

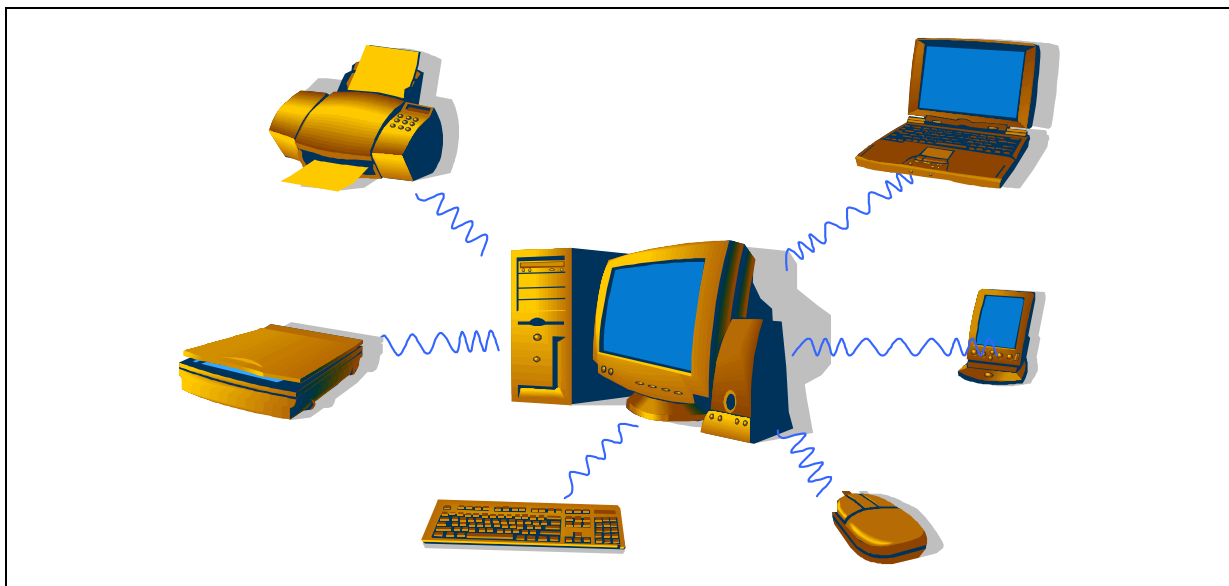




<b>1</b>	<b>INTRODUCTION .....</b>	<b>5</b>
<b>2</b>	<b>BLUETOOTH LAYERS.....</b>	<b>7</b>
2.1	OVERVIEW .....	7
2.2	BLUETOOTH RADIO .....	8
2.2.1	Class of the Device.....	8
2.2.2	Modulator Characteristics.....	8
2.2.3	Interference Performance.....	9
2.2.4	RF Front End Architecture.....	10
2.2.5	Frame and Clock Synchronization.....	10
2.2.6	Overall Performances.....	11
2.3	BASEBAND .....	11
2.3.1	Packets format .....	12
2.3.2	Packet Types .....	13
2.3.3	Logical Channels .....	14
2.3.4	Transmit/Receive Routines .....	14
2.3.5	Transmission Timings.....	15
2.3.6	Channel Control.....	16
2.3.7	Error Correction .....	17
2.3.8	Data Whitening.....	17
2.3.9	Hop Selection .....	18
2.3.10	Bluetooth Audio .....	18
2.3.11	Bluetooth Security .....	18
2.4	LMP: LINK MANAGER PROTOCOL.....	19
2.5	HCI: HOST CONTROLLER INTERFACE .....	20
2.5.1	Overview.....	20
2.5.2	Commands .....	20
2.5.3	Packets.....	21
2.6	L2CAP: LOGICAL LINK CONTROL & ADAPTATION PROTOCOL .....	22
2.6.1	General Operations .....	22
2.6.2	Data Packet Format.....	22
2.6.3	Signalling .....	23
2.6.4	Primitives.....	23
2.7	SDP: SERVICE DISCOVERY PROTOCOL .....	24
2.8	RFCOMM (BASED OVER ETSI TS GSM 07.10).....	25
2.9	TELEPHONY CONTROL PROTOCOL SPECIFICATION BINARY (TCS BIN) .....	26
<b>3</b>	<b>PROFILES .....</b>	<b>27</b>
3.1	GENERIC ACCESS PROFILE (GAP) .....	27
3.2	SERVICE DISCOVERY PROFILE (SDP) .....	27
3.3	BINARY TCS PROFILE.....	27
3.4	SERIAL PORT PROFILE .....	28
3.5	GENERIC OBJECT EXCHANGE PROFILE (GOEP).....	29
<b>4</b>	<b>ST STRATEGY .....</b>	<b>31</b>
<b>5</b>	<b>WEBSITES OF INTEREST .....</b>	<b>31</b>
<b>6</b>	<b>GLOSSARY .....</b>	<b>32</b>
<b>7</b>	<b>REVISION HISTORY .....</b>	<b>45</b>



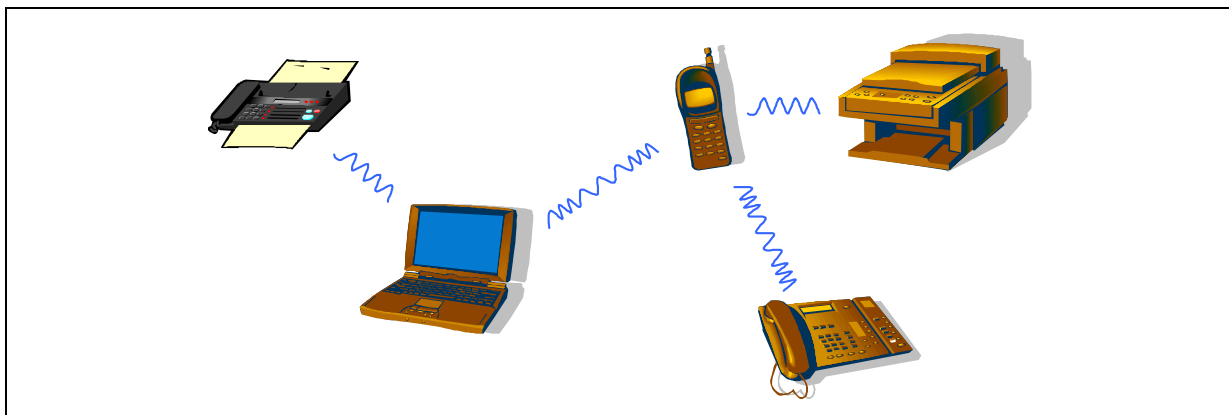
## 1 - INTRODUCTION



Bluetooth is a new wireless protocol that allows devices of any kind to discover themselves and communicate without need of user. Two Bluetooth units just have to be less than 10 meters away to be able to exchange information.

This affords a wireless world, especially in:

- Offices, with Bluetooth in: keypads, mice, printers, notebooks, mobile phones, PDAs, faxes...
- Living rooms, with Bluetooth in: TVs, game stations, Hifi, MP3 readers, headsets...
- Cars, with Bluetooth in: keys, headsets, mobile phones, and navigation platforms...

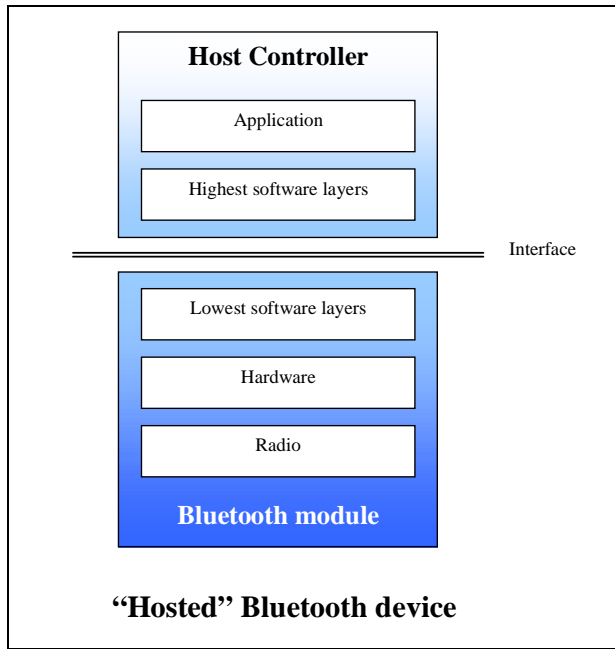


Besides cable replacement (e.g. between an application running on a PC and a modem), Bluetooth also provides numerous services as auto-detection, service browsing (discovering of available services delivered by the devices) and so on. It supports numerous protocols, and allows multiplexing (i.e. numerous links at the same time).

Bluetooth devices are organized in mini-networks, where one device plays the role of master and all other ones the role of slave.

Between devices either data or voice can be exchanged.

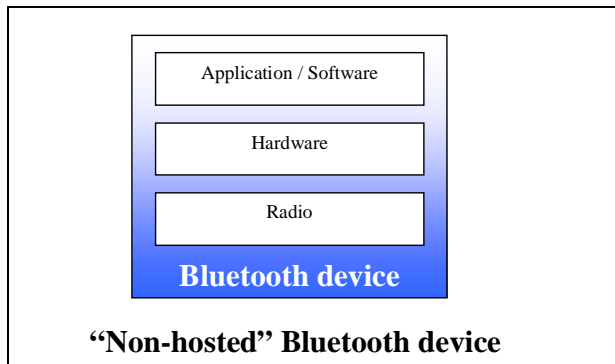
At present, two types of Bluetooth devices exist: "hosted" devices and "non-hosted" devices. Hosted devices are actually made of two parts. The first one is the Bluetooth module itself, integrating the Bluetooth radio, hardware and the lowest part of the software. The second part is hosted by the host controller (e.g. a PC). It consists in the highest part of the software, and application (typically legacy application) using it.



The two parts making up the Bluetooth device (i.e. the module + the host controller) are bounded with a physical link, typically USB or RS232.

To ensure portability and compatibility, a logical Host Controller Interface is defined, which lists all the commands that a Bluetooth module has to understand, so that any kind of application running on any kind of host controller is able to drive a bluetooth module.

Furthermore, the whole software stack functions are precisely defined, with its specific or adapted layers, once again to ensure compatibility between different vendors' solutions.

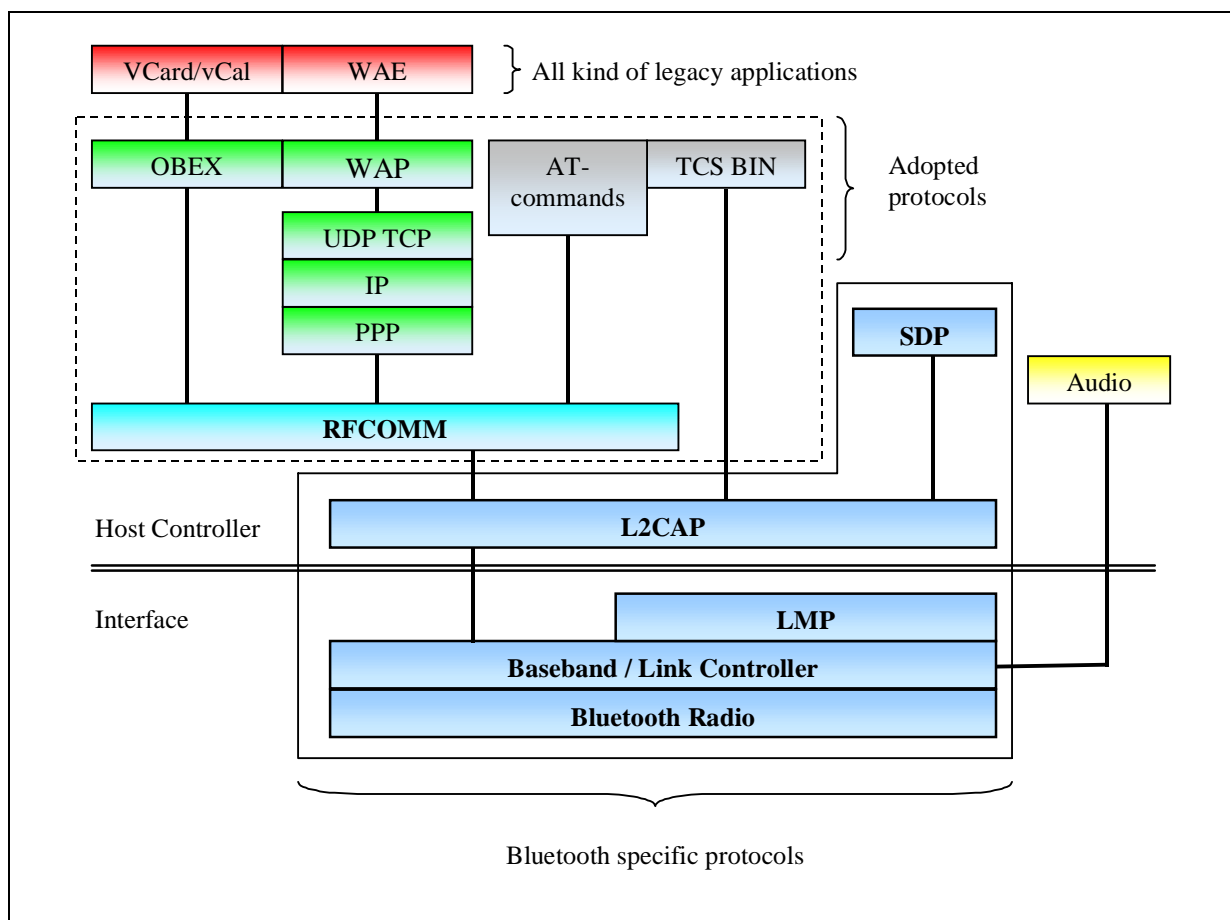


The second type of device is "non-hosted" device. In fact, it means that the whole software and the application (typically specific) is embedded in the Bluetooth module. For example, a Bluetooth headset is the perfect type of device that doesn't need anyway a host controller. A really basic application runs directly over the hardware and the radio. So it doesn't need generic interface (neither physical nor logical).

Note that finally, there is no difference from the user point of view between these two types of implementation.

## 2 - BLUETOOTH LAYERS

### 2.1 - Overview



Remark: the Bluetooth radio is totally hardware, and the baseband is both software and hardware.

Each part of the whole stack has a specific role to play. **The link controller and the link manager protocol (LMP)** are up to manage the physical connection between two Bluetooth devices in itself, following the specific protocol over the air Bluetooth protocol.

The **logical link controller and adaptation protocol (L2CAP)** has to transform data coming from higher layers into packets that can be handled by the baseband. It also manages logical connections between two Bluetooth devices, allotting logical links between two remote applications.

The **service discovery protocol (SDP)** has to find out which services are provided by other devices in range.

The **RFCOMM** module is mainly a multiplexer module, using virtual serial ports. It assigns a logical channel to each application using the Bluetooth module.

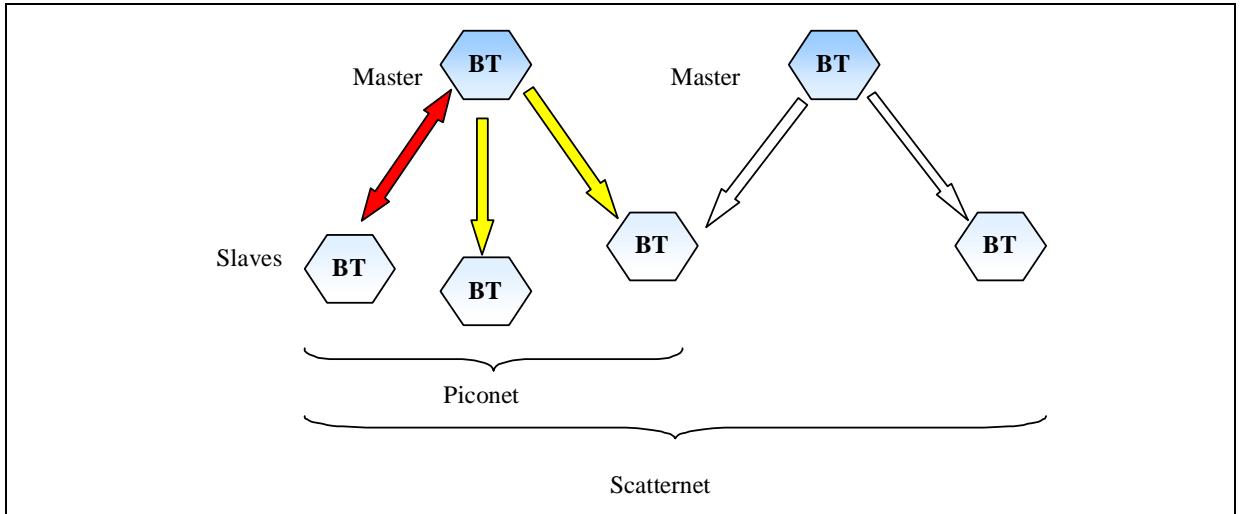
Higher software layers are used depending on the application that runs over the host controller.

While always using the same over-the-air interface (Bluetooth RF), two BT Devices can exchange data messages or control messages between two identical layers (e.g. from local Link Manager to remote Link Manager ) using specific protocols. Thus, messages are not always generated by the host controller.

To resume, exchanges between two devices rely on:

- A physical link, provided by the Bluetooth radio, and managed by the link controller and the link manager;
- A logical link, between two Bluetooth devices, managed by the L2CAP module;
- A local logical channel, assigned by the RFCOMM module.

In range Bluetooth devices are organized in mini-networks:



- Notes:
- A piconet is a temporary network between 1 master and up to 7 slaves.
  - A piconet is totally defined by its master. The master sets all the “rules” during the existence of this network.
  - A Bluetooth device can be at the same time master of a piconet and slave in other ones

**2.2 - Bluetooth Radio**

This module creates the physical link between two BT devices. The RF signal is transmitted in the ISM band ( Industrial – Scientific – Medical unregulated band ). Its uses 79 sub carriers with 1MHz interband between 2.402GHz and 2.480GHz. To avoid disturbance induced by equipment already operating in the same band of frequency, mainly microwave ovens, a fast frequency hopping sequence is used.

Within each piconet, a single pseudo-random hopping sequence synchronizes all the slave devices with the master. With a fast frequency of 1600hops/s the fading problem is solved, avoiding the necessity of a diversity antenna system.

**2.2.1 - Class of the Device**

Three classes of devices are defined according to the maximum transmit power level it must produce.

Class Name	Max Transmit Power	Power Control	Power Measurement
1	100mW (+20dBm)	Yes	Yes
2	2.5mW (+4dBm)	Optional	Optional
3	1mW (0dBm) and below	Optional	No

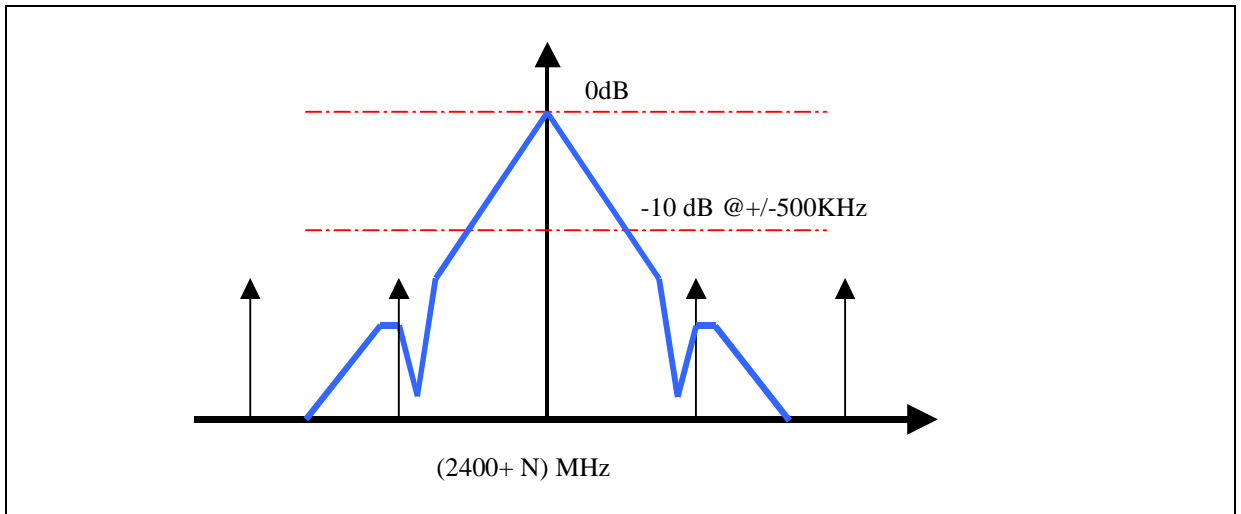
However, even in Class 2, devices generally implement Power Control and Measurement (RSSI) in order to provide less disturbance in a heavily saturated network. Class 1 devices use external PA amplifier.

**2.2.2 - Modulator Characteristics**

The main specification of the modulator are defined by:

Parameter	Value	Comments
Type of Modulation	GFSK	BT = 0.5
Modulation Index	0.32	-
Channel Spacing	1 MHz	-
Frequency Hopping	1600 Hops/s	-
Number of Channels	79	23 in few countries

This Modulator bloc shapes the transmit spectrum as shown below:

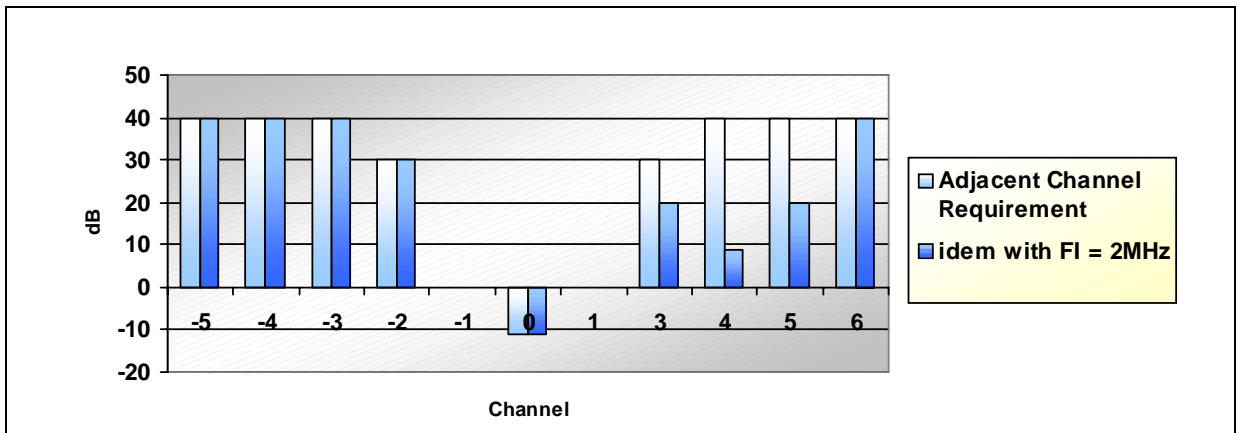


### 2.2.3 - Interference Performance

The Bluetooth system is intended to perform in a highly noisy environment. Two sorts of interferences are possible. A first source of interferences are static transmitters, like microwave oven, or strong static transmitter. In this case the performance of the BT device is only depending on its fast frequency hopping mechanism.

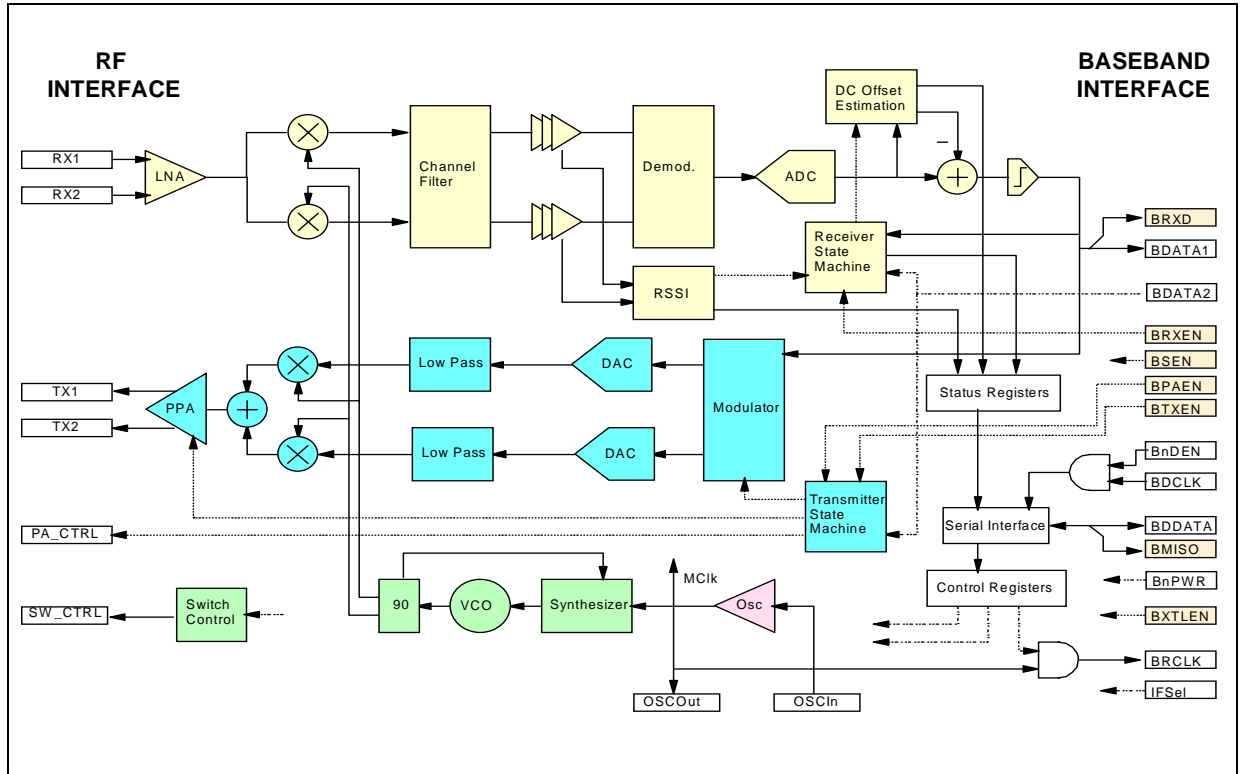
The second source of interference is the Bluetooth network itself, in a heavy multi piconet system, where each device interferes with adjacent piconet devices.

In order to minimize the interference and to guarantee a good level of service, the Bluetooth radio must meet the following adjacent channel rejection specification ( in dB of unwanted signal versus the signal's channel number ).



### 2.2.4 - RF Front End Architecture

Due to the fact that, since the beginning, the BT specification have been defined to be implemented with low cost solution; all the major implementations use Low IF Architecture. Below is a typical block diagram for low cost implementation:



### 2.2.5 - Frame and Clock Synchronization

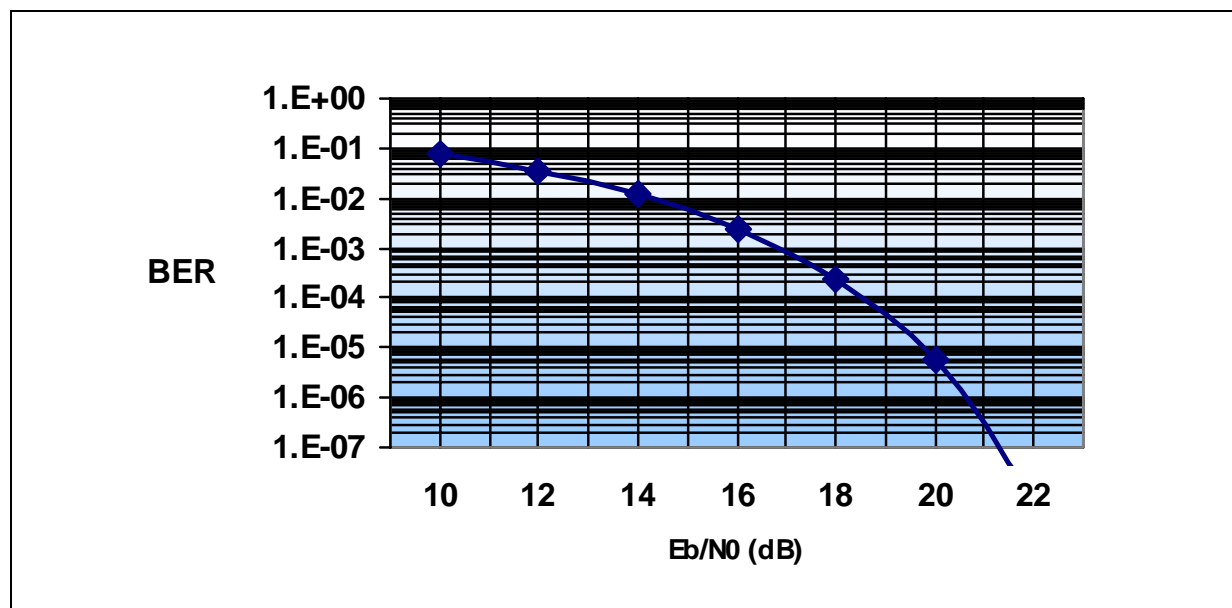
To complete the receiver and get demodulated bits from the RF receiver two additional blocks, generally contained in the Baseband part are needed:

- The Clock Recovery Block: this block will extract the bit clock from the digital signal coming from the demodulator included in the RF part. This block need to have both fast frequency recovery and long term tracking performance in order to be able to give good performances in Bluetooth long frame data mode (DM5 and DH5).
- The Address Correlator Block: this block extracts from the demodulator's raw data output both the Frame timing and the fact that the device have received the right address.

The Bluetooth protocol having a very low preamble (only 5 bits) these 2 mechanisms are closely synchronized with the block that computes the difference between the two units crystal oscillator frequencies ( DC Offset cancellation block ). This makes up the setup of both blocs difficult during the preamble.

## 2.2.6 - Overall Performances

Below is the Bit Error Rate versus  $E_b/N_0$  performance of the non-coherent Demodulator block:



The main performances of a Bluetooth receiver ( sensitivity and adjacent channel rejection ) are defined for a BER of 1E-3.

However, in order to be able to transmit high data rate packets (as DM5 and DH5) a BER of 1E-6 is mandatory. This will require, according to the demodulator performances an  $E_b/N_0$  improvement of about 4dB.

## 2.3 - Baseband

The Baseband module role is to transmit to the radio module the data to be sent. It encapsulates data from higher layer into specific packets according to the Bluetooth protocol.

Packets are sent via physical channels (i.e. RF transmission) on logical channels divided into 625 $\mu$ s time slots. Frequency hops are done (basically) each time slot. To ensure full duplex communication, baseband module uses time duplex division.

To perform such multiplexing, Bluetooth protocol uses a combination of circuit and packet switching. Circuit switching uses dedicated channels to send messages, and packet switching uses packets with a delivery address to transmit data.

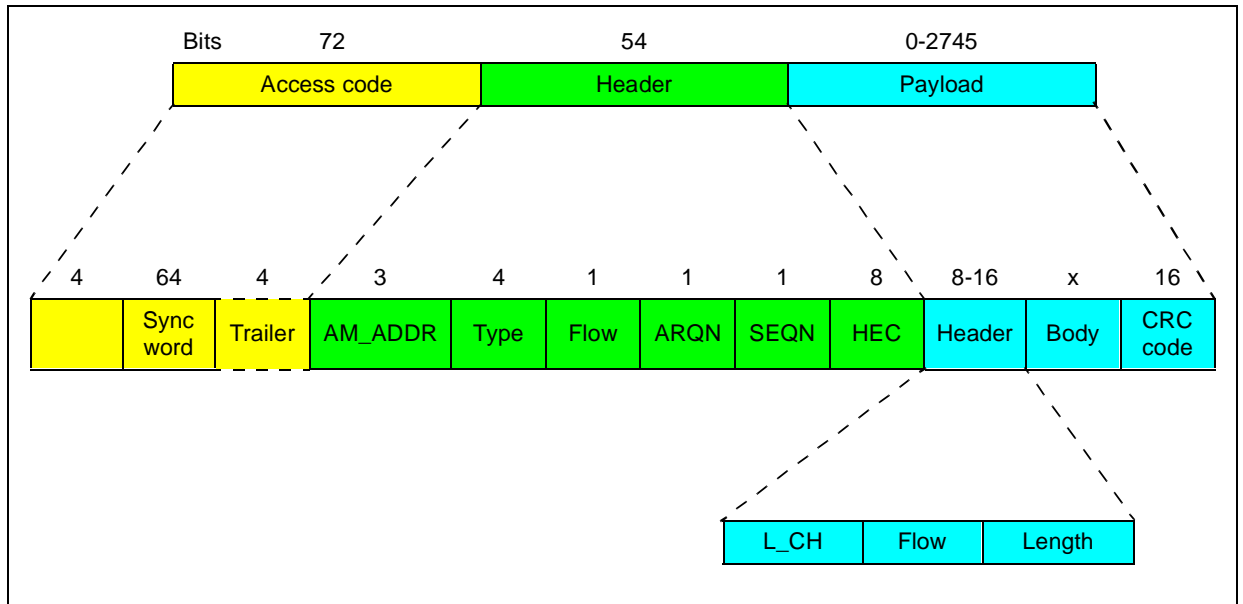
Different kind of physical links:

- SCO (Synchronous Connection Oriented) link: a point-to-point link (i.e. master to slave link);
- ACL (Asynchronous Connectionless) link: point-to-multipoint link.

A master Bluetooth device supports up to three SCO links in a symmetric link from a master to a specific slave. A slave supports either up to three SCO links from the same master or two SCO links from different masters.

SCO packets are exchanged during reserved time slots. ACL packets are exchanged during remaining slots.

## 2.3.1 - Packets format



### – ACCESS CODE

There are three types of access code, depending on the kind of message sent:

Channel Access Code ( CAC )

Device Access Code ( DAC )

Inquiry Access Code ( IAC )

These access codes have no trailer  
(see above diagram)

### – HEADER

AM\_ADDR : Active member address

Type : NULL, POLL, HV, DM... (see "packet type" )

Flow (ACL flow) : Rx buffer full -> 0; empty -> 1

ARQN : Acknowledge indication

SEQN : Sequence number

HEC : Header error check

### – PAYLOAD

#### Header:

L\_CH = Logical Channel. Possible values:

00: undefined  
01: continuation of an L2CAP message  
10: start of an L2CAP message  
11: LMP message

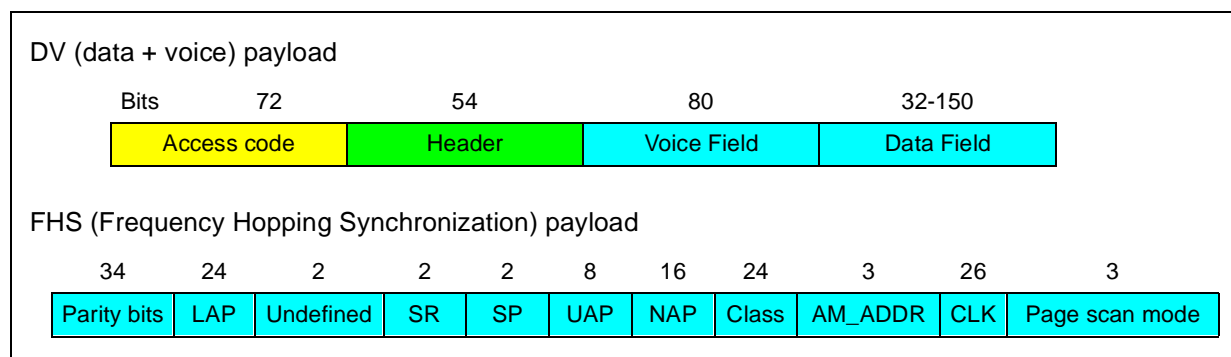
#### Body:

Includes the user host information

#### CRC code:

16-bit Cyclic Redundancy Code, to prevent transmission errors.

– SPECIAL PAYLOAD FORMATS



2.3.2 - Packet Types

Type	SCO link	ACL link	
0000	NULL	NULL	Command packets
0001	POLL	POLL	
0010	FHS	FHS	
0011	DM1	DM1	
0100	/	DH1	1 time-slot packets
0101	HV1	/	
0110	HV2	/	
0111	HV3	/	
1000	DV	/	3 time-slots packets
1001	/	AUX1	
1010	/	DM3	
1011	/	DH3	
1100	/	/	5 time-slots packets
1101	/	/	
1110	/	DM5	
1111	/	DH5	

– COMMAND PACKETS

- NULL packet has no payload. Fixed length of 126 bits. Used to return link information to the source.
- POLL packet is very similar to NULL packet. But in contrast to NULL packet, it requires a confirmation from the recipient.
- FHS packet is used to set frequency hopping corresponding to a physical channel (deducted from the master’s BD\_ADDR and the master’s clock). FHS packet is also used to synchronize devices clocks. The payload contains 16-bit CRC code plus 144 information bits coded with a rate of 2/3 FEC, increasing its length to 240 bits.
- DM1 packet can support any kind of control message or regular user data.
- In addition to the four commands packets, the ID packet consists in a Device Access Code (DAC) or an Inquiry Access Code (IAC), with a fixed length of 68 bits. It is used, for example, in paging, inquiry and response routines.

## – SCO PACKETS

Type	User payload (bytes)	FEC	CRC	Rate (Kb/s)
HV1	10	1/3	No	64
HV2	20	2/3	No	64
HV3	30	No	No	64

HV: High Voice quality.

In addition to HV types, you can find the DV (Data-Voice) packet, which consists in 1 payload header byte; 10 voice bytes plus 0 to 9 data bytes user payload. Data is coded with a 2/3 FEC and has CRC. Rate is 64K bit/s for voice plus 57.6K bit/s for data (symmetric).

## – ACL PACKETS

Type	Payload header (bytes)	User payload (bytes)	FEC	CRC	Symmetric rate	Asymmetric rate (in Kbytes/s)	
						Forward	Reverse
DM1	1	0-17	2/3	Yes	108.8	108.8	108.8
DH1	1	0-27	No	Yes	172.8	172.8	172.8
DM3	2	0-121	2/3	Yes	258.1	387.2	54.4
DH3	2	0-183	No	Yes	390.4	585.6	86.4
DM5	2	0-224	2/3	Yes	286.7	477.8	36.3
DH5	2	0-339	No	Yes	433.9	723.2	57.6
AUX1	1	0-29	No	No	185.6	185.6	185.6

DM: Data Medium rate packets. They are used when interferences may change the data.

DH: Data High rate packets. They are used in "clean" places: they have no FEC.

### 2.3.3 - Logical Channels

In Bluetooth, five logical channels are defined:

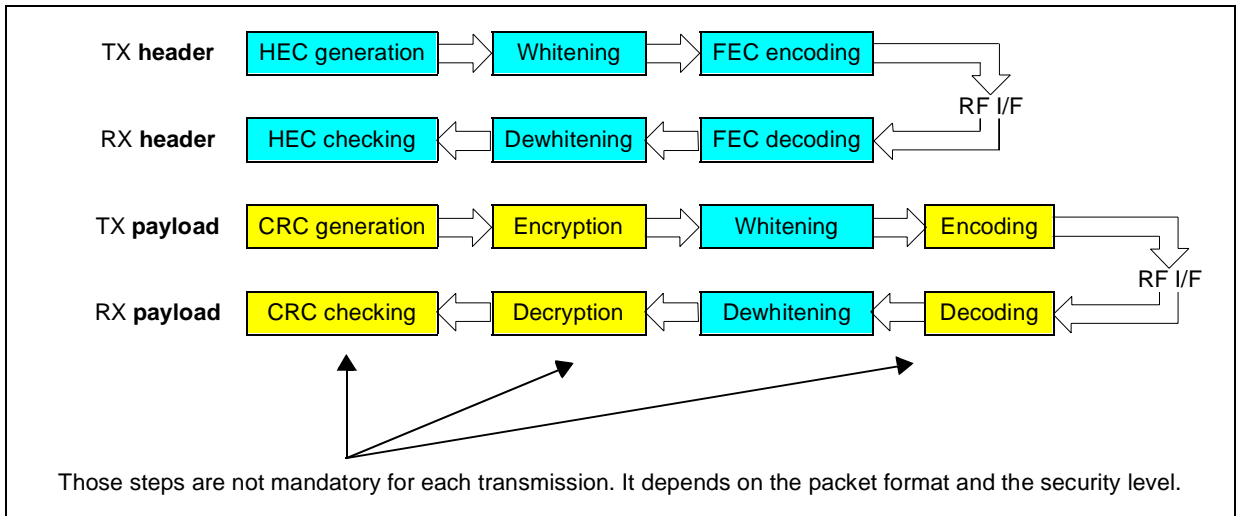
Channel	Type	Role
LC (Link Control) channel	Control	Carries low-level link information (e.g. ARQ, flow control...)
LM (Link Manager) channel	Control	Carries information between link managers of master and slave
UA (User Asynchronous) channel	ACL user	Carries L2CAP transparent asynchronous user data
UI (User Isochronous) channel	ACL user	Carries L2CAP transparent isochronous user data (special kind of ACL data)
US (User Synchronous) channel	SCO user	Transparent synchronous data

### 2.3.4 - Transmit/Receive Routines

There is a separate ACL buffer for each slave, and one or more SCO buffer for each SCO slave (separate TX and RX buffers).

Flow control: if RX ACL buffer is full: FLOW (header field) = 0.

The bitstream processes ( performed by Ericsson Blue Core) are described below:

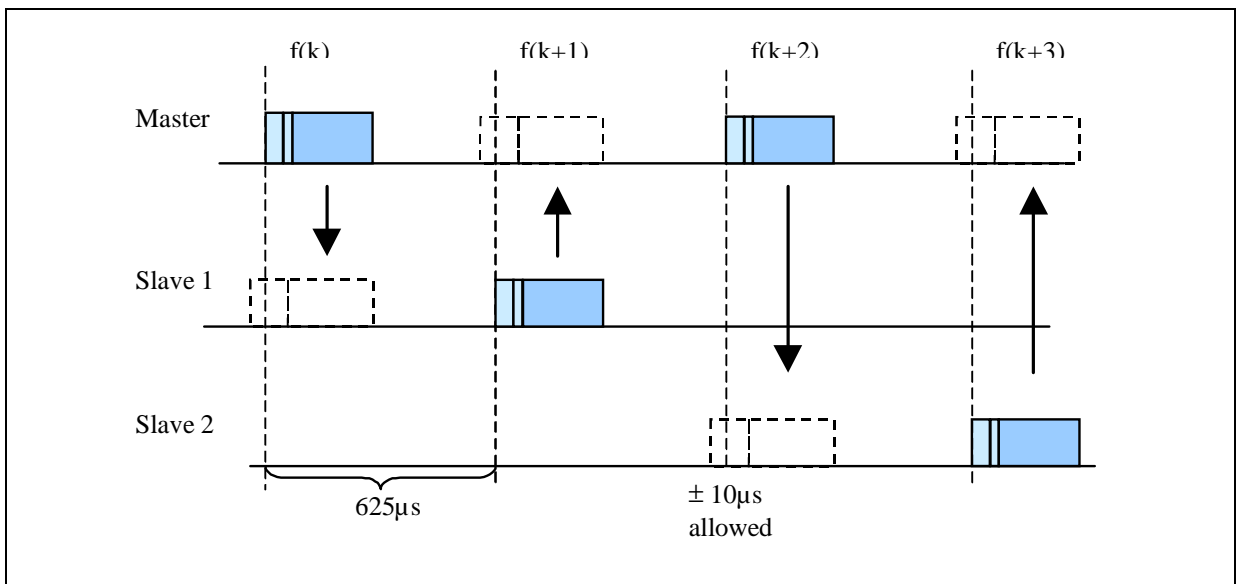


### 2.3.5 - Transmission Timings

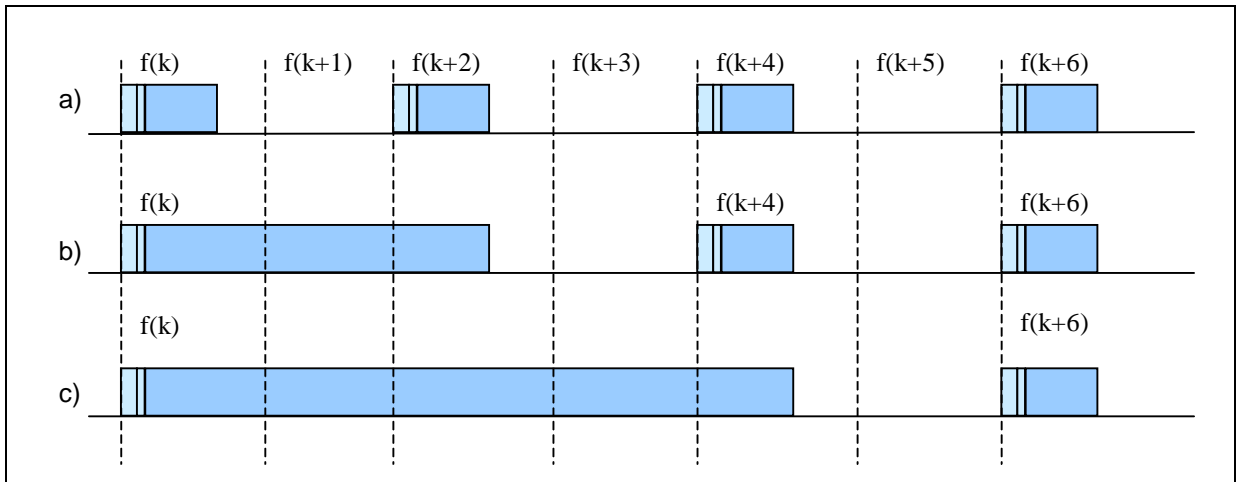
When establishing a piconet, all the slaves synchronize to the master's clock by adding an offset to their own clock (offset frequently updated when receiving RX packets), so that all devices use the same logical clock.

Packets are sent via a logical channel divided into 625µs time-slots. Frequency hopping occurs between each slot. The time-slots are numbered (cyclically from 0 to  $2^{27}-1$ ) so that, as the frequency hopping sequence is known by each participant (sent in the FHS messages), all devices in the piconet stay synchronized.

The Bluetooth transceiver uses a time-division duplex (TDD) scheme: RX and TX packets are sent alternatively, at the beginning of each 625µs time-slot. Packet size is up to 366µs. An uncertainty time-window of 20µs is defined, authorizing the RX burst to arrive up to 10µs too early or 10µs too late. A master shall always starts to transmit in odd time-slots, whereas the slave addressed transmits during even time-slots. (A slave can only answer to a master, and can't initiate a conversation).



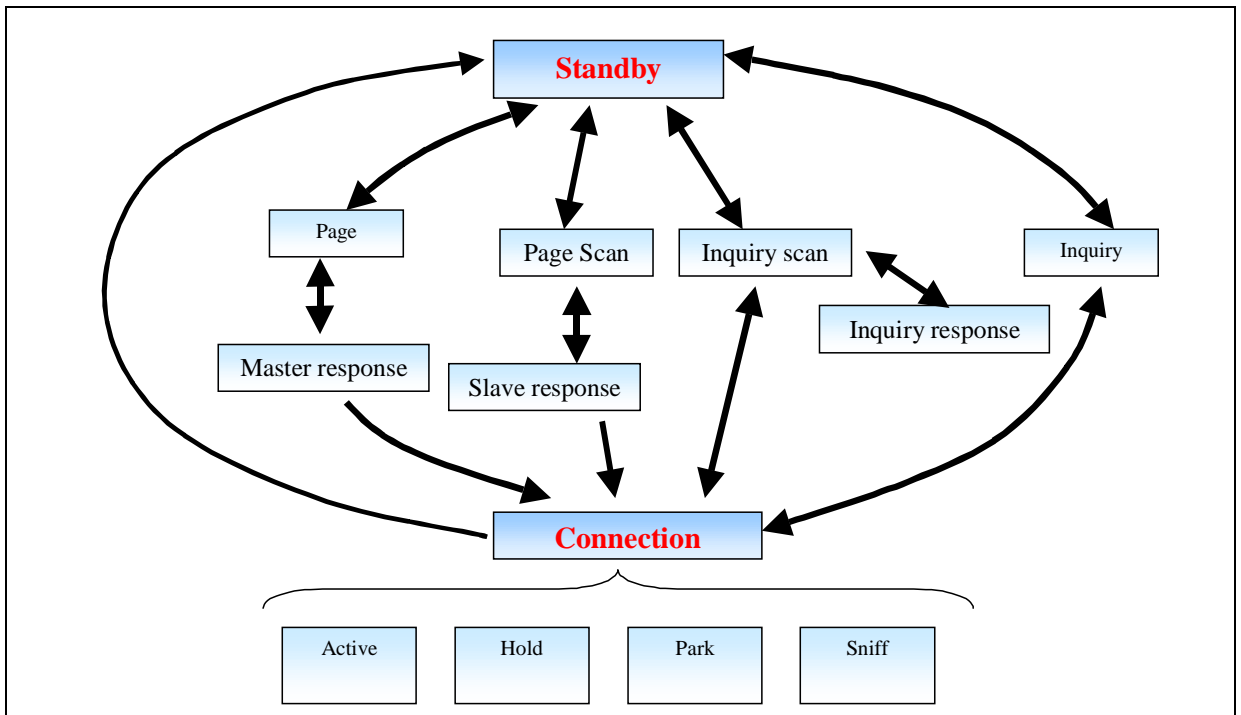
When multi-slots packets (DM3, DH3, DM5...) are sent, the frequency is the same during the whole length of the packet, and then hop to the frequency value  $f(k+n)$  corresponding to the time-slot number  $k+n$ .



**2.3.6 - Channel Control**

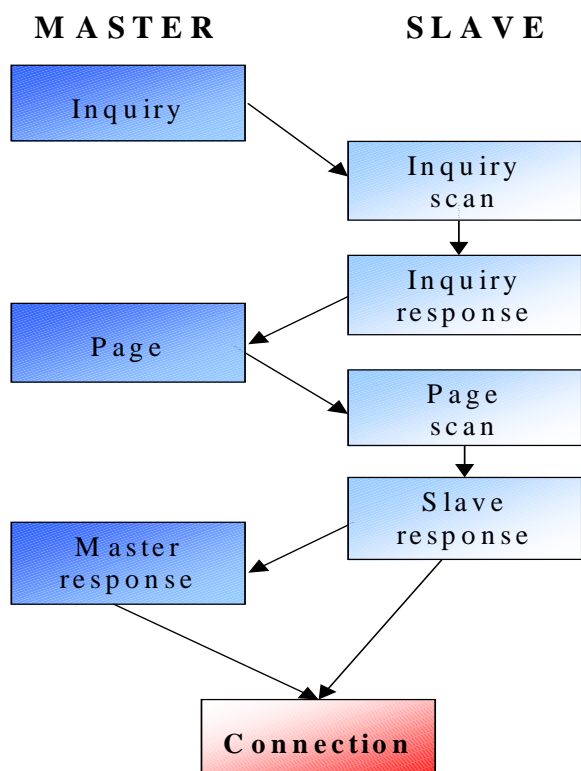
A piconet is entirely defined by the master ( Initially, the master is the device who first initiated the piconet, but then swaps between the master and a slave occur really often ). The master's BD\_ADDR generates the FH hopping sequence and the channel access code; and the master's clock is the root of the whole timing.

There are two main states for Bluetooth devices: standby and connection, plus a lot of sub-states between those two states:



Connection states	Description
Active	Both master and slave participate on the channel and are kept synchronized
Sniff	Slave listens for master message only on specified time-slots, thus enter a low-power mode
Hold	The device cannot support ACL and only accepts special kinds of messages as paging, scanning... Low power mode.
Park	Device only wants to remain synchronized, but doesn't need to participate actively to the channel: very low power mode

The different steps to connect two Bluetooth devices are described below.



Connection step	Description
Inquiry	A discovering unit collects BD_ADDR and clocks of all BT devices in range which answer its inquiry message
Inquiry scan	BT devices are listening for scan messages
Inquiry response	Only slave responds, with a FHS packet
Page	Master sends page message to a slave (with slave's Device Access Code) to activate and connect it.
Page scan	Slave listens for its own DAC
Slave response	Slaves respond to master's page message and enter connection state (after receiving FHS packet).
Master response	After slave response, master send a FHS message and wait for a replies to enter connection state

### 2.3.7 - Error Correction

There are three error correction schemes defined for Bluetooth:

- **1/3 rate FEC** (Forward Error Correction). With it, each bit is simply repeated 3 times for redundancy.
- **2/3 rate FEC**. A generator polynomial is used to encode a 10-bit code to a 15-bit code. (It is a shortened Hamming code).
- **ARQ schemes**. With an Automatic Repeat reQuest, packets are transmitted and retransmitted until an acknowledgement is returned (or until time-out is exceeded). ARQ scheme only works on the payload in the packet. To determine whether the payload has been successfully transmitted, a CRC code is added to the packet.

### 2.3.8 - Data Whitening

Before transmission, both the header and the payload are scrambled with a data whitening word in order to randomize the data and avoid redundant patterns. It is performed before FEC encoding.

Received data is later descrambled with the same data whitening word (after FEC decoding).

### 2.3.9 - Hop Selection

There are five different hopping sequences defined. (Note that in fact, ten different sequences are defined: five are for 79-hop system, the others for 23-hop system. But as the 23-hop system seems to be disappearing, only the former will be described.)

- A page hopping sequence, with 32 unique wake-up frequencies distributed equally over the 79MHz;
- A page response sequence, covering 32 unique response frequencies in correspondence to the current page hopping sequence;
- An inquiry sequence, with 32 unique wake-up frequencies distributed equally over the 79MHz;
- An inquiry response sequence, covering 32 unique response frequencies in correspondence to the current page hopping sequence;
- A channel hopping sequence, covering the 79MHz.

### 2.3.10 - Bluetooth Audio

Audio transmission on the Bluetooth air-interface is made directly between two basebands on SCO links (which don't pass through L2CAP). Codec use either a 64kb/s log PCM format (A-law or  $\mu$ -law) or a 64Kb/s CVSD (Continuous Variable Slope Delta Modulation). Though a little harder to be implemented, CVSD provides a graceful degradation of the signal in bad transmission conditions.

### 2.3.11 - Bluetooth Security

Security in Bluetooth is a great issue as any device can access another one via the over-the-air interface. There are two main built-in security techniques to prevent eavesdropping (spying) and falsifying: authentication and encryption. Authentication is used to prevent unwanted access to data and to prevent the falsification of the message originator. Encryption is used to prevent eavesdropping or spying.

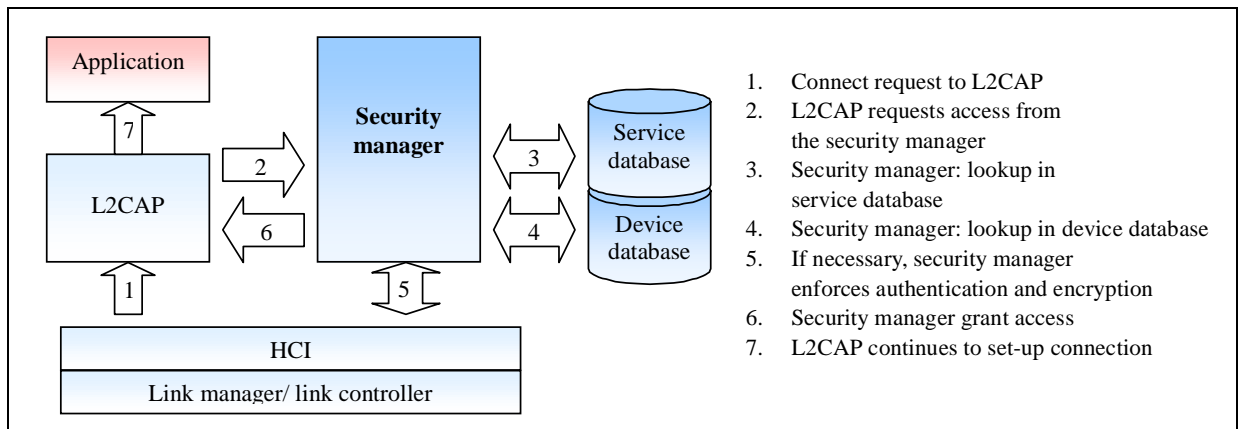
These features are based on a secret link key (shared by the paired devices) generated during pairing procedure. The Generic Access Profile defines three security modes for a device:

- Security mode 1 (non-secure): a device will initiate no security procedure;
- Security mode 2 (service-level enforced security): no security procedure is initiated until L2CAP channel establishment;
- Security mode 3 (link-level enforced security): security procedures are initiated before the link set-up at the LMP level is completed.

To provide such services, four entities are defined:

Entity	size
BD_ADDR	48 bits
Private user key, for authentication	128 bits
Private user key for encryption	8-128 bits
RAND (random number)	128 bits

A security manager can be implemented. It will manage all authentication and encryption procedures, store security-related information of both services and devices. For example:



1. Connect request to L2CAP
2. L2CAP requests access from the security manager
3. Security manager: lookup in service database
4. Security manager: lookup in device database
5. If necessary, security manager enforces authentication and encryption
6. Security manager grant access
7. L2CAP continues to set-up connection

The most common way to test a remote device is to send it a challenge ( based on a secret key ) and expect from it a correct challenge response.

## 2.4 - LMP: Link Manager Protocol

LMP module is used for:

- Link set-up
- Security
- Control

Format of LMP PDUs (Protocol Data Unit = LMP messages ), which are transferred in the payload ( with L\_CH = 11):

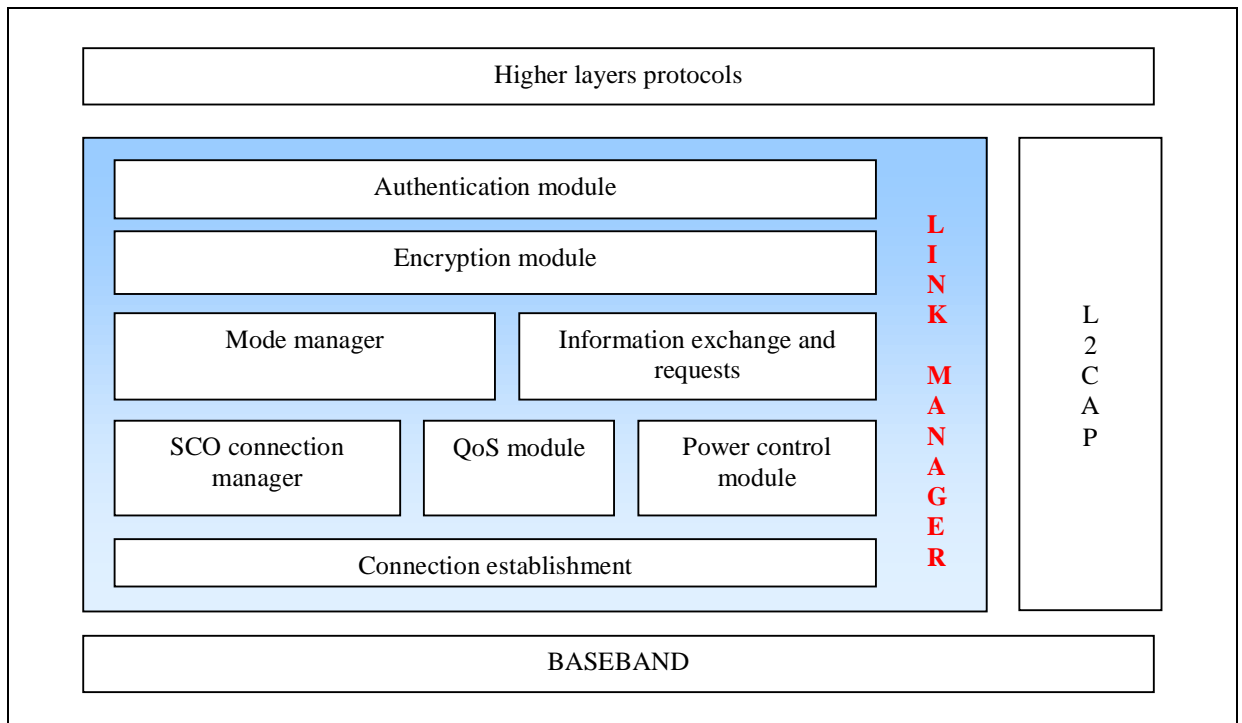
OpCode and transaction Id	Content
---------------------------	---------

A link manager essentially talks to the remote one to exchange information and control through the link controller. Exchanges with higher layers are possible but a little bit hazy (they could be of interest to inform the LM about all kind of settings such as security level, QoS, and so on...).

A BT LM can request from other LM all kind of information as:

- The clock offset of the remote device
- The slot offset (useful in master-slave switch)
- Timing accuracy
- LM version
- ...

The link manager also manages: authentication and encryption, mode and SCO connections:



## 2.5 - HCI: Host Controller Interface

### 2.5.1 - Overview

The HCI provides a uniform interface to access the Bluetooth hardware capabilities. For “hosted” devices, the software stack is divided into three parts:

- Legacy application over numerous layers (BT specific or not);
- Specific Bluetooth protocols embedded in the Bluetooth device: link manager and link controller;
- A logical and physical interface (USB, RS232...) to link the host and the BT device.

HCI provides generic (in compliance with Bluetooth specification) commands, events signalization, and data and voice transmission understandable by the HCI firmware, and so ensure portability.

Each HCI firmware is specific of its device: it translates former commands into hardware compatible commands to access baseband commands, registers...

Three host controller transport layers have been defined in Bluetooth specification: HCI USB, RS232 and UART transport layers.

### 2.5.2 - Commands

HCI provides different kinds of commands to access Bluetooth hardware capabilities:

- Link control commands (OGF = 0x01): control connections to other BT devices (inquiry, connection, PIN/key request...).
- Link policy commands (OGF = 0x02): how the LM manages the piconet: hold / sniff / park modes, QoS...
- Host Controller and Baseband commands (OGF = 0x03): provide access and control to capability of the BT hardware: reset, flush, setups (name, PIN...)...
- Informational parameters (OGF = 0x04): BD\_ADDR, buffer size...
- Status parameters (OGF = 0x05): current state of the HC, LM and BB.
- Testing commands (0x06).

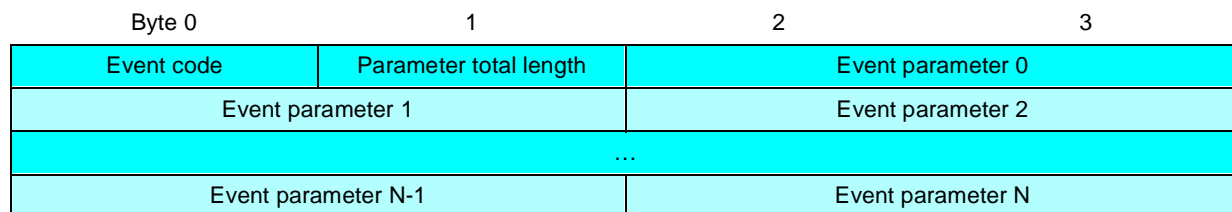
2.5.3 - Packets

– HCI COMMAND PACKET



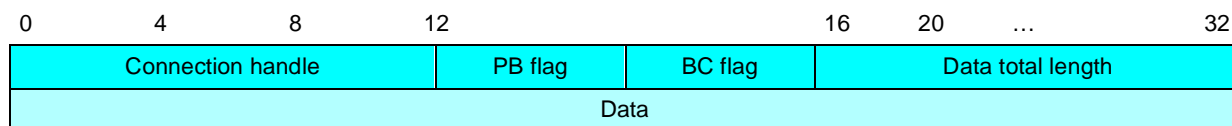
OpCode = OCF + OGF = OpCode Group Field

– HCI EVENT PACKET



– HCI DATA PACKET

**ACL packet**



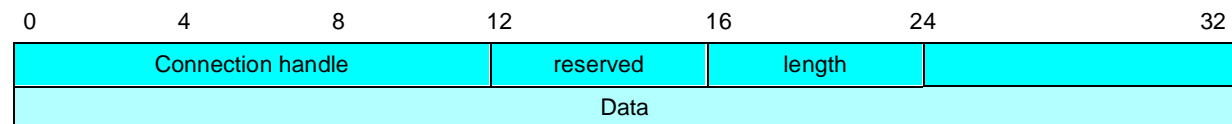
PB flag (packet boundary flag):

- 01 = continuing fragment packet
- 10 = starting fragment packet
- (00 and 11 are reserved for future use)

BC flag (Broadcast flag):

- 00 = no broadcast => point-to-point
- 01 = active broadcast
- 10 = piconet broadcast

**SCO packet**



## 2.6 - L2cap: Logical Link Control & Adaptation Protocol

### L2CAP Provides:

- Protocol multiplexing
- SAR: Segmentation And Reassembly
- QoS: Quality of Service
- Group abstractions

### 2.6.1 - General Operations

- L2CAP only supports **ACL links**
- Thanks to SAR and protocol multiplexing, L2CAP transforms any kind of data from higher layers into packets (up to 64 Kbytes) understandable by the baseband
- L2CAP uses the concept of channels to establish pathways between different applications on BT devices. Channel endpoints are given local ID: CID.
- L2CAP provides two types of channels: connectionless and connection-oriented (different from SCO link) channels. Connection-oriented data channels represent a connection between two devices (so CID identifies each endpoint of this channel), and on the other hand, connectionless channels restrict data flow to one direction, from a master to a “group” of BT slaves. In this case CID represents one or more remote devices.

CID	Description
0x000	Null ID
0x0001	Signaling channel
0x0002	Connectionless reception channel
0x0003-0x003F	Reserved
0x0040-0xFFFF	Dynamically allocated

Each channel is bound to one and only one protocol. Each packet received on a channel is directed toward corresponding higher-level protocol.

### 2.6.2 - Data Packet Format

*Connection-oriented channel:*

LSB	16	16	0/64kbytes
Length		CID	Payload (info)

*Connectionless data channel:*

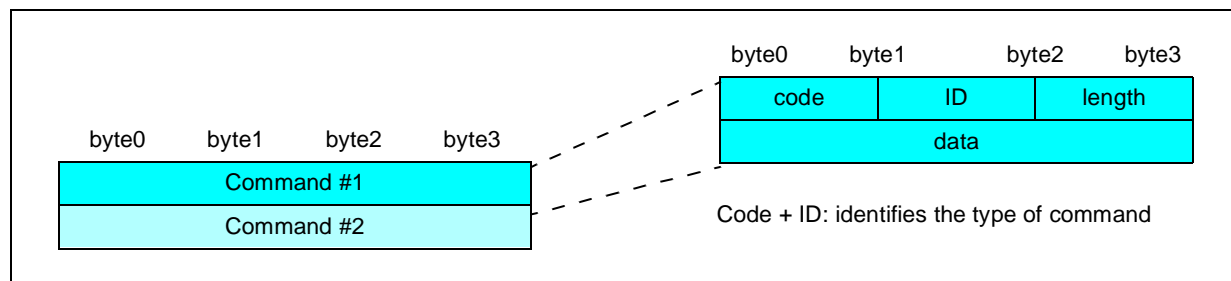
byte0	byte1	byte2	byte3
Length		0x0002	
PSM		information (payload)	
Information			

Length = information length + PSM

PSM: Protocol/Service Multiplexer

### 2.6.3 - Signalling

These signaling commands are passed between two L2CAP entities on remote devices (using CID 0x0001).



Code	Description
0x00	Reserved
0x01	Command reject
0x02	Connection request
0x03	Connection response
0x04	Configure request
0x05	Configure response
0x06	Disconnection request
0x07	Disconnection response
0x08	Echo request
0x09	Echo response
0x0A	Information request
0x0B	Information response

Note that configuration parameters options can also be transmitted as information elements, in order to set parameters such as MTU (maximum transmission unit), flush timeout, QoS,

### 2.6.4 - Primitives

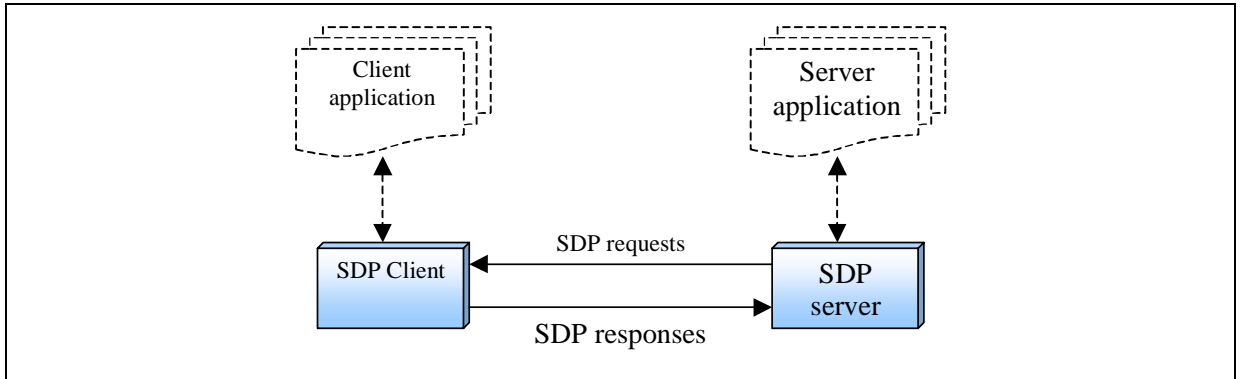
L2CAP possible services are described in terms of service primitives and parameters. The service interface is implementation independent. Primitives are such as: event indication, connect, ping, read...

**2.7 - SDP: Service Discovery Protocol**

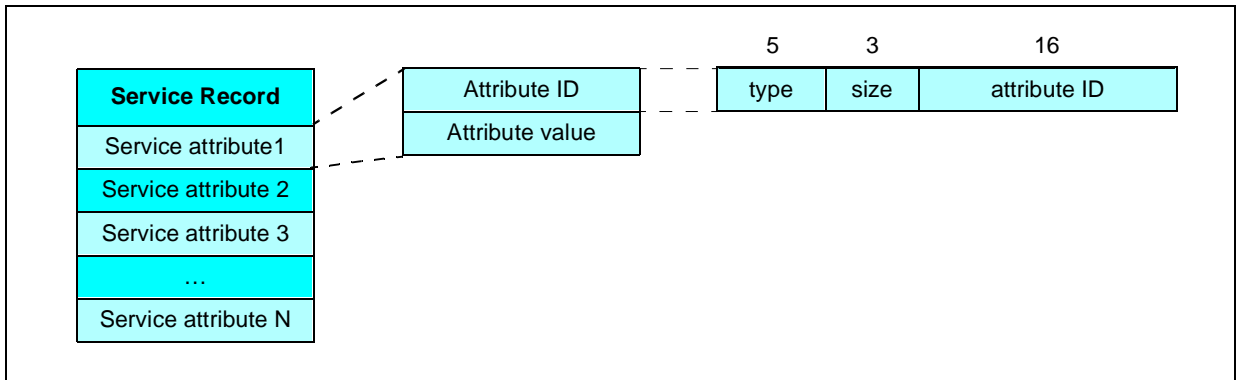
Service Discovery module provides some kind of virtual plug'n'play over the air protocol: it allows discovery of all Bluetooth devices in range and lists their available services characteristics. The set of available services is dynamically updated regarding to RF proximity of the devices.

**SDP Client-Server Functioning**

SDP protocol involves communication between an SDP server and an SDP client. (A single BT device may act both as a server or a client, depending on the overhead application.)



A service is actually any kind (HW or SW or both) of entity, which can provide information, perform an action or control a resource on behalf of another entity. For each service discovered, the SDP server maintain a service record in which all the service description is described and assign an handler ID to it. Service description is composed of service attributes, which all describe a single characteristic of the service, e.g.: service ID, provider name, service name, service class list (services are organized in layers: each service is part of a service class), and so on...



A server can either search for precise services (using Universally Unique ID: UUID) or browse to discover available services.

**2.8 - RFCOMM (Based over ETSI TS GSM 07.10)**

The RFCOMM module provides emulation and multiplexing of serial (RS232C) ports over L2CAP. It supports up to sixty simultaneous connections between two Bluetooth devices (i.e. up to 60 applications can use the same Bluetooth module, for example numerous applications running on the same PC can use a single BT transmitter: one wants to access a printer, another one (or more) a remote keypad...). RFCOMM is intended to cover applications that make use of the serial ports of the devices in which they reside. It emulates all RS232 signals: signal common, TD, RD, RTS, CTS, DSR, DTR, DCD and RI. It also supports multiple emulated serial ports. Each ongoing connection is identified by a DLCI: Data Link Connection Identifier.

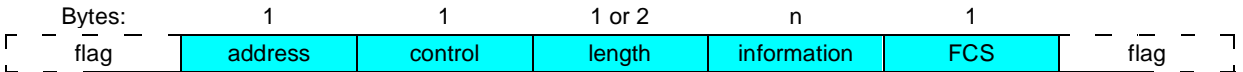
DLCI	Use
0	Control channel
1	/
62-63	Reserved
2-61	Free

**Following frame types from TS 07.10 are supported:**

- SABM : (Set Asynchronous Balanced Mode) command
- UA : (Unnumbered Acknowledgement) response
- DM : (Disconnected Mode) response
- DISC : (Disconnect) command
- UIH : (Unnumbered Information with Header check) command and response.

TS 07.10 following commands are supported (on DLCI 0): Test, Fcon, Fcoff, MSC, RPN, RLS, PN, and NSC.

**TS 07.10 packets adapted for RFCOMM:**



At any time, there must be at most one RFCOMM session between any pair of device.

If a device has to create a new DLC, it must check if an RFCOMM session is active between him and the other target device. If such link exists the device will establish a new DLC through it.

**Start-up procedure:**

- Establish L2CAP channel
- Start RFCOMM multiplexer with SABM (on DLC 0) and wait for UA response.

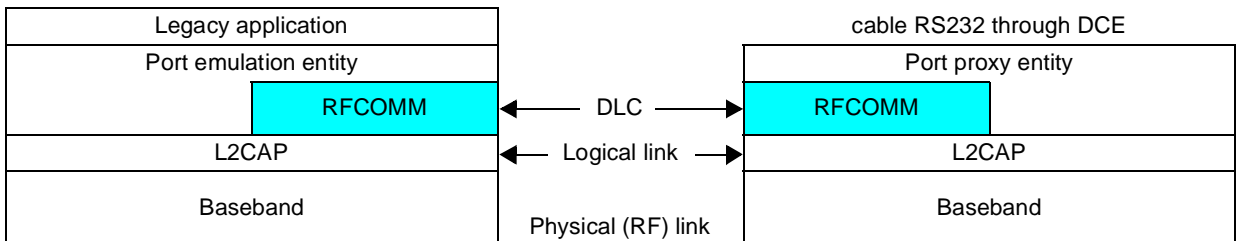
**Flow control:**

L2CAP flow control: provided by the LM.

Wired serial ports flow control: software control with XON/XOFF or circuits RTS/CTS.

RFCOMM: FCON and FCOFF, modem status command

**Interactions between two devices:**

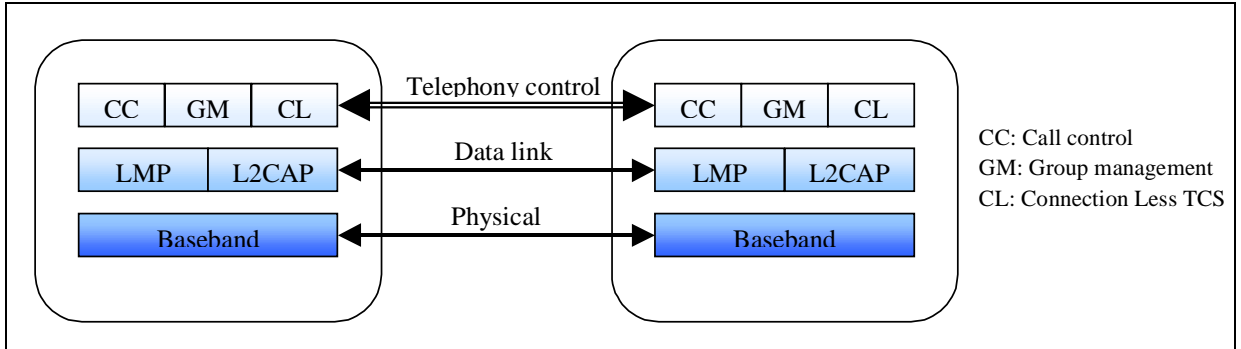


**Connexion steps:**

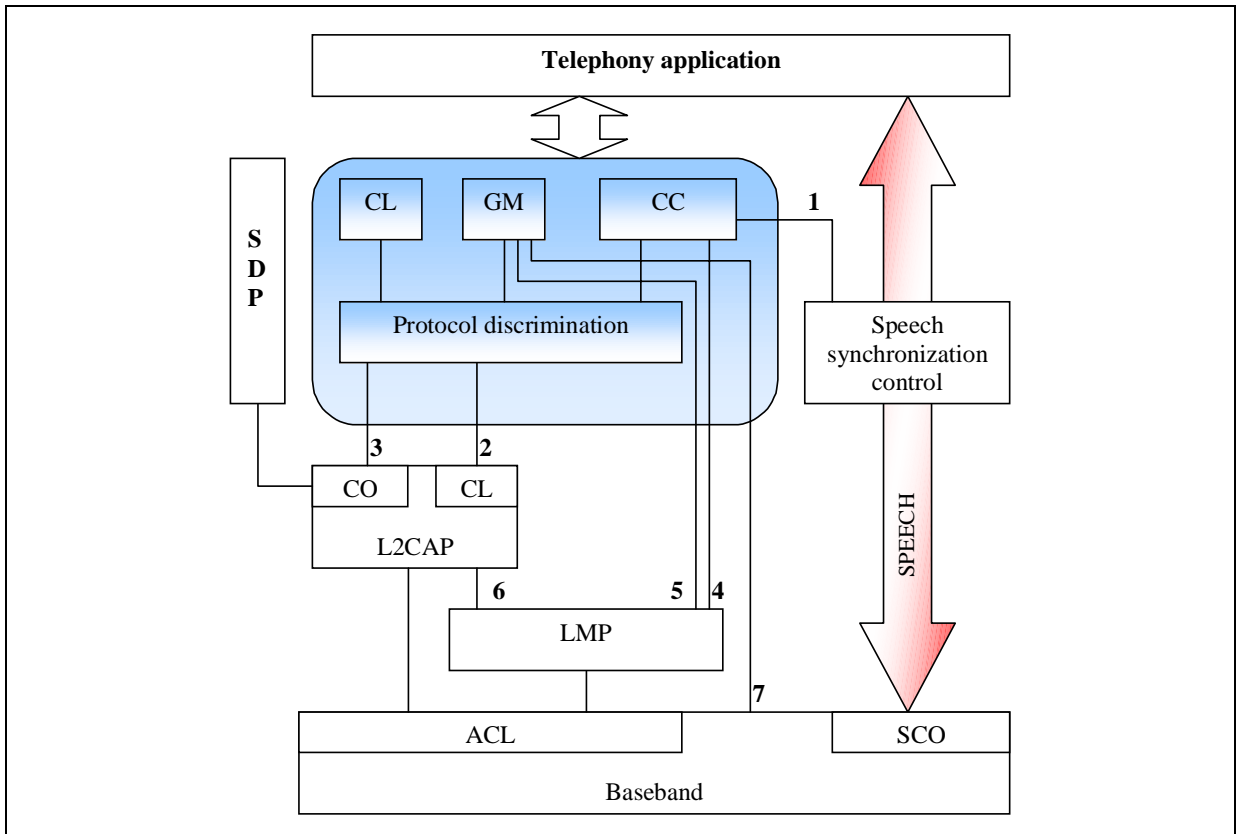
- Create a physical link over the air (RF) between both basebands;
- Open a logical channel between L2CAP layers;
- Attribute a DLC in the RFCOMM multiplexer.

2.9 - Telephony Control Protocol Specification Binary (TCS Bin)

Just like the RFCOMM protocol, TCS bin is an adaptation of an existing protocol: it is based on the ITU-T recommendation Q.931. It deals with telephony application between an “incoming side “ and an “outgoing side”.

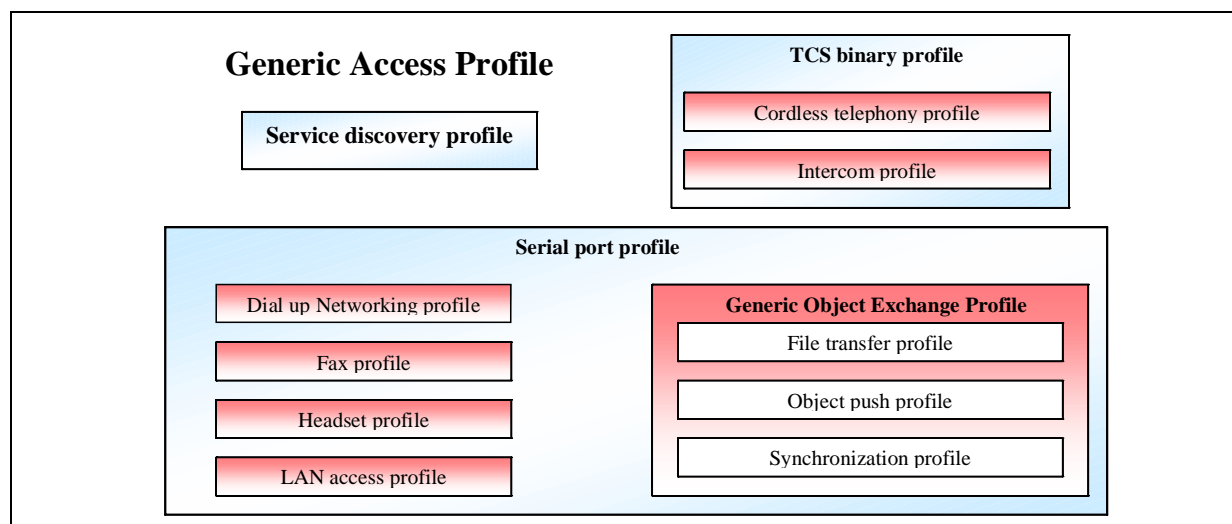


TCS should follow the general architecture described below:



- Notes:
- 1: The Call Control entity provides information to the speech synchronisation control about when to connect.
  - 2: TCS/L2CAP emitting or receiving a SETUP message on the connectionless channel.
  - 3: TCS point-to-point message delivered to L2CAP for transmission on a Connection Oriented channel.
  - 4: The Call control entity controls the LMP directly for the purpose of establishing and releasing SCO links.
  - 5 and 7: The Group Management entity controls the LMP and the Baseband during initialisation procedures.
  - 6: During L2CAP channel establishment, QoS is indicated to the LMP.

### 3 - PROFILES



To ensure compliance, all the applications of Bluetooth devices are officially depicted in the profiles. Profiles are organized in layers, so that profiles using the same low-level protocol can be grouped.

Keep in mind that the SIG profiles are always evolving ( for example, print profile is nowadays debated ).

#### 3.1 - Generic Access Profile (GAP)

This profile defines how two BT devices discover each other and establish a connection. GAP ensures that any two BT units, even from different manufacturers, can exchange information in order to discover what type of applications both units support. A Bluetooth device has four main parameters:

- Its address: BD\_ADDR
- Its name
- Its passkey (Bluetooth PIN)
- Its class of device

The GAP profile specifies the use of these elements through all the steps to authenticate, begin a communication, set security and so on, so that links and channels are established.

All the others profiles rely on this first one: it is the basis of any application.

#### 3.2 - Service Discovery Profile (SDP)

This profile defines precisely how SDP interacts with the other layers to discover and list all available services. Scenarios covered by this profile are the following:

- Search for services by service class
- Search for services by service attributes
- Service browsing

Of course, the Service Discovery Profile relies on the Service Discovery Protocol.

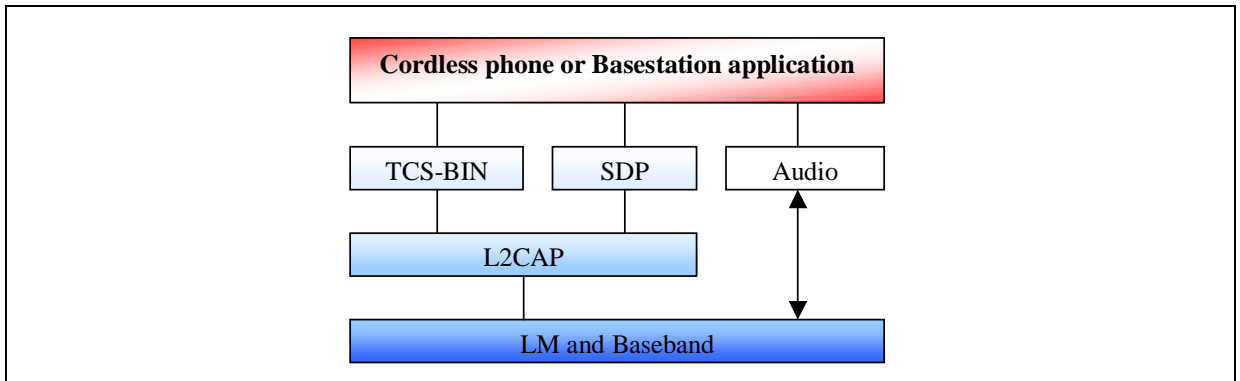
#### 3.3 - Binary TCS Profile

The Cordless Telephony Profile defines the protocol and procedures that shall be used by devices implementing the so called '3-in-1-phone'. Telephone handsets built according to this profile may:

- Act as cordless phones connecting to the public switched telephone network (PSTN) at home or the office. This includes calls via a voice base station, direct calls between two terminals via the base station and accessing additional services provided by an external network.
- Connect directly in full duplex to other telephones in range like italkie-walkie or handset extension. It is referred to as the intercom profile.
- Act as a cellular phone connecting to the cellular infrastructure and incurring cellular charges (and actually, Bluetooth is no more involved).

## TECHNICAL NOTE

The cordless and intercom scenarios use the same protocol stack, which is shown below. The audio stream is directly connected to the baseband protocol.

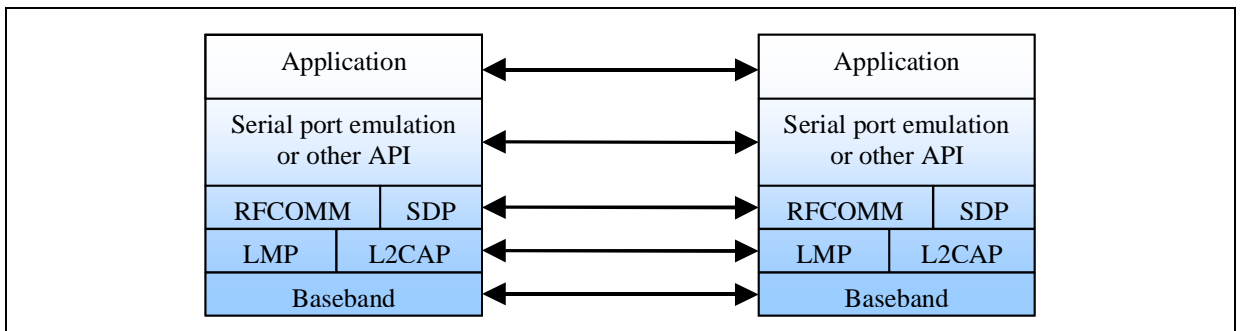


In TCS cordless phone profile, two main roles are defined: Gateway (GW) or Terminal (TL). The cordless telephony profile supports a topology of one gateway and up to 7 terminals.

As the intercom profile is totally symmetric, there are no specific roles defined

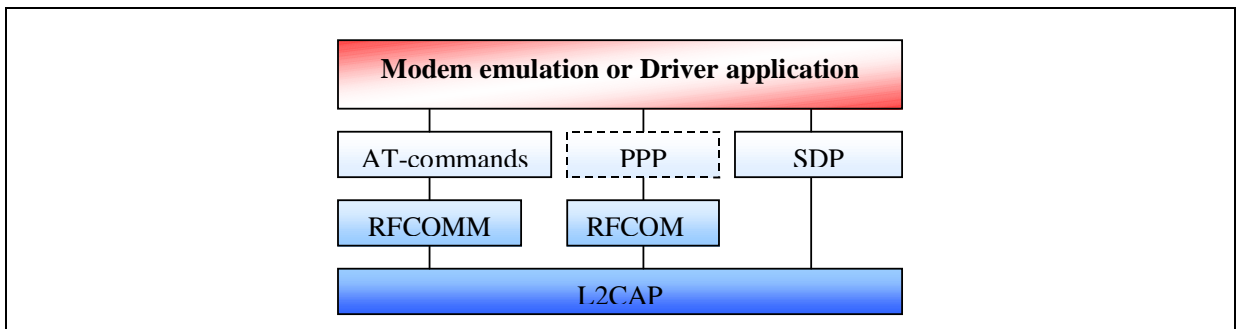
### 3.4 - SERIAL PORT PROFILE

The serial port profile defines the protocols and procedures that shall be used by devices using Bluetooth for RS232 serial cable emulation. So the scenario covered by this profile deals with legacy applications using Bluetooth as a cable substitute, through a virtual serial port.



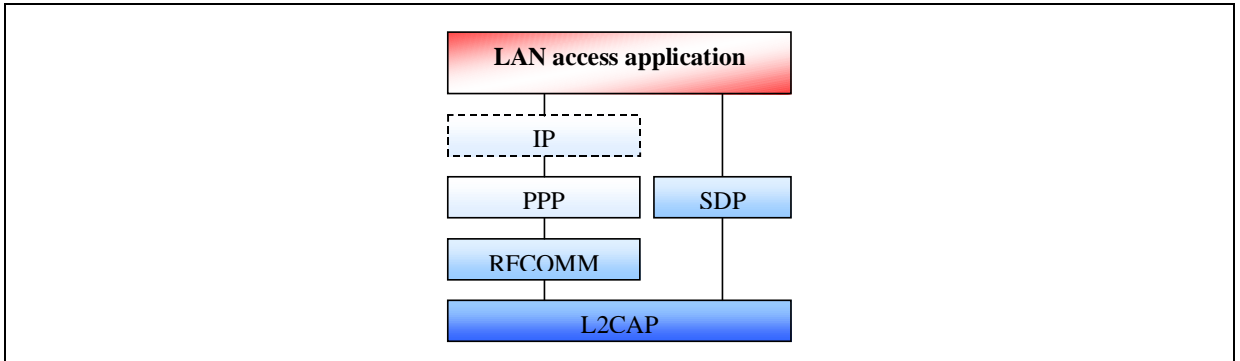
Main applications of the serial port profile are:

- Internet Bridge. In this profile, mobile phone or cordless modem acts as a modem to the PC providing **dial up networking** and **fax** capabilities without need for connection to the PC. In the stack, the AT-commands are needed to control the mobile phone or modem and another protocol (here PPP over RFCOMM) is needed to transfer payload data.

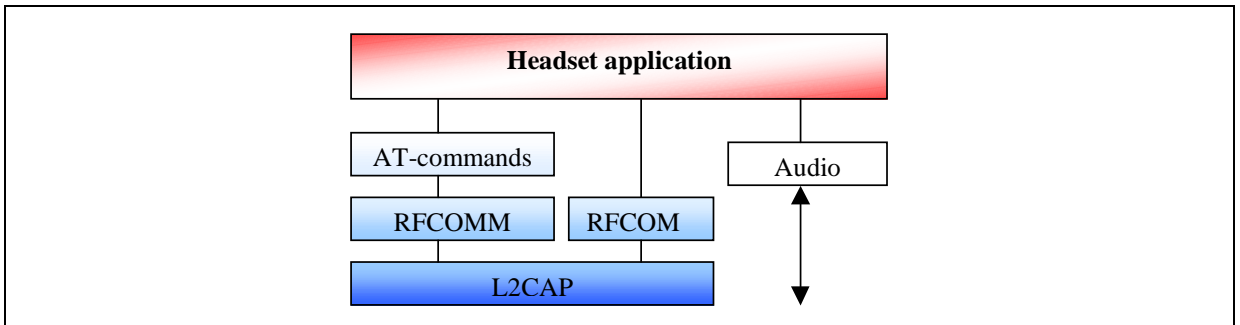


The fax scenario has a similar protocol stack but PPP and the networking protocols above PPP are not used as the application software directly sends fax-simile over RFCOMM.

- **LAN Access.** In this profile, multiple data terminals use a LAN (Local Area Network) access point as a wireless connection to a LAN.

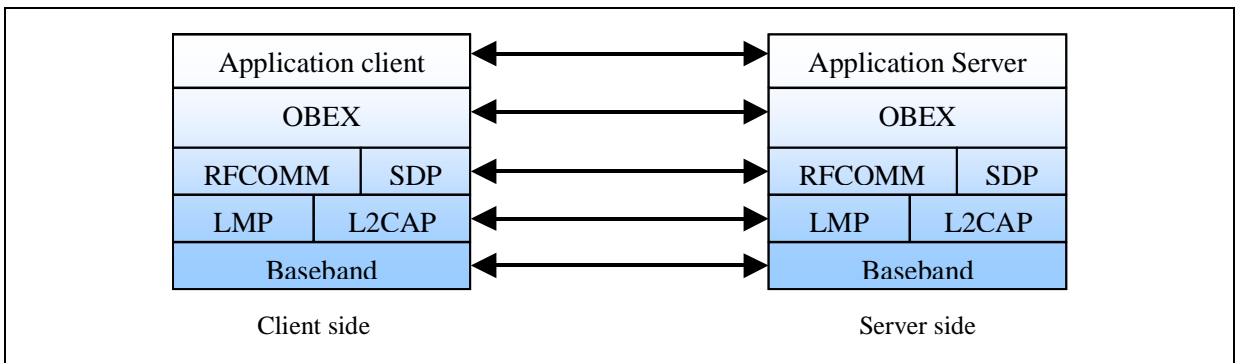


- **Ultimate headset.** The headset can be wirelessly connected for the purpose of acting as a remote device's audio input and output interface. The headset must be able to send AT-commands and receive result codes so that it can answer incoming calls and then terminate without physically manipulating the telephone handset. Audio passes by L2CAP and run directly towards Baseband.



### 3.5 - Generic Object Exchange Profile (GOEP)

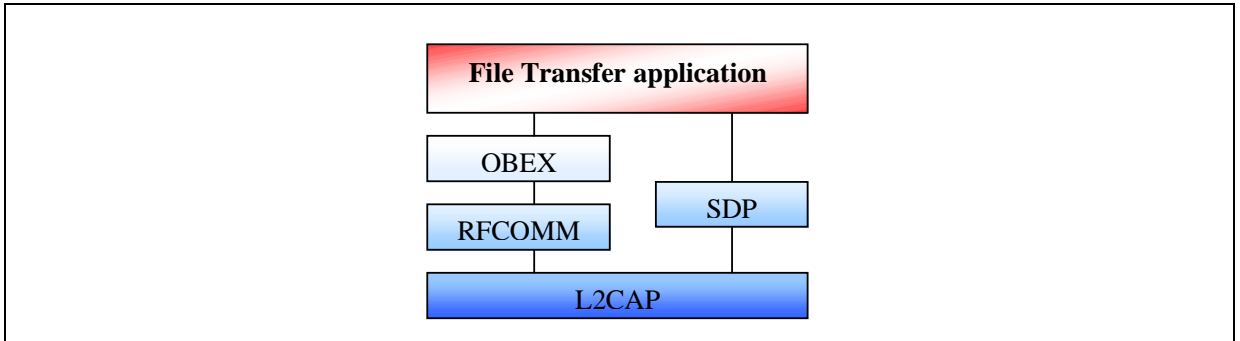
This profile defines the set of protocols and procedures to be used by applications handling object exchanges. Typical Bluetooth units using this profile are notebook PCs, PDAs, mobile phones and smart phones. The GOEP describes the procedures for pushing (or pulling) data from one Bluetooth unit to another. For this profile, devices are either server or client. Client is the device, which can push/pull data objects to/from the server.



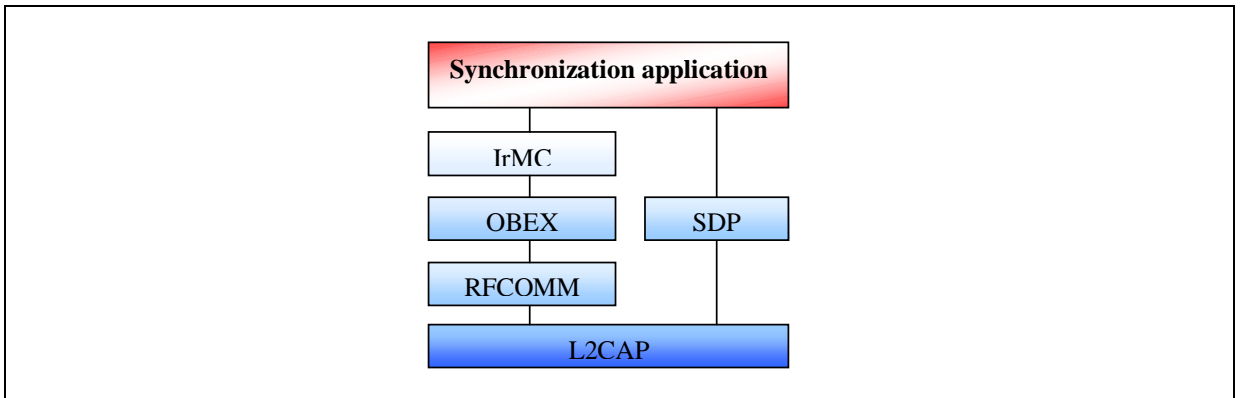
## TECHNICAL NOTE

Within this profiles are specified three scenarios:

- File transfer. The file transfer usage model offers the ability to transfer data objects from one device to another. Object types include (but are not limited to) .doc, .xls, .wav, .jpg files, entire folders or directories or streaming media formats. Also this profile offers the possibility to browse the contents of a remote device.



- Object push profile. This profile covers simple push and exchange operations, e.g. business card exchange (vCard format).
- Synchronization. This provides a device-to-device synchronization of the PIM (personal information management) information, typically phonebook, calendar (vCal), message and note information.



## 4 - ST STRATEGY

ST's offer of the underlying Bluetooth Wireless Technology begins first with the silicon process technology offering. For optimized, integrated RF transceivers, ST proposes the BiCMOS6G process, a 0.35 µm CMOS and bipolar process supporting silicon germanium (SiGe) transistors. These transistors are ideally suited for high frequency (2GHz+), low noise and low power operation. ST's first RF product offering, called Rainbow, is a Blue Rf compliant radio transceiver featuring an advanced architecture with auto calibration, low power consumption and low external component count for small form factor implementations.

One of the keys to the success of Bluetooth Wireless Technology will be the availability of low cost solutions. ST addresses this need with BlueVelvet, its single chip RF and Baseband IC. BlueVelvet is designed on the advanced RFCMOS8 process, a 0.18µm pure CMOS process that allows the design on the same die of high speed, highly integrated digital circuitry and very efficient radio front-end. Interoperability between different Bluetooth core implementation is another key feature of ST solution. To ensure widest possible coverage, ST has licensed Ericsson's Bluetooth Core Technology.

ST is aware that providing chips is only one part of the story, particularly in the RF section, where many customers do not necessarily have RF expertise. ST will provide the complete protocol software as well as advanced module solutions. The pre-certified modules will guarantee interoperability and speed up time-to-market.

According to the most recent data from independent sources, ST is the world's leading supplier of automotive ICs, MPEG-2 decoder ICs, digital Set-Top-Box ICs, and Smart card MCUs, and is also the second leading supplier of analog and mixed-signal ASSPs and ASICs and disk drive ICs. Based on ST's expertise and dominant positions in these segments, new products are currently under development that will incorporate and exploit Bluetooth Wireless technology.

## 5 - WEBSITES OF INTEREST

### Official Site

<http://www.bluetooth.com>

### Presentations and tutorials

<http://www.softtooth.com>

<http://www.infotooth.com>

### Links to articles, white papers...

<http://www.AnywhereYouGo.com/ayg/ayg/bluetooth/Index.po>

### Links to top Bluetooth sites

<http://www.palopt.com.au/bluetooth/index.htm>

<http://new.topsitelists.com/bestsites/bluetooth/topsites.html>

### Ericsson site and presentation

<http://bluetooth.ericsson.se>

<http://www.ericsson.se/microe/bluetooth.html>

### Rohde-Schwarz presentation

[http://www.rohde-schwarz.com/www/dev\\_center.nsf/html/1115216](http://www.rohde-schwarz.com/www/dev_center.nsf/html/1115216)

### Stanford Bluetooth project

<http://bluetooth.stanford.edu>

### About OBEX

<http://www.ravioli.pasta.cs.uit.no/open-obex/index.html>

### 6 - GLOSSARY

#### **3-in-1 Wireless Communication**

Cellular phone systems that provide wireless telephony, Short Message Service, and Dispatch. acceptor. The Bluetooth device receiving an action from another Bluetooth device. The device sending the action is called the initiator. The acceptor is typically part of an established link.

#### **ACK**

Acknowledge.

#### **ACL**

Asynchronous Connectionless Link. An Asynchronous (packet-switched) connection between two devices created on the LMP level. This type of link is used primarily to transmit ACL packet data.

#### **AM\_ADDR**

Active Member Address.

#### **API**

Application Programming Interface. (This applies to any software interface, not just software components deemed application.)

#### **Application Layer**

The group of protocols at the user level. The application layer in the Bluetooth protocol layers will contain those protocols involved with the user interface (UI).

#### **AR\_ADDR**

Access Request Address

#### **ARQ**

Automatic Repeat reQuest.

#### **AT Command Handler**

A module that handles the AT commands which control a phone or modem (between a DTE and a DCE).

#### **ATM**

Asynchronous Transfer Mode.

#### **Authentication**

The process of verifying 'who' is at the other end of the link. Authentication is performed for devices. In Bluetooth, this is achieved by the authentication procedure based on the stored link key or by pairing (entering a PIN).

#### **Authentication Device**

A device whose identity has been verified during the lifetime of the current link based on the authentication procedure.

#### **Authenticate Using a Passkey**

The procedure where a user is requested to enter a passkey during the establishment procedure, where the devices did not share a common link key beforehand. This differs from the bonding procedure where the user enters the passkey without it being requested.

#### **Authorization**

The process of deciding if device X is allowed to have access to service Y. This is where the concept of trusted exists. Trusted devices (the device is authenticated and indicated as "trusted"), are allowed access to services. Untrusted or unknown devices may require authorization based on user interaction before it is allowed access to the services. This does not principally exclude that the authorization might be given by an application automatically. Authorization always includes authentication.

#### **Baseband**

The baseband describes the specifications of the digital signal processing part of the hardware: the Bluetooth link controller, which carries out the baseband protocols and other low-level link routines.

**BB**

See Baseband.

**BD\_ADDR**

Bluetooth Device Address.

**Bluetooth**

An open specification for wireless communication of data and voice. It is based on a low-cost short-range radio link facilitating protected ad hoc connections for stationary and mobile communication environments. Bluetooth clock the master timing mechanism defined by the master of the piconet. Bluetooth device A device that contains hardware and software allowing it to communicate with another Bluetooth device.

**Bluetooth device class**

A parameter that indicates the type of device and which types of services that are supported. The class is received during the discovery procedure. The parameter contains the major and minor device class fields. The term “Bluetooth device class” is used on the UI level.

**Bluetooth Device Name**

The name of the device. (248 bytes maximum)

**Bluetooth Device Type**

The term “Bluetooth device type” is used on the UI level. This term overrides the terms “Bluetooth device class” and “Bluetooth service type” when there is a mix of information containing both Bluetooth Device Class and Bluetooth Service Types.

**Bluetooth Passkey**

The name of the PIN. The term “Bluetooth passkey” is used in the UI. See PIN.

**Bluetooth Service Type**

One or more services a device can provide to other devices. The service information is defined in the service class field of the Bluetooth device class parameter.

**Bluetooth Session**

The activity and participation of a device on a piconet.

**Bond**

A link key that is exchanged between two devices. The key is used for future authentication between the devices. See also bonding.

**Bonding**

Bonding is the creation of a relationship between two devices. The bond is a link key .The relationship is created when the link key is exchanged between two devices. The devices are known to each other prior to the bonding procedure. A user initiates the bonding procedure and enters a passkey with the explicit purpose of creating a bond between two devices. This differs from the authenticate using a passkey procedure where the user is requested to enter a passkey during the establishment of the link.

**Browser**

An application that allows interaction with Internet web pages.

**BT**

Bluetooth.

**Business Card**

The electronic data equivalent to a printed business card. This electronic version of the business card is treated like a file and can be exchanged between Bluetooth devices. See vCard.

**Channel**

A logical connection on L2CAP level between two devices serving a single application or higher layer protocol.

### **Circuit Switched**

The application of a network where a dedicated line is used to transmit information. Only one user may employ the resources of the line at a time.

### **Circuit Switched Bluetooth**

The application of a network where a dedicated line is used to transmit bluetooth data.

### **Circuit Switched Cellular/Radio**

The application of a network where a dedicated line is used to transmit cellular/radio data.

### **CL**

Connectionless.

### **Class of device**

See Bluetooth device class. Also abbreviated as CoD.

### **CO**

Connection-oriented.

### **CODEC**

Coder/Decoder. A device that converts analog to digital, and digital to analog for transmission over a digital communications system.

### **Component**

An architecture element denoting an identifiable set of software that performs a well-defined purpose.

### **Connect to Service**

The establishment of a connection to a service. If not already done, this includes establishment of a physical link, link and channel as well.

### **Connectable Devices**

Any device within range that will respond to paging from an initiator device.

### **Connectable Mode**

A device that responds to paging (an attempt to establish a communication link) is said to be in connectable mode. The opposite of connectable mode is non-connectable mode.

### **Connected Device**

A device that is currently connected to the LocDev.

### **Connection**

A connection between two peer applications or higher layer protocols mapped onto a channel.

### **Connectionless Packet**

A packet of data is broadcast over the network without targeting a specific recipient to receive the packet.

### **Connecting**

A phase in the communication between devices when a connection between them is being established. (Connecting phase follows after the link establishment phase is completed.)

### **Connectivity**

A domain of interconnected components that adhere to a defined set of connection rules. The set of rules is termed Connectivity Architecture.

### **CRC**

Cyclic Redundancy Check.

### **CTP**

Cordless Telephone Profile.

### **CVSD**

Continuous Variable Slope Delta Modulation.

**DAC**

Device Access Code.

**DCE**

Data Circuit-Terminating Equipment. In serial communications, DCE refers to a device between the communication endpoints whose sole task is to facilitate the communications process; typically a modem.

**Device Discovery**

The mechanism to request and receive the Bluetooth address, clock, and class of device, used page scan mode, and names of devices.

**Device Layer**

The group of protocols that handles the hardware in a Bluetooth device. The device layer handles components such as the display, keypad, and RF communications.

**Device Name**

See Bluetooth device name.

**Device Security Level**

Access to a device can be denied based on the required device security level. There are two levels of device security: trusted device and untrusted device. See also service security level.

**DH**

Data-High Rate. Data packet type for high rate data.

**DIAC**

Dedicated inquire access code.

**Discoverable Device**

A Bluetooth device in range that will respond to an inquiry (normally in addition to responding to page).

**Discoverable Mode**

A device that can respond to an inquiry is said to be in a discoverable mode. There are two types of discoverable modes: limited discoverable mode and general discoverable mode. The opposite of discoverable mode is non-discoverable mode. See also silent device.

**Dispatch**

Walkie-talkie mode where one subscriber talks and other subscribers listen on the same talk group.

**DLCI**

Data Link Connection Identifier.

**DM**

Data - Medium Rate. Data packet type for medium rate data.

**DSR**

Data Set Ready. A device sets an RS-232 DSR signal when it is ready to accept data.

**DTE**

Data Terminal Equipment. In serial communications, DTE refers to a device at the endpoint of the communications path; typically a computer or terminal.

**DTMF**

Dual Tone Multiple Frequency. Dumb peripheral A peripheral that does not communicate any information to the handset. Typically, the only information the handset receives from a dumb peripheral is a signal that a connection has been made to a port on the handset. This signal is also called a cable detect.

**DV**

Data Voice. Data packet type for data and voice.

**ETSI**

European Telecommunications Standards Institute.

**FEC**

Forward Error Correction.

**FH**

Frequency Hopping.

**FHS**

Frequency Hopping Synchronization.

**FIFO**

First In First Out.

**FSK**

Frequency Shift Keying. A type of modulation.

**GAP**

Generic Access Profile. This profile describes the mechanism by which one device discovers and accesses another device when they do not share a common application.

**GFSK**

Gaussian Frequency Shift Keying.

**GIAC**

General Inquire Access Code. See also general discoverable mode.

General discoverable mode

A device that can be discovered continuously or for no specific condition is said to be in general discoverable mode. See also discoverable mode.

**GM**

Group Management.

**GOEP**

Generic Object Exchange Profile.

**GSM**

Global System for Mobile communications. GSM is a digital cellular communications technology that is available in Europe and the US. GSM offers multiple services for the subscriber such as short message service.

**GW**

Gateway. A Bluetooth enabled base station, which is connected to external network.

**HA**

Host Application. A software program that uses Bluetooth.

**HCI**

Host Controller Interface.

**Headset**

A microphone and earpiece used to conduct conversations. Headsets can be connected directly to a cellular device or remotely using Bluetooth communications technology.

**HEC**

Header-Error-Check.

**Host**

A software and hardware platform in which the Bluetooth package runs.

**HV**

High quality Voice. (e.g., HV1 packet.)

**HW**

Hardware.

**IAC**

Inquiry Access Code.

**Idle mode**

A device is in idle mode when it has no established links to other devices. In this mode, the device may discover other devices. In general, a device sends inquiry codes (GIAC, LIAC) to other devices. Any device that allows inquiries will respond with information. If the devices decide to form a link, then (bonding will occur.

**IEEE**

Institute of Electronic and Electrical Engineering.

**IETF**

Internet Engineering Task Force.

**Initiator**

The Bluetooth device initiating an action to another Bluetooth device. The device receiving the action is called the acceptor. The initiator is typically part of an established link.

**Inquiry Procedure**

The inquiry procedure enables a device to discover which devices are in range, and determine the addresses and clocks for the devices. After the inquiry procedure has completed, a connection can be established using the paging procedure.

**Inquiry State**

A mode that a LocDev enters when searching for services.

**Inquiry Scan State**

A mode that a RemDev enters when advertising that a service is available.

**Intelligent Peripheral**

A peripheral that is capable of exchanging information with the handset. Information may include battery status, charging status, data storage status, or other high-level functionality. Also referred to as a smart peripheral.

**Internet bridge**

Method of using a wireless modem for connecting to Internet access.

**IP**

Internet Protocol.

**IPX**

Internetwork Packet eXchange. Novell's protocol used by Netware. A router with IPX routing can interconnect LANs so that Novell Netware clients and servers can communicate.

**IrDA**

Infrared Data Association. A method for communication between electronic devices, using 880-nm infrared light.

**ISDN**

Integrated Services Digital Networks.

**ISM**

Industrial, Scientific, Medical.

**ITU**

International Telecommunication Union

**Key Management**

The handling and control of encryption keys.

**Known device**

A device for which at least the BD\_ADDR is stored.

### **L2CA**

Logical Link Controller and Adaptation.

### **L2CAP**

Logical Link Controller and Adaptation Protocol.

### **L\_CH**

Logical Channel.

### **LAN**

Local Area Network.

### **LAP**

LAN Access Point.

### **LAP**

Lower Address Part.

### **Link**

Shorthand for an ACL link.

### **LC**

Link Controller. The Link Controller manages the link to the other Bluetooth devices. It is the low-level baseband protocol handler.

### **LCP**

Link Controller Protocol.

### **LFSR**

Linear Feedback Shift Register.

### **LIAC**

Limited Inquiry Access Code. See also limited discoverable mode.

### **Limited discoverable mode**

A device that responds to an inquiry for limited purposes. For example, a device may respond for a limited period of time, during temporary conditions, or for a specific event. Typically, the device is responding to a limited inquiry based on an inquiry using the LIAC. See also discoverable mode.

### **Link key**

The authentication key used to establish a link between devices. See also bonding.

### **LM**

Link Manager. The Link Manager software entity carries out link setup, authentication, link configuration, and other protocols.

### **LMP**

Link Manager Protocol. The LMP is used for peer-to-peer communication.

### **LMP-Authentication**

An LMP level procedure for verifying the identity of a remote device. The procedure is based on a challenge-response mechanism using a random number, a secret key and the BD\_ADDR of the non-initiating device. The secret key used can be a previously exchanged link key or an initialization key created based on a PIN (as used when pairing).

### **LMP-Pairing**

A LMP procedure that authenticates two devices based on a PIN and subsequently creates a common link key that can be used as a basis for a trusted relationship or a (single) secure connection. The procedure consists of the steps: creation of an initialization key (based on a random number and a PIN), Lmp-authentication based on the initialization key and creation of a common link key.

**LocDev**

Local Device. A Bluetooth device, which initiates a SDP procedure. A Local Device is typically a master device on the piconet. However, a Local Device may not always have a master connection relationship to other devices. See also RemDev.

**LSB**

Least Significant Bit.

**M\_ADDR**

Medium Access Control Address.

**Master Device**

A device that initiates an action or requests a service on a piconet. See also LocDev.

**MS**

Multiplexing sub layer.

**MSB**

Most Significant Bit.

**MSC**

Modem Status Command.

**MTU**

Maximum Transmission Unit.

**MUX**

Multiplexer. A device that combines one or more data signals into a single composite signal for communication over one data channel.

**MUX**

Multiplexing Sub layer. A sub layer of the L2CAP layer.

**N/A**

Not applicable.

**NAK**

Negative Acknowledge.

**Name Discovery**

The mechanism to request and receive a device name.

**NAP**

Non-significant Address Part.

**NDIS**

Network Driver Interface Specification.

**New Device**

See unknown device.

**Non-Connectable Mode**

A device that does not respond to paging (an attempt to establish a communication link) is said to be in non-connectable mode. The opposite of non-connectable mode is connectable mode.

**Non-Discoverable Mode**

A device that cannot respond to an inquiry is said to be in non-discoverable mode. The device will not enter the INQUIRY\_RESPONSE state in this mode. See also discoverable mode.

**Non-Pairable Mode**

A device that does not accept pairing is said to be in non-pairable mode. The opposite of non-pairing mode is pairable mode.

### **OBEX**

Object EXchange Protocol.

### **OEM**

Original Equipment Manufacturer.

### **OS**

Operating System.

### **Packet Switched**

A network that routes data packets based on an address contained in the data packet is said to be a packet switched network. Multiple data packets can share the same network resources.

### **Packet Switched Bluetooth**

The application of routing bluetooth data packets on a network using addresses contained in the bluetooth data packets.

### **Packet Switched Cellular/Radio**

The application of routing cellular/radio data packets on a network using addresses contained in the cellular/radio data packets.

### **Page**

A baseband state where a device transmits page trains and processes any eventual responses to the page trains.

### **Page Scan State**

A mode where a device listens for page trains containing its own device access code (DAC). A mode that a RemDev enters when advertising that a service is available.

### **Page State**

A mode that a LocDev enters when searching for services. The LocDev sends out a page to notify other devices that it wants to know about the other devices and/or their services.

### **Page train**

A series of paging messages sent over the baseband.

### **Paged device**

A paged device is typically contacted by a paging device to establish a communication link. See acceptor.

### **Paging**

The act of attempting to establish a communication link.

### **Paging device**

A paging device is typically attempting to establish a communication link with other devices. See initiator.

### **Paging Procedure**

With the paging procedure, an actual connection can be established. The paging procedure typically follows the inquiry procedure. Only the Bluetooth device address is required to set up a connection. Knowledge about the clock will accelerate the setup procedure. A unit that establishes a connection will carry out a page procedure and will automatically be the master of the connection.

### **Pairable mode**

A device that accepts pairing is said to be in pairable mode. The opposite of pairing mode is non-pairable mode. paired device A device with which a link key has been exchanged (either before connection establishment was requested or during connecting phase). See also pre-paired device and un-paired device.

### **Pairing**

The creation and exchange of a link key between two devices. The devices (LocDev and RemDev) use the link key for future authentication when exchanging information. Pairing is also called an association between a LocDev and a RemDev based on a common link key. The link key is also referred to as a bond. Pairing can also establish a link by the user entering a PIN, which is authenticated by the device providing the service.

**Parked Unit(s)**

Devices in a piconet, which are synchronized but do not have a MAC addresses.

**PC**

Personal Computer.

**PC Card**

A hardware device that is attached to or installed in a PC to enable the PC to communicate with other Bluetooth devices.

**PCM**

Pulse Coded Modulation.

**PCMCIA**

Personal Computer Memory Card International Association.

**PDA**

Personal Digital Assistant.

**PDU**

Protocol Data Unit. (i.e., a message.)

**Phone Services Database**

The portion of the BT implementation that stores information about device services, both local services and remote services.

**Physical channel**

A synchronized Bluetooth baseband-compliant RF hopping sequence.

**Physical link**

A Baseband level association between two devices established using paging. A physical link comprises a sequence of transmission slots on a physical channel alternating between master and slave transmission slots.

**Piconet**

A collection of devices connected via Bluetooth technology in an ad hoc fashion. A piconet starts with two connected devices, such as a portable PC and cellular phone, and may grow to eight connected devices. All Bluetooth devices are peer units and have identical implementations. However, when establishing a piconet, one unit will act as a master and the other(s) as slave(s) for the duration of the piconet connection. All devices have the same physical channel defined by the master device parameters (clock and BD\_ADDR).

**PIN**

Personal Identification Number. The Bluetooth PIN is used to authenticate two devices that have not previously exchanged link key. By exchanging a PIN, the devices create a trusted relationship. The PIN is used in the pairing procedure to generate the initial link that is used for further identification.

**PIN (BB)**

The PIN used on the baseband level. The PIN (BB) is used by the baseband mechanism for calculating the initialization key during the pairing procedure. (128 bits)

**PIN (UI)**

The PIN used on the user interface level. The PIN (UI) is the character representation of the PIN that is entered on the UI level.

**PM\_ADDR**

Parked Member Address.

**PnP**

Plug and Play.

**PPP**

Point-to-Point Protocol.

### **Profile**

A description of the operation of a device or application.

### **PSM**

Protocole/Service Multiplexer.

### **PSTN**

Public switched telephone network.

### **QoS**

Quality of Service.

### **RAND**

Random number.

### **RemDev**

Remote Device. A Bluetooth device that participates in the SDP process. A Remote Device must contain a SDP server along with a service record database. A Remote Device is typically a slave device, however, a Remote Device may not always have a slave connection with a LocDev. requestor An entity that requests information from another entity via the Bluetooth API.

### **RF**

Radio Frequency.

### **RFCOMM**

Serial Cable Emulation Protocol based on ETSI TS 07.10.

### **RS-232**

A serial communications interface. Serial communication standards are defined by the Electronic Industries Association (EIA).

### **RTOS**

Real Time Operating System.

### **RX**

Receiver.

### **SAR**

Segmentation and Reassembly.

### **Scatternet**

Multiple independent and non-synchronized piconets form a scatternet.

### **SCO**

Synchronous Connection Oriented link. A synchronous (circuit-switched) connection for reserved bandwidth communications, e.g. voice, between two devices created on the LMP level by reserving slots periodically on a physical channel. This type of link is used primarily to transport SCO packets (voice data). Supports time-bounded information like voice. (Master to single slave.) SCO links can be established only after an ACL link has first been established.

### **SD**

Service Discovery.

### **SDA**

Service Discovery Application. Also sometimes called the Service Discovery User Application.

### **SDAP**

Service Discovery Application Profile.

### **SDP**

Service Discovery Protocol.

**SDP client**

The SDP in a Local Device (LocDev). The SDP client requests service information from SDP servers.

**SDP server**

The SDP in a Remote Device (RemDev). The SDP server responds to requests made by SDP clients.

**SDP Session**

The exchange of information between an SDP client and an SDP server. The exchange of information is referred to as an SDP transaction.

**SDP Transaction**

The exchange of an SDP request from an SDP client to an SDP server, and the corresponding SDP response from an SDP server back to the SDP client.

**Security Manager**

The module in a Bluetooth device that controls security aspects of communications to other Bluetooth devices.

**Security Mode 1**

A device will not initiate any security. A non-secure mode.

**Security Mode 2**

A device does not initiate security procedures before channel establishment on L2CAP level. This mode allows different and flexible access policies for applications, especially running applications with different security requirements in parallel. A service level enforced security mode.

**Security Mode 3**

A device initiates security procedures before the link setup on LMP level is completed. A link level enforced security mode.

**SEQN**

Sequential Numbering scheme.

**SerDscApp**

Service Discovery Application.

**Serial Interface**

An interface to provide serial communications. service This term refers to a service that one device provides for others. Examples are printers, PIM. Synchronization servers, modems (or modem emulators).

**Service Discovery**

See SDP.

**Service Layer**

The group of protocols that provides services to the application layer and the driver layer in a Bluetooth device.

**Service Record Database**

A database that contains the service discovery-related information.

**Service security level**

Access to services can be denied based on the required service security level. There are three levels of service security: authorization and authentication; authentication only, and no security (open to all). Encryption can be another security requirement for service use in addition to the requirements listed above. Encryption is typically applied at the physical level (baseband). See also device security level.

**SIG**

Special Interest Group.

### **Silent device**

A device that is in discoverable mode but cannot respond due to other baseband activity is said to be a silent device. The device could also be in non-discoverable mode and would also not respond to an inquiry.

### **SIM**

Subscriber Identity Module. The SIM is a non-volatile storage device that contains information about your phone. This allows the SIM to be used in any GSM phone.

### **Slave Unit**

All devices in a piconet that are not the master.

### **Smart peripheral**

See intelligent peripheral.

### **SP**

Service Provider.

### **SrvDscApp**

Service Discovery Application.

### **SSI**

Signal Strength Indication.

### **SW**

Software.

### **TBD**

To Be Defined.

### **TCP**

Transport Control Protocol.

### **TCP/IP**

Transport Control Protocol/Internet Protocol.

### **TCS**

Telephone Control protocol Specification.

### **TCS-AT**

A set of AT-commands by which a mobile phone and modem can be controlled in the multiple usage models. In BT, AT-commands are based on ITU-T recommendation v.250 and ETS 300 916(GSM 07.07). In addition, the commands used for fax services are specified by the implementation. TCS-AT will also be used for dial-up networking and headset profiles.

### **TCS Binary**

Bluetooth Telephony Control protocol Specification using bit-Oriented protocol. It is also referred to as the TCS-BIN system. TCS-BIN will be used for cordless telephony profiles.

### **TDD**

Time Division Duplex

### **TL**

Terminal.

### **TLO**

Terminal Originating a Call.

### **TLT**

Terminal terminating a call.

**TTP**

Tiny Transport Protocol between OBEX and UDP [TBD].

**TX**

Transmit.

**UA**

User Asynchronous. Asynchronous user data.

**UAP**

Upper Address Part.

**UART**

Universal Asynchronous Receiver Transmitter. A device which converts parallel data into serial data for transmission, or it converts serial data into parallel data for receiving data.

**UDP**

User Datagram Protocol.

**UDP/IP**

User Datagram Protocol/Internet Protocol.

**UI**

User Interface. The area on a device that contains interface mechanisms such as displays, dialog boxes, manuals, packaging, advertising, etc., where the user is likely to encounter Bluetooth terminology and parameters.

**UIAC**

Unlimited Inquiry Access Code.

**Unknown device**

A device that is currently not connected with the (LocDev and the LocDev has not paired with it in the past. Also called a new device. No information about the device is stored (e.g., BD\_ADDR, link key, or other information).

**UUID**

Universal Unique ID.

**7 - REVISION HISTORY**

Date	Comments
August 8 <sup>th</sup> 2000	First version.
August 10 <sup>th</sup> 2000	Added Bluetooth Radio properties.
September 5 <sup>th</sup> 2000	Added summary on software protocol. Language corrections.
September 11 <sup>th</sup> , 2000	Last corrections and new schema about connections steps

Information furnished is believed to be accurate and reliable. However, STMicroelectronics assumes no responsibility for the consequences of use of such information nor for any infringement of patents or other rights of third parties which may result from its use. No license is granted by implication or otherwise under any patent or patent rights of STMicroelectronics. Specifications mentioned in this publication are subject to change without notice. This publication supersedes and replaces all information previously supplied. STMicroelectronics products are not authorized for use as critical components in life support devices or systems without express written approval of STMicroelectronics.

The ST logo is a registered trademark of STMicroelectronics

© 2001 STMicroelectronics - All Rights Reserved

STMicroelectronics GROUP OF COMPANIES

Australia - Brazil - China - Finland - France - Germany - Hong Kong - India - Italy - Japan - Malaysia - Malta - Morocco  
Singapore - Spain - Sweden - Switzerland - United Kingdom - U.S.A.

<http://www.st.com>