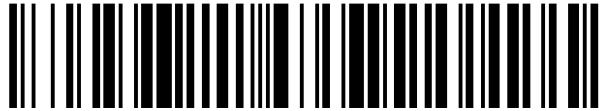


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 992 340**

51 Int. Cl.:

H04L 1/00

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **17.12.2014** **E 22208501 (1)**

97 Fecha y número de publicación de la concesión europea: **18.09.2024** **EP 4203359**

54 Título: **Método y dispositivo de recepción de datos, y método y dispositivo de envío de datos**

30 Prioridad:

17.12.2013 WO PCT/CN2013/089697
28.12.2013 WO PCT/CN2013/090803

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
11.12.2024

73 Titular/es:

HUAWEI TECHNOLOGIES CO., LTD. (100.0%)
Huawei Administration Building Bantian
Longgang District Shenzhen, Guangdong 518129,
CN

72 Inventor/es:

YANG, WENBIN;
WANG, TONGTONG y
WANG, XINYUAN

74 Agente/Representante:

ELZABURU, S.L.P

ES 2 992 340 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

Método y dispositivo de recepción de datos, y método y dispositivo de envío de datos

5 CAMPO TÉCNICO

La presente invención está relacionada con el campo de comunicaciones, y en particular, con un método y un dispositivo de recepción de datos, y un método y un dispositivo de envío de datos.

10 ANTECEDENTES

Con el desarrollo de internet, el tráfico de una red troncal de telecomunicaciones está creciendo rápidamente del 50 % al 80 % cada año. Al comienzo del año 2011, el grupo de trabajo IEEE 802.3 empezó a recoger un requisito de desarrollo de ancho de banda tras una interfaz de Ethernet 100GE. Con relación al desarrollo de ancho de banda de red en el futuro, el grupo de trabajo IEEE 802.3 consideró que el tráfico de red en el año 2015 podría alcanzar 10 veces el del año 2010. Según un análisis preliminar, habría dos tasas de 400GE/1TE para una interfaz Ethernet en el futuro, y tales requisitos aparecerían y comenzarían a aplicarse en el año 2015 y el año 2020, respectivamente.

20 Conforme aumenta la tasa de transmisión, aumenta la pérdida de alta frecuencia de una señal en un enlace de transmisión de alta velocidad; por lo tanto, la interferencia entre símbolos afecta un indicador tal como calidad de señal o una tasa de error de bits. Sin embargo, actualmente, un ecualizador de retroinformación de decisión (nombre completo en inglés: Decision Feedback Equalizer, inglés acrónimo: DFE) se usa universalmente en un lado de recepción de una interfaz de alta velocidad, que incorpora distorsión provocada por ruido blanco en el enlace en forma de ráfaga de errores. Por lo tanto, la industria comienza a estudiar cómo realizar corrección de errores en un error de bits en Ethernet usando un algoritmo FEC.

En la técnica anterior, un entrelazador (inglés: Interleaver) en un dispositivo de envío escribe por fila y luego lee por columna una sección de datos en la que se ha realizado codificación FEC. El dispositivo de envío entonces envía, a un dispositivo de recepción, los datos leídos por columna. Por ejemplo, cuando datos 1 en los que se ha realizado la codificación FEC incluyen unidades de datos U0 a U4, y datos 2 en los que se ha realizado la codificación FEC incluyen unidades de datos U5 a U9, las 10 unidades de datos se escriben por fila, y se establece que cada fila incluye cinco unidades de datos, y entonces los datos escritos se muestran en la Tabla 1.

35 Tabla 1

U4	U3	U2	U1	U0
U9	U8	U7	U6	U5

Los datos mostrados en la Tabla 1 se leen por columna, y entonces se pueden obtener los datos mostrados en la Tabla 2.

40 Tabla 2

U9	U4	U8	U3	U7	U2	U6	U1	U5	U0
----	----	----	----	----	----	----	----	----	----

Los datos leídos por columna se envían al dispositivo de recepción. El dispositivo de recepción realiza una operación inversa correspondiente al dispositivo de envío en los datos recibidos, y en un caso normal, se pueden obtener de nuevo los datos mostrados en la Tabla 1.

50 Durante un proceso de transmisión, puede ocurrir un error de bits en los datos leídos por columna y mostrados en la Tabla 3, por ejemplo, un error de bits ocurre en las tres unidades de datos U6, U2 y U7 sucesivas. El dispositivo de recepción restaura un orden de disposición de las unidades de datos en la Tabla 3 al mostrado en la Tabla 1. Por lo tanto, las tres unidades de datos sucesivas U6, U2 y U7 ya no son sucesivas. En otras palabras, disminuye la probabilidad de que ocurran múltiples códigos de error en una palabra de código. De esta manera, la presión de corrección de errores de una unidad de FEC en el dispositivo de recepción se puede reducir hasta cierto punto.

55 Sin embargo, en el método anterior, tiene que realizarse este tipo de manera especial de entrelazar de escritura por fila y lectura por columna. En esta manera de entrelazar, una sección de datos tiene que escribirse antes de que la sección de datos comience a leerse, lo que genera un obvio retraso. El documento US20040258167A1 describe una operación de entrelazado que puede codificar (permutar) un flujo de datos, o cada dimensión (conjunto de símbolos (a, b, c,...)) en un flujo de datos, inmediatamente después de la codificación FEC o la multiplexación de dimensiones del flujo de datos. Se pueden combinar ráfagas de errores con los datos permutados antes, durante o después de la transmisión. Un desentrelazador reordena los símbolos recibidos y, en el proceso, difunde (separa) las ráfagas de errores. Además, el uso del entrelazado y desentrelazado multidimensional puede equilibrar la SNR en cada canal.

5 Distribuir los errores y/o equilibrar la SNR puede evitar que las ráfagas abrumen al decodificador FEC o a un decodificador FEC en cualquier canal. En un ejemplo, se puede utilizar el entrelazado y el desentrelazado para codificar datos a través de pares de cables trenzados de Ethernet. En otro ejemplo, se puede utilizar el entrelazado y el desintercalado para codificar datos o información difundida a través de canales de telecomunicaciones inalámbricas (por ejemplo, canales de radiofrecuencia, canales de múltiples antenas, etc.).

10 El documento WO2013066739A1 divulga un método para generar una unidad de datos de capa física (PHY) para la transmisión por medio de un canal de comunicación, la unidad de datos se genera para incluir un primer campo de entrenamiento largo cuando la unidad de datos se va a transmitir en un modo normal. La unidad de datos se genera para incluir un segundo campo de entrenamiento largo cuando la unidad de datos se va a transmitir en un modo de ancho de banda bajo. El primer campo de entrenamiento y el segundo campo de entrenamiento están configurados de manera que un dispositivo receptor pueda detectar automáticamente si la unidad de datos corresponde al modo de ancho de banda bajo o al modo normal.

15 El documento US20130216221A1 divulga un sistema y un método con una modulación codificada entrelazada por bits modificada con decodificación iterativa (BICM-ID).

COMPENDIO

20 Un objetivo de realizaciones de la presente invención es proporcionar un método de envío de datos y un método de recepción de datos, y un dispositivo de envío correspondiente y un dispositivo de recepción correspondiente, para corregir un error de bits generado durante un proceso de transmisión de datos.

25 La invención está definida por las reivindicaciones independientes. Las reivindicaciones dependientes definen realizaciones ventajosas.

30 Una ventaja según un aspecto de la presente invención radica en que, un dispositivo de recepción puede enviar, a unidades de FEC diferentes del dispositivo de recepción, unidades de datos en las que ocurre un error; por lo tanto, se puede impedir en gran medida un caso en el que se realiza corrección de errores usando únicamente una unidad de FEC. En este caso, para una unidad de FEC, disminuye el número de errores de bits recibidos por la unidad de FEC. En otras palabras, aumenta la probabilidad de que la unidad de FEC realice corrección de errores con éxito. En global, cuando aumentan las tasas de éxito de corrección de errores de una o más unidades de FEC en el dispositivo de recepción, se mejora la capacidad de corrección de errores del dispositivo de recepción. Adicionalmente, en esta realización, no es necesaria una manera de entrelazar de escritura por fila y lectura por columna; por lo tanto, no se genera retraso.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

40 La FIGURA 1 es un diagrama esquemático de envío de datos según una realización de la presente invención; la FIGURA 2 es un diagrama esquemático de envío de datos según una realización de la presente invención; la FIGURA 3 es un diagrama esquemático de recepción de datos según una realización de la presente invención; la FIGURA 4 es un diagrama esquemático de recepción de datos según una realización de la presente invención; la FIGURA 5 es un diagrama esquemático de envío de datos según una realización de la presente invención; la FIGURA 6 es un diagrama esquemático de recepción de datos según una realización de la presente invención; la FIGURA 7 es un diagrama esquemático de envío de datos según una realización de la presente invención; la FIGURA 8 es un diagrama esquemático de recepción de datos según una realización de la presente invención; la FIGURA 9 es un diagrama esquemático de un aparato de envío según una realización de la presente invención; y la FIGURA 10 es un diagrama esquemático de un aparato de recepción según una realización de la presente invención. la FIGURA 11 es un diagrama esquemático de realización de un tipo de entrelazado según una realización de la presente invención; la FIGURA 12 es un diagrama esquemático de realización de otro tipo de entrelazado según una realización de la presente invención; la FIGURA 13 es un diagrama esquemático de realización de entrelazado basado en un resultado del entrelazado mostrado en la FIGURA 11 según una realización de la presente invención; la FIGURA 14 es un diagrama esquemático de realización de otro tipo de entrelazado adicional basado en un resultado del entrelazado mostrado en la FIGURA 11 según una realización de la presente invención; y la FIGURA 15 es un diagrama esquemático de seguir realizando entrelazado basado en un resultado del entrelazado mostrado en la FIGURA 12 según una realización de la presente invención.

DESCRIPCIÓN DE REALIZACIONES

A continuación se describe la presente invención en detalle con referencia a los dibujos adjuntos realizaciones específicas. Sin embargo, se debe observar que, las siguientes realizaciones son únicamente ejemplos dados para facilitar el entendimiento de las soluciones técnicas, y no pretenden limitar la presente invención.

Una realización de la presente invención proporciona un método de envío de datos. El método es realizado por un dispositivo de envío, y el dispositivo de envío incluye al menos dos unidades de corrección de errores hacia delante (nombre completo en inglés: Forward Error Correction, acrónimo en inglés: FEC), por ejemplo, una FECa0 y una FECa1 que se muestran en la FIGURA 1. Opcionalmente, el dispositivo de envío puede incluir además una u otras más unidades de FEC.

En esta realización, cada unidad de FEC puede tener al menos un canal, y el canal se usa para tener como salida datos en los que se ha realizado codificación FEC por una unidad de FEC conectada a este canal. Por ejemplo, la FECa0 en la FIGURA 1 incluye canales La0 a La3, y la FECa1 incluye canales La4 a La7. Los cuatro canales La0 a La3 se usan para tener como salida un flujo de datos codificado por la FECa0, y los cuatro canales La4 a La7 se usan para tener como salida un flujo de datos codificado por la FECa1. Cuando al menos dos canales se conectan a una unidad de FEC, al menos dos canales pueden sacar datos al mismo tiempo, o algunos de al menos dos canales pueden sacar datos. Un flujo de datos sacado por cada canal puede incluir un identificador de canal del canal, por ejemplo, un flujo de datos a0 sacado por el canal La0 incluye un identificador de canal del canal La0, y un flujo de datos a4 sacado por el canal La4 incluye un identificador de canal del canal La4. Opcionalmente, un identificador de canal identifica de manera única un canal en el dispositivo de envío.

Las reglas de codificación usadas por unidades de FEC diferentes pueden ser iguales o diferentes. Adicionalmente, reglas de codificación FEC para flujos de datos sacados por diferentes canales de una misma unidad de FEC pueden ser iguales o diferentes. Por ejemplo, una regla de codificación FEC para el flujo de datos a0 puede ser igual o diferente a una regla de codificación FEC para el flujo de datos a4. La regla de codificación para el flujo de datos a0 puede ser igual o diferente a una regla de codificación para un flujo de datos a1, y el flujo de datos a1 se refiere a un flujo de datos sacado del canal La1.

En esta realización, como se muestra en la FIGURA 1, el flujo de datos a0 sacado por el canal La0 conectado a la FECa0 incluye unidades de datos 00 a 06, el flujo de datos a4 sacado por el canal La4 conectado a la FECa1 incluye unidades de datos 40 a 46, y en las 14 unidades de datos, una longitud de cada unidad de datos puede ser un bit (inglés: bit) o múltiples bits. Preferiblemente, las longitudes de las unidades de datos deben ser iguales. Cuando una unidad de datos incluye al menos dos bits, la unidad de datos puede denominarse símbolo (inglés: symbol).

En el flujo de datos a0 y el flujo de datos a4 se realiza entrelazado, para obtener un flujo de datos i0. Se debe observar que, el flujo de datos i0 puede ser un flujo de datos o dos o más flujos de datos.

Cuando el flujo de datos i0 es un flujo de datos, como se muestra en la FIGURA 1, el flujo de datos i0 obtenido tras el entrelazado incluye las siete unidades de datos: las unidades de datos 00 a 06, e incluye además las siete unidades de datos: las unidades de datos 40 a 46, y un orden de las 14 unidades de datos se muestra en la FIGURA 1. Como se muestra en la FIGURA 1, dos unidades de datos adyacentes en el flujo de datos i0 son de diferentes flujos de datos. Preferiblemente, en el flujo de datos i0, un orden de las unidades de datos que son de un mismo flujo de datos es igual que el de estas unidades de datos en un flujo de datos original, por ejemplo, como se muestra en la FIGURA 1, en el flujo de datos i0, un orden de las unidades de datos 00 a 06 que son del flujo de datos a0 es igual que el de las siete unidades de datos 00 a 06 en el flujo de datos a0.

Cuando el flujo de datos i0 incluye dos flujos de datos, como se muestra en la FIGURA 2, el flujo de datos i0 obtenido tras el entrelazado es un flujo de datos i00 y un flujo de datos i01. El flujo de datos i00 incluye la unidad de datos 00 y las unidades de datos 41, 02, 43, 04, 45 y 06 que se disponen en orden, y el flujo de datos i01 incluye las unidades de datos 40, 01, 42, 03, 44, 05 y 46 que se disponen en orden.

Cuando el flujo de datos i0 son dos flujos de datos, el dispositivo de envío puede no seguir realizando el entrelazado en los dos flujos de datos, y puede enviar el flujo de datos i00 y el flujo de datos i01 como dos flujos de datos a un dispositivo de recepción.

Cuando el entrelazado se realiza en un flujo de datos 1 y un flujo de datos 2, para obtener un flujo de datos 3, en el flujo de datos 3, un orden de disposición de unidades de datos que son del flujo de datos 1 debe ser igual que el de estas unidades de datos en el flujo de datos 1, y un orden de disposición de unidades de datos que son del flujo de datos 2 debe ser igual que el de estas unidades de datos en el flujo de datos 2. Usando la FIGURA 1 como ejemplo, las unidades de datos 00, 01 y 02 son del flujo de datos a0, y un orden de disposición de las tres unidades de datos en el flujo de datos a0 es que la unidad de datos 01 está detrás de la unidad de datos 00, y la unidad de datos 02 está detrás de la unidad de datos 01. En el flujo de datos i0 obtenido tras el entrelazado, la unidad de datos 01 todavía está detrás de la unidad de datos 00, y la unidad de datos 02 todavía está detrás de la unidad de datos 01.

Adicionalmente, se debe observar que, para el entrelazado, el entrelazado puede realizarse en función de una unidad de datos o múltiples unidades de datos. Métodos de entrelazado mostrados en la FIGURA 1 y FIGURA 2 se basan en una unidad de datos, y en un flujo de datos después del entrelazado, dos unidades de datos adyacentes son de diferentes flujos de datos, por ejemplo, la unidad de datos 00 y la unidad de datos 40 son de diferentes flujos de datos. Si el entrelazado se realiza basado en dos unidades de datos, usando que el entrelazado se realiza en las primeras seis unidades de datos en el flujo de datos a0 y las primeras seis unidades de datos en el flujo de datos a4 en la FIGURA 1 por ejemplo, un orden de disposición de las unidades de datos en un flujo de datos obtenido tras el entrelazado puede ser las unidades de datos 00-01-40-41-02-03-42-43-04-05-44-45.

Una operación de entrelazado en esta realización es realizada por una unidad de entrelazado, y la unidad de entrelazado puede ser uno o más chips de hardware o procesadores, o algunas funciones de lógica en un chip de hardware o procesador.

En esta realización, una unidad de entrelazado de un dispositivo de envío realiza entrelazado en al menos dos flujos de datos codificados por unidades de FEC diferentes, y una unidad de envío en el dispositivo de envío envía datos obtenidos a través del entrelazado a un dispositivo de recepción. De esta manera, el dispositivo de recepción puede realizar desentrelazado en un flujo de datos después del entrelazado, y enviar al menos flujos de datos obtenidos a través del desentrelazado a unidades de FEC diferentes para decodificar. Cuando ocurre un error de bits durante un proceso de transmisión del flujo de datos obtenido a través del entrelazado, los flujos de datos obtenidos tras el desentrelazado se envían a las unidades de FEC diferentes para decodificar. Por lo tanto, para una unidad de FEC, disminuye el número de errores de bits recibidos por la unidad de FEC. En otras palabras, aumenta la probabilidad de que la unidad de FEC realice corrección de errores con éxito. En global, cuando aumentan las tasas de éxito de corrección de errores de una o más unidades de FEC en el dispositivo de recepción, se mejora la capacidad de corrección de errores del dispositivo de recepción. Adicionalmente, en esta realización, no es necesaria una manera de entrelazar de escritura por fila y lectura por columna; por lo tanto, no se genera retraso. Adicionalmente, un entrelazador que implementa la manera de entrelazar de escribir por fila y leer por columna es de diseño complejo y consume mucha potencia. En esta realización, este tipo de especial entrelazador no es necesario; por lo tanto, la implementación es simple y se reduce el consumo de potencia de un dispositivo. En resumen, el método de envío de datos en esta realización ayuda a implementar un método de corrección de errores simple, que ahorra energía y es eficiente.

Una realización de la presente invención proporciona un método de recepción de datos. El método es realizado por un dispositivo de recepción, y el dispositivo de envío incluye al menos dos unidades de FEC, por ejemplo, una FECb1 y una FECb2 que se muestran en la FIGURA 3. Opcionalmente, el dispositivo de envío puede incluir además una u otras más unidades de FEC.

Una unidad de recepción en el dispositivo de recepción recibe un primer flujo de datos de un dispositivo de envío.

El primer flujo de datos en esta realización puede ser un flujo de datos o dos o más flujos de datos.

Como se muestra en la FIGURA 3, cuando el primer flujo de datos es el flujo de datos i0 mostrado en la FIGURA 1, el flujo de datos i0 es desentrelazado por una unidad de desentrelazado en el dispositivo de recepción hasta un flujo de datos a0 y un flujo de datos a4. En un caso en el que no ocurre fallo, el flujo de datos a0 en la FIGURA 3 es igual que el flujo de datos a0 en la FIGURA 1, y el flujo de datos a4 en la FIGURA 3 es igual que el flujo de datos a4 en la FIGURA 1. El flujo de datos a0 en la FIGURA 1 incluye el identificador de canal del canal La0, y el flujo de datos a4 incluye el identificador de canal del canal La4; por lo tanto, el flujo de datos a0 en la FIGURA 3 también debe incluir el identificador de canal del canal La0, y el flujo de datos a4 en la FIGURA 3 también debe incluir el identificador de canal del canal La4.

Una unidad de determinación en el dispositivo de recepción determina que el flujo de datos a0 incluye el identificador de canal del canal La0 y el flujo de datos a4 incluye el identificador de canal del canal La4.

Cuando el dispositivo de recepción puede determinar que el flujo de datos a0 incluye el identificador de canal del canal La0 y el flujo de datos a4 incluye el identificador de canal del canal La4, indica que el flujo de datos a0 obtenido a través del desentrelazado por el dispositivo de recepción incluye el identificador de canal de la La0, y un flujo de datos a1 obtenido a través del desentrelazado incluye un identificador de canal de un La1. Además, el dispositivo de recepción puede determinar que el flujo de datos i0 es un flujo de datos que se obtiene tras procesamiento de entrelazado. Esto es porque, si el flujo de datos i0 no se adquiere al realizar procesamiento de entrelazado, los flujos de datos obtenidos después de realizar forzosamente el desentrelazado en el flujo de datos i0 pueden estar desordenados, y el dispositivo de recepción no puede determinar el identificador de canal del La0 o el identificador de canal del La1 de los flujos de datos desordenados. Adicionalmente, cuando un canal se conecta a una unidad de FEC, se puede determinar que un flujo de datos que incluye un identificador de canal del canal es de la unidad de FEC, esto es, se ha realizado codificación FEC en el flujo de datos.

El dispositivo de recepción determina, según el identificador de canal del canal La0 y una correspondencia 1, un canal en el dispositivo de recepción y correspondiente al canal La0. En esta realización, la correspondencia 1 incluye una

correspondencia entre el identificador de canal del canal La0 y un identificador de canal de un canal Lb1. Por lo tanto, el dispositivo de recepción determina que el canal Lb1 es un canal correspondiente al canal La0, determinando de ese modo que el flujo de datos a0 tiene que ser enviado usando el canal Lb1. Como se muestra en la FIGURA 3, el flujo de datos a0 se envía, usando el canal Lb1, a una FECb0 conectado al canal Lb1.

5 De manera similar, el dispositivo de recepción determina, según el identificador de canal del canal La4 y una correspondencia 2, un canal en el dispositivo de recepción y correspondiente al canal La4. En esta realización, la correspondencia 2 incluye una correspondencia entre el identificador de canal del canal La4 y un identificador de canal de un canal Lb7. Por lo tanto, el dispositivo de recepción determina que el canal Lb7 es un canal correspondiente al canal La0, determinando de ese modo que el flujo de datos a4 tiene que ser enviado usando el canal Lb7. Como se muestra en la FIGURA 3, el flujo de datos a4 se envía, usando el canal Lb7, a una FECb1 conectada al canal Lb7.

15 Como se muestra en la FIGURA 3, cuando ocurre un error en unidades de datos 43, 04, 44, 05 y 45 en el flujo de datos i0 recibidos por el dispositivo de recepción, el dispositivo de recepción desentrelaza el flujo de datos i0 usando la unidad de desentrelazado, para obtener los flujos de datos a0 y a4 que se muestran en la FIGURA 3. Por lo tanto, las unidades de datos 04 y 05 en las que ocurre el error se envían a la FECb0, y las unidades de datos 43, 44, y 45 en las que ocurre el error se envían a la FECb1. Las unidades de datos en las que ocurre el error se envían a unidades de FEC diferentes. Por lo tanto, para una unidad de FEC, disminuye el número de errores de bits recibidos por la unidad de FEC. En otras palabras, aumenta la probabilidad de que la unidad de FEC realice corrección de errores con éxito. En global, cuando aumentan las tasas de éxito de corrección de errores de una o más unidades de FEC en el dispositivo de recepción, se mejora la capacidad de corrección de errores del dispositivo de recepción. Adicionalmente, en esta realización, no es necesaria una manera de entrelazar de escritura por fila y lectura por columna; por lo tanto, no se genera retraso. Adicionalmente, un entrelazador que implementa la manera de entrelazar de escribir por fila y leer por columna es de diseño complejo y consume mucha potencia. En esta realización, este tipo de especial entrelazador no es necesario; por lo tanto, la implementación es simple y se reduce el consumo de potencia de un dispositivo.

20 Cuando el primer flujo de datos son los flujos de datos i00 y i01 mostrados en la FIGURA 2, como se muestra en la FIGURA 4, los flujos de datos i00 y i01 son desentrelazados por la unidad de desentrelazado en el dispositivo de recepción hasta el flujo de datos a0 y el flujo de datos a4.

35 Operaciones que son realizadas por el dispositivo de recepción tras obtener el flujo de datos a0 y el flujo de datos a4 son iguales que las de la realización correspondiente a la FIGURA 3, esto es, determinar el identificador de canal del canal La0 a partir del flujo de datos a0, y determinar el identificador de canal del canal La4 a partir del flujo de datos a4; y entonces, determinar, según el identificador de canal del canal La0, que el canal Lb1 es un canal correspondiente al canal La0, y determinar, según el identificador de canal del canal La4, que el canal Lb7 es un canal correspondiente al canal La4. De esta manera, el dispositivo de recepción puede enviar el flujo de datos a0 a la FECb0 usando el canal Lb1 para procesar, y enviar el flujo de datos a1 a la FECb1 usando el canal Lb7 para procesar.

40 Una realización de la presente invención incluye un proceso durante el que un dispositivo de envío envía datos a un dispositivo de recepción, y el dispositivo de recepción realiza una recepción correspondiente.

45 Como se muestra en la FIGURA 5, el dispositivo de envío incluye dos unidades de FEC, que son una FECa0 y una FECa1, respectivamente. En la FIGURA 5, la FECa0 incluye canales La0 a La3, y la FECa1 incluye canales La4 a La7. Los cuatro canales La0 a La3 se usan para tener como salida un flujo de datos codificado por la FECa0, y los cuatro canales La4 a La7 se usan para tener como salida un flujo de datos codificado por la FECa1.

50 Como se muestra en la FIGURA 5, el canal La0 tiene como salida un flujo de datos a00, el canal La2 tiene como salida un flujo de datos a20, el canal La4 tiene como salida un flujo de datos a40, y el canal La6 tiene como salida un flujo de datos a60. El flujo de datos a00 incluye cinco unidades de datos: unidades de datos 00 a 04 que se disponen en orden, el flujo de datos a20 incluye cinco unidades de datos: unidades de datos 20 a 24 que se disponen en orden, el flujo de datos a40 incluye cinco unidades de datos: unidades de datos 40 a 44 que se disponen en orden, y el flujo de datos a60 incluye cinco unidades de datos: unidades de datos 60 a 64 que se disponen en orden. El flujo de datos a00 incluye un identificador de canal del canal La0, el flujo de datos a20 incluye un identificador de canal del canal La2, el flujo de datos a40 incluye un identificador de canal del canal La4, y el flujo de datos a60 incluye un identificador de canal del canal La6.

55 El dispositivo de envío realiza primer entrelazado usando una unidad de entrelazado, donde el primer entrelazado específicamente incluye: realizar entrelazado en el flujo de datos a00 y el flujo de datos a40, para obtener un flujo de datos i0. Cuando cada unidad de datos en el flujo de datos a00 y el flujo de datos a40 incluye al menos dos bits, puede considerarse que el primer entrelazado se realiza según una primera unidad de datos. Cuando cada unidad de datos en el flujo de datos a00 y el flujo de datos a40 tiene únicamente un bit, puede considerar que el primer entrelazado se realiza según un bit.

60 El dispositivo de envío realiza segundo entrelazado usando la unidad de entrelazado, donde el segundo entrelazado específicamente incluye: realizar entrelazado en el flujo de datos a20 y el flujo de datos a60, para obtener un flujo de

datos i1. Cuando cada unidad de datos en el flujo de datos a20 y el flujo de datos a60 incluye al menos dos bits, puede considerarse que el segundo entrelazado se realiza según una segunda unidad de datos. Cuando cada unidad de datos en el flujo de datos a20 y el flujo de datos a60 tiene únicamente un bit, puede considerarse que el segundo entrelazado se realiza según un bit.

5 La unidad de entrelazado realiza tercer entrelazado en el flujo de datos i0 y el flujo de datos i1, para obtener un flujo de datos i2.

10 El flujo de datos i2 se envía al dispositivo de recepción.

15 El tercer entrelazado mostrado en la FIGURA 5 es un caso en el que el entrelazado se realiza usando cada unidad de datos como unidad básica, esto es, el entrelazado se realiza según una unidad de datos. Cada unidad de datos puede ser únicamente un bit, o puede incluir al menos dos bits. Cuando cada unidad de datos incluye al menos dos bits, en un caso en el que el entrelazado se realiza según una unidad de datos, un orden de disposición de bits en una unidad de datos en un flujo de datos antes del entrelazado no se debe cambiar debido al hecho de que se realiza el entrelazado. Esto es porque una unidad de datos es una unidad básica del entrelazado.

20 Opcionalmente, cuando cada unidad de datos obtenidos a través del primer entrelazado incluye múltiples bits, para el tercer entrelazado, el entrelazado también puede realizarse según un bit.

25 El dispositivo de envío mostrado en la FIGURA 5 incluye dos unidades de FEC; sin embargo, en una aplicación real, un dispositivo de envío puede incluir más unidades de FEC, por ejemplo, un dispositivo de envío puede incluir cuatro unidades de FEC. Cuando un dispositivo de envío incluye n unidades de FEC, el dispositivo de envío puede realizar el entrelazado en n flujos de datos que son de las n unidades de FEC, donde los n flujos de datos son de unidades de FEC diferentes. Por ejemplo, cuando el dispositivo de envío incluye cuatro unidades de FEC: la FECa0, la FECa1, una FECa2, y una FECa3, y la FECa0 tiene como salida un flujo de datos a0, la FECa1 tiene como salida un flujo de datos a1, la FECa2 tiene como salida un flujo de datos a2, y la FECa3 tiene como salida un flujo de datos a3, el dispositivo de envío puede realizar el entrelazado en los cuatro flujos de datos: los flujos de datos a0 a a3.

30 Como se muestra en la FIGURA 6, después de que el dispositivo de recepción recibe el flujo de datos i2, se realiza primer desentrelazado en el flujo de datos i2 usando la unidad de desentrelazado, para obtener el flujo de datos i0 y el flujo de datos i1.

35 El dispositivo de recepción entonces realiza segundo desentrelazado en el flujo de datos i0 usando la unidad de desentrelazado, y realiza tercer desentrelazado en el flujo de datos i1. El segundo desentrelazado específicamente incluye: realizar el desentrelazado en el flujo de datos i0, para obtener el flujo de datos a00 y el flujo de datos a40; y el tercer desentrelazado específicamente incluye: realizar el desentrelazado en el flujo de datos i1, para obtener el flujo de datos a20 y el flujo de datos a60.

40 Cuando el flujo de datos i0 se obtiene después de que el dispositivo de envío realiza el entrelazado según la primera unidad de datos, el segundo desentrelazado se refiere a realizar el desentrelazado según la primera unidad de datos. Cuando el flujo de datos i1 se obtiene después de que el dispositivo de envío realiza el entrelazado según la segunda unidad de datos, el tercer desentrelazado se refiere a realizar el desentrelazado según la segunda unidad de datos.

45 Cuando el flujo de datos i0 se obtiene después de que el dispositivo de envío realiza el entrelazado según un bit, para el segundo desentrelazado, el desentrelazado se realiza según un bit. Cuando el flujo de datos i1 se obtiene después de que el dispositivo de envío realiza el entrelazado según un bit, para el tercer desentrelazado, el desentrelazado se realiza según un bit.

50 El flujo de datos a00 incluye el identificador de canal del canal La0, el flujo de datos a20 incluye el identificador de canal del canal La2, el flujo de datos a40 incluye el identificador de canal del canal La4, y el flujo de datos a60 incluye el identificador de canal del canal La6. Por lo tanto, el dispositivo de recepción puede obtener los anteriores cuatro identificadores de canal según los anteriores cuatro flujos de datos que se obtienen a través del segundo desentrelazado y el tercer desentrelazado. Entonces, los correspondientes canales en el dispositivo de recepción se encuentran por separado según los anteriores cuatro identificadores de canal. Por ejemplo, se determina, según el identificador de canal del canal La0 y una correspondencia 0, que un canal Lb0 es un canal correspondiente al canal La0, y se usa para recibir el flujo de datos a00, donde la correspondencia 0 incluye una correspondencia entre el identificador de canal del canal La0 y un identificador de canal del canal Lb0. Según el principio anterior, el dispositivo de recepción determina, según el identificador de canal del canal La2 y una correspondencia 2, que un canal Lb2 es un canal correspondiente al canal La2, y se usa para recibir el flujo de datos a20; determina, según el identificador de canal del canal La4 y una correspondencia 4, que un canal Lb4 es un canal correspondiente al canal La4, y se usa para recibir el flujo de datos a40; y determina, según el identificador de canal del canal La6 y una correspondencia 6, que un canal Lb6 es un canal correspondiente al canal La6, y se usa para recibir el flujo de datos a60.

65 Después de eso, el dispositivo de recepción envía, a una FECb0 usando el canal Lb0, el flujo de datos a00 que se obtiene según el flujo de datos i0, para decodificación FEC, y envía, a una FECb1 usando el canal Lb4, el flujo de

datos a40 que se obtiene según el flujo de datos i0, para decodificación FEC; y envía, a la FECb0 usando el canal Lb2, el flujo de datos a20 que se obtiene según el flujo de datos i1, para decodificación FEC, y envía, a la FECb1 usando el canal Lb6, el flujo de datos a60 que se obtiene según el flujo de datos i1, para decodificación FEC.

5 Puede ocurrir un error durante un proceso de transmisión del flujo de datos i2 desde el dispositivo de envío al dispositivo de recepción. Por ejemplo, cuando ocurre un error en las unidades de datos 41, 61, 02 y 22 en el flujo de datos i2, después del primer desentrelazado, la unidad de datos 41 y la unidad de datos 02 se adjudican al flujo de datos i0, y la unidad de datos 61 y la unidad de datos 22 se adjudican al flujo de datos i1. Después del segundo desentrelazado y el tercer desentrelazado, la unidad de datos 02 se adjudica al flujo de datos a00, la unidad de datos 41 se adjudica al flujo de datos a40, la unidad de datos 21 se adjudica al flujo de datos a20, y la unidad de datos 61 se adjudica al flujo de datos a60. De esta manera, las cuatro unidades de datos en las que ocurre el error se adjudican a unidades de FEC diferentes para decodificación FEC. De esta manera, disminuye la probabilidad de que un gran número de errores de bits se adjudiquen a una misma unidad de FEC, y se pueden utilizar totalmente capacidades de corrección de errores de múltiples unidades de FEC en el dispositivo de recepción. Por lo tanto, una probabilidad de que el dispositivo de recepción corrija con éxito errores de bits aumenta en global.

Haciendo referencia a la FIGURA 11, una realización de la presente invención proporciona un método para realizar entrelazado. El método es realizado por un dispositivo de envío, el dispositivo de envío incluye cuatro unidades de corrección de errores hacia delante de FEC, que son una FEC0, una FEC1, una FEC2 y una FEC3, y cada unidad de FEC tiene como salida cuatro flujos de datos. Como se muestra en la FIGURA 11, la FEC0 incluye cuatro canales, esto es, un Lane0 a un Lane3, y cada canal tiene como salida un flujo de datos; y por lo tanto el Lane0 al Lane3 sacan flujos de datos L0 a L3. Los cuatro canales Lane4 a Lane7 de la FEC1 sacan flujos de datos L4 a L7, los cuatro canales Lane8 a Lane11 de la FEC2 sacan flujos de datos L8 a L11, y los cuatro canales Lane12 a Lane15 de la FEC3 sacan flujos de datos L12 a L15. Cada flujo de datos incluye un identificador de canal de un canal correspondiente al flujo de datos, por ejemplo, el L0 incluye un identificador del Lane0. Una manera específica para llevar un identificador de canal puede ser que múltiples paquetes o cada paquete en un flujo de datos incluye un identificador de canal de un canal a través del que se envía el flujo de datos.

De L0 a L15 sacan correspondientemente flujos de datos de símbolo (inglés: Symbol). Por ejemplo, los bits 000.0 a 000.9 en el Lane0 representan un Símbolo; y los bits 004.0 y 004.1 representan los primeros dos bits de un siguiente Símbolo, y subsiguientes bits no se muestran. Este tipo de flujo de datos que incluye múltiples Símbolos es un flujo de datos de símbolos. De manera similar, los flujos de datos del Lane1 al Lane15 también son flujos de datos de símbolos.

Antes del entrelazado, 000.0 a 000.9 en la FEC0 representan diez bits de un Symbol00, 001.0 a 001.9 representan diez bits de un Symbol01, 002.0 a 002.9 representan diez bits de un Symbol02 y 003.0 a 003.9 representan diez bits de un Symbol03. Las estructuras de flujos de datos sacados por la FEC1, la FEC2 y la FEC3 son iguales que la estructura de un flujo de datos sacado por la FEC0, y los flujos de datos son todos flujos de datos de símbolos, y cada símbolo incluye diez bits. Adicionalmente, un orden en el que un bit en cada Símbolo se saca es relevante para un número de secuencia de este bit, por ejemplo, primero se saca 000.0, luego se saca 000.1, luego se saca 000.2 después de 000.1, etc. Se puede entender que en un Símbolo, un bit cuyo número de secuencia es más pequeño se saca antes que un bit cuyo número de secuencia es mayor. Los datos en L1 se usan como ejemplo: 000.0 es menor que 000.1, y por lo tanto 000.0 se saca antes que 000.1; y 000.9 es menor que 004.0, y por lo tanto 000.9 se saca antes que 004.0. En otro flujo de datos, un principio para determinar un orden de salida es igual que el anterior.

45 A continuación, una manera específica del entrelazado se introduce según la FIGURA 11.

La primera ronda de datos de cuatro flujos de datos de la FEC0, esto es, 000.0, 001.0, 002.0 y 003.0, se usa como primera ronda de datos de cuatro flujos entrelazados EL0 a EL3 que se obtienen tras haber realizado el entrelazado, donde los flujos entrelazados son flujos de datos obtenidos tras haber realizado el entrelazado; la primera ronda de datos de cuatro flujos de la FEC1, esto es, 100.0, 101.0, 102.0 y 103.0, se usa como segunda ronda de unidades de datos del EL0 al EL3; la primera ronda de datos de cuatro flujos de la FEC2, esto es, 200.0, 201.0, 202.0 y 203.0, se usa como tercera ronda de datos del EL0 al EL3; y la primera ronda de datos de cuatro flujos de la FEC3, esto es, 300.0, 301.0, 302.0 y 303.0, se usa como cuarta ronda de datos del EL0 al EL3.

55 La segunda ronda de datos de cuatro flujos de datos de la FEC0, esto es, 000.1, 001.1, 002.1 y 003.1, se usa como primera ronda de datos de cuatro flujos entrelazados EL4 a EL7 que se obtienen tras haber realizado el entrelazado; la segunda ronda de datos de cuatro flujos de datos de la FEC1, esto es, 100.1, 101.1, 102.1 y 103.1, se usa como segunda ronda de unidades de datos del EL4 al EL7; la segunda ronda de datos de cuatro flujos de la FEC2, esto es, 200.1, 201.1, 202.1 y 203.1, se usa como tercera ronda de unidades de datos del EL4 al EL7; y la segunda ronda de unidades de datos de la FEC3, esto es, 300.1, 301.1, 302.1 y 303.1, se usan como cuarta ronda de unidades de datos del EL4 al EL7.

La tercera ronda de datos de cuatro flujos de datos de la FEC0, esto es, 000.2, 001.2, 002.2 y 003.2, se usa como primera ronda de datos de cuatro flujos entrelazados EL8 a EL11 que se obtienen tras haber realizado el entrelazado; la tercera ronda de datos de cuatro flujos de datos de la FEC1, esto es, 100.2, 101.2, 102.2 y 103.2, se usa como segunda ronda de datos del EL8 al EL11; la tercera ronda de datos de cuatro flujos de datos de la FEC2, esto es,

200.2, 201.2, 202.2 y 203.2, se usa como tercera ronda de datos del EL8 al EL11; y la tercera ronda de datos de cuatro flujos de datos de la FEC3, esto es, 300.2, 301.2, 302.2 y 303.2, se usa como cuarta ronda de datos del EL8 al EL11.

La cuarta ronda de datos de cuatro flujos de datos de la FEC0: 000.3, 001.3, 002.3 y 003.3, se usa como primera ronda de datos de cuatro flujos entrelazados EL12 a EL15 que se obtienen tras haber realizado el entrelazado; la cuarta ronda de datos de cuatro flujos de datos de la FEC1, esto es, 100.3, 101.3, 102.3 y 103.3, se usa como segunda ronda de datos del EL12 al EL15; la cuarta ronda de datos de cuatro flujos de datos de la FEC2, esto es, 200.3, 201.3, 202.3 y 203.3, se usa como tercera ronda de datos del EL12 al EL15; y la cuarta ronda de datos de cuatro flujos de datos de la FEC3, esto es, 300.3, 301.3, 302.3 y 303.3, se usa como cuarta ronda de datos del EL12 al EL15.

Por analogía, se siguen obteniendo datos subsiguientes tras el entrelazado.

Por ejemplo, la quinta ronda de datos de cuatro flujos de datos de la FEC0 se usa como quinta ronda de unidades de datos del EL4 al EL7; la quinta ronda de datos de cuatro flujos de datos de la FEC1 se usa como sexta ronda de datos del EL4 al EL7; la quinta ronda de datos de cuatro flujos de datos de la FEC2 se usa como séptima ronda de datos del EL4 al EL7; y la quinta ronda de unidades de datos de cuatro flujos de datos de la FEC3 se usan como octava ronda de unidades de datos del EL4 al EL7. Adicionalmente, cada bit en esta realización puede sustituirse con otra unidad de datos, por ejemplo, un bit se sustituye con un byte (byte), y entonces 000.0 indica un byte.

Tras haber realizado el entrelazado, se obtienen 16 flujos entrelazados, esto es, de EL0 a EL15. Cada flujo en los 16 flujos de datos entrelazados es ortogonal, esto es, en cada flujo entrelazado, bits adyacentes provienen todos de unidades de FEC diferentes. Por ejemplo, en el EL0, cuatro bits, que son 000.0, 100.0, 200.0 y 300.0, provienen de cuatro unidades de FEC diferentes, que refleja un rasgo ortogonal. De esta manera, cuando una cadena de errores de bits ocurren en un flujo entrelazado, tras ser desentrelazados en un extremo de recepción, los errores de bits se adjudican a las cuatro unidades de FEC para corrección de errores. De esta manera, una cantidad de errores de bits que cada unidad de FEC tiene que procesar es mucho menor que una cantidad de la cadena de errores de bits.

Haciendo referencia a la FIGURA 12, una realización de la presente invención proporciona un método para realizar entrelazado. En la FIGURA 12, flujos de datos a entrelazar sacados por una FEC0, una FEC1, una FEC2 y una FEC3 en un dispositivo de envío son iguales que los flujos de datos a entrelazar mostrados en la FIGURA 11. En esta realización, un símbolo incluye diez unidades de datos, por ejemplo, un Símbolo incluye diez unidades de datos, que son 000.0, 000.1... y 000.9 y 100.0, 100.1... y 100.9 pertenecen a otro Símbolo.

Desde luego, una cantidad de unidades de datos incluidas en un Símbolo puede variar, y no es necesariamente 10.

A continuación, una manera específica del entrelazado en esta realización se introduce según la FIGURA 12.

En la FIGURA 12, cuatro flujos entrelazados obtenidos por medio del entrelazado, que son de EL0 a EL3, se obtienen de la siguiente manera:

- la primera ronda de datos del EL0 al EL3, esto es, 000.0, 000.1, 000.2 y 000.3, proviene de las unidades de datos 1ª a 4ª en un Lane0;
- la segunda ronda de datos del EL0 al EL3, esto es, 100.0, 100.1, 100.2 y 100.3, proviene de las unidades de datos 1ª a 4ª en un Lane4;
- la tercera ronda de datos del EL0 al EL3, esto es, 200.0, 200.1, 200.2 y 200.3, proviene de las unidades de datos 1ª a 4ª en un Lane8;
- la cuarta ronda de datos del EL0 al EL3, esto es, 300.0, 300.1, 300.2 y 300.3, proviene de las unidades de datos 1ª a 4ª en un Lane12;
- la quinta ronda de datos del EL0 al EL3, esto es, 000.4, 000.5, 000.6 y 000.7, proviene de las unidades de datos 5ª a 8ª en el Lane0;
- la sexta ronda de datos del EL0 al EL3, esto es, 100.4, 100.5, 100.6 y 100.7, proviene de las unidades de datos 5ª a 8ª en el Lane4;
- la séptima ronda de datos del EL0 al EL3, esto es, 200.4, 200.5, 200.6 y 200.7, proviene de las unidades de datos 5ª a 8ª en el Lane8;
- la octava ronda de datos del EL0 al EL3, esto es, 300.4, 300.5, 300.6 y 300.7, proviene de las unidades de datos 5ª a 8ª en el Lane12;
- la novena ronda de datos del EL0 al EL3, esto es, 000.8, 000.9, 004.0 y 004.1, proviene de las unidades de datos 9ª a 12ª en el Lane0;
- la décima ronda de datos del EL0 al EL3, esto es, 100.8, 100.9, 104.0 y 104.1, proviene de las unidades de datos 9ª a la 12ª en el Lane4;
- la decimoprimera ronda de datos del EL0 al EL3, esto es, 200.8, 200.9, 204.0 y 204.1, proviene de las unidades de datos 9ª a la 12ª en el Lane8; y
- la decimosegunda ronda de datos del EL0 al EL3, esto es, 300.8, 300.9, 304.0 y 304.1, proviene de las unidades de datos 9ª a la 12ª en el Lane12.

ES 2 992 340 T3

000.8 y 000.9 en la novena ronda de datos del EL0 al EL3 y ocho unidades de datos, que son 000.0 y 000.1 a 000.7, pertenecen a un Símbolo, y las unidades de datos 004.0 y 004.1 pertenecen a otro Símbolo. También es similar para los datos de la décima ronda a la decimosegunda ronda de EL0 a EL3.

5 12 flujos entrelazados, que son EL4 a EL7, EL8 a EL11, y EL12 a EL15, también se obtienen usando el método anterior.

Cuatro flujos entrelazados, que son el EL4 al EL7, se obtienen por medio de entrelazado de la siguiente manera:

10 la primera ronda de datos del EL4 al EL7, esto es, 001.0, 001.1, 001.2 y 001.3, proviene de las unidades de datos 1ª a 4ª en un Lane1;
la segunda ronda de datos del EL4 al EL7, esto es, 101.0, 101.1, 101.2 y 101.3, proviene de las unidades de datos 1ª a 4ª en un Lane5;
la tercera ronda de datos del EL4 al EL7, esto es, 201.0, 201.1, 201.2 y 201.3, proviene de las unidades de datos 1ª a 4ª en un Lane9;
15 la cuarta ronda de datos del EL4 al EL7, esto es, 301.0, 301.1, 301.2 y 301.3, proviene de las unidades de datos 1ª a 4ª en un Lane13;
la quinta ronda de datos del EL4 al EL7, esto es, 001.4, 001.5, 001.6 y 001.7, proviene de las unidades de datos 5ª a 8ª en el Lane1;
la sexta ronda de datos del EL4 al EL7, esto es, 101.4, 101.5, 101.6 y 101.7, proviene de las unidades de datos 5ª a 8ª en el Lane5;
20 la séptima ronda de datos del EL4 al EL7, esto es, 201.4, 201.5, 201.6 y 201.7, proviene de las unidades de datos 5ª a 8ª en el Lane9;
la octava ronda de datos del EL4 al EL7, esto es, 301.4, 301.5, 301.6 y 301.7, proviene de las unidades de datos 5ª a 8ª en el Lane13;
25 la novena ronda de datos del EL4 al EL7, esto es, 001.8, 001.9, 005.0 y 005.1, proviene de las unidades de datos 9ª a 12ª en el Lane1;
la décima ronda de datos del EL4 al EL7, esto es, 101.8, 101.9, 105.0 y 105.1, proviene de las unidades de datos 9ª a 12ª en el Lane5;
30 la decimoprimer ronda de datos del EL4 al EL7, esto es, 201.8, 201.9, 205.0 y 205.1, proviene de las unidades de datos 9ª a 12ª en el Lane9; y
la decimosegunda ronda de datos del EL4 al EL7, esto es, 301.8, 301.9, 305.0 y 305.1, proviene de las unidades de datos 9ª a 12ª en el Lane13.

Cuatro flujos entrelazados, que son el EL8 al EL11, se obtienen por medio de entrelazado de la siguiente manera:

35 la primera ronda de datos del EL8 al EL11, esto es, 002.0, 002.1, 002.2 y 002.3, proviene de las unidades de datos 1ª a 4ª en un Lane2;
la segunda ronda de datos del EL8 al EL11, esto es, 102.0, 102.1, 102.2 y 102.3, proviene de las unidades de datos 1ª a 4ª en un Lane6;
40 la tercera ronda de datos del EL8 al EL11, esto es, 202.0, 202.1, 202.2 y 202.3, proviene de las unidades de datos 1ª a 4ª en un Lane10;
la cuarta ronda de datos del EL8 al EL11, esto es, 302.0, 302.1, 302.2 y 302.3, proviene de las unidades de datos 1ª a 4ª en un Lane14;
45 la quinta ronda de datos del EL8 al EL11, esto es, 002.4, 002.5, 002.6 y 002.7, proviene de las unidades de datos 5ª a 8ª en el Lane2;
la sexta ronda de datos del EL8 al EL11, esto es, 102.4, 102.5, 102.6 y 102.7, proviene de las unidades de datos 5ª a 8ª en el Lane6;
la séptima ronda de datos del EL8 al EL11, esto es, 202.4, 202.5, 202.6 y 202.7, proviene de las unidades de datos 5ª a 8ª en un Carril 10;
50 la octava ronda de datos del EL8 al EL11, esto es, 302.4, 302.5, 302.6 y 302.7, proviene de las unidades de datos 5ª a 8ª en el Lane14;
la novena ronda de datos del EL8 al EL11, esto es, 002.8, 002.9, 006.0 y 006.1, proviene de las unidades de datos 9ª a 12ª en el Lane2;
55 la décima ronda de datos del EL8 al EL11, esto es, 102.8, 102.9, 106.0 y 106.1, proviene de las unidades de datos 9ª a 12ª en el Lane6;
la decimoprimer ronda de datos del EL8 al EL11, esto es, 202.8, 202.9, 206.0 y 206.1, proviene de las unidades de datos 9ª a 12ª en el Lane10; y
60 la decimosegunda ronda de datos del EL8 al EL11, esto es, 302.8, 302.9, 306.0 y 306.1, proviene de las unidades de datos 9ª a 12ª en el Lane14;

Cuatro flujos entrelazados, que son el EL12 al EL15, se obtienen por medio de entrelazado de la siguiente manera:

65 la primera ronda de datos del EL12 al EL15, esto es, 003.0, 003.1, 003.2 y 003.3, proviene de las unidades de datos 1ª a 4ª en un Lane3;
la segunda ronda de datos del EL12 al EL15, esto es, 103.0, 103.1, 103.2 y 103.3, proviene de las unidades de datos 1ª a 4ª en un Lane7;

la tercera ronda de datos del EL12 al EL15, esto es, 203.0, 203.1, 203.2 y 203.3, proviene de las unidades de datos 1ª a 4ª en un Lane11;
 la cuarta ronda de datos del EL12 al EL15, esto es, 303.0, 303.1, 303.2 y 303.3, proviene de las unidades de datos 1ª a 4ª en un Lane15;
 5 la quinta ronda de datos del EL12 al EL15, esto es, 003.4, 003.5, 003.6 y 003.7, proviene de las unidades de datos 5ª a 8ª en el Lane3;
 la sexta ronda de datos del EL12 al EL15, esto es, 103.4, 103.5, 103.6 y 103.7, proviene de las unidades de datos 5ª a 8ª en el Lane7;
 10 la séptima ronda de datos del EL12 al EL15, esto es, 203.4, 203.5, 203.6 y 203.7, proviene de las unidades de datos 5ª a 8ª en el Lane11;
 la octava ronda de datos del EL12 al EL15, esto es, 303.4, 303.5, 303.6 y 303.7, proviene de las unidades de datos 5ª a 8ª en el Lane15;
 la novena ronda de datos del EL12 al EL15, esto es, 003.8, 003.9, 007.0 y 007.1, proviene de las unidades de datos 9ª a 12ª en el Lane3;
 15 la décima ronda de datos del EL12 al EL15, esto es, 103.8, 103.9, 107.0 y 107.1, proviene de las unidades de datos 9ª a 12ª en el Lane7;
 la decimoprimer ronda de datos del EL12 al EL15, esto es, 203.8, 203.9, 207.0 y 207.1, proviene de las unidades de datos 9ª a 12ª en el Lane11; y
 20 la decimosegunda ronda de datos del EL12 al EL15, esto es, 303.8, 303.9, 307.0 y 307.1, proviene de las unidades de datos 9ª a 12ª en el Lane15.

En esta realización, cada unidad de datos puede ser un bit, o puede ser un byte, o puede ser con otra longitud de datos.

25 La FIGURA 13 muestra un caso en el que se realiza entrelazado adicional en el EL1 al EL15 que se obtienen tras el entrelazado mostrado en la FIGURA 11. Como se muestra en la FIGURA 13, antes de realizarse el entrelazado adicional, durante la transmisión, puede ocurrir distorsión (inglés: skew) en los EL, esto es, se generan retrasos. Sin embargo, esto no afecta la ejecución del entrelazado adicional. La FIGURA 13 claramente muestra el entrelazado se realiza entre cuales EL, por ejemplo, el entrelazado se realiza entre el EL2 y el EL6, y el entrelazado se realiza entre el EL1 y el EL4. El entrelazado puede realizarse entre cualesquiera dos o más EL, sin importar si los EL provienen de una misma FEC o diferentes FEC.
 30

La FIGURA 14 muestra un caso en el que se realiza otro tipo de entrelazado adicional en el EL1 al EL15 que se obtienen tras el entrelazado mostrado en la FIGURA 11. Como se muestra en la FIGURA 14, antes de realizarse el entrelazado adicional, durante la transmisión, puede ocurrir distorsión (inglés: skew) en los EL, esto es, se generan retrasos. Sin embargo, esto no afecta la ejecución del otro tipo de entrelazado adicional. La FIGURA 14 muestra claramente el entrelazado se realiza entre cuales EL, por ejemplo, el entrelazado se realiza entre el EL2, el EL6, el EL1, y el EL4. El entrelazado puede realizarse entre cualesquiera cuatro EL, sin importar si los EL provienen de una misma FEC o diferentes FEC.
 35

40 Cuando se realiza el entrelazado adicional, una cantidad de los EL no se limita a 2 o 4, y la cantidad puede ser cualquier valor que sea mayor o igual 2.

45 Los métodos anteriores para realizar el entrelazado adicional y el otro tipo de entrelazado adicional no son aplicables únicamente a los flujos de datos obtenidos por medio del entrelazado mostrado en la FIGURA 11, sino también son aplicables a los flujos de datos obtenidos por medio del entrelazado mostrado en la FIGURA 12. La FIGURA 15 muestra un caso en el que se continúa realizando entrelazado en los flujos de datos obtenidos por medio del entrelazado mostrado en la FIGURA 12.

50 Una realización de la presente invención incluye un proceso durante el que un dispositivo de envío envía datos a un dispositivo de recepción, y el dispositivo de recepción realiza una recepción correspondiente.

Como se muestra en la FIGURA 7, el dispositivo de envío incluye cuatro unidades de FEC, que son una FECa0, una FECa1, una FECa2 y una FECa3, respectivamente. En la FIGURA 7, la FECa0 incluye canales La0 a La3, la FECa1 incluye canales La4 a La7, la FECa2 incluye canales La8 a La11, y la FECa3 incluye canales La12 a La15. Los cuatro canales La0 a La3 se usan para tener como salida un flujo de datos codificado por la FECa0, los cuatro canales La4 a La7 se usan para tener como salida un flujo de datos codificado por la FECa1, los cuatro canales La8 a La11 se usan para tener como salida un flujo de datos codificado por la FECa2, y los cuatro canales La12 a La15 se usan para tener como salida un flujo de datos codificado por la FECa3.
 55

60 Como se muestra en la FIGURA 7, el canal La0 tiene como salida un flujo de datos a00, el canal La4 tiene como salida un flujo de datos a40, el canal La8 tiene como salida un flujo de datos a80, y el canal La12 tiene como salida un flujo de datos a120. El flujo de datos a00 incluye cinco unidades de datos: unidades de datos 00 a 04 que se disponen en orden, el flujo de datos a40 incluye cinco unidades de datos: unidades de datos 40 a 44 que se disponen en orden, el flujo de datos a80 incluye cinco unidades de datos: unidades de datos 80 a 84 que se disponen en orden, y el flujo de datos a120 incluye cinco unidades de datos: unidades de datos 120 a 124 que se disponen en orden. El flujo de datos
 65

a00 incluye un identificador de canal del canal La0, el flujo de datos a40 incluye un identificador de canal del canal La4, el flujo de datos a80 incluye un identificador de canal del canal La8, y el flujo de datos a120 incluye un identificador de canal del canal La12.

5 El dispositivo de envío realiza primer entrelazado y segundo entrelazado usando una unidad de entrelazado. Para el primer entrelazado, el entrelazado se realiza según una primera unidad de datos, y para el segundo entrelazado, el entrelazado se realiza según una segunda unidad de datos, donde la primera unidad de datos incluye al menos dos bits, y la segunda unidad de datos incluye al menos dos bits. El número de bits incluido en la primera unidad de datos y el número de bits incluido en la segunda unidad de datos pueden ser iguales o diferentes.

10 Realizar entrelazado según una unidad de datos se refiere a dividir, según una longitud de la unidad de datos, cada uno de al menos dos flujos de datos a entrelazar hasta bloques de datos, y entonces realizar el entrelazado en los bloques de datos obtenidos, donde la longitud de la unidad de datos se refiere al número de bits incluidos en la unidad de datos.

15 El primer entrelazado específicamente incluye: realizar el entrelazado en el flujo de datos a00 y el flujo de datos a40, para obtener un flujo de datos i0'; y el segundo entrelazado incluye realizar el entrelazado en el flujo de datos a80 y el flujo de datos a120, para obtener un flujo de datos i1'.

20 En la FIGURA 7, para la descripción se usa un ejemplo en el que cada unidad de datos en cada flujo de datos incluye cuatro bits. Como se muestra en la FIGURA 7, en el flujo de datos i0', la unidad de datos 00 incluye cuatro bits: 00.0, 00.1, 00.2 y 00.3, y la unidad de datos 40 incluye cuatro bits: 40.0, 40.1, 40.2, y 40.3; y la unidad de datos 80 en el flujo de datos i1' incluye cuatro bits: 80.0, 80.1, 80.2, y 80.3, y la unidad de datos 120 en el flujo de datos i1' incluye cuatro bits: 120.0, 120.1, 120.2 y 120.3. Se debe observar que, en una aplicación real, el número de bits incluido en cada unidad de datos también puede tener otro valor no menor que dos, por ejemplo, cada unidad de datos incluye 10 bits.

25 De la FIGURA 7 se puede ver que, los números de bits incluidos en la primera unidad de datos y el número de bits incluido en la segunda unidad de datos son ambos cuatro.

30 La unidad de entrelazado realiza tercer entrelazado en el flujo de datos i0' y el flujo de datos i1', para obtener un flujo de datos i2', donde para el tercer entrelazado, el entrelazado se realiza según un bit. El entrelazado se realiza según un bit; por lo tanto, en el flujo de datos i2', los cuatro bits: bits 00.0, 00.1, 00.2 y 00.3 ya no son consecutivos, pero se entrelazan mutuamente con los cuatro bits en la unidad de datos 80.

35 En el tercer entrelazado mostrado en la FIGURA 7, no hay distorsión (inglés: skew) entre el flujo de datos i0' y el flujo de datos i1', esto es, el entrelazado se realiza en el flujo de datos i0' y el flujo de datos i1' en un caso en el que se alinea el flujo de datos i0' y el flujo de datos i1'. Opcionalmente, el entrelazado puede realizarse en el flujo de datos i0' y el flujo de datos i1' en un caso en el que no haya distorsión entre el flujo de datos i0' y el flujo de datos i1'. Por ejemplo, cuando la distorsión es de dos bits, un orden de disposición de los múltiples bits en el flujo de datos i2' incluye 00.2-80.0-00.3-80.1-40.0-80.2-40.1-80.3-40.2-120.0-40.3....

40 Como se muestra en la FIGURA 8, después de que el dispositivo de recepción recibe el flujo de datos i2', se realiza primer desentrelazado en el flujo de datos i2 usando una unidad de desentrelazado, para obtener el flujo de datos i0' y el flujo de datos i1', donde para el primer desentrelazado, el desentrelazado se realiza según un bit. Para el tercer entrelazado, el entrelazado se realiza según un bit. Por lo tanto, para el correspondiente primer desentrelazado, el desentrelazado también tiene que realizarse según un bit.

45 El dispositivo de recepción entonces realiza segundo desentrelazado en el flujo de datos i0' usando la unidad de desentrelazado, y realiza tercer desentrelazado en el flujo de datos i1'. El segundo desentrelazado específicamente incluye: realizar el desentrelazado en el flujo de datos i0' según la primera unidad de datos, para obtener el flujo de datos a00 y el flujo de datos a40; y el tercer desentrelazado específicamente incluye: realizar el desentrelazado en el flujo de datos i1' según la segunda unidad de datos, para obtener el flujo de datos a80 y el flujo de datos a120.

50 Realizar el desentrelazado según una unidad de datos es un proceso inverso de realizar el entrelazado según una unidad de datos. Por lo tanto, la realización del desentrelazado según la primera unidad de datos es un proceso inverso de la realización del entrelazado según la primera unidad de datos, y la realización del desentrelazado según la segunda unidad de datos es un proceso inverso de la realización del entrelazado según la segunda unidad de datos.

55 Después de eso, según el método en la realización correspondiente a la FIGURA 6, el dispositivo de recepción envía el flujo de datos a00 a la FECb0 en el dispositivo de recepción, envía el flujo de datos a40 a la FECb1 en el dispositivo de recepción, envía el flujo de datos a80 a una FECb2 en el dispositivo de recepción, y envía el flujo de datos a120 a un FECb3 en el dispositivo de recepción.

60 Como se muestra en la FIGURA 9, una realización de la presente invención proporciona un dispositivo de envío 10, donde el dispositivo de envío 10 incluye al menos dos unidades de FEC, por ejemplo, el dispositivo de envío 10 puede

incluir una unidad de FEC 110 y una unidad de FEC 111, y puede incluir además una o más de otras unidades de FEC. Cada unidad de FEC se puede conectar a uno o al menos dos canales. Cada unidad de FEC se conecta a una unidad de entrelazado 12 usando un canal conectado a la unidad de FEC. Por ejemplo, como se muestra en la FIGURA 9, la unidad de FEC 110 se conecta a la unidad de entrelazado 12 usando $m+1$ canales desde un canal L1100 a un canal L110m. De manera similar, la unidad de FEC 111 se conecta a la unidad de entrelazado 12 usando $m+1$ canales desde un canal L1110 a un canal L111m; y una unidad de FEC 11n se conecta a la unidad de entrelazado 12 usando $m+1$ canales desde un canal L11n0 a un canal L11nm. Se debe observar que, el número de las unidades de FEC en el dispositivo de envío 10 puede variar, y es aceptable siempre que el número no sea menor que dos. Por ejemplo, en el dispositivo de envío 10, puede haber únicamente dos unidades de FEC o más unidades de FEC.

La unidad de FEC 110 se configura para enviar, usando el canal L1100, un flujo de datos s1100 en el que se ha realizado primera codificación FEC, donde el flujo de datos s1100 incluye un identificador de canal del canal L1100; y

la unidad de FEC 111 se configura para enviar, usando el canal L1110, un flujo de datos s1110 en el que se ha realizado segunda codificación FEC, donde el flujo de datos s1110 incluye un identificador de canal del canal L 1110, y una regla de codificación de la primera codificación FEC y que de la segunda codificación FEC son iguales o diferentes.

La unidad de entrelazado 12 se configura para realizar entrelazado en el flujo de datos s1100 y el flujo de datos s1110 que son recibidos, para obtener un flujo de datos de salida.

El flujo de datos de salida obtenido a través del entrelazado puede ser un flujo de datos o puede incluir al menos dos flujos de datos. Por ejemplo, cuando la unidad de entrelazado 12 realiza el entrelazado de la manera mostrada en la FIGURA 1, el flujo de datos de salida incluye un flujo de datos; y cuando la unidad de entrelazado 12 realiza el entrelazado de la manera mostrada en la FIGURA 2, el flujo de datos de salida incluye dos flujos de datos que son independientes uno de otro.

El dispositivo de envío 10 incluye además una unidad de envío 13, configurada para enviar el flujo de datos de salida a un dispositivo de recepción.

Opcionalmente, la unidad de FEC 110 se configura además para enviar, usando un canal L1101, un flujo de datos s1101 en el que se ha realizado tercera codificación FEC, donde el flujo de datos s1101 incluye un identificador de canal del canal L1101, y una regla de codificación de la tercera codificación FEC y que de la primera codificación FEC son iguales o diferentes; y la unidad de FEC 111 se configura además para enviar, usando un canal L1111, un flujo de datos s1111 en el que se ha realizado cuarta codificación FEC, donde el flujo de datos s1111 incluye un identificador de canal del canal L1 111, y una regla de codificación de la cuarta codificación FEC y que de la segunda codificación FEC son iguales o diferentes.

Cuando la unidad de FEC 110 envía el flujo de datos s1100 y el flujo de datos s1101, y la unidad de FEC 111 envía el flujo de datos s1110 y el flujo de datos s1111, la unidad de entrelazado 12 se configura específicamente para:

realizar primer entrelazado en el flujo de datos s1100 y el flujo de datos s1110;
 realizar segundo entrelazado en el flujo de datos s1101 y el flujo de datos s1111; y
 realizar tercer entrelazado en un flujo de datos que se obtiene al realizar el primer entrelazado y un flujo de datos que se obtiene al realizar el segundo entrelazado, para obtener el flujo de datos de salida.

Preferiblemente, para el primer entrelazado, el entrelazado se realiza según una primera unidad de datos, para el segundo entrelazado, el entrelazado se realiza según una segunda unidad de datos, donde la primera unidad de datos incluye al menos dos bits, y la segunda unidad de datos incluye al menos dos bits, y para el tercer entrelazado, el entrelazado se realiza según un bit.

Opcionalmente, cuando la unidad de FEC 110 envía el flujo de datos s1100 y el flujo de datos s1101, y la unidad de FEC 111 envía el flujo de datos s1110 y el flujo de datos s1111, la unidad de entrelazado 12 puede configurarse además específicamente para:

realizar primer entrelazado en el flujo de datos s1100 y el flujo de datos s1110; y
 realizar segundo entrelazado en el flujo de datos s1101 y el flujo de datos s1111, donde el flujo de datos de salida incluye un flujo de datos que se obtiene al realizar el primer entrelazado y un flujo de datos que se obtiene al realizar el segundo entrelazado. De esta manera, el entrelazado ya no se realiza en el flujo de datos que se obtiene al realizar el primer entrelazado y el flujo de datos que se obtiene al realizar el segundo entrelazado.

Preferiblemente, para el primer entrelazado, el entrelazado se realiza según una primera unidad de datos, y para el segundo entrelazado, el entrelazado se realiza según una segunda unidad de datos, donde la primera unidad de datos incluye al menos dos bits, y la segunda unidad de datos incluye al menos dos bits.

En el dispositivo de envío 10, la unidad de FEC 11 y la unidad de entrelazado 12 se implementan por chips de hardware que son independientes entre sí, o por un mismo chip de hardware. Múltiples unidades de FEC pueden pertenecer a un mismo chip de hardware, y cada unidad de FEC es una unidad de hardware en el chip de hardware. La unidad de envío 10 también es una unidad que se implementa por hardware.

El dispositivo de envío 10 puede ser el dispositivo de envío en la realización correspondiente a la FIGURA 1, la realización correspondiente a la FIGURA 2, la realización correspondiente a la FIGURA 5, o la realización correspondiente a la FIGURA 7, y puede implementar todas las funciones de los dispositivos de envío en las cuatro realizaciones. El dispositivo de envío 10 puede además ser el dispositivo de envío en la realización correspondiente a la FIGURA 11, FIGURA 12, FIGURA 13, o FIGURA 14, y puede implementar todas las funciones de los dispositivos de envío en la realizaciones anteriores.

El dispositivo de envío en esta realización realiza entrelazado en al menos dos flujos de datos codificados por unidades de FEC diferentes, y envía datos obtenidos a través del entrelazado a un dispositivo de recepción. De esta manera, el dispositivo de recepción puede realizar desentrelazado en un flujo de datos después del entrelazado, y enviar al menos flujos de datos obtenidos a través del desentrelazado a unidades de FEC diferentes para decodificar. Cuando ocurre un error de bits durante un proceso de transmisión del flujo de datos obtenido a través del entrelazado, los flujos de datos obtenidos tras el desentrelazado se envían a las unidades de FEC diferentes para decodificar. Por lo tanto, para una unidad de FEC, disminuye el número de errores de bits recibidos por la unidad de FEC. En otras palabras, aumenta la probabilidad de que la unidad de FEC realice corrección de errores con éxito. En global, cuando aumentan las tasas de éxito de corrección de errores de una o más unidades de FEC en el dispositivo de recepción, se mejora la capacidad de corrección de errores del dispositivo de recepción. Adicionalmente, en esta realización, no es necesaria una manera de entrelazar de escritura por fila y lectura por columna; por lo tanto, no se genera retraso. Adicionalmente, un entrelazador que implementa la manera de entrelazar de escribir por fila y leer por columna es de diseño complejo y consume mucha potencia. En esta realización, este tipo de especial entrelazador no es necesario; por lo tanto, la implementación es simple y se reduce el consumo de potencia de un dispositivo. En resumen, el dispositivo de envío en esta realización ayuda a implementar un método de corrección de errores simple, que ahorra energía y es eficiente.

Como se muestra en la FIGURA 10, una realización de la presente invención proporciona un dispositivo de recepción 20, donde el dispositivo de recepción 20 incluye una unidad de recepción 21, una unidad de desentrelazado 22, y una unidad de determinación 23. El dispositivo de recepción 20 incluye además una unidad de FEC 240 y una unidad de FEC 241. Adicionalmente, el dispositivo de recepción 20 puede incluir además una o más de otras unidades de FEC.

La unidad de recepción 21 se configura para recibir un primer flujo de datos de un dispositivo de envío. El primer flujo de datos puede ser un flujo de datos o al menos dos flujos de datos. Por ejemplo, el primer flujo de datos en esta realización puede ser el flujo de datos i0 mostrado en la FIGURA 1, o los flujos de datos i00 y i01 mostrados en la FIGURA 2.

La unidad de desentrelazado 22 se configura para realizar desentrelazar en el primer flujo de datos, para obtener un segundo flujo de datos y un tercer flujo de datos.

La unidad de determinación 23 se configura para realizar la siguiente operación:

determinar que el segundo flujo de datos incluye un primer identificador de canal y el tercer flujo de datos incluye un segundo identificador de canal, donde el primer identificador de canal es un identificador de canal de un primer canal que se conecta a una unidad de FEC a en el dispositivo de envío, y el segundo identificador de canal es un identificador de canal de un segundo canal que se conecta a una unidad de FEC b en el dispositivo de envío.

Cuando el primer flujo de datos es enviado por el dispositivo de envío 10 mostrado en la FIGURA 9, el primer identificador de canal puede ser un identificador de canal de un canal L1100, la unidad de FEC a puede ser una unidad de FEC FEC110, el segundo identificador de canal puede ser un identificador de canal de un canal L1110, y la unidad de FEC b puede ser una unidad de FEC 110.

Cuando la unidad de determinación 23 puede determinar que un flujo de datos incluye un identificador de canal, indica que este flujo de datos incluye este identificador de canal, y este identificador de canal existe de manera identificable. Por ejemplo, en esta realización, indica que el segundo flujo de datos obtenido a través del desentrelazador ya incluye el primer identificador de canal, y el tercer flujo de datos obtenido a través del desentrelazado ya incluye el segundo identificador de canal. Si el dispositivo de recepción 20 recibe un flujo de datos a, el desentrelazado se realiza en el flujo de datos a para obtener un flujo de datos b y un flujo de datos c. Sin embargo, si la unidad de determinación 23 determina que al menos un flujo de datos del flujo de datos b y el flujo de datos c no incluye un identificador de canal, el flujo de datos a no se consideran como primer flujo de datos en esta realización.

La unidad de determinación 23 continúa realizando la siguiente operación:

determinar, según el primer identificador de canal y una primera correspondencia que se configura en el dispositivo de recepción, un tercer identificador de canal correspondiente al primer identificador de canal, donde la primera

correspondencia incluye una correspondencia entre el primer identificador de canal y el tercer identificador de canal, y el tercer identificador de canal es un identificador de canal de un tercer canal que se conecta a una unidad de FEC en el dispositivo de recepción 20, por ejemplo, el tercer canal puede ser un canal L2400, y la unidad de FEC conectada al tercer canal puede ser la unidad de FEC 240. La unidad de FEC conectada al tercer canal necesita tener una capacidad de realizar decodificación FEC en el segundo flujo de datos.

Tras determinar el tercer identificador de canal, la unidad de determinación 23 envía, usando el tercer canal, el segundo flujo de datos a la unidad de FEC en el dispositivo de recepción 20 y conectado al tercer canal, para decodificación FEC.

El dispositivo de recepción 20 además determina, según el segundo identificador de canal y una segunda correspondencia que se configura en el dispositivo de recepción, un cuarto identificador de canal correspondiente al segundo identificador de canal, donde la segunda correspondencia incluye una correspondencia entre el segundo identificador de canal y el cuarto identificador de canal, el cuarto identificador de canal es un identificador de canal de un cuarto canal conectado a otra unidad de FEC en el dispositivo de recepción, y la otra unidad de FEC se refiere a una unidad de FEC que es diferente de la unidad de FEC en el dispositivo de recepción 20 y conectado al tercer canal. Por ejemplo, cuando la unidad de FEC conectada al tercer canal es la unidad de FEC 240, la otra unidad de FEC puede ser la unidad de FEC 241 u otra unidad de FEC en el dispositivo de recepción 20 y no se muestra en la FIGURA 10. Cuando la otra unidad de FEC es la unidad de FEC 241, el cuarto canal puede ser un canal L2410, un canal L2411, u otro canal conectado a la unidad de FEC 241. La unidad de FEC conectada al cuarto canal debe tener la capacidad de realizar decodificación FEC en el tercer flujo de datos.

El dispositivo de recepción 20 además envía, usando el cuarto canal, el tercer flujo de datos a la unidad de FEC en el dispositivo de recepción 20 y conectado al cuarto canal, para decodificación FEC.

Cuando existe un error de bits en el primer flujo de datos, se realiza una operación de desentrelazado; por lo tanto, el error de bits puede adjudicarse al segundo flujo de datos y el tercer flujo de datos. El segundo flujo de datos y el tercer flujo de datos se adjudican a unidades de FEC diferentes para decodificación FEC; por lo tanto, un caso en el que la corrección de errores se realiza usando únicamente una unidad de FEC se puede impedir en gran medida, y una o más de otras unidades de FEC en el dispositivo de recepción se puede utilizar más totalmente para la corrección de errores, para mejorar una capacidad de corrección de errores del dispositivo de recepción.

En esta realización, el dispositivo de envío puede realizar el entrelazado en flujos de datos en múltiples maneras; por lo tanto, puede haber múltiples maneras de procesamiento de desentrelazado por la unidad de desentrelazado 22.

La unidad de desentrelazado 22 puede realizar el desentrelazado de la manera mostrada en la FIGURA 3, la FIGURA 4, la FIGURA 6, o la FIGURA 8.

Cuando la unidad de desentrelazado 22 realiza el desentrelazado de la manera mostrada en la FIGURA 3, la unidad de desentrelazado 22 realiza el desentrelazado en el flujo de datos i_0 recibido, para obtener un flujo de datos a_0 y un flujo de datos a_4 . El flujo de datos i_0 puede considerarse como primer flujo de datos recibido por el dispositivo de recepción 20, el flujo de datos a_0 puede considerarse como segundo flujo de datos obtenido por la unidad de desentrelazado 22, y el flujo de datos a_4 puede considerarse como tercer flujo de datos obtenido por la unidad de desentrelazado 22.

Cuando la unidad de desentrelazado 22 realiza el desentrelazado de la manera mostrada en la FIGURA 4, la unidad de desentrelazado 22 realiza el desentrelazado en los dos flujos de datos: el flujo de datos i_{00} y el flujo de datos i_{01} que son recibidos, para obtener el flujo de datos a_0 y el flujo de datos a_4 . El flujo de datos i_{00} puede considerarse como primer flujo de datos recibido por el dispositivo de recepción 20, el flujo de datos i_{01} puede considerarse como otro flujo de datos recibido por el dispositivo de recepción 20, el flujo de datos a_0 puede considerarse como segundo flujo de datos obtenido por la unidad de desentrelazado 22, y el flujo de datos a_4 puede considerarse como tercer flujo de datos obtenido por la unidad de desentrelazado 22.

Cuando la unidad de desentrelazado 22 realiza el desentrelazado de la manera mostrada en la FIGURA 6, la unidad de desentrelazado 22 realiza primer desentrelazado en un flujo de datos i_2 recibido, para obtener el flujo de datos i_0 y un flujo de datos i_1 . Se realiza segundo desentrelazado en el flujo de datos i_0 , para obtener un flujo de datos a_{00} y un flujo de datos a_{40} desde el flujo de datos i_0 ; y se realiza tercer desentrelazado en el flujo de datos i_1 , para obtener flujos de datos a_{20} y a_{60} desde el flujo de datos i_1 . El flujo de datos i_2 puede considerarse como primer flujo de datos, el flujo de datos a_{00} puede considerarse como segundo flujo de datos, y el flujo de datos a_{40} puede considerarse como tercer flujo de datos. El flujo de datos i_0 puede considerarse como cuarto flujo de datos, el flujo de datos i_1 puede considerarse como quinto flujo de datos, el flujo de datos a_{20} puede considerarse como sexto flujo de datos, y el flujo de datos a_{60} puede considerarse como séptimo flujo de datos.

Cuando la unidad de desentrelazado 22 realiza el desentrelazado de la manera mostrada en la FIGURA 8, la unidad de desentrelazado 22 realiza primer desentrelazado en un flujo de datos i_2' recibido, para obtener un flujo de datos i_0' y un flujo de datos i_1' , donde para el primer desentrelazado, el desentrelazado se realiza según un bit. Se realiza

segundo desentrelazado en un flujo de datos i0', para obtener un flujo de datos a00 y un flujo de datos a40 desde el flujo de datos i0'; y se realiza tercer desentrelazado en el flujo de datos i1', para obtener flujos de datos a20 y a60 del flujo de datos i1'. Preferiblemente, para el segundo desentrelazado, el desentrelazado se realiza según una primera unidad de datos, y para el tercer desentrelazado, el desentrelazado se realiza según una segunda unidad de datos, donde la primera unidad de datos incluye al menos dos bits, y la segunda unidad de datos incluyen al menos dos bits. El flujo de datos i2' puede considerarse como primer flujo de datos, el flujo de datos a00 puede considerarse como segundo flujo de datos, y el flujo de datos a40 puede considerarse como tercer flujo de datos. El flujo de datos i0' puede considerarse como cuarto flujo de datos, el flujo de datos i1' puede considerarse como quinto flujo de datos, un flujo de datos a80 puede considerarse como sexto flujo de datos, y un flujo de datos a120 puede considerarse como séptimo flujo de datos.

Cuando la unidad de desentrelazado 22 realiza el desentrelazado de la manera mostrada en la FIGURA 3 o FIGURA 4, operaciones realizadas por la unidad de determinación 23 incluyen las operaciones anteriores realizadas por la unidad de determinación 23 en el segundo flujo de datos y el tercer flujo de datos en esta realización.

Cuando la unidad de desentrelazado 22 realiza el desentrelazado de la manera mostrada en la FIGURA 6 o FIGURA 8, flujos de datos obtenidos a través del desentrelazado no únicamente incluyen el segundo flujo de datos y el tercer flujo de datos, sino también incluyen el sexto flujo de datos y el séptimo flujo de datos.

Por lo tanto, preferiblemente, la unidad de determinación 23 tiene además que procesar el sexto flujo de datos y el séptimo flujo de datos, y una manera de procesamiento es la misma que la manera de procesamiento del segundo flujo de datos y el tercer flujo de datos que pertenecen a una misma realización que el sexto flujo de datos y el séptimo flujo de datos.

Específicamente, cuando la unidad de desentrelazado realiza el desentrelazado de la manera mostrada en la FIGURA 6, la unidad de determinación 23 determina que el sexto flujo de datos incluye un quinto identificador de canal y el séptimo flujo de datos incluye un sexto identificador de canal. El quinto identificador de canal es un identificador de canal de un quinto canal en el dispositivo de envío y conectado a la unidad de FEC un, y el sexto identificador de canal es un identificador de canal de un sexto canal en el dispositivo de envío y conectado a la unidad de FEC b. Por ejemplo, la unidad de FEC a puede ser la unidad de FEC 110 mostrado en la FIGURA 7, la unidad de FEC b puede ser la unidad de FEC 111 mostrada en la FIGURA 7, el quinto canal puede ser el canal L1101 mostrado en la FIGURA 7, y el sexto canal puede ser el canal L1111 mostrado en la FIGURA 7.

La unidad de determinación 23 determina, según el quinto identificador de canal y una tercera correspondencia que se configura en el dispositivo de recepción, un séptimo identificador de canal correspondiente al quinto identificador de canal. La tercera correspondencia incluye una correspondencia entre el quinto identificador de canal y el séptimo identificador de canal, el séptimo identificador de canal es un identificador de canal de un séptimo canal, y el séptimo canal y el tercer canal se conectan a una misma unidad de FEC en el dispositivo de recepción. Por ejemplo, el séptimo canal puede ser el canal L2401 en la FIGURA 8, el tercer canal es el L2400, y el canal L2401 y el canal L2400 se conectan ambos a la unidad de FEC 240. La unidad de determinación 23 se configura además para enviar, usando el séptimo canal, el sexto flujo de datos a la unidad de FEC conectada al séptimo canal, para decodificación FEC. La unidad de FEC conectada al séptimo canal necesita tener una capacidad de realizar decodificación FEC en el sexto flujo de datos.

La unidad de determinación 23 determina, según el sexto identificador de canal y una cuarta correspondencia que se configura en el dispositivo de recepción, un octavo identificador de canal correspondiente al sexto identificador de canal. La cuarta correspondencia incluye una correspondencia entre el sexto identificador de canal y el octavo identificador de canal, el octavo identificador de canal es un identificador de un octavo canal, y el octavo canal y el cuarto canal se conectan a una misma unidad de FEC en el dispositivo de recepción. Por ejemplo, el octavo canal puede ser el canal L241n en la FIGURA 8, el cuarto canal es el canal L2410, y el canal L241n y el canal L2410 se conectan ambos a la unidad de FEC 241. La unidad de determinación 23 se configura además para enviar, usando el octavo canal, el séptimo flujo de datos a la unidad de FEC conectada al octavo canal, para decodificación FEC. La unidad de FEC conectada a el octavo canal necesita tener una capacidad de realizar decodificación FEC en el séptimo flujo de datos.

Cuando la unidad de desentrelazado realiza el desentrelazado de la manera mostrada en la FIGURA 8, una operación realizada por la unidad de determinación 23 es similar a la realizada por la unidad de determinación 23 cuando la unidad de desentrelazado realiza el desentrelazado de la manera mostrada en la FIGURA 6. En esta memoria no se describen de nuevo detalles.

En esta realización, la unidad de determinación 23 puede ser un chip de hardware o puede incluir múltiples chips de hardware que son independientes entre sí. La unidad de desentrelazado 22 puede ser independiente de la unidad de determinación 23, o integrarse con la unidad de determinación 23 o una parte de la unidad de determinación 23. El dispositivo de recepción 20 puede ser el dispositivo de recepción en la realización correspondiente a la FIGURA 3, FIGURA 4, o FIGURA 6, y puede realizar todas las operaciones realizadas por el dispositivo de recepción en la realización correspondiente a la FIGURA 3, FIGURA 4, o FIGURA 6.

5 En esta realización, las unidades de datos en las que ocurre un error se envían a unidades de FEC diferentes. Por lo tanto, para una unidad de FEC, disminuye el número de errores de bits recibidos por la unidad de FEC. En otras palabras, aumenta la probabilidad de que la unidad de FEC realice corrección de errores con éxito. En global, cuando aumentan las tasas de éxito de corrección de errores de una o más unidades de FEC en el dispositivo de recepción, se mejora la capacidad de corrección de errores del dispositivo de recepción. Adicionalmente, en esta realización, no es necesaria una manera de entrelazar de escritura por fila y lectura por columna; por lo tanto, no se genera retraso. Adicionalmente, un entrelazador que implementa la manera de entrelazar de escribir por fila y leer por columna es de diseño complejo y consume mucha potencia. En esta realización, este tipo de especial entrelazador no es necesario; por lo tanto, la implementación es simple y se reduce el consumo de potencia de un dispositivo.

15 Un experto en la técnica puede entender que todas o una parte de las etapas de las realizaciones de método pueden ser implementadas por un programa que da instrucciones a hardware relevante. El programa se puede almacenar en un soporte de almacenamiento legible por ordenador. Cuando el programa está en marcha, se realizan las etapas de las realizaciones del método. El soporte de almacenamiento anterior incluye: cualquier soporte que pueda almacenar código de programa, tal como una ROM, una RAM, un disco magnético o un disco óptico.

20 Las descripciones anteriores son meramente maneras de implementación específicas ejemplares de la presente invención, pero no pretenden limitar el alcance de protección de la presente invención. Cualquier variación o sustitución fácilmente imaginada por un experto en la técnica dentro del alcance técnico divulgado en la presente invención se encontrará dentro del alcance de protección de la presente invención como se define en las reivindicaciones adjuntas.

REIVINDICACIONES

1. Un método de generación de flujos de datos, que comprende:

5 generar al menos dos flujos de datos (i00, i01) al entrelazar primeros datos codificados (00, 01) de Corrección de Errores Hacia Delante, FEC, con segundos datos codificados (40, 41) de FEC según una unidad de datos que comprende al menos dos bits, en donde dos unidades de datos (00, 41; 40, 01) uno al lado del otro en uno cualquiera de al menos dos flujos de datos son de los primeros datos codificados de FEC y de los segundos datos codificados de FEC, respectivamente; y en el que en cada par de flujos de datos adyacentes de al menos
10 dos flujos de datos (i00, i01), las unidades de datos (00, 40; 41, 01) ubicadas en una misma posición en los flujos de datos del par respectivo comprenden una unidad de datos de los primeros datos codificados (00, 01) FEC y una unidad de datos de los segundos datos codificados (40, 41) FEC; y generar un flujo de datos entrelazados (i2') al entrelazar al menos dos flujos de datos (i00, i01) según un bit.

15 2. El método de la reivindicación 1, que comprende además:

generar los primeros datos codificados (00, 01) de FEC; y generar los segundos datos codificados (40, 41) de FEC.

20 3. El método de la reivindicación 2, en donde los primeros datos codificados (00, 01) de FEC son generados por una primera unidad de FEC y los segundos datos codificados (40, 41) de FEC son generados por una segunda unidad de FEC.

25 4. El método de una cualquiera de las reivindicaciones 1 a 3, en donde al menos dos flujos de datos (i00, i01) comprenden un primer flujo de datos (i0), el primer flujo de datos (i0) comprende una primera unidad de datos (00) de los primeros datos codificados de FEC y una segunda unidad de datos (01) de los primeros datos codificados de FEC, en donde un orden de la primera unidad de datos (00) y la segunda unidad de datos (01) en el primer flujo de datos (i00) es el mismo que el orden de la primera unidad de datos (00) y la segunda unidad de datos (01) en los primeros datos codificados de FEC.

30 5. El método de una cualquiera de las reivindicaciones 1 a 4, que comprende además:

transmitir el flujo de datos entrelazados (i2').

35 6. El método de una cualquiera de las reivindicaciones 1 a 5, en donde dos bits uno al lado del otro en el flujo de datos entrelazados (i2') proviene de diferentes flujos de datos de al menos dos flujos de datos (i0, i01).

40 7. El método de una cualquiera de las reivindicaciones 1 a 6, en donde el método es realizado por una interfaz de Ethernet de alta velocidad.

8. El método de la reivindicación 7, en donde la interfaz Ethernet de alta velocidad está configurada para 100GE, 400GE o 1TE.

45 9. Un dispositivo, que comprende una unidad de entrelazado (12) configurado para:

generar al menos dos flujos de datos (i0, i1) al entrelazar primeros datos codificados de Corrección de Errores Hacia Delante, FEC, con segundos datos codificados de FEC según una unidad de datos que comprende al menos dos bits, en donde dos unidades de datos una la lado de la otra en uno cualquiera de al menos dos flujos de datos (i0, i1) son de los primeros datos codificados de FEC y de los segundos datos codificados de FEC, respectivamente; y en el que en cada par de flujos de datos adyacentes de al menos dos flujos de datos (i00, i01), las unidades de datos (00, 40; 41, 01) ubicadas en una misma posición en los flujos de datos del par respectivo comprenden una unidad de datos a partir de los primeros datos codificados FEC (00, 01) y una unidad de datos de los segundos datos codificados FEC (40, 41); y
50 generar un flujo de datos (i2') entrelazados al entrelazar al menos dos flujos de datos (i0, i1) según un bit.

55 10. El dispositivo de la reivindicación 9, en donde el dispositivo comprende además:

una primera unidad de FEC (110), configurada para generar los primeros datos codificados de FEC; y una segunda unidad de FEC (111), configurada para generar los segundos datos codificados de FEC.

60 11. El dispositivo de la reivindicación 9 o la reivindicación 10, en donde al menos dos flujos de datos comprenden un primer flujo de datos, el primer flujo de datos comprende una primera unidad de datos de los primeros datos codificados de FEC y una segunda unidad de datos de los primeros datos codificados de FEC, y un orden de la primera unidad de datos y la segunda unidad de datos en el primer flujo de datos es el mismo que el orden de la primera unidad de datos y la segunda unidad de datos en los primeros datos codificados de FEC.
65

12. El dispositivo de una cualquiera de las reivindicaciones 9 a 11, que comprende además:

una unidad de envío (13), configurada para transmitir el flujo de datos entrelazados.

5 13. El dispositivo de una cualquiera de las reivindicaciones 9 a 12, en donde dos bits uno al lado del otro en el flujo de datos entrelazados provienen de diferentes flujos de datos de al menos dos flujos de datos.

14. El dispositivo de una cualquiera de las reivindicaciones 9 a 13, en donde el dispositivo es uno o más chips, o el dispositivo es un dispositivo de red.

10 15. El dispositivo de una cualquiera de las reivindicaciones 9 a 14, en donde el dispositivo es una interfaz de Ethernet de alta velocidad.

15 16. El dispositivo de la reivindicación 15, en donde la interfaz de Ethernet de alta velocidad está configurada para 100GE, 400GE o 1TE.

17. Un método, el método que comprende:

20 recibir un flujo de datos entrelazados ($i2'$);
 obtener al menos dos flujos de datos ($i0, i1$) al desentrelazar el flujo de datos ($i2'$) entrelazados según un bit; y
 obtener primeros datos codificados (00, 01) de Corrección de Errores Hacia Delante, FEC, y segundos datos
 codificados (40, 41) de FEC al desentrelazar al menos dos flujos de datos ($i0, i1$) según una unidad de datos
 que comprende al menos dos bits, y en el que en cada par de flujos de datos adyacentes de al menos dos
 flujos de datos ($i00, i01$), las unidades de datos (00, 40; 41, 01) ubicadas en una misma posición en los flujos
 25 de datos del par respectivo comprenden una unidad de datos a partir de los primeros datos codificados (00,
 01) FEC y una unidad de datos de los segundos datos codificados (40, 41) FEC.

18. El método de la reivindicación 17, que comprende además:

30 decodificar los primeros datos codificados (00, 01) de FEC y los segundos datos codificados (40, 41) de FEC.

35 19. El método de la reivindicación 18, en donde al menos dos flujos de datos comprenden un primer flujo de datos y un segundo flujo de datos, el primer flujo de datos comprende una primera unidad de datos de los primeros datos codificados de FEC y una segunda unidad de datos de los segundos datos codificados de FEC, el segundo flujo de datos comprende una tercera unidad de datos de los primeros datos codificados de FEC y una cuarta unidad de datos de los segundos datos codificados de FEC, en donde obtener los primeros datos codificados de FEC y los segundos datos codificados de FEC al desentrelazar al menos dos flujos de datos ($i0, i1$) según una unidad de datos que comprende al menos dos bits comprende:

40 obtener la primera unidad de datos de los primeros datos codificados de FEC, la tercera unidad de datos de los primeros datos codificados de FEC, la segunda unidad de datos de los segundos datos codificados de FEC y la cuarta unidad de datos de los segundos datos codificados de FEC al desentrelazar el primer flujo de datos y el segundo flujo de datos según la unidad de datos que comprende al menos dos bits.

45 20. El método de la reivindicación 19, en donde decodificar los primeros datos codificados de FEC y los segundos datos codificados de FEC comprende:

50 decodificar la primera unidad de datos y la tercera unidad de datos; y
 decodificar la segunda unidad de datos y la cuarta unidad de datos.

55 21. El método de una cualquiera de las reivindicaciones 18 a 20, en donde al menos dos flujos de datos comprenden un tercer flujo de datos y un cuarto flujo de datos, el tercer flujo de datos comprende una quinta unidad de datos de los primeros datos codificados de FEC y una sexta unidad de datos de los segundos datos codificados de FEC, y el cuarto flujo de datos comprende una séptima unidad de datos de los primeros datos codificados de FEC y una octava unidad de datos de los segundos datos codificados de FEC, en donde obtener los primeros datos codificados de FEC y los segundos datos codificados de FEC al desentrelazar al menos dos flujos de datos ($i0, i1$) según una unidad de datos que comprende al menos dos bits comprende:

60 obtener la quinta unidad de datos de los primeros datos codificados de FEC, la séptima unidad de datos de los primeros datos codificados de FEC, la sexta unidad de datos de los segundos datos codificados de FEC y la octava unidad de datos de los segundos datos codificados de FEC al desentrelazar el tercer flujo de datos y el cuarto flujo de datos según la unidad de datos que comprende al menos dos bits.

65 22. El método de la reivindicación 21, en donde decodificar los primeros datos codificados de FEC y los segundos datos codificados de FEC comprende:

decodificar la quinta unidad de datos y la séptima unidad de datos; y
decodificar la sexta unidad de datos y la octava unidad de datos.

- 5 23. El método de una cualquiera de las reivindicaciones 17 a 22, en donde al menos dos flujos de datos comprenden un quinto flujo de datos, el quinto flujo de datos comprende una novena unidad de datos de los primeros datos codificados de FEC y una décima unidad de datos de los primeros datos codificados de FEC, en el que un orden de la novena unidad de datos y la décima unidad de datos en el quinto flujo de datos es el mismo que el orden de la novena unidad de datos y la décima unidad de datos en los primeros datos codificados de FEC.
- 10 24. El método de una cualquiera de las reivindicaciones 17 a 23, en donde dos bits uno al lado del otro en el flujo de datos entrelazados se desentrelazan a diferentes flujos de datos de al menos dos flujos de datos.
- 15 25. El método de una cualquiera de las reivindicaciones 17 a 24, en donde dos unidades de datos uno al lado del otro en uno cualquiera de al menos dos flujos de datos se desentrelazan a los primeros datos codificados de FEC y a los segundos datos codificados de FEC, respectivamente.
- 20 26. El método de una cualquiera de las reivindicaciones 17 a 25, en donde el método es realizado por una interfaz de Ethernet de alta velocidad.
27. El método de la reivindicación 26, en donde la interfaz de Ethernet de alta velocidad está configurada para 100GE, 400GE o 1TE.
28. Un dispositivo que comprende:
- 25 una unidad de recepción (21), configurada para recibir un flujo de datos entrelazados (i2'); y
una unidad de desentrelazado, configurada para obtener al menos dos flujos de datos (i0, i1) al desentrelazar el flujo de datos (i2') entrelazados según un bit, y configurada para obtener primeros datos codificados de Código de Errores Hacia Delante, FEC, y segundos datos codificados de FEC al desentrelazar al menos dos flujos de datos (i0, i1) según una unidad de datos que comprende al menos dos bits, y en el que en cada par de flujos de datos adyacentes de al menos dos flujos de datos (i00, i01), unidades de datos (00, 40; 41, 01) ubicadas en una misma posición en los flujos de datos del par respectivo comprenden una unidad de datos de los primeros datos codificados (00, 01) FEC y una unidad de datos de los segundos datos codificados (40, 41) FEC.
- 30 29. El dispositivo de la reivindicación 28, que comprende además:
- 35 una primera unidad de FEC, configurada para descodificar los primeros datos codificados de FEC; y
una segunda unidad de FEC, configurada para descodificar los segundos datos codificados de FEC.
- 40 30. El dispositivo de la reivindicación 29, en donde al menos dos flujos de datos comprenden un primer flujo de datos y un segundo flujo de datos, el primer flujo de datos comprende una primera unidad de datos de los primeros datos codificados de FEC y una segunda unidad de datos de los segundos datos codificados de FEC, el segundo flujo de datos comprende una tercera unidad de datos de los primeros datos codificados de FEC y una cuarta unidad de datos de los segundos datos codificados de FEC, y la unidad de desentrelazado está configurada para:
- 45 obtener la primera unidad de datos de los primeros datos codificados de FEC, la tercera unidad de datos de los primeros datos codificados de FEC, la segunda unidad de datos de los segundos datos codificados de FEC y la cuarta unidad de datos de los segundos datos codificados de FEC al desentrelazar el primer flujo de datos y el segundo flujo de datos según la unidad de datos que comprende al menos dos bits.
- 50 31. El dispositivo de la reivindicación 30, en donde la primera unidad de FEC se configura para descodificar la primera unidad de datos y la tercera unidad de datos; y la segunda unidad de FEC se configura para descodificar la segunda unidad de datos y la cuarta unidad de datos.
- 55 32. El dispositivo de una cualquiera de las reivindicaciones 29 a 31, en donde al menos dos flujos de datos comprenden un tercer flujo de datos y un cuarto flujo de datos, y el tercer flujo de datos comprende una quinta unidad de datos de los primeros datos codificados de FEC y una sexta unidad de datos de los segundos datos codificados de FEC, el cuarto flujo de datos comprende una séptima unidad de datos de los primeros datos codificados de FEC y una octava unidad de datos de los segundos datos codificados de FEC, y la unidad de desentrelazado está configurada para:
- 60 obtener la quinta unidad de datos de los primeros datos codificados de FEC, la séptima unidad de datos de los primeros datos codificados de FEC, la sexta unidad de datos de los segundos datos codificados de FEC y la octava unidad de datos de los segundos datos codificados de FEC al desentrelazar el tercer flujo de datos y el cuarto flujo de datos según la unidad de datos que comprende al menos dos bits.
- 65 33. El dispositivo de la reivindicación 32, en donde

la primera unidad de FEC se configura específicamente para decodificar la quinta unidad de datos y la séptima unidad de datos; y

la segunda unidad de FEC se configura específicamente para decodificar la sexta unidad de datos y la octava unidad de datos.

5 34. El dispositivo de una cualquiera de la reivindicación 28 a 33, en donde al menos dos flujos de datos comprenden un quinto flujo de datos, y el quinto flujo de datos comprende una novena unidad de datos de los primeros datos codificados de FEC y una décima unidad de datos de los primeros datos codificados de FEC, en donde un orden de
10 la novena unidad de datos y la décima unidad de datos en el quinto flujo de datos es el mismo que el orden de la novena unidad de datos y la décima unidad de datos en los primeros datos codificados de FEC.

35. El dispositivo de una cualquiera de las reivindicaciones 28 a 34, en donde dos bits uno al lado del otro en el flujo de datos entrelazados se desentrelazan a diferentes flujos de datos de al menos dos flujos de datos.

15 36. El dispositivo de una cualquiera de las reivindicaciones 28 a 35, en donde dos unidades de datos una al lado de la otra en uno cualquiera de al menos dos flujos de datos se desentrelazan a los primeros datos codificados de FEC y a los segundos datos codificados de FEC, respectivamente.

20 37. El dispositivo de una cualquiera de las reivindicaciones 28 a 36, en donde el dispositivo es uno o más chips, o el dispositivo es un dispositivo de red.

38. El dispositivo de una cualquiera de las reivindicaciones 28 a 37, en donde el dispositivo es una interfaz de Ethernet de alta velocidad.

25 39. El dispositivo de la reivindicación 38, en donde la interfaz de Ethernet de alta velocidad está configurada para 100GE, 400GE o 1TE.

40. Un sistema, que comprende:

30 un dispositivo según una cualquiera de las reivindicaciones 9 a 16, y un dispositivo según una cualquiera de las reivindicaciones 28 a 39.

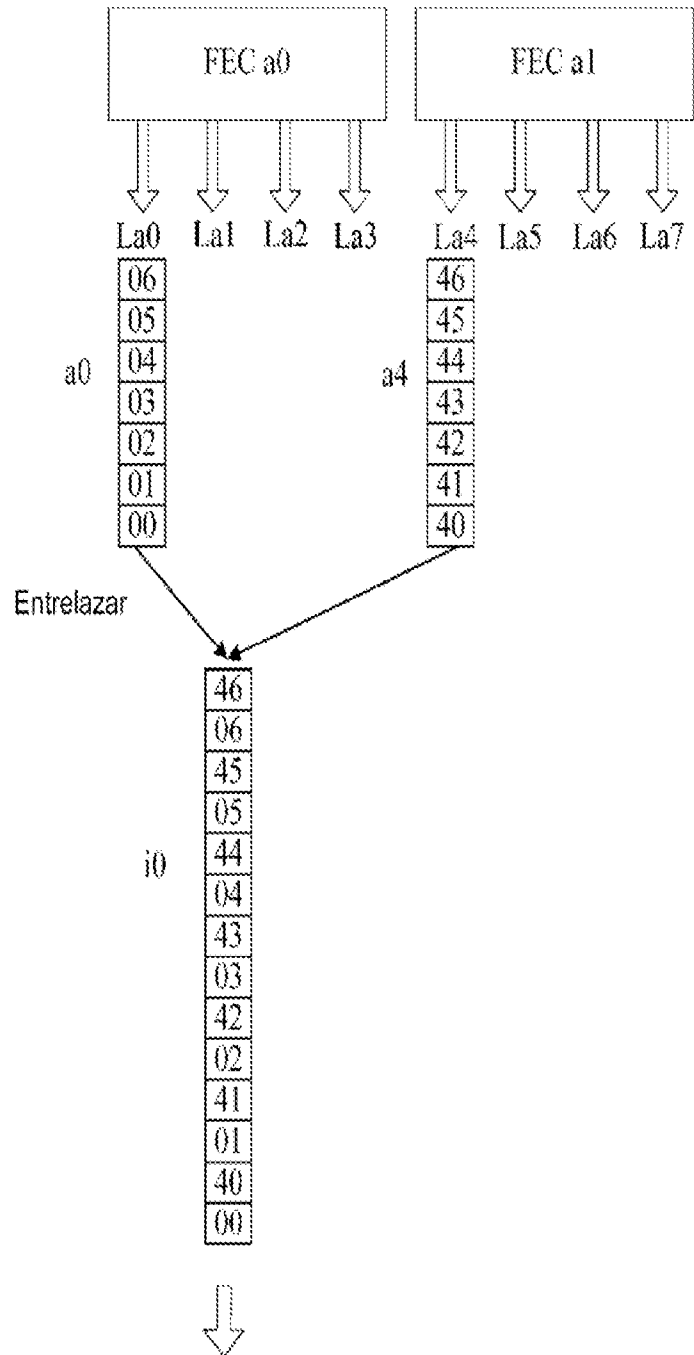


FIG. 1

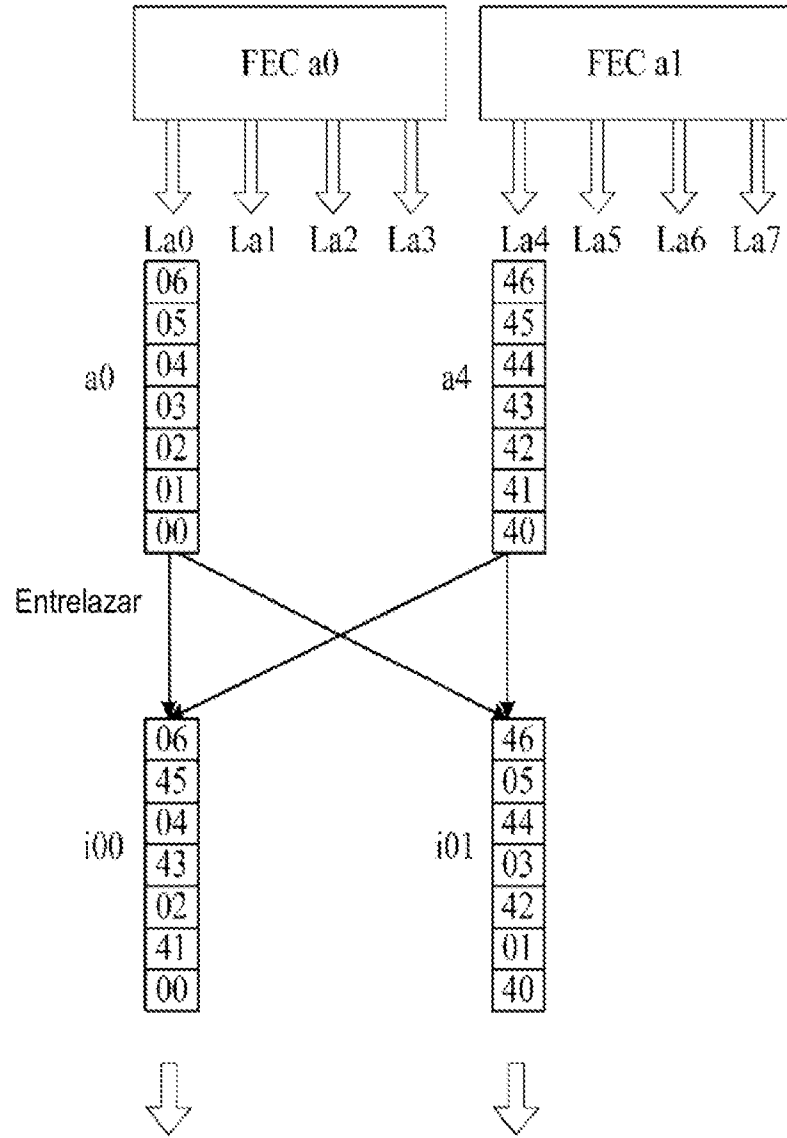


FIG. 2

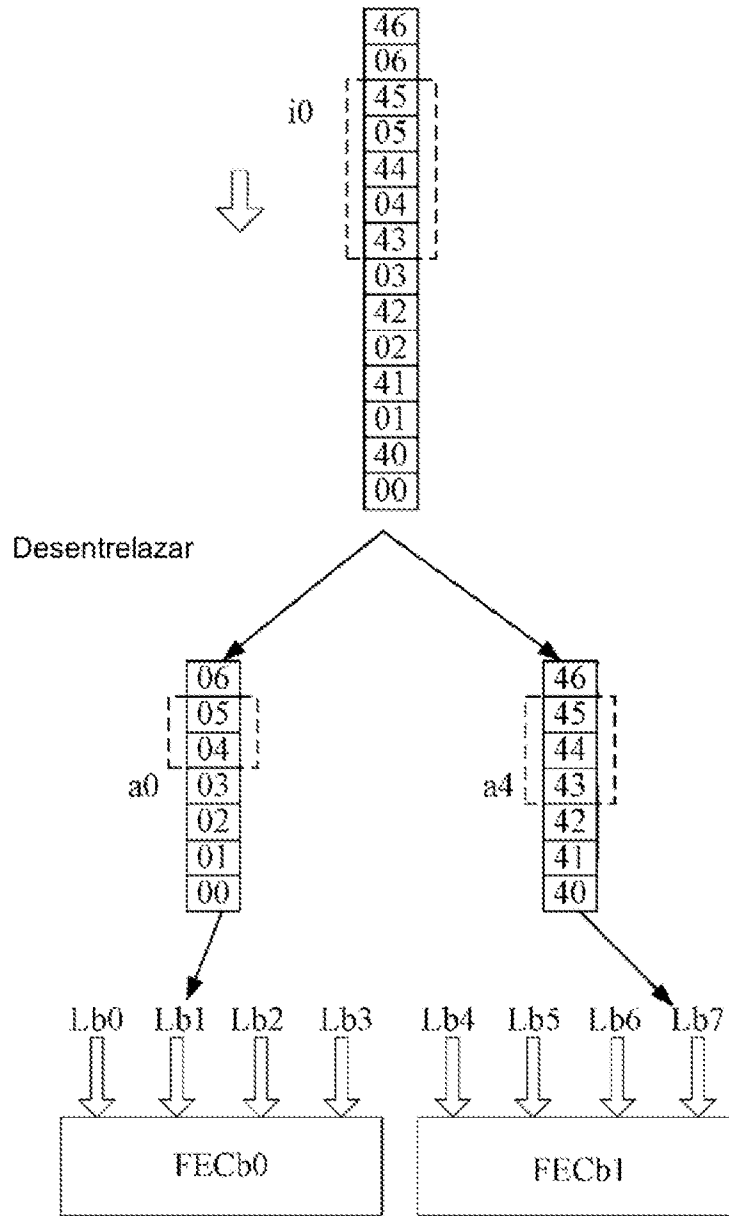


FIG. 3

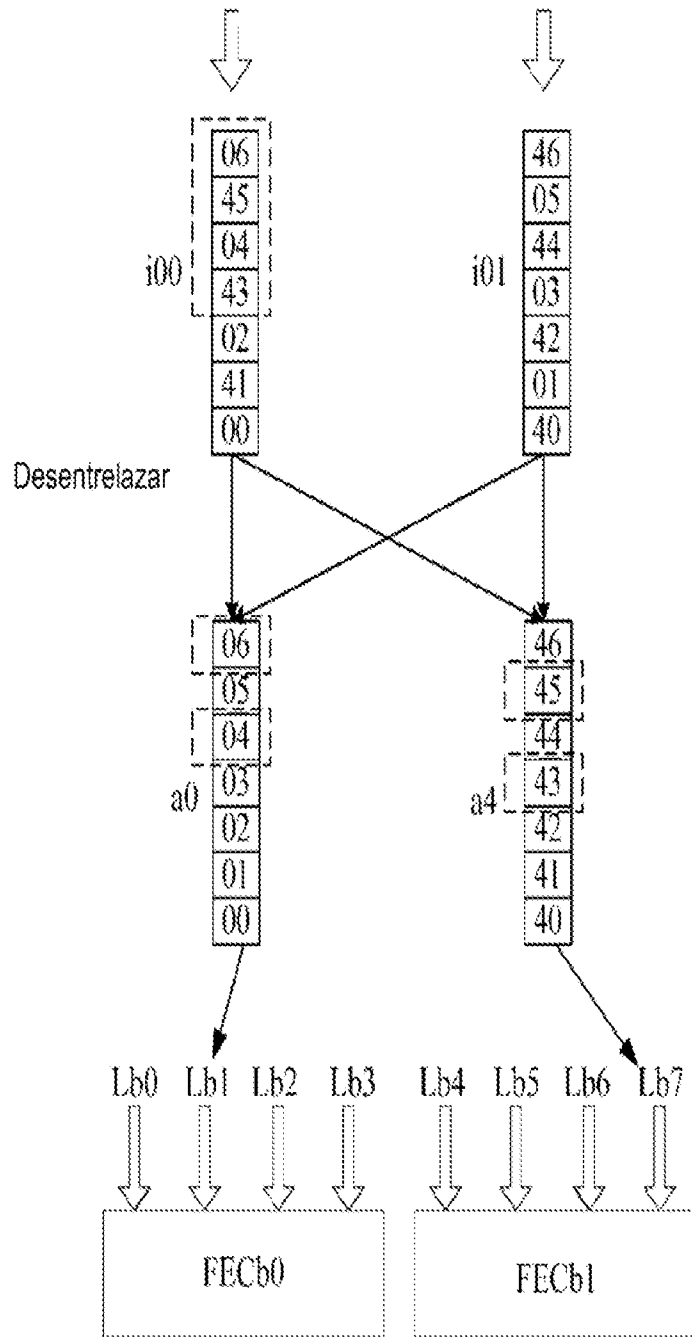


FIG. 4

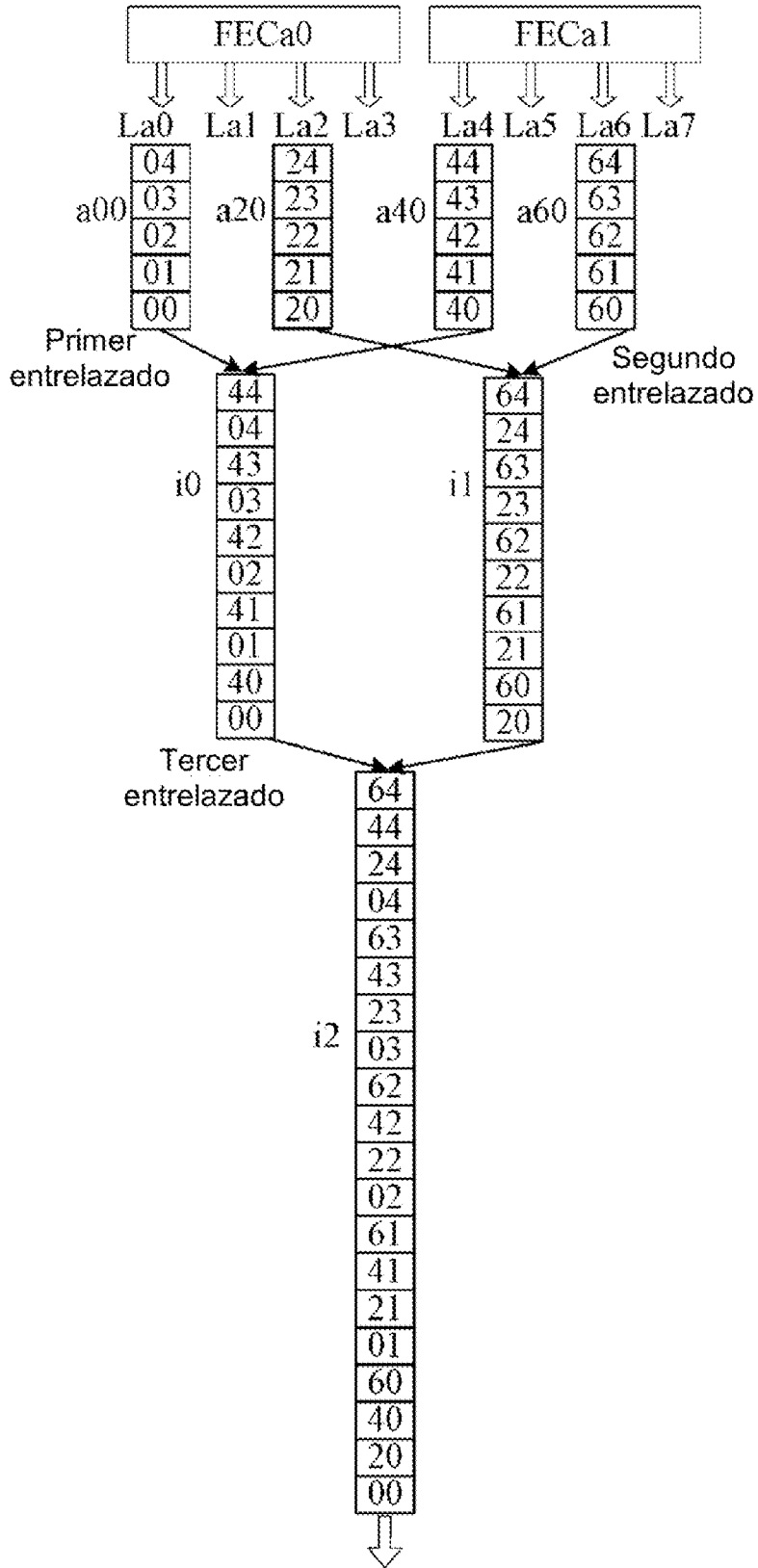


FIG. 5

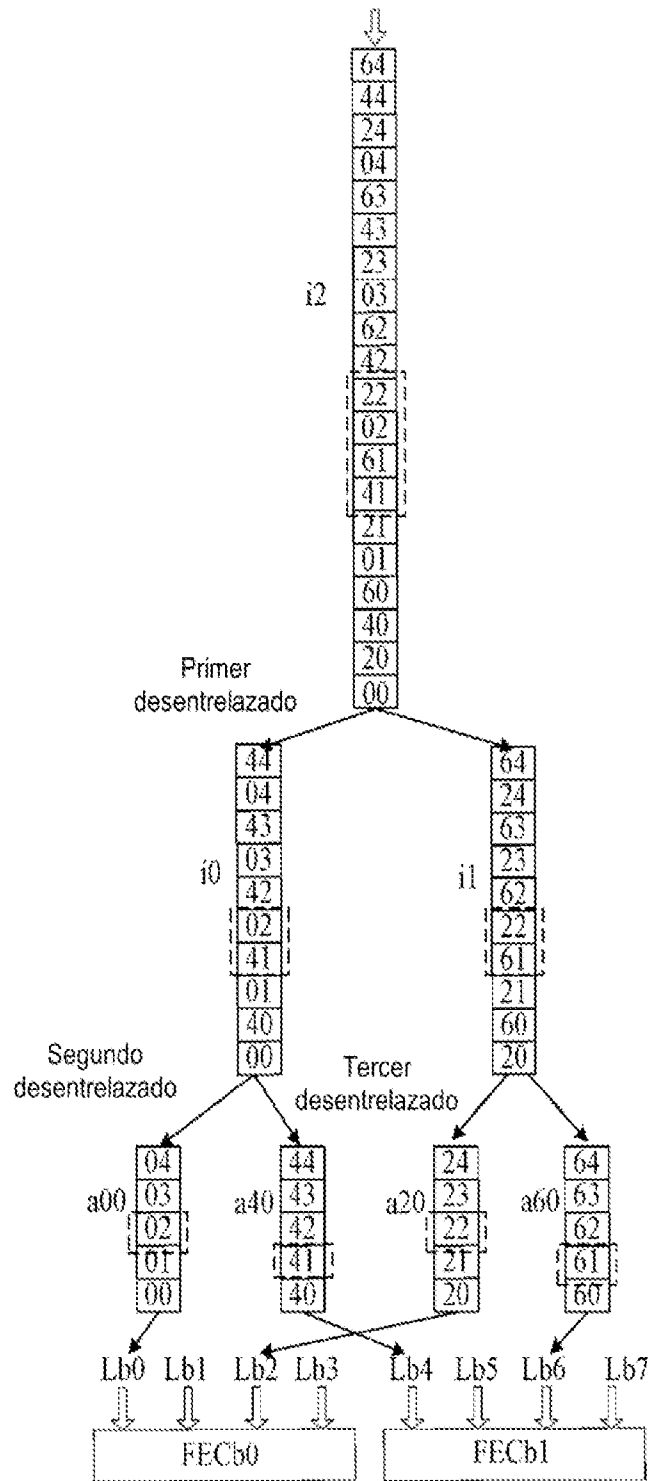


FIG. 6

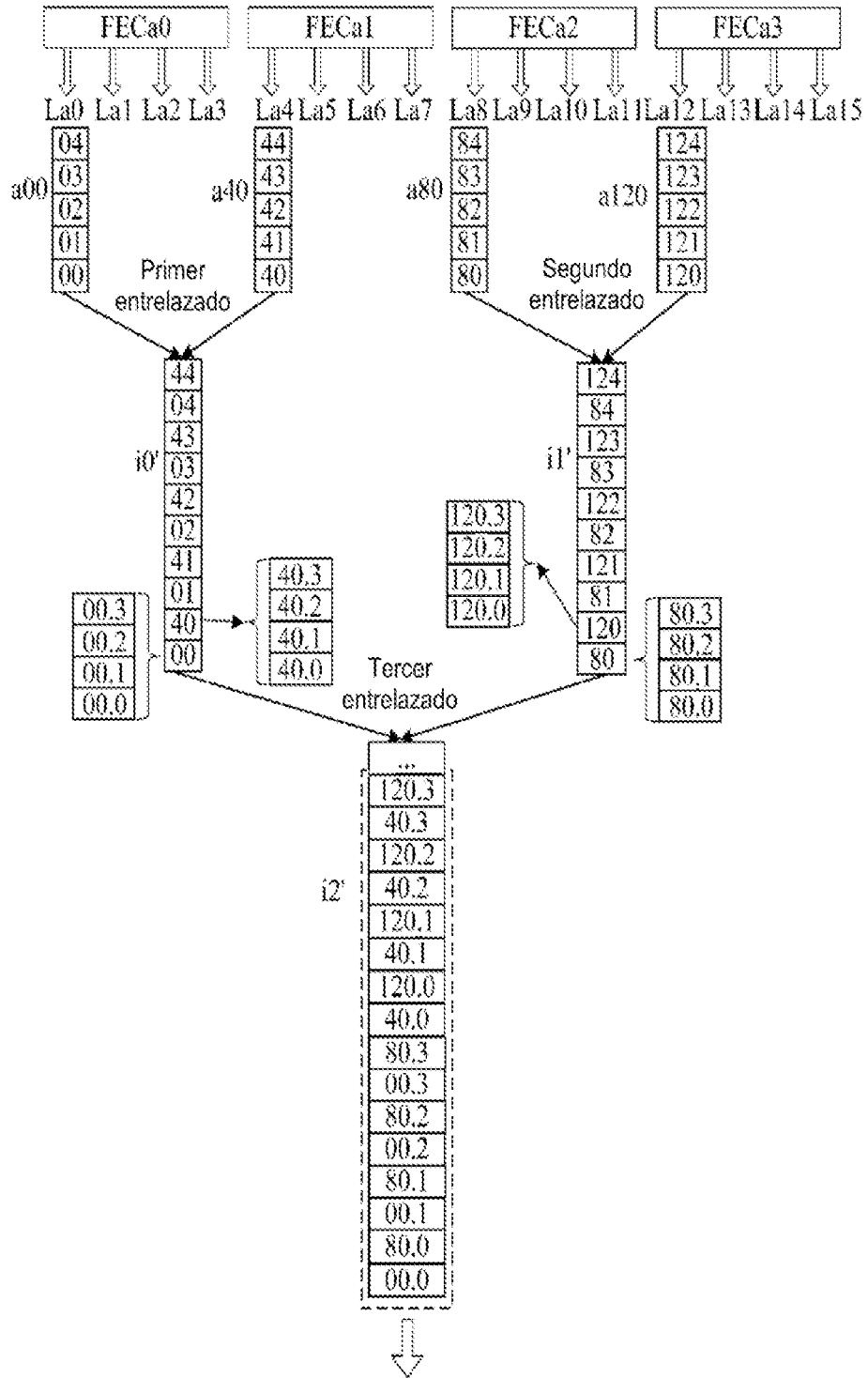


FIG. 7

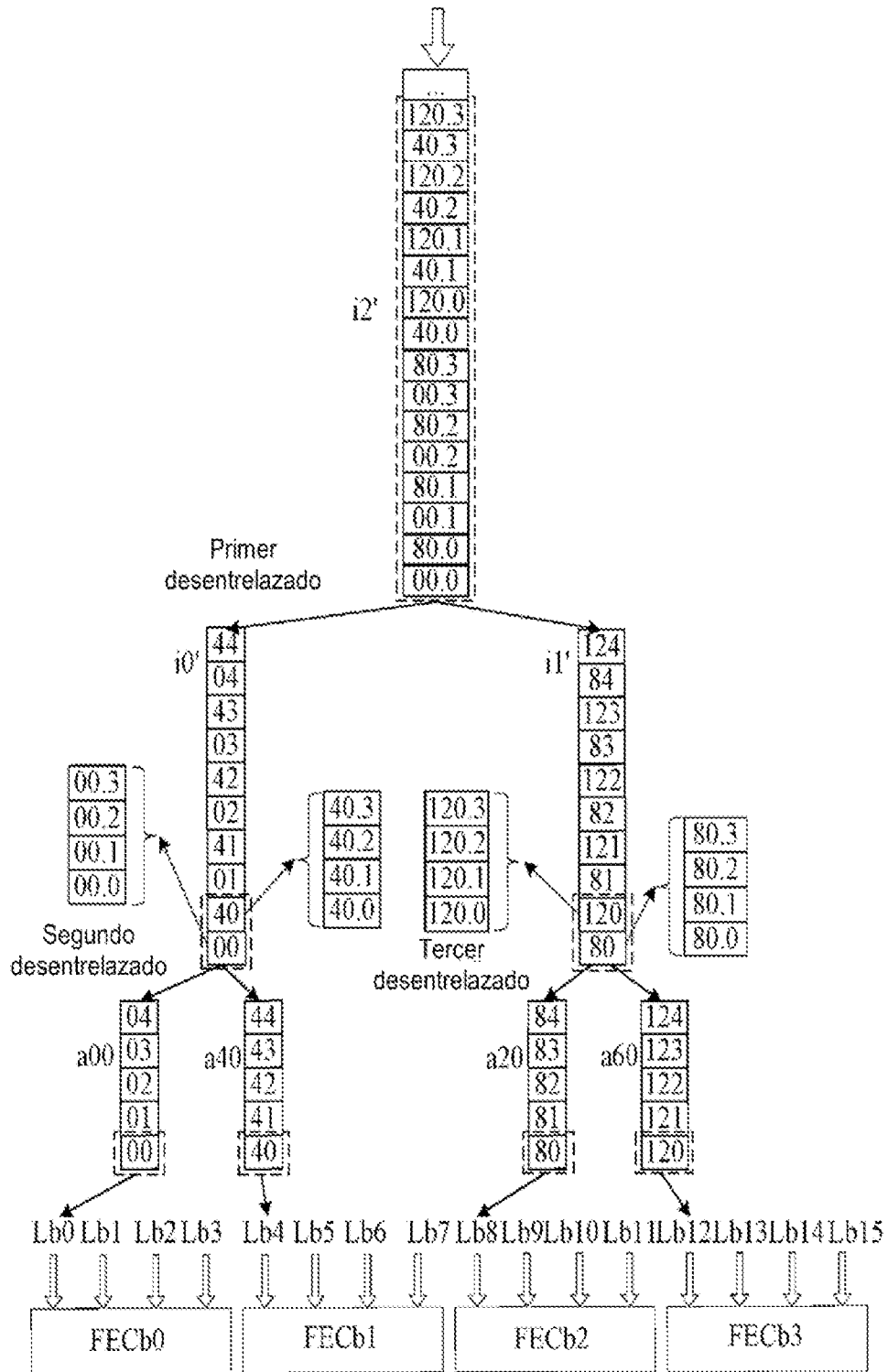


FIG. 8

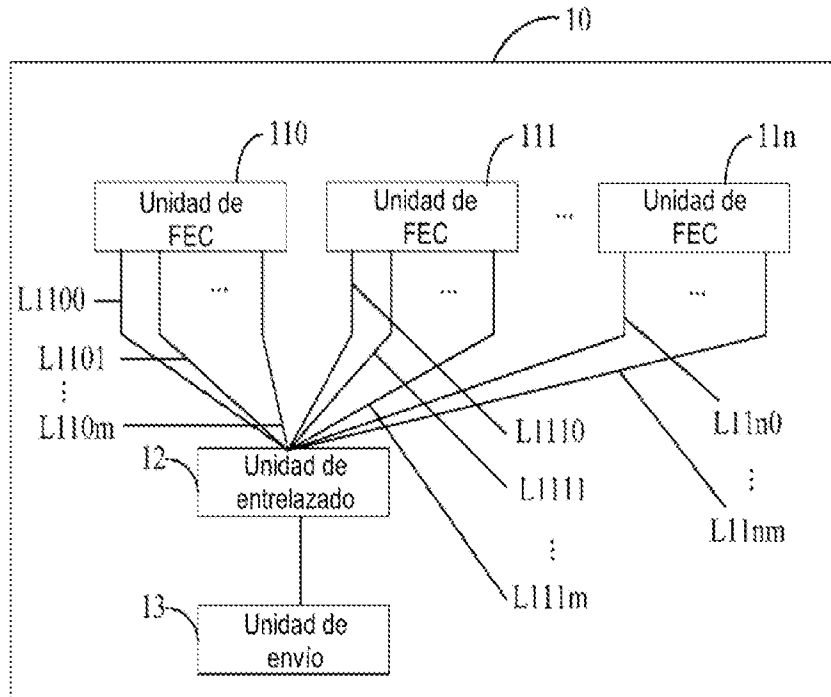


FIG. 9

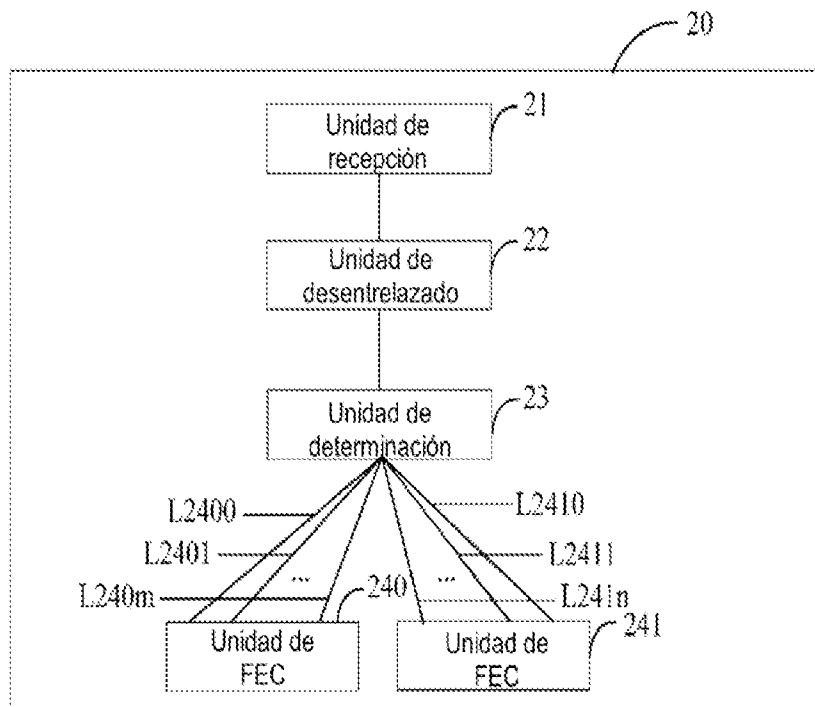


FIG. 10

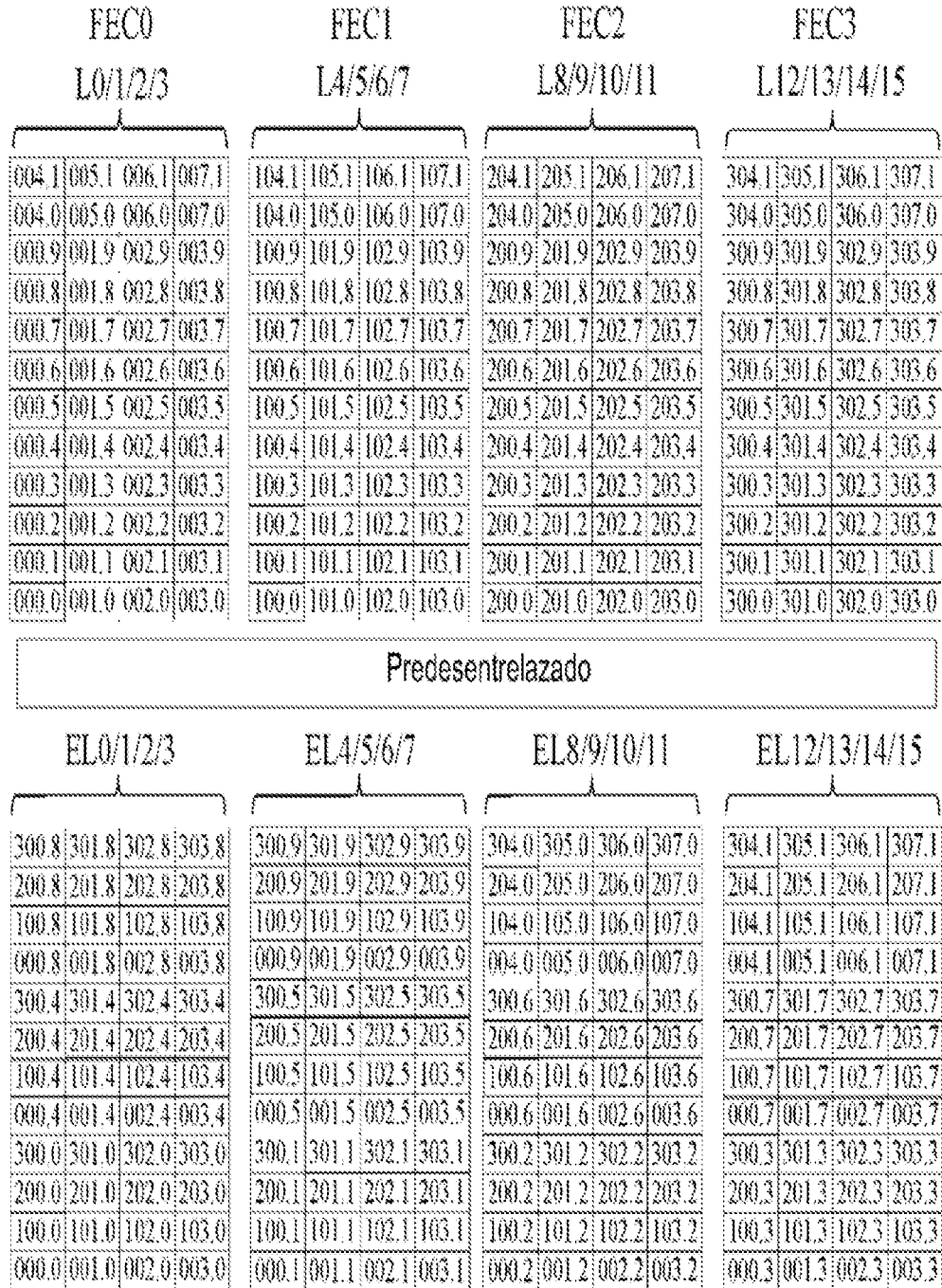


FIG. 11

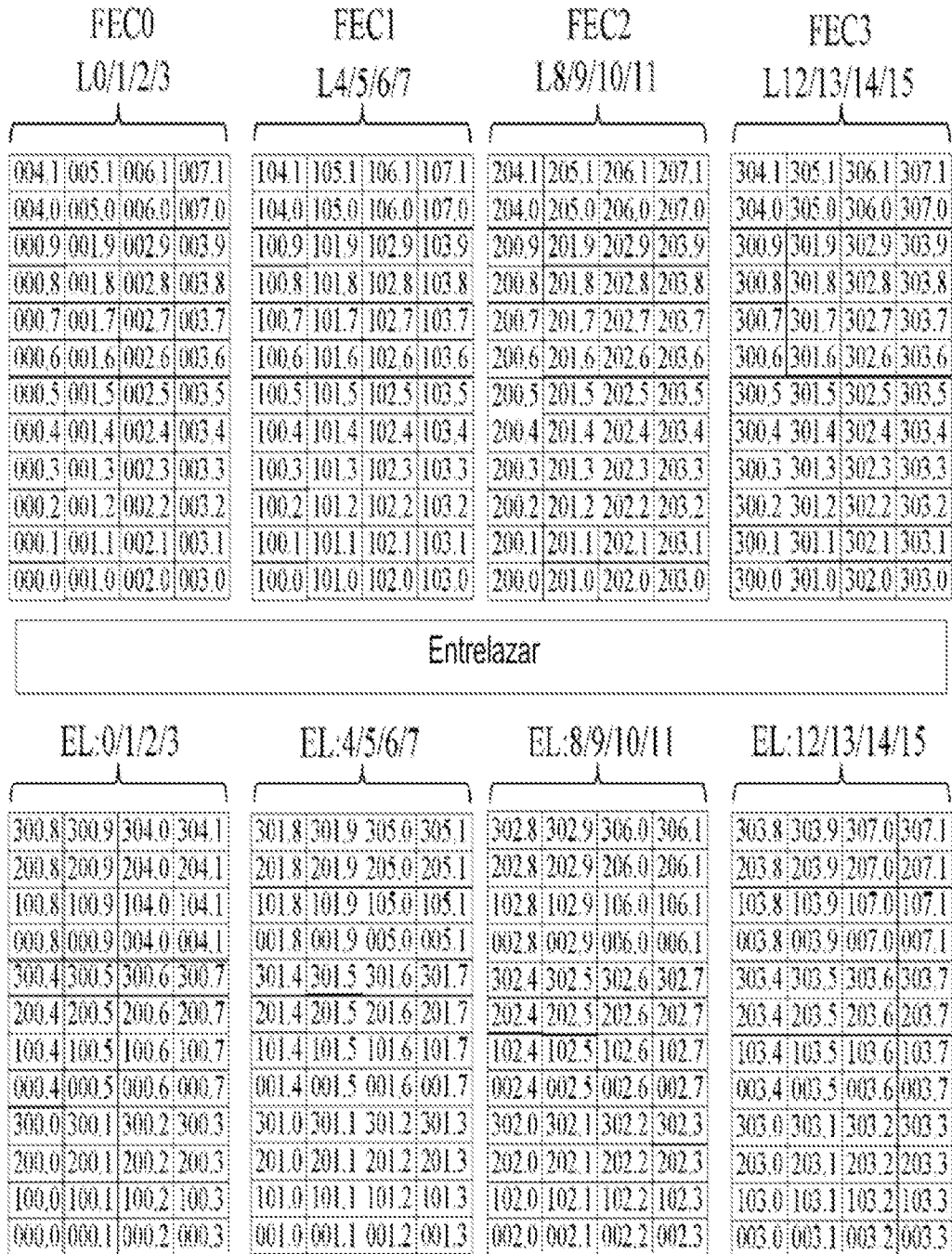


FIG. 12

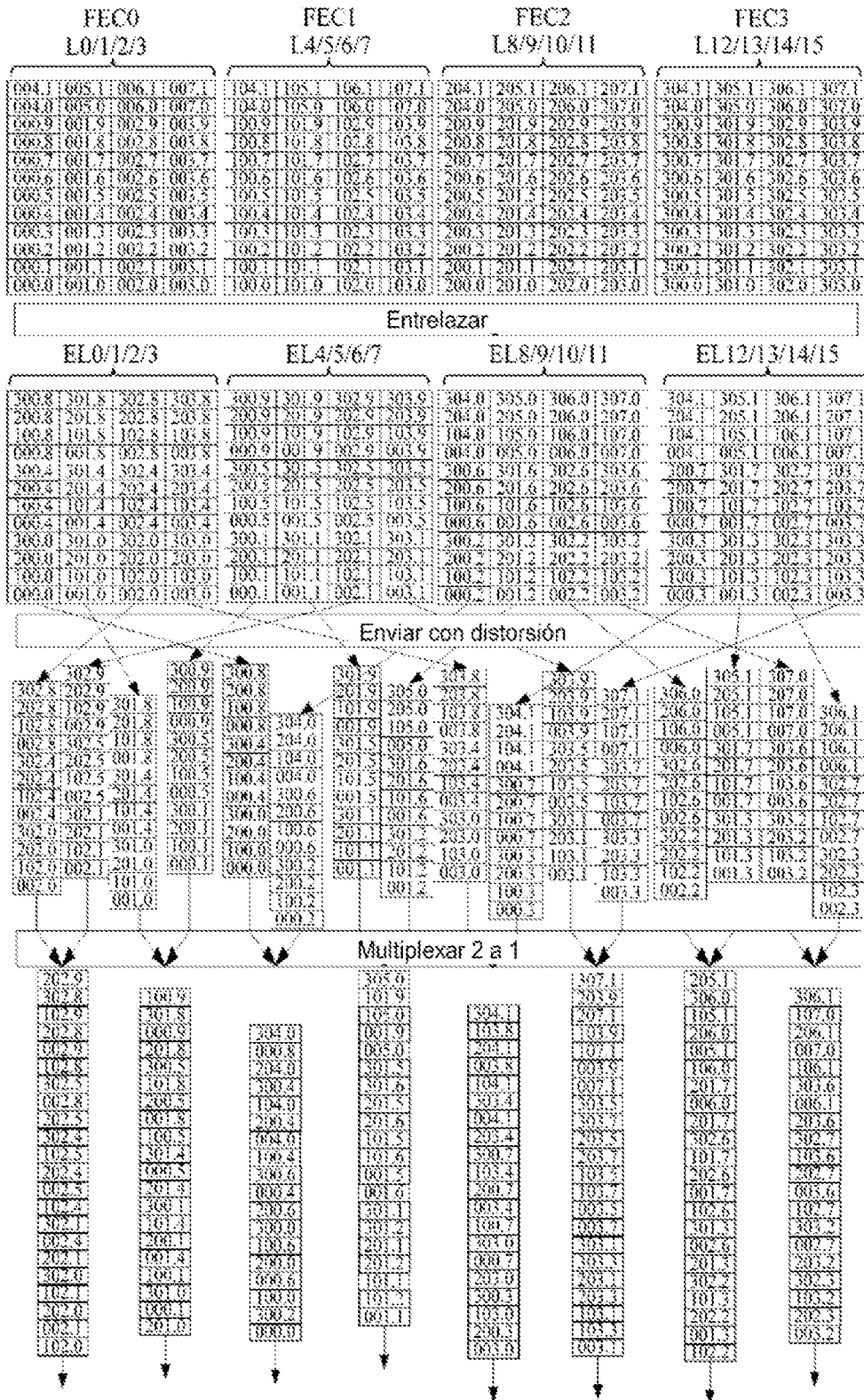


FIG. 13

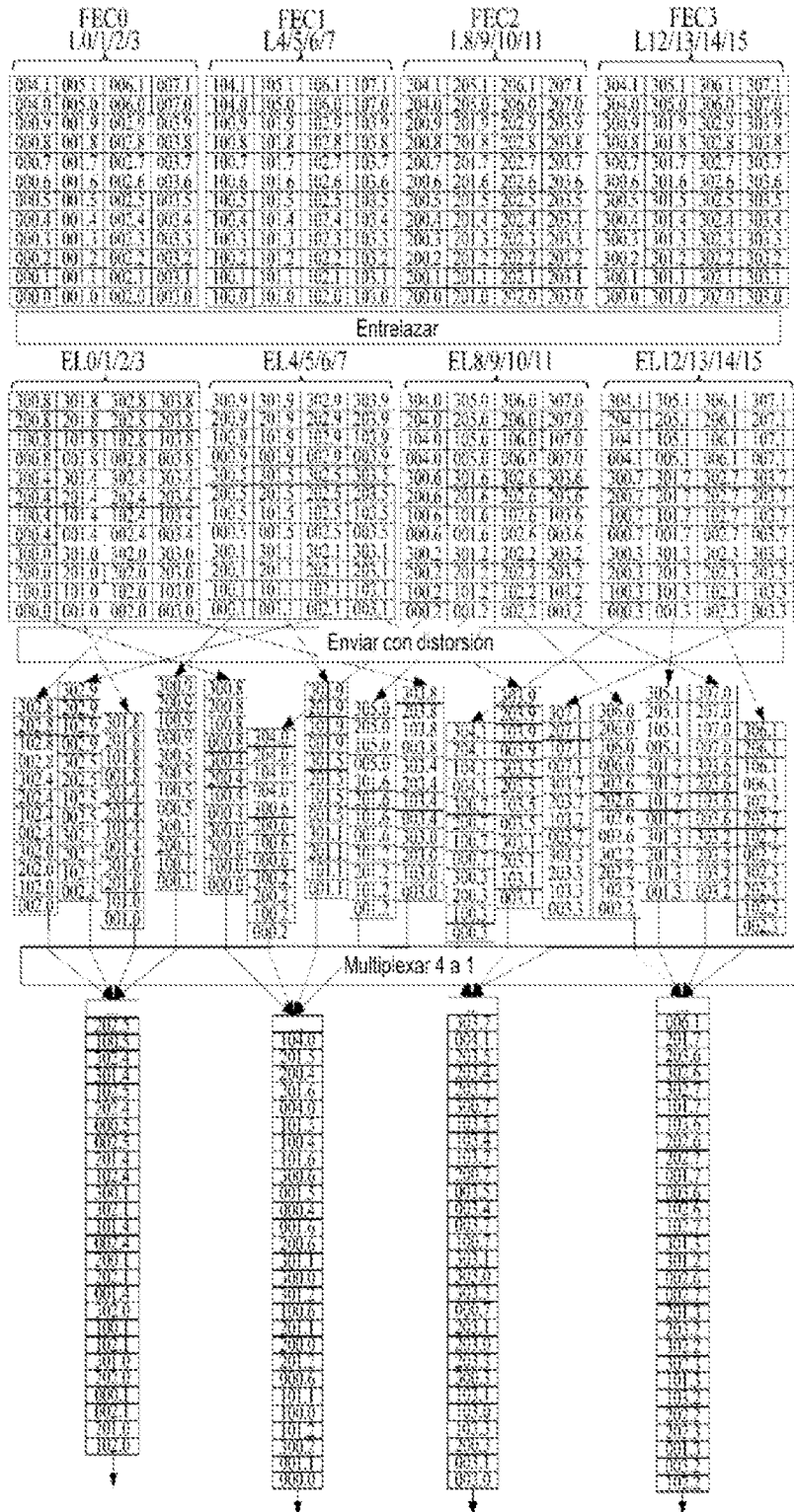


FIG. 14

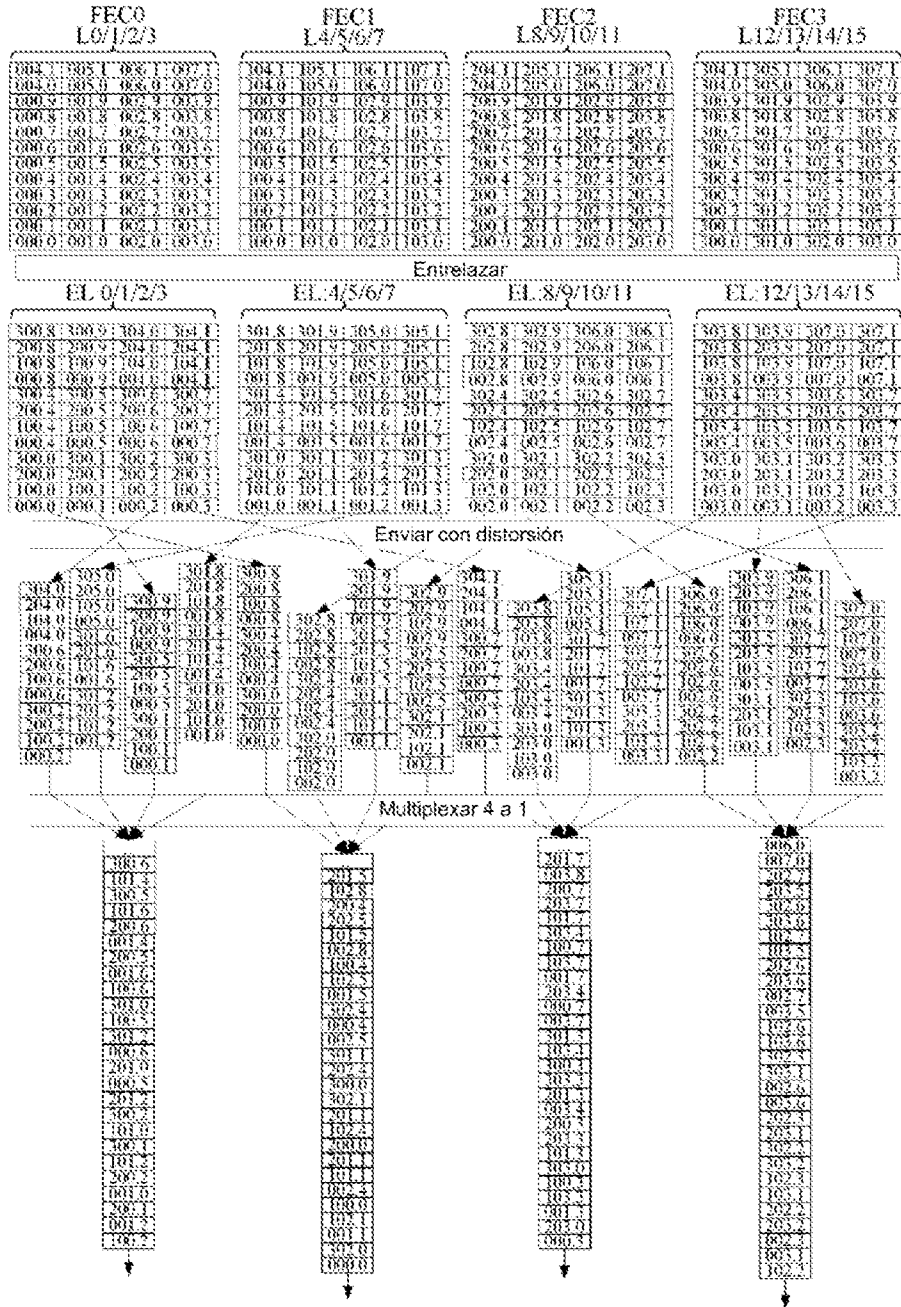


FIG. 15