

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 917 025**

51 Int. Cl.:

**H04L 1/00** (2006.01)

**H03M 13/09** (2006.01)

**H03M 13/13** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **06.02.2018 PCT/IB2018/050734**

87 Fecha y número de publicación internacional: **09.08.2018 WO18142374**

96 Fecha de presentación y número de la solicitud europea: **06.02.2018 E 18708205 (2)**

97 Fecha y número de publicación de la concesión europea: **06.04.2022 EP 3549292**

54 Título: **Código Polar concatenado con detección de errores adaptativa**

30 Prioridad:

**06.02.2017 US 201762455128 P**

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

**06.07.2022**

73 Titular/es:

**TELEFONAKTIEBOLAGET LM ERICSSON (PUBL)  
(100.0%)  
164 83 Stockholm, SE**

72 Inventor/es:

**BLANKENSHIP, YUFEI y  
HUI, DENNIS**

74 Agente/Representante:

**ELZABURU, S.L.P**

ES 2 917 025 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

Código Polar concatenado con detección de errores adaptativa

**Campo técnico**

5 Las realizaciones de la invención se refieren al campo de la comunicación inalámbrica; y más concretamente, a métodos, aparatos y sistemas para implementar código Polar concatenado con detección de errores adaptativa.

**Antecedentes**

10 Los códigos Polares, propuestos por E. Arikan en "Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric Binary-Input Memoryless Channels", Actas del IEEE sobre Teoría de la Información, vol. 55, págs. 3051-3073, julio de 2009, son la primera clase de esquemas de codificación constructivos que pueden demostrarse para lograr la capacidad simétrica de los canales discretos sin memoria de entrada binaria bajo un decodificador de cancelación sucesiva (SC) de baja complejidad. Sin embargo, el rendimiento de longitud finita de los códigos Polares bajo SC no es competitivo en comparación con otros esquemas de codificación de canales modernos, tales como los códigos de comprobación de paridad de baja densidad (LDPC) y los códigos Turbo. Más tarde, un decodificador de lista de SC (SCL) fue propuesto por I. Tal y A. Vardy en "List Decoding of Polar codes", en 15 Actas del Simposio del IEEE de Teoría de la Información, págs. 1-5, 2011, que puede aproximarse al rendimiento de un decodificador de máxima verosimilitud (ML) óptimo. Mediante la concatenación de una codificación simple de Comprobación de Redundancia Cíclica (CRC), se demostró que el rendimiento del código Polar concatenado es competitivo con el de los códigos de LDPC y Turbo bien optimizados. Como resultado, los códigos Polares se han adoptado como técnica de codificación de canales para la información de control de enlace descendente (DCI) y la 20 información de control de enlace ascendente (UCI) de Nueva Radio (NR) del 3GPP, donde NR es un sistema de comunicación inalámbrica de 5G.

25 La idea principal de la codificación Polar es transformar un par de canales de entrada binaria idénticos en dos canales distintos de diferentes calidades, uno mejor y otro peor que el canal de entrada binaria original. Repitiendo tal operación de polarización por pares en un conjunto de  $2^M$  usos independientes de un canal de entrada binaria, se puede obtener un conjunto de  $2^M$  "canales de bits" de diversas calidades. Algunos de estos canales de bits son casi perfectos (es decir, sin errores) mientras que el resto son casi inútiles (es decir, totalmente ruidosos). La cuestión es usar el canal casi perfecto para transmitir datos al receptor mientras que se configura la entrada a los canales inútiles para tener valores fijos o congelados (por ejemplo, 0) conocidos por el receptor. Por esta razón, se hace referencia comúnmente a esos bits de entrada al canal casi inútil y al casi perfecto como bits congelados y bits no 30 congelados (o de información), respectivamente. Solo los bits no congelados se utilizan para transportar datos en un código Polar. En la FIGURA 1 se ilustra una ilustración de la estructura de un código Polar de longitud 8.

35 Aunque se demostró que el código Polar original, como se propuso por Arikan, lograba la capacidad con un decodificador de cancelación sucesiva (SC) de baja complejidad, el rendimiento de longitud finita de los códigos Polares bajo SC no es competitivo en comparación con otros esquemas de codificación de canal modernos tales como códigos de LDPC y Turbo. Un decodificador más complejo, esto es, el decodificador de lista de SC (SCL), se propone por I. Tal y A. Vardy, donde se mantiene una lista de más de una ruta de decisión superviviente en el proceso de decodificación, pero el rendimiento resultante sigue siendo insatisfactorio. I. Tal y A. Vardy propusieron además que concatenando un código externo lineal, esto es, un código de CRC, con el código Polar original como 40 código interno, el código externo se puede usar para comprobar si alguna de las rutas candidatas en la lista se descifra correctamente. Tal proceso de decodificación de dos pasos mejora significativamente el rendimiento y hace que los códigos Polares sean competitivos con los códigos de LDPC y Turbo bien optimizados.

45 Cuando se diseñan códigos Polares concatenados para el control de corrección de errores de la información de control de enlace descendente (DCI) y la información de control de enlace ascendente (UCI) de NR, un tipo es la codificación Polar asistida por CRC (CA-Polar), donde los bits de CRC se adjuntan como un precodificador de códigos Polares para dos propósitos: detección de errores y corrección de errores. Según los enfoques anteriores, el número de bits de CRC utilizados para la detección y corrección de errores, respectivamente, es fijo. Sin embargo, un número fijo de bits de CRC implica capacidades fijas de detección y corrección de errores. Esto no se ajusta a las necesidades de los distintos tipos de tráfico en NR. Por ejemplo, se espera que la DCI asociada con los datos de Comunicaciones de Baja Latencia y Ultrafiabiles (URLLC) requiera una mayor fiabilidad que la DCI asociada con los 50 datos de eMBB. Por lo tanto, existe la necesidad de explorar métodos en los que la detección de errores adaptativa y la capacidad de corrección de errores se puedan soportar en NR.

55 Otro tipo de códigos Polares concatenados es PC-Polar, donde se genera una secuencia de bits de suma de comprobación de paridad (PC) antes de la codificación Polar. Similar a CA-Polar, la secuencia de bits de PC se fija para un  $(K, M)$  dado, y todos los bits de PC se utilizan para la corrección de errores. Aquí  $K$  es el número de bits de información y  $M$  el número de bits codificados para enviar por el aire. Para PC-Polar, también es deseable tener detección de errores adaptativa y capacidad de corrección de errores.

"Considerations in Polar code design" de Intel, R1-1700385, del WG1 de RAN de TSG del 3GPP, disponible en <https://portal.3gpp.org/ngppapp/CreateTdoc.aspx?mode=view&contributionId=754652> a partir del 18 de junio de

2019 trata consideraciones en el diseño de código polar con respecto a los canales de control, que incluyen: ordenamiento de bits de fiabilidad, tamaño de lista para evaluaciones/comparaciones de código y límites superiores en el tamaño del código. El documento R1-1700832 del 3GPP "Diseño de códigos Polares para canal de control", Qualcomm, describe un diseño de bits de CRC para la detección de errores y la decodificación de listas de códigos Polares. El documento US 2016/0079999 describe el intercalado de los bits de CRC antes de la codificación polar. El documento R1-1700088 del 3GPP, "Summary of polar code design for control channels", Huawei, HiSilicon, describe un diseño de códigos Polares de comprobación de paridad (PC). El documento R1-1700386 del 3GPP, "Polar code design", Intel, describe un diseño de código polar en el que el número de bits congelados de PC depende del tamaño del bloque de información y con bits de CRC para la detección de errores.

10 **Compendio**

Los aspectos de las realizaciones proporcionan sistemas y métodos para seleccionar de forma adaptativa un número total de bits de CRC o PC y/o asignar una cantidad diferente de bits de CRC disponibles entre detección y corrección de errores para códigos Polares.

15 Ciertas realizaciones de la presente descripción pueden proporcionar una o más ventajas técnicas. Por ejemplo, según varias realizaciones, una ventaja de las características en la presente memoria es permitir que la concatenación del código de CRC se personalice para diferentes cantidades de cargas útiles, diferentes parámetros de codificación necesarios para las condiciones del canal de comunicación variables y diferentes tipos de aplicaciones con diferentes requisitos en términos de, por ejemplo, latencia y fiabilidad. Dado que las comunicaciones inalámbricas de 5G necesitan cubrir una amplia gama de circunstancias y aplicaciones, las realizaciones de esta descripción permiten que el sistema asigne juiciosamente los recursos de radio en base a los compromisos de coste-beneficio.

Varias otras características y ventajas llegarán a ser obvias para un experto en la técnica a la luz de la descripción detallada y los dibujos siguientes. Ciertas realizaciones pueden tener ninguna, algunas o todas las ventajas citadas.

**Breve descripción de los dibujos**

25 Las figuras de dibujos adjuntas incorporadas y que forman parte de esta especificación ilustran varios aspectos de la descripción y, junto con la descripción, sirven para explicar los principios de la descripción.

La FIGURA 1 ilustra un ejemplo de una estructura de código Polar con  $N=8$ ;

la FIGURA 2 ilustra una red de comunicaciones celular de ejemplo, según ciertas realizaciones;

la FIGURA 3 ilustra un nodo de acceso por radio de ejemplo, según ciertas realizaciones;

30 la FIGURA 4 ilustra un nodo de acceso por radio virtualizado de ejemplo, según ciertas realizaciones;

la FIGURA 5 ilustra otro nodo de acceso por radio de ejemplo, según ciertas realizaciones;

la FIGURA 6 ilustra un equipo de usuario de ejemplo, según ciertas realizaciones;

la FIGURA 7 ilustra un equipo de usuario virtualizado de ejemplo, según ciertas realizaciones;

35 la FIGURA 8 ilustra una estructura de codificador de código polar concatenado sin intercalado de ejemplo, según ciertas realizaciones;

la FIGURA 9 ilustra una estructura de codificador con bits de CRC intercalados de ejemplo, según ciertas realizaciones;

la FIGURA 10 ilustra un código Polar de comprobación de paridad de ejemplo, según ciertas realizaciones;

40 la FIGURA 11 ilustra un método de ejemplo para seleccionar de forma adaptativa un número total de bits de CRC o PC, según ciertas realizaciones, según ciertas realizaciones;

la FIGURA 12 ilustra un dispositivo informático virtual, según ciertas realizaciones;

la FIGURA 13 ilustra un método de ejemplo mediante un transmisor para generar de forma adaptativa bits de precodificador para un código Polar, según ciertas realizaciones;

la FIGURA 14 ilustra otro dispositivo informático virtual, según ciertas realizaciones;

45 la FIGURA 15 ilustra otro método de ejemplo mediante un transmisor para generar de forma adaptativa bits de precodificador para un código Polar, según ciertas realizaciones;

la FIGURA 16 ilustra otro dispositivo informático virtual, según ciertas realizaciones;

la FIGURA 17 ilustra un método de ejemplo mediante un receptor para usar de forma adaptativa bits de precodificador para ayudar en la decodificación de un código Polar, según ciertas realizaciones; y

la FIGURA 18 ilustra otro dispositivo informático virtual, según ciertas realizaciones.

### Descripción de realizaciones

5 En la siguiente descripción, se exponen numerosos detalles específicos. Sin embargo, se entiende que las realizaciones de la invención se pueden poner en práctica sin estos detalles específicos. En otros casos, circuitos, estructuras y técnicas bien conocidos no se han mostrado en detalle con el fin de no dificultar la comprensión de esta descripción. Los expertos en la técnica, con las descripciones incluidas, serán capaces de implementar la funcionalidad apropiada sin experimentación indebida.

10 Las referencias en la memoria descriptiva a "una realización", "un ejemplo de realización", etc., indican que la realización descrita puede incluir un rasgo, estructura o característica particular, pero cada realización puede no incluir necesariamente el rasgo, estructura o característica particular. Además, tales frases no se refieren necesariamente a la misma realización. Además, cuando se describe un rasgo, estructura o característica en particular en conexión con una realización, se afirma que está dentro del conocimiento de un experto en la técnica implementar tal rasgo, estructura o característica en conexión con otras realizaciones ya sea o no descrito explícitamente.

15 En la siguiente descripción y reivindicaciones, se pueden usar los términos "acoplado" y "conectado", junto con sus derivados. Se debería entender que estos términos no pretenden ser sinónimos entre sí. "Acoplado" se utiliza para indicar que dos o más elementos, que pueden o no estar en contacto físico o eléctrico directo entre sí, cooperan o interactúan unos con otros. "Conectado" se utiliza para indicar el establecimiento de comunicación entre dos o más elementos que se acoplan unos con otros.

20 Las realizaciones expuestas a continuación representan información que permite a los expertos en la técnica poner en práctica las realizaciones e ilustrar el mejor modo de poner en práctica las realizaciones. Tras leer la siguiente descripción a la luz de las figuras de los dibujos adjuntos, los expertos en la técnica comprenderán los conceptos de la descripción y reconocerán aplicaciones de estos conceptos que no se abordan particularmente en la presente memoria.

25 **Nodo de Radio:** como se usa en la presente memoria, un "nodo de radio" es o bien un nodo de acceso por radio o bien un dispositivo inalámbrico.

30 **Nodo de Acceso por Radio:** Como se usa en la presente memoria, un "nodo de acceso por radio" es cualquier nodo en una red de acceso por radio de una red de comunicaciones celulares que opera para transmitir y/o recibir señales de forma inalámbrica. Algunos ejemplos de un nodo de acceso por radio incluyen, pero no se limitan a, una estación base (por ejemplo, un Nodo B mejorado o evolucionado (eNB) en una red de Evolución a Largo Plazo (LTE) del Proyecto de Asociación de Tercera Generación (3GPP), una estación base de alta potencia o macro estación base, una estación base de baja potencia (por ejemplo, una micro estación base, una pico estación base, un eNB doméstico o similar) y un nodo de retransmisión.

35 **Nodo de Red Central:** Como se usa en la presente memoria, un "nodo de red central" es cualquier tipo de nodo en una red central. Algunos ejemplos de un nodo de red central incluyen, por ejemplo, una Entidad de Gestión de Movilidad (MME), una Pasarela de Red de Datos por Paquetes (PDN) (P-GW), una Función de Exposición de Capacidad de Servicio (SCEF) o similares.

40 **Dispositivo Inalámbrico:** Como se usa en la presente memoria, un "dispositivo inalámbrico" es cualquier tipo de dispositivo que tiene acceso a (es decir, es atendido por) una red de comunicaciones celular mediante la transmisión y/o recepción de forma inalámbrica de señales a un(os) nodo(s) de acceso por radio. Algunos ejemplos de un dispositivo inalámbrico incluyen, pero no se limitan a, un dispositivo de Equipo de Usuario (UE) en una red de 3GPP y un dispositivo de Comunicación de Tipo Máquina (MTC).

45 **Nodo de Red:** Como se usa en la presente memoria, un "nodo de red" es cualquier nodo que sea parte o bien de la red de acceso por radio o bien de la red central de una red/sistema de comunicaciones celulares.

50 Obsérvese que la descripción dada en la presente memoria se centra en un sistema de comunicaciones celulares del 3GPP y, como tal, a menudo se usa terminología de LTE del 3GPP o terminología similar a la terminología de LTE del 3GPP. Sin embargo, los conceptos descritos en la presente memoria no se limitan a LTE o un sistema del 3GPP.

La FIGURA 2 ilustra un ejemplo de una red de comunicaciones celulares 10 según algunas realizaciones de la presente descripción. En las realizaciones descritas en la presente memoria, la red de comunicaciones celulares 10 es una red de LTE en la que algunos o todos los nodos de acceso por radio operan en una(s) portadora(s) en un espectro sin licencia. En una realización particular, por ejemplo, la red de comunicaciones celulares 10 puede operar en el espectro de 5 gigahercios (GHz). Sin embargo, la presente descripción no se limita a ello. En consecuencia, en otro ejemplo, la red de comunicaciones celulares 10 puede implementar LAA, LTE-U, MulteFire o alguna otra tecnología en la que los nodos de acceso por radio operen sobre una(s) portadora(s) sin licencia.

Como se representa, la red de comunicaciones celulares 10 incluye las estaciones base 12-1 y 12-2, a las que se hace referencia en LTE como eNB, que controlan las correspondientes macroceldas 14-1 y 14-2. Se hace referencia en la presente memoria a las estaciones base 12-1 y 12-2 generalmente colectivamente como estaciones base 12 e individualmente como estación base 12. Asimismo, se hace referencia en la presente memoria a las macroceldas 14-1 y 14-2 generalmente colectivamente como macroceldas 14 y individualmente como macrocelda 14. La red de comunicaciones celulares 10 también incluye una serie de nodos de baja potencia 16-1 hasta 16-4 que controlan las celdas pequeñas 18-1 hasta 18-4 correspondientes. En LTE, los nodos de baja potencia 16-1 hasta 16-4 pueden ser estaciones base pequeñas (tales como pico o femto estaciones base) o Cabeceras de Radio Remotas (RRH), o similares. En particular, aunque no se ilustre, una o más de las celdas pequeñas 18-1 hasta 18-4 se pueden proporcionar alternativamente por las estaciones base 12. Se hace referencia en la presente memoria a los nodos de baja potencia 16-1 hasta 16-4 generalmente colectivamente como de baja potencia. 16 e individualmente como nodo de baja potencia 16. Asimismo, se hace referencia en la presente memoria a las celdas pequeñas 18-1 hasta 18-4 generalmente colectivamente como celdas pequeñas 18 e individualmente como celda pequeña 18. Las estaciones base 12 (y opcionalmente los nodos de baja potencia 16) están conectados a una red central 20.

Las estaciones base 12 y los nodos de baja potencia 16 proporcionan servicio a los dispositivos inalámbricos 22-1 hasta 22-5 en las celdas 14 y 18 correspondientes. Se hace referencia en la presente memoria a los dispositivos inalámbricos 22-1 hasta 22-5 generalmente colectivamente como dispositivos inalámbricos 22 e individualmente como dispositivo inalámbrico 22. En LTE, se hace referencia a los dispositivos inalámbricos 22 como UE.

Según ciertas realizaciones, las macroceldas 14 se pueden proporcionar en un espectro de frecuencia con licencia (es decir, en el espectro de frecuencia dedicado a la red de comunicaciones celulares 10) tal como, por ejemplo, para la operación de LAA. En otras realizaciones, las macroceldas 14 se pueden proporcionar en un espectro de frecuencia sin licencia tal como, por ejemplo, para la operación de LAA en el espectro sin licencia (LAA-U) o MulteFire. Según ciertas realizaciones, una o más (y posiblemente todas) de las celdas pequeñas 18 se pueden proporcionar en un espectro de frecuencia sin licencia tal como, por ejemplo, el espectro de frecuencia de 5 GHz.

En una realización particular, las estaciones base 12, 14 que operan sobre una(s) portadora(s) en un espectro sin licencia pueden operar para realizar LBT y transmitir datos de Servicios de Multidifusión y Difusión Multimedia (MBMS) datos según cualquiera de las realizaciones descritas en la presente memoria.

La FIGURA 3 es un diagrama de bloques esquemático del nodo de acceso por radio 24, según ciertas realizaciones. El nodo de acceso por radio 24 puede ser, por ejemplo, una estación base 12, 16. Como se ilustra, el nodo de acceso por radio 24 incluye un sistema de control 26 que incluye uno o más procesadores 28 (por ejemplo, Unidades Centrales de Procesamiento (CPU), Circuitos Integrados de Aplicaciones Específicas (ASIC), Agrupaciones de Puertas Programables en Campo (FPGA) y/o similares), memoria 30 y una interfaz de red 32. Además, el nodo de acceso por radio 24 incluye una o más unidades de radio 34 que incluyen, cada una, uno o más transmisores 36 y uno o más receptores 38 acoplados a una o más antenas 40. En algunas realizaciones, la(s) unidad(es) de radio 34 son externas al sistema de control 26 y están conectadas al sistema de control 26 a través de, por ejemplo, una conexión por cable tal como, por ejemplo, un cable óptico. Sin embargo, en algunas otras realizaciones, la(s) unidad(es) de radio 34 y potencialmente la(s) antena(s) 40 se pueden integrar junto con el sistema de control 26. El uno o más procesadores 28 pueden operar para proporcionar una o más funciones de un nodo de acceso por radio 24 como se describe en la presente memoria. En algunas realizaciones, la(s) función/funciones se puede(n) implementar en software que está almacenado, tal como por ejemplo, en la memoria 30 y ejecutado por uno o más procesadores 28.

La FIGURA 4 es un diagrama de bloques esquemático que ilustra una realización virtualizada del nodo de acceso por radio 24, según ciertas realizaciones. Sin embargo, su descripción puede ser igualmente aplicable a otros tipos de nodos de red. Además, cualquiera de los tipos de nodos de red puede tener arquitecturas virtualizadas similares.

Como se usa en la presente memoria, un nodo de acceso por radio "virtualizado" es una implementación del nodo de acceso por radio 24 en el que al menos una parte de la funcionalidad del nodo de acceso por radio 24 se implementa como un(os) componente(s) virtual(es). Por ejemplo, la funcionalidad del nodo de acceso por radio se puede implementar a través de una(s) máquina(s) virtual(es) ejecutándose en un(os) nodo(s) de procesamiento físico en una(s) red(es), en una realización particular. En la realización de ejemplo ilustrada, el nodo de acceso por radio 24 incluye el sistema de control 26 que incluye uno o más procesadores 28 (por ejemplo, CPU, ASIC, FPGA y/o similares), la memoria 30 y la interfaz de red 32 y la una o más unidades de radio 34 que incluyen cada una el uno o más transmisores 36 y el uno o más receptores 38 acoplados a una o más antenas 40, como se ha descrito anteriormente. El sistema de control 26 está conectado a la(s) unidad(es) de radio 34 a través de, por ejemplo, un cable óptico o similar. El sistema de control 26 está conectado a uno o más nodos de procesamiento 42 acoplados a o incluidos como parte de una(s) red(es) 44 a través de la interfaz de red 32. Cada nodo de procesamiento 42 incluye uno o más procesadores 46 (por ejemplo, CPU, ASIC, FPGA, y/o similares), una memoria 48 y una interfaz de red 50.

Según ciertas realizaciones, las funciones 52 del nodo de acceso por radio 24 descrito en la presente memoria se pueden implementar en uno o más nodos de procesamiento 42 o distribuir a través del sistema de control 26 y uno o más nodos de procesamiento 42 de cualquier manera deseada. En algunas realizaciones particulares, algunas o todas las funciones 52 del nodo de acceso por radio 24 descritas en la presente memoria se implementan como

componentes virtuales ejecutados por una o más máquinas virtuales implementadas en un(os) entorno(s) virtual(es) alojados por el/los nodo(s) de procesamiento 42. Como se puede apreciar por un experto en la técnica, se puede utilizar señalización o comunicación adicional entre el/los nodo(s) de procesamiento 42 y el sistema de control 26 con el fin de llevar a cabo al menos algunas de las funciones deseadas 52. En particular, en algunas realizaciones, el sistema de control 26 puede no estar incluido, en cuyo caso la(s) unidad(es) de radio 34 se comunica(n) directamente con el/los nodo(s) de procesamiento 42 a través de una(s) interfaz/interfaces de red apropiada(s).

Según ciertas realizaciones, se proporciona un programa informático que incluye instrucciones que, cuando se ejecuta por al menos un procesador, hace que el al menos un procesador lleve a cabo la funcionalidad del nodo de acceso por radio 24 o un nodo (por ejemplo, un nodo de procesamiento 42) que implementa una o más de las funciones 52 del nodo de acceso por radio 24 en un entorno virtual según cualquiera de las realizaciones descritas en la presente memoria. En algunas realizaciones, se proporciona un portador que comprende el producto de programa informático antes mencionado. El portador es uno de una señal electrónica, una señal óptica, una señal de radio o un medio de almacenamiento legible por ordenador (por ejemplo, un medio legible por ordenador no transitorio tal como una memoria).

La FIGURA 5 es un diagrama de bloques esquemático que ilustra otro nodo de acceso por radio 24 de ejemplo, según otras ciertas realizaciones. El nodo de acceso por radio 24 incluye uno o más módulos 54, cada uno de los cuales está implementado en software. El/los módulo(s) 54 proporciona(n) la funcionalidad del nodo de acceso por radio 24 descrito en la presente memoria. Esta discusión es igualmente aplicable al nodo de procesamiento 42 de la FIGURA 7 (descrito a continuación) donde los módulos 54 se pueden implementar en uno de los nodos de procesamiento 42 o distribuir a través de múltiples nodos de procesamiento 42 y/o distribuir a través del/de los nodo(s) de procesamiento 42 y el sistema de control 26.

La FIGURA 6 es un diagrama de bloques esquemático de un UE 56, según ciertas realizaciones. Como se ilustra, el UE 56 incluye uno o más procesadores 58 (por ejemplo, CPU, ASIC, FPGA y/o similares), una memoria 60 y uno o más transceptores 62, cada uno que incluye uno o más transmisores 64 y uno o más receptores 66 acoplados a una o más antenas 68. En algunas realizaciones, la funcionalidad del UE 56 descrita anteriormente se puede implementar total o parcialmente en software que, por ejemplo, está almacenado en la memoria 60 y ejecutado por el/los procesador(es) 58.

En algunas realizaciones, se proporciona un programa informático que incluye instrucciones que, cuando se ejecuta por al menos un procesador, hace que el al menos un procesador lleve a cabo la funcionalidad del UE 56 según cualquiera de las realizaciones descritas en la presente memoria. En algunas realizaciones, se proporciona un portador que comprende el producto de programa informático antes mencionado. El portador es uno de una señal electrónica, una señal óptica, una señal de radio o un medio de almacenamiento legible por ordenador (por ejemplo, un medio legible por ordenador no transitorio tal como una memoria).

La FIGURA 7 es un diagrama de bloques esquemático del UE 56 según algunas otras realizaciones de la presente descripción. El UE 56 incluye uno o más módulos 70, cada uno de los cuales está implementado en software. El/los módulo(s) 70 proporcionan la funcionalidad del UE 56 descrito en la presente memoria.

Varios nodos de red pueden realizar la funcionalidad descrita a continuación. Por ejemplo, un nodo de acceso (por ejemplo, un eNB) podría realizar los diversos pasos de intercalado proporcionados en la presente memoria. Un experto en la técnica se daría cuenta de que un receptor (por ejemplo, un UE) sería capaz de realizar la decodificación correspondiente, según un ejemplo. Por supuesto, se entendería fácilmente por un experto en la técnica que se podrían implementar varias combinaciones de nodos de radio para realizar la funcionalidad descrita en la presente memoria.

Se debería observar que, si bien la discusión en la presente memoria utiliza la información de control de enlace descendente (DCI) y la información de control de enlace ascendente (UCI) como ejemplo, los métodos descritos a continuación se pueden usar para cualquier tipo de transmisión de paquetes de información que requiera tanto la función de detección de errores como la función de corrección de errores. Por lo tanto, por ejemplo, los mismos métodos se pueden aplicar a paquetes de datos de canal de enlace ascendente físico, paquetes de canal de datos de enlace descendente físico, paquetes de control de capa más alta, etc.

Obsérvese que mientras que dos tipos de precodificadores, CRC y PC, se usan como ejemplo en la discusión a continuación, el mismo principio se puede aplicar a otros tipos de precodificadores, por ejemplo, otros tipos de códigos de bloque lineal.

Según ciertas realizaciones, cuando se diseñan códigos Polares concatenados para el control de corrección de errores de DCI y UCI de NR, los bits de CRC se adjuntan como un precodificador de códigos Polares con dos propósitos:

- Una secuencia de  $L_{d,0}$  bits de CRC, donde  $L_{d,0}$  es el número mínimo de bits de CRC necesarios para garantizar el nivel mínimo de capacidad de detección de errores. Usando LTE del 3GPP como referencia,  $L_{d,0} = 16$  para DCI,  $L_{d,0} = 8$  para UCI de mayor tamaño de bloque de información K.

- Unos  $L_{c,0}$  bits de CRC adjuntos. Los  $L_{c,0}$  bits de CRC adicionales se pueden utilizar con propósitos de detección de errores y/o corrección de errores.

Por lo tanto, un total de  $L_{total} = (L_{d,0} + L_{c,0})$  bits de CRC se adjuntan en el precodificador de códigos Polares. En una realización particular, los  $L_{total}$  bits de CRC se generan utilizando un polinomio generador de CRC de longitud  $L_{total}$ . En otra realización, los  $L_{d,0}$  bits de CRC se generan utilizando un primer polinomio generador de CRC de longitud  $L_{d,0}$ , mientras que los  $L_{c,0}$  bits de CRC se generan utilizando un segundo polinomio generador de CRC de longitud  $L_{c,0}$ .

Según ciertas realizaciones, se proporciona una asignación adaptativa de bits de CRC entre la corrección de errores y la detección de errores. El conjunto de  $L_{total}$  bits de CRC se pueden usar de una manera adaptativa para lograr la mejor combinación de rendimiento de detección de errores y rendimiento de corrección de errores. La adaptación se puede realizar de manera que se soporten al menos dos de las siguientes tres opciones de asignación de bits de CRC para diferentes propósitos, según una realización particular:

- OPCIÓN A: Menor capacidad de detección de errores, mayor capacidad de corrección de errores. En esta opción, los  $L_{d,1}$  bits de CRC se utilizan para la detección de errores, los  $L_{c,1}$  bits de CRC se utilizan para ayudar con la decodificación de SCL (es decir, la corrección de errores).
- OPCIÓN B: Capacidad media de detección de errores, capacidad media de corrección de errores. En esta opción, los  $L_{d,2}$  bits de CRC se utilizan para la detección de errores, los  $L_{c,2}$  bits de CRC se utilizan para ayudar con la decodificación de SCL (es decir, la corrección de errores).
- OPCIÓN C: Mayor capacidad de detección de errores, menor capacidad de corrección de errores. En esta opción, los  $L_{d,3}$  bits de CRC se utilizan para la detección de errores, los  $L_{c,3}$  bits de CRC se utilizan para ayudar con la decodificación de SCL (es decir, la corrección de errores).

En los escenarios anteriores,  $L_{total} = (L_{d,1} + L_{c,1}) = (L_{d,2} + L_{c,2}) = (L_{d,3} + L_{c,3})$ , con:  $L_{d,0} \leq L_{d,1} < L_{d,2} < L_{d,3}$ ,  $L_{c,0} \geq L_{c,1} > L_{c,2} > L_{c,3}$ . Se puede tener en cuenta que una mayor cantidad de bits de CRC utilizados para la decodificación de SCL generalmente requiere una implementación de decodificador más compleja, que luego es capaz de lograr un mejor rendimiento de BLER del mismo tamaño de bloque de información  $K$  y tamaño de palabra de código  $M$ . Obsérvese que en cada uno de los escenarios mencionados anteriormente, un subconjunto predeterminado de  $L_{c,i}$  bits de CRC se utilizan para la corrección de errores, para cualquier  $i = 1, 2, 3$ , pero los bits elegidos para la corrección de errores no necesitan ser consecutivos o contiguos y pueden estar, por ejemplo, separados uniformemente a través de los bits de CRC disponibles, en una realización particular. Si bien en las Opciones A-C se describen tres formas diferentes de equilibrar las capacidades de detección de errores y corrección de errores, se puede entender que son posibles más formas de equilibrar utilizando los mismos principios o principios similares.

Según ciertas realizaciones particulares, la adaptación se puede realizar en función de varios parámetros de configuración, incluyendo:

- Según diversas realizaciones particulares, la adaptación se puede realizar según diferentes niveles objetivo de fiabilidad del servicio. Por ejemplo, para aplicaciones de alta fiabilidad (por ejemplo, URLLC), se puede utilizar la Opción A, en una realización particular. Por otra parte, para aplicaciones de baja fiabilidad (por ejemplo, mMTC), se puede utilizar la Opción C, en una realización particular.
- Según diversas realizaciones particulares, la adaptación se puede realizar según el objetivo de latencia del paquete de datos asociado. La latencia objetivo también se puede reflejar en el número máximo de retransmisiones posibles para el paquete de datos asociado. Por ejemplo, para paquetes de datos con un requisito de baja latencia, se puede utilizar la Opción A, en una realización particular. Ejemplos de escenarios de baja latencia incluyen: paquete de video, paquete de voz y realimentación de canal instantánea. Se utilizan más bits de CRC para la corrección de errores, dado que la detección de un error puede que no ayude a las aplicaciones, en la medida que no vale la pena retrasar la retransmisión. Por otro lado, para paquetes de datos que puedan tolerar una alta latencia, se puede utilizar la Opción C, en una realización particular.
- Según diversas realizaciones particulares, la adaptación se puede realizar según diferentes categorías de receptores (o tipos de receptores). Típicamente, el receptor es el UE en el enlace descendente. Por ejemplo, para UE de menor coste, se desea una implementación de decodificador de SCL de menor complejidad, por lo tanto, se puede usar la Opción C, en una realización particular. Para los UE de mayor coste, se puede abordar una implementación de decodificador de SCL de mayor complejidad, por lo tanto, la Opción A se puede usar en una realización particular. Para un UE de coste medio, la Opción B se puede usar en un compromiso ejemplar.

Según ciertas realizaciones, el número total de bits de CRC ( $L_{total}$ ) se puede elegir de forma adaptativa según diferentes parámetros de configuración. Ejemplos incluyen:

- Según diversas realizaciones particulares, la elección del número total de bits de CRC se puede

seleccionar según diferentes niveles objetivo de fiabilidad del servicio. Por ejemplo, para aplicaciones de alta fiabilidad (por ejemplo, URLLC), se puede usar un número total mayor de bits de CRC. Por otro lado, para aplicaciones de baja fiabilidad (por ejemplo, mMTC), se usa un número menor de bits de CRC totales.

- Según varias realizaciones particulares, la elección del número total de bits de CRC se elige según la longitud del bloque de información  $K$ , y también se puede elegir según la longitud del bloque de código  $N$ , y/o la tasa de código  $R = K/N$ . Para una longitud de código fija  $N$ , se pueden usar más bits de CRC para una pequeña cantidad de bits de información  $K$ , o de manera equivalente, para una tasa más baja  $R$ , y viceversa.

Obsérvese que la adaptación del número total de bits de CRC  $L_{total}$  se puede hacer junto con su asignación correspondiente entre el propósito de corrección de errores ( $L_{c,i}$  bits) y el propósito de detección de errores ( $L_{d,i}$  bits) como se ha descrito anteriormente.

Según ciertas realizaciones, la colocación de los bits de CRC se puede elegir de forma adaptativa. La FIGURA 8 ilustra una estructura de codificador de código Polar concatenado sin intercalado, según ciertas realizaciones. Como se muestra, los bits de CRC se adjuntan como un bloque contiguo y no se intercalan con los bits de información. Típicamente, la secuencia de bits de CRC se adjunta al final de la secuencia de bits de información, como se muestra en la FIGURA 8.

Cuando la secuencia de los  $L_{total}$  bits de CRC se adjuntan como un bloque contiguo al final de la secuencia de bits de información, los bits de CRC se deben usar como un bloque para realizar la comprobación de CRC. Los bits de CRC no se pueden usar individualmente para realizar la comprobación de CRC. Por lo tanto, los bits de CRC se tratan igual que los bits de información en el proceso de decodificación de SCL hasta el final del Trellis. Al final del trellis, los bits de CRC se utilizan para realizar la comprobación de CRC y seleccionar la mejor palabra de código candidata como salida del decodificador.

Aunque los bits de CRC se adjuntan al final del bloque, algunos de los bits de CRC aún se pueden integrar en un decodificador de lista para el código interno Polar para limitar la decodificación dentro de un subespacio que sea consistente con esos bits de CRC integrados en el proceso de decodificación de listas. Esto mejoraría la capacidad de corrección de errores del proceso de decodificación de lista integrada resultante. Sin embargo, dado que los bits de CRC se agrupan entre sí al final, el beneficio de rendimiento es limitado. Si se desea un beneficio de rendimiento adicional, uno puede considerar el uso de una construcción de "bits de CRC entrelazados" como se describe a continuación.

La FIGURA 9 ilustra una estructura de codificador que usa bits de CRC intercalados, según ciertas realizaciones. Específicamente, la secuencia de los  $L_{total}$  bits de CRC se pueden intercalar con bits de información antes de la codificación Polar. En una realización particular, los bits de CRC se pueden acoplar con un intercalador tal que los bits de CRC se pueden usar individualmente para realizar una comprobación de CRC. En la FIGURA 9 se ilustra un ejemplo de la estructura de codificador que utiliza bits de CRC intercalados.

Según ciertas realizaciones, y como se representa, se puede añadir un intercalador entre el código externo de CRC y el código interno Polar. De este modo, los bits de CRC se intercalan con los bits de información antes de que se envíen a la entrada del codificador Polar interno. El intercalador está diseñado para facilitar la decodificación de listas del código Polar interno, donde el decodificador de SCL Polar puede tener en cuenta la estructura de dependencia de los bits de paridad y los bits de datos del código externo. En algunas aplicaciones, solo algunos ( $L_{c,i}$ ) bits de CRC se utilizan para tener en cuenta la estructura de dependencia de los bits de paridad para mejorar la capacidad de corrección de errores del código, mientras que los otros ( $L_{d,i}$ ) bits de CRC se tratan como otros bits de información en los que el decodificador formularía hipótesis sobre sus valores durante el proceso de decodificación. En esta realización alternativa, algunos de los bits de CRC distribuidos no se tratan igual que los bits de información. Los bits de CRC de corrección de errores se utilizan en el proceso de expansión del árbol para seleccionar las mejores rutas de decodificación de tal manera que cada ruta de decodificación superviviente seleccionada debe ser coherente con la estructura de dependencia de los bits de paridad. La comprobación de CRC no tiene que esperar hasta el final de la expansión de trellis.

Según ciertas realizaciones, en lugar de bits de CRC, en su lugar se pueden generar bits de suma de comprobación de paridad (PC) en el precodificador. Generalmente se reconoce que cualquiera de los métodos y técnicas anteriores descritos anteriormente como que son aplicables a bits de CRC son igualmente aplicables a bits de PC.

La FIGURA 10 ilustra un código PC-Polar, según ciertas realizaciones. Cada uno de los bits de PC, a los que también se puede hacer referencia como bits congelados de PC, se deriva como la suma de control de paridad de un subconjunto seleccionado de bits de información. Los bits de PC son similares a los bits de CRC intercalados, donde cada uno de los bits de PC se puede usar individualmente para seleccionar las mejores rutas de decodificación.

Similar a la asignación adaptativa de bits de CRC entre la corrección de errores y la detección de errores, los bits de PC se pueden usar para ambas funciones, según ciertas realizaciones. La división exacta de bits de PC entre estas dos funciones se puede realizar de manera adaptativa, como se trata a continuación. Específicamente, según ciertas realizaciones, el conjunto de  $L_{total}$  bits de PC se pueden usar de manera adaptativa para lograr la mejor combinación de rendimiento de detección de errores y rendimiento de corrección de errores. La adaptación se realiza de manera

que se soporten al menos dos de las siguientes tres opciones de asignación de bits de PC para diferentes propósitos, según una realización:

- 5 • OPCIÓN A: Menor capacidad de detección de errores, mayor capacidad de corrección de errores. En esta opción,  $L_{d,1}$  bits de PC se utilizan para la detección de errores,  $L_{c,1}$  bits de PC se utilizan para ayudar con la decodificación de SCL (es decir, la corrección de errores).
- OPCIÓN B: Capacidad media de detección de errores, capacidad media de corrección de errores. En esta opción,  $L_{d,2}$  bits de PC se utilizan para la detección de errores,  $L_{c,2}$  bits de PC se utilizan para ayudar con la decodificación de SCL (es decir, la corrección de errores).
- 10 • OPCIÓN C: Mayor capacidad de detección de errores, menor capacidad de corrección de errores. En esta opción,  $L_{d,3}$  bits de PC se utilizan para la detección de errores,  $L_{c,3}$  bits de PC se utilizan para ayudar con la decodificación de SCL (es decir, la corrección de errores).

En los escenarios anteriores,  $L_{total} = (L_{d,1} + L_{c,1}) = (L_{d,2} + L_{c,2}) = (L_{d,3} + L_{c,3})$ , con:  $0 \leq L_{d,1} < L_{d,2} < L_{d,3}$ ,  $L_{total} \geq L_{c,1} > L_{c,2} > L_{c,3}$ .

15 Similar a los bits CRC, el número total de bits de PC,  $L_{total}$ , se puede fijar para una combinación de tamaño ( $K$ ,  $M$ ), o variar con la combinación de tamaño. Aquí  $K$  es el número de bits de información, y  $M$  el número de bits codificados para enviar por el aire.

A diferencia de los bits de CRC, no hay un número mínimo distinto de cero de bits de PC que tengan que ser asignados para propósitos de detección de errores; es decir, todos los bits de PC se pueden usar para propósitos de corrección de errores.

20 Según ciertas realizaciones, una vez que se selecciona una opción de la Opción A-C, el decodificador Polar se ejecuta en consecuencia.

- Los  $L_{c,i}$ ,  $i=1, 2, 3$ , bits de PC se utilizan en medio de la decodificación de SCL para seleccionar la mejor ruta de decodificación durante la expansión de la ruta.
- 25 • Los  $L_{d,i}$ ,  $i=1, 2, 3$ , bits de PC no se utilizan en medio de la decodificación de SCL. En su lugar, los  $L_{d,i}$  bits de PC se tratan como bits de información durante la expansión de la ruta. Se toman decisiones difíciles para los  $L_{d,i}$  bits de PC. Entonces, los  $L_{d,i}$  bits de PC se usan como suma de comprobación al final de la decodificación de SCL para detectar si la salida de SCL es una palabra de código válida o no.

Similar a Polar asistido por CRC, la adaptación se puede realizar en función de varios parámetros de configuración, tales como el tipo de servicio de datos, el requisito de latencia, el objetivo de tasa de error de bloque y la categoría de UE. Además, tanto el valor  $L_{total}$ , como la división de  $L_{total}$  entre  $L_{c,i}$  y  $L_{d,i}$ , se pueden adaptar.

30 La FIGURA 11 es un diagrama de flujo que ilustra un método de ejemplo para seleccionar de forma adaptativa un número total de bits de CRC o PC, según ciertas realizaciones. Como se muestra en la FIGURA 11, en el paso 1100, un nodo de red selecciona adaptativamente el número total de, por ejemplo, bits de CRC. En el paso 1110, se puede asignar una cantidad diferente de bits de CRC disponibles entre detección de errores y corrección de errores para códigos Polares. En el paso 1120, los bits de CRC se pueden colocar dentro de un bloque de código. Como se ha descrito anteriormente, los bits de precodificador pueden ser bits de CRC o bits de suma de comprobación de paridad (PC), según diversas realizaciones.

35 En ciertas realizaciones, el método para seleccionar de forma adaptativa un número total de bits de CRC o PC como se ha descrito anteriormente se puede realizar por un aparato virtual de interconexión de redes informáticas. La FIGURA 12 ilustra un dispositivo informático virtual de ejemplo 1200 para seleccionar de forma adaptativa un número total de bits de CRC o PC, según ciertas realizaciones. En ciertas realizaciones, el dispositivo informático virtual 1200 puede incluir módulos para realizar pasos similares a los descritos anteriormente con respecto al método ilustrado y descrito en la FIGURA 11. Por ejemplo, el dispositivo informático virtual 1200 puede incluir un módulo de selección 1210, un módulo de asignación 1220, un módulo de colocación 1230 y cualquier otro módulo adecuado para seleccionar de forma adaptativa un número total de bits de CRC o PC. En algunas realizaciones, uno o más de los módulos se pueden implementar usando uno o más procesador(es) 28 de la FIGURA 3 o uno o más procesadores 58 de la FIGURA 6. En ciertas realizaciones, las funciones de dos o más de los diversos módulos se pueden combinar en un solo módulo.

40 El módulo de selección 1210 puede realizar las funciones de selección del dispositivo informático virtual 1200. Por ejemplo, en una realización particular, el módulo de selección 1210 puede seleccionar de forma adaptativa el número total de bits de CRC o PC.

45 El módulo de asignación 1220 puede realizar las funciones de asignación del dispositivo informático virtual 1200. Por ejemplo, en una realización particular, el módulo de asignación 1220 puede asignar una cantidad diferente de bits de CRC disponibles, por ejemplo, entre detección de errores y corrección de errores para códigos Polares.

El módulo de colocación 1230 puede realizar las funciones de colocación del dispositivo informático virtual 1200. Por ejemplo, en una realización particular, el módulo de colocación 1230 puede colocar los bits de CRC, por ejemplo, dentro de un bloque de código.

5 Otras realizaciones del dispositivo informático virtual 1200 pueden incluir componentes adicionales más allá de los mostrados en la FIGURA 12 que pueden ser responsables de proporcionar ciertos aspectos de la funcionalidad, incluyendo cualquier funcionalidad descrita anteriormente y/o cualquier funcionalidad adicional (incluyendo cualquier funcionalidad necesaria para soportar las soluciones descritas anteriormente). Los diversos tipos diferentes de dispositivos y nodos de radio pueden incluir componentes que tienen el mismo hardware físico pero configurados (por ejemplo, a través de programación) para soportar diferentes tecnologías de acceso por radio, o pueden representar componentes físicos parcial o totalmente diferentes.

10 La FIGURA 13 ilustra un método 1300 de ejemplo mediante un transmisor para generar de forma adaptativa bits de precodificador para un código Polar, según ciertas realizaciones. Según ciertas realizaciones, el transmisor puede ser un dispositivo inalámbrico tal como el dispositivo inalámbrico 56, descrito anteriormente. Según otras ciertas realizaciones, el transmisor puede ser un nodo de red tal como el nodo de acceso por radio 24 u otro nodo de red. En varias realizaciones, los bits de precodificador pueden incluir bits de CRC o bits de PC.

El método comienza en el paso 1310 cuando el transmisor adquiere al menos un parámetro de configuración del que depende un número total de bits de precodificador. El al menos un parámetro de configuración incluye una longitud de bloque de información  $K$ , y puede incluir al menos una de una longitud de bloque de código  $N$  y/o una tasa de código  $R = K/N$ .

20 En el paso 1320, el transmisor determina el número total de bits de precodificador. En una realización particular, por ejemplo, el transmisor puede asignar un primer número de los bits de precodificador disponibles ( $L_d$ ) a la detección de errores y un segundo número de los bits de precodificador disponibles ( $L_c$ ) a la corrección de errores. Según una realización particular, el primer número de bits de precodificador disponibles ( $L_d$ ) puede ser un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para un número de bits de información y el segundo número de bits de precodificador disponibles ( $L_c$ ) se puede determinar restando el primer número de bits de precodificador disponibles ( $L_d$ ) de un número total de bits de precodificador disponibles ( $L_{total}$ ). En otra realización particular, el primer número de los bits de precodificador disponibles ( $L_d$ ) asignado a la detección de errores puede ser mayor que un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores.

30 En aún otra realización particular, el transmisor puede determinar el número total de bits de precodificador realizando uno de:

- asignar un primer número de los bits de precodificador disponibles ( $L_{d,1}$ ) a la detección de errores y un segundo número de los bits de precodificador disponibles ( $L_{c,1}$ ) a la corrección de errores para una capacidad de detección de errores más baja y una capacidad de corrección de errores más alta;
- 35 • asignar un tercer número de los bits de precodificador disponibles ( $L_{d,2}$ ) a la detección de errores y un cuarto número de los bits de precodificador disponibles ( $L_{c,2}$ ) a la corrección de errores para una capacidad de detección de errores media y una capacidad de corrección de errores media;
- asignar un quinto número de los bits de precodificador disponibles ( $L_{d,3}$ ) a la detección de errores y un sexto número de los bits de precodificador disponibles ( $L_{c,3}$ ) a la corrección de errores para una mayor capacidad de detección de errores y una menor capacidad de corrección de errores, y

40 En cualquiera de los escenarios anteriores, para un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores, lo siguiente puede ser cierto:

$$L_{d,0} < L_{d,1} < L_{d,2} < L_{d,3}, \text{ y}$$

$$L_{c,1} > L_{c,2} > L_{c,3}.$$

45 En el paso 1330, el transmisor genera los bits de precodificador para un bloque de código según el número total determinado de bits del precodificador. Según varias realizaciones particulares, los bits del precodificador se pueden generar en base a al menos uno de requisitos de latencia, requisitos de fiabilidad, condiciones de canal inalámbrico como se indica por una tasa de código objetivo y recursos de radio disponibles como se indica por una longitud de código. Cuando los bits de precodificador son bits de CRC, por ejemplo, los bits de CRC se pueden generar usando un solo polinomio generado por CRC, en una realización particular. En otra realización particular, los bits de CRC se pueden generar utilizando dos o más polinomios generados por CRC.

50 En el paso 1340, el transmisor coloca los bits del precodificador dentro del bloque de código. En una realización particular, el transmisor puede colocar los bits del precodificador en el bloque de código como un bloque contiguo.

En otra realización, el transmisor puede usar un intercalador para colocar los bits de precodificador en posiciones intercaladas dentro del bloque de código.

En ciertas realizaciones, el método para generar de forma adaptativa bits de precodificador para un código Polar como se ha descrito anteriormente se puede realizar por un dispositivo informático virtual. La FIGURA 14 ilustra un dispositivo informático virtual 1400 de ejemplo para generar de forma adaptativa bits de precodificador para un código Polar, según ciertas realizaciones. En ciertas realizaciones, el dispositivo informático virtual 1400 puede incluir módulos para realizar pasos similares a los descritos anteriormente con respecto al método 1300 ilustrado y descrito en la FIGURA 13. Por ejemplo, el dispositivo informático virtual 1400 puede incluir al menos un módulo de adquisición 1410, un módulo de determinación 1420, un módulo de generación 1430, un módulo de colocación 1440 y cualquier otro módulo adecuado para generar de forma adaptativa bits de precodificador para un código Polar. En algunas realizaciones, uno o más de los módulos se pueden implementar usando uno o más procesadores 28 de la FIGURA 3 o uno o más procesadores 58 de la FIGURA 6. En ciertas realizaciones, las funciones de dos o más de los diversos módulos se pueden combinar en un solo módulo.

El módulo de adquisición 1410 puede realizar las funciones de adquisición del dispositivo informático virtual 1400. Por ejemplo, en una realización particular, el módulo de adquisición 1410 puede adquirir al menos un parámetro de configuración del que depende un número total de bits de precodificador.

El módulo de determinación 1420 puede realizar las funciones de determinación del dispositivo informático virtual 1400. Por ejemplo, en una realización particular, el módulo de determinación 1420 puede determinar el número total de bits de precodificador.

El módulo generador 1430 puede realizar las funciones de generación del dispositivo informático virtual 1400. Por ejemplo, en una realización particular, el módulo de generación 1430 puede generar los bits de precodificador para un bloque de código según el número total determinado de bits de precodificador.

El módulo de colocación 1440 puede realizar las funciones de colocación del dispositivo informático virtual 1400. Por ejemplo, en una realización particular, el módulo de colocación 1440 puede colocar los bits de precodificador dentro del bloque de código.

Otras realizaciones del dispositivo informático virtual 1400 pueden incluir componentes adicionales más allá de los mostrados en la FIGURA 14 que pueden ser responsables de proporcionar ciertos aspectos de la funcionalidad del transmisor, incluyendo cualquier funcionalidad descrita anteriormente y/o cualquier funcionalidad adicional (incluyendo cualquier funcionalidad necesaria para soportar las soluciones descritas anteriormente). Los diversos tipos diferentes de transmisores pueden incluir componentes que tienen el mismo hardware físico pero configurados (por ejemplo, a través de programación) para soportar diferentes tecnologías de acceso por radio, o pueden representar parcial o totalmente diferentes componentes físicos.

La FIGURA 15 ilustra otro método 1500 de ejemplo mediante un transmisor para generar de forma adaptativa bits de precodificador para un código Polar, según ciertas realizaciones. Según ciertas realizaciones, el transmisor puede ser un dispositivo inalámbrico tal como el dispositivo inalámbrico 56, descrito anteriormente. Según otras realizaciones determinadas, el transmisor puede ser un nodo de red tal como el nodo de acceso por radio 24 u otro nodo de red. En varias realizaciones, los bits de precodificador pueden incluir bits de CRC o bits de PC.

El método comienza en el paso 1510 cuando el transmisor asigna una cantidad diferente de bits de precodificador disponibles entre la detección de errores y la corrección de errores para el código Polar. Según varias realizaciones particulares, los bits de precodificador se pueden asignar en base a al menos uno de requisitos de latencia, requisitos de fiabilidad, condiciones de canal inalámbrico como se indica por una tasa de código objetivo y recursos de radio disponibles como se indica por una longitud de código.

En una realización particular, el transmisor puede asignar un primer número de los bits de precodificador disponibles ( $L_d$ ) a la detección de errores y un segundo número de los bits de precodificador disponibles ( $L_c$ ) a la corrección de errores. 24. Por ejemplo, el primer número de los bits de precodificador disponibles ( $L_d$ ) puede ser un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para un número de bits de información y el segundo número de bits de precodificador disponibles ( $L_c$ ) se determina restando el primer número de bits de precodificador disponibles ( $L_d$ ) de un número total de bits de precodificador disponibles ( $L_{total}$ ), en una realización particular. En otra realización, el primer número de los bits de precodificador disponibles ( $L_d$ ) asignado a la detección de errores puede ser mayor que un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores.

En aún otra realización particular, el transmisor puede determinar el número total de bits de precodificador realizando uno de:

- asignar un primer número de los bits de precodificador disponibles ( $L_{d,1}$ ) a la detección de errores y un segundo número de los bits de precodificador disponibles ( $L_{c,1}$ ) a la corrección de errores para una menor capacidad de detección de errores y una mayor capacidad de corrección de errores;

- asignar un tercer número de los bits de precodificador disponibles ( $L_{d,2}$ ) a la detección de errores y un cuarto número de los bits de precodificador disponibles ( $L_{c,2}$ ) a la corrección de errores para una capacidad de detección de errores media y una capacidad de corrección de errores media;
- asignar un quinto número de los bits de precodificador disponibles ( $L_{d,3}$ ) a la detección de errores y un sexto número de los bits de precodificador disponibles ( $L_{c,3}$ ) a la corrección de errores para una mayor capacidad de detección de errores y una menor capacidad de corrección de errores, y

En cualquiera de los escenarios anteriores, para un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores, lo siguiente puede ser cierto:

$$L_{d,0} \leftarrow L_{d,1} < L_{d,2} < L_{d,3}, \text{ y}$$

$$L_{c,1} > L_{c,2} > L_{c,3}.$$

En el paso 1520, el transmisor genera los bits de precodificador para un bloque de código según la asignación y el número total de bits de CRC. En una realización particular, por ejemplo, los bits de CRC se pueden generar utilizando un único polinomio generado por CRC. En otra realización particular, los bits de CRC se pueden generar utilizando dos o más polinomios generados por CRC.

En el paso 1530, el transmisor coloca los bits de precodificador dentro del bloque de código. En una realización particular, el transmisor puede colocar los bits de precodificador en el bloque de código como un bloque contiguo. En otra realización, el transmisor puede usar un intercalador para colocar los bits de precodificador en posiciones intercaladas dentro del bloque de código.

En ciertas realizaciones, el método para generar de forma adaptativa bits de precodificación para un código Polar como se ha descrito anteriormente se puede realizar por un dispositivo informático virtual. La FIGURA 16 ilustra un dispositivo informático virtual 1600 de ejemplo para generar de forma adaptativa bits de precodificador para un código Polar, según ciertas realizaciones. En ciertas realizaciones, el dispositivo informático virtual 1600 puede incluir módulos para realizar pasos similares a los descritos anteriormente con respecto al método 1500 ilustrado y descrito en la FIGURA 15. Por ejemplo, el dispositivo informático virtual 1600 puede incluir al menos un módulo de asignación 1610, un módulo de generación 1620, un módulo de colocación 1630 y cualquier otro módulo adecuado para generar de forma adaptativa bits de precodificador para un código Polar. En algunas realizaciones, uno o más de los módulos se pueden implementar usando uno o más procesadores 28 de la FIGURA 3 o uno o más procesadores 58 de la FIGURA 6. En ciertas realizaciones, las funciones de dos o más de los diversos módulos se pueden combinar en un solo módulo.

El módulo de asignación 1610 puede realizar las funciones de asignación del dispositivo informático virtual 1600. Por ejemplo, en una realización particular, el módulo de asignación 1610 puede asignar una cantidad diferente de bits de precodificador disponibles entre detección de errores y corrección de errores para el código Polar.

El módulo de generación 1620 puede realizar las funciones de generación del dispositivo informático virtual 1600. Por ejemplo, en una realización particular, el módulo de generación 1620 puede generar los bits de precodificador para un bloque de código según la asignación y un número total de bits de CRC.

El módulo de colocación 1630 puede realizar las funciones de colocación del dispositivo informático virtual 1600. Por ejemplo, en una realización particular, el módulo de colocación 1630 puede colocar los bits de precodificador dentro del bloque de código.

Otras realizaciones del dispositivo informático virtual 1600 pueden incluir componentes adicionales más allá de los mostrados en la FIGURA 16 que pueden ser responsables de proporcionar ciertos aspectos de la funcionalidad del transmisor, incluyendo cualquier funcionalidad descrita anteriormente y/o cualquier funcionalidad adicional (incluyendo cualquier funcionalidad necesaria para soportar las soluciones descritas anteriormente). Los diversos tipos diferentes de transmisores pueden incluir componentes que tienen el mismo hardware físico pero configurados (por ejemplo, a través de programación) para soportar diferentes tecnologías de acceso por radio, o pueden representar componentes físicos parcial o totalmente diferentes.

La FIGURA 17 ilustra un método 1700 de ejemplo por un receptor para usar de forma adaptativa bits de precodificador para ayudar a decodificar un código Polar, según ciertas realizaciones. Según ciertas realizaciones, el receptor puede ser un dispositivo inalámbrico tal como el dispositivo inalámbrico 56, descrito anteriormente. Según otras realizaciones determinadas, el receptor puede ser un nodo de red tal como el nodo de acceso por radio 24 u otro nodo de red. En varias realizaciones, los bits de precodificador pueden incluir bits de CRC o bits de PC.

El método comienza en el paso 1710 cuando el receptor asigna una cantidad diferente de bits de precodificador disponibles entre una detección de errores y una corrección de errores para un código Polar. Según varias realizaciones particulares, los bits de precodificador se pueden asignar en base a al menos uno de requisitos de

latencia, requisitos de fiabilidad, condiciones de canal inalámbrico como se indica por una tasa de código objetivo y recursos de radio disponibles como se indica por una longitud de código.

En una realización particular, el receptor puede asignar un primer número de los bits de precodificador disponibles ( $L_d$ ) a una detección de errores y un segundo número de los bits de precodificador disponibles ( $L_c$ ) a una corrección de errores. 24. Por ejemplo, el primer número de los bits de precodificador disponibles ( $L_d$ ) puede ser un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para un número de bits de información y el segundo número de los bits de precodificador disponibles ( $L_c$ ) se puede determinar restando el primer número de bits de precodificador disponibles ( $L_d$ ) de un número total de bits de precodificador disponibles ( $L_{total}$ ), en una realización particular. En otra realización, el primer número de bits de precodificador disponibles ( $L_d$ ) asignado a la detección de errores puede ser mayor que un número mínimo de bits de precodificador ( $L_{d,0}$ ) asociado con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores.

Según ciertas otras realizaciones, cuando se asignan los bits de precodificador, el receptor puede realizar uno de los siguientes pasos:

- asignar un primer número de los bits de precodificador disponibles ( $L_{d,1}$ ) a la detección de errores y un segundo número de los bits de precodificador disponibles ( $L_{c,1}$ ) a la corrección de errores para una menor capacidad de detección de errores y una mayor capacidad de corrección de errores;
- asignar un tercer número de los bits de precodificador disponibles ( $L_{d,2}$ ) a la detección de errores y un cuarto número de los bits de precodificador disponibles ( $L_{c,2}$ ) a la corrección de errores para la capacidad de detección de errores media y la capacidad de corrección de errores media; o
- asignar un quinto número de los bits de precodificador disponibles ( $L_{d,3}$ ) a la detección de errores y un sexto número de los bits de precodificador disponibles ( $L_{c,3}$ ) a la corrección de errores para una mayor capacidad de detección de errores y una menor capacidad de corrección de errores;

En los escenarios anteriores, el número mínimo de bits de precodificador ( $L_{d,0}$ ) se puede asociar con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores, y lo siguiente puede ser cierto:

$$L_{d,0} < L_{d,1} < L_{d,2} < L_{d,3}, \text{ y}$$

$$L_{c,1} > L_{c,2} > L_{c,3}$$

En el paso 1720, el receptor usa los bits de precodificador asignados para la corrección de errores para ayudar a la decodificación de un bloque de código. Después de decodificar el bloque de código, el receptor usa los bits de precodificador asignados para la detección de errores para realizar la detección de errores en los bits decodificados, en el paso 1730.

Según ciertas realizaciones, el método puede incluir además que el receptor reciba, desde un transmisor, una indicación de la cantidad diferente de bits de precodificador disponibles para asignación entre detección de errores y corrección de errores para los códigos Polares. El receptor puede realizar el paso de asignación 1710 en base a la indicación.

En ciertas realizaciones, el método para usar de forma adaptativa bits de precodificador para ayudar a la decodificación de un código Polar como se ha descrito anteriormente se puede realizar por un dispositivo informático virtual.

La FIGURA 18 ilustra un dispositivo informático virtual 1800 de ejemplo para usar de forma adaptativa bits de precodificador para ayudar a la decodificación de un código Polar, según ciertas realizaciones. En ciertas realizaciones, el dispositivo informático virtual 1800 puede incluir módulos para realizar pasos similares a los descritos anteriormente con respecto al método 1700 ilustrado y descrito en la FIGURA 17. Por ejemplo, el dispositivo informático virtual 1800 puede incluir al menos un módulo de asignación 1810, un primer módulo de uso 1820, un segundo módulo de uso 1830, y cualquier otro módulo adecuado para usar de forma adaptativa bits de precodificador para ayudar a la decodificación de un código Polar. En algunas realizaciones, uno o más de los módulos se pueden implementar usando uno o más procesadores 28 de la FIGURA 3 o uno o más procesadores 58 de la FIGURA 6. En ciertas realizaciones, las funciones de dos o más de los diversos módulos se pueden combinar en un solo módulo.

El módulo de asignación 1810 puede realizar las funciones de asignación del dispositivo informático virtual 1800. Por ejemplo, en una realización particular, el módulo de asignación 1810 puede asignar una cantidad diferente de bits de precodificador disponibles entre detección de errores y corrección de errores para un código Polar.

El primer módulo de uso 1820 puede realizar ciertas de las funciones de uso del dispositivo informático virtual 1800. Por ejemplo, en una realización particular, el primer módulo de uso 1820 puede usar los bits de precodificador asignados para la corrección de errores para ayudar a la decodificación de un bloque de código.

El segundo módulo de uso 1830 puede realizar otras ciertas de las funciones de uso del dispositivo informático

virtual 1800. Por ejemplo, en una realización particular, el segundo módulo de uso 1830 puede usar los bits de precodificador asignados para detección de errores para realizar la detección de errores en los bits decodificados.

5 Otras realizaciones del dispositivo informático virtual 1800 pueden incluir componentes adicionales más allá de los mostrados en la FIGURA 18 que pueden ser responsables de proporcionar ciertos aspectos de la funcionalidad del receptor, incluyendo cualquier funcionalidad descrita anteriormente y/o cualquier funcionalidad adicional (incluyendo cualquier funcionalidad necesaria para soportar las soluciones descritas anteriormente). Los diversos tipos diferentes de receptores pueden incluir componentes que tienen el mismo hardware físico pero configurados (por ejemplo, a través de programación) para soportar diferentes tecnologías de acceso por radio, o pueden representar parcial o totalmente diferentes componentes físicos.

10 Como se describe en la presente memoria, se describen métodos, sistemas y aparatos para seleccionar de forma adaptativa el número total de bits de CRC o PC, asignar una cantidad diferente de bits de CRC disponibles entre detección de errores y corrección de errores para códigos Polares y colocar los bits de CRC dentro de un bloque de código. Como resultado, en el decodificador SCL se pueden lograr varios equilibrios de capacidades de corrección de errores y de detección de errores. Las características descritas permiten que la concatenación de código de CRC y PC se personalice para diferentes cantidades de cargas útiles, diferentes parámetros de codificación necesarios para las condiciones de canal de comunicación variables y diferentes tipos de aplicaciones con diferentes requisitos en términos de, por ejemplo, latencia y fiabilidad. Dicho de otra manera, según varias realizaciones, las características anteriores se pueden basar en los requisitos (por ejemplo, latencia o fiabilidad) de la aplicación subyacente, las condiciones de canal inalámbrico según se indica por la tasa de código objetivo y/o los recursos de radio disponibles según se indica por la longitud de código.

20 En una realización, los bits de precodificador pueden ser bits de CRC. Según una realización alternativa, los bits de precodificador son bits de suma de comprobación de paridad (PC).

25 Según varias realizaciones, una ventaja de las características en la presente memoria es permitir que la concatenación del código de CRC se personalice para diferentes cantidades de cargas útiles, diferentes parámetros de codificación necesarios para las condiciones de canal de comunicación variables y diferentes tipos de aplicaciones con diferentes requisitos en términos de, por ejemplo, latencia y fiabilidad. Dado que las comunicaciones inalámbricas de 5G necesitan cubrir una amplia gama de circunstancias y aplicaciones, las realizaciones de esta descripción permiten que el sistema asigne juiciosamente los recursos de radio en débase a compromisos de coste-beneficio.

30 Si bien los procesos en las figuras pueden mostrar un orden particular de operaciones realizadas por ciertas realizaciones de la invención, se debería entender que tal orden es ejemplar (por ejemplo, realizaciones alternativas pueden realizar las operaciones en un orden diferente, combinar ciertas operaciones, superponer ciertas operaciones, etc.).

35 Si bien la invención se ha descrito en términos de varias realizaciones, los expertos en la técnica reconocerán que la invención no se limita a las realizaciones descritas, se puede poner en práctica con modificaciones y alteraciones dentro del alcance de las reivindicaciones adjuntas. Por lo tanto, la descripción se ha de considerar como ilustrativa en lugar de limitativa.

**REIVINDICACIONES**

1. Un método por un transmisor (24, 56) para generar adaptativamente un número total de bits de precodificador para un código Polar, comprendiendo el método:

5 adquirir al menos un parámetro de configuración del que depende el número total de bits de precodificador para el código Polar, el al menos un parámetro de configuración que comprende una longitud de bloque de información K, todos los bits del precodificador que son de Comprobación de Redundancia Cíclica, CRC, o todos los bits del precodificador que son de Comprobación de Paridad, PC;

10 determinar el número total de bits de precodificador en base a al menos un parámetro de configuración, y asignar los bits de precodificador para equilibrar las capacidades de detección de errores y corrección de errores del código Polar, en donde determinar el número total de bits de precodificador comprende:

asignar un número dado de bits de precodificador disponibles para la detección de errores; y

asignar un número dado adicional de los bits de precodificador disponibles para la corrección de errores;

generar los bits de precodificador para un bloque de código según el número total determinado de bits de precodificador y la asignación de los bits de precodificador; y

15 colocar los bits de precodificador dentro del bloque de código,

en donde, en la determinación del número total de bits de precodificador, la asignación de bits de precodificador para equilibrar las capacidades de detección de errores y corrección de errores del código Polar comprende uno de:

20 asignar, como el número dado, un primer número de los bits de precodificador disponibles  $L_{d,1}$  a la detección de errores y, como el número dado adicional, un segundo número de los bits de precodificador disponibles  $L_{c,1}$  a la corrección de errores para una primera capacidad de detección de errores y una primera capacidad de corrección de errores; o

25 asignar, como el número dado, un tercer número de los bits de precodificador disponibles  $L_{d,2}$  a la detección de errores y, como el número dado adicional, un cuarto número de los bits de precodificador disponibles  $L_{c,2}$  a la corrección de errores para una segunda capacidad de detección de errores y una segunda capacidad de corrección de errores; o

asignar, como el número dado, un quinto número de los bits de precodificador disponibles  $L_{d,3}$  a la detección de errores y, como el número dado adicional, un sexto número de los bits de precodificador disponibles  $L_{c,3}$  a la corrección de errores para una tercera capacidad de detección de errores y una tercera capacidad de corrección de errores,

30 en donde la primera capacidad de detección de errores es menor que la segunda capacidad de detección de errores, la segunda capacidad de detección de errores es menor que la tercera capacidad de detección de errores, la primera capacidad de corrección de errores es mayor que la segunda capacidad de corrección de errores y la segunda capacidad de corrección de errores es mayor que la tercera capacidad de corrección de errores; y

35 en donde para un número mínimo de bits de precodificador  $L_{d,0}$  asociado con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores, lo siguiente es cierto:

$$L_{d,0} \leq L_{d,1} < L_{d,2} < L_{d,3} , \text{ y}$$

$$L_{c,1} > L_{c,2} > L_{c,3}.$$

40 2. El método de la reivindicación 1, en donde colocar los bits de precodificador dentro del bloque de código comprende colocar los bits de precodificador en posiciones intercaladas a través del uso de un intercalador.

3. El método de cualquiera de las reivindicaciones 1 y 2, en donde el paso de generar los bits de precodificador se basa en al menos uno de:

requisitos de latencia;

requisitos de fiabilidad;

45 condiciones de canal inalámbrico según se indica por una tasa de código objetivo; y

recursos de radio disponibles según se indica por una longitud de código.

4. El método de cualquiera de las reivindicaciones 1 a 3, en donde los bits de precodificador son bits de CRC.

5. El método de la reivindicación 4, en donde generar los bits de precodificador comprende generar bits de CRC utilizando un único polinomio generador de CRC, o en donde generar los bits de precodificador comprende generar bits de CRC utilizando dos o más polinomios generadores de CRC.

6. El método de cualquiera de las reivindicaciones 1 a 3, en donde los bits de precodificador son bits de PC.

5 7. El método de cualquiera de las reivindicaciones 1 a 6, en donde el transmisor comprende un dispositivo inalámbrico (56) o en donde el transmisor comprende un nodo de red (24).

8. Un transmisor (24, 56) para generar adaptativamente un número total de bits de precodificador para un código Polar, comprendiendo el transmisor:

al menos un procesador (58) configurado para:

10 adquirir al menos un parámetro de configuración del que depende el número total de bits de precodificador, el al menos un parámetro de configuración que comprende una longitud de bloque de información K, todos los bits de precodificador que son de Comprobación de Redundancia Cíclica, CRC, o todos los bits de precodificador que son de Comprobación de Paridad, PC;

15 determinar el número total de bits de precodificador en base a al menos un parámetro de configuración y asignar los bits de precodificador para equilibrar las capacidades de detección de errores y corrección de errores del código Polar, en donde al menos un procesador (58) está configurado además para determinar el número total de bits de precodificador mediante:

asignar un número dado de bits de precodificador disponibles para la detección de errores; y

asignar un número dado adicional de los bits de precodificador disponibles para la corrección de errores;

20 generar los bits de precodificador para un bloque de código según el número total determinado de bits de precodificador y la asignación de los bits de precodificador; y

colocar los bits del precodificador dentro del bloque de código,

25 en donde, cuando se asignan bits de precodificador para equilibrar las capacidades de detección de errores y corrección de errores del código Polar en la determinación del número total de bits de precodificador, el al menos un procesador está configurado para:

asignar, como el número dado, un primer número de los bits de precodificador disponibles  $L_{d,1}$  a la detección de errores y, como el número dado adicional, un segundo número de los bits de precodificador disponibles  $L_{c,1}$  a la corrección de errores para una primera capacidad de detección de errores y una primera capacidad de corrección de errores; o

30 asignar, como el número dado, un tercer número de los bits de precodificador disponibles  $L_{d,2}$  a la detección de errores y, como el número dado adicional, un cuarto número de los bits de precodificador disponibles  $L_{c,2}$  a la corrección de errores para una segunda capacidad de detección de errores y una segunda capacidad de corrección de errores; o

35 asignar, como el número dado, un quinto número de los bits de precodificador disponibles  $L_{d,3}$  a la detección de errores y, como el número dado adicional, un sexto número de los bits de precodificador disponibles  $L_{c,3}$  a la corrección de errores para una tercera capacidad de detección de errores y una tercera capacidad de corrección de errores,

40 en donde la primera capacidad de detección de errores es menor que la segunda capacidad de detección de errores, la segunda capacidad de detección de errores es menor que la tercera capacidad de detección de errores, la primera capacidad de corrección de errores es mayor que la segunda capacidad de corrección de errores y la segunda capacidad de corrección de errores es mayor que la tercera capacidad de corrección de errores; y

en donde para un número mínimo de bits de precodificador  $L_{d,0}$  asociado con un nivel mínimo de detección de errores para proporcionar una mayor capacidad de detección de errores, lo siguiente es cierto:

45 
$$L_{d,0} \leq L_{d,1} < L_{d,2} < L_{d,3} , \text{ y}$$

$$L_{c,1} > L_{c,2} > L_{c,3}.$$

9. El transmisor (24, 56) de la reivindicación 8, en donde cuando se colocan los bits de precodificador dentro del bloque de código, el al menos un procesador (58) está configurado para colocar los bits de precodificador en posiciones intercaladas a través del uso de un intercalador.

10. El transmisor (24, 56) de cualquiera de las reivindicaciones 8 y 9, en donde los bits de precodificador se generan en base a al menos uno de:
- requisitos de latencia;
  - requisitos de fiabilidad;
- 5 condiciones de canal inalámbrico según se indica por una tasa de código objetivo; y
- recursos de radio disponibles según se indica por una longitud de código.
11. El transmisor (24, 56) de cualquiera de las reivindicaciones 8 a 10, en donde los bits de precodificador son bits de CRC.
- 10 12. El transmisor (24, 56) de la reivindicación 11, en donde el al menos un procesador (58) está configurado para generar los bits de CRC utilizando un único polinomio generador de CRC, o en donde el al menos un procesador (58) está configurado para generar los bits de CRC bits utilizando dos o más polinomios generadores de CRC.
13. El transmisor (24, 56) de cualquiera de las reivindicaciones 8 a 10, en donde los bits de precodificador son bits de PC.
- 15 14. El transmisor (24) de cualquiera de las reivindicaciones 8 a 13, en donde el transmisor comprende un dispositivo inalámbrico (24) o en donde el transmisor comprende un nodo de red (56).

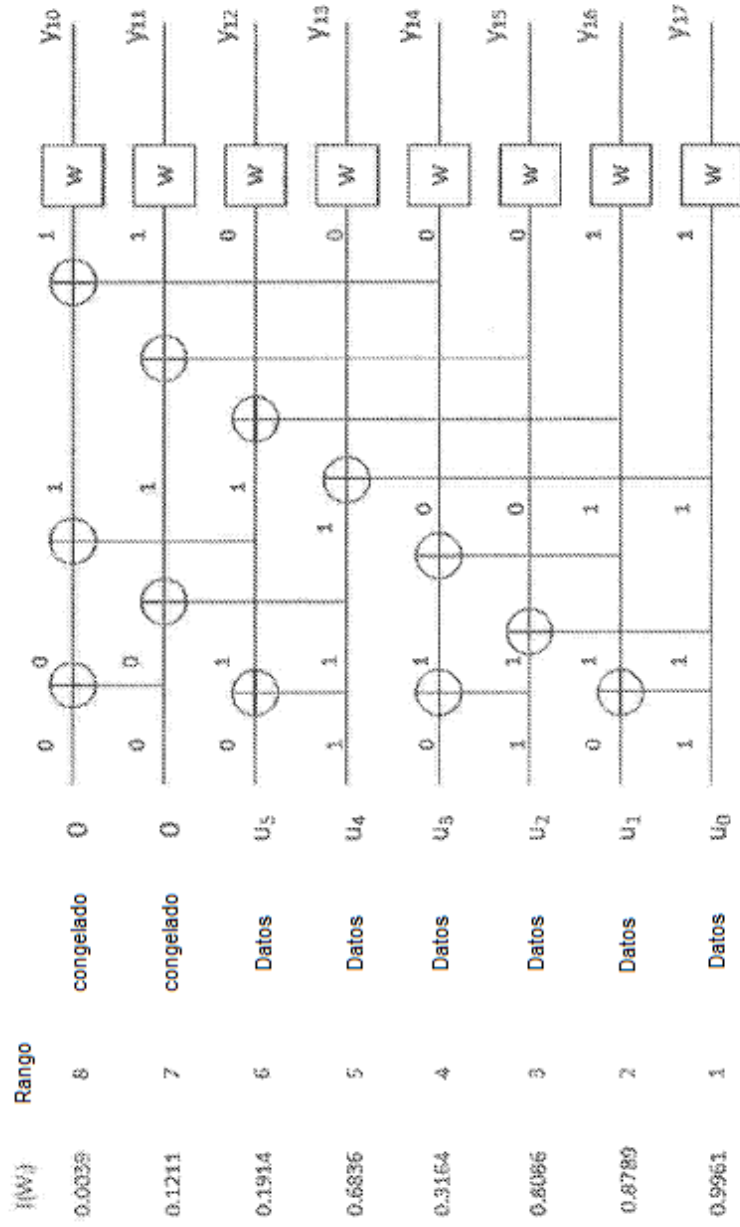


FIGURA 1

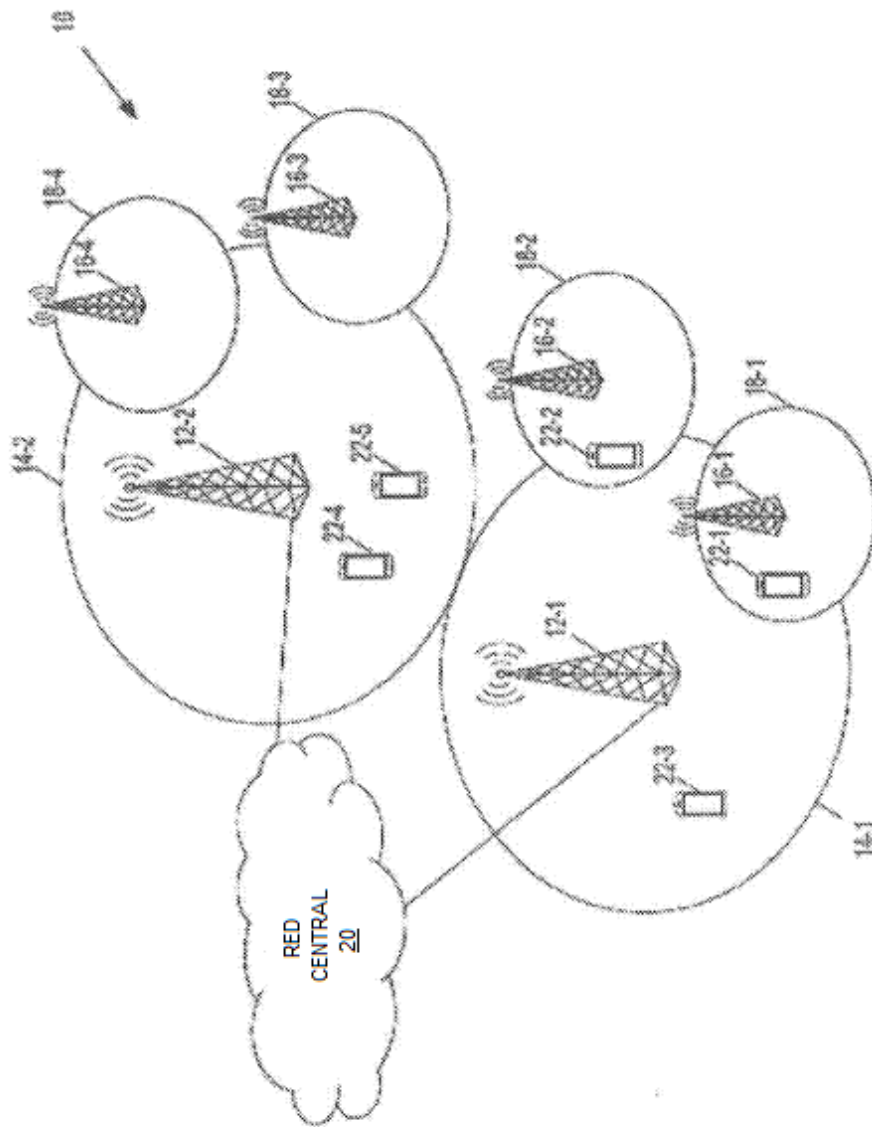


FIGURA 2

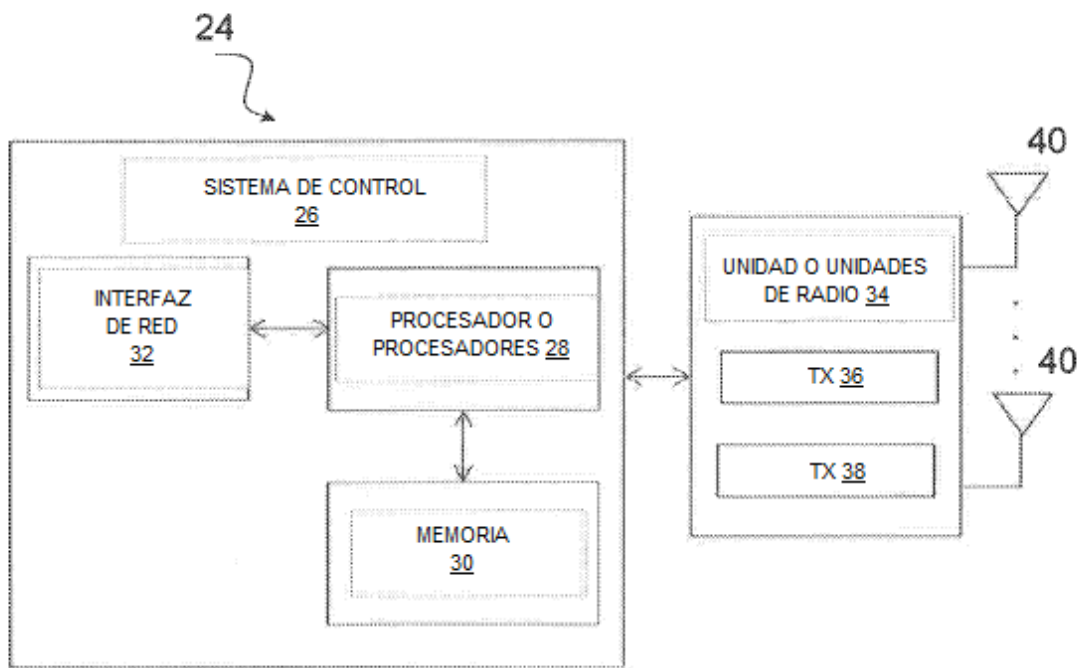


FIGURA 3

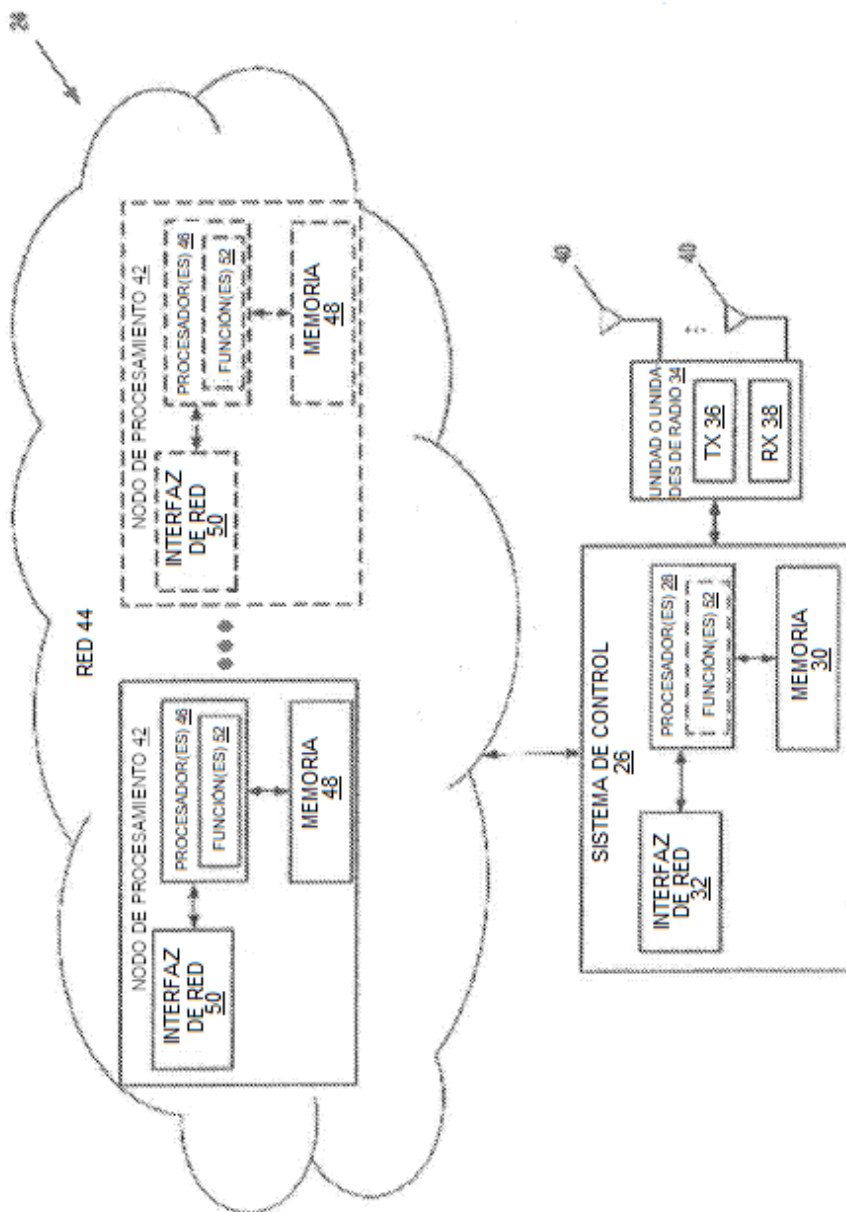


FIGURA 4

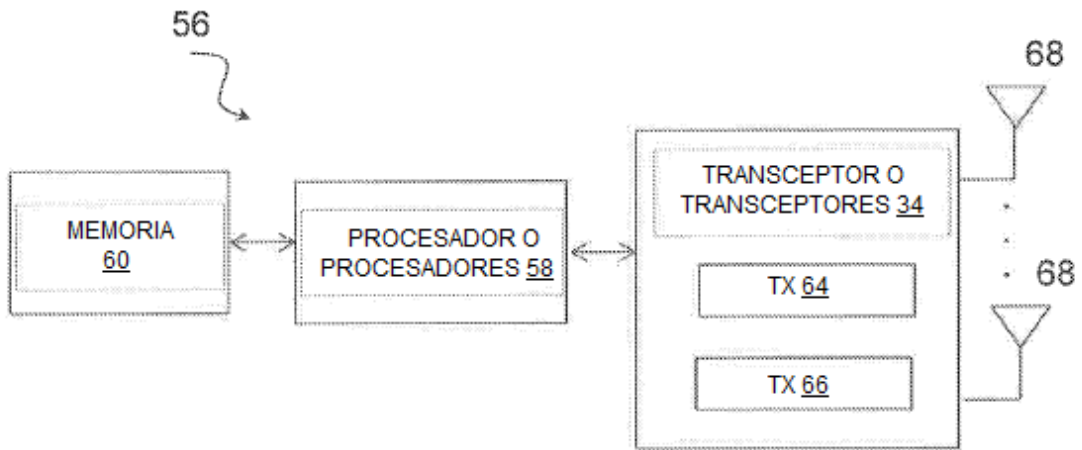


FIGURA 6

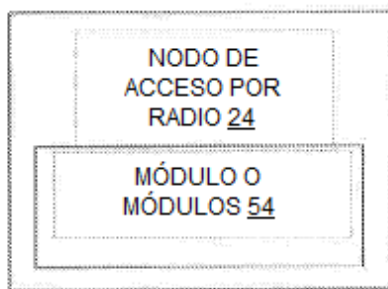


FIGURA 5

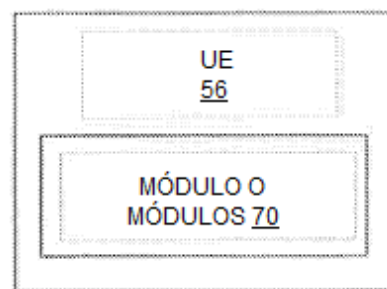


FIGURA 7

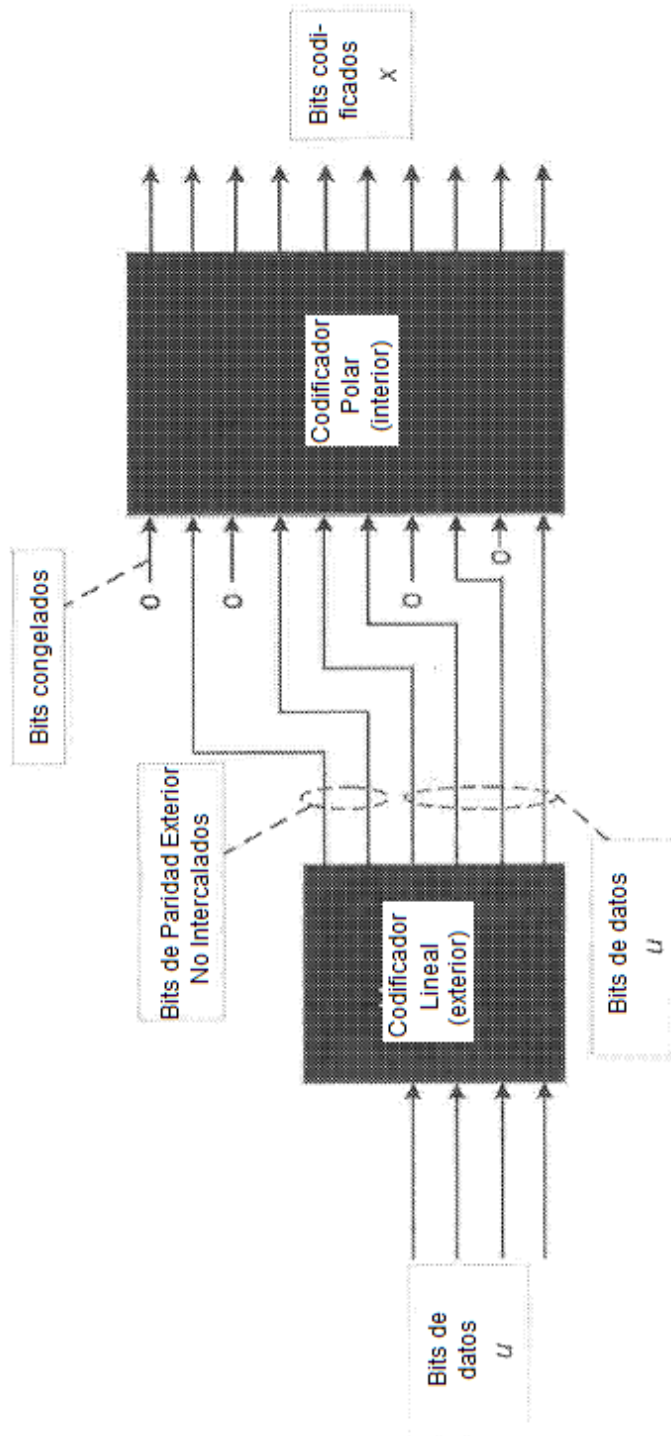


FIGURA 8

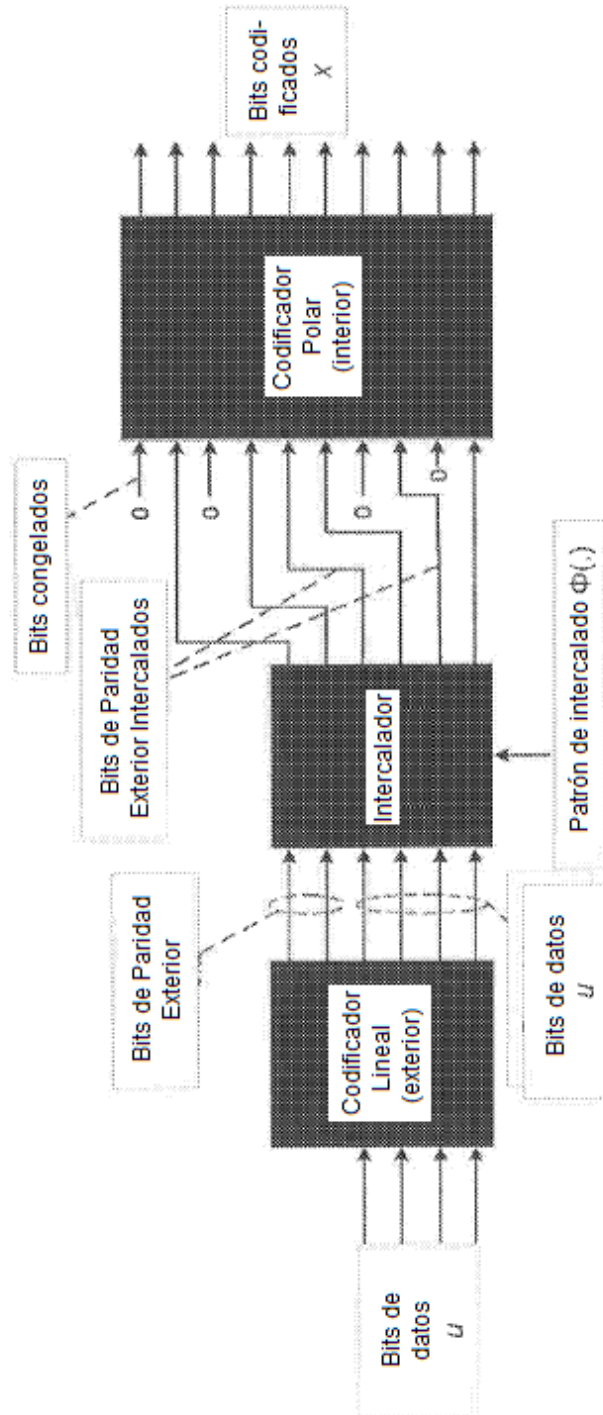


FIGURA 9

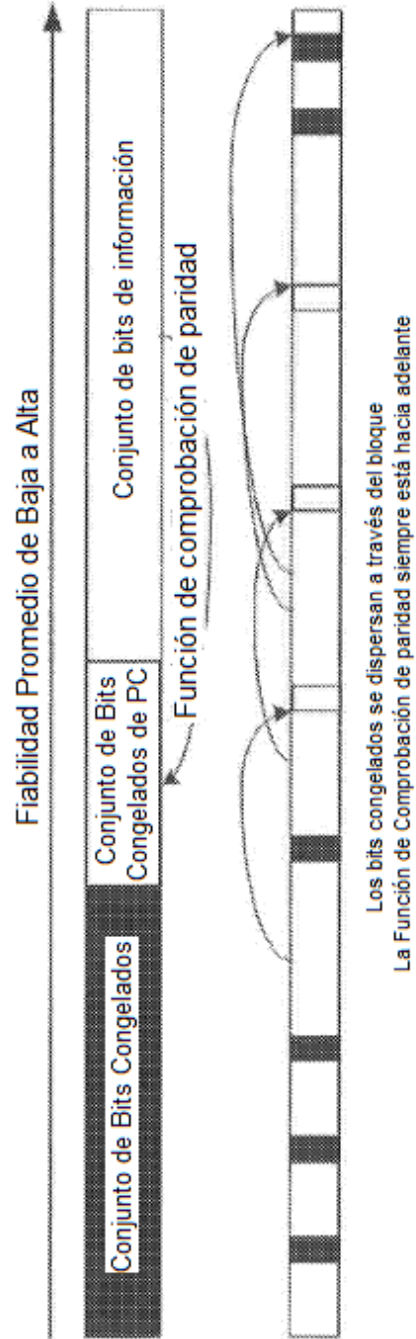


FIGURA 10

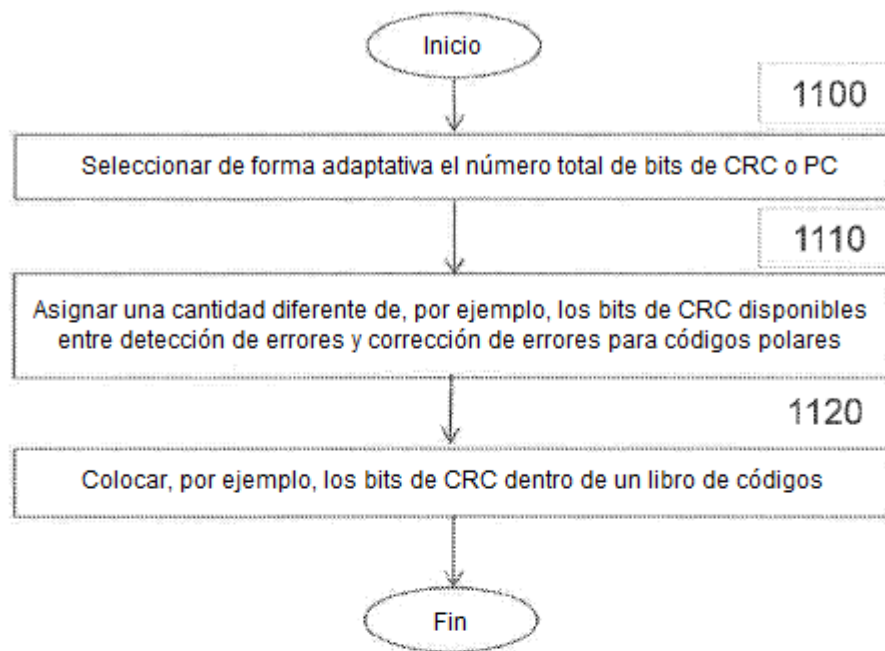


FIGURA 11

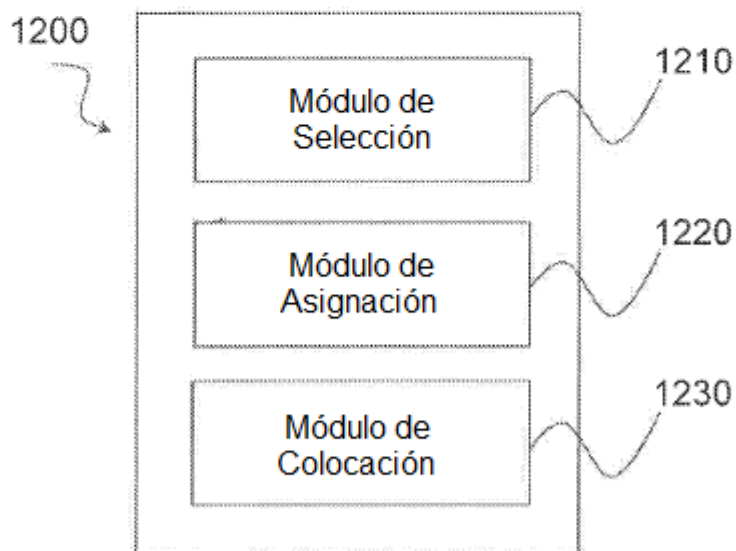


FIGURA 12

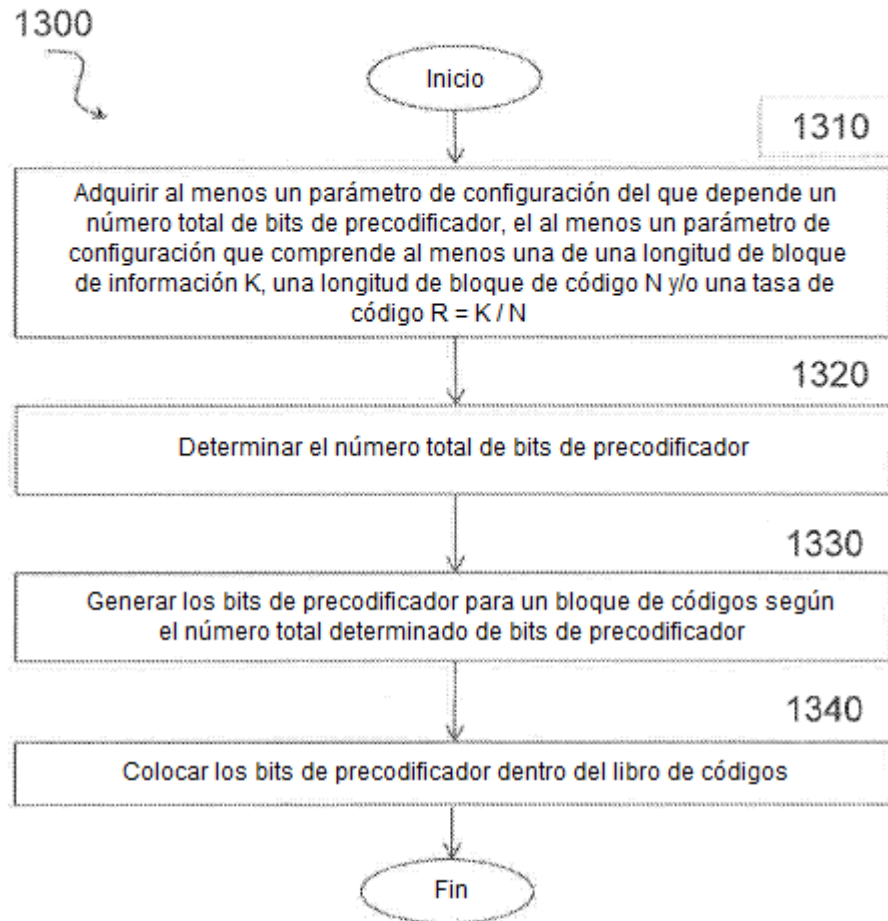


FIGURA 13

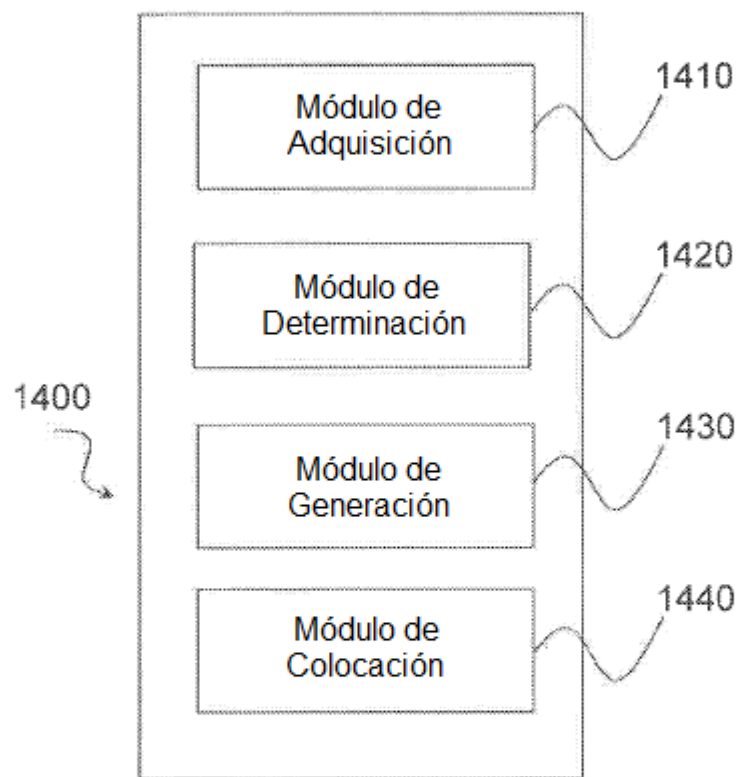


FIGURA 14

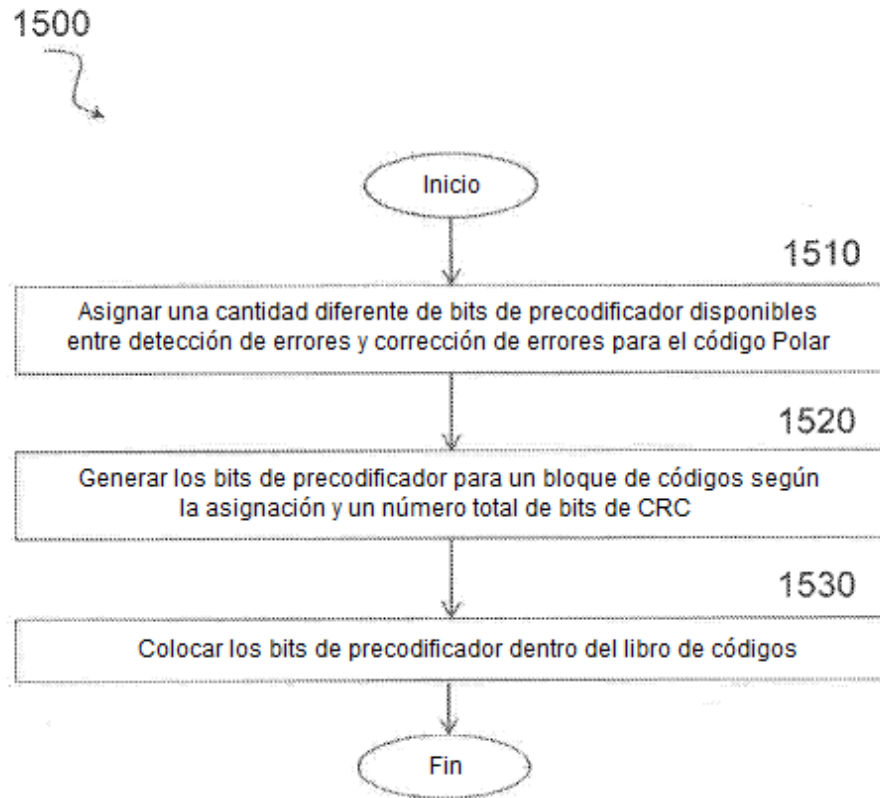


FIGURA 15

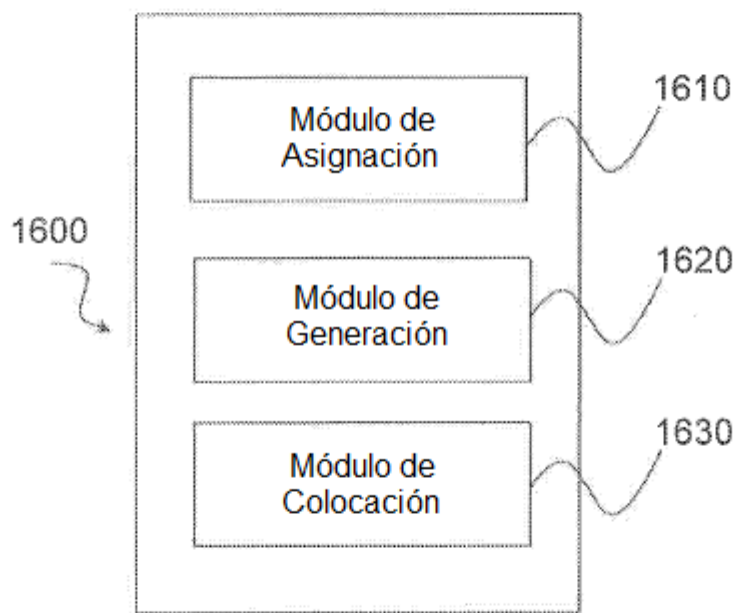


FIGURA 16

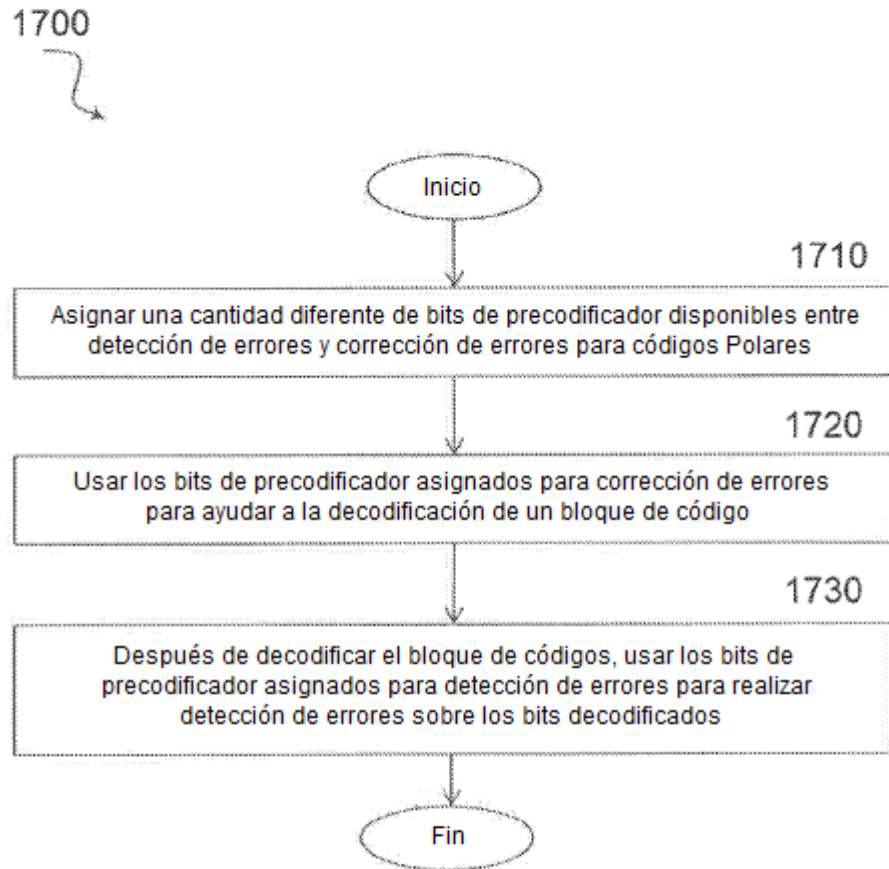


FIGURA 17

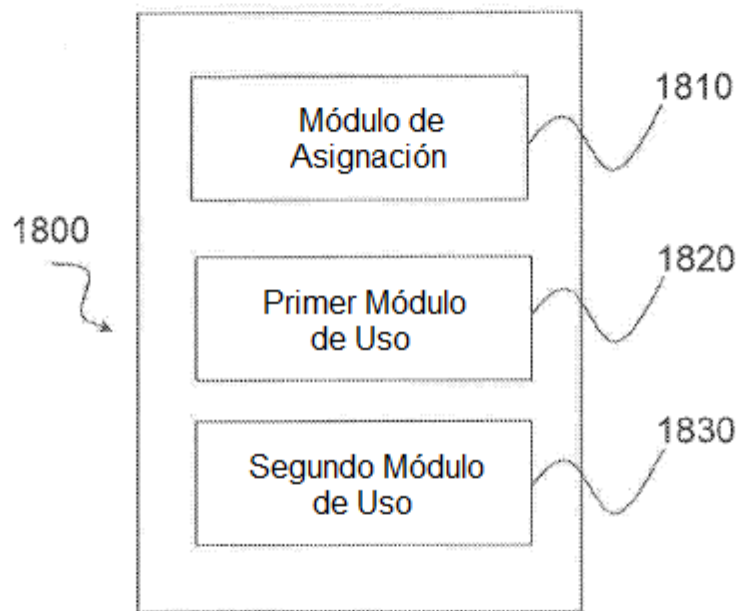


FIGURA 18