

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 688 878**

51 Int. Cl.:

**G06F 9/30**

(2008.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **12.03.2014 PCT/US2014/023849**

87 Fecha y número de publicación internacional: **25.09.2014 WO14150636**

96 Fecha de presentación y número de la solicitud europea: **12.03.2014 E 14730231 (9)**

97 Fecha y número de publicación de la concesión europea: **25.07.2018 EP 2972792**

54 Título: **Modo de direccionamiento vertical de elementos indirecto vectorial con permutación horizontal**

30 Prioridad:

**15.03.2013 US 201313834785**

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

**07.11.2018**

73 Titular/es:

**QUALCOMM INCORPORATED (100.0%)  
5775 Morehouse Drive  
San Diego, CA 92121, US**

72 Inventor/es:

**INGLE, AJAY, ANANT;  
HOYLE, DAVID, J. y  
HOFFMAN, MARC, M.**

74 Agente/Representante:

**FORTEA LAGUNA, Juan José**

ES 2 688 878 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

Modo de direccionamiento vertical de elementos indirecto vectorial con permutación horizontal

5 **CAMPO DE DIVULGACIÓN**

[1] La presente divulgación se refiere en general a procesadores, y, más concretamente a elementos de procesamiento en un procesador.

10 **ANTECEDENTES**

[2] Un procesador puede recibir un flujo de datos y almacenar los datos en registros. El flujo de datos puede abarcar múltiples registros e incluir un subconjunto de datos que es de interés para el procesador. En un ejemplo, un usuario puede usar un teléfono que incluye un procesador. Una señal puede incluir datos asociados con el usuario y se puede enviar una señal de referencia desde una torre al teléfono. La señal de referencia puede entrelazarse con los datos del usuario junto con otros datos. Por ejemplo, la señal puede incluir la señal de referencia y datos asociados con otros usuarios. La señal puede incluir los datos del usuario junto con los datos de otros usuarios porque, si se pierde un paquete, se pierde una cantidad menor de los datos del usuario en comparación con un paquete completo que incluya los datos del usuario. Esto puede mitigar el efecto de perder un paquete en una base por usuario.

[3] Puede ser deseable que el procesador recupere eficazmente, por ejemplo, los datos de interés (por ejemplo, la señal de referencia) y organice los datos de interés que están entrelazados en el flujo de datos.

[4] La publicación de solicitud de patente de Estados Unidos n° US 2012/260062 se refiere a un sistema y procedimiento para dar a conocer un direccionamiento dinámico de elementos de datos en un archivo de registro con paralelismo de subpalabras. La publicación de solicitud de patente internacional n° WO 2006/033056 se refiere a un dispositivo de microprocesador y procedimiento para operaciones de mezcla. La publicación de solicitud de patente de Estados Unidos n° US 2012/060016 se refiere a cargas vectoriales de posiciones de memoria dispersas.

30 **BREVE SUMARIO**

[5] Esta divulgación se refiere a procesadores. Se dan a conocer procedimientos, sistemas y técnicas para procesar elementos en un procesador.

[6] Según un modo de realización, un procedimiento para colocar uno o más valores de datos de elementos en un vector de salida incluye identificar un vector de control de permutación vertical que incluye una pluralidad de elementos. Cada elemento de la pluralidad de elementos incluye una dirección de registro. El procedimiento también incluye, para cada elemento de la pluralidad de elementos, leer una dirección de registro del vector de control de permutación vertical. El procedimiento incluye además recuperar una pluralidad de valores de datos de elementos basándose en las direcciones de registro. El procedimiento también incluye identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones correspondientes a un vector de salida. El procedimiento incluye además colocar al menos algunos de los valores de datos de elementos recuperados de la pluralidad de valores de datos de elementos en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.

[7] Según otro modo de realización, un aparato incluye un procesador que se puede operar para identificar un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro. El procesador también se puede operar para, para cada elemento de la pluralidad de elementos, leer una dirección de registro del vector de control de permutación vertical. El procesador también se puede operar para recuperar una pluralidad de valores de datos de elementos basándose en las direcciones de registro. El procesador también se puede operar para identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones correspondientes a un vector de salida. El procesador también se puede operar para colocar al menos algunos de los valores de datos de elementos recuperados en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.

[8] Según otro modo de realización, un medio legible por ordenador tiene almacenadas en el mismo instrucciones ejecutables por ordenador para realizar operaciones, que incluyen identificar un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro; para cada elemento de la pluralidad de elementos, leer una dirección de registro del vector de control de permutación vertical; recuperar una pluralidad de valores de datos de elementos basándose en la lectura de la una o más direcciones de registro; identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones que corresponden a un vector de salida; y colocar al menos

algunos de los valores de datos de elementos recuperados de la pluralidad de valores de datos de elementos en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.

5 **[9]** Según otro modo de realización, un aparato para colocar uno o más valores de datos de elementos en un vector de salida incluye medios para identificar un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro; medios para, para cada elemento de la pluralidad de elementos, leer una dirección de registro del vector de control de permutación vertical; medios para recuperar una pluralidad de valores de datos de elementos basándose en la lectura de la una o más direcciones de registro; medios para identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones que corresponden a un vector de salida; y medios para colocar al menos algunos de los valores de datos de elementos recuperados de la pluralidad de valores de datos de elementos en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.

15 **[10]** Según otro modo de realización, un procedimiento para escribir valores de datos de elementos en un vector de salida incluye leer un vector de entrada que incluye una pluralidad de valores de datos de elementos. El procedimiento también incluye identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones. El procedimiento incluye además reorganizar al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal. El procedimiento también incluye colocar la pluralidad reorganizada de valores de datos de elementos en un vector temporal. El procedimiento incluye además identificar un vector de control de permutación vertical que incluye una pluralidad de elementos. Cada elemento de la pluralidad de elementos incluye una dirección de registro. El procedimiento también incluye colocar los valores de datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical.

30 **[11]** Según otro modo de realización, un aparato incluye un procesador que se puede operar para leer un vector de entrada que incluye una pluralidad de valores de datos de elementos. El procesador también se puede operar para identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones. El procesador también se puede operar para reorganizar al menos algunos de los valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal. El procesador también se puede operar para colocar la pluralidad reorganizada de valores de datos de elementos en un vector temporal. El procesador también se puede operar para identificar un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro. El procesador también se puede operar para colocar los valores de datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical.

40 **[12]** Según otro modo de realización, un medio legible por ordenador tiene almacenadas en el mismo instrucciones ejecutables por ordenador para realizar operaciones, que incluyen leer un vector de entrada que incluye una pluralidad de valores de datos de elementos; identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones; reorganizar al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal; colocar la pluralidad reorganizada de valores de datos de elementos en un vector temporal; identificar un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro; y colocar los valores de datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical.

50 **[13]** Según otro modo de realización, un aparato para escribir valores de datos de elementos en un vector de salida incluye medios para leer un vector de entrada que incluye una pluralidad de valores de datos de elementos; medios para identificar un vector de control de permutación horizontal que incluye un conjunto de direcciones; medios para reorganizar al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal; medios para colocar la pluralidad reorganizada de valores de datos de elementos en un vector temporal; medios para identificar un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro; y medios para colocar los valores de datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical.

## 60 **BREVE DESCRIPCIÓN DE LOS DIBUJOS**

65 **[14]** Los dibujos adjuntos, que forman parte de la memoria descriptiva, ilustran modos de realización de la invención y, junto con la descripción, sirven además para explicar los principios de los modos de realización. En los dibujos, los números de referencia similares pueden indicar elementos idénticos o funcionalmente similares.

El dibujo en el que aparece por primera vez un elemento en general se indica con el dígito más a la izquierda en el número de referencia correspondiente.

5 La FIG. 1 es un diagrama de bloques que ilustra un procesador que incluye un archivo de registros vectoriales, según un modo de realización.

La FIG. 2 es un diagrama de bloques que ilustra uno o más valores de datos de elementos que se colocan en un vector de salida, según un modo de realización.

10 La FIG. 3 es un diagrama de bloques que ilustra los valores de datos de elementos que se escriben en al menos un vector de salida, según un modo de realización.

15 La FIG. 4 es un diagrama de bloques que ilustra un archivo de registros vectoriales, según un modo de realización.

La FIG. 5 es un diagrama de flujo que ilustra un procedimiento para colocar uno o más valores de datos de elementos en un vector de salida, según un modo de realización.

20 La FIG. 6 es un diagrama de flujo que ilustra un procedimiento para escribir valores de datos de elementos en al menos un vector de salida, según un modo de realización.

La FIG. 7 es un diagrama de bloques que ilustra un dispositivo inalámbrico que incluye un procesador de señales digitales, según un modo de realización.

## 25 DESCRIPCIÓN DETALLADA

[15]

- 30 I. Visión general
- II. Instrucciones de ejemplo
  - A Instrucción de lectura de ejemplo
  - B Instrucción de escritura de ejemplo
  - 35 C Valores escalares
- III. Archivo de registro de ejemplo
- IV. Procedimientos de ejemplo
- V. Dispositivo inalámbrico de ejemplo

### 40 *I. Visión general*

[16] Debe entenderse que la siguiente divulgación da a conocer muchos modos de realización, o ejemplos, diferentes para implementar diferentes características de la presente divulgación. Algunos modos de realización pueden llevarse a la práctica sin algunos o todos estos detalles específicos. A continuación se describen ejemplos específicos de componentes, módulos y disposiciones para simplificar la presente divulgación. Por supuesto, son solo ejemplos y no pretenden ser limitativos.

50 [17] Un procesador puede incluir un archivo de registros vectoriales que incluye una pluralidad de registros vectoriales. Los registros vectoriales pueden almacenar valores de datos de elementos, y un subconjunto de los datos asociados con los valores de datos de elementos puede ser de interés. Puede ser deseable leer el subconjunto de los valores de datos de elementos de la pluralidad de registros vectoriales (por ejemplo, una señal de referencia entrelazada con datos de usuario) y distribuir el subconjunto de los valores de datos de elementos en un vector de salida.

55 [18] En un ejemplo, se pueden usar dos conjuntos de controles para recuperar el subconjunto de valores de datos de elementos y colocar uno o más del subconjunto de valores de datos de elementos en un vector de salida. El primer conjunto de controles puede ser un vector de control de permutación vertical que incluye direcciones de registro para determinar qué elementos de columna en los registros vectoriales de la pluralidad de registros vectoriales se deben leer. Por ejemplo, el vector de control de permutación vertical puede incluir las direcciones verticales de las que se deben leer los valores de datos de elementos. El segundo conjunto de controles puede ser un vector de control de permutación horizontal que incluye un conjunto de direcciones correspondientes a un vector de salida. El vector de control de permutación horizontal puede determinar la distribución horizontal del subconjunto recuperado de valores de datos de elementos en el vector de salida. Al menos parte del subconjunto recuperado de valores de datos de elementos puede colocarse en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.

[19] En otro ejemplo, se pueden usar dos conjuntos de controles para escribir valores de datos de elementos en al menos un vector de salida. En un ejemplo, puede leerse un vector de entrada que incluye una pluralidad de valores de datos de elementos. Al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada pueden reorganizarse basándose en el conjunto de direcciones en el vector de control de permutación horizontal, y los valores de datos de elementos reorganizados pueden colocarse en un vector temporal. Los valores de datos de elementos en el vector temporal pueden colocarse en al menos un vector de salida basándose en el vector de control de permutación vertical. El vector de control de permutación vertical puede incluir una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro de un registro vectorial en el archivo de registros vectoriales.

[20] La FIG. 1 es un diagrama de bloques que ilustra un procesador 110 que incluye un archivo de registros vectoriales 120, según un modo de realización.

[21] El archivo de registros vectoriales 120 puede incluir N registros vectoriales de propósito general, donde N es un número entero que es mayor que cero. Cada registro vectorial puede empaquetarse con un vector de M elementos, donde M es un número entero que es mayor que cero. Un elemento puede referirse a la granularidad más pequeña de datos que puede leerse del registro vectorial. Dentro de cada registro vectorial, los datos pueden escribirse o leerse, por ejemplo, como bytes (8 bits), palabras cortas (16 bits), palabras (32 bits) o palabras dobles (64 bits). En un ejemplo, el archivo de registros vectoriales 120 incluye 32 registros vectoriales (N=32) y 32 elementos de 32 bits (M=32), teniendo cada registro vectorial 1024 bits de ancho (32 elementos x 32 bits).

[22] En la FIG. 1, el archivo de registros vectoriales 120 incluye los registros vectoriales V0 (122), V1 (124), V2 (126), V3 (127), V4 (128), V5 (130), V6 (132) y V7 (134). Aunque el archivo de registros vectoriales 120 se ilustra en la FIG. 1 como que incluye ocho registros vectoriales, otros modos de realización que tienen menos de ocho o más de ocho registros vectoriales en el archivo de registros vectoriales 120 están dentro del alcance de esta divulgación. En un ejemplo, el archivo de registros vectoriales 120 incluye 32 registros vectoriales V0-V31.

[23] El procesador 110 puede cargar elementos en los registros vectoriales y leer valores de los registros vectoriales. En un ejemplo, un subconjunto de elementos asociados con los registros vectoriales V4, V5, V6 y V7 puede procesarse basándose en los vectores de control de permutación vertical y horizontal.

### **III. Instrucciones de ejemplo**

#### **A. Instrucción de lectura de ejemplo**

[24] El procesador 110 puede recibir una instrucción de lectura que coloca uno o más valores de datos de elementos en un vector de salida.

[25] La FIG. 2 es un diagrama de bloques 200 que ilustra uno o más valores de datos de elementos que se colocan en un vector de salida, según un modo de realización.

[26] El diagrama 200 incluye una instrucción de lectura de ejemplo 201 que incluye  $V_d = \text{vidx}$  ( $V_u$ ,  $V_v$ ). La instrucción 201 incluye leer dos registros vectoriales de entrada (por ejemplo,  $V_u$  y  $V_v$ ) y escribir una salida en un registro vectorial de salida (por ejemplo,  $V_d$ ). Los registros vectoriales  $V_u$ ,  $V_v$  y  $V_d$  pueden ser cualquiera de los registros vectoriales V0-V7 del archivo de registros vectoriales 120, donde u, v y d pueden ser cualquier valor de cero a siete. La instrucción puede convertirse en un código de operación que especifica un valor particular para  $V_u$ ,  $V_v$  y  $V_d$ . En un ejemplo,  $V_u = V_0$ ,  $V_v = V_1$  y  $V_d = V_3$ .  $V_u$  puede incluir el vector de control de permutación vertical,  $V_v$  puede incluir el vector de control de permutación horizontal, y  $V_d$  puede incluir el vector de salida y cada uno de los registros vectoriales. Cada uno de  $V_u$ ,  $V_x$  y  $V_v$  puede ser cualquiera de los registros vectoriales en el archivo de registros vectoriales 120.

[27] El procesador 110 puede recibir y ejecutar la instrucción 201. Ejecutar la instrucción 201 puede incluir identificar el vector de control de permutación vertical, leer una o más direcciones de memoria del vector de control de permutación vertical, y recuperar una pluralidad de valores de datos de elementos basándose en las direcciones de memoria leídas. Ejecutar la instrucción 201 también puede incluir identificar el vector de control de permutación horizontal y colocar al menos algunos de los valores de datos de elementos recuperados en el vector de salida basándose en el vector de control de permutación horizontal.

[28] El diagrama 200 incluye una pluralidad de registros vectoriales que pueden incluirse en el archivo de registros vectoriales 120. La pluralidad de registros vectoriales incluye los registros vectoriales V4, V5, V6 y V7, y cada registro vectorial puede incluir valores de datos de elementos. Puede ser deseable recuperar un subconjunto de los valores de datos de elementos almacenados en la pluralidad de registros vectoriales.

[29] El procesador 110 puede identificar un vector de control de permutación vertical 202 que incluye una pluralidad de elementos. En un modo de realización, el procesador 110 recibe datos asociados con una

instrucción y divide los datos recibidos en la pluralidad de elementos. La pluralidad de elementos se puede colocar en el vector de control de permutación vertical, incluyendo cada elemento de la pluralidad de elementos una dirección de registro.

5 **[30]** En la FIG. 2, el vector de control de permutación vertical 202 puede almacenarse en el registro vectorial Vu 122 del archivo de registros vectoriales 120. En consecuencia, cuando el procesador 110 identifica el vector de control de permutación vertical, el procesador 110 puede identificar el registro vectorial Vu en el archivo de registros vectoriales 120. Los registros vectoriales en el archivo de registros vectoriales 120 pueden ser visibles para un programador y pueden usarse para leer indirectamente de los registros vectoriales en el archivo de registros vectoriales 120. Con referencia al ejemplo anterior, solo un subconjunto de valores de datos de elementos en los registros vectoriales V4, V5, V6 y V7 pueden almacenar la señal de referencia, y el resto de los valores de datos de elementos pueden incluir otros datos que no son de interés. El subconjunto de valores de datos de elementos de interés puede abarcar múltiples registros vectoriales. Para recuperar estos valores de datos de elementos particulares, el programador puede programar este patrón en Vu para extraer los valores de datos de elementos de interés almacenados en las direcciones de registro apropiadas.

15 **[31]** El registro vectorial Vu puede incluir 32 elementos, y cada elemento puede incluir una dirección de registro de 32 bits. Por ejemplo, Vu[0]=4 e incluye la dirección de registro para el registro vectorial V4[0], Vu[1]=5 e incluye la dirección de registro para el registro vectorial V5[1], Vu[2]=6 e incluye la dirección de registro para el registro vectorial V6[2], y Vu[3]=7 e incluye la dirección de registro para el registro vectorial V7[3].

20 **[32]** Se puede recuperar una pluralidad de valores de datos de elementos basándose en la lectura de las direcciones de registro. En particular, la dirección de registro en el vector de control de permutación vertical se puede usar como una dirección indirecta para recuperar el valor de datos de elementos apropiado del archivo de registros vectoriales. De esta forma, la una o más direcciones de registro almacenadas en el vector de control de permutación vertical 202 pueden usarse como un acceso indirecto al valor de datos de elementos almacenado en la dirección de registro correspondiente. Una lectura indirecta puede ser contraria a una lectura directa, que especifica un valor de datos de elementos en lugar de una dirección de registro. Para una lectura indirecta, la dirección de registro puede leerse y usarse para recuperar el valor de datos de elementos de interés.

25 **[33]** En un ejemplo, después de leer la dirección de registro para el registro vectorial V4 de Vu[0], el procesador 110 puede recuperar el valor de datos de elementos "i0", que es el valor de datos de elementos almacenado en el registro vectorial V4[0]. De forma similar, después de leer la dirección de registro para el registro vectorial V5 de Vu[1], el procesador 110 puede recuperar el valor de datos de elementos "j1", que es el valor de datos de elementos almacenado en el registro vectorial V5[1]. De forma similar, después de leer la dirección de registro para el registro vectorial V6 de Vu[2], el procesador 110 puede recuperar el valor de datos de elementos "k2", que es el valor de datos de elementos almacenado en el registro vectorial V6[2]. De forma similar, después de leer la dirección de registro para el registro vectorial V7 de Vu[3], el procesador 110 puede recuperar el valor de datos de elementos "l3", que es el valor de datos de elementos almacenado en el registro vectorial V7[3]. El procesador 110 puede realizar acciones similares para las otras direcciones de registro almacenadas en Vu. En un modo de realización, el procesador 110 es un procesador de instrucción única y múltiples datos (SIMD) y lleva a cabo las instrucciones descritas en esta divulgación en paralelo. Por ejemplo, el procesador 110 puede leer las direcciones de registro en Vv en paralelo y recupera los valores de datos de elementos basándose en la dirección de registro leída en paralelo.

30 **[34]** El procesador 110 puede colocar la pluralidad recuperada de valores de datos de elementos en un registro vectorial temporal. El diagrama 200 incluye un registro vectorial Vtmp 204. El procesador 110 puede ser un procesador canalizado, y Vtmp 204 puede ser un registro vectorial canalizado temporal usado para almacenar resultados intermedios. En otro ejemplo, Vtmp 204 puede ser un registro vectorial en el archivo de registros vectoriales 120. Los valores de datos de elementos recuperados basándose en las direcciones de registro en Vu pueden colocarse en Vtmp 204.

35 **[35]** Por ejemplo, en la FIG. 2 el "i0" que se leyó del registro vectorial V4[0] se coloca en Vtmp[0], el "j1" que se leyó del registro vectorial V5[1] se coloca en Vtmp[1], el "k2" que se leyó del registro vectorial V6[2] se coloca en Vtmp[2], y el "l3" que se leyó del registro vectorial V7[3] se coloca en Vtmp[3]. El procesador 110 puede realizar acciones similares para el resto de valores de datos de elementos recuperados basándose en las direcciones de registro en el vector de control de permutación vertical.

40 **[36]** En un modo de realización, el procesador 110 identifica un vector de control de permutación horizontal 206 que incluye un conjunto de direcciones que corresponden a un vector de salida. El vector de control de permutación horizontal 206 tiene un conjunto de controles que determinan la colocación de los elementos de valores de datos recuperados en el vector de salida. El procesador 110 puede colocar al menos algunos de los valores de datos de elementos recuperados de la pluralidad de valores de datos de elementos en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal. Por ejemplo, los valores de datos de elementos en Vtmp 204 pueden permutarse y colocarse en el vector de salida. En un ejemplo, la permutación horizontal se implementa usando una barra cruzada. En este ejemplo, al menos algunos

de los valores de datos de elementos recuperados se pueden colocar en el vector de salida usando la barra cruzada.

5 **[37]** En la FIG. 2, el vector de control de permutación horizontal puede almacenarse en el registro vectorial Vv (124), y el vector de salida puede ser Vd (126) del archivo de registros vectoriales 120.

10 **[38]** En un ejemplo, Vv[0]=3, que indica que el valor de datos de elementos correspondiente a Vtmp[3] pertenece a Vd[0]. En consecuencia, el valor de datos de elementos correspondiente a Vtmp[3], se coloca en el vector de salida Vd[0]. De forma similar, Vv[1]=1, que indica que el valor de datos de elementos correspondiente a Vtmp[1] pertenece a Vd[1]. En consecuencia, "j1", el valor de datos de elementos correspondiente a Vtmp[1], se coloca en el vector de salida Vd[1]. De forma similar, Vv[2]=0, que indica que el valor de datos de elementos correspondiente a Vtmp[0] pertenece a Vd[2]. En consecuencia, "i0", el valor de datos de elementos correspondiente a Vtmp[0], se coloca en el vector de salida Vd[2]. De forma similar, Vv[3]=28, que indica que el valor de datos de elementos correspondiente a Vtmp[28] pertenece a Vd[3]. En consecuencia, "i28", el valor de datos de elementos correspondiente a Vtmp[28], se coloca en el vector de salida Vd[3]. El procesador 110 puede realizar acciones similares para colocar los otros valores de datos de elementos en Vtmp 204 en el vector de salida Vd 126.

20 **[39]** El registro vectorial 120 puede tener un puerto de lectura al que se le da una dirección. En un ejemplo, en un primer ciclo, el procesador 110 lee la dirección de registro del vector de control de permutación vertical (por ejemplo, el registro vectorial Vu en el archivo de registros vectoriales 120 de la FIG. 1). El procesador 110 puede realizar 32 operaciones diferentes en una instrucción del mismo tipo. Por ejemplo, si el registro vectorial Vu incluye el vector de control de permutación vertical, el procesador 110 puede realizar la operación de lectura en Vu[0]-Vu[31] de tal manera que el procesador 110 lee simultáneamente las 32 direcciones de registro del registro vectorial Vu. Internamente en el siguiente ciclo, el procesador 110 puede leer de nuevo el archivo de registros vectoriales 120 basándose en las direcciones de registro leídas para recuperar los valores de datos de elementos apropiados. El procesador 110 puede realizar la operación de recuperación de tal manera que el procesador 110 recupera simultáneamente los 32 valores de datos de elementos basándose en las direcciones de registro leídas del registro vectorial Vu. El procesador 110 puede colocar simultáneamente los 32 valores de datos de elementos recuperados en Vtmp.

35 **[40]** Los 32 valores de datos de elementos recuperados almacenados en Vtmp pueden distribuirse horizontalmente en el vector de salida usando el vector de control de permutación horizontal. El vector de control de permutación horizontal puede incluir el conjunto de direcciones correspondientes al vector de salida. El procesador 110 puede identificar el vector de control de permutación horizontal y colocar al menos algunos de los valores de datos de elementos recuperados de la pluralidad de valores de datos de elementos en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal. El procesador 110 puede colocar simultáneamente los 32 valores de datos de elementos recuperados en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal. El vector de salida puede incluir entonces los valores de datos de elementos de interés.

### **B. Instrucción de escritura de ejemplo**

45 **[41]** El procesador 110 también puede recibir una instrucción de escritura que escribe valores de datos de elementos en al menos un registro vectorial. El procesador 110 puede usar el vector de control de permutación vertical y el vector de control de permutación horizontal para escribir valores de datos de elementos en al menos un registro vectorial. La instrucción de escritura puede ser una versión de escritura análoga a la instrucción de lectura analizada anteriormente.

50 **[42]** La FIG. 3 es un diagrama de bloques 300 que ilustra los valores de datos de elementos que se escriben en al menos un registro vectorial, según un modo de realización.

55 **[43]** El diagrama 300 incluye una instrucción de escritura de ejemplo 301 que incluye  $V_{idx}(V_d, V_v)=V_u$ . La instrucción 301 incluye leer un registro vectorial de entrada (por ejemplo, el registro vectorial Vu) y escribir valores de datos de elementos del registro vectorial de entrada en al menos un registro vectorial (por ejemplo, los registros vectoriales V4, V5, V6 y V7) basándose en el vector de control de permutación vertical (por ejemplo, el registro vectorial Vd) y el vector de control de permutación horizontal (por ejemplo, el registro vectorial Vv).

60 **[44]** Los registros vectoriales Vu, Vv y Vd pueden ser cualquiera de los registros vectoriales V0-V7 del archivo de registros vectoriales 120, donde u, v y d pueden ser cualquier valor de cero a siete. La instrucción puede convertirse en un código de operación que especifica un valor particular para Vu, Vv y Vd. En un ejemplo,  $V_u=V_0$ ,  $V_v=V_1$  y  $V_d=V_3$ . Vu puede incluir el registro vectorial de entrada, Vv puede incluir el vector de control de permutación horizontal, y Vd puede incluir el vector de control de permutación vertical.

65 **[45]** El procesador 110 puede recibir y ejecutar la instrucción 301. Ejecutar la instrucción 301 puede incluir leer un vector de entrada Vu (122) que incluye una pluralidad de valores de datos de elementos. Por ejemplo,

$Vu[0]=u0$ ,  $Vu[1]=u1$ ,  $Vu[2]=u2$  y  $Vu[3]=u3$ . En un ejemplo, el vector de entrada incluye los valores de datos de elementos reales que se escriben en al menos un vector de salida.

**[46]** Los valores de datos de elementos en  $Vu$  pueden leerse y permutarse basándose en el vector de control de permutación horizontal. El procesador 110 puede identificar el vector de control de permutación horizontal 206 que incluye un conjunto de direcciones. El vector de control de permutación horizontal 206 puede almacenarse en el registro vectorial  $Vv$  (124) del archivo de registros vectoriales 120. En consecuencia, cuando el procesador 110 identifica el vector de control de permutación horizontal, el procesador 110 puede identificar el registro vectorial  $Vv$  en el archivo de registros vectoriales 120. Al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada  $Vu$  pueden reorganizarse basándose en el conjunto de direcciones en el vector de control de permutación horizontal 206. La pluralidad de valores de datos de elementos reorganizados pueden colocarse en un vector temporal  $Vtmp$  204.

**[47]** En un ejemplo, la dirección de registro de  $Vv[0]=3$ , y el procesador 110 puede recuperar el valor de datos de elementos "u3" almacenado en  $Vu[3]$  y colocar el valor de datos de elementos recuperado en  $Vtmp[0]$ . De manera similar, la dirección de registro de  $Vv[1]=1$ , y el procesador 110 puede recuperar el valor de datos de elementos "u1" almacenado en  $Vu[1]$  y colocar el valor de datos de elementos recuperado en  $Vtmp[1]$ . De manera similar, la dirección de registro de  $Vv[2]=0$ , y el procesador 110 puede recuperar el valor de datos de elementos "u0" almacenado en  $Vu[0]$  y colocar el valor de datos de elementos recuperado en  $Vtmp[2]$ . El procesador 110 puede realizar acciones similares para colocar los otros valores de datos de elementos en  $Vu$  en el vector temporal  $Vtmp$ .

**[48]** Ejecutar la instrucción 301 también puede incluir identificar un vector de control de permutación vertical 202 que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro. El vector de control de permutación vertical 202 puede almacenarse en el registro vectorial  $Vd$  (126) del archivo de registros vectoriales 120. Los valores de datos de elementos en  $Vtmp$  se pueden colocar en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical. En particular, las direcciones de registro en  $Vd$  pueden usarse para escribir los valores de datos de elementos en  $Vtmp$  de una manera vertical en al menos un registro vectorial.

**[49]** Por ejemplo, en la FIG. 3,  $Vd[0]=4$ , que indica que el valor de datos de elementos correspondiente a  $Vtmp[0]$  pertenece a  $V4[0]$ . En consecuencia, "i0", el valor de datos de elementos correspondiente a  $Vtmp[0]$ , se coloca en el registro vectorial  $V4[0]$ . De forma similar,  $Vd[1]=5$ , que indica que el valor de datos de elementos correspondiente a  $Vtmp[1]$  pertenece a  $V5[1]$ . En consecuencia, "j1", el valor de datos de elementos correspondiente a  $Vtmp[1]$ , se coloca en el registro vectorial  $V5[1]$ . De forma similar,  $Vd[2]=6$ , que indica que el valor de datos de elementos correspondiente a  $Vtmp[2]$  pertenece a  $V6[2]$ . En consecuencia, "k2", el valor de datos de elementos correspondiente a  $Vtmp[2]$ , se coloca en el registro vectorial  $V6[2]$ . De forma similar,  $Vd[3]=7$ , que indica que el valor de datos de elementos correspondiente a  $Vtmp[3]$  pertenece a  $V7[3]$ . En consecuencia, el valor de datos de elementos correspondiente a  $Vtmp[3]$ , se coloca en el registro vectorial  $V7[3]$ . El procesador 110 puede realizar acciones similares para colocar los otros valores de datos de elementos en  $Vtmp$  204 en uno o más registros vectoriales (por ejemplo, los registros vectoriales  $V4$ ,  $V5$ ,  $V6$  y  $V7$ ). Los registros vectoriales pueden almacenar entonces en los lugares apropiados los valores de datos de elementos del vector de entrada.

**[50]** Como se ha analizado anteriormente y se ha resaltado adicionalmente aquí, las FIG. 1-3 son solo ejemplos, que no deberían limitar indebidamente el alcance de las reivindicaciones.

**[51]** Como se ha ilustrado anteriormente, los vectores de control de permutación (por ejemplo, el vector de control de permutación vertical y/o el vector de control de permutación horizontal) se pueden almacenar en registros dinámicos que se actualizan cuando el procesador 110 recibe datos. Por ejemplo, en la FIG. 1, el vector de control de permutación vertical está incluido en un registro vectorial (por ejemplo,  $Vu$ ) y el vector de control de permutación horizontal también está incluido en un registro vectorial (por ejemplo,  $Vv$ ) del archivo de registros vectoriales. Esto no pretende ser limitativo, y los vectores de control de permutación se pueden recuperar de varias maneras.

**[52]** Por ejemplo, los patrones ilustrados en los vectores de control de permutación pueden ser diferentes de los ilustrados en las FIG. 2 y 3. Por ejemplo, el vector de control de permutación vertical en la FIG. 2 tiene un patrón que especifica la lectura de los valores de datos de elementos de los registros vectoriales  $V4$ ,  $V5$ ,  $V6$ ,  $V7$ ,  $V4$ ,  $V5$ ,  $V6$ ,  $V7$ , etc. Esto no pretende ser limitativo, y se puede usar cualquier patrón. Por ejemplo, el programador puede programar cualquier patrón en los vectores de control de permutación.

**[53]** Además, en la FIG. 3, todos los valores en el vector de entrada  $Vu$  se escriben en al menos un vector de salida. Esto no pretende ser limitativo. Por ejemplo, en otro modo de realización, uno o más valores de datos de elementos en el vector de entrada  $Vu$  no se escriben en un vector de salida. Además, los valores de datos de elementos en el vector de entrada  $Vu$  se pueden escribir en un único registro vectorial o en más de un registro vectorial. En un ejemplo, si el control de permutación vertical incluye las mismas direcciones de registro,

entonces todos los valores de datos de elementos en el Vtmp se pueden escribir en el mismo registro vectorial (por ejemplo, el registro vectorial V4).

### C. Valores escalares

5

**[54]** Adicionalmente, un vector de control de permutación (por ejemplo, el vector de control de permutación vertical o el vector de control de permutación horizontal) puede almacenarse en una posición de memoria distinta de un registro vectorial. En un ejemplo, el vector de control de permutación se puede almacenar en memoria de solo lectura (ROM) y leerse de la ROM. En una instrucción de lectura de ejemplo, el procesador 110 puede recibir una instrucción tal como  $Vd=Vidx(R, Vv)$ , donde R es un valor escalar que hace referencia a una estructura de datos en la ROM. El vector de control de permutación vertical puede estar almacenado en la estructura de datos (por ejemplo, una tabla) en la ROM. En consecuencia, para identificar el vector de control de permutación vertical, el procesador 110 puede identificar el vector de control de permutación vertical en la estructura de datos en la ROM usando el valor escalar.

10

15

**[55]** En otra instrucción de lectura de ejemplo, el procesador 110 puede recibir una instrucción tal como  $Vd=Vidx(Vu, R)$ , donde R es un valor escalar que hace referencia a una estructura de datos en la ROM. El vector de control de permutación horizontal puede estar almacenado en la estructura de datos (por ejemplo, una tabla) en la ROM. En consecuencia, para identificar el vector de control de permutación horizontal, el procesador 110 puede identificar el vector de control de permutación horizontal en la estructura de datos en la ROM usando el valor escalar.

20

25

**[56]** En otro ejemplo de lectura, el procesador 110 puede recibir una instrucción tal como  $Vd=Vidx(R1, R2)$ , donde R1 es un primer valor escalar que hace referencia a una primera estructura de datos en la ROM, y R2 es un segundo valor escalar que hace referencia a una segunda estructura de datos en la ROM. El vector de control de permutación vertical puede estar almacenado en la primera estructura de datos (por ejemplo, una tabla) en la ROM, y el vector de control de permutación horizontal puede estar almacenado en la segunda estructura de datos (por ejemplo, una tabla) en la ROM. Por consiguiente, para identificar el vector de control de permutación vertical, el procesador 110 puede identificar el vector de control de permutación vertical en la primera estructura de datos en la ROM usando el primer valor escalar, y para identificar el vector de control de permutación horizontal, el procesador 110 puede identificar el vector de control de permutación horizontal en la segunda estructura de datos en la ROM usando el segundo valor escalar.

30

35

**[57]** Del mismo modo, las instrucciones de escritura también pueden incluir valores escalares. Por ejemplo, el procesador 110 puede recibir una instrucción tal como  $Vidx(R, Vv)=Vu$ , donde R es un valor escalar que hace referencia a una estructura de datos en la ROM. El vector de control de permutación vertical puede estar almacenado en la estructura de datos (por ejemplo, una tabla) en la ROM. En consecuencia, para identificar el vector de control de permutación vertical, el procesador 110 puede identificar el vector de control de permutación vertical en la estructura de datos en la ROM usando el valor escalar.

40

45

**[58]** En otra instrucción de escritura de ejemplo, el procesador 110 puede recibir una instrucción tal como  $Vidx(Vd, R)=Vu$ , donde R es un valor escalar que hace referencia a una estructura de datos en la ROM. El vector de control de permutación horizontal puede estar almacenado en la estructura de datos (por ejemplo, una tabla) en la ROM. En consecuencia, para identificar el vector de control de permutación horizontal, el procesador 110 puede identificar el vector de control de permutación horizontal en la estructura de datos en la ROM usando el valor escalar.

50

55

**[59]** En otro ejemplo de escritura, el procesador 110 puede recibir una instrucción tal como  $Vidx(R1, R2)=Vu$ , donde R1 es un primer valor escalar que hace referencia a una primera estructura de datos en la ROM, y R2 es un segundo valor escalar que hace referencia a una segunda estructura de datos en la ROM. El vector de control de permutación vertical puede estar almacenado en la primera estructura de datos (por ejemplo, una tabla) en la ROM, y el vector de control de permutación horizontal puede estar almacenado en la segunda estructura de datos (por ejemplo, una tabla) en la ROM. Por consiguiente, para identificar el vector de control de permutación vertical, el procesador 110 puede identificar el vector de control de permutación vertical en la primera estructura de datos en la ROM usando el primer valor escalar, y para identificar el vector de control de permutación horizontal, el procesador 110 puede identificar el vector de control de permutación horizontal en la segunda estructura de datos en la ROM usando el segundo valor escalar.

60

**[60]** Esto puede ser ventajoso en algunos modos de realización, porque puede ser innecesario usar un registro vectorial para almacenar el vector de control de permutación. Así pues, los registros vectoriales pueden liberarse para almacenar otros datos. Adicionalmente, los valores del vector de control de permutación pueden ser valores almacenados previamente de tal manera que puede ser innecesario consumir ciclos de cálculo para programar el patrón en el vector de control de permutación.

65

**[61]** El valor escalar se puede especificar en una instrucción. En un modo de realización, el procesador 110 es un coprocesador y recibe el valor escalar de otro procesador. En un modo de realización, el procesador 110

puede recuperar el valor escalar de la memoria principal o ROM. Además, el procesador 110 puede recibir el vector de control de permutación vertical y/o el vector de control de permutación horizontal del otro procesador.

### III. Archivo de registro de ejemplo

5

**[62]** La FIG. 4 es un diagrama de bloques que ilustra un archivo de registros vectoriales, según un modo de realización.

10

**[63]** En un modo de realización, el procesador 110 es un procesador SIMD que lee de y escribe en registros vectoriales en el archivo de registros vectoriales 120. En un ejemplo, cada registro vectorial en el archivo de registros vectoriales 120 puede incluir Palabra 0 (W0) 122 que tiene P bits de ancho, Palabra 1 (W1) 124 que tiene P bits de ancho, ..., hasta Palabra 31 (W31) que tiene P bits de ancho. P puede ser un número entero mayor que cero. Los registros vectoriales pueden construirse a partir de la granularidad más pequeña de bytes. En un ejemplo, el archivo de registros vectoriales 120 incluye 32 registros vectoriales, y cada registro vectorial tiene 1024 bits de ancho. Cada registro vectorial puede incluir 32 palabras, incluyendo cada palabra 32 bits.

15

**[64]** En un ejemplo, P puede ser cualquier múltiplo de 8 (por ejemplo, 8, 16, 32, 64, etc.). Incluso si P=32, los controles de permutación pueden ser adyacentes entre sí de tal manera que se pueden usar construcciones mayores de 32 bits de datos. Por ejemplo, si las palabras tienen 32 bits de ancho y es deseable gestionar datos que tienen 64 bits de ancho, el mismo control en los fragmentos de 32 bits puede colocarse de manera adyacente, de tal manera que el sistema puede gestionar y permutar 64 bits de datos.

20

### IV. Procedimientos de ejemplo

25

**[65]** La FIG. 5 es un diagrama de flujo que ilustra un procedimiento 500 para colocar uno o más valores de datos de elementos en un vector de salida, según un modo de realización. El procedimiento 500 no pretende ser limitativo, y puede usarse en otras aplicaciones.

30

**[66]** El procedimiento 500 incluye las etapas 510-550. En una etapa 510, se identifica un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro. En un ejemplo, el procesador 110 identifica un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro. Las direcciones de registro almacenadas en el vector de control de permutación vertical pueden usarse para leer valores de datos de elementos de uno o más registros vectoriales en el archivo de registros vectoriales. El vector de control de permutación vertical puede almacenarse en un registro dinámico o en memoria estática. Por ejemplo, en la instrucción de lectura en la FIG. 2, el procesador 110 puede identificar Vu del archivo de registros vectoriales como el vector de control de permutación vertical. En otro ejemplo, el procesador recupera el vector de control de permutación vertical de la ROM.

35

40

**[67]** En una etapa 520, para cada elemento de la pluralidad de elementos, se lee una dirección de registro del vector de control de permutación vertical. En un ejemplo, para cada elemento de la pluralidad de elementos, el procesador 110 lee una dirección de registro del vector de control de permutación vertical. Las direcciones de registro en el vector de control de permutación vertical se pueden usar para leer de uno o más registros vectoriales de una manera vertical. Los valores de datos de elementos que son de interés para el procesador pueden ubicarse en las direcciones de registro.

45

50

**[68]** En una etapa 530, se recupera una pluralidad de valores de datos de elementos basándose en las direcciones de registro leídas. En un ejemplo, el procesador 110 recupera una pluralidad de valores de datos de elementos basándose en las direcciones de registro leídas. La pluralidad recuperada de valores de datos de elementos puede colocarse en un vector temporal. Por ejemplo, en la FIG. 2, la dirección del registro vectorial V4[0] se almacena en el vector de control de permutación vertical en Vu[0]. El valor de datos de elementos en el registro vectorial V4[0] puede colocarse entonces en el vector temporal. Los valores de datos de elementos recuperados pueden ser de interés para el procesador.

55

60

**[69]** En una etapa 540, se identifica un vector de control de permutación horizontal que incluye un conjunto de direcciones correspondientes a un vector de salida. En un ejemplo, el procesador 110 identifica un vector de control de permutación horizontal que incluye un conjunto de direcciones que corresponden a un vector de salida. El conjunto de direcciones almacenadas en el vector de control de permutación horizontal puede usarse para organizar la pluralidad recuperada de valores de datos de elementos (por ejemplo, almacenados en el registro vectorial temporal) en el vector de salida de una manera horizontal. El vector de control de permutación horizontal puede almacenarse en un registro dinámico o en memoria estática. Por ejemplo, en la instrucción de lectura en la FIG. 2, el procesador 110 puede identificar Vv del archivo de registros vectoriales como el vector de control de permutación horizontal. En otro ejemplo, el procesador recupera el vector de control de permutación horizontal de la ROM.

65

65

**[70]** En una etapa 550, al menos algunos de los valores de datos de elementos recuperados de la pluralidad de valores de datos de elementos se colocan en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal. En un ejemplo, el procesador 110 coloca al menos algunos de los valores de datos de elementos recuperados de la pluralidad de valores de datos de elementos en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal. Los valores de datos de elementos de interés pueden haber abarcado originalmente múltiples registros con otros datos dispersos en el medio. El vector de salida puede almacenar entonces solo los valores de datos de elementos de interés.

**[71]** También se entiende que pueden realizarse etapas de procedimiento adicionales antes, durante o después de las etapas 510-550 analizadas anteriormente. Por ejemplo, el procedimiento 500 puede incluir etapas de escribir valores de datos de elementos en al menos un vector de salida. También se entiende que una o más de las etapas del procedimiento 500 descrito en el presente documento pueden omitirse, combinarse o realizarse en una secuencia diferente según se desee.

**[72]** La FIG. 6 es un diagrama de flujo que ilustra un procedimiento 600 para escribir valores de datos de elementos en un registro vectorial, según un modo de realización. El procedimiento 600 no pretende ser limitativo, y puede usarse en otras aplicaciones.

**[73]** El procedimiento 600 incluye las etapas 610-660. En una etapa 610, se lee un vector de entrada que incluye una pluralidad de valores de datos de elementos. En un ejemplo, el procesador 110 lee un vector de entrada que incluye una pluralidad de valores de datos de elementos. Por ejemplo, en la FIG. 3, el procesador 110 puede identificar Vu del archivo de registros vectoriales como el vector de entrada. En otro ejemplo, el procesador recupera el vector de entrada de la ROM. Uno o más de los valores de datos de elementos almacenados en el vector de entrada pueden escribirse en al menos un registro vectorial basándose en el vector de control de permutación horizontal y el vector de control de permutación vertical.

**[74]** En una etapa 620, se identifica un vector de control de permutación horizontal que incluye un conjunto de direcciones. En un ejemplo, el procesador 110 identifica un vector de control de permutación horizontal que incluye un conjunto de direcciones. El vector de control de permutación horizontal puede almacenarse en un registro dinámico o en memoria estática. Por ejemplo, en la instrucción de escritura en la FIG. 3, el procesador 110 puede identificar el registro vectorial Vv del archivo de registros vectoriales como el vector de control de permutación horizontal. En otro ejemplo, el procesador recupera el vector de control de permutación horizontal de la ROM.

**[75]** En una etapa 630, al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada se reorganizan basándose en el conjunto de direcciones en el vector de control de permutación horizontal. En un ejemplo, el procesador 110 reorganiza al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal. En particular, el conjunto de direcciones almacenadas en el vector de control de permutación horizontal puede usarse para reorganizar la pluralidad de valores de datos de elementos en el vector de entrada de una manera horizontal.

**[76]** En una etapa 640, la pluralidad reorganizada de valores de datos de elementos se coloca en un vector temporal. En un ejemplo, el procesador 110 coloca la pluralidad reorganizada de valores de datos de elementos en un vector temporal. Por ejemplo, en la FIG. 3, Vtmp 204 puede ser el vector temporal que almacena la pluralidad reorganizada de valores de datos de elementos.

**[77]** En una etapa 650, se identifica un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro. En un ejemplo, el procesador 110 identifica un vector de control de permutación vertical que incluye una pluralidad de elementos, incluyendo cada elemento de la pluralidad de elementos una dirección de registro. Las direcciones de registro almacenadas en el vector de control de permutación vertical pueden usarse para escribir valores de datos de elementos del vector temporal en al menos un registro vectorial de una manera vertical. El vector de control de permutación vertical puede almacenarse en un registro dinámico o en memoria estática. Por ejemplo, en la instrucción de escritura en la FIG. 3, el procesador 110 puede identificar el registro vectorial Vd del archivo de registros vectoriales como el vector de control de permutación vertical. En otro ejemplo, el procesador recupera el vector de control de permutación vertical de la ROM.

**[78]** En una etapa 660, los valores de datos de elementos en el vector temporal se colocan en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical. En un ejemplo, el procesador 110 coloca los valores de los datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical. Por ejemplo, en la FIG. 3, los valores de datos de elementos en el vector temporal se colocan en los registros vectoriales V4, V5, V6 y/o V7 basándose en las direcciones de registro en el registro vectorial Vd. Los valores de

elementos de datos que se almacenaron en el vector de entrada Vu se pueden escribir entonces apropiadamente en uno o más registros vectoriales.

5 [79] También se entiende que pueden realizarse etapas de procedimiento adicionales antes, durante o después de las etapas 610-660 analizadas anteriormente. Por ejemplo, el procedimiento 600 puede incluir una etapa de colocar uno o más valores de datos de elementos en un registro vectorial. También se entiende que una o más de las etapas del procedimiento 600 descrito en el presente documento pueden omitirse, combinarse o realizarse en una secuencia diferente según se desee.

## 10 V. Dispositivo inalámbrico de ejemplo

15 [80] La FIG. 7 es un diagrama de bloques que ilustra un dispositivo inalámbrico 700 que incluye un procesador de señales digitales, según un modo de realización. El dispositivo 700 incluye un procesador, tal como un procesador de señales digitales (DSP) 701. La instrucción 201 y/o la instrucción 301 pueden almacenarse en una memoria 750, y el VRF 120 puede estar incluido en el DSP 701. En un ejemplo, el DSP 701 y el VFR 129 pueden procesar la instrucción 201 y/o la instrucción 301 según una o más de las FIG. 1-4, y/o según uno o más de los procedimientos de las FIG. 5 y 6, o cualquier combinación de los mismos.

20 [81] La FIG. 7 también muestra un controlador de pantalla 730 que está acoplado al DSP 701 y a una pantalla 732. También se puede acoplar un codificador/descodificador (CÓDEC) 734 al DSP 701. Un altavoz 736 y un micrófono 738 se pueden acoplar al CÓDEC 734. Adicionalmente, un controlador inalámbrico 740 se puede acoplar al DSP 701 y a una antena inalámbrica 748. En un modo de realización, el DSP 701, el controlador de pantalla 730, la memoria 750, el CÓDEC 734 y el controlador inalámbrico 740 están incluidos en un dispositivo de sistema en paquete o sistema en chip 756.

25 [82] En un modo de realización, un dispositivo de entrada 730 y una fuente de alimentación 760 están acoplados al dispositivo de sistema en chip 756. Además, en un modo de realización, como se ilustra en la FIG. 7, la pantalla 732, el dispositivo de entrada 730, el altavoz 736, el micrófono 738, la antena inalámbrica 748 y la fuente de alimentación 760 son externos con respecto al dispositivo de sistema en chip 756. Cada uno de la pantalla 732, el dispositivo de entrada 730, el altavoz 736, el micrófono 738, la antena inalámbrica 748 y la fuente de alimentación 760 se puede acoplar a un componente del dispositivo de sistema en chip 756, tal como una interfaz o un controlador.

30 [83] Los expertos apreciarán además que los diversos bloques lógicos, configuraciones, módulos, circuitos y etapas ilustrativos descritos en conexión con los modos de realización divulgados en el presente documento se pueden implementar como hardware electrónico, software informático o combinaciones de ambos. Para ilustrar claramente esta intercambiabilidad de hardware y software, anteriormente se han descrito diversos componentes, bloques, configuraciones, módulos, circuitos y etapas ilustrativos en general desde el punto de vista de su funcionalidad. Si dicha funcionalidad se implementa como hardware o software depende de la aplicación y las restricciones de diseño particulares impuestas al sistema global. Los expertos en la técnica pueden implementar la funcionalidad descrita de varias maneras para cada aplicación particular, pero no se debería interpretar que dichas decisiones de implementación suponen apartarse del alcance de la presente divulgación.

35 [84] Las etapas de un procedimiento descrito en relación con los modos de realización divulgados en el presente documento se pueden realizar directamente en hardware, en un módulo de software ejecutado por un procesador o en una combinación de los dos. Un módulo de software puede residir en memoria de acceso aleatorio (RAM), memoria flash, memoria de solo lectura (ROM), memoria de solo lectura programable (PROM), memoria de solo lectura programable y borrrable (EPROM), memoria de solo lectura programable y borrrable eléctricamente (EEPROM), registros, disco duro, disco extraíble, disco compacto con memoria de solo lectura (CD-ROM) o en cualquier otra forma de medio de almacenamiento conocido en la técnica. Un medio de almacenamiento a modo de ejemplo está acoplado al procesador de tal manera que el procesador puede leer información del medio de almacenamiento y escribir información en el mismo. De forma alternativa, el medio de almacenamiento puede estar integrado en el procesador. El procesador y el medio de almacenamiento pueden residir en un circuito integrado específico de la aplicación (ASIC). El ASIC puede residir en un dispositivo informático o en un terminal de usuario. Como alternativa, el procesador y el medio de almacenamiento pueden residir como componentes discretos en un dispositivo informático o en un terminal de usuario.

40 [85] La descripción anterior de los modos de realización divulgados se da a conocer para posibilitar que un experto en la técnica elabore o use los modos de realización divulgados. Varias modificaciones a estos modos de realización serán fácilmente evidentes para los expertos en la técnica. Por lo tanto, la presente divulgación no pretende limitarse a los modos de realización mostrados en el presente documento, sino que se le concede el alcance más amplio posible limitado únicamente por las reivindicaciones.

**REIVINDICACIONES**

1. Un procedimiento (500) para ejecutar, mediante un procesador, una instrucción de lectura vectorial para colocar uno o más valores de datos de elementos en un vector de salida, que comprende:
- 5 identificar (510) un vector de control de permutación vertical que comprende una pluralidad de elementos, comprendiendo cada elemento de la pluralidad de elementos una dirección de registro que indica un registro vectorial de una pluralidad de registros vectoriales del procesador;
- 10 para cada elemento de la pluralidad de elementos:
- leer (520) una dirección de registro del vector de control de permutación vertical; y
- 15 recuperar (530) un valor de datos de elementos del registro vectorial indicado por la dirección de registro, **caracterizado en que** una posición del valor de datos de elementos dentro del registro vectorial se indica mediante una posición de la dirección de registro dentro del vector de control de permutación vertical; y
- 20 además **caracterizado** por identificar (540) un vector de control de permutación horizontal que comprende un conjunto de direcciones correspondientes a un vector de salida; y
- colocar (550) al menos algunos de los valores de datos de elementos recuperados en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.
- 25 2. El procedimiento según la reivindicación 1, que comprende además:
- recibir datos asociados con una instrucción;
- 30 dividir los datos recibidos en la pluralidad de elementos; y
- colocar la pluralidad de elementos en el vector de control de permutación vertical.
3. El procedimiento según la reivindicación 1, en el que recuperar el valor de datos de elementos comprende usar una dirección de memoria de lectura en el vector de control de permutación vertical como una dirección indirecta para recuperar un valor de datos de elementos respectivo.
- 35 4. El procedimiento según la reivindicación 1, en el que colocar al menos algunos de los valores de datos de elementos recuperados en el vector de salida comprende usar una barra cruzada para colocar al menos algunos de los valores de datos de elementos recuperados en el vector de salida.
- 40 5. El procedimiento según la reivindicación 1, que comprende además:
- colocar los valores de datos de elementos recuperados en un vector temporal, en el que colocar al menos algunos de los valores de datos de elementos recuperados en el vector de salida comprende permutar el vector temporal usando el vector de control de permutación horizontal.
- 45 6. Un medio legible por ordenador que tiene almacenadas en el mismo instrucciones ejecutables por ordenador que, cuando se ejecutan mediante un procesador, hacen que el procesador lleve a cabo una instrucción de lectura vectorial para colocar uno o más valores de datos de elementos en un vector de salida, que comprende:
- 50 identificar un vector de control de permutación vertical que comprende una pluralidad de elementos, comprendiendo cada elemento de la pluralidad de elementos una dirección de registro que indica un registro vectorial de una pluralidad de registros vectoriales del procesador;
- 55 para cada elemento de la pluralidad de elementos:
- leer una dirección de registro del vector de control de permutación vertical; y
- 60 recuperar un valor de datos de elementos del registro vectorial indicado por la dirección de registro **caracterizado en que** una posición del valor de datos de elementos dentro del registro vectorial se indica mediante una posición de la dirección de registro dentro del vector de control de permutación vertical; y
- 65 además **caracterizado** por identificar un vector de control de permutación horizontal que comprende un conjunto de direcciones correspondientes a un vector de salida; y

colocar al menos algunos de los valores de datos de elementos recuperados en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.

5 7. Un procesador para ejecutar una instrucción de lectura vectorial para colocar uno o más valores de datos de elementos en un vector de salida, que comprende:

10 medios para identificar un vector de control de permutación vertical que comprende una pluralidad de elementos, comprendiendo cada elemento de la pluralidad de elementos una dirección de registro que indica un registro vectorial de una pluralidad de registros vectoriales del procesador;

medios para, para cada elemento de la pluralidad de elementos:

15 leer una dirección de registro del vector de control de permutación vertical; y

recuperar un valor de datos de elementos del registro vectorial indicado por la dirección de registro **caracterizado en que** una posición del valor de datos de elementos dentro del registro vectorial se indica mediante una posición de la dirección de registro dentro del vector de control de permutación vertical; y

20 además **caracterizado** por medios para identificar un vector de control de permutación horizontal que comprende un conjunto de direcciones correspondientes a un vector de salida; y

25 medios para colocar al menos algunos de los valores de datos de elementos recuperados en el vector de salida basándose en el conjunto de direcciones en el vector de control de permutación horizontal.

8. Un procedimiento (600) para ejecutar, mediante un procesador, una instrucción de escritura vectorial para escribir valores de datos de elementos en un vector de salida, que comprende:

30 leer (610) un vector de entrada que comprende una pluralidad de valores de datos de elementos;

identificar (620) un vector de control de permutación horizontal que comprende un conjunto de direcciones;

35 reorganizar (630) al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal;

40 colocar (640) la pluralidad reorganizada de valores de datos de elementos en un vector temporal;

identificar (650) un vector de control de permutación vertical que comprende una pluralidad de elementos, comprendiendo cada elemento de la pluralidad de elementos una dirección de registro que indica un registro vectorial de una pluralidad de registros vectoriales del procesador; y

45 colocar (660) la pluralidad de valores de datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical y las posiciones de las direcciones de registro dentro del vector de control de permutación vertical.

50 9. Un procedimiento según la reivindicación 1 o la reivindicación 8, en el que identificar el vector de control de permutación horizontal comprende identificar un registro vectorial en un archivo de registros vectoriales.

55 10. Un procedimiento según la reivindicación 1 o la reivindicación 8, en el que identificar el vector de control de permutación vertical comprende identificar un registro vectorial en un archivo de registros vectoriales.

11. Un procedimiento según la reivindicación 1 o la reivindicación 8, que comprende además:

60 recibir un valor escalar que hace referencia a una estructura de datos, en el que identificar el vector de control de permutación horizontal comprende identificar el vector de control de permutación horizontal en la estructura de datos usando el valor escalar.

12. El procedimiento según la reivindicación 1 o la reivindicación 8, que comprende además:

recibir un valor escalar que hace referencia a una estructura de datos, en el que identificar el vector de control de permutación vertical comprende identificar el vector de control de permutación vertical en la estructura de datos usando el valor escalar.

- 5 **13.** Un medio legible por ordenador que tiene almacenadas en el mismo instrucciones ejecutables por ordenador que, cuando se ejecutan mediante un procesador, hacen que el procesador lleve a cabo una instrucción de escritura vectorial para escribir valores de datos de elementos en un vector de salida, que comprende:
- 10 leer un vector de entrada que comprende una pluralidad de valores de datos de elementos;
- identificar un vector de control de permutación horizontal que comprende un conjunto de direcciones;
- 15 reorganizar al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal;
- colocar la pluralidad reorganizada de valores de datos de elementos en un vector temporal;
- 20 identificar un vector de control de permutación vertical que comprende una pluralidad de elementos, comprendiendo cada elemento de la pluralidad de elementos una dirección de registro que indica un registro vectorial de una pluralidad de registros vectoriales del procesador; y
- 25 colocar la pluralidad de valores de datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical y las posiciones de las direcciones de registro dentro del vector de control de permutación vertical.
- 14.** Un procesador para ejecutar una instrucción de escritura vectorial para escribir valores de datos de elementos en un vector de salida, que comprende:
- 30 medios para leer un vector de entrada que comprende una pluralidad de valores de datos de elementos;
- 35 medios para identificar un vector de control de permutación horizontal que comprende un conjunto de direcciones;
- medios para reorganizar al menos algunos de los valores de datos de elementos de la pluralidad de valores de datos de elementos en el vector de entrada basándose en el conjunto de direcciones en el vector de control de permutación horizontal;
- 40 medios para colocar la pluralidad reorganizada de valores de datos de elementos en un vector temporal;
- 45 medios para identificar un vector de control de permutación vertical que comprende una pluralidad de elementos, comprendiendo cada elemento de la pluralidad de elementos una dirección de registro que indica un registro vectorial de una pluralidad de registros vectoriales del procesador; y
- 50 medios para colocar la pluralidad de valores de datos de elementos en el vector temporal en al menos un registro vectorial basándose en las direcciones de registro en el vector de control de permutación vertical y las posiciones de las direcciones de registro dentro del vector de control de permutación vertical.

100  
↙

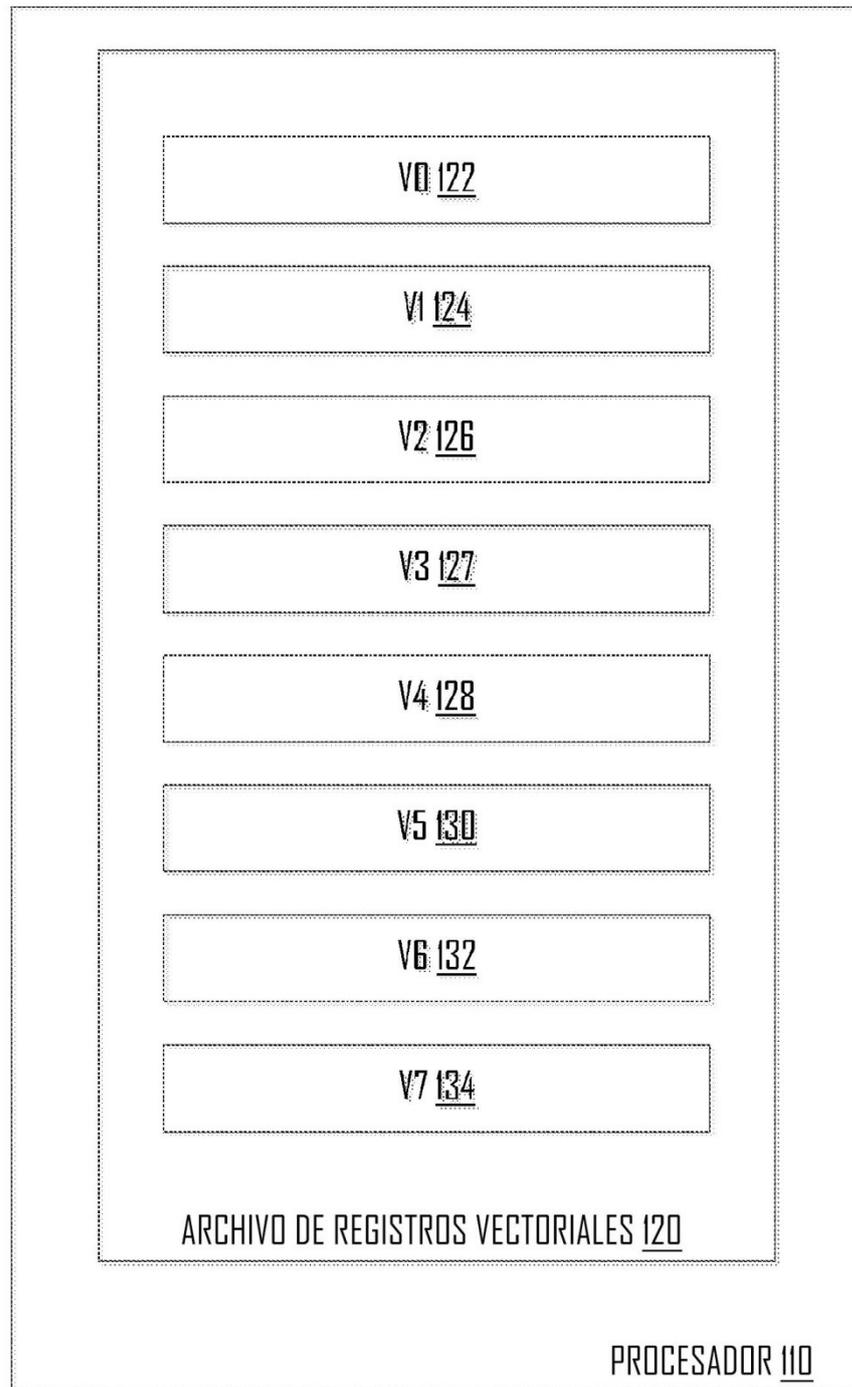


FIG. 1

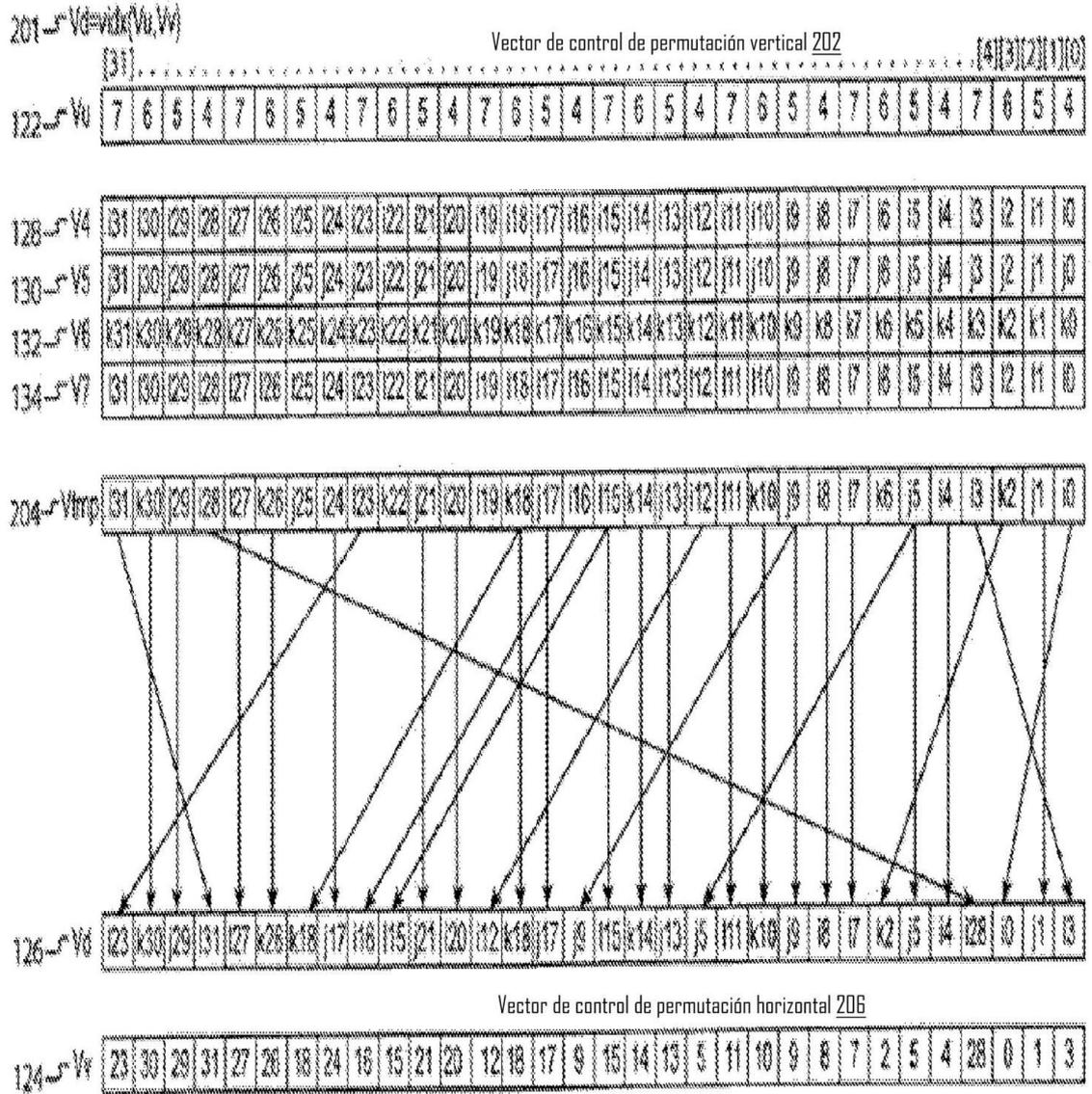
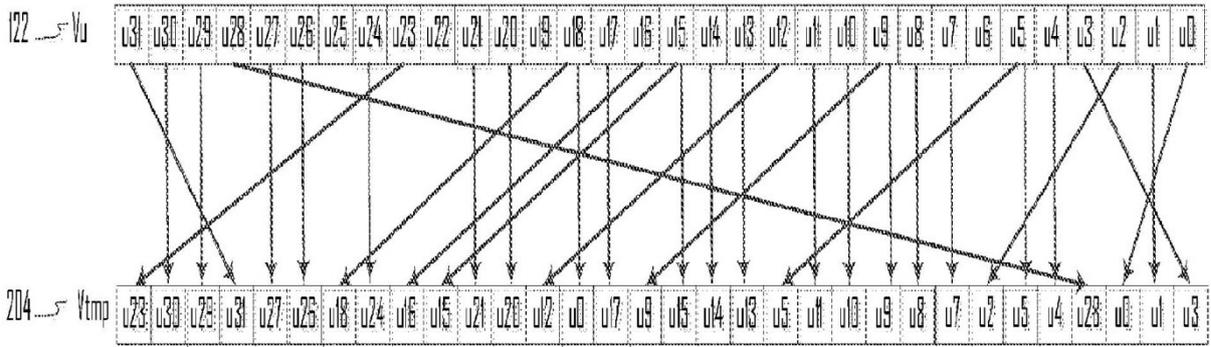


FIG. 2

300



301 Vidx(Vd, Vv=Vu)



Vector de control de permutación horizontal 206

124 Vv

23	30	29	31	27	26	18	24	16	15	21	20	12	18	17	9	15	14	13	5	11	10	9	8	7	2	5	4	28	0	1	3
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	----	----	----	---	----	----	---	---	---	---	---	---	----	---	---	---

Vector de control de permutación vertical 202

126 Vd

7	6	5	4	7	6	5	4	7	6	5	4	7	6	5	4	7	6	5	4	7	6	5	4	7	5	6	4	7	6	5	4
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

128	v4	u31	u30	u29	u31	u27	u26	u25	u24	u23	u22	u21	u20	u19	u18	u17	u9	u15	u14	u13	u5	u11	u10	u9	u8	u7	u6	u5	u4	u3	u2	u1	u3
130	v5	u31	u30	u29	u28	u27	u26	u18	u24	u23	u22	u21	u20	u19	u18	u17	u16	u15	u14	u13	u12	u11	u10	u9	u8	u7	u6	u5	u4	u3	u2	u1	u0
132	v6	u31	u30	u29	u28	u27	u26	u25	u24	u23	u15	u21	u20	u19	u18	u17	u16	u15	u14	u13	u12	u11	u10	u9	u8	u7	u2	u5	u4	u3	u0	u1	u0
134	v7	u23	u30	u29	u28	u27	u26	u25	u24	u16	u22	u21	u20	u12	u18	u17	u16	u15	u14	u13	u12	u11	u10	u9	u8	u7	u6	u5	u4	u28	u2	u1	u0

FIG. 3

400  
↙

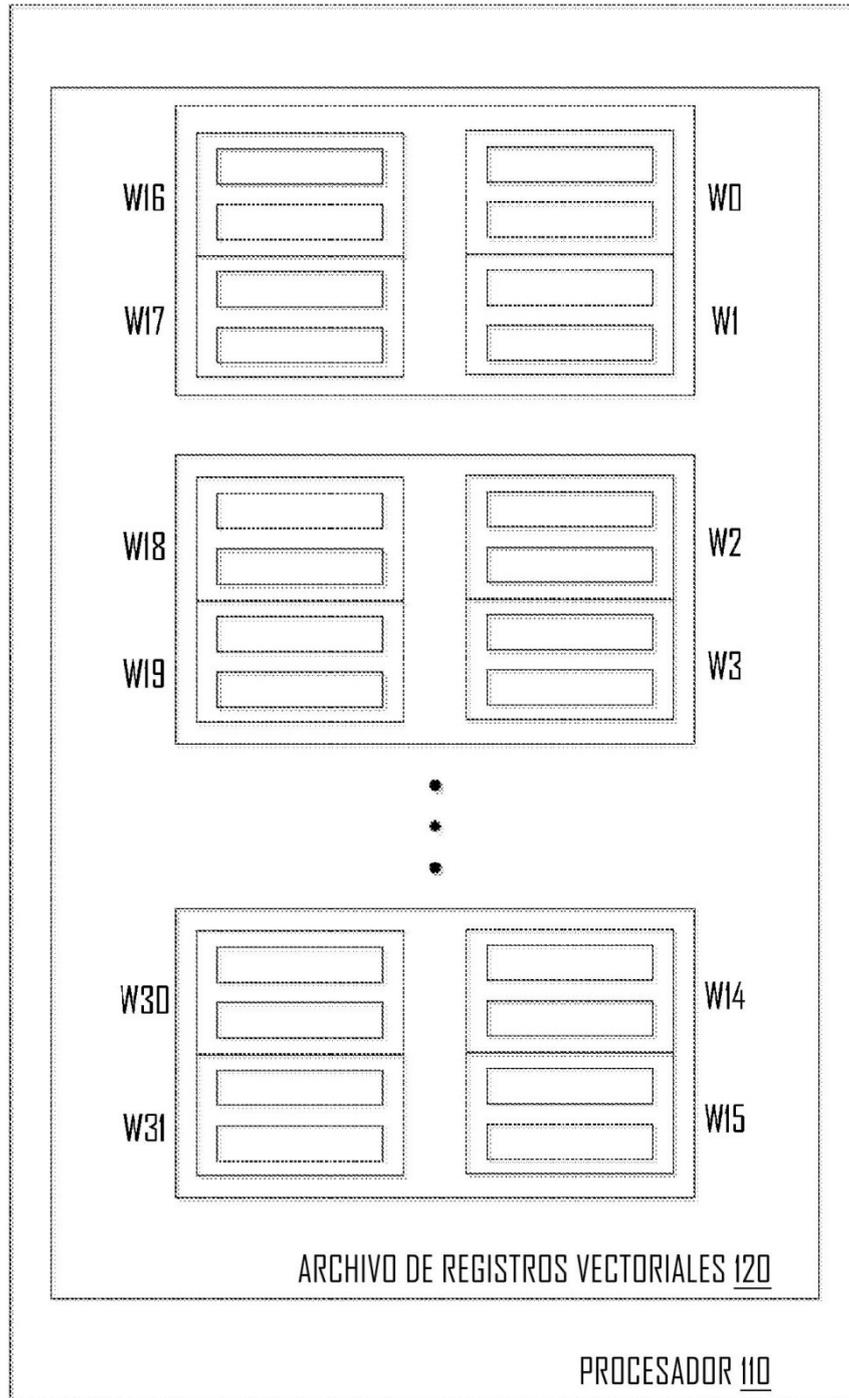


FIG. 4

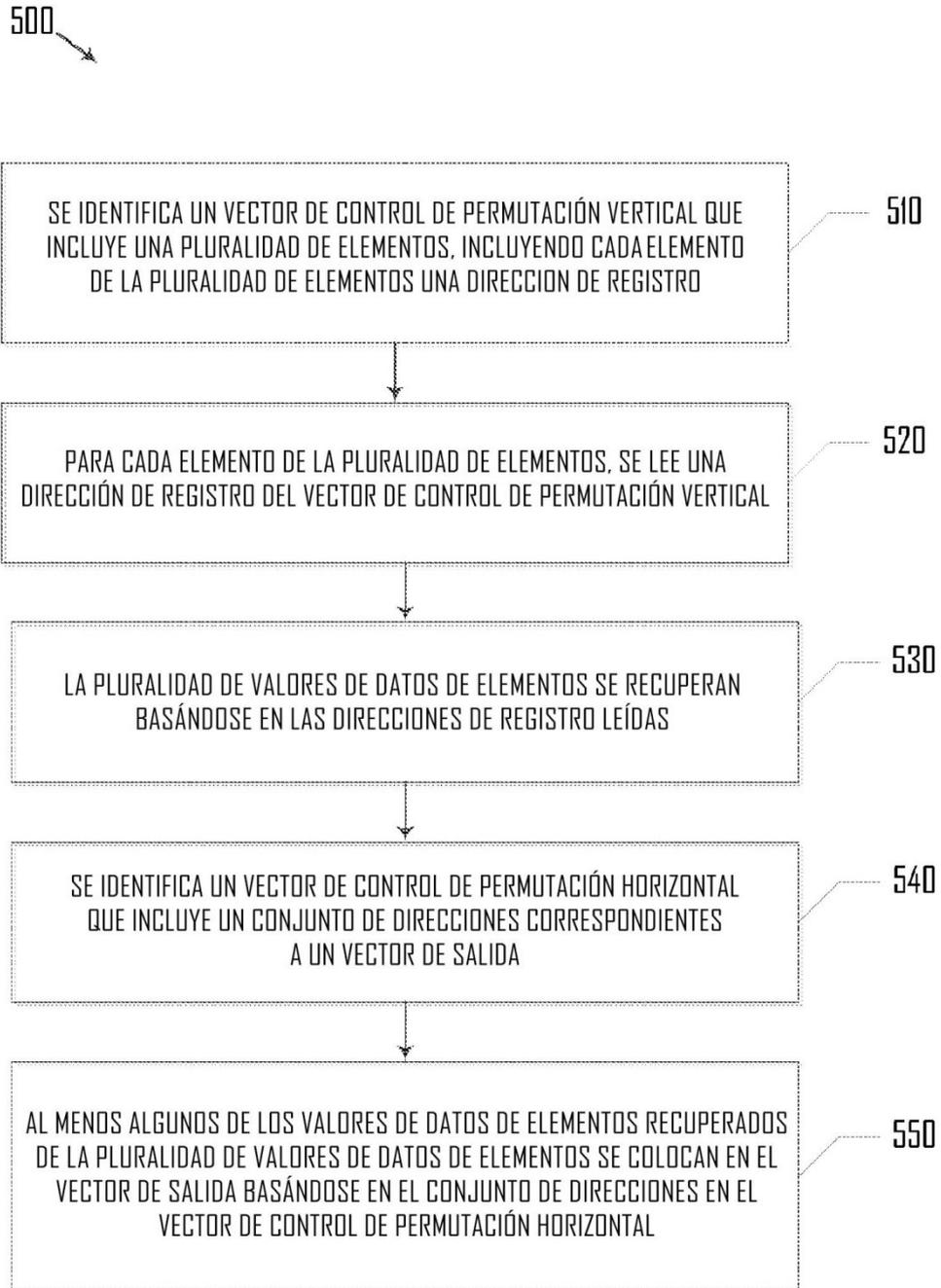


FIG. 5

600

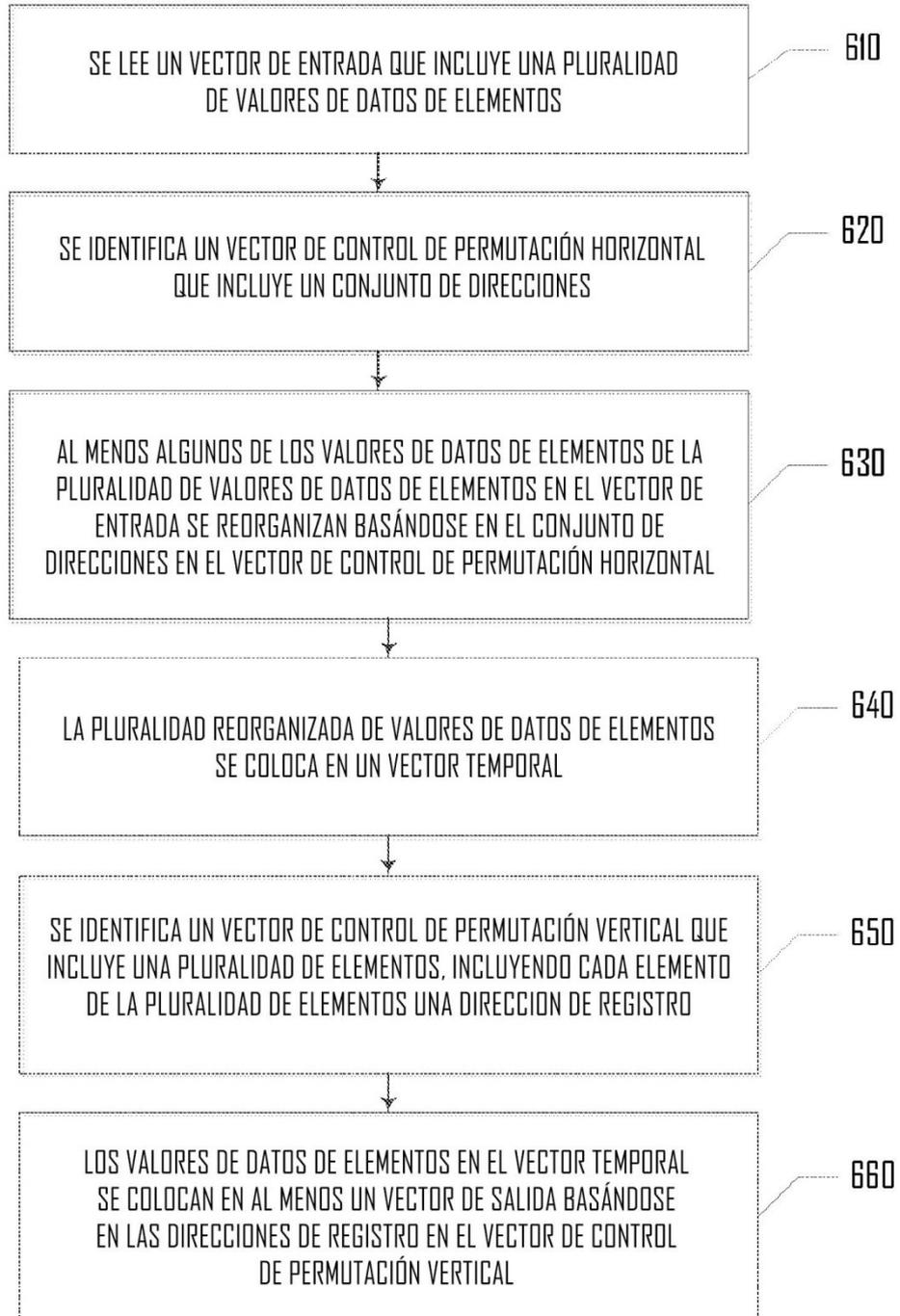


FIG. 6

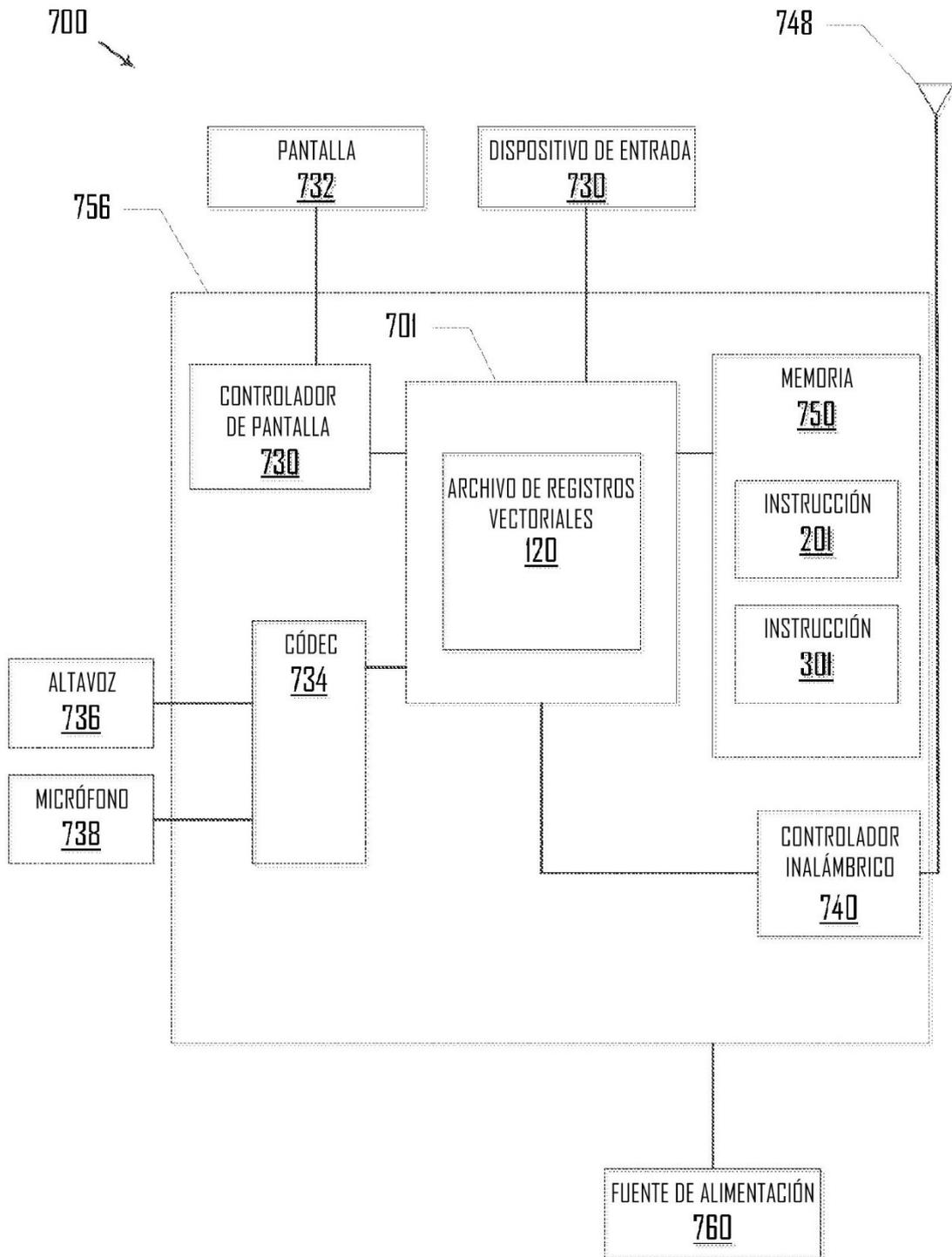


FIG. 7