

Neterion's (Formerly S2io) Xframe I/II PCI-X 10GbE driver

Release notes for Neterion's (Formerly S2io) Xframe I/II PCI-X 10GbE driver.

1. Introduction

This Linux driver supports Neterion's Xframe I PCI-X 1.0 and Xframe II PCI-X 2.0 adapters. It supports several features such as jumbo frames, MSI/MSI-X, checksum offloads, TSO, UFO and so on. See below for complete list of features.

All features are supported for both IPv4 and IPv6.

2. Identifying the adapter/interface

- a. Insert the adapter(s) in your system.
- b. Build and load driver:

```
# insmod s2io.ko
```

- c. View log messages:

```
# dmesg | tail -40
```

You will see messages similar to:

```
eth3: Neterion Xframe I 10GbE adapter (rev 3), Version 2.0.9.1, Intr type INTA
eth4: Neterion Xframe II 10GbE adapter (rev 2), Version 2.0.9.1, Intr type INTA
eth4: Device is on 64 bit 133MHz PCIX(M1) bus
```

The above messages identify the adapter type(Xframe I/II), adapter revision, driver version, interface name(eth3, eth4), Interrupt type(INTA, MSI, MSI-X). In case of Xframe II, the PCI/PCI-X bus width and frequency are displayed as well.

To associate an interface with a physical adapter use "ethtool -p <ethX>". The corresponding adapter's LED will blink multiple times.

3. Features supported

- a. Jumbo frames. Xframe I/II supports MTU up to 9600 bytes, modifiable using ip command.
- b. Offloads. Supports checksum offload(TCP/UDP/IP) on transmit and receive, TSO.
- c. Multi-buffer receive mode. Scattering of packet across multiple buffers. Currently driver supports 2-buffer mode which yields significant performance improvement on certain platforms(SGI Altix, IBM xSeries).
- d. MSI/MSI-X. Can be enabled on platforms which support this feature (IA64, Xeon) resulting in noticeable performance improvement(up to 7% on certain platforms).
- e. Statistics. Comprehensive MAC-level and software statistics displayed using "ethtool -S" option.
- f. Multi-FIFO/Ring. Supports up to 8 transmit queues and receive rings, with multiple steering options.

4. Command line parameters

- a. tx_fifo_num
Number of transmit queues

Valid range: 1-8

Default: 1

- b. rx_ring_num
Number of receive rings

Valid range: 1-8

Default: 1

- c. tx_fifo_len
Size of each transmit queue

Valid range: Total length of all queues should not exceed 8192

Default: 4096

- d. rx_ring_sz
Size of each receive ring(in 4K blocks)

Valid range: Limited by memory on system

Default: 30

- e. `intr_type`

Specifies interrupt type. Possible values 0(INTA), 2(MSI-X)

Valid values: 0, 2

Default: 2

5. Performance suggestions

General:

- a. Set MTU to maximum(9000 for switch setup, 9600 in back-to-back configuration)
- b. Set TCP windows size to optimal value.

For instance, for MTU=1500 a value of 210K has been observed to result in good performance:

```
# sysctl -w net.ipv4.tcp_rmem="210000 210000 210000"
# sysctl -w net.ipv4.tcp_wmem="210000 210000 210000"
```

For MTU=9000, TCP window size of 10 MB is recommended:

```
# sysctl -w net.ipv4.tcp_rmem="10000000 10000000 10000000"
# sysctl -w net.ipv4.tcp_wmem="10000000 10000000 10000000"
```

Transmit performance:

- a. By default, the driver respects BIOS settings for PCI bus parameters. However, you may want to experiment with PCI bus parameters `max-split-transactions(MOST)` and `MMRBC` (use `setpci` command).

A MOST value of 2 has been found optimal for Opterons and 3 for Itanium.

It could be different for your hardware.

Set MMRBC to 4K**.

For example you can set

For opteron:

```
#setpci -d 17d5:* 62=1d
```

For Itanium:

```
#setpci -d 17d5:* 62=3d
```

For detailed description of the PCI registers, please see Xframe User Guide.

- b. Ensure Transmit Checksum offload is enabled. Use `ethtool` to set/verify this parameter.
- c. Turn on TSO(using "`ethtool -K`");

```
# ethtool -K <ethX> tso on
```

Receive performance:

- a. By default, the driver respects BIOS settings for PCI bus parameters. However, you may want to set PCI latency timer to 248:

```
#setpci -d 17d5:* LATENCY_TIMER=f8
```

For detailed description of the PCI registers, please see Xframe User Guide.

- b. Use 2-buffer mode. This results in large performance boost on certain platforms(eg. SGI Altix, IBM xSeries).
- c. Ensure Receive Checksum offload is enabled. Use "`ethtool -K ethX`" command to set/verify this option.
- d. Enable NAPI feature(in kernel configuration Device Drivers ---> Network device support ---> Ethernet (10000 Mbit) ---> S2IO 10Gbe Xframe NIC) to bring down CPU utilization.

Note

For AMD opteron platforms with 8131 chipset, MMRBC=1 and MOST=1 are recommended as safe parameters.

For more information, please review the AMD8131 errata at http://vip.amd.com/us-en/assets/content_type/white_papers_and_tech_docs/26310_AMD-8131_HyperTransport_PCI-X_Tunnel_Revision_Guide_rev_3_18.pdf

6. Support

For further support please contact either your 10GbE Xframe NIC vendor (IBM, HP, SGI etc.)