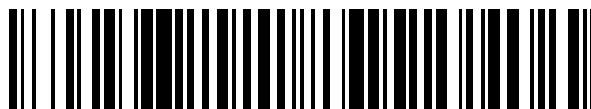


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 429 030**

51 Int. Cl.:

H04L 1/00 (2006.01)

H04L 27/26 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **03.06.2009** **E 11189878 (9)**

97 Fecha y número de publicación de la concesión europea: **31.07.2013** **EP 2424145**

54 Título: **Aparato para transmitir y recibir una señal y método para transmitir y recibir una señal**

30 Prioridad:

13.02.2009 US 152265 P

45 Fecha de publicación y mención en BOPI de la traducción de la patente:

12.11.2013

73 Titular/es:

LG ELECTRONICS INC. (100.0%)
20, Yeouido-dong, Yeongdeungpo-gu
Seoul 150-721, KR

72 Inventor/es:

KO, WOO SUK y
MOON, SANG CHUL

74 Agente/Representante:

DE ELZABURU MÁRQUEZ, Alberto

ES 2 429 030 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Aparato para transmitir y recibir una señal y método para transmitir y recibir una señal

Antecedentes de la invención

Campo de la invención

- 5 La presente invención se refiere a un método para la transmisión y recepción de una señal y a un aparato para la transmisión y recepción de una señal y, más particularmente, a un método para la transmisión y recepción de una señal y un aparato para la transmisión y recepción de una señal, que sean capaces de mejorar la eficiencia en la transmisión de datos.

Descripción de la técnica relacionada

- 10 Según se ha ido desarrollando la tecnología digital de difusión, los usuarios han recibido imágenes en movimiento en alta definición (HD). Con el continuo desarrollo de un algoritmo de compresión y el elevado rendimiento del hardware, se proporcionará a los usuarios en el futuro un mejor entorno. Un sistema de televisión digital (DTV) puede recibir una señal de difusión digital y proporcionar una variedad de servicios suplementarios a los usuarios así como una señal de video y una señal de audio.
- 15 La Difusión del Video Digital (DVB)-C2 es la tercera especificación en unirse a la familia de DVB de la segunda generación de sistemas de transmisión. Desarrollada en 1994, hoy en día DVB-C está desplegada en más de 50 millones de sintonizadores de cable en el todo el mundo. En línea con los otros sistemas DVB de segunda generación, DVB-C2 usa una combinación de códigos de Comprobación de paridad de Baja densidad (LDPC) y BCH. Esta potente corrección de error directo (FEC) proporciona aproximadamente 5 dB de mejora de la relación
- 20 portadora a ruido sobre el DVB-C. Los esquemas de entrelazado de bits apropiados optimizan la robustez global del sistema FEC. Extendidas mediante una cabecera, estas tramas se denominan Conductos de la Capa Física (PLP). Uno o más de estos PLP se multiplexan en un segmento de datos. Se aplican entrelazados de dos dimensiones (en los dominios del tiempo y de la frecuencia) a cada segmento permitiendo al receptor eliminar el impacto de los problemas de ráfagas y de interferencia selectiva de frecuencia tal como la entrada de frecuencia única.
- 25 Con el desarrollo de estas tecnologías de difusión digital, aumenta el requerimiento de un servicio tal como una señal de video y una señal de audio y aumenta gradualmente el tamaño de los datos deseados por los usuarios o el número de canales de difusión.

Compendio de la invención

- 30 En consecuencia, la presente invención está dirigida a un método para la transmisión y recepción de una señal y un aparato para la transmisión y recepción de una señal que obvian sustancialmente uno o más problemas debidos a las limitaciones y desventajas de la técnica relacionada.

Un objeto de la presente invención es proporcionar un método para la transmisión y recepción de una señal y un aparato para la transmisión y recepción de una señal, que sean capaces de mejorar la eficiencia en la transmisión de datos.

- 35 Otro objeto de la presente invención es proporcionar un método para la transmisión y recepción de una señal y un aparato para la transmisión y recepción de una señal que sean capaces de mejorar la capacidad de corrección de error de los bits que configuran un servicio.

- 40 Las ventajas, objetos, y características adicionales de la invención se expondrán en parte en la descripción a continuación y en parte serán evidentes para los expertos en la técnica tras el examen de lo que sigue. Los objetivos y otras ventajas de la invención se pueden realizar y conseguir mediante una estructura particularmente señalada en la descripción escrita y las reivindicaciones del presente documento así como en los dibujos adjuntos.

- 45 Para conseguir los objetivos, la presente invención proporciona un transmisor para la transmisión de datos de difusión, el transmisor que comprende: unos primeros medios de codificación de Corrección de Error Directo, FEC, para datos de Conducto de la Capa Física, PLP, de codificación FEC; unos segundos medios de codificación FEC para codificación FEC de datos de señalización de Capa 1; y medios de construcción de tramas para ensamblar los datos de PLP codificados FEC y los datos de señalización de Capa 1 codificados FEC para formar una trama de señal, en donde la trama de señal incluye símbolos de preámbulo que transportan los datos de señalización de Capa 1 y los símbolos de datos que transportan los datos del PLP, en donde los símbolos de preámbulo incluyen bloques
- 50 subportadoras, en donde los datos de señalización de Capa 1 incluyen una información PLP_type que indica si un PLP es un PLP común o un PLP de datos normal o un PLP de datos agrupados y los datos de señalización de Capa 1 además incluyen una información de PLP_bundle_Flag, la información de PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.

Otra realización más de la presente invención proporciona un receptor para el procesamiento de los datos de difusión, comprendiendo el receptor: unos medios de análisis sintáctico de tramas para analizar una trama de señal que incluyen símbolos de preámbulo que transportan datos de señalización de Capa 1 y símbolos de datos que transportan datos de Conducto de la Capa Física, PLP, y sacar los datos de PLP y los datos de señalización de Capa 1; unos primeros medios de decodificación de Corrección de Error Directo, FEC, para decodificación FEC de los datos de PLP; y unos segundos medios de decodificación FEC para decodificación FEC de los datos de señalización de Capa 1, en donde los símbolos de preámbulo incluyen bloques de L1 que se repiten en un dominio de la frecuencia, en donde al menos uno de los bloques de L1 incluye 3408 subportadoras, en donde los datos de señalización de Capa 1 incluyen una información PLP_type que indica si un PLP es un PLP común o un PLP de datos normales o un PLP de datos agrupados y los datos de señalización de Capa 1 además incluyen una información PLP_bundle_Flag, la información PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.

Otra realización más de la presente invención proporciona un método para el procesamiento de datos de difusión, el método que comprende: analizar una trama de señal incluyendo símbolos de preámbulo que transportan datos de señalización de Capa 1 y símbolos de datos que transportan datos de Conducto de la Capa Física, PLP y sacar los datos de PLP y los datos de señalización de Capa 1; decodificación de Corrección de Error Directo, FEC, que decodifica los datos de PLP; y decodificación FEC de los datos de señalización de Capa 1, en donde los símbolos de preámbulo incluyen bloques de L1 que se repiten en un dominio de frecuencia, en donde al menos uno de los bloques de L1 incluye 3408 subportadoras, en donde los datos de señalización de Capa 1 incluyen una información PLP_type que indica si un PLP es un PLP común o un PLP de datos normal o un PLP de datos agrupado y los datos de señalización de Capa 1 además incluyen una información PLP_bundle_Flag, la información PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.

Otra realización más de la presente invención proporciona un método para la transmisión de datos de difusión, comprendiendo el método: codificación de Corrección de Error Directo, FEC, de datos de Conducto de la Capa Física, PLP; codificación FEC de datos de señalización de Capa 1; y ensamblar los datos de PLP codificados FEC y los datos de señalización de Capa 1 codificados FEC para formar una trama de señal, en donde la trama de señal incluye símbolos de preámbulo que transportan los datos de señalización de Capa 1 y símbolos de datos que transportan los datos de PLP, en donde los símbolos de preámbulo incluyen bloques de L1 que se repiten en un dominio de la frecuencia, en donde al menos uno de los bloques de L1 incluye 3408 subportadoras, en donde los datos de señalización de Capa 1 incluyen una información PLP_type que indica si un PLP es un PLP común o un PLP de datos normales o un PLP de datos agrupados y los datos de señalización de Capa 1 además incluyen una información PLP_bundle_Flag, la información PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.

Descripción de las realizaciones preferidas

Los dibujos adjuntos, que se incluyen para proporcionar una comprensión adicional de la invención y se incorporan en, y constituyen una parte de esta solicitud, ilustran la(s) realización(realizaciones) de la invención y junto con la descripción sirven para explicar los principios de la invención. En los dibujos:

La Fig. 1 es un ejemplo de un sistema de transmisión digital.

La Fig. 2 es un ejemplo de un procesador de entrada.

La Fig. 3 es una información que se puede incluir en la banda base (BB).

La Fig. 4 es un ejemplo de un módulo BICM.

La Fig. 5 es un ejemplo de un codificador recortado/perforado.

La Fig. 6 es un ejemplo de aplicación de varias constelaciones.

La Fig. 7 es otro ejemplo de casos en los que se considera la compatibilidad entre sistemas convencionales.

La Fig. 8 es una estructura de trama que comprende el preámbulo para la señalización de L1 y los Símbolos de datos para los datos PLP.

La Fig. 9 es un ejemplo de un constructor de tramas.

La Fig. 10 es un ejemplo del módulo de inserción piloto 404 mostrado en la Fig. 4.

La Fig. 11 es una estructura de SP.

La Fig. 12 es una nueva estructura de SP o Patrón Piloto (PP5').

- La Fig. 13 es una estructura PP5' sugerida.
- La Fig. 14 es una relación entre los símbolos de datos y el preámbulo.
- La Fig. 15 es otra relación entre los símbolos de datos y el preámbulo
- La Fig. 16 es un ejemplo de un perfil retardado del canal por cable.
- 5 La Fig. 17 es una estructura de piloto disperso que usa $z=56$ y $z=112$.
- La Fig. 18 es un ejemplo de modulador en base a OFDM.
- La Fig. 19 es un ejemplo de estructura de preámbulo.
- La Fig. 20 es un ejemplo de la decodificación del preámbulo.
- La Fig. 21 es un proceso para el diseño de un preámbulo más optimizado.
- 10 La Fig. 22 es otro ejemplo de estructura de preámbulo.
- La Fig. 23 es otro ejemplo de decodificación del preámbulo.
- La Fig. 24 es un ejemplo de estructura de preámbulo.
- La Fig. 25 es un ejemplo de la decodificación de L1.
- La Fig. 26 es un ejemplo de procesador analógico.
- 15 La Fig. 27 es un ejemplo de sistema receptor digital.
- La Fig. 28 es un ejemplo de un procesador analógico usado en el receptor.
- La Fig. 29 es un ejemplo de demodulador.
- La Fig. 30 es un ejemplo de analizador de tramas.
- La Fig. 31 un ejemplo de demodulador BICM.
- 20 La Fig. 32 es un ejemplo de decodificación LDPC usando recorte/perforación.
- La Fig. 33 es un ejemplo de procesador de salida.
- La Fig. 34 es un ejemplo de una tasa de repetición del bloque L1 de 8 MHz.
- La Fig. 35 es un ejemplo de una tasa de repetición del bloque L1 de 8 MHz.
- La Fig. 36 es una nueva tasa de repetición del bloque L1 de 7, 61 MHz.
- 25 La Fig. 37 es un ejemplo de señalización de L1 que se transmite en la cabecera de trama.
- La Fig. 38 es el resultado de simulación del preámbulo y la estructura L1.
- La Fig. 39 es un ejemplo de entrelazador de símbolos.
- La Fig. 40 es un ejemplo de una transmisión de bloque de L1.
- La Fig. 41 es otro ejemplo de la señalización de L1 transmitida dentro de una cabecera de trama.
- 30 La Fig. 42 es un ejemplo de entrelazado/desentrelazado de frecuencia o tiempo.
- La Fig. 43 es una tabla que analiza la sobrecarga de la señalización de L1 que se transmite en la Cabecera FECFRAME en el módulo de Inserción de Cabecera ModCod 307 en la ruta de datos del módulo BICM mostrado en la Fig. 3.
- La Fig. 44 muestra la estructura de la cabecera FECFRAME para minimizar la sobrecarga.
- 35 La Fig. 45 muestra un rendimiento de la tasa de error de bits (BER) de la protección de L1 anteriormente mencionada.
- La Fig. 46 muestra ejemplos de una trama de transmisión y la estructura de la trama FEC.
- La Fig. 47 muestra un ejemplo de la señalización de L1.

- La Fig. 48 muestra un ejemplo de la señalización previa de L1.
- La Fig. 49 muestra la estructura del bloque de señalización de L1.
- La Fig. 50 muestra un entrelazado de tiempos de L1.
- La Fig. 51 muestra un ejemplo de la extracción de la información de modulación y códigos.
- 5 La Fig. 52 muestra otro ejemplo de la señalización previa de L1.
- La Fig. 53 muestra un ejemplo de la planificación del bloque de señalización de L1 que se transmite en un preámbulo.
- La Fig. 54 muestra un ejemplo de la señalización previa de L1 en el que se considera un refuerzo de la potencia.
- La Fig. 55 muestra un ejemplo de la señalización de L1.
- 10 La Fig. 56 muestra otro ejemplo de la extracción de la información de modulación y código.
- La Fig. 57 muestra otro ejemplo de la extracción de la información de modulación y código.
- La Fig. 58 muestra un ejemplo de la sincronización previa de L1.
- La Fig. 59 muestra un ejemplo de la señalización previa de L1.
- La Fig. 60 muestra un ejemplo de la señalización de L1.
- 15 La Fig. 61 muestra un ejemplo del trayecto de señalización de L1.
- La Fig. 62 es otro ejemplo de la señalización de L1 transmitida dentro de una cabecera de trama.
- La Fig. 63 es otro ejemplo de la señalización de L1 transmitida dentro de una cabecera de trama.
- La Fig. 64 es otro ejemplo de la señalización de L1 transmitida dentro de una cabecera de trama.
- La Fig. 65 muestra un ejemplo de señalización de L1.
- 20 La Fig. 66 es un ejemplo de entrelazado de símbolos.
- La Fig. 67 muestra un rendimiento de entrelazado del entrelazador de tiempos de la Fig. 66.
- La Fig. 68 es un ejemplo de entrelazado de símbolos.
- La Fig. 69 muestra un rendimiento de entrelazado del entrelazador de tiempos de la Fig. 68. La Fig. 70 es un ejemplo de entrelazado de símbolos.
- 25 La Fig. 71 es otro ejemplo de entrelazado de tiempos.
- La Fig. 72 es un resultado de entrelazado usando el procedimiento mostrado en la Fig. 71.
- La Fig. 73 es un ejemplo del procedimiento de direccionamiento de la Fig. 72.
- La Fig. 74 es otro ejemplo de entrelazado de tiempos del L1.
- La Fig. 75 es un ejemplo de desentrelazado de símbolos.
- 30 La Fig. 76 es otro ejemplo de desentrelazado.
- La Fig. 77 es un ejemplo de desentrelazado de símbolos.
- La Fig. 78 es un ejemplo de direcciones de fila y columna para el desentrelazador de tiempos.
- La Fig. 79 muestra un ejemplo del entrelazado del bloque general en un dominio de símbolos de datos en el que no se usan pilotos.
- 35 La Fig. 80 es un ejemplo de un transmisor OFDM que usa segmentos de datos.
- La Fig. 81 es un ejemplo de un receptor OFDM que usa segmentos de datos.
- La Fig. 82 es un ejemplo de entrelazador de tiempos y un ejemplo de desentrelazador de tiempos.
- La Fig. 83 es un ejemplo de la formación de símbolos OFDM.

- La Fig. 84 un ejemplo de un Entrelazador de Tiempos (TI).
- La Fig. 85 un ejemplo de un Entrelazador de Tiempos (TI).
- La Fig. 86 es un ejemplo de una estructura de preámbulo en un transmisor y un ejemplo de un proceso en un receptor.
- 5 La Fig. 87 es un ejemplo de un proceso en un receptor para tener el L1_XFEC_FRAME a partir del preámbulo.
- La Fig. 88 es un ejemplo de una estructura de preámbulo en un transmisor y un ejemplo de un proceso en un receptor.
- La Fig. 89 un ejemplo de un Entrelazador de Tiempos (TI).
- La Fig. 90 es un ejemplo de un transmisor OFDM que usa segmentos de datos.
- 10 La Fig. 91 es un ejemplo de un receptor OFDM que usa segmentos de datos.
- La Fig. 92 un ejemplo de un Entrelazador de Tiempos (TI).
- La Fig. 93 un ejemplo de un Desentrelazador de Tiempos (TDI).
- La Fig. 94 un ejemplo de un Entrelazador de Tiempos (TI).
- La Fig. 95 es un ejemplo del flujo de entrelazado y desentrelazado de tiempos del preámbulo.
- 15 La Fig. 96 es un parámetro de profundidad del Entrelazado de Tiempos en la señalización de la cabecera de L1.
- La Fig. 97 es un ejemplo de una señalización de cabecera de L1, estructura de L1 y un procedimiento de rellenado.
- La Fig. 98 es un ejemplo de señalización de L1.
- La Fig. 99 es un ejemplo de dslice_ti_depth.
- La Fig. 100 es un ejemplo de dslice_type.
- 20 La Fig. 101 es un ejemplo de plp_type.
- La Fig. 102 es un ejemplo de Plp_payload_type.
- La Fig. 103 es un ejemplo de Plp_modcod.
- La Fig. 104 es un ejemplo de GI.
- La Fig. 105 es un ejemplo de PAPR.
- 25 La Fig. 106 es un ejemplo de señalización de L1.
- La Fig. 107 es un ejemplo de plp_type. La Fig. 108 es un ejemplo de señalización de L1.
- La Fig. 109 es un ejemplo de señalización de cabecera de L1, estructura de L1 y un procedimiento de rellenado.
- La Fig. 110 es un ejemplo de señalización de L1.
- La Fig. 111 muestra ejemplos de campos de la señalización de L1.
- 30 La Fig. 112 es un ejemplo de señalización de L1.
- La Fig. 113 es un ejemplo de plp_type.
- La Fig. 114 es un ejemplo de la señalización de L1 y la señalización de L2 para tipos de PLP normal y empaquetadas.
- 35 La Fig. 115 es un ejemplo de flujo de la acción de decodificación de L1 y L2 de un receptor DVB-C2 convencional con un sintonizador simple de 8 MHz.
- La Fig. 116 es un ejemplo del flujo de la acción de decodificación de L1 y L2 de un receptor DVB-C2 de alta calidad con múltiples sintonizadores o un único sintonizador de banda ancha.
- La Fig. 117 es un ejemplo de una señalización de L2 para C2.
- La Fig. 118 es un ejemplo de la duración del símbolo OFDM activo.

La Fig. 119 es un ejemplo de los valores del intervalo de guarda.

La Fig. 120 es un ejemplo de señalización de L1.

La Fig. 121 es un ejemplo del entrelazado de tiempos del bloque de L1.

Descripción de las realizaciones preferidas

- 5 Se hace referencia ahora en detalle a las realizaciones preferidas de la presente invención, cuyos ejemplos se ilustran en los dibujos adjuntos. Cuando es posible, se usarán los mismos números de referencia a todo lo largo de los dibujos para referirse a las partes iguales o similares.

En la siguiente descripción, el término "servicio" es indicativo de cualquier contenido de difusión que se puede transmitir/recibir por el aparato de transmisión/recepción de la señal.

- 10 La Fig. 1 muestra un ejemplo del sistema de transmisión digital de acuerdo con una realización de la presente invención. Las entradas pueden incluir un número de corrientes MPEG-TS o corrientes GSE (Encapsulado General de la Transmisión Continua). Un procesador de entrada 101 puede añadir parámetros de transmisión a la corriente de entrada y realizar la planificación del módulo BCIM 102. El módulo BCIM 102 puede añadir redundancia y entrelazar datos para corrección de error en el canal de transmisión. El constructor de tramas 103 puede construir tramas mediante la adición de información de señalización de la capa física y pilotos. Un modulador 104 puede realizar la modulación sólo los símbolos de entrada en procedimientos eficientes. Un procesador analógico 105 puede realizar varios procesos para convertir las señales digitales de entrada en señales analógicas de salida.

- La Fig. 2 muestra un ejemplo de un procesador de entrada. La corriente MPEG-TS o GSE de entrada se puede transformar por el procesador de entrada en un total de n corrientes que se pueden procesar independientemente. Cada una de estas corrientes puede ser o bien trama TS completa que incluye múltiples componentes de servicio o bien una trama TS mínima que incluye componentes del servicio (por ejemplo, video o audio). Además, cada una de estas corrientes puede ser una corriente GSE que transmite o bien múltiples servicios o bien un servicio simple.

- La interfaz de entrada 202-1 puede asignar un número de bits de entrada igual a la capacidad máxima del campo de datos de una trama de banda base (BB). Se puede insertar un relleno para completar la capacidad del bloque de códigos LDPC/BCH. El sincronizador de la corriente de entrada 203-1 puede proporcionar un mecanismo para regenerar, en el receptor, el reloj de la Transmisión Continua de Transporte (o Transmisión Continua Genérica en Paquetes), para garantizar tasas de bits y retardos constantes de punto a punto.

- Para permitir la recombinación de la Transmisión Continua de Transporte sin requerir una memoria adicional en el receptor, la Transmisión Continua de Transporte se retarda mediante el compensador de retardo 204-1~ n considerando los parámetros de entrelazado de los PLP de datos en un grupo y el PLP común correspondiente. El módulo de borrado de paquetes Nulos 205-1- n puede aumentar la eficiencia de la transmisión mediante la eliminación de los paquetes nulos insertados para un caso de servicio en VBR (tasa de bits variable). Los módulos del codificador de Comprobación de Redundancia Cíclica (CRC) 206-1~ n pueden añadir la paridad de CRC para aumentar la fiabilidad de la transmisión de la trama de BB. Los módulos de inserción de la cabecera de BB (207-1~ n) pueden añadir la cabecera de la trama de BB en la parte inicial de la trama de BB. La información que se puede incluir en una cabecera de BB se muestra en la Fig. 3.

- Un módulo Fusionador/segmentador 208 puede realizar el segmentado de las tramas de BB a partir de cada PLP, mezclando las tramas de BB desde múltiples PLP y planificando cada trama de BB dentro de una trama de transmisión. Por lo tanto, el módulo fusionador/segmentador 208 puede producir la salida de la información de señalización de L1 en relación a la asignación del PLP en la trama. Finalmente, un módulo de codificación 209 puede aleatorizar los flujos de bits de entrada para minimizar la correlación entre bits dentro de los flujos de bits. Los módulos sombreados en la Fig. 2 son módulos usados cuando el sistema de transmisión usa un PLP simple, los otros módulos de la Fig. 2 son módulos usados cuando el dispositivo de transmisión usa múltiples PLP.

- La Fig. 4 muestra una realización del módulo BICM de acuerdo con la presente invención. La Fig. 4a muestra un BICM para un recorrido de datos y la Fig. 4b muestra un BICM para un recorrido de señalización de la L1.

- Con referencia a la Fig. 4a, un codificador exterior 301 y un codificador interior 303 pueden añadir redundancia a los flujos de bits de entrada para corrección de error. Un entrelazador exterior 302 y un entrelazador interior 304 pueden entrelazar los bits para impedir el error de ráfagas. El entrelazador exterior 302 se puede omitir si el BICM es específicamente para DVB-C2. Un demultiplexor de bits 305 puede controlar la fiabilidad de cada salida de bits desde el entrelazador interior 304. Un mapeador de símbolos 306 puede mapear flujos continuos de bits de entrada en transmisiones de símbolos. En este momento, es posible usar cualquiera de entre un QAM convencional, un MQAM que usa el BRGC anteriormente mencionado para mejorar el rendimiento, un NU-QAM que usa modulación no uniforme o un NU-MQAM que usa modulación no uniforme aplicada al BRGC para mejorar el rendimiento. Para construir un sistema que sea más robusto contra el ruido, se pueden considerar combinaciones o modulaciones que usen MQAM y/o NU-MQAM dependiendo de la tasa de códigos del código de corrección de error y la capacidad de la constelación. En este momento, el mapeador de Símbolos 306 puede usar una constelación apropiada de

acuerdo con la tasa de códigos y la capacidad de la constelación. La Fig. 6 muestra un ejemplo de tales combinaciones.

El caso 1 muestra un ejemplo del uso solamente del NU-MQAM en una tasa de códigos baja para una implementación del sistema simplificada. El caso 2 muestra un ejemplo del uso de la constelación optimizada en cada tasa de códigos. El transmisor puede enviar información acerca de la tasa de códigos del código de corrección de error y la capacidad de la constelación al receptor de modo que el receptor pueda usar una constelación apropiada. La Fig. 7 muestra otro ejemplo de casos en los que se considera la compatibilidad entre los sistemas convencionales. Además de estos ejemplos, son posibles combinaciones adicionales para la optimización del sistema.

El módulo de inserción de la Cabecera ModCod 307 mostrado en la Fig. 4 puede tomar la información de realimentación de la modulación y codificación adaptativas (ACM) / modulación y codificación variable (VCM) y añadir información del parámetro usado en la modulación y codificación a un bloque FEC como cabecera. La cabecera del tipo de Modulación/Tasa de códigos (ModCod) puede incluir la siguiente información:

* Tipo FEC (1 bit) – LDPC largo o corto

* Tasa de códigos (3 bits)

* Modulación (3 bits) -hasta QAM de 64K

* Identificador PLP (8 bits)

El Entrelazador de Símbolos 308 puede realizar el entrelazado en el dominio de los símbolos para obtener efectos de entrelazado adicional. Se pueden realizar procesos similares realizados en el recorrido de los datos sobre el recorrido de señalización de la L1 pero con posiblemente parámetros diferentes 301-1 ~ 308-1. En este punto, se puede usar un codificador de recorte/perforación 303-1 para código interior.

La Fig. 5 muestra un ejemplo de la codificación LDPC usando recorte/perforación. El proceso de recorte se puede realizar sobre los bloques de entrada que tienen menos bits que el número de bits requerido para la codificación LDPC dado que muchos bits cero requeridos para la codificación LDPC se pueden rellenar mediante el módulo de rellenado de ceros 301 c. Los flujos de bits de entrada Rellenados con Ceros pueden tener bits de paridad a través del codificador LDPC 302c. En este momento, para flujos de bits que correspondan a los flujos de bits originales, los ceros se pueden eliminar (303c) y para flujos de bits de paridad, se puede realizar la perforación de acuerdo con las tasas de código por el módulo de perforación de paridad 304c. Estos flujos de bits procesados de información y flujos de bits de paridad se pueden multiplexar en las secuencias originales y producir su salida mediante el multiplexor 305c.

La Fig. 8 muestra una estructura de trama que comprende el preámbulo para la señalización de la L1 y símbolos de datos para los datos del PLP. Se puede ver que el preámbulo y los símbolos de datos se pueden generar cíclicamente, usando una trama como unidad. Los símbolos de datos comprenden el tipo 0 de PLP que se transmite usando una modulación/codificación fija y el tipo 1 de PLP que se transmite usando una modulación/codificación variable. Para el tipo 0 de PLP, la información tal como la modulación, el tipo FEC y la tasa de códigos de FEC se transmiten en el preámbulo (véase la Fig. 9 en relación al módulo de inserción de la cabecera de Trama 401). En el tipo 1 de PLP, la información correspondiente se puede transmitir en la cabecera del bloque FEC de un símbolo de datos (véase la Fig. 3 en relación al módulo de inserción de la cabecera ModCod 307). Mediante la separación de los tipos PLP, se puede reducir la sobrecarga del ModCod en un 3~4% a partir de una velocidad de transmisión total, para el tipo 0 de PLP que se transmite con una tasa de bits fija. En el receptor, para la modulación/codificación fija de PLP del tipo 0 de PLP, el eliminador de la cabecera de Trama r401 mostrado en la Fig. 30 puede extraer la información sobre Modulación y tasa de código FEC y proporcionar la información extraída a un módulo de decodificación BICM. Para la modulación/codificación de PLP variable del tipo 1 de PLP, el extractor del ModCod r307, r307-1 mostrado en la Fig. 31 puede extraer y proporcionar los parámetros necesarios para la decodificación BICM.

La Fig. 9 muestra un ejemplo de un constructor de tramas. Un módulo de inserción de cabecera de trama 401 puede formar una trama a partir de las corrientes de símbolos de entrada y pueden añadir la cabecera de tramas en el frontal de cada trama transmitida. La cabecera de trama puede incluir la siguiente información:

* Número de canales adheridos (4 bits)

* Intervalo de guarda (2 bits)

* PAPR (2 bits)

* Patrón piloto (2 bits)

* Identificación del sistema digital (16 bits)

	* Identificación de la trama (16 bits)
	* Longitud de la trama (16 bits) -número de símbolos de Multiplexado por División de Frecuencia Ortogonal (OFDM) por trama
	* Longitud de supertrama (16 bits) -número de tramas por supertrama
5	* Número de PLP (8 bits)
	* para cada PLP
	Identificación del PLP (8 bits)
	ID de canales adheridos (4 bits)
	Comienzo del PLP (9 bits)
10	Tipo de PLP (2 bits) – PLP común u otras
	Tipo de contenido útil del PLP (5 bits)
	Tipo de MC (1 bit) – modulación y codificación fija/variable
	si el tipo de MC == modulación y codificación fija
	Tipo de FEC (1 bits) - LDPC largo corto
15	Tasa de código (3 bits)
	Modulación (3 bits) -hasta QAM de 64K
	fin si;
	Número de canales de supresión (2 bits)
	para cada supresión
20	Inicio de la supresión (9 bits)
	Ancho de la supresión (9 bits)
	fin para;
	Ancho del PLP (9 bits) – número máximo de bloques FEC del PLP
	Tipo de entrelazado de tiempos de PLP (2 bits)
25	fin para;
	* CRC-32 (32 bits)
30	Se supone un entorno de canales adheridos para la información L1 transmitida en la cabecera de Trama y los datos que corresponden a cada segmento de datos se definen como PLP. Por lo tanto, la información tal como el identificador de PLP, identificador de canales adheridos y dirección de comienzo del PLP se requiere para cada canal usado en la adhesión. Una realización de la presente invención sugiere la transmisión del campo ModCod en la cabecera de la trama FEC si el tipo de PLP soporta modulación/codificación variable y la transmisión del campo ModCod en la cabecera de Trama si el tipo de PLP soporta modulación/codificación fija para reducir la sobrecarga de señalización. Además, si existe una banda de Supresión para cada PLP, mediante la transmisión de la dirección de inicio de la Supresión y su ancho, la codificación que corresponde a los portadores en el receptor puede llegar a ser innecesaria.
35	
	La Fig. 10 muestra un ejemplo del patrón piloto (PP5) aplicado en un entorno de canales adheridos. Como se muestra, si las posiciones de los SP coinciden con las posiciones piloto del preámbulo, puede tener lugar una estructura piloto irregular.
40	La Fig. 10a muestra un ejemplo del módulo de inserción del piloto 404 como se muestra en la Fig. 9. Como se representa en la Fig. 10a, si se usa una banda de frecuencia simple (por ejemplo, 8 MHz), el ancho de banda disponible es de 7,61 MHz, pero si se adhieren múltiples bandas de frecuencia, se pueden eliminar las bandas de guarda, y por ello, la eficiencia en frecuencia puede aumentar grandemente. La Fig. 10b es un ejemplo del módulo de inserción del preámbulo 504 como se muestra en la Fig. 18 que se transmite en la parte frontal de la trama e incluso con canales adheridos, el preámbulo tiene una tasa de repetición de 7, 61 MHz, que es un ancho de banda

del bloque de L1. Esta es una estructura que considera el ancho de banda de un sintonizador que realiza un barrido de canal inicial.

Existen patrones piloto tanto para los Símbolos del Preámbulo como de Datos. Para símbolo de datos, se pueden usar patrones piloto dispersos (SP). El Patrón Piloto (PP5) el Patrón Piloto (PP7) de T2 pueden ser buenos candidatos para la interpolación solamente de frecuencia. El PP5 tiene $x=12$, $y=4$, $z=48$ para $GI=1/64$ y PP7 tiene $x=24$, $y=4$, $z=96$ para $GI=1/128$. Es también posible una interpolación adicional en el tiempo para una mejor estimación del canal. Los patrones piloto para el preámbulo pueden cubrir todas las posibles posiciones del piloto para la adquisición del canal inicial. Además, las posiciones del piloto del preámbulo deberían coincidir con las posiciones de los SP y se desea un único patrón de pilotos tanto para el preámbulo como para los SP. Los pilotos del preámbulo se podrían usar también para una interpolación en el tiempo y cada preámbulo podría tener un patrón piloto idéntico. Estos requisitos son importantes para la detección del C2 en el barrido y necesarios para una estimación del desplazamiento de frecuencia con la correlación de la secuencia de codificación. En un entorno de canales adheridos, la coincidencia en las posiciones del piloto se debería mantener también para unos canales adheridos debido a que una estructura de piloto irregular puede degradar el rendimiento de la interpolación.

En detalle, si una distancia z entre los pilotos dispersos (SP) en un símbolo OFDM es 48 y si una distancia y entre los SP que corresponden a una portadora de SP específica a lo largo del eje del tiempo es 4, una distancia efectiva x tras la interpolación en el tiempo llega a ser 12. Esto es cuando una fracción del intervalo de guarda (GI) es $1/64$. Si la fracción del GI es $1/128$, se pueden usar $x=24$, $y=4$ y $z=96$. Si se usan canales adheridos, las posiciones del SP se pueden hacer coincidentes con las posiciones del piloto del preámbulo mediante la generación de puntos no continuos en la estructura de piloto dispersa.

En este momento, las posiciones del piloto de preámbulo se pueden hacer coincidentes con cada una de las posiciones de SP de los símbolos de datos. Cuando se usan canales adheridos, el segmento de datos en el que se transmite un servicio, se puede determinar independientemente de la granularidad del ancho de banda de 8 MHz. Sin embargo, para reducción de la sobrecarga en el direccionamiento de los segmentos de datos, se puede elegir la transmisión que comienza desde la posición de los SP y que finaliza la posición de los SP.

Cuando un receptor recibe tales SP, si es necesario, la estimación del canal (r501) mostrada en la Fig. 29 puede realizar una interpolación en el tiempo para obtener los pilotos mostrados en las líneas de puntos de la Fig. 10 y 10 y realizar la interpolación de frecuencia. En este momento, los puntos no continuos cuyos intervalos se designan como '32' en la Fig. 10a, se pueden implementar tanto realizando las interpolaciones por separado a la izquierda y derecha como realizando las interpolaciones solamente en un lado y a continuación realizar la interpolación en el otro lado mediante el uso de las posiciones piloto ya interpoladas cuyo intervalo es 12 como un punto de referencia. En este momento, el ancho del segmento de datos puede variar dentro de 7, 61 MHz, por ello, un receptor puede minimizar el consumo de potencia mediante la realización de una estimación del canal y decodificando solamente las portadoras necesarias.

La Fig. 11 muestra otro ejemplo de PP5 aplicado en el entorno de canales adheridos o una estructura de SP que mantiene una distancia efectiva x como 12 para evitar la estructura de SP irregular mostrada en la Fig. 10 cuando se usan canales adheridos. Como se muestra, si la distancia SP se mantiene consistente en caso de canales adheridos, no habrá problema en la interpolación de frecuencia pero las posiciones piloto entre los símbolos de datos del preámbulo pueden no ser coincidentes. En otras palabras, esa estructura no requiere estimación de canal adicional para una estructura de SP irregular, sin embargo, las posiciones de SP usadas en canales adheridos y las posiciones del piloto del preámbulo se convierten en diferentes para cada canal.

La Fig. 12 muestra una nueva estructura de SP o PP5' para proporcionar una solución a los dos problemas anteriormente mencionados en el entorno de canales adheridos. Específicamente, una distancia piloto de $x=16$ puede resolver esos problemas. Para preservar la densidad de pilotos o para mantener la misma sobrecarga, un PP5' puede tener $x=16$, $y=3$, $z=48$ para $GI=1/64$ y un PP7' puede tener $x=16$, $y=6$, $z=96$ para $GI=1/128$. Puede mantenerse aún la capacidad de interpolación solamente en frecuencia. Las posiciones piloto se representan en la Fig. 12 para comparación con la estructura PP5.

La Fig. 13 muestra un ejemplo de un nuevo patrón de SP o estructura PP5' en el entorno de canales adheridos. Como se muestra en la Fig. 46, tanto si se usa un canal simple como canales adheridos, se puede proporcionar una distancia piloto efectiva de $x=16$. Además, debido que las posiciones de SP se pueden hacer coincidentes con las posiciones piloto del preámbulo, se puede evitar cada deterioro de la estimación del canal producido por la irregularidad de los SP o posiciones de SP no coincidentes. En otras palabras, no existe ninguna posición irregular de SP para un interpolador de frecuencia y se proporciona coincidencia entre el preámbulo y las posiciones de SP.

En consecuencia, los patrones de SP nuevos propuestos pueden ser ventajosos en que se puede usar un único patrón de SP tanto para canal único como adheridos; no se puede producir ninguna estructura piloto irregular, por ello es posible una buena estimación del canal; tanto las posiciones del piloto del preámbulo como de SP se pueden mantener coincidentes; la densidad de piloto se puede mantener la misma que para el PP5 y el PP7 respectivamente y se puede preservar también la capacidad de interpolación solamente en Frecuencia.

Además, la estructura de preámbulo pueden satisfacer los requisitos tales como que las posiciones de piloto de preámbulo deberían cubrir todas las posibles posiciones de SP para una adquisición de canal inicial; el número máximo de portadoras debería ser 3409 (7, 61 MHz) para un barrido inicial; se deberían usar exactamente los mismos patrones piloto y la secuencia de codificación para la detección de C2 y no se requiere un preámbulo de detección específica como P1 en T2.

En términos de relación con la estructura de la trama, se puede modificar la granularidad de la posición del segmento de datos a 16 portadoras en lugar de 12, por ello, puede tener lugar una menor sobrecarga en el direccionamiento de la posición y no se puede esperar ningún otro problema en relación a la condición del segmento de datos, condición de ranura nula, etc.

Por lo tanto, en el módulo de estimación del canal r501 de la Fig. 62, se pueden usar pilotos en cada preámbulo cuando se realiza el tiempo de interpolación de los SP de los símbolos de datos. Por lo tanto, se pueden mejorar la adquisición de canal y la estimación del canal en los límites de la trama.

Ahora, en relación con los requisitos relacionados con el preámbulo y la estructura de piloto, hay un consenso en que las posiciones de los pilotos de preámbulo y los SP deberían coincidir independientemente de los canales adheridos; el número total de portadoras en el bloque de L1 debería ser divisible por la distancia de pilotos para evitar una estructura irregular en los límites de la banda; los bloques de L1 se deberían repetir en el dominio de la frecuencia y los bloques de L1 deberían ser siempre decodificables en una posición de ventana de sintonizador arbitrario. Los requisitos adicionales serían que las posiciones de piloto y los patrones se deberían repetir en periodos de 8 MHz; el desplazamiento de frecuencia de portadora correcta se debería estimar sin conocimiento de los canales adheridos y la decodificación del L1 (reordenación) es imposible antes de que se compense el desplazamiento de la frecuencia.

La Fig. 14 muestra una relación entre un símbolo de datos y un preámbulo cuando se usan estructuras de preámbulo como se muestra en la Fig. 19 y la Fig. 20. El bloque de L1 se puede repetir en periodos de 6 MHz. Para la decodificación del L1, se debería hallar tanto el desplazamiento de frecuencia como el patrón de desplazamiento de preámbulo. La decodificación del L1 no es posible en una posición de sintonizador arbitrario sin información de los canales adheridos y un receptor no puede diferenciar entre el valor de desplazamiento del preámbulo y el desplazamiento de frecuencia.

Por ello, un receptor, específicamente para que un eliminador de cabecera de Trama (r401), mostrado en la Fig. 30, realice la decodificación de la señal de L1, se necesita obtener una estructura de los canales adheridos. Debido a que la cantidad de desplazamiento del preámbulo esperada entre regiones sombreadas verticalmente en la Fig. 30 es conocida, el sintonizador de tiempo/frecuencia r505 en la Fig. 29 puede estimar el desplazamiento de frecuencia de portadora. En base a la estimación, la ruta de señalización del L1 r308-1 ~ r301-1 en la Fig. 31 puede decodificar el bloque L1.

La Fig. 15 muestra una relación entre los símbolos de datos y el preámbulo cuando se usa una estructura del preámbulo como se muestra en la Fig. 22. El bloque L1 se puede repetir en periodos de 8 MHz. Para la decodificación del L1, sólo se necesita hallar el desplazamiento de frecuencia y puede que no se requiera un conocimiento de los canales adheridos. El desplazamiento de frecuencia se estima fácilmente mediante el uso de la conocida secuencia de Secuencia Binaria Seudoraleatoria (PRBS). Como se muestra en la Fig. 48, se alinean el preámbulo y los símbolos de datos, por ello, puede llegar a ser innecesario una búsqueda de sincronismo adicional. Por lo tanto, para un receptor, específicamente para el eliminador de la cabecera de Trama r401 mostrado en la Fig. 63, es posible que sólo se necesite obtener la correlación de pico con la secuencia de decodificación piloto para realizar la decodificación de la señal L1. El sintonizador de tiempo/frecuencia r505 en la Fig. 29 puede estimar el desplazamiento de frecuencia de la portadora a partir de la posición del pico.

La Fig. 16 muestra un ejemplo de un perfil de retardo de los canales por cable.

A partir del punto de vista de diseño del piloto, el GI actual ya sobreprotege la dispersión del retardo del canal por cable. En el peor caso, el rediseño del modelo del canal puede ser una opción. Para repetir el patrón exactamente cada 8 MHz, la distancia piloto debería ser un divisor de 3584 portadoras ($z=32$ ó 56). Una densidad de piloto de $z=32$ incrementa la sobrecarga de pilotos, por ello, puede ser elegido $z=56$. Una cobertura de retardo ligeramente menor puede no ser importante en un canal por cable. Por ejemplo, puede ser de $8 \mu s$ para PP5' y de $4 \mu s$ para PP7' comparados con $9,3 \mu s$ (PP5) y $4,7 \mu s$ (PP7). Se pueden cubrir retardos no significativos por ambos patrones piloto incluso en el peor caso. Para la posición del piloto de preámbulo son necesarias no más de todas las posiciones de SP en el símbolo de datos.

Si el recorrido de retardo de -40 dB se puede ignorar, la dispersión de retardo real pueden llegar a ser de $2,5 \mu s$, $1/64 GI = 7 \mu s$, o $1/128 GI = 3,5 \mu s$. Esto muestra que el parámetro de distancia del piloto, $z=56$ puede ser un valor suficientemente bueno. Además, $z=56$ puede ser un valor conveniente para la estructura del patrón de piloto que permite una estructura de preámbulo mostrada en la Fig. 48.

La Fig. 17 muestra la estructura de piloto dispersa que usa $z=56$ y $z=112$ que se construye en el módulo de inserción del piloto 404 en la Fig. 42. Se proponen PP5' ($x=14$, $y=4$, $z=56$) y PP7' ($x=28$, $y=4$, $z=112$). Las portadoras de los límites se deberían insertar para el cierre de los límites.

Como se muestra en la Fig. 50, los pilotos se alinean en 8 MHz desde cada límite de la banda, cada posición de piloto y cada estructura de piloto se puede repetir cada 8 MHz. Por ello, esta estructura puede soportar la estructura de preámbulo mostrada en la Fig. 48. Además, se puede usar una estructura de piloto común entre los símbolos del piloto y de datos. Por lo tanto, el módulo de estimación del canal r501 en la Fig. 29 puede realizar una estimación del canal usando la interpolación en los símbolos de preámbulo y datos debido a que no puede tener lugar ningún patrón de piloto irregular, independientemente de la posición de la ventana que se decida para las localizaciones del segmento de datos. En este momento, el uso solamente de interpolación de frecuencia puede ser suficiente para compensar la distorsión del canal respecto a la dispersión de retardo. Si se realiza adicionalmente la interpolación de tiempos, se puede realizar una estimación del canal más precisa.

En consecuencia, en el nuevo patrón de pilotos propuesto, la posición del piloto y el patrón se podrá repetir en base a un periodo de 8 MHz. Se puede usar un único patrón de piloto tanto para los símbolos del preámbulo como de datos. La decodificación de L1 puede ser siempre posible sin un conocimiento de los canales adheridos. Además, el patrón de piloto propuesto puede no estar afectado en común con T2 debido a que se puede usar la misma estrategia de piloto de patrón de pilotos dispersos; el T2 ya usa 8 patrones de piloto diferentes y no hay ninguna complejidad de receptor significativa que se incremente mediante los patrones piloto modificados. Para una secuencia de codificación de piloto, el periodo del PRBS puede ser de 2047 (secuencia m); la generación de PRBS se puede reponer cada 8 MHz, de los que el periodo es de 1564; la tasa de repetición de pilotos de 56 puede ser también prima en común con 2047 y no se puede esperar ningún problema de PAPR.

La Fig. 18 muestra un ejemplo de un modulador en base a OFDM. Las corrientes de símbolos de entrada se pueden transformar en el dominio del tiempo mediante el módulo IFFT 501. Si es necesario, se puede reducir la relación de potencia de pico a promedio (PAPR) en el módulo de reducción de la PAPR 502. Para procedimientos de PAPR, se pueden usar la Extensión de la constelación activa (ACE) o la reserva de tono. El módulo de inserción de GI 503 puede copiar al menos parte del símbolo OFDM efectivo para rellenar el intervalo de guarda en una forma de prefijo cíclico.

El módulo de inserción del preámbulo 504 puede insertar el preámbulo en el frontal de cada trama transmitida de modo que un receptor pueda detectar una señal digital, trama y adquirir el desplazamiento de tiempo/frecuencia. En este tiempo, la señal del preámbulo puede realizar una señalización de la capa física tal como el tamaño del FFT (3 bits) y el tamaño del intervalo de Guarda (3 bits). El módulo de inserción del preámbulo 504 se puede omitir si el modulador es específicamente para DVB-C2.

La Fig. 19 muestra un ejemplo de una estructura de preámbulo para canales adheridos, generados en el módulo de inserción del preámbulo 504 en la Fig. 51. Un bloque L1 completo debería ser "siempre decodificable" en cualquier posición de la ventana de sintonía de 7, 61 MHz arbitraria y no debería tener lugar una pérdida de la señalización de L1 independientemente de la posición de la ventana del sintonizador. Como se muestra, los bloques de L1 se pueden repetir en el dominio de la frecuencia en periodos de 6 MHz. Los símbolos de datos pueden ser de canales adheridos cada 8 MHz. Si, para la decodificación de L1, un receptor usa un sintonizador tal como el sintonizador r603 representado en la Fig. 28 que usa un ancho de banda de 7, 61 MHz, el eliminador de la cabecera de Trama r401 en la Fig. 30 necesita redistribuir el bloque de L1 desplazado recibido cíclicamente (Fig. 20) a su forma original. Esta redistribución es posible debido a que el bloque de L1 se repite cada bloque de 6 MHz.

La Fig. 21 muestra un proceso para el diseño de un preámbulo más optimizado. La estructura del preámbulo de la Fig. 19 usa solamente 6 MHz del ancho de banda total del sintonizador de 7, 61 MHz para la decodificación de L1. En términos de eficiencia del espectro, el ancho de banda del sintonizador de 7, 61 MHz no está completamente utilizado. Por lo tanto, puede haber una optimización adicional en la eficiencia del espectro.

La Fig. 22 muestra otro ejemplo de estructura de preámbulo o estructura de símbolos de preámbulo para una eficiencia del espectro completa, generado en el módulo de Inserción de la Cabecera de Trama 401 en la Fig. 42. Justamente como los símbolos de datos, los bloques de L1 se puedan repetir en el dominio de la frecuencia en periodos de 8 MHz. Un bloque L1 completo aún es "siempre decodificable" en cualquier posición de la ventana de sintonía de 7, 61 MHz arbitraria. Después de la sintonía, los datos a 7, 61 MHz se pueden considerar como un código perforado virtualmente. Que tiene exactamente el mismo ancho de banda tanto para los símbolos de preámbulo como de datos y exactamente la misma estructura de piloto tanto para los símbolos de preámbulo como de datos puede maximizar la eficiencia del espectro. Otras características tal como la propiedad de desplazamiento cíclico y no envío del bloque L1 en caso de que no se pueda mantener sin cambios el segmento de datos. En otras palabras, el ancho de banda de los símbolos de preámbulo pueden ser idéntico al ancho de banda de los símbolos de datos o, como se muestra en la Fig. 57, el ancho de banda de los símbolos de preámbulo pueden ser el ancho de banda del sintonizador (aquí, es de 7, 61 MHz). El ancho de banda del sintonizador se puede definir como un ancho de banda que corresponde a un número de portadoras activas totales cuando se usa un canal simple. Esto es, el ancho de banda del símbolo de preámbulo puede corresponder al número de portadoras activas totales (aquí, éste es de 7, 61 MHz).

- La Fig. 23 muestra un código perforado virtualmente. Los datos de 7, 61 MHz entre el bloque de L1 de 8 MHz se pueden considerar como código perforado. Cuando un sintonizador r603 mostrado en la Fig. 28 usa un ancho de banda de 7, 61 MHz para la decodificación de L1, el eliminador de la cabecera de Trama r401 en la Fig. 30 necesita disponer el bloque de L1 desplazado cíclico, recibido en su forma original como se muestra en la Fig. 56. En este momento, se realiza la decodificación de L1 usando el ancho de banda completo del sintonizador. Una vez que se redispone el bloque L1, un espectro del bloque L1 redispuesto puede tener una región en blanco dentro del espectro como se muestra en el lado derecho superior de la Fig. 23 debido a que un tamaño original del bloque L1 es de 8 MHz de ancho de banda.
- Una vez que la región en blanco se rellena con ceros, tanto después del desentrelazado en el dominio de los símbolos por el desentrelazador de frecuencia r403 en la Fig. 30 como mediante el desentrelazador de símbolos r308-1 en la Fig. 31 como después del desentrelazado en el dominio de los bits por el desmapeador de símbolos r306-1, el multiplexor de bits r305-1 y el desentrelazador interno r304-1 en la Fig. 31, el bloque puede tener una forma que parece estar perforada como se muestra en el lado derecho inferior de la Fig. 23.
- Este bloque L1 se puede decodificar en el módulo de decodificación perforador/recortador r303-1 en la Fig. 31. Mediante el uso de esta estructura de preámbulo, se puede utilizar el ancho de banda del sintonizador completo, por tanto la eficiencia del espectro y la ganancia de codificación se pueden aumentar. Además, se pueden usar una estructura de piloto y ancho de banda idénticos para el preámbulo y los símbolos de datos.
- Además, el ancho de banda del preámbulo o el ancho de banda de los símbolos del preámbulo se ajustan como un ancho de banda del sintonizador como se muestra en la Fig. 25, (esta es 7, 61 MHz en el ejemplo), se puede obtener un bloque L1 completo después de la redistribución incluso sin perforación. En otras palabras, para una trama que tenga símbolos de preámbulo, en el que los símbolos del preámbulo tienen al menos un bloque de la capa 1 (L1), se puede decir que el bloque L1 tiene 3408 subportadoras activas y las 3408 subportadoras activas corresponden a los 7, 61 MHz de la banda de Frecuencia de Radio de 8 MHz (RF).
- Por ello, la eficiencia del espectro y el rendimiento de la decodificación de L1 se pueden maximizar. En otras palabras, en un receptor, se puede realizar la decodificación en el bloque del módulo de decodificación perforador/recortador r303-1 en la Fig. 31, después solamente de la realización del desentrelazado en el dominio de símbolos.
- En consecuencia, la estructura de preámbulo nuevo propuesta puede ser ventajosa en que es completamente compatible con el preámbulo usado previamente excepto en que el ancho de banda es diferente; los bloques L1 se repiten en periodos de 8 MHz; el bloque L1 se puede decodificar siempre independientemente de la posición de la ventana del sintonizador; el ancho de banda del sintonizador completo se puede usar para la decodificación del L1; la máxima eficiencia del espectro puede garantizar más ganancia de codificación; un bloque de L1 incompleto se puede considerar como código perforado; se puede usar la misma estructura de piloto simple tanto para preámbulo como para datos y se puede usar un ancho de banda idéntico tanto para preámbulo como para datos.
- La Fig. 26 muestra un ejemplo de un procesador analógico. Un DAC (601) puede convertir la señal de entrada digital en una señal analógica. Después de que la frecuencia de transmisión en el ancho de banda se eleva en el elevador 602 y la señal analógica se filtra a través del filtro analógico 603, se puede transmitir.
- La Fig. 27 muestra un ejemplo de un sistema receptor digital de acuerdo con una realización de la presente invención. La señal recibida se convierte en una señal digital en un procesador analógico r105. Un demodulador r104 puede convertir la señal en datos en el dominio de la frecuencia. Un analizador de tramas r103 puede eliminar los pilotos y las cabeceras para permitir la selección de información del servicio que necesita ser decodificada. Un demodulador de BICM r102 puede corregir errores en el canal de transmisión. Un procesador de salida r101 puede restaurar la corriente del servicio transmitida originalmente y la información de tiempos.
- La Fig. 28 muestra un ejemplo de un procesador analógico usado en el receptor. Un módulo Sintonizador/AGC (Controlador Automático de Ganancia) r603 puede seleccionar el ancho de banda de frecuencia deseado a partir de la señal recibida. El convertidor de reducción r602 puede restaurar la banda base. Un ADC r601 puede convertir la señal analógica en una señal digital.
- La Fig. 29 muestra un ejemplo de demodulador. Un detector de trama r506 puede detectar el preámbulo, comprobar si existe una señal digital correspondiente y detectar un comienzo de una trama. Un sintonizador de tiempos/frecuencia r505 puede realizar la sincronización en los dominios de tiempo y frecuencia. En este momento, para la sincronización el dominio del tiempo, se puede usar una correlación del intervalo de guarda. Para la sincronización en el dominio de la frecuencia, se puede usar la correlación o se puede estimar un desplazamiento a partir de la información de fase de una subportadora que se transmite en el dominio de la frecuencia. Un eliminador del preámbulo r504 puede eliminar el preámbulo del frontal de la trama del estado. Un eliminador del GI r503 puede eliminar el intervalo de guarda. Un módulo de FFT r501 puede transformar la señal en el dominio del tiempo en una señal en el dominio de la frecuencia. Un módulo de estimación/ecualización r501 puede compensar errores por distorsión de la estimación en el canal de transmisión usando el símbolo piloto. El eliminador del preámbulo r504 se puede omitir si el de modulador es específicamente para DVB-C2.

La Fig. 30 muestra un ejemplo de un analizador de trama. El eliminador de piloto (r404) puede eliminar el símbolo piloto. Un desentrelazador de frecuencia r403 puede realizar el desentrelazado en el dominio de la frecuencia. Un fusionador de símbolos OFDM r402 puede restaurar la trama de datos a partir de las corrientes de símbolos transmitidas en los símbolos OFDM. Un eliminador de la cabecera de trama r401 puede extraer la señalización de la capa física a partir de la cabecera de cada trama transmitida y eliminar la cabecera. La información extraída se puede usar como parámetros para los siguientes procesos en el receptor.

La Fig. 31 muestra un ejemplo de un demodulador BICM. La Fig. 31a muestra un recorrido de datos y la Fig. 31b muestra un recorrido de señalización de la L1. Un desentrelazador de símbolos r308 puede realizar el desentrelazado en el dominio de los símbolos. Un extractor ModCod r307 puede extraer los parámetros ModCod del frontal de cada trama de BB y dejar los parámetros disponibles para los siguientes procesos de demodulación adaptativa/variable y decodificación. Un desmapeador de Símbolos r306 puede desmapear las corrientes de símbolos de entrada en corrientes de prueba de relación de similitud (LLR). Las corrientes de los bits de LLR de salida se pueden calcular mediante el uso de una constelación usada en un mapeador de Símbolos 306 del transmisor como punto de referencia. En este punto, cuando se usan el MQAM o NU-MAQM anteriormente mencionados, mediante el cálculo tanto del eje I como del eje Q en el que se calcula el bit más próximo al MSB y mediante el cálculo tanto del eje I como del eje Q cuando se calcula el resto de los bits, se puede implementar un desmapeador de símbolos eficiente. Este procedimiento se puede aplicar a, por ejemplo, LLR Aproximado, LLR Exacto o de Decisión difícil.

Cuando se usa una constelación optimizada de acuerdo con la capacidad de la constelación y la tasa de códigos del código de corrección de error en el mapeador de Símbolos 306 del transmisor, el desmapeador de Símbolos r306 del receptor puede obtener una constelación usando la información de la tasa de códigos y de la capacidad de la constelación transmitida desde el transmisor. El multiplexor de bits r305 del receptor puede realizar una función inversa a la del demultiplexor de bits 305 del transmisor. El desentrelazado interior r304 y el desentrelazador exterior r302 del receptor puedan realizar funciones inversas a las del entrelazador interior (304) y entrelazador exterior 302 del transmisor, respectivamente para obtener el flujo de bits en su secuencia original. El desentrelazador exterior r302 se puede omitir si el demodulador del BICM es específicamente para DVB-C2.

El decodificador interior r303 y el decodificador exterior r301 del receptor pueden realizar los procesos de decodificación correspondientes al codificador interior 310 y codificador exterior 301 del transmisor, respectivamente, para corregir errores en el canal de transmisión. Los procesos similares realizados en el recorrido de datos se pueden realizar en el recorrido de señalización de L1, pero con diferentes parámetros r308-1 ~ r301-1. En este punto, como se explicó en la parte de preámbulo, se puede usar un módulo de codificación recortador/perforador r303-1 para la decodificación de la señal de L1.

La Fig. 32 muestra un ejemplo de la decodificación LDPC usando el módulo de recortado/perforación r303-1. Un demultiplexor r301a puede sacar por separado la parte de información y la parte de paridad del código sistemático a partir de los flujos de bits de entrada. Para la parte de información, el módulo de rellenado de ceros r302a puede realizar un rellenado de ceros de acuerdo con un número de flujos de bits de entrada del decodificador de LDPC, y para la parte de paridad, los flujos de bits de entrada para el decodificador de LDPC se puede generar mediante la desperforación de la parte perforada en el módulo de desperforación de paridad r303a. La decodificación del LDPC por el módulo r304a se puede realizar en los flujos de bits generados y se pueden eliminar los ceros en la parte de información mediante el eliminador de ceros r305a y producir su salida.

La Fig. 33 muestra un ejemplo de un procesador de salida. Un descodificador de BB r209 puede restaurar los flujos de bits codificados en el transmisor. Un repartidor r202 puede restaurar las tramas de BB que corresponden a múltiples PLP que se multiplexan y transmiten desde el transmisor de acuerdo con el recorrido del PLP. Para cada recorrido de PLP, un eliminador de la cabecera de BB r207-1~n puede eliminar la cabecera que se transmite en el frontal de cada trama de BB. Un decodificador de CRC r206-1~n puede realizar la decodificación de la CRC y hacer viable las tramas de BB disponibles para selección. Un módulo de inserción del paquete Nulo r205-1~n puede restaurar los paquetes Nulos que se eliminaron para una mayor eficiencia de la transmisión en su localización original. Un módulo de recuperación del retardo r204-1~n puede restaurar un retardo que existe entre cada recorrido de PLP.

Un módulo de recuperación del reloj de salida r203-1~n puede restaurar los tiempos originales de la corriente de servicio a partir de la información de tiempos transmitida desde el sintonizador de corriente de entrada 203-1~n. Un módulo de interfaz de salida r202-1~n puede restaurar los datos en el paquete TS/GS a partir de los flujos de bits de entrada que se segmentan en la trama de BB. Un procesador posterior de salida r201-1~n puede restaurar las múltiples corrientes de TS/GS en una corriente TS/GS, si es necesario. Los bloques sombreados mostrados en la Fig. 33 representan módulos que se pueden usar cuando se procesa un PLP simple en un momento y el resto de los bloques representan módulos que se pueden usar cuando se procesan múltiples PLP al mismo tiempo.

Los patrones de piloto de preámbulo se diseñaron cuidadosamente para evitar el aumento de PAPR, por ello, se ha de considerar si la tasa de repetición de L1 puede aumentar el PAPR. El número de bits de información de L1 varía dinámicamente de acuerdo con los canales adheridos, el número de PLP, etc. En detalle, es necesario considerar cosas tales como que el tamaño del bloque de L1 fijado puede introducir una sobrecarga innecesaria; la señalización

de L1 se debería proteger más fuertemente que los símbolos de datos y el entrelazado tiempos del bloque de L1 puede mejorar la robustez sobre los obstáculos del canal tales como la necesidad de ruido de impulsión.

Para una tasa de repetición del bloque L1 de 8 MHz, como se muestra en la Fig. 34, se exhibe una eficiencia de espectro completa (un aumento del Ancho de Banda del 26, 8%) con perforación virtual pero se puede incrementar el PAPR dado que el ancho de banda de la L1 es el mismo que el de los símbolos de datos. Para la tasa de repetición de 8 MHz, se puede usar un entrelazado de frecuencias del DVB-T2 de 4K-FFT para el grupo completo y el mismo patrón se puede repetir a sí mismo en un período de 8 MHz después del entrelazado.

Para una tasa de repetición del bloque L1 de 6 MHz, como se muestra en la Fig. 35, se puede exhibir una eficiencia de espectro reducida sin perforación virtual. Puede tener lugar un problema similar de PAPR que para el caso de 8 MHz dado que los anchos de banda de L1 y los símbolos de datos comparten el LCM=24 MHz. Para la tasa de repetición de 6 MHz, se puede usar un entrelazado de frecuencias del DVB-T2 de 4K-FFT para el grupo completo y el mismo patrón se puede repetir a sí mismo en un período de 24 MHz después del entrelazado.

La Fig. 36 muestra una nueva tasa de repetición del bloque L1 de 7, 61 MHz o ancho de banda de sintonizador completo. Se puede obtener una eficiencia del espectro completa (un aumento del BW del 26, 8%) sin perforación virtual. No puede haber un problema de PAPR dado que los anchos de banda de L1 y los símbolos de datos comparten el LCM=1704 MHz. Para la tasa de repetición de 7, 61 MHz, se puede usar un entrelazado de frecuencias del DVB-T2 de 4K-FFT para el grupo completo y el mismo patrón se puede repetir a sí mismo en un período de aproximadamente 1704 MHz después del entrelazado.

La Fig. 37 es un ejemplo de la señalización de L1 que se transmite en la cabecera de trama. Cada información en la señalización de L1 se puede transmitir al receptor y se puede usar como un parámetro de decodificación. Especialmente, la información se puede usar en el recorrido de la señal de L1 mostrado en la Fig. 31 y los PLP se pueden transmitir en cada segmento de datos. Se puede obtener una robustez incrementada para cada PLP.

La Fig. 39 es un ejemplo de un entrelazador de símbolos 308-1 como se muestra en el recorrido de señalización de L1 en la Fig. 4 y también puede ser un ejemplo de su desentrelazador de símbolos correspondiente r308-1 como se muestra en el recorrido de la señalización de L1 en la Fig. 31. Los bloques con líneas inclinadas representan los bloques de L1 y los bloques sólidos representan portadoras de datos. Los bloques de L1 se pueden transmitir no solamente dentro de un único preámbulo, sino también se pueden transmitir dentro de múltiples bloques OFDM. Dependiendo del tamaño del bloque de L1, el tamaño del bloque de entrelazado puede variar. En otras palabras, el num_L1_sym y el lapso del L1 pueden ser diferentes entre sí. Para minimizar una sobrecarga innecesaria, los datos se pueden transmitir dentro del resto de las portadoras de los símbolos de OFDM en el que se transmite el bloque de L1. En este punto, se puede garantizar una eficiencia del espectro completa debido a que la repetición del ciclo del bloque de L1 está todavía en un ancho de banda de sintonizador completo. En la Fig. 39, los números en los bloques con las líneas inclinadas representan el orden del bit dentro de un único bloque LDPC.

En consecuencia, cuando se escriben los bits en una memoria de entrelazado en la dirección de filas de acuerdo con un índice de símbolos como se muestra en la Fig. 72 y se lee en la dirección de las columnas de acuerdo con un índice de portadora, se puede obtener un efecto de entrelazado de bloques. En otras palabras, se puede entrelazar un bloque LDPC en el dominio del tiempo y en el dominio de la frecuencia y a continuación se puede transmitir. El num_L1_sym puede ser un valor predeterminado, por ejemplo, se puede establecer un número entre 2-4 como el número de símbolos OFDM. En este punto, para aumentar la granularidad del tamaño de bloques de L1, se puede usar para la protección de L1 un código de LDPC perforado/recortado que tenga una longitud mínima de palabra de código.

La Fig. 40 es un ejemplo de una transmisión de bloque L1. La Fig. 40 ilustra la Fig. 39 en el dominio de las tramas. Como se muestra en el lado izquierdo de la Fig. 40, los bloques L1 se pueden extender en un ancho de banda de sintonizador completo o, como se muestra en el lado derecho de la Fig. 40, los bloques L1 se pueden extender parcialmente y el resto de las portadoras se pueden usar para portadoras de datos. En cualquier caso, se puede ver que la tasa de repetición del bloque L1 puede ser idéntica a un ancho de banda de sintonizador completo. Adicionalmente, para los símbolos OFDM que usan la señalización de L1 que incluye el preámbulo, solamente se puede realizar el entrelazado de símbolos mientras que no se permite la transmisión de datos en esos símbolos OFDM. En consecuencia, para los símbolos OFDM usados para la señalización de L1, un receptor puede decodificar la L1 mediante la realización del desentrelazado sin la decodificación de datos. En este punto, el bloque L1 puede transmitir la señalización de L1 de la trama actual o la señalización de L1 de una trama posterior. En el lado del receptor, los parámetros de L1 decodificados a partir de recorrido de decodificación de la señalización de L1 mostrado en la Fig. 31 se puede usar para procesos de decodificación para el recorrido de los datos desde el analizador de tramas de la trama posterior.

En resumen, en un transmisor, los bloques de entrelazado de la región de L1 se puede realizar mediante los bloques de escritura en una memoria en una dirección de filas y la lectura de los bloques escritos desde la memoria en una dirección de columnas. En un receptor, el desentrelazado de los bloques de la región L1 se puede realizar mediante la escritura de bloques en una memoria en una dirección de columnas y la lectura de los bloques escritos desde la

memoria en una dirección de filas. Las direcciones de lectura y escritura del transmisor y receptor se pueden intercambiar.

Cuando se realiza una simulación con suposiciones tales como $CR=1/2$ para la protección de L1 para la agrupación de T2; mapeado de símbolos 16-QAM; densidad de pilotos de 6 en el preámbulo; número de LDPC cortos implica que se realiza una cantidad requerida de perforación/recortado, los resultados o conclusiones tales como preámbulo único para la transmisión de L1 puede no ser suficiente; el número de símbolos OFDM depende de la cantidad del tamaño de bloque de L1; la palabra de código LDPC más corta (por ejemplo información de 192 bis) entre el código recortado/perforado se puede usar por flexibilidad y una mejor granularidad; y se puede añadir el Rellenado si se requiere con una sobrecarga despreciable, se puede obtener. El resultado se resume en la Fig. 38.

En consecuencia, para una tasa de repetición del bloque L1, un ancho de banda de sintonizador completo sin perforación virtual puede ser una buena solución y aún puede no surgir un problema de PAPR con la eficiencia de espectro completo. Para la señalización de L1, una estructura de señalización eficiente puede permitir una configuración máxima en un entorno de 8 canales adheridos, 32 supresiones, 256 segmentos de datos y 256 PLP. Para la estructura del bloque de L1, se puede implementar una señalización de L1 flexible de acuerdo con el tamaño del bloque de L1. Se puede realizar una entrelazado de tiempos para una mejor robustez para la agrupación de T2. Una menor sobrecarga puede permitir la transmisión de datos en el preámbulo.

El entrelazado de bloques del bloque L1 se puede realizar para una mejor robustez. El entrelazado se puede realizar con un número predefinido fijo de símbolos de L1 (num_L1_sym) y un número de portadoras extendidas por L1 como un parámetro ($L1_span$). La misma técnica se usa para el entrelazado del preámbulo P2 en DVB-T2.

Se puede usar el bloque de L1 de tamaño variable. El tamaño se puede adaptar a la cantidad de bits de señalización de L1, dando como resultado una sobrecarga reducida. Se puede obtener una eficiencia de espectro completa sin problemas de PAPR. Menos de 7, 61 MHz de repetición puede significar que se puede enviar más redundancia pero no se usa. No puede surgir con ningún problema de PAPR dada la tasa de repetición de 7, 61 MHz para el bloque de L1.

La Fig. 41 es otro ejemplo de señalización de L1 transmitida dentro de una cabecera de trama. La Fig. 41 es diferente de la Fig. 37 en que el campo de $L1_span$ que tiene 12 bits se divide en 2 campos. En otras palabras, el campo $L1_span$ se divide en un $L1_column$ que tiene 9 bits y un $L1_row$ que tiene 3 bits. El $L1_column$ representa el índice de la portadora que abarca el L1. Debido a que el segmento de datos comienza y finaliza cada 12 portadoras, que es la densidad de piloto, los 12 bits de cabecera adicional se pueden reducir en 3 bits para alcanzar los 9 bits.

El $L1_row$ representa el número de símbolos OFDM en el que se extiende el L1 cuando se aplica entrelazado de tiempos. En consecuencia, el entrelazado de tiempos que puede realizar dentro de un área de $L1_columns$ multiplicado por $L1_rows$. Alternativamente, se puede transmitir un tamaño total de bloques de L1 de modo que el $L1_span$ mostrado en la Fig. 37 se puede usar cuando no se realiza entrelazado de tiempos. Para tal caso, el tamaño del bloque de L1 es de 11.776×2 bits en el ejemplo, por ello 15 bits es suficiente. En consecuencia, el campo de $L1_span$ se puede construir con 15 bits.

La Fig. 42 es un ejemplo de entrelazado/desentrelazado de frecuencia o tiempo. La Fig. 42 muestra una parte de una trama de transmisión completo. La Fig. 42 también muestra la adhesión de múltiples anchos de banda de 8 MHz. Una trama puede consistir en un preámbulo que transmite bloques L1 y un símbolo de datos que transmite datos. Las diferentes clases de símbolos de datos representan segmentos de datos para diferentes servicios. Como se muestra en la Fig. 42, el preámbulo transmite bloques de L1 para cada 7, 61 MHz.

Para el preámbulo, se realiza el entrelazado de frecuencia o tiempo dentro de los bloques de L1 y no se realiza entre bloques de L1. Esto es, para el preámbulo, se puede decir que el entrelazado se realiza en el nivel del bloque de L1. Esto permite la decodificación de los bloques de L1 mediante la transmisión de los bloques de L1 dentro del ancho de banda de una ventana del sintonizador incluso cuando la ventana del sintonizador se ha trasladado a una localización aleatoria dentro de un sistema de canales adheridos.

Para la decodificación del símbolo de datos en un ancho de banda de la ventana del sintonizador aleatoria, no debería tener lugar el entrelazado entre los segmentos de datos. Esto es, para segmentos de datos, se puede decir que el entrelazado se realiza en un nivel del segmento de datos. En consecuencia, el entrelazado de la frecuencia y el entrelazado del tiempo se debían realizar dentro de un segmento de datos. Por lo tanto, un entrelazado de símbolos 308 en un recorrido de datos de un módulo BICM del transmisor como se muestra en la Fig. 4 puede realizar el entrelazado de símbolos para cada segmento de datos. Un entrelazado de símbolos 308-1 en un recorrido de señal de L1 puede realizar el entrelazado de símbolos para cada bloque de L1.

Un entrelazado de frecuencia 403 mostrado en la Fig. 9 necesita realizar el entrelazado en el preámbulo y los símbolos de datos por separado. Específicamente, para el preámbulo, el entrelazado de frecuencia se puede realizar para cada bloque de L1 para cada símbolo de datos, el entrelazado de frecuencia se puede realizar para cada segmento de datos. En este punto, el entrelazado de tiempos en el recorrido de datos o recorrido de la señal de L1 no se puede realizar considerando el modo de baja latencia.

La Fig. 43 es una tabla que analiza la sobrecarga de la señalización de L1 que se transmite en una cabecera FECFRAME en la Inserción de Cabecera de ModCod (307) sobre el recorrido de datos del módulo BICM como se muestra en la Fig. 37. Como se ve en la Fig. 76, para un bloque LDPC corto (tamaño = 16.200), puede tener lugar una sobrecarga máxima del 3, 3% lo que puede no ser despreciable. En el análisis, se suponen 45 símbolos para la protección del FECFRAME y el preámbulo es una trama C2 específico de señalización de L1 y la cabecera de FECFRAME es una señalización de L1 específica de FECFRAME, es decir, Mod, Cod e identificador PLP.

Para reducir la sobrecarga de L1, se pueden considerar los enfoques de acuerdo con los dos tipos de segmentos de datos. Para el tipo ADMNCM y el caso de PLP múltiples, la trama se puede mantener el mismo que para la cabecera FECFRAME. Para el tipo ADMNCM y los casos de PLP simple, se puede eliminar el identificador del PLP de la cabecera FECFRAME, dando como resultado hasta el 1, 8% de reducción de sobrecarga. Para el tipo CCM y los casos de múltiples PLP, el campo Mod/Cod se puede eliminar de la cabecera FECFRAME, dando como resultado hasta el 1, 5% de reducción de sobrecarga. Para CCM y los casos de PLP simple, no se requiere cabecera FECFRAME, por ello, se puede obtener una reducción de la sobrecarga hasta del 3, 3%.

En una señalización de L1 recortada, se puede transmitir o bien el Mod/Cod (7 bits) o el identificador de PLP (8 bits), pero puede ser demasiado corto para obtener ninguna ganancia de codificación. Sin embargo, es posible que no se requiera un sintonizador debido a que los PLP se pueden alinear con la trama de transmisión de C2; cada ModCod de cada PLP puede ser conocido a partir del preámbulo y un cálculo simple puede permitir la sincronización con el FECFRAME específico.

La Fig. 44 muestra una estructura de una cabecera FECFRAME para minimizar la sobrecarga. En la Fig. 44, los bloques con las líneas inclinadas y el Constructor de FECFRAME representan un diagrama de bloques detallado del módulo de Inserción de la Cabecera ModCod 307 en el recorrido de datos del módulo BICM como se muestra la Fig.

4. Los bloques sólidos representan un ejemplo de un módulo de codificación interior 303, entrelazado interior 304, demultiplexor de bits 305 y mapeador de símbolos 306 en el recorrido que los datos del módulo BICM como se muestra en la Fig. 4. En este punto, la señalización de L1 recortada se puede realizar debido a que el CCM no requiere un campo de Mod/Cod y el PLP simple no requiere un identificador de PLP. En esta señal de L1 con un número reducido de bits, la señal de L1 se puede repetir tres veces en el preámbulo y se puede realizar una modulación BPSK, por ello, es posible una señalización robusta. Finalmente, el módulo de inserción de la Cabecera ModCod 307 puede insertar la cabecera generada dentro de cada trama FEC. La Fig. 51 muestra un ejemplo del extractor de ModCod r307 en el recorrido de los datos del módulo de demodulación BICM mostrado en la Fig. 31.

Como se muestra en la Fig. 51, la cabecera FECFRAME se puede analizar en el analizador r301b, a continuación los símbolos que transmiten información idéntica en símbolos repetidos se pueden retrasar, alinear y a continuación combinar en el módulo de combinación de Rake r302b. Finalmente, cuando se realiza una demodulación BPSK en el módulo r303b, el campo de la señal de L1 recibido se puede restaurar y este campo de señal de L1 es restaurado se puede enviar al controlador del sistema para ser usado como parámetros para la decodificación. El FECFRAME analizado se puede enviar al desmapeador de símbolos.

La Fig. 45 muestra un rendimiento de la tasa de error de bits (BER) de la protección de L1 anteriormente mencionada. Se puede ver que aproximadamente se obtienen 4, 8 dB de ganancia de SNR a través de la repetición en tres veces. La SNR requerida es de 8, 7 dB en BER = 1E-11.

La Fig. 46 muestra ejemplos de trama de transmisión y las estructuras de la trama FEC. Las estructuras de la trama FEC mostradas en el lado superior derecho de la Fig. 46 representan la cabecera FECFRAME insertada mediante el módulo de Inserción de Cabecera ModCod 307 en la Fig. 4. Se puede ver que dependiendo de varias combinaciones de condiciones, es decir, tipo CCM o ADMNCM y PLP simple o múltiple, se pueden insertar diferentes tamaños de cabeceras. O, puede no insertarse ninguna cabecera. Las tramas de transmisión formadas de acuerdo con los tipos de segmentos de datos y mostrados en el lado inferior izquierdo de la Fig. 46 se pueden formar mediante el módulo de Inserción de la Cabecera de Trama 401 del constructor de Tramas como se muestra en la Fig. 9 y el fusionador/segmentador 208 del procesador de entrada mostrado en la Fig. 2. En este punto, se puede transmitir el FECFRAME de acuerdo con los diferentes tipos de segmentos de datos. Usando este procedimiento, se puede reducir un máximo del 3, 3% de la sobrecarga. En el lado superior derecho de la Fig. 79, se muestran cuatro tipos diferentes de estructuras, pero un experto en la materia comprenderá que éstos son solamente ejemplos y que cualquiera de estos tipos y sus combinaciones se pueden usar para el segmento de datos.

En el lado del receptor, el eliminador de la cabecera de Trama r401 del módulo analizador de Tramas como se muestra en la Fig. 30 y el extractor de ModCod r307 del módulo de modulador de BCIM mostrado la Fig. 31 puede extraer un parámetro del campo ModCod que se requiere para la decodificación. En este punto, de acuerdo con los tipos de segmento de datos de la transmisión se pueden extraer los parámetros de la trama. Por ejemplo, para el tipo CCM, los parámetros se pueden extraer a partir de la señalización de L1 que se transmite en el preámbulo y para el tipo ACMNCM, se pueden extraer los parámetros a partir de la cabecera FECFRAME.

Como se muestra en el lado superior derecho de la Fig. 79, la estructura fecframe se puede dividir en dos grupos, en los que el primer grupo son las tres primeras estructuras de trama con cabecera y el segundo grupo es la última estructura de trama sin cabecera.

La Fig. 47 muestra un ejemplo de la señalización de L1 que se puede transmitir dentro del preámbulo por el módulo de inserción de la cabecera de Trama 401 del módulo constructor de Tramas mostrado la Fig. 42. Esta señalización de L1 es diferente de la señalización de L1 previa en que el tamaño del bloque de L1 se puede transmitir en bits (L1_size, 14 bits); la conexión/desconexión del entrelazado de tiempos sobre el segmento de datos es posible (dslice_time_intrlv, 1 bit) y mediante la definición del tipo de segmento de datos (dslice_type, 1 bit), se reduce la sobrecarga de la señalización de L1. En este punto, cuando el tipo de segmento de datos es CCM, el campo de Mod/Cod se puede transmitir dentro del preámbulo en lugar de dentro de la cabecera FECFRAME (plp_mod (3 bits), plp_fec_type (1 bit), plp_cod (3 bits)).

En el lado del receptor, el módulo de decodificación interior del recortador/perforador r303-1 del demodulador de BICM como se muestra en la Fig. 31 puede obtener el primer bloque LDPC, que tiene un tamaño de bloque de L1 fijo, transmitido dentro del preámbulo, a través de la decodificación. Los números del tamaño del resto de los bloques de LDPC se pueden obtener también.

Se puede usar entrelazado de tiempos cuando se necesitan múltiples símbolos OFDM para la transmisión de L1 o cuando hay un segmento de datos entrelazado en el tiempo. Es posible una conexión/desconexión flexible del entrelazado de tiempos con un marcador de entrelazado. Para el entrelazado de tiempos del preámbulo, se puede requerir un marcador de entrelazado de tiempos (1 bit) y un número de símbolos OFDM entrelazados (3 bits), por ello, un total de 4 bits se pueden proteger mediante una forma similar a una cabecera FECFRAME recortada.

La Fig. 48 muestra un ejemplo de la señalización previa de L1 que se puede realizar en el módulo de inserción de la Cabecera ModCod 307-1 en el recorrido de los datos del módulo de BCIM mostrado en la Fig. 4. Los bloques con las líneas inclinadas y el Constructor del Preámbulo son ejemplos del módulo de Inserción de Cabecera de ModCod 307-1 en el recorrido de señalización de L1 del módulo de BCIM mostrado en la Fig. 4. Los bloques sólidos son ejemplos del módulo de inserción de cabecera de Trama 401 del constructor de Tramas como se muestra en la Fig. 42.

También, los bloques sólidos pueden ser ejemplos del módulo de decodificación interior recortado/perforado 303-1, el entrelazador interior 304-1, el demultiplexor de bits 305-1 y del mapeado de símbolos 306-1 en el recorrido de la señalización de L1 del módulo de BCIM mostrado en la Fig. 4.

Como se ve en la Fig. 48, la señal de L1 que se transmite en el preámbulo se puede proteger usando codificación LDPC recortada/perforada. Se pueden insertar los parámetros relacionados dentro de la Cabecera en una forma de señalización previa de L1. En este punto, solamente los parámetros de entrelazado de tiempo se pueden transmitir en la Cabecera del preámbulo. Para asegurar una mayor robustez, se puede realizar una repetición en cuatro veces. En el lado del receptor, para ser capaces de decodificar la señal de L1 que se transmite en el preámbulo, el extractor de ModCod 307-1 en el recorrido de señalización de L1 del modulador de BICM como se muestra la Fig. 31 necesita usar el módulo de decodificación mostrado en la Fig. 18. En este punto, debido a una repetición de cuatro veces a diferencia de la decodificación previa de la cabecera FECFRAME, se requiere un proceso receptor Rake que sincroniza los símbolos repetidos cuatro veces y añade los símbolos.

La Fig. 49 muestra una estructura del bloque de señalización de L1 que se transmite a partir del módulo de inserción de la cabecera de Trama 401 del módulo constructor de Tramas como se muestra en la Fig. 42. Se muestra un caso en el que no se usa entrelazado de tiempos en un preámbulo. Como se muestra en la Fig. 49, se pueden transmitir diferentes clases de bloques LDPC en el orden de las portadoras. Una vez que se ha formado y transmitido un símbolo OFDM entonces se forma y transmite un símbolo OFDM siguiente. Para el último símbolo OFDM a ser transmitido, si hay cualquier portadora restante, esas portadoras se pueden usar para la transmisión de datos o se pueden rellenar con datos sin sentido. El ejemplo en la Fig. 49 muestra un preámbulo que comprende tres símbolos OFDM. En un lado del receptor, para este caso sin entrelazado, el desentrelazador de símbolos r308-1 en el recorrido de señalización de L1 del demodulador BCIM como se muestra en la Fig. 31 se puede omitir.

La Fig. 50 muestra un caso en el que se realiza el entrelazado de tiempos de L1. Como se muestra en la Fig. 50, el entrelazado de bloques se puede realizar en un modo de formación del símbolo OFDM para índices de portadora idénticos y a continuación formar unos símbolos OFDM para los siguientes índices de portadora. Como en el caso en que no se realiza entrelazado, si hay cualquier portadora restante, esas portadoras se pueden usar para transmisión de datos o se pueden rellenar con datos sin sentido. En un lado receptor, para este caso sin entrelazado, el desentrelazador de símbolos r308-1 en el recorrido de señalización de L1 del demodulador BCIM mostrado en la Fig. 31 puede realizar un desentrelazado de bloques mediante la lectura de los bloques LDPC en un orden creciente de números de los bloques LDPC.

Además, puede haber al menos 2 tipos de segmentos de datos. El tipo de segmento de datos 1 tiene dslice_type = 0 en los campos de señalización de L1. Este tipo de segmento de datos no tiene cabecera de XFECFrame y tiene sus valores mod/cod en los campos de señalización de L1. El tipo de segmento de datos 2 tiene dslice_type = 1 en los

campos de señalización de L1. Este tipo de segmento de datos tiene la cabecera XFECFrame y tiene sus valores mod/cod en la cabecera XFECFrame.

XFECFrame significa XFEC trama compleja de corrección de errores directos y mod/cod significa tipo de modulación/tasa de códigos.

5 En un receptor, un analizador de tramas puede formar una trama a partir de las señales demoduladas. La trama tiene símbolos de datos y los símbolos de datos pueden tener un primer tipo de segmento de datos que tiene una cabecera XFECFrame y un segundo tipo de segmento de datos que tiene XFECFrame sin cabecera XFECFrame. También, un receptor puede extraer un campo para indicación de si realizar un desentrelazado de tiempos en los
10 símbolos de preámbulo o no para realizar un desentrelazado de tiempos en los símbolos de preámbulos, a partir de la L1 de los símbolos de preámbulo.

En un transmisor, un constructor de tramas puede construir una trama. Los símbolos de datos de la trama comprenden un primer tipo de segmento de datos que tiene un XFECFrame y una cabecera XFECFrame y un segundo tipo de segmento de datos que tiene un XFECFrame sin cabecera XFECFrame. Además, se puede insertar
15 en la L1 de los símbolos de preámbulo un campo para una indicación de si realizar o no una entrelazado de tiempos sobre los símbolos del preámbulo para realizar un entrelazado de tiempos sobre los símbolos de preámbulos.

Finalmente, para códigos recortados/perforados para el módulo de inserción de cabecera de Trama 401 del constructor de Tramas mostrado en la Fig. 9, se puede determinar un mínimo tamaño de palabra de código que puede obtener ganancia de codificación y se puede transmitir en un primer bloque LDPC. En esa forma, el resto de los tamaños de bloques LDPC se pueden obtener a partir del tamaño de bloque de L1 transmitido.

20 La Fig. 52 muestra otro ejemplo de la señalización previa de L1 que se puede transmitir a partir del módulo de inserción de Cabecera ModCod 307-1 sobre el recorrido de señalización de L1 del módulo BCIM mostrado en la Fig.

4. La Fig. 52 es diferente de la Fig. 4 en que se ha modificado el mecanismo de protección de la parte de Cabecera. Como se ve en la Fig. 52, la información del tamaño de bloque de L1, L1_size (14 bits) no se transmite en el bloque de L1, sino que se transmite en la Cabecera. En la Cabecera, la información de entrelazado de 4 bits se puede
25 transmitir también. Para el total de 18 bits de entrada, se usa el código BCH (45, 18) que produce la salida de 45 bits y se copia en los dos recorridos y, finalmente, se mapea el QPSK. Para el recorrido Q, se puede realizar un desplazamiento cíclico de 1 bit para una diversidad de ganancia y se puede realizar una modulación PRBS de acuerdo con la palabra de sincronización. El total de 45 símbolos QPSK se pueden obtener a partir de estas entradas de recorrido I/Q. En este punto, si la profundidad de entrelazado en el tiempo se fija como un número de preámbulos que se requieren para transmitir el bloque de L1, L1_span (3 bits) que indican que puede no necesitarse
30 transmitir una profundidad de entrelazado de tiempo. En otras palabras, solamente se puede transmitir el marcador de entrelazado conectado/desconectado (1 bit). En el lado receptor, mediante la comparación solamente de un número de preámbulos transmitidos, sin el uso de L1_span, desentrelazado de tiempos.

La Fig. 53 muestra un ejemplo de planificación del bloque de señalización de L1 que se transmite en el preámbulo. Si un tamaño de la información de L1 que se puede transmitir en un preámbulo es Nmax, cuando el tamaño de L1 es más pequeño que Nmax, un preámbulo puede transmitir la información. Sin embargo, cuando el tamaño de L1 es mayor que Nmax, la información de L1 puede ser igualmente dividida de modo que el subbloque de L1 dividido sea más pequeño que Nmax, a continuación se puede transmitir el subbloque de L1 dividido en un preámbulo. En este
35 punto, en una portadora que no se use debido a que la información de L1 sea más pequeña que Nmax, no se transmiten datos.

En su lugar, como se muestra en la Fig. 55, se puede reforzar la potencia de las portadoras en las que se transmiten los bloques de L1 hasta mantener la potencia de la señal de preámbulo total igual a la potencia de los símbolos de datos. El factor de refuerzo de potencia se puede variar dependiendo del tamaño de L1 transmitido y un transmisor y un receptor pueden tener un valor establecido para este factor de refuerzo de potencia. Por ejemplo, si solamente se
40 usan la mitad del total de portadoras, el factor de refuerzo de potencia puede ser dos.

La Fig. 54 muestra un ejemplo de señalización previa de L1 en el que se considera el refuerzo de la potencia. Cuando se compara con la Fig. 52, se puede ver que la potencia del símbolo QPSK se puede reforzar y enviar al constructor de preámbulo.

La Fig. 56 muestra otro ejemplo del extractor ModCod r307-1 en el recorrido de señalización de L1 del módulo demodulador mostrado en la Fig. 31. A partir del símbolo de preámbulo, el FECFRAME de señalización de L1 se puede sacar dentro del desmapeador de símbolos y solamente se puede decodificar la parte de cabecera.

Para el símbolo de cabecera de entrada, se puede realizar el desmapeado QPSK y se puede obtener el valor de la Relación de Verosimilitud de Registro (LLR). Para el recorrido Q, se puede realizar la demodulación PRBS de acuerdo con la palabra de sincronización y se puede realizar el proceso inverso al desplazamiento cíclico de 1 bit
55 para restauración.

Estos dos valores del recorrido I/Q alineados se pueden combinar y se puede obtener la ganancia de SNR. La salida de la decisión de hardware se puede introducir en el decodificador BCH. El decodificador BCH puede restaurar los 18 bits del L1 previos a partir de los 45 bits de entrada.

La Fig. 57 muestra una parte contraria, el extractor de ModCod de un receptor. Cuando se compara con la Fig. 56, se puede realizar el control de potencia en los símbolos de entrada del desmapeador QPSK para restaurarlo a partir del nivel de potencia reforzado por un transmisor a su valor original. En este punto, se puede realizar un control de potencia considerando un número de portadoras usadas para la señalización de L1 en un preámbulo y tomando una inversa del factor de refuerzo de potencia obtenido de un transmisor. El factor de refuerzo de potencia fija la potencia del preámbulo y la potencia de los símbolos de datos idénticos entre sí.

La Fig. 58 muestra un ejemplo de la sincronización previa de L1 que se puede realizar en el extractor ModCod r3071 en el recorrido de señalización de L1 del módulo de demodulación BICM mostrado en la Fig. 31. Este es un proceso de sincronización para obtener una posición de inicio de la Cabecera en un preámbulo. Los símbolos de entrada pueden estar desmapeados en QPSK a continuación para la salida del recorrido Q, se puede realizar una inversa del desplazamiento cíclico de 1 bit y se puede realizar la alineación. Se pueden multiplicar dos valores de los recorridos I/Q y los valores modulados por la señalización previa de L1 se pueden demodular. Por ello, la salida del multiplicador puede expresar solamente el PRBS que es una palabra de sincronización. Cuando la salida se correlaciona con una secuencia PRBS conocida, se pueden obtener picos de correlación en la Cabecera. Por ello, se puede obtener una posición de inicio de la Cabecera en un preámbulo. Si es necesario, el control de potencia que se realiza para restaurar el nivel de potencia original, como en la Fig. 57, se puede realizar sobre la entrada del desmapeador QPSK.

La Fig. 59 muestra otro ejemplo de campo de Cabecera del bloque de L1 que se envía al módulo de inserción de Cabecera 307-1 en el recorrido de señalización de L1 del módulo BICM como se muestra en la Fig. 4. La Fig. 59 es diferente de la Fig. 52 en que L1_span que representa la profundidad del entrelazado de tiempo se reduce a 2 bits y los bits reservados se incrementan en 1 bit. Un receptor puede obtener el parámetro de entrelazado de tiempos del bloque de L1 a partir del L1_span transmitido.

La Fig. 60 muestra procesos de división por igual de un bloque de L1 en tantas porciones como el número de preámbulos, la inserción a continuación de la cabecera dentro de cada uno de los bloques de L1 divididos y la asignación a continuación de los bloques de L1 insertados en la cabecera en un preámbulo. Esto se puede realizar cuando se realiza un entrelazado de tiempos con un número de preámbulos en el que el número de preámbulos es mayor que un número mínimo de preámbulos que se requieren para transmitir el bloque de L1. Esto se puede realizar en el bloque de L1 en el recorrido de señalización de L1 del módulo BCIM como se muestra en la Fig. 37. El resto de las portadoras, después de la transmisión de los bloques de L1, pueden tener unos patrones de repetición cíclicos en lugar de ser rellenadas con ceros.

La Fig. 61 muestra un ejemplo del Desmapeador de Símbolos r306-1 del módulo de demodulación BCIM como se muestra en la Fig. 31. Para un caso en el que los bloques FEC de L1 se repiten como se muestra en la Fig. 60, cada punto de comienzo del bloque FEC de L1 se puede alinear, combinado en el módulo r301f y a continuación desmapeado QAM en el desmapeador QAM r302f para obtener una ganancia de diversidad y una ganancia de SNR. En este punto, el combinador puede incluir procesos de alineación y sumar cada bloque de FEC de L1 y divide el bloque de FEC de L1 añadido. Para ocasiones en que sólo parte del último bloque de FEC se repite como se muestra en la Fig. 60, solamente la parte dividida se puede dividir en tantas partes como el número de la cabecera del bloque FEC y la otra parte se puede dividir por un valor que es 1 menor que un número de la cabecera del bloque FEC. En otras palabras, el número divisor corresponde a un número de portadoras que se añade a cada portadora.

La Fig. 65 muestra otro ejemplo de planificación del bloque de L1. La Fig. 65 es diferente de la Fig. 60 en que, en lugar de realizar un rellenado con ceros o repetición cuando los bloques de L1 no llenan un símbolo OFDM, el símbolo OFDM se puede rellenar con redundancia de paridad mediante la realización de una perforación menor o codificación de recorte/perforación en el transmisor. En otras palabras, cuando el módulo de perforación de paridad 304c se realiza en la Fig. 5, la tasa de códigos efectiva se puede determinar de acuerdo con la relación de perforación, por ello, mediante la perforación de tan pocos bits como se hayan añadido, la tasa de códigos efectiva se puede disminuir y se puede obtener una mejor ganancia de código. El módulo de desperforación de paridad r303a de un receptor como se muestra en la Fig. 32 puede realizar la desperforación considerando la redundancia de paridad menos perforada. En este punto, debido a que un receptor y un transmisor puedan tener información del tamaño del bloque de L1 total, se puede calcular la relación de perforación.

La Fig. 62 muestra otro ejemplo del campo de señalización de L1. La Fig. 62 es diferente de la Fig. 41 en que, para un caso en que el tipo de segmento de datos es CCM, se puede transmitir una dirección de comienzo (21 bits) del PLP. Esto puede permitir que un FECFRAME de cada PLP forme una trama de transmisión, sin que el FECFRAME esté alineado con una posición de inicio de una trama de transmisión. Por ello, la sobrecarga de rellenado, que puede tener lugar cuando es estrecho el ancho del segmento de datos, se puede eliminar. Un receptor, cuando el tipo de segmento de datos es CCM, puede obtener información de ModCod a partir del preámbulo en el recorrido de señalización de L1 del módulo de demodulación BICM como se muestra en la Fig. 31, en lugar de obtenerlo a partir

de la cabecera FECFRAME. Además, incluso cuando tiene lugar un cambio a una localización aleatoria de la trama de transmisión, la sincronización del FECFRAME se puede realizar sin retardo debido a que la dirección de inicio del PLP se puede obtener ya a partir del preámbulo.

La Fig. 63 muestra otro ejemplo de campos de señalización de L1 que pueden reducir la sobrecarga de dirección lado del PLP.

La Fig. 64 muestra el número de símbolos de QAM que corresponden a un FECFRAME dependiendo de los tipos de modulación. En este punto, el máximo común divisor del símbolo QAM es 135, por ello, se puede reducir una sobrecarga de $\log_2(135) = 7$ bits. Por ello, la Fig. 63 es diferente de la Fig. 62 en que un número de bits del campo PLP_start se pueden reducir a partir de los 21 bits a 14 bits. Esto es un resultado de considerar 135 símbolos como un único grupo y direccionar el grupo. Un receptor puede obtener un índice de portadora OFDM en la que comienza el PLP en una trama de transmisión después de obtener el valor del campo PLP_start y multiplicarlo por 135.

La Fig. 66 y la Fig. 68 muestran ejemplos de un entrelazado de símbolos 308 que puede entrelazar en el tiempo símbolos de datos que se envían desde el módulo de inserción de Cabecera ModCod 307 en el recorrido de datos del módulo BCIM como se muestra en la Fig. 4.

La Fig. 66 es un ejemplo de entrelazador de bloques para el entrelazado en el tiempo que puede funcionar en base a segmentos de datos. El valor de fila significa un número de células de contenido útil en cuatro de los símbolos OFDM dentro de un segmento de datos. El entrelazado en base a los símbolos OFDM puede no ser posible debido a que el número de células puede cambiar entre las células OFDM adyacentes. El valor K de columna significa una profundidad de entrelazado de tiempo, que puede ser 1, 2, 4, 8 ó 16... La señalización de K para cada segmento de datos se puede realizar dentro de la señalización de L1. El entrelazado de frecuencia 403 como se muestra en la Fig. 9 se puede realizar previamente al entrelazador de tiempo 308 como se muestra en la Fig. 4.

La Fig. 67 muestra un rendimiento de entrelazado del entrelazador de tiempo como se muestra en la Fig. 66. Se supone que un valor de columna es 2, un valor de fila es 8, un ancho del segmento de datos son 12 células de datos y que no hay pilotos continuos en el segmento de datos. La figura superior en la Fig. 67 es una estructura de símbolos OFDM cuando no se realiza el entrelazado de tiempo y la figura inferior es la estructura de símbolos OFDM cuando se realiza un entrelazado de tiempo. Las células negras representan pilotos dispersos y las células no negras representan células de datos. Las mismas clases de células de datos representan un símbolo OFDM. En la Fig. 100, las células de datos que corresponden a un único símbolo OFDM se entrelazan en dos símbolos. Se usa una memoria de entrelazado que corresponde a ocho símbolos OFDM pero la profundidad de entrelazado corresponde sólo a dos símbolos, por ello, no se tiene una profundidad de entrelazado completa.

La Fig. 68 se sugiere para conseguir una profundidad de entrelazado completa. En la Fig. 68, las células negras representan pilotos dispersos y las células no negras representan células de datos. El entrelazado de tiempo como se muestra en la Fig. 68 se puede implementar en una forma de entrelazado de bloques y puede entrelazar segmentos de datos. En la Fig. 68, un número de columna, K representa un ancho del segmento de datos, un número de fila, N representa la profundidad de entrelazado de tiempo y el valor, K puede tener valores aleatorios, es decir, K=1, 2, 3, ... El proceso de entrelazado incluye la escritura de células de datos en una forma de giro de columna y la lectura en una dirección de columnas, excluyendo las posiciones pilotos. Esto es, se puede decir que se realiza un entrelazado en una forma de giro fila-columna.

Además, en un transmisor, las células que se leen en una forma de giro de columna desde la memoria de entrelazado corresponden a un único símbolo OFDM y las posiciones pilotos de los símbolos OFDM se pueden mantener mientras se entrelazan las células.

También, en un receptor, las células que se leen en una forma de giro de columna desde la memoria del desentrelazador corresponden a un único símbolo OFDM y las posiciones piloto de los símbolos OFDM se puede mantener mientras se desentrelazan las células.

La Fig. 69 muestra un rendimiento del entrelazado de tiempo de la Fig. 68. Por comparación con la Fig. 66, se supone que un número de filas es 8, un ancho del segmento de datos es 12 células de datos y que no hay pilotos continuos en el segmento de datos. En la Fig. 69, las células de datos que corresponden a un único símbolo OFDM se entrelazan en ocho símbolos OFDM. Como se muestra en la Fig. 102, se usa una memoria de entrelazado que corresponde a ocho símbolos OFDM y la profundidad de entrelazado resultante corresponde a ocho símbolos OFDM, por ello, se tiene una profundidad de entrelazado completa.

El entrelazador de tiempo como se muestra en la Fig. 68 puede ser ventajoso en que se puede tener una profundidad de entrelazado completa usando memoria idéntica; la profundidad de entrelazado puede ser flexible, en oposición a la Fig. 66; en consecuencia, una longitud de la trama de transmisión puede ser también flexible, es decir, las filas no necesitan ser múltiplos de cuatro. Adicionalmente, el entrelazado de tiempos usado para el segmento de datos, puede ser idéntico al procedimiento de entrelazado usado para el preámbulo y también puede estar en común con un sistema de transmisión digital que use OFDM general. Específicamente, el entrelazador de tiempos 308 como se muestra en la Fig. 4 se puede usar antes de que se use el entrelazador de frecuencia 403 como se muestra

en la Fig. 9. En relación a la complejidad del receptor, puede no requerirse memoria adicional distinta de la lógica de control de direcciones adicionales que puede requerir una complejidad muy pequeña.

La Fig. 70 muestra un desentrelazador de símbolos correspondiente (r308) en un receptor. Puede realizar el desentrelazado después de recibir la salida del Eliminador de la Cabecera de Trama r401. En el proceso de desentrelazado, comparado con la Fig. 66, los procesos de escritura y lectura de entrelazado de bloques se invierten. Mediante el uso de la información de la posición de piloto, el desentrelazador de tiempos puede realizar un desentrelazado virtual mediante una no escritura o no lectura desde la posición del piloto en la memoria del entrelazador y mediante la escritura a o la lectura desde una posición de célula de datos en la memoria del entrelazador. La información de desentrelazado se puede extraer hacia el extractor del ModCod r307.

La Fig. 71 muestra otro ejemplo del entrelazado de tiempos. Se puede realizar la escritura en la dirección diagonal y la lectura fila a fila. Como en la Fig. 68, el entrelazado se realiza teniendo en cuenta las posiciones de pilotos. La lectura y la escritura no se realizan para posiciones de pilotos sino que se accede a la memoria de entrelazado mediante la consideración solamente de las posiciones de las células de datos.

La Fig. 72 muestra un resultado de entrelazado usando el procedimiento mostrado en la Fig. 71. Cuando se compara con la Fig. 69, las células con los mismos patrones se dispersan no solamente en el dominio del tiempo, sino también en el dominio de la frecuencia. En otras palabras, se puede tener una profundidad de entrelazado completa tanto en el dominio del tiempo como en el de la frecuencia.

La Fig. 75 muestra un desentrelazador de símbolos r308 de un receptor correspondiente. La salida del Eliminador de la Cabecera de Trama r401 se puede desentrelazar. Cuando se compara con la Fig. 66, el desentrelazado ha conmutado el orden de lectura y escritura. El desentrelazador de tiempos puede usar la información de la posición piloto para realizar un desentrelazado virtual de modo que no se realice ninguna lectura o escritura en las posiciones piloto sino que la lectura o escritura se puede realizar solamente en las posiciones de las células de datos. El desentrelazado de datos se puede extraer hacia el Extractor de ModCod r307.

La Fig. 73 muestra un ejemplo del procedimiento de direccionado de la Fig. 72. NT significa la profundidad de entrelazado de tiempos y ND significa el ancho del segmento de datos. Se supone que el valor de fila, N es 8, un ancho del segmento de datos es de 12 células de datos y no hay pilotos continuos en el segmento de datos. La Fig. 73 representa un procedimiento de generación de direcciones para los datos de escritura en una memoria de entrelazado de tiempos, cuando un transmisor realiza entrelazado de tiempos. El direccionado comienza desde una primera dirección con una dirección de fila (RA) = 0 y una dirección de columna (CA) = 0. En cada situación de direccionado, se incrementan la RA y CA. Para RA, se puede realizar una operación de módulo con los símbolos OFDM usados en el entrelazado de tiempos. Para CA, se puede realizar una operación de módulo con un número de portadoras que corresponde a un ancho de segmento de datos. RA se incrementa en uno cuando las portadoras que corresponden al segmento de datos escriben en una memoria. La escritura en una memoria se puede realizar solamente cuando la localización de la dirección actual no es una localización del piloto. Si la localización de la dirección actual es una localización del piloto, sólo se puede incrementar el valor de la dirección.

En la Fig. 73, un número de columna, K representa un ancho del segmento de datos, un número de fila, N representa la profundidad de entrelazado de tiempos y el valor, K puede tener valores aleatorios, es decir, K=1, 2, 3, ... El proceso de entrelazado incluye la escritura de células de datos en una forma de giro de columna y la lectura en una dirección de columnas, excluyendo las posiciones pilotos. En otras palabras, la memoria de entrelazado virtual puede incluir posiciones piloto pero las posiciones piloto se pueden excluir en el entrelazado real.

La Fig. 76 muestra el desentrelazado, un proceso inverso del entrelazado de tiempos como se muestra en la Fig. 71. La escritura fila a fila y la lectura en la dirección diagonal puede restaurar las células en sus secuencias originales.

El procedimiento de direccionado usado en un transmisor se puede usar en un receptor. El receptor puede escribir los datos recibidos en una memoria del desentrelazador de tiempos fila a fila y puede leer los datos escritos usando los valores de dirección generados y la información de localización del piloto que se puede generar en una forma similar a la de un transmisor. Como una forma alternativa, los valores de dirección generados y la información del piloto que se usaron para la escritura se podrán usar para la lectura fila a fila.

Estos procedimientos se pueden aplicar en un preámbulo que transmita la L1. Debido a que cada símbolo OFDM que comprende el preámbulo puede tener los pilotos en dos localizaciones idénticas, se pueden realizar tanto un entrelazado con referencia a los valores de dirección teniendo en cuenta las localizaciones piloto, como un entrelazado con referencia a los valores de dirección sin tener en cuenta las localizaciones piloto. Para el caso de referirse a los valores de dirección sin tener en cuenta las localizaciones piloto, el transmisor almacena datos en una memoria del entrelazador de tiempos cada vez. En tal caso, un tamaño de la memoria requerido para realizar preámbulos de entrelazado/desentrelazado en un receptor o un transmisor son idénticas a un número de células de contenido útil que existen en los símbolos OFDM usados para entrelazado de tiempos.

La Fig. 74 es otro ejemplo de entrelazado de tiempos de L1. En este ejemplo, el entrelazado de tiempos puede colocar las portadoras en todos los símbolos OFDM mientras que las portadoras estarían todas localizadas en un único símbolo OFDM si no se realizara entrelazado de tiempos. Por ejemplo, para un dato localizado en un primer

símbolo OFDM, la primera portadora del primer símbolo OFDM se localizará en su localización original. La segunda portadora del primer símbolo OFDM se localizará en un segundo índice de portadora del segundo símbolo OFDM. En otras palabras, la portadora de datos i -ésima que se localiza en el n -ésimo símbolo OFDM se localizará en un índice de portadora i -ésimo del símbolo OFDM en el lugar $(i+n) \bmod N$, en el que $i = 0, 1, 2, \dots$ [número de portadora]1, $n = 0, 1, 2, \dots N-1$ y N es un número de símbolos OFDM usados en el entrelazado de tiempos de L1. En este procedimiento de entrelazado de tiempos de L1, se puede decir que se realiza un entrelazado para todos los símbolos OFDM en una forma de giro como se muestra en la Fig. 107. Incluso aunque las posiciones piloto no se ilustran en la Fig. 107, como se ha mencionado anteriormente, se puede aplicar el entrelazado a todos los símbolos OFDM incluyendo los símbolos piloto. Esto es, se puede decir que el entrelazado se puede realizar para todos los símbolos OFDM sin consideración a las posiciones piloto o independientemente de si los símbolos OFDM son símbolos piloto o no.

Si un tamaño de un bloque LDPC usado en L1 es más pequeño que el tamaño de un símbolo OFDM simple, las portadoras restantes pueden tener copias de partes del bloque LDPC o estar rellenas con ceros. En este punto, se puede realizar un mismo entrelazado de tiempos como el anterior. De modo similar, en la Fig. 74, un receptor puede realizar un desentrelazado mediante el almacenamiento de todos los bloques usados en el entrelazado de tiempos de L1 en una memoria y mediante la lectura de los bloques en el orden en que se han entrelazado, es decir, en el orden de los números escritos en los bloques mostrados en la Fig. 74.

Cuando se usa un entrelazado de bloques como se muestra en la Fig. 73, se usan dos memorias intermedias. Específicamente, mientras una memoria intermedia almacena símbolos de entrada, previamente los símbolos de entrada se pueden leer desde la otra memoria intermedia. Una vez que estos procesos se han realizado para un bloque de entrelazado de símbolos, se puede realizar el desentrelazado mediante la conmutación del orden de lectura y escritura, para evitar conflictos de acceso a memoria. Este estilo "ping-pong" de desentrelazado puede tener una lógica de generación de direcciones simple. Sin embargo, se puede incrementar la complejidad del hardware cuando se usan dos memorias intermedias de entrelazado de símbolos.

La Fig. 77 muestra un ejemplo de un desentrelazador de símbolos r308 o r308-1 como se muestra en la Fig. 31. Esta realización propuesta de la invención puede usar solamente una única memoria intermedia para realizar el desentrelazado. Una vez se genera un valor de dirección por la lógica de generación de direcciones, el valor de la dirección se puede sacar desde la memoria intermedia y se puede realizar una operación de colocación mediante el almacenamiento del símbolo que se introduce dentro de la misma dirección. Mediante estos procesos, se puede evitar un conflicto de acceso a memoria mientras se lee y escribe. Además, se puede realizar el desentrelazado de símbolos usando solamente una única memoria intermedia. Se pueden definir los parámetros para explicar esta regla de generación de direcciones. Como se muestra en la Fig. 73, un número de filas de una memoria de desentrelazado se puede definir como la profundidad de entrelazado de tiempos, D y un número de columnas de la memoria de desentrelazado se puede definir como el ancho del segmento de datos, W . A continuación el generador de direcciones puede generar las siguientes direcciones.

i -ésima muestra en el bloque j -ésimo, incluyendo el piloto

$i = 0, 1, 2, \dots N-1;$

$N = D * W;$

$C_i, j = i \bmod W;$

$T_w = ((C_i, j \bmod D) * j) \bmod D;$

$R_i, j = ((i \div W) + T_w) \bmod D;$

$L_i, j(1) = R_i, j * W + C_i, j;$

o

$L_i, j(2) = C_i, j * D + R_i, j-;$

Las direcciones incluyen las posiciones piloto, por ello, se supone que los símbolos de entrada incluyen las posiciones piloto. Si se necesita procesar símbolos piloto que incluyan solamente símbolos de datos, se puede requerir una lógica de control adicional que omita las direcciones correspondientes. En este punto, i representa un índice del símbolo de entrada, j representa un índice del bloque de entrelazado de entradas y $N=D*W$ representa una longitud del bloque de entrelazado. La operación mod representa la operación módulo que proporciona el resultado del resto tras la división. La operación div representa la operación de división que proporciona el resultado del cociente tras la división. R_i, j y C_i, j representan direcciones de fila y direcciones de columna de la entrada del símbolo i -ésimo del bloque de entrelazado j -ésimo, respectivamente. T_w representa el valor de giro de la columna

para direcciones en las que se sitúan los símbolos. En otras palabras, cada columna se puede considerar como una memoria intermedia en la que se realiza un giro independiente de acuerdo con los valores de T_w . L_i, j representa la dirección cuando se implementa una memoria intermedia simple en una memoria secuencial de una dimensión, no en dos dimensiones. L_i, j puede tener valores desde 0 a $(N-1)$. Son posibles dos procedimientos diferentes. Se usa L_i, j (1) cuando la matriz de la memoria se conecta fila a fila y se usa L_i, j (2) cuando la matriz de la memoria se conecta columna a columna.

La Fig. 78 muestra un ejemplo de direcciones de fila y de columna para desentrelazado de tiempos cuando D es 8 y W es 12. J comienza desde $j=0$ y, para cada valor de j , una primera fila puede representar la dirección de fila y una segunda fila puede representar la dirección de columna. La Fig. 78 muestra sólo direcciones de los primeros 24 símbolos. Cada índice de columna puede ser idéntico al índice del símbolo de entrada i .

La Fig. 80 muestra un ejemplo de un transmisor OFDM que usa un segmento de datos. Como se muestra en la Fig. 80, el transmisor puede comprender un recorrido del PLP de datos, un recorrido de señalización de L_1 , un constructor de tramas y una parte de modulación OFDM. El recorrido del PLP de datos se indica por bloques con líneas horizontales y líneas verticales. El recorrido de la señalización de L_1 se indica por bloques con líneas inclinadas. Los módulos de procesamiento de entrada 701-0, 701-N, 701-K y 701-M pueden comprender bloques y secuencias del módulo de interfaz de entrada 202-1, el sincronizador de la corriente de entrada 203-1, el compensador de retardo 204-1, el módulo de borrado de paquetes nulos 205-1, el codificador de CRC 206-1, el módulo de inserción de la cabecera de BB 207-1 y el codificador de BB 209 realizado para cada PLP como se muestra en la Fig. 2. Los módulos de FEC 702-0, 702-N, 702-K y 702-M pueden comprender bloques y secuencias del módulo de codificación exterior 301 y del módulo de codificación interior 303 como se muestra en la Fig. 4. Un módulo de FEC 702-L1 usado en el recorrido de L_1 puede comprender bloques y secuencias del módulo de codificación exterior 301-1 y del módulo de codificación interior recortado/perforado 303-1 como se muestra en la Fig. 4. El módulo de la señal de L_1 700-L1 puede generar información de L_1 requerida para comprender una trama.

Los módulos de entrelazado de bits 703-0, 703-N, 703-K y 703-M pueden comprender bloques y secuencias del entrelazador interior 304 y del demultiplexor de bits 305 como se muestra en la Fig. 37. El módulo de desentrelazado de bits 703-L1 usado en el recorrido de L_1 puede comprender bloques y secuencias del módulo de entrelazado interior 304-1 y del demultiplexor de bits 305-1 como se muestra en la Fig. 4. Los módulos del mapeador de símbolos 704-0, 704-N, 704-K y 704-M pueden realizar funciones idénticas a las funciones del mapeador de símbolos 306 mostrado en la Fig. 4. El módulo mapeador de símbolos 704-L1 usado en el recorrido de L_1 puede realizar funciones idénticas a las funciones del mapeador de símbolos 306-1 mostrado en la Fig. 4. Los módulos de cabecera del FEC 705-0, 705-N, 705-K y 705-M pueden realizar funciones idénticas a las funciones del módulo de inserción de la Cabecera ModCod 307 mostrado en la Fig. 4. El módulo de cabecera FEC 705-L1 para el recorrido de L_1 puede realizar funciones idénticas a las funciones del módulo de inserción de Cabecera ModCod 307-1 mostrado en la Fig. 4.

Los módulos del mapeador de segmentos 706-0 y 706-K pueden planificar los bloques FEC en las correspondientes segmentos de datos y pueden transmitir los bloques FEC planificados, en los que los bloques FEC corresponden a los PLP que se asignan a cada segmento de datos. El mapeador del preámbulo 707-L1 puede planificar los bloques FEC de señalización de L_1 a los preámbulos. Los bloques FEC de señalización de L_1 se transmiten en preámbulos. Los módulos del entrelazado de tiempos 708-0 y 708-K pueden realizar funciones idénticas a las funciones del entrelazador de símbolos 308 mostrado en la Fig. 4 que puede entrelazar segmentos de datos. El entrelazador de tiempos 708-L1 usado en el recorrido de L_1 puede realizar funciones idénticas a las funciones del entrelazador de símbolos 308-1 mostrado en la Fig. 4.

Alternativamente, el entrelazador de tiempos 708-L1 usado en el recorrido de L_1 puede realizar funciones idénticas a las del entrelazador de símbolos 308-1 mostrado en la Fig. 3, pero solamente sobre los símbolos del preámbulo.

Los bloques del entrelazador de frecuencia 709-0 y 709-K pueden realizar entrelazado de frecuencia sobre los segmentos de datos. El entrelazador de frecuencia 709-L1 usado en el recorrido de L_1 puede realizar entrelazado de frecuencia de acuerdo con el ancho de banda del preámbulo.

El módulo de generación de pilotos 710 puede generar pilotos que sean adecuados para pilotos continuos (CP), pilotos dispersos (SP), bordes de segmentos de datos y preámbulos. Una trama se puede construir a partir de la planificación del segmento de datos, del preámbulo y del piloto en el módulo 711. El módulo de IFFT 712 y el módulo de inserción de GI 713 pueden realizar funciones idénticas a las funciones del módulo de IFFT 501 y del módulo de inserción de GI 503 bloques mostrados en la Fig. 18, respectivamente. Finalmente, el DAC 714 puede convertir señales digitales en señales analógicas y se pueden transmitir las señales convertidas.

La Fig. 81 muestra un ejemplo de un receptor OFDM que usa segmentos de datos. En la Fig. 81, el sintonizador r700 puede realizar las funciones del sintonizador/AGC r603 y las funciones del convertidor reductor r602 mostrado en la Fig. 61. El ADC r701 puede convertir las señales analógicas recibidas en señales digitales. El sintonizador de tiempos/frecuencia r702 puede realizar funciones idénticas a las funciones del sintonizador de tiempos/frecuencia r505 mostrado en la Fig. 62. El detector de tramas r703 puede realizar funciones idénticas a las funciones del detector de tramas r506 mostrado en la Fig. 62.

En este punto, después de que se realice la sincronización de tiempos/frecuencia, se puede mejorar la sincronización mediante el uso de un preámbulo en cada trama que se envíe desde el detector de tramas r703 durante el proceso de seguimiento.

5 El eliminador de GI r704 y el módulo de FFT r705 pueden realizar funciones idénticas a las funciones del eliminador de GI r503 y del módulo de FFT r502 mostrados en la Fig. 62, respectivamente.

10 El estimador del canal r706 y el EQ del canal r707 pueden realizar una parte de estimación del canal y una parte de cualitativa del canal del Estimador/Ecualizador del canal r501 como se muestra en la Fig. 62. El analizador de tramas r708 puede producir la salida de un segmento de datos y un preámbulo en los que se transmiten los servicios seleccionados por un usuario. Los bloques indicados por las líneas inclinadas procesan un preámbulo. Los bloques indicados por líneas horizontales que pueden incluir los PLP comunes, procesan segmentos de datos. El desentrelazador de frecuencia r709-L1 usado en el recorrido de L1 puede realizar el desentrelazado de la frecuencia dentro del ancho de banda del preámbulo. El desentrelazador de frecuencia r709 usado en el recorrido del segmento de datos puede realizar un desentrelazado de frecuencias dentro de un segmento de datos. El módulo de decodificación de la cabecera FEC r7012-L1, el desentrelazador de tiempos r710-L1 y el desmapeador de símbolos r713-L1 usados en el recorrido del L1 pueden realizar funciones idénticas a las funciones del extractor ModCod r307-1, el desentrelazador de símbolos r308-1 y el desmapeador de símbolos r306-1 mostrados en la Fig. 31.

15 El desentrelazador de bits r714-L1 puede comprender módulos y secuencias del demultiplexor de bits r305-1 y el desentrelazador interior r304-1 como se muestra en la Fig. 31. El módulo de decodificación del FEC r715-L1 puede comprender módulos y secuencias del módulo de codificación interior recortado/perforado r303-1 y del módulo de decodificación exterior r301-1 mostrados en la Fig. 31. En este punto, la salida del recorrido de L1 puede ser la información de señalización de L1 y se puede enviar a un controlador del sistema para restaurar los datos del PLP que se transmiten en segmentos de datos.

20 El desentrelazador de tiempos r710 usado en el recorrido del segmento de datos puede realizar funciones idénticas a las funciones del desentrelazador de símbolos r308 mostrado en la Fig. 31. El analizador del segmento de datos r711 puede producir la salida de los PLP preseleccionados por usuario de entre los segmentos de datos y, si es necesario, los PLP comunes asociadas con los PLP preseleccionados por el usuario. Los módulos de decodificación de la cabecera FEC r712-C y r712-K pueden realizar funciones idénticas a las funciones del extractor ModCod r307 mostrado en la Fig. 31. Los desmapeadores de Símbolos r713-C y r713-K pueden realizar funciones idénticas a las funciones del desmapeador de símbolos r306 mostrado en la Fig. 31.

25 Los desentrelazadores de bits r714-C y r714-K pueden comprender bloques y secuencias del demultiplexor de bits r305 y del desentrelazador interior r304 como se muestran en la Fig. 31. Los módulos de decodificación del FEC r715-C y r715-K pueden comprender bloques y secuencias del módulo de decodificación interior r303 y del módulo de decodificación exterior r301 como se muestran en la Fig. 31. Finalmente, los procesadores exteriores r716-C y r716-K pueden comprender bloques y secuencias del descodificador de BB r209, del eliminador de la cabecera de BB r207-1, del decodificador de CRC r206-1, del módulo de inserción de paquetes nulos r205-1, del recuperador de retardos r204-1, del módulo de recuperación del reloj de salida r203-1 y del módulo de interfaz de salida r202-1 que se realizan para cada PLP en la Fig. 2. Si se usa un PLP común, el PLP común y los datos del PLP asociados con el PLP común se pueden transmitir a un recombinador de TS y se pueden transformar en un PLP seleccionada por el usuario.

30 Se debería indicar a partir de la Fig. 81, que en un receptor, los bloques del recorrido de L1 no se secuencian simétricamente hacia un transmisor en oposición al recorrido de datos en el que los bloques se posicionan simétricamente o en la secuencia opuesta de un transmisor. En otras palabras, para el recorrido de datos, se sitúan el desentrelazador de Frecuencia r709, el desentrelazador de tiempos r710, el analizador del segmento de datos r711 y los módulos de decodificación de la cabecera FEC r712-C y r712-K. Sin embargo, para el recorrido de L1, se sitúan el desentrelazador de Frecuencia r709, el módulo de decodificación de la cabecera FEC r712-L1 y el desentrelazador de tiempos r710-L1.

35 La Fig. 79 muestra un ejemplo de un entrelazador de bloques general en un dominio de símbolos de datos en el que no se usan pilotos. Como se puede ver en la figura de la izquierda, la memoria de entrelazado se puede rellenar sin pilotos negros. Para formar una memoria rectangular, se pueden usar células de relleno si es necesario. En la figura de la izquierda, las células de relleno se indican como células con líneas inclinadas. En el ejemplo, debido a que un piloto continuo puede solaparse con una clase de patrón de pilotos dispersos, se requieren un total de tres células de relleno durante la duración de cuatro símbolos OFDM. Finalmente, en la mitad de la figura, se muestra el contenido de la memoria entrelazada.

40 Como en la figura izquierda de la Fig. 79, se puede realizar tanto la escritura fila a fila y la realización de un giro de columna como la escritura en una forma de giro desde el principio. La salida del entrelazador no puede comprender la lectura fila a fila desde la memoria. Los datos de salida que se han leído se pueden colocar como se muestra en la figura derecha cuando se considera la transmisión OFDM. En este momento, por simplicidad, se puede ignorar el entrelazado de frecuencia. Como se ve en la figura, la diversidad de frecuencia no es tan alta como la de la Fig. 73, pero se mantiene en un nivel similar. Sobre todo, puede ser ventajoso que la memoria requerida para realizar el

entrelazado y el desentrelazado se puede optimizar. En el ejemplo, el tamaño de la memoria se puede reducir desde $W * D$ a $(W-1) * D$. Según se hace más grande el ancho del segmento de datos, se puede reducir adicionalmente el tamaño de la memoria.

Para las entradas del desentrelazador de tiempos, un receptor debería restaurar el contenido de la memoria intermedia en una forma de la mitad de la figura cuando se consideran las células de relleno. Básicamente, los símbolos OFDM se puede leer símbolo a símbolo y se pueden guardar fila a fila. Se puede realizar una reversión del giro correspondiente al giro de la columna. La salida del desentrelazador se puede extraer en una forma de lectura fila a fila desde la memoria de la figura izquierda. En esta forma, cuando se compara con el procedimiento mostrado en la Fig. 73, se puede minimizar la sobrecarga del piloto y en consecuencia se puede minimizar la memoria de entrelazador/desentrelazador.

La Fig. 82 muestra un ejemplo de un entrelazador de tiempos 708-L1 para el recorrido de L1 de la Fig. 80. Como se muestra en la Fig. 82, el entrelazador de tiempos para el preámbulo en el que se transmite L1, puede incluir las células de datos de L1 entrelazadas, excluyendo pilotos que se transmiten usualmente en el preámbulo. El procedimiento de entrelazado puede incluir la escritura de datos de entrada en una dirección diagonal (líneas continuas) y la lectura de los datos fila a fila (líneas de puntos), usando procedimientos idénticos a los que se muestran con referencia a la Fig. 73.

La Fig. 82 muestra un ejemplo de un desentrelazador de tiempos r712-L1 en el recorrido de L1 como se muestra en la Fig. 81. Como se muestra en la Fig. 82, para el preámbulo en el que se transmite L1, se puede realizar el desentrelazado de las células de datos de L1, excluyendo los pilotos que se transmiten regularmente en el preámbulo. El procedimiento de desentrelazado puede ser idéntico al procedimiento mostrado en la Fig. 76 en el que los datos de entrada se escriben fila a fila (líneas continuas) y se leen en una dirección diagonal (líneas de puntos). Los datos de entrada no incluyen ningún piloto, en consecuencia, los datos de salida tienen las células de datos de L1 que no incluyen tampoco pilotos. Cuando un receptor usa una memoria intermedia única en un desentrelazador de tiempos para el preámbulo, se puede usar la estructura del generador de direcciones que tiene una memoria de desentrelazado como se muestra en la Fig. 77.

Se puede realizar el desentrelazado r412-L1 usando operaciones de direcciones como sigue:

i-ésima muestra en el bloque j-ésimo, incluyendo el piloto

$i = 0, 1, 2, \dots N-1;$

$N = D * W;$

$Ci, j = i \bmod W;$

$Tw = ((Ci, j \bmod D) * j) \bmod D;$

$Ri, j = ((i \div W) + Tw) \bmod D;$

$Li, j(1) = Ri, j * W + Ci, j;$

O

$Li, j(2) = Ci, j * D + Ri, j-;$

En las operaciones anteriores, una longitud de una fila, W es una longitud de una fila en una memoria de entrelazado como se muestra en la Fig. 82. La longitud de la columna, D es una profundidad de entrelazado de tiempos del preámbulo, es un número de los símbolos OFDM que se requieren para la transmisión de preámbulos.

La Fig. 83 muestra un ejemplo de la formación de símbolos OFDM mediante la planificación de pilotos y preámbulos de entrada a partir del constructor de Tramas 711 como se muestra en la Fig. 80. Las células vacías forman una cabecera de L1 que es una señal de salida de la cabecera FEC 705-L1 en el recorrido de L1, como se muestra en la Fig. 80. Las células en gris representan los pilotos continuos para el preámbulo que se generan por el módulo de generación de pilotos 710 como se muestra en la Fig. 80. Las células con patrones representan las células de señalización de L1 que son una señal de salida del mapeador de preámbulos 707-L1 como se muestra en la Fig. 80. La figura de la izquierda representa los símbolos OFDM cuando el entrelazador de tiempos está desconectado y la figura de la derecha representa los símbolos OFDM cuando el entrelazador de tiempos está conectado. La cabecera de L1 se puede excluir del entrelazado de tiempos debido a que la cabecera de L1 transmite una longitud del campo de señalización de L1 en una información del marcador de desconexión/conexión del entrelazado de tiempos. Esto se debe a que la cabecera de L1 se añade antes del entrelazado de tiempos. Como se ha mencionado anteriormente, el entrelazado de tiempos se realiza excluyendo las células piloto. El resto de las células de datos de

L1 se pueden entrelazar cómo se muestra en la Fig. 82, a continuación se pueden asignar a las portadoras de OFDM.

La Fig. 84 muestra un ejemplo de unos Entrelazadores de Tiempos 708-0 – 708-K que pueden entrelazar símbolos de datos que se envían desde los Mapeadores de los Segmentos de Datos 706-0 – 706-K en el recorrido de datos de un transmisor OFDM que usa segmentos de datos mostrado en la Fig. 80. El entrelazador de tiempos se puede 45 realizar para cada segmento de datos. Los símbolos entrelazados en el tiempo se puede sacar hacia los Entrelazadores de Frecuencia 709-0 – 709-K.

La Fig. 84 muestra también un ejemplo de un entrelazador de tiempos simple que usa una memoria intermedia única. La Fig. 84a muestra una estructura de símbolos OFDM antes del entrelazado de Tiempos. Los bloques con los mismos patrones representan la misma clase de símbolos OFDM. Las Figs. 84b y 84c muestran una estructura de símbolos OFDM después del entrelazado de Tiempos. El procedimiento de entrelazado de Tiempos se puede dividir en el Tipo 1 y en el Tipo 2. Cada tipo se puede realizar alternativamente para los símbolos pares y los símbolos impares. Un receptor puede realizar el desentrelazado en consecuencia. Una de las razones de usar 15 alternativamente el tipo 1 y el tipo 2 es reducir la memoria requerida en un receptor mediante el uso de una memoria intermedia simple durante el desentrelazado de tiempos.

La Fig. 84b muestra un entrelazado de tiempos usando el tipo de entrelazado 1. Los símbolos de entrada se pueden escribir en una dirección diagonal hacia abajo y se pueden leer en la dirección de filas. La Fig. 84c muestra un entrelazado de tiempos usando el tipo de entrelazado 2. Los símbolos de entrada se pueden escribir en una 20 dirección diagonal hacia arriba y se puede leer en una dirección de filas. La diferencia entre el tipo 1 y el tipo 2 es si la dirección de escritura de los símbolos de entrada es hacia arriba o hacia abajo. Los dos procedimientos son diferentes en la forma de la escritura de símbolos, sin embargo los dos procedimientos son idénticos en términos de mostrar una profundidad de entrelazado de tiempos completa y una diversidad de frecuencia completa. Sin embargo, usando estos procedimientos puede producirse un problema durante la sincronización en un receptor debido al uso de los dos esquemas de entrelazado.

Puede haber dos soluciones posibles. La primera solución puede ser 1 bit de señalización de un tipo de entrelazado en un primer bloque de entrelazado que llegue el primero después de cada preámbulo, a través del preámbulo de señalización de L1. Este procedimiento realiza el entrelazado correcto por medio de la señalización. La segunda 25 solución puede ser la formación de una trama para que tenga una longitud de un número par de bloques de entrelazado. Usando este procedimiento, el primer bloque de entrelazado de cada trama puede tener un tipo idéntico, por ello, se puede resolver el problema de la sincronización del bloque de entrelazado. Por ejemplo, el problema de la sincronización se puede resolver mediante la aplicación del tipo 1 de entrelazado a un primer bloque de entrelazado y la aplicación secuencialmente a los siguientes bloques de entrelazado dentro de cada trama, finalizando a continuación un último bloque de entrelazado de cada trama con el entrelazado de tipo 2. Este 30 procedimiento requiere que se componga una trama con dos bloques de entrelazado pero puede ser ventajoso en que no se requiere ninguna señalización adicional como en el primer procedimiento.

La Fig. 89 muestra una estructura de un desentrelazador de Tiempos r707 de un receptor mostrado en la Fig. 81. El Desentrelazado de Tiempos se puede realizar sobre las salidas del desentrelazador de Frecuencia r709. El desentrelazador de tiempos de la Fig. 89 representa un esquema de desentrelazado que es un proceso inverso de un entrelazado de tiempos mostrado en la Fig. 84. El desentrelazado, comparado con la Fig. 84, tendrán una 40 manera opuesta en la lectura y la escritura. En otras palabras, el tipo 1 de desentrelazado puede escribir los símbolos de entrada en una dirección de filas y puede leer los símbolos escritos en una dirección diagonal hacia abajo. El tipo 2 de desentrelazado puede escribir los símbolos de entrada en una dirección diagonal hacia abajo y puede leer los símbolos escritos en una dirección de filas. Estos procedimientos pueden permitir la escritura de los símbolos recibidos en los que los símbolos se leen previamente mediante la realización de una dirección de 45 símbolos de escritura del desentrelazador de tipo 2 idéntica a la dirección de lectura de los símbolos del desentrelazador de tipo 1. Por ello, un receptor puede realizar el desentrelazado usando una memoria intermedia única. Además, se puede realizar una implementación simple debido a que los procedimientos de desentrelazado del tipo 1 y del tipo 2 se realizan tanto con la escritura y la lectura de símbolos en una dirección diagonal como en una dirección de filas.

Sin embargo, usando estos procedimientos puede producirse un problema en la sincronización en un receptor debido al uso de los dos esquemas de entrelazado. Por ejemplo, los símbolos entrelazados con el entrelazado de tipo 1 en una forma de tipo 2 pueden producir un deterioro en el rendimiento. Puede haber dos soluciones posibles. La primera solución puede ser la determinación de un tipo de un bloque de entrelazado que llega después del preámbulo, usando 1 bit de un tipo de entrelazado de una parte de la señalización de L1 transmitida. La segunda 50 solución puede ser la realización de un desentrelazado usando un tipo de acuerdo con un primer bloque de entrelazado dentro de una trama, si el número de bloques de entrelazado dentro de una trama es un número par. El símbolo de ese entrelazado se puede extraer dentro del Analizador de los Segmentos de Datos r711.

La Fig. 85 muestra una lógica de generación de direcciones que es idéntica a la lógica de generación de direcciones de una memoria intermedia única, cuando un entrelazador de bloques usa dos memorias intermedias como en la Fig. 73. La lógica de generación de direcciones puede realizar funciones idénticas a las funciones mostradas en la 60

Fig. 73. Mediante la definición de una profundidad de entrelazado de tiempos D como un número de filas de una memoria de desentrelazado y la definición de un ancho del segmento de datos W como un número de columna, las direcciones mostradas en la Fig. 85 se puede generar mediante el generador de direcciones. Las direcciones pueden incluir las posiciones piloto. Para los símbolos de entrada de entrelazado de tiempos que incluye solamente símbolos de datos, una lógica de control puede saltarse direcciones según se requiera. Las direcciones usadas en los preámbulos de entrelazado pueden no requerir posiciones piloto y se puede realizar el entrelazado usando los bloques de L1. El i representa un índice de un símbolo de entrada, $N=W*D$ representa una longitud del bloque de entrelazado. R_i y C_i representan una dirección de fila y una dirección de columna de un símbolo de entrada i -ésimo, respectivamente. T_w representa un valor de giro de columna o parámetro de giro a partir de una dirección en la que se localiza un símbolo. L_i representa las direcciones cuando se implementa una memoria dimensional que tenga una memoria intermedia única. Los valores de L_i pueden ser desde 0 a $(N-1)$. En esta memoria de una dimensión, al menos son posibles dos procedimientos. $L_i(1)$ se acopla a una matriz de memoria fila a fila y $L_i(2)$ se acopla a una matriz de memoria columna a columna. Un receptor puede usar la lógica de generación de direcciones en la lectura de símbolos durante un desentrelazado.

La Fig. 86 muestra otro ejemplo de un preámbulo. Para un caso en el que se usen un símbolo OFDM que tenga un tamaño de 4K-FFT en el ancho de banda de 7, 61 MHz y una sexta portadora dentro del símbolo OFDM y se usen las portadoras en ambos extremos como pilotos, un número de portadoras que se puede usar en la señalización de L1 se puede asumir que sea 2840. Cuando se adhieren múltiples canales, pueden existir anchos de banda de preámbulo múltiples. El número de portadoras puede cambiar dependiendo del tipo de los pilotos a ser usado, un tamaño de FFT, un número de canales adheridos y otros factores. Si se ha de asignar un tamaño de un L1_XFEC_FRAME que incluye la L1_header (H) a un único símbolo OFDM y el bloque de FEC de L1 (L1_FEC1) es más pequeño que un símbolo OFDM simple (5w-a-1), L1_XFEC_FRAME que incluye L1_header se puede repetir para rellenar una parte restante del símbolo OFDM simple (5w-a-2). Esto es similar a la estructura del preámbulo de la Fig. 60. Para que un receptor reciba un segmento de datos que se sitúe en un cierto ancho de banda de canales adheridos, una ventana del sintonizador se puede localizar en un cierto ancho de banda.

Si una ventana de sintonizador del receptor se localiza como el 5w-a-3 de la Fig. 86, puede tener lugar un resultado incorrecto durante la mezcla repetida de las L1_XFEC_FRAME. El caso 1 de la Fig. 86 puede ser un ejemplo así. Un receptor halla la L1_Header (H) para localizar una posición de comienzo de una L1_Header (H) dentro de una ventana del sintonizador, pero la L1_Header hallada puede ser una cabecera de una L1_XFEC_FRAME (5w-a-4) incompleta. La información de señalización de L1 puede que no se obtenga correctamente si se obtiene una longitud de la L1_XFEC_FRAME en base a esa L1_Header y se añade un resto de la parte (5w-a-5) a una posición de comienzo de esa L1_Header. Para impedir tal caso, un receptor puede necesitar operaciones adicionales para hallar una cabecera de una L1_XFEC_FRAME completa. La Fig. 87 muestra tales operaciones. En el ejemplo, para hallar una cabecera de L1_XFEC_FRAME completa, si existe una L1_XFEC_FRAME incompleta en un preámbulo, un receptor puede usar al menos dos L1-Headers para hallar la localización de comienzo de la L1_Header para la mezcla de la L1_XFEC_FRAME. Primero, un receptor puede hallar la L1_Header a partir de un símbolo OFDM de preámbulo (5w-b-1). A continuación usando una longitud de una L1_XFEC_FRAME dentro de la L1_Header hallada, el receptor puede comprobar si cada L1_XFEC_FRAME dentro de un símbolo OFDM actual es un bloque completo (5w-b-2). Si no lo es, el receptor pueda hallar otra L1_Header a partir del símbolo de preámbulo actual (5w-b-3). A partir de una distancia calculada entre una L1_Header hallada de nuevo y una L1_Header previa, se puede determinar si una cierta L1_XFEC_FRAME es un bloque completo (5w-b-4). A continuación, se puede usar una L1_Header de una L1_XFEC_FRAME completa como el punto de comienzo para la mezcla. Usando este punto de comienzo, se pueden mezclar la L1_XFEC_FRAME (5w-b-5). Usando estos procesos, se puede esperar en un receptor el caso 2 o la mezcla correcta mostrada en la Fig. 86. Estos procesos se pueden realizar en el Decodificador de la Cabecera FEC r712-L1 en el recorrido de la señal L1 de la Fig. 81.

La Fig. 88 es un ejemplo de una estructura de preámbulo que puede eliminar las operaciones adicionales anteriormente mencionadas en un receptor. En oposición a la estructura de preámbulo previa, cuando se rellena una parte restante del símbolo OFDM, solamente se pueden rellenar repetidamente las L1_FEC1 de una L1_XFEC_FRAME, excluyendo la L1_Header (H) (5w-c-2). En esta forma, cuando un receptor halla una posición de comienzo de una L1_Header (H) para mezclar la L1_XFEC_FRAME, se puede hallar la L1_Header sólo de las L1_XFEC_FRAME completas (5w-c-4), por ello, sin operaciones adicionales, se pueden mezclar las L1_XFEC_FRAME usando la L1_Header hallada. Por lo tanto se puede eliminar procesos tales como los (5w-b-2), (5w-b-3) y (5w-b-4) mostrados en la Fig. 87, en un receptor. Estos procesos y los procesos contrarios de los procesos se pueden realizar en el Decodificador de Cabecera del FEC 712-L1 en el recorrido de la señal de L1 de un receptor de la Fig. 81 y en la Cabecera FEC 705-L1 en el recorrido de la señal de L1 de un transmisor de la Fig. 80.

El desentrelazador de tiempos r712-L1 del recorrido de L1 de un receptor de la Fig. 81 puede desentrelazar celdas de bloques de L1 o células con patrones, excluyendo otras células tales como las cabeceras de preámbulo y células piloto. Las células del bloque L1 se representan por células con patrones como se muestra en la Fig. 83. La Fig. 90 muestra otro ejemplo de un transmisor OFDM que usa segmentos de datos. Este transmisor puede tener una estructura idéntica y puede realizar una función idéntica a la del transmisor de la Fig. 80, excepto los bloques añadidos y modificados. El mapeador del preámbulo 1007-L1 puede mapear los bloques de L1 y las cabeceras de bloque de L1 que salen de la cabecera FEC 705-L1 en símbolos de preámbulo usados en una trama de transmisión.

Específicamente, la cabecera del bloque de L1 se puede repetir para cada preámbulo y el bloque de L1 se puede dividir tanto como el número de preámbulos usados. El entrelazador de tiempos 1008-L1 puede entrelazar los bloques de L1 que se dividen en preámbulos. En este punto, la cabecera del bloque de L1 puede o bien ser incluida en el entrelazador o bien no incluida en el entrelazador. Si se incluye la cabecera del bloque de L1 o no puede no cambiar una estructura de señal de la cabecera del bloque de L1 pero cambia un orden de entrelazado y de transmisión de los bloques de L1. El repetidor de L1_XFEC 1015-L1 puede repetir los bloques L1_XFEC entrelazados en el tiempo dentro de un ancho de banda de preámbulo. En este punto, la cabecera del bloque de L1 puede o bien ser repetida dentro de un preámbulo o bien no repetida dentro de un preámbulo.

La Fig. 91 muestra otro ejemplo de un receptor OFDM que usa segmentos de datos. Este receptor tiene una estructura idéntica y puede realizar una función idéntica a la del receptor de la Fig. 81, excepto los bloques añadidos y modificados. El módulo de decodificación de la cabecera FEC r1012-L1 puede sincronizar las cabeceras de L1 dentro de un preámbulo. Si las cabeceras de L1 se repiten, las cabeceras de L1 pueden combinarse para tener una ganancia del SNR. A continuación, el módulo de decodificación de la cabecera FEC r712-L1 de la Fig. 81 puede realizar una decodificación del FEC. El proceso de sincronización puede dar una localización de una cabecera mediante la correlación de la palabra de sincronismo de la cabecera y los preámbulos. Para desplazamiento de frecuencia de múltiplos de un entero, se puede determinar un intervalo de correlación a partir de un direccionamiento circular.

El módulo de combinación de L1_XFEC r1017-L1 puede combinar los bloques L1_XFEC para obtener una ganancia de SNR, cuando se reciben bloques de L1 divididos dentro de un preámbulo. El desentrelazador de tiempos r1010-L1 puede desentrelazar los tiempos de los bloques de L1 dentro de un preámbulo. Dependiendo de si las cabeceras del bloque de L1 están entrelazadas en el tiempo en el transmisor o no, las cabeceras del bloque de L1 se pueden desentrelazar en consecuencia en un receptor. El orden de desentrelazado de los bloques de L1 se puede cambiar dependiendo de si las cabeceras del bloque de L1 están entrelazadas en el tiempo en el transmisor o no. Por ejemplo, cuando el entrelazado de tiempos está conectado como en la Fig. 83, puede cambiar una localización de la célula número 33 que es una célula del primer bloque de L1 dentro de un primer preámbulo. En otras palabras, cuando las cabeceras del bloque de L1 no se incluyen en un entrelazado, se recibirá la señal de entrelazado que tienen las localizaciones de las células como se muestra en la Fig. 83. Si las cabeceras del bloque de L1 se incluyen en un entrelazado, una localización de la célula número 33 necesita ser cambiada a las células desentrelazadas que se entrelazan diagonalmente, usando una primera célula de una primera cabecera del bloque de L1 dentro de un primer preámbulo como referencia. El fusionador de L1_FEC r1018-L1 puede cambiar los bloques de L1 que se dividen en muchos preámbulos dentro de un único bloque de L1 para la decodificación FEC.

Con 1 bit adicional, el campo PLP_type de los campos de señalización de L1 que se transmiten en un preámbulo puede tener los siguientes valores.

PLP_type = 00 (PLP común)

PLP_type = 01 (PLP de datos normales)

PLP_type = 10 (PLP de datos demultiplexados)

PLP_type = 11 (reservado)

Un PLP de datos normales representa un PLP de datos cuando se transmite un servicio simple en un segmento de datos simple. Un PLP de datos demultiplexados representa un PLP de datos cuando se demultiplexa un servicio único en múltiples segmentos de datos. Cuando un usuario cambia el servicio, si se almacena la señalización de L1 y la señalización de L2 en un receptor, se puede eliminar la espera a una información de señalización de L1 dentro de una trama siguiente. Por lo tanto, un receptor puede cambiar los servicios de modo eficiente y un usuario puede beneficiarse de un menor retardo durante un cambio de servicio. La Fig. 95 muestra la estructura de la señal del bloque de L1 que se transmite en un preámbulo, para el flujo del entrelazado de tiempos y el flujo del desentrelazado de tiempos. Como se puede ver en la Fig. 95, el entrelazado y el desentrelazado se pueden realizar no sobre un ancho de banda total del preámbulo, sino sobre un bloque de L1 dividido.

La Fig. 96 es un ejemplo de un campo de entrelazado de tiempos de L1 de los campos de señalización de L1, procesados por la cabecera de FEC 705-L1 en el recorrido de L1 mostrado en la Fig. 90. Como se muestra en la Fig. 96, se pueden usar un bit o dos bits para parámetro de entrelazado de tiempos. Si se usa un bit, el entrelazado no se realiza cuando el valor del bit es 0 y se puede realizar un entrelazado que tiene una profundidad de símbolos OFDM usados en los símbolos de preámbulo cuando el valor del bit es 1. Si se usan dos bits, se realiza un entrelazado con una profundidad de entrelazado de 0 o ningún entrelazado cuando el valor de los bits es 00 y se puede realizar un entrelazado que tenga la profundidad de los símbolos OFDM usados en los símbolos de preámbulo cuando el valor de los bits es 01. Se puede realizar un entrelazado que tenga la profundidad de cuatro símbolos OFDM cuando el valor de los bits es 10. Se puede realizar un entrelazado que tenga la profundidad de ocho símbolos OFDM cuando el valor de los bits es 11.

Un receptor, específicamente, un decodificador de cabecera FEC r1012-L1 en el recorrido de L1 mostrado en la Fig. 91 puede extraer los parámetros de Entrelazador de Tiempos (TI) mostrado en la Fig. 96. Usando los parámetros, el

desentrelazador de Tiempos r1010-L1 puede realizar el desentrelazado de acuerdo con la profundidad de entrelazado. Los parámetros que se transmiten en la cabecera de L1 son tamaño de la información de L1 (15 bits), parámetro de entrelazado de tiempos (máximo 2 bits) y CRC (máximo 2 bits). Si se usa el código Reed-Muller RM (16, 32) para codificación del campo de señalización de la cabecera de L1, debido que los bits que se pueden transmitir son 16 bits, no existe un número suficiente de bits. La Fig. 97 muestra un ejemplo del campo de señalización de L1 que se puede usar en tal caso.

La Fig. 97 muestra un procesamiento realizado en una cabecera FEC 705-L1 en el recorrido de L1 de la Fig. 90. En la Fig. 97a, L1 () en la columna de los campos de señalización representa el tamaño de L1 y TI () representa el tamaño para los parámetros de entrelazado de tiempo. Para el primer caso o cuando se transmiten el tamaño de L1 (15 bits) y de TI (1 bit), puede no ser necesario un relleno adicional y se puede tener un rendimiento de la codificación sustancial en la cabecera de L1, sin embargo, debido a que la información o bien no realiza un entrelazado de tiempos o no se transmite, para un bloque de L1 corto, no se puede obtener el efecto de entrelazado.

Para el segundo caso o cuando el tamaño de L1 se reduce a 1/8 de su tamaño original, se hace posible la transmisión de la información con números de bits tales como L1 (12 bits), TI (2 bits) y CRC (2 bits). Por ello, para el segundo caso, se puede esperar el mejor rendimiento de decodificación de L1 y el efecto de entrelazado de tiempos. Sin embargo, el segundo caso requiere un proceso de relleno adicional para hacer el tamaño de L1 un múltiplo de ocho si el tamaño de L1 no es un múltiplo de ocho. La Fig. 97b representa el procedimiento de relleno que se puede realizar en la señal de L1 700-L1 de la Fig. 90. Muestra que el relleno se localiza después del bloque de L1 y se cubre con la codificación CRC. En consecuencia, en un receptor, el decodificador de FEC BCH/LDPC r715-L1 en el recorrido de L1 de la Fig. 91 puede realizar la decodificación FEC, a continuación si no hay error cuando se comprueba el campo CRC, se puede realizar un análisis de bits de acuerdo con el campo de señalización de L1, a continuación se requiere un proceso que define el resto de los bits como relleno o CRC32 y que excluye el resto de los bits de los parámetros.

Para el tercer caso cuando el tamaño de L1 se expresa con un número de células mapeadas QAM, no un número de bits, se puede reducir el número de bits. Para el cuarto caso, el tamaño de L1 se expresa no como un tamaño de un bloque de L1 total, sino como un tamaño de L1 por cada símbolo OFDM. Por ello, para que un receptor obtenga un tamaño para un bloque de L1 total, se necesita realizar la multiplicación del tamaño del bloque de L1 en un símbolo OFDM simple por el número de símbolos OFDM usados en el preámbulo. En este caso, el tamaño de L1 real necesita excluir el relleno.

Para el quinto caso, mediante la expresión del bloque de L1 no como un número de bits sino como un número de células mapeadas QAM, es posible más reducción en los bits. Para el tercer a quinto casos, se muestran los parámetros TI, CRC y un número de bits de relleno necesarios. Para el caso en que el tamaño del bloque de L1 se expresa como un número de células, para que un receptor obtenga el tamaño de L1 en bits, el receptor necesita multiplicar un número de bits en el que sólo se transmiten células por un tamaño de L1 recibido. Además, se necesita excluir un número de los bits de relleno.

El último caso muestra un número total incrementado de bits hasta 32 bits mediante el uso de dos bloques de código RM en la cabecera. Los campos de CRC total llegan a ser cuatro bits debido a que cada bloque de código RM necesita dos bits en el campo CRC. Un receptor o decodificador de la cabecera FEC r1012-L1 en el recorrido de L1 de la Fig. 91, necesita obtener los parámetros necesarios mediante la realización de la decodificación FEC sobre un total de dos bloques. Usando los parámetros obtenidos, un receptor, específicamente desentrelazador de tiempos r1010-L1 en el recorrido de L1 de la Fig. 91, puede determinar si realizar o no el desentrelazado y puede obtener una profundidad de desentrelazado, si se determina realizar el desentrelazado. Además, el decodificador de FEC BCH/LDPC r715-L1 puede obtener la longitud del bloque LDPC requerida para realizar la decodificación FEC y los parámetros de recortado/perforado. Se pueden eliminar los campos de relleno y necesarios requeridos para enviar la señal de L1 a un controlador del sistema.

La Fig. 92 muestra un ejemplo de un Entrelazador de Tiempos (TI) de segmento de datos. El proceso del TI supone que son conocidas todas las posiciones piloto. El TI puede producir la salida solamente de células de datos, excluyendo pilotos. El conocimiento de las posiciones piloto habilita un número correcto de células de salida para cada símbolo OFDM. También, se puede implementar el TI mediante una memoria intermedia única en un receptor.

La Fig. 93 muestra un ejemplo de una implementación eficiente del Desentrelazador de Tiempos en un receptor. La Fig. 93a muestra cuatro esquemas diferentes de desentrelazado de acuerdo con una realización de la presente invención. La Fig. 93b muestra una memoria intermedia única que realiza el desentrelazado. La Fig. 93c muestra un esquema de ejemplo para dirigir los bloques de L1 en una matriz 2D o en una secuencia 1D.

Como se muestra en la Fig. 93a-c, el uso de un algoritmo de memoria intermedia única puede ser una implementación más eficiente del desentrelazador de tiempos. El algoritmo se puede caracterizar por la lectura de las células de salida primero desde la memoria, a continuación la escritura de las células de entrada en el que se leen las células de salida. Se puede considerar el direccionamiento diagonal como un direccionamiento circular en cada columna.

Más específicamente, con referencia a la Fig. 93a, estos cuatro procedimientos de escritura y lectura se aplican secuencialmente a las tramas C2 que se reciben en un receptor. La primera trama recibida en un receptor se escribe dentro de la memoria del desentrelazador en la Fig. 93b en el camino para el bloque 0° en la Fig. 93a y se lee en el camino para el 1er bloque. La segunda trama recibida se escribe dentro de la memoria de desentrelazador en la Fig. 93b en el camino para el 1er bloque y se lee para el 2° bloque. La tercera trama recibida se escribe dentro de la memoria del desentrelazador en la Fig. 93b en el camino para el 2° bloque y se lee en el camino para el 3er bloque. La cuarta trama recibida se escribe dentro de la memoria del desentrelazador en la Fig. 93b en el camino para el 3er bloque y se lee en el camino para el 0° bloque y así sucesivamente. Esto es, los procedimientos de escritura y lectura de la Fig. 93a se pueden aplicar secuencial y cíclicamente a las tramas C2 que se reciben secuencialmente.

El proceso de Entrelazado de Tiempos (TI) se puede realizar en los preámbulos como se muestra en la Fig. 94. Las posiciones piloto se eliminan periódica y fácilmente y no es necesario ningún entrelazado para la cabecera del bloque de L1. Esto es debido a que la cabecera del preámbulo lleva los parámetros del TI y tanto el entrelazado como el no entrelazado tienen los mismos resultados debido a la repetición. Por ello, solamente las células de señalización de L1 se entrelazan. Se puede aplicar una memoria intermedia única usada en el segmento de datos de TI.

La Fig. 95 muestra el flujo de Entrelazado/Desentrelazado de Tiempos. El entrelazado se puede realizar dentro de un bloque de L1, en lugar de en el preámbulo total. En un transmisor, como se muestra en la Fig. 128a, el bloque de L1 se pueden codificar a continuación se puede realizar un entrelazado dentro del bloque de L1 y se puede repetir el entrelazado del bloque de L1 dentro de un preámbulo. En un receptor, como se muestra en la Fig. 128b, a partir del preámbulo recibido, se puede combinar o sincronizar el bloque de L1 y se puede obtener un período simple del bloque de L1 y se puede desentrelazar el bloque de L1 combinado.

La Fig. 96 muestra unos parámetros de profundidad de entrelazado de Tiempos en la señalización de la cabecera de L1. Para la estructura de la cabecera de L1, RM (16, 32) tiene una capacidad de 16 bits. Un máximo de 2 bits de CRC puede mejorar el rendimiento BER de RM. Los campos de señalización requeridos de la cabecera de L1 son L1_info_size (15 bis) que puede requerir un máximo de 5 símbolos OFDM y TI_depth (2 bits o 1 bit). Sin embargo, un total de 18 ó 19 bis excede la capacidad de la cabecera de L1.

La Fig. 97 muestra un ejemplo de señalización de la cabecera de L1 y una estructura y un procedimiento de rellenado.

La Fig. 98 muestra un ejemplo de una señalización de L1 transmitida en una cabecera de trama. La información de señalización de L1 se puede usar como parámetros de decodificación en un receptor. Especialmente, los módulos en el recorrido de la señal de L1 de la Fig. 91 pueden realizar la decodificación de la señalización de L1 y los módulos en el recorrido del PLP en la Fig. 91 pueden usar los parámetros, por ello, se pueden decodificar los servicios. Un receptor puede obtener los parámetros de la señalización de L1 a partir de las señales del recorrido de L1 que se decodifican de acuerdo con un orden para cada campo y longitud de campo. A continuación se explica el significado de cada campo y su utilización. Se puede modificar un nombre cada campo, un número de bits para cada campo o un ejemplo de cada campo.

Num_chbon: Este campo indica un número de canales usados en unos canales adheridos. Usando este campo, un receptor puede obtener un ancho de banda total de los canales usados. El canal puede tener 6 MHz, 7 MHz, 8 MHz u otros valores de ancho de banda.

Num_dslice: Este campo indica un número de segmentos de datos que existen en un canal adherido. Después de la decodificación de la señalización de L1, un receptor accede a un bucle en el que está contenida la información de los segmentos de datos, para obtener la información de los segmentos de datos. Usando este campo, un receptor puede obtener un tamaño del bucle para decodificación.

Num_notch: este campo indica un número de bandas de supresión que existen en un canal adherido. Después de la decodificación de la señalización de L1, un receptor accede a un bloque en el que está contenida la información de la banda de supresión, para obtener la información de la banda de supresión. Usando este campo, un receptor puede obtener un tamaño del bucle para la decodificación.

Para cada segmento de datos, se puede transmitir en un preámbulo de una cabecera de tramas dslice_id, dslice_start, dslice_width, dslice_ti_depth, dslice_type, dslice_pwr_allocation y la información del PLP. El segmento de datos se puede considerar como un ancho de banda específico que contiene una o más PLP. Los servicios se pueden transmitir en los PLP. Un receptor necesita acceder al segmento de datos que contiene un PLP específica, para decodificar un servicio.

Dslice_id: Este campo se puede usar para la identificación del segmento de datos. Cada segmento de datos en un canal adherido puede tener un valor único. Cuando un receptor accede a una de los PLP para decodificar los servicios, este campo se puede usar por el receptor para diferenciar un segmento de datos en el que se localiza el PLP, de otros segmentos de datos.

Dslice_start: Este campo indica una dirección de comienzo de un segmento de datos dentro de un canal adherido. Usando este campo, un receptor puede obtener una frecuencia en la que comienza el segmento de datos. Además, la sintonía para acceder a un segmento de datos se puede realizar usando este campo.

Dslice_width: Este campo indica un ancho de banda de un segmento de datos. Usando este campo, un receptor puede obtener un tamaño de un segmento de datos. Especialmente, este campo se puede usar en un desentrelazado de tiempos para habilitar la decodificación. Junto con el campo dslice_start, un receptor puede determinar qué frecuencia decodificar a partir de las señales de RF recibidas. Este proceso se puede realizar en el Sintonizador r700 de la Fig. 91. Se puede usar información tal como dslice_start y dslice_width como señal de control del Sintonizador (r700).

Dslice_ti_depth: Este campo indica la profundidad de entrelazado de tiempos usado en los segmentos de datos de entrelazado de tiempos. Junto con dslice_width, un receptor puede obtener un ancho y una profundidad de un desentrelazado de tiempos y puede realizar el desentrelazado de tiempos. La Fig. 99 muestra un ejemplo de una dslice_ti_depth. En el ejemplo, se usan 1, 4, 8 ó 16 de los símbolos OFDM en el entrelazado de tiempos. Esto se realiza en el desentrelazador de tiempos r710 de la Fig. 91. Dslice_width y dslice_ti_depth se pueden usar como señal de control.

Dslice_type: Este campo indica un tipo de segmento de datos. El segmento de datos de Tipo 1 tiene un PLP simple dentro de ella y en el PLP se aplica CCM (codificación y modulación constante). El segmento de datos de Tipo 2 representa todas las otras clases de segmentos de datos. Usando este campo, un receptor puede realizar la decodificación de acuerdo con el PLP. El PLP de tipo 1 no tiene una cabecera FECFRAME, por ello un receptor no busca la cabecera FECFRAME. Para el tipo 2, un receptor busca la cabecera FECFRAME del PLP para obtener la información MODCOD. La Fig. 100 muestra un ejemplo de dslice_type. Usando este campo, el analizador del segmento de datos r711 de la Fig. 91 puede controlar el decodificador de cabecera del FEC r712-c, k.

Dslice_pwr_allocation: Este campo indica una potencia de un segmento de datos. Cada segmento de datos puede tener una potencia diferente de los otros segmentos de datos. Esto es para la adaptación del enlace en el sistema de cable. Un receptor puede usar este campo para controlar la potencia del segmento de datos recibida. El Sintonizador r700 de la Fig. 91 puede ajustar la ganancia de la señal usando este campo.

Num_plp: Este campo indica un número de PLP en un segmento de datos. Después de la decodificación de la señalización de L1, un receptor accede a un bucle que incluye la información del PLP. Usando este campo un receptor puede obtener un tamaño del bucle y decodificar los PLP.

Para cada PLP, se puede transmitir en una cabecera de trama (preámbulo) plp_id, plp_type, reprocesado PSI/SI, plp_payload_type, plp_modcod y plp_start_addr. Cada PLP puede transmitir una o más corrientes o paquetes tales como TS y GSE. Un receptor puede obtener los servicios mediante la decodificación de los PLP en las que se transmiten los servicios.

Plp_id: Este campo es un identificador del PLP y tiene un valor único para cada PLP en un canal adherido. Usando este campo, un receptor pueda acceder a el PLP en la que existe un servicio a decodificar. Este campo puede servir para una finalidad idéntica que el plp_id transmitido en una cabecera FECFRAME. El decodificador de la Cabecera FEC r712-c, k de la Fig. 91 puede acceder a los PLP necesarias usando este campo.

Plp_type: Este campo indica si el tipo del PLP es un PLP común o un PLP de datos. Usando este campo, un receptor puede encontrar el PLP común y puede obtener la información requerida para la decodificación de un paquete TS a partir del PLP común. Adicionalmente, el receptor puede decodificar un paquete TS dentro de un PLP de datos. La Fig. 101 muestra un ejemplo de plp_type.

Reprocesamiento PSI/SI: Este campo indica si se reprocesa o no un PSI/SI de una señal recibida. Usando este campo, un receptor puede determinar si referirse a un PSI/SI de un servicio específico de un servicio transmitido. Si un receptor no puede referirse al PSI/SI de un servicio específico de un servicio transmitido, el PSI/SI al que se puede referir mediante un servicio específico se puede transmitir a través del PLP común, por ejemplo. Usando esta información, un receptor puede decodificar los servicios.

Plp_payload_type: Este campo indica el tipo de datos de contenido útil que transmite el PLP. Un receptor puede usar este campo antes de decodificar los datos dentro de los PLP. Si un receptor no puede decodificar un tipo específico de datos, se puede impedir la decodificación del PLP que contenga ese tipo específico de datos. La Fig. 102 muestra un ejemplo de plp_payload_type. Si el segmento de datos tiene un PLP única y se aplica CCM al segmento de datos, es decir, al segmento de datos de tipo 1, se pueden transmitir adicionalmente campos tales como plp_modcod y plp_start_addr.

Plp_modcod: Este campo indica el tipo de modulación y tasa de códigos FEC usados en el PLP. Usando este campo, un receptor puede realizar la demodulación QAM y la decodificación FEC. La Fig. 103 muestra un ejemplo de plp_modcod. Esos valores mostrados en la figura se pueden usar en el modcod que se transmite en una cabecera de FECFRAME. El Desmapeador de Símbolos r713-c, k y el Decodificador de FEC BHC/LDPC r715-c, k de la Fig. 91 pueden usar este campo para la decodificación.

- Plp_start_addr: Este campo indica si aparece un primer FECFRAME de un PLP en una trama de transmisión. Usando este campo, un receptor puede obtener una localización de comienzo del FECFRAME y realizar una decodificación del FEC. Usando este campo, el Analizador de segmentos de Datos r711 de la Fig. 91 puede sincronizar los FECFRAME para los PLP de tipo 1. Para cada banda de supresión, se puede transmitir una información tal como notch_start y notch_width en una cabecera de trama (preámbulo).
- Notch_start: Este campo indica una localización de comienzo de una banda de supresión. Notch_width: Este campo indica un ancho de una banda de supresión. Usando notch_start y notch_width, un receptor puede obtener una localización y un tamaño de una banda de supresión dentro de un canal adherido. Además, se puede obtener una localización de sintonía para una decodificación del servicio correcta y se puede comprobar una existencia de un servicio dentro de un cierto ancho de banda. El Sintonizador r700 de la Fig. 91 puede realizar la sintonía usando esta información.
- GI: Este campo indica la información del intervalo de guarda usado en un sistema. Un receptor puede obtener información del intervalo de guarda usando este campo. El Sincronizador de Tiempos/Frecuencia r702 y el eliminador del GI r704 de la Fig. 91 pueden usar este campo. La Fig. 104 muestra un ejemplo.
- Num_data_Symbols: Este campo indica un número de símbolos OFDM de datos, excepto el preámbulo, usados en una trama. Se puede definir una longitud de la trama de transmisión mediante este campo. Usando este campo, un receptor puede predecir una localización de un preámbulo siguiente, por ello, se puede usar este campo para la decodificación de la señalización de L1. El Analizador de Tramas r708 de la Fig. 91 puede usar este campo y predecir los símbolos OFDM que son de preámbulo y enviar la señal al recorrido de decodificación del preámbulo.
- Num_c2_frames: Este campo indica el número de tramas que existen en una supertrama. Usando este campo, un receptor puede obtener los límites de una supertrama y puede predecir la información repetida por cada supertrama.
- Frame_idx: Este campo es un índice de trama y se repone para cada supertrama. Usando este campo, un receptor puede obtener un número de trama actual y hallar una localización de la trama actual dentro de una supertrama. Usando este campo, el analizador de Tramas r708 de la Fig. 91 puede hallar cuántas tramas preceden a una trama actual en una supertrama. Junto con num_c2_frames, se pueden predecir los cambios que ocurren en una señalización de L1 y se puede controlar la decodificación de L1.
- PAPR: Este campo indica si se usa no una reserva de tono para reducir una PAPR. Usando este campo, un receptor puede procesarlo en consecuencia. La Fig. 105 muestra un ejemplo. Por ejemplo, si se usa una reserva de tono, un receptor puede excluir las portadoras usadas en una reserva de tono, de la decodificación. Específicamente, el analizador del segmento de datos r711 de la Fig. 91 puede usar este campo para excluir las portadoras de la decodificación.
- Reservado: Este campo son bits adicionales reservados para uso futuro.
- La Fig. 106 muestra otro ejemplo de señalización de L1 transmitida en una cabecera de trama.
- En la Fig. 106, la información adicionalmente añadida a la Fig. 98 puede hacer más eficiente la decodificación del servicio por parte de un receptor. Los siguientes campos explican solamente la información adicional. Los otros campos son los mismos que en la Fig. 98.
- Network_id: Este campo indica una red a la que pertenecen las señales transmitidas. Usando este campo, un receptor puede encontrar una red actual. Cuando un receptor sintoniza con otra red para hallar un servicio en la red, el receptor puede procesarlo más rápidamente debido a que el uso solamente de la decodificación de L1 es suficiente para tomar una decisión de si la red sintonizada es una red deseada o no.
- C2_system_id: Este campo identifica un sistema al que pertenece una señal transmitida. Usando este campo, un receptor puede encontrar el sistema actual. Cuando un receptor sintoniza con otro sistema para hallar un servicio del sistema, el receptor puede procesarlo más rápidamente debido al que solamente el uso de la decodificación de L1 es suficiente para tomar una decisión de si el sistema sintonizado es un sistema deseado o no.
- C2_signal_start_frequency: Este campo indica una frecuencia de comienzo de los canales adheridos. C2_signal_stop_frequency: Este campo indica un fin de la frecuencia de los canales adheridos. Usando c2_signal_start_frequency y c2_signal_stop_frequency, se pueden encontrar los anchos de banda de RF de todos los segmentos de datos mediante la decodificación de L1 de cierto ancho de banda dentro de los canales adheridos. Además, este campo se puede usar para obtener una cantidad de desplazamiento de frecuencia requerida en la sincronización de los L1_XFEC_FRAME. El Combinador de XFEC de L1 r1017-L1 de la Fig. 91 puede usar este campo. Además, cuando un receptor recibe segmentos de datos situados en ambos extremos del canal adherido, este campo se puede usar para sintonizar a una frecuencia apropiada. El Sintonizador r700 de la Fig. 91 puede usar esta información.
- Plp_type: Este campo indica si un PLP es un PLP común, un PLP de datos normales o un PLP de datos agrupados. Usando este campo, un receptor puede identificar los PLP comunes y puede obtener la información requerida para

la decodificación del paquete TS a partir del PLP común, a continuación puede decodificar el paquete TS dentro de un PLP de datos agrupados. En este caso, el PLP común puede ser un PLP que contenga datos compartidos por múltiples PLP. La Fig. 107 muestra un ejemplo de este campo. El PLP de datos normales es un PLP de datos que no tiene un PLP común. En este caso, un receptor no necesita hallar un PLP común. Un PLP común o PLP agrupada puede transmitir información tal como `plp_group_id`. Para los otros tipos de PLP, es posible una transmisión más eficiente debido a que no se necesita transmitir información adicional.

`Plp_group_id`: Este campo indica un grupo al que pertenece un PLP actual. El PLP de datos agrupados puede transmitir los parámetros de TS común usando el PLP común. Usando este campo, si el PLP decodificada actualmente es un PLP agrupada, un receptor puede hallar un PLP común necesaria, obtener los parámetros requeridos para el paquete TS de PLP agrupadas y formar un paquete TS completo.

`Reserved_1/reserved_2/reserved_3`: estos campos son bits adicionales reservados para uso futuro para un bucle de segmento de datos, un bucle de PLP y una trama de transmisión, respectivamente.

La Fig. 108 muestra otro ejemplo de señalización de L1 transmitida en una cabecera de trama. Comparada con la Fig. 106, se puede transmitir una información más optimizada, por ello, puede tener lugar una sobrecarga de señalización menor. En consecuencia, un receptor puede decodificar los servicios eficientemente. Especialmente, los módulos en el recorrido de la señal de L1 de la Fig. 91 pueden realizar la decodificación de la señalización de L1 y los módulos en el recorrido del PLP de la Fig. 91 pueden usar los parámetros, por ello, se pueden decodificar los servicios. Un receptor puede obtener los parámetros de señalización de L1 a partir de las señales del recorrido de L1 que se decodifican de acuerdo con un orden para cada campo y longitud de campo. Se puede modificar un nombre de cada campo, un número de bits para cada campo o un ejemplo de cada campo. Las descripciones de los campos excepto `dslice_width` son idénticas a las anteriormente mencionadas descripciones de campos. Una función de `dslice_width` de acuerdo con un ejemplo es como sigue.

`Dslice_width`: Este campo indica un ancho de banda de un segmento de datos. Usando este campo, un receptor puede obtener un tamaño de un segmento de datos. Especialmente, este campo se puede usar en un desentrelazado de tiempos para habilitar la decodificación. Junto con el campo `dslice_start`, un receptor puede determinar qué frecuencia decodificar de las señales de RF recibidas. Este proceso se puede realizar en el Sintonizador r700 de la Fig. 91. Se puede usar una información tal como `dslice_start` y `dslice_width` como señal de control del Sintonizador r700. En este punto, el ancho de un segmento de datos se puede extender hasta 64 MHz mediante el uso de 12 bits para este campo de `dslice_width`. Usando este campo, un receptor puede determinar si un sintonizador disponible actualmente puede decodificar el segmento de datos actual. Si un ancho de un segmento de datos es mayor que un ancho de banda de un sintonizador antiguo de un receptor, para decodificar tal segmento de datos, un receptor puede usar o bien al menos dos sintonizadores antiguos o un sintonizador con un ancho de banda suficientemente grande. En el ejemplo, una granularidad de los valores usados en `dslice_start`, `dslice_width`, `notch_start` y `notch_width` puede ser 12 portadoras (células) OFDM. En otras palabras, un receptor puede hallar una localización de una célula OFDM actual mediante la multiplicación de los valores transmitidos por 12. En el ejemplo, para una granularidad de `Plp_start_addr`, se puede usar una portadora (célula) OFDM. En otras palabras, un receptor puede hallar cuantos símbolos OFDM y células OFDM van por delante de una localización de comienzo de un PLP dentro del símbolo OFDM. `Dslice_start` y `dslice_width` se pueden usar para esta finalidad. El Analizador del segmento de Datos r711 de la Fig. 91 puede realizar tal proceso.

La Fig. 109 muestra un ejemplo del procesamiento en la cabecera FEC 705-L1 en el recorrido de L1 de la Fig. 90. Se pueden transmitir un total de 16 bits en la cabecera FEC de un recorrido de L1. Se pueden asignar catorce bits para `L1_info_size`. Si `L1_info_size` tiene un valor que es la mitad de la longitud del bloque de L1 transmitido realmente, un receptor puede multiplicar el `L1_info_size` recibido por dos para obtener la longitud real del bloque de L1 y comenzar la decodificación de L1. Esta longitud obtenida del bloque de L1 es una longitud que incluye el relleno.

Para el bloque de L1 que se determina que no tiene error a través de la comprobación de CRC, un receptor puede considerar el resto de los bits después de la decodificación de L1 como relleno. Los últimos dos bits, de modo similar a procedimientos previos, se pueden usar para indicación de la profundidad de entrelazado de tiempos de los preámbulos. El mapeador del preámbulo 1007-L1 de la Fig. 90 puede determinar los símbolos OFDM requeridos para transmitir los bloques de L1. Posteriormente, el Entrelazado de tiempos 1008-L1 de la Fig. 90 puede realizar el entrelazado de tiempos. Usando la información de la profundidad de entrelazado de tiempos y `L1_info_size`, un receptor puede hallar qué tamaño del bloque de L1 se transmite y cuantos símbolos OFDM. La combinación, mezcla y desentrelazado de tiempos de los bloques de L1 se pueden realizar en el Combinador de XFEC de L112417-L1, el Fusionador de L1_FEC 12418-L1 y el desentrelazador de Tiempos 12410-L1 de la Fig. 91, respectivamente.

En un receptor en la Fig. 91, una longitud de un bloque XFEC de L1 dentro del símbolo OFDM se puede obtener mediante la división de una longitud de bloque de L1 total por un número de los símbolos OFDM usados en un preámbulo. El número de los símbolos OFDM se puede obtener a partir de un valor definido en `ti_depth`. El Combinador de XFEC de L112417-L1 de un receptor puede obtener el bloque XFEC de L1. A continuación, se puede realizar el desentrelazado de Tiempos 12410-L1 usando `ti_depth`. Finalmente, se pueden mezclar los bloques XFEC de L1 para obtener un bloque L1_FEC. Después de que el mezclado de L1_FEC 12418-L1, el Desentrelazado

de bits r714-L1 y la decodificación LDPC/BCH r715-L1, se puede obtener el bloque de L1. L1_info_size se puede multiplicar por dos, se puede comprobar el CRC del bloque de L1 y se pueden codificar L1. Se puede descartar el relleno innecesario.

La Fig. 110 muestra otro ejemplo de señalización de L1 transmitida en una cabecera de trama. Comparada con la Fig. 108, se modifican los números de bits de algunos campos y se añaden algunos campos para mejorar una eficiencia de decodificación del servicio por parte de un receptor. Especialmente, los módulos en el recorrido de la señal de L1 de la Fig. 91 pueden realizar la decodificación de la señalización de L1 y los módulos en el recorrido del PLP de la Fig. 91 pueden usar parámetros, por ello, se pueden decodificar los servicios. Un receptor puede obtener los parámetros de la señalización de L1 a partir de las señales del recorrido de L1 que se decodifican de acuerdo con un orden de cada campo y una longitud de campo. Se puede modificar un nombre de cada campo, un número de bits de cada campo o un ejemplo de cada campo. Excepto los campos modificados respecto a la figura previa, las descripciones de los campos son idénticas a las descripciones de campos mencionadas anteriormente.

RESERVED_1, RESERVED_2, RESERVED_3 y RESERVED_4 son campos reservados para uso futuro. En el ejemplo PLP_START puede indicar una información idéntica a la anteriormente mencionada plp_start_addr.

L1_PART2_CHANGE_COUNTER indica un número de tramas desde la primera trama a una trama que tiene un cambio cualquiera en la información de señalización de L1, excluyendo el cambio en PLP_START, respecto a las tramas previas. Esto es, este campo indica el número de tramas por delante en los que cambia la configuración. Usando este campo, un receptor puede omitir la decodificación de L1 para cada trama para obtener la información de L1. En otras palabras, mediante el uso del valor de L1_PART2_CHANGE_COUNTER, un receptor puede determinar qué trama ha cambiado en la información de L1 respecto a las tramas previas, por ello, no se realiza ninguna decodificación de L1 para tramas antes de que tenga lugar una trama con cambios en L1, a continuación se puede realizar la decodificación de L1 para la trama que tiene el cambio en L1. Por ello, se pueden omitir las operaciones no necesarias. Usando este campo, un receptor puede evitar una operación de decodificación de L1 redundante. Este valor se puede calcular también por un receptor con información de L1 ya decodificada.

Si L1_PART2_CHANGE_COUNTER es 0, significa que no ha habido cambios en L1 durante al menos 256 tramas (2^8 , 8 es el número de los bits usados para L1_PART2_CHANGE_COUNTER). En este de los mejores casos, un receptor necesita decodificar L1 solamente cada 51 segundos. Este proceso se puede realizar en el Analizador de Tramas r708 de la Fig. 91. El Analizador de Tramas puede determinar si el preámbulo actual tiene un cambio en L1 y puede controlar los procesos posteriores en el recorrido de señalización de L1. Un receptor puede calcular PLP_START para una trama específica a partir de los PLP_START y PLP_MODCOD ya obtenidos, sin realizar la decodificación de L1 para obtener PLP_START.

La Fig. 111 muestra ejemplos de campos mostrados en la Fig. 110. Los bloques del receptor puedan realizar los procesos de acuerdo con los valores indicados por los campos en los ejemplos.

La Fig. 112 muestra otro ejemplo de la señalización de L1 transmitida en una cabecera de trama. Comparada con la Fig. 110, se modifican algunos campos y se añaden algunos campos para mejorar una eficiencia de la decodificación del servicio por parte de un receptor. Especialmente, los módulos en el recorrido de la señal de L1 de la Fig. 91 pueden realizar la decodificación de la señalización de L1 y los módulos en el recorrido del PLP de la Fig. 91 pueden usar parámetros, por ello, se pueden decodificar los servicios. Un receptor puede obtener los parámetros de la señalización de L1 a partir de las señales del recorrido de L1 que se decodifican de acuerdo con un orden de cada campo y una longitud de campo. Se puede modificar un nombre de cada campo, un número de bits de cada campo o un ejemplo de cada campo. Excepto los campos modificados respecto a la figura previa, las descripciones de los campos son idénticas a las descripciones de campos mencionadas anteriormente.

Las descripciones de DSLICE_START, DSLICE_WIDTH, NOTCH_START y NOTCH_WIDTH son idénticas a las descripciones previas. Sin embargo se puede minimizar la sobrecarga de la señalización mediante la señalización de los campos con un número mínimo de bits de acuerdo con el modo de GI. En consecuencia, se puede decir que la señalización de DSLICE_START, DSLICE_WIDTH, NOTCH_START y NOTCH_WIDTH se basa en el modo de GI. La información de L1 se puede obtener a partir del recorrido de la señal de L1 de un receptor de la Fig. 91. Un controlador del sistema puede determinar un número de bits usados para cada campo de acuerdo con el valor de GI obtenido y puede leer los campos en consecuencia. Se necesita transmitir el valor de GI antes que otros valores.

En lugar de DSLICE_START y DSLICE_WIDTH, se pueden transmitir 12 bits de posición de sintonía que indican una localización optimizada para obtener el segmento de datos y 11 bits del valor de desplazamiento a partir de la posición de sintonía para indicar un ancho de un segmento de datos. Especialmente, mediante el uso de los 11 bits del valor de desplazamiento, se pueden señalar los segmentos de datos que ocupan un máximo de 8 canales adheridos y un receptor que pueda recibir tales segmentos de datos puede funcionar apropiadamente. Un sintonizador r700 de un receptor de la Fig. 91 puede determinar el ancho de banda de RF usando una posición de sintonía y puede obtener un ancho del segmento de datos usando el valor de desplazamiento, para servir a las mismas finalidades que la DSLICE_WIDTH anteriormente mencionada.

DSLICE_CONST_FLAG es un campo para la indicación de si una configuración de un segmento de datos específica se mantiene constante. Usando este campo obtenido a partir de una L1 desde un cierto ancho de banda, un receptor puede determinar si un segmento de datos específico tiene una configuración constante, a continuación el receptor puede recibir los PLP del segmento de datos específico sin decodificación de L1 adicional. Esta clase de proceso puede ser útil para la recepción de segmentos de datos que se localizan en un ancho de banda en el que no está disponible la decodificación de L1.

DSLICE_NOTCH_FLAG es un campo o un marcador para indicación de las bandas de supresión en ambos bordes de un segmento de datos específico. Se puede usar el Bit Más Significativo (MSB) como un indicador para la vecindad de la banda de supresión en un ancho de banda bajo y se puede usar el Bit Menos Significativo (LSB) como un indicador de la vecindad de la banda de supresión en un ancho de banda elevado. Usando el campo, cuando un receptor decodifica un segmento de datos específico, el receptor puede tener en cuenta una banda de supresión hallando los cambios en las portadoras activas producidos por la vecindad de pilotos continuos en ambos extremos de una banda de supresión. Esta información se puede obtener también a partir de la información de la supresión transmitida en NOTCH_START y NOTCH_WIDTH. El Desentrelazador de Tiempos r710 de un receptor de la Fig. 91 puede usar la información para hallar la localización de portadoras activas y enviar solamente datos que correspondan a las portadoras activas, al analizador de segmentos de datos.

Para PLP_TYPE, se añade un bit adicional a la Fig. 110. La Fig. 113 muestra un ejemplo de PLP_TYPE de la Fig.

112. Se puede transmitir un valor que indique PLP de datos empaquetados. Se puede multiplexar una corriente de TS grande que tenga una tasa de datos elevada en múltiples PLP. Los PLP de datos empaquetados se pueden usar para indicar los PLP en las que se transmiten las corrientes multiplexadas. Para un receptor heredado que sea incapaz de decodificar un PLP específica, este campo puede impedir que el receptor acceda a el PLP, por ello, se puede impedir un posible fallo.

Como un procedimiento alternativo más, si se usa el dslice_width anteriormente mencionado junto con el campo dslice_start y la información de supresión, un receptor puede determinar qué frecuencia decodificar a partir de las señales de RF recibidas. Este proceso se puede realizar en el Sintonizador (r700) de la Fig. 91. Se puede usar información tal como dslice_start, dslice_width, notch_start y notch_width como señales de control del Sintonizador r700. Así, la obtención de un segmento de datos y la sintonía simultánea de una banda de RF en la que no existen problemas de decodificación de L1 se hace posible, al evitar la supresión.

En relación a la señalización de L1 de la Fig. 112, la Fig. 114 muestra una relación entre la señalización de L1 y la señalización de L2 cuando un PLP es del tipo empaquetado. Además, la Fig. 114 muestra también una acción que se puede tomar por un receptor en tal caso. El TS1 se puede mapear en el PLP37 a través de c2dsd de L2. Este TS1 corresponde a un PLP normal de L1, por ello, el PLP se puede decodificar por un receptor normal (sintonizador de 8 MHz simple) y un receptor de alta calidad (sintonizador múltiple o sintonizador de banda ancha (>8 MHz)). TS2 y TS3 se mapean en PLP39 y PLP44 respectivamente, a través de c2dsd. Éstas corresponden a el PLP empaquetada de L1, por ello, estas PLP se pueden decodificar por un receptor de alta calidad (sintonizador múltiple o sintonizador de banda ancha (>8 MHz)) pero no por un receptor normal (sintonizador de 8 MHz simple). En consecuencia, de acuerdo con la información de L1, un receptor puede comprobar si se recibió o no el TS correspondiente.

La Fig. 115 y la Fig. 116 son diagramas de flujo que describen las acciones de decodificación de L1 y de decodificación de L2 para los tipos de PLP empaquetada y tipos de PLP normales en un receptor normal y un receptor de alta calidad, respectivamente. La Fig. 117 muestra un ejemplo de la estructura c2_deliver y _system_descriptor y la sintaxis para la señalización de L2 que se tiene cuenta en la Fig. 112. Este descriptor puede mapear TS_id en plp_id como se muestra en la Fig. 114. La información empaquetada se puede procesar en L1, por ello, no se necesita señalizarla en L2. Las variables mostradas en la Fig. 117 se describen a continuación.

Plp_id: Este campo de 8 bits identifica de modo único un PLP de datos dentro de un Sistema C2.

C2_system_id: Este campo de 16 bits identifica de modo único un sistema C2. La parte restante de este descriptor, que sigue inmediatamente al campo C2_system_id sólo está presente una vez por sistema C2, debido a que los parámetros son únicamente aplicables a todos los segmentos de datos llevadas a través de un Sistema C2 particular. Una presencia o ausencia de esa parte se puede deducir del campo de longitud del descriptor. En ausencia de la parte restante, esta longitud es igual a 0x07, en caso contrario se asigna un valor más grande.

C2_Sytem_tuning_frequency: Este campo de 32 bits indica un valor de frecuencia. El intervalo de codificación puede ser desde un mínimo de 1 Hz (0x00000001) hasta un máximo de 4, 294, 967, 295 Hz (0xFFFFFFFF). Este campo de datos puede dar una frecuencia de sintonía, en la que se transmite un preámbulo completo dentro de una ventana de sintonía. En general la C2_Sytem_tuning_frequency es la frecuencia central de un C2_System, pero se puede desviar de la frecuencia central en caso de que existan supresiones en esta área.

Active_OFDM_symbol_duration: Este campo de 3 bits indica una duración del símbolo OFDM activo. Se muestra un ejemplo en la Fig. 118.

Guard_interval: Este campo de 3 bits indica un intervalo de guarda. Se muestra un ejemplo en la Fig. 119.

En los ejemplos previos de entrelazador/desentrelazador de tiempos de L1, para los casos en los que TI_DEPTH es "10" o "11", el mapeador de Preámbulos 1007-L1 de la Fig. 90 puede dividir uniformemente el bloque de L1 original en cuatro u ocho subbloques. Sin embargo, si un tamaño del subbloque es más pequeño que un tamaño mínimo requerido para realizar una codificación FEC, no se puede realizar apropiadamente la codificación FEC. Una posible solución puede ser el establecimiento de un umbral. Si un tamaño del bloque de L1 es más pequeño que un umbral establecido, el bloque de L1 se puede repetir cuatro u ocho veces para casos en los que TI_DEPTH es "10" o "11". Si el tamaño de un bloque de L1 es más grande que un umbral establecido, se puede dividir uniformemente el bloque de L1 en cuatro u ocho subbloques. Se puede establecer el umbral como cuatro u ocho veces un tamaño máximo requerido para realizar una codificación FEC.

Además, el establecimiento de TI_DEPTH como "10" o "11" es para los casos en los que el efecto de entrelazado de tiempos no se obtiene debido a un tamaño de bloque de L1 pequeño. Por ello, se puede definir el umbral como un tamaño de bits de información que se puede transmitir mediante un símbolo de preámbulo simple. Por ejemplo, si se supone una codificación de FEC de L1 idéntica a DVB-T2, un umbral puede ser 4.772 bits.

Para los casos en los que TI_DEPTH es "10" o "11", el uso de la información del tamaño de L1, profundidad de TI, como un valor de umbral compartido entre un transmisor y un receptor, los módulos de un receptor, desde el decodificador de cabecera de FEC r1012-L1 a L1_FEC_Merger r1018-L1 de la Fig. 91 pueden determinar un tamaño del subbloque de L1, combinando y mezclando los bloques de L1 que se transmiten en un símbolo de OFDM de un preámbulo.

Si un tamaño de L1 es más pequeño que un valor de umbral, L1_FEC_Merger r1018-L1 de la Fig. 91 no necesita mezclar los subbloques divididos debido a que el bloque de L1 original se transmite repetidamente de acuerdo con un TI_DEPTH en cuatro u ocho símbolos OFDM. Sin embargo, si un tamaño de L1 es mayor que un valor de umbral, debido a que se usa un número de símbolos que es mayor que el número de los símbolos OFDM requeridos para transmitir el bloque de L1, el decodificador de cabecera de FEC r1012-L1 de la Fig. 91 puede obtener un tamaño de un subbloque usando TI_DEPTH. A continuación, el combinador de FEC de L1 r1017-L1 puede combinar los bloques FEC de L1 y el desentrelazador de tiempos r1010-L1 puede realizar el desentrelazado. Finalmente, el fusionador de FEC de L1 r1018-L1 puede mezclar los bloques de FEC de L1 para restaurar el bloque L1 original.

La Fig. 120 muestra otro ejemplo de la señalización de L1 que se transmite en una cabecera de trama. Comparada con la Fig. 112, se modifican algunos campos y se añaden algunos campos para mejorar una eficiencia de la decodificación del servicio por parte de un receptor. Especialmente, los módulos en el recorrido de la señal de L1 de la Fig. 91 pueden realizar la decodificación de la señalización de L1 y los módulos en el recorrido del PLP de la Fig. 91 pueden usar parámetros, por ello, se pueden decodificar los servicios. Un receptor puede obtener los parámetros de la señalización de L1 a partir de las señales del recorrido de L1 que se decodifican de acuerdo con un orden de cada campo y una longitud de campo. Se puede modificar un nombre de cada campo, un número de bits de cada campo o un ejemplo de cada campo. Excepto los campos modificados respecto a la figura previa, las descripciones de los campos son idénticas a las descripciones de campos mencionadas anteriormente.

DSLICE_TUNE_POS indica una posición de sintonía para que un receptor obtenga un segmento de datos. Dependiendo del modo de GI, este valor se puede expresar en 12 u 11 bits. DSLICE_OFFSET_RIGHT y DSLICE_OFFSET_LEFT que indican el valor de desplazamiento a partir de una posición de sintonía o un ancho de un segmento de datos, se pueden expresar en 9 u 8 bits, dependiendo de un modo de GI. Si el desplazamiento puede tener un valor con signo, es decir, un valor positivo o negativo, se puede expresar también una posición de un ancho de un segmento de datos que tenga una banda estrecha. El Sintonizador r700 de un receptor en la Fig. 91 puede determinar una banda de RF usando una posición de sintonía. A continuación usando este valor de desplazamiento con signo, se puede obtener el ancho del segmento de datos. De ese modo, este campo puede servir para la misma finalidad que el DSLICE_WIDTH mencionado anteriormente. Un receptor puede obtener el ancho de bits usando un valor de GI.

DSLICE_NOTCH_FLAG es un marcador que indica que un cierto segmento de datos es adyacente a una banda de supresión. Puede servir para la misma finalidad que los ejemplos mencionados anteriormente pero aquí, solamente se usa 1 bit para este campo por cada segmento de datos. Usando este 1 bit de información, un receptor puede realizar la misma función que los ejemplos mencionados anteriormente.

PLP_BUNDLED_FLAG indica que un PLP es un PLP de datos empaquetados. Esto es, PLP_BUNDLED_FLAG indica si un PLP está o no empaquetada con otras PLP dentro del sistema de difusión. Este campo puede servir para una misma finalidad que el PLP de datos empaquetados anteriormente mencionada de PLP_TYPE de la Fig.

112. El PLP_TYPE se muestra en la Fig. 110.

La Fig. 121 muestra otros dos ejemplos de entrelazado de tiempos que se pueden usar en el recorrido de L1 de la Fig. 90. Como se ve en Entrelazado de Tiempos CONECTADO (1), el entrelazado puede ser solamente entrelazado de bloques. Comparado con el procedimiento mostrado en la Fig. 83, el rendimiento del entrelazado de frecuencias puede no ser tan bueno como el procedimiento mostrado en la Fig. 83. Sin embargo, en los casos en que TI_DEPTH

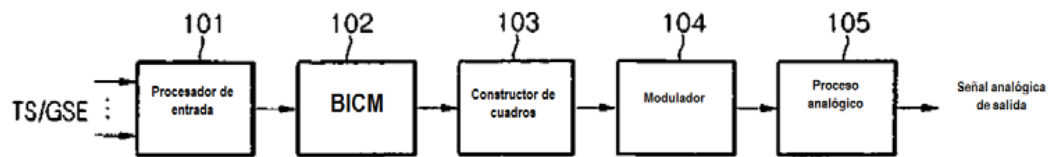
- es "10" o "11", sin repetición o división de los bloques de L1 de acuerdo con un umbral, los bloques de L1 se pueden extender en la dirección del tiempo independientemente del tamaño del bloque de L1 entonces se puede repetir en un preámbulo si hay espacio en el preámbulo, por ello, este procedimiento puede ser ventajoso en que se puede simplificar el control. El entrelazado se puede realizar mediante la escritura de las corrientes de símbolos de entrada en la dirección del tiempo y la lectura de las corrientes de símbolos escritos en la dirección de la frecuencia. El desentrelazador de tiempos r1010-L1 en el recorrido de L1 de un receptor de la Fig. 91 puede realizar el desentrelazado mediante la escritura de las corrientes de símbolos de entrada en la dirección de la frecuencia y la lectura de las corrientes de símbolos escritas en la dirección del tiempo.
- 5
- 10 Un segundo ejemplo o el Entrelazado de Tiempos CONECTADO (2) de la Fig. 121 incluye procesos adicionales al Entrelazado de Tiempos CONECTADO (1), que es un desplazamiento circular en una dirección de filas. Mediante este proceso, además de las ventajas del Entrelazado de Tiempos CONECTADO (1), se puede obtener un efecto de extensión en un dominio de la frecuencia. El desentrelazador de tiempos r1010-L1 en el recorrido de L1 de un receptor de la Fig. 91 necesita realizar un nuevo desplazamiento circular en una dirección de filas antes de realizar el proceso del Entrelazado de Tiempos CONECTADO (1).
- 15 Usando los procedimientos y dispositivos sugeridos, entre otras ventajas es posible implementar un transmisor, receptor digital eficiente y una estructura de señalización de la capa física.
- Mediante la transmisión de la información de ModCod en cada cabecera de trama de BB que sea necesaria para ACM/VCM y la transmisión del resto de la señalización de la capa física en una cabecera de trama, se puede 5 minimizar la sobrecarga por señalización.
- 20 Se puede implementar un QAM modificado para una transmisión energéticamente más eficiente o un sistema de difusión digital más robusto frente al ruido. El sistema puede incluir el transmisor y receptor para cada ejemplo desvelado y las combinaciones de los mismos.
- Se puede implementar una QAM no uniforme mejorada para una transmisión energéticamente más eficiente o un sistema de difusión digital más robusto frente al ruido. Se describe también un procedimiento del uso de la tasa de códigos del código de corrección de error de NU-QAM y MQAM. El sistema puede incluir el transmisor y receptor para cada ejemplo desvelado y las combinaciones de los mismos.
- 25 El procedimiento de señalización de L1 sugerido puede producir la sobrecarga en un 3~4% minimizando la sobrecarga de señalización durante la adhesión del canal.
- 30 Será evidente para los expertos en la materia que se pueden realizar varias modificaciones y variaciones en la presente invención sin apartarse de la misma.

REIVINDICACIONES

1. Un transmisor para la transmisión de datos de difusión, comprendiendo el transmisor:
 - unos primeros medios de codificación de Corrección de Error Directo, FEC, (702-0) para los datos del Conducto de la Capa Física, PLP, de codificación FEC;
- 5 unos segundos medios de codificación FEC (702-L1) para los datos de señalización de Capa 1 de codificación FEC;
 - unos medios de construcción de tramas (711) para ensamblar los datos PLP codificados FEC y los datos de señalización de Capa 1 codificados FEC para formar una trama de señal, en donde la trama de señal incluye símbolos de preámbulo que transportan los datos de señalización de Capa 1 y símbolos de datos que transportan los datos de PLP,
- 10 en donde los símbolos de preámbulo incluyen bloques de Capa 1, L1, que se repiten en un dominio de frecuencia, en donde cada uno de los bloques de L1 incluye 3408 subportadoras,
 - en donde los datos de señalización de la Capa 1 incluyen una información del PLP_type que indica si un PLP es un PLP común o un PLP de datos normales o un PLP de datos agrupados y los datos de señalización de la Capa 1 incluyen además una información PLP_bundle_Flag, la información PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y
- 15 en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.
2. El transmisor de la reivindicación 1, en donde el transmisor además comprende:
 - medios de entrelazado en el tiempo (1008-L1) para entrelazar en el tiempo los datos de señalización de L1 del bloque de L1 con una profundidad de entrelazado en el tiempo.
- 20 3. El transmisor de la reivindicación 2, en donde cada uno de los bloques de L1 incluye una cabecera de L1 la cual señala la profundidad de entrelazado en el tiempo.
4. Un receptor para el procesamiento de los datos de difusión, comprendiendo el receptor:
 - unos medios de análisis de tramas (r708) para analizar una trama de señal que incluye símbolos de preámbulo que transportan datos de señalización de Capa 1 y símbolos de datos que transportan datos del Conducto de la Capa Física, PLP; y sacar los datos de PLP y los datos de señalización de Capa 1;
- 25 unos primeros medios de decodificación de Corrección de Error Directo, FEC, (r715-C) para decodificar FEC los datos del PLP; y
- 30 unos segundos medios de decodificación de FEC (r715-L1) para decodificar FEC los datos de señalización de Capa 1,
 - en donde los símbolos de preámbulo incluyen bloques de Capa 1, L1, que se repiten en un dominio de frecuencia, en donde cada uno de los bloques de L1 incluye 3408 subportadoras,
 - en donde los datos de señalización de la Capa 1 incluyen una información del PLP_type que indica si un PLP es un PLP común o un PLP de datos normales o un PLP de datos agrupados y los datos de señalización de la Capa 1 incluyen además una información de PLP_bundle_Flag, la información PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y
- 35 en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.
5. El receptor de la reivindicación 4, en donde el receptor comprende:
 - medios de desentrelazado en el tiempo (r1010-L1) para desentrelazar en el tiempo los datos de señalización de L1 del bloque de L1 con una profundidad de entrelazado en el tiempo.
- 40 6. El receptor de la reivindicación 5, en donde cada uno de los bloques de L1 incluye una cabecera de L1 que señala la profundidad de entrelazado en el tiempo.
7. Un método para la transmisión de datos de difusión, comprendiendo el método:
 - codificar de Corrección de Error Directo, FEC, los datos del Conducto de la Capa Física, PLP;
- 45 codificar FEC los datos de señalización de Capa 1; y

- ensamblar los datos de PLP codificados FEC y los datos de señalización de Capa 1 codificados FEC para formar una trama de señal, en donde la trama de señal incluye símbolos de preámbulo que transportan los datos de señalización de Capa 1 y símbolos de datos que transportan los datos del PLP,
- 5 en donde los símbolos de preámbulo incluyen bloques de Capa 1, L1, que se repiten en un dominio de frecuencia, en donde cada uno de los bloques de L1 incluye 3408 subportadoras,
- en donde los datos de señalización de la Capa 1 incluyen una información del PLP_type que indica si un PLP es un PLP común o un PLP de datos normales o un PLP de datos agrupados y los datos de señalización de la Capa 1 incluyen además una información PLP_bundle_Flag, la información PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y
- 10 en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.
8. El método de la reivindicación 7, en donde el método además comprende:
- entrelazar en el tiempo los datos de señalización de L1 del bloque de L1 con una profundidad de entrelazado en el tiempo.
- 15 9. El método de la reivindicación 8, en donde cada uno de los bloques de L1 incluye una cabecera de L1 la cual señala la profundidad de entrelazado en el tiempo.
10. Un método para el procesamiento de los datos de difusión, comprendiendo el método:
- analizar una trama de señal que incluye símbolos de preámbulo que transportan datos de señalización de Capa 1 y símbolos de datos que transportan datos de Conducto de la Capa Física, PLP; y sacar los datos de PLP y los datos de señalización de Capa 1;
- 20 decodificar con Corrección de Error Directo, FEC, decodificando los datos de PLP; y
- decodificar FEC los datos de señalización de Capa 1,
- en donde los símbolos de preámbulo incluyen bloques de Capa 1, L1, que se repiten en un dominio de frecuencia, en donde cada uno de los bloques de L1 incluye 3408 subportadoras,
- 25 en donde los datos de señalización de la Capa 1 incluyen una información del PLP_type que indica si un PLP es un PLP común o un PLP de datos normales o un PLP de datos agrupados y los datos de señalización de la Capa 1 incluyen además una información de PLP_bundle_Flag, la información PLP_bundle_Flag que indica si el PLP está empaquetado o no con otros PLP dentro de un sistema de difusión, y
- 30 en donde el PLP empaquetado que transporta al menos un servicio se demultiplexa en múltiples segmentos de datos.
11. El método de la reivindicación 10, en donde el método comprende:
- desentrelazar en el tiempo los datos de señalización de L1 del bloque de L1 con una profundidad de entrelazado en el tiempo.
- 35 12. El método de la reivindicación 11, en donde cada uno de los bloques de L1 incluye una cabecera de L1 que señala la profundidad de entrelazado en el tiempo.

Fig. 1



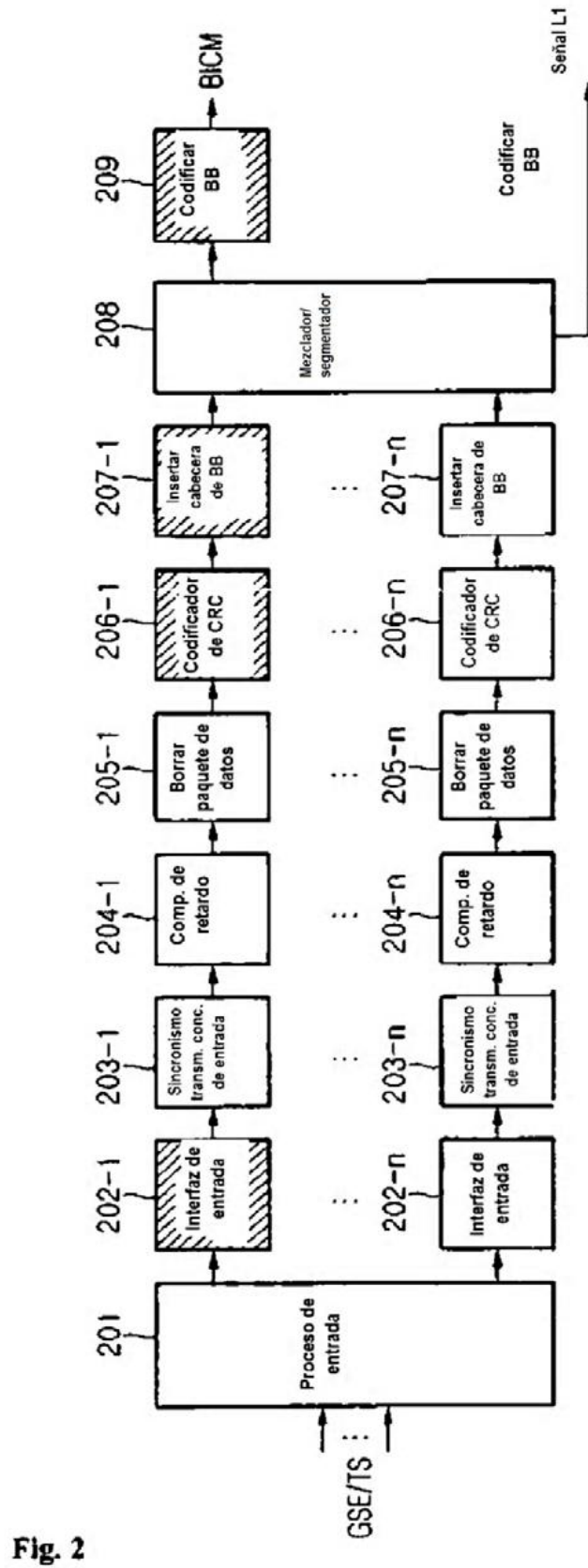


Fig. 3

TS/GS (2 bits)	SIS/MIS (1 bit)	CCM/ACM (1 bit)	ISSYI (1 bit)	NPD(lbit)	EXT (2bits)
00 = GFPS 11 = TS 01 = GCS 10 = GSE	1 = single 0 = múltiple	1 = CCM 0 = ACM	1 = activo 0 = no activo	1= activo 0 = no activo	Reservado para uso futuro

Campo	Tamaño (Octetos)	Descripción
MATYPE	2	Como se ha descrito anteriormente
UPL	2	Usar longitud del paquete de bits, en el intervalo [0,65535]
DFL	2	Longitud del campo de datos en bits, en el intervalo [0,53760]
SYNC	1	Una copia del octeto de sincronismo del paquete usuario
SYNCD	2	La distancia en bits desde el comienzo del campo de DATOS al primer UP completo del campo de datos. SYNCD = 0 _D significa que el primer UP está alineado con el comienzo del campo de datos. SYNCD = 65535 _D significa que ningún Up comienza en el campo de DATOS.
MODO CRC-8	1	El XOR del campo CRC-8 (1 octeto) con el campo MODE (1 octeto). CRC-8 es el código de detección de error aplicado a los primeros 9 octetos del BBHEADER. MODE (8 octetos) deberá ser: • 0_D Modo normal • 1_D Modo de alta eficiencia • Otros valores: reservados para uso futuro.

Fig.4

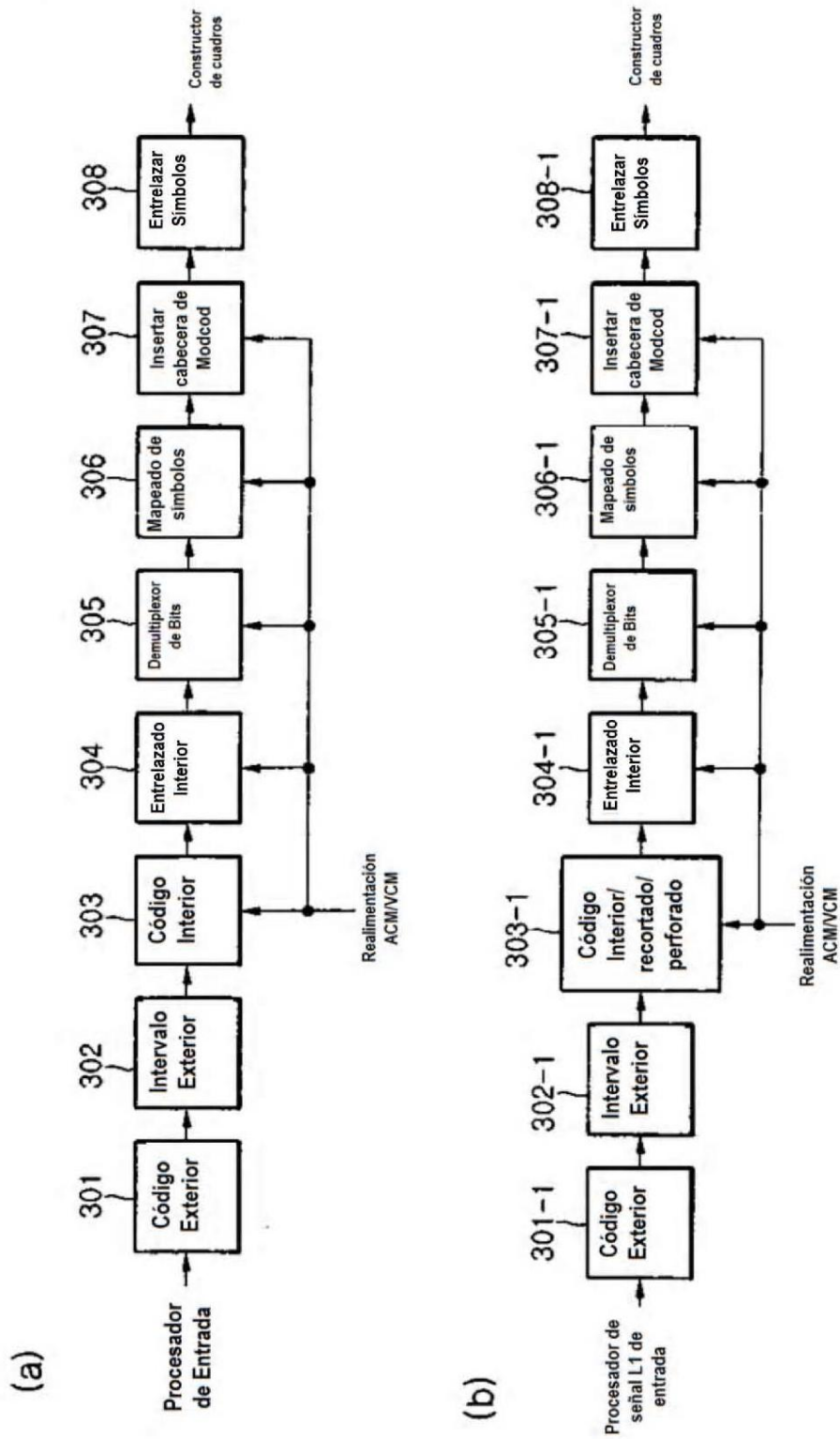


Fig. 5

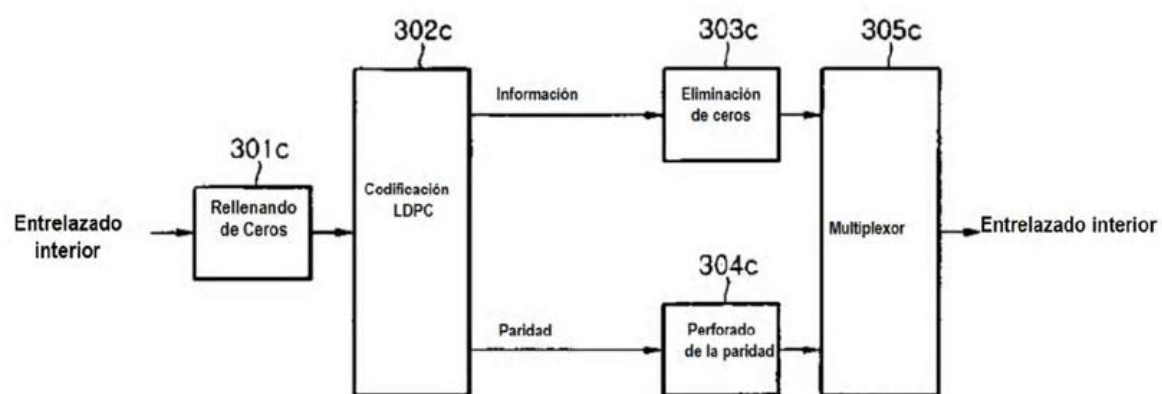


Fig. 6

			Caso 1	Caso 2
Capacidad (bit/s/Hz)			Modulación	Modulación
6	1/2	3,0	NU-MQAM	NU-QAM
	2/3	4,0	NU-MQAM	NU-QAM
	3/4	4,5	NU-MQAM	NU-MQAM
	4/5	4,8	MQAM	MQAM
	5/6	5,0	MQAM	MQAM
	8/9	5,3	MQAM	MQAM
	9/10	5,4	MQAM	MQAM
8	1/2	4,0	NU-MQAM	NU-QAM
	2/3	5,3	NU-MQAM	NU-QAM
	3/4	6,0	NU-MQAM	NU-MQAM
	4/5	6,4	NU-MQAM	NU-MQAM
	5/6	6,7	MQAM	MQAM
	8/9	7,1	MQAM	MQAM
	9/10	7,2	MQAM	MQAM
10	1/2	5,0	NU-MQAM	NU-QAM
	2/3	6,7	NU-MQAM	NU-QAM
	3/4	7,5	NU-MQAM	NU-MQAM
	4/5	8,0	NU-MQAM	NU-MQAM
	5/6	8,3	NU-MQAM	NU-MQAM
	8/9	8,9	MQAM	MQAM
	9/10	9,0	MQAM	MQAM
12	1/2	6,0	NU-MQAM	NU-QAM
	2/3	8,0	NU-MQAM	NU-QAM
	3/4	9,0	NU-MQAM	NU-MQAM
	4/5	9,6	NU-MQAM	NU-MQAM
	5/6	10,0	NU-MQAM	NU-MQAM
	8/9	10,7	MQAM	MQAM
	9/10	10,8	MQAM	MQAM

Fig. 7

			Caso 1	Caso 2	Caso 2
Capacidad (bit/s/Hz)			Modulación	Modulación	Modulación
2	1/2	3.0	QAM	QAM	QAM
	2/3	4.0	QAM	QAM	QAM
	3/4	4.5	QAM	QAM	QAM
	4/5	4.8	QAM	QAM	QAM
	5/6	5.0	QAM	QAM	QAM
	8/9	5.3	QAM	QAM	QAM
	9/10	5.4	QAM	QAM	QAM
4	1/2	3.0	QAM	QAM	QAM
	2/3	4.0	QAM	QAM	QAM
	3/4	4.5	QAM	QAM	QAM
	4/5	4.8	QAM	QAM	QAM
	5/6	5.0	QAM	QAM	QAM
	8/9	5.3	QAM	QAM	QAM
	9/10	5.4	QAM	QAM	QAM
6	1/2	3.0	QAM	QAM	QAM
	2/3	4.0	QAM	QAM	QAM
	3/4	4.5	QAM	QAM	QAM
	4/5	4.8	QAM	QAM	QAM
	5/6	5.0	QAM	QAM	QAM
	8/9	5.3	QAM	QAM	QAM
	9/10	5.4	QAM	QAM	QAM
8	1/2	4.0	QAM	QAM	QAM
	2/3	5.3	QAM	QAM	QAM
	3/4	6.0	QAM	QAM	QAM
	4/5	6.4	QAM	QAM	QAM
	5/6	6.7	QAM	QAM	QAM
	8/9	7.1	QAM	QAM	QAM
	9/10	7.2	QAM	QAM	QAM
10	1/2	5.0	NU-MQAM	NU-QAM	MQAM
	2/3	6.7	NU-MQAM	NU-QAM	MQAM
	3/4	7.5	NU-MQAM	NU-MQAM	MQAM
	4/5	8.0	NU-MQAM	NU-MQAM	MQAM
	5/6	8.3	NU-MQAM	NU-MQAM	MQAM
	8/9	8.9	MQAM	MQAM	MQAM
	9/10	9.0	MQAM	MQAM	MQAM
12	1/2	6.0	NU-MQAM	NU-QAM	MQAM
	2/3	8.0	NU-MQAM	NU-QAM	MQAM
	3/4	9.0	NU-MQAM	NU-MQAM	MQAM
	4/5	9.6	NU-MQAM	NU-MQAM	MQAM
	5/6	10.0	NU-MQAM	NU-MQAM	MQAM
	8/9	10,7	MQAM	MQAM	MQAM
	9/10	10.8	MQAM	MQAM	MQAM

Fig.8

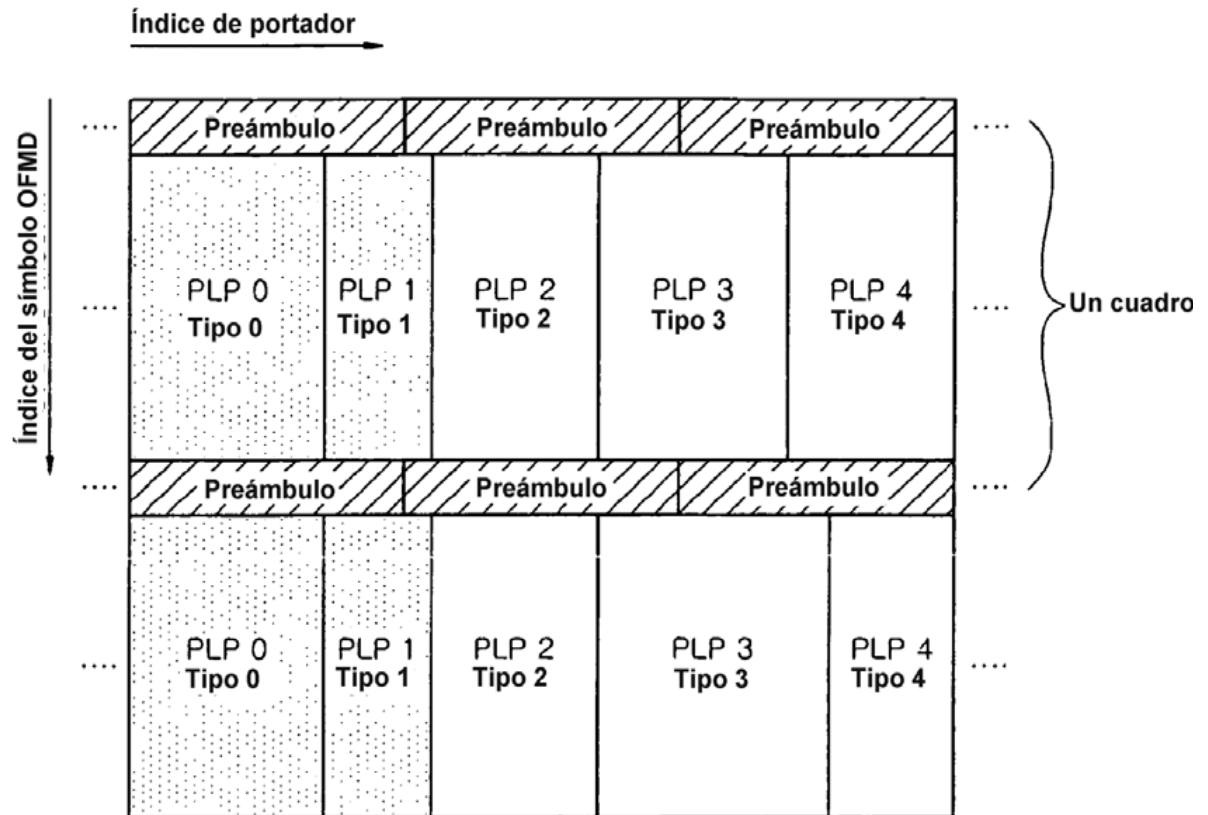


Fig. 9

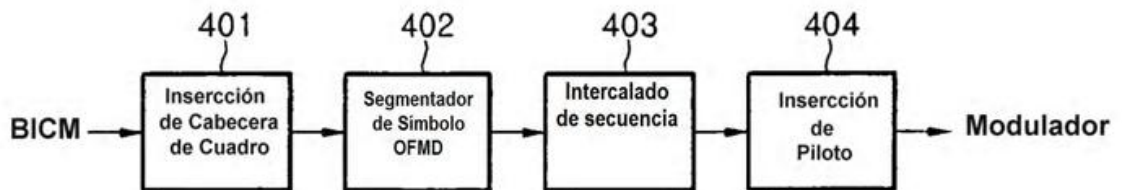


Fig. 10

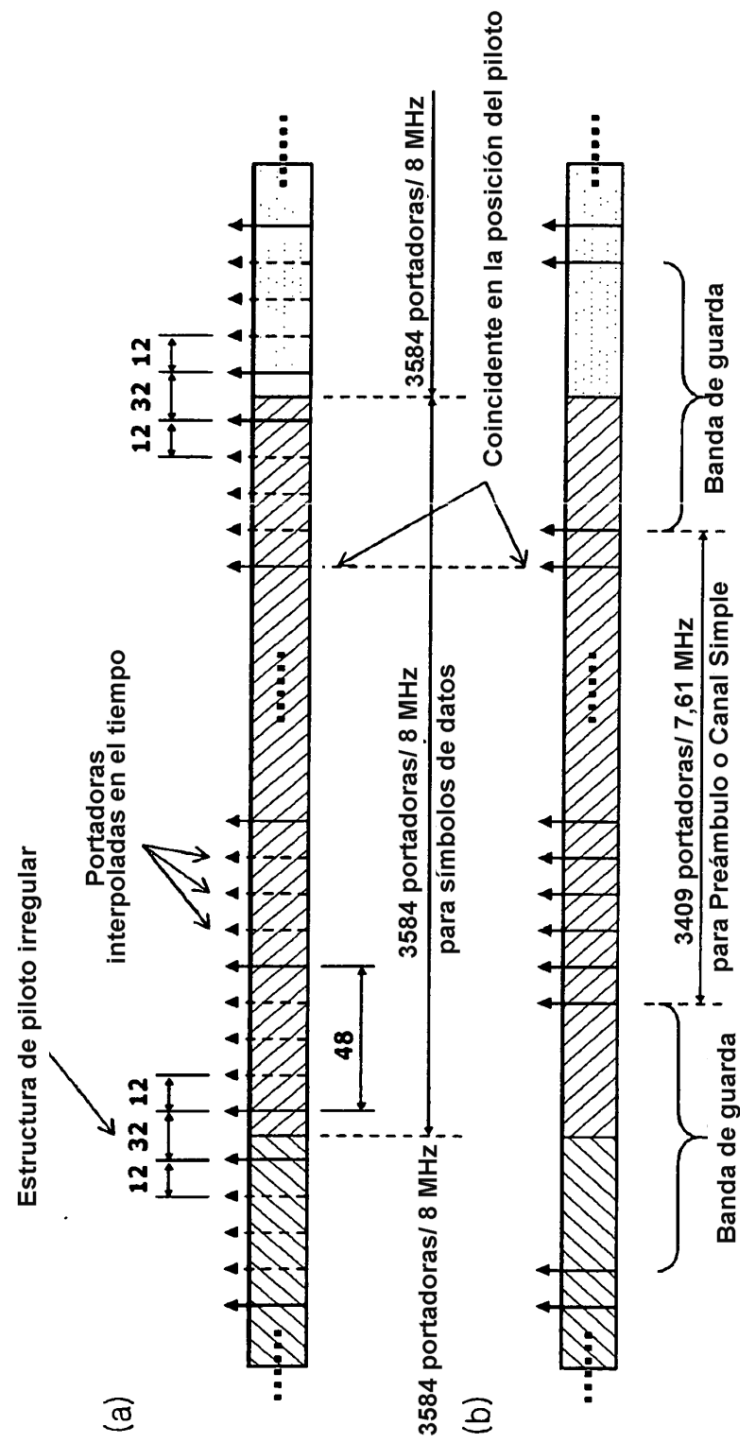


Fig. 11

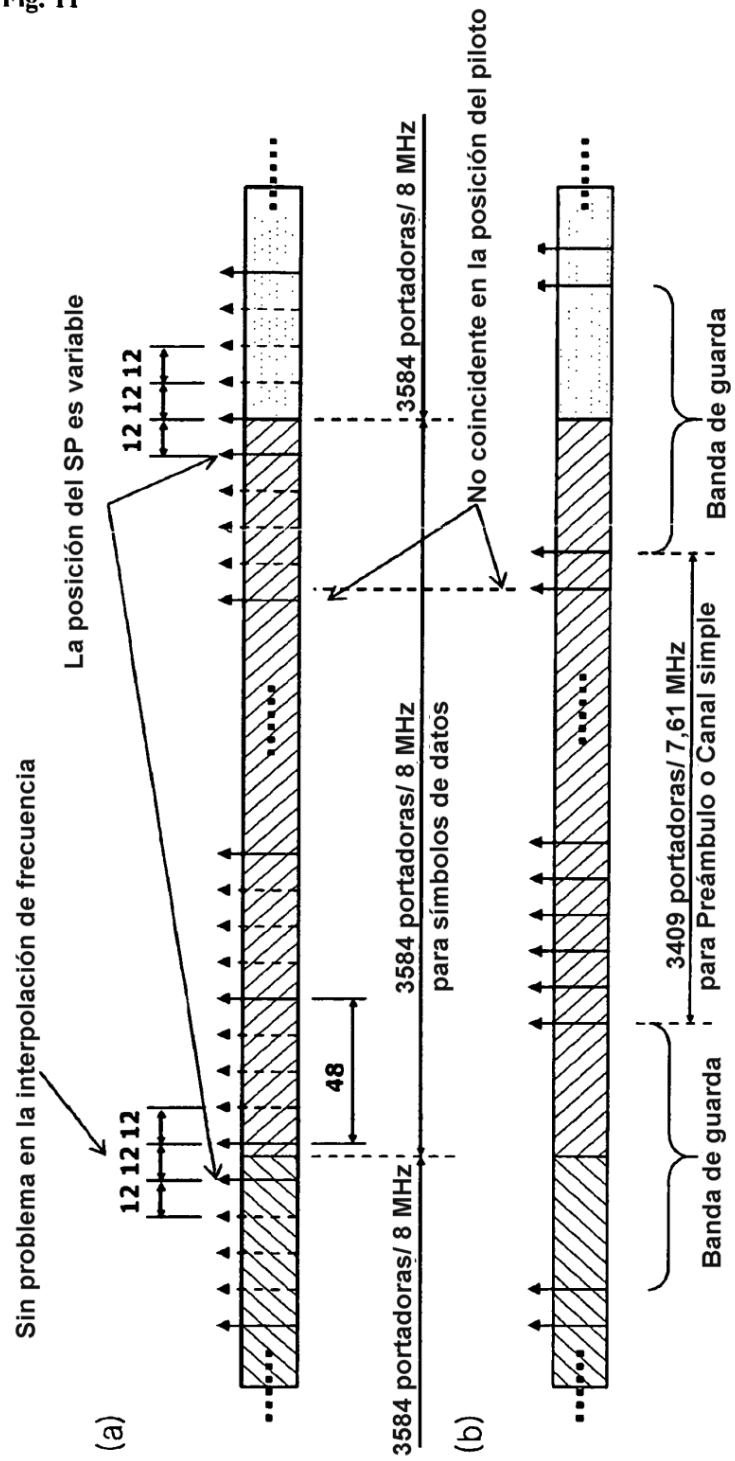


Fig. 12

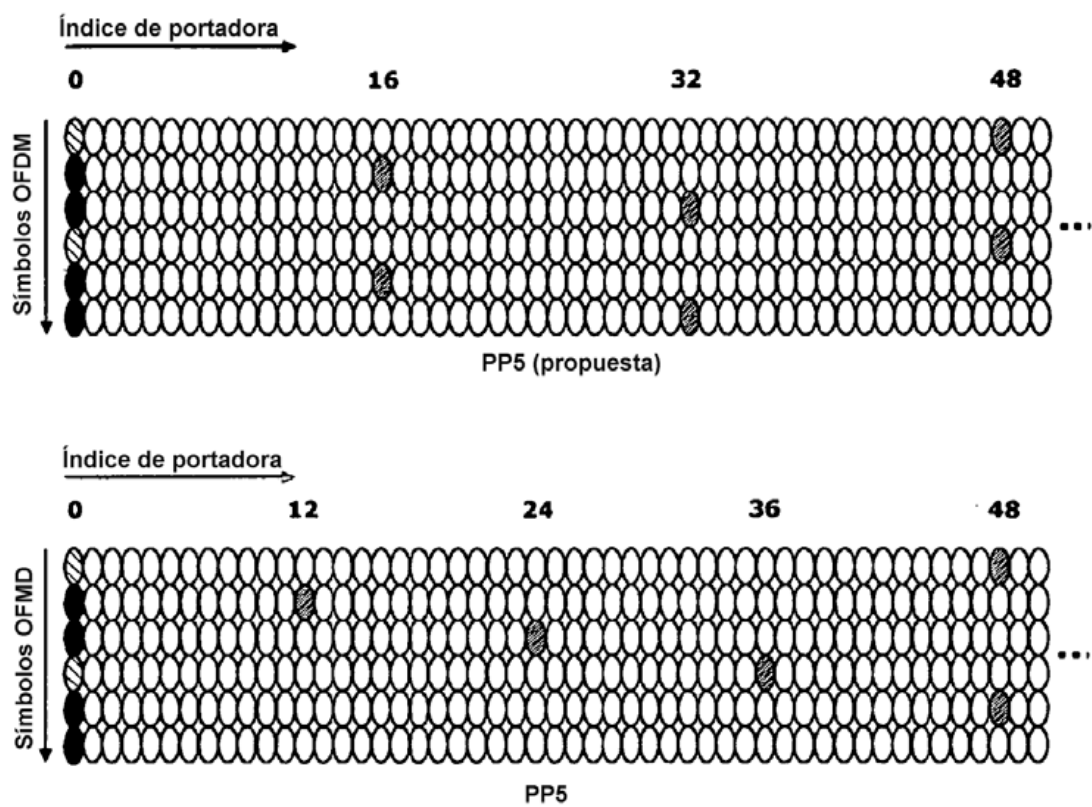


Fig. 13

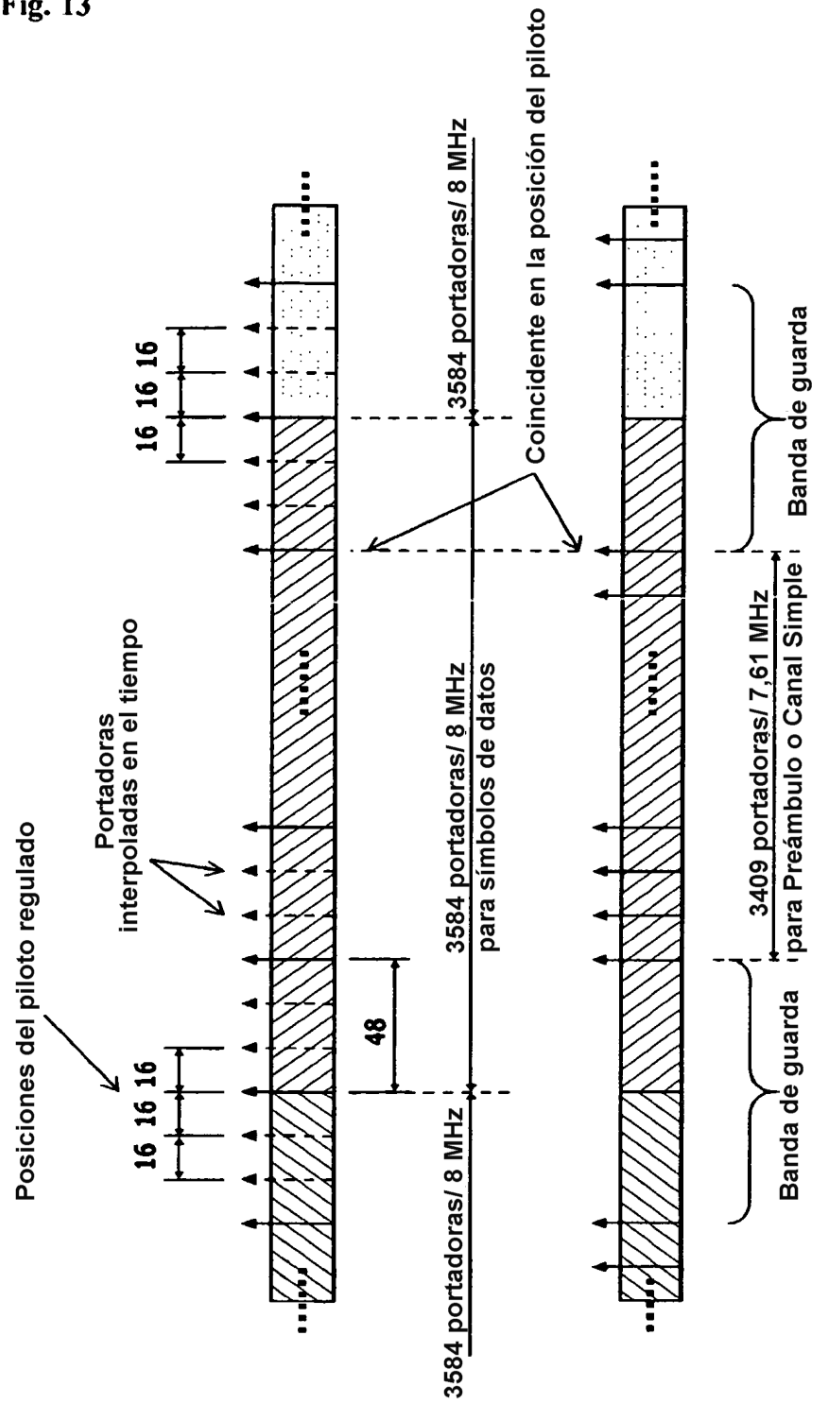


Fig. 14

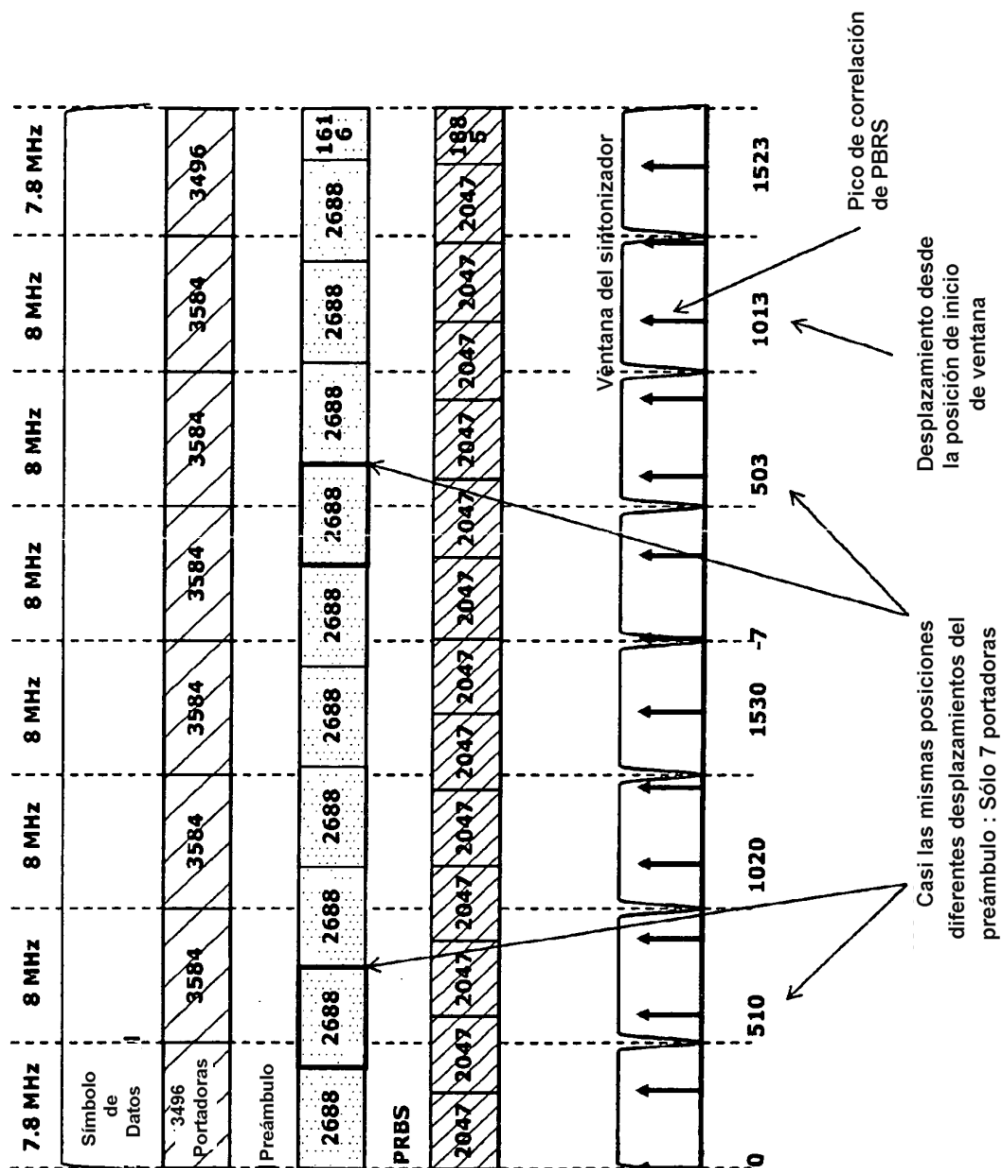


Fig. 15

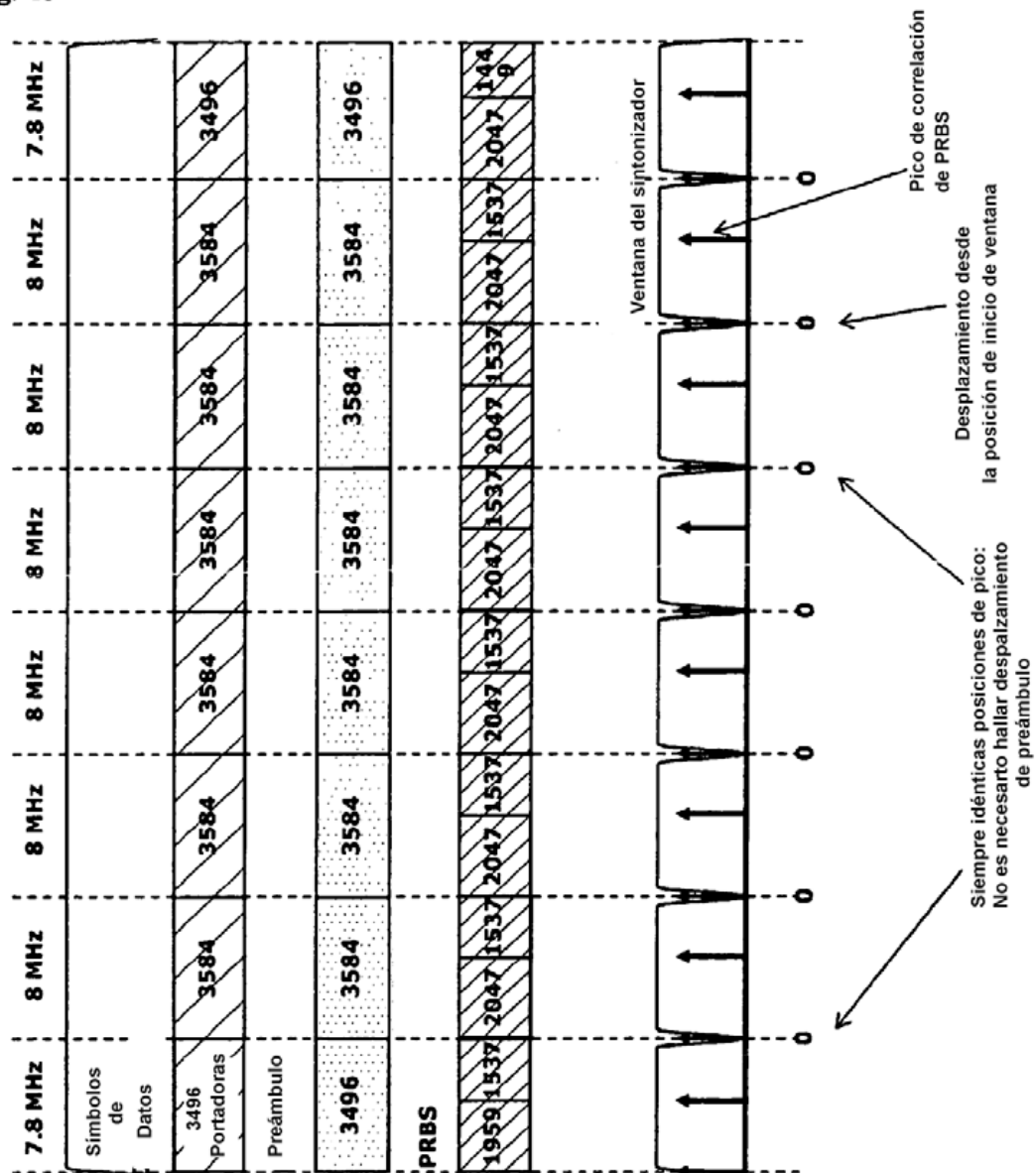


Fig. 16

Retardo (μs)	0.2	0.4	0.8	1.2	2.5	15
Aten. (dB)	-11	-14	-17	-23	-32	-40

**Recorrido de
retardo despreciable**

Fig. 17

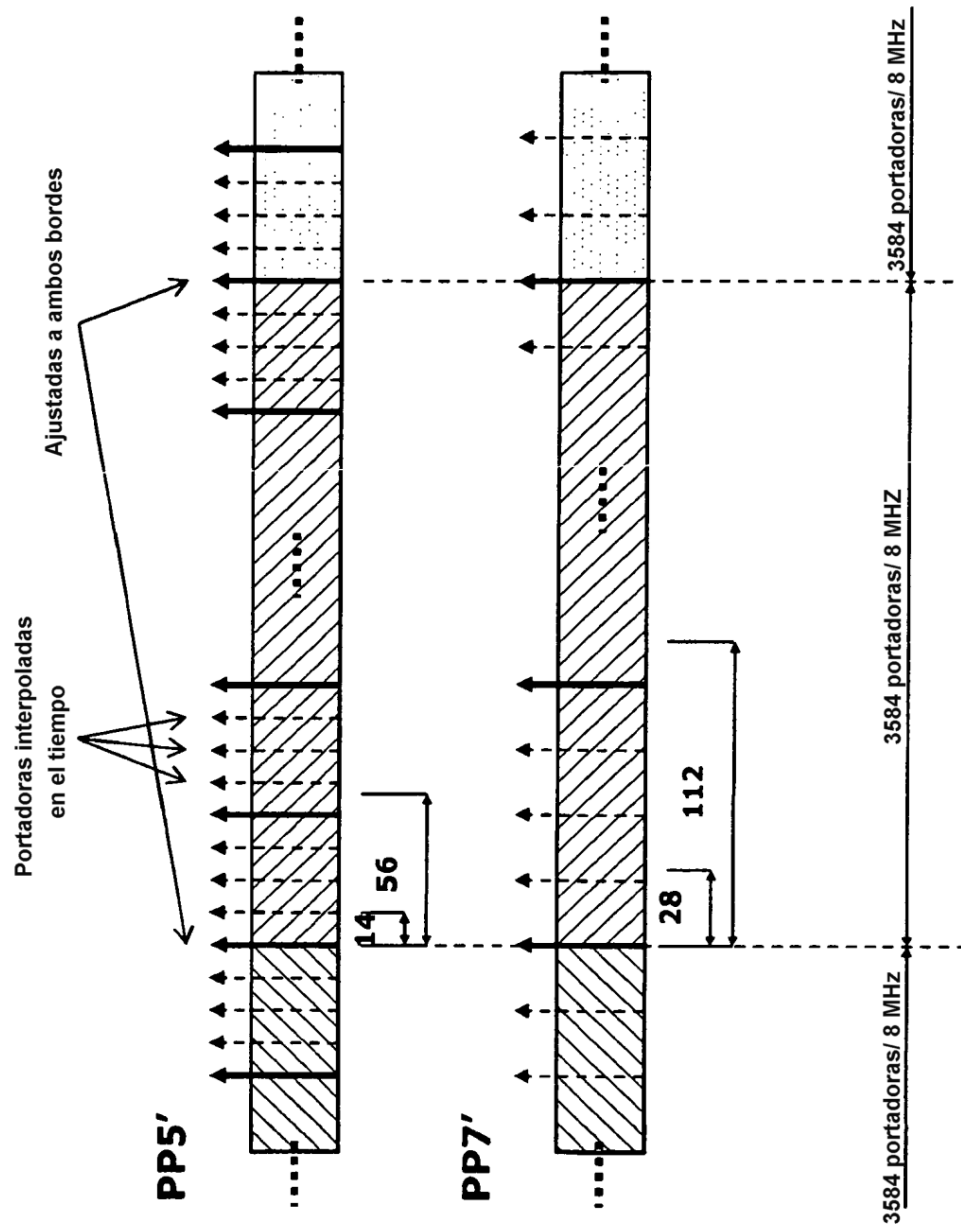


Fig. 18

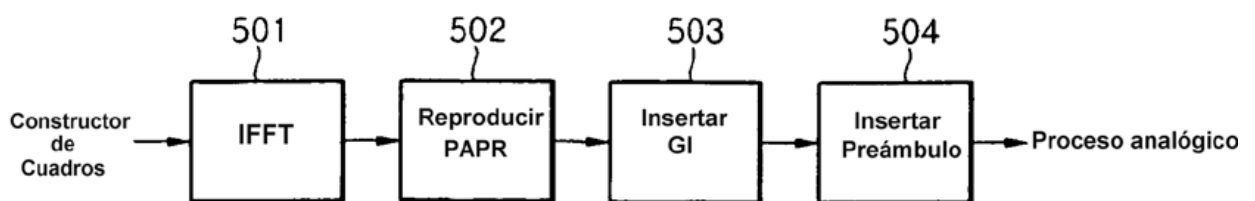


Fig. 19

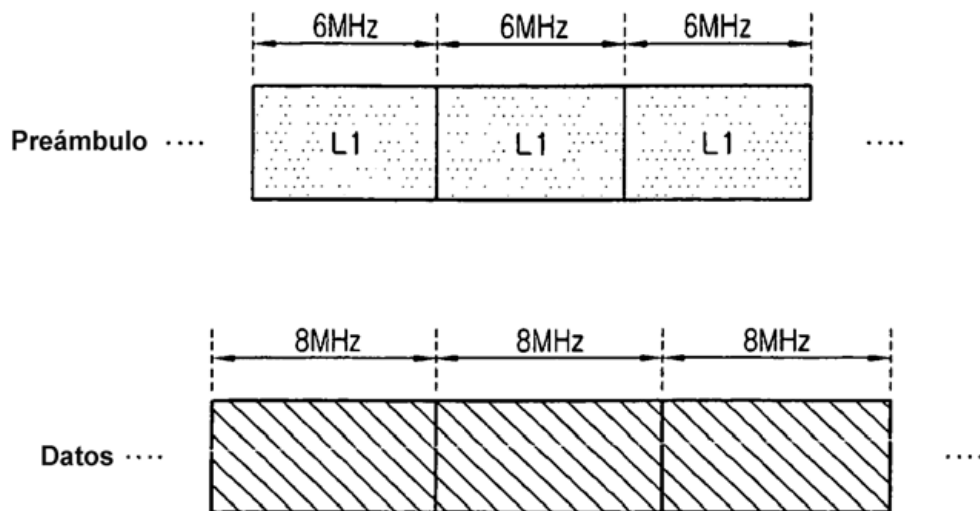


Fig. 20

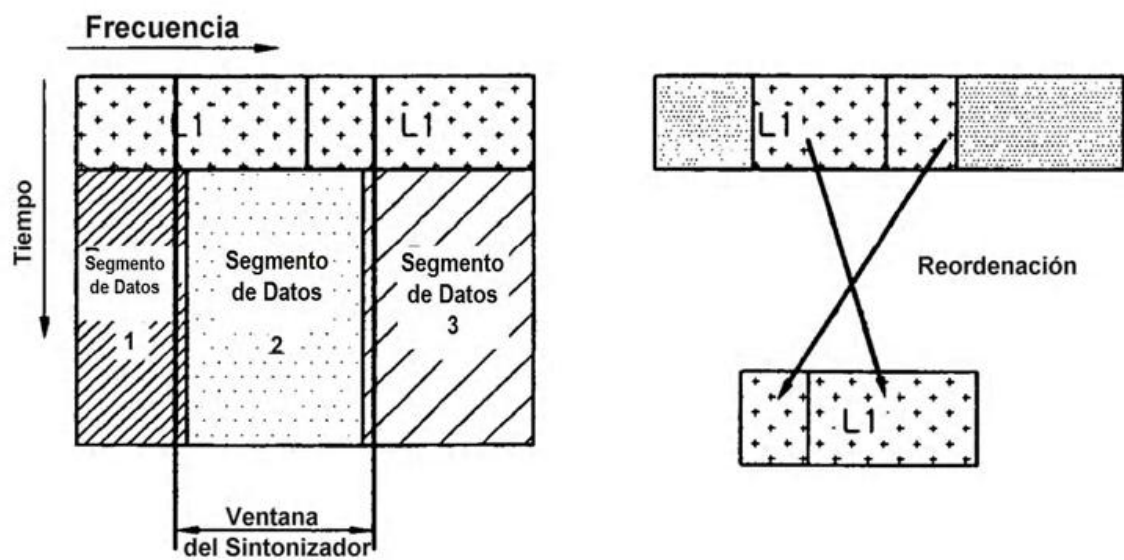


Fig. 21

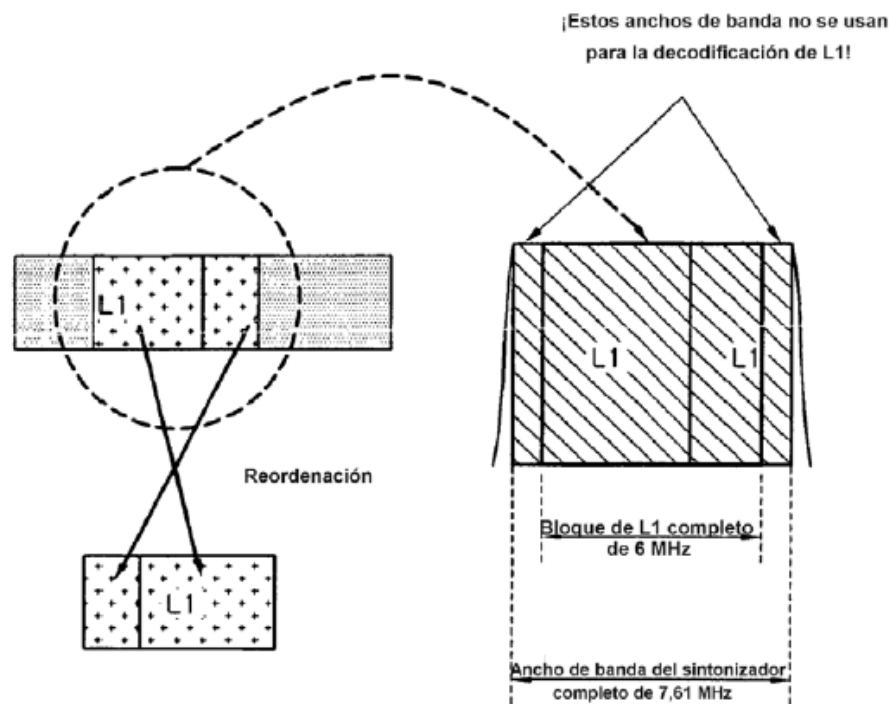


Fig. 22

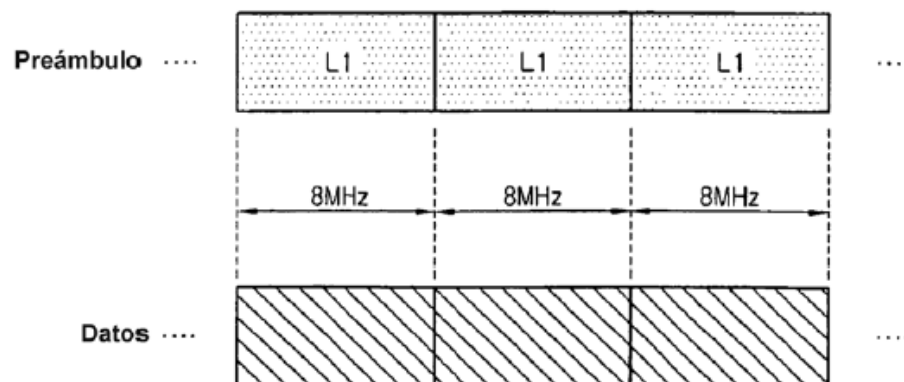


Fig. 23

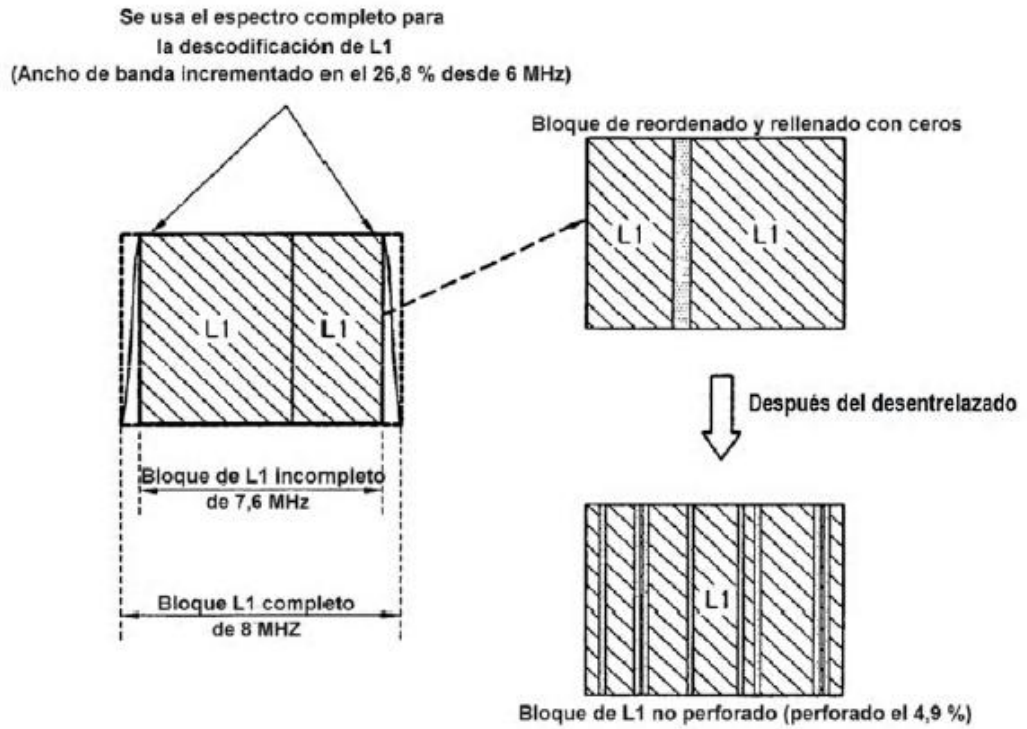


Fig. 24

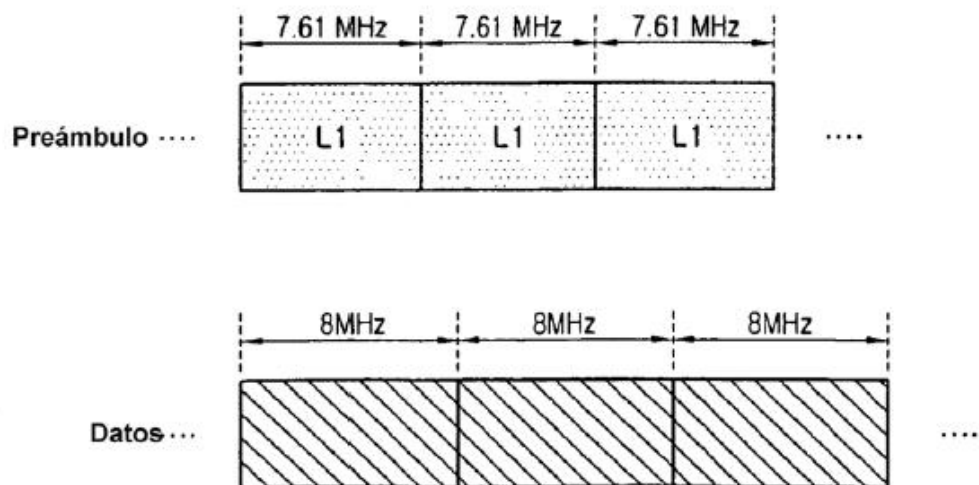


Fig. 25

Se usa el espectro completo para
la decodificación de L1
(Ancho de banda incrementado en el 28,6 % desde 6 MHz)

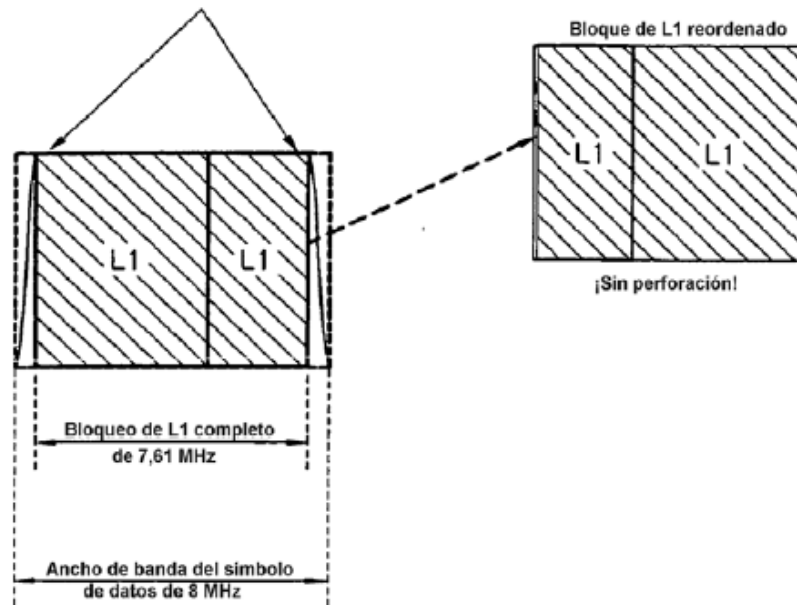


Fig. 26

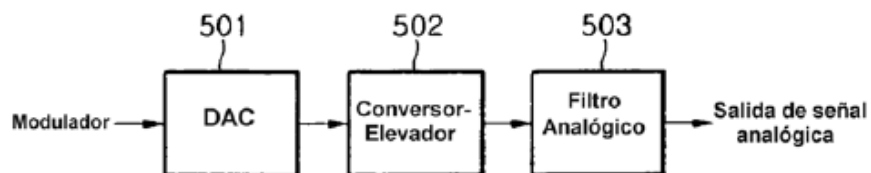


Fig. 27



Fig. 28

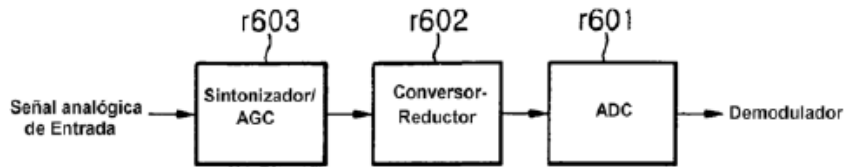


Fig. 29

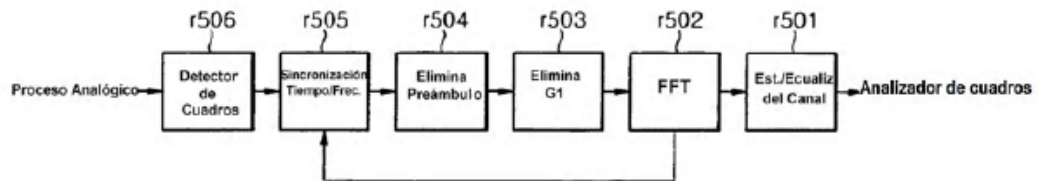


Fig. 30

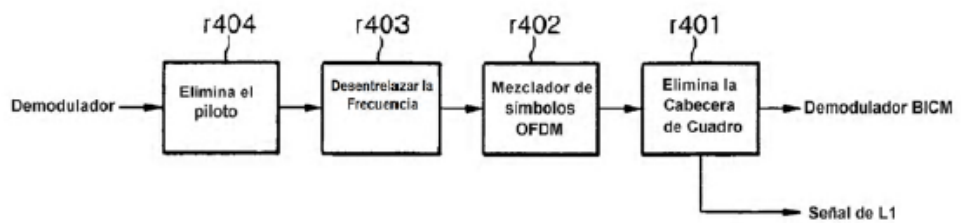


Fig. 31

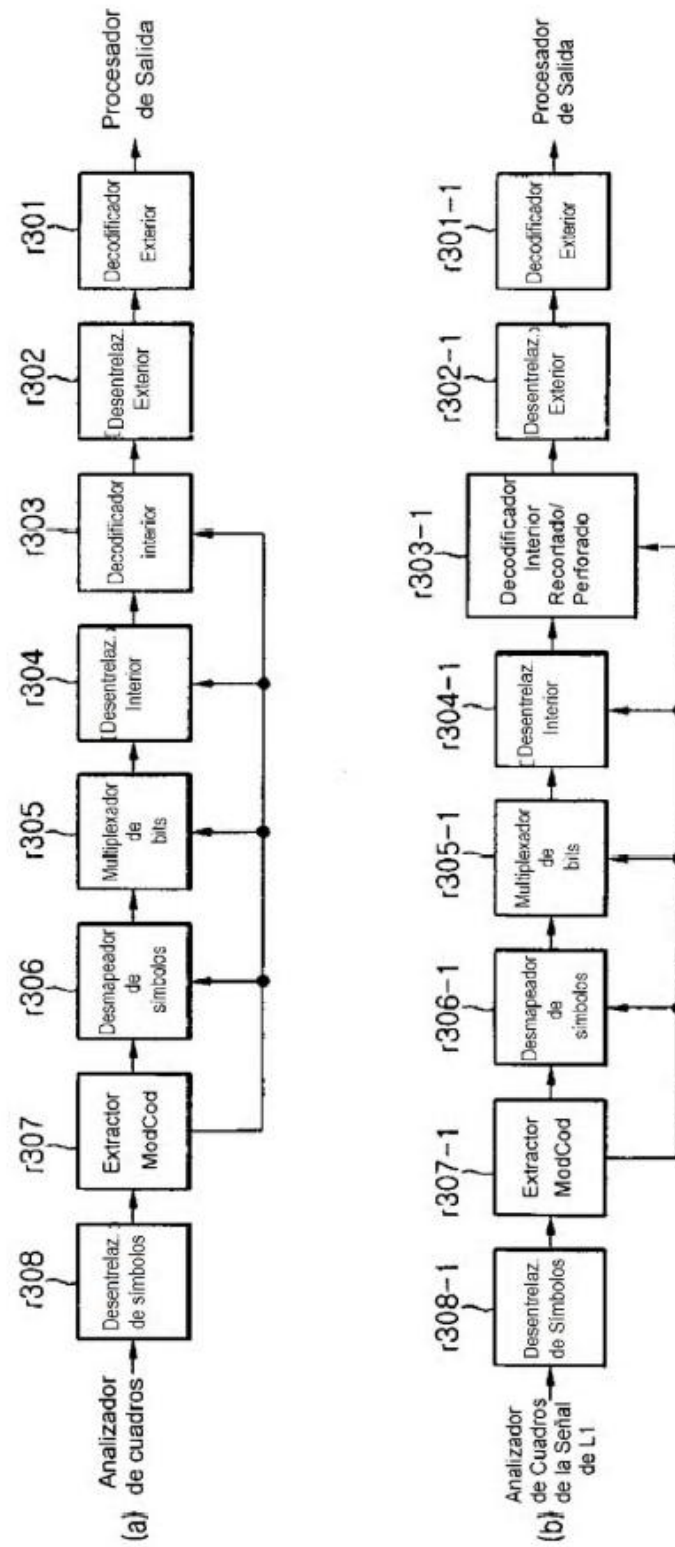


Fig. 32

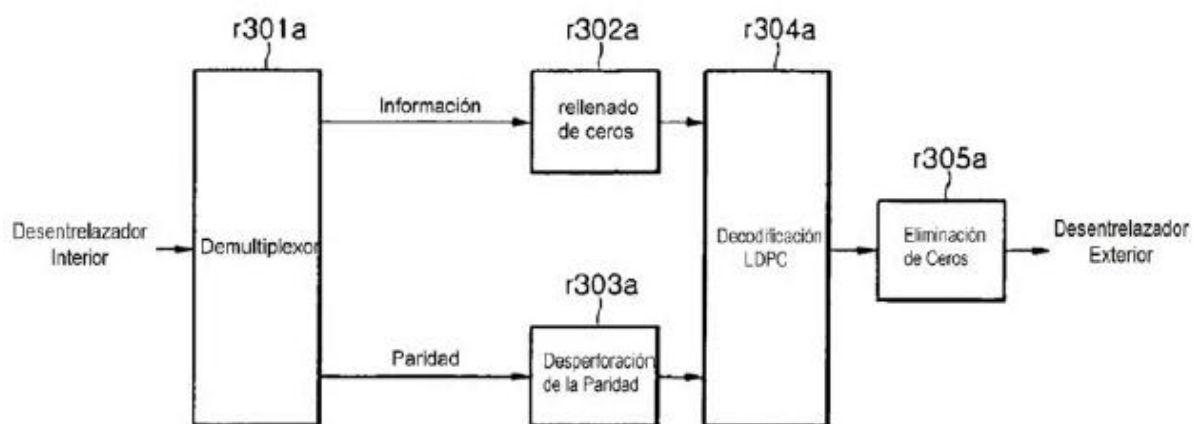


Fig. 33

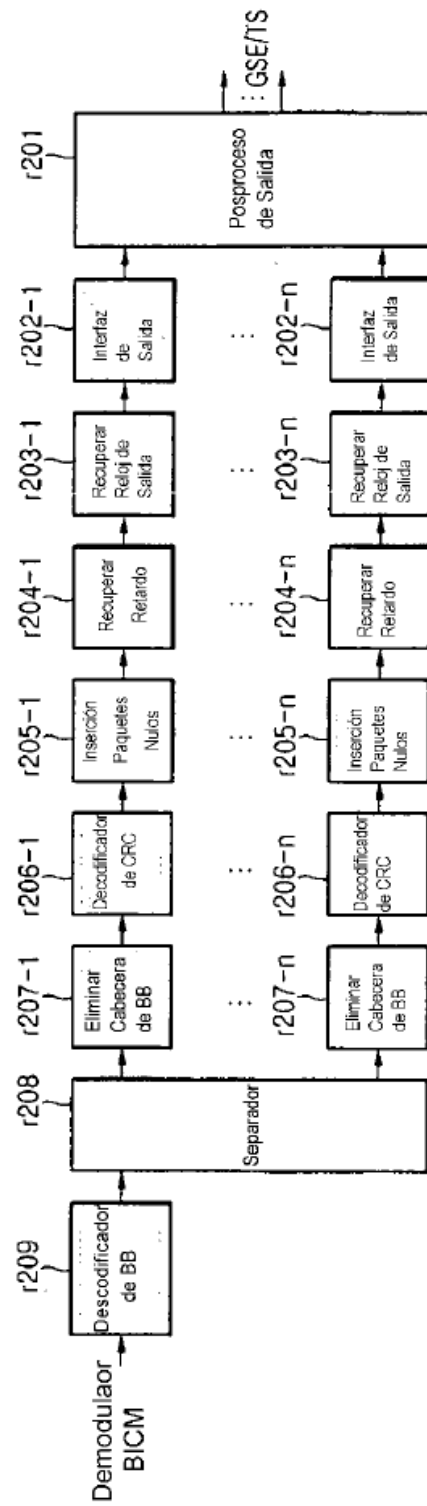


Fig. 34

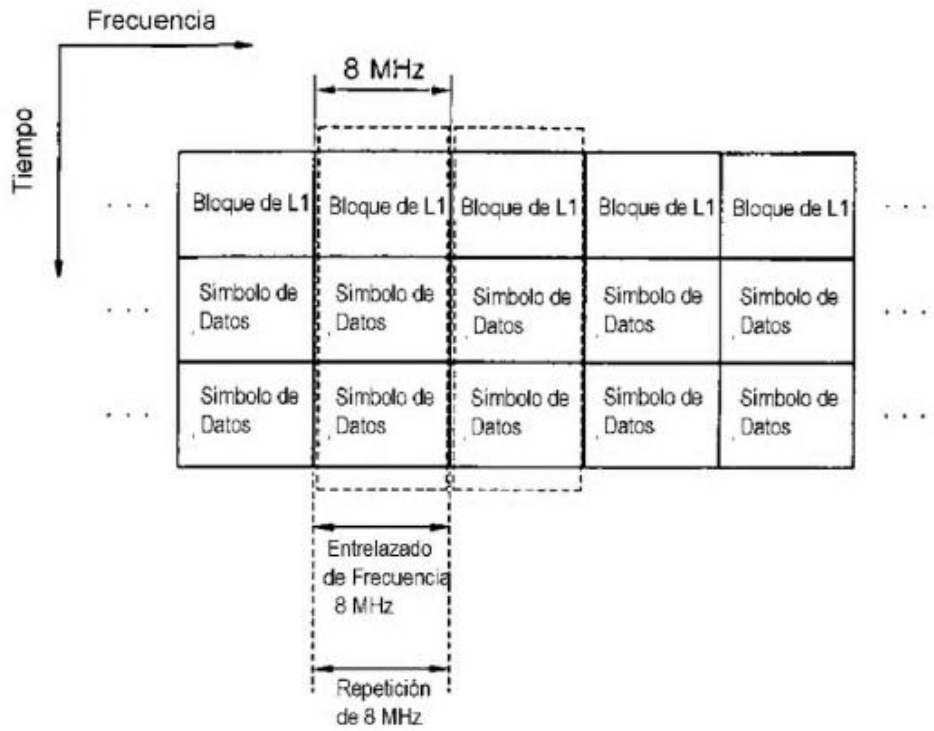


Fig. 35

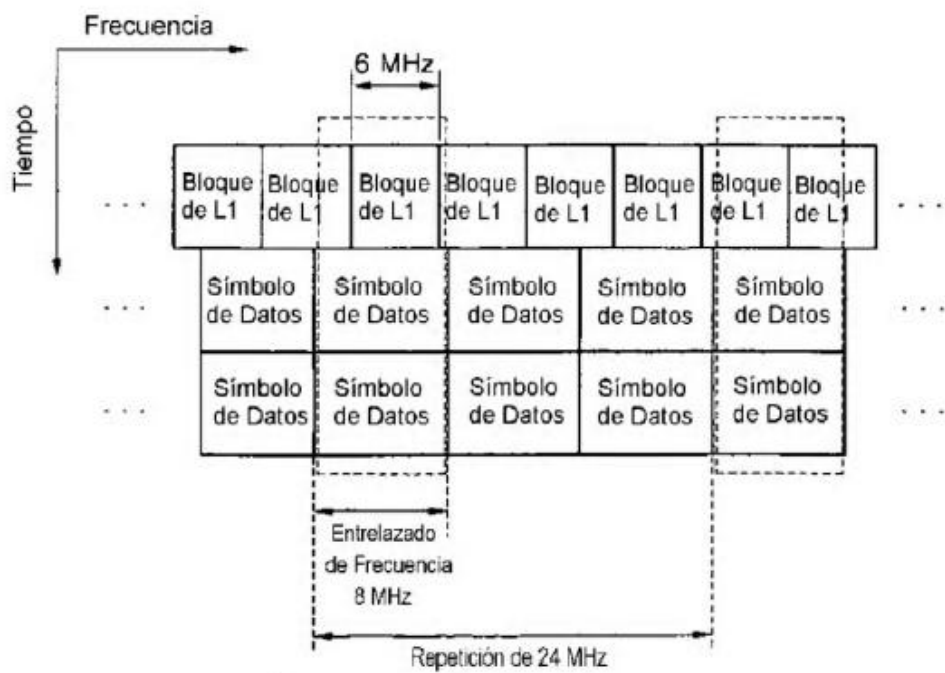


Fig. 36

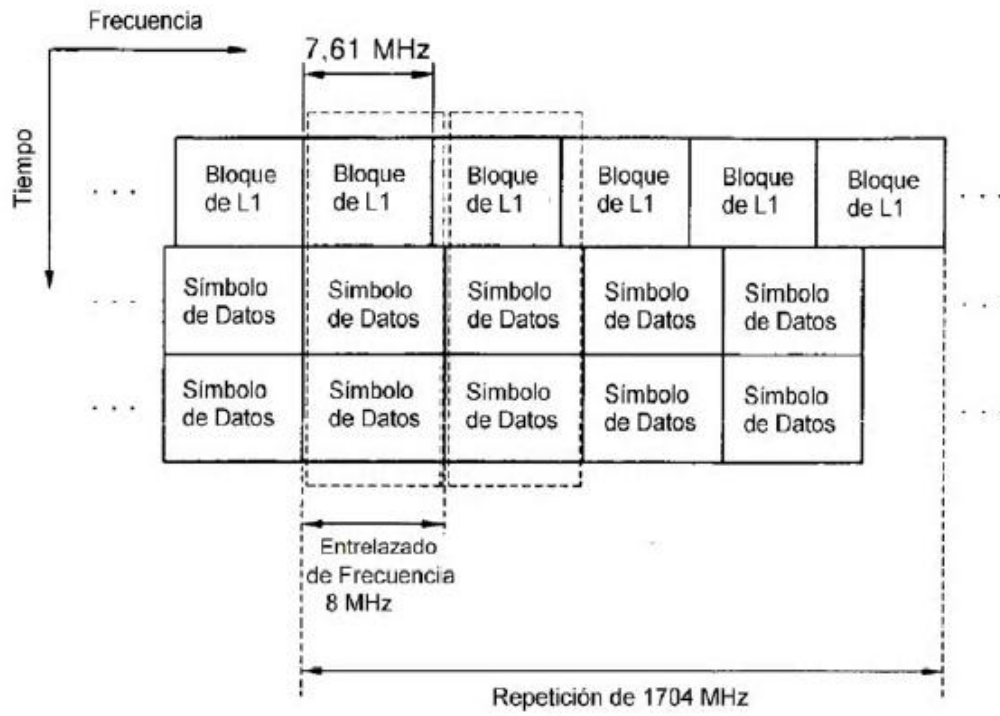


Fig. 37

Campo	Bits	
L1_span	12	número de portadoras extendidas por el bloque de L1 dentro de un símbolo OFMD (Máximo = 7,61 MHz)
num.chbon	3	número de canales adheridos
num.dslice	8	número de ranura de datos
num_plp	8	número de PLP
num_notch	5	número de banda de muescas
for dslice {		
chbon_index	3	índice del canal adherido
dslice_start	9	Comienzo del segmento de datos dentro de un canal (8 MHz)
dslice_width	9	Ancho del segmento de datos
}		
for pip {		
dslicejd	8	ID del segmento de datos
pip_id	8	ID de PLP
pip_type	1	Tipo de PLP (común/datos)
plp_payload_type	5	tipo de contenido útil de la PLP (TS, GS,...)
}		
for notch {		
chbon_index	3	índice de canal adherido
notch_start	9	comienzo de la banda de muesca dentro de un canal (8 MHz)
notch_width	9	Ancho de la banda de la supresión
}		
gi	1	modo de intervalo de guarda
sframe id	16	ID del duper cuadro
frame id	16	ID del cuadro
reserved	0	rfu
crc32	32	CRC32
Total	11760	

El número de bits de información L1 varía de acuerdo con varias configuraciones /condiciones

Fig. 38

Información de L1 (bits)	11760
Bloques de L1 (bits)	23520
Símbolos de 16-QAM	5880
Portadoras totales	3408
Distancia de SP del preámbulo	6
Portadoras de datos	2840
LDPC corto	1.45
Símbolos OFDM	2.07

↙
Tamaño máximo

Fig. 39

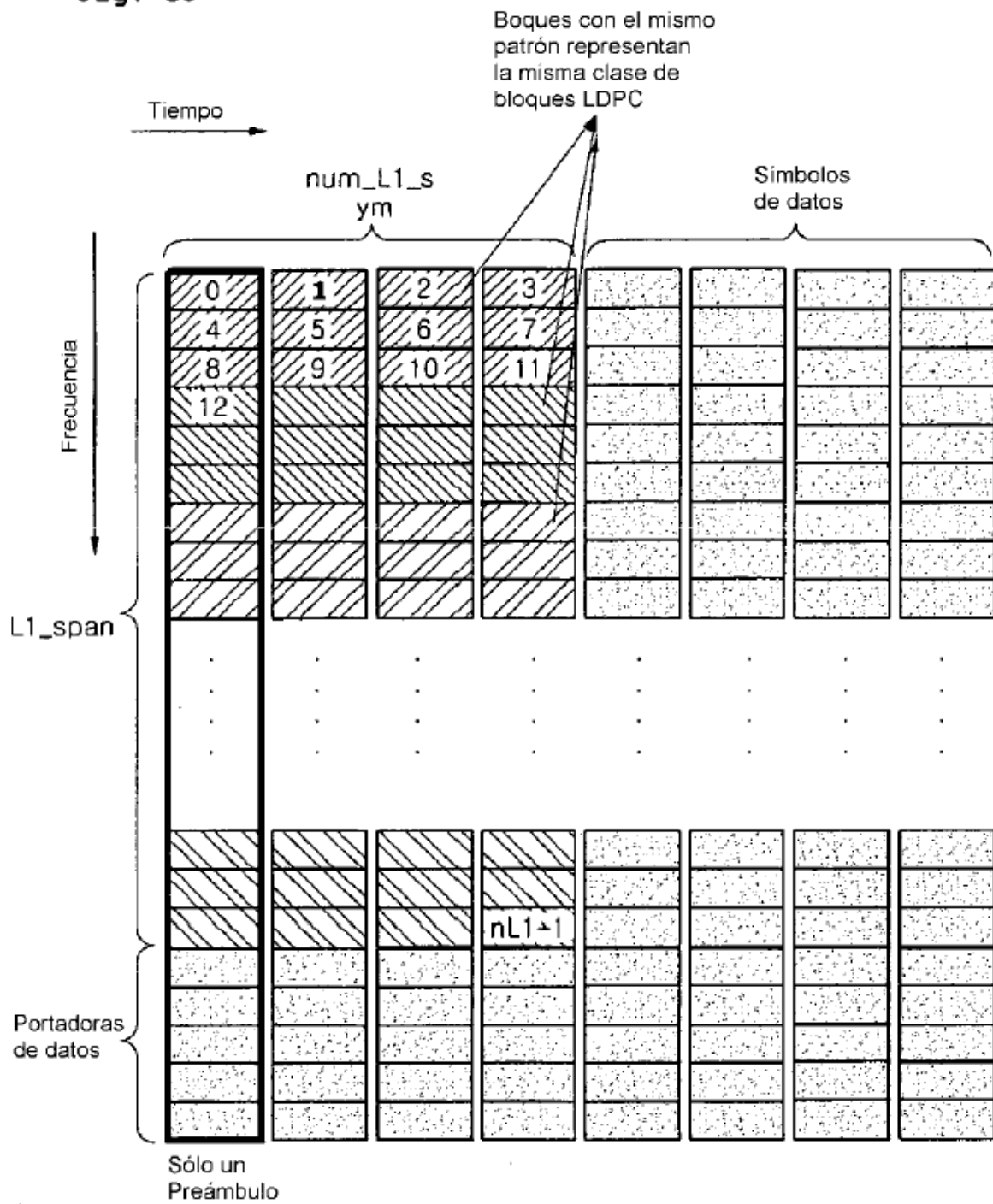


Fig. 40

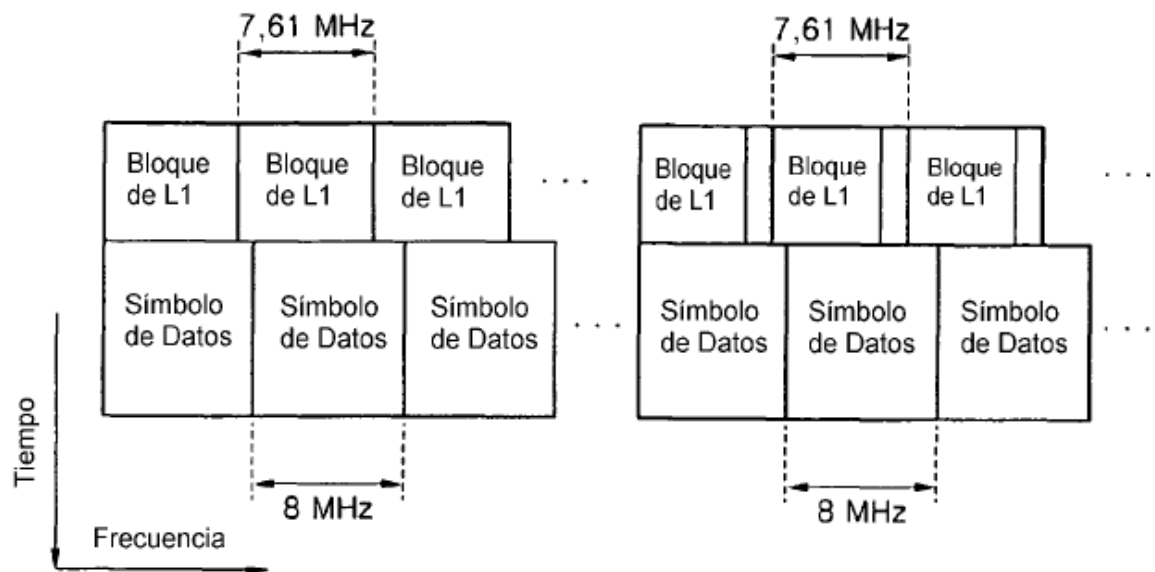


Fig. 41

Campo	Bits	
L_column	9	número de portadoras extendidas por el bloque de L1 dentro de un símbolo OFDM (Máximo = 7,61 MHz)
L_row	3	número de símbolos OFDM expandidos por el bloque L1
num_chbon	3	número de canales adheridos
num_plp	8	número de ranuras de datos
num_dslice	8	número de PLP
num_notch	5	número de bandas de supresión
for dslice {		
chbonindex	3	índice del canal adherido
dslice.start	9	comienzo del segmento de datos dentro de un canal (8 MHz)
dslice_width	9	ancho de la ranura de datos
}		
for plp {		
dslice_id	8	ID del segmento de datos
plp_id	8	ID de la PLP
plp_type	1	tipo de PLP (común/datos)
plp_payload_type	5	tipo de contenido útil de PLP (TS, GS,...)
}		
for notch {		
chbon_index	3	índice del canal adherido
notch_start	9	comienzo de la banda de muestra dentro de un canal (8 MHz)
notch_width	9	modo del intervalo de guarda
}		
gi	1	modo del intervalo de guarda
sframe_id	16	ID del supercuadro
frame_id	16	ID del cuadro
reserved	16	rfu
crc32	32	CRC32
Total	11776	

El número de bits de la información de L1 varía de acuerdo con varias configuraciones/condiciones

Fig. 42

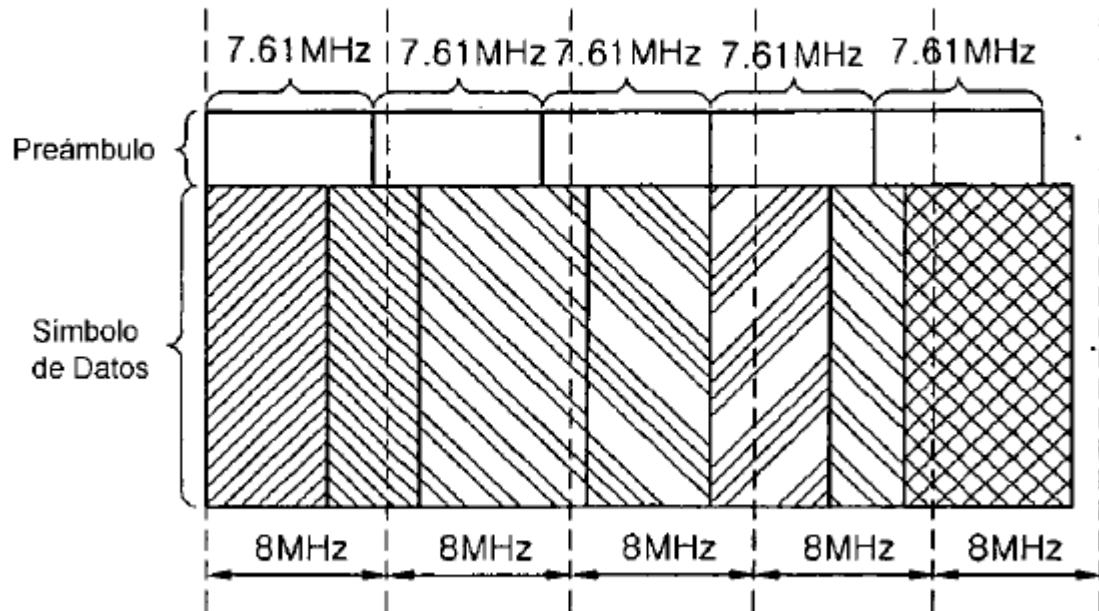


Fig. 43

Tamaño del bloque LDPC	Modulación (bps/Hz)	Símbolos QAM	Cabecera adicional ModCod
64800	4	16200	0.28%
	6	10800	0.42%
	8	8100	0.56%
	10	6480	0.69%
	12	5400	0.83%
16200	4	4050	1.11%
	6	2700	1.67%
	8	2025	2.22%
	10	1620	2.78%
	12	1350	3.33%

Fig. 44

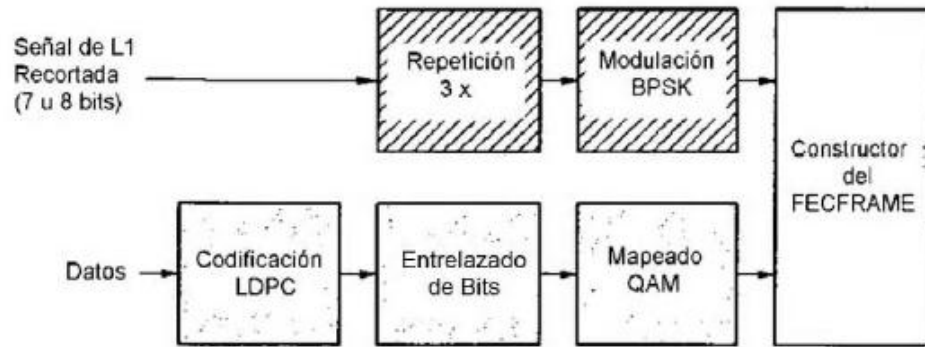


Fig. 45

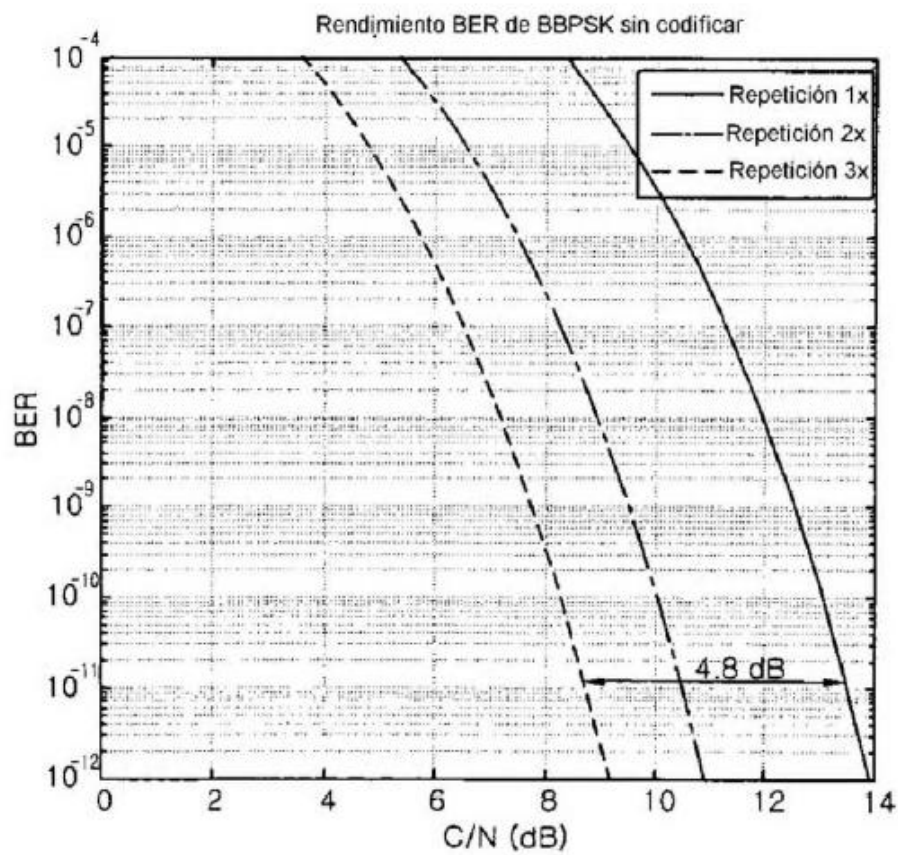


Fig. 46

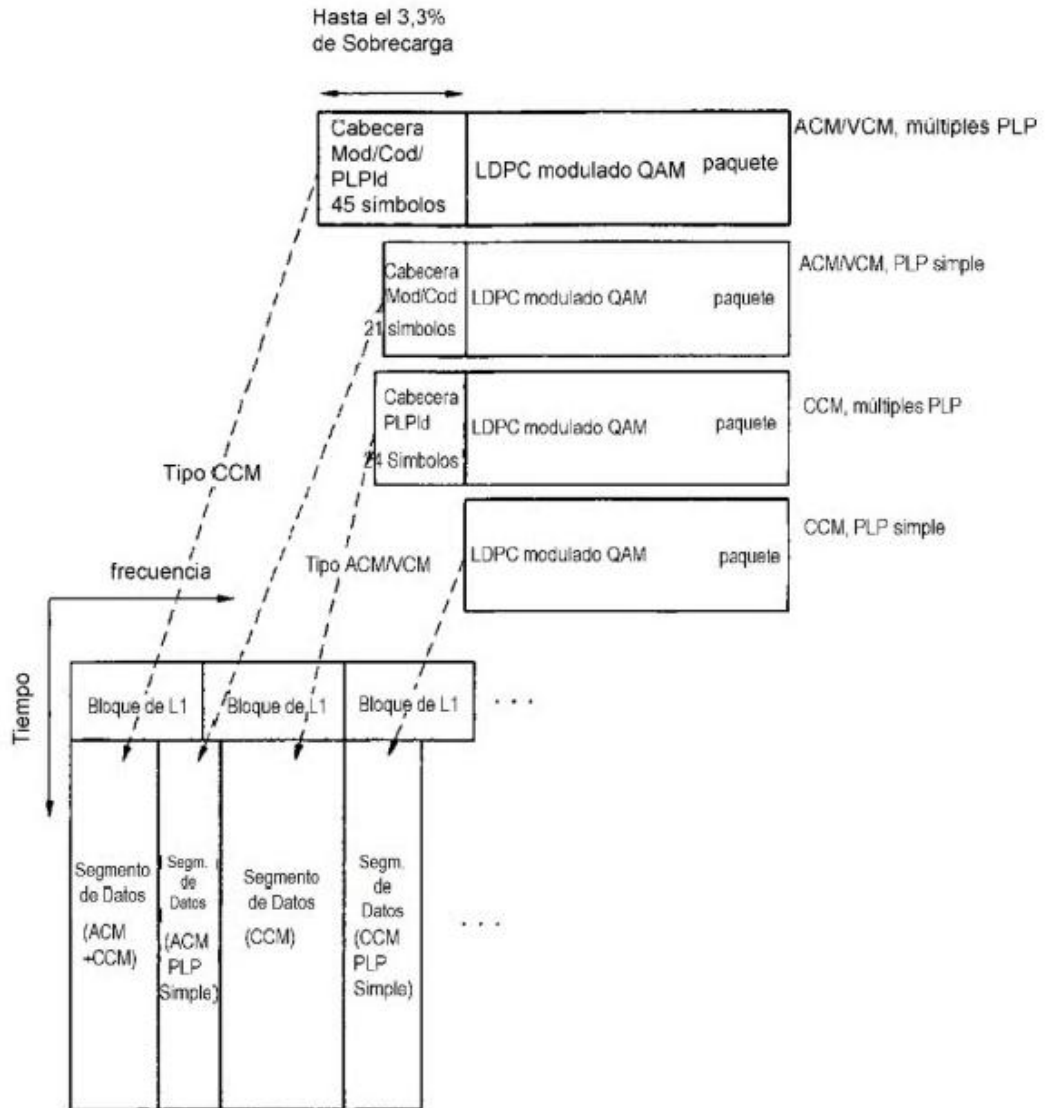


Fig. 47

Campo	Bits
L1_size	14 tamaño del bloque (bits)
num_chbon	3 número de canales adheridos
num_dslice	8 número de segmentos de datos
num_plp	8 número de PLP
num_notch	5 número de bandas de supresión
for dslice {	
chbon_index	3 índice del canal adherido
dslice_start	9 comienzo del segmento de datos dentro de un canal (8 MHz)
dslice_width	9 ancho del segmento de datos
dslice_time_intriv	1 conexión/desconexión del entrelazado de tiempos
dslice_type	1 tipo de segmento de datos (CCM, ACM/VCM)
for plp {	
dslice_id	8 ID del segmento de datos
plp_id	8 ID de la PLP
plp_type	1 tipo de PLP (común/ datos)
plp_payload_type	5 contenido útil de la PLP (TS, GS,...)
if dslice_type=CCM {	
plp_mod	3 tipo de modulación de PLP
plp_fec_type	1 tipo de FEC de la PLP (largo/corto)
plp_cod	3 tasa de códigos de la PLP
}	
for notch {	
chbon_index	3 índice de canal adherido
notch_start	9 comienzo del segmento de datos dentro de un canal (8 MHz)
notch_width	9 ancho de la banda de supresión
}	
gi	1 modo del intervalo de guarda
sframe_id	16 ID del supercuadro
frame_id	16 ID del cuadro
reserved	16 rfu
crc32	32 CRC32
Total	14082

El tamaño del bloque de L1 se transmite en el primer bloque LDCP (longitud más corta fija)

Conexión/desconexión del entrelazado de tiempos para dar soporte al modo de latencia

El tipo de segmento de datos para la reducción de la sobrecarga de señalización de L1

Los campos Mod/Cod se transmiten en el preámbulo solo para el tipo CCM

Fig. 48

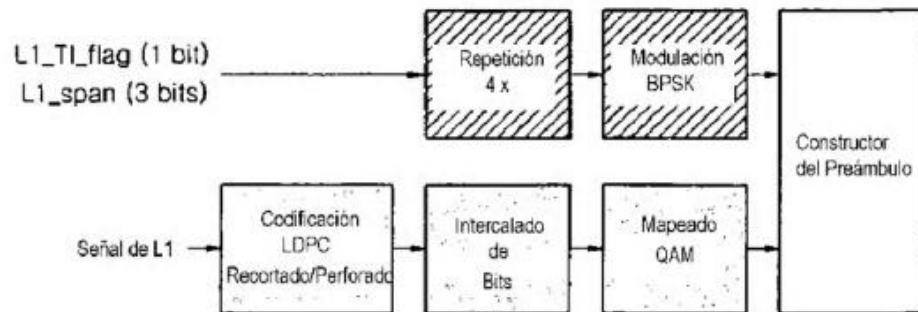


Fig. 49

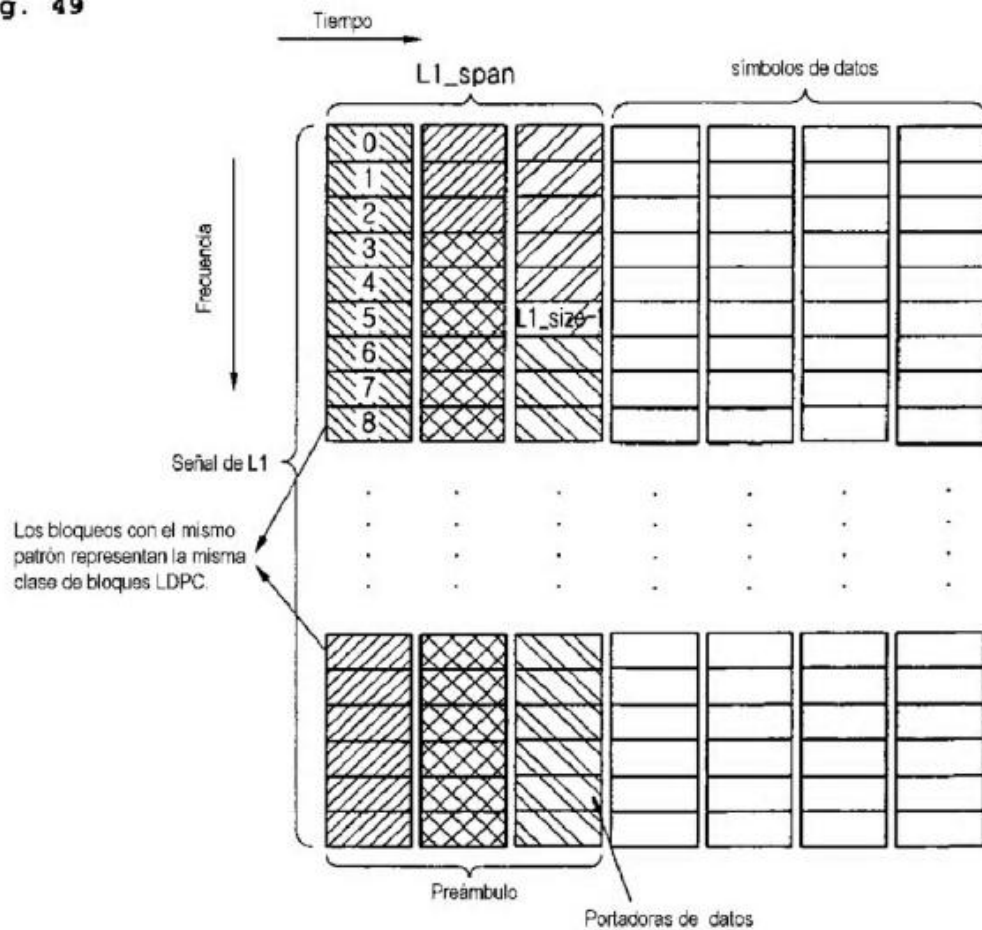


Fig. 50

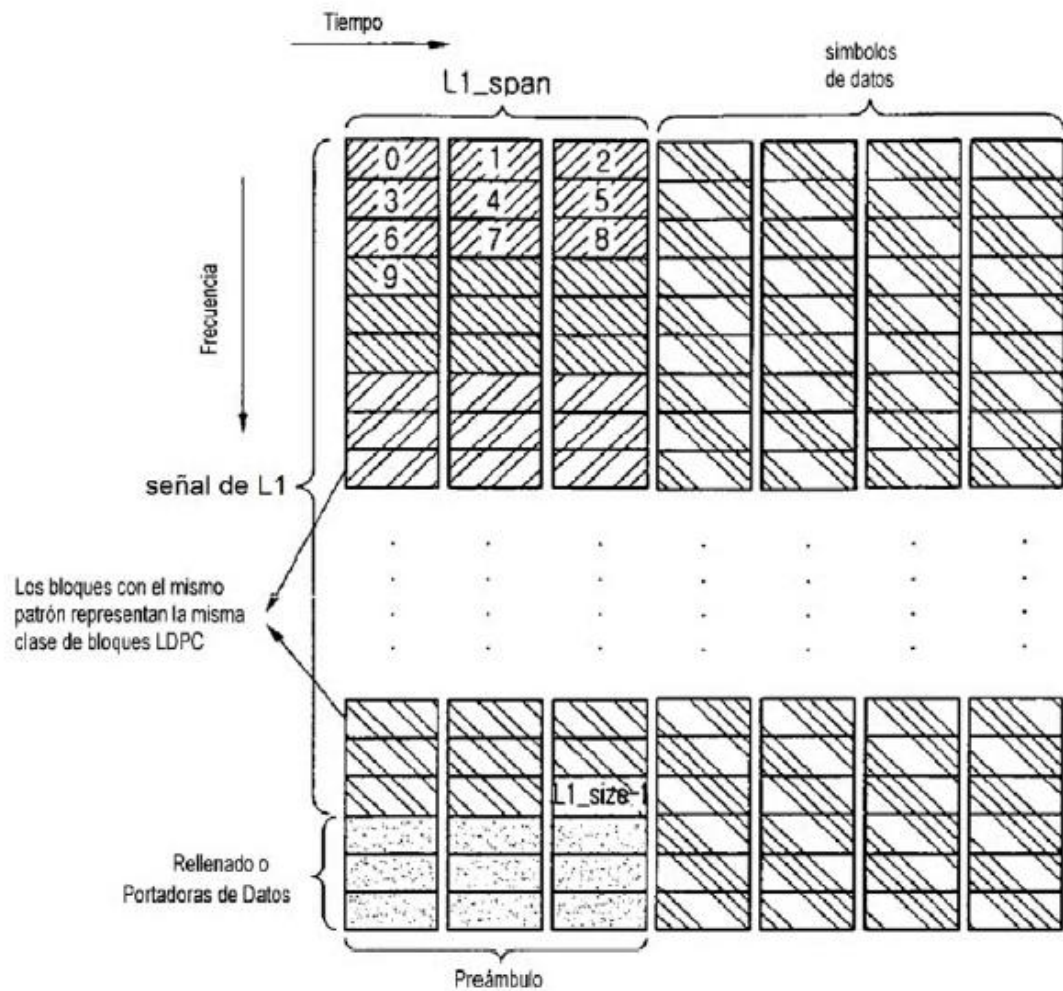


Fig. 51

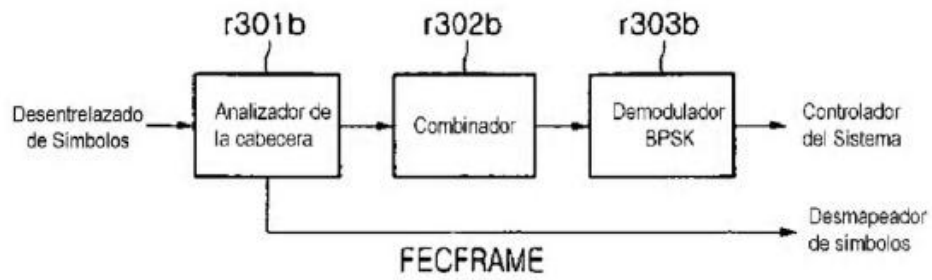


Fig. 52

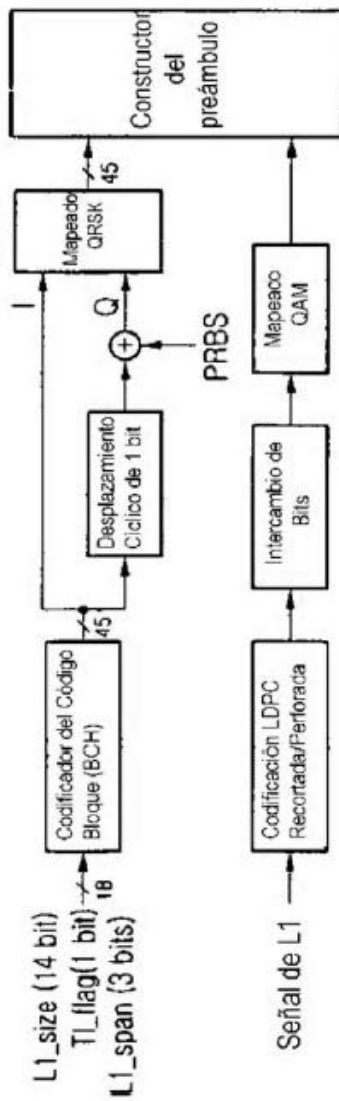


Fig. 53

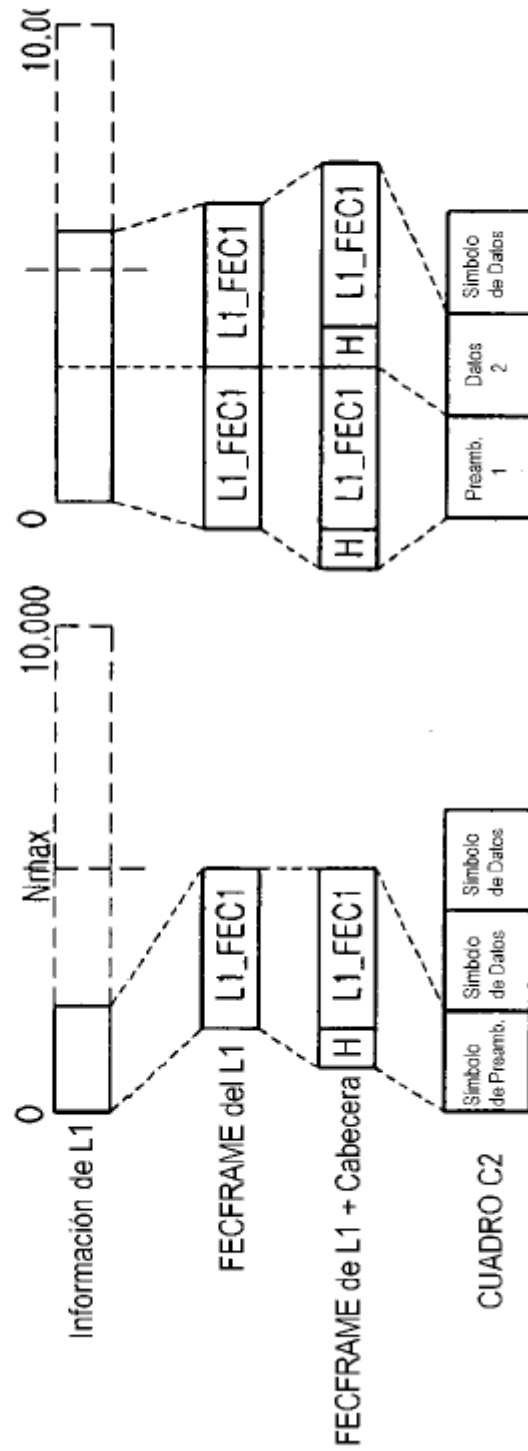


Fig. 54

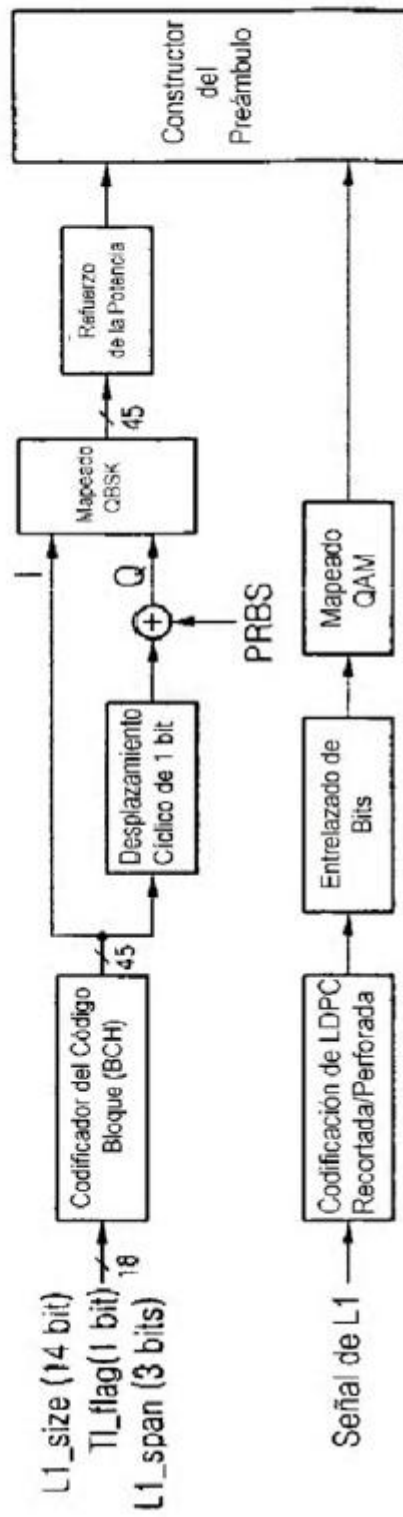


Fig. 55

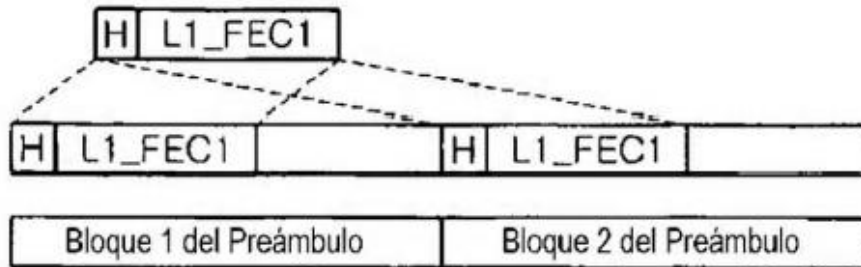


Fig. 56

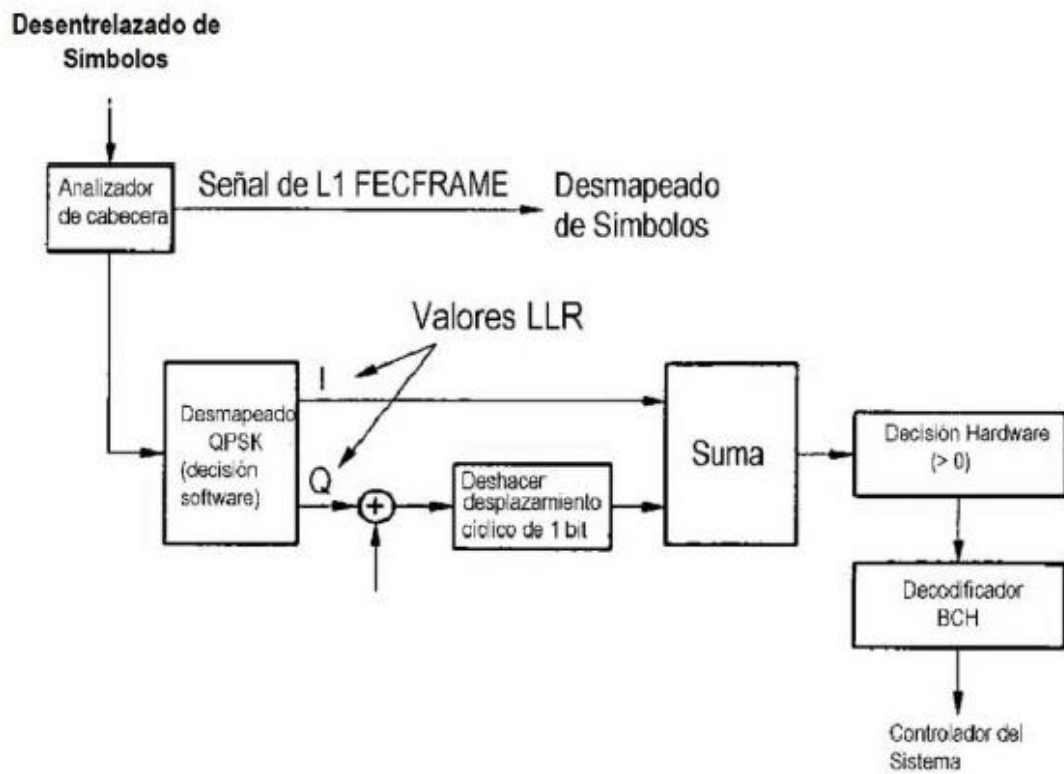


Fig. 57

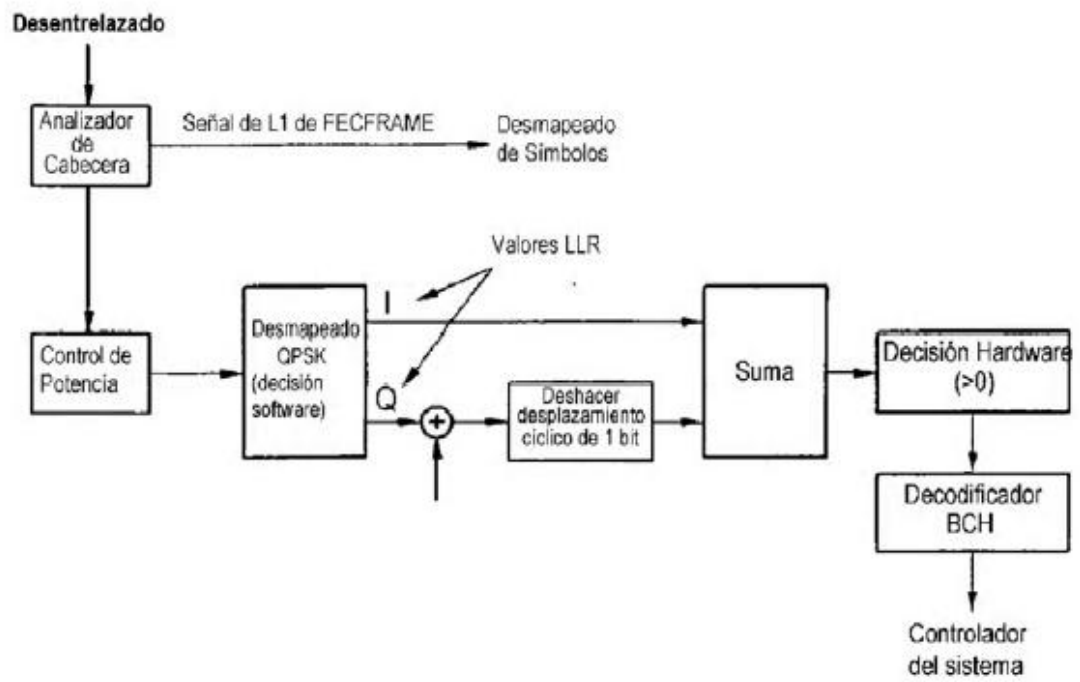


Fig. 58

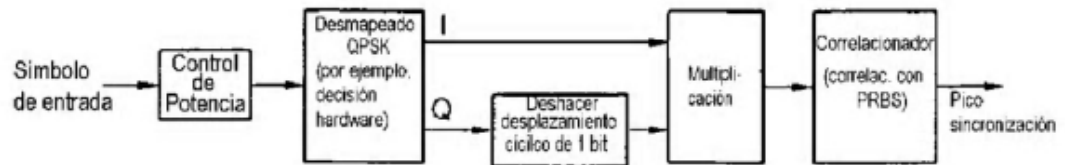


Fig. 59

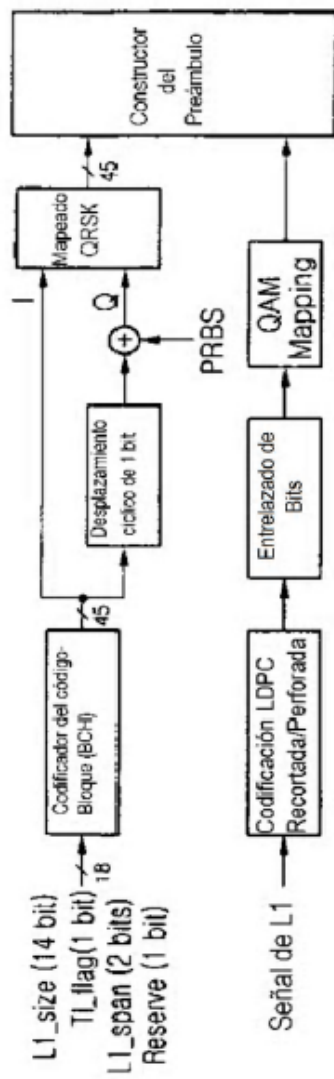


Fig. 60

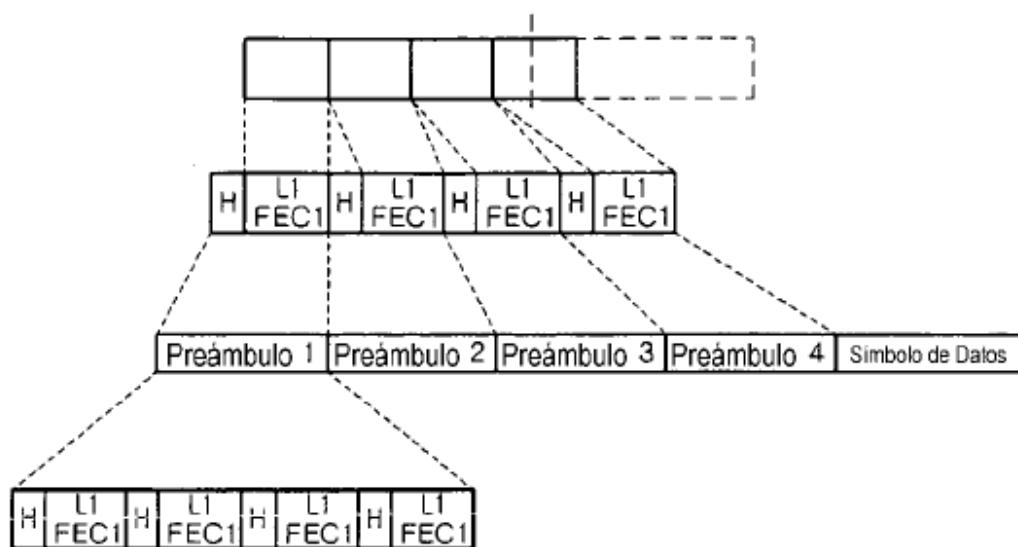


Fig. 61

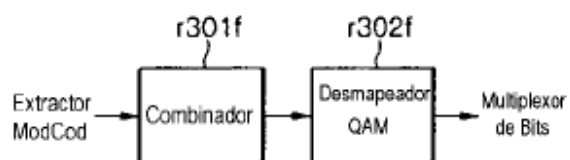


Fig. 62

Campo	Bits	
L1_size	14	tamaño del bloque L1(bits)
num_chbon	3	número de canales adheridos
num_dslice	8	número de segmentos de datos
num_plp	8	número de PLP
num_notch	5	número de bandas de supresión
for dslice {		
chbon_index	3	índice del canal adherido
dslice_start	9	comienzo del segmento de datos dentro de un canal (8 MHz)
dslice_width	9	ancho del segmento de datos
dslice_time_intrlv	1	conexión/desconexión del entrelazado de tiempos
dslice_type	1	tipo de segmento de datos (CCM, ACM/VCM)
}		
for plp {		
dslice_id	8	ID del segmento de datos
plp_id	8	ID de la PLP
plp_type	1	ID de la PLP
plp_payload_type	5	tipo de PLP (común/datos)
if dslice_type=CCM {		
plp_mod	3	tipo de modulación PLP
plp_fec_type	1	tipo de contenido útil de la PLP (TS, GS,...)
plp_cod	3	tasa de códigos de la PLP
plp_start	21	dirección de comienzo de la PLP
}		
}		
for notch {		
chbon_index	3	índice del canal adherido
notch_start	9	comienzo de la banda de supresión dentro de un canal (8 MHz)
notch_width	9	ancho de la banda de muesca
}		
gi	1	modo del intervalo de guarda
sframe_id	16	ID del supercuadro
frame_id	16	ID del cuadro
reserved	16	rfu
crc32	32	CRC32
Total	19458	

El número de bits de información de L1 varía de acuerdo con varias configuraciones/condiciones

Fig. 63

Total	Bits	
L1_size	14	tamaño de bloque (bits)
num_chbon	3	número de canales adheridos
num_dslice	8	número de segmentos de datos
num_plp	8	número de PLP
num_notch	5	número de bandas de muesca
for dslice {		
chbon_index	3	índice del canal adherido
dslice_start	9	comienzo de la ranura de datos de un canal (8 MHz)
dslice_width	9	ancho de la ranura de datos
dslice_time_intrlv	1	conexión/desconexión de datos del intercalado de tiempos
dslice_type	1	tipo de ranura de datos (CCM, ACM/VCM)
}		
for plp {		
dslice_id	8	ID de la ranura de datos
plp_id	8	ID de la PLP
plp_type	1	tipo de PLP (largo/corto)
plp_payload_type	5	tipo de contenido útil de la PLP (TS, GS,...)
if dslice_type=CCM {		
plp_mod	3	tipo de modulación PLP
plp_fec_type	1	tipo del FEC de la PLP (largo/corto)
plp_cod	3	tasa de códigos de la PLP
plp_start	14	dirección de comienzo de la PLP
}		
}		
for notch {		
chbon_index	3	índice del canal adherido
notch_start	9	comienzo de la banda de muesca dentro de un canal (8 MHz)
notch_width	9	ancho de la banda de muesca
}		
gi	1	modo del intervalo de guarda
sframe_id	16	ID del supercuadro
frame_id	16	ID del cuadro
reserved	16	rfu
crc32	32	CRC32
Total	17666	

En número de bits de información de L1 varía
de acuerdo con varias configuraciones/condiciones

Fig. 64

Longitud LDPC	Tipo QAM	Símbolos QAM
64800	16 QAM	16200
64800	64 QAM	10800
64800	256 QAM	8100
64800	1024 QAM	6480
64800	4096 QAM	5400
16200	16 QAM	4050
16200	64 QAM	2700
16200	256 QAM	2025
16200	1024 QAM	1620
16200	4096 QAM	1350

Fig. 65

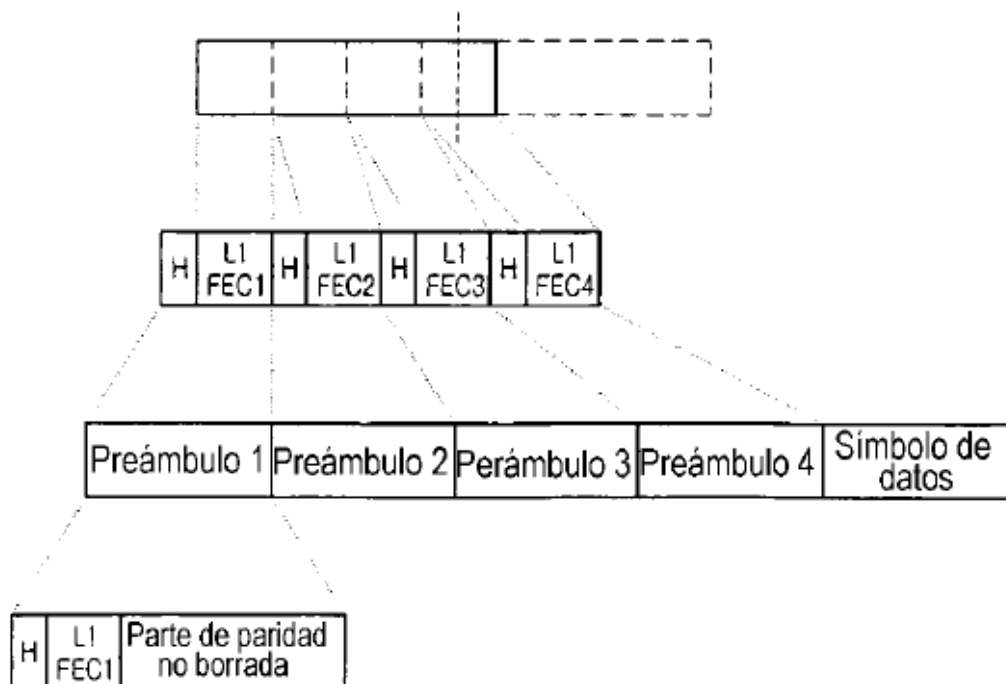


Fig. 66

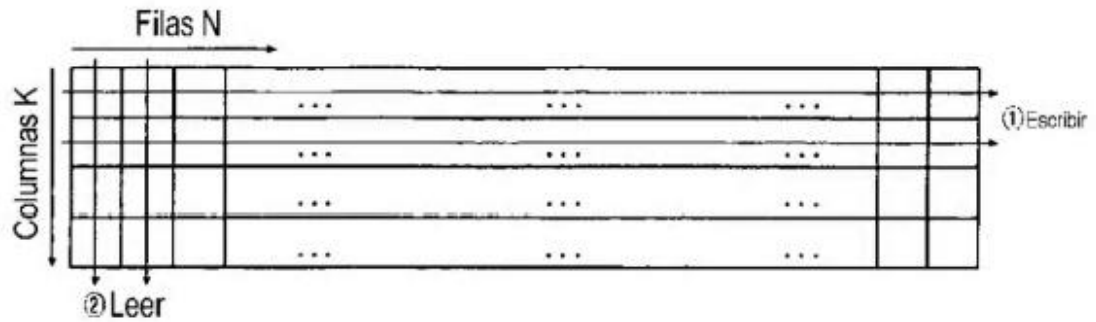


Fig. 67

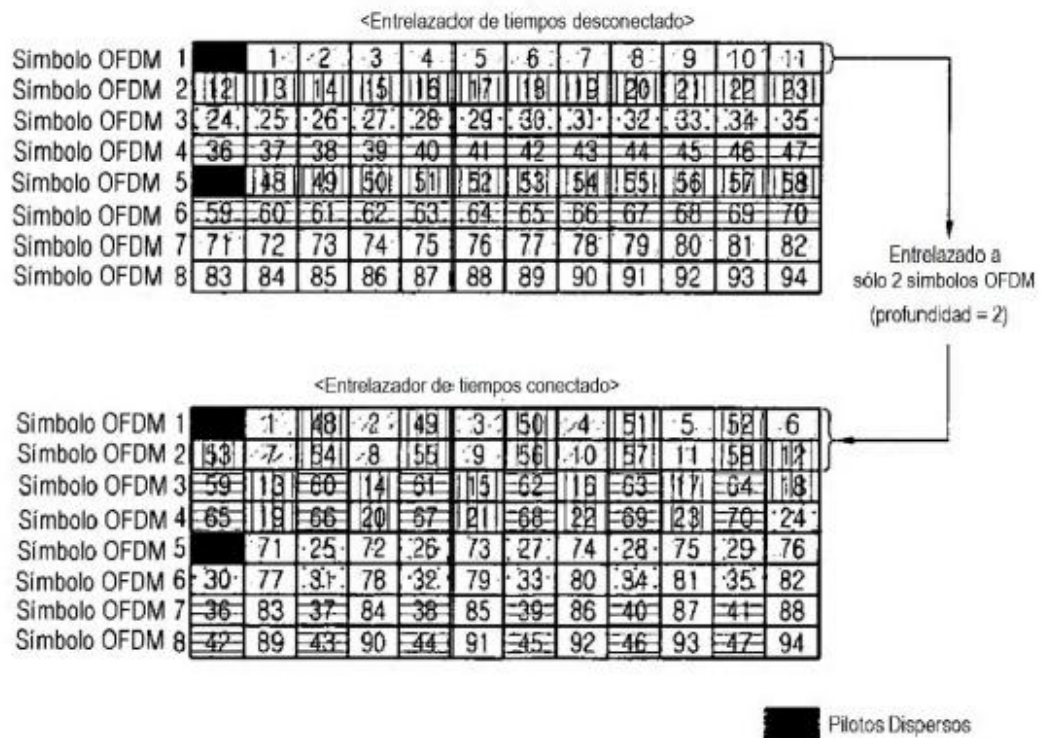


Fig. 68

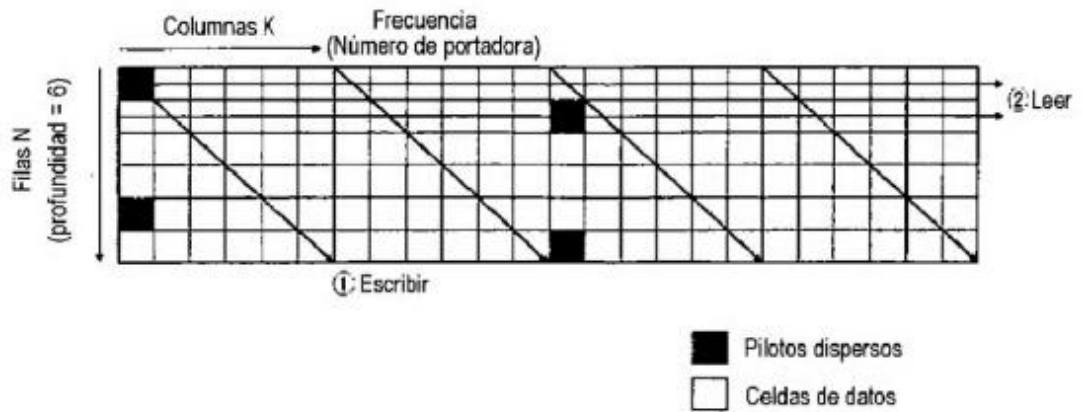


Fig. 69

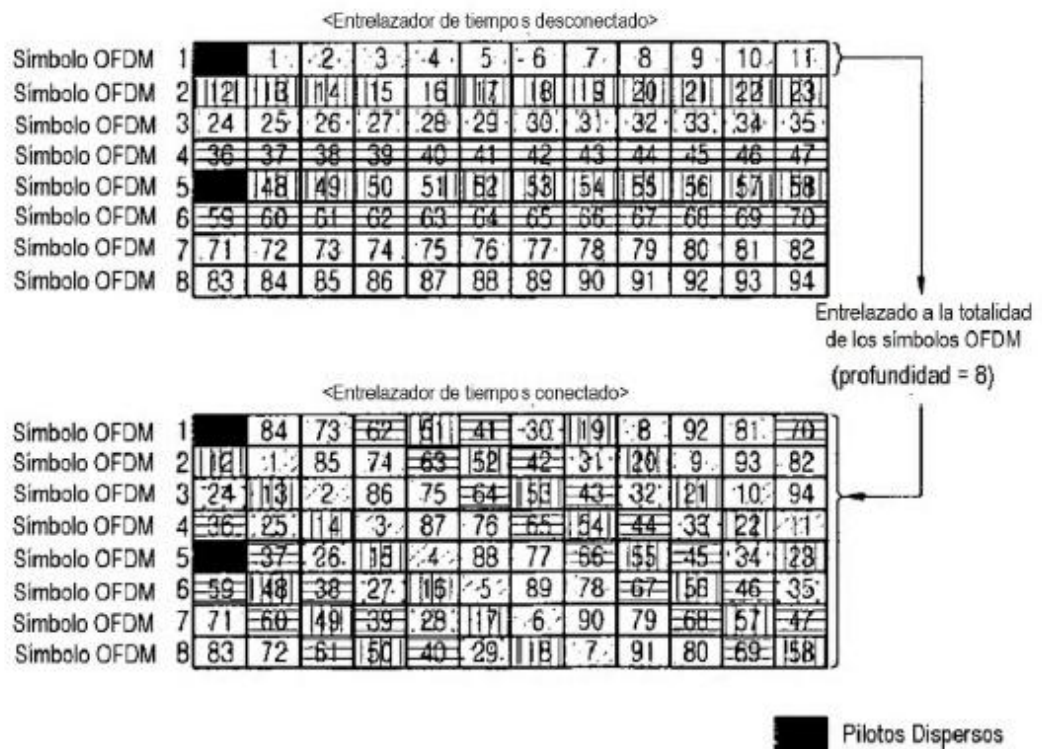


Fig. 70

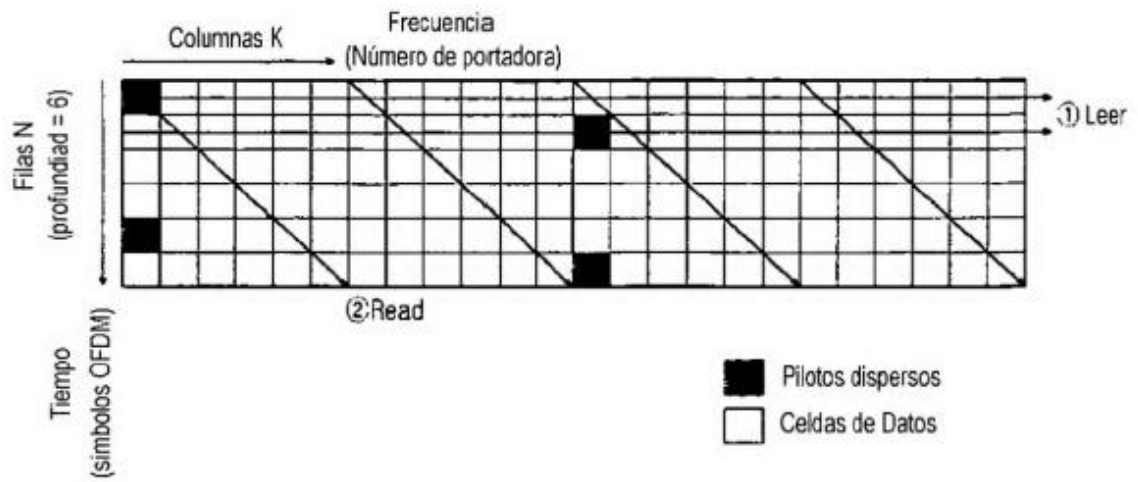


Fig. 71

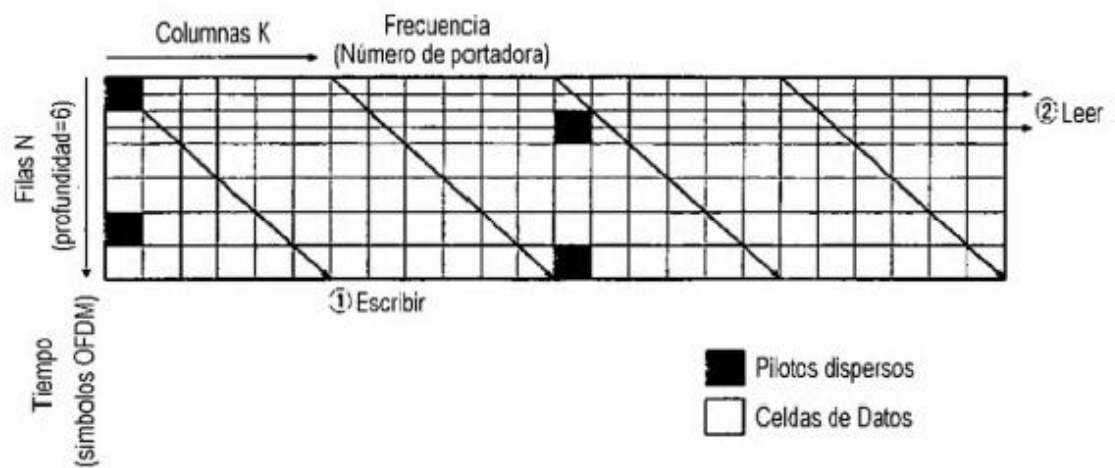


Fig. 72

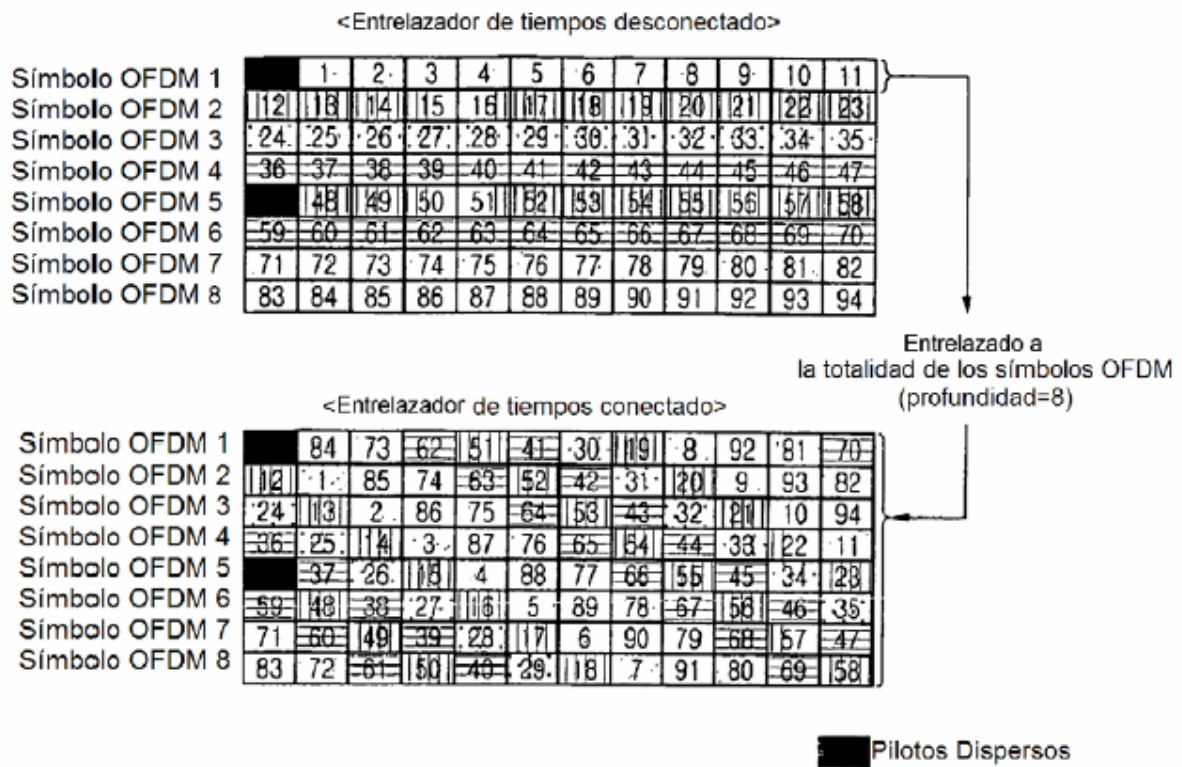
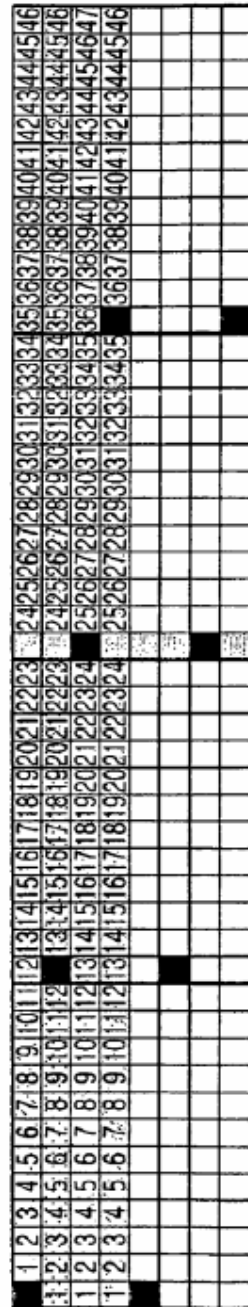


Fig. 73



Tras la aplicación del
Entrelazado de Tiempos



Antes de la aplicación del
Entrelazado de Tiempos

RA=0; CA=0;

while K<nCELL loop

 If addr = posición piloto

 RA = [RA+1] mod NT;

 CA = [CA+1] mod ND;

 end if ;

ESCRIBIR;

 RA = [RA+1] mod NT;

 CA = [CA+1] mod ND;

 if CA = 0

 RA = RA+1;

 end if ;

end loop;

■ Piloto disperso

□ Piloto continuo

RA = dirección de fila

CA = dirección de columna

cCell = número de células dentro del bloque T

Sobrecarga de memoria = 2%

Fig. 74

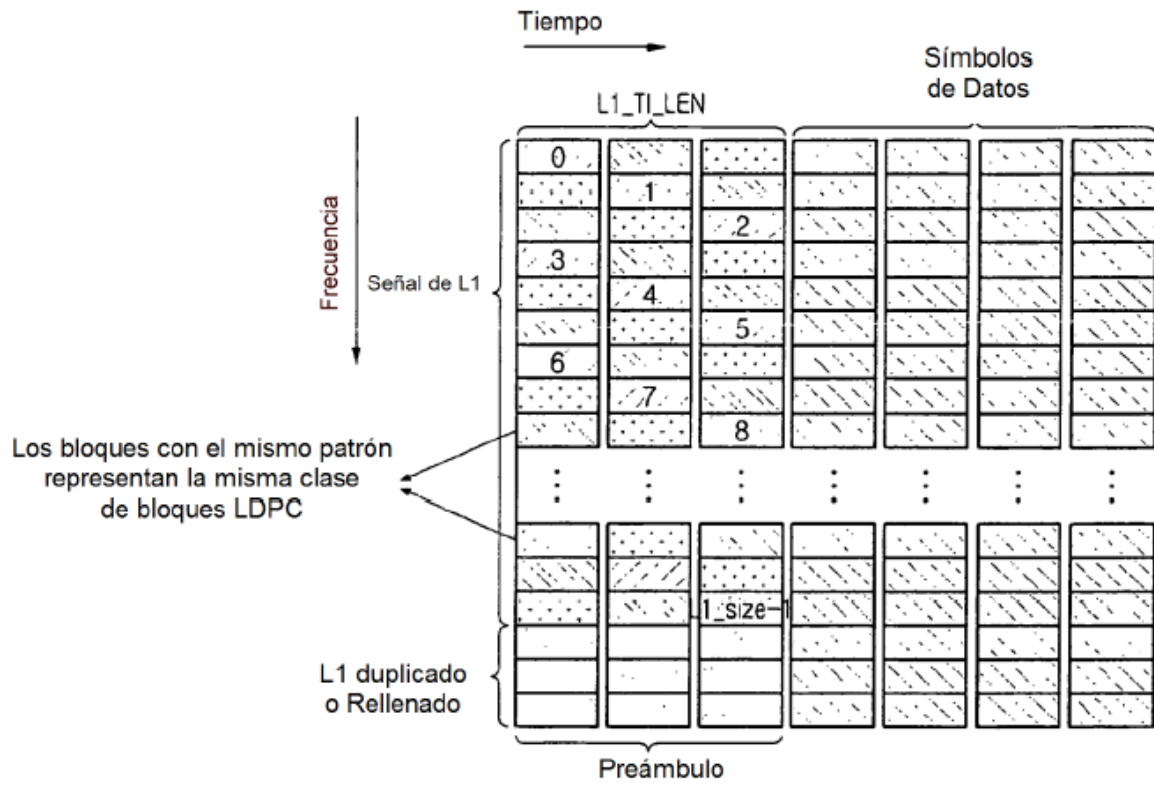


Fig. 75

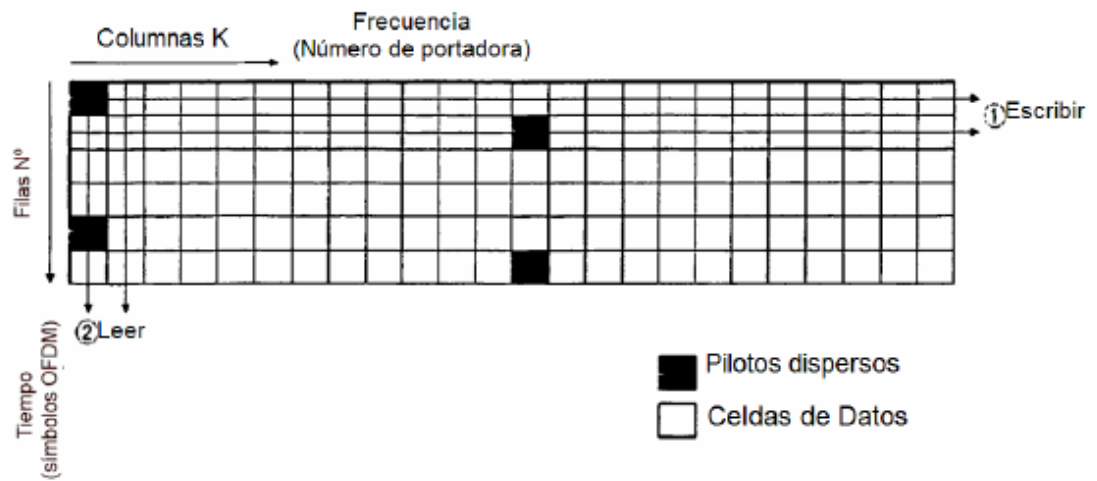


Fig. 76

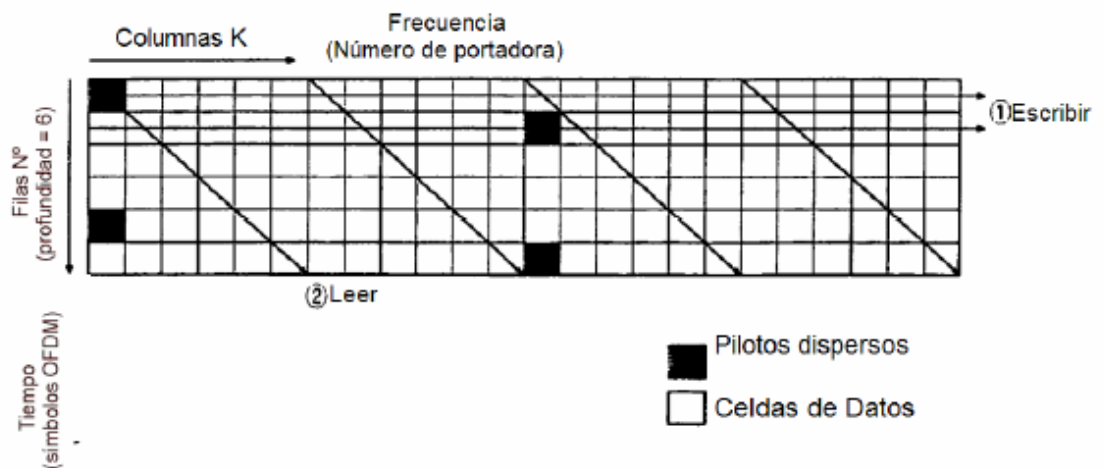


Fig. 77

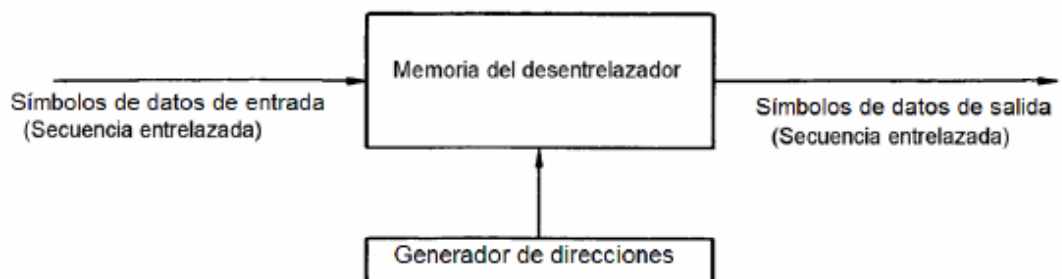


Fig. 78

		i=0	i=12												i=23											
j=0	Ri,j	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
	Ci,j	0	1	2	3	4	5	6	7	8	9	10	11	0	1	2	3	4	5	6	7	8	9	10	11	11
j=1	Ri,j	0	1	2	3	4	5	6	7	0	1	2	3	1	2	3	4	5	6	7	0	1	2	3	4	4
	Ci,j	0	1	2	3	4	5	6	7	8	9	10	11	0	1	2	3	4	5	6	7	8	9	10	11	11
j=2	Ri,j	0	2	4	6	0	2	4	6	0	2	4	6	1	3	5	7	1	3	5	7	1	3	5	7	7
	Ci,j	0	1	2	3	4	5	6	7	8	9	10	11	0	1	2	3	4	5	6	7	8	9	10	11	11
j=3	Ri,j	0	3	6	1	4	7	2	5	0	3	6	1	1	4	7	2	5	0	3	6	1	4	7	2	2
	Ci,j	0	1	2	3	4	5	6	7	8	9	10	11	0	1	2	3	4	5	6	7	8	9	10	11	11

Fig. 79

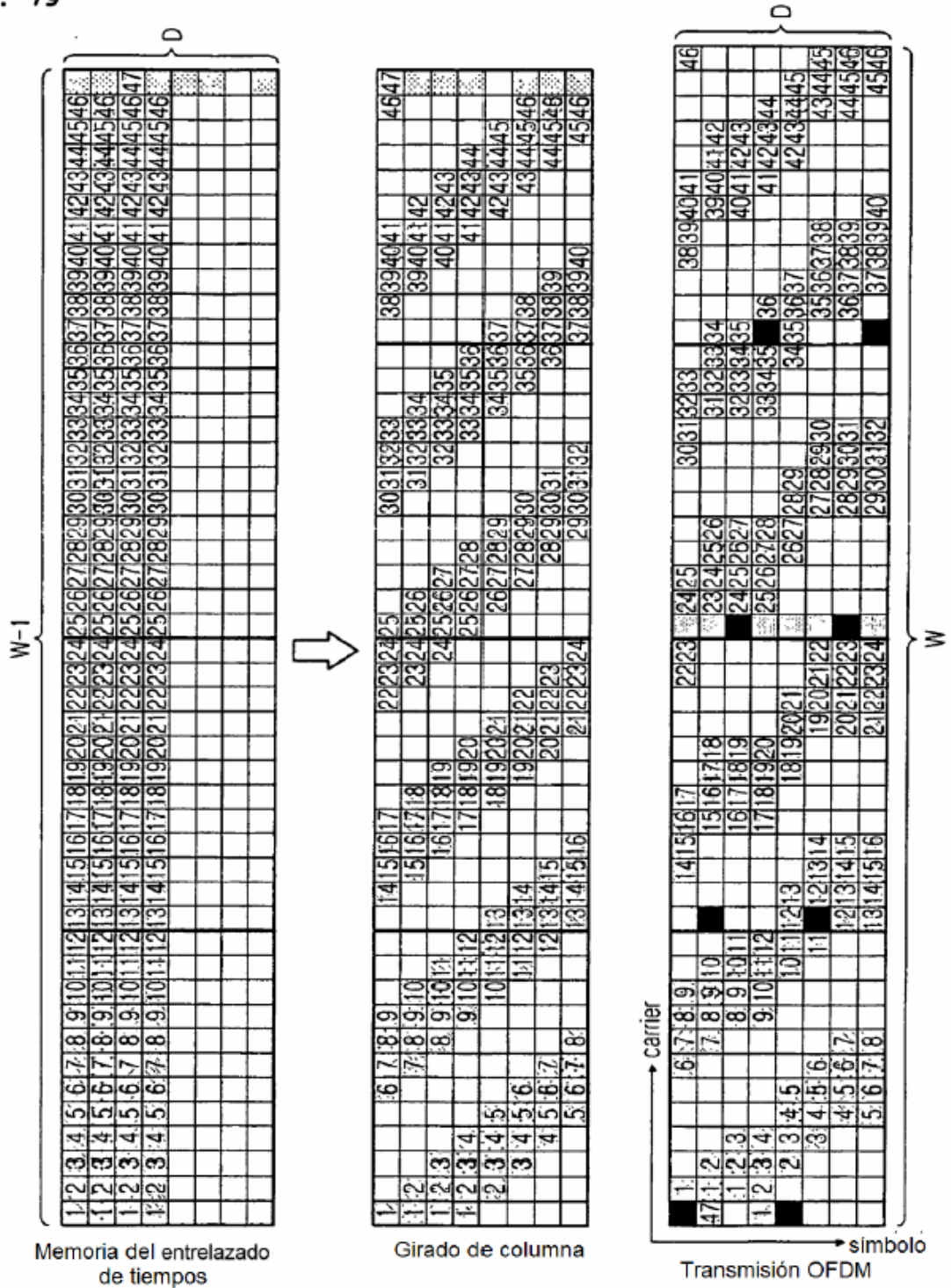


Fig. 80

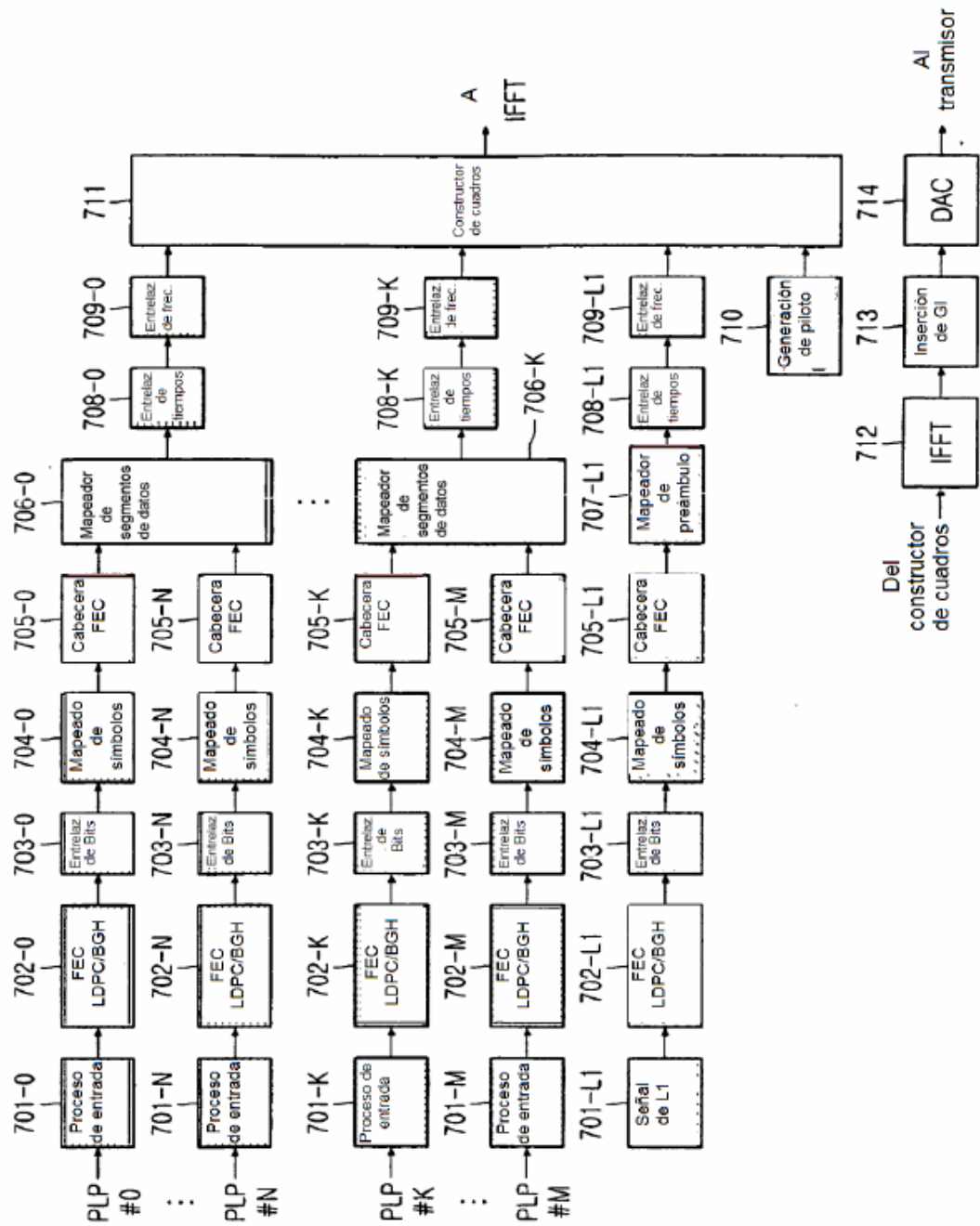


Fig. 81

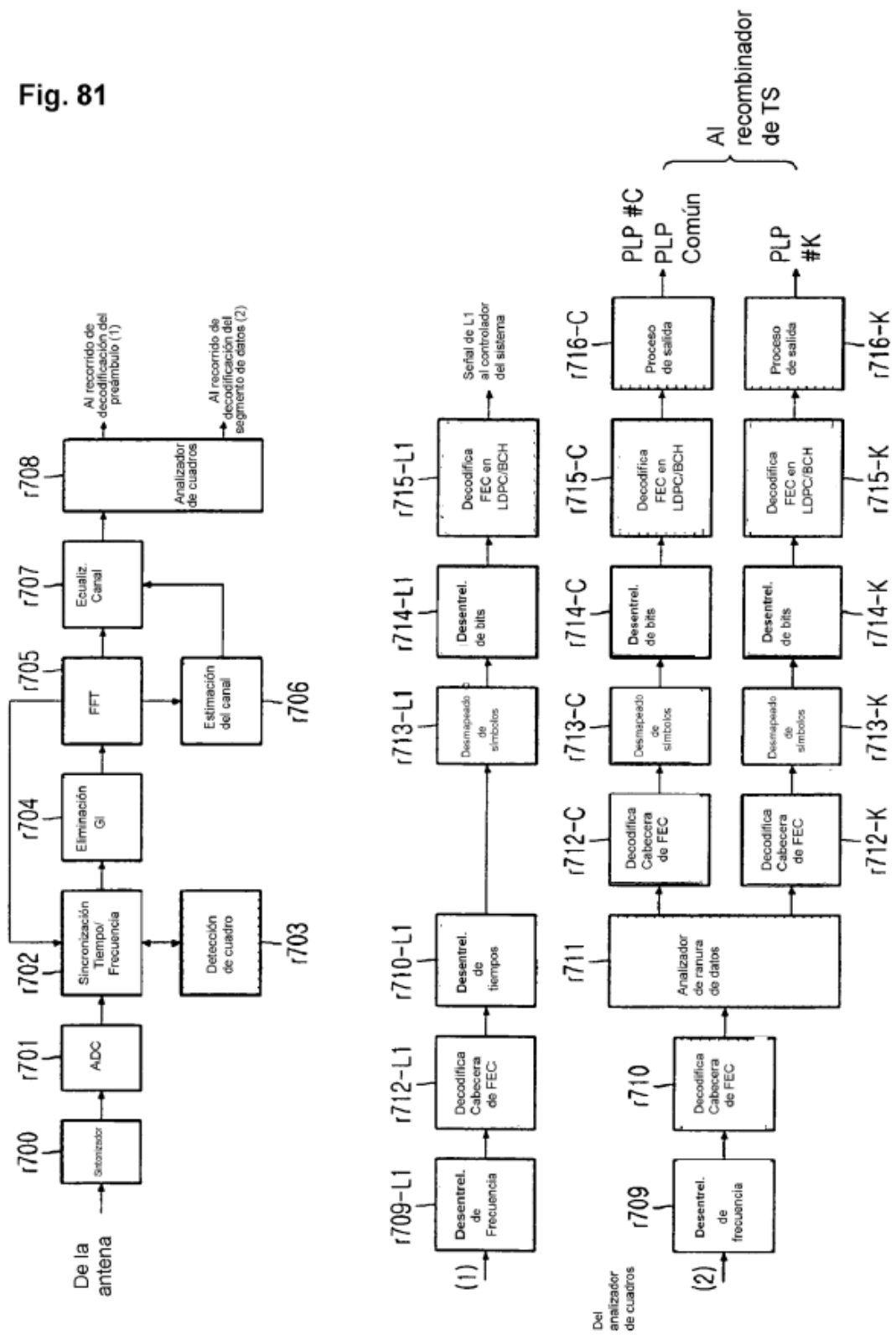
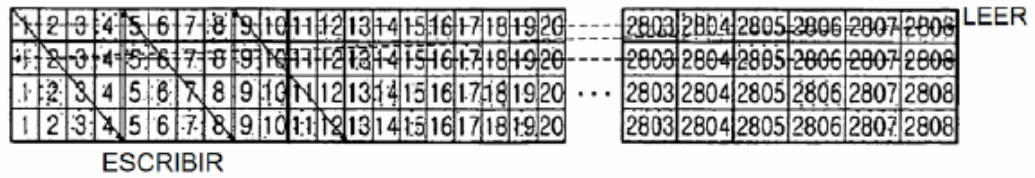


Fig. 82

Entrelazado de tiempos



Desentrelazado de tiempos

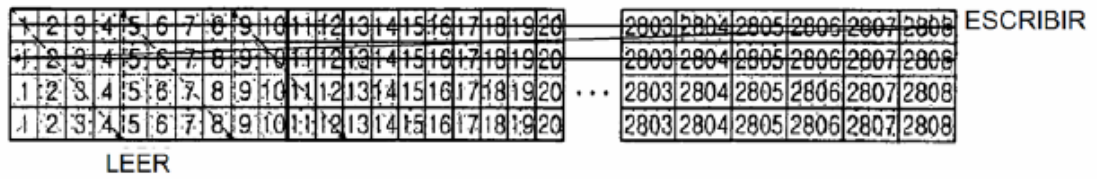


Fig. 83

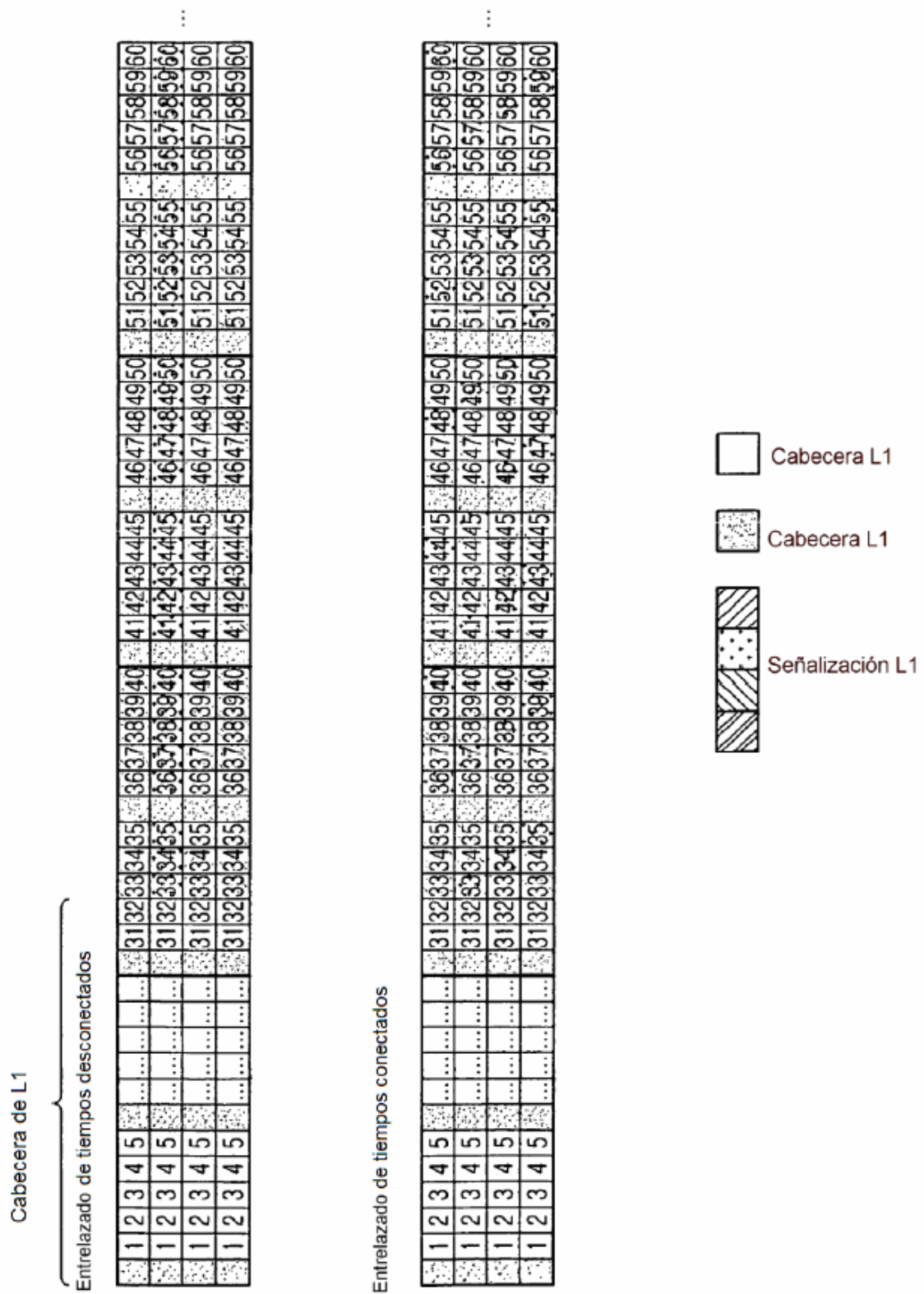


Fig. 85

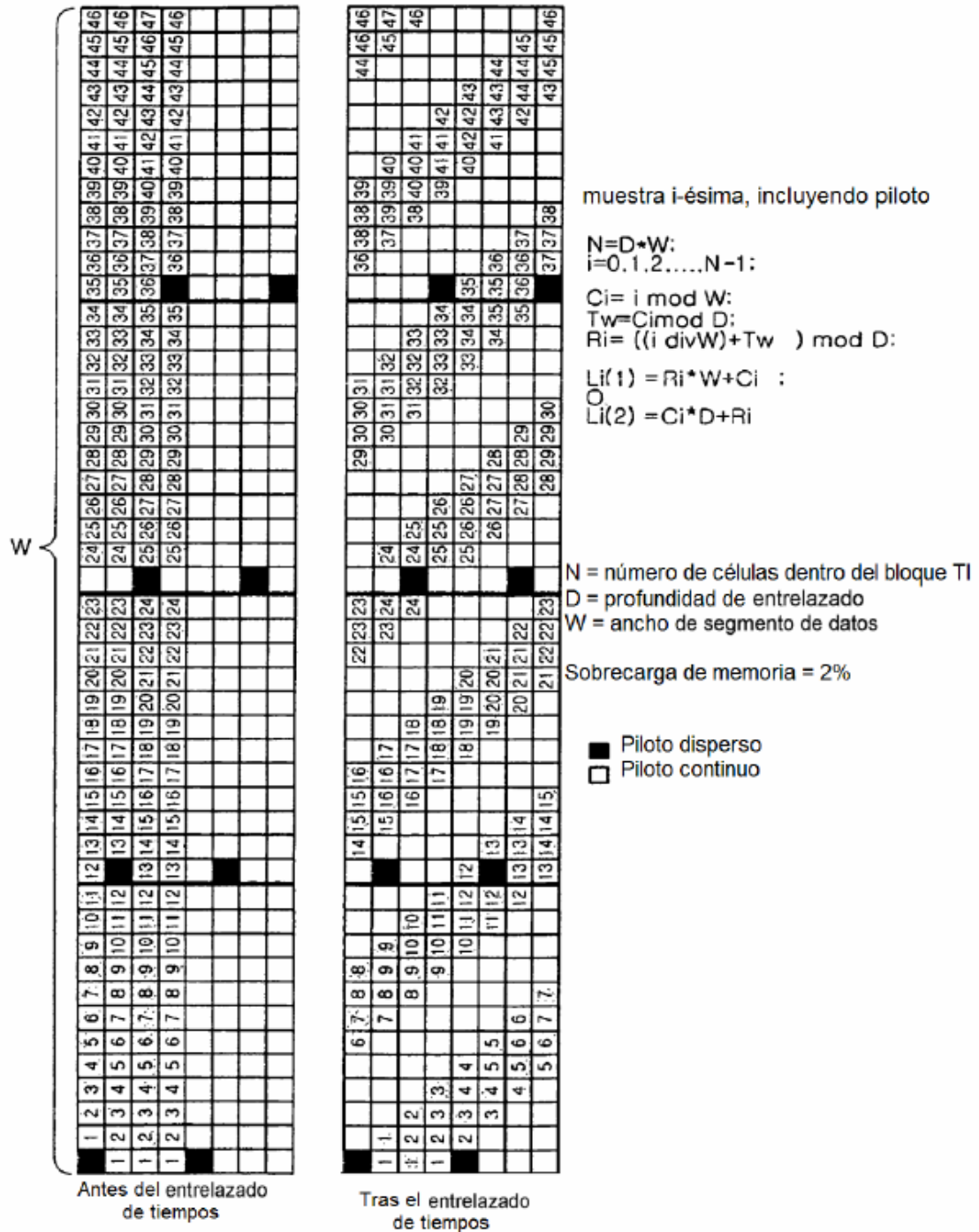


Fig. 86

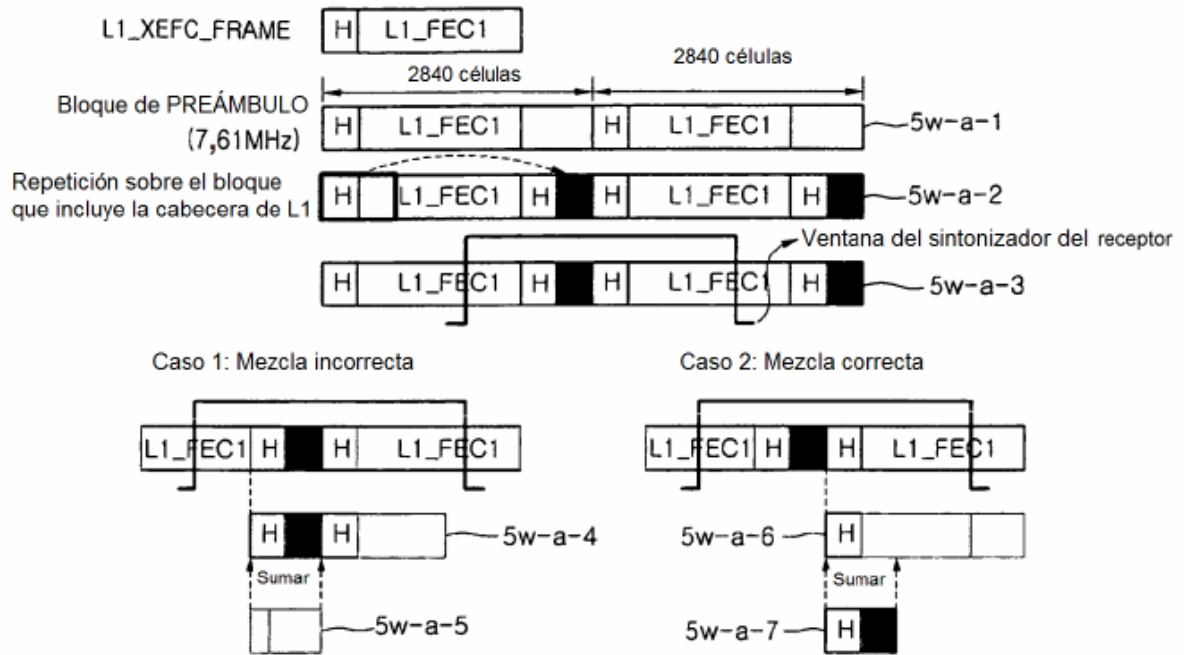


Fig. 87

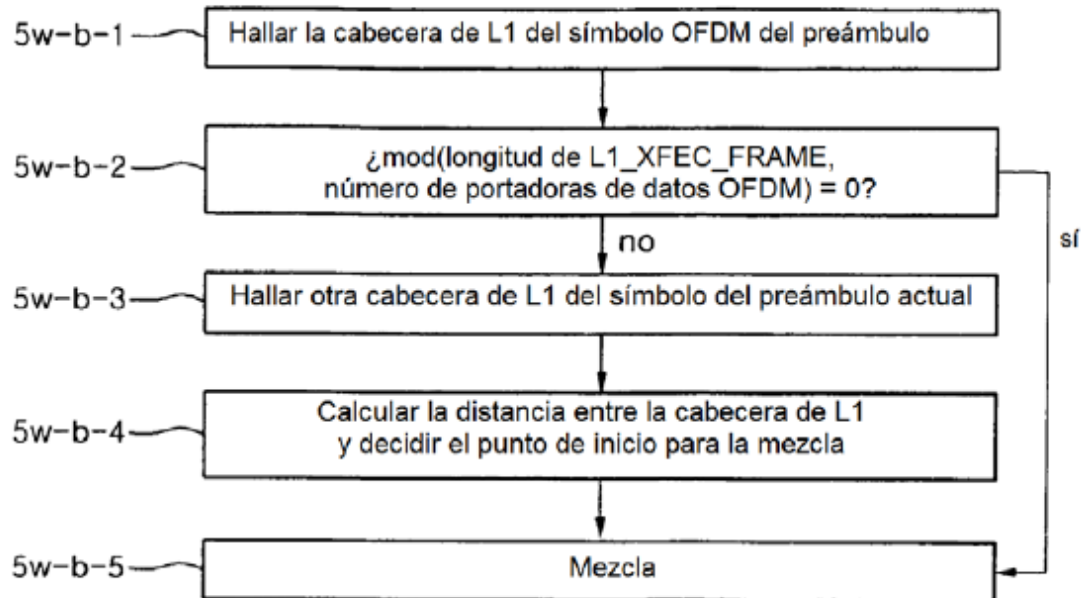


Fig. 88

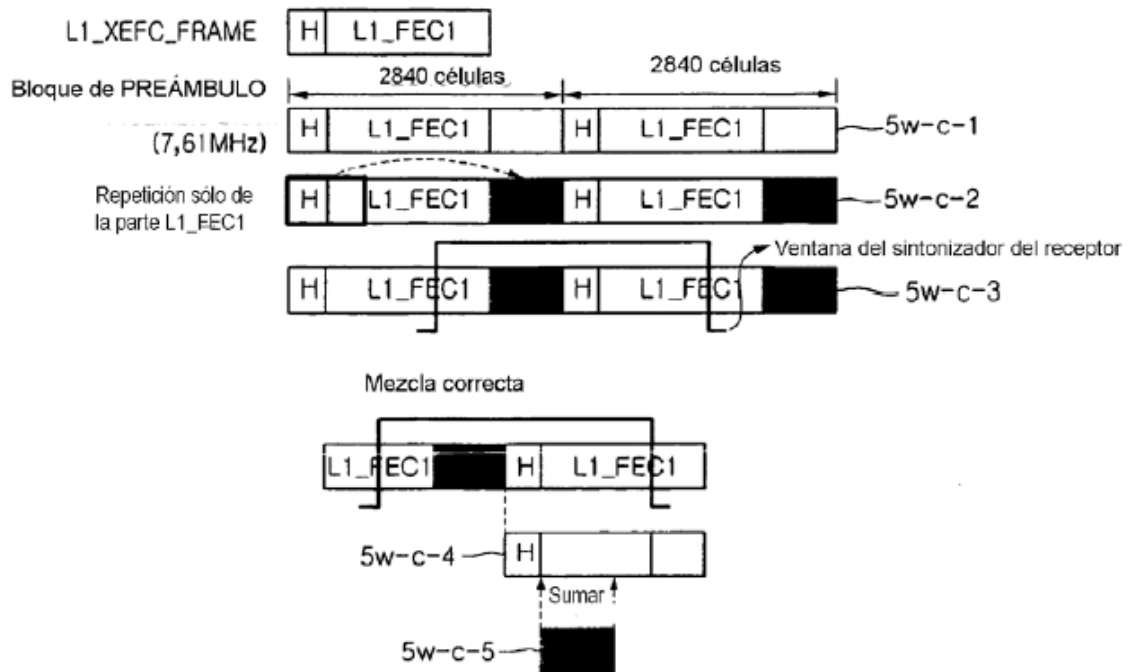
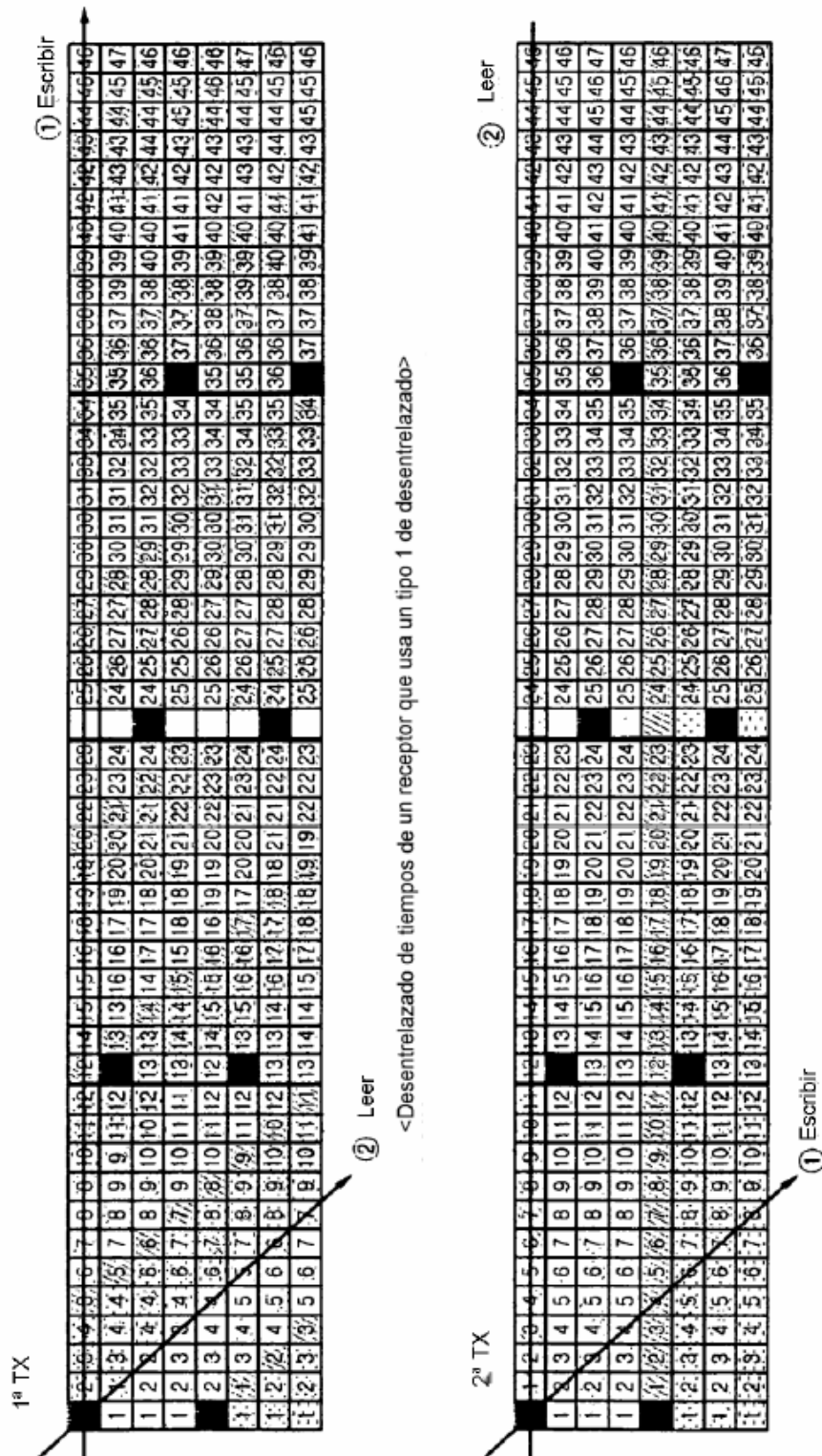


Fig. 89



<Desentrelazado de tiempos de un receptor que usa un tipo 2 de desentrelazado>

Fig. 90

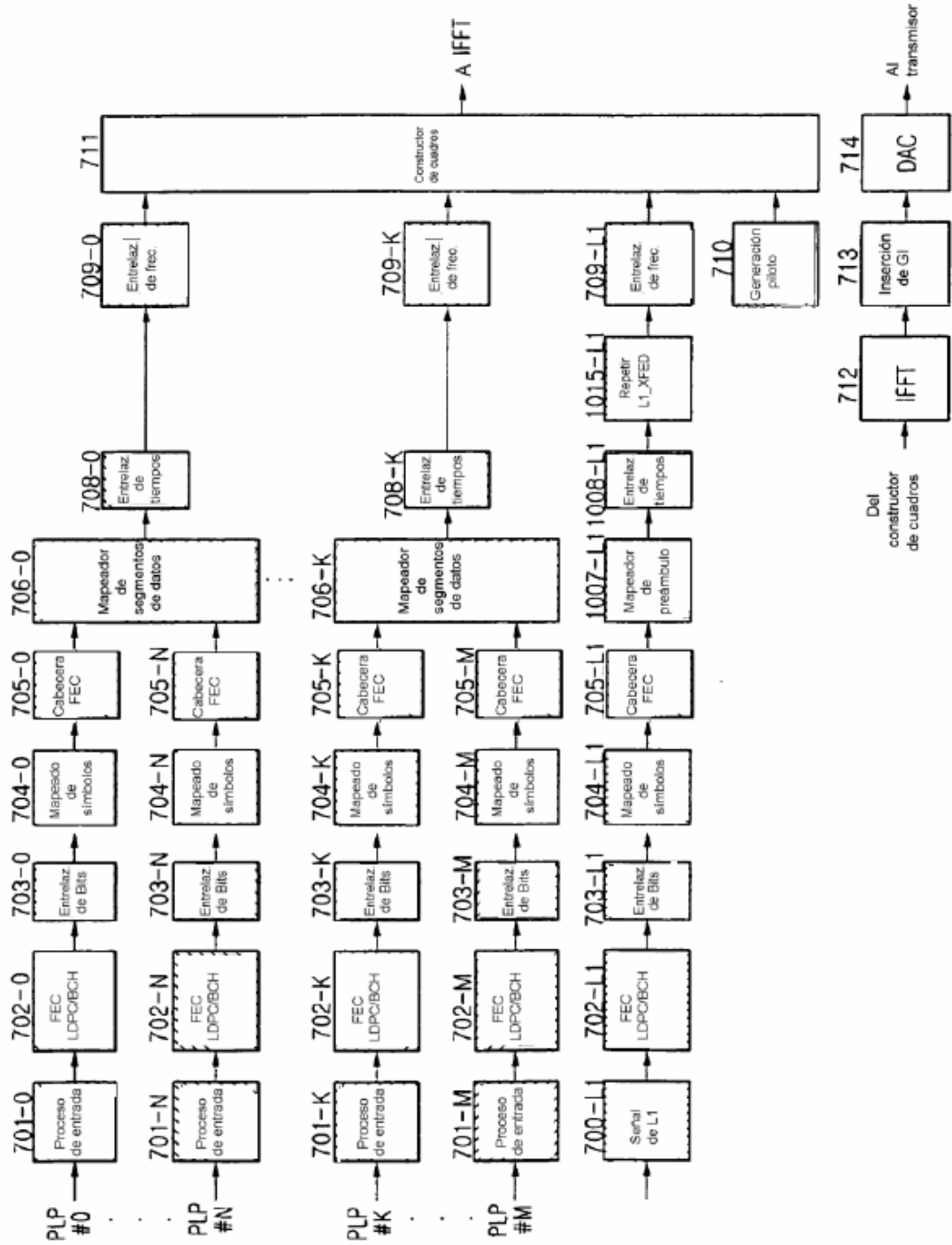


Fig. 91

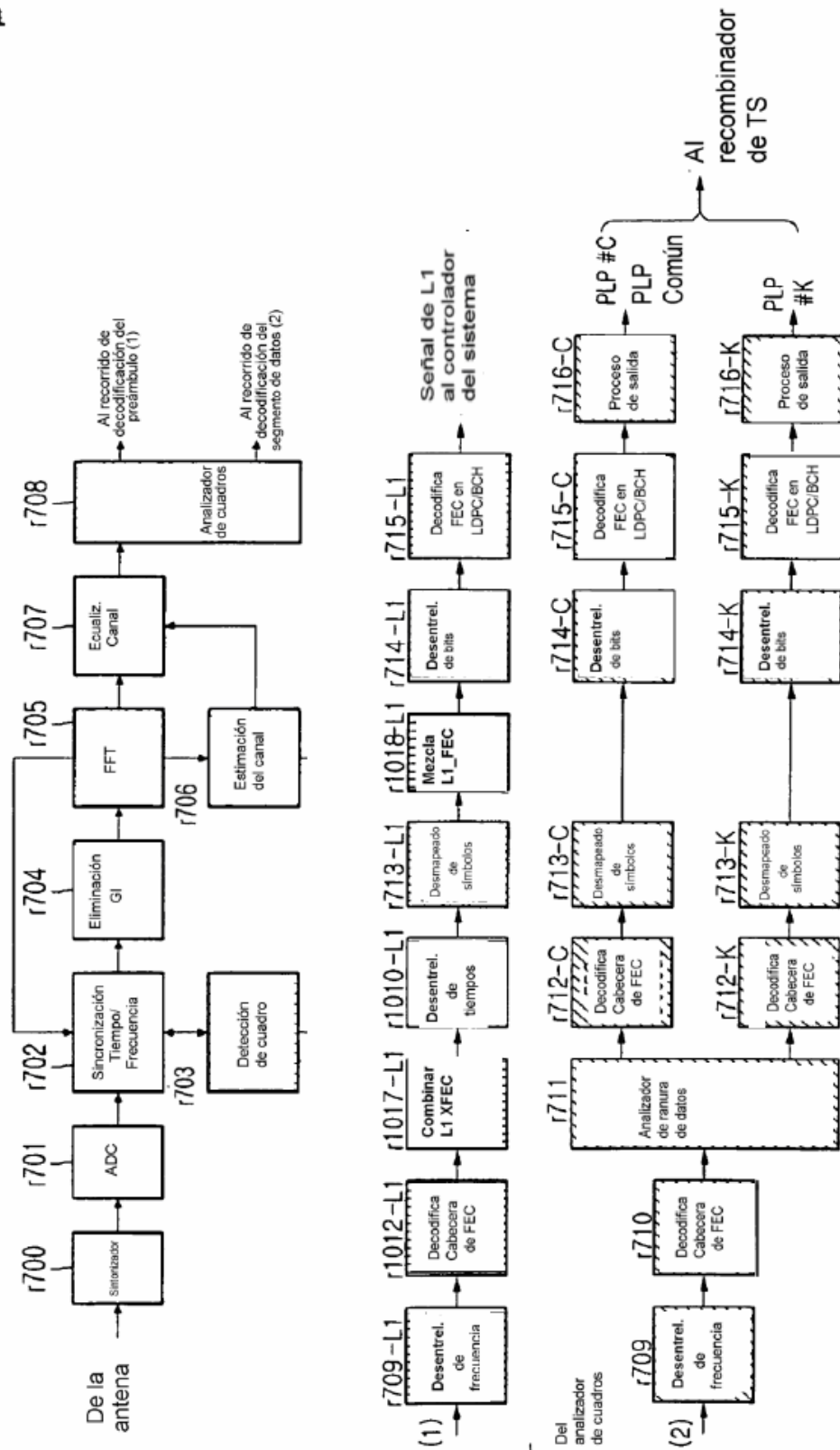
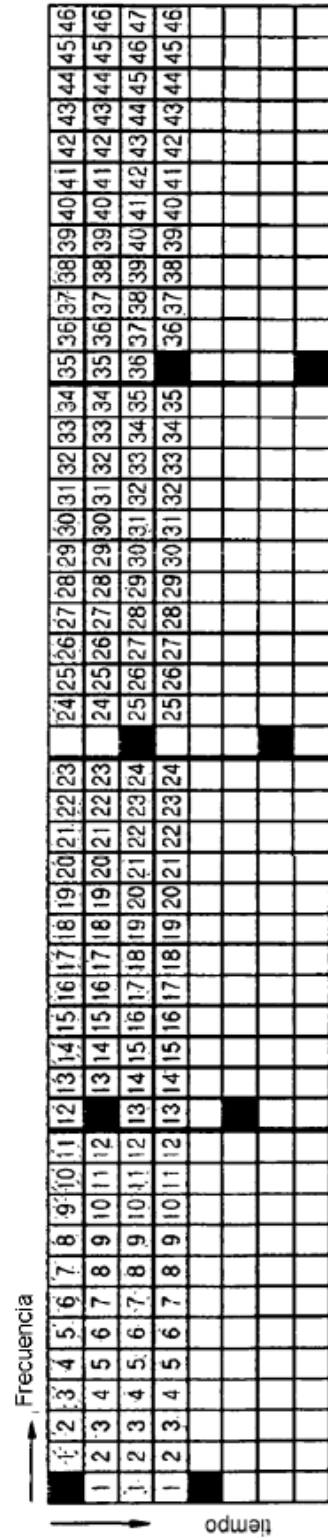
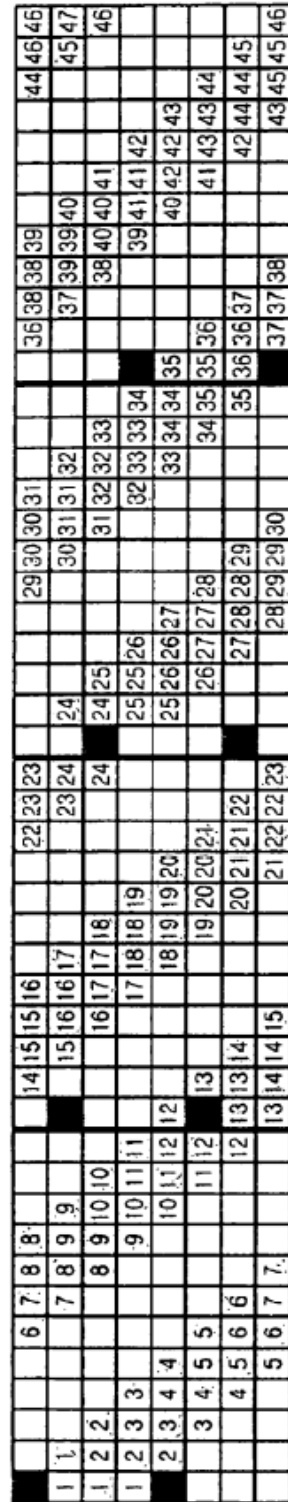


Fig. 92



Entrelazado de tiempos DESCONECTADO



Entrelazado de tiempos CONECTADO

Fig. 93

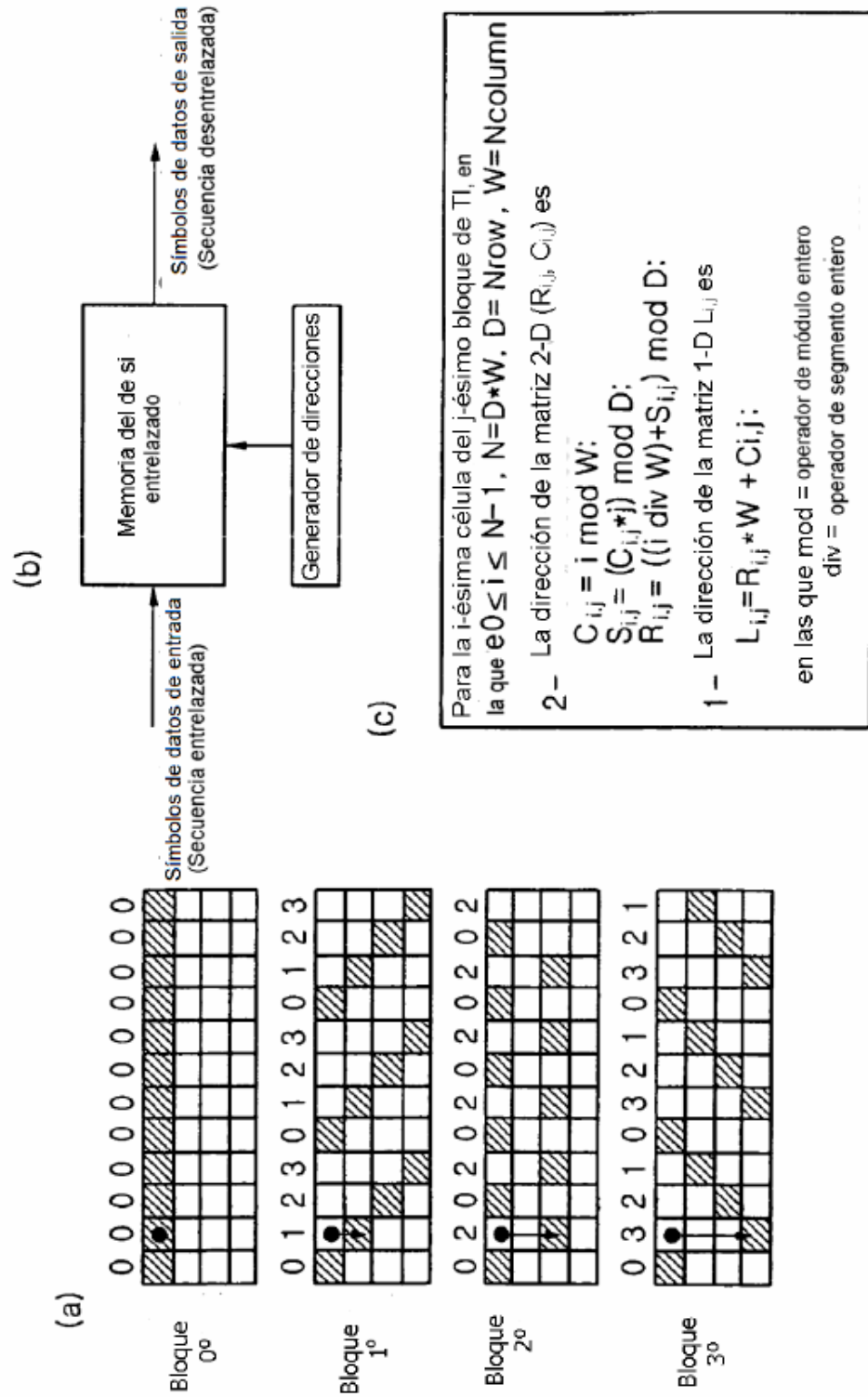


Fig. 94

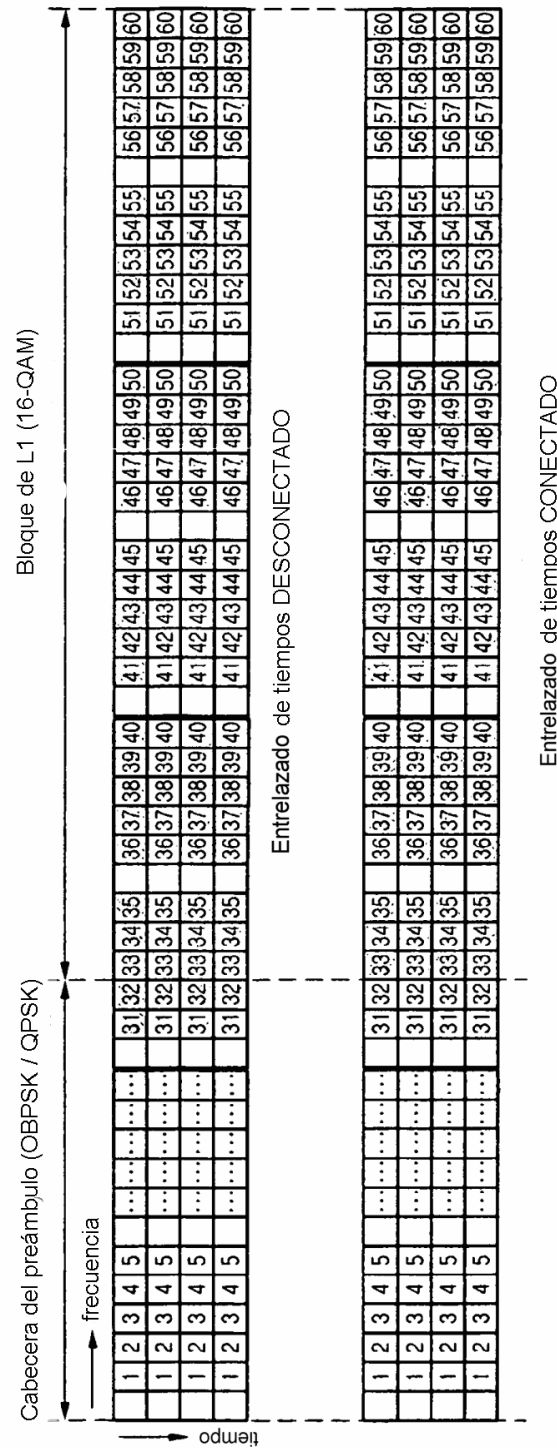


Fig. 95

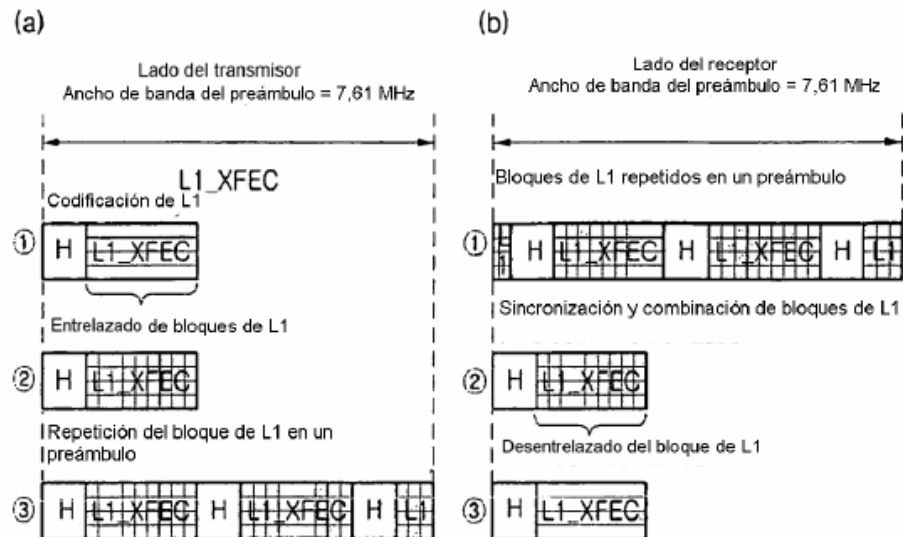


Fig. 96

Valor	Profundidad de TI	Cuando se acusa el marcador de TI (1 bit) TI para el bloque de L1 corto Para un mejor rendimiento de TI
00	No TI	
01	mejor ajuste	
10	4 símbolos OFDM	
11	8 símbolos OFDM	

Fig. 97**(a)** Señalización y estructura de la cabecera

Nº	Características Clave	Campos de Señalización	Relleno	Rendimiento de PER de la cabecera de L1	Rendimiento TI del bloque L1	Sobre-descarga adicional
1	Solo marcador de activo/inactivo	L1 (15), TI (1), no CRC	No	Normal	Malo para bloque L1 corto	No
2	L1_size_bits / 8	L1 (12), TI (2), CRC (2)	Máx 7 bits	Mejor	Bueno	No
3	L1_size_cells	L1 (13), TI (2), CRC (1)	Máx 3 bits	Bueno	Bueno	No
4	L1_size_bits por símbolo	L1 (13), TI (2), CRC (1)	Máx 4 bits	Bueno	Bueno	No
5	L1_size_cells por símbolo	L1 (11), TI (2), CRC (2), Rfu(1)	Máx 19 bits	Mejor	Bueno	No
6	Dos cabeceras (32 bits)	L1 (15), TI (2), CRC (4), Rfu(11)	No	Mejor	Bueno	0,01% (despreciable)

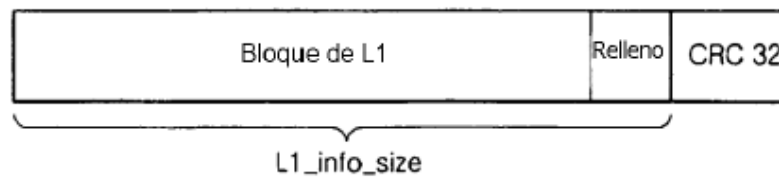
(b) Procedimiento de relleno

Fig. 98

Campo	Bits	Descripción
num_chbon	3	número de canales adheridos
num_dslice	8	número de segmentos de datos
num_notch	4	número de bandas de supresión
for i=0..num_dslice-1 {		
dslice_id	8	identificador de del segmento de datos
dslice_start	12	comienzo de del segmento de datos dentro de un canal C2 (Máx. 64 MHz)
dslice_width	9	ancho de del segmento de datos (Máx- 7,61 MHz)
dslice_ti_depth	2	profundidad del entrelazado de tiempos
dslice_type	1	tipo de segmento de datos (tipo 1, tipo 2)*
dslice_pwr_allocation	2	asignación de potencia a del segmento de datos
num_plp	8	número de PLP
for i=0..num_plp-1 {		
plp_id	8	identificador de la PLP
plp_type	1	tipo de PLP (común/datos)
PSI/SI reprocessing	1	marcador de reprocesamiento PSP/SI
plp_payload_type	5	tipo de contenido útil de la PLP (TS, GS,...)
if dslice_type=='0' {		
plp_modcod	5	modulación/decodificación de la PLP
plp_start_addr	14	dirección de inicio de la PLP
}		
}		
}		
for i=0..num_notch-1{		
notch_start	12	comienzo de la banda de supresión dentro de un canal de C2 (Máx. 64 MHz)
notch_width	9	ancho de la banda de supresión (Máx. 7,61 MHz)
}		
gi	2	modo del intervalo de guarda
num_data_symbols	10	número de símbolos OFDM de datos por cuadro de C2
num_c2_frames	8	número de cuadros C2 por supercuadro
frame_idx	8	índice del cuadro C2 dentro de un supercuadro
PAPR	1	marcador PAPR
reserved	0	reservado para uso futuro
crc32	32	CRC32

* tipo 1: PLP simple con CCM

* tipo 2: Otros

Fig. 99

dslice_ti_depth	Profundidad de TI
"00"	1
"01"	4
"10"	8
"11"	16

Fig. 100

dslice_type	Tipo
"0"	PLP simple con CCM
"1"	Otros

Fig. 101

plp_type	Tipo
"0"	plp común
"1"	plp de datos

Fig. 102

plp_payload_type	Tipo de contenido útil
"00000"	GFPS
"00001"	GCS
"00010"	GSE
"00011"	TS
Otros	reservado para uso futuro

Fig. 103

plp_modcod	Modulación	Tasa de códigos	Bloque LDPC
"00000"	16QAM	4/5	largo
"00001"			corto
"00010"		9/10	largo
"00011"			corto
"00100"	64QAM	2/3	largo
"00101"			corto
"00110"		4/5	largo
"00111"			corto
"01000"		9/10	largo
"01001"			corto
"01010"	256QAM	3/4	largo
"01011"			corto
"01100"		5/6	largo
"01101"			corto
"01110"		9/10	largo
"01111"			corto
"10000"	1024QAM	3/4	largo
"10001"			corto
"10010"		5/6	largo
"10011"			corto
"10100"		9/10	largo
"10101"			corto
"10110"	4096QAM	5/6	largo
"10111"			corto
"11000"		9/10	largo
"11001"			corto
otros	reservado para uso futuro	reservado para uso futuro	reservado para uso futuro

Fig. 104

gi	valor
"00"	1/128
"01"	1/64
Otros	reservado para uso futuro

Fig. 105

PAPR	valor
"0"	inhabilitar
"1"	habilitar

Fig. 106

Campo	Bits	Descripción
network_id	16	ID de la red DVB-C2 actual
c2_system_id	16	ID del sistema C2 dentro de la red DVB-C2
C2_signal_start-frequency	32	frecuencia de inicio de la señal C2
C2_signal_stop_frequency	32	frecuencia de parada de la señal C2
num_chbon	3	número de canales adheridos
num_dslice	8	número de segmentos de datos
num_notch	4	número de bandas de supresión
for i=0..num_dslice-1{		
dslice_id	8	identificador de del segmento de datos
dslice_start	12	comienzo de del segmento de datos dentro de un canal C2 (Máx. 64 MHz)
dslice_width	9	ancho de del segmento de datos (Máx- 7,61 MHz)
dslice_ti_depth	2	profundidad del entrelazador de tiempos
dslice_type	1	tipo de segmento de datos (tipo 1, tipo 2)*
dslice_pwr_allocation	2	asignación de potencia a del segmento de datos
reserved_1	0	reservado para uso futuro
num_plp	8	número de PLP
for i=0..num_plp-1 {		
plp_id	8	identificador de la PLP
plp_type	2	tipo de PLP (común/datos normales/datos agrupados)
If plp_type=='common' or grouped'{		
Plp_group_id	8	identificador del grupo de PLP
}		
plp_payload_type	5	tipo de contenido útil de la PLP (TS, GS,...)
PSI/SI reprocessing	1	marcador de reprocesamiento PSP/SI
reserved_2	0	reservado para uso futuro
If dslice_type=='0' {		
plp_modmod	5	modulación/decodificación de la PLP
plp_start_addr	14	dirección de inicio de la PLP
}		
}		
for i=0..num_notch-1 {		
notch_start	12	comienzo de la banda de supresión dentro de un canal de C2 (Máx. 64 MHz)
notch_width	9	ancho de la banda de supresión (Máx. 7,61 MHz)
}		
gi	2	modo del intervalo de guarda
num_data_symbols	10	número de símbolos OFDM de datos por cuadro de C2
num_c2-frames	8	número de cuadros C2 por supercuadro
frame_idx	8	índice del cuadro C2 dentro de un supercuadro
PAPR	1	marcador PAPR
reserved_3	0	reservado para uso futuro
crc32	32	CRC32

* tipo 1: PLP simple con CCM

* tipo 2: Otros

Fig. 107

plp_type	Tipo
"00"	plp común
"01"	plp de datos normales
"10"	plp de datos agrupados
otros	reservado para uso futuro

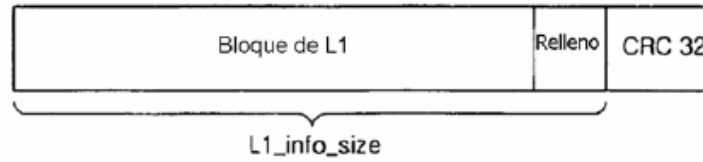
Fig. 108

Campo	Bits	Descripción
network_id	16	ID de la red DVB-C2 actual
c2_system_id	16	ID del sistema C2 dentro de la red DVB-C2
C2_signal_start-frequency	32	frecuencia de inicio de la señal C2
num_chbon	3	número de canales adheridos
num_dslice	8	número de segmentos de datos
num_notch	4	número de bandas de supresión
for i=0..num_dslice-1 {		
dslice_id	8	identificador de del segmento de datos
dslice_start	12	comienzo de del segmento de datos dentro de un canal C2 (Máx. 64 MHz)
dslice_width	12	ancho de del segmento de datos (Máx- 64 MHz)
dslice_ti_depth	2	profundidad del entrelazador de tiempos
dslice_type	1	tipo de segmento de datos (tipo 1, tipo 2)*
num_plp	8	asignación de potencia a del segmento de datos
reserved_1	0	número de PLP
for i=0..num_plp-1 {		
plp_id	8	identificador de la PLP
plp_type	2	tipo de PLP (común/datos normales/datos agrupados)
If plp_type=='common' or grouped {		
plp_group_id	8	identificador del grupo de PLP
}		
PSI/SI reprocessing	1	marcador de reprocesamiento PSP/SI
plp_payload_type	5	tipo de contenido útil de la PLP (TS, GS,...)
reserved_2	0	reservado para uso futuro
If dslice_type=='0' {		
plp_modcod	5	modulación/decodificación de la PLP
plp_start_addr	14	dirección de inicio de la PLP
}		
}		
for i=0..num_notch-1 {		
notch_start	12	comienzo de la banda de supresión dentro de un canal de C2 (Máx. 64 MHz)
notch_width	9	ancho de la banda de supresión (Máx. 7,61 MHz)
}		
gi	2	modo del intervalo de guarda
PAPR	1	marcador PAPR
reserved_3	0	reservado para uso futuro
crc32	32	CRC32
num_data_symbols	221	

* tipo 1: PLP simple con CCM

* tipo 2: Otros

Fig. 109



Valor	Profundidad de TI	Cuando se acusa el marcador de TI (1 bit) TI para el bloque de L1 corto Para un mejor rendimiento de TI
00	No TI	
01	mejor ajuste	
10	4 símbolos OFDM	
11	8 símbolos OFDM	

Fig. 110

Campo	Bits	Descripción
NETWORK_ID	16	ID de la red DVB-C2 actual
C2_SYSTEM_ID	16	ID del sistema C2 dentro de la red DVB-C2
C2_SIGNAL_START-FREQUENCY	32	frecuencia de inicio de la señal C2
NUM_CHBON	3	número de canales adheridos
NUM_DSLICE	8	número de segmentos de datos
NUM_NOTCH	4	número de bandas de supresión
FOR I=0..NUM_DSLICE-1{		
DSLICE_ID	8	identificador de del segmento de datos
DSLICE_START	12	comienzo de del segmento de datos dentro de un canal C2 (Máx. 64 MHz)
DSLICE_WIDTH	12	ancho de del segmento de datos (Máx- 64 MHz)
DSLICE_TI_DEPTH	2	profundidad del entrelazador de tiempos
DSLICE_TYPE	1	tipo de segmento de datos (tipo 1, tipo 2)*
DSLICE_NUM_PLP	8	asignación de potencia a del segmento de datos
FOR I=0..NUM_PLP-1 {		
PLP_ID	8	identificador de la PLP
PLP_TYPE	2	tipo de PLP (común/datos normales/datos agrupados)
PLP_PAYLOAD_TYPE	5	tipo de contenido útil de la PLP (GFPS, GCS, GSE, TS, RFU)
IF PLP_TYPE== '0' o '01'{		
PLP_GROUP_ID	8	identificador del grupo de PLP
}		
}		
If DSLICE_TYPE == '0' {		
PLP_START	14	dirección de inicio de la PLP (unidad = 1 célula OFDM)
PLP_MODCOD	5	modulación/decodificación de la PLP
}		
PSI/SI_REPROCESSING	1	marcador de reprocesamiento PSP/SI
RESERVED_1	8	reservado para uso futuro
}		
RESERVED_2	8	reservado para uso futuro
}		
For i=0.NUM_NOTCH-1{		
NOTCH_START	12	comienzo de la banda de supresión dentro de un canal de C2 (Máx. 64 MHz) (unidad = 12 células OFDM)
NOTCH_WIDTH	9	ancho de la banda de supresión (Máx. 7,61 MHz) (unidad = 12 células OFDM)
RESERVED_3	8	reservado para uso futuro
}		
GI	2	Modo del intervalo de guarda
PAPR	1	marcador PAPR
L1_PART2_CHANGE_COUNTER	8	Número de cuadros C2 por delante que la parte 2 de L1 cambiará excepto en PLP_START (unidad = 1 cuadro C2)
RESERVED_4	16	reservado para uso futuro
CRC32	32	CRC32
Total	269	

Fig. 111

DSLICE_TI_DEPTH	
Valor	Profundidad de TI
00	1 símbolo OFDM
01	4 símbolos OFDM
10	8 símbolos OFDM
11	16 símbolos OFDM

PAPR	
Valor	PAPR
0	inhabilitar
1	habilitar

DSLICE_TYPE	
Valor	Tipo de segmento de datos
0	PLP simple con CCMV
1	Otros

GI	
Valor	Fracción del intervalo de guarda
00	1/128
01	1/64
10 a 11	Reservado para uso futuro

PLP_TYPE	
Valor	Tipo de PLP
00	PLP común
01	PLP de datos agrupados
10	PLP de datos normales
11	Reservado para uso futuro

PLP_PAYLOAD_TYPE	
Valor	Tipo de contenido útil
00000	GFPS
00001	GCS
00010	GSE
00011	TS
00100 a 11111	Reservado para uso futuro

PLP_MODCOD			
Valor	Modulación	T. de códigos	Tipo FEC
00000	16QAM	4/5	64K LDPC
00001			16K LDPC
00010		9/10	64K LDPC
00011		8/9	16K LDPC
00100	64QAM	2/3	64K LDPC
00101			16K LDPC
00110		4/5	64K LDPC
00111			16K LDPC
01000	256QAM	9/10	64K LDPC
01001		8/9	16K LDPC
01010		3/4	64K LDPC
01011			16K LDPC
01100	1024QAM	5/6	64K LDPC
01101			16K LDPC
01110		9/10	64K LDPC
01111		8/9	16K LDPC
10000	4096QAM	3/4	64K LDPC
10001			16K LDPC
10010		5/6	64K LDPC
10011			16K LDPC
10100	16K LDPC	9/10	64K LDPC
10101		8/9	16K LDPC
10110		5/6	64K LDPC
10111			16K LDPC
11000	4096QAM	9/10	64K LDPC
11001		8/9	16K LDPC
11010 a 11111	Reservado para uso futuro		

Fig. 112

Campo	Bits	Descripción
NETWORK_ID	16	ID de la red DVB-C2 actual
C2_SYSTEM_ID	16	ID del sistema C2 dentro de la red DVB-C2
C2_SIGNAL_START_FREQUENCY	32	frecuencia de inicio de la señal C2
GI	2	modo del intervalo de guarda
L1_PART2_CHANGE_COUNTER	8	Número de cuadros C2 por delante que la parte 2 de L1 cambiará excepto en PLP_START (unidad = 1 cuadro C2)
NUM_CHBON	3	número de canales adheridos
NUM_DSLICE	8	número de segmentos de datos
NUM_NOTCH	4	número de bandas de supresión
For i=0NUM_DSLICE-{		
DSLICE_ID	8	identificador de del segmento de datos
DSLICE_START	12(11)	Comienzo de la segmento de datos dentro de un canal C2 (Máx. 64 MHz) (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente)
DSLICE_WIDTH	12(11)	Ancho de la segmento de datos (Máx. 64 MHz) (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente)
DSLICE_TI_DEPTH	2	profundidad del entrelazador de tiempos
DSLICE_TYPE	1	tipo de segmento de datos (tipo 1, tipo 2)*
DSLICE_CONST_FLAG	1	Indicador para segmento de datos constante
DSLICE_NOTCH_FLAG	2	indicador de supresión contigua
DSLICE_NUM_PLP	8	número de PLP en el segmento de datos actual
For i=0DSLICE_NUMPLP		
PLP_ID	8	identificador de la PLP
PLP_TYPE	3	tipo de PLP (común/datos normales/datos agrupados/datos empaquetados)
PLP_PAYLOAD_TYPE	5	tipo de contenido útil de la PLP (GFPS, GCS, GSE, TS, RFU)
iPLP_TYPE='00' o '01' {		
PLP_GROUP_ID	8	identificador del grupo de PLP
}		
IDSLICE_TYPE '0' {		
PLP_START	14	dirección de inicio de la PLP (unidad = 1 célula OFDM)
PLP_MODCOD	5	modulación/decodificación de la PLP
}		
PSI/SI_REPROCESSING	1	marcador de reprocesamiento PSP/SI
RESERVED_1	8	reservado para uso futuro
}		
RESERVED_2	8	reservado para uso futuro
}		
For i=0..NUM_NOTCH-1 {		
NOTCH_START	12(11)	Comienzo de la banda de supresión dentro de un canal de C2 (Máx. 64 MHz) (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente)
NOTCH_WIDTH	9(8)	Ancho de la banda de supresión (Máx. 7,61 MHz) (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente)
RESERVED_3	8	reservado para uso futuro
}		
PAPR	1	marcador PAPR
RESERVED_4	16	reservado para uso futuro
CRC32	32	CRC32
Total	228	

Fig. 113

Valor	Tipo de PLP
000	PLP común
001	PLP de datos agrupados
010	PLP de datos normales
011	PLP de datos empaquetados
100 a 111	Reservado para uso futuro

Fig. 114

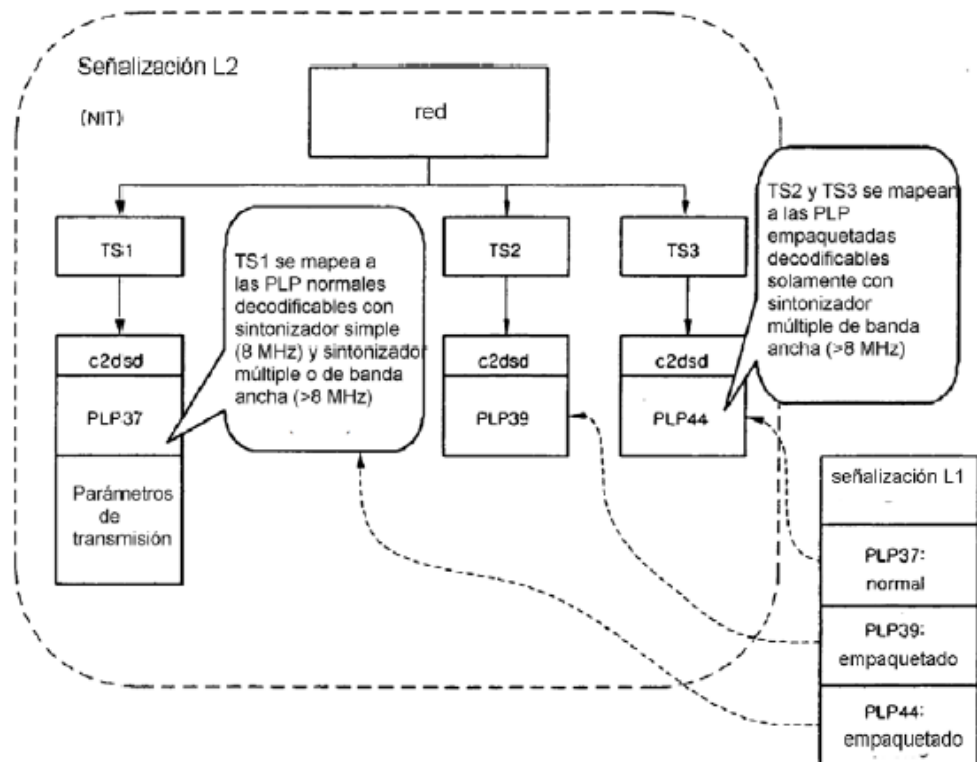


Fig. 115

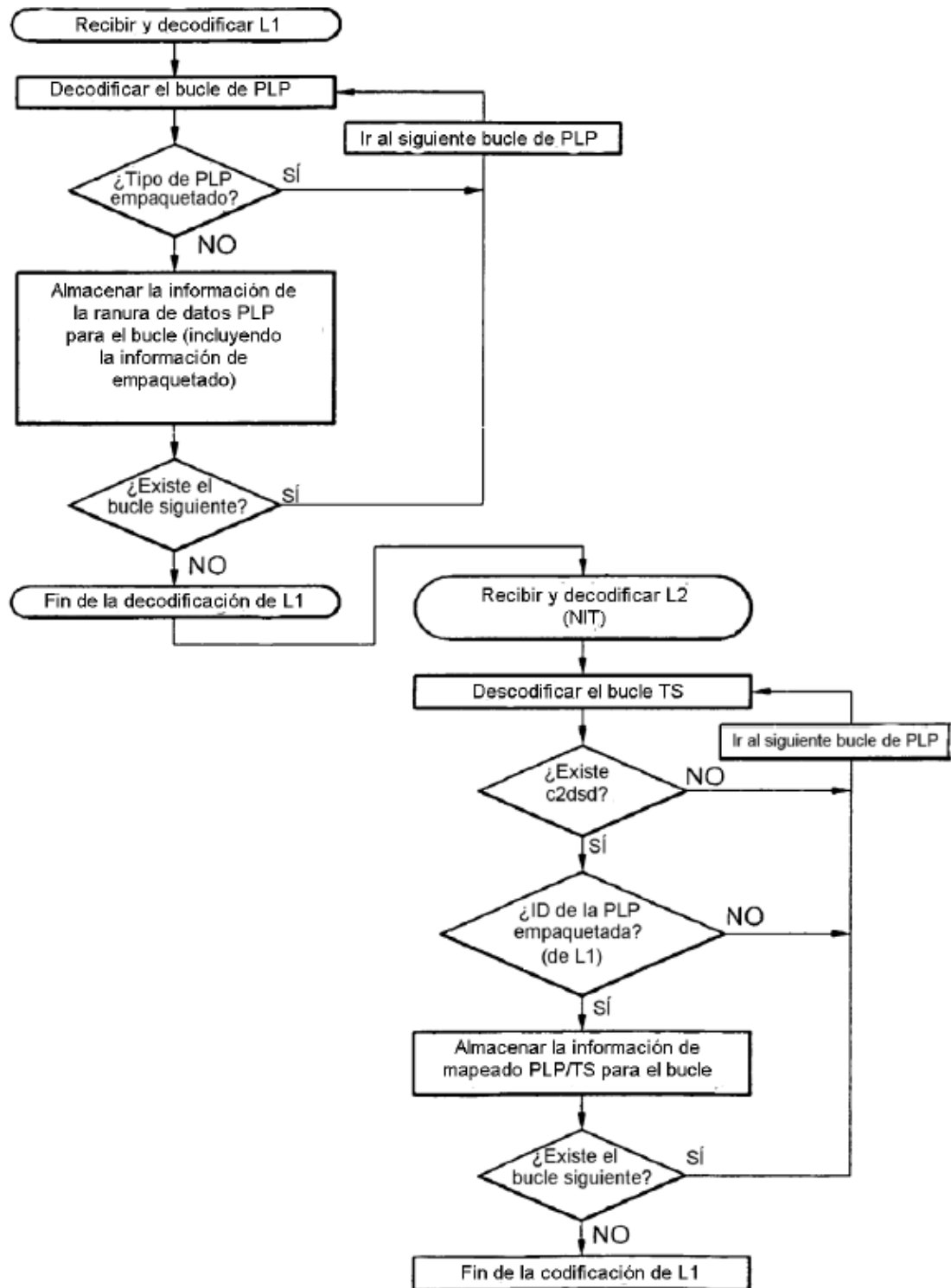


Fig. 116

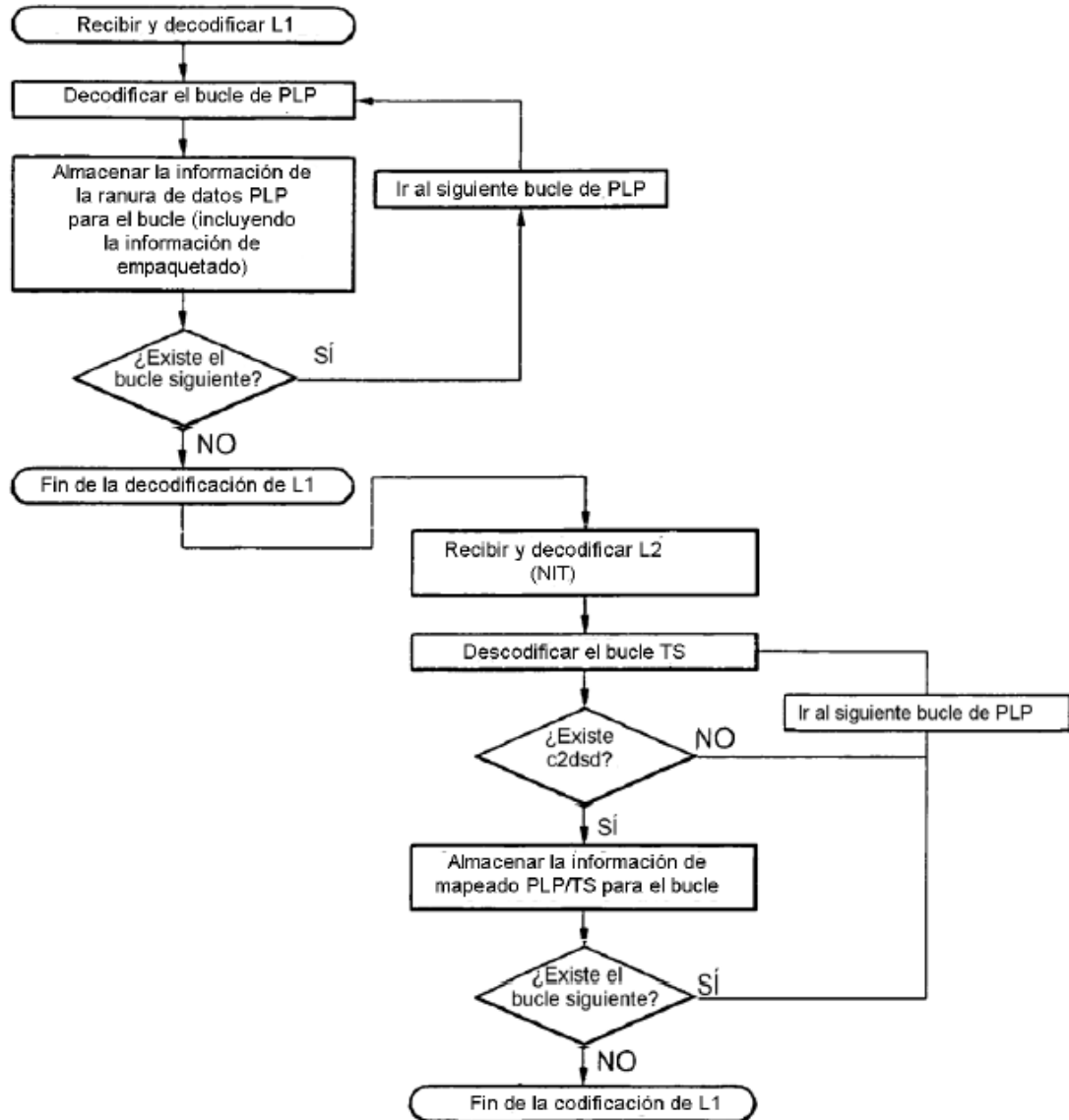


Fig. 117

Sintaxis	Número de bits	Identificador
C2_delivery_system_descriptor() {		
descriptor_tag	8	uimsbf
descriptor_length	8	uimsbf
descriptor_tag_extension	8	uimsbf
plp_id	8	uimsbf
C2_system_id	16	uimsbf
if (descriptor_length > 5){		
C2_System_tuning_frequency	32	bslbf
active OFDM symbol duration	3	bslbf
guard_interval	3	bslbf
reserved	2	bslbf
}		
}		

Fig. 118

Active_OFDM_symbol_duration	Descripción
000	448 μ s (modo 4K FFT para un ancho de banda de 8 MHz de sistemas CATV)
001	597 μ s (modo 4K FFT para un ancho de banda de 6 MHz de sistemas CATV)
010 a 111	reservado para uso futuro

Fig. 119

guard_interval	Valores del intervalo de guarda
000	1/128
001	1/64
010 a 111	reservado para uso futuro

Fig. 120

Campo	Bits	Descripción
NETWORK_ID	16	ID de la red DVB-C2 actual
C2_SYSTEM_ID	16	ID del sistema C2 dentro de la red DVB-C2
C2_SIGNAL_START_FREQUENCY	32	frecuencia de inicio de la señal C2
GI	2	modo del intervalo de guarda
LI_PART2_CHANGE_COUNTER	8	número de cuadros C2 por delante que la parte 2 de L1 cambiará excepto en PLP_START (unidad = 1 cuadro C2)
NUM_CHBON	3	número de canales adheridos
NUM_DSLICE	8	número de segmentos de datos
NUM_NOTCH	4	número de bandas de supresión
For i=0NUM_DSLICE-{		
DSLICE_ID	8	identificador de del segmento de datos
DSLICE_TUNE_POS	12(11)	Posición de sintonía para el segmento de datos actual (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente)
DSLICE_OFFSET_LEFT	9(8)	Desplazamiento del borde izquierdo del segmento de datos desde la posición de sintonía (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente), firmado
DSLICE_OFFSET_RIGHT	9(8)	Desplazamiento del borde derecho del segmento de datos desde la posición de sintonía (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente), firmado
DSLICE_TI_DEPTH	2	profundidad del entrelazador de tiempos
DSLICE_TYPE	1	tipo de segmento de datos (tipo 1, tipo 2)*
DSLICE_CONST_FLAG	1	indicador para segmento de datos constante
DSLICE_NOTCH_FLAG	1	indicador de supresión contigua
DSLICE_NUM_PLP	8	número de PLP en el segmento de datos actual
For i=0DSLICE_NUM_PLP-1 {		
PLP_ID	8	identificador de la PLP
PLP_BUNDLED_FLAG	1	marcador de PLP empaquetado
PLP_TYPE	2	tipo de PLP (común/ /datos agrupados/datos normales)
PLP_PAYLOAD_TYPE	5	tipo de contenido útil de la PLP (GFPS, GCS, GSE, TS, RFU)
iPLP_TYPE='00' o '01' {		
PLP_GROUP_ID	8	identificador del grupo de PLP
}		
IDSLICE_TYPE=='0'{		
PLP_START	14	dirección de inicio de la PLP (unidad = 1 célula OFDM)
PLP_MODCOD	5	modulación/decodificación de la PLP
}		
PSI/SI_REPROCESSING	1	marcador de reprocesamiento PSP/SI
RESERVED_1	8	reservado para uso futuro
}		
RESERVED_2	8	reservado para uso futuro
}		
For i=0..NUM_NOTCH-1 {		
NOTCH_START	12(11)	Comienzo de la banda de supresión dentro de un canal de C2 (Máx. 64 MHz) (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente)
NOTCH_WIDTH	9 (8)	Ancho de la banda de supresión (Máx. 7,61 MHz) (unidad = 12 ó 24 células OFDM para GI 1/64, 1/128, respectivamente)
RESERVED_3	8	reservado para uso futuro
}		
PAPR	1	marcador PAPR
RESERVED_4	16	reservado para uso futuro
CRC32	32	CRC32
Total	227	

Fig. 121

