLuint program, GLint location, GLuint64EXT x, GLuint64EXT y);
GLAPI PFNGLPROGRAMUNIFORM2UI64NVPROC glad_glProgramUniform2ui64NV;
#define glProgramUniform2ui64NV glad_glProgramUniform2ui64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UI64NVPROC)(GLuint program, GLint location, GLuint64EXT x, GLuint64EXT y, GLuint64EXT z);
GLAPI PFNGLPROGRAMUNIFORM3UI64NVPROC glad_glProgramUniform3ui64NV;
#define glProgramUniform3ui64NV glad_glProgramUniform3ui64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UI64NVPROC)(GLuint program, GLint location, GLuint64EXT x, GLuint64EXT y, GLuint64EXT z, GLuint64EXT w);
GLAPI PFNGLPROGRAMUNIFORM4UI64NVPROC glad_glProgramUniform4ui64NV;
#define glProgramUniform4ui64NV glad_glProgramUniform4ui64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UI64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM1UI64VNVPROC glad_glProgramUniform1ui64vNV;
#define glProgramUniform1ui64vNV glad_glProgramUniform1ui64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UI64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM2UI64VNVPROC glad_glProgramUniform2ui64vNV;
#define glProgramUniform2ui64vNV glad_glProgramUniform2ui64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UI64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM3UI64VNVPROC glad_glProgramUniform3ui64vNV;
#define glProgramUniform3ui64vNV glad_glProgramUniform3ui64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UI64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORM4UI64VNVPROC glad_glProgramUniform4ui64vNV;
#define glProgramUniform4ui64vNV glad_glProgramUniform4ui64vNV
#endif
#ifndef GL_AMD_interleaved_elements
#define GL_AMD_interleaved_elements 1
GLAPI int GLAD_GL_AMD_interleaved_elements;
typedef void (APIENTRYP PFNGLVERTEXATTRIBPARAMETERIAMDPROC)(GLuint index, GLenum pname, GLint param);
GLAPI PFNGLVERTEXATTRIBPARAMETERIAMDPROC glad_glVertexAttribParameteriAMD;
#define glVertexAttribParameteriAMD glad_glVertexAttribParameteriAMD
#endif
#ifndef GL_AMD_multi_draw_indirect
#define GL_AMD_multi_draw_indirect 1
GLAPI int GLAD_GL_AMD_multi_draw_indirect;
typedef void (APIENTRYP PFNGLMULTIDRAWARRAYSINDIRECTAMDPROC)(GLenum mode, const void *indirect, GLsizei primcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWARRAYSINDIRECTAMDPROC glad_glMultiDrawArraysIndirectAMD;
#define glMultiDrawArraysIndirectAMD glad_glMultiDrawArraysIndirectAMD
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSINDIRECTAMDPROC)(GLenum mode, GLenum type, const void *indirect, GLsizei primcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTAMDPROC glad_glMultiDrawElementsIndirectAMD;
#define glMultiDrawElementsIndirectAMD glad_glMultiDrawElementsIndirectAMD
#endif
#ifndef GL_AMD_name_gen_delete
#define GL_AMD_name_gen_delete 1
GLAPI int GLAD_GL_AMD_name_gen_delete;
typedef void (APIENTRYP PFNGLGENNAMESAMDPROC)(GLenum identifier, GLuint num, GLuint *names);
GLAPI PFNGLGENNAMESAMDPROC glad_glGenNamesAMD;
#define glGenNamesAMD glad_glGenNamesAMD
typedef void (APIENTRYP PFNGLDELETENAMESAMDPROC)(GLenum identifier, GLuint num, const GLuint *names);
GLAPI PFNGLDELETENAMESAMDPROC glad_glDeleteNamesAMD;
#define glDeleteNamesAMD glad_glDeleteNamesAMD
typedef GLboolean (APIENTRYP PFNGLISNAMEAMDPROC)(GLenum identifier, GLuint name);
GLAPI PFNGLISNAMEAMDPROC glad_glIsNameAMD;
#define glIsNameAMD glad_glIsNameAMD
#endif
#ifndef GL_AMD_occlusion_query_event
#define GL_AMD_occlusion_query_event 1
GLAPI int GLAD_GL_AMD_occlusion_query_event;
typedef void (APIENTRYP PFNGLQUERYOBJECTPARAMETERUIAMDPROC)(GLenum target, GLuint id, GLenum pname, GLuint param);
GLAPI PFNGLQUERYOBJECTPARAMETERUIAMDPROC glad_glQueryObjectParameteruiAMD;
#define glQueryObjectParameteruiAMD glad_glQueryObjectParameteruiAMD
#endif
#ifndef GL_AMD_performance_monitor
#define GL_AMD_performance_monitor 1
GLAPI int GLAD_GL_AMD_performance_monitor;
typedef void (APIENTRYP PFNGLGETPERFMONITORGROUPSAMDPROC)(GLint *numGroups, GLsizei groupsSize, GLuint *groups);
```

```
GLAPI PFNGLGETPERFMONITORGROUPSAMDPROC glad_glGetPerfMonitorGroupsAMD;
#define glGetPerfMonitorGroupsAMD glad_glGetPerfMonitorGroupsAMD
typedef void (APIENTRYP PFNGLGETPERFMONITORCOUNTERSAMDPROC)(GLuint group, GLint *numCounters, GLint *maxActiveCounters, GLsizei counterSize, GLuint *counters);
GLAPI PFNGLGETPERFMONITORCOUNTERSAMDPROC glad_glGetPerfMonitorCountersAMD;
#define glGetPerfMonitorCountersAMD glad_glGetPerfMonitorCountersAMD
typedef void (APIENTRYP PFNGLGETPERFMONITORGROUPSTRINGAMDPROC)(GLuint group, GLsizei bufSize, GLsizei *length, GLchar *groupString);
GLAPI PFNGLGETPERFMONITORGROUPSTRINGAMDPROC glad_glGetPerfMonitorGroupStringAMD;
#define glGetPerfMonitorGroupStringAMD glad_glGetPerfMonitorGroupStringAMD
typedef void (APIENTRYP PFNGLGETPERFMONITORCOUNTERSTRINGAMDPROC)(GLuint group, GLuint counter, GLsizei bufSize, GLsizei *length, GLchar *counterString);
GLAPI PFNGLGETPERFMONITORCOUNTERSTRINGAMDPROC glad_glGetPerfMonitorCounterStringAMD;
#define glGetPerfMonitorCounterStringAMD glad_glGetPerfMonitorCounterStringAMD
typedef void (APIENTRYP PFNGLGETPERFMONITORCOUNTERINFOAMDPROC)(GLuint group, GLuint counter, GLenum pname, void *data);
GLAPI PFNGLGETPERFMONITORCOUNTERINFOAMDPROC glad_glGetPerfMonitorCounterInfoAMD;
#define glGetPerfMonitorCounterInfoAMD glad_glGetPerfMonitorCounterInfoAMD
typedef void (APIENTRYP PFNGLGENPERFMONITORSAMDPROC)(GLsizei n, GLuint *monitors);
GLAPI PFNGLGENPERFMONITORSAMDPROC glad_glGenPerfMonitorsAMD;
#define glGenPerfMonitorsAMD glad_glGenPerfMonitorsAMD
typedef void (APIENTRYP PFNGLDELETEPERFMONITORSAMDPROC)(GLsizei n, GLuint *monitors);
GLAPI PFNGLDELETEPERFMONITORSAMDPROC glad_glDeletePerfMonitorsAMD;
#define glDeletePerfMonitorsAMD glad_glDeletePerfMonitorsAMD
typedef void (APIENTRYP PFNGLSELECTPERFMONITORCOUNTERSAMDPROC)(GLuint monitor, GLboolean enable, GLuint group, GLint numCounters, GLuint *counterList);
GLAPI PFNGLSELECTPERFMONITORCOUNTERSAMDPROC glad_glSelectPerfMonitorCountersAMD;
#define glSelectPerfMonitorCountersAMD glad_glSelectPerfMonitorCountersAMD
typedef void (APIENTRYP PFNGLBEGINPERFMONITORAMDPROC)(GLuint monitor);
GLAPI PFNGLBEGINPERFMONITORAMDPROC glad_glBeginPerfMonitorAMD;
#define glBeginPerfMonitorAMD glad_glBeginPerfMonitorAMD
typedef void (APIENTRYP PFNGLENDPERFMONITORAMDPROC)(GLuint monitor);
GLAPI PFNGLENDPERFMONITORAMDPROC glad_glEndPerfMonitorAMD;
#define glEndPerfMonitorAMD glad_glEndPerfMonitorAMD
typedef void (APIENTRYP PFNGLGETPERFMONITORCOUNTERDATAAMDPROC)(GLuint monitor, GLenum pname, GLsizei dataSize, GLuint *data, GLint *bytesWritten);
GLAPI PFNGLGETPERFMONITORCOUNTERDATAAMDPROC glad_glGetPerfMonitorCounterDataAMD;
#define glGetPerfMonitorCounterDataAMD glad_glGetPerfMonitorCounterDataAMD
#endif
#ifndef GL_AMD_pinned_memory
#define GL_AMD_pinned_memory 1
GLAPI int GLAD_GL_AMD_pinned_memory;
#endif
#ifndef GL_AMD_query_buffer_object
#define GL_AMD_query_buffer_object 1
GLAPI int GLAD_GL_AMD_query_buffer_object;
#endif
#ifndef GL_AMD_sample_positions
#define GL_AMD_sample_positions 1
GLAPI int GLAD_GL_AMD_sample_positions;
typedef void (APIENTRYP PFNGLSETMULTISAMPLEFVAMDPROC)(GLenum pname, GLuint index, const GLfloat *val);
GLAPI PFNGLSETMULTISAMPLEFVAMDPROC glad_glSetMultisamplefvAMD;
#define glSetMultisamplefvAMD glad_glSetMultisamplefvAMD
#endif
#ifndef GL_AMD_seamless_cubemap_per_texture
#define GL_AMD_seamless_cubemap_per_texture 1
GLAPI int GLAD_GL_AMD_seamless_cubemap_per_texture;
#endif
#ifndef GL_AMD_shader_atomic_counter_ops
#define GL_AMD_shader_atomic_counter_ops 1
GLAPI int GLAD_GL_AMD_shader_atomic_counter_ops;
#endif
#ifndef GL_AMD_shader_ballot
#define GL_AMD_shader_ballot 1
GLAPI int GLAD_GL_AMD_shader_ballot;
#endif
#ifndef GL_AMD_shader_explicit_vertex_parameter
#define GL_AMD_shader_explicit_vertex_parameter 1
GLAPI int GLAD_GL_AMD_shader_explicit_vertex_parameter;
#endif
#ifndef GL_AMD_shader_gpu_shader_half_float_fetch
#define GL_AMD_shader_gpu_shader_half_float_fetch 1
GLAPI int GLAD_GL_AMD_shader_gpu_shader_half_float_fetch;
#endif
#ifndef GL_AMD_shader_image_load_store_lod
#define GL_AMD_shader_image_load_store_lod 1
GLAPI int GLAD_GL_AMD_shader_image_load_store_lod;
#endif
#ifndef GL_AMD_shader_stencil_export
#define GL_AMD_shader_stencil_export 1
GLAPI int GLAD_GL_AMD_shader_stencil_export;
#endif
#ifndef GL_AMD_shader_trinary_minmax
#define GL_AMD_shader_trinary_minmax 1
GLAPI int GLAD_GL_AMD_shader_trinary_minmax;
#endif
#ifndef GL_AMD_sparse_texture
```

```
#define GL_AMD_sparse_texture 1
GLAPI int GLAD_GL_AMD_sparse_texture;
typedef void (APIENTRYP PFNGLTEXSTORAGESPARSEAMDPROC)(GLenum target, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLsizei layers, GLbitfield flags);
GLAPI PFNGLTEXSTORAGESPARSEAMDPROC glad_glTexStorageSparseAMD;
#define glTexStorageSparseAMD glad_glTexStorageSparseAMD
typedef void (APIENTRYP PFNGLTEXTURESTORAGESPARSEAMDPROC)(GLuint texture, GLenum target, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLsizei layers, GLbitfield flags);
GLAPI PFNGLTEXTURESTORAGESPARSEAMDPROC glad_glTextureStorageSparseAMD;
#define glTextureStorageSparseAMD glad_glTextureStorageSparseAMD
#endif
#ifndef GL_AMD_stencil_operation_extended
#define GL_AMD_stencil_operation_extended 1
GLAPI int GLAD_GL_AMD_stencil_operation_extended;
typedef void (APIENTRYP PFNGLSTENCILOPVALUEAMDPROC)(GLenum face, GLuint value);
GLAPI PFNGLSTENCILOPVALUEAMDPROC glad_glStencilOpValueAMD;
#define glStencilOpValueAMD glad_glStencilOpValueAMD
#endif
#ifndef GL_AMD_texture_gather_bias_lod
#define GL_AMD_texture_gather_bias_lod 1
GLAPI int GLAD_GL_AMD_texture_gather_bias_lod;
#endif
#ifndef GL_AMD_texture_texture4
#define GL_AMD_texture_texture4 1
GLAPI int GLAD_GL_AMD_texture_texture4;
#endif
#ifndef GL_AMD_transform_feedback3_lines_triangles
#define GL_AMD_transform_feedback3_lines_triangles 1
GLAPI int GLAD_GL_AMD_transform_feedback3_lines_triangles;
#endif
#ifndef GL_AMD_transform_feedback4
#define GL_AMD_transform_feedback4 1
GLAPI int GLAD_GL_AMD_transform_feedback4;
#endif
#ifndef GL_AMD_vertex_shader_layer
#define GL_AMD_vertex_shader_layer 1
GLAPI int GLAD_GL_AMD_vertex_shader_layer;
#endif
#ifndef GL_AMD_vertex_shader_tessellator
#define GL_AMD_vertex_shader_tessellator 1
GLAPI int GLAD_GL_AMD_vertex_shader_tessellator;
typedef void (APIENTRYP PFNGLTESSELLATIONFACTORAMDPROC)(GLfloat factor);
GLAPI PFNGLTESSELLATIONFACTORAMDPROC glad_glTessellationFactorAMD;
#define glTessellationFactorAMD glad_glTessellationFactorAMD
typedef void (APIENTRYP PFNGLTESSELLATIONMODEAMDPROC)(GLenum mode);
GLAPI PFNGLTESSELLATIONMODEAMDPROC glad_glTessellationModeAMD;
#define glTessellationModeAMD glad_glTessellationModeAMD
#endif
#ifndef GL_AMD_vertex_shader_viewport_index
#define GL_AMD_vertex_shader_viewport_index 1
GLAPI int GLAD_GL_AMD_vertex_shader_viewport_index;
#endif
#ifndef GL_APPLE_aux_depth_stencil
#define GL_APPLE_aux_depth_stencil 1
GLAPI int GLAD_GL_APPLE_aux_depth_stencil;
#endif
#ifndef GL_APPLE_client_storage
#define GL_APPLE_client_storage 1
GLAPI int GLAD_GL_APPLE_client_storage;
#endif
#ifndef GL_APPLE_element_array
#define GL_APPLE_element_array 1
GLAPI int GLAD_GL_APPLE_element_array;
typedef void (APIENTRYP PFNGLELEMENTPOINTERAPPLEPROC)(GLenum type, const void *pointer);
GLAPI PFNGLELEMENTPOINTERAPPLEPROC glad_glElementPointerAPPLE;
#define glElementPointerAPPLE glad_glElementPointerAPPLE
typedef void (APIENTRYP PFNGLDRAWELEMENTARRAYAPPLEPROC)(GLenum mode, GLint first, GLsizei count);
GLAPI PFNGLDRAWELEMENTARRAYAPPLEPROC glad_glDrawElementArrayAPPLE;
#define glDrawElementArrayAPPLE glad_glDrawElementArrayAPPLE
typedef void (APIENTRYP PFNGLDRAWRANGEELEMENTARRAYAPPLEPROC)(GLenum mode, GLuint start, GLuint end, GLint first, GLsizei count);
GLAPI PFNGLDRAWRANGEELEMENTARRAYAPPLEPROC glad_glDrawRangeElementArrayAPPLE;
#define glDrawRangeElementArrayAPPLE glad_glDrawRangeElementArrayAPPLE
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTARRAYAPPLEPROC)(GLenum mode, const GLint *first, const GLsizei *count, GLsizei primcount);
GLAPI PFNGLMULTIDRAWELEMENTARRAYAPPLEPROC glad_glMultiDrawElementArrayAPPLE;
#define glMultiDrawElementArrayAPPLE glad_glMultiDrawElementArrayAPPLE
typedef void (APIENTRYP PFNGLMULTIDRAWRANGEELEMENTARRAYAPPLEPROC)(GLenum mode, GLuint start, GLuint end, const GLint *first, const GLsizei *count, GLsizei primcount);
GLAPI PFNGLMULTIDRAWRANGEELEMENTARRAYAPPLEPROC glad_glMultiDrawRangeElementArrayAPPLE;
#define glMultiDrawRangeElementArrayAPPLE glad_glMultiDrawRangeElementArrayAPPLE
#endif
#ifndef GL_APPLE_fence
#define GL_APPLE_fence 1
GLAPI int GLAD_GL_APPLE_fence;
typedef void (APIENTRYP PFNGLGENFENCESAPPLEPROC)(GLsizei n, GLuint *fences);
GLAPI PFNGLGENFENCESAPPLEPROC glad_glGenFencesAPPLE;
```

```
#define glGenFencesAPPLE glad_glGenFencesAPPLE
typedef void (APIENTRYP PFNGLDELETEFENCESAPPLEPROC)(GLsizei n, const GLuint *fences);
GLAPI PFNGLDELETEFENCESAPPLEPROC glad_glDeleteFencesAPPLE;
#define glDeleteFencesAPPLE glad_glDeleteFencesAPPLE
typedef void (APIENTRYP PFNGLSETFENCEAPPLEPROC)(GLuint fence);
GLAPI PFNGLSETFENCEAPPLEPROC glad_glSetFenceAPPLE;
#define glSetFenceAPPLE glad_glSetFenceAPPLE
typedef GLboolean (APIENTRYP PFNGLISFENCEAPPLEPROC)(GLuint fence);
GLAPI PFNGLISFENCEAPPLEPROC glad_glIsFenceAPPLE;
#define glIsFenceAPPLE glad_glIsFenceAPPLE
typedef GLboolean (APIENTRYP PFNGLTESTFENCEAPPLEPROC)(GLuint fence);
GLAPI PFNGLTESTFENCEAPPLEPROC glad_glTestFenceAPPLE;
#define glTestFenceAPPLE glad_glTestFenceAPPLE
typedef void (APIENTRYP PFNGLFINISHFENCEAPPLEPROC)(GLuint fence);
GLAPI PFNGLFINISHFENCEAPPLEPROC glad_glFinishFenceAPPLE;
#define glFinishFenceAPPLE glad_glFinishFenceAPPLE
typedef GLboolean (APIENTRYP PFNGLTESTOBJECTAPPLEPROC)(GLenum object, GLuint name);
GLAPI PFNGLTESTOBJECTAPPLEPROC glad_glTestObjectAPPLE;
#define glTestObjectAPPLE glad_glTestObjectAPPLE
typedef void (APIENTRYP PFNGLFINISHOBJECTAPPLEPROC)(GLenum object, GLint name);
GLAPI PFNGLFINISHOBJECTAPPLEPROC glad_glFinishObjectAPPLE;
#define glFinishObjectAPPLE glad_glFinishObjectAPPLE
#endif
#ifndef GL_APPLE_float_pixels
#define GL_APPLE_float_pixels 1
GLAPI int GLAD_GL_APPLE_float_pixels;
#endif
#ifndef GL_APPLE_flush_buffer_range
#define GL_APPLE_flush_buffer_range 1
GLAPI int GLAD_GL_APPLE_flush_buffer_range;
typedef void (APIENTRYP PFNGLBUFFERPARAMETERIAPPLEPROC)(GLenum target, GLenum pname, GLint param);
GLAPI PFNGLBUFFERPARAMETERIAPPLEPROC glad_glBufferParameteriAPPLE;
#define glBufferParameteriAPPLE glad_glBufferParameteriAPPLE
typedef void (APIENTRYP PFNGLFLUSHMAPPEDBUFFERRANGEAPPLEPROC)(GLenum target, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLFLUSHMAPPEDBUFFERRANGEAPPLEPROC glad_glFlushMappedBufferRangeAPPLE;
#define glFlushMappedBufferRangeAPPLE glad_glFlushMappedBufferRangeAPPLE
#endif
#ifndef GL_APPLE_object_purgeable
#define GL_APPLE_object_purgeable 1
GLAPI int GLAD_GL_APPLE_object_purgeable;
typedef GLenum (APIENTRYP PFNGLOBJECTPURGEABLEAPPLEPROC)(GLenum objectType, GLuint name, GLenum option);
GLAPI PFNGLOBJECTPURGEABLEAPPLEPROC glad_glObjectPurgeableAPPLE;
#define glObjectPurgeableAPPLE glad_glObjectPurgeableAPPLE
typedef GLenum (APIENTRYP PFNGLOBJECTUNPURGEABLEAPPLEPROC)(GLenum objectType, GLuint name, GLenum option);
GLAPI PFNGLOBJECTUNPURGEABLEAPPLEPROC glad_glObjectUnpurgeableAPPLE;
#define glObjectUnpurgeableAPPLE glad_glObjectUnpurgeableAPPLE
typedef void (APIENTRYP PFNGLGETOBJECTPARAMETERIVAPPLEPROC)(GLenum objectType, GLuint name, GLenum pname, GLint *params);
GLAPI PFNGLGETOBJECTPARAMETERIVAPPLEPROC glad_glGetObjectParameterivAPPLE;
#define glGetObjectParameterivAPPLE glad_glGetObjectParameterivAPPLE
#endif
#ifndef GL_APPLE_rgb_422
#define GL_APPLE_rgb_422 1
GLAPI int GLAD_GL_APPLE_rgb_422;
#endif
#ifndef GL_APPLE_row_bytes
#define GL_APPLE_row_bytes 1
GLAPI int GLAD_GL_APPLE_row_bytes;
#endif
#ifndef GL_APPLE_specular_vector
#define GL_APPLE_specular_vector 1
GLAPI int GLAD_GL_APPLE_specular_vector;
#endif
#ifndef GL_APPLE_texture_range
#define GL_APPLE_texture_range 1
GLAPI int GLAD_GL_APPLE_texture_range;
typedef void (APIENTRYP PFNGLTEXTURERANGEAPPLEPROC)(GLenum target, GLsizei length, const void *pointer);
GLAPI PFNGLTEXTURERANGEAPPLEPROC glad_glTextureRangeAPPLE;
#define glTextureRangeAPPLE glad_glTextureRangeAPPLE
typedef void (APIENTRYP PFNGLGETTEXPARAMETERPOINTERVAPPLEPROC)(GLenum target, GLenum pname, void **params);
GLAPI PFNGLGETTEXPARAMETERPOINTERVAPPLEPROC glad_glGetTexParameterPointervAPPLE;
#define glGetTexParameterPointervAPPLE glad_glGetTexParameterPointervAPPLE
#endif
#ifndef GL_APPLE_transform_hint
#define GL_APPLE_transform_hint 1
GLAPI int GLAD_GL_APPLE_transform_hint;
#endif
#ifndef GL_APPLE_vertex_array_object
#define GL_APPLE_vertex_array_object 1
GLAPI int GLAD_GL_APPLE_vertex_array_object;
typedef void (APIENTRYP PFNGLBINDVERTEXARRAYAPPLEPROC)(GLuint array);
GLAPI PFNGLBINDVERTEXARRAYAPPLEPROC glad_glBindVertexArrayAPPLE;
```

```
#define glBindVertexArrayAPPLE glad_glBindVertexArrayAPPLE
typedef void (APIENTRYP PFNGLDELETEVERTEXARRAYSAPPLEPROC)(GLsizei n, const GLuint *arrays);
GLAPI PFNGLDELETEVERTEXARRAYSAPPLEPROC glad_glDeleteVertexArraysAPPLE;
#define glDeleteVertexArraysAPPLE glad_glDeleteVertexArraysAPPLE
typedef void (APIENTRYP PFNGLGENVERTEXARRAYSAPPLEPROC)(GLsizei n, GLuint *arrays);
GLAPI PFNGLGENVERTEXARRAYSAPPLEPROC glad_glGenVertexArraysAPPLE;
#define glGenVertexArraysAPPLE glad_glGenVertexArraysAPPLE
typedef GLboolean (APIENTRYP PFNGLISVERTEXARRAYAPPLEPROC)(GLuint array);
GLAPI PFNGLISVERTEXARRAYAPPLEPROC glad_glIsVertexArrayAPPLE;
#define glIsVertexArrayAPPLE glad_glIsVertexArrayAPPLE
#endif
#ifndef GL_APPLE_vertex_array_range
#define GL_APPLE_vertex_array_range 1
GLAPI int GLAD_GL_APPLE_vertex_array_range;
typedef void (APIENTRYP PFNGLVERTEXARRAYRANGEAPPLEPROC)(GLsizei length, void *pointer);
GLAPI PFNGLVERTEXARRAYRANGEAPPLEPROC glad_glVertexArrayRangeAPPLE;
#define glVertexArrayRangeAPPLE glad_glVertexArrayRangeAPPLE
typedef void (APIENTRYP PFNGLFLUSHVERTEXARRAYRANGEAPPLEPROC)(GLsizei length, void *pointer);
GLAPI PFNGLFLUSHVERTEXARRAYRANGEAPPLEPROC glad_glFlushVertexArrayRangeAPPLE;
#define glFlushVertexArrayRangeAPPLE glad_glFlushVertexArrayRangeAPPLE
typedef void (APIENTRYP PFNGLVERTEXARRAYPARAMETERIAPPLEPROC)(GLenum pname, GLint param);
GLAPI PFNGLVERTEXARRAYPARAMETERIAPPLEPROC glad_glVertexArrayParameteriAPPLE;
#define glVertexArrayParameteriAPPLE glad_glVertexArrayParameteriAPPLE
#endif
#ifndef GL_APPLE_vertex_program_evaluators
#define GL_APPLE_vertex_program_evaluators 1
GLAPI int GLAD_GL_APPLE_vertex_program_evaluators;
typedef void (APIENTRYP PFNGLENABLEVERTEXATTRIBAPPLEPROC)(GLuint index, GLenum pname);
GLAPI PFNGLENABLEVERTEXATTRIBAPPLEPROC glad_glEnableVertexAttribAPPLE;
#define glEnableVertexAttribAPPLE glad_glEnableVertexAttribAPPLE
typedef void (APIENTRYP PFNGLDISABLEVERTEXATTRIBAPPLEPROC)(GLuint index, GLenum pname);
GLAPI PFNGLDISABLEVERTEXATTRIBAPPLEPROC glad_glDisableVertexAttribAPPLE;
#define glDisableVertexAttribAPPLE glad_glDisableVertexAttribAPPLE
typedef GLboolean (APIENTRYP PFNGLISVERTEXATTRIBENABLEDAPPLEPROC)(GLuint index, GLenum pname);
GLAPI PFNGLISVERTEXATTRIBENABLEDAPPLEPROC glad_glIsVertexAttribEnabledAPPLE;
#define glIsVertexAttribEnabledAPPLE glad_glIsVertexAttribEnabledAPPLE
typedef void (APIENTRYP PFNGLMAPVERTEXATTRIB1DAPPLEPROC)(GLuint index, GLuint size, GLdouble u1, GLdouble u2, GLint stride, GLint order, const GLdouble *points);
GLAPI PFNGLMAPVERTEXATTRIB1DAPPLEPROC glad_glMapVertexAttrib1dAPPLE;
#define glMapVertexAttrib1dAPPLE glad_glMapVertexAttrib1dAPPLE
typedef void (APIENTRYP PFNGLMAPVERTEXATTRIB1FAPPLEPROC)(GLuint index, GLuint size, GLfloat u1, GLfloat u2, GLint stride, GLint order, const GLfloat *points);
GLAPI PFNGLMAPVERTEXATTRIB1FAPPLEPROC glad_glMapVertexAttrib1fAPPLE;
#define glMapVertexAttrib1fAPPLE glad_glMapVertexAttrib1fAPPLE
typedef void (APIENTRYP PFNGLMAPVERTEXATTRIB2DAPPLEPROC)(GLuint index, GLuint size, GLdouble u1, GLdouble u2, GLint ustride, GLint uorder, GLdouble v1, GLdouble v2, GLint vstride, GLint vorder, const GLdouble *points);
GLAPI PFNGLMAPVERTEXATTRIB2DAPPLEPROC glad_glMapVertexAttrib2dAPPLE;
#define glMapVertexAttrib2dAPPLE glad_glMapVertexAttrib2dAPPLE
typedef void (APIENTRYP PFNGLMAPVERTEXATTRIB2FAPPLEPROC)(GLuint index, GLuint size, GLfloat u1, GLfloat u2, GLint ustride, GLint uorder, GLfloat v1, GLfloat v2, GLint vstride, GLint vorder, const GLfloat *points);
GLAPI PFNGLMAPVERTEXATTRIB2FAPPLEPROC glad_glMapVertexAttrib2fAPPLE;
#define glMapVertexAttrib2fAPPLE glad_glMapVertexAttrib2fAPPLE
#endif
#ifndef GL_APPLE_ycbcr_422
#define GL_APPLE_ycbcr_422 1
GLAPI int GLAD_GL_APPLE_ycbcr_422;
#endif
#ifndef GL_ARB_ES2_compatibility
#define GL_ARB_ES2_compatibility 1
GLAPI int GLAD_GL_ARB_ES2_compatibility;
#endif
#ifndef GL_ARB_ES3_1_compatibility
#define GL_ARB_ES3_1_compatibility 1
GLAPI int GLAD_GL_ARB_ES3_1_compatibility;
typedef void (APIENTRYP PFNGLMEMORYBARRIERBYREGIONPROC)(GLbitfield barriers);
GLAPI PFNGLMEMORYBARRIERBYREGIONPROC glad_glMemoryBarrierByRegion;
#define glMemoryBarrierByRegion glad_glMemoryBarrierByRegion
#endif
#ifndef GL_ARB_ES3_2_compatibility
#define GL_ARB_ES3_2_compatibility 1
GLAPI int GLAD_GL_ARB_ES3_2_compatibility;
typedef void (APIENTRYP PFNGLPRIMITIVEBOUNDINGBOXARBPROC)(GLfloat minX, GLfloat minY, GLfloat minZ, GLfloat minW, GLfloat maxX, GLfloat maxY, GLfloat maxZ, GLfloat maxW);
GLAPI PFNGLPRIMITIVEBOUNDINGBOXARBPROC glad_glPrimitiveBoundingBoxARB;
#define glPrimitiveBoundingBoxARB glad_glPrimitiveBoundingBoxARB
#endif
#ifndef GL_ARB_ES3_compatibility
#define GL_ARB_ES3_compatibility 1
GLAPI int GLAD_GL_ARB_ES3_compatibility;
#endif
#ifndef GL_ARB_arrays_of_arrays
#define GL_ARB_arrays_of_arrays 1
GLAPI int GLAD_GL_ARB_arrays_of_arrays;
#endif
#ifndef GL_ARB_base_instance
```

```
#define GL_ARB_base_instance 1
GLAPI int GLAD_GL_ARB_base_instance;
typedef void (APIENTRYP PFNGLDRAWARRAYSINSTANCEDBASEINSTANCEPROC)(GLenum mode, GLint first, GLsizei count, GLsizei instancecount, GLuint baseinstance);
GLAPI PFNGLDRAWARRAYSINSTANCEDBASEINSTANCEPROC glad_glDrawArraysInstancedBaseInstance;
#define glDrawArraysInstancedBaseInstance glad_glDrawArraysInstancedBaseInstance
typedef void (APIENTRYP PFNGLDRAWELEMENTSINSTANCEDBASEINSTANCEPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices, GLsizei instancecount, GLuint baseinstance);
GLAPI PFNGLDRAWELEMENTSINSTANCEDBASEINSTANCEPROC glad_glDrawElementsInstancedBaseInstance;
#define glDrawElementsInstancedBaseInstance glad_glDrawElementsInstancedBaseInstance
typedef void (APIENTRYP PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXBASEINSTANCEPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices, GLsizei instancecount, GLint basevertex, GLuint baseinstance);
GLAPI PFNGLDRAWELEMENTSINSTANCEDBASEVERTEXBASEINSTANCEPROC glad_glDrawElementsInstancedBaseVertexBaseInstance;
#define glDrawElementsInstancedBaseVertexBaseInstance glad_glDrawElementsInstancedBaseVertexBaseInstance
#endif
#ifndef GL_ARB_bindless_texture
#define GL_ARB_bindless_texture 1
GLAPI int GLAD_GL_ARB_bindless_texture;
typedef GLuint64 (APIENTRYP PFNGLGETTEXTUREHANDLEARBPROC)(GLuint texture);
GLAPI PFNGLGETTEXTUREHANDLEARBPROC glad_glGetTextureHandleARB;
#define glGetTextureHandleARB glad_glGetTextureHandleARB
typedef GLuint64 (APIENTRYP PFNGLGETTEXTURESAMPLERHANDLEARBPROC)(GLuint texture, GLuint sampler);
GLAPI PFNGLGETTEXTURESAMPLERHANDLEARBPROC glad_glGetTextureSamplerHandleARB;
#define glGetTextureSamplerHandleARB glad_glGetTextureSamplerHandleARB
typedef void (APIENTRYP PFNGLMAKETEXTUREHANDLERESIDENTARBPROC)(GLuint64 handle);
GLAPI PFNGLMAKETEXTUREHANDLERESIDENTARBPROC glad_glMakeTextureHandleResidentARB;
#define glMakeTextureHandleResidentARB glad_glMakeTextureHandleResidentARB
typedef void (APIENTRYP PFNGLMAKETEXTUREHANDLENONRESIDENTARBPROC)(GLuint64 handle);
GLAPI PFNGLMAKETEXTUREHANDLENONRESIDENTARBPROC glad_glMakeTextureHandleNonResidentARB;
#define glMakeTextureHandleNonResidentARB glad_glMakeTextureHandleNonResidentARB
typedef GLuint64 (APIENTRYP PFNGLGETIMAGEHANDLEARBPROC)(GLuint texture, GLint level, GLboolean layered, GLint layer, GLenum format);
GLAPI PFNGLGETIMAGEHANDLEARBPROC glad_glGetImageHandleARB;
#define glGetImageHandleARB glad_glGetImageHandleARB
typedef void (APIENTRYP PFNGLMAKEIMAGEHANDLERESIDENTARBPROC)(GLuint64 handle, GLenum access);
GLAPI PFNGLMAKEIMAGEHANDLERESIDENTARBPROC glad_glMakeImageHandleResidentARB;
#define glMakeImageHandleResidentARB glad_glMakeImageHandleResidentARB
typedef void (APIENTRYP PFNGLMAKEIMAGEHANDLENONRESIDENTARBPROC)(GLuint64 handle);
GLAPI PFNGLMAKEIMAGEHANDLENONRESIDENTARBPROC glad_glMakeImageHandleNonResidentARB;
#define glMakeImageHandleNonResidentARB glad_glMakeImageHandleNonResidentARB
typedef void (APIENTRYP PFNGLUNIFORMHANDLEUI64ARBPROC)(GLint location, GLuint64 value);
GLAPI PFNGLUNIFORMHANDLEUI64ARBPROC glad_glUniformHandleui64ARB;
#define glUniformHandleui64ARB glad_glUniformHandleui64ARB
typedef void (APIENTRYP PFNGLUNIFORMHANDLEUI64VARBPROC)(GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLUNIFORMHANDLEUI64VARBPROC glad_glUniformHandleui64vARB;
#define glUniformHandleui64vARB glad_glUniformHandleui64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMHANDLEUI64ARBPROC)(GLuint program, GLint location, GLuint64 value);
GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64ARBPROC glad_glProgramUniformHandleui64ARB;
#define glProgramUniformHandleui64ARB glad_glProgramUniformHandleui64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMHANDLEUI64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLuint64 *values);
GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64VARBPROC glad_glProgramUniformHandleui64vARB;
#define glProgramUniformHandleui64vARB glad_glProgramUniformHandleui64vARB
typedef GLboolean (APIENTRYP PFNGLISTEXTUREHANDLERESIDENTARBPROC)(GLuint64 handle);
GLAPI PFNGLISTEXTUREHANDLERESIDENTARBPROC glad_glIsTextureHandleResidentARB;
#define glIsTextureHandleResidentARB glad_glIsTextureHandleResidentARB
typedef GLboolean (APIENTRYP PFNGLISIMAGEHANDLERESIDENTARBPROC)(GLuint64 handle);
GLAPI PFNGLISIMAGEHANDLERESIDENTARBPROC glad_glIsImageHandleResidentARB;
#define glIsImageHandleResidentARB glad_glIsImageHandleResidentARB
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1UI64ARBPROC)(GLuint index, GLuint64EXT x);
GLAPI PFNGLVERTEXATTRIBL1UI64ARBPROC glad_glVertexAttribL1ui64ARB;
#define glVertexAttribL1ui64ARB glad_glVertexAttribL1ui64ARB
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1UI64VARBPROC)(GLuint index, const GLuint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL1UI64VARBPROC glad_glVertexAttribL1ui64vARB;
#define glVertexAttribL1ui64vARB glad_glVertexAttribL1ui64vARB
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBLUI64VARBPROC)(GLuint index, GLenum pname, GLuint64EXT *params);
GLAPI PFNGLGETVERTEXATTRIBLUI64VARBPROC glad_glGetVertexAttribLui64vARB;
#define glGetVertexAttribLui64vARB glad_glGetVertexAttribLui64vARB
#endif
#ifndef GL_ARB_blend_func_extended
#define GL_ARB_blend_func_extended 1
GLAPI int GLAD_GL_ARB_blend_func_extended;
#endif
#ifndef GL_ARB_buffer_storage
#define GL_ARB_buffer_storage 1
GLAPI int GLAD_GL_ARB_buffer_storage;
typedef void (APIENTRYP PFNGLBUFFERSTORAGEPROC)(GLenum target, GLsizeiptr size, const void *data, GLbitfield flags);
GLAPI PFNGLBUFFERSTORAGEPROC glad_glBufferStorage;
#define glBufferStorage glad_glBufferStorage
#endif
```

```
#ifndef GL_ARB_cl_event
#define GL_ARB_cl_event 1
GLAPI int GLAD_GL_ARB_cl_event;
typedef GLsync (APIENTRYP PFNGLCREATESYNCFROMCLEVENTARBPROC)(struct _cl_context *context, struct _cl_event *event, GLbitfield flags);
GLAPI PFNGLCREATESYNCFROMCLEVENTARBPROC glad_glCreateSyncFromCLeventARB;
#define glCreateSyncFromCLeventARB glad_glCreateSyncFromCLeventARB
#endif
#ifndef GL_ARB_clear_buffer_object
#define GL_ARB_clear_buffer_object 1
GLAPI int GLAD_GL_ARB_clear_buffer_object;
typedef void (APIENTRYP PFNGLCLEARBUFFERDATAPROC)(GLenum target, GLenum internalformat, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARBUFFERDATAPROC glad_glClearBufferData;
#define glClearBufferData glad_glClearBufferData
typedef void (APIENTRYP PFNGLCLEARBUFFERSUBDATAPROC)(GLenum target, GLenum internalformat, GLintptr offset, GLsizeiptr size, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARBUFFERSUBDATAPROC glad_glClearBufferSubData;
#define glClearBufferSubData glad_glClearBufferSubData
#endif
#ifndef GL_ARB_clear_texture
#define GL_ARB_clear_texture 1
GLAPI int GLAD_GL_ARB_clear_texture;
typedef void (APIENTRYP PFNGLCLEARTEXIMAGEPROC)(GLuint texture, GLint level, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARTEXIMAGEPROC glad_glClearTexImage;
#define glClearTexImage glad_glClearTexImage
typedef void (APIENTRYP PFNGLCLEARTEXSUBIMAGEPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARTEXSUBIMAGEPROC glad_glClearTexSubImage;
#define glClearTexSubImage glad_glClearTexSubImage
#endif
#ifndef GL_ARB_clip_control
#define GL_ARB_clip_control 1
GLAPI int GLAD_GL_ARB_clip_control;
typedef void (APIENTRYP PFNGLCLIPCONTROLPROC)(GLenum origin, GLenum depth);
GLAPI PFNGLCLIPCONTROLPROC glad_glClipControl;
#define glClipControl glad_glClipControl
#endif
#ifndef GL_ARB_color_buffer_float
#define GL_ARB_color_buffer_float 1
GLAPI int GLAD_GL_ARB_color_buffer_float;
typedef void (APIENTRYP PFNGLCLAMPCOLORARBPROC)(GLenum target, GLenum clamp);
GLAPI PFNGLCLAMPCOLORARBPROC glad_glClampColorARB;
#define glClampColorARB glad_glClampColorARB
#endif
#ifndef GL_ARB_compatibility
#define GL_ARB_compatibility 1
GLAPI int GLAD_GL_ARB_compatibility;
#endif
#ifndef GL_ARB_compressed_texture_pixel_storage
#define GL_ARB_compressed_texture_pixel_storage 1
GLAPI int GLAD_GL_ARB_compressed_texture_pixel_storage;
#endif
#ifndef GL_ARB_compute_shader
#define GL_ARB_compute_shader 1
GLAPI int GLAD_GL_ARB_compute_shader;
typedef void (APIENTRYP PFNGLDISPATCHCOMPUTEPROC)(GLuint num_groups_x, GLuint num_groups_y, GLuint num_groups_z);
GLAPI PFNGLDISPATCHCOMPUTEPROC glad_glDispatchCompute;
#define glDispatchCompute glad_glDispatchCompute
typedef void (APIENTRYP PFNGLDISPATCHCOMPUTEINDIRECTPROC)(GLintptr indirect);
GLAPI PFNGLDISPATCHCOMPUTEINDIRECTPROC glad_glDispatchComputeIndirect;
#define glDispatchComputeIndirect glad_glDispatchComputeIndirect
#endif
#ifndef GL_ARB_compute_variable_group_size
#define GL_ARB_compute_variable_group_size 1
GLAPI int GLAD_GL_ARB_compute_variable_group_size;
typedef void (APIENTRYP 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);
GLAPI PFNGLDISPATCHCOMPUTEGROUPSIZEARBPROC glad_glDispatchComputeGroupSizeARB;
#define glDispatchComputeGroupSizeARB glad_glDispatchComputeGroupSizeARB
#endif
#ifndef GL_ARB_conditional_render_inverted
#define GL_ARB_conditional_render_inverted 1
GLAPI int GLAD_GL_ARB_conditional_render_inverted;
#endif
#ifndef GL_ARB_conservative_depth
#define GL_ARB_conservative_depth 1
GLAPI int GLAD_GL_ARB_conservative_depth;
#endif
#ifndef GL_ARB_copy_buffer
#define GL_ARB_copy_buffer 1
GLAPI int GLAD_GL_ARB_copy_buffer;
#endif
#ifndef GL_ARB_copy_image
```

```
#define GL_ARB_copy_image 1
GLAPI int GLAD_GL_ARB_copy_image;
typedef void (APIENTRYP 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);
GLAPI PFNGLCOPYIMAGESUBDATAPROC glad_glCopyImageSubData;
#define glCopyImageSubData glad_glCopyImageSubData
#endif
#ifndef GL_ARB_cull_distance
#define GL_ARB_cull_distance 1
GLAPI int GLAD_GL_ARB_cull_distance;
#endif
#ifndef GL_ARB_debug_output
#define GL_ARB_debug_output 1
GLAPI int GLAD_GL_ARB_debug_output;
typedef void (APIENTRYP PFNGLDEBUGMESSAGECONTROLARBPROC)(GLenum source, GLenum type, GLenum severity, GLsizei count, const GLuint *ids, GLboolean enabled);
GLAPI PFNGLDEBUGMESSAGECONTROLARBPROC glad_glDebugMessageControlARB;
#define glDebugMessageControlARB glad_glDebugMessageControlARB
typedef void (APIENTRYP PFNGLDEBUGMESSAGEINSERTARBPROC)(GLenum source, GLenum type, GLuint id, GLenum severity, GLsizei length, const GLchar *buf);
GLAPI PFNGLDEBUGMESSAGEINSERTARBPROC glad_glDebugMessageInsertARB;
#define glDebugMessageInsertARB glad_glDebugMessageInsertARB
typedef void (APIENTRYP PFNGLDEBUGMESSAGECALLBACKARBPROC)(GLDEBUGPROCARB callback, const void *userParam);
GLAPI PFNGLDEBUGMESSAGECALLBACKARBPROC glad_glDebugMessageCallbackARB;
#define glDebugMessageCallbackARB glad_glDebugMessageCallbackARB
typedef GLuint (APIENTRYP PFNGLGETDEBUGMESSAGELOGARBPROC)(GLuint count, GLsizei bufSize, GLenum *sources, GLenum *types, GLuint *ids, GLenum *severities, GLsizei *lengths, GLchar *messageLog);
GLAPI PFNGLGETDEBUGMESSAGELOGARBPROC glad_glGetDebugMessageLogARB;
#define glGetDebugMessageLogARB glad_glGetDebugMessageLogARB
#endif
#ifndef GL_ARB_depth_buffer_float
#define GL_ARB_depth_buffer_float 1
GLAPI int GLAD_GL_ARB_depth_buffer_float;
#endif
#ifndef GL_ARB_depth_clamp
#define GL_ARB_depth_clamp 1
GLAPI int GLAD_GL_ARB_depth_clamp;
#endif
#ifndef GL_ARB_depth_texture
#define GL_ARB_depth_texture 1
GLAPI int GLAD_GL_ARB_depth_texture;
#endif
#ifndef GL_ARB_derivative_control
#define GL_ARB_derivative_control 1
GLAPI int GLAD_GL_ARB_derivative_control;
#endif
#ifndef GL_ARB_direct_state_access
#define GL_ARB_direct_state_access 1
GLAPI int GLAD_GL_ARB_direct_state_access;
typedef void (APIENTRYP PFNGLCREATETRANSFORMFEEDBACKSPROC)(GLsizei n, GLuint *ids);
GLAPI PFNGLCREATETRANSFORMFEEDBACKSPROC glad_glCreateTransformFeedbacks;
#define glCreateTransformFeedbacks glad_glCreateTransformFeedbacks
typedef void (APIENTRYP PFNGLTRANSFORMFEEDBACKBUFFERBASEPROC)(GLuint xfb, GLuint index, GLuint buffer);
GLAPI PFNGLTRANSFORMFEEDBACKBUFFERBASEPROC glad_glTransformFeedbackBufferBase;
#define glTransformFeedbackBufferBase glad_glTransformFeedbackBufferBase
typedef void (APIENTRYP PFNGLTRANSFORMFEEDBACKBUFFERRANGEPROC)(GLuint xfb, GLuint index, GLuint buffer, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLTRANSFORMFEEDBACKBUFFERRANGEPROC glad_glTransformFeedbackBufferRange;
#define glTransformFeedbackBufferRange glad_glTransformFeedbackBufferRange
typedef void (APIENTRYP PFNGLGETTRANSFORMFEEDBACKIVPROC)(GLuint xfb, GLenum pname, GLint *param);
GLAPI PFNGLGETTRANSFORMFEEDBACKIVPROC glad_glGetTransformFeedbackiv;
#define glGetTransformFeedbackiv glad_glGetTransformFeedbackiv
typedef void (APIENTRYP PFNGLGETTRANSFORMFEEDBACKI_VPROC)(GLuint xfb, GLenum pname, GLuint index, GLint *param);
GLAPI PFNGLGETTRANSFORMFEEDBACKI_VPROC glad_glGetTransformFeedbacki_v;
#define glGetTransformFeedbacki_v glad_glGetTransformFeedbacki_v
typedef void (APIENTRYP PFNGLGETTRANSFORMFEEDBACKI64_VPROC)(GLuint xfb, GLenum pname, GLuint index, GLint64 *param);
GLAPI PFNGLGETTRANSFORMFEEDBACKI64_VPROC glad_glGetTransformFeedbacki64_v;
#define glGetTransformFeedbacki64_v glad_glGetTransformFeedbacki64_v
typedef void (APIENTRYP PFNGLCREATEBUFFERSPROC)(GLsizei n, GLuint *buffers);
GLAPI PFNGLCREATEBUFFERSPROC glad_glCreateBuffers;
#define glCreateBuffers glad_glCreateBuffers
typedef void (APIENTRYP PFNGLNAMEDBUFFERSTORAGEPROC)(GLuint buffer, GLsizeiptr size, const void *data, GLbitfield flags);
GLAPI PFNGLNAMEDBUFFERSTORAGEPROC glad_glNamedBufferStorage;
#define glNamedBufferStorage glad_glNamedBufferStorage
typedef void (APIENTRYP PFNGLNAMEDBUFFERDATAPROC)(GLuint buffer, GLsizeiptr size, const void *data, GLenum usage);
GLAPI PFNGLNAMEDBUFFERDATAPROC glad_glNamedBufferData;
#define glNamedBufferData glad_glNamedBufferData
typedef void (APIENTRYP PFNGLNAMEDBUFFERSUBDATAPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, const void *data);
GLAPI PFNGLNAMEDBUFFERSUBDATAPROC glad_glNamedBufferSubData;
```

```
#define glNamedBufferSubData glad_glNamedBufferSubData
typedef void (APIENTRYP PFNGLCOPYNAMEDBUFFERSUBDATAPROC)(GLuint readBuffer, GLuint writeBuffer, GLintptr readOffset, GLintptr writeOffset, GLsizeiptr size);
GLAPI PFNGLCOPYNAMEDBUFFERSUBDATAPROC glad_glCopyNamedBufferSubData;
#define glCopyNamedBufferSubData glad_glCopyNamedBufferSubData
typedef void (APIENTRYP PFNGLCLEARNAMEDBUFFERDATAPROC)(GLuint buffer, GLenum internalformat, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARNAMEDBUFFERDATAPROC glad_glClearNamedBufferData;
#define glClearNamedBufferData glad_glClearNamedBufferData
typedef void (APIENTRYP PFNGLCLEARNAMEDBUFFERSUBDATAPROC)(GLuint buffer, GLenum internalformat, GLintptr offset, GLsizeiptr size, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARNAMEDBUFFERSUBDATAPROC glad_glClearNamedBufferSubData;
#define glClearNamedBufferSubData glad_glClearNamedBufferSubData
typedef void * (APIENTRYP PFNGLMAPNAMEDBUFFERPROC)(GLuint buffer, GLenum access);
GLAPI PFNGLMAPNAMEDBUFFERPROC glad_glMapNamedBuffer;
#define glMapNamedBuffer glad_glMapNamedBuffer
typedef void * (APIENTRYP PFNGLMAPNAMEDBUFFERRANGEPROC)(GLuint buffer, GLintptr offset, GLsizeiptr length, GLbitfield access);
GLAPI PFNGLMAPNAMEDBUFFERRANGEPROC glad_glMapNamedBufferRange;
#define glMapNamedBufferRange glad_glMapNamedBufferRange
typedef GLboolean (APIENTRYP PFNGLUNMAPNAMEDBUFFERPROC)(GLuint buffer);
GLAPI PFNGLUNMAPNAMEDBUFFERPROC glad_glUnmapNamedBuffer;
#define glUnmapNamedBuffer glad_glUnmapNamedBuffer
typedef void (APIENTRYP PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEPROC)(GLuint buffer, GLintptr offset, GLsizeiptr length);
GLAPI PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEPROC glad_glFlushMappedNamedBufferRange;
#define glFlushMappedNamedBufferRange glad_glFlushMappedNamedBufferRange
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERPARAMETERIVPROC)(GLuint buffer, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDBUFFERPARAMETERIVPROC glad_glGetNamedBufferParameteriv;
#define glGetNamedBufferParameteriv glad_glGetNamedBufferParameteriv
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERPARAMETERI64VPROC)(GLuint buffer, GLenum pname, GLint64 *params);
GLAPI PFNGLGETNAMEDBUFFERPARAMETERI64VPROC glad_glGetNamedBufferParameteri64v;
#define glGetNamedBufferParameteri64v glad_glGetNamedBufferParameteri64v
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERPOINTERVPROC)(GLuint buffer, GLenum pname, void **params);
GLAPI PFNGLGETNAMEDBUFFERPOINTERVPROC glad_glGetNamedBufferPointerv;
#define glGetNamedBufferPointerv glad_glGetNamedBufferPointerv
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERSUBDATAPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, void *data);
GLAPI PFNGLGETNAMEDBUFFERSUBDATAPROC glad_glGetNamedBufferSubData;
#define glGetNamedBufferSubData glad_glGetNamedBufferSubData
typedef void (APIENTRYP PFNGLCREATEFRAMEBUFFERSPROC)(GLsizei n, GLuint *framebuffers);
GLAPI PFNGLCREATEFRAMEBUFFERSPROC glad_glCreateFramebuffers;
#define glCreateFramebuffers glad_glCreateFramebuffers
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERRENDERBUFFERPROC)(GLuint framebuffer, GLenum attachment, GLenum renderbuffertarget, GLuint renderbuffer);
GLAPI PFNGLNAMEDFRAMEBUFFERRENDERBUFFERPROC glad_glNamedFramebufferRenderbuffer;
#define glNamedFramebufferRenderbuffer glad_glNamedFramebufferRenderbuffer
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERPARAMETERIPROC)(GLuint framebuffer, GLenum pname, GLint param);
GLAPI PFNGLNAMEDFRAMEBUFFERPARAMETERIPROC glad_glNamedFramebufferParameteri;
#define glNamedFramebufferParameteri glad_glNamedFramebufferParameteri
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTUREPROC)(GLuint framebuffer, GLenum attachment, GLuint texture, GLint level);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREPROC glad_glNamedFramebufferTexture;
#define glNamedFramebufferTexture glad_glNamedFramebufferTexture
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTURELAYERPROC)(GLuint framebuffer, GLenum attachment, GLuint texture, GLint level, GLint layer);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURELAYERPROC glad_glNamedFramebufferTextureLayer;
#define glNamedFramebufferTextureLayer glad_glNamedFramebufferTextureLayer
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERDRAWBUFFERPROC)(GLuint framebuffer, GLenum buf);
GLAPI PFNGLNAMEDFRAMEBUFFERDRAWBUFFERPROC glad_glNamedFramebufferDrawBuffer;
#define glNamedFramebufferDrawBuffer glad_glNamedFramebufferDrawBuffer
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERDRAWBUFFERSPROC)(GLuint framebuffer, GLsizei n, const GLenum *bufs);
GLAPI PFNGLNAMEDFRAMEBUFFERDRAWBUFFERSPROC glad_glNamedFramebufferDrawBuffers;
#define glNamedFramebufferDrawBuffers glad_glNamedFramebufferDrawBuffers
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERREADBUFFERPROC)(GLuint framebuffer, GLenum src);
GLAPI PFNGLNAMEDFRAMEBUFFERREADBUFFERPROC glad_glNamedFramebufferReadBuffer;
#define glNamedFramebufferReadBuffer glad_glNamedFramebufferReadBuffer
typedef void (APIENTRYP PFNGLINVALIDATENAMEDFRAMEBUFFERDATAPROC)(GLuint framebuffer, GLsizei numAttachments, const GLenum *attachments);
GLAPI PFNGLINVALIDATENAMEDFRAMEBUFFERDATAPROC glad_glInvalidateNamedFramebufferData;
#define glInvalidateNamedFramebufferData glad_glInvalidateNamedFramebufferData
typedef void (APIENTRYP PFNGLINVALIDATENAMEDFRAMEBUFFERSUBDATAPROC)(GLuint framebuffer, GLsizei numAttachments, const GLenum *attachments, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLINVALIDATENAMEDFRAMEBUFFERSUBDATAPROC glad_glInvalidateNamedFramebufferSubData;
#define glInvalidateNamedFramebufferSubData glad_glInvalidateNamedFramebufferSubData
typedef void (APIENTRYP PFNGLCLEARNAMEDFRAMEBUFFERIVPROC)(GLuint framebuffer, GLenum buffer, GLint drawbuffer, const GLint *value);
GLAPI PFNGLCLEARNAMEDFRAMEBUFFERIVPROC glad_glClearNamedFramebufferiv;
#define glClearNamedFramebufferiv glad_glClearNamedFramebufferiv
typedef void (APIENTRYP PFNGLCLEARNAMEDFRAMEBUFFERUIVPROC)(GLuint framebuffer, GLenum buffer, GLint drawbuffer, const GLuint *value);
GLAPI PFNGLCLEARNAMEDFRAMEBUFFERUIVPROC glad_glClearNamedFramebufferuiv;
#define glClearNamedFramebufferuiv glad_glClearNamedFramebufferuiv
```

```
typedef void (APIENTRYP PFNGLCLEARNAMEDFRAMEBUFFERFVPROC)(GLuint framebuffer, GLenum buffer, GLint drawbuffer, const GLfloat *value);
GLAPI PFNGLCLEARNAMEDFRAMEBUFFERFVPROC glad_glClearNamedFramebufferfv;
#define glClearNamedFramebufferfv glad_glClearNamedFramebufferfv
typedef void (APIENTRYP PFNGLCLEARNAMEDFRAMEBUFFERFIPROC)(GLuint framebuffer, GLenum buffer, GLint drawbuffer, GLfloat depth, GLint stencil);
GLAPI PFNGLCLEARNAMEDFRAMEBUFFERFIPROC glad_glClearNamedFramebufferfi;
#define glClearNamedFramebufferfi glad_glClearNamedFramebufferfi
typedef void (APIENTRYP PFNGLBLITNAMEDFRAMEBUFFERPROC)(GLuint readFramebuffer, GLuint drawFramebuffer, GLint srcX0, GLint srcY0, GLint srcX1, GLint srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLbitfield mask, GLenum filter);
GLAPI PFNGLBLITNAMEDFRAMEBUFFERPROC glad_glBlitNamedFramebuffer;
#define glBlitNamedFramebuffer glad_glBlitNamedFramebuffer
typedef GLenum (APIENTRYP PFNGLCHECKNAMEDFRAMEBUFFERSTATUSPROC)(GLuint framebuffer, GLenum target);
GLAPI PFNGLCHECKNAMEDFRAMEBUFFERSTATUSPROC glad_glCheckNamedFramebufferStatus;
#define glCheckNamedFramebufferStatus glad_glCheckNamedFramebufferStatus
typedef void (APIENTRYP PFNGLGETNAMEDFRAMEBUFFERPARAMETERIVPROC)(GLuint framebuffer, GLenum pname, GLint *param);
GLAPI PFNGLGETNAMEDFRAMEBUFFERPARAMETERIVPROC glad_glGetNamedFramebufferParameteriv;
#define glGetNamedFramebufferParameteriv glad_glGetNamedFramebufferParameteriv
typedef void (APIENTRYP PFNGLGETNAMEDFRAMEBUFFERATTACHMENTPARAMETERIVPROC)(GLuint framebuffer, GLenum attachment, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDFRAMEBUFFERATTACHMENTPARAMETERIVPROC glad_glGetNamedFramebufferAttachmentParameteriv;
#define glGetNamedFramebufferAttachmentParameteriv glad_glGetNamedFramebufferAttachmentParameteriv
typedef void (APIENTRYP PFNGLCREATERENDERBUFFERSPROC)(GLsizei n, GLuint *renderbuffers);
GLAPI PFNGLCREATERENDERBUFFERSPROC glad_glCreateRenderbuffers;
#define glCreateRenderbuffers glad_glCreateRenderbuffers
typedef void (APIENTRYP PFNGLNAMEDRENDERBUFFERSTORAGEPROC)(GLuint renderbuffer, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEPROC glad_glNamedRenderbufferStorage;
#define glNamedRenderbufferStorage glad_glNamedRenderbufferStorage
typedef void (APIENTRYP PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEPROC)(GLuint renderbuffer, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEPROC glad_glNamedRenderbufferStorageMultisample;
#define glNamedRenderbufferStorageMultisample glad_glNamedRenderbufferStorageMultisample
typedef void (APIENTRYP PFNGLGETNAMEDRENDERBUFFERPARAMETERIVPROC)(GLuint renderbuffer, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDRENDERBUFFERPARAMETERIVPROC glad_glGetNamedRenderbufferParameteriv;
#define glGetNamedRenderbufferParameteriv glad_glGetNamedRenderbufferParameteriv
typedef void (APIENTRYP PFNGLCREATETEXTURESPROC)(GLenum target, GLsizei n, GLuint *textures);
GLAPI PFNGLCREATETEXTURESPROC glad_glCreateTextures;
#define glCreateTextures glad_glCreateTextures
typedef void (APIENTRYP PFNGLTEXTUREBUFFERPROC)(GLuint texture, GLenum internalformat, GLuint buffer);
GLAPI PFNGLTEXTUREBUFFERPROC glad_glTextureBuffer;
#define glTextureBuffer glad_glTextureBuffer
typedef void (APIENTRYP PFNGLTEXTUREBUFFERRANGEPROC)(GLuint texture, GLenum internalformat, GLuint buffer, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLTEXTUREBUFFERRANGEPROC glad_glTextureBufferRange;
#define glTextureBufferRange glad_glTextureBufferRange
typedef void (APIENTRYP PFNGLTEXTURESTORAGE1DPROC)(GLuint texture, GLsizei levels, GLenum internalformat, GLsizei width);
GLAPI PFNGLTEXTURESTORAGE1DPROC glad_glTextureStorage1D;
#define glTextureStorage1D glad_glTextureStorage1D
typedef void (APIENTRYP PFNGLTEXTURESTORAGE2DPROC)(GLuint texture, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLTEXTURESTORAGE2DPROC glad_glTextureStorage2D;
#define glTextureStorage2D glad_glTextureStorage2D
typedef void (APIENTRYP PFNGLTEXTURESTORAGE3DPROC)(GLuint texture, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth);
GLAPI PFNGLTEXTURESTORAGE3DPROC glad_glTextureStorage3D;
#define glTextureStorage3D glad_glTextureStorage3D
typedef void (APIENTRYP PFNGLTEXTURESTORAGE2DMULTISAMPLEPROC)(GLuint texture, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXTURESTORAGE2DMULTISAMPLEPROC glad_glTextureStorage2DMultisample;
#define glTextureStorage2DMultisample glad_glTextureStorage2DMultisample
typedef void (APIENTRYP PFNGLTEXTURESTORAGE3DMULTISAMPLEPROC)(GLuint texture, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXTURESTORAGE3DMULTISAMPLEPROC glad_glTextureStorage3DMultisample;
#define glTextureStorage3DMultisample glad_glTextureStorage3DMultisample
typedef void (APIENTRYP PFNGLTEXTURESUBIMAGE1DPROC)(GLuint texture, GLint level, GLint xoffset, GLsizei width, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTURESUBIMAGE1DPROC glad_glTextureSubImage1D;
#define glTextureSubImage1D glad_glTextureSubImage1D
typedef void (APIENTRYP PFNGLTEXTURESUBIMAGE2DPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTURESUBIMAGE2DPROC glad_glTextureSubImage2D;
#define glTextureSubImage2D glad_glTextureSubImage2D
typedef void (APIENTRYP PFNGLTEXTURESUBIMAGE3DPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTURESUBIMAGE3DPROC glad_glTextureSubImage3D;
#define glTextureSubImage3D glad_glTextureSubImage3D
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTURESUBIMAGE1DPROC)(GLuint texture, GLint level, GLint xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE1DPROC glad_glCompressedTextureSubImage1D;
#define glCompressedTextureSubImage1D glad_glCompressedTextureSubImage1D
```

```
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTURESUBIMAGE2DPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE2DPROC glad_glCompressedTextureSubImage2D;
#define glCompressedTextureSubImage2D glad_glCompressedTextureSubImage2D
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTURESUBIMAGE3DPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE3DPROC glad_glCompressedTextureSubImage3D;
#define glCompressedTextureSubImage3D glad_glCompressedTextureSubImage3D
typedef void (APIENTRYP PFNGLCOPYTEXTURESUBIMAGE1DPROC)(GLuint texture, GLint level, GLint xoffset, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYTEXTURESUBIMAGE1DPROC glad_glCopyTextureSubImage1D;
#define glCopyTextureSubImage1D glad_glCopyTextureSubImage1D
typedef void (APIENTRYP PFNGLCOPYTEXTURESUBIMAGE2DPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXTURESUBIMAGE2DPROC glad_glCopyTextureSubImage2D;
#define glCopyTextureSubImage2D glad_glCopyTextureSubImage2D
typedef void (APIENTRYP PFNGLCOPYTEXTURESUBIMAGE3DPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXTURESUBIMAGE3DPROC glad_glCopyTextureSubImage3D;
#define glCopyTextureSubImage3D glad_glCopyTextureSubImage3D
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERFPROC)(GLuint texture, GLenum pname, GLfloat param);
GLAPI PFNGLTEXTUREPARAMETERFPROC glad_glTextureParameterf;
#define glTextureParameterf glad_glTextureParameterf
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERFVPROC)(GLuint texture, GLenum pname, const GLfloat *param);
GLAPI PFNGLTEXTUREPARAMETERFVPROC glad_glTextureParameterfv;
#define glTextureParameterfv glad_glTextureParameterfv
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIPROC)(GLuint texture, GLenum pname, GLint param);
GLAPI PFNGLTEXTUREPARAMETERIPROC glad_glTextureParameteri;
#define glTextureParameteri glad_glTextureParameteri
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIIVPROC)(GLuint texture, GLenum pname, const GLint *params);
GLAPI PFNGLTEXTUREPARAMETERIIVPROC glad_glTextureParameterIiv;
#define glTextureParameterIiv glad_glTextureParameterIiv
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIUIVPROC)(GLuint texture, GLenum pname, const GLuint *params);
GLAPI PFNGLTEXTUREPARAMETERIUIVPROC glad_glTextureParameterIuiv;
#define glTextureParameterIuiv glad_glTextureParameterIuiv
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIVPROC)(GLuint texture, GLenum pname, const GLint *param);
GLAPI PFNGLTEXTUREPARAMETERIVPROC glad_glTextureParameteriv;
#define glTextureParameteriv glad_glTextureParameteriv
typedef void (APIENTRYP PFNGLGENERATETEXTUREMIPMAPPROC)(GLuint texture);
GLAPI PFNGLGENERATETEXTUREMIPMAPPROC glad_glGenerateTextureMipmap;
#define glGenerateTextureMipmap glad_glGenerateTextureMipmap
typedef void (APIENTRYP PFNGLBINDTEXTUREUNITPROC)(GLuint unit, GLuint texture);
GLAPI PFNGLBINDTEXTUREUNITPROC glad_glBindTextureUnit;
#define glBindTextureUnit glad_glBindTextureUnit
typedef void (APIENTRYP PFNGLGETTEXTUREIMAGEPROC)(GLuint texture, GLint level, GLenum format, GLenum type, GLsizei bufSize, void *pixels);
GLAPI PFNGLGETTEXTUREIMAGEPROC glad_glGetTextureImage;
#define glGetTextureImage glad_glGetTextureImage
typedef void (APIENTRYP PFNGLGETCOMPRESSEDTEXTUREIMAGEPROC)(GLuint texture, GLint level, GLsizei bufSize, void *pixels);
GLAPI PFNGLGETCOMPRESSEDTEXTUREIMAGEPROC glad_glGetCompressedTextureImage;
#define glGetCompressedTextureImage glad_glGetCompressedTextureImage
typedef void (APIENTRYP PFNGLGETTEXTURELEVELPARAMETERFVPROC)(GLuint texture, GLint level, GLenum pname, GLfloat *params);
GLAPI PFNGLGETTEXTURELEVELPARAMETERFVPROC glad_glGetTextureLevelParameterfv;
#define glGetTextureLevelParameterfv glad_glGetTextureLevelParameterfv
typedef void (APIENTRYP PFNGLGETTEXTURELEVELPARAMETERIVPROC)(GLuint texture, GLint level, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXTURELEVELPARAMETERIVPROC glad_glGetTextureLevelParameteriv;
#define glGetTextureLevelParameteriv glad_glGetTextureLevelParameteriv
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERFVPROC)(GLuint texture, GLenum pname, GLfloat *params);
GLAPI PFNGLGETTEXTUREPARAMETERFVPROC glad_glGetTextureParameterfv;
#define glGetTextureParameterfv glad_glGetTextureParameterfv
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERIIVPROC)(GLuint texture, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXTUREPARAMETERIIVPROC glad_glGetTextureParameterIiv;
#define glGetTextureParameterIiv glad_glGetTextureParameterIiv
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERIUIVPROC)(GLuint texture, GLenum pname, GLuint *params);
GLAPI PFNGLGETTEXTUREPARAMETERIUIVPROC glad_glGetTextureParameterIuiv;
#define glGetTextureParameterIuiv glad_glGetTextureParameterIuiv
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERIVPROC)(GLuint texture, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXTUREPARAMETERIVPROC glad_glGetTextureParameteriv;
#define glGetTextureParameteriv glad_glGetTextureParameteriv
typedef void (APIENTRYP PFNGLCREATEVERTEXARRAYSPROC)(GLsizei n, GLuint *arrays);
GLAPI PFNGLCREATEVERTEXARRAYSPROC glad_glCreateVertexArrays;
#define glCreateVertexArrays glad_glCreateVertexArrays
typedef void (APIENTRYP PFNGLDISABLEVERTEXARRAYATTRIBPROC)(GLuint vaobj, GLuint index);
GLAPI PFNGLDISABLEVERTEXARRAYATTRIBPROC glad_glDisableVertexArrayAttrib;
#define glDisableVertexArrayAttrib glad_glDisableVertexArrayAttrib
typedef void (APIENTRYP PFNGLENABLEVERTEXARRAYATTRIBPROC)(GLuint vaobj, GLuint index);
```

```
GLAPI PFNGLENABLEVERTEXARRAYATTRIBPROC glad_glEnableVertexArrayAttrib;
#define glEnableVertexArrayAttrib glad_glEnableVertexArrayAttrib
typedef void (APIENTRYP PFNGLVERTEXARRAYELEMENTBUFFERPROC)(GLuint vaobj, GLuint buffer);
GLAPI PFNGLVERTEXARRAYELEMENTBUFFERPROC glad_glVertexArrayElementBuffer;
#define glVertexArrayElementBuffer glad_glVertexArrayElementBuffer
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXBUFFERPROC)(GLuint vaobj, GLuint bindingindex, GLuint buffer, GLintptr offset, GLsizei stride);
GLAPI PFNGLVERTEXARRAYVERTEXBUFFERPROC glad_glVertexArrayVertexBuffer;
#define glVertexArrayVertexBuffer glad_glVertexArrayVertexBuffer
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXBUFFERSPROC)(GLuint vaobj, GLuint first, GLsizei count, const GLuint *buffers, const GLintptr *offsets, const GLsizei *strides);
GLAPI PFNGLVERTEXARRAYVERTEXBUFFERSPROC glad_glVertexArrayVertexBuffers;
#define glVertexArrayVertexBuffers glad_glVertexArrayVertexBuffers
typedef void (APIENTRYP PFNGLVERTEXARRAYATTRIBBINDINGPROC)(GLuint vaobj, GLuint attribindex, GLuint bindingindex);
GLAPI PFNGLVERTEXARRAYATTRIBBINDINGPROC glad_glVertexArrayAttribBinding;
#define glVertexArrayAttribBinding glad_glVertexArrayAttribBinding
typedef void (APIENTRYP PFNGLVERTEXARRAYATTRIBFORMATPROC)(GLuint vaobj, GLuint attribindex, GLint size, GLenum type, GLboolean normalized, GLuint relativeoffset);
GLAPI PFNGLVERTEXARRAYATTRIBFORMATPROC glad_glVertexArrayAttribFormat;
#define glVertexArrayAttribFormat glad_glVertexArrayAttribFormat
typedef void (APIENTRYP PFNGLVERTEXARRAYATTRIBIFORMATPROC)(GLuint vaobj, GLuint attribindex, GLint size, GLenum type, GLuint relativeoffset);
GLAPI PFNGLVERTEXARRAYATTRIBIFORMATPROC glad_glVertexArrayAttribIFormat;
#define glVertexArrayAttribIFormat glad_glVertexArrayAttribIFormat
typedef void (APIENTRYP PFNGLVERTEXARRAYATTRIBLFORMATPROC)(GLuint vaobj, GLuint attribindex, GLint size, GLenum type, GLuint relativeoffset);
GLAPI PFNGLVERTEXARRAYATTRIBLFORMATPROC glad_glVertexArrayAttribLFormat;
#define glVertexArrayAttribLFormat glad_glVertexArrayAttribLFormat
typedef void (APIENTRYP PFNGLVERTEXARRAYBINDINGDIVISORPROC)(GLuint vaobj, GLuint bindingindex, GLuint divisor);
GLAPI PFNGLVERTEXARRAYBINDINGDIVISORPROC glad_glVertexArrayBindingDivisor;
#define glVertexArrayBindingDivisor glad_glVertexArrayBindingDivisor
typedef void (APIENTRYP PFNGLGETVERTEXARRAYIVPROC)(GLuint vaobj, GLenum pname, GLint *param);
GLAPI PFNGLGETVERTEXARRAYIVPROC glad_glGetVertexArrayiv;
#define glGetVertexArrayiv glad_glGetVertexArrayiv
typedef void (APIENTRYP PFNGLGETVERTEXARRAYINDEXEDIVPROC)(GLuint vaobj, GLuint index, GLenum pname, GLint *param);
GLAPI PFNGLGETVERTEXARRAYINDEXEDIVPROC glad_glGetVertexArrayIndexediv;
#define glGetVertexArrayIndexediv glad_glGetVertexArrayIndexediv
typedef void (APIENTRYP PFNGLGETVERTEXARRAYINDEXED64IVPROC)(GLuint vaobj, GLuint index, GLenum pname, GLint64 *param);
GLAPI PFNGLGETVERTEXARRAYINDEXED64IVPROC glad_glGetVertexArrayIndexed64iv;
#define glGetVertexArrayIndexed64iv glad_glGetVertexArrayIndexed64iv
typedef void (APIENTRYP PFNGLCREATESAMPLERSPROC)(GLsizei n, GLuint *samplers);
GLAPI PFNGLCREATESAMPLERSPROC glad_glCreateSamplers;
#define glCreateSamplers glad_glCreateSamplers
typedef void (APIENTRYP PFNGLCREATEPROGRAMPIPELINESPROC)(GLsizei n, GLuint *pipelines);
GLAPI PFNGLCREATEPROGRAMPIPELINESPROC glad_glCreateProgramPipelines;
#define glCreateProgramPipelines glad_glCreateProgramPipelines
typedef void (APIENTRYP PFNGLCREATEQUERIESPROC)(GLenum target, GLsizei n, GLuint *ids);
GLAPI PFNGLCREATEQUERIESPROC glad_glCreateQueries;
#define glCreateQueries glad_glCreateQueries
typedef void (APIENTRYP PFNGLGETQUERYBUFFEROBJECTI64VPROC)(GLuint id, GLuint buffer, GLenum pname, GLintptr offset);
GLAPI PFNGLGETQUERYBUFFEROBJECTI64VPROC glad_glGetQueryBufferObjecti64v;
#define glGetQueryBufferObjecti64v glad_glGetQueryBufferObjecti64v
typedef void (APIENTRYP PFNGLGETQUERYBUFFEROBJECTIVPROC)(GLuint id, GLuint buffer, GLenum pname, GLintptr offset);
GLAPI PFNGLGETQUERYBUFFEROBJECTIVPROC glad_glGetQueryBufferObjectiv;
#define glGetQueryBufferObjectiv glad_glGetQueryBufferObjectiv
typedef void (APIENTRYP PFNGLGETQUERYBUFFEROBJECTUI64VPROC)(GLuint id, GLuint buffer, GLenum pname, GLintptr offset);
GLAPI PFNGLGETQUERYBUFFEROBJECTUI64VPROC glad_glGetQueryBufferObjectui64v;
#define glGetQueryBufferObjectui64v glad_glGetQueryBufferObjectui64v
typedef void (APIENTRYP PFNGLGETQUERYBUFFEROBJECTUIVPROC)(GLuint id, GLuint buffer, GLenum pname, GLintptr offset);
GLAPI PFNGLGETQUERYBUFFEROBJECTUIVPROC glad_glGetQueryBufferObjectuiv;
#define glGetQueryBufferObjectuiv glad_glGetQueryBufferObjectuiv
#endif
#ifndef GL_ARB_draw_buffers
#define GL_ARB_draw_buffers 1
GLAPI int GLAD_GL_ARB_draw_buffers;
typedef void (APIENTRYP PFNGLDRAWBUFFERSARBPROC)(GLsizei n, const GLenum *bufs);
GLAPI PFNGLDRAWBUFFERSARBPROC glad_glDrawBuffersARB;
#define glDrawBuffersARB glad_glDrawBuffersARB
#endif
#ifndef GL_ARB_draw_buffers_blend
#define GL_ARB_draw_buffers_blend 1
GLAPI int GLAD_GL_ARB_draw_buffers_blend;
typedef void (APIENTRYP PFNGLBLENDEQUATIONIARBPROC)(GLuint buf, GLenum mode);
GLAPI PFNGLBLENDEQUATIONIARBPROC glad_glBlendEquationiARB;
#define glBlendEquationiARB glad_glBlendEquationiARB
typedef void (APIENTRYP PFNGLBLENDEQUATIONSEPARATEIARBPROC)(GLuint buf, GLenum modeRGB, GLenum modeAlpha);
GLAPI PFNGLBLENDEQUATIONSEPARATEIARBPROC glad_glBlendEquationSeparateiARB;
#define glBlendEquationSeparateiARB glad_glBlendEquationSeparateiARB
```

```
typedef void (APIENTRYP PFNGLBLENDFUNCIARBPROC)(GLuint buf, GLenum src, GLenum dst);
GLAPI PFNGLBLENDFUNCIARBPROC glad_glBlendFunciARB;
#define glBlendFunciARB glad_glBlendFunciARB
typedef void (APIENTRYP PFNGLBLENDFUNCSEPARATEIARBPROC)(GLuint buf, GLenum srcRGB, GLenum dstRGB, GLenum srcAlpha, GLenum dstAlpha);
GLAPI PFNGLBLENDFUNCSEPARATEIARBPROC glad_glBlendFuncSeparateiARB;
#define glBlendFuncSeparateiARB glad_glBlendFuncSeparateiARB
#endif
#ifndef GL_ARB_draw_elements_base_vertex
#define GL_ARB_draw_elements_base_vertex 1
GLAPI int GLAD_GL_ARB_draw_elements_base_vertex;
#endif
#ifndef GL_ARB_draw_indirect
#define GL_ARB_draw_indirect 1
GLAPI int GLAD_GL_ARB_draw_indirect;
#endif
#ifndef GL_ARB_draw_instanced
#define GL_ARB_draw_instanced 1
GLAPI int GLAD_GL_ARB_draw_instanced;
typedef void (APIENTRYP PFNGLDRAWARRAYSINSTANCEDARBPROC)(GLenum mode, GLint first, GLsizei count, GLsizei primcount);
GLAPI PFNGLDRAWARRAYSINSTANCEDARBPROC glad_glDrawArraysInstancedARB;
#define glDrawArraysInstancedARB glad_glDrawArraysInstancedARB
typedef void (APIENTRYP PFNGLDRAWELEMENTSINSTANCEDARBPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices, GLsizei primcount);
GLAPI PFNGLDRAWELEMENTSINSTANCEDARBPROC glad_glDrawElementsInstancedARB;
#define glDrawElementsInstancedARB glad_glDrawElementsInstancedARB
#endif
#ifndef GL_ARB_enhanced_layouts
#define GL_ARB_enhanced_layouts 1
GLAPI int GLAD_GL_ARB_enhanced_layouts;
#endif
#ifndef GL_ARB_explicit_attrib_location
#define GL_ARB_explicit_attrib_location 1
GLAPI int GLAD_GL_ARB_explicit_attrib_location;
#endif
#ifndef GL_ARB_explicit_uniform_location
#define GL_ARB_explicit_uniform_location 1
GLAPI int GLAD_GL_ARB_explicit_uniform_location;
#endif
#ifndef GL_ARB_fragment_coord_conventions
#define GL_ARB_fragment_coord_conventions 1
GLAPI int GLAD_GL_ARB_fragment_coord_conventions;
#endif
#ifndef GL_ARB_fragment_layer_viewport
#define GL_ARB_fragment_layer_viewport 1
GLAPI int GLAD_GL_ARB_fragment_layer_viewport;
#endif
#ifndef GL_ARB_fragment_program
#define GL_ARB_fragment_program 1
GLAPI int GLAD_GL_ARB_fragment_program;
typedef void (APIENTRYP PFNGLPROGRAMSTRINGARBPROC)(GLenum target, GLenum format, GLsizei len, const void *string);
GLAPI PFNGLPROGRAMSTRINGARBPROC glad_glProgramStringARB;
#define glProgramStringARB glad_glProgramStringARB
typedef void (APIENTRYP PFNGLBINDPROGRAMARBPROC)(GLenum target, GLuint program);
GLAPI PFNGLBINDPROGRAMARBPROC glad_glBindProgramARB;
#define glBindProgramARB glad_glBindProgramARB
typedef void (APIENTRYP PFNGLDELETEPROGRAMSARBPROC)(GLsizei n, const GLuint *programs);
GLAPI PFNGLDELETEPROGRAMSARBPROC glad_glDeleteProgramsARB;
#define glDeleteProgramsARB glad_glDeleteProgramsARB
typedef void (APIENTRYP PFNGLGENPROGRAMSARBPROC)(GLsizei n, GLuint *programs);
GLAPI PFNGLGENPROGRAMSARBPROC glad_glGenProgramsARB;
#define glGenProgramsARB glad_glGenProgramsARB
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETER4DARBPROC)(GLenum target, GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLPROGRAMENVPARAMETER4DARBPROC glad_glProgramEnvParameter4dARB;
#define glProgramEnvParameter4dARB glad_glProgramEnvParameter4dARB
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETER4DVARBPROC)(GLenum target, GLuint index, const GLdouble *params);
GLAPI PFNGLPROGRAMENVPARAMETER4DVARBPROC glad_glProgramEnvParameter4dvARB;
#define glProgramEnvParameter4dvARB glad_glProgramEnvParameter4dvARB
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETER4FARBPROC)(GLenum target, GLuint index, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLPROGRAMENVPARAMETER4FARBPROC glad_glProgramEnvParameter4fARB;
#define glProgramEnvParameter4fARB glad_glProgramEnvParameter4fARB
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETER4FVARBPROC)(GLenum target, GLuint index, const GLfloat *params);
GLAPI PFNGLPROGRAMENVPARAMETER4FVARBPROC glad_glProgramEnvParameter4fvARB;
#define glProgramEnvParameter4fvARB glad_glProgramEnvParameter4fvARB
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETER4DARBPROC)(GLenum target, GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLPROGRAMLOCALPARAMETER4DARBPROC glad_glProgramLocalParameter4dARB;
#define glProgramLocalParameter4dARB glad_glProgramLocalParameter4dARB
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETER4DVARBPROC)(GLenum target, GLuint index, const GLdouble *params);
GLAPI PFNGLPROGRAMLOCALPARAMETER4DVARBPROC glad_glProgramLocalParameter4dvARB;
```

```
#define glProgramLocalParameter4dvARB glad_glProgramLocalParameter4dvARB
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETER4FARBPROC)(GLenum target, GLuint index, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLPROGRAMLOCALPARAMETER4FARBPROC glad_glProgramLocalParameter4fARB;
#define glProgramLocalParameter4fARB glad_glProgramLocalParameter4fARB
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETER4FVARBPROC)(GLenum target, GLuint index, const GLfloat *params);
GLAPI PFNGLPROGRAMLOCALPARAMETER4FVARBPROC glad_glProgramLocalParameter4fvARB;
#define glProgramLocalParameter4fvARB glad_glProgramLocalParameter4fvARB
typedef void (APIENTRYP PFNGLGETPROGRAMENVPARAMETERDVARBPROC)(GLenum target, GLuint index, GLdouble *params);
GLAPI PFNGLGETPROGRAMENVPARAMETERDVARBPROC glad_glGetProgramEnvParameterdvARB;
#define glGetProgramEnvParameterdvARB glad_glGetProgramEnvParameterdvARB
typedef void (APIENTRYP PFNGLGETPROGRAMENVPARAMETERFVARBPROC)(GLenum target, GLuint index, GLfloat *params);
GLAPI PFNGLGETPROGRAMENVPARAMETERFVARBPROC glad_glGetProgramEnvParameterfvARB;
#define glGetProgramEnvParameterfvARB glad_glGetProgramEnvParameterfvARB
typedef void (APIENTRYP PFNGLGETPROGRAMLOCALPARAMETERDVARBPROC)(GLenum target, GLuint index, GLdouble *params);
GLAPI PFNGLGETPROGRAMLOCALPARAMETERDVARBPROC glad_glGetProgramLocalParameterdvARB;
#define glGetProgramLocalParameterdvARB glad_glGetProgramLocalParameterdvARB
typedef void (APIENTRYP PFNGLGETPROGRAMLOCALPARAMETERFVARBPROC)(GLenum target, GLuint index, GLfloat *params);
GLAPI PFNGLGETPROGRAMLOCALPARAMETERFVARBPROC glad_glGetProgramLocalParameterfvARB;
#define glGetProgramLocalParameterfvARB glad_glGetProgramLocalParameterfvARB
typedef void (APIENTRYP PFNGLGETPROGRAMIVARBPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETPROGRAMIVARBPROC glad_glGetProgramivARB;
#define glGetProgramivARB glad_glGetProgramivARB
typedef void (APIENTRYP PFNGLGETPROGRAMSTRINGARBPROC)(GLenum target, GLenum pname, void *string);
GLAPI PFNGLGETPROGRAMSTRINGARBPROC glad_glGetProgramStringARB;
#define glGetProgramStringARB glad_glGetProgramStringARB
typedef GLboolean (APIENTRYP PFNGLISPROGRAMARBPROC)(GLuint program);
GLAPI PFNGLISPROGRAMARBPROC glad_glIsProgramARB;
#define glIsProgramARB glad_glIsProgramARB
#endif
#ifndef GL_ARB_fragment_program_shadow
#define GL_ARB_fragment_program_shadow 1
GLAPI int GLAD_GL_ARB_fragment_program_shadow;
#endif
#ifndef GL_ARB_fragment_shader
#define GL_ARB_fragment_shader 1
GLAPI int GLAD_GL_ARB_fragment_shader;
#endif
#ifndef GL_ARB_fragment_shader_interlock
#define GL_ARB_fragment_shader_interlock 1
GLAPI int GLAD_GL_ARB_fragment_shader_interlock;
#endif
#ifndef GL_ARB_framebuffer_no_attachments
#define GL_ARB_framebuffer_no_attachments 1
GLAPI int GLAD_GL_ARB_framebuffer_no_attachments;
typedef void (APIENTRYP PFNGLFRAMEBUFFERPARAMETERIPROC)(GLenum target, GLenum pname, GLint param);
GLAPI PFNGLFRAMEBUFFERPARAMETERIPROC glad_glFramebufferParameteri;
#define glFramebufferParameteri glad_glFramebufferParameteri
typedef void (APIENTRYP PFNGLGETFRAMEBUFFERPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETFRAMEBUFFERPARAMETERIVPROC glad_glGetFramebufferParameteriv;
#define glGetFramebufferParameteriv glad_glGetFramebufferParameteriv
#endif
#ifndef GL_ARB_framebuffer_object
#define GL_ARB_framebuffer_object 1
GLAPI int GLAD_GL_ARB_framebuffer_object;
#endif
#ifndef GL_ARB_framebuffer_sRGB
#define GL_ARB_framebuffer_sRGB 1
GLAPI int GLAD_GL_ARB_framebuffer_sRGB;
#endif
#ifndef GL_ARB_geometry_shader4
#define GL_ARB_geometry_shader4 1
GLAPI int GLAD_GL_ARB_geometry_shader4;
typedef void (APIENTRYP PFNGLPROGRAMPARAMETERIARBPROC)(GLuint program, GLenum pname, GLint value);
GLAPI PFNGLPROGRAMPARAMETERIARBPROC glad_glProgramParameteriARB;
#define glProgramParameteriARB glad_glProgramParameteriARB
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTUREARBPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level);
GLAPI PFNGLFRAMEBUFFERTEXTUREARBPROC glad_glFramebufferTextureARB;
#define glFramebufferTextureARB glad_glFramebufferTextureARB
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURELAYERARBPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level, GLint layer);
GLAPI PFNGLFRAMEBUFFERTEXTURELAYERARBPROC glad_glFramebufferTextureLayerARB;
#define glFramebufferTextureLayerARB glad_glFramebufferTextureLayerARB
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTUREFACEARBPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level, GLenum face);
GLAPI PFNGLFRAMEBUFFERTEXTUREFACEARBPROC glad_glFramebufferTextureFaceARB;
#define glFramebufferTextureFaceARB glad_glFramebufferTextureFaceARB
#endif
#ifndef GL_ARB_get_program_binary
#define GL_ARB_get_program_binary 1
```

```
GLAPI int GLAD_GL_ARB_get_program_binary;
#endif
#ifndef GL_ARB_get_texture_sub_image
#define GL_ARB_get_texture_sub_image 1
GLAPI int GLAD_GL_ARB_get_texture_sub_image;
typedef void (APIENTRYP 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);
GLAPI PFNGLGETTEXTURESUBIMAGEPROC glad_glGetTextureSubImage;
#define glGetTextureSubImage glad_glGetTextureSubImage
typedef void (APIENTRYP PFNGLGETCOMPRESSEDTEXTURESUBIMAGEPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLsizei bufSize, void *pixels);
GLAPI PFNGLGETCOMPRESSEDTEXTURESUBIMAGEPROC glad_glGetCompressedTextureSubImage;
#define glGetCompressedTextureSubImage glad_glGetCompressedTextureSubImage
#endif
#ifndef GL_ARB_gl_spirv
#define GL_ARB_gl_spirv 1
GLAPI int GLAD_GL_ARB_gl_spirv;
typedef void (APIENTRYP PFNGLSPECIALIZESHADERARBPROC)(GLuint shader, const GLchar *pEntryPoint, GLuint numSpecializationConstants, const GLuint *pConstantIndex, const GLuint *pConstantValue);
GLAPI PFNGLSPECIALIZESHADERARBPROC glad_glSpecializeShaderARB;
#define glSpecializeShaderARB glad_glSpecializeShaderARB
#endif
#ifndef GL_ARB_gpu_shader5
#define GL_ARB_gpu_shader5 1
GLAPI int GLAD_GL_ARB_gpu_shader5;
#endif
#ifndef GL_ARB_gpu_shader_fp64
#define GL_ARB_gpu_shader_fp64 1
GLAPI int GLAD_GL_ARB_gpu_shader_fp64;
#endif
#ifndef GL_ARB_gpu_shader_int64
#define GL_ARB_gpu_shader_int64 1
GLAPI int GLAD_GL_ARB_gpu_shader_int64;
typedef void (APIENTRYP PFNGLUNIFORM1I64ARBPROC)(GLint location, GLint64 x);
GLAPI PFNGLUNIFORM1I64ARBPROC glad_glUniform1i64ARB;
#define glUniform1i64ARB glad_glUniform1i64ARB
typedef void (APIENTRYP PFNGLUNIFORM2I64ARBPROC)(GLint location, GLint64 x, GLint64 y);
GLAPI PFNGLUNIFORM2I64ARBPROC glad_glUniform2i64ARB;
#define glUniform2i64ARB glad_glUniform2i64ARB
typedef void (APIENTRYP PFNGLUNIFORM3I64ARBPROC)(GLint location, GLint64 x, GLint64 y, GLint64 z);
GLAPI PFNGLUNIFORM3I64ARBPROC glad_glUniform3i64ARB;
#define glUniform3i64ARB glad_glUniform3i64ARB
typedef void (APIENTRYP PFNGLUNIFORM4I64ARBPROC)(GLint location, GLint64 x, GLint64 y, GLint64 z, GLint64 w);
GLAPI PFNGLUNIFORM4I64ARBPROC glad_glUniform4i64ARB;
#define glUniform4i64ARB glad_glUniform4i64ARB
typedef void (APIENTRYP PFNGLUNIFORM1I64VARBPROC)(GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLUNIFORM1I64VARBPROC glad_glUniform1i64vARB;
#define glUniform1i64vARB glad_glUniform1i64vARB
typedef void (APIENTRYP PFNGLUNIFORM2I64VARBPROC)(GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLUNIFORM2I64VARBPROC glad_glUniform2i64vARB;
#define glUniform2i64vARB glad_glUniform2i64vARB
typedef void (APIENTRYP PFNGLUNIFORM3I64VARBPROC)(GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLUNIFORM3I64VARBPROC glad_glUniform3i64vARB;
#define glUniform3i64vARB glad_glUniform3i64vARB
typedef void (APIENTRYP PFNGLUNIFORM4I64VARBPROC)(GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLUNIFORM4I64VARBPROC glad_glUniform4i64vARB;
#define glUniform4i64vARB glad_glUniform4i64vARB
typedef void (APIENTRYP PFNGLUNIFORM1UI64ARBPROC)(GLint location, GLuint64 x);
GLAPI PFNGLUNIFORM1UI64ARBPROC glad_glUniform1ui64ARB;
#define glUniform1ui64ARB glad_glUniform1ui64ARB
typedef void (APIENTRYP PFNGLUNIFORM2UI64ARBPROC)(GLint location, GLuint64 x, GLuint64 y);
GLAPI PFNGLUNIFORM2UI64ARBPROC glad_glUniform2ui64ARB;
#define glUniform2ui64ARB glad_glUniform2ui64ARB
typedef void (APIENTRYP PFNGLUNIFORM3UI64ARBPROC)(GLint location, GLuint64 x, GLuint64 y, GLuint64 z);
GLAPI PFNGLUNIFORM3UI64ARBPROC glad_glUniform3ui64ARB;
#define glUniform3ui64ARB glad_glUniform3ui64ARB
typedef void (APIENTRYP PFNGLUNIFORM4UI64ARBPROC)(GLint location, GLuint64 x, GLuint64 y, GLuint64 z, GLuint64 w);
GLAPI PFNGLUNIFORM4UI64ARBPROC glad_glUniform4ui64ARB;
#define glUniform4ui64ARB glad_glUniform4ui64ARB
typedef void (APIENTRYP PFNGLUNIFORM1UI64VARBPROC)(GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLUNIFORM1UI64VARBPROC glad_glUniform1ui64vARB;
#define glUniform1ui64vARB glad_glUniform1ui64vARB
typedef void (APIENTRYP PFNGLUNIFORM2UI64VARBPROC)(GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLUNIFORM2UI64VARBPROC glad_glUniform2ui64vARB;
#define glUniform2ui64vARB glad_glUniform2ui64vARB
typedef void (APIENTRYP PFNGLUNIFORM3UI64VARBPROC)(GLint location, GLsizei count, const GLuint64 *value);
```

```
GLAPI PFNGLUNIFORM3UI64VARBPROC glad_glUniform3ui64vARB;
#define glUniform3ui64vARB glad_glUniform3ui64vARB
typedef void (APIENTRYP PFNGLUNIFORM4UI64VARBPROC)(GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLUNIFORM4UI64VARBPROC glad_glUniform4ui64vARB;
#define glUniform4ui64vARB glad_glUniform4ui64vARB
typedef void (APIENTRYP PFNGLGETUNIFORMI64VARBPROC)(GLuint program, GLint location, GLint64 *params);
GLAPI PFNGLGETUNIFORMI64VARBPROC glad_glGetUniformi64vARB;
#define glGetUniformi64vARB glad_glGetUniformi64vARB
typedef void (APIENTRYP PFNGLGETUNIFORMUI64VARBPROC)(GLuint program, GLint location, GLuint64 *params);
GLAPI PFNGLGETUNIFORMUI64VARBPROC glad_glGetUniformui64vARB;
#define glGetUniformui64vARB glad_glGetUniformui64vARB
typedef void (APIENTRYP PFNGLGETNUNIFORMI64VARBPROC)(GLuint program, GLint location, GLsizei bufSize, GLint64 *params);
GLAPI PFNGLGETNUNIFORMI64VARBPROC glad_glGetnUniformi64vARB;
#define glGetnUniformi64vARB glad_glGetnUniformi64vARB
typedef void (APIENTRYP PFNGLGETNUNIFORMUI64VARBPROC)(GLuint program, GLint location, GLsizei bufSize, GLuint64 *params);
GLAPI PFNGLGETNUNIFORMUI64VARBPROC glad_glGetnUniformui64vARB;
#define glGetnUniformui64vARB glad_glGetnUniformui64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1I64ARBPROC)(GLuint program, GLint location, GLint64 x);
GLAPI PFNGLPROGRAMUNIFORM1I64ARBPROC glad_glProgramUniform1i64ARB;
#define glProgramUniform1i64ARB glad_glProgramUniform1i64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2I64ARBPROC)(GLuint program, GLint location, GLint64 x, GLint64 y);
GLAPI PFNGLPROGRAMUNIFORM2I64ARBPROC glad_glProgramUniform2i64ARB;
#define glProgramUniform2i64ARB glad_glProgramUniform2i64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3I64ARBPROC)(GLuint program, GLint location, GLint64 x, GLint64 y, GLint64 z);
GLAPI PFNGLPROGRAMUNIFORM3I64ARBPROC glad_glProgramUniform3i64ARB;
#define glProgramUniform3i64ARB glad_glProgramUniform3i64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4I64ARBPROC)(GLuint program, GLint location, GLint64 x, GLint64 y, GLint64 z, GLint64 w);
GLAPI PFNGLPROGRAMUNIFORM4I64ARBPROC glad_glProgramUniform4i64ARB;
#define glProgramUniform4i64ARB glad_glProgramUniform4i64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1I64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLPROGRAMUNIFORM1I64VARBPROC glad_glProgramUniform1i64vARB;
#define glProgramUniform1i64vARB glad_glProgramUniform1i64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2I64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLPROGRAMUNIFORM2I64VARBPROC glad_glProgramUniform2i64vARB;
#define glProgramUniform2i64vARB glad_glProgramUniform2i64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3I64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLPROGRAMUNIFORM3I64VARBPROC glad_glProgramUniform3i64vARB;
#define glProgramUniform3i64vARB glad_glProgramUniform3i64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4I64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLint64 *value);
GLAPI PFNGLPROGRAMUNIFORM4I64VARBPROC glad_glProgramUniform4i64vARB;
#define glProgramUniform4i64vARB glad_glProgramUniform4i64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UI64ARBPROC)(GLuint program, GLint location, GLuint64 x);
GLAPI PFNGLPROGRAMUNIFORM1UI64ARBPROC glad_glProgramUniform1ui64ARB;
#define glProgramUniform1ui64ARB glad_glProgramUniform1ui64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UI64ARBPROC)(GLuint program, GLint location, GLuint64 x, GLuint64 y);
GLAPI PFNGLPROGRAMUNIFORM2UI64ARBPROC glad_glProgramUniform2ui64ARB;
#define glProgramUniform2ui64ARB glad_glProgramUniform2ui64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UI64ARBPROC)(GLuint program, GLint location, GLuint64 x, GLuint64 y, GLuint64 z);
GLAPI PFNGLPROGRAMUNIFORM3UI64ARBPROC glad_glProgramUniform3ui64ARB;
#define glProgramUniform3ui64ARB glad_glProgramUniform3ui64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UI64ARBPROC)(GLuint program, GLint location, GLuint64 x, GLuint64 y, GLuint64 z, GLuint64 w);
GLAPI PFNGLPROGRAMUNIFORM4UI64ARBPROC glad_glProgramUniform4ui64ARB;
#define glProgramUniform4ui64ARB glad_glProgramUniform4ui64ARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UI64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLPROGRAMUNIFORM1UI64VARBPROC glad_glProgramUniform1ui64vARB;
#define glProgramUniform1ui64vARB glad_glProgramUniform1ui64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UI64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLPROGRAMUNIFORM2UI64VARBPROC glad_glProgramUniform2ui64vARB;
#define glProgramUniform2ui64vARB glad_glProgramUniform2ui64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UI64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLPROGRAMUNIFORM3UI64VARBPROC glad_glProgramUniform3ui64vARB;
#define glProgramUniform3ui64vARB glad_glProgramUniform3ui64vARB
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UI64VARBPROC)(GLuint program, GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLPROGRAMUNIFORM4UI64VARBPROC glad_glProgramUniform4ui64vARB;
#define glProgramUniform4ui64vARB glad_glProgramUniform4ui64vARB
#endif
#ifndef GL_ARB_half_float_pixel
#define GL_ARB_half_float_pixel 1
GLAPI int GLAD_GL_ARB_half_float_pixel;
```

```
#endif
#ifndef GL_ARB_half_float_vertex
#define GL_ARB_half_float_vertex 1
GLAPI int GLAD_GL_ARB_half_float_vertex;
#endif
#ifndef GL_ARB_imaging
#define GL_ARB_imaging 1
GLAPI int GLAD_GL_ARB_imaging;
typedef void (APIENTRYP PFNGLCOLORTABLEPROC)(GLenum target, GLenum internalformat, GLsizei width, GLenum format, GLenum type, const void *table);
GLAPI PFNGLCOLORTABLEPROC glad_glColorTable;
#define glColorTable glad_glColorTable
typedef void (APIENTRYP PFNGLCOLORTABLEPARAMETERFVPROC)(GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLCOLORTABLEPARAMETERFVPROC glad_glColorTableParameterfv;
#define glColorTableParameterfv glad_glColorTableParameterfv
typedef void (APIENTRYP PFNGLCOLORTABLEPARAMETERIVPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLCOLORTABLEPARAMETERIVPROC glad_glColorTableParameteriv;
#define glColorTableParameteriv glad_glColorTableParameteriv
typedef void (APIENTRYP PFNGLCOPYCOLORTABLEPROC)(GLenum target, GLenum internalformat, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYCOLORTABLEPROC glad_glCopyColorTable;
#define glCopyColorTable glad_glCopyColorTable
typedef void (APIENTRYP PFNGLGETCOLORTABLEPROC)(GLenum target, GLenum format, GLenum type, void *table);
GLAPI PFNGLGETCOLORTABLEPROC glad_glGetColorTable;
#define glGetColorTable glad_glGetColorTable
typedef void (APIENTRYP PFNGLGETCOLORTABLEPARAMETERFVPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCOLORTABLEPARAMETERFVPROC glad_glGetColorTableParameterfv;
#define glGetColorTableParameterfv glad_glGetColorTableParameterfv
typedef void (APIENTRYP PFNGLGETCOLORTABLEPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETCOLORTABLEPARAMETERIVPROC glad_glGetColorTableParameteriv;
#define glGetColorTableParameteriv glad_glGetColorTableParameteriv
typedef void (APIENTRYP PFNGLCOLORSUBTABLEPROC)(GLenum target, GLsizei start, GLsizei count, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCOLORSUBTABLEPROC glad_glColorSubTable;
#define glColorSubTable glad_glColorSubTable
typedef void (APIENTRYP PFNGLCOPYCOLORSUBTABLEPROC)(GLenum target, GLsizei start, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYCOLORSUBTABLEPROC glad_glCopyColorSubTable;
#define glCopyColorSubTable glad_glCopyColorSubTable
typedef void (APIENTRYP PFNGLCONVOLUTIONFILTER1DPROC)(GLenum target, GLenum internalformat, GLsizei width, GLenum format, GLenum type, const void *image);
GLAPI PFNGLCONVOLUTIONFILTER1DPROC glad_glConvolutionFilter1D;
#define glConvolutionFilter1D glad_glConvolutionFilter1D
typedef void (APIENTRYP PFNGLCONVOLUTIONFILTER2DPROC)(GLenum target, GLenum internalformat, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *image);
GLAPI PFNGLCONVOLUTIONFILTER2DPROC glad_glConvolutionFilter2D;
#define glConvolutionFilter2D glad_glConvolutionFilter2D
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERFPROC)(GLenum target, GLenum pname, GLfloat params);
GLAPI PFNGLCONVOLUTIONPARAMETERFPROC glad_glConvolutionParameterf;
#define glConvolutionParameterf glad_glConvolutionParameterf
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERFVPROC)(GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLCONVOLUTIONPARAMETERFVPROC glad_glConvolutionParameterfv;
#define glConvolutionParameterfv glad_glConvolutionParameterfv
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERIPROC)(GLenum target, GLenum pname, GLint params);
GLAPI PFNGLCONVOLUTIONPARAMETERIPROC glad_glConvolutionParameteri;
#define glConvolutionParameteri glad_glConvolutionParameteri
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERIVPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLCONVOLUTIONPARAMETERIVPROC glad_glConvolutionParameteriv;
#define glConvolutionParameteriv glad_glConvolutionParameteriv
typedef void (APIENTRYP PFNGLCOPYCONVOLUTIONFILTER1DPROC)(GLenum target, GLenum internalformat, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYCONVOLUTIONFILTER1DPROC glad_glCopyConvolutionFilter1D;
#define glCopyConvolutionFilter1D glad_glCopyConvolutionFilter1D
typedef void (APIENTRYP PFNGLCOPYCONVOLUTIONFILTER2DPROC)(GLenum target, GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYCONVOLUTIONFILTER2DPROC glad_glCopyConvolutionFilter2D;
#define glCopyConvolutionFilter2D glad_glCopyConvolutionFilter2D
typedef void (APIENTRYP PFNGLGETCONVOLUTIONFILTERPROC)(GLenum target, GLenum format, GLenum type, void *image);
GLAPI PFNGLGETCONVOLUTIONFILTERPROC glad_glGetConvolutionFilter;
#define glGetConvolutionFilter glad_glGetConvolutionFilter
typedef void (APIENTRYP PFNGLGETCONVOLUTIONPARAMETERFVPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCONVOLUTIONPARAMETERFVPROC glad_glGetConvolutionParameterfv;
#define glGetConvolutionParameterfv glad_glGetConvolutionParameterfv
typedef void (APIENTRYP PFNGLGETCONVOLUTIONPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETCONVOLUTIONPARAMETERIVPROC glad_glGetConvolutionParameteriv;
#define glGetConvolutionParameteriv glad_glGetConvolutionParameteriv
typedef void (APIENTRYP PFNGLGETSEPARABLEFILTERPROC)(GLenum target, GLenum format, GLenum type, void
*row, void *column, void *span);
GLAPI PFNGLGETSEPARABLEFILTERPROC glad_glGetSeparableFilter;
#define glGetSeparableFilter glad_glGetSeparableFilter
typedef void (APIENTRYP PFNGLSEPARABLEFILTER2DPROC)(GLenum target, GLenum internalformat, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *row, const void *column);
GLAPI PFNGLSEPARABLEFILTER2DPROC glad_glSeparableFilter2D;
#define glSeparableFilter2D glad_glSeparableFilter2D
typedef void (APIENTRYP PFNGLGETHISTOGRAMPROC)(GLenum target, GLboolean reset, GLenum format, GLenum type, void *values);
GLAPI PFNGLGETHISTOGRAMPROC glad_glGetHistogram;
#define glGetHistogram glad_glGetHistogram
typedef void (APIENTRYP PFNGLGETHISTOGRAMPARAMETERFVPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETHISTOGRAMPARAMETERFVPROC glad_glGetHistogramParameterfv;
#define glGetHistogramParameterfv glad_glGetHistogramParameterfv
typedef void (APIENTRYP PFNGLGETHISTOGRAMPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETHISTOGRAMPARAMETERIVPROC glad_glGetHistogramParameteriv;
#define glGetHistogramParameteriv glad_glGetHistogramParameteriv
typedef void (APIENTRYP PFNGLGETMINMAXPROC)(GLenum target, GLboolean reset, GLenum format, GLenum type, void *values);
GLAPI PFNGLGETMINMAXPROC glad_glGetMinmax;
#define glGetMinmax glad_glGetMinmax
typedef void (APIENTRYP PFNGLGETMINMAXPARAMETERFVPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMINMAXPARAMETERFVPROC glad_glGetMinmaxParameterfv;
#define glGetMinmaxParameterfv glad_glGetMinmaxParameterfv
typedef void (APIENTRYP PFNGLGETMINMAXPARAMETERIVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETMINMAXPARAMETERIVPROC glad_glGetMinmaxParameteriv;
#define glGetMinmaxParameteriv glad_glGetMinmaxParameteriv
typedef void (APIENTRYP PFNGLHISTOGRAMPROC)(GLenum target, GLsizei width, GLenum internalformat, GLboolean sink);
GLAPI PFNGLHISTOGRAMPROC glad_glHistogram;
#define glHistogram glad_glHistogram
typedef void (APIENTRYP PFNGLMINMAXPROC)(GLenum target, GLenum internalformat, GLboolean sink);
GLAPI PFNGLMINMAXPROC glad_glMinmax;
#define glMinmax glad_glMinmax
typedef void (APIENTRYP PFNGLRESETHISTOGRAMPROC)(GLenum target);
GLAPI PFNGLRESETHISTOGRAMPROC glad_glResetHistogram;
#define glResetHistogram glad_glResetHistogram
typedef void (APIENTRYP PFNGLRESETMINMAXPROC)(GLenum target);
GLAPI PFNGLRESETMINMAXPROC glad_glResetMinmax;
#define glResetMinmax glad_glResetMinmax
#endif
#ifndef GL_ARB_indirect_parameters
#define GL_ARB_indirect_parameters 1
GLAPI int GLAD_GL_ARB_indirect_parameters;
typedef void (APIENTRYP PFNGLMULTIDRAWARRAYSINDIRECTCOUNTARBPROC)(GLenum mode, const void *indirect, GLintptr drawcount, GLsizei maxdrawcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWARRAYSINDIRECTCOUNTARBPROC glad_glMultiDrawArraysIndirectCountARB;
#define glMultiDrawArraysIndirectCountARB glad_glMultiDrawArraysIndirectCountARB
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSINDIRECTCOUNTARBPROC)(GLenum mode, GLenum type, const void *indirect, GLintptr drawcount, GLsizei maxdrawcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTCOUNTARBPROC glad_glMultiDrawElementsIndirectCountARB;
#define glMultiDrawElementsIndirectCountARB glad_glMultiDrawElementsIndirectCountARB
#endif
#ifndef GL_ARB_instanced_arrays
#define GL_ARB_instanced_arrays 1
GLAPI int GLAD_GL_ARB_instanced_arrays;
typedef void (APIENTRYP PFNGLVERTEXATTRIBDIVISORARBPROC)(GLuint index, GLuint divisor);
GLAPI PFNGLVERTEXATTRIBDIVISORARBPROC glad_glVertexAttribDivisorARB;
#define glVertexAttribDivisorARB glad_glVertexAttribDivisorARB
#endif
#ifndef GL_ARB_internalformat_query
#define GL_ARB_internalformat_query 1
GLAPI int GLAD_GL_ARB_internalformat_query;
typedef void (APIENTRYP PFNGLGETINTERNALFORMATIVPROC)(GLenum target, GLenum internalformat, GLenum pname, GLsizei count, GLint *params);
GLAPI PFNGLGETINTERNALFORMATIVPROC glad_glGetInternalformativ;
#define glGetInternalformativ glad_glGetInternalformativ
#endif
#ifndef GL_ARB_internalformat_query2
#define GL_ARB_internalformat_query2 1
GLAPI int GLAD_GL_ARB_internalformat_query2;
typedef void (APIENTRYP PFNGLGETINTERNALFORMATI64VPROC)(GLenum target, GLenum internalformat, GLenum pname, GLsizei count, GLint64 *params);
GLAPI PFNGLGETINTERNALFORMATI64VPROC glad_glGetInternalformati64v;
#define glGetInternalformati64v glad_glGetInternalformati64v
#endif
#ifndef GL_ARB_invalidate_subdata
#define GL_ARB_invalidate_subdata 1
GLAPI int GLAD_GL_ARB_invalidate_subdata;
typedef void (APIENTRYP PFNGLINVALIDATETEXSUBIMAGEPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth);
GLAPI PFNGLINVALIDATETEXSUBIMAGEPROC glad_glInvalidateTexSubImage;
#define glInvalidateTexSubImage glad_glInvalidateTexSubImage
typedef void (APIENTRYP PFNGLINVALIDATETEXIMAGEPROC)(GLuint texture, GLint level);
GLAPI PFNGLINVALIDATETEXIMAGEPROC glad_glInvalidateTexImage;
#define glInvalidateTexImage glad_glInvalidateTexImage
```

```
typedef void (APIENTRYP PFNGLINVALIDATEBUFFERSUBDATAPROC)(GLuint buffer, GLintptr offset, GLsizeiptr length);
GLAPI PFNGLINVALIDATEBUFFERSUBDATAPROC glad_glInvalidateBufferSubData;
#define glInvalidateBufferSubData glad_glInvalidateBufferSubData
typedef void (APIENTRYP PFNGLINVALIDATEBUFFERDATAPROC)(GLuint buffer);
GLAPI PFNGLINVALIDATEBUFFERDATAPROC glad_glInvalidateBufferData;
#define glInvalidateBufferData glad_glInvalidateBufferData
typedef void (APIENTRYP PFNGLINVALIDATEFRAMEBUFFERPROC)(GLenum target, GLsizei numAttachments, const GLenum *attachments);
GLAPI PFNGLINVALIDATEFRAMEBUFFERPROC glad_glInvalidateFramebuffer;
#define glInvalidateFramebuffer glad_glInvalidateFramebuffer
typedef void (APIENTRYP PFNGLINVALIDATESUBFRAMEBUFFERPROC)(GLenum target, GLsizei numAttachments, const GLenum *attachments, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLINVALIDATESUBFRAMEBUFFERPROC glad_glInvalidateSubFramebuffer;
#define glInvalidateSubFramebuffer glad_glInvalidateSubFramebuffer
#endif
#ifndef GL_ARB_map_buffer_alignment
#define GL_ARB_map_buffer_alignment 1
GLAPI int GLAD_GL_ARB_map_buffer_alignment;
#endif
#ifndef GL_ARB_map_buffer_range
#define GL_ARB_map_buffer_range 1
GLAPI int GLAD_GL_ARB_map_buffer_range;
#endif
#ifndef GL_ARB_matrix_palette
#define GL_ARB_matrix_palette 1
GLAPI int GLAD_GL_ARB_matrix_palette;
typedef void (APIENTRYP PFNGLCURRENTPALETTEMATRIXARBPROC)(GLint index);
GLAPI PFNGLCURRENTPALETTEMATRIXARBPROC glad_glCurrentPaletteMatrixARB;
#define glCurrentPaletteMatrixARB glad_glCurrentPaletteMatrixARB
typedef void (APIENTRYP PFNGLMATRIXINDEXUBVARBPROC)(GLint size, const GLubyte *indices);
GLAPI PFNGLMATRIXINDEXUBVARBPROC glad_glMatrixIndexubvARB;
#define glMatrixIndexubvARB glad_glMatrixIndexubvARB
typedef void (APIENTRYP PFNGLMATRIXINDEXUSVARBPROC)(GLint size, const GLushort *indices);
GLAPI PFNGLMATRIXINDEXUSVARBPROC glad_glMatrixIndexusvARB;
#define glMatrixIndexusvARB glad_glMatrixIndexusvARB
typedef void (APIENTRYP PFNGLMATRIXINDEXUIVARBPROC)(GLint size, const GLuint *indices);
GLAPI PFNGLMATRIXINDEXUIVARBPROC glad_glMatrixIndexuivARB;
#define glMatrixIndexuivARB glad_glMatrixIndexuivARB
typedef void (APIENTRYP PFNGLMATRIXINDEXPOINTERARBPROC)(GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLMATRIXINDEXPOINTERARBPROC glad_glMatrixIndexPointerARB;
#define glMatrixIndexPointerARB glad_glMatrixIndexPointerARB
#endif
#ifndef GL_ARB_multi_bind
#define GL_ARB_multi_bind 1
GLAPI int GLAD_GL_ARB_multi_bind;
typedef void (APIENTRYP PFNGLBINDBUFFERSBASEPROC)(GLenum target, GLuint first, GLsizei count, const GLuint *buffers);
GLAPI PFNGLBINDBUFFERSBASEPROC glad_glBindBuffersBase;
#define glBindBuffersBase glad_glBindBuffersBase
typedef void (APIENTRYP PFNGLBINDBUFFERSRANGEPROC)(GLenum target, GLuint first, GLsizei count, const GLuint *buffers, const GLintptr *offsets, const GLsizeiptr *sizes);
GLAPI PFNGLBINDBUFFERSRANGEPROC glad_glBindBuffersRange;
#define glBindBuffersRange glad_glBindBuffersRange
typedef void (APIENTRYP PFNGLBINDTEXTURESPROC)(GLuint first, GLsizei count, const GLuint *textures);
GLAPI PFNGLBINDTEXTURESPROC glad_glBindTextures;
#define glBindTextures glad_glBindTextures
typedef void (APIENTRYP PFNGLBINDSAMPLERSPROC)(GLuint first, GLsizei count, const GLuint *samplers);
GLAPI PFNGLBINDSAMPLERSPROC glad_glBindSamplers;
#define glBindSamplers glad_glBindSamplers
typedef void (APIENTRYP PFNGLBINDIMAGETEXTURESPROC)(GLuint first, GLsizei count, const GLuint *textures);
GLAPI PFNGLBINDIMAGETEXTURESPROC glad_glBindImageTextures;
#define glBindImageTextures glad_glBindImageTextures
typedef void (APIENTRYP PFNGLBINDVERTEXBUFFERSPROC)(GLuint first, GLsizei count, const GLuint *buffers, const GLintptr *offsets, const GLsizei *strides);
GLAPI PFNGLBINDVERTEXBUFFERSPROC glad_glBindVertexBuffers;
#define glBindVertexBuffers glad_glBindVertexBuffers
#endif
#ifndef GL_ARB_multi_draw_indirect
#define GL_ARB_multi_draw_indirect 1
GLAPI int GLAD_GL_ARB_multi_draw_indirect;
typedef void (APIENTRYP PFNGLMULTIDRAWARRAYSINDIRECTPROC)(GLenum mode, const void *indirect, GLsizei drawcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWARRAYSINDIRECTPROC glad_glMultiDrawArraysIndirect;
#define glMultiDrawArraysIndirect glad_glMultiDrawArraysIndirect
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSINDIRECTPROC)(GLenum mode, GLenum type, const void *indirect, GLsizei drawcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTPROC glad_glMultiDrawElementsIndirect;
#define glMultiDrawElementsIndirect glad_glMultiDrawElementsIndirect
#endif
#ifndef GL_ARB_multisample
#define GL_ARB_multisample 1
GLAPI int GLAD_GL_ARB_multisample;
typedef void (APIENTRYP PFNGLSAMPLECOVERAGEARBPROC)(GLfloat value, GLboolean invert);
GLAPI PFNGLSAMPLECOVERAGEARBPROC glad_glSampleCoverageARB;
```

```
#define glSampleCoverageARB glad_glSampleCoverageARB
#endif
#ifndef GL_ARB_multitexture
#define GL_ARB_multitexture 1
GLAPI int GLAD_GL_ARB_multitexture;
typedef void (APIENTRYP PFNGLACTIVETEXTUREARBPROC)(GLenum texture);
GLAPI PFNGLACTIVETEXTUREARBPROC glad_glActiveTextureARB;
#define glActiveTextureARB glad_glActiveTextureARB
typedef void (APIENTRYP PFNGLCLIENTACTIVETEXTUREARBPROC)(GLenum texture);
GLAPI PFNGLCLIENTACTIVETEXTUREARBPROC glad_glClientActiveTextureARB;
#define glClientActiveTextureARB glad_glClientActiveTextureARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1DARBPROC)(GLenum target, GLdouble s);
GLAPI PFNGLMULTITEXCOORD1DARBPROC glad_glMultiTexCoord1dARB;
#define glMultiTexCoord1dARB glad_glMultiTexCoord1dARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1DVARBPROC)(GLenum target, const GLdouble *v);
GLAPI PFNGLMULTITEXCOORD1DVARBPROC glad_glMultiTexCoord1dvARB;
#define glMultiTexCoord1dvARB glad_glMultiTexCoord1dvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1FARBPROC)(GLenum target, GLfloat s);
GLAPI PFNGLMULTITEXCOORD1FARBPROC glad_glMultiTexCoord1fARB;
#define glMultiTexCoord1fARB glad_glMultiTexCoord1fARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1FVARBPROC)(GLenum target, const GLfloat *v);
GLAPI PFNGLMULTITEXCOORD1FVARBPROC glad_glMultiTexCoord1fvARB;
#define glMultiTexCoord1fvARB glad_glMultiTexCoord1fvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1IARBPROC)(GLenum target, GLint s);
GLAPI PFNGLMULTITEXCOORD1IARBPROC glad_glMultiTexCoord1iARB;
#define glMultiTexCoord1iARB glad_glMultiTexCoord1iARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1IVARBPROC)(GLenum target, const GLint *v);
GLAPI PFNGLMULTITEXCOORD1IVARBPROC glad_glMultiTexCoord1ivARB;
#define glMultiTexCoord1ivARB glad_glMultiTexCoord1ivARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1SARBPROC)(GLenum target, GLshort s);
GLAPI PFNGLMULTITEXCOORD1SARBPROC glad_glMultiTexCoord1sARB;
#define glMultiTexCoord1sARB glad_glMultiTexCoord1sARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD1SVARBPROC)(GLenum target, const GLshort *v);
GLAPI PFNGLMULTITEXCOORD1SVARBPROC glad_glMultiTexCoord1svARB;
#define glMultiTexCoord1svARB glad_glMultiTexCoord1svARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2DARBPROC)(GLenum target, GLdouble s, GLdouble t);
GLAPI PFNGLMULTITEXCOORD2DARBPROC glad_glMultiTexCoord2dARB;
#define glMultiTexCoord2dARB glad_glMultiTexCoord2dARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2DVARBPROC)(GLenum target, const GLdouble *v);
GLAPI PFNGLMULTITEXCOORD2DVARBPROC glad_glMultiTexCoord2dvARB;
#define glMultiTexCoord2dvARB glad_glMultiTexCoord2dvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2FARBPROC)(GLenum target, GLfloat s, GLfloat t);
GLAPI PFNGLMULTITEXCOORD2FARBPROC glad_glMultiTexCoord2fARB;
#define glMultiTexCoord2fARB glad_glMultiTexCoord2fARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2FVARBPROC)(GLenum target, const GLfloat *v);
GLAPI PFNGLMULTITEXCOORD2FVARBPROC glad_glMultiTexCoord2fvARB;
#define glMultiTexCoord2fvARB glad_glMultiTexCoord2fvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2IARBPROC)(GLenum target, GLint s, GLint t);
GLAPI PFNGLMULTITEXCOORD2IARBPROC glad_glMultiTexCoord2iARB;
#define glMultiTexCoord2iARB glad_glMultiTexCoord2iARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2IVARBPROC)(GLenum target, const GLint *v);
GLAPI PFNGLMULTITEXCOORD2IVARBPROC glad_glMultiTexCoord2ivARB;
#define glMultiTexCoord2ivARB glad_glMultiTexCoord2ivARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2SARBPROC)(GLenum target, GLshort s, GLshort t);
GLAPI PFNGLMULTITEXCOORD2SARBPROC glad_glMultiTexCoord2sARB;
#define glMultiTexCoord2sARB glad_glMultiTexCoord2sARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD2SVARBPROC)(GLenum target, const GLshort *v);
GLAPI PFNGLMULTITEXCOORD2SVARBPROC glad_glMultiTexCoord2svARB;
#define glMultiTexCoord2svARB glad_glMultiTexCoord2svARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3DARBPROC)(GLenum target, GLdouble s, GLdouble t, GLdouble r);
GLAPI PFNGLMULTITEXCOORD3DARBPROC glad_glMultiTexCoord3dARB;
#define glMultiTexCoord3dARB glad_glMultiTexCoord3dARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3DVARBPROC)(GLenum target, const GLdouble *v);
GLAPI PFNGLMULTITEXCOORD3DVARBPROC glad_glMultiTexCoord3dvARB;
#define glMultiTexCoord3dvARB glad_glMultiTexCoord3dvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3FARBPROC)(GLenum target, GLfloat s, GLfloat t, GLfloat r);
GLAPI PFNGLMULTITEXCOORD3FARBPROC glad_glMultiTexCoord3fARB;
#define glMultiTexCoord3fARB glad_glMultiTexCoord3fARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3FVARBPROC)(GLenum target, const GLfloat *v);
GLAPI PFNGLMULTITEXCOORD3FVARBPROC glad_glMultiTexCoord3fvARB;
#define glMultiTexCoord3fvARB glad_glMultiTexCoord3fvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3IARBPROC)(GLenum target, GLint s, GLint t, GLint r);
GLAPI PFNGLMULTITEXCOORD3IARBPROC glad_glMultiTexCoord3iARB;
#define glMultiTexCoord3iARB glad_glMultiTexCoord3iARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3IVARBPROC)(GLenum target, const GLint *v);
GLAPI PFNGLMULTITEXCOORD3IVARBPROC glad_glMultiTexCoord3ivARB;
#define glMultiTexCoord3ivARB glad_glMultiTexCoord3ivARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3SARBPROC)(GLenum target, GLshort s, GLshort t, GLshort r);
GLAPI PFNGLMULTITEXCOORD3SARBPROC glad_glMultiTexCoord3sARB;
#define glMultiTexCoord3sARB glad_glMultiTexCoord3sARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD3SVARBPROC)(GLenum target, const GLshort *v);
GLAPI PFNGLMULTITEXCOORD3SVARBPROC glad_glMultiTexCoord3svARB;
#define glMultiTexCoord3svARB glad_glMultiTexCoord3svARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4DARBPROC)(GLenum target, GLdouble s, GLdouble t, GLdouble r, GLdouble q);
GLAPI PFNGLMULTITEXCOORD4DARBPROC glad_glMultiTexCoord4dARB;
```

```
#define glMultiTexCoord4dARB glad_glMultiTexCoord4dARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4DVARBPROC)(GLenum target, const GLdouble *v);
GLAPI PFNGLMULTITEXCOORD4DVARBPROC glad_glMultiTexCoord4dvARB;
#define glMultiTexCoord4dvARB glad_glMultiTexCoord4dvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4FARBPROC)(GLenum target, GLfloat s, GLfloat t, GLfloat r, GLfloat q);
GLAPI PFNGLMULTITEXCOORD4FARBPROC glad_glMultiTexCoord4fARB;
#define glMultiTexCoord4fARB glad_glMultiTexCoord4fARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4FVARBPROC)(GLenum target, const GLfloat *v);
GLAPI PFNGLMULTITEXCOORD4FVARBPROC glad_glMultiTexCoord4fvARB;
#define glMultiTexCoord4fvARB glad_glMultiTexCoord4fvARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4IARBPROC)(GLenum target, GLint s, GLint t, GLint r, GLint q);
GLAPI PFNGLMULTITEXCOORD4IARBPROC glad_glMultiTexCoord4iARB;
#define glMultiTexCoord4iARB glad_glMultiTexCoord4iARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4IVARBPROC)(GLenum target, const GLint *v);
GLAPI PFNGLMULTITEXCOORD4IVARBPROC glad_glMultiTexCoord4ivARB;
#define glMultiTexCoord4ivARB glad_glMultiTexCoord4ivARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4SARBPROC)(GLenum target, GLshort s, GLshort t, GLshort r, GLshort q);
GLAPI PFNGLMULTITEXCOORD4SARBPROC glad_glMultiTexCoord4sARB;
#define glMultiTexCoord4sARB glad_glMultiTexCoord4sARB
typedef void (APIENTRYP PFNGLMULTITEXCOORD4SVARBPROC)(GLenum target, const GLshort *v);
GLAPI PFNGLMULTITEXCOORD4SVARBPROC glad_glMultiTexCoord4svARB;
#define glMultiTexCoord4svARB glad_glMultiTexCoord4svARB
#endif
#ifndef GL_ARB_occlusion_query
#define GL_ARB_occlusion_query 1
GLAPI int GLAD_GL_ARB_occlusion_query;
typedef void (APIENTRYP PFNGLGENQUERIESARBPROC)(GLsizei n, GLuint *ids);
GLAPI PFNGLGENQUERIESARBPROC glad_glGenQueriesARB;
#define glGenQueriesARB glad_glGenQueriesARB
typedef void (APIENTRYP PFNGLDELETEQUERIESARBPROC)(GLsizei n, const GLuint *ids);
GLAPI PFNGLDELETEQUERIESARBPROC glad_glDeleteQueriesARB;
#define glDeleteQueriesARB glad_glDeleteQueriesARB
typedef GLboolean (APIENTRYP PFNGLISQUERYARBPROC)(GLuint id);
GLAPI PFNGLISQUERYARBPROC glad_glIsQueryARB;
#define glIsQueryARB glad_glIsQueryARB
typedef void (APIENTRYP PFNGLBEGINQUERYARBPROC)(GLenum target, GLuint id);
GLAPI PFNGLBEGINQUERYARBPROC glad_glBeginQueryARB;
#define glBeginQueryARB glad_glBeginQueryARB
typedef void (APIENTRYP PFNGLENDQUERYARBPROC)(GLenum target);
GLAPI PFNGLENDQUERYARBPROC glad_glEndQueryARB;
#define glEndQueryARB glad_glEndQueryARB
typedef void (APIENTRYP PFNGLGETQUERYIVARBPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETQUERYIVARBPROC glad_glGetQueryivARB;
#define glGetQueryivARB glad_glGetQueryivARB
typedef void (APIENTRYP PFNGLGETQUERYOBJECTIVARBPROC)(GLuint id, GLenum pname, GLint *params);
GLAPI PFNGLGETQUERYOBJECTIVARBPROC glad_glGetQueryObjectivARB;
#define glGetQueryObjectivARB glad_glGetQueryObjectivARB
typedef void (APIENTRYP PFNGLGETQUERYOBJECTUIVARBPROC)(GLuint id, GLenum pname, GLuint *params);
GLAPI PFNGLGETQUERYOBJECTUIVARBPROC glad_glGetQueryObjectuivARB;
#define glGetQueryObjectuivARB glad_glGetQueryObjectuivARB
#endif
#ifndef GL_ARB_occlusion_query2
#define GL_ARB_occlusion_query2 1
GLAPI int GLAD_GL_ARB_occlusion_query2;
#endif
#ifndef GL_ARB_parallel_shader_compile
#define GL_ARB_parallel_shader_compile 1
GLAPI int GLAD_GL_ARB_parallel_shader_compile;
typedef void (APIENTRYP PFNGLMAXSHADERCOMPILERTHREADSARBPROC)(GLuint count);
GLAPI PFNGLMAXSHADERCOMPILERTHREADSARBPROC glad_glMaxShaderCompilerThreadsARB;
#define glMaxShaderCompilerThreadsARB glad_glMaxShaderCompilerThreadsARB
#endif
#ifndef GL_ARB_pipeline_statistics_query
#define GL_ARB_pipeline_statistics_query 1
GLAPI int GLAD_GL_ARB_pipeline_statistics_query;
#endif
#ifndef GL_ARB_pixel_buffer_object
#define GL_ARB_pixel_buffer_object 1
GLAPI int GLAD_GL_ARB_pixel_buffer_object;
#endif
#ifndef GL_ARB_point_parameters
#define GL_ARB_point_parameters 1
GLAPI int GLAD_GL_ARB_point_parameters;
typedef void (APIENTRYP PFNGLPOINTPARAMETERFARBPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLPOINTPARAMETERFARBPROC glad_glPointParameterfARB;
#define glPointParameterfARB glad_glPointParameterfARB
typedef void (APIENTRYP PFNGLPOINTPARAMETERFVARBPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLPOINTPARAMETERFVARBPROC glad_glPointParameterfvARB;
#define glPointParameterfvARB glad_glPointParameterfvARB
#endif
#ifndef GL_ARB_point_sprite
#define GL_ARB_point_sprite 1
GLAPI int GLAD_GL_ARB_point_sprite;
#endif
```

```
#ifndef GL_ARB_polygon_offset_clamp
#define GL_ARB_polygon_offset_clamp 1
GLAPI int GLAD_GL_ARB_polygon_offset_clamp;
typedef void (APIENTRYP PFNGLPOLYGONOFFSETCLAMPPROC)(GLfloat factor, GLfloat units, GLfloat clamp);
GLAPI PFNGLPOLYGONOFFSETCLAMPPROC glad_glPolygonOffsetClamp;
#define glPolygonOffsetClamp glad_glPolygonOffsetClamp
#endif
#ifndef GL_ARB_post_depth_coverage
#define GL_ARB_post_depth_coverage 1
GLAPI int GLAD_GL_ARB_post_depth_coverage;
#endif
#ifndef GL_ARB_program_interface_query
#define GL_ARB_program_interface_query 1
GLAPI int GLAD_GL_ARB_program_interface_query;
typedef void (APIENTRYP PFNGLGETPROGRAMINTERFACEIVPROC)(GLuint program, GLenum programInterface, GLenum pname, GLint *params);
GLAPI PFNGLGETPROGRAMINTERFACEIVPROC glad_glGetProgramInterfaceiv;
#define glGetProgramInterfaceiv glad_glGetProgramInterfaceiv
typedef GLuint (APIENTRYP PFNGLGETPROGRAMRESOURCEINDEXPROC)(GLuint program, GLenum programInterface, const GLchar *name);
GLAPI PFNGLGETPROGRAMRESOURCEINDEXPROC glad_glGetProgramResourceIndex;
#define glGetProgramResourceIndex glad_glGetProgramResourceIndex
typedef void (APIENTRYP PFNGLGETPROGRAMRESOURCENAMEPROC)(GLuint program, GLenum programInterface, GLuint index, GLsizei bufSize, GLsizei *length, GLchar *name);
GLAPI PFNGLGETPROGRAMRESOURCENAMEPROC glad_glGetProgramResourceName;
#define glGetProgramResourceName glad_glGetProgramResourceName
typedef void (APIENTRYP PFNGLGETPROGRAMRESOURCEIVPROC)(GLuint program, GLenum programInterface, GLuint index, GLsizei propCount, const GLenum *props, GLsizei count, GLsizei *length, GLint *params);
GLAPI PFNGLGETPROGRAMRESOURCEIVPROC glad_glGetProgramResourceiv;
#define glGetProgramResourceiv glad_glGetProgramResourceiv
typedef GLint (APIENTRYP PFNGLGETPROGRAMRESOURCELOCATIONPROC)(GLuint program, GLenum programInterface, const GLchar *name);
GLAPI PFNGLGETPROGRAMRESOURCELOCATIONPROC glad_glGetProgramResourceLocation;
#define glGetProgramResourceLocation glad_glGetProgramResourceLocation
typedef GLint (APIENTRYP PFNGLGETPROGRAMRESOURCELOCATIONINDEXPROC)(GLuint program, GLenum programInterface, const GLchar *name);
GLAPI PFNGLGETPROGRAMRESOURCELOCATIONINDEXPROC glad_glGetProgramResourceLocationIndex;
#define glGetProgramResourceLocationIndex glad_glGetProgramResourceLocationIndex
#endif
#ifndef GL_ARB_provoking_vertex
#define GL_ARB_provoking_vertex 1
GLAPI int GLAD_GL_ARB_provoking_vertex;
#endif
#ifndef GL_ARB_query_buffer_object
#define GL_ARB_query_buffer_object 1
GLAPI int GLAD_GL_ARB_query_buffer_object;
#endif
#ifndef GL_ARB_robust_buffer_access_behavior
#define GL_ARB_robust_buffer_access_behavior 1
GLAPI int GLAD_GL_ARB_robust_buffer_access_behavior;
#endif
#ifndef GL_ARB_robustness
#define GL_ARB_robustness 1
GLAPI int GLAD_GL_ARB_robustness;
typedef GLenum (APIENTRYP PFNGLGETGRAPHICSRESETSTATUSARBPROC)(void);
GLAPI PFNGLGETGRAPHICSRESETSTATUSARBPROC glad_glGetGraphicsResetStatusARB;
#define glGetGraphicsResetStatusARB glad_glGetGraphicsResetStatusARB
typedef void (APIENTRYP PFNGLGETNTEXIMAGEARBPROC)(GLenum target, GLint level, GLenum format, GLenum type, GLsizei bufSize, void *img);
GLAPI PFNGLGETNTEXIMAGEARBPROC glad_glGetnTexImageARB;
#define glGetnTexImageARB glad_glGetnTexImageARB
typedef void (APIENTRYP PFNGLREADNPIXELSARBPROC)(GLint x, GLint y, GLsizei width, GLsizei height, GLenum format, GLenum type, GLsizei bufSize, void *data);
GLAPI PFNGLREADNPIXELSARBPROC glad_glReadnPixelsARB;
#define glReadnPixelsARB glad_glReadnPixelsARB
typedef void (APIENTRYP PFNGLGETNCOMPRESSEDTEXIMAGEARBPROC)(GLenum target, GLint lod, GLsizei bufSize, void *img);
GLAPI PFNGLGETNCOMPRESSEDTEXIMAGEARBPROC glad_glGetnCompressedTexImageARB;
#define glGetnCompressedTexImageARB glad_glGetnCompressedTexImageARB
typedef void (APIENTRYP PFNGLGETNUNIFORMFVARBPROC)(GLuint program, GLint location, GLsizei bufSize, GLfloat *params);
GLAPI PFNGLGETNUNIFORMFVARBPROC glad_glGetnUniformfvARB;
#define glGetnUniformfvARB glad_glGetnUniformfvARB
typedef void (APIENTRYP PFNGLGETNUNIFORMIVARBPROC)(GLuint program, GLint location, GLsizei bufSize, GLint *params);
GLAPI PFNGLGETNUNIFORMIVARBPROC glad_glGetnUniformivARB;
#define glGetnUniformivARB glad_glGetnUniformivARB
typedef void (APIENTRYP PFNGLGETNUNIFORMUIVARBPROC)(GLuint program, GLint location, GLsizei bufSize, GLuint *params);
GLAPI PFNGLGETNUNIFORMUIVARBPROC glad_glGetnUniformuivARB;
#define glGetnUniformuivARB glad_glGetnUniformuivARB
typedef void (APIENTRYP PFNGLGETNUNIFORMDVARBPROC)(GLuint program, GLint location, GLsizei bufSize, GLdouble *params);
GLAPI PFNGLGETNUNIFORMDVARBPROC glad_glGetnUniformdvARB;
#define glGetnUniformdvARB glad_glGetnUniformdvARB
typedef void (APIENTRYP PFNGLGETNMAPDVARBPROC)(GLenum target, GLenum query, GLsizei bufSize, GLdouble *v);
```

```
GLAPI PFNGLGETNMAPDVARBPROC glad_glGetnMapdvARB;
#define glGetnMapdvARB glad_glGetnMapdvARB
typedef void (APIENTRYP PFNGLGETNMAPFVARBPROC)(GLenum target, GLenum query, GLsizei bufSize, GLfloat *v);
GLAPI PFNGLGETNMAPFVARBPROC glad_glGetnMapfvARB;
#define glGetnMapfvARB glad_glGetnMapfvARB
typedef void (APIENTRYP PFNGLGETNMAPIVARBPROC)(GLenum target, GLenum query, GLsizei bufSize, GLint *v);
GLAPI PFNGLGETNMAPIVARBPROC glad_glGetnMapivARB;
#define glGetnMapivARB glad_glGetnMapivARB
typedef void (APIENTRYP PFNGLGETNPIXELMAPFVARBPROC)(GLenum map, GLsizei bufSize, GLfloat *values);
GLAPI PFNGLGETNPIXELMAPFVARBPROC glad_glGetnPixelMapfvARB;
#define glGetnPixelMapfvARB glad_glGetnPixelMapfvARB
typedef void (APIENTRYP PFNGLGETNPIXELMAPUIVARBPROC)(GLenum map, GLsizei bufSize, GLuint *values);
GLAPI PFNGLGETNPIXELMAPUIVARBPROC glad_glGetnPixelMapuivARB;
#define glGetnPixelMapuivARB glad_glGetnPixelMapuivARB
typedef void (APIENTRYP PFNGLGETNPIXELMAPUSVARBPROC)(GLenum map, GLsizei bufSize, GLushort *values);
GLAPI PFNGLGETNPIXELMAPUSVARBPROC glad_glGetnPixelMapusvARB;
#define glGetnPixelMapusvARB glad_glGetnPixelMapusvARB
typedef void (APIENTRYP PFNGLGETNPOLYGONSTIPPLEARBPROC)(GLsizei bufSize, GLubyte *pattern);
GLAPI PFNGLGETNPOLYGONSTIPPLEARBPROC glad_glGetnPolygonStippleARB;
#define glGetnPolygonStippleARB glad_glGetnPolygonStippleARB
typedef void (APIENTRYP PFNGLGETNCOLORTABLEARBPROC)(GLenum target, GLenum format, GLenum type, GLsizei bufSize, void *table);
GLAPI PFNGLGETNCOLORTABLEARBPROC glad_glGetnColorTableARB;
#define glGetnColorTableARB glad_glGetnColorTableARB
typedef void (APIENTRYP PFNGLGETNCONVOLUTIONFILTERARBPROC)(GLenum target, GLenum format, GLenum type, GLsizei bufSize, void *image);
GLAPI PFNGLGETNCONVOLUTIONFILTERARBPROC glad_glGetnConvolutionFilterARB;
#define glGetnConvolutionFilterARB glad_glGetnConvolutionFilterARB
typedef void (APIENTRYP PFNGLGETNSEPARABLEFILTERARBPROC)(GLenum target, GLenum format, GLenum type, GLsizei rowBufSize, void *row, GLsizei columnBufSize, void *column, void *span);
GLAPI PFNGLGETNSEPARABLEFILTERARBPROC glad_glGetnSeparableFilterARB;
#define glGetnSeparableFilterARB glad_glGetnSeparableFilterARB
typedef void (APIENTRYP PFNGLGETNHISTOGRAMARBPROC)(GLenum target, GLboolean reset, GLenum format, GLenum type, GLsizei bufSize, void *values);
GLAPI PFNGLGETNHISTOGRAMARBPROC glad_glGetnHistogramARB;
#define glGetnHistogramARB glad_glGetnHistogramARB
typedef void (APIENTRYP PFNGLGETNMINMAXARBPROC)(GLenum target, GLboolean reset, GLenum format, GLenum type, GLsizei bufSize, void *values);
GLAPI PFNGLGETNMINMAXARBPROC glad_glGetnMinmaxARB;
#define glGetnMinmaxARB glad_glGetnMinmaxARB
#endif
#ifndef GL_ARB_robustness_isolation
#define GL_ARB_robustness_isolation 1
GLAPI int GLAD_GL_ARB_robustness_isolation;
#endif
#ifndef GL_ARB_sample_locations
#define GL_ARB_sample_locations 1
GLAPI int GLAD_GL_ARB_sample_locations;
typedef void (APIENTRYP PFNGLFRAMEBUFFERSAMPLELOCATIONSFVARBPROC)(GLenum target, GLuint start, GLsizei count, const GLfloat *v);
GLAPI PFNGLFRAMEBUFFERSAMPLELOCATIONSFVARBPROC glad_glFramebufferSampleLocationsfvARB;
#define glFramebufferSampleLocationsfvARB glad_glFramebufferSampleLocationsfvARB
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVARBPROC)(GLuint framebuffer, GLuint start, GLsizei count, const GLfloat *v);
GLAPI PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVARBPROC glad_glNamedFramebufferSampleLocationsfvARB;
#define glNamedFramebufferSampleLocationsfvARB glad_glNamedFramebufferSampleLocationsfvARB
typedef void (APIENTRYP PFNGLEVALUATEDEPTHVALUESARBPROC)(void);
GLAPI PFNGLEVALUATEDEPTHVALUESARBPROC glad_glEvaluateDepthValuesARB;
#define glEvaluateDepthValuesARB glad_glEvaluateDepthValuesARB
#endif
#ifndef GL_ARB_sample_shading
#define GL_ARB_sample_shading 1
GLAPI int GLAD_GL_ARB_sample_shading;
typedef void (APIENTRYP PFNGLMINSAMPLESHADINGARBPROC)(GLfloat value);
GLAPI PFNGLMINSAMPLESHADINGARBPROC glad_glMinSampleShadingARB;
#define glMinSampleShadingARB glad_glMinSampleShadingARB
#endif
#ifndef GL_ARB_sampler_objects
#define GL_ARB_sampler_objects 1
GLAPI int GLAD_GL_ARB_sampler_objects;
#endif
#ifndef GL_ARB_seamless_cube_map
#define GL_ARB_seamless_cube_map 1
GLAPI int GLAD_GL_ARB_seamless_cube_map;
#endif
#ifndef GL_ARB_seamless_cubemap_per_texture
#define GL_ARB_seamless_cubemap_per_texture 1
GLAPI int GLAD_GL_ARB_seamless_cubemap_per_texture;
#endif
#ifndef GL_ARB_separate_shader_objects
#define GL_ARB_separate_shader_objects 1
GLAPI int GLAD_GL_ARB_separate_shader_objects;
#endif
#ifndef GL_ARB_shader_atomic_counter_ops
#define GL_ARB_shader_atomic_counter_ops 1
```

```
GLAPI int GLAD_GL_ARB_shader_atomic_counter_ops;
#endif
#ifndef GL_ARB_shader_atomic_counters
#define GL_ARB_shader_atomic_counters 1
GLAPI int GLAD_GL_ARB_shader_atomic_counters;
typedef void (APIENTRYP PFNGLGETACTIVEATOMICCOUNTERBUFFERIVPROC)(GLuint program, GLuint bufferIndex, GLenum pname, GLint *params);
GLAPI PFNGLGETACTIVEATOMICCOUNTERBUFFERIVPROC glad_glGetActiveAtomicCounterBufferiv;
#define glGetActiveAtomicCounterBufferiv glad_glGetActiveAtomicCounterBufferiv
#endif
#ifndef GL_ARB_shader_ballot
#define GL_ARB_shader_ballot 1
GLAPI int GLAD_GL_ARB_shader_ballot;
#endif
#ifndef GL_ARB_shader_bit_encoding
#define GL_ARB_shader_bit_encoding 1
GLAPI int GLAD_GL_ARB_shader_bit_encoding;
#endif
#ifndef GL_ARB_shader_clock
#define GL_ARB_shader_clock 1
GLAPI int GLAD_GL_ARB_shader_clock;
#endif
#ifndef GL_ARB_shader_draw_parameters
#define GL_ARB_shader_draw_parameters 1
GLAPI int GLAD_GL_ARB_shader_draw_parameters;
#endif
#ifndef GL_ARB_shader_group_vote
#define GL_ARB_shader_group_vote 1
GLAPI int GLAD_GL_ARB_shader_group_vote;
#endif
#ifndef GL_ARB_shader_image_load_store
#define GL_ARB_shader_image_load_store 1
GLAPI int GLAD_GL_ARB_shader_image_load_store;
typedef void (APIENTRYP PFNGLBINDIMAGETEXTUREPROC)(GLuint unit, GLuint texture, GLint level, GLboolean layered, GLint layer, GLenum access, GLenum format);
GLAPI PFNGLBINDIMAGETEXTUREPROC glad_glBindImageTexture;
#define glBindImageTexture glad_glBindImageTexture
typedef void (APIENTRYP PFNGLMEMORYBARRIERPROC)(GLbitfield barriers);
GLAPI PFNGLMEMORYBARRIERPROC glad_glMemoryBarrier;
#define glMemoryBarrier glad_glMemoryBarrier
#endif
#ifndef GL_ARB_shader_image_size
#define GL_ARB_shader_image_size 1
GLAPI int GLAD_GL_ARB_shader_image_size;
#endif
#ifndef GL_ARB_shader_objects
#define GL_ARB_shader_objects 1
GLAPI int GLAD_GL_ARB_shader_objects;
typedef void (APIENTRYP PFNGLDELETEOBJECTARBPROC)(GLhandleARB obj);
GLAPI PFNGLDELETEOBJECTARBPROC glad_glDeleteObjectARB;
#define glDeleteObjectARB glad_glDeleteObjectARB
typedef GLhandleARB (APIENTRYP PFNGLGETHANDLEARBPROC)(GLenum pname);
GLAPI PFNGLGETHANDLEARBPROC glad_glGetHandleARB;
#define glGetHandleARB glad_glGetHandleARB
typedef void (APIENTRYP PFNGLDETACHOBJECTARBPROC)(GLhandleARB containerObj, GLhandleARB attachedObj);
GLAPI PFNGLDETACHOBJECTARBPROC glad_glDetachObjectARB;
#define glDetachObjectARB glad_glDetachObjectARB
typedef GLhandleARB (APIENTRYP PFNGLCREATESHADEROBJECTARBPROC)(GLenum shaderType);
GLAPI PFNGLCREATESHADEROBJECTARBPROC glad_glCreateShaderObjectARB;
#define glCreateShaderObjectARB glad_glCreateShaderObjectARB
typedef void (APIENTRYP PFNGLSHADERSOURCEARBPROC)(GLhandleARB shaderObj, GLsizei count, const GLcharARB **string, const GLint *length);
GLAPI PFNGLSHADERSOURCEARBPROC glad_glShaderSourceARB;
#define glShaderSourceARB glad_glShaderSourceARB
typedef void (APIENTRYP PFNGLCOMPILESHADERARBPROC)(GLhandleARB shaderObj);
GLAPI PFNGLCOMPILESHADERARBPROC glad_glCompileShaderARB;
#define glCompileShaderARB glad_glCompileShaderARB
typedef GLhandleARB (APIENTRYP PFNGLCREATEPROGRAMOBJECTARBPROC)(void);
GLAPI PFNGLCREATEPROGRAMOBJECTARBPROC glad_glCreateProgramObjectARB;
#define glCreateProgramObjectARB glad_glCreateProgramObjectARB
typedef void (APIENTRYP PFNGLATTACHOBJECTARBPROC)(GLhandleARB containerObj, GLhandleARB obj);
GLAPI PFNGLATTACHOBJECTARBPROC glad_glAttachObjectARB;
#define glAttachObjectARB glad_glAttachObjectARB
typedef void (APIENTRYP PFNGLLINKPROGRAMARBPROC)(GLhandleARB programObj);
GLAPI PFNGLLINKPROGRAMARBPROC glad_glLinkProgramARB;
#define glLinkProgramARB glad_glLinkProgramARB
typedef void (APIENTRYP PFNGLUSEPROGRAMOBJECTARBPROC)(GLhandleARB programObj);
GLAPI PFNGLUSEPROGRAMOBJECTARBPROC glad_glUseProgramObjectARB;
#define glUseProgramObjectARB glad_glUseProgramObjectARB
typedef void (APIENTRYP PFNGLVALIDATEPROGRAMARBPROC)(GLhandleARB programObj);
GLAPI PFNGLVALIDATEPROGRAMARBPROC glad_glValidateProgramARB;
#define glValidateProgramARB glad_glValidateProgramARB
typedef void (APIENTRYP PFNGLUNIFORM1FARBPROC)(GLint location, GLfloat v0);
GLAPI PFNGLUNIFORM1FARBPROC glad_glUniform1fARB;
#define glUniform1fARB glad_glUniform1fARB
typedef void (APIENTRYP PFNGLUNIFORM2FARBPROC)(GLint location, GLfloat v0, GLfloat v1);
GLAPI PFNGLUNIFORM2FARBPROC glad_glUniform2fARB;
```

```
#define glUniform2fARB glad_glUniform2fARB
typedef void (APIENTRYP PFNGLUNIFORM3FARBPROC)(GLint location, GLfloat v0, GLfloat v1, GLfloat v2);
GLAPI PFNGLUNIFORM3FARBPROC glad_glUniform3fARB;
#define glUniform3fARB glad_glUniform3fARB
typedef void (APIENTRYP PFNGLUNIFORM4FARBPROC)(GLint location, GLfloat v0, GLfloat v1, GLfloat v2, GLfloat v3);
GLAPI PFNGLUNIFORM4FARBPROC glad_glUniform4fARB;
#define glUniform4fARB glad_glUniform4fARB
typedef void (APIENTRYP PFNGLUNIFORM1IARBPROC)(GLint location, GLint v0);
GLAPI PFNGLUNIFORM1IARBPROC glad_glUniform1iARB;
#define glUniform1iARB glad_glUniform1iARB
typedef void (APIENTRYP PFNGLUNIFORM2IARBPROC)(GLint location, GLint v0, GLint v1);
GLAPI PFNGLUNIFORM2IARBPROC glad_glUniform2iARB;
#define glUniform2iARB glad_glUniform2iARB
typedef void (APIENTRYP PFNGLUNIFORM3IARBPROC)(GLint location, GLint v0, GLint v1, GLint v2);
GLAPI PFNGLUNIFORM3IARBPROC glad_glUniform3iARB;
#define glUniform3iARB glad_glUniform3iARB
typedef void (APIENTRYP PFNGLUNIFORM4IARBPROC)(GLint location, GLint v0, GLint v1, GLint v2, GLint v3);
GLAPI PFNGLUNIFORM4IARBPROC glad_glUniform4iARB;
#define glUniform4iARB glad_glUniform4iARB
typedef void (APIENTRYP PFNGLUNIFORM1FVARBPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM1FVARBPROC glad_glUniform1fvARB;
#define glUniform1fvARB glad_glUniform1fvARB
typedef void (APIENTRYP PFNGLUNIFORM2FVARBPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM2FVARBPROC glad_glUniform2fvARB;
#define glUniform2fvARB glad_glUniform2fvARB
typedef void (APIENTRYP PFNGLUNIFORM3FVARBPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM3FVARBPROC glad_glUniform3fvARB;
#define glUniform3fvARB glad_glUniform3fvARB
typedef void (APIENTRYP PFNGLUNIFORM4FVARBPROC)(GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLUNIFORM4FVARBPROC glad_glUniform4fvARB;
#define glUniform4fvARB glad_glUniform4fvARB
typedef void (APIENTRYP PFNGLUNIFORM1IVARBPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM1IVARBPROC glad_glUniform1ivARB;
#define glUniform1ivARB glad_glUniform1ivARB
typedef void (APIENTRYP PFNGLUNIFORM2IVARBPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM2IVARBPROC glad_glUniform2ivARB;
#define glUniform2ivARB glad_glUniform2ivARB
typedef void (APIENTRYP PFNGLUNIFORM3IVARBPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM3IVARBPROC glad_glUniform3ivARB;
#define glUniform3ivARB glad_glUniform3ivARB
typedef void (APIENTRYP PFNGLUNIFORM4IVARBPROC)(GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLUNIFORM4IVARBPROC glad_glUniform4ivARB;
#define glUniform4ivARB glad_glUniform4ivARB
typedef void (APIENTRYP PFNGLUNIFORMMATRIX2FVARBPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX2FVARBPROC glad_glUniformMatrix2fvARB;
#define glUniformMatrix2fvARB glad_glUniformMatrix2fvARB
typedef void (APIENTRYP PFNGLUNIFORMMATRIX3FVARBPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX3FVARBPROC glad_glUniformMatrix3fvARB;
#define glUniformMatrix3fvARB glad_glUniformMatrix3fvARB
typedef void (APIENTRYP PFNGLUNIFORMMATRIX4FVARBPROC)(GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLUNIFORMMATRIX4FVARBPROC glad_glUniformMatrix4fvARB;
#define glUniformMatrix4fvARB glad_glUniformMatrix4fvARB
typedef void (APIENTRYP PFNGLGETOBJECTPARAMETERFVARBPROC)(GLhandleARB obj, GLenum pname, GLfloat *params);
GLAPI PFNGLGETOBJECTPARAMETERFVARBPROC glad_glGetObjectParameterfvARB;
#define glGetObjectParameterfvARB glad_glGetObjectParameterfvARB
typedef void (APIENTRYP PFNGLGETOBJECTPARAMETERIVARBPROC)(GLhandleARB obj, GLenum pname, GLint *params);
GLAPI PFNGLGETOBJECTPARAMETERIVARBPROC glad_glGetObjectParameterivARB;
#define glGetObjectParameterivARB glad_glGetObjectParameterivARB
typedef void (APIENTRYP PFNGLGETINFOLOGARBPROC)(GLhandleARB obj, GLsizei maxLength, GLsizei *length, GLcharARB *infoLog);
GLAPI PFNGLGETINFOLOGARBPROC glad_glGetInfoLogARB;
#define glGetInfoLogARB glad_glGetInfoLogARB
typedef void (APIENTRYP PFNGLGETATTACHEDOBJECTSARBPROC)(GLhandleARB containerObj, GLsizei maxCount, GLsizei *count, GLhandleARB *obj);
GLAPI PFNGLGETATTACHEDOBJECTSARBPROC glad_glGetAttachedObjectsARB;
#define glGetAttachedObjectsARB glad_glGetAttachedObjectsARB
typedef GLint (APIENTRYP PFNGLGETUNIFORMLOCATIONARBPROC)(GLhandleARB programObj, const GLcharARB *name);
GLAPI PFNGLGETUNIFORMLOCATIONARBPROC glad_glGetUniformLocationARB;
#define glGetUniformLocationARB glad_glGetUniformLocationARB
typedef void (APIENTRYP PFNGLGETACTIVEUNIFORMARBPROC)(GLhandleARB programObj, GLuint index, GLsizei maxLength, GLsizei *length, GLint *size, GLenum *type, GLcharARB *name);
GLAPI PFNGLGETACTIVEUNIFORMARBPROC glad_glGetActiveUniformARB;
#define glGetActiveUniformARB glad_glGetActiveUniformARB
typedef void (APIENTRYP PFNGLGETUNIFORMFVARBPROC)(GLhandleARB programObj, GLint location, GLfloat *params);
GLAPI PFNGLGETUNIFORMFVARBPROC glad_glGetUniformfvARB;
#define glGetUniformfvARB glad_glGetUniformfvARB
typedef void (APIENTRYP PFNGLGETUNIFORMIVARBPROC)(GLhandleARB programObj, GLint location, GLint *params);
```

```
GLAPI PFNGLGETUNIFORMIVARBPROC glad_glGetUniformivARB;
#define glGetUniformivARB glad_glGetUniformivARB
typedef void (APIENTRYP PFNGLGETSHADERSOURCEARBPROC)(GLhandleARB obj, GLsizei maxLength, GLsizei *length, GLcharARB *source);
GLAPI PFNGLGETSHADERSOURCEARBPROC glad_glGetShaderSourceARB;
#define glGetShaderSourceARB glad_glGetShaderSourceARB
#endif
#ifndef GL_ARB_shader_precision
#define GL_ARB_shader_precision 1
GLAPI int GLAD_GL_ARB_shader_precision;
#endif
#ifndef GL_ARB_shader_stencil_export
#define GL_ARB_shader_stencil_export 1
GLAPI int GLAD_GL_ARB_shader_stencil_export;
#endif
#ifndef GL_ARB_shader_storage_buffer_object
#define GL_ARB_shader_storage_buffer_object 1
GLAPI int GLAD_GL_ARB_shader_storage_buffer_object;
typedef void (APIENTRYP PFNGLSHADERSTORAGEBLOCKBINDINGPROC)(GLuint program, GLuint storageBlockIndex, GLuint storageBlockBinding);
GLAPI PFNGLSHADERSTORAGEBLOCKBINDINGPROC glad_glShaderStorageBlockBinding;
#define glShaderStorageBlockBinding glad_glShaderStorageBlockBinding
#endif
#ifndef GL_ARB_shader_subroutine
#define GL_ARB_shader_subroutine 1
GLAPI int GLAD_GL_ARB_shader_subroutine;
#endif
#ifndef GL_ARB_shader_texture_image_samples
#define GL_ARB_shader_texture_image_samples 1
GLAPI int GLAD_GL_ARB_shader_texture_image_samples;
#endif
#ifndef GL_ARB_shader_texture_lod
#define GL_ARB_shader_texture_lod 1
GLAPI int GLAD_GL_ARB_shader_texture_lod;
#endif
#ifndef GL_ARB_shader_viewport_layer_array
#define GL_ARB_shader_viewport_layer_array 1
GLAPI int GLAD_GL_ARB_shader_viewport_layer_array;
#endif
#ifndef GL_ARB_shading_language_100
#define GL_ARB_shading_language_100 1
GLAPI int GLAD_GL_ARB_shading_language_100;
#endif
#ifndef GL_ARB_shading_language_420pack
#define GL_ARB_shading_language_420pack 1
GLAPI int GLAD_GL_ARB_shading_language_420pack;
#endif
#ifndef GL_ARB_shading_language_include
#define GL_ARB_shading_language_include 1
GLAPI int GLAD_GL_ARB_shading_language_include;
typedef void (APIENTRYP PFNGLNAMEDSTRINGARBPROC)(GLenum type, GLint namelen, const GLchar *name, GLint stringlen, const GLchar *string);
GLAPI PFNGLNAMEDSTRINGARBPROC glad_glNamedStringARB;
#define glNamedStringARB glad_glNamedStringARB
typedef void (APIENTRYP PFNGLDELETENAMEDSTRINGARBPROC)(GLint namelen, const GLchar *name);
GLAPI PFNGLDELETENAMEDSTRINGARBPROC glad_glDeleteNamedStringARB;
#define glDeleteNamedStringARB glad_glDeleteNamedStringARB
typedef void (APIENTRYP PFNGLCOMPILESHADERINCLUDEARBPROC)(GLuint shader, GLsizei count, const GLchar *const*path, const GLint *length);
GLAPI PFNGLCOMPILESHADERINCLUDEARBPROC glad_glCompileShaderIncludeARB;
#define glCompileShaderIncludeARB glad_glCompileShaderIncludeARB
typedef GLboolean (APIENTRYP PFNGLISNAMEDSTRINGARBPROC)(GLint namelen, const GLchar *name);
GLAPI PFNGLISNAMEDSTRINGARBPROC glad_glIsNamedStringARB;
#define glIsNamedStringARB glad_glIsNamedStringARB
typedef void (APIENTRYP PFNGLGETNAMEDSTRINGARBPROC)(GLint namelen, const GLchar *name, GLsizei bufSize, GLint *stringlen, GLchar *string);
GLAPI PFNGLGETNAMEDSTRINGARBPROC glad_glGetNamedStringARB;
#define glGetNamedStringARB glad_glGetNamedStringARB
typedef void (APIENTRYP PFNGLGETNAMEDSTRINGIVARBPROC)(GLint namelen, const GLchar *name, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDSTRINGIVARBPROC glad_glGetNamedStringivARB;
#define glGetNamedStringivARB glad_glGetNamedStringivARB
#endif
#ifndef GL_ARB_shading_language_packing
#define GL_ARB_shading_language_packing 1
GLAPI int GLAD_GL_ARB_shading_language_packing;
#endif
#ifndef GL_ARB_shadow
#define GL_ARB_shadow 1
GLAPI int GLAD_GL_ARB_shadow;
#endif
#ifndef GL_ARB_shadow_ambient
#define GL_ARB_shadow_ambient 1
GLAPI int GLAD_GL_ARB_shadow_ambient;
#endif
#ifndef GL_ARB_sparse_buffer
#define GL_ARB_sparse_buffer 1
```

```
GLAPI int GLAD_GL_ARB_sparse_buffer;
typedef void (APIENTRYP PFNGLBUFFERPAGECOMMITMENTARBPROC)(GLenum target, GLintptr offset, GLsizeiptr size, GLboolean commit);
GLAPI PFNGLBUFFERPAGECOMMITMENTARBPROC glad_glBufferPageCommitmentARB;
#define glBufferPageCommitmentARB glad_glBufferPageCommitmentARB
typedef void (APIENTRYP PFNGLNAMEDBUFFERPAGECOMMITMENTEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, GLboolean commit);
GLAPI PFNGLNAMEDBUFFERPAGECOMMITMENTEXTPROC glad_glNamedBufferPageCommitmentEXT;
#define glNamedBufferPageCommitmentEXT glad_glNamedBufferPageCommitmentEXT
typedef void (APIENTRYP PFNGLNAMEDBUFFERPAGECOMMITMENTARBPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, GLboolean commit);
GLAPI PFNGLNAMEDBUFFERPAGECOMMITMENTARBPROC glad_glNamedBufferPageCommitmentARB;
#define glNamedBufferPageCommitmentARB glad_glNamedBufferPageCommitmentARB
#endif
#ifndef GL_ARB_sparse_texture
#define GL_ARB_sparse_texture 1
GLAPI int GLAD_GL_ARB_sparse_texture;
typedef void (APIENTRYP PFNGLTEXPAGECOMMITMENTARBPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLboolean commit);
GLAPI PFNGLTEXPAGECOMMITMENTARBPROC glad_glTexPageCommitmentARB;
#define glTexPageCommitmentARB glad_glTexPageCommitmentARB
#endif
#ifndef GL_ARB_sparse_texture2
#define GL_ARB_sparse_texture2 1
GLAPI int GLAD_GL_ARB_sparse_texture2;
#endif
#ifndef GL_ARB_sparse_texture_clamp
#define GL_ARB_sparse_texture_clamp 1
GLAPI int GLAD_GL_ARB_sparse_texture_clamp;
#endif
#ifndef GL_ARB_spirv_extensions
#define GL_ARB_spirv_extensions 1
GLAPI int GLAD_GL_ARB_spirv_extensions;
#endif
#ifndef GL_ARB_stencil_texturing
#define GL_ARB_stencil_texturing 1
GLAPI int GLAD_GL_ARB_stencil_texturing;
#endif
#ifndef GL_ARB_sync
#define GL_ARB_sync 1
GLAPI int GLAD_GL_ARB_sync;
#endif
#ifndef GL_ARB_tessellation_shader
#define GL_ARB_tessellation_shader 1
GLAPI int GLAD_GL_ARB_tessellation_shader;
#endif
#ifndef GL_ARB_texture_barrier
#define GL_ARB_texture_barrier 1
GLAPI int GLAD_GL_ARB_texture_barrier;
typedef void (APIENTRYP PFNGLTEXTUREBARRIERPROC)(void);
GLAPI PFNGLTEXTUREBARRIERPROC glad_glTextureBarrier;
#define glTextureBarrier glad_glTextureBarrier
#endif
#ifndef GL_ARB_texture_border_clamp
#define GL_ARB_texture_border_clamp 1
GLAPI int GLAD_GL_ARB_texture_border_clamp;
#endif
#ifndef GL_ARB_texture_buffer_object
#define GL_ARB_texture_buffer_object 1
GLAPI int GLAD_GL_ARB_texture_buffer_object;
typedef void (APIENTRYP PFNGLTEXBUFFERARBPROC)(GLenum target, GLenum internalformat, GLuint buffer);
GLAPI PFNGLTEXBUFFERARBPROC glad_glTexBufferARB;
#define glTexBufferARB glad_glTexBufferARB
#endif
#ifndef GL_ARB_texture_buffer_object_rgb32
#define GL_ARB_texture_buffer_object_rgb32 1
GLAPI int GLAD_GL_ARB_texture_buffer_object_rgb32;
#endif
#ifndef GL_ARB_texture_buffer_range
#define GL_ARB_texture_buffer_range 1
GLAPI int GLAD_GL_ARB_texture_buffer_range;
typedef void (APIENTRYP PFNGLTEXBUFFERRANGEPROC)(GLenum target, GLenum internalformat, GLuint buffer, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLTEXBUFFERRANGEPROC glad_glTexBufferRange;
#define glTexBufferRange glad_glTexBufferRange
#endif
#ifndef GL_ARB_texture_compression
#define GL_ARB_texture_compression 1
GLAPI int GLAD_GL_ARB_texture_compression;
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXIMAGE3DARBPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXIMAGE3DARBPROC glad_glCompressedTexImage3DARB;
#define glCompressedTexImage3DARB glad_glCompressedTexImage3DARB
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXIMAGE2DARBPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXIMAGE2DARBPROC glad_glCompressedTexImage2DARB;
```

```
#define glCompressedTexImage2DARB glad_glCompressedTexImage2DARB
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXIMAGE1DARBPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLint border, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXIMAGE1DARBPROC glad_glCompressedTexImage1DARB;
#define glCompressedTexImage1DARB glad_glCompressedTexImage1DARB
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXSUBIMAGE3DARBPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE3DARBPROC glad_glCompressedTexSubImage3DARB;
#define glCompressedTexSubImage3DARB glad_glCompressedTexSubImage3DARB
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXSUBIMAGE2DARBPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE2DARBPROC glad_glCompressedTexSubImage2DARB;
#define glCompressedTexSubImage2DARB glad_glCompressedTexSubImage2DARB
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXSUBIMAGE1DARBPROC)(GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *data);
GLAPI PFNGLCOMPRESSEDTEXSUBIMAGE1DARBPROC glad_glCompressedTexSubImage1DARB;
#define glCompressedTexSubImage1DARB glad_glCompressedTexSubImage1DARB
typedef void (APIENTRYP PFNGLGETCOMPRESSEDTEXIMAGEARBPROC)(GLenum target, GLint level, void *img);
GLAPI PFNGLGETCOMPRESSEDTEXIMAGEARBPROC glad_glGetCompressedTexImageARB;
#define glGetCompressedTexImageARB glad_glGetCompressedTexImageARB
#endif
#ifndef GL_ARB_texture_compression_bptc
#define GL_ARB_texture_compression_bptc 1
GLAPI int GLAD_GL_ARB_texture_compression_bptc;
#endif
#ifndef GL_ARB_texture_compression_rgtc
#define GL_ARB_texture_compression_rgtc 1
GLAPI int GLAD_GL_ARB_texture_compression_rgtc;
#endif
#ifndef GL_ARB_texture_cube_map
#define GL_ARB_texture_cube_map 1
GLAPI int GLAD_GL_ARB_texture_cube_map;
#endif
#ifndef GL_ARB_texture_cube_map_array
#define GL_ARB_texture_cube_map_array 1
GLAPI int GLAD_GL_ARB_texture_cube_map_array;
#endif
#ifndef GL_ARB_texture_env_add
#define GL_ARB_texture_env_add 1
GLAPI int GLAD_GL_ARB_texture_env_add;
#endif
#ifndef GL_ARB_texture_env_combine
#define GL_ARB_texture_env_combine 1
GLAPI int GLAD_GL_ARB_texture_env_combine;
#endif
#ifndef GL_ARB_texture_env_crossbar
#define GL_ARB_texture_env_crossbar 1
GLAPI int GLAD_GL_ARB_texture_env_crossbar;
#endif
#ifndef GL_ARB_texture_env_dot3
#define GL_ARB_texture_env_dot3 1
GLAPI int GLAD_GL_ARB_texture_env_dot3;
#endif
#ifndef GL_ARB_texture_filter_anisotropic
#define GL_ARB_texture_filter_anisotropic 1
GLAPI int GLAD_GL_ARB_texture_filter_anisotropic;
#endif
#ifndef GL_ARB_texture_filter_minmax
#define GL_ARB_texture_filter_minmax 1
GLAPI int GLAD_GL_ARB_texture_filter_minmax;
#endif
#ifndef GL_ARB_texture_float
#define GL_ARB_texture_float 1
GLAPI int GLAD_GL_ARB_texture_float;
#endif
#ifndef GL_ARB_texture_gather
#define GL_ARB_texture_gather 1
GLAPI int GLAD_GL_ARB_texture_gather;
#endif
#ifndef GL_ARB_texture_mirror_clamp_to_edge
#define GL_ARB_texture_mirror_clamp_to_edge 1
GLAPI int GLAD_GL_ARB_texture_mirror_clamp_to_edge;
#endif
#ifndef GL_ARB_texture_mirrored_repeat
#define GL_ARB_texture_mirrored_repeat 1
GLAPI int GLAD_GL_ARB_texture_mirrored_repeat;
#endif
#ifndef GL_ARB_texture_multisample
#define GL_ARB_texture_multisample 1
GLAPI int GLAD_GL_ARB_texture_multisample;
#endif
#ifndef GL_ARB_texture_non_power_of_two
#define GL_ARB_texture_non_power_of_two 1
GLAPI int GLAD_GL_ARB_texture_non_power_of_two;
#endif
```

```
#ifndef GL_ARB_texture_query_levels
#define GL_ARB_texture_query_levels 1
GLAPI int GLAD_GL_ARB_texture_query_levels;
#endif
#ifndef GL_ARB_texture_query_lod
#define GL_ARB_texture_query_lod 1
GLAPI int GLAD_GL_ARB_texture_query_lod;
#endif
#ifndef GL_ARB_texture_rectangle
#define GL_ARB_texture_rectangle 1
GLAPI int GLAD_GL_ARB_texture_rectangle;
#endif
#ifndef GL_ARB_texture_rg
#define GL_ARB_texture_rg 1
GLAPI int GLAD_GL_ARB_texture_rg;
#endif
#ifndef GL_ARB_texture_rgb10_a2ui
#define GL_ARB_texture_rgb10_a2ui 1
GLAPI int GLAD_GL_ARB_texture_rgb10_a2ui;
#endif
#ifndef GL_ARB_texture_stencil8
#define GL_ARB_texture_stencil8 1
GLAPI int GLAD_GL_ARB_texture_stencil8;
#endif
#ifndef GL_ARB_texture_storage
#define GL_ARB_texture_storage 1
GLAPI int GLAD_GL_ARB_texture_storage;
typedef void (APIENTRYP PFNGLTEXSTORAGE1DPROC)(GLenum target, GLsizei levels, GLenum internalformat, GLsizei width);
GLAPI PFNGLTEXSTORAGE1DPROC glad_glTexStorage1D;
#define glTexStorage1D glad_glTexStorage1D
typedef void (APIENTRYP PFNGLTEXSTORAGE2DPROC)(GLenum target, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLTEXSTORAGE2DPROC glad_glTexStorage2D;
#define glTexStorage2D glad_glTexStorage2D
typedef void (APIENTRYP PFNGLTEXSTORAGE3DPROC)(GLenum target, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth);
GLAPI PFNGLTEXSTORAGE3DPROC glad_glTexStorage3D;
#define glTexStorage3D glad_glTexStorage3D
#endif
#ifndef GL_ARB_texture_storage_multisample
#define GL_ARB_texture_storage_multisample 1
GLAPI int GLAD_GL_ARB_texture_storage_multisample;
typedef void (APIENTRYP PFNGLTEXSTORAGE2DMULTISAMPLEPROC)(GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXSTORAGE2DMULTISAMPLEPROC glad_glTexStorage2DMultisample;
#define glTexStorage2DMultisample glad_glTexStorage2DMultisample
typedef void (APIENTRYP PFNGLTEXSTORAGE3DMULTISAMPLEPROC)(GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXSTORAGE3DMULTISAMPLEPROC glad_glTexStorage3DMultisample;
#define glTexStorage3DMultisample glad_glTexStorage3DMultisample
#endif
#ifndef GL_ARB_texture_swizzle
#define GL_ARB_texture_swizzle 1
GLAPI int GLAD_GL_ARB_texture_swizzle;
#endif
#ifndef GL_ARB_texture_view
#define GL_ARB_texture_view 1
GLAPI int GLAD_GL_ARB_texture_view;
typedef void (APIENTRYP PFNGLTEXTUREVIEWPROC)(GLuint texture, GLenum target, GLuint origtexture, GLenum internalformat, GLuint minlevel, GLuint numlevels, GLuint minlayer, GLuint numlayers);
GLAPI PFNGLTEXTUREVIEWPROC glad_glTextureView;
#define glTextureView glad_glTextureView
#endif
#ifndef GL_ARB_timer_query
#define GL_ARB_timer_query 1
GLAPI int GLAD_GL_ARB_timer_query;
#endif
#ifndef GL_ARB_transform_feedback2
#define GL_ARB_transform_feedback2 1
GLAPI int GLAD_GL_ARB_transform_feedback2;
#endif
#ifndef GL_ARB_transform_feedback3
#define GL_ARB_transform_feedback3 1
GLAPI int GLAD_GL_ARB_transform_feedback3;
#endif
#ifndef GL_ARB_transform_feedback_instanced
#define GL_ARB_transform_feedback_instanced 1
GLAPI int GLAD_GL_ARB_transform_feedback_instanced;
typedef void (APIENTRYP PFNGLDRAWTRANSFORMFEEDBACKINSTANCEDPROC)(GLenum mode, GLuint id, GLsizei instancecount);
GLAPI PFNGLDRAWTRANSFORMFEEDBACKINSTANCEDPROC glad_glDrawTransformFeedbackInstanced;
#define glDrawTransformFeedbackInstanced glad_glDrawTransformFeedbackInstanced
typedef void (APIENTRYP PFNGLDRAWTRANSFORMFEEDBACKSTREAMINSTANCEDPROC)(GLenum mode, GLuint id, GLuint stream, GLsizei instancecount);
GLAPI PFNGLDRAWTRANSFORMFEEDBACKSTREAMINSTANCEDPROC glad_glDrawTransformFeedbackStreamInstanced;
#define glDrawTransformFeedbackStreamInstanced glad_glDrawTransformFeedbackStreamInstanced
```

```
#endif
#ifndef GL_ARB_transform_feedback_overflow_query
#define GL_ARB_transform_feedback_overflow_query 1
GLAPI int GLAD_GL_ARB_transform_feedback_overflow_query;
#endif
#ifndef GL_ARB_transpose_matrix
#define GL_ARB_transpose_matrix 1
GLAPI int GLAD_GL_ARB_transpose_matrix;
typedef void (APIENTRYP PFNGLLOADTRANSPOSEMATRIXFARBPROC)(const GLfloat *m);
GLAPI PFNGLLOADTRANSPOSEMATRIXFARBPROC glad_glLoadTransposeMatrixfARB;
#define glLoadTransposeMatrixfARB glad_glLoadTransposeMatrixfARB
typedef void (APIENTRYP PFNGLLOADTRANSPOSEMATRIXDARBPROC)(const GLdouble *m);
GLAPI PFNGLLOADTRANSPOSEMATRIXDARBPROC glad_glLoadTransposeMatrixdARB;
#define glLoadTransposeMatrixdARB glad_glLoadTransposeMatrixdARB
typedef void (APIENTRYP PFNGLMULTTRANSPOSEMATRIXFARBPROC)(const GLfloat *m);
GLAPI PFNGLMULTTRANSPOSEMATRIXFARBPROC glad_glMultTransposeMatrixfARB;
#define glMultTransposeMatrixfARB glad_glMultTransposeMatrixfARB
typedef void (APIENTRYP PFNGLMULTTRANSPOSEMATRIXDARBPROC)(const GLdouble *m);
GLAPI PFNGLMULTTRANSPOSEMATRIXDARBPROC glad_glMultTransposeMatrixdARB;
#define glMultTransposeMatrixdARB glad_glMultTransposeMatrixdARB
#endif
#ifndef GL_ARB_uniform_buffer_object
#define GL_ARB_uniform_buffer_object 1
GLAPI int GLAD_GL_ARB_uniform_buffer_object;
#endif
#ifndef GL_ARB_vertex_array_bgra
#define GL_ARB_vertex_array_bgra 1
GLAPI int GLAD_GL_ARB_vertex_array_bgra;
#endif
#ifndef GL_ARB_vertex_array_object
#define GL_ARB_vertex_array_object 1
GLAPI int GLAD_GL_ARB_vertex_array_object;
#endif
#ifndef GL_ARB_vertex_attrib_64bit
#define GL_ARB_vertex_attrib_64bit 1
GLAPI int GLAD_GL_ARB_vertex_attrib_64bit;
#endif
#ifndef GL_ARB_vertex_attrib_binding
#define GL_ARB_vertex_attrib_binding 1
GLAPI int GLAD_GL_ARB_vertex_attrib_binding;
typedef void (APIENTRYP PFNGLBINDVERTEXBUFFERPROC)(GLuint bindingindex, GLuint buffer, GLintptr offset, GLsizei stride);
GLAPI PFNGLBINDVERTEXBUFFERPROC glad_glBindVertexBuffer;
#define glBindVertexBuffer glad_glBindVertexBuffer
typedef void (APIENTRYP PFNGLVERTEXATTRIBFORMATPROC)(GLuint attribindex, GLint size, GLenum type, GLboolean normalized, GLuint relativeoffset);
GLAPI PFNGLVERTEXATTRIBFORMATPROC glad_glVertexAttribFormat;
#define glVertexAttribFormat glad_glVertexAttribFormat
typedef void (APIENTRYP PFNGLVERTEXATTRIBIFORMATPROC)(GLuint attribindex, GLint size, GLenum type, GLuint relativeoffset);
GLAPI PFNGLVERTEXATTRIBIFORMATPROC glad_glVertexAttribIFormat;
#define glVertexAttribIFormat glad_glVertexAttribIFormat
typedef void (APIENTRYP PFNGLVERTEXATTRIBLFORMATPROC)(GLuint attribindex, GLint size, GLenum type, GLuint relativeoffset);
GLAPI PFNGLVERTEXATTRIBLFORMATPROC glad_glVertexAttribLFormat;
#define glVertexAttribLFormat glad_glVertexAttribLFormat
typedef void (APIENTRYP PFNGLVERTEXATTRIBBINDINGPROC)(GLuint attribindex, GLuint bindingindex);
GLAPI PFNGLVERTEXATTRIBBINDINGPROC glad_glVertexAttribBinding;
#define glVertexAttribBinding glad_glVertexAttribBinding
typedef void (APIENTRYP PFNGLVERTEXBINDINGDIVISORPROC)(GLuint bindingindex, GLuint divisor);
GLAPI PFNGLVERTEXBINDINGDIVISORPROC glad_glVertexBindingDivisor;
#define glVertexBindingDivisor glad_glVertexBindingDivisor
#endif
#ifndef GL_ARB_vertex_blend
#define GL_ARB_vertex_blend 1
GLAPI int GLAD_GL_ARB_vertex_blend;
typedef void (APIENTRYP PFNGLWEIGHTBVARBPROC)(GLint size, const GLbyte *weights);
GLAPI PFNGLWEIGHTBVARBPROC glad_glWeightbvARB;
#define glWeightbvARB glad_glWeightbvARB
typedef void (APIENTRYP PFNGLWEIGHTSVARBPROC)(GLint size, const GLshort *weights);
GLAPI PFNGLWEIGHTSVARBPROC glad_glWeightsvARB;
#define glWeightsvARB glad_glWeightsvARB
typedef void (APIENTRYP PFNGLWEIGHTIVARBPROC)(GLint size, const GLint *weights);
GLAPI PFNGLWEIGHTIVARBPROC glad_glWeightivARB;
#define glWeightivARB glad_glWeightivARB
typedef void (APIENTRYP PFNGLWEIGHTFVARBPROC)(GLint size, const GLfloat *weights);
GLAPI PFNGLWEIGHTFVARBPROC glad_glWeightfvARB;
#define glWeightfvARB glad_glWeightfvARB
typedef void (APIENTRYP PFNGLWEIGHTDVARBPROC)(GLint size, const GLdouble *weights);
GLAPI PFNGLWEIGHTDVARBPROC glad_glWeightdvARB;
#define glWeightdvARB glad_glWeightdvARB
typedef void (APIENTRYP PFNGLWEIGHTUBVARBPROC)(GLint size, const GLubyte *weights);
GLAPI PFNGLWEIGHTUBVARBPROC glad_glWeightubvARB;
#define glWeightubvARB glad_glWeightubvARB
typedef void (APIENTRYP PFNGLWEIGHTUSVARBPROC)(GLint size, const GLushort *weights);
GLAPI PFNGLWEIGHTUSVARBPROC glad_glWeightusvARB;
#define glWeightusvARB glad_glWeightusvARB
```

```
typedef void (APIENTRYP PFNGLWEIGHTUIVARBPROC)(GLint size, const GLuint *weights);
GLAPI PFNGLWEIGHTUIVARBPROC glad_glWeightuivARB;
#define glWeightuivARB glad_glWeightuivARB
typedef void (APIENTRYP PFNGLWEIGHTPOINTERARBPROC)(GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLWEIGHTPOINTERARBPROC glad_glWeightPointerARB;
#define glWeightPointerARB glad_glWeightPointerARB
typedef void (APIENTRYP PFNGLVERTEXBLENDARBPROC)(GLint count);
GLAPI PFNGLVERTEXBLENDARBPROC glad_glVertexBlendARB;
#define glVertexBlendARB glad_glVertexBlendARB
#endif
#ifndef GL_ARB_vertex_buffer_object
#define GL_ARB_vertex_buffer_object 1
GLAPI int GLAD_GL_ARB_vertex_buffer_object;
typedef void (APIENTRYP PFNGLBINDBUFFERARBPROC)(GLenum target, GLuint buffer);
GLAPI PFNGLBINDBUFFERARBPROC glad_glBindBufferARB;
#define glBindBufferARB glad_glBindBufferARB
typedef void (APIENTRYP PFNGLDELETEBUFFERSARBPROC)(GLsizei n, const GLuint *buffers);
GLAPI PFNGLDELETEBUFFERSARBPROC glad_glDeleteBuffersARB;
#define glDeleteBuffersARB glad_glDeleteBuffersARB
typedef void (APIENTRYP PFNGLGENBUFFERSARBPROC)(GLsizei n, GLuint *buffers);
GLAPI PFNGLGENBUFFERSARBPROC glad_glGenBuffersARB;
#define glGenBuffersARB glad_glGenBuffersARB
typedef GLboolean (APIENTRYP PFNGLISBUFFERARBPROC)(GLuint buffer);
GLAPI PFNGLISBUFFERARBPROC glad_glIsBufferARB;
#define glIsBufferARB glad_glIsBufferARB
typedef void (APIENTRYP PFNGLBUFFERDATAARBPROC)(GLenum target, GLsizeiptrARB size, const void *data, GLenum usage);
GLAPI PFNGLBUFFERDATAARBPROC glad_glBufferDataARB;
#define glBufferDataARB glad_glBufferDataARB
typedef void (APIENTRYP PFNGLBUFFERSUBDATAARBPROC)(GLenum target, GLintptrARB offset, GLsizeiptrARB size, const void *data);
GLAPI PFNGLBUFFERSUBDATAARBPROC glad_glBufferSubDataARB;
#define glBufferSubDataARB glad_glBufferSubDataARB
typedef void (APIENTRYP PFNGLGETBUFFERSUBDATAARBPROC)(GLenum target, GLintptrARB offset, GLsizeiptrARB size, void *data);
GLAPI PFNGLGETBUFFERSUBDATAARBPROC glad_glGetBufferSubDataARB;
#define glGetBufferSubDataARB glad_glGetBufferSubDataARB
typedef void * (APIENTRYP PFNGLMAPBUFFERARBPROC)(GLenum target, GLenum access);
GLAPI PFNGLMAPBUFFERARBPROC glad_glMapBufferARB;
#define glMapBufferARB glad_glMapBufferARB
typedef GLboolean (APIENTRYP PFNGLUNMAPBUFFERARBPROC)(GLenum target);
GLAPI PFNGLUNMAPBUFFERARBPROC glad_glUnmapBufferARB;
#define glUnmapBufferARB glad_glUnmapBufferARB
typedef void (APIENTRYP PFNGLGETBUFFERPARAMETERIVARBPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETBUFFERPARAMETERIVARBPROC glad_glGetBufferParameterivARB;
#define glGetBufferParameterivARB glad_glGetBufferParameterivARB
typedef void (APIENTRYP PFNGLGETBUFFERPOINTERVARBPROC)(GLenum target, GLenum pname, void **params);
GLAPI PFNGLGETBUFFERPOINTERVARBPROC glad_glGetBufferPointervARB;
#define glGetBufferPointervARB glad_glGetBufferPointervARB
#endif
#ifndef GL_ARB_vertex_program
#define GL_ARB_vertex_program 1
GLAPI int GLAD_GL_ARB_vertex_program;
typedef void (APIENTRYP PFNGLVERTEXATTRIB1DARBPROC)(GLuint index, GLdouble x);
GLAPI PFNGLVERTEXATTRIB1DARBPROC glad_glVertexAttrib1dARB;
#define glVertexAttrib1dARB glad_glVertexAttrib1dARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB1DVARBPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB1DVARBPROC glad_glVertexAttrib1dvARB;
#define glVertexAttrib1dvARB glad_glVertexAttrib1dvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB1FARBPROC)(GLuint index, GLfloat x);
GLAPI PFNGLVERTEXATTRIB1FARBPROC glad_glVertexAttrib1fARB;
#define glVertexAttrib1fARB glad_glVertexAttrib1fARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB1FVARBPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB1FVARBPROC glad_glVertexAttrib1fvARB;
#define glVertexAttrib1fvARB glad_glVertexAttrib1fvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB1SARBPROC)(GLuint index, GLshort x);
GLAPI PFNGLVERTEXATTRIB1SARBPROC glad_glVertexAttrib1sARB;
#define glVertexAttrib1sARB glad_glVertexAttrib1sARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB1SVARBPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB1SVARBPROC glad_glVertexAttrib1svARB;
#define glVertexAttrib1svARB glad_glVertexAttrib1svARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB2DARBPROC)(GLuint index, GLdouble x, GLdouble y);
GLAPI PFNGLVERTEXATTRIB2DARBPROC glad_glVertexAttrib2dARB;
#define glVertexAttrib2dARB glad_glVertexAttrib2dARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB2DVARBPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB2DVARBPROC glad_glVertexAttrib2dvARB;
#define glVertexAttrib2dvARB glad_glVertexAttrib2dvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB2FARBPROC)(GLuint index, GLfloat x, GLfloat y);
GLAPI PFNGLVERTEXATTRIB2FARBPROC glad_glVertexAttrib2fARB;
#define glVertexAttrib2fARB glad_glVertexAttrib2fARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB2FVARBPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB2FVARBPROC glad_glVertexAttrib2fvARB;
#define glVertexAttrib2fvARB glad_glVertexAttrib2fvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB2SARBPROC)(GLuint index, GLshort x, GLshort y);
GLAPI PFNGLVERTEXATTRIB2SARBPROC glad_glVertexAttrib2sARB;
#define glVertexAttrib2sARB glad_glVertexAttrib2sARB
```

```
typedef void (APIENTRYP PFNGLVERTEXATTRIB2SVARBPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB2SVARBPROC glad_glVertexAttrib2svARB;
#define glVertexAttrib2svARB glad_glVertexAttrib2svARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB3DARBPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLVERTEXATTRIB3DARBPROC glad_glVertexAttrib3dARB;
#define glVertexAttrib3dARB glad_glVertexAttrib3dARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB3DVARBPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB3DVARBPROC glad_glVertexAttrib3dvARB;
#define glVertexAttrib3dvARB glad_glVertexAttrib3dvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB3FARBPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLVERTEXATTRIB3FARBPROC glad_glVertexAttrib3fARB;
#define glVertexAttrib3fARB glad_glVertexAttrib3fARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB3FVARBPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB3FVARBPROC glad_glVertexAttrib3fvARB;
#define glVertexAttrib3fvARB glad_glVertexAttrib3fvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB3SARBPROC)(GLuint index, GLshort x, GLshort y, GLshort z);
GLAPI PFNGLVERTEXATTRIB3SARBPROC glad_glVertexAttrib3sARB;
#define glVertexAttrib3sARB glad_glVertexAttrib3sARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB3SVARBPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB3SVARBPROC glad_glVertexAttrib3svARB;
#define glVertexAttrib3svARB glad_glVertexAttrib3svARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NBVARBPROC)(GLuint index, const GLbyte *v);
GLAPI PFNGLVERTEXATTRIB4NBVARBPROC glad_glVertexAttrib4NbvARB;
#define glVertexAttrib4NbvARB glad_glVertexAttrib4NbvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NIVARBPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIB4NIVARBPROC glad_glVertexAttrib4NivARB;
#define glVertexAttrib4NivARB glad_glVertexAttrib4NivARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NSVARBPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB4NSVARBPROC glad_glVertexAttrib4NsvARB;
#define glVertexAttrib4NsvARB glad_glVertexAttrib4NsvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUBARBPROC)(GLuint index, GLubyte x, GLubyte y, GLubyte z,
      GLubyte w);
GLAPI PFNGLVERTEXATTRIB4NUBARBPROC glad_glVertexAttrib4NubARB;
#define glVertexAttrib4NubARB glad_glVertexAttrib4NubARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUBVARBPROC)(GLuint index, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIB4NUBVARBPROC glad_glVertexAttrib4NubvARB;
#define glVertexAttrib4NubvARB glad_glVertexAttrib4NubvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUIVARBPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIB4NUIVARBPROC glad_glVertexAttrib4NuivARB;
#define glVertexAttrib4NuivARB glad_glVertexAttrib4NuivARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4NUSVARBPROC)(GLuint index, const GLushort *v);
GLAPI PFNGLVERTEXATTRIB4NUSVARBPROC glad_glVertexAttrib4NusvARB;
#define glVertexAttrib4NusvARB glad_glVertexAttrib4NusvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4BVARBPROC)(GLuint index, const GLbyte *v);
GLAPI PFNGLVERTEXATTRIB4BVARBPROC glad_glVertexAttrib4bvARB;
#define glVertexAttrib4bvARB glad_glVertexAttrib4bvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4DARBPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z,
      GLdouble w);
GLAPI PFNGLVERTEXATTRIB4DARBPROC glad_glVertexAttrib4dARB;
#define glVertexAttrib4dARB glad_glVertexAttrib4dARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4DVARBPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB4DVARBPROC glad_glVertexAttrib4dvARB;
#define glVertexAttrib4dvARB glad_glVertexAttrib4dvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4FARBPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z,
      GLfloat w);
GLAPI PFNGLVERTEXATTRIB4FARBPROC glad_glVertexAttrib4fARB;
#define glVertexAttrib4fARB glad_glVertexAttrib4fARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4FVARBPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB4FVARBPROC glad_glVertexAttrib4fvARB;
#define glVertexAttrib4fvARB glad_glVertexAttrib4fvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4IVARBPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIB4IVARBPROC glad_glVertexAttrib4ivARB;
#define glVertexAttrib4ivARB glad_glVertexAttrib4ivARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4SARBPROC)(GLuint index, GLshort x, GLshort y, GLshort z,
      GLshort w);
GLAPI PFNGLVERTEXATTRIB4SARBPROC glad_glVertexAttrib4sARB;
#define glVertexAttrib4sARB glad_glVertexAttrib4sARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4SVARBPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB4SVARBPROC glad_glVertexAttrib4svARB;
#define glVertexAttrib4svARB glad_glVertexAttrib4svARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4UBVARBPROC)(GLuint index, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIB4UBVARBPROC glad_glVertexAttrib4ubvARB;
#define glVertexAttrib4ubvARB glad_glVertexAttrib4ubvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4UIVARBPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIB4UIVARBPROC glad_glVertexAttrib4uivARB;
#define glVertexAttrib4uivARB glad_glVertexAttrib4uivARB
typedef void (APIENTRYP PFNGLVERTEXATTRIB4USVARBPROC)(GLuint index, const GLushort *v);
GLAPI PFNGLVERTEXATTRIB4USVARBPROC glad_glVertexAttrib4usvARB;
#define glVertexAttrib4usvARB glad_glVertexAttrib4usvARB
typedef void (APIENTRYP PFNGLVERTEXATTRIBPOINTERARBPROC)(GLuint index, GLint size, GLenum type,
      GLboolean normalized, GLsizei stride, const void *pointer);
GLAPI PFNGLVERTEXATTRIBPOINTERARBPROC glad_glVertexAttribPointerARB;
#define glVertexAttribPointerARB glad_glVertexAttribPointerARB
typedef void (APIENTRYP PFNGLENABLEVERTEXATTRIBARRAYARBPROC)(GLuint index);
GLAPI PFNGLENABLEVERTEXATTRIBARRAYARBPROC glad_glEnableVertexAttribArrayARB;
#define glEnableVertexAttribArrayARB glad_glEnableVertexAttribArrayARB
typedef void (APIENTRYP PFNGLDISABLEVERTEXATTRIBARRAYARBPROC)(GLuint index);
```

```
GLAPI PFNGLDISABLEVERTEXATTRIBARRAYARBPROC glad_glDisableVertexAttribArrayARB;
#define glDisableVertexAttribArrayARB glad_glDisableVertexAttribArrayARB
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBDVARBPROC)(GLuint index, GLenum pname, GLdouble *params);
GLAPI PFNGLGETVERTEXATTRIBDVARBPROC glad_glGetVertexAttribdvARB;
#define glGetVertexAttribdvARB glad_glGetVertexAttribdvARB
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBFVARBPROC)(GLuint index, GLenum pname, GLfloat *params);
GLAPI PFNGLGETVERTEXATTRIBFVARBPROC glad_glGetVertexAttribfvARB;
#define glGetVertexAttribfvARB glad_glGetVertexAttribfvARB
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBIVARBPROC)(GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETVERTEXATTRIBIVARBPROC glad_glGetVertexAttribivARB;
#define glGetVertexAttribivARB glad_glGetVertexAttribivARB
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBPOINTERVARBPROC)(GLuint index, GLenum pname, void **pointer);
GLAPI PFNGLGETVERTEXATTRIBPOINTERVARBPROC glad_glGetVertexAttribPointervARB;
#define glGetVertexAttribPointervARB glad_glGetVertexAttribPointervARB
#endif
#ifndef GL_ARB_vertex_shader
#define GL_ARB_vertex_shader 1
GLAPI int GLAD_GL_ARB_vertex_shader;
typedef void (APIENTRYP PFNGLBINDATTRIBLOCATIONARBPROC)(GLhandleARB programObj, GLuint index, const GLcharARB *name);
GLAPI PFNGLBINDATTRIBLOCATIONARBPROC glad_glBindAttribLocationARB;
#define glBindAttribLocationARB glad_glBindAttribLocationARB
typedef void (APIENTRYP PFNGLGETACTIVEATTRIBARBPROC)(GLhandleARB programObj, GLuint index, GLsizei maxLength, GLsizei *length, GLint *size, GLenum *type, GLcharARB *name);
GLAPI PFNGLGETACTIVEATTRIBARBPROC glad_glGetActiveAttribARB;
#define glGetActiveAttribARB glad_glGetActiveAttribARB
typedef GLint (APIENTRYP PFNGLGETATTRIBLOCATIONARBPROC)(GLhandleARB programObj, const GLcharARB *name);
GLAPI PFNGLGETATTRIBLOCATIONARBPROC glad_glGetAttribLocationARB;
#define glGetAttribLocationARB glad_glGetAttribLocationARB
#endif
#ifndef GL_ARB_vertex_type_10f_11f_11f_rev
#define GL_ARB_vertex_type_10f_11f_11f_rev 1
GLAPI int GLAD_GL_ARB_vertex_type_10f_11f_11f_rev;
#endif
#ifndef GL_ARB_vertex_type_2_10_10_10_rev
#define GL_ARB_vertex_type_2_10_10_10_rev 1
GLAPI int GLAD_GL_ARB_vertex_type_2_10_10_10_rev;
#endif
#ifndef GL_ARB_viewport_array
#define GL_ARB_viewport_array 1
GLAPI int GLAD_GL_ARB_viewport_array;
typedef void (APIENTRYP PFNGLDEPTHRANGEARRAYDVNVPROC)(GLuint first, GLsizei count, const GLdouble *v);
GLAPI PFNGLDEPTHRANGEARRAYDVNVPROC glad_glDepthRangeArraydvNV;
#define glDepthRangeArraydvNV glad_glDepthRangeArraydvNV
typedef void (APIENTRYP PFNGLDEPTHRANGEINDEXEDDNVPROC)(GLuint index, GLdouble n, GLdouble f);
GLAPI PFNGLDEPTHRANGEINDEXEDDNVPROC glad_glDepthRangeIndexeddNV;
#define glDepthRangeIndexeddNV glad_glDepthRangeIndexeddNV
#endif
#ifndef GL_ARB_window_pos
#define GL_ARB_window_pos 1
GLAPI int GLAD_GL_ARB_window_pos;
typedef void (APIENTRYP PFNGLWINDOWPOS2DARBPROC)(GLdouble x, GLdouble y);
GLAPI PFNGLWINDOWPOS2DARBPROC glad_glWindowPos2dARB;
#define glWindowPos2dARB glad_glWindowPos2dARB
typedef void (APIENTRYP PFNGLWINDOWPOS2DVARBPROC)(const GLdouble *v);
GLAPI PFNGLWINDOWPOS2DVARBPROC glad_glWindowPos2dvARB;
#define glWindowPos2dvARB glad_glWindowPos2dvARB
typedef void (APIENTRYP PFNGLWINDOWPOS2FARBPROC)(GLfloat x, GLfloat y);
GLAPI PFNGLWINDOWPOS2FARBPROC glad_glWindowPos2fARB;
#define glWindowPos2fARB glad_glWindowPos2fARB
typedef void (APIENTRYP PFNGLWINDOWPOS2FVARBPROC)(const GLfloat *v);
GLAPI PFNGLWINDOWPOS2FVARBPROC glad_glWindowPos2fvARB;
#define glWindowPos2fvARB glad_glWindowPos2fvARB
typedef void (APIENTRYP PFNGLWINDOWPOS2IARBPROC)(GLint x, GLint y);
GLAPI PFNGLWINDOWPOS2IARBPROC glad_glWindowPos2iARB;
#define glWindowPos2iARB glad_glWindowPos2iARB
typedef void (APIENTRYP PFNGLWINDOWPOS2IVARBPROC)(const GLint *v);
GLAPI PFNGLWINDOWPOS2IVARBPROC glad_glWindowPos2ivARB;
#define glWindowPos2ivARB glad_glWindowPos2ivARB
typedef void (APIENTRYP PFNGLWINDOWPOS2SARBPROC)(GLshort x, GLshort y);
GLAPI PFNGLWINDOWPOS2SARBPROC glad_glWindowPos2sARB;
#define glWindowPos2sARB glad_glWindowPos2sARB
typedef void (APIENTRYP PFNGLWINDOWPOS2SVARBPROC)(const GLshort *v);
GLAPI PFNGLWINDOWPOS2SVARBPROC glad_glWindowPos2svARB;
#define glWindowPos2svARB glad_glWindowPos2svARB
typedef void (APIENTRYP PFNGLWINDOWPOS3DARBPROC)(GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLWINDOWPOS3DARBPROC glad_glWindowPos3dARB;
#define glWindowPos3dARB glad_glWindowPos3dARB
typedef void (APIENTRYP PFNGLWINDOWPOS3DVARBPROC)(const GLdouble *v);
GLAPI PFNGLWINDOWPOS3DVARBPROC glad_glWindowPos3dvARB;
#define glWindowPos3dvARB glad_glWindowPos3dvARB
typedef void (APIENTRYP PFNGLWINDOWPOS3FARBPROC)(GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLWINDOWPOS3FARBPROC glad_glWindowPos3fARB;
#define glWindowPos3fARB glad_glWindowPos3fARB
typedef void (APIENTRYP PFNGLWINDOWPOS3FVARBPROC)(const GLfloat *v);
```

```
GLAPI PFNGLWINDOWPOS3FVARBPROC glad_glWindowPos3fvARB;
#define glWindowPos3fvARB glad_glWindowPos3fvARB
typedef void (APIENTRYP PFNGLWINDOWPOS3IARBPROC)(GLint x, GLint y, GLint z);
GLAPI PFNGLWINDOWPOS3IARBPROC glad_glWindowPos3iARB;
#define glWindowPos3iARB glad_glWindowPos3iARB
typedef void (APIENTRYP PFNGLWINDOWPOS3IVARBPROC)(const GLint *v);
GLAPI PFNGLWINDOWPOS3IVARBPROC glad_glWindowPos3ivARB;
#define glWindowPos3ivARB glad_glWindowPos3ivARB
typedef void (APIENTRYP PFNGLWINDOWPOS3SARBPROC)(GLshort x, GLshort y, GLshort z);
GLAPI PFNGLWINDOWPOS3SARBPROC glad_glWindowPos3sARB;
#define glWindowPos3sARB glad_glWindowPos3sARB
typedef void (APIENTRYP PFNGLWINDOWPOS3SVARBPROC)(const GLshort *v);
GLAPI PFNGLWINDOWPOS3SVARBPROC glad_glWindowPos3svARB;
#define glWindowPos3svARB glad_glWindowPos3svARB
#endif
#ifndef GL_ATI_draw_buffers
#define GL_ATI_draw_buffers 1
GLAPI int GLAD_GL_ATI_draw_buffers;
typedef void (APIENTRYP PFNGLDRAWBUFFERSATIPROC)(GLsizei n, const GLenum *bufs);
GLAPI PFNGLDRAWBUFFERSATIPROC glad_glDrawBuffersATI;
#define glDrawBuffersATI glad_glDrawBuffersATI
#endif
#ifndef GL_ATI_element_array
#define GL_ATI_element_array 1
GLAPI int GLAD_GL_ATI_element_array;
typedef void (APIENTRYP PFNGLELEMENTPOINTERATIPROC)(GLenum type, const void *pointer);
GLAPI PFNGLELEMENTPOINTERATIPROC glad_glElementPointerATI;
#define glElementPointerATI glad_glElementPointerATI
typedef void (APIENTRYP PFNGLDRAWELEMENTARRAYATIPROC)(GLenum mode, GLsizei count);
GLAPI PFNGLDRAWELEMENTARRAYATIPROC glad_glDrawElementArrayATI;
#define glDrawElementArrayATI glad_glDrawElementArrayATI
typedef void (APIENTRYP PFNGLDRAWRANGEELEMENTARRAYATIPROC)(GLenum mode, GLuint start, GLuint end, GLsizei count);
GLAPI PFNGLDRAWRANGEELEMENTARRAYATIPROC glad_glDrawRangeElementArrayATI;
#define glDrawRangeElementArrayATI glad_glDrawRangeElementArrayATI
#endif
#ifndef GL_ATI_envmap_bumpmap
#define GL_ATI_envmap_bumpmap 1
GLAPI int GLAD_GL_ATI_envmap_bumpmap;
typedef void (APIENTRYP PFNGLTEXBUMPPARAMETERIVATIPROC)(GLenum pname, const GLint *param);
GLAPI PFNGLTEXBUMPPARAMETERIVATIPROC glad_glTexBumpParameterivATI;
#define glTexBumpParameterivATI glad_glTexBumpParameterivATI
typedef void (APIENTRYP PFNGLTEXBUMPPARAMETERFVATIPROC)(GLenum pname, const GLfloat *param);
GLAPI PFNGLTEXBUMPPARAMETERFVATIPROC glad_glTexBumpParameterfvATI;
#define glTexBumpParameterfvATI glad_glTexBumpParameterfvATI
typedef void (APIENTRYP PFNGLGETTEXBUMPPARAMETERIVATIPROC)(GLenum pname, GLint *param);
GLAPI PFNGLGETTEXBUMPPARAMETERIVATIPROC glad_glGetTexBumpParameterivATI;
#define glGetTexBumpParameterivATI glad_glGetTexBumpParameterivATI
typedef void (APIENTRYP PFNGLGETTEXBUMPPARAMETERFVATIPROC)(GLenum pname, GLfloat *param);
GLAPI PFNGLGETTEXBUMPPARAMETERFVATIPROC glad_glGetTexBumpParameterfvATI;
#define glGetTexBumpParameterfvATI glad_glGetTexBumpParameterfvATI
#endif
#ifndef GL_ATI_fragment_shader
#define GL_ATI_fragment_shader 1
GLAPI int GLAD_GL_ATI_fragment_shader;
typedef GLuint (APIENTRYP PFNGLGENFRAGMENTSHADERSATIPROC)(GLuint range);
GLAPI PFNGLGENFRAGMENTSHADERSATIPROC glad_glGenFragmentShadersATI;
#define glGenFragmentShadersATI glad_glGenFragmentShadersATI
typedef void (APIENTRYP PFNGLBINDFRAGMENTSHADERATIPROC)(GLuint id);
GLAPI PFNGLBINDFRAGMENTSHADERATIPROC glad_glBindFragmentShaderATI;
#define glBindFragmentShaderATI glad_glBindFragmentShaderATI
typedef void (APIENTRYP PFNGLDELETEFRAGMENTSHADERATIPROC)(GLuint id);
GLAPI PFNGLDELETEFRAGMENTSHADERATIPROC glad_glDeleteFragmentShaderATI;
#define glDeleteFragmentShaderATI glad_glDeleteFragmentShaderATI
typedef void (APIENTRYP PFNGLBEGINFRAGMENTSHADERATIPROC)(void);
GLAPI PFNGLBEGINFRAGMENTSHADERATIPROC glad_glBeginFragmentShaderATI;
#define glBeginFragmentShaderATI glad_glBeginFragmentShaderATI
typedef void (APIENTRYP PFNGLENDFRAGMENTSHADERATIPROC)(void);
GLAPI PFNGLENDFRAGMENTSHADERATIPROC glad_glEndFragmentShaderATI;
#define glEndFragmentShaderATI glad_glEndFragmentShaderATI
typedef void (APIENTRYP PFNGLPASSTEXCOORDATIPROC)(GLuint dst, GLuint coord, GLenum swizzle);
GLAPI PFNGLPASSTEXCOORDATIPROC glad_glPassTexCoordATI;
#define glPassTexCoordATI glad_glPassTexCoordATI
typedef void (APIENTRYP PFNGLSAMPLEMAPATIPROC)(GLuint dst, GLuint interp, GLenum swizzle);
GLAPI PFNGLSAMPLEMAPATIPROC glad_glSampleMapATI;
#define glSampleMapATI glad_glSampleMapATI
typedef void (APIENTRYP PFNGLCOLORFRAGMENTOP1ATIPROC)(GLenum op, GLuint dst, GLuint dstMask, GLuint dstMod, GLuint arg1, GLuint arg1Rep, GLuint arg1Mod);
GLAPI PFNGLCOLORFRAGMENTOP1ATIPROC glad_glColorFragmentOp1ATI;
#define glColorFragmentOp1ATI glad_glColorFragmentOp1ATI
typedef void (APIENTRYP PFNGLCOLORFRAGMENTOP2ATIPROC)(GLenum op, GLuint dst, GLuint dstMask, GLuint dstMod, GLuint arg1, GLuint arg1Rep, GLuint arg1Mod, GLuint arg2, GLuint arg2Rep, GLuint arg2Mod);
GLAPI PFNGLCOLORFRAGMENTOP2ATIPROC glad_glColorFragmentOp2ATI;
#define glColorFragmentOp2ATI glad_glColorFragmentOp2ATI
typedef void (APIENTRYP 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);
```

```
GLAPI PFNGLCOLORFRAGMENTOP3ATIPROC glad_glColorFragmentOp3ATI;
#define glColorFragmentOp3ATI glad_glColorFragmentOp3ATI
typedef void (APIENTRYP PFNGLALPHAFRAGMENTOP1ATIPROC)(GLenum op, GLuint dst, GLuint dstMod, GLuint arg1, GLuint arg1Rep, GLuint arg1Mod);
GLAPI PFNGLALPHAFRAGMENTOP1ATIPROC glad_glAlphaFragmentOp1ATI;
#define glAlphaFragmentOp1ATI glad_glAlphaFragmentOp1ATI
typedef void (APIENTRYP PFNGLALPHAFRAGMENTOP2ATIPROC)(GLenum op, GLuint dst, GLuint dstMod, GLuint arg1, GLuint arg1Rep, GLuint arg1Mod, GLuint arg2, GLuint arg2Rep, GLuint arg2Mod);
GLAPI PFNGLALPHAFRAGMENTOP2ATIPROC glad_glAlphaFragmentOp2ATI;
#define glAlphaFragmentOp2ATI glad_glAlphaFragmentOp2ATI
typedef void (APIENTRYP 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);
GLAPI PFNGLALPHAFRAGMENTOP3ATIPROC glad_glAlphaFragmentOp3ATI;
#define glAlphaFragmentOp3ATI glad_glAlphaFragmentOp3ATI
typedef void (APIENTRYP PFNGLSETFRAGMENTSHADERCONSTANTATIPROC)(GLuint dst, const GLfloat *value);
GLAPI PFNGLSETFRAGMENTSHADERCONSTANTATIPROC glad_glSetFragmentShaderConstantATI;
#define glSetFragmentShaderConstantATI glad_glSetFragmentShaderConstantATI
#endif
#ifndef GL_ATI_map_object_buffer
#define GL_ATI_map_object_buffer 1
GLAPI int GLAD_GL_ATI_map_object_buffer;
typedef void * (APIENTRYP PFNGLMAPOBJECTBUFFERATIPROC)(GLuint buffer);
GLAPI PFNGLMAPOBJECTBUFFERATIPROC glad_glMapObjectBufferATI;
#define glMapObjectBufferATI glad_glMapObjectBufferATI
typedef void (APIENTRYP PFNGLUNMAPOBJECTBUFFERATIPROC)(GLuint buffer);
GLAPI PFNGLUNMAPOBJECTBUFFERATIPROC glad_glUnmapObjectBufferATI;
#define glUnmapObjectBufferATI glad_glUnmapObjectBufferATI
#endif
#ifndef GL_ATI_meminfo
#define GL_ATI_meminfo 1
GLAPI int GLAD_GL_ATI_meminfo;
#endif
#ifndef GL_ATI_pixel_format_float
#define GL_ATI_pixel_format_float 1
GLAPI int GLAD_GL_ATI_pixel_format_float;
#endif
#ifndef GL_ATI_pn_triangles
#define GL_ATI_pn_triangles 1
GLAPI int GLAD_GL_ATI_pn_triangles;
typedef void (APIENTRYP PFNGLPNTRIANGLESIATIPROC)(GLenum pname, GLint param);
GLAPI PFNGLPNTRIANGLESIATIPROC glad_glPNTrianglesiATI;
#define glPNTrianglesiATI glad_glPNTrianglesiATI
typedef void (APIENTRYP PFNGLPNTRIANGLESFATIPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLPNTRIANGLESFATIPROC glad_glPNTrianglesfATI;
#define glPNTrianglesfATI glad_glPNTrianglesfATI
#endif
#ifndef GL_ATI_separate_stencil
#define GL_ATI_separate_stencil 1
GLAPI int GLAD_GL_ATI_separate_stencil;
typedef void (APIENTRYP PFNGLSTENCILOPSEPARATEATIPROC)(GLenum face, GLenum sfail, GLenum dpfail, GLenum dppass);
GLAPI PFNGLSTENCILOPSEPARATEATIPROC glad_glStencilOpSeparateATI;
#define glStencilOpSeparateATI glad_glStencilOpSeparateATI
typedef void (APIENTRYP PFNGLSTENCILFUNCSEPARATEATIPROC)(GLenum frontfunc, GLenum backfunc, GLint ref, GLuint mask);
GLAPI PFNGLSTENCILFUNCSEPARATEATIPROC glad_glStencilFuncSeparateATI;
#define glStencilFuncSeparateATI glad_glStencilFuncSeparateATI
#endif
#ifndef GL_ATI_text_fragment_shader
#define GL_ATI_text_fragment_shader 1
GLAPI int GLAD_GL_ATI_text_fragment_shader;
#endif
#ifndef GL_ATI_texture_env_combine3
#define GL_ATI_texture_env_combine3 1
GLAPI int GLAD_GL_ATI_texture_env_combine3;
#endif
#ifndef GL_ATI_texture_float
#define GL_ATI_texture_float 1
GLAPI int GLAD_GL_ATI_texture_float;
#endif
#ifndef GL_ATI_texture_mirror_once
#define GL_ATI_texture_mirror_once 1
GLAPI int GLAD_GL_ATI_texture_mirror_once;
#endif
#ifndef GL_ATI_vertex_array_object
#define GL_ATI_vertex_array_object 1
GLAPI int GLAD_GL_ATI_vertex_array_object;
typedef GLuint (APIENTRYP PFNGLNEWOBJECTBUFFERATIPROC)(GLsizei size, const void *pointer, GLenum usage);
GLAPI PFNGLNEWOBJECTBUFFERATIPROC glad_glNewObjectBufferATI;
#define glNewObjectBufferATI glad_glNewObjectBufferATI
typedef GLboolean (APIENTRYP PFNGLISOBJECTBUFFERATIPROC)(GLuint buffer);
GLAPI PFNGLISOBJECTBUFFERATIPROC glad_glIsObjectBufferATI;
#define glIsObjectBufferATI glad_glIsObjectBufferATI
typedef void (APIENTRYP PFNGLUPDATEOBJECTBUFFERATIPROC)(GLuint buffer, GLuint offset, GLsizei size, const void *pointer, GLenum preserve);
```

```
GLAPI PFNGLUPDATEOBJECTBUFFERATIPROC glad_glUpdateObjectBufferATI;
#define glUpdateObjectBufferATI glad_glUpdateObjectBufferATI
typedef void (APIENTRYP PFNGLGETOBJECTBUFFERFVATIPROC)(GLuint buffer, GLenum pname, GLfloat *params);
GLAPI PFNGLGETOBJECTBUFFERFVATIPROC glad_glGetObjectBufferfvATI;
#define glGetObjectBufferfvATI glad_glGetObjectBufferfvATI
typedef void (APIENTRYP PFNGLGETOBJECTBUFFERIVATIPROC)(GLuint buffer, GLenum pname, GLint *params);
GLAPI PFNGLGETOBJECTBUFFERIVATIPROC glad_glGetObjectBufferivATI;
#define glGetObjectBufferivATI glad_glGetObjectBufferivATI
typedef void (APIENTRYP PFNGLFREEOBJECTBUFFERATIPROC)(GLuint buffer);
GLAPI PFNGLFREEOBJECTBUFFERATIPROC glad_glFreeObjectBufferATI;
#define glFreeObjectBufferATI glad_glFreeObjectBufferATI
typedef void (APIENTRYP PFNGLARRAYOBJECTATIPROC)(GLenum array, GLint size, GLenum type, GLsizei stride, GLuint buffer, GLuint offset);
GLAPI PFNGLARRAYOBJECTATIPROC glad_glArrayObjectATI;
#define glArrayObjectATI glad_glArrayObjectATI
typedef void (APIENTRYP PFNGLGETARRAYOBJECTFVATIPROC)(GLenum array, GLenum pname, GLfloat *params);
GLAPI PFNGLGETARRAYOBJECTFVATIPROC glad_glGetArrayObjectfvATI;
#define glGetArrayObjectfvATI glad_glGetArrayObjectfvATI
typedef void (APIENTRYP PFNGLGETARRAYOBJECTIVATIPROC)(GLenum array, GLenum pname, GLint *params);
GLAPI PFNGLGETARRAYOBJECTIVATIPROC glad_glGetArrayObjectivATI;
#define glGetArrayObjectivATI glad_glGetArrayObjectivATI
typedef void (APIENTRYP PFNGLVARIANTARRAYOBJECTATIPROC)(GLuint id, GLenum type, GLsizei stride, GLuint buffer, GLuint offset);
GLAPI PFNGLVARIANTARRAYOBJECTATIPROC glad_glVariantArrayObjectATI;
#define glVariantArrayObjectATI glad_glVariantArrayObjectATI
typedef void (APIENTRYP PFNGLGETVARIANTARRAYOBJECTFVATIPROC)(GLuint id, GLenum pname, GLfloat *params);
GLAPI PFNGLGETVARIANTARRAYOBJECTFVATIPROC glad_glGetVariantArrayObjectfvATI;
#define glGetVariantArrayObjectfvATI glad_glGetVariantArrayObjectfvATI
typedef void (APIENTRYP PFNGLGETVARIANTARRAYOBJECTIVATIPROC)(GLuint id, GLenum pname, GLint *params);
GLAPI PFNGLGETVARIANTARRAYOBJECTIVATIPROC glad_glGetVariantArrayObjectivATI;
#define glGetVariantArrayObjectivATI glad_glGetVariantArrayObjectivATI
#endif
#ifndef GL_ATI_vertex_attrib_array_object
#define GL_ATI_vertex_attrib_array_object 1
GLAPI int GLAD_GL_ATI_vertex_attrib_array_object;
typedef void (APIENTRYP PFNGLVERTEXATTRIBARRAYOBJECTATIPROC)(GLuint index, GLint size, GLenum type, GLboolean normalized, GLsizei stride, GLuint buffer, GLuint offset);
GLAPI PFNGLVERTEXATTRIBARRAYOBJECTATIPROC glad_glVertexAttribArrayObjectATI;
#define glVertexAttribArrayObjectATI glad_glVertexAttribArrayObjectATI
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBARRAYOBJECTFVATIPROC)(GLuint index, GLenum pname, GLfloat *params);
GLAPI PFNGLGETVERTEXATTRIBARRAYOBJECTFVATIPROC glad_glGetVertexAttribArrayObjectfvATI;
#define glGetVertexAttribArrayObjectfvATI glad_glGetVertexAttribArrayObjectfvATI
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBARRAYOBJECTIVATIPROC)(GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETVERTEXATTRIBARRAYOBJECTIVATIPROC glad_glGetVertexAttribArrayObjectivATI;
#define glGetVertexAttribArrayObjectivATI glad_glGetVertexAttribArrayObjectivATI
#endif
#ifndef GL_ATI_vertex_streams
#define GL_ATI_vertex_streams 1
GLAPI int GLAD_GL_ATI_vertex_streams;
typedef void (APIENTRYP PFNGLVERTEXSTREAM1SATIPROC)(GLenum stream, GLshort x);
GLAPI PFNGLVERTEXSTREAM1SATIPROC glad_glVertexStream1sATI;
#define glVertexStream1sATI glad_glVertexStream1sATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM1SVATIPROC)(GLenum stream, const GLshort *coords);
GLAPI PFNGLVERTEXSTREAM1SVATIPROC glad_glVertexStream1svATI;
#define glVertexStream1svATI glad_glVertexStream1svATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM1IATIPROC)(GLenum stream, GLint x);
GLAPI PFNGLVERTEXSTREAM1IATIPROC glad_glVertexStream1iATI;
#define glVertexStream1iATI glad_glVertexStream1iATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM1IVATIPROC)(GLenum stream, const GLint *coords);
GLAPI PFNGLVERTEXSTREAM1IVATIPROC glad_glVertexStream1ivATI;
#define glVertexStream1ivATI glad_glVertexStream1ivATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM1FATIPROC)(GLenum stream, GLfloat x);
GLAPI PFNGLVERTEXSTREAM1FATIPROC glad_glVertexStream1fATI;
#define glVertexStream1fATI glad_glVertexStream1fATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM1FVATIPROC)(GLenum stream, const GLfloat *coords);
GLAPI PFNGLVERTEXSTREAM1FVATIPROC glad_glVertexStream1fvATI;
#define glVertexStream1fvATI glad_glVertexStream1fvATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM1DATIPROC)(GLenum stream, GLdouble x);
GLAPI PFNGLVERTEXSTREAM1DATIPROC glad_glVertexStream1dATI;
#define glVertexStream1dATI glad_glVertexStream1dATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM1DVATIPROC)(GLenum stream, const GLdouble *coords);
GLAPI PFNGLVERTEXSTREAM1DVATIPROC glad_glVertexStream1dvATI;
#define glVertexStream1dvATI glad_glVertexStream1dvATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2SATIPROC)(GLenum stream, GLshort x, GLshort y);
GLAPI PFNGLVERTEXSTREAM2SATIPROC glad_glVertexStream2sATI;
#define glVertexStream2sATI glad_glVertexStream2sATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2SVATIPROC)(GLenum stream, const GLshort *coords);
GLAPI PFNGLVERTEXSTREAM2SVATIPROC glad_glVertexStream2svATI;
#define glVertexStream2svATI glad_glVertexStream2svATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2IATIPROC)(GLenum stream, GLint x, GLint y);
GLAPI PFNGLVERTEXSTREAM2IATIPROC glad_glVertexStream2iATI;
#define glVertexStream2iATI glad_glVertexStream2iATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2IVATIPROC)(GLenum stream, const GLint *coords);
GLAPI PFNGLVERTEXSTREAM2IVATIPROC glad_glVertexStream2ivATI;
```

```
#define glVertexStream2ivATI glad_glVertexStream2ivATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2FATIPROC)(GLenum stream, GLfloat x, GLfloat y);
GLAPI PFNGLVERTEXSTREAM2FATIPROC glad_glVertexStream2fATI;
#define glVertexStream2fATI glad_glVertexStream2fATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2FVATIPROC)(GLenum stream, const GLfloat *coords);
GLAPI PFNGLVERTEXSTREAM2FVATIPROC glad_glVertexStream2fvATI;
#define glVertexStream2fvATI glad_glVertexStream2fvATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2DATIPROC)(GLenum stream, GLdouble x, GLdouble y);
GLAPI PFNGLVERTEXSTREAM2DATIPROC glad_glVertexStream2dATI;
#define glVertexStream2dATI glad_glVertexStream2dATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM2DVATIPROC)(GLenum stream, const GLdouble *coords);
GLAPI PFNGLVERTEXSTREAM2DVATIPROC glad_glVertexStream2dvATI;
#define glVertexStream2dvATI glad_glVertexStream2dvATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3SATIPROC)(GLenum stream, GLshort x, GLshort y, GLshort z);
GLAPI PFNGLVERTEXSTREAM3SATIPROC glad_glVertexStream3sATI;
#define glVertexStream3sATI glad_glVertexStream3sATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3SVATIPROC)(GLenum stream, const GLshort *coords);
GLAPI PFNGLVERTEXSTREAM3SVATIPROC glad_glVertexStream3svATI;
#define glVertexStream3svATI glad_glVertexStream3svATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3IATIPROC)(GLenum stream, GLint x, GLint y, GLint z);
GLAPI PFNGLVERTEXSTREAM3IATIPROC glad_glVertexStream3iATI;
#define glVertexStream3iATI glad_glVertexStream3iATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3IVATIPROC)(GLenum stream, const GLint *coords);
GLAPI PFNGLVERTEXSTREAM3IVATIPROC glad_glVertexStream3ivATI;
#define glVertexStream3ivATI glad_glVertexStream3ivATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3FATIPROC)(GLenum stream, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLVERTEXSTREAM3FATIPROC glad_glVertexStream3fATI;
#define glVertexStream3fATI glad_glVertexStream3fATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3FVATIPROC)(GLenum stream, const GLfloat *coords);
GLAPI PFNGLVERTEXSTREAM3FVATIPROC glad_glVertexStream3fvATI;
#define glVertexStream3fvATI glad_glVertexStream3fvATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3DATIPROC)(GLenum stream, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLVERTEXSTREAM3DATIPROC glad_glVertexStream3dATI;
#define glVertexStream3dATI glad_glVertexStream3dATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM3DVATIPROC)(GLenum stream, const GLdouble *coords);
GLAPI PFNGLVERTEXSTREAM3DVATIPROC glad_glVertexStream3dvATI;
#define glVertexStream3dvATI glad_glVertexStream3dvATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4SATIPROC)(GLenum stream, GLshort x, GLshort y, GLshort z, GLshort w);
GLAPI PFNGLVERTEXSTREAM4SATIPROC glad_glVertexStream4sATI;
#define glVertexStream4sATI glad_glVertexStream4sATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4SVATIPROC)(GLenum stream, const GLshort *coords);
GLAPI PFNGLVERTEXSTREAM4SVATIPROC glad_glVertexStream4svATI;
#define glVertexStream4svATI glad_glVertexStream4svATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4IATIPROC)(GLenum stream, GLint x, GLint y, GLint z, GLint w);
GLAPI PFNGLVERTEXSTREAM4IATIPROC glad_glVertexStream4iATI;
#define glVertexStream4iATI glad_glVertexStream4iATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4IVATIPROC)(GLenum stream, const GLint *coords);
GLAPI PFNGLVERTEXSTREAM4IVATIPROC glad_glVertexStream4ivATI;
#define glVertexStream4ivATI glad_glVertexStream4ivATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4FATIPROC)(GLenum stream, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLVERTEXSTREAM4FATIPROC glad_glVertexStream4fATI;
#define glVertexStream4fATI glad_glVertexStream4fATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4FVATIPROC)(GLenum stream, const GLfloat *coords);
GLAPI PFNGLVERTEXSTREAM4FVATIPROC glad_glVertexStream4fvATI;
#define glVertexStream4fvATI glad_glVertexStream4fvATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4DATIPROC)(GLenum stream, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLVERTEXSTREAM4DATIPROC glad_glVertexStream4dATI;
#define glVertexStream4dATI glad_glVertexStream4dATI
typedef void (APIENTRYP PFNGLVERTEXSTREAM4DVATIPROC)(GLenum stream, const GLdouble *coords);
GLAPI PFNGLVERTEXSTREAM4DVATIPROC glad_glVertexStream4dvATI;
#define glVertexStream4dvATI glad_glVertexStream4dvATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3BATIPROC)(GLenum stream, GLbyte nx, GLbyte ny, GLbyte nz);
GLAPI PFNGLNORMALSTREAM3BATIPROC glad_glNormalStream3bATI;
#define glNormalStream3bATI glad_glNormalStream3bATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3BVATIPROC)(GLenum stream, const GLbyte *coords);
GLAPI PFNGLNORMALSTREAM3BVATIPROC glad_glNormalStream3bvATI;
#define glNormalStream3bvATI glad_glNormalStream3bvATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3SATIPROC)(GLenum stream, GLshort nx, GLshort ny, GLshort nz);
GLAPI PFNGLNORMALSTREAM3SATIPROC glad_glNormalStream3sATI;
#define glNormalStream3sATI glad_glNormalStream3sATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3SVATIPROC)(GLenum stream, const GLshort *coords);
GLAPI PFNGLNORMALSTREAM3SVATIPROC glad_glNormalStream3svATI;
#define glNormalStream3svATI glad_glNormalStream3svATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3IATIPROC)(GLenum stream, GLint nx, GLint ny, GLint nz);
GLAPI PFNGLNORMALSTREAM3IATIPROC glad_glNormalStream3iATI;
#define glNormalStream3iATI glad_glNormalStream3iATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3IVATIPROC)(GLenum stream, const GLint *coords);
GLAPI PFNGLNORMALSTREAM3IVATIPROC glad_glNormalStream3ivATI;
#define glNormalStream3ivATI glad_glNormalStream3ivATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3FATIPROC)(GLenum stream, GLfloat nx, GLfloat ny, GLfloat nz);
```

```
GLAPI PFNGLNORMALSTREAM3FATIPROC glad_glNormalStream3fATI;
#define glNormalStream3fATI glad_glNormalStream3fATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3FVATIPROC)(GLenum stream, const GLfloat *coords);
GLAPI PFNGLNORMALSTREAM3FVATIPROC glad_glNormalStream3fvATI;
#define glNormalStream3fvATI glad_glNormalStream3fvATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3DATIPROC)(GLenum stream, GLdouble nx, GLdouble ny, GLdouble nz);
GLAPI PFNGLNORMALSTREAM3DATIPROC glad_glNormalStream3dATI;
#define glNormalStream3dATI glad_glNormalStream3dATI
typedef void (APIENTRYP PFNGLNORMALSTREAM3DVATIPROC)(GLenum stream, const GLdouble *coords);
GLAPI PFNGLNORMALSTREAM3DVATIPROC glad_glNormalStream3dvATI;
#define glNormalStream3dvATI glad_glNormalStream3dvATI
typedef void (APIENTRYP PFNGLCLIENTACTIVEVERTEXSTREAMATIPROC)(GLenum stream);
GLAPI PFNGLCLIENTACTIVEVERTEXSTREAMATIPROC glad_glClientActiveVertexStreamATI;
#define glClientActiveVertexStreamATI glad_glClientActiveVertexStreamATI
typedef void (APIENTRYP PFNGLVERTEXBLENDENVIATIPROC)(GLenum pname, GLint param);
GLAPI PFNGLVERTEXBLENDENVIATIPROC glad_glVertexBlendEnviATI;
#define glVertexBlendEnviATI glad_glVertexBlendEnviATI
typedef void (APIENTRYP PFNGLVERTEXBLENDENVFATIPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLVERTEXBLENDENVFATIPROC glad_glVertexBlendEnvfATI;
#define glVertexBlendEnvfATI glad_glVertexBlendEnvfATI
#endif
#ifndef GL_EXT_422_pixels
#define GL_EXT_422_pixels 1
GLAPI int GLAD_GL_EXT_422_pixels;
#endif
#ifndef GL_EXT_EGL_image_storage
#define GL_EXT_EGL_image_storage 1
GLAPI int GLAD_GL_EXT_EGL_image_storage;
typedef void (APIENTRYP PFNGLEGLIMAGETARGETTEXSTORAGEEXTPROC)(GLenum target, GLeglImageOES image, const GLint* attrib_list);
GLAPI PFNGLEGLIMAGETARGETTEXSTORAGEEXTPROC glad_glEGLImageTargetTexStorageEXT;
#define glEGLImageTargetTexStorageEXT glad_glEGLImageTargetTexStorageEXT
typedef void (APIENTRYP PFNGLEGLIMAGETARGETTEXTURESTORAGEEXTPROC)(GLuint texture, GLeglImageOES image, const GLint* attrib_list);
GLAPI PFNGLEGLIMAGETARGETTEXTURESTORAGEEXTPROC glad_glEGLImageTargetTextureStorageEXT;
#define glEGLImageTargetTextureStorageEXT glad_glEGLImageTargetTextureStorageEXT
#endif
#ifndef GL_EXT_EGL_sync
#define GL_EXT_EGL_sync 1
GLAPI int GLAD_GL_EXT_EGL_sync;
#endif
#ifndef GL_EXT_abgr
#define GL_EXT_abgr 1
GLAPI int GLAD_GL_EXT_abgr;
#endif
#ifndef GL_EXT_bgra
#define GL_EXT_bgra 1
GLAPI int GLAD_GL_EXT_bgra;
#endif
#ifndef GL_EXT_bindable_uniform
#define GL_EXT_bindable_uniform 1
GLAPI int GLAD_GL_EXT_bindable_uniform;
typedef void (APIENTRYP PFNGLUNIFORMBUFFEREXTPROC)(GLuint program, GLint location, GLuint buffer);
GLAPI PFNGLUNIFORMBUFFEREXTPROC glad_glUniformBufferEXT;
#define glUniformBufferEXT glad_glUniformBufferEXT
typedef GLint (APIENTRYP PFNGLGETUNIFORMBUFFERSIZEEXTPROC)(GLuint program, GLint location);
GLAPI PFNGLGETUNIFORMBUFFERSIZEEXTPROC glad_glGetUniformBufferSizeEXT;
#define glGetUniformBufferSizeEXT glad_glGetUniformBufferSizeEXT
typedef GLintptr (APIENTRYP PFNGLGETUNIFORMOFFSETEXTPROC)(GLuint program, GLint location);
GLAPI PFNGLGETUNIFORMOFFSETEXTPROC glad_glGetUniformOffsetEXT;
#define glGetUniformOffsetEXT glad_glGetUniformOffsetEXT
#endif
#ifndef GL_EXT_blend_color
#define GL_EXT_blend_color 1
GLAPI int GLAD_GL_EXT_blend_color;
typedef void (APIENTRYP PFNGLBLENDCOLOREXTPROC)(GLfloat red, GLfloat green, GLfloat blue, GLfloat alpha);
GLAPI PFNGLBLENDCOLOREXTPROC glad_glBlendColorEXT;
#define glBlendColorEXT glad_glBlendColorEXT
#endif
#ifndef GL_EXT_blend_equation_separate
#define GL_EXT_blend_equation_separate 1
GLAPI int GLAD_GL_EXT_blend_equation_separate;
typedef void (APIENTRYP PFNGLBLENDEQUATIONSEPARATEEXTPROC)(GLenum modeRGB, GLenum modeAlpha);
GLAPI PFNGLBLENDEQUATIONSEPARATEEXTPROC glad_glBlendEquationSeparateEXT;
#define glBlendEquationSeparateEXT glad_glBlendEquationSeparateEXT
#endif
#ifndef GL_EXT_blend_func_separate
#define GL_EXT_blend_func_separate 1
GLAPI int GLAD_GL_EXT_blend_func_separate;
typedef void (APIENTRYP PFNGLBLENDFUNCSEPARATEEXTPROC)(GLenum sfactorRGB, GLenum dfactorRGB, GLenum sfactorAlpha, GLenum dfactorAlpha);
GLAPI PFNGLBLENDFUNCSEPARATEEXTPROC glad_glBlendFuncSeparateEXT;
#define glBlendFuncSeparateEXT glad_glBlendFuncSeparateEXT
#endif
#ifndef GL_EXT_blend_logic_op
```

```
#define GL_EXT_blend_logic_op 1
GLAPI int GLAD_GL_EXT_blend_logic_op;
#endif
#ifndef GL_EXT_blend_minmax
#define GL_EXT_blend_minmax 1
GLAPI int GLAD_GL_EXT_blend_minmax;
typedef void (APIENTRYP PFNGLBLENDEQUATIONEXTPROC)(GLenum mode);
GLAPI PFNGLBLENDEQUATIONEXTPROC glad_glBlendEquationEXT;
#define glBlendEquationEXT glad_glBlendEquationEXT
#endif
#ifndef GL_EXT_blend_subtract
#define GL_EXT_blend_subtract 1
GLAPI int GLAD_GL_EXT_blend_subtract;
#endif
#ifndef GL_EXT_clip_volume_hint
#define GL_EXT_clip_volume_hint 1
GLAPI int GLAD_GL_EXT_clip_volume_hint;
#endif
#ifndef GL_EXT_cmyka
#define GL_EXT_cmyka 1
GLAPI int GLAD_GL_EXT_cmyka;
#endif
#ifndef GL_EXT_color_subtable
#define GL_EXT_color_subtable 1
GLAPI int GLAD_GL_EXT_color_subtable;
typedef void (APIENTRYP PFNGLCOLORSUBTABLEEXTPROC)(GLenum target, GLsizei start, GLsizei count, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCOLORSUBTABLEEXTPROC glad_glColorSubTableEXT;
#define glColorSubTableEXT glad_glColorSubTableEXT
typedef void (APIENTRYP PFNGLCOPYCOLORSUBTABLEEXTPROC)(GLenum target, GLsizei start, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYCOLORSUBTABLEEXTPROC glad_glCopyColorSubTableEXT;
#define glCopyColorSubTableEXT glad_glCopyColorSubTableEXT
#endif
#ifndef GL_EXT_compiled_vertex_array
#define GL_EXT_compiled_vertex_array 1
GLAPI int GLAD_GL_EXT_compiled_vertex_array;
typedef void (APIENTRYP PFNGLLOCKARRAYSEXTPROC)(GLint first, GLsizei count);
GLAPI PFNGLLOCKARRAYSEXTPROC glad_glLockArraysEXT;
#define glLockArraysEXT glad_glLockArraysEXT
typedef void (APIENTRYP PFNGLUNLOCKARRAYSEXTPROC)(void);
GLAPI PFNGLUNLOCKARRAYSEXTPROC glad_glUnlockArraysEXT;
#define glUnlockArraysEXT glad_glUnlockArraysEXT
#endif
#ifndef GL_EXT_convolution
#define GL_EXT_convolution 1
GLAPI int GLAD_GL_EXT_convolution;
typedef void (APIENTRYP PFNGLCONVOLUTIONFILTER1DEXTPROC)(GLenum target, GLenum internalformat, GLsizei width, GLenum format, GLenum type, const void *image);
GLAPI PFNGLCONVOLUTIONFILTER1DEXTPROC glad_glConvolutionFilter1DEXT;
#define glConvolutionFilter1DEXT glad_glConvolutionFilter1DEXT
typedef void (APIENTRYP PFNGLCONVOLUTIONFILTER2DEXTPROC)(GLenum target, GLenum internalformat, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *image);
GLAPI PFNGLCONVOLUTIONFILTER2DEXTPROC glad_glConvolutionFilter2DEXT;
#define glConvolutionFilter2DEXT glad_glConvolutionFilter2DEXT
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERFEXTPROC)(GLenum target, GLenum pname, GLfloat params);
GLAPI PFNGLCONVOLUTIONPARAMETERFEXTPROC glad_glConvolutionParameterfEXT;
#define glConvolutionParameterfEXT glad_glConvolutionParameterfEXT
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERFVEXTPROC)(GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLCONVOLUTIONPARAMETERFVEXTPROC glad_glConvolutionParameterfvEXT;
#define glConvolutionParameterfvEXT glad_glConvolutionParameterfvEXT
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERIEXTPROC)(GLenum target, GLenum pname, GLint params);
GLAPI PFNGLCONVOLUTIONPARAMETERIEXTPROC glad_glConvolutionParameteriEXT;
#define glConvolutionParameteriEXT glad_glConvolutionParameteriEXT
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERIVEXTPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLCONVOLUTIONPARAMETERIVEXTPROC glad_glConvolutionParameterivEXT;
#define glConvolutionParameterivEXT glad_glConvolutionParameterivEXT
typedef void (APIENTRYP PFNGLCOPYCONVOLUTIONFILTER1DEXTPROC)(GLenum target, GLenum internalformat, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYCONVOLUTIONFILTER1DEXTPROC glad_glCopyConvolutionFilter1DEXT;
#define glCopyConvolutionFilter1DEXT glad_glCopyConvolutionFilter1DEXT
typedef void (APIENTRYP PFNGLCOPYCONVOLUTIONFILTER2DEXTPROC)(GLenum target, GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYCONVOLUTIONFILTER2DEXTPROC glad_glCopyConvolutionFilter2DEXT;
#define glCopyConvolutionFilter2DEXT glad_glCopyConvolutionFilter2DEXT
typedef void (APIENTRYP PFNGLGETCONVOLUTIONFILTEREXTPROC)(GLenum target, GLenum format, GLenum type, void *image);
GLAPI PFNGLGETCONVOLUTIONFILTEREXTPROC glad_glGetConvolutionFilterEXT;
#define glGetConvolutionFilterEXT glad_glGetConvolutionFilterEXT
typedef void (APIENTRYP PFNGLGETCONVOLUTIONPARAMETERFVEXTPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCONVOLUTIONPARAMETERFVEXTPROC glad_glGetConvolutionParameterfvEXT;
#define glGetConvolutionParameterfvEXT glad_glGetConvolutionParameterfvEXT
typedef void (APIENTRYP PFNGLGETCONVOLUTIONPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint
*params);
GLAPI PFNGLGETCONVOLUTIONPARAMETERIVEXTPROC glad_glGetConvolutionParameterivEXT;
#define glGetConvolutionParameterivEXT glad_glGetConvolutionParameterivEXT
typedef void (APIENTRYP PFNGLGETSEPARABLEFILTEREXTPROC)(GLenum target, GLenum format, GLenum type,
void *row, void *column, void *span);
GLAPI PFNGLGETSEPARABLEFILTEREXTPROC glad_glGetSeparableFilterEXT;
#define glGetSeparableFilterEXT glad_glGetSeparableFilterEXT
typedef void (APIENTRYP PFNGLSEPARABLEFILTER2DEXTPROC)(GLenum target, GLenum internalformat, GLsizei
width, GLsizei height, GLenum format, GLenum type, const void *row, const void *column);
GLAPI PFNGLSEPARABLEFILTER2DEXTPROC glad_glSeparableFilter2DEXT;
#define glSeparableFilter2DEXT glad_glSeparableFilter2DEXT
#endif
#ifndef GL_EXT_coordinate_frame
#define GL_EXT_coordinate_frame 1
GLAPI int GLAD_GL_EXT_coordinate_frame;
typedef void (APIENTRYP PFNGLTANGENT3BEXTPROC)(GLbyte tx, GLbyte ty, GLbyte tz);
GLAPI PFNGLTANGENT3BEXTPROC glad_glTangent3bEXT;
#define glTangent3bEXT glad_glTangent3bEXT
typedef void (APIENTRYP PFNGLTANGENT3BVEXTPROC)(const GLbyte *v);
GLAPI PFNGLTANGENT3BVEXTPROC glad_glTangent3bvEXT;
#define glTangent3bvEXT glad_glTangent3bvEXT
typedef void (APIENTRYP PFNGLTANGENT3DEXTPROC)(GLdouble tx, GLdouble ty, GLdouble tz);
GLAPI PFNGLTANGENT3DEXTPROC glad_glTangent3dEXT;
#define glTangent3dEXT glad_glTangent3dEXT
typedef void (APIENTRYP PFNGLTANGENT3DVEXTPROC)(const GLdouble *v);
GLAPI PFNGLTANGENT3DVEXTPROC glad_glTangent3dvEXT;
#define glTangent3dvEXT glad_glTangent3dvEXT
typedef void (APIENTRYP PFNGLTANGENT3FEXTPROC)(GLfloat tx, GLfloat ty, GLfloat tz);
GLAPI PFNGLTANGENT3FEXTPROC glad_glTangent3fEXT;
#define glTangent3fEXT glad_glTangent3fEXT
typedef void (APIENTRYP PFNGLTANGENT3FVEXTPROC)(const GLfloat *v);
GLAPI PFNGLTANGENT3FVEXTPROC glad_glTangent3fvEXT;
#define glTangent3fvEXT glad_glTangent3fvEXT
typedef void (APIENTRYP PFNGLTANGENT3IEXTPROC)(GLint tx, GLint ty, GLint tz);
GLAPI PFNGLTANGENT3IEXTPROC glad_glTangent3iEXT;
#define glTangent3iEXT glad_glTangent3iEXT
typedef void (APIENTRYP PFNGLTANGENT3IVEXTPROC)(const GLint *v);
GLAPI PFNGLTANGENT3IVEXTPROC glad_glTangent3ivEXT;
#define glTangent3ivEXT glad_glTangent3ivEXT
typedef void (APIENTRYP PFNGLTANGENT3SEXTPROC)(GLshort tx, GLshort ty, GLshort tz);
GLAPI PFNGLTANGENT3SEXTPROC glad_glTangent3sEXT;
#define glTangent3sEXT glad_glTangent3sEXT
typedef void (APIENTRYP PFNGLTANGENT3SVEXTPROC)(const GLshort *v);
GLAPI PFNGLTANGENT3SVEXTPROC glad_glTangent3svEXT;
#define glTangent3svEXT glad_glTangent3svEXT
typedef void (APIENTRYP PFNGLBINORMAL3BEXTPROC)(GLbyte bx, GLbyte by, GLbyte bz);
GLAPI PFNGLBINORMAL3BEXTPROC glad_glBinormal3bEXT;
#define glBinormal3bEXT glad_glBinormal3bEXT
typedef void (APIENTRYP PFNGLBINORMAL3BVEXTPROC)(const GLbyte *v);
GLAPI PFNGLBINORMAL3BVEXTPROC glad_glBinormal3bvEXT;
#define glBinormal3bvEXT glad_glBinormal3bvEXT
typedef void (APIENTRYP PFNGLBINORMAL3DEXTPROC)(GLdouble bx, GLdouble by, GLdouble bz);
GLAPI PFNGLBINORMAL3DEXTPROC glad_glBinormal3dEXT;
#define glBinormal3dEXT glad_glBinormal3dEXT
typedef void (APIENTRYP PFNGLBINORMAL3DVEXTPROC)(const GLdouble *v);
GLAPI PFNGLBINORMAL3DVEXTPROC glad_glBinormal3dvEXT;
#define glBinormal3dvEXT glad_glBinormal3dvEXT
typedef void (APIENTRYP PFNGLBINORMAL3FEXTPROC)(GLfloat bx, GLfloat by, GLfloat bz);
GLAPI PFNGLBINORMAL3FEXTPROC glad_glBinormal3fEXT;
#define glBinormal3fEXT glad_glBinormal3fEXT
typedef void (APIENTRYP PFNGLBINORMAL3FVEXTPROC)(const GLfloat *v);
GLAPI PFNGLBINORMAL3FVEXTPROC glad_glBinormal3fvEXT;
#define glBinormal3fvEXT glad_glBinormal3fvEXT
typedef void (APIENTRYP PFNGLBINORMAL3IEXTPROC)(GLint bx, GLint by, GLint bz);
GLAPI PFNGLBINORMAL3IEXTPROC glad_glBinormal3iEXT;
#define glBinormal3iEXT glad_glBinormal3iEXT
typedef void (APIENTRYP PFNGLBINORMAL3IVEXTPROC)(const GLint *v);
GLAPI PFNGLBINORMAL3IVEXTPROC glad_glBinormal3ivEXT;
#define glBinormal3ivEXT glad_glBinormal3ivEXT
typedef void (APIENTRYP PFNGLBINORMAL3SEXTPROC)(GLshort bx, GLshort by, GLshort bz);
GLAPI PFNGLBINORMAL3SEXTPROC glad_glBinormal3sEXT;
#define glBinormal3sEXT glad_glBinormal3sEXT
typedef void (APIENTRYP PFNGLBINORMAL3SVEXTPROC)(const GLshort *v);
GLAPI PFNGLBINORMAL3SVEXTPROC glad_glBinormal3svEXT;
#define glBinormal3svEXT glad_glBinormal3svEXT
typedef void (APIENTRYP PFNGLTANGENTPOINTEREXTPROC)(GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLTANGENTPOINTEREXTPROC glad_glTangentPointerEXT;
#define glTangentPointerEXT glad_glTangentPointerEXT
typedef void (APIENTRYP PFNGLBINORMALPOINTEREXTPROC)(GLenum type, GLsizei stride, const void
*pointer);
GLAPI PFNGLBINORMALPOINTEREXTPROC glad_glBinormalPointerEXT;
#define glBinormalPointerEXT glad_glBinormalPointerEXT
#endif
#ifndef GL_EXT_copy_texture
#define GL_EXT_copy_texture 1
GLAPI int GLAD_GL_EXT_copy_texture;
typedef void (APIENTRYP PFNGLCOPYTEXIMAGE1DEXTPROC)(GLenum target, GLint level, GLenum internalformat,
GLint x, GLint y, GLsizei width, GLint border);
GLAPI PFNGLCOPYTEXIMAGE1DEXTPROC glad_glCopyTexImage1DEXT;
#define glCopyTexImage1DEXT glad_glCopyTexImage1DEXT
typedef void (APIENTRYP PFNGLCOPYTEXIMAGE2DEXTPROC)(GLenum target, GLint level, GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
GLAPI PFNGLCOPYTEXIMAGE2DEXTPROC glad_glCopyTexImage2DEXT;
#define glCopyTexImage2DEXT glad_glCopyTexImage2DEXT
typedef void (APIENTRYP PFNGLCOPYTEXSUBIMAGE1DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYTEXSUBIMAGE1DEXTPROC glad_glCopyTexSubImage1DEXT;
#define glCopyTexSubImage1DEXT glad_glCopyTexSubImage1DEXT
typedef void (APIENTRYP PFNGLCOPYTEXSUBIMAGE2DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXSUBIMAGE2DEXTPROC glad_glCopyTexSubImage2DEXT;
#define glCopyTexSubImage2DEXT glad_glCopyTexSubImage2DEXT
typedef void (APIENTRYP PFNGLCOPYTEXSUBIMAGE3DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXSUBIMAGE3DEXTPROC glad_glCopyTexSubImage3DEXT;
#define glCopyTexSubImage3DEXT glad_glCopyTexSubImage3DEXT
#endif
#ifndef GL_EXT_cull_vertex
#define GL_EXT_cull_vertex 1
GLAPI int GLAD_GL_EXT_cull_vertex;
typedef void (APIENTRYP PFNGLCULLPARAMETERDVEXTPROC)(GLenum pname, GLdouble *params);
GLAPI PFNGLCULLPARAMETERDVEXTPROC glad_glCullParameterdvEXT;
#define glCullParameterdvEXT glad_glCullParameterdvEXT
typedef void (APIENTRYP PFNGLCULLPARAMETERFVEXTPROC)(GLenum pname, GLfloat *params);
GLAPI PFNGLCULLPARAMETERFVEXTPROC glad_glCullParameterfvEXT;
#define glCullParameterfvEXT glad_glCullParameterfvEXT
#endif
#ifndef GL_EXT_debug_label
#define GL_EXT_debug_label 1
GLAPI int GLAD_GL_EXT_debug_label;
typedef void (APIENTRYP PFNGLLABELOBJECTEXTPROC)(GLenum type, GLuint object, GLsizei length, const GLchar *label);
GLAPI PFNGLLABELOBJECTEXTPROC glad_glLabelObjectEXT;
#define glLabelObjectEXT glad_glLabelObjectEXT
typedef void (APIENTRYP PFNGLGETOBJECTLABELEXTPROC)(GLenum type, GLuint object, GLsizei bufSize, GLsizei *length, GLchar *label);
GLAPI PFNGLGETOBJECTLABELEXTPROC glad_glGetObjectLabelEXT;
#define glGetObjectLabelEXT glad_glGetObjectLabelEXT
#endif
#ifndef GL_EXT_debug_marker
#define GL_EXT_debug_marker 1
GLAPI int GLAD_GL_EXT_debug_marker;
typedef void (APIENTRYP PFNGLINSERTEVENTMARKEREXTPROC)(GLsizei length, const GLchar *marker);
GLAPI PFNGLINSERTEVENTMARKEREXTPROC glad_glInsertEventMarkerEXT;
#define glInsertEventMarkerEXT glad_glInsertEventMarkerEXT
typedef void (APIENTRYP PFNGLPUSHGROUPMARKEREXTPROC)(GLsizei length, const GLchar *marker);
GLAPI PFNGLPUSHGROUPMARKEREXTPROC glad_glPushGroupMarkerEXT;
#define glPushGroupMarkerEXT glad_glPushGroupMarkerEXT
typedef void (APIENTRYP PFNGLPOPGROUPMARKEREXTPROC)(void);
GLAPI PFNGLPOPGROUPMARKEREXTPROC glad_glPopGroupMarkerEXT;
#define glPopGroupMarkerEXT glad_glPopGroupMarkerEXT
#endif
#ifndef GL_EXT_depth_bounds_test
#define GL_EXT_depth_bounds_test 1
GLAPI int GLAD_GL_EXT_depth_bounds_test;
typedef void (APIENTRYP PFNGLDEPTHBOUNDSEXTPROC)(GLclampd zmin, GLclampd zmax);
GLAPI PFNGLDEPTHBOUNDSEXTPROC glad_glDepthBoundsEXT;
#define glDepthBoundsEXT glad_glDepthBoundsEXT
#endif
#ifndef GL_EXT_direct_state_access
#define GL_EXT_direct_state_access 1
GLAPI int GLAD_GL_EXT_direct_state_access;
typedef void (APIENTRYP PFNGLMATRIXLOADFEXTPROC)(GLenum mode, const GLfloat *m);
GLAPI PFNGLMATRIXLOADFEXTPROC glad_glMatrixLoadfEXT;
#define glMatrixLoadfEXT glad_glMatrixLoadfEXT
typedef void (APIENTRYP PFNGLMATRIXLOADDEXTPROC)(GLenum mode, const GLdouble *m);
GLAPI PFNGLMATRIXLOADDEXTPROC glad_glMatrixLoaddEXT;
#define glMatrixLoaddEXT glad_glMatrixLoaddEXT
typedef void (APIENTRYP PFNGLMATRIXMULTFEXTPROC)(GLenum mode, const GLfloat *m);
GLAPI PFNGLMATRIXMULTFEXTPROC glad_glMatrixMultfEXT;
#define glMatrixMultfEXT glad_glMatrixMultfEXT
typedef void (APIENTRYP PFNGLMATRIXMULTDEXTPROC)(GLenum mode, const GLdouble *m);
GLAPI PFNGLMATRIXMULTDEXTPROC glad_glMatrixMultdEXT;
#define glMatrixMultdEXT glad_glMatrixMultdEXT
typedef void (APIENTRYP PFNGLMATRIXLOADIDENTITYEXTPROC)(GLenum mode);
GLAPI PFNGLMATRIXLOADIDENTITYEXTPROC glad_glMatrixLoadIdentityEXT;
#define glMatrixLoadIdentityEXT glad_glMatrixLoadIdentityEXT
typedef void (APIENTRYP PFNGLMATRIXROTATEFEXTPROC)(GLenum mode, GLfloat angle, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLMATRIXROTATEFEXTPROC glad_glMatrixRotatefEXT;
#define glMatrixRotatefEXT glad_glMatrixRotatefEXT
typedef void (APIENTRYP PFNGLMATRIXROTATEDEXTPROC)(GLenum mode, GLdouble angle, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLMATRIXROTATEDEXTPROC glad_glMatrixRotatedEXT;
```

```
#define glMatrixRotatedEXT glad_glMatrixRotatedEXT
typedef void (APIENTRYP PFNGLMATRIXSCALEFEXTPROC)(GLenum mode, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLMATRIXSCALEFEXTPROC glad_glMatrixScalefEXT;
#define glMatrixScalefEXT glad_glMatrixScalefEXT
typedef void (APIENTRYP PFNGLMATRIXSCALEDEXTPROC)(GLenum mode, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLMATRIXSCALEDEXTPROC glad_glMatrixScaledEXT;
#define glMatrixScaledEXT glad_glMatrixScaledEXT
typedef void (APIENTRYP PFNGLMATRIXTRANSLATEFEXTPROC)(GLenum mode, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLMATRIXTRANSLATEFEXTPROC glad_glMatrixTranslatefEXT;
#define glMatrixTranslatefEXT glad_glMatrixTranslatefEXT
typedef void (APIENTRYP PFNGLMATRIXTRANSLATEDEXTPROC)(GLenum mode, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLMATRIXTRANSLATEDEXTPROC glad_glMatrixTranslatedEXT;
#define glMatrixTranslatedEXT glad_glMatrixTranslatedEXT
typedef void (APIENTRYP PFNGLMATRIXFRUSTUMEXTPROC)(GLenum mode, GLdouble left, GLdouble right, GLdouble bottom, GLdouble top, GLdouble zNear, GLdouble zFar);
GLAPI PFNGLMATRIXFRUSTUMEXTPROC glad_glMatrixFrustumEXT;
#define glMatrixFrustumEXT glad_glMatrixFrustumEXT
typedef void (APIENTRYP PFNGLMATRIXORTHOEXTPROC)(GLenum mode, GLdouble left, GLdouble right, GLdouble bottom, GLdouble top, GLdouble zNear, GLdouble zFar);
GLAPI PFNGLMATRIXORTHOEXTPROC glad_glMatrixOrthoEXT;
#define glMatrixOrthoEXT glad_glMatrixOrthoEXT
typedef void (APIENTRYP PFNGLMATRIXPOPEXTPROC)(GLenum mode);
GLAPI PFNGLMATRIXPOPEXTPROC glad_glMatrixPopEXT;
#define glMatrixPopEXT glad_glMatrixPopEXT
typedef void (APIENTRYP PFNGLMATRIXPUSHEXTPROC)(GLenum mode);
GLAPI PFNGLMATRIXPUSHEXTPROC glad_glMatrixPushEXT;
#define glMatrixPushEXT glad_glMatrixPushEXT
typedef void (APIENTRYP PFNGLCLIENTATTRIBDEFAULTEXTPROC)(GLbitfield mask);
GLAPI PFNGLCLIENTATTRIBDEFAULTEXTPROC glad_glClientAttribDefaultEXT;
#define glClientAttribDefaultEXT glad_glClientAttribDefaultEXT
typedef void (APIENTRYP PFNGLPUSHCLIENTATTRIBDEFAULTEXTPROC)(GLbitfield mask);
GLAPI PFNGLPUSHCLIENTATTRIBDEFAULTEXTPROC glad_glPushClientAttribDefaultEXT;
#define glPushClientAttribDefaultEXT glad_glPushClientAttribDefaultEXT
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERFEXTPROC)(GLuint texture, GLenum target, GLenum pname, GLfloat param);
GLAPI PFNGLTEXTUREPARAMETERFEXTPROC glad_glTextureParameterfEXT;
#define glTextureParameterfEXT glad_glTextureParameterfEXT
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERFVEXTPROC)(GLuint texture, GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLTEXTUREPARAMETERFVEXTPROC glad_glTextureParameterfvEXT;
#define glTextureParameterfvEXT glad_glTextureParameterfvEXT
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIEXTPROC)(GLuint texture, GLenum target, GLenum pname, GLint param);
GLAPI PFNGLTEXTUREPARAMETERIEXTPROC glad_glTextureParameteriEXT;
#define glTextureParameteriEXT glad_glTextureParameteriEXT
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIVEXTPROC)(GLuint texture, GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLTEXTUREPARAMETERIVEXTPROC glad_glTextureParameterivEXT;
#define glTextureParameterivEXT glad_glTextureParameterivEXT
typedef void (APIENTRYP PFNGLTEXTUREIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint internalformat, GLsizei width, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTUREIMAGE1DEXTPROC glad_glTextureImage1DEXT;
#define glTextureImage1DEXT glad_glTextureImage1DEXT
typedef void (APIENTRYP PFNGLTEXTUREIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint internalformat, GLsizei width, GLsizei height, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTUREIMAGE2DEXTPROC glad_glTextureImage2DEXT;
#define glTextureImage2DEXT glad_glTextureImage2DEXT
typedef void (APIENTRYP PFNGLTEXTURESUBIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTURESUBIMAGE1DEXTPROC glad_glTextureSubImage1DEXT;
#define glTextureSubImage1DEXT glad_glTextureSubImage1DEXT
typedef void (APIENTRYP PFNGLTEXTURESUBIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTURESUBIMAGE2DEXTPROC glad_glTextureSubImage2DEXT;
#define glTextureSubImage2DEXT glad_glTextureSubImage2DEXT
typedef void (APIENTRYP PFNGLCOPYTEXTUREIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum internalformat, GLint x, GLint y, GLsizei width, GLint border);
GLAPI PFNGLCOPYTEXTUREIMAGE1DEXTPROC glad_glCopyTextureImage1DEXT;
#define glCopyTextureImage1DEXT glad_glCopyTextureImage1DEXT
typedef void (APIENTRYP PFNGLCOPYTEXTUREIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
GLAPI PFNGLCOPYTEXTUREIMAGE2DEXTPROC glad_glCopyTextureImage2DEXT;
#define glCopyTextureImage2DEXT glad_glCopyTextureImage2DEXT
typedef void (APIENTRYP PFNGLCOPYTEXTURESUBIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint xoffset, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYTEXTURESUBIMAGE1DEXTPROC glad_glCopyTextureSubImage1DEXT;
#define glCopyTextureSubImage1DEXT glad_glCopyTextureSubImage1DEXT
typedef void (APIENTRYP PFNGLCOPYTEXTURESUBIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXTURESUBIMAGE2DEXTPROC glad_glCopyTextureSubImage2DEXT;
#define glCopyTextureSubImage2DEXT glad_glCopyTextureSubImage2DEXT
typedef void (APIENTRYP PFNGLGETTEXTUREIMAGEEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum format, GLenum type, void *pixels);
GLAPI PFNGLGETTEXTUREIMAGEEXTPROC glad_glGetTextureImageEXT;
```

```
#define glGetTextureImageEXT glad_glGetTextureImageEXT
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERFVEXTPROC)(GLuint texture, GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETTEXTUREPARAMETERFVEXTPROC glad_glGetTextureParameterfvEXT;
#define glGetTextureParameterfvEXT glad_glGetTextureParameterfvEXT
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERIVEXTPROC)(GLuint texture, GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXTUREPARAMETERIVEXTPROC glad_glGetTextureParameterivEXT;
#define glGetTextureParameterivEXT glad_glGetTextureParameterivEXT
typedef void (APIENTRYP PFNGLGETTEXTURELEVELPARAMETERFVEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum pname, GLfloat *params);
GLAPI PFNGLGETTEXTURELEVELPARAMETERFVEXTPROC glad_glGetTextureLevelParameterfvEXT;
#define glGetTextureLevelParameterfvEXT glad_glGetTextureLevelParameterfvEXT
typedef void (APIENTRYP PFNGLGETTEXTURELEVELPARAMETERIVEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXTURELEVELPARAMETERIVEXTPROC glad_glGetTextureLevelParameterivEXT;
#define glGetTextureLevelParameterivEXT glad_glGetTextureLevelParameterivEXT
typedef void (APIENTRYP PFNGLTEXTUREIMAGE3DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXTUREIMAGE3DEXTPROC glad_glTextureImage3DEXT;
#define glTextureImage3DEXT glad_glTextureImage3DEXT
typedef void (APIENTRYP 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);
GLAPI PFNGLTEXTURESUBIMAGE3DEXTPROC glad_glTextureSubImage3DEXT;
#define glTextureSubImage3DEXT glad_glTextureSubImage3DEXT
typedef void (APIENTRYP PFNGLCOPYTEXTURESUBIMAGE3DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYTEXTURESUBIMAGE3DEXTPROC glad_glCopyTextureSubImage3DEXT;
#define glCopyTextureSubImage3DEXT glad_glCopyTextureSubImage3DEXT
typedef void (APIENTRYP PFNGLBINDMULTITEXTUREEXTPROC)(GLenum texunit, GLenum target, GLuint texture);
GLAPI PFNGLBINDMULTITEXTUREEXTPROC glad_glBindMultiTextureEXT;
#define glBindMultiTextureEXT glad_glBindMultiTextureEXT
typedef void (APIENTRYP PFNGLMULTITEXCOORDPOINTEREXTPROC)(GLenum texunit, GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLMULTITEXCOORDPOINTEREXTPROC glad_glMultiTexCoordPointerEXT;
#define glMultiTexCoordPointerEXT glad_glMultiTexCoordPointerEXT
typedef void (APIENTRYP PFNGLMULTITEXENVFEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLfloat param);
GLAPI PFNGLMULTITEXENVFEXTPROC glad_glMultiTexEnvfEXT;
#define glMultiTexEnvfEXT glad_glMultiTexEnvfEXT
typedef void (APIENTRYP PFNGLMULTITEXENVFVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLMULTITEXENVFVEXTPROC glad_glMultiTexEnvfvEXT;
#define glMultiTexEnvfvEXT glad_glMultiTexEnvfvEXT
typedef void (APIENTRYP PFNGLMULTITEXENVIEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLint param);
GLAPI PFNGLMULTITEXENVIEXTPROC glad_glMultiTexEnviEXT;
#define glMultiTexEnviEXT glad_glMultiTexEnviEXT
typedef void (APIENTRYP PFNGLMULTITEXENVIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLMULTITEXENVIVEXTPROC glad_glMultiTexEnvivEXT;
#define glMultiTexEnvivEXT glad_glMultiTexEnvivEXT
typedef void (APIENTRYP PFNGLMULTITEXGENDEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, GLdouble param);
GLAPI PFNGLMULTITEXGENDEXTPROC glad_glMultiTexGendEXT;
#define glMultiTexGendEXT glad_glMultiTexGendEXT
typedef void (APIENTRYP PFNGLMULTITEXGENDVEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, const GLdouble *params);
GLAPI PFNGLMULTITEXGENDVEXTPROC glad_glMultiTexGendvEXT;
#define glMultiTexGendvEXT glad_glMultiTexGendvEXT
typedef void (APIENTRYP PFNGLMULTITEXGENFEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, GLfloat param);
GLAPI PFNGLMULTITEXGENFEXTPROC glad_glMultiTexGenfEXT;
#define glMultiTexGenfEXT glad_glMultiTexGenfEXT
typedef void (APIENTRYP PFNGLMULTITEXGENFVEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, const GLfloat *params);
GLAPI PFNGLMULTITEXGENFVEXTPROC glad_glMultiTexGenfvEXT;
#define glMultiTexGenfvEXT glad_glMultiTexGenfvEXT
typedef void (APIENTRYP PFNGLMULTITEXGENIEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, GLint param);
GLAPI PFNGLMULTITEXGENIEXTPROC glad_glMultiTexGeniEXT;
#define glMultiTexGeniEXT glad_glMultiTexGeniEXT
typedef void (APIENTRYP PFNGLMULTITEXGENIVEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, const GLint *params);
GLAPI PFNGLMULTITEXGENIVEXTPROC glad_glMultiTexGenivEXT;
#define glMultiTexGenivEXT glad_glMultiTexGenivEXT
typedef void (APIENTRYP PFNGLGETMULTITEXENVFVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMULTITEXENVFVEXTPROC glad_glGetMultiTexEnvfvEXT;
#define glGetMultiTexEnvfvEXT glad_glGetMultiTexEnvfvEXT
typedef void (APIENTRYP PFNGLGETMULTITEXENVIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETMULTITEXENVIVEXTPROC glad_glGetMultiTexEnvivEXT;
#define glGetMultiTexEnvivEXT glad_glGetMultiTexEnvivEXT
typedef void (APIENTRYP PFNGLGETMULTITEXGENDVEXTPROC)(GLenum texunit, GLenum coord, GLenum pname,
GLdouble *params);
GLAPI PFNGLGETMULTITEXGENDVEXTPROC glad_glGetMultiTexGendvEXT;
#define glGetMultiTexGendvEXT glad_glGetMultiTexGendvEXT
typedef void (APIENTRYP PFNGLGETMULTITEXGENFVEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMULTITEXGENFVEXTPROC glad_glGetMultiTexGenfvEXT;
#define glGetMultiTexGenfvEXT glad_glGetMultiTexGenfvEXT
typedef void (APIENTRYP PFNGLGETMULTITEXGENIVEXTPROC)(GLenum texunit, GLenum coord, GLenum pname, GLint *params);
GLAPI PFNGLGETMULTITEXGENIVEXTPROC glad_glGetMultiTexGenivEXT;
#define glGetMultiTexGenivEXT glad_glGetMultiTexGenivEXT
typedef void (APIENTRYP PFNGLMULTITEXPARAMETERIEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLint param);
GLAPI PFNGLMULTITEXPARAMETERIEXTPROC glad_glMultiTexParameteriEXT;
#define glMultiTexParameteriEXT glad_glMultiTexParameteriEXT
typedef void (APIENTRYP PFNGLMULTITEXPARAMETERIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLMULTITEXPARAMETERIVEXTPROC glad_glMultiTexParameterivEXT;
#define glMultiTexParameterivEXT glad_glMultiTexParameterivEXT
typedef void (APIENTRYP PFNGLMULTITEXPARAMETERFEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLfloat param);
GLAPI PFNGLMULTITEXPARAMETERFEXTPROC glad_glMultiTexParameterfEXT;
#define glMultiTexParameterfEXT glad_glMultiTexParameterfEXT
typedef void (APIENTRYP PFNGLMULTITEXPARAMETERFVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLMULTITEXPARAMETERFVEXTPROC glad_glMultiTexParameterfvEXT;
#define glMultiTexParameterfvEXT glad_glMultiTexParameterfvEXT
typedef void (APIENTRYP PFNGLMULTITEXIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLint internalformat, GLsizei width, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLMULTITEXIMAGE1DEXTPROC glad_glMultiTexImage1DEXT;
#define glMultiTexImage1DEXT glad_glMultiTexImage1DEXT
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);
GLAPI PFNGLMULTITEXIMAGE2DEXTPROC glad_glMultiTexImage2DEXT;
#define glMultiTexImage2DEXT glad_glMultiTexImage2DEXT
typedef void (APIENTRYP PFNGLMULTITEXSUBIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLMULTITEXSUBIMAGE1DEXTPROC glad_glMultiTexSubImage1DEXT;
#define glMultiTexSubImage1DEXT glad_glMultiTexSubImage1DEXT
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);
GLAPI PFNGLMULTITEXSUBIMAGE2DEXTPROC glad_glMultiTexSubImage2DEXT;
#define glMultiTexSubImage2DEXT glad_glMultiTexSubImage2DEXT
typedef void (APIENTRYP PFNGLCOPYMULTITEXIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum internalformat, GLint x, GLint y, GLsizei width, GLint border);
GLAPI PFNGLCOPYMULTITEXIMAGE1DEXTPROC glad_glCopyMultiTexImage1DEXT;
#define glCopyMultiTexImage1DEXT glad_glCopyMultiTexImage1DEXT
typedef void (APIENTRYP PFNGLCOPYMULTITEXIMAGE2DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum internalformat, GLint x, GLint y, GLsizei width, GLsizei height, GLint border);
GLAPI PFNGLCOPYMULTITEXIMAGE2DEXTPROC glad_glCopyMultiTexImage2DEXT;
#define glCopyMultiTexImage2DEXT glad_glCopyMultiTexImage2DEXT
typedef void (APIENTRYP PFNGLCOPYMULTITEXSUBIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLint xoffset, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYMULTITEXSUBIMAGE1DEXTPROC glad_glCopyMultiTexSubImage1DEXT;
#define glCopyMultiTexSubImage1DEXT glad_glCopyMultiTexSubImage1DEXT
typedef void (APIENTRYP PFNGLCOPYMULTITEXSUBIMAGE2DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYMULTITEXSUBIMAGE2DEXTPROC glad_glCopyMultiTexSubImage2DEXT;
#define glCopyMultiTexSubImage2DEXT glad_glCopyMultiTexSubImage2DEXT
typedef void (APIENTRYP PFNGLGETMULTITEXIMAGEEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum format, GLenum type, void *pixels);
GLAPI PFNGLGETMULTITEXIMAGEEXTPROC glad_glGetMultiTexImageEXT;
#define glGetMultiTexImageEXT glad_glGetMultiTexImageEXT
typedef void (APIENTRYP PFNGLGETMULTITEXPARAMETERFVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMULTITEXPARAMETERFVEXTPROC glad_glGetMultiTexParameterfvEXT;
#define glGetMultiTexParameterfvEXT glad_glGetMultiTexParameterfvEXT
typedef void (APIENTRYP PFNGLGETMULTITEXPARAMETERIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETMULTITEXPARAMETERIVEXTPROC glad_glGetMultiTexParameterivEXT;
#define glGetMultiTexParameterivEXT glad_glGetMultiTexParameterivEXT
typedef void (APIENTRYP PFNGLGETMULTITEXLEVELPARAMETERFVEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMULTITEXLEVELPARAMETERFVEXTPROC glad_glGetMultiTexLevelParameterfvEXT;
#define glGetMultiTexLevelParameterfvEXT glad_glGetMultiTexLevelParameterfvEXT
typedef void (APIENTRYP PFNGLGETMULTITEXLEVELPARAMETERIVEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum pname, GLint *params);
GLAPI PFNGLGETMULTITEXLEVELPARAMETERIVEXTPROC glad_glGetMultiTexLevelParameterivEXT;
#define glGetMultiTexLevelParameterivEXT glad_glGetMultiTexLevelParameterivEXT
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);
GLAPI PFNGLMULTITEXIMAGE3DEXTPROC glad_glMultiTexImage3DEXT;
#define glMultiTexImage3DEXT glad_glMultiTexImage3DEXT
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);
GLAPI PFNGLMULTITEXSUBIMAGE3DEXTPROC glad_glMultiTexSubImage3DEXT;
#define glMultiTexSubImage3DEXT glad_glMultiTexSubImage3DEXT
typedef void (APIENTRYP PFNGLCOPYMULTITEXSUBIMAGE3DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLCOPYMULTITEXSUBIMAGE3DEXTPROC glad_glCopyMultiTexSubImage3DEXT;
#define glCopyMultiTexSubImage3DEXT glad_glCopyMultiTexSubImage3DEXT
typedef void (APIENTRYP PFNGLENABLECLIENTSTATEINDEXEDEXTPROC)(GLenum array, GLuint index);
GLAPI PFNGLENABLECLIENTSTATEINDEXEDEXTPROC glad_glEnableClientStateIndexedEXT;
#define glEnableClientStateIndexedEXT glad_glEnableClientStateIndexedEXT
typedef void (APIENTRYP PFNGLDISABLECLIENTSTATEINDEXEDEXTPROC)(GLenum array, GLuint index);
GLAPI PFNGLDISABLECLIENTSTATEINDEXEDEXTPROC glad_glDisableClientStateIndexedEXT;
#define glDisableClientStateIndexedEXT glad_glDisableClientStateIndexedEXT
typedef void (APIENTRYP PFNGLGETFLOATINDEXEDVEXTPROC)(GLenum target, GLuint index, GLfloat *data);
GLAPI PFNGLGETFLOATINDEXEDVEXTPROC glad_glGetFloatIndexedvEXT;
#define glGetFloatIndexedvEXT glad_glGetFloatIndexedvEXT
typedef void (APIENTRYP PFNGLGETDOUBLEINDEXEDVEXTPROC)(GLenum target, GLuint index, GLdouble *data);
GLAPI PFNGLGETDOUBLEINDEXEDVEXTPROC glad_glGetDoubleIndexedvEXT;
#define glGetDoubleIndexedvEXT glad_glGetDoubleIndexedvEXT
typedef void (APIENTRYP PFNGLGETPOINTERINDEXEDVEXTPROC)(GLenum target, GLuint index, void **data);
GLAPI PFNGLGETPOINTERINDEXEDVEXTPROC glad_glGetPointerIndexedvEXT;
#define glGetPointerIndexedvEXT glad_glGetPointerIndexedvEXT
typedef void (APIENTRYP PFNGLENABLEINDEXEDEXTPROC)(GLenum target, GLuint index);
GLAPI PFNGLENABLEINDEXEDEXTPROC glad_glEnableIndexedEXT;
#define glEnableIndexedEXT glad_glEnableIndexedEXT
typedef void (APIENTRYP PFNGLDISABLEINDEXEDEXTPROC)(GLenum target, GLuint index);
GLAPI PFNGLDISABLEINDEXEDEXTPROC glad_glDisableIndexedEXT;
#define glDisableIndexedEXT glad_glDisableIndexedEXT
typedef GLboolean (APIENTRYP PFNGLISENABLEDINDEXEDEXTPROC)(GLenum target, GLuint index);
GLAPI PFNGLISENABLEDINDEXEDEXTPROC glad_glIsEnabledIndexedEXT;
#define glIsEnabledIndexedEXT glad_glIsEnabledIndexedEXT
typedef void (APIENTRYP PFNGLGETINTEGERINDEXEDVEXTPROC)(GLenum target, GLuint index, GLint *data);
GLAPI PFNGLGETINTEGERINDEXEDVEXTPROC glad_glGetIntegerIndexedvEXT;
#define glGetIntegerIndexedvEXT glad_glGetIntegerIndexedvEXT
typedef void (APIENTRYP PFNGLGETBOOLEANINDEXEDVEXTPROC)(GLenum target, GLuint index, GLboolean *data);
GLAPI PFNGLGETBOOLEANINDEXEDVEXTPROC glad_glGetBooleanIndexedvEXT;
#define glGetBooleanIndexedvEXT glad_glGetBooleanIndexedvEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTUREIMAGE3DEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDTEXTUREIMAGE3DEXTPROC glad_glCompressedTextureImage3DEXT;
#define glCompressedTextureImage3DEXT glad_glCompressedTextureImage3DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTUREIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDTEXTUREIMAGE2DEXTPROC glad_glCompressedTextureImage2DEXT;
#define glCompressedTextureImage2DEXT glad_glCompressedTextureImage2DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTUREIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint level, GLenum internalformat, GLsizei width, GLint border, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDTEXTUREIMAGE1DEXTPROC glad_glCompressedTextureImage1DEXT;
#define glCompressedTextureImage1DEXT glad_glCompressedTextureImage1DEXT
typedef void (APIENTRYP 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);
GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE3DEXTPROC glad_glCompressedTextureSubImage3DEXT;
#define glCompressedTextureSubImage3DEXT glad_glCompressedTextureSubImage3DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTURESUBIMAGE2DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE2DEXTPROC glad_glCompressedTextureSubImage2DEXT;
#define glCompressedTextureSubImage2DEXT glad_glCompressedTextureSubImage2DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDTEXTURESUBIMAGE1DEXTPROC)(GLuint texture, GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDTEXTURESUBIMAGE1DEXTPROC glad_glCompressedTextureSubImage1DEXT;
#define glCompressedTextureSubImage1DEXT glad_glCompressedTextureSubImage1DEXT
typedef void (APIENTRYP PFNGLGETCOMPRESSEDTEXTUREIMAGEEXTPROC)(GLuint texture, GLenum target, GLint lod, void *img);
GLAPI PFNGLGETCOMPRESSEDTEXTUREIMAGEEXTPROC glad_glGetCompressedTextureImageEXT;
#define glGetCompressedTextureImageEXT glad_glGetCompressedTextureImageEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDMULTITEXIMAGE3DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDMULTITEXIMAGE3DEXTPROC glad_glCompressedMultiTexImage3DEXT;
#define glCompressedMultiTexImage3DEXT glad_glCompressedMultiTexImage3DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDMULTITEXIMAGE2DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLint border, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDMULTITEXIMAGE2DEXTPROC glad_glCompressedMultiTexImage2DEXT;
#define glCompressedMultiTexImage2DEXT glad_glCompressedMultiTexImage2DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDMULTITEXIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLenum internalformat, GLsizei width, GLint border, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDMULTITEXIMAGE1DEXTPROC glad_glCompressedMultiTexImage1DEXT;
#define glCompressedMultiTexImage1DEXT glad_glCompressedMultiTexImage1DEXT
typedef void (APIENTRYP 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);
```

```
GLAPI PFNGLCOMPRESSEDMULTITEXSUBIMAGE3DEXTPROC glad_glCompressedMultiTexSubImage3DEXT;
#define glCompressedMultiTexSubImage3DEXT glad_glCompressedMultiTexSubImage3DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDMULTITEXSUBIMAGE2DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDMULTITEXSUBIMAGE2DEXTPROC glad_glCompressedMultiTexSubImage2DEXT;
#define glCompressedMultiTexSubImage2DEXT glad_glCompressedMultiTexSubImage2DEXT
typedef void (APIENTRYP PFNGLCOMPRESSEDMULTITEXSUBIMAGE1DEXTPROC)(GLenum texunit, GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLsizei imageSize, const void *bits);
GLAPI PFNGLCOMPRESSEDMULTITEXSUBIMAGE1DEXTPROC glad_glCompressedMultiTexSubImage1DEXT;
#define glCompressedMultiTexSubImage1DEXT glad_glCompressedMultiTexSubImage1DEXT
typedef void (APIENTRYP PFNGLGETCOMPRESSEDMULTITEXIMAGEEXTPROC)(GLenum texunit, GLenum target, GLint lod, void *img);
GLAPI PFNGLGETCOMPRESSEDMULTITEXIMAGEEXTPROC glad_glGetCompressedMultiTexImageEXT;
#define glGetCompressedMultiTexImageEXT glad_glGetCompressedMultiTexImageEXT
typedef void (APIENTRYP PFNGLMATRIXLOADTRANSPOSEFEXTPROC)(GLenum mode, const GLfloat *m);
GLAPI PFNGLMATRIXLOADTRANSPOSEFEXTPROC glad_glMatrixLoadTransposefEXT;
#define glMatrixLoadTransposefEXT glad_glMatrixLoadTransposefEXT
typedef void (APIENTRYP PFNGLMATRIXLOADTRANSPOSEDEXTPROC)(GLenum mode, const GLdouble *m);
GLAPI PFNGLMATRIXLOADTRANSPOSEDEXTPROC glad_glMatrixLoadTransposedEXT;
#define glMatrixLoadTransposedEXT glad_glMatrixLoadTransposedEXT
typedef void (APIENTRYP PFNGLMATRIXMULTTRANSPOSEFEXTPROC)(GLenum mode, const GLfloat *m);
GLAPI PFNGLMATRIXMULTTRANSPOSEFEXTPROC glad_glMatrixMultTransposefEXT;
#define glMatrixMultTransposefEXT glad_glMatrixMultTransposefEXT
typedef void (APIENTRYP PFNGLMATRIXMULTTRANSPOSEDEXTPROC)(GLenum mode, const GLdouble *m);
GLAPI PFNGLMATRIXMULTTRANSPOSEDEXTPROC glad_glMatrixMultTransposedEXT;
#define glMatrixMultTransposedEXT glad_glMatrixMultTransposedEXT
typedef void (APIENTRYP PFNGLNAMEDBUFFERDATAEXTPROC)(GLuint buffer, GLsizeiptr size, const void *data, GLenum usage);
GLAPI PFNGLNAMEDBUFFERDATAEXTPROC glad_glNamedBufferDataEXT;
#define glNamedBufferDataEXT glad_glNamedBufferDataEXT
typedef void (APIENTRYP PFNGLNAMEDBUFFERSUBDATAEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, const void *data);
GLAPI PFNGLNAMEDBUFFERSUBDATAEXTPROC glad_glNamedBufferSubDataEXT;
#define glNamedBufferSubDataEXT glad_glNamedBufferSubDataEXT
typedef void * (APIENTRYP PFNGLMAPNAMEDBUFFEREXTPROC)(GLuint buffer, GLenum access);
GLAPI PFNGLMAPNAMEDBUFFEREXTPROC glad_glMapNamedBufferEXT;
#define glMapNamedBufferEXT glad_glMapNamedBufferEXT
typedef GLboolean (APIENTRYP PFNGLUNMAPNAMEDBUFFEREXTPROC)(GLuint buffer);
GLAPI PFNGLUNMAPNAMEDBUFFEREXTPROC glad_glUnmapNamedBufferEXT;
#define glUnmapNamedBufferEXT glad_glUnmapNamedBufferEXT
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERPARAMETERIVEXTPROC)(GLuint buffer, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDBUFFERPARAMETERIVEXTPROC glad_glGetNamedBufferParameterivEXT;
#define glGetNamedBufferParameterivEXT glad_glGetNamedBufferParameterivEXT
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERPOINTERVEXTPROC)(GLuint buffer, GLenum pname, void **params);
GLAPI PFNGLGETNAMEDBUFFERPOINTERVEXTPROC glad_glGetNamedBufferPointervEXT;
#define glGetNamedBufferPointervEXT glad_glGetNamedBufferPointervEXT
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERSUBDATAEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, void *data);
GLAPI PFNGLGETNAMEDBUFFERSUBDATAEXTPROC glad_glGetNamedBufferSubDataEXT;
#define glGetNamedBufferSubDataEXT glad_glGetNamedBufferSubDataEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1FEXTPROC)(GLuint program, GLint location, GLfloat v0);
GLAPI PFNGLPROGRAMUNIFORM1FEXTPROC glad_glProgramUniform1fEXT;
#define glProgramUniform1fEXT glad_glProgramUniform1fEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2FEXTPROC)(GLuint program, GLint location, GLfloat v0, GLfloat v1);
GLAPI PFNGLPROGRAMUNIFORM2FEXTPROC glad_glProgramUniform2fEXT;
#define glProgramUniform2fEXT glad_glProgramUniform2fEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3FEXTPROC)(GLuint program, GLint location, GLfloat v0, GLfloat v1, GLfloat v2);
GLAPI PFNGLPROGRAMUNIFORM3FEXTPROC glad_glProgramUniform3fEXT;
#define glProgramUniform3fEXT glad_glProgramUniform3fEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4FEXTPROC)(GLuint program, GLint location, GLfloat v0, GLfloat v1, GLfloat v2, GLfloat v3);
GLAPI PFNGLPROGRAMUNIFORM4FEXTPROC glad_glProgramUniform4fEXT;
#define glProgramUniform4fEXT glad_glProgramUniform4fEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1IEXTPROC)(GLuint program, GLint location, GLint v0);
GLAPI PFNGLPROGRAMUNIFORM1IEXTPROC glad_glProgramUniform1iEXT;
#define glProgramUniform1iEXT glad_glProgramUniform1iEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2IEXTPROC)(GLuint program, GLint location, GLint v0, GLint v1);
GLAPI PFNGLPROGRAMUNIFORM2IEXTPROC glad_glProgramUniform2iEXT;
#define glProgramUniform2iEXT glad_glProgramUniform2iEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3IEXTPROC)(GLuint program, GLint location, GLint v0, GLint v1, GLint v2);
GLAPI PFNGLPROGRAMUNIFORM3IEXTPROC glad_glProgramUniform3iEXT;
#define glProgramUniform3iEXT glad_glProgramUniform3iEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4IEXTPROC)(GLuint program, GLint location, GLint v0, GLint v1, GLint v2, GLint v3);
GLAPI PFNGLPROGRAMUNIFORM4IEXTPROC glad_glProgramUniform4iEXT;
#define glProgramUniform4iEXT glad_glProgramUniform4iEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1FVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM1FVEXTPROC glad_glProgramUniform1fvEXT;
#define glProgramUniform1fvEXT glad_glProgramUniform1fvEXT
```

```
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2FVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM2FVEXTPROC glad_glProgramUniform2fvEXT;
#define glProgramUniform2fvEXT glad_glProgramUniform2fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3FVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM3FVEXTPROC glad_glProgramUniform3fvEXT;
#define glProgramUniform3fvEXT glad_glProgramUniform3fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4FVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORM4FVEXTPROC glad_glProgramUniform4fvEXT;
#define glProgramUniform4fvEXT glad_glProgramUniform4fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1IVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM1IVEXTPROC glad_glProgramUniform1ivEXT;
#define glProgramUniform1ivEXT glad_glProgramUniform1ivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2IVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM2IVEXTPROC glad_glProgramUniform2ivEXT;
#define glProgramUniform2ivEXT glad_glProgramUniform2ivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3IVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM3IVEXTPROC glad_glProgramUniform3ivEXT;
#define glProgramUniform3ivEXT glad_glProgramUniform3ivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4IVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLint *value);
GLAPI PFNGLPROGRAMUNIFORM4IVEXTPROC glad_glProgramUniform4ivEXT;
#define glProgramUniform4ivEXT glad_glProgramUniform4ivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2FVEXTPROC glad_glProgramUniformMatrix2fvEXT;
#define glProgramUniformMatrix2fvEXT glad_glProgramUniformMatrix2fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3FVEXTPROC glad_glProgramUniformMatrix3fvEXT;
#define glProgramUniformMatrix3fvEXT glad_glProgramUniformMatrix3fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4FVEXTPROC glad_glProgramUniformMatrix4fvEXT;
#define glProgramUniformMatrix4fvEXT glad_glProgramUniformMatrix4fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X3FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3FVEXTPROC glad_glProgramUniformMatrix2x3fvEXT;
#define glProgramUniformMatrix2x3fvEXT glad_glProgramUniformMatrix2x3fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X2FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2FVEXTPROC glad_glProgramUniformMatrix3x2fvEXT;
#define glProgramUniformMatrix3x2fvEXT glad_glProgramUniformMatrix3x2fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X4FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4FVEXTPROC glad_glProgramUniformMatrix2x4fvEXT;
#define glProgramUniformMatrix2x4fvEXT glad_glProgramUniformMatrix2x4fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X2FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2FVEXTPROC glad_glProgramUniformMatrix4x2fvEXT;
#define glProgramUniformMatrix4x2fvEXT glad_glProgramUniformMatrix4x2fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X4FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4FVEXTPROC glad_glProgramUniformMatrix3x4fvEXT;
#define glProgramUniformMatrix3x4fvEXT glad_glProgramUniformMatrix3x4fvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X3FVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLfloat *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3FVEXTPROC glad_glProgramUniformMatrix4x3fvEXT;
#define glProgramUniformMatrix4x3fvEXT glad_glProgramUniformMatrix4x3fvEXT
typedef void (APIENTRYP PFNGLTEXTUREBUFFEREXTPROC)(GLuint texture, GLenum target, GLenum internalformat, GLuint buffer);
GLAPI PFNGLTEXTUREBUFFEREXTPROC glad_glTextureBufferEXT;
#define glTextureBufferEXT glad_glTextureBufferEXT
typedef void (APIENTRYP PFNGLMULTITEXBUFFEREXTPROC)(GLenum texunit, GLenum target, GLenum internalformat, GLuint buffer);
GLAPI PFNGLMULTITEXBUFFEREXTPROC glad_glMultiTexBufferEXT;
#define glMultiTexBufferEXT glad_glMultiTexBufferEXT
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIIVEXTPROC)(GLuint texture, GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLTEXTUREPARAMETERIIVEXTPROC glad_glTextureParameterIivEXT;
#define glTextureParameterIivEXT glad_glTextureParameterIivEXT
typedef void (APIENTRYP PFNGLTEXTUREPARAMETERIUIVEXTPROC)(GLuint texture, GLenum target, GLenum pname, const GLuint *params);
GLAPI PFNGLTEXTUREPARAMETERIUIVEXTPROC glad_glTextureParameterIuivEXT;
#define glTextureParameterIuivEXT glad_glTextureParameterIuivEXT
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERIIVEXTPROC)(GLuint texture, GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXTUREPARAMETERIIVEXTPROC glad_glGetTextureParameterIivEXT;
#define glGetTextureParameterIivEXT glad_glGetTextureParameterIivEXT
typedef void (APIENTRYP PFNGLGETTEXTUREPARAMETERIUIVEXTPROC)(GLuint texture, GLenum target, GLenum pname, GLuint *params);
GLAPI PFNGLGETTEXTUREPARAMETERIUIVEXTPROC glad_glGetTextureParameterIuivEXT;
```

```
#define glGetTextureParameterIuivEXT glad_glGetTextureParameterIuivEXT
typedef void (APIENTRYP PFNGLMULTITEXPARAMETERIIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLMULTITEXPARAMETERIIVEXTPROC glad_glMultiTexParameterIivEXT;
#define glMultiTexParameterIivEXT glad_glMultiTexParameterIivEXT
typedef void (APIENTRYP PFNGLMULTITEXPARAMETERIUIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, const GLuint *params);
GLAPI PFNGLMULTITEXPARAMETERIUIVEXTPROC glad_glMultiTexParameterIuivEXT;
#define glMultiTexParameterIuivEXT glad_glMultiTexParameterIuivEXT
typedef void (APIENTRYP PFNGLGETMULTITEXPARAMETERIIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETMULTITEXPARAMETERIIVEXTPROC glad_glGetMultiTexParameterIivEXT;
#define glGetMultiTexParameterIivEXT glad_glGetMultiTexParameterIivEXT
typedef void (APIENTRYP PFNGLGETMULTITEXPARAMETERIUIVEXTPROC)(GLenum texunit, GLenum target, GLenum pname, GLuint *params);
GLAPI PFNGLGETMULTITEXPARAMETERIUIVEXTPROC glad_glGetMultiTexParameterIuivEXT;
#define glGetMultiTexParameterIuivEXT glad_glGetMultiTexParameterIuivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UIEXTPROC)(GLuint program, GLint location, GLuint v0);
GLAPI PFNGLPROGRAMUNIFORM1UIEXTPROC glad_glProgramUniform1uiEXT;
#define glProgramUniform1uiEXT glad_glProgramUniform1uiEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UIEXTPROC)(GLuint program, GLint location, GLuint v0, GLuint v1);
GLAPI PFNGLPROGRAMUNIFORM2UIEXTPROC glad_glProgramUniform2uiEXT;
#define glProgramUniform2uiEXT glad_glProgramUniform2uiEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UIEXTPROC)(GLuint program, GLint location, GLuint v0, GLuint v1, GLuint v2);
GLAPI PFNGLPROGRAMUNIFORM3UIEXTPROC glad_glProgramUniform3uiEXT;
#define glProgramUniform3uiEXT glad_glProgramUniform3uiEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UIEXTPROC)(GLuint program, GLint location, GLuint v0, GLuint v1, GLuint v2, GLuint v3);
GLAPI PFNGLPROGRAMUNIFORM4UIEXTPROC glad_glProgramUniform4uiEXT;
#define glProgramUniform4uiEXT glad_glProgramUniform4uiEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1UIVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM1UIVEXTPROC glad_glProgramUniform1uivEXT;
#define glProgramUniform1uivEXT glad_glProgramUniform1uivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2UIVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM2UIVEXTPROC glad_glProgramUniform2uivEXT;
#define glProgramUniform2uivEXT glad_glProgramUniform2uivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3UIVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM3UIVEXTPROC glad_glProgramUniform3uivEXT;
#define glProgramUniform3uivEXT glad_glProgramUniform3uivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4UIVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLPROGRAMUNIFORM4UIVEXTPROC glad_glProgramUniform4uivEXT;
#define glProgramUniform4uivEXT glad_glProgramUniform4uivEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETERS4FVEXTPROC)(GLuint program, GLenum target, GLuint index, GLsizei count, const GLfloat *params);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERS4FVEXTPROC glad_glNamedProgramLocalParameters4fvEXT;
#define glNamedProgramLocalParameters4fvEXT glad_glNamedProgramLocalParameters4fvEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETERI4IEXTPROC)(GLuint program, GLenum target, GLuint index, GLint x, GLint y, GLint z, GLint w);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERI4IEXTPROC glad_glNamedProgramLocalParameterI4iEXT;
#define glNamedProgramLocalParameterI4iEXT glad_glNamedProgramLocalParameterI4iEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETERI4IVEXTPROC)(GLuint program, GLenum target, GLuint index, const GLint *params);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERI4IVEXTPROC glad_glNamedProgramLocalParameterI4ivEXT;
#define glNamedProgramLocalParameterI4ivEXT glad_glNamedProgramLocalParameterI4ivEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETERSI4IVEXTPROC)(GLuint program, GLenum target, GLuint index, GLsizei count, const GLint *params);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERSI4IVEXTPROC glad_glNamedProgramLocalParametersI4ivEXT;
#define glNamedProgramLocalParametersI4ivEXT glad_glNamedProgramLocalParametersI4ivEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETERI4UIEXTPROC)(GLuint program, GLenum target, GLuint index, GLuint x, GLuint y, GLuint z, GLuint w);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERI4UIEXTPROC glad_glNamedProgramLocalParameterI4uiEXT;
#define glNamedProgramLocalParameterI4uiEXT glad_glNamedProgramLocalParameterI4uiEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETERI4UIVEXTPROC)(GLuint program, GLenum target, GLuint index, const GLuint *params);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERI4UIVEXTPROC glad_glNamedProgramLocalParameterI4uivEXT;
#define glNamedProgramLocalParameterI4uivEXT glad_glNamedProgramLocalParameterI4uivEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETERSI4UIVEXTPROC)(GLuint program, GLenum target, GLuint index, GLsizei count, const GLuint *params);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETERSI4UIVEXTPROC glad_glNamedProgramLocalParametersI4uivEXT;
#define glNamedProgramLocalParametersI4uivEXT glad_glNamedProgramLocalParametersI4uivEXT
typedef void (APIENTRYP PFNGLGETNAMEDPROGRAMLOCALPARAMETERIIVEXTPROC)(GLuint program, GLenum target, GLuint index, GLint *params);
GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERIIVEXTPROC glad_glGetNamedProgramLocalParameterIivEXT;
#define glGetNamedProgramLocalParameterIivEXT glad_glGetNamedProgramLocalParameterIivEXT
typedef void (APIENTRYP PFNGLGETNAMEDPROGRAMLOCALPARAMETERIUIVEXTPROC)(GLuint program, GLenum target, GLuint index, GLuint *params);
GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERIUIVEXTPROC glad_glGetNamedProgramLocalParameterIuivEXT;
#define glGetNamedProgramLocalParameterIuivEXT glad_glGetNamedProgramLocalParameterIuivEXT
typedef void (APIENTRYP PFNGLENABLECLIENTSTATEIEXTPROC)(GLenum array, GLuint index);
GLAPI PFNGLENABLECLIENTSTATEIEXTPROC glad_glEnableClientStateiEXT;
#define glEnableClientStateiEXT glad_glEnableClientStateiEXT
```

```
typedef void (APIENTRYP PFNGLDISABLECLIENTSTATEIEXTPROC)(GLenum array, GLuint index);
GLAPI PFNGLDISABLECLIENTSTATEIEXTPROC glad_glDisableClientStateiEXT;
#define glDisableClientStateiEXT glad_glDisableClientStateiEXT
typedef void (APIENTRYP PFNGLGETFLOATI_VEXTPROC)(GLenum pname, GLuint index, GLfloat *params);
GLAPI PFNGLGETFLOATI_VEXTPROC glad_glGetFloati_vEXT;
#define glGetFloati_vEXT glad_glGetFloati_vEXT
typedef void (APIENTRYP PFNGLGETDOUBLEI_VEXTPROC)(GLenum pname, GLuint index, GLdouble *params);
GLAPI PFNGLGETDOUBLEI_VEXTPROC glad_glGetDoublei_vEXT;
#define glGetDoublei_vEXT glad_glGetDoublei_vEXT
typedef void (APIENTRYP PFNGLGETPOINTERI_VEXTPROC)(GLenum pname, GLuint index, void **params);
GLAPI PFNGLGETPOINTERI_VEXTPROC glad_glGetPointeri_vEXT;
#define glGetPointeri_vEXT glad_glGetPointeri_vEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMSTRINGEXTPROC)(GLuint program, GLenum target, GLenum format, GLsizei len, const void *string);
GLAPI PFNGLNAMEDPROGRAMSTRINGEXTPROC glad_glNamedProgramStringEXT;
#define glNamedProgramStringEXT glad_glNamedProgramStringEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETER4DEXTPROC)(GLuint program, GLenum target, GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4DEXTPROC glad_glNamedProgramLocalParameter4dEXT;
#define glNamedProgramLocalParameter4dEXT glad_glNamedProgramLocalParameter4dEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETER4DVEXTPROC)(GLuint program, GLenum target, GLuint index, const GLdouble *params);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4DVEXTPROC glad_glNamedProgramLocalParameter4dvEXT;
#define glNamedProgramLocalParameter4dvEXT glad_glNamedProgramLocalParameter4dvEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETER4FEXTPROC)(GLuint program, GLenum target, GLuint index, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4FEXTPROC glad_glNamedProgramLocalParameter4fEXT;
#define glNamedProgramLocalParameter4fEXT glad_glNamedProgramLocalParameter4fEXT
typedef void (APIENTRYP PFNGLNAMEDPROGRAMLOCALPARAMETER4FVEXTPROC)(GLuint program, GLenum target, GLuint index, const GLfloat *params);
GLAPI PFNGLNAMEDPROGRAMLOCALPARAMETER4FVEXTPROC glad_glNamedProgramLocalParameter4fvEXT;
#define glNamedProgramLocalParameter4fvEXT glad_glNamedProgramLocalParameter4fvEXT
typedef void (APIENTRYP PFNGLGETNAMEDPROGRAMLOCALPARAMETERDVEXTPROC)(GLuint program, GLenum target, GLuint index, GLdouble *params);
GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERDVEXTPROC glad_glGetNamedProgramLocalParameterdvEXT;
#define glGetNamedProgramLocalParameterdvEXT glad_glGetNamedProgramLocalParameterdvEXT
typedef void (APIENTRYP PFNGLGETNAMEDPROGRAMLOCALPARAMETERFVEXTPROC)(GLuint program, GLenum target, GLuint index, GLfloat *params);
GLAPI PFNGLGETNAMEDPROGRAMLOCALPARAMETERFVEXTPROC glad_glGetNamedProgramLocalParameterfvEXT;
#define glGetNamedProgramLocalParameterfvEXT glad_glGetNamedProgramLocalParameterfvEXT
typedef void (APIENTRYP PFNGLGETNAMEDPROGRAMIVEXTPROC)(GLuint program, GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDPROGRAMIVEXTPROC glad_glGetNamedProgramivEXT;
#define glGetNamedProgramivEXT glad_glGetNamedProgramivEXT
typedef void (APIENTRYP PFNGLGETNAMEDPROGRAMSTRINGEXTPROC)(GLuint program, GLenum target, GLenum pname, void *string);
GLAPI PFNGLGETNAMEDPROGRAMSTRINGEXTPROC glad_glGetNamedProgramStringEXT;
#define glGetNamedProgramStringEXT glad_glGetNamedProgramStringEXT
typedef void (APIENTRYP PFNGLNAMEDRENDERBUFFERSTORAGEEXTPROC)(GLuint renderbuffer, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEEXTPROC glad_glNamedRenderbufferStorageEXT;
#define glNamedRenderbufferStorageEXT glad_glNamedRenderbufferStorageEXT
typedef void (APIENTRYP PFNGLGETNAMEDRENDERBUFFERPARAMETERIVEXTPROC)(GLuint renderbuffer, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDRENDERBUFFERPARAMETERIVEXTPROC glad_glGetNamedRenderbufferParameterivEXT;
#define glGetNamedRenderbufferParameterivEXT glad_glGetNamedRenderbufferParameterivEXT
typedef void (APIENTRYP PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEEXTPROC)(GLuint renderbuffer, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLEEXTPROC glad_glNamedRenderbufferStorageMultisampleEXT;
#define glNamedRenderbufferStorageMultisampleEXT glad_glNamedRenderbufferStorageMultisampleEXT
typedef void (APIENTRYP PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLECOVERAGEEXTPROC)(GLuint renderbuffer, GLsizei coverageSamples, GLsizei colorSamples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLNAMEDRENDERBUFFERSTORAGEMULTISAMPLECOVERAGEEXTPROC glad_glNamedRenderbufferStorageMultisampleCoverageEXT;
#define glNamedRenderbufferStorageMultisampleCoverageEXT glad_glNamedRenderbufferStorageMultisampleCoverageEXT
typedef GLenum (APIENTRYP PFNGLCHECKNAMEDFRAMEBUFFERSTATUSEXTPROC)(GLuint framebuffer, GLenum target);
GLAPI PFNGLCHECKNAMEDFRAMEBUFFERSTATUSEXTPROC glad_glCheckNamedFramebufferStatusEXT;
#define glCheckNamedFramebufferStatusEXT glad_glCheckNamedFramebufferStatusEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTURE1DEXTPROC)(GLuint framebuffer, GLenum attachment, GLenum textarget, GLuint texture, GLint level);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURE1DEXTPROC glad_glNamedFramebufferTexture1DEXT;
#define glNamedFramebufferTexture1DEXT glad_glNamedFramebufferTexture1DEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTURE2DEXTPROC)(GLuint framebuffer, GLenum attachment, GLenum textarget, GLuint texture, GLint level);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURE2DEXTPROC glad_glNamedFramebufferTexture2DEXT;
#define glNamedFramebufferTexture2DEXT glad_glNamedFramebufferTexture2DEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTURE3DEXTPROC)(GLuint framebuffer, GLenum attachment, GLenum textarget, GLuint texture, GLint level, GLint zoffset);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURE3DEXTPROC glad_glNamedFramebufferTexture3DEXT;
#define glNamedFramebufferTexture3DEXT glad_glNamedFramebufferTexture3DEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERRENDERBUFFEREXTPROC)(GLuint framebuffer, GLenum attachment, GLenum renderbuffertarget, GLuint renderbuffer);
GLAPI PFNGLNAMEDFRAMEBUFFERRENDERBUFFEREXTPROC glad_glNamedFramebufferRenderbufferEXT;
#define glNamedFramebufferRenderbufferEXT glad_glNamedFramebufferRenderbufferEXT
typedef void (APIENTRYP PFNGLGETNAMEDFRAMEBUFFERATTACHMENTPARAMETERIVEXTPROC)(GLuint framebuffer, GLenum attachment, GLenum pname, GLint *params);
```

```
GLAPI PFNGLGETNAMEDFRAMEBUFFERATTACHMENTPARAMETERIVEXTPROC glad_glGetNamedFramebufferAttachmentParameterivEXT;
#define glGetNamedFramebufferAttachmentParameterivEXT glad_glGetNamedFramebufferAttachmentParameterivEXT
typedef void (APIENTRYP PFNGLGENERATETEXTUREMIPMAPEXTPROC)(GLuint texture, GLenum target);
GLAPI PFNGLGENERATETEXTUREMIPMAPEXTPROC glad_glGenerateTextureMipmapEXT;
#define glGenerateTextureMipmapEXT glad_glGenerateTextureMipmapEXT
typedef void (APIENTRYP PFNGLGENERATEMULTITEXMIPMAPEXTPROC)(GLenum texunit, GLenum target);
GLAPI PFNGLGENERATEMULTITEXMIPMAPEXTPROC glad_glGenerateMultiTexMipmapEXT;
#define glGenerateMultiTexMipmapEXT glad_glGenerateMultiTexMipmapEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERDRAWBUFFEREXTPROC)(GLuint framebuffer, GLenum mode);
GLAPI PFNGLFRAMEBUFFERDRAWBUFFEREXTPROC glad_glFramebufferDrawBufferEXT;
#define glFramebufferDrawBufferEXT glad_glFramebufferDrawBufferEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERDRAWBUFFERSEXTPROC)(GLuint framebuffer, GLsizei n, const GLenum *bufs);
GLAPI PFNGLFRAMEBUFFERDRAWBUFFERSEXTPROC glad_glFramebufferDrawBuffersEXT;
#define glFramebufferDrawBuffersEXT glad_glFramebufferDrawBuffersEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERREADBUFFEREXTPROC)(GLuint framebuffer, GLenum mode);
GLAPI PFNGLFRAMEBUFFERREADBUFFEREXTPROC glad_glFramebufferReadBufferEXT;
#define glFramebufferReadBufferEXT glad_glFramebufferReadBufferEXT
typedef void (APIENTRYP PFNGLGETFRAMEBUFFERPARAMETERIVEXTPROC)(GLuint framebuffer, GLenum pname, GLint *params);
GLAPI PFNGLGETFRAMEBUFFERPARAMETERIVEXTPROC glad_glGetFramebufferParameterivEXT;
#define glGetFramebufferParameterivEXT glad_glGetFramebufferParameterivEXT
typedef void (APIENTRYP PFNGLNAMEDCOPYBUFFERSUBDATAEXTPROC)(GLuint readBuffer, GLuint writeBuffer, GLintptr readOffset, GLintptr writeOffset, GLsizeiptr size);
GLAPI PFNGLNAMEDCOPYBUFFERSUBDATAEXTPROC glad_glNamedCopyBufferSubDataEXT;
#define glNamedCopyBufferSubDataEXT glad_glNamedCopyBufferSubDataEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTUREEXTPROC)(GLuint framebuffer, GLenum attachment, GLuint texture, GLint level);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREEXTPROC glad_glNamedFramebufferTextureEXT;
#define glNamedFramebufferTextureEXT glad_glNamedFramebufferTextureEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTURELAYEREXTPROC)(GLuint framebuffer, GLenum attachment, GLuint texture, GLint level, GLint layer);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTURELAYEREXTPROC glad_glNamedFramebufferTextureLayerEXT;
#define glNamedFramebufferTextureLayerEXT glad_glNamedFramebufferTextureLayerEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTUREFACEEXTPROC)(GLuint framebuffer, GLenum attachment, GLuint texture, GLint level, GLenum face);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREFACEEXTPROC glad_glNamedFramebufferTextureFaceEXT;
#define glNamedFramebufferTextureFaceEXT glad_glNamedFramebufferTextureFaceEXT
typedef void (APIENTRYP PFNGLTEXTURERENDERBUFFEREXTPROC)(GLuint texture, GLenum target, GLuint renderbuffer);
GLAPI PFNGLTEXTURERENDERBUFFEREXTPROC glad_glTextureRenderbufferEXT;
#define glTextureRenderbufferEXT glad_glTextureRenderbufferEXT
typedef void (APIENTRYP PFNGLMULTITEXRENDERBUFFEREXTPROC)(GLenum texunit, GLenum target, GLuint renderbuffer);
GLAPI PFNGLMULTITEXRENDERBUFFEREXTPROC glad_glMultiTexRenderbufferEXT;
#define glMultiTexRenderbufferEXT glad_glMultiTexRenderbufferEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLint size, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYVERTEXOFFSETEXTPROC glad_glVertexArrayVertexOffsetEXT;
#define glVertexArrayVertexOffsetEXT glad_glVertexArrayVertexOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYCOLOROFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLint size, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYCOLOROFFSETEXTPROC glad_glVertexArrayColorOffsetEXT;
#define glVertexArrayColorOffsetEXT glad_glVertexArrayColorOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYEDGEFLAGOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYEDGEFLAGOFFSETEXTPROC glad_glVertexArrayEdgeFlagOffsetEXT;
#define glVertexArrayEdgeFlagOffsetEXT glad_glVertexArrayEdgeFlagOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYINDEXOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYINDEXOFFSETEXTPROC glad_glVertexArrayIndexOffsetEXT;
#define glVertexArrayIndexOffsetEXT glad_glVertexArrayIndexOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYNORMALOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYNORMALOFFSETEXTPROC glad_glVertexArrayNormalOffsetEXT;
#define glVertexArrayNormalOffsetEXT glad_glVertexArrayNormalOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYTEXCOORDOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLint size, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYTEXCOORDOFFSETEXTPROC glad_glVertexArrayTexCoordOffsetEXT;
#define glVertexArrayTexCoordOffsetEXT glad_glVertexArrayTexCoordOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYMULTITEXCOORDOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLenum texunit, GLint size, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYMULTITEXCOORDOFFSETEXTPROC glad_glVertexArrayMultiTexCoordOffsetEXT;
#define glVertexArrayMultiTexCoordOffsetEXT glad_glVertexArrayMultiTexCoordOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYFOGCOORDOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYFOGCOORDOFFSETEXTPROC glad_glVertexArrayFogCoordOffsetEXT;
#define glVertexArrayFogCoordOffsetEXT glad_glVertexArrayFogCoordOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYSECONDARYCOLOROFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLint size, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYSECONDARYCOLOROFFSETEXTPROC glad_glVertexArraySecondaryColorOffsetEXT;
#define glVertexArraySecondaryColorOffsetEXT glad_glVertexArraySecondaryColorOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLuint index, GLint size, GLenum type, GLboolean normalized, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBOFFSETEXTPROC glad_glVertexArrayVertexAttribOffsetEXT;
```

```
#define glVertexArrayVertexAttribOffsetEXT glad_glVertexArrayVertexAttribOffsetEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBIOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLuint index, GLint size, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBIOFFSETEXTPROC glad_glVertexArrayVertexAttribIOffsetEXT;
#define glVertexArrayVertexAttribIOffsetEXT glad_glVertexArrayVertexAttribIOffsetEXT
typedef void (APIENTRYP PFNGLENABLEVERTEXARRAYEXTPROC)(GLuint vaobj, GLenum array);
GLAPI PFNGLENABLEVERTEXARRAYEXTPROC glad_glEnableVertexArrayEXT;
#define glEnableVertexArrayEXT glad_glEnableVertexArrayEXT
typedef void (APIENTRYP PFNGLDISABLEVERTEXARRAYEXTPROC)(GLuint vaobj, GLenum array);
GLAPI PFNGLDISABLEVERTEXARRAYEXTPROC glad_glDisableVertexArrayEXT;
#define glDisableVertexArrayEXT glad_glDisableVertexArrayEXT
typedef void (APIENTRYP PFNGLENABLEVERTEXARRAYATTRIBEXTPROC)(GLuint vaobj, GLuint index);
GLAPI PFNGLENABLEVERTEXARRAYATTRIBEXTPROC glad_glEnableVertexArrayAttribEXT;
#define glEnableVertexArrayAttribEXT glad_glEnableVertexArrayAttribEXT
typedef void (APIENTRYP PFNGLDISABLEVERTEXARRAYATTRIBEXTPROC)(GLuint vaobj, GLuint index);
GLAPI PFNGLDISABLEVERTEXARRAYATTRIBEXTPROC glad_glDisableVertexArrayAttribEXT;
#define glDisableVertexArrayAttribEXT glad_glDisableVertexArrayAttribEXT
typedef void (APIENTRYP PFNGLGETVERTEXARRAYINTEGERVEXTPROC)(GLuint vaobj, GLenum pname, GLint *param);
GLAPI PFNGLGETVERTEXARRAYINTEGERVEXTPROC glad_glGetVertexArrayIntegervEXT;
#define glGetVertexArrayIntegervEXT glad_glGetVertexArrayIntegervEXT
typedef void (APIENTRYP PFNGLGETVERTEXARRAYPOINTERVEXTPROC)(GLuint vaobj, GLenum pname, void **param);
GLAPI PFNGLGETVERTEXARRAYPOINTERVEXTPROC glad_glGetVertexArrayPointervEXT;
#define glGetVertexArrayPointervEXT glad_glGetVertexArrayPointervEXT
typedef void (APIENTRYP PFNGLGETVERTEXARRAYINTEGERI_VEXTPROC)(GLuint vaobj, GLuint index, GLenum pname, GLint *param);
GLAPI PFNGLGETVERTEXARRAYINTEGERI_VEXTPROC glad_glGetVertexArrayIntegeri_vEXT;
#define glGetVertexArrayIntegeri_vEXT glad_glGetVertexArrayIntegeri_vEXT
typedef void (APIENTRYP PFNGLGETVERTEXARRAYPOINTERI_VEXTPROC)(GLuint vaobj, GLuint index, GLenum pname, void **param);
GLAPI PFNGLGETVERTEXARRAYPOINTERI_VEXTPROC glad_glGetVertexArrayPointeri_vEXT;
#define glGetVertexArrayPointeri_vEXT glad_glGetVertexArrayPointeri_vEXT
typedef void * (APIENTRYP PFNGLMAPNAMEDBUFFERRANGEEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr length, GLbitfield access);
GLAPI PFNGLMAPNAMEDBUFFERRANGEEXTPROC glad_glMapNamedBufferRangeEXT;
#define glMapNamedBufferRangeEXT glad_glMapNamedBufferRangeEXT
typedef void (APIENTRYP PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr length);
GLAPI PFNGLFLUSHMAPPEDNAMEDBUFFERRANGEEXTPROC glad_glFlushMappedNamedBufferRangeEXT;
#define glFlushMappedNamedBufferRangeEXT glad_glFlushMappedNamedBufferRangeEXT
typedef void (APIENTRYP PFNGLNAMEDBUFFERSTORAGEEXTPROC)(GLuint buffer, GLsizeiptr size, const void *data, GLbitfield flags);
GLAPI PFNGLNAMEDBUFFERSTORAGEEXTPROC glad_glNamedBufferStorageEXT;
#define glNamedBufferStorageEXT glad_glNamedBufferStorageEXT
typedef void (APIENTRYP PFNGLCLEARNAMEDBUFFERDATAEXTPROC)(GLuint buffer, GLenum internalformat, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARNAMEDBUFFERDATAEXTPROC glad_glClearNamedBufferDataEXT;
#define glClearNamedBufferDataEXT glad_glClearNamedBufferDataEXT
typedef void (APIENTRYP PFNGLCLEARNAMEDBUFFERSUBDATAEXTPROC)(GLuint buffer, GLenum internalformat, GLsizeiptr offset, GLsizeiptr size, GLenum format, GLenum type, const void *data);
GLAPI PFNGLCLEARNAMEDBUFFERSUBDATAEXTPROC glad_glClearNamedBufferSubDataEXT;
#define glClearNamedBufferSubDataEXT glad_glClearNamedBufferSubDataEXT
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERPARAMETERIEXTPROC)(GLuint framebuffer, GLenum pname, GLint param);
GLAPI PFNGLNAMEDFRAMEBUFFERPARAMETERIEXTPROC glad_glNamedFramebufferParameteriEXT;
#define glNamedFramebufferParameteriEXT glad_glNamedFramebufferParameteriEXT
typedef void (APIENTRYP PFNGLGETNAMEDFRAMEBUFFERPARAMETERIVEXTPROC)(GLuint framebuffer, GLenum pname, GLint *params);
GLAPI PFNGLGETNAMEDFRAMEBUFFERPARAMETERIVEXTPROC glad_glGetNamedFramebufferParameterivEXT;
#define glGetNamedFramebufferParameterivEXT glad_glGetNamedFramebufferParameterivEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1DEXTPROC)(GLuint program, GLint location, GLdouble x);
GLAPI PFNGLPROGRAMUNIFORM1DEXTPROC glad_glProgramUniform1dEXT;
#define glProgramUniform1dEXT glad_glProgramUniform1dEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2DEXTPROC)(GLuint program, GLint location, GLdouble x, GLdouble y);
GLAPI PFNGLPROGRAMUNIFORM2DEXTPROC glad_glProgramUniform2dEXT;
#define glProgramUniform2dEXT glad_glProgramUniform2dEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3DEXTPROC)(GLuint program, GLint location, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLPROGRAMUNIFORM3DEXTPROC glad_glProgramUniform3dEXT;
#define glProgramUniform3dEXT glad_glProgramUniform3dEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4DEXTPROC)(GLuint program, GLint location, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLPROGRAMUNIFORM4DEXTPROC glad_glProgramUniform4dEXT;
#define glProgramUniform4dEXT glad_glProgramUniform4dEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM1DVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM1DVEXTPROC glad_glProgramUniform1dvEXT;
#define glProgramUniform1dvEXT glad_glProgramUniform1dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM2DVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM2DVEXTPROC glad_glProgramUniform2dvEXT;
#define glProgramUniform2dvEXT glad_glProgramUniform2dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM3DVEXTPROC)(GLuint program, GLint location, GLsizei count, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM3DVEXTPROC glad_glProgramUniform3dvEXT;
#define glProgramUniform3dvEXT glad_glProgramUniform3dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORM4DVEXTPROC)(GLuint program, GLint location, GLsizei count,
const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORM4DVEXTPROC glad_glProgramUniform4dvEXT;
#define glProgramUniform4dvEXT glad_glProgramUniform4dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2DVEXTPROC glad_glProgramUniformMatrix2dvEXT;
#define glProgramUniformMatrix2dvEXT glad_glProgramUniformMatrix2dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3DVEXTPROC glad_glProgramUniformMatrix3dvEXT;
#define glProgramUniformMatrix3dvEXT glad_glProgramUniformMatrix3dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4DVEXTPROC glad_glProgramUniformMatrix4dvEXT;
#define glProgramUniformMatrix4dvEXT glad_glProgramUniformMatrix4dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X3DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X3DVEXTPROC glad_glProgramUniformMatrix2x3dvEXT;
#define glProgramUniformMatrix2x3dvEXT glad_glProgramUniformMatrix2x3dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX2X4DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX2X4DVEXTPROC glad_glProgramUniformMatrix2x4dvEXT;
#define glProgramUniformMatrix2x4dvEXT glad_glProgramUniformMatrix2x4dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X2DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X2DVEXTPROC glad_glProgramUniformMatrix3x2dvEXT;
#define glProgramUniformMatrix3x2dvEXT glad_glProgramUniformMatrix3x2dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX3X4DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX3X4DVEXTPROC glad_glProgramUniformMatrix3x4dvEXT;
#define glProgramUniformMatrix3x4dvEXT glad_glProgramUniformMatrix3x4dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X2DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X2DVEXTPROC glad_glProgramUniformMatrix4x2dvEXT;
#define glProgramUniformMatrix4x2dvEXT glad_glProgramUniformMatrix4x2dvEXT
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMMATRIX4X3DVEXTPROC)(GLuint program, GLint location, GLsizei count, GLboolean transpose, const GLdouble *value);
GLAPI PFNGLPROGRAMUNIFORMMATRIX4X3DVEXTPROC glad_glProgramUniformMatrix4x3dvEXT;
#define glProgramUniformMatrix4x3dvEXT glad_glProgramUniformMatrix4x3dvEXT
typedef void (APIENTRYP PFNGLTEXTUREBUFFERRANGEEXTPROC)(GLuint texture, GLenum target, GLenum internalformat, GLuint buffer, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLTEXTUREBUFFERRANGEEXTPROC glad_glTextureBufferRangeEXT;
#define glTextureBufferRangeEXT glad_glTextureBufferRangeEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGE1DEXTPROC)(GLuint texture, GLenum target, GLsizei levels, GLenum internalformat, GLsizei width);
GLAPI PFNGLTEXTURESTORAGE1DEXTPROC glad_glTextureStorage1DEXT;
#define glTextureStorage1DEXT glad_glTextureStorage1DEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGE2DEXTPROC)(GLuint texture, GLenum target, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLTEXTURESTORAGE2DEXTPROC glad_glTextureStorage2DEXT;
#define glTextureStorage2DEXT glad_glTextureStorage2DEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGE3DEXTPROC)(GLuint texture, GLenum target, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth);
GLAPI PFNGLTEXTURESTORAGE3DEXTPROC glad_glTextureStorage3DEXT;
#define glTextureStorage3DEXT glad_glTextureStorage3DEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGE2DMULTISAMPLEEXTPROC)(GLuint texture, GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXTURESTORAGE2DMULTISAMPLEEXTPROC glad_glTextureStorage2DMultisampleEXT;
#define glTextureStorage2DMultisampleEXT glad_glTextureStorage2DMultisampleEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGE3DMULTISAMPLEEXTPROC)(GLuint texture, GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedsamplelocations);
GLAPI PFNGLTEXTURESTORAGE3DMULTISAMPLEEXTPROC glad_glTextureStorage3DMultisampleEXT;
#define glTextureStorage3DMultisampleEXT glad_glTextureStorage3DMultisampleEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYBINDVERTEXBUFFEREXTPROC)(GLuint vaobj, GLuint bindingindex, GLuint buffer, GLintptr offset, GLsizei stride);
GLAPI PFNGLVERTEXARRAYBINDVERTEXBUFFEREXTPROC glad_glVertexArrayBindVertexBufferEXT;
#define glVertexArrayBindVertexBufferEXT glad_glVertexArrayBindVertexBufferEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBFORMATEXTPROC)(GLuint vaobj, GLuint attribindex, GLint size, GLenum type, GLboolean normalized, GLuint relativeoffset);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBFORMATEXTPROC glad_glVertexArrayVertexAttribFormatEXT;
#define glVertexArrayVertexAttribFormatEXT glad_glVertexArrayVertexAttribFormatEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBIFORMATEXTPROC)(GLuint vaobj, GLuint attribindex, GLint size, GLenum type, GLuint relativeoffset);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBIFORMATEXTPROC glad_glVertexArrayVertexAttribIFormatEXT;
#define glVertexArrayVertexAttribIFormatEXT glad_glVertexArrayVertexAttribIFormatEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBLFORMATEXTPROC)(GLuint vaobj, GLuint attribindex, GLint size, GLenum type, GLuint relativeoffset);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBLFORMATEXTPROC glad_glVertexArrayVertexAttribLFormatEXT;
#define glVertexArrayVertexAttribLFormatEXT glad_glVertexArrayVertexAttribLFormatEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBBINDINGEXTPROC)(GLuint vaobj, GLuint attribindex, GLuint bindingindex);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBBINDINGEXTPROC glad_glVertexArrayVertexAttribBindingEXT;
#define glVertexArrayVertexAttribBindingEXT glad_glVertexArrayVertexAttribBindingEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXBINDINGDIVISOREXTPROC)(GLuint vaobj, GLuint bindingindex, GLuint divisor);
```

```
GLAPI PFNGLVERTEXARRAYVERTEXBINDINGDIVISOREXTPROC glad_glVertexArrayVertexBindingDivisorEXT;
#define glVertexArrayVertexBindingDivisorEXT glad_glVertexArrayVertexBindingDivisorEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBLOFFSETEXTPROC)(GLuint vaobj, GLuint buffer, GLuint index, GLint size, GLenum type, GLsizei stride, GLintptr offset);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBLOFFSETEXTPROC glad_glVertexArrayVertexAttribLOffsetEXT;
#define glVertexArrayVertexAttribLOffsetEXT glad_glVertexArrayVertexAttribLOffsetEXT
typedef void (APIENTRYP PFNGLTEXTUREPAGECOMMITMENTEXTPROC)(GLuint texture, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLboolean commit);
GLAPI PFNGLTEXTUREPAGECOMMITMENTEXTPROC glad_glTexturePageCommitmentEXT;
#define glTexturePageCommitmentEXT glad_glTexturePageCommitmentEXT
typedef void (APIENTRYP PFNGLVERTEXARRAYVERTEXATTRIBDIVISOREXTPROC)(GLuint vaobj, GLuint index, GLuint divisor);
GLAPI PFNGLVERTEXARRAYVERTEXATTRIBDIVISOREXTPROC glad_glVertexArrayVertexAttribDivisorEXT;
#define glVertexArrayVertexAttribDivisorEXT glad_glVertexArrayVertexAttribDivisorEXT
#endif
#ifndef GL_EXT_draw_buffers2
#define GL_EXT_draw_buffers2 1
GLAPI int GLAD_GL_EXT_draw_buffers2;
typedef void (APIENTRYP PFNGLCOLORMASKINDEXEDEXTPROC)(GLuint index, GLboolean r, GLboolean g, GLboolean b, GLboolean a);
GLAPI PFNGLCOLORMASKINDEXEDEXTPROC glad_glColorMaskIndexedEXT;
#define glColorMaskIndexedEXT glad_glColorMaskIndexedEXT
#endif
#ifndef GL_EXT_draw_instanced
#define GL_EXT_draw_instanced 1
GLAPI int GLAD_GL_EXT_draw_instanced;
typedef void (APIENTRYP PFNGLDRAWARRAYSINSTANCEDEXTPROC)(GLenum mode, GLint start, GLsizei count, GLsizei primcount);
GLAPI PFNGLDRAWARRAYSINSTANCEDEXTPROC glad_glDrawArraysInstancedEXT;
#define glDrawArraysInstancedEXT glad_glDrawArraysInstancedEXT
typedef void (APIENTRYP PFNGLDRAWELEMENTSINSTANCEDEXTPROC)(GLenum mode, GLsizei count, GLenum type, const void *indices, GLsizei primcount);
GLAPI PFNGLDRAWELEMENTSINSTANCEDEXTPROC glad_glDrawElementsInstancedEXT;
#define glDrawElementsInstancedEXT glad_glDrawElementsInstancedEXT
#endif
#ifndef GL_EXT_draw_range_elements
#define GL_EXT_draw_range_elements 1
GLAPI int GLAD_GL_EXT_draw_range_elements;
typedef void (APIENTRYP PFNGLDRAWRANGEELEMENTSEXTPROC)(GLenum mode, GLuint start, GLuint end, GLsizei count, GLenum type, const void *indices);
GLAPI PFNGLDRAWRANGEELEMENTSEXTPROC glad_glDrawRangeElementsEXT;
#define glDrawRangeElementsEXT glad_glDrawRangeElementsEXT
#endif
#ifndef GL_EXT_external_buffer
#define GL_EXT_external_buffer 1
GLAPI int GLAD_GL_EXT_external_buffer;
typedef void (APIENTRYP PFNGLBUFFERSTORAGEEXTERNALEXTPROC)(GLenum target, GLintptr offset, GLsizeiptr size, GLeglClientBufferEXT clientBuffer, GLbitfield flags);
GLAPI PFNGLBUFFERSTORAGEEXTERNALEXTPROC glad_glBufferStorageExternalEXT;
#define glBufferStorageExternalEXT glad_glBufferStorageExternalEXT
typedef void (APIENTRYP PFNGLNAMEDBUFFERSTORAGEEXTERNALEXTPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, GLeglClientBufferEXT clientBuffer, GLbitfield flags);
GLAPI PFNGLNAMEDBUFFERSTORAGEEXTERNALEXTPROC glad_glNamedBufferStorageExternalEXT;
#define glNamedBufferStorageExternalEXT glad_glNamedBufferStorageExternalEXT
#endif
#ifndef GL_EXT_fog_coord
#define GL_EXT_fog_coord 1
GLAPI int GLAD_GL_EXT_fog_coord;
typedef void (APIENTRYP PFNGLFOGCOORDFEXTPROC)(GLfloat coord);
GLAPI PFNGLFOGCOORDFEXTPROC glad_glFogCoordfEXT;
#define glFogCoordfEXT glad_glFogCoordfEXT
typedef void (APIENTRYP PFNGLFOGCOORDFVEXTPROC)(const GLfloat *coord);
GLAPI PFNGLFOGCOORDFVEXTPROC glad_glFogCoordfvEXT;
#define glFogCoordfvEXT glad_glFogCoordfvEXT
typedef void (APIENTRYP PFNGLFOGCOORDDEXTPROC)(GLdouble coord);
GLAPI PFNGLFOGCOORDDEXTPROC glad_glFogCoorddEXT;
#define glFogCoorddEXT glad_glFogCoorddEXT
typedef void (APIENTRYP PFNGLFOGCOORDDVEXTPROC)(const GLdouble *coord);
GLAPI PFNGLFOGCOORDDVEXTPROC glad_glFogCoorddvEXT;
#define glFogCoorddvEXT glad_glFogCoorddvEXT
typedef void (APIENTRYP PFNGLFOGCOORDPOINTEREXTPROC)(GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLFOGCOORDPOINTEREXTPROC glad_glFogCoordPointerEXT;
#define glFogCoordPointerEXT glad_glFogCoordPointerEXT
#endif
#ifndef GL_EXT_framebuffer_blit
#define GL_EXT_framebuffer_blit 1
GLAPI int GLAD_GL_EXT_framebuffer_blit;
typedef void (APIENTRYP PFNGLBLITFRAMEBUFFEREXTPROC)(GLint srcX0, GLint srcY0, GLint srcX1, GLint srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLbitfield mask, GLenum filter);
GLAPI PFNGLBLITFRAMEBUFFEREXTPROC glad_glBlitFramebufferEXT;
#define glBlitFramebufferEXT glad_glBlitFramebufferEXT
#endif
#ifndef GL_EXT_framebuffer_blit_layers
#define GL_EXT_framebuffer_blit_layers 1
GLAPI int GLAD_GL_EXT_framebuffer_blit_layers;
typedef void (APIENTRYP PFNGLBLITFRAMEBUFFERLAYERSEXTPROC)(GLint srcX0, GLint srcY0, GLint srcX1,
GLint srcY1, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLbitfield mask, GLenum filter);
GLAPI PFNGLBLITFRAMEBUFFERLAYERSEXTPROC glad_glBlitFramebufferLayersEXT;
#define glBlitFramebufferLayersEXT glad_glBlitFramebufferLayersEXT
typedef void (APIENTRYP PFNGLBLITFRAMEBUFFERLAYEREXTPROC)(GLint srcX0, GLint srcY0, GLint srcX1, GLint srcY1, GLint srcLayer, GLint dstX0, GLint dstY0, GLint dstX1, GLint dstY1, GLint dstLayer, GLbitfield mask, GLenum filter);
GLAPI PFNGLBLITFRAMEBUFFERLAYEREXTPROC glad_glBlitFramebufferLayerEXT;
#define glBlitFramebufferLayerEXT glad_glBlitFramebufferLayerEXT
#endif
#ifndef GL_EXT_framebuffer_multisample
#define GL_EXT_framebuffer_multisample 1
GLAPI int GLAD_GL_EXT_framebuffer_multisample;
typedef void (APIENTRYP PFNGLRENDERBUFFERSTORAGEMULTISAMPLEEXTPROC)(GLenum target, GLsizei samples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLRENDERBUFFERSTORAGEMULTISAMPLEEXTPROC glad_glRenderbufferStorageMultisampleEXT;
#define glRenderbufferStorageMultisampleEXT glad_glRenderbufferStorageMultisampleEXT
#endif
#ifndef GL_EXT_framebuffer_multisample_blit_scaled
#define GL_EXT_framebuffer_multisample_blit_scaled 1
GLAPI int GLAD_GL_EXT_framebuffer_multisample_blit_scaled;
#endif
#ifndef GL_EXT_framebuffer_object
#define GL_EXT_framebuffer_object 1
GLAPI int GLAD_GL_EXT_framebuffer_object;
typedef GLboolean (APIENTRYP PFNGLISRENDERBUFFEREXTPROC)(GLuint renderbuffer);
GLAPI PFNGLISRENDERBUFFEREXTPROC glad_glIsRenderbufferEXT;
#define glIsRenderbufferEXT glad_glIsRenderbufferEXT
typedef void (APIENTRYP PFNGLBINDRENDERBUFFEREXTPROC)(GLenum target, GLuint renderbuffer);
GLAPI PFNGLBINDRENDERBUFFEREXTPROC glad_glBindRenderbufferEXT;
#define glBindRenderbufferEXT glad_glBindRenderbufferEXT
typedef void (APIENTRYP PFNGLDELETERENDERBUFFERSEXTPROC)(GLsizei n, const GLuint *renderbuffers);
GLAPI PFNGLDELETERENDERBUFFERSEXTPROC glad_glDeleteRenderbuffersEXT;
#define glDeleteRenderbuffersEXT glad_glDeleteRenderbuffersEXT
typedef void (APIENTRYP PFNGLGENRENDERBUFFERSEXTPROC)(GLsizei n, GLuint *renderbuffers);
GLAPI PFNGLGENRENDERBUFFERSEXTPROC glad_glGenRenderbuffersEXT;
#define glGenRenderbuffersEXT glad_glGenRenderbuffersEXT
typedef void (APIENTRYP PFNGLRENDERBUFFERSTORAGEEXTPROC)(GLenum target, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLRENDERBUFFERSTORAGEEXTPROC glad_glRenderbufferStorageEXT;
#define glRenderbufferStorageEXT glad_glRenderbufferStorageEXT
typedef void (APIENTRYP PFNGLGETRENDERBUFFERPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETRENDERBUFFERPARAMETERIVEXTPROC glad_glGetRenderbufferParameterivEXT;
#define glGetRenderbufferParameterivEXT glad_glGetRenderbufferParameterivEXT
typedef GLboolean (APIENTRYP PFNGLISFRAMEBUFFEREXTPROC)(GLuint framebuffer);
GLAPI PFNGLISFRAMEBUFFEREXTPROC glad_glIsFramebufferEXT;
#define glIsFramebufferEXT glad_glIsFramebufferEXT
typedef void (APIENTRYP PFNGLBINDFRAMEBUFFEREXTPROC)(GLenum target, GLuint framebuffer);
GLAPI PFNGLBINDFRAMEBUFFEREXTPROC glad_glBindFramebufferEXT;
#define glBindFramebufferEXT glad_glBindFramebufferEXT
typedef void (APIENTRYP PFNGLDELETEFRAMEBUFFERSEXTPROC)(GLsizei n, const GLuint *framebuffers);
GLAPI PFNGLDELETEFRAMEBUFFERSEXTPROC glad_glDeleteFramebuffersEXT;
#define glDeleteFramebuffersEXT glad_glDeleteFramebuffersEXT
typedef void (APIENTRYP PFNGLGENFRAMEBUFFERSEXTPROC)(GLsizei n, GLuint *framebuffers);
GLAPI PFNGLGENFRAMEBUFFERSEXTPROC glad_glGenFramebuffersEXT;
#define glGenFramebuffersEXT glad_glGenFramebuffersEXT
typedef GLenum (APIENTRYP PFNGLCHECKFRAMEBUFFERSTATUSEXTPROC)(GLenum target);
GLAPI PFNGLCHECKFRAMEBUFFERSTATUSEXTPROC glad_glCheckFramebufferStatusEXT;
#define glCheckFramebufferStatusEXT glad_glCheckFramebufferStatusEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURE1DEXTPROC)(GLenum target, GLenum attachment, GLenum textarget, GLuint texture, GLint level);
GLAPI PFNGLFRAMEBUFFERTEXTURE1DEXTPROC glad_glFramebufferTexture1DEXT;
#define glFramebufferTexture1DEXT glad_glFramebufferTexture1DEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURE2DEXTPROC)(GLenum target, GLenum attachment, GLenum textarget, GLuint texture, GLint level);
GLAPI PFNGLFRAMEBUFFERTEXTURE2DEXTPROC glad_glFramebufferTexture2DEXT;
#define glFramebufferTexture2DEXT glad_glFramebufferTexture2DEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURE3DEXTPROC)(GLenum target, GLenum attachment, GLenum textarget, GLuint texture, GLint level, GLint zoffset);
GLAPI PFNGLFRAMEBUFFERTEXTURE3DEXTPROC glad_glFramebufferTexture3DEXT;
#define glFramebufferTexture3DEXT glad_glFramebufferTexture3DEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERRENDERBUFFEREXTPROC)(GLenum target, GLenum attachment, GLenum renderbuffertarget, GLuint renderbuffer);
GLAPI PFNGLFRAMEBUFFERRENDERBUFFEREXTPROC glad_glFramebufferRenderbufferEXT;
#define glFramebufferRenderbufferEXT glad_glFramebufferRenderbufferEXT
typedef void (APIENTRYP PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVEXTPROC)(GLenum target, GLenum attachment, GLenum pname, GLint *params);
GLAPI PFNGLGETFRAMEBUFFERATTACHMENTPARAMETERIVEXTPROC glad_glGetFramebufferAttachmentParameterivEXT;
#define glGetFramebufferAttachmentParameterivEXT glad_glGetFramebufferAttachmentParameterivEXT
typedef void (APIENTRYP PFNGLGENERATEMIPMAPEXTPROC)(GLenum target);
GLAPI PFNGLGENERATEMIPMAPEXTPROC glad_glGenerateMipmapEXT;
#define glGenerateMipmapEXT glad_glGenerateMipmapEXT
#endif
#ifndef GL_EXT_framebuffer_sRGB
#define GL_EXT_framebuffer_sRGB 1
GLAPI int GLAD_GL_EXT_framebuffer_sRGB;
#endif
```

```
#ifndef GL_EXT_geometry_shader4
#define GL_EXT_geometry_shader4 1
GLAPI int GLAD_GL_EXT_geometry_shader4;
typedef void (APIENTRYP PFNGLPROGRAMPARAMETERIEXTPROC)(GLuint program, GLenum pname, GLint value);
GLAPI PFNGLPROGRAMPARAMETERIEXTPROC glad_glProgramParameteriEXT;
#define glProgramParameteriEXT glad_glProgramParameteriEXT
#endif
#ifndef GL_EXT_gpu_program_parameters
#define GL_EXT_gpu_program_parameters 1
GLAPI int GLAD_GL_EXT_gpu_program_parameters;
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETERS4FVEXTPROC)(GLenum target, GLuint index, GLsizei count, const GLfloat *params);
GLAPI PFNGLPROGRAMENVPARAMETERS4FVEXTPROC glad_glProgramEnvParameters4fvEXT;
#define glProgramEnvParameters4fvEXT glad_glProgramEnvParameters4fvEXT
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETERS4FVEXTPROC)(GLenum target, GLuint index, GLsizei count, const GLfloat *params);
GLAPI PFNGLPROGRAMLOCALPARAMETERS4FVEXTPROC glad_glProgramLocalParameters4fvEXT;
#define glProgramLocalParameters4fvEXT glad_glProgramLocalParameters4fvEXT
#endif
#ifndef GL_EXT_gpu_shader4
#define GL_EXT_gpu_shader4 1
GLAPI int GLAD_GL_EXT_gpu_shader4;
typedef void (APIENTRYP PFNGLGETUNIFORMUIVEXTPROC)(GLuint program, GLint location, GLuint *params);
GLAPI PFNGLGETUNIFORMUIVEXTPROC glad_glGetUniformuivEXT;
#define glGetUniformuivEXT glad_glGetUniformuivEXT
typedef void (APIENTRYP PFNGLBINDFRAGDATALOCATIONEXTPROC)(GLuint program, GLuint color, const GLchar *name);
GLAPI PFNGLBINDFRAGDATALOCATIONEXTPROC glad_glBindFragDataLocationEXT;
#define glBindFragDataLocationEXT glad_glBindFragDataLocationEXT
typedef GLint (APIENTRYP PFNGLGETFRAGDATALOCATIONEXTPROC)(GLuint program, const GLchar *name);
GLAPI PFNGLGETFRAGDATALOCATIONEXTPROC glad_glGetFragDataLocationEXT;
#define glGetFragDataLocationEXT glad_glGetFragDataLocationEXT
typedef void (APIENTRYP PFNGLUNIFORM1UIEXTPROC)(GLint location, GLuint v0);
GLAPI PFNGLUNIFORM1UIEXTPROC glad_glUniform1uiEXT;
#define glUniform1uiEXT glad_glUniform1uiEXT
typedef void (APIENTRYP PFNGLUNIFORM2UIEXTPROC)(GLint location, GLuint v0, GLuint v1);
GLAPI PFNGLUNIFORM2UIEXTPROC glad_glUniform2uiEXT;
#define glUniform2uiEXT glad_glUniform2uiEXT
typedef void (APIENTRYP PFNGLUNIFORM3UIEXTPROC)(GLint location, GLuint v0, GLuint v1, GLuint v2);
GLAPI PFNGLUNIFORM3UIEXTPROC glad_glUniform3uiEXT;
#define glUniform3uiEXT glad_glUniform3uiEXT
typedef void (APIENTRYP PFNGLUNIFORM4UIEXTPROC)(GLint location, GLuint v0, GLuint v1, GLuint v2, GLuint v3);
GLAPI PFNGLUNIFORM4UIEXTPROC glad_glUniform4uiEXT;
#define glUniform4uiEXT glad_glUniform4uiEXT
typedef void (APIENTRYP PFNGLUNIFORM1UIVEXTPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM1UIVEXTPROC glad_glUniform1uivEXT;
#define glUniform1uivEXT glad_glUniform1uivEXT
typedef void (APIENTRYP PFNGLUNIFORM2UIVEXTPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM2UIVEXTPROC glad_glUniform2uivEXT;
#define glUniform2uivEXT glad_glUniform2uivEXT
typedef void (APIENTRYP PFNGLUNIFORM3UIVEXTPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM3UIVEXTPROC glad_glUniform3uivEXT;
#define glUniform3uivEXT glad_glUniform3uivEXT
typedef void (APIENTRYP PFNGLUNIFORM4UIVEXTPROC)(GLint location, GLsizei count, const GLuint *value);
GLAPI PFNGLUNIFORM4UIVEXTPROC glad_glUniform4uivEXT;
#define glUniform4uivEXT glad_glUniform4uivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1IEXTPROC)(GLuint index, GLint x);
GLAPI PFNGLVERTEXATTRIBI1IEXTPROC glad_glVertexAttribI1iEXT;
#define glVertexAttribI1iEXT glad_glVertexAttribI1iEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2IEXTPROC)(GLuint index, GLint x, GLint y);
GLAPI PFNGLVERTEXATTRIBI2IEXTPROC glad_glVertexAttribI2iEXT;
#define glVertexAttribI2iEXT glad_glVertexAttribI2iEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3IEXTPROC)(GLuint index, GLint x, GLint y, GLint z);
GLAPI PFNGLVERTEXATTRIBI3IEXTPROC glad_glVertexAttribI3iEXT;
#define glVertexAttribI3iEXT glad_glVertexAttribI3iEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4IEXTPROC)(GLuint index, GLint x, GLint y, GLint z, GLint w);
GLAPI PFNGLVERTEXATTRIBI4IEXTPROC glad_glVertexAttribI4iEXT;
#define glVertexAttribI4iEXT glad_glVertexAttribI4iEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1UIEXTPROC)(GLuint index, GLuint x);
GLAPI PFNGLVERTEXATTRIBI1UIEXTPROC glad_glVertexAttribI1uiEXT;
#define glVertexAttribI1uiEXT glad_glVertexAttribI1uiEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2UIEXTPROC)(GLuint index, GLuint x, GLuint y);
GLAPI PFNGLVERTEXATTRIBI2UIEXTPROC glad_glVertexAttribI2uiEXT;
#define glVertexAttribI2uiEXT glad_glVertexAttribI2uiEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3UIEXTPROC)(GLuint index, GLuint x, GLuint y, GLuint z);
GLAPI PFNGLVERTEXATTRIBI3UIEXTPROC glad_glVertexAttribI3uiEXT;
#define glVertexAttribI3uiEXT glad_glVertexAttribI3uiEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4UIEXTPROC)(GLuint index, GLuint x, GLuint y, GLuint z, GLuint w);
GLAPI PFNGLVERTEXATTRIBI4UIEXTPROC glad_glVertexAttribI4uiEXT;
#define glVertexAttribI4uiEXT glad_glVertexAttribI4uiEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1IVEXTPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIBI1IVEXTPROC glad_glVertexAttribI1ivEXT;
#define glVertexAttribI1ivEXT glad_glVertexAttribI1ivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2IVEXTPROC)(GLuint index, const GLint *v);
```

```
GLAPI PFNGLVERTEXATTRIBI2IVEXTPROC glad_glVertexAttribI2ivEXT;
#define glVertexAttribI2ivEXT glad_glVertexAttribI2ivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3IVEXTPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIBI3IVEXTPROC glad_glVertexAttribI3ivEXT;
#define glVertexAttribI3ivEXT glad_glVertexAttribI3ivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4IVEXTPROC)(GLuint index, const GLint *v);
GLAPI PFNGLVERTEXATTRIBI4IVEXTPROC glad_glVertexAttribI4ivEXT;
#define glVertexAttribI4ivEXT glad_glVertexAttribI4ivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI1UIVEXTPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI1UIVEXTPROC glad_glVertexAttribI1uivEXT;
#define glVertexAttribI1uivEXT glad_glVertexAttribI1uivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI2UIVEXTPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI2UIVEXTPROC glad_glVertexAttribI2uivEXT;
#define glVertexAttribI2uivEXT glad_glVertexAttribI2uivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI3UIVEXTPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI3UIVEXTPROC glad_glVertexAttribI3uivEXT;
#define glVertexAttribI3uivEXT glad_glVertexAttribI3uivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4UIVEXTPROC)(GLuint index, const GLuint *v);
GLAPI PFNGLVERTEXATTRIBI4UIVEXTPROC glad_glVertexAttribI4uivEXT;
#define glVertexAttribI4uivEXT glad_glVertexAttribI4uivEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4BVEXTPROC)(GLuint index, const GLbyte *v);
GLAPI PFNGLVERTEXATTRIBI4BVEXTPROC glad_glVertexAttribI4bvEXT;
#define glVertexAttribI4bvEXT glad_glVertexAttribI4bvEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4SVEXTPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIBI4SVEXTPROC glad_glVertexAttribI4svEXT;
#define glVertexAttribI4svEXT glad_glVertexAttribI4svEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4UBVEXTPROC)(GLuint index, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIBI4UBVEXTPROC glad_glVertexAttribI4ubvEXT;
#define glVertexAttribI4ubvEXT glad_glVertexAttribI4ubvEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBI4USVEXTPROC)(GLuint index, const GLushort *v);
GLAPI PFNGLVERTEXATTRIBI4USVEXTPROC glad_glVertexAttribI4usvEXT;
#define glVertexAttribI4usvEXT glad_glVertexAttribI4usvEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBIPOINTEREXTPROC)(GLuint index, GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLVERTEXATTRIBIPOINTEREXTPROC glad_glVertexAttribIPointerEXT;
#define glVertexAttribIPointerEXT glad_glVertexAttribIPointerEXT
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBIIVEXTPROC)(GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETVERTEXATTRIBIIVEXTPROC glad_glGetVertexAttribIivEXT;
#define glGetVertexAttribIivEXT glad_glGetVertexAttribIivEXT
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBIUIVEXTPROC)(GLuint index, GLenum pname, GLuint *params);
GLAPI PFNGLGETVERTEXATTRIBIUIVEXTPROC glad_glGetVertexAttribIuivEXT;
#define glGetVertexAttribIuivEXT glad_glGetVertexAttribIuivEXT
#endif
#ifndef GL_EXT_histogram
#define GL_EXT_histogram 1
GLAPI int GLAD_GL_EXT_histogram;
typedef void (APIENTRYP PFNGLGETHISTOGRAMEXTPROC)(GLenum target, GLboolean reset, GLenum format, GLenum type, void *values);
GLAPI PFNGLGETHISTOGRAMEXTPROC glad_glGetHistogramEXT;
#define glGetHistogramEXT glad_glGetHistogramEXT
typedef void (APIENTRYP PFNGLGETHISTOGRAMPARAMETERFVEXTPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETHISTOGRAMPARAMETERFVEXTPROC glad_glGetHistogramParameterfvEXT;
#define glGetHistogramParameterfvEXT glad_glGetHistogramParameterfvEXT
typedef void (APIENTRYP PFNGLGETHISTOGRAMPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETHISTOGRAMPARAMETERIVEXTPROC glad_glGetHistogramParameterivEXT;
#define glGetHistogramParameterivEXT glad_glGetHistogramParameterivEXT
typedef void (APIENTRYP PFNGLGETMINMAXEXTPROC)(GLenum target, GLboolean reset, GLenum format, GLenum type, void *values);
GLAPI PFNGLGETMINMAXEXTPROC glad_glGetMinmaxEXT;
#define glGetMinmaxEXT glad_glGetMinmaxEXT
typedef void (APIENTRYP PFNGLGETMINMAXPARAMETERFVEXTPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMINMAXPARAMETERFVEXTPROC glad_glGetMinmaxParameterfvEXT;
#define glGetMinmaxParameterfvEXT glad_glGetMinmaxParameterfvEXT
typedef void (APIENTRYP PFNGLGETMINMAXPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETMINMAXPARAMETERIVEXTPROC glad_glGetMinmaxParameterivEXT;
#define glGetMinmaxParameterivEXT glad_glGetMinmaxParameterivEXT
typedef void (APIENTRYP PFNGLHISTOGRAMEXTPROC)(GLenum target, GLsizei width, GLenum internalformat, GLboolean sink);
GLAPI PFNGLHISTOGRAMEXTPROC glad_glHistogramEXT;
#define glHistogramEXT glad_glHistogramEXT
typedef void (APIENTRYP PFNGLMINMAXEXTPROC)(GLenum target, GLenum internalformat, GLboolean sink);
GLAPI PFNGLMINMAXEXTPROC glad_glMinmaxEXT;
#define glMinmaxEXT glad_glMinmaxEXT
typedef void (APIENTRYP PFNGLRESETHISTOGRAMEXTPROC)(GLenum target);
GLAPI PFNGLRESETHISTOGRAMEXTPROC glad_glResetHistogramEXT;
#define glResetHistogramEXT glad_glResetHistogramEXT
typedef void (APIENTRYP PFNGLRESETMINMAXEXTPROC)(GLenum target);
GLAPI PFNGLRESETMINMAXEXTPROC glad_glResetMinmaxEXT;
#define glResetMinmaxEXT glad_glResetMinmaxEXT
#endif
#ifndef GL_EXT_index_array_formats
#define GL_EXT_index_array_formats 1
GLAPI int GLAD_GL_EXT_index_array_formats;
#endif
```

```
#ifndef GL_EXT_index_func
#define GL_EXT_index_func 1
GLAPI int GLAD_GL_EXT_index_func;
typedef void (APIENTRYP PFNGLINDEXFUNCEXTPROC)(GLenum func, GLclampf ref);
GLAPI PFNGLINDEXFUNCEXTPROC glad_glIndexFuncEXT;
#define glIndexFuncEXT glad_glIndexFuncEXT
#endif
#ifndef GL_EXT_index_material
#define GL_EXT_index_material 1
GLAPI int GLAD_GL_EXT_index_material;
typedef void (APIENTRYP PFNGLINDEXMATERIALEXTPROC)(GLenum face, GLenum mode);
GLAPI PFNGLINDEXMATERIALEXTPROC glad_glIndexMaterialEXT;
#define glIndexMaterialEXT glad_glIndexMaterialEXT
#endif
#ifndef GL_EXT_index_texture
#define GL_EXT_index_texture 1
GLAPI int GLAD_GL_EXT_index_texture;
#endif
#ifndef GL_EXT_light_texture
#define GL_EXT_light_texture 1
GLAPI int GLAD_GL_EXT_light_texture;
typedef void (APIENTRYP PFNGLAPPLYTEXTUREEXTPROC)(GLenum mode);
GLAPI PFNGLAPPLYTEXTUREEXTPROC glad_glApplyTextureEXT;
#define glApplyTextureEXT glad_glApplyTextureEXT
typedef void (APIENTRYP PFNGLTEXTURELIGHTEXTPROC)(GLenum pname);
GLAPI PFNGLTEXTURELIGHTEXTPROC glad_glTextureLightEXT;
#define glTextureLightEXT glad_glTextureLightEXT
typedef void (APIENTRYP PFNGLTEXTUREMATERIALEXTPROC)(GLenum face, GLenum mode);
GLAPI PFNGLTEXTUREMATERIALEXTPROC glad_glTextureMaterialEXT;
#define glTextureMaterialEXT glad_glTextureMaterialEXT
#endif
#ifndef GL_EXT_memory_object
#define GL_EXT_memory_object 1
GLAPI int GLAD_GL_EXT_memory_object;
typedef void (APIENTRYP PFNGLGETUNSIGNEDBYTEVEXTPROC)(GLenum pname, GLubyte *data);
GLAPI PFNGLGETUNSIGNEDBYTEVEXTPROC glad_glGetUnsignedBytevEXT;
#define glGetUnsignedBytevEXT glad_glGetUnsignedBytevEXT
typedef void (APIENTRYP PFNGLGETUNSIGNEDBYTEI_VEXTPROC)(GLenum target, GLuint index, GLubyte *data);
GLAPI PFNGLGETUNSIGNEDBYTEI_VEXTPROC glad_glGetUnsignedBytei_vEXT;
#define glGetUnsignedBytei_vEXT glad_glGetUnsignedBytei_vEXT
typedef void (APIENTRYP PFNGLDELETEMEMORYOBJECTSEXTPROC)(GLsizei n, const GLuint *memoryObjects);
GLAPI PFNGLDELETEMEMORYOBJECTSEXTPROC glad_glDeleteMemoryObjectsEXT;
#define glDeleteMemoryObjectsEXT glad_glDeleteMemoryObjectsEXT
typedef GLboolean (APIENTRYP PFNGLISMEMORYOBJECTEXTPROC)(GLuint memoryObject);
GLAPI PFNGLISMEMORYOBJECTEXTPROC glad_glIsMemoryObjectEXT;
#define glIsMemoryObjectEXT glad_glIsMemoryObjectEXT
typedef void (APIENTRYP PFNGLCREATEMEMORYOBJECTSEXTPROC)(GLsizei n, GLuint *memoryObjects);
GLAPI PFNGLCREATEMEMORYOBJECTSEXTPROC glad_glCreateMemoryObjectsEXT;
#define glCreateMemoryObjectsEXT glad_glCreateMemoryObjectsEXT
typedef void (APIENTRYP PFNGLMEMORYOBJECTPARAMETERIVEXTPROC)(GLuint memoryObject, GLenum pname, const GLint *params);
GLAPI PFNGLMEMORYOBJECTPARAMETERIVEXTPROC glad_glMemoryObjectParameterivEXT;
#define glMemoryObjectParameterivEXT glad_glMemoryObjectParameterivEXT
typedef void (APIENTRYP PFNGLGETMEMORYOBJECTPARAMETERIVEXTPROC)(GLuint memoryObject, GLenum pname, GLint *params);
GLAPI PFNGLGETMEMORYOBJECTPARAMETERIVEXTPROC glad_glGetMemoryObjectParameterivEXT;
#define glGetMemoryObjectParameterivEXT glad_glGetMemoryObjectParameterivEXT
typedef void (APIENTRYP PFNGLTEXSTORAGEMEM2DEXTPROC)(GLenum target, GLsizei levels, GLenum internalFormat, GLsizei width, GLsizei height, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXSTORAGEMEM2DEXTPROC glad_glTexStorageMem2DEXT;
#define glTexStorageMem2DEXT glad_glTexStorageMem2DEXT
typedef void (APIENTRYP PFNGLTEXSTORAGEMEM2DMULTISAMPLEEXTPROC)(GLenum target, GLsizei samples, GLenum internalFormat, GLsizei width, GLsizei height, GLboolean fixedSampleLocations, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXSTORAGEMEM2DMULTISAMPLEEXTPROC glad_glTexStorageMem2DMultisampleEXT;
#define glTexStorageMem2DMultisampleEXT glad_glTexStorageMem2DMultisampleEXT
typedef void (APIENTRYP PFNGLTEXSTORAGEMEM3DEXTPROC)(GLenum target, GLsizei levels, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXSTORAGEMEM3DEXTPROC glad_glTexStorageMem3DEXT;
#define glTexStorageMem3DEXT glad_glTexStorageMem3DEXT
typedef void (APIENTRYP PFNGLTEXSTORAGEMEM3DMULTISAMPLEEXTPROC)(GLenum target, GLsizei samples, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXSTORAGEMEM3DMULTISAMPLEEXTPROC glad_glTexStorageMem3DMultisampleEXT;
#define glTexStorageMem3DMultisampleEXT glad_glTexStorageMem3DMultisampleEXT
typedef void (APIENTRYP PFNGLBUFFERSTORAGEMEMEXTPROC)(GLenum target, GLsizeiptr size, GLuint memory, GLuint64 offset);
GLAPI PFNGLBUFFERSTORAGEMEMEXTPROC glad_glBufferStorageMemEXT;
#define glBufferStorageMemEXT glad_glBufferStorageMemEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGEMEM2DEXTPROC)(GLuint texture, GLsizei levels, GLenum internalFormat, GLsizei width, GLsizei height, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXTURESTORAGEMEM2DEXTPROC glad_glTextureStorageMem2DEXT;
#define glTextureStorageMem2DEXT glad_glTextureStorageMem2DEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGEMEM2DMULTISAMPLEEXTPROC)(GLuint texture, GLsizei samples, GLenum internalFormat, GLsizei width, GLsizei height, GLboolean fixedSampleLocations, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXTURESTORAGEMEM2DMULTISAMPLEEXTPROC glad_glTextureStorageMem2DMultisampleEXT;
```

```
#define glTextureStorageMem2DMultisampleEXT glad_glTextureStorageMem2DMultisampleEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGEMEM3DEXTPROC)(GLuint texture, GLsizei levels, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXTURESTORAGEMEM3DEXTPROC glad_glTextureStorageMem3DEXT;
#define glTextureStorageMem3DEXT glad_glTextureStorageMem3DEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGEMEM3DMULTISAMPLEEXTPROC)(GLuint texture, GLsizei samples, GLenum internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXTURESTORAGEMEM3DMULTISAMPLEEXTPROC glad_glTextureStorageMem3DMultisampleEXT;
#define glTextureStorageMem3DMultisampleEXT glad_glTextureStorageMem3DMultisampleEXT
typedef void (APIENTRYP PFNGLNAMEDBUFFERSTORAGEMEMEXTPROC)(GLuint buffer, GLsizeiptr size, GLuint memory, GLuint64 offset);
GLAPI PFNGLNAMEDBUFFERSTORAGEMEMEXTPROC glad_glNamedBufferStorageMemEXT;
#define glNamedBufferStorageMemEXT glad_glNamedBufferStorageMemEXT
typedef void (APIENTRYP PFNGLTEXSTORAGEMEM1DEXTPROC)(GLenum target, GLsizei levels, GLenum internalFormat, GLsizei width, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXSTORAGEMEM1DEXTPROC glad_glTexStorageMem1DEXT;
#define glTexStorageMem1DEXT glad_glTexStorageMem1DEXT
typedef void (APIENTRYP PFNGLTEXTURESTORAGEMEM1DEXTPROC)(GLuint texture, GLsizei levels, GLenum internalFormat, GLsizei width, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXTURESTORAGEMEM1DEXTPROC glad_glTextureStorageMem1DEXT;
#define glTextureStorageMem1DEXT glad_glTextureStorageMem1DEXT
#endif
#ifndef GL_EXT_memory_object_fd
#define GL_EXT_memory_object_fd 1
GLAPI int GLAD_GL_EXT_memory_object_fd;
typedef void (APIENTRYP PFNGLIMPORTMEMORYFDEXTPROC)(GLuint memory, GLuint64 size, GLenum handleType, GLint fd);
GLAPI PFNGLIMPORTMEMORYFDEXTPROC glad_glImportMemoryFdEXT;
#define glImportMemoryFdEXT glad_glImportMemoryFdEXT
#endif
#ifndef GL_EXT_memory_object_win32
#define GL_EXT_memory_object_win32 1
GLAPI int GLAD_GL_EXT_memory_object_win32;
typedef void (APIENTRYP PFNGLIMPORTMEMORYWIN32HANDLEEXTPROC)(GLuint memory, GLuint64 size, GLenum handleType, void *handle);
GLAPI PFNGLIMPORTMEMORYWIN32HANDLEEXTPROC glad_glImportMemoryWin32HandleEXT;
#define glImportMemoryWin32HandleEXT glad_glImportMemoryWin32HandleEXT
typedef void (APIENTRYP PFNGLIMPORTMEMORYWIN32NAMEEXTPROC)(GLuint memory, GLuint64 size, GLenum handleType, const void *name);
GLAPI PFNGLIMPORTMEMORYWIN32NAMEEXTPROC glad_glImportMemoryWin32NameEXT;
#define glImportMemoryWin32NameEXT glad_glImportMemoryWin32NameEXT
#endif
#ifndef GL_EXT_misc_attribute
#define GL_EXT_misc_attribute 1
GLAPI int GLAD_GL_EXT_misc_attribute;
#endif
#ifndef GL_EXT_multi_draw_arrays
#define GL_EXT_multi_draw_arrays 1
GLAPI int GLAD_GL_EXT_multi_draw_arrays;
typedef void (APIENTRYP PFNGLMULTIDRAWARRAYSEXTPROC)(GLenum mode, const GLint *first, const GLsizei *count, GLsizei primcount);
GLAPI PFNGLMULTIDRAWARRAYSEXTPROC glad_glMultiDrawArraysEXT;
#define glMultiDrawArraysEXT glad_glMultiDrawArraysEXT
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSEXTPROC)(GLenum mode, const GLsizei *count, GLenum type, const void *const*indices, GLsizei primcount);
GLAPI PFNGLMULTIDRAWELEMENTSEXTPROC glad_glMultiDrawElementsEXT;
#define glMultiDrawElementsEXT glad_glMultiDrawElementsEXT
#endif
#ifndef GL_EXT_multisample
#define GL_EXT_multisample 1
GLAPI int GLAD_GL_EXT_multisample;
typedef void (APIENTRYP PFNGLSAMPLEMASKEXTPROC)(GLclampf value, GLboolean invert);
GLAPI PFNGLSAMPLEMASKEXTPROC glad_glSampleMaskEXT;
#define glSampleMaskEXT glad_glSampleMaskEXT
typedef void (APIENTRYP PFNGLSAMPLEPATTERNEXTPROC)(GLenum pattern);
GLAPI PFNGLSAMPLEPATTERNEXTPROC glad_glSamplePatternEXT;
#define glSamplePatternEXT glad_glSamplePatternEXT
#endif
#ifndef GL_EXT_multiview_tessellation_geometry_shader
#define GL_EXT_multiview_tessellation_geometry_shader 1
GLAPI int GLAD_GL_EXT_multiview_tessellation_geometry_shader;
#endif
#ifndef GL_EXT_multiview_texture_multisample
#define GL_EXT_multiview_texture_multisample 1
GLAPI int GLAD_GL_EXT_multiview_texture_multisample;
#endif
#ifndef GL_EXT_multiview_timer_query
#define GL_EXT_multiview_timer_query 1
GLAPI int GLAD_GL_EXT_multiview_timer_query;
#endif
#ifndef GL_EXT_packed_depth_stencil
#define GL_EXT_packed_depth_stencil 1
GLAPI int GLAD_GL_EXT_packed_depth_stencil;
#endif
#ifndef GL_EXT_packed_float
#define GL_EXT_packed_float 1
```

```
GLAPI int GLAD_GL_EXT_packed_float;
#endif
#ifndef GL_EXT_packed_pixels
#define GL_EXT_packed_pixels 1
GLAPI int GLAD_GL_EXT_packed_pixels;
#endif
#ifndef GL_EXT_paletted_texture
#define GL_EXT_paletted_texture 1
GLAPI int GLAD_GL_EXT_paletted_texture;
typedef void (APIENTRYP PFNGLCOLORTABLEEXTPROC)(GLenum target, GLenum internalFormat, GLsizei width, GLenum format, GLenum type, const void *table);
GLAPI PFNGLCOLORTABLEEXTPROC glad_glColorTableEXT;
#define glColorTableEXT glad_glColorTableEXT
typedef void (APIENTRYP PFNGLGETCOLORTABLEEXTPROC)(GLenum target, GLenum format, GLenum type, void *data);
GLAPI PFNGLGETCOLORTABLEEXTPROC glad_glGetColorTableEXT;
#define glGetColorTableEXT glad_glGetColorTableEXT
typedef void (APIENTRYP PFNGLGETCOLORTABLEPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETCOLORTABLEPARAMETERIVEXTPROC glad_glGetColorTableParameterivEXT;
#define glGetColorTableParameterivEXT glad_glGetColorTableParameterivEXT
typedef void (APIENTRYP PFNGLGETCOLORTABLEPARAMETERFVEXTPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCOLORTABLEPARAMETERFVEXTPROC glad_glGetColorTableParameterfvEXT;
#define glGetColorTableParameterfvEXT glad_glGetColorTableParameterfvEXT
#endif
#ifndef GL_EXT_pixel_buffer_object
#define GL_EXT_pixel_buffer_object 1
GLAPI int GLAD_GL_EXT_pixel_buffer_object;
#endif
#ifndef GL_EXT_pixel_transform
#define GL_EXT_pixel_transform 1
GLAPI int GLAD_GL_EXT_pixel_transform;
typedef void (APIENTRYP PFNGLPIXELTRANSFORMPARAMETERIEXTPROC)(GLenum target, GLenum pname, GLint param);
GLAPI PFNGLPIXELTRANSFORMPARAMETERIEXTPROC glad_glPixelTransformParameteriEXT;
#define glPixelTransformParameteriEXT glad_glPixelTransformParameteriEXT
typedef void (APIENTRYP PFNGLPIXELTRANSFORMPARAMETERFEXTPROC)(GLenum target, GLenum pname, GLfloat param);
GLAPI PFNGLPIXELTRANSFORMPARAMETERFEXTPROC glad_glPixelTransformParameterfEXT;
#define glPixelTransformParameterfEXT glad_glPixelTransformParameterfEXT
typedef void (APIENTRYP PFNGLPIXELTRANSFORMPARAMETERIVEXTPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLPIXELTRANSFORMPARAMETERIVEXTPROC glad_glPixelTransformParameterivEXT;
#define glPixelTransformParameterivEXT glad_glPixelTransformParameterivEXT
typedef void (APIENTRYP PFNGLPIXELTRANSFORMPARAMETERFVEXTPROC)(GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLPIXELTRANSFORMPARAMETERFVEXTPROC glad_glPixelTransformParameterfvEXT;
#define glPixelTransformParameterfvEXT glad_glPixelTransformParameterfvEXT
typedef void (APIENTRYP PFNGLGETPIXELTRANSFORMPARAMETERIVEXTPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETPIXELTRANSFORMPARAMETERIVEXTPROC glad_glGetPixelTransformParameterivEXT;
#define glGetPixelTransformParameterivEXT glad_glGetPixelTransformParameterivEXT
typedef void (APIENTRYP PFNGLGETPIXELTRANSFORMPARAMETERFVEXTPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETPIXELTRANSFORMPARAMETERFVEXTPROC glad_glGetPixelTransformParameterfvEXT;
#define glGetPixelTransformParameterfvEXT glad_glGetPixelTransformParameterfvEXT
#endif
#ifndef GL_EXT_pixel_transform_color_table
#define GL_EXT_pixel_transform_color_table 1
GLAPI int GLAD_GL_EXT_pixel_transform_color_table;
#endif
#ifndef GL_EXT_point_parameters
#define GL_EXT_point_parameters 1
GLAPI int GLAD_GL_EXT_point_parameters;
typedef void (APIENTRYP PFNGLPOINTPARAMETERFEXTPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLPOINTPARAMETERFEXTPROC glad_glPointParameterfEXT;
#define glPointParameterfEXT glad_glPointParameterfEXT
typedef void (APIENTRYP PFNGLPOINTPARAMETERFVEXTPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLPOINTPARAMETERFVEXTPROC glad_glPointParameterfvEXT;
#define glPointParameterfvEXT glad_glPointParameterfvEXT
#endif
#ifndef GL_EXT_polygon_offset
#define GL_EXT_polygon_offset 1
GLAPI int GLAD_GL_EXT_polygon_offset;
typedef void (APIENTRYP PFNGLPOLYGONOFFSETEXTPROC)(GLfloat factor, GLfloat bias);
GLAPI PFNGLPOLYGONOFFSETEXTPROC glad_glPolygonOffsetEXT;
#define glPolygonOffsetEXT glad_glPolygonOffsetEXT
#endif
#ifndef GL_EXT_polygon_offset_clamp
#define GL_EXT_polygon_offset_clamp 1
GLAPI int GLAD_GL_EXT_polygon_offset_clamp;
typedef void (APIENTRYP PFNGLPOLYGONOFFSETCLAMPEXTPROC)(GLfloat factor, GLfloat units, GLfloat clamp);
GLAPI PFNGLPOLYGONOFFSETCLAMPEXTPROC glad_glPolygonOffsetClampEXT;
#define glPolygonOffsetClampEXT glad_glPolygonOffsetClampEXT
#endif
#ifndef GL_EXT_post_depth_coverage
```

```
#define GL_EXT_post_depth_coverage 1
GLAPI int GLAD_GL_EXT_post_depth_coverage;
#endif
#ifndef GL_EXT_provoking_vertex
#define GL_EXT_provoking_vertex 1
GLAPI int GLAD_GL_EXT_provoking_vertex;
typedef void (APIENTRYP PFNGLPROVOKINGVERTEXEXTPROC)(GLenum mode);
GLAPI PFNGLPROVOKINGVERTEXEXTPROC glad_glProvokingVertexEXT;
#define glProvokingVertexEXT glad_glProvokingVertexEXT
#endif
#ifndef GL_EXT_raster_multisample
#define GL_EXT_raster_multisample 1
GLAPI int GLAD_GL_EXT_raster_multisample;
typedef void (APIENTRYP PFNGLRASTERSAMPLESEXTPROC)(GLuint samples, GLboolean fixedsamplelocations);
GLAPI PFNGLRASTERSAMPLESEXTPROC glad_glRasterSamplesEXT;
#define glRasterSamplesEXT glad_glRasterSamplesEXT
#endif
#ifndef GL_EXT_rescale_normal
#define GL_EXT_rescale_normal 1
GLAPI int GLAD_GL_EXT_rescale_normal;
#endif
#ifndef GL_EXT_secondary_color
#define GL_EXT_secondary_color 1
GLAPI int GLAD_GL_EXT_secondary_color;
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3BEXTPROC)(GLbyte red, GLbyte green, GLbyte blue);
GLAPI PFNGLSECONDARYCOLOR3BEXTPROC glad_glSecondaryColor3bEXT;
#define glSecondaryColor3bEXT glad_glSecondaryColor3bEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3BVEXTPROC)(const GLbyte *v);
GLAPI PFNGLSECONDARYCOLOR3BVEXTPROC glad_glSecondaryColor3bvEXT;
#define glSecondaryColor3bvEXT glad_glSecondaryColor3bvEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3DEXTPROC)(GLdouble red, GLdouble green, GLdouble blue);
GLAPI PFNGLSECONDARYCOLOR3DEXTPROC glad_glSecondaryColor3dEXT;
#define glSecondaryColor3dEXT glad_glSecondaryColor3dEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3DVEXTPROC)(const GLdouble *v);
GLAPI PFNGLSECONDARYCOLOR3DVEXTPROC glad_glSecondaryColor3dvEXT;
#define glSecondaryColor3dvEXT glad_glSecondaryColor3dvEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3FEXTPROC)(GLfloat red, GLfloat green, GLfloat blue);
GLAPI PFNGLSECONDARYCOLOR3FEXTPROC glad_glSecondaryColor3fEXT;
#define glSecondaryColor3fEXT glad_glSecondaryColor3fEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3FVEXTPROC)(const GLfloat *v);
GLAPI PFNGLSECONDARYCOLOR3FVEXTPROC glad_glSecondaryColor3fvEXT;
#define glSecondaryColor3fvEXT glad_glSecondaryColor3fvEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3IEXTPROC)(GLint red, GLint green, GLint blue);
GLAPI PFNGLSECONDARYCOLOR3IEXTPROC glad_glSecondaryColor3iEXT;
#define glSecondaryColor3iEXT glad_glSecondaryColor3iEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3IVEXTPROC)(const GLint *v);
GLAPI PFNGLSECONDARYCOLOR3IVEXTPROC glad_glSecondaryColor3ivEXT;
#define glSecondaryColor3ivEXT glad_glSecondaryColor3ivEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3SEXTPROC)(GLshort red, GLshort green, GLshort blue);
GLAPI PFNGLSECONDARYCOLOR3SEXTPROC glad_glSecondaryColor3sEXT;
#define glSecondaryColor3sEXT glad_glSecondaryColor3sEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3SVEXTPROC)(const GLshort *v);
GLAPI PFNGLSECONDARYCOLOR3SVEXTPROC glad_glSecondaryColor3svEXT;
#define glSecondaryColor3svEXT glad_glSecondaryColor3svEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3UBEXTPROC)(GLubyte red, GLubyte green, GLubyte blue);
GLAPI PFNGLSECONDARYCOLOR3UBEXTPROC glad_glSecondaryColor3ubEXT;
#define glSecondaryColor3ubEXT glad_glSecondaryColor3ubEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3UBVEXTPROC)(const GLubyte *v);
GLAPI PFNGLSECONDARYCOLOR3UBVEXTPROC glad_glSecondaryColor3ubvEXT;
#define glSecondaryColor3ubvEXT glad_glSecondaryColor3ubvEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3UIEXTPROC)(GLuint red, GLuint green, GLuint blue);
GLAPI PFNGLSECONDARYCOLOR3UIEXTPROC glad_glSecondaryColor3uiEXT;
#define glSecondaryColor3uiEXT glad_glSecondaryColor3uiEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3UIVEXTPROC)(const GLuint *v);
GLAPI PFNGLSECONDARYCOLOR3UIVEXTPROC glad_glSecondaryColor3uivEXT;
#define glSecondaryColor3uivEXT glad_glSecondaryColor3uivEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3USEXTPROC)(GLushort red, GLushort green, GLushort blue);
GLAPI PFNGLSECONDARYCOLOR3USEXTPROC glad_glSecondaryColor3usEXT;
#define glSecondaryColor3usEXT glad_glSecondaryColor3usEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3USVEXTPROC)(const GLushort *v);
GLAPI PFNGLSECONDARYCOLOR3USVEXTPROC glad_glSecondaryColor3usvEXT;
#define glSecondaryColor3usvEXT glad_glSecondaryColor3usvEXT
typedef void (APIENTRYP PFNGLSECONDARYCOLORPOINTEREXTPROC)(GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLSECONDARYCOLORPOINTEREXTPROC glad_glSecondaryColorPointerEXT;
#define glSecondaryColorPointerEXT glad_glSecondaryColorPointerEXT
#endif
#ifndef GL_EXT_semaphore
#define GL_EXT_semaphore 1
GLAPI int GLAD_GL_EXT_semaphore;
typedef void (APIENTRYP PFNGLGENSEMAPHORESEXTPROC)(GLsizei n, GLuint *semaphores);
GLAPI PFNGLGENSEMAPHORESEXTPROC glad_glGenSemaphoresEXT;
#define glGenSemaphoresEXT glad_glGenSemaphoresEXT
typedef void (APIENTRYP PFNGLDELETESEMAPHORESEXTPROC)(GLsizei n, const GLuint *semaphores);
GLAPI PFNGLDELETESEMAPHORESEXTPROC glad_glDeleteSemaphoresEXT;
#define glDeleteSemaphoresEXT glad_glDeleteSemaphoresEXT
typedef GLboolean (APIENTRYP PFNGLISSEMAPHOREEXTPROC)(GLuint semaphore);
```

```
GLAPI PFNGLISSEMAPHOREEXTPROC glad_glIsSemaphoreEXT;
#define glIsSemaphoreEXT glad_glIsSemaphoreEXT
typedef void (APIENTRYP PFNGLSEMAPHOREPARAMETERUI64VEXTPROC)(GLuint semaphore, GLenum pname, const GLuint64 *params);
GLAPI PFNGLSEMAPHOREPARAMETERUI64VEXTPROC glad_glSemaphoreParameterui64vEXT;
#define glSemaphoreParameterui64vEXT glad_glSemaphoreParameterui64vEXT
typedef void (APIENTRYP PFNGLGETSEMAPHOREPARAMETERUI64VEXTPROC)(GLuint semaphore, GLenum pname, GLuint64 *params);
GLAPI PFNGLGETSEMAPHOREPARAMETERUI64VEXTPROC glad_glGetSemaphoreParameterui64vEXT;
#define glGetSemaphoreParameterui64vEXT glad_glGetSemaphoreParameterui64vEXT
typedef void (APIENTRYP PFNGLWAITSEMAPHOREEXTPROC)(GLuint semaphore, GLuint numBufferBarriers, const GLuint *buffers, GLuint numTextureBarriers, const GLuint *textures, const GLenum *srcLayouts);
GLAPI PFNGLWAITSEMAPHOREEXTPROC glad_glWaitSemaphoreEXT;
#define glWaitSemaphoreEXT glad_glWaitSemaphoreEXT
typedef void (APIENTRYP PFNGLSIGNALSEMAPHOREEXTPROC)(GLuint semaphore, GLuint numBufferBarriers, const GLuint *buffers, GLuint numTextureBarriers, const GLuint *textures, const GLenum *dstLayouts);
GLAPI PFNGLSIGNALSEMAPHOREEXTPROC glad_glSignalSemaphoreEXT;
#define glSignalSemaphoreEXT glad_glSignalSemaphoreEXT
#endif
#ifndef GL_EXT_semaphore_fd
#define GL_EXT_semaphore_fd 1
GLAPI int GLAD_GL_EXT_semaphore_fd;
typedef void (APIENTRYP PFNGLIMPORTSEMAPHOREFDEXTPROC)(GLuint semaphore, GLenum handleType, GLint fd);
GLAPI PFNGLIMPORTSEMAPHOREFDEXTPROC glad_glImportSemaphoreFdEXT;
#define glImportSemaphoreFdEXT glad_glImportSemaphoreFdEXT
#endif
#ifndef GL_EXT_semaphore_win32
#define GL_EXT_semaphore_win32 1
GLAPI int GLAD_GL_EXT_semaphore_win32;
typedef void (APIENTRYP PFNGLIMPORTSEMAPHOREWIN32HANDLEEXTPROC)(GLuint semaphore, GLenum handleType, void *handle);
GLAPI PFNGLIMPORTSEMAPHOREWIN32HANDLEEXTPROC glad_glImportSemaphoreWin32HandleEXT;
#define glImportSemaphoreWin32HandleEXT glad_glImportSemaphoreWin32HandleEXT
typedef void (APIENTRYP PFNGLIMPORTSEMAPHOREWIN32NAMEEXTPROC)(GLuint semaphore, GLenum handleType, const void *name);
GLAPI PFNGLIMPORTSEMAPHOREWIN32NAMEEXTPROC glad_glImportSemaphoreWin32NameEXT;
#define glImportSemaphoreWin32NameEXT glad_glImportSemaphoreWin32NameEXT
#endif
#ifndef GL_EXT_separate_shader_objects
#define GL_EXT_separate_shader_objects 1
GLAPI int GLAD_GL_EXT_separate_shader_objects;
typedef void (APIENTRYP PFNGLUSESHADERPROGRAMEXTPROC)(GLenum type, GLuint program);
GLAPI PFNGLUSESHADERPROGRAMEXTPROC glad_glUseShaderProgramEXT;
#define glUseShaderProgramEXT glad_glUseShaderProgramEXT
typedef void (APIENTRYP PFNGLACTIVEPROGRAMEXTPROC)(GLuint program);
GLAPI PFNGLACTIVEPROGRAMEXTPROC glad_glActiveProgramEXT;
#define glActiveProgramEXT glad_glActiveProgramEXT
typedef GLuint (APIENTRYP PFNGLCREATESHADERPROGRAMEXTPROC)(GLenum type, const GLchar *string);
GLAPI PFNGLCREATESHADERPROGRAMEXTPROC glad_glCreateShaderProgramEXT;
#define glCreateShaderProgramEXT glad_glCreateShaderProgramEXT
typedef void (APIENTRYP PFNGLACTIVESHADERPROGRAMEXTPROC)(GLuint pipeline, GLuint program);
GLAPI PFNGLACTIVESHADERPROGRAMEXTPROC glad_glActiveShaderProgramEXT;
#define glActiveShaderProgramEXT glad_glActiveShaderProgramEXT
typedef void (APIENTRYP PFNGLBINDPROGRAMPIPELINEEXTPROC)(GLuint pipeline);
GLAPI PFNGLBINDPROGRAMPIPELINEEXTPROC glad_glBindProgramPipelineEXT;
#define glBindProgramPipelineEXT glad_glBindProgramPipelineEXT
typedef GLuint (APIENTRYP PFNGLCREATESHADERPROGRAMVEXTPROC)(GLenum type, GLsizei count, const GLchar *const*strings);
GLAPI PFNGLCREATESHADERPROGRAMVEXTPROC glad_glCreateShaderProgramvEXT;
#define glCreateShaderProgramvEXT glad_glCreateShaderProgramvEXT
typedef void (APIENTRYP PFNGLDELETEPROGRAMPIPELINESEXTPROC)(GLsizei n, const GLuint *pipelines);
GLAPI PFNGLDELETEPROGRAMPIPELINESEXTPROC glad_glDeleteProgramPipelinesEXT;
#define glDeleteProgramPipelinesEXT glad_glDeleteProgramPipelinesEXT
typedef void (APIENTRYP PFNGLGENPROGRAMPIPELINESEXTPROC)(GLsizei n, GLuint *pipelines);
GLAPI PFNGLGENPROGRAMPIPELINESEXTPROC glad_glGenProgramPipelinesEXT;
#define glGenProgramPipelinesEXT glad_glGenProgramPipelinesEXT
typedef void (APIENTRYP PFNGLGETPROGRAMPIPELINEINFOLOGEXTPROC)(GLuint pipeline, GLsizei bufSize, GLsizei *length, GLchar *infoLog);
GLAPI PFNGLGETPROGRAMPIPELINEINFOLOGEXTPROC glad_glGetProgramPipelineInfoLogEXT;
#define glGetProgramPipelineInfoLogEXT glad_glGetProgramPipelineInfoLogEXT
typedef void (APIENTRYP PFNGLGETPROGRAMPIPELINEIVEXTPROC)(GLuint pipeline, GLenum pname, GLint *params);
GLAPI PFNGLGETPROGRAMPIPELINEIVEXTPROC glad_glGetProgramPipelineivEXT;
#define glGetProgramPipelineivEXT glad_glGetProgramPipelineivEXT
typedef GLboolean (APIENTRYP PFNGLISPROGRAMPIPELINEEXTPROC)(GLuint pipeline);
GLAPI PFNGLISPROGRAMPIPELINEEXTPROC glad_glIsProgramPipelineEXT;
#define glIsProgramPipelineEXT glad_glIsProgramPipelineEXT
typedef void (APIENTRYP PFNGLUSEPROGRAMSTAGESEXTPROC)(GLuint pipeline, GLbitfield stages, GLuint program);
GLAPI PFNGLUSEPROGRAMSTAGESEXTPROC glad_glUseProgramStagesEXT;
#define glUseProgramStagesEXT glad_glUseProgramStagesEXT
typedef void (APIENTRYP PFNGLVALIDATEPROGRAMPIPELINEEXTPROC)(GLuint pipeline);
GLAPI PFNGLVALIDATEPROGRAMPIPELINEEXTPROC glad_glValidateProgramPipelineEXT;
#define glValidateProgramPipelineEXT glad_glValidateProgramPipelineEXT
#endif
#ifndef GL_EXT_separate_specular_color
#define GL_EXT_separate_specular_color 1
```

```
GLAPI int GLAD_GL_EXT_separate_specular_color;
#endif
#ifndef GL_EXT_shader_framebuffer_fetch
#define GL_EXT_shader_framebuffer_fetch 1
GLAPI int GLAD_GL_EXT_shader_framebuffer_fetch;
#endif
#ifndef GL_EXT_shader_framebuffer_fetch_non_coherent
#define GL_EXT_shader_framebuffer_fetch_non_coherent 1
GLAPI int GLAD_GL_EXT_shader_framebuffer_fetch_non_coherent;
typedef void (APIENTRYP PFNGLFRAMEBUFFERFETCHBARRIEREXTPROC)(void);
GLAPI PFNGLFRAMEBUFFERFETCHBARRIEREXTPROC glad_glFramebufferFetchBarrierEXT;
#define glFramebufferFetchBarrierEXT glad_glFramebufferFetchBarrierEXT
#endif
#ifndef GL_EXT_shader_image_load_formatted
#define GL_EXT_shader_image_load_formatted 1
GLAPI int GLAD_GL_EXT_shader_image_load_formatted;
#endif
#ifndef GL_EXT_shader_image_load_store
#define GL_EXT_shader_image_load_store 1
GLAPI int GLAD_GL_EXT_shader_image_load_store;
typedef void (APIENTRYP PFNGLBINDIMAGETEXTUREEXTPROC)(GLuint index, GLuint texture, GLint level, GLboolean layered, GLint layer, GLenum access, GLint format);
GLAPI PFNGLBINDIMAGETEXTUREEXTPROC glad_glBindImageTextureEXT;
#define glBindImageTextureEXT glad_glBindImageTextureEXT
typedef void (APIENTRYP PFNGLMEMORYBARRIEREXTPROC)(GLbitfield barriers);
GLAPI PFNGLMEMORYBARRIEREXTPROC glad_glMemoryBarrierEXT;
#define glMemoryBarrierEXT glad_glMemoryBarrierEXT
#endif
#ifndef GL_EXT_shader_integer_mix
#define GL_EXT_shader_integer_mix 1
GLAPI int GLAD_GL_EXT_shader_integer_mix;
#endif
#ifndef GL_EXT_shader_samples_identical
#define GL_EXT_shader_samples_identical 1
GLAPI int GLAD_GL_EXT_shader_samples_identical;
#endif
#ifndef GL_EXT_shadow_funcs
#define GL_EXT_shadow_funcs 1
GLAPI int GLAD_GL_EXT_shadow_funcs;
#endif
#ifndef GL_EXT_shared_texture_palette
#define GL_EXT_shared_texture_palette 1
GLAPI int GLAD_GL_EXT_shared_texture_palette;
#endif
#ifndef GL_EXT_sparse_texture2
#define GL_EXT_sparse_texture2 1
GLAPI int GLAD_GL_EXT_sparse_texture2;
#endif
#ifndef GL_EXT_stencil_clear_tag
#define GL_EXT_stencil_clear_tag 1
GLAPI int GLAD_GL_EXT_stencil_clear_tag;
typedef void (APIENTRYP PFNGLSTENCILCLEARTAGEXTPROC)(GLsizei stencilTagBits, GLuint stencilClearTag);
GLAPI PFNGLSTENCILCLEARTAGEXTPROC glad_glStencilClearTagEXT;
#define glStencilClearTagEXT glad_glStencilClearTagEXT
#endif
#ifndef GL_EXT_stencil_two_side
#define GL_EXT_stencil_two_side 1
GLAPI int GLAD_GL_EXT_stencil_two_side;
typedef void (APIENTRYP PFNGLACTIVESTENCILFACEEXTPROC)(GLenum face);
GLAPI PFNGLACTIVESTENCILFACEEXTPROC glad_glActiveStencilFaceEXT;
#define glActiveStencilFaceEXT glad_glActiveStencilFaceEXT
#endif
#ifndef GL_EXT_stencil_wrap
#define GL_EXT_stencil_wrap 1
GLAPI int GLAD_GL_EXT_stencil_wrap;
#endif
#ifndef GL_EXT_subtexture
#define GL_EXT_subtexture 1
GLAPI int GLAD_GL_EXT_subtexture;
typedef void (APIENTRYP PFNGLTEXSUBIMAGE1DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLsizei width, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXSUBIMAGE1DEXTPROC glad_glTexSubImage1DEXT;
#define glTexSubImage1DEXT glad_glTexSubImage1DEXT
typedef void (APIENTRYP PFNGLTEXSUBIMAGE2DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLsizei width, GLsizei height, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXSUBIMAGE2DEXTPROC glad_glTexSubImage2DEXT;
#define glTexSubImage2DEXT glad_glTexSubImage2DEXT
#endif
#ifndef GL_EXT_texture
#define GL_EXT_texture 1
GLAPI int GLAD_GL_EXT_texture;
#endif
#ifndef GL_EXT_texture3D
#define GL_EXT_texture3D 1
GLAPI int GLAD_GL_EXT_texture3D;
typedef void (APIENTRYP PFNGLTEXIMAGE3DEXTPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLint border, GLenum format, GLenum type, const void
*pixels);
GLAPI PFNGLTEXIMAGE3DEXTPROC glad_glTexImage3DEXT;
#define glTexImage3DEXT glad_glTexImage3DEXT
typedef void (APIENTRYP PFNGLTEXSUBIMAGE3DEXTPROC)(GLenum target, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXSUBIMAGE3DEXTPROC glad_glTexSubImage3DEXT;
#define glTexSubImage3DEXT glad_glTexSubImage3DEXT
#endif
#ifndef GL_EXT_texture_array
#define GL_EXT_texture_array 1
GLAPI int GLAD_GL_EXT_texture_array;
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTURELAYEREXTPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level, GLint layer);
GLAPI PFNGLFRAMEBUFFERTEXTURELAYEREXTPROC glad_glFramebufferTextureLayerEXT;
#define glFramebufferTextureLayerEXT glad_glFramebufferTextureLayerEXT
#endif
#ifndef GL_EXT_texture_buffer_object
#define GL_EXT_texture_buffer_object 1
GLAPI int GLAD_GL_EXT_texture_buffer_object;
typedef void (APIENTRYP PFNGLTEXBUFFEREXTPROC)(GLenum target, GLenum internalformat, GLuint buffer);
GLAPI PFNGLTEXBUFFEREXTPROC glad_glTexBufferEXT;
#define glTexBufferEXT glad_glTexBufferEXT
#endif
#ifndef GL_EXT_texture_compression_latc
#define GL_EXT_texture_compression_latc 1
GLAPI int GLAD_GL_EXT_texture_compression_latc;
#endif
#ifndef GL_EXT_texture_compression_rgtc
#define GL_EXT_texture_compression_rgtc 1
GLAPI int GLAD_GL_EXT_texture_compression_rgtc;
#endif
#ifndef GL_EXT_texture_compression_s3tc
#define GL_EXT_texture_compression_s3tc 1
GLAPI int GLAD_GL_EXT_texture_compression_s3tc;
#endif
#ifndef GL_EXT_texture_cube_map
#define GL_EXT_texture_cube_map 1
GLAPI int GLAD_GL_EXT_texture_cube_map;
#endif
#ifndef GL_EXT_texture_env_add
#define GL_EXT_texture_env_add 1
GLAPI int GLAD_GL_EXT_texture_env_add;
#endif
#ifndef GL_EXT_texture_env_combine
#define GL_EXT_texture_env_combine 1
GLAPI int GLAD_GL_EXT_texture_env_combine;
#endif
#ifndef GL_EXT_texture_env_dot3
#define GL_EXT_texture_env_dot3 1
GLAPI int GLAD_GL_EXT_texture_env_dot3;
#endif
#ifndef GL_EXT_texture_filter_anisotropic
#define GL_EXT_texture_filter_anisotropic 1
GLAPI int GLAD_GL_EXT_texture_filter_anisotropic;
#endif
#ifndef GL_EXT_texture_filter_minmax
#define GL_EXT_texture_filter_minmax 1
GLAPI int GLAD_GL_EXT_texture_filter_minmax;
#endif
#ifndef GL_EXT_texture_integer
#define GL_EXT_texture_integer 1
GLAPI int GLAD_GL_EXT_texture_integer;
typedef void (APIENTRYP PFNGLTEXPARAMETERIIVEXTPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLTEXPARAMETERIIVEXTPROC glad_glTexParameterIivEXT;
#define glTexParameterIivEXT glad_glTexParameterIivEXT
typedef void (APIENTRYP PFNGLTEXPARAMETERIUIVEXTPROC)(GLenum target, GLenum pname, const GLuint *params);
GLAPI PFNGLTEXPARAMETERIUIVEXTPROC glad_glTexParameterIuivEXT;
#define glTexParameterIuivEXT glad_glTexParameterIuivEXT
typedef void (APIENTRYP PFNGLGETTEXPARAMETERIIVEXTPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETTEXPARAMETERIIVEXTPROC glad_glGetTexParameterIivEXT;
#define glGetTexParameterIivEXT glad_glGetTexParameterIivEXT
typedef void (APIENTRYP PFNGLGETTEXPARAMETERIUIVEXTPROC)(GLenum target, GLenum pname, GLuint *params);
GLAPI PFNGLGETTEXPARAMETERIUIVEXTPROC glad_glGetTexParameterIuivEXT;
#define glGetTexParameterIuivEXT glad_glGetTexParameterIuivEXT
typedef void (APIENTRYP PFNGLCLEARCOLORIIEXTPROC)(GLint red, GLint green, GLint blue, GLint alpha);
GLAPI PFNGLCLEARCOLORIIEXTPROC glad_glClearColorIiEXT;
#define glClearColorIiEXT glad_glClearColorIiEXT
typedef void (APIENTRYP PFNGLCLEARCOLORIUIEXTPROC)(GLuint red, GLuint green, GLuint blue, GLuint alpha);
GLAPI PFNGLCLEARCOLORIUIEXTPROC glad_glClearColorIuiEXT;
#define glClearColorIuiEXT glad_glClearColorIuiEXT
#endif
#ifndef GL_EXT_texture_lod_bias
#define GL_EXT_texture_lod_bias 1
```

```
GLAPI int GLAD_GL_EXT_texture_lod_bias;
#endif
#ifndef GL_EXT_texture_mirror_clamp
#define GL_EXT_texture_mirror_clamp 1
GLAPI int GLAD_GL_EXT_texture_mirror_clamp;
#endif
#ifndef GL_EXT_texture_object
#define GL_EXT_texture_object 1
GLAPI int GLAD_GL_EXT_texture_object;
typedef GLboolean (APIENTRYP PFNGLARETEXTURESRESIDENTEXTPROC)(GLsizei n, const GLuint *textures, GLboolean *residences);
GLAPI PFNGLARETEXTURESRESIDENTEXTPROC glad_glAreTexturesResidentEXT;
#define glAreTexturesResidentEXT glad_glAreTexturesResidentEXT
typedef void (APIENTRYP PFNGLBINDTEXTUREEXTPROC)(GLenum target, GLuint texture);
GLAPI PFNGLBINDTEXTUREEXTPROC glad_glBindTextureEXT;
#define glBindTextureEXT glad_glBindTextureEXT
typedef void (APIENTRYP PFNGLDELETETEXTURESEXTPROC)(GLsizei n, const GLuint *textures);
GLAPI PFNGLDELETETEXTURESEXTPROC glad_glDeleteTexturesEXT;
#define glDeleteTexturesEXT glad_glDeleteTexturesEXT
typedef void (APIENTRYP PFNGLGENTEXTURESEXTPROC)(GLsizei n, GLuint *textures);
GLAPI PFNGLGENTEXTURESEXTPROC glad_glGenTexturesEXT;
#define glGenTexturesEXT glad_glGenTexturesEXT
typedef GLboolean (APIENTRYP PFNGLISTEXTUREEXTPROC)(GLuint texture);
GLAPI PFNGLISTEXTUREEXTPROC glad_glIsTextureEXT;
#define glIsTextureEXT glad_glIsTextureEXT
typedef void (APIENTRYP PFNGLPRIORITIZETEXTURESEXTPROC)(GLsizei n, const GLuint *textures, const GLclampf *priorities);
GLAPI PFNGLPRIORITIZETEXTURESEXTPROC glad_glPrioritizeTexturesEXT;
#define glPrioritizeTexturesEXT glad_glPrioritizeTexturesEXT
#endif
#ifndef GL_EXT_texture_perturb_normal
#define GL_EXT_texture_perturb_normal 1
GLAPI int GLAD_GL_EXT_texture_perturb_normal;
typedef void (APIENTRYP PFNGLTEXTURENORMALEXTPROC)(GLenum mode);
GLAPI PFNGLTEXTURENORMALEXTPROC glad_glTextureNormalEXT;
#define glTextureNormalEXT glad_glTextureNormalEXT
#endif
#ifndef GL_EXT_texture_sRGB
#define GL_EXT_texture_sRGB 1
GLAPI int GLAD_GL_EXT_texture_sRGB;
#endif
#ifndef GL_EXT_texture_sRGB_R8
#define GL_EXT_texture_sRGB_R8 1
GLAPI int GLAD_GL_EXT_texture_sRGB_R8;
#endif
#ifndef GL_EXT_texture_sRGB_RG8
#define GL_EXT_texture_sRGB_RG8 1
GLAPI int GLAD_GL_EXT_texture_sRGB_RG8;
#endif
#ifndef GL_EXT_texture_sRGB_decode
#define GL_EXT_texture_sRGB_decode 1
GLAPI int GLAD_GL_EXT_texture_sRGB_decode;
#endif
#ifndef GL_EXT_texture_shadow_lod
#define GL_EXT_texture_shadow_lod 1
GLAPI int GLAD_GL_EXT_texture_shadow_lod;
#endif
#ifndef GL_EXT_texture_shared_exponent
#define GL_EXT_texture_shared_exponent 1
GLAPI int GLAD_GL_EXT_texture_shared_exponent;
#endif
#ifndef GL_EXT_texture_snorm
#define GL_EXT_texture_snorm 1
GLAPI int GLAD_GL_EXT_texture_snorm;
#endif
#ifndef GL_EXT_texture_storage
#define GL_EXT_texture_storage 1
GLAPI int GLAD_GL_EXT_texture_storage;
typedef void (APIENTRYP PFNGLTEXSTORAGE1DEXTPROC)(GLenum target, GLsizei levels, GLenum internalformat, GLsizei width);
GLAPI PFNGLTEXSTORAGE1DEXTPROC glad_glTexStorage1DEXT;
#define glTexStorage1DEXT glad_glTexStorage1DEXT
typedef void (APIENTRYP PFNGLTEXSTORAGE2DEXTPROC)(GLenum target, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLTEXSTORAGE2DEXTPROC glad_glTexStorage2DEXT;
#define glTexStorage2DEXT glad_glTexStorage2DEXT
typedef void (APIENTRYP PFNGLTEXSTORAGE3DEXTPROC)(GLenum target, GLsizei levels, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth);
GLAPI PFNGLTEXSTORAGE3DEXTPROC glad_glTexStorage3DEXT;
#define glTexStorage3DEXT glad_glTexStorage3DEXT
#endif
#ifndef GL_EXT_texture_swizzle
#define GL_EXT_texture_swizzle 1
GLAPI int GLAD_GL_EXT_texture_swizzle;
#endif
#ifndef GL_EXT_timer_query
#define GL_EXT_timer_query 1
```

```
GLAPI int GLAD_GL_EXT_timer_query;
typedef void (APIENTRYP PFNGLGETQUERYOBJECTI64VEXTPROC)(GLuint id, GLenum pname, GLint64 *params);
GLAPI PFNGLGETQUERYOBJECTI64VEXTPROC glad_glGetQueryObjecti64vEXT;
#define glGetQueryObjecti64vEXT glad_glGetQueryObjecti64vEXT
typedef void (APIENTRYP PFNGLGETQUERYOBJECTUI64VEXTPROC)(GLuint id, GLenum pname, GLuint64 *params);
GLAPI PFNGLGETQUERYOBJECTUI64VEXTPROC glad_glGetQueryObjectui64vEXT;
#define glGetQueryObjectui64vEXT glad_glGetQueryObjectui64vEXT
#endif
#ifndef GL_EXT_transform_feedback
#define GL_EXT_transform_feedback 1
GLAPI int GLAD_GL_EXT_transform_feedback;
typedef void (APIENTRYP PFNGLBEGINTRANSFORMFEEDBACKEXTPROC)(GLenum primitiveMode);
GLAPI PFNGLBEGINTRANSFORMFEEDBACKEXTPROC glad_glBeginTransformFeedbackEXT;
#define glBeginTransformFeedbackEXT glad_glBeginTransformFeedbackEXT
typedef void (APIENTRYP PFNGLENDTRANSFORMFEEDBACKEXTPROC)(void);
GLAPI PFNGLENDTRANSFORMFEEDBACKEXTPROC glad_glEndTransformFeedbackEXT;
#define glEndTransformFeedbackEXT glad_glEndTransformFeedbackEXT
typedef void (APIENTRYP PFNGLBINDBUFFERRANGEEXTPROC)(GLenum target, GLuint index, GLuint buffer, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLBINDBUFFERRANGEEXTPROC glad_glBindBufferRangeEXT;
#define glBindBufferRangeEXT glad_glBindBufferRangeEXT
typedef void (APIENTRYP PFNGLBINDBUFFEROFFSETEXTPROC)(GLenum target, GLuint index, GLuint buffer, GLintptr offset);
GLAPI PFNGLBINDBUFFEROFFSETEXTPROC glad_glBindBufferOffsetEXT;
#define glBindBufferOffsetEXT glad_glBindBufferOffsetEXT
typedef void (APIENTRYP PFNGLBINDBUFFERBASEEXTPROC)(GLenum target, GLuint index, GLuint buffer);
GLAPI PFNGLBINDBUFFERBASEEXTPROC glad_glBindBufferBaseEXT;
#define glBindBufferBaseEXT glad_glBindBufferBaseEXT
typedef void (APIENTRYP PFNGLTRANSFORMFEEDBACKVARYINGSEXTPROC)(GLuint program, GLsizei count, const GLchar *const*varyings, GLenum bufferMode);
GLAPI PFNGLTRANSFORMFEEDBACKVARYINGSEXTPROC glad_glTransformFeedbackVaryingsEXT;
#define glTransformFeedbackVaryingsEXT glad_glTransformFeedbackVaryingsEXT
typedef void (APIENTRYP PFNGLGETTRANSFORMFEEDBACKVARYINGEXTPROC)(GLuint program, GLuint index, GLsizei bufSize, GLsizei *length, GLsizei *size, GLenum *type, GLchar *name);
GLAPI PFNGLGETTRANSFORMFEEDBACKVARYINGEXTPROC glad_glGetTransformFeedbackVaryingEXT;
#define glGetTransformFeedbackVaryingEXT glad_glGetTransformFeedbackVaryingEXT
#endif
#ifndef GL_EXT_vertex_array
#define GL_EXT_vertex_array 1
GLAPI int GLAD_GL_EXT_vertex_array;
typedef void (APIENTRYP PFNGLARRAYELEMENTEXTPROC)(GLint i);
GLAPI PFNGLARRAYELEMENTEXTPROC glad_glArrayElementEXT;
#define glArrayElementEXT glad_glArrayElementEXT
typedef void (APIENTRYP PFNGLCOLORPOINTEREXTPROC)(GLint size, GLenum type, GLsizei stride, GLsizei count, const void *pointer);
GLAPI PFNGLCOLORPOINTEREXTPROC glad_glColorPointerEXT;
#define glColorPointerEXT glad_glColorPointerEXT
typedef void (APIENTRYP PFNGLDRAWARRAYSEXTPROC)(GLenum mode, GLint first, GLsizei count);
GLAPI PFNGLDRAWARRAYSEXTPROC glad_glDrawArraysEXT;
#define glDrawArraysEXT glad_glDrawArraysEXT
typedef void (APIENTRYP PFNGLEDGEFLAGPOINTEREXTPROC)(GLsizei stride, GLsizei count, const GLboolean *pointer);
GLAPI PFNGLEDGEFLAGPOINTEREXTPROC glad_glEdgeFlagPointerEXT;
#define glEdgeFlagPointerEXT glad_glEdgeFlagPointerEXT
typedef void (APIENTRYP PFNGLGETPOINTERVEXTPROC)(GLenum pname, void **params);
GLAPI PFNGLGETPOINTERVEXTPROC glad_glGetPointervEXT;
#define glGetPointervEXT glad_glGetPointervEXT
typedef void (APIENTRYP PFNGLINDEXPOINTEREXTPROC)(GLenum type, GLsizei stride, GLsizei count, const void *pointer);
GLAPI PFNGLINDEXPOINTEREXTPROC glad_glIndexPointerEXT;
#define glIndexPointerEXT glad_glIndexPointerEXT
typedef void (APIENTRYP PFNGLNORMALPOINTEREXTPROC)(GLenum type, GLsizei stride, GLsizei count, const void *pointer);
GLAPI PFNGLNORMALPOINTEREXTPROC glad_glNormalPointerEXT;
#define glNormalPointerEXT glad_glNormalPointerEXT
typedef void (APIENTRYP PFNGLTEXCOORDPOINTEREXTPROC)(GLint size, GLenum type, GLsizei stride, GLsizei count, const void *pointer);
GLAPI PFNGLTEXCOORDPOINTEREXTPROC glad_glTexCoordPointerEXT;
#define glTexCoordPointerEXT glad_glTexCoordPointerEXT
typedef void (APIENTRYP PFNGLVERTEXPOINTEREXTPROC)(GLint size, GLenum type, GLsizei stride, GLsizei count, const void *pointer);
GLAPI PFNGLVERTEXPOINTEREXTPROC glad_glVertexPointerEXT;
#define glVertexPointerEXT glad_glVertexPointerEXT
#endif
#ifndef GL_EXT_vertex_array_bgra
#define GL_EXT_vertex_array_bgra 1
GLAPI int GLAD_GL_EXT_vertex_array_bgra;
#endif
#ifndef GL_EXT_vertex_attrib_64bit
#define GL_EXT_vertex_attrib_64bit 1
GLAPI int GLAD_GL_EXT_vertex_attrib_64bit;
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1DEXTPROC)(GLuint index, GLdouble x);
GLAPI PFNGLVERTEXATTRIBL1DEXTPROC glad_glVertexAttribL1dEXT;
#define glVertexAttribL1dEXT glad_glVertexAttribL1dEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2DEXTPROC)(GLuint index, GLdouble x, GLdouble y);
GLAPI PFNGLVERTEXATTRIBL2DEXTPROC glad_glVertexAttribL2dEXT;
#define glVertexAttribL2dEXT glad_glVertexAttribL2dEXT
```

```
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3DEXTPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLVERTEXATTRIBL3DEXTPROC glad_glVertexAttribL3dEXT;
#define glVertexAttribL3dEXT glad_glVertexAttribL3dEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4DEXTPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLVERTEXATTRIBL4DEXTPROC glad_glVertexAttribL4dEXT;
#define glVertexAttribL4dEXT glad_glVertexAttribL4dEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1DVEXTPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBL1DVEXTPROC glad_glVertexAttribL1dvEXT;
#define glVertexAttribL1dvEXT glad_glVertexAttribL1dvEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2DVEXTPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBL2DVEXTPROC glad_glVertexAttribL2dvEXT;
#define glVertexAttribL2dvEXT glad_glVertexAttribL2dvEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3DVEXTPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBL3DVEXTPROC glad_glVertexAttribL3dvEXT;
#define glVertexAttribL3dvEXT glad_glVertexAttribL3dvEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4DVEXTPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBL4DVEXTPROC glad_glVertexAttribL4dvEXT;
#define glVertexAttribL4dvEXT glad_glVertexAttribL4dvEXT
typedef void (APIENTRYP PFNGLVERTEXATTRIBLPOINTEREXTPROC)(GLuint index, GLint size, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLVERTEXATTRIBLPOINTEREXTPROC glad_glVertexAttribLPointerEXT;
#define glVertexAttribLPointerEXT glad_glVertexAttribLPointerEXT
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBLDVEXTPROC)(GLuint index, GLenum pname, GLdouble *params);
GLAPI PFNGLGETVERTEXATTRIBLDVEXTPROC glad_glGetVertexAttribLdvEXT;
#define glGetVertexAttribLdvEXT glad_glGetVertexAttribLdvEXT
#endif
#ifndef GL_EXT_vertex_shader
#define GL_EXT_vertex_shader 1
GLAPI int GLAD_GL_EXT_vertex_shader;
typedef void (APIENTRYP PFNGLBEGINVERTEXSHADEREXTPROC)(void);
GLAPI PFNGLBEGINVERTEXSHADEREXTPROC glad_glBeginVertexShaderEXT;
#define glBeginVertexShaderEXT glad_glBeginVertexShaderEXT
typedef void (APIENTRYP PFNGLENDVERTEXSHADEREXTPROC)(void);
GLAPI PFNGLENDVERTEXSHADEREXTPROC glad_glEndVertexShaderEXT;
#define glEndVertexShaderEXT glad_glEndVertexShaderEXT
typedef void (APIENTRYP PFNGLBINDVERTEXSHADEREXTPROC)(GLuint id);
GLAPI PFNGLBINDVERTEXSHADEREXTPROC glad_glBindVertexShaderEXT;
#define glBindVertexShaderEXT glad_glBindVertexShaderEXT
typedef GLuint (APIENTRYP PFNGLGENVERTEXSHADERSEXTPROC)(GLuint range);
GLAPI PFNGLGENVERTEXSHADERSEXTPROC glad_glGenVertexShadersEXT;
#define glGenVertexShadersEXT glad_glGenVertexShadersEXT
typedef void (APIENTRYP PFNGLDELETEVERTEXSHADEREXTPROC)(GLuint id);
GLAPI PFNGLDELETEVERTEXSHADEREXTPROC glad_glDeleteVertexShaderEXT;
#define glDeleteVertexShaderEXT glad_glDeleteVertexShaderEXT
typedef void (APIENTRYP PFNGLSHADEROP1EXTPROC)(GLenum op, GLuint res, GLuint arg1);
GLAPI PFNGLSHADEROP1EXTPROC glad_glShaderOp1EXT;
#define glShaderOp1EXT glad_glShaderOp1EXT
typedef void (APIENTRYP PFNGLSHADEROP2EXTPROC)(GLenum op, GLuint res, GLuint arg1, GLuint arg2);
GLAPI PFNGLSHADEROP2EXTPROC glad_glShaderOp2EXT;
#define glShaderOp2EXT glad_glShaderOp2EXT
typedef void (APIENTRYP PFNGLSHADEROP3EXTPROC)(GLenum op, GLuint res, GLuint arg1, GLuint arg2, GLuint arg3);
GLAPI PFNGLSHADEROP3EXTPROC glad_glShaderOp3EXT;
#define glShaderOp3EXT glad_glShaderOp3EXT
typedef void (APIENTRYP PFNGLSWIZZLEEXTPROC)(GLuint res, GLuint in, GLenum outX, GLenum outY, GLenum outZ, GLenum outW);
GLAPI PFNGLSWIZZLEEXTPROC glad_glSwizzleEXT;
#define glSwizzleEXT glad_glSwizzleEXT
typedef void (APIENTRYP PFNGLWRITEMASKEXTPROC)(GLuint res, GLuint in, GLenum outX, GLenum outY, GLenum outZ, GLenum outW);
GLAPI PFNGLWRITEMASKEXTPROC glad_glWriteMaskEXT;
#define glWriteMaskEXT glad_glWriteMaskEXT
typedef void (APIENTRYP PFNGLINSERTCOMPONENTEXTPROC)(GLuint res, GLuint src, GLuint num);
GLAPI PFNGLINSERTCOMPONENTEXTPROC glad_glInsertComponentEXT;
#define glInsertComponentEXT glad_glInsertComponentEXT
typedef void (APIENTRYP PFNGLEXTRACTCOMPONENTEXTPROC)(GLuint res, GLuint src, GLuint num);
GLAPI PFNGLEXTRACTCOMPONENTEXTPROC glad_glExtractComponentEXT;
#define glExtractComponentEXT glad_glExtractComponentEXT
typedef GLuint (APIENTRYP PFNGLGENSYMBOLSEXTPROC)(GLenum datatype, GLenum storagetype, GLenum range, GLuint components);
GLAPI PFNGLGENSYMBOLSEXTPROC glad_glGenSymbolsEXT;
#define glGenSymbolsEXT glad_glGenSymbolsEXT
typedef void (APIENTRYP PFNGLSETINVARIANTEXTPROC)(GLuint id, GLenum type, const void *addr);
GLAPI PFNGLSETINVARIANTEXTPROC glad_glSetInvariantEXT;
#define glSetInvariantEXT glad_glSetInvariantEXT
typedef void (APIENTRYP PFNGLSETLOCALCONSTANTEXTPROC)(GLuint id, GLenum type, const void *addr);
GLAPI PFNGLSETLOCALCONSTANTEXTPROC glad_glSetLocalConstantEXT;
#define glSetLocalConstantEXT glad_glSetLocalConstantEXT
typedef void (APIENTRYP PFNGLVARIANTBVEXTPROC)(GLuint id, const GLbyte *addr);
GLAPI PFNGLVARIANTBVEXTPROC glad_glVariantbvEXT;
#define glVariantbvEXT glad_glVariantbvEXT
typedef void (APIENTRYP PFNGLVARIANTSVEXTPROC)(GLuint id, const GLshort *addr);
GLAPI PFNGLVARIANTSVEXTPROC glad_glVariantsvEXT;
#define glVariantsvEXT glad_glVariantsvEXT
typedef void (APIENTRYP PFNGLVARIANTIVEXTPROC)(GLuint id, const GLint *addr);
```

```
GLAPI PFNGLVARIANTIVEXTPROC glad_glVariantivEXT;
#define glVariantivEXT glad_glVariantivEXT
typedef void (APIENTRYP PFNGLVARIANTFVEXTPROC)(GLuint id, const GLfloat *addr);
GLAPI PFNGLVARIANTFVEXTPROC glad_glVariantfvEXT;
#define glVariantfvEXT glad_glVariantfvEXT
typedef void (APIENTRYP PFNGLVARIANTDVEXTPROC)(GLuint id, const GLdouble *addr);
GLAPI PFNGLVARIANTDVEXTPROC glad_glVariantdvEXT;
#define glVariantdvEXT glad_glVariantdvEXT
typedef void (APIENTRYP PFNGLVARIANTUBVEXTPROC)(GLuint id, const GLubyte *addr);
GLAPI PFNGLVARIANTUBVEXTPROC glad_glVariantubvEXT;
#define glVariantubvEXT glad_glVariantubvEXT
typedef void (APIENTRYP PFNGLVARIANTUSVEXTPROC)(GLuint id, const GLushort *addr);
GLAPI PFNGLVARIANTUSVEXTPROC glad_glVariantusvEXT;
#define glVariantusvEXT glad_glVariantusvEXT
typedef void (APIENTRYP PFNGLVARIANTUIVEXTPROC)(GLuint id, const GLuint *addr);
GLAPI PFNGLVARIANTUIVEXTPROC glad_glVariantuivEXT;
#define glVariantuivEXT glad_glVariantuivEXT
typedef void (APIENTRYP PFNGLVARIANTPOINTEREXTPROC)(GLuint id, GLenum type, GLuint stride, const void *addr);
GLAPI PFNGLVARIANTPOINTEREXTPROC glad_glVariantPointerEXT;
#define glVariantPointerEXT glad_glVariantPointerEXT
typedef void (APIENTRYP PFNGLENABLEVARIANTCLIENTSTATEEXTPROC)(GLuint id);
GLAPI PFNGLENABLEVARIANTCLIENTSTATEEXTPROC glad_glEnableVariantClientStateEXT;
#define glEnableVariantClientStateEXT glad_glEnableVariantClientStateEXT
typedef void (APIENTRYP PFNGLDISABLEVARIANTCLIENTSTATEEXTPROC)(GLuint id);
GLAPI PFNGLDISABLEVARIANTCLIENTSTATEEXTPROC glad_glDisableVariantClientStateEXT;
#define glDisableVariantClientStateEXT glad_glDisableVariantClientStateEXT
typedef GLuint (APIENTRYP PFNGLBINDLIGHTPARAMETEREXTPROC)(GLenum light, GLenum value);
GLAPI PFNGLBINDLIGHTPARAMETEREXTPROC glad_glBindLightParameterEXT;
#define glBindLightParameterEXT glad_glBindLightParameterEXT
typedef GLuint (APIENTRYP PFNGLBINDMATERIALPARAMETEREXTPROC)(GLenum face, GLenum value);
GLAPI PFNGLBINDMATERIALPARAMETEREXTPROC glad_glBindMaterialParameterEXT;
#define glBindMaterialParameterEXT glad_glBindMaterialParameterEXT
typedef GLuint (APIENTRYP PFNGLBINDTEXGENPARAMETEREXTPROC)(GLenum unit, GLenum coord, GLenum value);
GLAPI PFNGLBINDTEXGENPARAMETEREXTPROC glad_glBindTexGenParameterEXT;
#define glBindTexGenParameterEXT glad_glBindTexGenParameterEXT
typedef GLuint (APIENTRYP PFNGLBINDTEXTUREUNITPARAMETEREXTPROC)(GLenum unit, GLenum value);
GLAPI PFNGLBINDTEXTUREUNITPARAMETEREXTPROC glad_glBindTextureUnitParameterEXT;
#define glBindTextureUnitParameterEXT glad_glBindTextureUnitParameterEXT
typedef GLuint (APIENTRYP PFNGLBINDPARAMETEREXTPROC)(GLenum value);
GLAPI PFNGLBINDPARAMETEREXTPROC glad_glBindParameterEXT;
#define glBindParameterEXT glad_glBindParameterEXT
typedef GLboolean (APIENTRYP PFNGLISVARIANTENABLEDEXTPROC)(GLuint id, GLenum cap);
GLAPI PFNGLISVARIANTENABLEDEXTPROC glad_glIsVariantEnabledEXT;
#define glIsVariantEnabledEXT glad_glIsVariantEnabledEXT
typedef void (APIENTRYP PFNGLGETVARIANTBOOLEANVEXTPROC)(GLuint id, GLenum value, GLboolean *data);
GLAPI PFNGLGETVARIANTBOOLEANVEXTPROC glad_glGetVariantBooleanvEXT;
#define glGetVariantBooleanvEXT glad_glGetVariantBooleanvEXT
typedef void (APIENTRYP PFNGLGETVARIANTINTEGERVEXTPROC)(GLuint id, GLenum value, GLint *data);
GLAPI PFNGLGETVARIANTINTEGERVEXTPROC glad_glGetVariantIntegervEXT;
#define glGetVariantIntegervEXT glad_glGetVariantIntegervEXT
typedef void (APIENTRYP PFNGLGETVARIANTFLOATVEXTPROC)(GLuint id, GLenum value, GLfloat *data);
GLAPI PFNGLGETVARIANTFLOATVEXTPROC glad_glGetVariantFloatvEXT;
#define glGetVariantFloatvEXT glad_glGetVariantFloatvEXT
typedef void (APIENTRYP PFNGLGETVARIANTPOINTERVEXTPROC)(GLuint id, GLenum value, void **data);
GLAPI PFNGLGETVARIANTPOINTERVEXTPROC glad_glGetVariantPointervEXT;
#define glGetVariantPointervEXT glad_glGetVariantPointervEXT
typedef void (APIENTRYP PFNGLGETINVARIANTBOOLEANVEXTPROC)(GLuint id, GLenum value, GLboolean *data);
GLAPI PFNGLGETINVARIANTBOOLEANVEXTPROC glad_glGetInvariantBooleanvEXT;
#define glGetInvariantBooleanvEXT glad_glGetInvariantBooleanvEXT
typedef void (APIENTRYP PFNGLGETINVARIANTINTEGERVEXTPROC)(GLuint id, GLenum value, GLint *data);
GLAPI PFNGLGETINVARIANTINTEGERVEXTPROC glad_glGetInvariantIntegervEXT;
#define glGetInvariantIntegervEXT glad_glGetInvariantIntegervEXT
typedef void (APIENTRYP PFNGLGETINVARIANTFLOATVEXTPROC)(GLuint id, GLenum value, GLfloat *data);
GLAPI PFNGLGETINVARIANTFLOATVEXTPROC glad_glGetInvariantFloatvEXT;
#define glGetInvariantFloatvEXT glad_glGetInvariantFloatvEXT
typedef void (APIENTRYP PFNGLGETLOCALCONSTANTBOOLEANVEXTPROC)(GLuint id, GLenum value, GLboolean *data);
GLAPI PFNGLGETLOCALCONSTANTBOOLEANVEXTPROC glad_glGetLocalConstantBooleanvEXT;
#define glGetLocalConstantBooleanvEXT glad_glGetLocalConstantBooleanvEXT
typedef void (APIENTRYP PFNGLGETLOCALCONSTANTINTEGERVEXTPROC)(GLuint id, GLenum value, GLint *data);
GLAPI PFNGLGETLOCALCONSTANTINTEGERVEXTPROC glad_glGetLocalConstantIntegervEXT;
#define glGetLocalConstantIntegervEXT glad_glGetLocalConstantIntegervEXT
typedef void (APIENTRYP PFNGLGETLOCALCONSTANTFLOATVEXTPROC)(GLuint id, GLenum value, GLfloat *data);
GLAPI PFNGLGETLOCALCONSTANTFLOATVEXTPROC glad_glGetLocalConstantFloatvEXT;
#define glGetLocalConstantFloatvEXT glad_glGetLocalConstantFloatvEXT
#endif
#ifndef GL_EXT_vertex_weighting
#define GL_EXT_vertex_weighting 1
GLAPI int GLAD_GL_EXT_vertex_weighting;
typedef void (APIENTRYP PFNGLVERTEXWEIGHTFEXTPROC)(GLfloat weight);
GLAPI PFNGLVERTEXWEIGHTFEXTPROC glad_glVertexWeightfEXT;
#define glVertexWeightfEXT glad_glVertexWeightfEXT
typedef void (APIENTRYP PFNGLVERTEXWEIGHTFVEXTPROC)(const GLfloat *weight);
GLAPI PFNGLVERTEXWEIGHTFVEXTPROC glad_glVertexWeightfvEXT;
#define glVertexWeightfvEXT glad_glVertexWeightfvEXT
typedef void (APIENTRYP PFNGLVERTEXWEIGHTPOINTEREXTPROC)(GLint size, GLenum type, GLsizei stride,
const void *pointer);
GLAPI PFNGLVERTEXWEIGHTPOINTEREXTPROC glad_glVertexWeightPointerEXT;
#define glVertexWeightPointerEXT glad_glVertexWeightPointerEXT
#endif
#ifndef GL_EXT_win32_keyed_mutex
#define GL_EXT_win32_keyed_mutex 1
GLAPI int GLAD_GL_EXT_win32_keyed_mutex;
typedef GLboolean (APIENTRYP PFNGLACQUIREKEYEDMUTEXWIN32EXTPROC)(GLuint memory, GLuint64 key, GLuint
timeout);
GLAPI PFNGLACQUIREKEYEDMUTEXWIN32EXTPROC glad_glAcquireKeyedMutexWin32EXT;
#define glAcquireKeyedMutexWin32EXT glad_glAcquireKeyedMutexWin32EXT
typedef GLboolean (APIENTRYP PFNGLRELEASEKEYEDMUTEXWIN32EXTPROC)(GLuint memory, GLuint64 key);
GLAPI PFNGLRELEASEKEYEDMUTEXWIN32EXTPROC glad_glReleaseKeyedMutexWin32EXT;
#define glReleaseKeyedMutexWin32EXT glad_glReleaseKeyedMutexWin32EXT
#endif
#ifndef GL_EXT_window_rectangles
#define GL_EXT_window_rectangles 1
GLAPI int GLAD_GL_EXT_window_rectangles;
typedef void (APIENTRYP PFNGLWINDOWRECTANGLESEXTPROC)(GLenum mode, GLsizei count, const GLint *box);
GLAPI PFNGLWINDOWRECTANGLESEXTPROC glad_glWindowRectanglesEXT;
#define glWindowRectanglesEXT glad_glWindowRectanglesEXT
#endif
#ifndef GL_EXT_x11_sync_object
#define GL_EXT_x11_sync_object 1
GLAPI int GLAD_GL_EXT_x11_sync_object;
typedef GLsync (APIENTRYP PFNGLIMPORTSYNCEXTPROC)(GLenum external_sync_type, GLintptr external_sync,
GLbitfield flags);
GLAPI PFNGLIMPORTSYNCEXTPROC glad_glImportSyncEXT;
#define glImportSyncEXT glad_glImportSyncEXT
#endif
#ifndef GL_GREMEDY_frame_terminator
#define GL_GREMEDY_frame_terminator 1
GLAPI int GLAD_GL_GREMEDY_frame_terminator;
typedef void (APIENTRYP PFNGLFRAMETERMINATORGREMEDYPROC)(void);
GLAPI PFNGLFRAMETERMINATORGREMEDYPROC glad_glFrameTerminatorGREMEDY;
#define glFrameTerminatorGREMEDY glad_glFrameTerminatorGREMEDY
#endif
#ifndef GL_GREMEDY_string_marker
#define GL_GREMEDY_string_marker 1
GLAPI int GLAD_GL_GREMEDY_string_marker;
typedef void (APIENTRYP PFNGLSTRINGMARKERGREMEDYPROC)(GLsizei len, const void *string);
GLAPI PFNGLSTRINGMARKERGREMEDYPROC glad_glStringMarkerGREMEDY;
#define glStringMarkerGREMEDY glad_glStringMarkerGREMEDY
#endif
#ifndef GL_HP_convolution_border_modes
#define GL_HP_convolution_border_modes 1
GLAPI int GLAD_GL_HP_convolution_border_modes;
#endif
#ifndef GL_HP_image_transform
#define GL_HP_image_transform 1
GLAPI int GLAD_GL_HP_image_transform;
typedef void (APIENTRYP PFNGLIMAGETRANSFORMPARAMETERIHPPROC)(GLenum target, GLenum pname, GLint
param);
GLAPI PFNGLIMAGETRANSFORMPARAMETERIHPPROC glad_glImageTransformParameteriHP;
#define glImageTransformParameteriHP glad_glImageTransformParameteriHP
typedef void (APIENTRYP PFNGLIMAGETRANSFORMPARAMETERFHPPROC)(GLenum target, GLenum pname, GLfloat
param);
GLAPI PFNGLIMAGETRANSFORMPARAMETERFHPPROC glad_glImageTransformParameterfHP;
#define glImageTransformParameterfHP glad_glImageTransformParameterfHP
typedef void (APIENTRYP PFNGLIMAGETRANSFORMPARAMETERIVHPPROC)(GLenum target, GLenum pname, const GLint
*params);
GLAPI PFNGLIMAGETRANSFORMPARAMETERIVHPPROC glad_glImageTransformParameterivHP;
#define glImageTransformParameterivHP glad_glImageTransformParameterivHP
typedef void (APIENTRYP PFNGLIMAGETRANSFORMPARAMETERFVHPPROC)(GLenum target, GLenum pname, const
GLfloat *params);
GLAPI PFNGLIMAGETRANSFORMPARAMETERFVHPPROC glad_glImageTransformParameterfvHP;
#define glImageTransformParameterfvHP glad_glImageTransformParameterfvHP
typedef void (APIENTRYP PFNGLGETIMAGETRANSFORMPARAMETERIVHPPROC)(GLenum target, GLenum pname, GLint
*params);
GLAPI PFNGLGETIMAGETRANSFORMPARAMETERIVHPPROC glad_glGetImageTransformParameterivHP;
#define glGetImageTransformParameterivHP glad_glGetImageTransformParameterivHP
typedef void (APIENTRYP PFNGLGETIMAGETRANSFORMPARAMETERFVHPPROC)(GLenum target, GLenum pname, GLfloat
*params);
GLAPI PFNGLGETIMAGETRANSFORMPARAMETERFVHPPROC glad_glGetImageTransformParameterfvHP;
#define glGetImageTransformParameterfvHP glad_glGetImageTransformParameterfvHP
#endif
#ifndef GL_HP_occlusion_test
#define GL_HP_occlusion_test 1
GLAPI int GLAD_GL_HP_occlusion_test;
#endif
#ifndef GL_HP_texture_lighting
#define GL_HP_texture_lighting 1
GLAPI int GLAD_GL_HP_texture_lighting;
#endif
#ifndef GL_IBM_cull_vertex
#define GL_IBM_cull_vertex 1
GLAPI int GLAD_GL_IBM_cull_vertex;
```

```
#endif
#ifndef GL_IBM_multimode_draw_arrays
#define GL_IBM_multimode_draw_arrays 1
GLAPI int GLAD_GL_IBM_multimode_draw_arrays;
typedef void (APIENTRYP PFNGLMULTIMODEDRAWARRAYSIBMPROC)(const GLenum *mode, const GLint *first, const GLsizei *count, GLsizei primcount, GLint modestride);
GLAPI PFNGLMULTIMODEDRAWARRAYSIBMPROC glad_glMultiModeDrawArraysIBM;
#define glMultiModeDrawArraysIBM glad_glMultiModeDrawArraysIBM
typedef void (APIENTRYP PFNGLMULTIMODEDRAWELEMENTSIBMPROC)(const GLenum *mode, const GLsizei *count, GLenum type, const void *const*indices, GLsizei primcount, GLint modestride);
GLAPI PFNGLMULTIMODEDRAWELEMENTSIBMPROC glad_glMultiModeDrawElementsIBM;
#define glMultiModeDrawElementsIBM glad_glMultiModeDrawElementsIBM
#endif
#ifndef GL_IBM_rasterpos_clip
#define GL_IBM_rasterpos_clip 1
GLAPI int GLAD_GL_IBM_rasterpos_clip;
#endif
#ifndef GL_IBM_static_data
#define GL_IBM_static_data 1
GLAPI int GLAD_GL_IBM_static_data;
typedef void (APIENTRYP PFNGLFLUSHSTATICDATAIBMPROC)(GLenum target);
GLAPI PFNGLFLUSHSTATICDATAIBMPROC glad_glFlushStaticDataIBM;
#define glFlushStaticDataIBM glad_glFlushStaticDataIBM
#endif
#ifndef GL_IBM_texture_mirrored_repeat
#define GL_IBM_texture_mirrored_repeat 1
GLAPI int GLAD_GL_IBM_texture_mirrored_repeat;
#endif
#ifndef GL_IBM_vertex_array_lists
#define GL_IBM_vertex_array_lists 1
GLAPI int GLAD_GL_IBM_vertex_array_lists;
typedef void (APIENTRYP PFNGLCOLORPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride, const void **pointer, GLint ptrstride);
GLAPI PFNGLCOLORPOINTERLISTIBMPROC glad_glColorPointerListIBM;
#define glColorPointerListIBM glad_glColorPointerListIBM
typedef void (APIENTRYP PFNGLSECONDARYCOLORPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride, const void **pointer, GLint ptrstride);
GLAPI PFNGLSECONDARYCOLORPOINTERLISTIBMPROC glad_glSecondaryColorPointerListIBM;
#define glSecondaryColorPointerListIBM glad_glSecondaryColorPointerListIBM
typedef void (APIENTRYP PFNGLEDGEFLAGPOINTERLISTIBMPROC)(GLint stride, const GLboolean **pointer, GLint ptrstride);
GLAPI PFNGLEDGEFLAGPOINTERLISTIBMPROC glad_glEdgeFlagPointerListIBM;
#define glEdgeFlagPointerListIBM glad_glEdgeFlagPointerListIBM
typedef void (APIENTRYP PFNGLFOGCOORDPOINTERLISTIBMPROC)(GLenum type, GLint stride, const void **pointer, GLint ptrstride);
GLAPI PFNGLFOGCOORDPOINTERLISTIBMPROC glad_glFogCoordPointerListIBM;
#define glFogCoordPointerListIBM glad_glFogCoordPointerListIBM
typedef void (APIENTRYP PFNGLINDEXPOINTERLISTIBMPROC)(GLenum type, GLint stride, const void **pointer, GLint ptrstride);
GLAPI PFNGLINDEXPOINTERLISTIBMPROC glad_glIndexPointerListIBM;
#define glIndexPointerListIBM glad_glIndexPointerListIBM
typedef void (APIENTRYP PFNGLNORMALPOINTERLISTIBMPROC)(GLenum type, GLint stride, const void **pointer, GLint ptrstride);
GLAPI PFNGLNORMALPOINTERLISTIBMPROC glad_glNormalPointerListIBM;
#define glNormalPointerListIBM glad_glNormalPointerListIBM
typedef void (APIENTRYP PFNGLTEXCOORDPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride, const void **pointer, GLint ptrstride);
GLAPI PFNGLTEXCOORDPOINTERLISTIBMPROC glad_glTexCoordPointerListIBM;
#define glTexCoordPointerListIBM glad_glTexCoordPointerListIBM
typedef void (APIENTRYP PFNGLVERTEXPOINTERLISTIBMPROC)(GLint size, GLenum type, GLint stride, const void **pointer, GLint ptrstride);
GLAPI PFNGLVERTEXPOINTERLISTIBMPROC glad_glVertexPointerListIBM;
#define glVertexPointerListIBM glad_glVertexPointerListIBM
#endif
#ifndef GL_INGR_blend_func_separate
#define GL_INGR_blend_func_separate 1
GLAPI int GLAD_GL_INGR_blend_func_separate;
typedef void (APIENTRYP PFNGLBLENDFUNCSEPARATEINGRPROC)(GLenum sfactorRGB, GLenum dfactorRGB, GLenum sfactorAlpha, GLenum dfactorAlpha);
GLAPI PFNGLBLENDFUNCSEPARATEINGRPROC glad_glBlendFuncSeparateINGR;
#define glBlendFuncSeparateINGR glad_glBlendFuncSeparateINGR
#endif
#ifndef GL_INGR_color_clamp
#define GL_INGR_color_clamp 1
GLAPI int GLAD_GL_INGR_color_clamp;
#endif
#ifndef GL_INGR_interlace_read
#define GL_INGR_interlace_read 1
GLAPI int GLAD_GL_INGR_interlace_read;
#endif
#ifndef GL_INTEL_blackhole_render
#define GL_INTEL_blackhole_render 1
GLAPI int GLAD_GL_INTEL_blackhole_render;
#endif
#ifndef GL_INTEL_conservative_rasterization
#define GL_INTEL_conservative_rasterization 1
GLAPI int GLAD_GL_INTEL_conservative_rasterization;
```

```
#endif
#ifndef GL_INTEL_fragment_shader_ordering
#define GL_INTEL_fragment_shader_ordering 1
GLAPI int GLAD_GL_INTEL_fragment_shader_ordering;
#endif
#ifndef GL_INTEL_framebuffer_CMAA
#define GL_INTEL_framebuffer_CMAA 1
GLAPI int GLAD_GL_INTEL_framebuffer_CMAA;
typedef void (APIENTRYP PFNGLAPPLYFRAMEBUFFERATTACHMENTCMAAINTELPROC)(void);
GLAPI PFNGLAPPLYFRAMEBUFFERATTACHMENTCMAAINTELPROC glad_glApplyFramebufferAttachmentCMAAINTEL;
#define glApplyFramebufferAttachmentCMAAINTEL glad_glApplyFramebufferAttachmentCMAAINTEL
#endif
#ifndef GL_INTEL_map_texture
#define GL_INTEL_map_texture 1
GLAPI int GLAD_GL_INTEL_map_texture;
typedef void (APIENTRYP PFNGLSYNCTEXTUREINTELPROC)(GLuint texture);
GLAPI PFNGLSYNCTEXTUREINTELPROC glad_glSyncTextureINTEL;
#define glSyncTextureINTEL glad_glSyncTextureINTEL
typedef void (APIENTRYP PFNGLUNMAPTEXTURE2DINTELPROC)(GLuint texture, GLint level);
GLAPI PFNGLUNMAPTEXTURE2DINTELPROC glad_glUnmapTexture2DINTEL;
#define glUnmapTexture2DINTEL glad_glUnmapTexture2DINTEL
typedef void * (APIENTRYP PFNGLMAPTEXTURE2DINTELPROC)(GLuint texture, GLint level, GLbitfield access, GLint *stride, GLenum *layout);
GLAPI PFNGLMAPTEXTURE2DINTELPROC glad_glMapTexture2DINTEL;
#define glMapTexture2DINTEL glad_glMapTexture2DINTEL
#endif
#ifndef GL_INTEL_parallel_arrays
#define GL_INTEL_parallel_arrays 1
GLAPI int GLAD_GL_INTEL_parallel_arrays;
typedef void (APIENTRYP PFNGLVERTEXPOINTERVINTELPROC)(GLint size, GLenum type, const void **pointer);
GLAPI PFNGLVERTEXPOINTERVINTELPROC glad_glVertexPointervINTEL;
#define glVertexPointervINTEL glad_glVertexPointervINTEL
typedef void (APIENTRYP PFNGLNORMALPOINTERVINTELPROC)(GLenum type, const void **pointer);
GLAPI PFNGLNORMALPOINTERVINTELPROC glad_glNormalPointervINTEL;
#define glNormalPointervINTEL glad_glNormalPointervINTEL
typedef void (APIENTRYP PFNGLCOLORPOINTERVINTELPROC)(GLint size, GLenum type, const void **pointer);
GLAPI PFNGLCOLORPOINTERVINTELPROC glad_glColorPointervINTEL;
#define glColorPointervINTEL glad_glColorPointervINTEL
typedef void (APIENTRYP PFNGLTEXCOORDPOINTERVINTELPROC)(GLint size, GLenum type, const void **pointer);
GLAPI PFNGLTEXCOORDPOINTERVINTELPROC glad_glTexCoordPointervINTEL;
#define glTexCoordPointervINTEL glad_glTexCoordPointervINTEL
#endif
#ifndef GL_INTEL_performance_query
#define GL_INTEL_performance_query 1
GLAPI int GLAD_GL_INTEL_performance_query;
typedef void (APIENTRYP PFNGLBEGINPERFQUERYINTELPROC)(GLuint queryHandle);
GLAPI PFNGLBEGINPERFQUERYINTELPROC glad_glBeginPerfQueryINTEL;
#define glBeginPerfQueryINTEL glad_glBeginPerfQueryINTEL
typedef void (APIENTRYP PFNGLCREATEPERFQUERYINTELPROC)(GLuint queryId, GLuint *queryHandle);
GLAPI PFNGLCREATEPERFQUERYINTELPROC glad_glCreatePerfQueryINTEL;
#define glCreatePerfQueryINTEL glad_glCreatePerfQueryINTEL
typedef void (APIENTRYP PFNGLDELETEPERFQUERYINTELPROC)(GLuint queryHandle);
GLAPI PFNGLDELETEPERFQUERYINTELPROC glad_glDeletePerfQueryINTEL;
#define glDeletePerfQueryINTEL glad_glDeletePerfQueryINTEL
typedef void (APIENTRYP PFNGLENDPERFQUERYINTELPROC)(GLuint queryHandle);
GLAPI PFNGLENDPERFQUERYINTELPROC glad_glEndPerfQueryINTEL;
#define glEndPerfQueryINTEL glad_glEndPerfQueryINTEL
typedef void (APIENTRYP PFNGLGETFIRSTPERFQUERYIDINTELPROC)(GLuint *queryId);
GLAPI PFNGLGETFIRSTPERFQUERYIDINTELPROC glad_glGetFirstPerfQueryIdINTEL;
#define glGetFirstPerfQueryIdINTEL glad_glGetFirstPerfQueryIdINTEL
typedef void (APIENTRYP PFNGLGETNEXTPERFQUERYIDINTELPROC)(GLuint queryId, GLuint *nextQueryId);
GLAPI PFNGLGETNEXTPERFQUERYIDINTELPROC glad_glGetNextPerfQueryIdINTEL;
#define glGetNextPerfQueryIdINTEL glad_glGetNextPerfQueryIdINTEL
typedef void (APIENTRYP PFNGLGETPERFCOUNTERINFOINTELPROC)(GLuint queryId, GLuint counterId, GLuint counterNameLength, GLchar *counterName, GLuint counterDescLength, GLchar *counterDesc, GLuint *counterOffset, GLuint *counterDataSize, GLuint *counterTypeEnum, GLuint *counterDataTypeEnum, GLuint64 *rawCounterMaxValue);
GLAPI PFNGLGETPERFCOUNTERINFOINTELPROC glad_glGetPerfCounterInfoINTEL;
#define glGetPerfCounterInfoINTEL glad_glGetPerfCounterInfoINTEL
typedef void (APIENTRYP PFNGLGETPERFQUERYDATAINTELPROC)(GLuint queryHandle, GLuint flags, GLsizei dataSize, void *data, GLuint *bytesWritten);
GLAPI PFNGLGETPERFQUERYDATAINTELPROC glad_glGetPerfQueryDataINTEL;
#define glGetPerfQueryDataINTEL glad_glGetPerfQueryDataINTEL
typedef void (APIENTRYP PFNGLGETPERFQUERYIDBYNAMEINTELPROC)(GLchar *queryName, GLuint *queryId);
GLAPI PFNGLGETPERFQUERYIDBYNAMEINTELPROC glad_glGetPerfQueryIdByNameINTEL;
#define glGetPerfQueryIdByNameINTEL glad_glGetPerfQueryIdByNameINTEL
typedef void (APIENTRYP PFNGLGETPERFQUERYINFOINTELPROC)(GLuint queryId, GLuint queryNameLength, GLchar *queryName, GLuint *dataSize, GLuint *noCounters, GLuint *noInstances, GLuint *capsMask);
GLAPI PFNGLGETPERFQUERYINFOINTELPROC glad_glGetPerfQueryInfoINTEL;
#define glGetPerfQueryInfoINTEL glad_glGetPerfQueryInfoINTEL
#endif
#ifndef GL_KHR_blend_equation_advanced
#define GL_KHR_blend_equation_advanced 1
GLAPI int GLAD_GL_KHR_blend_equation_advanced;
typedef void (APIENTRYP PFNGLBLENDBARRIERKHRPROC)(void);
GLAPI PFNGLBLENDBARRIERKHRPROC glad_glBlendBarrierKHR;
```

```
#define glBlendBarrierKHR glad_glBlendBarrierKHR
#endif
#ifndef GL_KHR_blend_equation_advanced_coherent
#define GL_KHR_blend_equation_advanced_coherent 1
GLAPI int GLAD_GL_KHR_blend_equation_advanced_coherent;
#endif
#ifndef GL_KHR_context_flush_control
#define GL_KHR_context_flush_control 1
GLAPI int GLAD_GL_KHR_context_flush_control;
#endif
#ifndef GL_KHR_debug
#define GL_KHR_debug 1
GLAPI int GLAD_GL_KHR_debug;
typedef void (APIENTRYP PFNGLDEBUGMESSAGECONTROLPROC)(GLenum source, GLenum type, GLenum severity, GLsizei count, const GLuint *ids, GLboolean enabled);
GLAPI PFNGLDEBUGMESSAGECONTROLPROC glad_glDebugMessageControl;
#define glDebugMessageControl glad_glDebugMessageControl
typedef void (APIENTRYP PFNGLDEBUGMESSAGEINSERTPROC)(GLenum source, GLenum type, GLuint id, GLenum severity, GLsizei length, const GLchar *buf);
GLAPI PFNGLDEBUGMESSAGEINSERTPROC glad_glDebugMessageInsert;
#define glDebugMessageInsert glad_glDebugMessageInsert
typedef void (APIENTRYP PFNGLDEBUGMESSAGECALLBACKPROC)(GLDEBUGPROC callback, const void *userParam);
GLAPI PFNGLDEBUGMESSAGECALLBACKPROC glad_glDebugMessageCallback;
#define glDebugMessageCallback glad_glDebugMessageCallback
typedef GLuint (APIENTRYP PFNGLGETDEBUGMESSAGELOGPROC)(GLuint count, GLsizei bufSize, GLenum *sources, GLenum *types, GLuint *ids, GLenum *severities, GLsizei *lengths, GLchar *messageLog);
GLAPI PFNGLGETDEBUGMESSAGELOGPROC glad_glGetDebugMessageLog;
#define glGetDebugMessageLog glad_glGetDebugMessageLog
typedef void (APIENTRYP PFNGLPUSHDEBUGGROUPPROC)(GLenum source, GLuint id, GLsizei length, const GLchar *message);
GLAPI PFNGLPUSHDEBUGGROUPPROC glad_glPushDebugGroup;
#define glPushDebugGroup glad_glPushDebugGroup
typedef void (APIENTRYP PFNGLPOPDEBUGGROUPPROC)(void);
GLAPI PFNGLPOPDEBUGGROUPPROC glad_glPopDebugGroup;
#define glPopDebugGroup glad_glPopDebugGroup
typedef void (APIENTRYP PFNGLOBJECTLABELPROC)(GLenum identifier, GLuint name, GLsizei length, const GLchar *label);
GLAPI PFNGLOBJECTLABELPROC glad_glObjectLabel;
#define glObjectLabel glad_glObjectLabel
typedef void (APIENTRYP PFNGLGETOBJECTLABELPROC)(GLenum identifier, GLuint name, GLsizei bufSize, GLsizei *length, GLchar *label);
GLAPI PFNGLGETOBJECTLABELPROC glad_glGetObjectLabel;
#define glGetObjectLabel glad_glGetObjectLabel
typedef void (APIENTRYP PFNGLOBJECTPTRLABELPROC)(const void *ptr, GLsizei length, const GLchar *label);
GLAPI PFNGLOBJECTPTRLABELPROC glad_glObjectPtrLabel;
#define glObjectPtrLabel glad_glObjectPtrLabel
typedef void (APIENTRYP PFNGLGETOBJECTPTRLABELPROC)(const void *ptr, GLsizei bufSize, GLsizei *length, GLchar *label);
GLAPI PFNGLGETOBJECTPTRLABELPROC glad_glGetObjectPtrLabel;
#define glGetObjectPtrLabel glad_glGetObjectPtrLabel
typedef void (APIENTRYP PFNGLGETPOINTERVPROC)(GLenum pname, void **params);
GLAPI PFNGLGETPOINTERVPROC glad_glGetPointerv;
#define glGetPointerv glad_glGetPointerv
typedef void (APIENTRYP PFNGLDEBUGMESSAGECONTROLKHRPROC)(GLenum source, GLenum type, GLenum severity, GLsizei count, const GLuint *ids, GLboolean enabled);
GLAPI PFNGLDEBUGMESSAGECONTROLKHRPROC glad_glDebugMessageControlKHR;
#define glDebugMessageControlKHR glad_glDebugMessageControlKHR
typedef void (APIENTRYP PFNGLDEBUGMESSAGEINSERTKHRPROC)(GLenum source, GLenum type, GLuint id, GLenum severity, GLsizei length, const GLchar *buf);
GLAPI PFNGLDEBUGMESSAGEINSERTKHRPROC glad_glDebugMessageInsertKHR;
#define glDebugMessageInsertKHR glad_glDebugMessageInsertKHR
typedef void (APIENTRYP PFNGLDEBUGMESSAGECALLBACKKHRPROC)(GLDEBUGPROCKHR callback, const void *userParam);
GLAPI PFNGLDEBUGMESSAGECALLBACKKHRPROC glad_glDebugMessageCallbackKHR;
#define glDebugMessageCallbackKHR glad_glDebugMessageCallbackKHR
typedef GLuint (APIENTRYP PFNGLGETDEBUGMESSAGELOGKHRPROC)(GLuint count, GLsizei bufSize, GLenum *sources, GLenum *types, GLuint *ids, GLenum *severities, GLsizei *lengths, GLchar *messageLog);
GLAPI PFNGLGETDEBUGMESSAGELOGKHRPROC glad_glGetDebugMessageLogKHR;
#define glGetDebugMessageLogKHR glad_glGetDebugMessageLogKHR
typedef void (APIENTRYP PFNGLPUSHDEBUGGROUPKHRPROC)(GLenum source, GLuint id, GLsizei length, const GLchar *message);
GLAPI PFNGLPUSHDEBUGGROUPKHRPROC glad_glPushDebugGroupKHR;
#define glPushDebugGroupKHR glad_glPushDebugGroupKHR
typedef void (APIENTRYP PFNGLPOPDEBUGGROUPKHRPROC)(void);
GLAPI PFNGLPOPDEBUGGROUPKHRPROC glad_glPopDebugGroupKHR;
#define glPopDebugGroupKHR glad_glPopDebugGroupKHR
typedef void (APIENTRYP PFNGLOBJECTLABELKHRPROC)(GLenum identifier, GLuint name, GLsizei length, const GLchar *label);
GLAPI PFNGLOBJECTLABELKHRPROC glad_glObjectLabelKHR;
#define glObjectLabelKHR glad_glObjectLabelKHR
typedef void (APIENTRYP PFNGLGETOBJECTLABELKHRPROC)(GLenum identifier, GLuint name, GLsizei bufSize, GLsizei *length, GLchar *label);
GLAPI PFNGLGETOBJECTLABELKHRPROC glad_glGetObjectLabelKHR;
#define glGetObjectLabelKHR glad_glGetObjectLabelKHR
typedef void (APIENTRYP PFNGLOBJECTPTRLABELKHRPROC)(const void *ptr, GLsizei length, const GLchar *label);
```

```
GLAPI PFNGLOBJECTPTRLABELKHRPROC glad_glObjectPtrLabelKHR;
#define glObjectPtrLabelKHR glad_glObjectPtrLabelKHR
typedef void (APIENTRYP PFNGLGETOBJECTPTRLABELKHRPROC)(const void *ptr, GLsizei bufSize, GLsizei *length, GLchar *label);
GLAPI PFNGLGETOBJECTPTRLABELKHRPROC glad_glGetObjectPtrLabelKHR;
#define glGetObjectPtrLabelKHR glad_glGetObjectPtrLabelKHR
typedef void (APIENTRYP PFNGLGETPOINTERVKHRPROC)(GLenum pname, void **params);
GLAPI PFNGLGETPOINTERVKHRPROC glad_glGetPointervKHR;
#define glGetPointervKHR glad_glGetPointervKHR
#endif
#ifndef GL_KHR_no_error
#define GL_KHR_no_error 1
GLAPI int GLAD_GL_KHR_no_error;
#endif
#ifndef GL_KHR_parallel_shader_compile
#define GL_KHR_parallel_shader_compile 1
GLAPI int GLAD_GL_KHR_parallel_shader_compile;
typedef void (APIENTRYP PFNGLMAXSHADERCOMPILERTHREADSKHRPROC)(GLuint count);
GLAPI PFNGLMAXSHADERCOMPILERTHREADSKHRPROC glad_glMaxShaderCompilerThreadsKHR;
#define glMaxShaderCompilerThreadsKHR glad_glMaxShaderCompilerThreadsKHR
#endif
#ifndef GL_KHR_robust_buffer_access_behavior
#define GL_KHR_robust_buffer_access_behavior 1
GLAPI int GLAD_GL_KHR_robust_buffer_access_behavior;
#endif
#ifndef GL_KHR_robustness
#define GL_KHR_robustness 1
GLAPI int GLAD_GL_KHR_robustness;
typedef GLenum (APIENTRYP PFNGLGETGRAPHICSRESETSTATUSPROC)(void);
GLAPI PFNGLGETGRAPHICSRESETSTATUSPROC glad_glGetGraphicsResetStatus;
#define glGetGraphicsResetStatus glad_glGetGraphicsResetStatus
typedef void (APIENTRYP PFNGLREADNPIXELSPROC)(GLint x, GLint y, GLsizei width, GLsizei height, GLenum format, GLenum type, GLsizei bufSize, void *data);
GLAPI PFNGLREADNPIXELSPROC glad_glReadnPixels;
#define glReadnPixels glad_glReadnPixels
typedef void (APIENTRYP PFNGLGETNUNIFORMFVPROC)(GLuint program, GLint location, GLsizei bufSize, GLfloat *params);
GLAPI PFNGLGETNUNIFORMFVPROC glad_glGetnUniformfv;
#define glGetnUniformfv glad_glGetnUniformfv
typedef void (APIENTRYP PFNGLGETNUNIFORMIVPROC)(GLuint program, GLint location, GLsizei bufSize, GLint *params);
GLAPI PFNGLGETNUNIFORMIVPROC glad_glGetnUniformiv;
#define glGetnUniformiv glad_glGetnUniformiv
typedef void (APIENTRYP PFNGLGETNUNIFORMUIVPROC)(GLuint program, GLint location, GLsizei bufSize, GLuint *params);
GLAPI PFNGLGETNUNIFORMUIVPROC glad_glGetnUniformuiv;
#define glGetnUniformuiv glad_glGetnUniformuiv
typedef GLenum (APIENTRYP PFNGLGETGRAPHICSRESETSTATUSKHRPROC)(void);
GLAPI PFNGLGETGRAPHICSRESETSTATUSKHRPROC glad_glGetGraphicsResetStatusKHR;
#define glGetGraphicsResetStatusKHR glad_glGetGraphicsResetStatusKHR
typedef void (APIENTRYP PFNGLREADNPIXELSKHRPROC)(GLint x, GLint y, GLsizei width, GLsizei height, GLenum format, GLenum type, GLsizei bufSize, void *data);
GLAPI PFNGLREADNPIXELSKHRPROC glad_glReadnPixelsKHR;
#define glReadnPixelsKHR glad_glReadnPixelsKHR
typedef void (APIENTRYP PFNGLGETNUNIFORMFVKHRPROC)(GLuint program, GLint location, GLsizei bufSize, GLfloat *params);
GLAPI PFNGLGETNUNIFORMFVKHRPROC glad_glGetnUniformfvKHR;
#define glGetnUniformfvKHR glad_glGetnUniformfvKHR
typedef void (APIENTRYP PFNGLGETNUNIFORMIVKHRPROC)(GLuint program, GLint location, GLsizei bufSize, GLint *params);
GLAPI PFNGLGETNUNIFORMIVKHRPROC glad_glGetnUniformivKHR;
#define glGetnUniformivKHR glad_glGetnUniformivKHR
typedef void (APIENTRYP PFNGLGETNUNIFORMUIVKHRPROC)(GLuint program, GLint location, GLsizei bufSize, GLuint *params);
GLAPI PFNGLGETNUNIFORMUIVKHRPROC glad_glGetnUniformuivKHR;
#define glGetnUniformuivKHR glad_glGetnUniformuivKHR
#endif
#ifndef GL_KHR_shader_subgroup
#define GL_KHR_shader_subgroup 1
GLAPI int GLAD_GL_KHR_shader_subgroup;
#endif
#ifndef GL_KHR_texture_compression_astc_hdr
#define GL_KHR_texture_compression_astc_hdr 1
GLAPI int GLAD_GL_KHR_texture_compression_astc_hdr;
#endif
#ifndef GL_KHR_texture_compression_astc_ldr
#define GL_KHR_texture_compression_astc_ldr 1
GLAPI int GLAD_GL_KHR_texture_compression_astc_ldr;
#endif
#ifndef GL_KHR_texture_compression_astc_sliced_3d
#define GL_KHR_texture_compression_astc_sliced_3d 1
GLAPI int GLAD_GL_KHR_texture_compression_astc_sliced_3d;
#endif
#ifndef GL_MESAX_texture_stack
#define GL_MESAX_texture_stack 1
GLAPI int GLAD_GL_MESAX_texture_stack;
#endif
```

```
#ifndef GL_MESA_framebuffer_flip_x
#define GL_MESA_framebuffer_flip_x 1
GLAPI int GLAD_GL_MESA_framebuffer_flip_x;
#endif
#ifndef GL_MESA_framebuffer_flip_y
#define GL_MESA_framebuffer_flip_y 1
GLAPI int GLAD_GL_MESA_framebuffer_flip_y;
typedef void (APIENTRYP PFNGLFRAMEBUFFERPARAMETERIMESAPROC)(GLenum target, GLenum pname, GLint param);
GLAPI PFNGLFRAMEBUFFERPARAMETERIMESAPROC glad_glFramebufferParameteriMESA;
#define glFramebufferParameteriMESA glad_glFramebufferParameteriMESA
typedef void (APIENTRYP PFNGLGETFRAMEBUFFERPARAMETERIVMESAPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETFRAMEBUFFERPARAMETERIVMESAPROC glad_glGetFramebufferParameterivMESA;
#define glGetFramebufferParameterivMESA glad_glGetFramebufferParameterivMESA
#endif
#ifndef GL_MESA_framebuffer_swap_xy
#define GL_MESA_framebuffer_swap_xy 1
GLAPI int GLAD_GL_MESA_framebuffer_swap_xy;
#endif
#ifndef GL_MESA_pack_invert
#define GL_MESA_pack_invert 1
GLAPI int GLAD_GL_MESA_pack_invert;
#endif
#ifndef GL_MESA_program_binary_formats
#define GL_MESA_program_binary_formats 1
GLAPI int GLAD_GL_MESA_program_binary_formats;
#endif
#ifndef GL_MESA_resize_buffers
#define GL_MESA_resize_buffers 1
GLAPI int GLAD_GL_MESA_resize_buffers;
typedef void (APIENTRYP PFNGLRESIZEBUFFERSMESAPROC)(void);
GLAPI PFNGLRESIZEBUFFERSMESAPROC glad_glResizeBuffersMESA;
#define glResizeBuffersMESA glad_glResizeBuffersMESA
#endif
#ifndef GL_MESA_shader_integer_functions
#define GL_MESA_shader_integer_functions 1
GLAPI int GLAD_GL_MESA_shader_integer_functions;
#endif
#ifndef GL_MESA_tile_raster_order
#define GL_MESA_tile_raster_order 1
GLAPI int GLAD_GL_MESA_tile_raster_order;
#endif
#ifndef GL_MESA_window_pos
#define GL_MESA_window_pos 1
GLAPI int GLAD_GL_MESA_window_pos;
typedef void (APIENTRYP PFNGLWINDOWPOS2DMESAPROC)(GLdouble x, GLdouble y);
GLAPI PFNGLWINDOWPOS2DMESAPROC glad_glWindowPos2dMESA;
#define glWindowPos2dMESA glad_glWindowPos2dMESA
typedef void (APIENTRYP PFNGLWINDOWPOS2DVMESAPROC)(const GLdouble *v);
GLAPI PFNGLWINDOWPOS2DVMESAPROC glad_glWindowPos2dvMESA;
#define glWindowPos2dvMESA glad_glWindowPos2dvMESA
typedef void (APIENTRYP PFNGLWINDOWPOS2FMESAPROC)(GLfloat x, GLfloat y);
GLAPI PFNGLWINDOWPOS2FMESAPROC glad_glWindowPos2fMESA;
#define glWindowPos2fMESA glad_glWindowPos2fMESA
typedef void (APIENTRYP PFNGLWINDOWPOS2FVMESAPROC)(const GLfloat *v);
GLAPI PFNGLWINDOWPOS2FVMESAPROC glad_glWindowPos2fvMESA;
#define glWindowPos2fvMESA glad_glWindowPos2fvMESA
typedef void (APIENTRYP PFNGLWINDOWPOS2IMESAPROC)(GLint x, GLint y);
GLAPI PFNGLWINDOWPOS2IMESAPROC glad_glWindowPos2iMESA;
#define glWindowPos2iMESA glad_glWindowPos2iMESA
typedef void (APIENTRYP PFNGLWINDOWPOS2IVMESAPROC)(const GLint *v);
GLAPI PFNGLWINDOWPOS2IVMESAPROC glad_glWindowPos2ivMESA;
#define glWindowPos2ivMESA glad_glWindowPos2ivMESA
typedef void (APIENTRYP PFNGLWINDOWPOS2SMESAPROC)(GLshort x, GLshort y);
GLAPI PFNGLWINDOWPOS2SMESAPROC glad_glWindowPos2sMESA;
#define glWindowPos2sMESA glad_glWindowPos2sMESA
typedef void (APIENTRYP PFNGLWINDOWPOS2SVMESAPROC)(const GLshort *v);
GLAPI PFNGLWINDOWPOS2SVMESAPROC glad_glWindowPos2svMESA;
#define glWindowPos2svMESA glad_glWindowPos2svMESA
typedef void (APIENTRYP PFNGLWINDOWPOS3DMESAPROC)(GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLWINDOWPOS3DMESAPROC glad_glWindowPos3dMESA;
#define glWindowPos3dMESA glad_glWindowPos3dMESA
typedef void (APIENTRYP PFNGLWINDOWPOS3DVMESAPROC)(const GLdouble *v);
GLAPI PFNGLWINDOWPOS3DVMESAPROC glad_glWindowPos3dvMESA;
#define glWindowPos3dvMESA glad_glWindowPos3dvMESA
typedef void (APIENTRYP PFNGLWINDOWPOS3FMESAPROC)(GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLWINDOWPOS3FMESAPROC glad_glWindowPos3fMESA;
#define glWindowPos3fMESA glad_glWindowPos3fMESA
typedef void (APIENTRYP PFNGLWINDOWPOS3FVMESAPROC)(const GLfloat *v);
GLAPI PFNGLWINDOWPOS3FVMESAPROC glad_glWindowPos3fvMESA;
#define glWindowPos3fvMESA glad_glWindowPos3fvMESA
typedef void (APIENTRYP PFNGLWINDOWPOS3IMESAPROC)(GLint x, GLint y, GLint z);
GLAPI PFNGLWINDOWPOS3IMESAPROC glad_glWindowPos3iMESA;
#define glWindowPos3iMESA glad_glWindowPos3iMESA
typedef void (APIENTRYP PFNGLWINDOWPOS3IVMESAPROC)(const GLint *v);
GLAPI PFNGLWINDOWPOS3IVMESAPROC glad_glWindowPos3ivMESA;
#define glWindowPos3ivMESA glad_glWindowPos3ivMESA
```

```
typedef void (APIENTRYP PFNGLWINDOWPOS3SMESAPROC)(GLshort x, GLshort y, GLshort z);
GLAPI PFNGLWINDOWPOS3SMESAPROC glad_glWindowPos3sMESA;
#define glWindowPos3sMESA glad_glWindowPos3sMESA
typedef void (APIENTRYP PFNGLWINDOWPOS3SVMESAPROC)(const GLshort *v);
GLAPI PFNGLWINDOWPOS3SVMESAPROC glad_glWindowPos3svMESA;
#define glWindowPos3svMESA glad_glWindowPos3svMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4DMESAPROC)(GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLWINDOWPOS4DMESAPROC glad_glWindowPos4dMESA;
#define glWindowPos4dMESA glad_glWindowPos4dMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4DVMESAPROC)(const GLdouble *v);
GLAPI PFNGLWINDOWPOS4DVMESAPROC glad_glWindowPos4dvMESA;
#define glWindowPos4dvMESA glad_glWindowPos4dvMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4FMESAPROC)(GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLWINDOWPOS4FMESAPROC glad_glWindowPos4fMESA;
#define glWindowPos4fMESA glad_glWindowPos4fMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4FVMESAPROC)(const GLfloat *v);
GLAPI PFNGLWINDOWPOS4FVMESAPROC glad_glWindowPos4fvMESA;
#define glWindowPos4fvMESA glad_glWindowPos4fvMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4IMESAPROC)(GLint x, GLint y, GLint z, GLint w);
GLAPI PFNGLWINDOWPOS4IMESAPROC glad_glWindowPos4iMESA;
#define glWindowPos4iMESA glad_glWindowPos4iMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4IVMESAPROC)(const GLint *v);
GLAPI PFNGLWINDOWPOS4IVMESAPROC glad_glWindowPos4ivMESA;
#define glWindowPos4ivMESA glad_glWindowPos4ivMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4SMESAPROC)(GLshort x, GLshort y, GLshort z, GLshort w);
GLAPI PFNGLWINDOWPOS4SMESAPROC glad_glWindowPos4sMESA;
#define glWindowPos4sMESA glad_glWindowPos4sMESA
typedef void (APIENTRYP PFNGLWINDOWPOS4SVMESAPROC)(const GLshort *v);
GLAPI PFNGLWINDOWPOS4SVMESAPROC glad_glWindowPos4svMESA;
#define glWindowPos4svMESA glad_glWindowPos4svMESA
#endif
#ifndef GL_MESA_ycbcr_texture
#define GL_MESA_ycbcr_texture 1
GLAPI int GLAD_GL_MESA_ycbcr_texture;
#endif
#ifndef GL_NVX_blend_equation_advanced_multi_draw_buffers
#define GL_NVX_blend_equation_advanced_multi_draw_buffers 1
GLAPI int GLAD_GL_NVX_blend_equation_advanced_multi_draw_buffers;
#endif
#ifndef GL_NVX_conditional_render
#define GL_NVX_conditional_render 1
GLAPI int GLAD_GL_NVX_conditional_render;
typedef void (APIENTRYP PFNGLBEGINCONDITIONALRENDERNVXPROC)(GLuint id);
GLAPI PFNGLBEGINCONDITIONALRENDERNVXPROC glad_glBeginConditionalRenderNVX;
#define glBeginConditionalRenderNVX glad_glBeginConditionalRenderNVX
typedef void (APIENTRYP PFNGLENDCONDITIONALRENDERNVXPROC)(void);
GLAPI PFNGLENDCONDITIONALRENDERNVXPROC glad_glEndConditionalRenderNVX;
#define glEndConditionalRenderNVX glad_glEndConditionalRenderNVX
#endif
#ifndef GL_NVX_gpu_memory_info
#define GL_NVX_gpu_memory_info 1
GLAPI int GLAD_GL_NVX_gpu_memory_info;
#endif
#ifndef GL_NVX_gpu_multicast2
#define GL_NVX_gpu_multicast2 1
GLAPI int GLAD_GL_NVX_gpu_multicast2;
typedef void (APIENTRYP PFNGLUPLOADGPUMASKNVXPROC)(GLbitfield mask);
GLAPI PFNGLUPLOADGPUMASKNVXPROC glad_glUploadGpuMaskNVX;
#define glUploadGpuMaskNVX glad_glUploadGpuMaskNVX
typedef void (APIENTRYP PFNGLMULTICASTVIEWPORTARRAYVNVXPROC)(GLuint gpu, GLuint first, GLsizei count, const GLfloat *v);
GLAPI PFNGLMULTICASTVIEWPORTARRAYVNVXPROC glad_glMulticastViewportArrayvNVX;
#define glMulticastViewportArrayvNVX glad_glMulticastViewportArrayvNVX
typedef void (APIENTRYP PFNGLMULTICASTVIEWPORTPOSITIONWSCALENVXPROC)(GLuint gpu, GLuint index, GLfloat xcoeff, GLfloat ycoeff);
GLAPI PFNGLMULTICASTVIEWPORTPOSITIONWSCALENVXPROC glad_glMulticastViewportPositionWScaleNVX;
#define glMulticastViewportPositionWScaleNVX glad_glMulticastViewportPositionWScaleNVX
typedef void (APIENTRYP PFNGLMULTICASTSCISSORARRAYVNVXPROC)(GLuint gpu, GLuint first, GLsizei count, const GLint *v);
GLAPI PFNGLMULTICASTSCISSORARRAYVNVXPROC glad_glMulticastScissorArrayvNVX;
#define glMulticastScissorArrayvNVX glad_glMulticastScissorArrayvNVX
typedef GLuint (APIENTRYP 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);
GLAPI PFNGLASYNCCOPYBUFFERSUBDATANVXPROC glad_glAsyncCopyBufferSubDataNVX;
#define glAsyncCopyBufferSubDataNVX glad_glAsyncCopyBufferSubDataNVX
typedef GLuint (APIENTRYP 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);
GLAPI PFNGLASYNCCOPYIMAGESUBDATANVXPROC glad_glAsyncCopyImageSubDataNVX;
#define glAsyncCopyImageSubDataNVX glad_glAsyncCopyImageSubDataNVX
#endif
#ifndef GL_NVX_linked_gpu_multicast
```

```
#define GL_NVX_linked_gpu_multicast 1
GLAPI int GLAD_GL_NVX_linked_gpu_multicast;
typedef void (APIENTRYP PFNGLLGPUNAMEDBUFFERSUBDATANVXPROC)(GLbitfield gpuMask, GLuint buffer, GLintptr offset, GLsizeiptr size, const void *data);
GLAPI PFNGLLGPUNAMEDBUFFERSUBDATANVXPROC glad_glLGPUNamedBufferSubDataNVX;
#define glLGPUNamedBufferSubDataNVX glad_glLGPUNamedBufferSubDataNVX
typedef void (APIENTRYP PFNGLLGPUCOPYIMAGESUBDATANVXPROC)(GLuint sourceGpu, GLbitfield destinationGpuMask, GLuint srcName, GLenum srcTarget, GLint srcLevel, GLint srcX, GLint srxY, GLint srcZ, GLuint dstName, GLenum dstTarget, GLint dstLevel, GLint dstX, GLint dstY, GLint dstZ, GLsizei width, GLsizei height, GLsizei depth);
GLAPI PFNGLLGPUCOPYIMAGESUBDATANVXPROC glad_glLGPUCopyImageSubDataNVX;
#define glLGPUCopyImageSubDataNVX glad_glLGPUCopyImageSubDataNVX
typedef void (APIENTRYP PFNGLLGPUINTERLOCKNVXPROC)(void);
GLAPI PFNGLLGPUINTERLOCKNVXPROC glad_glLGPUInterlockNVX;
#define glLGPUInterlockNVX glad_glLGPUInterlockNVX
#endif
#ifndef GL_NVX_progress_fence
#define GL_NVX_progress_fence 1
GLAPI int GLAD_GL_NVX_progress_fence;
typedef GLuint (APIENTRYP PFNGLCREATEPROGRESSFENCENVXPROC)(void);
GLAPI PFNGLCREATEPROGRESSFENCENVXPROC glad_glCreateProgressFenceNVX;
#define glCreateProgressFenceNVX glad_glCreateProgressFenceNVX
typedef void (APIENTRYP PFNGLSIGNALSEMAPHOREUI64NVXPROC)(GLuint signalGpu, GLsizei fenceObjectCount, const GLuint *semaphoreArray, const GLuint64 *fenceValueArray);
GLAPI PFNGLSIGNALSEMAPHOREUI64NVXPROC glad_glSignalSemaphoreui64NVX;
#define glSignalSemaphoreui64NVX glad_glSignalSemaphoreui64NVX
typedef void (APIENTRYP PFNGLWAITSEMAPHOREUI64NVXPROC)(GLuint waitGpu, GLsizei fenceObjectCount, const GLuint *semaphoreArray, const GLuint64 *fenceValueArray);
GLAPI PFNGLWAITSEMAPHOREUI64NVXPROC glad_glWaitSemaphoreui64NVX;
#define glWaitSemaphoreui64NVX glad_glWaitSemaphoreui64NVX
typedef void (APIENTRYP PFNGLCLIENTWAITSEMAPHOREUI64NVXPROC)(GLsizei fenceObjectCount, const GLuint *semaphoreArray, const GLuint64 *fenceValueArray);
GLAPI PFNGLCLIENTWAITSEMAPHOREUI64NVXPROC glad_glClientWaitSemaphoreui64NVX;
#define glClientWaitSemaphoreui64NVX glad_glClientWaitSemaphoreui64NVX
#endif
#ifndef GL_NV_alpha_to_coverage_dither_control
#define GL_NV_alpha_to_coverage_dither_control 1
GLAPI int GLAD_GL_NV_alpha_to_coverage_dither_control;
typedef void (APIENTRYP PFNGLALPHATOCOVERAGEDITHERCONTROLNVPROC)(GLenum mode);
GLAPI PFNGLALPHATOCOVERAGEDITHERCONTROLNVPROC glad_glAlphaToCoverageDitherControlNV;
#define glAlphaToCoverageDitherControlNV glad_glAlphaToCoverageDitherControlNV
#endif
#ifndef GL_NV_bindless_multi_draw_indirect
#define GL_NV_bindless_multi_draw_indirect 1
GLAPI int GLAD_GL_NV_bindless_multi_draw_indirect;
typedef void (APIENTRYP PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSNVPROC)(GLenum mode, const void *indirect, GLsizei drawCount, GLsizei stride, GLint vertexBufferCount);
GLAPI PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSNVPROC glad_glMultiDrawArraysIndirectBindlessNV;
#define glMultiDrawArraysIndirectBindlessNV glad_glMultiDrawArraysIndirectBindlessNV
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSNVPROC)(GLenum mode, GLenum type, const void *indirect, GLsizei drawCount, GLsizei stride, GLint vertexBufferCount);
GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSNVPROC glad_glMultiDrawElementsIndirectBindlessNV;
#define glMultiDrawElementsIndirectBindlessNV glad_glMultiDrawElementsIndirectBindlessNV
#endif
#ifndef GL_NV_bindless_multi_draw_indirect_count
#define GL_NV_bindless_multi_draw_indirect_count 1
GLAPI int GLAD_GL_NV_bindless_multi_draw_indirect_count;
typedef void (APIENTRYP PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSCOUNTNVPROC)(GLenum mode, const void *indirect, GLsizei drawCount, GLsizei maxDrawCount, GLsizei stride, GLint vertexBufferCount);
GLAPI PFNGLMULTIDRAWARRAYSINDIRECTBINDLESSCOUNTNVPROC glad_glMultiDrawArraysIndirectBindlessCountNV;
#define glMultiDrawArraysIndirectBindlessCountNV glad_glMultiDrawArraysIndirectBindlessCountNV
typedef void (APIENTRYP PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSCOUNTNVPROC)(GLenum mode, GLenum type, const void *indirect, GLsizei drawCount, GLsizei maxDrawCount, GLsizei stride, GLint vertexBufferCount);
GLAPI PFNGLMULTIDRAWELEMENTSINDIRECTBINDLESSCOUNTNVPROC glad_glMultiDrawElementsIndirectBindlessCountNV;
#define glMultiDrawElementsIndirectBindlessCountNV glad_glMultiDrawElementsIndirectBindlessCountNV
#endif
#ifndef GL_NV_bindless_texture
#define GL_NV_bindless_texture 1
GLAPI int GLAD_GL_NV_bindless_texture;
typedef GLuint64 (APIENTRYP PFNGLGETTEXTUREHANDLENVPROC)(GLuint texture);
GLAPI PFNGLGETTEXTUREHANDLENVPROC glad_glGetTextureHandleNV;
#define glGetTextureHandleNV glad_glGetTextureHandleNV
typedef GLuint64 (APIENTRYP PFNGLGETTEXTURESAMPLERHANDLENVPROC)(GLuint texture, GLuint sampler);
GLAPI PFNGLGETTEXTURESAMPLERHANDLENVPROC glad_glGetTextureSamplerHandleNV;
#define glGetTextureSamplerHandleNV glad_glGetTextureSamplerHandleNV
typedef void (APIENTRYP PFNGLMAKETEXTUREHANDLERESIDENTNVPROC)(GLuint64 handle);
GLAPI PFNGLMAKETEXTUREHANDLERESIDENTNVPROC glad_glMakeTextureHandleResidentNV;
#define glMakeTextureHandleResidentNV glad_glMakeTextureHandleResidentNV
typedef void (APIENTRYP PFNGLMAKETEXTUREHANDLENONRESIDENTNVPROC)(GLuint64 handle);
GLAPI PFNGLMAKETEXTUREHANDLENONRESIDENTNVPROC glad_glMakeTextureHandleNonResidentNV;
#define glMakeTextureHandleNonResidentNV glad_glMakeTextureHandleNonResidentNV
typedef GLuint64 (APIENTRYP PFNGLGETIMAGEHANDLENVPROC)(GLuint texture, GLint level, GLboolean layered, GLint layer, GLenum format);
GLAPI PFNGLGETIMAGEHANDLENVPROC glad_glGetImageHandleNV;
#define glGetImageHandleNV glad_glGetImageHandleNV
```

```
typedef void (APIENTRYP PFNGLMAKEIMAGEHANDLERESIDENTNVPROC)(GLuint64 handle, GLenum access);
GLAPI PFNGLMAKEIMAGEHANDLERESIDENTNVPROC glad_glMakeImageHandleResidentNV;
#define glMakeImageHandleResidentNV glad_glMakeImageHandleResidentNV
typedef void (APIENTRYP PFNGLMAKEIMAGEHANDLENONRESIDENTNVPROC)(GLuint64 handle);
GLAPI PFNGLMAKEIMAGEHANDLENONRESIDENTNVPROC glad_glMakeImageHandleNonResidentNV;
#define glMakeImageHandleNonResidentNV glad_glMakeImageHandleNonResidentNV
typedef void (APIENTRYP PFNGLUNIFORMHANDLEUI64NVPROC)(GLint location, GLuint64 value);
GLAPI PFNGLUNIFORMHANDLEUI64NVPROC glad_glUniformHandleui64NV;
#define glUniformHandleui64NV glad_glUniformHandleui64NV
typedef void (APIENTRYP PFNGLUNIFORMHANDLEUI64VNVPROC)(GLint location, GLsizei count, const GLuint64 *value);
GLAPI PFNGLUNIFORMHANDLEUI64VNVPROC glad_glUniformHandleui64vNV;
#define glUniformHandleui64vNV glad_glUniformHandleui64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMHANDLEUI64NVPROC)(GLuint program, GLint location, GLuint64 value);
GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64NVPROC glad_glProgramUniformHandleui64NV;
#define glProgramUniformHandleui64NV glad_glProgramUniformHandleui64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMHANDLEUI64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLuint64 *values);
GLAPI PFNGLPROGRAMUNIFORMHANDLEUI64VNVPROC glad_glProgramUniformHandleui64vNV;
#define glProgramUniformHandleui64vNV glad_glProgramUniformHandleui64vNV
typedef GLboolean (APIENTRYP PFNGLISTEXTUREHANDLERESIDENTNVPROC)(GLuint64 handle);
GLAPI PFNGLISTEXTUREHANDLERESIDENTNVPROC glad_glIsTextureHandleResidentNV;
#define glIsTextureHandleResidentNV glad_glIsTextureHandleResidentNV
typedef GLboolean (APIENTRYP PFNGLISIMAGEHANDLERESIDENTNVPROC)(GLuint64 handle);
GLAPI PFNGLISIMAGEHANDLERESIDENTNVPROC glad_glIsImageHandleResidentNV;
#define glIsImageHandleResidentNV glad_glIsImageHandleResidentNV
#endif
#ifndef GL_NV_blend_equation_advanced
#define GL_NV_blend_equation_advanced 1
GLAPI int GLAD_GL_NV_blend_equation_advanced;
typedef void (APIENTRYP PFNGLBLENDPARAMETERINVPROC)(GLenum pname, GLint value);
GLAPI PFNGLBLENDPARAMETERINVPROC glad_glBlendParameteriNV;
#define glBlendParameteriNV glad_glBlendParameteriNV
typedef void (APIENTRYP PFNGLBLENDBARRIERNVPROC)(void);
GLAPI PFNGLBLENDBARRIERNVPROC glad_glBlendBarrierNV;
#define glBlendBarrierNV glad_glBlendBarrierNV
#endif
#ifndef GL_NV_blend_equation_advanced_coherent
#define GL_NV_blend_equation_advanced_coherent 1
GLAPI int GLAD_GL_NV_blend_equation_advanced_coherent;
#endif
#ifndef GL_NV_blend_minmax_factor
#define GL_NV_blend_minmax_factor 1
GLAPI int GLAD_GL_NV_blend_minmax_factor;
#endif
#ifndef GL_NV_blend_square
#define GL_NV_blend_square 1
GLAPI int GLAD_GL_NV_blend_square;
#endif
#ifndef GL_NV_clip_space_w_scaling
#define GL_NV_clip_space_w_scaling 1
GLAPI int GLAD_GL_NV_clip_space_w_scaling;
typedef void (APIENTRYP PFNGLVIEWPORTPOSITIONWSCALENVPROC)(GLuint index, GLfloat xcoeff, GLfloat ycoeff);
GLAPI PFNGLVIEWPORTPOSITIONWSCALENVPROC glad_glViewportPositionWScaleNV;
#define glViewportPositionWScaleNV glad_glViewportPositionWScaleNV
#endif
#ifndef GL_NV_command_list
#define GL_NV_command_list 1
GLAPI int GLAD_GL_NV_command_list;
typedef void (APIENTRYP PFNGLCREATESTATESNVPROC)(GLsizei n, GLuint *states);
GLAPI PFNGLCREATESTATESNVPROC glad_glCreateStatesNV;
#define glCreateStatesNV glad_glCreateStatesNV
typedef void (APIENTRYP PFNGLDELETESTATESNVPROC)(GLsizei n, const GLuint *states);
GLAPI PFNGLDELETESTATESNVPROC glad_glDeleteStatesNV;
#define glDeleteStatesNV glad_glDeleteStatesNV
typedef GLboolean (APIENTRYP PFNGLISSTATENVPROC)(GLuint state);
GLAPI PFNGLISSTATENVPROC glad_glIsStateNV;
#define glIsStateNV glad_glIsStateNV
typedef void (APIENTRYP PFNGLSTATECAPTURENVPROC)(GLuint state, GLenum mode);
GLAPI PFNGLSTATECAPTURENVPROC glad_glStateCaptureNV;
#define glStateCaptureNV glad_glStateCaptureNV
typedef GLuint (APIENTRYP PFNGLGETCOMMANDHEADERNVPROC)(GLenum tokenID, GLuint size);
GLAPI PFNGLGETCOMMANDHEADERNVPROC glad_glGetCommandHeaderNV;
#define glGetCommandHeaderNV glad_glGetCommandHeaderNV
typedef GLushort (APIENTRYP PFNGLGETSTAGEINDEXNVPROC)(GLenum shadertype);
GLAPI PFNGLGETSTAGEINDEXNVPROC glad_glGetStageIndexNV;
#define glGetStageIndexNV glad_glGetStageIndexNV
typedef void (APIENTRYP PFNGLDRAWCOMMANDSNVPROC)(GLenum primitiveMode, GLuint buffer, const GLintptr *indirects, const GLsizei *sizes, GLuint count);
GLAPI PFNGLDRAWCOMMANDSNVPROC glad_glDrawCommandsNV;
#define glDrawCommandsNV glad_glDrawCommandsNV
typedef void (APIENTRYP PFNGLDRAWCOMMANDSADDRESSNVPROC)(GLenum primitiveMode, const GLuint64 *indirects, const GLsizei *sizes, GLuint count);
GLAPI PFNGLDRAWCOMMANDSADDRESSNVPROC glad_glDrawCommandsAddressNV;
#define glDrawCommandsAddressNV glad_glDrawCommandsAddressNV
```

```
typedef void (APIENTRYP PFNGLDRAWCOMMANDSSTATESNVPROC)(GLuint buffer, const GLintptr *indirects, const GLsizei *sizes, const GLuint *states, const GLuint *fbos, GLuint count);
GLAPI PFNGLDRAWCOMMANDSSTATESNVPROC glad_glDrawCommandsStatesNV;
#define glDrawCommandsStatesNV glad_glDrawCommandsStatesNV
typedef void (APIENTRYP PFNGLDRAWCOMMANDSSTATESADDRESSNVPROC)(const GLuint64 *indirects, const GLsizei *sizes, const GLuint *states, const GLuint *fbos, GLuint count);
GLAPI PFNGLDRAWCOMMANDSSTATESADDRESSNVPROC glad_glDrawCommandsStatesAddressNV;
#define glDrawCommandsStatesAddressNV glad_glDrawCommandsStatesAddressNV
typedef void (APIENTRYP PFNGLCREATECOMMANDLISTSNVPROC)(GLsizei n, GLuint *lists);
GLAPI PFNGLCREATECOMMANDLISTSNVPROC glad_glCreateCommandListsNV;
#define glCreateCommandListsNV glad_glCreateCommandListsNV
typedef void (APIENTRYP PFNGLDELETECOMMANDLISTSNVPROC)(GLsizei n, const GLuint *lists);
GLAPI PFNGLDELETECOMMANDLISTSNVPROC glad_glDeleteCommandListsNV;
#define glDeleteCommandListsNV glad_glDeleteCommandListsNV
typedef GLboolean (APIENTRYP PFNGLISCOMMANDLISTNVPROC)(GLuint list);
GLAPI PFNGLISCOMMANDLISTNVPROC glad_glIsCommandListNV;
#define glIsCommandListNV glad_glIsCommandListNV
typedef void (APIENTRYP PFNGLLISTDRAWCOMMANDSSTATESCLIENTNVPROC)(GLuint list, GLuint segment, const void **indirects, const GLsizei *sizes, const GLuint *states, const GLuint *fbos, GLuint count);
GLAPI PFNGLLISTDRAWCOMMANDSSTATESCLIENTNVPROC glad_glListDrawCommandsStatesClientNV;
#define glListDrawCommandsStatesClientNV glad_glListDrawCommandsStatesClientNV
typedef void (APIENTRYP PFNGLCOMMANDLISTSEGMENTSNVPROC)(GLuint list, GLuint segments);
GLAPI PFNGLCOMMANDLISTSEGMENTSNVPROC glad_glCommandListSegmentsNV;
#define glCommandListSegmentsNV glad_glCommandListSegmentsNV
typedef void (APIENTRYP PFNGLCOMPILECOMMANDLISTNVPROC)(GLuint list);
GLAPI PFNGLCOMPILECOMMANDLISTNVPROC glad_glCompileCommandListNV;
#define glCompileCommandListNV glad_glCompileCommandListNV
typedef void (APIENTRYP PFNGLCALLCOMMANDLISTNVPROC)(GLuint list);
GLAPI PFNGLCALLCOMMANDLISTNVPROC glad_glCallCommandListNV;
#define glCallCommandListNV glad_glCallCommandListNV
#endif
#ifndef GL_NV_compute_program5
#define GL_NV_compute_program5 1
GLAPI int GLAD_GL_NV_compute_program5;
#endif
#ifndef GL_NV_compute_shader_derivatives
#define GL_NV_compute_shader_derivatives 1
GLAPI int GLAD_GL_NV_compute_shader_derivatives;
#endif
#ifndef GL_NV_conditional_render
#define GL_NV_conditional_render 1
GLAPI int GLAD_GL_NV_conditional_render;
typedef void (APIENTRYP PFNGLBEGINCONDITIONALRENDERNVPROC)(GLuint id, GLenum mode);
GLAPI PFNGLBEGINCONDITIONALRENDERNVPROC glad_glBeginConditionalRenderNV;
#define glBeginConditionalRenderNV glad_glBeginConditionalRenderNV
typedef void (APIENTRYP PFNGLENDCONDITIONALRENDERNVPROC)(void);
GLAPI PFNGLENDCONDITIONALRENDERNVPROC glad_glEndConditionalRenderNV;
#define glEndConditionalRenderNV glad_glEndConditionalRenderNV
#endif
#ifndef GL_NV_conservative_raster
#define GL_NV_conservative_raster 1
GLAPI int GLAD_GL_NV_conservative_raster;
typedef void (APIENTRYP PFNGLSUBPIXELPRECISIONBIASNVPROC)(GLuint xbits, GLuint ybits);
GLAPI PFNGLSUBPIXELPRECISIONBIASNVPROC glad_glSubpixelPrecisionBiasNV;
#define glSubpixelPrecisionBiasNV glad_glSubpixelPrecisionBiasNV
#endif
#ifndef GL_NV_conservative_raster_dilate
#define GL_NV_conservative_raster_dilate 1
GLAPI int GLAD_GL_NV_conservative_raster_dilate;
typedef void (APIENTRYP PFNGLCONSERVATIVERASTERPARAMETERFNVPROC)(GLenum pname, GLfloat value);
GLAPI PFNGLCONSERVATIVERASTERPARAMETERFNVPROC glad_glConservativeRasterParameterfNV;
#define glConservativeRasterParameterfNV glad_glConservativeRasterParameterfNV
#endif
#ifndef GL_NV_conservative_raster_pre_snap
#define GL_NV_conservative_raster_pre_snap 1
GLAPI int GLAD_GL_NV_conservative_raster_pre_snap;
#endif
#ifndef GL_NV_conservative_raster_pre_snap_triangles
#define GL_NV_conservative_raster_pre_snap_triangles 1
GLAPI int GLAD_GL_NV_conservative_raster_pre_snap_triangles;
typedef void (APIENTRYP PFNGLCONSERVATIVERASTERPARAMETERINVPROC)(GLenum pname, GLint param);
GLAPI PFNGLCONSERVATIVERASTERPARAMETERINVPROC glad_glConservativeRasterParameteriNV;
#define glConservativeRasterParameteriNV glad_glConservativeRasterParameteriNV
#endif
#ifndef GL_NV_conservative_raster_underestimation
#define GL_NV_conservative_raster_underestimation 1
GLAPI int GLAD_GL_NV_conservative_raster_underestimation;
#endif
#ifndef GL_NV_copy_depth_to_color
#define GL_NV_copy_depth_to_color 1
GLAPI int GLAD_GL_NV_copy_depth_to_color;
#endif
#ifndef GL_NV_copy_image
#define GL_NV_copy_image 1
GLAPI int GLAD_GL_NV_copy_image;
typedef void (APIENTRYP 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);
GLAPI PFNGLCOPYIMAGESUBDATANVPROC glad_glCopyImageSubDataNV;
#define glCopyImageSubDataNV glad_glCopyImageSubDataNV
#endif
#ifndef GL_NV_deep_texture3D
#define GL_NV_deep_texture3D 1
GLAPI int GLAD_GL_NV_deep_texture3D;
#endif
#ifndef GL_NV_depth_buffer_float
#define GL_NV_depth_buffer_float 1
GLAPI int GLAD_GL_NV_depth_buffer_float;
typedef void (APIENTRYP PFNGLDEPTHRANGEDNVPROC)(GLdouble zNear, GLdouble zFar);
GLAPI PFNGLDEPTHRANGEDNVPROC glad_glDepthRangedNV;
#define glDepthRangedNV glad_glDepthRangedNV
typedef void (APIENTRYP PFNGLCLEARDEPTHDNVPROC)(GLdouble depth);
GLAPI PFNGLCLEARDEPTHDNVPROC glad_glClearDepthdNV;
#define glClearDepthdNV glad_glClearDepthdNV
typedef void (APIENTRYP PFNGLDEPTHBOUNDSDNVPROC)(GLdouble zmin, GLdouble zmax);
GLAPI PFNGLDEPTHBOUNDSDNVPROC glad_glDepthBoundsdNV;
#define glDepthBoundsdNV glad_glDepthBoundsdNV
#endif
#ifndef GL_NV_depth_clamp
#define GL_NV_depth_clamp 1
GLAPI int GLAD_GL_NV_depth_clamp;
#endif
#ifndef GL_NV_draw_texture
#define GL_NV_draw_texture 1
GLAPI int GLAD_GL_NV_draw_texture;
typedef void (APIENTRYP PFNGLDRAWTEXTURENVPROC)(GLuint texture, GLuint sampler, GLfloat x0, GLfloat y0, GLfloat x1, GLfloat y1, GLfloat z, GLfloat s0, GLfloat t0, GLfloat s1, GLfloat t1);
GLAPI PFNGLDRAWTEXTURENVPROC glad_glDrawTextureNV;
#define glDrawTextureNV glad_glDrawTextureNV
#endif
#ifndef GL_NV_draw_vulkan_image
#define GL_NV_draw_vulkan_image 1
GLAPI int GLAD_GL_NV_draw_vulkan_image;
typedef void (APIENTRYP PFNGLDRAWVKIMAGENVPROC)(GLuint64 vkImage, GLuint sampler, GLfloat x0, GLfloat y0, GLfloat x1, GLfloat y1, GLfloat z, GLfloat s0, GLfloat t0, GLfloat s1, GLfloat t1);
GLAPI PFNGLDRAWVKIMAGENVPROC glad_glDrawVkImageNV;
#define glDrawVkImageNV glad_glDrawVkImageNV
typedef GLVULKANPROCNV (APIENTRYP PFNGLGETVKPROCADDRNVPROC)(const GLchar *name);
GLAPI PFNGLGETVKPROCADDRNVPROC glad_glGetVkProcAddrNV;
#define glGetVkProcAddrNV glad_glGetVkProcAddrNV
typedef void (APIENTRYP PFNGLWAITVKSEMAPHORENVPROC)(GLuint64 vkSemaphore);
GLAPI PFNGLWAITVKSEMAPHORENVPROC glad_glWaitVkSemaphoreNV;
#define glWaitVkSemaphoreNV glad_glWaitVkSemaphoreNV
typedef void (APIENTRYP PFNGLSIGNALVKSEMAPHORENVPROC)(GLuint64 vkSemaphore);
GLAPI PFNGLSIGNALVKSEMAPHORENVPROC glad_glSignalVkSemaphoreNV;
#define glSignalVkSemaphoreNV glad_glSignalVkSemaphoreNV
typedef void (APIENTRYP PFNGLSIGNALVKFENCENVPROC)(GLuint64 vkFence);
GLAPI PFNGLSIGNALVKFENCENVPROC glad_glSignalVkFenceNV;
#define glSignalVkFenceNV glad_glSignalVkFenceNV
#endif
#ifndef GL_NV_evaluators
#define GL_NV_evaluators 1
GLAPI int GLAD_GL_NV_evaluators;
typedef void (APIENTRYP PFNGLMAPCONTROLPOINTSNVPROC)(GLenum target, GLuint index, GLenum type, GLsizei ustride, GLsizei vstride, GLint uorder, GLint vorder, GLboolean packed, const void *points);
GLAPI PFNGLMAPCONTROLPOINTSNVPROC glad_glMapControlPointsNV;
#define glMapControlPointsNV glad_glMapControlPointsNV
typedef void (APIENTRYP PFNGLMAPPARAMETERIVNVPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLMAPPARAMETERIVNVPROC glad_glMapParameterivNV;
#define glMapParameterivNV glad_glMapParameterivNV
typedef void (APIENTRYP PFNGLMAPPARAMETERFVNVPROC)(GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLMAPPARAMETERFVNVPROC glad_glMapParameterfvNV;
#define glMapParameterfvNV glad_glMapParameterfvNV
typedef void (APIENTRYP PFNGLGETMAPCONTROLPOINTSNVPROC)(GLenum target, GLuint index, GLenum type, GLsizei ustride, GLsizei vstride, GLboolean packed, void *points);
GLAPI PFNGLGETMAPCONTROLPOINTSNVPROC glad_glGetMapControlPointsNV;
#define glGetMapControlPointsNV glad_glGetMapControlPointsNV
typedef void (APIENTRYP PFNGLGETMAPPARAMETERIVNVPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETMAPPARAMETERIVNVPROC glad_glGetMapParameterivNV;
#define glGetMapParameterivNV glad_glGetMapParameterivNV
typedef void (APIENTRYP PFNGLGETMAPPARAMETERFVNVPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMAPPARAMETERFVNVPROC glad_glGetMapParameterfvNV;
#define glGetMapParameterfvNV glad_glGetMapParameterfvNV
typedef void (APIENTRYP PFNGLGETMAPATTRIBPARAMETERIVNVPROC)(GLenum target, GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETMAPATTRIBPARAMETERIVNVPROC glad_glGetMapAttribParameterivNV;
#define glGetMapAttribParameterivNV glad_glGetMapAttribParameterivNV
typedef void (APIENTRYP PFNGLGETMAPATTRIBPARAMETERFVNVPROC)(GLenum target, GLuint index, GLenum pname, GLfloat *params);
GLAPI PFNGLGETMAPATTRIBPARAMETERFVNVPROC glad_glGetMapAttribParameterfvNV;
#define glGetMapAttribParameterfvNV glad_glGetMapAttribParameterfvNV
typedef void (APIENTRYP PFNGLEVALMAPSNVPROC)(GLenum target, GLenum mode);
GLAPI PFNGLEVALMAPSNVPROC glad_glEvalMapsNV;
```

```
#define glEvalMapsNV glad_glEvalMapsNV
#endif
#ifndef GL_NV_explicit_multisample
#define GL_NV_explicit_multisample 1
GLAPI int GLAD_GL_NV_explicit_multisample;
typedef void (APIENTRYP PFNGLGETMULTISAMPLEFVNVPROC)(GLenum pname, GLuint index, GLfloat *val);
GLAPI PFNGLGETMULTISAMPLEFVNVPROC glad_glGetMultisamplefvNV;
#define glGetMultisamplefvNV glad_glGetMultisamplefvNV
typedef void (APIENTRYP PFNGLSAMPLEMASKINDEXEDNVPROC)(GLuint index, GLbitfield mask);
GLAPI PFNGLSAMPLEMASKINDEXEDNVPROC glad_glSampleMaskIndexedNV;
#define glSampleMaskIndexedNV glad_glSampleMaskIndexedNV
typedef void (APIENTRYP PFNGLTEXRENDERBUFFERNVPROC)(GLenum target, GLuint renderbuffer);
GLAPI PFNGLTEXRENDERBUFFERNVPROC glad_glTexRenderbufferNV;
#define glTexRenderbufferNV glad_glTexRenderbufferNV
#endif
#ifndef GL_NV_fence
#define GL_NV_fence 1
GLAPI int GLAD_GL_NV_fence;
typedef void (APIENTRYP PFNGLDELETEFENCESNVPROC)(GLsizei n, const GLuint *fences);
GLAPI PFNGLDELETEFENCESNVPROC glad_glDeleteFencesNV;
#define glDeleteFencesNV glad_glDeleteFencesNV
typedef void (APIENTRYP PFNGLGENFENCESNVPROC)(GLsizei n, GLuint *fences);
GLAPI PFNGLGENFENCESNVPROC glad_glGenFencesNV;
#define glGenFencesNV glad_glGenFencesNV
typedef GLboolean (APIENTRYP PFNGLISFENCENVPROC)(GLuint fence);
GLAPI PFNGLISFENCENVPROC glad_glIsFenceNV;
#define glIsFenceNV glad_glIsFenceNV
typedef GLboolean (APIENTRYP PFNGLTESTFENCENVPROC)(GLuint fence);
GLAPI PFNGLTESTFENCENVPROC glad_glTestFenceNV;
#define glTestFenceNV glad_glTestFenceNV
typedef void (APIENTRYP PFNGLGETFENCEIVNVPROC)(GLuint fence, GLenum pname, GLint *params);
GLAPI PFNGLGETFENCEIVNVPROC glad_glGetFenceivNV;
#define glGetFenceivNV glad_glGetFenceivNV
typedef void (APIENTRYP PFNGLFINISHFENCENVPROC)(GLuint fence);
GLAPI PFNGLFINISHFENCENVPROC glad_glFinishFenceNV;
#define glFinishFenceNV glad_glFinishFenceNV
typedef void (APIENTRYP PFNGLSETFENCENVPROC)(GLuint fence, GLenum condition);
GLAPI PFNGLSETFENCENVPROC glad_glSetFenceNV;
#define glSetFenceNV glad_glSetFenceNV
#endif
#ifndef GL_NV_fill_rectangle
#define GL_NV_fill_rectangle 1
GLAPI int GLAD_GL_NV_fill_rectangle;
#endif
#ifndef GL_NV_float_buffer
#define GL_NV_float_buffer 1
GLAPI int GLAD_GL_NV_float_buffer;
#endif
#ifndef GL_NV_fog_distance
#define GL_NV_fog_distance 1
GLAPI int GLAD_GL_NV_fog_distance;
#endif
#ifndef GL_NV_fragment_coverage_to_color
#define GL_NV_fragment_coverage_to_color 1
GLAPI int GLAD_GL_NV_fragment_coverage_to_color;
typedef void (APIENTRYP PFNGLFRAGMENTCOVERAGECOLORNVPROC)(GLuint color);
GLAPI PFNGLFRAGMENTCOVERAGECOLORNVPROC glad_glFragmentCoverageColorNV;
#define glFragmentCoverageColorNV glad_glFragmentCoverageColorNV
#endif
#ifndef GL_NV_fragment_program
#define GL_NV_fragment_program 1
GLAPI int GLAD_GL_NV_fragment_program;
typedef void (APIENTRYP PFNGLPROGRAMNAMEDPARAMETER4FNVPROC)(GLuint id, GLsizei len, const GLubyte *name, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLPROGRAMNAMEDPARAMETER4FNVPROC glad_glProgramNamedParameter4fNV;
#define glProgramNamedParameter4fNV glad_glProgramNamedParameter4fNV
typedef void (APIENTRYP PFNGLPROGRAMNAMEDPARAMETER4FVNVPROC)(GLuint id, GLsizei len, const GLubyte *name, const GLfloat *v);
GLAPI PFNGLPROGRAMNAMEDPARAMETER4FVNVPROC glad_glProgramNamedParameter4fvNV;
#define glProgramNamedParameter4fvNV glad_glProgramNamedParameter4fvNV
typedef void (APIENTRYP PFNGLPROGRAMNAMEDPARAMETER4DNVPROC)(GLuint id, GLsizei len, const GLubyte *name, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLPROGRAMNAMEDPARAMETER4DNVPROC glad_glProgramNamedParameter4dNV;
#define glProgramNamedParameter4dNV glad_glProgramNamedParameter4dNV
typedef void (APIENTRYP PFNGLPROGRAMNAMEDPARAMETER4DVNVPROC)(GLuint id, GLsizei len, const GLubyte *name, const GLdouble *v);
GLAPI PFNGLPROGRAMNAMEDPARAMETER4DVNVPROC glad_glProgramNamedParameter4dvNV;
#define glProgramNamedParameter4dvNV glad_glProgramNamedParameter4dvNV
typedef void (APIENTRYP PFNGLGETPROGRAMNAMEDPARAMETERFVNVPROC)(GLuint id, GLsizei len, const GLubyte *name, GLfloat *params);
GLAPI PFNGLGETPROGRAMNAMEDPARAMETERFVNVPROC glad_glGetProgramNamedParameterfvNV;
#define glGetProgramNamedParameterfvNV glad_glGetProgramNamedParameterfvNV
typedef void (APIENTRYP PFNGLGETPROGRAMNAMEDPARAMETERDVNVPROC)(GLuint id, GLsizei len, const GLubyte *name, GLdouble *params);
GLAPI PFNGLGETPROGRAMNAMEDPARAMETERDVNVPROC glad_glGetProgramNamedParameterdvNV;
#define glGetProgramNamedParameterdvNV glad_glGetProgramNamedParameterdvNV
#endif
```

```
#ifndef GL_NV_fragment_program2
#define GL_NV_fragment_program2 1
GLAPI int GLAD_GL_NV_fragment_program2;
#endif
#ifndef GL_NV_fragment_program4
#define GL_NV_fragment_program4 1
GLAPI int GLAD_GL_NV_fragment_program4;
#endif
#ifndef GL_NV_fragment_program_option
#define GL_NV_fragment_program_option 1
GLAPI int GLAD_GL_NV_fragment_program_option;
#endif
#ifndef GL_NV_fragment_shader_barycentric
#define GL_NV_fragment_shader_barycentric 1
GLAPI int GLAD_GL_NV_fragment_shader_barycentric;
#endif
#ifndef GL_NV_fragment_shader_interlock
#define GL_NV_fragment_shader_interlock 1
GLAPI int GLAD_GL_NV_fragment_shader_interlock;
#endif
#ifndef GL_NV_framebuffer_mixed_samples
#define GL_NV_framebuffer_mixed_samples 1
GLAPI int GLAD_GL_NV_framebuffer_mixed_samples;
typedef void (APIENTRYP PFNGLCOVERAGEMODULATIONTABLENVPROC)(GLsizei n, const GLfloat *v);
GLAPI PFNGLCOVERAGEMODULATIONTABLENVPROC glad_glCoverageModulationTableNV;
#define glCoverageModulationTableNV glad_glCoverageModulationTableNV
typedef void (APIENTRYP PFNGLGETCOVERAGEMODULATIONTABLENVPROC)(GLsizei bufSize, GLfloat *v);
GLAPI PFNGLGETCOVERAGEMODULATIONTABLENVPROC glad_glGetCoverageModulationTableNV;
#define glGetCoverageModulationTableNV glad_glGetCoverageModulationTableNV
typedef void (APIENTRYP PFNGLCOVERAGEMODULATIONNVPROC)(GLenum components);
GLAPI PFNGLCOVERAGEMODULATIONNVPROC glad_glCoverageModulationNV;
#define glCoverageModulationNV glad_glCoverageModulationNV
#endif
#ifndef GL_NV_framebuffer_multisample_coverage
#define GL_NV_framebuffer_multisample_coverage 1
GLAPI int GLAD_GL_NV_framebuffer_multisample_coverage;
typedef void (APIENTRYP PFNGLRENDERBUFFERSTORAGEMULTISAMPLECOVERAGENVPROC)(GLenum target, GLsizei coverageSamples, GLsizei colorSamples, GLenum internalformat, GLsizei width, GLsizei height);
GLAPI PFNGLRENDERBUFFERSTORAGEMULTISAMPLECOVERAGENVPROC glad_glRenderbufferStorageMultisampleCoverageNV;
#define glRenderbufferStorageMultisampleCoverageNV glad_glRenderbufferStorageMultisampleCoverageNV
#endif
#ifndef GL_NV_geometry_program4
#define GL_NV_geometry_program4 1
GLAPI int GLAD_GL_NV_geometry_program4;
typedef void (APIENTRYP PFNGLPROGRAMVERTEXLIMITNVPROC)(GLenum target, GLint limit);
GLAPI PFNGLPROGRAMVERTEXLIMITNVPROC glad_glProgramVertexLimitNV;
#define glProgramVertexLimitNV glad_glProgramVertexLimitNV
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTUREEXTPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level);
GLAPI PFNGLFRAMEBUFFERTEXTUREEXTPROC glad_glFramebufferTextureEXT;
#define glFramebufferTextureEXT glad_glFramebufferTextureEXT
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTUREFACEEXTPROC)(GLenum target, GLenum attachment, GLuint texture, GLint level, GLenum face);
GLAPI PFNGLFRAMEBUFFERTEXTUREFACEEXTPROC glad_glFramebufferTextureFaceEXT;
#define glFramebufferTextureFaceEXT glad_glFramebufferTextureFaceEXT
#endif
#ifndef GL_NV_geometry_shader4
#define GL_NV_geometry_shader4 1
GLAPI int GLAD_GL_NV_geometry_shader4;
#endif
#ifndef GL_NV_geometry_shader_passthrough
#define GL_NV_geometry_shader_passthrough 1
GLAPI int GLAD_GL_NV_geometry_shader_passthrough;
#endif
#ifndef GL_NV_gpu_multicast
#define GL_NV_gpu_multicast 1
GLAPI int GLAD_GL_NV_gpu_multicast;
typedef void (APIENTRYP PFNGLRENDERGPUMASKNVPROC)(GLbitfield mask);
GLAPI PFNGLRENDERGPUMASKNVPROC glad_glRenderGpuMaskNV;
#define glRenderGpuMaskNV glad_glRenderGpuMaskNV
typedef void (APIENTRYP PFNGLMULTICASTBUFFERSUBDATANVPROC)(GLbitfield gpuMask, GLuint buffer, GLintptr offset, GLsizeiptr size, const void *data);
GLAPI PFNGLMULTICASTBUFFERSUBDATANVPROC glad_glMulticastBufferSubDataNV;
#define glMulticastBufferSubDataNV glad_glMulticastBufferSubDataNV
typedef void (APIENTRYP PFNGLMULTICASTCOPYBUFFERSUBDATANVPROC)(GLuint readGpu, GLbitfield writeGpuMask, GLuint readBuffer, GLuint writeBuffer, GLintptr readOffset, GLintptr writeOffset, GLsizeiptr size);
GLAPI PFNGLMULTICASTCOPYBUFFERSUBDATANVPROC glad_glMulticastCopyBufferSubDataNV;
#define glMulticastCopyBufferSubDataNV glad_glMulticastCopyBufferSubDataNV
typedef void (APIENTRYP 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);
GLAPI PFNGLMULTICASTCOPYIMAGESUBDATANVPROC glad_glMulticastCopyImageSubDataNV;
#define glMulticastCopyImageSubDataNV glad_glMulticastCopyImageSubDataNV
typedef void (APIENTRYP 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);
GLAPI PFNGLMULTICASTBLITFRAMEBUFFERNVPROC glad_glMulticastBlitFramebufferNV;
#define glMulticastBlitFramebufferNV glad_glMulticastBlitFramebufferNV
typedef void (APIENTRYP PFNGLMULTICASTFRAMEBUFFERSAMPLELOCATIONSFVNVPROC)(GLuint gpu, GLuint framebuffer, GLuint start, GLsizei count, const GLfloat *v);
GLAPI PFNGLMULTICASTFRAMEBUFFERSAMPLELOCATIONSFVNVPROC glad_glMulticastFramebufferSampleLocationsfvNV;
#define glMulticastFramebufferSampleLocationsfvNV glad_glMulticastFramebufferSampleLocationsfvNV
typedef void (APIENTRYP PFNGLMULTICASTBARRIERNVPROC)(void);
GLAPI PFNGLMULTICASTBARRIERNVPROC glad_glMulticastBarrierNV;
#define glMulticastBarrierNV glad_glMulticastBarrierNV
typedef void (APIENTRYP PFNGLMULTICASTWAITSYNCNVPROC)(GLuint signalGpu, GLbitfield waitGpuMask);
GLAPI PFNGLMULTICASTWAITSYNCNVPROC glad_glMulticastWaitSyncNV;
#define glMulticastWaitSyncNV glad_glMulticastWaitSyncNV
typedef void (APIENTRYP PFNGLMULTICASTGETQUERYOBJECTIVNVPROC)(GLuint gpu, GLuint id, GLenum pname, GLint *params);
GLAPI PFNGLMULTICASTGETQUERYOBJECTIVNVPROC glad_glMulticastGetQueryObjectivNV;
#define glMulticastGetQueryObjectivNV glad_glMulticastGetQueryObjectivNV
typedef void (APIENTRYP PFNGLMULTICASTGETQUERYOBJECTUIVNVPROC)(GLuint gpu, GLuint id, GLenum pname, GLuint *params);
GLAPI PFNGLMULTICASTGETQUERYOBJECTUIVNVPROC glad_glMulticastGetQueryObjectuivNV;
#define glMulticastGetQueryObjectuivNV glad_glMulticastGetQueryObjectuivNV
typedef void (APIENTRYP PFNGLMULTICASTGETQUERYOBJECTI64VNVPROC)(GLuint gpu, GLuint id, GLenum pname, GLint64 *params);
GLAPI PFNGLMULTICASTGETQUERYOBJECTI64VNVPROC glad_glMulticastGetQueryObjecti64vNV;
#define glMulticastGetQueryObjecti64vNV glad_glMulticastGetQueryObjecti64vNV
typedef void (APIENTRYP PFNGLMULTICASTGETQUERYOBJECTUI64VNVPROC)(GLuint gpu, GLuint id, GLenum pname, GLuint64 *params);
GLAPI PFNGLMULTICASTGETQUERYOBJECTUI64VNVPROC glad_glMulticastGetQueryObjectui64vNV;
#define glMulticastGetQueryObjectui64vNV glad_glMulticastGetQueryObjectui64vNV
#endif
#ifndef GL_NV_gpu_program4
#define GL_NV_gpu_program4 1
GLAPI int GLAD_GL_NV_gpu_program4;
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETERI4INVPROC)(GLenum target, GLuint index, GLint x, GLint y, GLint z, GLint w);
GLAPI PFNGLPROGRAMLOCALPARAMETERI4INVPROC glad_glProgramLocalParameterI4iNV;
#define glProgramLocalParameterI4iNV glad_glProgramLocalParameterI4iNV
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETERI4IVNVPROC)(GLenum target, GLuint index, const GLint *params);
GLAPI PFNGLPROGRAMLOCALPARAMETERI4IVNVPROC glad_glProgramLocalParameterI4ivNV;
#define glProgramLocalParameterI4ivNV glad_glProgramLocalParameterI4ivNV
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETERSI4IVNVPROC)(GLenum target, GLuint index, GLsizei count, const GLint *params);
GLAPI PFNGLPROGRAMLOCALPARAMETERSI4IVNVPROC glad_glProgramLocalParametersI4ivNV;
#define glProgramLocalParametersI4ivNV glad_glProgramLocalParametersI4ivNV
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETERI4UINVPROC)(GLenum target, GLuint index, GLuint x, GLuint y, GLuint z, GLuint w);
GLAPI PFNGLPROGRAMLOCALPARAMETERI4UINVPROC glad_glProgramLocalParameterI4uiNV;
#define glProgramLocalParameterI4uiNV glad_glProgramLocalParameterI4uiNV
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETERI4UIVNVPROC)(GLenum target, GLuint index, const GLuint *params);
GLAPI PFNGLPROGRAMLOCALPARAMETERI4UIVNVPROC glad_glProgramLocalParameterI4uivNV;
#define glProgramLocalParameterI4uivNV glad_glProgramLocalParameterI4uivNV
typedef void (APIENTRYP PFNGLPROGRAMLOCALPARAMETERSI4UIVNVPROC)(GLenum target, GLuint index, GLsizei count, const GLuint *params);
GLAPI PFNGLPROGRAMLOCALPARAMETERSI4UIVNVPROC glad_glProgramLocalParametersI4uivNV;
#define glProgramLocalParametersI4uivNV glad_glProgramLocalParametersI4uivNV
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETERI4INVPROC)(GLenum target, GLuint index, GLint x, GLint y, GLint z, GLint w);
GLAPI PFNGLPROGRAMENVPARAMETERI4INVPROC glad_glProgramEnvParameterI4iNV;
#define glProgramEnvParameterI4iNV glad_glProgramEnvParameterI4iNV
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETERI4IVNVPROC)(GLenum target, GLuint index, const GLint *params);
GLAPI PFNGLPROGRAMENVPARAMETERI4IVNVPROC glad_glProgramEnvParameterI4ivNV;
#define glProgramEnvParameterI4ivNV glad_glProgramEnvParameterI4ivNV
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETERSI4IVNVPROC)(GLenum target, GLuint index, GLsizei count, const GLint *params);
GLAPI PFNGLPROGRAMENVPARAMETERSI4IVNVPROC glad_glProgramEnvParametersI4ivNV;
#define glProgramEnvParametersI4ivNV glad_glProgramEnvParametersI4ivNV
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETERI4UINVPROC)(GLenum target, GLuint index, GLuint x, GLuint y, GLuint z, GLuint w);
GLAPI PFNGLPROGRAMENVPARAMETERI4UINVPROC glad_glProgramEnvParameterI4uiNV;
#define glProgramEnvParameterI4uiNV glad_glProgramEnvParameterI4uiNV
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETERI4UIVNVPROC)(GLenum target, GLuint index, const GLuint *params);
GLAPI PFNGLPROGRAMENVPARAMETERI4UIVNVPROC glad_glProgramEnvParameterI4uivNV;
#define glProgramEnvParameterI4uivNV glad_glProgramEnvParameterI4uivNV
typedef void (APIENTRYP PFNGLPROGRAMENVPARAMETERSI4UIVNVPROC)(GLenum target, GLuint index, GLsizei count, const GLuint *params);
GLAPI PFNGLPROGRAMENVPARAMETERSI4UIVNVPROC glad_glProgramEnvParametersI4uivNV;
#define glProgramEnvParametersI4uivNV glad_glProgramEnvParametersI4uivNV
typedef void (APIENTRYP PFNGLGETPROGRAMLOCALPARAMETERIIVNVPROC)(GLenum target, GLuint index, GLint *params);
GLAPI PFNGLGETPROGRAMLOCALPARAMETERIIVNVPROC glad_glGetProgramLocalParameterIivNV;
#define glGetProgramLocalParameterIivNV glad_glGetProgramLocalParameterIivNV
typedef void (APIENTRYP PFNGLGETPROGRAMLOCALPARAMETERIUIVNVPROC)(GLenum target, GLuint index, GLuint
*params);
GLAPI PFNGLGETPROGRAMLOCALPARAMETERIUIVNVPROC glad_glGetProgramLocalParameterIuivNV;
#define glGetProgramLocalParameterIuivNV glad_glGetProgramLocalParameterIuivNV
typedef void (APIENTRYP PFNGLGETPROGRAMENVPARAMETERIIVNVPROC)(GLenum target, GLuint index, GLint *params);
GLAPI PFNGLGETPROGRAMENVPARAMETERIIVNVPROC glad_glGetProgramEnvParameterIivNV;
#define glGetProgramEnvParameterIivNV glad_glGetProgramEnvParameterIivNV
typedef void (APIENTRYP PFNGLGETPROGRAMENVPARAMETERIUIVNVPROC)(GLenum target, GLuint index, GLuint *params);
GLAPI PFNGLGETPROGRAMENVPARAMETERIUIVNVPROC glad_glGetProgramEnvParameterIuivNV;
#define glGetProgramEnvParameterIuivNV glad_glGetProgramEnvParameterIuivNV
#endif
#ifndef GL_NV_gpu_program5
#define GL_NV_gpu_program5 1
GLAPI int GLAD_GL_NV_gpu_program5;
typedef void (APIENTRYP PFNGLPROGRAMSUBROUTINEPARAMETERSUIVNVPROC)(GLenum target, GLsizei count, const GLuint *params);
GLAPI PFNGLPROGRAMSUBROUTINEPARAMETERSUIVNVPROC glad_glProgramSubroutineParametersuivNV;
#define glProgramSubroutineParametersuivNV glad_glProgramSubroutineParametersuivNV
typedef void (APIENTRYP PFNGLGETPROGRAMSUBROUTINEPARAMETERUIVNVPROC)(GLenum target, GLuint index, GLuint *param);
GLAPI PFNGLGETPROGRAMSUBROUTINEPARAMETERUIVNVPROC glad_glGetProgramSubroutineParameteruivNV;
#define glGetProgramSubroutineParameteruivNV glad_glGetProgramSubroutineParameteruivNV
#endif
#ifndef GL_NV_gpu_program5_mem_extended
#define GL_NV_gpu_program5_mem_extended 1
GLAPI int GLAD_GL_NV_gpu_program5_mem_extended;
#endif
#ifndef GL_NV_gpu_shader5
#define GL_NV_gpu_shader5 1
GLAPI int GLAD_GL_NV_gpu_shader5;
#endif
#ifndef GL_NV_half_float
#define GL_NV_half_float 1
GLAPI int GLAD_GL_NV_half_float;
typedef void (APIENTRYP PFNGLVERTEX2HNVPROC)(GLhalfNV x, GLhalfNV y);
GLAPI PFNGLVERTEX2HNVPROC glad_glVertex2hNV;
#define glVertex2hNV glad_glVertex2hNV
typedef void (APIENTRYP PFNGLVERTEX2HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLVERTEX2HVNVPROC glad_glVertex2hvNV;
#define glVertex2hvNV glad_glVertex2hvNV
typedef void (APIENTRYP PFNGLVERTEX3HNVPROC)(GLhalfNV x, GLhalfNV y, GLhalfNV z);
GLAPI PFNGLVERTEX3HNVPROC glad_glVertex3hNV;
#define glVertex3hNV glad_glVertex3hNV
typedef void (APIENTRYP PFNGLVERTEX3HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLVERTEX3HVNVPROC glad_glVertex3hvNV;
#define glVertex3hvNV glad_glVertex3hvNV
typedef void (APIENTRYP PFNGLVERTEX4HNVPROC)(GLhalfNV x, GLhalfNV y, GLhalfNV z, GLhalfNV w);
GLAPI PFNGLVERTEX4HNVPROC glad_glVertex4hNV;
#define glVertex4hNV glad_glVertex4hNV
typedef void (APIENTRYP PFNGLVERTEX4HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLVERTEX4HVNVPROC glad_glVertex4hvNV;
#define glVertex4hvNV glad_glVertex4hvNV
typedef void (APIENTRYP PFNGLNORMAL3HNVPROC)(GLhalfNV nx, GLhalfNV ny, GLhalfNV nz);
GLAPI PFNGLNORMAL3HNVPROC glad_glNormal3hNV;
#define glNormal3hNV glad_glNormal3hNV
typedef void (APIENTRYP PFNGLNORMAL3HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLNORMAL3HVNVPROC glad_glNormal3hvNV;
#define glNormal3hvNV glad_glNormal3hvNV
typedef void (APIENTRYP PFNGLCOLOR3HNVPROC)(GLhalfNV red, GLhalfNV green, GLhalfNV blue);
GLAPI PFNGLCOLOR3HNVPROC glad_glColor3hNV;
#define glColor3hNV glad_glColor3hNV
typedef void (APIENTRYP PFNGLCOLOR3HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLCOLOR3HVNVPROC glad_glColor3hvNV;
#define glColor3hvNV glad_glColor3hvNV
typedef void (APIENTRYP PFNGLCOLOR4HNVPROC)(GLhalfNV red, GLhalfNV green, GLhalfNV blue, GLhalfNV alpha);
GLAPI PFNGLCOLOR4HNVPROC glad_glColor4hNV;
#define glColor4hNV glad_glColor4hNV
typedef void (APIENTRYP PFNGLCOLOR4HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLCOLOR4HVNVPROC glad_glColor4hvNV;
#define glColor4hvNV glad_glColor4hvNV
typedef void (APIENTRYP PFNGLTEXCOORD1HNVPROC)(GLhalfNV s);
GLAPI PFNGLTEXCOORD1HNVPROC glad_glTexCoord1hNV;
#define glTexCoord1hNV glad_glTexCoord1hNV
typedef void (APIENTRYP PFNGLTEXCOORD1HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLTEXCOORD1HVNVPROC glad_glTexCoord1hvNV;
#define glTexCoord1hvNV glad_glTexCoord1hvNV
typedef void (APIENTRYP PFNGLTEXCOORD2HNVPROC)(GLhalfNV s, GLhalfNV t);
GLAPI PFNGLTEXCOORD2HNVPROC glad_glTexCoord2hNV;
#define glTexCoord2hNV glad_glTexCoord2hNV
typedef void (APIENTRYP PFNGLTEXCOORD2HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLTEXCOORD2HVNVPROC glad_glTexCoord2hvNV;
#define glTexCoord2hvNV glad_glTexCoord2hvNV
typedef void (APIENTRYP PFNGLTEXCOORD3HNVPROC)(GLhalfNV s, GLhalfNV t, GLhalfNV r);
GLAPI PFNGLTEXCOORD3HNVPROC glad_glTexCoord3hNV;
#define glTexCoord3hNV glad_glTexCoord3hNV
```

```
typedef void (APIENTRYP PFNGLTEXCOORD3HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLTEXCOORD3HVNVPROC glad_glTexCoord3hvNV;
#define glTexCoord3hvNV glad_glTexCoord3hvNV
typedef void (APIENTRYP PFNGLTEXCOORD4HNVPROC)(GLhalfNV s, GLhalfNV t, GLhalfNV r, GLhalfNV q);
GLAPI PFNGLTEXCOORD4HNVPROC glad_glTexCoord4hNV;
#define glTexCoord4hNV glad_glTexCoord4hNV
typedef void (APIENTRYP PFNGLTEXCOORD4HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLTEXCOORD4HVNVPROC glad_glTexCoord4hvNV;
#define glTexCoord4hvNV glad_glTexCoord4hvNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD1HNVPROC)(GLenum target, GLhalfNV s);
GLAPI PFNGLMULTITEXCOORD1HNVPROC glad_glMultiTexCoord1hNV;
#define glMultiTexCoord1hNV glad_glMultiTexCoord1hNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD1HVNVPROC)(GLenum target, const GLhalfNV *v);
GLAPI PFNGLMULTITEXCOORD1HVNVPROC glad_glMultiTexCoord1hvNV;
#define glMultiTexCoord1hvNV glad_glMultiTexCoord1hvNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD2HNVPROC)(GLenum target, GLhalfNV s, GLhalfNV t);
GLAPI PFNGLMULTITEXCOORD2HNVPROC glad_glMultiTexCoord2hNV;
#define glMultiTexCoord2hNV glad_glMultiTexCoord2hNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD2HVNVPROC)(GLenum target, const GLhalfNV *v);
GLAPI PFNGLMULTITEXCOORD2HVNVPROC glad_glMultiTexCoord2hvNV;
#define glMultiTexCoord2hvNV glad_glMultiTexCoord2hvNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD3HNVPROC)(GLenum target, GLhalfNV s, GLhalfNV t, GLhalfNV r);
GLAPI PFNGLMULTITEXCOORD3HNVPROC glad_glMultiTexCoord3hNV;
#define glMultiTexCoord3hNV glad_glMultiTexCoord3hNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD3HVNVPROC)(GLenum target, const GLhalfNV *v);
GLAPI PFNGLMULTITEXCOORD3HVNVPROC glad_glMultiTexCoord3hvNV;
#define glMultiTexCoord3hvNV glad_glMultiTexCoord3hvNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD4HNVPROC)(GLenum target, GLhalfNV s, GLhalfNV t, GLhalfNV r, GLhalfNV q);
GLAPI PFNGLMULTITEXCOORD4HNVPROC glad_glMultiTexCoord4hNV;
#define glMultiTexCoord4hNV glad_glMultiTexCoord4hNV
typedef void (APIENTRYP PFNGLMULTITEXCOORD4HVNVPROC)(GLenum target, const GLhalfNV *v);
GLAPI PFNGLMULTITEXCOORD4HVNVPROC glad_glMultiTexCoord4hvNV;
#define glMultiTexCoord4hvNV glad_glMultiTexCoord4hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1HNVPROC)(GLuint index, GLhalfNV x);
GLAPI PFNGLVERTEXATTRIB1HNVPROC glad_glVertexAttrib1hNV;
#define glVertexAttrib1hNV glad_glVertexAttrib1hNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1HVNVPROC)(GLuint index, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIB1HVNVPROC glad_glVertexAttrib1hvNV;
#define glVertexAttrib1hvNV glad_glVertexAttrib1hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2HNVPROC)(GLuint index, GLhalfNV x, GLhalfNV y);
GLAPI PFNGLVERTEXATTRIB2HNVPROC glad_glVertexAttrib2hNV;
#define glVertexAttrib2hNV glad_glVertexAttrib2hNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2HVNVPROC)(GLuint index, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIB2HVNVPROC glad_glVertexAttrib2hvNV;
#define glVertexAttrib2hvNV glad_glVertexAttrib2hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3HNVPROC)(GLuint index, GLhalfNV x, GLhalfNV y, GLhalfNV z);
GLAPI PFNGLVERTEXATTRIB3HNVPROC glad_glVertexAttrib3hNV;
#define glVertexAttrib3hNV glad_glVertexAttrib3hNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3HVNVPROC)(GLuint index, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIB3HVNVPROC glad_glVertexAttrib3hvNV;
#define glVertexAttrib3hvNV glad_glVertexAttrib3hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4HNVPROC)(GLuint index, GLhalfNV x, GLhalfNV y, GLhalfNV z, GLhalfNV w);
GLAPI PFNGLVERTEXATTRIB4HNVPROC glad_glVertexAttrib4hNV;
#define glVertexAttrib4hNV glad_glVertexAttrib4hNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4HVNVPROC)(GLuint index, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIB4HVNVPROC glad_glVertexAttrib4hvNV;
#define glVertexAttrib4hvNV glad_glVertexAttrib4hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS1HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIBS1HVNVPROC glad_glVertexAttribs1hvNV;
#define glVertexAttribs1hvNV glad_glVertexAttribs1hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS2HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIBS2HVNVPROC glad_glVertexAttribs2hvNV;
#define glVertexAttribs2hvNV glad_glVertexAttribs2hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS3HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIBS3HVNVPROC glad_glVertexAttribs3hvNV;
#define glVertexAttribs3hvNV glad_glVertexAttribs3hvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS4HVNVPROC)(GLuint index, GLsizei n, const GLhalfNV *v);
GLAPI PFNGLVERTEXATTRIBS4HVNVPROC glad_glVertexAttribs4hvNV;
#define glVertexAttribs4hvNV glad_glVertexAttribs4hvNV
typedef void (APIENTRYP PFNGLFOGCOORDHNVPROC)(GLhalfNV fog);
GLAPI PFNGLFOGCOORDHNVPROC glad_glFogCoordhNV;
#define glFogCoordhNV glad_glFogCoordhNV
typedef void (APIENTRYP PFNGLFOGCOORDHVNVPROC)(const GLhalfNV *fog);
GLAPI PFNGLFOGCOORDHVNVPROC glad_glFogCoordhvNV;
#define glFogCoordhvNV glad_glFogCoordhvNV
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3HNVPROC)(GLhalfNV red, GLhalfNV green, GLhalfNV blue);
GLAPI PFNGLSECONDARYCOLOR3HNVPROC glad_glSecondaryColor3hNV;
#define glSecondaryColor3hNV glad_glSecondaryColor3hNV
typedef void (APIENTRYP PFNGLSECONDARYCOLOR3HVNVPROC)(const GLhalfNV *v);
GLAPI PFNGLSECONDARYCOLOR3HVNVPROC glad_glSecondaryColor3hvNV;
#define glSecondaryColor3hvNV glad_glSecondaryColor3hvNV
typedef void (APIENTRYP PFNGLVERTEXWEIGHTHNVPROC)(GLhalfNV weight);
GLAPI PFNGLVERTEXWEIGHTHNVPROC glad_glVertexWeighthNV;
#define glVertexWeighthNV glad_glVertexWeighthNV
```

```
typedef void (APIENTRYP PFNGLVERTEXWEIGHTHVNVPROC)(const GLhalfNV *weight);
GLAPI PFNGLVERTEXWEIGHTHVNVPROC glad_glVertexWeighthvNV;
#define glVertexWeighthvNV glad_glVertexWeighthvNV
#endif
#ifndef GL_NV_internalformat_sample_query
#define GL_NV_internalformat_sample_query 1
GLAPI int GLAD_GL_NV_internalformat_sample_query;
typedef void (APIENTRYP PFNGLGETINTERNALFORMATSAMPLEIVNVPROC)(GLenum target, GLenum internalformat, GLsizei samples, GLenum pname, GLsizei count, GLint *params);
GLAPI PFNGLGETINTERNALFORMATSAMPLEIVNVPROC glad_glGetInternalformatSampleivNV;
#define glGetInternalformatSampleivNV glad_glGetInternalformatSampleivNV
#endif
#ifndef GL_NV_light_max_exponent
#define GL_NV_light_max_exponent 1
GLAPI int GLAD_GL_NV_light_max_exponent;
#endif
#ifndef GL_NV_memory_attachment
#define GL_NV_memory_attachment 1
GLAPI int GLAD_GL_NV_memory_attachment;
typedef void (APIENTRYP PFNGLGETMEMORYOBJECTDETACHEDRESOURCESUIVNVPROC)(GLuint memory, GLenum pname, GLint first, GLsizei count, GLuint *params);
GLAPI PFNGLGETMEMORYOBJECTDETACHEDRESOURCESUIVNVPROC glad_glGetMemoryObjectDetachedResourcesuivNV;
#define glGetMemoryObjectDetachedResourcesuivNV glad_glGetMemoryObjectDetachedResourcesuivNV
typedef void (APIENTRYP PFNGLRESETMEMORYOBJECTPARAMETERNVPROC)(GLuint memory, GLenum pname);
GLAPI PFNGLRESETMEMORYOBJECTPARAMETERNVPROC glad_glResetMemoryObjectParameterNV;
#define glResetMemoryObjectParameterNV glad_glResetMemoryObjectParameterNV
typedef void (APIENTRYP PFNGLTEXATTACHMEMORYNVPROC)(GLenum target, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXATTACHMEMORYNVPROC glad_glTexAttachMemoryNV;
#define glTexAttachMemoryNV glad_glTexAttachMemoryNV
typedef void (APIENTRYP PFNGLBUFFERATTACHMEMORYNVPROC)(GLenum target, GLuint memory, GLuint64 offset);
GLAPI PFNGLBUFFERATTACHMEMORYNVPROC glad_glBufferAttachMemoryNV;
#define glBufferAttachMemoryNV glad_glBufferAttachMemoryNV
typedef void (APIENTRYP PFNGLTEXTUREATTACHMEMORYNVPROC)(GLuint texture, GLuint memory, GLuint64 offset);
GLAPI PFNGLTEXTUREATTACHMEMORYNVPROC glad_glTextureAttachMemoryNV;
#define glTextureAttachMemoryNV glad_glTextureAttachMemoryNV
typedef void (APIENTRYP PFNGLNAMEDBUFFERATTACHMEMORYNVPROC)(GLuint buffer, GLuint memory, GLuint64 offset);
GLAPI PFNGLNAMEDBUFFERATTACHMEMORYNVPROC glad_glNamedBufferAttachMemoryNV;
#define glNamedBufferAttachMemoryNV glad_glNamedBufferAttachMemoryNV
#endif
#ifndef GL_NV_memory_object_sparse
#define GL_NV_memory_object_sparse 1
GLAPI int GLAD_GL_NV_memory_object_sparse;
typedef void (APIENTRYP PFNGLBUFFERPAGECOMMITMENTMEMNVPROC)(GLenum target, GLintptr offset, GLsizeiptr size, GLuint memory, GLuint64 memOffset, GLboolean commit);
GLAPI PFNGLBUFFERPAGECOMMITMENTMEMNVPROC glad_glBufferPageCommitmentMemNV;
#define glBufferPageCommitmentMemNV glad_glBufferPageCommitmentMemNV
typedef void (APIENTRYP PFNGLTEXPAGECOMMITMENTMEMNVPROC)(GLenum target, GLint layer, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLuint memory, GLuint64 offset, GLboolean commit);
GLAPI PFNGLTEXPAGECOMMITMENTMEMNVPROC glad_glTexPageCommitmentMemNV;
#define glTexPageCommitmentMemNV glad_glTexPageCommitmentMemNV
typedef void (APIENTRYP PFNGLNAMEDBUFFERPAGECOMMITMENTMEMNVPROC)(GLuint buffer, GLintptr offset, GLsizeiptr size, GLuint memory, GLuint64 memOffset, GLboolean commit);
GLAPI PFNGLNAMEDBUFFERPAGECOMMITMENTMEMNVPROC glad_glNamedBufferPageCommitmentMemNV;
#define glNamedBufferPageCommitmentMemNV glad_glNamedBufferPageCommitmentMemNV
typedef void (APIENTRYP PFNGLTEXTUREPAGECOMMITMENTMEMNVPROC)(GLuint texture, GLint layer, GLint level, GLint xoffset, GLint yoffset, GLint zoffset, GLsizei width, GLsizei height, GLsizei depth, GLuint memory, GLuint64 offset, GLboolean commit);
GLAPI PFNGLTEXTUREPAGECOMMITMENTMEMNVPROC glad_glTexturePageCommitmentMemNV;
#define glTexturePageCommitmentMemNV glad_glTexturePageCommitmentMemNV
#endif
#ifndef GL_NV_mesh_shader
#define GL_NV_mesh_shader 1
GLAPI int GLAD_GL_NV_mesh_shader;
typedef void (APIENTRYP PFNGLDRAWMESHTASKSNVPROC)(GLuint first, GLuint count);
GLAPI PFNGLDRAWMESHTASKSNVPROC glad_glDrawMeshTasksNV;
#define glDrawMeshTasksNV glad_glDrawMeshTasksNV
typedef void (APIENTRYP PFNGLDRAWMESHTASKSINDIRECTNVPROC)(GLintptr indirect);
GLAPI PFNGLDRAWMESHTASKSINDIRECTNVPROC glad_glDrawMeshTasksIndirectNV;
#define glDrawMeshTasksIndirectNV glad_glDrawMeshTasksIndirectNV
typedef void (APIENTRYP PFNGLMULTIDRAWMESHTASKSINDIRECTNVPROC)(GLintptr indirect, GLsizei drawcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWMESHTASKSINDIRECTNVPROC glad_glMultiDrawMeshTasksIndirectNV;
#define glMultiDrawMeshTasksIndirectNV glad_glMultiDrawMeshTasksIndirectNV
typedef void (APIENTRYP PFNGLMULTIDRAWMESHTASKSINDIRECTCOUNTNVPROC)(GLintptr indirect, GLintptr drawcount, GLsizei maxdrawcount, GLsizei stride);
GLAPI PFNGLMULTIDRAWMESHTASKSINDIRECTCOUNTNVPROC glad_glMultiDrawMeshTasksIndirectCountNV;
#define glMultiDrawMeshTasksIndirectCountNV glad_glMultiDrawMeshTasksIndirectCountNV
#endif
#ifndef GL_NV_multisample_coverage
#define GL_NV_multisample_coverage 1
GLAPI int GLAD_GL_NV_multisample_coverage;
#endif
#ifndef GL_NV_multisample_filter_hint
#define GL_NV_multisample_filter_hint 1
```

```
GLAPI int GLAD_GL_NV_multisample_filter_hint;
#endif
#ifndef GL_NV_occlusion_query
#define GL_NV_occlusion_query 1
GLAPI int GLAD_GL_NV_occlusion_query;
typedef void (APIENTRYP PFNGLGENOCCLUSIONQUERIESNVPROC)(GLsizei n, GLuint *ids);
GLAPI PFNGLGENOCCLUSIONQUERIESNVPROC glad_glGenOcclusionQueriesNV;
#define glGenOcclusionQueriesNV glad_glGenOcclusionQueriesNV
typedef void (APIENTRYP PFNGLDELETEOCCLUSIONQUERIESNVPROC)(GLsizei n, const GLuint *ids);
GLAPI PFNGLDELETEOCCLUSIONQUERIESNVPROC glad_glDeleteOcclusionQueriesNV;
#define glDeleteOcclusionQueriesNV glad_glDeleteOcclusionQueriesNV
typedef GLboolean (APIENTRYP PFNGLISOCCLUSIONQUERYNVPROC)(GLuint id);
GLAPI PFNGLISOCCLUSIONQUERYNVPROC glad_glIsOcclusionQueryNV;
#define glIsOcclusionQueryNV glad_glIsOcclusionQueryNV
typedef void (APIENTRYP PFNGLBEGINOCCLUSIONQUERYNVPROC)(GLuint id);
GLAPI PFNGLBEGINOCCLUSIONQUERYNVPROC glad_glBeginOcclusionQueryNV;
#define glBeginOcclusionQueryNV glad_glBeginOcclusionQueryNV
typedef void (APIENTRYP PFNGLENDOCCLUSIONQUERYNVPROC)(void);
GLAPI PFNGLENDOCCLUSIONQUERYNVPROC glad_glEndOcclusionQueryNV;
#define glEndOcclusionQueryNV glad_glEndOcclusionQueryNV
typedef void (APIENTRYP PFNGLGETOCCLUSIONQUERYIVNVPROC)(GLuint id, GLenum pname, GLint *params);
GLAPI PFNGLGETOCCLUSIONQUERYIVNVPROC glad_glGetOcclusionQueryivNV;
#define glGetOcclusionQueryivNV glad_glGetOcclusionQueryivNV
typedef void (APIENTRYP PFNGLGETOCCLUSIONQUERYUIVNVPROC)(GLuint id, GLenum pname, GLuint *params);
GLAPI PFNGLGETOCCLUSIONQUERYUIVNVPROC glad_glGetOcclusionQueryuivNV;
#define glGetOcclusionQueryuivNV glad_glGetOcclusionQueryuivNV
#endif
#ifndef GL_NV_packed_depth_stencil
#define GL_NV_packed_depth_stencil 1
GLAPI int GLAD_GL_NV_packed_depth_stencil;
#endif
#ifndef GL_NV_parameter_buffer_object
#define GL_NV_parameter_buffer_object 1
GLAPI int GLAD_GL_NV_parameter_buffer_object;
typedef void (APIENTRYP PFNGLPROGRAMBUFFERPARAMETERSFVNVPROC)(GLenum target, GLuint bindingIndex, GLuint wordIndex, GLsizei count, const GLfloat *params);
GLAPI PFNGLPROGRAMBUFFERPARAMETERSFVNVPROC glad_glProgramBufferParametersfvNV;
#define glProgramBufferParametersfvNV glad_glProgramBufferParametersfvNV
typedef void (APIENTRYP PFNGLPROGRAMBUFFERPARAMETERSIIVNVPROC)(GLenum target, GLuint bindingIndex, GLuint wordIndex, GLsizei count, const GLint *params);
GLAPI PFNGLPROGRAMBUFFERPARAMETERSIIVNVPROC glad_glProgramBufferParametersIivNV;
#define glProgramBufferParametersIivNV glad_glProgramBufferParametersIivNV
typedef void (APIENTRYP PFNGLPROGRAMBUFFERPARAMETERSIUIVNVPROC)(GLenum target, GLuint bindingIndex, GLuint wordIndex, GLsizei count, const GLuint *params);
GLAPI PFNGLPROGRAMBUFFERPARAMETERSIUIVNVPROC glad_glProgramBufferParametersIuivNV;
#define glProgramBufferParametersIuivNV glad_glProgramBufferParametersIuivNV
#endif
#ifndef GL_NV_parameter_buffer_object2
#define GL_NV_parameter_buffer_object2 1
GLAPI int GLAD_GL_NV_parameter_buffer_object2;
#endif
#ifndef GL_NV_path_rendering
#define GL_NV_path_rendering 1
GLAPI int GLAD_GL_NV_path_rendering;
typedef GLuint (APIENTRYP PFNGLGENPATHSNVPROC)(GLsizei range);
GLAPI PFNGLGENPATHSNVPROC glad_glGenPathsNV;
#define glGenPathsNV glad_glGenPathsNV
typedef void (APIENTRYP PFNGLDELETEPATHSNVPROC)(GLuint path, GLsizei range);
GLAPI PFNGLDELETEPATHSNVPROC glad_glDeletePathsNV;
#define glDeletePathsNV glad_glDeletePathsNV
typedef GLboolean (APIENTRYP PFNGLISPATHNVPROC)(GLuint path);
GLAPI PFNGLISPATHNVPROC glad_glIsPathNV;
#define glIsPathNV glad_glIsPathNV
typedef void (APIENTRYP PFNGLPATHCOMMANDSNVPROC)(GLuint path, GLsizei numCommands, const GLubyte *commands, GLsizei numCoords, GLenum coordType, const void *coords);
GLAPI PFNGLPATHCOMMANDSNVPROC glad_glPathCommandsNV;
#define glPathCommandsNV glad_glPathCommandsNV
typedef void (APIENTRYP PFNGLPATHCOORDSNVPROC)(GLuint path, GLsizei numCoords, GLenum coordType, const void *coords);
GLAPI PFNGLPATHCOORDSNVPROC glad_glPathCoordsNV;
#define glPathCoordsNV glad_glPathCoordsNV
typedef void (APIENTRYP PFNGLPATHSUBCOMMANDSNVPROC)(GLuint path, GLsizei commandStart, GLsizei commandsToDelete, GLsizei numCommands, const GLubyte *commands, GLsizei numCoords, GLenum coordType, const void *coords);
GLAPI PFNGLPATHSUBCOMMANDSNVPROC glad_glPathSubCommandsNV;
#define glPathSubCommandsNV glad_glPathSubCommandsNV
typedef void (APIENTRYP PFNGLPATHSUBCOORDSNVPROC)(GLuint path, GLsizei coordStart, GLsizei numCoords, GLenum coordType, const void *coords);
GLAPI PFNGLPATHSUBCOORDSNVPROC glad_glPathSubCoordsNV;
#define glPathSubCoordsNV glad_glPathSubCoordsNV
typedef void (APIENTRYP PFNGLPATHSTRINGNVPROC)(GLuint path, GLenum format, GLsizei length, const void *pathString);
GLAPI PFNGLPATHSTRINGNVPROC glad_glPathStringNV;
#define glPathStringNV glad_glPathStringNV
typedef void (APIENTRYP PFNGLPATHGLYPHSNVPROC)(GLuint firstPathName, GLenum fontTarget, const void *fontName, GLbitfield fontStyle, GLsizei numGlyphs, GLenum type, const void *charcodes, GLenum handleMissingGlyphs, GLuint pathParameterTemplate, GLfloat emScale);
```

```
GLAPI PFNGLPATHGLYPHSNVPROC glad_glPathGlyphsNV;
#define glPathGlyphsNV glad_glPathGlyphsNV
typedef void (APIENTRYP PFNGLPATHGLYPHRANGENVPROC)(GLuint firstPathName, GLenum fontTarget, const void *fontName, GLbitfield fontStyle, GLuint firstGlyph, GLsizei numGlyphs, GLenum handleMissingGlyphs, GLuint pathParameterTemplate, GLfloat emScale);
GLAPI PFNGLPATHGLYPHRANGENVPROC glad_glPathGlyphRangeNV;
#define glPathGlyphRangeNV glad_glPathGlyphRangeNV
typedef void (APIENTRYP PFNGLWEIGHTPATHSNVPROC)(GLuint resultPath, GLsizei numPaths, const GLuint *paths, const GLfloat *weights);
GLAPI PFNGLWEIGHTPATHSNVPROC glad_glWeightPathsNV;
#define glWeightPathsNV glad_glWeightPathsNV
typedef void (APIENTRYP PFNGLCOPYPATHNVPROC)(GLuint resultPath, GLuint srcPath);
GLAPI PFNGLCOPYPATHNVPROC glad_glCopyPathNV;
#define glCopyPathNV glad_glCopyPathNV
typedef void (APIENTRYP PFNGLINTERPOLATEPATHSNVPROC)(GLuint resultPath, GLuint pathA, GLuint pathB, GLfloat weight);
GLAPI PFNGLINTERPOLATEPATHSNVPROC glad_glInterpolatePathsNV;
#define glInterpolatePathsNV glad_glInterpolatePathsNV
typedef void (APIENTRYP PFNGLTRANSFORMPATHNVPROC)(GLuint resultPath, GLuint srcPath, GLenum transformType, const GLfloat *transformValues);
GLAPI PFNGLTRANSFORMPATHNVPROC glad_glTransformPathNV;
#define glTransformPathNV glad_glTransformPathNV
typedef void (APIENTRYP PFNGLPATHPARAMETERIVNVPROC)(GLuint path, GLenum pname, const GLint *value);
GLAPI PFNGLPATHPARAMETERIVNVPROC glad_glPathParameterivNV;
#define glPathParameterivNV glad_glPathParameterivNV
typedef void (APIENTRYP PFNGLPATHPARAMETERINVPROC)(GLuint path, GLenum pname, GLint value);
GLAPI PFNGLPATHPARAMETERINVPROC glad_glPathParameteriNV;
#define glPathParameteriNV glad_glPathParameteriNV
typedef void (APIENTRYP PFNGLPATHPARAMETERFVNVPROC)(GLuint path, GLenum pname, const GLfloat *value);
GLAPI PFNGLPATHPARAMETERFVNVPROC glad_glPathParameterfvNV;
#define glPathParameterfvNV glad_glPathParameterfvNV
typedef void (APIENTRYP PFNGLPATHPARAMETERFNVPROC)(GLuint path, GLenum pname, GLfloat value);
GLAPI PFNGLPATHPARAMETERFNVPROC glad_glPathParameterfNV;
#define glPathParameterfNV glad_glPathParameterfNV
typedef void (APIENTRYP PFNGLPATHDASHARRAYNVPROC)(GLuint path, GLsizei dashCount, const GLfloat *dashArray);
GLAPI PFNGLPATHDASHARRAYNVPROC glad_glPathDashArrayNV;
#define glPathDashArrayNV glad_glPathDashArrayNV
typedef void (APIENTRYP PFNGLPATHSTENCILFUNCNVPROC)(GLenum func, GLint ref, GLuint mask);
GLAPI PFNGLPATHSTENCILFUNCNVPROC glad_glPathStencilFuncNV;
#define glPathStencilFuncNV glad_glPathStencilFuncNV
typedef void (APIENTRYP PFNGLPATHSTENCILDEPTHOFFSETNVPROC)(GLfloat factor, GLfloat units);
GLAPI PFNGLPATHSTENCILDEPTHOFFSETNVPROC glad_glPathStencilDepthOffsetNV;
#define glPathStencilDepthOffsetNV glad_glPathStencilDepthOffsetNV
typedef void (APIENTRYP PFNGLSTENCILFILLPATHNVPROC)(GLuint path, GLenum fillMode, GLuint mask);
GLAPI PFNGLSTENCILFILLPATHNVPROC glad_glStencilFillPathNV;
#define glStencilFillPathNV glad_glStencilFillPathNV
typedef void (APIENTRYP PFNGLSTENCILSTROKEPATHNVPROC)(GLuint path, GLint reference, GLuint mask);
GLAPI PFNGLSTENCILSTROKEPATHNVPROC glad_glStencilStrokePathNV;
#define glStencilStrokePathNV glad_glStencilStrokePathNV
typedef void (APIENTRYP PFNGLSTENCILFILLPATHINSTANCEDNVPROC)(GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLenum fillMode, GLuint mask, GLenum transformType, const GLfloat *transformValues);
GLAPI PFNGLSTENCILFILLPATHINSTANCEDNVPROC glad_glStencilFillPathInstancedNV;
#define glStencilFillPathInstancedNV glad_glStencilFillPathInstancedNV
typedef void (APIENTRYP PFNGLSTENCILSTROKEPATHINSTANCEDNVPROC)(GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLint reference, GLuint mask, GLenum transformType, const GLfloat *transformValues);
GLAPI PFNGLSTENCILSTROKEPATHINSTANCEDNVPROC glad_glStencilStrokePathInstancedNV;
#define glStencilStrokePathInstancedNV glad_glStencilStrokePathInstancedNV
typedef void (APIENTRYP PFNGLPATHCOVERDEPTHFUNCNVPROC)(GLenum func);
GLAPI PFNGLPATHCOVERDEPTHFUNCNVPROC glad_glPathCoverDepthFuncNV;
#define glPathCoverDepthFuncNV glad_glPathCoverDepthFuncNV
typedef void (APIENTRYP PFNGLCOVERFILLPATHNVPROC)(GLuint path, GLenum coverMode);
GLAPI PFNGLCOVERFILLPATHNVPROC glad_glCoverFillPathNV;
#define glCoverFillPathNV glad_glCoverFillPathNV
typedef void (APIENTRYP PFNGLCOVERSTROKEPATHNVPROC)(GLuint path, GLenum coverMode);
GLAPI PFNGLCOVERSTROKEPATHNVPROC glad_glCoverStrokePathNV;
#define glCoverStrokePathNV glad_glCoverStrokePathNV
typedef void (APIENTRYP PFNGLCOVERFILLPATHINSTANCEDNVPROC)(GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLenum coverMode, GLenum transformType, const GLfloat *transformValues);
GLAPI PFNGLCOVERFILLPATHINSTANCEDNVPROC glad_glCoverFillPathInstancedNV;
#define glCoverFillPathInstancedNV glad_glCoverFillPathInstancedNV
typedef void (APIENTRYP PFNGLCOVERSTROKEPATHINSTANCEDNVPROC)(GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLenum coverMode, GLenum transformType, const GLfloat *transformValues);
GLAPI PFNGLCOVERSTROKEPATHINSTANCEDNVPROC glad_glCoverStrokePathInstancedNV;
#define glCoverStrokePathInstancedNV glad_glCoverStrokePathInstancedNV
typedef void (APIENTRYP PFNGLGETPATHPARAMETERIVNVPROC)(GLuint path, GLenum pname, GLint *value);
GLAPI PFNGLGETPATHPARAMETERIVNVPROC glad_glGetPathParameterivNV;
#define glGetPathParameterivNV glad_glGetPathParameterivNV
typedef void (APIENTRYP PFNGLGETPATHPARAMETERFVNVPROC)(GLuint path, GLenum pname, GLfloat *value);
GLAPI PFNGLGETPATHPARAMETERFVNVPROC glad_glGetPathParameterfvNV;
#define glGetPathParameterfvNV glad_glGetPathParameterfvNV
typedef void (APIENTRYP PFNGLGETPATHCOMMANDSNVPROC)(GLuint path, GLubyte *commands);
GLAPI PFNGLGETPATHCOMMANDSNVPROC glad_glGetPathCommandsNV;
```

```
#define glGetPathCommandsNV glad_glGetPathCommandsNV
typedef void (APIENTRYP PFNGLGETPATHCOORDSNVPROC)(GLuint path, GLfloat *coords);
GLAPI PFNGLGETPATHCOORDSNVPROC glad_glGetPathCoordsNV;
#define glGetPathCoordsNV glad_glGetPathCoordsNV
typedef void (APIENTRYP PFNGLGETPATHDASHARRAYNVPROC)(GLuint path, GLfloat *dashArray);
GLAPI PFNGLGETPATHDASHARRAYNVPROC glad_glGetPathDashArrayNV;
#define glGetPathDashArrayNV glad_glGetPathDashArrayNV
typedef void (APIENTRYP PFNGLGETPATHMETRICSNVPROC)(GLbitfield metricQueryMask, GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLsizei stride, GLfloat *metrics);
GLAPI PFNGLGETPATHMETRICSNVPROC glad_glGetPathMetricsNV;
#define glGetPathMetricsNV glad_glGetPathMetricsNV
typedef void (APIENTRYP PFNGLGETPATHMETRICRANGENVPROC)(GLbitfield metricQueryMask, GLuint firstPathName, GLsizei numPaths, GLsizei stride, GLfloat *metrics);
GLAPI PFNGLGETPATHMETRICRANGENVPROC glad_glGetPathMetricRangeNV;
#define glGetPathMetricRangeNV glad_glGetPathMetricRangeNV
typedef void (APIENTRYP PFNGLGETPATHSPACINGNVPROC)(GLenum pathListMode, GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLfloat advanceScale, GLfloat kerningScale, GLenum transformType, GLfloat *returnedSpacing);
GLAPI PFNGLGETPATHSPACINGNVPROC glad_glGetPathSpacingNV;
#define glGetPathSpacingNV glad_glGetPathSpacingNV
typedef GLboolean (APIENTRYP PFNGLISPOINTINFILLPATHNVPROC)(GLuint path, GLuint mask, GLfloat x, GLfloat y);
GLAPI PFNGLISPOINTINFILLPATHNVPROC glad_glIsPointInFillPathNV;
#define glIsPointInFillPathNV glad_glIsPointInFillPathNV
typedef GLboolean (APIENTRYP PFNGLISPOINTINSTROKEPATHNVPROC)(GLuint path, GLfloat x, GLfloat y);
GLAPI PFNGLISPOINTINSTROKEPATHNVPROC glad_glIsPointInStrokePathNV;
#define glIsPointInStrokePathNV glad_glIsPointInStrokePathNV
typedef GLfloat (APIENTRYP PFNGLGETPATHLENGTHNVPROC)(GLuint path, GLsizei startSegment, GLsizei numSegments);
GLAPI PFNGLGETPATHLENGTHNVPROC glad_glGetPathLengthNV;
#define glGetPathLengthNV glad_glGetPathLengthNV
typedef GLboolean (APIENTRYP PFNGLPOINTALONGPATHNVPROC)(GLuint path, GLsizei startSegment, GLsizei numSegments, GLfloat distance, GLfloat *x, GLfloat *y, GLfloat *tangentX, GLfloat *tangentY);
GLAPI PFNGLPOINTALONGPATHNVPROC glad_glPointAlongPathNV;
#define glPointAlongPathNV glad_glPointAlongPathNV
typedef void (APIENTRYP PFNGLMATRIXLOAD3X2FNVPROC)(GLenum matrixMode, const GLfloat *m);
GLAPI PFNGLMATRIXLOAD3X2FNVPROC glad_glMatrixLoad3x2fNV;
#define glMatrixLoad3x2fNV glad_glMatrixLoad3x2fNV
typedef void (APIENTRYP PFNGLMATRIXLOAD3X3FNVPROC)(GLenum matrixMode, const GLfloat *m);
GLAPI PFNGLMATRIXLOAD3X3FNVPROC glad_glMatrixLoad3x3fNV;
#define glMatrixLoad3x3fNV glad_glMatrixLoad3x3fNV
typedef void (APIENTRYP PFNGLMATRIXLOADTRANSPOSE3X3FNVPROC)(GLenum matrixMode, const GLfloat *m);
GLAPI PFNGLMATRIXLOADTRANSPOSE3X3FNVPROC glad_glMatrixLoadTranspose3x3fNV;
#define glMatrixLoadTranspose3x3fNV glad_glMatrixLoadTranspose3x3fNV
typedef void (APIENTRYP PFNGLMATRIXMULT3X2FNVPROC)(GLenum matrixMode, const GLfloat *m);
GLAPI PFNGLMATRIXMULT3X2FNVPROC glad_glMatrixMult3x2fNV;
#define glMatrixMult3x2fNV glad_glMatrixMult3x2fNV
typedef void (APIENTRYP PFNGLMATRIXMULT3X3FNVPROC)(GLenum matrixMode, const GLfloat *m);
GLAPI PFNGLMATRIXMULT3X3FNVPROC glad_glMatrixMult3x3fNV;
#define glMatrixMult3x3fNV glad_glMatrixMult3x3fNV
typedef void (APIENTRYP PFNGLMATRIXMULTTRANSPOSE3X3FNVPROC)(GLenum matrixMode, const GLfloat *m);
GLAPI PFNGLMATRIXMULTTRANSPOSE3X3FNVPROC glad_glMatrixMultTranspose3x3fNV;
#define glMatrixMultTranspose3x3fNV glad_glMatrixMultTranspose3x3fNV
typedef void (APIENTRYP PFNGLSTENCILTHENCOVERFILLPATHNVPROC)(GLuint path, GLenum fillMode, GLuint mask, GLenum coverMode);
GLAPI PFNGLSTENCILTHENCOVERFILLPATHNVPROC glad_glStencilThenCoverFillPathNV;
#define glStencilThenCoverFillPathNV glad_glStencilThenCoverFillPathNV
typedef void (APIENTRYP PFNGLSTENCILTHENCOVERSTROKEPATHNVPROC)(GLuint path, GLint reference, GLuint mask, GLenum coverMode);
GLAPI PFNGLSTENCILTHENCOVERSTROKEPATHNVPROC glad_glStencilThenCoverStrokePathNV;
#define glStencilThenCoverStrokePathNV glad_glStencilThenCoverStrokePathNV
typedef void (APIENTRYP PFNGLSTENCILTHENCOVERFILLPATHINSTANCEDNVPROC)(GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLenum fillMode, GLuint mask, GLenum coverMode, GLenum transformType, const GLfloat *transformValues);
GLAPI PFNGLSTENCILTHENCOVERFILLPATHINSTANCEDNVPROC glad_glStencilThenCoverFillPathInstancedNV;
#define glStencilThenCoverFillPathInstancedNV glad_glStencilThenCoverFillPathInstancedNV
typedef void (APIENTRYP PFNGLSTENCILTHENCOVERSTROKEPATHINSTANCEDNVPROC)(GLsizei numPaths, GLenum pathNameType, const void *paths, GLuint pathBase, GLint reference, GLuint mask, GLenum coverMode, GLenum transformType, const GLfloat *transformValues);
GLAPI PFNGLSTENCILTHENCOVERSTROKEPATHINSTANCEDNVPROC glad_glStencilThenCoverStrokePathInstancedNV;
#define glStencilThenCoverStrokePathInstancedNV glad_glStencilThenCoverStrokePathInstancedNV
typedef GLenum (APIENTRYP PFNGLPATHGLYPHINDEXRANGENVPROC)(GLenum fontTarget, const void *fontName, GLbitfield fontStyle, GLuint pathParameterTemplate, GLfloat emScale, GLuint *baseAndCount);
GLAPI PFNGLPATHGLYPHINDEXRANGENVPROC glad_glPathGlyphIndexRangeNV;
#define glPathGlyphIndexRangeNV glad_glPathGlyphIndexRangeNV
typedef GLenum (APIENTRYP PFNGLPATHGLYPHINDEXARRAYNVPROC)(GLuint firstPathName, GLenum fontTarget, const void *fontName, GLbitfield fontStyle, GLuint firstGlyphIndex, GLsizei numGlyphs, GLuint pathParameterTemplate, GLfloat emScale);
GLAPI PFNGLPATHGLYPHINDEXARRAYNVPROC glad_glPathGlyphIndexArrayNV;
#define glPathGlyphIndexArrayNV glad_glPathGlyphIndexArrayNV
typedef GLenum (APIENTRYP PFNGLPATHMEMORYGLYPHINDEXARRAYNVPROC)(GLuint firstPathName, GLenum fontTarget, GLsizeiptr fontSize, const void *fontData, GLsizei faceIndex, GLuint firstGlyphIndex, GLsizei numGlyphs, GLuint pathParameterTemplate, GLfloat emScale);
GLAPI PFNGLPATHMEMORYGLYPHINDEXARRAYNVPROC glad_glPathMemoryGlyphIndexArrayNV;
#define glPathMemoryGlyphIndexArrayNV glad_glPathMemoryGlyphIndexArrayNV
typedef void (APIENTRYP PFNGLPROGRAMPATHFRAGMENTINPUTGENNVPROC)(GLuint program, GLint location, GLenum genMode, GLint components, const GLfloat *coeffs);
```

```
GLAPI PFNGLPROGRAMPATHFRAGMENTINPUTGENNVPROC glad_glProgramPathFragmentInputGenNV;
#define glProgramPathFragmentInputGenNV glad_glProgramPathFragmentInputGenNV
typedef void (APIENTRYP PFNGLGETPROGRAMRESOURCEFVNVPROC)(GLuint program, GLenum programInterface, GLuint index, GLsizei propCount, const GLenum *props, GLsizei count, GLsizei *length, GLfloat *params);
GLAPI PFNGLGETPROGRAMRESOURCEFVNVPROC glad_glGetProgramResourcefvNV;
#define glGetProgramResourcefvNV glad_glGetProgramResourcefvNV
typedef void (APIENTRYP PFNGLPATHCOLORGENNVPROC)(GLenum color, GLenum genMode, GLenum colorFormat, const GLfloat *coeffs);
GLAPI PFNGLPATHCOLORGENNVPROC glad_glPathColorGenNV;
#define glPathColorGenNV glad_glPathColorGenNV
typedef void (APIENTRYP PFNGLPATHTEXGENNVPROC)(GLenum texCoordSet, GLenum genMode, GLint components, const GLfloat *coeffs);
GLAPI PFNGLPATHTEXGENNVPROC glad_glPathTexGenNV;
#define glPathTexGenNV glad_glPathTexGenNV
typedef void (APIENTRYP PFNGLPATHFOGGENNVPROC)(GLenum genMode);
GLAPI PFNGLPATHFOGGENNVPROC glad_glPathFogGenNV;
#define glPathFogGenNV glad_glPathFogGenNV
typedef void (APIENTRYP PFNGLGETPATHCOLORGENIVNVPROC)(GLenum color, GLenum pname, GLint *value);
GLAPI PFNGLGETPATHCOLORGENIVNVPROC glad_glGetPathColorGenivNV;
#define glGetPathColorGenivNV glad_glGetPathColorGenivNV
typedef void (APIENTRYP PFNGLGETPATHCOLORGENFVNVPROC)(GLenum color, GLenum pname, GLfloat *value);
GLAPI PFNGLGETPATHCOLORGENFVNVPROC glad_glGetPathColorGenfvNV;
#define glGetPathColorGenfvNV glad_glGetPathColorGenfvNV
typedef void (APIENTRYP PFNGLGETPATHTEXGENIVNVPROC)(GLenum texCoordSet, GLenum pname, GLint *value);
GLAPI PFNGLGETPATHTEXGENIVNVPROC glad_glGetPathTexGenivNV;
#define glGetPathTexGenivNV glad_glGetPathTexGenivNV
typedef void (APIENTRYP PFNGLGETPATHTEXGENFVNVPROC)(GLenum texCoordSet, GLenum pname, GLfloat *value);
GLAPI PFNGLGETPATHTEXGENFVNVPROC glad_glGetPathTexGenfvNV;
#define glGetPathTexGenfvNV glad_glGetPathTexGenfvNV
#endif
#ifndef GL_NV_path_rendering_shared_edge
#define GL_NV_path_rendering_shared_edge 1
GLAPI int GLAD_GL_NV_path_rendering_shared_edge;
#endif
#ifndef GL_NV_pixel_data_range
#define GL_NV_pixel_data_range 1
GLAPI int GLAD_GL_NV_pixel_data_range;
typedef void (APIENTRYP PFNGLPIXELDATARANGENVPROC)(GLenum target, GLsizei length, const void *pointer);
GLAPI PFNGLPIXELDATARANGENVPROC glad_glPixelDataRangeNV;
#define glPixelDataRangeNV glad_glPixelDataRangeNV
typedef void (APIENTRYP PFNGLFLUSHPIXELDATARANGENVPROC)(GLenum target);
GLAPI PFNGLFLUSHPIXELDATARANGENVPROC glad_glFlushPixelDataRangeNV;
#define glFlushPixelDataRangeNV glad_glFlushPixelDataRangeNV
#endif
#ifndef GL_NV_point_sprite
#define GL_NV_point_sprite 1
GLAPI int GLAD_GL_NV_point_sprite;
typedef void (APIENTRYP PFNGLPOINTPARAMETERINVPROC)(GLenum pname, GLint param);
GLAPI PFNGLPOINTPARAMETERINVPROC glad_glPointParameteriNV;
#define glPointParameteriNV glad_glPointParameteriNV
typedef void (APIENTRYP PFNGLPOINTPARAMETERIVNVPROC)(GLenum pname, const GLint *params);
GLAPI PFNGLPOINTPARAMETERIVNVPROC glad_glPointParameterivNV;
#define glPointParameterivNV glad_glPointParameterivNV
#endif
#ifndef GL_NV_present_video
#define GL_NV_present_video 1
GLAPI int GLAD_GL_NV_present_video;
typedef void (APIENTRYP PFNGLPRESENTFRAMEKEYEDNVPROC)(GLuint video_slot, GLuint64EXT minPresentTime, GLuint beginPresentTimeId, GLuint presentDurationId, GLenum type, GLenum target0, GLuint fill0, GLuint key0, GLenum target1, GLuint fill1, GLuint key1);
GLAPI PFNGLPRESENTFRAMEKEYEDNVPROC glad_glPresentFrameKeyedNV;
#define glPresentFrameKeyedNV glad_glPresentFrameKeyedNV
typedef void (APIENTRYP 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);
GLAPI PFNGLPRESENTFRAMEDUALFILLNVPROC glad_glPresentFrameDualFillNV;
#define glPresentFrameDualFillNV glad_glPresentFrameDualFillNV
typedef void (APIENTRYP PFNGLGETVIDEOIVNVPROC)(GLuint video_slot, GLenum pname, GLint *params);
GLAPI PFNGLGETVIDEOIVNVPROC glad_glGetVideoivNV;
#define glGetVideoivNV glad_glGetVideoivNV
typedef void (APIENTRYP PFNGLGETVIDEOUIVNVPROC)(GLuint video_slot, GLenum pname, GLuint *params);
GLAPI PFNGLGETVIDEOUIVNVPROC glad_glGetVideouivNV;
#define glGetVideouivNV glad_glGetVideouivNV
typedef void (APIENTRYP PFNGLGETVIDEOI64VNVPROC)(GLuint video_slot, GLenum pname, GLint64EXT *params);
GLAPI PFNGLGETVIDEOI64VNVPROC glad_glGetVideoi64vNV;
#define glGetVideoi64vNV glad_glGetVideoi64vNV
typedef void (APIENTRYP PFNGLGETVIDEOUI64VNVPROC)(GLuint video_slot, GLenum pname, GLuint64EXT *params);
GLAPI PFNGLGETVIDEOUI64VNVPROC glad_glGetVideoui64vNV;
#define glGetVideoui64vNV glad_glGetVideoui64vNV
#endif
#ifndef GL_NV_primitive_restart
#define GL_NV_primitive_restart 1
GLAPI int GLAD_GL_NV_primitive_restart;
```

```
typedef void (APIENTRYP PFNGLPRIMITIVERESTARTNVPROC)(void);
GLAPI PFNGLPRIMITIVERESTARTNVPROC glad_glPrimitiveRestartNV;
#define glPrimitiveRestartNV glad_glPrimitiveRestartNV
typedef void (APIENTRYP PFNGLPRIMITIVERESTARTINDEXNVPROC)(GLuint index);
GLAPI PFNGLPRIMITIVERESTARTINDEXNVPROC glad_glPrimitiveRestartIndexNV;
#define glPrimitiveRestartIndexNV glad_glPrimitiveRestartIndexNV
#endif
#ifndef GL_NV_primitive_shading_rate
#define GL_NV_primitive_shading_rate 1
GLAPI int GLAD_GL_NV_primitive_shading_rate;
#endif
#ifndef GL_NV_query_resource
#define GL_NV_query_resource 1
GLAPI int GLAD_GL_NV_query_resource;
typedef GLint (APIENTRYP PFNGLQUERYRESOURCENVPROC)(GLenum queryType, GLint tagId, GLuint count, GLint *buffer);
GLAPI PFNGLQUERYRESOURCENVPROC glad_glQueryResourceNV;
#define glQueryResourceNV glad_glQueryResourceNV
#endif
#ifndef GL_NV_query_resource_tag
#define GL_NV_query_resource_tag 1
GLAPI int GLAD_GL_NV_query_resource_tag;
typedef void (APIENTRYP PFNGLGENQUERYRESOURCETAGNVPROC)(GLsizei n, GLint *tagIds);
GLAPI PFNGLGENQUERYRESOURCETAGNVPROC glad_glGenQueryResourceTagNV;
#define glGenQueryResourceTagNV glad_glGenQueryResourceTagNV
typedef void (APIENTRYP PFNGLDELETEQUERYRESOURCETAGNVPROC)(GLsizei n, const GLint *tagIds);
GLAPI PFNGLDELETEQUERYRESOURCETAGNVPROC glad_glDeleteQueryResourceTagNV;
#define glDeleteQueryResourceTagNV glad_glDeleteQueryResourceTagNV
typedef void (APIENTRYP PFNGLQUERYRESOURCETAGNVPROC)(GLint tagId, const GLchar *tagString);
GLAPI PFNGLQUERYRESOURCETAGNVPROC glad_glQueryResourceTagNV;
#define glQueryResourceTagNV glad_glQueryResourceTagNV
#endif
#ifndef GL_NV_register_combiners
#define GL_NV_register_combiners 1
GLAPI int GLAD_GL_NV_register_combiners;
typedef void (APIENTRYP PFNGLCOMBINERPARAMETERFVNVPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLCOMBINERPARAMETERFVNVPROC glad_glCombinerParameterfvNV;
#define glCombinerParameterfvNV glad_glCombinerParameterfvNV
typedef void (APIENTRYP PFNGLCOMBINERPARAMETERFNVPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLCOMBINERPARAMETERFNVPROC glad_glCombinerParameterfNV;
#define glCombinerParameterfNV glad_glCombinerParameterfNV
typedef void (APIENTRYP PFNGLCOMBINERPARAMETERIVNVPROC)(GLenum pname, const GLint *params);
GLAPI PFNGLCOMBINERPARAMETERIVNVPROC glad_glCombinerParameterivNV;
#define glCombinerParameterivNV glad_glCombinerParameterivNV
typedef void (APIENTRYP PFNGLCOMBINERPARAMETERINVPROC)(GLenum pname, GLint param);
GLAPI PFNGLCOMBINERPARAMETERINVPROC glad_glCombinerParameteriNV;
#define glCombinerParameteriNV glad_glCombinerParameteriNV
typedef void (APIENTRYP PFNGLCOMBINERINPUTNVPROC)(GLenum stage, GLenum portion, GLenum variable, GLenum input, GLenum mapping, GLenum componentUsage);
GLAPI PFNGLCOMBINERINPUTNVPROC glad_glCombinerInputNV;
#define glCombinerInputNV glad_glCombinerInputNV
typedef void (APIENTRYP PFNGLCOMBINEROUTPUTNVPROC)(GLenum stage, GLenum portion, GLenum abOutput, GLenum cdOutput, GLenum sumOutput, GLenum scale, GLenum bias, GLboolean abDotProduct, GLboolean cdDotProduct, GLboolean muxSum);
GLAPI PFNGLCOMBINEROUTPUTNVPROC glad_glCombinerOutputNV;
#define glCombinerOutputNV glad_glCombinerOutputNV
typedef void (APIENTRYP PFNGLFINALCOMBINERINPUTNVPROC)(GLenum variable, GLenum input, GLenum mapping, GLenum componentUsage);
GLAPI PFNGLFINALCOMBINERINPUTNVPROC glad_glFinalCombinerInputNV;
#define glFinalCombinerInputNV glad_glFinalCombinerInputNV
typedef void (APIENTRYP PFNGLGETCOMBINERINPUTPARAMETERFVNVPROC)(GLenum stage, GLenum portion, GLenum variable, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCOMBINERINPUTPARAMETERFVNVPROC glad_glGetCombinerInputParameterfvNV;
#define glGetCombinerInputParameterfvNV glad_glGetCombinerInputParameterfvNV
typedef void (APIENTRYP PFNGLGETCOMBINERINPUTPARAMETERIVNVPROC)(GLenum stage, GLenum portion, GLenum variable, GLenum pname, GLint *params);
GLAPI PFNGLGETCOMBINERINPUTPARAMETERIVNVPROC glad_glGetCombinerInputParameterivNV;
#define glGetCombinerInputParameterivNV glad_glGetCombinerInputParameterivNV
typedef void (APIENTRYP PFNGLGETCOMBINEROUTPUTPARAMETERFVNVPROC)(GLenum stage, GLenum portion, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCOMBINEROUTPUTPARAMETERFVNVPROC glad_glGetCombinerOutputParameterfvNV;
#define glGetCombinerOutputParameterfvNV glad_glGetCombinerOutputParameterfvNV
typedef void (APIENTRYP PFNGLGETCOMBINEROUTPUTPARAMETERIVNVPROC)(GLenum stage, GLenum portion, GLenum pname, GLint *params);
GLAPI PFNGLGETCOMBINEROUTPUTPARAMETERIVNVPROC glad_glGetCombinerOutputParameterivNV;
#define glGetCombinerOutputParameterivNV glad_glGetCombinerOutputParameterivNV
typedef void (APIENTRYP PFNGLGETFINALCOMBINERINPUTPARAMETERFVNVPROC)(GLenum variable, GLenum pname, GLfloat *params);
GLAPI PFNGLGETFINALCOMBINERINPUTPARAMETERFVNVPROC glad_glGetFinalCombinerInputParameterfvNV;
#define glGetFinalCombinerInputParameterfvNV glad_glGetFinalCombinerInputParameterfvNV
typedef void (APIENTRYP PFNGLGETFINALCOMBINERINPUTPARAMETERIVNVPROC)(GLenum variable, GLenum pname, GLint *params);
GLAPI PFNGLGETFINALCOMBINERINPUTPARAMETERIVNVPROC glad_glGetFinalCombinerInputParameterivNV;
#define glGetFinalCombinerInputParameterivNV glad_glGetFinalCombinerInputParameterivNV
#endif
#ifndef GL_NV_register_combiners2
#define GL_NV_register_combiners2 1
```

```
GLAPI int GLAD_GL_NV_register_combiners2;
typedef void (APIENTRYP PFNGLCOMBINERSTAGEPARAMETERFVNVPROC)(GLenum stage, GLenum pname, const GLfloat *params);
GLAPI PFNGLCOMBINERSTAGEPARAMETERFVNVPROC glad_glCombinerStageParameterfvNV;
#define glCombinerStageParameterfvNV glad_glCombinerStageParameterfvNV
typedef void (APIENTRYP PFNGLGETCOMBINERSTAGEPARAMETERFVNVPROC)(GLenum stage, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCOMBINERSTAGEPARAMETERFVNVPROC glad_glGetCombinerStageParameterfvNV;
#define glGetCombinerStageParameterfvNV glad_glGetCombinerStageParameterfvNV
#endif
#ifndef GL_NV_representative_fragment_test
#define GL_NV_representative_fragment_test 1
GLAPI int GLAD_GL_NV_representative_fragment_test;
#endif
#ifndef GL_NV_robustness_video_memory_purge
#define GL_NV_robustness_video_memory_purge 1
GLAPI int GLAD_GL_NV_robustness_video_memory_purge;
#endif
#ifndef GL_NV_sample_locations
#define GL_NV_sample_locations 1
GLAPI int GLAD_GL_NV_sample_locations;
typedef void (APIENTRYP PFNGLFRAMEBUFFERSAMPLELOCATIONSFVNVPROC)(GLenum target, GLuint start, GLsizei count, const GLfloat *v);
GLAPI PFNGLFRAMEBUFFERSAMPLELOCATIONSFVNVPROC glad_glFramebufferSampleLocationsfvNV;
#define glFramebufferSampleLocationsfvNV glad_glFramebufferSampleLocationsfvNV
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVNVPROC)(GLuint framebuffer, GLuint start, GLsizei count, const GLfloat *v);
GLAPI PFNGLNAMEDFRAMEBUFFERSAMPLELOCATIONSFVNVPROC glad_glNamedFramebufferSampleLocationsfvNV;
#define glNamedFramebufferSampleLocationsfvNV glad_glNamedFramebufferSampleLocationsfvNV
typedef void (APIENTRYP PFNGLRESOLVEDEPTHVALUESNVPROC)(void);
GLAPI PFNGLRESOLVEDEPTHVALUESNVPROC glad_glResolveDepthValuesNV;
#define glResolveDepthValuesNV glad_glResolveDepthValuesNV
#endif
#ifndef GL_NV_sample_mask_override_coverage
#define GL_NV_sample_mask_override_coverage 1
GLAPI int GLAD_GL_NV_sample_mask_override_coverage;
#endif
#ifndef GL_NV_scissor_exclusive
#define GL_NV_scissor_exclusive 1
GLAPI int GLAD_GL_NV_scissor_exclusive;
typedef void (APIENTRYP PFNGLSCISSOREXCLUSIVENVPROC)(GLint x, GLint y, GLsizei width, GLsizei height);
GLAPI PFNGLSCISSOREXCLUSIVENVPROC glad_glScissorExclusiveNV;
#define glScissorExclusiveNV glad_glScissorExclusiveNV
typedef void (APIENTRYP PFNGLSCISSOREXCLUSIVEARRAYVNVPROC)(GLuint first, GLsizei count, const GLint *v);
GLAPI PFNGLSCISSOREXCLUSIVEARRAYVNVPROC glad_glScissorExclusiveArrayvNV;
#define glScissorExclusiveArrayvNV glad_glScissorExclusiveArrayvNV
#endif
#ifndef GL_NV_shader_atomic_counters
#define GL_NV_shader_atomic_counters 1
GLAPI int GLAD_GL_NV_shader_atomic_counters;
#endif
#ifndef GL_NV_shader_atomic_float
#define GL_NV_shader_atomic_float 1
GLAPI int GLAD_GL_NV_shader_atomic_float;
#endif
#ifndef GL_NV_shader_atomic_float64
#define GL_NV_shader_atomic_float64 1
GLAPI int GLAD_GL_NV_shader_atomic_float64;
#endif
#ifndef GL_NV_shader_atomic_fp16_vector
#define GL_NV_shader_atomic_fp16_vector 1
GLAPI int GLAD_GL_NV_shader_atomic_fp16_vector;
#endif
#ifndef GL_NV_shader_atomic_int64
#define GL_NV_shader_atomic_int64 1
GLAPI int GLAD_GL_NV_shader_atomic_int64;
#endif
#ifndef GL_NV_shader_buffer_load
#define GL_NV_shader_buffer_load 1
GLAPI int GLAD_GL_NV_shader_buffer_load;
typedef void (APIENTRYP PFNGLMAKEBUFFERRESIDENTNVPROC)(GLenum target, GLenum access);
GLAPI PFNGLMAKEBUFFERRESIDENTNVPROC glad_glMakeBufferResidentNV;
#define glMakeBufferResidentNV glad_glMakeBufferResidentNV
typedef void (APIENTRYP PFNGLMAKEBUFFERNONRESIDENTNVPROC)(GLenum target);
GLAPI PFNGLMAKEBUFFERNONRESIDENTNVPROC glad_glMakeBufferNonResidentNV;
#define glMakeBufferNonResidentNV glad_glMakeBufferNonResidentNV
typedef GLboolean (APIENTRYP PFNGLISBUFFERRESIDENTNVPROC)(GLenum target);
GLAPI PFNGLISBUFFERRESIDENTNVPROC glad_glIsBufferResidentNV;
#define glIsBufferResidentNV glad_glIsBufferResidentNV
typedef void (APIENTRYP PFNGLMAKENAMEDBUFFERRESIDENTNVPROC)(GLuint buffer, GLenum access);
GLAPI PFNGLMAKENAMEDBUFFERRESIDENTNVPROC glad_glMakeNamedBufferResidentNV;
#define glMakeNamedBufferResidentNV glad_glMakeNamedBufferResidentNV
typedef void (APIENTRYP PFNGLMAKENAMEDBUFFERNONRESIDENTNVPROC)(GLuint buffer);
GLAPI PFNGLMAKENAMEDBUFFERNONRESIDENTNVPROC glad_glMakeNamedBufferNonResidentNV;
#define glMakeNamedBufferNonResidentNV glad_glMakeNamedBufferNonResidentNV
typedef GLboolean (APIENTRYP PFNGLISNAMEDBUFFERRESIDENTNVPROC)(GLuint buffer);
```

```
GLAPI PFNGLISNAMEDBUFFERRESIDENTNVPROC glad_glIsNamedBufferResidentNV;
#define glIsNamedBufferResidentNV glad_glIsNamedBufferResidentNV
typedef void (APIENTRYP PFNGLGETBUFFERPARAMETERUI64VNVPROC)(GLenum target, GLenum pname, GLuint64EXT *params);
GLAPI PFNGLGETBUFFERPARAMETERUI64VNVPROC glad_glGetBufferParameterui64vNV;
#define glGetBufferParameterui64vNV glad_glGetBufferParameterui64vNV
typedef void (APIENTRYP PFNGLGETNAMEDBUFFERPARAMETERUI64VNVPROC)(GLuint buffer, GLenum pname, GLuint64EXT *params);
GLAPI PFNGLGETNAMEDBUFFERPARAMETERUI64VNVPROC glad_glGetNamedBufferParameterui64vNV;
#define glGetNamedBufferParameterui64vNV glad_glGetNamedBufferParameterui64vNV
typedef void (APIENTRYP PFNGLGETINTEGERUI64VNVPROC)(GLenum value, GLuint64EXT *result);
GLAPI PFNGLGETINTEGERUI64VNVPROC glad_glGetIntegerui64vNV;
#define glGetIntegerui64vNV glad_glGetIntegerui64vNV
typedef void (APIENTRYP PFNGLUNIFORMUI64NVPROC)(GLint location, GLuint64EXT value);
GLAPI PFNGLUNIFORMUI64NVPROC glad_glUniformui64NV;
#define glUniformui64NV glad_glUniformui64NV
typedef void (APIENTRYP PFNGLUNIFORMUI64VNVPROC)(GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLUNIFORMUI64VNVPROC glad_glUniformui64vNV;
#define glUniformui64vNV glad_glUniformui64vNV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMUI64NVPROC)(GLuint program, GLint location, GLuint64EXT value);
GLAPI PFNGLPROGRAMUNIFORMUI64NVPROC glad_glProgramUniformui64NV;
#define glProgramUniformui64NV glad_glProgramUniformui64NV
typedef void (APIENTRYP PFNGLPROGRAMUNIFORMUI64VNVPROC)(GLuint program, GLint location, GLsizei count, const GLuint64EXT *value);
GLAPI PFNGLPROGRAMUNIFORMUI64VNVPROC glad_glProgramUniformui64vNV;
#define glProgramUniformui64vNV glad_glProgramUniformui64vNV
#endif
#ifndef GL_NV_shader_buffer_store
#define GL_NV_shader_buffer_store 1
GLAPI int GLAD_GL_NV_shader_buffer_store;
#endif
#ifndef GL_NV_shader_storage_buffer_object
#define GL_NV_shader_storage_buffer_object 1
GLAPI int GLAD_GL_NV_shader_storage_buffer_object;
#endif
#ifndef GL_NV_shader_subgroup_partitioned
#define GL_NV_shader_subgroup_partitioned 1
GLAPI int GLAD_GL_NV_shader_subgroup_partitioned;
#endif
#ifndef GL_NV_shader_texture_footprint
#define GL_NV_shader_texture_footprint 1
GLAPI int GLAD_GL_NV_shader_texture_footprint;
#endif
#ifndef GL_NV_shader_thread_group
#define GL_NV_shader_thread_group 1
GLAPI int GLAD_GL_NV_shader_thread_group;
#endif
#ifndef GL_NV_shader_thread_shuffle
#define GL_NV_shader_thread_shuffle 1
GLAPI int GLAD_GL_NV_shader_thread_shuffle;
#endif
#ifndef GL_NV_shading_rate_image
#define GL_NV_shading_rate_image 1
GLAPI int GLAD_GL_NV_shading_rate_image;
typedef void (APIENTRYP PFNGLBINDSHADINGRATEIMAGENVPROC)(GLuint texture);
GLAPI PFNGLBINDSHADINGRATEIMAGENVPROC glad_glBindShadingRateImageNV;
#define glBindShadingRateImageNV glad_glBindShadingRateImageNV
typedef void (APIENTRYP PFNGLGETSHADINGRATEIMAGEPALETTENVPROC)(GLuint viewport, GLuint entry, GLenum *rate);
GLAPI PFNGLGETSHADINGRATEIMAGEPALETTENVPROC glad_glGetShadingRateImagePaletteNV;
#define glGetShadingRateImagePaletteNV glad_glGetShadingRateImagePaletteNV
typedef void (APIENTRYP PFNGLGETSHADINGRATESAMPLELOCATIONIVNVPROC)(GLenum rate, GLuint samples, GLuint index, GLint *location);
GLAPI PFNGLGETSHADINGRATESAMPLELOCATIONIVNVPROC glad_glGetShadingRateSampleLocationivNV;
#define glGetShadingRateSampleLocationivNV glad_glGetShadingRateSampleLocationivNV
typedef void (APIENTRYP PFNGLSHADINGRATEIMAGEBARRIERNVPROC)(GLboolean synchronize);
GLAPI PFNGLSHADINGRATEIMAGEBARRIERNVPROC glad_glShadingRateImageBarrierNV;
#define glShadingRateImageBarrierNV glad_glShadingRateImageBarrierNV
typedef void (APIENTRYP PFNGLSHADINGRATEIMAGEPALETTENVPROC)(GLuint viewport, GLuint first, GLsizei count, const GLenum *rates);
GLAPI PFNGLSHADINGRATEIMAGEPALETTENVPROC glad_glShadingRateImagePaletteNV;
#define glShadingRateImagePaletteNV glad_glShadingRateImagePaletteNV
typedef void (APIENTRYP PFNGLSHADINGRATESAMPLEORDERNVPROC)(GLenum order);
GLAPI PFNGLSHADINGRATESAMPLEORDERNVPROC glad_glShadingRateSampleOrderNV;
#define glShadingRateSampleOrderNV glad_glShadingRateSampleOrderNV
typedef void (APIENTRYP PFNGLSHADINGRATESAMPLEORDERCUSTOMNVPROC)(GLenum rate, GLuint samples, const GLint *locations);
GLAPI PFNGLSHADINGRATESAMPLEORDERCUSTOMNVPROC glad_glShadingRateSampleOrderCustomNV;
#define glShadingRateSampleOrderCustomNV glad_glShadingRateSampleOrderCustomNV
#endif
#ifndef GL_NV_stereo_view_rendering
#define GL_NV_stereo_view_rendering 1
GLAPI int GLAD_GL_NV_stereo_view_rendering;
#endif
#ifndef GL_NV_tessellation_program5
```

```
#define GL_NV_tessellation_program5 1
GLAPI int GLAD_GL_NV_tessellation_program5;
#endif
#ifndef GL_NV_texgen_emboss
#define GL_NV_texgen_emboss 1
GLAPI int GLAD_GL_NV_texgen_emboss;
#endif
#ifndef GL_NV_texgen_reflection
#define GL_NV_texgen_reflection 1
GLAPI int GLAD_GL_NV_texgen_reflection;
#endif
#ifndef GL_NV_texture_barrier
#define GL_NV_texture_barrier 1
GLAPI int GLAD_GL_NV_texture_barrier;
typedef void (APIENTRYP PFNGLTEXTUREBARRIERNVPROC)(void);
GLAPI PFNGLTEXTUREBARRIERNVPROC glad_glTextureBarrierNV;
#define glTextureBarrierNV glad_glTextureBarrierNV
#endif
#ifndef GL_NV_texture_compression_vtc
#define GL_NV_texture_compression_vtc 1
GLAPI int GLAD_GL_NV_texture_compression_vtc;
#endif
#ifndef GL_NV_texture_env_combine4
#define GL_NV_texture_env_combine4 1
GLAPI int GLAD_GL_NV_texture_env_combine4;
#endif
#ifndef GL_NV_texture_expand_normal
#define GL_NV_texture_expand_normal 1
GLAPI int GLAD_GL_NV_texture_expand_normal;
#endif
#ifndef GL_NV_texture_multisample
#define GL_NV_texture_multisample 1
GLAPI int GLAD_GL_NV_texture_multisample;
typedef void (APIENTRYP PFNGLTEXIMAGE2DMULTISAMPLECOVERAGENVPROC)(GLenum target, GLsizei coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height, GLboolean fixedSampleLocations);
GLAPI PFNGLTEXIMAGE2DMULTISAMPLECOVERAGENVPROC glad_glTexImage2DMultisampleCoverageNV;
#define glTexImage2DMultisampleCoverageNV glad_glTexImage2DMultisampleCoverageNV
typedef void (APIENTRYP PFNGLTEXIMAGE3DMULTISAMPLECOVERAGENVPROC)(GLenum target, GLsizei coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations);
GLAPI PFNGLTEXIMAGE3DMULTISAMPLECOVERAGENVPROC glad_glTexImage3DMultisampleCoverageNV;
#define glTexImage3DMultisampleCoverageNV glad_glTexImage3DMultisampleCoverageNV
typedef void (APIENTRYP PFNGLTEXTUREIMAGE2DMULTISAMPLENVPROC)(GLuint texture, GLenum target, GLsizei samples, GLint internalFormat, GLsizei width, GLsizei height, GLboolean fixedSampleLocations);
GLAPI PFNGLTEXTUREIMAGE2DMULTISAMPLENVPROC glad_glTextureImage2DMultisampleNV;
#define glTextureImage2DMultisampleNV glad_glTextureImage2DMultisampleNV
typedef void (APIENTRYP PFNGLTEXTUREIMAGE3DMULTISAMPLENVPROC)(GLuint texture, GLenum target, GLsizei samples, GLint internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations);
GLAPI PFNGLTEXTUREIMAGE3DMULTISAMPLENVPROC glad_glTextureImage3DMultisampleNV;
#define glTextureImage3DMultisampleNV glad_glTextureImage3DMultisampleNV
typedef void (APIENTRYP PFNGLTEXTUREIMAGE2DMULTISAMPLECOVERAGENVPROC)(GLuint texture, GLenum target, GLsizei coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height, GLboolean fixedSampleLocations);
GLAPI PFNGLTEXTUREIMAGE2DMULTISAMPLECOVERAGENVPROC glad_glTextureImage2DMultisampleCoverageNV;
#define glTextureImage2DMultisampleCoverageNV glad_glTextureImage2DMultisampleCoverageNV
typedef void (APIENTRYP PFNGLTEXTUREIMAGE3DMULTISAMPLECOVERAGENVPROC)(GLuint texture, GLenum target, GLsizei coverageSamples, GLsizei colorSamples, GLint internalFormat, GLsizei width, GLsizei height, GLsizei depth, GLboolean fixedSampleLocations);
GLAPI PFNGLTEXTUREIMAGE3DMULTISAMPLECOVERAGENVPROC glad_glTextureImage3DMultisampleCoverageNV;
#define glTextureImage3DMultisampleCoverageNV glad_glTextureImage3DMultisampleCoverageNV
#endif
#ifndef GL_NV_texture_rectangle
#define GL_NV_texture_rectangle 1
GLAPI int GLAD_GL_NV_texture_rectangle;
#endif
#ifndef GL_NV_texture_rectangle_compressed
#define GL_NV_texture_rectangle_compressed 1
GLAPI int GLAD_GL_NV_texture_rectangle_compressed;
#endif
#ifndef GL_NV_texture_shader
#define GL_NV_texture_shader 1
GLAPI int GLAD_GL_NV_texture_shader;
#endif
#ifndef GL_NV_texture_shader2
#define GL_NV_texture_shader2 1
GLAPI int GLAD_GL_NV_texture_shader2;
#endif
#ifndef GL_NV_texture_shader3
#define GL_NV_texture_shader3 1
GLAPI int GLAD_GL_NV_texture_shader3;
#endif
#ifndef GL_NV_timeline_semaphore
#define GL_NV_timeline_semaphore 1
GLAPI int GLAD_GL_NV_timeline_semaphore;
typedef void (APIENTRYP PFNGLCREATESEMAPHORESNVPROC)(GLsizei n, GLuint *semaphores);
```

```
GLAPI PFNGLCREATESEMAPHORESNVPROC glad_glCreateSemaphoresNV;
#define glCreateSemaphoresNV glad_glCreateSemaphoresNV
typedef void (APIENTRYP PFNGLSEMAPHOREPARAMETERIVNVPROC)(GLuint semaphore, GLenum pname, const GLint *params);
GLAPI PFNGLSEMAPHOREPARAMETERIVNVPROC glad_glSemaphoreParameterivNV;
#define glSemaphoreParameterivNV glad_glSemaphoreParameterivNV
typedef void (APIENTRYP PFNGLGETSEMAPHOREPARAMETERIVNVPROC)(GLuint semaphore, GLenum pname, GLint *params);
GLAPI PFNGLGETSEMAPHOREPARAMETERIVNVPROC glad_glGetSemaphoreParameterivNV;
#define glGetSemaphoreParameterivNV glad_glGetSemaphoreParameterivNV
#endif
#ifndef GL_NV_transform_feedback
#define GL_NV_transform_feedback 1
GLAPI int GLAD_GL_NV_transform_feedback;
typedef void (APIENTRYP PFNGLBEGINTRANSFORMFEEDBACKNVPROC)(GLenum primitiveMode);
GLAPI PFNGLBEGINTRANSFORMFEEDBACKNVPROC glad_glBeginTransformFeedbackNV;
#define glBeginTransformFeedbackNV glad_glBeginTransformFeedbackNV
typedef void (APIENTRYP PFNGLENDTRANSFORMFEEDBACKNVPROC)(void);
GLAPI PFNGLENDTRANSFORMFEEDBACKNVPROC glad_glEndTransformFeedbackNV;
#define glEndTransformFeedbackNV glad_glEndTransformFeedbackNV
typedef void (APIENTRYP PFNGLTRANSFORMFEEDBACKATTRIBSNVPROC)(GLsizei count, const GLint *attribs, GLenum bufferMode);
GLAPI PFNGLTRANSFORMFEEDBACKATTRIBSNVPROC glad_glTransformFeedbackAttribsNV;
#define glTransformFeedbackAttribsNV glad_glTransformFeedbackAttribsNV
typedef void (APIENTRYP PFNGLBINDBUFFERRANGENVPROC)(GLenum target, GLuint index, GLuint buffer, GLintptr offset, GLsizeiptr size);
GLAPI PFNGLBINDBUFFERRANGENVPROC glad_glBindBufferRangeNV;
#define glBindBufferRangeNV glad_glBindBufferRangeNV
typedef void (APIENTRYP PFNGLBINDBUFFEROFFSETNVPROC)(GLenum target, GLuint index, GLuint buffer, GLintptr offset);
GLAPI PFNGLBINDBUFFEROFFSETNVPROC glad_glBindBufferOffsetNV;
#define glBindBufferOffsetNV glad_glBindBufferOffsetNV
typedef void (APIENTRYP PFNGLBINDBUFFERBASENVPROC)(GLenum target, GLuint index, GLuint buffer);
GLAPI PFNGLBINDBUFFERBASENVPROC glad_glBindBufferBaseNV;
#define glBindBufferBaseNV glad_glBindBufferBaseNV
typedef void (APIENTRYP PFNGLTRANSFORMFEEDBACKVARYINGSNVPROC)(GLuint program, GLsizei count, const GLint *locations, GLenum bufferMode);
GLAPI PFNGLTRANSFORMFEEDBACKVARYINGSNVPROC glad_glTransformFeedbackVaryingsNV;
#define glTransformFeedbackVaryingsNV glad_glTransformFeedbackVaryingsNV
typedef void (APIENTRYP PFNGLACTIVEVARYINGNVPROC)(GLuint program, const GLchar *name);
GLAPI PFNGLACTIVEVARYINGNVPROC glad_glActiveVaryingNV;
#define glActiveVaryingNV glad_glActiveVaryingNV
typedef GLint (APIENTRYP PFNGLGETVARYINGLOCATIONNVPROC)(GLuint program, const GLchar *name);
GLAPI PFNGLGETVARYINGLOCATIONNVPROC glad_glGetVaryingLocationNV;
#define glGetVaryingLocationNV glad_glGetVaryingLocationNV
typedef void (APIENTRYP PFNGLGETACTIVEVARYINGNVPROC)(GLuint program, GLuint index, GLsizei bufSize, GLsizei *length, GLsizei *size, GLenum *type, GLchar *name);
GLAPI PFNGLGETACTIVEVARYINGNVPROC glad_glGetActiveVaryingNV;
#define glGetActiveVaryingNV glad_glGetActiveVaryingNV
typedef void (APIENTRYP PFNGLGETTRANSFORMFEEDBACKVARYINGNVPROC)(GLuint program, GLuint index, GLint *location);
GLAPI PFNGLGETTRANSFORMFEEDBACKVARYINGNVPROC glad_glGetTransformFeedbackVaryingNV;
#define glGetTransformFeedbackVaryingNV glad_glGetTransformFeedbackVaryingNV
typedef void (APIENTRYP PFNGLTRANSFORMFEEDBACKSTREAMATTRIBSNVPROC)(GLsizei count, const GLint *attribs, GLsizei nbuffers, const GLint *bufstreams, GLenum bufferMode);
GLAPI PFNGLTRANSFORMFEEDBACKSTREAMATTRIBSNVPROC glad_glTransformFeedbackStreamAttribsNV;
#define glTransformFeedbackStreamAttribsNV glad_glTransformFeedbackStreamAttribsNV
#endif
#ifndef GL_NV_transform_feedback2
#define GL_NV_transform_feedback2 1
GLAPI int GLAD_GL_NV_transform_feedback2;
typedef void (APIENTRYP PFNGLBINDTRANSFORMFEEDBACKNVPROC)(GLenum target, GLuint id);
GLAPI PFNGLBINDTRANSFORMFEEDBACKNVPROC glad_glBindTransformFeedbackNV;
#define glBindTransformFeedbackNV glad_glBindTransformFeedbackNV
typedef void (APIENTRYP PFNGLDELETETRANSFORMFEEDBACKSNVPROC)(GLsizei n, const GLuint *ids);
GLAPI PFNGLDELETETRANSFORMFEEDBACKSNVPROC glad_glDeleteTransformFeedbacksNV;
#define glDeleteTransformFeedbacksNV glad_glDeleteTransformFeedbacksNV
typedef void (APIENTRYP PFNGLGENTRANSFORMFEEDBACKSNVPROC)(GLsizei n, GLuint *ids);
GLAPI PFNGLGENTRANSFORMFEEDBACKSNVPROC glad_glGenTransformFeedbacksNV;
#define glGenTransformFeedbacksNV glad_glGenTransformFeedbacksNV
typedef GLboolean (APIENTRYP PFNGLISTRANSFORMFEEDBACKNVPROC)(GLuint id);
GLAPI PFNGLISTRANSFORMFEEDBACKNVPROC glad_glIsTransformFeedbackNV;
#define glIsTransformFeedbackNV glad_glIsTransformFeedbackNV
typedef void (APIENTRYP PFNGLPAUSETRANSFORMFEEDBACKNVPROC)(void);
GLAPI PFNGLPAUSETRANSFORMFEEDBACKNVPROC glad_glPauseTransformFeedbackNV;
#define glPauseTransformFeedbackNV glad_glPauseTransformFeedbackNV
typedef void (APIENTRYP PFNGLRESUMETRANSFORMFEEDBACKNVPROC)(void);
GLAPI PFNGLRESUMETRANSFORMFEEDBACKNVPROC glad_glResumeTransformFeedbackNV;
#define glResumeTransformFeedbackNV glad_glResumeTransformFeedbackNV
typedef void (APIENTRYP PFNGLDRAWTRANSFORMFEEDBACKNVPROC)(GLenum mode, GLuint id);
GLAPI PFNGLDRAWTRANSFORMFEEDBACKNVPROC glad_glDrawTransformFeedbackNV;
#define glDrawTransformFeedbackNV glad_glDrawTransformFeedbackNV
#endif
#ifndef GL_NV_uniform_buffer_std430_layout
#define GL_NV_uniform_buffer_std430_layout 1
GLAPI int GLAD_GL_NV_uniform_buffer_std430_layout;
#endif
```

```
#ifndef GL_NV_uniform_buffer_unified_memory
#define GL_NV_uniform_buffer_unified_memory 1
GLAPI int GLAD_GL_NV_uniform_buffer_unified_memory;
#endif
#ifndef GL_NV_vdpau_interop
#define GL_NV_vdpau_interop 1
GLAPI int GLAD_GL_NV_vdpau_interop;
typedef void (APIENTRYP PFNGLVDPAUINITNVPROC)(const void *vdpDevice, const void *getProcAddress);
GLAPI PFNGLVDPAUINITNVPROC glad_glVDPAUInitNV;
#define glVDPAUInitNV glad_glVDPAUInitNV
typedef void (APIENTRYP PFNGLVDPAUFININVPROC)(void);
GLAPI PFNGLVDPAUFININVPROC glad_glVDPAUFiniNV;
#define glVDPAUFiniNV glad_glVDPAUFiniNV
typedef GLvdpauSurfaceNV (APIENTRYP PFNGLVDPAUREGISTERVIDEOSURFACENVPROC)(const void *vdpSurface, GLenum target, GLsizei numTextureNames, const GLuint *textureNames);
GLAPI PFNGLVDPAUREGISTERVIDEOSURFACENVPROC glad_glVDPAURegisterVideoSurfaceNV;
#define glVDPAURegisterVideoSurfaceNV glad_glVDPAURegisterVideoSurfaceNV
typedef GLvdpauSurfaceNV (APIENTRYP PFNGLVDPAUREGISTEROUTPUTSURFACENVPROC)(const void *vdpSurface, GLenum target, GLsizei numTextureNames, const GLuint *textureNames);
GLAPI PFNGLVDPAUREGISTEROUTPUTSURFACENVPROC glad_glVDPAURegisterOutputSurfaceNV;
#define glVDPAURegisterOutputSurfaceNV glad_glVDPAURegisterOutputSurfaceNV
typedef GLboolean (APIENTRYP PFNGLVDPAUISSURFACENVPROC)(GLvdpauSurfaceNV surface);
GLAPI PFNGLVDPAUISSURFACENVPROC glad_glVDPAUIsSurfaceNV;
#define glVDPAUIsSurfaceNV glad_glVDPAUIsSurfaceNV
typedef void (APIENTRYP PFNGLVDPAUUNREGISTERSURFACENVPROC)(GLvdpauSurfaceNV surface);
GLAPI PFNGLVDPAUUNREGISTERSURFACENVPROC glad_glVDPAUUnregisterSurfaceNV;
#define glVDPAUUnregisterSurfaceNV glad_glVDPAUUnregisterSurfaceNV
typedef void (APIENTRYP PFNGLVDPAUGETSURFACEIVNVPROC)(GLvdpauSurfaceNV surface, GLenum pname, GLsizei count, GLsizei *length, GLint *values);
GLAPI PFNGLVDPAUGETSURFACEIVNVPROC glad_glVDPAUGetSurfaceivNV;
#define glVDPAUGetSurfaceivNV glad_glVDPAUGetSurfaceivNV
typedef void (APIENTRYP PFNGLVDPAUSURFACEACCESSNVPROC)(GLvdpauSurfaceNV surface, GLenum access);
GLAPI PFNGLVDPAUSURFACEACCESSNVPROC glad_glVDPAUSurfaceAccessNV;
#define glVDPAUSurfaceAccessNV glad_glVDPAUSurfaceAccessNV
typedef void (APIENTRYP PFNGLVDPAUMAPSURFACESNVPROC)(GLsizei numSurfaces, const GLvdpauSurfaceNV *surfaces);
GLAPI PFNGLVDPAUMAPSURFACESNVPROC glad_glVDPAUMapSurfacesNV;
#define glVDPAUMapSurfacesNV glad_glVDPAUMapSurfacesNV
typedef void (APIENTRYP PFNGLVDPAUUNMAPSURFACESNVPROC)(GLsizei numSurface, const GLvdpauSurfaceNV *surfaces);
GLAPI PFNGLVDPAUUNMAPSURFACESNVPROC glad_glVDPAUUnmapSurfacesNV;
#define glVDPAUUnmapSurfacesNV glad_glVDPAUUnmapSurfacesNV
#endif
#ifndef GL_NV_vdpau_interop2
#define GL_NV_vdpau_interop2 1
GLAPI int GLAD_GL_NV_vdpau_interop2;
typedef GLvdpauSurfaceNV (APIENTRYP PFNGLVDPAUREGISTERVIDEOSURFACEWITHPICTURESTRUCTURENVPROC)(const void *vdpSurface, GLenum target, GLsizei numTextureNames, const GLuint *textureNames, GLboolean isFrameStructure);
GLAPI PFNGLVDPAUREGISTERVIDEOSURFACEWITHPICTURESTRUCTURENVPROC glad_glVDPAURegisterVideoSurfaceWithPictureStructureNV;
#define glVDPAURegisterVideoSurfaceWithPictureStructureNV glad_glVDPAURegisterVideoSurfaceWithPictureStructureNV
#endif
#ifndef GL_NV_vertex_array_range
#define GL_NV_vertex_array_range 1
GLAPI int GLAD_GL_NV_vertex_array_range;
typedef void (APIENTRYP PFNGLFLUSHVERTEXARRAYRANGENVPROC)(void);
GLAPI PFNGLFLUSHVERTEXARRAYRANGENVPROC glad_glFlushVertexArrayRangeNV;
#define glFlushVertexArrayRangeNV glad_glFlushVertexArrayRangeNV
typedef void (APIENTRYP PFNGLVERTEXARRAYRANGENVPROC)(GLsizei length, const void *pointer);
GLAPI PFNGLVERTEXARRAYRANGENVPROC glad_glVertexArrayRangeNV;
#define glVertexArrayRangeNV glad_glVertexArrayRangeNV
#endif
#ifndef GL_NV_vertex_array_range2
#define GL_NV_vertex_array_range2 1
GLAPI int GLAD_GL_NV_vertex_array_range2;
#endif
#ifndef GL_NV_vertex_attrib_integer_64bit
#define GL_NV_vertex_attrib_integer_64bit 1
GLAPI int GLAD_GL_NV_vertex_attrib_integer_64bit;
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1I64NVPROC)(GLuint index, GLint64EXT x);
GLAPI PFNGLVERTEXATTRIBL1I64NVPROC glad_glVertexAttribL1i64NV;
#define glVertexAttribL1i64NV glad_glVertexAttribL1i64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2I64NVPROC)(GLuint index, GLint64EXT x, GLint64EXT y);
GLAPI PFNGLVERTEXATTRIBL2I64NVPROC glad_glVertexAttribL2i64NV;
#define glVertexAttribL2i64NV glad_glVertexAttribL2i64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3I64NVPROC)(GLuint index, GLint64EXT x, GLint64EXT y, GLint64EXT z);
GLAPI PFNGLVERTEXATTRIBL3I64NVPROC glad_glVertexAttribL3i64NV;
#define glVertexAttribL3i64NV glad_glVertexAttribL3i64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4I64NVPROC)(GLuint index, GLint64EXT x, GLint64EXT y, GLint64EXT z, GLint64EXT w);
GLAPI PFNGLVERTEXATTRIBL4I64NVPROC glad_glVertexAttribL4i64NV;
#define glVertexAttribL4i64NV glad_glVertexAttribL4i64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1I64VNVPROC)(GLuint index, const GLint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL1I64VNVPROC glad_glVertexAttribL1i64vNV;
```

```
#define glVertexAttribL1i64vNV glad_glVertexAttribL1i64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2I64VNVPROC)(GLuint index, const GLint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL2I64VNVPROC glad_glVertexAttribL2i64vNV;
#define glVertexAttribL2i64vNV glad_glVertexAttribL2i64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3I64VNVPROC)(GLuint index, const GLint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL3I64VNVPROC glad_glVertexAttribL3i64vNV;
#define glVertexAttribL3i64vNV glad_glVertexAttribL3i64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4I64VNVPROC)(GLuint index, const GLint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL4I64VNVPROC glad_glVertexAttribL4i64vNV;
#define glVertexAttribL4i64vNV glad_glVertexAttribL4i64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1UI64NVPROC)(GLuint index, GLuint64EXT x);
GLAPI PFNGLVERTEXATTRIBL1UI64NVPROC glad_glVertexAttribL1ui64NV;
#define glVertexAttribL1ui64NV glad_glVertexAttribL1ui64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2UI64NVPROC)(GLuint index, GLuint64EXT x, GLuint64EXT y);
GLAPI PFNGLVERTEXATTRIBL2UI64NVPROC glad_glVertexAttribL2ui64NV;
#define glVertexAttribL2ui64NV glad_glVertexAttribL2ui64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3UI64NVPROC)(GLuint index, GLuint64EXT x, GLuint64EXT y, GLuint64EXT z);
GLAPI PFNGLVERTEXATTRIBL3UI64NVPROC glad_glVertexAttribL3ui64NV;
#define glVertexAttribL3ui64NV glad_glVertexAttribL3ui64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4UI64NVPROC)(GLuint index, GLuint64EXT x, GLuint64EXT y, GLuint64EXT z, GLuint64EXT w);
GLAPI PFNGLVERTEXATTRIBL4UI64NVPROC glad_glVertexAttribL4ui64NV;
#define glVertexAttribL4ui64NV glad_glVertexAttribL4ui64NV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL1UI64VNVPROC)(GLuint index, const GLuint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL1UI64VNVPROC glad_glVertexAttribL1ui64vNV;
#define glVertexAttribL1ui64vNV glad_glVertexAttribL1ui64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL2UI64VNVPROC)(GLuint index, const GLuint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL2UI64VNVPROC glad_glVertexAttribL2ui64vNV;
#define glVertexAttribL2ui64vNV glad_glVertexAttribL2ui64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL3UI64VNVPROC)(GLuint index, const GLuint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL3UI64VNVPROC glad_glVertexAttribL3ui64vNV;
#define glVertexAttribL3ui64vNV glad_glVertexAttribL3ui64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBL4UI64VNVPROC)(GLuint index, const GLuint64EXT *v);
GLAPI PFNGLVERTEXATTRIBL4UI64VNVPROC glad_glVertexAttribL4ui64vNV;
#define glVertexAttribL4ui64vNV glad_glVertexAttribL4ui64vNV
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBLI64VNVPROC)(GLuint index, GLenum pname, GLint64EXT *params);
GLAPI PFNGLGETVERTEXATTRIBLI64VNVPROC glad_glGetVertexAttribLi64vNV;
#define glGetVertexAttribLi64vNV glad_glGetVertexAttribLi64vNV
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBLUI64VNVPROC)(GLuint index, GLenum pname, GLuint64EXT *params);
GLAPI PFNGLGETVERTEXATTRIBLUI64VNVPROC glad_glGetVertexAttribLui64vNV;
#define glGetVertexAttribLui64vNV glad_glGetVertexAttribLui64vNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBLFORMATNVPROC)(GLuint index, GLint size, GLenum type, GLsizei stride);
GLAPI PFNGLVERTEXATTRIBLFORMATNVPROC glad_glVertexAttribLFormatNV;
#define glVertexAttribLFormatNV glad_glVertexAttribLFormatNV
#endif
#ifndef GL_NV_vertex_buffer_unified_memory
#define GL_NV_vertex_buffer_unified_memory 1
GLAPI int GLAD_GL_NV_vertex_buffer_unified_memory;
typedef void (APIENTRYP PFNGLBUFFERADDRESSRANGENVPROC)(GLenum pname, GLuint index, GLuint64EXT address, GLsizeiptr length);
GLAPI PFNGLBUFFERADDRESSRANGENVPROC glad_glBufferAddressRangeNV;
#define glBufferAddressRangeNV glad_glBufferAddressRangeNV
typedef void (APIENTRYP PFNGLVERTEXFORMATNVPROC)(GLint size, GLenum type, GLsizei stride);
GLAPI PFNGLVERTEXFORMATNVPROC glad_glVertexFormatNV;
#define glVertexFormatNV glad_glVertexFormatNV
typedef void (APIENTRYP PFNGLNORMALFORMATNVPROC)(GLenum type, GLsizei stride);
GLAPI PFNGLNORMALFORMATNVPROC glad_glNormalFormatNV;
#define glNormalFormatNV glad_glNormalFormatNV
typedef void (APIENTRYP PFNGLCOLORFORMATNVPROC)(GLint size, GLenum type, GLsizei stride);
GLAPI PFNGLCOLORFORMATNVPROC glad_glColorFormatNV;
#define glColorFormatNV glad_glColorFormatNV
typedef void (APIENTRYP PFNGLINDEXFORMATNVPROC)(GLenum type, GLsizei stride);
GLAPI PFNGLINDEXFORMATNVPROC glad_glIndexFormatNV;
#define glIndexFormatNV glad_glIndexFormatNV
typedef void (APIENTRYP PFNGLTEXCOORDFORMATNVPROC)(GLint size, GLenum type, GLsizei stride);
GLAPI PFNGLTEXCOORDFORMATNVPROC glad_glTexCoordFormatNV;
#define glTexCoordFormatNV glad_glTexCoordFormatNV
typedef void (APIENTRYP PFNGLEDGEFLAGFORMATNVPROC)(GLsizei stride);
GLAPI PFNGLEDGEFLAGFORMATNVPROC glad_glEdgeFlagFormatNV;
#define glEdgeFlagFormatNV glad_glEdgeFlagFormatNV
typedef void (APIENTRYP PFNGLSECONDARYCOLORFORMATNVPROC)(GLint size, GLenum type, GLsizei stride);
GLAPI PFNGLSECONDARYCOLORFORMATNVPROC glad_glSecondaryColorFormatNV;
#define glSecondaryColorFormatNV glad_glSecondaryColorFormatNV
typedef void (APIENTRYP PFNGLFOGCOORDFORMATNVPROC)(GLenum type, GLsizei stride);
GLAPI PFNGLFOGCOORDFORMATNVPROC glad_glFogCoordFormatNV;
#define glFogCoordFormatNV glad_glFogCoordFormatNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBFORMATNVPROC)(GLuint index, GLint size, GLenum type, GLboolean normalized, GLsizei stride);
GLAPI PFNGLVERTEXATTRIBFORMATNVPROC glad_glVertexAttribFormatNV;
#define glVertexAttribFormatNV glad_glVertexAttribFormatNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBIFORMATNVPROC)(GLuint index, GLint size, GLenum type, GLsizei stride);
GLAPI PFNGLVERTEXATTRIBIFORMATNVPROC glad_glVertexAttribIFormatNV;
```

```
#define glVertexAttribIFormatNV glad_glVertexAttribIFormatNV
typedef void (APIENTRYP PFNGLGETINTEGERUI64I_VNVPROC)(GLenum value, GLuint index, GLuint64EXT *result);
GLAPI PFNGLGETINTEGERUI64I_VNVPROC glad_glGetIntegerui64i_vNV;
#define glGetIntegerui64i_vNV glad_glGetIntegerui64i_vNV
#endif
#ifndef GL_NV_vertex_program
#define GL_NV_vertex_program 1
GLAPI int GLAD_GL_NV_vertex_program;
typedef GLboolean (APIENTRYP PFNGLAREPROGRAMSRESIDENTNVPROC)(GLsizei n, const GLuint *programs, GLboolean *residences);
GLAPI PFNGLAREPROGRAMSRESIDENTNVPROC glad_glAreProgramsResidentNV;
#define glAreProgramsResidentNV glad_glAreProgramsResidentNV
typedef void (APIENTRYP PFNGLBINDPROGRAMNVPROC)(GLenum target, GLuint id);
GLAPI PFNGLBINDPROGRAMNVPROC glad_glBindProgramNV;
#define glBindProgramNV glad_glBindProgramNV
typedef void (APIENTRYP PFNGLDELETEPROGRAMSNVPROC)(GLsizei n, const GLuint *programs);
GLAPI PFNGLDELETEPROGRAMSNVPROC glad_glDeleteProgramsNV;
#define glDeleteProgramsNV glad_glDeleteProgramsNV
typedef void (APIENTRYP PFNGLEXECUTEPROGRAMNVPROC)(GLenum target, GLuint id, const GLfloat *params);
GLAPI PFNGLEXECUTEPROGRAMNVPROC glad_glExecuteProgramNV;
#define glExecuteProgramNV glad_glExecuteProgramNV
typedef void (APIENTRYP PFNGLGENPROGRAMSNVPROC)(GLsizei n, GLuint *programs);
GLAPI PFNGLGENPROGRAMSNVPROC glad_glGenProgramsNV;
#define glGenProgramsNV glad_glGenProgramsNV
typedef void (APIENTRYP PFNGLGETPROGRAMPARAMETERDVNVPROC)(GLenum target, GLuint index, GLenum pname, GLdouble *params);
GLAPI PFNGLGETPROGRAMPARAMETERDVNVPROC glad_glGetProgramParameterdvNV;
#define glGetProgramParameterdvNV glad_glGetProgramParameterdvNV
typedef void (APIENTRYP PFNGLGETPROGRAMPARAMETERFVNVPROC)(GLenum target, GLuint index, GLenum pname, GLfloat *params);
GLAPI PFNGLGETPROGRAMPARAMETERFVNVPROC glad_glGetProgramParameterfvNV;
#define glGetProgramParameterfvNV glad_glGetProgramParameterfvNV
typedef void (APIENTRYP PFNGLGETPROGRAMIVNVPROC)(GLuint id, GLenum pname, GLint *params);
GLAPI PFNGLGETPROGRAMIVNVPROC glad_glGetProgramivNV;
#define glGetProgramivNV glad_glGetProgramivNV
typedef void (APIENTRYP PFNGLGETPROGRAMSTRINGNVPROC)(GLuint id, GLenum pname, GLubyte *program);
GLAPI PFNGLGETPROGRAMSTRINGNVPROC glad_glGetProgramStringNV;
#define glGetProgramStringNV glad_glGetProgramStringNV
typedef void (APIENTRYP PFNGLGETTRACKMATRIXIVNVPROC)(GLenum target, GLuint address, GLenum pname, GLint *params);
GLAPI PFNGLGETTRACKMATRIXIVNVPROC glad_glGetTrackMatrixivNV;
#define glGetTrackMatrixivNV glad_glGetTrackMatrixivNV
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBDVNVPROC)(GLuint index, GLenum pname, GLdouble *params);
GLAPI PFNGLGETVERTEXATTRIBDVNVPROC glad_glGetVertexAttribdvNV;
#define glGetVertexAttribdvNV glad_glGetVertexAttribdvNV
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBFVNVPROC)(GLuint index, GLenum pname, GLfloat *params);
GLAPI PFNGLGETVERTEXATTRIBFVNVPROC glad_glGetVertexAttribfvNV;
#define glGetVertexAttribfvNV glad_glGetVertexAttribfvNV
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBIVNVPROC)(GLuint index, GLenum pname, GLint *params);
GLAPI PFNGLGETVERTEXATTRIBIVNVPROC glad_glGetVertexAttribivNV;
#define glGetVertexAttribivNV glad_glGetVertexAttribivNV
typedef void (APIENTRYP PFNGLGETVERTEXATTRIBPOINTERVNVPROC)(GLuint index, GLenum pname, void **pointer);
GLAPI PFNGLGETVERTEXATTRIBPOINTERVNVPROC glad_glGetVertexAttribPointervNV;
#define glGetVertexAttribPointervNV glad_glGetVertexAttribPointervNV
typedef GLboolean (APIENTRYP PFNGLISPROGRAMNVPROC)(GLuint id);
GLAPI PFNGLISPROGRAMNVPROC glad_glIsProgramNV;
#define glIsProgramNV glad_glIsProgramNV
typedef void (APIENTRYP PFNGLLOADPROGRAMNVPROC)(GLenum target, GLuint id, GLsizei len, const GLubyte *program);
GLAPI PFNGLLOADPROGRAMNVPROC glad_glLoadProgramNV;
#define glLoadProgramNV glad_glLoadProgramNV
typedef void (APIENTRYP PFNGLPROGRAMPARAMETER4DNVPROC)(GLenum target, GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLPROGRAMPARAMETER4DNVPROC glad_glProgramParameter4dNV;
#define glProgramParameter4dNV glad_glProgramParameter4dNV
typedef void (APIENTRYP PFNGLPROGRAMPARAMETER4DVNVPROC)(GLenum target, GLuint index, const GLdouble *v);
GLAPI PFNGLPROGRAMPARAMETER4DVNVPROC glad_glProgramParameter4dvNV;
#define glProgramParameter4dvNV glad_glProgramParameter4dvNV
typedef void (APIENTRYP PFNGLPROGRAMPARAMETER4FNVPROC)(GLenum target, GLuint index, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLPROGRAMPARAMETER4FNVPROC glad_glProgramParameter4fNV;
#define glProgramParameter4fNV glad_glProgramParameter4fNV
typedef void (APIENTRYP PFNGLPROGRAMPARAMETER4FVNVPROC)(GLenum target, GLuint index, const GLfloat *v);
GLAPI PFNGLPROGRAMPARAMETER4FVNVPROC glad_glProgramParameter4fvNV;
#define glProgramParameter4fvNV glad_glProgramParameter4fvNV
typedef void (APIENTRYP PFNGLPROGRAMPARAMETERS4DVNVPROC)(GLenum target, GLuint index, GLsizei count, const GLdouble *v);
GLAPI PFNGLPROGRAMPARAMETERS4DVNVPROC glad_glProgramParameters4dvNV;
#define glProgramParameters4dvNV glad_glProgramParameters4dvNV
typedef void (APIENTRYP PFNGLPROGRAMPARAMETERS4FVNVPROC)(GLenum target, GLuint index, GLsizei count, const GLfloat *v);
GLAPI PFNGLPROGRAMPARAMETERS4FVNVPROC glad_glProgramParameters4fvNV;
#define glProgramParameters4fvNV glad_glProgramParameters4fvNV
```

```
typedef void (APIENTRYP PFNGLREQUESTRESIDENTPROGRAMSNVPROC)(GLsizei n, const GLuint *programs);
GLAPI PFNGLREQUESTRESIDENTPROGRAMSNVPROC glad_glRequestResidentProgramsNV;
#define glRequestResidentProgramsNV glad_glRequestResidentProgramsNV
typedef void (APIENTRYP PFNGLTRACKMATRIXNVPROC)(GLenum target, GLuint address, GLenum matrix, GLenum transform);
GLAPI PFNGLTRACKMATRIXNVPROC glad_glTrackMatrixNV;
#define glTrackMatrixNV glad_glTrackMatrixNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBPOINTERNVPROC)(GLuint index, GLint fsize, GLenum type, GLsizei stride, const void *pointer);
GLAPI PFNGLVERTEXATTRIBPOINTERNVPROC glad_glVertexAttribPointerNV;
#define glVertexAttribPointerNV glad_glVertexAttribPointerNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1DNVPROC)(GLuint index, GLdouble x);
GLAPI PFNGLVERTEXATTRIB1DNVPROC glad_glVertexAttrib1dNV;
#define glVertexAttrib1dNV glad_glVertexAttrib1dNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1DVNVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB1DVNVPROC glad_glVertexAttrib1dvNV;
#define glVertexAttrib1dvNV glad_glVertexAttrib1dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1FNVPROC)(GLuint index, GLfloat x);
GLAPI PFNGLVERTEXATTRIB1FNVPROC glad_glVertexAttrib1fNV;
#define glVertexAttrib1fNV glad_glVertexAttrib1fNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1FVNVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB1FVNVPROC glad_glVertexAttrib1fvNV;
#define glVertexAttrib1fvNV glad_glVertexAttrib1fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1SNVPROC)(GLuint index, GLshort x);
GLAPI PFNGLVERTEXATTRIB1SNVPROC glad_glVertexAttrib1sNV;
#define glVertexAttrib1sNV glad_glVertexAttrib1sNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB1SVNVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB1SVNVPROC glad_glVertexAttrib1svNV;
#define glVertexAttrib1svNV glad_glVertexAttrib1svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2DNVPROC)(GLuint index, GLdouble x, GLdouble y);
GLAPI PFNGLVERTEXATTRIB2DNVPROC glad_glVertexAttrib2dNV;
#define glVertexAttrib2dNV glad_glVertexAttrib2dNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2DVNVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB2DVNVPROC glad_glVertexAttrib2dvNV;
#define glVertexAttrib2dvNV glad_glVertexAttrib2dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2FNVPROC)(GLuint index, GLfloat x, GLfloat y);
GLAPI PFNGLVERTEXATTRIB2FNVPROC glad_glVertexAttrib2fNV;
#define glVertexAttrib2fNV glad_glVertexAttrib2fNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2FVNVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB2FVNVPROC glad_glVertexAttrib2fvNV;
#define glVertexAttrib2fvNV glad_glVertexAttrib2fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2SNVPROC)(GLuint index, GLshort x, GLshort y);
GLAPI PFNGLVERTEXATTRIB2SNVPROC glad_glVertexAttrib2sNV;
#define glVertexAttrib2sNV glad_glVertexAttrib2sNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB2SVNVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB2SVNVPROC glad_glVertexAttrib2svNV;
#define glVertexAttrib2svNV glad_glVertexAttrib2svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3DNVPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z);
GLAPI PFNGLVERTEXATTRIB3DNVPROC glad_glVertexAttrib3dNV;
#define glVertexAttrib3dNV glad_glVertexAttrib3dNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3DVNVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB3DVNVPROC glad_glVertexAttrib3dvNV;
#define glVertexAttrib3dvNV glad_glVertexAttrib3dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3FNVPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLVERTEXATTRIB3FNVPROC glad_glVertexAttrib3fNV;
#define glVertexAttrib3fNV glad_glVertexAttrib3fNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3FVNVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB3FVNVPROC glad_glVertexAttrib3fvNV;
#define glVertexAttrib3fvNV glad_glVertexAttrib3fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3SNVPROC)(GLuint index, GLshort x, GLshort y, GLshort z);
GLAPI PFNGLVERTEXATTRIB3SNVPROC glad_glVertexAttrib3sNV;
#define glVertexAttrib3sNV glad_glVertexAttrib3sNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB3SVNVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB3SVNVPROC glad_glVertexAttrib3svNV;
#define glVertexAttrib3svNV glad_glVertexAttrib3svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4DNVPROC)(GLuint index, GLdouble x, GLdouble y, GLdouble z, GLdouble w);
GLAPI PFNGLVERTEXATTRIB4DNVPROC glad_glVertexAttrib4dNV;
#define glVertexAttrib4dNV glad_glVertexAttrib4dNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4DVNVPROC)(GLuint index, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIB4DVNVPROC glad_glVertexAttrib4dvNV;
#define glVertexAttrib4dvNV glad_glVertexAttrib4dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4FNVPROC)(GLuint index, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLVERTEXATTRIB4FNVPROC glad_glVertexAttrib4fNV;
#define glVertexAttrib4fNV glad_glVertexAttrib4fNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4FVNVPROC)(GLuint index, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIB4FVNVPROC glad_glVertexAttrib4fvNV;
#define glVertexAttrib4fvNV glad_glVertexAttrib4fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4SNVPROC)(GLuint index, GLshort x, GLshort y, GLshort z, GLshort w);
GLAPI PFNGLVERTEXATTRIB4SNVPROC glad_glVertexAttrib4sNV;
#define glVertexAttrib4sNV glad_glVertexAttrib4sNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4SVNVPROC)(GLuint index, const GLshort *v);
GLAPI PFNGLVERTEXATTRIB4SVNVPROC glad_glVertexAttrib4svNV;
#define glVertexAttrib4svNV glad_glVertexAttrib4svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4UBNVPROC)(GLuint index, GLubyte x, GLubyte y, GLubyte z,
GLubyte w);
GLAPI PFNGLVERTEXATTRIB4UBNVPROC glad_glVertexAttrib4ubNV;
#define glVertexAttrib4ubNV glad_glVertexAttrib4ubNV
typedef void (APIENTRYP PFNGLVERTEXATTRIB4UBVNVPROC)(GLuint index, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIB4UBVNVPROC glad_glVertexAttrib4ubvNV;
#define glVertexAttrib4ubvNV glad_glVertexAttrib4ubvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS1DVNVPROC)(GLuint index, GLsizei count, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBS1DVNVPROC glad_glVertexAttribs1dvNV;
#define glVertexAttribs1dvNV glad_glVertexAttribs1dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS1FVNVPROC)(GLuint index, GLsizei count, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIBS1FVNVPROC glad_glVertexAttribs1fvNV;
#define glVertexAttribs1fvNV glad_glVertexAttribs1fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS1SVNVPROC)(GLuint index, GLsizei count, const GLshort *v);
GLAPI PFNGLVERTEXATTRIBS1SVNVPROC glad_glVertexAttribs1svNV;
#define glVertexAttribs1svNV glad_glVertexAttribs1svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS2DVNVPROC)(GLuint index, GLsizei count, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBS2DVNVPROC glad_glVertexAttribs2dvNV;
#define glVertexAttribs2dvNV glad_glVertexAttribs2dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS2FVNVPROC)(GLuint index, GLsizei count, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIBS2FVNVPROC glad_glVertexAttribs2fvNV;
#define glVertexAttribs2fvNV glad_glVertexAttribs2fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS2SVNVPROC)(GLuint index, GLsizei count, const GLshort *v);
GLAPI PFNGLVERTEXATTRIBS2SVNVPROC glad_glVertexAttribs2svNV;
#define glVertexAttribs2svNV glad_glVertexAttribs2svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS3DVNVPROC)(GLuint index, GLsizei count, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBS3DVNVPROC glad_glVertexAttribs3dvNV;
#define glVertexAttribs3dvNV glad_glVertexAttribs3dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS3FVNVPROC)(GLuint index, GLsizei count, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIBS3FVNVPROC glad_glVertexAttribs3fvNV;
#define glVertexAttribs3fvNV glad_glVertexAttribs3fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS3SVNVPROC)(GLuint index, GLsizei count, const GLshort *v);
GLAPI PFNGLVERTEXATTRIBS3SVNVPROC glad_glVertexAttribs3svNV;
#define glVertexAttribs3svNV glad_glVertexAttribs3svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS4DVNVPROC)(GLuint index, GLsizei count, const GLdouble *v);
GLAPI PFNGLVERTEXATTRIBS4DVNVPROC glad_glVertexAttribs4dvNV;
#define glVertexAttribs4dvNV glad_glVertexAttribs4dvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS4FVNVPROC)(GLuint index, GLsizei count, const GLfloat *v);
GLAPI PFNGLVERTEXATTRIBS4FVNVPROC glad_glVertexAttribs4fvNV;
#define glVertexAttribs4fvNV glad_glVertexAttribs4fvNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS4SVNVPROC)(GLuint index, GLsizei count, const GLshort *v);
GLAPI PFNGLVERTEXATTRIBS4SVNVPROC glad_glVertexAttribs4svNV;
#define glVertexAttribs4svNV glad_glVertexAttribs4svNV
typedef void (APIENTRYP PFNGLVERTEXATTRIBS4UBVNVPROC)(GLuint index, GLsizei count, const GLubyte *v);
GLAPI PFNGLVERTEXATTRIBS4UBVNVPROC glad_glVertexAttribs4ubvNV;
#define glVertexAttribs4ubvNV glad_glVertexAttribs4ubvNV
#endif
#ifndef GL_NV_vertex_program1_1
#define GL_NV_vertex_program1_1 1
GLAPI int GLAD_GL_NV_vertex_program1_1;
#endif
#ifndef GL_NV_vertex_program2
#define GL_NV_vertex_program2 1
GLAPI int GLAD_GL_NV_vertex_program2;
#endif
#ifndef GL_NV_vertex_program2_option
#define GL_NV_vertex_program2_option 1
GLAPI int GLAD_GL_NV_vertex_program2_option;
#endif
#ifndef GL_NV_vertex_program3
#define GL_NV_vertex_program3 1
GLAPI int GLAD_GL_NV_vertex_program3;
#endif
#ifndef GL_NV_vertex_program4
#define GL_NV_vertex_program4 1
GLAPI int GLAD_GL_NV_vertex_program4;
#endif
#ifndef GL_NV_video_capture
#define GL_NV_video_capture 1
GLAPI int GLAD_GL_NV_video_capture;
typedef void (APIENTRYP PFNGLBEGINVIDEOCAPTURENVPROC)(GLuint video_capture_slot);
GLAPI PFNGLBEGINVIDEOCAPTURENVPROC glad_glBeginVideoCaptureNV;
#define glBeginVideoCaptureNV glad_glBeginVideoCaptureNV
typedef void (APIENTRYP PFNGLBINDVIDEOCAPTURESTREAMBUFFERNVPROC)(GLuint video_capture_slot, GLuint stream, GLenum frame_region, GLintptrARB offset);
GLAPI PFNGLBINDVIDEOCAPTURESTREAMBUFFERNVPROC glad_glBindVideoCaptureStreamBufferNV;
#define glBindVideoCaptureStreamBufferNV glad_glBindVideoCaptureStreamBufferNV
typedef void (APIENTRYP PFNGLBINDVIDEOCAPTURESTREAMTEXTURENVPROC)(GLuint video_capture_slot, GLuint stream, GLenum frame_region, GLenum target, GLuint texture);
GLAPI PFNGLBINDVIDEOCAPTURESTREAMTEXTURENVPROC glad_glBindVideoCaptureStreamTextureNV;
#define glBindVideoCaptureStreamTextureNV glad_glBindVideoCaptureStreamTextureNV
typedef void (APIENTRYP PFNGLENDVIDEOCAPTURENVPROC)(GLuint video_capture_slot);
GLAPI PFNGLENDVIDEOCAPTURENVPROC glad_glEndVideoCaptureNV;
#define glEndVideoCaptureNV glad_glEndVideoCaptureNV
typedef void (APIENTRYP PFNGLGETVIDEOCAPTUREIVNVPROC)(GLuint video_capture_slot, GLenum pname, GLint *params);
GLAPI PFNGLGETVIDEOCAPTUREIVNVPROC glad_glGetVideoCaptureivNV;
#define glGetVideoCaptureivNV glad_glGetVideoCaptureivNV
```

```
typedef void (APIENTRYP PFNGLGETVIDEOCAPTURESTREAMIVNVPROC)(GLuint video_capture_slot, GLuint stream, GLenum pname, GLint *params);
GLAPI PFNGLGETVIDEOCAPTURESTREAMIVNVPROC glad_glGetVideoCaptureStreamivNV;
#define glGetVideoCaptureStreamivNV glad_glGetVideoCaptureStreamivNV
typedef void (APIENTRYP PFNGLGETVIDEOCAPTURESTREAMFVNVPROC)(GLuint video_capture_slot, GLuint stream, GLenum pname, GLfloat *params);
GLAPI PFNGLGETVIDEOCAPTURESTREAMFVNVPROC glad_glGetVideoCaptureStreamfvNV;
#define glGetVideoCaptureStreamfvNV glad_glGetVideoCaptureStreamfvNV
typedef void (APIENTRYP PFNGLGETVIDEOCAPTURESTREAMDVNVPROC)(GLuint video_capture_slot, GLuint stream, GLenum pname, GLdouble *params);
GLAPI PFNGLGETVIDEOCAPTURESTREAMDVNVPROC glad_glGetVideoCaptureStreamdvNV;
#define glGetVideoCaptureStreamdvNV glad_glGetVideoCaptureStreamdvNV
typedef GLenum (APIENTRYP PFNGLVIDEOCAPTURENVPROC)(GLuint video_capture_slot, GLuint *sequence_num, GLuint64EXT *capture_time);
GLAPI PFNGLVIDEOCAPTURENVPROC glad_glVideoCaptureNV;
#define glVideoCaptureNV glad_glVideoCaptureNV
typedef void (APIENTRYP PFNGLVIDEOCAPTURESTREAMPARAMETERIVNVPROC)(GLuint video_capture_slot, GLuint stream, GLenum pname, const GLint *params);
GLAPI PFNGLVIDEOCAPTURESTREAMPARAMETERIVNVPROC glad_glVideoCaptureStreamParameterivNV;
#define glVideoCaptureStreamParameterivNV glad_glVideoCaptureStreamParameterivNV
typedef void (APIENTRYP PFNGLVIDEOCAPTURESTREAMPARAMETERFVNVPROC)(GLuint video_capture_slot, GLuint stream, GLenum pname, const GLfloat *params);
GLAPI PFNGLVIDEOCAPTURESTREAMPARAMETERFVNVPROC glad_glVideoCaptureStreamParameterfvNV;
#define glVideoCaptureStreamParameterfvNV glad_glVideoCaptureStreamParameterfvNV
typedef void (APIENTRYP PFNGLVIDEOCAPTURESTREAMPARAMETERDVNVPROC)(GLuint video_capture_slot, GLuint stream, GLenum pname, const GLdouble *params);
GLAPI PFNGLVIDEOCAPTURESTREAMPARAMETERDVNVPROC glad_glVideoCaptureStreamParameterdvNV;
#define glVideoCaptureStreamParameterdvNV glad_glVideoCaptureStreamParameterdvNV
#endif
#ifndef GL_NV_viewport_array2
#define GL_NV_viewport_array2 1
GLAPI int GLAD_GL_NV_viewport_array2;
#endif
#ifndef GL_NV_viewport_swizzle
#define GL_NV_viewport_swizzle 1
GLAPI int GLAD_GL_NV_viewport_swizzle;
typedef void (APIENTRYP PFNGLVIEWPORTSWIZZLENVPROC)(GLuint index, GLenum swizzlex, GLenum swizzley, GLenum swizzlez, GLenum swizzlew);
GLAPI PFNGLVIEWPORTSWIZZLENVPROC glad_glViewportSwizzleNV;
#define glViewportSwizzleNV glad_glViewportSwizzleNV
#endif
#ifndef GL_OES_byte_coordinates
#define GL_OES_byte_coordinates 1
GLAPI int GLAD_GL_OES_byte_coordinates;
typedef void (APIENTRYP PFNGLMULTITEXCOORD1BOESPROC)(GLenum texture, GLbyte s);
GLAPI PFNGLMULTITEXCOORD1BOESPROC glad_glMultiTexCoord1bOES;
#define glMultiTexCoord1bOES glad_glMultiTexCoord1bOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD1BVOESPROC)(GLenum texture, const GLbyte *coords);
GLAPI PFNGLMULTITEXCOORD1BVOESPROC glad_glMultiTexCoord1bvOES;
#define glMultiTexCoord1bvOES glad_glMultiTexCoord1bvOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD2BOESPROC)(GLenum texture, GLbyte s, GLbyte t);
GLAPI PFNGLMULTITEXCOORD2BOESPROC glad_glMultiTexCoord2bOES;
#define glMultiTexCoord2bOES glad_glMultiTexCoord2bOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD2BVOESPROC)(GLenum texture, const GLbyte *coords);
GLAPI PFNGLMULTITEXCOORD2BVOESPROC glad_glMultiTexCoord2bvOES;
#define glMultiTexCoord2bvOES glad_glMultiTexCoord2bvOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD3BOESPROC)(GLenum texture, GLbyte s, GLbyte t, GLbyte r);
GLAPI PFNGLMULTITEXCOORD3BOESPROC glad_glMultiTexCoord3bOES;
#define glMultiTexCoord3bOES glad_glMultiTexCoord3bOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD3BVOESPROC)(GLenum texture, const GLbyte *coords);
GLAPI PFNGLMULTITEXCOORD3BVOESPROC glad_glMultiTexCoord3bvOES;
#define glMultiTexCoord3bvOES glad_glMultiTexCoord3bvOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD4BOESPROC)(GLenum texture, GLbyte s, GLbyte t, GLbyte r, GLbyte q);
GLAPI PFNGLMULTITEXCOORD4BOESPROC glad_glMultiTexCoord4bOES;
#define glMultiTexCoord4bOES glad_glMultiTexCoord4bOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD4BVOESPROC)(GLenum texture, const GLbyte *coords);
GLAPI PFNGLMULTITEXCOORD4BVOESPROC glad_glMultiTexCoord4bvOES;
#define glMultiTexCoord4bvOES glad_glMultiTexCoord4bvOES
typedef void (APIENTRYP PFNGLTEXCOORD1BOESPROC)(GLbyte s);
GLAPI PFNGLTEXCOORD1BOESPROC glad_glTexCoord1bOES;
#define glTexCoord1bOES glad_glTexCoord1bOES
typedef void (APIENTRYP PFNGLTEXCOORD1BVOESPROC)(const GLbyte *coords);
GLAPI PFNGLTEXCOORD1BVOESPROC glad_glTexCoord1bvOES;
#define glTexCoord1bvOES glad_glTexCoord1bvOES
typedef void (APIENTRYP PFNGLTEXCOORD2BOESPROC)(GLbyte s, GLbyte t);
GLAPI PFNGLTEXCOORD2BOESPROC glad_glTexCoord2bOES;
#define glTexCoord2bOES glad_glTexCoord2bOES
typedef void (APIENTRYP PFNGLTEXCOORD2BVOESPROC)(const GLbyte *coords);
GLAPI PFNGLTEXCOORD2BVOESPROC glad_glTexCoord2bvOES;
#define glTexCoord2bvOES glad_glTexCoord2bvOES
typedef void (APIENTRYP PFNGLTEXCOORD3BOESPROC)(GLbyte s, GLbyte t, GLbyte r);
GLAPI PFNGLTEXCOORD3BOESPROC glad_glTexCoord3bOES;
#define glTexCoord3bOES glad_glTexCoord3bOES
typedef void (APIENTRYP PFNGLTEXCOORD3BVOESPROC)(const GLbyte *coords);
GLAPI PFNGLTEXCOORD3BVOESPROC glad_glTexCoord3bvOES;
#define glTexCoord3bvOES glad_glTexCoord3bvOES
```

```
typedef void (APIENTRYP PFNGLTEXCOORD4BOESPROC)(GLbyte s, GLbyte t, GLbyte r, GLbyte q);
GLAPI PFNGLTEXCOORD4BOESPROC glad_glTexCoord4bOES;
#define glTexCoord4bOES glad_glTexCoord4bOES
typedef void (APIENTRYP PFNGLTEXCOORD4BVOESPROC)(const GLbyte *coords);
GLAPI PFNGLTEXCOORD4BVOESPROC glad_glTexCoord4bvOES;
#define glTexCoord4bvOES glad_glTexCoord4bvOES
typedef void (APIENTRYP PFNGLVERTEX2BOESPROC)(GLbyte x, GLbyte y);
GLAPI PFNGLVERTEX2BOESPROC glad_glVertex2bOES;
#define glVertex2bOES glad_glVertex2bOES
typedef void (APIENTRYP PFNGLVERTEX2BVOESPROC)(const GLbyte *coords);
GLAPI PFNGLVERTEX2BVOESPROC glad_glVertex2bvOES;
#define glVertex2bvOES glad_glVertex2bvOES
typedef void (APIENTRYP PFNGLVERTEX3BOESPROC)(GLbyte x, GLbyte y, GLbyte z);
GLAPI PFNGLVERTEX3BOESPROC glad_glVertex3bOES;
#define glVertex3bOES glad_glVertex3bOES
typedef void (APIENTRYP PFNGLVERTEX3BVOESPROC)(const GLbyte *coords);
GLAPI PFNGLVERTEX3BVOESPROC glad_glVertex3bvOES;
#define glVertex3bvOES glad_glVertex3bvOES
typedef void (APIENTRYP PFNGLVERTEX4BOESPROC)(GLbyte x, GLbyte y, GLbyte z, GLbyte w);
GLAPI PFNGLVERTEX4BOESPROC glad_glVertex4bOES;
#define glVertex4bOES glad_glVertex4bOES
typedef void (APIENTRYP PFNGLVERTEX4BVOESPROC)(const GLbyte *coords);
GLAPI PFNGLVERTEX4BVOESPROC glad_glVertex4bvOES;
#define glVertex4bvOES glad_glVertex4bvOES
#endif
#ifndef GL_OES_compressed_paletted_texture
#define GL_OES_compressed_paletted_texture 1
GLAPI int GLAD_GL_OES_compressed_paletted_texture;
#endif
#ifndef GL_OES_fixed_point
#define GL_OES_fixed_point 1
GLAPI int GLAD_GL_OES_fixed_point;
typedef void (APIENTRYP PFNGLALPHAFUNCXOESPROC)(GLenum func, GLfixed ref);
GLAPI PFNGLALPHAFUNCXOESPROC glad_glAlphaFuncxOES;
#define glAlphaFuncxOES glad_glAlphaFuncxOES
typedef void (APIENTRYP PFNGLCLEARCOLORXOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed alpha);
GLAPI PFNGLCLEARCOLORXOESPROC glad_glClearColorxOES;
#define glClearColorxOES glad_glClearColorxOES
typedef void (APIENTRYP PFNGLCLEARDEPTHXOESPROC)(GLfixed depth);
GLAPI PFNGLCLEARDEPTHXOESPROC glad_glClearDepthxOES;
#define glClearDepthxOES glad_glClearDepthxOES
typedef void (APIENTRYP PFNGLCLIPPLANEXOESPROC)(GLenum plane, const GLfixed *equation);
GLAPI PFNGLCLIPPLANEXOESPROC glad_glClipPlanexOES;
#define glClipPlanexOES glad_glClipPlanexOES
typedef void (APIENTRYP PFNGLCOLOR4XOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed alpha);
GLAPI PFNGLCOLOR4XOESPROC glad_glColor4xOES;
#define glColor4xOES glad_glColor4xOES
typedef void (APIENTRYP PFNGLDEPTHRANGEXOESPROC)(GLfixed n, GLfixed f);
GLAPI PFNGLDEPTHRANGEXOESPROC glad_glDepthRangexOES;
#define glDepthRangexOES glad_glDepthRangexOES
typedef void (APIENTRYP PFNGLFOGXOESPROC)(GLenum pname, GLfixed param);
GLAPI PFNGLFOGXOESPROC glad_glFogxOES;
#define glFogxOES glad_glFogxOES
typedef void (APIENTRYP PFNGLFOGXVOESPROC)(GLenum pname, const GLfixed *param);
GLAPI PFNGLFOGXVOESPROC glad_glFogxvOES;
#define glFogxvOES glad_glFogxvOES
typedef void (APIENTRYP PFNGLFRUSTUMXOESPROC)(GLfixed l, GLfixed r, GLfixed b, GLfixed t, GLfixed n, GLfixed f);
GLAPI PFNGLFRUSTUMXOESPROC glad_glFrustumxOES;
#define glFrustumxOES glad_glFrustumxOES
typedef void (APIENTRYP PFNGLGETCLIPPLANEXOESPROC)(GLenum plane, GLfixed *equation);
GLAPI PFNGLGETCLIPPLANEXOESPROC glad_glGetClipPlanexOES;
#define glGetClipPlanexOES glad_glGetClipPlanexOES
typedef void (APIENTRYP PFNGLGETFIXEDVOESPROC)(GLenum pname, GLfixed *params);
GLAPI PFNGLGETFIXEDVOESPROC glad_glGetFixedvOES;
#define glGetFixedvOES glad_glGetFixedvOES
typedef void (APIENTRYP PFNGLGETTEXENVXVOESPROC)(GLenum target, GLenum pname, GLfixed *params);
GLAPI PFNGLGETTEXENVXVOESPROC glad_glGetTexEnvxvOES;
#define glGetTexEnvxvOES glad_glGetTexEnvxvOES
typedef void (APIENTRYP PFNGLGETTEXPARAMETERXVOESPROC)(GLenum target, GLenum pname, GLfixed *params);
GLAPI PFNGLGETTEXPARAMETERXVOESPROC glad_glGetTexParameterxvOES;
#define glGetTexParameterxvOES glad_glGetTexParameterxvOES
typedef void (APIENTRYP PFNGLLIGHTMODELXOESPROC)(GLenum pname, GLfixed param);
GLAPI PFNGLLIGHTMODELXOESPROC glad_glLightModelxOES;
#define glLightModelxOES glad_glLightModelxOES
typedef void (APIENTRYP PFNGLLIGHTMODELXVOESPROC)(GLenum pname, const GLfixed *param);
GLAPI PFNGLLIGHTMODELXVOESPROC glad_glLightModelxvOES;
#define glLightModelxvOES glad_glLightModelxvOES
typedef void (APIENTRYP PFNGLLIGHTXOESPROC)(GLenum light, GLenum pname, GLfixed param);
GLAPI PFNGLLIGHTXOESPROC glad_glLightxOES;
#define glLightxOES glad_glLightxOES
typedef void (APIENTRYP PFNGLLIGHTXVOESPROC)(GLenum light, GLenum pname, const GLfixed *params);
GLAPI PFNGLLIGHTXVOESPROC glad_glLightxvOES;
#define glLightxvOES glad_glLightxvOES
typedef void (APIENTRYP PFNGLLINEWIDTHXOESPROC)(GLfixed width);
GLAPI PFNGLLINEWIDTHXOESPROC glad_glLineWidthxOES;
```

```
#define glLineWidthxOES glad_glLineWidthxOES
typedef void (APIENTRYP PFNGLLOADMATRIXXOESPROC)(const GLfixed *m);
GLAPI PFNGLLOADMATRIXXOESPROC glad_glLoadMatrixxOES;
#define glLoadMatrixxOES glad_glLoadMatrixxOES
typedef void (APIENTRYP PFNGLMATERIALXOESPROC)(GLenum face, GLenum pname, GLfixed param);
GLAPI PFNGLMATERIALXOESPROC glad_glMaterialxOES;
#define glMaterialxOES glad_glMaterialxOES
typedef void (APIENTRYP PFNGLMATERIALXVOESPROC)(GLenum face, GLenum pname, const GLfixed *param);
GLAPI PFNGLMATERIALXVOESPROC glad_glMaterialxvOES;
#define glMaterialxvOES glad_glMaterialxvOES
typedef void (APIENTRYP PFNGLMULTMATRIXXOESPROC)(const GLfixed *m);
GLAPI PFNGLMULTMATRIXXOESPROC glad_glMultMatrixxOES;
#define glMultMatrixxOES glad_glMultMatrixxOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD4XOESPROC)(GLenum texture, GLfixed s, GLfixed t, GLfixed r, GLfixed q);
GLAPI PFNGLMULTITEXCOORD4XOESPROC glad_glMultiTexCoord4xOES;
#define glMultiTexCoord4xOES glad_glMultiTexCoord4xOES
typedef void (APIENTRYP PFNGLNORMAL3XOESPROC)(GLfixed nx, GLfixed ny, GLfixed nz);
GLAPI PFNGLNORMAL3XOESPROC glad_glNormal3xOES;
#define glNormal3xOES glad_glNormal3xOES
typedef void (APIENTRYP PFNGLORTHOXOESPROC)(GLfixed l, GLfixed r, GLfixed b, GLfixed t, GLfixed n, GLfixed f);
GLAPI PFNGLORTHOXOESPROC glad_glOrthoxOES;
#define glOrthoxOES glad_glOrthoxOES
typedef void (APIENTRYP PFNGLPOINTPARAMETERXVOESPROC)(GLenum pname, const GLfixed *params);
GLAPI PFNGLPOINTPARAMETERXVOESPROC glad_glPointParameterxvOES;
#define glPointParameterxvOES glad_glPointParameterxvOES
typedef void (APIENTRYP PFNGLPOINTSIZEXOESPROC)(GLfixed size);
GLAPI PFNGLPOINTSIZEXOESPROC glad_glPointSizexOES;
#define glPointSizexOES glad_glPointSizexOES
typedef void (APIENTRYP PFNGLPOLYGONOFFSETXOESPROC)(GLfixed factor, GLfixed units);
GLAPI PFNGLPOLYGONOFFSETXOESPROC glad_glPolygonOffsetxOES;
#define glPolygonOffsetxOES glad_glPolygonOffsetxOES
typedef void (APIENTRYP PFNGLROTATEXOESPROC)(GLfixed angle, GLfixed x, GLfixed y, GLfixed z);
GLAPI PFNGLROTATEXOESPROC glad_glRotatexOES;
#define glRotatexOES glad_glRotatexOES
typedef void (APIENTRYP PFNGLSCALEXOESPROC)(GLfixed x, GLfixed y, GLfixed z);
GLAPI PFNGLSCALEXOESPROC glad_glScalexOES;
#define glScalexOES glad_glScalexOES
typedef void (APIENTRYP PFNGLTEXENVXOESPROC)(GLenum target, GLenum pname, GLfixed param);
GLAPI PFNGLTEXENVXOESPROC glad_glTexEnvxOES;
#define glTexEnvxOES glad_glTexEnvxOES
typedef void (APIENTRYP PFNGLTEXENVXVOESPROC)(GLenum target, GLenum pname, const GLfixed *params);
GLAPI PFNGLTEXENVXVOESPROC glad_glTexEnvxvOES;
#define glTexEnvxvOES glad_glTexEnvxvOES
typedef void (APIENTRYP PFNGLTEXPARAMETERXOESPROC)(GLenum target, GLenum pname, GLfixed param);
GLAPI PFNGLTEXPARAMETERXOESPROC glad_glTexParameterxOES;
#define glTexParameterxOES glad_glTexParameterxOES
typedef void (APIENTRYP PFNGLTEXPARAMETERXVOESPROC)(GLenum target, GLenum pname, const GLfixed *params);
GLAPI PFNGLTEXPARAMETERXVOESPROC glad_glTexParameterxvOES;
#define glTexParameterxvOES glad_glTexParameterxvOES
typedef void (APIENTRYP PFNGLTRANSLATEXOESPROC)(GLfixed x, GLfixed y, GLfixed z);
GLAPI PFNGLTRANSLATEXOESPROC glad_glTranslatexOES;
#define glTranslatexOES glad_glTranslatexOES
typedef void (APIENTRYP PFNGLGETLIGHTXVOESPROC)(GLenum light, GLenum pname, GLfixed *params);
GLAPI PFNGLGETLIGHTXVOESPROC glad_glGetLightxvOES;
#define glGetLightxvOES glad_glGetLightxvOES
typedef void (APIENTRYP PFNGLGETMATERIALXVOESPROC)(GLenum face, GLenum pname, GLfixed *params);
GLAPI PFNGLGETMATERIALXVOESPROC glad_glGetMaterialxvOES;
#define glGetMaterialxvOES glad_glGetMaterialxvOES
typedef void (APIENTRYP PFNGLPOINTPARAMETERXOESPROC)(GLenum pname, GLfixed param);
GLAPI PFNGLPOINTPARAMETERXOESPROC glad_glPointParameterxOES;
#define glPointParameterxOES glad_glPointParameterxOES
typedef void (APIENTRYP PFNGLSAMPLECOVERAGEXOESPROC)(GLclampx value, GLboolean invert);
GLAPI PFNGLSAMPLECOVERAGEXOESPROC glad_glSampleCoveragexOES;
#define glSampleCoveragexOES glad_glSampleCoveragexOES
typedef void (APIENTRYP PFNGLACCUMXOESPROC)(GLenum op, GLfixed value);
GLAPI PFNGLACCUMXOESPROC glad_glAccumxOES;
#define glAccumxOES glad_glAccumxOES
typedef void (APIENTRYP PFNGLBITMAPXOESPROC)(GLsizei width, GLsizei height, GLfixed xorig, GLfixed yorig, GLfixed xmove, GLfixed ymove, const GLubyte *bitmap);
GLAPI PFNGLBITMAPXOESPROC glad_glBitmapxOES;
#define glBitmapxOES glad_glBitmapxOES
typedef void (APIENTRYP PFNGLBLENDCOLORXOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed alpha);
GLAPI PFNGLBLENDCOLORXOESPROC glad_glBlendColorxOES;
#define glBlendColorxOES glad_glBlendColorxOES
typedef void (APIENTRYP PFNGLCLEARACCUMXOESPROC)(GLfixed red, GLfixed green, GLfixed blue, GLfixed alpha);
GLAPI PFNGLCLEARACCUMXOESPROC glad_glClearAccumxOES;
#define glClearAccumxOES glad_glClearAccumxOES
typedef void (APIENTRYP PFNGLCOLOR3XOESPROC)(GLfixed red, GLfixed green, GLfixed blue);
GLAPI PFNGLCOLOR3XOESPROC glad_glColor3xOES;
#define glColor3xOES glad_glColor3xOES
typedef void (APIENTRYP PFNGLCOLOR3XVOESPROC)(const GLfixed *components);
GLAPI PFNGLCOLOR3XVOESPROC glad_glColor3xvOES;
```

```
#define glColor3xvOES glad_glColor3xvOES
typedef void (APIENTRYP PFNGLCOLOR4XVOESPROC)(const GLfixed *components);
GLAPI PFNGLCOLOR4XVOESPROC glad_glColor4xvOES;
#define glColor4xvOES glad_glColor4xvOES
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERXOESPROC)(GLenum target, GLenum pname, GLfixed param);
GLAPI PFNGLCONVOLUTIONPARAMETERXOESPROC glad_glConvolutionParameterxOES;
#define glConvolutionParameterxOES glad_glConvolutionParameterxOES
typedef void (APIENTRYP PFNGLCONVOLUTIONPARAMETERXVOESPROC)(GLenum target, GLenum pname, const GLfixed *params);
GLAPI PFNGLCONVOLUTIONPARAMETERXVOESPROC glad_glConvolutionParameterxvOES;
#define glConvolutionParameterxvOES glad_glConvolutionParameterxvOES
typedef void (APIENTRYP PFNGLEVALCOORD1XOESPROC)(GLfixed u);
GLAPI PFNGLEVALCOORD1XOESPROC glad_glEvalCoord1xOES;
#define glEvalCoord1xOES glad_glEvalCoord1xOES
typedef void (APIENTRYP PFNGLEVALCOORD1XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLEVALCOORD1XVOESPROC glad_glEvalCoord1xvOES;
#define glEvalCoord1xvOES glad_glEvalCoord1xvOES
typedef void (APIENTRYP PFNGLEVALCOORD2XOESPROC)(GLfixed u, GLfixed v);
GLAPI PFNGLEVALCOORD2XOESPROC glad_glEvalCoord2xOES;
#define glEvalCoord2xOES glad_glEvalCoord2xOES
typedef void (APIENTRYP PFNGLEVALCOORD2XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLEVALCOORD2XVOESPROC glad_glEvalCoord2xvOES;
#define glEvalCoord2xvOES glad_glEvalCoord2xvOES
typedef void (APIENTRYP PFNGLFEEDBACKBUFFERXOESPROC)(GLsizei n, GLenum type, const GLfixed *buffer);
GLAPI PFNGLFEEDBACKBUFFERXOESPROC glad_glFeedbackBufferxOES;
#define glFeedbackBufferxOES glad_glFeedbackBufferxOES
typedef void (APIENTRYP PFNGLGETCONVOLUTIONPARAMETERXVOESPROC)(GLenum target, GLenum pname, GLfixed *params);
GLAPI PFNGLGETCONVOLUTIONPARAMETERXVOESPROC glad_glGetConvolutionParameterxvOES;
#define glGetConvolutionParameterxvOES glad_glGetConvolutionParameterxvOES
typedef void (APIENTRYP PFNGLGETHISTOGRAMPARAMETERXVOESPROC)(GLenum target, GLenum pname, GLfixed *params);
GLAPI PFNGLGETHISTOGRAMPARAMETERXVOESPROC glad_glGetHistogramParameterxvOES;
#define glGetHistogramParameterxvOES glad_glGetHistogramParameterxvOES
typedef void (APIENTRYP PFNGLGETLIGHTXOESPROC)(GLenum light, GLenum pname, GLfixed *params);
GLAPI PFNGLGETLIGHTXOESPROC glad_glGetLightxOES;
#define glGetLightxOES glad_glGetLightxOES
typedef void (APIENTRYP PFNGLGETMAPXVOESPROC)(GLenum target, GLenum query, GLfixed *v);
GLAPI PFNGLGETMAPXVOESPROC glad_glGetMapxvOES;
#define glGetMapxvOES glad_glGetMapxvOES
typedef void (APIENTRYP PFNGLGETMATERIALXOESPROC)(GLenum face, GLenum pname, GLfixed param);
GLAPI PFNGLGETMATERIALXOESPROC glad_glGetMaterialxOES;
#define glGetMaterialxOES glad_glGetMaterialxOES
typedef void (APIENTRYP PFNGLGETPIXELMAPXVPROC)(GLenum map, GLint size, GLfixed *values);
GLAPI PFNGLGETPIXELMAPXVPROC glad_glGetPixelMapxv;
#define glGetPixelMapxv glad_glGetPixelMapxv
typedef void (APIENTRYP PFNGLGETTEXGENXVOESPROC)(GLenum coord, GLenum pname, GLfixed *params);
GLAPI PFNGLGETTEXGENXVOESPROC glad_glGetTexGenxvOES;
#define glGetTexGenxvOES glad_glGetTexGenxvOES
typedef void (APIENTRYP PFNGLGETTEXLEVELPARAMETERXVOESPROC)(GLenum target, GLint level, GLenum pname, GLfixed *params);
GLAPI PFNGLGETTEXLEVELPARAMETERXVOESPROC glad_glGetTexLevelParameterxvOES;
#define glGetTexLevelParameterxvOES glad_glGetTexLevelParameterxvOES
typedef void (APIENTRYP PFNGLINDEXXOESPROC)(GLfixed component);
GLAPI PFNGLINDEXXOESPROC glad_glIndexxOES;
#define glIndexxOES glad_glIndexxOES
typedef void (APIENTRYP PFNGLINDEXXVOESPROC)(const GLfixed *component);
GLAPI PFNGLINDEXXVOESPROC glad_glIndexxvOES;
#define glIndexxvOES glad_glIndexxvOES
typedef void (APIENTRYP PFNGLLOADTRANSPOSEMATRIXXOESPROC)(const GLfixed *m);
GLAPI PFNGLLOADTRANSPOSEMATRIXXOESPROC glad_glLoadTransposeMatrixxOES;
#define glLoadTransposeMatrixxOES glad_glLoadTransposeMatrixxOES
typedef void (APIENTRYP PFNGLMAP1XOESPROC)(GLenum target, GLfixed u1, GLfixed u2, GLint stride, GLint order, GLfixed points);
GLAPI PFNGLMAP1XOESPROC glad_glMap1xOES;
#define glMap1xOES glad_glMap1xOES
typedef void (APIENTRYP PFNGLMAP2XOESPROC)(GLenum target, GLfixed u1, GLfixed u2, GLint ustride, GLint uorder, GLfixed v1, GLfixed v2, GLint vstride, GLint vorder, GLfixed points);
GLAPI PFNGLMAP2XOESPROC glad_glMap2xOES;
#define glMap2xOES glad_glMap2xOES
typedef void (APIENTRYP PFNGLMAPGRID1XOESPROC)(GLint n, GLfixed u1, GLfixed u2);
GLAPI PFNGLMAPGRID1XOESPROC glad_glMapGrid1xOES;
#define glMapGrid1xOES glad_glMapGrid1xOES
typedef void (APIENTRYP PFNGLMAPGRID2XOESPROC)(GLint n, GLfixed u1, GLfixed u2, GLfixed v1, GLfixed v2);
GLAPI PFNGLMAPGRID2XOESPROC glad_glMapGrid2xOES;
#define glMapGrid2xOES glad_glMapGrid2xOES
typedef void (APIENTRYP PFNGLMULTTRANSPOSEMATRIXXOESPROC)(const GLfixed *m);
GLAPI PFNGLMULTTRANSPOSEMATRIXXOESPROC glad_glMultTransposeMatrixxOES;
#define glMultTransposeMatrixxOES glad_glMultTransposeMatrixxOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD1XOESPROC)(GLenum texture, GLfixed s);
GLAPI PFNGLMULTITEXCOORD1XOESPROC glad_glMultiTexCoord1xOES;
#define glMultiTexCoord1xOES glad_glMultiTexCoord1xOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD1XVOESPROC)(GLenum texture, const GLfixed *coords);
GLAPI PFNGLMULTITEXCOORD1XVOESPROC glad_glMultiTexCoord1xvOES;
#define glMultiTexCoord1xvOES glad_glMultiTexCoord1xvOES
```

```
typedef void (APIENTRYP PFNGLMULTITEXCOORD2XOESPROC)(GLenum texture, GLfixed s, GLfixed t);
GLAPI PFNGLMULTITEXCOORD2XOESPROC glad_glMultiTexCoord2xOES;
#define glMultiTexCoord2xOES glad_glMultiTexCoord2xOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD2XVOESPROC)(GLenum texture, const GLfixed *coords);
GLAPI PFNGLMULTITEXCOORD2XVOESPROC glad_glMultiTexCoord2xvOES;
#define glMultiTexCoord2xvOES glad_glMultiTexCoord2xvOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD3XOESPROC)(GLenum texture, GLfixed s, GLfixed t, GLfixed r);
GLAPI PFNGLMULTITEXCOORD3XOESPROC glad_glMultiTexCoord3xOES;
#define glMultiTexCoord3xOES glad_glMultiTexCoord3xOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD3XVOESPROC)(GLenum texture, const GLfixed *coords);
GLAPI PFNGLMULTITEXCOORD3XVOESPROC glad_glMultiTexCoord3xvOES;
#define glMultiTexCoord3xvOES glad_glMultiTexCoord3xvOES
typedef void (APIENTRYP PFNGLMULTITEXCOORD4XVOESPROC)(GLenum texture, const GLfixed *coords);
GLAPI PFNGLMULTITEXCOORD4XVOESPROC glad_glMultiTexCoord4xvOES;
#define glMultiTexCoord4xvOES glad_glMultiTexCoord4xvOES
typedef void (APIENTRYP PFNGLNORMAL3XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLNORMAL3XVOESPROC glad_glNormal3xvOES;
#define glNormal3xvOES glad_glNormal3xvOES
typedef void (APIENTRYP PFNGLPASSTHROUGHXOESPROC)(GLfixed token);
GLAPI PFNGLPASSTHROUGHXOESPROC glad_glPassThroughxOES;
#define glPassThroughxOES glad_glPassThroughxOES
typedef void (APIENTRYP PFNGLPIXELMAPXPROC)(GLenum map, GLint size, const GLfixed *values);
GLAPI PFNGLPIXELMAPXPROC glad_glPixelMapx;
#define glPixelMapx glad_glPixelMapx
typedef void (APIENTRYP PFNGLPIXELSTOREXPROC)(GLenum pname, GLfixed param);
GLAPI PFNGLPIXELSTOREXPROC glad_glPixelStorex;
#define glPixelStorex glad_glPixelStorex
typedef void (APIENTRYP PFNGLPIXELTRANSFERXOESPROC)(GLenum pname, GLfixed param);
GLAPI PFNGLPIXELTRANSFERXOESPROC glad_glPixelTransferxOES;
#define glPixelTransferxOES glad_glPixelTransferxOES
typedef void (APIENTRYP PFNGLPIXELZOOMXOESPROC)(GLfixed xfactor, GLfixed yfactor);
GLAPI PFNGLPIXELZOOMXOESPROC glad_glPixelZoomxOES;
#define glPixelZoomxOES glad_glPixelZoomxOES
typedef void (APIENTRYP PFNGLPRIORITIZETEXTURESXOESPROC)(GLsizei n, const GLuint *textures, const GLfixed *priorities);
GLAPI PFNGLPRIORITIZETEXTURESXOESPROC glad_glPrioritizeTexturesxOES;
#define glPrioritizeTexturesxOES glad_glPrioritizeTexturesxOES
typedef void (APIENTRYP PFNGLRASTERPOS2XOESPROC)(GLfixed x, GLfixed y);
GLAPI PFNGLRASTERPOS2XOESPROC glad_glRasterPos2xOES;
#define glRasterPos2xOES glad_glRasterPos2xOES
typedef void (APIENTRYP PFNGLRASTERPOS2XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLRASTERPOS2XVOESPROC glad_glRasterPos2xvOES;
#define glRasterPos2xvOES glad_glRasterPos2xvOES
typedef void (APIENTRYP PFNGLRASTERPOS3XOESPROC)(GLfixed x, GLfixed y, GLfixed z);
GLAPI PFNGLRASTERPOS3XOESPROC glad_glRasterPos3xOES;
#define glRasterPos3xOES glad_glRasterPos3xOES
typedef void (APIENTRYP PFNGLRASTERPOS3XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLRASTERPOS3XVOESPROC glad_glRasterPos3xvOES;
#define glRasterPos3xvOES glad_glRasterPos3xvOES
typedef void (APIENTRYP PFNGLRASTERPOS4XOESPROC)(GLfixed x, GLfixed y, GLfixed z, GLfixed w);
GLAPI PFNGLRASTERPOS4XOESPROC glad_glRasterPos4xOES;
#define glRasterPos4xOES glad_glRasterPos4xOES
typedef void (APIENTRYP PFNGLRASTERPOS4XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLRASTERPOS4XVOESPROC glad_glRasterPos4xvOES;
#define glRasterPos4xvOES glad_glRasterPos4xvOES
typedef void (APIENTRYP PFNGLRECTXOESPROC)(GLfixed x1, GLfixed y1, GLfixed x2, GLfixed y2);
GLAPI PFNGLRECTXOESPROC glad_glRectxOES;
#define glRectxOES glad_glRectxOES
typedef void (APIENTRYP PFNGLRECTXVOESPROC)(const GLfixed *v1, const GLfixed *v2);
GLAPI PFNGLRECTXVOESPROC glad_glRectxvOES;
#define glRectxvOES glad_glRectxvOES
typedef void (APIENTRYP PFNGLTEXCOORD1XOESPROC)(GLfixed s);
GLAPI PFNGLTEXCOORD1XOESPROC glad_glTexCoord1xOES;
#define glTexCoord1xOES glad_glTexCoord1xOES
typedef void (APIENTRYP PFNGLTEXCOORD1XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLTEXCOORD1XVOESPROC glad_glTexCoord1xvOES;
#define glTexCoord1xvOES glad_glTexCoord1xvOES
typedef void (APIENTRYP PFNGLTEXCOORD2XOESPROC)(GLfixed s, GLfixed t);
GLAPI PFNGLTEXCOORD2XOESPROC glad_glTexCoord2xOES;
#define glTexCoord2xOES glad_glTexCoord2xOES
typedef void (APIENTRYP PFNGLTEXCOORD2XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLTEXCOORD2XVOESPROC glad_glTexCoord2xvOES;
#define glTexCoord2xvOES glad_glTexCoord2xvOES
typedef void (APIENTRYP PFNGLTEXCOORD3XOESPROC)(GLfixed s, GLfixed t, GLfixed r);
GLAPI PFNGLTEXCOORD3XOESPROC glad_glTexCoord3xOES;
#define glTexCoord3xOES glad_glTexCoord3xOES
typedef void (APIENTRYP PFNGLTEXCOORD3XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLTEXCOORD3XVOESPROC glad_glTexCoord3xvOES;
#define glTexCoord3xvOES glad_glTexCoord3xvOES
typedef void (APIENTRYP PFNGLTEXCOORD4XOESPROC)(GLfixed s, GLfixed t, GLfixed r, GLfixed q);
GLAPI PFNGLTEXCOORD4XOESPROC glad_glTexCoord4xOES;
#define glTexCoord4xOES glad_glTexCoord4xOES
typedef void (APIENTRYP PFNGLTEXCOORD4XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLTEXCOORD4XVOESPROC glad_glTexCoord4xvOES;
#define glTexCoord4xvOES glad_glTexCoord4xvOES
typedef void (APIENTRYP PFNGLTEXGENXOESPROC)(GLenum coord, GLenum pname, GLfixed param);
GLAPI PFNGLTEXGENXOESPROC glad_glTexGenxOES;
```

```
#define glTexGenxOES glad_glTexGenxOES
typedef void (APIENTRYP PFNGLTEXGENXVOESPROC)(GLenum coord, GLenum pname, const GLfixed *params);
GLAPI PFNGLTEXGENXVOESPROC glad_glTexGenxvOES;
#define glTexGenxvOES glad_glTexGenxvOES
typedef void (APIENTRYP PFNGLVERTEX2XOESPROC)(GLfixed x);
GLAPI PFNGLVERTEX2XOESPROC glad_glVertex2xOES;
#define glVertex2xOES glad_glVertex2xOES
typedef void (APIENTRYP PFNGLVERTEX2XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLVERTEX2XVOESPROC glad_glVertex2xvOES;
#define glVertex2xvOES glad_glVertex2xvOES
typedef void (APIENTRYP PFNGLVERTEX3XOESPROC)(GLfixed x, GLfixed y);
GLAPI PFNGLVERTEX3XOESPROC glad_glVertex3xOES;
#define glVertex3xOES glad_glVertex3xOES
typedef void (APIENTRYP PFNGLVERTEX3XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLVERTEX3XVOESPROC glad_glVertex3xvOES;
#define glVertex3xvOES glad_glVertex3xvOES
typedef void (APIENTRYP PFNGLVERTEX4XOESPROC)(GLfixed x, GLfixed y, GLfixed z);
GLAPI PFNGLVERTEX4XOESPROC glad_glVertex4xOES;
#define glVertex4xOES glad_glVertex4xOES
typedef void (APIENTRYP PFNGLVERTEX4XVOESPROC)(const GLfixed *coords);
GLAPI PFNGLVERTEX4XVOESPROC glad_glVertex4xvOES;
#define glVertex4xvOES glad_glVertex4xvOES
#endif
#ifndef GL_OES_query_matrix
#define GL_OES_query_matrix 1
GLAPI int GLAD_GL_OES_query_matrix;
typedef GLbitfield (APIENTRYP PFNGLQUERYMATRIXXOESPROC)(GLfixed *mantissa, GLint *exponent);
GLAPI PFNGLQUERYMATRIXXOESPROC glad_glQueryMatrixxOES;
#define glQueryMatrixxOES glad_glQueryMatrixxOES
#endif
#ifndef GL_OES_read_format
#define GL_OES_read_format 1
GLAPI int GLAD_GL_OES_read_format;
#endif
#ifndef GL_OES_single_precision
#define GL_OES_single_precision 1
GLAPI int GLAD_GL_OES_single_precision;
typedef void (APIENTRYP PFNGLCLEARDEPTHFOESPROC)(GLclampf depth);
GLAPI PFNGLCLEARDEPTHFOESPROC glad_glClearDepthfOES;
#define glClearDepthfOES glad_glClearDepthfOES
typedef void (APIENTRYP PFNGLCLIPPLANEFOESPROC)(GLenum plane, const GLfloat *equation);
GLAPI PFNGLCLIPPLANEFOESPROC glad_glClipPlanefOES;
#define glClipPlanefOES glad_glClipPlanefOES
typedef void (APIENTRYP PFNGLDEPTHRANGEFOESPROC)(GLclampf n, GLclampf f);
GLAPI PFNGLDEPTHRANGEFOESPROC glad_glDepthRangefOES;
#define glDepthRangefOES glad_glDepthRangefOES
typedef void (APIENTRYP PFNGLFRUSTUMFOESPROC)(GLfloat l, GLfloat r, GLfloat b, GLfloat t, GLfloat n,
GLfloat f);
GLAPI PFNGLFRUSTUMFOESPROC glad_glFrustumfOES;
#define glFrustumfOES glad_glFrustumfOES
typedef void (APIENTRYP PFNGLGETCLIPPLANEFOESPROC)(GLenum plane, GLfloat *equation);
GLAPI PFNGLGETCLIPPLANEFOESPROC glad_glGetClipPlanefOES;
#define glGetClipPlanefOES glad_glGetClipPlanefOES
typedef void (APIENTRYP PFNGLORTHOFOESPROC)(GLfloat l, GLfloat r, GLfloat b, GLfloat t, GLfloat n,
GLfloat f);
GLAPI PFNGLORTHOFOESPROC glad_glOrthofOES;
#define glOrthofOES glad_glOrthofOES
#endif
#ifndef GL_OML_interlace
#define GL_OML_interlace 1
GLAPI int GLAD_GL_OML_interlace;
#endif
#ifndef GL_OML_resample
#define GL_OML_resample 1
GLAPI int GLAD_GL_OML_resample;
#endif
#ifndef GL_OML_subsample
#define GL_OML_subsample 1
GLAPI int GLAD_GL_OML_subsample;
#endif
#ifndef GL_OVR_multiview
#define GL_OVR_multiview 1
GLAPI int GLAD_GL_OVR_multiview;
typedef void (APIENTRYP PFNGLFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC)(GLenum target, GLenum attachment,
GLuint texture, GLint level, GLint baseViewIndex, GLsizei numViews);
GLAPI PFNGLFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC glad_glFramebufferTextureMultiviewOVR;
#define glFramebufferTextureMultiviewOVR glad_glFramebufferTextureMultiviewOVR
typedef void (APIENTRYP PFNGLNAMEDFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC)(GLuint framebuffer, GLenum
attachment, GLuint texture, GLint level, GLint baseViewIndex, GLsizei numViews);
GLAPI PFNGLNAMEDFRAMEBUFFERTEXTUREMULTIVIEWOVRPROC glad_glNamedFramebufferTextureMultiviewOVR;
#define glNamedFramebufferTextureMultiviewOVR glad_glNamedFramebufferTextureMultiviewOVR
#endif
#ifndef GL_OVR_multiview2
#define GL_OVR_multiview2 1
GLAPI int GLAD_GL_OVR_multiview2;
#endif
#ifndef GL_PGI_misc_hints
```

```
#define GL_PGI_misc_hints 1
GLAPI int GLAD_GL_PGI_misc_hints;
typedef void (APIENTRYP PFNGLHINTPGIPROC)(GLenum target, GLint mode);
GLAPI PFNGLHINTPGIPROC glad_glHintPGI;
#define glHintPGI glad_glHintPGI
#endif
#ifndef GL_PGI_vertex_hints
#define GL_PGI_vertex_hints 1
GLAPI int GLAD_GL_PGI_vertex_hints;
#endif
#ifndef GL_REND_screen_coordinates
#define GL_REND_screen_coordinates 1
GLAPI int GLAD_GL_REND_screen_coordinates;
#endif
#ifndef GL_S3_s3tc
#define GL_S3_s3tc 1
GLAPI int GLAD_GL_S3_s3tc;
#endif
#ifndef GL_SGIS_detail_texture
#define GL_SGIS_detail_texture 1
GLAPI int GLAD_GL_SGIS_detail_texture;
typedef void (APIENTRYP PFNGLDETAILTEXFUNCSGISPROC)(GLenum target, GLsizei n, const GLfloat *points);
GLAPI PFNGLDETAILTEXFUNCSGISPROC glad_glDetailTexFuncSGIS;
#define glDetailTexFuncSGIS glad_glDetailTexFuncSGIS
typedef void (APIENTRYP PFNGLGETDETAILTEXFUNCSGISPROC)(GLenum target, GLfloat *points);
GLAPI PFNGLGETDETAILTEXFUNCSGISPROC glad_glGetDetailTexFuncSGIS;
#define glGetDetailTexFuncSGIS glad_glGetDetailTexFuncSGIS
#endif
#ifndef GL_SGIS_fog_function
#define GL_SGIS_fog_function 1
GLAPI int GLAD_GL_SGIS_fog_function;
typedef void (APIENTRYP PFNGLFOGFUNCSGISPROC)(GLsizei n, const GLfloat *points);
GLAPI PFNGLFOGFUNCSGISPROC glad_glFogFuncSGIS;
#define glFogFuncSGIS glad_glFogFuncSGIS
typedef void (APIENTRYP PFNGLGETFOGFUNCSGISPROC)(GLfloat *points);
GLAPI PFNGLGETFOGFUNCSGISPROC glad_glGetFogFuncSGIS;
#define glGetFogFuncSGIS glad_glGetFogFuncSGIS
#endif
#ifndef GL_SGIS_generate_mipmap
#define GL_SGIS_generate_mipmap 1
GLAPI int GLAD_GL_SGIS_generate_mipmap;
#endif
#ifndef GL_SGIS_multisample
#define GL_SGIS_multisample 1
GLAPI int GLAD_GL_SGIS_multisample;
typedef void (APIENTRYP PFNGLSAMPLEMASKSGISPROC)(GLclampf value, GLboolean invert);
GLAPI PFNGLSAMPLEMASKSGISPROC glad_glSampleMaskSGIS;
#define glSampleMaskSGIS glad_glSampleMaskSGIS
typedef void (APIENTRYP PFNGLSAMPLEPATTERNSGISPROC)(GLenum pattern);
GLAPI PFNGLSAMPLEPATTERNSGISPROC glad_glSamplePatternSGIS;
#define glSamplePatternSGIS glad_glSamplePatternSGIS
#endif
#ifndef GL_SGIS_pixel_texture
#define GL_SGIS_pixel_texture 1
GLAPI int GLAD_GL_SGIS_pixel_texture;
typedef void (APIENTRYP PFNGLPIXELTEXGENPARAMETERISGISPROC)(GLenum pname, GLint param);
GLAPI PFNGLPIXELTEXGENPARAMETERISGISPROC glad_glPixelTexGenParameteriSGIS;
#define glPixelTexGenParameteriSGIS glad_glPixelTexGenParameteriSGIS
typedef void (APIENTRYP PFNGLPIXELTEXGENPARAMETERIVSGISPROC)(GLenum pname, const GLint *params);
GLAPI PFNGLPIXELTEXGENPARAMETERIVSGISPROC glad_glPixelTexGenParameterivSGIS;
#define glPixelTexGenParameterivSGIS glad_glPixelTexGenParameterivSGIS
typedef void (APIENTRYP PFNGLPIXELTEXGENPARAMETERFSGISPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLPIXELTEXGENPARAMETERFSGISPROC glad_glPixelTexGenParameterfSGIS;
#define glPixelTexGenParameterfSGIS glad_glPixelTexGenParameterfSGIS
typedef void (APIENTRYP PFNGLPIXELTEXGENPARAMETERFVSGISPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLPIXELTEXGENPARAMETERFVSGISPROC glad_glPixelTexGenParameterfvSGIS;
#define glPixelTexGenParameterfvSGIS glad_glPixelTexGenParameterfvSGIS
typedef void (APIENTRYP PFNGLGETPIXELTEXGENPARAMETERIVSGISPROC)(GLenum pname, GLint *params);
GLAPI PFNGLGETPIXELTEXGENPARAMETERIVSGISPROC glad_glGetPixelTexGenParameterivSGIS;
#define glGetPixelTexGenParameterivSGIS glad_glGetPixelTexGenParameterivSGIS
typedef void (APIENTRYP PFNGLGETPIXELTEXGENPARAMETERFVSGISPROC)(GLenum pname, GLfloat *params);
GLAPI PFNGLGETPIXELTEXGENPARAMETERFVSGISPROC glad_glGetPixelTexGenParameterfvSGIS;
#define glGetPixelTexGenParameterfvSGIS glad_glGetPixelTexGenParameterfvSGIS
#endif
#ifndef GL_SGIS_point_line_texgen
#define GL_SGIS_point_line_texgen 1
GLAPI int GLAD_GL_SGIS_point_line_texgen;
#endif
#ifndef GL_SGIS_point_parameters
#define GL_SGIS_point_parameters 1
GLAPI int GLAD_GL_SGIS_point_parameters;
typedef void (APIENTRYP PFNGLPOINTPARAMETERFSGISPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLPOINTPARAMETERFSGISPROC glad_glPointParameterfSGIS;
#define glPointParameterfSGIS glad_glPointParameterfSGIS
typedef void (APIENTRYP PFNGLPOINTPARAMETERFVSGISPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLPOINTPARAMETERFVSGISPROC glad_glPointParameterfvSGIS;
#define glPointParameterfvSGIS glad_glPointParameterfvSGIS
```

```
#endif
#ifndef GL_SGIS_sharpen_texture
#define GL_SGIS_sharpen_texture 1
GLAPI int GLAD_GL_SGIS_sharpen_texture;
typedef void (APIENTRYP PFNGLSHARPENTEXFUNCSGISPROC)(GLenum target, GLsizei n, const GLfloat *points);
GLAPI PFNGLSHARPENTEXFUNCSGISPROC glad_glSharpenTexFuncSGIS;
#define glSharpenTexFuncSGIS glad_glSharpenTexFuncSGIS
typedef void (APIENTRYP PFNGLGETSHARPENTEXFUNCSGISPROC)(GLenum target, GLfloat *points);
GLAPI PFNGLGETSHARPENTEXFUNCSGISPROC glad_glGetSharpenTexFuncSGIS;
#define glGetSharpenTexFuncSGIS glad_glGetSharpenTexFuncSGIS
#endif
#ifndef GL_SGIS_texture4D
#define GL_SGIS_texture4D 1
GLAPI int GLAD_GL_SGIS_texture4D;
typedef void (APIENTRYP PFNGLTEXIMAGE4DSGISPROC)(GLenum target, GLint level, GLenum internalformat, GLsizei width, GLsizei height, GLsizei depth, GLsizei size4d, GLint border, GLenum format, GLenum type, const void *pixels);
GLAPI PFNGLTEXIMAGE4DSGISPROC glad_glTexImage4DSGIS;
#define glTexImage4DSGIS glad_glTexImage4DSGIS
typedef void (APIENTRYP PFNGLTEXSUBIMAGE4DSGISPROC)(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);
GLAPI PFNGLTEXSUBIMAGE4DSGISPROC glad_glTexSubImage4DSGIS;
#define glTexSubImage4DSGIS glad_glTexSubImage4DSGIS
#endif
#ifndef GL_SGIS_texture_border_clamp
#define GL_SGIS_texture_border_clamp 1
GLAPI int GLAD_GL_SGIS_texture_border_clamp;
#endif
#ifndef GL_SGIS_texture_color_mask
#define GL_SGIS_texture_color_mask 1
GLAPI int GLAD_GL_SGIS_texture_color_mask;
typedef void (APIENTRYP PFNGLTEXTURECOLORMASKSGISPROC)(GLboolean red, GLboolean green, GLboolean blue, GLboolean alpha);
GLAPI PFNGLTEXTURECOLORMASKSGISPROC glad_glTextureColorMaskSGIS;
#define glTextureColorMaskSGIS glad_glTextureColorMaskSGIS
#endif
#ifndef GL_SGIS_texture_edge_clamp
#define GL_SGIS_texture_edge_clamp 1
GLAPI int GLAD_GL_SGIS_texture_edge_clamp;
#endif
#ifndef GL_SGIS_texture_filter4
#define GL_SGIS_texture_filter4 1
GLAPI int GLAD_GL_SGIS_texture_filter4;
typedef void (APIENTRYP PFNGLGETTEXFILTERFUNCSGISPROC)(GLenum target, GLenum filter, GLfloat *weights);
GLAPI PFNGLGETTEXFILTERFUNCSGISPROC glad_glGetTexFilterFuncSGIS;
#define glGetTexFilterFuncSGIS glad_glGetTexFilterFuncSGIS
typedef void (APIENTRYP PFNGLTEXFILTERFUNCSGISPROC)(GLenum target, GLenum filter, GLsizei n, const GLfloat *weights);
GLAPI PFNGLTEXFILTERFUNCSGISPROC glad_glTexFilterFuncSGIS;
#define glTexFilterFuncSGIS glad_glTexFilterFuncSGIS
#endif
#ifndef GL_SGIS_texture_lod
#define GL_SGIS_texture_lod 1
GLAPI int GLAD_GL_SGIS_texture_lod;
#endif
#ifndef GL_SGIS_texture_select
#define GL_SGIS_texture_select 1
GLAPI int GLAD_GL_SGIS_texture_select;
#endif
#ifndef GL_SGIX_async
#define GL_SGIX_async 1
GLAPI int GLAD_GL_SGIX_async;
typedef void (APIENTRYP PFNGLASYNCMARKERSGIXPROC)(GLuint marker);
GLAPI PFNGLASYNCMARKERSGIXPROC glad_glAsyncMarkerSGIX;
#define glAsyncMarkerSGIX glad_glAsyncMarkerSGIX
typedef GLint (APIENTRYP PFNGLFINISHASYNCSGIXPROC)(GLuint *markerp);
GLAPI PFNGLFINISHASYNCSGIXPROC glad_glFinishAsyncSGIX;
#define glFinishAsyncSGIX glad_glFinishAsyncSGIX
typedef GLint (APIENTRYP PFNGLPOLLASYNCSGIXPROC)(GLuint *markerp);
GLAPI PFNGLPOLLASYNCSGIXPROC glad_glPollAsyncSGIX;
#define glPollAsyncSGIX glad_glPollAsyncSGIX
typedef GLuint (APIENTRYP PFNGLGENASYNCMARKERSSGIXPROC)(GLsizei range);
GLAPI PFNGLGENASYNCMARKERSSGIXPROC glad_glGenAsyncMarkersSGIX;
#define glGenAsyncMarkersSGIX glad_glGenAsyncMarkersSGIX
typedef void (APIENTRYP PFNGLDELETEASYNCMARKERSSGIXPROC)(GLuint marker, GLsizei range);
GLAPI PFNGLDELETEASYNCMARKERSSGIXPROC glad_glDeleteAsyncMarkersSGIX;
#define glDeleteAsyncMarkersSGIX glad_glDeleteAsyncMarkersSGIX
typedef GLboolean (APIENTRYP PFNGLISASYNCMARKERSGIXPROC)(GLuint marker);
GLAPI PFNGLISASYNCMARKERSGIXPROC glad_glIsAsyncMarkerSGIX;
#define glIsAsyncMarkerSGIX glad_glIsAsyncMarkerSGIX
#endif
#ifndef GL_SGIX_async_histogram
#define GL_SGIX_async_histogram 1
GLAPI int GLAD_GL_SGIX_async_histogram;
#endif
```

```
#ifndef GL_SGIX_async_pixel
#define GL_SGIX_async_pixel 1
GLAPI int GLAD_GL_SGIX_async_pixel;
#endif
#ifndef GL_SGIX_blend_alpha_minmax
#define GL_SGIX_blend_alpha_minmax 1
GLAPI int GLAD_GL_SGIX_blend_alpha_minmax;
#endif
#ifndef GL_SGIX_calligraphic_fragment
#define GL_SGIX_calligraphic_fragment 1
GLAPI int GLAD_GL_SGIX_calligraphic_fragment;
#endif
#ifndef GL_SGIX_clipmap
#define GL_SGIX_clipmap 1
GLAPI int GLAD_GL_SGIX_clipmap;
#endif
#ifndef GL_SGIX_convolution_accuracy
#define GL_SGIX_convolution_accuracy 1
GLAPI int GLAD_GL_SGIX_convolution_accuracy;
#endif
#ifndef GL_SGIX_depth_pass_instrument
#define GL_SGIX_depth_pass_instrument 1
GLAPI int GLAD_GL_SGIX_depth_pass_instrument;
#endif
#ifndef GL_SGIX_depth_texture
#define GL_SGIX_depth_texture 1
GLAPI int GLAD_GL_SGIX_depth_texture;
#endif
#ifndef GL_SGIX_flush_raster
#define GL_SGIX_flush_raster 1
GLAPI int GLAD_GL_SGIX_flush_raster;
typedef void (APIENTRYP PFNGLFLUSHRASTERSGIXPROC)(void);
GLAPI PFNGLFLUSHRASTERSGIXPROC glad_glFlushRasterSGIX;
#define glFlushRasterSGIX glad_glFlushRasterSGIX
#endif
#ifndef GL_SGIX_fog_offset
#define GL_SGIX_fog_offset 1
GLAPI int GLAD_GL_SGIX_fog_offset;
#endif
#ifndef GL_SGIX_fragment_lighting
#define GL_SGIX_fragment_lighting 1
GLAPI int GLAD_GL_SGIX_fragment_lighting;
typedef void (APIENTRYP PFNGLFRAGMENTCOLORMATERIALSGIXPROC)(GLenum face, GLenum mode);
GLAPI PFNGLFRAGMENTCOLORMATERIALSGIXPROC glad_glFragmentColorMaterialSGIX;
#define glFragmentColorMaterialSGIX glad_glFragmentColorMaterialSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTFSGIXPROC)(GLenum light, GLenum pname, GLfloat param);
GLAPI PFNGLFRAGMENTLIGHTFSGIXPROC glad_glFragmentLightfSGIX;
#define glFragmentLightfSGIX glad_glFragmentLightfSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTFVSGIXPROC)(GLenum light, GLenum pname, const GLfloat *params);
GLAPI PFNGLFRAGMENTLIGHTFVSGIXPROC glad_glFragmentLightfvSGIX;
#define glFragmentLightfvSGIX glad_glFragmentLightfvSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTISGIXPROC)(GLenum light, GLenum pname, GLint param);
GLAPI PFNGLFRAGMENTLIGHTISGIXPROC glad_glFragmentLightiSGIX;
#define glFragmentLightiSGIX glad_glFragmentLightiSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTIVSGIXPROC)(GLenum light, GLenum pname, const GLint *params);
GLAPI PFNGLFRAGMENTLIGHTIVSGIXPROC glad_glFragmentLightivSGIX;
#define glFragmentLightivSGIX glad_glFragmentLightivSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTMODELFSGIXPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLFRAGMENTLIGHTMODELFSGIXPROC glad_glFragmentLightModelfSGIX;
#define glFragmentLightModelfSGIX glad_glFragmentLightModelfSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTMODELFVSGIXPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLFRAGMENTLIGHTMODELFVSGIXPROC glad_glFragmentLightModelfvSGIX;
#define glFragmentLightModelfvSGIX glad_glFragmentLightModelfvSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTMODELISGIXPROC)(GLenum pname, GLint param);
GLAPI PFNGLFRAGMENTLIGHTMODELISGIXPROC glad_glFragmentLightModeliSGIX;
#define glFragmentLightModeliSGIX glad_glFragmentLightModeliSGIX
typedef void (APIENTRYP PFNGLFRAGMENTLIGHTMODELIVSGIXPROC)(GLenum pname, const GLint *params);
GLAPI PFNGLFRAGMENTLIGHTMODELIVSGIXPROC glad_glFragmentLightModelivSGIX;
#define glFragmentLightModelivSGIX glad_glFragmentLightModelivSGIX
typedef void (APIENTRYP PFNGLFRAGMENTMATERIALFSGIXPROC)(GLenum face, GLenum pname, GLfloat param);
GLAPI PFNGLFRAGMENTMATERIALFSGIXPROC glad_glFragmentMaterialfSGIX;
#define glFragmentMaterialfSGIX glad_glFragmentMaterialfSGIX
typedef void (APIENTRYP PFNGLFRAGMENTMATERIALFVSGIXPROC)(GLenum face, GLenum pname, const GLfloat *params);
GLAPI PFNGLFRAGMENTMATERIALFVSGIXPROC glad_glFragmentMaterialfvSGIX;
#define glFragmentMaterialfvSGIX glad_glFragmentMaterialfvSGIX
typedef void (APIENTRYP PFNGLFRAGMENTMATERIALISGIXPROC)(GLenum face, GLenum pname, GLint param);
GLAPI PFNGLFRAGMENTMATERIALISGIXPROC glad_glFragmentMaterialiSGIX;
#define glFragmentMaterialiSGIX glad_glFragmentMaterialiSGIX
typedef void (APIENTRYP PFNGLFRAGMENTMATERIALIVSGIXPROC)(GLenum face, GLenum pname, const GLint *params);
GLAPI PFNGLFRAGMENTMATERIALIVSGIXPROC glad_glFragmentMaterialivSGIX;
#define glFragmentMaterialivSGIX glad_glFragmentMaterialivSGIX
typedef void (APIENTRYP PFNGLGETFRAGMENTLIGHTFVSGIXPROC)(GLenum light, GLenum pname, GLfloat *params);
GLAPI PFNGLGETFRAGMENTLIGHTFVSGIXPROC glad_glGetFragmentLightfvSGIX;
```

```
#define glGetFragmentLightfvSGIX glad_glGetFragmentLightfvSGIX
typedef void (APIENTRYP PFNGLGETFRAGMENTLIGHTIVSGIXPROC)(GLenum light, GLenum pname, GLint *params);
GLAPI PFNGLGETFRAGMENTLIGHTIVSGIXPROC glad_glGetFragmentLightivSGIX;
#define glGetFragmentLightivSGIX glad_glGetFragmentLightivSGIX
typedef void (APIENTRYP PFNGLGETFRAGMENTMATERIALFVSGIXPROC)(GLenum face, GLenum pname, GLfloat *params);
GLAPI PFNGLGETFRAGMENTMATERIALFVSGIXPROC glad_glGetFragmentMaterialfvSGIX;
#define glGetFragmentMaterialfvSGIX glad_glGetFragmentMaterialfvSGIX
typedef void (APIENTRYP PFNGLGETFRAGMENTMATERIALIVSGIXPROC)(GLenum face, GLenum pname, GLint *params);
GLAPI PFNGLGETFRAGMENTMATERIALIVSGIXPROC glad_glGetFragmentMaterialivSGIX;
#define glGetFragmentMaterialivSGIX glad_glGetFragmentMaterialivSGIX
typedef void (APIENTRYP PFNGLLIGHTENVISGIXPROC)(GLenum pname, GLint param);
GLAPI PFNGLLIGHTENVISGIXPROC glad_glLightEnviSGIX;
#define glLightEnviSGIX glad_glLightEnviSGIX
#endif
#ifndef GL_SGIX_framezoom
#define GL_SGIX_framezoom 1
GLAPI int GLAD_GL_SGIX_framezoom;
typedef void (APIENTRYP PFNGLFRAMEZOOMSGIXPROC)(GLint factor);
GLAPI PFNGLFRAMEZOOMSGIXPROC glad_glFrameZoomSGIX;
#define glFrameZoomSGIX glad_glFrameZoomSGIX
#endif
#ifndef GL_SGIX_igloo_interface
#define GL_SGIX_igloo_interface 1
GLAPI int GLAD_GL_SGIX_igloo_interface;
typedef void (APIENTRYP PFNGLIGLOOINTERFACESGIXPROC)(GLenum pname, const void *params);
GLAPI PFNGLIGLOOINTERFACESGIXPROC glad_glIglooInterfaceSGIX;
#define glIglooInterfaceSGIX glad_glIglooInterfaceSGIX
#endif
#ifndef GL_SGIX_instruments
#define GL_SGIX_instruments 1
GLAPI int GLAD_GL_SGIX_instruments;
typedef GLint (APIENTRYP PFNGLGETINSTRUMENTSSGIXPROC)(void);
GLAPI PFNGLGETINSTRUMENTSSGIXPROC glad_glGetInstrumentsSGIX;
#define glGetInstrumentsSGIX glad_glGetInstrumentsSGIX
typedef void (APIENTRYP PFNGLINSTRUMENTSBUFFERSGIXPROC)(GLsizei size, GLint *buffer);
GLAPI PFNGLINSTRUMENTSBUFFERSGIXPROC glad_glInstrumentsBufferSGIX;
#define glInstrumentsBufferSGIX glad_glInstrumentsBufferSGIX
typedef GLint (APIENTRYP PFNGLPOLLINSTRUMENTSSGIXPROC)(GLint *marker_p);
GLAPI PFNGLPOLLINSTRUMENTSSGIXPROC glad_glPollInstrumentsSGIX;
#define glPollInstrumentsSGIX glad_glPollInstrumentsSGIX
typedef void (APIENTRYP PFNGLREADINSTRUMENTSSGIXPROC)(GLint marker);
GLAPI PFNGLREADINSTRUMENTSSGIXPROC glad_glReadInstrumentsSGIX;
#define glReadInstrumentsSGIX glad_glReadInstrumentsSGIX
typedef void (APIENTRYP PFNGLSTARTINSTRUMENTSSGIXPROC)(void);
GLAPI PFNGLSTARTINSTRUMENTSSGIXPROC glad_glStartInstrumentsSGIX;
#define glStartInstrumentsSGIX glad_glStartInstrumentsSGIX
typedef void (APIENTRYP PFNGLSTOPINSTRUMENTSSGIXPROC)(GLint marker);
GLAPI PFNGLSTOPINSTRUMENTSSGIXPROC glad_glStopInstrumentsSGIX;
#define glStopInstrumentsSGIX glad_glStopInstrumentsSGIX
#endif
#ifndef GL_SGIX_interlace
#define GL_SGIX_interlace 1
GLAPI int GLAD_GL_SGIX_interlace;
#endif
#ifndef GL_SGIX_ir_instrument1
#define GL_SGIX_ir_instrument1 1
GLAPI int GLAD_GL_SGIX_ir_instrument1;
#endif
#ifndef GL_SGIX_list_priority
#define GL_SGIX_list_priority 1
GLAPI int GLAD_GL_SGIX_list_priority;
typedef void (APIENTRYP PFNGLGETLISTPARAMETERFVSGIXPROC)(GLuint list, GLenum pname, GLfloat *params);
GLAPI PFNGLGETLISTPARAMETERFVSGIXPROC glad_glGetListParameterfvSGIX;
#define glGetListParameterfvSGIX glad_glGetListParameterfvSGIX
typedef void (APIENTRYP PFNGLGETLISTPARAMETERIVSGIXPROC)(GLuint list, GLenum pname, GLint *params);
GLAPI PFNGLGETLISTPARAMETERIVSGIXPROC glad_glGetListParameterivSGIX;
#define glGetListParameterivSGIX glad_glGetListParameterivSGIX
typedef void (APIENTRYP PFNGLLISTPARAMETERFSGIXPROC)(GLuint list, GLenum pname, GLfloat param);
GLAPI PFNGLLISTPARAMETERFSGIXPROC glad_glListParameterfSGIX;
#define glListParameterfSGIX glad_glListParameterfSGIX
typedef void (APIENTRYP PFNGLLISTPARAMETERFVSGIXPROC)(GLuint list, GLenum pname, const GLfloat *params);
GLAPI PFNGLLISTPARAMETERFVSGIXPROC glad_glListParameterfvSGIX;
#define glListParameterfvSGIX glad_glListParameterfvSGIX
typedef void (APIENTRYP PFNGLLISTPARAMETERISGIXPROC)(GLuint list, GLenum pname, GLint param);
GLAPI PFNGLLISTPARAMETERISGIXPROC glad_glListParameteriSGIX;
#define glListParameteriSGIX glad_glListParameteriSGIX
typedef void (APIENTRYP PFNGLLISTPARAMETERIVSGIXPROC)(GLuint list, GLenum pname, const GLint *params);
GLAPI PFNGLLISTPARAMETERIVSGIXPROC glad_glListParameterivSGIX;
#define glListParameterivSGIX glad_glListParameterivSGIX
#endif
#ifndef GL_SGIX_pixel_texture
#define GL_SGIX_pixel_texture 1
GLAPI int GLAD_GL_SGIX_pixel_texture;
typedef void (APIENTRYP PFNGLPIXELTEXGENSGIXPROC)(GLenum mode);
GLAPI PFNGLPIXELTEXGENSGIXPROC glad_glPixelTexGenSGIX;
```

```
#define glPixelTexGenSGIX glad_glPixelTexGenSGIX
#endif
#ifndef GL_SGIX_pixel_tiles
#define GL_SGIX_pixel_tiles 1
GLAPI int GLAD_GL_SGIX_pixel_tiles;
#endif
#ifndef GL_SGIX_polynomial_ffd
#define GL_SGIX_polynomial_ffd 1
GLAPI int GLAD_GL_SGIX_polynomial_ffd;
typedef void (APIENTRYP 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);
GLAPI PFNGLDEFORMATIONMAP3DSGIXPROC glad_glDeformationMap3dSGIX;
#define glDeformationMap3dSGIX glad_glDeformationMap3dSGIX
typedef void (APIENTRYP 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);
GLAPI PFNGLDEFORMATIONMAP3FSGIXPROC glad_glDeformationMap3fSGIX;
#define glDeformationMap3fSGIX glad_glDeformationMap3fSGIX
typedef void (APIENTRYP PFNGLDEFORMSGIXPROC)(GLbitfield mask);
GLAPI PFNGLDEFORMSGIXPROC glad_glDeformSGIX;
#define glDeformSGIX glad_glDeformSGIX
typedef void (APIENTRYP PFNGLLOADIDENTITYDEFORMATIONMAPSGIXPROC)(GLbitfield mask);
GLAPI PFNGLLOADIDENTITYDEFORMATIONMAPSGIXPROC glad_glLoadIdentityDeformationMapSGIX;
#define glLoadIdentityDeformationMapSGIX glad_glLoadIdentityDeformationMapSGIX
#endif
#ifndef GL_SGIX_reference_plane
#define GL_SGIX_reference_plane 1
GLAPI int GLAD_GL_SGIX_reference_plane;
typedef void (APIENTRYP PFNGLREFERENCEPLANESGIXPROC)(const GLdouble *equation);
GLAPI PFNGLREFERENCEPLANESGIXPROC glad_glReferencePlaneSGIX;
#define glReferencePlaneSGIX glad_glReferencePlaneSGIX
#endif
#ifndef GL_SGIX_resample
#define GL_SGIX_resample 1
GLAPI int GLAD_GL_SGIX_resample;
#endif
#ifndef GL_SGIX_scalebias_hint
#define GL_SGIX_scalebias_hint 1
GLAPI int GLAD_GL_SGIX_scalebias_hint;
#endif
#ifndef GL_SGIX_shadow
#define GL_SGIX_shadow 1
GLAPI int GLAD_GL_SGIX_shadow;
#endif
#ifndef GL_SGIX_shadow_ambient
#define GL_SGIX_shadow_ambient 1
GLAPI int GLAD_GL_SGIX_shadow_ambient;
#endif
#ifndef GL_SGIX_sprite
#define GL_SGIX_sprite 1
GLAPI int GLAD_GL_SGIX_sprite;
typedef void (APIENTRYP PFNGLSPRITEPARAMETERFSGIXPROC)(GLenum pname, GLfloat param);
GLAPI PFNGLSPRITEPARAMETERFSGIXPROC glad_glSpriteParameterfSGIX;
#define glSpriteParameterfSGIX glad_glSpriteParameterfSGIX
typedef void (APIENTRYP PFNGLSPRITEPARAMETERFVSGIXPROC)(GLenum pname, const GLfloat *params);
GLAPI PFNGLSPRITEPARAMETERFVSGIXPROC glad_glSpriteParameterfvSGIX;
#define glSpriteParameterfvSGIX glad_glSpriteParameterfvSGIX
typedef void (APIENTRYP PFNGLSPRITEPARAMETERISGIXPROC)(GLenum pname, GLint param);
GLAPI PFNGLSPRITEPARAMETERISGIXPROC glad_glSpriteParameteriSGIX;
#define glSpriteParameteriSGIX glad_glSpriteParameteriSGIX
typedef void (APIENTRYP PFNGLSPRITEPARAMETERIVSGIXPROC)(GLenum pname, const GLint *params);
GLAPI PFNGLSPRITEPARAMETERIVSGIXPROC glad_glSpriteParameterivSGIX;
#define glSpriteParameterivSGIX glad_glSpriteParameterivSGIX
#endif
#ifndef GL_SGIX_subsample
#define GL_SGIX_subsample 1
GLAPI int GLAD_GL_SGIX_subsample;
#endif
#ifndef GL_SGIX_tag_sample_buffer
#define GL_SGIX_tag_sample_buffer 1
GLAPI int GLAD_GL_SGIX_tag_sample_buffer;
typedef void (APIENTRYP PFNGLTAGSAMPLEBUFFERSGIXPROC)(void);
GLAPI PFNGLTAGSAMPLEBUFFERSGIXPROC glad_glTagSampleBufferSGIX;
#define glTagSampleBufferSGIX glad_glTagSampleBufferSGIX
#endif
#ifndef GL_SGIX_texture_add_env
#define GL_SGIX_texture_add_env 1
GLAPI int GLAD_GL_SGIX_texture_add_env;
#endif
#ifndef GL_SGIX_texture_coordinate_clamp
#define GL_SGIX_texture_coordinate_clamp 1
GLAPI int GLAD_GL_SGIX_texture_coordinate_clamp;
#endif
#ifndef GL_SGIX_texture_lod_bias
#define GL_SGIX_texture_lod_bias 1
GLAPI int GLAD_GL_SGIX_texture_lod_bias;
```

```
#endif
#ifndef GL_SGIX_texture_multi_buffer
#define GL_SGIX_texture_multi_buffer 1
GLAPI int GLAD_GL_SGIX_texture_multi_buffer;
#endif
#ifndef GL_SGIX_texture_scale_bias
#define GL_SGIX_texture_scale_bias 1
GLAPI int GLAD_GL_SGIX_texture_scale_bias;
#endif
#ifndef GL_SGIX_vertex_preclip
#define GL_SGIX_vertex_preclip 1
GLAPI int GLAD_GL_SGIX_vertex_preclip;
#endif
#ifndef GL_SGIX_ycrcb
#define GL_SGIX_ycrcb 1
GLAPI int GLAD_GL_SGIX_ycrcb;
#endif
#ifndef GL_SGIX_ycrcb_subsample
#define GL_SGIX_ycrcb_subsample 1
GLAPI int GLAD_GL_SGIX_ycrcb_subsample;
#endif
#ifndef GL_SGIX_ycrcba
#define GL_SGIX_ycrcba 1
GLAPI int GLAD_GL_SGIX_ycrcba;
#endif
#ifndef GL_SGI_color_matrix
#define GL_SGI_color_matrix 1
GLAPI int GLAD_GL_SGI_color_matrix;
#endif
#ifndef GL_SGI_color_table
#define GL_SGI_color_table 1
GLAPI int GLAD_GL_SGI_color_table;
typedef void (APIENTRYP PFNGLCOLORTABLESGIPROC)(GLenum target, GLenum internalformat, GLsizei width, GLenum format, GLenum type, const void *table);
GLAPI PFNGLCOLORTABLESGIPROC glad_glColorTableSGI;
#define glColorTableSGI glad_glColorTableSGI
typedef void (APIENTRYP PFNGLCOLORTABLEPARAMETERFVSGIPROC)(GLenum target, GLenum pname, const GLfloat *params);
GLAPI PFNGLCOLORTABLEPARAMETERFVSGIPROC glad_glColorTableParameterfvSGI;
#define glColorTableParameterfvSGI glad_glColorTableParameterfvSGI
typedef void (APIENTRYP PFNGLCOLORTABLEPARAMETERIVSGIPROC)(GLenum target, GLenum pname, const GLint *params);
GLAPI PFNGLCOLORTABLEPARAMETERIVSGIPROC glad_glColorTableParameterivSGI;
#define glColorTableParameterivSGI glad_glColorTableParameterivSGI
typedef void (APIENTRYP PFNGLCOPYCOLORTABLESGIPROC)(GLenum target, GLenum internalformat, GLint x, GLint y, GLsizei width);
GLAPI PFNGLCOPYCOLORTABLESGIPROC glad_glCopyColorTableSGI;
#define glCopyColorTableSGI glad_glCopyColorTableSGI
typedef void (APIENTRYP PFNGLGETCOLORTABLESGIPROC)(GLenum target, GLenum format, GLenum type, void *table);
GLAPI PFNGLGETCOLORTABLESGIPROC glad_glGetColorTableSGI;
#define glGetColorTableSGI glad_glGetColorTableSGI
typedef void (APIENTRYP PFNGLGETCOLORTABLEPARAMETERFVSGIPROC)(GLenum target, GLenum pname, GLfloat *params);
GLAPI PFNGLGETCOLORTABLEPARAMETERFVSGIPROC glad_glGetColorTableParameterfvSGI;
#define glGetColorTableParameterfvSGI glad_glGetColorTableParameterfvSGI
typedef void (APIENTRYP PFNGLGETCOLORTABLEPARAMETERIVSGIPROC)(GLenum target, GLenum pname, GLint *params);
GLAPI PFNGLGETCOLORTABLEPARAMETERIVSGIPROC glad_glGetColorTableParameterivSGI;
#define glGetColorTableParameterivSGI glad_glGetColorTableParameterivSGI
#endif
#ifndef GL_SGI_texture_color_table
#define GL_SGI_texture_color_table 1
GLAPI int GLAD_GL_SGI_texture_color_table;
#endif
#ifndef GL_SUNX_constant_data
#define GL_SUNX_constant_data 1
GLAPI int GLAD_GL_SUNX_constant_data;
typedef void (APIENTRYP PFNGLFINISHTEXTURESUNXPROC)(void);
GLAPI PFNGLFINISHTEXTURESUNXPROC glad_glFinishTextureSUNX;
#define glFinishTextureSUNX glad_glFinishTextureSUNX
#endif
#ifndef GL_SUN_convolution_border_modes
#define GL_SUN_convolution_border_modes 1
GLAPI int GLAD_GL_SUN_convolution_border_modes;
#endif
#ifndef GL_SUN_global_alpha
#define GL_SUN_global_alpha 1
GLAPI int GLAD_GL_SUN_global_alpha;
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORBSUNPROC)(GLbyte factor);
GLAPI PFNGLGLOBALALPHAFACTORBSUNPROC glad_glGlobalAlphaFactorbSUN;
#define glGlobalAlphaFactorbSUN glad_glGlobalAlphaFactorbSUN
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORSSUNPROC)(GLshort factor);
GLAPI PFNGLGLOBALALPHAFACTORSSUNPROC glad_glGlobalAlphaFactorsSUN;
#define glGlobalAlphaFactorsSUN glad_glGlobalAlphaFactorsSUN
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORISUNPROC)(GLint factor);
GLAPI PFNGLGLOBALALPHAFACTORISUNPROC glad_glGlobalAlphaFactoriSUN;
```

```
#define glGlobalAlphaFactoriSUN glad_glGlobalAlphaFactoriSUN
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORFSUNPROC)(GLfloat factor);
GLAPI PFNGLGLOBALALPHAFACTORFSUNPROC glad_glGlobalAlphaFactorfSUN;
#define glGlobalAlphaFactorfSUN glad_glGlobalAlphaFactorfSUN
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORDSUNPROC)(GLdouble factor);
GLAPI PFNGLGLOBALALPHAFACTORDSUNPROC glad_glGlobalAlphaFactordSUN;
#define glGlobalAlphaFactordSUN glad_glGlobalAlphaFactordSUN
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORUBSUNPROC)(GLubyte factor);
GLAPI PFNGLGLOBALALPHAFACTORUBSUNPROC glad_glGlobalAlphaFactorubSUN;
#define glGlobalAlphaFactorubSUN glad_glGlobalAlphaFactorubSUN
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORUSSUNPROC)(GLushort factor);
GLAPI PFNGLGLOBALALPHAFACTORUSSUNPROC glad_glGlobalAlphaFactorusSUN;
#define glGlobalAlphaFactorusSUN glad_glGlobalAlphaFactorusSUN
typedef void (APIENTRYP PFNGLGLOBALALPHAFACTORUISUNPROC)(GLuint factor);
GLAPI PFNGLGLOBALALPHAFACTORUISUNPROC glad_glGlobalAlphaFactoruiSUN;
#define glGlobalAlphaFactoruiSUN glad_glGlobalAlphaFactoruiSUN
#endif
#ifndef GL_SUN_mesh_array
#define GL_SUN_mesh_array 1
GLAPI int GLAD_GL_SUN_mesh_array;
typedef void (APIENTRYP PFNGLDRAWMESHARRAYSSUNPROC)(GLenum mode, GLint first, GLsizei count, GLsizei width);
GLAPI PFNGLDRAWMESHARRAYSSUNPROC glad_glDrawMeshArraysSUN;
#define glDrawMeshArraysSUN glad_glDrawMeshArraysSUN
#endif
#ifndef GL_SUN_slice_accum
#define GL_SUN_slice_accum 1
GLAPI int GLAD_GL_SUN_slice_accum;
#endif
#ifndef GL_SUN_triangle_list
#define GL_SUN_triangle_list 1
GLAPI int GLAD_GL_SUN_triangle_list;
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUISUNPROC)(GLuint code);
GLAPI PFNGLREPLACEMENTCODEUISUNPROC glad_glReplacementCodeuiSUN;
#define glReplacementCodeuiSUN glad_glReplacementCodeuiSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUSSUNPROC)(GLushort code);
GLAPI PFNGLREPLACEMENTCODEUSSUNPROC glad_glReplacementCodeusSUN;
#define glReplacementCodeusSUN glad_glReplacementCodeusSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUBSUNPROC)(GLubyte code);
GLAPI PFNGLREPLACEMENTCODEUBSUNPROC glad_glReplacementCodeubSUN;
#define glReplacementCodeubSUN glad_glReplacementCodeubSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUIVSUNPROC)(const GLuint *code);
GLAPI PFNGLREPLACEMENTCODEUIVSUNPROC glad_glReplacementCodeuivSUN;
#define glReplacementCodeuivSUN glad_glReplacementCodeuivSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUSVSUNPROC)(const GLushort *code);
GLAPI PFNGLREPLACEMENTCODEUSVSUNPROC glad_glReplacementCodeusvSUN;
#define glReplacementCodeusvSUN glad_glReplacementCodeusvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUBVSUNPROC)(const GLubyte *code);
GLAPI PFNGLREPLACEMENTCODEUBVSUNPROC glad_glReplacementCodeubvSUN;
#define glReplacementCodeubvSUN glad_glReplacementCodeubvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEPOINTERSUNPROC)(GLenum type, GLsizei stride, const void **pointer);
GLAPI PFNGLREPLACEMENTCODEPOINTERSUNPROC glad_glReplacementCodePointerSUN;
#define glReplacementCodePointerSUN glad_glReplacementCodePointerSUN
#endif
#ifndef GL_SUN_vertex
#define GL_SUN_vertex 1
GLAPI int GLAD_GL_SUN_vertex;
typedef void (APIENTRYP PFNGLCOLOR4UBVERTEX2FSUNPROC)(GLubyte r, GLubyte g, GLubyte b, GLubyte a, GLfloat x, GLfloat y);
GLAPI PFNGLCOLOR4UBVERTEX2FSUNPROC glad_glColor4ubVertex2fSUN;
#define glColor4ubVertex2fSUN glad_glColor4ubVertex2fSUN
typedef void (APIENTRYP PFNGLCOLOR4UBVERTEX2FVSUNPROC)(const GLubyte *c, const GLfloat *v);
GLAPI PFNGLCOLOR4UBVERTEX2FVSUNPROC glad_glColor4ubVertex2fvSUN;
#define glColor4ubVertex2fvSUN glad_glColor4ubVertex2fvSUN
typedef void (APIENTRYP PFNGLCOLOR4UBVERTEX3FSUNPROC)(GLubyte r, GLubyte g, GLubyte b, GLubyte a, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLCOLOR4UBVERTEX3FSUNPROC glad_glColor4ubVertex3fSUN;
#define glColor4ubVertex3fSUN glad_glColor4ubVertex3fSUN
typedef void (APIENTRYP PFNGLCOLOR4UBVERTEX3FVSUNPROC)(const GLubyte *c, const GLfloat *v);
GLAPI PFNGLCOLOR4UBVERTEX3FVSUNPROC glad_glColor4ubVertex3fvSUN;
#define glColor4ubVertex3fvSUN glad_glColor4ubVertex3fvSUN
typedef void (APIENTRYP PFNGLCOLOR3FVERTEX3FSUNPROC)(GLfloat r, GLfloat g, GLfloat b, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLCOLOR3FVERTEX3FSUNPROC glad_glColor3fVertex3fSUN;
#define glColor3fVertex3fSUN glad_glColor3fVertex3fSUN
typedef void (APIENTRYP PFNGLCOLOR3FVERTEX3FVSUNPROC)(const GLfloat *c, const GLfloat *v);
GLAPI PFNGLCOLOR3FVERTEX3FVSUNPROC glad_glColor3fVertex3fvSUN;
#define glColor3fVertex3fvSUN glad_glColor3fVertex3fvSUN
typedef void (APIENTRYP PFNGLNORMAL3FVERTEX3FSUNPROC)(GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLNORMAL3FVERTEX3FSUNPROC glad_glNormal3fVertex3fSUN;
#define glNormal3fVertex3fSUN glad_glNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLNORMAL3FVERTEX3FVSUNPROC)(const GLfloat *n, const GLfloat *v);
GLAPI PFNGLNORMAL3FVERTEX3FVSUNPROC glad_glNormal3fVertex3fvSUN;
#define glNormal3fVertex3fvSUN glad_glNormal3fVertex3fvSUN
typedef void (APIENTRYP PFNGLCOLOR4FNORMAL3FVERTEX3FSUNPROC)(GLfloat r, GLfloat g, GLfloat b, GLfloat
a, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLCOLOR4FNORMAL3FVERTEX3FSUNPROC glad_glColor4fNormal3fVertex3fSUN;
#define glColor4fNormal3fVertex3fSUN glad_glColor4fNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLCOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLfloat *c, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLCOLOR4FNORMAL3FVERTEX3FVSUNPROC glad_glColor4fNormal3fVertex3fvSUN;
#define glColor4fNormal3fVertex3fvSUN glad_glColor4fNormal3fVertex3fvSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLTEXCOORD2FVERTEX3FSUNPROC glad_glTexCoord2fVertex3fSUN;
#define glTexCoord2fVertex3fSUN glad_glTexCoord2fVertex3fSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FVERTEX3FVSUNPROC)(const GLfloat *tc, const GLfloat *v);
GLAPI PFNGLTEXCOORD2FVERTEX3FVSUNPROC glad_glTexCoord2fVertex3fvSUN;
#define glTexCoord2fVertex3fvSUN glad_glTexCoord2fVertex3fvSUN
typedef void (APIENTRYP PFNGLTEXCOORD4FVERTEX4FSUNPROC)(GLfloat s, GLfloat t, GLfloat p, GLfloat q, GLfloat x, GLfloat y, GLfloat z, GLfloat w);
GLAPI PFNGLTEXCOORD4FVERTEX4FSUNPROC glad_glTexCoord4fVertex4fSUN;
#define glTexCoord4fVertex4fSUN glad_glTexCoord4fVertex4fSUN
typedef void (APIENTRYP PFNGLTEXCOORD4FVERTEX4FVSUNPROC)(const GLfloat *tc, const GLfloat *v);
GLAPI PFNGLTEXCOORD4FVERTEX4FVSUNPROC glad_glTexCoord4fVertex4fvSUN;
#define glTexCoord4fVertex4fvSUN glad_glTexCoord4fVertex4fvSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FCOLOR4UBVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLubyte r, GLubyte g, GLubyte b, GLubyte a, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLTEXCOORD2FCOLOR4UBVERTEX3FSUNPROC glad_glTexCoord2fColor4ubVertex3fSUN;
#define glTexCoord2fColor4ubVertex3fSUN glad_glTexCoord2fColor4ubVertex3fSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FCOLOR4UBVERTEX3FVSUNPROC)(const GLfloat *tc, const GLubyte *c, const GLfloat *v);
GLAPI PFNGLTEXCOORD2FCOLOR4UBVERTEX3FVSUNPROC glad_glTexCoord2fColor4ubVertex3fvSUN;
#define glTexCoord2fColor4ubVertex3fvSUN glad_glTexCoord2fColor4ubVertex3fvSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FCOLOR3FVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLfloat r, GLfloat g, GLfloat b, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLTEXCOORD2FCOLOR3FVERTEX3FSUNPROC glad_glTexCoord2fColor3fVertex3fSUN;
#define glTexCoord2fColor3fVertex3fSUN glad_glTexCoord2fColor3fVertex3fSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FCOLOR3FVERTEX3FVSUNPROC)(const GLfloat *tc, const GLfloat *c, const GLfloat *v);
GLAPI PFNGLTEXCOORD2FCOLOR3FVERTEX3FVSUNPROC glad_glTexCoord2fColor3fVertex3fvSUN;
#define glTexCoord2fColor3fVertex3fvSUN glad_glTexCoord2fColor3fVertex3fvSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FNORMAL3FVERTEX3FSUNPROC)(GLfloat s, GLfloat t, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLTEXCOORD2FNORMAL3FVERTEX3FSUNPROC glad_glTexCoord2fNormal3fVertex3fSUN;
#define glTexCoord2fNormal3fVertex3fSUN glad_glTexCoord2fNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FNORMAL3FVERTEX3FVSUNPROC)(const GLfloat *tc, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLTEXCOORD2FNORMAL3FVERTEX3FVSUNPROC glad_glTexCoord2fNormal3fVertex3fvSUN;
#define glTexCoord2fNormal3fVertex3fvSUN glad_glTexCoord2fNormal3fVertex3fvSUN
typedef void (APIENTRYP 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);
GLAPI PFNGLTEXCOORD2FCOLOR4FNORMAL3FVERTEX3FSUNPROC glad_glTexCoord2fColor4fNormal3fVertex3fSUN;
#define glTexCoord2fColor4fNormal3fVertex3fSUN glad_glTexCoord2fColor4fNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLTEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLfloat *tc, const GLfloat *c, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLTEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC glad_glTexCoord2fColor4fNormal3fVertex3fvSUN;
#define glTexCoord2fColor4fNormal3fVertex3fvSUN glad_glTexCoord2fColor4fNormal3fVertex3fvSUN
typedef void (APIENTRYP 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);
GLAPI PFNGLTEXCOORD4FCOLOR4FNORMAL3FVERTEX4FSUNPROC glad_glTexCoord4fColor4fNormal3fVertex4fSUN;
#define glTexCoord4fColor4fNormal3fVertex4fSUN glad_glTexCoord4fColor4fNormal3fVertex4fSUN
typedef void (APIENTRYP PFNGLTEXCOORD4FCOLOR4FNORMAL3FVERTEX4FVSUNPROC)(const GLfloat *tc, const GLfloat *c, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLTEXCOORD4FCOLOR4FNORMAL3FVERTEX4FVSUNPROC glad_glTexCoord4fColor4fNormal3fVertex4fvSUN;
#define glTexCoord4fColor4fNormal3fVertex4fvSUN glad_glTexCoord4fColor4fNormal3fVertex4fvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUIVERTEX3FSUNPROC)(GLuint rc, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLREPLACEMENTCODEUIVERTEX3FSUNPROC glad_glReplacementCodeuiVertex3fSUN;
#define glReplacementCodeuiVertex3fSUN glad_glReplacementCodeuiVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUIVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUIVERTEX3FVSUNPROC glad_glReplacementCodeuiVertex3fvSUN;
#define glReplacementCodeuiVertex3fvSUN glad_glReplacementCodeuiVertex3fvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FSUNPROC)(GLuint rc, GLubyte r, GLubyte g, GLubyte b, GLubyte a, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FSUNPROC glad_glReplacementCodeuiColor4ubVertex3fSUN;
#define glReplacementCodeuiColor4ubVertex3fSUN glad_glReplacementCodeuiColor4ubVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FVSUNPROC)(const GLuint *rc, const GLubyte *c, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUICOLOR4UBVERTEX3FVSUNPROC glad_glReplacementCodeuiColor4ubVertex3fvSUN;
#define glReplacementCodeuiColor4ubVertex3fvSUN glad_glReplacementCodeuiColor4ubVertex3fvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FSUNPROC)(GLuint rc, GLfloat r, GLfloat g, GLfloat b, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FSUNPROC glad_glReplacementCodeuiColor3fVertex3fSUN;
#define glReplacementCodeuiColor3fVertex3fSUN glad_glReplacementCodeuiColor3fVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *c, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUICOLOR3FVERTEX3FVSUNPROC glad_glReplacementCodeuiColor3fVertex3fvSUN;
#define glReplacementCodeuiColor3fVertex3fvSUN glad_glReplacementCodeuiColor3fVertex3fvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FSUNPROC)(GLuint rc, GLfloat nx, GLfloat
ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FSUNPROC glad_glReplacementCodeuiNormal3fVertex3fSUN;
#define glReplacementCodeuiNormal3fVertex3fSUN glad_glReplacementCodeuiNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUINORMAL3FVERTEX3FVSUNPROC glad_glReplacementCodeuiNormal3fVertex3fvSUN;
#define glReplacementCodeuiNormal3fVertex3fvSUN glad_glReplacementCodeuiNormal3fVertex3fvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUICOLOR4FNORMAL3FVERTEX3FSUNPROC)(GLuint rc, GLfloat r, GLfloat g, GLfloat b, GLfloat a, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLREPLACEMENTCODEUICOLOR4FNORMAL3FVERTEX3FSUNPROC glad_glReplacementCodeuiColor4fNormal3fVertex3fSUN;
#define glReplacementCodeuiColor4fNormal3fVertex3fSUN glad_glReplacementCodeuiColor4fNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUICOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *c, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUICOLOR4FNORMAL3FVERTEX3FVSUNPROC glad_glReplacementCodeuiColor4fNormal3fVertex3fvSUN;
#define glReplacementCodeuiColor4fNormal3fVertex3fvSUN glad_glReplacementCodeuiColor4fNormal3fVertex3fvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FSUNPROC)(GLuint rc, GLfloat s, GLfloat t, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FSUNPROC glad_glReplacementCodeuiTexCoord2fVertex3fSUN;
#define glReplacementCodeuiTexCoord2fVertex3fSUN glad_glReplacementCodeuiTexCoord2fVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *tc, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FVERTEX3FVSUNPROC glad_glReplacementCodeuiTexCoord2fVertex3fvSUN;
#define glReplacementCodeuiTexCoord2fVertex3fvSUN glad_glReplacementCodeuiTexCoord2fVertex3fvSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FSUNPROC)(GLuint rc, GLfloat s, GLfloat t, GLfloat nx, GLfloat ny, GLfloat nz, GLfloat x, GLfloat y, GLfloat z);
GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FSUNPROC glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fSUN;
#define glReplacementCodeuiTexCoord2fNormal3fVertex3fSUN glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *tc, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FNORMAL3FVERTEX3FVSUNPROC glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fvSUN;
#define glReplacementCodeuiTexCoord2fNormal3fVertex3fvSUN glad_glReplacementCodeuiTexCoord2fNormal3fVertex3fvSUN
typedef void (APIENTRYP 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);
GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FCOLOR4FNORMAL3FVERTEX3FSUNPROC glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fSUN;
#define glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fSUN glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fSUN
typedef void (APIENTRYP PFNGLREPLACEMENTCODEUITEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC)(const GLuint *rc, const GLfloat *tc, const GLfloat *c, const GLfloat *n, const GLfloat *v);
GLAPI PFNGLREPLACEMENTCODEUITEXCOORD2FCOLOR4FNORMAL3FVERTEX3FVSUNPROC glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fvSUN;
#define glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fvSUN glad_glReplacementCodeuiTexCoord2fColor4fNormal3fVertex3fvSUN
#endif
#ifndef GL_WIN_phong_shading
#define GL_WIN_phong_shading 1
GLAPI int GLAD_GL_WIN_phong_shading;
#endif
#ifndef GL_WIN_specular_fog
#define GL_WIN_specular_fog 1
GLAPI int GLAD_GL_WIN_specular_fog;
#endif

#ifdef __cplusplus
}
#endif

#endif
```

## 7.5 GraphicsController.hpp

```
#ifndef GRAPHICSCONTROLLER_HPP
#define GRAPHICSCONTROLLER_HPP

#include <Error.hpp>
#include <SDL2/SDL.h>
#include <SDL2/SDL_ttf.h>
#include <ShaderProgram.hpp>
#include <functional>
#include <glad/glad.h>
#include <memory>
#include <optional>
#include <string>
#include <unordered_map>
```

```

class GraphicsController {
  public:
    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 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;

  private:
    static SDL_GLContext s_glContext;
};
#endif
```

## 7.6 IntroState.hpp

```
#ifndef INTROSTATE_HPP
#define INTROSTATE_HPP

#include "App.hpp"
#include "IState.hpp"
#include <TexturedRect2D.hpp>
#include <memory>

class IntroState : public IState {
    App &app;

  public:
    IntroState(App &app);
    ~IntroState() override;
    void Run() override;
    void SendEvent(SDL_Event &event) override;

  private:
    struct Slide {
        SDL_Surface *slideSurface;
        float startX;
        float stopX;
        float startY;
        float stopY;
        float panSpeedMS;
    };

    std::shared_ptr<TexturedRect2D> m_slideshow;
    Slide m_slides[3];
    int m_currSlide;

    void NextSlide();

    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;
};

#endif
```

## 7.7 IRender.hpp

```
#ifndef IRENDER_HPP
#define IRENDER_HPP

#include <SDL2/SDL.h>
#include <glad/glad.h>
#include <string>
```

```

class IRender {
  protected:
    GLuint m_vertexArrayObject;
    GLuint m_vertexBufferObject;
    GLuint m_indexBufferObject;
    std::string m_shaderProgram;
    virtual void GenBindBufferGL() = 0;

  public:
    virtual void Render() = 0;
    virtual void UpdateGL() = 0;
};

#endif
```

## 7.8 IState.hpp

```
#ifndef ISTATE_HPP
#define ISTATE_HPP

#include <SDL2/SDL.h>

class IState {
  public:
    virtual void Run() = 0;
    virtual void SendEvent(SDL_Event &event) = 0;
    virtual ~IState() = default;
};

#endif
```

## 7.9 khrplatform.h

```
#ifndef __khrplatform_h_
#define __khrplatform_h_

/*
** Copyright (c) 2008-2018 The Khronos Group Inc.
**
** Permission is hereby granted, free of charge, to any person obtaining a
** copy of this software and/or associated documentation files (the
** "Materials"), to deal in the Materials without restriction, including
** without limitation the rights to use, copy, modify, merge, publish,
** distribute, sublicense, and/or sell copies of the Materials, and to
** permit persons to whom the Materials are furnished to do so, subject to
** the following conditions:
**
** The above copyright notice and this permission notice shall be included
** in all copies or substantial portions of the Materials.
**
** THE MATERIALS ARE PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND,
** EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF
** MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT.
** IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY
** CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT,
** TORT OR OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE
** MATERIALS OR THE USE OR OTHER DEALINGS IN THE MATERIALS.
*/

/* Khronos platform-specific types and definitions.
 *
 * The master copy of khrplatform.h is maintained in the Khronos EGL
 * Registry repository at https://github.com/KhronosGroup/EGL-Registry
 * The last semantic modification to khrplatform.h was at commit ID:
 *      67a3e0864c2d75ea5287b9f3d2eb74a745936692
 *
 * Adopters may modify this file to suit their platform. Adopters are
 * encouraged to submit platform specific modifications to the Khronos
 * group so that they can be included in future versions of this file.
 * Please submit changes by filing pull requests or issues on
 * the EGL Registry repository linked above.
 *
 *
 * See the Implementer’s Guidelines for information about where this file
 * should be located on your system and for more details of its use:
 *    http://www.khronos.org/registry/implementers_guide.pdf
 *
 * This file should be included as
 *        #include <KHR/khrplatform.h>
 * by Khronos client API header files that use its types and defines.
 *
 * The types in khrplatform.h should only be used to define API-specific types.
 *
 * Types defined in khrplatform.h:
 *    khronos_int8_t              signed   8  bit
 *    khronos_uint8_t             unsigned 8  bit
 *    khronos_int16_t             signed   16 bit
 *    khronos_uint16_t            unsigned 16 bit
 *    khronos_int32_t             signed   32 bit
 *    khronos_uint32_t            unsigned 32 bit
 *    khronos_int64_t             signed   64 bit
 *    khronos_uint64_t            unsigned 64 bit
 *    khronos_intptr_t            signed   same number of bits as a pointer
 *    khronos_uintptr_t           unsigned same number of bits as a pointer
 *    khronos_ssize_t             signed   size
 *    khronos_usize_t             unsigned size
 *    khronos_float_t             signed   32 bit floating point
 *    khronos_time_ns_t           unsigned 64 bit time in nanoseconds
 *    khronos_utime_nanoseconds_t unsigned time interval or absolute time in
 *                                         nanoseconds
 *    khronos_stime_nanoseconds_t signed time interval in nanoseconds
 *    khronos_boolean_enum_t      enumerated boolean type. This should
 *      only be used as a base type when a client API's boolean type is
 *      an enum. Client APIs which use an integer or other type for
 *      booleans cannot use this as the base type for their boolean.
 *
 * Tokens defined in khrplatform.h:
 *
 *    KHRONOS_FALSE, KHRONOS_TRUE Enumerated boolean false/true values.
 *
 *    KHRONOS_SUPPORT_INT64 is 1 if 64 bit integers are supported; otherwise 0.
 *    KHRONOS_SUPPORT_FLOAT is 1 if floats are supported; otherwise 0.
 *
 * Calling convention macros defined in this file:
 *    KHRONOS_APICALL
 *    KHRONOS_APIENTRY
 *    KHRONOS_APIATTRIBUTES
 *
 * These may be used in function prototypes as:
 *
 *      KHRONOS_APICALL void KHRONOS_APIENTRY funcname(
 *                                  int arg1,
 *                                  int arg2) KHRONOS_APIATTRIBUTES;
 */

#if defined(__SCITECH_SNAP__) && !defined(KHRONOS_STATIC)
#   define KHRONOS_STATIC 1
#endif

/*-------------------------------------------------------------------------
 * Definition of KHRONOS_APICALL
 *-------------------------------------------------------------------------
 * This precedes the return type of the function in the function prototype.
 */
#if defined(KHRONOS_STATIC)
    /* If the preprocessor constant KHRONOS_STATIC is defined, make the
     * header compatible with static linking. */
#   define KHRONOS_APICALL
#elif defined(_WIN32)
#   define KHRONOS_APICALL __declspec(dllimport)
#elif defined (__SYMBIAN32__)
#   define KHRONOS_APICALL IMPORT_C
#elif defined(__ANDROID__)
#   define KHRONOS_APICALL __attribute__((visibility("default")))
#else
#   define KHRONOS_APICALL
#endif

/*-------------------------------------------------------------------------
 * Definition of KHRONOS_APIENTRY
 *-------------------------------------------------------------------------
 * This follows the return type of the function  and precedes the function
 * name in the function prototype.
 */
#if defined(_WIN32) && !defined(_WIN32_WCE) && !defined(__SCITECH_SNAP__)
    /* Win32 but not WinCE */
#   define KHRONOS_APIENTRY __stdcall
#else
#   define KHRONOS_APIENTRY
#endif

/*-------------------------------------------------------------------------
 * Definition of KHRONOS_APIATTRIBUTES
 *-------------------------------------------------------------------------
 * This follows the closing parenthesis of the function prototype arguments.
 */
#if defined (__ARMCC_2__)
#define KHRONOS_APIATTRIBUTES __softfp
#else
#define KHRONOS_APIATTRIBUTES
#endif

/*-------------------------------------------------------------------------
 * basic type definitions
 *-----------------------------------------------------------------------*/
#if (defined(__STDC_VERSION__) && __STDC_VERSION__ >= 199901L) || defined(__GNUC__) || defined(__SCO__) || defined(__USLC__)


/*
 * Using <stdint.h>
 */
#include <stdint.h>
typedef int32_t                 khronos_int32_t;
typedef uint32_t                khronos_uint32_t;
typedef int64_t                 khronos_int64_t;
typedef uint64_t                khronos_uint64_t;
#define KHRONOS_SUPPORT_INT64   1
#define KHRONOS_SUPPORT_FLOAT   1
/*
 * To support platform where unsigned long cannot be used interchangeably with
 * inptr_t (e.g. CHERI-extended ISAs), we can use the stdint.h intptr_t.
 * Ideally, we could just use (u)intptr_t everywhere, but this could result in
 * ABI breakage if khronos_uintptr_t is changed from unsigned long to
 * unsigned long long or similar (this results in different C++ name mangling).
 * To avoid changes for existing platforms, we restrict usage of intptr_t to
 * platforms where the size of a pointer is larger than the size of long.
 */
#if defined(__SIZEOF_LONG__) && defined(__SIZEOF_POINTER__)
#if __SIZEOF_POINTER__ > __SIZEOF_LONG__
#define KHRONOS_USE_INTPTR_T
#endif
#endif

#elif defined(__VMS ) || defined(__sgi)

/*
 * Using <inttypes.h>
 */
#include <inttypes.h>
typedef int32_t                 khronos_int32_t;
typedef uint32_t                khronos_uint32_t;
typedef int64_t                 khronos_int64_t;
typedef uint64_t                khronos_uint64_t;
#define KHRONOS_SUPPORT_INT64   1
#define KHRONOS_SUPPORT_FLOAT   1

#elif defined(_WIN32) && !defined(__SCITECH_SNAP__)

/*
 * Win32
 */
typedef __int32                 khronos_int32_t;
typedef unsigned __int32        khronos_uint32_t;
typedef __int64                 khronos_int64_t;
typedef unsigned __int64        khronos_uint64_t;
#define KHRONOS_SUPPORT_INT64   1
#define KHRONOS_SUPPORT_FLOAT   1

#elif defined(__sun__) || defined(__digital__)

/*
 * Sun or Digital
 */
typedef int                     khronos_int32_t;
typedef unsigned int            khronos_uint32_t;
#if defined(__arch64__) || defined(_LP64)
typedef long int                khronos_int64_t;
typedef unsigned long int       khronos_uint64_t;
#else
typedef long long int           khronos_int64_t;
typedef unsigned long long int  khronos_uint64_t;
#endif /* __arch64__ */
#define KHRONOS_SUPPORT_INT64   1
#define KHRONOS_SUPPORT_FLOAT   1

#elif 0

/*
 * Hypothetical platform with no float or int64 support
 */
typedef int                     khronos_int32_t;
typedef unsigned int            khronos_uint32_t;
#define KHRONOS_SUPPORT_INT64   0
#define KHRONOS_SUPPORT_FLOAT   0

#else

/*
 * Generic fallback
 */
#include <stdint.h>
typedef int32_t                 khronos_int32_t;
typedef uint32_t                khronos_uint32_t;
typedef int64_t                 khronos_int64_t;
typedef uint64_t                khronos_uint64_t;
#define KHRONOS_SUPPORT_INT64   1
#define KHRONOS_SUPPORT_FLOAT   1

#endif


/*
 * Types that are (so far) the same on all platforms
 */
typedef signed   char          khronos_int8_t;
typedef unsigned char          khronos_uint8_t;
typedef signed   short int     khronos_int16_t;
typedef unsigned short int     khronos_uint16_t;

/*
 * Types that differ between LLP64 and LP64 architectures - in LLP64,
 * pointers are 64 bits, but 'long' is still 32 bits. Win64 appears
 * to be the only LLP64 architecture in current use.
 */
#ifdef KHRONOS_USE_INTPTR_T
typedef intptr_t               khronos_intptr_t;
typedef uintptr_t              khronos_uintptr_t;
#elif defined(_WIN64)
typedef signed   long long int khronos_intptr_t;
typedef unsigned long long int khronos_uintptr_t;
#else
typedef signed   long  int     khronos_intptr_t;
typedef unsigned long  int     khronos_uintptr_t;
#endif

#if defined(_WIN64)
typedef signed   long long int khronos_ssize_t;
typedef unsigned long long int khronos_usize_t;
#else
typedef signed   long  int     khronos_ssize_t;
typedef unsigned long  int     khronos_usize_t;
#endif

#if KHRONOS_SUPPORT_FLOAT
/*
 * Float type
 */
typedef          float         khronos_float_t;
#endif

#if KHRONOS_SUPPORT_INT64
/* Time types
 *
 * These types can be used to represent a time interval in nanoseconds or
 * an absolute Unadjusted System Time.  Unadjusted System Time is the number
 * of nanoseconds since some arbitrary system event (e.g. since the last
 * time the system booted).  The Unadjusted System Time is an unsigned
 * 64 bit value that wraps back to 0 every 584 years.  Time intervals
 * may be either signed or unsigned.
 */
typedef khronos_uint64_t       khronos_utime_nanoseconds_t;
typedef khronos_int64_t        khronos_stime_nanoseconds_t;
#endif

/*
 * Dummy value used to pad enum types to 32 bits.
 */
#ifndef KHRONOS_MAX_ENUM
#define KHRONOS_MAX_ENUM 0x7FFFFFFF
#endif

/*
 * Enumerated boolean type
 *
 * Values other than zero should be considered to be true.  Therefore
 * comparisons should not be made against KHRONOS_TRUE.
 */
typedef enum {
    KHRONOS_FALSE = 0,
    KHRONOS_TRUE  = 1,
    KHRONOS_BOOLEAN_ENUM_FORCE_SIZE = KHRONOS_MAX_ENUM
} khronos_boolean_enum_t;

#endif /* __khrplatform_h_ */
```

## 7.10 MenuState.hpp

```
#ifndef MENUSTATE_HPP
#define MENUSTATE_HPP

#include "App.hpp"
#include "IState.hpp"
#include "TexturedRect2D.hpp"
#include <memory>

class MenuState : public IState {
    App &app;

  public:
    MenuState(App &app);
    ~MenuState() override;
    void Run() override;
    void SendEvent(SDL_Event &event) override;

  private:
    static constexpr SDL_Color TEXT_COLOR = {255, 255, 255};

    // Title Variables
    std::shared_ptr<TexturedRect2D> m_titleText;
    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;

    // Menu Variables
    static constexpr float MENU_W_SCALE = 0.0004;
    static constexpr float MENU_H_SCALE = 0.0004;

    std::shared_ptr<TexturedRect2D> m_startText;
    static constexpr float START_X_OFFSET = 0.0;
    static constexpr float START_Y_OFFSET = 0.0;

    std::shared_ptr<TexturedRect2D> m_quitText;
    static constexpr float QUIT_X_OFFSET = 0.0;
    static constexpr float QUIT_Y_OFFSET = -1.0 / 8.0;

    static constexpr int MENU_COUNT = 2;

    // Selector
    std::shared_ptr<TexturedRect2D> m_selector;
    int selector_pos;
    static constexpr float SELECTOR_PADDING = 1.0 / 32.0;
    void SelectorNext();
    void SelectorPrev();
    void Select();

    enum MenuOption { START = 0, QUIT = 1 };

    // Instructions
    std::shared_ptr<TexturedRect2D> m_instructionText;
    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;
};

#endif
```

## 7.11 MissionState.hpp

```
#ifndef MISSION_HPP
#define MISSION_HPP

#include "App.hpp"
#include "IState.hpp"
#include "VimEmulator.hpp"

```

```
class MissionState : public IState {
    App &app;

  public:
    MissionState(App &app);
    ~MissionState() override;
    void Run() override;
    void SendEvent(SDL_Event &event) override;

  private:
    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;

    std::shared_ptr<VimEmulator> m_vimEmulator;
};

#endif
```

## 7.12 Rect2D.hpp

```
#ifndef RECT2D_HPP
#define RECT2D_HPP

#include <GraphicsController.hpp>
#include <IRender.hpp>
#include <SDL2/SDL.h>
#include <ShaderProgram.hpp>
#include <glad/glad.h>
#include <vector>

class Rect2D : public IRender {
  protected:
    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;

    void GenBindBufferGL() override;

  public:
    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);

    [[nodiscard]] GLfloat GetX() const;
    [[nodiscard]] GLfloat GetY() const;
    [[nodiscard]] GLfloat GetWidth() const;
    [[nodiscard]] GLfloat GetHeight() const;

    [[nodiscard]] std::vector<GLfloat> GetVertexData() const;

    ~Rect2D();

    void Render() override;

    void UpdateGL() override;

    void SetShaderProgram(std::string shaderProgram);

  private:
    static constexpr int RECT_ELEMENTS = 6;
};

#endif
```

## 7.13 SDLToX11Keymap.hpp

```
#ifndef SDLTOX11KEYMAP_HPP
```

```
#define SDLTOX11KEYMAP_HPP

#include <SDL2/SDL.h>
#include <X11/Xlib.h>
#include <unordered_map>

class SDLToX11Keymap {
  public:
    SDLToX11Keymap();
    KeySym convert(SDL_Keycode sdlKey) const;

  private:
    std::unordered_map<SDL_Keycode, KeySym> m_keyMap;
};

#endif
```

## 7.14 ShaderProgram.hpp

```
#ifndef SHADERPROGRAM_HPP
#define SHADERPROGRAM_HPP

#include <glad/glad.h>

class ShaderProgram {
  private:
    friend class ShaderProgramBuilder;
    struct _passKey_t {
        _passKey_t() = default;
    };
    const GLuint m_program_id = 0;

  public:
    ~ShaderProgram();
    [[nodiscard]] GLuint GetProgramId() const;
    ShaderProgram(_passKey_t, GLuint program_id);
};
#endif
```

## 7.15 ShaderProgramBuilder.hpp

```
#ifndef SHADERPROGRAMBUILDER_HPP
#define SHADERPROGRAMBUILDER_HPP

#include <Error.hpp>
#include <ShaderProgram.hpp>
#include <glad/glad.h>
#include <memory>
#include <optional>
#include <string>
#include <variant>
#include <vector>

class ShaderProgramBuilder {
  public:
    ~ShaderProgramBuilder();
    ShaderProgramBuilder &LoadShaderFile(GLenum type, std::string file);
    ShaderProgramBuilder &LoadShader(GLenum type, std::string shaderCode);
    std::variant<Error, std::unique_ptr<ShaderProgram» GenerateShaderProgram();

  private:
    std::vector<GLuint> m_shaders;
    std::optional<Error> m_error = std::nullopt;
};

#endif
```

## 7.16 TexturedRect2D.hpp

```
#ifndef TEXTUREDRECT2D_HPP
#define TEXTUREDRECT2D_HPP

#include <GraphicsController.hpp>
#include <Rect2D.hpp>
#include <SDL2/SDL.h>
```

```
#include <glad/glad.h>
#include <iostream>
#include <vector>

class TexturedRect2D : public Rect2D {
  protected:
    GLuint m_texture;
    GLuint m_texture_width;
    GLuint m_texture_height;

  public:
    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;

  private:
    GLenum m_textureFormat;
    GLenum m_internalTextureFormat;
    bool m_enableBlend;
};

#endif
```

## 7.17 VimEmulator.hpp

```
#ifndef VIMEMULATOR_HPP
#define VIMEMULATOR_HPP
#include <SDL2/SDL.h>
#include <X11/Xlib.h>
#include <mutex>
#include <string>

#include "TexturedRect2D.hpp"

class VimEmulator : public TexturedRect2D {
  public:
    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;

  private:
    static constexpr int REFRESH_MS = 100;
    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;

    static constexpr Uint32 R_MASK = 0XFF0000;
    static constexpr Uint32 G_MASK = 0X00FF00;
    static constexpr Uint32 B_MASK = 0X0000FF;
    static constexpr Uint32 A_MASK = 0;
    XImage *m_xImage;
    SDL_Surface *m_surface;
    bool m_frameReady;

    unsigned int *m_modmask;

    SDL_Surface *GetFrameAsSurface();
    Window *findWindowByName(Window window);
    void RegisterWindowThread();
    void ResizeWindowThread(int w, int h);
    // Replace this with a generic event handler if needed
    bool MatchResizeEvent(int w, int h, XEvent *event);
    void QueueFrameThread();
};

#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