NAME

vge - VIA Networking Technologies Velocity Gigabit Ethernet adapter driver

SYNOPSIS

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

```
device miibus
device vge
```

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

```
if_vge_load="YES"
```

DESCRIPTION

The **vge** driver provides support for various NICs and embedded Ethernet interfaces based on the VIA Technologies VT6120, VT6122, VT6130 and VT6132 Velocity Family Gigabit Ethernet controller chips.

The VT6120/VT6122 is a 33/66MHz 64-bit PCI device which combines a tri-speed MAC with an integrated 10/100/1000 copper PHY. (Some older cards use an external PHY.) The VT6130/VT6132 is the PCI express version of Velocity family. The MAC supports TCP/IP hardware checksums (IPv4 only), TCP large send, VLAN tag insertion and stripping, as well as VLAN filtering, a 64-entry CAM filter and a 64-entry VLAN filter, 64-bit multicast hash filter, 4 separate transmit DMA queues, flow control and jumbo frames (not on VT6130/VT6132) up to 16K in size. The Velocity family controllers have a 16K receive FIFO and 48K transmit FIFO.

The **vge** driver takes advantage of the controller's checksum offload and VLAN tagging features, as well as the jumbo frame (except VT6130/VT6132) and CAM filter support. The CAM filter is used for multicast address filtering to provide 64 perfect multicast address filter support. If it is necessary for the interface to join more than 64 multicast groups, the driver will switch over to using the hash filter.

The jumbo frame support can be enabled by setting the interface MTU to any value larger than the default of 1500 bytes, up to a maximum of 9000 bytes. Jumbo frames are disabled on the VT6130/VT6132 controllers because the TX MAC will hang when trying to send a frame that is larger than 4K. The receive and transmit checksum offload support can be toggled on and off using the ifconfig(8) utility.

The **vge** driver supports the following media types:

autoselect Enable autoselection of the media type and options. The user can manually override the

autoselected mode by adding media options to rc.conf(5).

10baseT/UTP Set 10Mbps operation. The ifconfig(8) **mediaopt** option can also be used to select either **full-duplex** or **half-duplex** modes.

100baseTX Set 100Mbps (Fast Ethernet) operation. The ifconfig(8) **mediaopt** option can also be used to select either **full-duplex** or **half-duplex** modes.

1000baseTX Set 1000baseTX operation over twisted pair. The ifconfig(8) **mediaopt** option can also be used to select either **full-duplex** or **half-duplex** modes.

The **vge** driver supports the following media options:

full-duplex Force full duplex operation.

half-duplex
Force half duplex operation.

For more information on configuring this device, see ifconfig(8).

HARDWARE

The **vge** driver supports VIA Networking VT6120, VT6122, VT6130 and VT6132 based Gigabit Ethernet adapters including:

- VIA Networking LAN-on-motherboard Gigabit Ethernet
- ZyXEL GN650-T 64-bit PCI Gigabit Ethernet NIC (ZX1701)
- ZyXEL GN670-T 32-bit PCI Gigabit Ethernet NIC (ZX1702)

LOADER TUNABLES

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

hw.vge.msi_disable
This tunable disables MSI support on the Ethernet hardware. The default value is 0.

SYSCTL VARIABLES

The following variables are available as both sysctl(8) variables and loader(8) tunables:

dev.vge.%d.int_holdoff
Maximum number of time to delay interrupts. The valid range is 0 to 5100 in units of 1us, the default is 150 (150us). The resolution of timer is about 20us so finer tuning than 20us wouldn’t

be available. The interface should be brought down and up again before a change takes effect.

dev.vge.%d.rx_coal_pkt
Maximum number of packets to fire Rx completion interrupt. The valid range is 1 to 255, the default is 64.

dev.vge.%d.tx_coal_pkt
Maximum number of packets to fire Tx completion interrupt. The valid range is 1 to 255, the default is 128.

DIAGNOSTICS

vge%d: couldn't map memory A fatal initialization error has occurred.

vge%d: couldn't map ports A fatal initialization error has occurred.

vge%d: couldn't map interrupt A fatal initialization error has occurred.

vge%d: failed to enable memory mapping! The driver failed to initialize PCI shared memory mapping. This might happen if the card is not in a bus-master slot.

vge%d: watchdog timeout The device has stopped responding to the network, or there is a problem with the network connection (cable).

SEE ALSO

altq(4), arp(4), miibus(4), netintro(4), ng_ether(4), polling(4), vlan(4), ifconfig(8)

HISTORY

The **vge** device driver first appeared in FreeBSD 5.3.

AUTHORS

The **vge** driver was written by Bill Paul <*wpaul@windriver.com*>.