NAME

ddi_dma_movwin – shift current DMA window

SYNOPSIS

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

int ddi_dma_movwin(ddi_dma_handle_t *handle***, off_t** **offp***, u_int** **lenp***, ddi_dma_cookie_t** **cookiep***);**

INTERFACE LEVEL

Solaris SPARC DDI specific (Solaris SPARC DDI).

ARGUMENTS

handle The DMA handle filled in by a call to **ddi_dma_setup**(9F).

offp A pointer to an offset to set the DMA window to. Upon a successful return, it will be filled in with the new offset from the beginning of the object resources are allocated for.

lenp A pointer to a value which must either be the current size of the DMA window (as known from a call to **ddi_dma_curwin**(9F) or from a previous call to **ddi_dma_movwin()**). Upon a successful return, it will be filled in with the size, in bytes, of the current window.

cookiep A pointer to a DMA cookie (see **ddi_dma_cookie**(9S)). Upon a successful return, cookiep is filled in just as if an implicit **ddi_dma_htoc**(9F) had been made.

DESCRIPTION

ddi_dma_movwin( ) shifts the current DMA window. If a DMA request allows the sytem to allocate resources for less than the entire object by setting the **DDI_DMA_PARTIAL** flag in the **ddi_dma_req**(9S) structure, the current DMA window can be shifted by a call to **ddi_dma_movwin( )**.

The caller must first determine the current DMA window size by a call to **ddi_dma_curwin**(9F). Using the current offset and size of the window thus retrieved, the caller of **ddi_dma_movwin( )** may change the window onto the object by changing the offset by a value which is some multiple of the size of the DMA window.

ddi_dma_movwin( ) takes care of underlying resource synchronizations required to *shift* the window. However if you want to *access* the data prior or after moving the window, further synchronizations using **ddi_dma_sync**(9F) are required,

This function is normally called from an interrupt routine. The first invocation of the DMA engine is done from the driver. All subsequent invocations of the DMA engine are done from the interrupt routine. The interrupt routine checks to see if the request has been completed. If it has, it returns without invoking another DMA transfer. Otherwise it calls **ddi_dma_movwin( )** to shift the current window and starts another DMA transfer.

RETURN VALUES

ddi_dma_movwin( ) returns:

DDI_SUCCESS The current length and offset are legal and have been set.

DDI_FAILURE Otherwise.

CONTEXT

ddi_dma_movwin( ) can be called from user or interrupt context.

SEE ALSO

ddi_dma_curwin(9F), **ddi_dma_htoc**(9F), **ddi_dma_setup**(9F), **ddi_dma_sync**(9F), **ddi_dma_cookie**(9S), **ddi_dma_req**(9S)

Writing Device Drivers

WARNINGS

The caller must guarantee that the resources used by the object are inactive prior to calling this function.