NAME

iflibdd - Device Dependent Configuration Functions

SYNOPSIS

#include <ifdi_if.h>

Soft Queue Setup and Teardown Functions

Mandatory Functions

int
ifdi_tx_queues_alloc(*if_ctx_t ctx, caddr_t *vaddrs, uint64_t *paddrs, int ntxqs, int ntxqsets*);

int
ifdi_rx_queues_alloc(*if_ctx_t ctx, caddr_t *vaddrs, uint64_t *paddrs, int nrxqs, int nrxqsets*);

int
ifdi_queues_free(*if_ctx_t ctx*);

Optional Functions

int
ifdi_txq_setup(*if_ctx_t ctx, uint16_t qid*);

int
ifdi_rxq_setup(*if_ctx_t ctx, uint16_t qid*);

Device Setup and Teardown Functions

Mandatory Functions

int
ifdi_attach_pre(*if_ctx_t ctx*);

int
ifdi_attach_post(*if_ctx_t ctx*);

int
ifdi_detach(*if_ctx_t ctx*);

Optional Functions

void
ifdi_vlan_register(*if_ctx_t ctx, uint16_t vtag*);

void
ifdi_vlan_unregister(*if_ctx_t ctx, uint16_t vtag*);

int
ifdi_suspend(*if_ctx_t ctx*);

int
ifdi_resume(*if_ctx_t ctx*);

Device Configuration Functions

Mandatory Functions

void
ifdi_init(*if_ctx_t ctx*);

void
ifdi_stop(*if_ctx_t ctx*);

void
ifdi_multi_set(*if_ctx_t ctx*);

int
ifdi_mtu_set(*if_ctx_t ctx, uint32_t mtu*);

void
ifdi_media_status(*if_ctx_t ctx, struct ifmediareq *ifr*);

int
ifdi_media_change(*if_ctx_t ctx*);

void
ifdi_promisc_set(*if_ctx_t ctx, int flags*);

uint64_t
ifdi_get_counter(*if_ctx_t ctx, ift_counter cnt*);

void
ifdi_update_admin_status(*if_ctx_t ctx*);

Optional Functions

void
ifdi_media_set(*if_ctx_t ctx*);

Interrupt enable/disable

Mandatory Functions
void
ifdi_intr_enable(*if_ctx_t ctx*);

void
ifdi_queue_intr_enable(*if_ctx_t ctx, uint16_t qid*);

void
ifdi_intr_disable(*if_ctx_t ctx*);

IOV Support
init
iov_init(*if_ctx_t ctx, uint16_t num_vfs, const nvlist_t *params*);

void
iov_uinit(*if_ctx_t ctx*);

void
ifdi_vflr_handle(*if_ctx_t ctx*);

int
ifdi_vf_add(*if_ctx_t ctx, uint16_t vfnum, const nvlist_t *params*);

Optional Functions
void
ifdi_link_intr_enable(*if_ctx_t ctx*);

Optional Service Routines
void
ifdi_timer(*if_ctx_t ctx*);

void
ifdi_watchdog_reset(*if_ctx_t ctx*);

Additional Functions

void
ifdi_led_func(*if_ctx_t ctx, int onoff*);

int
ifdi_sysctl_int_delay(*if_ctx_t ctx, if_int_delay_info_t iidi*);

int
ifdi_i2c_req(*if_ctx_t ctx, struct ifi2creq *req*);

FUNCTIONS

The above named functions are device dependent configuration functions. These routines are registered with iflib by the driver and are called from the corresponding iflib function to configure device specific functions and registers.

Device Dependent Functions

Soft Queue Setup and Teardown

ifdi_tx_queues_alloc()
Mandatory function that is called during iflib_attach to allocate transmit queues. vaddrs and paddrs are arrays of virtual and physical addresses respectively of the hardware transmit queues. ntxqs is the number of queues per qset. ntxqsets is the number of qsets.

ifdi_rx_queues_alloc()
Mandatory function that is called during iflib_attach to allocate receive queues. vaddrs and paddrs are arrays of virtual and physical addresses respectively of the hardware receive queues. nrxqs is the number of queues per qset. nrxqsets is the number of qsets.

ifdi_queues_free()
Mandatory function that frees the allocated queues and associated transmit buffers.

ifdi_txq_setup()
Optional function for each transmit queue that handles device specific initialization.

ifdi_rxq_setup()
Optional function for each receive queue that handles device specific initialization.

Device Setup and Teardown

ifdi_attach_pre()
Mandatory function implemented by the driver to perform any attach logic that procedes interrupt and queue allocation, queue setup, and interrupt assignment.

ifdi_attach_post()
Mandatory function implemented by the driver to perform any attach logic that occurs after ifdi_attach_pre, and iflib's queue setup and MSI/MSIX(X) or legacy interrupt assignment.

ifdi_detach()
Mandatory function that frees any resources allocated by the driver in ifdi_attach_pre and ifdi_attach_post.

ifdi_vlan_register()
Optional function called by the VLAN config eventhandler. *vtag* is the new VLAN tag.

ifdi_vlan_unregister()
Optional function called by the VLAN unconfig eventhandler.

ifdi_suspend()
Optional function that suspends the driver.

ifdi_resume()
Optional function that resumes a driver.

Device Configuration Functions

ifdi_init()
Mandatory function that will initialize and bring up the hardware. For example, it will reset the chip and enable the receiver unit. It should mark the interface running, but not active (IFF_DRV_RUNNING, ~IIF_DRV_OACTIVE).

ifdi_stop()
Mandatory function that should disable all traffic on the interface by issuing a global reset on the MAC and deallocating the TX and RX buffers.

ifdi_multi_set()
Programs the interfaces multicast addresses

ifdi_media_status()
Media Ioctl Callback. Function is called whenever the user queries the status of the interface using ifconfig(8). The driver sets the appropriate link type and speed in ifmr->ifm_active.

ifdi_mtu_set()
Sets the mtu interface to the value of the second function parameter mtu.

ifdi_media_change()
Function is called when the user changes speed/duplex using the media/mediaopt option with ifconfig(8).

ifdi_promisc_set()
Enables or disables promisc settings depending upon the flags value. *flags* contains the interface's ifnet(9) flags.

ifdi_get_counter()
Returns the value for counter cnt depending upon counter type.

ifdi_update_admin_status()
Sets the link_up state to TRUE or FALSE depending upon the OS link state. A real check of the hardware only happens with a link interrupt.

ifdi_media_set()
Need to define

Interrupt Enable/Disable

ifdi_intr_enable()
Mandatory function that enables all interrupts.

ifdi_intr_disable()
Mandatory function that disables all interrupts.

ifdi_queue_intr_enable()
Mandatory function that enables interrupts on queue qid.

iov_init()
Initialize num_vfs VFs.

io_uninit()
Tear down the context for all VFs.

ifdi_vflr_handle()
Handle any VFs that have reset themselves via a Function Level Reset (FLR).

ifdi_vf_add()
Set parameters in params in VF vfnum.

Service Routines

ifdi_timer()
Optional timer routine that will be run every 500ms.

ifdi_watchdog_reset()
Optional function to run when a transmit queue is hung.

Additional Functions

ifdi_led_func()

ifdi_sysctl_int_delay()

ifdi_i2c_req()

SEE ALSO

ifconfig(8), iflibdi(9), iflibtxrx(9), ifnet(9)

AUTHORS

This manual page was written by Nicole Graziano