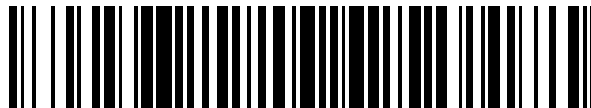


19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 718 914**

51 Int. Cl.:

**H03M 7/30** (2006.01)

**H04L 29/12** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **18.12.2014 E 14198880 (8)**

97 Fecha y número de publicación de la concesión europea: **09.01.2019 EP 2887622**

54 Título: **Procedimiento de codificación y procedimiento de decodificación de una lista de identificadores, productos de programa informático asociados, emisor y receptor que aplican dichos procedimientos**

30 Prioridad:

**18.12.2013 FR 1302989**

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

**05.07.2019**

73 Titular/es:

**THALES (100.0%)  
45, rue de Villiers  
92200 Neuilly Sur Seine, FR**

72 Inventor/es:

**SANVOISIN, PATRICK**

74 Agente/Representante:

**SALVÀ FERRER, Joan**

ES 2 718 914 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**

Procedimiento de codificación y procedimiento de decodificación de una lista de identificadores, productos de programa informático asociados, emisor y receptor que aplican dichos procedimientos

5

**[0001]** La presente invención se refiere a un procedimiento de codificación de una lista de identificadores utilizable en una red informática, constando la red informática de al menos un emisor y al menos un receptor, constando el o cada emisor y el o cada receptor de una misma lista global de identificadores diferentes, presentando cada identificador una característica técnica, siendo el o cada emisor apto para emitir un paquete de datos con destino a uno o varios receptores diferentes, constando cada paquete de datos de al menos un código correspondiente a una lista de identificadores codificada, en la que cada identificador está presente como máximo una vez.

10

**[0002]** En el campo de la telecomunicación o de la informática, es a menudo necesario transmitir una lista de identificadores, todos diferentes, que pertenecen a un conjunto mayor de identificadores conocidos.

15

**[0003]** Tal transmisión es necesaria, por ejemplo, cuando un emisor transmite un paquete de datos común para una pluralidad de receptores. En este caso, la lista de identificadores de estos receptores generalmente se incluye en el encabezamiento del paquete de datos que se va a transmitir y este encabezamiento se transmite a continuación al conjunto de receptores de la red informática.

20

**[0004]** A la recepción de este encabezamiento, cada receptor extrae la lista de identificadores y si su identificador está en la lista, este receptor continúa recibiendo el paquete de datos por completo. En el caso contrario, el receptor ignora la recepción del paquete o lo transmite a otro receptor sin haberlo leído.

25

**[0005]** En unas redes informáticas de grandes dimensiones, el identificador de un receptor se representa generalmente por un gran número de longitudes o por una combinación compleja de cifras y letras u otros símbolos. El tamaño del encabezamiento de un paquete de datos que consta de una lista de tales identificadores se vuelve, por tanto, muy importante y, a veces, supera el tamaño del resto del paquete. Esto conduce especialmente a una sobrecarga de la red informática.

30

**[0006]** En el estado de la técnica existen diferentes técnicas de transmisión que permiten disminuir el tamaño de tal lista de identificadores.

35

**[0007]** Se conoce por ejemplo el documento WO 02/46938 A2 que describe un procedimiento de codificación de una lista de identificadores que utilizan unas máscaras.

**[0008]** Se conoce igualmente el documento Donald E Knuth «Section 7.2.1.3 Generating all combinations» que describe unos procedimientos de codificación de una lista de elementos ordenada según un orden lexicográfico.

40

**[0009]** Por otro lado, es común codificar una lista de identificadores todos diferentes que pertenecen a un conjunto más amplio de identificadores conocidos.

**[0010]** La teoría de codificación muestra además que el número mínimo de bits necesario para transmitir una lista de identificadores que pertenecen a un conjunto mayor de identificadores es igual a la siguiente expresión:

45

$$N_b = \lceil \log_2 C_N^P \rceil$$

con

50  $\lceil \cdot \rceil$  que designa la parte entera por exceso,

$N_b$  que designa el número de bits,

$P$  que designa el número de destinatarios,

55

$N$  que designa el número total de receptores, y

$C_N^P$  que designa el número de subconjuntos de cardinal  $P$ , en un conjunto de cardinal  $N$ .

60

**[0011]** Una codificación óptima es una codificación que permite obtener una longitud de código de este tamaño. Una codificación óptima da por tanto un código de tamaño teórico mínimo para una lista de identificadores que pertenecen a un conjunto mayor de identificadores. Este tamaño teórico mínimo permite, en particular, transmitir de manera eficaz una lista de identificadores incluso en una red informática de gran dimensión.

**[0012]** Sin embargo, esta solución no es completamente satisfactoria.

**[0013]** Especialmente, la utilización de una codificación óptima generalmente necesita mantener en la memoria del emisor y de cada receptor una tabla de codificación que conste de todas las correspondencias entre los códigos y los identificadores. El tamaño de esta tabla es muy importante incluso para una red informática de dimensión razonable.

**[0014]** La presente invención tiene como objeto proporcionar un procedimiento de codificación óptimo sin la necesidad de mantener una tabla de codificación en la memoria.

**[0015]** A tal efecto, la invención tiene como objeto un procedimiento de codificación conforme a la reivindicación 1.

**[0016]** Según otros aspectos ventajosos de la invención, el procedimiento de codificación de una lista de identificadores comprende una característica de la reivindicación 2 o 3.

**[0017]** La invención tiene igualmente como objeto un producto de programa informático que consta de unas instrucciones informáticas que, cuando son aplicadas por un calculador, aplican el procedimiento de codificación tal como se ha definido anteriormente.

**[0018]** La invención tiene igualmente como objeto un emisor para una red informática conforme a la reivindicación 5.

**[0019]** La invención tiene igualmente como objeto un procedimiento de decodificación conforme a la reivindicación 6.

**[0020]** La invención tiene igualmente como objeto un producto de programa informático que consta de unas instrucciones informáticas que, cuando son aplicadas por un calculador, aplican el procedimiento de decodificación tal como se ha definido anteriormente.

**[0021]** La invención tiene igualmente como objeto un receptor para una red informática conforme a la reivindicación 8.

**[0022]** Estas características y ventajas de la invención se mostrarán con la lectura de la descripción que aparece a continuación, dada únicamente a título de ejemplo no limitativo y realizada en referencia a los dibujos anexos, en los que:

- la figura 1 es una vista esquemática de una red informática que consta de un emisor según la invención y una pluralidad de receptores según la invención;

- la figura 2 es una vista esquemática del emisor de la figura 1;

- la figura 3 es una vista esquemática de un receptor de la figura 1;

- la figura 4 es una vista esquemática de un paquete de datos apto para ser producido por el emisor de la figura 2 para ser transmitido al receptor de la figura 3;

- la figura 5 es una vista esquemática de una tabla de codificación según la invención que consta de 3 columnas y 35 líneas;

- la figura 6 es un organigrama de un procedimiento de codificación según la invención, aplicado por un software de codificación que forma parte del emisor de la figura 2; y

- la figura 7 es un organigrama de un procedimiento de decodificación según la invención aplicado por un software de decodificación que forma parte del receptor de la figura 3.

**[0023]** Una red 10 informática se ilustra en la figura 1. La red informática 10 comprende un emisor E según la invención y una pluralidad de receptores  $R_1$  a  $R_N$  según la invención, siendo el número total de receptores  $R_1$  a  $R_N$  igual a  $N$ .

**[0024]** Como variante, la red informática 10 consta de varios emisores E.

**[0025]** El o cada emisor E es, por ejemplo, por un equipo electrónico tal como un ordenador, un teléfono móvil o cualquier otro medio apto para enviar un paquete 12 de datos numéricos con destino a uno o varios receptores  $R_1$  a  $R_N$ . En el resto de la descripción, se designará por  $P$  el número de destinatarios correspondiente al número de los receptores  $R_1$  a  $R_N$  a los que está destinado el paquete 12.

**[0026]** La arquitectura del emisor E se ilustra en la figura 2. Así, como se ilustra en esta figura 2, el receptor E consta de una unidad 20 de tratamiento de datos.

- [0027]** La unidad de tratamiento de datos 20 está formada, por ejemplo, por un procesador 22 y una memoria 24 asociada con este procesador 22.
- [0028]** El procesador 22 es apto para ejecutar una pluralidad de software que la memoria 24 es apta para almacenar.
- [0029]** Así, por ejemplo, la memoria 24 es apta para almacenar un software 26 de generación de datos. Tal software 26 es apto, por ejemplo, para generar unos datos numéricos destinados a  $P$  receptores  $R_1$  a  $R_N$ .
- 10 **[0030]** El emisor E consta además de una unidad 28 de emisión conectada a la unidad de tratamiento de datos 20. La unidad de emisión 28 es apta para formar un paquete de datos 12 correspondiente a los datos numéricos generados por el software 26 y para emitir este paquete 12 en la red informática 10.
- [0031]** Cada receptor  $R_1$  a  $R_N$  es, por ejemplo, un equipo electrónico tal como un ordenador, un teléfono móvil  
15 o cualquier otro medio apto para recibir un paquete de datos 12 procedente del o de cada emisor E.
- [0032]** La arquitectura de un receptor  $R_j$  se ilustra en la figura 3. Así, como se ilustra en esta figura 3, el receptor  $R_j$  consta de una unidad 30 de tratamiento de datos.
- 20 **[0033]** La unidad 30 está formada, por ejemplo, por un procesador 32 y una memoria 34 asociada a este procesador 32.
- [0034]** El procesador 32 es apto para ejecutar una pluralidad de software que la memoria 34 es apta para almacenar.
- 25 **[0035]** Así, por ejemplo, la memoria 34 es apta para almacenar un software 36 de consumo de datos.
- [0036]** El receptor  $R_j$  consta además de una unidad 38 de recepción conectada a la unidad de tratamiento de datos 30. La unidad de recepción 38 es apta para recibir cada paquete de datos 12 emitido en la red informática 10  
30 por el o cada emisor E y transmitir los datos numéricos correspondientes al software 36 que es apto entonces para consumir estos datos 12 recibidos.
- [0037]** La arquitectura de todos los demás receptores  $R_1$  a  $R_N$  es similar a la del receptor  $R_j$ .
- 35 **[0038]** Cada receptor  $R_1$  a  $R_N$  consta de un identificador  $Id_k$  único en la red informática 10, todos los identificadores  $d_k$  son diferentes. Este identificador  $Id_k$  permite distinguir de manera única y segura cada receptor  $R_1$  a  $R_N$ .
- [0039]** Cada identificador  $Id_k$  es, por ejemplo, un número entero positivo.
- 40 **[0040]** Como variante, cada identificador  $Id_k$  es una sucesión de cifras y/o de letras o de otros símbolos aceptables por la red informática 10.
- [0041]** En los dos casos, cada identificador  $Id_k$  es apto para ser asociado a un número entero  $k$  comprendido  
45 entre 1 y  $N$ , siendo el número  $k$  diferente para unos identificadores  $Id_k$  diferentes.
- [0042]** Así, por ejemplo, el número  $k$  es el número del identificador  $Id_k$  en una lista ordenada que consta del conjunto de los identificadores  $Id_k$  de la red informática 10.
- 50 **[0043]** Tal asociación es apta para ser conocida por el o cada emisor E y el conjunto de receptores  $R_1$  a  $R_N$  antes de cualquier transmisión.
- [0044]** Como variante, esta asociación es apta para ser definida en una fase de construcción de la red 10.
- 55 **[0045]** La estructura del paquete de datos 12 se ilustra en detalle en la figura 4.
- [0046]** Así, como se ilustra en esta figura 4, el paquete de datos 12 está formado por un mensaje aplicativo 42 y un encabezamiento 44. Tal paquete de datos 12 es apto para ser generado por la unidad de emisión 28 del o de cada emisor E y ser recibido por la unidad de recepción 38 de cada receptor  $R_1$  a  $R_N$ .
- 60 **[0047]** El mensaje aplicativo 42 comprende los datos numéricos generados por el software de generación de datos 26 y destinados a uno o varios receptores  $R_1$  a  $R_N$ .
- [0048]** El encabezamiento 44 comprende cualquier información auxiliar que permite transmitir el paquete 12 al  
65 destinatario correspondiente a través de la red informática 10.

**[0049]** Así, en el ejemplo de realización ilustrado en la figura 4, el encabezamiento 44 incluye el número  $P$  de destinatarios y una lista codificada  $L$  de identificadores de receptores  $R_1$  a  $R_N$  a los que está destinado este paquete 12.

5

**[0050]** Tal lista de identificadores  $L$  permite, por ejemplo, al emisor  $E$  emitir el paquete 12 en la red informática 10 sin precisión explícita de desplazamiento.

**[0051]** Cada receptor  $R_1$  a  $R_N$  es apto para recibir al menos el encabezamiento 44 de cada paquete 12 emitido.

10 Así, la unidad de recepción 38 del receptor correspondiente es apta para leer la lista de identificadores  $L$  y, si su identificador  $Id_k$  está en la lista, continuar la recepción del paquete 12. En el caso contrario, el receptor es apto para ignorar la recepción del paquete 12 o transmitirlo hacia otro receptor  $R_1$  a  $R_N$ .

**[0052]** La lista de identificadores  $L$  está codificada por un código  $C$  a fin de minimizar su tamaño. El tamaño de

15 la lista de identificadores  $L$  se mide en el número  $N_b$  de bits que esta lista ocupa en el encabezamiento 44.

**[0053]** A tal efecto, la memoria 24 del emisor  $E$  es apto para almacenar además un software 46 de codificación que permite codificar la lista de identificadores  $L$  para cada paquete 12 emitido por este emisor  $E$ .

20 **[0054]** Como variante, tal codificación es asegurada por un módulo de codificación independiente que forma parte del o de cada emisor  $E$ .

**[0055]** De manera análoga, la memoria 34 de cada receptor  $R_1$  a  $R_N$  es apta para almacenar además un software 56 de decodificación que permite decodificar el código  $C$  de cada paquete 12 recibido por la unidad de

25 recepción 38 de este receptor  $R_1$  a  $R_N$  para obtener la lista de identificadores  $L$  correspondiente.

**[0056]** Como variante, tal decodificación es asegurada por un módulo de decodificación independiente que forma parte de cada receptor  $R_1$  a  $R_N$ .

30 **[0057]** El funcionamiento del software de codificación 46 del emisor  $E$  se va a explicar de aquí en adelante.

**[0058]** El software de codificación 46 es aplicado por un procedimiento 100 de codificación de una lista de identificadores según la invención.

35 **[0059]** Este procedimiento de codificación se basa en una tabla  $T_p$  de codificación para cada número  $P$  de identificadores de la lista de identificadores  $L$ .

**[0060]** Así, para un paquete de datos 12 considerado, tal tabla de codificación  $T_p$  está formada por una pluralidad de columnas con un número de  $P$  y una pluralidad de líneas con número igual al número de subconjuntos

40 de cardinal  $P$  en un conjunto de cardinal  $N$ , es decir:

$$[T_p] = C_N^P \times P$$

en el que  $[T_p]$  designa las dimensiones de la tabla  $T_p$ .

45

**[0061]** Cada línea de la tabla  $T_p$  es numerada de 0 a  $C_N^P - 1$  y está formada por una lista de identificadores  $L$  que consta de  $P$  identificadores  $Id_k$  diferentes. Cada identificador  $Id_k$  corresponde así a una columna.

**[0062]** En el resto de la descripción, se designará por  $LG$  una lista global ordenada de identificadores que

50 consta del conjunto de identificadores  $Id_k$  de la red informática 10 ordenados por el orden descendente del número  $k$  asociado a cada identificador  $Id_k$ .

**[0063]** La tabla de codificación  $T_p$  está construida de forma única para  $N$  receptores y  $P$  destinatarios de acuerdo con una ley que se describe a continuación.

55

**[0064]** Esta ley utiliza ciertas propiedades del triángulo de Pascal. En particular, la relación siguiente:

$$C_N^P = \sum_{q=P-1}^{N-1} C_q^{P-1}.$$

60 permite construir la primera columna de la tabla  $T_p$  con  $N - P + 1$  de identificadores diferentes.

**[0065]** Más particularmente, las  $C_{N-1}^{P-1}$  primeras líneas de la primera columna de la tabla  $T_p$  son completadas por el primer identificador  $Id_k$  de la lista global de identificadores  $LG$ .

5 **[0066]** Las  $C_{N-2}^{P-1}$  líneas siguientes de la primera columna son completadas por el segundo identificador  $Id_k$  de la lista global de identificadores  $LG$ .

**[0067]** Finalmente, la última línea de la primera columna correspondiente al número  $C_{N-1}^{P-1}$  es completada por el identificador  $Id_k$  que tiene la posición  $N - P + 1$  en la lista global de identificadores  $LG$ .

10

**[0068]** Así, todas las  $C_N^P$  líneas de la tabla son completadas con  $N - P + 1$  identificadores diferentes.

**[0069]** La segunda columna de la tabla  $T_p$  se construye de forma analógica al retirar el primer identificador  $Id_k$  de la lista global de identificadores  $LG$  y al disminuir los números  $N$  y  $P$  en uno.

15

**[0070]** Las otras columnas se obtienen aplicando la misma regla recursiva.

**[0071]** Al utilizar el análisis combinatorio, es posible demostrar que la tabla de codificación  $T_p$  así construida consta de *todas* las listas  $L$  diferentes posibles que constan de  $P$  identificadores diferentes entre los  $N$  identificadores de la red 10.

20

**[0072]** Así, el número de cada línea determina de forma única la lista de identificadores  $L$  correspondiente. El software de codificación 46 asocia entonces un código  $C$  igual al número de la línea a cada lista de identificadores  $L$  que consta de  $P$  identificadores.

25

**[0073]** En consecuencia, el número de bits  $N_b$  necesarios para transmitir este código  $C$  en el encabezamiento 44 es igual a la parte entera por exceso del logaritmo de base dos del número de subconjuntos de cardinal igual al número de destinatarios  $P$ , en un conjunto de cardinal igual al número total de receptores  $N$ , es decir,

$$N_B = \lceil \log_2 C_N^P \rceil.$$

30

**[0074]** Esto significa, especialmente, que la codificación por la tabla de codificación  $T_p$  es una codificación óptima y que no existen otras codificaciones que permitan obtener un código  $C$  de tamaño inferior.

35 **[0075]** Un ejemplo de una tabla de codificación  $T_p$  para  $N = 7$  y  $P = 3$  se representa en la figura 5. En este ejemplo, cada identificador  $Id_k$  corresponde a un número de entrada que va de 1 a 7. Cada número  $k$  asociado es simplemente igual al identificador  $Id_k$  correspondiente.

**[0076]** Así, en el ejemplo de la figura 5, el código  $C$  para una lista de identificadores que consta de los identificadores «5», «3» y «1» es igual a «29». El número de bits  $N_b$  necesarios para transmitir este código  $C$  es igual a 6 bits.

40

**[0077]** El procedimiento de codificación 100 permite obtener un código  $C$  de una lista de identificadores  $L$  que consta de  $P$  identificadores correspondientes al número de la línea que consta de esta lista  $L$  en la tabla de codificación  $T_p$  correspondiente, siendo realizado esto sin construcción explícita de la tabla  $T_p$  y sin necesidad de mantenerla en la memoria.

45

**[0078]** El organigrama del procedimiento de codificación 100 aplicado por la unidad de tratamiento 30 se representa en la figura 6. Se aplica, sobre la marcha, durante cada transmisión de un paquete 12.

50

**[0079]** Durante una etapa preliminar 105, el software de codificación 46 asocia a cada identificador  $Id_k$  un número  $k$  entero comprendido entre uno y el número total de receptores  $N$ , siendo el número  $k$  diferente para unos identificadores  $Id_k$  diferentes.

55 **[0080]** Esta etapa 105 se efectúa, por ejemplo, en la fase de construcción de la red informática 10. Los resultados de esta asociación se transmiten a continuación al conjunto de los receptores  $R_1$  a  $R_N$  de la red informática 10.

60

**[0081]** Durante la etapa 110, el software de codificación 46 recibe del software de generación de datos 26, una lista de identificadores  $L$  que consta de  $P$  identificadores  $Id_k$ .

**[0082]** Durante la etapa 115, el software de codificación 46 ordena la lista de identificadores  $L$  por orden

descendente del número  $k$  asociado a cada identificador  $ld_k$  para obtener una lista  $LT$  ordenada de identificadores.

**[0083]** Esta lista ordenada de identificadores  $LT$  está ordenada por tanto y tiene la siguiente forma:

$$LT = \{l_1, l_2, \dots, l_p\}$$

5

en la que cada  $l_j$  corresponde a un identificador  $ld_k$  de la lista de identificadores  $L$  ordenado según el orden descendente del número  $k$ .

10 **[0084]** Durante la etapa 120, el software de codificación 46 define una primera variable  $j$  entera igual al número de destinatarios  $P$ , un valor máximo intermedio  $max$  igual al número total de receptores  $N$ , es decir:

$$j = P, max = N.$$

15 **[0085]** En la misma etapa, el software de codificación 46 inicializa el código  $C$  por cero.

**[0086]** Siempre que la primera variable  $j$  sea superior a cero durante la etapa 125, el software de codificación 46 pasa a la etapa 130.

20 **[0087]** Durante la etapa 130, el software de codificación 46 varía una segunda variable  $i$  entera entre el número  $k_1$  asociado al primer identificador  $l_1$  de la lista ordenada de identificadores  $LT$  y el valor máximo intermedio  $max$  disminuido en uno.

**[0088]** En esta etapa 130, el software de codificación 46 para cada valor de la segunda variable  $i$ , calcula el número de subconjuntos de cardinal igual a la primera variable  $j$  disminuida en uno, en un conjunto de cardinal igual a la segunda variable  $i$ .

25 **[0089]** El software de codificación 46 calcula por otro lado un código elemental  $C_j$  correspondiente al primer identificador  $l_1$  de la lista ordenada de identificadores  $LT$  como la suma de los números de subconjunto, es decir:

30

$$C_j = \sum_{i=k_1}^{max-1} C_i^{j-1}.$$

**[0090]** Durante la etapa 135, el software de codificación 46 establece el valor máximo intermedio  $max$  igual al número  $k_1$  asociado al primer identificador  $l_1$  disminuido en uno, es decir:

35

$$max = k_1 - 1.$$

**[0091]** Durante la etapa 140, el software de codificación 46 elimina el identificador  $l_1$  de la lista ordenada de identificadores  $LT$  y disminuye en uno la primera variable  $j$ , es decir:

40

$$LT' = LT \setminus l_1 = \{l_2, \dots, l_j\}, \quad j' = j - 1.$$

**[0092]** El software de codificación 46 cambia los nombres de variables en la lista  $LT'$  sin permutación de estas, de forma que se obtenga una lista siguiente:

45

$$LT' = \{l_1, \dots, l_{j-1}\}.$$

**[0093]** El software de codificación 46 actualiza la lista  $LT$  y la variable  $j'$  del siguiente modo:

50

$$LT = LT', \quad j = j'.$$

**[0094]** Después, el software de codificación 46 vuelve a la etapa 125.

**[0095]** Si durante la etapa 125, la primera variable  $j$  es igual a cero, el software de codificación 46 pasa a la etapa 145.

**[0096]** Dicho de otro modo, durante unas etapas anteriores, siempre que la primera variable  $j$  sea mayor que cero, el software de codificación codifica el primer identificador  $l_1$  de la lista ordenada de identificadores  $LT$  por un código elemental  $C_j$  correspondiente al número de subconjuntos de la lista global  $LG$  de identificadores de cardinal igual al número de identificadores  $l_k$  de la lista ordenada de identificadores  $LT$ , siendo cada subconjunto ordenado por orden descendente del número  $k$  asociado a cada identificador  $l_k$  y que consta en la primera posición de un identificador  $l_k$  con un número  $k$  asociado superior al número  $k$  asociado al primer identificador  $l_1$  de la lista ordenada de identificadores  $LT$ , elimina este identificador  $l_1$  de la lista ordenada de identificadores  $LT$  y disminuye en uno la primera variable  $j$ .

**[0097]** Durante la etapa 145, el software de codificación 46 calcula el código  $C$  sumando todos los códigos elementales  $C_j$  obtenidos, es decir:

$$C = \sum_{j=1}^P C_j.$$

**[0098]** Durante la etapa 150, el software de codificación 46 transmite el código  $C$  obtenido a la unidad de emisión 28.

**[0099]** El funcionamiento del software de decodificación 56 aplicado por la unidad de tratamiento 30 de cada receptor  $R_1$  a  $R_N$ , aplicado por un procedimiento 200 de decodificación de una lista de identificadores según la invención se va a explicar de aquí en adelante.

**[0100]** El organigrama del procedimiento de decodificación 200 se representa en la figura 7. Este procedimiento 200 se aplica a cada recepción.

**[0101]** Durante una etapa 205 preliminar, el software de decodificación 56 recibe una lista que consta para cada identificador  $l_k$  de un número  $k$  asociado a este identificador. Esta etapa 205 se efectúa, por ejemplo, en la fase de construcción de la red informática 10.

**[0102]** Durante la etapa 210, el software de decodificación 56 recibe el número de identificadores  $P$  y el código  $C$  de la parte de la unidad de recepción 38.

**[0103]** Durante la etapa 215, el software de decodificación 56 define un valor máximo intermedio  $max$  igual al número total de receptores  $N$  y una primera variable  $j$  entera igual al número de destinatarios  $P$ , es decir:

$$max = N, \quad j = P.$$

**[0104]** El software de decodificación 56 inicializa además una lista de identificadores  $L$  por una lista vacía, es decir:

$$L = \{\emptyset\}.$$

**[0105]** Durante la etapa 220, el software de decodificación 56 calcula el número  $F$  de subconjuntos de cardinal igual a la primera variable  $j$  disminuida en uno, en un conjunto de cardinal igual al valor máximo intermedio  $max$  disminuido en uno, es decir:

$$F = C_{max-1}^{j-1}.$$

**[0106]** Si durante la etapa 225 el valor del código  $C$  es inferior a dicho número de subconjuntos, durante la etapa 230 el software de decodificación 56 añade en la lista de identificadores  $L$  el identificador  $l_k$  con un número  $k$  asociado igual al valor máximo intermedio  $max$ , disminuye el valor máximo intermedio  $max$  y la primera variable  $j$  en uno, es decir:

55



$$L = L \cup \{Id_{max}\}, \quad max' = max - 1, \quad j' = j - 1$$

$$max = max', \quad j = j'.$$

**[0107]** Si durante la etapa 235, la primera variable  $j$  es mayor que cero, el software de decodificación 56 vuelve a la etapa 225.

5

**[0108]** Si durante la etapa 225 el valor del código  $C$  es mayor o igual a dicho número de subconjuntos, durante la etapa 240 el software de decodificación 56 disminuye el valor máximo intermedio  $max$  en uno y calcula un nuevo valor del código igual a la diferencia entre el valor actual del código  $C$  y dicho número de subconjuntos  $F$ , es decir:

$$max' = max - 1, \quad C' = C - F$$

10

$$max = max', \quad C = C'.$$

**[0109]** El software de decodificación 56 vuelve entonces a la etapa 220.

15 **[0110]** Si durante la etapa 235, la primera variable  $j$  es igual a cero, durante la etapa 245 el software de decodificación 56 transmite la lista de identificadores  $L$  obtenida a la unidad de recepción 38.

**[0111]** El funcionamiento de la red informática 10 que consta de un emisor  $E$  según la invención y  $N$  receptores  $R_1$  a  $R_N$  según la invención, se va a explicar de aquí en adelante.

20

**[0112]** El software de generación de datos 26 del emisor  $E$  genera unos datos numéricos destinados a  $P$  receptores  $R_1$  a  $R_N$ .

25 **[0113]** El software 26 genera además una lista de identificadores  $L$  que constan de los  $P$  identificadores  $Id_k$  de los receptores  $R_1$  a  $R_N$  a los que están destinados estos datos.

**[0114]** Después, el software de codificación 46 codifica esta lista de identificadores  $L$  por un código  $C$  según el procedimiento de codificación de una lista de identificadores 100.

30 **[0115]** La unidad de emisión 28 forma entonces un paquete de datos 12 incluyendo un mensaje aplicativo 42 correspondiente a los datos numéricos generados por el software 26 y un encabezamiento 44. El encabezamiento 44 consta del número  $P$  y el código  $C$ .

**[0116]** El paquete de datos 12 se emite entonces en la red informática 10 por la unidad de emisión.

35

**[0117]** Después, este paquete de datos 12 o al menos su encabezamiento 44 es recibido por la unidad de recepción 38 de cada receptor  $R_1$  a  $R_N$ . La unidad 38 extrae el número  $P$  y el código  $C$ .

40 **[0118]** El software de decodificación 56 decodifica entonces el código  $C$  según el procedimiento de decodificación de una lista de identificadores 200, para obtener la lista de identificadores  $L$  correspondiente.

**[0119]** Si el identificador  $Id_k$  de un receptor  $R_1$  a  $R_N$  está en la lista  $L$ , la unidad de recepción 38 de este receptor  $R_1$  a  $R_N$  continúa recibiendo el paquete de datos 12 y transmite el mensaje aplicativo 42 correspondiente al software de consumo de datos 36.

45

**[0120]** En el caso contrario, la unidad de recepción 38 ignora la recepción del paquete 12 y/o lo transmite hacia otro receptor  $R_1$  a  $R_N$ .

50 **[0121]** El procedimiento de codificación 100 y el procedimiento de decodificación 200 permiten entonces codificar y decodificar una lista de identificadores  $L$  utilizando un código  $C$  óptimo sin necesidad de mantener una tabla de codificación  $T_p$  en la memoria del o cada emisor  $E$  y del conjunto de los receptores  $R_1$  a  $R_N$ . Esto permite, en particular, reducir considerablemente el tamaño del encabezamiento 44 para cada paquete de datos destinado a  $P$  receptores  $R_1$  a  $R_N$ .

55 **[0122]** Por supuesto, son igualmente posibles muchos otros modos de realización de los procedimientos de codificación y decodificación. Así, por ejemplo, en la etapa 115 del procedimiento de codificación 100, es posible ordenar cada lista de identificador  $L$  que se va a transmitir por orden ascendente del número  $k$  asociado. Las otras etapas del procedimiento de codificación 100 y del procedimiento de decodificación 200 se modifican en recurrencia.

60 **[0123]** La utilización de los procedimientos de codificación 100 y de decodificación 200 según la invención no

está restringida únicamente por la codificación y decodificación de una lista de identificadores  $L$  en el encabezamiento de un paquete de datos. De manera general, estos procedimientos son aplicables para codificar y decodificar una lista cualquiera de elementos que constan de elementos de un conjunto conocido.

REIVINDICACIONES

1. Procedimiento (100) de codificación de una lista de identificadores ( $L$ ) que se pueden utilizar en una red informática (10), constando la red informática (10) de al menos un emisor (E) y al menos un receptor ( $R_1, \dots, R_N$ ),  
 5 constando el o cada emisor y el o cada receptor de una misma lista global ( $LG$ ) de identificadores ( $Id_k$ ) diferentes, presentando cada identificador ( $Id_k$ ) una característica técnica, siendo el o cada emisor (E) apto para emitir un paquete (12) de datos con destino a uno o varios receptores ( $R_1, \dots, R_N$ ) diferentes,  
 constando cada paquete de datos (12) de al menos un código ( $C$ ) correspondiente a una lista de identificadores ( $L$ )  
 10 codificada, en la que cada identificador ( $Id_k$ ) solo está presente como máximo una vez,  
**caracterizado porque** el procedimiento de codificación (100) comprende las etapas siguientes:

- asociar (105) a cada identificador ( $Id_k$ ) un número ordinal ( $k$ ) comprendido entre uno y el número total ( $N$ ) de identificadores ( $Id_k$ ) en la lista global de identificadores ( $LG$ ), siendo diferente el número ( $k$ ) para unos identificadores  
 15 ( $Id_k$ ) diferentes;
- ordenar según una relación de orden estricta predeterminada, la lista de identificadores ( $L$ ) según el número ordinal ( $k$ ) asociado a cada identificador ( $Id_k$ ) para obtener una lista ordenada ( $LT$ ) de identificadores;
- definir (120) una primera variable ( $j$ ) entera igual al número ( $P$ ) de identificadores en la lista de identificadores ( $L$ ) que se va a transmitir, inicializar una lista ordenada de identificadores anterior ( $LT'$ ) por la lista ordenada de identificadores  
 20 ( $LT$ );
- codificar (130, 135) el primer identificador ( $I_1$ ) de la lista ordenada de identificadores anterior ( $LT'$ ) por un código elemental ( $C_j$ ) correspondiente al número de subconjuntos de la lista global de identificadores ( $LG$ ) de cardinal igual al número de identificadores ( $Id_k$ ) de la lista ordenada de identificadores anterior ( $LT$ ) y que consta al menos de un  
 25 identificador ( $Id_k$ ) cuyo número ordinal ( $k$ ) asociado está en una relación de orden estricto predeterminada con el número ordinal ( $k$ ) asociado al primer identificador ( $I_1$ ) de la lista ordenada de identificadores anterior ( $LT'$ ), eliminar (140) este identificador ( $I_1$ ) de la lista ordenada de identificadores anterior ( $LT'$ ) para obtener una lista ordenada de identificadores siguiente, y disminuir en uno la primera variable ( $j$ );
- codificar (145) la lista de identificadores por un código ( $C$ ) correspondiente a la suma de todos los códigos elementales ( $C_j$ ) obtenidos.

30 2. Procedimiento (100) según la reivindicación 1, **caracterizado porque** un primer número ordinal está en una relación de orden estricta predeterminada con un segundo número ordinal cuando el primer número ordinal es estrictamente superior al segundo número ordinal.

35 3. Procedimiento (100) según la reivindicación 2, **caracterizado porque** consta además de las etapas siguientes:

- definir (120) un valor máximo intermedio ( $max$ ) igual al número total ( $N$ ) de identificadores ( $Id_k$ ) en la lista global ( $LG$ ) de identificadores;
- 40 - variar (130) una segunda variable ( $i$ ) entera entre el número ordinal ( $k_1$ ) del primer identificador ( $I_1$ ) de la lista ordenada de identificadores anterior ( $LT'$ ) y el valor máximo intermedio ( $max$ ) disminuido en uno;
- para cada valor de la segunda variable ( $i$ ), calcular el número de subconjuntos de cardinal igual a la primera variable ( $j$ ) disminuido en uno, en un conjunto de cardinal igual a la segunda variable ( $i$ );
- calcular (130) un código elemental ( $C_j$ ) correspondiente al primer identificador ( $I_1$ ) de la lista ordenada de  
 45 identificadores anterior ( $LT'$ ) como la suma de dichos números de subconjunto, es decir:

$$C_j = \sum_{i=k_1}^{max-1} C_i^{j-1}$$

con

- 50  $k_1$  que designa el número asociado a dicho primer identificador,
- $i$  que designa dicha segunda variable,
- $max$  que designa dicho valor máximo,
- $C_i^{j-1}$  que designa dicho número de subconjuntos, y
- 55  $C_j$  que designa dicho código elemental.

- establecer (140) el valor máximo intermedio ( $max$ ) igual al número ( $k_1$ ) asociado al primer identificador ( $I_1$ ) disminuido en uno, y
- 60 - ejecutar las etapas de codificación del primer identificador de la lista ordenada del identificador anterior por un código elemental, eliminación y disminución siempre que la primera variable ( $j$ ) sea mayor que cero (125).

4. Producto de programa informático que consta de instrucciones informáticas que, cuando son aplicadas por un calculador, aplican el procedimiento de codificación (100) según cualquiera de las reivindicaciones anteriores.
- 5 5. Emisor (E) para una red informática (10), constando la red informática (10) de al menos un receptor ( $R_1, \dots, R_N$ ),  
constando el emisor y el o cada receptor de una misma lista global (LG) de identificadores ( $Id_k$ ) diferentes, presentando cada identificador ( $Id_k$ ) una característica técnica,  
10 siendo el emisor (E) apto para emitir un paquete (12) de datos con destino a uno o varios receptores ( $R_1, \dots, R_N$ ) diferentes,  
constando cada paquete de datos (12) de al menos un código (C) correspondiente a una lista de identificadores (L) codificada, en la que cada identificador ( $Id_k$ ) solo está presente como máximo una vez,  
**caracterizado porque** consta de un módulo (46) de codificación apto para:
- 15 - asociar (105) a cada identificador ( $Id_k$ ) un número ordinal ( $k$ ) comprendido entre uno y el número total ( $N$ ) de identificadores ( $Id_k$ ) en la lista global de identificadores (LG), siendo diferente el número ( $k$ ) para unos identificadores ( $Id_k$ ) diferentes;  
- ordenar según una relación de orden estricta predeterminada, la lista de identificadores (L) según el número ordinal  
20 ( $k$ ) asociado a cada identificador ( $Id_k$ ) para obtener una lista ordenada (LT) de identificadores;  
- definir (120) una primera variable ( $j$ ) entera igual al número ( $P$ ) de identificadores en la lista de identificadores (L) que se va a transmitir, inicializar una lista ordenada de identificadores anterior (LT') por la lista ordenada de identificadores (LT);  
- codificar (130, 135) el primer identificador ( $I_1$ ) de la lista ordenada de identificadores anterior (LT') por un código  
25 elemental ( $C_j$ ) correspondiente al número de subconjuntos de la lista global de identificadores (LG) de cardinal igual al número de identificadores ( $Id_k$ ) de la lista ordenada de identificadores anterior (LT') y que consta al menos de un identificador ( $Id_k$ ) cuyo número ordinal ( $k$ ) asociado está en una relación de orden estricto predeterminada con el número ordinal ( $k$ ) asociado al primer identificador ( $I_1$ ) de la lista ordenada de identificadores anterior (LT'), eliminar  
30 (140) este identificador ( $I_1$ ) de la lista ordenada de identificadores anterior (LT') para obtener una lista ordenada de identificadores siguiente, y disminuir en uno la primera variable ( $j$ );  
- codificar (145) la lista de identificadores por un código (C) correspondiente a la suma de todos los códigos elementales ( $C_j$ ) obtenidos.
6. Procedimiento (200) de decodificación de una lista de identificadores (iL) que se puede utilizar en una  
35 red informática (10), constando la red informática (10) de al menos un emisor (E) y al menos un receptor ( $R_1, \dots, R_N$ ),  
constando el o cada emisor y el o cada receptor de una misma lista global (LG) de identificadores ( $Id_k$ ) diferentes, presentando cada identificador ( $Id_k$ ) una característica técnica,  
siendo el o cada emisor (E) apto para emitir un paquete (12) de datos con destino a uno o varios receptores ( $R_1, \dots, R_N$ ) diferentes,  
40 constando cada paquete de datos (12) de al menos un código (C) correspondiente a una lista de identificadores (L) codificada, en la que cada identificador ( $Id_k$ ) solo está presente como máximo una vez,  
**caracterizado porque** el procedimiento (200) comprende las etapas siguientes:
- 45 - asociar (205) a cada identificador ( $Id_k$ ) un número ordinal ( $k$ ) comprendido entre uno y el número total ( $N$ ) de identificadores en la lista global de identificadores (LG), siendo el número ( $k$ ) diferente para unos identificadores ( $Id_k$ ) diferentes;  
- recibir (210) un número ( $P$ ) correspondiente al número de identificadores ( $Id_k$ ) en la lista de identificadores (L) que se va a decodificar y un código (C) correspondiente a esta lista de identificadores;  
- definir (215) un valor máximo intermedio ( $max$ ) igual al número total de receptores ( $N$ ) y una primera variable ( $j$ )  
50 entera igual al número de identificadores ( $P$ ) en la lista de identificadores (L) que se van a decodificar;  
- calcular (220) el número ( $F$ ) de subconjuntos de cardinal igual a la primera variable ( $j$ ) disminuido en uno, en un conjunto de cardinal igual al valor máximo intermedio ( $max$ ) disminuido en uno;  
- si el valor del código es inferior a dicho número ( $F$ ) de subconjuntos, añadir (230) en la lista de identificadores (L) el  
55 identificador ( $Id_k$ ) con un número ordinal ( $k$ ) igual al valor máximo intermedio ( $max$ ), disminuir el valor máximo intermedio ( $max$ ) y la primera variable ( $j$ ) en uno;  
- si el valor del código es superior o igual a dicho número ( $F$ ) de subconjuntos, disminuir (240) el valor máximo intermedio ( $max$ ) en uno y calcular un nuevo valor del código igual a la diferencia entre el valor actual del código (C) y dicho número ( $F$ ) de subconjuntos.
- 60 7. Producto de programa informático que consta de unas instrucciones informáticas que, cuando son aplicadas por un calculador, aplican el procedimiento (200) de decodificación según la reivindicación 6.
8. Receptor ( $R_1, \dots, R_N$ ) para una red informática (10), constando la red informática (10) de al menos un emisor (E), constando el o cada emisor y el receptor de una misma lista global (LG) de identificadores ( $Id_k$ ) diferentes,  
65 presentando cada identificador ( $Id_k$ ) una característica técnica,

siendo el o cada emisor (E) apto para emitir un paquete (12) de datos con destino a uno o varios receptores ( $R_1, \dots, R_N$ ) diferentes,

constando cada paquete de datos (12) de al menos un código (C) correspondiente a una lista de identificadores (L) codificada, en la que cada identificador ( $Id_k$ ) solo está presente como máximo una vez,

5 **caracterizado porque** consta de un módulo (56) de decodificación apto para:

- asociar (205) a cada identificador ( $Id_k$ ) un número ordinal ( $k$ ) comprendido entre uno y el número (N) de identificadores en la lista global de identificadores (LG), siendo el número ( $k$ ) diferente para unos identificadores ( $Id_k$ ) diferentes;

10 - recibir (210) un número (P) correspondiente al número de identificadores ( $Id_k$ ) en la lista de identificadores (L) que se va a codificar y un código (C) correspondiente a esta lista de identificadores;

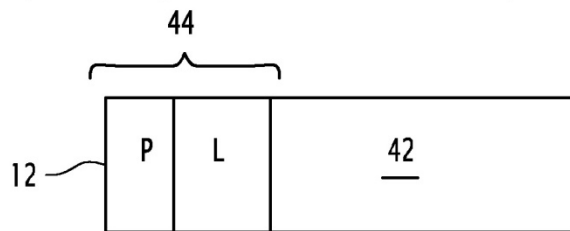
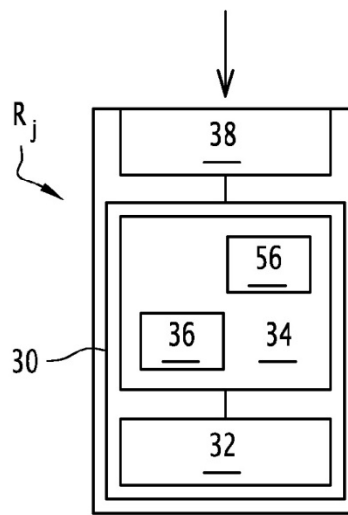
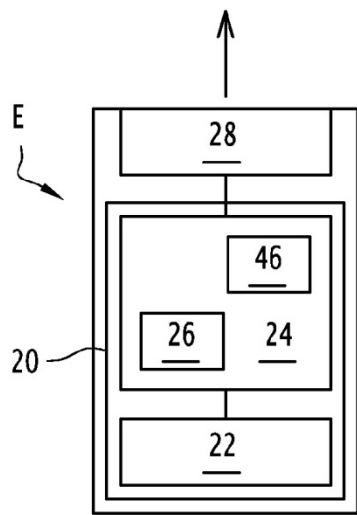
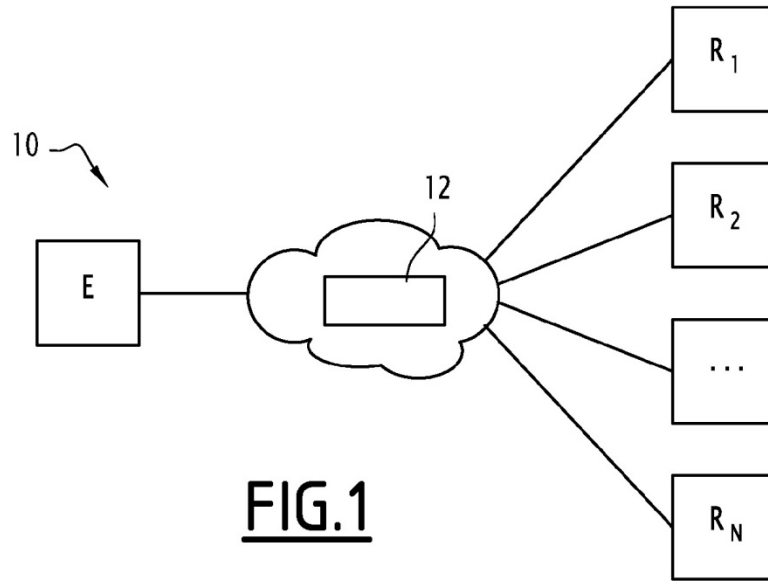
- definir (215) un valor máximo intermedio ( $max$ ) igual al número total de receptores (N) y una primera variable ( $j$ ) entera igual al número de identificadores (P) en la lista de identificadores (L) que se van a decodificar;

- calcular (220) el número (F) de subconjuntos de cardinal igual a la primera variable ( $j$ ) disminuido en uno,

15 en un conjunto de cardinal igual al valor máximo intermedio ( $max$ ) disminuido en uno;

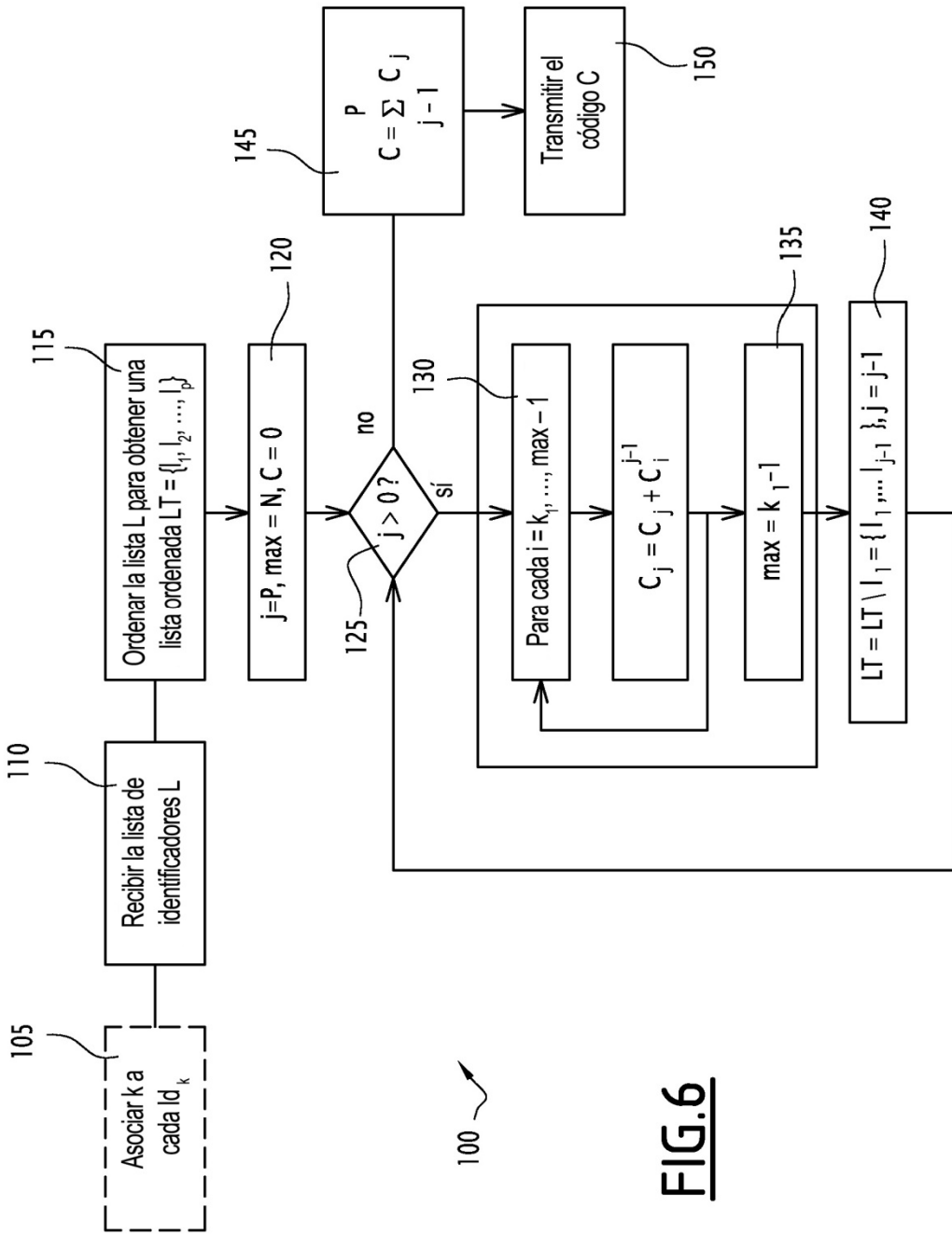
- si el valor del código es inferior a dicho número (F) de subconjuntos, añadir (230) en la lista de identificadores (L) el identificador ( $Id_k$ ) con un número ordinal (k) igual al valor máximo intermedio ( $max$ ), disminuir el valor máximo intermedio ( $max$ ) y la primera variable ( $j$ ) en uno;

20 - si el valor del código es superior o igual a dicho número (F) de subconjuntos, disminuir (240) el valor máximo intermedio ( $max$ ) en uno y calcular un nuevo valor del código igual a la diferencia entre el valor actual del código (C) y dicho número (F) de subconjuntos.



Código	1 <sup>er</sup> identificador	2 <sup>o</sup> identificador	3 <sup>er</sup> identificador
0	7	6	5
1	7	6	4
2	7	6	3
3	7	6	2
4	7	6	1
5	7	5	4
6	7	5	3
7	7	5	2
8	7	5	1
9	7	4	3
10	7	4	2
11	7	4	1
12	7	3	2
13	7	3	1
14	7	2	1
15	6	5	4
16	6	5	3
17	6	5	2
18	6	5	1
19	6	4	3
20	6	4	2
21	6	4	1
22	6	3	2
23	6	3	1
24	6	2	1
25	5	4	3
26	5	4	2
27	5	4	1
28	5	3	2
29	5	3	1
30	5	2	1
31	4	3	2
32	4	3	1
33	4	2	1
34	3	2	1

FIG.5



**FIG.6**



**FIG.7**

