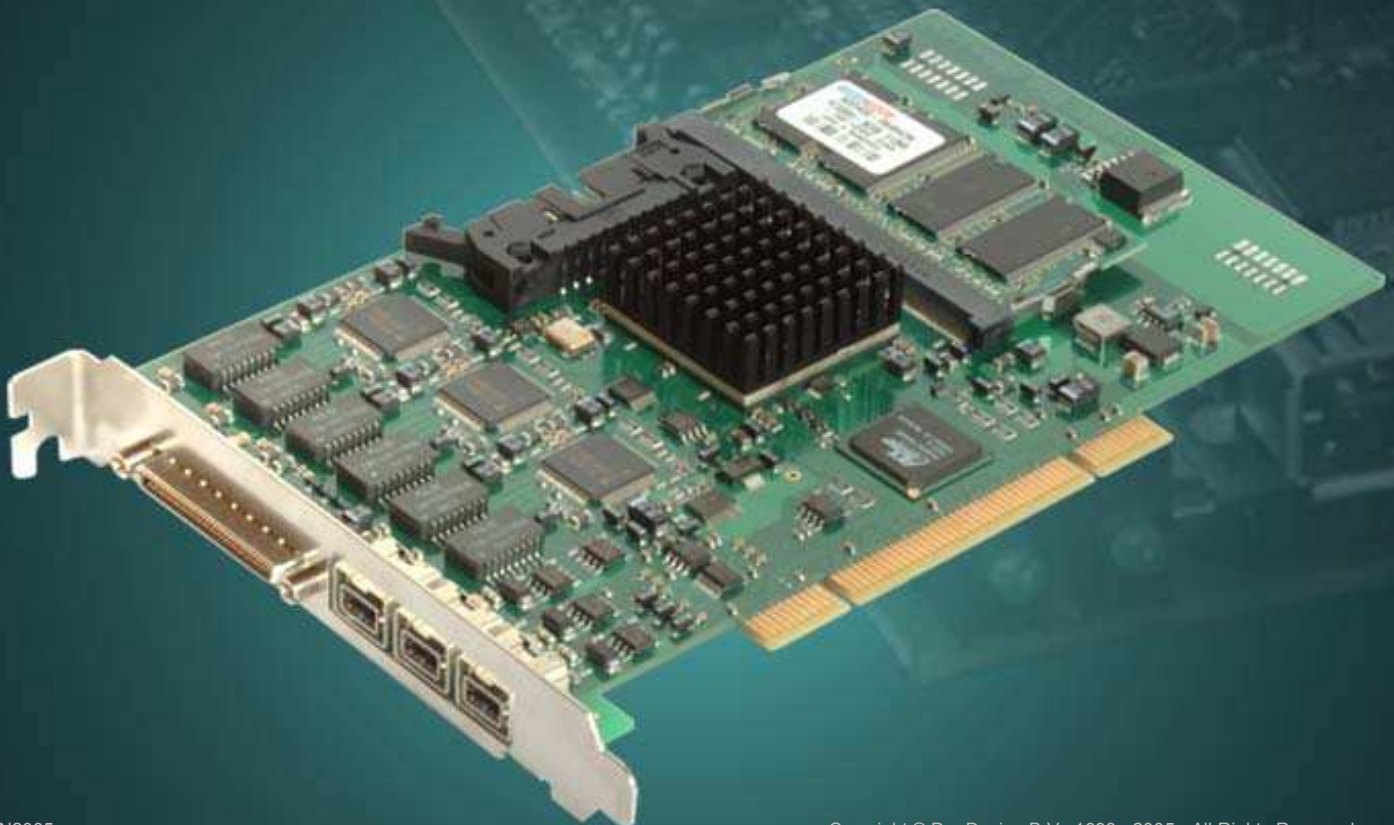


dap DESIGN[®]

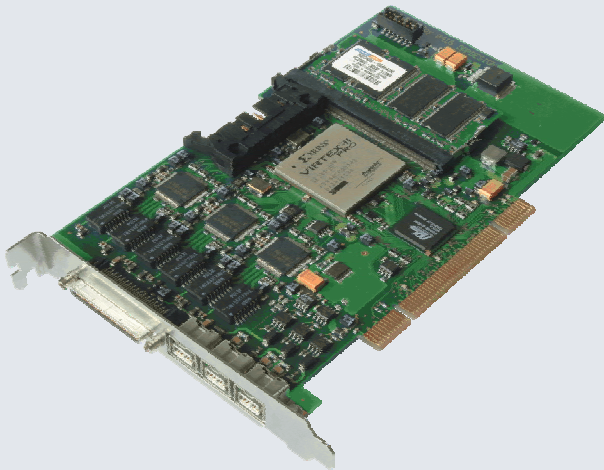
IEEE1394 Analysis

Triple Series

FireSpy **3850**



FireSpy 3850



PCI form factor

The **FireSpy3850** bus analyzer is the second in a new generation of FireWire analyzers from DapDesign. Based on the industry leading FireSpy800 and the new FireSpy810 architecture the **FireSpy3850** has raised the bar in IEEE 1394 test equipment! The **FireSpy3850** in fact combines three FireSpy analyzers in one single instrument. It comprises a significantly more powerful on-board processor and improved connectivity to the host.

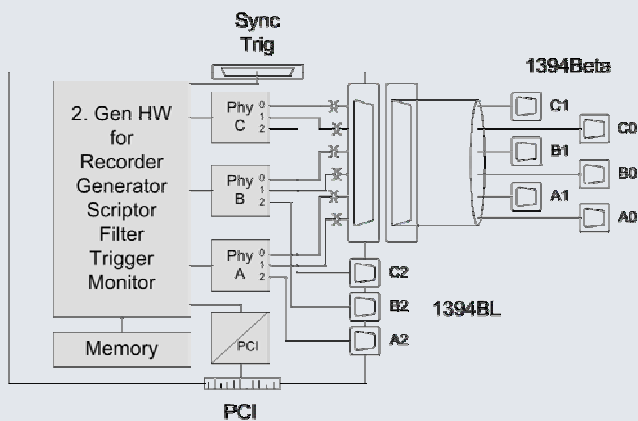
The PCI format takes flexibility to a higher level. This solution enables users to install the FireSpy hardware in any PC that has at least one PCI slot available. This allows for installation of the FireSpy in developers existing PC's or installing the analyzer in a rugged field service laptop or lunchbox computer.

The **FireSpy3850** has three 1394 nodes connected to three synchronized analyzer engines. They are controlled by a RISC processor running at 400 MHz. Each node is connected to three FireWire ports. One port of each node is connected to a separate standard bilingual FireWire connector. The other two ports of each node are connected to one high speed connector and include active transformer coupling.

The **FireSpy3850** is equipped with 512 MB memory and extensive hardware filter and trigger possibilities. On the host you can control the FireSpy using a graphical user interface to analyze and display the bus traffic in a user-friendly way, or you can use the API and program your own controlling software.

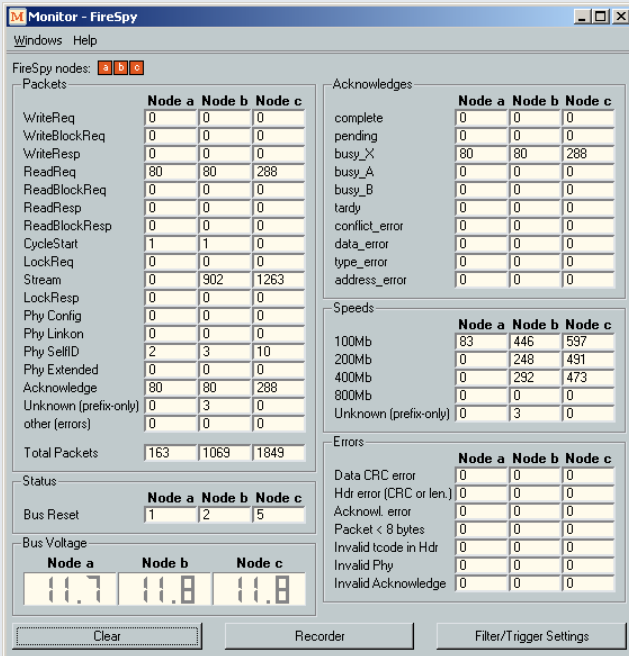
Key Features

- IEEE 1394-1995, 1394a-2000 and 1394b-2002 compliant
- Supports 100, 200, 400, 400b and 800Mb data transfer rates
- PCI 2.1 compliant
- On-board 400 MHz RISC processor and programmable logic
- 512 MByte internal memory
- Supports Windows NT, 98, 2K and XP
- 3 analyzing nodes with a total of 9 IEEE1394 ports.
 - 6 of the ports are active transformer coupled
 - 3 of the ports are standard FireWire bilingual connectors
- Enhanced Scriptor functionality for real-time monitoring and asynchronous packet generation
- Powerful software provides:
 - Monitor
 - Recorder
 - Commander
 - Scriptor (programmable in C-like language)
 - Generator
 - Filter and Trigger
 - Support for VS (Mil1394), AV/C, SBP2, IP1394 and IIDC protocols
- Internal SelfTest
- LabView™ interface via Scriptor or API (to be released)



FireSpy3850 block diagram

Monitor



The **Monitor** gives a quick indication of bus activity. This powerful tool displays the number of packets passed on the bus and some additional events like Bus Resets. The packets are differentiated by speed and type, including all isochronous, asynchronous, Phy packets and acknowledge packets types. Also the number of packets with different error types and the IEEE1394 bus voltage are monitored and displayed.

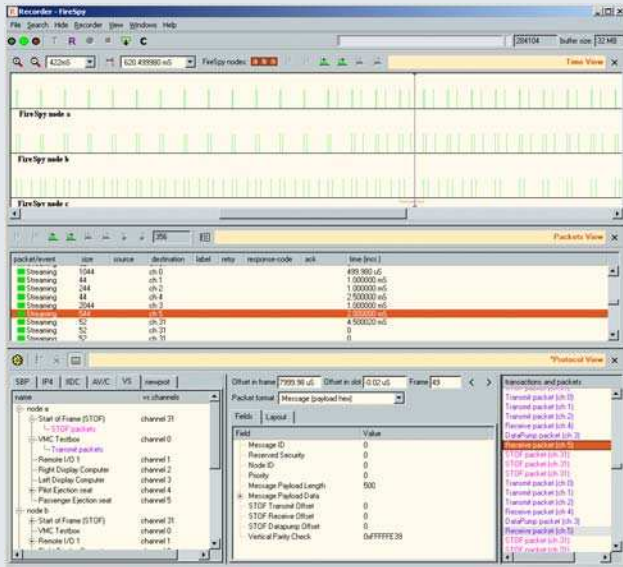
The following information for each connected network is shown by the monitor and updated in real time:

- Number of packets of specific types
- Number of packets using a specific speed
- Number of acknowledge packets of specific types
- Number of error packets
- Total number of packets
- Number of bus resets

Bus voltage measurement

•

Recorder



3-channel VS-protocol recording and display



Dataflow analysis of ConfigROM access in Transaction viewer

Comparable with a logic analyzer, events and packets can be stored into FireSpy's internal memory and viewed later on the computer to which the FireSpy is connected.

The **Recorder** contains the following views, which can all be switched on or off individually.

Time View

The *Time View* enables the user to view the timing of events and packets (including acknowledge packets and the packet prefix) at a resolution of 10ns. A time cursor is used to make time measurements.

Packet View

The *Packet View* displays all packets in a list. Each packet can be viewed in detail, by showing its packet fields, packet layout or packet errors. Packets can be displayed as any possible packet type, so the user can find out what kind of packet an erroneous packet may be.

Transaction View

The *Transaction View* displays a list of all complete and incomplete transactions. The details of each transaction can be displayed as a packet list or as a graphical flow diagram.

Topology View

The *Topology View* shows the bus topology at the cursor position in a graphical representation. Note that the topology may change during recording because of node connection or disconnection. Optionally, details of connected nodes may be displayed. These details include the configuration ROM contents, read during recording.

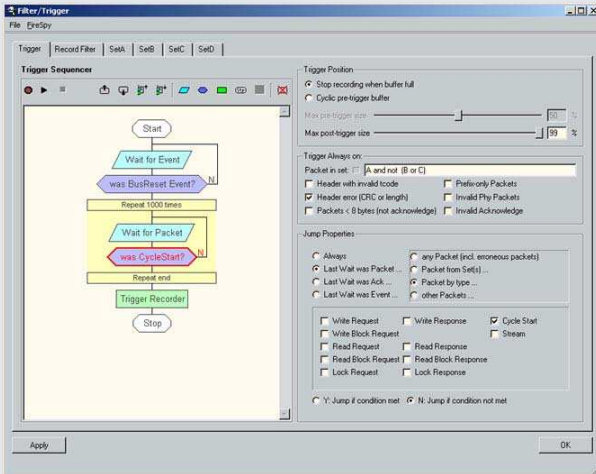
Protocol View

The data transactions are shown in a tree structure showing the result of the protocol analyzers supported by the FireSpy. The contents of the data transactions are shown in a format corresponding to the selected protocol format. Currently the following protocols are supported:

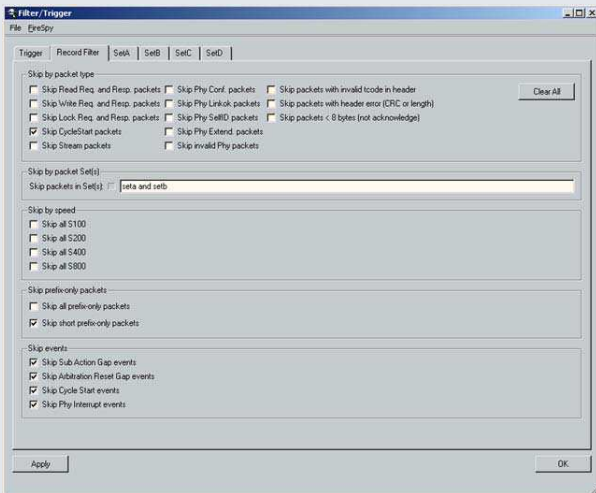
- VS (Mil1394)
- SBP2
- AV/C
- IP1394
- IICD

Support for protocols is optional and needs to be ordered separately. For further and more detailed information on all supported protocols please see the separate "FireSpy Protocol Options" datasheet.

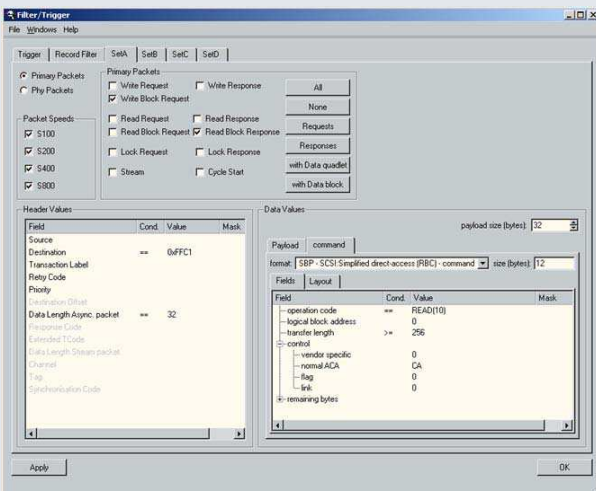
Filter/Trigger Settings



Trigger Sequencer



Comprehensive Filter Settings



Packet Set Definition for Filter/Trigger/Search Functions

The **FireSpy3850** has powerful hardware trigger and filter logic to control which data will be stored in memory by the Recorder, and to be able to create a reference point in the Recorded data called the trigger point.

Trigger

The Recorder can be triggered using extensive hardware trigger logic. You can select simple trigger conditions such as trigger on a particular type of error packet or trigger when a packet fits the description of a Packet Set or any Boolean combination of the four Packet Sets.

For more advanced trigger requirements a powerful Trigger Sequencer can be used. Designed to work like a flowchart this sequencer makes definition of the most complex multi-level trigger sequences a simple and straight forward task.

For optimized buffer resource management the user can manually define the maximum amount of data to be recorded after a trigger. And one can choose to use a cyclic buffer for the data stored before the trigger point. The size of this cyclic buffer can be changed for efficient storage management.

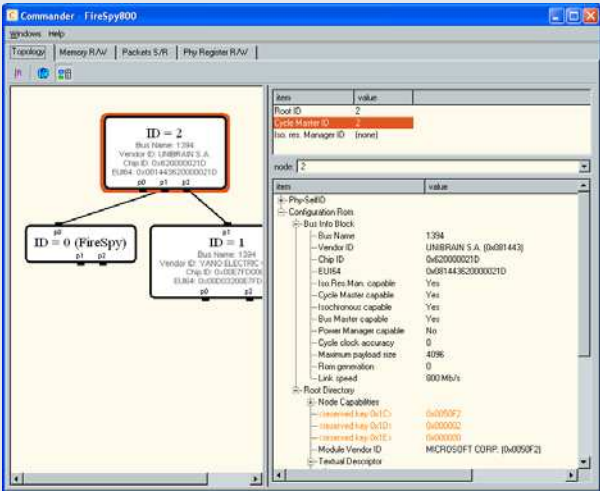
Filter

With the Recorder Filter you can skip specific packets or events. Skipped packets or events are not stored in memory by the Recorder. The packets to be skipped can be selected by type, speed, error type or by using any Boolean combination of the four Packet Sets.

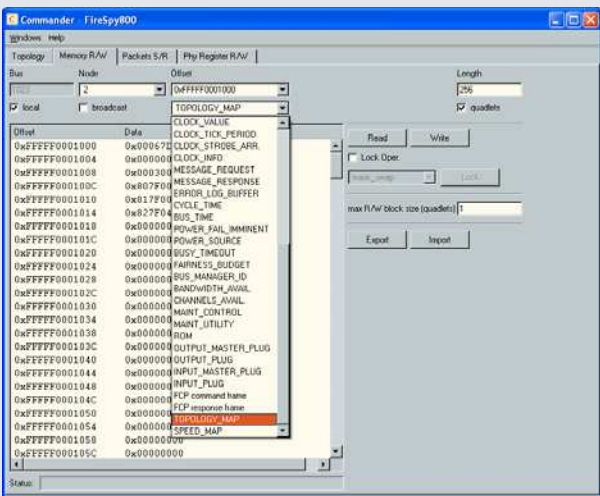
Packet Sets

Both Filter and Trigger logic can use any Boolean combination of four Packet Sets. These Sets consist of any combination of primary (asynchronous or stream) packets or a Phy packet, with optional value checking on packet fields. Value checking can be performed on header fields or data fields. In case of data fields, a powerful packet editor is included to be able to put conditions on data fields of a large variety of packets.

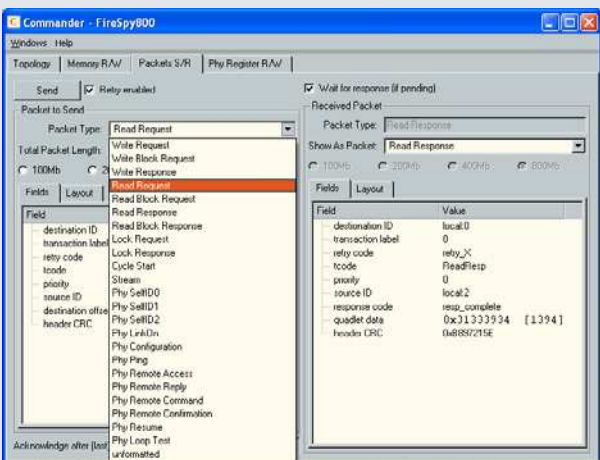
Commander



Bus Topology and ConfigROM decoder



Reads / Writes of/to predefined CSR Registers



Send / Receive to/off BusInfoBlock location

The **Commander** can be used to control several aspects of a connected FireSpy. Its functionality is split-up in the following four parts:

Topology

The *Topology* is used to display the current bus topology. When a node is connected or disconnected, the topology changes are displayed immediately. Optionally, more detailed node information can be displayed, including the configuration ROM contents. One can manually generate bus resets using a push button in the toolbar.

Memory Read/Write

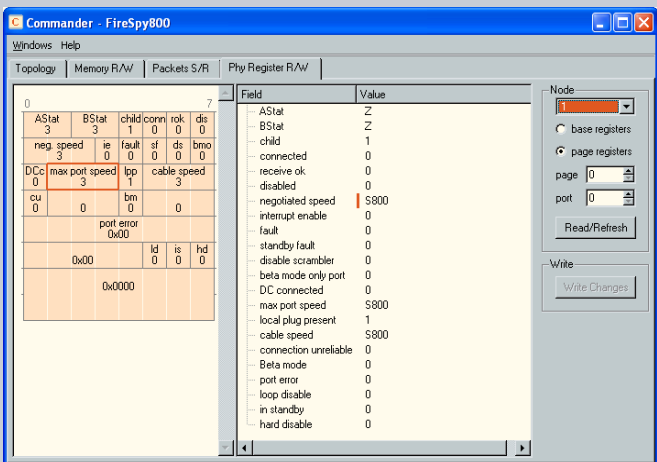
The *Memory RW* function is used to read and or write the memory of remote nodes. Read, Write and Lock operations can be performed on a selected set of predefined addresses (like the CSR registers) or on user definable addresses.

Packet Send/Receive

With the *Packet S/R* function, one can send all kinds of packets including Phy packets, unformatted packets and erroneous packets to a selected node. Optionally one can wait for and display the response. Both the packet sent and the packet received can be displayed as layout or field list.

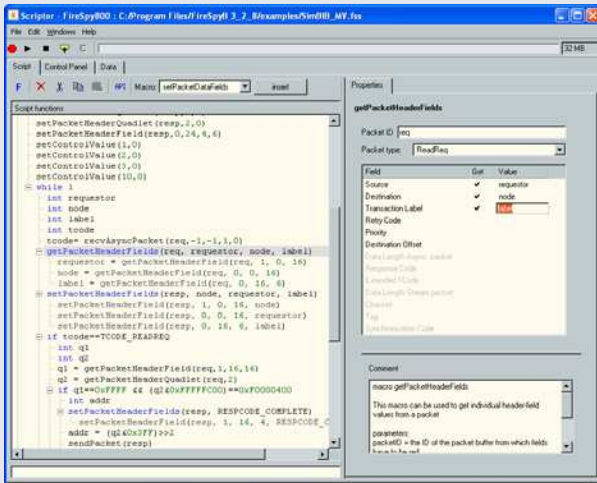
Phy Register Read/Write

With the *Phy Register RW* function, one can read and write the registers of the local and read the registers of a remote Phy-interface. The register layout and a list of all register fields are shown. The registers can easily be altered by editing the register fields and issue the "Write Changes" command.

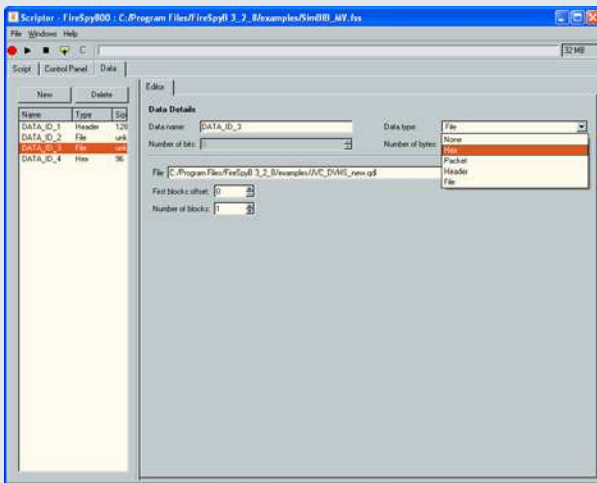


Phy Register Map

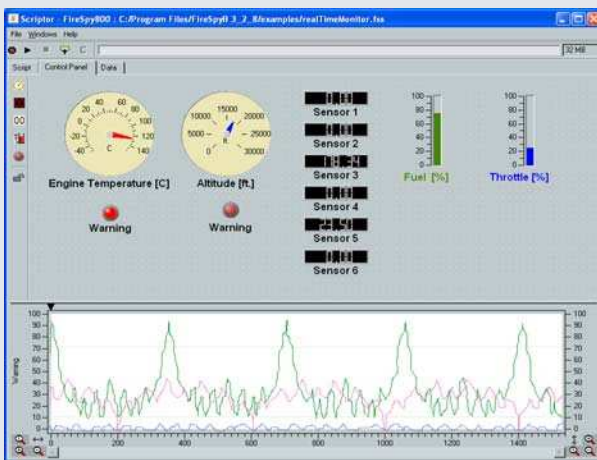
Scriptor



C-like Scripting Language in the Script Editor



Data Editor



Customizable Control Panel Display

The **Scriptor** can be used to define scripts and run them on a FireSpy device. The Scriptor uses a C-like language to control almost anything of the FireSpy, including sending and receiving of packets. The Scriptor is the preferred tool for the generation of individual asynchronous packets, asynchronous sequences and the simulation of entire handshake sequences between the FireSpy and the device under test.

Script editor

The Script tab is used to enter a C-like script. The powerful editor includes features like:

- Automatic indent
- Syntax coloring
- API function list
- Macros to automatically generate blocks of code
- Integrated Debugger (to be released)

Data editor

The Data tab is used to define data that can be used by the script. Several data editors are available to the user:

- Hex editor
- Packet Header editor
- Packet Data editor
- File input selector

Control Panel

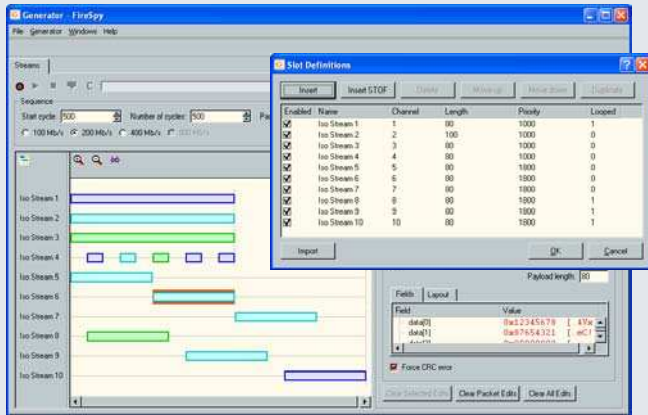
Values can be sent to a customizable Control Panel displayed on the host computer. The Control Panel displays values that it receives by using different kinds of meters (gauge, LCD, thermometer, etc).

It is possible to send control values (e.g. values of a certain data field with the packet payload section) to a history graph to display the change of values over time. Upper and lower bounds can be defined for each control and an alarm (visual and audible) is triggered when a value is out of bounds.

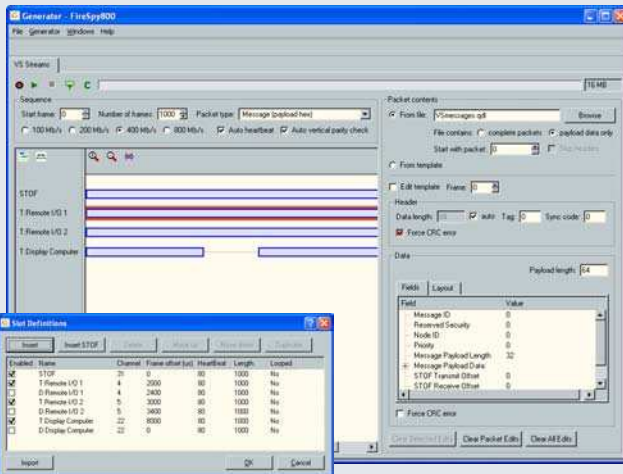
The Control Panel also allows users to manually define input fields. This feature provides an easy way to, for example, define payload field values for to be transmitted packets during Scriptor execution or to activate any simple or complex transaction with a click on a button. Provided control functions include buttons, sliders and numeric edit boxes.

A LabView™ interface is provided which allows sending/receiving all control values to/from LabView™ in real time.

Generator



Stream Generator



VS Stream Generator

The **Generator** has been optimized for the generation of isochronous and asynchronous stream data packets. It offers the most comprehensive feature set for the insertion of errors, streaming of simultaneous channels and payload definitions from stored files.

Stream Generator

The *Stream Generator* includes a powerful graphical editor to specify slots with stream sequences to be sent for up to 31 channels.

Each sequence consists of one or more stream packets with selectable data sources that can be fixed or from file. For each sequence one can select various options such as speed, packet size and header fields, including erroneous values. The overall sequence size is customizable in multiples of Cycle Periods. All Generator slots can be run in a looped-mode continuous transmission.

Both the *Stream Generator* and the *Scriptor* can run in parallel for advanced isochronous and asynchronous combination testing.

VS Stream Generator (optional)

A special version of the Generator is included in the optional VS (Mil1394) protocol package. Specific details of this protocol like the 12.5 ms time base for STOF packet generation have been implemented in this specialized version of the Generator:

- STOF packet generation including timing adjustments
- Frame Length adjustment (preset to 12.5 ms, adjustable from 10 – 20 ms with 1 μ s resolution)
- Time Slot definition for asynchronous streams generation with 1 μ s resolution
- Definitions for receive and transmit offsets
- Insertion of transmission errors
- Automatic Vertical Parity Check calculation option
- Automatic Heartbeat calculation option

Note that packets may also be inserted into the frame at a 1 μ s resolution using the *Scriptor*.

Feature Summary

GENERAL

- IEEE1394-1995, 1394a-2000 and 1394b-2002 compliant
- Supports 100, 200, 400 and 800 Mbps transfer rates
- Half Length PCI Card (106mm x 174mm)
- PCI 2.1 compliant
- 504 MByte memory for Recorder, Generator and Scriptor
- Self Powered, 0W or 2.8 W bus Power Provider per bus
- Firmware field upgradeable to enable future expansions
- Software runs on Windows98, 2000, NT, XP
- Combines five functions which can operate simultaneously:
 - Monitor
 - Recorder
 - Commander
 - Generator
 - Scriptor

MONITOR

- Displays bus activity by counting all kinds of packet types transmitted on the bus, including:
 - isochronous packets
 - all types of asynchronous packets
 - all types of Phy packets
 - all types of acknowledge packets
 - errors like:
 - header & data CRC error
 - acknowledge error
 - unknown packets
 - unknown Phy packet
 - unknown acknowledge
 - packet size to small
- Counts packets according to transmission speed
- Counts number of bus resets
- Measurement of bus power voltages

RECORDER

- Up to 504 MByte storage for recorded packets and events (shared with Generator and Scriptor)
- Time stamping of all packets and status events with 10ns resolution
- Packets hidden by slower connections are visible as 'prefix only' packets
- Extensive hardware packet/event filtering for selective storage
 - Packet type
 - Transmission speed
 - Boolean combination of 4 programmable packet sets
 - Error conditions
 - Various status events
- Extensive hardware trigger selection:
 - Packet type
 - Transmission speed
 - Boolean combination of 4 programmable packet sets
 - Error conditions
 - Various status events
 - Graphical Trigger Sequencer
 - Flowchart design for easy setup
 - Conditional jumps
 - Repeat loops
 - Trigger on errors
- Extensive Search capability:
 - Packet type
 - Boolean combination of 4 programmable packet sets
 - Error conditions
 - Various status events

- 4 programmable Packet Sets for Filter and/or Trigger include:
 - Packet type (tcode) or Phy layer packet type
 - Transmission Speed
 - Individual Field values within packet header
 - matching conditions (==, !=, >=, <=)
 - bit level masking
 - unformatted payload data patterns (quadlet based)
 - matching conditions (==, !=, >=, <=)
 - bit level masking
 - Protocol based payload definitions (field based)
 - matching conditions (==, !=, >=, <=)
 - bit level masking
- Adjustable trigger position within programmable record buffer size
- Cyclic pre-trigger buffer management option
- Different kinds of packet display views, including:
 - Time View, displays all packets on a time line, including the prefix
 - Packet View, displays packets as list plus selected packet options:
 - packet fields
 - packet layout
 - packet errors
 - Transaction View, displays transactions as list or flow graph
 - Topology View, graphically displays topology as it existed during the recording
- Relative cursor mode for easy time measurements
- Marking of individual packets or packet ranges

COMMANDER

- Reading and/or writing of local and reading of remote Phy registers
- Reading and/or writing of all local and remote memory locations (including CSR register space)
- Possibility to graphically view the current Topology
- Sending of user definable packets including:
 - isochronous packets
 - asynchronous packets
 - Phy packets
 - packets with errors, including
 - data CRC error
 - header CRC error
 - unknown packet
 - illegal packet length
 - any sequence of bytes
 - optionally automatic retransmit
 - optionally receive response

GENERATOR

- Simultaneous generation of up to 31 isochronous/asynchronous streams
 - Up to 504 MByte memory for Generator packets (shared with Recorder and Scriptor))
 - Graphically programming of stream transmit blocks, defining:
 - channel number
 - start and stop cycle time
 - send priority
 - speed
 - special header values (including forced errors)
 - data source, including
 - fixed data (using packet editor)
 - data from file
- Both Generator and Scriptor can run simultaneous for stream and asynchronous packet generation
- Special VS stream generator package (optional)

Specification

SCRIPOR

- Up to 504 MByte for Scriptor execution (shared with Recorder and Generator)
- Script Editor
 - C-like scripting language
 - API function list
 - Macros to automatically generate blocks of code
 - Syntax coloring
 - Integrated Debugger (to be released)
- Data Editor
 - Data Input used by Scriptor
 - Hex Editor
 - Packet Header editor
 - Packet Data editor
 - File input selector
 - Manual overwrite of individual payload fields
- Control Panel
 - Macro driven setup
 - diverse displays for data value presentation
 - LCDs
 - Gauges
 - Alarms
 - History graph
 - Manual data input capability
- LabView™ Interface
- Several Sample Scripts

FORMAT EDITOR

- Separate application for modification and extension of the following encapsulated protocol sets:
 - VS(Mil1394)
 - AV/C
 - SBP2

Dimensions:	Half length PCI card, 15mm x 106mm x 174mm
Weight:	150 g
Power Requirements:	1 A maximum
Regulatory compliance:	FCC Class A
Connections:	32bit/33MHz PCI connector, universal keyed (for 3.3V and 5V slots) 3 IEEE1394-connectors (bilingual) high-density connector (SCSI) for 6 transformer coupled 1394Beta connections
Indicators:	-
Switches:	-
Package Content:	FireSpy3850 1394b Cable (bilingual - bilingual) 1394b/1394a Cable (bilingual – 6pin) 1394 Adapter Cable (SCSI – 1394Beta (x6)) “FireWire System Architecture” Book, Mindshare
Product warranty:	6 months limited warranty
Part Number:	FS385 (FireSpy3850) HSS385 (Extended Warranty for 12 months) FS385P6HSS (FS3850 plus VS(Mil1394) protocol software package and 18 months warranty)
Optional Configuration:	-
SW Add-on modules:	SBP2 protocol software package IIDC protocol software package AV/C protocol software package IP4 protocol software package VS(Mil1394) protocol software package

Contact Information:

DapDesign The Netherlands
sales@dapdesign.com
 Ph: +31 541 532941
 Fax: +31 541 530193
www.dapdesign.com

Dap USA
USsales@dapdesign.com
 Ph: (480) 706 7027
 Fax: (480) 706 6398
www.dapdesign.com