

**NAME**

**igc** - Intel Ethernet Controller I225 driver

**SYNOPSIS**

To compile this driver into the kernel, place the following lines in your kernel configuration file:

```
device iflib  
device igc
```

Alternatively, to load the driver as a module at boot time, place the following line in loader.conf(5):

```
if_igc_load="YES"
```

**DESCRIPTION**

The **igc** driver provides support for any PCI Express adapter or LOM (LAN On Motherboard) based on the Intel I225 Multi Gigabit Controller. The driver supports Transmit/Receive checksum offload, Jumbo Frames, MSI/MSI-X, TSO, and RSS.

Support for Jumbo Frames is provided via the interface MTU setting. Selecting an MTU larger than 1500 bytes with the `ifconfig(8)` utility configures the adapter to receive and transmit Jumbo Frames. The maximum MTU size for Jumbo Frames is 9216 bytes.

This driver version supports VLAN hardware insertion / extraction, and VLAN checksum offload. For information on enabling VLANs, see `ifconfig(8)`. The **igc** driver supports the following media types:

**autoselect** Enables auto-negotiation for speed and duplex.

**10baseT/UTP** Sets 10Mbps operation. Use the **mediaopt** option to select **half-duplex** mode.

**100baseTX** Sets 100Mbps operation. Use the **mediaopt** option to select **half-duplex** mode.

**1000baseT** Sets 1000Mbps operation. Only **full-duplex** mode is supported at this speed.

**2500baseT** Sets 2500Mbps operation. Only **full-duplex** mode is supported at this speed.

**HARDWARE**

The **igc** driver supports the following models:

- ⊕ I225-LM
- ⊕ I225-V

- I225-IT
- I225-K

## LOADER TUNABLES

Tunables can be set at the loader(8) prompt before booting the kernel or stored in loader.conf(5).

### *hw.igc.igc\_disable\_crc\_stripping*

Disable or enable hardware stripping of CRC field. This is mostly useful on BMC/IPMI shared interfaces where stripping the CRC causes remote access over IPMI to fail. Default 0 (enabled).

### *hw.igc.rx\_int\_delay*

This value delays the generation of receive interrupts in units of 1.024 microseconds. The default value is 0, since adapters may hang with this feature being enabled.

### *hw.igc.rx\_abs\_int\_delay*

If *hw.igc.rx\_int\_delay* is non-zero, this tunable limits the maximum delay in which a receive interrupt is generated.

### *hw.igc.tx\_int\_delay*

This value delays the generation of transmit interrupts in units of 1.024 microseconds. The default value is 64.

### *hw.igc.tx\_abs\_int\_delay*

If *hw.igc.tx\_int\_delay* is non-zero, this tunable limits the maximum delay in which a transmit interrupt is generated.

### *hw.igc.sbp*

Show bad packets when in promiscuous mode. Default is false.

### *hw.igc.rx\_process\_limit*

Maximum number of received packets to process at a time. Default is 100. A value of -1 means unlimited.

### *hw.igc.eee\_setting*

Disable or enable Energy Efficient Ethernet. Default 1 (disabled).

### *hw.igc.max\_interrupt\_rate*

Maximum device interrupts per second. The default is 8000.

## DIAGNOSTICS

**igc%d: Hardware Initialization Failed** A fatal initialization error has occurred.

**igc%d: Unable to allocate bus resource: memory** A fatal initialization error has occurred.

**igc%d: Invalid MAC address** The MAC address programmed into the EEPROM is either empty or a multicast/broadcast address.

#### SEE ALSO

altq(4), arp(4), iflib(4), netintro(4), ng\_ether(4), vlan(4), ifconfig(8)

#### HISTORY

The **igc** device driver first appeared in FreeBSD 14.0.

#### AUTHORS

The **igc** was originally written by Intel Corporation and converted to the iflib(4) framework by Netgate.