

4. Tecnología Bluetooth

4.1. Introducción.

En primer lugar como curiosidad decir que el sobrenombre Bluetooth de la tecnología que expondremos en nuestro trabajo es un nombre tomado de un Rey Danés del siglo X, llamado Harald Blåtand (Bluetooth), que fue famoso por sus habilidades comunicativas, y por haber logrado el comienzo de la cristianización en su cerrada sociedad Vikinga.

En 1994, Ericsson Mobile Communications inició un estudio para investigar la posibilidad de una interfaz de radio de baja potencia y bajo costo entre teléfonos móviles y sus accesorios. El objetivo era eliminar los cables entre los teléfonos móviles y tarjetas de PCS, headsets, dispositivos desktop, etc. El estudio fue parte de otro gran proyecto de investigación que involucraba multicomunicadores conectados a la red celular por medio de los teléfonos celulares. El ultimo enlace en dicha conexión debería ser un radio enlace de corto rango. A medida que el proyecto progresaba, se vio claro que las aplicaciones que envuelven dicho enlace de corto rango serían ilimitadas. A comienzos de 1997, Ericsson se aproxima a otros fabricantes de dispositivos portátiles para incrementar el interés en esta tecnología.

El motivo era simple, para que el sistema fuera exitoso y verdaderamente utilizable, una cantidad crítica de dispositivos portátiles debiera utilizar la misma tecnología de radio enlaces de corto alcance. En Febrero de 1998, cinco compañías, Ericsson, Nokia, IBM, Toshiba e Intel, forman un Grupo de Interés Especial (SIG). Dicho grupo contiene la mezcla perfecta en lo que es el área de negocios, dos líderes del mercado en telefonía móvil, dos líderes del mercado en computadoras laptop y un líder del mercado en tecnología de procesamiento de señales digitales. La meta era establecer la creación de una especificación global para conectividad sin hilos de corto alcance.

El grupo Bluetooth SIG (Special Interest Group), ha desarrollado la especificación Bluetooth, que permite el desarrollo de aplicaciones de comunicación de datos de manera inalámbrica. El SIG fue rápidamente ganando miembros, como las compañías 3Com, Axis Communication, Compaq, Dell, Lucent Technologies UK Limited, Motorola, Qualcomm y Xircom.

La especificación, define un conjunto completo de protocolos, los cuales dan gran flexibilidad al estándar para operar una cierta variedad de aplicaciones. En adelante, se describirá las características principales de los conjuntos de protocolos usados, así como las principales aplicaciones que se han dividido tener bajo el estándar.

La iniciativa Bluetooth, tiene como objetivo aumentar la efectividad de las comunicaciones entre cortas distancias, tanto en el área de trabajo como en los espacios públicos.

4.2. ¿Qué es Bluetooth?

Bluetooth es un estándar desarrollado para la comunicación inalámbrica de datos de corto alcance. Entre sus principales características, pueden nombrarse su robustez, baja complejidad, bajo consumo y bajo costo.

La Tecnología Bluetooth es de pequeña escala y bajo costo. Esta tecnología opera en la banda de 2.4 GHz. Tiene la capacidad de atravesar paredes y maletines, por lo cual es ideal tanto para el trabajo móvil, como el trabajo en oficinas.

4.3. Clases Bluetooth:

La clasificación de los dispositivos Bluetooth como "Clase 1", "Clase 2" o "Clase 3" es únicamente una referencia de la potencia de transmisión del dispositivo, siendo totalmente compatibles los dispositivos de una clase con los de la otra. En la siguiente tabla se muestra los rangos de cada clase:

CLASE	POTENCIA MÁX. PERMITIDA (mW)	POTENCIA MÁX. PERMITIDA (dBm)	RANGO APROXIMADO
Clase 1	100 mW	20 dBm	~100 metros
Clase 2	2,5 mW	4 dBm	~20 metros
Clase 3	1 mW	0 dBm	~1 metro

Cabe mencionar que en la mayoría de los casos, la cobertura efectiva de un dispositivo de clase 2 se extiende cuando se conecta a uno de clase 1. Esto es así gracias a la mayor sensibilidad y potencia de transmisión del dispositivo de clase 1. Es decir, la mayor potencia de transmisión del dispositivo de clase 1 permite que la señal llegue con energía suficiente hasta el de clase 2. Por otra parte la mayor sensibilidad del dispositivo de clase 1 permite recibir la señal del otro pese a ser más débil.

Versiones Bluetooth:

- Bluetooth v.1.1
- Bluetooth v.1.2
- Bluetooth v.2.0
- Bluetooth v.2.1

La versión 1.2, a diferencia de la 1.1, provee una solución inalámbrica complementaria para co-existir Bluetooth y [Wi-Fi](#) en el espectro de los 2.4 GHz, sin interferencia entre ellos.

La versión 1.2 usa la técnica "Adaptive Frequency Hopping (AFH)", que ejecuta una transmisión más eficiente y un cifrado más seguro. Para mejorar las experiencias de

los usuarios, la V1.2 ofrece una calidad de voz (Voice Quality - Enhanced Voice Processing) con menor ruido ambiental, y provee una más rápida configuración de la comunicación con los otros dispositivos Bluetooth dentro del rango del alcance, como pueden ser PDAs, HIDs (Human Interface Devices), computadoras portátiles, computadoras de escritorio, Headsets, impresoras y celulares.

La versión 2.0, creada para ser una especificación separada, principalmente incorpora la técnica "Enhanced Data Rate" (EDR) que le permite mejorar las velocidades de transmisión en hasta 3Mbps a la vez que intenta solucionar algunos errores de la especificación 1.2.

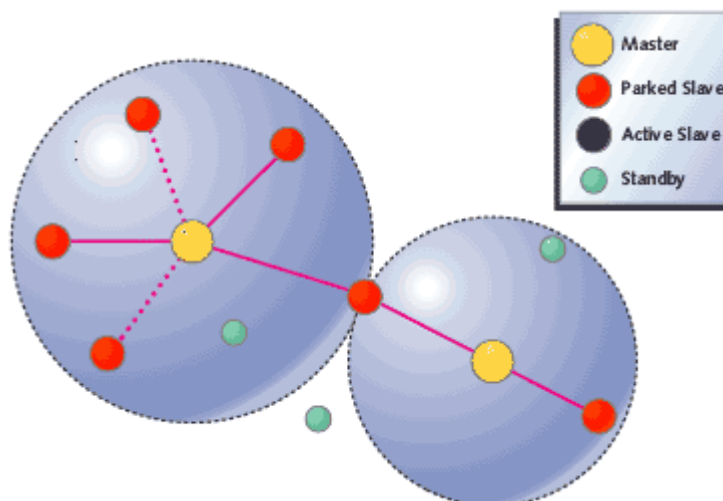
La versión 2.1, simplifica los pasos para crear la conexión entre dispositivos, además el consumo de potencia es 5 veces menor.

En cuanto al ancho de banda:

VERSIÓN	ANCHO DE BANDA (Mbit/s)
Versión 1.2	1 Mbits/s
Versión 2.0 + EDR	3 Mbits/s
UWB Bluetooth (propuesto)	53-480 Mbits/s

4.4. Redes Bluetooth:

La topología de las redes Bluetooth puede ser punto-a-punto o punto-a-multipunto.



Los dispositivos, se comunican en redes denominadas piconets. Estas redes tienen posibilidad de crecer hasta tener 8 conexiones punto a punto. Además, se puede extender la red mediante la formación de scatternets. Una scatternet es la red producida cuando dos dispositivos pertenecientes a dos piconets diferentes, se conectan.

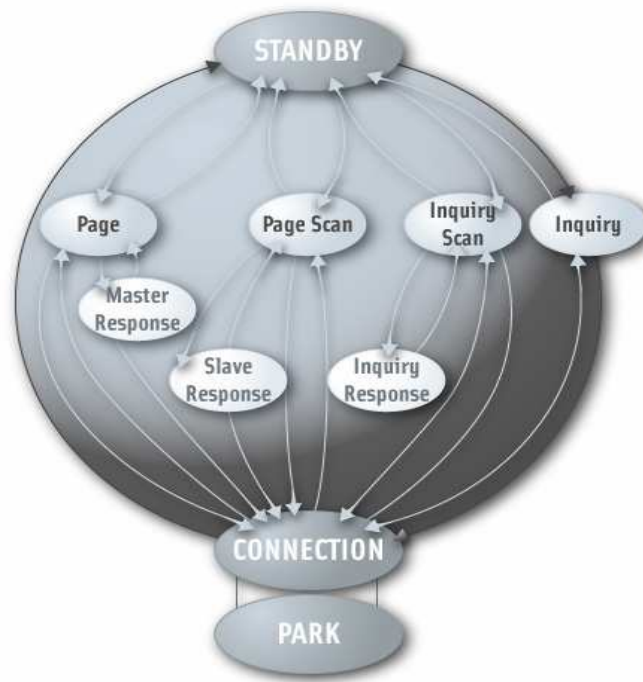
En una piconet, un dispositivo debe actuar como master, enviando la información del reloj (para sincronizarse) y la información de los saltos de frecuencia. El resto de los dispositivos actúan como slaves (esclavos).

4.5. Conexiones Bluetooth:

Las conexiones Bluetooth, son establecidas a través de la siguiente técnica:

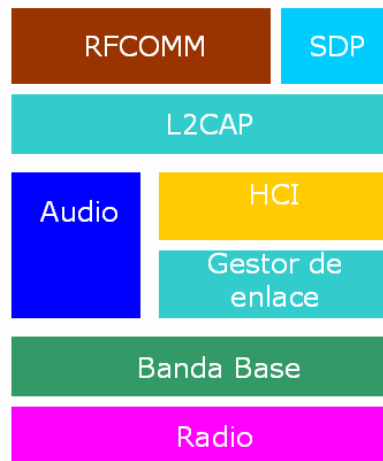
- **Standby:** Los dispositivos en un "piconet" que no están conectados, están en modo standby, ellos escuchan mensajes cada 1,28 segundos, sobre 32 saltos de frecuencias.
- **Page/Inquiry:** Si un dispositivo desea hacer una conexión con otro dispositivo, éste le envía un mensaje de tipo page, si la dirección es conocida; o una petición a través de un mensaje de page, si éste no es conocido. La unidad "master" envía 16 page message idénticos, en 16 saltos de frecuencias, a la unidad "slave". Si no hay respuesta, el "master" retransmite en los otros 16 saltos de frecuencia. El método de Petición (inquiry) requiere una respuesta extra por parte de la unidad "slave", desde la dirección MAC, que no es conocida por la unidad "master".
- **Active:** Ocurre la transmisión de datos.
- **Hold:** Cuando el "master" o el "slave" desean, puede ser establecido un modo en el cual no son transmitidos datos. El objetivo de esto es conservar el poder.
- **Sniff:** El modo sniff, es aplicable solo para las unidades "slaves", es para conserva el poder. Durante este modo, el "slave", no toma un rol activo en la "piconet", pero escucha a un reducido nivel.
- **Park:** El modo park es un nivel más reducido, que el modo hold. Durante este, el "slave" es sincronizado a la "piconet", por eso no requiere un reactivación completa, pero no es parte del tráfico. En este estado, ellos no tienen direcciones MAC y solo escuchan para mantener su sincronización con el "master" y chequear los mensajes de broadcast.

En la siguiente figura se puede ver un resumen de los distintos estados en los que se puede encontrar un dispositivo Bluetooth y sus posibles evoluciones entre estados,



4.6. Conjunto de protocolos

Uno de los principales objetivos de la tecnología Bluetooth es conseguir que aplicaciones de dispositivos diferentes mantengan un dialogo fluido. Para conseguirlo, ambos, deben ejecutarse sobre la misma pila de protocolos.



La pila esta constituida por dos clases de protocolos. Una primera clase llamada de protocolos específicos que implementa los protocolos propios de Bluetooth. Y una segunda clase formada por el conjunto de protocolos adoptados de otras especificaciones. Esta división en clases en el diseño de la pila de protocolos de Bluetooth permite aprovechar un conjunto muy amplio de ventajas de ambas. Por un lado, al implementar protocolos específicos de Bluetooth permite utilizar los beneficios que aporta la adopción de la tecnología Bluetooth. Por otro lado la utilización de

protocolos no específicos ofrece la ventaja de la interacción de esta tecnología con protocolos comerciales ya existentes. Así como la posibilidad de que Bluetooth este abierto a implementaciones libres o nuevos protocolos de aplicación de uso común. La pila de protocolos se puede dividir en cuatro capas lógicas:

- Núcleo de Bluetooth : Radio, Banda Base, LMP, L2CAP, SDP
- Sustitución de cable: RFCOMM
- Protocolos adoptados: PPP, UDP, TCP, IP, OBEX, WAP, IRMC, WAE
- Control de telefonía: TCS-binary, AT-Commands

El llamado núcleo de Bluetooth, ha sido implementado en su totalidad por el SIG, no obstante otros como RFCOMM y TCS-binary pese a ser desarrollados por el propio SIG, los han desarrollado siguiendo las recomendaciones de otras instituciones de telecomunicaciones.

El resto de capas lógicas de sustitución de cable, de control de telefonía y de protocolos adoptados agrupan a los protocolos orientados a aplicación, permitiendo así a las diferentes aplicaciones existentes o desarrolladas en el futuro poder correr sobre el núcleo de Bluetooth. Como es una norma abierta en cuanto a los protocolos que corren encima de los protocolos específicos de transporte, se pueden hacer implementaciones que usen protocolos tan usados como FTP o HTTP por ejemplo.

A continuación se va a realizar una breve descripción de los protocolos que emplea Bluetooth en su núcleo, y que constituyen la base de su funcionamiento.

Radio Bluetooth (RF):

Bluetooth fue diseñado para operar en un entorno de radio frecuencia ruidosa (LANs, mandos, hornos microondas), y para ello utiliza un esquema de reconocimiento rápido y saltos de frecuencia para garantizar la robustez del enlace. Este sistema opera en la banda de frecuencia de 2.4 GHz, libre para ISM (Industrial, Científica, Medica) más exactamente comenzando en 2.402 GHz y acabando en 2.4835 GHz. Con canales RF de $f = 2402 + k$ MHz siendo $k = 0..78$.

El espacio entre canales es de 1 MHz, no obstante es necesario tener unos márgenes de protección respecto al ancho de banda de trabajo, así pues, el límite superior de protección es de 2 MHz y un límite inferior es de 3,5 MHz.

▪ Características de la modulación:

La modulación que emplea Bluetooth es GFSK (Gaussian Frequency Shift Keying) con un producto ancho de banda por tiempo $BT=0.5$. Este tipo de modulación permite un bajo coste. El índice de modulación debe estar entre 0.28 y 0.35. Un uno binario se representa por una desviación positiva de frecuencia y un cero binario como una desviación negativa. La desviación mínima no ha de ser menor de 115 KHz.

- Características del dispositivo receptor:

El aspecto más importante en el dispositivo receptor es el nivel de sensibilidad. Para poder medir una tasa de error de bit, el equipo receptor envía de vuelta la información decodificada. Para una tasa de error o BER (Bit Error Rate) del 0.1% se define el nivel de sensibilidad de un receptor Bluetooth mayor o igual a -70dBm .

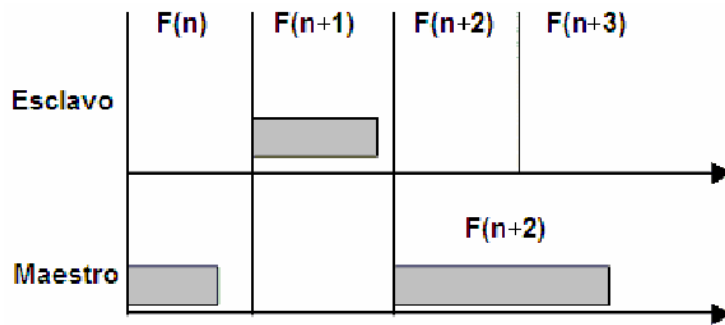
Banda Base (BaseBand)

Bluetooth brinda una conexión punto-a-punto o conexión punto-a-multipunto. Como se ha dicho anteriormente, dos o más unidades compartiendo el mismo canal forman una piconet o picored. Cada piconet tiene una secuencia de salto diferente y como elementos a destacar tenemos un maestro que puede tener hasta siete esclavos activos, además pueden haber muchos más esclavos en estado parked o aparcados, en realidad un número ilimitado de ellos. Estos esclavos no están activos en el canal sin embargo están sincronizados con el maestro con el fin de asegurar una rápida iniciación de comunicación. El maestro (Master) es el responsable de la sincronización entre los dispositivos de la piconet, su reloj y saltos de frecuencia controlan al resto de dispositivos. Además el maestro es quien, de manera predeterminada, lleva a cabo el procedimiento de búsqueda y establecimiento de la conexión. Los esclavos simplemente se sincronizan y siguen la secuencia de saltos determinada por el maestro.

La topología Bluetooth permite la interconexión de varias piconets formando una scatternet, Aunque no existe sincronización entre piconets, un dispositivo puede pertenecer a varias de ellas haciendo uso de la multiplexación por división del tiempo (TDD), aunque el dispositivo solo esta activo en una piconet a la vez.

Técnica TDD (Time Division Duplex):

El canal físico contiene 79 frecuencias de radio diferentes, las cuales son accedidas de acuerdo a una secuencia de saltos aleatoria. El valor de saltos estándar es de 1600 saltos/s. El canal está dividido en timeslots o slots (ranuras de tiempo), cada slot corresponde a una frecuencia de salto y tiene una longitud de 625 μs . Cada secuencia de salto en una piconet está determinada por la dirección del maestro (48 bits) de la piconet. Todos los dispositivos conectados a la piconet están sincronizados con el canal en salto y tiempo. En una transmisión, cada paquete debe estar alineado con el inicio de un slot y puede tener una duración de hasta cinco timeslots. Durante la transmisión de un paquete la frecuencia es fija. Para evitar fallos en la transmisión (crosstalk), el maestro inicia enviando en los timeslots pares y los esclavos en los timeslots impares.



Existen dos tipos de enlaces físicos entre maestros y esclavos:

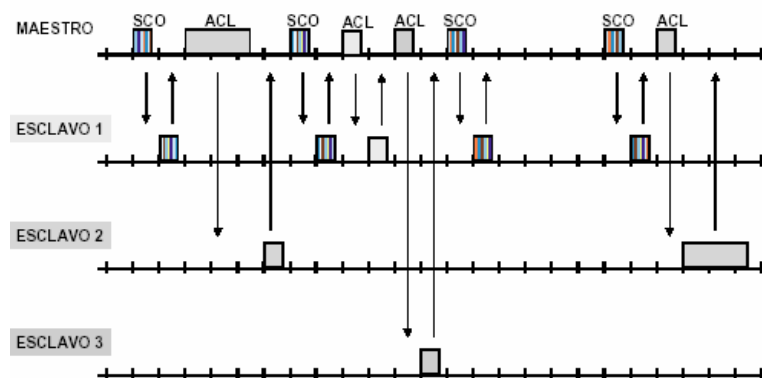
- Enlace SCO (Synchronous Connection-Oriented):

El enlace SCO es una conexión simétrica punto-a-punto con un ancho de banda fijo entre el maestro y un esclavo específico. Para lograr la comunicación, el enlace SCO reserva slots en intervalos regulares en la iniciación, por esto el enlace puede ser considerado como una conexión de conmutación de circuitos. En este tipo de enlace no es necesario asegurar la entrega y suele ser utilizado para comunicaciones de voz.

- Enlace ACL (Asynchronous Connection-Less):

El enlace ACL es una conexión simétrica o asimétrica punto-a-multipunto entre el maestro y uno o más esclavos activos en la piconet sin reserva de ancho de banda. Este enlace de comunicación es un tipo de conexión de conmutación de paquetes. Aquí, a diferencia del anterior, se necesita asegurar la entrega de datos y es utilizado para transferencia de datos sin requerimientos temporales.

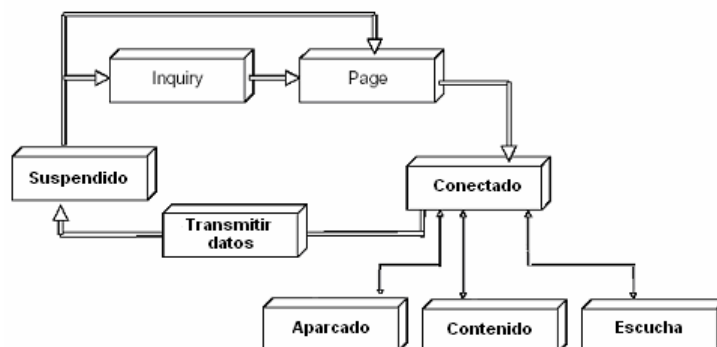
Ejemplo de tráfico SCO y ACL en una piconet:



- Establecimiento de conexiones en Bluetooth:

Para establecer nuevas conexiones se utilizan los procedimientos de acceso que son los de búsqueda o paging y los de pregunta o inquiry. Si no se conoce nada sobre el dispositivo remoto debe seguirse tanto el procedimiento inquiry como el de paging. Si

se conocen algunos detalles del dispositivo remoto sólo será necesario el procedimiento de paging.



❖ Pregunta (Inquiry):

El procedimiento de “inquiry” permite a un dispositivo descubrir qué dispositivos están en su zona de cobertura, determinando sus direcciones y el reloj de todos aquellos que respondan al mensaje de búsqueda. Entonces, si el dispositivo emisor lo desea, establecerá una conexión con alguno de los dispositivos descubiertos.

El mensaje de búsqueda no contiene ningún tipo de información sobre la fuente emisora del mensaje, no obstante, puede indicar qué clase de dispositivos deberían responder. Para poder conseguir esto existe un código de acceso de pregunta (GIAC) para preguntar por algún tipo de dispositivo en especial, y una serie de códigos de acceso de pregunta dedicados (DIAC) para tipos de dispositivos.

Así pues un dispositivo que quiera conectar con otro dispositivo en concreto continuamente transmite el mensaje GIAC en diferentes frecuencias de salto. La secuencia de saltos está determinada en la parte menos significativa de la dirección del GIAC, incluso cuando se utilizan los DIAC. Un dispositivo que quiera ser descubierto, cada cierto tiempo entrará en un estado de escáner de preguntas llamado “inquiry scan” para atender a estos mensajes.

Una vez atendida la pregunta, el dispositivo destino, entrará en el modo “inquiry response” y transmite un mensaje de respuesta que consiste en un paquete FHS (Frequency Hop Synchronization), que tiene los parámetros del dispositivo. El maestro escucha las diferentes respuestas, pero nada más leer una respuesta continua escaneando por diferentes respuestas. En el caso de que exista contienda entre diferentes dispositivos, éstos, al no recibir respuesta del maestro, esperan un número aleatorio de slots y se mantienen a la escucha de un nuevo mensaje de pregunta del maestro.

❖ Búsqueda (Paging)

El procedimiento de “paging” sigue al de “inquiry”. El procedimiento de paging pregunta por la dirección de un dispositivo Bluetooth con el que se quiere establecer la conexión. Este identificador del dispositivo se obtiene de las siguientes tres formas:

- ✓ Obtenida en la respuesta de un “inquiry”.
- ✓ Introducida por el usuario.

- ✓ Preprogramada por el fabricante del dispositivo.

Entonces el dispositivo maestro, que se encuentra en el estado page, inicia la transmisión transmite el código de acceso o DAC (Device Access Code) al dispositivo que se desea que sea esclavo de forma repetida en diferentes canales de salto. Debido a que los relojes del maestro y del esclavo no están sincronizados, el maestro no sabe exactamente cuando y en qué frecuencia de salto se activará el esclavo por lo tanto maestro se quedará a la escucha entre los diversos intervalos de transmisión hasta recibir respuesta del esclavo.

Después de haber recibido su propio código de acceso de dispositivo, el esclavo transmite un mensaje de respuesta, simplemente indicará su código de acceso, y se queda activado en espera de la llegada del paquete FHS (Frequency Hop Synchronization), Cuando el maestro ha recibido este paquete ACK, envía un paquete de control con información acerca de su reloj, dirección, clase de dispositivo, etc. El maestro se queda a la espera de una respuesta.

El esclavo se activa y responde con un nuevo mensaje ACK donde envía de nuevo su dirección y a la vez cambia el código de acceso del canal y su reloj, tomando los del maestro incluido en el paquete FHS. El esclavo establece la conexión usando para ello el reloj y la BD_ADDR del maestro para determinar la secuencia de salto del canal y el código de acceso.

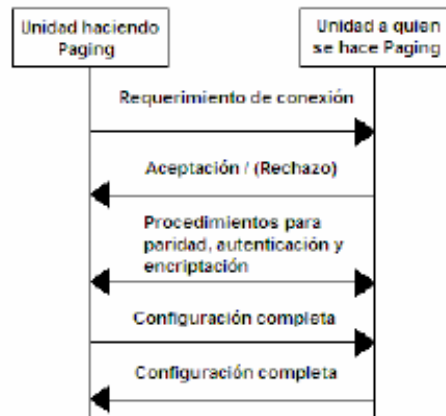
Si el maestro no obtiene esta respuesta en un determinado tiempo, él reenvía el paquete de control. Si el esclavo excede el tiempo de espera, entonces vuelve al estado de page scan. Si es el maestro quien lo excede, entonces vuelve al estado de page e informa a las capas superiores. Con el ACK, el maestro entra en modo de conexión establecida y usa su BD_ADDR para cambiar a una nueva secuencia.

LMP: Link Manager Protocol:

El siguiente protocolo específico se encarga de la gestión del enlace entre dispositivos Bluetooth, de la seguridad, del control de paquetes, potencia, calidad del servicio y control de la piconet (conmutación maestro esclavo).

Establecimiento de conexión:

Tras haberse completado el procedimiento de búsqueda o Paging, ya se está listo para establecer una conexión LMP. En primer lugar el dispositivo emisor envía la primitiva LMP_host_connection_req. El dispositivo receptor recibe el mensaje y obtiene información sobre la conexión que se va a abrir. Este dispositivo remoto puede aceptar (LMP_accepted) o rechazar (LMP_not_accepted) esa petición de conexión. Una vez establecidas todas las configuraciones necesarias, los dos dispositivos se mandan LMP_setup_complete. Después de esto, se procederá a la transmisión de los paquetes de los diferentes canales lógicos que emplea LMP.



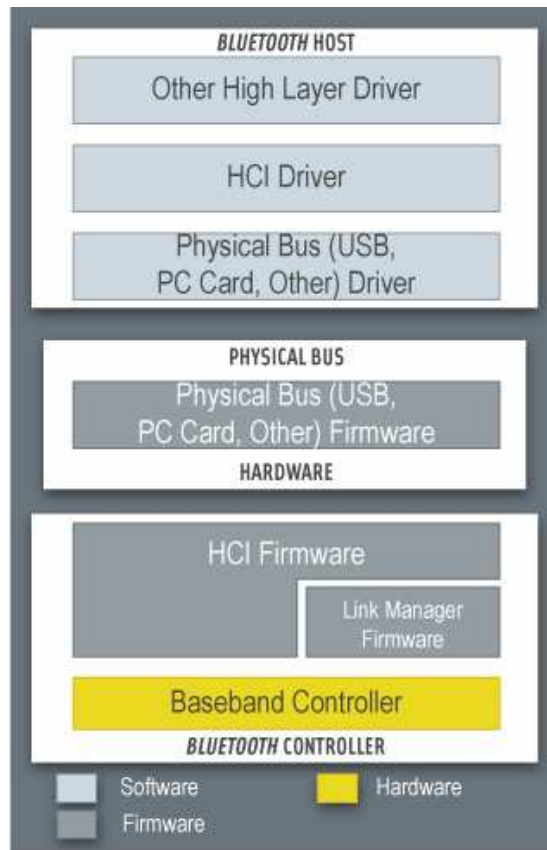
Procedimientos LMP:

LMP debe gestionar un gran número de procedimientos que permitan implementar toda su funcionalidad, esto hace que cada procedimiento tenga sus propias PDUs que se deben intercambiar entre las dos entidades de comunicación. Vamos a hacer una breve referencia a los procedimientos LMP:

1. Autenticación.
2. Pairing.
3. Cambio de la clave de enlace.
4. Cambio de la clave de enlace actual.
5. Encriptado.
6. Petición del offset del reloj.
7. Información del offset de slot.
8. Petición de información de temporización.
9. Información de la versión LMP.
10. Información de las características soportadas.
11. Conmutación del papel esclavo-maestro.
12. Petición de nombre.
13. Desconexión.
14. Control de potencia.
15. Petición modo hold.
16. Petición modo sniff.
17. Petición modo park.
18. Petición calidad de servicio.
19. Establecimiento enlaces SCO.
20. Control de paquetes multi-slot.
21. Supervisión del enlace.
22. Repuesta General.

HCI: Host Controller Interface:

Proporciona una interface de comandos al controlador de banda base y al manejador de enlace para acceder a los parámetros de configuración. Esta interface proporciona un uniforme método de acceso a las capacidades de banda base Bluetooth.



Pueden existir varias capas entre el driver HCI sobre el sistema host y el firmware HCI en el hardware Bluetooth. Estas capas intermedias, host controller transport layer, proporcionan la capacidad de intercambio de información.

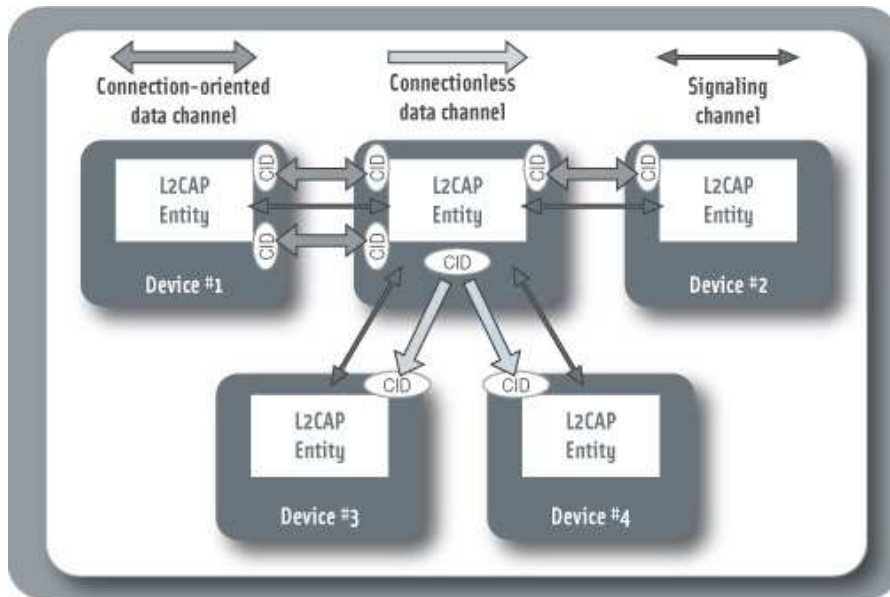
L2CAP: Logical Link Control and adaption Protocol:

L2CAP es un protocolo que se encuentra por encima del anterior protocolo (LMP), se encarga de adaptar los protocolos superiores al protocolo de banda base. Sus tres principales funciones son:

- ✓ Multiplexación de protocolos de alto nivel
- ✓ Segmentación y reensamblado de paquetes largos (hasta 64 kbytes)
- ✓ Descubrimiento de dispositivos y calidad de servicio

Para cumplir estas funciones la arquitectura L2CAP debe cumplir ciertos requisitos:

- a) L2CAP ofrece un servicio orientado a conexión donde el identificador del canal es utilizado en cada conexión, asumiendo que este canal es full-duplex y fiable, además se tiene que especificación del flujo de QoS asignada a cada dirección del canal.
- b) L2CAP esta basado en datagramas y no en flujos continuos. Se debe notar que en L2CAP, se conservan los límites del paquete, así como que no se realiza retransmisión ni control de flujo.



SDP: Service Discovery Protocol:

SDP proporciona un mecanismo que permite a las aplicaciones descubrir cuales son los servicios disponibles en su entorno y determinar las propiedades específicas de éstos. Los servicios disponibles cambian continuamente debido al dinamismo existente en el entorno, por lo que la búsqueda de servicios en Bluetooth difiere de la búsqueda de servicios en un red fija tradicional.

SDP debe proporcionar las siguientes funcionalidades en su versión 1.0:

- ✓ SDP debe permitir la búsqueda de servicios basados en atributos específicos.
- ✓ SDP debe permitir que los servicios sean descubiertos basándose en la clase de servicio.
- ✓ SDP debe permitir averiguar las características de un servicio sin tener conocimiento a priori de dicho servicio.
- ✓ SDP debe proporcionar medios para descubrir nuevos servicios (proximidad de un nuevo dispositivo, arranque de una aplicación) así como para indicar la no disponibilidad de servicios inicialmente visibles.
- ✓ SDP debe permitir el almacenar información sobre servicios de forma temporal para mejorar la eficiencia del protocolo.
- ✓ SDP debe descubrir la información de los servicios de forma incremental para evitar las transferencias excesivas de información sobre servicios no vayan a utilizarse.
- ✓ SDP proporcionar complejidad adecuada para ser utilizado en dispositivos con prestaciones limitadas.

No obstante SDP debe proporcionar los siguientes servicios en las sucesivas versiones:

- ✓ No se proporciona acceso a los servicios, sólo acceso a la información sobre los servicios.

- ✓ No aparece negociación de parámetros de servicio.
- ✓ No se proporcionan mecanismos para la tarificación por el uso de los servicios.
- ✓ No se proporciona al cliente la capacidad de controlar o cambiar la operación de un servicio.
- ✓ No se proporciona notificación de eventos para los casos en que los servicios no estén disponibles o cuando se modifican los atributos de los servicios.
- ✓ SDP 1.0 no define un API (Application Programming Interface).
- ✓ No se soportan agentes que realicen funciones tales como darse de alta en un servicio.

Registro de un servicio:

El registro de servicio está formado por un conjunto de atributos que describen un servicio determinado. Existen dos tipos de atributos, los llamados atributos universales que son comunes a todos los tipos de servicio y los llamados atributos específicos que tal como indica el nombre son específicos a una clase de servicio.

Cada atributo de un registro de servicio consta de dos partes, un identificador de propiedad y un valor de propiedad. El identificador de propiedad es un único número de 16 bits que distingue cada propiedad de servicio de otro dentro de un registro. El valor de propiedad es un campo variable que contiene la información.

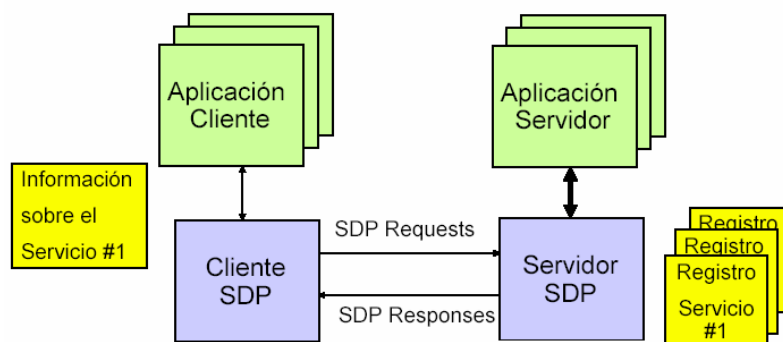
Un servidor SDP estructura los servicios en registros y mantiene una lista de apuntadores (Service Record Handle) a cada uno de ellos.

Funcionamiento de SDP

Tal como se ve en la figura, la comunicación SDP implica un servidor SDP y un cliente SDP. Se pueden dar dos escenarios posibles para que el cliente realice una petición de un servicio.

El cliente busca un servicio en un dispositivo en particular al cual el usuario se ha conectado “conscientemente”.

El cliente realizando conexiones “inconscientemente” con los dispositivos vecinos.



Vistos estos dos escenarios se pueden definir varios mensajes de petición /repuesta por parte tanto del cliente como del servidor:

- Mensajes de petición del cliente:
 - Petición de búsqueda de servicio: el cliente genera una petición para localizar los registros de servicio que concuerden con un patrón de búsqueda dado como parámetro.
 - Petición de propiedad de servicio: Una vez el cliente ya ha recibido los servicios deseados, puede obtener mayor información de uno de ellos dando como parámetros el registro de servicio y la lista de propiedades deseadas.
 - Petición de búsqueda y propiedad de servicio: se suministran un patrón de servicio con servicios deseados y una lista de propiedades deseadas que concuerden con la búsqueda.
- Mensajes de respuesta del servidor:
 - Repuesta a búsqueda de servicio: se genera por el servidor después de recibir una petición de búsqueda de servicio válida.
 - Repuesta a propiedad de servicio: el SDP genera una respuesta a una petición de propiedad de servicio. Ésta contiene una lista de propiedades de registro requerido.
 - Repuesta de búsqueda y propiedad de servicio: como resultado se puede obtener una lista de servicios que concuerden con un patrón dado y las propiedades deseadas de estos servicios.

Algunas apreciaciones sobre el funcionamiento es que un dispositivo Bluetooth puede actuar tanto cliente SDP como servidor SDP. Otro aspecto importante es que cuando el servidor deja de estar en la zona de acción del cliente, no realiza ninguna notificación vía SDP, por lo que el cliente en este caso deducirá que no está, porque ya no recibe las respuestas del servidor ante sus continuas peticiones.

RFCOMM

El protocolo RFCOMM permite emular el funcionamiento de los puertos serie sobre el protocolo L2CAP. Se basa en el estándar ETSI TS 07.10, tomando de éste un subconjunto con las partes más relevantes y realizando algunas adaptaciones.

RFCOMM permite emular los nueve circuitos de la norma RS-232(ETIATIA-232.E). Soporta hasta 60 conexiones simultáneas entre dos dispositivos Bluetooth.



Ante una configuración RFCOMM nos encontramos básicamente con dos tipos de dispositivos:

Tipo 1: Se trata de dispositivos terminales de comunicación, como los ordenadores, las impresoras.

Tipo 2: Son aquellos que forman parte de un segmento de comunicación, como por ejemplo, los módems.

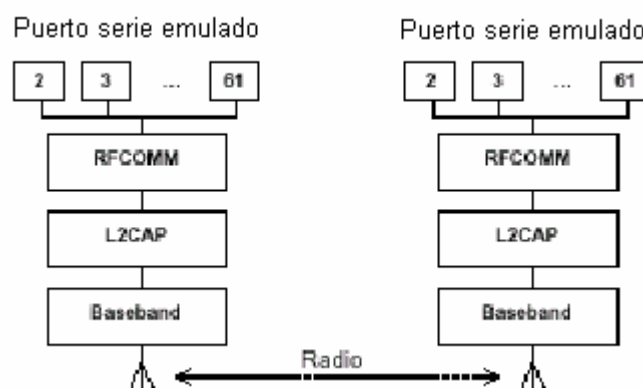
RFCOMM no hace distinción entre ambos tipos, pero el acomodarse a ellos tiene sus consecuencias en el protocolo. Por lo tanto, la transferencia de información entre dos entidades RFCOMM, se define tanto para los dispositivos tipo 1 y 2. Una parte de la información sólo se necesitará para el segundo tipo, mientras que otra se pretende que sea usada por ambos. Debido a que un dispositivo no es consciente del tipo del otro dispositivo en el camino de comunicación, cada uno debe pasar toda la información disponible especificada por el protocolo.

Emulación de múltiples puertos serie:

- Entre dos dispositivos:

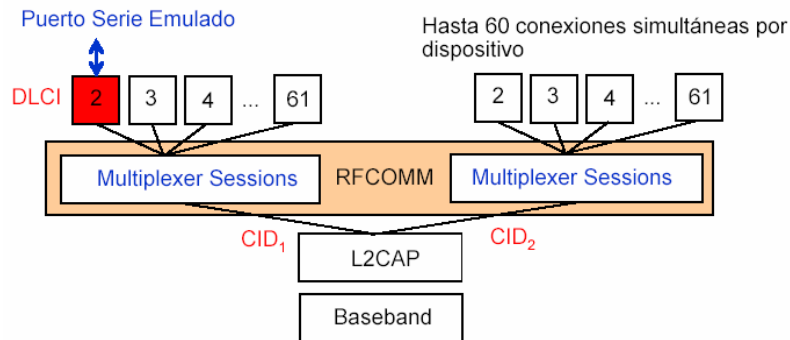
Dos dispositivos Bluetooth usando RFCOMM en su comunicación pueden abrir múltiples puertos serie. RFCOMM soporta hasta 60 puertos emulados abiertos, aunque esto es fuertemente dependiente de la implementación específica.

El identificador de Conexión de Datos de Enlace (DLCI) será el encargado de identificar la conexión entre una aplicación cliente y una servidora. El DLCI se representa con 6 bits pero su rango útil es [2,61], debido a que en TS 07.10 DLCI = 0 es el canal de control dedicado, DLCI = 1 no es útil y DLCI = 62-63 se encuentran reservados. El DLCI es único para una sesión RFCOMM establecida entre dos dispositivos. Para tener en cuenta que tanto la aplicación cliente como la servidora pueden residir en ambos lados de una sesión RFCOMM, con clientes en cualquier extremo realizando conexiones independientes, el valor del DLCI se divide entre los dos dispositivos que se comunican.



- Con múltiples dispositivos Bluetooth:

Si un dispositivo Bluetooth soporta emulación de múltiples puertos serie y las conexiones realizadas son con puntos terminales en diferentes dispositivos Bluetooth, entonces la entidad que implementa RFCOMM debe ser capaz de correr múltiples sesiones TS 07.10 multiplexadas, cada una con su propio identificador de canal L2CAP.



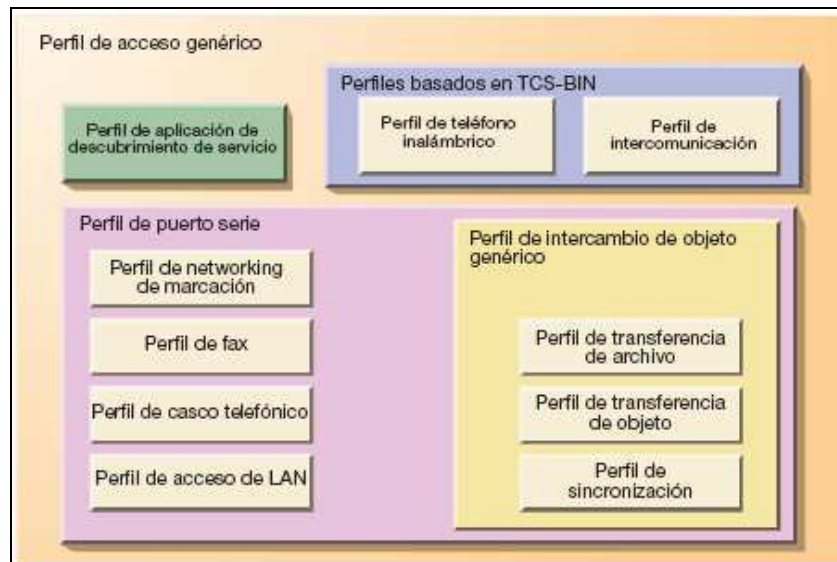
Esta capacidad de multiplexación es un elemento opcional a la hora de implementar RFCOMM, según la Especificación de Bluetooth.

4.7. Perfiles Bluetooth:

Son un conjunto de mensajes y procedimientos de la especificación Bluetooth para una situación de uso concreta del equipo. Los perfiles se encuentran asociados con las aplicaciones. Los perfiles permiten que no sea necesario implementar en un determinado dispositivo toda la pila de protocolos, sólo la parte que va a necesitar. Si el dispositivo tiene muy poca memoria y/o capacidad de procesamiento e implementamos en él toda la pila de protocolos con la carga de proceso y espacio que ello implica puede que provoquemos que el dispositivo sea totalmente ineficiente para la comunicación, por ejemplo, ratones, auriculares.

Además de la ventaja anterior, el concepto de perfil se utiliza para asegurar la interoperabilidad entre varias unidades Bluetooth que cumplan los mismos perfiles. Cada dispositivo Bluetooth tiene al menos un perfil, es decir, una aplicación para la cual se puede utilizar el dispositivo. Cuando dos dispositivos deben inter operar, es decir, comunicarse entre ellos, deben tener un perfil compartido. Si por ejemplo quiere transferir un archivo desde un ordenador preparado para Bluetooth a otro, ambos ordenadores deben admitir el perfil de transferencia de archivos.

Todos los dispositivos Bluetooth deben soportar el perfil de acceso genérico (Generic Access Profile) como mínimo. Este perfil en particular, define el descubrimiento o hallazgo de dispositivos, procedimientos de conexión y procedimientos para varios niveles de seguridad. También se describen algunos requerimientos de interfaz al usuario. Otro perfil universal, aunque no es requerido, es el perfil de acceso a descubrimiento de servicios (Service Discovery Access Profile), el cual define los protocolos y parámetros asociados requeridos para acceder a los perfiles. Un número de perfiles han sido definidos incluyendo TCS, RFCOMM y OBEX. Algunos de estos requieren la implementación de otros, y todos ellos requieren la implementación de perfiles genéricos.



A continuación se describen de forma breve los perfiles más utilizados en aplicaciones Bluetooth.

Perfil de acceso genérico. (GAP).

Este perfil define los procedimientos generales para el descubrimiento y establecimiento de conexión entre dispositivos Bluetooth. El GAP maneja el descubrimiento y establecimiento entre unidades que no están conectadas y asegura que cualquier par de unidades Bluetooth, sin importar su fabricante o aplicación, puedan intercambiar información a través de Bluetooth para descubrir qué tipo de aplicaciones soportan las unidades. También define aspectos relacionados con los niveles de seguridad.

Perfil de Aplicación del Descubrimiento de Servicio (SDAP).

Este define las características y los procedimientos que un dispositivo Bluetooth usa para descubrir los servicios registrados en otros dispositivos de Bluetooth y para recuperar cualquier información disponible deseada pertinente a estos servicios. El SDAP es dependiente del GAP.

Perfil de Puerto Serie. (SPP)

El Serial Port Profile (SPP) define los requerimientos específicos para posibilitar la emulación del puerto serie en los dispositivos Bluetooth usando el protocolo RFCOMM entre parejas de dispositivos.

Perfil genérico de intercambio de objetos. (GOEP)

Este perfil define como los dispositivos Bluetooth deben soportar los modelos de intercambio de objetos, incluyendo el perfil de transferencia de archivos, el de carga de

objetos y el de sincronización. Los dispositivos más comunes que usan este perfil son las agendas electrónicas, PDAs y teléfonos móviles.

Perfil de Teléfono Inalámbrico.

Este perfil define cómo un teléfono móvil puede ser usado para acceder a un servicio de telefonía de red fija a través de una estación base. Es usado para telefonía inalámbrica de hogares u oficinas pequeñas. El perfil incluye llamadas a través de una estación base, haciendo llamadas de intercomunicación directa entre dos terminales y accediendo adicionalmente a redes externas. Es usado por dispositivos que implementan el llamado "teléfono 3-en-1"

Perfil de Intercomunicación. (IP)

El Intercom Profile (IP) define los requerimientos necesarios por parte de los dispositivos Bluetooth para soportar comunicaciones entre parejas de teléfonos con soporte Bluetooth. Estos requerimientos son expresados en términos de servicios para el usuario final. Popularmente, este perfil se conoce con el nombre de "Walkie-Talkie".

Perfil de acceso telefónico a redes o perfil de networking de marcación (DNP)

Este perfil define los protocolos y procedimientos que deben ser usados por dispositivos que implementen el uso del modelo llamado Puente Internet. Este perfil es aplicado cuando un teléfono o modem es usado como un modem inalámbrico.

Perfil de FAX (FP)

El perfil Fax Profile define los requerimientos necesarios para que los dispositivos Bluetooth adquieran soporte de transferencia de Fax. Permitirá que teléfonos Bluetooth puedan enviar y recibir faxes.

Perfil de Manos Libres o de casco telefónico.

Este perfil define los requerimientos, para dispositivos Bluetooth, necesarios para soportar el uso de manos libres. En este caso el dispositivo puede ser usado como unidad de audio inalámbrico de entrada/salida. El perfil soporta comunicación segura y no segura.

Perfil de Acceso a LAN. (LAP)

El perfil de acceso a LAN define cómo pueden acceder a los servicios de una LAN mediante el protocolo PPP, los dispositivos que utilizan la tecnología inalámbrica Bluetooth. De esta manera pueden crearse puntos de acceso inalámbricos.

Perfil de Transferencia de Archivos.

Este ofrece la capacidad de transferir un archivo de un dispositivo Bluetooth a otro, inclusive permite navegar por el contenido de las carpetas del dispositivo.

Perfil de Transferencia de Objetos

Este ofrece la capacidad de transferir un objeto de un dispositivo Bluetooth a otro.

Perfil de Sincronización.

Este perfil provee sincronización dispositivo a dispositivo de programas de gestión de información personal, como agendas telefónicas, calendario, mensajes y notas de información.