

NAME

VDPAU Interoperability –

Functions

CUresult cuGraphicsVDPAURegisterOutputSurface (CUgraphicsResource *pCudaResource, VdpOutputSurface vdpSurface, unsigned int flags)

Registers a VDPAU VdpOutputSurface object.

CUresult cuGraphicsVDPAURegisterVideoSurface (CUgraphicsResource *pCudaResource, VdpVideoSurface vdpSurface, unsigned int flags)

Registers a VDPAU VdpVideoSurface object.

CUresult cuVDPAUCtxCreate (CUcontext *pCtx, unsigned int flags, CUdevice device, VdpDevice vdpDevice, VdpGetProcAddress *vdpGetProcAddress)

Create a CUDA context for interoperability with VDPAU.

CUresult cuVDPAUGetDevice (CUdevice *pDevice, VdpDevice vdpDevice, VdpGetProcAddress *vdpGetProcAddress)

Gets the CUDA device associated with a VDPAU device.

Detailed Description

\brief VDPAU interoperability functions of the low-level CUDA driver API (cudaVDPAU.h)

This section describes the VDPAU interoperability functions of the low-level CUDA driver application programming interface.

Function Documentation

CUresult cuGraphicsVDPAURegisterOutputSurface (CUgraphicsResource * pCudaResource, VdpOutputSurface vdpSurface, unsigned int flags)

Registers the VdpOutputSurface specified by vdpSurface for access by CUDA. A handle to the registered object is returned as pCudaResource. The surface's intended usage is specified using flags, as follows:

- CU_GRAPHICS_MAP_RESOURCE_FLAGS_NONE: Specifies no hints about how this resource will be used. It is therefore assumed that this resource will be read from and written to by CUDA. This is the default value.
- CU_GRAPHICS_MAP_RESOURCE_FLAGS_READ_ONLY: Specifies that CUDA will not write to this resource.
- CU_GRAPHICS_MAP_RESOURCE_FLAGS_WRITE_DISCARD: Specifies that CUDA will not read from this resource and will write over the entire contents of the resource, so none of the data previously stored in the resource will be preserved.

The VdpOutputSurface is presented as an array of subresources that may be accessed using pointers returned by **cuGraphicsSubResourceGetMappedArray**. The exact number of valid arrayIndex values depends on the VDPAU surface format. The mapping is shown in the table below. mipLevel must be 0.

Parameters:

pCudaResource - Pointer to the returned object handle

vdpSurface - The VdpOutputSurface to be registered

flags - Map flags

Returns:

CUDA_SUCCESS, CUDA_ERROR_INVALID_HANDLE,
CUDA_ERROR_ALREADY_MAPPED, CUDA_ERROR_INVALID_CONTEXT,

Note:

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

See also:

cuCtxCreate, cuVDPAUCtxCreate, cuGraphicsVDPAURegisterVideoSurface,
cuGraphicsUnregisterResource, cuGraphicsResourceSetMapFlags,
cuGraphicsMapResources, cuGraphicsUnmapResources,
cuGraphicsSubResourceGetMappedArray, cuVDPAUGetDevice



CUresult cuGraphicsVDPAURegisterVideoSurface (CUgraphicsResource * pCudaResource, VdpVideoSurface vdpSurface, unsigned int flags)

Registers the VdpVideoSurface specified by *vdpSurface* for access by CUDA. A handle to the registered object is returned as *pCudaResource*. The surface's intended usage is specified using *flags*, as follows:

- CU_GRAPHICS_MAP_RESOURCE_FLAGS_NONE: Specifies no hints about how this resource will be used. It is therefore assumed that this resource will be read from and written to by CUDA. This is the default value.
- CU_GRAPHICS_MAP_RESOURCE_FLAGS_READ_ONLY: Specifies that CUDA will not write to this resource.
- CU_GRAPHICS_MAP_RESOURCE_FLAGS_WRITE_DISCARD: Specifies that CUDA will not read from this resource and will write over the entire contents of the resource, so none of the data previously stored in the resource will be preserved.

The VdpVideoSurface is presented as an array of subresources that may be accessed using pointers returned by **cuGraphicsSubResourceGetMappedArray**. The exact number of valid *arrayIndex* values depends on the VDPAU surface format. The mapping is shown in the table below. *mipLevel* must be 0.

Parameters:

pCudaResource - Pointer to the returned object handle
vdpSurface - The VdpVideoSurface to be registered
flags - Map flags

Returns:

CUDA_SUCCESS, CUDA_ERROR_INVALID_HANDLE,
CUDA_ERROR_ALREADY_MAPPED, CUDA_ERROR_INVALID_CONTEXT,

Note:

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

See also:

cuCtxCreate, cuVDPAUCtxCreate, cuGraphicsVDPAURegisterOutputSurface,
cuGraphicsUnregisterResource, cuGraphicsResourceSetMapFlags,
cuGraphicsMapResources, cuGraphicsUnmapResources,
cuGraphicsSubResourceGetMappedArray, cuVDPAUGetDevice

CUresult cuVDPAUCtxCreate (CUcontext * pCtx, unsigned int flags, CUdevice device, VdpDevice vdpDevice, VdpGetProcAddress * vdpGetProcAddress)

Creates a new CUDA context, initializes VDPAU interoperability, and associates the CUDA context with the calling thread. It must be called before performing any other VDPAU interoperability operations. It may fail if the needed VDPAU driver facilities are not available. For usage of the *flags* parameter, see **cuCtxCreate()**.

Parameters:

pCtx - Returned CUDA context
flags - Options for CUDA context creation
device - Device on which to create the context
vdpDevice - The VdpDevice to interop with
vdpGetProcAddress - VDPAU's VdpGetProcAddress function pointer

Returns:

CUDA_SUCCESS, CUDA_ERROR_DEINITIALIZED,
CUDA_ERROR_NOT_INITIALIZED, CUDA_ERROR_INVALID_CONTEXT,
CUDA_ERROR_INVALID_VALUE, CUDA_ERROR_OUT_OF_MEMORY

Note:

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

See also:

cuCtxCreate, cuGraphicsVDPAURegisterVideoSurface,
cuGraphicsVDPAURegisterOutputSurface, cuGraphicsUnregisterResource,
cuGraphicsResourceSetMapFlags, cuGraphicsMapResources, cuGraphicsUnmapResources,



cuGraphicsSubResourceGetMappedArray, cuVDPAUGetDevice

CUresult cuVDPAUGetDevice (CUdevice * pDevice, VdpDevice vdpDevice, VdpGetProcAddress * vdpGetProcAddress)

Returns in `*pDevice` the CUDA device associated with a `vdpDevice`, if applicable.

Parameters:

pDevice - Device associated with `vdpDevice`

vdpDevice - A `VdpDevice` handle

vdpGetProcAddress - `VDPAU`'s `VdpGetProcAddress` function pointer

Returns:

`CUDA_SUCCESS`, `CUDA_ERROR_DEINITIALIZED`,
`CUDA_ERROR_NOT_INITIALIZED`, `CUDA_ERROR_INVALID_CONTEXT`,
`CUDA_ERROR_INVALID_VALUE`

Note:

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

See also:

`cuCtxCreate`, `cuVDPAUCtxCreate`, `cuGraphicsVDPAURegisterVideoSurface`,
`cuGraphicsVDPAURegisterOutputSurface`, `cuGraphicsUnregisterResource`,
`cuGraphicsResourceSetMapFlags`, `cuGraphicsMapResources`, `cuGraphicsUnmapResources`,
`cuGraphicsSubResourceGetMappedArray`

Author

Generated automatically by Doxygen from the source code.

