3COM PCI TOKEN LINK VELOCITY XL TOKEN RING CARDS README

Release 0.9.0 - Release
Jul 17th 2000 Mike Phillips

1.2.0 - Final
Feb 17th 2002 Mike Phillips
Updated for submission to the 2.4.x kernel.

Thanks:
Terry Murphy from 3Com for tech docs and support,
Adam D. Ligas for testing the driver.

Note:
This driver will NOT work with the 3C339 Token Ring cards, you need to use the tms380 driver instead.

Options:

The driver accepts three options: ringspeed, pkt_buf_sz and message_level.

These options can be specified differently for each card found.

ringspeed: Has one of three settings 0 (default), 4 or 16. 0 will make the card autosense the ringspeed and join at the appropriate speed, this will be the default option for most people. 4 or 16 allow you to explicitly force the card to operate at a certain speed. The card will fail if you try to insert it at the wrong speed. (Although some hubs will allow this so be *very* careful). The main purpose for explicitly setting the ring speed is for when the card is first on the ring. In autosense mode, if the card cannot detect any active monitors on the ring it will open at the same speed as its last opening. This can be hazardous if this speed does not match the speed you want the ring to operate at.

pkt_buf_sz: This is this initial receive buffer allocation size. This will default to 4096 if no value is entered. You may increase performance of the driver by setting this to a value larger than the network packet size, although the driver now re-sizes buffers based on MTU settings as well.

message_level: Controls level of messages created by the driver. Defaults to 0: which only displays start-up and critical messages. Presently any non-zero value will display all soft messages as well. NB This does not turn debugging messages on, that must be done by modified the source code.

Variable MTU size:

The driver can handle a MTU size upto either 4500 or 18000 depending upon ring speed. The driver also changes the size of the receive buffers as part of the mtu re-sizing, so if you set mtu = 18000, you will need to be able to allocate 16 * (sk_buff with 18000 buffer size) call it 18500 bytes per ring position = 296,000 bytes of memory space, plus of course anything necessary for the tx sk_buff's. Remember this is per card, so if you are building routers, gateway's etc, you could start to use a lot of memory real fast.

2/17/02 Mike Phillips