



①9



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA

①1 Número de publicación: **2 316 653**

⑤1 Int. Cl.:
G06F 12/08 (2006.01)
G06F 12/12 (2006.01)

①2

TRADUCCIÓN DE PATENTE EUROPEA

T3

⑨6 Número de solicitud europea: **02807417 .7**
⑨6 Fecha de presentación : **16.09.2002**
⑨7 Número de publicación de la solicitud: **1502191**
⑨7 Fecha de publicación de la solicitud: **02.02.2005**

⑤4 Título: **Métodos y aparatos para controlar una memoria caché.**

③0 Prioridad: **06.05.2002 US 378199 P**
01.07.2002 US 187150

④5 Fecha de publicación de la mención BOPI:
16.04.2009

④5 Fecha de la publicación del folleto de la patente:
16.04.2009

⑦3 Titular/es: **Sony Computer Entertainment Inc.**
2-6-21, Minami-Aoyama
Minato-ku, Tokyo 107-0062, JP

⑦2 Inventor/es: **Magoshi, H.**

⑦4 Agente: **Elzaburu Márquez, Alberto**

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

Métodos y aparatos para controlar una memoria caché.

5 Estado de la técnica

La presente invención se refiere a métodos y aparatos para controlar una memoria caché y, más en concreto, a una técnica de control en la que se prohíbe el almacenamiento de datos en coma flotante en la memoria caché cuando dicho almacenamiento deba sobrescribir datos enteros, válidos.

10 En los últimos años ha existido un deseo insaciable de salidas más rápidas de los datos de procesamiento de los ordenadores porque las aplicaciones de ordenadores más avanzadas son cada vez más complejas y están pidiendo a los sistemas de microprocesamiento demandas cada vez mayores. Los sistemas de microprocesamiento convencionales (que emplean un microprocesador y una memoria asociada) tienen ciclos (es decir, el tiempo en que un microproce-

15 sador es capaz de manipular los datos) muy cortos, tales como un nanosegundo. Sin embargo, el tiempo necesario para acceder a los datos almacenados en la memoria principal puede ser considerablemente mayor que el ciclo del microprocesador. Por ejemplo, el tiempo de acceso necesario para obtener un octeto de datos de una memoria principal (implementada utilizando tecnología DRAM (dynamic random access memory; memoria dinámica de acceso aleatorio)) es del orden de alrededor de 60 nanosegundos.

20 Con el fin de mejorar el cuello de botella impuesto por el tiempo de acceso relativamente elevado a una memoria DRAM, los expertos en la materia han utilizado memorias caché. Una memoria caché aumenta la memoria principal con el fin de mejorar el rendimiento del sistema. Mientras que la memoria principal se implementa a menudo utilizando tecnología DRAM relativamente lenta y barata, la memoria caché se implementa, normalmente, utilizando tecnología

25 SRAM (static random access memory; memoria estática de acceso aleatorio) más cara y rápida. Dado que la memoria caché se implementa utilizando una tecnología de alto coste, normalmente tiene un tamaño mucho menor que la memoria principal.

La memoria caché puede disponerse de forma interna (on-chip) al microprocesador, lo que recibe el nombre de

30 memoria caché de nivel 1 (L1), o puede disponerse separada del microprocesador (off-chip) y recibe el nombre de memoria caché de nivel 2 (L2). Las memorias caché L1 tienen, normalmente, un tiempo de acceso mucho más pequeño que las memorias caché L2. También puede formarse un sistema de memorias caché L1 y L2 combinadas, en el que se empleen memorias caché internas y externas.

35 Debido al tamaño relativamente pequeño de las memorias caché, se han empleado algoritmos convencionales para determinar que datos deberían almacenarse en la memoria caché en diversos momentos del funcionamiento del sistema de microprocesamiento. Estos algoritmos convencionales pueden basarse, por ejemplo, en el concepto teórico de “emplazamiento de referencia”, que se aprovecha del hecho de que el microprocesador utiliza en cualquier instante

40 concreto partes relativamente pequeñas de un programa ejecutable. De este modo, de acuerdo con el concepto de emplazamiento de referencia, solamente pequeñas partes de un programa ejecutable se almacenan en la memoria caché en un instante concreto. Estos y otros algoritmos pueden emplearse también para controlar el almacenamiento y recuperación de los datos en la memoria caché (que pueden ser utilizados por el programa ejecutable).

Las particularidades de los algoritmos conocidos para aprovechar el emplazamiento de referencia, o cualquier otro

45 concepto, para controlar el almacenamiento de programas ejecutables y/o datos en una memoria caché son demasiado numerosas para presentarlas en esta descripción. Sin embargo, es suficiente con decir que cualquier algoritmo puede no ser adecuado en todas las aplicaciones ya que los objetivos del procesamiento de datos de las diversas aplicaciones pueden diferir significativamente.

50 La patente de Estados Unidos US-A-4.942.518 (Weatherford) describe una memoria caché L1 con memoria en paralelo y funciones de direccionamiento para mejorar aún más las velocidades efectivas de acceso.

En algunos casos, un sistema de microprocesador que emplee una memoria caché puede ser requerido para que

55 procese datos enteros y en coma flotante. Las aplicaciones que emplean dichos datos en coma flotante pueden requerir disposiciones (arrays) de datos en coma flotante muy grandes, que son mucho mayores que las capacidades de tamaño de una memoria caché de nivel L1 y que tienen un emplazamiento de direcciones muy bajo. La carga de datos en coma flotante desde un array grande de este tipo puede corromper la memoria caché de nivel L1, en especial cuando la memoria caché de nivel L1 incluye datos enteros, tales como punteros de dirección y similares. Los métodos y aparatos

60 convencionales que se han diseñado para evitar la corrupción de los datos enteros de la memoria caché de nivel L1 imponen que no se acceda a la memoria caché de nivel L1 cuando se quieran cargar datos en coma flotante, y que en su lugar se acceda directamente a una memoria caché de nivel L2 (o a la memoria principal). Detalles adicionales relacionados con este tipo de control pueden encontrarse en la Patente de Estados Unidos número 5.510.934.

Desafortunadamente, el acceso reducido a la memoria caché de nivel L1 impuesto por estas técnicas de control

65 convencional conduce a un rendimiento global inferior del sistema de microprocesamiento. Realmente, no se optimiza el uso de la velocidad muy alta de la memoria caché de nivel L1 y, de hecho, dicho uso se prohíbe completamente a favor de la memoria caché de nivel L2, más lenta, a la hora de cargar datos en coma flotante.

Por consiguiente, existe en la técnica la necesidad de nuevos métodos y aparatos para el control de una memoria caché, que puede incluir una memoria caché de nivel L1, una memoria caché de nivel L2 y/o una combinación de las mismas, con el fin de mejorar la eficiencia de la memoria, aumentar el rendimiento del procesamiento y mejorar la calidad del procesamiento global de datos llevado a cabo por el sistema.

5

Resumen de la invención

10 Cuando una petición de acceso a memoria para datos en coma flotante no puede satisfacerse mediante el acceso a la memoria caché, es decir, cuando se produce un error de caché, es deseable ejecutar una secuencia de recarga de datos en la que el dato en coma flotante se obtiene de la memoria principal y se almacena en la memoria caché. Cuando se emplea una memoria caché de nivel L1 y se hace una petición de acceso a memoria para datos en coma flotante, sería deseable acceder primero a la memoria caché de nivel L1 para satisfacer la petición y, si no puede ser satisfecha, acceder a una memoria caché de nivel L2 o a la memoria principal para recargar la memoria caché de nivel L1. En cualquier caso, cuando el sistema de microprocesamiento opera con datos enteros y en coma flotante, es deseable evitar la sobreescritura de datos enteros válidos con datos en coma flotante en la memoria caché de nivel L1.

20 Con este propósito, de acuerdo con uno o más aspectos de la presente invención, un método para controlar una memoria caché comprende los pasos de: recibir una dirección para al menos una entre las opciones de escribir datos en y de recuperar datos de la memoria caché, incluyendo la dirección bits de dirección y bits de índice; acceder a una o más líneas de caché de la memoria caché correspondientes a los bits de índice de la dirección, incluyendo cada línea de caché una etiqueta de dirección, una bandera de validez de datos y una bandera de tipo de datos; determinar si los datos en al menos una de la una o más líneas de caché son válidos en base a las banderas de validez de datos; determinar qué tipo de datos se han almacenado en la al menos una línea de caché en base a la bandera de tipo de datos; y prohibir la sobreescritura de datos en coma flotante en la al menos una línea de caché cuando los datos en la misma sean válidos y los datos que han sido almacenados en la al menos una línea de caché sean datos enteros.

30 Preferentemente, el método comprende además los pasos de: (i) fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos en la misma son válidos; y (ii) fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos en la al menos una línea de caché son de un tipo entero, cuando se prohíba la sobreescritura de datos en coma flotante en la al menos una línea de caché.

35

Hay que indicar que la bandera de validez de datos y/o la bandera de tipo de datos pueden ser banderas de un único bit.

40 Preferentemente, el método comprende además los pasos de: permitir la sobreescritura de datos enteros en la al menos una línea de caché cuando la bandera de validez de datos indique que la al menos una línea de caché no contiene datos válidos; fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos enteros son válidos; y fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

45 Alternativamente, o adicionalmente, el método puede comprender además los pasos de: permitir la sobreescritura de datos en coma flotante en la al menos una línea de caché cuando la bandera de validez de datos indique que la al menos una línea de caché no contiene datos válidos; fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos en coma flotante son válidos; y fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos en coma flotante son de un tipo en coma flotante.

50

Alternativamente, o adicionalmente, el método comprende además, preferentemente, el paso de permitir la sobreescritura de datos enteros en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos de la al menos una línea de caché son de un tipo entero. En este caso, el método, preferentemente, comprende además los pasos de: fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos enteros son válidos; y fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

60 Alternativamente, o adicionalmente, el método comprende además, preferentemente, el paso de permitir la sobreescritura de datos enteros en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos en la al menos una línea de caché son de un tipo en coma flotante. En este caso, el método, preferentemente, comprende además los pasos de: fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos enteros son válidos; y fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

65

ES 2 316 653 T3

Alternativamente, o adicionalmente a lo anterior, el método comprende además, preferentemente, el paso de permitir la sobreescritura de datos en coma flotante en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos de la al menos una línea de caché son de un tipo en coma flotante. En este caso, el método, preferentemente, comprende además los pasos de: fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos en coma flotante son válidos; y fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos en tipo flotante son de un tipo de coma flotante.

Cuando la memoria caché se implementa utilizando tecnología asociativa de conjuntos de N vías, en la que muchas líneas de caché pueden ser invocadas por una única petición de acceso a memoria para un conjunto de datos, y de acuerdo con uno o más aspectos adicionales de la presente invención, el método, preferentemente, comprende además los pasos de: acceder a todas las líneas de caché de la memoria caché correspondientes a los bits de índice de la dirección; determinar cuáles de las líneas de caché son válidas en base a las banderas de validez de datos; y permitir la sobreescritura de datos en coma flotante o de datos enteros en cualquiera de las líneas de caché en las que la correspondiente bandera de validez de datos indique que dichas líneas de caché no contienen datos válidos.

Alternativamente, o adicionalmente a lo anterior, el método, preferentemente, comprende además los pasos de: acceder a todas las líneas de caché de la memoria caché correspondientes a los bits de índice de la dirección, determinar cuáles de las líneas de caché son válidas en base a las banderas de validez de datos; y permitir la sobreescritura de datos enteros en cualquiera de las líneas de caché cuando todas las banderas de validez de datos indiquen que las líneas de caché contienen datos válidos.

El método, preferentemente, comprende además los pasos de: determinar qué tipo de datos se han almacenado en cada una de las líneas de caché en base a las respectivas banderas de tipo de datos; y permitir la sobreescritura de datos en coma flotante en cualquiera de las líneas de caché en las que las correspondientes banderas de tipo de datos indiquen que el tipo de datos que se han almacenado en dichas líneas de caché son datos en coma flotante cuando todas las banderas de validez de datos indiquen que las líneas de caché contienen datos válidos.

Además, el método, preferentemente, comprende además el paso de prohibir la sobreescritura de datos en coma flotante cuando todas las banderas de validez de datos indiquen que las líneas de caché contienen datos válidos y todas las banderas de tipo de datos indiquen que las líneas de caché contienen datos enteros.

De acuerdo con uno o más aspectos adicionales de la presente invención, cuando se prohíbe la sobreescritura de los datos en coma flotante en cualquiera de las líneas de caché, el método comprende además, preferentemente, el paso de transferir los datos en coma flotante a una unidad de lectura/escritura o a otra memoria caché. La otra memoria caché puede incluir solamente una línea de caché.

De acuerdo con uno o más aspectos adicionales de la presente invención, los métodos para controlar una memoria caché descritos hasta ahora, y/o que se describirán después en este documento, pueden llevarse a la práctica utilizando el hardware adecuado, tal como el mostrado en los dibujos que se indican a continuación. Dicho hardware puede implementarse utilizando cualquiera de las tecnologías conocidas, tales como circuitos digitales estándar, circuitos analógicos, cualquiera de los procesadores conocidos que son operativos para ejecutar programas de software y/o de firmware, uno o más dispositivos o sistemas digitales programables, tales como memorias PROM (programmable read only memory; memoria programable de sólo lectura), dispositivos PAL (programmable array logic), cualquier combinación de los anteriores, etc.

Breve descripción de los dibujos

Con fines de ilustrar la invención, en los dibujos se muestran formas que son preferidas en el momento presente pero, sin embargo, debe entenderse que la invención no está limitada a las disposiciones concretas mostradas.

- la figura 1 es un diagrama de bloques que muestra aspectos de aparatos (y/o aspectos de métodos) adecuados para controlar una memoria caché de acuerdo con uno o más realizaciones de la presente invención;

- la figura 2 es un diagrama de bloques que muestra aspectos adicionales del aparato de la figura 1 (y/o aspectos adicionales de los métodos) para controlar una memoria caché de acuerdo con una o más realizaciones adicionales de la presente invención;

- la figura 3 es un diagrama de flujo que ilustra ciertas acciones/funciones que pueden llevarse a cabo y/o implementarse de acuerdo con uno o más aspectos adicionales de la presente invención, tales como los mostrados en las figuras 1 y 2; y

- la figura 4 es un diagrama de bloques que muestra aspectos adicionales de un aparato (y/o aspectos adicionales de los métodos) para controlar una memoria caché de acuerdo con una o más realizaciones adicionales de la presente invención.

Mejor modo de llevar a cabo la invención

Con referencia a los dibujos, en los que valores numéricos iguales representan elementos iguales, en la figura 1 se muestra un diagrama de bloques que ilustra uno o más aspectos de la presente invención. Con fines de brevedad y claridad, nos referiremos al diagrama de bloques de la figura 1 y le describiremos en esta memoria como que muestra un aparato 100, debiendo entenderse, sin embargo, que la descripción puede ser fácilmente aplicable a varios aspectos de uno o más métodos con igual valor. El aparato 100 representa, preferentemente, un microprocesador e incluye, preferentemente, una unidad de lanzamiento de instrucciones 102, una unidad de lectura/escritura 104, una unidad de enteros 106, una unidad de coma flotante 108, una memoria caché 110 y una memoria adicional 112 (tal como una memoria caché de nivel L2 o una memoria principal). Estos elementos pueden implementarse de forma separada y pueden estar divididos como se muestra, o pueden estar integrados o repartidos en diversas combinaciones. Preferentemente, los elementos mostrados en la figura 1 forman parte de una arquitectura de microprocesador y realizan en la misma diversas acciones/funciones.

La unidad de lanzamiento de instrucciones 102 emite, preferentemente, instrucciones para su ejecución dentro del microprocesador 100. Las instrucciones pueden requerir manipulaciones de datos en coma flotante, manipulaciones de datos enteros y/o combinaciones de ellas. Por consiguiente, las instrucciones se suministran, preferentemente, a la unidad de enteros 106 y/o a la unidad de coma flotante 108 de forma que estas respectivas unidades puedan llevar a cabo las manipulaciones de enteros y de coma flotante. Los datos enteros y los datos en coma flotante que han de ser manipulados por la unidad de enteros 106 y la unidad de coma flotante 108, respectivamente, se almacenan en la memoria caché 110, en la memoria adicional 112, o en ambas.

La unidad de lectura/escritura 104 ejecuta, preferentemente, operaciones de lectura y/o escritura respecto de la unidad de enteros 106 y de la unidad de coma flotante 108, de acuerdo con la instrucción recibida de la unidad de lanzamiento de instrucciones 102. Por ejemplo, cuando la instrucción indica que debe llevarse a cabo una operación en coma flotante, la unidad de lectura/escritura 104, preferentemente, carga y/o almacena datos en coma flotante desde/en la unidad de coma flotante 108 por medio de la memoria caché 110. Hay que indicar que la dirección se suministrará utilizando datos enteros procedentes de la unidad de enteros 106. Alternativamente, cuando la instrucción indica que ha de llevarse a cabo una operación con enteros, la unidad de lectura/escritura 104, preferentemente, carga y/o almacena datos enteros desde/en la unidad de enteros 106 por medio de la memoria caché 110.

Más en concreto, cuando la unidad de lectura/escritura 104 es requerida para que obtenga datos enteros y/o en coma flotante, accede, preferentemente, a la memoria caché 110 y, si los encuentra, los datos se cargan desde la memoria caché 110 en la respectiva unidad de enteros 106 y/o unidad de coma flotante 108. Por el contrario, si no los encuentra en la memoria caché, la memoria caché 110, preferentemente, accede a la memoria adicional 112, que, como se comentó anteriormente, puede ser una memoria caché de nivel L2 o la memoria principal. En el caso de que no los encuentre en la memoria caché, los datos obtenidos de la memoria adicional 112 se vuelven a cargar en la memoria caché 110. Una vez que se lleva a cabo la manipulación de los datos en la unidad de enteros 106 y/o en la unidad de coma flotante 108, pueden ser necesarias operaciones de almacenamiento en las que la unidad de lectura/escritura 104 almacene los datos resultantes de estas operaciones en la memoria caché 110. Las particularidades asociadas a la recarga de datos y a los procesos de almacenamiento de datos se describirán más tarde.

Ahora se hace referencia a la figura 2 que ilustra un diagrama de bloques que muestra aspectos adicionales de la presente invención. Más en concreto, se proporcionan detalles adicionales que tienen que ver con la memoria caché 110 y su control. De acuerdo con la presente invención, la memoria caché 110 puede implementarse utilizando una representación libre (fully associative mapping) y/o una representación asociativa por conjuntos (set associative mapping). Para los fines de brevedad y claridad, la figura 2 muestra un único conjunto de líneas de caché 200, que puede ser parte de una implementación con representación rígida (direct mapping) y/o puede ser una "vía" de una implementación asociativa por conjuntos de N vías. En la medida en que la descripción que se haga aquí se refiera a un único conjunto de líneas de caché 200 y direccionamiento, datos, circuitos, etc. asociados, debe entenderse que dicha descripción puede ser fácilmente aplicable a una implementación asociativa por conjuntos de N vías.

Volviendo ahora a los detalles mostrados en la figura 2, la memoria caché 110, incluye, preferentemente, una pluralidad de líneas de caché 200, un controlador de memoria 202, una función o capacidad de direccionamiento 204, un selector de datos 206, y una función de comparación 208. Los datos se leen o se escriben en las líneas de caché 200 por medio del selector de datos 206, de acuerdo con un algoritmo ejecutado por el controlador de memoria 202. Las líneas de caché se acceden de acuerdo con la información de dirección recibida desde la unidad de lectura/escritura 104 y procesada de acuerdo con la función de direccionamiento. En concreto, la información de dirección incluye, preferentemente, bits de etiqueta, bits de índice y bits de localización. Los bits de etiqueta se utilizan para identificar (o nombrar) los datos deseados y comparar con un identificador almacenado dentro de una línea de caché 200 concreta. Los bits de índice proporcionan un puntero hacia una línea de caché 200 concreta dentro del conjunto, y los bits de localización identifican una localización de comienzo dentro de la parte de almacenamiento de datos de la línea de caché 200.

Cada línea de caché 200 incluye, preferentemente, una etiqueta de dirección 210, banderas de estado 212 y datos 214. La etiqueta de dirección 210 se utiliza para nombrar o identificar los datos almacenados en la respectiva línea de caché 200. Las banderas de estado 212 se utilizan para proporcionar cierta información relativa a los datos almacenados en la línea de caché 200. Las banderas de estado 212 pueden incluir una bandera de validez de datos 212A y una

ES 2 316 653 T3

bandera de tipo de datos 212B. La bandera de validez de datos 212A proporciona, preferentemente, información relativa a si los datos almacenados dentro de la línea de caché 200 son válidos o no. La bandera de validez de datos 212A puede ser un único bit de información, donde un valor lógico alto (1, cierto, activado, etc.) indica que los datos son válidos y un valor lógico bajo (0, falso, no activado, etc.) indica que los datos son inválidos (o viceversa). La
5 bandera de tipo de datos 212B incluye, preferentemente, información indicativa de si los datos almacenados en la línea de caché 200 son datos enteros o datos en coma flotante. La bandera de tipo de datos 212B puede ser un único bit de información, en donde un valor lógico alto indica que los datos son datos en coma flotante y un valor lógico bajo indica que los datos son datos enteros (o viceversa). Las banderas de estado 212 pueden incluir otra información, por ejemplo si los datos incluidos en la línea de caché 200 están “modificados” (dirty) (es decir, si los datos han sido
10 modificados y la memoria principal no ha sido actualizada, etc.).

Con referencia a las figuras 2 y 3, se realizará ahora una descripción de ciertos aspectos de la invención relativos al control de la memoria caché 110. En la acción 300, la memoria caché 110 recibe una dirección, tal como una dirección efectiva, procedente de la unidad de lectura/escritura 104. Como se ha indicado anteriormente, la dirección incluye
15 bits de etiqueta, bits de índice y bits de localización. Los bits de índice apuntan a una línea de caché concreta 200 dentro del conjunto (acción 302), en la que la línea de caché incluye una etiqueta de dirección 210, banderas de estado 212 y datos 214. En la acción 304, el controlador de memoria 202 analiza la bandera de validez de datos 212A para determinar si los datos almacenados en la línea de caché 200 son válidos. Si los datos no son válidos, entonces el flujo del proceso salta, preferentemente, a la acción 312 (que se analizará posteriormente en esta descripción). Si los datos
20 son válidos, entonces el flujo del proceso salta, preferentemente, a la acción 306.

En la acción 306, la función de comparación 208 (que puede estar integrada con, o separada del controlador de memoria 202) compara los bits de etiqueta contenidos en la dirección recibida de la unidad de lectura/escritura 104 con la etiqueta de dirección 210 de la línea de caché 200. Cuando los bits de etiqueta de la dirección solicitada no coinciden
25 con la etiqueta de dirección 210 de la línea de caché 200, entonces el flujo del proceso salta, preferentemente, a la acción 312 (que se analizará posteriormente en esta descripción). Por el contrario, cuando los bits de etiqueta de la dirección solicitada coinciden con la etiqueta de dirección 210 de la línea de caché 200, entonces el flujo del proceso salta, preferentemente, a la acción 308.

En la acción 308 se hace una pregunta sobre si la unidad de lectura/escritura 104 está dando servicio a una instrucción de lectura o a una instrucción de escritura. Si la unidad de lectura/escritura 104 está dando servicio a una instrucción de lectura (es decir, los datos se obtienen de la memoria caché 110 para su entrega a la unidad de enteros 106 o a la unidad de coma flotante 108), entonces se obtiene un acierto de caché y el flujo del proceso avanza
30 hasta la acción 310 en la que los datos se cargan desde la línea de caché 200 en la unidad de lectura/escritura 104 a través del selector de datos 206. Si, en la acción 308, se determina que la unidad de lectura/escritura está dando servicio a una instrucción de escritura, entonces se sigue obteniendo un acierto de caché, pero el flujo del proceso salta, preferentemente a la acción 312, que se discutirá a continuación.

Hay que indicar que cuando el flujo del proceso salta a la acción 312 desde la acción 304 o desde la acción 306 y la unidad de lectura/escritura 104 está dando servicio a una instrucción de lectura, entonces hay que realizar una recarga de datos (es decir, los datos solicitados por la unidad de lectura/escritura no están contenidos en la memoria caché 110). En este caso, los datos solicitados se obtienen de la memoria adicional 112 (que, como ya se ha comentado, puede ser una memoria caché de nivel L2 o la memoria principal). Cuando el flujo del proceso salta a la acción 312 desde la
40 acción 308, la unidad de lectura/escritura 104 está dando servicio a una instrucción de escritura, en la que los datos se están escribiendo en la memoria caché 110, por ejemplo, como resultado de una operación llevada a cabo por la unidad de enteros 106 o por la unidad de coma flotante 108. Con independencia de si se está llevando a cabo una operación de recarga o de escritura en esta etapa del proceso, el controlador de la memoria 202 determina, preferentemente, si se permite o se prohíbe una sobreescritura de los datos en una línea de caché 200 en base a la bandera de validez de datos 212A, a la bandera de tipo de datos 212B y/o el tipo de datos (es decir, entero o en coma flotante) que ha de almacenarse en la línea de caché 200.
50

Para poder hacer la discusión, el algoritmo básico para controlar la sobreescritura de datos en la línea de caché 200 se tabula en la Tabla 1. Hay que indicar que el algoritmo de la Tabla 1 puede ilustrarse en numerosos diagramas de flujo del proceso diferentes. El ilustrar todos estos diagramas de flujo alternativos iría en contra de la claridad y de la
55 brevedad y, por consiguiente, en la figura 3 sólo se muestra un ejemplo de tales diagramas de flujo del proceso.

ES 2 316 653 T3

TABLA 1

5	BANDERA DE VALIDEZ DE DATOS	BANDERA DE TIPO DE DATOS	TIPO DE DATOS	CONTROL DE LA MEMORIA
10	0	X	Enteros	Sobreescribir
	0	X	En coma flotante	Sobreescribir
15	1	0	Enteros	Sobreescribir
	1	0	En coma flotante	Prohibir sobreescritura
	1	1	Enteros	Sobreescribir
20	1	1	En coma flotante	Sobreescribir

Refiriéndonos ahora a las figuras 2 y 3, y a la Tabla 1, en la acción 312 el controlador de memoria 202 lleva a cabo, preferentemente, una determinación del tipo de datos a almacenar en la línea de caché 200. A modo de ejemplo, información referente al tipo de datos a almacenar puede obtenerse de la unidad de lectura/escritura 104 porque la instrucción recibida por la unidad de lectura/escritura (figura 1) incluye, preferentemente, dicha información. Cuando los datos a almacenar en la línea de caché 200 son datos en coma flotante, entonces el flujo del proceso avanza, preferentemente, hasta la acción 316 (que se analizará posteriormente). Cuando los datos a almacenar en la línea de caché 200 no son datos en coma flotante (es decir, los datos son datos enteros), entonces el flujo del proceso avanza, preferentemente, hasta la acción 314, en la que cualquier dato de la línea de caché 200 es sobrescrito con los datos enteros. Esto corresponde con las líneas 1, 3 y 5 de la Tabla 1. En concreto, las líneas 1, 3 y 5 de la Tabla 1 indican que debe permitirse una sobreescritura independientemente del estado de la bandera de validez de datos 212A y de la bandera de tipo de datos 212B, siempre que el tipo de datos a almacenar en la línea de caché 200 sea de datos enteros. Volviendo de nuevo a la acción 314 de la figura 3, cuando la línea de caché 200 se sobrescribe con datos enteros, la bandera de validez de datos 212A se pone, preferentemente, por ejemplo a un valor lógico alto (estado cierto) para indicar la validez de los datos y la bandera de tipo de datos 212B se pone, preferentemente, por ejemplo a un valor lógico bajo (estado falso) para indicar datos enteros.

Cuando se ha determinado que los datos a almacenar en la línea de caché 200 son datos en coma flotante (acción 312 afirmativa), entonces el proceso salta, preferentemente, a la acción 316, donde se lleva a cabo una determinación de si la bandera de tipo de datos 212B indica que los datos que se han almacenado previamente en la línea de caché 200 (es decir, los datos que están almacenados en ese momento en la línea de caché 200) son datos en coma flotante. Si la respuesta es negativa, entonces el flujo del proceso avanza, preferentemente, a la acción 320 (que se analizará más tarde). Si esta determinación es afirmativa, entonces el flujo del proceso avanza, preferentemente, a la acción 318, donde se permite la sobreescritura de los datos en la línea de caché 200. Esto corresponde a la línea 6 de la Tabla 1, en la que la bandera de validez de datos 212A indica que los datos almacenados en ese momento en la línea de caché 200 son válidos (aunque esto no es determinante), la bandera de tipo de datos 212B indica que el tipo de datos almacenados en ese momento en la línea de caché son datos en coma flotante, y el tipo de datos a escribir en la línea de caché 200 son también datos en coma flotante. Esto corresponde también a la línea 2 de la tabla 1, en la que la bandera de validez de datos 212A indica que los datos almacenados en ese momento en la línea de caché 200 son no válidos (aunque esto no es determinante), la bandera de tipo de datos 212B indica que el tipo de datos almacenados en ese momento en la línea de caché son datos en coma flotante, y el tipo de datos a escribir en la línea de caché 200 son datos en coma flotante.

Si, en la acción 316, se determina que la bandera de tipo de datos 212B indica que los datos almacenados en ese momento en la línea de caché 200 son datos enteros, entonces el proceso pasa, preferentemente, a la acción 320 en la que se determina si los datos enteros almacenados en ese momento en la línea de caché 200 son válidos. (Hay que indicar que esta determinación de validez puede haber sido hecha previamente y la información guardada en la acción 304 o puede determinarse de nuevo). Cuando dichos enteros no son válidos, por ejemplo la bandera de validez de datos 212A tiene un valor lógico bajo (estado falso), entonces el flujo del proceso avanza, preferentemente, a la acción 318 donde, como se ha indicado anteriormente, se permite la sobreescritura de los datos en coma flotante en la línea de caché 200. Esto corresponde a la línea 1 de la Tabla 1, en la que la bandera de validez de datos 212A indica que los datos almacenados en ese momento en la línea de caché 200 son no válidos y la bandera de tipo de datos 212B indica que dichos datos son datos enteros. Volviendo de nuevo a la figura 3, el controlador de memoria 202 fija, preferentemente, la bandera de validez de datos 212A de la línea de caché 200, por ejemplo a un valor lógico alto, estado verdadero. Además, el controlador de memoria 202, fija, preferentemente, la bandera de tipo de datos 212B para que indique datos en coma flotante, por ejemplo a un valor lógico alto, estado verdadero.

ES 2 316 653 T3

Si la respuesta en la acción 320 es afirmativa, es decir que los datos enteros almacenados en ese momento en la línea de caché 200 son válidos, entonces el flujo del proceso avanza, preferentemente, a la acción 322, en la que se prohíbe la sobreescritura de los datos en coma flotante en la línea de caché 200. Esto corresponde a la línea 4 de la Tabla 1 en la que la bandera de validez de datos 212A indica que los datos almacenados en ese momento en la línea de caché 200 son válidos, la bandera de tipo de datos 212B indica que dichos datos son datos enteros, y el tipo de datos a almacenar en la línea de caché 200 son datos en coma flotante. De forma ventajosa, el controlador de memoria 202 evita escribir datos en coma flotante sobre datos enteros válidos en la memoria caché 110.

Si se prohíbe que dichos datos en coma flotante se escriban sobre datos enteros válidos de la línea de caché 200, entonces dichos datos se transfieren, preferentemente, a la unidad de lectura/escritura 104 o se almacenan en una memoria caché alternativa (acción 324).

Ahora se hace referencia a la figura 4, que es un diagrama de bloques que muestra aspectos adicionales de la presente invención. En concreto, la figura 4 muestra una memoria caché alternativa 110A, que es sustancialmente similar a la memoria caché 110 mostrada en la figura 2 con la excepción de que incluye la capacidad de transferir datos en coma flotante a una memoria caché alternativa 400. La memoria caché alternativa 400 puede estar integrada con la memoria caché 110A o puede estar separada de la misma. Hay que indicar que la memoria caché alternativa 400 actúa como una extensión del conjunto de líneas de caché 200. De acuerdo con esto, se incluye una función adicional de comparación 208A para determinar si los bits de etiqueta de la dirección solicitada coinciden con la etiqueta de dirección contenida en una línea de caché concreta 200 de la memoria caché alternativa 400. Hay que indicar que la memoria caché alternativa 400 puede incluir una única línea de caché o múltiples líneas de caché.

Como se ha comentado anteriormente, los diversos aspectos de los métodos y aparatos de la presente invención pueden aplicarse a una memoria caché de representación rígida y un conjunto único, o puede utilizarse en la implementación de una memoria caché asociativa por conjuntos de N vías.

Con referencia a la figura 3, en el caso de una implementación de memoria caché asociativa de conjuntos de N vías, en la acción 302 se accede a todas las líneas de caché 200 asociadas con los bits de índice dados y la determinación de cuáles de estas líneas de caché 200 son válidas se realiza en base a las respectivas banderas de validez de datos 212A. El controlador de memoria 202 permite, preferentemente, sobrescribir datos en coma flotante o datos enteros en cualquiera de las líneas de caché 200 en las que las correspondientes banderas de validez de datos 212A indiquen que dichas líneas de caché 200 no contienen datos válidos. Cuando todas las líneas de caché 200 contienen datos válidos (es decir, todas las banderas de validez de datos 212A asociadas están en estado cierto), cualquier línea de caché 200 puede ser sobreescrita si los datos a almacenar son datos enteros. Si los datos a almacenar son datos en coma flotante, entonces dichos datos sólo pueden almacenarse en una línea de caché 200 en la que la bandera de tipo de datos 212B indique que los datos almacenados en ese momento en dicha línea de caché 200 son datos en coma flotante. En el caso de que todas las líneas de caché 200 asociadas con los bits de índice contengan datos válidos (es decir, todas las banderas de validez de datos 212A están en estado cierto), y todas las banderas de tipo de datos 212B asociadas indiquen que los datos almacenados son datos enteros, entonces el controlador de memoria 202 prohíbe, preferentemente, la sobreescritura de los datos en coma flotante en cualquier línea de caché 200.

De forma ventajosa, los métodos y aparatos para controlar la memoria caché que se han descrito anteriormente aumentan el uso de una memoria caché de nivel L1 en un sistema de microprocesamiento que maneja datos enteros y en coma flotante. En consecuencia, se obtienen mejoras en la eficiencia de la memoria, capacidad de procesamiento y calidad del procesamiento de datos.

REIVINDICACIONES

1. Un método para controlar una memoria caché (110, 110A), que comprende los pasos de:

5 recibir (300) una dirección (204) para al menos una entre las opciones de escribir datos en y de recuperar datos de la memoria caché, incluyendo la dirección bits de etiqueta y bits de índice;

acceder (302) a una o más líneas de caché (200) de la memoria caché correspondientes a los bits de índice de la dirección; incluyendo cada línea de caché una etiqueta de dirección (210), una bandera de validez de datos (212A) y
10 una bandera de tipo de datos (212B);

determinar (304) si los datos en al menos una de la una o más líneas de caché son válidos en base a la bandera de validez de datos;

15 determinar (312) qué tipo de datos se han almacenado en la al menos una línea de caché en base a la bandera de tipo de datos; y

prohibir (322) la sobreescritura de datos en coma flotante en la al menos una línea de caché cuando los datos en la misma sean válidos y los datos que se han almacenado en la al menos una línea de caché sean datos enteros.
20

2. El método de la reivindicación 1, que comprende además los pasos de: (i) fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos en la misma son válidos; y (ii) fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos en la al menos una línea de caché son de un tipo entero, cuando se prohíba la sobreescritura de datos en coma flotante en la al menos una
25 línea de caché.

3. El método de la reivindicación 1, en el que al menos una de las banderas de validez de datos y de tipo de datos es una bandera de un único bit.

30 4. El método de la reivindicación 1, que comprende además los pasos de:

permitir la sobreescritura de datos enteros en la al menos una línea de caché cuando la bandera de validez de datos indique que la al menos una línea de caché no contiene datos válidos;

35 fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos enteros son válidos; y

fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.
40

5. El método de la reivindicación 1, que comprende además los pasos de:

permitir la sobreescritura de datos en coma flotante en la al menos una línea de caché cuando la bandera de validez de datos indique que la al menos una línea de caché no contiene datos válidos;

45 fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos en coma flotante son válidos; y

fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos en coma flotante son de un tipo de coma flotante.
50

6. El método de la reivindicación 1, que comprende además el paso de permitir la sobreescritura de datos enteros en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos en la al menos una línea de caché son de un tipo entero.
55

7. El método de la reivindicación 6, que comprende además al menos uno de los pasos de:

fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos enteros son válidos; y
60

fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

8. El método de la reivindicación 1, que comprende además el paso de permitir la sobreescritura de datos enteros en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos en la al menos una línea de caché son de un tipo en coma flotante.
65

ES 2 316 653 T3

9. El método de la reivindicación 8, que comprende además al menos uno de los pasos de:

fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos enteros son válidos; y

fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

10. El método de la reivindicación 1, que comprende además el paso de permitir la sobreescritura de datos en coma flotante en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos en la al menos una línea de caché son de un tipo de coma flotante.

11. El método de la reivindicación 10, que comprende además al menos uno de los pasos de:

fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos en coma flotante son válidos; y

fijar el valor de la bandera de tipo de datos de la al menos una línea de caché para que indique que los datos en coma flotante son de un tipo de coma flotante.

12. El método de la reivindicación 1, que comprende además el paso de transferir los datos en coma flotante a una unidad de lectura/escritura u otra memoria caché (400) cuando se prohíba la sobreescritura de los datos en coma flotante en la al menos una línea de caché.

13. El método de la reivindicación 12, en el que la otra memoria caché incluye solamente una línea de caché.

14. El método de la reivindicación 1, que comprende además los pasos de:

acceder a todas las líneas de caché de la memoria caché correspondientes a los bits de índice de la dirección;

determinar cuáles de las líneas de caché son válidas en base a la bandera de validez de datos; y

permitir la sobreescritura de datos en coma flotante o de datos enteros en cualquiera de las líneas de caché en las que la correspondiente bandera de validez de datos indique que dichas líneas de caché no contienen datos válidos.

15. El método de la reivindicación 1, que comprende además los pasos de:

acceder a todas las líneas de caché de la memoria caché correspondientes a los bits de índice de la dirección;

determinar cuáles de las líneas de caché son válidas en base a la bandera de validez de datos; y

permitir la sobreescritura de datos enteros en cualquiera de las líneas de caché cuando todas las banderas de validez de datos indiquen que dichas líneas de caché contienen datos válidos.

16. El método de la reivindicación 15, que comprende además los pasos de:

determinar qué tipo de datos se han almacenado en cada una de las líneas de caché en base a las respectivas banderas de tipo de datos; y

permitir la sobreescritura de datos en coma flotante en cualquiera de las líneas de caché en las que las correspondientes banderas de tipo de datos indiquen que el tipo de datos que se han almacenado en dichas líneas de caché son datos en coma flotante cuando todas las banderas de validez de datos indiquen que las líneas de caché contienen datos válidos.

17. El método de la reivindicación 16, que comprende además el paso de prohibir la sobreescritura de datos en coma flotante en cualquiera de las líneas de caché cuando todas las banderas de validez de datos indiquen que las líneas de caché contienen datos válidos y todas las banderas de tipo de datos indiquen que las líneas de caché contienen datos enteros.

18. El método de la reivindicación 17, que comprende además el paso de transferir los datos en coma flotante a una unidad de lectura/escritura u otra memoria de caché cuando se prohíba la sobreescritura de los datos en coma flotante en cualquiera de las líneas de caché.

19. El método de la reivindicación 18, en el que la otra memoria caché incluye solamente una línea de caché.

ES 2 316 653 T3

20. Un aparato para controlar una memoria caché (110, 110A), que comprende:

una memoria de dirección (204) capaz de recibir una dirección para al menos una entre las opciones de escribir datos en y de recuperar datos de la memoria caché, incluyendo la dirección bits de etiqueta y bits de índice; y

un controlador de memoria (202) capaz de: (i) acceder a una o más líneas de caché (200) de la memoria caché correspondientes a los bits de índice de la dirección, incluyendo cada línea de caché una etiqueta de dirección (210), una bandera de validez de datos (212A) y una bandera de tipo de datos (212B); (ii) determinar si los datos en al menos una de la una o más líneas de caché son válidos en base a la bandera de validez de datos; (iii) determinar qué tipo de datos se han almacenado en la al menos una línea de caché en base a la bandera de tipo de datos; y (iv) prohibir la sobrescritura de datos en coma flotante en la al menos una línea de caché cuando los datos en la misma sean válidos y los datos que se han almacenado en la al menos una línea de caché sean datos enteros.

21. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de: (i) fijar el valor de la bandera de validez de datos de la al menos una línea de caché para que indique que los datos en la misma son válidos; y (ii) fijar el valor de la bandera de tipo de datos para la al menos una línea de caché para que indique que los datos en la al menos una línea de caché son de un tipo entero, cuando se prohíba la sobrescritura de datos en coma flotante en la al menos una línea de caché.

22. El aparato de la reivindicación 20, en el que al menos una de las banderas de validez de datos y de tipo de datos es una bandera de un único bit.

23. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de: (i) permitir la sobrescritura de datos enteros en la al menos una línea de caché cuando la bandera de validez de datos indique que la al menos una línea de caché no contiene datos válidos; (ii) fijar el valor de la bandera de validez de datos para la al menos una línea de caché para que indique que los datos enteros son válidos; y (iii) fijar el valor de la bandera de tipo de datos para la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

24. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de: (i) permitir la sobrescritura de datos en coma flotante en la al menos una línea de caché cuando la bandera de validez de datos indique que la al menos una línea de caché no contiene datos válidos; (ii) fijar el valor de la bandera de validez de datos para la al menos una línea de caché para que indique que los datos en coma flotante son válidos; y (iii) fijar el valor de la bandera de tipo de datos para la al menos una línea de caché para que indique que los datos en coma flotante son de un tipo de coma flotante.

25. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de permitir la sobrescritura de datos enteros en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos de la al menos una línea de caché son de un tipo entero.

26. El aparato de la reivindicación 25, en el que el controlador de memoria es capaz adicionalmente de al menos una entre: (i) fijar el valor de la bandera de validez de datos para la al menos una línea de caché para que indique que los datos enteros son válidos; y (ii) fijar el valor de la bandera de tipo de datos para la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

27. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de permitir la sobrescritura de datos enteros en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos de la al menos una línea de caché son de un tipo de coma flotante.

28. El aparato de la reivindicación 27, en el que el controlador de memoria es capaz adicionalmente de al menos una entre: (i) fijar el valor de la bandera de validez de datos para la al menos una línea de caché para que indique que los datos enteros son válidos; y (ii) fijar el valor de la bandera del tipo de datos para la al menos una línea de caché para que indique que los datos enteros son de un tipo entero.

29. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de permitir la sobrescritura de datos en coma flotante en la al menos una línea de caché cuando: (i) la bandera de validez de datos indique que la al menos una línea de caché contiene datos válidos, y (ii) la bandera de tipo de datos indique que los datos de la al menos una línea de caché son de un tipo de coma flotante.

30. El aparato de la reivindicación 29, en el que el controlador de memoria es capaz adicionalmente de al menos una entre: (i) fijar el valor de la bandera de validez de datos para la al menos una línea de caché para que indique que los datos en coma flotante son válidos; y (ii) fijar el valor de la bandera de tipo de datos para la al menos una línea de caché para que indique que los datos en coma flotante son de un tipo de coma flotante.

31. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de transferir los datos en coma flotante a una unidad de lectura/escritura u otra memoria caché (400) cuando se prohíba la sobrescritura de los datos en coma flotante en la al menos una línea de caché.

ES 2 316 653 T3

32. El aparato de la reivindicación 31, en el que la otra memoria caché incluye solamente una línea de caché.

33. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de: (i) acceder a todas las líneas de caché de la memoria caché correspondientes a los bits de índice de la dirección; (ii) determinar
5 cuáles de las líneas de caché son válidas en base a las banderas de validez de datos; y (iii) permitir la sobrescritura de datos en coma flotante o de datos enteros en cualquiera de las líneas de caché en las que la correspondiente bandera de validez de datos indique que dichas líneas de caché no contienen datos válidos.

34. El aparato de la reivindicación 20, en el que el controlador de memoria es capaz adicionalmente de: (i) acceder
10 a todas las líneas de caché de la memoria caché correspondientes a los bits de índice de la dirección; (ii) determinar cuáles de las líneas de caché son válidas en base a las banderas de validez de datos; y (iii) permitir la sobrescritura de datos enteros en cualquiera de las líneas de caché cuando todas las banderas de validez de datos indiquen que dichas líneas de caché contienen datos válidos.

35. El aparato de la reivindicación 34, en el que el controlador de memoria es capaz adicionalmente de (i) deter-
15 minar qué tipo de datos se han almacenado en cada una de las líneas de caché en base a las respectivas banderas de tipo de datos; y (ii) permitir la sobrescritura de datos en coma flotante en cualquiera de las líneas de caché en las que las correspondientes banderas de tipo de datos indiquen que el tipo de datos que se han almacenado en dichas líneas de caché son datos en coma flotante cuando todas las banderas de validez de datos indiquen que las líneas de caché
20 contienen datos válidos.

36. El aparato de la reivindicación 35, en el que el controlador de memoria es capaz adicionalmente de prohibir la sobrescritura de datos en coma flotante en cualquiera de las líneas de caché cuando todas las banderas de validez de
25 datos indiquen que las líneas de caché contienen datos válidos y todas las banderas de tipo de datos indiquen que las líneas de caché contienen datos enteros.

37. El aparato de la reivindicación 36, en el que el controlador de memoria es capaz adicionalmente de transferir
30 los datos en coma flotante a una unidad de lectura/escritura u otra memoria caché cuando se prohíba la sobrescritura de los datos en coma flotante en cualquiera de las líneas de caché.

38. El aparato de la reivindicación 37, en el que la otra memoria caché incluye solamente una línea de caché.

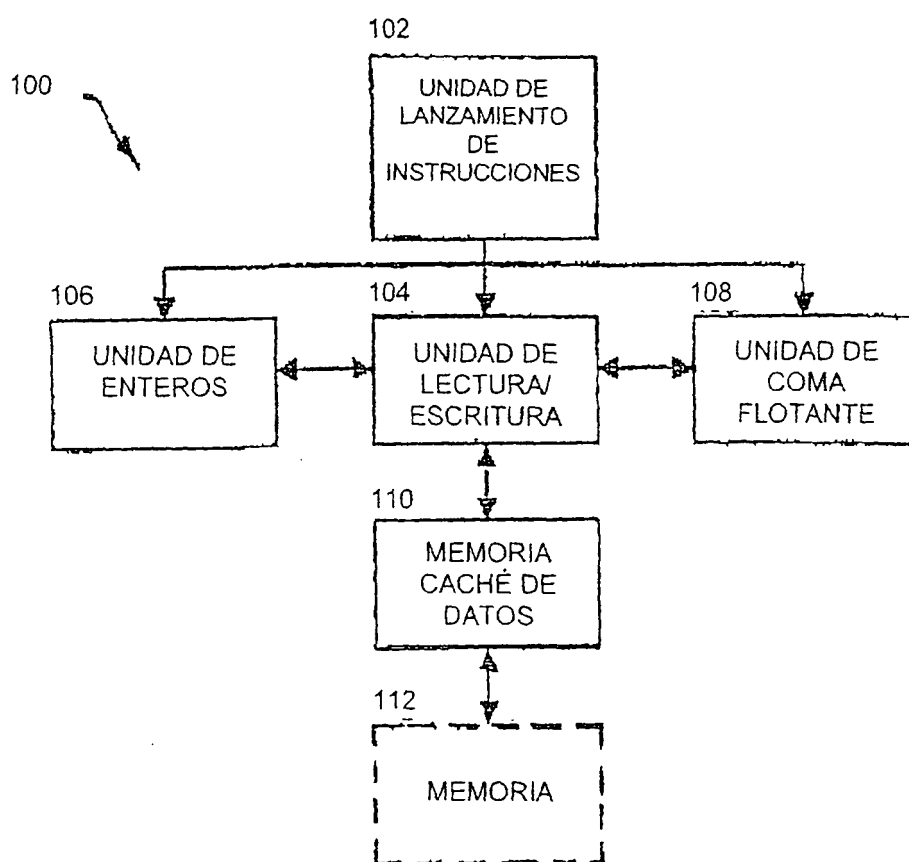


Fig. 1

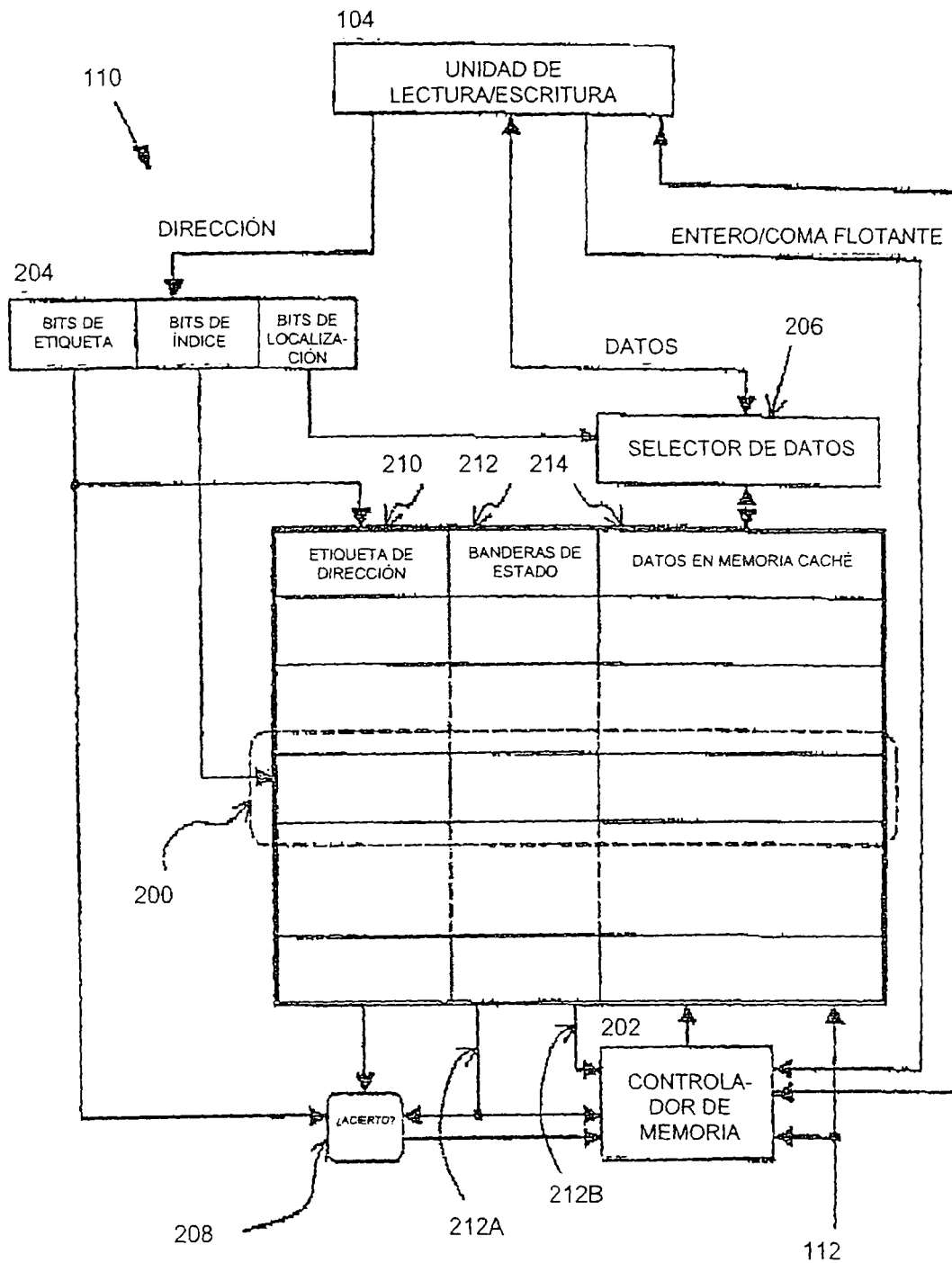


Fig. 2

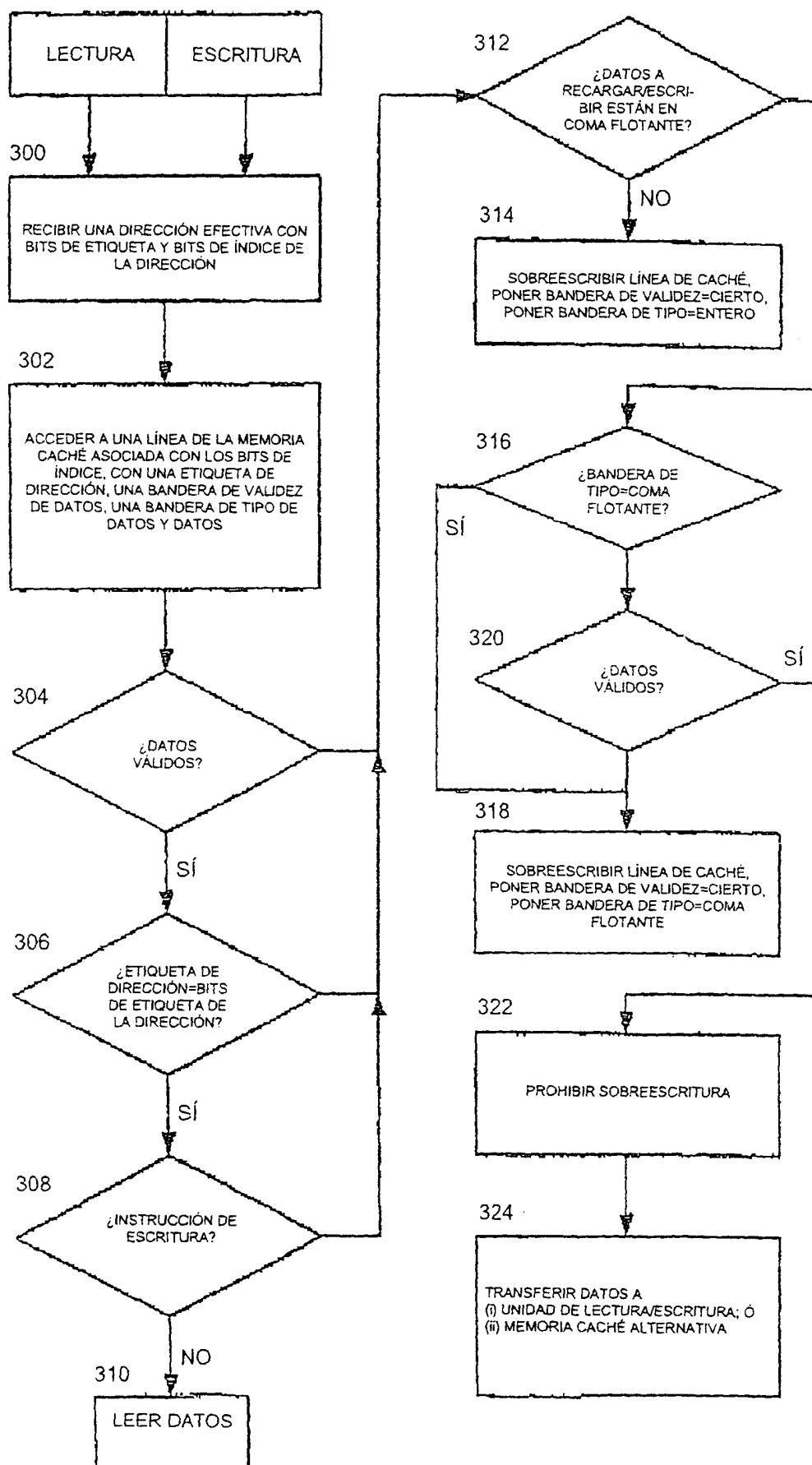


Fig. 3

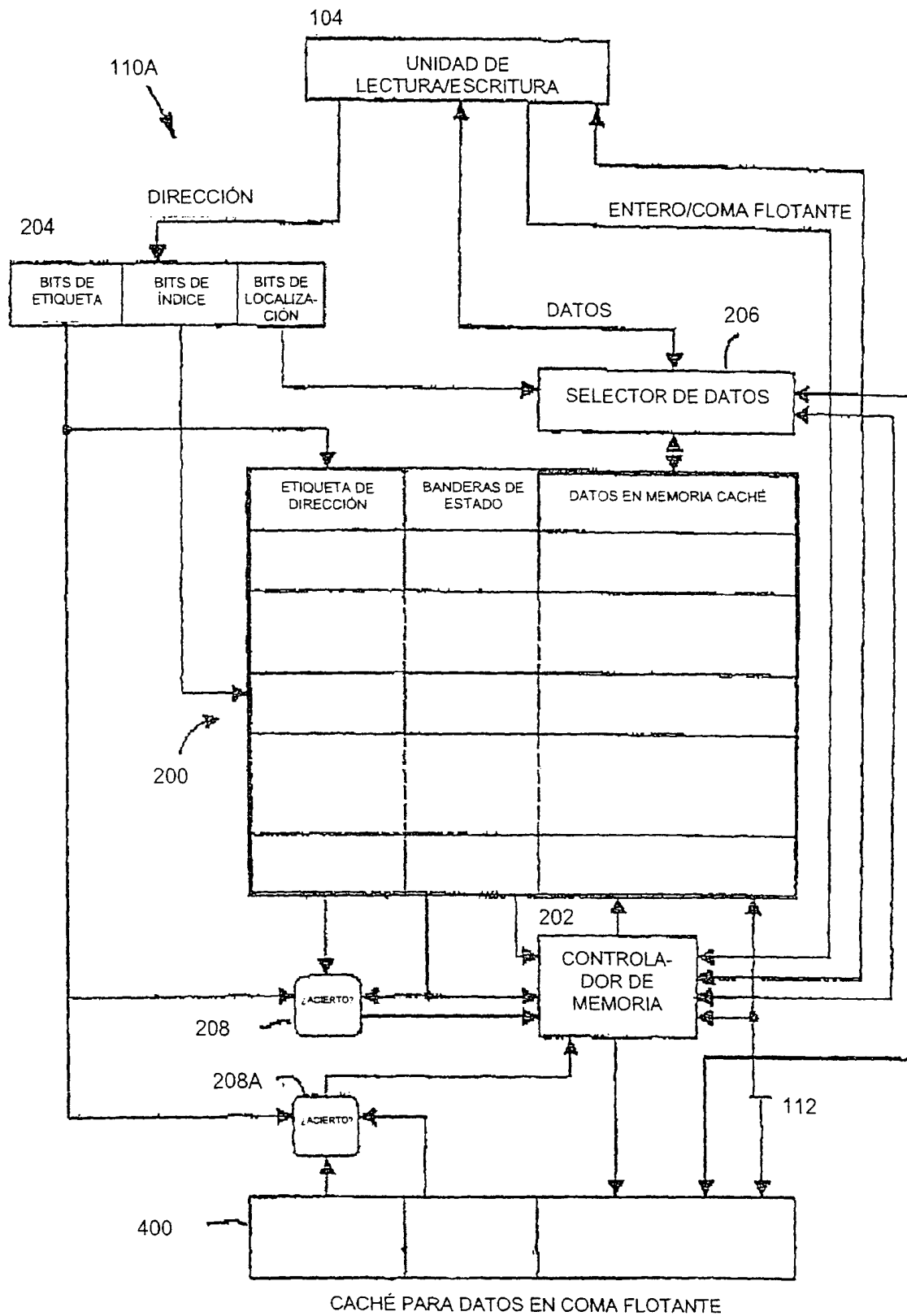


Fig. 4