



## Bluetooth Sniffer v. 3.0 + HS

### PC based Bluetooth Protocol Analyzer FTS4BT

- ◆ FTS for Bluetooth decodes all Bluetooth protocols and supports most Bluetooth profiles.
- ◆ Captures, decodes, displays, and filters in real time.
- ◆ Data is decoded at the frame, byte, and bit levels, which permits users to rapidly detect and isolate even the most minute and intermittent protocol-related problems.
- ◆ Continuous direct logging to disk with unlimited file size for data and speech (WAV file).
- ◆ Sniffs in the air and serial HCI simultaneously, with synchronized time stamping.
- ◆ Decodes and displays multiple protocol layers of multiple data frames simultaneously.
- ◆ Detects and displays protocol errors (in red) in real time.
- ◆ Supports data decryption and Simple Pairing with ECDH cryptography per BT version 2.1
- ◆ Channel chart with BER measurement and Throughput in Bits/sec over time.
- ◆ The Wi-Fi Option pin-points coexistence issues and payload efficiency.
- ◆ Bluetooth Low Energy Analyser also available

The Bluetooth Sniffer FTS4BT is complementary to the R&S Bluetooth testers:




CDMA2000 WCDMA Bluetooth  
GSM  
1xEV-DO  
Analog  
TDMA

Version: 00.00  
July 2007

**R&S® CMU200 Universal Radio Communication Tester**  
THE multiprotocol tester for current and future mobile radio networks

The image shows a rack-mounted R&S CMU200 Universal Radio Communication Tester. The device has a large color display on the front panel showing a signal waveform. To the right of the device, a list of supported protocols is displayed: CDMA2000, WCDMA, Bluetooth, GSM, 1xEV-DO, Analog, and TDMA. A blue banner at the bottom right indicates the version (00.00) and date (July 2007). Below the device image, the product name and a descriptive tagline are provided.



Version: 01.00  
November 2003

**WLAN Protocol Tester R&S® PTW70**  
IEEE 802.11 multimode protocol tester for development, integration and verification

The image shows the R&S PTW70 WLAN Protocol Tester, a rack-mounted device, connected to a laptop. The laptop screen displays a software interface with various data fields and graphs. A blue banner at the bottom right indicates the version (01.00) and date (November 2003). Below the image, the product name and a descriptive tagline are provided.



Version: 02.00  
July 2007

**R&S® CBT/CBT32 Bluetooth® Testers**  
Fast and comprehensive RF and audio measurements for development, production, and verification

The image shows the R&S CBT/CBT32 Bluetooth Testers, a rack-mounted device with a large display showing a Bluetooth test interface. A blue banner at the bottom right indicates the version (02.00) and date (July 2007). Below the image, the product name and a descriptive tagline are provided.

## FTS4BT (Frontline Test System for Bluetooth)

The USB dongle was developed in conjunction with CSR (Cambridge Silicon Radio) and was the first protocol analyzer for Bluetooth® version 2.1, complementing the R&S Products like R&S PTW60/70 and R&S CMU200.

In addition to the measurements of the physical RF parameters by R&S CMU200 or R&S CBT and the programming capability of protocols by the R&S PTW, Frontline is analysing the application protocols and interoperability of Bluetooth® devices.

The Frontline Bluetooth® protocol analyzer, even though it is also decoding the Baseband, is orientated to the software development of the higher layers. Baseband is of minor importance there, all efforts go to the software application around this chip and that's the field of main application and strength of Frontline's Bluetooth® protocol analyzer.

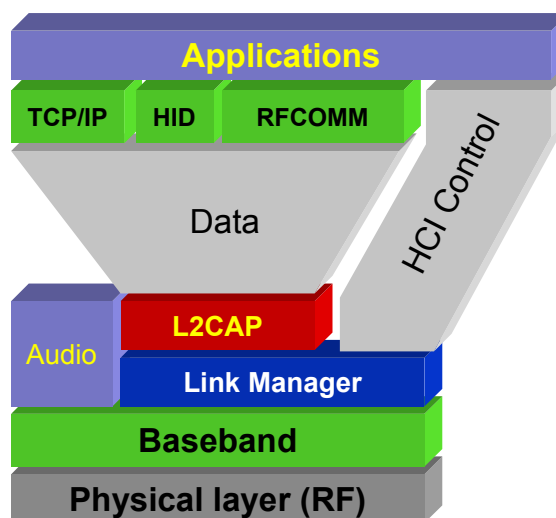
For application debugging, FTS4BT is used as a tool to figure out what is going on with the messaging in real time.

FTS4BT is being used also for verification. In difference to the R&S „Qualification and Compliance“ test, „verification“ is more a functional test of a Bluetooth® device.

Bluetooth® SIG (Special Interest Group) designates specialized organisations as a BQB (Bluetooth® Qualification Body). In order to qualify a Bluetooth® product (this could be a chip, mobile phone, PDA etc.) the BQB will verify the received documentation (they call this evidence) that shows that the product functions and conforms to the Bluetooth® spec.

FTS4BT is used to produce this documentation by capturing the traffic and then prepare the evidence for the BQB.

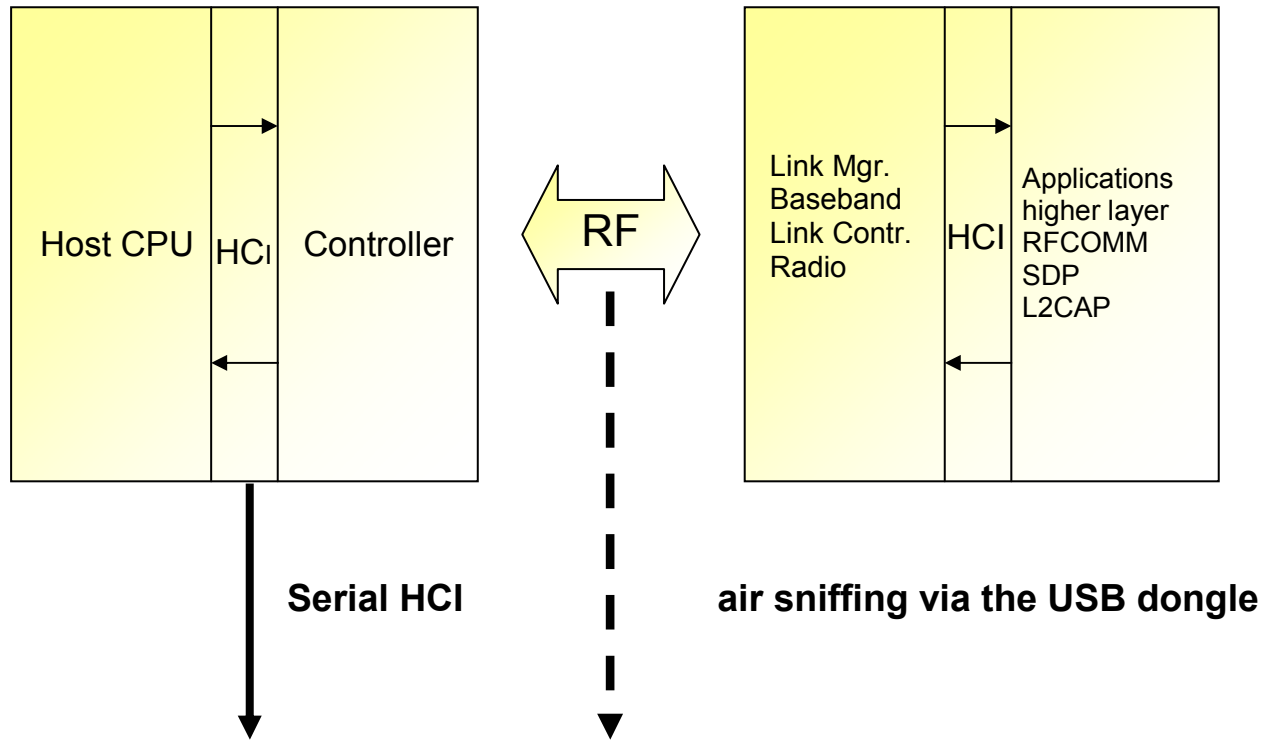
**FTS4BT's architecture** enables quick release of new decoders as protocols are announced by the Bluetooth SIG. The user can rapidly develop and seamlessly integrate Vendor Extension decoders using the built-in FrameDecoder™ feature.



A standard Bluetooth® device consists of a Bluetooth® controller and a CPU with the Bluetooth® application software. These two elements are connected by the HCI (host controller interface). The controller with the LMP (layer3 link manager protocol) incorporates the Baseband and Physical layer (Bluetooth®-radio). Many Bluetooth® devices give physical access to the HCI (three wires) e. g. via a serial interface.

**Bluetooth device with  
access to serial  
Host controller Interface HCI**

**Embedded  
Bluetooth Device**



**FTS4BT protocol analyzer software  
decoding simultaneously in real time**

FTS4BT can be connected to the **HCI** of the device via a RS232 port, via USB or virtually to analyze commands and events between these Bluetooth® elements. At this point it can be checked if the application profiles and the serial emulation to the L2CAP (logical link control and adaption protocol) inside of one Bluetooth® device is correct.

In an active Piconet, where at least two Bluetooth® devices (one master, one or more slaves) interact with each other, the USB dongle is **air sniffing** the communication between those. This analysis is required to check interoperability of Bluetooth® devices from different vendors and to troubleshoot problems by detailed protocol decoding.

## Pairing

Bluetooth devices on an encrypted link share a common link key in order to exchange encrypted data. This link key is derived from a shared PIN code, the master's Bluetooth slot clock, the master's BD\_ADDR and a random number that is passed between the two devices. The pairing process is the sequence of events used to create this key, as shown in the Link Manager Protocol below.

Frame#	AM_Addr	Original Opcode	Opcode	Role	Initiated by	Frame Size	Delta	Timestamp
1	1		name_req	Master	master	17		30.08.2007 09:09:51.533070
2	1		name_res	Slave	master	32	00:00:00...	30.08.2007 09:09:51.559946
3	1		detach	Master	master	17	00:00:00...	30.08.2007 09:09:51.564321
4	2		version_req	Master	slave	21	00:00:00...	30.08.2007 09:10:00.235519
5	2	host_connection_r_	accepted	Slave	master	11	00:00:00...	30.08.2007 09:10:00.262307
6	2		setup_complete	Master	master	16	00:00:00...	30.08.2007 09:10:00.265511
7	2		in_rand	Slave	slave	26	00:00:00...	30.08.2007 09:10:00.467395
8	2	hci_rand	accepted	Master	slave	17	00:00:04...	30.08.2007 09:10:04.478189
9	2		comb_key	Slave	slave	25	00:00:00...	30.08.2007 09:10:04.531215
10	2		comb_key	Master	slave	32	00:00:00...	30.08.2007 09:10:04.535681
11	2		au_rand	Slave	slave	26	00:00:00...	30.08.2007 09:10:04.583619
12	2		sres	Master	slave	29	00:00:00...	30.08.2007 09:10:04.633195
13	2		au_rand	Master	slave	32	00:00:00...	30.08.2007 09:10:04.683197
14	2		sres	Slave	slave	14	00:00:00...	30.08.2007 09:10:04.710074
15	2		encrypt_mode_req	Slave	slave	11	00:00:00...	30.08.2007 09:10:04.711324
16	2	encrypt_mode_req	accepted	Master	slave	17	00:00:00...	30.08.2007 09:10:04.763201
17	2	encrypt_key_size_req	accepted	Master	slave	17	00:00:00...	30.08.2007 09:10:04.768951
18	2	encrypt_key_size	accepted	Slave	slave	11	00:00:00...	30.08.2007 09:10:04.793827
19	2	start_encrypt_req	accepted	Master	slave	32	00:00:00...	30.08.2007 09:10:04.844654
20	2	start_encrypt_req	accepted	Slave	slave	11	00:00:00...	30.08.2007 09:10:04.896332
21	2		setup_complete	Slave	slave	10	00:00:00...	30.08.2007 09:10:04.897583

Frame 7 is the LMP\_in\_rand which is where a random number generated by the master is passed to the slave. The slave acknowledges that it has accepted the number in frame 8. In frames 9 and 10, the combination keys are passed between master and slave. In frames 11 and 13 the LMP\_au\_rand is being exchanged. This is the random number that has been encrypted by using the calculated link key. The response LMP\_sres confirms that the same number was computed. When the encryption mode request is accepted, encrypted data can start to be sent.

## Simple Pairing in BT version 2.1

The goal of simple pairing is to simplify the pairing process for the user and at the same time ensure the security in Bluetooth wireless technology. The maximum security in version 2.0 + EDR or earlier is achieved when applying the 16 character alphanumeric PIN. The difficulty for an attacker to calculate the secret link key is the same for simple pairing but easier for the user as the protection with simple pairing is independent of the length of the PIN code.

Simple pairing uses Elliptic Curve Diffie-Hellman (ECDH) public key cryptography. ECDH provides a very high degree of strength against passive eavesdropping attacks. Active eavesdropping, also known as man-in-the-middle (MITM) attack, is protected with a 1 in 1 000 000 chance with simple pairing.

## Decryption process

### Version 2.0

FTS4BT must compute the same link key being used by the devices being sniffed. Since the link key is never sent over the air, FTS4BT must know the PIN code and capture the LMP\_in\_rand, both LMP\_comb\_keys and both LMP\_au\_rand/LMP\_sres pairs. After FTS4BT has calculated the link key, it is capable to decrypt data successfully.

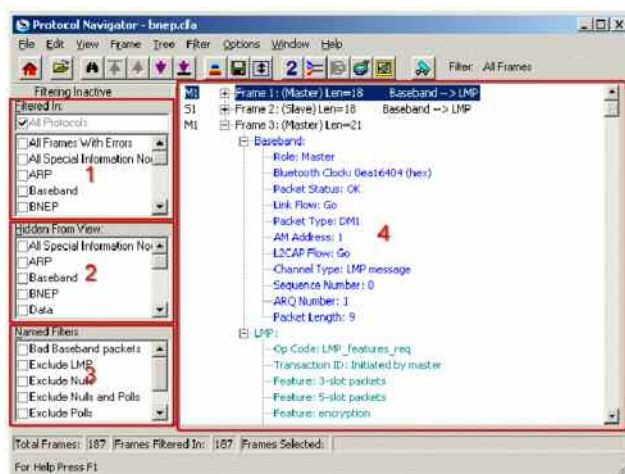
### Version 2.1

Secure simple pairing messages are captured but FTS4BT cannot compute the ECDH link key from the data in the messages. This means that any encrypted data will not be automatically decrypted.

Bluetooth SIG recommends to the chip producer to implement a debug mode, which can be activated just for error checking. This is unsecure but allows FTS4BT and other equipment to decode the protocols.

Alternatively, FTS4BT can decrypt messages if the link key is provided by the user during the capture process.

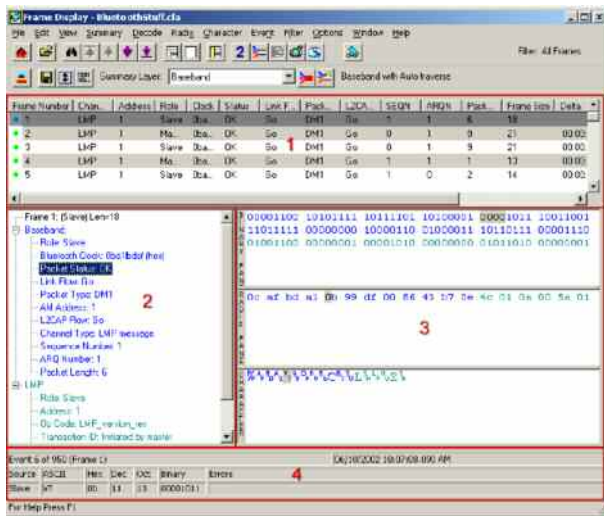
## Multiple Protocol Layers within multiple data frames



**Protocol Navigator Window:** presents decodes of multiple protocol layers within multiple data frames. Simplifies the process of understanding the complex relationships between multiple data frames and the protocol layers that comprise the frames.

1. Protocol Selection Pane: enables you to view only data frames that contain selected Bluetooth protocols.
2. Protocol Suppression Pane: enables you to suppress selected protocols within the Bluetooth data frames.
3. Frame Filter Pane: enables you to select pre-defined filter conditions that include or exclude specific data frame types.
4. Protocol Decode Pane: displays one or more decoded data frames, with text-based explanations of the various protocol elements.

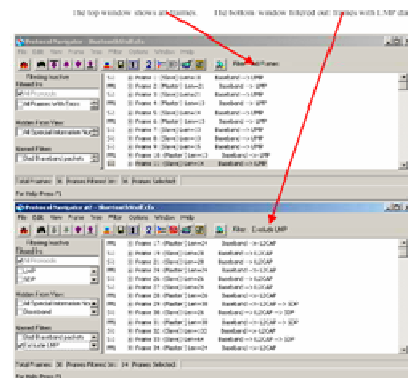
## User Data, from the frame level to the bit level



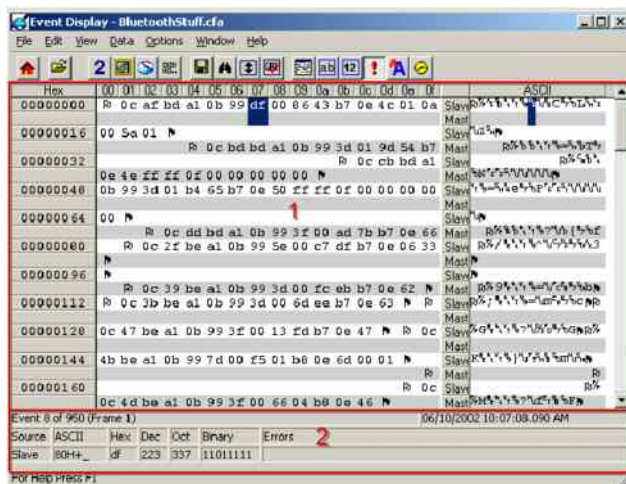
**Frame Display Window:** presents various views of user data, from the frame level to the bit level, including a detailed decode of a user-selected data frame using a layer-by-layer tree structure. The protocol decode presents text-based descriptions of the protocol-related information resident in each data frame.

1. Summary Pane: displays a high level overview of each Bluetooth data frame, including all significant fields associated with the selected protocol layer.
2. Frame Display Decode Pane: displays a single decoded data frame at levels of detail ranging from the frame level to the bit level.
3. Data Panes: display Bluetooth data in various formats, including binary, hexadecimal, and characters.
4. Selected Data Detail Section: provides protocol-related information pertaining to user-selected portions of the data frame.

**Window Duplication:** the Duplicate Button creates a second window that is identical to the first. Duplication enables you to view two segments of the data stream simultaneously, or to compare identical data streams with different filter conditions applied.



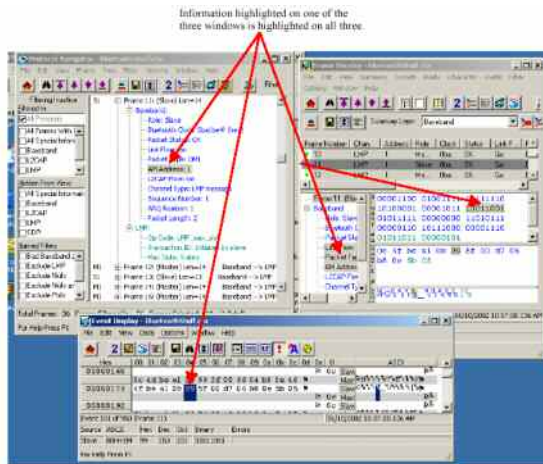
## Data at the byte level



**Event Display Window:** used to monitor and analyze data at the byte level, and to conduct searches for data patterns.

1. Byte Display Panes: display data flows among Bluetooth devices, in various formats, at the byte level.
2. Selected Data Detail Section: provides protocol-related information pertaining to user-selected, multi-frame portions of the data stream.

## Window Synchronization



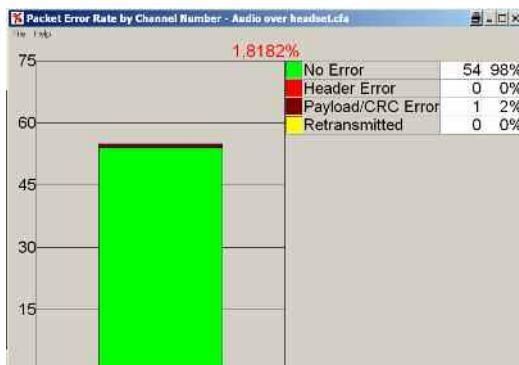
**Window Synchronization:** the Protocol Navigator, Frame display, and Event Display Windows are synchronized. Selecting information in any of these windows highlights the same information in the other two windows

## Packet Error Rate Analysis

Graphically viewing the packet error rate on all 79 channels at the same time provides immediate insight into baseband performance.



Clicking on a channel of interest opens a detailed view with the number of packets having used this channel with the percentage of header error, payload CRC error and retransmissions.





## Technical Description

### Data Capture Modes

Sniffs data in three modes simultaneously, with synchronized timestamping:

- ♦ Through the air via an attached device called the Bluetooth ComProbe, which connects to the USB port on the host PC.
- ♦ From the serial HCI interface between a Bluetooth Host CPU and a Bluetooth Host Controller (Bluetooth Device).
- ♦ By "virtual sniffing" via the product's Live Import feature, which permits any application to feed data into FTS for Bluetooth via Microsoft's COM interface for interprocess communication.

### Piconet Synchronization Methods

- ♦ Master Inquiry.
- ♦ Slave Inquiry.
- ♦ Passive Slave Page.

### Data Capture

- ♦ Captures, decodes, displays, and filters in realtime.
- ♦ Supports full piconet data capture.
- ♦ Captures, decrypts, and analyzes encrypted data.
- ♦ Supports Pairing, Master/Slave Switching, Park, Hold, and Sniff switches, and Null and Poll capture.
- ♦ Captured data includes data bytes and error conditions.
- ♦ Data can be captured to RAM or directly to disk.
- ♦ Capture buffer sizes are limited only by available memory or disk space.

### Data Displays

- ♦ Data can be analyzed in realtime using the Event Display, Frame Display, Signal Display, Protocol Navigator, and Statistics Windows.
- ♦ Decodes and displays multiple protocol layers of multiple data frames simultaneously.
- ♦ Data can be analyzed simultaneously in each of the product's three operating modes: air, serial HCI, and Live Import.
- ♦ Multiple synchronized windows can be viewed simultaneously.
- ♦ Byte level information is displayed in either a split-line format, or in a mixed format.
- ♦ Characters can be displayed in ASCII, EBCDIC, or Baudot.
- ♦ Nonprintable characters are displayed using hex and mnemonics.

### Error Detection

- ♦ Detects and displays protocol errors (in red) in realtime.
- ♦ Packet Timeline chart indicating packet type, retransmitted packets, error type and throughput over time.

### Search criteria include:

- ♦ Byte Level: timestamp, errors, and patterns (patterns include wildcard characters at the bit level, nibble level, and byte level);
- ♦ Frame Level: frame decodes can be searched on the text of the decode, with or without wildcards.

### Display Filters

- ♦ Display Filter parameters include protocol, data pattern, and an advanced mode for complex filters.

### Statistics Display

- ♦ Provides session totals for characters, frames, events, errors, characters per second, and percentage utilization.

### Data Transmission (serial HCI only)

- ♦ Data strings and files can be transmitted one time, multiple times, or continuously.
- ♦ Data transmission delays can be inserted in millisecond, second, or minute increments.

### Timestamping

- ♦ Sniffs data in three modes simultaneously, with synchronized timestamping.
- ♦ Provides both absolute and relative displays of event timing.
- ♦ Calculates time intervals between events (delta time) and data rate.
- ♦ Microsecond resolution.

### Configuration Management

- ♦ I/O configurations, protocol stacks, and filters can be saved for future use.
- ♦ Supports user-definable protocol stacks.

### Help

- ♦ Comprehensive online help provides complete operating instructions.
- ♦ Quick Start Guide enables rapid product set-up and operation.

### FrameDecoder

- ♦ Quickly create custom decoders for proprietary protocols and Vendor Extensions.

### Air Sniffing

- ♦ Air interface is a small lightweight USB dongle.
- ♦ Any Windows PC (except NT and 95) with a USB port and 64 MB RAM.

### Virtual Sniffing

- ♦ Any Windows PC with 64 MB RAM.

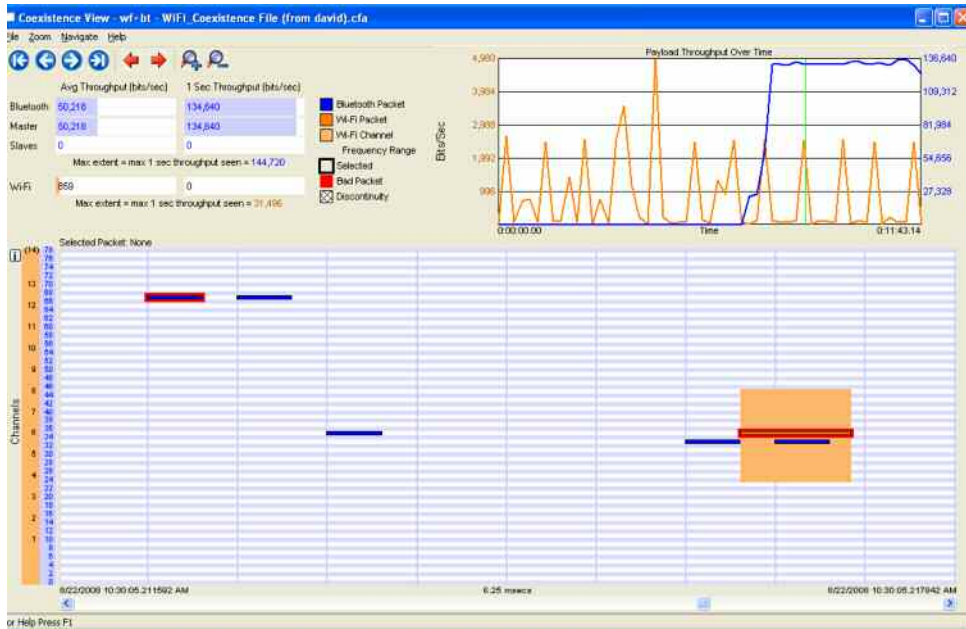
### Serial HCI Sniffing

- ♦ Supports HCI UART (H4) and BCSP.
- ♦ Up to 115.2 Kbps using any Windows PC with two serial ports and 64 MB RAM.
- ♦ FTS monitoring cables included.
- ♦ Up to 921 Kbps with optional PCI or PCMCIA card.
- ♦ Supports board-level (TTL) voltages with optional TTL to RS-232 converters.
- ♦ Optional HS-UART up to 8 Mb/s incl. HCI decoding.

<b>Protocols Supported:</b>	<p><b>Bluetooth® stack</b>  Baseband, LMP, L2CAP, HCI, AMP-mgr, MCAP, UART (H4), HCI, OBEX, TCS Binary, CMTTP, HDLC, SDP, RFCOMM, TCS, AVDTP, BNEP, HCRP, HID, AVCTP, AVDTP, OBEX, OBEX over L2CAP</p> <p><b>TCP/IP stack</b>  Decodes all key Dial-Up Networking and TCP/IP protocols IPv4, HTTP, NBNS, UDP, IPv6 DHCPDNS, NBSS, TCP, ICMP, SMB, ARP, NBDS, SMTP</p> <p><b>Common LAN Protocols</b>  802.2, SNAP, IPX, NCP, IBM_NETBIOS, AT commands</p> <p><b>Common WAN Protocols</b>  AsyncPPP, PPP</p>
<b>Profiles Supported:</b>	<p><b>Audio visual:</b> A2DP, AVRCP, GAVDP, VCP, VDP, HDP  <b>Car:</b> HFP, PBAP, SAP  <b>Printing:</b> BIP, BPP, HCRP, PBAB  <b>Other:</b> CIP, CTP, DUN, FAX, FTP, GOEP, HID, HSP, ICP, LPP, OPP, PAN, SDAP, SPP, SYNCH, UDI</p>
<b>Bit Order:</b>	LSB (normal) or MSB (reversed) first
<b>Character Sets:</b>	ASCII, 7-bit ASCII, EBCDIC, Baudot
<b>Parity:</b>	None, Even, Odd, Mark, Space
<b>Word Length:</b>	5, 6, 7 or 8 bits
<b>Stop Bits:</b>	1, 1.5 or 2 bits
<b>Timestamping:</b>	Absolute and relative timestamping: microsecond resolution. Timestamps are synchronized over all three operation modes.

## FTS4BT Wi-Fi option

Data Throughput stats enable to monitor Data throughput activity on Bluetooth and Wi-Fi simultaneously. A common graph assures that the application is operating at its intended efficiency. Packets are displayed by channel, in real time, timestamped and color coded. Combined capture log with full WiFi decoding and protocol analysis.



## Order numbers:

FTS for Bluetooth for a single user with twice hardware (ComProbe dongles) and software, warranty and one year software care (Premium maintenance)  
The main ComProbe can be used on any PC with the installed software. The secondary ComProbe only works in conjunction with the main ComProbe.

with standard license: FTS4BT  
Further secondary ComProbe dongle: COM-PROBE

FTS for Bluetooth for dual usage with hardware (ComProbe dongle) and software, warranty and one year software care (Premium maintenance)

with universal license: FTS4BT-U  
Second universal ComProbe dongle: COMPR-U

One year additional Premium Maintenance: BT-PM1  
Three years of additional Premium maintenance: BT-PM3  
Upgrade standard FTS4BT to universal: UNIV UPG

For further Options please see separate data sheet

