

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 391 808**

51 Int. Cl.:

H04L 1/16 (2006.01)

H04L 1/18 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **09743632 .3**

96 Fecha de presentación: **07.05.2009**

97 Número de publicación de la solicitud: **2294743**

97 Fecha de publicación de la solicitud: **16.03.2011**

54 Título: **Empaquetamiento de información ACK en un sistema de comunicación inalámbrica**

30 Prioridad:
07.05.2008 US 51296 P
05.05.2009 US 435717

45 Fecha de publicación de la mención BOPI:
30.11.2012

45 Fecha de la publicación del folleto de la patente:
30.11.2012

73 Titular/es:
QUALCOMM INCORPORATED (100.0%)
International IP Administration 5775 Morehouse
Drive
San Diego, California 92121, US

72 Inventor/es:
SARKAR, SANDIP

74 Agente/Representante:
FÀBREGA SABATÉ, Xavier

ES 2 391 808 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

Empaquetamiento de información ACK en un sistema de comunicación inalámbrica

ANTECEDENTES

I. Campo

5 La presente descripción se refiere en general a la comunicación, y más específicamente a técnicas para enviar asentimientos (ACK) de información en un sistema de comunicación inalámbrica.

II. Antecedentes

10 Los sistemas de comunicación inalámbrica están ampliamente desplegados para proporcionar diferentes contenidos de comunicación, tales como voz, vídeo, datos por paquetes, mensajería, difusión, etc. Estos sistemas inalámbricos pueden ser sistemas de acceso múltiple capaces de soportar múltiples usuarios que comparten los recursos de sistema disponibles. Ejemplos de tales sistemas de acceso múltiple incluyen sistemas de Acceso Múltiple por División de Código (CDMA), sistemas de Acceso Múltiple por División de Tiempo (TDMA), sistemas de Acceso Múltiple por División de Frecuencia (FDMA), sistemas de FDMA ortogonal (OFDMA), y sistemas FDMA de portadora única (SC-FDMA).

15 En un sistema de comunicación inalámbrica, una estación base puede comunicarse con un equipo de usuario (UE) en el enlace descendente y en el enlace ascendente. El enlace descendente (o enlace directo) se refiere al enlace de comunicación desde la estación base al UE, y el enlace ascendente (o enlace inverso) se refiere al enlace de comunicación desde el UE a la estación base. La estación base puede enviar datos al UE. El UE puede recibir y procesar los datos desde la estación base y puede enviar información de ACK a la estación base. La estación base puede determinar si volver a enviar los datos o enviar nuevos datos al UE en base a la información de ACK. Es deseable enviar de manera eficiente la información ACK.

D Tech, "ACK/NACK multiplexing in PUSCH in TDD"; Nokia et al, "Downlink scheduling for ACK/NACK bundling" y LG Electronics, "Bundled ACK/NACK in TDD" se dirigen a concesiones de enlace descendente perdidas o asignaciones en el contexto de ACK/NACK.

25 Ericsson, "Details of ACK/NAK bundling for TDD" se refiere a la asociación entre auxiliares subtramas DL y subtramas UL.

Huawei, "Consideration on AKC/NAK bundling and Multi-ACK/NACK multiplexing in TDD" se refiere a colisiones entre ACK/NACK empaquetados y CQI o SRI.

30 LG Electronics, "CCE to bundled ACK/NACK index mapping in TDD" se refiere a procedimientos de asignación de recursos ACK/NACK de enlace ascendente relacionados con los elementos del canal de control (CCE).

35 Texas Instruments, "ACK/NAK DTX Detection with ACK/NAK Bundling in TDD" está dirigida a empaquetamiento de ACK/NAK en TDD. Para cada transmisión planificada de forma dinámica en las subtramas DL, se necesitan transmitir bits ACK/NAK en una subtrama UL asociada para permitir el funcionamiento correcto de ARQ híbrido (HARQ). Si un UE está planificado en un múltiplo de subtramas DL, las cuales están todas asociadas con una subtrama UL, el UE tiene que transmitir múltiples bits ACK/NAK en esa subtrama UL. Para reducir el número de bits ACK/NAK un UE necesita transmitir ACK/NAK empaquetados. Típicamente, un UE comprime los múltiples bits ACK/NAK en 1 o 2 bits ACK/NAK, que son posteriormente transmitidos en la subtrama UL correspondiente. Una regla común de empaquetado de ACK/NAK es que si todos los bits ACK/NAK son "ACK", entonces se transmite un bit "ACK" en la subtrama UL, para asentar todos los paquetes recibidos en múltiples subtramas DL. De lo contrario, se transmite un bit "NAK" en la subtrama UL para solicitar la retransmisión de todos los paquetes DL.

US 2007/0195740 se refiere a técnicas de transmisión para los paquetes de enlace directo que pueden permitir paquetes que ocupen menos de una ranura convencional.

45 Ericsson, "Combination of ACK/NACKs for TDD" se dirige también a empaquetar ACK/NACK en TDD. D₁₅ propone asociar cada subtrama DL con una subtrama UL. Cada subtrama de UL se asocia entonces con K subtramas, donde K puede ser cero, uno o hasta nueve en función de la configuración UL-DL actual. Informes ACK/NACK desde el conjunto de subtramas DL asociadas se combinan para generar un único informe ACK/NACK para las K subtramas DL asociadas. El único informe ACK/NACK se genera como sigue: Se genera un ACK si todas las subtramas DL asignadas son ACK. Un NACK se genera si la totalidad o al menos una de las subtramas DL asignadas es NACK y el resto son ACK. De lo contrario, un NACK o DTX se genera.

50

RESUMEN

El deseo anteriormente mencionado se cumple mediante la materia se describen de las reivindicaciones independientes. Las realizaciones ventajosas están contenidas en las reivindicaciones dependientes.

5 En la presente memoria se describen técnicas para empaquetar información ACK en un sistema de comunicación inalámbrico. Las técnicas pueden utilizarse para reducir la cantidad de información ACK a notificar y puede ser especialmente aplicable en un sistema de duplexación por división de tiempo (TDD) con una configuración asimétrica enlace descendente - enlace ascendente.

10 En un diseño de transmisión de datos en el enlace descendente, un UE puede recibir múltiples palabras código en al menos una subtrama de enlace descendente. Cada palabra código puede ser codificada por separado por una estación base y puede ser decodificada por separado por el UE. El UE puede decodificar las múltiples palabras código y puede determinar un asentimiento (ACK) o un asentimiento negativo (NACK) para cada palabra código basada en el resultado de la decodificación de la palabra código. La UE puede empaquetar los ACKs y NACKs de las múltiples palabras código para obtener información ACK empaquetada. El UE puede enviar la información ACK empaquetada como retroalimentación para las múltiples palabras código.

15 En un diseño, la información ACK empaquetada puede comprender un único ACK/NACK empaquetado para las múltiples palabras código. El UE puede realizar el empaquetado con una operación AND lógica en los ACKs y NACKs para las múltiples palabras código. El UE puede generar (i) un ACK empaquetado si se obtienen ACKs para todas las palabras código o (ii) un NACK empaquetado si se obtiene un NACK para cualquier palabra código. El UE puede recibir retransmisiones de las múltiples palabras código si se envía un NACK empaquetado a la estación base y puede recibir palabras código nuevas si se envía un ACK empaquetado.

20 En otro diseño, la información de ACK empaquetada puede comprender múltiples ACKs/NACKs empaquetados para varios conjuntos de palabras código formadas con las múltiples palabras código. Cada ACK/NACK empaquetado puede comprender un ACK empaquetado o un NACK empaquetado para un conjunto de palabras código. El UE puede determinar los ACK/NACK empaquetados para cada conjunto de palabras código en base a los ACKs y NACKs para las palabras código en ese conjunto. El UE puede recibir la retransmisión de cada conjunto de palabras código para los que se envía un NACK empaquetado. El UE puede recibir un nuevo conjunto de palabras código para cada conjunto de palabras código para las que se envía un ACK empaquetado.

Varios aspectos y características de la divulgación se describen en mayor detalle más adelante.

30 **BREVE DESCRIPCIÓN DE LOS DIBUJOS**

La Figura 1 muestra un sistema de comunicación inalámbrica.

La Figura 2 muestra una estructura de trama de ejemplo para un sistema TDD.

La Figura 3 muestra una transmisión de datos de ejemplo en el enlace descendente con HARQ.

La Figura 4 muestra un proceso para recibir datos.

35 La Figura 5 muestra un aparato para recibir de datos.

La Figura 6 muestra un procedimiento para enviar datos.

La Figura 7 muestra un aparato para enviar datos.

La Figura 8 muestra un diagrama de bloques de una estación base y un UE.

DESCRIPCIÓN DETALLADA

40 Las técnicas descritas en este documento pueden usarse para diversos sistemas de comunicación inalámbricos, tales como CDMA, TDMA, FDMA, OFDMA, SC-FDMA y otros sistemas. Los términos "sistema" y "red" se usan indistintamente. Un sistema CDMA puede implementar una tecnología radio, tal como acceso universal de radio terrestre (UTRA), CDMA2000, etc.: UTRA incluye CDMA de banda ancha (WCDMA) y otras variantes del CDMA. CDMA2000 cubre los estándares IS-2000, 95-ES y ES-856. Un sistema TDMA puede implementar una tecnología radio como el Sistema Global para Comunicaciones Móviles (GSM). Un sistema OFDMA puede implementar una tecnología radio como Evolved UTRA (E-UTRA), Ultra Mobile Broadband (UMB), IEEE 802.11 (Wi-Fi), IEEE 802.16 (WiMAX), IEEE 802.20, Flash OFDM®, etc. UTRA y E-UTRA son parte del Sistema Universal de Telecomunicaciones Móviles (UMTS). 3GPP Long Term Evolution (LTE) y LTE Advanced-(LTE-A) son los nuevos lanzamientos de UMTS que utilizan E-UTRA, que emplea a OFDMA en el enlace descendente y SC-FDMA en el

enlace ascendente. UTRA, E-UTRA, UMTS, LTE, LTE-A y GSM se describen en los documentos de una organización llamada "3rd Generation Partnership Project" (3GPP). cdma2000 y UMB se describen en los documentos de una organización llamada "3rd Generation Partnership Project 2" (3GPP2). Las técnicas descritas en este documento pueden ser utilizadas para los sistemas y tecnologías radio mencionados anteriormente, así como otros sistemas y tecnologías radio. Para mayor claridad, ciertos aspectos de las técnicas se describen a continuación para LTE y se utiliza terminología LTE en gran parte de la descripción a continuación.

Las técnicas descritas en este documento también pueden utilizarse para la transmisión de datos en el enlace descendente, así como en el enlace ascendente. Para mayor claridad, ciertos aspectos de las técnicas se describen a continuación para la transmisión de datos en el enlace descendente, con la información ACK siendo enviada en el enlace ascendente.

La Figura 1 muestra un sistema de comunicación inalámbrico 100, que puede ser un sistema LTE o algún otro sistema. El sistema 100 puede incluir un número de Nodos B evolucionados (eNBs) y otras entidades de red 110. Un eNB puede ser una estación que se comunica con los UEs y también puede denominarse nodo B, estación base, punto de acceso, etc. Los UEs 120 pueden estar dispersos por todo el sistema, y cada UE puede ser estacionario o móvil. Un UE también puede denominarse estación móvil, terminal, terminal de acceso, unidad de abonado, estación, etc. Un UE puede ser un teléfono móvil, un asistente digital personal (PDA), un módem inalámbrico, un dispositivo de comunicación inalámbrica, un dispositivo portátil, un ordenador portátil, un teléfono inalámbrico, una estación de bucle local inalámbrico (WLL), etc.

El sistema puede utilizar duplexación por división de tiempo (TDD). Para TDD, el enlace descendente y el enlace ascendente comparten el mismo canal de frecuencia, que puede ser utilizado para el enlace descendente parte del tiempo y para el enlace ascendente en otro momento.

La Figura 2 muestra una estructura de trama de ejemplo 200 que puede utilizarse para un sistema TDD. La línea de tiempo de transmisión puede dividirse en unidades de tramas de radio. Cada trama de radio pueden tener una duración predeterminada, por ejemplo, 10 milisegundos (ms), y puede ser dividida en dos medias tramas. Cada trama de radio también puede ser dividida en 10 subtramas con índices de 0 a 9. Cada subtrama utilizable para la transmisión de datos puede dividirse en dos ranuras. Cada ranura puede incluir Q períodos de símbolo, por ejemplo, Q = 6 períodos de símbolo para un prefijo cíclico prolongado o Q = 7 períodos de símbolo para un prefijo cíclico normal. Se puede enviar un símbolo OFDMA o un símbolo SC-FDMA en cada periodo de símbolo.

La Tabla 1 enumera siete configuraciones de enlace descendente - enlace ascendente compatibles con LTE para TDD. Las subtramas 0 y 5 pueden utilizarse para el enlace descendente para todas las configuraciones enlace descendente - enlace ascendente y se denotan "DL" en la Figura 2 y "D" en la Tabla 1. La subtrama 2 se puede usar para el enlace ascendente para todas las configuraciones de enlace descendente - enlace ascendente y se denota "UL" en la Figura 2 y "U" en la Tabla 1. Las subtramas 3, 4, 7, 8 y 9 pueden utilizarse cada una para el enlace descendente o para el enlace ascendente en función de la configuración de enlace descendente - enlace ascendente. La subtrama 1 puede ser una subtrama especial (indicada como "S" en la Tabla 1) con tres campos especiales compuestos de ranura de tiempo de piloto de enlace descendente (DwPTS), un periodo de guarda (GP), y una ranura de tiempo de piloto de enlace ascendente (UpPTS). La subtrama 6 puede ser (i) una subtrama especial con solamente los DwPTS o los tres campos especiales o (ii) una subtrama de enlace descendente, en función de la configuración de enlace descendente - enlace ascendente. Los DwPTS, GP y campos DwPTS pueden tener duraciones diferentes para diferentes configuraciones de subtramas especiales.

Una subtrama utilizada para el enlace descendente puede denominarse subtrama de enlace descendente. Una subtrama utilizada para el enlace ascendente puede denominarse subtrama de enlace ascendente. La tabla 1 muestra el número de subtramas de enlace descendente (D #), el número de subtramas de enlace ascendente (# U), y el número de subtramas especiales (# S) en cada trama de radio para cada configuración de enlace descendente - enlace ascendente. Se pueden enviar 2Q símbolos OFDM en cada subtrama de enlace descendente, y se pueden enviar 2T símbolos SC-FDMA en cada subtrama de enlace ascendente, tal y como se muestra en la Figura 2.

Tabla 1 - Configuraciones de enlace descendente - enlace ascendente

Configuración DL:UL	Periodicidad de punto de conmutación	Número de subtrama										#D	#U	#S	N: M
		0	1	2	3	4	5	6	7	8	9				
0	5 ms	D	S	U	U	U	U	S	U	U	U	2	6	2	1:3
1	5 ms	D	S	U	U	D	D	S	U	U	D	4	4	2	1:1

2	5 ms	D	S	U	D	D	D	S	U	D	D	6	2	2	3:1
3	10 ms	D	S	U	U	U	U	D	D	D	D	6	3	1	2:1
4	10 ms	D	S	U	U	D	D	D	D	D	D	7	2	1	7:2
5	10 ms	D	S	U	D	D	D	D	D	D	D	8	1	1	8:1
6	10 ms	D	S	U	U	U	U	S	U	U	D	3	5	2	3:5

Un número de configuraciones N:M puede estar permitido para el enlace descendente y el enlace ascendente. Para un configuración N: M dada, un ciclo de enlace descendente - enlace ascendente puede incluir N subtramas de enlace descendente y M subtramas de enlace ascendente, donde en general $N \geq 1$, $M \geq 1$, y N puede o no puede ser igual a M. Existe asimetría en el enlace descendente y en el enlace ascendente cuando N no es igual a M. Un ciclo de enlace descendente - enlace ascendente puede abarcar 5 ms para las configuraciones de enlace descendente - enlace ascendente 0 hasta 2 o puede abarcar 10 ms para las configuraciones de enlace descendente - enlace ascendente 3 hasta 6. La última columna de la Tabla 1 da la configuración N: M para cada configuración de enlace descendente - enlace ascendente. Las siguientes configuraciones N: M pueden ser compatibles:

- 5 • Simétrica: 1:1 - igual número de subtramas enlace descendente y subtramas de enlace ascendente,
- Enlace descendente fuerte: 2:1, 3:1, 7:2 y 8:1 - más subtramas de enlace descendente que subtramas de enlace ascendente, y
- Enlace ascendente fuerte: 1:3 y 3:5 - más subtramas de enlace ascendente que subtramas de enlace descendente.

15 Se puede obtener una configuración 9:1 para la configuración 5 de enlace descendente - enlace ascendente mediante la configuración de la subtrama especial 1 para incluir sobre todo DwPTS y los mínimos GP y UpPTS.

20 El sistema puede permitir solicitud de repetición automática híbrida (HARQ). Para HARQ en el enlace descendente, un eNB puede procesar un bloque de transporte (o paquete) para obtener una palabra código (o paquete codificado). El eNB puede entonces enviar una transmisión de la palabra código a un UE y puede enviar una o más transmisiones adicionales hasta que la palabra código es decodificada correctamente por el UE, o el número máximo de transmisiones ha sido enviado, o se encontró alguna otra condición de terminación. La primera transmisión de la palabra código puede denominarse nueva transmisión, y cada transmisión adicional de la palabra código puede denominarse retransmisión. Después de cada transmisión de la palabra código, el UE puede decodificar todas las transmisiones recibidas de la palabra código para intentar recuperar la palabra código.

25 La Figura 3 muestra la transmisión de datos a modo de ejemplo en el enlace descendente con HARQ. Cada ciclo de enlace descendente - enlace ascendente puede incluir un periodo de transmisión de enlace descendente que abarca N subtramas de enlace descendente y un periodo de transmisión de enlace ascendente que abarca M subtramas de enlace ascendente. En un diseño, un eNB puede enviar datos en hasta N subtramas de enlace descendente de un periodo de transmisión de enlace descendente, y un UE puede enviar información de ACK para los datos en una subtrama de enlace ascendente del siguiente periodo de transmisión de enlace ascendente.

30 El UE puede estimar periódicamente la calidad del canal de enlace descendente para el eNB y puede enviar la información de indicador de calidad del canal (CQI) sobre un canal físico de control de enlace ascendente (PUCCH) al eNB. El eNB puede usar la información de CQI y/u otra información para planificar el UE para transmisión de datos en el enlace descendente para seleccionar un esquema de modulación y codificación (MCS) para el UE. Para cada subtrama del enlace descendente en la que el UE está planificado, el eNB puede procesar L bloques de transporte (o paquetes) para obtener L palabras código, donde $L \geq 1$, y puede enviar las L palabras código en un canal físico compartido de enlace descendente (PDSCH) al UE. El eNB también puede enviar una asignación de enlace descendente para el UE sobre un canal físico de control de enlace descendente (PDCCH) en cada subtrama en la que el UE está planificado (tal y como se muestra en la Figura 3) o sólo en la primera subtrama. El eNB puede no enviar ninguna asignación de enlace descendente y ninguna palabra código al UE en cada subtrama del enlace descendente en la que el UE no está planificado.

35 El UE puede procesar el PDCCH en cada subtrama del enlace descendente para obtener una asignación de enlace descendente, en su caso, enviada al UE. Si se recibe una asignación de enlace descendente, entonces el UE puede procesar el PDSCH y decodificar las L palabras código enviadas al UE. Para cada palabra código, el UE puede proporcionar un ACK si la palabra código es decodificada correctamente o un NACK si una palabra código es decodificada erróneamente. El UE puede generar información de ACK sobre la base de los ACKs y NACKs para todas las palabras código, tal y como se describe a continuación, y puede enviar la información de ACK en el

PUCCH al eNB. El eNB puede enviar una retransmisión de cada palabra código para la que se recibe un NACK y puede enviar una transmisión de una nueva palabra código para cada palabra código para la que se recibe un ACK.

Se puede definir un número de procesos HARQ para cada uno del enlace descendente y el enlace ascendente. Un proceso HARQ puede transportar todas las transmisiones de una palabra código hasta que la palabra código es decodificada correctamente y luego puede transportar las transmisiones de otra palabra código. Una palabra código nueva puede ser enviada en un proceso HARQ cuando el proceso se vuelve disponible. El número de procesos HARQ para cada enlace puede ser dependiente de (i) el número de subtramas de enlace descendente y el número de subtramas de enlace ascendente en cada ciclo de enlace descendente - enlace ascendente y (ii) el tiempo de procesado requerido en un receptor para cada palabra código. Por ejemplo, si el tiempo de procesado requerido es de 3 ms, entonces una transmisión de datos pueden ser enviada en la subtrama n y el correspondiente ACK/NACK puede ser enviado en la subtrama $n + k$, donde $k > 3$. La Tabla 2 muestra el número de procesos HARQ para el enlace descendente y el número de procesos HARQ para el enlace ascendente para cada configuración de enlace descendente - enlace ascendente permitida en LTE.

Tabla 2 - Número de procesos HARQ

Configuración DL:UL	Asignación DL:UL	Número de procesos HARQ de enlace descendente	Número de procesos HARQ de enlace ascendente
0	1 DL + DwPTS : 3 UL	4	7
1	2 DL + DwPTS : 2 UL	7	4
2	3 DL + DwPTS : 1 UL	10	2
6	3 DL + 2DwPTS : 5 UL	6	6
3	6 DL + DwPTS : 3 UL	9	3
4	7 DL + DwPTS : 2 UL	12	2
5	8 DL + DwPTS : 1 UL	15	1

El número de ACKs y NACKs para enviar como retroalimentación puede depender de varios factores tales como el número de procesos HARQ a aserir, el número de palabras código enviadas en cada proceso HARQ, si aserir o no una asignación de enlace descendente, etc. En un diseño, un eNB puede enviar datos en un máximo de N procesos HARQ para el UE, un proceso HARQ en cada subtrama del enlace descendente. En un diseño, el eNB puede enviar (i) una palabra código en cada proceso HARQ con transmisión de entrada única - salida múltiple (SIMO) o (ii) múltiples palabras código en cada proceso HARQ con transmisión de entrada múltiple - salida múltiple (MIMO). Las palabras código enviadas a través de una transmisión MIMO se pueden asignar a las capas, y el número de capas puede ser mayor que o igual al número de palabras código. El eNB puede por tanto enviar una o más palabras código en cada proceso HARQ. Por ejemplo, se pueden enviar un máximo de dos palabras código por transmisión MIMO, y el UE puede recibir de cero a $2N$ palabras código en N subtramas de enlace descendente. El UE puede generar información de ACK para todas las palabras código y puede enviar la información de ACK en una subtrama de enlace ascendente del siguiente período de transmisión de enlace ascendente.

En un aspecto, un UE puede agrupar o combinar ACKs y NACKs para múltiples (K) palabras código y puede generar información de ACK empaquetada para todas las K palabras código. En un diseño, el UE puede llevar a cabo el empaquetado con una operación AND lógica en los ACKs y NACKs para todas las K palabras código. El UE puede generar (i) un ACK empaquetado si se obtienen ACKs para todas las K palabras código o (ii) un NACK empaquetado si se obtiene un NACK para una cualquiera de las K palabras código. La información ACK empaquetada para las K palabras código puede comprender un ACK empaquetado o un NACK empaquetado. El UE también puede realizar el empaquetado de otras maneras. El empaquetado de ACKs y NACKs puede reducir la cantidad de información de ACK empaquetada en un factor K .

Un eNB puede recibir información de ACK empaquetada para las K palabras código desde el UE. Si se recibe un ACK empaquetado, entonces el eNB puede enviar el siguiente conjunto de palabras código al UE. De lo contrario, si se recibe un NACK empaquetado, entonces el eNB puede reenviar todas las K palabras código ya que el eNB no sabe qué palabras código han sido recibidas erróneamente por el UE.

En un diseño, las K palabras código pueden ser enviadas en un máximo de K procesos HARQ que pueden iniciarse al mismo tiempo, por ejemplo, en diferentes subtramas de enlace descendente del mismo periodo de transmisión de

enlace descendente. Las K palabras código pueden ser entonces procesadas (por ejemplo, codificadas, intercaladas y moduladas) por el eNB de forma que tengan una conclusión objetivo similar. Conclusión objetivo se refiere a la cantidad de transmisiones de una palabra código necesarias para lograr una probabilidad objetivo de decodificación correcta de la palabra código. El eNB puede enviar un nuevo conjunto de palabras código cada vez que se recibe un ACK empaquetado.

El empaquetado puede realizarse de varias maneras. En un diseño, el empaquetado se puede realizar para ACKs y NACKs para todas las palabras código recibidas en una subtrama de enlace descendente, por ejemplo, para dos palabras código recibidas a través de una transmisión MIMO en una subtrama de enlace descendente. En otro diseño, el empaquetado se puede realizar por ACKs y NACKs para palabras código recibidas en las subtramas de enlace descendente múltiples, por ejemplo, una palabra código en cada subtrama del enlace descendente. En otro diseño, el empaquetado se puede realizar para ACKs y NACKs para palabras código recibidas en todas las subtramas de enlace descendente de un periodo de transmisión de enlace descendente. En otro diseño, el empaquetado se puede realizar para ACKs y NACKs para palabras código enviadas en la misma capa a lo largo de múltiples subtramas de enlace descendente. Como ejemplo, para una transmisión MIMO de dos palabras código en cada una de N subtramas de enlace descendente, puede generarse un ACK/NACK empaquetado para palabras código enviadas en la primera capa en las N subtramas de enlace descendente, y puede generarse otro ACK/NACK empaquetado para palabras código enviadas en la segunda capa en las N subtramas de enlace descendente. En general, el empaquetado se puede realizar para cualquier número de palabras código recibidas en un número de subtramas de enlace descendente, con o sin MIMO.

El UE puede recibir K_{total} palabras código en N subtramas de enlace descendente y puede enviar información de ACK empaquetada para estas K_{total} palabras código en una subtrama de enlace ascendente. En un diseño, el UE puede generar un solo ACK/NACK empaquetado para todas las K_{total} palabras código, y la información de ACK empaquetada puede comprender el único ACK/NACK empaquetado. En otro diseño, el UE puede generar varios ACKs/NACKs empaquetados para varios conjuntos de palabras código formadas con las K_{total} palabras código, un ACK/NACK empaquetado para cada conjunto de palabras código. La información ACK empaquetada puede entonces comprender los múltiples ACK/NACKs empaquetados. Por ejemplo, se pueden enviar nueve palabras código en nueve procesos HARQ en nueve subtramas de enlace descendente con la configuración 5 de enlace descendente - enlace ascendente de la Tabla 1. El UE puede generar tres ACKs/NACKs empaquetados, por ejemplo, un primer ACK/NACK empaquetado para las tres primeras palabras código, un segundo ACK/NACK empaquetado para las tres próximas palabras código, y un tercer ACK/NACK empaquetado para las tres últimas palabras código. Como otro ejemplo, se pueden enviar cuatro palabras código en cuatro procesos HARQ en cuatro subtramas de enlace descendente con la configuración 4 ó 5 de enlace descendente - enlace ascendente de la Tabla 1. El UE puede generar dos ACKs/NACKs empaquetados, por ejemplo, un primer ACK/NACK empaquetado para las dos primeras palabras código, y un segundo ACK/NACK empaquetado para las dos últimas palabras código.

En general, la información de ACK empaquetada puede comprender cualquier número de ACKs/NACKs empaquetados, y cada ACK/NACK empaquetado puede ser para cualquier número de palabras código. Si se envían múltiples ACKs/NACKs empaquetados, a continuación, cada ACK/NACK empaquetado puede cubrir el mismo número de palabras código, o diferentes ACKs/NACKs empaquetados pueden cubrir diferentes números de palabras código.

En un diseño, el empaquetado de los ACKs y NACKs puede ser estático o semi-estático y puede ser configurado, por ejemplo, por una capa superior, en el establecimiento de llamada. En otro diseño, el empaquetado de los ACKs y NACKs puede ser dinámico y puede ser configurado, por ejemplo, a través de señalización enviada en el PDCCH por la capa física o alguna otra capa. Para ambos diseños, el empaquetado puede ser dependiente de varios factores tales como la configuración de enlace descendente - enlace ascendente, la cantidad de datos a enviar en el enlace descendente al UE, si se utiliza o no MIMO para la transmisión de datos, los requisitos de calidad de servicio (QoS) de los datos a enviar, la cantidad de recursos disponibles para el envío de información de ACK en el enlace ascendente por los UE, el tiempo de procesado requerido por los UEs, etc. Por ejemplo, se puede utilizar progresivamente más empaquetado para configuraciones progresivamente más asimétricas de enlace descendente - enlace ascendente o configuraciones para un desequilibrio progresivamente mayor entre la cantidad de datos a enviar y la cantidad de recursos disponibles para la información de ACK.

En un diseño, el UE puede enviar información de ACK que comprenda uno o más ACKs/NACKs empaquetados para K_{total} palabras código. En otro diseño, el UE puede enviar información de ACK que comprenda una combinación de ACKs y NACKs empaquetados e individuales para K_{total} palabras código. Cada paquete ACK/NACK empaquetado puede cubrir varias palabras código entre las K_{total} palabras código. Cada ACK/NACK individual puede cubrir una sola palabra código entre las K_{total} palabras código. Por ejemplo, pueden utilizarse ACKs/NACKs individuales para ciertos tipos de datos (por ejemplo, datos sensibles a retardos, tales como voz), y se pueden usar ACKs/NACKs empaquetados para otros tipos de datos (por ejemplo, datos tolerantes a retardos).

El eNB puede codificar un bloque de transporte de bits de información de acuerdo con un código de corrección de errores (FEC) (por ejemplo, un código Turbo) para obtener una palabra código de bits sistemáticos y bits de paridad.

Los bits sistemáticos son bits de información en el bloque de transporte, y los bits de paridad son bits de redundancia generados por el código FEC. El eNB puede dividir la palabra código en múltiples (N_{RV}) bloques de código, que pueden ser asignados versiones de redundancia (RV) 0 hasta $N_{RV} - 1$. El primer bloque de código con RV 0 puede contener sólo o principalmente bits sistemáticos. Cada bloque de código posterior con RV > 0 puede contener sólo o en su mayor parte bits de paridad, con bloques de código diferentes que contienen diferentes bits de paridad. El eNB puede enviar una asignación de enlace descendente en el PDCCH y una transmisión en un bloque de código en el PDSCH al UE.

El UE puede procesar el PDCCH para obtener la asignación de enlace descendente. Si el UE recibe la asignación de enlace descendente, entonces el UE puede procesar el PDSCH de acuerdo con la asignación de enlace descendente para recuperar la palabra código enviada al UE. Si el UE no detecta la asignación de enlace descendente, entonces el UE puede omitir el procesamiento del PDSCH. El eNB puede enviar la asignación de enlace descendente en el PDCCH para el UE, pero el UE puede perder la asignación de enlace descendente (por ejemplo, decodificar la asignación de enlace descendente por error) y no procesar el PDSCH. Este escenario puede denominarse transmisión discontinua (DTX).

Puede ser deseable distinguir entre DTX y NACK. En este caso, la realimentación para la palabra código puede ser una de los siguientes:

- DTX → el UE perdió el PDCCH y no recibió la asignación de enlace descendente,
- ACK → la palabra código se descifró correctamente, y
- NACK → la palabra código se descifró erróneamente.

El eNB puede volver a enviar el bloque de código si se recibe un DTX desde el UE y puede enviar el siguiente bloque de código si se recibe un NACK. El eNB puede por lo tanto enviar bloques de código diferentes, dependiendo de si se recibe un DTX o un NACK es recibido. El eNB puede enviar un bloque de código para una nueva palabra código si se recibe un ACK.

El eNB puede enviar N bloques de código con RV 0 de N palabras código en el PDSCH en diferentes subtramas de enlace descendente y puede enviar una asignación de enlace descendente en el PDCCH en cada subtrama del enlace descendente, por ejemplo, tal y como se muestra en la Figura 3. Estos N bloques de código pueden incluir bits sistemáticos para las N palabras código. En cada subtrama del enlace descendente, el UE puede procesar el PDCCH para detectar una asignación de enlace descendente y puede procesar el PDSCH si se recibe una asignación de enlace descendente. El UE puede generar y enviar un ACK/NACK empaquetado para todas las palabras código recibidas por el UE. El eNB puede no saber cuántas asignaciones de enlace descendente son recibidas por el UE en base al ACK/NACK empaquetado. Si el eNB recibe un NACK empaquetado desde el UE, entonces el eNB puede tener las siguientes opciones:

1. Interpretar el NACK empaquetado como que comprende NACKs para todas las N palabras código y enviar N bloques de código con RV 1 al UE, o
2. Interpretar el NACK empaquetado como que comprende DTXs para todas las N palabras código y enviar de nuevo los N bloques de código con RV 0 al UE.

Si el eNB implementa la opción 1 y el UE perdió en realidad las asignaciones de enlace descendente, entonces el UE perderá los N bloques de código con RV 0 que contienen los bits sistemáticos y puede recibir los N bloques de código con RV 1 que contienen los bits de paridad. El rendimiento puede verse afectado negativamente mediante al decodificar las palabras código sólo con los bits de paridad.

Si el eNB implementa la opción 2 y el UE recibió en realidad las asignaciones de enlace descendente, entonces el UE puede recibir los N bloques de código con RV 0 que contienen los bits sistemáticos dos veces. El eNB transmite entonces las palabras código utilizando un código de repetición en lugar de un código Turbo, y el rendimiento puede degradarse sin la ganancia de codificación Turbo. La pérdida en la ganancia de codificación y la pérdida correspondiente en el rendimiento puede ser más grave cuando se empaquetan juntos ACKs y NACKs para un gran número de palabras código, por ejemplo, con la configuración de 9:1.

En otro aspecto, el eNB puede generar bloques de código de forma que tenga en cuenta la ambigüedad entre NACK y DTX debido a un empaquetado. En un diseño, para luchar contra la posible pérdida en la ganancia de codificación debido al empaquetado de los ACKs y NACKs, cada bloque de código de una palabra código puede ser definido para incluir tanto bits sistemáticos como de paridad. El eNB puede generar N_{RV} bloques de código con RV 0 hasta $N_{RV} - 1$. El primer bloque de código con RV 0 puede contener sólo o principalmente bits sistemáticos. Cada bloque de código posterior con RV > 0 puede contener tanto bits sistemáticos como de paridad, con diferentes bloques de código conteniendo diferentes bits sistemáticos y/o diferentes bits de paridad. El porcentaje de los bits sistemáticos en cada bloque de código con RV > 0 puede depender de la solución de compromiso en la degradación del

rendimiento para los dos casos antes descritos. El eNB puede entonces enviar un bloque de código diferente para cada transmisión de la palabra código. Si el UE perdió la asignación de enlace descendente, entonces el UE puede recibir los bits sistemáticos a partir de un bloque de código subsiguiente. Si el UE recibe la asignación de enlace descendente pero ha decodificado la palabra código erróneamente, entonces el UE puede recibir los bits de paridad del bloque de código subsiguiente.

LTE soporta un número de formatos de PUCCH para el envío de información de control de enlace ascendente (UCI) tal como la información de ACK en el PUCCH. La Tabla 3 muestra los formatos soportados por PUCCH LTE y proporciona el número de bits que pueden ser enviados en cada subtrama de enlace ascendente para cada formato PUCCH.

Tabla 3 - Formatos PUCCH

Formato PUCCH	Número de bits por subtrama
1a	1
1b	2
2	20
2a	21
2b	22

El formato PUCCH 1a puede ser utilizado para enviar un bit de información ACK para SIMO sin empaquetado, por ejemplo, un bit de ACK o NACK para una palabra código. El formato PUCCH 1a también se puede utilizar para enviar un ACK/NACK empaquetado para múltiples palabras código, por ejemplo, para todas las palabras código recibidas en una o más subtramas de enlace descendente de un periodo de transmisión de enlace descendente.

El formato PUCCH 1b puede ser utilizado para enviar dos bits de información de ACK para MIMO sin empaquetado, por ejemplo, un bit de ACK o NACK para cada una de dos palabras código enviadas con MIMO. El formato PUCCH 1b también se puede utilizar para enviar hasta dos ACKs/NACKs empaquetados para un máximo de dos conjuntos de palabras código, un ACK/NACK empaquetado para cada conjunto de palabras código. Cada grupo podrá comprender todas las palabras código enviadas con MIMO. Cada conjunto también puede incluir palabras código de enlace descendente enviadas en subtramas diferentes. La cantidad de paquetes puede ser dependiente de la configuración N:M, el número de subtramas de enlace descendente utilizadas para la transmisión de datos, el número de palabras código enviadas en cada subtrama del enlace descendente, si se utiliza o no MIMO, etc.

Los formatos PUCCH 2, 2a y 2b pueden ser utilizados para enviar más de dos bits de información de ACK. Por ejemplo, hasta 14 bits de información de ACK pueden ser enviados con formato PUCCH 2, 2a o 2b con una tasa de código 0,7 o inferior. Se pueden enviar menos o más bits de información con formatos PUCCH 2, 2a y 2b con velocidades de código más altas o más bajas.

En un diseño, la información de ACK para todos los procesos HARQ puede ser codificada conjuntamente. Para una transmisión SIMO de una palabra código por cada proceso HARQ, se pueden usar 14 bits de información para enviar:

- DTX, ACK, NACK o para cada uno de 8 procesos HARQ, o
- ACK o NACK para cada uno de los 14 procesos HARQ.

Se pueden utilizar tres valores para cada proceso HARQ para transmitir ACK o NACK para una palabra código o DTX. Se pueden utilizar un total de 38 valores para 8 procesos HARQ y puede ser transportado en 14 bits de información, donde $3^8 < 214$. Alternativamente, se pueden usar dos valores para cada proceso HARQ para transmitir ACK o NACK para una palabra código.

Para una transmisión MIMO de dos palabras codificadas por cada proceso HARQ, se pueden usar 14 bits de información para enviar:

- ACK o NACK para cada palabra código o DTX para cada uno de 6 procesos HARQ, o
- ACK o NACK para cada palabra código para cada uno de 7 procesos HARQ.

5 Se pueden utilizar cinco valores para cada proceso HARQ para transmitir ACK para ambas palabras código, NACKs para ambas palabras código, ACK para la primera palabra código y NACK para la segunda palabra código, NACK para la primera palabra código y ACK para la segunda palabra código, o DTX. Se pueden utilizar un total de 56 valores para 6 procesos HARQ y puede ser transportado en 14 bits de información, donde $5^6 < 2^{14}$.
 5 Alternativamente, se pueden utilizar cuatro valores para cada proceso HARQ y se puede transmitir ACK o NACK para cada palabra código.

10 En otro diseño, la información de ACK para cada proceso HARQ se puede enviar por separado. Para una transmisión SIMO de una palabra código por proceso HARQ, la información de ACK para cada proceso HARQ puede enviarse en un bit sin DTX o en dos bits con DTX. Para una transmisión MIMO de dos palabras codificadas por proceso HARQ, la información de ACK para cada proceso de HARQ puede enviarse en dos bits sin DTX o en tres bits con DTX.

15 También se pueden usar otros formatos PUCCH además de los que se muestran en la Tabla 3 para transportar información de ACK para uno o más ACKs/NACKs empaquetados. Las señales para los formatos PUCCH mostrados en la Tabla 3 pueden generarse tal y como se describe en 3GPP TS 36.211, titulada " Evolved Universal Terrestrial Radio Access (E-UTRA); Physical Channels and Modulation", que está disponible públicamente.

20 En un diseño, el empaquetado se puede lograr mediante la realización de operación lógica AND de los ACKs y NACKs para todas las palabras código a empaquetar, tal y como se describió anteriormente. En otro diseño, el empaquetamiento puede conseguirse mapeando ACKs, NACKs y DTX para todas las palabras código a B bits de información de ACK empaquetada en base a una asignación predeterminada, donde $B \geq 1$. Por ejemplo, se puede
 20 definir una tabla 3 x 3 con tres filas para DTX, ACK y NACK para una palabra código primero y tres columnas para DTX, ACK y NACK para una segunda palabra código. Cada una de las nueve entradas en la tabla puede ser marcada con uno de 2^B posibles valores para los B bits empaquetados. Por ejemplo, B puede ser igual a 2 para el formato PUCCH 1b, y cada entrada de la tabla puede marcarse con '00', '01', '10' o '11'. El mapeo puede definirse para conseguir el empaquetado deseado de ACKs, NACKs y DTX para las dos palabras código. Como otro ejemplo,
 25 se puede definir una tabla 3 x 3 x 3 para tres palabras código, y cada una de las 27 entradas en la tabla puede marcarse con uno de 2^B posibles valores para los B bits empaquetados. El mapeo puede definirse para conseguir el empaquetado deseado de ACKs, NACKs y DTX para las tres palabras código.

30 La Figura 4 muestra un diseño de un proceso 400 para recibir datos en un sistema de comunicación inalámbrica. El proceso 400 puede ser llevado a cabo por un receptor, que puede ser el UE para transmitir datos en el enlace descendente, una estación base/eNB para transmitir datos en el enlace ascendente, o alguna otra entidad.

35 El receptor puede recibir múltiples palabras código en al menos una subtrama, por ejemplo, recibir transmisiones de bloques de código de un RV particular para las múltiples palabras código (bloque 412). El receptor puede recibir las múltiples palabras código (i) en subtramas múltiples, una palabra código en cada subtrama, (ii) a través de una transmisión MIMO en una subtrama, (iii) a través de transmisiones MIMO en múltiples subtramas, o (iv) a través de una o más transmisiones en una o más subtramas. En un diseño de transmisión de datos en el enlace descendente, el receptor puede recibir las múltiples palabras código en al menos una subtrama de enlace descendente en un sistema TDD con una configuración de enlace descendente - enlace ascendente asimétrica que tiene más subtramas de enlace descendente que subtramas de enlace ascendente. En un diseño de transmisión de datos en el enlace ascendente, el receptor puede recibir las múltiples palabras código en al menos una subtrama de enlace ascendente en un sistema TDD con más subtramas de enlace ascendente que subtramas de enlace descendente. El receptor puede recibir múltiples palabras código en el al menos un proceso HARQ, con todas las transmisiones de cada palabra código enviándose en un proceso HARQ.

45 El receptor puede decodificar las múltiples palabras código (bloque 414) y puede determinar un ACK o un NACK para cada palabra código en base al resultado de la decodificación de la palabra código (bloque 416). El receptor puede empaquetar ACKs y NACKs por las múltiples palabras código para obtener información ACK empaquetada (bloque 418). El receptor puede enviar la información de ACK empaquetada como retroalimentación para las múltiples palabras código (bloque 420). El receptor puede enviar la información de ACK empaquetada en una subtrama de enlace ascendente asociada con al menos una subtrama de enlace descendente en la que se recibieron múltiples palabras código.

50 En un diseño, la información de ACK empaquetada puede comprender un único ACK/ NACK empaquetado para las múltiples palabras código. En un diseño del bloque 418, el receptor puede realizar el empaquetado en base a una operación lógica AND de los ACKs y NACKs para las múltiples palabras código. El receptor puede generar (i) un ACK empaquetado si se obtienen ACKs para todas las múltiples palabras código o (ii) un NACK empaquetado si se obtiene un NACK para una cualquiera de las múltiples palabras código. En un diseño, el receptor puede recibir las retransmisiones de las múltiples palabras código si se envía un NACK empaquetado para las múltiples palabras código (bloque 422). El receptor puede recibir palabras código si se envía un nuevo ACK empaquetado para
 55 múltiples palabras código (bloque 424).

En otro diseño del bloque 418, el receptor puede obtener ACK, NACK o DTX para cada una de las múltiples palabras código. El receptor puede entonces mapear los ACKs, NACKs y DTX para las múltiples palabras código a múltiples bits de la información de ACK empaquetada en base a un mapeado predeterminado. El receptor también puede realizar el empaquetado de otras maneras.

5 En otro diseño, la información de paquete ACK puede comprender múltiples paquetes de ACKs/NACKs para varios conjuntos de palabras código formadas con las múltiples palabras código. Cada paquete ACK NACK/ACK puede comprender un paquete o un paquete NACK para un conjunto de palabras código. Cada conjunto puede incluir (i) todas las palabras código recibidas en una subtrama, (ii) palabras código enviado en una capa de transmisiones MIMO recibidas en múltiples subtramas, o (iii) las palabras código recibida en una o más subtramas. El receptor puede determinar el NACK paquete ACK/para cada conjunto de palabras código sobre la base de los ACKs y NACKs para las palabras código en ese conjunto. El receptor puede recibir la retransmisión de cada conjunto de palabras código para la que se envía un NACK incluido. El receptor puede recibir un nuevo conjunto de palabras código para cada conjunto de palabras código para las que se envía un ACK empaquetado.

10 En un diseño, el receptor puede obtener una configuración estática o semi-estática para empaquetar ACKs y NACKs, por ejemplo, desde las capas superiores al inicio de una llamada. El receptor puede entonces realizar el empaquetado de acuerdo con la configuración estática o semi-estática. En otro diseño, el receptor puede obtener una configuración dinámica para empaquetar los ACKs y NACKs para las múltiples palabras código, por ejemplo, a través de señalización enviada con las palabras código. El receptor puede entonces realizar el empaquetado de acuerdo con la configuración dinámica.

15 La Figura 5 muestra un diseño de un aparato 500 para recibir datos en un sistema de comunicación inalámbrica. El aparato 500 incluye un módulo 512 para recibir múltiples palabras código en al menos una subtrama, un módulo 514 para decodificar las múltiples palabras código, un módulo 516 para determinar un ACK o un NACK para cada palabra código en base al resultado de la decodificación de la palabra código, un módulo 518 para empaquetar ACKs y NACKs para las múltiples palabras código para obtener información de ACK empaquetada, un módulo 520 para enviar información de paquete ACK como retroalimentación para las múltiples palabras código, un módulo 522 para recibir las retransmisiones de las múltiples palabras código si se envía un paquete NACK para las múltiples palabras código, y un módulo 524 para recibir nuevas palabras código, si se envía un paquete ACK para las múltiples palabras código.

20 La Figura 6 muestra un diseño de un proceso 600 para enviar datos en un sistema de comunicación inalámbrica. El proceso 600 puede ser realizado por un transmisor, que puede ser una estación base/eNB para la transmisión de datos en el enlace descendente, un UE para transmisión de datos en el enlace ascendente, o alguna otra entidad.

25 El transmisor puede enviar múltiples palabras código en al menos una subtrama a un receptor (bloque 612). El transmisor puede enviar múltiples palabras código (i) en subtramas múltiples, una palabra código en cada subtrama, (ii) a través de una transmisión MIMO en una subtrama, (iii) a través de transmisiones MIMO en múltiples subtramas, o (iv) a través de una o más transmisiones en una o más subtramas.

30 El transmisor puede recibir información de ACK empaquetada generada por el receptor en base a ACKs y NACKs para las múltiples palabras código (bloque 614). En un diseño, el transmisor puede enviar las múltiples palabras código en al menos una subtrama de enlace descendente y puede recibir la información de ACK empaquetada en una subtrama de enlace ascendente en un sistema TDD con una configuración de enlace descendente - enlace ascendente asimétrica que tiene más subtramas de enlace descendente que subtramas de enlace ascendente. En otro diseño, el transmisor puede enviar las múltiples palabras código en al menos una subtrama de enlace ascendente y puede recibir la información de ACK empaquetada en una subtrama de enlace descendente en un sistema TDD con más subtramas de enlace ascendente que subtramas de enlace descendente.

35 El transmisor puede determinar si volver a enviar las múltiples palabras código o enviar nuevas palabras código en base a la información de ACK empaquetada (bloque 616). En un diseño, la información de ACK empaquetada puede comprender un único ACK/NACK empaquetado, que puede ser obtenido por el receptor sobre la base de una operación lógica AND de los ACKs y NACKs para las múltiples palabras código. El transmisor puede reenviar las múltiples palabras código si se recibe un paquete NACK para las múltiples palabras código (bloque 618). En un diseño, cada retransmisión de una palabra código puede comprender bits sistemáticos y bits de paridad de la palabra código con el fin de mitigar la degradación del rendimiento debido a la ambigüedad entre NACK y DTX con el empaquetado. El transmisor puede enviar nuevas palabras código si se recibe un paquete ACK para las múltiples palabras código (bloque 620).

40 En otro diseño, la información de ACK empaquetada puede comprender múltiples ACKs/NACKs empaquetados para varios conjuntos de palabras código formadas con las múltiples palabras código. El transmisor puede reenviar cada conjunto de palabras código para las cuales se recibe un NACK empaquetado. El transmisor puede enviar un nuevo conjunto de palabras código para cada conjunto de palabras código para las que se recibe un ACK empaquetado.

La Figura 7 muestra un diseño de un aparato 700 para recibir datos en un sistema de comunicación inalámbrica. El aparato 700 incluye un módulo 712 para enviar múltiples palabras código en al menos una subtrama a un receptor, un módulo 714 para recibir información de ACK empaquetada generada por el receptor en base a ACKs y NACKs para las múltiples palabras código, un módulo 716 para determinar si volver a enviar la múltiples palabras código o enviar nuevas palabras código en base a la información de ACK empaquetada, un módulo 718 para reenviar las múltiples palabras código si se recibe un paquete NACK por las múltiples palabras código, y un módulo 720 para enviar nuevas palabras código si se recibe un paquete para las múltiples palabras código.

Los módulos en las Figuras 5 y 7 pueden comprender procesadores, dispositivos electrónicos, dispositivos hardware, componentes electrónicos, circuitos lógicos, memorias, códigos software, códigos firmware, etc., o cualquier combinación de los mismos.

La Figura 8 muestra un diagrama de bloques de un diseño de una estación base/eNB 110 y un UE 120, que puede ser una de las estaciones base/eNBs y uno de los UEs en la Figura 1. La estación base 110 puede estar equipada con T antenas 832a hasta 832t, y el UE 120 puede estar equipado con R antenas 852a hasta 852r, donde en general $T \geq 1$ y $R \geq 1$.

En la estación base 110, un procesador de transmisión 820 puede recibir datos de uno o más UEs de una fuente de datos 812, procesar (por ejemplo, codificar y modular) los datos para cada UE, y proporcionar símbolos de datos para todos los UE. El procesador de transmisión 820 también puede recibir información de control (por ejemplo, asignaciones de enlace descendente) de un controlador/procesador 840, procesar la información de control, y proporcionar símbolos de control. El procesador de transmisión 820 puede también generar símbolos de referencia para señales de referencia y pueden multiplexar los símbolos de referencia con los símbolos de datos y los símbolos de control. Un procesador MIMO 822 puede procesar (por ejemplo, precodificar) los símbolos del procesador de transmisión 820 (si es aplicable) y proporcionar T flujos de símbolos de salida a T moduladores (MOD) 830a hasta 830t. Cada modulador 830 puede procesar su flujo de símbolos de salida (por ejemplo, para OFDM) para obtener un flujo de muestras de salida. Cada modulador 830 puede además acondicionar (por ejemplo, convertir a analógico, filtrar, amplificar y convertir de forma ascendente) su flujo de muestras de salida para generar una señal de enlace descendente. Se pueden transmitir T señales de enlace descendente de los moduladores 830a hasta 830t a través de T antenas 832a hasta 832t, respectivamente.

En el UE 120, las antenas 852a hasta 852r pueden recibir las señales de enlace descendente de la estación base 110. Cada antena 852 puede proporcionar una señal recibida a un demodulador asociado (DEMOD) 854. Cada demodulador 854 puede acondicionar (por ejemplo, filtrar, amplificar, convertir de forma descendente y digitalizar) su señal recibida para obtener muestras de entrada y puede además procesar las muestras de entrada (por ejemplo, para OFDM) para obtener símbolos recibidos. Un detector MIMO 856 puede realizar detección MIMO de los símbolos recibidos desde todos los R demoduladores 854a hasta 854r y proporcionar símbolos detectados. Un procesador de recepción 860 puede procesar (por ejemplo, demodular y decodificar) los símbolos detectados, proporcionar datos decodificados para el UE 120 a un sumidero de datos 862, y proporcionar la información de control decodificada a un controlador/procesador 870.

En el UE 120, los datos de una fuente de datos 878 y la información de control (por ejemplo, información ACK empaquetada) del controlador/procesador 870 puede ser procesada por un procesador 880 de transmisión y precodificada por un procesador MIMO 882 (si es aplicable) para obtener R flujos de símbolos de salida. Los R moduladores 854a hasta 854r pueden procesar los R flujos de símbolos de salida (por ejemplo, para el SC-FDM) para obtener R flujos de muestras de salida y puede además acondicionar los flujos de muestras de salida para obtener R señales de enlace ascendente, que pueden ser transmitidas a través de R antenas 852a hasta 852r. En la estación base 110, las señales de enlace ascendente desde el UE 120 pueden ser recibidas por las antenas 832a hasta 832t, acondicionadas y procesadas por los demoduladores 830a hasta 830t, y procesadas posteriormente por un detector MIMO 836 (si es aplicable) y un procesador de recepción 838 para recuperar los datos y la información de control enviada por el UE 120. El procesador de recepción 838 puede proporcionar datos decodificados a un sumidero de datos 839 y proporcionar información de control decodificada al controlador/procesador 840.

Los controladores/procesadores 840 y 870 pueden dirigir la operación en la estación base 110 y el UE 120, respectivamente. El procesador 840 y/o otros procesadores y módulos en la estación base 110 pueden realizar o dirigir el proceso 400 en la Figura 4 para transmitir datos en el enlace ascendente, el proceso 600 en la Figura 6 para transmitir datos en el enlace descendente, y/u otros procesos para las técnicas descritas en la presente memoria. El procesador 840 y/u otros procesadores pueden empaquetar ACKs y NACKs para transmitir de datos en el enlace ascendente. El procesador 870 y/u otros procesadores y módulos en el UE 120 puede realizar o dirigir el proceso 400 en la Figura 4 para transmitir datos en el enlace descendente, el proceso 600 en la Figura 6 para transmitir datos en el enlace ascendente, y/u otros procesos para las técnicas descritas en la presente memoria. El procesador 870 y/u otros procesadores pueden empaquetar ACKs y NACKs para transmitir datos en el enlace descendente. Las memorias 842 y 872 pueden almacenar datos y códigos de programa para la estación base 110 y el UE 120, respectivamente. Un programador puede programar los UEs 844 para transmitir datos en el enlace descendente y/o ascendente y puede asignar recursos a los UE planificados.

Los expertos en la técnica entenderán que la información y las señales pueden representarse utilizando cualquiera de una variedad de tecnologías y técnicas diferentes. Por ejemplo, los datos, instrucciones, comandos, información, señales, bits, símbolos, y chips a los que se puede hacer referencia a lo largo de la descripción anterior pueden ser representados mediante voltajes, corrientes, ondas electromagnéticas, campos magnéticos o partículas, campos ópticos o partículas, o cualquier combinación de los mismos.

Los expertos apreciarán además que los diversos bloques lógicos, módulos, circuitos y etapas de algoritmos descritos en conexión con la descripción de este documento pueden implementarse como hardware electrónico, software informático, o combinaciones de ambos. Para ilustrar claramente esta intercambiabilidad de hardware y software, diversos componentes, bloques, módulos, circuitos y etapas han sido descritos anteriormente generalmente en términos de su funcionalidad. Si tal funcionalidad se implementa como hardware o software depende de la aplicación en particular y de limitaciones de diseño impuestas sobre el sistema global. Los expertos pueden implementar la funcionalidad descrita de diversas maneras para cada aplicación particular, pero tales decisiones de implementación no deberían interpretarse como causantes de un alejamiento del alcance de la presente descripción.

Los diversos bloques lógicos, módulos y circuitos descritos en conexión con la descripción de este documento pueden implementarse o realizarse con un procesador de propósito general, un procesador digital de señal (DSP), un circuito integrado de aplicación específica (ASIC), una matriz programable de puertas (FPGA) u otro dispositivo lógico programable, puerta discreta o transistor, componentes de hardware discretos, o cualquier combinación de los mismos diseñada para realizar las funciones descritas en la presente memoria. Un procesador de propósito general puede ser un microprocesador, pero de forma alternativa, el procesador puede ser cualquier procesador, controlador, microcontrolador, o máquina de estados. Un procesador también puede implementarse como una combinación de dispositivos informáticos, por ejemplo, una combinación de un DSP y un microprocesador, una pluralidad de microprocesadores, uno o más microprocesadores en conjunción con un núcleo DSP, o cualquier otra configuración de ese tipo.

Las etapas de un procedimiento o algoritmo descritas en conexión con la descripción de este documento pueden realizarse directamente en hardware, en un módulo software ejecutado por un procesador, o en una combinación de los dos. Un módulo software puede residir en memoria RAM, memoria flash, memoria ROM, memoria EPROM, memoria EEPROM, registros, disco duro, un disco extraíble, un CD-ROM, o cualquier otra forma de medio de almacenamiento conocido en la técnica. Un medio de almacenamiento de ejemplo se acopla al procesador de forma que el procesador puede leer información de, y escribir información en, el medio de almacenamiento. De forma alternativa, el medio de almacenamiento puede estar integrado en el procesador. El procesador y el medio de almacenamiento pueden residir en un ASIC. El ASIC puede residir en un terminal de usuario. De forma alternativa, el procesador y el medio de almacenamiento pueden residir como componentes discretos en un terminal de usuario.

En uno o más diseños de ejemplo, las funciones descritas pueden implementarse en hardware, software, firmware, o cualquier combinación de los mismos. Si se implementan en software, las funciones pueden ser almacenadas o transmitidas en el control como una o más instrucciones o código en un medio legible por ordenador. Medios legibles por ordenador incluyen tanto medios de almacenamiento informáticos como medios de comunicación incluyendo cualquier medio que facilite la transferencia de un programa informático de un lugar a otro. Un medio de almacenamiento puede ser cualquier medio disponible al que se pueda acceder mediante un ordenador de propósito general o de propósito especial. A modo de ejemplo, y no limitante, tales medios legibles por ordenador pueden comprender RAM, ROM, EEPROM, CD-ROM u otro almacenamiento en disco óptico, almacenamiento en disco magnético u otros dispositivos de almacenamiento magnéticos, o cualquier otro medio que pueda ser utilizado para transportar o almacenar medios de código de programa deseados en forma de instrucciones o estructuras de datos y a los que se puede acceder mediante un ordenador de propósito general o de propósito especial, o un procesador de propósito general o de propósito especial. Además, cualquier conexión se denomina correctamente medio legible por ordenador. Por ejemplo, si el software se transmite desde un sitio web, servidor u otra fuente remota mediante un cable coaxial, cable de fibra óptica, par trenzado, línea de abonado digital (DSL), o tecnologías inalámbricas tales como infrarrojo, radio, microondas entonces el cable coaxial, cable de fibra óptica, par trenzado, la DSL o las tecnologías inalámbricas tales como infrarrojos, radio y microondas se incluyen en la definición de medio. Disco (del inglés disc) y disco (del inglés disk), tal y como se usan aquí, incluye disco compacto (CD), disco láser, disco óptico, discos versátiles digitales (DVD), disquete y disco Blu-ray en donde los discos (del inglés disk) por lo general reproducen datos magnéticamente, mientras los discos (del inglés disc) reproducen datos de forma óptica con láser. Las combinaciones de los anteriores también deberían incluirse dentro del alcance de medios legibles por ordenador.

La descripción anterior de la divulgación se proporciona para permitir a cualquier persona experta en la técnica llevar a cabo o usar la divulgación. Diversas modificaciones de la divulgación serán fácilmente evidentes para los expertos en la técnica, y los principios genéricos definidos aquí pueden aplicarse a otras variaciones sin alejarse del alcance de la descripción.

REIVINDICACIONES

1. Un procedimiento (400) para recibir datos en un sistema de comunicación inalámbrica, que comprende:
 recibir (412) múltiples palabras código en al menos una subtrama;
 5 decodificar (414) las múltiples palabras código;
 determinar (416) un asentimiento (ACK) o un asentimiento negativo (NACK) para cada palabra código en base al resultado de decodificación de la palabra código;
 10 empaquetar (418) ACKs y NACKs para las múltiples palabras código para obtener información de ACK empaquetada, en donde la información de ACK empaquetada comprende múltiples ACKs/NACKs empaquetados para varios conjuntos de palabras código formadas con las múltiples palabras código, cada paquete ACK/NACK comprendiendo un ACK empaquetado o un NACK empaquetado para un conjunto de palabras código; y
 enviar (420) la información de ACK empaquetada como retroalimentación para las múltiples palabras código.
2. El procedimiento (400) según la reivindicación 1, en donde las múltiples palabras código se reciben en al menos una subtrama de enlace descendente y la información de ACK empaquetada se envía en una subtrama de enlace ascendente en un sistema de duplexación por división de tiempo (TDD) con configuración de enlace descendente - enlace ascendente asimétrica que tiene más subtramas de enlace descendente que subtramas de enlace ascendente.
3. El procedimiento (400) según la reivindicación 1, en el que las múltiples palabras código se reciben en múltiples subtramas, una palabra código en cada subtrama.
4. El procedimiento (400) según la reivindicación 1, en el que las múltiples palabras código se reciben a través de una transmisión entrada múltiple - salida múltiple (MIMO) en una subtrama.
5. El procedimiento (400) según la reivindicación 1, en el que los ACKs y NACKs para empaquetar las palabras múltiples código comprende
 25 generar un ACK empaquetado si se obtienen ACKs para todas las múltiples palabras código, y
 generar un NACK empaquetado si se obtiene un NACK para una cualquiera de las múltiples palabras código.
6. El procedimiento (400) según la reivindicación 1, en el que la información de ACK empaquetada comprende un ACK o un NACK paquete integrado para las múltiples palabras código, el procedimiento comprende además:
 30 recibir (422) la retransmisión de las múltiples palabras código si un paquete NACK se envía para las múltiples palabras código, y
 recibir (424) palabras código si se envía un nuevo paquete ACK para las múltiples palabras código.
7. El procedimiento (400) según la reivindicación 1, en el que los ACKs y NACKs para empaquetamiento de las palabras código comprende múltiples
 35 obtener ACK, NACK o transmisión discontinua (DTX) para cada una de las múltiples palabras código, y
 mapear ACKs, NACKs y DTX para las múltiples palabras código a múltiples bits de la información de ACK empaquetada en base a una correlación predeterminada.
8. El procedimiento (400) según la reivindicación 1, en el que empaquetar (418) los ACKs y NACKs para las múltiples palabras código comprende determinar el ACK/NACK empaquetado para cada conjunto de palabras código en base a ACKs y NACKs para las palabras código en el conjunto de palabras código.
9. El procedimiento (400) según la reivindicación 1, que comprende además:
 40 recibir una retransmisión de cada conjunto de palabras código para el que se envía un NACK empaquetado, y
 recibir un nuevo conjunto de palabras código para cada conjunto de palabras código para las que se envía un ACK empaquetado.

10. El procedimiento (400) según la reivindicación 1, en el que cada conjunto de palabras código se recibe en una subtrama.
11. Un aparato (500) para comunicación inalámbrica, que comprende:
 medios (512) para recibir múltiples palabras código en al menos una subtrama;
 5 medios (514) para decodificar las múltiples palabras código;
 medios (516) para determinar un asentimiento (ACK) o un asentimiento negativo (NACK) para cada palabra código en base al resultado de la decodificación de la palabra código;
 10 medios (518) para empaquetar ACKs y NACKs para las múltiples palabras código para obtener información de ACK, en el que la información de ACK empaquetada comprende múltiples ACKs/NACKs empaquetados para varios conjuntos de palabras código formados con las múltiples palabras código, cada ACK/NACK empaquetado comprendiendo un ACK empaquetado o un NACK empaquetado para un conjunto de palabras código, y en donde los medios para empaquetar los ACKs y NACKs para las múltiples palabras código comprende medios para determinar el ACK/NACK empaquetado para cada conjunto de palabras código en base a ACKs y NACKs para las palabras código en el conjunto de palabras código; y
 15 medios (520) para enviar la información de ACK empaquetado como retroalimentación para las múltiples palabras código.
12. El aparato (500) según la reivindicación 11, que comprende además:
 20 al menos un procesador configurado para recibir múltiples palabras código en al menos una subtrama, para decodificar las múltiples palabras código, para determinar un asentimiento (ACK) o un asentimiento negativo (NACK) para cada palabra código en base al resultado de la decodificación de la palabra código, para empaquetar ACKs y NACKs para las múltiples palabras código para obtener información de ACK empaquetada, y para enviar información de ACK empaquetada como retroalimentación para las múltiples palabras código.
13. Un procedimiento (600) de envío de datos en un sistema de comunicación inalámbrica, que comprende:
 25 enviar (612) múltiples palabras código en al menos una subtrama a un receptor;
 recibir (614) información de asentimiento (ACK) empaquetada generada por el receptor en base a ACKs y asentimientos negativos (NACKs) para las múltiples palabras código, en donde la información de ACK empaquetada comprende múltiples ACKs/NACKs empaquetados para varios conjuntos de palabras código formados con las múltiples palabras código, cada ACK/NACK empaquetados comprendiendo un ACK empaquetado o un NACK empaquetado para un conjunto de palabras código; y
 30 determinar (616) si se deben reenviar las múltiples palabras código o enviar nuevas palabras código en base a la información de ACK empaquetada.
14. Un aparato para comunicación inalámbrica, que comprende:
 35 al menos un procesador configurado para enviar múltiples palabras código en al menos una subtrama a un receptor, para recibir información de asentimiento (ACK) empaquetada generada por el receptor en base a ACKs y asentimientos negativos (NACKs) para las múltiples palabras código, y para determinar si volver a enviar las múltiples palabras código o enviar nuevas palabras código en base a la información de ACK empaquetada, en donde la información de ACK empaquetada comprende múltiples ACKs/NACKs empaquetados para varios conjuntos de palabras código formados con las múltiples palabras código, cada ACK/NACK empaquetado comprendiendo un ACK empaquetado o un NACK empaquetado para un conjunto de palabras código, y en donde el al menos un procesador está configurado para reenviar cada conjunto de palabras código para las cuales se recibe un NACK empaquetado, y para enviar un nuevo conjunto de palabras código para cada conjunto de palabras código por el que un se recibe ACK empaquetado.
15. Un producto de programa de ordenador, que comprende:
 45 un medio legible por ordenador que comprende instrucciones que hacen que cuando se ejecuta, un ordenador lleve a cabo un procedimiento de una cualquiera de las reivindicaciones 1 a 10 y 13.

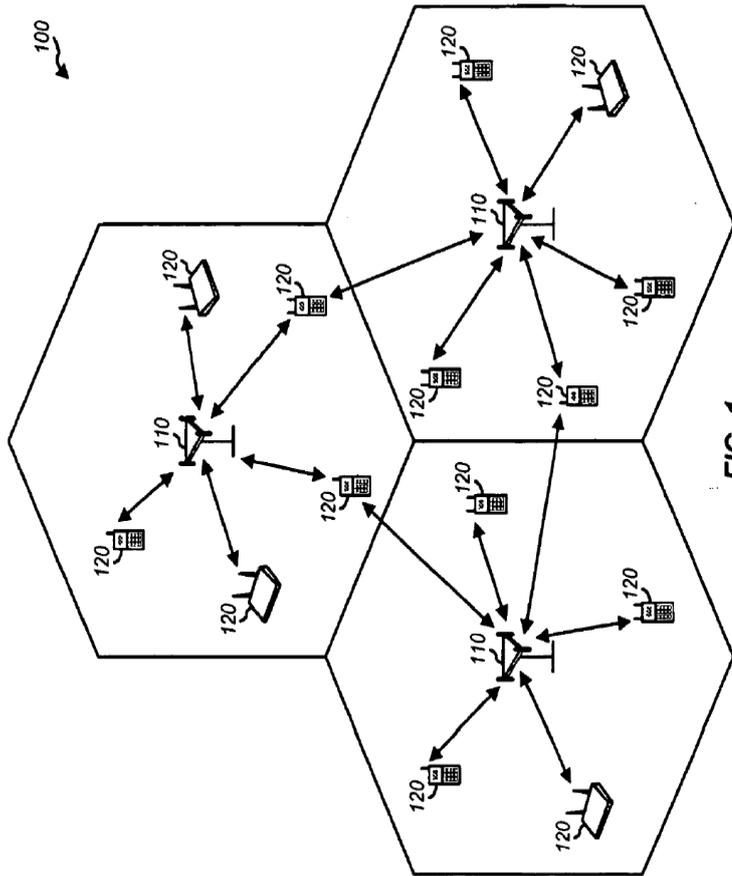


FIG. 1

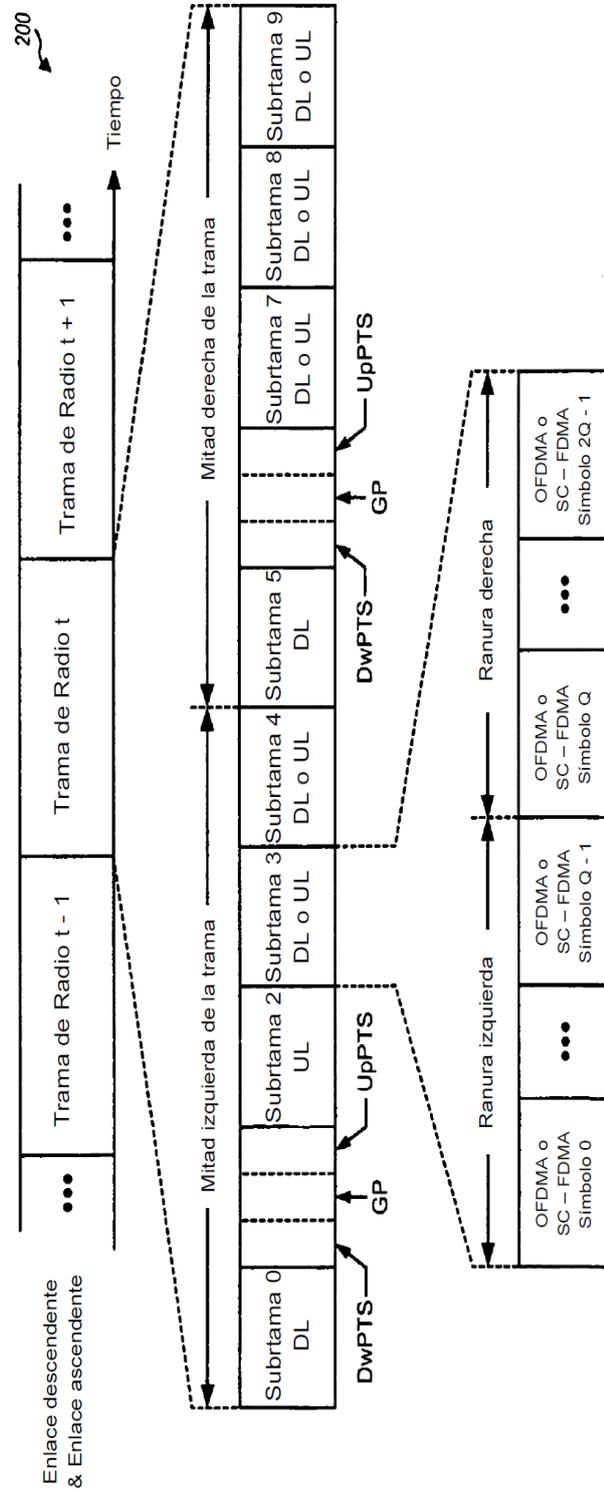


FIG. 2

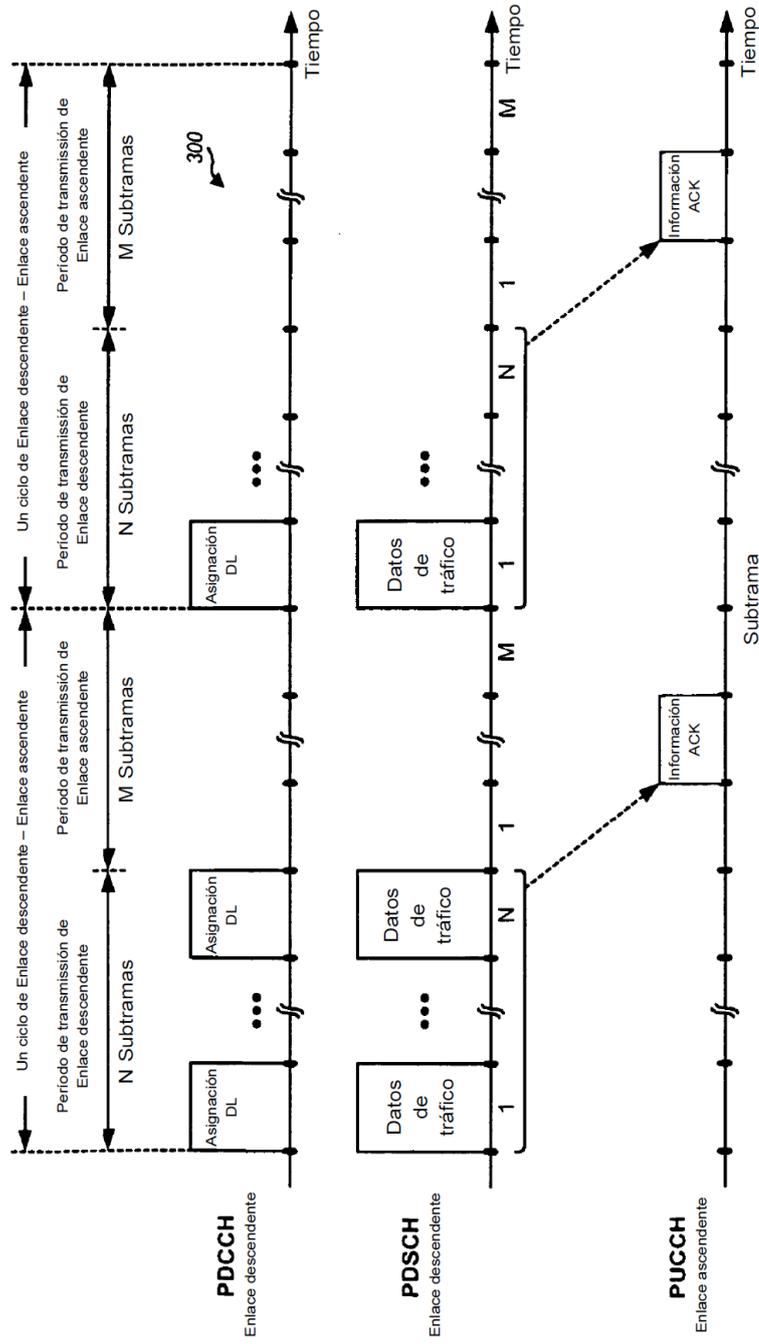


FIG. 3

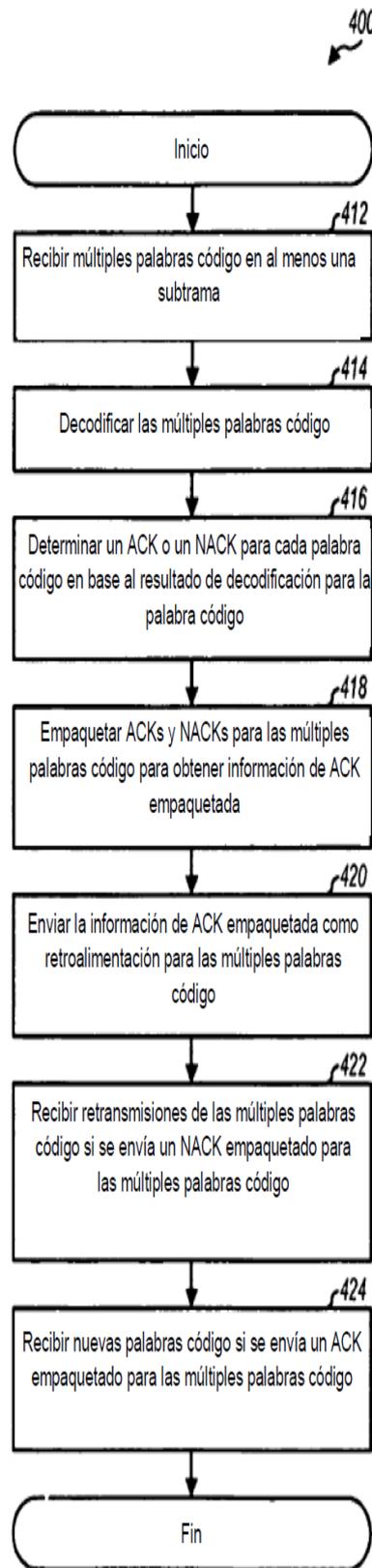


FIG. 4

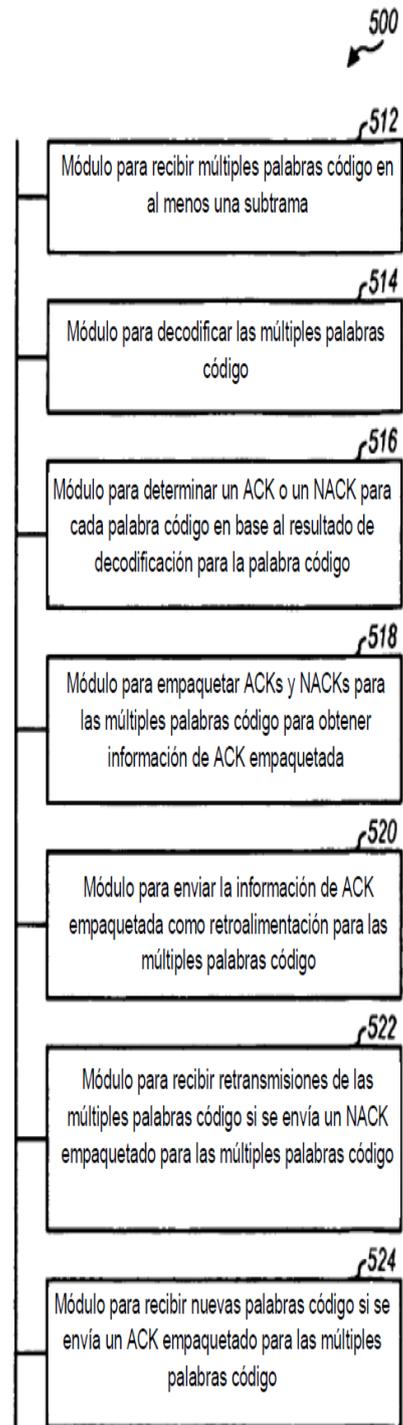


FIG. 5

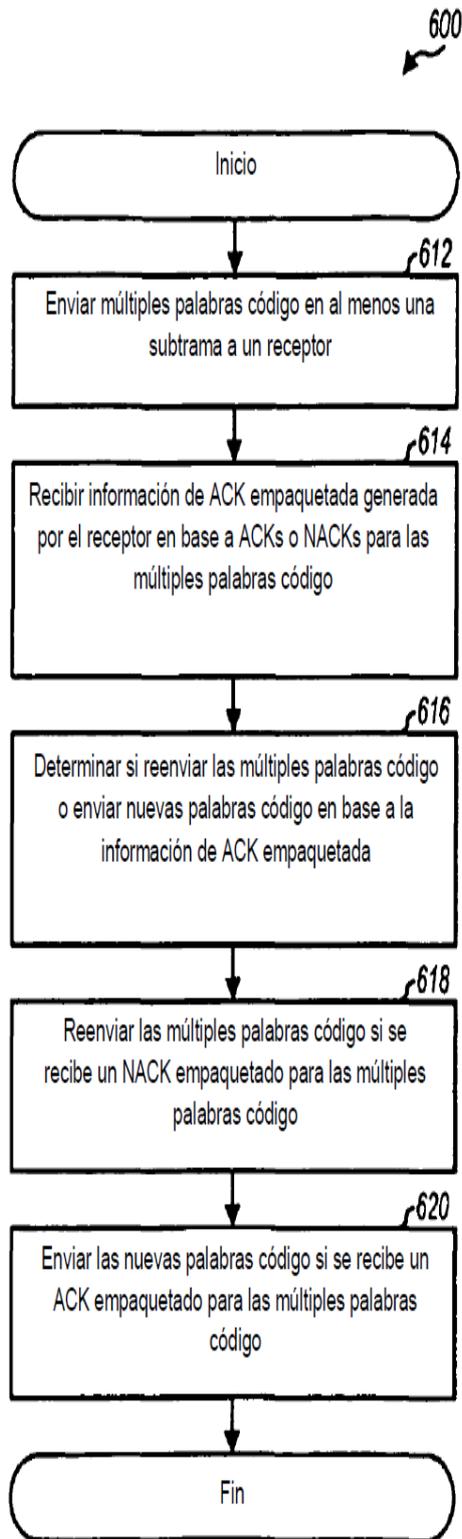


FIG. 6

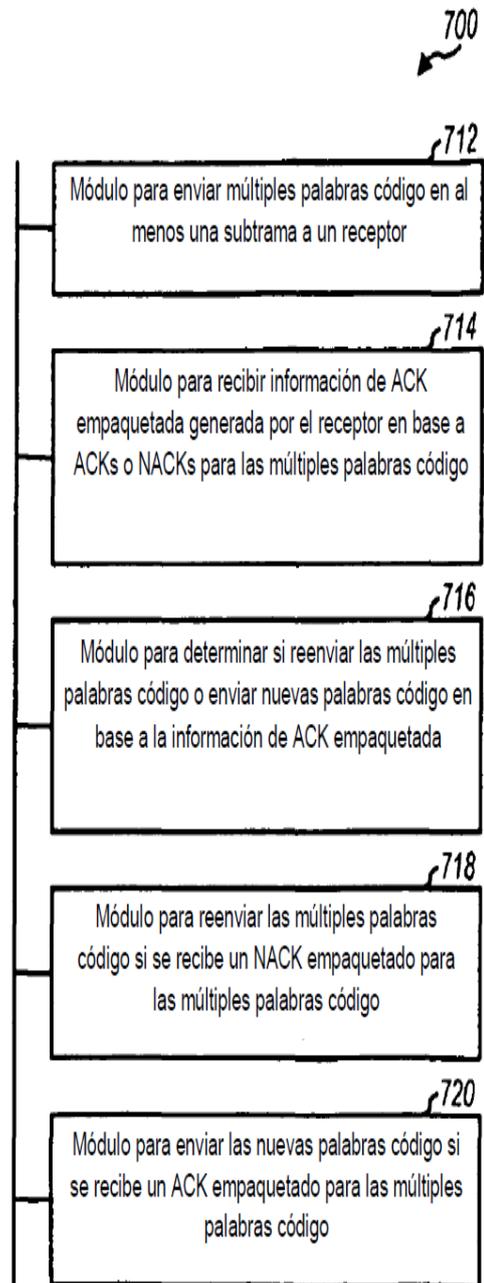


FIG. 7

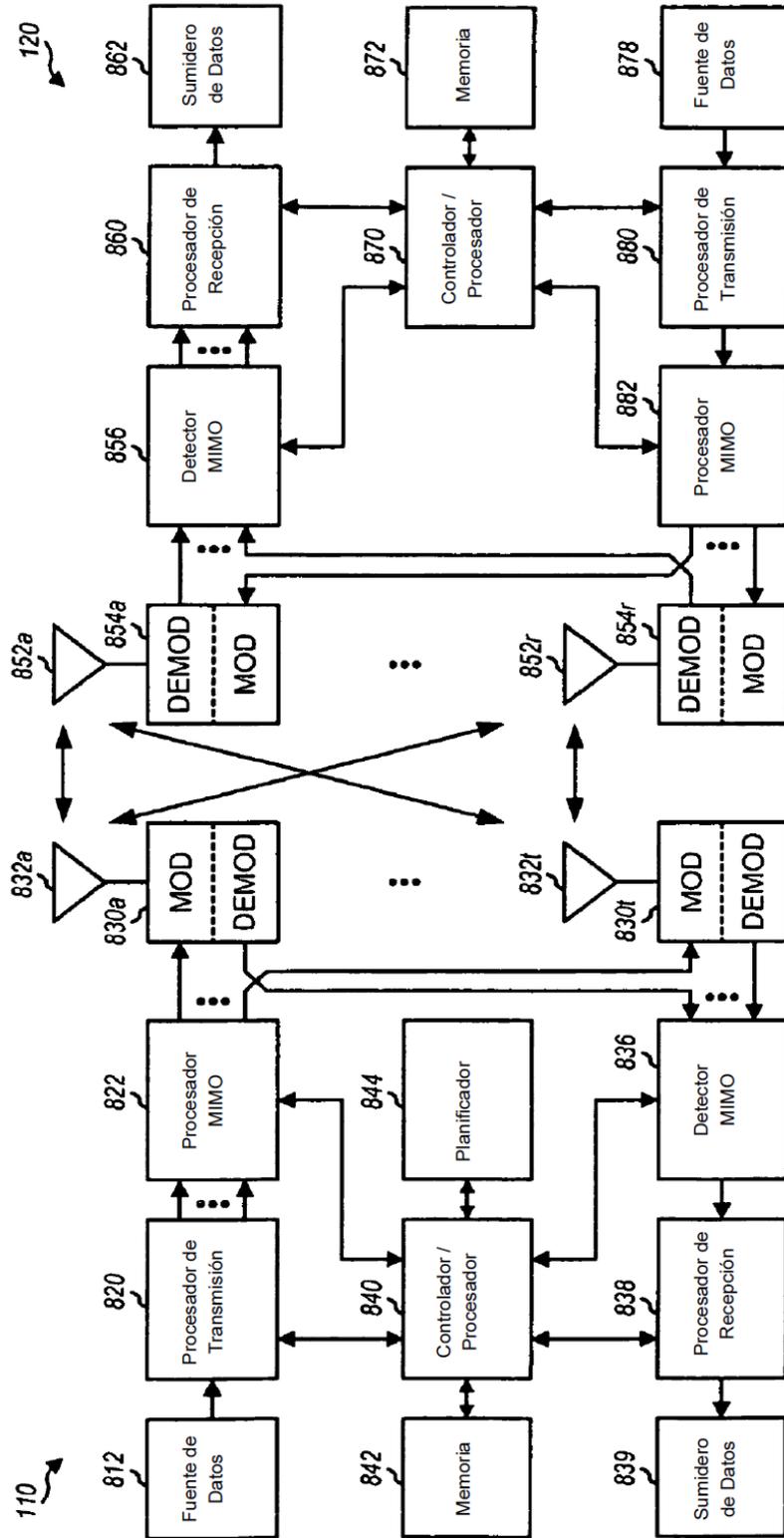


FIG. 8