

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 795 832**

51 Int. Cl.:

G06F 9/30

(2008.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **21.06.2016 PCT/US2016/038487**

87 Fecha y número de publicación internacional: **26.01.2017 WO17014892**

96 Fecha de presentación y número de la solicitud europea: **21.06.2016 E 16732213 (0)**

97 Fecha y número de publicación de la concesión europea: **25.03.2020 EP 3326060**

54 Título: **Operaciones SIMD de anchura mixta que tienen operaciones de elementos pares y de elementos impares usando un par de registros para elementos de datos anchos**

30 Prioridad:

21.07.2015 US 201514805456

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

24.11.2020

73 Titular/es:

**QUALCOMM INCORPORATED (100.0%)
5775 Morehouse Drive
San Diego, CA 92121-1714, US**

72 Inventor/es:

**MAHURIN, ERIC WAYNE y
INGLE, AJAY ANANT**

74 Agente/Representante:

FORTEA LAGUNA, Juan José

ES 2 795 832 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

Operaciones SIMD de anchura mixta que tienen operaciones de elementos pares y de elementos impares usando un par de registros para elementos de datos anchos

5

Campo de divulgación

[0001] Aspectos de esta divulgación se refieren a operaciones que implican dos o más vectores en los que elementos de datos de al menos un vector tienen una anchura de bits diferente que elementos de datos de al menos otro vector. Dichas operaciones se denominan operaciones de anchura mixta. Más específicamente, algunos aspectos se refieren a operaciones de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta que involucran al menos un primer operando vectorial y un segundo operando vectorial, donde al menos uno de los primero o segundo operandos vectoriales tienen elementos de datos que pueden almacenarse en pares de registros par e impar.

10

Antecedentes

[0002] Las instrucciones de tipo "una instrucción, múltiples datos" (SIMD) se pueden usar en sistemas de procesamiento para aprovechar el paralelismo de datos. El paralelismo de datos se produce cuando se necesita realizar una misma tarea, o una tarea común, en dos o más elementos de datos de un vector de datos, por ejemplo. En lugar de usar múltiples instrucciones, la tarea común se puede realizar en los dos o más elementos de datos en paralelo usando una sola instrucción SIMD que define la misma instrucción que se realizará en múltiples elementos de datos en los múltiples carriles SIMD correspondientes.

20

[0003] Las instrucciones SIMD pueden incluir uno o más operandos vectoriales, tales como los operandos vectoriales fuente y destino. Cada operando vectorial incluirá dos o más elementos de datos. En cuanto a las instrucciones SIMD, todos los elementos de datos que pertenecen al mismo operando vectorial pueden tener, en general, la misma anchura de bits. Sin embargo, algunas instrucciones SIMD pueden especificar operandos de anchura mixta, donde los elementos de datos de un primer operando vectorial pueden tener una primera anchura de bits y los elementos de datos de un segundo operando vectorial pueden tener una segunda anchura de bits, donde las primera y segunda anchuras de bits difieren entre sí. La ejecución de instrucciones SIMD con operandos de anchura mixta puede suponer varios desafíos.

25

30

[0004] Las FIGS. 1A-C ilustran ejemplos de desafíos relativos a implementaciones convencionales para ejecutar instrucciones SIMD con operandos de anchura mixta. Con referencia a la FIG. 1A, se ilustra una primera implementación convencional para ejecutar una instrucción SIMD 100. Se supone que la instrucción SIMD 100 puede ser ejecutada por un procesador convencional (no mostrado) que admita una arquitectura de conjunto de instrucciones (ISA) de 64 bits. Esto significa que instrucciones tales como la instrucción SIMD 100 pueden especificar operandos con anchuras de bits de hasta 64 bits. Los operandos de 64 bits pueden especificarse en términos de registros de 64 bits o un par de registros de 32 bits.

35

40

[0005] El objetivo de la instrucción SIMD 100 es ejecutar la misma instrucción en cada elemento de datos del operando fuente 102. El operando fuente 102 es un vector de 64 bits que comprende ocho elementos de datos de 8 bits etiquetados de 0 a 7. El operando fuente 102 puede almacenarse en un único registro de 64 bits o en un par de registros de 32 bits. La misma instrucción u operación común que se ejecutará en cada uno de los ocho elementos de datos 0-7 puede ser, por ejemplo, multiplicación, función de elevación al cuadrado, función de desplazamiento a la izquierda, función de incremento, suma (por ejemplo, con un valor constante o campos inmediatos en la instrucción o con valores proporcionados por otro operando vectorial), etc., cuyo resultado puede consumir más de 8 bits y hasta 16 bits de almacenamiento para cada uno de los ocho elementos de datos resultantes. Esto significa que el resultado de la instrucción SIMD 100 puede consumir el doble de espacio de almacenamiento que el operando fuente 102 puede consumir, es decir, dos registros de 64 bits o dos pares de registros de 32 bits.

45

50

[0006] Dado que el procesador convencional configurado para implementar la instrucción SIMD 100 no incluye instrucciones que especifiquen operandos de anchuras de bits superiores a 64 bits, la instrucción SIMD 100 puede dividirse en dos instrucciones SIMD componentes 100X y 100Y. La instrucción SIMD 100X especifica la operación común que se realizará en elementos de datos etiquetados con números pares (o "elementos de datos de numeración par") 0, 2, 4 y 6 del operando fuente 102. La instrucción SIMD 100X especifica el operando destino 104x que tiene una anchura de 64 bits e incluye elementos de datos de 16 bits etiquetados como A, C, E y G, cada uno de los cuales está compuesto por 8 bits altos (H) y 8 bits bajos (L). Los resultados de la operación común en elementos de datos de 8 bits de numeración par 0, 2, 4 y 6 del operando fuente 102 se escriben de manera correspondiente en elementos de datos de 16 bits A, C, E y G del operando destino 104x. La instrucción SIMD 100Y es similar a la instrucción SIMD 100X con la diferencia de que la instrucción SIMD 100Y especifica la operación común en elementos de datos etiquetados con números impares (o "elementos de datos de numeración impar") 1, 3, 5 y 7 del operando fuente 102, donde los resultados se escribirán en elementos de datos de 16 bits B, D, F, H del operando destino 104y, que también es un operando de 64 bits similar al operando destino 104x de la instrucción SIMD 100X. De esta manera, cada una de las instrucciones SIMD 100X y 100Y puede especificar un operando destino de 64 bits, y juntas, las instrucciones SIMD 100X y 100Y pueden realizar la ejecución de la operación común en cada uno de los elementos de datos 0-7

55

60

65

del operando fuente 102. Sin embargo, debido a las dos instrucciones individuales necesarias para implementar la instrucción SIMD 100, aumenta el espacio de código.

[0007] La FIG. 1B ilustra una segunda implementación convencional de la instrucción SIMD 100 usando un conjunto diferente de instrucciones SIMD componentes 120X y 120Y. Las instrucciones SIMD 120X y 120Y especifican la operación común en cada uno de los elementos de datos de 8 bits 0-7 del operando fuente 102. La instrucción SIMD 120X especifica el operando destino 124x en el que se escribirán los 8 bits bajos (L) de los resultados en relación con elementos de datos de resultado de 8 bits A-H correspondientes del operando destino 124x (mientras que los 8 bits altos (H) de los resultados se descartan). De manera similar, la instrucción SIMD 120Y especifica el operando destino 124y en el que se escribirán los 8 bits altos (H) de los resultados en relación con elementos de datos de 8 bits A-H correspondientes del operando destino 124y (mientras que los 8 bits bajos (L) de los resultados se descartan). Esta segunda implementación convencional de la instrucción SIMD 100 también adolece de un mayor espacio de código para las dos instrucciones SIMD componentes 120X y 120Y. Además, como se puede apreciar, la segunda implementación convencional también incurre en un malgasto de energía al calcular y descartar los 8 bits altos (H) (por ejemplo, en la ejecución de la instrucción 120X) o los 8 bits bajos (L) (por ejemplo, en la ejecución de la instrucción 120Y) para cada uno de los elementos de datos 0-7 del operando fuente 102.

[0008] La FIG. 1C ilustra una tercera implementación convencional de la instrucción SIMD 100 usando otro conjunto de instrucciones SIMD componentes 140X y 140Y, que son similares en algunos aspectos a las instrucciones SIMD 100X y 100Y de la FIG. 1A. La diferencia radica en cuáles de los elementos de datos del operando fuente 102 opera cada instrucción SIMD. Con más detalle, en lugar de elementos de datos de 8 bits de numeración par, la instrucción SIMD 140X especifica la operación común que se realizará en los cuatro elementos de datos más bajos 0-3 del operando fuente 102. Los resultados se escriben en elementos de datos de 16 bits A, B, C, D del operando destino 144x. Sin embargo, la ejecución de la instrucción SIMD 140X implica el esparcimiento de los resultados de la operación realizada en los cuatro elementos de datos de 8 bits más bajos (que abarcan 32 bits) a través de la totalidad de 64 bits del operando destino 140X. La instrucción SIMD 144y es similar y especifica el esparcimiento de los resultados de la operación realizada en los cuatro elementos de datos de 8 bits más altos 4-7 del operando fuente 102 a través de los elementos de datos de 16 bits E, F, G, H del operando destino de 64 bits 144y. Además del aumento del tamaño del código como en las primera y segunda implementaciones convencionales, estos movimientos de esparcimiento de datos como se observa en la tercera implementación convencional pueden necesitar hardware adicional, tal como una barra cruzada.

[0009] En consecuencia, existe la necesidad de implementaciones mejoradas de instrucciones SIMD de anchura mixta que eviten los inconvenientes mencionados anteriormente de las implementaciones convencionales. El documento US 2005/125631 (Symes et al.), 9 de junio de 2005, divulga un procesador SIMD que tiene instrucciones vectoriales para implementar operaciones de anchura mixta. El documento US 7.219.212 (Sanghavi et al.), 15 de mayo de 2007, divulga una instrucción de multiplicación que se ejecuta de manera iterativa, en la que en el primer ciclo se multiplican cuatro elementos de datos pares, y en el siguiente ciclo se multiplican cuatro elementos impares.

BREVE EXPLICACIÓN

[0010] Aspectos ejemplares incluyen sistemas y procedimientos relacionados con una instrucción de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta que tiene al menos un operando vectorial fuente que comprende elementos de datos de una primera anchura de bits y un operando vectorial destino que comprende elementos de datos de una segunda anchura de bits, donde la segunda anchura de bits es la mitad o el doble de la primera anchura de bits. En consecuencia, uno de los operandos vectoriales fuente o destino se expresa como un par de registros, un primer registro y un segundo registro. El otro operando vectorial se expresa como un único registro. Los elementos de datos del primer registro corresponden a elementos de datos de numeración par del otro operando vectorial expresado como un único registro, y los elementos de datos del segundo registro corresponden a elementos de datos del otro operando vectorial expresado como un único registro.

[0011] Por ejemplo, un aspecto ejemplar se refiere a un procedimiento para realizar una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta, donde el procedimiento comprende: recibir, por un procesador, una instrucción SIMD que comprende al menos un primer operando vectorial fuente que comprende un primer conjunto de elementos de datos fuente de una primera anchura de bits, y al menos un operando vectorial destino que comprende elementos de datos destino de una segunda anchura de bits, donde la segunda anchura de bits es dos veces la primera anchura de bits. El operando vectorial destino comprende un par de registros, que incluyen un primer registro que comprende un primer subconjunto de los elementos de datos destino y un segundo registro que comprende un segundo subconjunto de los elementos de datos destino. En base a un orden secuencial del primer conjunto de elementos de datos fuente, el procedimiento incluye ejecutar la instrucción SIMD en el procesador, que comprende generar el primer subconjunto de elementos de datos destino en el primer registro a partir de elementos de datos fuente de numeración par del primer conjunto, y generar el segundo subconjunto de elementos de datos destino en el segundo registro a partir de elementos de datos fuente de numeración impar del primer conjunto.

[0012] Otro aspecto ejemplar se refiere a un procedimiento para realizar una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta, donde el procedimiento comprende: recibir, por un procesador, una

instrucción SIMD que comprende al menos un operando vectorial fuente que comprende elementos de datos fuente de una primera anchura de bits, y al menos un operando vectorial destino que comprende elementos de datos destino de una segunda anchura de bits, donde la segunda anchura de bits es la mitad de la primera anchura de bits. El operando vectorial fuente comprende un par de registros, que incluyen un primer registro que comprende un primer subconjunto de los elementos de datos fuente y un segundo registro que comprende un segundo subconjunto de los elementos de datos fuente. En base a un orden secuencial de los elementos de datos destino, el procedimiento incluye ejecutar la instrucción SIMD en el procesador, que comprende generar elementos de datos destino de numeración par a partir del primer subconjunto correspondiente de elementos de datos fuente en el primer registro, y generar elementos de datos destino de numeración impar a partir del segundo subconjunto correspondiente de elementos de datos fuente en el segundo registro.

[0013] Otro aspecto ejemplar se refiere a un medio de almacenamiento no transitorio legible por ordenador que comprende instrucciones ejecutables por un procesador, que cuando se ejecuta por el procesador hace que el procesador realice una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta. El medio de almacenamiento no transitorio legible por ordenador, comprende una instrucción SIMD, que comprende al menos un primer operando vectorial fuente que comprende un primer conjunto de elementos de datos fuente de una primera anchura de bits, y al menos un operando vectorial destino que comprende elementos de datos destino de una segunda anchura de bits, donde la segunda anchura de bits es dos veces la primera anchura de bits. El operando vectorial destino comprende un par de registros, que incluyen un primer registro que comprende un primer subconjunto de los elementos de datos destino y un segundo registro que comprende un segundo subconjunto de los elementos de datos destino. En base a un orden secuencial del primer conjunto de elementos de datos fuente, el medio de almacenamiento no transitorio legible por ordenador incluye código para generar el primer subconjunto de elementos de datos destino en el primer registro a partir de elementos de datos fuente de numeración par del primer conjunto, y código para generar el segundo subconjunto de elementos de datos destino en el segundo registro a partir de elementos de datos fuente de numeración impar del primer conjunto.

[0014] Otro aspecto ejemplar adición se refiere a un medio de almacenamiento no transitorio legible por ordenador que comprende instrucciones ejecutables por un procesador que, cuando se ejecutan por el procesador, hacen que el procesador realice una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta, donde el medio de almacenamiento no transitorio legible por ordenador comprende una instrucción SIMD. La instrucción SIMD comprende al menos un operando vectorial fuente que comprende elementos de datos fuente de una primera anchura de bits, y al menos un operando vectorial destino que comprende elementos de datos destino de una segunda anchura de bits, donde la segunda anchura de bits es la mitad de la primera anchura de bits. El operando vectorial fuente comprende un par de registros, que incluyen un primer registro que comprende un primer subconjunto de los elementos de datos fuente y un segundo registro que comprende un segundo subconjunto de los elementos de datos fuente. En base a un orden secuencial de los elementos de datos destino, el medio de almacenamiento no transitorio legible por ordenador incluye código para generar elementos de datos destino de numeración par a partir del primer subconjunto correspondiente de elementos de datos fuente en el primer registro, y código para generar elementos de datos destino de numeración impar a partir del segundo subconjunto correspondiente de elementos de datos fuente en el segundo registro.

BREVE DESCRIPCIÓN DE LOS DIBUJOS

[0015] Los dibujos adjuntos se presentan para facilitar la descripción de los aspectos de la divulgación y se proporcionan únicamente para ilustrar los aspectos y no para limitar los mismos.

Las FIGS. 1A-C ilustran implementaciones convencionales de instrucciones SIMD de anchura mixta.

Las FIGS. 2A-C ilustran implementaciones ejemplares de instrucciones SIMD de anchura mixta de acuerdo con aspectos de esta divulgación.

Las FIGS. 3A-B ilustran procedimientos para realizar operaciones de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta.

La FIG. 4 ilustra un dispositivo inalámbrico 400 ejemplar en el que se puede emplear de forma ventajosa un aspecto de la divulgación.

DESCRIPCIÓN DETALLADA

[0016] Se divulgan aspectos de la invención en la siguiente descripción y en dibujos relacionados relativos a aspectos específicos de la invención. Pueden concebirse aspectos alternativos sin apartarse del alcance de la invención. Además, no se describirán con detalle elementos bien conocidos de la invención, o se omitirán para no complicar los detalles pertinentes de la invención.

[0017] El término "ejemplar" se usa en el presente documento para significar "que sirve de ejemplo, caso o ilustración". Cualquier aspecto descrito en el presente documento como "ejemplar" no ha de interpretarse

necesariamente como preferente o ventajoso con respecto a otros aspectos. Asimismo, el término "aspectos de la invención" no requiere que todos los aspectos de la invención incluyan la característica, ventaja o modo de funcionamiento analizados.

5 **[0018]** La terminología usada en el presente documento solo tiene el propósito de describir aspectos particulares y no está concebida para limitar los aspectos de la invención. Como se usa en el presente documento, las formas en singular "un", "una", "el" y "la" están concebidas para incluir también las formas en plural, a menos que el contexto indique claramente lo contrario. Se entenderá además que los términos "comprende", "comprendiendo", "incluye" y/o "incluyendo", cuando se usan en el presente documento, especifican la presencia de características, números enteros, etapas, operaciones, elementos y/o componentes indicados, pero no excluyen la presencia o adición de otra u otras características, números enteros, etapas, operaciones, elementos, componentes y/o grupos de los mismos.

15 **[0019]** Además, muchos aspectos se describen en términos de secuencias de acciones que se van a realizar, por ejemplo, mediante elementos de un dispositivo informático. Se reconocerá que diversas acciones descritas en el presente documento se pueden realizar mediante circuitos específicos (por ejemplo, circuitos integrados específicos de la aplicación (ASIC)), mediante instrucciones de programa ejecutadas por uno o más procesadores o mediante una combinación de ambas cosas. Además, se puede considerar que estas secuencias de acciones descritas en el presente documento se incorporan por completo dentro de cualquier forma de medio de almacenamiento legible por ordenador que tenga almacenado en el mismo un conjunto correspondiente de instrucciones informáticas que, al ejecutarse, harán que un procesador asociado realice la funcionalidad descrita en el presente documento. Por tanto, los diversos aspectos de la invención pueden realizarse de varias formas diferentes, todas ellas contempladas dentro del alcance de la materia objeto reivindicada. Además, en lo que respecta a cada uno de los aspectos descritos en el presente documento, la forma correspondiente de cualquiera de dichos aspectos se puede describir en el presente documento como, por ejemplo, "lógica configurada para" realizar la acción descrita.

25 **[0020]** Aspectos ejemplares de esta divulgación se refieren a la implementación de operaciones SIMD de anchura mixta que evitan el movimiento de datos a través de carriles SIMD y reducen el tamaño del código. Por ejemplo, en lugar de descomponer una operación SIMD en dos o más instrucciones SIMD componentes (por ejemplo, ejecución convencional de instrucción SIMD 100 en las FIGS. 1A-C), aspectos ejemplares incluyen una sola instrucción SIMD que especifica uno o más operandos vectoriales como un par de operandos, que pueden expresarse en términos de un par de registros. Al especificar al menos un operando vectorial (ya sea un operando fuente o destino) como un par de registros, se puede usar la única instrucción SIMD ejemplar en lugar de dos o más instrucciones SIMD convencionales componentes. Por lo tanto, el tamaño de código se reduce para operaciones SIMD de anchura mixta.

35 **[0021]** Cabe señalar que en esta divulgación se hace referencia a expresar operandos en términos de registros a fin de seguir los formatos de instrucción habituales en los que una instrucción especifica una operación a realizar en uno o más registros. Por lo tanto, una instrucción SIMD puede tener un formato en el que se especifica una operación común para uno o más operandos que se expresan en términos de registros. Por lo tanto, una instrucción SIMD de anchura mixta ejemplar de acuerdo con esta divulgación incluye al menos un operando vectorial expresado en términos de un único registro y al menos otro operando vectorial expresado en términos de un par de registros. Estas referencias a registros pueden pertenecer a registros lógicos o arquitectónicos usados por un programa que comprende instrucciones SIMD ejemplares. También pueden pertenecer a registros físicos de un archivo de registro físico, sin restricción. En general, las referencias a registros se refieren a elementos de almacenamiento de determinado tamaño.

45 **[0022]** Por consiguiente, un procedimiento ejemplar para ejecutar una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta en un procesador acoplado a un archivo de registro puede implicar la especificación de una instrucción SIMD con al menos un primer operando vectorial que comprende elementos de datos de una primera anchura de bits y al menos un segundo operando vectorial que comprende elementos de datos de una segunda anchura de bits. El primer operando vectorial puede ser un operando vectorial fuente y el segundo operando vectorial puede ser un operando vectorial destino. De manera correspondiente, los elementos de datos del operando vectorial fuente pueden denominarse elementos de datos fuente y los elementos de datos del operando vectorial destino pueden denominarse elementos de datos destino.

55 **[0023]** Existe una correspondencia uno a uno entre los elementos de datos fuente y los elementos de datos destino en una instrucción SIMD de anchura mixta ejemplar. En general, cuando la operación especificada en la instrucción SIMD de anchura mixta se realiza en un elemento de datos fuente, se genera un elemento de datos destino correspondiente específico. Por ejemplo, considérese una operación SIMD de anchura mixta para desplazamiento a la izquierda del operando vectorial fuente para formar un operando vectorial destino. En este ejemplo, cada elemento de datos fuente genera un elemento de datos destino específico cuando se realiza un desplazamiento a la izquierda del elemento de datos fuente.

65 **[0024]** En un aspecto ejemplar de esta divulgación, la segunda anchura de bits de los elementos de datos destino puede ser menor que y, específicamente, la mitad del tamaño de la primera anchura de bits de los elementos de datos fuente. En este aspecto, el operando vectorial destino puede expresarse como un par de registros y el operando vectorial fuente puede expresarse como un único registro.

[0025] En otro aspecto ejemplar de esta divulgación, la segunda anchura de bits de los elementos de datos destino puede ser mayor que y, específicamente, el doble del tamaño de la primera anchura de bits de los elementos de datos fuente. En este aspecto, el operando vectorial fuente puede expresarse como un único registro y el operando vectorial destino puede expresarse como un par de registros.

[0026] Para ilustrar la correlación específica entre elementos de datos fuente y vector de los operandos vectoriales fuente y destino, respectivamente, se asigna un orden secuencial a los elementos de datos del operando vectorial cuyos elementos de datos tienen un anchura de bits menor. Por ejemplo, se asigna un orden secuencial a los elementos de datos del operando vectorial que se expresa como un único registro. En base al orden secuencial, elementos de datos de numeración par (por ejemplo, correspondientes a los números 0, 2, 4, 6, etc.) y elementos de datos de numeración impar (por ejemplo, correspondientes a los números 1, 3, 5, 7, etc.) se identifican para el operando vectorial expresado como un único registro. El par de registros del otro operando vectorial se denominan primer registro y segundo registro, que comprenden un primer subconjunto y un segundo subconjunto de elementos de datos, respectivamente. En consecuencia, a los elementos de datos de numeración par del operando vectorial expresado como un solo registro se les asigna una correspondencia con elementos de datos del primer subconjunto o primer registro, y a los elementos de datos de numeración impar se les asigna una correspondencia con elementos de datos del segundo subconjunto o segundo registro. De esta manera, se evitan grandes movimientos de datos a través de carriles SIMD para elementos de datos fuente durante la ejecución de la operación SIMD especificada para generar elementos de datos destino correspondientes.

[0027] Los aspectos ejemplares también pueden relacionarse con operaciones SIMD que especifican más de dos operandos vectoriales, tales como, incluir un tercer operando de una tercera anchura de bits, y superior. Se describe un ejemplo en el que dos operandos vectoriales fuente, cada uno expresado como un único registro, se especifican para una instrucción SIMD de anchura mixta para generar un operando vectorial destino expresado como un par de registros. Otros numerosos formatos de instrucciones de este tipo son posibles dentro del alcance de esta divulgación. En aras de la simplicidad, se analizarán aspectos ejemplares para implementar operaciones SIMD de anchura mixta en relación con algunas instrucciones SIMD de ejemplo y anchuras de bits de operandos, teniendo en cuenta que esto tiene fines meramente explicativos. Como tal, las características analizadas en el presente documento pueden extenderse a cualquier número de operandos y anchuras de bits de elementos de datos para operaciones vectoriales de anchura mixta.

[0028] En las FIGS. 2A-C, se muestran aspectos ejemplares relacionados con las instrucciones SIMD 200, 220 y 240. Cada una de estas instrucciones SIMD 200, 220 y 240 pueden ejecutarse mediante un procesador (por ejemplo, el procesador 402 mostrado en las FIGS 4-5) configurado para ejecutar instrucciones SIMD. Más específicamente, cada una de estas instrucciones SIMD 200, 220 y 240 puede especificar uno o más operandos vectoriales fuente y uno o más operandos vectoriales destino, donde los operandos vectoriales fuente y destino pueden expresarse en términos de registros (por ejemplo, registros de 64 bits). Los operandos vectoriales fuente y destino de las instrucciones SIMD 200, 220 y 240 incluyen elementos de datos fuente y destino correspondientes, cada uno de los cuales se encuentra en uno o más carriles SIMD. El número de carriles SIMD en la ejecución de una instrucción SIMD corresponde al número de operaciones paralelas que se realizan en la ejecución de la instrucción SIMD. Un procesador o lógica de ejecución configurados para implementar las instrucciones SIMD 200, 220 y 240 de ejemplo, puede incluir en consecuencia hardware (por ejemplo, una unidad aritmética y lógica (ALU) que comprende una pluralidad de desplazadores hacia la izquierda/derecha, sumadores, multiplicadores, etc.) requerida para implementar las operaciones paralelas especificadas por las instrucciones SIMD 200, 220 y 240 .

[0029] Por consiguiente, con referencia a la FIG. 2A, se ilustra un primer aspecto ejemplar para la ejecución de la instrucción SIMD 200. En un ejemplo, se supone que el procesador es capaz de admitir una arquitectura de conjunto de instrucciones (ISA) de 64 bits. La instrucción SIMD 200 puede especificar la misma operación o instrucción común a realizar en elementos de datos fuente de operandos vectoriales fuente expresados en términos de un único registro de 64 bits.

[0030] La misma operación o instrucción común especificada en la instrucción SIMD 200 puede ser, por ejemplo, una función de elevación al cuadrado, una función de desplazamiento a la izquierda, una función de incremento, una suma con un valor constante, etc., en ocho elementos de datos fuente de 8 bits (lo que se puede implementar con elementos lógicos tales como ocho desplazadores a la izquierda de 8 bits, ocho sumadores de 8 bits, etc.) que produzca ocho elementos de datos destino resultantes correspondientes que puedan consumir hasta 16 bits de almacenamiento. Como se muestra, la instrucción SIMD 200 puede especificar un operando vectorial fuente 202 que comprende ocho elementos de datos de 8 bits. Se puede asignar un orden numérico a estos ocho elementos de datos de 8 bits del operando vectorial fuente 202, que se muestra mediante los números de referencia 0-7. El resultado de la instrucción SIMD 200 puede expresarse usando ocho elementos de datos destino de 16 bits o 128 bits en total, que no pueden almacenarse en un solo registro de 64 bits. En lugar de descomponer la instrucción SIMD 200 en dos o más instrucciones para manejar este problema (por ejemplo, como en las implementaciones convencionales de la instrucción SIMD 100 que se muestran en las FIGS. 1A-C), un operando vectorial destino se especifica como un par de operandos vectoriales componentes. El par de operandos vectoriales destino componentes se pueden expresar como un par correspondiente de registros 204x, 204y. Debe observarse que el par de registros no necesita

almacenarse en ubicaciones físicas consecutivas en un archivo de registro o incluso tener números de registro lógicos consecutivos. Como tal, la instrucción SIMD 200 especifica el operando vectorial destino expresado en términos de un par de operandos o registros vectoriales componentes 204x, 204y (por ejemplo, un par de registros de 64 bits) y el operando vectorial fuente 202, que se expresa como un único registro 202.

[0031] Además, el primer operando vectorial destino componente expresado como primer registro 204x del par incluye un primer subconjunto de los resultados de la instrucción SIMD 200 realizada en elementos de datos fuente de numeración par 0, 2, 4 y 6 del operando vectorial fuente 202. Estos resultados se ilustran con los elementos de datos destino A, C, E y G, que tienen una correspondencia uno a uno con elementos de datos fuente de numeración par 0, 2, 4 y 6, lo que significa que se evitan grandes movimientos a través de carriles SIMD para los resultados en esta disposición ejemplar de elementos de datos destino A, C, E y G. De manera similar, el segundo operando vectorial destino componente expresado como un segundo registro 204y del par incluye un segundo subconjunto de los resultados de la instrucción SIMD 200 realizada en elementos de datos fuente de numeración impar 1, 3, 5 y 7 del operando vectorial fuente 202. Estos resultados se ilustran con los elementos de datos destino B, D, F y H, que tienen una correspondencia uno a uno con elementos de datos fuente de numeración impar 1, 3, 5 y 7, lo que significa que, de nuevo, se evitan grandes movimientos en los carriles SIMD para los resultados en esta disposición ejemplar de elementos de datos destino B, D, F y H. En consecuencia, en este caso, los elementos de datos fuente de numeración par 0, 2, 4 y 6 del operando vectorial fuente 202 corresponden a o generan elementos de datos destino A, C, E y G del primer registro 204x; y los elementos de datos fuente de numeración impar 1, 3, 5 y 7 del operando vectorial fuente 202 corresponden a o generan elementos de datos destino B, D, F y H del segundo registro 204y.

[0032] Considerando ocho carriles SIMD de 8 bits, por ejemplo, denominados carriles SIMD 0-7, donde cada carril comprende un elemento de datos fuente respectivo 0-7, se observa que la cantidad de movimiento involucrado para generar un elemento de datos destino correspondiente A-H está contenida dentro del mismo carril SIMD o un carril SIMD adyacente. En otras palabras, un primer conjunto de elementos de datos fuente (por ejemplo, elemento de datos fuente 0-7) se encuentra en los carriles SIMD respectivos y, a partir de cada uno de los elementos de datos fuente, un elemento de datos destino (por ejemplo, un elemento de datos destino correspondiente A-H) se genera en el carril SIMD respectivo o en un carril SIMD adyacente al carril SIMD respectivo. Por ejemplo, los elementos de datos fuente de numeración par 0, 2, 4 y 6 en los carriles SIMD 0, 2, 4 y 6, respectivamente, generan elementos de datos destino A, C, E y G, que están contenidos, respectivamente, en carriles SIMD 0-1, 2-3, 4-5 y 6-7. Del mismo modo, elementos de datos fuente de numeración impar 1, 3, 5 y 7 en los carriles SIMD 0, 2, 4 y 6, respectivamente, generan elementos de datos destino B, D, F y H, que también están contenidos, respectivamente, en carriles SIMD 0-1, 2-3, 4-5 y 6-7.

[0033] Por consiguiente, en el primer aspecto ejemplar de la FIG. 2A, la instrucción SIMD de anchura mixta 200 implica el uso eficiente de un espacio de instrucción o de un espacio de código (ya que solo se usa una instrucción SIMD, en lugar de dos o más instrucciones SIMD componentes), cuya implementación o ejecución evita grandes movimientos de datos a través de carriles SIMD.

[0034] Con referencia ahora a la FIG. 2B, se ilustra otro aspecto ejemplar con relación a la instrucción SIMD de anchura mixta 220. La instrucción SIMD 220 implica dos operandos vectoriales fuente: un primer operando vectorial fuente expresado como un único registro 222 y un segundo operando vectorial fuente expresado como un único registro 223, que tienen un primer conjunto y un segundo conjunto, respectivamente, de cuatro elementos de datos fuente de 16 bits. La instrucción SIMD 220 puede especificar una operación idéntica o común, tal como una multiplicación (por ejemplo, con redondeo) en los dos operandos vectoriales fuente, donde cuatro elementos de datos fuente de 16 bits del primer conjunto (en el registro 222) se multiplican por cuatro elementos de datos fuente de 16 bits correspondientes del segundo conjunto (en el registro 223) para producir cuatro resultados de 32 bits (donde la implementación de la instrucción SIMD 220 puede involucrar elementos lógicos tales como cuatro multiplicadores 16x16). Dado que se necesitan 128 bits para poder almacenar estos cuatro resultados de 32 bits, un operando vectorial destino se especifica en términos de un par de operandos vectoriales componentes: un primer operando vectorial destino componente y un segundo operando vectorial destino componente (estos pueden expresarse, de manera correspondiente, como un primer registro de 64 bits 224x y un segundo registro de 64 bits 224y). Se observa que la instrucción SIMD 220 también puede ser aplicable a la adición de elementos de datos fuente del primer conjunto con elementos de datos fuente correspondientes del segundo conjunto, donde los resultados correspondientes pueden consumir más de 16 bits (incluso si no todos los 32 bits) para cada elemento de datos destino.

[0035] En la FIG. 2B, a los elementos de datos fuente del primer y segundo conjuntos se les asigna un orden secuencial, que se muestra de manera representativa como 0, 1, 2, 3 y 0', 1', 2', 3', respectivamente. El primer operando vectorial destino componente en el primer registro 224x contiene un primer subconjunto de los resultados de la instrucción SIMD 220 (mostrada como elementos de datos destino de 32 bits A y C) correspondientes a elementos de datos fuente de numeración par de operandos fuente 222 y 223; y de manera similar, el segundo operando vectorial destino componente en el segundo registro 224y contiene un segundo subconjunto de los resultados de la instrucción SIMD 220 (mostrada como elementos de datos de 32 bits B y D) correspondientes a elementos de datos fuente de numeración impar de los operandos fuente 222 y 223. En este caso, se observa que los elementos de datos fuente de numeración par (0, 0') y (2, 2') del primer operando vectorial fuente 222 y del segundo operando vectorial fuente 223, respectivamente, generan elementos de datos A y C del primer operando vectorial destino 224x; y los elementos de datos de numeración impar (1, 1') y (3, 3') del primer operando vectorial fuente 222

y del segundo operando vectorial fuente 223, respectivamente, generan elementos de datos B y D del segundo operando vectorial destino 224y.

[0036] De nuevo, en el segundo aspecto ejemplar de la FIG. 2B se observa que la instrucción SIMD 220 de anchura mixta logra una eficiencia de espacio de código utilizando una sola instrucción SIMD de anchura mixta en lugar de dos o más instrucciones SIMD componentes. Además, se observa que los movimientos a través de carriles SIMD también se minimizan en este aspecto. En general, el primer conjunto de elementos de datos fuente y el segundo conjunto de elementos de datos fuente están en carriles SIMD respectivos, y generan a partir de cada uno de los elementos de datos fuente del primer conjunto y un elemento correspondiente de los elementos de datos fuente del segundo conjunto, un elemento de datos destino en el carril SIMD respectivo o un carril SIMD adyacente al carril SIMD respectivo. Por ejemplo, considerando cuatro carriles SIMD de 16 bits 0-3 que comprenden el primer conjunto de elementos de datos fuente 0-3 (o el segundo conjunto de elementos de datos fuente 0'-3'), respectivamente, el movimiento de datos para un elemento de datos fuente del primero y el segundo para generar elementos de datos destino correspondientes A-D está contenido dentro del mismo carril SIMD y, como máximo, un carril SIMD adyacente (por ejemplo, elementos de datos fuente de numeración par (0, 0') y (2, 2') en carriles SIMD 0 y 2, respectivamente, generan elementos de datos destino A y C en los carriles SIMD 0-1 y 2-4; y de manera similar, elementos de datos fuente de numeración impar (1, 1') y (3, 3') en carriles SIMD 1 y 3, respectivamente, generan elementos de datos destino B y D en los carriles SIMD 0-1 y 2-4).

[0037] La FIG. 2C representa un tercer aspecto ejemplar relacionado con la instrucción SIMD de anchura mixta 240. A diferencia de las instrucciones SIMD de anchura mixta 200 y 220, un operando vectorial fuente de la instrucción SIMD de anchura mixta 240 se especifica como un par de operandos vectoriales componentes o se expresa como un par de registros. Cabe señalar que la instrucción SIMD de anchura mixta 240 es diferente de la instrucción SIMD de anchura mixta 220 porque la instrucción SIMD de anchura mixta 220 incluía dos operandos vectoriales fuente individuales, donde los elementos de datos de un operando vectorial fuente se especificaron para interactuar (por ejemplo, multiplicarse) con elementos de datos de otro operando vectorial fuente. Por otro lado, en la instrucción SIMD de anchura mixta 240, se especifica un par de operandos vectoriales componentes fuente porque no hacerlo habría consumido dos instrucciones individuales. Por ejemplo, la instrucción SIMD 240 puede implicar una operación común de una función de desplazamiento a la derecha de 16 bits a 8 bits que se realizará en ocho elementos de datos fuente de 16 bits para obtener un resultado de ocho elementos de datos destino de 8 bits (donde la implementación de la instrucción SIMD 240 puede involucrar elementos lógicos tales como ocho desplazadores a la derecha de 8 bits). Sin embargo, dado que ocho elementos de datos fuente de 16 bits consumen 128 bits, las implementaciones convencionales habrían dividido esta operación para realizarse usando dos instrucciones SIMD componentes. Por otro lado, en el aspecto ejemplar de la FIG. 2C, un par de operandos vectoriales fuente que comprenden el primer operando vectorial fuente componente en el primer registro 242x y un segundo operando vectorial fuente componente en el segundo registro 242y se especifican mediante la instrucción SIMD 240. En consecuencia, el espacio de código se usa de manera eficiente.

[0038] El operando vectorial destino se expresa como un único registro de 64 bits 244 en este caso y comprende ocho elementos de datos destino de 8 bits que son el resultado de la instrucción SIMD 240. Por consiguiente, se asigna un orden secuencial a los elementos de datos destino del operando vectorial destino en el registro 244, que se muestran con los números de referencia 0-7. Los elementos de datos fuente del par de operandos vectoriales componentes fuente (expresados como un par de registros 242x, 242y) están dispuestos de tal manera que el primer registro 242x que comprende un primer subconjunto de elementos de datos fuente A, C, E y G generará los resultados correspondientes a los elementos de datos destino de numeración par 0, 2, 4 y 6 del operando vectorial destino en el registro 244, respectivamente; y el segundo registro 242y que comprende un segundo subconjunto de elementos de datos fuente B, D, F y H, generará los resultados correspondientes a los elementos de datos destino de numeración impar 1, 3, 5 y 7, respectivamente, del operando vectorial destino en el registro 244.

[0039] Por lo tanto, el espacio de código se puede utilizar de manera efectiva y el movimiento de datos a través de carriles SIMD se puede minimizar incluso en los casos en que los operandos vectoriales fuente son más anchos que los operandos vectoriales destino, especificando un par de operandos vectoriales fuente componente o expresando el operando vectorial fuente como un par de registros. El movimiento a través de los carriles SIMD en la ejecución de la instrucción SIMD 240 también se minimiza. En general, se observa que los elementos de datos destino están en carriles SIMD respectivos, y cada uno de los elementos de datos destino se genera a partir de un elemento de datos fuente en el carril SIMD respectivo o un carril SIMD adyacente al carril SIMD respectivo. Por ejemplo, considerando ocho carriles SIMD de 8 bits correspondientes a los ocho elementos de datos destino 0-7, se observa que los elementos de datos fuente A, C, E y G se moverán desde los carriles SIMD 0-1, 2-3, 4-5 y 6-7, respectivamente, para generar los resultados correspondientes a elementos de datos destino de numeración par en los carriles SIMD 0, 2, 4 y 6; y los elementos de datos fuente B, D, F y H, se moverán desde los carriles SIMD 0-1, 2-3, 4-5 y 6-7, respectivamente, para generar los resultados correspondientes a los elementos de datos destino de numeración par en los carriles SIMD 1, 3, 5 y 7. En cualquier caso, el movimiento está contenido dentro de dos carriles SIMD.

[0040] En consecuencia, se apreciará que los aspectos incluyen diversos procedimientos para llevar a cabo los procesos, funciones y/o algoritmos divulgados en el presente documento. Por ejemplo, como se ilustra en la FIG. 3A, un aspecto puede incluir un procedimiento 300 para realizar una operación de tipo "una instrucción, múltiples datos"

(SIMD) de anchura mixta, de acuerdo con las FIGS. 2A-B, por ejemplo.

- 5 **[0041]** En el bloque 302, el procedimiento 300 incluye recibir, por un procesador (por ejemplo, el procesador 402 de la FIG. 4, que se explicará posteriormente), y con referencia, por ejemplo, a la FIG. 2A, una instrucción SIMD (por ejemplo, instrucción SIMD 200) que comprende al menos un primer operando vectorial fuente (por ejemplo, en el registro 202) que comprende un primer conjunto de elementos de datos fuente (por ejemplo, elementos de datos fuente 0-7) de una primera anchura de bits (por ejemplo, 8 bits); y al menos un operando vectorial destino (por ejemplo, en el par de registros 204x, 204y) que comprende elementos de datos destino (por ejemplo, elementos de datos destino A-H) de una segunda anchura de bits (por ejemplo, 16 bits), donde la segunda anchura de bits es dos veces la primera anchura de bits, donde el operando vectorial destino comprende un par de registros que incluyen un primer registro (por ejemplo, 204x) que comprende un primer subconjunto de los elementos de datos destino (por ejemplo, elementos de datos destino A, C, E, G) y un segundo registro que comprende un segundo subconjunto de los elementos de datos destino (por ejemplo, elementos de datos destino B, D, F, H).
- 15 **[0042]** En el bloque 303 (que se muestra que incluye los bloques 304 y 306), el procedimiento 300 incluye además ejecutar la instrucción SIMD de anchura mixta en el procesador. Específicamente, considerando un orden secuencial (por ejemplo, 0-7) asignado a los elementos de datos fuente en el bloque 304, el bloque 306 incluye la ejecución de la instrucción SIMD en el procesador. Con más detalle, el bloque 306 incluye los bloques componentes 306a y 306b, que pueden realizarse en paralelo.
- 20 **[0043]** El bloque 306a incluye generar el primer subconjunto de elementos de datos destino (por ejemplo, elementos de datos destino A, C, E, G) en el primer registro (por ejemplo, primer registro 204x) a partir de elementos de datos fuente de numeración par (por ejemplo, elementos de datos fuente 0, 2, 4, 6) del primer conjunto.
- 25 **[0044]** El bloque 306b incluye generar el segundo subconjunto de elementos de datos destino (por ejemplo, elementos de datos destino B, D, F, H) en el segundo registro (por ejemplo, segundo registro 204y) a partir de elementos de datos fuente de numeración impar (por ejemplo, elementos de datos fuente 1, 3, 5, 7) del primer conjunto.
- 30 **[0045]** En general, la instrucción SIMD del procedimiento 300 puede ser una de entre una función de elevación al cuadrado, función de desplazamiento a la izquierda, incremento o adición de un valor constante, de los elementos de datos fuente del primer conjunto. La eficacia del espacio de código se logra utilizando una única instrucción SIMD en el procedimiento 300. El movimiento a través de los carriles SIMD también se minimiza en el procedimiento 300, donde el primer conjunto de elementos de datos fuente se encuentra en carriles SIMD respectivos, y el procedimiento 300 incluye generar a partir de cada uno de los elementos de datos fuente (por ejemplo, el elemento de datos fuente 0 en el carril SIMD 0), un elemento de datos destino (por ejemplo, elemento de datos destino A) en el carril SIMD respectivo (por ejemplo, carril 0 SIMD) o un carril SIMD adyacente (por ejemplo, carril 1 SIMD) al carril SIMD respectivo.
- 35 **[0046]** También se observará que, aunque no se muestran por separado, el procedimiento 300 también puede incluir un procedimiento para implementar la instrucción SIMD 220 de la FIG. 2B, que comprende además, por ejemplo, recibir en el bloque 302, un segundo operando vectorial fuente, que comprende un segundo conjunto de elementos de datos fuente de la primera anchura de bits (por ejemplo, primer y segundo operandos vectoriales fuente en los registros 222 y 223), y el orden secuencial del primer conjunto de elementos de datos fuente corresponde a un orden secuencial del segundo conjunto de elementos de datos fuente. En este caso, en base al orden secuencial asignado en el bloque 304, el bloque 306 incluye la ejecución de la instrucción SIMD en el procesador, lo que comprende que el bloque 306a genere el primer subconjunto de elementos de datos destino en el primer registro a partir de elementos de datos fuente de numeración par del primer conjunto y elementos de datos fuente de numeración par del segundo conjunto, y que el bloque 306b genere el segundo subconjunto de elementos de datos destino en el segundo registro a partir de elementos de datos fuente de numeración impar del primer conjunto y elementos de datos fuente de numeración par del segundo conjunto. En este caso, la instrucción SIMD puede ser una multiplicación o adición de los elementos de datos fuente del primer conjunto con elementos de datos fuente correspondientes del segundo conjunto, donde el primer conjunto de elementos de datos fuente y el segundo conjunto de elementos de datos fuente están en carriles SIMD respectivos, y genera a partir de cada uno de los elementos de datos fuente del primer conjunto y un elemento correspondiente de los elementos de datos fuente del segundo conjunto, un elemento de datos destino en el carril SIMD respectivo o un carril SIMD adyacente al carril SIMD respectivo.
- 40 **[0047]** Con referencia a la FIG. 3B, se ilustra otro procedimiento para realizar los procesos, funciones y/o algoritmos divulgados en el presente documento. Por ejemplo, como se ilustra en la FIG. 3B, el procedimiento 300 incluye otro procedimiento para realizar una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta, de acuerdo con la FIG. 2C, por ejemplo.
- 45 **[0048]** En el bloque 352, el procedimiento 350 incluye recibir, por un procesador (por ejemplo, el procesador 402), una instrucción SIMD (por ejemplo, la instrucción SIMD 240) que comprende: al menos un operando vectorial fuente (por ejemplo, en los registros 242x, 242y) que comprende elementos de datos fuente (por ejemplo, elementos de datos fuente A-H) de una primera anchura de bits (por ejemplo, 16 bits); y al menos un operando vectorial destino (por ejemplo, en el registro 244) que comprende elementos de datos destino (por ejemplo, elementos de datos destino 0-
- 50
- 55
- 60
- 65

7) de una segunda anchura de bits (por ejemplo, 8 bits), donde la segunda anchura de bits es la mitad de la primera anchura de bits, donde el operando vectorial fuente comprende un par de registros, que incluyen un primer registro (por ejemplo, primer registro 242x) que comprende un primer subconjunto de los elementos de datos fuente (por ejemplo, elementos de datos destino 0, 2, 4, 6) y un segundo registro (por ejemplo, segundo registro 242y) que comprende un segundo subconjunto de los elementos de datos fuente (por ejemplo, elementos de datos destino 1, 3, 5, 7).

[0049] En el bloque 354, se asigna un orden secuencial a los elementos de datos destino, y en el bloque 356, se ejecuta la instrucción SIMD. El bloque 356 incluye subbloques 356a y 356b, que también pueden llevarse a cabo en paralelo.

[0050] El bloque 356a incluye generar elementos de datos destino de numeración par (por ejemplo, elementos de datos destino 0, 2, 4, 6) a partir del primer subconjunto correspondiente de elementos de datos fuente en el primer registro (por ejemplo, elementos de datos fuente A, C, E, G).

[0051] El bloque 356b incluye generar elementos de datos destino de numeración impar (por ejemplo, elementos de datos destino 1, 3, 5, 7) a partir del segundo subconjunto correspondiente de elementos de datos fuente en el segundo registro (por ejemplo, elementos de datos fuente B, D, F, H).

[0052] En aspectos ejemplares, la instrucción SIMD del procedimiento 350 puede ser una función de desplazamiento a la derecha de los elementos de datos fuente, donde los elementos de datos destino están en carriles SIMD respectivos (por ejemplo, carriles SIMD 0-7), y generan cada uno de los elementos de datos destino (por ejemplo, elemento de datos destino 0) a partir de un elemento de datos fuente (por ejemplo, elemento de datos fuente A) en el carril SIMD respectivo (por ejemplo, carril SIMD 0) o un carril SIMD adyacente (por ejemplo, carril 1 SIMD) al carril SIMD respectivo.

[0053] Con referencia a la FIG. 4, se muestra un diagrama de bloques de un aspecto ilustrativo particular de un dispositivo inalámbrico 400 de acuerdo con aspectos ejemplares. El dispositivo inalámbrico 400 incluye el procesador 402, que puede configurarse (por ejemplo, incluir lógica de ejecución) para admitir e implementar la ejecución de instrucciones SIMD de anchura mixta ejemplares, por ejemplo, de acuerdo con los procedimientos 300 y 350 de la FIG. 3A y la FIG. 3B, respectivamente. Como se muestra en la FIG. 4, el procesador 402 puede estar en comunicación con la memoria 432. El procesador 402 puede incluir un archivo de registro (no mostrado) que contiene registros físicos correspondientes a los registros (por ejemplo, registros lógicos) en cuanto a qué operandos de las instrucciones SIMD ejemplares se expresan. Al archivo de registro se le pueden suministrar datos de la memoria 432 en algunos aspectos. Aunque no se muestra, también se pueden incluir una o más memorias caché u otras estructuras de memoria en el dispositivo inalámbrico 400.

[0054] La FIG. 4 también muestra un controlador de visualización 426 que está acoplado al procesador 402 y a un dispositivo de visualización 428. Un codificador/descodificador (CÓDEC) 434 (por ejemplo, un CODEC de audio y/o de voz) se puede acoplar al procesador 402. También se ilustran otros componentes, tales como un controlador inalámbrico 440 (que puede incluir un módem). Un altavoz 436 y un micrófono 438 pueden estar acoplados al CÓDEC 434. La FIG. 4 indica además que el controlador inalámbrico 440 puede estar acoplado a la antena inalámbrica 442. En un aspecto particular, el procesador 402, el controlador de visualización 426, la memoria 432, el CÓDEC 434 y el controlador inalámbrico 440 están incluidos en un dispositivo de sistema en cápsula o sistema en chip 422.

[0055] En un aspecto particular, un dispositivo de entrada 430 y una fuente de alimentación 444 están acoplados al dispositivo de sistema en chip 422. Además, en un aspecto particular, como se ilustra en la FIG. 4, el dispositivo de visualización 428, el dispositivo de entrada 430, el altavoz 436, el micrófono 438, la antena inalámbrica 442 y la fuente de alimentación 444 son externos al dispositivo de sistema en chip 422. Sin embargo, cada uno del dispositivo de visualización 428, el dispositivo de entrada 430, el altavoz 436, el micrófono 438, la antena inalámbrica 442 y la fuente de alimentación 444 se puede acoplar a un componente del dispositivo de sistema en chip 422, tal como una interfaz o un controlador.

[0056] Cabe destacar que, aunque la FIG. 4 representa un dispositivo de comunicaciones inalámbricas, el procesador 402 y la memoria 432 también pueden integrarse en un descodificador, un reproductor de música, un reproductor de vídeo, una unidad de entretenimiento, un dispositivo de navegación, un asistente digital personal (PDA), una unidad de datos de ubicación fija, un dispositivo de comunicaciones o un ordenador. Además, al menos uno o más aspectos ejemplares del dispositivo inalámbrico 400 pueden integrarse en al menos una pastilla semiconductora.

[0057] Los expertos en la técnica apreciarán que la información y las señales se pueden representar usando cualquiera de una variedad de tecnologías y técnicas diferentes. Por ejemplo, los datos, instrucciones, comandos, información, señales, bits, símbolos y segmentos a los que se puede haber hecho referencia a lo largo de la descripción anterior se pueden representar mediante tensiones, corrientes, ondas electromagnéticas, campos o partículas magnéticos, campos o partículas ópticos o cualquier combinación de los mismos.

5 **[0058]** Además, los expertos en la técnica apreciarán que los diversos bloques lógicos, módulos, circuitos y etapas de algoritmo ilustrativos descritos en relación con los aspectos 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, módulos, circuitos y etapas ilustrativos, en general, en lo que respecta a su funcionalidad. Que dicha funcionalidad se implemente como hardware o software depende de las restricciones particulares de aplicación y de diseño impuestas al sistema global. Los expertos en la técnica pueden implementar la funcionalidad descrita de formas distintas para cada solicitud particular, pero no debería interpretarse que dichas decisiones de implementación suponen apartarse del alcance de la presente invención.

10 **[0059]** Los procedimientos, secuencias y/o algoritmos descritos en relación con los aspectos divulgados en el presente documento pueden realizarse directamente en hardware, en un módulo de software ejecutado por un procesador o en una combinación de ambas cosas. Un módulo de software puede residir en una memoria RAM, memoria flash, memoria ROM, memoria EPROM, memoria EEPROM, registros, un disco duro, un disco extraíble, un CDROM o en cualquier otra forma de medio de almacenamiento conocida en la técnica. Un medio de almacenamiento ejemplar está acoplado al procesador de modo que el procesador pueda leer información de, y escribir información en, el medio de almacenamiento. De forma alternativa, el medio de almacenamiento puede estar integrado en el procesador.

15 **[0060]** Por consiguiente, un aspecto de la invención puede incluir medios legibles por ordenador (por ejemplo, un medio de almacenamiento legible por ordenador no transitorio) que incorporan un procedimiento para implementar instrucciones SIMD de anchura mixta (por ejemplo, de acuerdo con los procedimientos 300 y 350 descritos anteriormente, para implementar instrucciones SIMD de las FIGS. 2A-C). En consecuencia, la invención no se limita a los ejemplos ilustrados, y cualquier medio para realizar la funcionalidad descrita en el presente documento se incluye en los aspectos de la invención.

20 **[0061]** Si bien la divulgación anterior muestra aspectos ilustrativos de la invención, cabe destacar que podrían realizarse diversos cambios y modificaciones en el presente documento sin apartarse del alcance de la invención definida en las reivindicaciones adjuntas. Además, aunque los elementos de la invención se pueden describir o reivindicar en singular, se contempla el plural a menos que se indique explícitamente la limitación al singular.

REIVINDICACIONES

1. Un procedimiento para realizar una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta, comprendiendo el procedimiento:
- 5 recibir (300), por un procesador, una instrucción SIMD que especifica:
- un único operando vectorial fuente, que comprende
- 10 un primer conjunto de elementos de datos fuente de una primera anchura de bits; y
- un operando vectorial destino que comprende elementos de datos destino de una segunda anchura de bits, donde la segunda anchura de bits es dos veces la primera anchura de bits,
- 15 en el que el único operando vectorial fuente único se expresa como un único registro, y el operando vectorial destino se expresa como un par de registros, que incluyen un primer registro que comprende un primer subconjunto de los elementos de datos destino y un segundo registro que comprende un segundo subconjunto de los elementos de datos destino; y
- 20 en base a un orden secuencial del primer conjunto de elementos de datos fuente, ejecutar la instrucción SIMD en el procesador, lo que comprende:
- generar (306a) el primer subconjunto de elementos de datos destino en el primer registro a partir de elementos de datos fuente de numeración par del primer conjunto; y
- 25 generar (306b) el segundo subconjunto de elementos de datos destino en el segundo registro a partir de elementos de datos fuente de numeración impar del primer conjunto.
2. El procedimiento de la reivindicación 1, en el que la instrucción SIMD es una de entre una función de elevación al cuadrado, función de desplazamiento a la izquierda, incremento o adición de un valor constante de los elementos de datos fuente del primer conjunto.
3. El procedimiento de la reivindicación 1, en el que el primer conjunto de elementos de datos fuente está en carriles SIMD respectivos, y generar a partir de cada uno de los elementos de datos fuente un elemento de datos destino en el carril SIMD respectivo o un carril SIMD adyacente al carril SIMD respectivo.
- 35 4. Un medio de almacenamiento no transitorio legible por ordenador, que comprende instrucciones ejecutables por un procesador que, cuando se ejecutan por el procesador, hacen que el procesador realice una operación de tipo "una instrucción, múltiples datos" (SIMD) de anchura mixta, donde el medio de almacenamiento no transitorio legible por ordenador comprende:
- 40 una instrucción SIMD que especifica:
- un único operando vectorial fuente, que comprende
- 45 un primer conjunto de elementos de datos fuente de una primera anchura de bits; y
- al menos un operando vectorial destino que comprende elementos de datos destino de una segunda anchura de bits, donde la segunda anchura de bits es dos veces la primera anchura de bits,
- 50 en el que el único operando vectorial fuente único se expresa como un único registro, y el operando vectorial destino se expresa como un par de registros, que incluyen un primer registro que comprende un primer subconjunto de los elementos de datos destino y un segundo registro que comprende un segundo subconjunto de los elementos de datos destino; y
- 55 en base a un orden secuencial del primer conjunto de elementos de datos fuente:
- código para generar (306a) el primer subconjunto de elementos de datos destino en el primer registro a partir de elementos de datos fuente de numeración par del primer conjunto; y
- 60 código para generar (306b) el segundo subconjunto de elementos de datos destino en el segundo registro a partir de elementos de datos fuente de numeración impar del primer conjunto.
5. El medio de almacenamiento no transitorio legible por ordenador de la reivindicación 4, en el que la instrucción SIMD es una de entre una función de elevación al cuadrado, función de desplazamiento a la izquierda,
- 65

incremento o adición de un valor constante de los elementos de datos fuente del primer conjunto.

- 5
6. El medio de almacenamiento no transitorio legible por ordenador de la reivindicación 4, en el que el primer conjunto de elementos de datos fuente está en carriles SIMD respectivos, y comprende código para generar a partir de cada uno de los elementos de datos fuente un elemento de datos destino en el carril SIMD respectivo o un carril SIMD adyacente al carril SIMD respectivo.

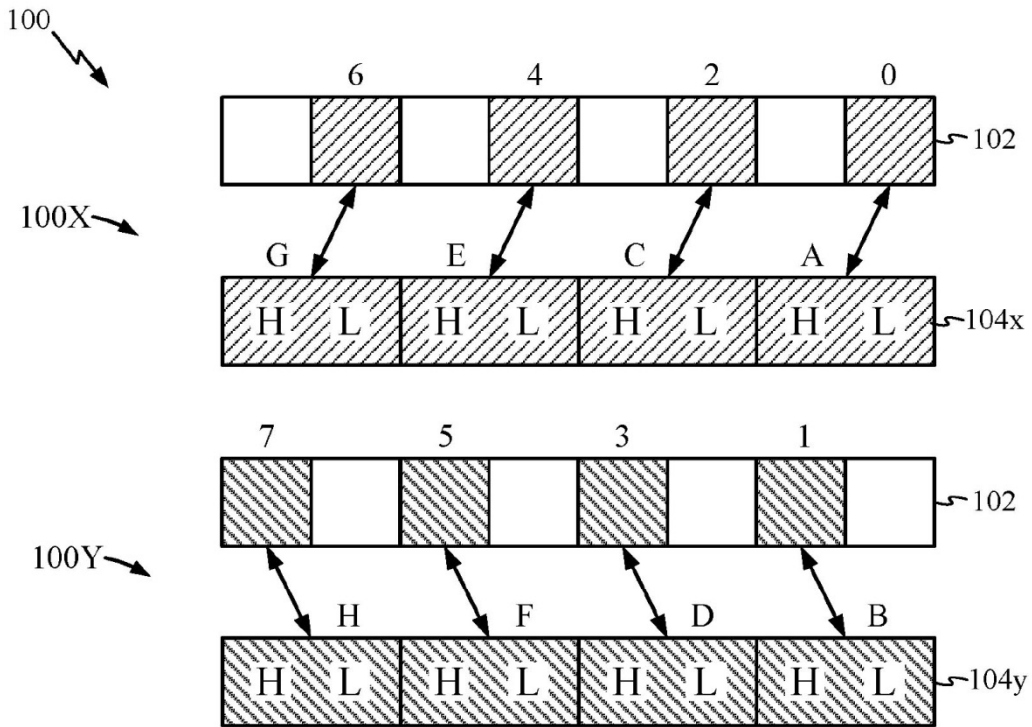


FIG. 1A
CONVENCIONAL

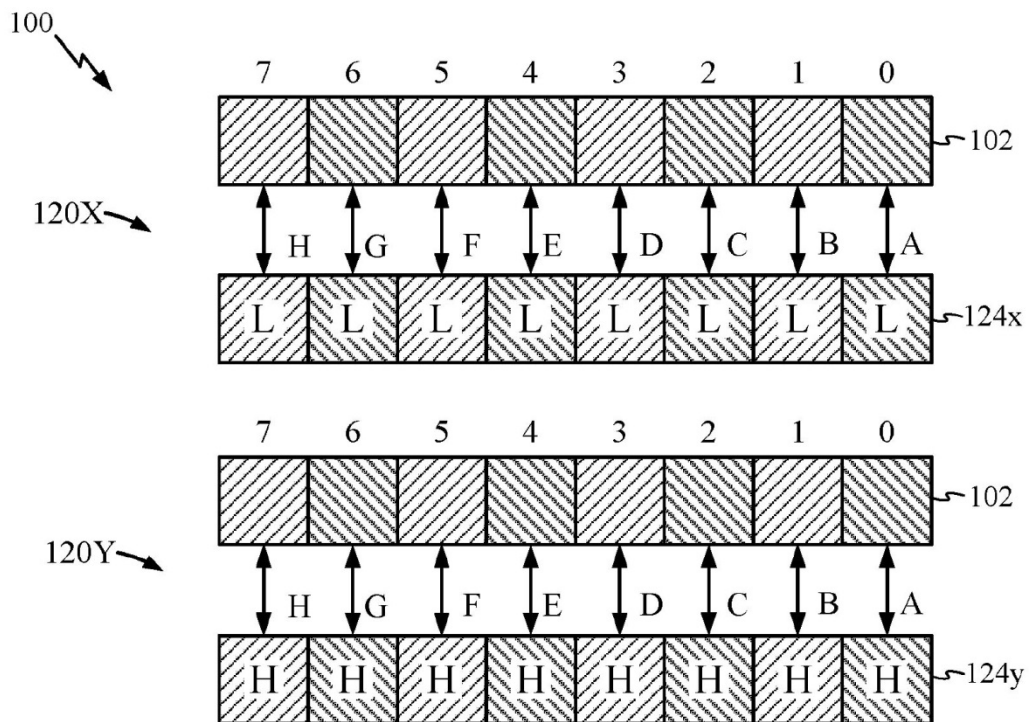


FIG. 1B
CONVENCIONAL

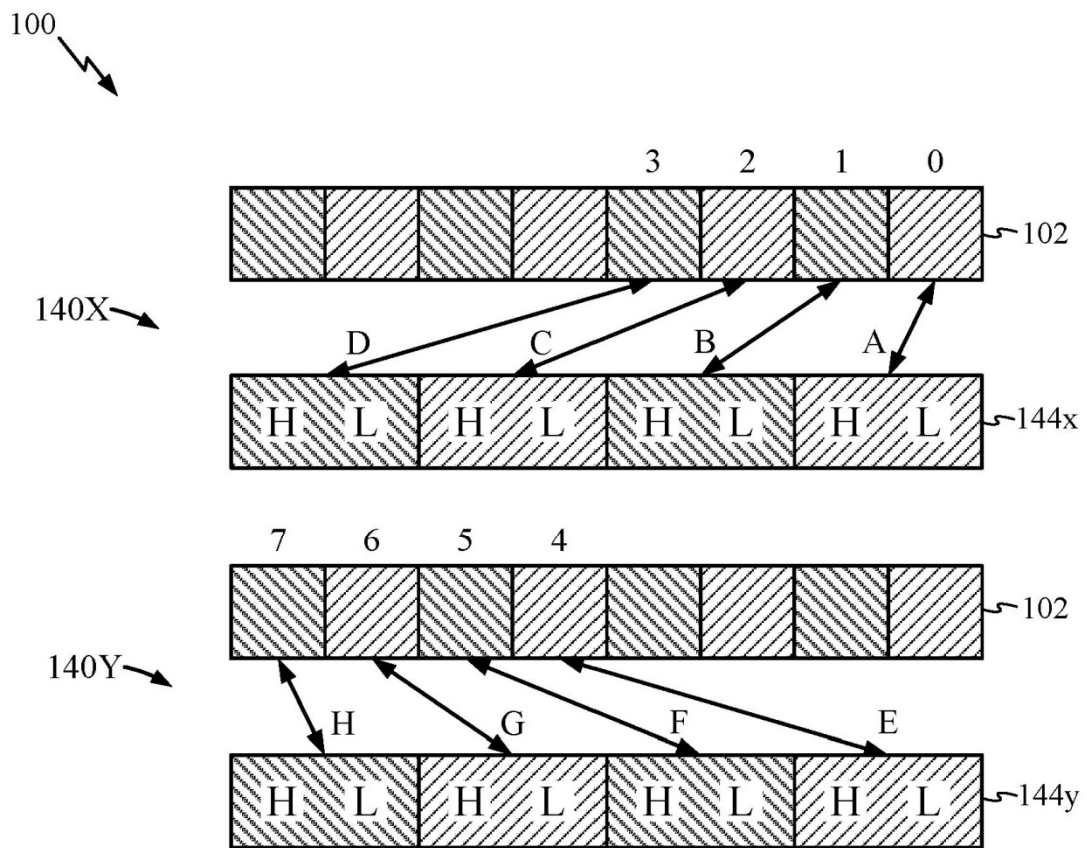


FIG. 1C
CONVENCIONAL

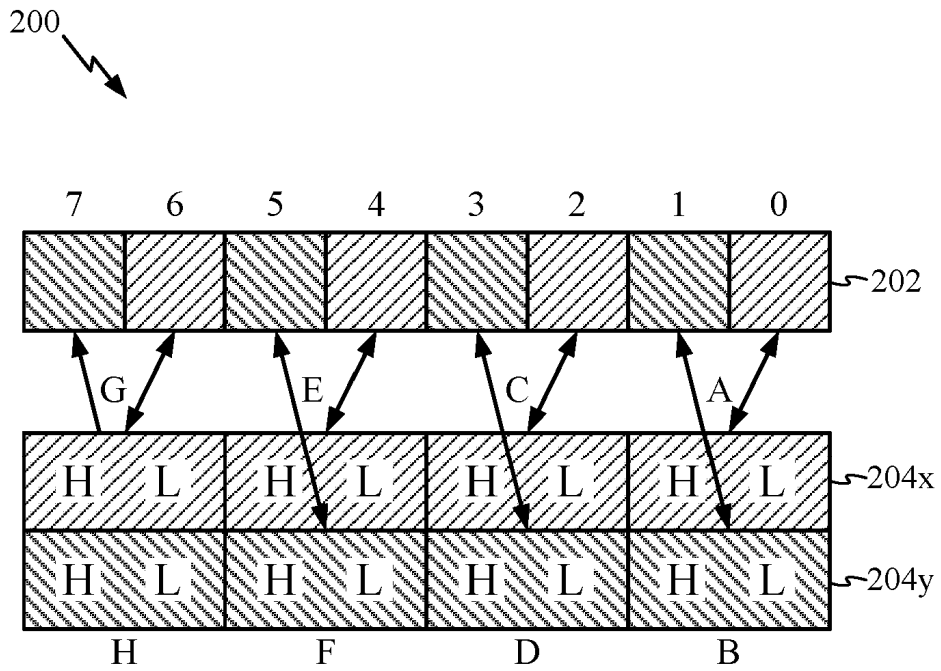


FIG. 2A

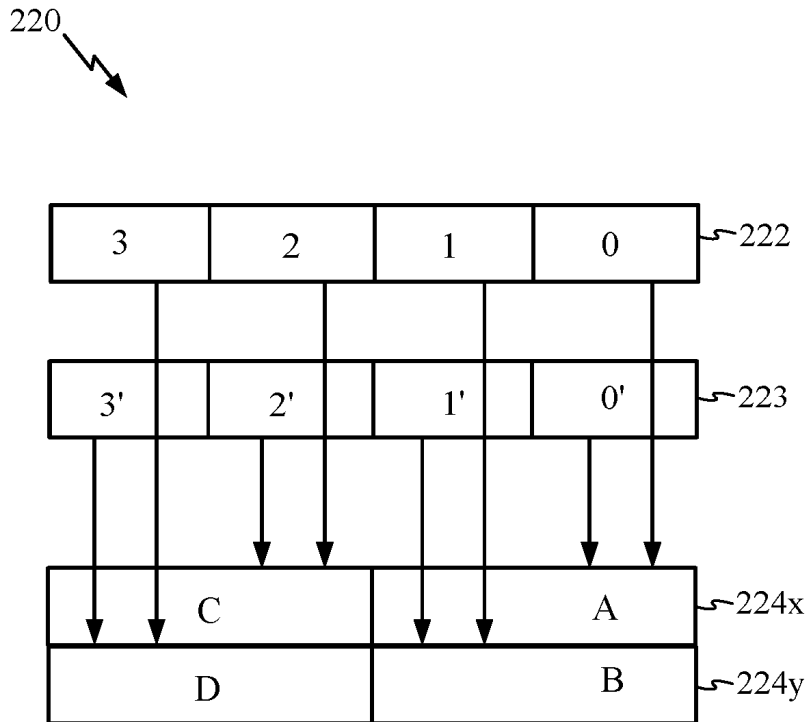


FIG. 2B

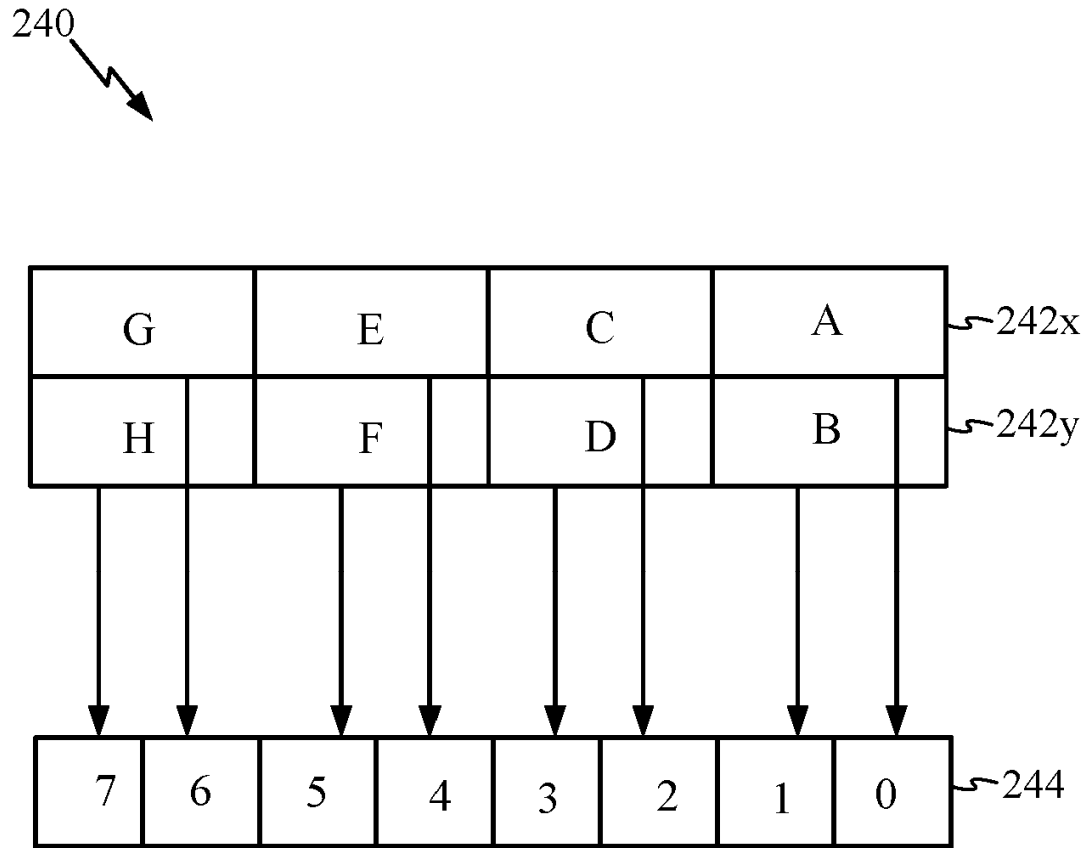


FIG. 2C

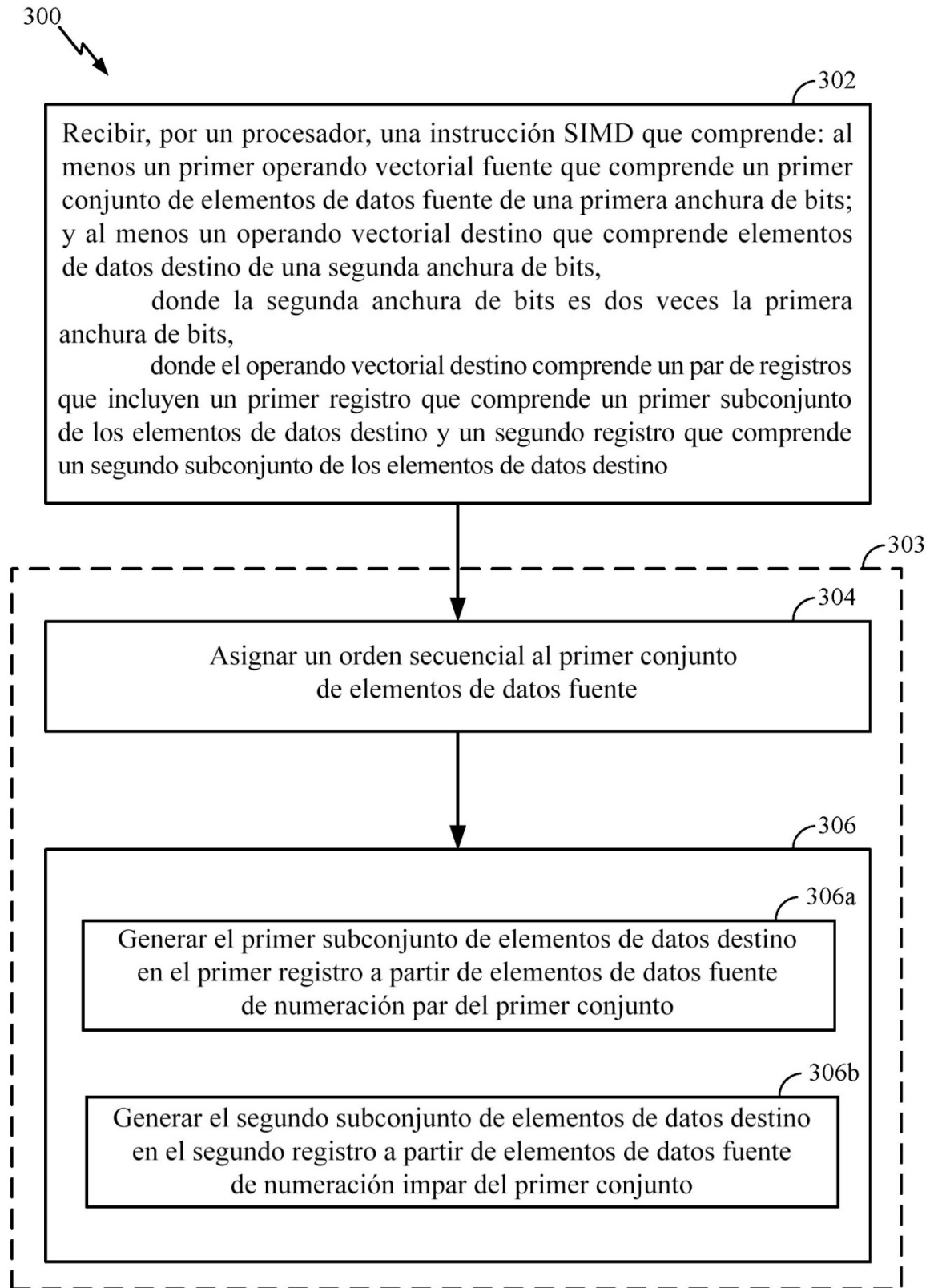


FIG. 3A

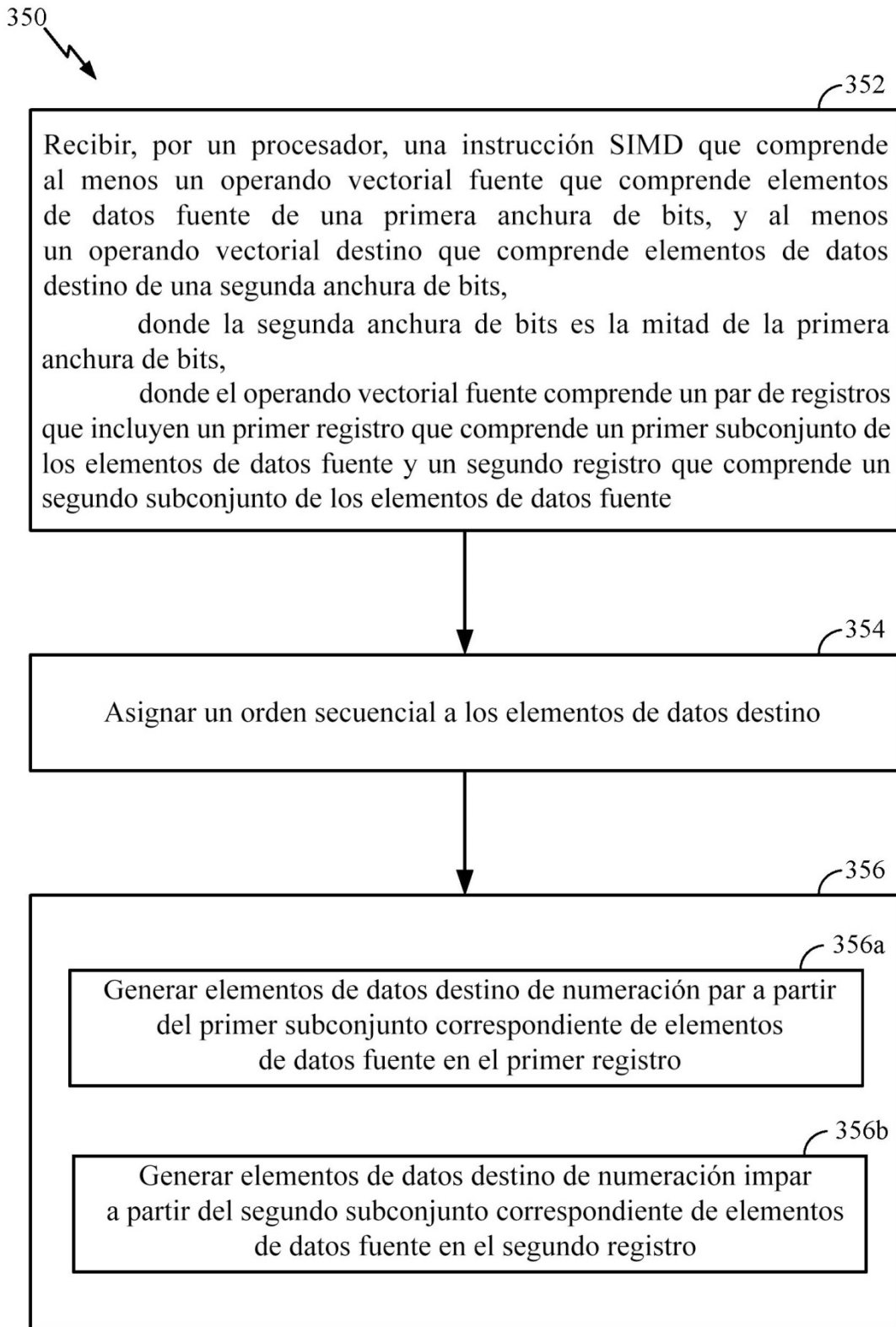


FIG. 3B

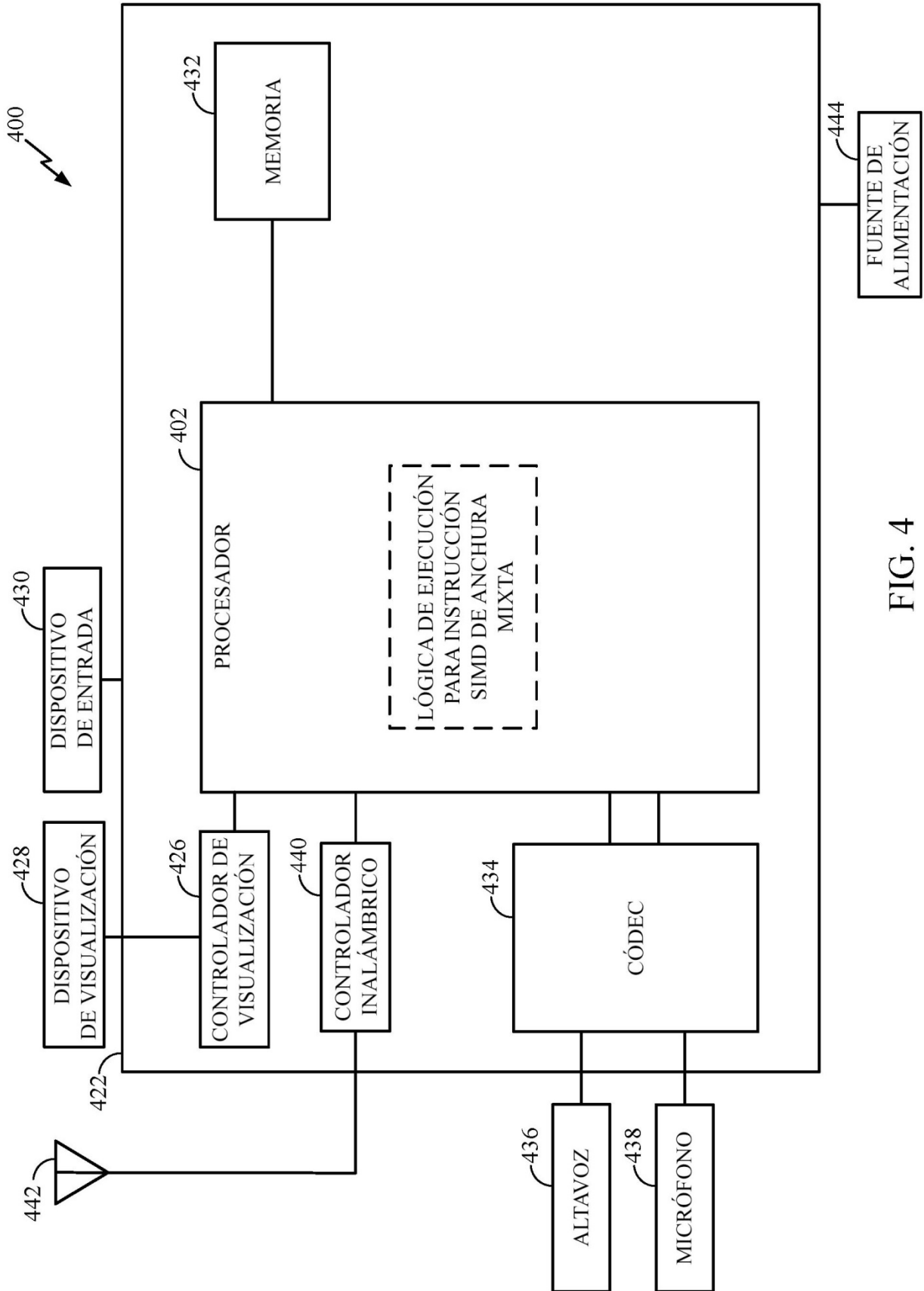


FIG. 4