



19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 364 558**

51 Int. Cl.:  
**H03M 13/11** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **05737553 .7**

96 Fecha de presentación : **19.04.2005**

97 Número de publicación de la solicitud: **1745551**

97 Fecha de publicación de la solicitud: **24.01.2007**

54 Título: **Método y aparato para codificar y decodificar datos.**

30 Prioridad: **06.05.2004 US 839995**

45 Fecha de publicación de la mención BOPI:  
**06.09.2011**

45 Fecha de la publicación del folleto de la patente:  
**06.09.2011**

73 Titular/es: **MOTOROLA MOBILITY, Inc.**  
**600 North US Highway 45**  
**Libertyville, Illinois 60048, US**

72 Inventor/es: **Classon, Brian K.;**  
**Blankenship, Yufei W. y**  
**Desai, Vipul A.**

74 Agente: **Elzaburu Márquez, Alberto**

ES 2 364 558 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 aparato para codificar y decodificar datos

**Campo de la invención**

5 La presente invención se refiere en general a la codificación y decodificación de datos y en particular, a un método y aparato para codificar y decodificar datos utilizando códigos de control de paridad de baja densidad (LDPC).

**Antecedentes de la invención**

10 Un código LDPC es un código de bloque lineal especificado por una matriz de control de paridad H. En general, un código LDPC se define sobre un campo Galois GF(q),  $q \geq 2$ . Si  $q = 2$ , el código es un código binario. Como con los códigos de bloque lineal, un bloque de k-bit de información  $S_{1 \times k}$  en general está codificado por la matriz generadora del código  $G_{k \times n}$  para convertir una contraseña de n-bit  $X_{1 \times n}$ , y la velocidad del código es  $r=k/n$ . La contraseña x es transmitida a través de un canal con ruido, y el vector de señal recibido y se pasa al decodificador para estimar el bloque de información  $S_{1 \times k}$ .

15 Dado un espacio n-dimensional, las hileras de G abarcan el subespacio C de la contraseña k-dimensional, y las hileras de la matriz de control de paridad  $H_{m \times n}$  abarcan el espacio dual m-dimensional  $C^\perp$ , donde  $m=n-k$ . Dado que  $x=sG$  y  $GH^T=0$ , entonces  $xH^T=0$  para todas las contraseñas en el subespacio C, donde "T" denota la trasposición de la matriz. En la discusión de los códigos LDPC, en general esto se escribe de la siguiente forma:

$$Hx^T = 0^T, \quad (1)$$

donde 0 es un vector de la hilera de todos ceros, y la contraseña  $x=[s \ p]=[s_0, s_1, \dots, s_{k-1}, p_0, p_1, \dots, p_{m-1}]$ ,

20 donde  $p_0, \dots, p_{m-1}$  son los bits de control de paridad; y  $s_0, \dots, s_{k-1}$  son los bits sistemáticos, igual a los bits de información dentro del bloque de información.

25 A fin de utilizar un código LDPC con buen comportamiento de corrección de errores, se debe definir una matriz H de control de paridad de baja densidad apropiada. Para la mayoría de los códigos LDPC irregulares, esto requiere la preparación de un gran parte de las columnas de H para que sean de peso-2 (es decir, dos unos y todos ceros en una columna) a fin de mantener la densidad total baja (es decir, la matriz general debe ser escasa). Este gran número de columnas de peso-2 puede permitir que altos pesos (por ej., 30) se asignen a algunas columnas mientras aún se mantiene el peso de la columna de bajo promedio en H. (Obsérvese que los pesos de la hilera usualmente están limitados en rango y son relativamente pequeños).

30 El diseño de una matriz de control de paridad con varios pesos de hileras y columnas es complicado cuando se considera el comportamiento de error. Por ejemplo, se puede construir una matriz con una serie de columnas generada al azar mientras se satisfacen las restricciones de peso de hilera y de peso de columna, sin embargo, con un gran porcentaje de columnas de peso-2 en la matriz, las columnas de peso-2 generadas al azar pueden contener fácilmente una mala estructura lo cual induce a un evento de error que no puede detectarse y una distancia mínima baja. En general, un evento de error no detectable de  $N_{ud}$  bits podría suceder si las columnas  $N_{ud}$  de de la matriz de control de paridad se suman (módulo 2) a la columna de todos ceros. La sumatoria de la columna de todos ceros se produce con mayor frecuencia cuando la matriz de control de paridad tiene un tamaño pequeño y contiene columnas de peso-2. El evento de error no detectable se enlaza directamente a la distancia mínima del código el cual es igual a  $\min(N_{ud})$ . Como resultado de ello, una matriz de control de paridad generada al azar puede tener una distancia mínima pequeña, la cual origina una elevada probabilidad de errores no detectables y un piso de error en relaciones de señal-a-ruido elevadas. Más aún, dado que los bits de código (elementos de x) asociados con las columnas de peso-2 son mucho más proclives a errores que los bits de código asociados con columnas de mayor peso, se espera que un gran porcentaje de errores de marco no detectables se involucre en las columnas de peso-2. Si bien hay varias guías de construcción de códigos en la técnica anterior que se mencionan o están implícitas en la literatura tales como (a) evitar los ciclos de longitud 4 y (b) evitar la superposición entre las columnas de peso-2 siempre que sea posible, estas guías no pueden ser suficientes para buenos códigos de comportamiento de errores. Por lo tanto, existe la necesidad de determinar la distribución de las columnas de peso-2 en la cuales la aparición de errores de marco no detectados se reduce a fin de mejorar significativamente el comportamiento del código en comparación con una matriz de control de paridad construida al azar.

35 No obstante el problema anterior, otra emisión de códigos LDPC es la elevada complejidad de codificación del método directo que utiliza la matriz generadora G correspondiente a la matriz H que define el código. Para un codificador LDPC sistemático, el bit de control de paridad  $p_i, i=0, \dots, m-1$ , en general se computa a partir de los

bits de información dados,  $s_0, \dots, s_{k-1}$ ,  $m=n-k$ . Siendo la contraseña  $\mathbf{x} = [\mathbf{s} \ \mathbf{p}] = [s_0, s_1, \dots, s_{k-1}, p_0, p_1, \dots, p_{m-1}]$ , los bits de control de paridad satisfacen las ecuaciones de control de paridad de (1). Un método de codificación convencional transforma la matriz de control de paridad H en una

forma sistemática  $\mathbf{H}_{\text{sys}} = [\mathbf{I}_{m \times m}, \mathbf{P}^T]$  a través de la eliminación gaussiana y el reordenamiento de la columna de modo que la matriz generadora del código correspondiente sea  $\mathbf{G} = [\mathbf{P}_{k \times m}, \mathbf{I}_{k \times k}]$ . Por lo tanto el codificador convencional calcula los bits de control de paridad utilizando  $[p_0, \dots, p_{m-1}] = [s_0, \dots, s_{k-1}] \mathbf{P}$

Este método directo puede originar problemas de implementación especialmente cuando el tamaño de la contraseña  $n$  es grande. Primero, si bien H tiene baja densidad (es decir con unas pocas 1's en la matriz y el número de 1's por hilera no crece con  $n$ ), P y por lo tanto C usualmente tienen alta densidad (es decir, muchas 1's en la matriz y el número de 1's por hilera aumenta a medida que  $n$  aumenta). La implementación del codificador convencional puede requerir una gran cantidad de memoria para almacenar las posiciones de las 1's en P. En segundo lugar, debido a la alta densidad de P, el número de adiciones binarias (sólo contando los términos cuando los elementos de P son "1") está en el orden de  $n^2$ ; lo que implica que la complejidad de codificación crece en forma cuadrática con  $n$ . Por lo tanto, existe la necesidad de un codificador eficiente para los códigos LDPC irregulares que toma ventaja de la estructura de un código LDPC irregular de buen comportamiento para minimizar el preprocesamiento y admitir un programa de codificación simple.

J00 PANYUH ET AL: "LDPC coding for OFDMA PHY" IEEE 802.16 BROADBAND WIRELESS ACCESS WORKING GROUP, CONTRIBUTED DOCUMENT, [Online] 1 May 2004 (2004-05-01), páginas 0-11, XP002438609 Recuperado de la Internet: URL: [http://www.ieee802.org/16/tgd/contrib/C80216d-04\\_86r1.pdf](http://www.ieee802.org/16/tgd/contrib/C80216d-04_86r1.pdf) [recuperado en 2007-06-21] divulga sugerencias con relación a la codificación LDPC para OFDMA PHY, que representan los procedimientos de codificación generales.

CUDAK M, CLASSON B. AND RHODES V.: "Revision of LDPC to accommodate HARQ" IEEE 802.16 BROADBAND WIRELESS ACCESS WORKING GROUP, CONTRIBUTED DOCUMENT, [Online] 28 de abril de 2004 (2004-04-28), páginas 0-9, XP002464006 Recuperado de la Internet: URL: [http://www.ieee802.org/16/tgd/contrib/C80216d-04\\_89.pdf](http://www.ieee802.org/16/tgd/contrib/C80216d-04_89.pdf) [recuperado el 1008-01-09] divulga una propuesta de texto para permitir características tales como los subcanales HARQ y AMC que operan con LDPC y otros modos de codificación.

### Síntesis de la invención

Los aspectos de la invención son los que se definen mediante las reivindicaciones:

#### Breve descripción de los dibujos

La Figura 1 es un diagrama de bloque de un codificador de acuerdo con la realización preferida de la presente invención.

La Figura 2 es un diagrama de bloque de un decodificador de acuerdo con la realización preferida de la presente invención.

La Figura 3 es un diagrama de flujo que muestra el funcionamiento del codificador de la figura 1 de acuerdo con la realización preferida de la presente invención.

La Figura 4 es un diagrama de flujo que muestra el funcionamiento del codificador de la figura 2 de acuerdo con la realización preferida de la presente invención.

#### Descripción detallada de los dibujos

Para tratar las necesidades antes mencionadas, en la presente se proporciona un método y aparato para codificar y decodificar datos. En particular, se proporcionan bits de control de paridad que se generan mediante una matriz de control de paridad H que reduce la aparición de errores de marco no detectados y mejora significativamente el comportamiento del código en comparación con la matriz de control de paridad construida al azar. H comprende una sección  $H_1$  no determinada y una sección  $H_2$  determinada, y donde  $H_2$  comprende un primera parte que comprende una columna  $h$  que tiene un peso impar superior a 2, y una segunda parte que comprende elementos de la matriz para la hilera  $i$ , columna  $j$  igual a 1 para  $i=j$ , 1 para  $i=j+1$ , 0 en otro lugar.

La presente invención abarca un método y un aparato según se define en las reivindicaciones adjuntas.

Volviendo ahora a los dibujos, donde números iguales designan componentes iguales, la Figura 1 es un diagrama de bloque del codificador 100 de acuerdo con una primera realización de la presente invención. Como se muestra, el codificador 100 comprende el microprocesador 101 y la tabla de búsqueda 103. En la primera realización de la

5 presente invención el microprocesador 101 comprende un procesador de señal digital (DSP), tal como, pero sin limitación MSC8100 y DSP56300 DSPs. Además, la tabla de búsqueda 103 sirve como medio de almacenamiento para almacenar una matriz, y comprende la memoria de solo lectura, sin embargo, un experto en la técnica reconocerá que se pueden utilizar también otras formas de memoria (por ej., memoria de acceso aleatorio, memoria de almacenamiento magnético, etc.). En una segunda realización, la funcionalidad del microprocesador 101 y la tabla de búsqueda 103 pueden incorporarse a un circuito integrado específico de la aplicación (ASIC) o a la matriz de puertas programables in situ (FPGA). En particular, la tabla de búsqueda 103 puede ser implementada en forma de memoria correspondiente a la existencia o no de trayectorias de señal en un circuito.

10 Como se mencionó anteriormente, los datos codificados en general se producen en forma de una pluralidad de bits de control de paridad además de los bits sistemáticos, donde juntos los bits de control de paridad y los sistemáticos son una solución al  $Hx^T = O^T$ . En la primera realización de la presente invención una matriz de control de paridad H se almacena en la tabla de búsqueda 103, y se accede mediante el microprocesador 101 para resolver la ecuación anterior. En particular, el microprocesador 101 determina los valores apropiados para los bits de paridad de control ( $P_0, \dots, P_{m-1}$ ) sobre la base del conjunto de símbolos corrientes  $s = (s_0, \dots, s_{k-1})$  y la matriz de control de paridad H.

15 Los bits de control de paridad y el conjunto de símbolos se pasan a un transmisor que transmite a un receptor. H comprende una sección  $H_1$  no determinante (es decir, el valor de cada entrada no se define o se determina para que tenga una estructura particular, y por lo tanto puede ser al azar en la medida que ciertas restricciones de diseño se cumplan) y una sección determinante  $H_2$ , (es decir, el valor de cada entrada se define o se determina como para que tenga una estructura particular).

20 La sección determinante  $H_2$  además está compuesta de dos partes. En particular, una primera parte comprende una primera columna h que tiene un peso impar superior a 2, la segunda parte  $H'_2$  comprende elementos de matriz que son iguales a 1 para  $i=j$  y a 1 para  $i=j+1$ , y 0 en otras partes, para la hilera i, la columna j de  $H'_2$ ,  $0 \leq i \leq m-1$ ,  $0 \leq j \leq m-2$ . De este modo, la matriz de control de paridad H general puede estar representada por

$$H = [H_1 \ H_2], \quad (2)$$

25 y la sección determinante  $H_2$  puede estar representada por

$$H_2 = \left[ \begin{array}{c|cccccccc} h_0 & 1 & & & & & & & \\ h_1 & 1 & 1 & & & & & & \\ \cdot & & 1 & 1 & & & & & 0 \\ \cdot & & & 1 & 1 & & & & \\ \cdot & & & & 1 & 1 & & & \\ \cdot & & & & & 1 & 1 & & \\ \cdot & & 0 & & & & 1 & 1 & \\ \cdot & & & & & & & 1 & 1 & \\ \cdot & & & & & & & & \ddots & \ddots \\ \cdot & & & & & & & & & 1 & 1 \\ \cdot & & & & & & & & & & 1 & 1 \\ h_{m-1} & & & & & & & & & & & 1 \end{array} \right]$$

$$(3)$$

5 En la realización preferida de la presente invención  $H_1$  puede ser cualquier matriz binaria de tamaño  $m$ -by- $k$ . En otras palabras,  $H_1$  puede ser regular o irregular o tener cualquier distribución de peso deseada). Mientras  $H_2$  no es menos triangular, tiene una estructura simple. La primera columna de  $H_2$  (indicada mediante  $h$ ) tiene un número impar de 1's (3 o más), y el resto de las columnas (indicadas mediante la matriz  $m$ -by- $(m-1)$   $H'_2$ ) son de peso-2 con un máximo de 1 superposición entre sí. (El primo en  $H_2$  indica una sub-matriz, no una trasposición).

10 Si bien la sub-matriz determinante  $H_2$  siempre tiene columnas  $(m-1)$  de peso-2, no se espera que afecte el comportamiento del código, dado que no hay ciclos entre las columnas de peso-2, y las distribuciones de peso óptimas informadas en la literatura incluyen en general un gran porcentaje de columnas de peso-2. Se espera que la sub-matriz  $H_1$  y la columna  $h$  sean diseñadas de modo que la matriz  $H$  completa tenga un comportamiento de corrección de errores bueno. Por ejemplo,  $H_1$  está compuesta típicamente de columnas con un peso superior a 2.

15 A diferencia de la técnica anterior, la matriz  $H_2$  se elige para evitar cualquier columna de peso-1. Una columna de peso-1 puede degradar el comportamiento dado que un bit correspondiente a una columna de peso-1 no actualiza la información liviana durante la decodificación iterativa. La sub-matriz  $H_2$  determinante está diseñada para corresponderse con los bits de control de paridad en lugar de los bits sistemáticos, de modo que los bits sistemáticos estén mejor protegidos debido a los pesos de de la columna más altos en  $H_1$ .

**Método de codificación**

La estructura  $H_2$  que se ejemplifica anteriormente hace que la complejidad de codificación sea lineal, en vez de cuadrática, en el tamaño del código. La codificación puede obtenerse sin hallar ni computar una matriz generadora.

20 Dado cualquier bloque de bits de información  $s$ , los bits de control de paridad  $m$  se pueden hallar resolviendo las ecuaciones que se definen por

$$\left[ \begin{matrix} (H_1)_{m \times k} & h_{m \times 1} & (H'_2)_{m \times (m-1)} \end{matrix} \right] \begin{bmatrix} s^T \\ p^T \end{bmatrix} = 0. \quad (4)$$

Debido al peso de  $h$  de la columna impar y las columnas de  $m-1$  de peso-2 de  $H'_2$ , la sumatoria de toda la ecuación en (4) da

$$p_0 = \left( \sum_{\text{row}} H_1 \right) s^T, \quad (5)$$

25 donde  $\sum_{\text{row}} H_1$  denota el vector de la hilera después de sumar todas las filas de  $H_1$ . Obsérvese que la suma puede realizarse en forma equivalente sobre el vector de la columna intermedia  $v = [v_0, v_1, \dots, v_{m-1}]^T = H_1 s^T$ , pero la realización de la suma sobre las hileras de  $H_1$  se puede llevar a cabo de antemano y da por resultado menor cantidad de operaciones para computar  $p_0$ .

30 Una vez determinado  $p_0$ , el resto de los bits de control de paridad,  $p_i$  a  $p_{m-1}$  se pueden hallar en forma recursiva. Por ejemplo,

$$\begin{aligned} p_1 &= h_0 p_0 + v_0, \\ p_2 &= h_1 p_0 + p_1 + v_1, \\ p_3 &= h_2 p_0 + p_2 + v_2, \quad \dots, \quad (6) \\ &\dots \\ p_{m-1} &= h_{m-2} p_0 + p_{m-2} + v_{m-2}, \end{aligned}$$

35 donde  $h = [h_0, h_1, \dots, h_{m-2}, h_{m-1}]^T$  es la columna  $h$  en (3). Dado que todas las variables son binarias en (6), la complejidad de codificación es muy baja. Además, para simplificar la implementación, se puede almacenar un vector  $w = [w_0, w_1, \dots, w_{m-2}, w_{m-1}] = [h_0 p_0 + v_0, h_1 p_0 + v_1, \dots, h_{m-2} p_0 + v_{m-2}, h_{m-1} p_0 + v_{m-1}]$  de modo que los bits de control de paridad se hallen mediante  $p_1 = w_0, p_i = p_{i-1} + w_{i-1}, i=2, 3, \dots, m-1$ .

Las ecuaciones de control de paridad además pueden resolverse en forma recursiva a partir de  $p_{m-1}$  a  $p_i$ . En este

caso, los bits de control de paridad se hallan mediante  $p_{m-1}=w_{m-1}, p_i=p_{i+1} + w_i, i=m-2, \dots, 2,1.$

La estructura adicional en h se puede utilizar para reducir además la complejidad de codificación. Por ejemplo, en (3) la mayoría de las entradas de h pueden ser los ceros posteriores (por ej.,  $h=[1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ \dots\ 0]^T$ ). Saber que todos los valores de  $h_i$  son 0 para  $z \leq i \leq m-1$  puede simplificar el paso de determinar los bits de control de paridad en forma recursiva, como en (6), permitiendo variantes de la estructura de codificación recursiva. En un ejemplo,  $w_i=v_i$  para  $z \leq i \leq m-1$ , de este modo  $p_i, z \leq i \leq m-1$ , se puede hallar antes de hallar  $p_0$ . Entonces  $p_0$  a  $p_{z-1}$  se pueden hallar sobre la base de  $s$  y  $p_z$ .

**Método de decodificación**

La Figura 2 es un diagrama de bloque del decodificador 200 de acuerdo con una realización de la presente invención. Como se muestra, el decodificador 200 comprende el microprocesador 201 y la tabla de búsqueda 203. En una primera realización de la presente invención el microprocesador 201 comprende un procesador de señal digital (DSP), tal como, pero sin limitación MSC8100 y DSP56300 DSPs. Además, la tabla de búsqueda 203 actúa como medio de almacenamiento para almacenar la matriz H, y comprende la memoria de sólo lectura. Sin embargo, un experto en la técnica reconocerá que también se pueden utilizar otras formas de memoria (por ej., memoria de acceso aleatorio, memoria de almacenamiento magnético, etc.). En una segunda realización, la funcionalidad del microprocesador 201 y la tabla de búsqueda 203 pueden incorporarse a un circuito integrado específico de la aplicación (ASIC) o a la matriz de puertas programables in situ (FPGA). En particular, la tabla de búsqueda 203 puede ser implementada en forma de memoria correspondiente a la existencia o no de trayectorias de señal en un circuito.

El vector de señal recibido (recibido mediante un receptor)  $y=(y_0 \dots y_{n-1})$  corresponde a la contraseña x transmitida a través de un canal de ruido, donde los datos x codificados, como se mencionó anteriormente, son una solución a  $Hx^T = 0^1$ . En la primera realización de la presente invención una matriz H de control de paridad se almacena en la tabla de búsqueda 203, y se accede mediante el microprocesador 201 para decodificar y y estimar el bloque de información s (es decir, el conjunto de símbolos corrientes ( $s_0, \dots, s_{k-1}$ ). En particular, el microprocesador 201 estima el conjunto de símbolos corrientes ( $s_0, \dots, s_{k-1}$ ) sobre la base del vector de señal recibido y la matriz H de control de paridad. Como se mencionó anteriormente, H comprende una sección no determinante  $H_1$  (es decir, el valor de cada entrada no se define ni se determina que tenga una estructura particular, y por lo tanto puede ser aleatoria en la medida que ciertas restricciones de diseño se cumplan) y una sección determinante  $H_2$  (es decir, el valor de cada entrada se define o se determina que tenga una estructura particular).

Como es bien conocido en la técnica, existen muchas formas en que el decodificador 200 puede utilizar la matriz H de control de paridad en el microprocesador 201 para decodificación. Una de estas formas es realizar una multiplicación vector-matriz con H para determinar un patrón de probabilidad de error. Otra forma consiste en utilizar H para construir un gráfico bipartito donde los extremos en el gráfico corresponden a las l's en H, y para procesar iterativamente y sobre el gráfico bipartito.

**Modificación del código**

A menudo se desea crear un nuevo código (un matriz de control de paridad nueva o diferente) modificando un código existente (una matriz de control de paridad existente). El nuevo código típicamente tiene una cantidad diferente de bits de información o control de paridad que el código existente. Si las modificaciones son apropiadas, el nuevo código puede ser codificado y decodificado mediante un procedimiento similar al código original. Las modificaciones para agregar o eliminar los bits de control de paridad son especialmente difíciles para los códigos LDPC porque el código resultante puede no tener buenas propiedades para codificar o decodificar incluso si el código original tenía buenas propiedades. Las siguientes subsecciones describen dos formas de modificar los códigos que mantienen las buenas propiedades del código.

**Modificación simple**

Con la estructura  $H_2$  que se describió anteriormente, la matriz H puede ser expandida o reducida mediante un número arbitrario de bits de control de paridad sin cambiar la estructura de  $H_2$ . Cada código expandido o reducido utiliza el mismo procedimiento de codificación que se ilustró anteriormente, y mantiene el buen comportamiento cuando su sub-matriz  $H_1$  es construida cuidadosamente.

Por ejemplo, si los bits de control de paridad  $g$  deben ser perforados para obtener un código de velocidad más alta ( $n-g, k$ ), la matriz H es modificada eliminando las columnas más hacia la derecha  $g$  y las hileras  $g$  más bajas. En forma alternativa, la matriz H puede ser extendida por los bits de control de paridad  $g$  para obtener un código de velocidad más bajo ( $n+g, k$ ) agregando columnas  $g$  a la derecha e hileras  $g$  en la parte inferior, manteniendo la estructura de la columna de peso-2 diagonal en  $H_2$ .

Para evitar tener una primera columna no deseable (por ej., peso inferior a 3, o, incluso peso) en  $H_2$ , cuando se perfora, se prefiere la estructura, donde las l's de h están distribuidas hacia la parte inferior. Por ejemplo, en (3) con  $h=[1\ 0\ 0\ 1\ 0\ 0\ 1\ 0\ \dots\ 0]^T$ , sólo 3 entradas de ha son 1 y están localizadas hacia la parte superior de h.

5 Mientras la modificación simple retiene la facilidad de codificar el código original, y promueve una reducción en la memoria para almacenar las matrices, los códigos creados por la modificación simple no se relacionan en el sentido de que los bits de control de paridad del código de velocidad más alta son un subconjunto de los bits de control de paridad del código de velocidad más baja. La relación donde los bits de control de paridad del código de velocidad más alto son un subconjunto de los bits de control de paridad del código de velocidad más baja dado el mismo conjunto de símbolos de información se denomina compatibilidad de velocidad.

10 **Modificación del código compatible con la velocidad**

La matriz H que se definió anteriormente puede ser modificada para crear códigos LDPC compatibles con la velocidad concatenando las porciones similares a  $H_2$ . Por ejemplo, considérese tres códigos, donde el superíndice indica el código 1, el código 2 y el código 3, respectivamente. Los bits de control de paridad del código 1 son un subconjunto de los bits de control de paridad del código 2 (es decir, los bits de control de paridad del código 1 se anidan dentro de los bits de control de paridad del código 2), y los bits de control de paridad del código 2 son un subconjunto de los bits de control de paridad del código 3 (es decir, los bits de control de paridad del código 2 se anidan dentro de los bits de control de paridad del código 3), y a su vez los bits de paridad del código 1 se anidan dentro de los bits de paridad del código 3). El código 1 tiene una matriz de control de paridad  $H^{(1)}$  que se define mediante

20 
$$\mathbf{H}^{(1)} = \begin{bmatrix} \mathbf{H}_1^{(1)} & \mathbf{H}_2^{(1)} \end{bmatrix}, \quad (7)$$

donde  $H^{(1)}$  es una matriz  $m_1$ -by- $k$  y  $H_2^{(1)}$  es una matriz  $m_1$ -by- $m_1$  cuya estructura puede seguir (3), y produce los bits de control de paridad  $P_0, P_1, \dots, P_{m_1-1}$ . Obsérvese que (7) es equivalente a (2), y los superíndices "(1)" sólo se utilizan para enfatizar el primer código (es decir,  $H^{(1)} = H$ ,  $m_1=m$ ).

El código 2 tiene una matriz de control de paridad  $H^{(2)}$  que se define mediante

25 
$$\mathbf{H}^{(2)} = \left[ \begin{array}{cc|c} \mathbf{H}_1^{(1)} & \mathbf{H}_2^{(1)} & \mathbf{0} \\ \hline \mathbf{H}_1^{(2)} & \mathbf{H}_2^{(2)} & \mathbf{H}_2^{(2)} \end{array} \right], \quad (8)$$

30 donde  $H_1^{(2)}$  es una matriz  $(m_2-m_1)$ -by- $(k+m_1)$ ,  $H_2^{(2)}$  es una matriz  $(m_2-m_1)$ -by- $(m_2-m_1)$  cuya estructura puede seguir a la establecida en la ecuación (3), y el código 2 produce los bits de control de paridad  $P_0, P_1, \dots, P_{m_2-1}$ . Como sigue a la estructura establecida en (3), la sección determinante  $H^{(2)}$  tiene una primera columna  $h^{(2)}$  que tiene un peso impar superior a 2, y una segunda parte que comprende un 1 para  $i=j$  y un 1 para  $i=j+1$ , y 0 en todas partes, para la hilera  $i$ , columna  $j$ ,  $0 \leq j < m-1$ ,  $0 \leq i < m-2$ . La primera columna  $H^{(2)}$  no tiene que en el mismo tamaño o composición que  $h^{(1)}$ , la cual es la columna h del código 1. Debido a la estructura de anidación, los bits de control de paridad  $P_{m_1}, \dots, P_{m_2-1}$  se pueden obtener a partir de  $H^{(2)}$ ,  $H_2^{(2)}$ , los bits sistemáticos s, y los bits de control de paridad previamente computados  $P_0, P_1, \dots, P_{m_1-1}$  utilizando las técnicas de codificación de las ecuaciones (5) y (6).

35 El código 3 tiene una matriz de control de paridad  $H^{(3)}$  que se define mediante

$$\mathbf{H}^{(3)} = \left[ \begin{array}{ccc|c} \mathbf{H}_1^{(1)} & \mathbf{H}_2^{(1)} & \mathbf{0} & \mathbf{0} \\ \hline \mathbf{H}_1^{(2)} & \mathbf{H}_2^{(2)} & \mathbf{H}_2^{(2)} & \mathbf{0} \\ \hline \mathbf{H}_1^{(3)} & \mathbf{H}_2^{(3)} & \mathbf{H}_2^{(3)} & \mathbf{H}_2^{(3)} \end{array} \right], \quad (9)$$

40 donde  $H_1^{(3)}$  es una matriz  $(m_3-m_2)$ -by- $(k+m_2)$ ,  $H_2^{(3)}$  es una matriz  $(m_3-m_2)$ -by- $(m_3-m_2)$  cuya estructura puede seguir la ecuación (3), y el código 3 produce bits de control de paridad  $P_0, P_1, \dots, P_{m_3-1}$ . Debido a la estructura de anidación, los bits de control de paridad  $P_{m_1}, \dots, P_{m_2-1}$  se pueden obtener a partir de  $H_1^{(3)}$ ,  $H_2^{(3)}$  los bits

5 sistemáticos  $s$ , y los bits de control de paridad previamente computados  $p_0, p_1, \dots, p_{m_2-1}$  utilizando las técnicas de codificación de ecuaciones (5) y (6). Debe observarse que, al igual que  $H$ , tanto  $H^{(2)}$  como  $H^{(3)}$  comprenden una sección determinante que tiene una primera columna con un peso impar superior a 2, y todas las columnas restantes comprenden un 1 para  $i=j$  y a 1 para  $i=j+1$ , y 0 en otras partes, para la hilera  $i$ , la columna  $j$  de  $H^{(2)}$ ,  $0 \leq i \leq m_2-1$ ,  $0 \leq j \leq m_2-2$ , como se ilustra en (3).

El "código madre" que se define en la ecuación (9) define efectivamente los códigos de tres velocidades diferentes:  $k/(k+m_1)$ ,  $k/(k+m_2)$ , and  $k/(k+m_3)$ .

10 Las tres piezas  $([s, p_0, p_1, \dots, p_{m_1-1}], [s, p_0, p_1, \dots, p_{m_2-1}], \text{ and } [s, p_0, p_1, \dots, p_{m_3-1}])$  se utilizan posteriormente para diferentes velocidades del código de transmisión /recepción, y dado que son compatibles con la velocidad, se pueden utilizar para redundancia de incremento (IR). En IR, una primera transmisión puede enviar  $[s, p_0, p_1, \dots, p_{m_1-1}]$ , una segunda transmisión puede enviar  $[p_{m_1}, \dots, p_{m_2-1}]$  y una tercera

15 transmisión puede enviar  $[p_{m_2}, \dots, p_{m_3-1}]$ . Las transmisiones más allá de la primera pueden ser enviadas si fallaron las transmisiones anteriores (no se recibió o se decodificó en forma incorrecta), o si el protocolo especifica múltiples transmisiones. Si bien la discusión anterior asume la transmisión de bits de paridad adicionales del código 2, el código 3, ..., etc., los expertos en la técnica entienden que son posibles otros modos de transmisión. Por ejemplo, cuando falla la primera transmisión, la segunda transmisión puede enviar bits de paridad adicionales tanto del código 2 como del código 3,  $[p_{m_1}, \dots, p_{m_3-1}]$ .

A continuación, se presenta un ejemplo para mostrar de qué manera los nuevos bits de control de paridad para el código 2 se pueden generar a partir de la información y de los bits de control de paridad del código 1. Para ilustrar

20 esta propiedad, se define la matriz  $k=9, m_1=9, m_2=13, \text{ the } m_1\text{-by-}(k+m_1)$

$$\mathbf{H}^{(1)} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix} \\ = [\mathbf{H}_1^{(1)} \quad \mathbf{H}_2^{(1)}]$$

y la matriz  $m_2\text{-by-}(k+m_2)$ .



$$p_{11} = \text{row}_1(\mathbf{H}_1^{(2)}) \begin{bmatrix} \mathbf{s}^T \\ p_0 \\ \vdots \\ p_{m_1-1} \end{bmatrix} + p_{10} = s_5 + p_2 + p_7 + p_{10},$$

$$p_{12} = \text{row}_2(\mathbf{H}_1^{(2)}) \begin{bmatrix} \mathbf{s}^T \\ p_0 \\ \vdots \\ p_{m_1-1} \end{bmatrix} + p_{11} + p_9 = s_6 + p_1 + p_3 + p_6 + p_{11} + p_9,$$

donde la hilera de función (H) selecciona la hilera *i*-th de la matriz H. Si bien el método preferido se presentó anteriormente para hallar  $p_{m_1}$ , otros métodos también son posibles. Por ejemplo, además se puede hallar por

$$p_{m_1} = \left( \sum_{\text{row}} \begin{bmatrix} \mathbf{H}_1^{(1)} & \mathbf{H}_2^{(1)} \\ & \mathbf{H}_1^{(2)} \end{bmatrix} \right) \begin{bmatrix} \mathbf{s}^T \\ p_0 \\ \vdots \\ p_{m_1-1} \end{bmatrix}.$$

5 Como se mostró anteriormente, una vez que los primeros bits de control de paridad para la matriz H del primer código se transmiten, y si se ha determinado que la primera transmisión ha fallado, se pueden computar y transmitir bits de control de paridad adicionales  $(p_m, \dots, p_{m_2-1})$  sobre la base de un conjunto de símbolos corrientes  $\mathbf{s}=(s_0, \dots, s_{k-1})$  y los primeros bits de control de paridad  $(p_0, \dots, p_{m-1})$ . La matriz  $H^{(2)}$  se utiliza para determinar los bits de control de paridad adicionales donde  $H^{(2)}$  se basa en H como se muestra en la ecuación (8). Este proceso puede continuar indefinidamente. De este modo durante la operación, el microprocesador 101 explota la estructura de H para computar los bits de control de paridad *m* sobre la base de las ecuaciones (5) y (6). A medida que se necesitan bits de control de paridad adicionales, se computan utilizando matrices que se definieron anteriormente con referencia a las ecuaciones (8) y (9). Nuevamente, recuérdese que  $H^{(1)} = H$  y  $m_1 = m$ .

15 La Figura 3 es un diagrama de flujo que muestra el funcionamiento del codificador 100, y en particular, el procesador 101. El flujo lógico comienza en el paso 301 donde se recibe un conjunto de símbolos corrientes  $(s_0, \dots, s_{k-1})$ . En el paso 303, los valores para los primeros bits de control de paridad se determinan sobre la base de H. En particular, los primeros bits de control de paridad  $(p_0, \dots, p_{m-1})$  se determinan como se describió anteriormente en las ecuaciones (5) y (6). En el paso 305 se transmiten el conjunto de símbolos corrientes y los primeros bits de control de paridad. En el paso 307 se realiza una determinación respecto a si es necesario transmitir bits de control de paridad adicionales. Esta determinación puede hacerse, por ejemplo, si falló la primera transmisión (no se recibió o se decodificó incorrectamente), o si el protocolo de transmisión siempre especifica una segunda transmisión. Si, en el paso 307, se determina que se deben transmitir bits de paridad de control adicionales, entonces el flujo lógico continúa en el paso 309 donde se determinan los bits de control de paridad adicionales sobre la base de una  $H^{(2)}$ ,  $H^{(3)}$ , ..., etc. apropiada. La elección del envío de bits de control de paridad de  $H^{(2)}$ ,  $H^{(3)}$ , ..., etc. se puede basar en, por ejemplo, el protocolo de transmisión, la métrica de calidad, etc. En el paso 311, se transmiten los bits de control de paridad adicionales y el flujo lógico vuelve al paso 307. Si, en el paso 307 se determina que no es necesario transmitir bits de control de paridad adicionales, entonces el flujo lógico finaliza en el paso 313.

30 La Figura 4 es un diagrama de flujo que muestra el funcionamiento del decodificador 200, y en particular, el microprocesador 201. El flujo lógico comienza en el paso 401 donde se recibe el vector de señal recibido  $\mathbf{y}=(y_0, \dots, y_{n-1})$ . En el paso 403, los estimados del bloque de información *s* (es decir el conjunto de símbolos corrientes  $(s_0, \dots, s_{k-1})$ ) se determinan sobre la base de H. Como es bien conocido en la técnica, existen muchas formas en que el decodificador 200 puede utilizar la matriz H de control de paridad en el microprocesador 201 para decodificación. Una de estas formas es realizar una multiplicación vector-matriz con H para determinar un patrón de

probabilidad de error. Otra forma consiste en utilizar  $H$  para construir un gráfico donde los extremos en el gráfico corresponden a las  $l$ 's de  $H$ , y para procesar iterativamente y sobre el gráfico. En el paso 407, se realiza la determinación respecto a si se recibirán los bits de control de paridad, con los bits de control de paridad adicionales recibidos en la forma de un vector de señal recibido adicional de los bits de control de paridad adicional pasados a través de un canal de ruido. Esta determinación puede hacerse, por ejemplo, si falló la primera transmisión (no se recibió o no se decodificó correctamente), o si el protocolo de transmisión siempre especifica una transmisión adicional. Si, en el paso 407 se determina que los bits de control de paridad adicionales se recibirán, entonces el flujo lógico continúa al paso 409 donde los estimados del bloque de información  $s$  (es decir, el conjunto de símbolos corrientes  $(s_0, \dots, s_{k-i})$ ) se determinan sobre la base de una  $H^{(2)}$ ,  $H(3)$ , ..., etc. apropiada y los bits de control de paridad adicionales recibidos. El flujo lógico vuelve a continuación al paso 407. Si, en el paso 407 se determina que no se recibirán bits de control de paridad adicionales, entonces el flujo lógico finaliza en el paso 413.

Mientras que la invención se ha mostrado y descrito en particular con referencia a una realización particular, los expertos en la técnica entenderán que se pueden realizar varios cambios en forma y detalles sin apartarse del espíritu y el alcance de la invención. Por ejemplo, mientras que la invención se mostró con el ordenamiento de  $s_i$  y  $p_i$  dentro de la definición de  $x$ , un experto en la técnica reconocerá que se puede producir otro ordenamiento de los bits dentro de  $x$  dado que los bits de contraseña se pueden recolectar en cualquier orden en la medida que las columnas de  $H$  se reordenen en consecuencia. Además, mientras que la descripción anterior ha sido mostrada y descrita en particular con referencia a los códigos binarios (es decir, los códigos que se definen en el Campo Galois ( $GF(2)$ )), un experto en la técnica reconocerá que también se puede utilizar un GF arbitrario. Si bien los ejemplos que se presentaron anteriormente se muestran en un formato, son posibles otros formatos que permiten un procedimiento de modificación de código y codificación similares. Por ejemplo, la columna  $h$  puede estar localizada en cualquier columna de  $H_2$ , no sólo la primera columna. En otro ejemplo, los bits de  $x$  pueden permutarse, requiriendo la permuta de las columnas de  $H$ . Incluso en otro ejemplo, las hileras de  $H$  pueden ser permutadas, sin afectar el valor de los bits de control de paridad. La matriz  $H$  además se puede utilizar en cualquier tipo de decodificador que confía en una matriz de control de paridad. Se intenta que estos cambios estén dentro del alcance de las siguientes reivindicaciones.

REIVINDICACIONES

1. Un método para operar un transmisor que genera bits de control de paridad sobre la base de un símbolo corriente  $\mathbf{s}=(s_0, \dots, s_{k-1})$ , comprendiendo el método los pasos de:

recibir (301) el conjunto de símbolos corrientes  $\mathbf{s}=(s_0, \dots, s_{k-1})$ ;

5 utilizar una matriz H para determinar (303) los bits de control de paridad, donde H comprende una sección no determinante  $H_1$  del tamaño  $m \times k$  correspondiente a los bits sistemáticos  $\mathbf{s}=(s_0, \dots, s_{k-1})$  y una sección determinante  $H_2$  del tamaño  $m \times m$  correspondiente a los bits de control de paridad  $(p_0, \dots, p_{m-1})$ , y donde  $H_2$  comprende una primera parte que comprende una columna  $h$  que tiene un peso impar superior a 2 y una segunda parte que comprende los elementos de matriz para la hilera  $i$ , la columna  $j$ ,  $0 \leq j \leq m-1$ ,  $0 \leq i \leq m-2$  igual a

- 10 1 para  $i=j$ ,
- 1 para  $i=j+1$ ,
- 0 en otras partes; y

transmitir (305) los bits de control de paridad junto con el conjunto de símbolos corrientes;

y donde  $[h_0, h_1, \dots, h_{m-2}, h_{m-1}]^T$  es la columna  $h$  al menos el elemento  $h_{m-1}$  es igual a 0.

15 2. El método de la reivindicación 1 donde el paso de utilizar la matriz H para determinar los bits de control de paridad  $(p_0, p_1, \dots, p_{m-1})$  comprende determinar  $p_{m-1}, \dots, p_1$  en forma recursiva mediante

$$p_{m-1} = w_{m-1}, p_i = p_{i+1} + w_i, i = m-2, \dots, 2, 1$$

donde

w es un vector  $= [w_0, w_1, \dots, w_{m-2}, w_{m-1}] = [h_0 p_0 + v_0, h_1 p_0 + v_1, \dots, h_{m-2} p_0 + v_{m-2},$

20  $h_{m-1} p_0 + v_{m-1}]$

y  $v = [v_0, v_1, \dots, v_{m-1}]^T = H_1 \mathbf{s}^T$ .

3. El método de la reivindicación 1 que comprende además el paso de:

determinar (307), que los bits de control de paridad adicionales deben ser transmitidos; y transmitir (311)

los bits de control de paridad  $(p_m, \dots, p_{m-1})$  sobre la base del conjunto de símbolos corrientes  $\mathbf{s}=(s_0, \dots, s_{k-1})$  y  $(p_0, \dots, p_{m-1})$ .

25

4. Un método para operar un receptor que estima un conjunto de símbolos corrientes  $\mathbf{s}=(s_0, \dots, s_{k-1})$ , comprendiendo el método los pasos de:

recibir (401) un vector de señal recibida  $\mathbf{y}=(y_0 \dots y_{n-1})$  que corresponde a una contraseña  $x$  que se

transmite que comprende un conjunto de símbolos  $\mathbf{s}=(s_0, \dots, s_{k-1})$  y bits de paridad  $(p_0, \dots, p_{m-1})$ ; y

30 utilizar una matriz H para estimar (403) el conjunto de símbolos corrientes  $(s_0, \dots, s_{k-1})$ , donde H comprende una sección no determinante  $H_1$  del tamaño  $m \times k$  correspondiente a los bits sistemáticos

$(s_0, \dots, s_{k-1})$  y una sección determinante  $H_2$  del tamaño  $m \times m$  correspondiente a los bits de control de paridad  $(p_0, \dots, p_{m-1})$ , y donde  $H_2$  comprende una primera parte que comprende una columna  $h$  que tiene un peso impar superior a 2 y una segunda parte que comprende los elementos de matriz para la hilera  $i$ , la columna  $j$ ,  $0 \leq i \leq m-1$ ,  $0 \leq j \leq m-2$  igual a

- 5            1 para  $i=j$ ,  
               1 para  $i=j+1$ ,  
               0 en todas partes;

y donde  $[h_0, h_1, \dots, h_{m-2}, h_{m-1}]^T$  es la columna  $h$  y al menos el elemento  $h_{m-1}$  es igual a 0.

5.            El método de la reivindicación 4 que comprende además el paso de:  
 10            determinar (407) que los bits de control de paridad adicionales serán recibidos; y  
               utilizar los bits de control de paridad adicionales recibidos para estimar (409) el conjunto de símbolos corrientes  $(s_0, \dots, s_{k-1})$ .

6.            El método de la reivindicación 4 que comprende además el paso de:  
 15            utilizar una matriz  $H^{(2)}$  para determinar el conjunto de símbolos corrientes  $(s_0, \dots, s_{k-1})$ , donde  $H^{(2)}$  se basa en  $H$ .

7.            Un aparato (100) que comprende:  
               medios de almacenamiento (103) que almacenan una matriz  $H$ ;  
               un microprocesador (101) que utiliza la matriz  $H$  para determinar los bits de control de paridad, donde  $H$  comprende una sección no determinante  $H_1$  del tamaño  $m \times k$  correspondiente a los bits sistemáticos  
 20             $(s_0, \dots, s_{k-1})$  y una sección determinante  $H_2$  del tamaño  $m \times m$  correspondiente a los bits de control de paridad  $(p_0, \dots, p_{m-1})$ , y donde  $H_2$  comprende una primera parte que comprende una columna  $h$  que tiene un peso impar superior a 2 y una segunda parte que comprende los elementos de matriz para la hilera  $i$ , la columna  $j$ ,  $0 \leq i \leq m-1$ ,  $0 \leq j \leq m-2$  igual a

- 1 para  $i=j$ ,  
 25            1 para  $i=j+1$ ,  
               0 en otras partes; y  
               un transmisor para transmitir los bits de control de paridad;

y donde  $[h_0, h_1, \dots, h_{m-2}, h_{m-1}]$  es la columna  $h$  al menos el elemento  $h_{m-1}$  es igual a 0.

8.            Un aparato (100) de acuerdo con la reivindicación 1, donde el uso de la matriz  $H$  para determinar los bits de  
 30            control de paridad  $(p_0, p_1, \dots, p_{m-1})$  comprende determinar  $p_{m-1}, \dots, p_1$  en forma recursiva mediante

$$p_{m-1} = w_{m-1}, p_i = p_{i+1} + w_i, i = m-2, \dots, 2, 1$$

donde

$w$  es un vector

$$= [w_0, w_1, \dots, w_{m-2}, w_{m-1}] = [h_0 p_0 + v_0, h_1 p_0 + v_1, \dots, h_{m-2} p_0 + v_{m-2}, h_{m-1} p_0 + v_{m-1}]$$

y  $\mathbf{v}=[v_0, v_1, \dots, v_{m-1}]^T = \mathbf{H}_1 \mathbf{s}^T$ .

9. Un aparato (200) que comprende:

medios de almacenamiento (203) que almacenan una matriz H;

un receptor para recibir un vector de señal  $\mathbf{y}=(y_0 \dots y_{n-1})$ , que corresponde a una contraseña x que se

5 transmite que comprende un conjunto de símbolos  $\mathbf{s}=(s_0, \dots, s_{k-1})$  y bits de paridad  $(\tilde{p}_0, \dots, \tilde{p}_{m-1})$ ; y

un microprocesador (201) que utiliza la matriz H para determinar un conjunto de símbolos corrientes

$(s_0, \dots, s_{k-1})$ , donde H comprende una sección no determinante  $H_1$  del tamaño  $m \times k$  correspondiente a los bits

sistemáticos  $(s_0, \dots, s_{k-1})$  y una sección determinante  $H_2$  del tamaño  $m \times m$  correspondiente a los bits de

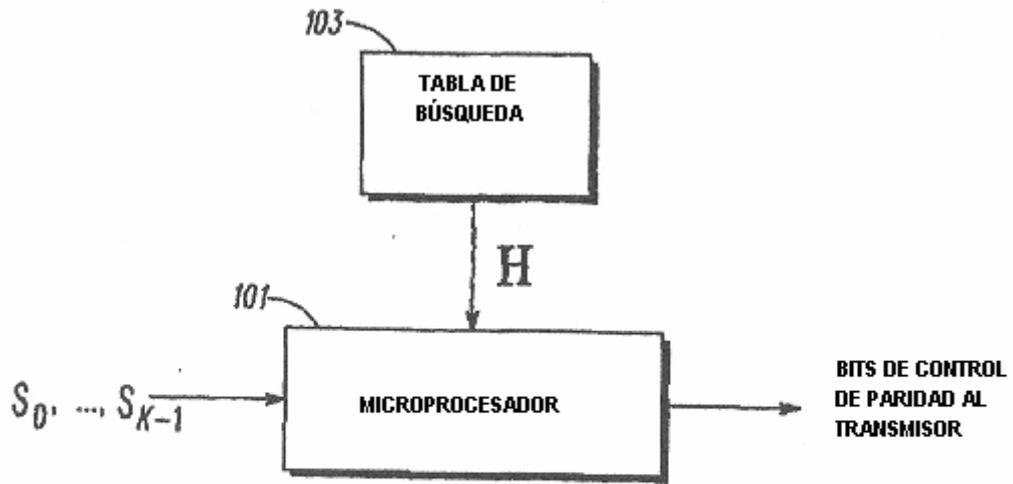
10 control de paridad  $(p_0, \dots, p_{m-1})$ , y donde  $H_2$  comprende una primera parte que comprende una columna h que tiene un peso impar superior a 2 y una segunda parte que comprende los elementos de matriz para la hilera i, la columna j,  $0 \leq i \leq m-1$ ,  $0 \leq j \leq m-2$  igual a

1 para  $i=j$ ,

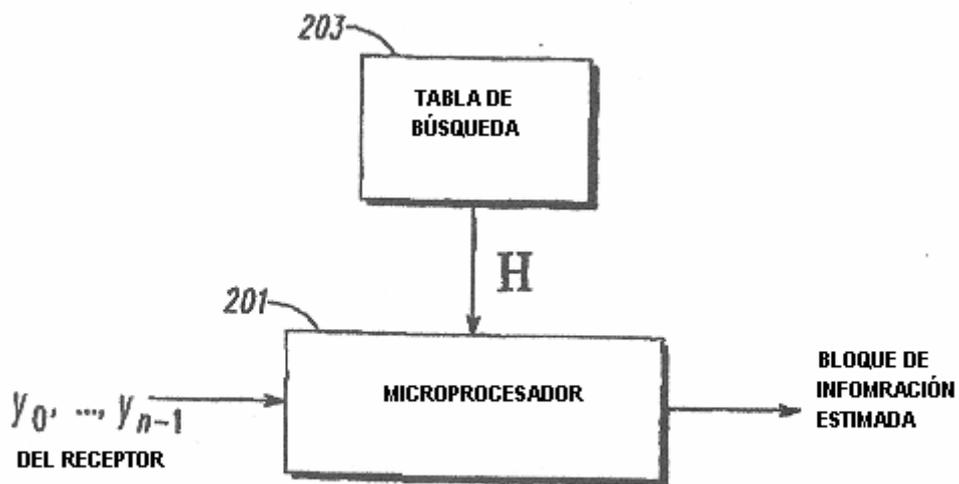
1 para  $i=j+1$ ,

0 en todas partes;

15 y donde  $[h_0, h_1, \dots, h_{m-2}, h_{m-1}]^T$  es la columna h y al menos el elemento  $h_{m-1}$  es igual a 0.



**FIG. 1**  
100



**FIG. 2**  
200

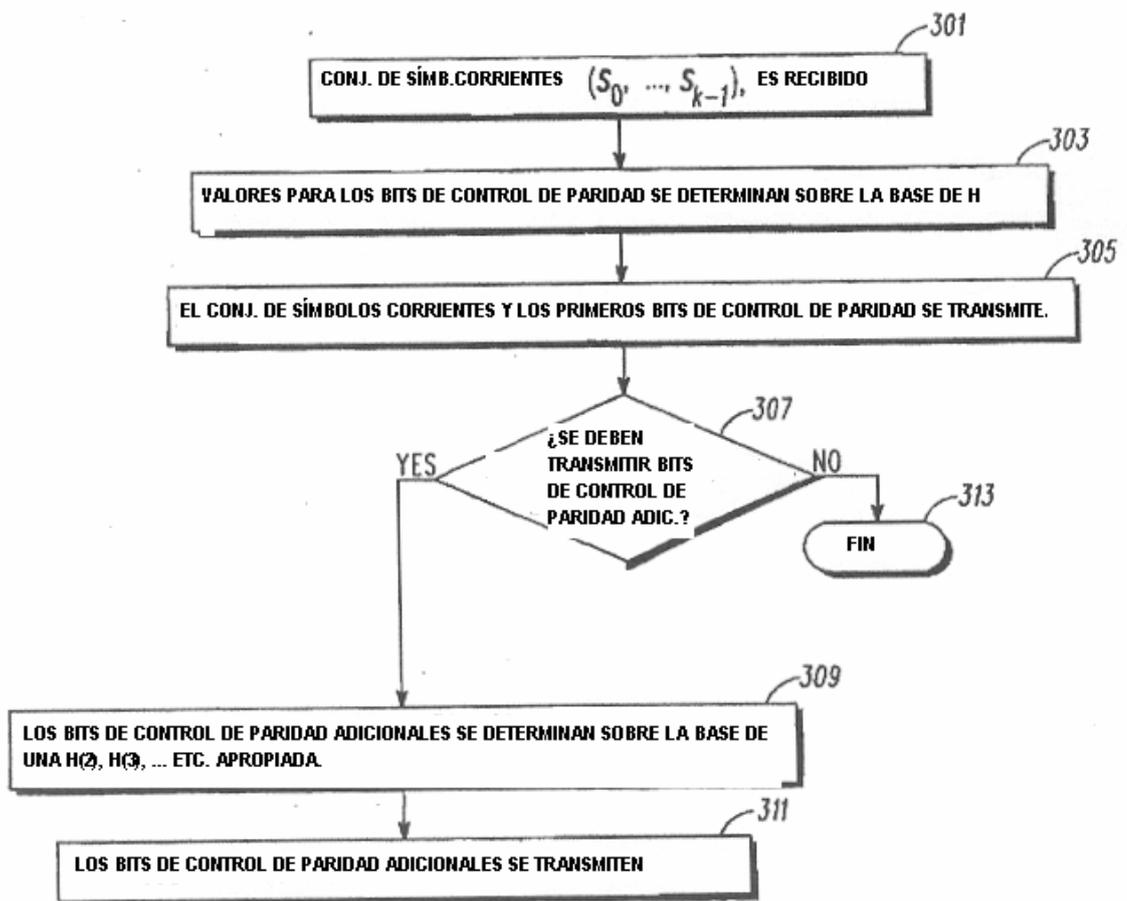


FIG. 3

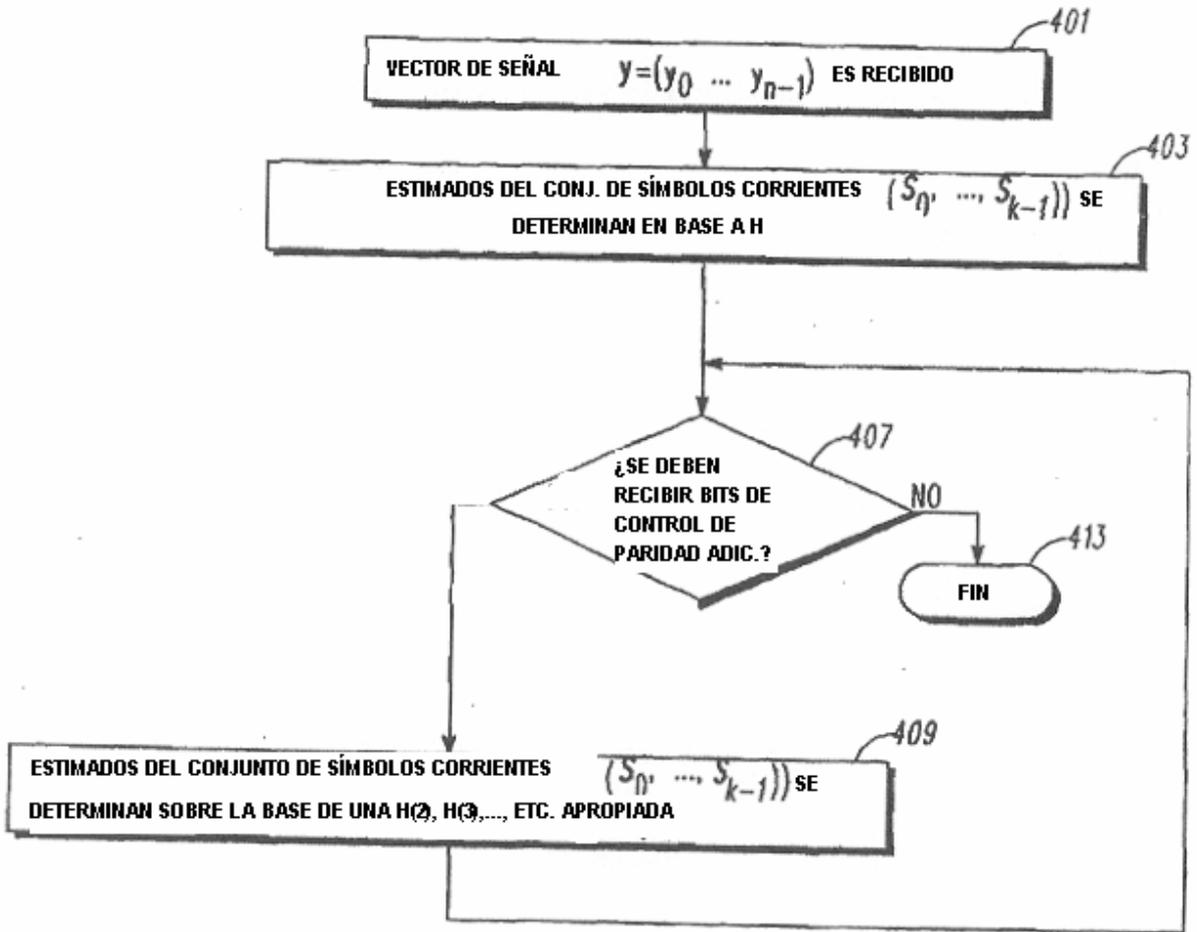


FIG. 4