

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **3 024 210**

51 Int. Cl.:

H03M 13/11 (2006.01)

H03M 13/29 (2006.01)

H03M 13/00 (2006.01)

H03M 13/09 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **04.11.2016 PCT/CN2016/104744**

87 Fecha y número de publicación internacional: **16.11.2017 WO17193558**

96 Fecha de presentación y número de la solicitud europea: **04.11.2016 E 16901513 (8)**

97 Fecha y número de publicación de la concesión europea: **23.04.2025 EP 3457574**

54 Título: **Método y dispositivo de procesamiento de datos para el código de LDPC estructurado**

30 Prioridad:

13.05.2016 CN 201610322409
30.09.2016 CN 201610879343

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
04.06.2025

73 Titular/es:

ZTE CORPORATION (100.00%)
ZTE Plaza, Keji Road South, Hi-Tech Industrial
Park, Nanshan District
Shenzhen, Guangdong 518057, CN

72 Inventor/es:

LI, LIGUANG;
XU, JUN y
XU, JIN

74 Agente/Representante:

UNGRÍA LÓPEZ, Javier

ES 3 024 210 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 procesamiento de datos para el código de LDPC estructurado

Campo técnico

La presente divulgación se refiere a la tecnología de comunicación y, más particularmente, a un método y un aparato para el procesamiento de datos con códigos de Verificación de Paridad de Baja Densidad (LDPC) estructurados.

Antecedentes

Un sistema de comunicación digital normalmente incluye tres porciones: un transmisor, un canal y un receptor. El transmisor puede realizar la codificación de canal en una secuencia de información para obtener palabras codificadas, entrelazarlas palabras codificadas, mapear los bits entrelazados en símbolos modulados y luego procesar y transmitir los símbolos modulados en base a la información del canal de comunicación. El canal puede tener una respuesta de canal particular debido a varios factores, tales como la multirruta y los movimientos, lo que hace que se distorsione la transmisión de datos. Más aún, los ruidos y las interferencias pueden deteriorar aún más la transmisión de datos. El receptor recibe los datos de símbolos modulados que han pasado a través del canal. En este caso, los datos de símbolos modulados se han distorsionado y se requerirá algún procesamiento antes de que se pueda restaurar la secuencia de información original.

De acuerdo con el esquema de codificación utilizado para la secuencia de información en el transmisor, el receptor puede realizar el procesamiento correspondiente en los datos recibidos, a fin de restaurar la secuencia de información original de manera confiable. El esquema de codificación debe ser visible tanto para el transmisor como para el receptor. En general, el esquema de procesamiento de codificación es una codificación basada en la Corrección de Errores Hacia Adelante (FEC), que agrega alguna información redundante a la secuencia de información. El receptor puede utilizar la información redundante para restaurar la secuencia de información original de forma fiable.

Algunos esquemas de codificación de FEC comunes incluyen la codificación convolucional, codificación turbo y codificación de LDPC. En un proceso de codificación de FEC, una secuencia de información de k bits está codificada en FEC en palabras codificadas en FEC de n bits (con n-k bits redundantes), con una tasa de codificación de FEC de k/n. La codificación convolucional puede codificar fácilmente bloques que tengan cualesquier tamaños arbitrarios. En la codificación turbo, se pueden admitir diferentes tamaños de secuencias de información al utilizar dos componentes de codificación utilizados para procesar la secuencia de información y un esquema de entrelazado de código capaz de admitir diferentes tamaños.

La codificación de LDPC es una codificación de bloques lineales que se puede definir utilizando una matriz de verificación de paridad muy dispersa o un grafo bipartito. La baja complejidad de la codificación/decodificación es posible gracias a la dispersión de la matriz de verificación, de tal manera que la LDPC puede ser aplicable en la práctica. Se ha demostrado mediante diversas prácticas y teorías que la codificación de LDPC es la codificación de canal que tiene el rendimiento óptimo en un canal de Ruido Gaussiano Blanco Aditivo (AWGN). Su rendimiento está muy cerca del límite de Shannon y es superior a los de la codificación convolucional y la codificación turbo.

En particular, los códigos de LDPC estructurados se han convertido en una aplicación principal debido a su característica estructurada. Se han aplicado ampliamente, por ejemplo, en IEEE 802.11ac, IEEE 802.11ad, IEEE 802.11aj, IEEE 802.16e, IEEE 802.11n, DVB, comunicaciones de microondas y comunicaciones de fibra óptica. Dichos códigos de LDPC estructurados tienen una matriz de verificación de paridad H que tiene $mb \times z$ filas y $nb \times z$ columnas. Está compuesto de $mb \times nb$ matrices de bloques. Las respectivas matrices de bloques representan diferentes potencias de una matriz de permutación básica $z \times z$ y son matrices desplazadas cíclicamente de la matriz de identidad. La matriz H tiene la siguiente forma:

$$H = \begin{bmatrix} P^{h_{0,0}^b} & P^{h_{0,1}^b} & P^{h_{0,2}^b} & \Lambda & P^{h_{0,nb}^b} \\ P^{h_{1,0}^b} & P^{h_{1,1}^b} & P^{h_{1,2}^b} & \Lambda & P^{h_{1,nb}^b} \\ \Lambda & \Lambda & \Lambda & \Lambda & \Lambda \\ P^{h_{mb,0}^b} & P^{h_{mb,1}^b} & P^{h_{mb,2}^b} & \Lambda & P^{h_{mb,nb}^b} \end{bmatrix} = P^{H_b}$$

Sea $h_{ij}^b = -1$, $P^{h_{ij}^b} = 0$, que es una matriz de ceros. Si h_{ij}^b es un número entero mayor o igual que 0, sea $P^{h_{ij}^b} = (P^{h_{ij}^b})$, donde P es una matriz de permutación estándar $z \times z$, como sigue:

$$P = \begin{bmatrix} 0 & 1 & 0 & \Lambda & 0 \\ 0 & 0 & 1 & \Lambda & 0 \\ \Lambda & \Lambda & \Lambda & \Lambda & \Lambda \\ 0 & 0 & 0 & \Lambda & 1 \\ 1 & 0 & 0 & \Lambda & 0 \end{bmatrix}.$$

De esta manera, el exponente h_{ij}^b puede identificar de forma única cada matriz de bloques. Si una matriz de bloques es una matriz de ceros, generalmente se puede representar como -1. Si se obtiene una matriz de bloques al desplazar cíclicamente la matriz de identidad en s, h_{ij}^b es igual a s. Por lo tanto, todos los ' h_{ij}^b 's pueden constituir una matriz de verificación básica Hb. z indica el número de dimensiones de la matriz de permutación estándar y se denomina como factor de expansión en el presente documento. En este caso, los códigos de LDPC de estructura se pueden determinar de forma única a partir de la matriz de verificación básica Hb y el factor de expansión z. La matriz de verificación básica Hb tiene varios parámetros: mb , nb y kb , donde mb es el número de filas de la matriz de verificación básica (o el número de columnas de verificación de la matriz de verificación básica), nb es el número de columnas de la matriz de verificación básica y $kb = nb - mb$ es el número de columnas del sistema de la matriz de verificación básica.

Por ejemplo, cuando una matriz de verificación básica Hb (con 2 filas y 4 columnas) es como sigue y el factor de expansión z es igual a 4:

$$\begin{bmatrix} 0 & 1 & 0 & -1 \\ 2 & 1 & 2 & 1 \end{bmatrix},$$

la matriz de verificación de paridad será:

$$H = \left[\begin{array}{cccc|cccc|cccc|cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \end{array} \right].$$

Se puede aplicar la decodificación en capas, es decir, un esquema de decodificación parcialmente paralelo a los códigos LPDC. La matriz de verificación de paridad anterior tiene 8 filas, lo que significa que hay 8 códigos de verificación de paridad. En la decodificación, cada matriz de verificación de paridad se debe decodificar individualmente. Las actualizaciones de datos que se completan para los 8 códigos de verificación de paridad se denominan como una iteración. Durante cada iteración, si se utiliza la decodificación parcialmente paralela (es decir, en capas), por ejemplo, con un grado de paralelismo, p, lo que significa que los códigos de verificación de paridad p se actualizan en paralelo, los códigos de verificación de paridad p actuales y siguientes operarán utilizando el mismo módulo de actualización. En este caso, la complejidad del decodificador puede ser mucho menor. Más aún, en la decodificación en capas, los datos actualizados actualmente se pueden utilizar para actualizar los datos en la siguiente capa, lo que significa que se requerirá un número menor de iteraciones y el rendimiento de decodificación será mayor. Para el H anterior como un ejemplo, si el grado de paralelismo es 2, por cada 4 filas de la matriz de verificación de paridad (una fila de la matriz de verificación básica), dos (el grado de paralelismo) de los códigos de verificación de paridad se actualizarán en paralelo.

En general, la matriz de verificación de paridad puede ser muy grande y difícil de almacenar y procesar. Sería mucho más fácil almacenar la matriz de verificación básica.

De acuerdo con el conocimiento matemático fundamental, hay un número infinito de números primos. Un número primo es un número natural mayor que 1 que no se puede formar al multiplicar dos números naturales más pequeños. Normalmente, un número primo es un número entero mayor que 1, por ejemplo, 2, 3, 5, 7, 11, 13, 17, 19, etc.

En casi todos los métodos de codificación de FEC, el codificador tiene una complejidad relativamente baja y la complejidad radica básicamente en el decodificador en el receptor. La codificación de LDPC tiene una mayor velocidad de decodificación y rendimiento, ya que tiene la característica de paralelismo de tal manera que se puede aplicar la

decodificación paralela. Sin embargo, normalmente la matriz de verificación básica para los códigos de LDPC solo se puede adaptar a longitudes particulares y puede que no sea fácil acomodar varias longitudes flexibles de bloques de código. Será más complicado acomodar diferentes longitudes de bloques de código.

5 Actualmente no existen soluciones efectivas al problema en la técnica relacionada asociada con la baja flexibilidad en el procesamiento de secuencias de datos con codificación de LDPC.

10 El documento CN 102 412 842 A se refiere a un método y dispositivo para codificar un código de verificación de paridad de baja densidad. El método comprende las etapas de: comparar una tasa de código madre y una tasa de código de transmisión que se admiten por una matriz básica uniforme; determinar la longitud K' de un bit de agrupación de información de entrada de acuerdo con el resultado de una comparación; determinar una matriz de codificación del código de verificación de paridad de baja densidad (LDPC) de acuerdo con la matriz básica uniforme y un factor de expansión; determinar un bit de información que se codificará de acuerdo con el bit de agrupación de información recibido con la longitud de K' ; y realizar la codificación de LDPC en el bit de información que se va a codificar al utilizar la matriz de codificación, y procesar un bit de palabra de código madre obtenido a través de la codificación para adquirir un bit de palabra de código que se va a transmitir.

15 El documento US 2008/178065 A1 se refiere a técnicas para admitir la codificación y decodificación de verificación de paridad de baja densidad (LDPC). La codificación y decodificación de LDPC de paquetes de diferentes tamaños se admite con un conjunto de matrices de verificación de paridad base de diferentes dimensiones y un conjunto de valores de elevación de diferentes potencias de dos. Se utiliza una matriz de verificación de paridad básica G de dimensión $m \times n$ para codificar un paquete de $k = n - m$ bits de información para obtener una palabra de código de n bits de código.

Resumen

La presente invención está definida por las reivindicaciones independientes.

Las reivindicaciones dependientes se refieren a las realizaciones preferidas.

25 Las realizaciones de la presente divulgación proporcionan un método y un aparato para el procesamiento de datos con códigos de LDPC estructurados, capaces de resolver el problema en la técnica relacionada asociada con la baja flexibilidad en el procesamiento de secuencias de datos con codificación de LDPC.

30 De acuerdo con una realización de la presente divulgación, se proporciona un método para el procesamiento de datos con códigos de verificación de paridad de baja densidad (LDPC) estructurados. El método incluye: obtener un tamaño de bloque de código para la codificación de LDPC estructurada; determinar un factor de expansión de codificación z en base a al menos uno del tamaño de bloque de código, un parámetro k_b de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica que tenga m filas y n columnas; y codificar una secuencia de datos que se va a codificar, o decodificar una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación, donde $k_b = n - m$ y cada uno de k_b , p , m , z y n es un número entero mayor que 1.

35 Opcionalmente, la operación de determinación del factor de expansión de codificación se basa al menos en el tamaño de bloque de código CBS , el parámetro k_b de la matriz de verificación básica y el valor entero positivo p incluye: determinar el factor de expansión de codificación de acuerdo con: $\lceil CBS / (k_b \times p) \rceil \times p$, donde $\lceil \cdot \rceil$ denota una operación de techo.

40 Opcionalmente, la operación de obtención del tamaño de bloque de código para la codificación de LDPC estructurada incluye: que el tamaño de bloque de código sea igual a un valor obtenido al multiplicar el parámetro k_b de la matriz de verificación básica por uno de un conjunto de números naturales ordenados en orden ascendente.

Opcionalmente, el conjunto de números naturales ordenados en orden ascendente incluye una pluralidad de subconjuntos, cada uno de los cuales tiene una primera diferencia igual entre los elementos vecinos.

45 Opcionalmente, la primera diferencia para el i -ésimo subconjunto en el conjunto de números naturales ordenados en orden ascendente es menor que la primera diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo.

50 Opcionalmente, la operación de determinación del factor de expansión de codificación basado en el tamaño de bloque de código incluye: el tamaño de bloque de código que es uno de un conjunto de tamaños de bloque de código, el conjunto de tamaños de bloque de código es un primer conjunto de números naturales almacenados en orden ascendente; y el factor de expansión de codificación es uno de un conjunto de factores de expansión de codificación, el conjunto de factores de expansión de codificación es un segundo conjunto de números naturales almacenados en orden ascendente.

Opcionalmente, el conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times k_b$ y el último de los elementos sucesivos es menor o igual

que $z(j+1) \times kb$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo, $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación, j es un número entero positivo y kb es el parámetro de la matriz de verificación básica.

5 Opcionalmente, el factor de expansión de codificación es $z(j+1)$ cuando el tamaño de bloque de código obtenido es uno de los elementos sucesivos.

Opcionalmente, el factor de expansión de codificación se determina como $z(j+1)$ cuando una relación entre el tamaño de bloque de código y el parámetro kb de la matriz de verificación básica es mayor que $z(j)$ y menor o igual que $z(j+1)$, donde $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación.

10 Opcionalmente, el conjunto de factores de expansión de codificación incluye una pluralidad de subconjuntos, cada uno tiene una segunda diferencia igual entre los elementos vecinos, y la segunda diferencia para el h -ésimo subconjunto de la pluralidad de subconjuntos es menor que la segunda diferencia para el $(h+1)$ -ésimo subconjunto, donde h es un número entero positivo.

Opcionalmente, cada uno de los elementos del conjunto de factores de expansión de codificación que es mayor que el valor entero positivo p es n veces el valor entero positivo p , donde n es un número entero positivo.

15 Opcionalmente, el conjunto de tamaños de bloques de código incluye al menos los productos del parámetro kb de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación, y números enteros espaciados por un número entero positivo B entre todos los productos.

20 Opcionalmente, el método incluye además, antes de la codificación de la secuencia de datos que se va a codificar en base a la matriz de verificación básica y el factor de expansión de codificación: agregar bits de relleno a una primera secuencia de datos al dividir la primera secuencia de datos en una pluralidad de subsecuencias de datos y agregar bits de relleno a cada una de las pluralidades de subsecuencias de datos, de tal manera que todas las subsecuencias de datos con los bits de relleno constituyen la secuencia de datos que se va a codificar.

25 Opcionalmente, el método incluye además, antes de codificar la secuencia de datos que se va a codificar en base a la matriz de verificación básica y el factor de expansión de codificación: agregar bits de relleno a una segunda secuencia de datos al dividir la segunda secuencia de datos en kb subsecuencias de datos, cada una de las cuales incluye z o menos bits, y agregar bits de relleno a cada una de las subsecuencias de datos de tal manera que cada una de las subsecuencias de datos incluya z bits y todas las subsecuencias de datos con los bits de relleno constituyen la secuencia de datos que se va a codificar, donde kb es el parámetro de la matriz de verificación básica y z es el factor de expansión de codificación.

30 Opcionalmente, el método incluye además, antes de agregar los bits de relleno a la primera o segunda secuencia de datos: agregar una secuencia de Verificación de Redundancia Cíclica (CRC) de bits L a una tercera secuencia de datos para obtener la primera o segunda secuencia de datos, donde L es un número entero mayor o igual que 0.

35 Opcionalmente, el parámetro kb de la matriz de verificación básica es una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica, donde kb es un número entero mayor o igual que 4 y menor o igual que 64.

Opcionalmente, el valor entero positivo p es un grado de paralelismo de la decodificación de LDPC.

Opcionalmente, el valor entero positivo p es un número entero positivo fijo, y el grado p de paralelismo de la decodificación de LDPC incluye: el valor entero positivo p que es una potencia entera positiva de 2 o un producto de una potencia entera positiva de 2 y un número primo.

40 Opcionalmente, el valor entero positivo p es un elemento de un subconjunto P , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{\text{máx}}$, donde $P_{\text{máx}}$ es un número entero mayor que 3 y menor o igual que 1024.

45 Opcionalmente, todos los valores del valor entero positivo p constituyen un subconjunto $n\text{Set}$, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

50 Opcionalmente, el subconjunto $n\text{Set}$ incluye uno de: un subconjunto de los primeros F elementos del conjunto de números enteros positivos N_s , un subconjunto de los últimos F elementos del conjunto de números enteros positivos N_s , un subconjunto de F números primos en el conjunto de números enteros positivos N_s , un subconjunto de F números impares en el conjunto de números enteros positivos N_s o un subconjunto de F números pares en el conjunto de números enteros positivos N_s , donde F es un número entero positivo menor que M .

Opcionalmente, el factor de expansión de codificación es igual a una potencia entera positiva de 2 menos 1, o a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de: 3, 5, 7, 11, 13, 17, 19, 23 o 29.

Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de 3 o 5.

Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de 5 o 7.

- 5 Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de 3, 5 o 7.

Opcionalmente, el factor de expansión de codificación incluye uno de: 7, 15, 31, 63, 127, 255, 511, 1023, 2047 o 4095.

- 10 Opcionalmente, el método incluye además, después de determinación del factor de expansión de codificación z en base a al menos uno del tamaño de bloque de código, el parámetro kb de la matriz de verificación básica, el valor entero positivo p o la matriz de verificación básica: almacenar al menos la matriz de verificación básica.

- 15 Opcionalmente, el método incluye además, antes de codificar la secuencia de datos que se va a codificar: agregar k' bits ficticios a la secuencia de datos que se va a codificar, para obtener la secuencia de datos que se va a codificar tenga una longitud de $kb \cdot z$ bits, donde una relación de k' a $kb \cdot z$ es menor o igual a $1/4$, k' es un número entero mayor o igual que 0, kb es un número de columnas del sistema de la matriz de verificación básica y es un número entero mayor que 1, y z es el factor de expansión que se utilizará en la codificación y es un número entero mayor que 0.

Opcionalmente, el factor de expansión z es n veces un número entero positivo pl .

Opcionalmente, el número entero positivo pl es un elemento de un subconjunto P_{set} , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{máx}$, donde n es un número natural y $P_{máx}$ es un número entero mayor o igual que 4.

- 20 Opcionalmente, todos los valores del número entero positivo n constituyen un subconjunto $nSet$, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

- 25 Opcionalmente, el número entero positivo pl es un elemento de un subconjunto P_{set} , que es un subconjunto compuesto por todos los valores del número entero positivo n , $K_{máx} \leq kb \times pl_{máx} \times n_{máx} \leq 1.2 \times K_{máx}$, donde $pl_{máx}$ es el valor más grande del subconjunto P_{set} , $n_{máx}$ es el valor más grande del subconjunto $nSet$ y $K_{máx}$ es un número entero mayor que 1024.

Opcionalmente, $K_{máx}$ es igual a 2000, 2048, 4000, 4096, 6000, 6144, 8000, 8192, 12000 o 12288.

- 30 Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = 0$, donde $i=0, 1, 2, \dots, mb-1$, $j=0, 1, \dots, nb-1$ y $P_{máx}$ es un número entero mayor o igual que 4.

Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = A$, donde i es un índice de fila de la matriz de verificación básica, j es un índice de columna de la matriz de verificación básica, hb_{ij} indica una matriz cuadrada que no es de ceros, A es un número entero fijo menor que pl y mayor o igual que 0, y $P_{máx}$ es un número entero mayor o igual que 4.

- 35 Opcionalmente, el parámetro kb de la matriz de verificación básica es un número de columnas del sistema de la matriz de verificación básica.

- 40 De acuerdo con otra realización de la presente divulgación, se proporciona un aparato para el procesamiento de datos con códigos de verificación de paridad de baja densidad (LDPC) estructurados. El aparato incluye: un módulo de obtención configurado para obtener un tamaño de bloque de código para la codificación de LDPC estructurada; un módulo de determinación configurado para determinar un factor de expansión de codificación z en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica que tiene mb filas y nb columnas; y un módulo de procesamiento configurado para codificar una secuencia de datos que se va a codificar, o decodificar una secuencia de datos que se va a decodificar, basado en la matriz de verificación básica y el factor de expansión de codificación, donde $kb = nb - mb$ y cada uno de kb , p , mb , z y nb es un número entero mayor que 1.

- 45 Opcionalmente, el módulo de determinación se configura para: determinar el factor de expansión de codificación de acuerdo con $\lceil CBS/(kb \times p) \rceil \times p$, donde CBS es el tamaño de bloque de código y $\lceil \rceil$ denota una operación de techo.

- 50 Opcionalmente, el módulo de obtención se configura de tal manera que el tamaño de bloque de código sea igual a un valor obtenido al multiplicar el parámetro kb de la matriz de verificación básica por uno de un conjunto de números naturales ordenados en orden ascendente.

Opcionalmente, el conjunto de números naturales ordenados en orden ascendente incluye una pluralidad de subconjuntos, cada uno de los cuales tiene una primera diferencia igual entre los elementos vecinos.

Opcionalmente, la primera diferencia para el i -ésimo subconjunto en el conjunto de números naturales ordenados en orden ascendente es menor que la primera diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo.

5 Opcionalmente, el módulo de determinación se configura de tal manera que: el tamaño de bloque de código es uno de un conjunto de tamaños de bloque de código, el conjunto de tamaños de bloque de código es un primer conjunto de números naturales almacenados en orden ascendente; y el factor de expansión de codificación es uno de un conjunto de factores de expansión de codificación, el conjunto de factores de expansión de codificación es un segundo conjunto de números naturales almacenados en orden ascendente.

10 Opcionalmente, el conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times kb$ y el último de los elementos sucesivos es menor o igual que $z(j+1) \times kb$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo, $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación, j es un número entero positivo y kb es el parámetro de la matriz de verificación básica.

15 Opcionalmente, el factor de expansión de codificación es $z(j+1)$ cuando el tamaño de bloque de código obtenido es uno de los elementos sucesivos.

Opcionalmente, el factor de expansión de codificación se determina como $z(j+1)$ cuando una relación entre el tamaño de bloque de código y el parámetro kb de la matriz de verificación básica es mayor que $z(j)$ y menor o igual que $z(j+1)$, donde $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación.

20 Opcionalmente, el conjunto de factores de expansión de codificación incluye una pluralidad de subconjuntos, cada uno tiene una segunda diferencia igual entre los elementos vecinos, y la segunda diferencia para el h -ésimo subconjunto de la pluralidad de subconjuntos es menor que la segunda diferencia para el $(h+1)$ -ésimo subconjunto, donde h es un número entero positivo.

Opcionalmente, cada uno de los elementos en el conjunto de factores de expansión de codificación que es mayor que el valor entero positivo p es n veces el valor entero positivo p , donde n es un número entero positivo.

25 Opcionalmente, el conjunto de tamaños de bloques de código incluye al menos los productos del parámetro kb de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación, y números enteros espaciados por un número entero positivo B entre todos los productos.

30 Opcionalmente, el aparato incluye además: un módulo de relleno de primeros bits configurado para agregar bits de relleno a una primera secuencia de datos al dividir la primera secuencia de datos en una pluralidad de subsecuencias de datos y agregar bits de relleno a cada una de las pluralidades de subsecuencias de datos, de tal manera que todas las subsecuencias de datos con los bits de relleno constituyen la secuencia de datos que se va a codificar.

35 Opcionalmente, el aparato incluye además: un segundo módulo de relleno de bits configurado para agregar bits de relleno a una segunda secuencia de datos al dividir la segunda secuencia de datos en kb subsecuencias de datos, cada una de las cuales incluye z o menos bits, y agregar bits de relleno a cada una de las subsecuencias de datos de tal manera que cada una de las subsecuencias de datos incluya z bits y todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar, donde kb es el parámetro de la matriz de verificación básica y z es el factor de expansión de codificación.

40 Opcionalmente, el aparato incluye además: un módulo de adición configurado para agregar una secuencia de Verificación de Redundancia Cíclica (CRC) de bits L a una tercera secuencia de datos para obtener la primera o segunda secuencia de datos, donde L es un número entero mayor o igual que 0.

Opcionalmente, el parámetro kb de la matriz de verificación básica es una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica, donde kb es un número entero mayor o igual que 4 y menor o igual que 64.

Opcionalmente, el valor entero positivo p es un grado de paralelismo de la decodificación de LDPC.

45 Opcionalmente, el valor entero positivo p es un número entero positivo fijo, y el grado p de paralelismo de la decodificación de LDPC incluye: el valor entero positivo p es una potencia entera positiva de 2 o un producto de una potencia entera positiva de 2 y un número primo.

50 Opcionalmente, el valor entero positivo p es un elemento de un subconjunto P , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{\max}$, donde $P_{\max}$ es un número entero mayor que 3 y menor o igual que 1024.

Opcionalmente, todos los valores del valor entero positivo p constituyen un subconjunto $nSet$, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

- Opcionalmente, el subconjunto nSet incluye uno de: un subconjunto de los primeros F elementos del conjunto de números enteros positivos Ns, un subconjunto de los últimos F elementos del conjunto de números enteros positivos Ns, un subconjunto de F números primos en el conjunto de números enteros positivos Ns, un subconjunto de F números impares en el conjunto de números enteros positivos Ns o un subconjunto de F números pares en el conjunto de números enteros positivos Ns, donde F es un número entero positivo menor que M.
- Opcionalmente, el factor de expansión de codificación es igual a una potencia entera positiva de 2 menos 1, o a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de: 3, 5, 7, 11, 13, 17, 19, 23 o 29.
- Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de 3 o 5.
- Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de 5 o 7.
- Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo. El número primo incluye uno de 3, 5 o 7.
- Opcionalmente, el factor de expansión de codificación incluye uno de: 7, 15, 31, 63, 127, 255, 511, 1023, 2047 o 4095.
- Opcionalmente, el aparato incluye además: un módulo de almacenamiento configurado para almacenar al menos la matriz de verificación básica.
- Opcionalmente, el aparato se configura aún más para, antes de codificar la secuencia de datos que se va a codificar: agregar k' bits ficticios a la secuencia de datos que se va a codificar, para obtener la secuencia de datos que se va a codificar con una longitud de kb*z bits, donde una relación de k' a kb*z es menor o igual a 1/4, k' es un número entero mayor o igual que 0, kb es un número de columnas del sistema de la matriz de verificación básica y es un número entero mayor que 1, y z es el factor de expansión que se utilizará en la codificación y es un número entero mayor que 0.
- Opcionalmente, el factor de expansión z es n veces un número entero positivo pl.
- Opcionalmente, el número entero positivo pl es un elemento de un subconjunto Pset, que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de Pmáx, donde n es un número natural y Pmáx es un número entero mayor o igual que 4.
- Opcionalmente, todos los valores del número entero positivo n constituyen un subconjunto nSet, que es un subconjunto de un conjunto de números enteros positivos, Ns, y tiene una longitud de F, donde el conjunto de números enteros positivos Ns = {1, 2, 3, ..., M}, donde M es un número entero mayor que 1.
- Opcionalmente, el número entero positivo pl es un elemento de un subconjunto Pset, que es un subconjunto compuesto por todos los valores del número entero positivo n, $K_{máx} \leq kb \times pl_{máx} \times n_{máx} \leq 1.2 \times K_{máx}$, donde plmáx es el valor más grande del subconjunto Pset, nmáx es el valor más grande del subconjunto nSet y Kmáx es un número entero mayor que 1024.
- Opcionalmente, Kmáx es igual a 2000, 2048, 4000, 4096, 6000, 6144, 8000, 8192, 12000 o 12288.
- Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = 0$, donde i=0, 1, 2..., mb-1, j=0, 1, ..., nb-1, y Pmáx es un número entero mayor o igual que 4.
- Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = A$, donde i es un índice de fila de la matriz de verificación básica, j es un índice de columna de la matriz de verificación básica, hb_{ij} indica una matriz cuadrada que no es de ceros, A es un número entero fijo menor que pl y mayor o igual que 0, y Pmáx es un número entero mayor o igual que 4.
- Opcionalmente, el parámetro kb de la matriz de verificación básica es un número de columnas del sistema de la matriz de verificación básica.
- Con la presente divulgación, se obtiene un tamaño de bloque de código para la codificación de LDPC estructurada. Un factor de expansión de codificación z se determina en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica que tiene mb filas y nb columnas. Se codifica una secuencia de datos que se va a codificar, o se decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación.
- En el presente documento, $kb = nb - mb$ y cada uno de kb, p, mb, z y nb es un número entero mayor que 1. Se puede ver que, con la solución anterior, el factor de expansión de codificación se determina en base a al menos uno del tamaño de bloque de código, el parámetro kb de la matriz de verificación básica, el valor entero positivo p o la matriz de verificación básica que tiene mb filas y nb columnas. La codificación y decodificación de LDPC se realiza en base

a la matriz de verificación básica y el factor de expansión de codificación. De esta manera, se puede mejorar la flexibilidad en el procesamiento de datos con codificación de LDPC, resolviendo de esta manera el problema en la técnica relacionada asociada con la baja flexibilidad en el procesamiento de datos con codificación de LDPC.

Breve descripción de los dibujos

5 La presente divulgación se puede entender con más detalle con referencia a las figuras que se describen a continuación, que constituyen una parte de la presente divulgación. Las realizaciones ilustrativas de la presente divulgación y las descripciones de la misma se proporcionan para explicar, en lugar de limitar, la presente divulgación. En las figuras:

10 La FIG. 1 es un diagrama de bloques que muestra una estructura de hardware de un terminal móvil en el que se puede aplicar un método para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación.

La FIG. 2 es un diagrama de flujo que ilustra un método para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación.

15 La FIG. 3 es un primer diagrama de bloques de un aparato para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación.

La FIG. 4 es un segundo diagrama de bloques de un aparato para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación.

La FIG. 5 es un tercer diagrama de bloques de un aparato para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación.

20 La FIG. 6 es un diagrama esquemático que muestra una matriz de verificación básica de acuerdo con una realización opcional de la presente divulgación.

La FIG. 7 es un primer diagrama esquemático que muestra una matriz de verificación básica H_b' de acuerdo con una realización opcional de la presente divulgación.

25 La FIG. 8 es un segundo diagrama esquemático que muestra una matriz de verificación básica H_b' de acuerdo con una realización opcional de la presente divulgación.

La FIG. 9 es un diagrama esquemático que muestra una comparación de rendimiento entre diferentes esquemas de relleno en un canal de ruido blanco aditivo de acuerdo con una realización opcional de la presente divulgación.

La FIG. 10 es un diagrama esquemático que muestra un almacenamiento en un decodificador de LDPC de acuerdo con una realización opcional de la presente divulgación.

30 Descripción detallada de las realizaciones

A continuación, la presente divulgación se describirá en detalle con referencia a las figuras, tomadas en conjunto con las realizaciones. Las realizaciones, y las características de las mismas, se pueden combinar entre sí, siempre que no entren en conflicto.

35 Cabe señalar que los términos tales como “primero”, “segundo y así sucesivamente, en la descripción, las reivindicaciones y las figuras se utilizan para distinguir entre objetos similares y no implican necesariamente ningún orden o secuencia en particular.

Realización 1

40 El método de acuerdo con la Realización 1 de la presente divulgación se puede realizar en un terminal móvil, un terminal de ordenador o un dispositivo informático similar. Cuando el método se realiza en un terminal móvil, por ejemplo, la FIG. 1 es un diagrama de bloques que muestra una estructura de hardware de un terminal móvil en el que se puede aplicar un método para el procesamiento de datos con códigos de LDPC de acuerdo con una realización de la presente divulgación. Como se muestra en la FIG. 1, el terminal móvil 10 puede incluir: uno o más procesadores 102 (solo se muestra uno, que incluye, pero no se limita a, un dispositivo de procesamiento tal como un microprocesador o una Unidad de Control Micro (MCU) o un dispositivo lógico programable como Matriz de Puertas Programables en Campo (FPGA)), una memoria 104 para almacenar datos, y un dispositivo de transmisión 106 para proporcionar funciones de comunicación. Los expertos en la técnica pueden apreciar que la estructura que se muestra en la FIG. 1 es solo ilustrativa, y la estructura del dispositivo electrónico anterior no se limita a esta. Por ejemplo, el terminal móvil 10 puede incluir más o menos componentes que los que se muestran en la FIG. 1, o tener una configuración diferente de la que se muestra en la FIG. 1.

50 La memoria 104 puede almacenar programas de software y módulos de aplicaciones de software, por ejemplo, instrucciones/módulos de programas asociados con el método para la recuperación de la fase portadora de acuerdo

- con una realización de la presente divulgación. El procesador 102 realiza diversas aplicaciones funcionales y operaciones de procesamiento de datos, es decir, realiza el método anterior, al ejecutar los programas de software y módulos almacenados en la memoria 104. La memoria 104 puede incluir una memoria caché aleatoria o una memoria no volátil, tal como uno o más dispositivos de almacenamiento magnético, memorias flash u otras memorias de estado sólido no volátiles. En algunos ejemplos, la memoria 104 puede incluir además una o más memorias proporcionadas a distancia desde el procesador 102, que se pueden conectar al terminal móvil 10 a través de una red. Ejemplos de dicha red incluyen, pero no se limitan a, Internet, una intranet de una empresa, una Red de Área Local (LAN), una red de comunicación móvil o cualquier combinación de las mismas.
- El dispositivo de transmisión 106 puede transmitir o recibir datos a través de una red. La red puede ser, por ejemplo, una red inalámbrica proporcionada por un proveedor de comunicación del terminal móvil 10. En un ejemplo, el dispositivo de transmisión 106 incluye un Controlador de Interfaz de Red (NIC), que se puede conectar a otros dispositivos de red a través de una estación base para la comunicación con Internet. En un ejemplo, el dispositivo de transmisión 106 puede ser un módulo de Radiofrecuencia (RF) para comunicarse con Internet de forma inalámbrica.
- En esta realización, se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. La FIG. 2 es un diagrama de flujo que ilustra un método para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación. Como se muestra en la FIG. 2, el método incluye las siguientes etapas.
- En la etapa S202, se obtiene un tamaño de bloque de código para la codificación de LDPC estructurada.
- En la etapa S204, se determina un factor de expansión de codificación z en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica que tiene mb filas y nb columnas.
- En la etapa S206, se codifica una secuencia de datos que se va a codificar, o se decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación.
- En el presente documento, $kb=nb-mb$ y cada uno de kb , p , mb , z y nb es un número entero mayor que 1.
- Opcionalmente, el método anterior para el procesamiento de datos con códigos de LDPC estructurados se puede aplicar, pero no se limita a, escenarios de procesamiento de secuencias de datos, por ejemplo, un escenario en el que una secuencia de datos se procesa con codificación o decodificación de LDPC.
- Opcionalmente, el método anterior para el procesamiento de datos con códigos de LDPC estructurados se puede aplicar, pero no se limita a, un codificador de LDPC o un decodificador de LDPC, por ejemplo, un transmisor con codificación de LDPC o un receptor con decodificación de LDPC.
- Con las etapas anteriores, se obtiene un tamaño de bloque de código para la codificación de LDPC estructurada. Un factor de expansión de codificación z se determina en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica o un valor entero positivo p . Se codifica una secuencia de datos que se va a codificar, o se decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación. Se puede ver que, con la solución anterior, el factor de expansión de codificación se determina en base a al menos uno del tamaño del bloque de código, el parámetro kb de la matriz de verificación básica o el valor entero positivo p . La codificación y decodificación de LDPC se realiza en base a la matriz de verificación básica y el factor de expansión de codificación. De esta manera, se puede mejorar la flexibilidad en el procesamiento de datos con codificación de LDPC, resolviendo de esta manera el problema en la técnica relacionada asociada con la baja flexibilidad en el procesamiento de datos con codificación de LDPC. Además, la codificación y decodificación de LDPC se puede aplicar de manera efectiva a diferentes tamaños de bloques de código, de tal manera que se pueda reducir la complejidad de la codificación de LDPC.
- Opcionalmente, como un ejemplo no limitante, el factor de expansión de codificación se puede determinar de acuerdo con: $\lceil CBS/(kb \times p) \rceil \times p$, donde CBS es el tamaño de bloque de código y $\lceil \rceil$ denota una operación de techo.
- Opcionalmente, en la etapa anterior S202, el tamaño de bloque de código es igual a un valor obtenido al multiplicar el parámetro kb de la matriz de verificación básica con uno de un conjunto de números naturales ordenados en orden ascendente.
- Opcionalmente, como un ejemplo no limitante, el conjunto de números naturales ordenados en orden ascendente puede incluir una pluralidad de subconjuntos, cada uno tiene una primera diferencia igual entre los elementos vecinos.
- Opcionalmente, la primera diferencia para el i -ésimo subconjunto en el conjunto de números naturales ordenados en orden ascendente es menor que la primera diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo.
- Opcionalmente, como un ejemplo no limitante, en la etapa anterior S206, la operación de determinación del factor de expansión de codificación en base al tamaño de bloque de código puede incluir: el tamaño de bloque de código que

es uno de un conjunto de tamaños de bloque de código, el conjunto de tamaños de bloque de código que es un primer conjunto de números naturales almacenados en orden ascendente; y el factor de expansión de codificación que es uno de un conjunto de factores de expansión de codificación, el conjunto de factores de expansión de codificación es un segundo conjunto de números naturales almacenados en orden ascendente.

- 5 Opcionalmente, el conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times kb$ y el último de los elementos sucesivos es menor o igual que $z(j+1) \times kb$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo, $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación, j es un número entero positivo y kb es el parámetro de la matriz de verificación básica.

- 10 Opcionalmente, como un ejemplo no limitante, el factor de expansión de codificación puede ser $z(j+1)$ cuando el tamaño de bloque de código obtenido es uno de los elementos sucesivos.

Opcionalmente, el factor de expansión de codificación se determina como $z(j+1)$ cuando una relación entre el tamaño de bloque de código y el parámetro kb de la matriz de verificación básica es mayor que $z(j)$ y menor o igual que $z(j+1)$, donde $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación.

- 15 Opcionalmente, como un ejemplo no limitante, el conjunto de factores de expansión de codificación puede incluir una pluralidad de subconjuntos, cada uno tiene una segunda diferencia igual entre los elementos vecinos, y la segunda diferencia para el h -ésimo subconjunto de la pluralidad de subconjuntos es menor que la segunda diferencia para el subconjunto $(h+1)$ -ésimo, donde h es un número entero positivo.

- 20 Opcionalmente, cada uno de los elementos del conjunto de factores de expansión de codificación que es mayor que el valor entero positivo p es n veces el valor entero positivo p , donde n es un número entero positivo.

Opcionalmente, como un ejemplo no limitante, el conjunto de tamaños de bloques de código puede incluir al menos productos del parámetro kb de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación, y números enteros espaciados por un número entero positivo B entre todos los productos.

- 25 Opcionalmente, como un ejemplo no limitante, antes de la etapa anterior S206, se pueden agregar bits de relleno a una primera secuencia de datos al dividir la primera secuencia de datos en una pluralidad de subsecuencias de datos y agregar bits de relleno a cada una de la pluralidad de subsecuencias de datos, de tal manera que todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar.

- 30 Opcionalmente, como un ejemplo no limitante, antes de la etapa anterior S206, se pueden agregar bits de relleno a una segunda secuencia de datos al dividir la segunda secuencia de datos en kb subsecuencias de datos, cada una de las cuales incluye z o menos bits, y agregar bits de relleno a cada una de las subsecuencias de datos de tal manera que cada una de las subsecuencias de datos incluya z bits y todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar, donde kb es el parámetro de la matriz de verificación básica y z es el factor de expansión de codificación.

- 35 Opcionalmente, como un ejemplo no limitante, antes de agregar los bits de relleno a la primera o segunda secuencia de datos, se puede agregar una secuencia de Verificación de Redundancia Cíclica (CRC) de bits L a una tercera secuencia de datos para obtener la primera o segunda secuencia de datos, donde L es un número entero mayor o igual que 0.

- 40 Opcionalmente, el parámetro kb de la matriz de verificación básica es una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica, donde kb es un número entero mayor o igual que 4 y menor o igual que 64.

Opcionalmente, el valor entero positivo p es un grado de paralelismo de la decodificación de LDPC.

Opcionalmente, el valor entero positivo p es un número entero positivo fijo, y el grado p de paralelismo de la decodificación de LDPC incluye: el valor entero positivo p es una potencia entera positiva de 2 o un producto de una potencia entera positiva de 2 y un número primo.

- 45 Opcionalmente, el valor entero positivo p es un elemento de un subconjunto P , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{\text{máx}}$, donde $P_{\text{máx}}$ es un número entero mayor que 3 y menor o igual que 1024.

- 50 Opcionalmente, todos los valores del valor entero positivo p constituyen un subconjunto $n\text{Set}$, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

Opcionalmente, el subconjunto $n\text{Set}$ incluye uno de: un subconjunto de los primeros F elementos del conjunto de números enteros positivos N_s , un subconjunto de los últimos F elementos del conjunto de números enteros positivos N_s , un subconjunto de F números primos en el conjunto de números enteros positivos N_s , un subconjunto de F

números impares en el conjunto de números enteros positivos N_s o un subconjunto de F números pares en el conjunto de números enteros positivos N_s , donde F es un número entero positivo menor que M .

Opcionalmente, como un ejemplo no limitante, el factor de expansión de codificación puede ser igual a una potencia entera positiva de 2 menos 1, o un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de: 3, 5, 7, 11, 13, 17, 19, 23 o 29. Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de 3 o 5. Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de 5 o 7. Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de 3, 5 o 7.

Opcionalmente, el factor de expansión de codificación incluye uno de: 7, 15, 31, 63, 127, 255, 511, 1023, 2047 o 4095.

Opcionalmente, después de la etapa anterior S204, se puede almacenar la matriz de verificación básica.

Opcionalmente, antes de codificar la secuencia de datos que se va a codificar, se pueden agregar k' bits ficticios a la secuencia de datos que se va a codificar, para obtener que la secuencia de datos que se va a codificar tenga una longitud de $kb \cdot z$ bits, donde una relación de k' a $kb \cdot z$ es menor o igual a $1/4$, k' es un número entero mayor o igual que 0, kb es un número de columnas del sistema de la matriz de verificación básica y es un número entero mayor que 1, y z es el factor de expansión que se utilizará en la codificación y es un número entero mayor que 0. En la codificación de LDPC, la matriz de verificación básica de los códigos de LDPC tiene una cierta característica de distribución de grado, que se puede dañar si se agregan demasiados bits de relleno, lo que resulta en un rendimiento de decodificación degradado de los códigos de LDPC. Al limitar la longitud del relleno dentro de $1/4$, es ventajoso en el sentido de que las características fundamentales de la matriz de verificación básica de los códigos de LDPC y, por lo tanto, el rendimiento de decodificación de los códigos de LDPC, pueden seguir siendo sustancialmente las mismas. En el presente documento, el número y la posición de los bits de relleno son conocidos tanto por el transmisor como por el receptor. Los bits de relleno no se incluyen en la transmisión de datos. Durante la decodificación de los códigos de LDPC, estos bits conocidos no afectarán al resultado de la decodificación. Si todos los bits de una columna particular de la matriz de verificación básica son bits de relleno, esta columna se debe eliminar de la matriz de verificación básica y, en este caso, se cambiará la distribución de grados de la matriz de verificación básica.

Opcionalmente, el factor de expansión z es n veces un número entero positivo pl .

Opcionalmente, el número entero positivo pl es un elemento de un subconjunto P_{set} , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{máx}$, donde n es un número natural y $P_{máx}$ es un número entero mayor o igual que 4.

Opcionalmente, todos los valores del número entero positivo n constituyen un subconjunto $nSet$, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = 0$, donde $i=0, 1, 2, \dots, mb-1$, $j=0, 1, \dots, nb-1$ y $P_{máx}$ es un número entero mayor o igual que 4.

Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = A$, donde i es un índice de fila de la matriz de verificación básica, j es un índice de columna de la matriz de verificación básica, hb_{ij} indica una matriz cuadrada que no es de ceros, A es un número entero fijo menor que pl y mayor o igual que 0, y $P_{máx}$ es un número entero mayor o igual que 4.

Opcionalmente, el parámetro kb de la matriz de verificación básica es un número de columnas del sistema de la matriz de verificación básica.

Realización 2

De acuerdo con una realización de la presente divulgación, también se proporciona un aparato para el procesamiento de datos con códigos de LDPC estructurados. El aparato puede implementar las realizaciones anteriores y las realizaciones opcionales (los detalles de las mismas se omitirán en el presente documento). Como se utiliza a continuación, el término "módulo" puede ser software, hardware o una combinación de los mismos, capaz de realizar una función predeterminada. Mientras que los aparatos que se van a describir en las siguientes realizaciones se implementan preferiblemente en software, se puede contemplar que también se pueden implementar en hardware o en una combinación de software y hardware.

La FIG. 3 es un primer diagrama de bloques de un aparato para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación. Como se muestra en la FIG. 3, el aparato incluye:

1) un módulo de obtención 32 configurado para obtener un tamaño de bloque de código para la codificación de LDPC estructurada;

2) un módulo de determinación 34 acoplado al módulo 32 de obtención y configurado para determinar un factor de expansión de codificación z en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica o un valor entero positivo p ; y

3) un módulo de procesamiento 36 acoplado al módulo de determinación 34 y configurado para codificar una secuencia de datos que se va a codificar, o decodificar una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación,

donde $kb = nb - mb$ y cada uno de kb , p , mb , z y nb es un número entero mayor que 1.

Opcionalmente, el aparato anterior para el procesamiento de datos con códigos de LDPC estructurados se puede aplicar, pero no se limita a, escenarios de procesamiento de secuencias de datos, por ejemplo, un escenario en el que una secuencia de datos se procesa con codificación o decodificación de LDPC.

Opcionalmente, el aparato mencionado anteriormente para el procesamiento de datos con códigos de LDPC estructurados se puede aplicar, pero no se limita a, un codificador de LDPC o un decodificador de LDPC, por ejemplo, un transmisor con codificación de LDPC o un receptor con decodificación de LDPC.

Con el aparato anterior, un módulo de obtención obtiene un tamaño de bloque de código para la codificación de LDPC estructurada. Un módulo de determinación determina un factor de expansión de codificación z en base a al menos uno del tamaño del bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica que tiene mb filas y nb columnas. Un módulo de procesamiento codifica una secuencia de datos que se va a codificar, o decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación. En el presente documento, $kb = nb - mb$ y cada uno de kb , p , mb , z y nb es un número entero mayor que 1. Se puede ver que, con la solución anterior, el factor de expansión de codificación se determina en base a al menos uno del tamaño del bloque de código, el parámetro kb de la matriz de verificación básica o el valor entero positivo p . La codificación y decodificación de LDPC se realiza en base a la matriz de verificación básica y el factor de expansión de codificación. De esta manera, se puede mejorar la flexibilidad en el procesamiento de datos con codificación de LDPC, resolviendo de esta manera el problema en la técnica relacionada asociada con la baja flexibilidad en el procesamiento de datos con codificación de LDPC. Además, la codificación y decodificación de LDPC se puede aplicar de manera efectiva a diferentes tamaños de bloques de código, de tal manera que se pueda reducir la complejidad de la codificación de LDPC.

Opcionalmente, como un ejemplo no limitante, el módulo de determinación se puede configurar para: determinar el factor de expansión de codificación de acuerdo con $\lceil CBS / (kb \times p) \rceil \times p$, donde CBS es el tamaño de bloque de código y $\lceil \rceil$ denota una operación de techo.

Opcionalmente, como un ejemplo no limitante, el módulo de obtención se puede configurar de tal manera que el tamaño de bloque de código sea igual a un valor obtenido al multiplicar el parámetro kb de la matriz de verificación básica por uno de un conjunto de números naturales ordenados en orden ascendente.

Opcionalmente, como un ejemplo no limitante, el conjunto de números naturales ordenados en orden ascendente puede incluir una pluralidad de subconjuntos, cada uno tiene una primera diferencia igual entre los elementos vecinos.

Opcionalmente, la primera diferencia para el i -ésimo subconjunto en el conjunto de números naturales ordenados en orden ascendente es menor que la primera diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo.

Opcionalmente, el módulo de determinación se configura de tal manera que: el tamaño de bloque de código es uno de un conjunto de tamaños de bloque de código, el conjunto de tamaños de bloque de código es un primer conjunto de números naturales almacenados en orden ascendente; y el factor de expansión de codificación es uno de un conjunto de factores de expansión de codificación, el conjunto de factores de expansión de codificación es un segundo conjunto de números naturales almacenados en orden ascendente.

Opcionalmente, el conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times kb$ y el último de los elementos sucesivos es menor o igual que $z(j+1) \times kb$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo, $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación, j es un número entero positivo y kb es el parámetro de la matriz de verificación básica.

Opcionalmente, el factor de expansión de codificación es $z(j+1)$ cuando el tamaño de bloque de código obtenido es uno de los elementos sucesivos.

Opcionalmente, el factor de expansión de codificación se determina como $z(j + 1)$ cuando una relación entre el tamaño de bloque de código y el parámetro kb de la matriz de verificación básica es mayor que $z(j)$ y menor o igual que $z(j + 1)$, donde $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación.

- 5 Opcionalmente, el conjunto de factores de expansión de codificación incluye una pluralidad de subconjuntos, cada uno tiene una segunda diferencia igual entre los elementos vecinos, y la segunda diferencia para el h -ésimo subconjunto de la pluralidad de subconjuntos es menor que la segunda diferencia para el $(h + 1)$ -ésimo subconjunto, donde h es un número entero positivo.

Opcionalmente, cada uno de los elementos del conjunto de factores de expansión de codificación que es mayor que el valor entero positivo p es n veces el valor entero positivo p , donde n es un número entero positivo.

- 10 Opcionalmente, el conjunto de tamaños de bloques de código incluye al menos los productos del parámetro kb de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación, y números enteros espaciados por un número entero positivo B entre todos los productos.

- 15 La FIG. 4 es un segundo diagrama de bloques de un aparato para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación. Como se muestra en la FIG. 4, opcionalmente, el aparato incluye además: un primer módulo de relleno de bits 42 acoplado al módulo de procesamiento 36 y configurado para agregar bits de relleno a una primera secuencia de datos al dividir la primera secuencia de datos en una pluralidad de subsecuencias de datos y agregar bits de relleno a cada una de las pluralidades de subsecuencias de datos, de tal manera que todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar.

- 20 La FIG. 5 es un tercer diagrama de bloques de un aparato para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con una realización de la presente divulgación. Como se muestra en la FIG. 5, opcionalmente, el aparato incluye además: un segundo módulo de relleno de bits 52 acoplado al módulo de procesamiento 36 y configurado para agregar bits de relleno a una segunda secuencia de datos al dividir la segunda secuencia de datos en kb subsecuencias de datos, cada una de las cuales incluye z o menos bits, y agregar bits de relleno a cada una de las subsecuencias de datos de tal manera que cada una de las subsecuencias de datos incluya z bits y todas las subsecuencias de datos con los bits de relleno constituyen la secuencia de datos que se va a codificar, donde kb es el parámetro de la matriz de verificación básica y z es el factor de expansión de codificación.

- 25 Opcionalmente, el aparato incluye además: un módulo de adición acoplado al primer módulo de relleno de bits 42 o al segundo módulo de relleno de bits 52 y configurado para agregar una secuencia de Verificación de Redundancia Cíclica (CRC) de bits L a una tercera secuencia de datos para obtener la primera o segunda secuencia de datos, donde L es un número entero mayor o igual que 0.

Opcionalmente, el parámetro kb de la matriz de verificación básica es una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica, donde kb es un número entero mayor o igual que 4 y menor o igual que 64.

- 35 Opcionalmente, como un ejemplo no limitante, el valor entero positivo p puede ser un grado de paralelismo de la decodificación de LDPC.

Opcionalmente, el valor entero positivo p es un número entero positivo fijo, y el grado p de paralelismo de la decodificación de LDPC incluye: el valor entero positivo p es una potencia entera positiva de 2 o un producto de una potencia entera positiva de 2 y un número primo.

- 40 Opcionalmente, el valor entero positivo p es un elemento de un subconjunto P , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{\text{máx}}$, donde $P_{\text{máx}}$ es un número entero mayor que 3 y menor o igual que 1024.

- 45 Opcionalmente, todos los valores del valor entero positivo p constituyen un subconjunto $n\text{Set}$, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

- 50 Opcionalmente, el subconjunto $n\text{Set}$ incluye uno de: un subconjunto de los primeros F elementos del conjunto de números enteros positivos N_s , un subconjunto de los últimos F elementos del conjunto de números enteros positivos N_s , un subconjunto de F números primos en el conjunto de números enteros positivos N_s , un subconjunto de F números impares en el conjunto de números enteros positivos N_s o un subconjunto de F números pares en el conjunto de números enteros positivos N_s , donde F es un número entero positivo menor que M .

Cabe señalar en el presente documento que el primer caso en el que el valor entero positivo p es una potencia entera positiva de 2 y el segundo caso en el que el valor entero positivo p es un producto de una potencia entera positiva de 2 y un número primo son mutuamente excluyentes. Si el número primo es 2, el primer caso y el segundo caso son iguales, lo cual no es razonable. Por lo tanto, el caso en el que el número primo es 2 se considera el primer caso.

Además, en este contexto, el primer caso incluye una potencia de 1^2 , pero 1 no es un número primo. Por lo tanto, el número primo en el presente documento es al menos mayor o igual que 3.

Opcionalmente, el factor de expansión de codificación es igual a una potencia entera positiva de 2 menos 1, o un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de: 3, 5, 7, 11, 13, 17, 19, 23 o 29. Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de 3 o 5. Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de 5 o 7. Opcionalmente, el factor de expansión de codificación es igual a un producto de una potencia entera positiva de 2 y un número primo, y el número primo incluye uno de 3, 5 o 7.

Opcionalmente, el factor de expansión de codificación incluye uno de: 7, 15, 31, 63, 127, 255, 511, 1023, 2047 o 4095.

Opcionalmente, el aparato incluye además: un módulo de almacenamiento configurado para almacenar al menos la matriz de verificación básica.

Opcionalmente, el aparato se configura aún más para, antes de codificar la secuencia de datos que se va a codificar: agregar k' bits ficticios a la secuencia de datos que se va a codificar, para obtener la secuencia de datos que se va a codificar que tiene una longitud de $kb \cdot z$ bits, donde una relación de k' a $kb \cdot z$ es menor o igual a $1/4$, k' es un número entero mayor o igual que 0, kb es un número de columnas del sistema de la matriz de verificación básica y es un número entero mayor que 1, y z es el factor de expansión que se utilizará en la codificación y es un número entero mayor que 0.

Opcionalmente, el factor de expansión z es n veces un número entero positivo pl .

Opcionalmente, el número entero positivo pl es un elemento de un subconjunto P_{set} , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{máx}$, donde n es un número natural y $P_{máx}$ es un número entero mayor o igual que 4.

Opcionalmente, todos los valores del número entero positivo n constituyen un subconjunto n_{set} , que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = 0$, donde $i=0, 1, 2, \dots, mb-1$, $j=0, 1, \dots, nb-1$ y $P_{máx}$ es un número entero mayor o igual que 4.

Opcionalmente, cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{máx}/pl)) = A$, donde i es un índice de fila de la matriz de verificación básica, j es un índice de columna de la matriz de verificación básica, hb_{ij} indica una matriz cuadrada que no es de ceros, A es un número entero fijo menor que pl y mayor o igual que 0, y $P_{máx}$ es un número entero mayor o igual que 4.

Opcionalmente, el parámetro kb de la matriz de verificación básica es un número de columnas del sistema de la matriz de verificación básica.

Cabe señalar que cada uno de los módulos descritos anteriormente se puede implementar por medio de software o hardware, y este último se puede implementar, pero no se limita a, de la siguiente manera: los módulos mencionados anteriormente se pueden ubicar en el mismo procesador, o se pueden distribuir en una pluralidad de procesadores.

A continuación, se describirán en detalle las realizaciones opcionales de la presente divulgación.

En una realización opcional de la presente divulgación, se proporciona un método para la codificación/decodificación de LDPC. El método incluye las siguientes etapas.

En la etapa 101, se obtiene un tamaño de bloque de código para la codificación de LDPC estructurada.

En la etapa 102, se determina un factor de expansión de codificación z en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica.

En la etapa 103, se codifica una secuencia de datos que se va a codificar, o se decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación.

Opcionalmente, la operación de determinación del factor de expansión de codificación se basa al menos en el tamaño de bloque de código CBS, el parámetro kb de la matriz de verificación básica y el valor entero positivo p incluye: el factor de expansión de codificación es igual a $\lceil CBS / (kb \times p) \rceil \times p$, donde CBS es el tamaño de bloque de código. En un proceso de decodificación de LDPC en capas real, el valor entero positivo p corresponde a un grado de paralelismo de la decodificación de LDPC o a un múltiplo del mismo. En este caso, el factor de expansión de codificación depende del grado de paralelismo, lo cual es ventajoso en el sentido de que la implementación de hardware para la

decodificación de LDPC en capas puede ser sencilla, sin ningún desajuste entre el entrelazado y el desentrelazado en el decodificador.

Opcionalmente, el tamaño de bloque de código puede ser igual a un valor obtenido al multiplicar el parámetro k_b de la matriz de verificación básica por uno de un conjunto de números naturales ordenados en orden ascendente.

- 5 Opcionalmente, el parámetro k_b de la matriz de verificación básica es una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica, donde k_b es un número entero mayor que varía desde 4 hasta 64. El conjunto de números naturales ordenados en orden ascendente incluye una pluralidad de subconjuntos, cada uno tiene una diferencia igual dentro del subconjunto. Opcionalmente, en el conjunto de números naturales ordenados en orden ascendente, la diferencia para el i -ésimo subconjunto es menor que la diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo.

Opcionalmente, el tamaño de bloque de código es uno de un conjunto de tamaños de bloque de código. El conjunto de tamaños de bloques de código es un conjunto de números naturales almacenados en orden ascendente. El factor de expansión de codificación es uno de un conjunto de factores de expansión de codificación. El conjunto de factores de expansión de codificación es un conjunto de números naturales almacenados en orden ascendente.

- 15 Opcionalmente, el conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times k_b$ y el último de los elementos sucesivos es menor o igual que $z(j+1) \times k_b$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo y $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación. Cuando el tamaño de bloque de código es uno de los elementos sucesivos, el factor de expansión de codificación se limita a $z(j+1)$.

- Opcionalmente, el conjunto de factores de expansión de codificación incluye una pluralidad de subconjuntos, cada uno tiene una diferencia igual dentro del subconjunto. La diferencia para el h -ésimo subconjunto de la pluralidad de subconjuntos es menor que la diferencia para el $(h+1)$ -ésimo subconjunto, donde h es un número entero positivo. El conjunto de tamaños de bloques de código incluye al menos los productos del parámetro k_b de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación, y números enteros espaciados por un número entero positivo B entre todos los productos. Los productos del parámetro k_b de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación son números de bits máximos del sistema que se pueden admitir por códigos de LDPC para diferentes factores de expansión de codificación. También puede haber tamaños de bloques de código entre los respectivos números de bits del sistema, de tal manera que los tamaños de bloques de código pueden ser más flexibles para ser utilizados en varios sistemas de comunicación. Se puede resolver el problema asociado con la falta de flexibilidad en las longitudes de código y las tasas de código de los códigos de LDPC.

- Opcionalmente, el factor de expansión de codificación puede ser igual a una potencia entera positiva de 2 menos 1. Más particularmente, el factor de expansión de codificación puede incluir uno de: 7, 15, 31, 63, 127, 255, 511, 1023, 2047 o 4095. Esto es ventajoso en el sentido de que cada subconjunto de z bits en los códigos de LDPC incluye un número correspondiente de bits de CRC. Debido a la característica del código cíclico de CRC de que un código cíclico de CRC desplazado cíclicamente en un número arbitrario de bits sigue perteneciendo a un espacio de palabras de código cíclico de CRC (es decir, sigue siendo un código cíclico de CRC), cada subconjunto de z bits en los bits de control para los códigos de LDPC sigue siendo un código cíclico de CRC y, por lo tanto, se puede seguir utilizando para la verificación de CRC. Esto facilita la mejora del rendimiento de decodificación de los códigos de LDPC y la aceleración de la convergencia de decodificación en un proceso de decodificación iterativo, lo que puede reducir el tiempo de decodificación y mejorar la velocidad de decodificación (z es igual al valor del factor de expansión). Alternativamente, el factor de expansión de codificación puede ser igual a un producto de una potencia entera positiva de 2 y un número primo. Más particularmente, el número primo puede ser, por ejemplo, uno de 3, 5, 7, 11, 13, 17, 19, 23 o 29. Por definición, un número entero es un número primo si no tiene ningún factor entero distinto de 1 y él mismo. Por lo tanto, 1 no es un número primo. El uso de dicho factor de expansión de codificación puede facilitar la simplificación de las redes de entrelazado y desentrelazado de los códigos de LDPC. De acuerdo con el principio de los códigos de LDPC, cada elemento de la matriz de verificación básica corresponde a un número de bits desplazados cíclicamente. De acuerdo con algún principio, el uso de los valores anteriores puede simplificar las redes de entrelazado y desentrelazado, lo que a su vez reduce la complejidad de los códigos de LDPC (casi el 40 % de la complejidad de decodificación de los códigos de LDPC reside en la red de entrelazado y, una vez que se simplifica la red de entrelazado, se puede reducir la complejidad de decodificación de los códigos de LDPC).

- Opcionalmente, se pueden dar ejemplos de los valores del factor de expansión en el presente documento. Por ejemplo, el factor de expansión de codificación puede ser igual a un producto de una potencia entera positiva de 2 y un número primo y el número primo puede incluir uno de 3 o 5 y el número entero positivo puede incluir uno de 1, 2, 3, 4, 5, 6, 7 u 8. Es ventajoso utilizar el factor de expansión para reducir la complejidad del decodificador de LDPC y admitir un diseño flexible de la longitud del código. Un decodificador de LDPC normalmente incluye un módulo de almacenamiento, un módulo de red de entrelazado, un módulo de red de desentrelazado y un módulo de cálculo. El módulo de almacenamiento almacena información blanda total actualizada, que es la suma de toda la información extrínseca. En el decodificador, se debe asignar un bloque de almacenamiento para cada columna de la matriz de

verificación básica (por ejemplo, para una matriz de verificación básica que tiene m_b filas y n_b columnas, se necesitan n_b bloques de almacenamiento) y cada bloque de almacenamiento se divide en un número de bloques de direcciones, cada uno definido como una palabra. Por lo tanto, un bloque de almacenamiento puede almacenar una serie de palabras. Al diseñar la decodificación de LDPC, el grado de paralelismo de la decodificación normalmente es igual al tamaño de una palabra, que se refiere al número de bits blandos almacenados en una palabra. En este ejemplo, el número primo máximo es 5 y la potencia máxima de 2 es 2^8 . De acuerdo con lo anterior, en el decodificador de LDPC, cada bloque de almacenamiento tiene cinco palabras y cada palabra tiene un tamaño de $2^8 = 256$. Durante el proceso de decodificación de LDPC, los datos se leen primero desde el módulo de almacenamiento, luego se someten a procesamiento por el módulo de red de entrelazado, el módulo de cálculo y el módulo de red de desentrelazado, y finalmente se vuelven a escribir en el módulo de almacenamiento. Cuando el factor de expansión es igual a $5 \times 2^8 = 1280$, se pueden completar cinco palabras exactamente y cada palabra tiene 256 bits blandos de datos. En un caso con una longitud de código flexible (es decir, donde se obtiene una longitud de código diferente al cambiar el factor de expansión), por ejemplo, cuando el factor de expansión es $5 \times 2^4 = 80$, todavía hay cinco palabras, y cada palabra almacena solo $2^4 = 16$ bits de datos, lo que ocurre cada 16 bits en cada palabra. En este caso, cuando los datos de una palabra se leen para actualizarlos, sólo hay 16 nodos de verificación efectivos que se deben actualizar en la unidad de cálculo del decodificador, ya que sólo hay disponibles 16 bits de datos efectivos. Cuando el factor de expansión es igual a un múltiplo (una potencia de 2) del número primo 3, solo se requieren 3 palabras para la operación. En el decodificador, cuando el factor de expansión es igual a un producto de una potencia entera positiva (1, 2, 3, 4, 5, 6, 7 u 8) de 2 y un número primo, el número primo puede ser igual a 3 o 5 y el hardware anterior se puede reutilizar, es decir, es posible un diseño de longitud de código flexible.

Además, el número primo puede incluir 5 o 7 y la potencia de 2 puede ser 2 a la potencia de 1, 2, 3, 4, 5, 6, 7 u 8. En este caso, el tamaño de la palabra también puede ser de 256 y se necesitan 7 palabras. En el presente documento, se selecciona un número primo más grande 7 de tal manera que el factor de expansión pueda ser mayor para admitir una longitud de código mayor. Opcionalmente, el número primo puede incluir 3, 5 o 7 y la potencia de 2 puede ser 2 a la potencia de 1, 2, 3, 4, 5, 6, 7 u 8. En este caso, el tamaño de la palabra también puede ser 256 y se necesitan 7 palabras ya que el número primo máximo es 7. Cuando el factor de expansión es una potencia de 3, se seleccionan 3 palabras para la operación. Cuando el factor de expansión es una potencia de 5, se seleccionan 5 palabras para la operación. Cuando el factor de expansión es una potencia de 7, se seleccionan 7 palabras para la operación. En el presente documento, un factor de expansión mayor puede garantizar que se puedan admitir los códigos de LDPC que tienen una longitud de código mayor, y un factor de expansión más pequeño puede garantizar que se puedan admitir los códigos de LDPC que tienen una longitud de código más pequeña, de tal manera que se pueda lograr un rango de valores razonable y se pueda admitir una amplia gama de longitudes de código de LDPC. En una realización opcional de la presente divulgación, se proporciona un aparato para la codificación/decodificación de LDPC. El aparato incluye:

un módulo de obtención 201 configurado para obtener un tamaño de bloque de código para la codificación de LDPC estructurada;

un módulo de determinación 202 configurado para determinar un factor de expansión de codificación z en base a al menos uno del tamaño de bloque de código, un parámetro k_b de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica;

un módulo de procesamiento 203 configurado para codificar una secuencia de datos que se va a codificar, o decodificar una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación; y

un módulo de almacenamiento 204 configurado para almacenar al menos la matriz de verificación básica.

En el presente documento, en el módulo de obtención, el tamaño de bloque de código se puede limitar a un valor obtenido al multiplicar el parámetro k_b de la matriz de verificación básica por uno de un conjunto de números naturales ordenados en orden ascendente. El parámetro k_b de la matriz de verificación básica puede ser una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica. El parámetro k_b de la matriz de verificación básica puede ser un número entero que varía desde 4 hasta 64. El conjunto de números naturales ordenados en orden ascendente puede incluir una pluralidad de subconjuntos, cada uno tiene una diferencia igual dentro del subconjunto. La diferencia para el i -ésimo subconjunto en el conjunto de números naturales ordenados en orden ascendente puede ser menor que la diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo.

Opcionalmente, en el módulo de determinación, la operación de determinación del factor de expansión de codificación se basa al menos en el tamaño de bloque de código y el valor entero positivo p puede incluir: determinar el factor de expansión de codificación de acuerdo con: $\lceil CBS / (k_b \times p) \rceil \times p$, donde CBS es el tamaño de bloque de código, k_b es una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica, y p es el número entero positivo.

Opcionalmente, el tamaño de bloque de código puede ser uno de un conjunto de tamaños de bloque de código. El factor de expansión de codificación es uno de un conjunto de factores de expansión de codificación. El conjunto de tamaños de bloques de código es un conjunto de números naturales almacenados en orden ascendente. El conjunto

de factores de expansión de codificación es un conjunto de números naturales almacenados en orden ascendente. El número de dimensiones del conjunto de tamaños de bloque de código es mayor o igual que el del conjunto de factores de expansión de codificación. Opcionalmente, el conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times kb$ y el último de los elementos sucesivos es menor o igual que $z(j+1) \times kb$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo, $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación. El factor de expansión de codificación se puede limitar a $z(j+1)$ cuando el tamaño de bloque de código obtenido es uno de los elementos sucesivos. El conjunto de factores de expansión de codificación puede incluir una pluralidad de subconjuntos, cada uno tiene una diferencia igual dentro del subconjunto. La diferencia para el i -ésimo subconjunto de la pluralidad de subconjuntos es menor que la diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo.

Opcionalmente, el conjunto de tamaños de bloques de código puede incluir al menos productos del parámetro kb de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación, y números enteros espaciados por un número entero positivo B entre todos los productos.

Opcionalmente, el factor de expansión de codificación puede ser igual a una potencia entera positiva de 2 menos 1, o un producto de una potencia entera positiva de 2 y un número primo. En particular, el factor de expansión de codificación puede incluir uno de: 7, 15, 31, 63, 127, 255, 511, 1023, 2047 o 4095.

Realización 3

De acuerdo con una realización de la presente divulgación, también se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. El método puede implementar el método anterior para la codificación/decodificación de LDPC (los detalles del mismo se omitirán en el presente documento). Como se utiliza más adelante, el término "método" o "módulo" puede ser software, hardware o una combinación de los mismos, capaz de realizar una función predeterminada. Mientras que los aparatos que se van a describir en las siguientes realizaciones se implementan preferiblemente en software, se puede contemplar que también se pueden implementar en hardware o en una combinación de software y hardware.

De acuerdo con una realización de la presente divulgación, se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. El método se aplica en un transmisor e incluye las siguientes etapas.

1. Segmentación de Bloques de Código: Un bloque de transporte que tiene una longitud de CL se segmenta en una pluralidad de bloques de código.

2. Un tamaño de bloque de código para la codificación de LDPC estructurada se obtiene en base a un tamaño de bloque de código después de la segmentación del bloque de código.

3. Un factor de expansión de codificación z se determina en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica.

4. Una secuencia de datos que se va a codificar se codifica en base a la matriz de verificación básica y el factor de expansión de codificación.

Opcionalmente, en la etapa 2, el factor de expansión de codificación se determina en base al tamaño de bloque de código. En el presente documento, la operación de determinación del factor de expansión de codificación en base al tamaño de bloque de código puede incluir: el tamaño de bloque de código que es uno de un conjunto de tamaños de bloque de código, el conjunto de tamaños de bloque de código que es un primer conjunto de números naturales almacenados en orden ascendente; y el factor de expansión de codificación que es uno de un conjunto de factores de expansión de codificación, el conjunto de factores de expansión de codificación es un segundo conjunto de números naturales almacenados en orden ascendente.

Opcionalmente, cada uno de los elementos en el conjunto de factores de expansión de codificación que es mayor que el valor entero positivo p es n veces el valor entero positivo p , donde n es un número entero positivo.

Opcionalmente, el valor entero positivo p es un elemento de un subconjunto P_{set} , que a su vez es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{máx}$, donde $P_{máx}$ es un número entero mayor que 3 y menor o igual que 1024. En el presente documento, $P_{máx}$ puede ser un grado máximo de paralelismo de decodificación que se puede admitir por un decodificador de LDPC en un receptor de datos. El grado de paralelismo de la decodificación se refiere al número de actualizaciones de código de verificación de paridad que se pueden realizar simultáneamente en un decodificador de LDPC. $P_{máx}$ puede ser un grado máximo de paralelismo que se puede admitir por el decodificador de LDPC. Normalmente, cuando el grado máximo de paralelismo de decodificación que puede admitir el decodificador de LDPC es $P_{máx}$, el decodificador de LDPC puede admitir otros grados de paralelismo más pequeños que $P_{máx}$, tal vez con algunos circuitos de control adicionales.

Además, opcionalmente, todos los valores del valor entero positivo p constituyen un subconjunto $nSet$, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números

enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1. El subconjunto $nSet$ puede incluir uno de: un subconjunto de los primeros F elementos del conjunto de números enteros positivos N_s , un subconjunto de los últimos F elementos del conjunto de números enteros positivos N_s , un subconjunto de F números primos en el conjunto de números enteros positivos N_s , un subconjunto de F números impares en el conjunto de números enteros positivos N_s o un subconjunto de F números pares en el conjunto de números enteros positivos N_s , donde F es un número entero positivo menor que M .

En un ejemplo de codificación de LDPC en un transmisor, el tamaño de un bloque de transporte que se va a transmitir es de 4000 bits y una matriz de verificación básica para los códigos de LDPC es la siguiente, que corresponde a un conjunto de factores de expansión de codificación: [2 4 6 8 10 12 16 20 24 32 40 48 64 80 96 128 160 192 256 320 384 512 640 768 1024 1280]. El conjunto de factores de expansión incluye productos de 5 y 2 a la potencia de [1 2 3 4 5 6 7 8], es decir, [10 20 40 80 160 320 640 1280]; productos de 3 y 2 a la potencia de [1 2 3 4 5 6 7 8], es decir, [6 12 24 48 96 192 384 768]; productos de 2 y 2 a la potencia de [1 2 3 4 5 6 7 8], es decir, [4 8 16 32 64 128 256 512]; y productos de 1 y 2 a la potencia de [1 2 3 4 5 6 7 8], es decir, [2 4 8 16 32 64 128 256]. Por lo tanto, el subconjunto nSet incluye valores de [1 2 3 5] y el subconjunto Pset incluye valores de [2 4 8 16 32 64 128 256]. Cada uno de los elementos en el conjunto de factores de expansión de codificación es n veces el valor entero positivo p , donde n es un elemento del subconjunto nSet y p es un elemento del subconjunto Pset. La matriz de verificación básica también se puede denominar como matriz de verificación básica o similar. El factor de expansión también se puede denominar como tamaño de elevación, tamaño de la submatriz o similar. El parámetro k_b de la matriz de verificación básica es el número de columnas del sistema de la matriz de verificación básica. Cuando el número total de columnas de la matriz de verificación básica es n_b y el número total de filas de la matriz de verificación básica es m_b , $k_b = n_b - m_b$. En este ejemplo, el número de columnas del sistema de la matriz de verificación básica $k_b = 8$, el número de m_b filas = 18, el número total de n_b columnas = 26, que satisfacen $k_b = n_b - m_b$. La matriz de verificación básica para los códigos de LDPC es igual a:

[illegible]

El conjunto de tamaños de bloques de código incluye al menos productos de $kb=8$ y los respectivos del conjunto de factores de expansión de codificación, es decir, [16 32 48 64 80 96 128 160 192 256 320 384 512 640 768 1024 1280 1536 2048 2560 3072 4096 5120 6144 8192 10240].

Este ejemplo incluye las siguientes etapas.

1. Segmentación de Bloques de Código: el bloque de transporte que tiene una longitud de $CL=4000$ se segmenta en una pluralidad de bloques de código. Dado que la longitud del bloque de transporte, 4000 bits, es menor que el valor máximo (10240) en el conjunto anterior de tamaños de bloques de código, solo se obtiene un bloque de código que tiene una longitud de 4000 bits en la segmentación de bloques de código.

2. En base al tamaño de bloque de código (4000) después de la segmentación del bloque de código, se obtiene el tamaño de bloque de código (4000) para la codificación de LDPC estructurada.

3. Un factor de expansión de codificación z se determina en base a , al menos, el tamaño de bloque de código. En este caso, dado que el valor mínimo en el conjunto de tamaños de bloque de código que es mayor que 4000 es 4096, la longitud de bits del sistema de los códigos de LDPC (es decir, el número de bits ingresados al codificador de LDPC) es 4096 y el número de bits de relleno que se deben agregar es $4096 - 4000 = 96$. En el conjunto de tamaños de bloques de código, 4096 corresponde al factor de expansión de 512 y, por lo tanto, el factor de expansión utilizado en la codificación es $z = 512$. En base al factor de expansión $z=512$, todos los elementos de la matriz de verificación básica se modifican de acuerdo con la siguiente ecuación:

$$(h_{ij}^b)_{modificado} = \begin{cases} -1 & \text{si } (h_{ij}^b)_{uniforme} = -1 \\ (h_{ij}^b)_{uniforme} \times z/z_{max} & \text{de lo contrario} \end{cases}$$

4. Una secuencia de datos que se va a codificar se codifica en base a la matriz de verificación básica y el factor de expansión de codificación, para obtener una palabra de código de LDPC que tenga una longitud de 13312 bits. Una

palabra de código que tenga una longitud de código correspondiente se puede seleccionar de la palabra de código de LDPC en base a una tasa de codificación R. Por ejemplo, cuando la tasa de codificación $R = 1/2$, la longitud del código a transmitir es $4000/(1/2) = 8000$ bits. En este caso, se seleccionan 8000 bits de los 13312 bits para la transmisión. En particular, la selección se realiza de la siguiente manera. La palabra de código que tiene la longitud de $512 \times 26 = 13312$ se permuta en unidades de $z = 512$ con un vector de permutación de $[0 \sim 7, [8, 9, 11, 10], 12:25]$. A partir del $z \times 2 = 1024^{\circ}$ bit, la selección se realiza de manera envolvente, es decir, la selección vuelve al primer bit una vez que llega al último bit, y se omitirán los bits de relleno, para seleccionar una palabra de código de LDPC de $N = 8000$ bits. A continuación, se transmite la palabra de código de LDPC de $N = 8000$ bits.

En correspondencia con el ejemplo anterior, se proporciona un receptor para recibir la palabra de código de LDPC transmitida. Un método de decodificación de LDPC en el receptor incluye las siguientes etapas.

1. Los datos se reciben y demodulan para obtener información que tiene una longitud de 8000 bits blandos, es decir, una secuencia de datos que se va a decodificar y que tiene una longitud de 8000.

2. Se obtiene un tamaño de bloque de código (4000) para la codificación de LDPC estructurada. El tamaño de bloque de código se puede obtener a través de la señalización de la configuración del sistema.

3. Un factor de expansión de codificación z se determina en base a, al menos, el tamaño de bloque de código. Los métodos para determinar el factor de expansión de codificación z y modificar la matriz de verificación básica son los mismos que los realizados en el transmisor como se describió anteriormente y los detalles de los mismos se omitirán en el presente documento.

4. La secuencia de datos que se va a decodificar y que tiene una longitud de 8000 se decodifica en base a la matriz de verificación básica y el factor de expansión de codificación. Dado que la longitud de código real de la matriz de verificación básica es 13312 y el número de bits de relleno es 96, la información de bits blandos en las posiciones de relleno se puede establecer en un valor muy grande, y la información de los 8000 bits blandos se puede colocar en las posiciones de los bits seleccionados en la palabra de código correspondiente que tiene la longitud de 13312, mientras que otra información blanda (en posiciones de bits perforados o no transmitidos) puede ser igual a 0. A continuación, la secuencia de información suave que se va a decodificar y que tiene una longitud de 13312 se decodifica para emitir 4096 bits del sistema, de los que se eliminan los bits de relleno para obtener los datos que tienen una longitud de 4000 bits. Durante el proceso de decodificación, el conjunto correspondiente de factores de expansión de codificación es $[2 \ 4 \ 6 \ 8 \ 10 \ 12 \ 16 \ 20 \ 24 \ 32 \ 40 \ 48 \ 64 \ 80 \ 96 \ 128 \ 160 \ 192 \ 256 \ 320 \ 384 \ 512 \ 640 \ 768 \ 1024 \ 1280]$. El conjunto de factores de expansión incluye productos de 5 y 2 a la potencia de $[1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$, es decir, $[10 \ 20 \ 40 \ 80 \ 160 \ 320 \ 640 \ 1280]$, denotados como Conjunto Z0; productos de 3 y 2 a la potencia de $[1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$, es decir, $[6 \ 12 \ 24 \ 48 \ 96 \ 192 \ 384 \ 768]$, denotados como Conjunto Z1; productos de 2 y 2 a la potencia de $[1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$, es decir, $[4 \ 8 \ 16 \ 32 \ 64 \ 128 \ 256 \ 512]$, denotados como Conjunto Z2; y productos de 1 y 2 a la potencia de $[1 \ 2 \ 3 \ 4 \ 5 \ 6 \ 7 \ 8]$, es decir, $[2 \ 4 \ 8 \ 16 \ 32 \ 64 \ 128 \ 256]$, denotados como Conjunto Z3. Por lo tanto, el grado máximo de paralelismo de decodificación que se puede admitir en el decodificador es $2^8 = 256$, es decir, $P_{\text{máx}} = 256$. En el decodificador, se pueden almacenar 5 palabras como máximo, cada una tiene una longitud de 256. Cuando el factor de expansión utilizado en la codificación está en cualquiera de los conjuntos anteriores, por ejemplo, cuando está en el Conjunto Z0, la información blanda en el decodificador se almacena en 5 palabras; cuando está en el Conjunto Z1, la información blanda en el decodificador se almacena en 3 palabras; cuando está en el Conjunto Z2, la información blanda en el decodificador se almacena en 2 palabras; o cuando está en el Conjunto Z3, la información blanda en el decodificador se almacena en 1 palabra. Se puede utilizar el método de decodificación similar al descrito en la Realización 2, de tal manera que se pueda admitir un diseño de longitud de código flexible.

Además, los códigos de LDPC pueden incluir el siguiente ejemplo. La matriz de verificación básica, de la siguiente manera, tiene $m_b = 8$ filas, $k_b = 8$ columnas del sistema y $n_b = 16$ columnas. El conjunto de factores de expansión es $[18 \ 21 \ 30 \ 35 \ 42 \ 63 \ 70 \ 105 \ 126 \ 210]$, que incluye dos conjuntos $[6 \ 7 \ 14 \ 21 \ 42] \times 3$ y $[6 \ 7 \ 14 \ 21 \ 42] \times 5$. El subconjunto Pset es $[6 \ 7 \ 14 \ 21 \ 42]$ y el subconjunto nSet es $[3 \ 5]$. La matriz de verificación básica es igual a:

50	76	141	188	127	124	-1	194	127	-1	-1	-1	-1	-1	-1	-1
47	15	146	106	192	169	194	49	126	15	-1	-1	-1	-1	-1	-1
0	104	-1	22	16	101	185	106	39	39	113	-1	-1	-1	-1	-1
189	186	-1	57	139	15	57	67	145	-1	195	67	-1	-1	-1	-1
-1	44	79	150	65	49	193	183	-1	150	77	74	191	-1	-1	-1
149	-1	2	21	-1	24	37	69	175	26	4	42	190	68	-1	-1
123	199	31	-1	128	-1	-1	-1	132	119	9	-1	14	187	203	-1
-1	171	179	6	193	-1	198	-1	-1	165	188	175	64	110	150	1

Del mismo modo, el transmisor puede realizar el procesamiento de datos de acuerdo con las etapas anteriores, determinar el factor de expansión y la matriz de verificación básica y realizar una codificación de LDPC estructurada. Los detalles de estas etapas se omitirán en el presente documento. Del mismo modo, el receptor puede operar de acuerdo con las etapas anteriores. El grado máximo de paralelismo que puede utilizar el decodificador en el receptor

es el elemento más grande 42 en el subconjunto Pset. En el decodificador de LDPC, el grado de paralelismo de la decodificación puede ser 42 o cualquier factor entero menor de 42. En el decodificador, si el factor de expansión utilizado en la codificación es uno de $[6\ 7\ 14\ 21\ 42]*3$, la información blanda requerida para la decodificación se puede almacenar en un bloque de almacenamiento que consiste en 3 palabras. Si el factor de expansión utilizado en la codificación se determina como 63, el grado real de paralelismo de la decodificación es 31, ya que se divide en 3 palabras. Si el factor de expansión utilizado en la codificación es uno de $[6\ 7\ 14\ 21\ 42]*5$, la información blanda requerida para la decodificación se puede almacenar en un bloque de almacenamiento que consiste en 5 palabras. Cuando el factor de expansión utilizado en la codificación es 70, por ejemplo, el grado real de paralelismo de la decodificación es 14, ya que se divide en 5 palabras.

Realización 4

De acuerdo con una realización de la presente divulgación, también se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. El método puede implementar el método anterior para la codificación/decodificación de LDPC (los detalles del mismo se omitirán en el presente documento). Como se utiliza más adelante, el término "método" o "módulo" puede ser software, hardware o una combinación de los mismos, capaz de realizar una función predeterminada. Mientras que los aparatos que se van a describir en las siguientes realizaciones se implementan preferiblemente en software, se puede contemplar que también se pueden implementar en hardware o en una combinación de software y hardware.

De acuerdo con una realización de la presente divulgación, se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. El método se aplica en un transmisor e incluye las siguientes etapas.

1. Segmentación de Bloques de Código: Un bloque de transporte que tiene una longitud de CL se segmenta en una pluralidad de bloques de código.

2. Un tamaño de bloque de código para la codificación de LDPC estructurada se obtiene en base a un tamaño de bloque de código después de la segmentación del bloque de código.

3. Un factor de expansión de codificación z se determina en base al tamaño de bloque de código, un parámetro k_b de una matriz de verificación básica, un valor entero positivo p y la matriz de verificación básica. El factor de expansión z es n veces un número entero positivo pl . En particular, el número entero positivo pl es un elemento de un subconjunto Pset, que es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{\max}$, donde n es un número natural y $P_{\max}$ es un número entero mayor o igual que 4. Todos los valores del número entero positivo n constituyen un subconjunto nSet, que es un subconjunto de un conjunto de números enteros positivos, N_s , y tiene una longitud de F , donde el conjunto de números enteros positivos $N_s = \{1, 2, 3, \dots, M\}$, donde M es un número entero mayor que 1.

La matriz de verificación básica tiene los siguientes parámetros: el número de columnas del sistema, k_b , el número de filas (o columnas de verificación), mb , y el número total de columnas, nb , donde $k_b = nb - mb$.

La matriz de verificación básica incluye al menos dos tipos de elementos: 1) elementos, cada uno de los cuales indica una matriz cuadrada que es de ceros, denotada en el presente documento como -1 (que en su lugar se puede denotar como nula o cualquier otro descriptor); y 2) elementos, cada uno de los cuales indica una matriz cuadrada obtenida al desplazar cíclicamente una matriz de identidad (es decir, una matriz que no es de ceros), denotada como un número entero que es mayor o igual que 0 y menor que el factor de expansión e indica el número de bits en los que la matriz de identidad se desplaza cíclicamente. La matriz de verificación básica también puede incluir una estructura de múltiples lados, lo que significa que incluye al menos: dos elementos en el mismo par de índice de fila i e índice de columna, lo que indica una suma de dos matrices cuadradas desplazadas cíclicamente por un número particular de bits.

Cualquier elemento hb_{ij} en la matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{\max}/pl)) = 0$, donde $i=0, 1, 2, \dots, mb-1$, $j=0, 1, \dots, nb-1$, y $P_{\max}$ es un número entero mayor o igual que 4.

4. Una secuencia de datos que se va a codificar se codifica en base a la matriz de verificación básica y el factor de expansión de codificación.

Como se describió anteriormente, a continuación se proporciona una matriz de verificación básica utilizada en el método para la codificación de LDPC estructurada. Los parámetros son $P_{\max}=42$, $P_{\text{set}} = \{6\ 21\ 42\}$ y $n\text{Set} = \{3\ 5\}$. El valor del factor de expansión para la matriz de verificación básica depende de P_{set} y $n\text{Set}$, es decir, es igual a $3*P_{\text{set}}$ y $5*P_{\text{set}}$. Es decir, todos los factores de expansión constituyen un conjunto $\{18\ 30\ 63\ 105\ 126\ 210\}$. Se puede ver que P_{set} es un subconjunto de un conjunto de todos los factores enteros positivos de $P_{\max}$. A continuación se describen las matrices de verificación básicas correspondientes a los respectivos factores de expansión, donde el número de columnas del sistema $k_b=8$, el número de filas (o columnas de verificación) $mb=4$, el número total de nb columnas=12, es decir, $k_b=nb-mb$.

Para el factor de expansión de 18, correspondiente a $p=6$ y $n=3$, la matriz de verificación básica Hb_0 es:

ES 3 024 210 T3

14	14	14	7	14	7	7	0	7	-1	-1	-1
14	14	0	0	0	7	7	0	14	0	-1	-1
0	14	7	7	7	14	14	14	7	0	0	-1
0	14	7	0	0	7	14	14	7	7	7	14

Para el factor de expansión de 30, correspondiente a $p=6$ y $n=5$, la matriz de verificación básica Hb1 es:

7	7	21	28	28	21	28	0	7	-1	-1	-1
14	21	28	21	0	14	21	7	7	0	-1	-1
28	7	7	21	21	7	0	21	28	7	21	-1
28	7	21	21	21	7	28	7	28	7	21	21

Para el factor de expansión de 63, correspondiente a $p=21$ y $n=3$, la matriz de verificación básica matriz Hb2 es:

60	32	20	48	56	10	4	34	4	-1	-1	-1
4	2	14	22	20	38	8	38	36	44	-1	-1
48	54	52	54	32	24	56	60	46	30	4	-1
28	40	8	52	0	52	58	6	16	50	10	40

Para el factor de expansión de 105, correspondiente a $p=21$ y $n=5$, la matriz de verificación básica matriz Hb3 es:

52	40	94	70	2	2	12	14	90	-1	-1	-1
96	12	44	22	100	94	64	54	14	100	-1	-1
16	82	64	90	62	98	30	36	14	24	62	-1
14	42	8	4	50	62	48	54	34	46	0	80

Para el factor de expansión de 126, correspondiente a $p=42$ y $n=3$, la matriz de verificación básica Hb4 es:

37	83	46	86	83	29	52	83	122	-1	-1	-1
98	26	51	71	48	65	109	27	1	2	-1	-1
24	76	54	40	68	19	35	80	76	37	109	-1
71	43	108	27	39	18	121	52	104	70	45	117

Para el factor de expansión de 210, correspondiente a $p=42$ y $n=5$, la matriz de verificación básica Hb5 es:

187	58	193	192	136	18	124	102	110	-1	-1	-1
121	74	189	200	85	103	71	123	52	52	-1	-1
120	46	147	101	51	36	126	147	5	152	51	-1
96	41	41	170	146	51	169	49	58	193	129	85

Se puede ver a partir de las matrices de verificación básicas anteriores que cualquier elemento hb_{ij} en cada matriz de verificación básica que indique una matriz cuadrada que no es de ceros satisface $\text{mod}(hb_{ij}, (P_{\text{máx}}/p)) = 0$, donde hb_{ij} en cada matriz de verificación básica es un elemento que no es igual a -1.

Los efectos ventajosos de utilizar las matrices de verificación básicas anteriores como matrices de verificación básicas para la codificación de LDPC son similares a los descritos en relación con la Realización 2. En este caso, la información se almacena en direcciones en unidades de palabras, lo que hace que el almacenamiento de información sea muy simple y fácil de admitir un diseño de longitud de código flexible. El Hb0 anterior se toma como ejemplo en la siguiente descripción. En este caso, el módulo de almacenamiento del decodificador se muestra en la FIG. 10. El número total de columnas de la matriz de verificación básica es 12. De acuerdo con lo anterior, hay 12 bloques de almacenamiento que tiene 5 palabras cada uno. Como se muestra en la FIG. 10, 1001 corresponde al bloque de almacenamiento para la Columna 0 de la matriz de verificación básica y 1002 corresponde a la Columna 1 de la matriz de verificación básica. El tamaño de cada palabra es igual a $P_{\text{máx}}=42$, es decir, el grado máximo de paralelismo admitido es $P_{\text{máx}}$. Los bloques de almacenamiento se pueden compartir para diferentes factores de expansión. Para Hb0 y Hb1, se pueden utilizar los módulos de almacenamiento correspondientes a 1003 como se muestra en la FIG. 10, con el color negro que indica la información almacenada. Dado que en este caso los factores de expansión son 18 y 30, la información de bits blandos correspondiente a cada columna de la matriz de verificación básica incluye únicamente 18 y 30. Por

lo tanto, la información de bits blandos se coloca cada 7 bits en cada palabra. Hb0 utiliza 3 palabras y Hb1 utiliza 5 palabras.

Del mismo modo, para Hb2 y Hb3, se pueden utilizar los módulos de almacenamiento correspondientes a 1004 como se muestra en la FIG. 10, con el color negro indicando la información almacenada. Dado que en este caso los factores de expansión son 63 y 105, la información de bits blandos correspondiente a cada columna de la matriz de verificación básica incluye solo 63 y 105. Por lo tanto, la información de bits blandos se coloca cada 2 bits en cada palabra. Hb2 utiliza 3 palabras y Hb3 utiliza 5 palabras.

Del mismo modo, para Hb4 y Hb5, se pueden utilizar los módulos de almacenamiento correspondientes a 1005 como se muestra en la FIG. 10, con el color negro indicando la información almacenada. Dado que en este caso los factores de expansión son 126 y 210, la información de bits blandos correspondiente a cada columna de la matriz de verificación básica incluye solo 126 y 210. Por lo tanto, cada palabra se rellena con la información de bits blandos. Hb2 utiliza 3 palabras y Hb3 utiliza 5 palabras.

Con el conjunto de factores de expansión y la matriz de verificación básica como se describió anteriormente, es ventajoso porque solo un decodificador de LDPC es suficiente, lo que reduce significativamente la complejidad debido al uso de una serie de decodificadores. Además, la red de entrelazado de cada columna en la matriz de verificación básica se puede compartir o implementar utilizando una red de circuito fijo, de tal manera que se pueda reducir la complejidad y la latencia de decodificación.

Además, se proporciona una matriz de verificación básica de la siguiente manera, donde el factor de expansión es 63, correspondiente a $p = 21$ y $n = 3$. La matriz de verificación básica Hb6 satisface $\text{mod}(h_{ij}, (P_{\text{máx}}/p)) = A$, donde $A=1$:

51	11	1	17	51	23	57	33	33	-1	-1	-1
27	53	53	13	41	57	57	17	9	15	-1	-1
3	19	61	59	9	45	13	39	33	55	29	-1
41	43	31	43	7	13	43	41	7	51	29	43

Los métodos de acuerdo con las realizaciones opcionales de la presente divulgación se pueden aplicar en el sistema de comunicación móvil de Evolución a Largo Plazo (LTE) o en el sistema de comunicación móvil de 5ª Generación (5G) en el futuro. La dirección de transmisión de datos puede ser datos transmitidos desde una estación base hasta un dispositivo de usuario móvil (es decir, datos de tráfico de transmisión de enlace descendente) o datos transmitidos desde un dispositivo de usuario móvil hasta una estación base (es decir, datos de tráfico de transmisión de enlace ascendente). El dispositivo móvil de usuario puede incluir un dispositivo móvil, que se puede denominar como un terminal de acceso, un terminal de usuario, una estación de usuario, una unidad de usuario, una estación móvil, una estación remota, un terminal remoto, un agente de usuario, un equipo de usuario, un dispositivo de usuario o cualquier otra terminología. Una estación base puede incluir un Punto de Acceso (AP), que se puede denominar Nodo B, un Controlador de Red de Radio (RNC), un Nodo B evolucionado (eNB), un Controlador de Estación Base (BSC), una Estación Transceptora Base (BTS), una Estación Base (BS), una Función de Transceptor (TF), un enrutador de radio, un transceptor de radio, un Conjunto de Servicios Básicos (BSS), un Conjunto de Servicio Extendido (ESS), una Estación Base de Radio (RBS) o cualquier otra terminología.

De acuerdo con un aspecto de una realización opcional de la presente divulgación, se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. Se puede aplicar en escenarios tales como la Banda Ancha Móvil mejorada (eMBB), las Comunicaciones Ultra Confiables de Baja Latencia (URLLC) o las Comunicaciones de Tipo Máquina masivas (mMTC) en la nueva Tecnología de Acceso de Radio (RAT). En el escenario eMBB, el rendimiento máximo de enlace descendente puede ser de hasta 20 Gbps y el rendimiento máximo de datos de enlace ascendente puede ser de hasta 10 Gbps. En el escenario de URLLC, se puede admitir una Tasa de Error de Bloque (BLER) tan baja como $10e-5$ en términos de confiabilidad, y la latencia más baja en el enlace ascendente y descendente podría ser tan baja como 0.5 ms. En el escenario mMTC, las baterías de los dispositivos pueden durar años. El método para el procesamiento de datos con códigos de LDPC estructurados de acuerdo con esta realización incluye las siguientes etapas.

1. Se obtiene un tamaño de bloque de código para la codificación de LDPC estructurada.

2. Un factor de expansión de codificación z se determina en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica o un valor entero positivo p .

3. Se codifica una secuencia de datos que se va a codificar, o se decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación.

Opcionalmente, la operación de determinación del factor de expansión de codificación basado en el tamaño de bloque de código CBS, el parámetro kb de la matriz de verificación básica y el valor entero positivo p incluye: determinar el factor de expansión de codificación de acuerdo con: $\lceil CBS/(kb \times p) \rceil \times p$, donde $\lceil \cdot \rceil$ denota una operación de techo.

Opcionalmente, la operación de obtención del tamaño de bloque de código para la codificación de LDPC estructurada incluye: el tamaño de bloque de código que es igual a un valor obtenido al multiplicar el parámetro k_b de la matriz de verificación básica por uno de un conjunto de números naturales ordenados en orden ascendente. El conjunto de números naturales ordenados en orden ascendente incluye una pluralidad de subconjuntos, cada uno de los cuales tiene una primera diferencia igual entre los elementos vecinos. La primera diferencia para el i -ésimo subconjunto en el conjunto de números naturales ordenados en orden ascendente es menor que la primera diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo. Con este diseño, es ventajoso en el sentido de que los números naturales en el conjunto de números naturales ordenados en orden ascendente aumentan etapa a etapa, de tal manera que los tamaños de bloque de código que admiten varían en un intervalo pequeño para una longitud pequeña y para una longitud grande. Es decir, no se necesitarían tamaños de bloques de código más densos, de tal manera que el sistema de comunicación pueda ser más flexible. Más aún, se necesitarían menos indicaciones de señalización y menos interacciones de señalización, de tal manera que se pueda reducir la complejidad del sistema.

Alternativamente, opcionalmente, la operación de determinación del factor de expansión de codificación basado en el tamaño de bloque de código incluye: el tamaño de bloque de código es uno de un conjunto de tamaños de bloque de código, el conjunto de tamaños de bloque de código es un primer conjunto de números naturales almacenados en orden ascendente; y el factor de expansión de codificación es uno de un conjunto de factores de expansión de codificación, el conjunto de factores de expansión de codificación es un segundo conjunto de números naturales almacenados en orden ascendente. Opcionalmente, el conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times k_b$ y el último de los elementos sucesivos es menor o igual que $z(j+1) \times k_b$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo, $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación, j es un número entero positivo y k_b es el parámetro de la matriz de verificación básica. El factor de expansión de codificación es $z(j+1)$ cuando el tamaño de bloque de código obtenido es uno de los elementos sucesivos. El conjunto de factores de expansión de codificación incluye una pluralidad de subconjuntos, cada uno de los cuales tiene una segunda diferencia igual entre los elementos vecinos. La segunda diferencia para el h -ésimo subconjunto de la pluralidad de subconjuntos es menor que la segunda diferencia para el $(h+1)$ -ésimo subconjunto, donde h es un número entero positivo. El conjunto de tamaños de bloques de código incluye al menos los productos del parámetro k_b de la matriz de verificación básica y los respectivos del conjunto de factores de expansión de codificación, y números enteros espaciados por un número entero positivo B entre todos los productos.

Alternativamente, el factor de expansión de codificación se determina como $z(j+1)$ cuando una relación entre el tamaño de bloque de código y el parámetro k_b de la matriz de verificación básica es mayor que $z(j)$ y menor o igual que $z(j+1)$, donde $z(j)$ es el j -ésimo elemento del conjunto de factores de expansión de codificación. Cada uno de los elementos del conjunto de factores de expansión de codificación que es mayor que el primer valor entero positivo p es un múltiplo del primer valor entero positivo p . El valor entero positivo p es un grado de paralelismo de la decodificación de LDPC. El propósito de limitar el factor de expansión a igual a un múltiplo del grado p de paralelismo es que el proceso de decodificación de LDPC en capas puede eliminar el problema de colisión de direcciones en las redes de entrelazado y desentrelazado en el decodificador, sin ninguna sobrecarga adicional de indicaciones de circuito de hardware, de tal manera que la complejidad de decodificación se puede reducir y se requiere un área más pequeña de circuito de hardware.

Opcionalmente, el método puede incluir además, antes de codificar la secuencia de datos que se va a codificar en base a la matriz de verificación básica y el factor de expansión de codificación: agregar bits de relleno a una primera secuencia de datos al dividir la primera secuencia de datos en una pluralidad de subsecuencias de datos y agregar bits de relleno a cada una de las pluralidades de subsecuencias de datos, de tal manera que todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar. Alternativamente, el método puede incluir además, antes de codificar la secuencia de datos que se va a codificar en base a la matriz de verificación básica y el factor de expansión de codificación: agregar bits de relleno a una segunda secuencia de datos al dividir la segunda secuencia de datos en subsecuencias de datos en k_b , cada una de las cuales incluye z o menos bits, y agregar bits de relleno a cada una de las subsecuencias de datos de tal manera que cada una de las subsecuencias de datos incluya z bits y todas las subsecuencias de datos con bits de relleno constituyen la secuencia de datos que se va a codificar, donde k_b es el parámetro de la matriz de verificación básica y z es el factor de expansión de codificación. El método puede incluir además, antes de agregar los bits de relleno a la primera o segunda secuencia de datos: agregar una secuencia de Verificación de Redundancia Cíclica (CRC) de bits L a una tercera secuencia de datos para obtener la primera o segunda secuencia de datos, donde L es un número entero mayor o igual que 0. La FIG. 9 muestra una comparación de rendimiento en un canal de ruido blanco aditivo. En el presente documento, el tamaño de bloque de código es 1120, el factor de expansión de los códigos de LDPC estructurados es 150, el número de columnas del sistema de la matriz de verificación básica es $k_b = 8$ y el número de bits de relleno es 80. En la figura, el esquema de relleno convencional (es decir, en el que se agregan 80 bits de relleno en la porción principal o en la porción final) se denota como padding1 y el esquema de acuerdo con la presente divulgación se denota como padding2. 1120 se divide en 8 subsecuencias de datos, cada una tiene 140 bits más 10 bits de relleno (es decir, 150 bits). Las subsecuencias de datos se combinan para obtener una secuencia de bits del sistema. A partir de los rendimientos mostrados en la FIG. 9, se puede ver que la solución de relleno de acuerdo con la presente divulgación puede obtener una ganancia mayor (aproximadamente 0.4 dB). Alternativamente, antes de codificar la secuencia de datos que se va a codificar en base a la matriz de verificación básica y el factor de expansión de codificación, se

incluye una operación de relleno de bits, que incluye: dividir la primera secuencia de datos en una pluralidad de subsecuencias de datos y agregar bits de relleno a cada una de las pluralidades de subsecuencias de datos, de tal manera que todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar. Antes de codificar la secuencia de datos que se va a codificar en base a la matriz de verificación básica y el factor de expansión de codificación, se incluye una operación de relleno de bits, que incluye: dividir la primera secuencia de datos en kb subsecuencias de datos, cada una de las cuales incluye z o menos bits, y agregar bits de relleno a cada una de las subsecuencias de datos de tal manera que cada una de las subsecuencias de datos incluya z bits y todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar, donde kb es el parámetro de la matriz de verificación básica y z es el factor de expansión de codificación. Antes de la operación de relleno de bits, se puede agregar una secuencia de Verificación de Redundancia Cíclica (CRC) de bits L a una segunda secuencia de datos para obtener la primera secuencia de datos, donde L es un número entero mayor o igual que 0.

El valor entero positivo p anterior es un número entero positivo fijo. El grado p de paralelismo de la decodificación de LDPC incluye una potencia entera positiva de 2 o un producto de una potencia entera positiva de 2 y un número primo. Es ventajoso en que p puede ser un múltiplo de un valor pequeño. En el proceso de decodificación de LDPC por capas, si se desea una menor complejidad de hardware, aún se puede utilizar un grado p menor de paralelismo de la decodificación en capas, que a su vez puede soportar un diseño de circuito con un menor consumo de energía para ser utilizado, por ejemplo, en el escenario nMTC.

Opcionalmente, el parámetro kb de la matriz de verificación básica es una diferencia entre el número de columnas de la matriz de verificación básica y el número de filas de la matriz de verificación básica, donde kb es un número entero mayor o igual que 4 y menor o igual que 64. El parámetro kb de la matriz de verificación básica es el número de columnas del sistema de la matriz de verificación básica. El factor de expansión de codificación es igual a una potencia entera positiva de 2 menos 1, o un producto de una potencia entera positiva de 2 y un número primo. En particular, el factor de expansión de codificación incluye uno de: 7, 15, 31, 63, 127, 255, 511, 1023, 2047 o 4095.

A continuación se da un ejemplo más específico.

Ejemplo 1

La FIG. 6 es un diagrama esquemático que muestra una matriz de verificación básica de acuerdo con una realización opcional de la presente divulgación. En las comunicaciones móviles 5G, en un transmisor en un escenario eMBB, un paquete de datos de origen que se va a transmitir tiene un tamaño de 7952 bits y un tamaño de bloque de código es de 4000. Se sabe que la matriz de verificación básica (16 filas*24 columnas) es la siguiente. El factor de expansión máximo admitido es $z_{\max}=1000$.

Se puede saber que la matriz de verificación básica como se muestra en la FIG. 6 tiene los siguientes parámetros: el número de filas de la matriz de verificación básica $m_b = 16$, el número de columnas de la matriz de verificación básica $n_b = 24$ y el número de columnas del sistema de la matriz de verificación básica $k_b = n_b - m_b = 8$. Se supone que el grado de paralelismo utilizado en la decodificación es $p = 20$, cada tamaño de bloque de código tiene una secuencia de CRC de 24 bits y la tasa de codificación es 1/2. Como se describió anteriormente, el método para el procesamiento de datos con códigos de LDPC estructurados incluye las siguientes etapas.

En la etapa 1, se obtiene un tamaño de bloque de código para la codificación de LDPC estructurada, es decir, $CBS=4000$.

En la etapa 2, se determina un factor de expansión de codificación z en base a al menos uno del tamaño del bloque de código, un parámetro kb de una matriz de verificación básica o un primer valor entero positivo p.

En la etapa 2, la operación de determinación del factor de expansión de codificación basado en el tamaño de bloque de código de CBS, el parámetro kb de la matriz de verificación básica y el primer valor entero positivo p incluye: determinación del factor de expansión de codificación de acuerdo con: $[CBS/(k_b \times p)] \times p$, donde $\lceil \cdot \rceil$ denota una operación de techo. Se puede saber que el factor de expansión es $z'=500$.

En la etapa 3, se codifica una secuencia de datos que se va a codificar en base a la matriz de verificación básica y el factor de expansión de codificación.

En la etapa 3, dado que la matriz de verificación básica anterior se basa en el factor de expansión de 1000, se debe modificar de acuerdo con:

$$(h_{ij}^b)_{\text{modificado}} = \begin{cases} -1 & (h_{ij}^b)_{\text{uniforme}} = -1 \\ \lfloor (h_{ij}^b)_{\text{uniforme}} \times z/z_{\max} \rfloor & (h_{ij}^b)_{\text{uniforme}} \neq -1 \end{cases}$$

donde $(h_{ij}^b)_{modificado}$ es el valor del elemento en la i-ésima fila y la j-ésima columna de la matriz de verificación básica modificada, $(h_{ij}^b)_{uniforme}$ es el valor del elemento en la i-ésima fila y la j-ésima columna de la matriz de verificación básica anterior. Dado que la tasa de codificación es 1/2, el número de filas de la matriz de verificación básica según se requiera se puede calcular a partir de kb , como: $kb/R = 16$. Los elementos de 8 filas y 16 columnas se pueden extraer directamente de la matriz de verificación básica modificada, o los elementos de 8 filas y 16 columnas se pueden extraer primero de la matriz de verificación básica y luego modificar. La FIG. 7 es un primer diagrama esquemático que muestra una matriz de verificación básica Hb' de acuerdo con una realización opcional de la presente divulgación. Como se muestra en la FIG. 7, se puede obtener la matriz de verificación básica Hb' que se utilizará en la codificación real.

La segmentación de bloques de código se puede aplicar al paquete de datos de origen que tiene un tamaño de 7952 bits, el número de bits en cada paquete de datos es lo más cercano posible a $CBS-L=3976$, donde L es el número de bits de la secuencia de CRC) para obtener dos segundas secuencias de datos cada una con tiene longitud de 3976 bits. Se agrega una secuencia de CRC de $L=24$ bits a cada una de las dos segundas secuencias de datos para obtener dos primeras secuencias de datos, cada una tiene una longitud de 4000 bits. Dado que el número de los bits del sistema real de la LDPC estructurada es $kb \times z' = 8 \times 500 = 4000$ y $kb \times z = 8 \times 500 = 4000$, el tamaño de bloque de código ya es igual al número de bits reales del sistema (4000) y, por lo tanto, no se requiere relleno adicional.

Las secuencias de datos que se van a codificar (las dos primeras secuencias de datos tienen una longitud de 4000 bits cada una) se codifican en base a la matriz de verificación básica $Hb'=8$ anterior y el factor de expansión de codificación $z'=500$ que se utilizará en la codificación real, para obtener dos bloques de código de LDPC que tienen cada uno una longitud de 8000 bits. Los bloques de código de LDPC se modulan y transmiten.

En un receptor, los datos recibidos se pueden demodular para obtener dos secuencias de datos de información blanda (es decir, secuencias de datos que se van a decodificar), cada una tiene una longitud de 8000 bits. A continuación, cada una de las secuencias de datos de información blanda (cada una tiene una longitud de 8000 bits) se decodifica en base a la matriz de verificación básica Hb' anterior y el factor de expansión de codificación $z' = 500$ utilizado en la codificación real, para obtener una secuencia de bits del sistema de la codificación de LDPC estructurada, de la que se extrae una secuencia de CRC de 24 bits para obtener el paquete de datos de origen con una longitud de 7952 bits.

Ejemplo 2

Este ejemplo difiere del Ejemplo 1 en que el paquete de datos de origen que se va a transmitir tiene un tamaño de 3256 bits. Se supone que el tamaño de bloque de código para los códigos de LDPC estructurados es 3360. El tamaño de bloque de código para la codificación de LDPC estructurada es igual a un valor obtenido al multiplicar el parámetro $kb=8$ de la matriz de verificación básica por uno de un conjunto de números naturales ordenados en orden ascendente. El conjunto de números naturales ordenados en orden ascendente incluye una pluralidad de subconjuntos, cada uno de los cuales tiene una primera diferencia igual entre los elementos vecinos. La primera diferencia para el i-ésimo subconjunto en el conjunto de números naturales ordenados en orden ascendente es menor que la primera diferencia para el $(i+1)$ -ésimo subconjunto, donde i es un número entero positivo. Por ejemplo, el conjunto de números naturales ordenados en orden ascendente puede ser $[[2:1:200], [202:2:400], [406:6:796], [804:8:1004]]$.

La relación entre el tamaño de bloque de código 3360 y el parámetro $kb = 8$ de la matriz de verificación básica es 420, que es mayor que $z(302) = 418$ y menor o igual que $z(303) = 424$. Por lo tanto, el factor de expansión de codificación se determina como $z(303)=424$. La matriz de verificación básica se modifica de acuerdo con $z'=424$. La FIG. 8 es un segundo diagrama esquemático que muestra una matriz de verificación básica Hb' de acuerdo con una realización opcional de la presente divulgación. Como se muestra en la FIG. 8, se puede obtener la siguiente matriz de verificación básica Hb' . Es decir, se cumple la siguiente condición. El conjunto de tamaños de bloques de código incluye un número, a , de elementos sucesivos. Cuando el primero de los elementos sucesivos es mayor que $z(j) \times kb$ y el último de los elementos sucesivos es menor o igual que $z(j+1) \times kb$, el factor de expansión de codificación correspondiente a los elementos sucesivos es $z(j+1)$, donde a es un número entero positivo, $z(j)$ es el j-ésimo elemento del conjunto de factores de expansión de codificación, j es un número entero positivo y kb es el parámetro de la matriz de verificación básica. El factor de expansión de codificación es $z(j+1)$ cuando el tamaño de bloque de código obtenido es uno de los elementos sucesivos.

También se cumple la siguiente condición. El conjunto de factores de expansión de codificación incluye una pluralidad de subconjuntos, cada uno de los cuales tiene una segunda diferencia igual entre los elementos vecinos. La segunda diferencia para el h-ésimo subconjunto de la pluralidad de subconjuntos es menor que la segunda diferencia para el $(h+1)$ -ésimo subconjunto, donde h es un número entero positivo.

Dado que el número de bits reales del sistema para la LDPC estructurado es $kb \times z = 8 \times 424 = 3392$, se agrega una secuencia de CRC de $L=24$ bits al paquete de datos de origen (es decir, la tercera secuencia de datos) tiene un tamaño de 3256 bits, para obtener una segunda secuencia de datos que tiene una longitud de 3384 bits. En este caso, se pueden agregar 8 bits de relleno para obtener 3392 bits, por ejemplo, al dividir la segunda secuencia de datos en $kb = 8$ subsecuencias de datos, cada una de las cuales incluye $z = 434$ bits o menos, y agregar 1 bit de relleno a cada una de las subsecuencias de datos de tal manera que cada una de las subsecuencias de datos incluya $z = 434$ bits y todas las subsecuencias de datos con los bits de relleno constituyan la secuencia de datos que se va a codificar y

tienen una longitud de 3392. La secuencia de datos que se va a codificar (la secuencia de datos que tiene una longitud de 3392 bits) se codifica en base a la matriz de verificación básica H_b' y el factor de expansión de codificación $z' = 434$ que se utilizará en la codificación real, para obtener un bloque de código de LDPC que tiene una longitud de 6784 bits. El bloque de código de LDPC se modula y se transmite.

- 5 Opcionalmente, en las realizaciones y realizaciones opcionales de la presente divulgación, el número de columnas de la matriz es nb , el número de filas de la matriz es mb y el número de columnas del sistema de la matriz es $kb=nb-mb$.

Las realizaciones anteriores solo ilustran, en lugar de limitar, las soluciones de la presente divulgación. Se pueden hacer varias modificaciones y alternativas a las soluciones de la presente divulgación por los expertos en la técnica sin apartarse del alcance de la presente divulgación. El alcance de la presente divulgación se define por las reivindicaciones adjuntas.

Realización 5

De acuerdo con una realización de la presente divulgación, también se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. Como se utiliza más adelante, el término "método" o "módulo" puede ser software, hardware o una combinación de los mismos, capaz de realizar una función predeterminada. Mientras que los aparatos que se describirán en las siguientes realizaciones se implementan preferiblemente en software, se puede contemplar que también se pueden implementar en hardware o en una combinación de software y hardware.

De acuerdo con una realización de la presente divulgación, se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. El método se aplica en un transmisor e incluye las siguientes etapas.

1. Segmentación de Bloques de Código: Un bloque de transporte que tiene una longitud de CL se segmenta en una pluralidad de bloques de código.

2. Un tamaño de bloque de código para la codificación de LDPC estructurada se obtiene en base a un tamaño de bloque de código después de la segmentación del bloque de código.

3. Un factor de expansión de codificación z se determina en base al tamaño de bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p y la matriz de verificación básica. El factor de expansión z es n veces un número entero positivo pl . En particular, el número entero positivo pl es un elemento de un subconjunto P_{set} , y todos los valores del número entero positivo n constituyen un subconjunto $nSet$. $K_{m\acute{a}x} \leq kb \times pl_{m\acute{a}x} \times nm\acute{a}x \leq 1.2 \times K_{m\acute{a}x}$, donde $pl_{m\acute{a}x}$ es el valor más grande en el subconjunto P_{set} , $nm\acute{a}x$ es el valor más grande en el subconjunto $nSet$ y $K_{m\acute{a}x}$ es un número entero mayor que 1024.

Se obtiene la matriz de verificación básica para la codificación de LDPC estructurada. La matriz de verificación básica tiene los siguientes parámetros: el número de columnas del sistema, kb , el número de filas (o columnas de verificación), mb , y el número total de columnas, nb , donde $kb=nb-mb$. La matriz de verificación básica incluye al menos dos tipos de elementos: 1) elementos, cada uno de los cuales indica una matriz cuadrada que es de ceros, denotada en el presente documento como -1 (que en su lugar se puede denotar como nula o cualquier otro descriptor); y 2) elementos, cada uno de los cuales indica una matriz cuadrada obtenida al desplazar cíclicamente una matriz de identidad (es decir, una matriz que no es de ceros), denotada como un número entero que es mayor o igual que 0 y menor que el factor de expansión e indica el número de bits en los que la matriz de identidad se desplaza cíclicamente. La matriz de verificación básica también puede incluir una estructura de múltiples lados, lo que significa que incluye al menos dos elementos en el mismo par de índice de fila e índice de columna, lo que indica una suma de dos matrices cuadradas desplazadas cíclicamente por un número particular de bits.

4. Se codifica una secuencia de datos que se va a codificar, o se decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación z . El número de bits en la secuencia de datos que se va a codificar es $kb \cdot z$ y la longitud de la secuencia de datos que se va a decodificar es $nb \cdot z$.

Además, $K_{m\acute{a}x}$ es igual a 2000, 2048, 4000, 4096, 6000, 6144, 8000, 8192, 12000 o 12288.

En el presente documento se da un ejemplo más específico.

En el método de procesamiento de datos con códigos de LDPC estructurados, los valores del factor de expansión z son {4 6 8 10 12 14 16 20 24 28 32 40 48 56 64 80 96 112 128 160 192 224 256 320 384 448 512 640 768 896 1024} y la matriz de verificación básica para los códigos de LDPC (correspondiente al factor de expansión máximo de 1024) es:

[illegible]

5 128} y nSet es igual a {2 3 5 6 7 8}. En este ejemplo, $K_{\max}=8000$. El valor más grande del subconjunto Pset es $pl_{\max}=128$, y el valor más grande del subconjunto nSet es $nm_{\max}=8$. Se puede ver que $kb \times pl_{\max} \times nm_{\max} = 8 \times 128 \times 8 = 8192$, lo que satisface $K_{\max} \leq kb \times pl_{\max} \times nm_{\max} \leq 1.2 \times K_{\max}$.

10 codificación de LDPC estructurada es $CBS = 1024$. El valor más pequeño que es mayor o igual que $CBS/kb=1024/8=256$ se selecciona del conjunto de factores de expansión, como el factor de expansión de codificación para los códigos de LDPC estructurados. En este caso, se sabe que el factor de expansión es 256. La matriz de verificación básica se modifica de acuerdo con la Realización 5 para obtener una matriz de verificación básica de codificación que tiene un factor de expansión de 256 (los detalles de la misma se omitirán en el presente documento).

15 La secuencia de datos que se va a codificar y que tiene una longitud de 1024 bits se puede codificar en base al factor de expansión obtenido y la matriz de verificación básica de codificación. Cuando la tasa de codificación es 1/3, se puede obtener una palabra de código de LDPC con una longitud de 3072 bits. En un receptor, la secuencia de datos que se va a decodificar y que tiene una longitud de 3072 bits se puede decodificar en base al factor de expansión obtenido y la matriz de verificación básica de codificación, para obtener datos decodificados que tienen una longitud

20 de 1024 bits. Los datos decodificados de todos los bloques de código se pueden combinar para obtener los datos originales del bloque de transporte transmitidos desde el transmisor.

Se puede utilizar la implementación de hardware del decodificador, como se describe en relación con la Realización 3, lo que es ventajoso al permitir que los códigos de LDPC soporten una longitud de código flexible y una tasa de código flexible.

25 Realización 6

30 se puede contemplar que también se pueden implementar en hardware o en una combinación de software y hardware.

De acuerdo con una realización de la presente divulgación, se proporciona un método para el procesamiento de datos con códigos de LDPC estructurados. El método se aplica en un transmisor e incluye las siguientes etapas.

1. Segmentación de Bloques de Código: Un bloque de transporte que tiene una longitud de CL se segmenta en una pluralidad de bloques de código.

35 2. Un tamaño de bloque de código para la codificación de LDPC estructurada se obtiene en base a un tamaño de bloque de código después de la segmentación del bloque de código.

40 Pset, y todos los valores del número entero positivo n constituyen un subconjunto $nSet$. $K_{m\acute{a}x} \leq k_b \times p_{m\acute{a}x} \times n_{m\acute{a}x} \leq 1.2 \times K_{m\acute{a}x}$, donde $p_{m\acute{a}x}$ es el valor más grande en el subconjunto $Pset$, $n_{m\acute{a}x}$ es el valor más grande en el subconjunto $nSet$ y $K_{m\acute{a}x}$ es un número entero mayor que 1024. Se obtiene la matriz de verificación básica para la codificación de LDPC estructurada.

5 otro descriptor); y 2) elementos, cada uno de los cuales indica una matriz cuadrada obtenida al desplazar cíclicamente una matriz de identidad (es decir, una matriz que no es de ceros), denotada como un número entero que es mayor o igual que 0 y menor que el factor de expansión e indica el número de bits en los que la matriz de identidad se desplaza cíclicamente. La matriz de verificación básica también puede incluir una estructura de múltiples lados, lo que significa que incluye al menos: dos elementos en el mismo par de índice de fila e índice de columna, lo que indica una suma de dos matrices cuadradas desplazadas cíclicamente por un número particular de bits.

15 Además, $K_{\max}$ es igual a 2000, 2048, 4000, 4096, 6000, 6144, 8000, 8192, 12000 o 12288.

En el método de procesamiento de datos con códigos de LDPC estructurados, los valores del factor de expansión z son {4 6 8 10 12 14 16 20 24 28 32 40 48 56 64 80 96 112 128 160 192 224 256 320 384 448 512 640 768 896 1024} y la matriz de verificación básica para los códigos de LDPC (correspondiente al factor de expansión máximo de 1024) es:

[illegible]

de un número entero positivo i , donde i es igual a $\{1\ 2\ 3\ 4\ 5\ 6\ 7\}$, es decir, P_{set} es igual a $\{2\ 4\ 8\ 16\ 32\ 64\ 128\}$, y n_{set} es igual a $\{2\ 3\ 5\ 6\ 7\ 8\}$. De esta manera, se puede obtener el conjunto anterior de valores del factor de expansión. Se puede ver que tiene más factores enteros positivos, lo que hace que el decodificador sea más amigable. Es decir, se pueden utilizar más grados de paralelismo de decodificación para satisfacer los requisitos de rendimiento en diferentes escenarios. Se puede utilizar un alto grado de paralelismo de decodificación cuando se desea un alto rendimiento, o un bajo grado de paralelismo de decodificación cuando se desea un rendimiento bajo. El grado de paralelismo de la decodificación normalmente es un factor entero positivo del factor de expansión, de lo contrario, la implementación del hardware sería muy complicada. En este ejemplo, $K_{m\acute{a}x}=8000$. El valor más grande del subconjunto P_{set} es $pl_{m\acute{a}x}=128$, y el valor más grande del subconjunto n_{set} es $nm_{\acute{a}x}=8$. Se puede ver que $kb \times pl_{m\acute{a}x} \times nm_{\acute{a}x} = 8 \times 128 \times 8 = 8192$, lo que satisface $K_{m\acute{a}x} \leq kb \times pl_{m\acute{a}x} \times nm_{\acute{a}x} \leq 1.2 \times K_{m\acute{a}x}$.

La matriz básica anterior tiene las siguientes características. Durante un proceso de transmisión, cuando la tasa de codificación R es superior a un R_0 de código base, no se transmiten los $q \cdot Z$ bits en q de las columnas del sistema de la matriz básica correspondiente. En una submatriz de m_b filas por q columnas, compuesta por las q columnas no transmitidas de la matriz básica correspondiente, la diferencia entre los pesos de las columnas de dos columnas cualesquiera de la submatriz es menor o igual que 1. En el presente documento, el peso de una columna se refiere al número de elementos de la columna que indican una matriz cuadrada obtenida al desplazar cíclicamente una matriz de identidad. Alternativamente, la diferencia entre el número de elementos, cada uno de los cuales indica una matriz cuadrada que es ceros en dos columnas cualesquiera de la submatriz, es menor o igual que 1. Cada elemento que indica una matriz cuadrada que es ceros se denota como “-1” en esta realización. Cada elemento que indica una matriz cuadrada obtenida al desplazar cíclicamente una matriz de identidad se denota como un número entero que varía desde 0 hasta $Z-1$ en esta realización. En esta realización, $q=2$.

De acuerdo con el método de codificación o decodificación anterior, si el tamaño de bloque de código obtenido después de la segmentación del bloque de código es igual a 1024 bits, entonces el tamaño de bloque de código para la codificación de LDPC estructurada es $CBS = 1024$. El valor más pequeño que es mayor o igual que

CBS/kb=1024/8=256 se selecciona del conjunto de factores de expansión, como el factor de expansión de codificación para los códigos de LDPC estructurados. En este caso, se sabe que el factor de expansión es 256. La matriz de verificación básica se modifica de acuerdo con la Realización 5 para obtener una matriz de verificación básica de codificación que tiene un factor de expansión de 256 (los detalles de la misma se omitirán en el presente documento).

La secuencia de datos que se va a codificar y que tiene una longitud de 1024 bits se puede codificar en base al factor de expansión obtenido y la matriz de verificación básica de codificación. Cuando la tasa de codificación es 1/3, se puede obtener una palabra de código de LDPC con una longitud de 3072 bits. En un receptor, la secuencia de datos que se va a decodificar y que tiene una longitud de 3072 bits se puede decodificar en base al factor de expansión obtenido y la matriz de verificación básica de codificación, para obtener datos decodificados que tienen una longitud de 1024 bits. Los datos decodificados de todos los bloques de código se pueden combinar para obtener los datos originales del bloque de transporte transmitidos desde el transmisor.

Se puede utilizar la implementación de hardware del decodificador, como se describe en relación con la Realización 3, lo que es ventajoso al permitir que los códigos de LDPC admitan una longitud de código flexible y una tasa de código flexible.

Realización 7

Con la descripción de las realizaciones anteriores, será evidente para los expertos en la técnica que el método de acuerdo con las realizaciones anteriores se puede implementar por medio de software más una plataforma de hardware de propósito general necesaria. Por supuesto que se puede implementar en hardware, pero en muchos casos lo primero es la implementación óptima. En base a este entendimiento, la solución técnica de la presente divulgación en esencia, o partes de la misma que contribuyan a la técnica anterior, se puede incorporar en la forma de un producto de software. El producto de software informático se puede almacenar en un medio de almacenamiento (por ejemplo, ROM/RAM, disco magnético o disco óptico) e incluye instrucciones para hacer que un dispositivo terminal (que puede ser un teléfono móvil, un ordenador, un servidor o un dispositivo de red, etc.) realice el método descrito en las diversas realizaciones de la presente divulgación.

De acuerdo con una realización de la presente divulgación, también se proporciona un medio de almacenamiento. En esta realización, el medio de almacenamiento se puede configurar para almacenar códigos de programa para las siguientes etapas.

En S1, se obtiene un tamaño de bloque de código para la codificación de LDPC estructurada.

En S2, un factor de expansión de codificación z se determina en base a al menos uno del tamaño de bloque de código, un parámetro kb de una matriz de verificación básica, un valor entero positivo p o la matriz de verificación básica que tiene mb filas y nb columnas.

En S3, se codifica una secuencia de datos que se va a codificar, o se decodifica una secuencia de datos que se va a decodificar, en base a la matriz de verificación básica y el factor de expansión de codificación. En el presente documento, $kb=nb-mb$ y cada uno de kb , p , mb , z y nb es un número entero mayor que 1.

Opcionalmente, en esta realización, el medio de almacenamiento descrito anteriormente puede incluir, pero no se limita a, un disco USB, una Memoria de Solo Lectura (ROM), una Memoria de Acceso Aleatorio (RAM), un disco duro móvil, un disco magnético, un disco óptico u otros medios capaces de almacenar códigos de programa.

Opcionalmente, en esta realización, un procesador realiza las etapas del método de acuerdo con las realizaciones anteriores de acuerdo con los códigos de programa almacenados en el medio de almacenamiento.

Opcionalmente, para ejemplos específicos de esta realización, se puede hacer referencia a los ejemplos descritos en relación con las realizaciones anteriores y las realizaciones opcionales, y los detalles de los mismos se omitirán en el presente documento.

Los expertos en la técnica pueden apreciar que los módulos o etapas descritos anteriormente de la presente divulgación se pueden implementar por un dispositivo informático de propósito general, y se pueden centralizar en un solo dispositivo informático o distribuir a través de una red de múltiples dispositivos informáticos. Opcionalmente, se pueden implementar por medio de códigos de programa ejecutables por ordenador, que se pueden almacenar en un dispositivo de almacenamiento y ejecutar por uno o varios dispositivos informáticos. En algunos casos, las etapas mostradas o descritas en el presente documento se pueden realizar en un orden diferente al descrito anteriormente. Alternativamente, se pueden implementar por separado en módulos de circuitos integrados individuales, o uno o más de los módulos o etapas se pueden implementar en un solo módulo de circuito integrado. Por lo tanto, la presente divulgación no se limita a ningún hardware, software y combinación de los mismos en particular.

Lo anterior es solo ilustrativo de las realizaciones preferidas de la presente divulgación y no pretende limitar la presente divulgación. Varios cambios y modificaciones se pueden hacer por los expertos en la técnica. Cualquier modificación, alternativa equivalente o mejora que se realice sin apartarse del alcance de las reivindicaciones.

Aplicabilidad industrial

La presente divulgación se puede aplicar en el campo de las comunicaciones y es capaz de mejorar la flexibilidad en el procesamiento de datos con codificación de LDPC, resolviendo de esta manera el problema en la técnica relacionada asociada con la baja flexibilidad en el procesamiento de datos con codificación de LDPC.

REIVINDICACIONES

1. Un método para comunicaciones digitales, que comprende:
obtener (S202) un tamaño de bloque de código para una codificación de Verificación de Paridad de Baja Densidad Estructurada, LDPC;
- 5 determinar (S204) un factor de expansión de codificación z a partir de un conjunto de factores de expansión de codificación en base al tamaño de bloque de código y un parámetro k_b asociado con una matriz de verificación básica, en la que el conjunto de factores de expansión de codificación incluye un conjunto de números naturales; y
codificar o decodificar (S206) una secuencia de datos en base a la matriz de verificación básica y el factor de expansión de codificación z ,
- 10 en el que el parámetro k_b y el factor de expansión de codificación z son números enteros mayores que 1, en el que cada valor en el conjunto de factores de expansión de codificación es igual a un producto de una potencia entera positiva de dos y un número primo; y
en el que el número primo es igual a uno de: 3, 5, 7, 11 o 13.
- 15 2. El método de la reivindicación 1, en el que la matriz de verificación básica tiene m_b filas y n_b columnas, y en el que k_b es igual a n_b menos m_b .
3. El método de la reivindicación 1, en el que el factor de expansión de codificación z es uno de:
6, 12, 24, 48, 96, 192 o 384;
10, 20, 40, 80, 160 o 320; o
14, 28, 56, 112, o 224.
- 20 4. Un aparato para comunicación digital, que comprende:
un procesador (102) y
una memoria (104) que incluye el código ejecutable del procesador, en el que el código ejecutable del procesador luego de ejecución por el procesador, configura el procesador para:
- 25 obtener (S202) un tamaño de bloque de código para la codificación de Verificación de Paridad de Baja Densidad Estructurada, LDPC;
- 30 determinar (S204) un factor de expansión de codificación z a partir de un conjunto de factores de expansión de codificación en base al tamaño de bloque de código y un parámetro k_b asociado a una matriz de verificación básica, en la que el conjunto de factores de expansión de codificación incluye un conjunto de números naturales; y
codificar o decodificar (S206) una secuencia de datos en base a la matriz de verificación básica y el factor de expansión de codificación z ,
- 35 en el que el parámetro k_b y el factor de expansión de codificación z son números enteros mayores que 1, en el que cada valor en el conjunto de factores de expansión de codificación es igual a una potencia entera positiva de dos veces un número primo; y
en el que el número primo es igual a uno de: 3, 5, 7, 11 o 13.
- 40 5. El aparato de la reivindicación 4, en el que la matriz de verificación básica tiene m_b filas y n_b columnas, y en el que k_b es igual a n_b menos m_b .
6. El aparato de la reivindicación 4, en el que el factor de expansión de codificación z es uno de:
6, 12, 24, 48, 96, 192 o 384;
10, 20, 40, 80, 160 o 320; o
14, 28, 56, 112 o 224.
7. Un medio de almacenamiento no transitorio que tiene código almacenado en el mismo, el código luego de ejecución por un procesador, hace que el procesador implemente un método de cualquiera de las reivindicaciones 1 a 3.

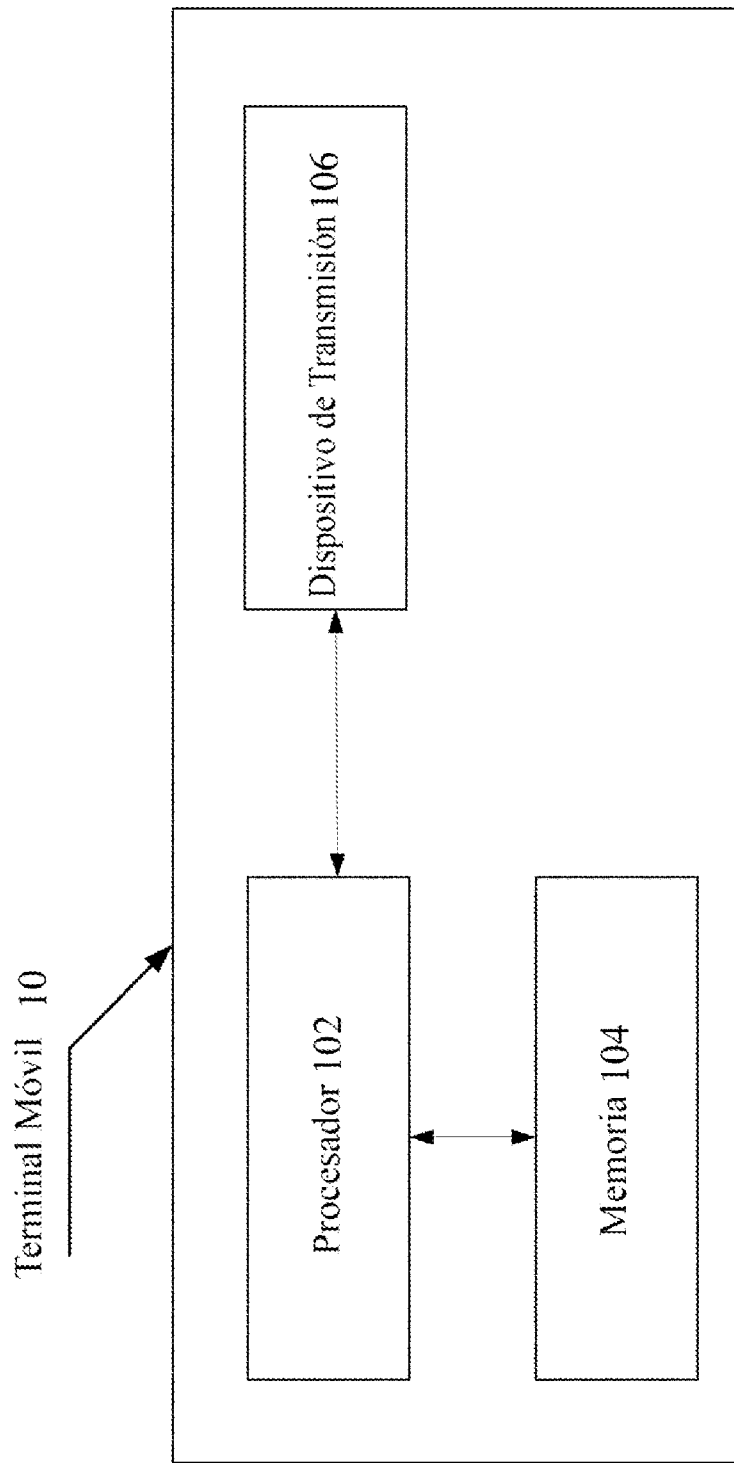


FIG. 1

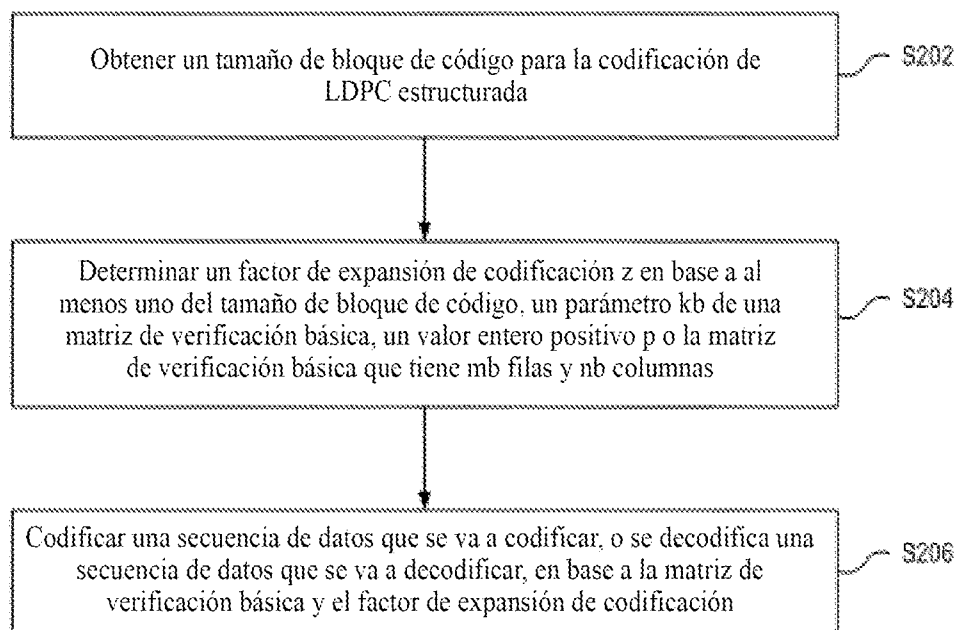


FIG. 2

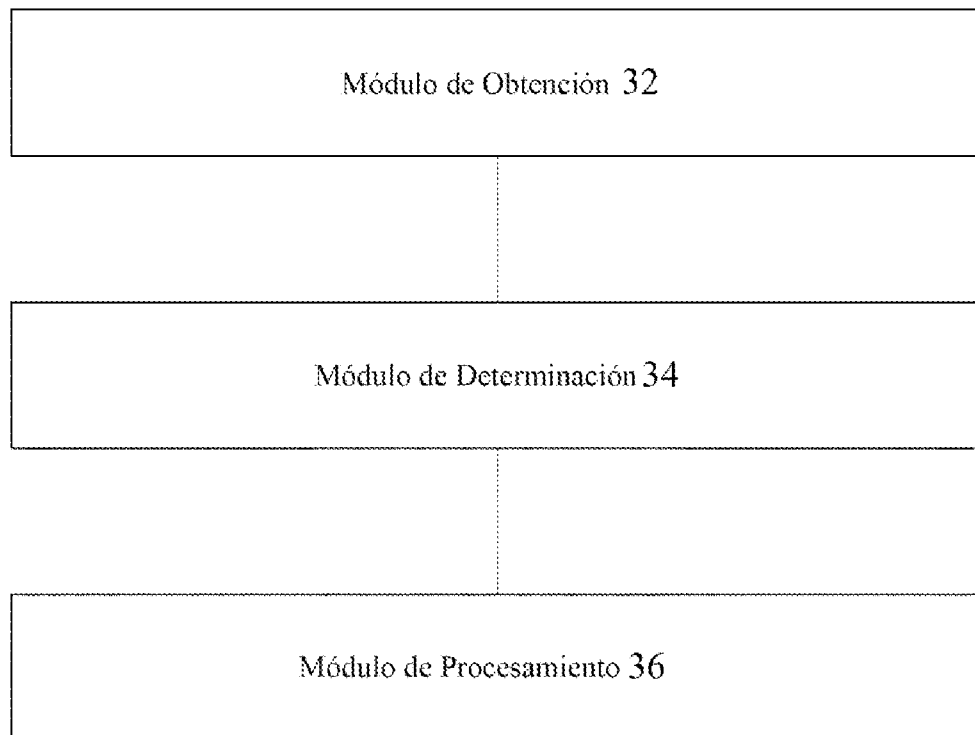


FIG. 3

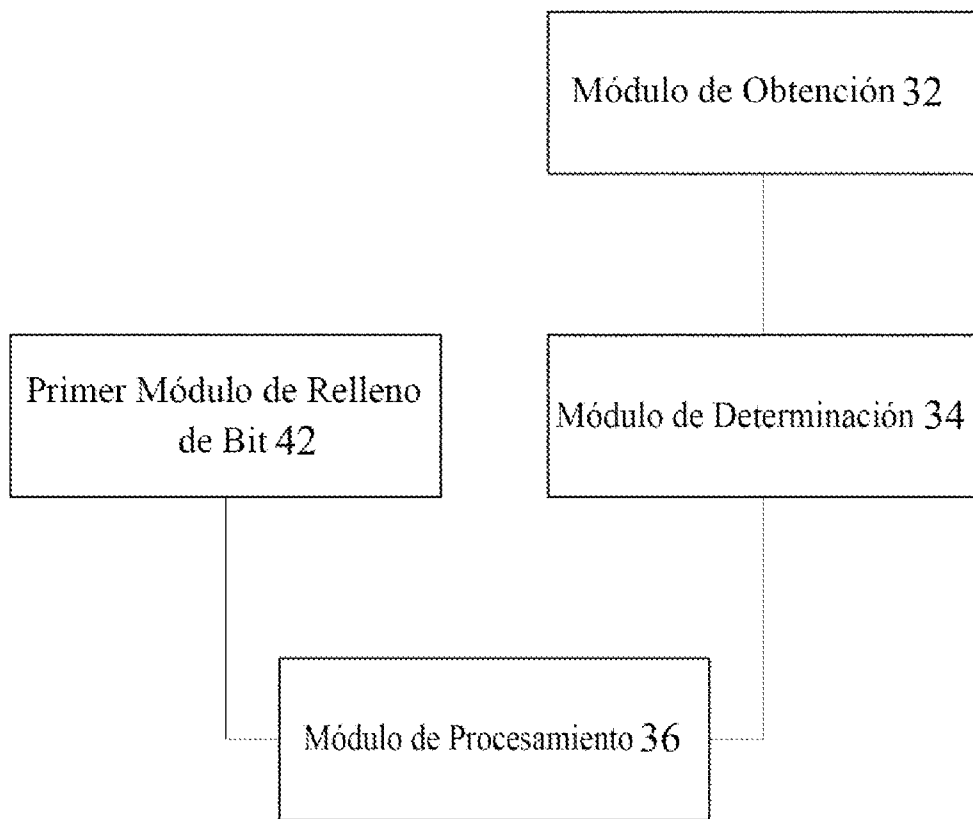


FIG. 4

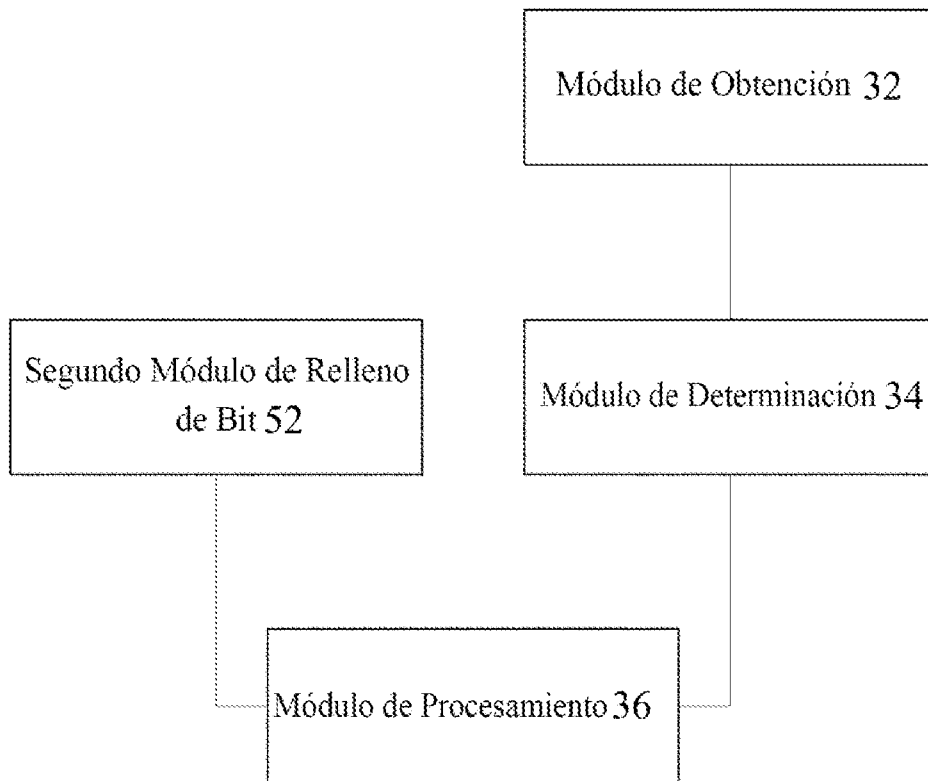


FIG. 5

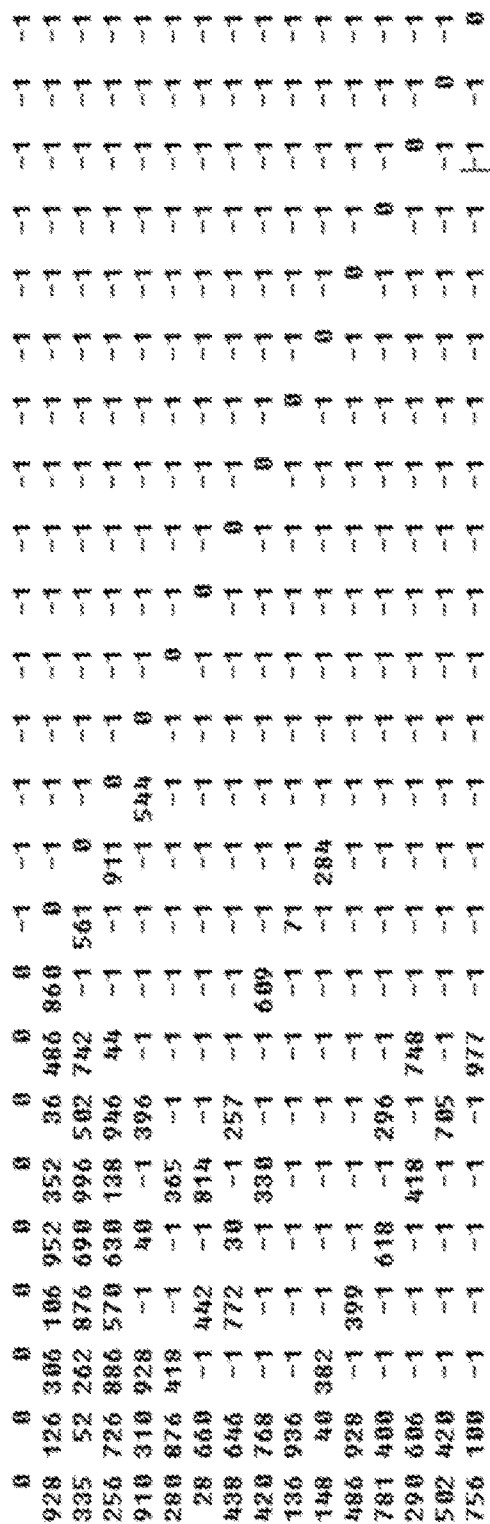


FIG. 6

0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	-1
464	63	153	53	476	176	18	243	430	0	-1	-1	-1	-1
167	26	131	438	345	498	251	371	-1	280	0	-1	-1	-1
128	363	443	285	315	69	473	22	-1	-1	455	0	-1	-1
455	155	464	-1	20	-1	198	-1	-1	-1	-1	272	0	-1
140	438	209	-1	-1	182	-1	-1	-1	-1	-1	-1	0	-1
14	330	-1	321	-1	407	-1	-1	-1	-1	-1	-1	-1	0
219	323	-1	386	15	-1	128	-1	-1	-1	-1	-1	-1	0

FIG. 7

0	0	0	0	0	0	0	0	0	-1	-1	-1	-1	-1
196	26	64	22	201	74	7	103	182	0	-1	-1	-1	-1
70	11	55	185	146	211	106	157	-1	118	0	-1	-1	-1
54	153	187	120	133	29	200	9	-1	-1	192	0	-1	-1
192	65	196	-1	8	-1	83	-1	-1	-1	-1	115	0	-1
59	185	98	-1	-1	77	-1	-1	-1	-1	-1	-1	0	-1
5	139	-1	93	-1	172	-1	-1	-1	-1	-1	-1	-1	0
92	136	-1	163	6	-1	54	-1	-1	-1	-1	-1	-1	0

FIG. 8

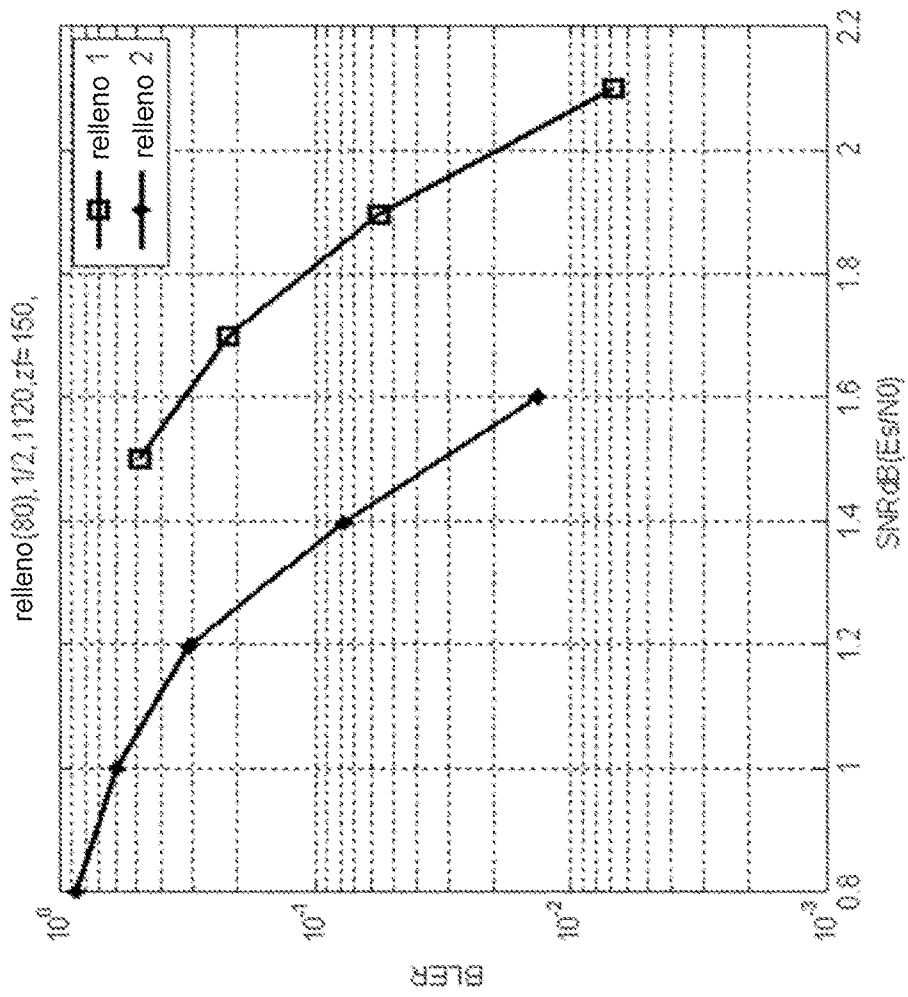


FIG. 9

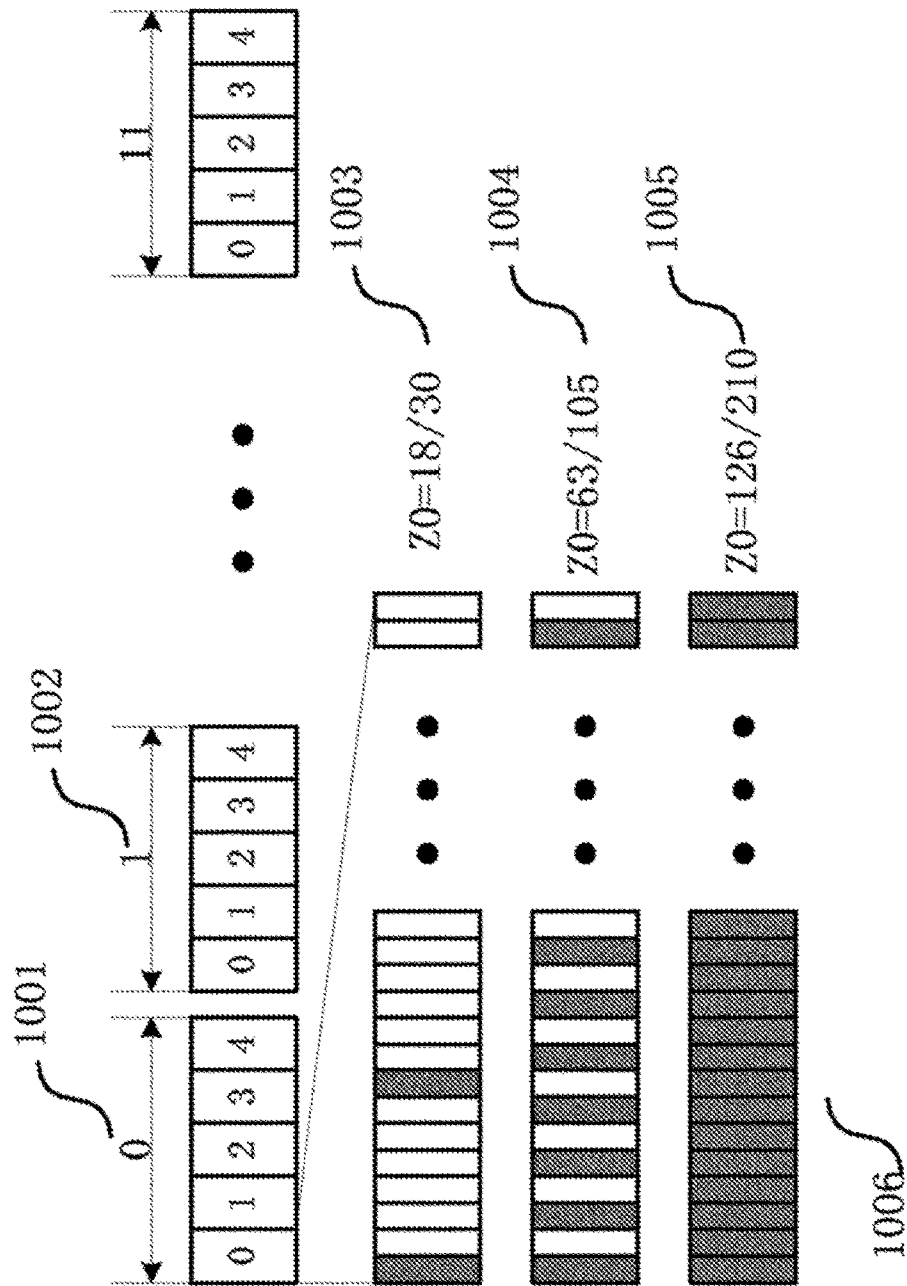


FIG. 10