NAME

tr – IBM 16/4 Token Ring Network Adapter device driver

SYNOPSIS

```
#include <sys/stropts.h>
#include <sys/ethernet.h>
#include <sys/dlpi.h>
```

AVAILABILITY

x86

DESCRIPTION

The **tr** token ring driver is a multi-threaded, loadable, clonable, STREAMS hardware driver supporting the connectionless Data Link Provider Interface, **dlpi**(7P), over IBM 16/4 Token Ring adapters. The driver supports installation of both a primary and secondary 16/4 Adapter within the system. The **tr** driver provides basic support for the IBM 16/4 Adapter hardware. Functions include chip initialization, frame transmit and receive, functional addresses, and "promiscuous" support, and error recovery and reporting.

The cloning, character-special device **/dev/tr** is used to access all 16/4 adapter devices installed within the system.

The **tr** driver is a "style 2" Data Link Service provider. All **M_PROTO** and **M_PCPROTO** type messages are interpreted as **DLPI** primitives. An explicit **DL_ATTACH_REQ** message by the user is required to associate the opened stream with a particular device (**ppa**). The **ppa** ID is interpreted as an unsigned long integer and indicates the corresponding device instance (unit) number. The unit numbers are assigned sequentially to each board found. The search order is determined by the order defined in the **/kernel/drv/tr.conf** file. An error (**DL_ERROR_ACK**) is returned by the driver if the **ppa** field value does not correspond to a valid device instance number for this system. The device is initialized on first attach and de-initialized (stopped) on last detach.

The values returned by the driver in the **DL_INFO_ACK** primitive in response to the **DL_INFO_REQ** from the user are as follows:

- The maximum SDU is **4084**.
- The minimum SDU is **0**.
- The **dlsap** address length is **7** or **8** bytes.
- The MAC type is **DL_TPR**.
- The **sap** length value is **–1** or **–2**, meaning the physical address component is followed immediately by a 1 or 2-byte **sap** component within the **DLSAP** address.
- The service mode is **DL_CLDLS**.
- No optional quality of service (QOS) support is included at present, so the QOS fields are **0**.
- The provider style is **DL_STYLE2**.
- The version is **DL_VERSION_2**.
- The broadcast address value is the IEEE broadcast address (FF:FF:FF:FF:FF:FF).

The token ring broadcast address (C0:00:FF:FF:FF:FF) is also supported.

Once in the **DL_ATTACHED** state, the user must send a **DL_BIND_REQ** to associate a particular Service Access Pointer (SAP) with the stream. The **tr** driver interprets the **sap** field within the **DL_BIND_REQ** as an IEEE 802.2 **sap**; therefore valid values for the **sap** field are in the **[0-0xFF]** range, of which only even values are legal.

In addition to 802.2 service, a "SNAP mode" is also provided by the driver. In this mode, **sap** values in the range **[0x5de-0xffff]** are used to indicate a request to use "SNAP" mode.

The **tr** driver **DLSAP** address format consists of the 6-byte physical token ring address component followed immediately by the 1 or 2-byte **sap** component, producing a 7 or 8-byte **DLSAP** address. Applications should *not* hardcode to this particular implementation-specific **DLSAP** address format, but should instead use information returned by the **DL_INFO_ACK** primitive to compose and decompose **DLSAP** addresses. The **sap** length, full **DLSAP** length, and **sap**/physical ordering are included within the **DL_INFO_ACK**. The physical address length can be computed by subtracting the **sap** length from the full **DLSAP** address length or by issuing the **DL_PHYS_ADDR_REQ** to obtain the current physical address associated with the stream.

Once in the **DL_BOUND** state, the user may transmit frames on the token ring by sending **DL_UNITDATA_REQ** messages to the **tr** driver. The **tr** driver will route received token ring frames up all open and bound streams that have a **sap** which matches the sap in the **DL_UNITDATA_IND** messages. Received token ring frames are duplicated and routed up multiple open streams if necessary. The **DLSAP** address contained within the **DL_UNITDATA_REQ** and **DL_UNITDATA_IND** messages consists of both the **sap** and physical (token ring) components.

tr Primitives

In addition to the mandatory connectionless **DLPI** message set, the driver also supports the following primitives:

The **DL_ENABMULTI_REQ** and **DL_DISABMULTI_REQ** primitives enable/disable reception of individual multicast group addresses. A set of multicast addresses may be iteratively created and modified on a per-stream basis using these primitives. These primitives are accepted by the driver in any state following **DL_ATTACHED**.

The **DL_PROMISCON_REQ** and **DL_PROMISCOFF_REQ** primitives with the **DL_PROMISC_PHYS** flag set in the **dl_level** field is currently unsupported for this driver.

When used with the **DL_PROMISC_SAP** flag set, this enables/disables reception of all **sap** values. When used with the **DL_PROMISC_MULTI** flag set, this enables/disables reception of all functional addresses. The effect of each is always on a per-stream basis and independent of the other **sap** and physical level configurations on this stream or other streams.

The **DL_PHYS_ADDR_REQ** primitive returns the 6-octet token ring address currently associated (attached) to the stream in the **DL_PHYS_ADDR_ACK** primitive. This primitive is valid only in states following a successful **DL_ATTACH_REQ**.

The **DL_SET_PHYS_ADDR_REQ** primitive changes the 6-octet token ring address currently associated (attached) to this stream. The credentials of the process which originally opened this stream must be superuser or an **EPERM** error is returned in the **DL_ERROR_ACK**. This primitive is destructive in that it affects all other current and future streams attached to this device. Once changed, all streams subsequently opened and attached to this device will obtain this new physical address. The new physical address will remain in effect until this primitive is used to change the physical address again or the system is rebooted, whichever comes first.

CONFIGURATION

The **/kernel/drv/tr.conf** file supports the following options:

intr Specifies the IRQ level for the board. Note that if the dip switches for the board are set to use the cascade interrupt, IRQ 2, the IRQ level specified in the configuration file should be IRQ 9.

ioaddr Specifies the beginning I/O port address occupied by the board.

reg The first register property specifies the location and size of the board's BIOS/MMIO area. The second register property specifies the location and size of the board's shared RAM.

It is important to ensure that there are no conflicts for the board's I/O port, shared RAM, or IRQ level.

FILES

/dev/tr

/kernel/drv/tr.conf **tr** configuration file.

SEE ALSO

dlpi(7P)

NOTE

IBM 16/4 Token Ring Network Adapters and compatibles are not capable of fully supporting the **snoop**(1M) program. This limitation is due to the hardware itself and not to a bug in the **tr** driver or the **snoop** program.