

3

El estándar Bluetooth

IEEE 802.15.1.

3.1 La tecnología Bluetooth.

Bluetooth empieza a concebirse en Ericsson Mobile Communications AB (Suecia) en 1994 como el efecto colateral de un proyecto sobre enlaces de comunicadores múltiples conectados a la red celular mediante teléfonos; cuatro años más tarde, aparece el grupo de interés en la tecnología Bluetooth SIG (Special Interest Group) en el que aparecían Ericsson, Toshiba, IBM e Intel como empresas promotoras de la tecnología; más tarde se sumaron otros promotores como 3Com, Lucent, Microsoft y Motorola generándose el consorcio Bluetooth; actualmente el Bluetooth SIG cuenta con más de

2000 empresas. El objetivo de este grupo es, fundamentalmente promover actividades relativas a mejorar la interoperabilidad de forma que con ello se favorezca el desarrollo del mercado. Para tratar el tema de la interoperabilidad, en términos únicamente de test bed, o sea, de métrica/evaluación de características técnicas, se realizan los tests de productos Bluetooth mediante unos protocolos denominados Blue Units que se basan en kits de desarrollo producidos por la firma inglesa Cadence Symbionics Group parte de la firma estadounidense Cadence Design Systems a los que se puede acceder a través de Ericsson Microelectronics AB (Suecia), AU System y Sigma ComTec. En estos momentos, para que un producto pueda considerarse como Bluetooth tiene que cumplir una serie de protocolos y perfiles; los protocolos describen cómo se realizan las tareas básicas como señalización telefónica, gestión de enlace y lo que se conoce como Service Discovery, es decir, determinación de qué servicios están disponibles desde o través de otros productos Bluetooth; los perfiles describen la forma en que diferentes protocolos y procedimientos básicos funcionan conjuntamente en diferentes productos y aplicaciones Bluetooth: los perfiles están considerados como la primera aproximación en términos de conseguir la interoperabilidad. Existen grupos de trabajo dentro del SIG Bluetooth que están redactando perfiles para impresión, interfase usuario-equipo y posicionamiento (aplicaciones GPS). Además, como un medio para tratar de acelerar el despegue del mercado, se ha creado recientemente el Bluetooth Measurement Initiative dentro del Bluetooth SIG cuyo objetivo es desarrollar hardware y software para el testing de interoperabilidad.

Desde un punto de vista de tecnología de transmisión, Bluetooth está basado en la tecnología clásica del escenario inalámbrico conocida como spread spectrum (espectro esparcido), en concreto en frequency hopping (salto de frecuencia): estos sistemas de frequency hopping dividen la banda de frecuencia en varios canales de salto (hopping); en el transcurso de la conexión se produce una transición brusca (salto o hopping) de un canal a otro de forma pseudo aleatoria. Desde un punto de vista de arquitectura, Bluetooth se basa en el concepto de picored, un concepto genérico en las redes de área personal que se refiere a la capacidad de varios equipos para configurarse como una red; de una forma más estricta, la picored se puede definir como la red de corto alcance formada por dos o más unidades o equipos que comparten un canal, es decir, que funcionan de forma síncrona y siguiendo al misma frecuencia de hopping. La picored se diferencia de otras redes similares de su entorno mediante la secuencia de frecuencia de hopping. Las picorredes se pueden enlazar de forma ad-hoc generando lo que se conoce como red de dispersión o scatternet: estas scatternets permiten que se pueda producir la comunicación entre configuraciones flexibles de forma continua; estas redes pueden actuar también como enlace inalámbrico a redes de datos ya existentes. El chip Bluetooth está formado por un transceiver de radio-frecuencia, una unidad de control de enlace banda-base conjuntamente con el software de gestión y un subsistema de antena; a los equipos que incluyen este chip y verifican las especificaciones Bluetooth se les conoce, en la literatura internacional, como productos Bluetooth enabled.

3.2 Arquitectura WPAN Bluetooth.

3.2.1 Vista de alto nivel.

El estándar IEEE 802.15.1 presenta una WPAN que utiliza tecnología inalámbrica Bluetooth. En este trabajo el término Bluetooth WPAN o simplemente el término IEEE 802.15.1 WPAN se refiere a una WPAN que utiliza tecnología Bluetooth inalámbrica.

3.3.1 Interconexión entre sistemas abiertos (OSI).

Hay dos formas de ver cualquier diseño de sistemas de comunicación; por su arquitectura y su funcionalidad. Un acercamiento hacia la arquitectura enfatiza las divisiones lógicas de un sistema y como son compatibles. El acercamiento hacia la funcionalidad hace énfasis en los componentes actuales y sus interconexiones.

A continuación se presenta una vista de la arquitectura de una WPAN (Figura 3.1). Se enfatiza la tradicional separación de larga escala del sistema en dos partes; physical layer (PHY) del IEEE 802.15.1 y la subcapa MAC (medium access control) del DLL (data link layer).

En la siguiente figura se muestra la pila del protocolo en la séptima capa del modelo del OSI en la tecnología inalámbrica de Bluetooth y su relación con este estándar. Como se puede observar en la siguiente figura las subcapas LLC (logical link control) y el MAC juntas abarcan las funciones deseadas para el DLL del modelo de OSI.

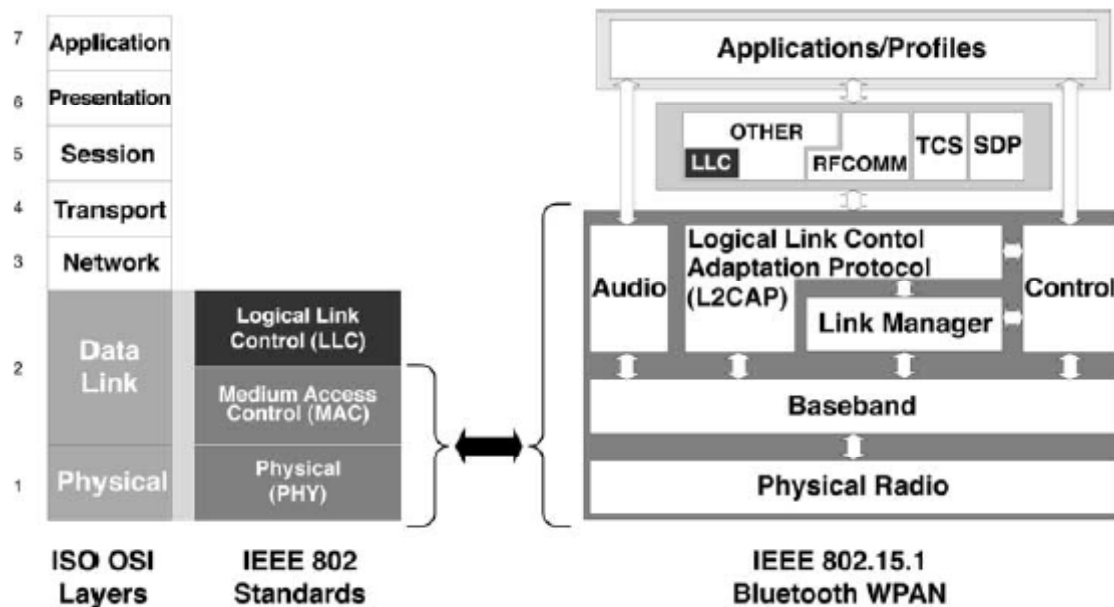


Figura 3.1 - Capas del protocolo Bluetooth [1].

3.3.2 Vistazo a la red WPAN Bluetooth.

La tecnología inalámbrica de Bluetooth utiliza un radio de corto alcance que ha sido optimizado para el ahorro de energía, operación adecuada de la batería, tamaño pequeño y para ser utilizada en aparatos personales de bajo peso. Una WPAN es capaz de soportar canales síncronos de comunicación para telefonía de voz y canales de comunicación asíncronas para comunicación de datos. Dichas facilidades permiten que una amplia gama de aplicaciones y de aparatos trabaje en una WPAN. Por ejemplo, un teléfono celular puede usar canales circuit-switched para transportar audio desde y para un receptor en el encabezado mientras se encuentra utilizando un canal packet-switched para intercambiar datos con una computadora portátil.

Una WPAN opera en la banda libre de los 2.4 GHz. Se utiliza un transceptor de

fast frequency-hop (1600 hops/s) para evitar la interferencia y la caída de señales. Para reducir la complejidad del transceptor se utiliza la técnica binaria FSK (frequency shift keying) para transmitir símbolos con un rango de 1 Símbolos/s. Se utiliza un canal ranurado, cada ranura de tiempo tiene una duración de 625 μ s. Una trama rápida de TDD (time division duplex) se utiliza para permitir comunicaciones full duplex en capas superiores. En el canal, la información se intercambia a través de paquetes. Cada paquete se transmite en una frecuencia diferente dentro de la secuencia de espera. Un paquete normalmente cubre una sola ranura, pero se puede extender de tres a cuatro ranuras. Para tráfico de datos, de manera unidireccional es posible transmitir un máximo de 723.2 kb/s entre dos dispositivos. Un canal vi direccional soporta un tráfico de voz entre dos dispositivos con una velocidad de hasta 64 kb/s. La inestabilidad para el tráfico de voz se mantiene bajo al usar ranuras de tiempo pequeñas en la transmisión.

La figura 3.2 muestra el formato general del contenido de una ranura de tiempo, transmitida al aire en una WPAN de Bluetooth. El paquete abarca un código de acceso de tamaño fijo, el cual es utilizado, entre otras cosas, para distinguir una WPAN de otra. El encabezado del paquete de tamaño fijo, que se utiliza para manejar la transmisión en una WPAN; y una carga de datos de tamaño variable, que transporta información de capas superiores. Debido al tamaño tan reducido de de estos paquetes, se necesita que una capa superior más larga sea dividida en segmentos antes de que sea transmitida al aire.

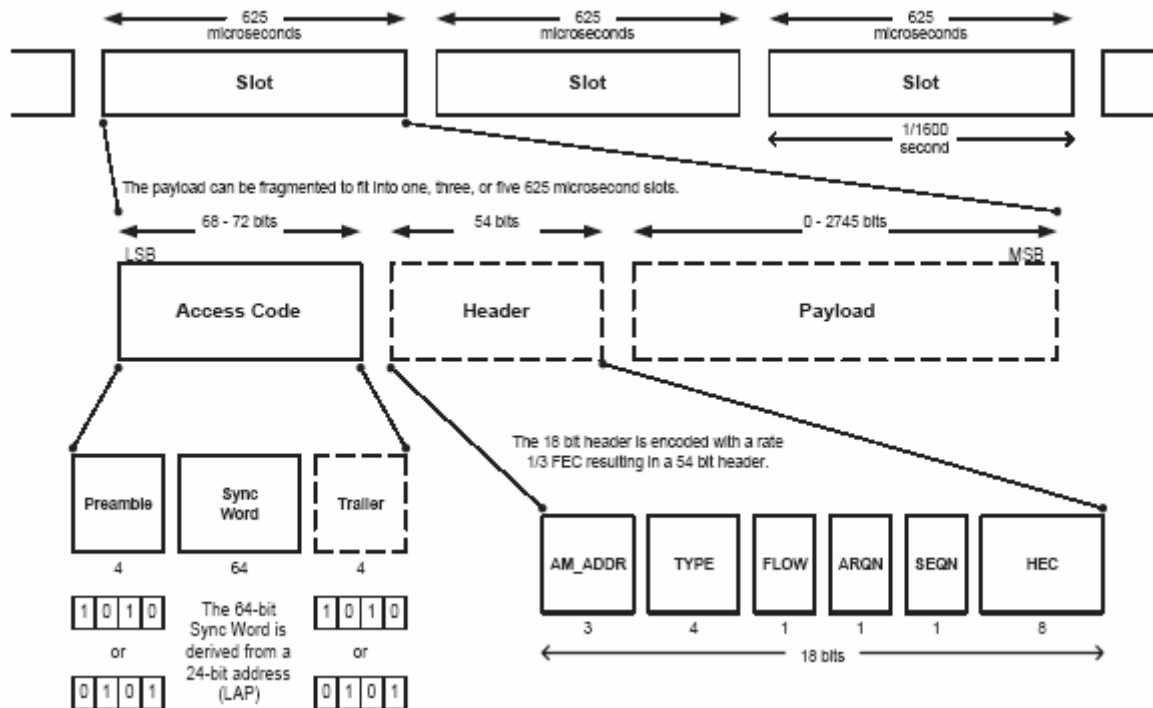


Figura 3.2 - Trama de información [1].

3.4 Topología de conexión de las WPAN Bluetooth.

3.4.1 Picoredes WPAN Bluetooth.

Una picored es una WPAN formada por dispositivos Bluetooth que sirven como maestros en la picored y uno o más dispositivos Bluetooth que sirven de esclavos. Un canal de frequency-hopping basado en la dirección del maestro define cada picored. Todos los elementos participantes en una comunicación dentro de una picored dada son sincronizados al canal de frequency-hopping, utilizando el reloj del aparato maestro de la picored. Los dispositivos esclavos solo se comunican con el maestro en una estructura punto a punto bajo el control del maestro. Las transmisiones del maestro deben de ser ya

sea de la forma punto a punto o bien punto a multipunto. Cada lugar determina que si algunos elementos deben ser siempre maestros o esclavos. Sin embargo, este estándar no distingue entre elementos que son permanentemente maestros o esclavos. Durante una sección de comunicación un elemento esclavo puede comportarse como maestro en un momento dado y viceversa.

3.4.2 Redes dispersas WPAN Bluetooth.

Una red dispersa (scatternet) es una colección de varias picoredes en operación que se traslapan en tiempo y en espacio. Un aparato Bluetooth puede participar en múltiples picoredes al mismo tiempo, por lo que existe la posibilidad de que la información fluya más allá de las fronteras de una picored. Un aparato en una scatternet puede ser esclava en varias picoredes, pero puede ser maestra en solo una de ellas. La figura 3.3 muestra las diferentes formas en que los dispositivos Bluetooth se pueden interconectar para formar varios sistemas de comunicación.

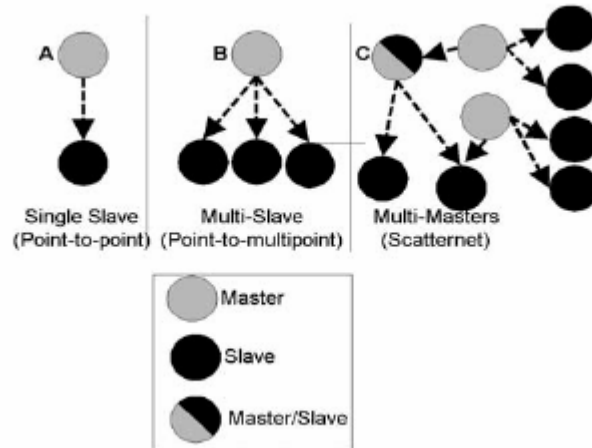


Figura 3.3 - Varios formatos en una picored. (a) Operación con un solo esclavo. (b) Operación multiesclavo; (c) Operación tipo scatternet [1].

3.5 El protocolo Bluetooth de pila.

La figura 3.4 muestra la pila del protocolo Bluetooth, la cual incluye los protocolos Bluetooth específicos (por ejemplo; Link Manager Protocol, LMP; Logical Link Control and Adaptation Protocol, L2CAP) y los protocolos bluetooth no específicos (también conocidos como los “otros”). Dichos protocolos no específicos incluyen al OBEX (Object Exchange Protocol), al PPP (point to point protocol), el WAP (wireless application protocol), y algunos más. Cuando se diseñó el protocolo se hizo pensando principalmente en optimizar la reutilización de los protocolos existentes. Este protocolo también reutiliza la ayuda para adaptar las aplicaciones existentes para trabajar con la tecnología inalámbrica de Bluetooth y asegurar una suave operación y interoperabilidad de dichas aplicaciones. Así, muchas aplicaciones ya desarrolladas por vendedores pueden tomar inmediatamente ventaja del hardware y el software de los

sistemas que cumplen con las especificaciones para Bluetooth. Las especificaciones están disponibles públicamente y permiten el desarrollo de un gran número de nuevas aplicaciones que toman todas las ventajas de las capacidades de la tecnología inalámbrica Bluetooth.

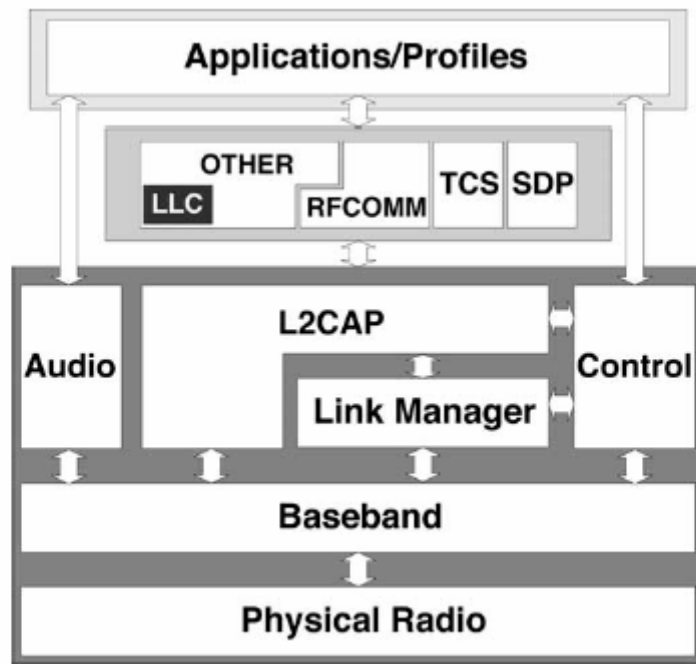


Figura 3.4 - El protocolo Bluetooth en una pila de capas [1].

La capa del LLC no es parte de las especificaciones Bluetooth. La figura 3.4 se pone con el fin de enseñar la posición en la que debe estar colocada con relación al resto de las capas del protocolo.

La capa de RFCOMM es un puerto serial de emulación que permite heredar las aplicaciones sobre los links Bluetooth. Las TCS son capas de control de telefonía y capas de señalización para aplicaciones avanzadas de telefonía. La DSP es una capa de servicio

que permite, a los dispositivos Bluetooth, pedir a otros dispositivos servicios que ellos pueden proveer.

3.6 Capa Física (PHY).

La figura 3.5 indica la relación entre el protocolo Bluetooth de pila y la PHY. La PHY es la primera de 7 capas del modelo OSI y es responsable de la transmisión de bits entre sistemas adyacentes sobre un canal aéreo. La descripción de esta capa se limita a los siguientes puntos:

- Recepción de una trama de bits de la subcapa MAC y la transmisión de una trama de bits vía ondas de radio hacia una estación asociada.
- Recepción de ondas de radio de alguna estación asociada y la conversión de estas a una trama de bits que se transmite hacia la MAC.

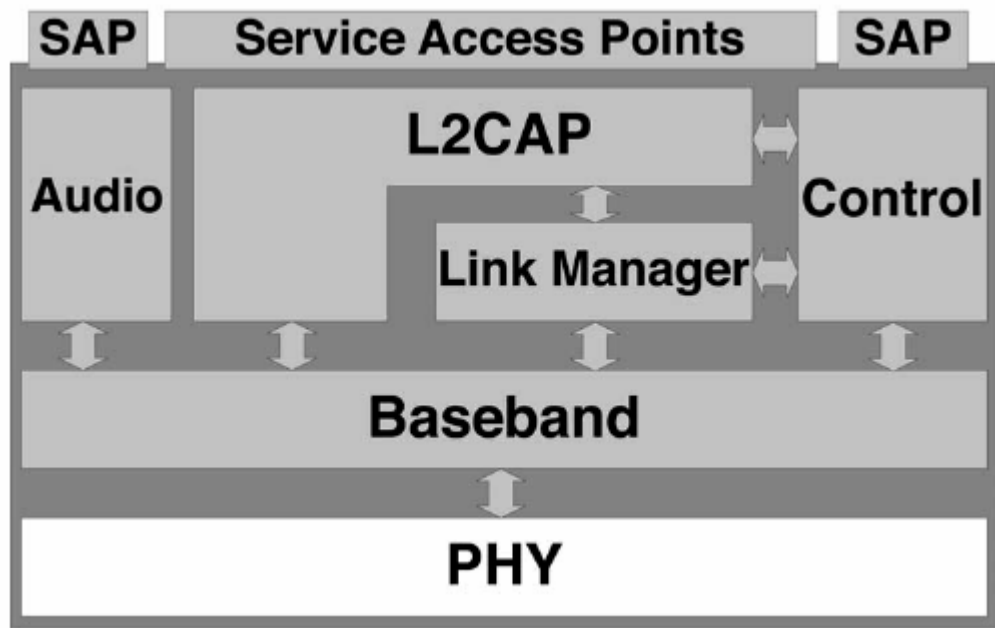


Figura 3.5 - Relación de la PHY y su interfase [1].

3.7 Arreglo de bandas y canales en frecuencia.

Los sistemas Bluetooth operan en la banda ISM (industrial, scientific, medical) de los 2.4 GHz. En la mayoría de los países alrededor del mundo, el rango de dichas frecuencias va de los 2400 MHz a los 2483.5 MHz. Sin embargo, en algunos países, tienen limitaciones nacionales en ese rango de frecuencias. Para cumplir con dichas limitaciones internacionales, se han diseñado algoritmos especiales de frequency-hopping para dichos países. Se debe tener en cuenta que los productos que implementan reducciones de banda no trabajaran con los productos que implementan la banda completa de frecuencias. Los productos que implementan bandas reducidas de frecuencias deben, por lo tanto, considerarse como versiones locales para un solo tipo de mercado (ver tabla 3.1).

Lugar	Rango de Frecuencias (GHz)	Canales de RF
USA, Europa, mayoría de los países.	2.400 – 2.4835	$f = 2402 + k \text{ MHz}, k = 0, \dots, 78$
Francia	2.4465 – 2.4835	$f = 2454 + k \text{ MHz}, k = 0, \dots, 22$

Tabla 3.1, Frecuencia de operación de operación.

3.8 Características del transmisor.

Los requerimientos en esta cláusula especifican los niveles de energía en el conector de la antena del equipo. Si el equipo no tiene un conector, se asume una antena de referencia con una ganancia de 0 dB.

Debido a que es complicado obtener una medida precisa en las mediciones en medidas de radiación, se prefiere que los sistemas con una antena integrada tengan un conector temporal de antena durante las pruebas.

Los equipos se dividen en tres clases de grupos dependiendo su nivel de potencia, dichos grupos se pueden apreciar en la tabla 3.2.

Power class	Maximum output power (P_{\max})	Nominal output power	Minimum output power ^a	Power control
1	100 mW (20 dBm)	N/A	1 mW (0 dBm)	$P_{\min} < +4 \text{ dBm to } P_{\max}$ Optional: P_{\min}^b to P_{\max}
2	2.5 mW (4 dBm)	1 mW (0 dBm)	0.25 mW (-6 dBm)	Optional: P_{\min}^b to P_{\max}
3	1 mW (0 dBm)	N/A	N/A	Optional: P_{\min}^b to P_{\max}

Tabla 3.2 - Tipos de energía [1].

^aEnergía mínima de salida en el ajuste máximo de energía.

^bLímite mínimo sugerido de energía $P_{\min} < -30\text{dBm}$.

Se requiere de control de energía para equipos que se encuentran en la clase 1. El control de energía se utiliza para limitar la transmisión de energía sobre los 0 dBm. Los controles de energía bajo los 0 dBm son opcionales y se pueden utilizar para optimizar el consumo de energía y reducir el total de interferencia. Los niveles de energía deben formar una secuencia monótona con un nivel máximo de 8 dB y un mínimo de 2 dB. Los equipos que pertenecen a la clase 1 que transmiten un máximo de energía de +20 dB deben de ser capaces de mantener la energía de transmisión por debajo de los 4 dBm.

Equipos con capacidades de control de energía optimizan la salida de energía en un link con comandos LMP. Esto se hace al medir el RSSI (receiver signal strength indication) y reportar si se debe incrementar o disminuir el nivel de energía.

3.8.1 Características de modulación.

La modulación es del tipo GFSK (Gaussian frequency shift keying) con un tiempo de ancho de banda (BT) = 0.5. El índice de modulación debe ser de entre 0.28 y 0.35. Un 1 binario se representa con una desviación positiva de frecuencia, y un 0 binario se representa con una desviación negativa de frecuencia. La sincronización de símbolo debe ser mejor que ± 20 ppm (ver figura 3.6).

Para cada canal de transmisión, la desviación mínima de frecuencia (F_{\min} = la menor de $\{F_{\min+}, F_{\min-}\}$) que corresponde a la secuencia 1010 debe de ser menor a $\pm 80\%$

de la desviación de frecuencia (f_d) que corresponde a la secuencia 00001111. Adicionalmente, la desviación mínima nunca debe de ser menor a 115 kHz. La transmisión de datos tiene una tasa de 1 Msímbolos/s.

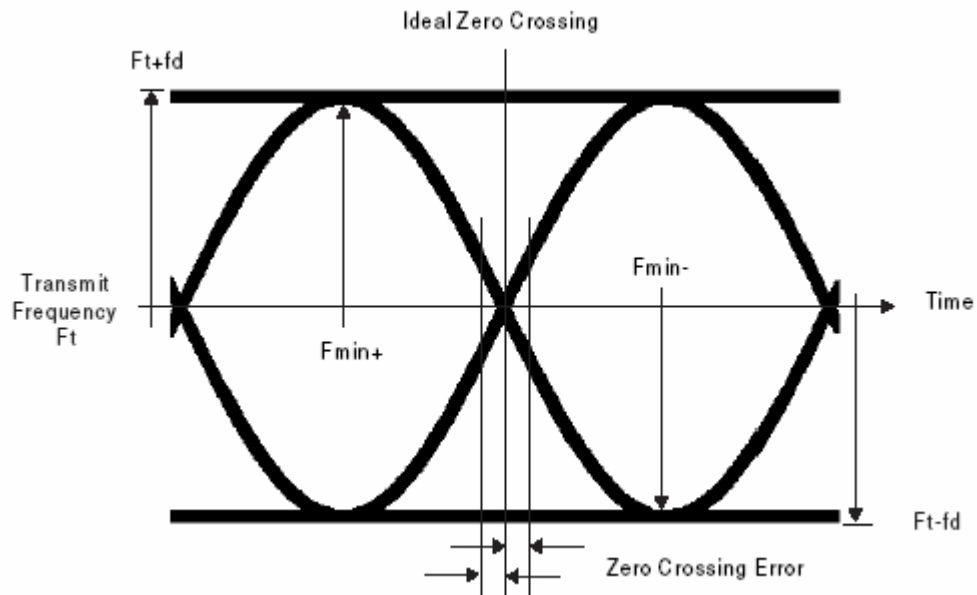


Figura 3.6 – Tipo de modulación para la transmisión de datos[1].

El error de cruce por cero es la diferencia de tiempo entre el periodo ideal del símbolo y el tiempo de cruce real. Este debe de ser menor a ± 0.125 del periodo de un símbolo. La desviación máxima de frecuencia debe de ser entre 140 kHz y 175 kHz.

3.8.2 Tolerancia de RF.

La exactitud de la frecuencia central inicial (F_c) debe ser de ± 75 kHz como máximo para F_c . La exactitud de la frecuencia inicial se define como la exactitud de

frecuencia antes de que cualquier información sea transmitida.

La derivación de frecuencia central se especifica en la tabla 3.3.

Tipo de paquete.	Derivación de Frecuencia.
Paquete con una sola ranura	± 25 kHz
Paquete con tres ranuras	± 40 kHz
Paquete de cinco ranuras	± 40 kHz
Máximo rango de derivación	400 Hz/ μ s

Tabla 3.3 - derivación de frecuencia en paquete.

^aLa máxima derivación de rango que se permite en cualquier paquete.

3.9 Características de recepción.

Para medir el desempeño de la tasa de errores de bit, el equipo debe tener algún medio de retroalimentación. El equipo manda la información decodificada de regreso.

El nivel de referencia de sensibilidad debe estar referido a -70 dBm.

3.9.1 Niveles actuales de sensibilidad.

Los niveles actuales de sensibilidad se define como los niveles de entrada para los cuales el BER (bit error rate) se consideran de un 0.1%. Los requerimientos de un

receptor Bluetooth es un nivel de sensibilidad de -70 dBm o mejor. El receptor debe de tener un nivel sensibilidad de -70 dBm con cualquier transmisor Bluetooth.

3.9.2 Desempeño con interferencia.

El desempeño con interferencia co-canal y adyacente a 1 MHz y un 2 MHz y se mide con la señal deseada a 10 dB sobre el nivel de referencia. En todas las demás frecuencias, la señal deseada debe ser de 3 dB sobre el nivel de sensibilidad de referencia. El BER debe de ser ≤ 0.1 %.

3.10 Emisiones falsas.

Las emisiones falsas para un receptor Bluetooth no deben ser mayores a lo indicado en la tabla 3.4.

Banda de Frecuencia (GHz)	Requerimientos (dBm)
0.030 – 1.000	-57
1.000 – 12.750	-47

Tabla 3.4 – Bandas.

La energía se debe medir en un ancho de banda de 100 kHz.

3.11 Especificaciones de banda base.

La figura 3.7 indica la relación entre la pila del protocolo Bluetooth y esta cláusula. Esta cláusula describe las especificaciones del control de link que transporta los protocolos y las demás rutinas de menor nivel.

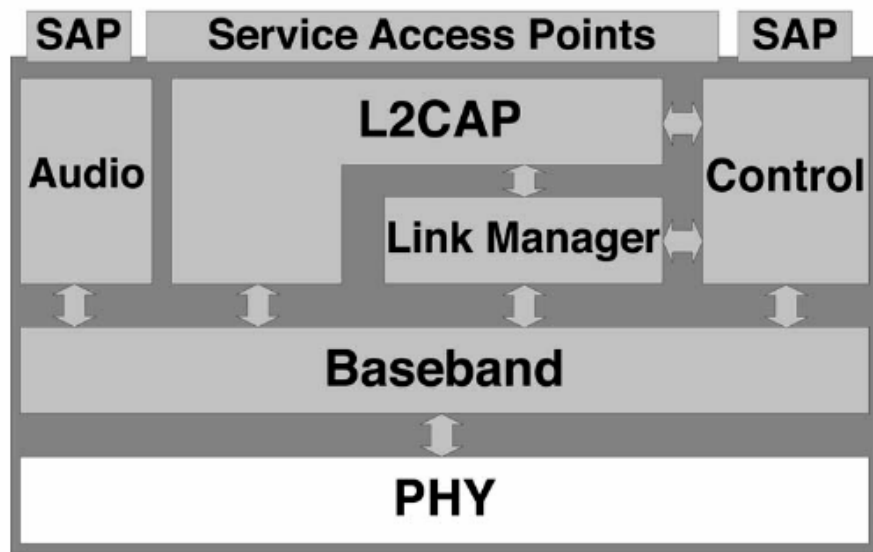


Figura 3.7 – Relaciones e interfaces de la BB [1].

3.11.1 Descripción General.

Bluetooth es un enlace de radio de corto alcance que pretende remplazar conexiones por cable(s) de dispositivos electrónicos portátiles o fijos. Sus principales características son; baja complejidad, bajo consumo de energía, bajos costos además de ser un dispositivo robusto.

Bluetooth opera en la banda libre ISM de los 2.4 GHz. Para evitar la interferencia

y la pérdida de información se utiliza un transmisor-receptor de frequency hop (salto de frecuencia). Para minimizar la complejidad del transceptor se utiliza una modulación binaria de FM. La tasa de transferencia es de 1 Msymbol/s. Se aplica un canal ranurado con una duración estándar de 625 μ s por cada slot (ranura) de tiempo. Para emular una transmisión full duplex, se utiliza una trama de TDD (time division duplex). En el canal, la información se intercambia por medio de paquetes. Bluetooth puede soportar un canal de datos asíncronos, hasta tres canales de voz síncronos simultáneamente, o un canal capaz de manejar simultáneamente datos asíncronos y voz síncrona. Cada canal de voz soporta 64 kb/s de datos síncronos (voz) en cada dirección. El canal asíncrono puede soportar un máximo de 723.2 kb/s asimétricos o 433.9 kb/s.

El sistema Bluetooth consiste de una unidad de radio, una unidad de control de link y una unidad de soporte para el manejo de las funciones y de la terminal de servicio de la interfase (ver figura 3.8). Esta cláusula describe las especificaciones del controlador de enlace Bluetooth, el cual transporta los protocolos de banda base y otras rutinas de bajo nivel.

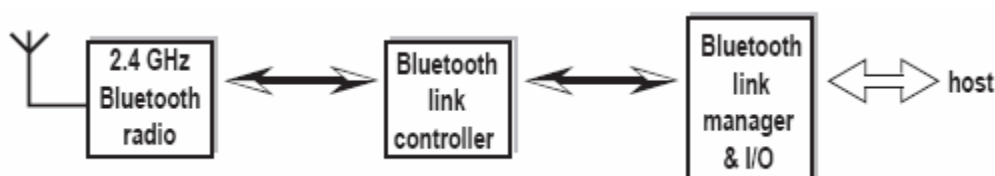


Figura 3.8 – Distintos bloques funcionales para un sistema Bluetooth [1].

Los sistemas Bluetooth proporcionan conexión punto a punto (solo se involucran dos unidades Bluetooth), o una conexión punto a multipunto (ver figura 3.9). En una

conexión punto a multipunto, el canal se comparte entre varias unidades Bluetooth. Dos o más unidades que comparten el mismo canal forman una picored. Un solo dispositivo Bluetooth trabaja como maestro de la picored, mientras que el(los) otro(s) trabajan como esclavos. Hasta siete esclavos pueden estar activos en una picored. Además, muchos más esclavos pueden estar conectados al dispositivo maestro en un estado inactivo o latente. Dichos esclavos latentes no se pueden estar activos en el canal, pero se mantienen sincronizados al maestro. Tanto para los esclavos activos como para los latentes, el canal de acceso es controlado por el maestro.

Múltiples picoredes con áreas de cobertura traslapada forman una red dispersa o una scatternet. Cada picored puede tener solamente un maestro. Sin embargo, los esclavos pueden participar en diferentes picoredes bajo una base de time-division multiplex. Además, un maestro en una picored puede ser esclavo en otra picored. Las picoredes no deben de estar sincronizadas en frecuencia. Cada picored tiene su propio canal de salto.

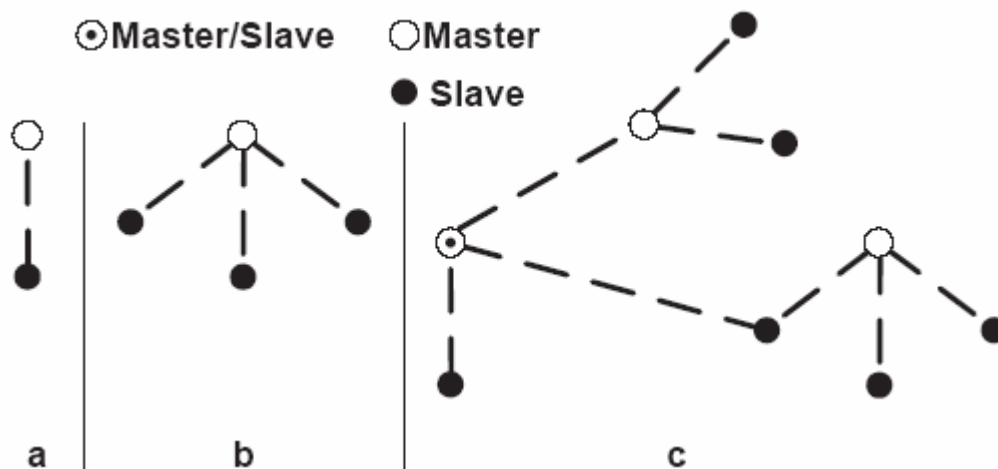


Figura 3.9 – Varias formaciones de picoredes: (a) Operación con un solo esclavo; (b) Operación Multiesclavo; (c) Operación tipo scatternet [1].

3.12 El canal físico.

3.12.1 Definición de canal.

El canal se representa por una secuencia aleatoria de salto a través de 79 o 23 canales de RF. La secuencia de salto es única para cada picored y es determinada por la dirección del dispositivo Bluetooth maestro. La fase en la secuencia de salto es determinada por el reloj del maestro Bluetooth. El canal se divide en rebanadas de tiempo mientras que cada rebanada corresponde a una frecuencia RF de salto. Consecuentemente cada salto corresponde una frecuencia diferente de RF. La tasa nominal de saltos es de 1600 saltos/s. Todas las unidades Bluetooth participantes en la picored están sincronizadas en tiempo y salto dentro del canal.

3.12.2 Ranuras de tiempo.

El canal se divide en ranuras de tiempo, cada una con una duración de 625 μ s. Las ranuras de tiempo se numeran acorde al reloj del dispositivo maestro de las picores. La numeración de las ranuras va de 0 a $2^{27}-1$ y es ciclo con una longitud de ciclo de 2^{27} .

Una secuencia TDD se utiliza para cuando transmite el maestro y el esclavo alternadamente (ver figura 3.10). El maestro debe iniciar su transmisión solo en una ranura numerada como par, y el esclavo debe de iniciar su transmisión en una ranura numerada como impar solamente. El paquete de inicio debe de estar alineado con el

inicio de la ranura. Los paquetes de información deben de extenderse a lo máximo cinco ranuras de tiempo.

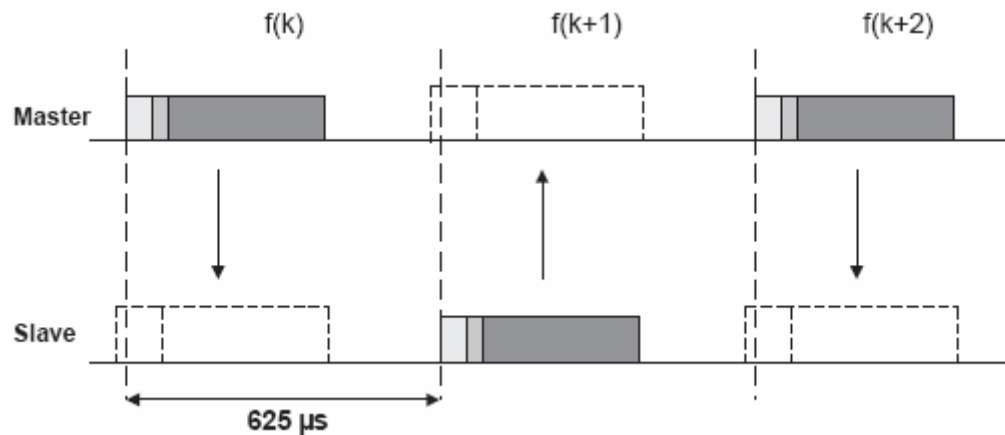


Figura 3.10 – TDD y sus tiempos [1].

La frecuencia RF de salto debe de mantenerse fija durante la duración del paquete. Para un solo paquete, la frecuencia de RF de salto a utilizarse se deriva del valor actual del reloj del dispositivo Bluetooth. Para un paquete con multi-ranura, la frecuencia de RF de salto a utilizarse para el paquete entero se deriva del reloj del dispositivo Bluetooth en la primera ranura del paquete. La frecuencia de salto en la primera ranura después de un paquete de multi-ranuras debe de usar la frecuencia como lo determine el valor actual del reloj Bluetooth. La figura 3.11 ilustra la definición de salto en una sola ranura y paquetes de multi-ranuras. Si un paquete ocupa más de una ranura de tiempo, el salto de frecuencia que se aplica debe de ser la frecuencia de salto aplicada a la ranura de tiempo en donde se inició la transmisión de dicho paquete.

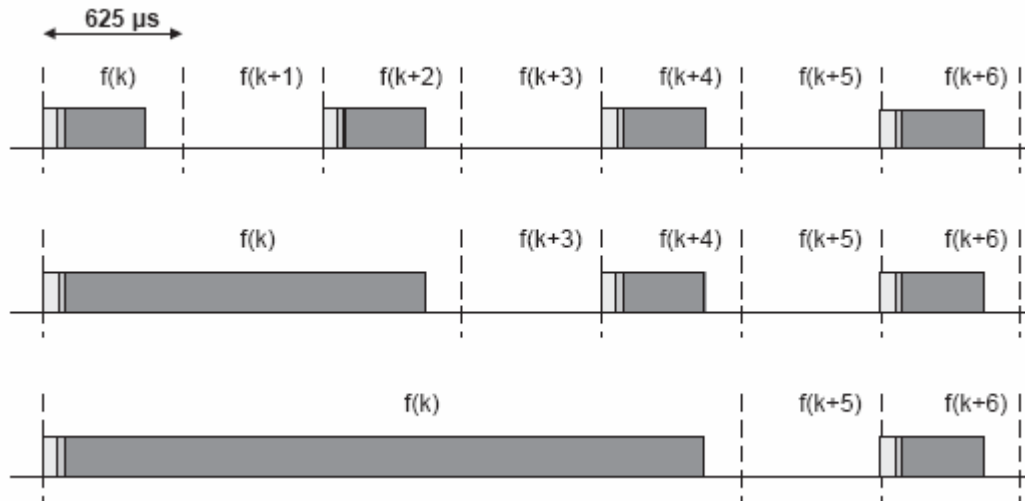


Figura 3.11 – Paquetes multislot [1].

3.13 Links físicos.

Se pueden establecer diferentes tipos de conexión entre maestro y esclavo(s). Se han definido dos tipos diferentes de link:

- Link síncrono de conexión orientada (SCO).
- Link asíncrono de conexión menor (ACL).

El link SCO es un link punto a punto entre el maestro y un esclavo en la picored. El maestro mantiene el link SCO al usar las ranuras reservadas en intervalos regulares. El link ACL es un link punto a multipunto entre el maestro y todos los esclavos que participan en la picored. En las ranuras no representadas por los links SCO, el maestro puede soportar hasta 3 links SCO al mismo tiempo hacia el mismo esclavo o hacia otros.

Un esclavo puede soportar hasta 3 links SCO del mismo maestro, o dos links SCO si los links se originan de diferentes maestros. Los paquetes SCO nunca son retransmitidos.

3.14 Paquetes.

3.14.1 Formato general.

El orden los bits cuando se definen paquetes y mensajes, sigue el formato Little Endian, por ejemplo, las siguientes reglas se aplican:

- El bit menos significativo (LSB) corresponde a b0.
- El LSB es el primer bit que se transmite.
- En las ilustraciones, el LSB se muestra en la parte izquierda.

Los datos en el canal de la picored se transportan en paquetes. El formato general de los paquetes se ilustra en la figura 3.12. Cada paquete consta de tres partes: el código de acceso, el encabezado, y los datos útiles.

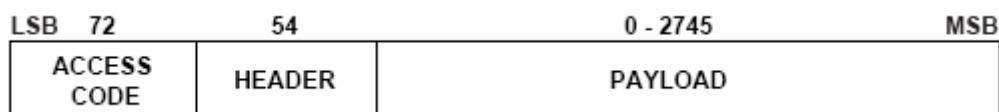


Figura 3.12 – Estándar para el formato del paquete [1].

Tanto el código de acceso como el encabezado son siempre de un tamaño estándar: 72 bits y 54 bits respectivamente. Los datos útiles pueden variar de cero a un máximo de 2745 bits. Se han definido diferentes tipos de paquetes. Los paquetes pueden consistir de códigos puros de acceso, el código de encabezado así como el código de acceso o puede contener los tres tipos de datos.

3.14.2 Código de acceso.

Cada paquete inicia con un código de acceso. Si lo sigue un paquete de encabezado, el código de acceso tiene 72 bits de información, de otra forma el código de acceso tiene 68 bits de información. Este código de información se utiliza para la sincronización, niveles de offset de compensación y como identificación. Los códigos de acceso identifican todos los paquetes de intercambio en el canal de la picored: todos los paquetes que se mandan en la picored son precedidos por el mismo código de acceso al canal. En el receptor de la unidad Bluetooth, un correlator de desfase compara las fases respecto al código de acceso y lo fija cuando detecta que se ha excedido el nivel de referencia.

El código de acceso también se utiliza para los procedimientos de voice. En este caso, el mismo código de acceso se utiliza como mensaje de señalización por lo que no es necesario mandar datos de encabezado o de información extra.

El código de acceso consiste en un preámbulo, en una palabra de sincronización, o información de prueba; ver figura 3.13.

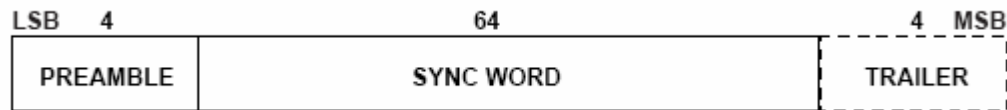


Figura 3.13 – Formato del código de acceso [1].

3.14.2.1 Tipos de códigos de acceso.

Existen tres tipos de códigos de acceso:

- Código de acceso a canal (CAC).
- Código de acceso a dispositivos (DAC).
- Código de acceso a información (IAC).

Cada código de acceso se utiliza respectivamente para diferentes modos de operación en las unidades Bluetooth. El código de acceso a canal identifica a la piconet. Este código se incluye en todos los paquetes de intercambio del canal de la piconet. El código de acceso al dispositivo se utiliza en procedimientos especiales de señalización, por ejemplo, para el voice y como respuesta a dicho voice. Para el código de acceso a información existen dos variantes. Un código general de acceso a información (GIAC) es común a todos los dispositivos. El GIAC puede ser utilizado para ver que otras unidades Bluetooth están en el rango. El código de acceso dedicado a información (DIAC) es común a para un grupo de unidades Bluetooth dedicadas que comparten algunas características. El DIAC se puede utilizar para descubrir solo aquellas unidades dedicadas en dicho rango.

El CAC consiste en un preámbulo, una palabra de sincronía y su longitud total es de 72 bits.

Los diferentes tipos de códigos de acceso utilizan diferentes tipos de LAPs (lower address Parts) para construir palabras de sincronía. En la tabla 3.5 se muestra un resumen de los diferentes tipos de códigos de acceso.

Tipo de Código	LAP	Longitud de Código
CAC	Maestro	72
DAC	Unidad Voceada	68/72
GIAC	Reservado	68/72
DIAC	Dedicado	68/72

Tabla 3.5 – Resumen de códigos de acceso.

3.14.2.2 Preámbulo.

El preámbulo es un patrón fijo cero-uno de cuatro símbolos utilizados para facilitar compensaciones de DC. La secuencia puede ser 1010 o bien 0101, dependiendo de si el LSB (el bit menos significativo), de la siguiente palabra de sincronización es 1 o 0. El preámbulo se muestra en la figura 3.14.



Figura 3.14 - Preámbulo [1].

3.14.2.3 Palabra de sincronía.

La palabra de sincronía es un código de 64 bits derivado de una dirección de 24 bits (LAP); para el CAC se utiliza el LAP del maestro; para el GIAC y el DIAC, se utilizan LAPs dedicadas; para el DAC, se utiliza la unidad LAP esclava. Una buena correlación entre las palabras de sincronía mejora los procesos de tiempo de sincronización.

3.14.2.4 Trailer.

El trailer es adherido a la palabra de sincronía tan pronto como el encabezado del código de acceso del paquete fluye. Este es el caso típico del CAC, pero el trailer también se utiliza en el DAC y el IAC cuando estos códigos se utilizan en paquetes FHS de intercambio durante procedimientos de respuesta e investigación.

El trailer se fija en un patrón de cero-uno de cuatro símbolos. El trailer junto con los tres MSBs de la palabra de sincronización forman un patrón de unos y ceros, que va alternando, el cual se usa para compensar extensiones de DC. La secuencia del trailer es 1010 o bien 0101 dependiendo del MSB de la palabra de sincronía es 0 o 1, respectivamente. El proceso anterior se ilustra en la figura 3.15.



Figura 3.15 – Código Trailer en CAC cuando la palabra de sincronía es 0 en el MSB (a), y cuando la palabra de sincronización es 1 en el MSB (b) [1].

3.14.3 Encabezado del paquete.

El encabezado contiene información del link de control (LC) y consiste de seis campos.

- AM_ADDR: 3 bits que direccional a los miembros activos.
- TYPE: 4-bits de tipo de código.
- FLOW: 1 bit de control de flujo.
- ARQN: 1 bit indicador de acknowledge.
- SEQN: 1 bit de numero de secuencia.
- HEC: 8 bits de detección de error.

El encabezado total, que incluye el HEC, consiste de 18 bits, ver figura 3.16, y es codificada a un rango de 1/3, resultando en un encabezado de 54 bits. Note que los campos de AM_ADDR y TYPE son mandados con el primer LSB.

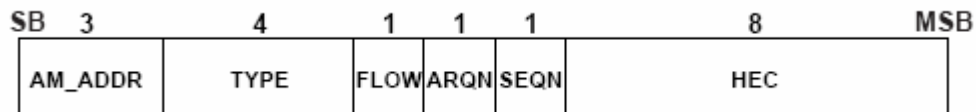


Figura 3.16 – Formato del encabezado [1].

3.14.3.1 AM_ADDR.

El AM_ADDR representa la dirección de un miembro y se utiliza para distinguir entre miembros activos y miembros participantes en la picored. En una picored, uno o más esclavos están conectados a un maestro. Para identificar a un esclavo por separado, cada esclavo es asignado temporalmente a una dirección de 3 bits para ser usada cuando se active. Todos los paquetes que se intercambian entre el maestro y el esclavo transportan el AM_ADDR del esclavo. La dirección de puros ceros es reservada para la transmisión de paquetes del maestro al esclavo, con excepción del paquete de FHS que debe de usar todos los miembros de su dirección, sin embargo no es un mensaje de transmisión.

3.14.3.2 Type.

Se pueden distinguir 16 tipos diferentes de paquetes. El código TYPE de 4-bits especifica que tipo de paquete se utiliza. Es importante notar que la interpretación de los códigos TYPE depende del link físico asociados con el paquete. Primero, se debe de determinar hacia donde se manda el paquete, a un link SCO o a un link ACL. Después se

puede determinar que tipo de paquete SCO o ACL se ha recibido. El código TYPE también revela cuantas ranuras ocupará el paquete.

3.14.3.3 Flow.

Este bit se utiliza para el control de paquetes sobre el link ACL. Cuando el buffer de RX para el link ACL en el recipiente esta lleno, se regresa una señal de alto (FLOW=0) para parar la transmisión de datos. Note que la señal de STOP solo contiene paquetes ACL. Los paquetes que solo contienen información (ID, POLL, y paquetes NULL) o paquetes SCO pueden seguir recibándose. Cuando el buffer de RX esta vacío, se manda una señal GO (FLOW = 1). Cuando no se reciben paquetes, o cuando hay un error en el encabezado, se asume una señal de GO. En este caso, el esclavo puede recibir un paquete nuevo con un CRC a través de su buffer de RX aunque no este vacío. El esclavo debe de regresar un acknowledgment negativo (NAK) en respuesta a este paquete aunque el paquete pase la revisión del CRC.

3.14.3.4 ARQN.

El indicador de acknowledgment de 1-bit ARQN se utiliza para informar de una fuente que ha transmitido exitosamente con un CRC, y puede ser un acknowledge positivo ACK o uno negativo NAK. Si la recepción fue exitosa, se regresa un ACK (ARQN = 1), de lo contrario se regresa un NAK (ARQN = 0). Cuando no se recibe ningún mensaje de

respuesta, se asume un NAK. De la misma manera un NAK es la respuesta que se tiene por default.

El ARQN va en el encabezado del paquete de respuesta. El éxito de la recepción se checa en términos del código de chequeo de redundancia cíclica.

3.14.3.5 SEQN.

El bit SEQN proporciona una trama secuencial de datos para ordenar los datos del paquete. Para cada nueva transmisión de paquetes que contengan datos con CRC, el SEQN bit es invertido. Se requiere que se filtren los datos de la retransmisión en el receptor; si una retransmisión ocurre por una falla del ACK, el destino recibe el mismo paquete dos veces. Por comparación del SEQN de paquetes consecutivos, se puede descartar las retransmisiones por información correcta en la recepción.

3.14.3.6 HEC.

Cada encabezado tiene un HEC (header-error-check) que checa la integridad del encabezado. El HEC consiste de una palabra de 8-bits generada por el polinomio 647 (representación octal). Antes de generar el HEC, el generador HEC se inicializa con un valor de 8-bits.

Después de la inicialización, un HEC es calculado para los 10 bits de encabezado. Antes de revisar el HEC, el receptor debe de inicializar el circuito de revisión con el UAP apropiado de 8-bits. Si el HEC no concuerda, el paquete entero es desechado.

3.14.4 Tipos de paquetes.

Los paquetes usados en las picores están relacionados con los links físicos en los que se utilizan. En el estándar IEEE 802.15.1 se han definido dos links físicos: el link SCO y el ACL. Para cada uno de estos links, se definen 12 tipos de paquetes diferentes. Cuatro paquetes de control son comunes a todos los tipos de links: su código TYPE es único e independiente de cada tipo de link.

Para indicar los diferentes tipos de paquetes en un link, se utiliza el código TYPE de 4-bits. El paquete se ha dividido en cuatro segmentos. El primer segmento está reservado para los cuatro paquetes de control comunes a todos los tipos de links físicos; los cuatro tipos de paquetes. El segundo segmento está reservado para los paquetes que solo ocupan una sola ranura; se han definido seis de estos tipos. El tercer segmento se reserva para paquetes que ocupan tres ranuras, se han definido dos tipos de estos paquetes. El cuarto segmento está reservado para paquetes que ocupan cinco ranuras de tiempo y de estos se han identificado dos tipos de paquetes. La ocupación de segmentemos se refleja en la segmentación y se puede derivar directamente del tipo de código. La tabla 3.6 resume los paquetes definidos por la SCO y los diferentes tipos de link ACL.

Segment	TYPE code $b_3b_2b_1b_0$	Slot occupancy	SCO link	ACL link
1	0000	1	NULL	NULL
	0001	1	POLL	POLL
	0010	1	FHS	FHS
	0011	1	DM1	DM1
2	0100	1	undefined	DH1
	0101	1	HV1	undefined
	0110	1	HV2	undefined
	0111	1	HV3	undefined
	1000	1	DV	undefined
	1001	1	undefined	AUX1
3	1010	3	undefined	DM3
	1011	3	undefined	DH3
	1100	3	undefined	undefined
	1101	3	undefined	undefined
4	1110	5	undefined	DM5
	1111	5	undefined	DH5

Tabla 3.6 – Paquetes definidos por el SCO y links para el ACL [1].

3.15 Corrección de errores.

Hay tres tipos de corrección de error definidos para Bluetooth.

- FEC a una tasa de 1/3.
- FEC a una tasa de 2/3.
- Esquema ARQ para datos.

El propósito del esquema FEC en los datos de carga útil es reducir el número de retransmisiones. Sin embargo, en un ambiente más o menos libre de errores, los FEC proporcionan demasiada información que reduce la velocidad del procesamiento. Los encabezados de los paquetes siempre se protegen por un FEC a un rango de 1/3; este contiene información importante de link y debe ser capaz de manejar más errores de bit.

3.16 Canales Lógicos.

En el sistema Bluetooth, se han definido cinco canales lógicos:

- Canal de control LC.
- Canal de control LM.
- Canal de usuario UA.
- Canal de usuario UI.
- Canal de usuario US.

Los canales de control LC y LM son usados en a nivel de link de control y de manejo respectivamente. Los canales de usuario UA, UI, y US, son usados para transportar datos asíncronos, isócronos e información sincronizada respectivamente. El canal LC se lleva en el encabezado del paquete; los demás canales van en el paquete de la carga de datos. El LM, UA, y el canal UI se indican en el campo de L_CH del encabezado de los datos del usuario. El canal US va sobre el link SCO solamente. Los

canales UA y el UI van normalmente sobre el link ACL; sin embargo, también pueden ir sobre los datos en el paquete DV del link SCO. El canal LM puede ir sobre el link ACL o sobre el SCO.

3.16.1 Canal LC (link de control).

El canal LS se encuentra dentro del encabezado del paquete. Este canal lleva información del link de control de bajo nivel como ARQ, control de flujo, y caracterización de la carga útil de datos. El canal LC va en cada paquete a excepción del paquete ID que no tiene encabezado de paquete.

3.16.2 Canal LM (link de manejo).

El canal de control LM transporta información de control que se intercambia entre el link de administración del maestro y los esclavos. , el canal LM utiliza paquetes protegido DM. El canal LM se indica por el código L_CH en el encabezado de la información del usuario. Para indicar continuidad los paquetes utilizan el código 01 L_CH. Si no hay fragmentación, todos los paquetes utilizan el código de inicio 10.

El canal de datos isócronos trabaja midiendo el tiempo de los paquetes de inicio. Al nivel de banda base, el uso del código L_CH es el mismo que el canal UA.

3.16.3 Canal US (datos síncronos de usuario).

El canal US lleva datos transparentes síncronos del usuario. Este canal va sobre el link SCO.

3.16.4 Localización del canal.

El canal LC se localiza en el encabezado del paquete. Los demás paquetes se localizan en la carga útil de información del usuario. Los canales US solo pueden estar dentro de los paquetes SCO. Los demás paquetes están localizados en los paquetes ACL o en también en el paquete SCO DV. Los canales LM, UA, y UI, pueden interrumpir el canal US si contiene información de mayor prioridad.

3.17 Datos de blanqueo (whitening).

Antes de la transmisión, el encabezado y la carga son mezclados con datos whitening, logrando datos con muchos parámetros de redundancia y reduciendo los paquetes con de DC.

3.18 Rutina de TX.

La rutina de TX va separada de cada link ACL y de cada link SCO. La figura 3.17 muestra los buffers ACL y SCO de la forma en que se utilizan en la rutina de TX. En esta

figura, solo se muestran un buffer de TX ACL y un solo buffer de TX SCO. En el dispositivo maestro, hay un buffer de TX ACL separado del esclavo. Además debe de haber uno o más buffers de TX SCO por cada esclavo SCO (diferentes links SCO pueden ya sea reutilizar el mismo buffer de TX SCO, o cada uno tener su propio buffer de TX SCO). Cada buffer de TX consiste de dos registros FIFO: un registro al que puede leer y tener acceso por el controlador Bluetooth para poder componer los paquetes, y otro registro que puede ser leído por el link Bluetooth de manejo para leer información nueva. La posición de los interruptores S1 y S2 determinan que registro es el actual y cual es el próximo. Los interruptores son controlados por el controlador de link Bluetooth. Los interruptores en la entrada y a la salida de los registros FIFO nunca pueden ser conectados al mismo registro simultáneamente.

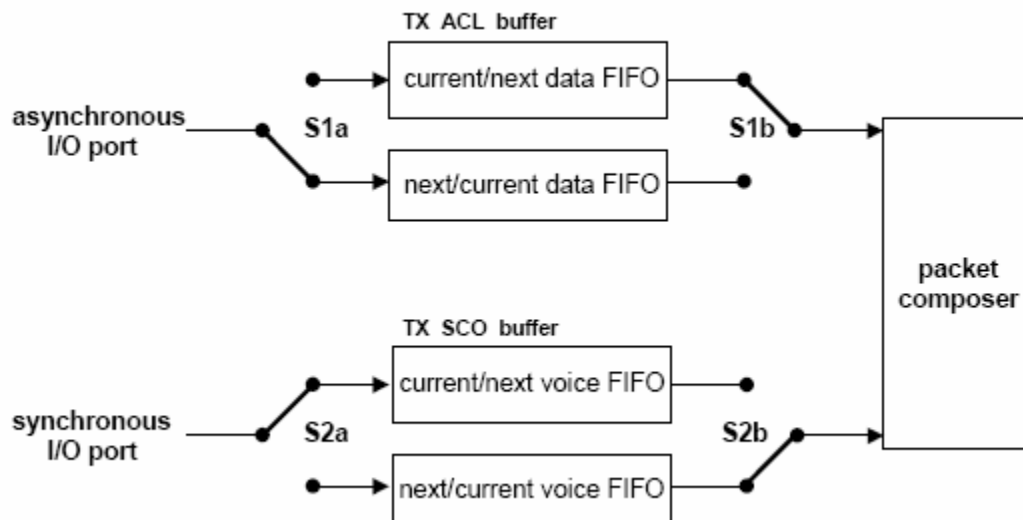


Figura 3.17– Diagrama funcional del buffer de TX [1].

3.18.2 Rutina de RX.

La rutina de RX va separa del link ACL y el link SCO. Sin embargo, en contraste con el buffer ACL maestro de TX, un solo buffer de RX es compartido entre todos los esclavos. Para el buffer SCO, depende en como los diferentes links SCO son distinguidos cuando se requieren extra buffers SCO o no. La figura 3.18 muestra los buffers ACL y SCO de la forma en que se utilizan en la rutina de RX. El buffer ACL de RX consiste de dos registros FIFO: un registro que puede ser leído por el link Bluetooth de control con la carga útil del último paquete RX, y otro registro al que puede tener acceso el link Bluetooth de control para leer la carga anterior de datos útiles. El buffer SCO de RX también consiste de dos registros FIFO: un registro que se llena con información de voz nueva y recién llegada, y otro registro que puede ser leído por la unidad procesadora de voz.

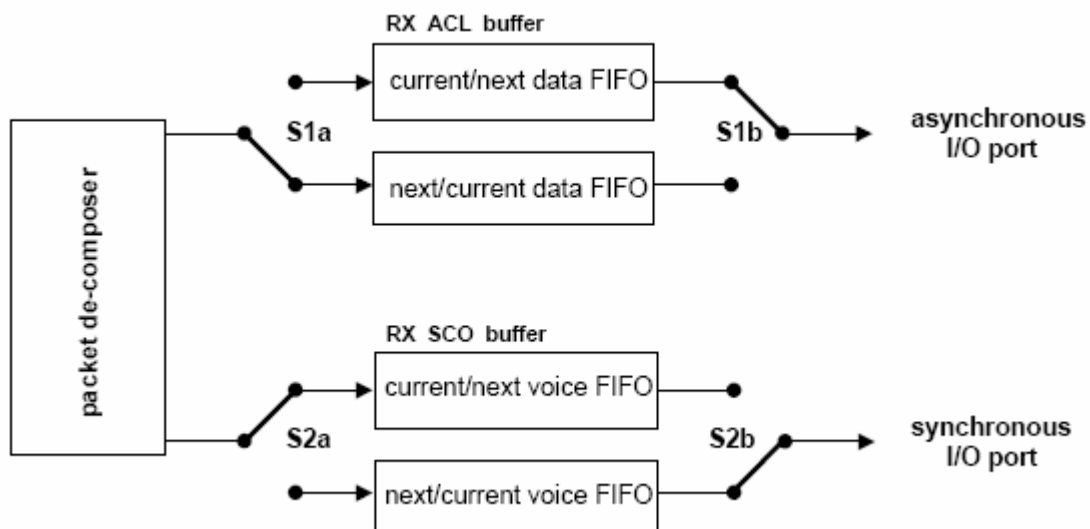


Figura 3.18 – Diagrama funcional del buffer de RX [1].

Dado que el indicador TYPE en el encabezado del paquete recibido indica si la información son datos y/o voz, el de-composer de paquetes puede dirigir

automáticamente el flujo de información hacia buffer apropiados. El interruptor S1 cambia cada vez que el link de manejo ha leído el registro anterior. Si la siguiente carga de información llega antes de que el registro de RX este vacío, una señal de STOP se debe de incluir en el próximo encabezado paquete que es regresado. El indicador de STOP se quita tan pronto como el registro de RX esta vacío.

3.18.3 Control de flujo.

Dado que el buffer de RX ACL puede estar lleno mientras una nueva carga llega, se requiere un control de flujo. Como fue mencionado anteriormente, el campo de FLOW correspondiente al encabezado puede utilizar comandos STOP o GO para detener o reanudar el flujo de los datos transmitidos.

3.18.4 Proceso de trama de bits (Bitstream).

Antes de que la información se mande sobre la interfase aérea, se realizan algunas manipulaciones de bit en el transmisor para incrementar confiabilidad y seguridad. Al encabezado del paquete se le agrega un HEC, los bits del encabezado son mezclados con una palabra de whitening, y se le aplica códigos FEC. En el receptor, se realiza el proceso contrario. La figura 3.19 muestra el proceso que se lleva a cabo en el transmisor y el receptor.

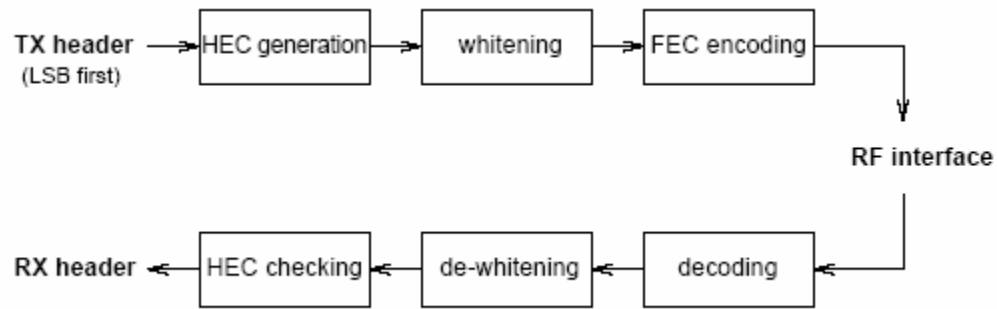


Figura 3.19 – Proceso de TX y RX de los bits de encabezado [1].

Para la carga útil de información, se realiza un proceso similar. Dependiendo del tipo de paquete, es el proceso que lleva la información. La figura 3.20 muestra el proceso que tiene la información del usuario. Además del proceso al que es sometido la carga que contiene información del encabezado, la carga útil de información puede ser sometida a un proceso de encriptación. Solo el proceso de whitening es obligatorio para cada carga de información del usuario; los demás procesos son opcionales y dependen del tipo de paquete. La figura 3.20, ilustra los procesos opcionales con líneas punteadas.

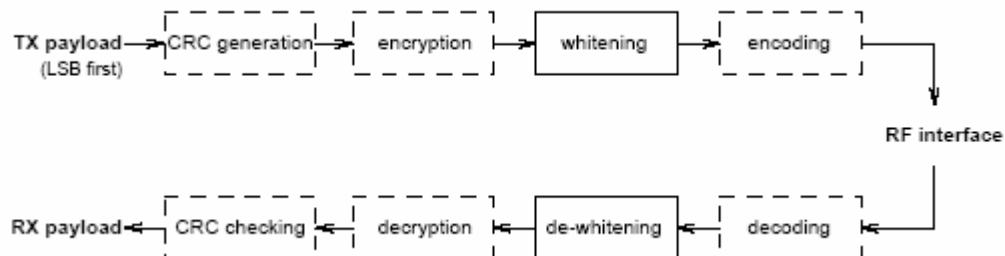


Figura 3.20 – Proceso de TX y RX de los bits de la carga de información [1].

3.19 Canal de control.

A continuación se describe como se establece el canal de una picored y como se pueden agregar y salir unidades a la picored.

3.19.1 Definición maestro-esclavo.

El canal en la picored se caracteriza enteramente por el maestro de la picored. La dirección del dispositivo Bluetooth (BD_ADDR) determina la secuencia de salto FH y el código de acceso al canal; el reloj del sistema del dispositivo maestro determina la fase en la secuencia de salto y determina los tiempos. Además el maestro, controla el tráfico en el canal.

Por definición, el maestro es representado por la unidad Bluetooth que inicia la conexión (a una o más unidades esclavas). Note que los nombres “maestro” y “esclavo” solo son referidas así en el protocolo del canal: las unidades por si mismas son idénticas; por lo que, cualquier unidad puede volverse maestra o esclava de la picored. Una vez que la picored se ha establecido, los roles de maestro-esclavo se pueden intercambiar.

3.19.2 Reloj Bluetooth.

Cada unidad Bluetooth tiene un sistema interno de reloj que determina el tiempo y la secuencia de salto del transmisor-receptor. El reloj del dispositivo Bluetooth se deriva

de un reloj que corre libre que nunca se ajusta y que nunca se apaga. Para sincronizarse con otras unidades, solo se utilizan offsets que al ser agregados al reloj nativo proporcionen relojes temporales Bluetooth que son mutuamente sincronizados. Se debe tener en cuenta que los relojes Bluetooth no tienen relación con la hora y el día; por lo que pueden ser inicializados en cualquier valor. El reloj Bluetooth proporciona los latidos del corazón del transmisor-receptor Bluetooth. Su resolución es de al menos la mitad de la longitud de la ranura del TX o RX, o de $312.5 \mu\text{s}$. El reloj tiene un ciclo de alrededor de un día. Si el reloj se implementa con un contador, se requiere de uno de 28-bit.

El tiempo y la frecuencia de salto en el canal de una picored se determina por el reloj del dispositivo Bluetooth maestro. Cuando se establece la picored, el reloj maestro se comunica a los esclavos. Cada esclavo agrega un offset a su propio reloj para sincronizarse con el reloj maestro. Dado que los relojes corren libremente, los offsets deben de ser actualizados regularmente.

El reloj determina periodos críticos y sujeta los eventos en el receptor Bluetooth. Existen cuatro periodos importantes en el sistema Bluetooth: $312.5 \mu\text{s}$, $625 \mu\text{s}$, 1.25 ms , y 1.28 s ; estos periodos corresponden a los tiempos de bits CLK_0 , CLK_1 , CLK_{12} , respectivamente (ver figura 3.21).

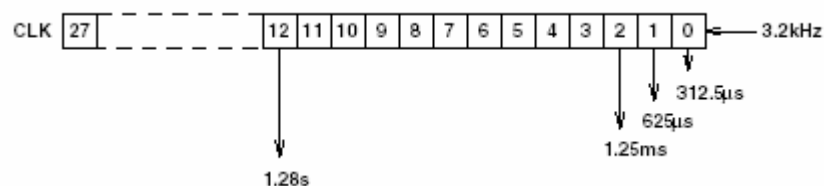


Figura 3.21 – Reloj Bluetooth [1].

En los diferentes modos y estados de una unidad Bluetooth, su reloj puede tener diferentes apariencias:

- CLKN , el reloj nativo.
- CLKE, reloj estimado.
- CLK, reloj maestro.

CLKN es el reloj nativo que corre libremente y es la referencia de las otras formas en las que aparece el reloj.

CLKE y CLK se derivan del CLKN de referencia al agregarse un offset. CLKE se aproxima al reloj del recipiente.

CLK es el reloj del dispositivo maestro de la picored. Se utiliza para todas las actividades de sincronización. Todos los dispositivos Bluetooth utilizan el CLK para sincronizar sus transmisiones y recepciones. El CLK se deriva del reloj nativo CLKN al agregarle un offset (ver figura 3.22). El offset es cero para el dispositivo maestro dado que CLK es idéntico a su reloj nativo. Cada esclavo agrega un offset apropiado a su CLKN. Así el CLK de los esclavos corresponde al CLKN del maestro.



Figura 3.22. Derivación de un CLK maestro (a) y en esclavo (b) [1].

3.19.3 Estados observados.

La figura 3.23 ilustra los diferentes estados utilizados en el control de link Bluetooth. Existen dos estados principales: STANDBY y CONNECTION; además, existen siete subestados, page, page scan, inquiry, inquiry scan, master response, slave response, y inquiry response. Las sub-estados son estados intermedios que se utilizan para agregar esclavos a la picored. Para moverse de un estado a otro, se utilizan ya sea comandos del link administrador Bluetooth, o señales internas en el link de control.

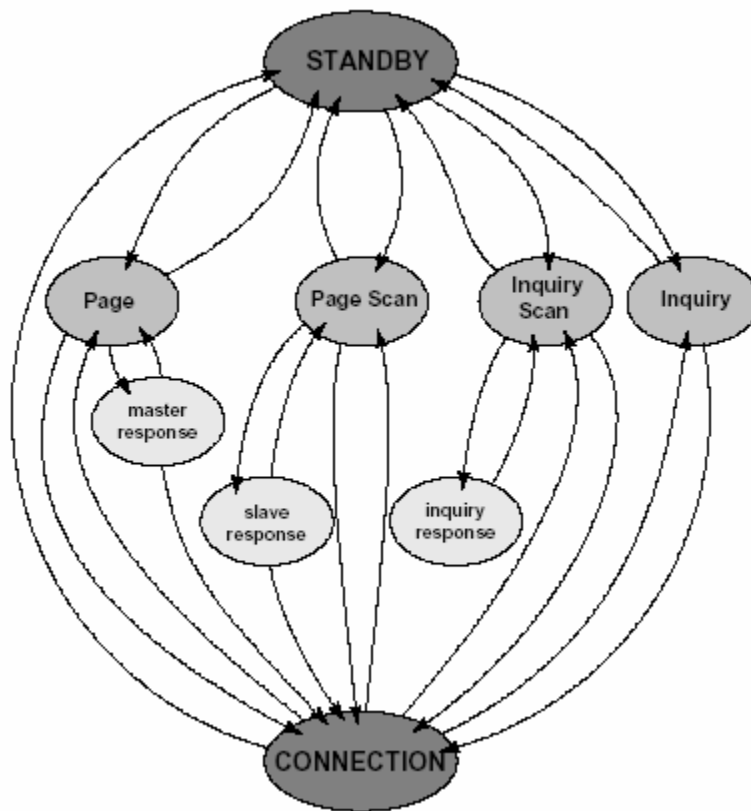


Figura 3.23 – Diagrama de estados Bluetooth [1].

3.19.4 Estado de Standby.

El estado de standby es el estado que tienen por default las unidades Bluetooth. En dicho estado, la unidad Bluetooth esta en un estado de bajo consumo de energía. Solo su propio reloj se encuentra corriendo hasta que ocurra una señal de cambio.

El controlador puede dejar el estado de STANDBY para buscar señales de voiceo, o para voicear el mismo. Cuando responde a un mensaje de voiceo, la unidad no regresa al estado de STANDBY sino entra al estado de CONNECTION como esclavo. Cuando transmite exitosamente un voiceo, la unidad entrará al estado de CONNECTION como unidad maestra.

3.19.5 Redes dispersas.

Es posible que múltiples redes cubran la misma área. Dado que cada picored tiene diferente maestro, las picoredes brincan independientemente, cada una con su propia secuencia de salto de canal y de fase de la forma en que cada maestro lo determine. Además, los paquetes trasportados en cada canal son precedidos por diferentes códigos de acceso a canal determinados por la dirección del dispositivo maestro. En la forma en que más picoredes se agregan, la probabilidad de colisiones se incrementa; por lo que se produce una degradación gradual del desempeño de estos sistemas.

Si muchas picoredes cubren una misma área, una unidad puede participar en dos o más picoredes al aplicar la técnica de multiplexado. Para participar en el canal apropiado, debe de usar la dirección del maestro asociado a dicha red y dar un offset adecuado al reloj para obtener la fase correspondiente. Una unidad Bluetooth puede actuar como

esclavo en varias picoredes, pero solo como maestro en una picored: dado que dos picoredes con el mismo maestro son sincronizadas por el mismo elemento utilizan la misma secuencia de salto, son una misma picored. Un grupo de picoredes en las que las conexiones están dadas entre diferentes picoredes reciben el nombre de redes dispersas (scatternet).

Un maestro o un esclavo puede volverse un esclavo en otra picored al ser voceado por el maestro de la otra picored. Como la unidad de voiceo siempre inicia fuera del dispositivo maestro, se requiere un rol de intercambio maestro-esclavo si se desea que algún dispositivo maestro actúe como esclavo.

El multiplexado en el tiempo debe de usarse para conmutar entre picoredes. En caso de usar solamente links ACL, una unidad puede solicitar entrar a una picored en modo hold o park durante un tiempo en el que debe de unirse a otra picored al solo hacer un cambio de parámetros.

Dado que los relojes de dos dispositivos maestros no están sincronizados, un esclavo que participa en dos picoredes tiene que tener dos offsets agregados a su propio reloj. Como los dos relojes maestros corren independientes, es necesario actualizar los offsets para que la unidad esclava se mantenga sincronizada con los dos maestros.

3.20 Intercambio Maestro-Esclavo.

Hay varias situaciones en las que se desea un cambio maestro-esclavo (MS). Primero, un cambio MS es necesario cuando alguna unidad voicea al maestro de una picored existente y quiere unirse a dicha picored, dado que por definición, la unidad

voceadora (o paging unit) inicialmente es maestra de la picored “pequeña” que solo comprende al pager (maestro) y al paged (esclavo). Segundo, cuando un esclavo en una picored existente quiere iniciar una nueva picored en la que esta sea el dispositivo maestro y la picored maestra en la que se encuentra como esclava. El caso tardío implica un rol doble del maestro original de la picored; este se vuelve esclavo en la nueva picored mientras se sigue manteniendo como dispositivo maestro en la picored original. Tercero, un ejemplo mucho mas complicado es cuando un esclavo quiere tomar completamente una picored existente. Por ejemplo, el cambio también implica la transferencia de otros esclavos de la picored existente a la nueva picored.

3.20.1 Manejo de energía.

3.20.1.1 Manejo de paquetes.

Para minimizar el consumo de energía, en la información en los paquetes se minimiza tanto en la TX y en la RX. Esto significa que si solo se necesita intercambiar información del link de control, se utilizarán paquetes NULL. No se realiza ningún tipo de intercambio de información si no existe un link de control o si existe un comando NAK (los comandos NAK implican que no hay respuesta). Si se necesita mandar datos, la longitud de la carga se adapta para solo mandar los datos de bytes que son validos. En el lado del receptor, el procesamiento de los paquetes se lleva acabo en diferentes etapas. Si se encuentra un código de acceso no valido en la ventana de búsqueda, el transmisor-receptor regresa al estado latente o dormido. Si se encuentra un código de acceso, la unidad receptora se despierta y comienza a procesar el encabezado. Si existe una falla en

el HEC, la entidad regresará al modo latente después de haber procesado el encabezado. Un encabezado indica si sigue información útil y cuantas ranuras de tiempo tiene.

3.20.1.2 Ocupación de la ranura.

Como se indicó anteriormente, el tipo de paquete indica cuantas ranuras ocupa un paquete. Un esclavo al que no se haga referencia en el primer slot puede ir al estado latente durante el resto de los slots que ocupe el paquete. Esto lo puede leer del código TYPE.

3.20.1.3 Supervisión de link.

Una conexión puede fallar por varias razones, por ejemplo, cuando un dispositivo se mueve fuera del rango o cuando hay condiciones de falla de energía. Dado que lo anterior puede pasar sin previo aviso, es importante monitorear el link tanto el maestro como en el esclavo para evitar posibles colisiones cuando el AM_ADDR es reasignado a otro esclavo.

Para ser capaces de supervisar pérdidas de link, tanto el maestro como el esclavo utilizan supervisores de link por contadores de tiempo. Sobre la recepción de un paquete que pasa el chequeo HEC y tiene el AM_ADDR correcto, el contador es iniciado. Si en cualquier momento en el estado de conexión se alcanza los niveles de supervisión llamados supervisionTO, la conexión se reinicia. El mismo valor se utiliza para las conexiones SCO y ACL.

El periodo supervisión, supervisionTO, se determina en los niveles de LM. Este valor es seleccionado de tal manera que el tiempo de supervisión sea más largo que el periodo de conexión. Los links de supervisión de un esclavo estático se realizan al mover y sujetar el estado estático a dicho elemento esclavo.

3.21 Selección de salto.

En total se definen diez tipos diferentes de secuencias de salto: cinco para los sistemas 79-hop y cinco para los sistemas 23-hop, estas secuencias son como sigue:

- Una secuencia de salto con 32 frecuencias únicas distribuidas equitativamente sobre 79 MHz, con un periodo de 32.
- Una secuencia de respuesta que cubre 32 frecuencias únicas que se encuentran a una correspondencia de uno a uno a la secuencia de salto. El maestro y el esclavo utilizan diferentes reglas para obtener la misma secuencia.
- Una secuencia para buscar información con 32 frecuencias únicas para despertar unidades dormidas distribuidas equitativamente sobre los 79 MHz, con un periodo de 32.
- Una secuencia de respuesta a la señal de búsqueda que cubre 32 frecuencias únicas de respuesta que se encuentran en correspondencia uno a uno a la secuencia de salto.

- Un canal de secuencia de saltos que tiene un periodo muy largo, que no muestra parámetros repetitivos sobre un intervalo de tiempo pequeño, pero que distribuye las frecuencias de salto de manera equitativa sobre 79 MHz.

Para la secuencia de salto, es importante que podamos cambiar fácilmente la fase hacia delante o hacia atrás, por lo que necesitamos un mapeo 1-1 de un contador hacia las frecuencias de salto. Para cada caso, es necesario una secuencia de salto del maestro hacia el esclavo y del esclavo hacia el maestro.

3.21.1 Esquema general de selección.

El esquema de selección consiste de dos partes:

- Selección de una secuencia.
- Localizar dicha secuencia en las frecuencias de salto.

El diagrama general en bloque de la selección de salto se muestra en la figura 3.24. La localización de la entrada hacia un salto particular de frecuencia se realiza en la caja de selección. Básicamente, la entrada es el reloj nativo y la dirección actual. En estado de CONNECTION, el reloj nativo (CLKN) es modificado por un offset para igualar al reloj maestro (CLK). Solo el MSBs 27 del reloj se utiliza. En los subastados de página y de investigación, todos los 28 bits del reloj se utilizan. Sin embargo, en el subastado de página el reloj nativo se modificara para que el maestro estime a la unidad esclava.

La dirección de entrada consiste de 28 bits, por ejemplo, la totalidad del LAP y los cuatro LSBs de una UAP. En el estado CONNECTION, se utiliza la dirección del maestro. Cuando esta en el subestado inquiry, correspondiente al GIAC se utiliza el UAP/LAP. La salida constituye una secuencia seudo aleatoria, ya sea cubriendo 79 saltos o 23, dependiendo en del estado en el que se encuentre.

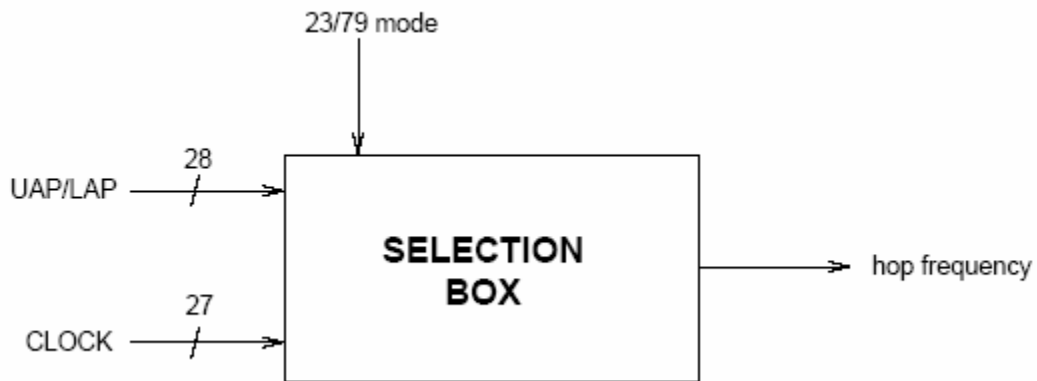


Figura 3.24 – Diagrama general de bloque de una selección de salto [1].

Para el sistema de salto 79, el esquema de selección selecciona un segmento de 32 saltos en frecuencia ocupando 64 MHz haciendo tres saltos una vez en un orden aleatorio. Después un segmento diferente de 32 saltos se escoge. En caso de una pagina, se busca una, o un sub-estado de pagina. En un estado de conexión la salida constituye una secuencia seudo aleatoria que se desliza sobre 79 o 23 saltos, dependiendo del sistema seleccionado de salto. Para sistemas 23-hop, el tamaño del segmento es de 16. Dicho principio es representado en la figura 3.25.

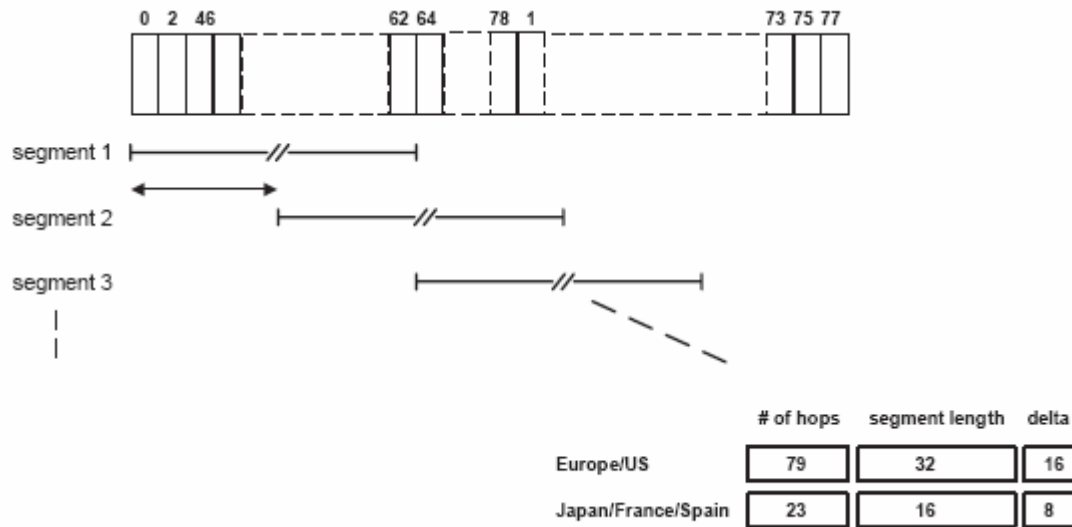


Figura 3.25 – Diagrama general en bloque de un esquema de selección de salto [1].

3.22 Audio Bluetooth.

Sobre la interfase aérea Bluetooth, se utiliza un formato PCM de 64 kb/s, o un CVSD (Continuous Variable Slope Delta Modulation) también de 64 kb/s. Posteriormente se aplica un formato de modulación delta de compresión silábica.

El código de voz en la interfase de línea debe tener una calidad igual o mejor que la calidad que proporciona el código PCM a 64 k/b/s.

La tabla 3.7 resume los esquemas de voz que soporta la interfase aérea. El código apropiado de voz se selecciona después de se realizan las negociaciones entre los links de manejo.

Voice Codecs	
linear	CVSD
8-bit logarithmic	A-law
	μ -law

Tabla 3.7- Esquema del código de voz soportados por la interfase aérea [1].

3.22.1 Código PCM de registro.

Dado que los canales de voz en la interfase aérea soportan tramas de 64 kb/s, se puede utilizar un código de registro PCM para transmisión de tráfico. Se puede aplicar una compresión tipo A-law o μ -law. En el caso en que la interfase del usuario utilice una compresión tipo A-law y el usuario una μ -law y viceversa, se utiliza una conversión A-law a μ -law. Los métodos de compresión utilizan las recomendaciones G.711 de la ITU-T.

3.22.2 Código CVSD.

Un formato más robusto sobre la interfase de voz es la modulación delta. Este esquema de modulación sigue la forma de la onda conforme la señal aumenta o disminuye su valor. La entrada al codificador CVSD es de 64 kmuestras/s de código PCM linear. En las figuras 3.26, 3.27, 3.28 se muestran los diagramas de bloques del codificador CVSD y del decodificador CVSD.

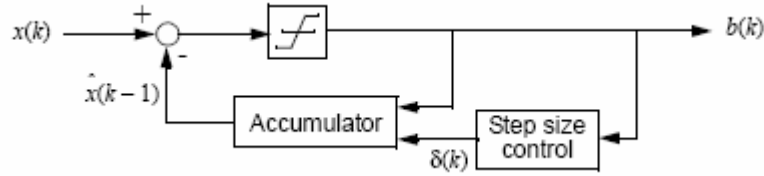


Figura 3.26 - Diagrama a bloques de un codificador CVSD comparación silábica [1].

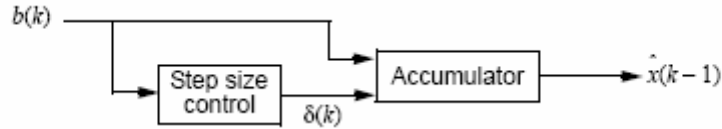


Figura 3.27 - Diagrama a bloques para un decodificador con compresor silábico [1].

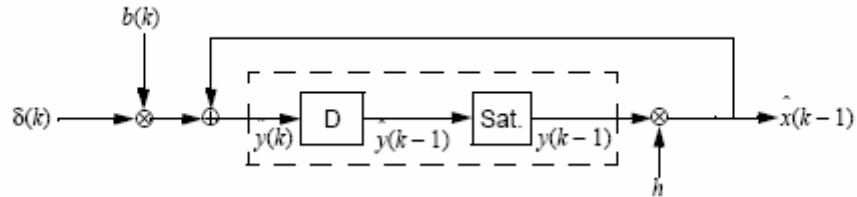


Figura 3.28 - Procedimiento de acumulación [1].

En donde $b(k)$ representa el codificador CVSD, el acumulador por $y(k)$, y la duración del pulso por $\delta(k)$. Además, h representa el factor de acumulación de atenuación, b representa el factor de atenuación del pulso de entrada y α representa el parámetro de comparación silábica.

3.22.3 Calidad de audio para CVSD.

Para tener una calidad de audio apropiada es necesario que las 64 kmuestras/s de código PCM de la señal de entrada tengan una densidad espectral despreciable alrededor de los 4 kHz.

Un punto de referencia para las señales de entrada se debe de codificar en el transmisor y mandarse a un decodificador de referencia. La densidad de energía espectral en la banda de los 4 - 32 kHz de la señal decodificada de salida, debe de estar por debajo del rango máximo unos 20 dB en el rango que va de 0 a 4 kHz.

3.23 Protocolo para el administrador de enlace (Link Manager).

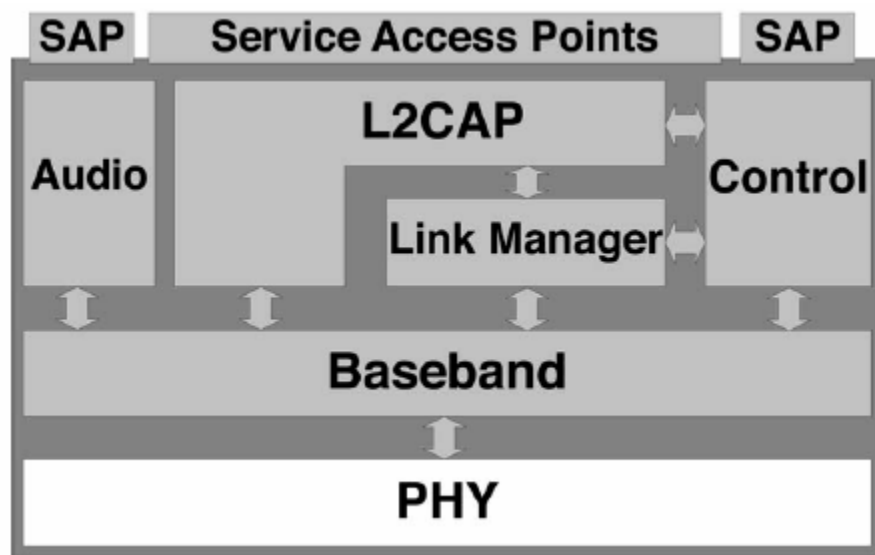


Figura 3.29 – Relación de la interfase para el LM [1].

La figura 3.29 muestra la relación entre la pila del protocolo Bluetooth con esta cláusula. Esta cláusula describe el protocolo de manejo de link o Link Manager (LMP) el cual se utiliza para controlar el link. Las señales son interpretadas y filtradas por el Link Manager en el receptor y no son propagadas hacia capas superiores.

3.23.1 General.

Los mensajes del LMP se utilizan para saber el estado del link, por seguridad y para el control del mismo. Dichos mensajes son transmitidos en la información del

usuario en vez de ser transmitidos por el L2CAP (Logical Link Control and Adaptation Protocol) y son distinguidos por un valor reservado en el campo de L_CH del encabezado de los datos del usuario. Los mensajes son filtrados e interpretados por el LM en el receptor y no se mandan a capas superiores como se ilustra en la figura 3.30.

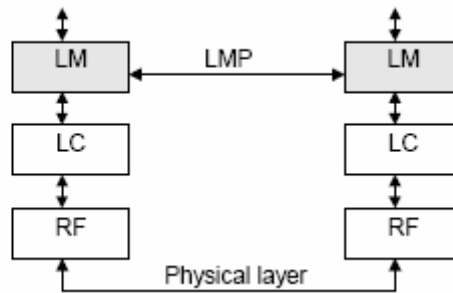


Figura 3.30 – Lugar del LM en la escena global [1].

Los mensajes del Link Manager tienen más prioridad que los datos del usuario. Esto significa que si el Link Manager necesita mandar mensajes, no deben ser retrasados el tráfico del L2CAP, aunque puede ser retrasado por algunas retransmisiones de paquetes de información individual de la banda base.

3.23.2 Formato del LMP.

LM PDUs siempre se manda en paquetes de un solo spot, y el encabezado de la información de la información tiene que ser de 1 byte. Los dos bits menos significativos en el encabezado de la información determinan el canal lógico. Para el LM PDUs esos bits se determinan de la forma en que se especifica en la tabla 3.8:

Código L_CH	Canal Lógico	Información.
00	NA	Indefinido
01	UA/I	Mensaje de continuación del L2CAP
10	UA/I	Mensaje de inicio del L2CAP.
11	Lm	Mensaje LMP

Tabla 3.8 – Contenido de campos en el canal lógico L_CH.

El bit FLOW en el encabezado de la carga siempre es uno y es ignorado en el receptor. Cada PDU (protocol data unit) tiene asignado un código de 7 bits. El ID de transacción es posicionada en el LSB. El opcode y el bit de ID de transacción se localizan en el primer byte del cuerpo de la información. (ver figura 3.31).



Figura 3.31 – Cuerpo de la carga con se manda un LM PDUs [1].

3.23.3 Establecimiento de conexión.

Después del procedimiento de paging (voceo), el dispositivo maestro debe de seguir al esclavo con un intervalo máximo de seguimiento. Los procedimientos determinados en el LMP para solicitar compensaciones de reloj, versiones del LMP, características y servicios soportadas y solicitud de nombres.

Cuando el dispositivo de voiceo desea crear una conexión que envuelve capas alrededor del LM, este manda LMP_host_connection_req. Cuando la contra parte recibe este mensaje, el host es informado sobre la próxima conexión. El dispositivo remoto puede aceptar o rechazar la solicitud de conexión al mandar el LMP_accepted o el LMP_no_accepted. Alternativamente, si el esclavo necesita un cambio maestro-esclavo, este manda el LMP_slot_pffset y el LMP_switch_req después de que halla recibido el LMP_host_connection_req. Cuando el cambio maestro-esclavo se ha realizado exitosamente, el antiguo esclavo responderá con un mensaje de LMP_accepted o con un LMP_not_accepted. La figura 3.32 resume las transacciones del LMP para el establecimiento de una conexión siguiendo el procedimiento de voiceo de banda base.

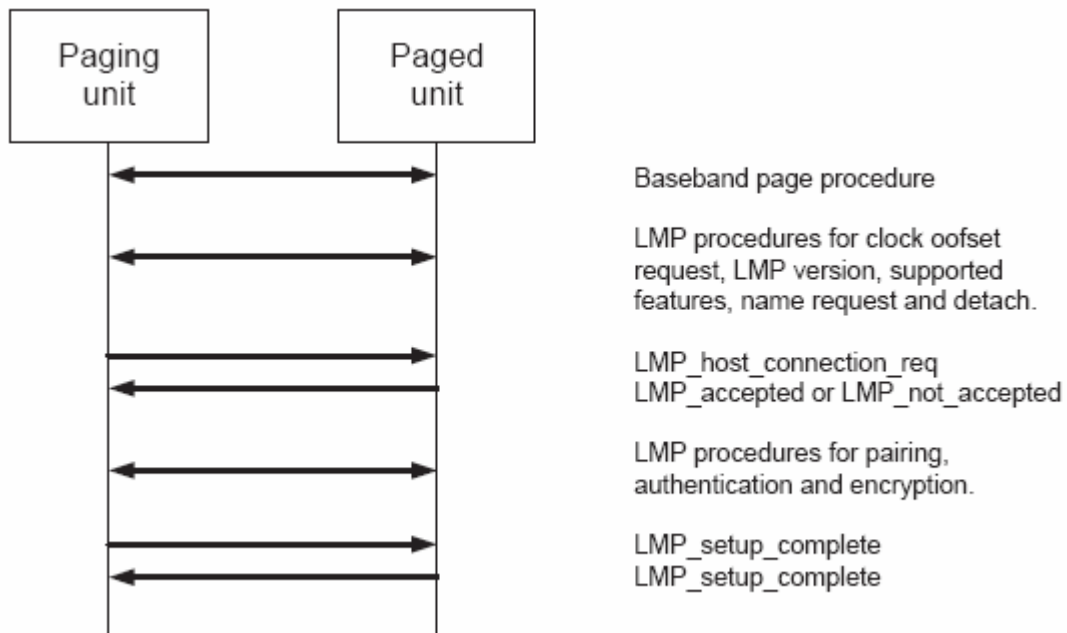


Figura 3.32 – Establecimiento de conexión [1].

3.23.4 Manejo de errores.

Si el Link Manager recibe un PDU con un código desconocido, este responde con el LMP_not_accepted con el mensaje de código desconocido LMP PDU.

Si el Link Manager recibe un PDU con parámetros inválidos, este responde con el LMP_not_accepted, con el mensaje de código con parámetros LMP no válidos.

Mensajes LMP erróneos pueden ser causados por errores en el canal o debido a errores sistemáticos en la parte de transmisión. Para detectar el último caso, el LM debe de monitorear el número de mensajes erróneos y desconectar si este excede el umbral, el cual depende de la implementación.

3.24 Control del enlace lógico y el protocolo de adaptación.

La figura 3.33 indica la relación entre las capas del protocolo Bluetooth con esta cláusula. Esta cláusula describe el control del enlace lógico y el protocolo de adaptación (L2CAP). Este protocolo soporta protocolos de más niveles de multiplexado, segmentación de paquetes, y reensamble. Este describe el protocolo para definir el estado de los dispositivos, formato de los paquetes y composición de los mismos.

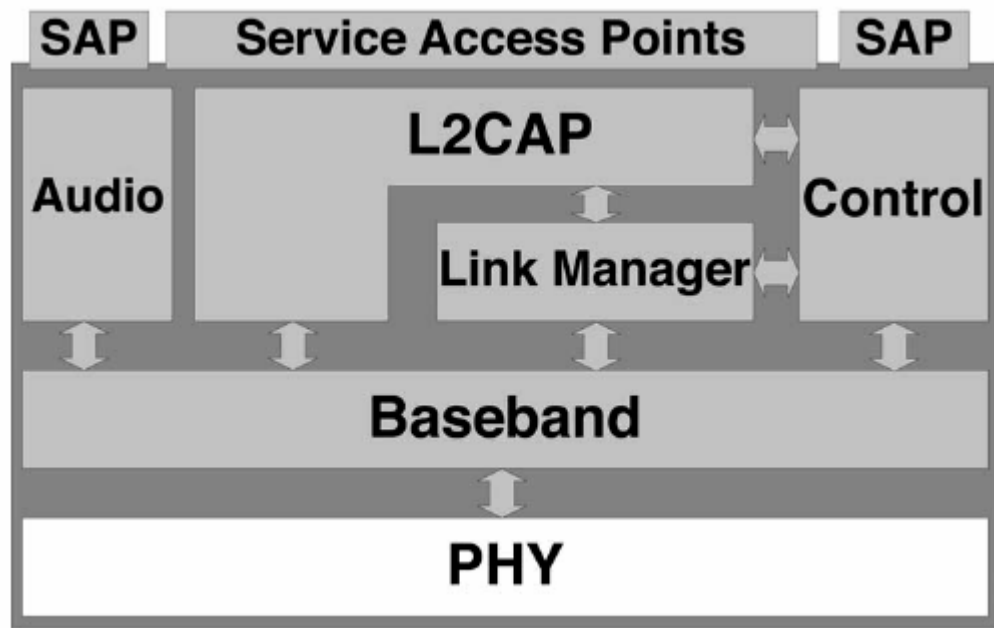


Figura 3.33 – L2CAP relaciones e interfaces [1].

Esta sección define el control del enlace lógico y protocolo de adaptación, referido como L2CAP. L2CAP se encuentra sobre el protocolo de banda base y reside en la capa de enlace de datos como se muestra en la figura 3.34. L2CAP proporciona conexión orientada y servicios de desconexión de datos de servicios de capas superiores. L2CAP permite transmitir y recibir a las capas superiores y aplicaciones. Los paquetes de datos del L2CAP tienen una longitud de 64 kbytes.

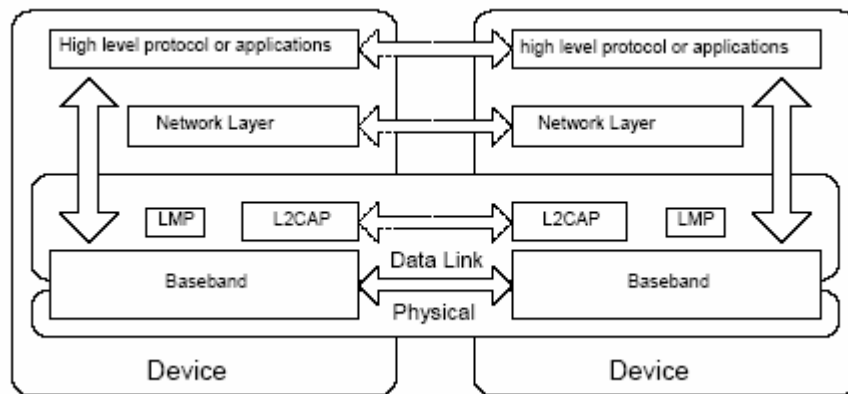


Figura 3.34 – L2CAP en la capa de protocolos [1].

3.24.1 Operación general.

El L2CAP se basa en el concepto de “canales”. Cada uno de los puntos finales de los canales del L2CAP se define como el identificador de canales.

3.24.2 identificador de canales.

Los identificadores de canales (CIDs) son nombres locales que representan canales lógicos de tipo end-point en el dispositivo. Identificadores del 0x0001 al 0x003F son reservados para funciones específicas del L2CAP. El identificador nulo (0x000) está definido como un identificador ilegal y nunca se debe de usar como un destino final. Las implementaciones son libres de manejar el resto de los CIDs en la manera que mejor trabaje en una implementación en particular, con la condición de que el mismo CID no se reutilice como canal local L2CAP de punto final para múltiples y simultáneos canales

L2CAP entre un dispositivo local y alguno remoto. La tabla 3.9 resume las definiciones y la participación del nombre del espacio CID.

CID	Descripción
0x0000	Identificador nulo
0x0001	Canal de señalización
0x0002	Canal de recepción de desconexiones
0x0003 – 0x003F	Reservado
0x0040 – 0Xffff	Localidades dinámicas

Tabla 3.9 – Definiciones CID.

La asignación de CID es relativa a un dispositivo en particular. Un dispositivo puede asignar CIDs independientemente de otros dispositivos (a menos que este necesite utilizar cualquiera de los reservados CIDs mostrados en la tabla 3.9). Así, aun si el mismo valor CID ha sido asignado a un canal remoto por medio de varios dispositivos remotos conectados al un solo dispositivo local, el dispositivo local puede seguir asociado únicamente con cada CID remoto en un dispositivo diferente.

3.24.3 Operación entre dispositivos.

La figura 3.35 ilustra el uso de CIDs en una comunicación entre entidades correspondientes a un L2CAP en dispositivos separados. Los datos de conexión orientada de canales de datos representan una conexión entre dos dispositivos, en donde un CID

identifica cada punto final de un canal. Los canales sin conexión, restringen el flujo de datos a una sola dirección. Dichos canales se utilizan para soportar un grupo de canales en donde un CID en la fuente representa uno o más dispositivos remotos. También hay un número de CIDs reservados para propósitos especiales. El canal de señalización es un ejemplo de canal reservado. Este canal se utiliza para crear y estabilizar conexiones orientadas en canales de datos y a negociar canales. Es esencial el soporte de un canal de señalización con una entidad L2CAP. También se reserva un CID para todas las conexiones de entrada para datos de tráfico. En el siguiente ejemplo, un CID es utilizado para representar un grupo que consiste del dispositivo #3 y #4. El tráfico se manda del canal ID y es directo al canal remoto reservado para conexiones de tráfico de datos.

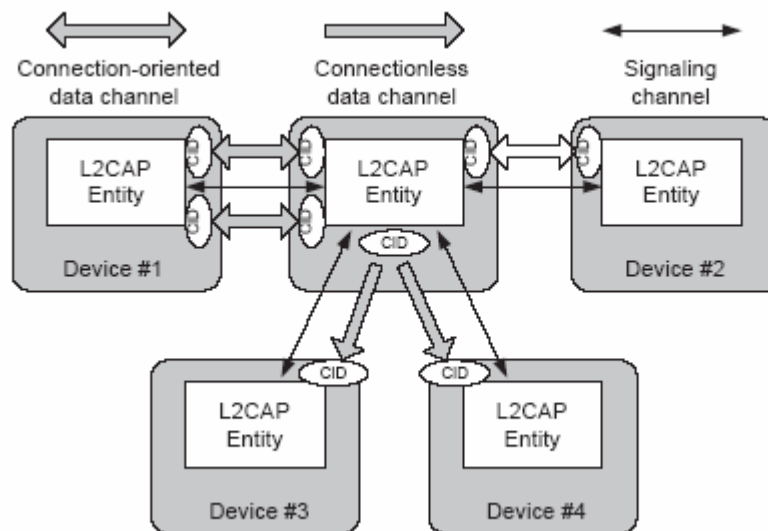


Figura 3.35 – Canales entre dispositivos [1].

Page Break

3.24.3.1 Operación entre capas.

Las implementaciones de L2CAP deben de seguir la arquitectura general descrita en la figura 3.36. Las implementaciones L2CAP deben de transferir datos entre capas superiores del protocolo y capas inferiores. Cada implementación debe también de soportar un número de comandos de señalización. Las implementaciones L2CAP deben de estar preparadas para aceptar ciertos tipos de eventos de capas inferiores y generar eventos en capas superiores. La forma en que estos eventos pasan entre las capas es un proceso que depende de la implementación.

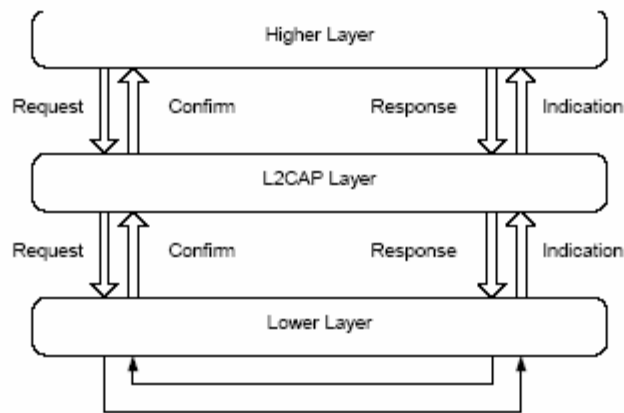


Figura 3.36 – Arquitectura L2CAP [1].

3.24.3.2 Segmentación y reensamble.

Las operaciones de SAR se utilizan para mejorar la eficiencia al permitir un máximo de transferencias de unidades (MUT), más largas que los paquetes de banda base. Esto reduce los encabezados al separar la red y transportar paquetes utilizados por capas superiores del protocolo sobre algunos paquetes de banda base. Todos los paquetes de L2CAP deben de segmentarse sobre la banda base de paquetes. El protocolo no realiza ninguna segmentación ni procesos de reensamble pero el formato de los paquetes permite

la adaptación a frames más pequeños. Una implementación L2CAP expone un MTU de salida y segmenta capas superiores de paquetes en pedazos que pueden pasar por el Link Manager vía el HCI. En el lado del receptor, una implementación L2CAP recibe pedazos del HCI y reensambla aquellos pedazos dentro de paquetes L2CAP utilizando información proveniente del HCI y forma encabezados de paquetes (ver figura 3.37).

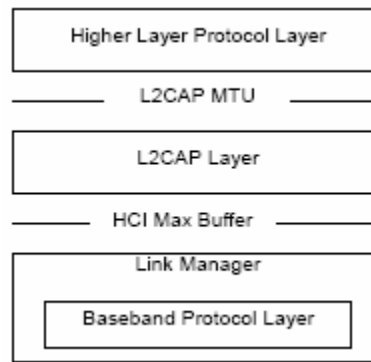


Figura 3.37 – Variables L2CAP SAR [1].

La segmentación y el reensamble es implementado utilizando encabezados muy pequeños en paquetes de banda base. Los dos bits de L_CH definidos en el primer byte de banda base de información payload (también llamado el encabezado de frame) se utilizan para señalar el inicio y continuación de paquetes L2CAP. L_CH debe de ser “10” (por definición del estándar) para la primera segmentación en un paquete L2CAP y “01” para un segmento de continuación. En la figura 3.38 se muestra el uso del SAR.

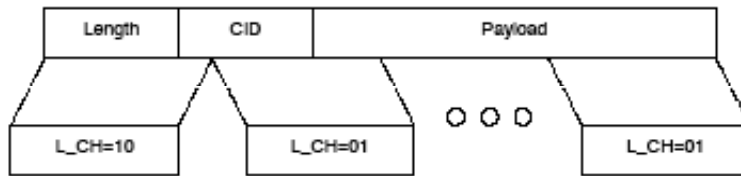


Figura 3.38 – Segmentación del L2CAP [1].

3.24.3.3 Procedimientos de segmentación.

La unidad máxima de transmisión (MUT) para el L2CAP se exportará usando una implementación específica para el servicio de la interfase. Es responsabilidad de la capa superior limitar el tamaño de los paquetes enviados a la capa L2CAP debajo del límite del MTU. Una implementación L2CAP segmentará el paquete dentro de las unidades de datos del protocolo (PDUs) para mandarlas a las capas inferiores. Si el L2CAP corre directamente sobre la banda base del protocolo, una implementación debe de segmentar el paquete dentro de la banda base de paquetes para ser transmitidos al aire. Si L2CAP corre bajo la interfase del controlador (que es lo más común), una implementación debe de mandar dedazos de información de un mismo tamaño hacia el controlador del host en donde estos se convertirán en paquetes de banda base. Todos los segmentos del L2CAP asociados con un paquete L2CAP debe de ser pasados a través de la banda base antes de cualquier otro paquete L2CAP destinado a la misma unidad se mande.

3.24.3.4 Procedimientos de reensamble.

El protocolo de banda base entrega paquetes ACL en secuencia y protege la integridad de los datos utilizando un CRC de 16 bits. La banda base también soporta conexiones confiables al utilizar mecanismos ARQ (automatic repeat request). Cuando el controlador de banda base recibe paquetes ACL, este puede señalar a la capa L2CAP la llegada de cada paquete de banda base, o bien acumular un número de paquetes antes de que el buffer del receptor se llene.

Las implementaciones L2CAP deben de usar el total de la longitud del campo del encabezado de los paquetes L2CAP, sino los paquetes L2CAP que no concuerden con la longitud del campo serán descartados.

La figura 3.39 ilustra el uso de las operaciones de segmentación y reensamble para transmitir un PDU a una capa superior.

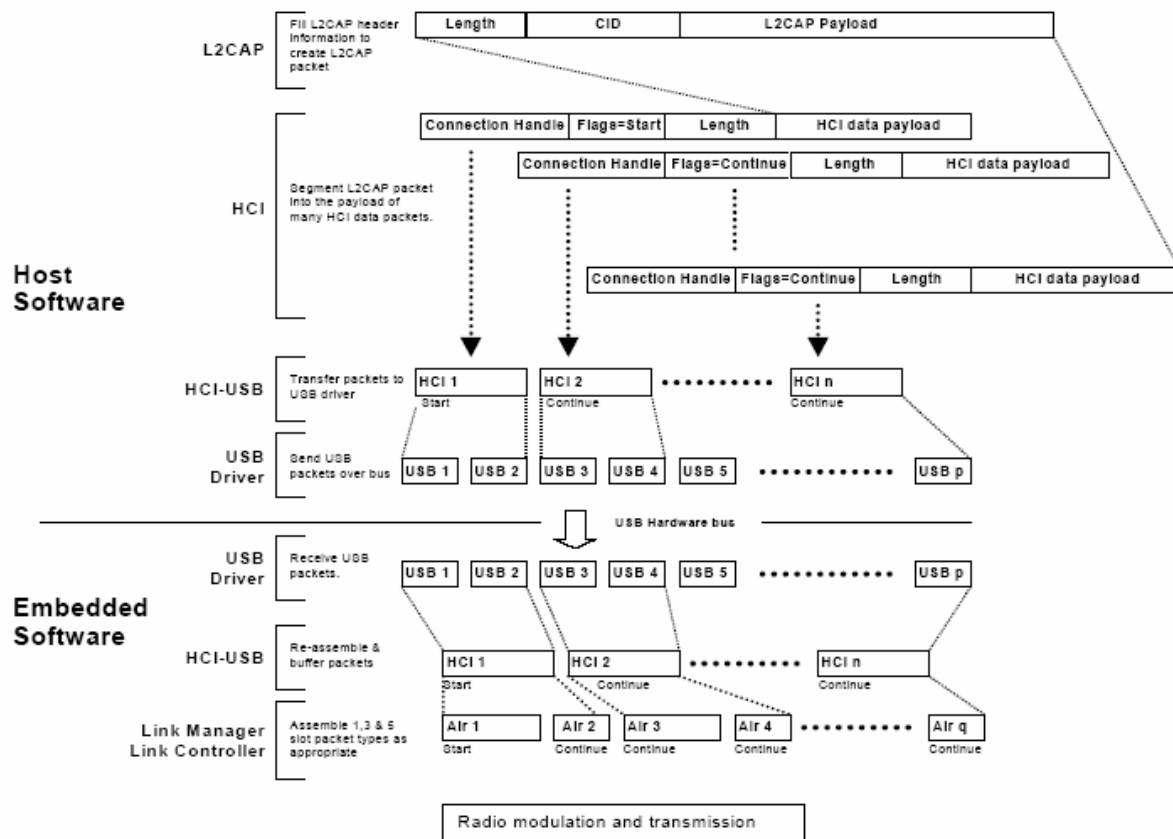


Figura 3.39 – Dispositivos de segmentación y reensamble en una unidad con un HCI [2].

El L2CAP tiene una base para sus paquetes de información que fluye sobre un modelo de comunicación basado en canales. Un canal representa un flujo de datos entre entidades L2CAP en dispositivos remotos. Los canales deben de tener conexiones orientadas o ser desconectadas.

Así el L2CAP es uno de dos protocolos de enlace de diferentes niveles que funcionan sobre la banda base. L2CAP es responsable de los protocolos de multiplexado de nivel superior, abstracción de MTU, manejo de grupo, y de transmitir la calidad de servicios de información hacia el enlace de nivel.

El protocolo puede soportar el multiplexado al definir canales. Cada canal está limitado a un protocolo en un modo de “muchos a uno”. Se puede limitar múltiples

canales al mismo protocolo, pero un canal no se puede limitar a múltiples protocolos. Cada paquete L2CAP recibido en un canal se manda directamente hacia un nivel superior apropiado del protocolo.

El grupo de administración proporciona la abstracción de un grupo de unidades permitiendo un mapeo más eficiente entre grupos y miembros de una piconet con dispositivos Bluetooth.

L2CAP converge información del QoS entre canales y proporciona algún control de admisión para prevenir que otros canales adicionales violen los niveles de QoS existentes.

3.25 Interfase de control.

La figura 3.40 indica la relación del protocolo Bluetooth con esta cláusula. Esta cláusula describe las especificaciones para la funcionalidad de la interfase de control para el IEEE Std 802.15.1. Esta se basa en la sección de HCI de las especificaciones Bluetooth. El término interfase de Control se eligió para hacer claro el origen del material.

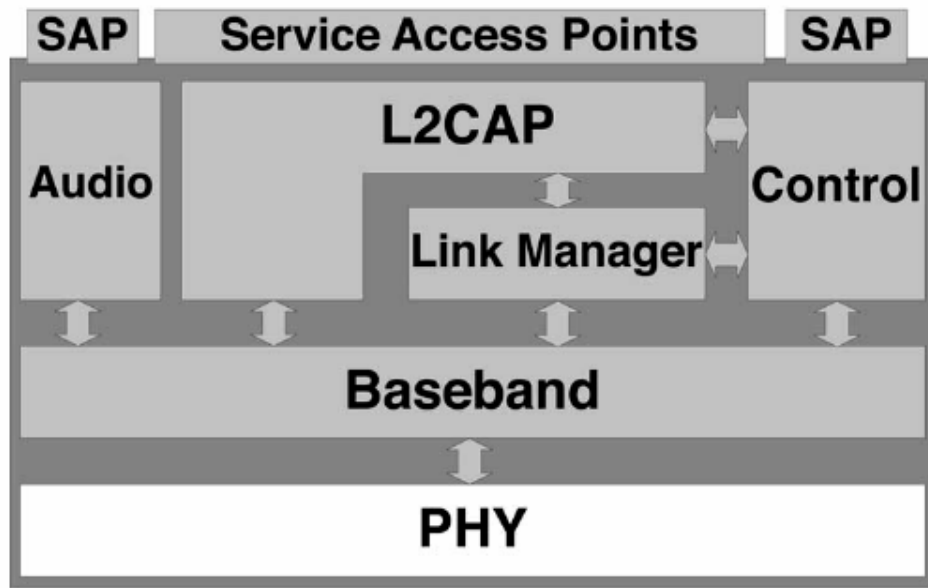


Figura 3.40– Relación de la interfase de control [1].

3.25.1 Introducción IEEE.

El HCI (host controller interface) proporciona un comando de interfaz hacia el controlador de banda base y al link manager, y acceso al estatus del hardware y al control del registro. Esta interfaz proporciona un método uniforme de acceso a la banda base Bluetooth. La sección de HCI tiene dos funciones en la especificación Bluetooth.

1. Define las bases para una interfaz física con un modulo externo Bluetooth.
2. Define las funciones necesarias de control para todas las implementaciones Bluetooth.

El Host recibe notificaciones asíncronas de eventos HCI independientemente de la capa de transporte de control que se utilice. Los eventos HCI se utilizan para notificar al Host cuando algo ocurre. Cuando el Host descubre que ha ocurrido algún evento, éste analizará el paquete de evento o acción recibido para determinar que evento ocurrió.

3.25.2 Los comandos HCI (Host Controller interface).

El HCI proporciona un método uniforme de comandos para el acceso a las capacidades del hardware Bluetooth. Los comandos de enlace del HCI proporcionan un servidor con la habilidad de controlar las conexiones de las capas de enlace hacia otros dispositivos Bluetooth. Dichos comandos típicamente envuelven al LM para el intercambio de comandos LMP con dispositivos Bluetooth remotos.

Los comandos HCI “policía” son utilizados para modificar el comportamiento del LM local o remoto. Dichos comandos proporcionan al servidor, métodos para influenciar la manera en que el LM maneja la piconet. El controlador del servidor y de banda base, los comandos de información y de estado proporcionan al servidor acceso a varios registros del controlador del servidor.

Los comandos HCI tardan diversos tiempos para que se transmitan completamente. Por lo tanto, el resultado de los comandos se reportarán de regreso con el servidor en la forma de un evento. Por ejemplo, para la mayoría de los comandos el controlador del servidor generará los códigos que indiquen que un comando ha sido completado correctamente. Este evento contiene los parámetros de regreso para que un comando HCI sea completado. Para habilitar a los servidores la capacidad de detectar

errores en la capa de transporte HCI, es necesario que haya un tiempo fuera o muerto entre la transmisión de los comandos del servidor y la recepción del controlador del servidor. Dado que el tiempo máximo que puede estar fuera la transmisión, esta está fuertemente ligada a la capa de transporte del HCI utilizada y por lo tanto es recomendable usar un valor predeterminado de 1s. La cantidad de tiempo también es dependiente del número de comandos que no han sido procesados.

3.25.2.1 Intercambio de información específica de HCI.

La capa del controlador de transporte del servidor proporciona un intercambio transparente de información específica del HCI. Estos mecanismos de transporte permiten al servidor mandar comandos HCI, datos ACL y datos SCO al controlador del servidor. Dichos mecanismos de transporte también le permiten al servidor recibir eventos HCI, datos ACL y datos SCO del controlador del servidor.

3.26 Punto de acceso a interfaces de servicios.

La figura 3.41 indica la relación del protocolo de pila Bluetooth con esta cláusula. Esta cláusula describe, las funciones, características, servicios e interfaces SAP entre el MAC y las subcapas LLC dentro de la capa de enlace de datos del ISO/IEC 8802 protocolo LAN (IEEE. 802.2). La subcapa LLC constituye la subcapa más alta en la capa en la capa del enlace de datos (ver figura 3.42) y es común a varios medios métodos de acceso que son definidos y soportados por las actividades del ISO/IEC 8802.

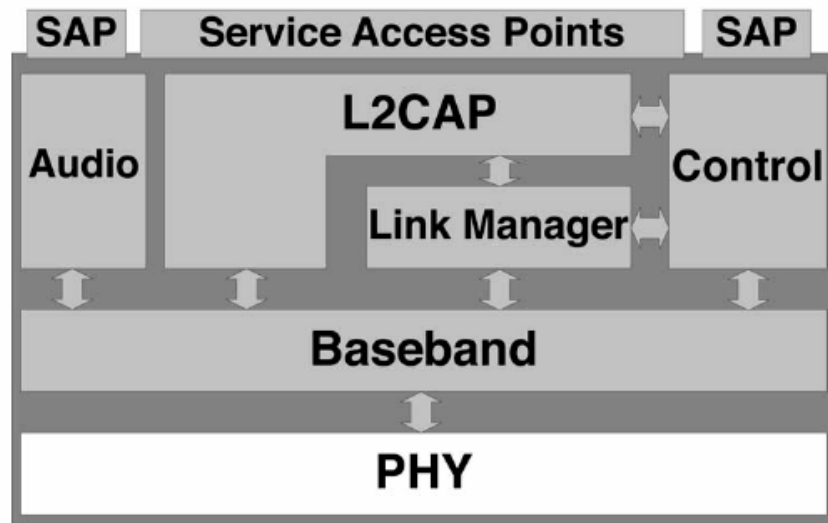


Figura 3.41 – Relación entre el Punto de Acceso a Servicios [1].

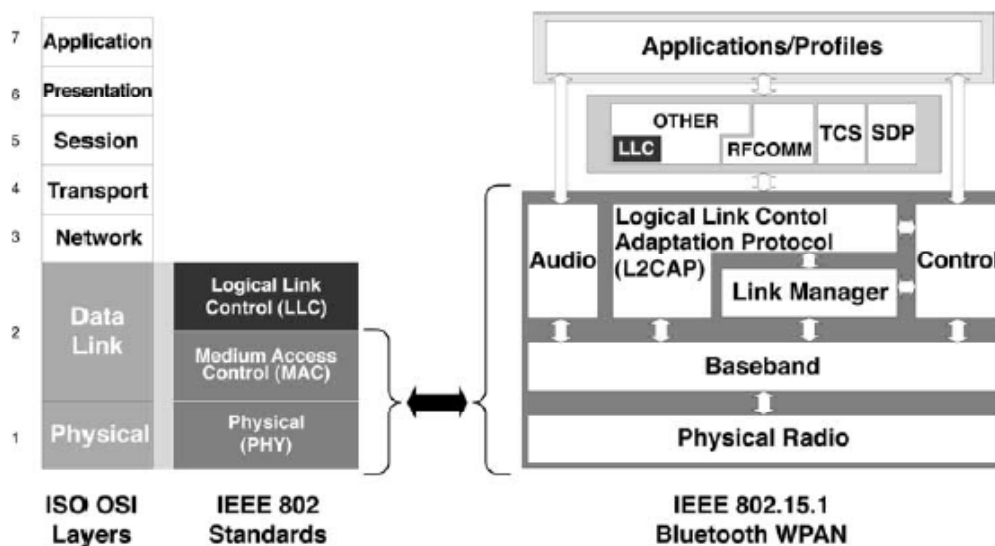


Figura 3.42 – Protocolos OSI y Bluetooth.[1]

A continuación se describirán los siguientes conceptos, útiles en esta cláusula:

- a) **Capa de enlace de datos (Data link layer):** Es la capa conceptual de control o de procesamiento lógico existente en la estructura jerárquica de una estación que

es responsable de mantener el control del enlace de datos. La capa de las funciones de enlace de datos proporciona una interfaz entre la estación lógica de capas superiores y en enlace a datos. Dichas funciones incluyen un campo de interpretación entre la dirección y el control, el canal de acceso y los comandos, envíos e interpretación.

- b) **Subcapa LLC:** Se define como la parte de la estación de datos que soporta el control de enlaces lógicos de uno o más enlaces lógicos. El LLC genera comandos PDUs para transmitirlos e interpreta los comandos PDUs recibidos.

Algunas de las responsabilidades asignadas a un LLC son las siguientes:

1. Inicialización del intercambio de señales de control.
2. Organización del flujo de datos.
3. Interpretación de comandos PDUs recientes y generación de respuestas apropiadas a dichos comandos PDUs.
4. Tomar acciones respecto al control de errores y recuperación de funciones en la subcapa LLC.

- c) **Subcapa MAC:** La parte de una estación de datos en la que residen las funciones MAC que se encuentran debajo de la subcapa LLC. Los procedimientos del MAC incluyen fragmentación y unión de las unidades de datos, realizar revisiones de error.

- d) **Subcapa de especificaciones de servicios del LLC hacia la subcapa MAC:** las especificaciones de servicio hacia la subcapa MAC proporcionan una descripción de los servicios que requiere la subcapa LLC de la subcapa MAC. Estos servicios están definidos para ser independientes a la metodología del

acceso a medios y de la naturaleza del medio en si. Todas las especificaciones antedichas están dadas en su forma primitiva que representan en una forma abstracta el intercambio lógico de información y el control entre la subcapa LLC y la función identificada (la subcapa MAC).

- e) **Tipo de operación:** Se define como un tipo de operación de control de enlace de datos (DLC). La operación de control “tipo 1” proporciona una forma de desconexión a través de un enlace de datos con un protocolo simple. Este tipo de operación resulta útil cuando las capas superiores proporcionan algún tipo de secuencia o esencia de recuperación para que dichas capas no necesiten replegarse en la capa de enlace de datos. Además este tipo de operación puede ser útil en aplicaciones en donde no sea necesario garantizar la entrega de cada unidad de datos de la capa de enlace de datos.
- f) **Clases de operación:** Es alguna clase de operación del LLC. La clase I de LLCs soporta únicamente operaciones de tipo 1. Los servicios de la clase 1 deben de ser aplicables de forma individual, grupal, global y para aplicaciones que no requieran procedimientos de control de procedimientos.

3.26.1 Subcapa LLC de especificaciones de servicios.

En general, los servicios de una capa (o subcapa) son las capacidades que esta ofrece a un usuario en la siguiente capa superior (o subcapa). Para lograr este servicio,

una capa (o subcapa) basa sus funciones en los servicios que ésta requiere de la capa (o subcapa) inferior inmediata (ver figura 3.43).

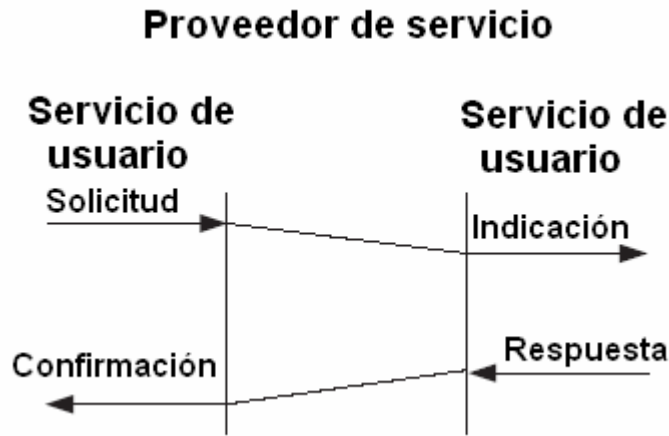


Figura 3.43. Primicias de servicio.

Los servicios se especifican cuando se describe la información que fluye entre el usuario N y la capa N (o subcapa). Dicho flujo de información se modela con eventos discretos e instantáneos que llevan a cabo un servicio. Cada evento consiste en hacer pasar un servicio primario de una capa (o subcapa) a otra por una capa N (o subcapa) de acceso a servicios asociado a un usuario N.

Los servicios se especifican al describir los servicios primarios y parámetros que caracterizan cada servicio. Un servicio puede tener uno o más servicios primarios que constituyen la activación que es relacionada a un servicio. Cada servicio primario puede tener ninguno o más parámetros de transporte de información requerida para proporcionar un servicio.

Los comandos se clasifican en cuatro tipos generales:

- a) **Solicitud:** La solicitud primaria o primitiva se pasa del usuario N hacia la capa N (o subcapa) para solicitar que se inicie un servicio.
- b) **Indicación:** El indicador primitivo se pasa de la capa N (o subcapa) al usuario N para que inicie un evento interno en la capa N (o subcapa) el cual es importante para el usuario N. Dicho evento está relacionado lógicamente a una solicitud de servicio remoto o debe ser causado por un elemento interno de la capa N (o subcapa).
- c) **Respuesta:** La respuesta primaria o primitiva es transmitida del usuario N a la capa (o subcapa) N para completar un procedimiento previamente iniciado por in indicador primitivo.
- d) **Confirmación:** La confirmación primitiva es transmitida de la capa N (o subcapa, según sea el caso) hacia el usuario N para transportar los resultados del servicio(s) previamente solicitado(s).

3.26.2 Interfaces Bluetooth.

La figura 3.44 ilustra los eventos y acciones realizadas por una implantación de la capa L2CAP. El cliente y el SAPs del servidor simplemente representan el inicio de la solicitud y la aceptación de la misma, respectivamente. Un cliente al nivel de aplicación puede ya sea iniciar o aceptar solicitudes. La convención de nombres es como sigue. La interfaz entre dos capas (interfaz vertical) utiliza el prefijo la capa inferior que ofrece el servicio hacia la capa superior. La interfaz entre dos entidades de la misma capa (interfaz horizontal) utiliza el prefijo del protocolo (agregando una P a la capa de identificación).

Los eventos que provienen del entorno se denominan solicitudes (Req), y sus correspondientes respuestas son llamadas confirmaciones (Cfm). Los eventos que vienen por debajo son llamados indicaciones (ind), y sus respuestas correspondientes son llamadas respuestas (Rsp). Respuestas que requieran procesamiento futuro son llamadas pendientes (Pnd). La notación de las confirmaciones y de las respuestas toman contestaciones positivas. Contestaciones negativas se denotan con un sufijo “Neg”.

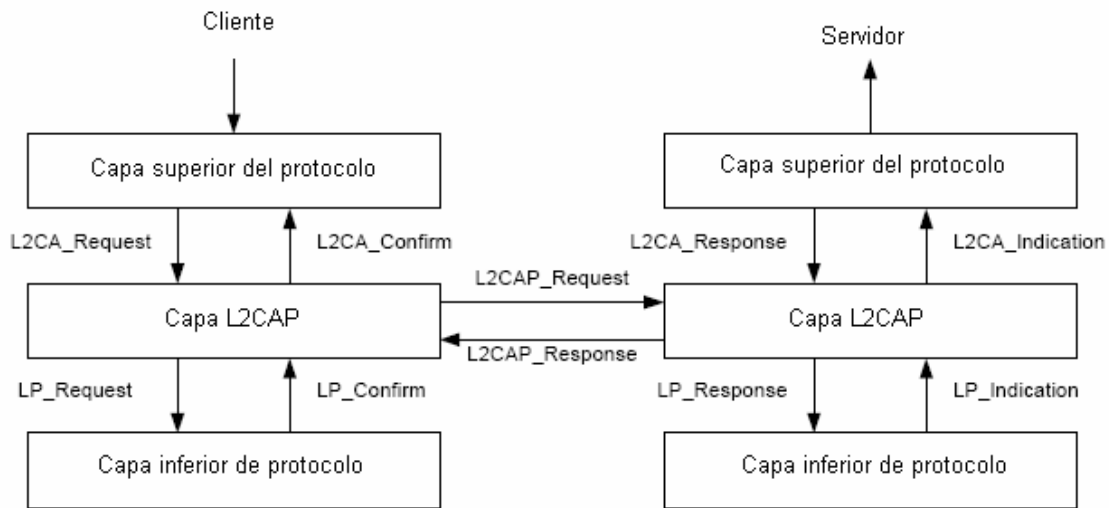


Figura 3.44 – Acciones y eventos L2CAP.

Mientras que por cada acción de solicitud hay una correspondiente confirmación (para el caso de aceptar la solicitud o una respuesta de negación para el caso de peticiones rechazadas), las indicaciones no siempre tienen respuesta. Esto sucede especialmente si las indicaciones son simplemente información sobre eventos locales.

3.26.3 Relación entre entidades con protocolo Bluetooth con construcciones IEEE 802.

La figura 3.45 muestra el mapa del concepto del protocolo IEEE 802 hacia componentes Bluetooth descritos en este documento. Como se puede apreciar la PHY esta compuesta por la parte de RF y de la banda base. El MAC contiene al L2CAP y al resto la banda base. Las funciones de manejo de la PHY, como la sincronización y generación de varias secuencias de frecuencia, están incorporadas a la banda base.

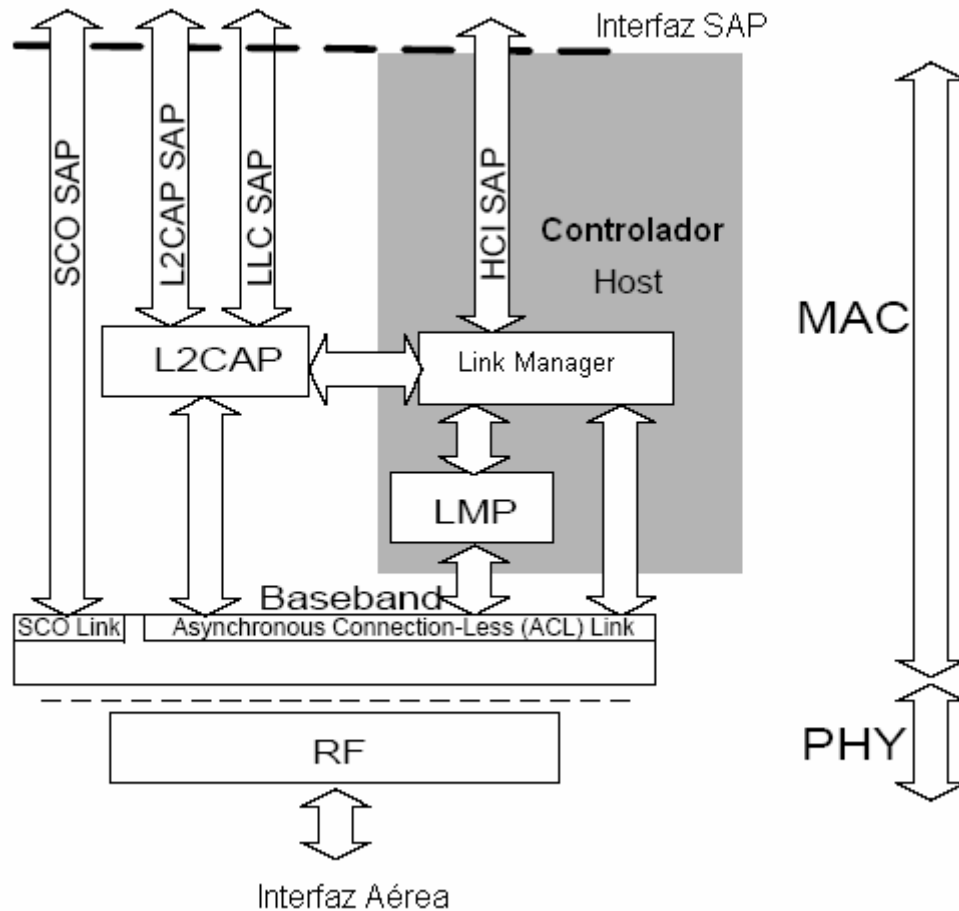


Figura 3.45 - Mapa del concepto del protocolo IEEE 802 hacia componentes Bluetooth.