



Appendix B

State Variables

This appendix lists the queryable OpenGL state variables, their default values, and the commands for obtaining the values of these variables. The *OpenGL Reference Manual* contains detailed information on all the commands and constants discussed in this appendix. This appendix has these major sections:

- "The Query Commands"
 - "OpenGL State Variables"
-

The Query Commands

In addition to the basic commands to obtain the values of simple state variables (commands such as **glGetIntegerv()** and **glIsEnabled()**, which are described in "Basic State Management" in Chapter 2), there are other specialized commands to return more complex state variables. The prototypes for these specialized commands are listed here. Some of these routines, such as **glGetError()** and **glGetString()**, have been discussed in more detail elsewhere in the book.

To find out when you need to use these commands and their corresponding symbolic constants, use the tables in the next section, "OpenGL State Variables." Also see the *OpenGL Reference Manual*.

```
void glGetClipPlane(GLenum plane, GLdouble *equation);
```

```
GLenum glGetError(void);
```

```
void glGetLight{if}v(GLenum light, GLenum pname, TYPE *params);
```

```
void glGetMap{ifd}v(GLenum target, GLenum query, TYPE *v);
```

```
void glGetMaterial{if}v(GLenum face, GLenum pname, TYPE *params);
```

```
void glGetPixelMap{f ui us}v(GLenum map, TYPE *values);
```

```
void glGetPolygonStipple(GLubyte *mask);
```

```
const GLubyte * glGetString(GLenum name);
```

```
void glGetTexEnv{if}v(GLenum target, GLenum pname, TYPE *params);
```

```
void glGetTexGen{ifd}v(GLenum coord, GLenum pname, TYPE *params);
```

```
void glGetTexImage(GLenum target, GLint level, GLenum format,  
GLenum type, GLvoid *pixels);
```

```
void glGetTexLevelParameter{if}v(GLenum target, GLint level,  
GLenum pname, TYPE *params);
```

```
void glGetTexParameter{if}v(GLenum target, GLenum pname,  
TYPE *params);
```

```
void gluGetNurbsProperty(GLUnurbsObj *nobj, GLenum property,  
GLfloat *value);
```

```
const GLubyte * gluGetString(GLenum name);
```

```
void gluGetTessProperty(GLUtesselator *tess, GLenum which,  
GLdouble *data);
```

OpenGL State Variables

The following pages contain tables that list the names of queryable state variables. For each variable, the tables list a description of it, its attribute group, its initial or minimum value, and the suggested **glGet*()** command to use for obtaining it. State variables that can be obtained using **glGetBooleanv()**, **glGetIntegerv()**, **glGetFloatv()**, or **glGetDoublev()** are listed with just one of these commands - the one that's most appropriate given the type of data to be returned. (Some vertex array variables can be queried only with **glGetPointerv()**.) These state variables can't be obtained using **glIsEnabled()**. However, state variables for which **glIsEnabled()** is listed as the query command can also be obtained using **glGetBooleanv()**, **glGetIntegerv()**, **glGetFloatv()**, and **glGetDoublev()**. State variables for which any other command is listed as the query command can be obtained only by using that command.

If one or more attribute groups are listed, the state variable belongs to the listed group or groups. If no attribute group is listed, the variable doesn't belong to any group. **glPushAttrib()**, **glPushClientAttrib()**, **glPopAttrib()**, and **glPopClientAttrib()** may be used to save and restore all state values that belong to an attribute group. (See "Attribute Groups" in Chapter 2 for more information.)

All queryable state variables, except the implementation-dependent ones, have initial values. If no initial value is listed, you need to consult either the section where that variable is discussed or the *OpenGL Reference Manual* to determine its initial value.

Current Values and Associated Data

Table B-1 : State Variables for Current Values and Associated Data

--

State Variable	Description	Attribute Group	Initial Value
GL_CURRENT_COLOR	Current color	current	1, 1, 1, 1
GL_CURRENT_INDEX	Current color index	current	1
GL_CURRENT_TEXTURE_COORDS	Current texture coordinates	current	0, 0, 0, 1
GL_CURRENT_NORMAL	Current normal	current	0, 0, 1
GL_CURRENT_RASTER_POSITION	Current raster position	current	0, 0, 0, 1
GL_CURRENT_RASTER_DISTANCE	Current raster distance	current	0
GL_CURRENT_RASTER_COLOR	Color associated with raster position	current	1, 1, 1, 1
GL_CURRENT_RASTER_INDEX	Color index associated with raster position	current	1
GL_CURRENT_RASTER_TEXTURE_COORDS	Texture coordinates associated with raster position	current	0, 0, 0, 1
GL_CURRENT_RASTER_POSITION_VALID	Raster position valid bit	current	GL_TRUE
GL_EDGE_FLAG	Edge flag	current	GL_TRUE

Vertex Array

Table B-2 : (continued) Vertex Array State Variables

State Variable	Description	Attribute Group	Initial Value
GL_VERTEX_ARRAY	Vertex array enable	vertex-array	GL_FALSE
GL_VERTEX_ARRAY_SIZE	Coordinates per vertex	vertex-array	4
GL_VERTEX_ARRAY_TYPE	Type of vertex coordinates	vertex-array	GL_FLOAT
GL_VERTEX_ARRAY_STRIDE	Stride between vertices	vertex-array	0
GL_VERTEX_ARRAY_POINTER	Pointer to the vertex array	vertex-array	NULL
GL_NORMAL_ARRAY	Normal array enable	vertex-array	GL_FALSE
GL_NORMAL_ARRAY_TYPE	Type of normal coordinates	vertex-array	GL_FLOAT
GL_NORMAL_ARRAY_STRIDE	Stride between normals	vertex-array	0
GL_NORMAL_ARRAY_POINTER	Pointer to the normal array	vertex-array	NULL
GL_COLOR_ARRAY	RGBA color array enable	vertex-array	GL_FALSE
GL_COLOR_ARRAY_SIZE	Colors per vertex	vertex-array	4

GL_COLOR_ARRAY_TYPE	Type of color components	vertex-array	GL_FLOAT
GL_COLOR_ARRAY_STRIDE	Stride between colors	vertex-array	0
GL_COLOR_ARRAY_POINTER	Pointer to the color array	vertex-array	NULL
GL_INDEX_ARRAY	Color-index array enable	vertex-array	GL_FALSE
GL_INDEX_ARRAY_TYPE	Type of color indices	vertex-array	GL_FLOAT
GL_INDEX_ARRAY_STRIDE	Stride between color indices	vertex-array	0
GL_INDEX_ARRAY_POINTER	Pointer to the index array	vertex-array	NULL
GL_TEXTURE_COORD_ARRAY	Texture coordinate array enable	vertex-array	GL_FALSE
GL_TEXTURE_COORD_ARRAY_SIZE	Texture coordinates per element	vertex-array	4
GL_TEXTURE_COORD_ARRAY_TYPE	Type of texture coordinates	vertex-array	GL_FLOAT
GL_TEXTURE_COORD_ARRAY_STRIDE	Stride between texture coordinates	vertex-array	0

GL_TEXTURE_COORD_ARRAY_POINTER	Pointer to the texture coordinate array	vertex-array	NULL
GL_EDGE_FLAG_ARRAY	Edge flag array enable	vertex-array	GL_FALSE
GL_EDGE_FLAG_ARRAY_STRIDE	Stride between edge flags	vertex-array	0
GL_EDGE_FLAG_ARRAY_POINTER	Pointer to the edge flag array	vertex-array	NULL

Transformation

Table B-3 : Transformation State Variables

State Variable	Description	Attribute Group	Initial Value	
GL_MODELVIEW_MATRIX	Modelview matrix stack	-	Identity	ξ
GL_PROJECTION_MATRIX	Projection matrix stack	-	Identity	ξ
GL_TEXTURE_MATRIX	Texture matrix stack	-	Identity	ξ
GL_VIEWPORT	Viewport origin and extent	viewport	-	ξ
GL_DEPTH_RANGE	Depth range near and far	viewport	0, 1	ξ
GL_MODELVIEW_STACK_DEPTH	Modelview matrix stack pointer	-	1	ξ

GL_PROJECTION_STACK_DEPTH	Projection matrix stack pointer	-	1	ξ
GL_TEXTURE_STACK_DEPTH	Texture matrix stack pointer	-	1	ξ
GL_MATRIX_MODE	Current matrix mode	transform	GL_MODELVIEW	ξ
GL_NORMALIZE	Current normal normalization on/off	transform/enable	GL_FALSE	ξ
GL_CLIP_PLANE _{<i>i</i>}	User clipping plane coefficients	transform	0, 0, 0, 0	ξ
GL_CLIP_PLANE _{<i>i</i>}	<i>i</i> th user clipping plane enabled	transform/enable	GL_FALSE	ξ

Coloring

Table B-4 : Coloring State Variables

State Variable	Description	Attribute Group	Initial Value	Get Command
GL_FOG_COLOR	Fog color	fog	0, 0, 0, 0	glGetFloatv()
GL_FOG_INDEX	Fog index	fog	0	glGetFloatv()
GL_FOG_DENSITY	Exponential fog density	fog	1.0	glGetFloatv()
GL_FOG_START	Linear fog start	fog	0.0	glGetFloatv()
GL_FOG_END	Linear fog end	fog	1.0	glGetFloatv()
GL_FOG_MODE	Fog mode	fog	GL_EXP	glGetIntegerv()
GL_FOG	True if fog enabled	fog/enable	GL_FALSE	glIsEnabled()
GL_SHADE_MODEL	glShadeModel() setting	lighting	GL_SMOOTH	glGetIntegerv()

Lighting

See also Table 5-1 and Table 5-3 for initial values.

Table B-5 : (continued) Lighting State Variables

State Variable	Description	Attribute Group	Initial Value	C
GL_LIGHTING	True if lighting is enabled	lighting/enable	GL_FALSE	g
GL_COLOR_MATERIAL	True if color tracking is enabled	lighting	GL_FALSE	g

GL_COLOR_MATERIAL_PARAMETER	Material properties tracking current color	lighting	GL_AMBIENT_AND_DIFFUSE	ε
GL_COLOR_MATERIAL_FACE	Face(s) affected by color tracking	lighting	GL_FRONT_AND_BACK	ε
GL_AMBIENT	Ambient material color	lighting	(0.2, 0.2, 0.2, 1.0)	ε
GL_DIFFUSE	Diffuse material color	lighting	(0.8, 0.8, 0.8, 1.0)	ε
GL_SPECULAR	Specular material color	lighting	(0.0, 0.0, 0.0, 1.0)	ε
GL_EMISSION	Emissive material color	lighting	(0.0, 0.0, 0.0, 1.0)	ε
GL_SHININESS	Specular exponent of material	lighting	0.0	ε
GL_LIGHT_MODEL_AMBIENT	Ambient scene color	lighting	(0.2, 0.2, 0.2, 1.0)	ε
GL_LIGHT_MODEL_LOCAL_VIEWER	Viewer is local	lighting	GL_FALSE	ε
GL_LIGHT_MODEL_TWO_SIDE	Use two-sided lighting	lighting	GL_FALSE	ε
GL_AMBIENT	Ambient intensity of light <i>i</i>	lighting	(0.0,0.0,0.0,1.0)	ε

GL_DIFFUSE	Diffuse intensity of light <i>i</i>	lighting	-	Ⓔ
GL_SPECULAR	Specular intensity of light <i>i</i>	lighting	-	Ⓔ
GL_POSITION	Position of light <i>i</i>	lighting	(0.0, 0.0, 1.0, 0.0)	Ⓔ
GL_CONSTANT_ATTENUATION	Constant attenuation factor	lighting	1.0	Ⓔ
GL_LINEAR_ATTENUATION	Linear attenuation factor	lighting	0.0	Ⓔ
GL_QUADRATIC_ATTENUATION	Quadratic attenuation factor	lighting	0.0	Ⓔ
GL_SPOT_DIRECTION	Spotlight direction of light <i>i</i>	lighting	(0.0, 0.0, -1.0)	Ⓔ
GL_SPOT_EXPONENT	Spotlight exponent of light <i>i</i>	lighting	0.0	Ⓔ
GL_SPOT_CUTOFF	Spotlight angle of light <i>i</i>	lighting	180.0	Ⓔ
GL_LIGHT <i>i</i>	True if light <i>i</i> enabled	lighting/enable	GL_FALSE	Ⓔ
GL_COLOR_INDEXES	ca, cd, and cs for color-index lighting	lighting/enable	0, 1, 1	Ⓔ

Rasterization

Table B-6 : (continued) Rasterization State Variables

State Variable	Description	Attribute Group	Initial Value
GL_POINT_SIZE	Point size	point	1.0
GL_POINT_SMOOTH	Point antialiasing on	point/enable	GL_FALSE
GL_LINE_WIDTH	Line width	line	1.0
GL_LINE_SMOOTH	Line antialiasing on	line/enable	GL_FALSE
GL_LINE_STIPPLE_PATTERN	Line stipple	line	1's
GL_LINE_STIPPLE_REPEAT	Line stipple repeat	line	1
GL_LINE_STIPPLE	Line stipple enable	line/enable	GL_FALSE
GL_CULL_FACE	Polygon culling enabled	polygon/enable	GL_FALSE
GL_CULL_FACE_MODE	Cull front-/back-facing polygons	polygon	GL_BACK
GL_FRONT_FACE	Polygon front-face CW/CCW indicator	polygon	GL_CCW
GL_POLYGON_SMOOTH	Polygon antialiasing on	polygon/enable	GL_FALSE
GL_POLYGON_MODE	Polygon rasterization mode (front and back)	polygon	GL_FILL
GL_POLYGON_OFFSET_FACTOR	Polygon offset factor	polygon	0
GL_POLYGON_OFFSET_BIAS	Polygon offset bias	polygon	0

GL_POLYGON_OFFSET_POINT	Polygon offset enable for GL_POINT mode rasterization	polygon/enable	GL_FALSE
GL_POLYGON_OFFSET_LINE	Polygon offset enable for GL_LINE mode rasterization	polygon/enable	GL_FALSE
GL_POLYGON_OFFSET_FILL	Polygon offset enable for GL_FILL mode rasterization	polygon/enable	GL_FALSE
GL_POLYGON_STIPPLE	Polygon stipple enable	polygon/enable	GL_FALSE
-	Polygon stipple pattern	polygon-stipple	1's

Texturing

Table B-7 : (continued) Texturing State Variables

State Variable	Description	Attribute Group	Initial Value	Ge
GL_TEXTURE_ <i>x</i>	True if <i>x</i> -D texturing enabled (<i>x</i> is 1D or 2D)	texture/enable	GL_FALSE	gl
GL_TEXTURE_BINDING_ <i>x</i>	Texture object bound to GL_TEXTURE_ <i>x</i> (<i>x</i> is 1D or 2D)	texture	GL_FALSE	gl
GL_TEXTURE	<i>x</i> -D texture image at level of detail <i>i</i>	-	-	gl
GL_TEXTURE_WIDTH	<i>x</i> -D texture image <i>i</i> 's width	-	0	gl
GL_TEXTURE_HEIGHT	<i>x</i> -D texture image <i>i</i> 's height	-	0	gl

GL_TEXTURE_BORDER	<i>x</i> -D texture image <i>i</i> 's border width	-	0	glC
GL_TEXTURE_INTERNAL_FORMAT	<i>x</i> -D texture image <i>i</i> 's internal image format	-	1	glC
GL_TEXTURE_RED_SIZE	<i>x</i> -D texture image <i>i</i> 's red resolution	-	0	glC
GL_TEXTURE_GREEN_SIZE	<i>x</i> -D texture image <i>i</i> 's green resolution	-	0	glC
GL_TEXTURE_BLUE_SIZE	<i>x</i> -D texture image <i>i</i> 's blue resolution	-	0	glC
GL_TEXTURE_ALPHA_SIZE	<i>x</i> -D texture image <i>i</i> 's alpha resolution	-	0	glC
GL_TEXTURE_LUMINANCE_SIZE	<i>x</i> -D texture image <i>i</i> 's luminance resolution	-	0	glC
GL_TEXTURE_INTENSITY_SIZE	<i>x</i> -D texture image <i>i</i> 's intensity resolution	-	0	glC
GL_TEXTURE_BORDER_COLOR	Texture border color	texture	0, 0, 0, 0	glC
GL_TEXTURE_MIN_FILTER	Texture minification function	texture	GL_NEAREST_MIPMAP_LINEAR	glC
GL_TEXTURE_MAG_FILTER	Texture magnification function	texture	GL_LINEAR	glC
GL_TEXTURE_WRAP_ <i>x</i>	Texture wrap mode (<i>x</i> is S or T)	texture	GL_REPEAT	glC
GL_TEXTURE_PRIORITY	Texture object priority	texture	1	glC

GL_TEXTURE_RESIDENCY	Texture residency	texture	GL_FALSE	glC
GL_TEXTURE_ENV_MODE	Texture application function	texture	GL_MODULATE	glC
GL_TEXTURE_ENV_COLOR	Texture environment color	texture	0, 0, 0, 0	glC
GL_TEXTURE_GEN_x	Texgen enabled (x is S, T, R, or Q)	texture/enable	GL_FALSE	glC
GL_EYE_PLANE	Texgen plane equation coefficients	texture	-	glC
GL_OBJECT_PLANE	Texgen object linear coefficients	texture	-	glC
GL_TEXTURE_GEN_MODE	Function used for texgen	texture	GL_EYE_LINEAR	glC

Pixel Operations

Table B-8 : (continued) Pixel Operations

State Variable	Description	Attribute Group	Initial Value	GL C
GL_SCISSOR_TEST	Scissoring enabled	scissor/enable	GL_FALSE	g
GL_SCISSOR_BOX	Scissor box	scissor	-	g
GL_ALPHA_TEST	Alpha test enabled	color-buffer/enable	GL_FALSE	g
GL_ALPHA_TEST_FUNC	Alpha test function	color-buffer	GL_ALWAYS	g
GL_ALPHA_TEST_REF	Alpha test reference value	color-buffer	0	g

GL_STENCIL_TEST	Stenciling enabled	stencil-buffer/enable	GL_FALSE	g
GL_STENCIL_FUNC	Stencil function	stencil-buffer	GL_ALWAYS	g
GL_STENCIL_VALUE_MASK	Stencil mask	stencil-buffer	1's	g
GL_STENCIL_REF	Stencil reference value	stencil-buffer	0	g
GL_STENCIL_FAIL	Stencil fail action	stencil-buffer	GL_KEEP	g
GL_STENCIL_PASS_DEPTH_FAIL	Stencil depth buffer fail action	stencil-buffer	GL_KEEP	g
GL_STENCIL_PASS_DEPTH_PASS	Stencil depth buffer pass action	stencil-buffer	GL_KEEP	g
GL_DEPTH_TEST	Depth buffer enabled	depth-buffer/enable	GL_FALSE	g
GL_DEPTH_FUNC	Depth buffer test function	depth-buffer	GL_LESS	g
GL_BLEND	Blending enabled	color-buffer/enable	GL_FALSE	g
GL_BLEND_SRC	Blending source function	color-buffer	GL_ONE	g
GL_BLEND_DST	Blending destination function	color-buffer	GL_ZERO	g
GL_DITHER	Dithering enabled	color-buffer/enable	GL_TRUE	g

GL_INDEX_LOGIC_OP	Color index logical operation enabled	color-buffer/enable	GL_FALSE	g
GL_COLOR_LOGIC_OP	RGBA color logical operation enabled	color-buffer/enable	GL_FALSE	g
GL_LOGIC_OP_MODE	Logical operation function	color-buffer	GL_COPY	g

Framebuffer Control

Table B-9 : Framebuffer Control State Variables

State Variable	Description	Attribute Group	Initial Value	Get
GL_DRAW_BUFFER	Buffers selected for drawing	color-buffer	-	glGe
GL_INDEX_WRITEMASK	Color-index writemask	color-buffer	1's	glGe
GL_COLOR_WRITEMASK	Color write enables; R, G, B, or A	color-buffer	GL_TRUE	glGe
GL_DEPTH_WRITEMASK	Depth buffer enabled for writing	depth-buffer	GL_TRUE	glGe
GL_STENCIL_WRITEMASK	Stencil-buffer writemask	stencil-buffer	1's	glGe
GL_COLOR_CLEAR_VALUE	Color-buffer clear value (RGBA mode)	color-buffer	0, 0, 0, 0	glGe
GL_INDEX_CLEAR_VALUE	Color-buffer clear value (color-index mode)	color-buffer	0	glGe
GL_DEPTH_CLEAR_VALUE	Depth-buffer clear value	depth-buffer	1	glGe
GL_STENCIL_CLEAR_VALUE	Stencil-buffer clear value	stencil-buffer	0	glGe
GL_ACCUM_CLEAR_VALUE	Accumulation-buffer clear value	accum-buffer	0	glGe

Pixels

Table B-10 : (continued) Pixel State Variables

State Variable	Description	Attribute Group	Initial Value
GL_UNPACK_SWAP_BYTES	Value of GL_UNPACK_SWAP_BYTES	pixel-store	GL_FALSE

GL_UNPACK_LSB_FIRST	Value of GL_UNPACK_LSB_FIRST	pixel-store	GL_FALSE
GL_UNPACK_ROW_LENGTH	Value of GL_UNPACK_ROW_LENGTH	pixel-store	0
GL_UNPACK_SKIP_ROWS	Value of GL_UNPACK_SKIP_ROWS	pixel-store	0
GL_UNPACK_SKIP_PIXELS	Value of GL_UNPACK_SKIP_PIXELS	pixel-store	0
GL_UNPACK_ALIGNMENT	Value of GL_UNPACK_ALIGNMENT	pixel-store	4
GL_PACK_SWAP_BYTES	Value of GL_PACK_SWAP_BYTES	pixel-store	GL_FALSE
GL_PACK_LSB_FIRST	Value of GL_PACK_LSB_FIRST	pixel-store	GL_FALSE
GL_PACK_ROW_LENGTH	Value of GL_PACK_ROW_LENGTH	pixel-store	0
GL_PACK_SKIP_ROWS	Value of GL_PACK_SKIP_ROWS	pixel-store	0
GL_PACK_SKIP_PIXELS	Value of GL_PACK_SKIP_PIXELS	pixel-store	0
GL_PACK_ALIGNMENT	Value of GL_PACK_ALIGNMENT	pixel-store	4
GL_MAP_COLOR	True if colors are mapped	pixel	GL_FALSE
GL_MAP_STENCIL	True if stencil values are mapped	pixel	GL_FALSE
GL_INDEX_SHIFT	Value of GL_INDEX_SHIFT	pixel	0
GL_INDEX_OFFSET	Value of GL_INDEX_OFFSET	pixel	0
GL_x_SCALE	Value of GL_x_SCALE; <i>x</i> is GL_RED, GL_GREEN, GL_BLUE, GL_ALPHA, or GL_DEPTH	pixel	1

GL_ <i>x</i> _BIAS	Value of GL_ <i>x</i> _BIAS; <i>x</i> is one of GL_RED, GL_GREEN, GL_BLUE, GL_ALPHA, or GL_DEPTH	pixel	0
GL_ZOOM_X	<i>x</i> zoom factor	pixel	1.0
GL_ZOOM_Y	<i>y</i> zoom factor	pixel	1.0
GL_ <i>x</i>	glPixelMap() translation tables; <i>x</i> is a map name from Table 8-1	-	0's
GL_ <i>x</i> _SIZE	Size of table <i>x</i>	-	1
GL_READ_BUFFER	Read source buffer	pixel	-

Evaluators

Table B-11 : Evaluator State Variables

State Variable	Description	Attribute Group	Initial Value	Get Command
GL_ORDER	1D map order	-	1	glGetMapiv()
GL_ORDER	2D map orders	-	1, 1	glGetMapiv()
GL_COEFF	1D control points	-	-	glGetMapfv()
GL_COEFF	2D control points	-	-	glGetMapfv()
GL_DOMAIN	1D domain endpoints	-	-	glGetMapfv()
GL_DOMAIN	2D domain endpoints	-	-	glGetMapfv()

GL_MAP1_x	1D map enables: <i>x</i> is map type	eval/enable	GL_FALSE	glIsEnabled()
GL_MAP2_x	2D map enables: <i>x</i> is map type	eval/enable	GL_FALSE	glIsEnabled()
GL_MAP1_GRID_DOMAIN	1D grid endpoints	eval	0, 1	glGetFloatv()
GL_MAP2_GRID_DOMAIN	2D grid endpoints	eval	0, 1; 0, 1	glGetFloatv()
GL_MAP1_GRID_SEGMENTS	1D grid divisions	eval	1	glGetFloatv()
GL_MAP2_GRID_SEGMENTS	2D grid divisions	eval	1,1	glGetFloatv()
GL_AUTO_NORMAL	True if automatic normal generation enabled	eval	GL_FALSE	glIsEnabled()

Hints

Table B-12 : Hint State Variables

State Variable	Description	Attribute Group	Initial Value
GL_PERSPECTIVE_CORRECTION_HINT	Perspective correction hint	hint	GL_DONT_CARE
GL_POINT_SMOOTH_HINT	Point smooth hint	hint	GL_DONT_CARE
GL_LINE_SMOOTH_HINT	Line smooth hint	hint	GL_DONT_CARE
GL_POLYGON_SMOOTH_HINT	Polygon smooth hint	hint	GL_DONT_CARE
GL_FOG_HINT	Fog hint	hint	GL_DONT_CARE

Implementation-Dependent Values

Table B-13 : (continued) Implementation-Dependent State Variables

State Variable	Description	Attribute Group	Minimum Value
GL_MAX_LIGHTS	Maximum number of lights	-	8
GL_MAX_CLIP_PLANES	Maximum number of user clipping planes	-	6
GL_MAX_MODELVIEW_STACK_DEPTH	Maximum modelview-matrix stack depth	-	32
GL_MAX_PROJECTION_STACK_DEPTH	Maximum projection-matrix stack depth	-	2
GL_MAX_TEXTURE_STACK_DEPTH	Maximum depth of texture matrix stack	-	2

GL_SUBPIXEL_BITS	Number of bits of subpixel precision in x and y	-	4
GL_MAX_TEXTURE_SIZE	See discussion in "Texture Proxy" in Chapter 9	-	64
GL_MAX_PIXEL_MAP_TABLE	Maximum size of a glPixelMap() translation table	-	32
GL_MAX_NAME_STACK_DEPTH	Maximum selection-name stack depth	-	64
GL_MAX_LIST_NESTING	Maximum display-list call nesting	-	64
GL_MAX_EVAL_ORDER	Maximum evaluator polynomial order	-	8
GL_MAX_VIEWPORT_DIMS	Maximum viewport dimensions	-	-
GL_MAX_ATTRIB_STACK_DEPTH	Maximum depth of the attribute stack	-	16
GL_MAX_CLIENT_ATTRIB_STACK_DEPTH	Maximum depth of the client attribute stack	-	16
GL_AUX_BUFFERS	Number of auxiliary buffers	-	0
GL_RGBA_MODE	True if color buffers store RGBA	-	-
GL_INDEX_MODE	True if color buffers store indices	-	-

GL_DOUBLEBUFFER	True if front and back buffers exist	-	-
GL_STEREO	True if left and right buffers exist	-	-
GL_POINT_SIZE_RANGE	Range (low to high) of antialiased point sizes	-	1, 1
GL_POINT_SIZE_GRANULARITY	Antialiased point-size granularity	-	-
GL_LINE_WIDTH_RANGE	Range (low to high) of antialiased line widths	-	1, 1
GL_LINE_WIDTH_GRANULARITY	Antialiased line-width granularity	-	-

Implementation-Dependent Pixel Depths

Table B-14 : Implementation-Dependent Pixel-Depth State Variables (continued)

State Variable	Description	Attribute Group	Minimum Value	Get Command
GL_RED_BITS	Number of bits per red component in color buffers	-	-	glGetIntegerv()
GL_GREEN_BITS	Number of bits per green component in color buffers	-	-	glGetIntegerv()
GL_BLUE_BITS	Number of bits per blue component in color buffers	-	-	glGetIntegerv()

GL_ALPHA_BITS	Number of bits per alpha component in color buffers	-	-	glGetIntegerv()
GL_INDEX_BITS	Number of bits per index in color buffers	-	-	glGetIntegerv()
GL_DEPTH_BITS	Number of depth-buffer bitplanes	-	-	glGetIntegerv()
GL_STENCIL_BITS	Number of stencil bitplanes	-	-	glGetIntegerv()
GL_ACCUM_RED_BITS	Number of bits per red component in the accumulation buffer	-	-	glGetIntegerv()
GL_ACCUM_GREEN_BITS	Number of bits per green component in the accumulation buffer	-	-	glGetIntegerv()
GL_ACCUM_BLUE_BITS	Number of bits per blue component in the accumulation buffer	-	-	glGetIntegerv()
GL_ACCUM_ALPHA_BITS	Number of bits per alpha component in the accumulation buffer	-	-	glGetIntegerv()

Miscellaneous

Table B-15 : Miscellaneous State Variables

State Variable	Description	Attribute Group	Initial Value	
GL_LIST_BASE	Setting of glListBase()	list	0	ξ
GL_LIST_INDEX	Number of display list under construction; 0 if none	-	0	ξ
GL_LIST_MODE	Mode of display list under construction; undefined if none	-	0	ξ
GL_ATTRIB_STACK_DEPTH	Attribute stack pointer	-	0	ξ
GL_CLIENT_ATTRIB_STACK_DEPTH	Client attribute stack pointer	-	0	ξ
GL_NAME_STACK_DEPTH	Name stack depth	-	0	ξ
GL_RENDER_MODE	glRenderMode() setting	-	GL_RENDER	ξ
GL_SELECTION_BUFFER_POINTER	Pointer to selection buffer	select	0	ξ
GL_SELECTION_BUFFER_SIZE	Size of selection buffer	select	0	ξ
GL_FEEDBACK_BUFFER_POINTER	Pointer to feedback buffer	feedback	0	ξ
GL_FEEDBACK_BUFFER_SIZE	Size of feedback buffer	feedback	0	ξ
GL_FEEDBACK_BUFFER_TYPE	Type of feedback buffer	feedback	GL_2D	ξ

-	Current error code(s)	-	0	ξ
---	-----------------------	---	---	---



OpenGL Programming Guide
(Addison-Wesley Publishing Company)