NAME

ddi_device_zero – zero fill the device

SYNOPSIS

#include <sys/ddi.h>
#include <sys/sunddi.h>

int ddi_device_zero(ddi_acc_handle_t *handle*, **caddr_t** *dev_addr*, **size_t** *bytecount*, **long** *dev_advcnt*, **ulong_t** *dev_datasz***);**

INTERFACE LEVEL

Solaris DDI specific (Solaris DDI).

ARGUMENTS

handle The data access handle returned from setup calls, such as **ddi_regs_map_setup**(9F).

dev_addr Beginning of the device address.

bytecount Number of bytes to zero.

dev_advcnt Number of *dev_datasz* units to advance on every access.

dev_datasz The size of each data word. Possible values are defined as:

DDI_DATA_SZ01_ACC
1 byte data size

DDI_DATA_SZ02_ACC
2 bytes data size

DDI_DATA_SZ04_ACC
4 bytes data size

DDI_DATA_SZ08_ACC
8 bytes data size

DESCRIPTION

ddi_device_zero( ) function fills the given, *bytecount,* number of byte of zeroes to the device register or memory.

The *dev_advcnt* argument determines the value of the device address, *dev_addr*, on each access. A value of **0** will use the same device address, *dev_addr*, on every access. A positive value increments the device address in the next access while a negative value decrements the address. The device address is incremented/decremented in *dev_datasz* units.

The *dev_datasz* argument determines the size of data word on each access.

RETURN VALUES

ddi_device_zero( ) returns:

DDI_SUCCESS Successfully zeroed the data.

DDI_FAILURE The byte count is not a multiple of *dev_datasz*.

CONTEXT

ddi_device_zero( ) can be called from user, kernel, or interrupt context.

SEE ALSO

ddi_regs_map_setup(9F), **ddi_regs_map_free**(9F)

Writing Device Drivers