Help Volume

Toolsets: DataComm Analysis (Agilent Technologies B4640B)

Using the DataComm Analysis Toolset

The DataComm Analysis Toolset adds protocol-based triggering and display capabilities to the logic analyzer. This toolset helps you look at data traveling across parallel buses inside network switching systems.

The DataComm Analysis Toolset consists of:

- The "Find Packet" trigger function which lets you set up logic analyzer triggers in terms of network protocols and packet data.
- The network protocol decoder inverse assembler which lets you display captured data in terms of the network protocol being used.
- The ATM Tool which displays data and detects errors in data captured on UTOPIA buses.
- The POS-PHY Tool which displays data and detects errors in data captured in POS-PHY level 1-3 (packet over SONET) buses.

You can choose from a number of pre-defined protocols (including most of the Internet Protocol stack) when setting up triggers and displaying captured data. And, you can define your own custom or proprietary protocols by modifying protocol definitions or creating new ones.

NOTE: Because the DataComm Analysis Toolset requires VisiTrigger capabilities, it can only be used with the Agilent Technologies 16715/16/17/18/19/50/51/52A logic analyzer modules.

“Installation and Licensing” on page 8

- “To check if the DataComm Analysis Toolset is installed” on page 8
- “To install the DataComm Analysis Toolset” on page 9
- “To license the DataComm Analysis Toolset” on page 10

“Probing the Device Under Test” on page 12

- “Step 1. Connect the logic analyzer probes” on page 12
- “Step 2. Select the state sampling mode” on page 13
- “Step 3. Label the logic analyzer channels” on page 14

See Also

Main System Help (see the *Agilent Technologies 16700A/B-Series Logic Analysis System* help volume)

Glossary (see page 69)

Contents

Using the DataComm Analysis Toolset

1 Installation Guide

2 Task Guide

1

Installation Guide

Installation and Licensing

The DataComm Analysis Toolset software may already be installed with logic analysis system software (if they are ordered at the same time).

If the DataComm Analysis Toolset is not already installed, you need to install it.

Once the DataComm Analysis Toolset is installed, you must obtain a license to use it beyond the demo period.

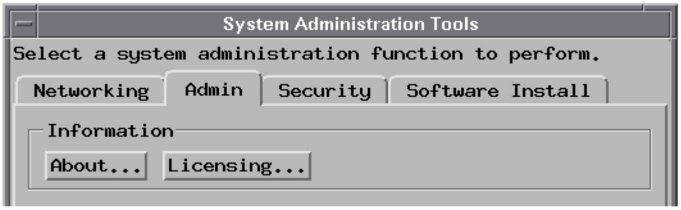
- "To check if the DataComm Analysis Toolset is installed" on page 8
- "To install the DataComm Analysis Toolset" on page 9
- "To license the DataComm Analysis Toolset" on page 10

See Also

Licensing Policy for the Logic Analysis System and Tool Sets (see the *Agilent Technologies 16700A/B-Series Logic Analysis System* help volume)

To check if the DataComm Analysis Toolset is installed

1. In the main logic analysis system window, select the System Admin button .
2. In the Admin tab of the System Administration Tools dialog, select the Licensing... button.



System Administration Tools

Select a system administration function to perform.

3. In the Tool Sets tab of the Licensing Dialog, look for the "B4640B - DataComm Analysis" product.

   - If it is not listed, go to “To install the DataComm Analysis Toolset” on page 9.

   - If it is listed, go to “To license the DataComm Analysis Toolset” on page 10.

To install the DataComm Analysis Toolset

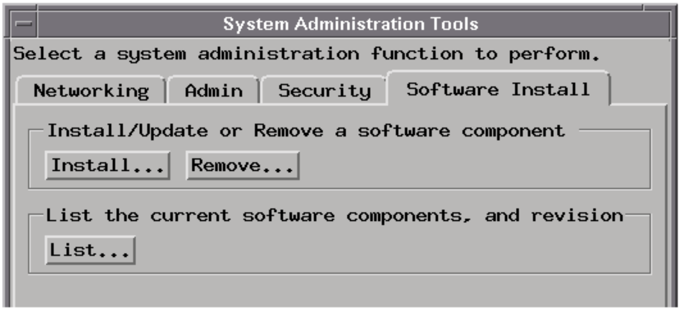
If the CD-ROM drive is not connected to the logic analysis system, see the instructions printed on the CD-ROM package.

1. Turn on the CD-ROM drive first; then, turn on the logic analysis system.

2. Insert the CD-ROM in the drive.

3. In the main logic analysis system window, select the System Admin button .

4. In the Software Install tab of the System Administration Tools dialog, select Install....



System Administration Tools

Select a system administration function to perform.

Networking
Admin
Security
Software Install
Install/Update or Remove a software component
Install...
Remove...
List the current software components, and revision
List...

5. Change the media type to "CD-ROM" if necessary, and select Apply.

6. Select the "AUXILIARY-SW" selection to open it.

7. Select the DataComm Analysis Toolset; then, select the Install button.

The dialog will display "Progress: completed successfully" when the installation is complete.

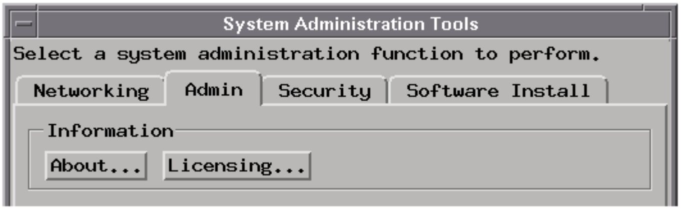
8. Select the Close button to close the Software Install window.
9. Select the Close button to close the System Administration Tools window.
10. Go to “To license the DataComm Analysis Toolset” on page 10.

See Also

See the instructions printed on the CD-ROM package for a summary of the installation instructions.

To license the DataComm Analysis Toolset

1. To obtain a password, contact the password center listed on the Entitlement Certificate you received after purchasing the DataComm Analysis Toolset.
2. In the main logic analysis system window, select the System Admin button 🔧.
3. In the Admin tab of the System Administration Tools dialog, select the Licensing... button.



System Administration Tools

Select a system administration function to perform.

4. In the Tool Sets tab of the Licensing Dialog, enter the password into the field for the "B4640B - DataComm Analysis" product.
5. Select OK to close the Licensing Dialog.
6. Select Close to close the System Administration Tools window.
7. Restart the session.

You are now ready to use the DataComm Analysis Toolset.

2

Task Guide

Probing the Device Under Test

To probe the device under test (also known as a target system):

- "Step 1. Connect the logic analyzer probes" on page 12
- "Step 2. Select the state sampling mode" on page 13
- "Step 3. Label the logic analyzer channels" on page 14

Step 1. Connect the logic analyzer probes

To use the DataComm Analysis Toolset, you need to connect logic analyzer probe channels to:

- A communication data bus (for example, Utopia).
- Control signals that identify the start of packet, the packet data (data valid), and the end of packet.

  The POS-PHY Tool requires that all of these control signals be probed.

You also need to connect a logic analyzer CLK input channel to:

- A clock signal that identifies when the data bus and control signals are valid and should be sampled by the logic analyzer.

You can also connect other signals to the remaining logic analyzer CLK input channels and use them to "qualify" the main sampling clock signal. This can be useful, for example, when packet transfers are interrupted.

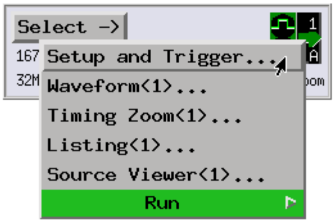
When setting up the logic analyzer's state sampling mode, you can specify that one CLK input signal be used to enable or disable the sampling signal on another CLK input. Disabling the sampling clock during interrupts will allow the logic analyzer to find a packet event that has been interrupted.

Next Step (see page 13)

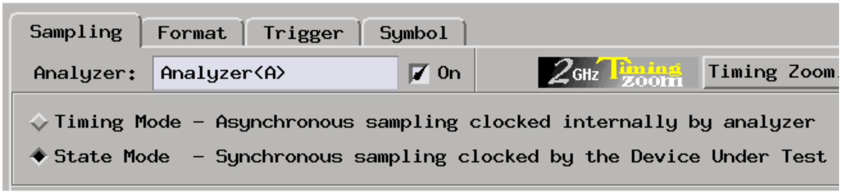
Step 2. Select the state sampling mode

When using the DataComm Analysis Toolset, the logic analyzer must be set up in the state (synchronous) sampling mode.

1. Choose the Setup... command.



2. In the Sampling tab, select the state mode.



Analyzer: Analyzer<A> ☑ On

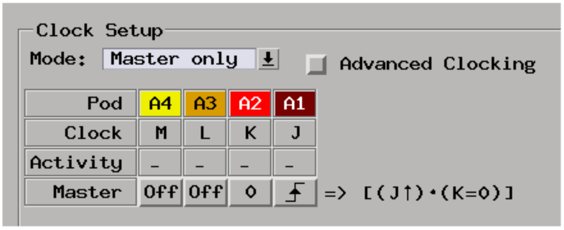
○ Timing Mode - Asynchronous sampling clocked internally by analyzer

● State Mode - Synchronous sampling clocked by the Device Under Test

3. Specify the state analyzer’s sampling clock input.

   If you’ve connected an additional signal from the device under test to another of the logic analyzer’s CLK inputs for the purpose of "qualifying" the main sampling clock signal, include that signal in the clock setup.

   For example, suppose the main communication data bus clock signal has been connected to the pod 1 CLK input (J) and a signal that is low when not in a packet transfer interrupt has been connected the pod 2 CLK input (K). In this case, you would use a clock setup like:



Clock Setup

Mode: Master only ⬇ □ Advanced Clocking

Pod	A4	A3	A2	A1
Clock	M	L	K	J
Activity	–	–	–	–
Master	Off	Off	0	↑

=> [(J↑)•(K=0)]

When there is a packet transfer interrupt, the signal connected to the pod 2 CLK input (K) is high, and there is no sampling clock.

Because no data values are sampled by the logic analyzer during the interrupt, the logic analyzer can find the rest of a packet after the interrupt.

Next Step (see page 14)

Step 3. Label the logic analyzer channels

The network protocol decoder inverse assembler requires you to set one "DATA" label for the communication data bus channels. The ATM Tool and the POS-PHY Tool will work with any data bus label name.

Also, you must set up one, 1-bit label for the start of packet signal. If there are signals for valid packet data and end of packet signals (the POS-PHY Tool requires them), set up 1-bit labels for those signals as well.

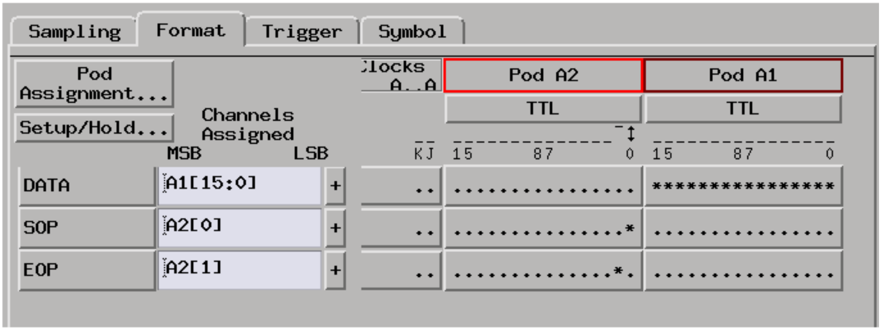
To label the logic analyzer channels:

1. In the Format tab of the logic analyzer’s setup window, insert or rename labels so that there is a label for the data bus (use the label name "DATA" if you’ll be using the network protocol decoder inverse assembler) and 1-bit labels for the start of packet, packet data, and end of packet signals.

2. For the data bus label, specify which logic analyzer pod channels are probing the data bus.

3. For the 1-bit labels, specify which logic analyzer pod channels are probing

the start of packet, packet data, and end of packet signals.



Channels Assigned MSB LSB		Clocks A..A KJ	Pod A2 15 87 0	Pod A1 15 87 0
DATA	A1[15:0]	..	....	****************
SOP	A2[0]	..	...............*	....
EOP	A2[1]	..	..............*.	....

In the Bus Editor dialog, you will specify which labels contain the start of packet, the packet data, and the end of packet signals.

NOTE: The network protocol decoder inverse assembler can only decode one bus at a time. If your logic analyzer probes multiple communications buses in the device under test, create labels for each bus. Then, for whichever bus you want decoded in the Listing window, rename its bus label to DATA.

See Also

“Capturing the Data” on page 16

Capturing the Data

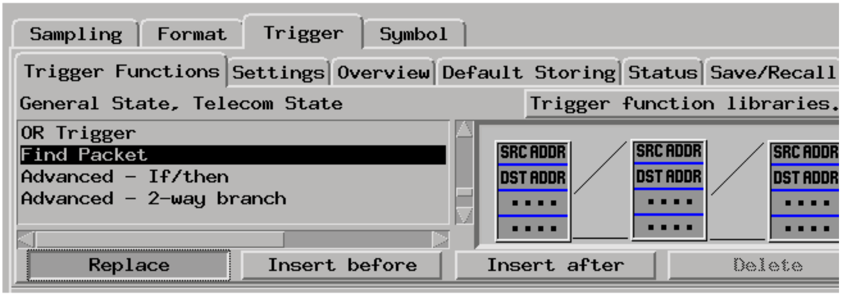
This section shows you how to set up packet triggers and run logic analyzer measurements.

- “To set up packet triggers” on page 16
- “To define and edit packet events” on page 21 (“Event Editor Dialog” on page 23)
- “To run the measurement” on page 24

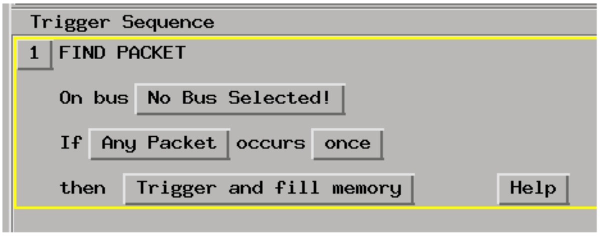
To set up packet triggers

1. In the Trigger tab of the logic analyzer's setup window, select the Trigger Functions tab.

2. Select the "Find Packet" trigger function and either replace the current trigger sequence level or insert a new level.

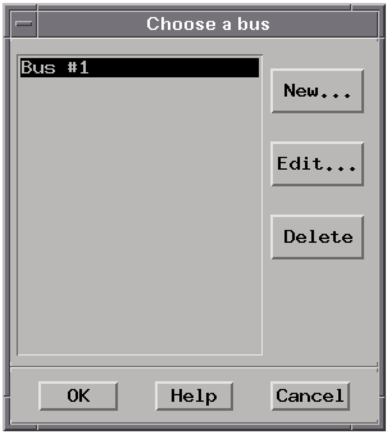


3. In the "Find Packet" trigger sequence level, select the bus button.



Trigger Sequence
1 FIND PACKET
On bus No Bus Selected!
If Any Packet occurs once
then Trigger and fill memory

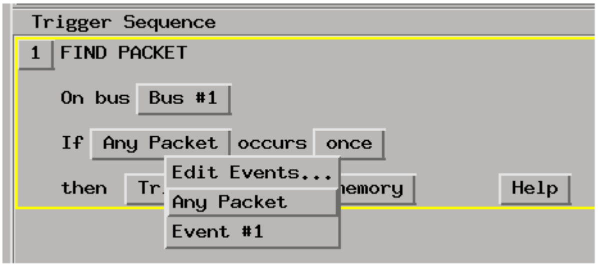
4. In the Choose a Bus dialog, select the bus definition you want to use and select the OK button.



Choose a bus
Bus #1

If you need to define or edit a bus, see “To define and edit buses” on page 18.

5. Specify the packet event to find.



Trigger Sequence
1 FIND PACKET
On bus Bus #1
If Any Packet occurs once
then Tr... memory

NOTE:

All packet fields start as "don’t care" values. The more fields you enter specific values into, the more of the logic analyzer’s internal triggering resources are used, and eventually you can run out of triggering resources.

So, enter the minimum number of fields necessary to get your trigger, and save trigger resources for other events.

See Also

“Modifying Protocol Definitions” on page 52

“To define and edit buses” on page 18

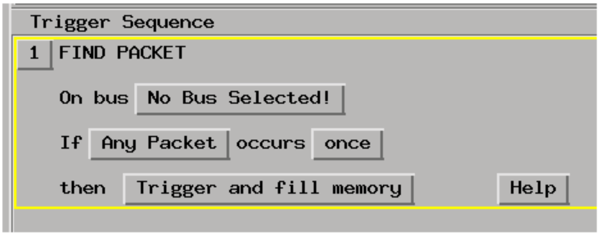
“To define and edit packet events” on page 21

“To run the measurement” on page 24

“Displaying Data with the Network Protocol Decoder IA” on page 25

To define and edit buses

1. In the "Find Packet" trigger function (or in the Invasm Preferences dialog), select the bus button.



Trigger Sequence

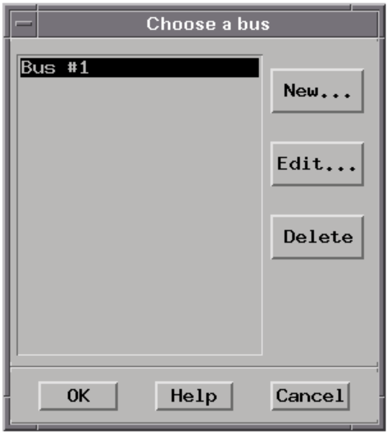
1 FIND PACKET

On bus No Bus Selected!

If Any Packet occurs once

then Trigger and fill memory

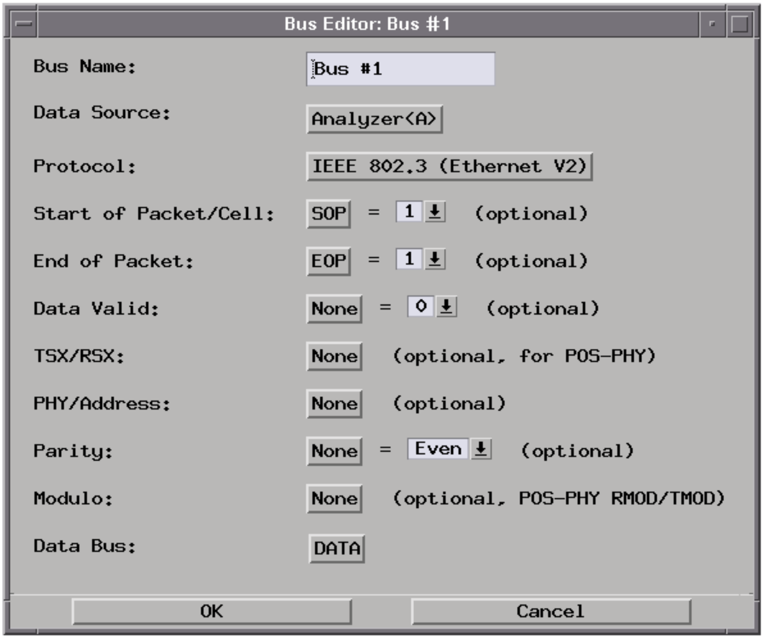
2. In the Choose a Bus dialog, either select New... to define a new bus or select the bus you want to edit from the list and select the Edit... button.



3. In the Bus Editor dialog, specify the characteristics of the bus.
4. When you are done editing the bus, select the OK button in the Bus Editor dialog and select the OK button in the Choose a Bus dialog.

See Also “Bus Editor Dialog” on page 20

Bus Editor Dialog



Bus Editor: Bus #1

Bus Name: Bus #1

Data Source: Analyzer<A>

Protocol: IEEE 802.3 (Ethernet V2)

Start of Packet/Cell: SOP = 1 (optional)

End of Packet: EOP = 1 (optional)

Data Valid: None = 0 (optional)

TSX/RSX: None (optional, for POS-PHY)

PHY/Address: None (optional)

Parity: None = Even (optional)

Modulo: None (optional, POS-PHY RMOD/TMOD)

Data Bus: DATA

Bus Name Specifies the bus definition name.

Data Source Specifies which logic analyzer module captures data on the bus.

Protocol Selects the protocol being used on the bus.

Start of Packet/ Cell Selects the 1-bit label for the signal and value that indicates a start of packet.

End of Packet Selects the 1-bit label for the signal and value that indicates an end of packet.

Data Valid Selects the 1-bit label for the signal and value that indicates valid data is on the bus.

TSX/RSX Selects the 1-bit label for the POS-PHY Transmit Start of Transfer (TSX) or Receive Start of Transfer (RSX) signal.

PHY/Address Some Utopia busses support Multi-PHY operation (multiple physical interfaces connected to one bus).

If you select a PHY label, it is added to the "Find Packet" trigger function so that you can trigger when a packet occurs with a selected PHY address.

Parity Selects the 1-bit label for the parity signal and the type of parity being used (Even or Odd).

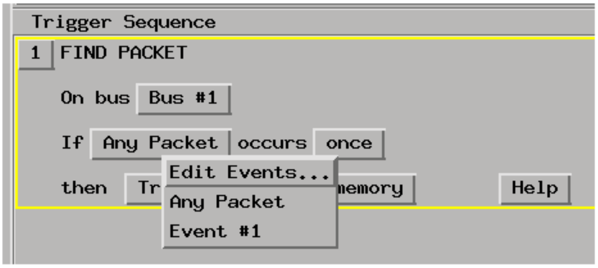
Modulo Selects the 2-bit label for the POS-PHY Receive Word Modulo (RMOD) or Transmit Word Modulo (TMOD) signals.

Data Bus You can select multiple data labels for a data bus in situations where the data bus is more than 32 bits (labels are limited to 32 bits).

See Also "To define and edit buses" on page 18

To define and edit packet events

1. In the "Find Packet" trigger function, select the packet button, and choose Edit Events....



Trigger Sequence

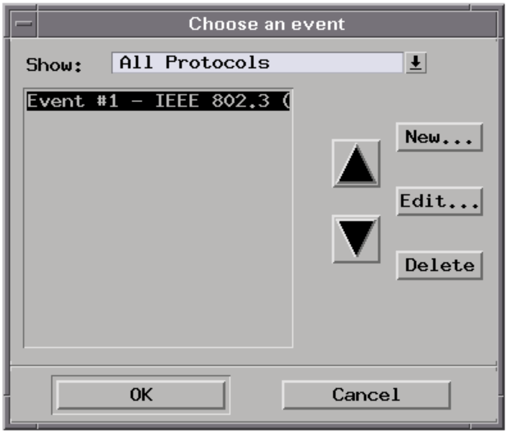
1 FIND PACKET

On bus Bus #1

If Any Packet occurs once

then Tr memory

2. In the "Choose an event" dialog, either select New... to define a new event, or select an event and select the Edit... button to edit it.



Choose an event

Show: All Protocols

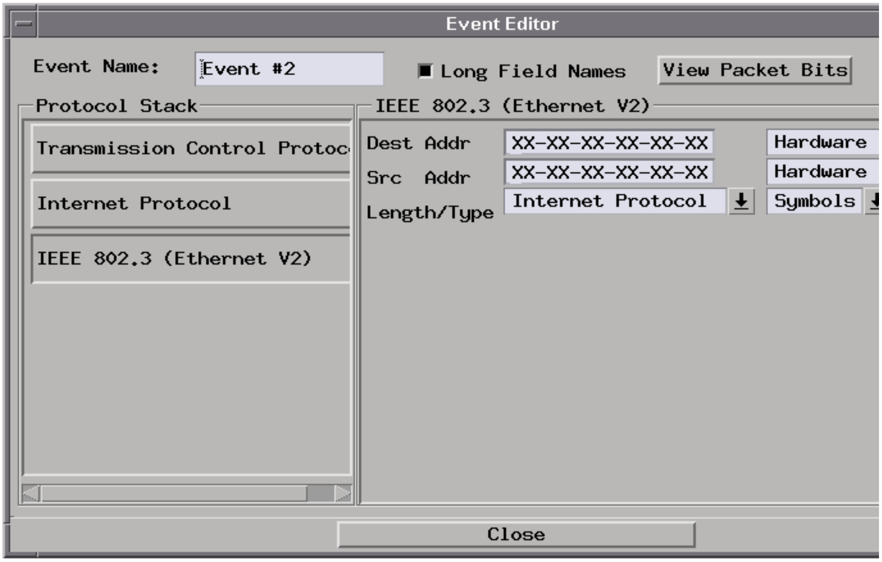
Event #1 - IEEE 802.3 (

To delete an event definition, select the Delete button.

3. When you are done defining or editing an event, select the Close button in the Event Editor dialog and select the OK button in the "Choose an event" dialog.

See Also “Event Editor Dialog” on page 23

Event Editor Dialog



Event Name: Event #2

■ Long Field Names

Protocol Stack

- Transmission Control Protoc
- Internet Protocol
- IEEE 802.3 (Ethernet V2)

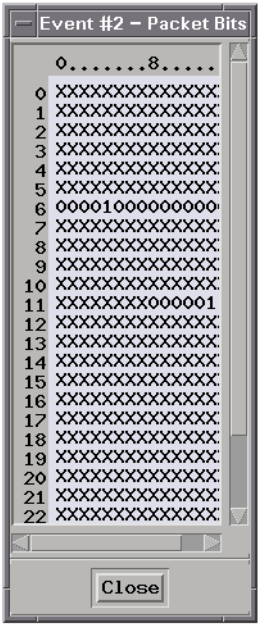
IEEE 802.3 (Ethernet V2)

Dest Addr XX-XX-XX-XX-XX-XX Hardware
Src Addr XX-XX-XX-XX-XX-XX Hardware
Length/Type Internet Protocol Symbols

Event Name Specifies the event definition name.

Long Field Names Specifies whether the full names of fields or the short 2- or 3-letter mnemonics for fields are displayed in the protocol column.

View Packet Bits Specifies whether the packet bits window is displayed. The Packet Bits window shows the packet event’s 1s and 0s.



```
    0.......8.....
 0  XXXXXXXXXXXXXX
 1  XXXXXXXXXXXXXX
 2  XXXXXXXXXXXXXX
 3  XXXXXXXXXXXXXX
 4  XXXXXXXXXXXXXX
 5  XXXXXXXXXXXXXX
 6  00001000000000
 7  XXXXXXXXXXXXXX
 8  XXXXXXXXXXXXXX
 9  XXXXXXXXXXXXXX
10  XXXXXXXXXXXXXX
11  XXXXXXXX000001
12  XXXXXXXXXXXXXX
13  XXXXXXXXXXXXXX
14  XXXXXXXXXXXXXX
15  XXXXXXXXXXXXXX
16  XXXXXXXXXXXXXX
17  XXXXXXXXXXXXXX
18  XXXXXXXXXXXXXX
19  XXXXXXXXXXXXXX
20  XXXXXXXXXXXXXX
21  XXXXXXXXXXXXXX
22  XXXXXXXXXXXXXX
```

Protocol Stack This column shows the protocol stack layers used in the event definition. You can select a layer to edit the fields associated with that layer.

(Protocol) This column displays the fields of the selected protocol stack layer and lets you enter the field values that define the event.

See Also "To define and edit packet events" on page 21

To run the measurement

- Select the Run button ▶ or the Group Run button ▶GROUP to start the measurement.

See Also "Displaying Data with the Network Protocol Decoder IA" on page 25

Displaying Data with the Network Protocol Decoder IA

This section describes the options available when using the network protocol decoder inverse assembler to display captured data.

- “To load the network protocol decoder inverse assembler” on page 25
- “To specify which protocol fields to display” on page 26
- “To filter data out of the display” on page 30

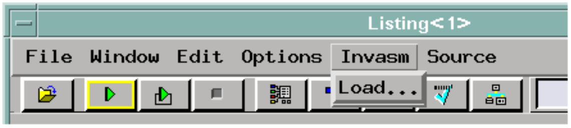
See Also

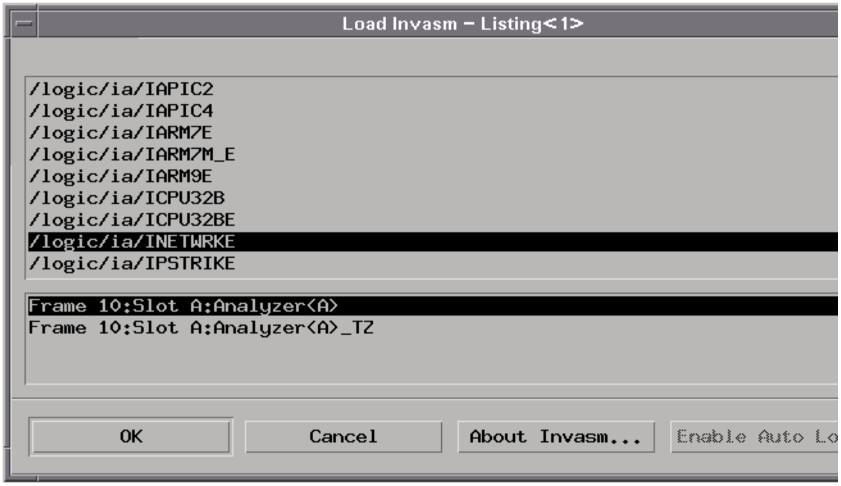
If you're capturing data on a UTOPIA bus, see “Displaying Data with the ATM Tool” on page 32.

If you're capturing data on a POS-PHY level 1-3 (packet over SONET) bus, see “Displaying Data with the POS-PHY Tool” on page 42.

To load the network protocol decoder inverse assembler

1. Open the logic analyzer's Listing window.
2. Choose the Invasm->Load... command to load the INETWRKE network protocol decoder inverse assembler.





```
/logic/ia/IAPIC2
/logic/ia/IAPIC4
/logic/ia/IARM7E
/logic/ia/IARM7M_E
/logic/ia/IARM9E
/logic/ia/ICPU32B
/logic/ia/ICPU32BE
/logic/ia/INETWRKE
/logic/ia/IPSTRIKE
```

```
Frame 10:Slot A:Analyzer<A>
Frame 10:Slot A:Analyzer<A>_TZ
```

If you don’t have a label named DATA, the inverse assembler will fail to load.

The inverse assembler will immediately try to decode the captured trace data. If no trace data has been captured, run a measurement.

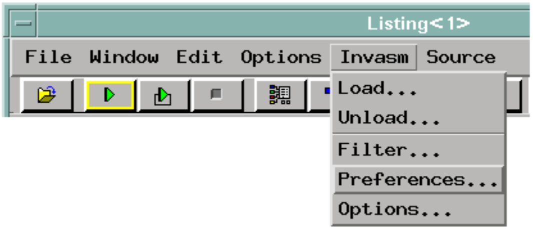
See Also

“To specify which protocol fields to display” on page 26

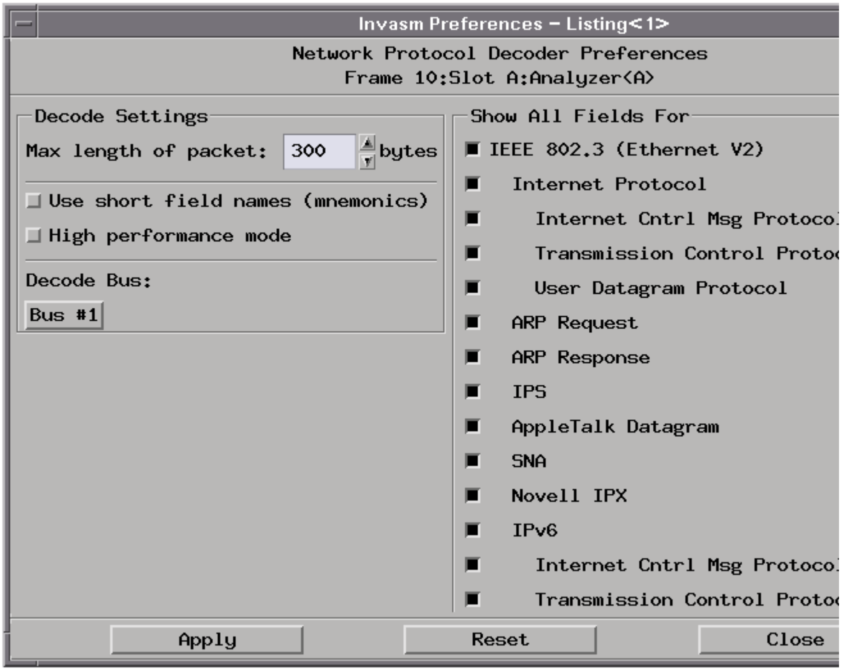
“To filter data out of the display” on page 30

To specify which protocol fields to display

1. In the logic analyzer's Listing window, choose the Invasm->Preferences... command.



2. In the Invasm Preferences dialog, make preference selections:


Invasm Preferences – Listing<1>

Network Protocol Decoder Preferences
Frame 10:Slot A:Analyzer<A>

Decode Settings

Max length of packet: 300 bytes

☐ Use short field names (mnemonics)

☐ High performance mode

Decode Bus:

Bus #1

Show All Fields For

■ IEEE 802.3 (Ethernet V2)
■ Internet Protocol
■ Internet Cntrl Msg Protoco
■ Transmission Control Proto
■ User Datagram Protocol
■ ARP Request
■ ARP Response
■ IPS
■ AppleTalk Datagram
■ SNA
■ Novell IPX
■ IPv6
■ Internet Cntrl Msg Protoco
■ Transmission Control Proto

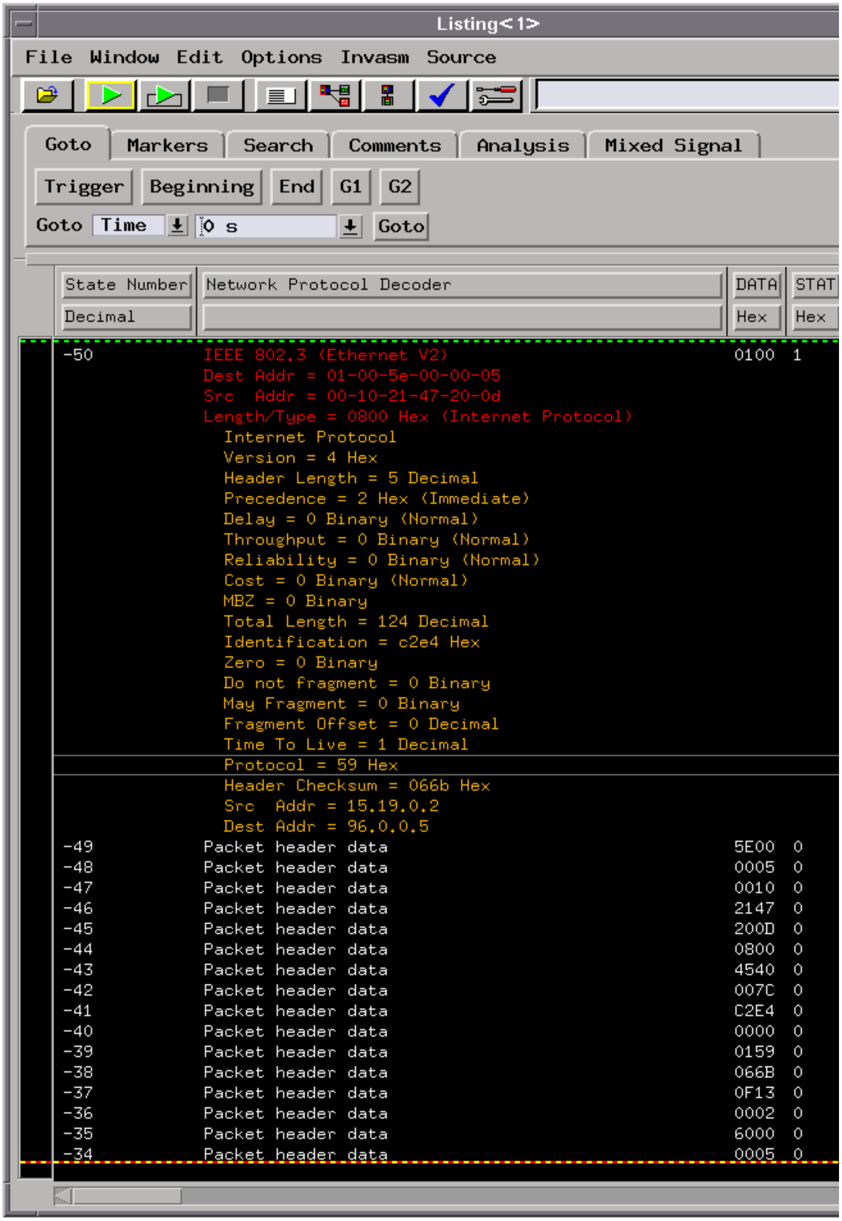
- In the Decode Settings group, specify the maximum packet length, whether to use short field names or the high performance mode, and select the kind of bus to decode.
- In the Show All Fields For group, select the fields that should be displayed in the Listing display.

NOTE: The packet data identification tells the inverse assembler the difference between idle states and data states.

When there is no packet data identification and a new trace has been captured, the inverse assembler will normally search backward in the data for a start of packet or an end of packet in order to determine whether captured states are packet data (start of packet found first) or idle states (end of packet found first).

If there are many idle states between packets, this backward searching can slow down inverse assembly. In this case, you can turn off the searching by selecting the high performance mode decode setting. When the high performance mode is selected, unknown states are assumed to be packet data.

3. Select Close (or Apply if you'd like to see the effect on the Listing window and leave the dialog open).


Listing<1>
File Window Edit Options Invasm Source
Goto Markers Search Comments Analysis Mixed Signal
Trigger Beginning End G1 G2
Goto Time 0 s Goto

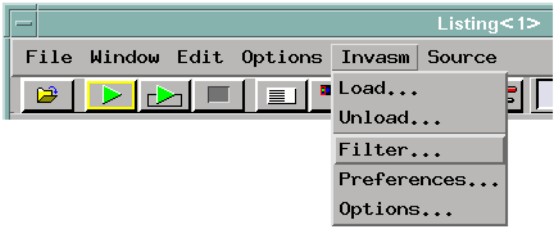
```
State Number  Network Protocol Decoder                          DATA  STAT
Decimal                                                         Hex   Hex
-50           IEEE 802.3 (Ethernet V2)                          0100  1
              Dest Addr = 01-00-5e-00-00-05
              Src  Addr = 00-10-21-47-20-0d
              Length/Type = 0800 Hex (Internet Protocol)
                Internet Protocol
                Version = 4 Hex
                Header Length = 5 Decimal
                Precedence = 2 Hex (Immediate)
                Delay = 0 Binary (Normal)
                Throughput = 0 Binary (Normal)
                Reliability = 0 Binary (Normal)
                Cost = 0 Binary (Normal)
                MBZ = 0 Binary
                Total Length = 124 Decimal
                Identification = c2e4 Hex
                Zero = 0 Binary
                Do not fragment = 0 Binary
                May Fragment = 0 Binary
                Fragment Offset = 0 Decimal
                Time To Live = 1 Decimal
                Protocol = 59 Hex
                Header Checksum = 066b Hex
                Src  Addr = 15.19.0.2
                Dest Addr = 96.0.0.5
-49           Packet header data                                5E00  0
-48           Packet header data                                0005  0
-47           Packet header data                                0010  0
-46           Packet header data                                2147  0
-45           Packet header data                                200D  0
-44           Packet header data                                0800  0
-43           Packet header data                                4540  0
-42           Packet header data                                007C  0
-41           Packet header data                                C2E4  0
-40           Packet header data                                0000  0
-39           Packet header data                                0159  0
-38           Packet header data                                066B  0
-37           Packet header data                                0F13  0
-36           Packet header data                                0002  0
-35           Packet header data                                6000  0
-34           Packet header data                                0005  0
```

See Also "Modifying Protocol Definitions" on page 52

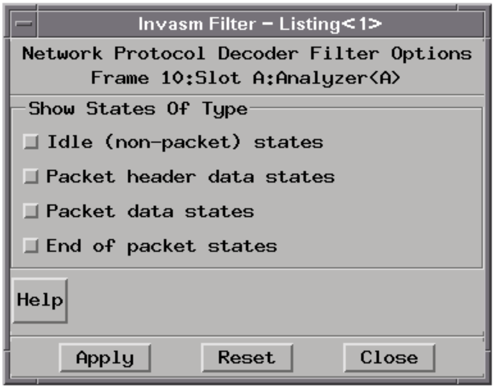
"To filter data out of the display" on page 30

To filter data out of the display

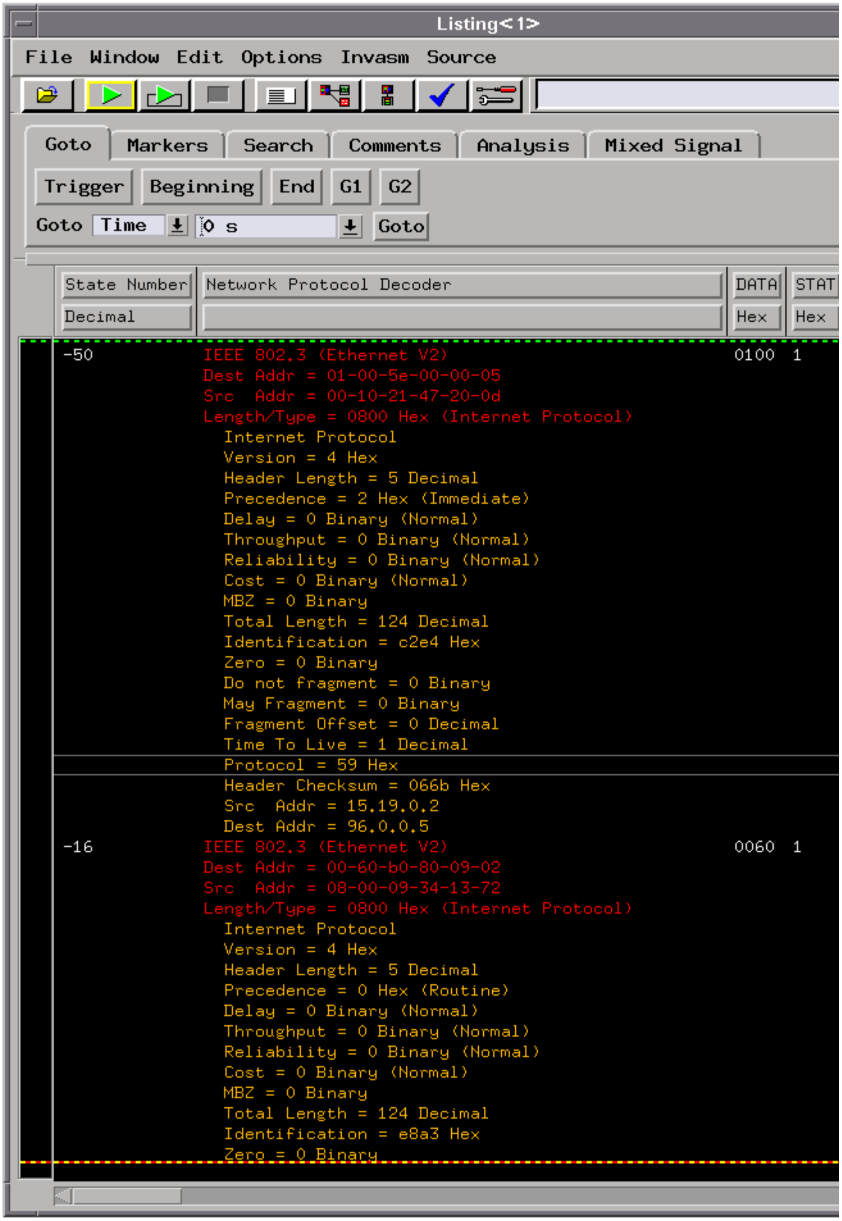
1. In the logic analyzer's Listing window, choose the Invasm->Filter... command.



2. In the Invasm Filter dialog, select the type of states that should be shown in the Listing display.


Invasm Filter – Listing<1>
Network Protocol Decoder Filter Options
Frame 10:Slot A:Analyzer<A>
Show States Of Type
□ Idle (non-packet) states
□ Packet header data states
□ Packet data states
□ End of packet states
Help
Apply
Reset
Close

3. Select Close (or Apply if you'd like to see the effect on the Listing window and leave the dialog open).



```
State Number   Network Protocol Decoder                              DATA  STAT
Decimal                                                              Hex   Hex
-50            IEEE 802.3 (Ethernet V2)                              0100  1
               Dest Addr = 01-00-5e-00-00-05
               Src  Addr = 00-10-21-47-20-0d
               Length/Type = 0800 Hex (Internet Protocol)
                 Internet Protocol
                 Version = 4 Hex
                 Header Length = 5 Decimal
                 Precedence = 2 Hex (Immediate)
                 Delay = 0 Binary (Normal)
                 Throughput = 0 Binary (Normal)
                 Reliability = 0 Binary (Normal)
                 Cost = 0 Binary (Normal)
                 MBZ = 0 Binary
                 Total Length = 124 Decimal
                 Identification = c2e4 Hex
                 Zero = 0 Binary
                 Do not fragment = 0 Binary
                 May Fragment = 0 Binary
                 Fragment Offset = 0 Decimal
                 Time To Live = 1 Decimal
                 Protocol = 59 Hex
                 Header Checksum = 066b Hex
                 Src  Addr = 15.19.0.2
                 Dest Addr = 96.0.0.5
-16            IEEE 802.3 (Ethernet V2)                              0060  1
               Dest Addr = 00-60-b0-80-09-02
               Src  Addr = 08-00-09-34-13-72
               Length/Type = 0800 Hex (Internet Protocol)
                 Internet Protocol
                 Version = 4 Hex
                 Header Length = 5 Decimal
                 Precedence = 0 Hex (Routine)
                 Delay = 0 Binary (Normal)
                 Throughput = 0 Binary (Normal)
                 Reliability = 0 Binary (Normal)
                 Cost = 0 Binary (Normal)
                 MBZ = 0 Binary
                 Total Length = 124 Decimal
                 Identification = e8a3 Hex
                 Zero = 0 Binary
```

See Also “To specify which protocol fields to display” on page 26

Displaying Data with the ATM Tool

The ATM Tool is for decoding data and detecting errors on UTOPIA buses. This section describes the options available when using the ATM Tool.

- "To connect the ATM Tool (in the workspace)" on page 32
- "To set up the ATM Tool" on page 34
- "To select the output data columns" on page 36
- "To filter data rows" on page 39

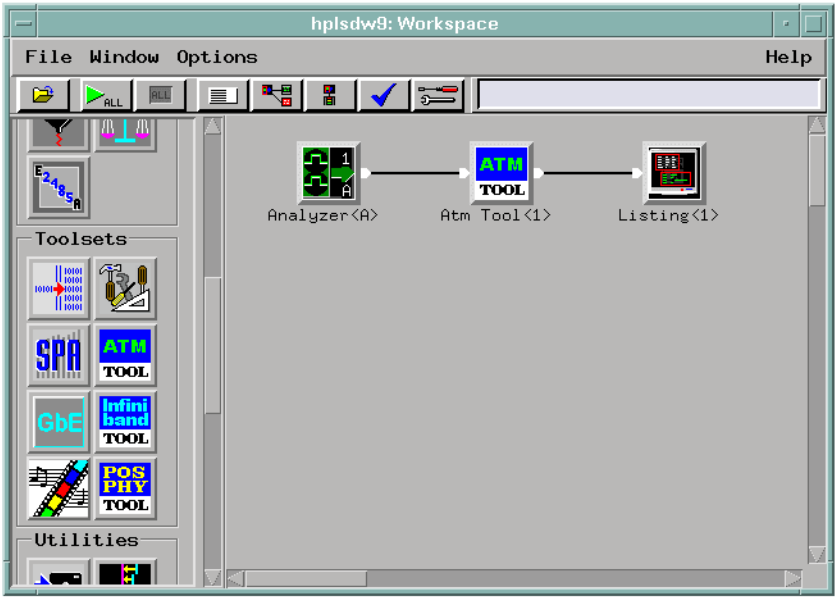
See Also

"Displaying Data with the Network Protocol Decoder IA" on page 25.

If you're capturing data on a POS-PHY level 1-3 (packet over SONET) bus, see "Displaying Data with the POS-PHY Tool" on page 42.

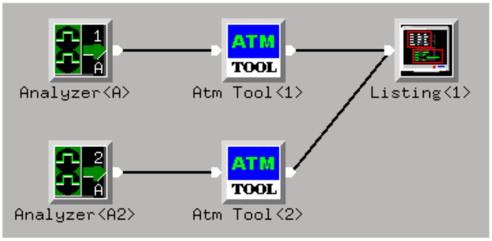
To connect the ATM Tool (in the workspace)

1. Drag the ATM Tool icon, and drop it between the logic analyzer instrument icon and the Listing display tool icon.



The ATM Tool will immediately try to decode the captured trace data. If no trace data has been captured, run a measurement.

Filtering Limitation



NOTE: When the output of two or more ATM Tools go to a single Listing display, filtering does not work.

See Also

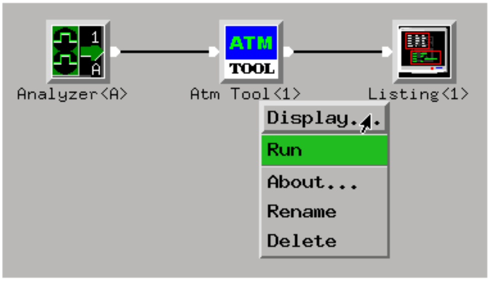
"To set up the ATM Tool" on page 34

"To select the output data columns" on page 36

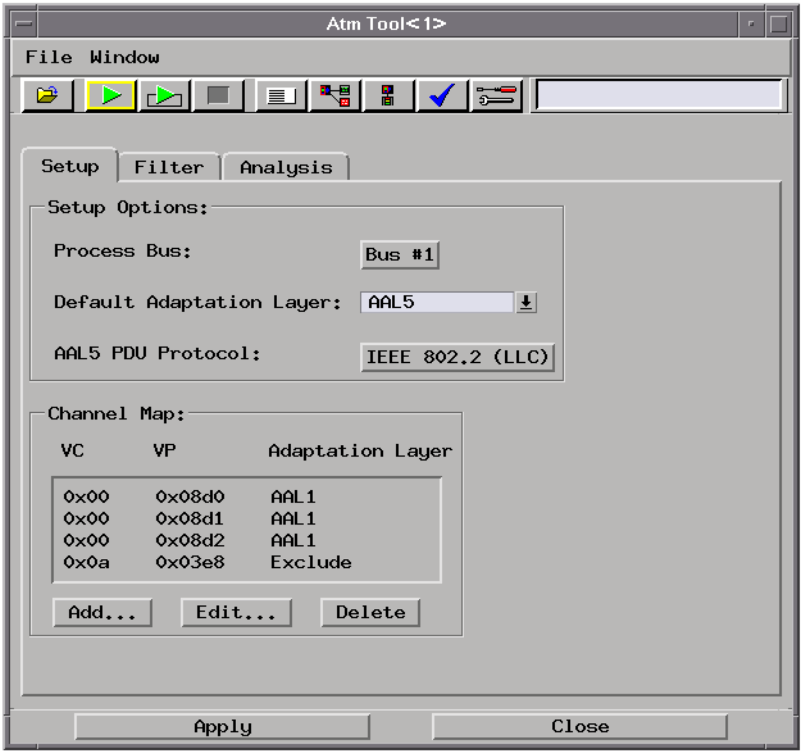
"To filter data rows" on page 39

To set up the ATM Tool

1. Display the ATM Tool.



2. In the ATM Tool's Setup tab, select the appropriate setup options:



Setup Options:

Process Bus: Bus #1

Default Adaptation Layer: AAL5

AAL5 PDU Protocol: IEEE 802.2 (LLC)

Channel Map:

VC	VP	Adaptation Layer
0x00	0x08d0	AAL1
0x00	0x08d1	AAL1
0x00	0x08d2	AAL1
0x0a	0x03e8	Exclude

Process Bus Selects the data bus to decode.

Default Adaptation Layer Selects the default ATM Adaptation Layer (AAL). You can choose from:

- AAL1 - Voice or video (constant bit rate).
- AAL2 - Voice or video (variable bit rate, compression).
- AAL5 - Data traffic.

AAL5 PDU Protocol Selects the protocol of the ATM Adaptation Layer 5 Protocol Data Unit.

3. Because it's unlikely that the logic analyzer will capture the signaling packets that contain channel setup messages, use the Channel Map

portion of the Setup dialog to identify the Virtual Channels, Virtual Paths, and their ATM Adaptation Layers.

You can use *Exclude* in the Adaptation Layer column to omit states with certain vitrual channels and paths from the output.

4. Select Apply to change the setup.
5. Select Close to change the setup and close the ATM Tool.

See Also

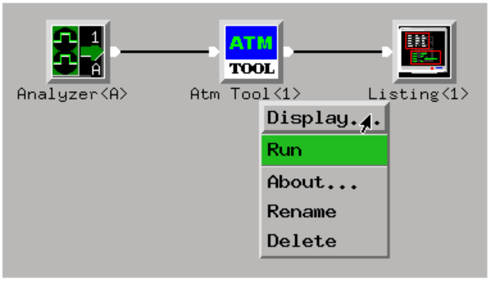
"To select the output data columns" on page 36

"To filter data rows" on page 39

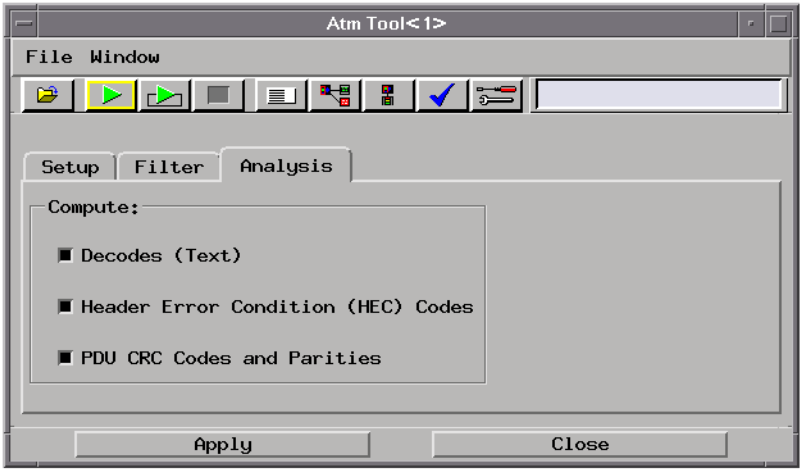
"Bus Editor Dialog" on page 20

To select the output data columns

1. Display the ATM Tool.



2. In the ATM Tool's Analysis tab, select the columns that should appear in the output data:



Compute:

■ Decodes (Text)

■ Header Error Condition (HEC) Codes

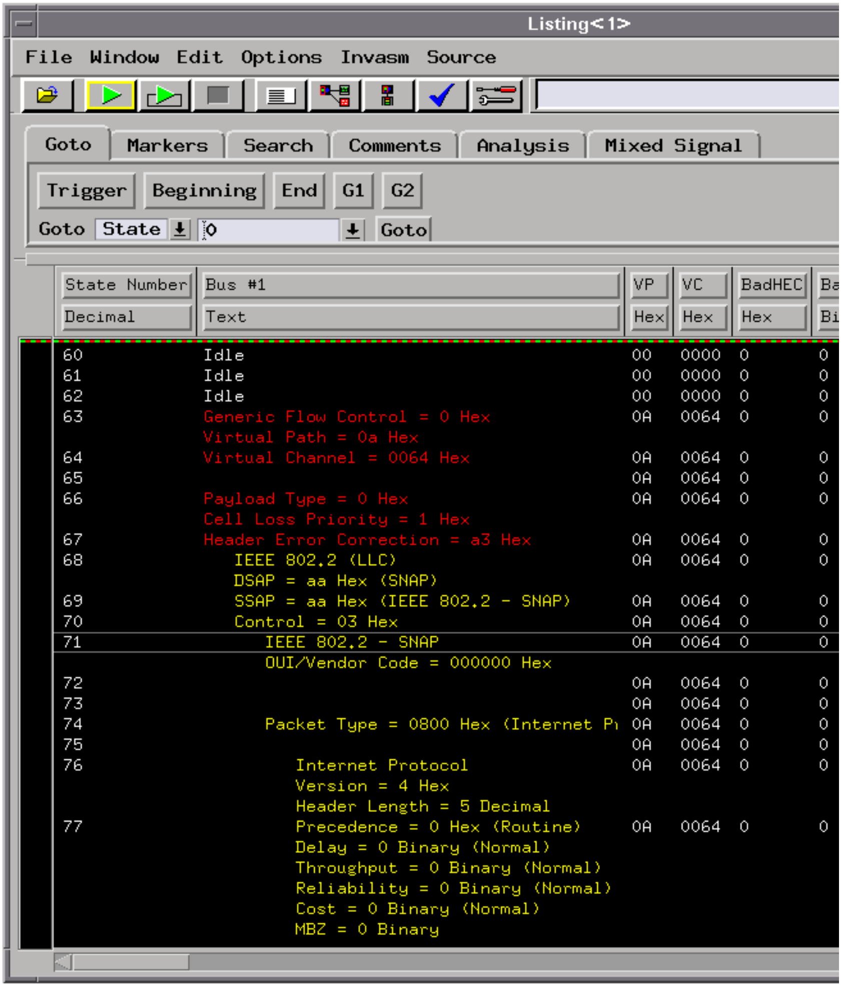
■ PDU CRC Codes and Parities

Decodes This text column contains a description of the decoded data.

Header Error Condition (HEC) Codes This 2-bit column describes the errors detected from the Header Error Control field in the cell header.

PDU CRC Codes and Parities This 1-bit column contains "1" whenever there is a Protocol Data Unit CRC code or parity error.

3. Select Apply to output the selected data columns.



```
State Number  Bus #1                                      VP   VC     BadHEC  Ba
Decimal       Text                                        Hex  Hex    Hex     Bi
60            Idle                                        00   0000   0       0
61            Idle                                        00   0000   0       0
62            Idle                                        00   0000   0       0
63            Generic Flow Control = 0 Hex                0A   0064   0       0
              Virtual Path = 0a Hex
64            Virtual Channel = 0064 Hex                  0A   0064   0       0
65                                                        0A   0064   0       0
66            Payload Type = 0 Hex                        0A   0064   0       0
              Cell Loss Priority = 1 Hex
67            Header Error Correction = a3 Hex            0A   0064   0       0
68               IEEE 802.2 (LLC)                         0A   0064   0       0
                 DSAP = aa Hex (SNAP)
69               SSAP = aa Hex (IEEE 802.2 - SNAP)        0A   0064   0       0
70               Control = 03 Hex                         0A   0064   0       0
71                  IEEE 802.2 - SNAP                     0A   0064   0       0
                    OUI/Vendor Code = 000000 Hex
72                                                        0A   0064   0       0
73                                                        0A   0064   0       0
74                  Packet Type = 0800 Hex (Internet Pı   0A   0064   0       0
75                                                        0A   0064   0       0
76                     Internet Protocol                  0A   0064   0       0
                       Version = 4 Hex
                       Header Length = 5 Decimal
77                     Precedence = 0 Hex (Routine)       0A   0064   0       0
                       Delay = 0 Binary (Normal)
                       Throughput = 0 Binary (Normal)
                       Reliability = 0 Binary (Normal)
                       Cost = 0 Binary (Normal)
                       MBZ = 0 Binary
```

4. Select Close to output the selected columns and close the ATM Tool.

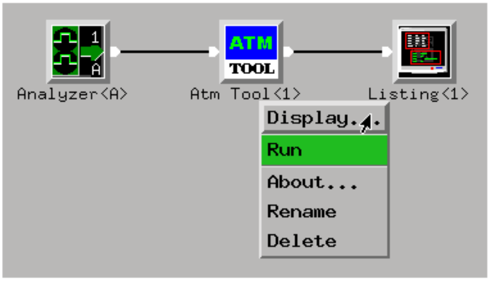
See Also

"To set up the ATM Tool" on page 34

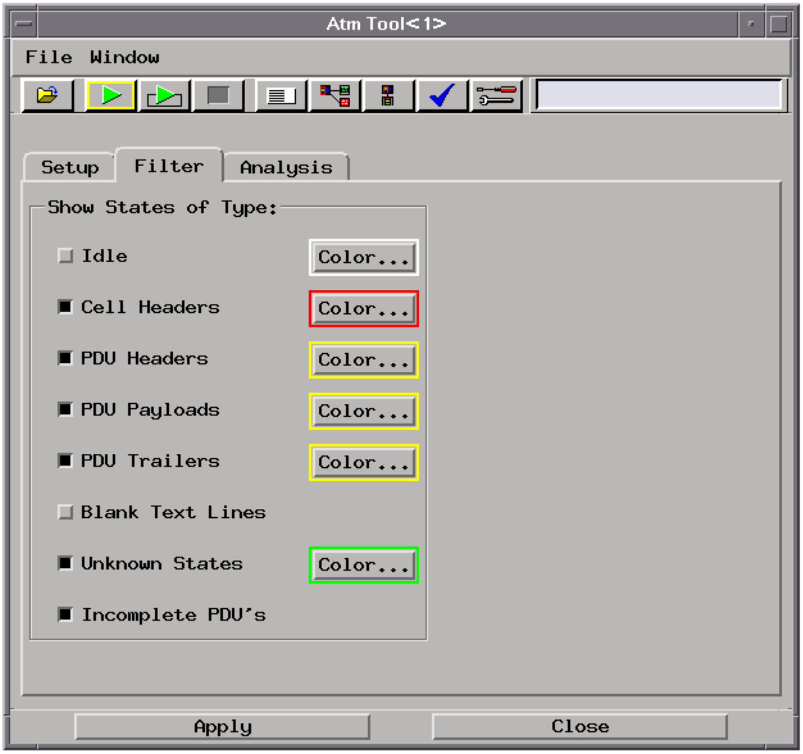
"To filter data rows" on page 39

To filter data rows

1. Display the ATM Tool.



2. In the ATM Tool’s Filter tab, select the type of states that should be output as rows of data:



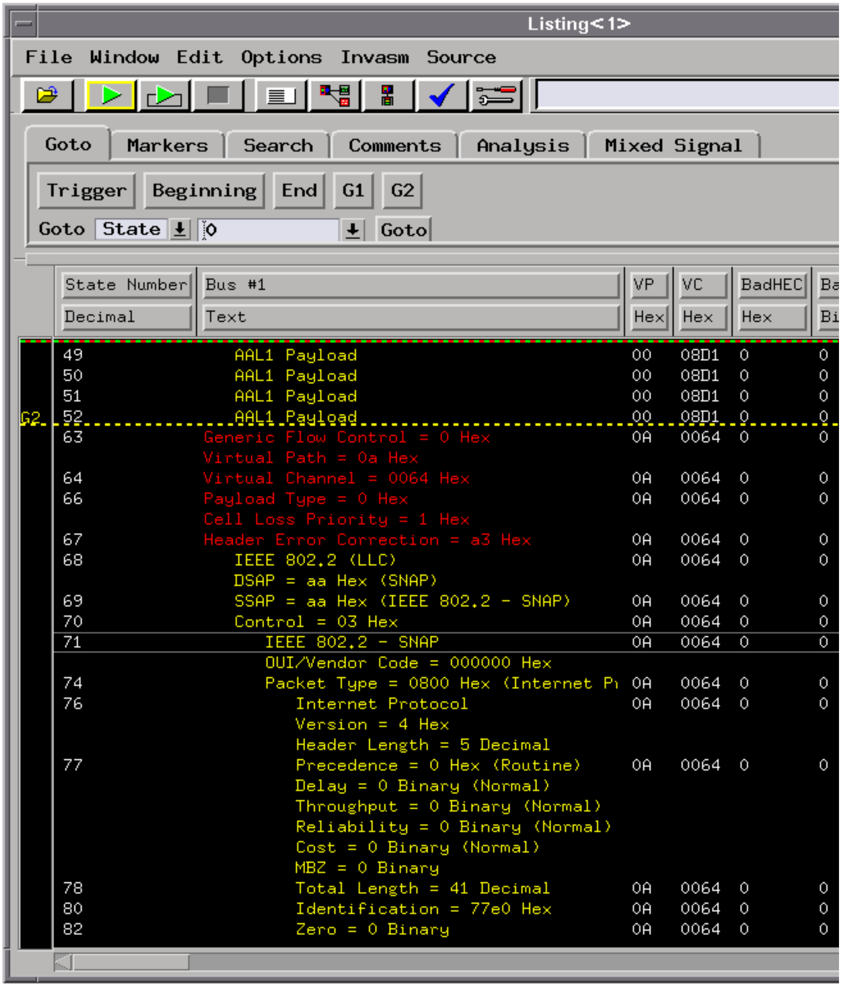
Show States of Type:

- ☐ Idle — Color...
- ■ Cell Headers — Color...
- ■ PDU Headers — Color...
- ■ PDU Payloads — Color...
- ■ PDU Trailers — Color...
- ☐ Blank Text Lines
- ■ Unknown States — Color...
- ■ Incomplete PDU's

- Idle.
- Cell Headers.
- PDU Headers.
- PDU Payloads.
- PDU Trailers.
- Blank Text Lines.
- Unknown States.
- Incomplete PDUs.

To change the color of a state type, select the Color... button.

3. Select Apply to output the selected data rows.



```
      State Number  Bus #1                                     VP   VC    BadHEC  Ba
      Decimal       Text                                       Hex  Hex   Hex     Bi
      49               AAL1 Payload                            00   08D1  0       0
      50               AAL1 Payload                            00   08D1  0       0
      51               AAL1 Payload                            00   08D1  0       0
G2    52               AAL1 Payload                            00   08D1  0       0
      63            Generic Flow Control = 0 Hex               0A   0064  0       0
                    Virtual Path = 0a Hex
      64            Virtual Channel = 0064 Hex                 0A   0064  0       0
      66            Payload Type = 0 Hex                       0A   0064  0       0
                    Cell Loss Priority = 1 Hex
      67            Header Error Correction = a3 Hex           0A   0064  0       0
      68               IEEE 802.2 (LLC)                        0A   0064  0       0
                       DSAP = aa Hex (SNAP)
      69               SSAP = aa Hex (IEEE 802.2 - SNAP)       0A   0064  0       0
      70               Control = 03 Hex                        0A   0064  0       0
      71                  IEEE 802.2 - SNAP                    0A   0064  0       0
                          OUI/Vendor Code = 000000 Hex
      74                  Packet Type = 0800 Hex (Internet Pı  0A   0064  0       0
      76                     Internet Protocol                 0A   0064  0       0
                             Version = 4 Hex
                             Header Length = 5 Decimal
      77                     Precedence = 0 Hex (Routine)      0A   0064  0       0
                             Delay = 0 Binary (Normal)
                             Throughput = 0 Binary (Normal)
                             Reliability = 0 Binary (Normal)
                             Cost = 0 Binary (Normal)
                             MBZ = 0 Binary
      78                     Total Length = 41 Decimal         0A   0064  0       0
      80                     Identification = 77e0 Hex         0A   0064  0       0
      82                     Zero = 0 Binary                   0A   0064  0       0
```

4. Select Close to close the ATM Tool.

See Also

Filtering Limitation in “To connect the ATM Tool (in the workspace)” on page 32

“To set up the ATM Tool” on page 34

“To select the output data columns” on page 36

Displaying Data with the POS-PHY Tool

The POS-PHY Tool is for decoding data and detecting errors on POS-PHY level 1-3 (packet over SONET) buses. This section describes the options available when using the POS-PHY tool.

- "To connect the POS-PHY Tool (in the workspace)" on page 42
- "To set up the POS-PHY Tool" on page 44
- "To select the output data columns" on page 46
- "To filter data rows" on page 49

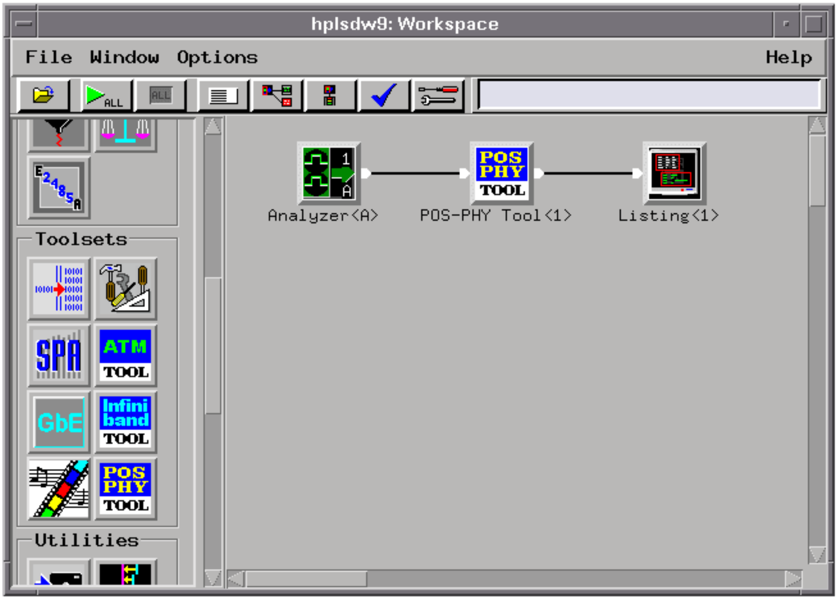
See Also

"Displaying Data with the Network Protocol Decoder IA" on page 25.

If you're capturing data on a UTOPIA bus, see "Displaying Data with the ATM Tool" on page 32.

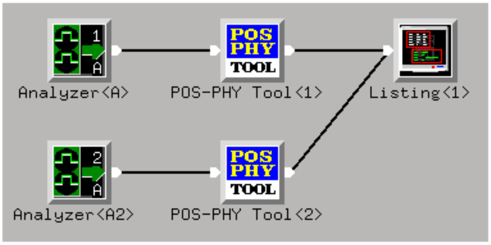
To connect the POS-PHY Tool (in the workspace)

1. Drag the POS-PHY Tool icon, and drop it between the logic analyzer instrument icon and the Listing display tool icon.



The POS-PHY Tool will immediately try to decode the captured trace data. If no trace data has been captured, run a measurement.

Filtering Limitation



NOTE: When the output of two or more POS-PHY Tools go to a single Listing display, filtering does not work.

See Also

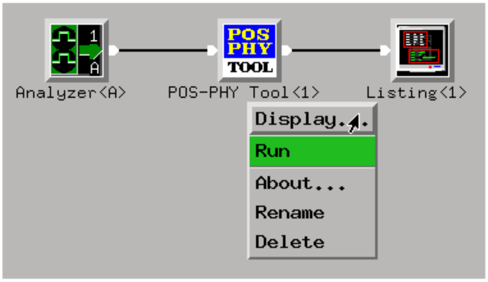
"To set up the POS-PHY Tool" on page 44

"To select the output data columns" on page 46

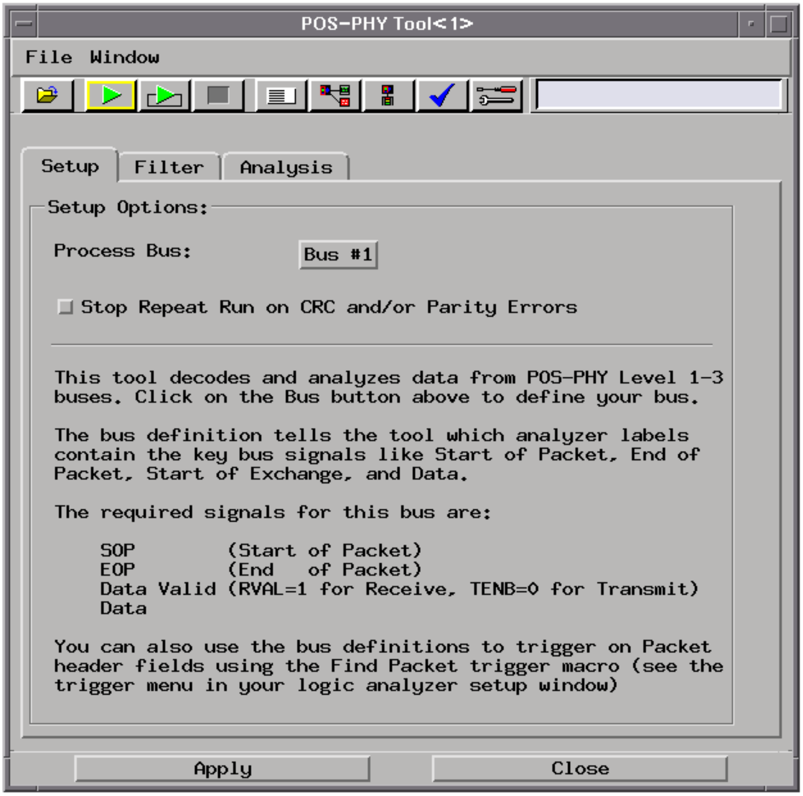
"To filter data rows" on page 49

To set up the POS-PHY Tool

1. Display the POS-PHY Tool.



2. In the POS-PHY Tool's Setup tab, select the appropriate options:



Setup Options:

Process Bus: Bus #1

□ Stop Repeat Run on CRC and/or Parity Errors

This tool decodes and analyzes data from POS-PHY Level 1-3 buses. Click on the Bus button above to define your bus.

The bus definition tells the tool which analyzer labels contain the key bus signals like Start of Packet, End of Packet, Start of Exchange, and Data.

The required signals for this bus are:

```
    SOP        (Start of Packet)
    EOP        (End   of Packet)
    Data Valid (RVAL=1 for Receive, TENB=0 for Transmit)
    Data
```

You can also use the bus definitions to trigger on Packet header fields using the Find Packet trigger macro (see the trigger menu in your logic analyzer setup window)

Process Bus Selects the data bus to decode.

Stop Repeat Run on CRC and/or Parity Errors Specifies whether errors will stop repetitive run measurements. Note that an error column must appear in the output data in order for an error of that type to be detected.

NOTE: When using repetitive runs, all of the captured data in an individual run must be decoded (in order to check for errors) before the next run can occur. If your logic analyzer has deep memory, decoding the entire trace can take a long time. Therefore, we recommend that you limit the trace memory depth to less than 1M states when using repetitive runs.

3. Select Apply to change the setup.
4. Select Close to change the setup and close the POS-PHY Tool.

See Also

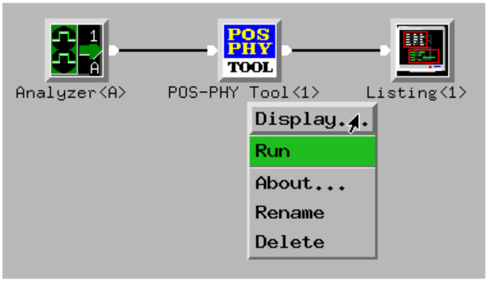
"To select the output data columns" on page 46

"To filter data rows" on page 49

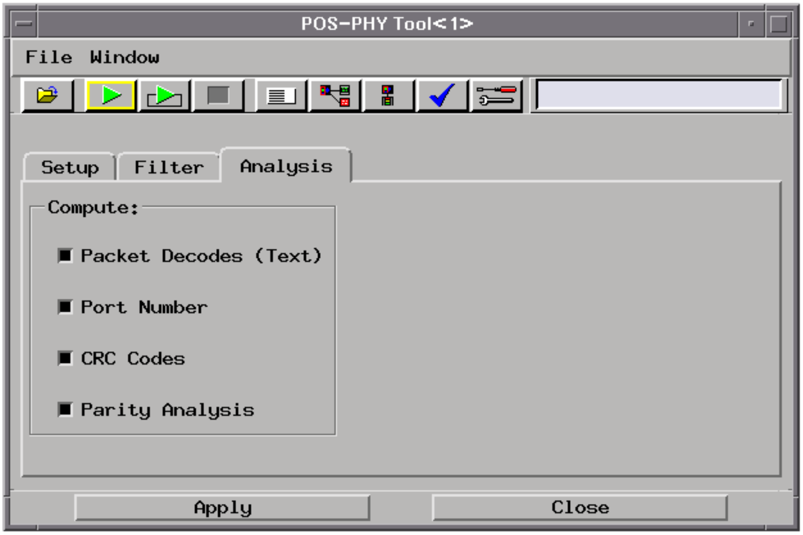
"Bus Editor Dialog" on page 20

To select the output data columns

1. Display the POS-PHY Tool.



2. In the POS-PHY Tool's Analysis tab, select the columns that should appear in the output data:



Compute:

■ Packet Decodes (Text)

■ Port Number

■ CRC Codes

■ Parity Analysis

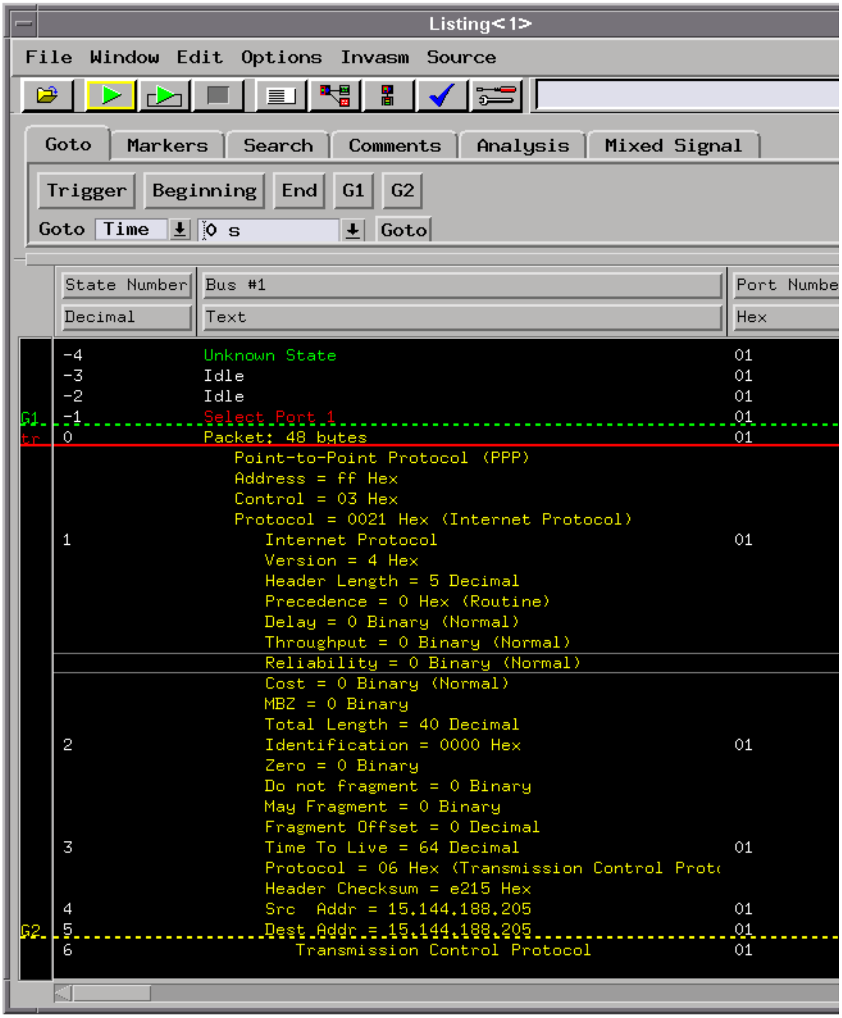
Packet Decodes This text column contains a description of the decoded data.

Port Number This 8-bit column contains the port address identified during the last RSX/TSX signal.

CRC Codes This 1-bit column contains "1" when there is a CRC error.

Parity Analysis This 1-bit column contains "1" when there is a bad parity value. This column is only available when the bus definition identified a label for the parity signal.

3. Select Apply to output the selected data columns.



```
State Number   Bus #1                                              Port Numbe
Decimal        Text                                                Hex
-4             Unknown State                                       01
-3             Idle                                                01
-2             Idle                                                01
-1             Select Port 1                                       01
0              Packet: 48 bytes                                    01
                  Point-to-Point Protocol (PPP)
                  Address = ff Hex
                  Control = 03 Hex
                  Protocol = 0021 Hex (Internet Protocol)
1                    Internet Protocol                             01
                     Version = 4 Hex
                     Header Length = 5 Decimal
                     Precedence = 0 Hex (Routine)
                     Delay = 0 Binary (Normal)
                     Throughput = 0 Binary (Normal)
                     Reliability = 0 Binary (Normal)
                     Cost = 0 Binary (Normal)
                     MBZ = 0 Binary
                     Total Length = 40 Decimal
2                    Identification = 0000 Hex                     01
                     Zero = 0 Binary
                     Do not fragment = 0 Binary
                     May Fragment = 0 Binary
                     Fragment Offset = 0 Decimal
3                    Time To Live = 64 Decimal                     01
                     Protocol = 06 Hex (Transmission Control Proto
                     Header Checksum = e215 Hex
4                    Src  Addr = 15.144.188.205                    01
5                    Dest Addr = 15.144.188.205                    01
6                       Transmission Control Protocol              01
```

4. Select Close to output the selected columns and close the POS-PHY Tool.

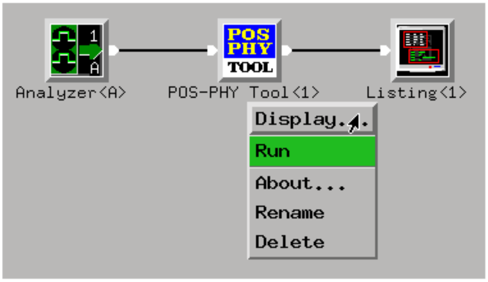
See Also

"To set up the POS-PHY Tool" on page 44

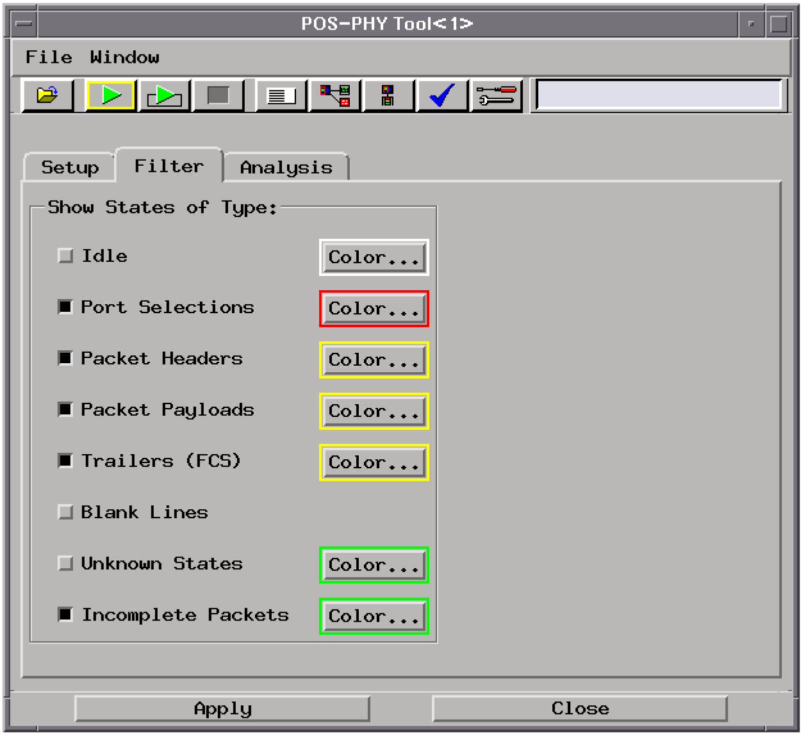
"To filter data rows" on page 49

To filter data rows

1. Display the POS-PHY Tool.



2. In the POS-PHY Tool's Filter tab, select the type of states that should be output as rows of data:



Show States of Type:

☐ Idle Color...

☒ Port Selections Color...

☒ Packet Headers Color...

☒ Packet Payloads Color...

☒ Trailers (FCS) Color...

☐ Blank Lines

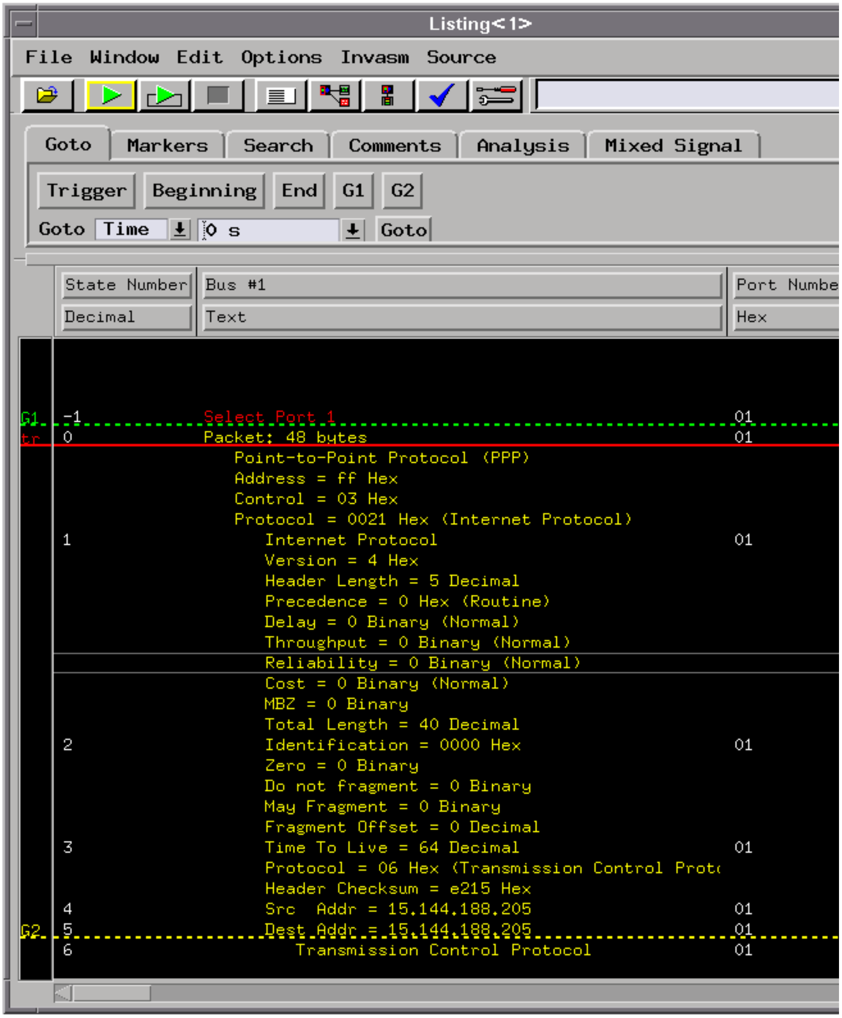
☐ Unknown States Color...

☒ Incomplete Packets Color...

- Idle.
- Port Selections.
- Packet Headers.
- Packet Payloads.
- Trailers (FCS).
- Blank Lines.
- Unknown States.
- Incomplete Packets.

To change the color of a state type, select the Color... button.

3. Select Apply to output the selected data rows.


Listing<1>
File Window Edit Options Invasm Source
Goto Markers Search Comments Analysis Mixed Signal
Trigger Beginning End G1 G2
Goto Time
0 s
Goto

```
     State Number  Bus #1                                              Port Numbe
     Decimal       Text                                                Hex

G1   -1            Select Port 1                                       01
tr    0            Packet: 48 bytes                                    01
                      Point-to-Point Protocol (PPP)
                      Address = ff Hex
                      Control = 03 Hex
                      Protocol = 0021 Hex (Internet Protocol)
      1                  Internet Protocol                             01
                         Version = 4 Hex
                         Header Length = 5 Decimal
                         Precedence = 0 Hex (Routine)
                         Delay = 0 Binary (Normal)
                         Throughput = 0 Binary (Normal)
                         Reliability = 0 Binary (Normal)
                         Cost = 0 Binary (Normal)
                         MBZ = 0 Binary
                         Total Length = 40 Decimal
      2                  Identification = 0000 Hex                     01
                         Zero = 0 Binary
                         Do not fragment = 0 Binary
                         May Fragment = 0 Binary
                         Fragment Offset = 0 Decimal
      3                  Time To Live = 64 Decimal                     01
                         Protocol = 06 Hex (Transmission Control Prot(
                         Header Checksum = e215 Hex
      4                  Src  Addr = 15.144.188.205                    01
G2    5                  Dest Addr = 15.144.188.205                    01
      6                     Transmission Control Protocol              01
```

4. Select Close to close the POS-PHY Tool.

See Also

Filtering Limitation in “To connect the POS-PHY Tool (in the workspace)” on page 42

“To set up the POS-PHY Tool” on page 44

“To select the output data columns” on page 46

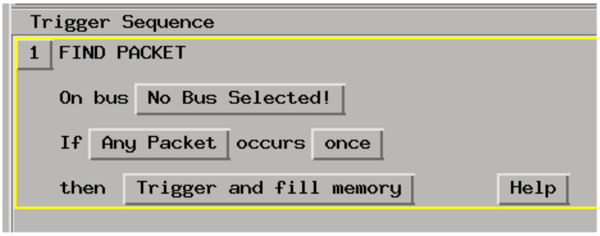
Modifying Protocol Definitions

This section shows you how to modify the protocol definitions used with the DataComm Analysis Toolset.

- “To choose a protocol” on page 52
- “To modify a protocol definition” on page 57
- “To reload the protocol definitions” on page 58
- “To reset the protocol definitions” on page 59

To choose a protocol

- In the "Find Packet" trigger function, in the Invasm Preferences dialog, in the ATM Tool's Setup tab, or in the POS-PHY Tool's Setup tab, select the bus button.



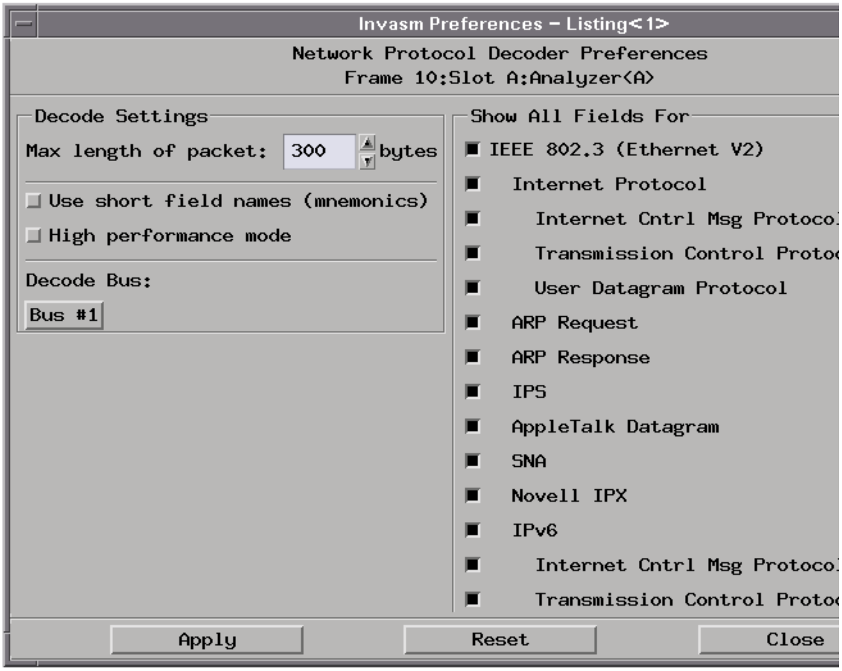
Trigger Sequence

1 FIND PACKET

On bus No Bus Selected!

If Any Packet occurs once

then Trigger and fill memory



Invasm Preferences – Listing<1>

Network Protocol Decoder Preferences
Frame 10:Slot A:Analyzer<A>

Decode Settings

Max length of packet: 300 bytes

☐ Use short field names (mnemonics)

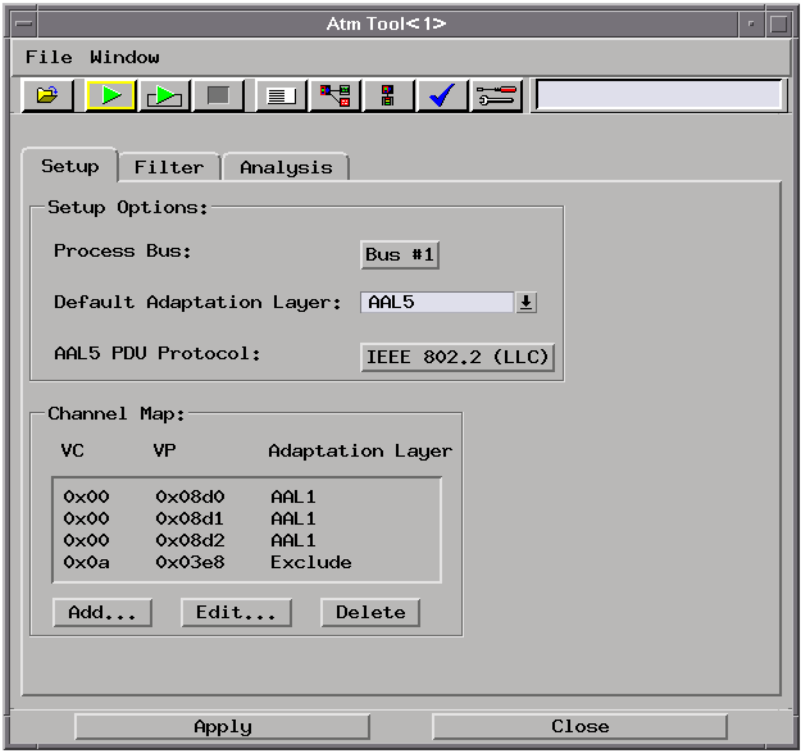
☐ High performance mode

Decode Bus:

Bus #1

Show All Fields For

■ IEEE 802.3 (Ethernet V2)
■ Internet Protocol
■ Internet Cntrl Msg Protoco
■ Transmission Control Proto
■ User Datagram Protocol
■ ARP Request
■ ARP Response
■ IPS
■ AppleTalk Datagram
■ SNA
■ Novell IPX
■ IPv6
■ Internet Cntrl Msg Protoco
■ Transmission Control Proto



Setup Options:

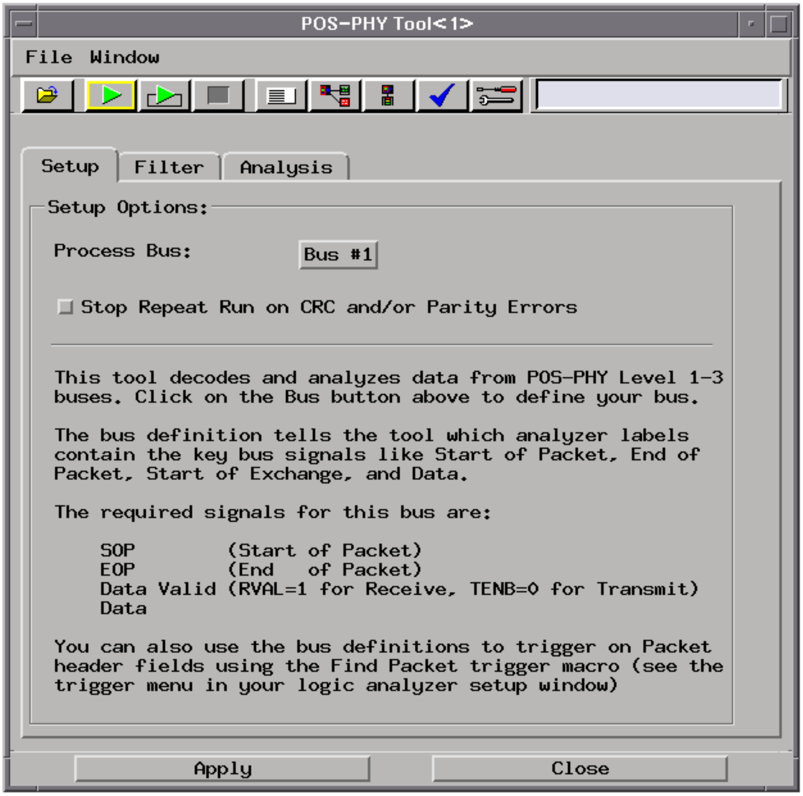
Process Bus: Bus #1

Default Adaptation Layer: AAL5

AAL5 PDU Protocol: IEEE 802.2 (LLC)

Channel Map:

VC	VP	Adaptation Layer
0x00	0x08d0	AAL1
0x00	0x08d1	AAL1
0x00	0x08d2	AAL1
0x0a	0x03e8	Exclude



Setup Options:

Process Bus: Bus #1

□ Stop Repeat Run on CRC and/or Parity Errors

This tool decodes and analyzes data from POS-PHY Level 1-3 buses. Click on the Bus button above to define your bus.

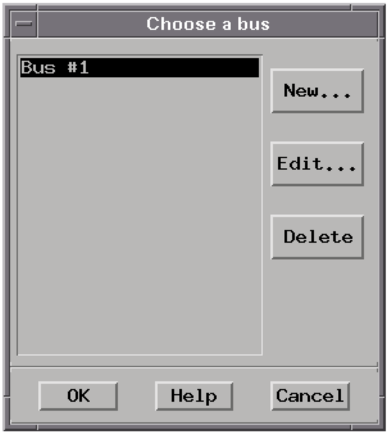
The bus definition tells the tool which analyzer labels contain the key bus signals like Start of Packet, End of Packet, Start of Exchange, and Data.

The required signals for this bus are:

```
    SOP        (Start of Packet)
    EOP        (End   of Packet)
    Data Valid (RVAL=1 for Receive, TENB=0 for Transmit)
    Data
```

You can also use the bus definitions to trigger on Packet header fields using the Find Packet trigger macro (see the trigger menu in your logic analyzer setup window)

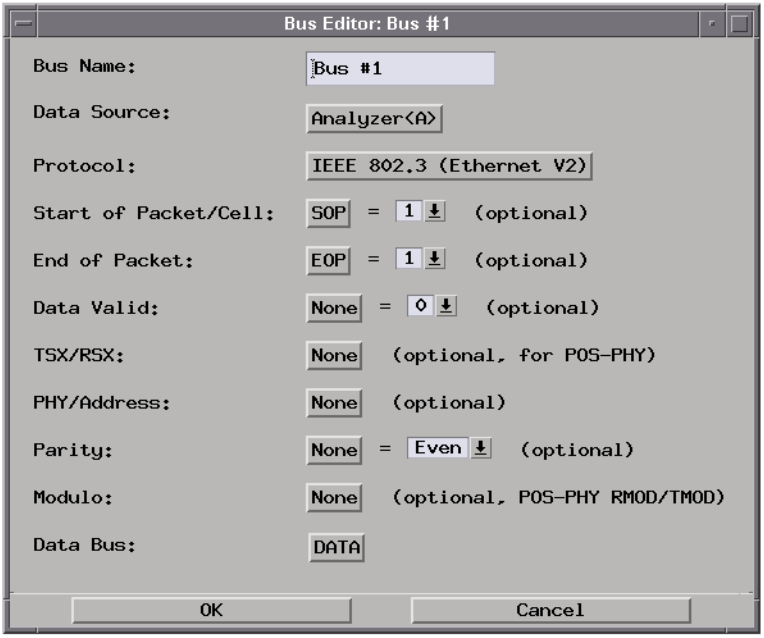
- In the Choose a Bus dialog, either select New... to define a new bus or select the bus you want to edit from the list and select the Edit... button.



Choose a bus

Bus #1

- In the Bus Editor dialog, select the Protocol button.



Bus Editor: Bus #1

Bus Name: Bus #1

Data Source: Analyzer<A>

Protocol: IEEE 802.3 (Ethernet V2)

Start of Packet/Cell: SOP = 1 (optional)

End of Packet: EOP = 1 (optional)

Data Valid: None = 0 (optional)

TSX/RSX: None (optional, for POS-PHY)

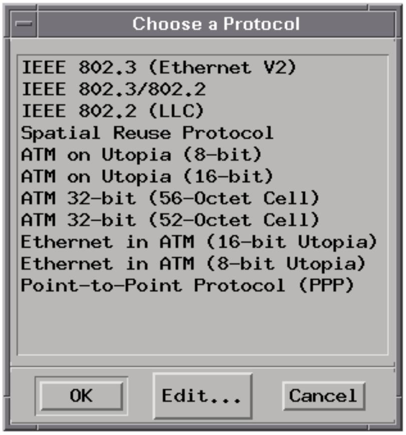
PHY/Address: None (optional)

Parity: None = Even (optional)

Modulo: None (optional, POS-PHY RMOD/TMOD)

Data Bus: DATA

- In the Choose a Protocol dialog, select the protocol from the list.



IEEE 802.3 (Ethernet V2)
IEEE 802.3/802.2
IEEE 802.2 (LLC)
Spatial Reuse Protocol
ATM on Utopia (8-bit)
ATM on Utopia (16-bit)
ATM 32-bit (56-Octet Cell)
ATM 32-bit (52-Octet Cell)
Ethernet in ATM (16-bit Utopia)
Ethernet in ATM (8-bit Utopia)
Point-to-Point Protocol (PPP)

See Also

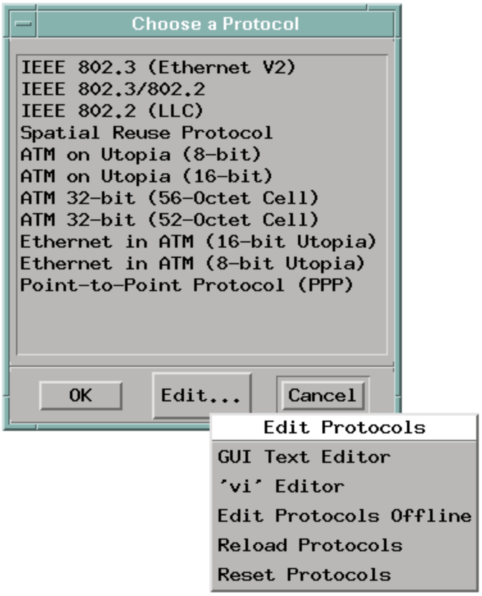
“To modify a protocol definition” on page 57

“To reload the protocol definitions” on page 58

“To reset the protocol definitions” on page 59

To modify a protocol definition

1. Open the Choose a Protocol dialog (see “To choose a protocol” on page 52), select the Edit... button, and choose either GUI Text Editor, 'vi' Editor, or Edit Protocols Offline.



IEEE 802.3 (Ethernet V2)
IEEE 802.3/802.2
IEEE 802.2 (LLC)
Spatial Reuse Protocol
ATM on Utopia (8-bit)
ATM on Utopia (16-bit)
ATM 32-bit (56-Octet Cell)
ATM 32-bit (52-Octet Cell)
Ethernet in ATM (16-bit Utopia)
Ethernet in ATM (8-bit Utopia)
Point-to-Point Protocol (PPP)

2. When you are done editing protocol definitions, choose the Reload Protocols command to tell the DataComm Analysis Toolset that changes have been made (see “To reload the protocol definitions” on page 58).

About the Protocol Definitions File

The protocol definitions file is:

```
/logic/auxiliary/TeleCom/protocols/protocols.pro
```

When the logic analysis system’s file system is mounted by (or made available to) another computer on the network, you can use any ASCII text editor to modify the protocol definitions file. Don’t forget to use the Reload Protocols command afterward.

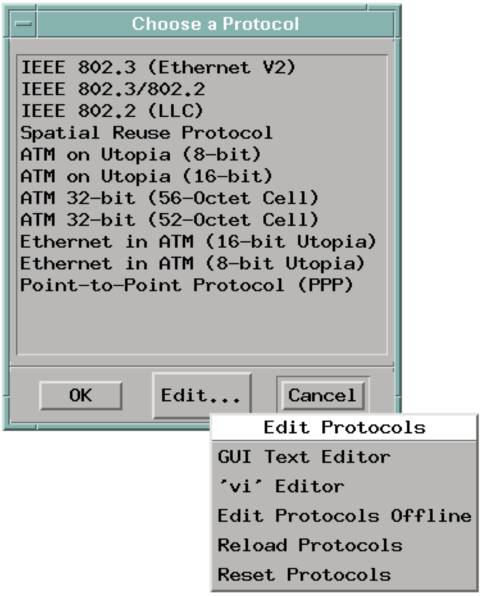
See Also

“Protocol Definition Syntax” on page 62

To reload the protocol definitions

After the protocol definitions file has been edited with "vi" (or some other ASCII text editor on a networked computer), you must use the Reload Protocols command to tell the DataComm Analysis Toolset that changes have been made.

1. Open the Choose a Protocol dialog (see “To choose a protocol” on page 52), select the Edit... button, and choose Reload Protocols.

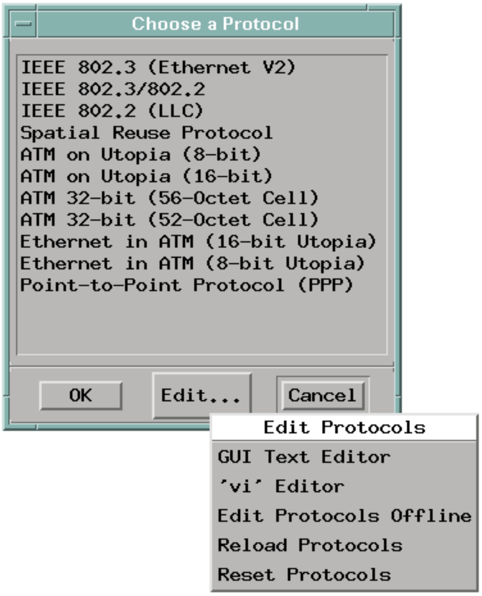


To reset the protocol definitions

If you want to return to the factory default protocol definitions, you can use the Reset Protocols command.

CAUTION: Resetting the protocol definitions will delete any new or modified protocol definitions.

1. Open the Choose a Protocol dialog (see “To choose a protocol” on page 52), select the Edit... button, and choose Reset Protocols.



Choose a Protocol

IEEE 802.3 (Ethernet V2)
IEEE 802.3/802.2
IEEE 802.2 (LLC)
Spatial Reuse Protocol
ATM on Utopia (8-bit)
ATM on Utopia (16-bit)
ATM 32-bit (56-Octet Cell)
ATM 32-bit (52-Octet Cell)
Ethernet in ATM (16-bit Utopia)
Ethernet in ATM (8-bit Utopia)
Point-to-Point Protocol (PPP)

3

Reference

Protocol Definition Syntax

This section describes the protocol definition syntax.

Protocol definitions are used by: the Event Editor to let you set up packet triggers, and by the network protocol decoder inverse assembler, ATM Tool, and POS-PHY Tool to decode captured data.

- "Protocol Block" on page 63
- "Physical Layer Definition" on page 63
- "Header Block" on page 63
- "Field Definition" on page 64
  - "Data" on page 65
  - "Symbols" on page 65
  - "ProtocolIndicator" on page 65
- "Numeric Values" on page 66

Example

Here is the definition for the IEEE 802.3 (Ethernet V2) protocol:

```
Protocol
{
  Name "IEEE 802.3 (Ethernet V2)"
  PhysicalLayer 1

  Header
  {
    DA  "Dest Addr"    48  HardwareAddress  Data
    SA  "Src  Addr"    48  HardwareAddress  Data
    PT  "Length/Type"  16  Hex              ProtocolIndicator
    {
      "Internet Protocol"   #h0800
      "ARP Request"         #h0806
      "ARP Response"        #h0835
      "AppleTalk Datagram"  #h809B
      "SNA"                 #h80d5
      "Novell IPX"          #h8137
      "IPv6"                #h86dd
      "IPS"                 #h2007
    }
  }
}
```

See Also

"Displaying Data with the Network Protocol Decoder IA" on page 25

Protocol Block

Syntax

```
Protocol
{
  Name "protocol name"

}
```

A protocol definition starts with the keyword "Protocol"; then, the definition is enclosed in brackets "{ }".

The first line inside defines the name of the protocol (which must be enclosed in double quotes).

Example

```
Protocol
{
  Name "IEEE 802.3 (Ethernet V2)"

}
```

See Also

"Physical Layer Definition" on page 63

"Header Block" on page 63

Physical Layer Definition

Syntax

```
PhysicalLayer 1
```

After the Name definition in a Protocol block is an optional physical layer definiton.

The "PhysicalLayer" definition is only required if the protocol can be used as a MAC layer or Physical layer. In other words, it is used for protocols that are at the bottom of the protocol stack on an Gbit Ethernet data bus.

Example

```
PhysicalLayer  1
```

Header Block

After the Protocol block and possibly a physical layer definition is the list of fields in the header, enclosed in the "Header { }" block.

Example

```
Header
{
  field definitions
}
```

See Also

"Field Definition" on page 64

Field Definition

Syntax

```
'Mnemonic' "'Full Name'" 'Width(in bits)' 'Format' 'Type'
```

For each field defined in a Header block, there is a line with a short 2- or 3-letter Mnemonic, a full name (enclosed in double-quotes), the length of the field (in bits), a format specification, and a type indicator.

The choices for field format are:

- Binary.
- Octal.
- Hex.
- Decimal.
- DotNotation (IP address notation, only available on 32-bit fields).
- HardwareAddress (ethernet hardware address, only available on 48-bit fields).

The choices for field type are:

- "Data" on page 65.
- "Symbols" on page 65.
- "ProtocolIndicator" on page 65.

Example

```
DA  "Dest Addr"  48  HardwareAddress  Data
```

This is the first field in the Ethernet header, the Destination Address. It is 48 bits long. It should be displayed as a Hardware Address (for example, 11-22-33-44-55-66), and it's a "Data" field.

Data

Data says to display the numeric value of the field in the format specified.

See Also

"Field Definition" on page 64

Symbols

The Symbols field type says there is a table of text names for various values in the field. If a value matches one of these, the decoder will display the name of the value, instead of the numeric value. Otherwise, the field will be displayed as a numeric value, in the format specified.

Example

Here is the "Precedence" field from the IP (Internet Protocol) definition:

```
PR  "Precedence"  3  Hex  Symbols
{
  "Routine"                0
  "Priority"               1
  "Immediate"              2
  "Flash"                  3
  "Flash Override"         4
  "Internetwork Control"   6
  "Network Control"        7
}
```

Notice that the "Symbols" field type is followed by a symbol table description, enclosed in "{ }" brackets. For each symbol, there is a name (in double-quotes) and a numeric value.

See Also

"Field Definition" on page 64

"Numeric Values" on page 66

ProtocolIndicator

The ProtocolIndicator field type is how a protocol references the next layer of the protocol stack.

The syntax for this type is exactly the same as for "Symbols" on page 65, except that each symbol's name is the name of another protocol, which must also be defined in the protocol file.

Example

Here is the "Length/Type" field of the Ethernet protocol:

```
PT  "Length/Type"  16  Hex  ProtocolIndicator
```

```
{
  "Internet Protocol"            #h0800
  "ARP Request"                  #h0806
  "ARP Response"                 #h0835
  "AppleTalk Datagram Protocol"  #h809B
  "Novell IPX"                   #h8137
  "IPS"                          #h2007
}
```

A value of 0x0800 in the length/type field indicates Internet Protocol, which must also be defined in the protocol definitions file.

NOTE: While many protocols are already defined, not all possible values and protocols have been included in the protocol definition.

See Also

“Field Definition” on page 64

“Symbols” on page 65

“Numeric Values” on page 66

Numeric Values

Numeric values can be plain decimal numbers, or they can be hex, octal, or binary numbers in the following formats:

```
For Hex,  a number is specified like this:  #hff00
For Binary, like this:                      #b1111111100000000
For Octal,  like this:                      #q177400
```

The reason for these somewhat unusual formats is because the letter ’X’ often means "Don’t Care" in the logic analysis system, so 0xff could be misinterpreted as a value with a don’t care digit. And, the ’q’ in the Octal specifier is to avoid the similarity between the letter ’o’ and the number ’0’.

Release Notes

This section contains notes about the DataComm Analysis Toolset.

- "Working with Packet-in-ATM Cell Data" on page 67

Working with Packet-in-ATM Cell Data

When packet data is traveling across a Utopia bus in ATM cells, the physical layer protocol is ATM. Unfortunately, ATM has no "Protocol Indicator" field in it.

ATM also has no "First in Sequence" bit to indicate that a cell is the first of a sequence. There is only a "Last in Sequence" bit. This makes triggering on packet headers inside ATM cells a little tricky.

To help, we've defined some new protocols, like "Ethernet in ATM on 16-bit Utopia". This simply makes a single header that contains the ATM header immediately followed by the Ethernet fields, essentially collapsing the two layers into one. Then, the Ethernet protocol has a Protocol Indicator (type field) to point to higher layers.

This lets you trigger on upper protocol layers in an ATM cell, but it has a few side effects:

1. You will be searching every ATM cell as though it was the first in sequence; there's a small chance of a false positive because the bits you are looking for in the packet header could occur again somewhere in a payload cell.

   We will work on this issue shortly; you can perhaps first find a packet that is the Last in sequence; then, search the next packet for the header values by using one "Find Packet" trigger function followed by another.

2. The Listing tool will decode every cell as though it were the first in sequence. The cells that ARE first in sequence will decode properly, but the others will contain bogus fields.

   We will work for a future release to either perform cell reconstruction or to only decode cells when a VPI/VCI changes or a "Last in Sequence" bit is set.

Glossary

absolute Denotes the time period or count of states between a captured state and the trigger state. An absolute count of -10 indicates the state was captured ten states before the trigger state was captured.

acquisition Denotes one complete cycle of data gathering by a measurement module. For example, if you are using an analyzer with 128K memory depth, one complete acquisition will capture and store 128K states in acquisition memory.

analysis probe A probe connected to a microprocessor or standard bus in the device under test. An analysis probe provides an interface between the signals of the microprocessor or standard bus and the inputs of the logic analyzer. Also called a *preprocessor.*

analyzer 1 In a logic analyzer with two *machines*, refers to the machine that is on by default. The default name is *Analyzer<N>*, where N is the slot letter.

analyzer 2 In a logic analyzer with two *machines*, refers to the machine that is off by default. The default name is *Analyzer<N2>*, where N is the slot letter.

arming An instrument tool must be armed before it can search for its trigger condition. Typically, instruments are armed immediately when *Run* or *Group Run* is selected. You can set up one instrument to arm another using the *Intermodule Window*. In these setups, the second instrument cannot search for its trigger condition until it receives the arming signal from the first instrument. In some analyzer instruments, you can set up one analyzer *machine* to arm the other analyzer machine in the *Trigger Window*.

asterisk (*) See *edge terms*, *glitch*, and *labels*.

bits Bits represent the physical logic analyzer channels. A bit is a *channel* that has or can be assigned to a *label*. A bit is also a position in a label.

card This refers to a single instrument intended for use in the Agilent Technologies 16600A-series or 16700A/B-series mainframes. One card fills one slot in the mainframe. A module may comprise a single card or multiple cards cabled together.

cell The basic unit of transmission in an ATM network. It is a fixed-size *packet* of 53 bytes, made up of 5 header bytes and 48 payload bytes.

channel The entire signal path from the probe tip, through the cable and module, up to the label grouping.

click When using a mouse as the pointing device, to click an item, position the cursor over the item. Then quickly press and release the *left mouse button*.

clock channel A logic analyzer *channel* that can be used to carry the clock signal. When it is not needed for clock signals, it can be used as a *data channel*, except in the Agilent Technologies 16517A.

count The count function records periods of time or numbers of state transactions between states stored in memory. You can set up the analyzer count function to count occurrences of a selected event during the trace, such as counting how many times a variable is read between each of the writes to the variable. The analyzer can also be set up to count elapsed time, such as counting the time spent executing within a particular function during a run of your target program.

CRC (Cyclic Redundancy Check)

A common technique for detecting data transmission errors.

cross triggering Using intermodule capabilities to have measurement modules trigger each other. For example, you can have an external instrument arm a logic analyzer, which subsequently triggers an oscilloscope when it finds the trigger state.

data channel A *channel* that carries data. Data channels cannot be used to clock logic analyzers.

data set A data set is made up of all labels and data stored in memory of any single analyzer machine or instrument tool. Multiple data sets can be displayed together when sourced into a single display tool. The Filter tool is used to pass on partial data sets to analysis or display tools.

delay The delay function sets the horizontal position of the waveform on the screen for the oscilloscope and timing analyzer. Delay time is measured from the trigger point in seconds or states.

deskewing To cancel or nullify the effects of differences between two different internal delay paths for a signal. Deskewing is normally done by routing a single test signal to the inputs of two different modules, then adjusting the Intermodule Skew so that both modules recognize the signal at the same time.

device under test The system under test, which contains the circuitry you are probing. Also known as a *target system*.

don't care For *terms*, a "don't care" means that the state of the signal (high or low) is not relevant to the measurement. The analyzer ignores the state of this signal when determining whether a match occurs on an input label. "Don't care" signals are still sampled and their values can be displayed with the rest of the data. Don't cares are represented by the *X* character in numeric values and the dot (.) in timing edge specifications.

dot (.) See *edge terms*, *glitch*, *labels*, and *don't care*.

double-click When using a mouse as the pointing device, to double-click an item, position the cursor over the item, and then quickly press and release the *left mouse button* twice.

drag and drop Using a Mouse: Position the cursor over the item, and then press and hold the *left mouse button*. While holding the left mouse button down, move the mouse to drag the item to a new location. When the item is positioned where you want it, release the mouse button.

Using the Touchscreen: Position your finger over the item, then press and hold finger to the screen. While holding the finger down, slide the finger along the screen dragging the item to a new location. When the item is positioned where you want it, release your finger.

edge terms Logic analyzer trigger resources that allow detection of transitions on a signal. An edge term can be set to detect a rising edge, falling edge, or either edge. Some logic analyzers can also detect no edge or a *glitch* on an input signal. Edges are specified by selecting arrows. The dot (.) ignores the bit. The asterisk (*) specifies a glitch on the bit.

events Events are the things you are looking for in your target system. In the logic analyzer interface, they take a single line. Examples of events are *Label1 = XX* and *Timer 1 > 400 ns*.

filter expression The filter expression is the logical *OR* combination of all of the filter terms. States in your data that match the filter expression can be filtered out or passed through the Pattern Filter.

filter term A variable that you

define in order to specify which states to filter out or pass through. Filter terms are logically OR'ed together to create the filter expression.

Format The selections under the logic analyzer *Format* tab tell the logic analyzer what data you want to collect, such as which channels represent buses (labels) and what logic threshold your signals use.

frame The Agilent Technologies 16600A-series or 16700A/B-series logic analysis system mainframe. See also *logic analysis system*.

glitch A glitch occurs when two or more transitions cross the logic threshold between consecutive timing analyzer samples. You can specify glitch detection by choosing the asterisk (*) for *edge terms* under the timing analyzer Trigger tab.

grouped event A grouped event is a list of *events* that you have grouped, and optionally named. It can be reused in other trigger sequence levels. Only available in Agilent Technologies 16715A, 16716A, and 16717A logic analyzers.

held value A value that is held until the next sample. A held value can exist in multiple data sets.

intermodule bus The intermodule bus (IMB) is a bus in the frame that allows the measurement modules to communicate with each other. Using the IMB, you can set up one instrument to *arm* another. Data acquired by instruments using the IMB is time-correlated.

intermodule Intermodule is a term used when multiple instrument tools are connected together for the purpose of one instrument arming another. In such a configuration, an arming tree is developed and the group run function is designated to start all instrument tools. Multiple instrument configurations are done in the Intermodule window.

labels Labels are used to group and identify logic analyzer channels. A label consists of a name and an associated bit or group of bits. Labels are created in the Format tab.

local session A local session is when you run the logic analysis system using the local display connected to the product hardware.

logic analysis system The Agilent Technologies 16600A-series or 16700A/B-series mainframes, and all tools designed to work with it. Usually used to mean the specific system and tools you are working with right now.

MAC layer The Medium Access Control layer is one of two layers that make up the Data Link Layer of the *OSI Reference Model*. The MAC layer is responsible for moving data packets to and from one Network Intercafe Card (NIC) to another across a shared channel.

machine Some logic analyzers allow you to set up two measurements at the same time. Each measurement is handled by a different machine. This is represented in the Workspace window by two icons, differentiated by a *1* and a *2* in the upper right-hand corner of the icon. Logic analyzer resources such as pods and trigger terms cannot be shared by the machines.

markers Markers are the green and yellow lines in the display that are labeled *x*, *o*, *G1*, and *G2*. Use them to measure time intervals or sample intervals. Markers are assigned to patterns in order to find patterns or track sequences of states in the data. The x and o markers are local to the immediate display, while G1 and G2 are global between time correlated displays.

master card In a module, the master card controls the data acquisition or output. The logic analysis system references the module by the slot in which the master card is plugged. For example, a 5-card Agilent Technologies 16555D would be referred to as *Slot C: machine* because the master card is in slot C of the mainframe. The other cards of the module are called *expansion cards*.

menu bar The menu bar is located at the top of all windows. Use it to select *File* operations, tool or system *Options*, and tool or system level *Help*.

message bar The message bar displays mouse button functions for the window area or field directly beneath the mouse cursor. Use the mouse and message bar together to prompt yourself to functions and shortcuts.

module An instrument that uses a single timebase in its operation. Modules can have from one to five cards functioning as a single instrument. When a module has more than one card, system window will show the instrument icon in the slot of the *master card*.

OSI Reference Model The Open System Interconnection Reference Model is an ISO standard for worldwide communications that defines a networking framework for implementing protocols in seven layers. Control is passed from one layer to the next, starting at the application layer in one station, proceeding to the bottom (physical) layer, over the channel to the next station, and back up the hierarchy. Logic analyzers typically capture data at the *physical layer* or *MAC layer*.

packet A piece of a message transmitted over a packet-switching network, switch fabric, or multiplexed with other packets (like in an MPEG-2 transport stream). A packet has a *header* which identifies the packet and a *payload* which contains the actual data. Packets are also sometimes called *cells*.

packetized data Data that has been broken down into smaller pieces for transmission over a packet-switching network or switch fabric, or for multiplexing with other data streams (like in an MPEG-2 transport stream).

panning The action of moving the waveform along the timebase by varying the delay value in the Delay field. This action allows you to control the portion of acquisition memory that will be displayed on the screen.

pattern terms Logic analyzer resources that represent single states to be found on labeled sets of bits; for example, an address on the address bus or a status on the status lines.

period (.) See *edge terms*, *glitch*, *labels*, and *don't care*.

physical layer The first layer of the *OSI Reference Model* which manages placing data on and taking data off the transmission medium. In reference to protocol definitions, physical layer describes a *protocol* that is used at the bottom of the *protocol stack* on a data bus.

pod pair A group of two pods containing 16 channels each, used to physically connect data and clock signals from the unit under test to the analyzer. Pods are assigned by pairs in the analyzer interface. The number of pod pairs avalaible is determined by the channel width of the instrument.

pod See *pod pair*

point To point to an item, move the mouse cursor over the item, or position your finger over the item.

preprocessor See *analysis probe*.

primary branch The primary

branch is indicated in the *Trigger sequence step* dialog box as either the *Then find* or *Trigger on* selection. The destination of the primary branch is always the next state in the sequence, except for the Agilent Technologies 16517A. The primary branch has an optional occurrence count field that can be used to count a number of occurrences of the branch condition. See also *secondary branch*.

probe A device to connect the various instruments of the logic analysis system to the target system. There are many types of probes and the one you should use depends on the instrument and your data requirements. As a verb, "to probe" means to attach a probe to the target system.

protocol stack A set of protocol layers that work together. The *OSI Reference Model* that defines seven protocol layers is often called a stack, as is the set of TCP/IP protocols that define communications over the internet.

protocol An agreed-upon format for transmitting data between two devices. The protocol determines: the type of error checking, data compression, encoding, how sending devices indicate they have finished sending a message, and how receiving devices indicate they have received a messaage.

range terms Logic analyzer resources that represent ranges of values to be found on labeled sets of bits. For example, range terms could identify a range of addresses to be found on the address bus or a range of data values to be found on the data bus. In the trigger sequence, range terms are considered to be true when any value within the range occurs.

relative Denotes time period or count of states between the current state and the previous state.

remote display A remote display is a display other than the one connected to the product hardware. Remote displays must be identified to the network through an address location.

remote session A remote session is when you run the logic analyzer using a display that is located away from the product hardware.

right-click When using a mouse for a pointing device, to right-click an item, position the cursor over the item, and then quickly press and release the *right mouse button*.

sample A data sample is a portion of a *data set*, sometimes just one point. When an instrument samples the target system, it is taking a single measurement as part of its data acquisition cycle.

Sampling Use the selections under the logic analyzer Sampling tab to tell the logic analyzer how you want to make measurements, such as State vs. Timing.

secondary branch The secondary branch is indicated in the *Trigger sequence step* dialog box as the *Else on* selection. The destination of the secondary branch can be specified as any other active sequence state. See also *primary branch*.

session A session begins when you start a *local session* or *remote session* from the session manager, and ends when you select *Exit* from the main window. Exiting a session returns all tools to their initial configurations.

skew Skew is the difference in channel delays between measurement channels. Typically, skew between modules is caused by differences in designs of measurement channels, and differences in characteristics of the electronic components within those channels. You should adjust measurement modules to eliminate as much skew as possible so that it does not affect the accuracy of your measurements.

state measurement In a state measurement, the logic analyzer is clocked by a signal from the system under test. Each time the clock signal becomes valid, the analyzer samples data from the system under test. Since the analyzer is clocked by the system, state measurements are *synchronous* with the test system.

store qualification Store qualification is only available in a *state measurement*, not *timing measurements*. Store qualification allows you to specify the type of information (all samples, no samples, or selected states) to be stored in memory. Use store qualification to prevent memory from being filled with unwanted activity such as no-ops or wait-loops. To set up store qualification, use the *While storing* field in a logic analyzer trigger sequence dialog.

target system The system under test, which contains the microprocessor you are probing.

terms Terms are variables that can be used in trigger sequences. A term can be a single value on a label or set of labels, any value within a range of values on a label or set of labels, or a glitch or edge transition on bits within a label or set of labels.

time-correlated Time correlated measurements are measurements involving more than one instrument in which all instruments have a common time or trigger reference.

timer terms Logic analyzer resources that are used to measure the time the trigger sequence remains within one sequence step, or a set of sequence steps. Timers can be used to detect when a condition lasts too long or not long enough. They can be used to measure pulse duration, or duration of a wait loop. A single timer term can be used to delay trigger until a period of time after detection of a significant event.

timing measurement In a timing measurement, the logic analyzer samples data at regular intervals according to a clock signal internal to the timing analyzer. Since the analyzer is clocked by a signal that is not related to the system under test, timing measurements capture traces of electrical activity over time. These measurements are *asynchronous* with the test system.

tool icon Tool icons that appear in the workspace are representations of the hardware and software tools selected from the toolbox. If they are placed directly over a current measurement, the tools automatically connect to that measurement. If they are placed on an open area of the main window, you must connect them to a measurement using the mouse.

toolbox The Toolbox is located on the left side of the main window. It is used to display the available hardware and software tools. As you add new tools to your system, their icons will appear in the Toolbox.

tools A tool is a stand-alone piece of functionality. A tool can be an instrument that acquires data, a display for viewing data, or a post-processing analysis helper. Tools are represented as icons in the main window of the interface.

trace See *acquisition*.

trigger sequence A trigger sequence is a sequence of events that you specify. The logic analyzer compares this sequence with the samples it is collecting to determine when to *trigger*.

trigger specification A trigger specification is a set of conditions that must be true before the instrument triggers.

trigger Trigger is an event that occurs immediately after the instrument recognizes a match between the incoming data and the trigger specification. Once trigger occurs, the instrument completes its *acquisition*, including any store qualification that may be specified.

workspace The workspace is the large area under the message bar and to the right of the toolbox. The workspace is where you place the different instrument, display, and analysis tools. Once in the workspace, the tool icons graphically represent a complete picture of the measurements.

zooming In the oscilloscope or timing analyzer, to expand and contract the waveform along the time base by varying the value in the s/Div field. This action allows you to select specific portions of a particular waveform in acquisition memory that will be displayed on the screen. You can view any portion of the waveform record in acquisition memory.

Index