Bluetooth

Bluetooth is an open wireless protocol stack for low-power, short-range wireless data communications between fixed and mobile devices, and can be used to create *Personal Area Networks* (PANs). It was conceived as a wireless alternative to the use of serial data cables, but has the advantage of being able to connect a number of devices simultaneously. Bluetooth uses *frequency-hopping spread spectrum* radio technology, which transmits each block of data at a different frequency. The basic modulation scheme used is *Gaussian frequency-shift keying* (GFSK), with a nominal maximum data rate of 1 Mbps. Bluetooth has found a diverse range of applications, including the exchange of data between a computer and a mobile phone, and between a computer and various forms of wireless peripheral devices, including mice, keyboards and printers. The frequency band used for Bluetooth is the *Industrial, Scientific and Medical* (ISM) 2.4 GHz frequency band, which does not require a license.

Bluetooth was first developed in 1994 by the Swedish company *Ericsson Mobile Platforms*, and the Bluetooth specifications are now developed and maintained by the *Bluetooth Special Interest Group* (SIG), which consists of companies involved in telecommunication, computing, networking, and consumer electronics including *Ericsson*, *IBM*, *Intel*, *Toshiba* and *Nokia*. Bluetooth devices can advertise all of the services they provide, and a computer can communicate with a Bluetooth device such as a mobile telephone or a wireless peripheral providing it has a Bluetooth adapter. Some desktop computers and most recent-model laptop computers come with a Bluetooth adapter built-in. Others will need to be equipped with an external adapter in the form of a dongle. A single Bluetooth adapter will allow a computer to communicate with multiple Bluetooth devices. Most current operating systems support Bluetooth, including Mac OS X, MS Windows (XP Service Pack 2 and later) and most recent distributions of Linux.

Early versions of the Bluetooth specification suffered from interoperability problems, but the specification has continued to evolve and improvements have included shorter connection and discovery times, higher data rates, more efficient use of power, and better resistance to interference. The most recent version at the time of writing is version 3.0, introduced in 2009, which can take advantage of IEEE 802.11 wireless protocols for data transfer, although Bluetooth radio is still used for device discovery, and for connection setup and configuration.

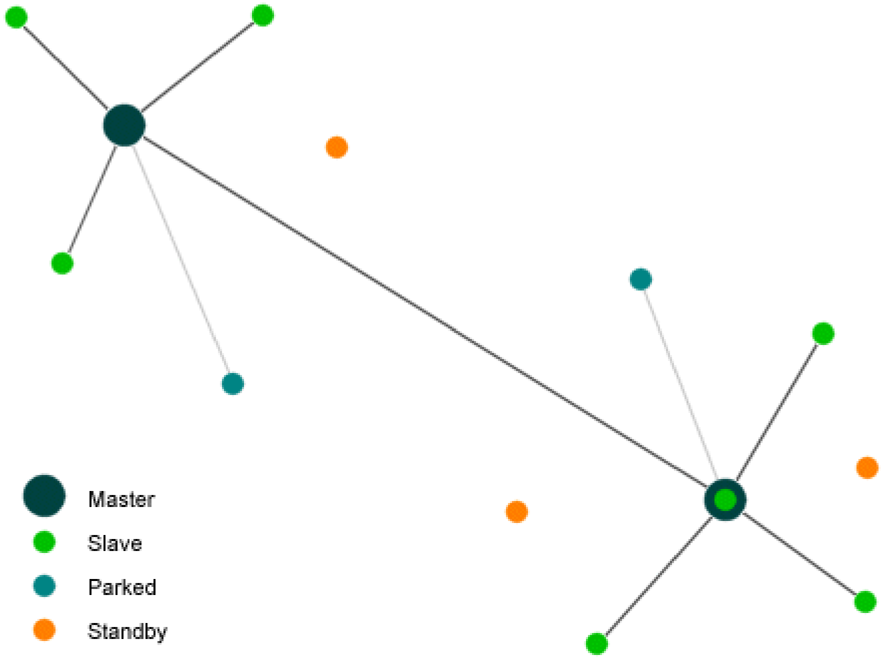
Bluetooth Radio

Bluetooth Radio is the lowest layer defined in the Bluetooth specification. Bluetooth uses frequencies between 2.4000 and 2.4835 GHz, and divides the band into 79 1-MHz channels (numbered 0-78), with frequency hopping occurring at a rate of 1600 times per second. Channel 0 has a frequency centred at 2.4020 GHz, allowing a lower guard band of 2 MHz. Channel 78 has a frequency centred at 2.4800 GHz, allowing an upper guard band of 3.5 MHz. Bluetooth devices are divided into three classes, depending on their maximum transmitted power (and hence their maximum range):

- ***Power Class 1*** - long range (in the order of 100 metres)
- ***Power Class 2*** - medium range (in the order of 10 metres)
- ***Power Class 3*** - short range (in the order of 10 centimetres)

A device may optionally vary its transmitted power according to the requirements of the receiving device, allowing it to conserve battery power and reduce the likelihood of signals interfering with devices that are not participating in the link. Essentially, the receiver measures the signal strength of the incoming signal, and sends a request to the transmitting device to either increase or decrease its transmitted power.

Up to eight Bluetooth devices may participate in a link simultaneously, forming a *piconet*. At any given time, one of the devices in the link acts as the master device, and may exchange data with one other (slave) device. The device acting as the master can exchange roles with a slave device frequently and at any time, whereby the slave becomes the new master device and the master becomes a slave. Networks of piconets can also be created by joining two or more piconets together in a *scatternet*. One of the devices becomes a bridge between the two piconets, acting as a slave in one piconet and a master in the other. Bluetooth connections are set up automatically when two Bluetooth-enabled devices come within range of each other. If the devices determine that they have data to share, or that one device needs to control the other, they form a piconet.



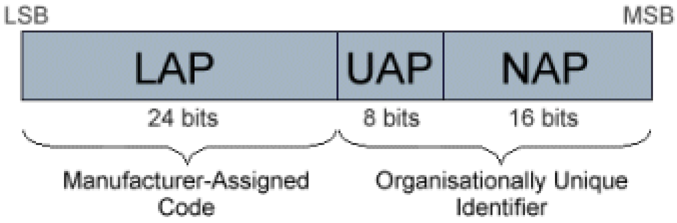
A Bluetooth scatternet consisting of two piconets

Bluetooth Addressing

The 48-bit Bluetooth Device Address (BD_ADDR) is essentially a globally unique MAC address assigned to each Bluetooth adapter by the device manufacturer, and can be divided into three fields as follows:

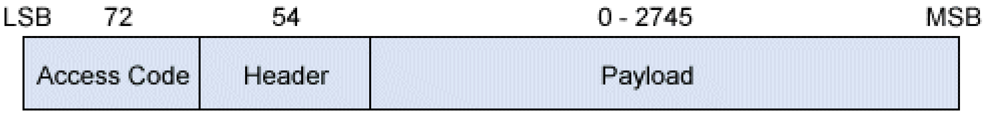
- ***Lower Address Part (LAP)*** - the 24-bit portion of the MAC address that is allocated by the manufacturer, and forms part of the *Access Code*that precedes the Bluetooth baseband header in transmitted packets.
- ***Upper Address Part (UAP)*** - an 8-bit part of the 24-bit *Organisationally Unique Identifier* (OUI) portion of the MAC address allocated to manufacturers by IEEE. The UAP is used to (among other things) generate the *Header Error Correct* (HEC) field used to detect errors in Bluetooth packets.
- ***Non-significant Address Part (NAP)*** - the remaining 16 bits of the OUI. The NAP is not particularly significant for Bluetooth networking.

The format of the Bluetooth Device Address is illustrated below.


LSB
MSB
LAP
UAP
NAP
24 bits
8 bits
16 bits
Manufacturer-Assigned Code
Organisationally Unique Identifier

The format of the Bluetooth Device Address

All data on a Bluetooth piconet channel is transmitted as packets. The packet format is shown below.



The Bluetooth packet format

The *Access Code* is used for timing synchronisation, paging and inquiry, and can take three forms. The *Channel Access Code* (CAC) uniquely identifies a piconet. The *Device Access Code* (DAC) is used for paging, and the *Inquiry Access Code* (IAC) is used for Inquiries.

The *Header* contains acknowledgement information, a packet sequence number, flow control information, the address of the slave device, and a header checksum. The *Payload* contains either speech or data (if the payload is data, the field will also contain a payload header).

Bluetooth Baseband

The *Bluetooth Baseband* protocol is implemented as a *Link Controller* that determines how communication takes place between Bluetooth devices in a piconet, and includes error handling and flow control mechanisms. In the *Time Division Duplex* (TDD) scheme used, the master transmits in even-numbered time slots, and slaves transmit in odd-numbered time slots. The channel is divided into 625 ms time slots, with each slot corresponding to a different RF hop frequency. The *hopping sequence* is unique for each piconet and is determined by the *Bluetooth Device ADDRess* (BD_ADDR) of the master device. Likewise, the *phase*of the hopping sequence is determined by the clock in the master.

Links are either *Synchronous Connection-Oriented* (SCO) or *Asynchronous Connection-Less* (ACL). An SCO link is a point-to-point link between a master and a slave, and is used mainly for 64kbps speech transmission (dropped packets are not retransmitted). The SCO link uses reserved slots set at regular time intervals. A Bluetooth device can support up to three separate SCO links at any one time. An ACL link is a point-to-multipoint link between the master and all of the slaves in the piconet, and can make use of any slot not reserved for an SCO link. Dropped or lost packets are normally retransmitted. The default low power state for a Bluetooth device is *Standby*. When the device is in Standby mode, it relies on its internal clock alone. There is no interaction whatsoever with any other Bluetooth device.

Link Manager Protocol (LMP)

The *Link Manager* in a Bluetooth device is responsible for setting up, authenticating and configuring a link. It communicates with Link Managers on other Bluetooth devices using the *Link Manager Protocol* (LMP), which in turn uses the services of the underlying *Link Controller*. A number of specialised LMP protocol data units are sent and received by the Link Managers in Bluetooth devices to enable them to carry out the necessary link management functions

If a Bluetooth device wishes to connect to one or more neighbouring devices that are hitherto unknown to it, it must undertake an *inquiry* procedure before a connection can be established in order to discover what devices are in range, and to determine the address and clock information for each device. Once the remote devices have responded, a connection may be established using a *paging* procedure. Essentially, this involves the Bluetooth device that wishes to initiate the connection (and that will become the master device) sending a message to (paging) the device or devices discovered during the inquiry process, to which the remote devices (which will become the slave devices) will respond. Connection setup is complete once the slave devices have switched to the master's timing and frequency hopping channel parameters. When a connection has been established between two Bluetooth devices the connection consists of an ACL link. One or more SCO links can then be established. If a slave device in a piconet does not need to communicate with other devices on the channel but needs to remain synchronised, it can be put into *parked* mode by the master device. This essentially means that it goes to sleep, but wakes up periodically to check whether the master device needs to communicate with it.

In addition to the unique 48-bit address, Bluetooth devices have-user friendly names (set by the user) which can be up to 248 bytes in length. It is these names, rather than the device's 48-bit device address, that are used in inquiries. Any Bluetooth device will transmit the following information on demand:

- Device name
- Device class
- List of services
- Technical information (e.g. device features, version number etc.)

Before one Bluetooth device can use the services of another, the devices may need to establish a relationship. Two devices can establish a relationship (a process known as pairing) using a *link key*, which is a key value know only to the two devices involved. Either device can subsequently delete the link key, ending the relationship. Prior to version 2.1, only one mechanism (now known as *legacy pairing*) was available for establishing a relationship. A PIN code was used, and pairing was only successful if both devices used the same PIN code. The PIN code itself can consist of up to 16 characters from the ASCII character set. The number of characters used in a given situation depends on the type of device involved. Limited input devices such as a Bluetooth headset or speakers usually have a fixed 4-digit PIN code that is hard-wired into the device. Numeric input devices such as mobile phones allow the user to enter numeric values of up to 16 digits in length, and devices capable of alpha-numeric input, such as a Smartphone or a personal computer allow the user to enter up to 16 alpha-numeric characters in any combination.

From Bluetooth version 2.1 onwards, legacy pairing is only allowed when connecting to an older Bluetooth device. The preferred mechanism is *Secure Simple Pairing*, which employs public key encryption and offers the following operational modes:

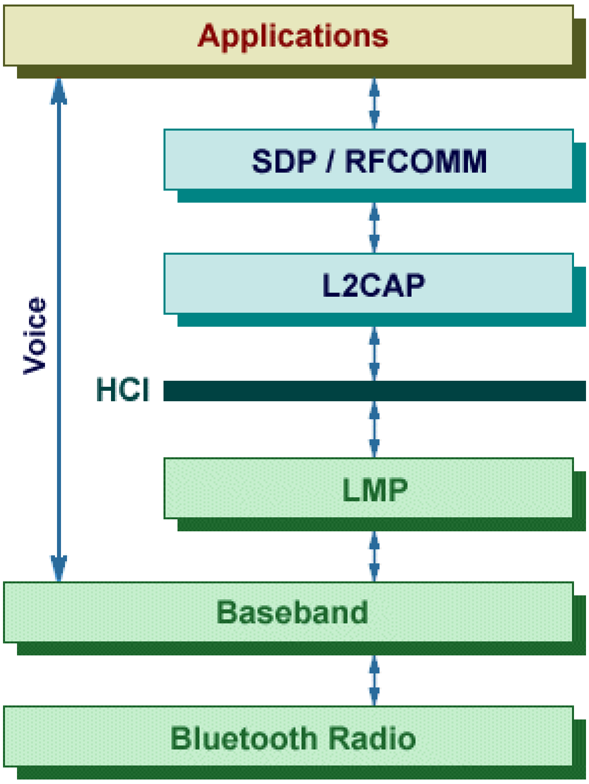
- ***Just Works*** - typically used with devices having limited or no display functionality such as a headset, and requires little or no user intervention.
- ***Numeric Comparison*** - typically used where two devices (for example, a mobile phone and a personal computer) each display a 6-digit number. The user is asked to compare the numbers and answer "Yes" or "No", depending on whether the displayed numbers are the same. If the user answers "Yes", the pairing is successful.
- ***Passkey Entry*** - typically used where one device has an input capability but no display, and the second device has display capabilities. An example would be a connection between a keyboard and a personal computer. The computer displays a 6-digit number, which the user must then enter on the keyboard. If the keyboard

recognises the code as valid, the pairing is successful.

- ***Out of Band*** - uses an alternative method of radio communication such as *Near Field Communication* (where the two devices are initially brought into very close proximity) for both device discovery and to exchange information that will be used in the pairing process. Pairing is completed using Bluetooth radio using the information acquired, provided the user confirms the pairing.

The Host Controller Interface (HCI)

The *Host Controller Interface* (HCI) provides the command interface between the Baseband Link Controller and Link Manager on the one hand, and the upper layers of the Bluetooth protocol stack on the other. It consists of a *Host Controller* (implemented as firmware on the Bluetooth device) and a *Host* (implemented as software, and including driver software). The most commonly used interface technologies are USB (in personal computers) and UART (in mobile phones and PDAs). Less commonly, RS232 may also be used. The USB hardware interface is often manifested in the form of a USB dongle. Outbound data is subject to flow control implemented between the Host and the Host Controller in order to ensure that the Host Controller's data buffers do not get filled with ACL data destined for a remote device that is not responding (essentially, then, the Host manages the data buffers of the Host Controller).



The Bluetooth Protocol Stack

Logical Link Control and Adaptation Protocol (L2CAP)

The *Logical Link Control and Adaptation Protocol* (L2CAP) is only used for ACL links. It provides both connection-oriented and connectionless data services to upper layer protocols using multiple logical channels. Reliable connection-oriented services are provided by the *Enhanced Retransmission Mode* (ERTM), while unreliable connectionless services are provided by the *Streaming Mode* (SM). The latter provides no error handling or flow control. Although payloads of up to 64 kB are possible, the default *Maximum Transmission Unit* (MTU) size is 672 bytes. L2CAP packets may be3 segmented for transmission over Baseband links. The end-point of an L2CAP channel on a device is identified using a local name known as a *Channel Identifier* (CID). Connectionless channels are restricted to sending data in one direction only, and are mainly used for control signals such as those used to set up and configure connection-oriented channels.

RFCOMM Protocol

RFCOMM is a simple transport protocol that emulates an EIA-232/RS-232 interface between two devices over the L2CAP protocol. The devices in question are classed as *Type 1* (for example, communication end points such as computers and printers) and *Type 2* (for example, part of the communication hardware, such as a modem). The interface may involve multiple emulated serial ports, although only one RFCOMM session can exist between any two devices at any one time. Each connection between a client application and a server application is identified by a 6-bit *Data Link Connection Identifier* (DLCI) that is unique within an RFCOMM session (each session has its own L2CAP *Channel ID* (CID). The device opening the first emulated serial port must first establish an L2CAP channel with the remote device. The device that closes the last connection for a particular session will be responsible for closing the L2CAP channel. Various flow control mechanisms may be available between RFCOMM and the lower level L2CAP protocol, depending on the implementation. RFCOMM has, in addition, its own flow control mechanisms.

Service Discovery Protocol (SDP)

The *Service Discovery Profile* (SDP) is used by Bluetooth devices to advertise the services they can offer to other Bluetooth devices, and to discover what services those other devices can provide. Information about each service provided by a device is held in a service record consisting of a list of service attributes, and each service is identified by a 128-bit *Universally Unique Identifier* (UUID). A service attribute consists of a 16-bit attribute ID and a variable length attribute value. Each transaction consists of a *request protocol data unit* and a *response protocol data unit*.

If an L2CAP connection is being used, a client must receive a response to each request sent before it may issue a further request on the same connection. An SDP protocol data unit consists of a header, followed by a set of PDU type-specific parameters. The header consists of three fields, which are described below.

- ***PDU ID*** - identifies the type of protocol data unit being sent, i.e.:
    - *SDP_ErrorResponse*
    - *SDP_ServiceSearch*
    - *SDP_ServiceAttribute*
    - *SDP_ServiceSearchAttribute*
- ***TransactionID*** - identifies a PDU as belonging to a specific transaction
- ***ParameterLength*** - specifies the length in bytes of the parameters field

An SDP_ErrorResponse PDU is generated in response to an incorrectly formatted request PDU, or if the device receiving the request cannot respond to it for some other reason.

Bluetooth security

Versions of Bluetooth from 2.1 onwards require encryption to be enabled for all connections except *Service Discovery Protocol* (SDP) connections. An*Encryption Pause and Resume* feature is used for operations requiring encryption to be disabled, and the disabling of encryption for any other reason indicates a security breach. An encryption key, which is normally generated using the Bluetooth PIN that has been entered into one or both devices during pairing, is used to encrypt all data subsequently sent via the air interface. Each encryption key has an expiry time, and must be replaced with a new encryption key before it elapses. Security measures include authorisation and identification procedures that limit the use of Bluetooth services to the registered user, and a requirement for users to make a conscious decision to open a file or accept a data transfer. As long as these measures are enabled, the chances of unauthorised access are small. A user can also switch the Bluetooth mode on a device to "non-discoverable" to prevent connections with other Bluetooth devices from occurring.