

LeCroy USB*Tracer/Trainer*TM

USB 2.0 Design & Verification System

User Manual

Manual Version 2.15



For Software Version 2.15

15 November, 2005

Document Disclaimer

The information in this document has been carefully checked and is believed to be reliable. However, no responsibility can be assumed for inaccuracies that may not have been detected. CATC reserves the right to revise the information in this document without notice or penalty.

Changes or Modifications

Any change or modification not expressly approved by CATC voids the user's authority to operate this equipment.

Trademarks and Servicemarks

CATC, Advisor, USBTracer, USBTrainer, USBMobile, USBMobileHS, and BusEngine are trademarks of Computer Access Technology Corporation.

Microsoft, Windows 98SE, Windows Me, Windows 2000, Windows XP, and Windows NT are registered trademarks of Microsoft Inc.

All other trademarks are property of their respective companies.

Copyright

Copyright © 2004, Computer Access Technology Corporation (CATC); All Rights Reserved. This document may be printed and reproduced without additional permission, but all copies should contain this copyright notice.

FCC Conformance Statement

This equipment has been tested and found to comply with the limits for a Class A digital device, pursuant to Part 15 of the FCC Rules. These limits are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the instruction manual, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense. The end user of this product should be aware that any changes or modifications made to this equipment without the approval of CATC could result in the product not meeting the Class A limits, in which case the FCC could void the user's authority to operate the equipment.

EU Conformance Statement

This equipment complies with the EMC Directive 89/336/EEC and the Low Voltage Directive 73/23/EEC, and their associated amendments for Class A Information Technology Equipment. It has been tested and found to comply with EN55022:1998 Class A (EN61000-3-2:1998, EN61000-3-3:1995) and EN55024:1998 (EN61000-4-2:1998, EN61000-4-3:1996, EN61000-4-4:1996, EN61000-4-5:1996, EN61000-4-6:1997, EN61000-4-11:1998), and EN60950:1999. In a domestic environment this product may cause radio interference in which case the user may be required to take adequate measures.

Part number: 730-0020-00

TABLE OF CONTENTS

Chapter 1 Overview.....	1
General Description	1
Features	3
Traffic Generation	5
Hi-Speed Slow Clock	5
Graphical Bus Traffic Display	6
Accurate Time Management	6
Comprehensive Error Detection and Analysis	7
Real-Time Event Triggering and Capture Filtering	7
Advanced Event Counting & Sequencing	8
BusEngine Technology	8
Specifications	9
Package	9
Connectors	9
Weight	9
Power Requirements	9
Environmental Conditions	9
Probing Characteristics	9
Switches	10
Indicators (LEDs)	10
UPAS	10
Recording Memory Size	10
Chapter 2 Quick Installation	13
Installing the Software and Starting the Program	14
Your First USB Recording	14
Capturing Your First Trace	16
Trace View Features	16
Chapter 3 Detailed Installation	19
System Components/Packing List	19
The Installed USBTracer Unit	19
LED and Button Descriptions	19
USBTracer System Setup	21
AC Power Source	21
External Interface Breakout Board	21
Pin-Outs for the Data In/Out Connector	22
Prototype Rework Area	23
PC Connection	23
Analyzer PC Requirements	23
Setting Up the Analyzer	23
Installing the USBTracer Application Program	24

Installing the Analyzer Software on the PC	25
USBTracer Program Startup	25
Making a Recording	26
Chapter 4 Updates	27
Software, Firmware, and BusEngine Revisions	27
Software Updates	28
BusEngine and Firmware Updates	28
Updating the BusEngine	28
Updating the Firmware	29
Automatic Updates	29
Manual Updates to BusEngine and Firmware	31
Manually Updating the Firmware	32
Resetting the Analyzer	33
Chapter 5 Software Overview	35
Starting the USBTracer Program	35
Tool Tips	35
The Main Display Windows	36
View Options	38
Resetting the Tool bar	39
Tool Bar	40
Status Bar	41
Recording Progress	41
Recording Status	42
Recording Activity	43
Search Status	44
Navigation Tools	44
Zoom In	44
Zoom Out	44
Wrap	44
USBTracer Analyzer Keyboard Shortcuts	44
Chapter 6 Recording Options	47
General Recording Options	48
Recording Type	48
Options	48
Buffer Size	49
Trigger Position	50
Options Name	50
Trace File Name & Path	51
Recording Options - Misc	51
Analyzer Speed	52
Generator/Analyzer Clocking Overrides	53

USB On-The-Go	53
USBTrainer Generator-related Parameters	53
Slow Clock Selection	55
Events Recording Options	55
Packet Identifiers	56
Token Patterns	57
Frame Patterns	57
Device Requests	58
Data Pattern	58
Bus Conditions	60
Errors	60
External Input Signals	61
Transactions	62
Data Length	62
Splits	64
Extern Data7 - Data0 In	64
Actions Recording Options	65
Actions Window Layout	65
Connecting Events to Actions	66
Connecting Counters to Events	66
Using Action Buttons	69
Trigger	69
Restart	69
Filter Out/In	69
Other Actions: External Output Signals	70
Saving Recording Options	71
Recording Bus Data	72
Merging Trace Files	73
Chapter 7 Display Options	75
General Display Options	76
Color/Format/Hiding Display Options	79
Color Display Options	80
Formats Display Options	81
Hiding Display Options	82
Level Hiding Display Options	83
Saving Display Options	83
Chapter 8 Reading a Trace	85
Trace View Features	85
Set Marker	86
Edit or Clear Marker	86
View Raw Bits	87
Expanding and Collapsing Data Fields	88

Using the Expand/Collapse Data Field Arrows	88
Double-Clicking to Expand/Collapse Data Fields	88
Expanding or Collapsing All Data Fields	88
Using the Data Field Pop-up Menus	89
View Data Block	90
Pop-up Tool-tips	91
Hide SOF Packets	91
Hide NAKs	91
Hide Devices	91
Hide Chirps	91
Switch to Transactions View	91
View Decoded Transactions	93
Expanded and Collapsed Transactions	94
Switch to Split Transaction View	95
Switch to Transfer View	96
View Decoded Transfers	98
Expanded and Collapsed Transfers	98
Decoding Protocol-Specific Fields in Transactions/Transfers	99
Using the Trace Navigator	100
Navigator Bar Attributes	102
Chapter 9 Decode Requests	103
General Options	103
USB Request	103
Decoding Standard Requests	104
Decoding Class Requests	105
Decoding Vendor Requests	105
Decoding USB Requests	106
Class and Vendor Definition Files	106
Class/Vendor Decoding Options	108
Request Recipient to Class/Vendor Decoding	108
Class/Vendor Endpoint Decoding	110
Load most recent Decoding Mapping	112
Refresh Decoding for this Trace File	113
Chapter 10 Other Features	115
Search	115
Go to Trigger	115
Go to Packet/Transaction/Transfer	115
Go to Marker	116
Go To	116
Find	122
Find Next	123
Search Direction	123

Edit Comment	123
Reports	124
File Information	124
Error Summary	125
Timing Calculations	125
Traffic Summary	127
Bus Utilization	128
Real Time Monitoring	131
Chapter 11 Traffic Generation with USBTrainer	135
USBTrainer Traffic Generation Files	135
Creating Traffic Generation Files	136
Editing a Generation File	138
Toolbar	138
File Tabs	140
Error Log	140
Tooltips	140
Loading the Generation File	140
Traffic Generation Modes: Bitstream vs. Intelliframe . .	142
Starting Traffic Generation	142
Repeating a Generation Session	142
Stop Traffic Generation	143
Device Emulation	143
Creating a Generation File	143
Setting Generation Options	143
Run the Traffic Generation Script File	144
Format of the Traffic Generation Files	145

1. Overview

The LeCroy USBTracer USB 2.0 Design & Verification System is the latest in LeCroy's broad portfolio of analysis tools. As a fifth generation product, it builds upon solid knowledge of the needs of the USB development and test communities. The result is a USB bus & protocol analyzer with unprecedented functionality, unparalleled flexibility, and uncompromising user friendliness. The USBTracer interfaces with standard USB cables and connections to capture and display all speeds of USB bus traffic.



1.1 General Description

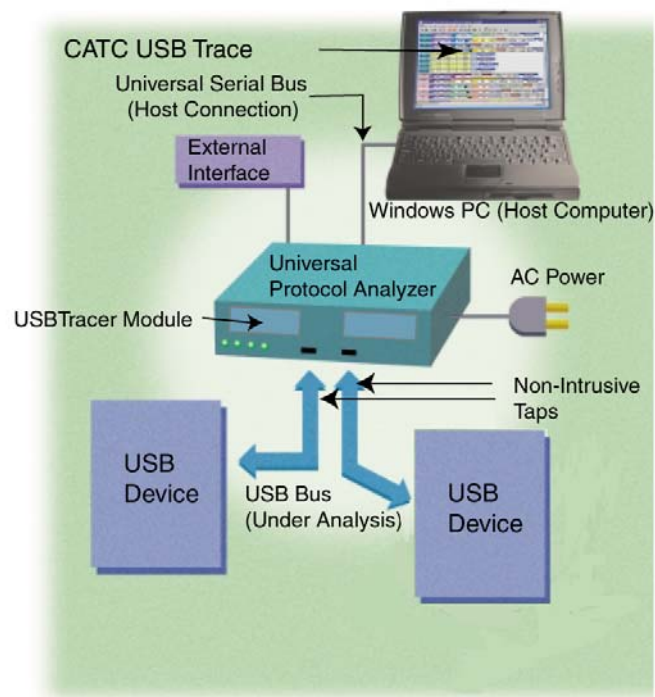
USBTracer™ is a hardware module that installs into the LeCroy Universal Protocol Analyzer System. Once installed, USBTracer can be easily configured and controlled by a portable or desktop PC connected via its USB port. USBTracer provides customers with the familiar "CATC Trace" user interface that is the *de facto* industry standard for documenting the performance of high-speed serial protocols.

USBTracer is a non-intrusive testing tool for the USB protocol providing traffic capture and analysis. Hardware triggering allows real-time events to be captured. Hardware filtering allows the different types of packets to be filtered in or out of the recording. Filtering allows users to focus recordings on events of interest and to preserve recording memory so that the recording time can be extended.

Recorded data is displayed in colored graphics in a trace viewer application. This application has advanced search and viewing capabilities that allow you to quickly locate specific data, errors and other desired conditions, thereby focussing the user's attention on events of interest.

USBTracer functions with any personal computer using the Windows 98SE, Window 2000, Windows ME, or Windows XP operating systems and equipped with a functional USB interface.

The Analyzer is a plug-in module that is installed into a LeCroy Universal Protocol Analyzer System (UPAS). Together, the Analyzer and UPAS are controlled from a personal computer USB port across a USB connection.



USBTracer provides on-the-fly detection of and triggering on such events as Tokens and Errors. Whether recording manually or with a specified trigger condition, USBTracer continuously records the link data in a wrap-around fashion until manually stopped or until the Trigger Event is detected and a specified post-Trigger amount of link data is recorded.

Upon detection of a triggering event, the analyzer continues to record data up to a point specified by the user. Real-time detection of events can be individually enabled or disabled to allow triggering on events as they happen. This includes predefined exception or error conditions and a

user-defined set of trigger events. The unit can also be triggered by an externally supplied signal. An external DB-25 connector provides a path for externally supplied data or timing data to be recorded along with traffic.

This DB-25 connector also provides a path for USBTracer to transmit externally the software trigger signal or a user-defined event for purposes of probing and use by other circuitry.

The USBTracer software provides powerful search functions that enable investigation of particular events and allow the software to identify and highlight specific events. In addition to immediate analysis, you can print any part of the data. Use the **Save As** feature to save the data on disk for later viewing. The program also provides a variety of timing information and data analysis reports.

Please refer to the *Universal Serial Bus Specification, version 2.0* for details on the protocol. The USB specification is available from the USB Implementers Forum (USB-IF) at

USB Implementers Forum
1730 SW Skyline Blvd.
Suite 203
Portland, OR 97221

Tel: +1/503.296.9892
Fax: +1/503.297.1090
Web: <http://www.usb.org/>

1.2 Features

General

- Fully compliant with USB specification revision 2.0.
- Flexible design - reconfigurable hardware for future enhancements.
- Convenient modular system architecture with field upgradeable firmware and recording engine.
- Supports all USB speeds (480 Mb/s, 12Mb/s and 1.5 Mb/s).
- On the UPAS 2500H platform, Hi-Speed recording and generation requires an optional, separately priced key. See your LeCroy representative for more information.
- Dual recording channels aid in the development of multiple speed functions upstream and downstream of speed-matching hub or transaction translator.
- User friendly "CATC Trace" interface that displays bus traffic using color and graphics.
- Trace Viewer available as free non-recording, view-only software.
- Online and context-sensitive help.

- Power-on self-diagnostics.
- Three year warranty and hot-line customer support.
- Can be used in conjunction with the LeCroy USBTrainer USB Traffic Generator hardware module to create a fully customizable USB test platform.

Physical Components

- Hardware module for the LeCroy Universal Protocol Analyzer System.
- Operated by any desktop or portable Microsoft Windows 98, Windows 98SE, Windows 2000, Windows ME, or Windows XP computer with USB capability.
- Convenient Plug-and-Play USB installation.
- High impedance tap inserts non-intrusively in any branch of a USB system.
- 512 MB of physical data recording memory.
- Two all speed recording channels, which include Low, Full, or Hi-Speed.
- Full-Speed USB connection to desktop or portable host PC. Hi-Speed on UPAS 2500H platform.
- Internal wide-range AC power supply.
- Break-out board (included) - used to interface to external test equipment.

Recording Options

- Versatile triggering - bit-wise value and mask data patterns up to sixteen bytes wide for Setup transactions and data packets.
- Triggering on new High-speed PIDs and split transaction special tokens (ERR, SPLIT, PING, NYET, DATA2, and MDATA).
- CATC Trace displays and enumerates High-speed Micro Frames.
- Three forms of triggering: Snapshot, Manual and Event.
- Transaction sequencer allows triggering on a token qualified by a data pattern and /or specific handshake or alternately transactions can be filtered (i.e., NAK'd transactions).
- Advanced triggering with event counting and sequencing.
- Dedicated trigger for recording input and output used to interface to external test equipment.
- Triggering on multiple error conditions - PID bad, bit stuffing bad, CRC bad, end-of-packet bad, babble, loss of activity, frame length violation, time-out or turn-around violation, data toggle violation, Token, Bus Conditions, Data Length, excessive empty frames.

- Real-time traffic capture filtering and data packet truncation variable up to 256 bytes.
- Adjustable buffer size from 0.4 MB to 512 MB.

Display Options

- Utilizes "CATC Trace" graphical display of bus packets, transactions, split transactions and transfers.
- Numerous packets and transactions can be grouped under a single transfer while quickly decoding all essential information.
- Decoding of split transactions up and down stream of a transaction translator is accomplished with a special hierarchical view.
- Variety of reports provided to summarize key statistics and conditions of interest with the ability to jump to the selected item in the trace display.
- Flexible input signaling can be recorded with the CATC Trace.
- Trace Viewer is backward compatible with Advisor, Chief, Inspector and Detective™ trace files.
- User-friendly trigger position indicated by different colors of pre- and post-trigger packet color.
- Markers can be set to assist with navigation and time calculations.
- Hide start-of-frame (SOF) packets as well as any packet or transaction.
- Search for a specific PID.
- Detects & alerts the user to every potential bus error, protocol violation, & combinations thereof.
- High resolution, accurate time stamping of bus packets, timing measurement and analysis functions.
- Extensive search and packet hiding capabilities.
- Comprehensive device class decoding plus user defined protocol decoding.

Traffic Generation

- Traffic generation capability available as an add-on module for the Universal Protocol Analyzer System called USBTrainer.

Hi-Speed Slow Clock

The ability to Trace and Generate Hi-Speed traffic at fractional (slow) clock rate is available on both USBTracer platforms (2500 and 2500H), and can be purchased via a License Key. Contact LeCroy for more information.

1.3 Graphical Bus Traffic Display

The USBTracer transaction displays use color and graphics to show the captured transactions in an immediate, understandable and useful format.

Packets are shown on separate rows, with their individual fields both labeled and color-coded. Packets are also numbered (sequentially, as recorded), time-stamped. Protocol errors are automatically detected and visibly highlighted in red so they will not be overlooked.

The display is customizable, allowing the user to control the color scheme and the formatting of field contents. A hide feature allows users to enable the suppression of SOF packets and user defined packets or fields that might be uninteresting in a given context. Display formats can be named and saved for later use. Pop-up "tool-tips" annotate packet fields with even more detailed information about their contents.

The display software operates independently of the hardware, allowing it to function as a stand-alone "trace viewer" that is freely distributed.

The screenshot shows the CATC USBTracer Bus And Protocol Analyzer software interface. The window title is "CATC USBTracer Bus And Protocol Analyzer - [data_1_Sample.usb - Range from Packet 860 to Packet 3789]". The menu bar includes File, Setup, Record, Generate, Report, Search, View, Window, and Help. The toolbar contains various icons for file operations, recording, and analysis. Below the toolbar, there are buttons for "Pkt", "Trs", "Trs", and "Xfr", along with a "Run once" button. The main display area shows a list of USB packets, each with a row of color-coded fields. The fields are: Packet, Dir, F, Sync, NAK, EOP, Idle, and Time Stamp. The packets are numbered 860 through 866. The fields are color-coded: Sync is green, NAK is black, EOP is blue, Idle is purple, and Time Stamp is blue. The fields are also labeled with their respective values: Packet (860-866), Dir (<-- or -->), F (S), Sync (00000001), NAK (0x5A), EOP (250 ns), Idle (9.367 μs, 9.533 μs, 29.450 μs, 9.383 μs), and Time Stamp (00003.3734 3985, 00003.3734 4, 00003.3734 4835, 00003.3734 5, 00003.3734 5695, 00003.3735 0, 00003.3735 0249). The status bar at the bottom shows "Ready" and "Search: Fwd".

Packet	Dir	F	Sync	NAK	EOP	Idle	Time Stamp
860	<--	S	00000001	0x5A	250 ns	9.367 μs	00003.3734 3985
861	-->	S	00000001	0x96	2	0	0x15 233 ns 316 ns 00003.3734 4
862	<--	S	00000001	0x5A	250 ns	9.533 μs	00003.3734 4835
863	-->	S	00000001	0x96	2	0	0x15 233 ns 316 ns 00003.3734 5
864	<--	S	00000001	0x5A	250 ns	29.450 μs	00003.3734 5695
865	-->	S	00000001	0x96	2	0	0x15 233 ns 300 ns 00003.3735 0
866	<--	S	00000001	0x5A	250 ns	9.383 μs	00003.3735 0249

1.4 Accurate Time Management

USBTracer uses internal counter/timer circuitry to enable a reliable, accurate (16.67 ns resolution) time stamping of recorded bus traffic. This timing information is displayed in the trace and in measurement and

analysis functions. Any number of markers can be added by the user to denote specific packets, and further timing measurements can be made from one marker to another or from marker to trigger. All time fields are presented as a time stamp, idle time, bit time or in a new time delta in either decimal or hexadecimal format.

An essential feature of time management is that idle traffic will not consume the analyzer memory. Because of this unique technology, accurate timing calculations can be made while still preserving valuable recording memory for important bus traffic.

1.5 Comprehensive Error Detection and Analysis

USBTracer detects and alerts the user to every potential bus error, protocol violation and/or combination thereof. The analyzer contains circuitry in the BusEngine to perform real-time triggering on multiple error conditions, such as PID bad, bit stuffing bad, header or data CRC bad, end-of-packet bad, babble, loss of activity, frame length violation, time-out or turn-around violation and data toggle violation. The analyzer program highlights all the hardware detected errors and further examines the trace file for additional protocol errors, including wrong packet length, data payload violation, and packet termination not on a byte boundary.

1.6 Real-Time Event Triggering and Capture Filtering

No attribute of a bus and protocol analyzer is more important than its capacity for extracting useful information from a crowded stream of bus traffic. The ability to accurately identify and selectively record transactions of interest is the trait most valuable to the user. USBTracer offers a unique approach to this challenge. By using a set of "recording resources", over a dozen configureable hardware building blocks, each can be optimized to perform a particular activity appropriate to the task. A recording resource can independently await an initialization signal, monitor its external environment (external signals, other resources) in search of a particular event and take a subsequent action (triggering, inclusive or exclusive filtering, counting, etc.). Under control of the user interface, these resources can be selected, configured and combined to search for complex trigger conditions and selectively capture the associated transactions.

Thus, as well as triggering on basic events, such as specific bus conditions, packet identifiers (PID), etc. (see the Specifications section for a complete list), USBTracer can easily manage the most challenging requirements. For example: "trigger on the fifth occurrence of a SETUP Token device number

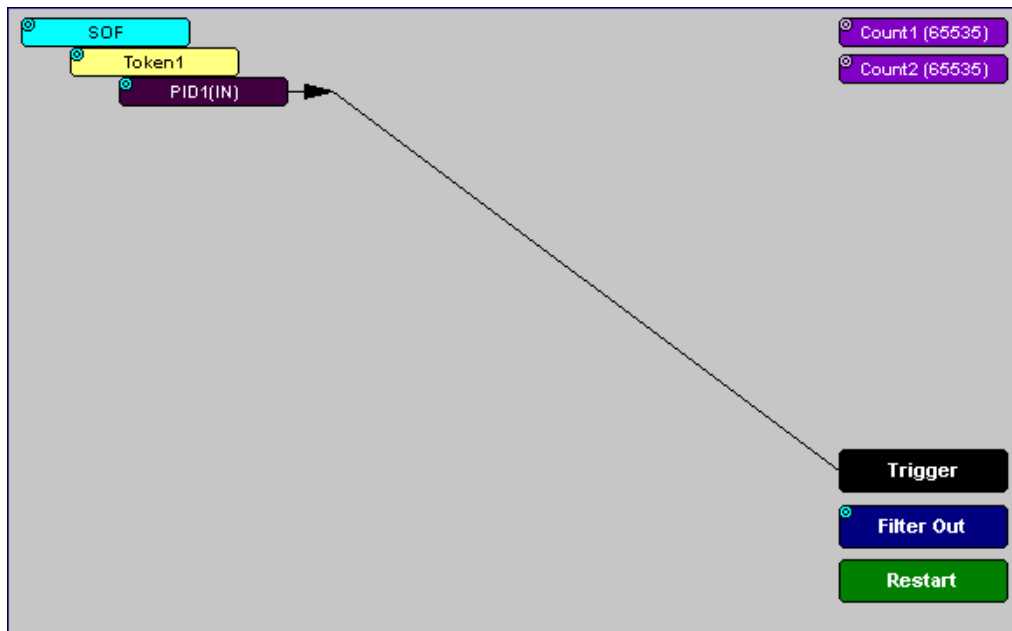
nine", or "trigger on a SET INTERFACE request, following a specified eight-byte bulk data pattern match from this scanner, and do not capture any start-of-frame (SOF) packets."

The user can set the size of the recording memory, specify the pre- and post-trigger capture ratio, and truncate large data packets up to 256 bytes.

1.7 Advanced Event Counting & Sequencing

The count and sequence options define rules for data recording sessions. These options provide the ability to configure and control the order of events that have been selected for triggering or filtering. There are two counters and a restart option that causes the sequencing to start again.

By utilizing this feature a designer can specify a specific sequence of up to seven events that must occur before the analyzer triggers and finishes capturing data. This is extremely important in allowing a designer to pinpoint certain types of events for recording. Without this feature a designers must scroll through potentially megabytes of recorded data to locate a specific sequence or occurrence of a sequence that can take hours to locate before being able to begin analyzing the data of interest.



1.8 BusEngine Technology

The LeCroy BusEngine Technology is at the heart of the new USBTracer Analyzer. The BusEngine core uses state-of-the-art Electrically Programmable Logic Device (EPLD) technology and incorporates both the

real-time recording engine and the configurable building blocks that implement data/state/error detection, triggering, capture filtering, external signal monitoring, and event counting & sequencing. Like the flash-memory-based firmware that controls its operation, all BusEngine logic is fully field upgradeable, using configuration files.

1.9 Specifications

USBTracer is a hardware module that installs into the Universal Protocol Analyzer System. The following specifications describe a combined USBTracer/Universal Protocol Analyzer System.

Package

UPAS	12.2 x 12.2 x 3.5 inches (31.1 x 31.1 x 8.9 cm)
------	--

USBTracer Plug-in	4.5 x 6.7 x 1.3 inches (11.3 x 17.0 x 3.2 cm)
-------------------	--

Connectors

UPAS	AC power connection External trigger connection Host connection (USB, type "B")
------	---

Weight

UPAS	7.5 lb. (3.4 kg)
USBTracer Plug-in	1.0 lb. (0.5 kg)

Power Requirements

90-254VAC, 47-63Hz (universal input), 100W maximum

Environmental Conditions

Operating Range:	0 to 55 °C (32 to 131 °F)
Storage Range:	-20 to 80 °C (-4 to 176 °F)
Humidity:	10 to 90%, non-condensing

Probing Characteristics

Connection:	High Speed USB Connectors Standard cables
-------------	--

Switches

Power:	On/off
Manual Trigger:	When pressed forces a trigger event

Indicators (LEDs)

UPAS

Power (PWR):	Illuminated when the analyzer is powered on.
Status (STATUS):	Illuminated solid during self test, blinking during failure, off when the analyzer is functioning properly
Recording (REC):	Illuminated when the analyzer is actively recording traffic data
Triggered (TRG):	Illuminate during power-on testing and when the analyzer has detected a valid trigger condition
Uploading (UPLD):	Illuminated when the analyzer is uploading its recording memory to the Host PC for displaying the CATC Trace

Recording Memory Size

512 M x 8-bit DRAM for traffic data capture, timing, state and other data.

Host Compatibility

Works with any PC equipped with a functioning USB port and running Microsoft Windows 98SE, Windows ME, Windows 2000, and Windows XP operating systems.

Required Setup

Microsoft Internet Explorer, Version 5 or newer.

Hardware

Processor: For optimum performance, use processors of Intel's PentiumIII/Pentium4 family, AMD's Athlon/Duron family, or other compatible processors with clock speed of 500mHz or higher (Processors of Intel's Pentium II/Celeron family or AMD's K6 family with clock speed of 300mHz is a minimum).

Memory: For the best performance, it is recommended to have physical RAM twice the size of the recording buffer setup - 512MB or more (minimum of 128MB of RAM).

Hard Disk: At least 20MB of free hard disk space is required for the installation. Additional disk space is needed for storing the recorded data in files during the recordings process (can be as much as 500MB when recording a full buffer size).

Display: Resolution of 1024 x 768 with at least 16-bit color is recommended (resolution of 800 × 600 with 16-bit color is a minimum).

Connectivity: A USB interface is required to connect to the USBTracer analyzer. This is not a requirement if the application is going to be used only as a viewer.

Power Cord Set

Cord Manufacturer and Models:

180-0005-00 Quail 5000.079 UL, CSA

180-0013-00 Quail 8500.098 VDE, KEMA, CEBERC, NEMDO, DEMDO, SETI, OVE, SEV

180-0014-00 Quail 9650.098 ASTA

USBTracers on the UPAS 2500H Platform

UPAS 2500H USBTracer is a USB 2.0 Certified Hi-Speed Device. To upload at Hi-Speed, you must have a USB 2.0 Certified Host Controller and be running Windows 2000 SP3 or Windows XP.

The UPAS 2500H ships in one of two configurations: Classic and Hi-Speed. The Classic version can trace and generate traffic at Low and Full Speeds. The Hi-Speed version can trace and generate traffic at All speeds: Low, Full, and Hi.

The Classic model may be upgraded to a Hi-Speed model by the purchase of a License Key from LeCroy. Contact your LeCroy representative for more information.

Note: The UPAS 2500 and UPAS 2500H products include internal 1.1 and 2.0 Hubs. You might see the drivers for these generic hubs being installed upon the initial plug-in.

2. Quick Installation

USBTracer is factory-installed hardware module that is sold as part of the LeCroy Universal Protocol Analyzer System.

The Universal Protocol Analyzer System (UPAS) together with the USBTracer module and associated software are easily installed on most Windows-based personal computer systems. You can begin making USB recordings after following these initial steps. However, if you are new to protocol analyzers, or are unsure about what to do after reading the Quick Installation instructions, or if your analyzer does not work after you follow these instructions, read through the subsequent sections in this manual.

- Step 1** Connect the AC power cable to the rear of the Analyzer.
- Step 2** Connect the USB port on the back of the Analyzer to the PC using the LONG (6-foot/2-meter) USB cable.
- Step 3** Insert the CD.
- Step 4** Turn on the power switch on the rear of the Analyzer.
- Step 5** Click **Next** when you see the window of "Add New Hardware Wizard."
- Step 6** Follow the on-screen Plug-and-Play instructions under windows.
- Step 7** Click **Finish** when you see the message that says "Windows has finished installing the software that your new hardware requires" and the file "Tracer.inf" has been installed in your PC.
- Step 8** Run the program a:\setup and follow the on-screen instructions.
- Step 9** Select **About USBTracer...** from the Help menu.

If you can see all of the information listed below, you can record a trace:

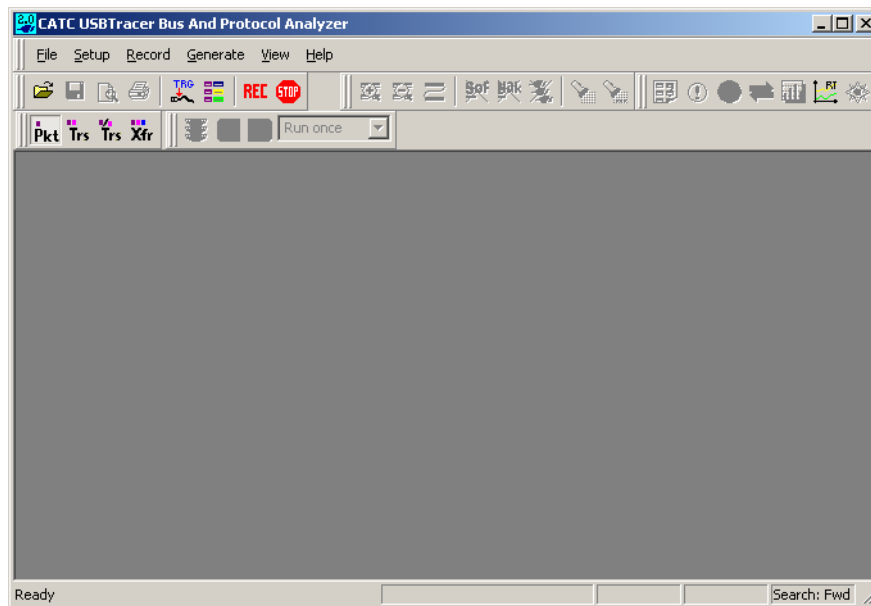
- USBTracer Software Version
- USBTracer Firmware Version
- BusEngine Version
- Unit Serial Number

2.1 Installing the Software and Starting the Program

- Step 1** Insert the USBTracer CD into the CD-ROM drive on the host PC and follow the on-screen instructions to install the *USBTracer* application.
- Step 2** To start the application, launch the **LeCroy USBTracer** program from the **Start Menu**:

Start>Programs>LeCroy>USBTracer

The USBTracer main window will then open.

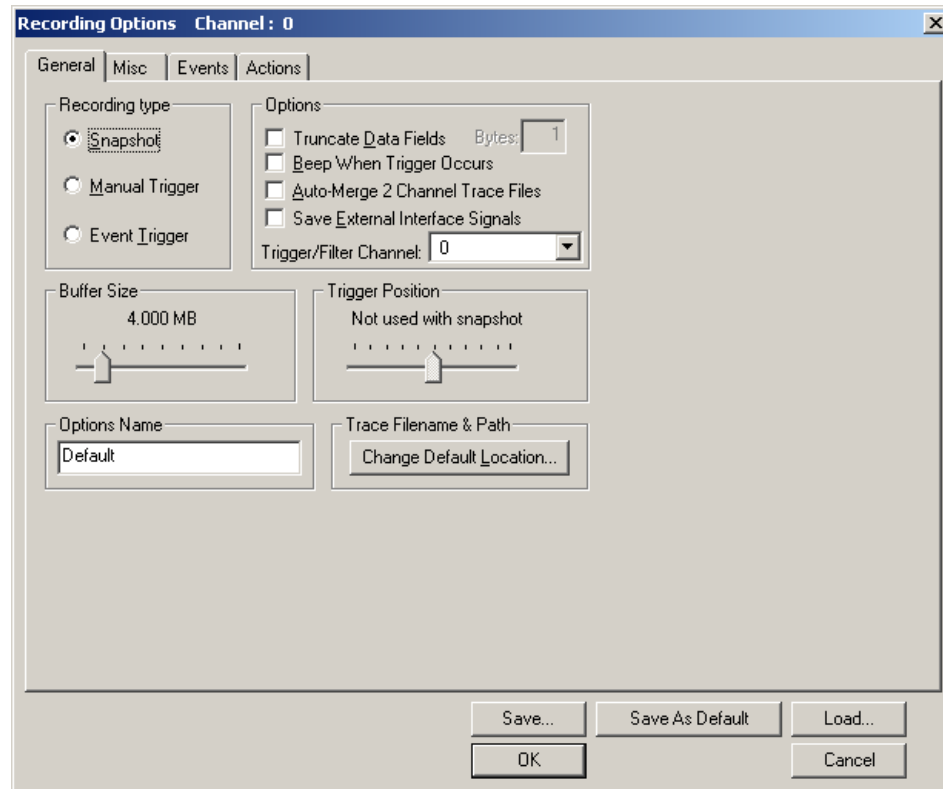


2.2 Your First USB Recording

After installing and launching the software, you can test *USBTracer* by performing the following steps.

- Step 1** Connect a USB cable to each of the two connectors on the *USBTracer* module, then connect the other ends to the USB device under test and USB host system.
- Step 2** Select **Recording Options** under **Setup** on the Menu Bar.
- Step 3** Select the **General** tab.

The following dialog box will open showing factory default settings such as “Snapshot” and 4 Mbytes buffer size. For your first recording, you can leave these settings unchanged.




Step 4 Click OK to activate the recording options you selected.

Step 5 Turn on the USB devices that are to be tested and cause them to generate USB traffic.

Step 6 Click  on the Tool Bar.


USBTracer starts to record the USB traffic immediately. After 4 Mbytes of traffic is recorded, the analyzer will upload the data and display the packets in the trace window.

Step 7 If you wish to terminate the recording before the snapshot automatically completes, click  on the Tool Bar.

When the recording session is finished, the traffic is uploaded from the Analyzer to the hard drive on your PC as a file named **data.usb** or whatever name you assigned as the default filename. While the file is being uploaded, you should see a brown progress bar at the bottom of the screen. When the bar disappears, it indicates that the data has been uploaded to disk.

Step 8 To save a current recording for future reference, select **Save As** under **File** on the Menu Bar.

OR

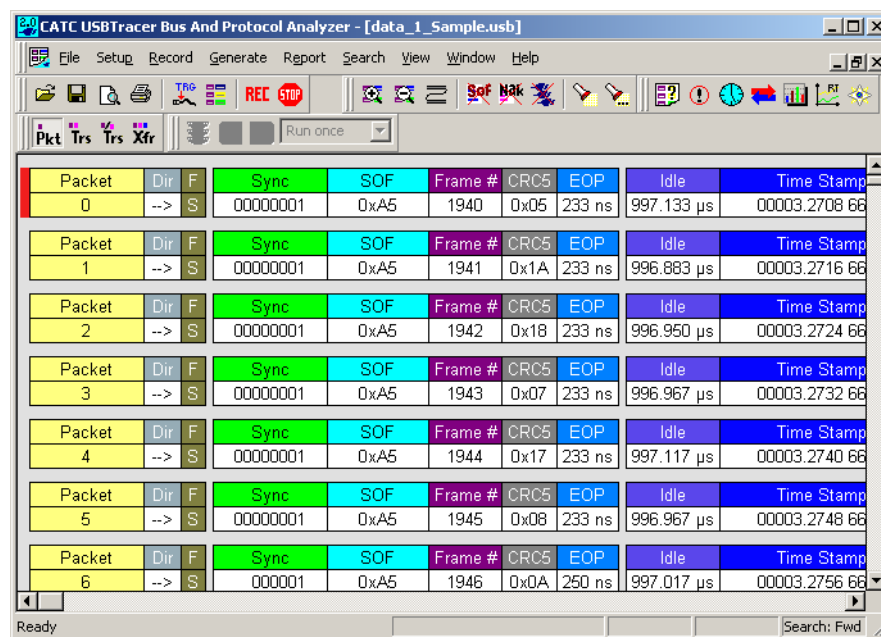
Click  on the Tool Bar.

You see the standard **Save As** screen.

Step 9 Give the recording a unique name and save it to the appropriate directory.

Capturing Your First Trace

After a moment, the recording will terminate and the results will display.



The screenshot shows the CATC USBTracer Bus And Protocol Analyzer window. The main display is a table of captured packets. The table has columns for Packet number, Direction, Frame type, Sync, SOF, Frame #, CRC5, EOP, Idle, and Time Stamp. The packets are numbered 0 through 6. The first six packets are all 'Sync' frames with a direction of '-->' and a frame type of 'S'. The seventh packet is an 'Idle' frame with a direction of '-->' and a frame type of 'S'. The table is color-coded: Sync frames have a green background, SOF frames have a cyan background, and Idle frames have a blue background. The status bar at the bottom shows 'Ready' and 'Search: Fwd'.

Packet	Dir	F	Sync	SOF	Frame #	CRC5	EOP	Idle	Time Stamp
0	-->	S	00000001	0xA5	1940	0x05	233 ns	997.133 μs	00003.2708 66
1	-->	S	00000001	0xA5	1941	0x1A	233 ns	996.883 μs	00003.2716 66
2	-->	S	00000001	0xA5	1942	0x18	233 ns	996.950 μs	00003.2724 66
3	-->	S	00000001	0xA5	1943	0x07	233 ns	996.967 μs	00003.2732 66
4	-->	S	00000001	0xA5	1944	0x17	233 ns	997.117 μs	00003.2740 66
5	-->	S	00000001	0xA5	1945	0x08	233 ns	996.967 μs	00003.2748 66
6	-->	S	00000001	0xA5	1946	0x0A	250 ns	997.017 μs	00003.2756 66

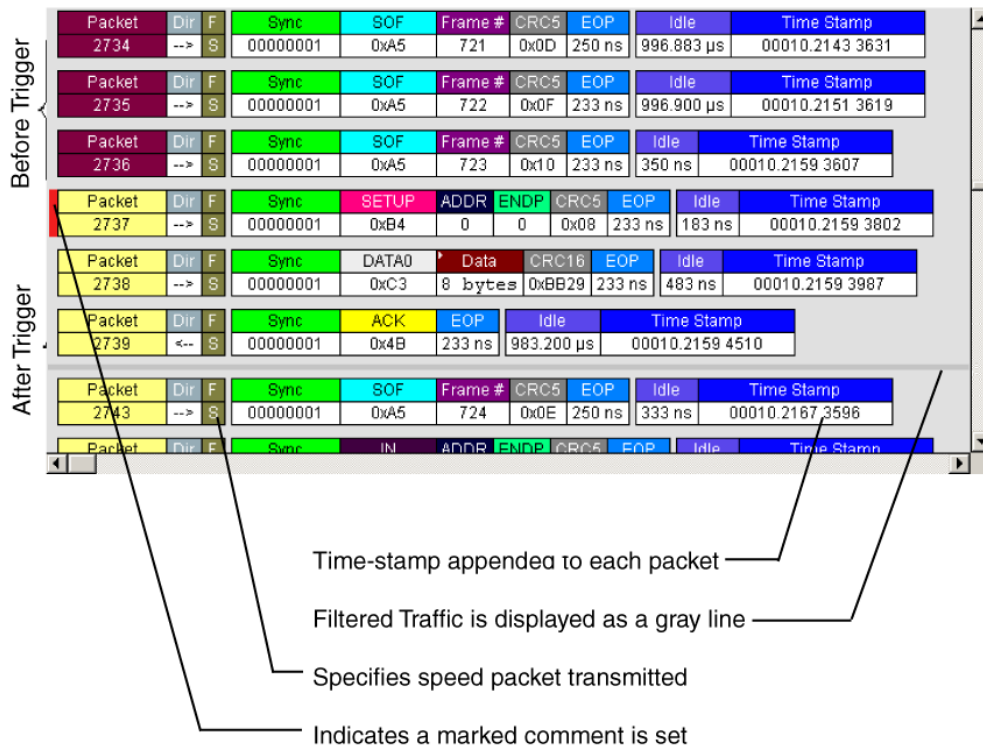
Trace View Features

- The USBTracer packet view display makes extensive use of color and graphics to fully document the captured traffic.
- Packets are shown on separate rows, with their individual fields both labeled and color coded.
- Packets are numbered (sequentially, as recorded), time-stamped (with a resolution of 16.67 ns), and highlighted to show the transmitted speed (low-speed, full-speed or high-speed).
- Display formats can be named and saved for later use.

- Pop-up Tool Tips annotate packet fields with detailed information about their contents
- Data fields can be collapsed to occupy minimal space in the display (which can in turn be zoomed in and out to optimize screen utilization).

The display software can operate independent of the hardware and so can function as a stand-alone Trace Viewer that may be freely distributed.

Each row numerates, labels, and color-codes a USB packet



3. Detailed Installation

3.1 System Components/Packing List

- One stand-alone USBTracer Analyzer module
- Five USB cables
- USBTracer software program installation diskettes
- Product documentation

3.2 The Installed USBTracer Unit

USBTracer is a module that inserts into the left slot on the LeCroy Universal Protocol Analyzer System (UPAS). Once installed, the USBTracer Analyzer will activate the user-accessible controls and LEDs on the front and rear panels of the UPAS.

Figure 1: Front Panel



LED and Button Descriptions

The following section describes the LEDs, buttons, and connectors on the module and UPAS.

LEDs on the Far Left Side of the UPAS

- Green **PWR** (power) indicator LED for UPAS (lights when the unit power is switched on).
- Red **Status** indicator LED for UPAS (lights during the boot up)

LEDs and Button on the UPAS under the Left Module Slot

The LEDs and button on the UPAS enclosure under the left module slot function in conjunction with the USBTracer module inserted above it.

- Green **REC** (recording) LED (lights when the unit is recording).
- Orange **TRG** (triggered) LED (lights when the unit triggers on an event).
- Green **UPLD** (Upload) (lights when unit is uploading data to PC)

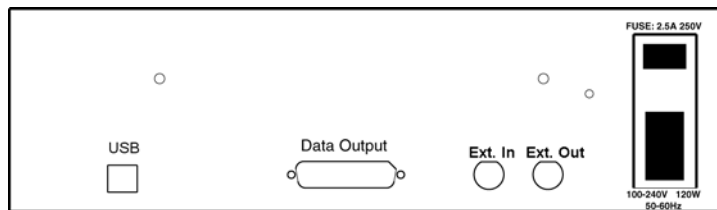
- **Manual Trigger** push-button (allows a manual Trace capture)

LEDs and Button on the UPAS under the Right Module Slot

The LEDs and button under the right module slot are reserved for LeCroy generator modules such as the USBTrainer USB Traffic Generator (described in Chapter 11: **Traffic Generation with USBTrainer** on page 135).

- Green **HS** - Illuminates when Hi-Speed is being generated.
- Orange **Classic** - Illuminates when Full or Low Speed is being generated. Classic also illuminates during power-on testing and is turned off at the end of the power on cycle. If the LED blinks at the end of this cycle, the hardware is faulty.
- Green **Intelliframe** - Illuminates when Intelliframe traffic is being generated using Intelliframe mode, an intelligent adaptive traffic mode. If off, the generation is in Bitstream mode. See the section on "Traffic Generation Modes: Bitstream versus Intelliframe" in Chapter 11 for more information.
- **Start/Stop** - push-button allows manual Trace capture.

Figure 2: Universal Protocol Analyzer Rear Panel



- Wide range AC connector module
 - Power socket
 - Enclosed 5x20 mm 2.0A 250 V fast acting glass fuse

Warning For continued protection against fire, replace fuse only with the type and rating specified above.

- Power on/off switch
- Two External Ports marked "Ext. 1" and Ext. 2."
- USB type "B" host computer connector
- **Data In/Out** DB-25 (25-pin) external interface connector

Warning Do not open the UPAS enclosure. There are no operator serviceable parts inside. Refer servicing to LeCroy.

3.3 USBTracer System Setup

USBTracer is a module that inserts into the Universal Protocol Analyzer System. The UPAS, in turn, is connected and controlled by a desktop or laptop PC via a USB connection. To set up the system hardware,

- Insert the USBTracer module into the UPAS.
- Connect the Analyzer to an AC power source.
- Connect to the analyzing PC via USB.
- Connect USB cable to the device under test.

3.4 AC Power Source

Step 1 Connect the Analyzer box to a 100-volt to 240-volt, 50 Hz to 60 Hz, 120 W power outlet using the provided power cord.

Note The Analyzer is capable of supporting supply voltages between 100-volt and 240-volt, 50 Hz or 60 Hz, thus supporting all known supply voltages around the world.

Step 2 Use the power switch located on the rear panel to turn the analyzer unit on and off.

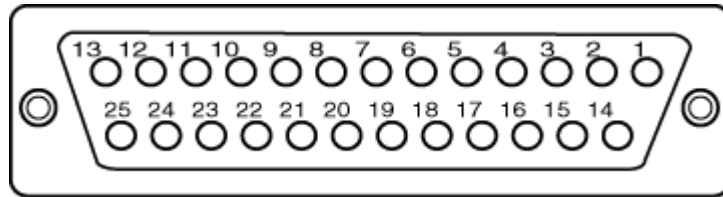
Note At power-on, the analyzer initializes itself in approximately 25 seconds and performs an exhaustive self-diagnostic that lasts about 45 seconds. The Status LED illuminates during the power-on testing and turns off when testing is finished. If the diagnostics fail, the Status LED blinks, indicating a hardware failure. If this occurs, call LeCroy Customer Support for assistance.

3.5 External Interface Breakout Board

The External Interface Breakout Board is an accessory that allows convenient access to several potentially useful standard, LV TTL output and input signals. It also offers a simple way to connect logic analyzers or other tools to the USBTracer Analyzer unit. Six ground pins and one 5-volt pin are provided.

The Breakout Board connects via a cable to the **Data In/Out** connector located on the rear of the analyzer box. Each signaling pin is isolated by a 100 Ω series resistor and a buffer inside the Analyzer box.

Data In/Out Connector (on cable)



Pin-Outs for the Data In/Out Connector

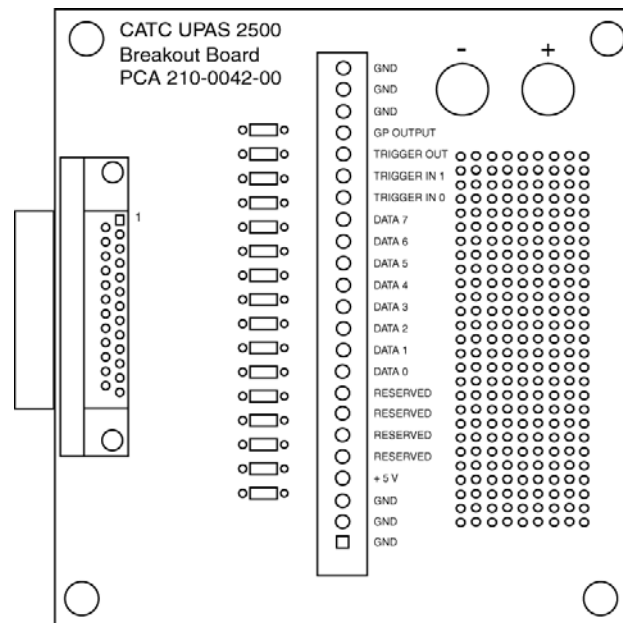
Table lists the pin-out and signal descriptions for the **Data In/Out** connector on a cable that connects to the Breakout board.

Data In/Out Connector – Pin-Out

Pin	Signal Name	Signal Description
1	RSV	Reserved
2	GND	Ground
3	GP OUT	General Purpose Output
4	TRG IN 1	Trigger In 1
5	GND	Ground
6	DATA 6	Data 6
7	DATA 4	Data 4
8	DATA 3	Data 3
9	DATA 1	Data 1
10	GND	Ground
11	RSV	Reserved
12	RSV	Reserved
13	+5V	+5 Volts, 250 mA DC Source
14	RSV	Reserved
15	GND	Ground
16	TRG OUT	Trigger Out
17	TRG IN 0	Trigger In 0
18	DATA 7	Data 7
19	DATA 5	Data 5
20	GND	Ground

Note (*) Pins 4 and 17 have the same function: they allow external signals to be used to cause triggering or recording. Pins 3 and 16 are used to transmit output signals.

External Interface Breakout Board



Prototype Rework Area

The Breakout Board contains a prototype rework area for making custom circuits for rapid development. The area consists of plated-through holes, 20 columns wide by 27 rows long. The top row of holes is connected to GND and the bottom row is connected to +5V. The remaining holes are not connected. Use the rework area to insert custom components and wire-wrap their respective signal, power, and ground pins.

3.6 PC Connection

Use the **LONGEST (6-foot/2-meter)** of the five USB cables provided to connect the host computer to the USBTracer Analyzer box.

3.7 Analyzer PC Requirements

Please refer to the Readme.txt file on the installation CD for PC and operating system requirements.

3.8 Setting Up the Analyzer

- Step 1** Connect the provided AC power cord to the rear of the UPAS and to a 100-volt to 240-volt, 50 Hz to 60 Hz, 100 W power outlet.

Note The Analyzer is capable of supporting supply voltages between 100-volt and 240-volt, 50 Hz or 60 Hz, thus supporting all known supply voltages around the world.

Step 2 Turn on the power switch on the rear of the analyzer.

Note At power-on, the analyzer initializes itself in approximately ten seconds and performs an exhaustive self-diagnostic that lasts about five seconds. The Trigger LED illuminates during the power-on testing and turns off when testing is finished. If the diagnostics fail, the trigger LED blinks continuously, indicating a hardware failure. If this occurs, call LeCroy Customer Support for assistance.

Step 3 Connect the USB cable between the USB port on the back of the analyzer and a USB port on the analyzing PC.

The host operating system detects the analyzer and begins to install the USB driver.

Installing the USBTracer Application Program

3.9 Installing the Analyzer Software on the PC

Once *USBTracer* has been recognized as a USB device, install the *USBTracer* software on the PC administering the analyzer.

- Step 1** Insert the *USBTracer* CD into the CD ROM drive of the PC that will be administering the Analyzer.
- Step 2** Follow Windows on-screen Plug-and-Play instructions for the automatic installation of the *USBTracer* Analyzer as a USB device on your analyzing PC (the required USB files are included on the *USBTracer* CD).
- Step 3** Select **Install Software** from the installation CD and follow the on-screen installation instructions.

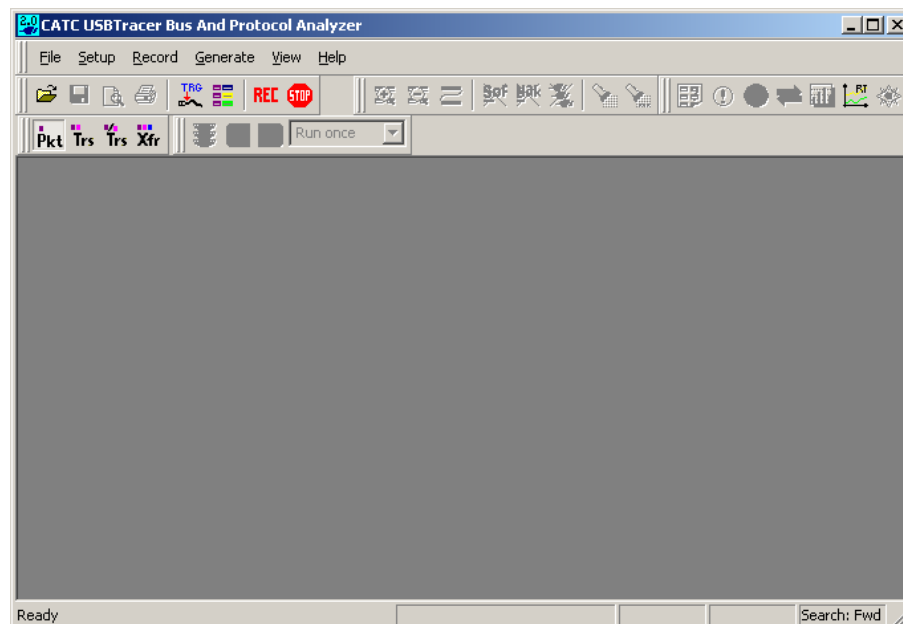
The *USBTracer* application will install on the PC hard disk.

- Step 4** To start the application, launch the **LeCroy USBTracer** program from the **Start Menu: Start>Programs>LeCroy>USBTracer**.

The *USBTracer* program opens.

3.10 USBTracer Program Startup

You can start the *USBTracer* program from the Desktop or from the installed directory. The program always begins with its main screen active:



The software may be used with or without *USBTracer*. When used without *USBTracer*, the program functions in a Trace Viewer mode to view, analyze, and print captured protocol traffic.

When the program is used with the *USBTracer* Protocol Analyzer attached to the computer, you can set trigger conditions, record, monitor and analyze the activity of your USB bus.

3.11 Making a Recording

After installation, the software is configured to make a Snapshot recording of USB traffic.

To make your first recording of this traffic, see See Section 2.2 "Your First USB Recording" on page 14.

4. Updates

From time to time as modifications are made to the USBTracer Analyzer, it is necessary to update the USBTracer for optimal performance. Updates can be performed two ways: either automatically or manually. This chapter describes both procedures.

4.1 Software, Firmware, and BusEngine Revisions

The **Readme.txt** file on the first installation disk and in the installed directory gives last-minute updates about the current release. Included with each release are the most recent downloadable images of the Firmware and the BusEngine.

Once the Analyzer has completed the self diagnostics and is connected to the PC, you can check the latest revision of the software and BusEngine by selecting **About USBTracer...** from the **Help** menu.



About USBTracer details revisions of the following software and hardware:

- USBTracer Software Version
- USBTracer Firmware Version
- BusEngine Version
- Unit Serial Number

Note When contacting LeCroy for technical support, please have available all the revisions reported in the **About USBTracer** window.

4.2 Software Updates

When a new software release is available, it is posted on the Support page of the LeCroy website at **www.lecroy.com/support.html**.

To update the software,

- Step 1** In the **About USBTracer** screen, verify which version of USBTracer Software you are currently running.
- Step 2** Find the latest released software version on the LeCroy website under **Support**.

If you are running the latest version of the software, no further action is needed.

If you are **not** running the latest version, continue to Step 3.
- Step 3** Download the Zip files from the website.
- Step 4** Unzip the files into your choice of directory.
- Step 5** Click **Start**, then **Run**, and browse to where you unzipped the files.
- Step 6** Select the program named **Setup** and click **Open**.
- Step 7** Click **OK** to run the Setup and begin the installation.
- Step 8** Follow the on-screen instructions to complete the installation.
- Step 9** Read the Readme file for important information on changes in the release.

4.3 BusEngine and Firmware Updates

BusEngine and Firmware updates often need to be performed when you update the USBTracer software. These updates can be performed automatically or manually. Both processes are described.

Updating the BusEngine

The BusEngine core is the heart of the USBTracer Analyzer. Using state-of-the-art Electronically Programmable Logic Device (EPLD) technology, it incorporates both the high speed recording engine and the configurable building blocks that implement data/state/error detections, triggering, capture filtering, external signal monitoring, and event counting and sequencing. Both the BusEngine program and the Firmware that manages the internal microcontroller are fully field upgradeable.

Within a new software release, it may be necessary to update the Analyzer's BusEngine hardware for proper operation. The Readme file lets you know if this is necessary.

Updating the Firmware

Within a new software release, it may also be necessary to update the Analyzer's firmware for proper operation. The Readme file informs you if this is necessary.

Automatic Updates

When the USBTracer software is upgraded, the software may become incompatible with the BusEngine and Firmware. If a recording is attempted, USBTracer will display an error message and then automatically begin an update process for the BusEngine and Firmware.

To perform an automatic BusEngine and Firmware update, follow these steps:

Step 1 If needed, update the USBTracer software using the steps outlined in "Software Updates" described above.

Step 1 Turn on the Analyzer.

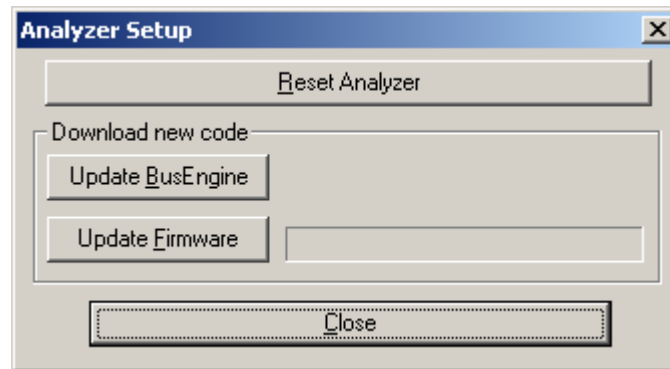
Step 2 On the Tool Bar, click the  button.

Since the BusEngine and/or the Firmware are incompatible with the current USBTracer software version, an error message will appear displaying your current versions and indicating what versions you need to install.



Step 3 Click **Yes**.

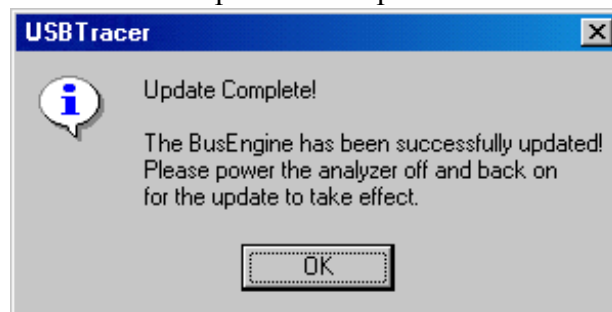
The above window closes and the Analyzer Setup window opens.



- Step 4** Click **Update BusEngine** or **Update Firmware** on the **Analyzer Setup** screen.

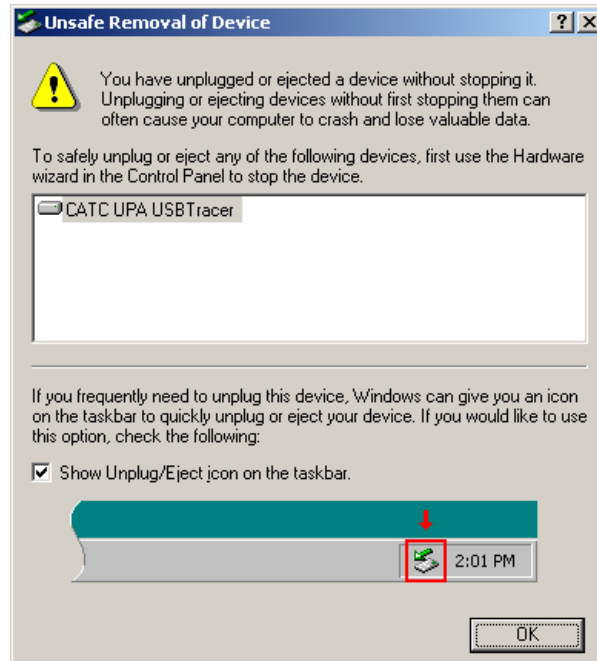
You can select only one item at this point. If both the BusEngine and the Firmware need to be updated, the update will complete for the first item and then return to the above screen so the second update can be performed.

If you are running Windows 98, you will get the following message when the second update is completed:



- Step 5** Power cycle *USBTracer* to complete the update.

If you are running Windows 2000, the analyzer will automatically reboot and cause the following message to appear. When this message appears, click **OK**. **You do not need to power cycle the analyzer.**

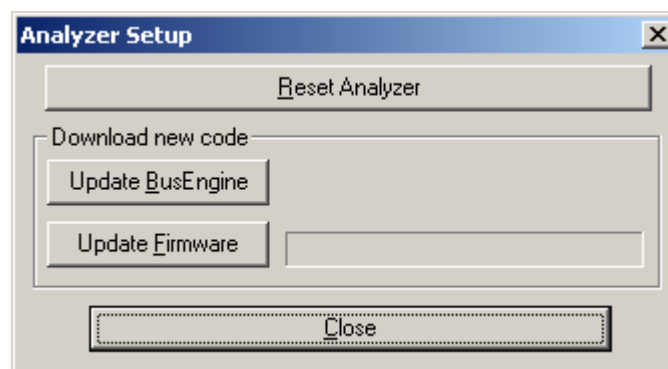


Manual Updates to BusEngine and Firmware

You can manually update the USBTracer Firmware and/or BusEngine by performing the following steps:

Step 1 Select **Analyzer** under **Setup** on the Menu Bar.

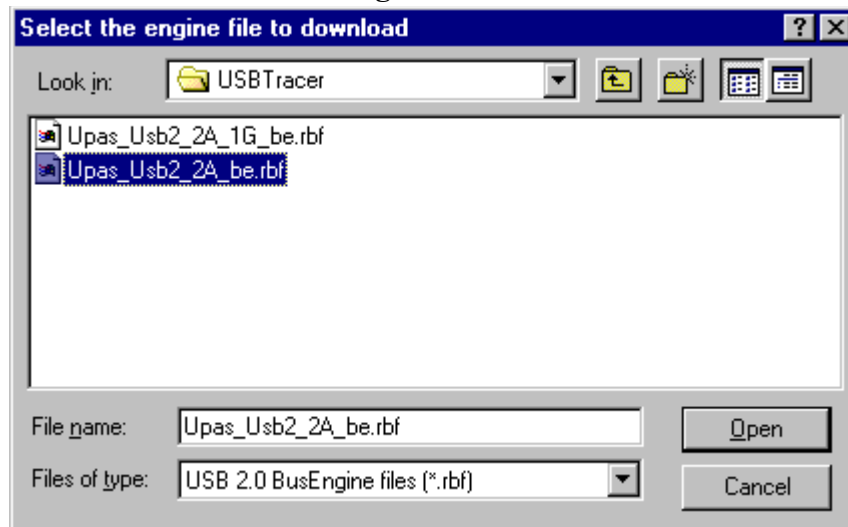
You see the **Analyzer Setup** screen:



To update the BusEngine,

Step 2 Click **Update BusEngine** on the **Analyzer Setup** screen.

You see the **Select engine file** window:



The program displays the correct file (**Upas_Usb2_2A_be.rbf**) in the **File name** field. This file is used to update the analyzer alone (i.e., without a Traffic Generator). The other file (**Upas_Usb2_2A_1G_be.rbf**) should be selected if you upgrading a UPAS system with both an analyzer and Traffic Generator.

If you have purchased device emulation, then the needed BusEngine file is **Upas_usb2_1A_1G-be.rbf**.

Note The most current Primary BusEngine file was copied to your **\\LeCroy\\USBTracer** directory when you installed the program.

Step 3 Click **Open**.

Step 4 Power cycle the analyzer.

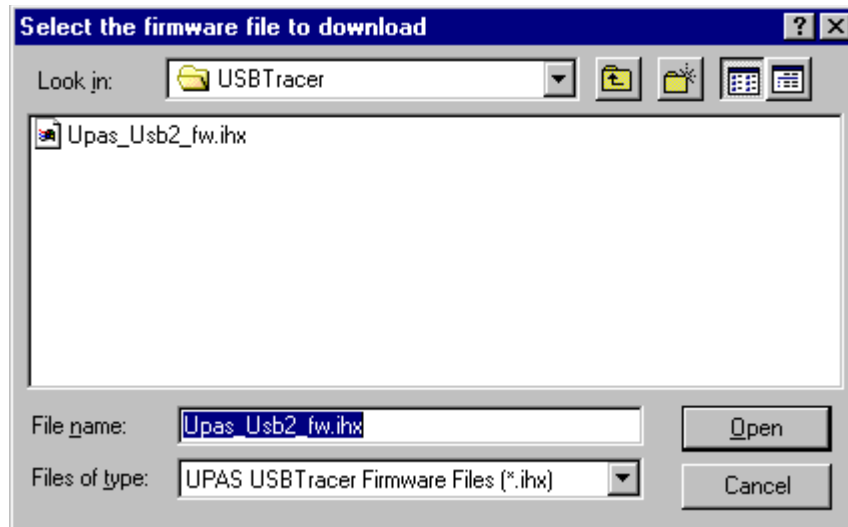
Re-initialization takes a couple of minutes.

Manually Updating the Firmware

To update the firmware,

Step 1 Click **Update Firmware** on the **Analyzer Setup** screen.

You see the **Select firmware file** window:



The program has already automatically searched for the correct file and displays it in the **File name** field.

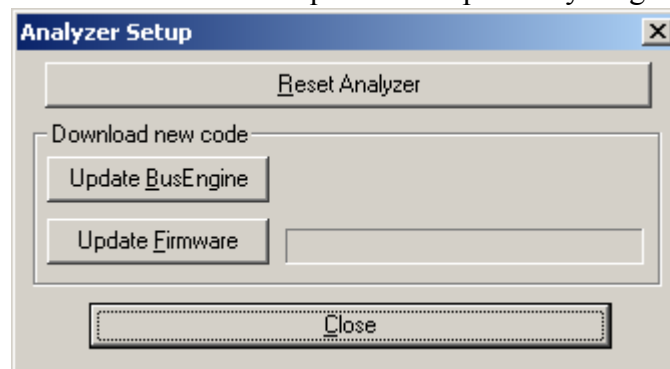
Step 2 Click **Open**.

The Analyzer updates the Firmware.

Step 3 Unplug the USB cable from the back of the Analyzer box and then reinsert it so the new Firmware update can take effect.

4.4 Resetting the Analyzer

The Analyzer Setup window contains a reset button at the top of the window. Its function is the equivalent of power cycling the Analyzer.



5. Software Overview

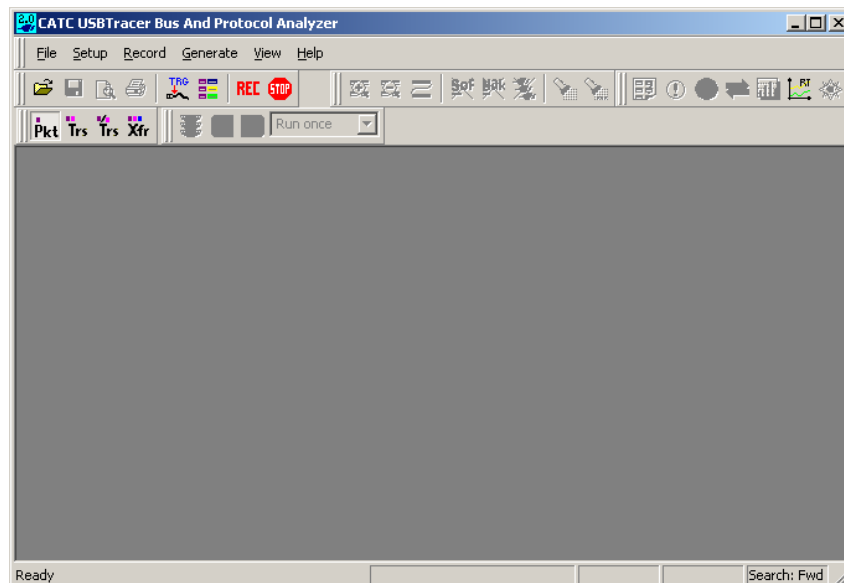
The USBTracer Trace Viewer is an application that may be used with or without the Analyzer box. When used without an Analyzer box, the program functions in a Trace Viewer mode to view, analyze, and print captured protocol traffic from USBTracer Analyzers. The software will also allow you to view trace files created by the LeCroy Chief, Advisor, Detective, and Inspector. Opening a file created with either of these Analyzers displays a screen asking if you want to convert the old file to the new format under the name **convert.usb**.

When the program is used with the USBTracer Analyzer attached to the computer, you can monitor and analyze the activity of your USB branch from either of the USB ports on the front of the Analyzer.

5.1 Starting the USBTracer Program

To starting the USBTracer Program:

Step 1 Select **Start > Programs > CATC > USBTracer**.



5.2 Tool Tips

Tool tips provide useful information about cells in the trace and buttons in the application. To display a tool tip, position the mouse pointer over an item.



5.3 The Main Display Windows

While some of the analyzer's Main Display window options are familiar, many contain options specific to the analyzer program.



Table 1: Main Display Pull-Down Windows

Menu	Function
<u>F</u>ile	
New .utg file ...	Creates a new, empty traffic generation file. This menu item appears only if a trace file (.usb) is open. To edit, click the Edit as Text button in the toolbar.
<u>O</u> pen...	Opens a file.
<u>C</u> lose	Closes the current file.
Save <u>A</u> s...	Saves all or a range of packets from the current file.
<u>P</u> rint...	Prints part or all of the current traffic data file.
Print <u>P</u> review	Produces an on-screen preview before printing.
<u>P</u> rint Setup...	Sets up your current or new printer.
<u>E</u> dit as Text	Opens the Script Editor. This command is present in the menu only when a traffic generation file (.utg) is open.
<u>E</u> dit Comment...	Creates or edits the Trace file comment field.
Check Syntax of .utg file	Reads open .utg file and checks syntax for errors. This command is present in the menu only when a traffic generation file (.utg) is open.
Export>>	Saves all or part of a trace to a text file or generator file.
Packets to Text (Packet View Format)	Packets to Text (Packet View Format)- Saves trace as text file.
Packets to Text (Generator Text File Format)	Packets to Text (Generator Text File) - Saves trace as script file that can be used by a Generator to generate a trace.
Packets to Device Emulation Traffic Generation Text File	Exports packets to Device Emulation files. This option does not export transactions.
Data	Data - Allows Transfer data to be exported as text or binary file.
Merge Trace Files	Merges two simultaneously recorded files into a single file. (This command will not work if the files were recorded at different times).
<u>I</u> mport	Import data from .vcd file
Compare Endpoint Data	This command allows the user to select two endpoints of different directions with same address and verify that the data OUT/IN is identical to the data IN/OUT from the other endpoint. This command is useful when running echo-types of tests for data integrity. This command is present in the menu only when a trace file (.usb) is open.
<i>Last File</i>	Lists the last files that were opened.
<u>E</u> xit	Exits the USBTracer program.

Menu	Function
Setup	
Display Options...	Provides the control of various display options such as color, formats, and filters.
Recording Options...	Provides setup options for recording, triggering events and filtering.
Analyzer...	Allows the operator to update the BusEngine and Firmware.
Record	
Start	Causes the Analyzer to begin recording USB activity.
Stop	Causes the Analyzer to stop recording.
Upload Again	If the previous upload was only partially uploaded, use this to upload a different portion of the captured trace.
Generate	
Start	Starts traffic generation. Once traffic generation has begun, the Start command becomes Stop and lets you stop traffic generation.
Report	
File Information	Displays information about the recording such as the number of packets and triggering setup.
Error Summary	Summarizes the errors throughout the recording. Allows for fast navigation to packet with errors.
Timing Calculation	Calculates timing between two packets and bus utilization.
Traffic Summary	Summarizes the numbers and types of packets, transactions, split transactions, and transfers that occurred in the open trace.
Bus Utilization	Opens a window that displays graphs of bus utilization data. Once the window is open, the Bus Utilization menu will let you access the following sub-menu options: Hide, Save, Send, Print, Full Screen, View Options, Set Range, Synchronize, Graph Areas.
Search	
Go to Trigger	Positions the display to show the first packet that follows the trigger event.
Go to Packet/Transaction/Transfer ...	Positions the display to the indicated packet/transaction/transfer number.
Go to Marker »	Positions the display to the selected marked packet.
Go to »	Enables quick searching for specific events using a cascade of pop-up windows.
Find	Allows complex searches on multiple criteria.
Find Next	Repeats the previous Find operation.
Search Direction	Allows the search direction to be changed from forward to backwards or vice versa.
View	
Toolbars	Displays list of available toolbars.
Status Bar	Switches display of the Status Bar on or off.
Zoom In	Increases the size of the displayed elements.

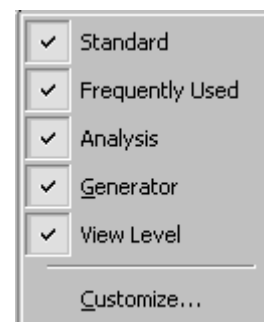
Menu	Function
Zoom <u>O</u> t	Decreases the size of the displayed elements.
Real-time Statistics	Allows you to view traffic statistics as they occur.
<u>W</u> rap	Wraps displayed packets within the window.
Hide SOF's	Hides Start of Frames.
Hide NAK's	Hides NAK'ed Transactions.
Hide Devices	Hides packets belonging to specified devices by address and endpoint.
Hide Chirps	Hide the Chirp-K and Chirp-J Bus conditions. These are recorded only in USBTracer.
Packet Level	Displays Packets.
Transaction Level	Displays Transactions.
Split Transaction Level	Displays Split Transactions.
Transfer Level	Displays Transfers.
Refresh Decoding	Force the software to re-decode transactions and transfers. Useful if you have applied a decoding mapping which will help fully decode a sequence of transfers, as is the case with Mass Storage decoding.

Window	
<u>N</u> ew Window	Switches display of the Tool Bar on or off.
<u>C</u> ascade	Displays all open windows in an overlapping arrangement.
<u>T</u> ile	Displays all open windows in a side-by-side arrangement.
<u>A</u> rrange Icons	Arranges minimized windows at the bottom of the display.
Windows ...	Displays a list of open windows.

Help	
<u>H</u> elp Topics ...	Displays online help.
<u>U</u> pdate License...	Opens a dialog box for updating your LeCroy license.
<u>D</u> isplay License Information...	Displays information related to licensing.
<u>A</u> bout USB USBTracer	Displays version information about USB USBTracer.

5.4 View Options

You can hide, display or reset toolbars by selecting **View > Tool bars** from the menu bar.



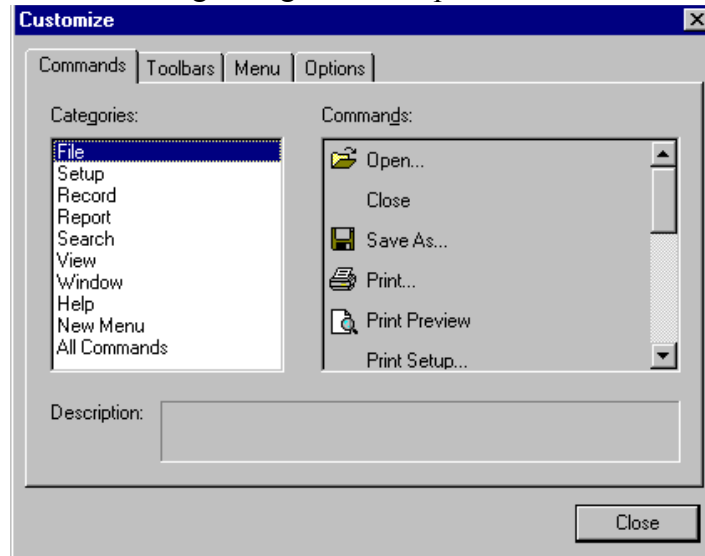
Resetting the Tool bar

From time to time (such as following a software upgrade) it is possible for the buttons on the toolbar to not match their intended function. You can reset the toolbar by performing the following steps:

Step 1 Select **View >Tool bars** from the menu bar.

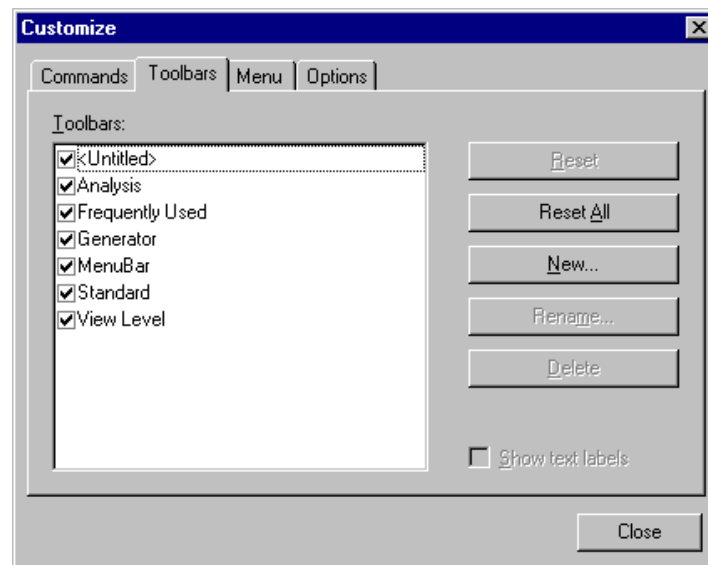
Step 2 Select **Customize** from the sub-menu.

The following dialog box will open.



Step 3 Select the **Toolbars** tab.

The following dialog box will open.












Step 4 Click the **Reset All** button.

The toolbar resets to the factory defaults.

Tool Bar



The Tool Bar provides quick and convenient access to the most popular USBTracer program functions. Tool tips briefly describe the functionality of each icon and menu item as the mouse arrow is moved over the icon/item.

	Open file		Find Next
	Save As		File Information Report
	Preview		Error Report
	Edit as Text		Timing and Bus Usage Calculations
	Print...		Traffic Summary
	Setup Record Options		Bus Utilization
	Setup Display Options		Display Packets
	Start Recording		Display Transactions
	Stop Recording		Display Split Transactions
	Zoom In		Display Transfers
	Zoom Out		Start or Stop Traffic Generation (available if you have the USBTrainer module)
	Wrap		Intelliframe/Bitstream modes
	Hide SOFs		Display Realtime Statistics monitor
	Hide Chirps		Open the Navigator bar



Assign High Level Decodes



Hide Devices



Find



Repeat Upload



Hide NAK'ed transactions



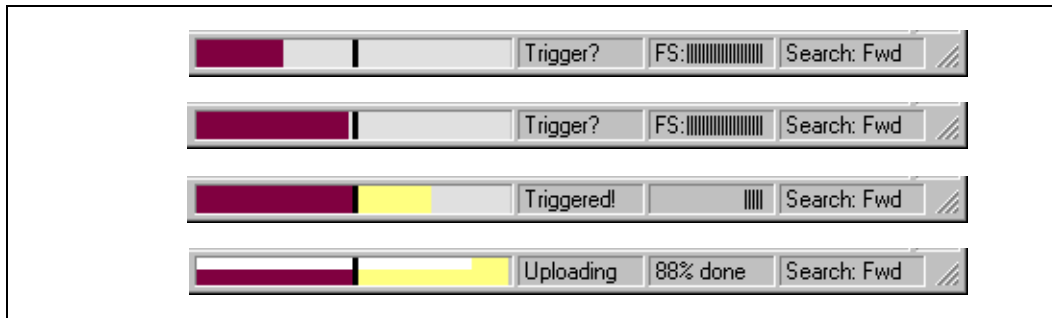
Repeat Mode. This button becomes active when a traffic generation file (*.utg) is open.

5.5 Status Bar

The Status Bar is located at the bottom of the main display window. Depending on the current activity, the bar can be divided into as many as four segments.

Recording Progress

When you begin recording, the left-most segment of the Status Bar displays a Recording Progress Indicator:



As recording progresses, the Progress Indicator changes to reflect the recording progress graphically:

- In the Progress Indicator, a black vertical line illustrates the location of the Trigger Position you selected in Recording Options.
 - Pre-Trigger progress is displayed in the field to the left of the Trigger Position in the before-Trigger color specified in the Display Options.
 - When the Trigger Position is reached, the progress indicator wiggles as it waits for the trigger.

- After the trigger occurs, the field to the right of the Trigger Position fills in the after-Trigger color specified in the Display Options.
- When recording is complete, the upper half of the progress indicator fills in white, indicating the progress of the data upload to the host computer.

You should be aware of two exceptional conditions:

- If a Trigger Event occurs during the before-Trigger recording, the before-Trigger color changes to the after-Trigger color to indicate that not all the expected data was recorded pre-Trigger.
- When you click **Stop** before or after a Trigger Event, the Progress Bar adjusts accordingly to begin uploading the most recently recorded data.

The Progress Bar fills with color in proportion to the specified size and actual rate at which the hardware is writing and reading the recording memory. However, the Progress Indicator is normalized to fill the space within the Status Bar.

Recording Status

During recording, the current Recording Status is displayed in the next segment. When you activate the **Record** function, this segment flashes one of the following messages (depending on the selected Recording Options):

- Trigger?
- Triggered!
- Uploading

After recording stops,

- The flashing message changes to **Uploading data-x% done** (**x%** indicates the percentage completion of the data uploading process).
- The traffic data is copied to disk (overwriting any previous version of this file) using the default file name **data.usb**. If two channels of traffic are recorded, USBTracer will create two files: **data_0.usb** and **data_1.usb**. You can also create a file name of your choice by specifying one in the Recording Options dialog box.

To abort the upload process,

- Press **Esc** on your keyboard

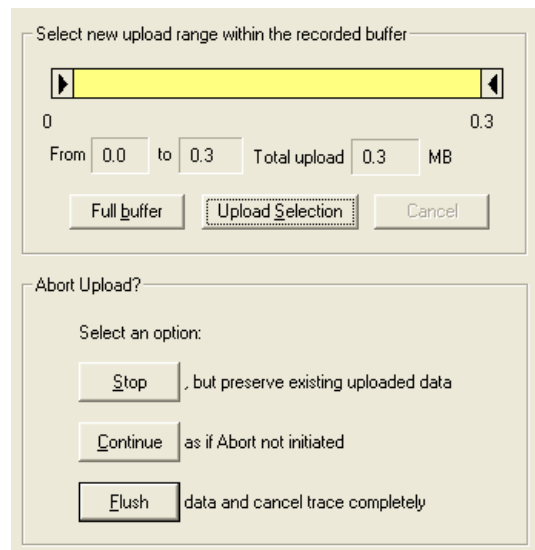
OR

- Again click  in the Tool Bar.

You are asked if you want to keep or discard the partially uploaded data.

Note: While uploading is in progress, clicking the stop button again brings up a dialog that allows you to do a partial upload, flush the current file, keep what has uploaded at this point, and to continue uploading.

The Partial Upload button is enabled when you have partially uploaded data. When you click this button, a dialog box appears that gives you options on what portion of data you want to upload again.



When the data is saved, the Recorded Data file appears in the main display window and the Recording Status window is cleared.

- If the recording resulted from a Trigger Event, the first packet following the Trigger (or the packet that caused the Trigger) is initially positioned second from the top of the display.
- If the recording did not result from a Trigger Event, the display begins with the first packet in the traffic file.

Recording Activity

During recording, the fourth segment from the left of the Status Bar displays Recording activity as a series of vertical bars.

The more vertical bars that are displayed, the greater the amount of activity being recorded. If there are no vertical bars, there is no recorded activity.

During uploading, the percent of the completed upload is displayed.

Note If packets are filtered from the recording or data are truncated, the recording activity is reduced.

Search Status


The rightmost segment displays the current search direction: **Fwd** (forward) or **Bwd** (backward). Change the search direction from the Search Menu or double-click in the Search Status segment.

5.6 Navigation Tools

You can zoom in and out, and wrap packets/transactions/transfers to fit within the screen using the following buttons:


Zoom In

Zoom In increases the size of the displayed elements, allowing fewer (but larger) packet fields per screen.

- Click  on the Tool Bar.


Zoom Out

Zoom Out decreases the size of the displayed elements, allowing more (but smaller) packet fields per screen.

- Click  on the Tool Bar.

Wrap

Select **Wrap** to adjust the Trace View so that packets fit onto one line. If a packet is longer than the size of the window, the horizontal scroll bar can be used to see the hidden part of the packet.

- Click  on the Tool Bar or select **Wrap** under **View** on the Menu Bar.

5.7 USBTracer Analyzer Keyboard Shortcuts

Several frequently-used operations are bound to keyboard shortcuts.

Table 2: Keyboard Shortcuts

Operation	Key Combination
Trace Navigation	
Find Next	F3
Search Backwards	Ctrl+B

Search Forwards	Ctrl+F
Jump to First Packet	Ctrl+Home
Jump to Last Packet	Ctrl+End
Goto Any Error	Shift+E
Goto Channel 0	Ctrl+Shift+0
Goto Channel 1	Ctrl+Shift+1
PID	
Goto ACK	Shift+A
Goto DATA0	Shift+0
Goto DATA1	Shift+1
Goto DATA2	Shift+2
Goto DATAx	Shift+D
Goto IN	Shift+I
Goto MDATA	Shift+M
Goto NAK	Shift+N
Goto NYET	Shift+Y
Goto OUT	Shift+O
Goto PING	Shift+G
Goto PRE/ERR	Shift+P
Goto SETUP	Shift+S
Goto SOF	Shift+F
Goto SPLIT	Shift+X
Goto STALL	Shift+L
Bus Conditions	
Goto Reset	Shift+T
Goto Resume	Shift+6
Goto SE0	Shift+Z
Goto SE1	Shift+7
Goto Keep-Alive	Shift+5
Goto Suspend	Shift+U
Goto Chirp	Shift+C

Goto Full Speed J	Shift+J
Goto Full Speed K	Shift+K
OTG	
Goto SRP	Ctrl+Q
Goto HNP	Shift+H
Goto VBus Voltage Change	Shift+V
Goto OTG Host A	Ctrl+Shift+A
Goto OTG Host B	Ctrl+Shift+B
Misc.	
Marker Menu	Ctrl+M
Open File	Ctrl+O
Print...	Ctrl+P
Record	Ctrl+R
Stop Recording	Ctrl+T
Open Display Options dialog	Ctrl+Shift+D
Open Recording Options dialog	Ctrl+Shift+R
Hide SOFs	Ctrl+Shift+S
Hide NAKs	Ctrl+Shift+N
Hide Chirps	Ctrl+Shift+C


6. Recording Options

Use **Recording Options** to create and change various features that control the way information is recorded by the USBTracer Analyzer.

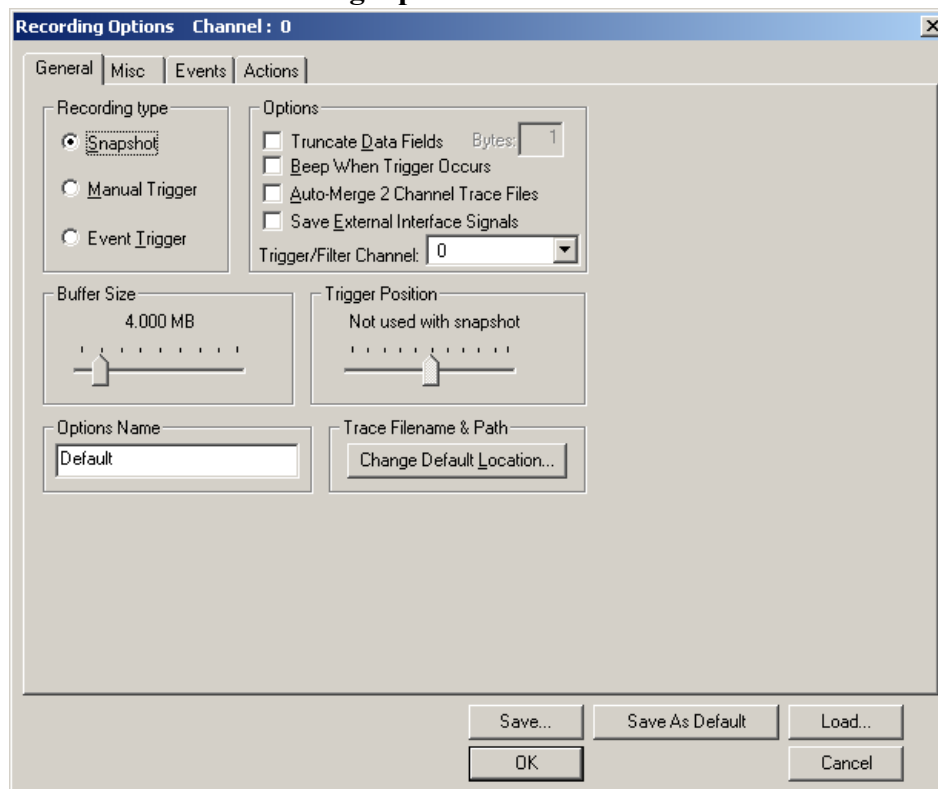
To open the **Recording Options** menu,

- Select **Recording Options** under **Setup** on the Menu Bar

OR

- Click  on the Tool Bar.

You see the **Recording Options** window:



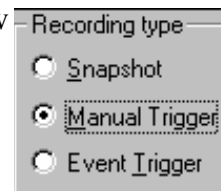
The **Recording Options** window always opens with the **General** tab showing.

6.1 General Recording Options



The General Recording Options allow you to select or adjust the recording type, the buffer size, the amount of post-trigger recording, and the trace filename and path.

Recording Type

The **Recording Type** box presents three options that allow you to set how USBTracer begins and ends a recording. The options are: *Snapshot*, *Manual Trigger*, and *Event Trigger*.





Snapshot


A Snapshot is a fixed-length recording. The size of this recording is set by the "Buffer Size" box. Recording begins when  is clicked and ends when either the selected buffer size is filled or the  button is pressed.

Manual Trigger

A Manual Trigger is a recording whose trigger point is caused by pressing the trigger button on the front panel.

Recording begins when you click  on the Tool Bar. Recording continues in a circular manner within the limits set by the buffer size. Recording ends when  is clicked on the Tool Bar or after post-trigger memory has been filled following depression of the trigger button on the front panel.

Event Trigger

Recording begins when you click  on the Tool Bar.

Recording continues in a circular manner within the limits set by the buffer size until an event is detected that meets the Trigger conditions specified in the Triggering Options and the defined amount of data has been recorded after the Trigger Event.

Options

- **Truncate Data Fields:** Allows data fields to be truncated during recording in order to save Analyzer memory and allow recording of more packets. Enter a minimum data length value in the box marked "Bytes." USBTracer will truncate the data to the stated value or slightly larger.

Note Truncation of data may cause incorrect transaction or transfer decoding.

- **Beep When Trigger Occurs:** The computer connected to the USBTracer will beep three times when a Trigger condition is first detected.
- **Auto-Merge 2 Channel Trace Files:** Causes USBTracer to merge traffic of the two recording channels into a file called `data_merged.usb` (or whatever name you wish to give it). USBTracer will also make two other files during this recording: `data_0.usb` and `data_1.usb` (or whatever names you choose to give them). If unchecked, USBTracer will just create two files: one for Classic-Speed traffic and another for Hi-Speed traffic.
- **Save External Interface Signals:** This option will cause USBTracer to save signals from a Break-out board (pins *Data7* - *Data0*) save them as fields in the trace.
- **Trigger/Filter Channel:** Presents a choice of Channel 0 or Channel 1. The selected choice tells USBTracer what channel it should use to perform its triggers/filters. If you want to set triggers/filters on *both* channels, then you will need to select a channel, assign trigger/filter events and actions, then select the other channel and assign trigger filter events and actions to it. In other words, you must set your events and actions *twice* - once for each channel. The channel you are currently applying the triggers/filters to is shown in the Dialog Title Bar.

Buffer Size

You can adjust the size of the recording buffer from 0.4 megabytes to 512 megabytes.

The **Recording type** option determines how this buffer is used. Although there are 512 megabytes of physical memory in the Analyzer, the efficiency of the recording is about a 2:1 ratio of physical memory to actual USB traffic. Shorter USB packets yield a slightly less efficient recording. The non-traffic portion of physical memory is utilized for control and timing information.

Note The scale is not linear and affords more granularity in the smaller buffer sizes.

Trigger Position

You can adjust the amount of recording to be done post-Trigger or select where you want the Trigger located within the defined buffer. You can adjust the Triggering Position between 1 and 99% post-Trigger. **Trigger Position** is available only when **Manual Trigger** or **Event Trigger** is selected as **Recording type**.

As an example, if the buffer size is set to 16MB, then for the following Trigger Position settings, the amount of pre- and post-Trigger data is

- 95% post-triggering: 0.8MB pre-trigger, 15.2MB post-trigger
- 75% post-triggering: 4MB pre-trigger, 12MB post-trigger
- 50% post-triggering: 8MB pre-trigger, 8MB post-trigger
- 25% post-triggering: 12MB pre-trigger, 4MB post-trigger
- 5% post-triggering: 15.2MB pre-trigger, 0.8MB post-trigger

Note When a Trigger occurs, recording continues until the post-Trigger amount of the buffer is filled or when **Stop** is selected.

Options Name

The **Options Name** is a descriptive label of the current Recording Options settings. Options Names are associated with files that have a **.rec** suffix.

The default option name is **default**. **Default** preserves the current Recording Options settings.

The purpose of the **Options Name** box is to give you a place to preserve different Recording Options that you use on a recurrent basis. For example, if you use two or three different Recording Options configurations, you can save these configurations and load them the next time they are needed.

Because Options Names are descriptive labels and not file names, you can enter in any text you like into the box. Your labels can be very descriptive such as "Trigger on High Speed traffic when CRC errors occur."

To create a new Recording Options name,

Step 1 Enter a comment for the new file in the **Options name** field.

Step 2 Click **Save...**

You see the **Save As** window.

Step 3 Specify a filename (***.rec**)

Step 4 Click **Save**.

To load a Recording Options name,

Step 1 Click **Load ...**

You see the **Open** window.

Step 2 From the list of **.rec** files, select the one that represents your Options Name.

The options settings for that name then display.

Trace File Name & Path

Trace File Name & Path opens a **Save As** dialog box for saving your trace file. The default recording file name is **data.usb** for Low, Full, and High Speed recordings. If you are recording on both channels, then *USBTracer* will create two files: **data_0.usb** for Channel 0, and **data_1.usb** for Channel 1.

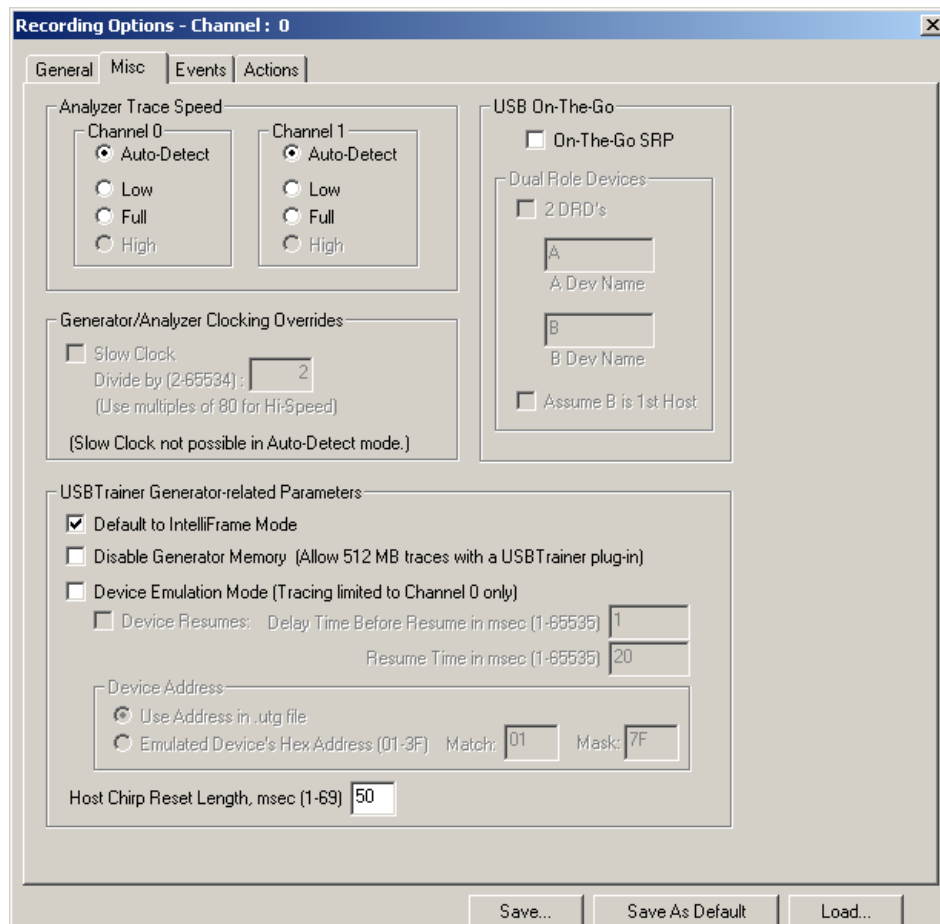
- Click **Trace File Name & Path**.

You see a **Save As...** window in which to enter the recording file name ***.usb** for all subsequent recordings.

6.2 Recording Options - Misc

The **Misc** page presents options for setting:

- Analyzer speed
- Device emulation
- OTG
- Clocking Overrides
- Traffic Generation



Analyzer Speed

This option sets the speed of the traffic recorded by the analyzer. Normally, the default setting "Auto-detect" is used. This setting tells the analyzer to discover what speed traffic is running and to label packets accordingly. If you are having problems with your recordings, you might try setting the traffic speed to one of the fixed values -- Low, Full, and Hi speed. These settings are used when you want to manually set the traffic speed.

In some rare cases, auto-detection circuitry causes Full Speed devices to fail to enumerate on plug-in. Changing the analyzer speed to Full Speed can sometimes solve this problem.

Setting the speed to one of the fixed values is sometimes useful for debugging purposes. For example, if a device that is producing numerous errors at a particular speed, you may wish to set the recording to that speed in order to ensure that the analyzer does not misread the error packets and

label them the wrong speed. While it is unlikely that the analyzer will mislabel packets in this way, manually setting the recording speed guarantees that the analyzer will always record packets at the correct speed.

You can set the speed for one or both channels.

Notes on Hi Speed Recordings

Erroneous chirp blocks can be recorded on an idle bus when the Device has its FS terminations on while the Host has HS terminations connected. This causes a small differential voltage (a.k.a "tiny-J") on the USB bus that causes false Chirp detection.

This condition occurs during speed negotiation.

- 1 On a HS bus, the condition is momentary just before the device chirps.
- 2 On a CS bus, the condition occurs both before and after the device chirp (until the end of Reset). The user is discouraged from using Speed=HIGH to record signals on a classic speed bus.
- 3 The analyzer will stop recording anything for 2.5ms following a FS_K state (which is at least 2us long). This is to avoid presenting "garbage" which is a by-product of the high-speed probe settling down.

Generator/Analyzer Clocking Overrides

Generator/Analyzer Clocking Overrides allows changes to be made to the analyzer/generator clocking. Select **Slow Clock**, then enter a value in the box on the right. The value that is entered tells the analyzer how much to divide the base clock by. For example, entering a 4 will cause Full Speed traffic to be generated at a 3-megabit rate as opposed to the standard 12 megabit rate.

USB On-The-Go

USB On-The-Go option sets analyzer to record the USB On-The-Go traffic. This protocol lets you run two devices, specify one of them as the host, one of them as the device, and to assign each device a name.

USBTrainer Generator-related Parameters

Default to IntelliFrame Mode - Sets IntelliFrame default to "ON."

In IntelliFrame mode, the generator can wait for Device responses to complete before it continues generating more host packets. For example, after issuing an IN, the generator looks for the DATAx packet issued by the device to finish, and then issues an ACK. This way, the data length can vary and does not need to be pre-calculated. Also, using the "idle=TO_EOF" statement allows the generator to calculate where the end of the frame occurs, so that a subsequent "frame=AUTO" statement will create a Start of

Frame at the correct time. Moreover, the Generator can be made to retry transactions that are NAKed automatically, and to use the PING protocol where appropriate. Retries can be made to occur within the same frame or in the next frame. For further information on the flexible behavior that can be instituted during the generation, see the syntax descriptions for the generation files. An additional advantage of the IntelliFrame mode is that it creates an image in the Generator's memory that is MUCH smaller than that of the Bitstream mode, so it downloads much faster.

Disable Generator Memory [Allow 512 MB traces with a USBTrainer plug-in] - Allows you to specify 512 MB traces on systems with a USBTrainer plug-in.

Device Emulation Mode [Tracing limited to Channel 0 only] - Allows changes to the Resume time and the device address.

Device Resumes - If checked, the Device will be responsible for issuing the Resume signal after waiting the specified amount of time following a suspend. If the **Device Resumes** option is not selected, the device will wait indefinitely for the Host to resume signalling.

Delay Time - The time (in msec) to wait after suspend before initiating a Resume.

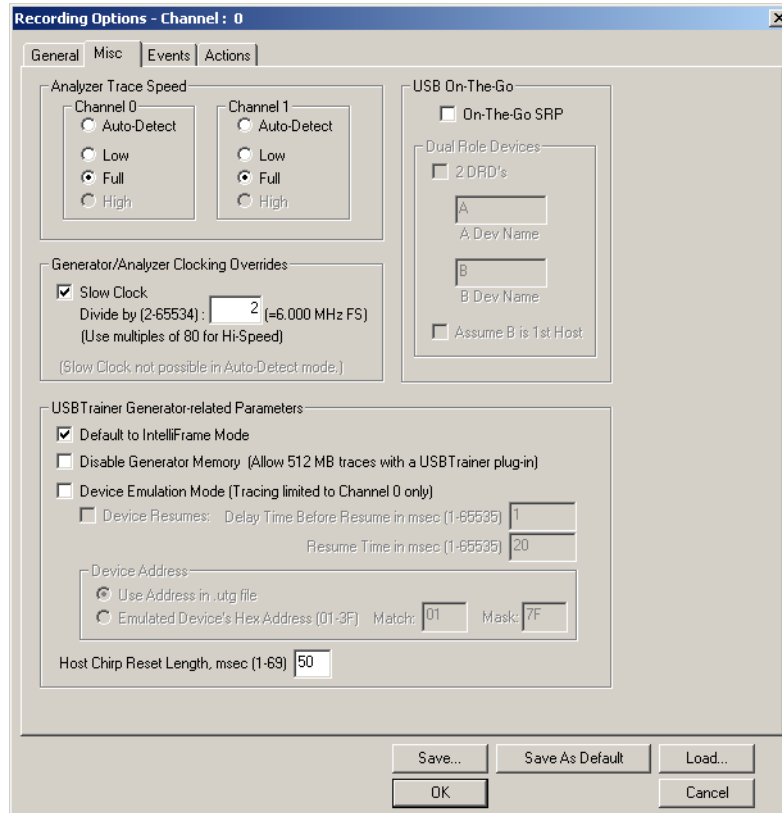
Resume Time - The Resume Time (in msec) is used to configure how long a device should issue a Resume (K) signal once the delay time has occurred. Be careful to make this shorter than what the host will be driving once it recognizes the Resume signal as a Remote Wakeup.

If the **Device Resumes** option is not selected, the device will wait indefinitely for the Host to resume signalling.

- **Use Address in .utg file** - Causes the analyzer to use whatever Device address has been specified in the .utg file.
- **Emulated Device's Hex Address** - Allows the user to specify a different address than the one specified in the .utg file. This option is useful in cases where the host does not use the same address each time.
- **Chirp Reset Length** - Allows the Reset Length for Chirps to be edited. The default is 50 msecs. (Host generation only).

6.3 Slow Clock Selection

You can use the slow clock selection to slow down the base clock during generation. This also changes the Analyzer's clock base to match.



- Step 1 In the **Misc** tab, make sure you are out of Auto-Detect mode.
- Step 2 Select the **Slow Clock** checkbox.
- Step 3 In the **Divide By** field, enter a value.
- Step 4 Click **OK**.

Using this feature for Hi-Speed requires a License Key from LeCroy. Contact your LeCroy representative.

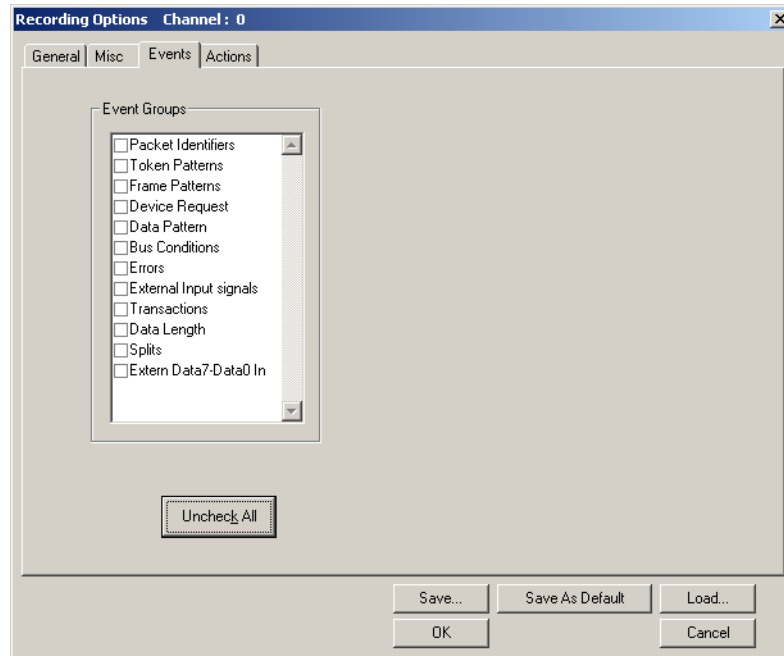
6.4 Events Recording Options

The Event triggering and filtering options allow you to set specific parameters for each Event Group. When an Event Group is selected, a field appears that allows you to select specific conditions within the corresponding Event Group. As details are selected, other Event Group details may become "grayed out" because of limited hardware resources in the Analyzer. Information about the resources available is displayed below

the Event Group details. When a detail is grayed out, it is inactive and cannot be selected. If an Event Group remains inactive, the current version of the application or BusEngine does not support it.

- Click the **Events** tab on the **Recording Options** screen.

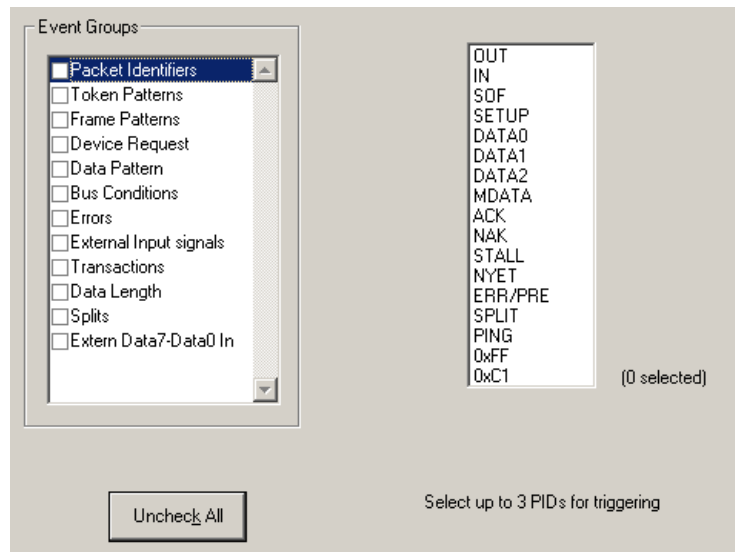
You see the **Event Groups** window:



Packet Identifiers

- Select **Packet Identifiers** under **Event Groups**.

You see the **Packet Identifiers** window:

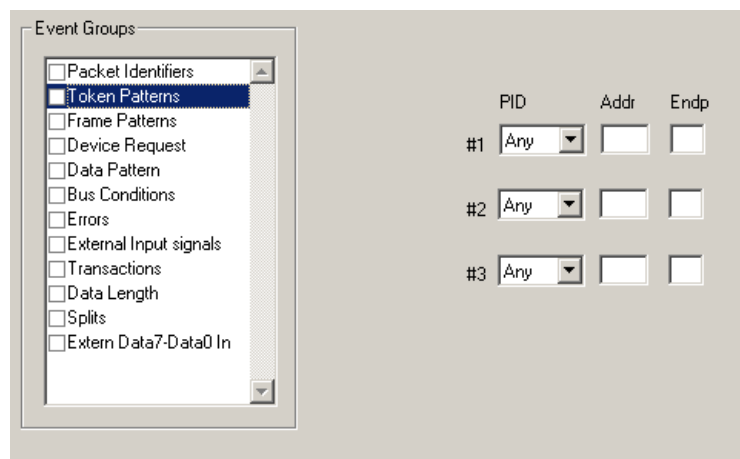


The Packet Identifier (PID) field lists the available packet types. Select up to three packet types to use as a Recording Trigger.

Token Patterns

- Select **Token Patterns** under **Event Groups**.

You see the **Token Patterns** window:

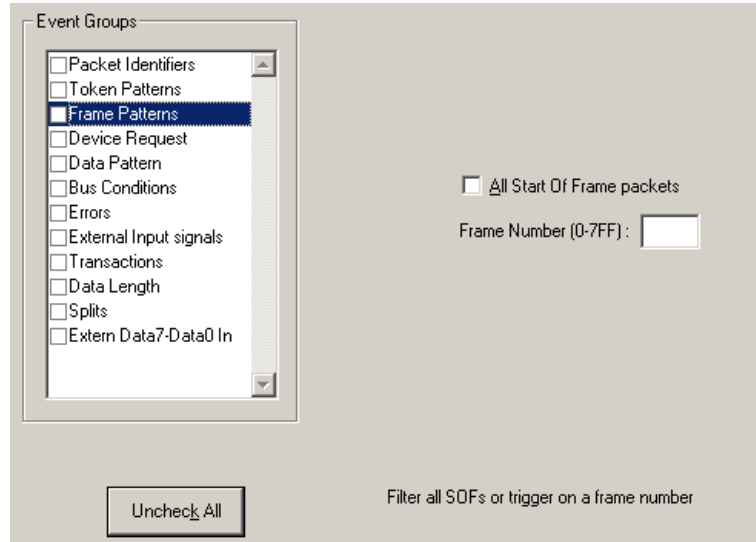


Specify up to three combinations of Address/Endpoint with any of the token PIDs. They can be triggered on or filtered in/out.

Frame Patterns

- Select **Frame Patterns** under **Event Groups**.

You see the **Frame Patterns** window:

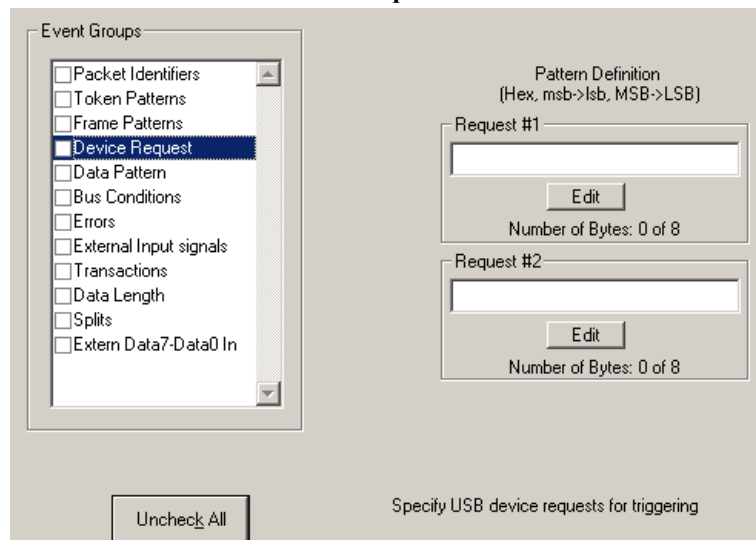


Identify frame patterns you want either to filter out of a Trace View or to use as a Trigger. You can select either **All Start of Frame packets (SOF)** to be filtered out or specify the **Frame Number** of a frame to trigger on.

Device Requests

- Select **Device Requests** under **Event Groups**.

You see the **Device Requests** window:

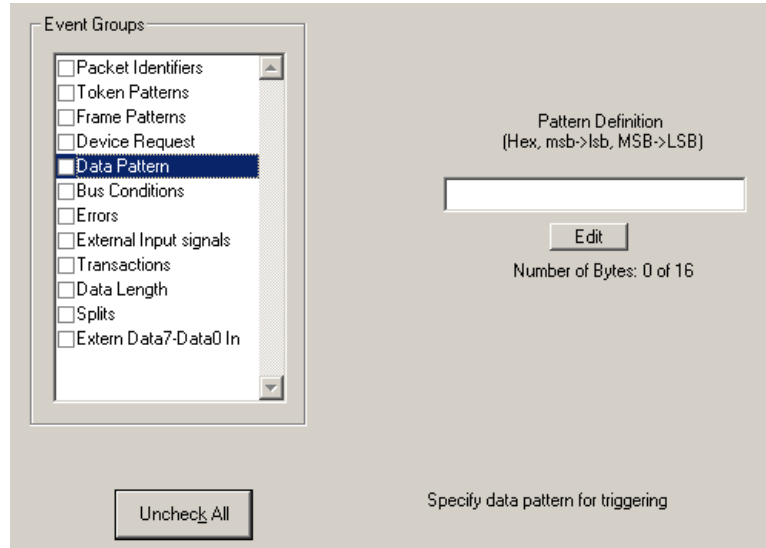


You can select one or two Device Requests for triggering or filtering.

Data Pattern

- Select **Data Pattern** under **Event Groups**.

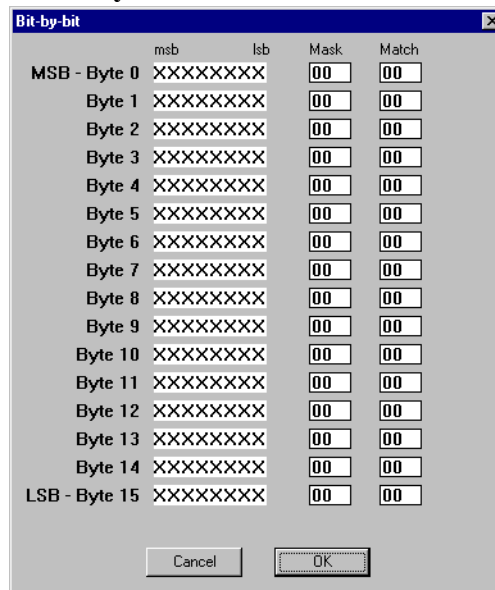
You see the **Data Pattern** window:



Enter a Data Pattern to be triggered upon or click **Edit** to open a pattern editor and enter data on a bit-by-bit basis.

Step 1 Press the **Edit** button.

You see a **Bit-by-Bit** window:



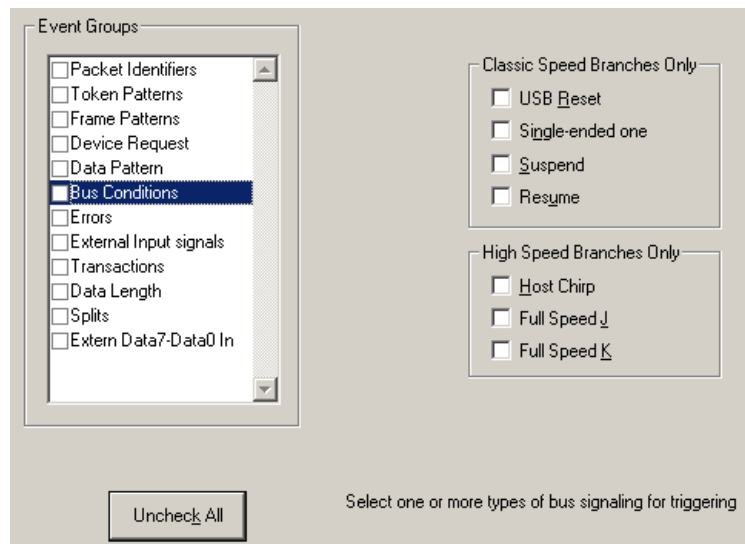
Step 2 Click your mouse on the bits within each byte that you wish to edit and enter in the appropriate text.

The Mask and Match fields will change to reflect your changes.

Bus Conditions

- Select **Data Pattern** under **Event Groups**.

You see the **Bus Conditions** window:

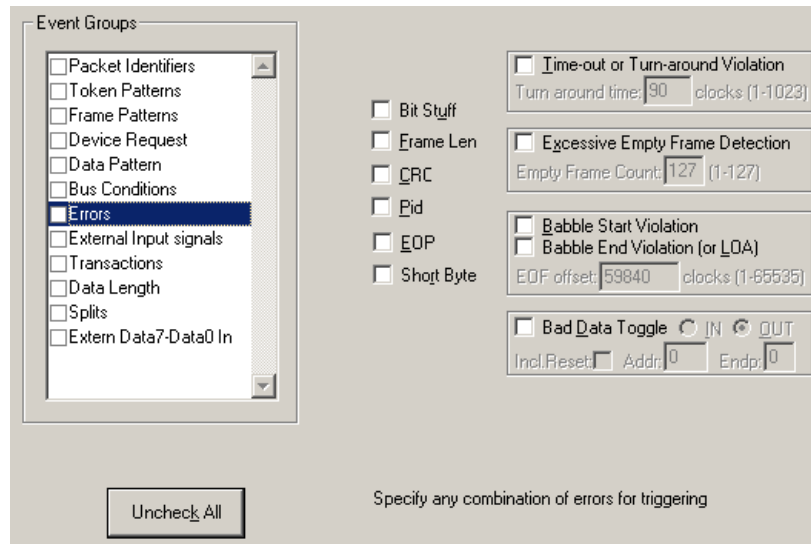


Use any combination of the listed errors as a Trigger.

Errors

- Select **Errors** under **Event Groups**.

You see the **Errors** window:



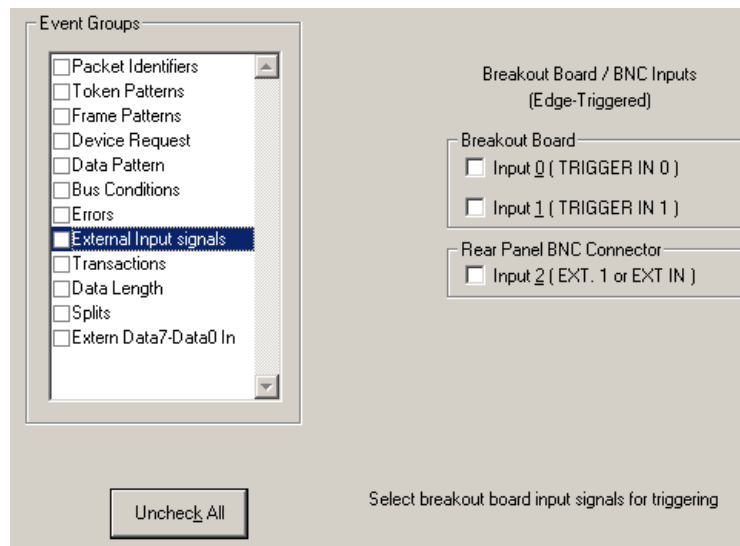
Use any combination of the listed errors as a Trigger.

- **Bit Stuffing, Frame-Length, CRC, Pid, EOP, Short Byte Checkboxes** - Select one or more of these errors to set these as the basis for triggering or filtering. A Short Byte is a condition where EOP begins on a non-byte boundary.
- **Time-out or Turn-around Violation** - Select this checkbox and then enter in the Turn-around time in the text box. This value tells the analyzer how much time should elapse before triggering/filtering. Clocks are 60 MHz (16.66 nS).
- **Excessive Empty Frame Detection** - Select this option and enter the empty frame count in the text box. This value tells the analyzer how many frames should elapse before triggering/filtering.
- **Babble Start Violation** - Select this checkbox to trigger or filter whenever the start of a packet occurs too late in a Frame.
- **Babble End Violation (or LOA)** - Select this to trigger/filter when the end of a packet occurs too late in a Frame. Enter a value in the **EOF offset** box from 1 to 65,535 clocks to indicate the size of the offset from start of previous SOF.
- **Bad Data Toggle** - Select this checkbox to trigger or filter whenever a Data0, Data1, Data2 toggle violation occurs.
 - Select **IN** or **OUT** if you only want triggering/filtering to occur only on IN or OUT transactions.
 - Check **Incl Reset** if you want the first toggle value after a reset to be considered "good" regardless of the value prior to that reset. If Unchecked, toggle state will be assumed to be preserved through the reset.
 - Select **Addr** and/or **Endp** to select Token Addresses and/or Endpoints.

External Input Signals

- Select **External Input Signals** under **Event Groups**.

You see the **External Input Signals** window:

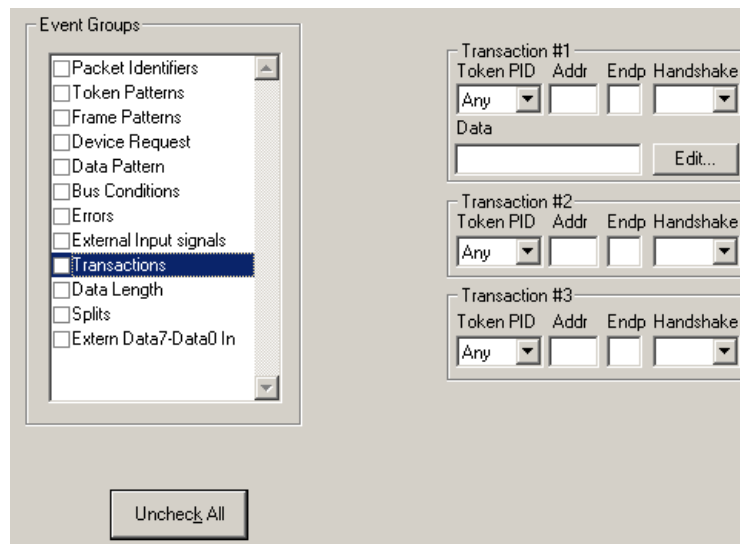


This window has three checkboxes: **Input 0 (TRIG IN 0)**, **Input 1 (TRIGGER IN 1)**, and **Input 2 (Ext. 1 or Ext. IN)**. Use any combination of the inputs as a Trigger.

Transactions

- Select **Transactions** under **Event Groups**.

You see the **Transactions** window:

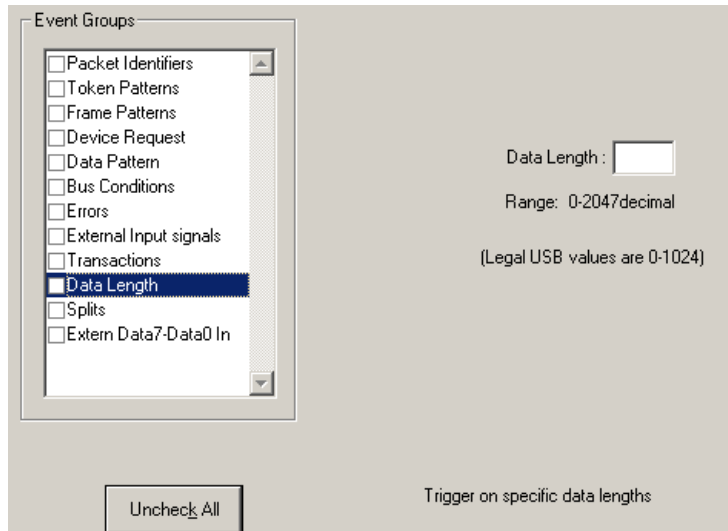


Use any combination of the listed errors as a Trigger.

Data Length

- Select **Data Length** under **Event Groups**.

You see the **Data Length** window:



To trigger on data length, enter a value between 0 and 1024 in the box marked **Data Length**.

Splits

- Select **Splits** under **Event Groups**.

You see the following window.

The screenshot shows the 'Event Groups' window. On the left, a list of event groups includes 'Splits', which is selected. Below the list is an 'Uncheck All' button. On the right, there are several configuration sections: 'Split Type' with radio buttons for 'Start', 'Complete', and 'Don't Care' (selected); 'Endpoint Type' with a dropdown menu set to 'Any'; 'Addresses' with 'Hub' and 'Port' checkboxes; 'FS Isoch Out' with a dropdown menu set to 'Any'; 'Speed (S)' with radio buttons for 'Full', 'Low', and 'Don't Care' (selected); and 'E' with radio buttons for '0', '1', and 'Don't Care' (selected). At the bottom right, the text 'Specify USB Split events for triggering/filtering' is displayed.

To trigger on splits, select a split type, and endpoint type, and an address. You can also select FS Isoch Out when you select the Isoch endpoint type.

Extern Data7 - Data0 In

- Select **Extern Data7 - Data0 In** under **Event Groups**.

The screenshot shows the 'Event Groups' window. On the left, a list of event groups includes 'Extern Data7 - Data0 In', which is selected. Below the list is an 'Uncheck All' button. On the right, there are configuration sections: 'Breakout Board Data7-Data0 (Edge-Triggered)' with 'Match' and 'Mask' fields, both labeled '(Hex)'; and at the bottom right, the text 'Trigger on Breakout Board Data7 - Data0 Inputs'.

To trigger on Extern Data 7 - Data0 In, enter values in the Match and Mask fields.

6.5 Actions Recording Options

The **Actions** screen serves as a means of setting **Triggers**, **Filters (In or Out)**, and **Counts** for the events selected in the Events window.

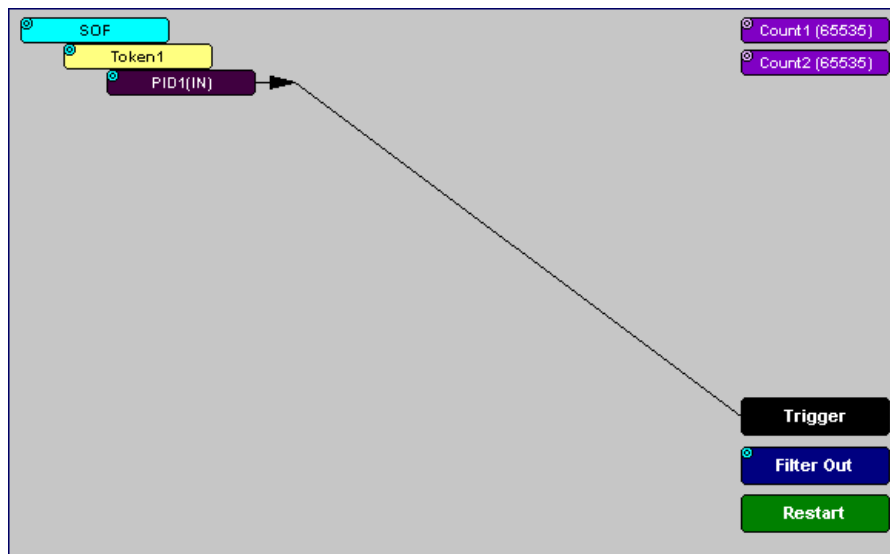
Actions Window Layout

The Actions window divides into three sections: left, center, and right.

The left section displays buttons for the events selected in the **Events** window. Events buttons tell USBTracer what it should look for during a recording. The number of Event buttons displayed depends on the number of Events selected in the Events window. If no Events were selected, no Event buttons will display.

The right side of the window displays a pair of counters called **Count1** and **Count2**. Counters provide a way for setting triggers based on a passage of a certain number of events. When an event is linked to a counter, the counter appears below the event on the left side of the screen.

The right section displays three Action buttons: **Trigger**, **Filter Out**, and **Restart**. Action buttons controls USBTracer's response to Event



conditions. For example, if an Error button is linked to a Trigger button, the Action button tells USBTracer to trigger when the error condition occurs.

The lines that join the different buttons indicate the links between Actions and Events.

6.6 Connecting Events to Actions

To create or edit links between Event buttons, Counters, and Action buttons, you click on an Event button and then click on a Counter or Action. By clicking on an Event, you will cause an arrow to appear that connects the Event button to your mouse. When you click on a Counter or Action button, the arrow becomes anchored to the selected button.

To make an association,

Step 1 Select one or more Events from the Events window.

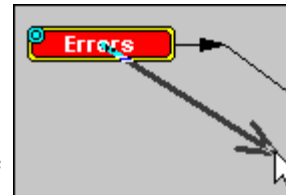
Step 2 Open the Actions window.

Step 3 Click the left mouse button on an Event button such as **Errors**.

The elastic arrow appears.

Step 4 Point your mouse at the desired Action button.

The elastic arrow will move with your mouse pointer.

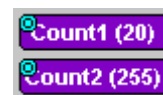


Step 5 With the pointer over an Actions button, click again the left mouse button again.

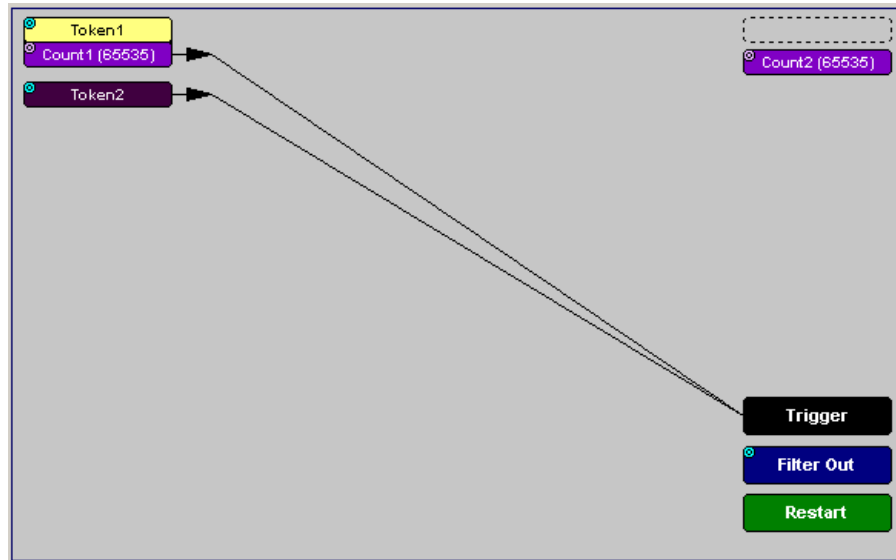
The arrow is anchored to the Action button. replaced with a black line connecting the Event button to the Action button.

6.7 Connecting Counters to Events

The right side of the Actions window contains two Counter buttons marked **Count1** and **Count2**. These counters provide a way of triggering after a number of events have passed, such as



"Trigger after the 20th IN packet." When an event is linked to a counter, the



counter appears under that event on the left side of the Actions window. Counters have blue dots in their top left corner that provide access to pop-up menus. The menus provide the means of setting the counter value. The counter can be set between 1 and 65,535.

To connect an event to a counter,

Step 1 Open the Event window and select an Event.

Selecting an Event will cause an Event button to automatically appear in the Actions window.

Step 2 Open the Actions window.

The window will display an Event button that is connected to the Trigger button on the right via a line.

Step 3 Click the Event button.

An arrow will appear that will connect the Event button to your mouse pointer.

Step 4 Click one of the two counters.

The Event will automatically connect to the Counter button and then connect to the Trigger button.

Step 5 Click the blue dot in top left corner of your selected counter.

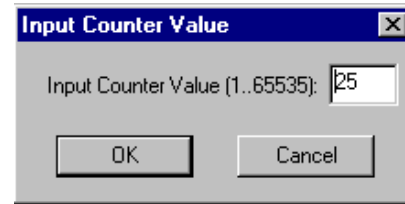
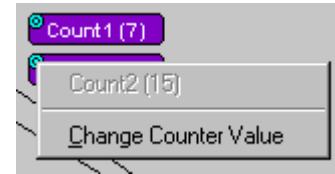
A pop-up menu will open. This menu lets you configure the counter.

Step 6 Choose **Change Counter Value**.

A dialog box will open.

Step 7 Set the counter to a value of your choice, then click OK.

The dialog box will close and the Counter button will display your selection.

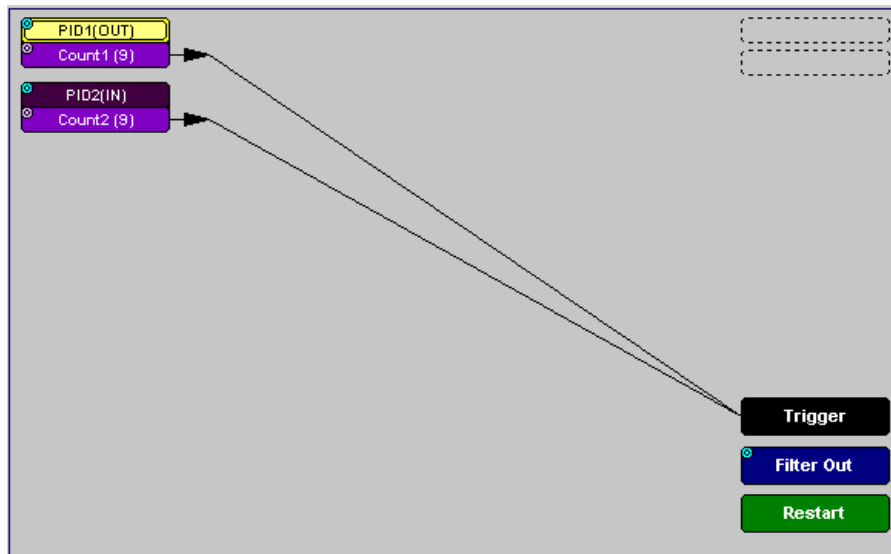


Setting Multiple Conditions with Counters

You can create multiple event conditions by linking two counters to two or more events.

Linking Two Events to Two or More Counters - If an Event is linked to **Count1** and a second event is linked to **Count2**, it creates an "or" statement. This statement reads "Trigger when Count1 OR Count2 has reached their specified values."

This example reads "Trigger when Count1 has counted 9 Out packets or Count2 has counted 9 In packets."



6.8 Using Action Buttons

The right section of the Actions window displays three Action buttons labeled **Trigger**, **Filter Out/In**, and **Restart**.



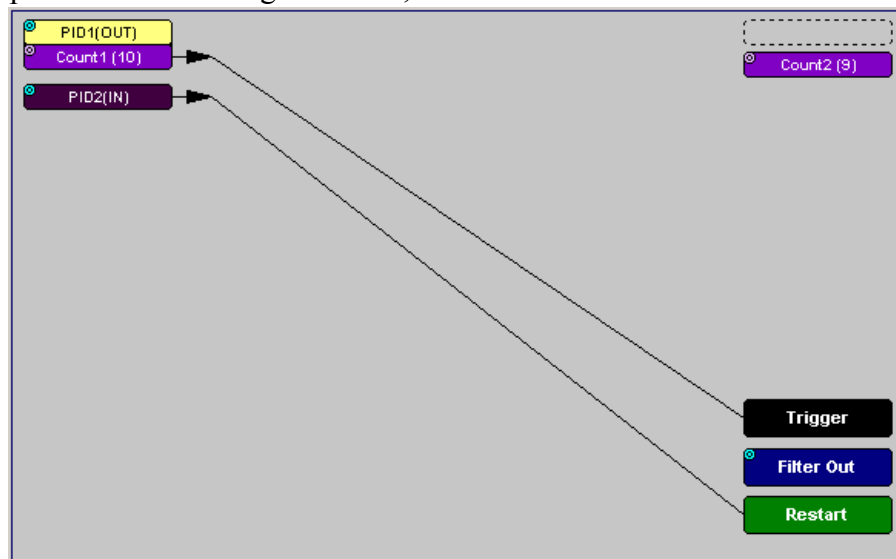
Trigger

The **Trigger** button enables event triggering.

Restart

Restart causes counters to restart counting when a certain event occurs. When Restart is used there will be at least two links on the screen: one between an Event and the Restart button, and a second between an Event, Counter and an Action button.

The screenshot below gives an example. The screen below reads "Look for IN and Out packets. When 10 OUT packets, trigger. However, if an IN packet occurs during the count, restart the count."



Filter Out/In

The **Filter In/Out** button causes events to be filtered in or out of the recording. **Filter Out/In** toggles between "**Filter Out**" and "**Filter In**".

- **Filter In** records ONLY those packets related to the specified event.
- **Filter Out** records all packets EXCEPT those related to the

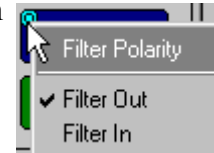
specified event.

Note Only events from the Token, Frame Event, Device Request, and Transaction Groups can be used to filter a recording.

To filter in or out Token, Frame Event, Device Request, and/or Transaction Group events from a recording,

Step 1 Click the blue dot on **Filter Out**. (Note: the button may say **Filter In** depending on the last action specified.)

You see the **Filter Out/In** menu:



Use this menu to toggle the selection between **Filter Out** and **Filter In**.

Step 2 Select "**Filter In**".


The button changes to read "Filter In".

6.9 Other Actions: External Output Signals

Most Event buttons can be configured to trigger external output signals.

If a blue dot is present in the Event button, it means that a menu is present with three options for configuring external output signalling:

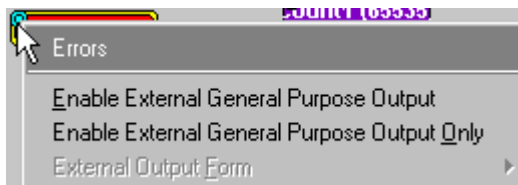
- Enable External General Purpose Output
- Enable External General Purpose Output only
- External Output Form

If you choose to enable External Output signalling, a small blue arrow  will project from the Event button. This arrow is a reminder that External Signals have been set.

To enable or disable external trigger output,

Step 1 Click the blue dot on an Event button.

A menu similar to the one below will open. Your menu may say "Disable" instead of "Enable."



Step 2 Select "**Enable External Trigger Output**" (or "**Disable External Trigger Output**" if that is the choice presented.)

If you have chosen "**Enable External Trigger Output**", a small arrow will appear on the right side of the button. This arrow indicates that a condition has been set for creating an external output signal. Choosing "**Disable External Trigger**" will cause the arrow to disappear.



Enabling High Pulse, Low Pulse or Pulse Toggle Signal Outputs

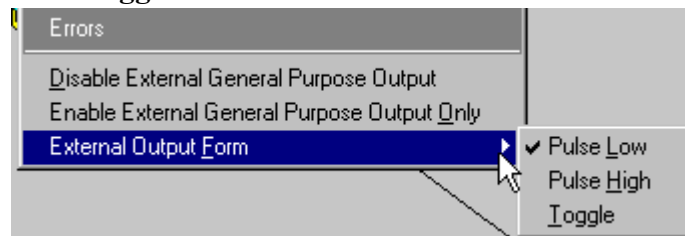
If you chose "**External Trigger Output**" in the previous steps, then an additional choice will appear in the Blue Dot Menu called "**Enable External Trigger**". This option lets you control the output signal.

Step 1 Click the blue dot on an event button that has a small arrow attached to it like the one shown above.

A Blue Dot Menu will open. "**External Trigger Form**" should be a choice available. If it is not, you will need to choose "**Enable External Trigger**" and then reopen the menu.

Step 2 Choose "**External Trigger Form**"

A menu will appear with choices for "**Pulse Low**", "**Pulse High**", and "**Toggle**".



Step 3 Choose an option not currently selected.

The menu closes.

Step 4 Reopen the menu.

Note that your new selection is now checked.

6.10 Saving Recording Options

To complete your Recording Options settings, use the features at the bottom of the **Recording Options** screen. These features remain the same no matter which of the three Recording Options screens you are working in.

- Click **Save** to save the currently specified Recording Options for use in future recording sessions. Any file name can be specified, though use of the **.rec** is recommended; if no extension is specified, **.rec** is added by default.


- Click **Load** to load a previously saved *.rec file, thus restoring a previous set of Recording Options.
- The **Save as Default** function is equivalent to the **Save** function, specifying the file name **default.rec**. Whenever you start up the Analyzer, it automatically loads the **default.rec** file if one exists.
- Click **OK** to apply any changes and close this dialog box.
- Click **Cancel** to cancel any immediate changes you have made and exit the Recording Options menu.


6.11 Recording Bus Data

To start recording USB traffic once the appropriate Recording Options have been set perform the following steps. Note: If you have inserted any event triggers, be sure to select *Event Trigger* under the General tab in the Recording Options dialog box.

Step 1 Select **Start** under **Record** on the Menu Bar

OR


Click  on the Tool Bar.


Your recording session can continue until it has finished naturally or you may need to stop manually by clicking  on the Tool Bar, depending on how you set the Recording Options.

To manually stop recording,

Step 2 Select **Stop** under **Record** on the Menu Bar

OR

Click  on the Tool Bar.

Click  again during the uploading to upload only a portion of the recorded memory.


Note The manual Stop Recording feature is primarily of use when recording low-speed traffic, which can take a long time to fill the recording buffer.

When the recording session is finished, the bus traffic is saved to the hard drive as a file named **data.usb** or whatever name you assign as the default filename.

To save a current recording for future reference,

Step 3 Select **Save As** under **File** on the Menu Bar.

OR

Click  on the Tool Bar.

You see the standard **Save As** screen.

Step 4 Give the recording a unique name and save it to the appropriate directory.

6.12 Merging Trace Files

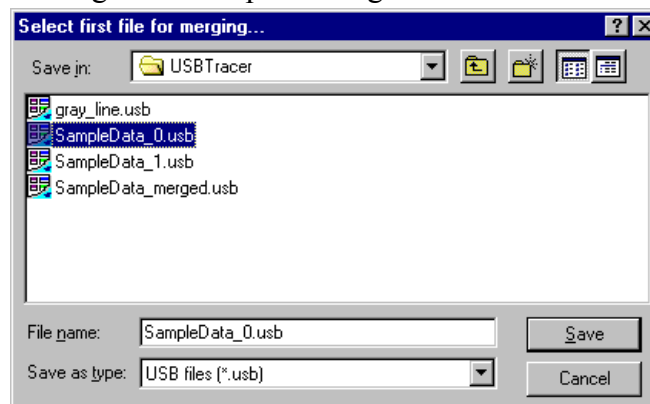
It is possible to merge a Classic-Speed trace file with a Hi-Speed trace file using the "Merge Trace File" command under the File menu. This option only works with files that were created simultaneously through a single recording session. If the files were recorded during separate recording sessions, USBTracer will generate an error message and prevent the merge from completing.

Note USBTracer is capable of merging High Speed and Classic Speeds traffic into a single merged file if the **Auto-Merge 2 Channel Trace Files** option is checked in the Recording Options dialog box. See page 49 for details.

To merge two trace files,

Step 1 Select
File>Merge Trace Files

A dialog box will open asking for the first source file.



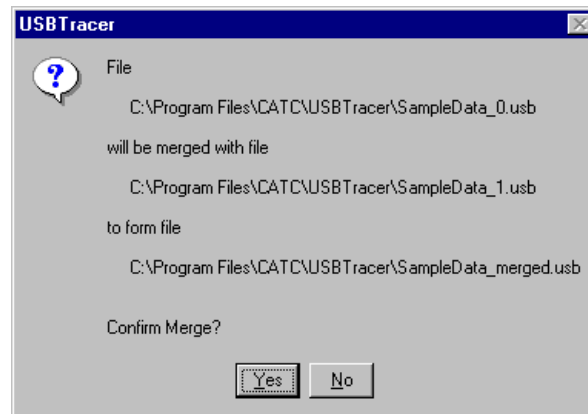
Note The **Merge Trace Files** command can be run with or without a trace file open on the screen. The merge process will ignore the open file.

Step 2 Select the first trace file to be merged, then click Save.

Note It does not matter which of the two trace files is first selected so long as both were recorded in the same session.

Step 3 Select the second trace file to be merged, then click Save.

At this point, you will be asked to confirm your choices.



Step 4 Click Yes.

At this point the two files will be merged into the new file data_merged.usb.

Note If you attempt to merge two files that were recorded in separate recording sessions, the following error message will appear:




7. Display Options

Use the **Display Options** menu to specify the way trace information is displayed.

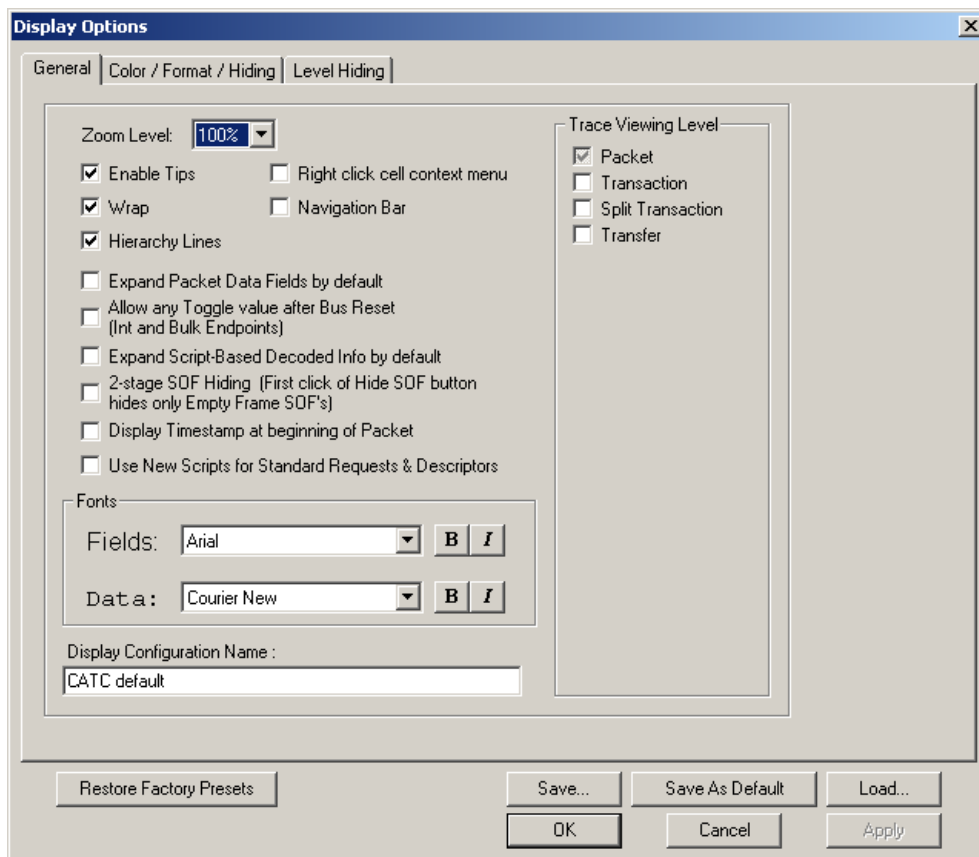
To open the **Display Options** menu,

- Select **Display Options** under **Setup** on the Menu Bar

OR

- Click  on the Tool Bar

You see the **Display Options** window:



Use the General Display Options to specify the basic appearance of a Trace View.

7.1 General Display Options

Use the General Display Options to specify the basic appearance of a Trace View.

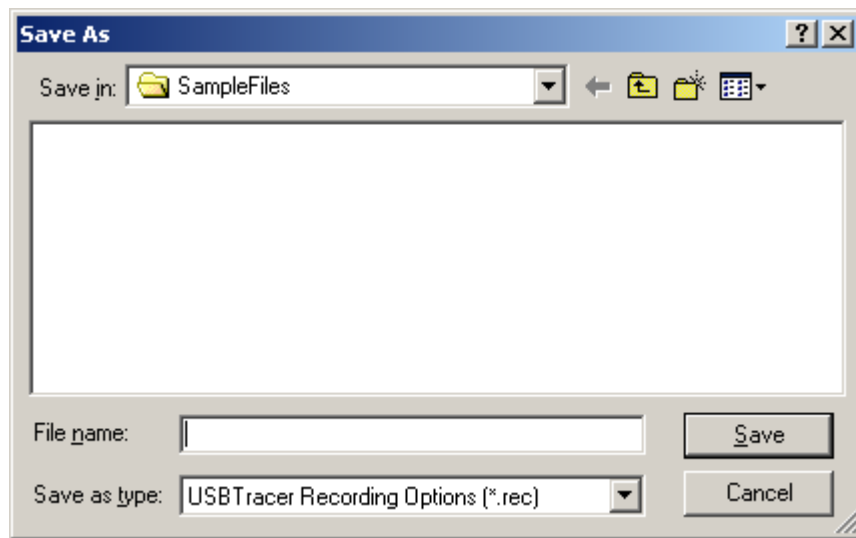
- **Zoom Level:** Adjustable in discrete increments from 10% to 200% percent.
- **Enable Tool Tips:** Select to enable tool tips with explanation text to pop up when you position your cursor over various fields in the Trace View.
- **Right click cell context menu:** This option activates the right mouse button so that when clicked it displays pop-up menus in the trace. Normally, the left mouse button is used to display context-sensitive pop-up menus.
- **Navigation Bar:** This option opens the Navigator.
- **Wrap:** This option causes the traffic to wrap within the display.
- **Hierarchy Lines:** Displays lines along the left side of the trace window showing the hierarchical relationship between Packets, Transactions, Split Transactions, and Transfers.
- **Expand Packet Data Fields by Default:** Causes the trace to display packet data fields in expanded mode.
- **Allow Any Toggle Value after Bus Reset:** After a Bus reset, the toggle state of an endpoint might or might be reset. Depending on the device you are using, selecting this option might avoid showing a toggle violation where it is not really an error.
- **Expand Script-Based Decoded Info by default:** Causes the trace view to display decoded transfer fields in expanded mode when the file is first loaded.
- **2-stage SOF Hiding:** Allows three possibilities for SOF Viewing: ALL SOF's visible, All SOF's Hidden, and a third mode which only displays SOF's that start frames which contain some endpoint traffic. This is also referred to as Hiding Empty Frames.
- **Display Timestamp at beginning of Packet:** This moves the Timestamp field to the left side of the screen, where it aligns in a column. This makes it easier to visually compare the value against the previous or next packet's timestamp.

- **Use new Std Reqs & Descriptors Scripts:** To support decoding of the Video Class, it was necessary to re-write the mechanism by which Standard Requests and Descriptors are decoded. This is because most of the Video configuration information is passed in the Standard GET_DESCRIPTOR (Config) descriptor bundle, along with the endpoint and interface descriptors, etc. For non-Video traffic, leave this unchecked so the existing Hid, Audio, and other descriptors get decoded by the legacy decoder files.
- **Trace Viewing Level:** Allows packets to be displayed as
 - Packets
 - Transactions
 - Split Transactions
 - Transfers
- **Fonts:** Lets you define the appearance of Field and Data text.
- **Display Configuration Name:** A Comment field associated with the *.opt file containing the current Display Options values. You can also create and store your unique Display Options for future use.

To create a new Display Options file,

- Step 1** Enter a comment for the new file in the **Display Configuration Name** field.
- Step 2** Click **Save...**

You see the **Save As** window.

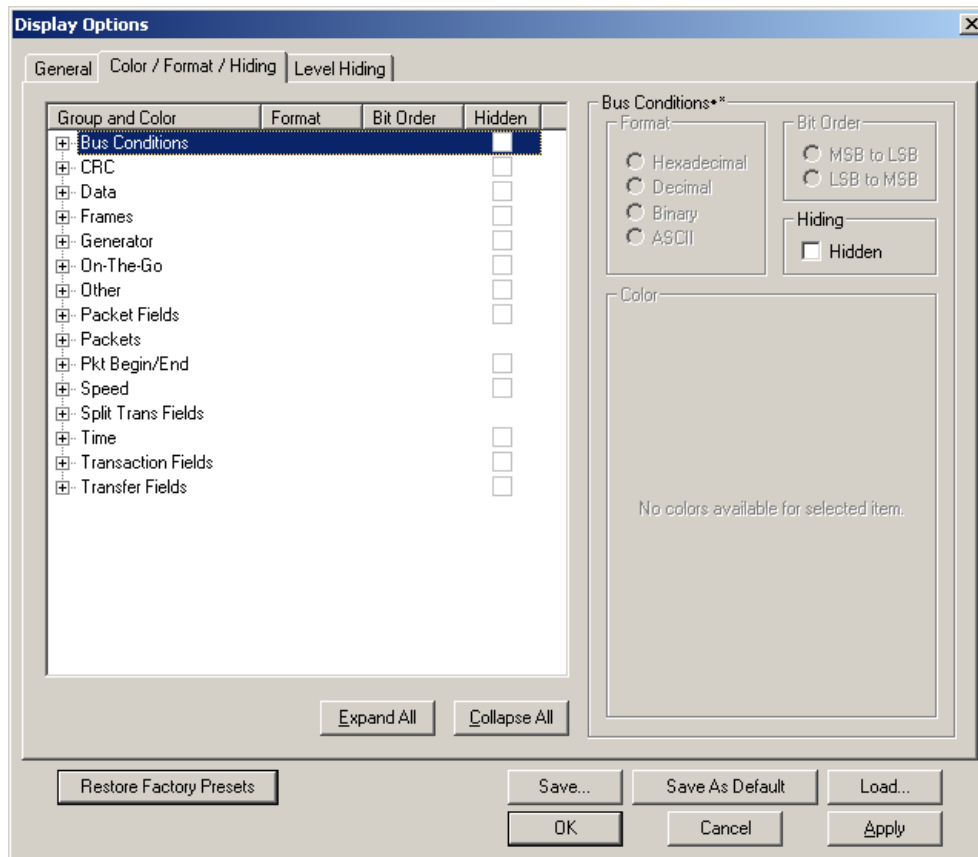


Step 3 Specify a filename (*.opt).

Step 4 Click **Save**.

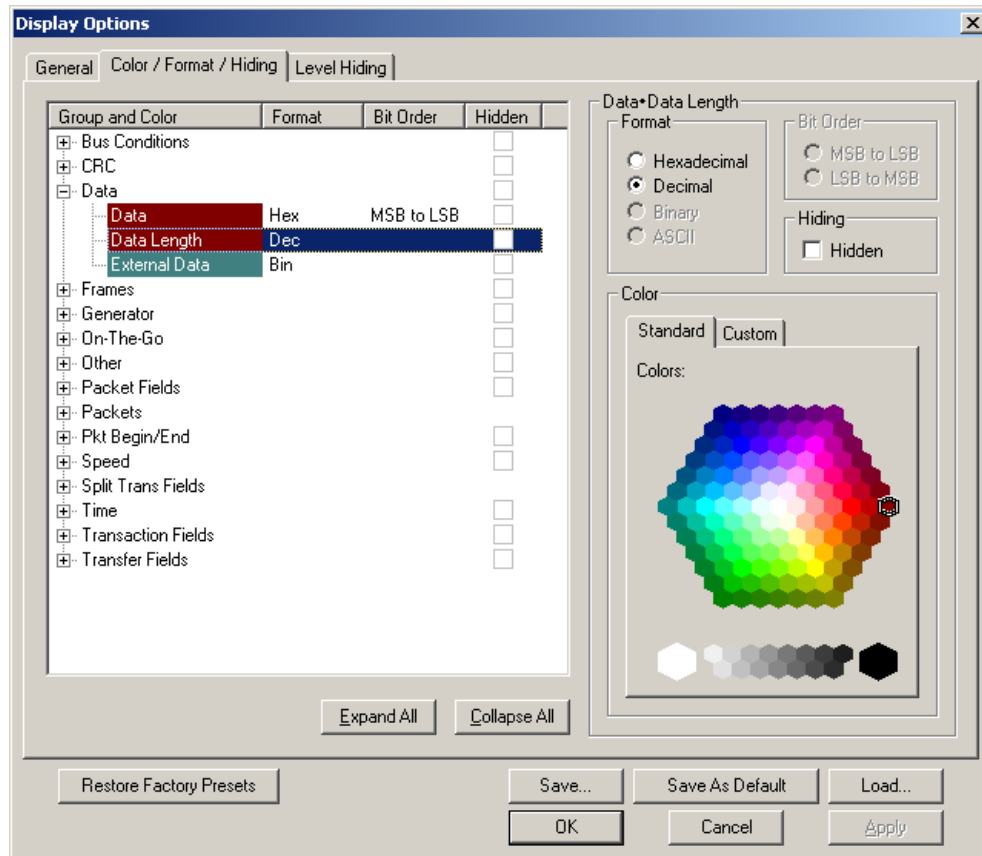
7.2 Color/Format/Hiding Display Options

Select the Color/Format/Hiding tab in the Display Options window.



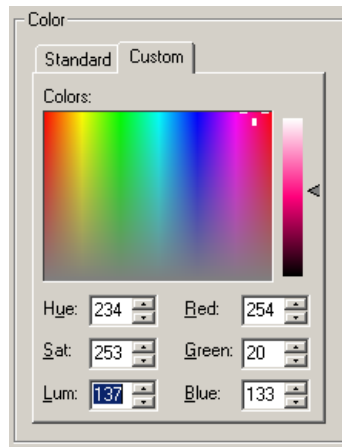
7.3 Color Display Options

To specify colors in the display, select an item in the Group and Color column and make the desired color changes in the color pallet screen.



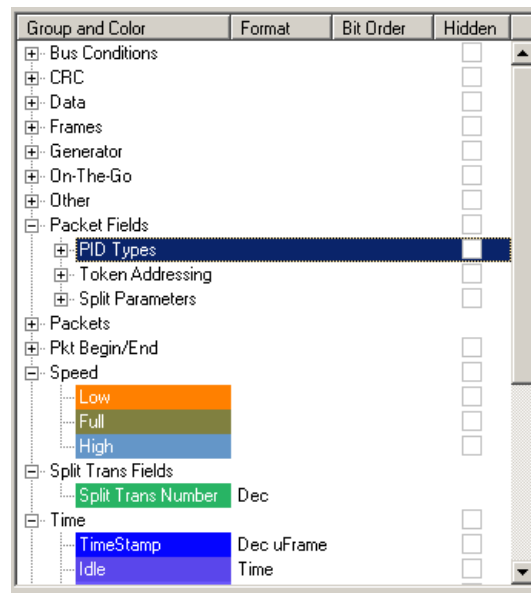
Use this menu to customize the colors associated with each field in the Trace View. You can experiment with this option to achieve the color combination best suited to a particular graphic system. A brighter color might be appropriate for a specific field that should stand out in the display (e.g. the PID Types).

You can also customize the colors by using the options in the Custom tab.

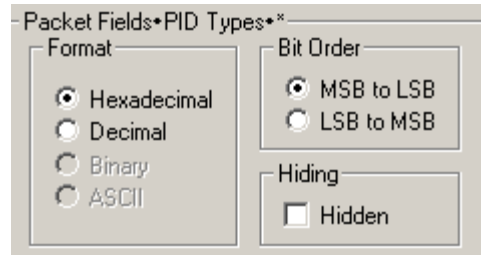


7.4 Formats Display Options

Select an item under the Group and Color column to enable the formats radio buttons in the Format section. The format types changes with respect to the item you select under the Group and Color column. For example, if you select an item under Packet Fields, you get the following:



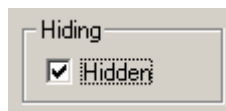
The following formats are available:



Not every format is available for every item.

7.5 Hiding Display Options

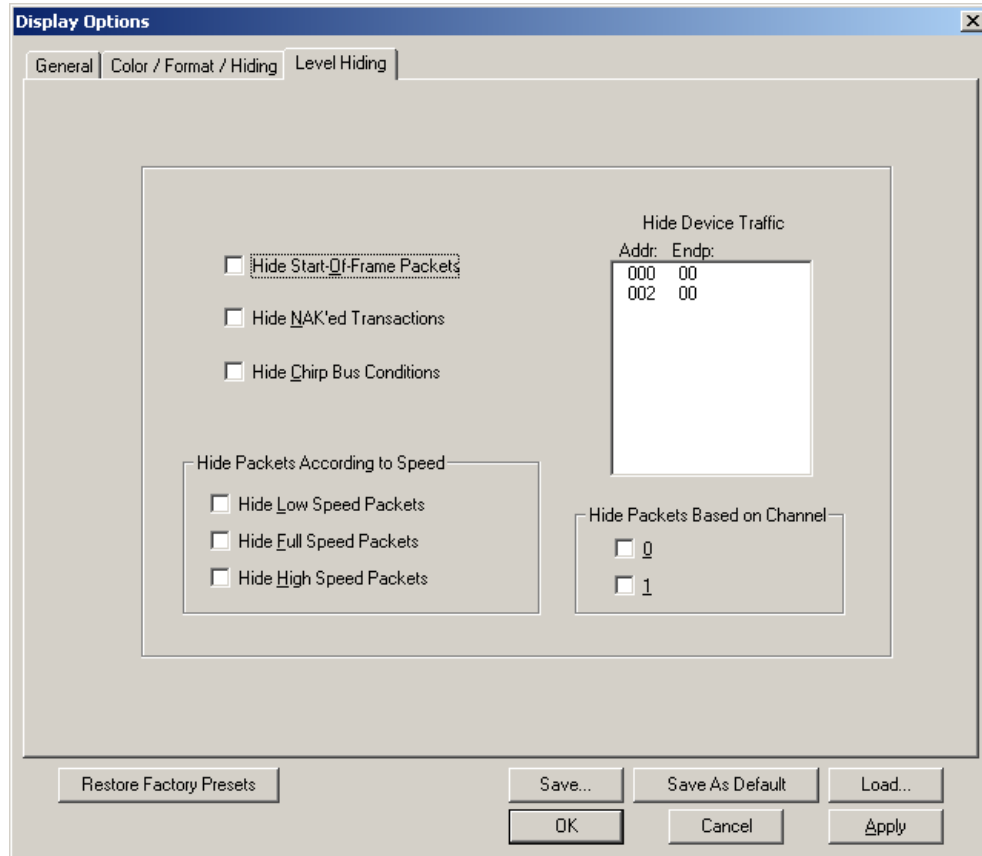
To hide one or more fields from the trace, select the appropriate item from the Group and Color column, click the Hidden checkbox, and click the Save button.



You can also hide packets and transactions from a trace, selecting the desired options from the checkboxes. You can hide SOFs, NAKs, High, Full or Low Speed packets, traffic from one or both recording channels, and Address and Endpoint.

7.6 Level Hiding Display Options

The Level Hiding tab allows you hide whole packets such as Start of Frame packets. Select the Level Hiding tab.



7.7 Saving Display Options

To complete your Display Options settings, use the features at the bottom of the **Display Options** screen. These features remain the same no matter which of the four Display Options screens you are working in.

- Click **Save** to save the currently specified Display Options for use in future sessions. Any file name can be specified, but you must use the **.opt** extension. If no extension is specified, **.opt** is added by default.
- Click **Load** to load a previously saved ***.opt** file, thus restoring a previous set of Display Options.

- The **Save as Default** function is equivalent to the **Save** function, specifying the file name **default.opt**. Whenever you start up the Analyzer, it automatically loads the **default.opt** file if one exists.
- Click **OK** to apply any changes you have made to Display Options and close this dialog box.
- Click **Cancel** to cancel any immediate changes you have made and exit the Display Options menu.
- Click **Apply** to apply your changes while keeping the Display Options window open.


8. Reading a Trace

Packet	Dir	H	SOF	Frame #	CRC5	Pkt Len	Idle	Time Stamp
0	-->	S	0xA5	1156.?	0x1C	12	124.767 μ s	00000.4056 3910
Packet	Dir	H	SOF	Frame #	CRC5	Pkt Len	Idle	Time Stamp
1	-->	S	0xA5	1156.?	0x1C	14	124.767 μ s	00000.4057 3908
Packet	Dir	H	SOF	Frame #	CRC5	Pkt Len	Idle	Time Stamp
2	-->	S	0xA5	1156.?	0x1C	12	124.800 μ s	00000.4058 3908
Packet	Dir	H	SOF	Frame #	CRC5	Pkt Len	Idle	Time Stamp
3	-->	S	0xA5	1156.?	0x1C	12	124.800 μ s	00000.4059 3908
Packet	Dir	H	SOF	Frame #	CRC5	Pkt Len	Idle	Time Stamp
4	-->	S	0xA5	1156.?	0x1C	12	124.767 μ s	00000.4060 3908

8.1 Trace View Features

- The USBTracer packet view display makes extensive use of color and graphics to fully document the captured traffic.
- Packets are shown on separate rows, with their individual fields both labeled and color coded.
- Packets are numbered (sequentially, as recorded), time-stamped (with a resolution of 16.67 ns), and highlighted to show the transmitted speed (low-speed, full-speed or high-speed).
- Display formats can be named and saved for later use.
- Pop-up Tool Tips annotate packet fields with detailed information about their contents.
- Data fields can be collapsed to occupy minimal space in the display (which can in turn be zoomed in and out to optimize screen utilization).
- The display software can operate independent of the hardware and so can function as a stand-alone Trace Viewer that may be freely distributed.
- High Speed SOFs display Microframes (shown below.)

Packet	Dir	H	SOF	Frame #	CRC5	Pkt Len	Idle	Time Stamp
0	-->	S	0xA5	1156.?	0x1C	12	124.767 μ s	00000.4056 3910
Packet	Dir	H	SOF	Frame #	CRC5	Pkt Len	Idle	Time Stamp
1	-->	S	0xA5	1156.?	0x1C	14	124.767 μ s	00000.4057 3908



Microframes

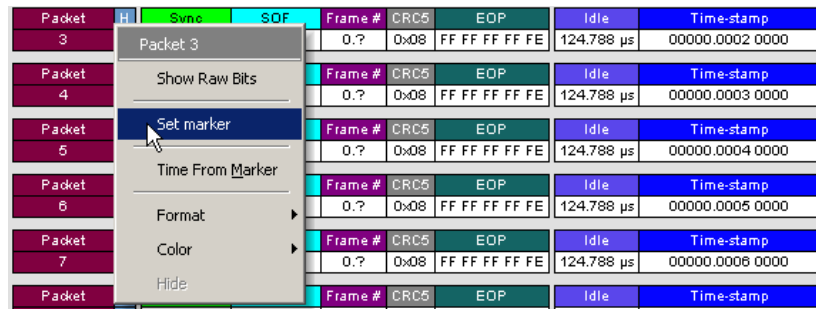
8.2 Set Marker

You can define a unique Marker for each packet.

To place a marker on a packet,

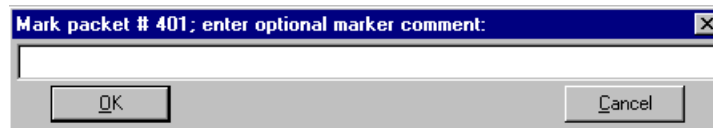
Step 1 Left-click on **Packet #** for the packet you wish to mark.

You see the **Packet** menu:



Step 2 Select **Set Marker**.

You see the **Edit Marker Comment** window where you can enter a unique comment about this packet.:



Step 3 Enter your comment.

Step 4 Click **OK**.

A marked packet is indicated by a vertical red bar along the left edge of the packet # block:

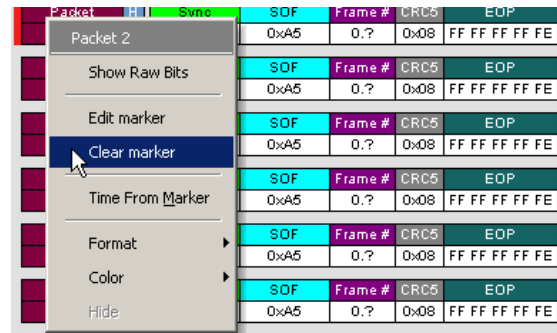
Packet #	F	Sync	SETUP	ADDR	ENDP	CRC5	EOP	Idle
398	S	00000001	0xB4	2	0	0x15	2.50	2

8.3 Edit or Clear Marker

To clear or edit the comments associated with a packet marker,

Step 1 Left-click on **Packet #** for the chosen packet.

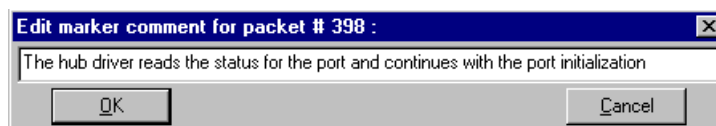
You see the **Packet** menu:



To edit the Marker Comment,

Step 2 Select **Edit Marker Comment**.

You see the **Edit marker comment** window:



Step 3 Edit the comment as desired.

Step 4 Click **OK**.

To clear a Marker,

Step 5 Click **Clear Marker**.

The vertical red Marker bar disappears.

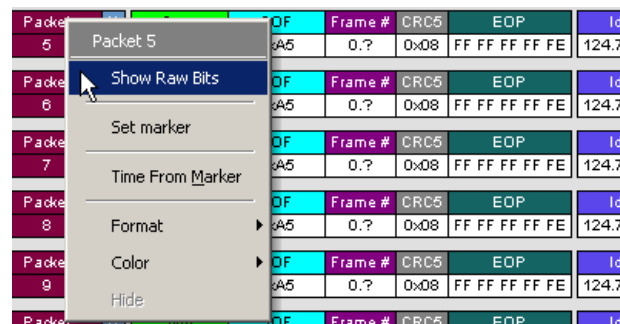
8.4 View Raw Bits

You can expand a specific packet to view the raw bits in detail.

To view the raw bits,

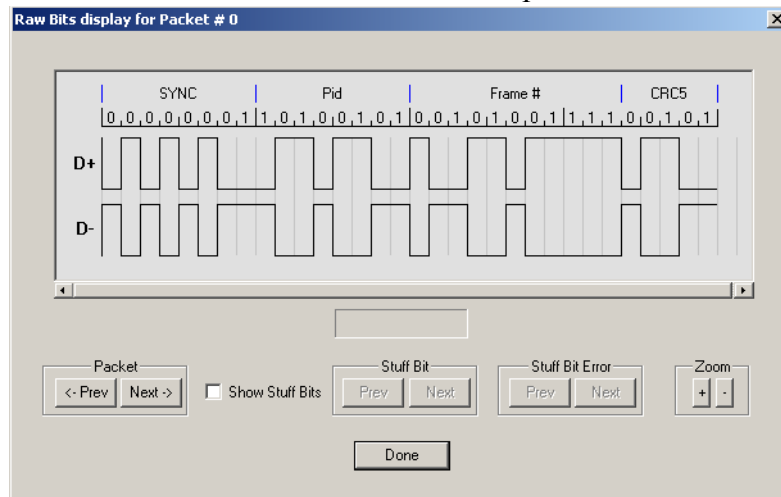
Step 1 Left-click on **Packet #** for the packet you wish to view.

You see the **Packet** menu:



Step 2 Select **Show Raw Bits**.

You see the Raw Bits View for that packet:



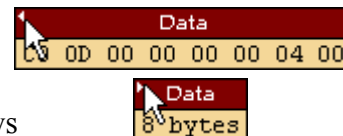
Along the top of the Raw Bits View is a linear strip of the logical bit values with corresponding field demarcations. Bit stuffing is displayed in color. Below the logical bit values is a representation of the D+/D- signaling complete with NRZ encoding. A scroll bar assists in navigation of larger packets. Two buttons under the label **Packet** are used to view previous or next packets. Two buttons under the label **Zoom** allow you to zoom in or out on packets.

8.5 Expanding and Collapsing Data Fields

You can expand a Data field to view it in greater detail or collapse it when you want a more compact view.

Using the Expand/Collapse Data Field Arrows

Data Fields can be easily expanded or collapsed by clicking the small triangular arrows on the left side of the data field.



Double-Clicking to Expand/Collapse Data Fields

Data fields can also be expanded or collapsed by double-clicking anywhere in the data field.

Expanding or Collapsing All Data Fields

Expand or collapse all data fields by holding down the button for more than a second.

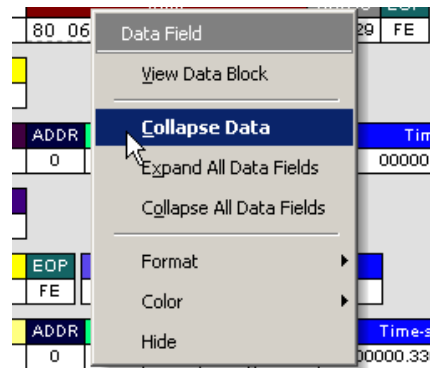
Using the Data Field Pop-up Menus

Data fields can be expanded or collapsed by clicking in a data field and selecting **Expand Data** or **Collapse Data** from the pop-up menu.

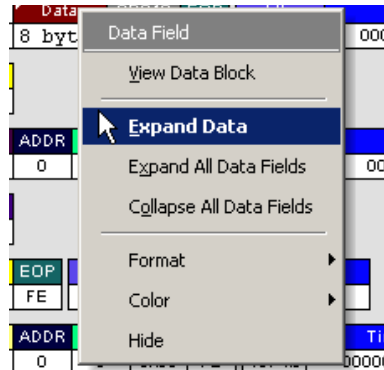
To expand and collapse data using the menu,

- Step 1** Left-click on **Data** in the Data packet you want to expand or collapse.

If your Data Trace View is currently expanded, you see the **Collapse Data** menu:



If your Data Trace View is currently collapsed, you see the **Expand Data** menu:



- Step 2** Select the desired **Expand Data** or **Collapse Data** menu item.

The Trace View is repositioned with the selected packet(s) adjusted in the format you have specified.

Expand or Collapse All Data Fields

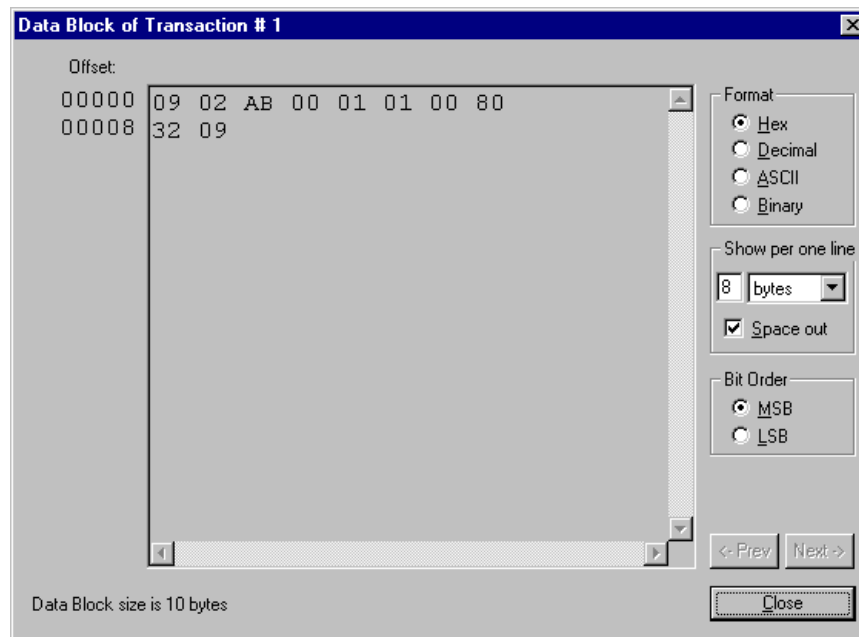
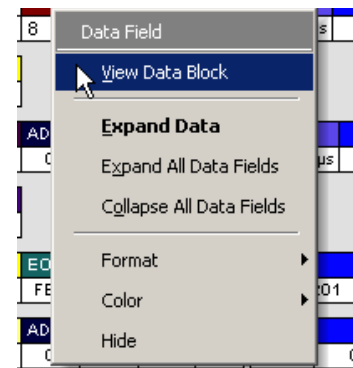
All data fields can be expanded or collapsed by selecting **Expand All Data Fields** or **Collapse All Data Fields** from the data field pop-up menus.

8.6 View Data Block

The data field pop-up menu has an option for viewing the raw bits in a data field. To view these bits,

- Step 1** Click in the data field to open the data field pop-up menu.
- Step 2** Select View Data Block.

The following dialog box will open.



The View Data Block window has options for displaying the raw bits in different formats:

Format - Lets you display data in Hex, Decimal, ASCII or Binary formats

Show Per Line - Lets you control how many bits are displayed per line

Bit Order - Most Significant Bit, Least Significant Bit

8.7 Pop-up Tool-tips


Many fields within the trace will display pop-up tool-tips when the mouse pointer is suspended over them. These tips provide added details about the field.

SETUP	ADDR	ENDP	CRC5	EOP	Idle	Time Stamp
0xEB	2	0	0x15	233 ns	183 ns	00004.2833 6330

Starts SETUP transaction to a control pipe


8.8 Hide SOF Packets

You can hide Start-of-Frame (SOF) packets that may be uninteresting in a given context from a Trace View by clicking the Hide SOF Packets button on the Tool Bar:


- Click  to hide all SOF packets.

8.9 Hide NAKs


You can hide NAKs that may be uninteresting in a given context from a Trace View by clicking the Hide NAKs button on the Tool Bar:

- Click  to hide all NAK packets.

8.10 Hide Devices

Click the  button to open a menu for select any address/endpoint combination to be hidden. Hit the Ctrl key to select multiple values. This menu can be moved from its default location and floated in the trace window for later use.

8.11 Hide Chirps


Click the  button to hide any Chirped-J or Chirped-K packets recorded in a USB trace.

8.12 Switch to Transactions View

A **Transaction** is defined in the USB specification as the delivery of service to an endpoint. This consists of a token packet, an optional data packet, and an optional handshake packet. The specific packets that make up the transaction vary based upon the transaction type.

The program default display mode is Packet View. Before you can view decoded transactions, you must switch from Packet View to Transactions View.

To select Transactions View,

Step 1 Click  to on the toolbar.

The Trace View screen is re-drawn to display Transactions.

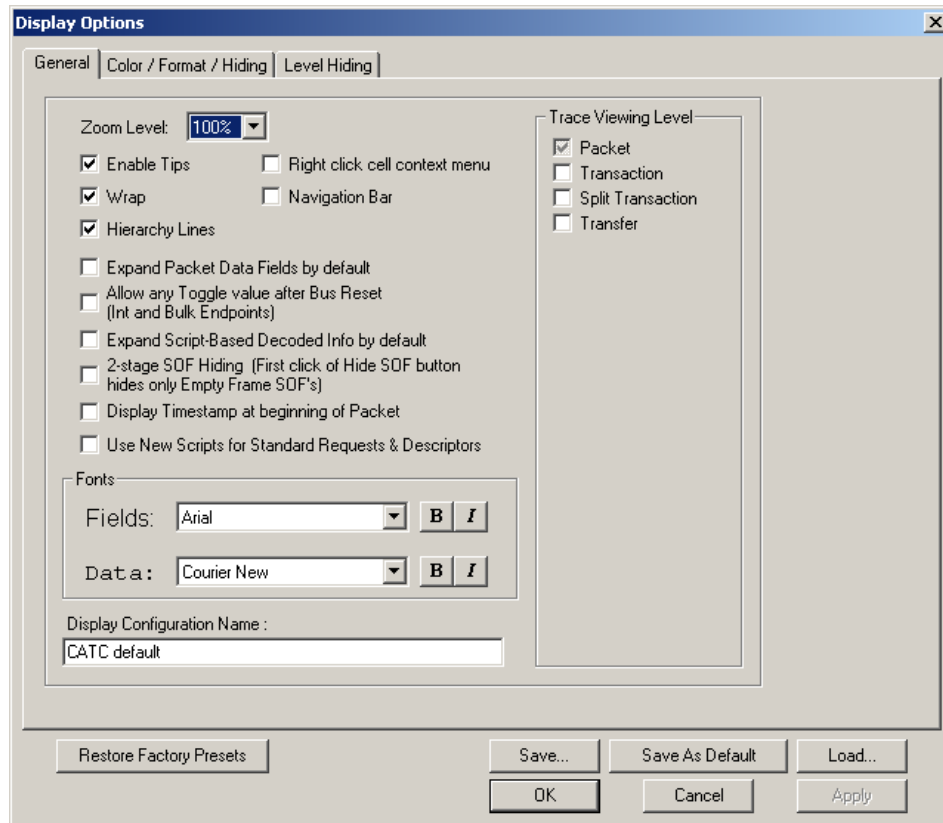
Transaction	F	OUT	ADDR	ENDP	T	Data	ACK	Time		
1336	S	0x87	1	0	0	8 bytes	0x4B	999.717 μs		
Transaction	F	OUT	ADDR	ENDP	T	Data	ACK			
1337	S	0x87	1	0	1	8 bytes	0x4B			
Packet	Dir	F	Sync	OUT	ADDR	ENDP	CRC5	EOP	Idle	Time Stamp
6946	-->	S	00000001	0x87	1	0	0x17	233 ns	183 ns	00011.6037 1516
Packet	Dir	F	Sync	DATA1	Data	CRC16	EOP	Idle	Time Stamp	
6947	-->	S	00000001	0xD2	8 bytes	0x4FB5	233 ns	500 ns	00011.6037 1701	
Packet	Dir	F	Sync	ACK	EOP	Time	Time Stamp			
6948	<--	S	00000001	0x4B	250 ns	988.000 μs	00011.6037 2225			
Transaction	F	OUT	ADDR	ENDP	T	Data	ACK	Time		
1338	S	0x87	1	0	0	8 bytes	0x4B	999.800 μs		

Note This menu selection will display a checkmark next to **Show USB transactions** when you have selected it. When you want to switch back to Packet View mode, right-click anywhere in the trace window and then left-click on **Show USB transactions**.

You can also switch to Transaction View from the Menu Bar:

Step 1 Select **Display Options** under **Setup**.

You see the **Display Options General** window:



Step 2 Check Transactions.

Step 3 Click **OK**.

8.13 View Decoded Transactions

Once you set Display Options, the Trace View screen is re-drawn to display decoded transactions in the colors and format you selected.

Transaction	F	OUT	ADDR	ENDP	T	Data	ACK	
1337	S	0x87	1	0	1	8 bytes	0x4B	

Packet	Dir	F	Sync	OUT	ADDR	ENDP	CRC5	EOP	Idle	Time Stamp
6946	-->	S	00000001	0x87	1	0	0x17	233 ns	183 ns	00011.6037 1516

Packet	Dir	F	Sync	DAT1	Data	CRC16	EOP	Idle	Time Stamp
6947	-->	S	00000001	0x02	8 bytes	0x4FB5	233 ns	500 ns	00011.6037 1701

Packet	Dir	F	Sync	ACK	EOP	Time	Time Stamp
6948	<--	S	00000001	0x4B	250 ns	998.000 μs	00011.6037 2225

Transaction	F	OUT	ADDR	ENDP	T	Data	ACK	Time
1338	S	0x87	1	0	0	8 bytes	0x4B	999.800 μs

Transaction	F	OUT	ADDR	ENDP	T	Data	ACK	Time
1339	S	0x87	1	0	1	8 bytes	0x4B	999.800 μs

Transaction	F	OUT	ADDR	ENDP	T	Data	ACK	
1340	S	0x87	1	0	0	8 bytes	0x4B	

Packet	Dir	F	Sync	OUT	ADDR	ENDP	CRC5	EOP	Idle	Time Stamp
6958	-->	S	00000001	0x87	1	0	0x17	250 ns	166 ns	00011.6061 1481

Packet	Dir	F	Sync	DAT0	Data	CRC16	EOP	Idle	Time Stamp
6959	-->	S	00000001	0xC3	8 bytes	0x155D	233 ns	483 ns	00011.6061 1666

When you instruct the Analyzer to display USB transactions, the components of each transaction are collected from the current recording and are grouped and indented below each decoded transaction. Each row shows a transaction with a unique numeration, a label, and color-coded decoding of important data.

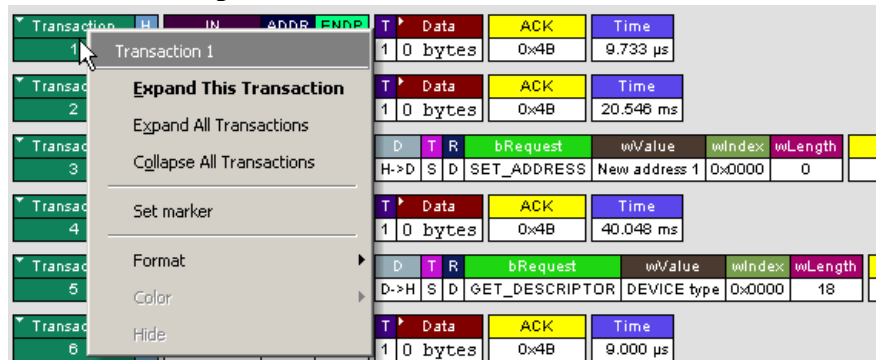
Expanded and Collapsed Transactions

You can expand a specific transaction to view its parts, which are grouped and indented below the transaction.

To expand a transaction,

Step 1 Left-click on the transaction number you wish to view.

You see the **Expand Transaction** menu:



Step 2 Select **Expand This Transaction**.

The screen displays the selected transaction in expanded format.

Note The Expand/Collapse transaction feature operates as a toggle: when one format is active, the other appears as an option on the Expand/Collapse drop-down menu.

To collapse a transaction, perform the same operation and select **Collapse This Transaction**.

Note that you can choose to expand or collapse

- **Only** the selected Transaction
- OR
- **All** Transactions.

It is not necessary to use the **Expand/Collapse Transactions** menu to shift between expanded and collapsed views of a transaction:

- Double-click in the Transaction number field to toggle back and forth between collapsed and expanded views.

8.14 Switch to Split Transaction View

To select Split Transaction View,

Step 1 Click the  button on the toolbar.

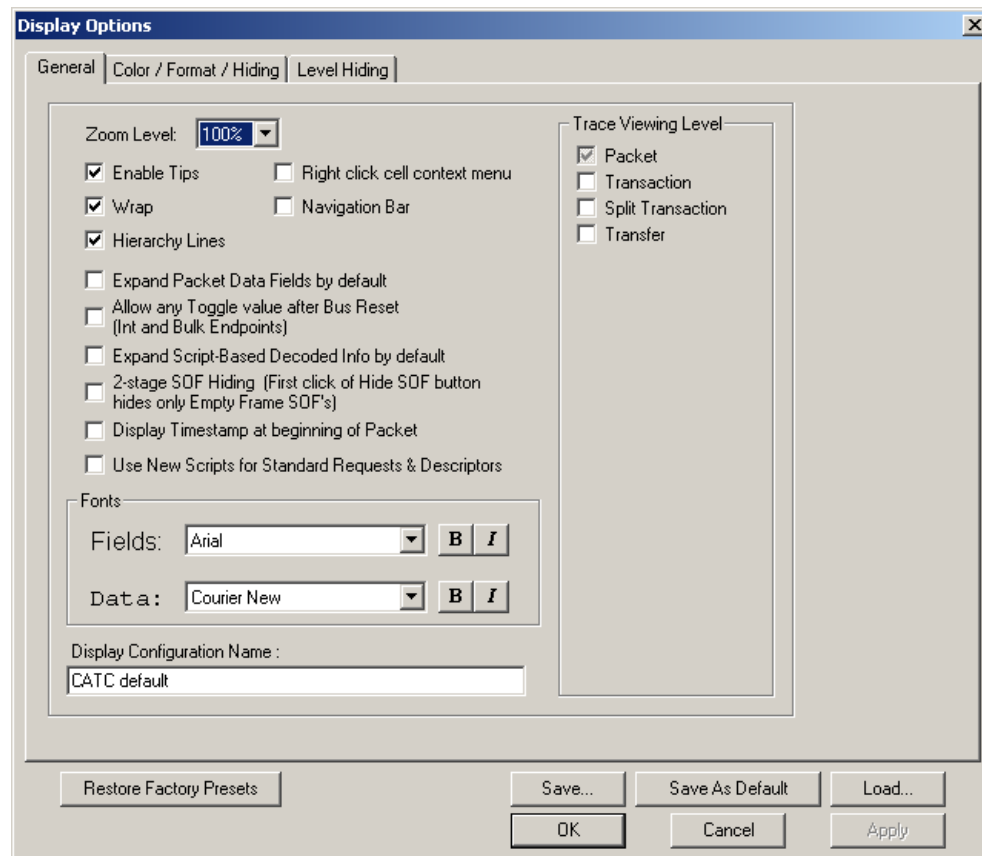
The Trace View screen is re-drawn to display Split Transactions.

Split Trans	L	IN	ADDR	ENDP	T	Data	ACK
44	H	0x96	3	0	1	12 01 00 01 00 00 00 08	0x4B

You can also switch to Split Transactions View from the Menu Bar:

Step 1 Select **Display Options** under **Setup**.

You see the **Display Options General** window:



Step 2 Check **Split Transactions**.

8.15 Switch to Transfer View

A **Transfer** is defined in the USB specification as one or more transactions between a software client and its function. USB transfers can be one of four kinds: Control, Interrupt, Bulk, and Isochronous. *USBTracer* is capable of displaying all four types.

The default display mode is Packet View. Before you can view decoded transfers, you must switch from Packet View (or Transaction View) to Transfer View.

To select Transfer View,

Step 1 Click  on the toolbar.

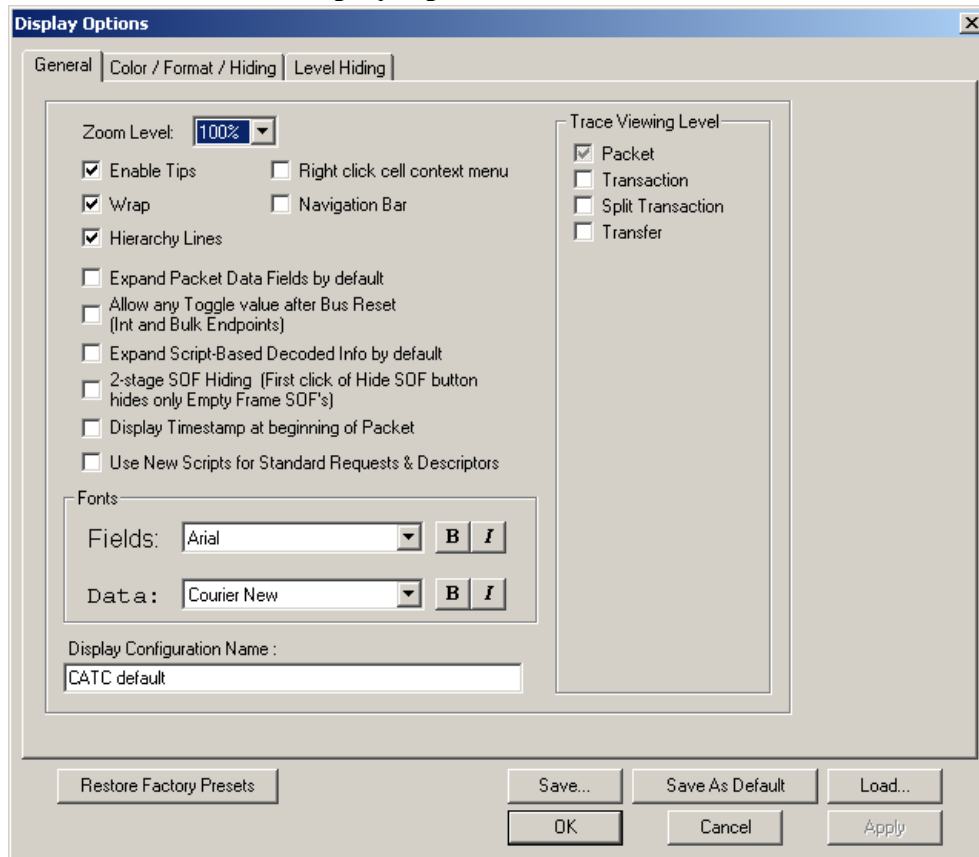
The Trace View screen is re-drawn to display Transfers.

Note Selecting **Show USB transfers** adds a checkmark next to this menu item. If you want to return to Packet View, open the menu and reselect **Show USB transfers**. This action will remove the checkmark and return the display to Packet View.

You can also switch to Transfer View from the Menu Bar:

Step 1 Select **Display Options** under **Setup**.

You see the **Display Options General** window:



Step 2 Check **Transfers**.

Step 3 Click **OK**.

8.16 View Decoded Transfers

Once you set Display Options, the Trace View screen is re-drawn to display decoded transfers in the colors and format you selected.

Transfer	H	Control	ADDR	ENDP	bRequest	wValue	wIndex	Descriptors
1	S	GET	1	0	GET_DESCRIPTOR	CONFIGURATION type	0x0000	23 descriptors

Transaction	H	SETUP	ADDR	ENDP	D	T	R	bRequest	wValue	wIndex	wLength	A
3	S	0xB4	1	0	D->H	S	D	GET_DESCRIPTOR	CONFIGURATION type	0x0000	171	0

Packet	H	SETUP	ADDR	ENDP	CRC5	Pkt Len	Idle	Time Stamp
16495	S	0xB4	1	0	0x17	8	200 ns	00002.0675 1688

Packet	H	DATA0	Data				CRC16	Pkt Len	Idle	Time Stamp				
16496	S	0xC3	80	06	00	02	00	00	AB	00	0xEB26	16	233 ns	00002.0675 1708

Packet	H	ACK	Pkt Len	Time	Time Stamp
16497	S	0x4B	6	7.167 μ s	00002.0675 1738

Transaction	H	IN	ADDR	ENDP	T	Data	ACK	Time
4	S	0x96	1	0	1	64 bytes	0x4B	7.600 μ s

Transaction	H	IN	ADDR	ENDP	T	Data	ACK	Time
5	S	0x96	1	0	0	64 bytes	0x4B	7.500 μ s

Transaction	H	IN	ADDR	ENDP	T	Data	ACK	Time
6	S	0x96	1	0	1	43 bytes	0x4B	9.733 μ s

Transaction	H	OUT	ADDR	ENDP	T	Data	ACK	Time
7	S	0x87	1	0	1		0x4B	525.027 ms

Transfer	H	Control	ADDR	ENDP	bRequest	wValue	wIndex	Descriptors	Time
2	S	GET	1	0	GET_DESCRIPTOR	DEVICE type	0x0000	DEVICE descriptor	1.224 sec

Transfer	H	Control	ADDR	ENDP	bRequest	wValue	wIndex	Descriptors	Time
3	S	GET	1	0	GET_DESCRIPTOR	STRING type, Index 1	Language ID 0x001B		362.4

When you instruct the Analyzer to display USB transfers, the components of each transfer are collected from the current recording and are grouped below each decoded transfer. Each transfer row shows a transfer with a unique numeration, a label, and color-coded decoding of important data.

Expanded and Collapsed Transfers

You can expand a specific transfer to view its parts, which are grouped and indented below the transfer.

To expand a transfer,

Step 1 Left-click on the transfer number you wish to view.

You see the **Expand Transfer** menu:

Transfer	F	Control	ADDR	ENDP	bRequest	wValue	wIndex
1					0x0D	0x0000	0x0000
2					0x09	0x0000	0x0000
3					0x0D	0x0000	0x0000
4					0x09	0x0000	0x0000
5					0x0D	0x0000	0x0000
6					0x09	0x0000	0x0000

Transfer 1
 Expand This Transfer
 Expand All Transfers
 Collapse All Transfers
 Set marker
 Format
 Color
 Hide

Step 2 Select **Expand This USB Transfer**.

The screen displays the selected transfer in expanded format.

Transfer	F	Control	ADDR	ENDP	bRequest	wValue	wIndex	
9	S	SET	1	0	SET_INTERFACE	New alternate setting 0	Interface # 0	

Transaction	F	SETUP	ADDR	ENDP	D	T	R	bRequest	wValue	wIndex	wLength	ACK
1332	S		0xB4	1	0	H->D	S	SET_INTERFACE	New alternate setting 0	Interface # 0	0	0x4B

Packet	Dir	F	Sync	SETUP	ADDR	ENDP	CRC5	EOP	Idle	Time Stamp
6925	-->	S	00000001	0xB4	1	0	0x17	233 ns	183 ns	00011.5989 1586

Packet	Dir	F	Sync	DATA0	Data	CRC16	EOP	Idle	Time Stamp
6926	-->	S	00000001	0xC3	8 bytes	0x231F	233 ns	483 ns	00011.5989 1771

Packet	Dir	F	Sync	ACK	EOP	Time	Time Stamp
6927	<--	S	00000001	0x4B	233 ns	988.017 μs	00011.5989 2294

Transaction	F	IN	ADDR	ENDP	T	Data	ACK	Time
1333	S		0x96	1	0	1 0 bytes	0x4B	2.000 ms

Transfer	F	Control	ADDR	ENDP	bRequest	wValue	wIndex	Data	Time
10	S	SET	1	0	0x01	0x0000	0x0000	64 bytes	10.998 ms

Transfer	F	Control	ADDR	ENDP	bRequest	wValue	wIndex	Data	Time
11	S	SET	1	0	0x01	0x0000	0x0001	64 bytes	10.998 ms

Note The Expand/Collapse transfer feature operates as a toggle: when one format is active, the other appears as an option on the Expand/Collapse drop-down menu.

To collapse a transfer, perform the same operation and select **Collapse This USB Transfer**.

Note that you can choose to expand or collapse

- **Only** the selected Transfer
- OR
- **All** Transfers.

It is not necessary to use the **Expand/Collapse Transfers** menu to shift between expanded and collapsed views of a transfers:

- Double-click in the Transfer number field to toggle back and forth between collapsed and expanded views.


8.17 Decoding Protocol-Specific Fields in Transactions/Transfers

When transfers or transactions are displayed, the fields in setup transactions or control, interrupt, and Bulk transfers by default do not get decoded and are shown in hexadecimal values. The exceptions are setup transactions and control transfers for standard USB device requests which are always decoded.

In order to show specific decoding for class- and vendor-specific device requests and endpoints, you have to use the decoding association mechanism that is described in Chapter 9 on decoding. When you have performed the association, you will see the protocol-specific fields of transfers and transactions decoded in the trace view.

8.18 Using the Trace Navigator

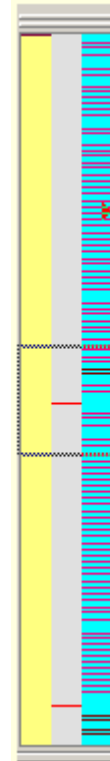
You can use the Trace Navigator to show the structure of the entire trace from different points on the Trace view, thereby refining the trace view to a packet range in the trace that is most interesting to you. This range can be set by you to show smaller parts of the trace.

To display the Navigator bar, click  in the toolbar or select the Navigation Bar checkbox in the Display Options General window.

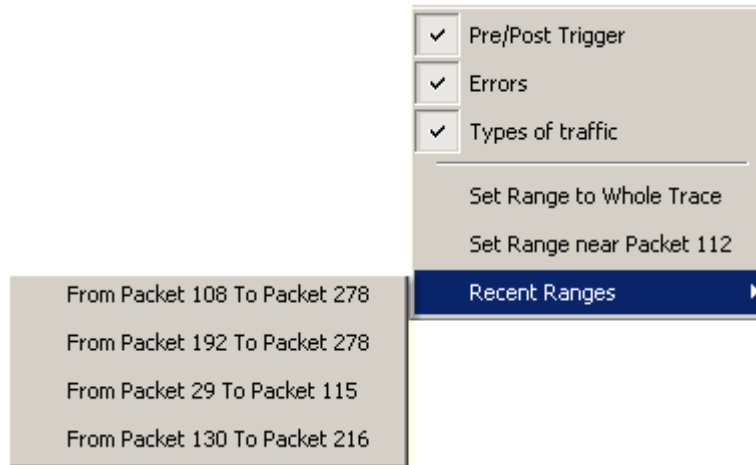
The Navigator bar can be repositioned in the trace and can be oriented horizontally or vertically, docked or undocked by dragging the parallel bars at the top or side of the Navigator bar. By default, the Navigator bar appears vertically to the right of the trace window.

The Navigator bar represents different types of trace information in the order of the packets. The top of each bar corresponds to the first packet in the trace, and the bottom corresponds to the last packet. The Navigator bar is made up of three parts: Pre and Post-Trigger traffic, errors, and types of traffic.

At any time, a line in the navigator bar of one pixel in height represents a fraction of the trace data. If the Navigation bar is 400 pixels high, then each bar in this example would represent 1/400 of the trace. If the trace had 4000 packets total, each bar would represent ten packets. In the Types of Traffic portion of the navigation bar, the color of the bar would be that of the most important item in those ten packets. See "Navigator Bar Attributes" on the next pages.

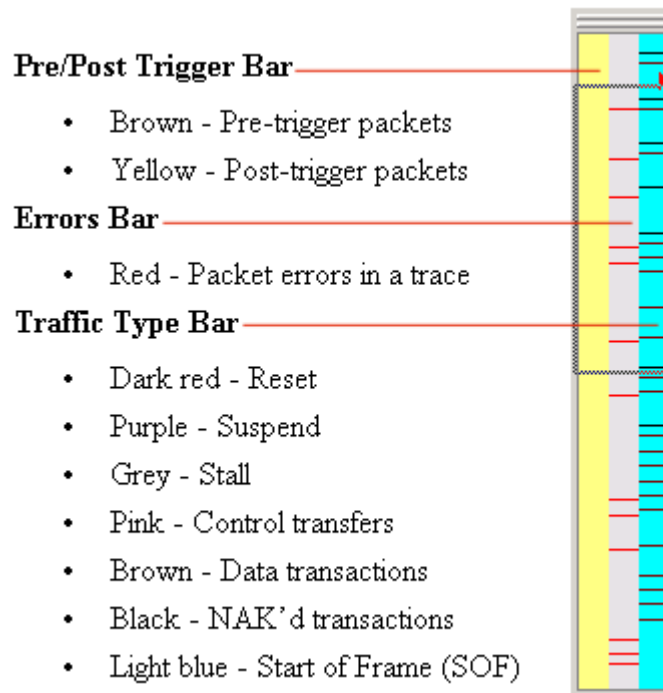


Drag the square at the top of bottom to set the size of a packet range. Drag the entire square to set the packet range to different values. A bubble appears showing the packet range while you are setting it. Right click on the Navigator to display the pop-up menu.



- Pre/Post Trigger, Errors, Types of Traffic shows or hides specific type of traffic from the Navigator.
- Set Range to Whole Trace resets packet range to the entire trace.
- Set range near packet number sets the packet range near the packet where your mouse pointer is positioned in the Navigator bar.
- Recently Used Ranges displays a history of ranges that were previously selected.

Navigator Bar Attributes



The traffic types above are shown in order of importance. For example, if a NAK'd transaction occurred in the same sample area as a SOF, the NAK would take precedence, and the Navigator would show the black bar.

9. Decode Requests

9.1 General Options

Commands are transferred on USB using special control transfers called USB Device Requests. The Analyzer can decode Device Requests as they are defined in the USB specifications and various Device Class and Vendor specifications.

USB Request

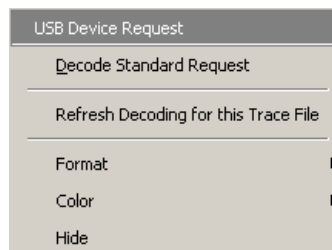
Each USB Device Request is sent using a Control Transfer. Each Control Transfer starts with a SETUP transaction.

To decode a USB Device Request,

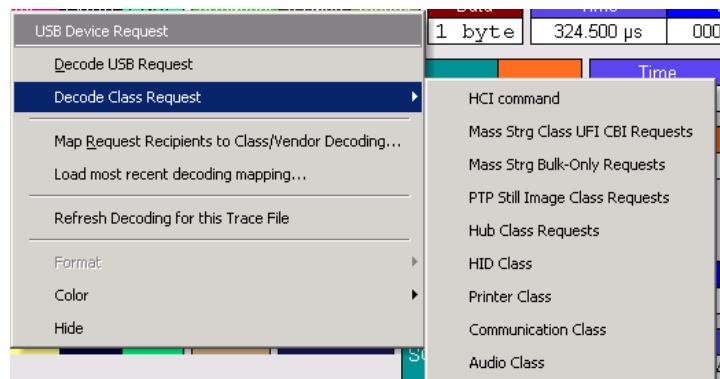
Step 1 Left-click once in the **SETUP** field of the packet that starts the Request.

Note The menu that appears when you left-click a setup field is context-sensitive and may not appear exactly as shown in the next few examples.

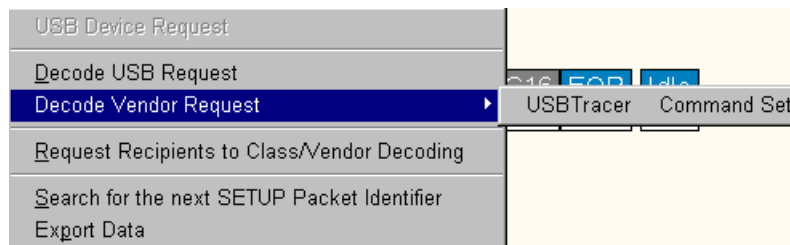
If you select **Standard Request**, you see the **Decode Standard Request** menu:



If you select **Class Request**, you see the **Decode Class Request** menu:

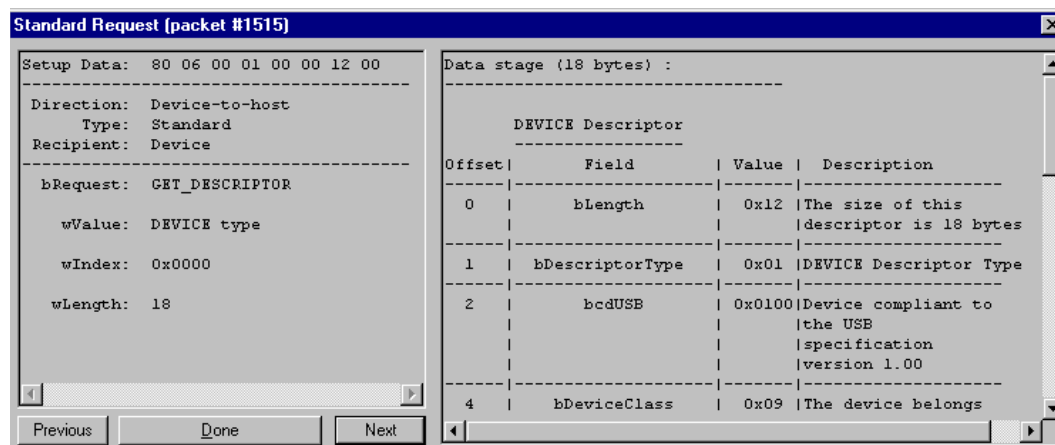


If you select **Vendor Request**, you see the **Decode Vendor Request** menu:



Decoding Standard Requests

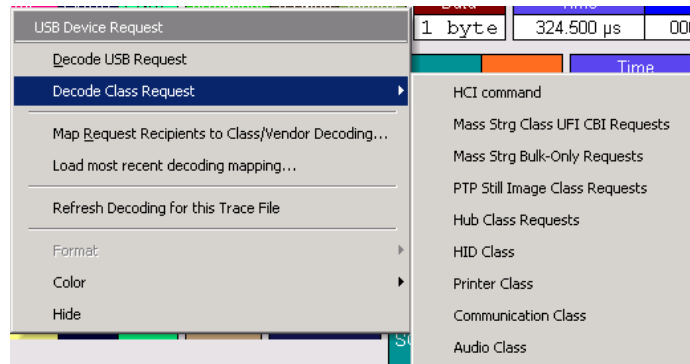
When you select **Decode Standard Request**, you see a window displaying the decoded Standard Request:



Shown here is a GET_DESCRIPTOR Standard Request.

Decoding Class Requests

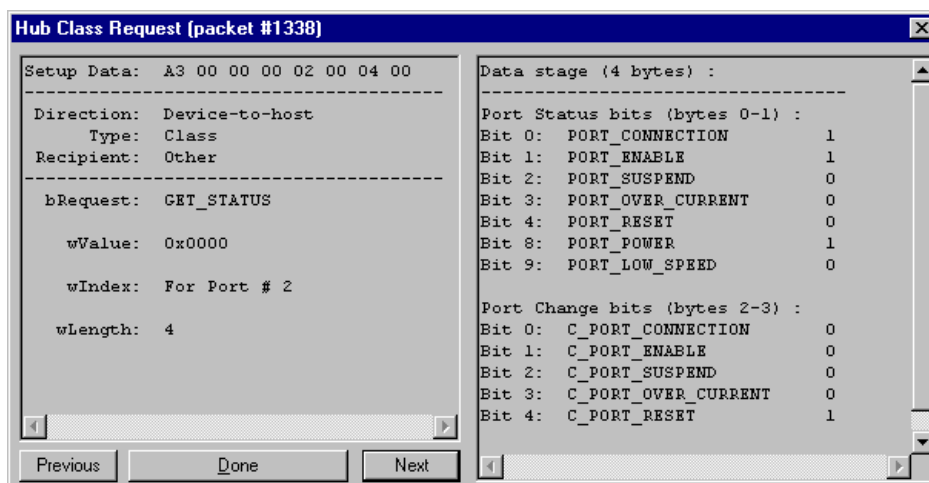
When you select **Decode Class Request**, you see the **Decode Class Request** drop-down menu:



The drop-down menu lists the Classes whose definitions are contained in the Analyzer.

Step 1 Select the Class which agrees with the Setup Address selected.

You see a Decode Request screen (similar to the one below) that provides you with definitions for the individual fields of the Request and the returned data:

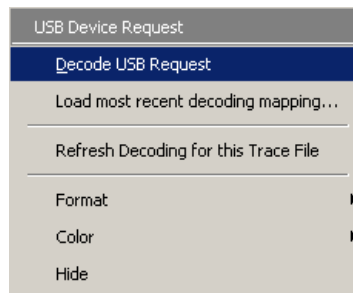


Shown here is a sample **GET_STATUS** Hub Class Request.

Decoding Vendor Requests

Decoding a Vendor Request is the same as decoding a Class Request except that you use Vendor Request definitions rather than Class Request definitions.

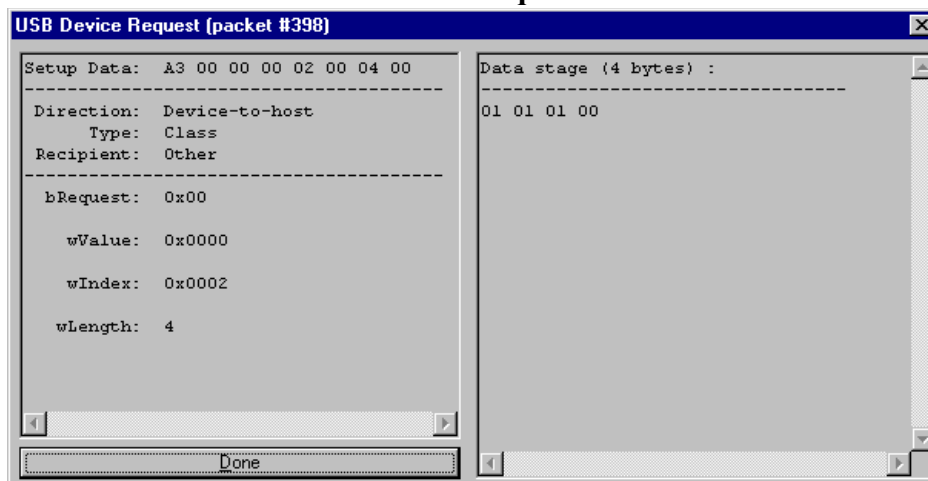
Decoding USB Requests



When your Decoding request does not belong to any of the defined Decoding groups (Standard, Class, or Vendor),

- Select **Decode USB Request** from the Request menu.

You see the **USB Decode Request** screen.



This screen provides definitions for the individual fields of the Request and the returned data.

Class and Vendor Definition Files

LeCroy analyzers use script files to decode class and vendor requests. These script files are read when the application is initialized. Once read, the analyzer then decodes class and vendor requests as instructed by the files.

.REQ Files and .DSC Files - Represent the old method of decoding. **.REQ** stands for Requests and **.DSC** stands for Descriptors. This method was originally developed for some of LeCroy's earlier USB analyzers and uses fixed definitional structures. LeCroy provides **.req** files for the following requests and classes:

- **Standard Requests** (standard.req) (Not used with Video Class decoding)

- **Hub Class** (hub.req) (No longer used: replaced by Script Decoder version .dec files)
- **HID Class** (hid.req)
- **Printer Class** (printer.req)
- **Communications Class** (communication.req)
- **Audio Class** (audio.req)
- **Bluetooth USB HCI commands and events** (bluetooth.req)
(No longer used: replaced by Script Decoder version .dec files)
- The Analyzer also includes a sample of a **Vendor Request definition file** (vendor.req).

To add your own **.req** file for a Class or Vendor Request,

Step 1 Use Notepad to create and edit your own Request Definition file (for example, my_vendor_commands.req).

Note To learn to write such a file, review the Request (.req) files provided by LeCroy.

Step 2 Add the name of your Request file to the request.lst file.

Step 3 Click **Setup**. The decoding menu in the Class/Vendor List appears.

.DEC Files - Represents the new method of decoding. This method is more powerful than the old and uses a C-like language. .DEC stands for "decoder" and describes both Class and Vendor requests.

.DEC files are found in the Scripts directory under the application directory. Each **.dec** file stores an endpoint or request decode. When the application starts, these files are loaded dynamically. Subdirectories are supported.

The following **.dec** files are provided:

- Bluetooth HCI
- Hubs
- Mass Storage
- Still Image/PTP
- Video

If desired, you can create your own **.dec** file for a Class or Vendor Request.

For more information on the format of these Script Decoder files and the Script Decoding language, read the ***Script Decoder Manual***.

Note If Video is selected, the Display Options need to be set to "Use new Std Reqs & Descriptors Scripts (for Video Class)" for video. The file **VideoCurrentEntityIdMap_EditMe.inc** in the VideoClass directory will also need to be edited. You will be assigning entity IDs to corresponding Control types. The instructions are in that file.

9.2 Class/Vendor Decoding Options

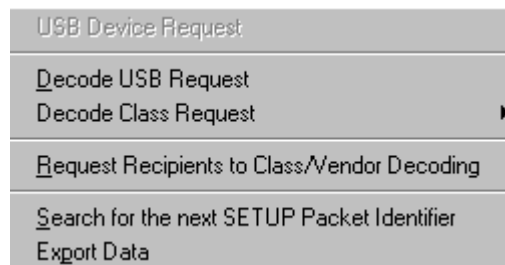
This interface allows you to permanently assign a class or vendor decoding for an address and/or endpoint or interface in a Trace file. Once assigned, the decoding occurs automatically when you choose to display transactions. You see the Decode Class menu for whichever Class type you have selected.

Request Recipient to Class/Vendor Decoding

To assign a decoding group to a request recipient,

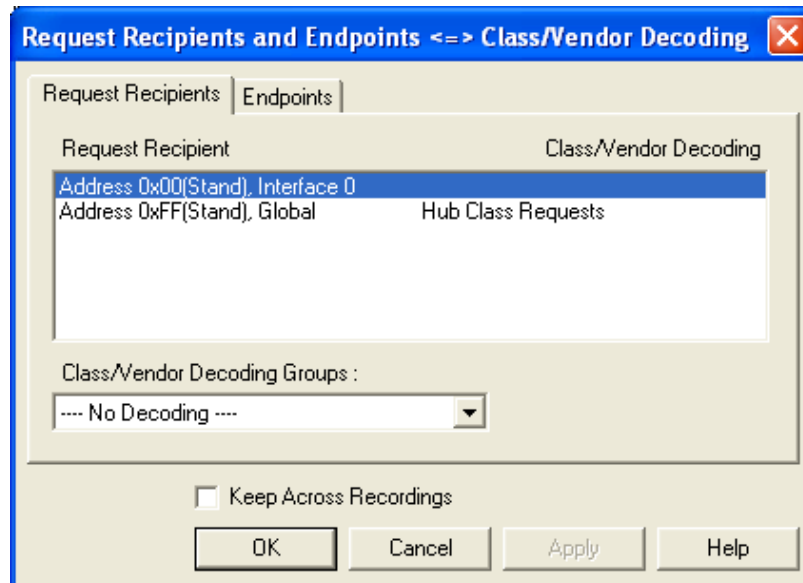
Step 1 Left-click once in the packet's **SETUP** field.

You see the **Decode Request** menu:



Step 2 Select **Request Recipient to Class/Vendor Decoding**.

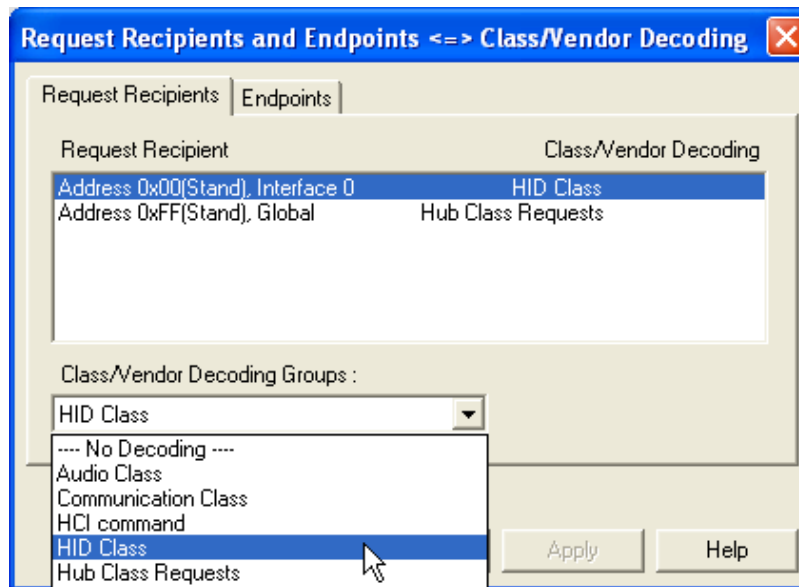
You see the **Request Recipient** window:



The **Request Recipient** field shows all the USB Class and Vendor Request Recipients found in the Trace file. Displayed on the right are the names of Class/Vendor decoding groups currently assigned to recipients. If no decoding is assigned for a recipient, nothing is displayed next to the address.

To select a recipient for which to assign or modify its Class/Vendor decoding,

Step 1 Display the **Class/Vendor Decoding Groups** drop-down menu.



The drop-down menu lists the defined Class/Vendor request decoding groups.

Step 2 In the **Request Recipient** field, highlight a Recipient.

Step 3 From the **Class/Vendor Decoding Groups** drop-down menu, select the decoding group you want to assign to the highlighted Recipient
OR

Select **No Decoding** if you do not want any specific decoding.

Step 4 Repeat the previous step for any additional recipients you would like to map.

Step 5 Click **OK**.

Class/Vendor Endpoint Decoding

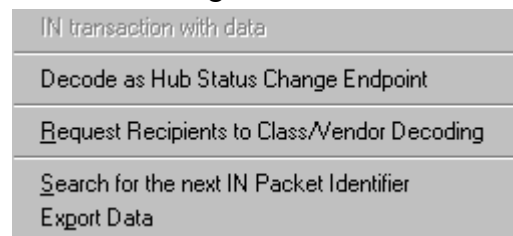
Some usb endpoints can transfer formatted data that is defined in various device class or vendor specifications.

USBTracer provides the same decoding capabilities for data sent on endpoints as for device requests. You can define decoding for endpoints by adding EndpointData construct to your .req file (see examples in hub.req and bluetooth.req.)

To assign a Class/Vendor Endpoint decoding,

Step 1 Left-click once in the packet's **IN** or **OUT** field.

You see the following menu.



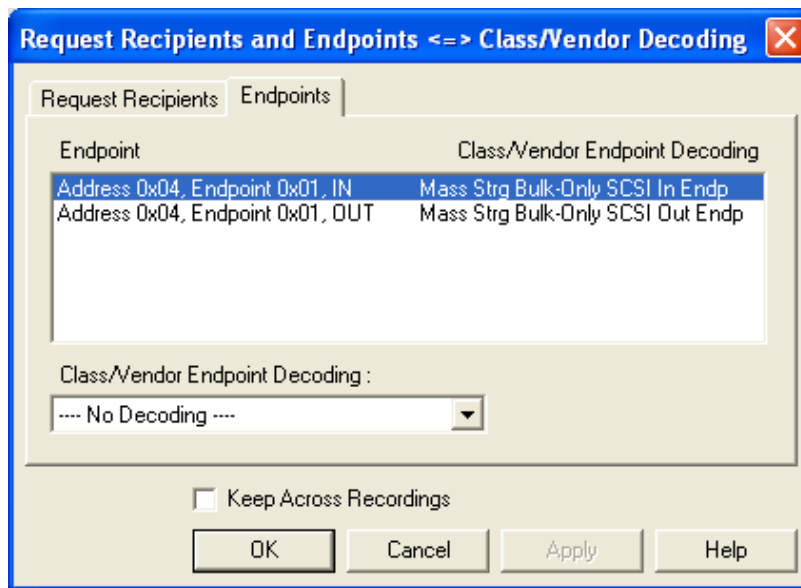
Step 2 Select **Request Recipient to Class/Vendor Decoding**

You see the Request Recipients and Endpoints dialog box.

Step 3 Click on the tab marked **Endpoints**.

You see the **Endpoints** dialog box.

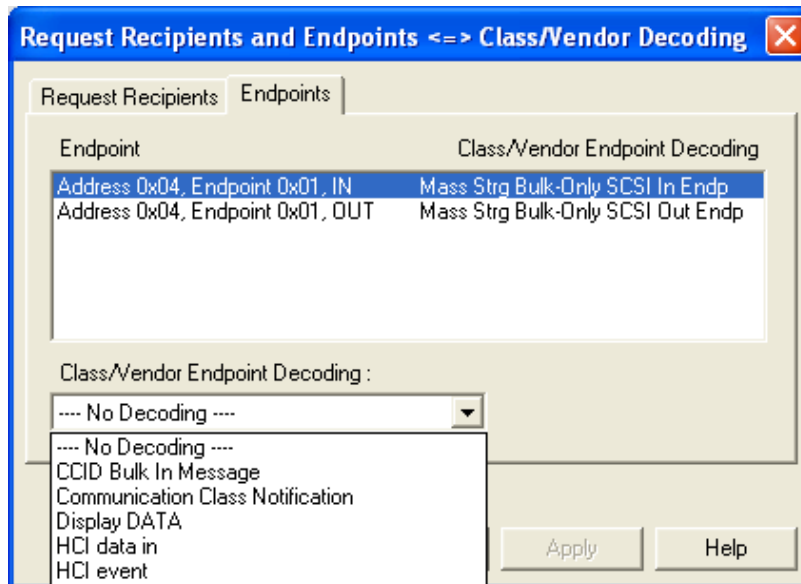
The **Endpoint** field shows all the Endpoints found in the Trace file. Displayed on the right are the names of Class/Vendor Decoding that are currently assigned to Endpoints.



To select an endpoint for assigning or modifying its Class/Vendor decoding,

Step 1 Display the **Class/Vendor Decoding Groups** drop-down menu

You see the Class/Vendor Endpoint Decoding drop down menu.



Step 2 In the **Endpoint** field, highlight an address, Endpoint, and Field.

Step 3 From the **Class/Vendor Endpoint Decoding** drop-down menu, select the type of decoding you want

OR

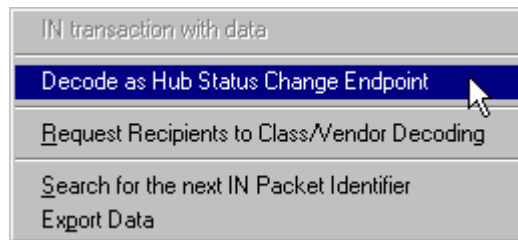
Select **No Decoding** if you don't want any specific decoding.

Step 4 Repeat the previous step for any additional endpoints you would like to map.

Step 5 Click **OK**.

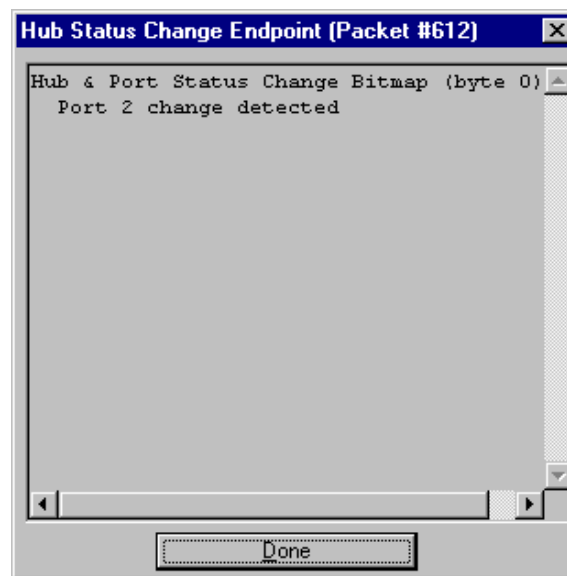
To decode

Step 1 Left-click the field marked **IN** or **OUT**. You will see the following or similar menu:



Step 2 Select **Decode as Hub Status Change Endpoint**

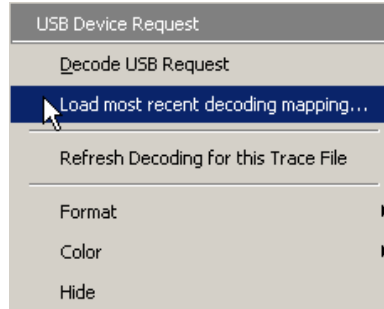
A text box will open that will display the Hub Status Change Endpoint.



Load most recent Decoding Mapping...

This option loads the most recent mapping of endpoints/requests to decoding types that was done on a previous trace. If the endpoints of the new trace are the same as the last one mapped, the mappings are applied to the current trace. This saves the user from having to constantly apply the mapping to a new trace every time the application is restarted and a new trace created.

The ability to retain the mapping from trace to trace DURING an application session already exists: the Keep Across Recordings button in the endpoint map dialog. This new feature simplifies the process when the application has been re-started.



Refresh Decoding for this Trace File

This option causes the software to reanalyze all transactions. This option is sometimes helpful when decoding traces.

10. Other Features

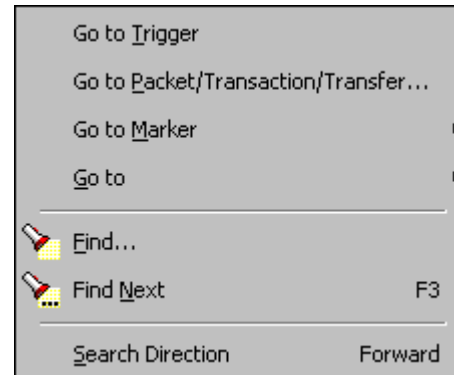
10.1 Search

The Search feature provides several options for searching through recorded traffic, allowing you to find specific packets based on triggering status, packet number, marking, or content.

To view the Search options,

- Click **Search** in the Menu bar.

You see the Search drop-down menu:



Go to Trigger

Note **Go to Trigger** is enabled only when a trigger has created the traffic file.

To display a Trigger Event,

- Select **Go to Trigger** under **Search** on the Menu Bar.

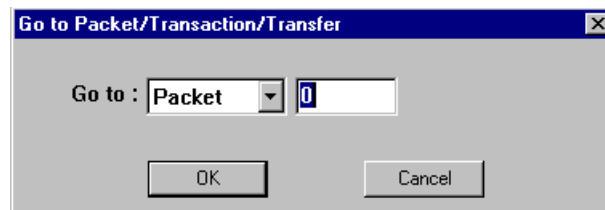
The Trace View is repositioned to the first packet following the Trigger event. This packet will be at the top of the screen.

Go to Packet/Transaction/Transfer ...

To display a specific packet,

Step 1 From the menu bar, select the command **Search > Go to Packet/Transaction/Transfer**

You see the **Go to Packet** window:



Step 2 Select the desired viewing level (packet, transaction etc.) from the drop-down menu next to the words **Go to**.

Step 3 Enter the number of the packet you want to display.

Step 4 Click **OK**.

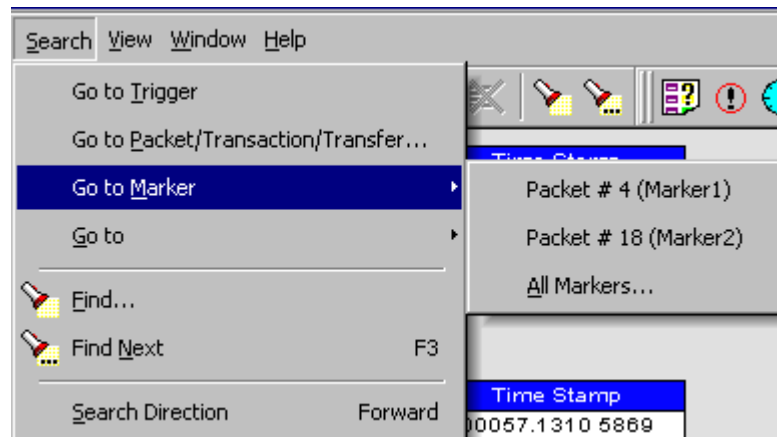
The Trace View is repositioned with the selected packet at the top of your screen.

Go to Marker

To instruct the Analyzer to display a marked packet,

Step 1 Select **Go to Marker** under **Search** on the Menu Bar.

You see a drop-down menu listing the marked packets in that Trace View:



Step 2 Select the desired packet from the displayed list.

The Trace View is repositioned with the selected packet at the top of your screen.

Note The **Go to Marker** feature functions in conjunction with the **Set Marker** feature. The comments within the parentheses following each marked packet are added or edited with the **Set Marker** feature. Please refer to **Set Marker in Reading a Trace**.

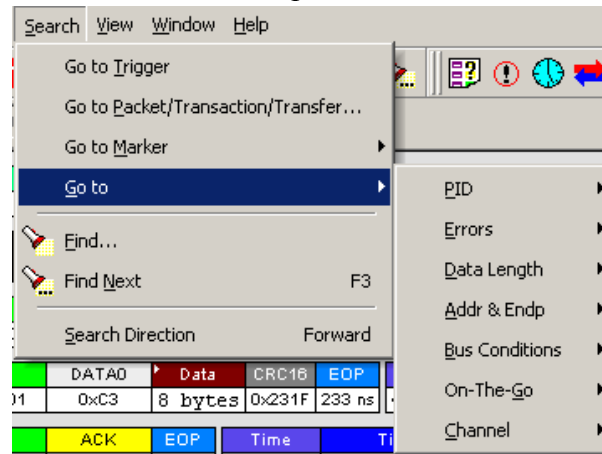
You can use **Ctrl+M** to go immediately to the All Markers dialog.

Go To

The **Go To** feature takes you directly to an event in a Trace.

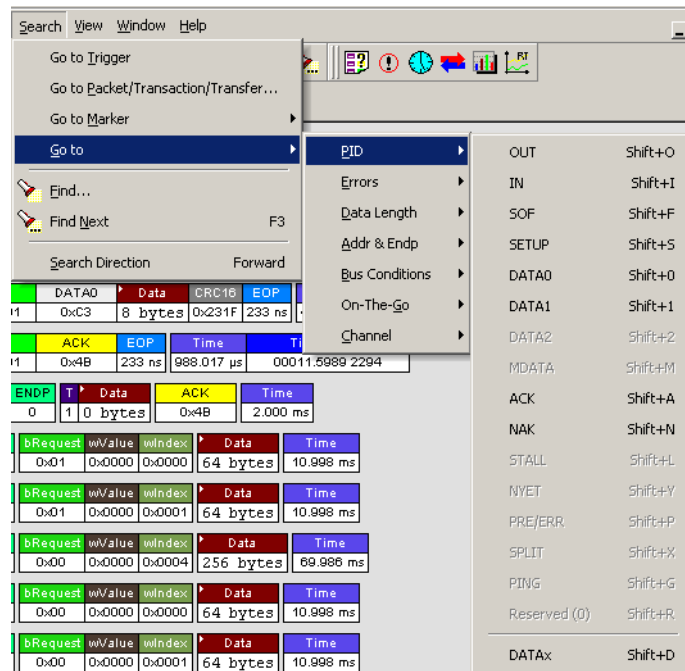
Step 1 Select **Go To** under **Search** on the Menu Bar.

You see the **Go To** drop-down menu:



Step 2 Select the event you want to go to and enter the necessary information.

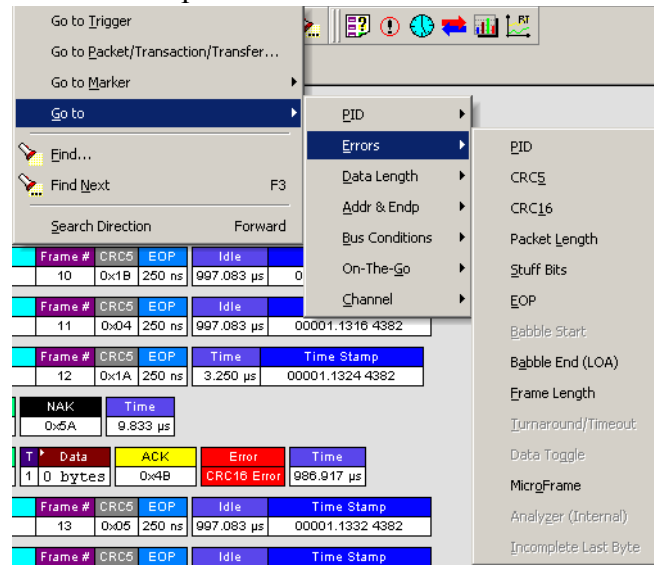
Packet IDs (PIDs)



Select the type of packet you want to go to.

Errors

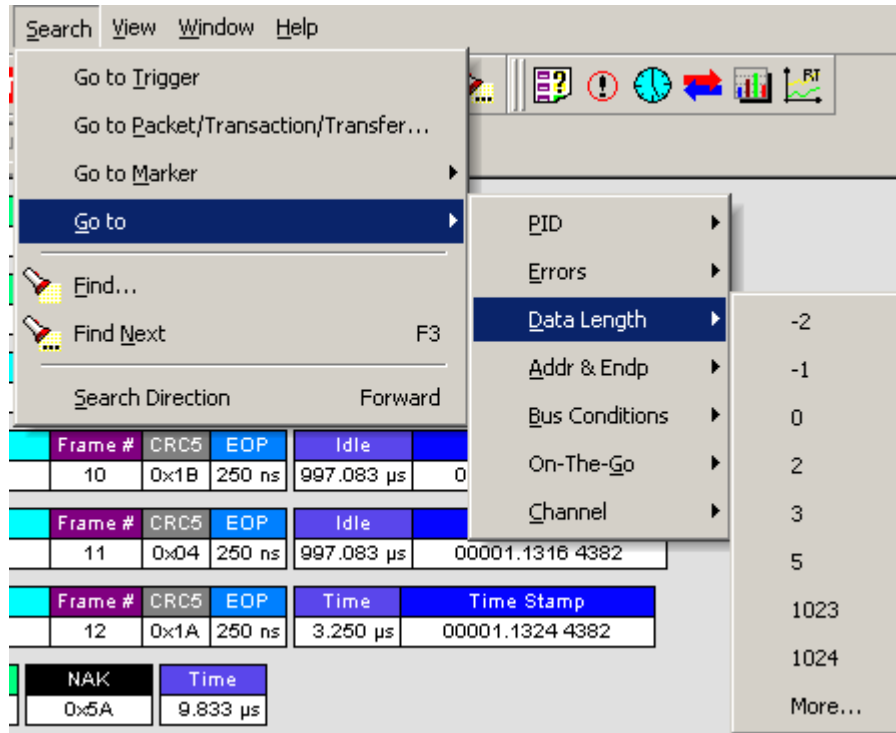
The Errors menu allows you to search for five different types of error: PID, CRC5, CRC16, Packet Length, and Stuff Bits. Menu items will appear in bold if they are present in the trace or grayed out if not present in the trace as shown in the example below.



You can press **Shift+E** to go to the first error of any type.

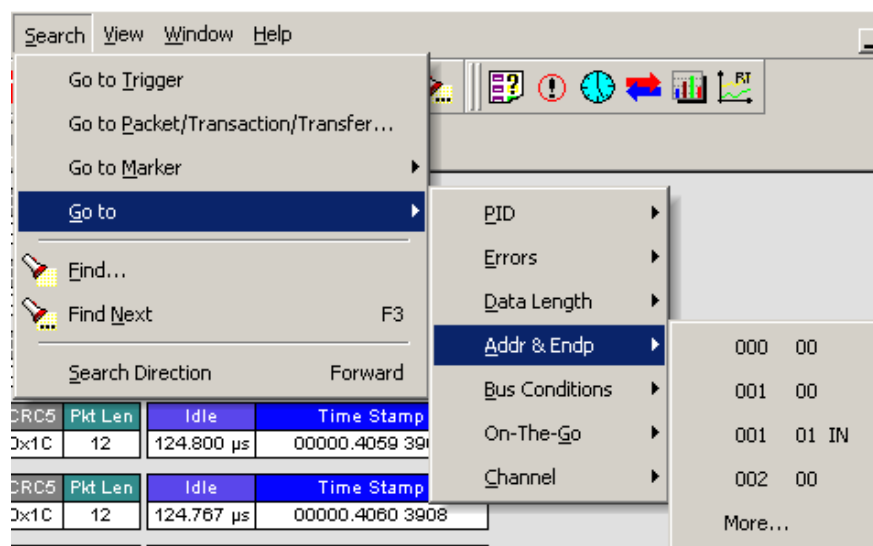
Data Length

Allows you to search for data packets of particular lengths. Lengths are displayed in Bytes in a drop down menu as shown below. Selecting a length will cause the display to move to the next instance of that packet length.



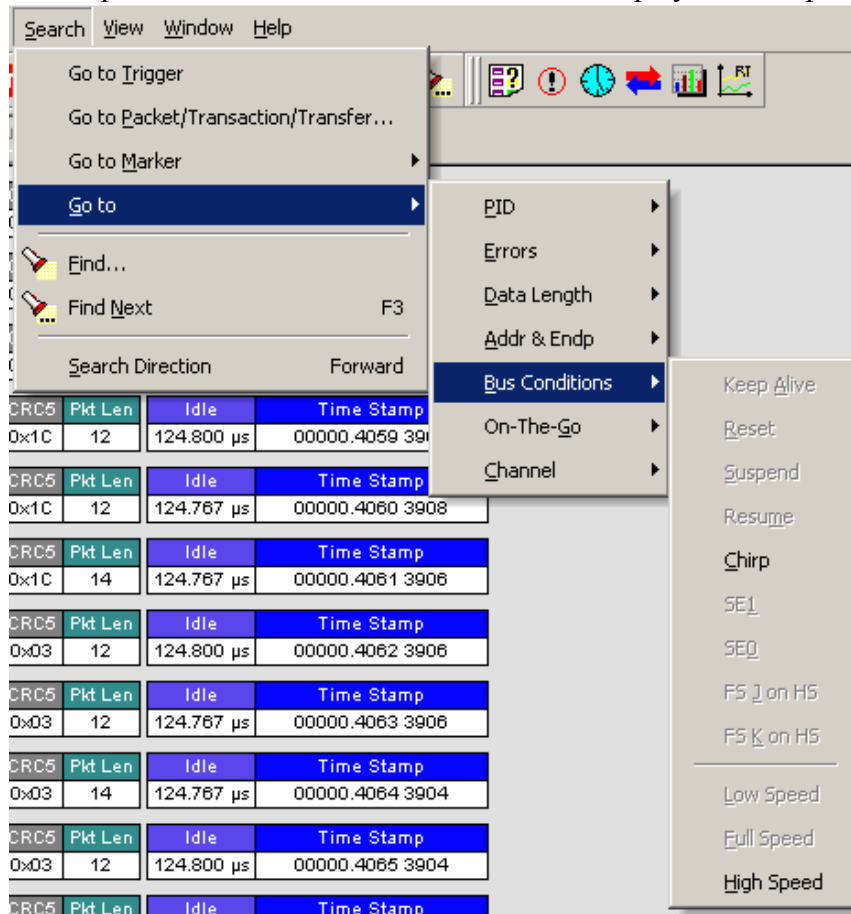
Addr & Endp

The Addr & Endp feature allows you to search for the next packet which contains a particular address and endpoint. All available address endpoint combinations will be displayed in the pull down menu.



Bus Conditions

Allows you to search by bus conditions such as traffic speed, reset, and suspend. All available bus conditions are displayed in the pull down menu.



On-the-Go

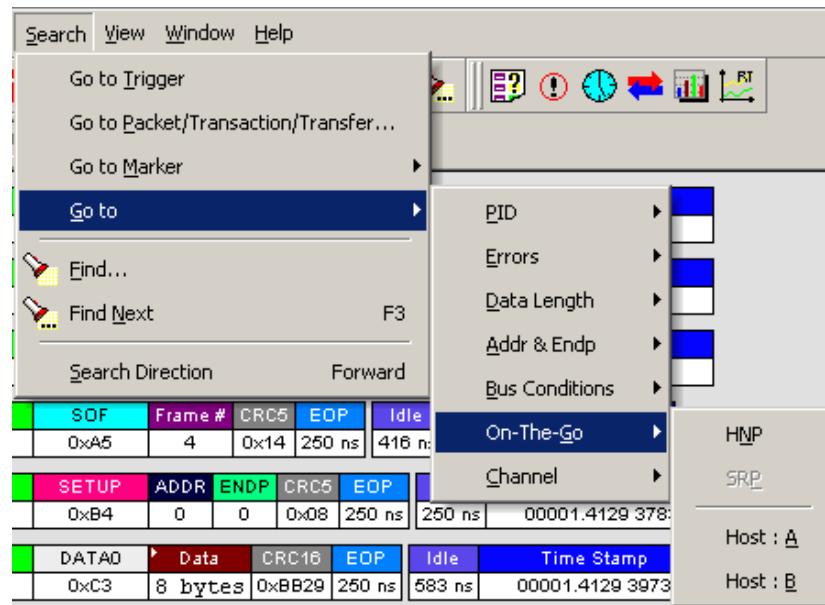
Allows you to search for On-the-go attributes. The On-the-Go sub-menu contains entries for:

HNP - Host Negotiation Protocol

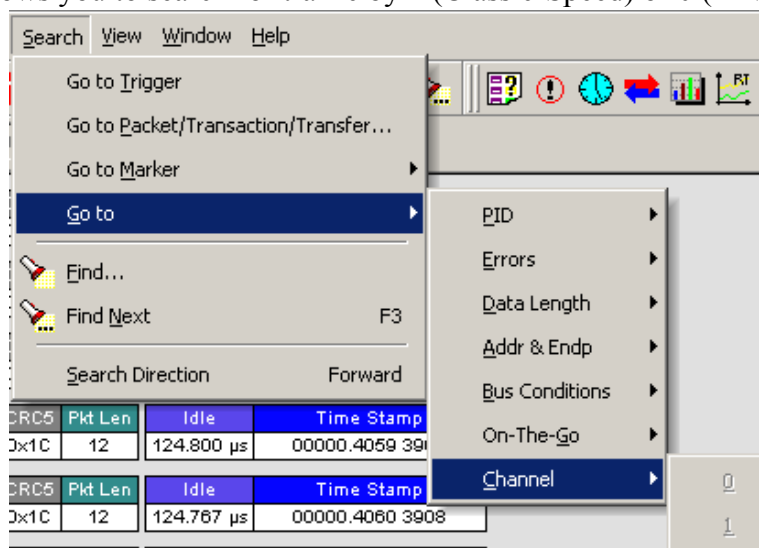
SRP - Session Request Protocol

Host : A - Hosts with an A plug

Host: B - Hosts with a B plug

*Channel*

Allows you to search for traffic by 1 (Classic-Speed) or 0 (Hi-Speed).



Find

Find allows searches to be conducted on an open trace using one or more criteria. You can search by packet, transactions, split transaction, transfer, by packet type and by fields within packets.

Find is run by selecting **Search > Find** or by clicking  on the toolbar.


Searches can combine criteria using the options **Intersection** and **Union**. **Intersection** creates AND statements such as "Find all packets with *x* and *y*." **Union** creates OR statements such as "Find all packets with *x* OR *y*."

You can also perform searches whereby packets or events are excluded from a trace. The **Exclusion** allows searches to be conducted.

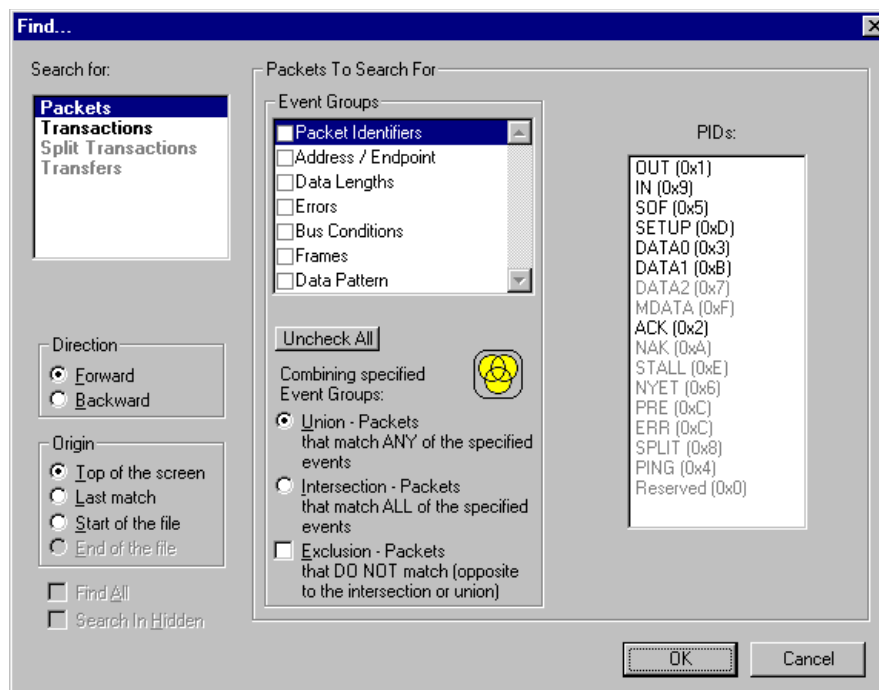
To perform a search,

Step 1 Select **Find...** under **Search** on the Menu Bar

OR

Click  in the Tool Bar.

You see the **User-Defined Find Events** screen:



Step 2 Select **Packet**, **Transaction**, **Split Transaction**, or **Transfer** from the top left list box.

Your choice will affect the options presented in the Events Group box.

Step 3 Select one or more events from the **Events Group** box.

Step 4 Select one of the following options:

- **Union:** Find all packets matching ANY of the specified events.



- **Intersection:** Find packets matching ALL of the specified events.



- **Exclusion:** Exclude packets matching any of the specified events.

Exclusion works with the other two options: Select **Union AND Exclusion** (=Exclude packets with ANY of the following fields) or **Intersection AND Exclusion** (=Exclude packets with ALL of the following fields.)



Step 5 If desired, set the search **Direction** and **Origin**.

Step 6 Click **OK**.


After the search finishes, the packets meeting the search criteria will display.

Find Next

To apply the previous **Find** parameters to the next search,

- Select **Find Next** under **Search** on the Menu Bar

OR

Click  on the Tool Bar.

Search Direction

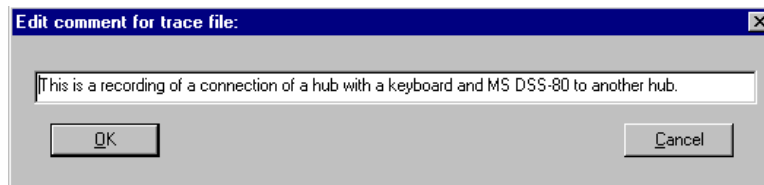
Toggles the search forward or backwards. The current direction is indicated in the menu.

10.2 Edit Comment

You can create, view, or edit the 100-character comment field associated with each Trace file. These comments will be visible in the Windows Explorer if the "Comments" attribute is included in the Details view.

Step 1 Select **Edit Comment** under **File** on the Menu Bar.

You see the **Edit comment for trace file** window:



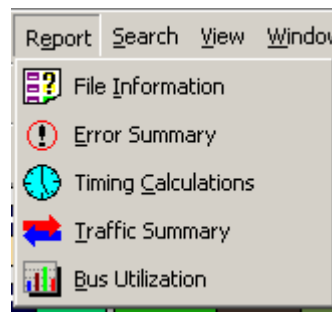
Step 2 Create, view, or edit the comment.

Step 3 Click **OK**.

You can view comments in Windows File Explorer by selecting the "Comments" attribute.

10.3 Reports

The Report menu provides several reports to assist you in analyzing USB traffic recorded by the Analyzer.




File Information

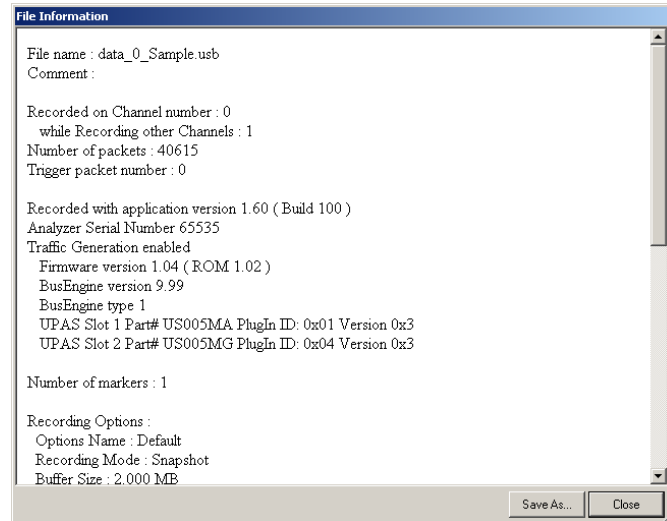
To display a File Information report,

- Select **File Information** under **Report** in the Menu Bar

OR

Click  in the Tool Bar.

You see the File Information screen:




The File Information report provides information about how the recording was made, what the buffer settings were, what the trigger options were, and what version of all the Analyzer hardware was used to make the recording.

Error Summary

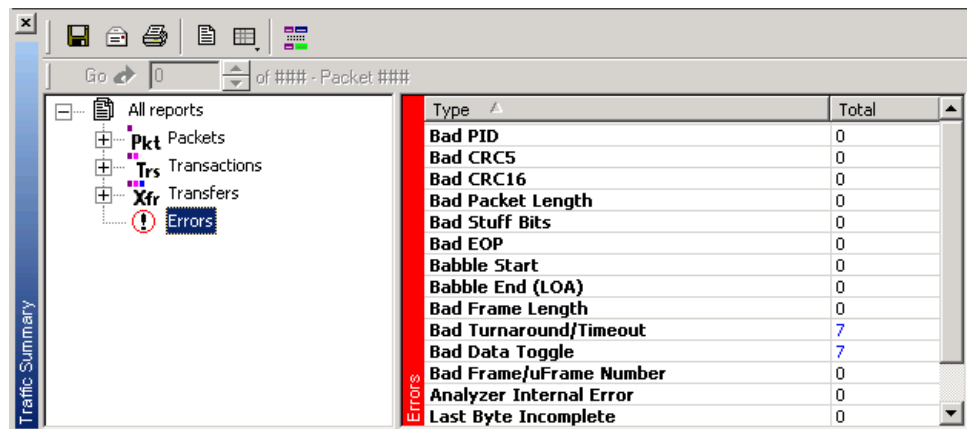
The Error Summary details all errors analyzed throughout the recording.

- Select **Error Summary** under **Report** in the Menu Bar

OR

Click  in the Tool Bar.

You see the Error screen below the Trace View:




Timing Calculations

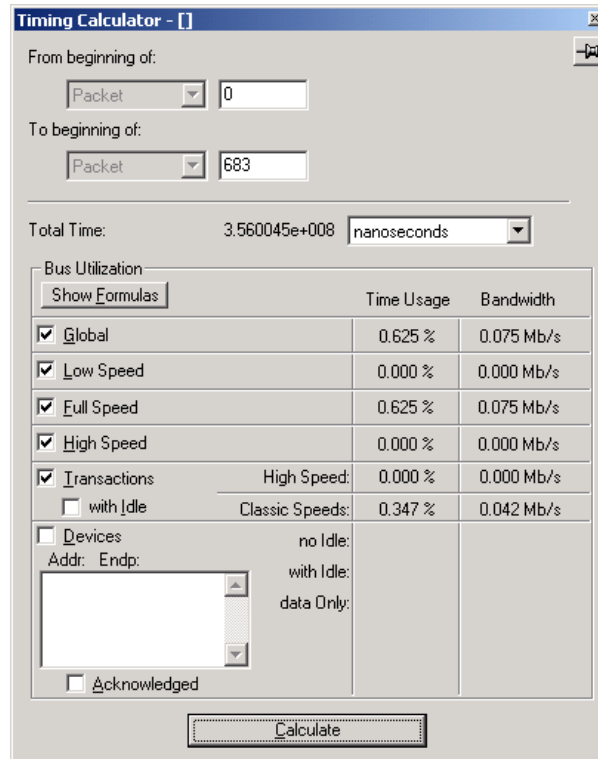
The Timing Calculator is used to measure timing between any two packets.

- Select **Timing Calculations** under **Report** in the Menu Bar

OR

Click  in the Tool Bar.

You see the Timing and Bus Usage Calculator screen:



Timing Calculator - [X]

From beginning of:
Packet 0

To beginning of:
Packet 683

Total Time: 3.560045e+008 nanoseconds

Bus Utilization
Show Formulas

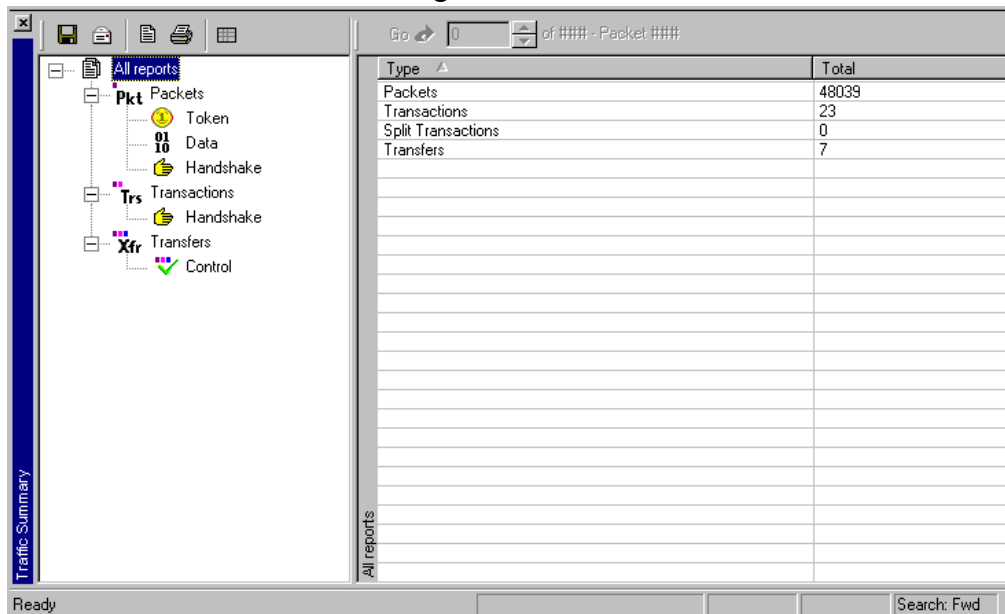
	Time Usage	Bandwidth
<input checked="" type="checkbox"/> Global	0.625 %	0.075 Mb/s
<input checked="" type="checkbox"/> Low Speed	0.000 %	0.000 Mb/s
<input checked="" type="checkbox"/> Full Speed	0.625 %	0.075 Mb/s
<input checked="" type="checkbox"/> High Speed	0.000 %	0.000 Mb/s
<input checked="" type="checkbox"/> Transactions	High Speed: 0.000 % Classic Speeds: 0.347 %	0.000 Mb/s 0.042 Mb/s
<input type="checkbox"/> with Idle		
<input type="checkbox"/> Devices	no Idle: with Idle: data Only:	
Addr: Endp:		
<input type="checkbox"/> Acknowledged		

Calculate

Traffic Summary


Traffic Summary summarizes the numbers and types of packets, transactions etc. that occurred in the open trace.

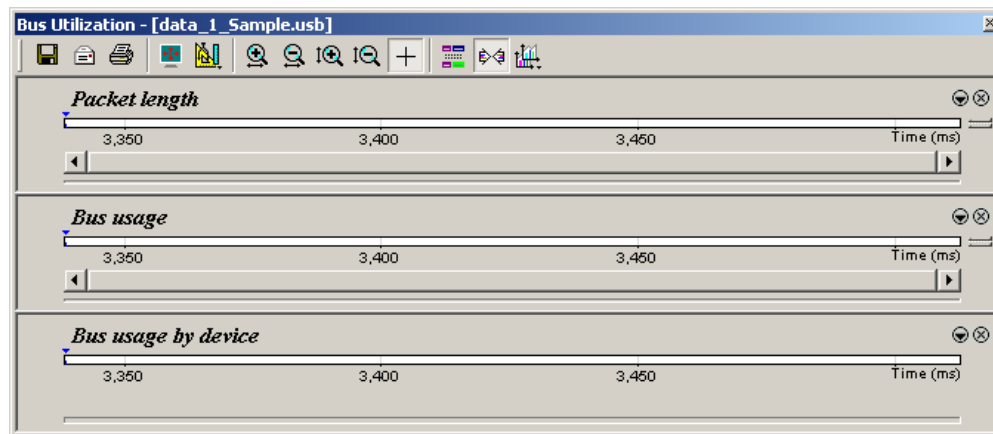
To run **Traffic Summary**, select **Report > Traffic Summary** or click the button marked . You will be prompted to specify a range of packets, then be shown the following window:



Bus Utilization

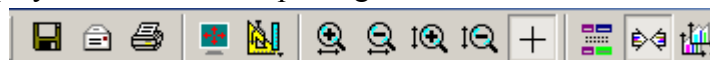
The **Bus Utilization** window displays information on bandwidth use for the three recording channels.

To open the Bus Utilization window, select **Report > Bus Utilization** or click the button marked . A window will open with two graphing areas:













Bus Utilization Buttons

The Bus Utilization window has a row of buttons for changing the format of the displayed data and for exporting data:



The buttons have the following functions:

	Save As - Saves the graphs as a bitmap file (*.bmp)		Vertical zoom in
	Email - Creates an email with a *.bmp file attachment of the graphs		Vertical zoom out
	Print		Click and Drag zoom - Click diagonally to select and zoom in on part of the graph
	Full Screen		Select Range
	View Settings - opens a sub-menu with options for formatting the display. See "View Settings Menu" below.		Sync and Graph areas - If two or more graphs are displayed, this button will synchronize the graphs to one another. Once synchronized, the positioning slider of one graph will move the other graphs



Horizontal zoom in




Graph Areas - Presents options for displaying additional graphs of data lengths, packet lengths, and percentage of bus utilized.

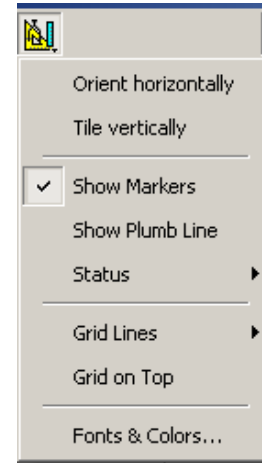


Horizontal zoom out

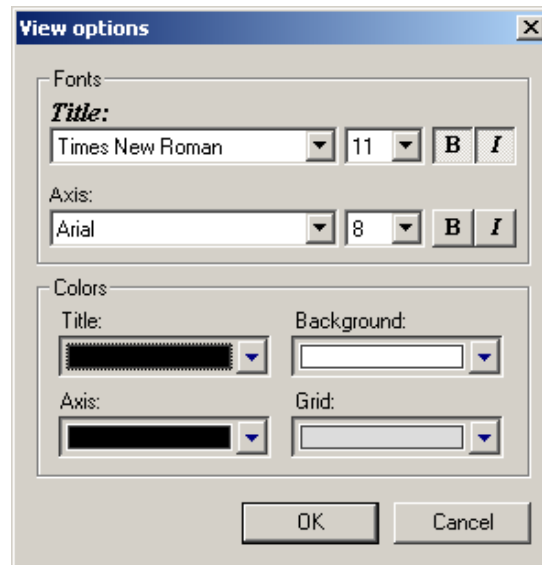
View Settings Menu

Clicking the View settings button  causes a menu to open with options for formatting the display.

- **Orient Horizontally** - changes the orientation of bus usage to horizontal. After selecting this option, the menu will say "Orient Vertically."
- **Tile Vertically** - tiles the two graphs vertically (i.e., side by side).
- **Show Markers** - Places "tick" marks along the x axis of each graph.
- **Show Plumb Line** -
- **Status** - Opens a sub-menu with the following options:
 - Bar - Displays a status bar at bottom of graph.
 - Tooltip - Causes a tooltip to appear if you position your mouse pointer over part of the graph and leave it there for a couple of seconds.
 - None - Turns off tooltips and the status bar.
- **Grid Lines** - Opens a sub-menu with the following options:
 - Both - Displays both X and Y axis gridlines
 - X Axis - Displays X axis gridlines
 - Y Axis - Display Y axis gridlines
 - None - Turns off gridlines
- **Grid on Top** - Moves the grid lines above the graph.




- **Fonts and Colors** - Opens a dialog box for setting the colors and fonts used in the graphs:

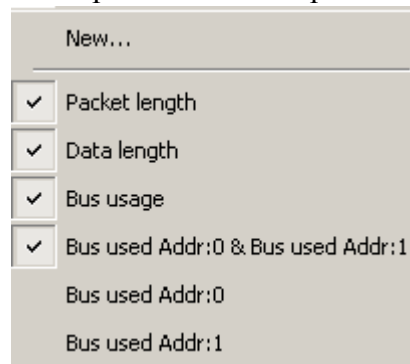


Graph Areas Menu

The Graph Areas menu allows you to view different information in the Bus Utilization window.

Step 1 Click the  button.

The Graph Areas menu opens.

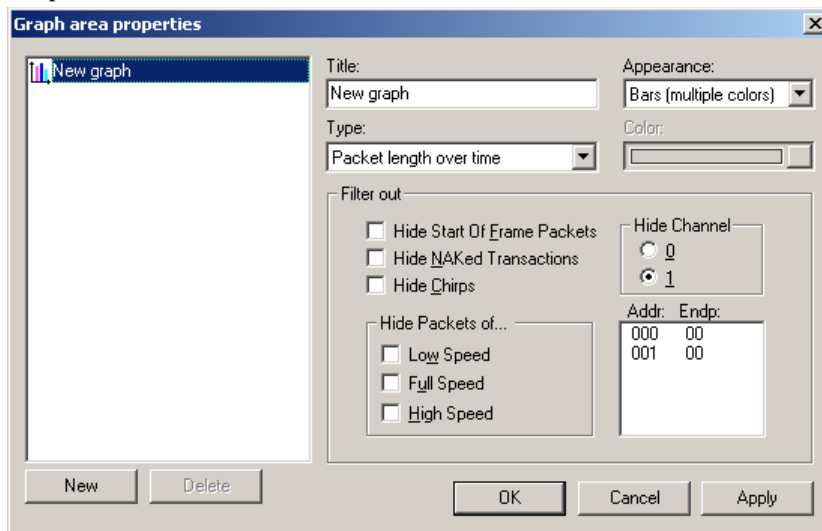


Step 2 Select the data you want to appear in the Graph Areas window.

To change the properties in the Bus Utilizations graph, follow these steps:

Step 1 In the **Graph Areas** menu, select what you want your graph to display.

Step 2 Click **OK**.




Or

To make a new graph, click **New**.

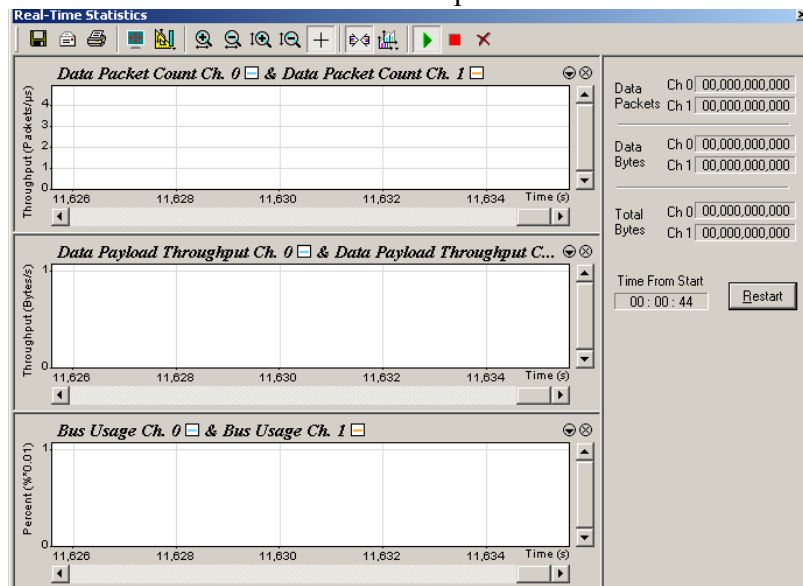
Real Time Monitoring


The Real-Time Statistics window displays a graph of real-time link activity.

Real Time Statistics displays a summary of the traffic currently being recorded by the analyzer.

To display the Real-Time Statistics window, click  in the Tool Bar.

The Real Time Statistics window opens:



In order to see a graph of traffic, you will need to start recording. After starting InfiniBand link activity, press  to start the Real-Time statistics monitor. As traffic is recorded, data will be streamed in real time to this window and presented in a format of your choice.

















To stop the monitor, press .

Real-Time Statistics Buttons

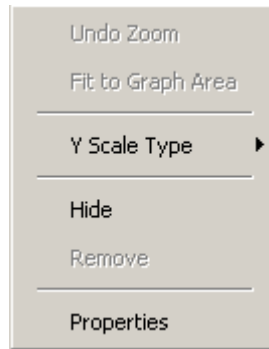
The Real-Time Statistics toolbar has buttons for changing the format of the displayed data and for exporting data:



The buttons have the following functions:

	Save As - Saves Real-Time graphs as bitmap files (*.bmp)		Vertical zoom in
	Email - Creates an email with a *.bmp file attachment of the graphs		Vertical zoom out
	Print		Click and Drag zoom - Click diagonally to select and zoom in on part of the graph
	Full Screen		Select Range
	View Settings - opens a sub-menu with options for formatting the display. See "View Settings Menu" below.		Sync and Graph areas - If two or more graphs are displayed, this button will synchronize the graphs to one another. Once synchronized, the positioning slider of one graph will move the other graphs
	Horizontal zoom in		Graph Areas - Presents options for displaying additional graphs of data lengths, packet lengths, and percentage of bus utilized.
	Horizontal zoom out		Start. Starts the Real-Time Monitor.
			Stop Real-Time Monitoring.
			Reset. Resets the graphs.

Real-Time Statistical Monitor Pop-up Menu



If you right-click a graph in the Real-Time window, a pop-up menu will appear with options for changing the format of the display.

Undo Zoom - If you have zoomed in, this command will undo the zoom

Fit to Graph Area - Redisplays graph so that the entire trace fits inside graph area

Y Scale Type -

Linear - Converts display to linear format

Logarithmic - Converts display to logarithmic format

Hide - Hides the selected graph

Properties - Opens a dialog box with options for changing the colors, titles and other features of the graphs

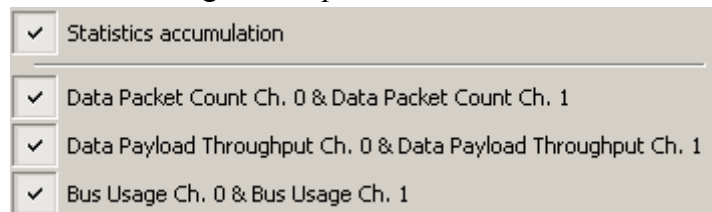
Displaying Multiple Graphs

The Real Time Statistics window gives you the ability to create up to three separate graphing windows so that you can create separate graphs of traffic and tile them vertically. Within these windows, you can format the graphs in a number of ways.

- To view two or three graphs simultaneously, click the **Graph Areas** button.



The following menu opens.



Selecting a checkbox causes the selected graph type to display. The options are:

- Statistics Accumulation** - Plots the percentage of Link utilization by non-idle traffic for both directions of the link.
- Data Packet Count (Packets/s)** - Plots counts of Data Packets per second for both directions of the link.
- Data Payload Throughput (MBytes)** - Plots data payload throughput for both directions of the link.

- **Bus Usage** - Plots to amount of Bus usage.

11. Traffic Generation with USBTrainer



USBTrainer is a USB Traffic Generator module that plugs into the right slot of the Universal Protocol Analyzer System (UPAS).

USBTrainer is designed to generate USB traffic so that engineers can test designs under realistic conditions. Traffic Generation can also be used to transmit known bad packets, providing an opportunity for engineers to observe how a device handles specific adverse conditions.

USBTrainer Traffic Generation Files

USBTrainer generates USB traffic from traffic generation files (*.utg) which are text-based script files that instruct the Generator how to generate USB traffic. These script files can be edited with either a simple text editor such as Notepad or with the **Script Editor** utility provided by the USBTracer application. The **Script Editor** utility has several aids to simplify the process of writing and editing scripts: tool-tips, drop-down menus, and colored fields.

The script example below shows the beginning of a traffic generation file created through the **Export** command. This command provides an easy way to create a generation file - you open a trace file, then run the **Export** command. The trace serves as a blueprint for the traffic generation file.

The example below shows several commented lines followed by some instructions.

```
; File C:\Documents and Settings\Administrator\Desktop\enumeration\High_Hub_Sample.usb.  
; Packets 0 to 24328.  
  
; Device Side Packets were filter out during Export  
; NAK'ed transactions were filtered out during Export  
; Saved from Channel 0.  
  
file_type=UPAS  
file_version=2  
file_speed=HIGH  
  
chirp=here ; This needs to be added by hand, since the Export function does not export it.  
  
frame=auto idle=TO_EOF  
frame=auto idle=TO_EOF  
frame=auto idle=TO_EOF  
frame=auto idle=TO_EOF  
frame=auto idle=TO_EOF  
frame=auto idle=TO_EOF  
frame=auto idle=TO_EOF  
frame=auto idle=TO_EOF
```

This generation file will cause *USBTrainer* to simulate a hub and to generate 24,320 packets. See “Device Emulation” on page 143 for details about the format of traffic generation files.

Creating Traffic Generation Files

If you choose to write a script with a text editor, a good way to start is to edit the sample generation file that is provided: *HS_Hub_Sample.utg*. This file is located in the same directory as the *USBTrainer* application (for example, C:\Program Files\LeCroy\USB Trainer). You can open this generation file with Notepad or other editor and then add or remove text as needed.

Creating a Traffic Generation File with the Export Command

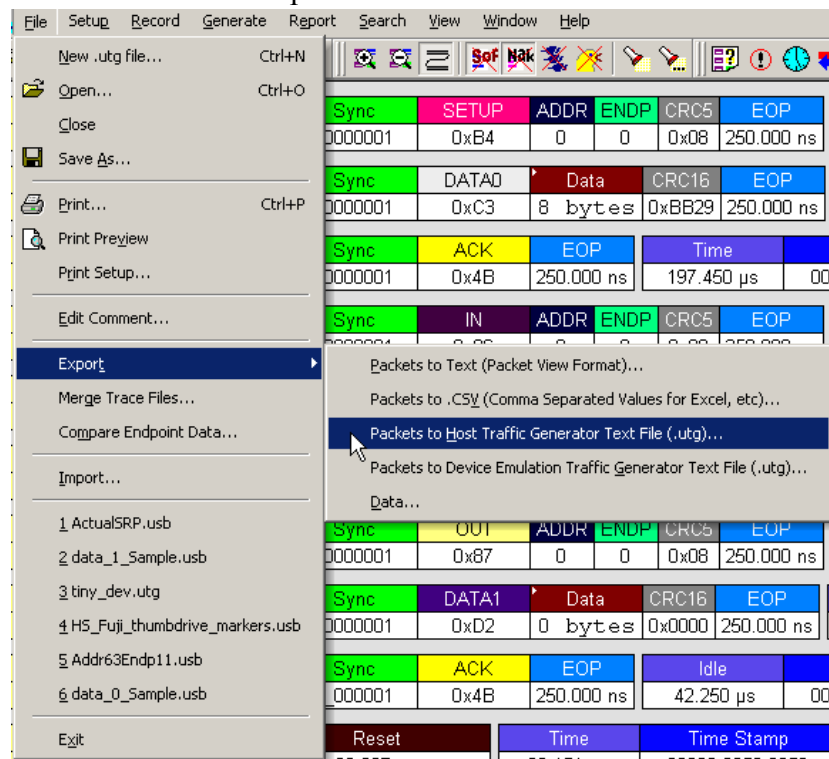
The **Export** command on *USBTracer* offers an easy alternative method of creating a generator file. This command converts the trace to a *.utg file, removes all device traffic, and leaves only the traffic from the host device. You can then use this file to emulate the host and determine whether the device under test is generating the correct traffic.

When creating a .utg file through the **Export Packets to Text (Generator Text File Format)** menu selection, it is suggested that you use the default values presented. Adding NAK transactions clutters the .utg file, and including the device side packets creates a .utg file which will not work with a real device attached.

To use the **Export** command, perform the following steps.

- Step 1 Start *USBTracer*.
- Step 2 Open a trace that has the pattern of traffic you would like to generate.
- Step 3 Select **File > Export** from the menu bar.

You see the File Export menu:

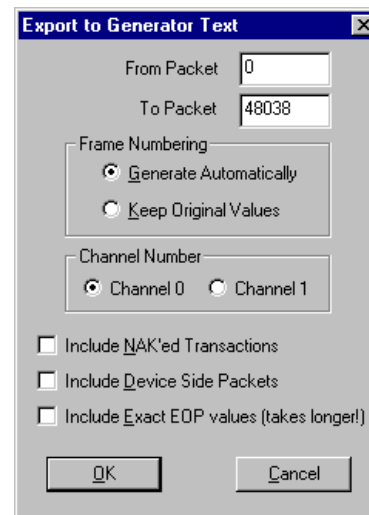


- Step 4** Select **Packets to Text (Generator Text File Format)** from the **Export** drop-down menu.

You see the **Export to Generator Text** window:

- Step 5** Enter the numbers of the first and last packets in the series.

Note The device packets are removed from the exported generator text. This is essential in creating a generator text file that can be used to handshake with your device.




- Step 6** You can opt to regenerate the frame numbers and remove the NAKed transactions.

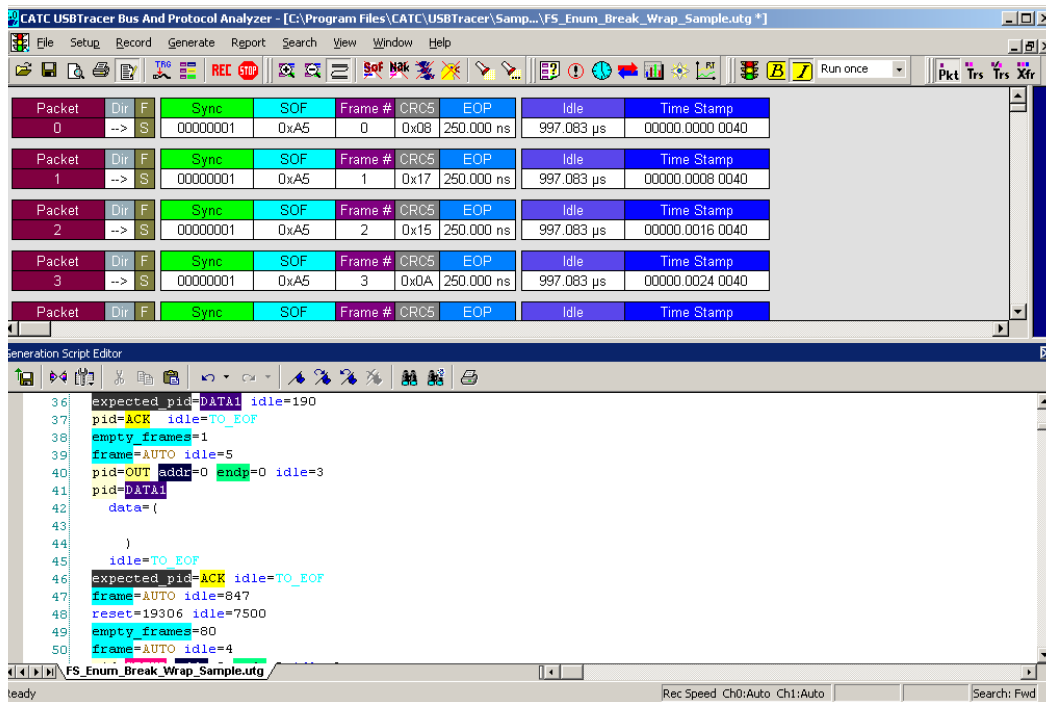
Once the generator text file is exported, you may need to edit the file and adjust idle time to properly anticipate the responses from your device.

11.1 Editing a Generation File

A .utg file is a text file that can be edited with any text editor such as Notepad. A better editing option, however, is *Script Editor*, an utility that is built into the USBTracer application. *Script Editor* provides the usual editing functions such as select, cut, copy, and paste but also adds tool-tips, colored keywords, drop-down parameter values, and expandable/collapsible packet data fields.

To launch the Script Editor, click the Script Editor button on the toolbar  or right-click in the trace window and choose **Edit as Text**.

The Script Editor window will open and display in the lower portion of the trace window.



The Script Editor divides into three areas: the toolbar, the script window, and the file tabs at the bottom of the window. If errors occur, a log will open at the bottom of the window.

Toolbar

The toolbar contains buttons for saving your edits, navigating, searching and other functions.



The buttons have the following functions:



Save. Saves your edits and immediately updates the setting bars and Frames shown in the trace window.



Add/Remove bookmark. Allows markers to be set or removed to aid in navigation.



View Options. Opens a menu with three options: Enable Outlining, Toggle Outlining, and Line Numbers. See **View Options Menu** below for descriptions.



Go to next bookmark.



Cut.



Go to previous bookmark.



Copy.



Clear all bookmarks.



Paste.



Find.



Undo.



Find and Replace.



Redo.

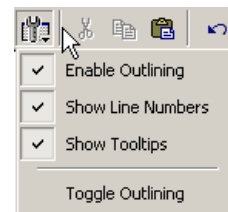


Print.

View Options Menu

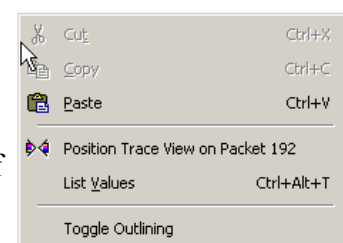
The View Options button has a menu with three options:

- **Enable Outlining** - Adds an expandable/collapsible tree structure to the left side of the Script Editor showing the hierarchical relationships of the script lines.
- **Show Line Numbers** - Adds line numbers to the left side of the Script Editor window.
- **Show Tooltips** - Enables tooltips to appear when the mouse pointer is suspended over a script item.



Pop-up Menu

Right click anywhere in the script window to open a pop-up menu with the following options: Cut, Copy, Paste, Position Trace View on Packet xxx, List Values, and Toggle Outlining. The List Values option will display the types of values that can be entered for a parameter in a line. To see the types of values, select the current parameter, then choose **Show Values** from the pop-up menu.



File Tabs

At the bottom of the window is tab that shows the name of the .utg file. If your .utg file has an Include statement in it, the supporting Include files will automatically open when the .utg file is first opened. Tabs for the opened Include files will appear at the bottom of this window.



Error Log

Whenever you create a scripting error, a log will open at the bottom of the application window. When the error is corrected, the window automatically closes.

Tooltips

The Script Editor window includes extensive tooltips for each keyword. To see a tooltip, hold the mouse pointer over a keyword.

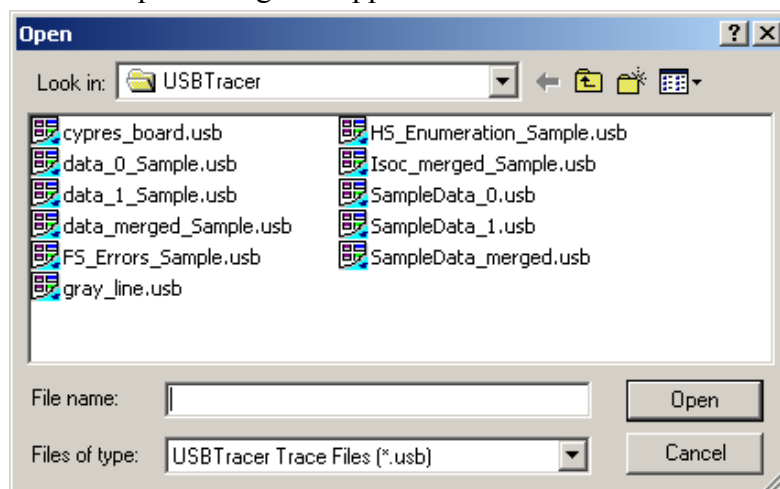
11.2 Loading the Generation File

The USB Traffic Generation files are scripts that instruct the analyzer how to generate USB traffic. A traffic generation file contains text in special format and is named with a "*.utg" extension. These files can be created by any text editor, or using the **File >Export >Packets to Text (Generator File Format)** menu selection when viewing a Trace File. There are several examples of Traffic Generation files included with the installation of the USBTracer software.

To load a generation file,

Step 1 Select **File >Open ...** from the menu.

The Open dialog will appear.



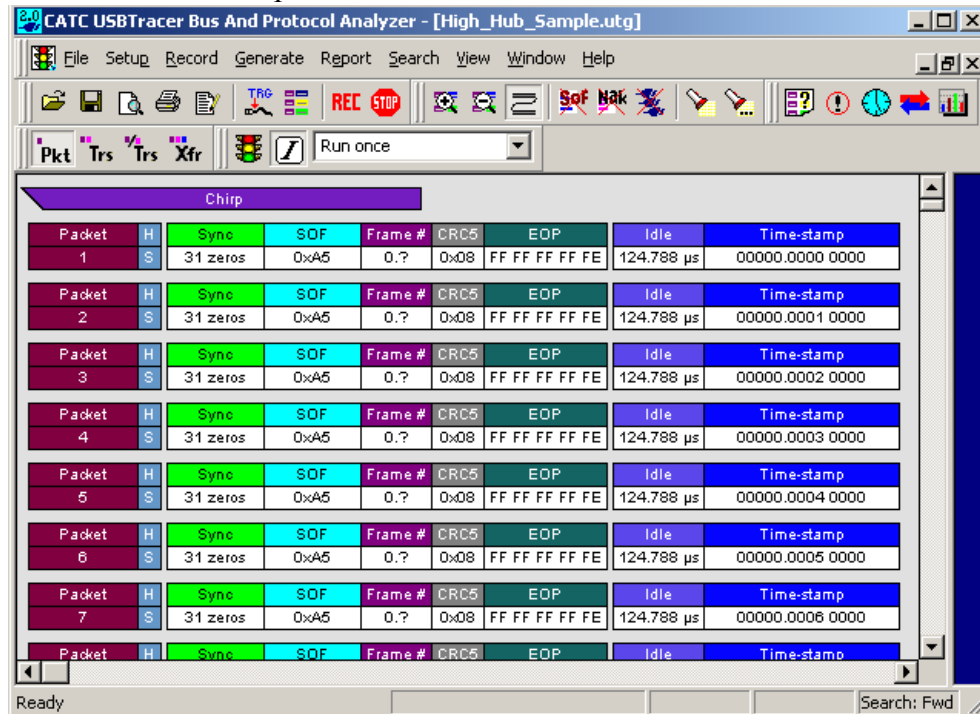
- Step 2 Select **USB Tracer Generation Files (*.utg)** from the drop-down menu marked **Files of Type**.

A list of Traffic Generation files will display.

- Step 3 Select a Traffic Generation file (*.utg).

- Step 4 Click **Open**.


The file opens.



- Step 5 Decide how many times you want the traffic pattern to be generated, then select a value from the **Repeat** drop-down menu.



The **Repeat mode** allows for a single pass through the generation file, looping forever, or looping 1 to 65,534 times, as desired by the user. If a "wrap=HERE" location is found in the .utg file, that will be the location at which the looping portion begins. The loop end is at the end of the .utg file or up to the "stop=HERE" statement in the file.

- Step 6 Decide what **Generation mode** you wish to generate traffic in (see below for an explanation), then depress or undepress the IntelliFrame button on the toolbar .

Depressed = IntelliFrame

Undepressed = Bitstream


Traffic Generation Modes: Bitstream vs. IntelliFrame

IntelliFrame and **Bitstream** are modes that control how the generator interacts with other devices when it is generating traffic.


- **Bitstream Mode** - In Bitstream Mode, the generator constructs a bitstream of traffic based on the UTG file and assumes that packets start at pre-calculated times. For example, after an IN packet, the generator will wait a calculated fixed time before presenting an ACK. The time between them is determined by the "idle=" time statement in between the IN and ACK packets in the .utg file. In this mode, every bit time of a generation stream (including idles and SE0's) is represented by 4 bits of data, so the file to be downloaded to the Generator can be very large.
- **IntelliFrame Mode** - In IntelliFrame mode, the generator can wait for Device responses to complete before it continues generating more host packets. For example, after issuing an IN, the generator looks for the DATAx packet issued by the device to finish, and then issues an ACK. This way, the data length can vary and does not need to be pre-calculated. Also, using the "idle=TO_EOF" statement allows the generator to calculate where the end of the frame occurs, so that a subsequent "frame=AUTO" statement will create a Start of Frame at the correct time. Moreover, the Generator can be made to retry transactions that are NAKed automatically, and to use the PING protocol where appropriate. Retries can be made to occur within the same frame or in the next frame. For further information on the flexible behavior that can be instituted during the generation, see the syntax descriptions for the generation files here. An additional advantage of the IntelliFrame mode is that it creates an image in the Generator's memory that is MUCH smaller than that of the Bitstream mode, so it downloads much faster.

Note Device emulation only works in IntelliFrame mode.

Starting Traffic Generation

To start traffic generation, click  on the Tool Bar.

Repeating a Generation Session

If you wish to repeat a generation session, press the Traffic Generation Start/Stop  button again. There is no need to reload the .utg file.

You can make an additional entry in a Generator Text File that allows you to specify a portion of the file for repetition:


- Enter **wrap=here** in the Generator Text File.

The traffic above the entry is run only once. The traffic below the entry is repeated continuously.

Note For a usage example, see the sample file **Wrap.utg**.

Stop Traffic Generation

To instruct the Analyzer to halt traffic generation,

- Click  on the Tool Bar.

11.3 Device Emulation

Device Emulation is a licensed option that allows USBTrainer to generate device-side traffic. Like host traffic generation, device emulation uses text-based generation files (.utg) to generate traffic.

Device emulation has three steps: create a generation file, configure generation settings, and then generate traffic.

Creating a Generation File

Generation files can be created one of two ways:

- 1 Write a generation script file using either LeCroy's context-sensitive script editor or any text editor that you choose or;
- 2 Use a pre-recorded trace file that has the type of traffic that you want to generate. This file will serve as a blueprint for the traffic generation script file.

If you decide to use a pre-recorded trace file as a generation blueprint, then verify that the trace file contains traffic for only one device. If it does not, hide all the other devices (Setup > Level Hiding) and save the file as a new file without the hidden traffic (File > Save As and check the option Do not save hidden packets/transactions/transfers).

Setting Generation Options

Step 1 Set the "Traffic Generation" options by configuring the Misc page in the Recording Options dialog box:

Setup > Recording Options > Misc

Step 2 In the Misc page, select **Device Emulation Mode**.

Step 3 Configure Resume settings.

- 1 If you want the simulated device to issue Resumes, then select **Device Resumes** and enter a delay time (in milliseconds). If you do not select Device Resumes, then the emulated device will wait for a Host to issue a Resume.
- 2 If you want the simulated device to take its address from the Traffic Generation (.utg) script file, then select **Use Address in .utg file**. This option causes the USBTrainer to read the .utg file and assign a Device Address based on the device traffic that it sees in the file.

Note In this release, the Device Emulator does not look at the Set Address in the script- so if your host controller is running more than one device, it may enumerate your device emulator incorrectly when you begin to generate traffic. The solution is to manually assign a device address. To manually configure the Device Address, select Emulated Device's Hex Address and enter an address or mask (note: a mask will allow the USBTrainer to respond to multiple device addresses).

Step 4 Click **OK** to close the Recording Options dialog and apply the changes. The generation settings will take effect as soon as you execute a script.

Run the Traffic Generation Script File

Step 1 Connect the USBTrainer's port "B" (either Hi Speed or Classic Speed) to the Host.

Step 2 Open the "Traffic Generation" .utg file.

Step 3 Begin traffic generation by clicking .

When execution begins, USBTrainer reads the entire generation file and then parses the generation commands into groups according to their endpoints. Each of the parsed groups of commands are then written into dedicated memory segments in the generator. **Up to eight memory segments can be created** for the various endpoints. In the case of Endpoint 0 or any Control endpoints, a single memory segment is created.

Each memory segment can be thought of as a queue of the commands and responses for a particular endpoint that occur in the generation file.

Commands are stored sequentially within each memory segment as they occur in the generation file. For example, if a generation file has a sequence of command X and Y for the Endpoint 1, the commands are extracted in this order and placed in the Endpoint 1 memory segment.

When the host calls for a particular device endpoint response, the first command listed in that endpoint memory segment is then executed. With each subsequent call to the same endpoint, commands are executed sequentially down the list as they occur in the memory segments. For example, if three calls were made to Endpoint 0 IN, then the first three commands in that endpoint's memory segment would then execute.

Note **Each of the endpoint memory segments execute independently: there are no behavioral interactions between the individual endpoints.**

11.4 Format of the Traffic Generation Files

Whether you create a traffic generation from scratch or use the **Export** command in *USBTracer*, there are a few rules about the format of the generation file that you should review. These are presented in this section.

Traffic is generated from a Text file named ***.utg**.

Within the text file,

- The data format for data fields such as Pids and Raw bits is hex.
- Each packet definition consists of this set of assignments:
key=value.
- White space is permitted around the equal sign.
- There are no restrictions on dividing packet definition to lines nor is it necessary to define each packet in a separate line, although it is recommended.
- The maximum number of characters in a line is 250.
- The characters # and; indicate end-of-line comments (i.e. the rest of the line is ignored)
- Comments are not allowed within brackets ().
- No keys or values are case-sensitive.

Each packet definition starts with one of these assignments:

- **pid=N** (where N is a string representing a valid packet identifier or an eight bit value)

OR

- **frame=N** (where N is an eleven bit value of the frame number or the strings **auto** and **keep_alive**).

A bus condition definition starts with one of these assignments:

- **reset=N** (where N is a positive integer or string)
- **suspend=N** (where N is a positive integer)
- **resume=N** (where N is a positive integer)
- **chirp=here**

Subsequent assignments after a packet's starting assignment define the values of particular fields within the packet. If a field is not defined, it is assumed to be **0**. Values are assumed to be decimal unless they are prefixed with **0x** and then are interpreted as hexadecimal. Values within a data block assignment are always assumed to be hexadecimal and should not be prefixed by **0x**.

Table 3: Support Keys for Defining Fields within a Packet

Key Code	Format	Description
File Control Keys		
file_type	'UPAS' 'CHIEF'	This must be included at the beginning of the file to determine the speed of packets to be generated. Value must be "file_type=UPAS" to allow for High speed traffic, IntelliFrame operation, 'expected_pid' and 'device_pid' key support. Full and Low speed traffic, and Bitstream operation are supported on all "file_type=" values. If this statement does not appear in the file, the default is "file_type=CHIEF", allowing backward compatibility with older USB Chief .usb files.
file_version	integer	This must be included after the "file_type=" key to determine the version of this file. Value is currently 3. Usage: file_version=4
file_mode	'HOST' 'DEVICE'	Sets generation mode to host or device. For generation to work, this must match the selection found in the Recording Options/Misc dialog.
file_speed	'HIGH' 'HI' 'FULL' 'LOW'	This must be included after the "file_version=" key to determine the speed of packets to be generated. Values are "HIGH", 'HI', 'FULL', or 'LOW'. Example:file_speed=FULLOnly one "file_speed=" is allowed per .utg file, but low speed traffic on a full speed bus can be created by adding the "speed=LOW" key to a low speed packet in a file defined as "file_speed=FULL". No other mixing of speeds is allowed. 'HI' is the same as 'HIGH', and was added to conform to the USB terms.

Key Code	Format	Description
loop_count wrap_count	0 through 16382 or 'infinite'	These two terms are interchangeable. Device Emulation Only. This key defines the loop count for each memory segment. Examples: loop_count=INFINITE or wrap_count=9
wrap loop	'HERE'	This key marks the "wrap point" in the traffic. When generation is in Repeat mode, all the traffic before the wrap point is going to be sent once, all the traffic after the wrap point is going to be repeated in a loop. The usage of the keyword is: wrap=HERE or loop=HERE For Device Emulation, the value would be a memory segment number, so that the looping is associated with only that one endpoint. wrap=4
skip	'HERE'	Causes a region in the .utg file to be ignored, as if commented out. Can be used multiple times in the file. Must be used in conjunction with skip_end=HERE. Example: skip=HERE frame=auto # this statement will now be ignored! skip_end=HERE
skip_end	'HERE'	Causes a region in the .utg file to be ignored, as if commented out. See above in skip=HERE.
stop	string	If you want to run only some first portion of the beginning of a .utg file, insert this statement where you want generation to halt. This saves having to edit a file into smaller files when testing a portion of traffic. The usage of the keyword is: stop=HERE
break	'HERE'	Host Generation Only. This enables the user to generate up to a point in the .utg file, then wait for user input before continuing in the file. During the breakpoint time, the traffic signal icon in the toolbar will flash yellow, indicating that a breakpoint was hit. When the user clicks on the traffic light icon, the generation will resume. If the user wants to stop rather than continue, he can select Stop from the Generation menu, or hit the Start/Stop button on the front of the generator module. Start-of-frames are issued automatically during the breakpoint duration. When the user resumes running, traffic will begin after another start-of-frame is issued. The "break=HERE" statement must be inserted between "frame=xxx" statements. Syntax example: frame=AUTO break=HERE frame=AUTO
Endpoint Configuration		

Key Code	Format	Description
begin_config	'HERE'	Defines the beginning of the region in the file used to configure the endpoint types. The configuration is necessary to determine the default behavior of the NAK retry mechanism during IntelliFrame operation. Between the "begin_config=HERE" and the "end_config=HERE" statements will be a series of "config_endpoint=xxx" statements which define each endpoint's type and default retry behavior. The config region must precede any actual packet or bus condition statements. For Device Emulation, the config region is mandatory. It provides the mapping of the endpoints into their corresponding segments of analyzer memory which contain the traffic for those endpoints. When exporting to .utg file from a trace file, these sections will be created automatically. Syntax example: begin_config=HERE
end_config	'HERE'	This terminates the region in the file used for configuring the endpoints. See 'begin_config'. Syntax example: end_config=HERE
config_endpoint	'CONTROL' 'INTERRUPT' 'ISOCRONOUS' 'BULK'	This begins a statement that defines a particular endpoint's type, default retry behavior (Host Generation only), and memory segment (Device Emulation only) Usage example: config_endpoint=BULK addr=1 endp=3 direction=OUT retry=TRUE retry_next_frame=FALSE
endp_mem_seg	1 2 3 4 5 6 7 8	Device Emulation Only. This key separates each endpoint function into a different queue (also referred to as a Memory Segment) of commands and responses. There are a maximum of eight of these queues and each has a unique address/direction combination. For control endpoints, one queue is shared by both directions of the endpoint. Traffic on the Default Endpoint (Address 0, Endpoint 0) will share the same queue as the endpoint 0 of the selected device address (the address it gets from the Host through the SetAddress request). These will always use endp_mem_seg=1. These will all be set automatically when exporting a trace file to a Device Emulation .utg file.
direction	'IN' 'OUT'	This defines the transfer direction of data for the specified endpoint. For all transfer types but CONTROL, there can be 2 distinct logical connections using the same address and endpoint. They would differ only in defined direction. Syntax example: direction=IN

Key Code	Format	Description
retry	'TRUE' 'FALSE'	<p>Host Generation Only. This is used to define whether an automatic retry should be performed on packets to/from this endpoint in the case that they are NAKed (or in some cases 'NYETed'), or if a timeout on device response occurs.</p> <p>RETRY only works for the Pids: Setup, In, and Out.</p> <p>If 'TRUE', the Trainer automatically retries the specified sequence if a Pid is received on the bus which does not match the 'expected_pid', or if a timeout occurs. The generator will re-issue the host packet(s) after waiting either 1/10th of a frame, or until after the next start-of-frame (depending on the value set for 'retry_next_frame') The retries continue until the 'expected_pid' is received. When a retry attempt finds its 'expected_pid', the Trainer automatically generates a new start-of-frame before continuing with the rest of the Gen File. (The start-of-frame is either an SOF packet or a keep-alive signal.)</p> <p>If 'FALSE', the Trainer waits for a pid before proceeding. There is no timeout. If a pid is received which does not match the 'expected_pid', the Trainer continues to wait for the correct pid to appear. If the user stops the generation, he will be informed that the generator was waiting patiently for the expected_pid, and it never showed up. The user will have to examine the problem by viewing the trace file.</p> <p>This statement can also be used in conjunction with any "pid=xxx" statement in the .utg file. This can be done to override the configured or default retry behavior.</p> <p>Syntax example: retry=TRUE</p>
retry_next_frame	'TRUE' 'FALSE'	<p>Host Generation Only. This determines when a retry will be attempted after a failed match of an 'expected_pid'. This statement only applies if "retry=TRUE" statement also exists.</p> <p>If 'TRUE', the Trainer waits until the current frame completes, issues a start of frame, and then retries the transaction.</p> <p>If 'FALSE', the Trainer waits 1/10th of a frame before retrying the transaction. Before each retry attempt, the Trainer checks to see where in the frame interval it is. If it is too close to the EOF, it automatically generates the next start-of-frame before performing the retry. If it is not too close to the EOF, it will perform the retry without generating a new frame.</p> <p>This statement can also be used in conjunction with any "pid=xxx" statement in the .utg file. This can be done to override the configured or default retry behavior.</p> <p>Syntax example: retry_next_frame=FALSE</p>

Key Code	Format	Description
ping_on_retry	'TRUE' 'FALSE'	Host Generation Only. This determines whether an OUT transaction is retried by repeating the OUT-DATAx sequence, or whether a PING sequence should be initiated. If 'TRUE', the Trainer automatically generates PINGs (and retries) before going through a Retry Loop. PINGs are repeated until an ACK is received, and then the original OUT-DATAx transaction Sequence is attempted. During the PING retries, the trainer automatically generates frames. When the original Retry Sequence receives its 'expected_pid', the Trainer generates a new frame before continuing with the Gen File. If 'FALSE', the original transaction is always retried. This statement can also be used in conjunction with any "pid=xxx" statement in the .utg file. This can be done to override the configured or default retry behavior. Syntax example: ping_on_retry=TRUE
ping_after_nyet		Host Generation Only. If 'TRUE', the Trainer automatically generates PINGs (and retries) after receiving a NYET instead of the 'expected_pid'. This feature is only used if the 'expected_pid' is an ACK. PINGs are retried until an ACK is received. When the PING is ACKed, the Trainer will generate a new frame before continuing with the Gen File. If 'FALSE', the Trainer continues as if an ACK occurred. This statement can also be used in conjunction with any "pid=xxx" statement in the .utg file. This can be done to override the configured or default retry behavior. Syntax example: ping_after_nyet=TRUE

Key Code	Format	Description
Packet Starting Keys		
pid	8 bits (0-0xFF) or pid string	<p>Host Generation Only. Use this as the first key of most packets sent by the Host (Exception: use "frame=" for SOF packets). The key should be assigned to a valid packet identifier string per the USB specification: 'SETUP', 'IN', 'OUT', 'DATA0', 'DATA1', 'ACK', 'PRE', 'PING', 'SPLIT', 'DATA2', or 'MDATA'. Optionally, you may assign this key a raw eight bit value to force an error condition.</p> <p>Warning: If you specify PID=0xNN, you must use raw_data = () to specify the rest of the packet data, since the packet structure will be unknown.</p>
expected_pid	pid string	<p>Host Generation Only. Use this as the PID key for packets which are expected to be sent by the device. In IntelliFrame mode, the generator will wait until this PID has completed before sending the next generated packet or bus condition. The key should be assigned to a valid packet identifier string per the USB specification: 'DATA0', 'DATA1', 'ACK', 'NAK', 'STALL', 'NYET', 'DATA2', or 'MDATA'. The generator engine will wait forever until this expected_pid appears, so the user may have to hand edit the file to achieve the desired results. For example, if a NAK comes where the expected_pid was a DATA1, the user should edit the .utg file to move the "pid=IN" command to later in the file by inserting some "frame=AUTO idle=TO_EOF" pairs before it. This will allow time for the device to be ready for the IN.</p> <p>If RETRY=TRUE for this address/endpoint, the NAKs will be ignored and the SOF's will be generated automatically until the expected PID occurs.</p>
device_pid	pid string	<p>Use this as the first key of most packets sent by a device. The key should be assigned to a valid packet identifier string per the USB specification: 'DATA0', 'DATA1', 'ACK', 'NAK', 'STALL', 'NYET', 'DATA2', or 'MDATA'. Normally, the "device_pid=" statement is NOT present in a Host Generation .utg file, since the device responses are intended to come from real devices. By default, Host Generation files exported from Trace Files do NOT include "device_pid=" statements.</p> <p>Warning: If you specify PID=0xNN, you must use raw_data = () to specify the rest of the packet data, since the packet structure will be unknown.</p>

Key Code	Format	Description
frame	11 bits or 'AUTO' 'KEEP_ALIVE'	This key creates a start of frame packet and will generate a SOF PID as expected. The key should be assigned a value of the frame number, 'AUTO', or 'KEEP_ALIVE'. The string 'AUTO' instructs the generator to increment the frame number automatically. The string 'KEEP_ALIVE' instructs the analyzer to generate a low-speed EOP in place of a SOF packet for traffic on a low-speed branch (file_speed=LOW). For Device Emulation, this statement is ignored by the Device Emulator, but serves to organize the .utg file in a more readable manner.
empty_frame	integer	Host Generation Only. This key creates a sequence of start of frame packets with "idle=TO_EOF" values for the idle time. This results in N empty frames, where N is the integer value specified. If the branch speed is LOW, the frames will contain only the keep-alive standalone EOP's. This key makes for an easier to manage .utg file by eliminating the need for many lines of "frame=AUTO idle=TO_EOF" statements. Usage: empty_frames=23 ; insert 23 empty frames here
host_exp_pid	PID	Device Emulation Only. This key defines the PID that is expected to be received from the Host. The Device Emulator will wait until this PID has completed before sending the next generated packet. The key should be assigned to a valid packet identifier string per the USB specification: 'SETUP', 'IN', 'OUT', 'DATA0', 'DATA1', 'ACK', 'PING', 'DATA2', or 'MDATA'. 'PRE' and 'PING' are for Hubs only, and are not supported. Each memory segment will wait forever until it receives the expected PID to it's address/endpoint, so the user may have to hand edit the file to achieve the desired results. Each endpoint memory segment acts independently.
Bus Condition Keys		
reset	positive integer or 'LS_EOP'	Host Generation Only. Assign a positive integer to this key that indicates the number of microseconds that single-ended zeros (SE0) are driven onto the bus. This key can also be assigned the string 'LS_EOP' to drive two low-speed bit times of SE0 followed by one bit time of J.
se0	positive integer <2500	Host Generation Only. Same SE0 signal as reset, but the range is in nanoseconds (accuracy: +/- 33ns).
suspend	positive integer	Host Generation Only. Assign a positive integer to this key that indicates the number of microseconds of idle and suspend after the previous packet (e.g. for ten milliseconds of suspend the key should equal 13000). Note that suspend begins after 3 milliseconds of idle.

Key Code	Format	Description
resume	positive integer	Host Generation Only. Assign a positive integer to this key that indicates the number of microseconds of K driven onto the bus. For a proper resume sequence this should be followed by the condition "reset=LS_EOP speed=LOW".
wait_resume	HERE	Host Generation Only Place this immediately before a resume=<> statement to implement a device resume, also known as Remote Wakeup. This will cause the host to wait for the device to issue the Resume (K) condition before the Host proceeds with it's own Resume signalling. This statement MUST be followed by the resume=<> statement.
chirp	'HERE'	Used to create a chirp sequence for a High Speed generation. Usage: chirp=HERE
wait_vbus	'VALID'	Device Emulation Only. Wait for VBus to go Hi
termination	'HERE'	Device Emulation Only. Set terminations now
Keys for Packet Fields		
speed	'LOW'	Used to cause a low speed packet on a full speed branch. The only legal value is "speed=LOW".
addr	7 bits (0-127)	Assign a value for the address field.
endp	4 bits (0-15)	Assign a value for the endpoint number field.
hub_addr	7 bits (0-127)	Host Generation Only. Assign a value for the hub_address field of a SPLIT packet.
port	7 bits (0-127)	Host Generation Only. Assign a value for port field of a SPLIT packet.
data	(AB CD)	Assign data bytes for the data field. Use the following syntax: data=(12 34 56 78 90 AB CD EF DC 13 40 78 11 CA 70 65). You can wrap bytes of the data field to the next line. The bytes are in the order they come across the bus, and the bits within the bytes are msb -> lsb order. Maximum number of bytes allowed is 1049.
crc	5 bits (0x0-0x1F) or 16 bits (0x0-0xFFFF)	Assign a value for the crc field. The default value is the correct crc calculated for the packet.
s	0 or 1	Host Generation Only. Assign a value for the s (High Speed SPLIT Start/Speed) field.
sc	0 or 1	Host Generation Only. Assign a value for the sc (High Speed SPLIT Start/Complete) field.

Key Code	Format	Description
e	0 or 1	Host Generation Only. Assign a value for the e (High Speed SPLIT End) field.
et	0-3	Host Generation Only. Assign a value for the et (High Speed SPLIT Endpoint Type) field. 0-Control, 1-Isoch, 2-bulk, 3-Interrupt
eop	positive integer	Assign a positive integer for the length of the end of packet (EOP). The default value is the correct length (2 bits of SE0, 1 bit of J). The value will be reflected by <val-1> bits of SE0 plus one bit of J.
hi_eop	(AB CD) Hex	Used to generate a high speed EOP pattern. The bytes are in the order they come across the bus, and the bits within the bytes are lsb ->msb order. Maximum number of bytes allowed is 13.
idle	positive integer or 'TO_EOP'	Assign a positive integer or string to this key defining the length of idle after the current packet. The positive integer indicates the number of full-speed or low-speed bit times. The default value is around 4 bits for Classic speeds, and around 150 bits for High speeds. When this keyword is assigned a string 'TO_EOF' the USB Chief automatically calculates the amount of idle remaining in the frame. When handshaking with a device you can use this keyword to add idle where you anticipate a packet being returned from the device. The "idle=" key is always associated with the proceeding packet or bus condition, so you should not put file control keys in between the packet or bus condition statement and the "idle=" statement.
marker	'string'	Assign a string in quotation marks to be put in the marker for the defined packet.
raw_data	(aa bb)	This statement is used if you want a non-modulo 8 number of bits sent in a classic speed 'raw_data' packet. It is only valid in conjunction with the 'raw_data' statement. Normally, every bit of the array defined in "raw_data=(xx yy)" type of statement would be sent. However, particularly in the case where "bit_stuff=off", you need more control over the number of bits to be output on the bus. By adding the statement "raw_data_bits = 11" only the first 11 bits of the defined bytes will be sent. For the following case, raw_data=(8F F0) raw_data_bits=13 the output bitstream would be: 1111 0001 0000 1 (Recall that each byte is described in msb->lsb format, but it is output on the wire in lsb->msb format.)

Key Code	Format	Description
raw_data_bits	integer	This statement is used if you want a non-modulo 8 number of bits sent in a classic speed 'raw_data' packet. It is only valid in conjunction with the 'raw_data' statement. Normally, every bit of the array defined in "raw_data=(xx yy)" type of statement would be sent. However, particularly in the case where "bit_stuff=off", you need more control over the number of bits to be output on the bus. By adding the statement "raw_data_bits = 11" only the first 11 bits of the defined bytes will be sent. For the following case, raw_data=(8F F0) raw_data_bits=13 the output bitstream would be: 1111 0001 0000 1 (Recall that each byte is described in msb->lsb format, but it is output on the wire in lsb->msb format.)
bit_stuff	'OFF' 'ON'	Assign the string OFF to disable bit stuffing within the current packet. The default value is ON .
sync	integer 1-40	Assign an integer from 1 to 40 to represent the sync field. The integer is the number of zeros transmitted before the one. The default value is 7 to give a sync of '00000001' for Low and Full Speed files, and 31 for High Speed files, resulting in a sync of 00000000000000000000000000000001.
retry ping_on_retry ping_on_nak retry_next_frame	'TRUE' 'FALSE'	Host Generation Only. These four may be applied to individual packets, to override the behavior in a given instance. Normally, they are used in the 'config_endpoint' statement. See their descriptions in the Endpoint Configuration section earlier.
Keys for Class Decoding		
These keys are populated into the utg. script file automatically whenever the menu command File > Export to .utg file is run. These keys do not need to be edited.		
sd_prod		
sd_vend		
sd_bm_req_type		
sd_interface		
sd_host_id		
sd_class_code		
sd_subclass		
sd_protocol		
sd_end_sd		

How to Contact LeCroy

Type of Service	Contact
Call for technical support...	US and Canada: 1 (800) 909-2282
	Worldwide: 1 (408) 727-6600
Fax your questions...	Worldwide: 1 (408) 727-6622
Write a letter...	LeCroy Customer Support 3385 Scott Blvd. Santa Clara, CA 95054
Send e-mail...	support@catc.com
Visit LeCroy's web site...	http://www.lecroy.com/

Limited Hardware Warranty

So long as you or your authorized representative ("you" or "your"), fully complete and return the registration card provided with the applicable hardware product or peripheral hardware products (each a "Product") within fifteen days of the date of receipt from LeCroy or one of its authorized representatives, LeCroy warrants that the Product will be free from defects in materials and workmanship for a period of three years (the "Warranty Period"). You may also complete your registration form via the internet by visiting <http://www.catc.com/support/register/>. The Warranty Period commences on the earlier of the date of delivery by LeCroy of a Product to a common carrier for shipment to you or to LeCroy's authorized representative from whom you purchase the Product.



What this Warranty Does Not Cover

This warranty does not cover damage due to external causes including accident, damage during shipment after delivery to a common carrier by LeCroy, abuse, misuse, problems with electrical power, including power surges and outages, servicing not authorized by LeCroy, usage or operation not in accordance with Product instructions, failure to perform required preventive maintenance, software related problems (whether or not provided by LeCroy), problems caused by use of accessories, parts or components not supplied by LeCroy, Products that have been modified or altered by someone other than LeCroy, Products with missing or altered service tags or serial numbers, and Products for which LeCroy has not received payment in full.

Coverage During Warranty Period

During the Warranty Period, LeCroy or its authorized representatives will repair or replace Products, at LeCroy's sole discretion, covered under this limited warranty that are returned directly to LeCroy's facility or through LeCroy's authorized representatives.

How to Obtain Warranty Service

To request warranty service, you must complete and return the registration card or register via the internet within the fifteen day period described above and report your covered warranty claim by contacting LeCroy Technical Support or its authorized representative.

You can reach LeCroy Technical Support at 800-909-7112 or via email at support@catc.com. You may also refer to the LeCroy website at <http://www.lecroy.com> for more information on how to contact an authorized representative in your region. If warranty service is required, LeCroy or its authorized representative will issue a Return Material Authorization Number. You must ship the Product back to LeCroy or its authorized representative, in its original or equivalent packaging, prepay shipping charges, and insure the shipment or accept the risk of loss or damage during shipment. LeCroy must receive the Product prior to expiration of the Warranty Period for the repair(s) to be covered. LeCroy or its authorized representative will thereafter ship the repaired or replacement Product to you freight prepaid by LeCroy if you are located in the continental United States. Shipments made outside the continental United States will be sent freight collect.

Please remove any peripheral accessories or parts before you ship the Product. LeCroy does not accept liability for lost or damaged peripheral accessories, data or software.

LeCroy owns all parts removed from Products it repairs. LeCroy may use new and/or reconditioned parts, at its sole discretion, made by various manufacturers in performing warranty repairs. If LeCroy repairs or replaces a Product, the Warranty Period for the Product is not extended.

If LeCroy evaluates and determines there is "no trouble found" in any Product returned or that the returned Product is not eligible for warranty coverage, LeCroy will inform you of its determination. If you thereafter request LeCroy to repair the Product, such labor and service shall be performed under the terms and conditions

of LeCroy's then current repair policy. If you chose not to have the Product repaired by LeCroy, you agree to pay LeCroy for the cost to return the Product to you and that LeCroy may require payment in advance of shipment.

General Provisions

THIS LIMITED WARRANTY GIVES YOU SPECIFIC LEGAL RIGHTS. YOU MAY HAVE ADDITIONAL RIGHTS THAT VARY BY JURISDICTION. LECROY'S RESPONSIBILITY FOR DEFECTS IN MATERIALS AND WORKMANSHIP IS LIMITED TO REPAIR AND REPLACEMENT AS SET FORTH IN THIS LIMITED WARRANTY STATEMENT. EXCEPT AS EXPRESSLY STATED IN THIS WARRANTY STATEMENT, LECROY DISCLAIMS ALL EXPRESS AND IMPLIED WARRANTIES FOR ANY PRODUCT INCLUDING, BUT NOT LIMITED TO, ANY IMPLIED WARRANTIES OF AND CONDITIONS OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, AND ANY WARRANTIES THAT MAY ARISE FROM ANY COURSE OF DEALING, COURSE OF PERFORMANCE OR TRADE USAGE. SOME JURISDICTIONS MAY NOT ALLOW LIMITATIONS ON HOW LONG AN IMPLIED WARRANTY LASTS, SO THE PRECEDING LIMITATION MAY NOT APPLY TO YOU.

LECROY DOES NOT ACCEPT LIABILITY BEYOND THE REMEDIES SET FORTH IN THIS LIMITED WARRANTY STATEMENT OR FOR INCIDENTAL OR CONSEQUENTIAL DAMAGES INCLUDING, WITHOUT LIMITATION, ANY LIABILITY FOR THIRD PARTY CLAIMS AGAINST YOU FOR DAMAGES, PRODUCTS NOT BEING AVAILABLE FOR USE, OR FOR LOST DATA OR SOFTWARE. LECROY'S LIABILITY TO YOU MAY NOT EXCEED THE AMOUNT YOU PAID FOR THE PRODUCT THAT IS THE SUBJECT OF A CLAIM. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE PRECEDING EXCLUSION OR LIMITATION MAY NOT APPLY TO YOU.

The limited warranty on a Product may be transferred for the remaining term if the then current owner transfers ownership of the Product and notifies LeCroy of the transfer. You may notify LeCroy of the transfer by writing to Technical Support at LeCroy, 3385 Scott Blvd., Santa Clara, CA 95054 USA or by email at: support@catc.com. Please include the transferring owner's name and address, the name and address of the new owner, the date of transfer, and the Product serial number.

Index

A

Actions

- action buttons 69
- recording options 65
- Addr & Endp search criteria 119
- Alert, beep 49
- Analyzer, resetting 33
- Application installation 25
- Application startup 35
- Architecture of network 2
- Auto Merge trace files 49
- Automatic updates 29

B

- Basic installation 13
- Beep when trigger occurs 49
- Bit-by-Bit window 59
- Bitstream vs. Intelliframe 142
- Breakout board 21
- Buffer size 49
- Bus Conditions 60, 120
- Bus data, recording 72
- Bus Utilization 128
- BusEngine
 - technology 8
 - updating 28
- Buttons
 - action 69
 - bus utilization 128
 - navigator 100
 - real-time statistics 132
 - tool bar descriptions 40

C

- Cable Connector 22
- CATC Trace
 - recording 16

- set marker 86

Channel

- search 121
 - trigger/filter 49
 - Class request, decode 105
 - Class/vendor decoding 108, 110
 - Classic LED 20
 - Clear marker 86
 - Clock selection 55
 - Clock, Hi-Speed 5
 - Collapse. See also Expand
 - collapse data fields 88
 - collapsed transactions 94
 - collapsed transfers 98
 - Colors in Display Options 79, 80
 - Comments, editing 123
 - Complex search 122
 - Components, physical 4
 - Connecting
 - counters to events 66
 - events to actions 66
 - Context-sensitive help 35
 - Counters
 - changing values 68
 - connect to events 66
 - setting multiple conditions for 68
- ## D
- Data block, viewing 90
 - Data connectors, pin descriptions 22
 - Data fields
 - expand/collapse 88
 - pop-up menus 89
 - Data length
 - in event groups 62
 - search for 119
 - Data pattern 58
 - DB-25 connector 20
 - Decode
 - class request 105
 - decoding class/vendor 110
 - decoding options 108
 - general options for requests 103
 - hub status 112

- protocol-specific fields 99
- request recipient 108
- standard USB request 103
- vendor request 105
- Decoded transactions, viewing 93
- Decoded transfers 98
- Defining packet fields 146
- Detailed installation 19
- Device requests 58
- Direction of search 123
- Display
 - configuration name 77
 - display windows 36
 - format 16
- Display Options
 - color/format/hiding 79
 - general 76
 - level hiding 83
 - saving 83
- E**
- Editing
 - data patterns 59
 - edit comment 123
 - edit marker 86, 87
- Electrical
 - fuse 20
 - power switch 20
 - source 21
- Endpoint
 - configuration keys 147
 - decoding 110
- Errors
 - error summary 125
 - in event groups 60
 - searching for 118
 - types of 60
- Events
 - connecting to counters 66
 - connection to actions 66
 - event groups 56
 - event trigger 48
 - in Recording Options 55
- Expand. See also Collapse

- data fields 88
- expanded transactions 94
- expanded transfers 98
- Extern Data7, Data0 In 64
- External
 - external input signals 61
 - external interface connector 20
 - external interface signals, saving 49
 - external output signals 70

F

- F1 help 35
- Files
 - editing comment for trace 123
 - File Control Keys 146
 - file information 124
 - filename of trace 51
 - information 124
 - loading generation 140
 - traffic generation 135
- Filter Out/In button 69
- Filtering events 55
- Find 122
- Find Next 123
- Firmware
 - automatic updates 29
 - manual updates 32
- First CATC recording 14
- Fonts, changing 77
- Format
 - in display options 81
 - of display 16
- Frame patterns 57
- Fuse 20
- G**
- General Display Options 76
- General Recording Options 48
- Generation
 - loading files 140
 - mode 141
 - session 142
- Go to
 - Go to 116

- Go to Marker 116
- Go to Packet/Transaction/Transfer... 115
- Go to Trigger search 115
- Graphs
 - displaying multiple 133
 - Graph Areas menu 130
 - Graphical Bus Traffic Display 6
- Green LEDs 19, 20

H

- Help
 - context sensitive (F1) 35
 - Help menu 38
- Hiding
 - in display options 82
 - levels 83
 - NAKs 91
 - SOF packets 91
 - toolbars 38
- High-pulse signals 71
- Hi-Speed Slow Clock 5
- HNP 121
- Hosts
 - Host Negotiation Protocol 121
 - with a B plug 121
 - with an A plug 121

- Hot keys 44

- HS LED 20

I

- Input signals, external, external 61
- Installation

- basic 13
 - detailed 19
 - installed unit 19
- Intelliframe LED 20
- Intersection search 123

K

- Keyboard shortcuts 44
- Keys for packet fields 153, 155

L

- Layout of Actions window 65

- LEDs 19, 20
- Linking events 68
- Low pulse signals 71

M

- Manual trigger 20, 48
- Manual updates 32
- Markers
 - clearing 86
 - editing 87
- Memory requirements 49
- Menus

- graph areas 130
 - pop-up 76
 - pull-down 36
 - real-time statistics 133
 - view settings 129

- Merging trace files 73

- Misc tab in recording options 51

- Monitoring statistics 131

N

- NAKs, hiding 91
- Name of settings 50
- Navigation tools 44
- Navigator 100
- Navigator bar 100

O

- On the Go, searching 121
- Orange LEDs 19, 20
- Output signals, external 70
- Overview of software 35

P

- Packets
 - defining fields 146
 - display level 77
 - hiding SOF 91
 - packet identifiers 56
 - packet starting keys 151
 - packet view 16
 - searching for IDs 117

- Packing list 19

- Partial Upload 43

Percentages, post triggering 50

Physical components 4

Physical memory 49

Pin out descriptions 22

Pop-up menus

 data field 89

 enabling 76

Pop-up tool tips 91

Position of trigger 50

Post triggering percentages 50

Power

 power source 21

 power switch 20

Product description 1

Program

 installation 25

Program startup 14, 35

Protocol-specific fields, decode 99

Prototype rework area 23

Pull-down menus 36

Pulse toggle signals 71

R

Raw bits, viewing 87

Real-time statistics 131, 133

REC LED 19

Recording

 activity 43

 bus data 72

 progress 41

 snapshot 48

 status 42

 type 48, 49

 USB traffic 26

Recording Options

 events 55

 general 48

 options name 50

Repeat mode 141

Repeating a Generation Session 142

Report menu 37

Reports 124

Resetting the Analyzer 33

Restart button 69

Revisions, software/firmware 27

S

Save

 Display Options 83

 external interface signals 49

 recording options 71

 Save As 51

 trace files 51

Search

 complex 122

 direction 123

 feature 115

 menu 37

 status 44

Session Request Protocol 121

Set marker in trace 86

Setup menu 37

Signals

 external input 61

 external output 70

Slow clock selection 55

Snapshot recording 48

SOF packets, hiding 91

Software

 installation 14, 25

 updates 28

Specifications 9

Split transaction view 95

Splits, in event groups 64

SRP 121

Start/Stop button 20

Starting

 software 25, 35

 traffic generation 142

Statistics, monitoring 131

Status bar 41

Stop Traffic Generation 143

Summary

 error 125

 traffic 127

System

 components 19

 setup 21

T

- Timing calculations 125
- Toggle signals, pulse 71
- Token patterns 57
- Tool bar
 - button descriptions 40
 - resetting 39
- Tool tips 35, 76, 91
- Trace
 - first recording 14, 16
 - Navigator 100
 - set marker in 86
 - view features 16, 85
 - viewing level 77
- Trace files
 - auto merge 49
 - edit comment 123
 - filename 51
 - merging 73
- Trace Navigator 100
- Traffic
 - recording 26
 - summary 127
- Traffic generation
 - files 135
 - modes 142
 - starting 142
 - stopping 143
 - USBTrainer 135
- Transaction view 91
- Transactions
 - decoded 93
 - decoding protocols 99
 - expanded/collapsed 94
 - in event groups 62
 - split view 95
- Transfer view 96
- Transfers
 - decode protocols 99
 - decoded 98
 - expanded/collapsed 98
- TRG LED 19
- Triggers

- manual 20
- Manual and Event 48
- trigger button 69
- trigger position 50
- trigger/Filter channel 49
- triggering events 55
- using errors for 61

Truncate Data Fields 48

U

- Union search 123
- Universal Serial Bus Specification 3
- Updates
 - automatic 29
 - BusEngine/Firmware 28, 29
 - manual 31, 32
 - Software 28
- UPLD LED 19
- Upload, partial 43
- USB
 - decode requests 103
 - recording traffic 26
 - trace architecture 2
- USBTrainer, traffic generation 135

V

- Vendor
 - decode request 105
- View
 - data block 90
 - decoded transactions 93
 - decoded transfers 98
 - menu 37
 - options 38
 - raw bits 87
 - split transaction 95
 - transaction 91
 - transfer 96
- Voltages 21

W

- Window layout, actions 65
- Window menu 38
- Wrap 44, 76

Z

Zoom in/out 44

Zoom level, adjusting 76