

June 2002

***WANPIPE™ WAN ROUTING
SUPPORT for the Sangoma
S-Series cards***

Installation and Operations Manual

LIMITED USE LICENSE AGREEMENT

Sangoma Technologies Inc. provides the computer software program contained on the medium in this package (hereinafter called the Program) and licenses its use.

THE LICENSEE SHOULD CAREFULLY READ THE FOLLOWING TERMS AND CONDITIONS BEFORE ATTEMPTING TO USE THIS PRODUCT. INSERTION OF ANY OF THE DISKETTES IN THIS PACKAGE INTO ANY MACHINE INDICATES THE LICENSEE'S ACCEPTANCE OF THESE TERMS AND CONDITIONS. IF THE LICENSEE DOES NOT AGREE WITH THE TERMS AND CONDITIONS, THE LICENSEE SHOULD PROMPTLY RETURN THE PACKAGE WITHIN 15 DAYS UNUSED AND UNCOPIED IN ANY WAY SHAPE OR FORM, AND MONIES WILL BE REFUNDED.

LICENSE:

- a. The purchaser of this license (hereinafter called the Licensee) is granted a personal, non-exclusive license to use the Program in accordance with the terms and conditions set out in this agreement.
- b. The Program may be used only on a single computer per license granted.
- c. The Licensee and the Licensee's agents and employees shall protect the confidentiality of the Program and shall not distribute or make available the Program or documentation to any third party.
- d. The Licensee may copy the programs into machine readable or printed form for backup or modification purposes only in support of the Licensee's use on a single machine. The Licensee must reproduce and include the copyright notice on any copy, modification or portion merged into another program.
- e. Any portion of the Program merged into or used in conjunction with another program will continue to be subject to the terms and conditions of this agreement.
- f. The Licensee may not assign or transfer the license or the program to any third party without the express prior consent of Sangoma Technologies Inc.
- g. The licensee acknowledges that this license is only a limited license to use the Program and documentation, and that Sangoma Technologies Inc. retains full title to the program and documentation.
- h. The Licensee shall not use, copy, modify or transfer the Program or documentation or any copy, modification or merged portion, in whole or in part, except as expressly provided for in this license. If the Licensee transfers possession of any copy, modification or merged portion of the program to a third party, the license is automatically terminated under this agreement.

TERM:

The license is effective until terminated. The licensee may terminate the license at any time by destroying the Program together with all copies, modifications and merged portion in any form. The Licensee agrees upon such termination to destroy the Program together with all copies, modifications and merged portion in any form.

LIMITED WARRANTY:

The Program is provided "as is" without warranty of any kind, either expressed or implied, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. The entire risk as to the performance of the Program is with the Licensee. Should the Program prove defective, the Licensee (and not Sangoma Technologies Inc. or an authorized dealer) shall assume the entire cost of all necessary servicing, or correction. However, Sangoma Technologies Inc. warrants the diskettes on which the Program is furnished will be free of defects in materials or workmanship under normal use for a period of 90 days from the date of delivery to the Licensee. In no event will Sangoma Technologies Inc. be liable for any damages, including incidental or consequential damages arising out of the use or inability to use the Program, even if Sangoma Technologies Inc. or an authorized dealer have been advised of the possibility of such damages, or for any claim by any other party.

The Licensee acknowledges that the Licensee has read this agreement, understands it, and agrees to be bound by its terms and conditions. The Licensee further agrees that it is the complete and exclusive statement of the agreement between the parties and supersedes any proposal or prior agreement, oral or written, and any other communications between the parties relating to the subject matter of this agreement.

Other Disclaimers for S508/FT1 and S514/FT1

1. This equipment complies with Part 68 of the FCC rules. On the trim plate of this equipment is a label that contains, among other information, the FCC registration number for this equipment. If requested, this information must be provided to the telephone company.
2. The following are the applicable registration jack USOCs for the S514/FT1 and S508/FT1 Cards.

Port ID	REN/SOC	FIC	USOC
T1 SF	04DU9-BN	6.0F	RJ48C
T1 SF and B8ZS	04DU9-DN		
T1 ANSI	04DU9-1KN		
T1 ANSI ESF and B8ZS	04DU9-1SN		

4. If the terminal equipment, S508/FT-1 or S514/FT1, causes harm to the telephone network, the telephone company will notify you in advance that temporary discontinuance of service may be required. But, if advance notice isn't practical, the telephone company will notify the customer as soon as possible. Also, you will be advised of your right to file a complaint with the FCC if you believe it is necessary.
5. The telephone company may make changes in its facilities, equipment, operation or procedures that could affect the operation of the equipment. If this happens, the telephone company will provide advance notice in order for you to make necessary modifications to maintain uninterrupted service.
6. If trouble is experienced with this equipment please contact Sangoma for repair or warranty information. If the equipment is causing harm to the telephone network, the telephone company may request that you disconnect the equipment until the problem is resolved.
7. There are no user-serviceable parts on this board. If a problem occurs, try a different cable and/or board. If problems persist, contact Sangoma for assistance.

1. Introduction

WANPIPE™ software and Sangoma cards provide the means to leverage the existing operating system features so that the server or workstation becomes a high speed, high performance WAN router in addition to its other duties.

The resulting router is:

- Easy to install
- Easy to manage and debug
- More reliable than separate boxes
- Faster than comparable external devices.

This manual describes **WANPIPE™** support for Windows NT4, Windows 2000/XP, Windows 95/98, and Novel Netware.

WANPIPE™ support for Linux, FrreBSD and OpenBSD is available in source under GNU license. Code can be found at **<ftp.sangoma.com/pub>**.

All brand names and product names are trademarks of their respective companies.

2. Quick Install

1. Choose the jumper settings for the card based on the other hardware installed on your system (See Section 3.). Your new **WANPIPE™** card will be the PCI based S514 which has no jumpers that need to be set, or the S508, which only has one jumper setting: the I/O base address. On the S508, the default base I/O address setting of 0360 HEX is almost always satisfactory.

Note: For EISA machines, you may have to generate a configuration file for the card.

2. Install the card in the host machine.
3. Install the software and drivers for your particular operating environment.

Windows 2000/XP the S514 card will be detected automatically.

PLEASE NOTE: The CD contains support for many different applications under many different operating systems. In particular, there are many different types of Windows drivers that work on the card that you have purchased.

You cannot use the automatic search facility in the New Hardware Wizard to find your correct driver, because the system will simply choose the first one it finds, and that may not be suitable for your application.

For installation instructions for **Windows 2000/XP** please refer to INSTALL.PDF or INSTALL.TXT in `.\WANPIPE\Win_2000_XP` directory.

Windows NT4 and Windows 9x: Use the Add Adapter function in the Network applet.

Please note that for Linux or FreeBSD users, the latest code, including source, can be found on our web site at ftp.sangoma.com.

4. Configure the card, protocol and line parameters: For the S508 ISA card, once you have chosen the hardware parameters, make use of the **Test** button to ensure that the I/O port, shared memory and IRQ settings do not clash with other hardware.
5. Set up the IP routing stack on your host, including IP addresses, gateways and network routes.
6. If you are using the S514/FT1 or the S508/FT1, with integrated DSU/CSU, use the Monitor program **fpipemon**, **cpipemon**, or **ppipemon** to configure the DSU/CSU.

7. Make sure your “other end” is set up correctly. Many third party routers default to proprietary, non standard protocols, while **WANPIPE™** adheres strictly to Internet or IETF standards of encapsulation.

Some pointers:

- a. Set third party router for PPP, or IETF standard (RFC 1490) Frame Relay.
 - b. Turn off all password authentication such as CHAP and PAP.
 - c. Set up only **Static routes** to **WANPIPE™**. Turn off all learning and routing protocols such as ARP, Inverse ARP, Reverse ARP, RIP, OSPF etc.
8. Connect up. Ping the other end.
 9. If the connection does not work, problems with **WANPIPE™** fall into 2 categories:
 - a. **Hardware loading problems:**

We provide a comprehensive tool-set for resolving hardware conflicts under all operating environments. Generally you can see what is happening by examining the system logs. **WANPIPE™** also provides its own logs which are written in English.
 - b. **WAN problems:**

The Monitor programs **fpipemon, cpipemon, or ppipemon** provide everything you need to resolve every kind of WAN problem from incompatible DSU/CSU settings through noisy lines to protocol encapsulation problems.

For instance, if you cannot ping a host, first use the monitor to see that in fact you are both sending and receiving, and look to see if the interface is up. If it is, then ping the host that you cannot reach and watch the packet stats for the interface. If the packet count does not go up with each ping, there is something wrong with the routing table at your end. If the packet count goes up with each ping, then your routing table is fine, and the problem is at the other end. And so on.

You will find these utilities will turn you into a WAN guru. You will always know more about the state of the WAN connection than either the network provider or the third party at the other end.

3. Hardware

The S514 is a PCI based card. The S508 is an ISA card. Both cards are available with V.35, X.21, EIA530 or RS232 cabling, or with the optional integrated T1 DSU/CSU on board.

Prior to installing the card hardware into your PC, jumpers may need to be set. Some of the jumpers set factory configurations, and should not be touched.

S514 PCI card

No jumpers need to be set on this card as it is configured by the PC BIOS.

S508 Card

Jumpers JP1 on the S508 define the card I/O address range as specified in Table 3-1. The specified card I/O addresses must not conflict with I/O addresses in use by any other hardware installed on the server. Use the SNOOPER utility if you are in any doubt as to hardware settings.

Note that JP1-1 on the S508 is furthest to the left if the board is held such that the connectors are to the right. JP1-4 is reserved.

The 8k (2000 Hex) byte shared memory address and the IRQ level are set in software for the S508.

Internal Line Clocking

For back-to-back connections, the cards can provide their own Transmit and Receive clock signals, which, with the appropriate cable, can also provide the clock for third party devices.

All cards are capable of generating the transmit and receive clocks as long as the appropriate back-to-back cable is used. The generated line speed is set by software.

However, the cards have a very large configurable range and therefore cannot easily be tabulated. When asked for the line speed during setup, you may specify any value in kbps from 1 to 2600. The actual generated line speed will be reasonably close the specified value, but will deviate more as the line speed increases.

S514 Cable Pinouts

NB: Port PA is the Primary 4Mbps port
 Port PB is the Secondary 512Kbps port.

PIN #	PA:RS232	PA:V.35	PB:RS232	PB:V.35
1	RTS		RTS	
2	CTS		CTS	
3	GND	GND	GND	GND
4	DCD		DCD	
5	DTR		DTR (V.10)	
6	TXD			
7	RXD			
8	TXC			
9	RXC			
10		RTS		RTS
11		CTS		CTS
12		DCD		DCD
13		DTR		DTR (V.10)
14		TXD		
15		RXD		
16		TXC		
17		RXC		
18			TXA	
19			TXB	
20			RXA	
21			RXB	
22			TX Clock A	
23			TX Clock B	
24			RX Clock A	
25			RX Clock B	
26			DTR A (V.11)	
27			DTR B (V.11)	
28				TXA
29				TXB
30				RXA

PIN #	PA:RS232	PA:V.35	PB:RS232	PB:V.35
31				RXB
32				TX Clock A
33				TX Clock B
34				RX Clock A
35				RX Clock B
36				DTR A (V.11)
37				DTR B (V.11)

S508 Cable Pinouts

RS232

Pin #	Function
2	TxD
3	RxD
7	GND
4	RTS
5	CTS
20	DTR
6	DSR
8	DCD
15	TxC
17	RxC
24	BxC

V.35/X.21

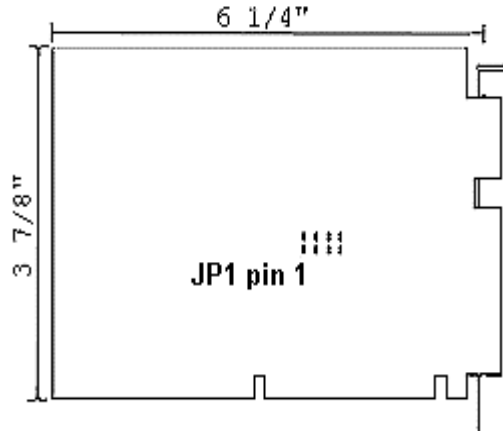
Pin #	Function
4	RTS
5	CTS
6	DSR
7	GND
8	DCD
10	TxA
9	TxB
12	RxA
11	RxB
19	Tx Clock A
20	DTR (V10 signal)
13	DTRA (V11 signal)
14	DTRB (V11 signal)
21	Tx Clock B
22	RI
23	Rx Clock A
25	Rx Clock B
18	Aux. Clock A (On board clock source)
16	Aux. Clock B (On board clock source)

S508 I/O port Jumper Settings

Table 3-1 JP1 - I/O Address Selection

JP1-1	JP1-2	JP1-3	Selection
CLOSED	CLOSED	CLOSED	I/O Address 250-253 (Hex)
OPEN	CLOSED	CLOSED	I/O Address 270-273 (Hex)
CLOSED	OPEN	CLOSED	I/O Address 280-283 (Hex)
OPEN	OPEN	CLOSED	I/O Address 300-303 (Hex)
CLOSED	CLOSED	OPEN	I/O Address 350-353 (Hex)
OPEN	CLOSED	OPEN	I/O Address 360-363 (Hex)
CLOSED	OPEN	OPEN	I/O Address 380-383 (Hex)
OPEN	OPEN	OPEN	I/O Address 390-393 (Hex)

Factory default.



WANPIPE™ for Windows NT4/2000/XP

Introduction

The **FPIPE/PTPIPE/CPIPE** NDIS driver for Frame Relay, PPP and HDLC respectively are incorporated into the networking system of Windows NT and Windows 2000. They behave like standard Ethernet drivers encapsulating both IP and IPX packets for transmission over Frame Relay or Point-to-Point (PPP or HDLC). A separate set of drivers is provided for **XPIPE**, the X.25 routing facility.

All the applications that can run over TCP/IP or IPX will run transparently over **WANPIPE™**. These include Internet applications of all types as well as Microsoft Networking and NetWare clients.

Windows NT and Windows 2000 have the capability of being an IP and IPX router, and so may function as a gateway for a LAN.

Installation

Windows 2000/XP

For installation instructions for **Windows 2000/XP** please refer to INSTALL.PDF or INSTALL.TXT in **.\WANPIPE\Win_2000_XP** directory.

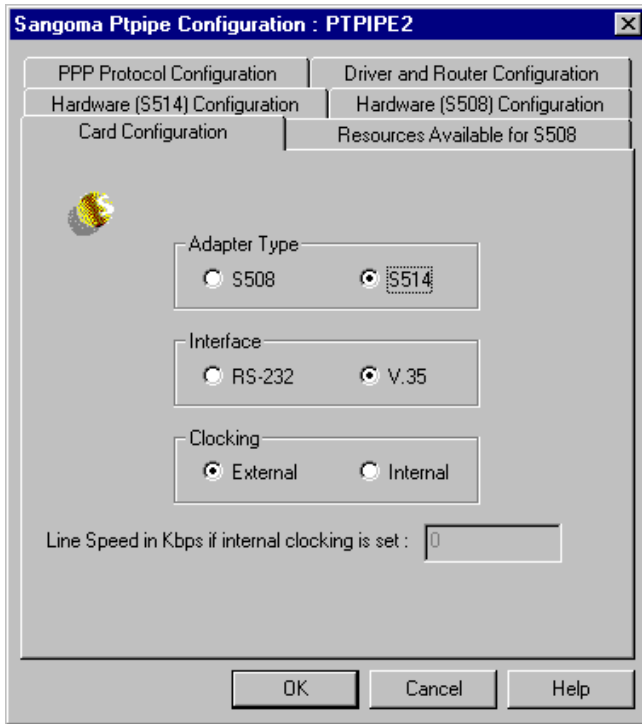
To set protocol and line configuration right click on “**Sangoma WAN Connection Driver**” in “**Device Manager**” and choose “**Properties**”. Context sensitive help is available for each Property Page in WinHelp format. Use “**Help**” button to access the help.

Windows NT

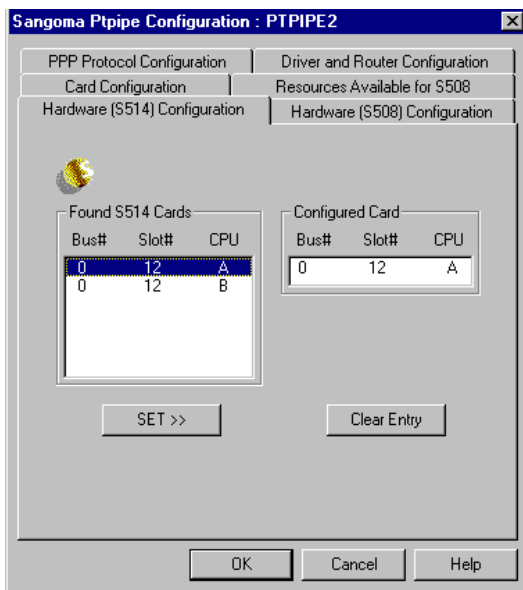
For both the S514 and the S508:

1. Open **Control Panel**.
2. Run the **Network** applet.
3. Click on the **Add Adapter..** button..
4. Select 'Other' from the network adapter card list and click on 'Continue'.
5. Click on the **Have Disk...** button.
6. Insert the **WANPIPE™** distribution disk in the drive and choose the appropriate source path (it defaults to **A:**).

7. Select the correct adapter, interface and clocking and click 'OK'. Note that clocking is **External** unless you are connecting locally back-to-back with a special null modem cable. A typical screen is shown below.

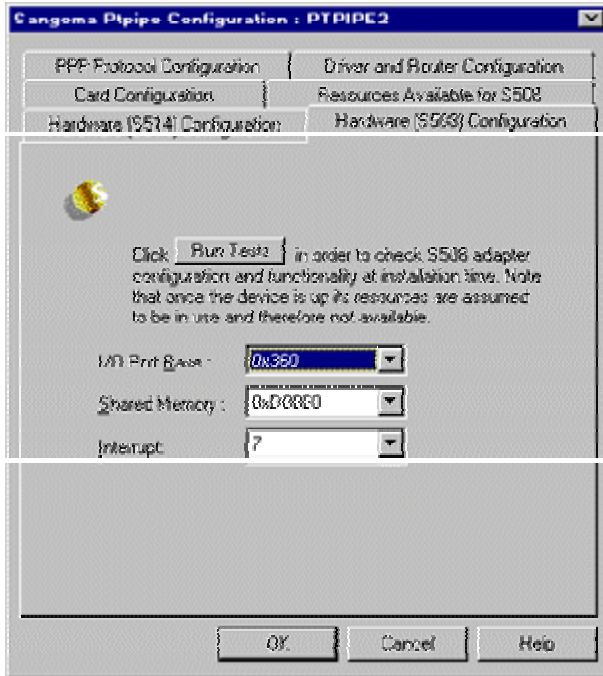


8. Set the properties and configuration of the adapter accordingly. A detailed explanation of these settings may be found in the **Driver Properties** section. View the **Available Resources** before choosing the card hardware settings to use. The S514 card, being PCI, has no user settable Resources, set-up being done by the BIOS.



However, as shown above, the S5142 has dual CPUs, each of which can be used independently for a different link. Thus for example, you could have X.25 on CPU A and PPP on CPU B.

If using the S508, once you have set the hardware specific properties of I/O port base address, Shared memory base address, and IRQ, use the **Run Tests** button to ensure that all those resources are available, and that the card functions properly with those settings. Do this before setting the rest of the link properties.



9. Click the **OK** button.

You can also save on a reboot if you configure the hardware and protocol parameters before rebooting.

Configuration

508-Hardware Properties

The S508 as an ISA card needs to have its hardware set up manually.

Property	Value	Description
I/O Port Addr	(JP1)	See JP1 table in hardware installation.
IRQ Level	IRQ	One of: 3, 4, 5, 7, 10, 11, 12.
Shared Memory	XY000	Physical starting address (20 bits) of shared memory window of size 8192 bytes. X = { A, C, D, E } Y = { 0, 2, 4, 6, 8, A, C } Try D0000 or C8000

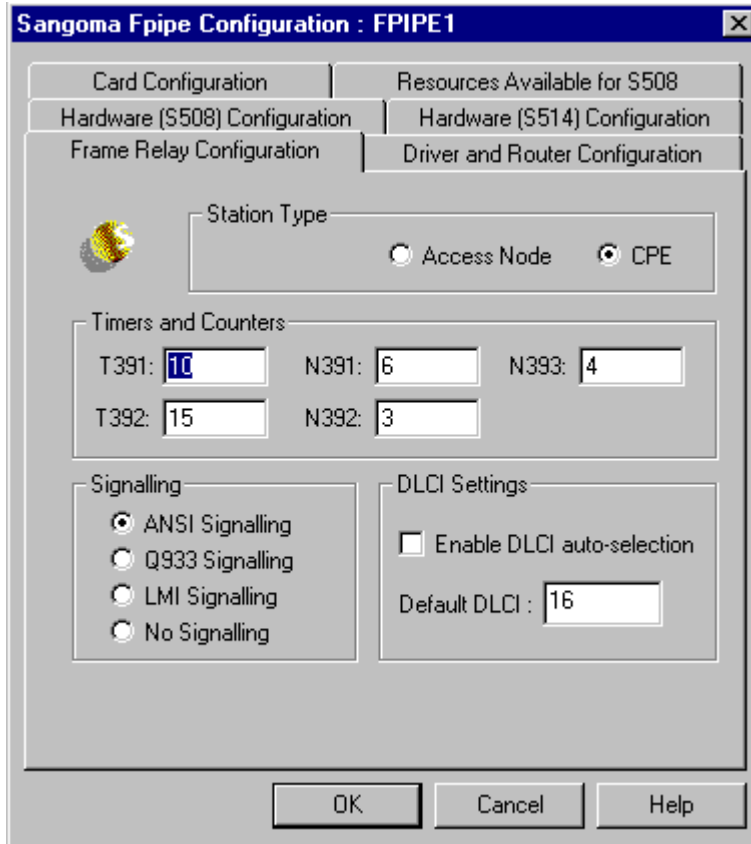
General Hardware Properties

These apply to both card types.

Property	Value	Description
Clocking	External	If the DSU/CSU or other device is providing clocking. This is ignored if you are using the S508/FT1 or S514/FT1 card with internal DSU/CSU.
	Internal	If this card is providing the clock signal in a back-to-back configuration. This is ignored if you are using the S508/FT1 or S514/FT1 card with internal DSU/CSU.
Line Speed	(kbps)	Line speed in kbps. Setting this to 0 also implies external clocking. This is ignored if you are using the S508/FT1 or S514/FT1 card with internal DSU/CSU.
RS232/V.35		Set to V.35 also for X.21 and EIA530 interfaces. This is ignored if you are using the S508/FT1 or S514/FT1 card with internal DSU/CSU.

Frame Relay Properties (FPIPE)

These properties are set in this window under Windows NT4:



Note that most Frame Relay connections with only one DLCI will work with the default settings.

<u>Property</u>	<u>Value</u>	<u>Description</u>
Baud Rate	0 - 2048	If you know the line speed enter it here. Do this even for external clocking.
Clocking/Interface	RS232/V.35, Internal/External	Note that V.35 is the setting for V.35, X.21, RS422, EIA530 and other balanced interfaces. Internal clocking is only used for local serial back-to-back connections. S508/FT1 and S514/FT1 cards are always set for External clocking.
Default DLCI (0=Auto)	0 or as defined by your Telco	If set to 0, and if there is only one DLCI, then WANPIPE will discover the active DLCI. This is the DLCI to be used if an explicit DLCI cannot be found in the routing table, see below.
Interface Name		This name can be used by Fpipemon instead of an IP address to monitor the board. Type in any name (must not start with a number). The name must be unique for

<u>Property</u>	<u>Value</u>	<u>Description</u>
		each instance of FPIPE installed.
IP/DLCI Table		FPIPE supports 100 DLCIs. The mapping of IP addresses to Frame Relay DLCIs is done in a special text table that looks a bit like a routing table. Type in IP/DLCI routing table file name. Fpipetbl.sam on the distribution disk is an example of such a table. Note that the file should be in system32\drivers directory. Note that if there is only one DLCI, the table is not necessary.
Log Errors and Info	Yes or No	YES if you want resource allocation and Frame Relay channel errors and status messages to be written to system32\drivers\fpipelog.wri file. Default:NO
N391		Full Status Polling Cycle Counter. Do not change without advice from your Telco. Default:6
N392		Error Threshold Counter. Do not change without advice from your Telco. Default:3
N393		Monitored Events Counter. Do not change without advice from your Telco. Default:4
Signalling	ANSI, LMI, Q.933 or None	Network management signalling. Check this with your Telco. Default:ANSI
Station Type	CPE or Access Node	CPE (Customer Premises Equipment). Use when connecting to a real Frame Relay network or as one end of a back-to-back connection. The Access Node setting is used to emulate a switch as one end of a back-to-back connection where there is no real Frame Relay network involved.
T391		Link Integrity Verification Timer. Do not change without advice from your Telco. Default:10
T392		Polling Verification Timer. Do not change without advice from your Telco. Default:15
UDP Management Port		UDP Management Port used by Fpipemon to monitor the driver. Normally 9000 unless it conflicts with another device. Setting to 0 disables the monitor.
TTL		Time-to-Live parameter in IP packet sent by Fpipe driver in reply to Fpipemon information request. By changing this parameter, you can change the range at which the line statistics can be read by a remote copy of

<u>Property</u>	<u>Value</u>	<u>Description</u>
		fpipemon. Default 32.

PPP Properties (PTPIPE)

The properties are accessed in the same way as the Frame Relay properties.

Note that most PPP connections will work with the default settings.

<u>Property</u>	<u>Value</u>	<u>Description</u>
Baud Rate	0 – 2048	Data transfer rate in kbps, between 0 and 2600. If 0 then the clock source is External. External is always the correct setting unless you are connecting locally back-to-back without a DSU/CSU.
Interface	RS232/V.35	Physical interface type, RS232/V35. Use V.35 for X.21, EIA530.
Interface Name		This name can be used by Ppipemon instead of an IP address to monitor the board. Type in any name (must not start with a number). The name must be unique for each instance of PTPPIPE installed.
Connect	0-2621	Time to wait in 10ths of a second for DCD to go high during a connection attempt before dropping DTR# to hang up. If this peer must wait for incoming connections, this should be set to zero.
DCD/CTS Drop	1-2621	Time to waite in 10ths of a second since DCD or CTS dropped before declaring the link disconnected. Recommended value is 10.
Drop DTR	1-2621	Minimum time in 10ths of a second to keep DTR and CTS low during disconnection. This is normally greater than DCD/CTS Drop. Recommended value is 15.
Log Errors and Info	Yes or No	YES if you want resource allocation and Frame Relay channel errors and status messages to be written to system32\drivers\ppipellog.wri file.
MaxConfigure	1-65535	Number of Configure-Request packets sent without receiving a replay before terminating the link. Recommended value is 10.
MaxFailure	1-65535	Number of Configure-Nak packets sent without sending a Configure-Ack before assuming the configuration is not converging. In some cases, the Configure-Nak packets are then converted to Configure-Reject packets; otherwise the link will be immediately terminated. Recommended value is 5.
MaxTerminate	1-65535	Number of Terminate-Request packets sent without receiving a reply before terminating the link. Recommended value is 2.
Restart	1-2621	Time in 10ths of a second between successive

<u>Property</u>	<u>Value</u>	<u>Description</u>
		retransmissions of Configure-Request or Terminate-Request packets. Recommended value 30.
UDP Management Port		UDP Management Port used by Ppipemon to monitor the driver. Normally 9000 unless it conflicts with another device. Setting to 0 disables the monitor.
TTL		Time-to-Live parameter in IP packet sent by Ppipe driver in reply to Ppipemon information request. By changing this parameter, you can change the range at which the line statistics can be read by a remote copy of Ppipemon . Default is 32.

HDLC Properties (CPIPE)

The properties are accessed in the same way as the Frame Relay properties.

Note that most HDLC connections will work with the default settings.

<u>Property</u>	<u>Value</u>	<u>Description</u>
Baud Rate	0 - 2048	Data transfer rate in kbps, between 0 and 2600. If 0 then the clock source is External. External is always the correct setting unless you are connecting locally back-to-back without a DSU/CSU.
CHDLC option Ignore CTS		Ignore CTS when determining active link status.
CHDLC option Ignore DCD		Ignore DCD when determining active link status.
CHDLC option Ignore Keepalive		Ignore Keepalive statistics when determining active link status.
Electrical Interface	RS232/V.35	Physical interface type, RS232/V35. Use V.35 for X.21, EIA530.
Instance Name		This name can be used by Cpipemon instead of an IP address to monitor the board. Type in any name (must not start with a number). The name must be unique for each instance of CPIPE installed.
Keepalive error margin	1 – 20	Consecutive keepalive timeouts before bringing down the link.
Keepalive RX timer	10-60000 ms	Interval to expect Keepalives .
Keepalive TX timer	10-60000 ms	Interval to transmit Keepalives .
Log Errors and	Yes or No	YES if you want resource allocation and Frame Relay

Property	Value	Description
Info		channel errors and status messages to be written to system32\drivers\cpipelog.wri file.
SLARP timer	0-60000ms	Time (ms) between sending SLARP requests range. If SLARP timer is 0: Turns off dynamic IP addressing and uses IP address and Subnet mask values from the "Network and Dial-up Connections"->"Local Area Connection"->"TCP/IP Properties". If SLARP timer is NOT 0: will use dynamic IP addressing as provided by the other side of the link. Note that even if you are expecting to use dynamic IP addressing, <i>you must still specify an IP address for this device</i> in order for it to work.
Statistics option RX throughput		If YES then the throughput of data frames received will be available for Cpipemon.
Statistics option TX throughput		If YES then the throughput of data frames transmitted will be available for Cpipemon.
UDP Management Port		UDP Management Port used by Cpipemon to monitor the driver. Normally 9000 unless it conflicts with another device. Setting to 0 disables the monitor.
TTL		Time-to-Live parameter in IP packet sent by Cpipe driver in reply to Cpipemon information request. By changing this parameter, you can change the range at which the line statistics can be read by a remote copy of Cpipemon . Default is 32.

IP Configuration

WANPIPE™ is configured for IP in exactly the same way as any NIC card.

Bind TCP/IP to the card as you would with an Ethernet card. Set always for a *static IP address*. There are some situations where the address may be changed dynamically on connection, but the system always requires a static IP address to function.

Usually the default route will also be set on the **WANPIPE™** connection. Remember that there can only be one default route on a system, thus all the other interfaces must have their Default Addresses set to <Blank>. For standard routing, be sure to turn on IP forwarding.

The Windows environment requires that each network segment have its own unique network address. If you are using real IP addresses on your LAN (i.e., no NAT or Proxy), then you will have to define a small two node network for your WAN link. For details of IP configurations, see our tutorial on <http://www.sangoma.com/TCP.pdf>.

Because **WANPIPE™** looks to the system like any NIC, any of the standard routing additions such as Proxy, NAT, VPN, and Firewalls work transparently on this system.

Frame Relay, where **WANPIPE™** supports up to 100 DLCIs (Logical Channels) through a single interface can be a little more complicated. This is done through the DLCI/Routing table that you can provide a pointer to in the card's Frame Relay (Advanced) properties.

IP/DLCI Routing Table for Frame Relay (Windows NT4 and 9x)

For every packet routed by TCP/IP to the Frame relay **FPIPE** adapter, the destination IP network address is compared to entries in this table to locate the correct DLCI to use for its transmission. If a matching IP network is not found, the DLCI in **Properties** is used as a default. IPX packets always use this default DLCI.

See the file **fpipetbl.sam** for a sample. Any text editor may be used.

There is a maximum of 100 DLCIs that may be configured. Note however that Windows 95 only allows you to configure **one IP address** for the adapter. Thus under Windows 95, you can only configure subnets of the network address you have assigned the adapter when configuring TCP/IP from the **Control Panel**.

The table is of the format:

```
<ip_network_address>      <netmask> <DLCI>
```

Where:

```
<ip_network_address>
```

is the IP network address (with zero host part);

```
<netmask>
```

is the netmask used to isolate the network portion; and

```
<DLCI>
```

is the data link connection identifier associated with this logical IP network.

Below is a sample table. Use it as a guide for creating your own routing tables. There may be many more entries listed here than are configured for TCP/IP. This is because you only need to tell TCP/IP about the higher-classed networks, unless there is a need to assign the host an IP address for every subnet. The subnets are created here for each DLCI. For this sample table, the following IP/Netmask entries would need to be configured for TCP/IP:

Host IP Address	Netmask	Site
-----	-----	----
200.1.1.1	255.255.255.0	1-4
200.1.2.1	255.255.255.0	5-6
200.1.3.254	255.255.255.0	7
OR...		
200.1.1.1	255.255.255.192	1
200.1.1.65	255.255.255.192	2
200.1.1.129	255.255.255.192	3
200.1.1.193	255.255.255.192	4
200.1.2.1	255.255.255.128	5

200.1.2.129	255.255.255.128	6
200.1.3.1	255.255.255.0	7

Note that this is *not a routing table* in the normal meaning of the term, although it performs the limited routing functions needed by FPIPE.

Only 5 subnets may be defined using Control Panel, but the extras can be added manually in the registry under:

```
HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\
Services\FPIPE\Parameters\Tcpip.
```

You may be able to save entering so many addresses if you can get away with adding a default route to this adapter, in which case the table below can do all the routing. The remote routers at each site will have to use a phantom IP address as the gateway to reach this host, but be aware that it may cause administrative confusion as it can't be pinged for example.

200.1.1.0	255.255.255.192	30	# Site 1
200.1.1.64	255.255.255.192	31	# Site 2
200.1.1.128	255.255.255.192	32	# Site 3
200.1.1.192	255.255.255.192	33	# Site 4
200.1.2.0	255.255.255.128	40	# Bigger Site 5
200.1.2.128	255.255.255.128	41	# Bigger Site 6
200.1.3.0	255.255.255.0	50	# Really Big Site 7

Any packet routed to the FPIPE adapter that does not have a specific route specified for it in this table will use the default DLCI as configured in **Properties**.

Note: Some routers do not accept a subnet number in an IP address of all ones or all zeros. Therefore, the netmask 255.255.255.128 and the networks of Site 1 and 4 of the netmask 255.255.255.192 may not be usable. However, Windows 95 and Windows NT will accept these tables.

Diagnostics

Log files

If you have set up the card's properties to include Logging, then the file
..\system32\drivers\?pipelog.wri (on NT4) or
..\system32\drivers\wanpipelog.wri (on Win 2000/XP).
will contain information in English relating to the operation of the card.

A typical log file for PPP is shown below:

```
16/8/2000 12:14:16 PTPIPE514_0:Found S514 adapter.
16/8/2000 12:14:16 PTPIPE514_0:connecting to IRQ 10
16/8/2000 12:14:16 PTPIPE514_0:Adapter memory at 0xF5E43000
16/8/2000 12:14:16 PTPIPE514_0:Loading S514 adapter CPU A
16/8/2000 12:14:16 PTPIPE514_0:Entering interrupts test...
16/8/2000 12:14:16 PTPIPE514_0:Passed interrupts test. Link connecting...
16/8/2000 18:16:32 PTPIPE514_0:Found S514 adapter.
16/8/2000 18:16:32 PTPIPE514_0:connecting to IRQ 10
16/8/2000 18:16:32 PTPIPE514_0:Adapter memory at 0xF6F1E000
16/8/2000 18:16:32 PTPIPE514_0:Loading S514 adapter CPU A
16/8/2000 18:16:32 PTPIPE514_0:Entering interrupts test...
16/8/2000 18:16:32 PTPIPE514_0:Passed interrupts test. Link connecting...
16/8/2000 18:29:24 PTPIPE514_0:The protocol reached 'Opened' state, data may
now be transmitted and received.
```

If there are driver start up problems or resource conflicts, these log entries will help to resolve them.

Using the monitors FPIPEMON, CPIPEMON and PPIPEMON

The monitor programs FPIPEMON, CPIPEMON and PPIPEMON work identically on any Windows platform. In fact, you can run the monitor on a Win 95 machine to debug a Win 2000 host and vice versa. You can also use them to debug a Linux or Unix **WANPIPE™** installation.

The appropriate monitor will be found installed on the Start menu. They are 32-bit programs that may be run on any Windows 95 or Windows NT/2000/XP platform that is logically connected to the **WANPIPE™** host.

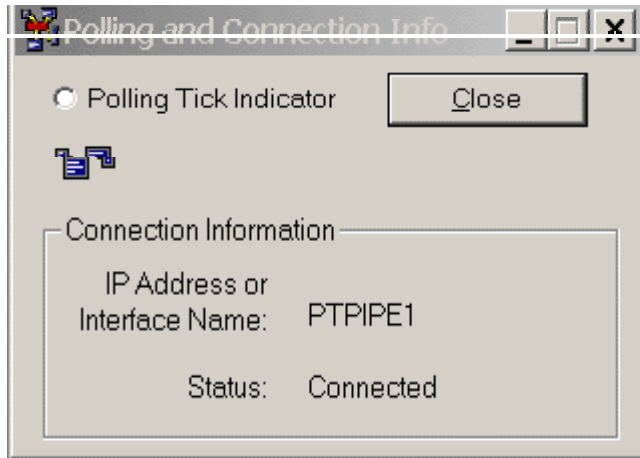
The monitors provide real time debugging information of the WAN links to any machine anywhere that can logically reach the host machine. They provide real time data on the performance of the link, the DSU/CSU and the traffic.

When you start the program, the **Polling and Connection Information** dialogue box is active. Close this box and open **File/Select Card**. If you are running the monitor on the host containing the Sangoma card, **File/Select Card or Monitor/Select Card**

will show you the name(s) of the card(s) installed on the machine. Chose the one you need to monitor.

If this is not the machine hosting the cards, and you are accessing the cards remotely, see the note below on IP address to use. You may also need to **Configure** if you are not using the default UDP Port of 9000.

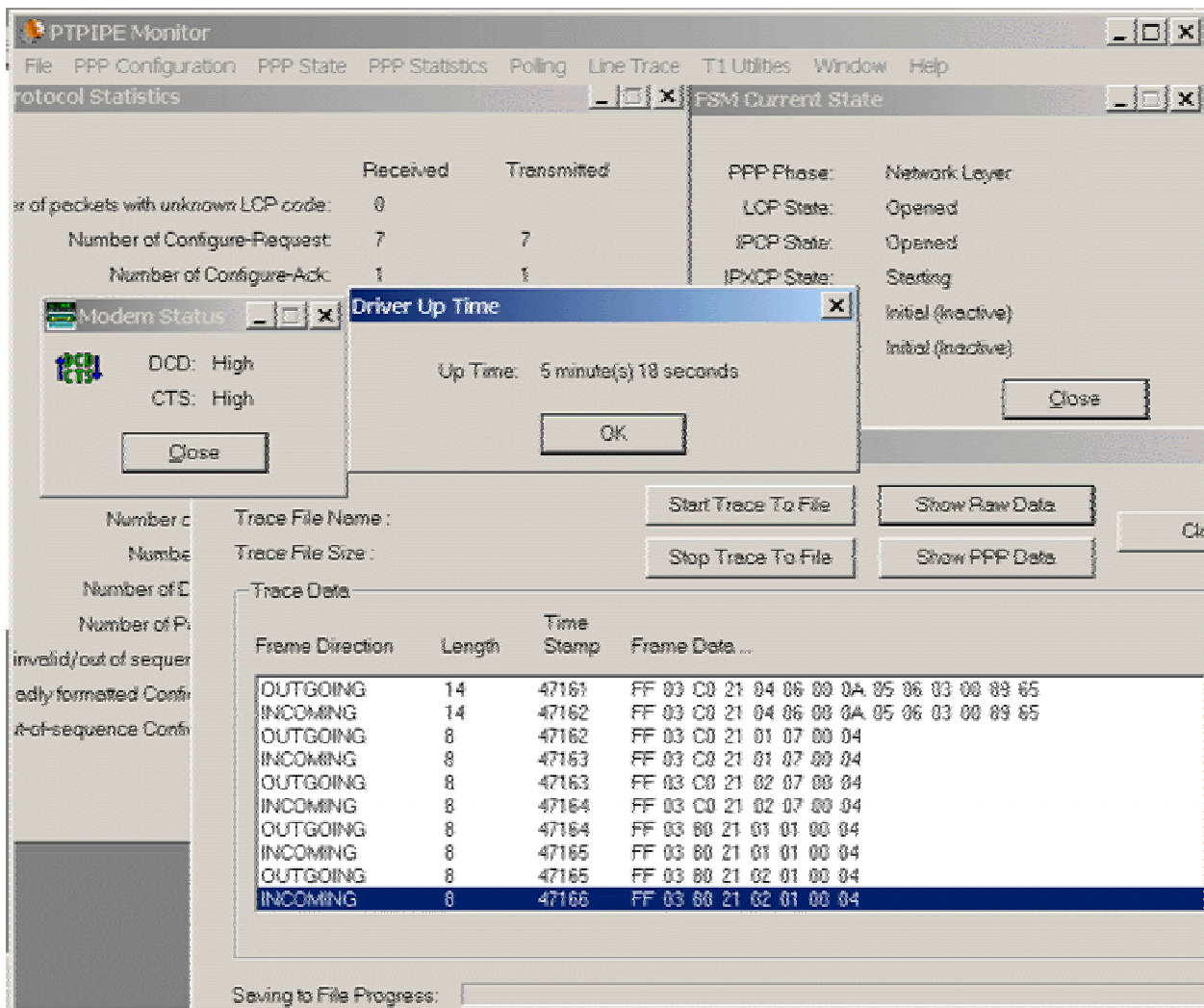
Now, when you chose **Connect** , a window similar to the following window will appear:



This window shows, for instance that the monitor is connected to the device named PTPipe1. If, for some reason, a connection cannot be made, the window will show up as **Disconnected**.

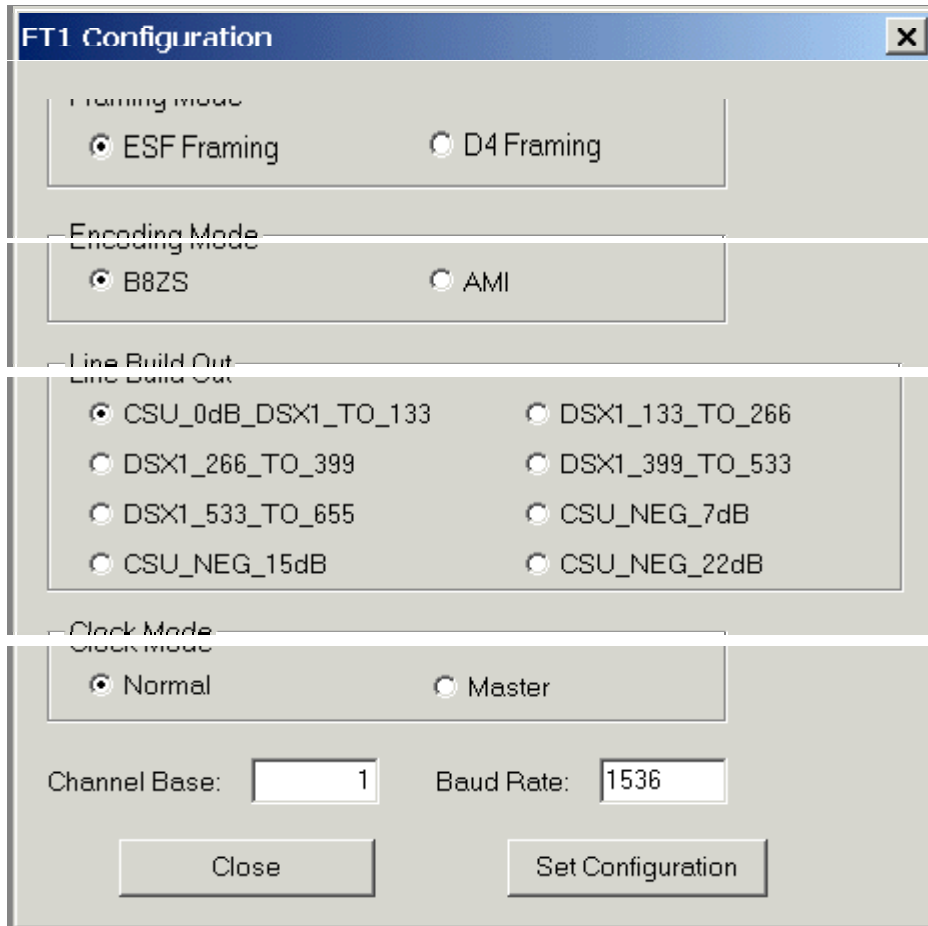
Once you are connected, you can look at statistics that can help debug line, DSU/CSU and configuration problems in real time. There is even a protocol analyzer that can provide a dump of all the WAN network traffic.

An example of a PPP monitoring screen is shown below.

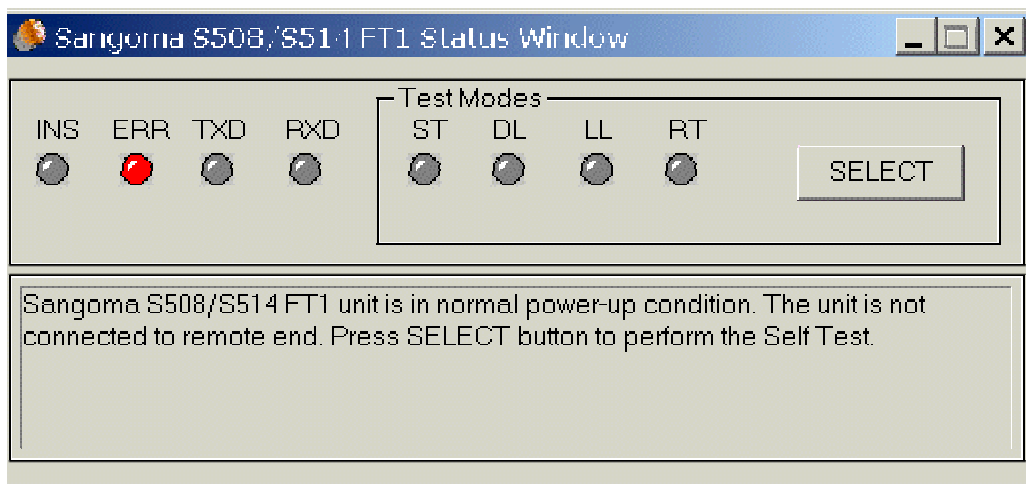


If the card you are using has the integrated T1 and Fractional T1 DSU/CSU, you can monitor and set up the unit using the T1 utilities button. The T1 configuration can be done dynamically without any kind of reset of the system. Chose the **FT1 configuration** selection to configure the unit.

The T1 settings should be set to match those given to you by your Telco. The default settings shown in the screen below are typical of those for a full T1 (1.54Mbps). Once **Set Configuration** has been clicked, the T1 parameters are set, and the unit returns to the communication mode.



You can monitor the status of the DSU/CSU unit through **T1 utilities/Status Monitor**.



The DSU/CSU status as indicated by the lights will have meaning to your Telco.

The **INS** light has the following meaning:

Solid Green	Normal
Blink Green	Blue Alarm
Blink Orange	Yellow Alarm
Blink Red	Red Alarm

The **ERR** light has the following meaning:

Off	Normal
Green	Errorred Seconds
Orange	Bursty Error seconds
Red	SES

Note: Orange= Red + Green

By pressing the **select** button, you can cycle through the different tests:

- **ST** = Self Test. The unit is looped back locally and a test pattern is transmitted. The Self Test will always fail if the unit is plugged into a working line.
- **DL**= Digital Loopback. The unit is looped back locally, as well as looped back through the line. The Telco can send a signal to place the DSU/CSU in this mode for testing.
- **LL**= Local Loopback. Like Self Test, except that no automatic signal is generated. In both **DL** and **LL**, what you transmit will be received.
- **RT**=Remote Test. This is a round trip test across the network. It will only work if both the DSU/CSU units are Sangoma FT1 devices.

Accessing the monitor data via IP Address

If you are monitoring a Sangoma card that is installed in the same machine as the monitor, use the card name as described above. However, you can also use the monitors to examine cards that are remote, but logically accessible. To do this, you will need to know:

- The UDP Port set in the **Driver Properties** of the card. The Default is **UDP port 9000**. Setting the UDP port to 0 disables the monitor.
- The interface name or an IP address that forces the stack to try to send the UDP packet through the **WANPIPE™** driver. For instance, if you are on a LAN attached to the host machine for the **WANPIPE™** installation, you might use the IP address of the **opposite end of the link**. Note that you would not use the IP address of the **WANPIPE™** card itself if the access was “from the top”, because the stack loopback function returns the UDP packet before it even reaches the driver. If **WANPIPE™** is the default gateway, then any address not on the local network will do. The address does not have to exist, as the UDP

packets are intercepted in the driver before they are transmitted, and are replied to in the form of a UDP response packet given to Windows as if it was received off the link.

If you are coming in through the WAN interface (from “underneath” the driver), then you can use the IP address of the card itself. This is because in this situation, the host’s loopback route is not in the way.

You can use these utilities to monitor any Sangoma device, whether or not it is running Windows. For instance, you can use these graphical utilities to monitor character based systems such as Linux or FreeBSD.

Using the Performance Monitor and SNMP

When you install **WANPIPE™** software a set of performance counters is automatically added to the system that allows you to monitor **WANPIPE™** using the NT or Windows 2000 **Performance Monitor**, and **SNMP**.

You can use **Performance Monitor** in the **Administrative Tools** group to access individual interface statistics and counters. Use the **Report View** and add the appropriate counters as follows:

- Select a computer you would like to monitor (the one that has **WANPIPE™** installed)
- Select the correct object in the Object box.
- The **Instance** box shows **WANPIPE™** devices installed on the system and the **Counters** list box shows the list of available counters.
- Click on **Explain** button for details on counter meanings.
- Click on **Add** button to add the counter to the Performance Monitor window.

Windows allows you to use SNMP to display the performance counters installed on the system. Therefore since the performance counters are installed with **WANPIPE™** you can use SNMP utilities to monitor the **WANPIPE™** devices. Note that you have to have SNMP Service installed on the machine that you wish to monitor.

The `/snmp` directory on the distribution floppy contains several SNMP related files that were automatically generated from the performance counters for WANPIPE by Perf2mib.exe tool from the Windows NT Resource Kit 4.0. Please refer to **SnmprReadme.doc** or **SnmprReadme.txt** documents found in `/snmp` directory for detailed information about installing and working with performance information for **WANPIPE™** via SNMP.

Uninstalling WANPIPE™

Windows NT4

The WANPIPE™ adapter may be easily removed using the **Network** applet in **Control Panel** . Select the adapter from the network adapter card list and click 'Remove'. Note that all the files and registry entries are removed by this step.

Windows 2000/XP

Open "Device Manager" and switch to "Devices by Connection" view. You will see something similar to this:

```
"Sangoma S5145 (56K) PCI Adapter"  
  |  
  ----" Sangoma WAN Connection Driver for S5145 56K card"  
    |  
    ----"Sangoma Frame Relay interface (on S514)"
```

The uninstall process should start “from the bottom up” :
the "Sangoma Frame Relay interface (on S514)" should be uninstalled first,
then " Sangoma WAN Connection Driver for S5145 56K card" and then
"Sangoma S5145 (56K) PCI Adapter".
The program group “Sangoma Wanpipe” in “Start->Programs” should be removed manually.

4. WANPIPE™ for Windows 95

Introduction

Currently, **WANPIPE™** is only supported on the ISA S508 under Windows 95. Only Frame Relay and PPP are supported. Frame Relay is also supported under Windows 9x on the S514 PCI card.

If the client for Microsoft Networks is installed, the Frame Relay/PPP link may be used to communicate with another Windows NT or Windows 95 machine using the TCP/IP or IPX protocol. The host running **FPIPE/PTPIPE** will look as if it is on the Microsoft Network. Note that for TCP/IP, **FPIPE** running in a Windows 95 environment can be used as a static router for a Local Area Network. To use Windows 95 as a router, you must install the **W95PATCH** included on the Utilities diskette.

FPIPE/PTPIPE may also be used by the Microsoft NetWare Client to connect to third party routers such as Novell's MultiProtocol Router (MPR) 3.0 using IPX, or it may be used for TCP/IP access to NetWare servers using FTP or TELNET.

A typical use for **FPIPE/PTPIPE** is to connect to an Internet provider, either as a standalone machine or as an Internet gateway for a LAN. All the TCP/IP tools that come bundled with Windows 95 may be used, as well as any other 16 or 32-bit Winsock 1.1-compliant application that is capable of running on Windows 95 and using its protocol stack. The bundled utilities include **FTP, TELNET, PING, ROUTE, NETSTAT**, etc.

Frame Relay Support for Windows 95: FPIPE

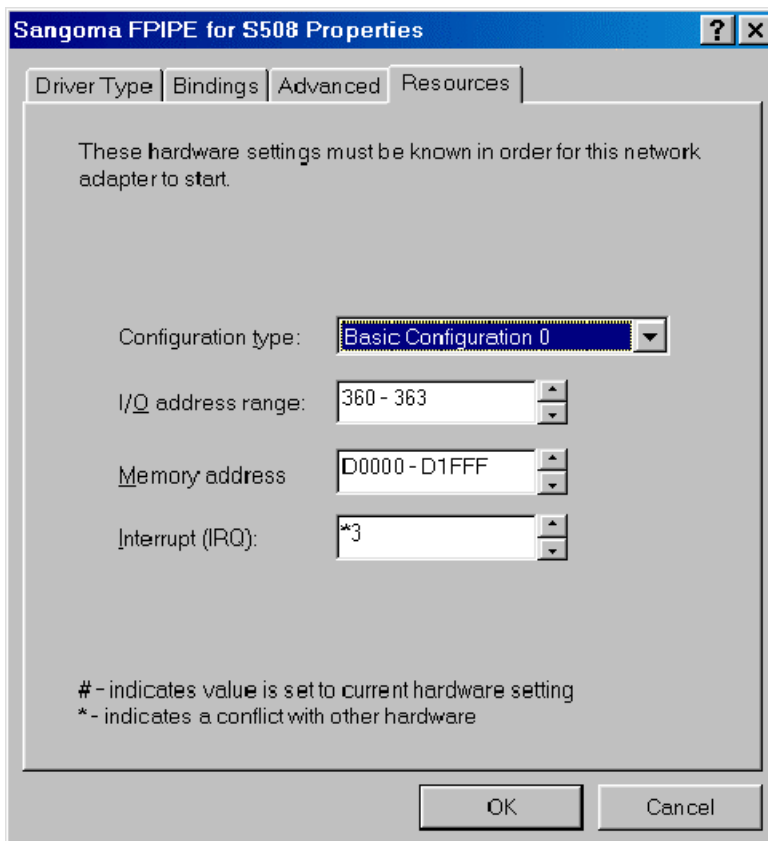
Installation

Install the card (see Section 3.) and if necessary, configure the integrated CSU/DSU under MS-DOS (See Section 4.).

Boot Windows 95, and follow these steps.

1. Open **Control Panel**.
2. Run the **Network** applet. **WARNING: Do not use the 'Add New Hardware' applet to install FPIPE!**
3. On the 'Configuration' tab (default), click on the 'Add...' button.
4. Select 'Adapter' from the network component list and click on 'Add . . .'
5. Click on the 'Have Disk . . .' button.
6. Insert the **WANPIPE™** distribution disk in the drive and choose the appropriate source path such as **A: \FPIPE\508\WIN95** (it defaults to A: \).
7. Select the model 'Sangoma **FPIPE** from the list and click 'OK'.

8. You are now back on the main configuration screen. Select the appropriate adapter from the list and click on '**Properties...**'.
9. Chose the '**Resources**' tab and fill in the I/O port, Memory Address and IRQ to be used by the card. Only the I/O port is set by jumpers. Chose values that do not conflict with other hardware. The settings you chose must match any board jumper settings. Use **SNOOPER** if you need to resolve hardware conflicts.
9. Choose the 'Advanced' tab at the top.
10. Edit the properties accordingly. A detailed explanation of these properties may be found in the 'Driver Properties' section.
11. Click the 'OK' button



12. Configure the TCP/IP and/or IPX/SPX protocol bindings accordingly. If they are not installed yet, add the desired protocols by clicking on the '**Add...**' button and then select '**Protocol**'. You may remove the NetBEUI binding if you wish, but is not necessary as the driver will discard all NetBEUI packets. If you will not be using IPX, set TCP/IP to be the default protocol in the 'Advanced' tab of TCP/IP. If you will be using IPX, it is advisable that you set the **Frame Type** advanced property to **Ethernet II**. For more

information on setting up TCP/IP and/or IPX/SPX, see the Windows 95 Resource Kit on Networks and the Sangoma IP/IPX Tutorial.

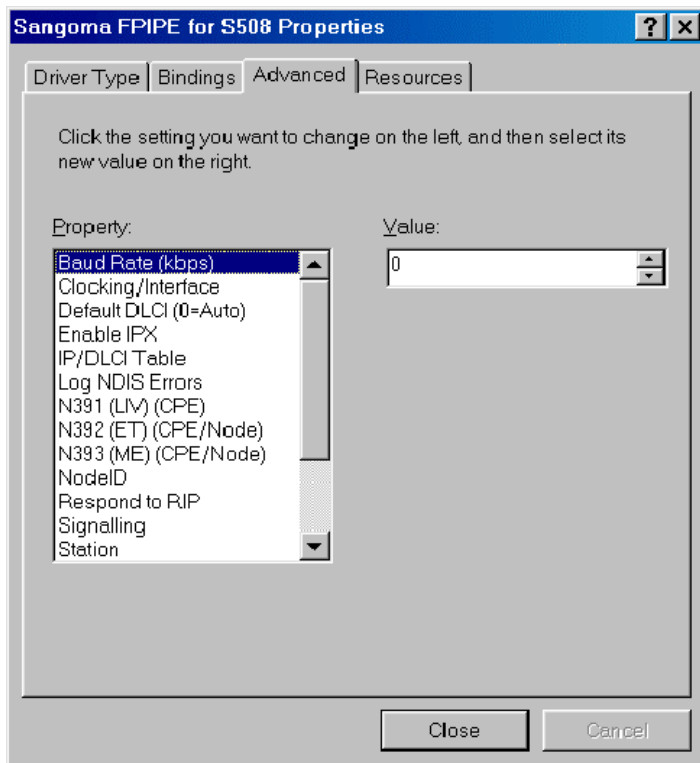
13. Once you click 'OK' on the main settings window, files will be copied from the distribution disk. In addition, it will ask you for some Windows 95 distribution disks. Note that it will always ask to copy files even if they have already been installed previously.
14. Restart the computer when prompted, and of course remove the floppy.
15. If you plan to connect to MPR 3.0 using IPX, then refer to the section 'Connecting to Novell's MPR' in this manual.
16. If you plan on using Windows 95 as an IP router for an attached LAN, an extra key value must be added to the registry. Run **REGEDIT** and select the following registry key:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP.

Then add the value **EnableRouting** = "1". *Apply the VIP.386 patch provided on the Utilities disk*, and restart the computer again.

Driver Properties

The following is a detailed description of each driver property that is included in the **Advanced Properties** tab of the Sangoma **FPIPE** adapter. The defaults shown are OK for most installations, and should only be changed for a good reason.



<u>Property</u>	<u>Value</u>	<u>Description</u>
Baud Rate	0-2600	Line speed in kbps if locally generated
Clocking/ Interface	INTERNAL	Clock generated by S508 on auxiliary clock pins.
	EXTERNAL	Clocking provided externally. Use this for any connection that uses a DSU/CSU over a real network.
RS-232		Interface is RS-232.
V.35		Interface is V.35 or X.21 or EIA530.
I/O Port Base (JP1)		See JP1 table in hardware installation.
Interrupt Num IRQ		One of: 3, 4, 5, 7, 10, 11, 12.
IP/DLCI Table	(filename)	Filename of IP/DLCI routing table. It must be in \windows\system . If the file does not exist, no multiple DLCI routing will take place, and the default DLCI will be used for all IP and IPX traffic.
Shared Memory	XY000	Physical starting address (20 bits) of shared memory window of size 8192 bytes. X = { A, C, D, E } Y = { 0, 2, 4, 6, 8, A, C } Try D0000 or C8000 , or as found by SNOOPER . EXE.
DLCI	16-991	Logical channel number to use for sending and receiving packets.
	0	Enable auto-selection of DLCI. You can get the DLCI from your network provider. SNOOPER can find which ones are active. On a back-to-back connection (no Frame Relay network) use 16. Auto-selection will choose the first active DLCI when the driver loads if configured as CPE, and will choose 16 if Access Node.
Enable IPX	NO YES	All IPX packets (Ethertype 8137) discarded. IPX packets handled.
Log NDIS Err	YES	Frame Relay channel status messages are logged to \windows\ndislog.txt . This is not recommended by Microsoft.
	NO	Messages are not logged.
N391	1-255	Full Status Polling Cycle counter (CPE) - generally set to 6.
N392	1-10	Error Threshold counter (CPE and NODE) - generally set to 3.
N393	1-10	Monitored Events counter (CPE and NODE) - generally set to 4.

<u>Property</u>	<u>Value</u>	<u>Description</u>
NodeID	4 hex bytes	Refer to “More on IPX and NodeID” section.
Respond to RIP	NO YES	Not connecting to IPX router. Connecting to IPX router.
Station	CPE	Customer Premises Equipment (CPE) configuration. Use when connecting to a real Frame Relay network or as one end of a back-to-back connection.
Node		Access Node (switch) configuration. This is only used as one end of a back-to-back connection.
T391	5-30	Link Integrity Verification timer in secs (CPE) - generally set to 10.
T392	5-30	Polling Verification timer in secs (NODE) - generally set to 15.
UDP Port	0-65535	FPIPEMON manager port. Normally set to 9000 unless it conflicts with another service. Set to 0 to disable it.
TTL	0-255	Time-to-live parameter in IP packet set by FPIPE in reply to FPIPEMON. Set to 1 to prevent the reply from being routed away from the host.
Win95 Platform	YES	Always YES - internal use only.

IP/DLCI Routing Table

See the equivalent section in the Windows NT/Windows 2000 section.

Note that Windows 95 only supports 1 subnet, and so each DLCI must represent a subnet of the subnet.

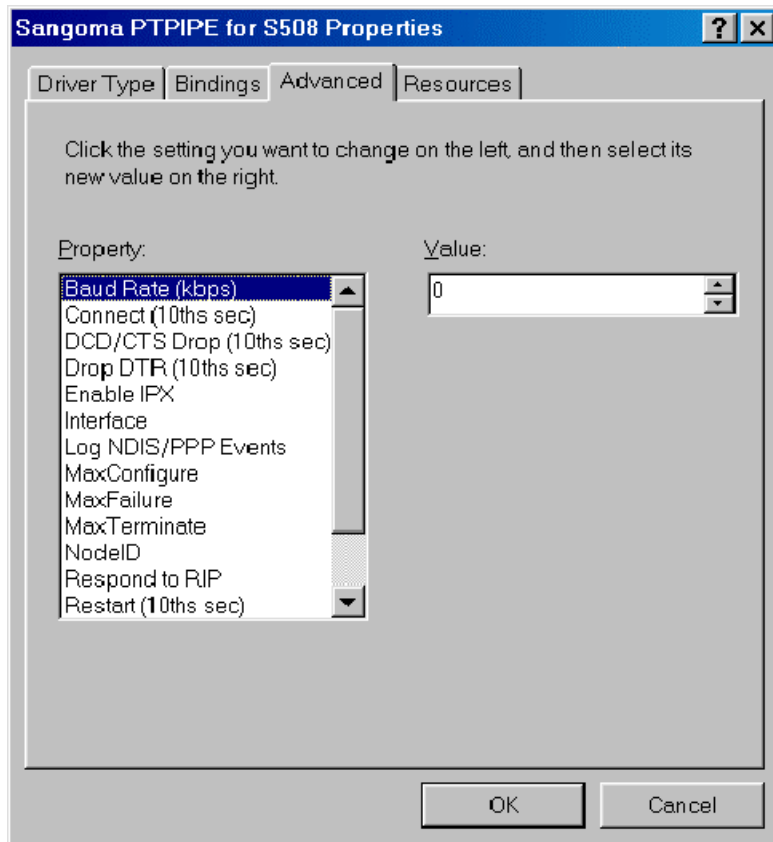
Point-to-Point Support for Windows 95: PTPIPE

Installation

Once the card has been installed, boot Windows 95, and follow these steps.

1. Open 'Control Panel'.
2. Run the **Network** applet. **WARNING:** Do **not** use the 'Add New Hardware' applet to install PTPIPE!
3. On the 'Configuration' tab (default), click on the 'Add...' button.
4. Select 'Adapter' from the network component list and click on 'Add. ...'.
5. Click on the 'Have Disk...' button.

6. Insert the **WANPIPE™** distribution disk in the drive and choose the appropriate source path, such as A:\PTPIPE\508\WIN95 (it defaults to A:\).
7. Select the model 'Sangoma PTPIPE' from the list and click 'OK'.
8. You are now back on the main configuration screen. Select the 'Sangoma PTPIPE' or 'Sangoma PTPIPE508' adapter from the list and click on 'Properties...'
9. Choose the 'Advanced' tab at the top.



10. Edit the properties accordingly. A detailed explanation of these properties may be found in the 'Driver Properties' section.
11. Click the 'OK' button
12. Configure the TCP/IP and/or IPX/SPX protocol bindings accordingly. If they are not installed yet, add the desired protocols by clicking on the 'Add...' button and then select 'Protocol'. You may remove the NetBEUI binding if you wish, but is not necessary as the driver will discard all NetBEUI packets. If you will not be using IPX, set TCP/IP to be the default protocol in the 'Advanced' tab of TCP/IP. If you will be using IPX, it is advisable that you set the **Frame Type** advanced property to **Ethernet II**. For more information on setting up TCP/IP and/or IPX/SPX, see the Windows 95 Resource Kit on Networks and the Sangoma IP/IPX Tutorial.
13. Once you click 'OK' on the main settings window, files will be copied from the distribution disk. In addition, it will ask you for some Windows 95 distribution disks. Note that it will always ask to copy files even if they have already been installed previously.

14. Restart the computer when prompted, and of course remove the floppy.
15. If you plan to connect to MPR 3.0 using IPX, then refer to the section 'Connecting to Novell's MPR' in this manual.
16. If you plan on using Windows 95 as an IP router for an attached LAN, an extra key value must be added to the registry. Run REGEDIT and select the following registry key:
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\VxD\MSTCP.

Then add the value **EnableRouting = "1"**. Apply the VIP.386 patch provided, and restart the computer again.

Driver Properties

The following is a detailed description of each driver property that is included in the 'Advanced Properties' tab of the Sangoma PTPIPE adapter. The defaults shown are OK for most installations, and should only be changed for a good reason.

<u>Property</u>	<u>Value</u>	<u>Description</u>
Baud Rate	0-2600	Line speed in kbps if locally generated
Clocking/ Interface	INTERNAL	Clock generated by S508 on auxiliary clock pins.
	EXTERNAL	Clocking provided externally. Use this for any connection that uses a DSU/CSU over a real network.
RS-232		Interface is RS-232.
V.35		Interface is V.35 or X.21 or EIA530.
I/O Port Base (JP1)		See JP1 table in hardware installation.
Interrupt Num IRQ		One of: 3, 4, 5, 7, 10, 11, 12.
IP/DLCI Table	(filename)	Filename of IP/DLCI routing table. It must be in \windows\system . If the file does not exist, no multiple DLCI routing will take place, and the default DLCI will be used for all IP and IPX traffic.
Shared Memory	XY000	Physical starting address (20 bits) of shared memory window of size 8192 bytes.
		X = { A, C, D, E } Y = { 0, 2, 4, 6, 8, A, C } Try D0000 or C8000 , or as found by SNOOPER . EXE.
Enable IPX	NO	All IPX packets (Ethertype 8137) discarded.
	YES	IPX packets handled.
Respond to RIP	NO	Not connecting to IPX router.
	YES	IPX packets handled.
NodeID	4 hex bytes	Refer to "More on IPX and NodeID" section.
Log NDIS Err	YES	Frame Relay channel status messages are logged to \windows\ndislog.txt. This is not recommended by Microsoft.
	NO	Messages are not logged.
Restart	1-2621	Time in 10ths of a second between successive retransmissions of Configure-Request or Terminate-Request packets. Recommended value is 30.
		Time to wait in 10ths of a second since DCD or CTS dropped before declaring the link disconnected. Recommended value is 10.
DCD/CTS Drop	1-2621	Time to wait in 10ths of a second since DCD or CTS dropped before declaring the link disconnected. Recommended value is 10.
Drop DTR	1-2621	Minimum time in 10ths of a second to keep DTR and CTS low during disconnection. This is normally greater than DCD/CTS Drop. Recommended value is 15.

<u>Property</u>	<u>Value</u>	<u>Description</u>
Connect	0-2621	Time to wait in 10ths of a second for DCD to go high during a connection attempt before dropping DTR to hang up. If this peer must wait for incoming connections, this should be set to zero.
MaxConfigure	1-65535	Number of Configure-Request packets sent without receiving a reply before terminating the link. Recommended value is 10.
MaxTerminate	1-65535	Number of Terminate-Request packets sent without receiving a reply before terminating the link. Recommended value is 2.
MaxFailure	1-65535	Number of Configure-Nak packets sent without sending a Configure-Ack before assuming the configuration is not converging. In some cases, the Configure-Nak packets are then converted to Configure-Reject packets; otherwise the link will be immediately terminated. Recommended value is 5.
UDP Port	0-65535	PPIPEMON manager port. Normally set to 9000 unless it conflicts with another service. Set to 0 to disable it.
UDP TTL	0-255	Time-to-live parameter in IP packet set by PPIPE in reply to PPIPEMON. Set to 1 to prevent the reply from being routed away from the host
Win95 Platform	YES	Always YES - internal use only.

Asynchronous autoconnect mode is always enabled. This mode persistently attempts to connect to the peer the moment the driver is started. If a connection is not desired, the physical link must be disconnected (by turning off the synchronous modem for example).

Authentication is currently not supported under Windows 95.

IPX and NodeID

Windows 95 will only connect to one NetWare Server. Therefore, if you have a local NetWare server, it is better to use it as the router rather than a Windows 95 machine. Use Windows 95 as a router only to connect to a *remote* NetWare Server, if using **NWLink IPX/SPX Compatible Transport**.

The value is used to construct the 6-byte physical address of the card, and is the same number as the **IPX Network Address** if IPX is to be used. This is also called the Network Number of the LAN segment.

Given that the four consecutive hexadecimal bytes of **NodeID** are N0, N1, N2, and N3, then the physical address of the card will be:

02-N0-N1-N2-N3-00.

If only TCP/IP will be used, the **NodeID** may be set to any hexadecimal value between 00000001 and FFFFFFFE, and does not need to be unique. The default value is 0000FEFE. The card address is used locally only and will not appear on the network in this case.

If the **Microsoft IPX/SPX-compatible protocol** is used, it is necessary that in the 'Advanced Properties' tab for this protocol, the **Network Address** value must either correspond exactly to the **NodeID** value, or it must be zero. The **NodeID** value in this case must be unique from all other Network Numbers or Internal Network Numbers.

If you are connecting to another Windows machine, such as NT Server, it is recommended that you use TCP/IP only, if this is possible, as it would allow the remote machine to route packets onto other network segments, if any.

Diagnostics: Using FPIPEMON and PPIPEMON

See the equivalent subsection in the Windows NT/Windows 2000 section. The monitor programs work identically on all Windows platforms. In fact, you can run the monitor on a Win 95 machine to debug a Win NT/2000 host and vice versa. You can also monitor other types of hosts such as Linux, Unix or NetWare hosts.

Adapter and/or Driver Startup Problems

If no message appears when Windows is started to the effect that the network device is not operating, then it has likely started with no problems. To check if the device is functioning at any time, run the **System** applet in 'Control Panel' and choose the 'Device Manager' tab.

The 'Properties' of the 'Sangoma FPIPE' or the '**Sangoma PPIPE**' adapter should show the device is functioning properly. If not, there was a startup problem.

Whether or not logging was enabled in the 'Driver Properties', the file `\windows\ndislog.txt` will always contain serious initialization errors as recorded by the operating system. If enabled, and a frame relay-level error occurred just as the driver was started (likely due to a network client trying to discover network resources), a window will appear on startup indicating the adapter is not functioning properly. If this problem persists, either disable error logging in the 'Driver Properties', or configure the Microsoft Network Client to connect to drives on demand only, if used.

This file is purged each time the system is booted, and so shows errors relating to the current session only. The format of an error in this file is:

```
Sangoma FPIPE (000?). Error Code=0X(-----)
[, 0X(-----), ...]
```

or

```
Sangoma PPIPE (000?). Error Code=0X(-----)
[, 0X(-----), ...]
```

Error Codes may be followed by extra information in the form of 32-bit hexadecimal numbers. These numbers will be referred to in sequence by the notation E0, E1, E2, etc. from left to right after the Error Code.

The following error codes may be reported in this file that relate to initialization:

<u>Error Code</u>	<u>Extra</u>	<u>Reason/Description</u>
0XC0001389	E0	Exceeded the number of configurable DLCIs. Maximum number.
		<i>(Frame Relay only)</i>
0XC000138A	E0	Unable to load the Frame Relay or PPP code onto the adapter. Interrupt number.
	E1	DLCI.
	E2	I/O port base address.
	E3	Shared memory base address.
		The following problems would give this error:
	a)	The card was not found at the specified I/O port.
	b)	The specified shared memory address is invalid.
	c)	The Frame Relay or PPP code version is incorrect.
		Reasons (b) and (c) will be reported individually just before this error is logged. If the reason is (a) then make sure the specified I/O port is correct and that it matches the jumper setting; otherwise, try another port address.
0XC000138C	E0	Unable to register interrupt routine. Status code returned by NdisMRegisterInterrupt.

	E1	Interrupt number. Try another interrupt number, and make sure it does not conflict with another device. IRQ 7 has been known to work fine except for a possible conflict with the printer driver. IRQ 5 may conflict with a LAN card. IRQ 3 may conflict with COM2. IRQ 2 (9) may not work.
0XC000138E	E0	Bad Frame Relay or PPP code version. Code version read from card after code was downloaded (0X30302E34 represents v4.00). You should have the correct code, so this error will likely be a consequence of the code not being downloaded correctly. Try another shared memory address. If this fails, try another port address. If this fails, the file <code>FPIPE.502/508</code> or <code>PTPIPE.502/508</code> may be corrupt. Record E0 and contact us for more information.
0XC0001390	E0	DLCI in routing table is out of range. It must be between 16 and 991. Rejected DLCI. <i>(Frame Relay only)</i>
0XC0001391	E0	The specified shared memory segment is invalid. Segment (a value other than the correct ones: 0A, 0C, 0D, 0E). An invalid shared memory address was specified. This error should not occur if the address was chosen from the list presented when configuring the card. If the registry entry was changed manually, it will have been done so incorrectly.

The following status codes may be reported in this file that relate to Frame Relay setup, but appear only if logging is enabled:

<u>Error Code</u>	<u>Extra</u>	<u>Reason/Description</u>
0X40001390	E0	A DLCI was auto-selected and will be used for the entire Windows session. DLCI value.

Communication error log

If the card loads without error, you should use FPIPEMON or PPIPEMON for link debugging of the link. We do, however, provide event logging that can also be used for line debugging.

Make sure Error Logging is ON. Check the 'Driver Properties' and ensure 'Log NDIS Errors' is ON by using the 'Network' applet in 'Control Panel' as described in the installation section. If it is not ON, it will be difficult to diagnose problems.

Status codes are logged to `\windows\ndislog.txt` in the following format:

For Frame Relay:

```
Sangoma FPIPE (000?). Error Code=0X4000138A,  
<Current_Error_Code>, <Last_Error_Code>, <Last_Error_Count>
```

The **Current_Error_Code** is the error that caused this line to appear in the log. The **Last_Error_Code** is the last error that occurred before this one. **Last_Error_Count** is the number of additional times **Last_Error_Code** occurred since the time it was last reported.

For PPP:

```
Sangoma PTPIPE (000?). Error Code=0X4000138A, <Event_Code>,  
<States>, <Connection_Status>
```

The **Event_Code** is the event that caused this line to appear in the log. The **States** value reflects the state of the PPP link at the time. **Connection_Status** is a value that indicates the modem line status and reasons for disconnection if any.

The following table lists the Frame Relay and PPP error codes that may occur when trying to send a packet. If an 'R' appears after the error code, then the packet send attempt will be retried a bit later until it succeeds.

<u>Error Code</u>	<u>Reason</u>
0X00000000	The current attempt succeeded. This is logged if an error occurred beforehand and the number of consecutive errors of the same type needs to be reported. (<i>Frame Relay only</i>)
0X00000001	rotocol open event took place. This occurs when IP and/or IPX network control protocols signal an up event. Data packets may be transferred in this phase. (<i>PPP only</i>)
0X00000001	ommunications has not been enabled.

The channel should be in a data transfer state when the driver and card are initialized. If for some reason this error occurs, you will likely have to reboot and try again. (**Frame Relay only**)

- 0X00000002 he channel is currently inoperative.
It may be that the link was temporarily brought down at the other end.
Wait a while and try sending again. (**Frame Relay only**)
- 0x00000002 isconnection event took place.
The reason for the disconnection is set in **Connection_Status**. (**PPP only**)
- 0X00000003 he DLCI is currently inactive.
The logical connection should be active, unless it was made inactive at the other end. Wait a while and try sending again. Verify that you are using the correct DLCI - check the 'Driver Properties'. (**Frame Relay only**)
- 0x00000003 TR raised to retry connection.
On expiry of the **Drop DTR** timer property after a local disconnection, DTR is raised again with the intention to retry the connection. (**PPP only**)

The following errors apply to **Frame Relay only**:

- 0X00000004 he DLCI is invalid.
Verify that you are using the correct DLCI - check the 'Driver Properties'.
Otherwise, the problem may be at the Access Node end.
- 0X00000005 uffer length error.
This will not occur unless the **FPIPE . 502/508** file is corrupt, or somehow the operating system does not properly detect the MTU of the driver. The MTU must be 1500 bytes.
- 0X00000006 uffer length/flow control error.
This will not occur unless the **FPIPE . 502/508** file is corrupt.
- 0X0000000A Hardware timeout.
This is a very severe error, and indicates that the code running on the card most likely crashed. The link will be in an unknown state.
- 0X00000010 odem failure occurred - DCD and/or CTS found low
- 0X00000011 hannel status changed from OPERATIVE to INOPERATIVE
- 0X00000012 R Channel status changed from INOPERATIVE to OPERATIVE

0X00000013 R	Access Node reported a change in the status of one or more DLCIs. This notification will always be logged when the driver and card are initialized when Windows is started, so long as it is configured to be CPE. If this is not logged, it is likely that there is a problem at the other end.
0X00000014 R	A change in the DLCI configuration list was made by the Access Node. It may be possible that the DLCI being used is now invalid - this will be reported on the next send attempt.
0X0000001F R	Invalid command. Please contact us immediately if you encounter this error. There is a good chance that Windows may crash very badly if this error was encountered.
0XFFFFFFFF	Indicates Windows shut down. This is reported if an error occurred previously, and the number of consecutive errors needs to be reported. Unfortunately, Windows seems to purge the log file each time it starts, and so this particular entry may never be observed.

In the case of a **PPP connection**, the following additional parameters have meaning:

Let the hexadecimal value of **States = 0xQQRSSSTT**. The following tables explain the values of QQ, RR, SS, TT.

<u>IPXCP STATE (0xQQ)</u>	<u>Description</u>
0x09	The IPX protocol is open for packet transfer.
0x??	Any other value indicates IPX packet transfer will not be permitted (yet). Consult RFC 1552/1661 for more information.

<u>IPCP STATE (0xRR)</u>	<u>Description</u>
0x09	The IP protocol is open for packet transfer.
0x??	Any other value indicates IP packet transfer will not be permitted (yet). Consult RFC 1332/1661 for more information.

<u>PPP PHASE (0xSS)</u>	<u>Description</u>
0x00	The link is dead.
0x01	Link Establishment (LCP) phase.
0x02	Authentication phase (PAP/CHAP).
0x03	Network Layer phase (IP/IPX).
0x04	Link Termination phase.

LCP STATE (0xTT)

escription

0x09	LCP is open, allowing data transfer at upper layers.
0x??	Any other value indicates LCP is not up. Consult RFC 1661 for more information.

Let the hexadecimal value of **Connection_Status** = **0xWWXXYYYY**. The following tables explain the values of WW, YYYY. The value XX is reserved.

DCD/CTS (0xWW)

escription

0x00	Neither DCD nor CTS are high.
0x08	DCD is high and CTS is low.
0x20	DCD is low and CTS is high.
0x28	DCD and CTS are high.

Disconnection Causes (0xYYYY)

Bit Set

escription

0	Physical link termination by local request.
1	DCD and/or CTS dropped indicating link termination.
2	Local disconnection request.
3	Inbound/outbound authentication failed.
4	Failed to negotiate inbound authentication protocol.
5	Rejected peer's request for authentication.
6	Peer rejected MRU option on LCP Configure-Request.
7	MRU of peer was below required minimum.
8	Rejected peer's LCP option(s) too many times.
9	Rejected peer's IPCP option(s) too many times.
10	Rejected peer's IPXCP option(s) too many times. This should not occur as IPXWAN is used which requires IPXCP to not negotiate any particular configuration.

Note the Maximum Receive Unit (MRU) of **PTPIPE** is 1500.

Uninstalling WANPIPE

The FPIPE/PTPIPE adapter may be easily removed using the **Network** applet in 'Control Panel'. Select the adapter from the network component list and click 'Remove'.

If you wish to install it again at a later time, the setup information file and the driver files will still remain on the system, and so it may be added again without the use of the distribution disk.

If you wish to completely remove all related files, delete the following:

\windows\inf\netsango.inf
\windows\system\fpipesys.sys
\windows\system\fpipesys508.sys
\windows\system\fpipesys502
\windows\system\fpipesys508
\windows\system\frmon.exe
\windows\system\fpipesysbl.sam

for Frame Relay, or

\windows\inf\netsango.inf
\windows\system\ptpipesys.sys
\windows\system\ptpipesys8.sys
\windows\system\ppipemon.exe
\windows\system\ptpipesys502
\windows\system\ptpipesys508

for PPP.

Note that \windows\ndislog.txt is part of the NDIS logging system, and does not belong to WANPIPE.

If your Windows 95 directory is not \windows, just substitute it for the correct name throughout the manual.

5. WANPIPE™ LAN HSM for NetWare

Introduction

The **FPIPE (Frame Relay)** and **PTPIPE (PPP)** LAN HSM (Hardware Specific Module) is incorporated into the networking system of NetWare, and behaves as a standard Ethernet driver. It functions as a router by encapsulating both IP and IPX packets for transmission over Frame Relay, or synch. Point-to-Point or as a bridge by transparently transporting Ethernet_II frames from one NetWare server to another.

In addition, **FPIPE** is capable of bridging two NetWare servers over a Frame Relay link.

When functioning as a router, **FPIPE** will be able to connect a NetWare server on one LAN to NetWare's MultiProtocol Router (MPR) on another LAN. When functioning as a Frame Relay bridge, two NetWare servers on separate LANs may be connected as if the link were another LAN network segment. In either case, **FPIPE** provides the means to connect NetWare networks together without the need for an external router or bridge.

Currently, only the ISA S508 is supported under NetWare.

Installation

The **FPIPE** HSM package includes the following files:

FPIPE508.LAN	Ethernet_II LAN driver (S508)
FPIPE.CFG	Command line parameter file
PINSTALL.NLM/HLP PSCRIPT.DAT	Redistributable installation program by Novell INSTALL program product information file

The **PTPIPE** HSM package includes the following files:

PTPIPE8.LAN	Ethernet_II LAN driver (S508)
PTPIPE.CFG PINSTALL.NLM/HLP PSCRIPT.DAT	ommand line parameter file Redistributable installation program by Novell INSTALL program product information file

A separate file is used to contain the driver's command line parameters since there are so many. It is possible to not use this file if the appropriate parameters may fit on the load command line. See the following section 'Driver Properties' for a detailed description of every parameter and the default values.

FPIPE/PTPIPE may be installed on a server with or without MPR 3.0. The INSTALL NLM is used on both configurations for installing the product on the system. If you have MPR, you may use **INETCFG** to add the board. Otherwise, you will have to use **INSTALL** to edit the **AUTOEXEC.NCF** file which will load the driver. The following sections describe this process in detail.

Installing FPIPE or PTPIPE

Note: The distribution disk must have the volume label **FPIPE** for Frame Relay or **PTPIPE** for PPP for **PINSTALL** to recognize it.

1. Boot the server and run **INSTALL**:
`<server>:LOAD INSTALL`
2. Select 'Product Options' menu.
3. Hit the **INS**ert key to add a new product to the list.
4. Insert the **FPIPE** or PTPIPE distribution disk in the appropriate drive and enter the path.
5. **PINSTALL** will run and you will be presented with 'Installation Options'. Choose the desired installation option. The files will then be copied.
6. Exit **PINSTALL**. You will return to the **INSTALL** program with the FPIPE or PTPIPE in the installed products list.
7. Select the FPIPE or **PTPIPE** product and hit **ENTER**. You may now edit the driver parameter file. See the section 'Driver Properties'.
8. Proceed to step 9 in the appropriate section below, depending on whether or not you have MPR and **INETCFG** installed. If the S508 driver is installed, change all FPIPE and PTPIPE references in the following steps to FPIPE508 and PTPIPE8 respectively.

Adding Network Board with INSTALL (no MPR)

9. Exit to the main menu of the **INSTALL** program.
10. Select 'System Options' menu.
11. Choose 'Edit AUTOEXEC.NCF File'.
12. Add the following lines:
`load fpipe @fpipe.cfg`
`bind ipx to fpipe net=<net_number>`

And optionally,

```
bind ip to fpipe address=<ip_addr> mask=<net_mask>
gate=<ip_addr>
```

which requires the **TCPIP** module to be loaded first. The IPX **net_number** may be chosen arbitrarily so long as it is unique throughout the internetwork. It is a 4-byte hexadecimal number.

13. Save and exit, and exit **INSTALL**. Reboot the server.

Adding Network Board with INETCFG (MPR)

9. Exit the **INSTALL** program.
10. Run **INETCFG**:

```
<server>:LOAD INETCFG
```
11. Select 'Boards' menu.
12. Hit the **INS**ert key to add a new board to the list.
13. Select the **FPIPE** or **PTPIPE** driver from the list. It will tell you that no description file exists for the driver. Proceed anyway.
15. You must now enter information for the following items:
Board Name: FPIPE or PTPPIPE (or something similar)
Board Parameters: @fpipe.cfg or ptpipe.cfg (or type them in here if there is sufficient space)
Comment: (any comment you like)
If you type the parameters in this field directly, the contents of fpipe.cfg or ptpipe.cfg will be ignored.
16. Exit and save, and select 'Bindings' from the main menu.
17. Select 'IPX', and then choose the **FPIPE** or **PTPIPE** interface.
18. Enter the **Network Number**. It may be chosen arbitrarily so long as it is unique throughout the internetwork. It is a 4-byte hexadecimal number. For **Frame Type**, you must choose **Ethernet_II**.
19. Exit and save all changes, and then reboot the server.

Driver Properties

Each property in the table below corresponds to a load command line keyword. If a value is to be specified, the command line argument will be of the form **Property = Value**. Otherwise, '**Present**' and '**Absent**' under the **Value** column indicates whether or not the property keyword will appear on the command line and the effects either will have.

Property keywords and values are not case sensitive.

<u>Property</u>	<u>Value</u>	<u>Description</u>
INT	IRQ	One of: 3, 4, 5, 7, 10, 11, 12.
PORT	(JP1)	See JP1 table in hardware installation.
RS232	Absent	RS-485 interface.
	Present	RS-232 interface.
SPEED	0-2600	Line speed in kbps. See hardware installation. Must be set to zero if clocking is external.
Default: 56		
MEM	XY000	Physical starting address (20 bits) of shared memory window of size 8192 bytes. X = { A, C, D, E } Y = { 0, 2, 4, 6, 8, A, C } Try D0000 or C8000, or as found by SNOOPER . EXE.
DLCI	16-991	Logical channel number to use for sending and receiving packets.
	Absent	Enable auto-selection of DLCI. You can get the DLCI from your network provider. SNOOPER can find which ones are active. On a back-to-back connection (no Frame Relay network) use 16. Auto-selection will choose the first active DLCI when the driver loads if configured as CPE, and will choose 16 if Access Node.
SWITCH	Absent	Customer Premises Equipment (CPE) configuration.
	Present	Access Node (switch) configuration (normally used for back-to-back testing, or for a direct connection between two servers without the use of an external Frame Relay network).
EXTERNAL	Absent	Clock generated by S508.
	Present	Clocking provided externally.
T391	5-30	Link Integrity Verification timer in secs (CPE). Default: 10
T392	5-30	Polling Verification timer in secs (NODE). Default: 15
N391	1-255	Full Status Polling Cycle counter (CPE). Default: 6
N392	1-10	Error Threshold counter (CPE and NODE). Default: 3
N393	1-10	Monitored Events counter (CPE and NODE). Default: 4
NODE	6 hex bytes	An arbitrarily chosen physical node address. Bit 41 should be set high to indicate a local address. It must only be unique to this network, which will consist of only two nodes. Default: FFFFFFFFFFFFFFFF
FRAME	Ethernet_II	Must always be this value.
BRIDGE	Absent	Routing Mode set - expects MPR or other router at other end.
Present		Bridging Mode set - expects FPIPE HSM at other end.

NOTIFY	Absent	No router connection console messages. Present <code>FPIPE</code> router IPXWAN connection messages will be displayed on the console when negotiating with MPR - ignored if in Bridging Mode.
QUEUE	Absent	Discard queued packets for transmission if they arrive faster than they can be sent (recommended under most circumstances).
	Present	Keep packets in adapter queue until they can be sent on the wire. * See the 'Troubleshooting' section.
IPONLY	Absent	Permits transmission of IP and IPX.
	Present	Discards IPX packets on transmission.

General Properties for PPP

MEM	XY000	Physical starting address (20 bits) of shared memory window of size 8192 bytes. X = { A, C, D, E } Y = { 0, 2, 4, 6, 8, A, C } Try D0000 or C8000, or as found by <code>SNOOPER.EXE</code> .
NODE	6 hex bytes	An arbitrarily chosen physical node address. Bit 41 should be set high to indicate a local address. It must only be unique to this network, which will consist of only two nodes. Default: FFFFFFFFFF
FRAME	Ethernet_II	Must always be this value.
NOTIFY	Absent	No router or PPP connection console messages. Present <code>PTPIPE</code> router IPXWAN and PPP connection messages will be displayed on the console.
QUEUE	Absent	Discard queued packets for transmission if they arrive faster than they can be sent (recommended under most circumstances).
	Present	Keep packets in adapter queue until they can be sent on the wire. * See the 'Troubleshooting' section.
IPONLY	Absent	Permits transmission of IP and IPX.
	Present	Discards IPX packets on transmission.
RESTART	1-2621	Time in 10ths of a second between successive retransmissions of Configure-Request or Terminate-Request packets. Default: 30
DCDCTS	1-2621	Time to wait in 10ths of a second since DCD or CTS dropped before declaring the link disconnected. Default: 10
DROPDTR	1-2621	Minimum time in 10ths of a second to keep DTR and CTS low during disconnection. This is normally greater than DCDCTS . Default: 15
CONNECT	0-2621	Time to wait in 10ths of a second for DCD to go high during a connection attempt before dropping DTR to hang up. If this peer must wait for incoming connections, this should be set to zero.

		Default: 0
MAXCONF	1-65535	Number of Configure-Request packets sent without receiving a reply before terminating the link. Default: 10
MAXTERM	1-65535	Number of Terminate-Request packets sent without receiving a reply before terminating the link. Default: 2
MAXFAIL	1-65535	Number of Configure-Nak packets sent without sending a Configure-Ack before assuming the configuration is not converging. In some cases, the Configure-Nak packets are then converted to Configure-Reject packets; otherwise the link will be immediately terminated. Default: 5

Using FPIPE as an Ethernet Bridge over Frame Relay

Two NetWare servers may be connected by an **Ethernet_II** network segment bridged over Frame Relay, so long as the FPIPE HSM is at both ends. When configuring both ends, make sure they share the same physical IPX network number (as specified as a bind parameter) but have different node numbers. The network number must of course be unique, as well as the server internal network numbers. The **BRIDGE** keyword must be on the load command lines of both FPIPE drivers.

Local LAN traffic at each end may be routed across this point-to-point link over a particular logical channel. The frame type used for the LANs may be different than **Ethernet_II**, and this mode of operation has the advantage of being able to handle any packet-level protocol such as IP, IPX, or AppleTalk.

These frames are passed as raw data over the link, without following the RFC 1490 SNAP encapsulation specification. This saves 8 bytes of overhead, but requires that FPIPE be at both ends, as no other device will be able to recognize the content of the frames.

Console Messages

Initialization

FPIPE/PTPIPE-NW-22: Could not allocate memory. Virtual board did not load.

No RCB's were available on startup. It is likely the server is very low on memory. It may have to be restarted.

FPIPE/PTPIPE-NW-50: The board cannot be found.

The board could not be found at the specified I/O port address. Make sure the specified value corresponds to the correct jumper setting. Otherwise, try another port address.

FPIPE/PTPIPE-NW-51: Board RAM failed the memory test.

The specified shared memory address was invalid.

FPIPE/PTPIPE-NW-200: Driver could not initialize due to link failure (rcode RC).

It is possible the cable might be disconnected at the other end. The value of RC will likely be 10, indicating successive modem failures that prevented the board from being initialized properly. Make sure the cable is properly connected at both ends. If the driver must be loaded in the case where the cable is not connected at the remote end, make sure it is unplugged from the board at the host end.

FPIPE/PTPIPE-NW-201: The board is not configured for ETHERNET_II frame type.

The keyword 'frame=ethernet_ii' must appear on the load command line. The driver is not capable of discerning frame types and expects data to be passed to it in this form.

Driver Started

FPIPE/PTPIPE-NW-202:

The board is being reset directly by MSM, and will no longer function properly. **PIPE/PTPIPE** must be unloaded and then loaded to restore proper operation.

If this situation occurs, the driver is not capable of handling it and therefore must be reloaded. This will of course reset the link as well.

Router Mode Messages (only appear if NOTIFY driver property was enabled)

FPIPE/PTPIPE-NW-203:

IPXWAN: Beginning routing type negotiation with remote router.

Indicates that the remote router sent a Timer Request packet and this station has responded with a request to use the Unnumbered RIP/SAP routing type.

FPIPE/PTPIPE-NW-204:

IPXWAN: Failed to respond to remote router's Timer Request packet.

Waiting for the next request...

Indicates that the remote router sent a Timer Request packet and this station attempted to respond, but failed due to a link-related error or notification from the board. The next Timer Request packet should arrive in about 20 seconds.

FPIPE/PTPIPE-NW-205:

IPXWAN: Successfully negotiated Unnumbered RIP/SAP routing type.

Connection will be established on remote reception of FPIPE/PTPIPE router name. Indicates that the remote router sent an Information Request packet and this station has responded. Assuming the remote router received this response, the IPX connection is established.

FPIPE/PTPIPE-NW-206:

IPXWAN: Failed to negotiate Unnumbered RIP/SAP routing type.

Set RIP/SAP to ON, and Header Compression to OFF at remote router. A NAK packet was sent by the remote router indicating that routing type negotiation has failed. Make sure that in the IPX bind options of the remote router that RIP and SAP are ON and NLSP is OFF, as well as IPX header compression being OFF. Enabling header compression may confuse FPIPE/PTPIPE, although it should have been able to negotiate it off.

PTPIPE-NW-207: Disconnection occurred (code DC). (*PPP only*)

The link was disconnected. DC is a hexadecimal number indicating the cause of disconnection as follows:

Disconnection Causes (DC)

<u>Bit Set</u>	<u>Reason</u>
0	Physical link termination by local request.
1	DCD and/or CTS dropped indicating link termination.
2	Local disconnection request.
3	Inbound/outbound authentication failed.
4	Failed to negotiate inbound authentication protocol.
5	Rejected peer's request for authentication.
6	Peer rejected MRU option on LCP Configure-Request.
7	MRU of peer was below required minimum.
8	Rejected peer's LCP option(s) too many times.
9	Rejected peer's IPCP option(s) too many times.
10	Rejected peer's IPXCP option(s) too many times. This should not occur as IPXWAN is used which requires IPXCP to not negotiate any particular configuration.

Note the Maximum Receive Unit (MRU) of PTPPIPE is 1500.

PTPIPE-NW-208:

Protocol opened event occurred (IPCP_STATE: IPS, IPXCP_STATE: IPXS). (*PPP only*) IP and/or IPX packets may be transferred when the IPCP/IPXCP state values IPS and IPXS reach 0x09, indicating the layer is up. Any value other than 0x09 indicates the corresponding protocol is not up. See RFC 1332, 1552, 1661 for more information on states.

Low Throughput and When to 'Queue'

The throughput of the network partly depends on the version of software running on the servers and workstations. NetWare Server v3.11, v3.12, v4.01 without a patch to the Packet Burst algorithm may significantly impede performance over the WAN link. This patch works in

conjunction with updated VLMs on the workstation. If you have v4.10 of the server and the corresponding DOS Workstation VLMs then the packet burst algorithm need not be patched (at this point in time).

This patch is provided with the FPIPE distribution in the **\nwpatch** directory, which was originally acquired from Novell's FTP site (ftp.novell.com). Copy the PBURST.EXE file to a temporary directory and run the program to extract the files. Read the documentation provided with this patch for information on how to install the new packet burst algorithm on your server(s).

As mentioned in the patch documentation, it is possible to change the packet burst window sizes in the DOS VLM client. There is also a way to affect the window size in Windows NT. The following table summarizes the parameter settings that may improve performance, given the network topology.

<u>Client Type</u>	<u>No. Of Clients</u>	<u>Recommendation</u>
DOS VLM	1	Leave window sizes at default.
	2	Set window sizes to 3.
	3+	Set window sizes to 1.
Windows NT	1	Leave window sizes at default.
	2+	Set window sizes to 1.

The number of clients refers to the total number of client workstations (DOS or Windows) that may be transferring files over the WAN link in the same direction simultaneously. If the link itself is not very reliable, model it by adding an extra client when looking up the recommended window sizes.

The window sizes for DOS are specified in the NET.CFG file as follows:

```
NetWare DOS Requester
    pburst read window size=number
    pburst write window size=number
```

The window size may be set to 1 in Windows NT by adding the value

```
MaxBurstSize: REG_DWORD: 0
```

to the registry key

```
HKEY_LOCAL_MACHINE/SYSTEM/CurrentControlSet/
Services/NWCWorkstation/Parameters
```

This also causes the IPX packet lengths to be reduced to a maximum of 562 bytes. Delete this key value to restore the defaults.

The use of the **QUEUE** property of the FPIPE or PTPIPE driver depends on the situation. However, it is recommended that it be disabled at first, and only enabled as a last resort. If TCP/IP is used over the link and it is slow and possibly buisy, such as during an FTP transfer, try enabling **QUEUE**.

If **QUEUE** is disabled, but the **Tx: Packets in adapter queue discarded** statistic gets too high relative to the **Tx: Packets sent successfully** counter (see "Using MONITOR" below), you may want to try enabling **QUEUE**. However, it is highly possible that user data transfer rates will worsen on the average with this property enabled. Also, if **QUEUE** is enabled, and the **Adapter Queue Depth** statistic gets too large you will have to turn **QUEUE** off (take it off the load command line). There is a nasty bug in NetWare where for an Ethernet MLID, there seems to be no upper bound on the send queue size. If the adapter queue is large enough that it uses up all of the available TCBS then the server may stall - i.e. it will not allow any more transmissions of packets anywhere, on ANY network interface.

Network Timeouts

There may be times when your DOS workstation running the NetWare DOS requester may time out, giving an error such as 'General failure reading device NETWORK'. If it happens often, and always recovers when **retry** is chosen, add the following to NET .CFG on the workstation:

```
Protocol IPX
ipx retry count 30
```

A retry count of 1 will wait for 30 system timer ticks, or about 1.67 seconds. The above setting will provide a timeout duration of 50 seconds.

Using MONITOR for Frame Relay

Run the **MONITOR** utility and examine the FPIPE LAN card statistics to diagnose Frame Relay link problems. The physical link should be up if all of the following statistics match the observed ones after things settle:

Tx/Oppflag: Busy counter	>0
Tx: Packets sent successfully	>0
Interrupts: Transmit	>0
Pending event notification code	0
DCD status	1
CTS status	1
Frames queued for reception	0
Transmission buffers available	1

If **Busy counter** exceeds 20,000 then the card may be hung. If it exceeds 40,000 then the driver decides that it IS hung. It is obviously not expected that the card will crash.

Note that **Total Packets Sent** includes both the packets sent successfully and the ones discarded.

If no **Transmit interrupts** occurred on startup, it is likely that the IRQ is not functioning properly, or may conflict with another device.

If all goes well, the **Pending event notification code** may start non-zero but should eventually settle to 0. It will not change value until a packet send attempt was made. All sent packets will be discarded until this value changes to 0. The reason for the discard should be indicated in the appropriate counter, such as **Tx discards: DLCI status change**.

The meaning of the Statistics shown on the monitor are as follows:

Tx/OppFlag: Busy counter

A count of the number of wasted busy-waiting loops before the card responds to a command.

Frame Relay

Tx discards: DCD or CTS low

DCD or CTS has dropped.

Tx discards: Hardware timeout

This is a very severe error, and indicates that the code running on the card most likely crashed. The link will be in an unknown state.

Tx discards: Channel changed to inoperative

The switch is no longer responding to polls.

Tx discards: Channel changed to operative

The switch is now responding to polls.

Tx discards: DLCI status change

Access Node reported a change in the status of one or more DLCIs. This statistic may be set to 1 when the link is first established to indicate the DLCI going active. The current state of the DLCI will be indicated as inactive if the corresponding statistic increases for packets transmitted after this.

Tx discards: DLCI configuration change

A change in the DLCI configuration list was made by the Access Node. It may be possible that the DLCI being used is now invalid - this will be reported on the next send attempt.

Tx discards: Communications not enabled

The channel should be in a data transfer state when the driver and card are initialized. If for some reason this error occurs, you will likely have to reboot and try again.

Tx discards: Channel currently inoperative

It may be that the link was temporarily brought down at the other end. Wait a while and try sending again. Check the physical connections.

Tx discards: DLCI inactive

The logical connection should be active, unless it was made inactive at the other end. Wait a while and try sending again. Verify that you are using the correct DLCI - check the 'Driver Properties'.

Tx discards: DLCI invalid

Verify that you are using the correct DLCI - check the 'Driver Properties'. Otherwise, the problem may be at the Access Node end.

Tx discards: Packet too big

This will not occur unless the driver is corrupt, or somehow the operating system does not properly detect the MTU of the driver. The MTU must be 1500 bytes.

Tx discards: CIR exceeded (n/a)

This will not occur unless the driver is corrupt.

Tx discards: Unknown error

An unexpected response to a command occurred. This should not happen.

Data Link Connection Identifier (DLCI)

This is the DLCI that is being used. If it was set to zero for auto-detection, this value will remain zero until one or more packets are sent or one is received for it to determine the DLCI.

Driver Statistics

Tx: Discards - tx ints not enabled

This value should always be zero.

Tx: Peak burst send count

Maximum number of packets that were sent back-to-back from the adapter queue at one time.

Tx: No buffers to send queued packet(s)

An attempt was made to send a packet when no transmit buffers were available on the card. The packet will either be queued or discarded.

Tx: Packets in adapter queue discarded

Number of packets discarded as a result of packets arriving too quickly to be sent on the line.

Tx: Packets sent successfully

Number of packets determined to have been successfully queued for transmission on the card. Except in rare cases, it can be assumed that these packets have been transmitted over the link. Of course, the network is free to discard packets at its discretion.

Tx: Buffers available after a send

This statistic is incremented for each packet sent successfully where after the packet is queued for transmission, it has been determined that there is still room in the queue for more. If there is no more room, a transmit interrupt should occur when a buffer becomes available.

Tx: B/A after send and tried another send

This is a count of events where a packet has been queued for transmission on the card, and there was room for another, and another was found in the adapter queue to be sent.

Tx: Calling DriverSend inside ISR (tx)

Number of times a packet was found on the adapter queue to be sent on the occurrence of a transmit interrupt. The transmit attempt was then made inside the interrupt service routine.

Interrupts: Transmit enabled

This value should always be 1.

Interrupts: Transmit

This value should always start at 1 when the driver is loaded. If not, the IRQ is not functioning properly.

Interrupts: Receive

This is incremented every time a packet is available on the card to be read.

Interrupts: State

This value should always be 0. It is currently unused.

Interrupts: Unknown

This may be non-zero when the driver is loaded, indicating a spurious interrupt when loading the frame relay code onto the card. It should not increase during normal operation.

Rx discards: Different DLCI

A packet was received on a different DLCI than was configured. These are always discarded.

Rx discards: RCB allocation failure

A memory allocation failure occurred when trying to pass a packet up the stack. The packet is discarded at this point, and another attempt at memory allocation will be made next time a packet is received. While this is happening, it is likely that all other network drivers will be encountering the same problem.

Sent RIP/SAP successfully on priority

If the **QUEUE** property has been disabled, RIPs and SAPs are held in a high priority queue for transmission if the first attempt failed. If the RIP/SAP was then successfully sent from the priority queue, this value is incremented.

Failed on RIP/SAP priority send

If the second attempt at sending the RIP/SAP contained in the high priority queue failed, it is finally discarded.

Last priority send error code

This is non-zero if the last high priority RIP/SAP send failed. The actual value is for internal use.

Router Statistics

Tx: ARP intercepts

This is the number of times the driver responded to an ARP request. The driver must spoof the ARP module into believing that there is a corresponding physical hardware address for the given destination IP address on FPIPE's network, or the packet will never be sent. This is part of the driver's Ethernet_II emulation.

Tx discards: Unknown protocol

This is the number of Ethernet_II frames given for transmission with an unknown or unsupported Ethertype field. This statistic will increase for packets that are not IP or IPX.

Rx discards: Unknown encapsulation

This is the number of frames received with an unknown RFC 1490 encapsulation. It may be that non-IP/IPX packets are being received, or the remote router is not encapsulating the packets properly according to RFC 1490. It is the encapsulation that determines the packet type.

Rx: IP packets processed at driver level

This is the number of IP packets received and passed up the stack as Ethernet_II frames.

Rx: IPX packets processed at driver level

This is the number of IPX packets received and passed up the stack as Ethernet_II frames.

MPR: Responses to Timer Request

This is the number of IPXWAN Timer Request responses sent. This applies only to IPX connections made to an IPX router, such as NetWare's MPR. Timer Requests are sent periodically by the remote router to negotiate router parameters.

MPR: Responses to Information Request

This is the number of IPXWAN Information Request responses sent. This applies only to IPX connections made to an IPX router, such as NetWare's MPR. Information Requests are sent periodically after successful negotiation made through the Timer Requests to determine FPIPE's router name. If this response was received by the remote router, the IPX layer should be up.

MPR: Timer Request NAKs received

This is the number of times the remote router rejected the parameters set in the IPXWAN Timer Response packet. See error message FPIPE-NW-206 for more information.

Detected IPX network number

This is the decimal representation of the detected IPX network number used for the link. This is updated for each RIP/SAP sent.

Board Status (updated only on startup and on transmit requests)

Pending event notification code

If non-zero, the next transmit attempt will fail and the cause will be reflected in the Frame Relay Statistics.

DCD status

This is set to 1 if DCD is high, and 0 if DCD is low.

CTS status

This is set to 1 if CTS is high, and 0 if CTS is low.

Frames queued for reception

The number of frames queued for reception on the card. They should be read from the card very quickly on receive interrupts, so this value should normally be zero. If it remains non-zero, it is likely that the IRQ is not functioning properly.

Transmission buffers available

If during the last transmission attempt, another transmit buffer was found to be available on the card, this value is set to 1.

Using MONITOR for PPP

Run the **MONITOR** utility and examine the `PTPIPE` LAN card statistics to diagnose link problems. The physical link should be up if all of the following statistics match the observed ones after things settle:

Tx/Oppflag: Busy counter	>0
Tx: Packets sent successfully	>0
Interrupts: Protocol open	>0
Interrupts: Transmit	>0
Interrupts: Receive	>0

If **Busy counter** exceeds 20,000 then the card may be hung. If it exceeds 40,000 then the driver decides that it IS hung. It is obviously not expected that the card will crash.

Note that **Total Packets Sent** should include both the packets sent successfully and the ones discarded.

If no interrupts are shown to occur in the above statistics, it may be that the chosen IRQ is conflicting with another device, or is otherwise not operational.

The meaning of the Statistics shown on the monitor are as follows:

Tx/OppFlag: Busy counter

A count of the number of wasted busy-waiting loops before the card responds to a command.

PPP

Init: Hardware timeout

This is a very severe error, and indicates that the code running on the card most likely crashed on initialization.

Tx: IPCP not in opened state

An IP packet has been discarded on transmission as no IP connection exists. The IP Control Protocol (IPCP) must be opened before IP packets may be transferred.

Tx: IPXCP not in opened state

An IPX packet has been discarded on transmission as no IPX connection exists. The IPX Control Protocol (IPXCP) must be opened before IPX packets may be transferred. Normally IPXCP negotiates default values to allow IPXWAN negotiation to proceed over the opened IPX link.

Driver Statistics

Tx: Peak burst send count

Maximum number of packets that were sent back-to-back from the adapter queue at one time.

Tx: No buffers to send queued packet(s)

An attempt was made to send a packet when no transmit buffers were available on the card. The packet will either be queued or discarded.

Tx: Packets in adapter queue discarded

Number of packets discarded as a result of packets arriving too quickly to be sent on the line.

Tx: Packets sent successfully

Number of packets determined to have been successfully queued for transmission on the card. Except in rare cases, it can be assumed that these packets have been transmitted over the link. Of course, the network is free to discard packets at its discretion.

Tx: Buffers available after a send

This statistic is incremented for each packet sent successfully where after the packet is queued for transmission, it has been determined that there is still room in the queue for more. If there is no more room, a transmit interrupt should occur when a buffer becomes available.

Tx: B/A after send and tried another send

This is a count of events where a packet has been queued for transmission on the card, and there was room for another, and another was found in the adapter queue to be sent.

Tx: Calling DriverSend inside ISR (tx)

Number of times a packet was found on the adapter queue to be sent on the occurrence of a transmit interrupt. The transmit attempt was then made inside the interrupt service routine.

Interrupts: Transmit

This value should always start at 1 when the driver is loaded. If not, the IRQ is not functioning properly.

Interrupts: Receive

This is incremented every time a packet is available on the card to be read.

Interrupts: Disconnect

This interrupt occurs when the Link Control Protocol (LCP) layer has gone down.

Interrupts: Protocol open

This interrupt occurs when either the IPCP or IPXCP layers have gone up, indicating packet transfer is possible. A console message will be displayed when this happens to indicate which layer is open.

Interrupts: Unknown

This may be non-zero when the driver is loaded, indicating a spurious interrupt when loading the PPP code onto the card. It should not increase during normal operation.

Rx discards: RCB allocation failure

A memory allocation failure occurred when trying to pass a packet up the stack. The packet is discarded at this point, and another attempt at memory allocation will be made next time a packet is received. While this is happening, it is likely that all other network drivers will be encountering the same problem.

Sent RIP/SAP successfully on priority

If the **QUEUE** property has been disabled, RIPs and SAPs are held in a high priority queue for transmission if the first attempt failed. If the RIP/SAP was then successfully sent from the priority queue, this value is incremented.

Failed on RIP/SAP priority send

If the second attempt at sending the RIP/SAP contained in the high priority queue failed, it is finally discarded.

Last priority send error code

This is non-zero if the last high priority RIP/SAP send failed. The actual value is for internal use.

Router Statistics

Tx: ARP intercepts

This is the number of times the driver responded to an ARP request. The driver must spoof the ARP module into believing that there is a corresponding physical hardware address for the given destination IP address on PTPPIPE's network, or the packet will never be sent. This is part of the driver's Ethernet_II emulation.

Tx discards: Unknown protocol

This is the number of Ethernet_II frames given for transmission with an unknown or unsupported Ethertype field. This statistic will increase for packets that are not IP or IPX.

Rx discards: Unknown encapsulation

This value will always be zero as it is currently not used. Any frames received by the PPP code on the card of an unknown, unsupported, or unconfigured protocol are discarded.

Rx: IP packets processed at driver level

This is the number of IP packets received and passed up the stack as Ethernet_II frames.

Rx: IPX packets processed at driver level

This is the number of IPX packets received and passed up the stack as Ethernet_II frames.

MPR: Responses to Timer Request

This is the number of IPXWAN Timer Request responses sent. This applies only to IPX connections made to an IPX router, such as NetWare's MPR. Timer Requests are sent periodically by the remote router to negotiate router parameters.

MPR: Responses to Information Request

This is the number of IPXWAN Information Request responses sent. This applies only to IPX connections made to an IPX router, such as NetWare's MPR. Information Requests are sent periodically after successful negotiation made through the Timer Requests to determine PTPIPE's router name. If this response was received by the remote router, the IPX layer should be up.

MPR: Timer Request NAKs received

This is the number of times the remote router rejected the parameters set in the IPXWAN Timer Response packet. See error message PTPIPE-NW-206 for more information.

Detected IPX network number

This is the decimal representation of the detected IPX network number used for the link. This is updated for each RIP/SAP sent.

Board Status

No statistics currently defined.

NLSP vs. RIP

FPIPE is not designed to handle NLSP traffic in **Routing Mode**. In bridging mode, which is transparent, it will handle any traffic. **PTPIPE** is not designed to handle NLSP traffic at all. If

FPIPE/PTPIPE is connected to MPR, it will have negotiated RIP and so no NLSP should be observed over the link, unless the FPIPE host node sends NLSP packets. If it does send NLSP and routing conflicts occur (which would show up on the server consoles) then NLSP must be disabled at the host end. This is configurable at the NetWare level, such as by using **INETCFG** to select the appropriate expert IPX bind options for the appropriate board.

By definition, NLSP along with any other traffic may be carried across an **FPIPE** link in **Bridging Mode**, as it sends and receives transparently. The only exception is that RIP or SAP packets are intentionally not discarded if the **QUEUE** property is disabled. It is important that RIPs are not discarded since after a few consecutive discards, forwarding entries in the server routing tables will be purged. This was one reason for the development of NLSP. Note that RIP and SAP packets are intentionally not discarded if the **QUEUE** property is disabled.

Uninstalling WANPIPE™

FPIPE/PTPIPE may be uninstalled by running **INSTALL** and hitting **DELe**te on the selected FPIPE/PTPIPE product.

The following files must be removed manually:

For Frame Relay:

```
sys:system\FPIPE.lan  
sys:system\FPIPE508.lan  
sys:system\FPIPE.cfg
```

For PPP

```
sys:system\ptpipe.lan  
sys:system\ptpipe8.lan  
sys:system\ptpipe.cfg
```

6. WANPIPE™ specifications

Frame Relay

Encapsulation: RFC 1490 or RFC 1294 (choose IETF if connecting to Cisco router).
Compression: None
Packet fragmentation: Not supported. MTU of router must be 1500 bytes.
Inverse-ARP (RFC 1293): Support dependent on operating system

PPP

PPP Implementation: RFC 1661
Network Control Protocols (NCPs): IPCP (RFC 1332)
IPXCP (RFC 1552)
Authentication Protocols: PAP
CHAP MD5 (RFC 1321)
Both RFC 1334
Inbound or Outbound
Supported LCP Configuration Options: Maximum-Receive-Unit
Authentication-Protocol
Magic-Number
Supported IPCP Configuration Options: IP-Address
Supported IPXCP Configuration Options: IPX-Network-Number
IPX-Node-Number
IPX-Routing-Protocol
IPX-Router-Name
IPX-Configuration-Complete
Minimum/Maximum MRU=MTU: 1500 / 8178 bytes

Driver Limitations:

The specific MRU/MTU for all drivers is 1500 bytes. IPXCP negotiates to default options in all IPX-capable drivers on the expectation that IPXWAN will be used later. The IP-Address option is ignored on the NDIS and HSM drivers. Authentication is not supported by all drivers except the Packet Driver.

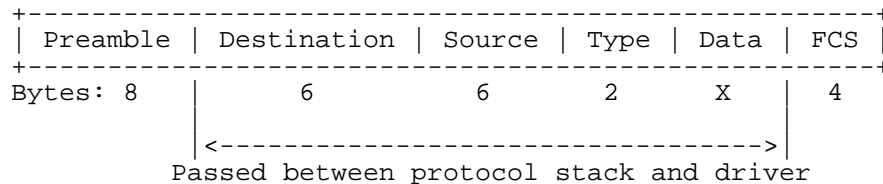
7. DEC, Intel, Xerox (DIX) Ethernet Frame Format and Frame Relay and PPP Encapsulations

Frame Relay routing format

The Maximum Transmission Unit (MTU) for FPIPE is 1500 bytes, as is the MTU for an Ethernet device. It is therefore necessary to ensure that the remote end does not attempt to send packets of a larger length; if it does, they will be silently discarded.

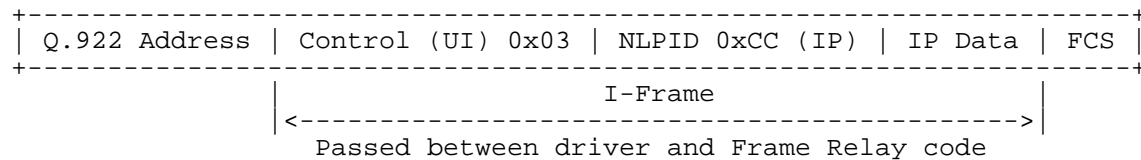
This driver emulates DIX Ethernet (ETHERNET_II) as follows:

DIX Ethernet (ETHERNET_II) Frame:



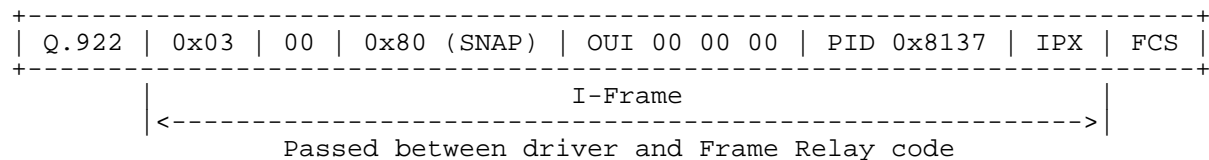
When the driver is passed a frame to transmit, the destination, source, and Ethertype fields are discarded and the raw Data field containing the packet is encapsulated for Frame Relay according to RFC 1490:

IP:



Note: For full Ethernet emulation, each driver responds to locally issued ARP requests with an arbitrary node address, so that the Ethernet transport layer will be able to transmit the IP packet. The same 6-byte physical node address is used as the target for all IP addresses, and will appear as such when displaying the ARP table (if permitted by the particular protocol stack in use).

IPX:

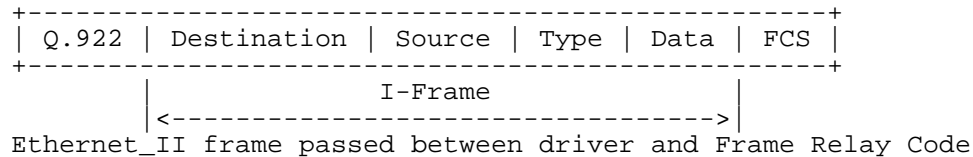


The Frame Relay code on the S508 card adds the Q.922 address field and the FCS field which contains the CRC bytes.

The maximum size of the data in the Frame Relay frame (the I-frame) is 1502 bytes for IP, and 1508 bytes for IPX which includes the encapsulation.

Bridging Mode for FPIPE LAN HSM

IP/IPX/Other:



Example Ethertypes are:

```

0800 = IP
0806 = ARP
8137 = IPX

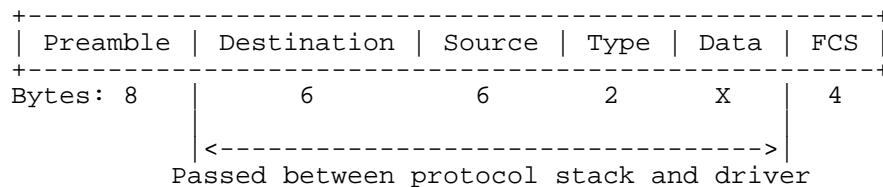
```

PPP Frame Formats

The Maximum Transmission Unit (MTU) for PTPPIPE is 1500 bytes, as is the MTU for an Ethernet device. It is therefore necessary to ensure that the remote end does not attempt to send packets of a larger length; if it does, they will be silently discarded.

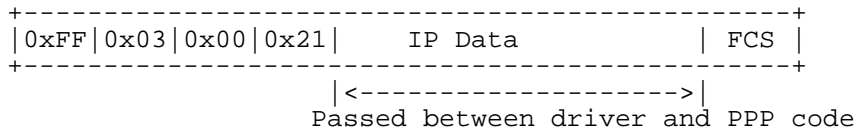
This driver emulates DIX Ethernet (ETHERNET_II) as follows:

DIX Ethernet (ETHERNET_II) Frame:



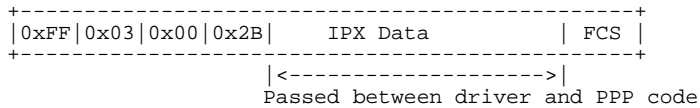
When the driver is passed a frame to transmit, the destination, source, and Ethertype fields are discarded and the raw Data field containing the packet is encapsulated for PPP according to RFC 1661:

IP:



Note: For full Ethernet emulation, each driver responds to locally issued ARP requests with an arbitrary node address, so that the Ethernet transport layer will be able to transmit the IP packet. The same 6-byte physical node address is used as the target for all IP addresses, and will appear as such when displaying the ARP table (if permitted by the particular protocol stack in use).

IPX:



The PPP code on the S508 card adds the protocol field and the FCS field which contains the CRC bytes.

8. Low level Debugging of WAN Connections on S508 ISA card: SNOOPER.EXE

Introduction

SNOOPER . EXE is an MS-DOS based diagnostic tool for assisting in the configuration of the Sangoma ISA adapters and for debugging communication lines. Please note that PCI cards are invisible to MS-DOS, and therefore cannot be debugged with SNOOPER.

SNOOPER provides detailed configurations of Frame Relay and SDLC links, and useful information for PPP and X.25 connections as well. SNOOPER incorporates both line statistics and a datascopes. **SNOOPER** only runs on ISA cards, PCI cards like the S514 being invisible to DOS.

Installing communications equipment can be frustrating and time consuming particularly when the connection is new. There are many parameters which must be set exactly correctly before you can even get to “first base” where host communications are established. The DOS program SNOOPER . EXE has been included to help wring out problems in host set ups, communication lines, modems or DSUs cables and Sangoma communication cards.

SNOOPER detects the type of Sangoma adapter installed, explores the server configuration, searching out available I/O ports, shared memory and interrupts, and then snoops the communications line, finds the correct line encoding and provides displays of useful line information.

Running SNOOPER.EXE

To run SNOOPER, boot up under MS-DOS *without any CONFIG.SYS file*. SNOOPER will also run in a DOS window of Windows or Windows '95. It will not run in a Windows NT DOS window.

To run SNOOPER enter:

```
SNOOPER
```

Interpreting SNOOPER Data

SNOOPER first detects the adapter type and finds its base port address. It then searches the PC's high memory area looking for a free 8k window in which to load itself.

```
MS-DOS Prompt - SNOOPER
10 x 20
SNOOPER Ver 2.00 (c)SANGOMA TECH, 1996 || CURRENT STATUS: ADPTATST (ver 1.00)

ADAPTER TYPE: $502E
ADAPTER I/O PORT BASE ADDRESS: 0x360

CHECKING FOR AVAILABLE 8K MEMORY BLOCKS (iteration #1)
0xA0-in use      0xC0-in use      0xD0-available   0xE0-in use
0xA2-in use      0xC2-in use      0xD2-in use      0xE2-in use
0xA4-in use      0xC4-in use      0xD4-in use      0xE4-in use
0xA6-in use      0xC6-in use      0xD6-in use      0xE6-in use
0xA8-in use      0xC8-available   0xD8-in use      0xE8-in use
0xAA-in use      0xCA-in use      0xDA-in use      0xEA-in use
0xAC-in use      0xCC-in use      0xDC-in use      0xEC-in use
0xAE-in use      0xCE-in use      0xDE-in use      0xEE-in use
```

A screen similar to the one above will display.

Having snooped the server for free memory space and I/O ports, SNOOPER will load the board and test the IRQs. If there is a free set of resources available, the program will load the card and proceed to snoop the communication line. It will load in the first available shared memory block, using the I/O port and IRQ it has found.

The second screen then comes up. At the top, the menu selections allow you to :

Configure:

Sets up the electrical connection (V.35 or RS232) and sets board clocking options. You can set to have the board provide the clocking (Internal) or the clocking can be External. If you are connecting to a DSU, chose External. If you are connected with a back-to-back cable, one end of the link will be set as External and the other as Internal. You can set the line speed for an Internal clock.

Protocol:

Either set for Monitor Line only (use for PPP or X.25) or Frame Relay -Emulate CPE. For **WANPIPE™** Frame Relay, normally you would chose Frame Relay-Emulate CPE.

In the Monitor only mode, **SNOOPER** will not transmit, but will display data that is received. You should be able to see the PPP startup sequence arrive from the other end of the link.

Even on a PPP connection, you can test whether or not you have a transmit clock signal by setting the Protocol to Frame Relay- Emulate CPE and seeing that frames are in fact transmitted. Of course the frames transmitted in this case will make no sense to the other side.

Trace:

Sets tracing and disk tracing options.

Reset and Quit

In Frame Relay protocol mode, **SNOOPER** is looking for a Frame Relay network connection. It will send out FSE supervisory messages to the switch to find out what Logical Channels (DLCIs) are listed as configured, and which are actively connected at the other end.

If this is a PPP or X.25 connection, you can ignore the DLCI data in the lower left box. FSE messages will be continually transmitted, but no FS messages will be received.

If the connection to the DSU is working correctly, DCD and, perhaps, CTS should be up. The FSE messages should go out every few seconds. If the DSU is not providing a transmit clock, an error message will pop up.

On a working Frame Relay connection:

1. FS Messages Received should be the same as FSE Messages Sent.
2. The Channel Status should become Active within 90 seconds.
3. At least One DLCI should be reported by the Switch, and it should be Active.

The Active DLCIs represent the Logical Channel(s) that you can use to configure the **WANPIPE™** or MPR connections.

The errors reported on the right hand side should remain zero.

Frames transmitted by the CPE emulation software are shown in inverse video. Only the first 24 data characters are displayed. In the case of a PPP or X.25 connection, the communication link

can be verified by watching for received frames in normal video. If you see no frames shown in standard video format, then there is a problem with the network or PPP connection.

The program terminates when you Quit.

When configuring Interrupt, I/O Base and Memory Base for the connection, use the data from this screen and you will not have any trouble.

SNOOPER may also be run with defined card settings. The full list of command line arguments are as follows:

```
-p[I/O PORT BASE ADDRESS] (default: unknown)
-m[SHARED MEMORY ADDRESS] (default: unknown)
  -t[ADAPTER TYPE] (default: unknown, 0=502A, 1=502E,
                    2=503, 3=508)
-i[IRQ TO BE TESTED](default: unknown)
```

For example:

```
SNOOPER -mCA -t3
```

Check to see if a S508 adapter is in the PC and if memory area 0xCA00 is available for use.

The first character of the shared memory window segment and offset specifies the 64K PC memory segment to be used. Valid values are A, C, D or E. The second character defines the 8K window within the selected 64K memory segment. Valid values are as follows:

<u>Value</u>	<u>PC memory window</u>
0	0000H to 1FFFH
2	2000H to 3FFFH
4	4000H to 5FFFH
6	6000H to 7FFFH
8	8000H to 9FFFH
A	A000H to BFFFH
C	C000H to DFFFH
E	E000H to FFFFH

9. Appendix: Frame Relay Basics

Frame Relay is basically a simplified form of Packet Switching in which synchronous frames of data are routed to different destinations depending on header information.

Framing

Frame Relay uses synchronous HDLC frames up to 4kbytes in length. Each frame starts and ends with a Flag character (7E). The first 2 bytes of each frame following the flag contain the information required for multiplexing across the link. The last 2 bytes of the frame are always generated by a Cyclic Redundancy Check (CRC) of the rest of the bytes between the flags. This allows easy checking of the data integrity. The rest of the frame contains the user data.

Virtual Circuits

Packets are routed through one or more Virtual Circuits known as Data Link Connection Identifiers (DLCIs). Each DLCI has a permanently configured switching path to a certain destination. Thus, by having a system with several DLCIs configured, you can communicate simultaneously with several different sites. Currently, only permanent virtual circuit connections are supported.

Flow Control and Information Rates

There is no flow control on Frame Relay. The network simply discards frames it cannot deliver. There is no error correction either. Bad frames are simply delivered as best they can be. It is up to the end users to introduce correction mechanisms for data loss due to bad or dropped frames.

When you subscribe, you will set the line speed (e.g. 56 kbps or T1) and also, typically, you will be asked to specify a Committed Information Rate (CIR) for each DLCI. This value specifies the maximum average data rate that the network undertakes to deliver under "normal conditions". If you send faster than that on a given DLCI, the network will flag some frames with a Discard Eligibility (DE) bit. The network will do its best to deliver all packets but will discard any DE packets first if there is congestion.

Frame Relay provides indications that the network is becoming congested by means of the Forward Explicit Congestion Notification (FECN) and Backward Explicit Congestion Notification (BECN). These are used to tell the application to slow down, hopefully before packets start to be discarded.

Frame Relay is cost effective, partly due to the fact that the network buffering requirements are carefully optimized. Many inexpensive Frame Relay services are based on a CIR of zero. This means that every frame is a DE frame, and the network will throw them away when it needs to.

Status Polling

The Frame Relay Customer Premises Equipment (CPE) polls the switch at set intervals to find out the status of the network and DLCI connections. A Link Integrity Verification (LIV) packet exchange takes place about every 10 seconds, which basically verifies that the connection is still good. It also provides information to the network that the CPE is active, and this status is reported at the other end.

There are three standards for Status Polling currently in use: LMI, the older standard, ANSI ANNEX D, and Q.933. **WANPIPE™** supports all three.

About every minute, a Full Status (FS) exchange occurs, which passes information on which DLCIs are configured and active. Until the first FS exchange has occurred, the CPE does not know which DLCIs are active, and so no data transfer can take place.

NOTE: You may have to wait for up to 2 minutes after first boot up before you try any kind of data communication. Such communication may fail if the first FS exchange has not yet occurred.

Frame Relay and FPIPE

The default settings for the Frame Relay layer of FPIPE are as follows:

We assume that you are configured as a CPE. It is also possible to configure the S508 card as an Access Node for back to back connections.

For the simpler implementations such as FPIPE under NetWare, we only support one DLCI. This can be configured, or the system can discover the first active DLCI and use that. Under Windows 95, NT and the Unix platforms, FPIPE supports multiple DLCIs.

The 508 and S514 cards and code are capable of sophisticated CIR monitoring and control. This is turned off in our default configuration. Data will be transmitted whenever it is provided by the protocol stack. All control of data flow is left up to the protocol handlers.