NAME

OpenGL Interoperability [DEPRECATED] −

Enumerations

enum **cudaGLMapFlags** { **cudaGLMapFlagsNone** = 0, **cudaGLMapFlagsReadOnly** = 1, **cudaGLMapFlagsWriteDiscard** = 2 }

Functions

cudaError_t cudaGLMapBufferObject (void **devPtr, GLuint bufObj)
Maps a buffer object for access by CUDA.
cudaError_t cudaGLMapBufferObjectAsync (void **devPtr, GLuint bufObj, **cudaStream_t** stream)
Maps a buffer object for access by CUDA.
cudaError_t cudaGLRegisterBufferObject (GLuint bufObj)
Registers a buffer object for access by CUDA.
cudaError_t cudaGLSetBufferObjectMapFlags (GLuint bufObj, unsigned int flags)
Set usage flags for mapping an OpenGL buffer.
cudaError_t cudaGLSetGLDevice (int device)
Sets a CUDA device to use OpenGL interoperability.
cudaError_t cudaGLUnmapBufferObject (GLuint bufObj)
Unmaps a buffer object for access by CUDA.
cudaError_t cudaGLUnmapBufferObjectAsync (GLuint bufObj, **cudaStream_t** stream)
Unmaps a buffer object for access by CUDA.
cudaError_t cudaGLUnregisterBufferObject (GLuint bufObj)
Unregisters a buffer object for access by CUDA.

Detailed Description

This section describes deprecated OpenGL interoperability functionality.

Enumeration Type Documentation

enum cudaGLMapFlags

CUDA GL Map Flags

Enumerator:

cudaGLMapFlagsNone
Default; Assume resource can be read/written

cudaGLMapFlagsReadOnly
CUDA kernels will not write to this resource

cudaGLMapFlagsWriteDiscard
CUDA kernels will only write to and will not read from this resource

Function Documentation

cudaError_t cudaGLMapBufferObject (void ** devPtr, GLuint bufObj)

Deprecated

This function is deprecated as of CUDA 3.0.

Maps the buffer object of ID `bufObj` into the address space of CUDA and returns in `*devPtr` the base pointer of the resulting mapping. The buffer must have previously been registered by calling **cudaGLRegisterBufferObject()**. While a buffer is mapped by CUDA, any OpenGL operation which references the buffer will result in undefined behavior. The OpenGL context used to create the buffer, or another context from the same share group, must be bound to the current thread when this is called.

All streams in the current thread are synchronized with the current GL context.

Parameters:

devPtr - Returned device pointer to CUDA object
bufObj - Buffer object ID to map

Returns:

cudaSuccess, **cudaErrorMapBufferObjectFailed**

Note:

Note that this function may also return error codes from previous, asynchronous launches.

See also:

cudaGraphicsMapResources

cudaError_t cudaGLMapBufferObjectAsync (void ** devPtr, GLuint bufObj, cudaStream_t stream)

Deprecated

This function is deprecated as of CUDA 3.0.

Maps the buffer object of ID `bufObj` into the address space of CUDA and returns in `*devPtr` the base pointer of the resulting mapping. The buffer must have previously been registered by calling **cudaGLRegisterBufferObject()**. While a buffer is mapped by CUDA, any OpenGL operation which references the buffer will result in undefined behavior. The OpenGL context used to create the buffer, or another context from the same share group, must be bound to the current thread when this is called.

Stream /p stream is synchronized with the current GL context.

Parameters:

devPtr - Returned device pointer to CUDA object
bufObj - Buffer object ID to map
stream - Stream to synchronize

Returns:

cudaSuccess, **cudaErrorMapBufferObjectFailed**

Note:

Note that this function may also return error codes from previous, asynchronous launches.

See also:

cudaGraphicsMapResources

cudaError_t cudaGLRegisterBufferObject (GLuint bufObj)

Deprecated

This function is deprecated as of CUDA 3.0.

Registers the buffer object of ID `bufObj` for access by CUDA. This function must be called before CUDA can map the buffer object. The OpenGL context used to create the buffer, or another context from the same share group, must be bound to the current thread when this is called.

Parameters:

bufObj - Buffer object ID to register

Returns:

cudaSuccess, **cudaErrorInitializationError**

Note:

Note that this function may also return error codes from previous, asynchronous launches.

See also:

cudaGraphicsGLRegisterBuffer

cudaError_t cudaGLSetBufferObjectMapFlags (GLuint bufObj, unsigned int flags)

Deprecated

This function is deprecated as of CUDA 3.0.

Set flags for mapping the OpenGL buffer `bufObj`

Changes to flags will take effect the next time `bufObj` is mapped. The `flags` argument may be any of the following:

- **cudaGLMapFlagsNone**: Specifies no hints about how this buffer will be used. It is therefore assumed that this buffer will be read from and written to by CUDA kernels. This is the default value.
- **cudaGLMapFlagsReadOnly**: Specifies that CUDA kernels which access this buffer will not write to the buffer.
- **cudaGLMapFlagsWriteDiscard**: Specifies that CUDA kernels which access this buffer will not read from the buffer and will write over the entire contents of the buffer, so none of the data previously stored in the buffer will be preserved.

If `bufObj` has not been registered for use with CUDA, then **cudaErrorInvalidResourceHandle** is returned. If `bufObj` is presently mapped for access by CUDA, then **cudaErrorUnknown** is returned.

Parameters:

bufObj - Registered buffer object to set flags for
flags - Parameters for buffer mapping

Returns:
cudaSuccess, **cudaErrorInvalidValue**, **cudaErrorInvalidResourceHandle**, **cudaErrorUnknown**

Note:
Note that this function may also return error codes from previous, asynchronous launches.

See also:
cudaGraphicsResourceSetMapFlags

cudaError_t cudaGLSetGLDevice (int device)

Deprecated
This function is deprecated as of CUDA 5.0.

This function is deprecated and should no longer be used. It is no longer necessary to associate a CUDA device with an OpenGL context in order to achieve maximum interoperability performance.

Parameters:
device - Device to use for OpenGL interoperability

Returns:
cudaSuccess, **cudaErrorInvalidDevice**, **cudaErrorSetOnActiveProcess**

Note:
Note that this function may also return error codes from previous, asynchronous launches.

See also:
cudaGraphicsGLRegisterBuffer, **cudaGraphicsGLRegisterImage**

cudaError_t cudaGLUnmapBufferObject (GLuint bufObj)

Deprecated
This function is deprecated as of CUDA 3.0.

Unmaps the buffer object of ID `bufObj` for access by CUDA. When a buffer is unmapped, the base address returned by **cudaGLMapBufferObject()** is invalid and subsequent references to the address result in undefined behavior. The OpenGL context used to create the buffer, or another context from the same share group, must be bound to the current thread when this is called.

All streams in the current thread are synchronized with the current GL context.

Parameters:
bufObj - Buffer object to unmap

Returns:
cudaSuccess, **cudaErrorInvalidDevicePointer**, **cudaErrorUnmapBufferObjectFailed**

Note:
Note that this function may also return error codes from previous, asynchronous launches.

See also:
cudaGraphicsUnmapResources

cudaError_t cudaGLUnmapBufferObjectAsync (GLuint bufObj, cudaStream_t stream)

Deprecated
This function is deprecated as of CUDA 3.0.

Unmaps the buffer object of ID `bufObj` for access by CUDA. When a buffer is unmapped, the base address returned by **cudaGLMapBufferObject()** is invalid and subsequent references to the address result in undefined behavior. The OpenGL context used to create the buffer, or another context from the same share group, must be bound to the current thread when this is called.

Stream /p stream is synchronized with the current GL context.

Parameters:
bufObj - Buffer object to unmap
stream - Stream to synchronize

Returns:

cudaSuccess, **cudaErrorInvalidDevicePointer**, **cudaErrorUnmapBufferObjectFailed**

Note:
Note that this function may also return error codes from previous, asynchronous launches.

See also:
cudaGraphicsUnmapResources

cudaError_t cudaGLUnregisterBufferObject (GLuint bufObj)

Deprecated
This function is deprecated as of CUDA 3.0.

Unregisters the buffer object of ID `bufObj` for access by CUDA and releases any CUDA resources associated with the buffer. Once a buffer is unregistered, it may no longer be mapped by CUDA. The GL context used to create the buffer, or another context from the same share group, must be bound to the current thread when this is called.

Parameters:
bufObj - Buffer object to unregister

Returns:
cudaSuccess

Note:
Note that this function may also return error codes from previous, asynchronous launches.

See also:
cudaGraphicsUnregisterResource

Author

Generated automatically by Doxygen from the source code.