

NAME

Execution Control [DEPRECATED] –

Functions

CUresult cuFuncSetBlockShape (CUfunction hfunc, int x, int y, int z)

Sets the block-dimensions for the function.

CUresult cuFuncSetSharedSize (CUfunction hfunc, unsigned int bytes)

Sets the dynamic shared-memory size for the function.

CUresult cuLaunch (CUfunction f)

Launches a CUDA function.

CUresult cuLaunchGrid (CUfunction f, int grid_width, int grid_height)

Launches a CUDA function.

CUresult cuLaunchGridAsync (CUfunction f, int grid_width, int grid_height, CUSTream hStream)

Launches a CUDA function.

CUresult cuParamSetf (CUfunction hfunc, int offset, float value)

Adds a floating-point parameter to the function's argument list.

CUresult cuParamSeti (CUfunction hfunc, int offset, unsigned int value)

Adds an integer parameter to the function's argument list.

CUresult cuParamSetSize (CUfunction hfunc, unsigned int numbytes)

Sets the parameter size for the function.

CUresult cuParamSetTexRef (CUfunction hfunc, int texunit, CUTexref hTexRef)

Adds a texture-reference to the function's argument list.

CUresult cuParamSetv (CUfunction hfunc, int offset, void *ptr, unsigned int numbytes)

Adds arbitrary data to the function's argument list.

Detailed Description

\brief deprecated execution control functions of the low-level CUDA driver API (**cuda.h**)

This section describes the deprecated execution control functions of the low-level CUDA driver application programming interface.

Function Documentation

CUresult cuFuncSetBlockShape (CUfunction hfunc, int x, int y, int z)

Deprecated

Specifies the x, y, and z dimensions of the thread blocks that are created when the kernel given by **hfunc** is launched.

Parameters:

hfunc - Kernel to specify dimensions of

x - X dimension

y - Y dimension

z - Z dimension

Returns:

**CUDA_SUCCESS, CUDA_ERROR_DEINITIALIZED,
CUDA_ERROR_NOT_INITIALIZED, CUDA_ERROR_INVALID_CONTEXT,
CUDA_ERROR_INVALID_HANDLE, CUDA_ERROR_INVALID_VALUE**

Note:

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

See also:

**cuFuncSetSharedSize, cuFuncSetCacheConfig, cuFuncGetAttribute, cuParamSetSize,
cuParamSeti, cuParamSetf, cuParamSetv, cuLaunch, cuLaunchGrid, cuLaunchGridAsync,
cuLaunchKernel**

CUresult cuFuncSetSharedSize (CUfunction hfunc, unsigned int bytes)

Deprecated

Sets through **bytes** the amount of dynamic shared memory that will be available to each thread block when the kernel given by **hfunc** is launched.

Parameters:

hfunc - Kernel to specify dynamic shared-memory size for

bytes - Dynamic shared-memory size per thread in bytes



Returns:

CUDA_SUCCESS, CUDA_ERROR_DEINITIALIZED,
 CUDA_ERROR_NOT_INITIALIZED, CUDA_ERROR_INVALID_CONTEXT,
 CUDA_ERROR_INVALID_HANDLE, CUDA_ERROR_INVALID_VALUE

Note:

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

See also:

cuFuncSetBlockShape, cuFuncSetCacheConfig, cuFuncGetAttribute, cuParamSetSize,
 cuParamSeti, cuParamSetf, cuParamSetv, cuLaunch, cuLaunchGrid, cuLaunchGridAsync,
 cuLaunchKernel

CUresult cuLaunch (CUfunction f)**Deprecated**

Invokes the kernel *f* on a 1 x 1 x 1 grid of blocks. The block contains the number of threads specified by a previous call to **cuFuncSetBlockShape()**.

Parameters:

f - Kernel to launch

Returns:

CUDA_SUCCESS, CUDA_ERROR_DEINITIALIZED,
 CUDA_ERROR_NOT_INITIALIZED, CUDA_ERROR_INVALID_CONTEXT,
 CUDA_ERROR_INVALID_VALUE, CUDA_ERROR_LAUNCH_FAILED,
 CUDA_ERROR_LAUNCH_OUT_OF_RESOURCES,
 CUDA_ERROR_LAUNCH_TIMEOUT,
 CUDA_ERROR_LAUNCH_INCOMPATIBLE_TEXTURING,
 CUDA_ERROR_SHARED_OBJECT_INIT_FAILED

Note:

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

See also:

cuFuncSetBlockShape, cuFuncSetSharedSize, cuFuncGetAttribute, cuParamSetSize,
 cuParamSetf, cuParamSeti, cuParamSetv, cuLaunchGrid, cuLaunchGridAsync,
 cuLaunchKernel

CUresult cuLaunchGrid (CUfunction f, int grid_width, int grid_height)**Deprecated**

Invokes the kernel *f* on a *grid_width* x *grid_height* grid of blocks. Each block contains the number of threads specified by a previous call to **cuFuncSetBlockShape()**.

Parameters:

f - Kernel to launch
grid_width - Width of grid in blocks
grid_height - Height of grid in blocks

Returns:

CUDA_SUCCESS, CUDA_ERROR_DEINITIALIZED,
 CUDA_ERROR_NOT_INITIALIZED, CUDA_ERROR_INVALID_CONTEXT,
 CUDA_ERROR_INVALID_VALUE, CUDA_ERROR_LAUNCH_FAILED,
 CUDA_ERROR_LAUNCH_OUT_OF_RESOURCES,
 CUDA_ERROR_LAUNCH_TIMEOUT,
 CUDA_ERROR_LAUNCH_INCOMPATIBLE_TEXTURING,
 CUDA_ERROR_SHARED_OBJECT_INIT_FAILED

Note:

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

See also:

cuFuncSetBlockShape, cuFuncSetSharedSize, cuFuncGetAttribute, cuParamSetSize,
 cuParamSetf, cuParamSeti, cuParamSetv, cuLaunch, cuLaunchGridAsync,
 cuLaunchKernel



CUresult cuLaunchGridAsync (CUfunction f, int grid_width, int grid_height, CUstream hStream)**Deprecated**

Invokes the kernel `f` on a `grid_width` x `grid_height` grid of blocks. Each block contains the number of threads specified by a previous call to `cuFuncSetBlockShape()`.

Parameters:

f - Kernel to launch
grid_width - Width of grid in blocks
grid_height - Height of grid in blocks
hStream - Stream identifier

Returns:

CUDA_SUCCESS, CUDA_ERROR_DEINITIALIZED,
 CUDA_ERROR_NOT_INITIALIZED, CUDA_ERROR_INVALID_CONTEXT,
 CUDA_ERROR_INVALID_HANDLE, CUDA_ERROR_INVALID_VALUE,
 CUDA_ERROR_LAUNCH_FAILED,
 CUDA_ERROR_LAUNCH_OUT_OF_RESOURCES,
 CUDA_ERROR_LAUNCH_TIMEOUT,
 CUDA_ERROR_LAUNCH_INCOMPATIBLE_TEXTURING,
 CUDA_ERROR_SHARED_OBJECT_INIT_FAILED

Note:

In certain cases where cubins are created with no ABI (i.e., using `ptxas -abi-compile no`), this function may serialize kernel launches. In order to force the CUDA driver to retain asynchronous behavior, set the `CU_CTX_LMEM_RESIZE_TO_MAX` flag during context creation (see `cuCtxCreate`).

This function uses standard semantics.

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

See also:

`cuFuncSetBlockShape`, `cuFuncSetSharedSize`, `cuFuncGetAttribute`, `cuParamSetSize`,
`cuParamSetf`, `cuParamSeti`, `cuParamSetv`, `cuLaunch`, `cuLaunchGrid`, `cuLaunchKernel`

CUresult cuParamSetf (CUfunction hfunc, int offset, float value)**Deprecated**

Sets a floating-point parameter that will be specified the next time the kernel corresponding to `hfunc` will be invoked. `offset` is a byte offset.

Parameters:

hfunc - Kernel to add parameter to
offset - Offset to add parameter to argument list
value - Value of parameter

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:

`cuFuncSetBlockShape`, `cuFuncSetSharedSize`, `cuFuncGetAttribute`, `cuParamSetSize`,
`cuParamSeti`, `cuParamSetv`, `cuLaunch`, `cuLaunchGrid`, `cuLaunchGridAsync`,
`cuLaunchKernel`

CUresult cuParamSeti (CUfunction hfunc, int offset, unsigned int value)**Deprecated**

Sets an integer parameter that will be specified the next time the kernel corresponding to `hfunc` will be invoked. `offset` is a byte offset.

Parameters:

hfunc - Kernel to add parameter to



offset - Offset to add parameter to argument list

value - Value of parameter

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:

cuFuncSetBlockShape, cuFuncSetSharedSize, cuFuncGetAttribute, cuParamSetSize,
cuParamSetf, cuParamSetv, cuLaunch, cuLaunchGrid, cuLaunchGridAsync,
cuLaunchKernel

CUresult cuParamSetSize (CUfunction hfunc, unsigned int numbytes)

Deprecated

Sets through *numbytes* the total size in bytes needed by the function parameters of the kernel corresponding to *hfunc*.

Parameters:

hfunc - Kernel to set parameter size for

numbytes - Size of parameter list in bytes

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:

cuFuncSetBlockShape, cuFuncSetSharedSize, cuFuncGetAttribute, cuParamSetf,
cuParamSeti, cuParamSetv, cuLaunch, cuLaunchGrid, cuLaunchGridAsync,
cuLaunchKernel

CUresult cuParamSetTexRef (CUfunction hfunc, int texunit, CUTexref hTexRef)

Deprecated

Makes the CUDA array or linear memory bound to the texture reference *hTexRef* available to a device program as a texture. In this version of CUDA, the texture-reference must be obtained via **cuModuleGetTexRef()** and the *texunit* parameter must be set to **CU_PARAM_TR_DEFAULT**.

Parameters:

hfunc - Kernel to add texture-reference to

texunit - Texture unit (must be **CU_PARAM_TR_DEFAULT**)

hTexRef - Texture-reference to add to argument list

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.

CUresult cuParamSetv (CUfunction hfunc, int offset, void * ptr, unsigned int numbytes)

Deprecated

Copies an arbitrary amount of data (specified in *numbytes*) from *ptr* into the parameter space of the kernel corresponding to *hfunc*. *offset* is a byte offset.

Parameters:

hfunc - Kernel to add data to

offset - Offset to add data to argument list

ptr - Pointer to arbitrary data



numbytes - Size of data to copy in bytes

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:

**cuFuncSetBlockShape, cuFuncSetSharedSize, cuFuncGetAttribute, cuParamSetSize,
cuParamSetf, cuParamSeti, cuLaunch, cuLaunchGrid, cuLaunchGridAsync,
cuLaunchKernel**

Author

Generated automatically by Doxygen from the source code.

