

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 383 588**

51 Int. Cl.:

G11C 7/16 (2006.01)

G11C 11/56 (2006.01)

G11C 16/10 (2006.01)

G11C 16/26 (2006.01)

G11C 16/34 (2006.01)

G11C 16/04 (2006.01)

G06F 11/00 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **10164179 .3**

96 Fecha de presentación: **14.05.2007**

97 Número de publicación de la solicitud: **2221823**

97 Fecha de publicación de la solicitud: **25.08.2010**

54 Título: **Operaciones de mantenimiento para celdas de almacenamiento de datos de múltiples niveles**

30 Prioridad:
15.05.2006 US 800357 P
30.03.2007 US 694739

45 Fecha de publicación de la mención BOPI:
22.06.2012

45 Fecha de la publicación del folleto de la patente:
22.06.2012

73 Titular/es:
APPLE INC.
1 INFINITE LOOP
CUPERTINO, CA 95014, US

72 Inventor/es:
Cornwell, Michael J. y
Dudte, Christopher P.

74 Agente/Representante:
Fàbrega Sabaté, Xavier

ES 2 383 588 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 de mantenimiento para celdas de almacenamiento de datos de múltiples niveles

5 **Campo técnico**

Varias implementaciones pueden referirse, en general, a dispositivos de memoria no volátiles, e implementaciones particulares pueden referirse a sistemas y a procedimientos para hacer funcionar celdas flash de múltiples niveles.

10 **Antecedentes**

A medida que los dispositivos informáticos han aumentado sus capacidades y prestaciones, ha crecido la demanda de dispositivos de almacenamiento de datos. Los dispositivos de almacenamiento de datos se han utilizado, por ejemplo, para almacenar instrucciones de programa (es decir, código) que puede ejecutarse por procesadores. Los dispositivos de almacenamiento de datos también se han utilizado para almacenar otros tipos de datos, incluyendo información de audio, de imagen y/o de texto, por ejemplo. Recientemente, los sistemas con dispositivos de almacenamiento de datos capaces de almacenar una gran cantidad de contenido de datos (por ejemplo, canciones, vídeos musicales, etc.) se han incorporado de manera generalizada en dispositivos portátiles.

Tales dispositivos portátiles incluyen dispositivos de almacenamiento de datos (DSD) compactos y que pueden hacerse funcionar mediante fuentes de alimentación portátiles, tales como baterías. Algunos DSD de dispositivos portátiles pueden proporcionar memoria no volátil que puede conservar los datos cuando se desconecta de la fuente de alimentación. Los dispositivos portátiles han utilizado varios dispositivos de almacenamiento de datos no volátiles, tales como unidades de disco duro, EEPROM (memoria de solo lectura programable que puede borrarse eléctricamente) y memoria flash.

La memoria flash se ha convertido en un tipo de DSD ampliamente utilizado. La memoria flash puede proporcionar una memoria no volátil en dispositivos electrónicos portátiles y en aplicaciones de consumidor, por ejemplo. Dos tipos de memoria flash son la flash NOR y la flash NAND. La flash NOR proporciona normalmente la capacidad de ejecutar código in situ y es accesible de manera aleatoria (es decir, como una RAM). Normalmente, la flash NAND puede borrar datos más rápidamente, acceder a datos en ráfagas (por ejemplo, bloques de 512 bytes) y puede proporcionar ciclos de borrado de mayor duración en comparación con la flash NOR. La flash NAND puede proporcionar generalmente un almacenamiento no volátil a un bajo coste por bit como un medio de almacenamiento de archivos de alta densidad para dispositivos de consumidores, tales como cámaras digitales y reproductores MP3, por ejemplo.

La memoria flash típica almacena una unidad de información almacenando una carga eléctrica en cada celda de memoria a una tensión representativa de un valor de datos digital. Las celdas de un solo nivel almacenan un bit de información en función de si la celda está cargándose a una tensión "alta" o está descargándose a una tensión "baja". La memoria flash NAND se ha desarrollado de tal manera que almacena hasta dos bits de información en una única celda descodificando la carga considerando que está dentro de uno de cuatro intervalos de tensión diferentes. La memoria flash NOR se ha desarrollado de tal manera que puede almacenar hasta 8 bits de información en una única celda descodificando la carga considerando que está dentro uno de 256 intervalos de tensión diferentes.

La solicitud de patente estadounidense US 2005/052934 divulga un sistema para el almacenamiento de datos según el preámbulo de la reivindicación 7.

La solicitud de patente estadounidense US 2005/0007801 divulga un sistema de memoria flash no volátil que normalmente hace funcionar sus celdas de memoria en múltiples estados de almacenamiento, pero tiene la capacidad de hacer funcionar todos o algunos de sus bloques de celda de memoria en dos estados. Los dos estados se seleccionan para que sean los más distantes de los múltiples estados, proporcionando de ese modo un mayor margen durante dos operaciones de estado.

La solicitud de patente europea EP 1182669 divulga un dispositivo de memoria no volátil que incluye una sección de comparación de datos para proporcionar resultados de comparación obtenidos comparando datos de las celdas de memoria con datos de elementos de referencia para operaciones de lectura, escritura y borrado, y los datos vuelven a escribirse en caso de una condición de baja energía.

60 **Resumen**

Los aparatos y sistemas asociados, los procedimientos y los productos de programa informático descritos se

refieren al almacenamiento de datos en múltiples niveles en dispositivos de memoria flash.

Según un primer aspecto de la presente invención, se proporciona un artículo de fabricación que comprende instrucciones legibles por máquina que, cuando se ejecutan, hacen que se lleven a cabo operaciones, caracterizado porque las operaciones comprenden: determinar si llevar a cabo una operación de mantenimiento donde la determinación de si llevar a cabo la operación de mantenimiento comprende determinar si un dispositivo principal tiene una fuente de alimentación (1210) que satisface una condición predeterminada; identificar información de error asociada con una página de celdas de memoria (121) en respuesta a determinar que debe llevarse a cabo una operación de mantenimiento; determinar si la información de error satisface un criterio de error; y en respuesta a determinar que la fuente de alimentación satisface la condición predeterminada, ajustar uno o más registros de resolución (166) correspondientes a la página de celdas de memoria (121) desde una primera resolución hasta una segunda resolución, definiendo cada una de la primera y la segunda resolución una pluralidad de intervalos de tensión, correspondiendo cada intervalo de tensión a un valor de datos posible, presentando la primera resolución más intervalos de tensión que la segunda resolución.

Las implementaciones pueden incluir una o más de las siguientes características. La determinación de si la fuente de alimentación satisface la condición predeterminada puede ser que el dispositivo principal esté recibiendo energía de CA o que el dispositivo principal tenga una batería cargada a un nivel de carga predeterminado. La determinación de si llevar a cabo la operación de mantenimiento puede incluir determinar si un procesador tiene un ancho de banda no utilizado que supera un umbral predeterminado.

Según un segundo aspecto de la presente invención, se proporciona un sistema para el almacenamiento de datos, caracterizado porque el sistema comprende: una pluralidad de celdas de memoria (121), estado adaptada cada celda de memoria para recibir carga durante una operación de escritura a un nivel de tensión correspondiente a un valor de datos que presenta un número especificado de bits; un registro de resolución (166) asociado con la pluralidad de celdas de memoria (121), incluyendo el registro de resolución entradas que indican cada una un número de bits almacenados en una o más celdas de memoria correspondientes; una interfaz principal (109) adaptada para recibir señales de un dispositivo principal que indican una condición de fuente de alimentación para el dispositivo principal; y un procesador (112) adaptado para reescribir valores de datos en la pluralidad de celdas de memoria (121) y para ajustar los registros de resolución (166) para pasar de indicar un primer número de bits a indicar un segundo número de bits en respuesta a una señal que indica una condición de fuente de alimentación predeterminada.

Las implementaciones pueden incluir una o más de las siguientes características. Un código software de direccionamiento lógico convierte direcciones lógicas recibidas desde el dispositivo principal en direcciones físicas que se utilizan para el acceso a los datos. La interfaz principal está adaptada además para recibir comandos desde un dispositivo principal y para intercambiar datos con el dispositivo principal.

Algunas implementaciones pueden proporcionar una o más ventajas. Por ejemplo, algunas implementaciones pueden proporcionar funciones de almacenamiento de datos a gran rendimiento. La densidad y/o la capacidad de almacenamiento pueden aumentar. Algunos ejemplos pueden proporcionar una mayor fiabilidad y/o menores tasas de error de datos. Varias implementaciones pueden permitir mayores niveles de integración, miniaturización, menos ruido electromagnético y/o mejores márgenes de ruido. Algunas implementaciones permiten menores costes de sistema en sistemas auxiliares, tales como suministros de tensión a circuitos lógicos y/o circuitos de programación/borrado.

Los detalles de una o más implementaciones de la invención se exponen en los dibujos adjuntos y en la siguiente descripción. Otras características de la invención resultarán evidentes a partir de la descripción, los dibujos y las reivindicaciones.

Descripción de los dibujos

La FIG. 1 muestra un ejemplo de una arquitectura de una cápsula de múltiples chips que incluye un chip de memoria flash NAND y un controlador de disco flash.

Las FIG. 2A y 2B muestran de manera conjunta un mapeo entre tensiones de celda y valores digitales almacenados en una celda de memoria.

Las FIG. 3A y 3B muestran diagramas de flujo que ilustran ejemplos de procesos para leer páginas de datos de una memoria flash con celdas de múltiples niveles.

La FIG. 4 muestra un diagrama de flujo que ilustra un ejemplo de un proceso para leer una página de datos de

memoria flash.

La FIG. 5 muestra un diagrama de flujo que ilustra un ejemplo de un proceso para llevar a cabo operaciones de corrección de errores para corregir una página de datos que contiene errores de bits.

5 Las FIG. 6A a 6C muestran de manera conjunta ejemplos de operaciones para ejecutar un comando de valor alternativo.

10 Las FIG. 7A y 7B muestran diagramas de flujo que ilustran un ejemplo de procesos para escribir datos en una página de memoria flash.

Las FIG. 8A y 8B muestran diagramas de flujo que ilustran ejemplos de procesos para ajustar la resolución de celda de una página de memoria.

15 La FIG. 9 muestra un diagrama de flujo que ilustra un ejemplo de un proceso de mantenimiento.

La FIG. 10 muestra un diagrama de flujo que ilustra un ejemplo de un proceso de direccionamiento lógico en el controlador de disco flash.

20 La FIG. 11 muestra un ejemplo de un sistema que incluye una bomba de carga y un convertidor de analógico a digital externos al chip de memoria flash NAND.

La FIG. 12 muestra un ejemplo de un sistema que incluye una entrada de energía desacoplada en un chip de memoria flash NAND.

25 Los símbolos de referencia similares en los diferentes dibujos indican elementos similares.

Descripción detallada de ejemplos ilustrativos

30 Varias implementaciones se refieren a una memoria flash que puede almacenar información en celdas profundas de múltiples niveles (MLC). Las celdas profundas de múltiples niveles pueden codificar al menos varios bits de datos según una tensión de celda. Algunas implementaciones se refieren a arquitecturas para implementar sistemas que incluyen memoria flash de MLC profundas. Algunas implementaciones se refieren a técnicas para llevar a cabo operaciones de almacenamiento de datos con memorias flash de MLC profundas.

35 La FIG. 1 muestra un ejemplo de una cápsula de múltiples chips (MCP) 100 que proporciona almacenamiento de datos para un dispositivo principal (no mostrado). La MCP 100 incluye un chip de memoria flash NAND 103 para almacenar datos y un controlador de disco flash (FDC) 106 que facilita el acceso a la memoria flash en respuesta a comandos de lectura y/o escritura procedentes del dispositivo principal. En algunas implementaciones, el chip de memoria flash NAND 103 almacena datos en MLC profundas. Por ejemplo, las celdas del chip de memoria flash 103 pueden incluir 3, 4, 5, 6, 7, 8, 9, 10 o más bits de información. La MCP 100 puede proporcionar almacenamiento de datos en varios dispositivos portátiles, tales como cámaras digitales, otros dispositivos de almacenamiento de imágenes, dispositivos de audio portátiles, asistentes personales digitales (PDA) y grabadores de vídeo digitales, por ejemplo. Algunas implementaciones también pueden utilizarse en otras aplicaciones, cuyos ejemplos pueden incluir ordenadores de escritorio, servidores, encaminadores inalámbricos o aplicaciones integradas (por ejemplo, de automoción), particularmente en situaciones en las que se desea un acceso rápido a los datos. En general, el aparato y las técnicas según los ejemplos descritos en este documento pueden implementarse para aumentar la densidad de la memoria flash y/o llevar a cabo operaciones de almacenamiento no volátil de datos a gran rendimiento y/o fiables.

50 Como un ejemplo ilustrativo, la MCP 100 puede almacenar un archivo de datos almacenando un octeto (es decir, 8 bits) de información en cada celda de un grupo de celdas (por ejemplo, una página o un bloque) de una memoria flash. Otros ejemplos pueden tener resoluciones de 16 bits, 32 bits, 64 bits o superiores. En algunas implementaciones, la resolución puede determinarse mediante la detección de uno o de múltiples electrones en una puerta de una celda. En otras implementaciones, cualquier número práctico de bits de información puede codificarse en una tensión a la que está cargada una celda individual de memoria flash.

60 El FDC 106 incluye una interfaz principal 109, un procesador 112 y una interfaz flash 115. El FDC 106 recibe comandos y/o datos (por ejemplo, actualizaciones de código software o datos de usuario) desde y/o transmite datos a un dispositivo principal, tal como un procesador o un ordenador de escritorio, un servidor o un dispositivo informático portátil, a través de la interfaz principal 109. La comunicación con el dispositivo principal puede utilizar protocolos personalizados o estándar, tales como *Advanced Technology Attachment (ATA)*, *Serial ATA (SATA)*,

Block Abstracted NAND, Secure Digital (SD) o Multi-Media Card (MMC), por ejemplo. En algunas implementaciones, la MCP 100 puede ser parte del mismo producto que el dispositivo principal. En otras implementaciones, el dispositivo principal puede estar en comunicación operativa con la MCP 100 a través de un enlace de comunicaciones (por ejemplo, USB, *Firewire, Bluetooth*) con al menos otro dispositivo basado en procesador. Por ejemplo, un dispositivo principal puede acceder a la MCP 100 de manera remota enviando mensajes de control y enviando y recibiendo mensajes de datos a través de al menos una red, la cual puede incluir enlaces cableados, inalámbricos, de fibra óptica o una combinación de los mismos. Tales redes pueden soportar comunicaciones basadas en paquetes y pueden incluir una red local o una red de área extensa, tal como Internet.

El procesador del dispositivo principal puede leer datos de y/o escribir datos en el chip de memoria flash NAND 103 utilizando un esquema de direccionamiento lógico procesado por el FDC 106 para identificar direcciones físicas de la memoria flash. En algunas implementaciones, la interfaz principal 109 puede estar configurada para comunicarse con el dispositivo principal utilizando una interfaz ATA/IDE. El procesador 112 puede procesar el comando recibido y utilizar la interfaz flash 115 para acceder al chip de memoria flash NAND 103. El FDC 106 puede estar configurado para proporcionar funcionalidades, tales como gestión del deterioro, gestión de bloques, corrección de errores y funcionalidades de gestión de direccionamiento lógico, para mejorar el rendimiento del chip de memoria flash NAND 103, tal como aumentar la fiabilidad, reducir el tiempo de lectura y de escritura, mejorar la eficiencia energética y aumentar la capacidad por volumen de chip. Determinadas técnicas y aparatos descritos en este documento pueden aplicarse a la memoria flash NAND y/o NOR, a otros tipos de memoria que pueden borrarse eléctricamente o en las que puede escribirse eléctricamente, o a memorias en las que la resolución del acceso a datos viene dada en páginas o en bloques.

Aunque en la FIG. 1 solo se muestra un chip de memoria flash NAND 103, la MCP 100 puede incluir más de un chip de memoria flash NAND 103. Algunas implementaciones pueden incluir cualquier combinación de memorias no volátiles, que pueden incluir flash NAND, flash NOR o memorias de solo lectura programables que pueden borrarse eléctricamente (EEPROM). En algunos ejemplos ilustrativos, la MCP 100 puede incluir dos, tres, cuatro o al menos ocho chips de memoria flash NAND 103. Por ejemplo, la MCP 100 puede incluir un controlador de disco flash 106 en un chip que está encapsulado con (por ejemplo, en una pila) cuatro chips de memoria flash NAND 103.

En algunas implementaciones, el controlador de disco flash 106 y el chip de memoria flash 103 pueden estar implementados en un único chip. En otras implementaciones, uno o más de los componentes del controlador de disco flash 106 pueden estar implementados de manera total o parcialmente externa a un único chip o a la MCP 100. Por ejemplo, toda o parte de la memoria dinámica y síncrona de acceso aleatorio (SDRAM) 151 y/o de la memoria no volátil (NVM) 154 puede estar implementada de manera externa a la MCP 100. En algunas implementaciones, todo o parte del controlador de disco flash 106 puede estar encapsulado por separado con respecto al chip de memoria flash 103. En un ejemplo ilustrativo, la NVM 154, la SDRAM 151, la interfaz principal 109 y al menos una parte del procesador 112 pueden estar implementados cada uno de manera externa a la MCP 100. En otras implementaciones, las señales analógicas y/o digitales entre la interfaz flash 115 y el chip de memoria flash 103 pueden encaminarse externamente a una cápsula integrada.

Pueden implementarse estructuras de transmisión remotas o distribuidas (por ejemplo, trayectorias de señales de impedancia protegidas y/o controladas) para transportar señales a y/o desde al menos un chip de memoria flash 103. En algunas implementaciones puede proporcionarse expansión de memoria instalando cápsulas adicionales de memoria no volátil. Pueden utilizarse dispositivos de almacenamiento intermedio y/o de encaminamiento para soportar la distribución de señales analógicas y/o digitales hacia un número variable de chips de memoria 103. Además, las funciones del procesador 112 pueden llevarse a cabo de manera externa a la MCP 100. En varios ejemplos, el procesador 112 puede implementarse, total o parcialmente, en un circuito sobre el mismo sustrato (por ejemplo, placa de circuito impreso) o en el mismo producto que la MCP 100. El procesador 112 puede implementarse a partir de otro dispositivo informático en comunicación operativa con la MCP 100 a través de un enlace de comunicaciones (por ejemplo, cableado, inalámbrico, de fibra óptica o una combinación de cualquiera de los mismos).

La MCP 100 puede tener cualquier tamaño de memoria práctico, tal como hasta al menos 100 gigabytes o superior. En el ejemplo descrito, el chip de memoria flash NAND 103 está organizado para incluir una pluralidad de bloques de memoria flash 118. En algunas implementaciones, el chip de memoria flash NAND 103 puede incluir cientos o miles de bloques de memoria flash 118. Cada bloque de memoria flash 118 incluye una pluralidad de páginas de memoria flash 121. Tal y como se muestra, cada página de memoria flash 121 incluye celdas que pueden almacenar datos 124 y celdas que pueden almacenar códigos de corrección de errores (ECC) 127 asociados con los datos. Como un ejemplo, la página de memoria flash 121 puede almacenar 2048 bytes de datos y 64 bytes de datos ECC. Las celdas de datos 124 almacenan información recibida desde el controlador de disco flash 106. Las celdas ECC 127 almacenan metadatos de integridad adicional (por ejemplo, datos ECC) que están asociados con los datos almacenados en las celdas de datos 124. En varias implementaciones, los datos ECC permiten que el

controlador de disco flash 106 detecte y/o corrija errores de bits en los datos.

5 En el ejemplo ilustrado, cada bloque de memoria flash 118 incluye además una o más celdas de referencia 130a, 130b, 130c. En algunas implementaciones, el FDC 106 puede supervisar la tensión de las celdas de referencia 130a, 130b, 130c para estimar el grado de variación o hueco de tensión en las celdas 124, 127. En cada bloque 118, la celda de referencia 130a puede estar situada al principio del bloque 118, y la celda de referencia 130b puede estar situada al final del bloque 118. Cada página de memoria flash 121 puede incluir la celda de referencia 130c. En algunas implementaciones, un número mayor o menor de celdas de referencia puede estar distribuido en cualquier patrón a través de las páginas, bloques y chips de la memoria 103 para determinar el posible rendimiento de las celdas 124, 127.

15 En algunas implementaciones, las celdas de referencia pueden estar situadas en o alrededor de celdas que experimentan niveles de uso de lectura/escritura que son representativos del nivel de uso de determinadas celdas de datos de interés. Los procedimientos de compensación pueden basarse en comparar celdas que no son de referencia con otras celdas que no son de referencia. Por ejemplo, si las tensiones en una pluralidad de celdas de la misma página o bloque son relativamente bajas, entonces la compensación puede incluir ajustar umbrales (por ejemplo, umbrales de tensión entre diferentes niveles de valor en una celda) hacia valores más bajos según los valores medidos, de manera que los errores de lectura pueden reducirse sustancialmente. Otros ejemplos incluyen determinar una función de corrección según las tensiones detectadas en las celdas de referencia, donde la función de corrección ajusta la tensión detectada antes de convertir la tensión detectada en el valor de datos digital representado por la celda de memoria.

25 En algunas implementaciones, las celdas de memoria pueden refrescarse aplicando una carga adicional a una pluralidad de celdas para corregir huecos de tensión detectados. Por ejemplo, si el nivel de tensión de una o más celdas de referencia supera alguna cantidad umbral de variación de tensión, entonces las celdas de memoria de la(s) página(s) o bloque(s) asociado(s) con la(s) celda(s) de referencia pueden ajustarse aplicando una carga adicional o pueden reescribirse para restaurar las celdas a niveles de tensión apropiados según los datos almacenados. Tales ajustes pueden llevarse a cabo inmediatamente tras detectarse la variación de tensión en la(s) celda(s) de referencia o como parte de una operación de mantenimiento posterior. En algunas implementaciones puede aplicarse una carga adicional o las celdas de memoria pueden reescribirse en función de la diferencia entre la(s) tensión(es) detectada(s) de celda de referencia y la(s) tensión(es) objetivo de celda de referencia, lo que se supone que indica la cantidad aproximada de variación o hueco de tensión tanto para la(s) celda(s) de referencia como para las celdas de datos correspondientes.

35 En otras implementaciones, la aplicación de una carga adicional o la reescritura de las celdas de memoria puede llevarse a cabo leyendo todas las celdas, llevando a cabo cualquier ajuste necesario en los umbrales (por ejemplo, usando una función de corrección basada en las tensiones de celda de referencia y/o usando otras técnicas descritas en este documento), y llevando a cabo una corrección de errores en los datos detectados para obtener los datos almacenados. Después, los datos pueden usarse para determinar niveles de tensión apropiados o la cantidad de carga adicional necesaria para las diversas celdas de memoria para corregir la variación o hueco de tensión identificado. En algunas implementaciones, la cantidad de carga adicional aplicada puede determinarse según una función de corrección que es idéntica o similar a la función de corrección utilizada para ajustar las tensiones detectadas antes de convertir las tensiones detectadas en valores de datos digitales.

45 En algunas implementaciones, las celdas de la memoria flash pueden reasignarse de manera adaptativa. Por ejemplo, las celdas de referencia pueden añadirse, eliminarse, reasignarse y/o redistribuirse según sea necesario en respuesta a información de uso de lectura o escritura, la temperatura, la antigüedad del producto, la tensión de suministro (por ejemplo, baja batería, energía de línea de CA), y/o los niveles de error detectados. Si hay pocos errores en determinados bloques o páginas de la memoria, entonces pueden asignarse menos celdas como celdas ECC 127 y/o como celdas de referencia 130, lo que permite un mayor número de celdas de datos 124. Las asignaciones relativas de celdas a funciones de referencia, de datos y ECC, así como la determinación de celdas individuales, pueden ajustarse dinámicamente en función de las condiciones de funcionamiento actuales y/o según condiciones predeterminadas. Por ejemplo, la determinación puede ajustarse en función de las tasas de error, el número de celdas ECC por página puede basarse en las tasas de error y en la información del histórico de lectura y escritura, y la ubicación y la distribución de las celdas de referencia pueden basarse en tasas de error y en la antigüedad del producto. Este ejemplo simplemente ilustra que el controlador 106 y el chip de memoria flash pueden ajustarse dinámicamente según varios criterios. Otros criterios pueden incluir la criticidad de los datos, la disponibilidad de la fuente de alimentación (por ejemplo, energía de línea de CA, energía de batería), y criterios definidos sobre la importancia relativa de maximizar el tamaño de la memoria, las prestaciones por velocidad y la integridad de los datos. Por ejemplo, mantener una alta resolución de celda que requiere un gran número de correcciones de software puede dar como resultado tiempos de acceso más largos. Los criterios pueden ajustarse por el usuario, el fabricante de producto o el software, según las necesidades de la aplicación.

En algunas implementaciones, los datos que requieren un elevado número de correcciones de software pueden reescribirse en una operación de mantenimiento para corregir las variaciones de carga asociadas con el paso del tiempo o para corregir páginas de celdas de memoria que han empezado a degradarse. Normalmente, durante el cambio de resolución de una o más celdas de memoria, los datos se escribirán en una página diferente de celdas de memoria, y pueden escribirse en la misma resolución o en otra diferente. En algunas implementaciones, la página original de celdas de memoria pasará a una resolución inferior, lo que se requerirá con frecuencia a media que las celdas de memoria envejecen y se degradan. Cuando se lleva a cabo la reescritura de los datos como resultado de una variación o hueco de tensión identificado, es posible escribir los datos en la misma página o bloque, o en otros diferentes, de las celdas de memoria.

La interfaz flash 115 proporciona un control directo, un establecimiento de comunicaciones y un acceso de transferencia de datos al chip de memoria flash 103. La interfaz flash 115 incluye una interfaz de control 133 y una interfaz analógica 136. En algunas implementaciones, la interfaz de control 133 puede enviar señales de control, de direcciones y de datos al chip de memoria flash 103. Los comandos y las direcciones de memoria pueden transmitirse en señales digitales o en señales analógicas. El controlador de disco flash 106 también puede recibir señales analógicas desde el chip de memoria flash 103. El controlador de disco flash 106 puede incluir un procesador para interactuar con la lógica de memoria flash del chip de memoria flash 103, y este procesador para interactuar con la lógica de memoria flash del chip de memoria flash puede estar integrado en la interfaz flash 115.

En respuesta a un comando de lectura, el chip de memoria flash 103 puede proporcionar tensiones de celda que representan los datos almacenados en celdas de datos individuales 124. El controlador de disco flash 106 puede recibir las señales de tensión analógicas proporcionadas desde cada celda de memoria del chip de memoria flash 103. Estas tensiones de celdas analógicas o señales de tensión analógicas pueden transmitirse a la interfaz analógica 136 del FDC 106. En algunas implementaciones, la interfaz flash 115 también puede incluir un bus de datos distinto de la interfaz de control 133 y de la interfaz analógica 136 para comunicarse con el chip de memoria flash 103.

La interfaz analógica 136 puede incluir un módulo de interfaz analógico (FE analógico) 139 y un convertidor de analógico a digital (ADC) 142. Tras recibirse las señales analógicas, el FE analógico 139 puede acondicionar las señales según sea necesario, por ejemplo proporcionar un desajuste, un desfase de nivel correctivo, ganancia, almacenamiento intermedio, filtrado o una impedancia controlada para minimizar las reflexiones. El FE analógico puede proporcionar una entrada de alta impedancia para minimizar la carga de la celda de memoria flash, y una salida de baja impedancia para activar un circuito de muestreo y retención o de seguimiento y retención que está acoplado a una entrada del ADC 142. En algunas implementaciones, el FE analógico 139 puede incluir además un multiplexor analógico (no mostrado) para seleccionar una línea de entre una pluralidad de líneas de salida analógicas de uno o más chips de memoria flash.

El ADC 142 procesa el valor analógico para determinar una representación de valor de datos digital correspondiente de la tensión en las celdas de datos 124, 127. El ADC 142 recibe la señal analógica acondicionada y convierte la señal analógica en una representación digital de la tensión analógica. Después, el ADC 142 (o un procesador del ADC) convierte la representación digital en un valor de datos digital representado por la tensión almacenada en la celda de memoria según, por ejemplo, una función de mapeo. El procesador 112 también puede usarse para convertir la representación digital en un valor de datos digital. La representación digital de la tensión analógica puede incluir suficiente información para permitir que el ADC 142 o un procesador distingan entre una pluralidad de niveles de tensión analógicos que representan cada uno un valor de datos digital particular. La representación digital puede comprender un mayor número de bits de datos que el valor de datos digital. En algunas implementaciones, el ADC 142 puede estar integrado en el chip de memoria flash 103 en lugar de estar incluido en el controlador de disco flash 106. En este caso, la interfaz flash 115 puede recibir representaciones digitales de tensiones de celda o valores de datos digitales del chip de memoria flash 103.

Se muestra un ejemplo de una función de mapeo 145. Según la función de mapeo 145, el ADC 142 o el procesador 112 puede convertir una tensión de celda analógica en una representación digital y/o en un valor de datos digital. Por ejemplo, puede haber una serie de umbrales de tensión analógica que pueden usarse para mapear una tensión analógica con una representación digital y/o un valor de datos digital. Asimismo, la función de mapeo 145 también puede ilustrar la conversión de una representación digital de la tensión analógica en un valor de datos digital. Por ejemplo, una o más representaciones digitales de la tensión analógica pueden mapearse con un valor de datos digital particular, donde cada valor de datos digital tiene un conjunto diferente correspondiente de una o más representaciones digitales.

En algunas implementaciones, el ADC 142 o el procesador 112 pueden recibir parámetros que modifican la función de mapeo 145. Por ejemplo, el FDC 106 puede adaptar la función de mapeo 145 en función de la temperatura

actual, la tensión de suministro, el número de lecturas y escrituras de los datos de página y/o la tensión en las celdas de referencia 130a, 130b y/o 130c. En algunas implementaciones, las adaptaciones a la función de mapeo pueden basarse en características de tensión de celdas de datos vecinas 124, celdas ECC 127 y/u otras celdas. El mapeo 145 entre tensiones de celda y valores de datos digitales se describe en mayor detalle con referencia a las

5 FIG. 2A y 2B. En algunas implementaciones, el ADC 142 o un procesador también pueden funcionar según un comando de valor alternativo para recuperar valores alternativos para las señales analógicas recibidas o representaciones digitales de las señales analógicas. Implementaciones de ejemplo del comando de valor alternativo se describen en mayor detalle con referencia a las FIG. 6A a 6C.

10 El controlador de disco flash 106 incluye además un motor ECC 148. En varias implementaciones, el motor ECC 148 puede llevar a cabo comprobaciones y correcciones de errores de hardware y/o software utilizando celdas ECC 127. En algunas implementaciones, el motor ECC 148 puede proporcionar una recuperación de datos basada en una máquina de estados. Por ejemplo, el motor ECC 148 puede detectar el número de bits de error en una página de datos. Después, el motor ECC 148 puede determinar qué algoritmo ECC utilizar. Como un ejemplo, el motor

15 ECC 148 puede configurarse para intentar en primer lugar un algoritmo ECC de hardware que utiliza, por ejemplo, códigos de Hamming o de Reed-Solomon. Si el algoritmo ECC de hardware no es satisfactorio para recuperar la página de datos, entonces puede intentarse una corrección ECC de software. Un procedimiento de ejemplo que ilustra el uso de ECC de hardware, de ECC de software y de otras técnicas combinadas se describe con referencia a la FIG. 5. En algunas implementaciones, el motor ECC 148 puede proporcionar una corrección de errores de

20 hasta al menos el 10% o más del tamaño de un página de datos. En algunos ejemplos, un procesador puede determinar qué algoritmo ECC utilizar.

En algunas implementaciones, el procesador 112 reescribirá o refrescará los datos almacenados en una página de memoria flash si se utiliza un algoritmo ECC para recuperar datos que superan un número predeterminado de

25 porcentaje de error. En otras implementaciones, el procesador 112 registrará en un registro de mantenimiento la posición, física y/o lógica, de los datos que incluyeron tales errores. Después, el procesador 112 reescribirá o refrescará esos datos durante una operación de mantenimiento (véase la FIG. 9). Las operaciones de mantenimiento pueden llevarse a cabo cuando el dispositivo principal está funcionando bajo unas condiciones de energía predeterminadas, cuando el procesador 112 tiene una cantidad predeterminada de exceso de ancho de

30 banda y/o a intervalos planificados.

El controlador de disco flash (FDC) 106 puede incluir una memoria de acceso aleatoria dinámica (DRAM). El controlador de disco flash 106 de este ejemplo incluye además una memoria de acceso aleatoria dinámica y

35 síncrona (SDRAM) 151. Por ejemplo, la SDRAM 151 puede ser una SDRAM de velocidad de transferencia datos única o una SDRAM de doble velocidad de transferencia datos. En algunas implementaciones, el FDC 106 puede utilizar la SDRAM 151 como una memoria intermedia de alta velocidad y de alta densidad para almacenar datos temporales tales como datos de salida para el dispositivo principal y valores digitales alternativos para una página de datos, por ejemplo. El FDC 106 puede incluir además otros tipos de RAM, tales como DRAM. Como un ejemplo, el FDC 106 puede recibir datos analógicos desde el chip de memoria flash NAND 103.

40 Después, el FDC 106 puede convertir las tensiones analógicas detectadas en datos digitales, incluyendo, en algunos casos, valores de datos digitales alternativos para una o más de las celdas. Después, el motor ECC 148 comprueba y corrige los datos digitales, comprobando posiblemente múltiples combinaciones diferentes de valores de datos y valores de datos alternativos para las celdas de cada página de memoria flash 121. Si la corrección de

45 errores es satisfactoria, entonces el procesador 112 puede almacenar los datos digitales en una memoria intermedia principal de salida de la SDRAM 151. En algunas implementaciones, el dispositivo principal puede recuperar datos de la memoria intermedia principal de salida. Como alternativa, el controlador de disco flash 106 puede reenviar datos desde la memoria intermedia principal de salida al dispositivo principal. La SDRAM 151, u otra memoria temporal, puede utilizarse además para almacenar datos que van a escribirse en el chip de memoria flash

50 103.

El FDC 106 incluye además una memoria no volátil (NVM) 154. En este ejemplo, la NVM 154 incluye código software de gestión de deterioro 157, código software de gestión de bloques 160, código software de direccionamiento lógico 163 y registros de resolución de celda 166, cada uno de los cuales contiene instrucciones

55 (o punteros a instrucciones en la memoria flash) que, cuando se ejecutan mediante el procesador 112, llevan a cabo determinadas operaciones. En algunas implementaciones, la NVM 154 puede estar separada del chip de memoria flash NAND 103. Por ejemplo, la NVM 154 puede ser una memoria flash NOR u otra memoria flash NAND. En otras implementaciones, la NVM 154 puede ser una o más páginas del chip de memoria flash NAND 103. En otras implementaciones, la NVM 154 puede almacenar punteros o posiciones de memoria a los datos almacenados

60 en el chip de memoria flash NAND 103. En algunas implementaciones, el procesador 112 puede ejecutar el código software de gestión de deterioro 157, el código software de gestión de bloques 160 y el código software de direccionamiento lógico 163 para mejorar la eficacia, el rendimiento y/o la fiabilidad de la MCP 100.

El procesador 112 puede utilizar el código software de gestión de deterioro 157 para gestionar el deterioro de las páginas 121, los bloques 118 o el chip 103 de la MCP 100. Por ejemplo, el código software de gestión de deterioro 157 puede incluir instrucciones que, cuando se ejecutan por el procesador 112, llevan a cabo operaciones que incluyen operaciones de equilibrado de carga para intercambiar los datos de la página de memoria más frecuentemente utilizada por los de una página de memoria menos utilizada. Las operaciones de intercambio también pueden incluir una actualización del código software de direccionamiento lógico 163.

El código software de gestión de deterioro 157 puede activarse durante una operación de mantenimiento. En algunas implementaciones, las direcciones físicas y/o lógicas de cada operación de lectura se registran en un registro de mantenimiento. Cada operación de escritura también puede registrarse en un registro de mantenimiento. Después, el código software de gestión de deterioro 157 puede utilizar valores umbral predeterminados para determinar el modo de reordenar los datos almacenados entre las páginas de celdas de memoria. Estos valores umbral, por ejemplo, pueden incluir 100 ó 1000 lecturas de la página de celdas de memoria durante el transcurso de una semana o un mes. En otras implementaciones, los valores umbral pueden estar basados en un porcentaje del número total de operaciones de lectura, o en una desviación con respecto al número medio de lecturas por página y por unidad de tiempo. En la FIG. 9 se muestra un ejemplo de una operación de mantenimiento.

El código software de gestión de bloques 160 puede incluir código para gestionar bloques no válidos del chip de memoria flash 103. Por ejemplo, el código software de gestión de bloques 160 puede incluir información del histórico de errores de los bloques de memoria flash 118. En algunas implementaciones, la información de errores puede utilizarse para mantener la resolución de celda en cada una de las páginas de memoria flash. Un ejemplo del código software de gestión de bloques se describe en mayor detalle con referencia a las FIG. 8A y 8B.

El código software de gestión de bloques 160, posiblemente junto con el código software de direccionamiento lógico 163 y/o los registros de resolución de celda 166, también puede utilizarse para emparejar conjuntos de bloques no válidos o páginas no válidas que tienen resoluciones reducidas (actualizadas en los registros de resolución de celda 166) en el chip de memoria flash 103 y tratar el conjunto de bloques no válidos o de páginas no válidas para fines de direccionamiento lógico (quizá actualizados en el código software de direccionamiento lógico 163 y/o en los registros de resolución de celda 166) de una manera equivalente a un único bloque o una única página de celdas de memoria que tienen la resolución superior inicial. El código software de gestión de bloques 157 puede activarse durante una operación de mantenimiento. Un ejemplo de una operación de mantenimiento se ilustra en la FIG. 9.

El código software de direccionamiento lógico 163 puede incluir código para convertir una dirección lógica de un comando de dispositivo principal en direcciones físicas del chip de memoria flash NAND 103. En algunos ejemplos, una página lógica puede estar asociada con múltiples páginas de memoria física en el chip de memoria flash NAND 103. El código software de direccionamiento lógico 163 gestiona la conversión y la actualización de la tabla de direcciones lógicas de la NVM 154. En un ejemplo, el código software de direccionamiento lógico 163 puede mantener dinámicamente enlaces entre direcciones de bloque lógicas del dispositivo principal y direcciones de página físicas cuando las páginas pasan de una resolución de 10 bits a una resolución de 8 bits, por ejemplo, o cuando el mapeo de direcciones de bloque lógicas con direcciones de página físicas diferentes se modifica para fines de gestión de deterioro. Formas intermedias de direcciones pueden generarse en el proceso de conversión entre direcciones lógicas y físicas, por ejemplo. Formas intermedias de direcciones pueden generarse, procesarse, almacenarse, utilizarse y/o manipularse de otro modo para llevar a cabo varias operaciones de memoria no volátil. Un ejemplo del código software de direccionamiento lógico se describe en mayor detalle con referencia a la FIG. 10.

Los registros de resolución de celda 166 almacenan información sobre la resolución de celda en cada página de memoria flash 121. Por ejemplo, el chip de memoria flash NAND 103 puede ser una memoria flash MLC de 8 bits. En algunas implementaciones, parte del bloque de memoria flash 118 puede degradarse o actualizarse en respuesta a varias condiciones. Ejemplos ilustrativos de tales condiciones incluyen un rendimiento erróneo, la temperatura, condiciones de tensión, el número de ciclos de lectura o escritura de celdas individuales, grupos de celdas, páginas, celdas en una posición vecina, celdas de referencia, celdas con un histórico de uso de lectura y/o escritura comparable, u otros factores, tales como la antigüedad del dispositivo. Información sobre todas o algunas de estas condiciones puede estar almacenada en un dispositivo de almacenamiento de datos, o puede determinarse o estimarse a partir de uno o más bits de información almacenada. En un ejemplo, la información almacenada puede incluir datos del histórico de uso de lectura y escritura que representan los niveles de utilización para al menos algunas de las celdas del chip de memoria 103. El procesador 112 puede actualizar los registros de resolución de celda 166 para reducir una resolución de celda de una página de memoria degradada a, por ejemplo, 4 bits, de manera que la página de memoria flash 121 todavía puede utilizarse con un tamaño de memoria más pequeño. En otras implementaciones, los registros de resolución de celda 166 también pueden almacenar la resolución de celda para cada bloque de memoria flash 118.

En algunas implementaciones, los registros de resolución de celda 166 se ajustan a la baja hasta una resolución de un solo bit u otra resolución de un número bajo de bits antes de transferir datos desde un dispositivo principal hasta las celdas de memoria de la MCP 100. Este proceso se describe en mayor detalle en la FIG. 7B. La reducción de valor en los registros de resolución de celda 166 antes de la transferencia de datos puede permitir mayores velocidades de transferencia de datos ya que se necesita menos precisión para cargar cada celda de memoria. Los datos transferidos pueden reescribirse posteriormente en celdas de memoria a una mayor resolución. En algunas implementaciones, los datos transferidos pueden reescribirse a una mayor resolución durante una operación de mantenimiento (por ejemplo, en un momento posterior cuando hay disponibles suficientes recursos de procesamiento y la reescritura no interfiere con otras operaciones de lectura o escritura). En algunas implementaciones, la transferencia de datos de baja resolución de celda se registra en un registro de mantenimiento.

En algunas implementaciones, el código software de direccionamiento lógico 163, los registros de resolución 166 y/o el código software de gestión de bloques 160 agruparán páginas de memoria degradadas (o bloques de memoria degradados) y tratará el grupo para fines de direccionamiento lógico como una única página (o bloque) de memoria no degradada. No es necesario que las páginas de memoria del grupo de páginas de memoria degradadas sean páginas de memoria adyacentes. El grupo de páginas de memoria degradadas puede incluir páginas de memoria de diferentes bloques e incluso de diferentes chips de memoria. En algunas implementaciones, cada página o bloque de memoria degradados de un grupo de páginas o bloques de memoria degradados se degrada en respuesta a una condición de error asociada con la página o bloque.

Las FIG. 2A y 2B muestran de manera conjunta mapeos entre tensiones de celda y valores de datos digitales almacenados en la celda de memoria. Tal y como se muestra en la FIG. 2A, se muestra una distribución ilustrativa de valores de datos digitales 200 de una celda de memoria de 8 bits. Una celda de memoria de 8 bits incluye 256 valores de datos digitales posibles; una celda de memoria de 4 bits incluye 16 valores de datos posibles. El número de valores de datos posibles es igual a 2^n (donde n es igual al número de bits), pero no es necesario que el número de valores de datos digitales posibles corresponda a un número de n bits de valores de datos digitales posibles. Cada celda de memoria puede tener cualquier número entero de posibles valores de datos digitales mayor que 1, por ejemplo, algunas celdas de memoria pueden tener 10 valores de datos posibles. La distribución de valores digitales 200 incluye curvas de distribución de valores digitales 205 a 210 que representan la distribución de tensión para cada valor de datos digital. Cada curva de distribución de valores digitales (por ejemplo, 205 a 210) representa un intervalo de valores de tensión digitales correspondientes a niveles de tensión asociados con cada posible valor de datos digital.

Durante una operación de escritura, cada celda de memoria recibe una carga para una tensión analógica correspondiente a un valor de datos digital seleccionado entre uno de los posibles valores de datos digitales. Normalmente, esta tensión correspondiente está dentro de las curvas de distribución 205 a 210 para los valores de datos digitales deseados. Esta tensión correspondiente también puede ser una tensión objetivo correspondiente al valor de datos digital. Por ejemplo, si una tensión de celda está dentro de la distribución 207, entonces el valor digital almacenado en la celda puede ser 02_H. Durante una operación de lectura, una señal de tensión analógica se detecta desde cada celda. Después, el ADC 136 convierte la señal de tensión analógica en una representación digital de la señal de tensión analógica. Después, esta representación digital se compara con al menos una curva de distribución de valores digitales para determinar el valor de datos digital representado por la tensión analógica almacenada en la celda de memoria leída.

La distribución de valores de datos digitales 200 incluye áreas grises 215 entre las curvas de distribución de valores de datos digitales 205 a 210. En algunas implementaciones, cuando el ADC 142 recibe una tensión de celda o detecta una señal de tensión analógica que está dentro de una de las áreas grises 215, el ADC 142 puede, por ejemplo, convertir la tensión de celda al valor de datos digital adyacente más cercano. Por ejemplo, si el ADC 142 recibe una tensión de celda sustancialmente cercana a un nivel de tensión 220, entonces el ADC 142 puede optar por el valor de datos digital adyacente más cercano, en concreto FEH. En algunas implementaciones, el FDC 106 también puede incluir un comando de valor alternativo que ordena al ADC 142 optar por un valor alternativo en lugar de por el valor adyacente más cercano en función de algunos parámetros.

En algunas implementaciones, el FDC 106 puede utilizar tanto el valor de datos digital adyacente más cercano como uno o más valores alternativos en un proceso de corrección de errores que intenta corregir una página o bloque de valores de datos. Además, el FDC 106 puede asignar una incertidumbre a tensiones de celda particulares o a valores de datos correspondientes en función de la ubicación de la tensión de celda en las curvas de distribución de valores de datos digitales 205 a 210 o en las áreas grises 215. La incertidumbre asignada puede utilizarse por un algoritmo que trate de corregir una página o bloque de valores de datos. Algunos ejemplos de estos parámetros pueden incluir uno o más de entre la temperatura, el número de lecturas en la celda, el número de escrituras en la celda, la tensión de suministro y la tensión en las celdas de referencia 130a, 130b, 130c. En

algunos ejemplos, la tensión de celda puede caer por debajo de una tensión mínima de celda (V_{min}). El FDC 106 puede implementar una corrección añadiendo un desfase a la tensión de celda recibida. Este desfase puede añadirse por el FE analógico 139 o añadirse digitalmente por el ADC 142 o el procesador 112.

5 En algunas implementaciones, el FDC 106 puede ajustar dinámicamente las posiciones y los anchos de las áreas grises 215 alternando la distribución de valores de datos digitales 200. Por ejemplo, el FDC 106 puede incluir código software de mantenimiento que ajusta las áreas grises 215 en función de parámetros tales como una o más tensiones de celdas de referencia, el uso de la celda de memoria y otros heurísticos que pueden haberse cargado previamente en la NVM 154. El código software de mantenimiento también puede llevar a cabo la actualización de los registros de resolución de celda 166. Por ejemplo, cada chip 103, la interfaz analógica 135 y/o la MCP 100 pueden caracterizarse en el tiempo de fabricación, y una tabla de linealización, factores de corrección u otro ajuste correctivo pueden almacenarse en la memoria no volátil de la MCP 100. En algunos casos, los niveles de tensión máximo y mínimo (V_{max} y V_{min}) así como las curvas de distribución de valores digitales 205 a 210 pueden ajustarse y/o redistribuirse en función de pruebas empíricas de las celdas durante su vida útil.

15 En la FIG 2B se muestra un gráfico 250 que relaciona tensiones de celda con valores digitales. El gráfico 250 incluye una característica de tensión ideal 255 que el ADC 142 utiliza para convertir tensiones analógicas en valores digitales. En algunos ejemplos, la celda de datos 124 puede almacenar valores digitales según características de tensión no ideales 260, 265 debido a, por ejemplo, los heurísticos de temperatura, la antigüedad de la celda, tolerancias de la tensión de suministro o de la bomba de carga, la no linealidad del ADC 136, los errores detectados en la celda de memoria y/o el número de lecturas y escrituras en la celda. El FDC 106 puede compensar de varias formas las características de tensión 260, 265 para que se aproximen a las características ideales 255. Procedimientos de compensación a modo de ejemplo se describen con referencia a las FIG. 3A, 3B, 4, 5 y 6A a 6C.

25 Las FIG. 3A y 3B muestran diagramas de flujo que ilustran ejemplos de procesos 350 y 300 para leer una página de datos de una memoria flash NAND. Los procesos 350 y 300 incluyen operaciones que pueden llevarse a cabo generalmente por el procesador 112. En algunas implementaciones los procesos 350 y 300 también pueden llevarse a cabo, complementarse o ampliarse por otros elementos de procesamiento y/o de control que pueden incorporarse con el ADC 142. Por ejemplo, puede haber un controlador o un compensador en la interfaz analógica 136 que lleve a cabo todas o algunas de las operaciones de los procesos 350 y 300.

30 La FIG. 3A ilustra un proceso que convierte niveles de tensión detectados de celdas de memoria de múltiples niveles en valores de datos digitales. El proceso 350 empieza con la detección de un nivel de tensión analógica de una celda de memoria de múltiples niveles (etapa 355). Esta tensión puede detectarse por la interfaz analógica 136, por ejemplo. La interfaz analógica 136 puede incluir una entrada que puede hacerse funcionar para recibir señales analógicas de un chip de memoria flash 103. El controlador de disco flash 106 puede incluir además un módulo de control para seleccionar celdas de memoria a partir de las cuales la entrada recibe señales analógicas. En la etapa 360, la señal de tensión analógica se convierte en una representación digital de la tensión analógica detectada. Esta conversión puede llevarse a cabo por el ADC 142. La representación digital puede tener suficientes datos como para permitir que el ADC 142 o el procesador 112 distingan el nivel de la tensión analógica almacenada por una celda de memoria de entre una pluralidad de posibles niveles de tensión que representan un valor de datos digital. Esto puede conseguirse mediante una representación digital que comprende más bits de datos que el valor de datos digital representado por la tensión almacenada en la celda de memoria.

45 La FIG. 2A ayuda a ilustrar este concepto. El intervalo de tensiones de celda analógicas posibles pueden dividirse en múltiples segmentos (por ejemplo, tal y como se representa mediante el nivel de tensión 220), correspondiendo cada uno a una representación digital de la tensión de celda analógica. Cada curva de distribución de valores digitales 205 a 210 y cada área gris 215 puede incluir múltiples segmentos de este tipo, lo que permite la utilización de representaciones digitales que tienen una mayor resolución que las curvas de distribución de valores digitales 205 a 210, que pueden proporcionar información adicional relacionada con, por ejemplo, el lugar que ocupa una tensión de celda en una curva de distribución de valores digitales 205 a 210 o en un área gris 215.

55 En la etapa 365, la representación digital se convierte en un valor de datos digital en función de una distribución de valores de datos digitales. La distribución de valores de datos digitales puede almacenarse en los registros de resolución de celda 166 y puede ser la distribución de valores de datos digitales 200 mostrada en la FIG. 2A. En la etapa 335, un procesador o controlador determina si hay más celdas de memoria que leer. Si es así, entonces el proceso vuelve a la etapa 355. En caso contrario, el proceso 350 finaliza.

60 La FIG. 3B ilustra en mayor detalle un proceso que almacena valores de datos digitales identificados y que marca la ubicación de valores de datos digitales inciertos. El proceso 300 comienza cuando, por ejemplo, el procesador 112 recibe un comando para recuperar una página de datos del chip de memoria flash NAND 103. En la etapa 305, el procesador 112 recupera información de resolución de celda para una página desde los registros de resolución de

celda 166. Después, en la etapa 310, el procesador 112 recibe desde el ADC 142 un valor de salida digital para una celda de datos. El valor de salida digital para la celda de datos es una representación digital de la tensión detectada desde la celda de datos. El ADC 142 determina en la etapa 310 el valor de datos digital recibido en función de los umbrales almacenados. En algunas implementaciones, el procesador 112 puede utilizar información de los registros de resolución de celda 166 para determinar el conjunto de umbrales utilizado. Estos umbrales pueden estar relacionados con las curvas de distribución de valores digitales 205 a 210 descritas anteriormente con relación a la FIG. 2A. Por ejemplo, el procesador 112 puede usar un conjunto de umbrales para una celda de 8 bits y otro conjunto de umbrales para una celda de 2 bits. En algunos casos, el procesador 112 puede usar un conjunto de umbrales para una celda de 8 bits y otro conjunto de umbrales para una celda de 8 bits diferente. Cada conjunto de umbrales puede corresponder a una distribución de valores de datos digitales posible y puede constituir intervalos de representaciones digitales de tensiones analógicas que corresponden a valores de datos digitales posibles.

En la etapa 320, el procesador 112 determina si los valores de datos digitales para un valor de tensión analógica recibido son inciertos. En algunas implementaciones, el procesador 112 puede determinar que un valor de datos digital es incierto si la tensión de celda está en una zona gris 215 de la distribución de valores digitales 200 o si la tensión de celda está cerca del límite entre una curva de distribución de valores digitales 205 a 210 y una zona gris 215. En algunas implementaciones pueden asignarse diferentes niveles de incertidumbre dependiendo de dónde esté la tensión de celda en la distribución de valores digitales 200 (por ejemplo, las tensiones más altas pueden tender a tener una mayor incertidumbre y/o la incertidumbre puede ser mayor para tensiones de celda que están más próximas a la parte central de una zona gris 215). En la etapa 320, si el procesador 112 determina que los valores digitales recibidos no son inciertos, entonces el procesador 112 almacena el valor digital recibido en una memoria intermedia principal de salida en la etapa 325. Si el procesador 112 determina en la etapa 320 que el valor digital recibido es incierto, entonces el procesador 112 puede marcar la ubicación del valor digital incierto en una tabla de máscaras en la etapa 330 y después ejecutar la etapa 325. En algunas implementaciones, también pueden almacenarse uno o más valores alternativos para su uso posterior para determinar qué valor (por ejemplo, el valor incierto o uno de los valores alternativos) es correcto.

Después de que el procesador 112 almacene el valor digital recibido, el procesador 112 determina, en la etapa 335, si hay más celdas que leer. Por ejemplo, el procesador 112 puede comprobar si se ha llegado al final de la página de memoria. Si hay más celdas que leer, entonces el proceso vuelve a la etapa 310. Si no hay más celdas que leer, el proceso 300 finaliza. En algunas implementaciones, el proceso también registrará en un registro de mantenimiento el número de valores de datos inciertos asociados con una página o bloque de celdas de memoria. En otras implementaciones, el proceso registrará la posición, física y/o lógica, de una página y/o bloque de celdas de memoria si el número de valores de datos inciertos supera un umbral predeterminado.

La FIG. 4 muestra un diagrama de flujo que ilustra un ejemplo de un proceso 400 para leer una página de datos de una memoria flash MLC, tal como el chip de memoria flash NAND 103, usando una función de corrección para ajustar el mapeo de las tensiones de celda con los valores digitales. El proceso 400 puede llevarse a cabo por el procesador 112, por ejemplo. El proceso 400 comienza en la etapa 405 cuando el procesador 112 determina si se ha recibido un comando de lectura. Por ejemplo, el FDC 106 puede recibir un comando de lectura desde el dispositivo principal a través de la interfaz principal 109. Si, en la etapa 405, el procesador 112 determina que no se ha recibido ningún comando de lectura, entonces se repite la etapa 405.

Si el procesador 112 determina que se ha recibido un comando de lectura en la etapa 405, entonces el procesador 112 actualiza una función de corrección en la etapa 410 basándose en la temperatura, el número de lecturas o escrituras en la página de memoria, la tensión suministrada y/u otras condiciones de funcionamiento del chip de memoria flash NAND 103. En algunas implementaciones, el ADC 142 o la interfaz analógica 136 puede utilizar la función de corrección para ajustar las tensiones de celda medidas en el módulo de interfaz analógico 139 antes de que las tensiones de celda se conviertan en valores digitales. En otras implementaciones, el procesador 112 puede utilizar la función de corrección para ajustar los umbrales en la función de mapeo, por lo que el ADC 142 puede convertir la tensión analógica en valores digitales ajustados. La función de corrección puede ser diferente para distintas celdas. Por ejemplo, las celdas de memoria que tienen tensiones detectadas más altas pueden tener un mayor ajuste debido a la función de corrección.

Después, el procesador 112 selecciona una celda de referencia en la etapa 415. Por ejemplo, el procesador 112 puede seleccionar una de las celdas de referencia 130a, 130b ó 130c. Después, el procesador 112 lee, en la etapa 420, una tensión de referencia almacenada en la celda de referencia seleccionada. En la etapa 425, el procesador 112 actualiza la función de corrección basándose en la tensión de referencia. Por ejemplo, si parece que una tensión de referencia varía en un diez por ciento, entonces el procesador 112 puede ajustar la función de corrección para compensar el hueco de tensión en los datos. En algunas implementaciones, la función de corrección ajustará de manera no lineal los niveles de tensión detectados. La función de corrección puede ajustar en mayor medida los niveles de tensión detectados más altos que los niveles de tensión detectados más bajos. La función de corrección

puede ajustar tensiones detectadas a niveles de tensión diferentes mediante diferentes cantidades de ajuste o mediante diferentes porcentajes de ajuste.

5 En algunas implementaciones, los umbrales pueden ajustarse dinámicamente sobre la marcha durante el funcionamiento. En algunas implementaciones, el procesador 112 puede almacenar un número fijo de muestras anteriores, por ejemplo cien muestras, de tensiones de referencia leídas anteriormente y utilizar un promedio variable de las tensiones de referencia almacenadas para actualizar la función de corrección. La función de corrección también puede actualizarse según otras funciones, lo que puede requerir el promedio, la mediana, la moda o un promedio ponderado, por ejemplo. Por ejemplo, puede utilizarse un promedio variable ponderado.

10 Después, el procesador 112, en la etapa 430, determina si seleccionar otra celda de referencia. Como un ejemplo, el procesador 112 puede determinar si hay suficiente información para ajustar la función de corrección. Como otro ejemplo, el procesador 112 puede configurarse para leer todas las celdas de referencia en algunos bloques de memoria así como en algunas páginas de memoria en función del comando de lectura.

15 Si, en la etapa 430, el procesador 112 determina que hay otra celda de referencia que leer, entonces el proceso 400 vuelve a la etapa 415. En algunas implementaciones, el proceso de ajustar la función de corrección leyendo la tensión en celdas de referencia se activa por los errores detectados en los datos recuperados a partir de un grupo de celdas de memoria. En otras implementaciones, los errores detectados darán como resultado un desajuste de los umbrales para determinar un valor de datos asociado con una tensión detectada. En algunas implementaciones

20 puede reducirse automáticamente el valor de estos umbrales pero, en otras implementaciones, los umbrales se ajustan en función de la tensión de una o más celdas de referencia. El error puede detectarse utilizando el ECC 127 asociado con el grupo de celdas de memoria.

25 Si el procesador 112 determina en la etapa 430 que no hay más celdas de referencia que leer, entonces el procesador, en la etapa 435, selecciona una página para su lectura basándose en el comando de lectura. Después, en la etapa 440, el procesador 112 lee la página de datos seleccionada de la memoria flash utilizando, por ejemplo, el proceso 300 (FIG. 3B). En la etapa 445, el procesador 112 corrige los datos de página utilizando la función de corrección. Por ejemplo, el procesador 112 puede fijar algunos parámetros en la interfaz analógica 136 para ajustar la función de mapeo. Como otro ejemplo, el procesador 112 puede ajustar la representación digital, proporcionada

30 por el ADC 142, utilizando la función de corrección. Después, el procesador 112 puede llevar a cabo operaciones de comprobación de errores para comprobar, en la etapa 450, si hay algún error en la página. En algunas implementaciones, las operaciones de comprobación de errores pueden hacerse en el motor ECC 148 utilizando circuitos de detección de errores de hardware. En otras implementaciones, las operaciones de comprobación de errores pueden hacerse en software, donde el procesador 112 puede ejecutar un código de detección de errores

35 almacenado en la NVM 154 para comprobar si hay errores en la página. Después de las operaciones de comprobación de errores, en la etapa 455, el procesador 112 puede determinar si se ha detectado algún error.

Si no se ha detectado ningún error, entonces el procesador 112 puede transmitir, en la etapa 460, los datos leídos al dispositivo principal. Después, el procesador 112 puede determinar, en la etapa 465, si hay otra página que leer.

40 Si hay más páginas que leer, entonces se repite la etapa 435. En caso contrario, el proceso 400 finaliza. Si hay uno o más errores detectados en la etapa 455, entonces, en la etapa 470, el procesador 112 puede llevar a cabo operaciones de corrección de errores, un ejemplo de las cuales se describe con referencia a la FIG. 5. Después, el procesador 112, en la etapa 475, puede determinar si la operación de corrección de errores es satisfactoria. Si la operación de corrección de errores es satisfactoria, entonces se repite la etapa 460. Si la operación de corrección

45 de errores no es satisfactoria, entonces el procesador 112 puede almacenar información de error (por ejemplo, un registro de errores) en la NVM 154 en la etapa 480 y el proceso puede continuar en la etapa 465. La información de error también puede almacenarse en un registro de mantenimiento. La información de error almacenada puede utilizarse para operaciones de gestión de bloques, para las cuales se describe un ejemplo con referencia a la FIG. 8A. En algunas implementaciones, el procesador 112 registrará la variación entre las celdas de referencia de una

50 página o bloque de celdas de memoria en un registro de mantenimiento en la NVM 154. En otras implementaciones, el procesador 112 solo registrará la posición, física y/o lógica, de una página y/o bloque de celdas de memoria en un registro de mantenimiento si el grado de variación en las celdas de referencia satisface una condición predeterminada. Por ejemplo, si la variación en la celda de referencia supera el 10% o si la diferencia entre el grado de variación en diferentes celdas de referencia supera el 10%, los datos almacenados en la página

55 y/o bloque de celdas de memoria pueden refrescarse aplicando una carga adicional a las celdas de memoria o reescribiendo completamente la página durante una operación de mantenimiento. Un ejemplo de una operación de mantenimiento se describe con referencia a la FIG. 9.

60 La FIG. 5 muestra un diagrama de flujo que ilustra un ejemplo de un proceso 500 para llevar a cabo operaciones de corrección de errores para corregir una página de datos que contiene errores de bit. El proceso 500 comienza cuando, por ejemplo, el procesador 112 detecta errores de bit en una página de datos leída de la memoria flash y envía un comando al motor ECC 148 para llevar a cabo un algoritmo ECC de hardware para corregir los errores de

bit en la etapa 505. En algunas implementaciones, el motor ECC 148 y el ADC 142, y/o la interfaz analógica 136 pueden actuar conjuntamente para corregir los errores de bit.

5 Después, el motor ECC 148 puede comprobar, en la etapa 510, si el algoritmo ECC de hardware es satisfactorio. Si el algoritmo ECC de hardware puede corregir todos los errores de la página de datos, entonces el algoritmo ECC de hardware es satisfactorio. Después, en la etapa 515, el motor ECC 148 almacena el resultado ECC en, por ejemplo, la SDRAM 151. Después, el motor ECC 148 genera un mensaje que indica "corrección de errores satisfactoria" en la etapa 518 y el proceso 500 finaliza.

10 Si el número de bits de error existentes supera el número de bits de error que el algoritmo ECC de hardware puede corregir, entonces el motor ECC 148 envía un mensaje a la interfaz analógica 136 para volver al leer, en la etapa 520, la página de datos de la memoria flash. Después, en la etapa 525, el motor ECC 148 ejecuta de nuevo un algoritmo ECC de hardware. En la etapa 530, el motor ECC 148 comprueba si el algoritmo ECC de hardware es satisfactorio. Si el algoritmo ECC de hardware puede llevar a cabo correcciones, entonces el algoritmo ECC de hardware es satisfactorio, y el proceso continúa en la etapa 515.

20 En la etapa 530, si el motor ECC 148 determina que el número de bits de error existente supera el número de bits de error que el algoritmo ECC de hardware puede corregir, entonces el motor ECC 148 ejecuta un comando de valor alternativo para corregir los errores de bits. Implementaciones de ejemplo del comando de valor alternativo se describen con referencia a las FIG. 6A a 6C. Después, el motor ECC 148 puede comprobar si el comando de valor alternativo corrige los errores de bit en la etapa 535. Si el motor ECC 148 determina que se han corregido los errores de bit, entonces el proceso continúa en la etapa 515.

25 Si el motor ECC 148 determina que no se han corregido los errores de bit, entonces el motor ECC 148 puede ejecutar un algoritmo ECC de software extendido en la etapa 540 para recuperar la página de datos. Por ejemplo, el algoritmo ECC de software extendido puede incluir algoritmos ECC más profundos que utilizan más bits ECC. Por ejemplo, el algoritmo ECC de hardware puede necesitar cuatro bits ECC y el algoritmo ECC de software extendido puede utilizar 128 bits ECC. Después, el motor ECC 148 puede comprobar, en la etapa 550, si el algoritmo ECC de software extendido es satisfactorio. Si el motor ECC 148 determina que el algoritmo ECC de software extendido es satisfactorio, entonces el proceso continúa en la etapa 515. Si, en la etapa 550, el algoritmo ECC de software extendido no es satisfactorio, entonces el motor ECC 148 genera, en la etapa 555, un mensaje: "Corrección de errores no satisfactoria" y el proceso 500 finaliza.

35 La FIG. 6A muestra un diagrama de flujo que ilustra un ejemplo de un proceso 600 para generar y utilizar valores de datos alternativos. El procesador 112, el motor ECC 148, la interfaz flash 115, u otras combinaciones de los elementos anteriores y de otros elementos pueden llevar a cabo las operaciones del proceso 600. En la etapa 605, el procesador 112 recupera información de una tabla de máscaras para identificar valores de datos digitales inciertos en una página de datos (véase, por ejemplo, la FIG. 3B, etapa 330) y, en algunos casos, para recuperar información relacionada con un grado de incertidumbre.

40 Después, el procesador 112 puede, en la etapa 610, recuperar datos de corrección basándose en parámetros (por ejemplo, temperatura, número de lecturas de la página de datos, número de escritas en la página de datos, información en los registros de resolución de celda 166, tensión de suministro, tensión de bomba de carga, la tensión de referencia en la página de datos, etc.). Por ejemplo, el procesador 112 puede calcular una función de corrección para determinar los datos de corrección para la página de datos. Además, o como alternativa, el procesador 112 utiliza los datos de corrección para determinar valores digitales alternativos para cada valor de datos incierto en la etapa 615. Los valores digitales alternativos para cada valor de datos incierto incluirán normalmente el valor digital adyacente más cercano y el siguiente valor digital adyacente más cercano. También pueden incluir los valores de datos digitales dos valores de datos digitales alejados de la representación digital de la tensión analógica detectada de la celda de memoria. Normalmente, no todas las celdas de memoria tienen un valor de datos incierto. En la etapa 620, el procesador 112 almacena los valores digitales alternativos identificados en una memoria intermedia junto con valores de datos digitales almacenados para celdas de memoria que tienen valores de datos digitales ciertos.

55 Después de que se hayan almacenado los valores digitales alternativos, el procesador 112 selecciona, en la etapa 625, una combinación de valores digitales alternativos de la memoria intermedia. La combinación de valores digitales alternativos puede seleccionarse en función de un algoritmo que, por ejemplo, trata de identificar los valores digitales alternativos con mayor probabilidad de ser correctos. Este algoritmo de selección puede utilizar datos relacionados con un grado de incertidumbre asociado con cada valor de datos digital. Además, independientemente de si se usa este algoritmo de selección, la combinación seleccionada de valores digitales alternativos no necesita incluir todos los valores digitales alternativos posibles. Dicho de otro modo, incluso entre los valores de datos identificados como inciertos, algunos de los valores de datos originales pueden utilizarse junto con

algún subconjunto de valores de datos alternativos.

Después, el procesador 112 almacena los datos de página en una memoria intermedia utilizando la combinación seleccionada de valores digitales alternativos en la etapa 630 junto con los valores de datos digitales determinados con la certeza adecuada. Después, el procesador 112 ejecuta, en la etapa 635, un algoritmo ECC con los datos de página almacenados. Por ejemplo, el procesador 112 puede llevar a cabo las operaciones descritas en el proceso 500. En algunos casos, la ejecución de un algoritmo ECC puede dar como resultado cambios en uno o más de los valores digitales alternativos e incluso en uno o más de los valores de datos digitales determinados con algún grado de certeza. En la etapa 640, el procesador 112 determina si el algoritmo ECC es satisfactorio. Si el procesador 112 determina que el algoritmo ECC es satisfactorio, entonces, en la etapa 645, el procesador 112 almacena los datos de página con el resultado del ECC satisfactorio y el proceso 600 finaliza.

En la etapa 640, si el procesador 112 determina que el algoritmo ECC no es satisfactorio, entonces, en la etapa 650, el procesador 112 determina si puede probarse otra combinación de valores alternativos. El número de posibles combinaciones de valores alternativos dependerá del número de celdas de memoria con valores de datos digitales inciertos y del número de valores digitales alternativos identificados. Normalmente, la mayoría de las celdas de memoria no tienen valores de datos digitales inciertos. Si el procesador 112 determina que puede probarse otra combinación de valores alternativos, entonces el proceso vuelve a la etapa 625.

Si, en la etapa 650, el procesador 112 determina que se han probado todas las combinaciones alternativas, entonces el procesador 112 genera un mensaje de error en la etapa 655 y el proceso 600 finaliza. En algunas implementaciones, también puede ser posible generar valores alternativos adicionales y/o ajustar umbrales de tensión para leer los diferentes valores de datos y volver a ejecutar el algoritmo ECC para identificar valores correctos para los datos de página. Por ejemplo, pueden identificarse valores alternativos para niveles de tensión que se determinaron anteriormente para representar un valor particular con la certeza adecuada pero que están relativamente cerca de un umbral para una de las curvas de distribución de valores digitales 205 a 210 (descritas anteriormente con relación a la FIG. 2A). Como alternativa, los umbrales de tensión para las diversas curvas de distribución de valores digitales 205 a 210 pueden ajustarse de la manera descrita anteriormente, y los valores de datos pueden volver a generarse, incluyendo la identificación de nuevos valores alternativos.

En algunas implementaciones, el mensaje de error en la etapa 655 se registra en un registro de mantenimiento de la NVM 154. Después, durante una operación de mantenimiento, como la mostrada en la FIG. 9, también puede ser posible generar valores alternativos adicionales y/o ajustar los umbrales de tensión para leer los diversos valores de datos y volver a ejecutar el algoritmo ECC para identificar valores correctos para los datos de página. Después, los valores correctos identificados pueden usarse para reescribir los datos.

La FIG. 6B muestra un diagrama de flujo que ilustra otro ejemplo de un proceso 660 para generar y utilizar valores alternativos. El proceso 660 tiene algunas etapas comunes con el proceso 600. En este ejemplo, después de identificar valores de datos inciertos utilizando una tabla de máscaras, o de otra manera, en la etapa 605, el procesador 112 determina, en la etapa 665, un valor alternativo para cada valor incierto utilizando el valor digital adyacente más cercano. Por ejemplo, el procesador 112 puede utilizar la distribución de valores digitales 200 (FIG. 2A) y seleccionar el segundo valor digital adyacente más cercano en lugar del valor digital adyacente más cercano para la tensión de celda. Después, el procesador 112 continúa con el proceso 600 llevando a cabo las operaciones descritas con relación a la FIG. 6A, empezando por la etapa 620.

La FIG. 6C muestra un diagrama de flujo que ilustra otro ejemplo de un proceso 670 para generar y utilizar valores alternativos. En este ejemplo, el procesador 112 no recupera necesariamente información de valores digitales inciertos de la tabla de máscaras. El proceso 670 empieza en la etapa 672 cuando el procesador 112 recibe un comando para llevar a cabo una identificación de valores alternativos y un análisis de una página seleccionada (véase, por ejemplo, la FIG. 5, etapa 535).

El procesador 112 inicia, en la etapa 674, la lectura de la página seleccionada. En la etapa 676, el procesador 112 selecciona una celda de la página para leer una tensión de celda. En la etapa 678, el procesador 112 determina si la tensión de celda es incierta. Por ejemplo, el procesador 112 puede utilizar la distribución de valores digitales 200 mostrada en la FIG. 2A para determinar si la tensión de celda recibida está en una de las áreas grises 220. Si el procesador 112 determina que la tensión de celda está en el área gris, entonces el procesador 112 determina un valor de datos digital de la celda utilizando el segundo valor de datos digital más cercano en la etapa 680. En otras implementaciones, el procesador 112 determina un valor de datos digital de la celda utilizando el primer valor de datos digital más cercano. Después, el procesador 112 almacena, en la etapa 682, los valores de datos digitales en una memoria intermedia.

Si, en la etapa 678, el procesador 112 determina que la tensión de celda no está en el área gris, entonces el

5 procesador 112 determina, en la etapa 684, un valor de datos digital de la celda basándose en los umbrales almacenados y el procesador 112 lleva a cabo la etapa 682. Después de la etapa 682, en la etapa 686, el procesador 112 determina si leer otra celda de la página. Si el procesador 112 determina leer otra celda, entonces el proceso vuelve a la etapa 676. Si el procesador 112 determina que no hay más celdas que leer, entonces el proceso 670 finaliza.

10 La FIG. 7A muestra un diagrama de flujo que ilustra un ejemplo de un proceso 700 para escribir datos en la página de memoria flash 121 utilizando las celdas de referencia 130a, 130b, 130c. El proceso 700 puede llevarse a cabo generalmente mediante el procesador 112. El proceso 700 comienza en la etapa 705 cuando el procesador 112 recibe un comando de escritura. Por ejemplo, el comando de escritura puede incluir una instrucción de escritura, los datos a escribir y una dirección de memoria en la que van a escribirse los datos, que puede recibirse, por ejemplo, como una dirección de bloque lógica desde el dispositivo principal. Después, basándose en el comando de escritura, el proceso 112 selecciona una página de memoria de la memoria flash en la etapa 710.

15 Después, el procesador 112 puede copiar, en la etapa 715, los datos que van a escribirse en una memoria intermedia, tal como la SDRAM 151. Los datos pueden transferirse desde un dispositivo principal externo o desde otra página de memoria. En algunas implementaciones, los datos almacenados en la página de memoria seleccionada se copian en la memoria intermedia para volver a copiarse en la página seleccionada. En otras implementaciones, los datos que van a escribirse en una página de memoria seleccionada no se copian en la memoria intermedia, sino que se escriben directamente desde la fuente de datos (desde un dispositivo principal externo o desde otras celdas de memoria) en la página de memoria seleccionada.

25 Después, el procesador 112 borra, en la etapa 725, los datos almacenados en la página seleccionada. En la etapa 730, el procesador 112 escribe los datos de la memoria intermedia en la página de memoria seleccionada, por ejemplo, aplicando cargas a las celdas de datos 124 y a la celda de referencia 130c. La etapa 730 aplica diferentes cantidades de carga a las celdas de memoria dependiendo del valor de datos deseado y del nivel de tensión analógica correspondiente para cada celda. En algunas implementaciones puede usarse una bomba de carga para aplicar cargas a las celdas de memoria de la página de memoria seleccionada. Después, el procesador 112, en la etapa 735, lee una tensión de referencia de la celda de referencia 130c de la página seleccionada. La tensión de referencia se lee detectando un nivel de tensión en la celda de referencia 130c. El procesador 112 comprueba, en la etapa 740, si la tensión de referencia es inferior a una tensión objetivo. Si el procesador 112 determina que la tensión de referencia es inferior a la tensión objetivo, entonces el proceso vuelve a la etapa 730 para aplicar una carga adicional y aumentar la tensión almacenada en las celdas de la página de memoria seleccionada. La cantidad de carga adicional aplicada puede escalarse dependiendo del modo en el que el nivel de tensión deseado se compara mediante porcentajes con la tensión de la(s) celda(s) de referencia (por ejemplo, si la tensión de celda de referencia detectada es un 10% inferior a una tensión objetivo y una celda de memoria particular debe tener el doble de nivel de tensión que la celda de referencia, entonces la cantidad de carga adicional aplicada a la celda de memoria particular puede ser el doble que la aplicada a la celda de referencia).

40 En la etapa 740, si el procesador 112 determina que la tensión de referencia no es inferior a la tensión objetivo, entonces el procesador 112, en la etapa 745, selecciona una celda de datos y lee, en la etapa 750, la tensión de la celda de datos seleccionada. Después, en la etapa 755, el procesador 112 determina si la tensión leída es demasiado alta. Por ejemplo, el procesador 112 puede comparar la tensión de celda leída con la distribución de valores digitales y comprobar si la tensión de celda está dentro de un intervalo de tensiones del valor digital objetivo. Si el procesador 112 determina que la tensión no es demasiado alta, entonces el procesador determina, en la etapa 760, si seleccionar otra celda de datos. Si el procesador 112 determina que no es necesario seleccionar otra celda de datos, entonces el proceso 700 finaliza. En caso contrario, el proceso 700 vuelve a la etapa 745 para probar una celda de datos adicional.

50 En algunas implementaciones, también puede ser posible probar las celdas de datos seleccionadas en la etapa 745 para determinar si tienen valores demasiado bajos. Si es así, el proceso 700 puede volver a la etapa 730 para aplicar una carga adicional a una o más celdas de datos. En algunas implementaciones, una vez finalizada la comprobación de una o más celdas de referencia en la etapa 740, el nivel de tensión de todas las celdas de datos puede seleccionarse en la etapa 745 (o en repeticiones iterativas de la etapa 745) para determinar si los niveles son demasiado altos y/o demasiado bajos. De esta manera, las celdas de referencia pueden utilizarse para llevar a cabo una carga inicial de la página o bloque, seguida de una comprobación y de un posible ajuste de los niveles de tensión de las celdas. Además, en algunas implementaciones, las tensiones objetivo para las celdas de referencia utilizadas en la etapa 740 pueden fijarse con un valor algo más bajo que la tensión umbral para un valor de datos deseado con el fin de evitar una sobrecarga, seguido de una comprobación de los valores de celda real de datos y de un ajuste de los niveles de tensión para obtener niveles de tensión correspondientes a los valores de datos deseados para la celda real de datos.

Si, en la etapa 755, el procesador 112 determina que la tensión es demasiado alta, entonces el procesador 112 determina si es necesario reescribir la página seleccionada. Por ejemplo, el procesador 112 puede comparar el número de errores de bit con un umbral que es inferior o igual al número de errores corregibles utilizando uno de los algoritmos de corrección descritos con referencia a la FIG. 5. Si el número de errores de bit es mayor que el umbral, entonces la página seleccionada se reescribe. En caso contrario, el procesador 112 puede determinar que no es necesario reescribir la página seleccionada. En la etapa 765, si el procesador 112 determina que no es necesario reescribir la página, entonces el proceso 700 continúa en la etapa 760. Si el procesador 112, en la etapa 765, determina que es necesario reescribir la página, entonces el proceso 700 vuelve a la etapa 725 para reiniciar la escritura de la página de memoria. En algunas implementaciones, la tensión objetivo puede reducirse incrementalmente después de la etapa 765 para reducir la probabilidad de superar la tensión objetivo.

La FIG. 7B muestra un diagrama de flujo que ilustra un ejemplo de un proceso 770 que consigue una mayor velocidad de transferencia de datos entre un dispositivo principal y la MCP 100. El proceso 770 comienza en la etapa 772 cuando el procesador 112 recibe un comando de escritura desde un dispositivo principal. Por ejemplo, el comando de escritura de un dispositivo principal puede incluir una instrucción de escritura, los datos a escribir y una dirección de memoria en la que van a escribirse los datos, que puede recibirse, por ejemplo, como una dirección de bloque lógica desde el dispositivo principal.

Después, el procesador 112 determina si llevar a cabo un proceso de escritura rápida (por ejemplo, escribir a una resolución de celda de un solo nivel o a otra resolución relativamente baja) o utilizar un proceso de escritura que requiere más tiempo, potencia y uso del procesador para escribir a una resolución superior. En la etapa 774, el procesador 112 determina si hay algún comando de la interfaz principal para llevar a cabo una escritura rápida. En algunas implementaciones, un dispositivo principal también puede especificar la resolución de la escritura rápida. Si un dispositivo principal no especifica una escritura rápida, entonces el procesador 112 puede determinar de manera independiente si se requiere una escritura rápida. Después, la etapa 776 determina si la MCP 100 o el dispositivo principal conectado a la MCP 100 satisface condiciones de fuente de alimentación predeterminadas. En la implementación mostrada, la etapa 776 determina si el dispositivo principal recibe energía de CA. En otras implementaciones, la etapa 776 determina en cambio si una batería que suministra energía al dispositivo principal está cargada a una carga predeterminada. En algunas implementaciones, la etapa 776 determinará si una batería que suministra carga al dispositivo principal está cargada a una capacidad máxima o al menos al 90% de su capacidad. Después, la etapa 778 determina si el procesador 112 tiene un exceso de ancho de banda que satisface una condición de ancho de banda predeterminada. En algunas implementaciones, la etapa 778 se satisface si el procesador 112 está inactivo. En otras implementaciones, la etapa 778 se satisface si no se utiliza un porcentaje predeterminado de ancho de banda del procesador 112. Si se satisfacen las condiciones 776 y 778, entonces el proceso utiliza el proceso 700 para escribir en celdas de memoria a una alta resolución, etapa 780. Si no se satisface una o ambas condiciones 776 y 778, entonces el proceso lleva a cabo un procedimiento de escritura rápida.

En un procedimiento de escritura rápida, el procesador 112, en la etapa 782, selecciona una o más páginas de celda de memoria disponibles para escribir datos desde el dispositivo principal a. En algunas implementaciones, el procesador puede copiar datos desde el dispositivo principal hasta una memoria intermedia, tal como la SDRAM 151. En otras implementaciones, los datos del dispositivo principal no se copian en la memoria intermedia, sino que se escriben directamente, en la etapa 786, en la página de memoria seleccionada, después de las etapas 725 y 784. En la etapa 725, el procesador borra los datos almacenados en la(s) página(s) seleccionada(s). En la etapa 774, el procesador 112 actualiza los registros de resolución de celda asociados con la(s) página(s) de memoria seleccionada(s) a una menor resolución. En algunas implementaciones, la baja resolución será una resolución de un bit por celda de memoria. En otras implementaciones, la baja resolución será de 2, 3 ó 4 bits por celda. La escritura a una resolución más baja cuando se copian datos desde un dispositivo principal hasta la MCP 100 aumenta la velocidad de transferencia de datos porque se necesita menos precisión cuando se carga cada celda de memoria y, por tanto, el grado de cuidado y la cantidad de ajustes de tensión necesarios cuando se escribe en las celdas de memoria pueden reducirse. Después de escribirse los datos en la(s) celda(s) de memoria a una baja resolución, la etapa 784 registrará una entrada de registro de mantenimiento que indica que los datos almacenados en la página de memoria seleccionada deben reescribirse a una mayor resolución durante un proceso de mantenimiento (proceso 900).

La FIG. 8A muestra un diagrama de flujo que ilustra un ejemplo de un proceso 800 para ajustar una resolución de celda de una página de memoria. El proceso 800 puede llevar a cabo las operaciones del proceso 800 cuando, por ejemplo, el procesador 112 ejecuta un programa de mantenimiento para actualizar los registros de resolución de celda 166. El proceso 800 comienza en la etapa 805 cuando el procesador 112 lee, en la etapa 805, información de error almacenada. La información de error puede almacenarse durante errores de lectura o errores de escritura, por ejemplo como se describe en la etapa 480 de la FIG. 4. Después, el procesador 112 selecciona una página en la etapa 810. En la etapa 815, el procesador 112 determina si el cómputo de errores de la página seleccionada es mayor que un umbral. Si el cómputo de errores de la página seleccionada no es mayor que el umbral, entonces el

procesador 112 comprueba, en la etapa 820, si hay que revisar más páginas. Si el procesador 112 determina que no hay más páginas que revisar, entonces el proceso 800 finaliza. Si, en la etapa 820, el procesador 112 determina que hay más páginas que revisar, entonces el proceso vuelve a la etapa 810. En algunas implementaciones, el procesador 112 puede revisar todas las páginas de memoria con errores. En otras implementaciones, el procesador 112 sólo puede revisar páginas de memoria con nuevos errores registrados en la información de error.

En la etapa 815, si el cómputo de errores de la página seleccionada es mayor que el umbral, entonces el procesador 112 copia una página de datos de la página seleccionada en una memoria intermedia en la etapa 825. Después, el procesador 112 actualiza los registros de resolución de celda 166 para reducir la resolución de celda de la página seleccionada. Por ejemplo, la interfaz flash 115 puede revisar los registros de resolución de celda 166 para determinar que la resolución de celda se ha reducido y, después, la interfaz flash 115 puede leer y escribir en la página seleccionada utilizando la nueva resolución de celda reducida.

Después, el procesador 112 puede asignar, en la etapa 835, direcciones físicas para los datos copiados. Dependiendo de las páginas de memoria disponibles, el procesador 112 puede asignar una, dos, cuatro u otro número de páginas de memoria física para almacenar los datos copiados. Después, el procesador 112 actualiza, en la etapa 840, una tabla de direcciones lógicas para corresponder con las direcciones físicas asignadas. La tabla de direcciones lógicas puede utilizarse para mapear una página lógica con una o más páginas físicas. Un ejemplo de uso de la tabla de direcciones lógicas durante una operación de acceso a memoria se describe con referencia a la FIG. 10. En la etapa 845, el procesador 112 mueve los datos copiados desde la memoria intermedia hasta las páginas en las direcciones físicas asignadas. Después, el procesador 112 determina, en la etapa 820, si hay más páginas que revisar. Si es así, el proceso 800 vuelve a la etapa 810. En caso contrario, el proceso 800 finaliza.

La FIG. 8B ilustra un proceso similar al de la FIG. 8A destinado a degradar grupos de páginas o bloques de celdas de memoria y a tratar el grupo de manera lógica como una única página o bloque de celdas de memoria que tiene la resolución original. En la FIG. 8B, el proceso 860 también lee información de error almacenada 805, selecciona una página 810 y determina si un cómputo de errores asociado con la página supera un umbral 815. Si el cómputo de errores asociado con la página supera un umbral, entonces los datos almacenados en la página se copian en una memoria intermedia 825 y el (los) registro(s) de resolución de celda asociado(s) con la página se actualiza(n) para reducir la resolución de la página 830. Sin embargo, en la implementación ilustrada en la FIG. 8B, un procesador también selecciona otra página de datos que tiene una resolución de celda reducida 855 y actualiza el código de gestión de bloques y/o el código de direccionamiento lógico para emparejar las dos páginas. Las dos páginas que tienen una resolución de celda reducida se tratan después de manera lógica como una única página con la resolución original superior. Este proceso puede agrupar más de dos páginas de celdas de memoria.

En algunas implementaciones, este proceso degradará bloques enteros de celdas de memoria y los emparejará o asociará de otra manera. En algunas implementaciones, cada página emparejada tendrá la misma resolución de celda ajustada a la baja e incluirá el mismo número de celdas de memoria. Por ejemplo, una página de celdas de memoria degradada desde celdas de memoria que almacenan 8 bits de datos hasta celdas de memoria que almacenan 4 bits de datos, se agrupa con otra página de celdas de memoria en la que cada celda de memoria almacena 4 bits de datos. La combinación de estas dos páginas de celdas de memoria se trata después de manera lógica por el controlador de disco Hash como una única página (o como un único bloque) que almacena 8 bits de datos por celda de memoria. Estas páginas emparejadas de celdas de memoria no necesitan estar en el mismo bloque y pueden estar posiblemente en diferentes chips de memoria flash. Después, el proceso 860 lleva a cabo la etapa 820 que determina si hay más páginas de memoria que revisar y procede de la manera descrita en la FIG. 8A.

La FIG. 9 es un diagrama de flujo que ilustra un proceso de mantenimiento 900. Una posible función del proceso de mantenimiento 900 es reescribir, a una resolución relativamente alta, datos almacenados en una memoria flash a una resolución relativamente baja (por ejemplo, véase la FIG. 7B). Los procesos de mantenimiento pueden utilizarse, por ejemplo, para maximizar la vida de la batería de un dispositivo principal a la vez que se maximiza la capacidad de almacenamiento de datos. En algunas implementaciones, el proceso de mantenimiento 900 se activa por el procesador como parte de una operación de mantenimiento planificada de manera rutinaria. En algunas implementaciones, el proceso de mantenimiento 900 se activa por una señal procedente de un dispositivo principal que indica que el dispositivo principal está alimentado con energía de CA. En otras implementaciones, otras condiciones pueden hacer que un dispositivo principal o que el procesador 112 activen el proceso de mantenimiento 900, tal como un procesador inactivo 112.

El proceso 900 comienza con la etapa 905, que puede determinar si la MCP 100 está funcionando bajo una condición de energía predeterminada. En algunas implementaciones, esta condición de energía se satisface por un dispositivo principal que recibe energía de CA. En algunas implementaciones, esta condición de energía se satisface por una batería de dispositivo principal que satisface una cantidad de carga predeterminada, por ejemplo

que la batería está totalmente cargada. Una batería totalmente cargada puede indicar que un dispositivo principal está siendo alimentado con energía de CA. Si la MCP 100 no satisface la condición de energía predeterminada, el proceso 900 finaliza.

- 5 Después, en la etapa 910, el procesador 112 puede determinar si el procesador 112 tiene un ancho de banda suficiente como para llevar a cabo completamente el proceso de mantenimiento 900. En algunas implementaciones, la operación de mantenimiento simplemente se ejecuta como un proceso en segundo plano que requiere un ancho de banda mínimo. En algunas implementaciones, la operación de mantenimiento necesita un procesador inactivo 112. En otras implementaciones, el proceso 900 no determina si el procesador 112 tiene un ancho de banda
- 10 suficiente. En algunas implementaciones, el requisito de ancho de banda cambia en función de la necesidad de las operaciones de mantenimiento, lo que puede medirse por el tiempo entre procesos de mantenimiento satisfactorios o por la cantidad de espacio disponible en la memoria flash. Si el procesador 112 no tiene el suficiente ancho de banda, el proceso 900 finaliza.
- 15 Si se satisface la condición de energía predeterminada y el procesador 112 tiene un ancho de banda suficiente, el proceso 900 puede leer entonces registros de mantenimiento almacenados, etapa 915. En algunas implementaciones, los registros de mantenimiento almacenados están almacenados en la NVM 154. En algunas implementaciones, los registros de mantenimiento almacenados indican la prioridad de posibles operaciones de mantenimiento. En algunas implementaciones, los registros de mantenimiento almacenados se utilizan para
- 20 determinar si alguna de las etapas de mantenimiento (tales como las etapas 920, 925, 930 y 935) puede llevarse a cabo en una operación simplificada. Por ejemplo, los registros de mantenimiento pueden indicar la necesidad de degradar una página particular de celdas de memoria y de reescribir los datos en la misma página de celdas de memoria. En otras implementaciones, las operaciones de mantenimiento son una secuencia predeterminada de,
- 25 por ejemplo, reescribir datos transferidos a una resolución más alta (etapa 920); degradar resoluciones de celda y emparejar grupos de páginas, por ejemplo, llevando a cabo los procesos 800 y 850 (etapa 925); reescribir datos que satisfagan una condición de error predeterminada (por ejemplo, usando el proceso 700) (etapa 930); intercambiar los datos a los que se accede con más frecuencia por los datos usados con menor frecuencia utilizando el código software de gestión de deterioro 157 (etapa 935); actualizar el código software de direccionamiento lógico 163 para cada operación de mantenimiento que movió datos desde una posición física
- 30 hasta otra posición física (por ejemplo, usando el proceso 1000) (etapa 940); y refrescar páginas de datos que superan una cantidad umbral de variación de tensión aplicando una carga adicional a la página de celdas de memoria (etapa 945). También pueden usarse otras secuencias que incluyen todas, algunas u otras operaciones adicionales. En algunas implementaciones, el proceso 900 repite la etapa 905 y/o la etapa 910 entre cada etapa de mantenimiento 920, 925, 930, 935 ó 945, y puede finalizar si cambia alguna de las condiciones 905 ó 910.
- 35 Después, el proceso 900 finaliza.

La FIG. 10 es un diagrama de flujo que ilustra un ejemplo de un proceso 1000 de direccionamiento lógico en el FDC 106. Por ejemplo, el FDC 106 puede mapear un comando de lectura o de escritura recibido que incluye una dirección lógica con una o más páginas físicas. En algunas implementaciones, el FDC 106 puede mapear de

40 manera dinámica una página lógica con una o más páginas físicas variables. Por ejemplo, el FDC 106 puede modificar el mapeo para equilibrar la carga de una página de memoria física. En algunas implementaciones, el mapeo entre páginas lógicas y páginas físicas puede almacenarse en una tabla de direcciones lógicas. En algunas implementaciones, el proceso 1000 puede llevarse a cabo por el procesador 112 cuando el procesador 112 está ejecutando el código de direccionamiento lógico 163. El proceso 1000 comienza cuando el FDC recibe un comando

45 desde el dispositivo principal que indica que va a accederse a una página de memoria (por ejemplo, leerse, escribir en la misma o borrarse). Después, en la etapa 1005, el procesador 112 recibe una dirección de página lógica para acceder a una página de la memoria flash.

Después, el procesador 112 determina, en la etapa 1010, una o más direcciones de página físicas asociadas con la

50 dirección lógica recibida. En un ejemplo, la dirección de página lógica recibida puede asociarse solamente con una dirección de página física. En otro ejemplo, la dirección de página lógica recibida puede asociarse con dos o más páginas físicas porque las páginas físicas tienen una menor resolución de celda de lo normal, o las páginas físicas no son contiguas en la memoria flash, o están en diferentes bloques o en chips diferentes.

- 55 Después, el procesador 112 selecciona en la etapa 1015 una primera de las direcciones de página físicas determinadas. En la etapa 1020, el procesador 112 lee los datos de página en la dirección física seleccionada. Después, el procesador 112 almacena, en la etapa 1025, los datos de página en la memoria intermedia principal de salida. En la etapa 1030, el procesador 112 determina si es necesario acceder a otra página de memoria. Por ejemplo, si hay más de una dirección de página física asociada con la dirección de página lógica, entonces el
- 60 procesador 112 puede acceder a otra página de memoria. Si, en la etapa 1030, el procesador 112 determina que es necesario acceder a otra página de memoria, entonces el procesador, en la etapa 1035, selecciona la siguiente dirección de página física determinada y el proceso vuelve a la etapa 1020. En caso contrario, el proceso 1000

finaliza.

La FIG. 11 muestra un sistema de ejemplo 1100 que incluye múltiples chips de memoria flash NAND 103 y el FDC 106. El FDC 106 incluye un multiplexor (MUX) 1105 en la interfaz analógica 115 y una bomba de carga 1110. Aunque el sistema 1100 se muestra utilizando chips de memoria flash NAND 103, algunas de las técnicas utilizadas en el sistema 1100 también pueden aplicarse a chips de memoria flash NOR, o a una combinación de chips NAND y NOR. El sistema 1100 puede implementarse utilizando CI discretos, o puede estar integrado total o parcialmente en una sola cápsula.

El FDC 106 recibe datos analógicos desde los chips de memoria flash NAND 103 a través de la interfaz analógica 115. En este ejemplo, el MUX 1105 recibe múltiples entradas analógicas. En algunas implementaciones, el MUX 1105 recibe las múltiples entradas analógicas desde múltiples chips de memoria flash 103. La interfaz analógica 115 puede controlar el MUX 1105 para seleccionar una entrada analógica que va a transmitirse al ADC 142. Por ejemplo, la interfaz analógica 115 puede controlar el MUX 1105 en función de un comando de lectura recibido. Durante una operación de escritura, el FDC 106 utiliza la bomba de carga 1110 para aplicar cargas a las celdas de memoria en uno de los chips de memoria flash NAND 103. En algunas implementaciones, la bomba de carga 1110 está adaptada para suministrar carga a celdas de memoria de una pluralidad de chips de memoria flash 103. Por ejemplo, el FDC 106 puede enviar una señal de control para seleccionar un chip de memoria designado para recibir cargas desde la bomba de carga 1110. Después, cuando la bomba de carga 1110 aplica cargas, el chip de memoria seleccionado recibe las cargas.

Compartiendo el ADC 1105 y la bomba de carga 1110 entre múltiples chips 103, el tamaño de almacenamiento de los chips de memoria 103 puede aumentar. Además, los chips de memoria flash 103 pueden fabricarse con un menor coste sin el ADC 142 y la bomba de carga 1110. En algunas implementaciones, la bomba de carga 1110 puede estar integrada en un chip con el FDC 106 o puede estar montada por separado en un chip diferente o en un sustrato diferente, tal como una placa de circuito impreso.

Con el fin de facilitar el uso de un ADC 1105 y de una bomba de carga 1110 adaptados para utilizarse con múltiples chips de memoria flash 103, algunos chips de memoria 103 pueden incluir una entrada adaptada para recibir una carga de programación desde un nodo de suministro externo. De este modo no es necesario que los chips de memoria flash 103 incluyan un sistema de circuitos adicional para alterar o regular la carga de programación suministrada. Los chips de memoria flash 103 también pueden incluir una salida adaptada para enviar una señal de tensión analógica a un controlador de disco flash 106.

En algunas implementaciones, el FDC 106 también puede incluir un procedimiento de intercalado de bomba de carga para escribir datos en los chips de memoria 103.

La FIG. 12 muestra un sistema de ejemplo 1200 que ilustra una arquitectura para proporcionar por separado potencia de nivel lógico y de programación al chip de memoria flash NAND 103. El sistema 1200 incluye la bomba de carga 1110 y un regulador de pequeña pérdida de información (LDO) 1205 que reciben energía eléctrica desde una fuente de alimentación 1210.

Tal y como se muestra, el chip de memoria flash NAND 103 incluye dos entradas de energía. Una entrada de energía para la tensión de la bomba de carga (V_{cp}) y una entrada de energía para la tensión lógica (V_{logic}). En algunos ejemplos, la V_{cp} puede ser sustancialmente mayor que la V_{logic} . Por ejemplo, la V_{cp} puede estar comprendida aproximadamente entre 12 y 20 V o aproximadamente entre 12 y 30 V, y la V_{logic} puede estar comprendida aproximadamente entre 1 y 3 V. En algunas implementaciones, los requisitos de regulación y de corriente para la V_{cp} pueden ser sustancialmente diferentes de los requisitos para la V_{logic} .

Como un ejemplo, el chip de memoria flash NAND 103 puede requerir que la V_{logic} tenga una tolerancia a la tensión estrechamente regulada (por ejemplo, 0,5%, 1,0%, 5%) a una baja tensión lógica para minimizar el consumo de energía, los tiempos de conmutación, etc. Además, la tensión lógica puede requerir una capacitancia de derivación de alta frecuencia a un bajo nivel de tensión. Por el contrario, los requisitos de regulación de suministro de la bomba de carga pueden estar entre el 5% y el 10% aproximadamente, con la necesidad de una capacitancia de mayor tensión y de frecuencia sustancialmente baja.

Con el fin de facilitar el sistema de la FIG. 12, el chip de memoria flash 103 puede incluir una primera interfaz para recibir energía para programar de manera selectiva cada celda de memoria flash, y una segunda interfaz para recibir energía suministrada al sistema de circuitos de nivel lógico para llevar a cabo la selección de celdas de memoria flash que van a alimentarse con energía procedente de la primera entrada durante una operación de escritura. El controlador de disco flash 106 puede comprender una primera fuente de alimentación para suministrar energía a la primera interfaz a una tensión de programación y una segunda fuente de alimentación para suministrar

energía de nivel lógico a la segunda interfaz. La primera y la segunda fuente de alimentación pueden ser externas al chip de memoria flash 103.

5 Aunque se han descrito varias implementaciones de procesos y de técnicas, otras implementaciones pueden llevar a cabo las etapas en una secuencia diferente o en una disposición modificada para lograr la misma función principal. Además, aunque en ocasiones se ha descrito que las operaciones de los diversos procesos se llevan a cabo por un dispositivo o componente particular, tales dispositivos o componentes son simplemente ejemplos, y las operaciones pueden llevarse a cabo utilizando dispositivos o componentes alternativos en alguna implementación.

10 En algunos ejemplos, el chip de memoria flash NAND 103 también puede tener cualquier número práctico de bits de resolución, tal como, por ejemplo, 6, 7, 10, 12 bits de resolución. Varias implementaciones pueden utilizarse para llevar a cabo operaciones ECC con una memoria flash que puede incluir una memoria flash NAND, una memoria flash NOR o una combinación de las mismas u otras memorias no volátiles. Chips de memoria flash de uno o más tipos pueden estar apilados y/o montados de manera adyacente entre sí en la MCP 100. Los expertos en la técnica
15 reconocerán que algunos ejemplos de las técnicas descritas en este documento pueden aplicarse de una manera particularmente ventajosa con tecnología flash NAND, y que algunos procedimientos descritos en este documento pueden aplicarse en general a memorias no volátiles tales como flash NAND y/o NOR.

20 Aunque un ejemplo de un sistema, que puede ser portátil, se ha descrito con referencia a las figuras anteriores, otras implementaciones pueden implantarse en otras aplicaciones de procesamiento, tales como instalaciones de escritorio y conectadas en red.

Aunque se han descrito características particulares de una arquitectura, otras características pueden incorporarse para mejorar el rendimiento. Por ejemplo, pueden utilizarse técnicas de almacenamiento temporal (por ejemplo, L1, L2, etc.) en el FDC 106. Pueden incluirse memorias de acceso aleatorio, por ejemplo, para proporcionar una memoria de trabajo temporal y/o cargar código ejecutable o información de parámetro almacenada en la memoria flash para su utilización durante operaciones en tiempo de ejecución. Puede proporcionarse otro hardware y software para llevar a cabo operaciones, tales como comunicaciones de red o de otro tipo que utilizan uno o más protocolos, comunicaciones inalámbricas (por ejemplo, infrarrojas), fuentes de alimentación y de energía operativa
25 almacenada (por ejemplo baterías), circuitos de fuente de alimentación lineales y/o de conmutación, mantenimiento de software (por ejemplo, autoverificaciones, actualizaciones, etc.) y similares. Pueden proporcionarse una o más interfaces de comunicación para dar soporte a operaciones de almacenamiento de datos y otras operaciones relacionadas.

35 En algunas implementaciones puede utilizarse un procedimiento o una combinación de procedimientos para mejorar la integridad de los datos. Por ejemplo, los errores de tensión de celda pueden tratarse ajustando los umbrales y/o reescribiendo las celdas al menos una vez. La reescritura de las celdas puede llevarse a cabo en respuesta a desviaciones con respecto a una tensión de celda ideal y/o como una actividad en segundo plano. Por ejemplo, las tensiones de celda de múltiples niveles pueden reescribirse para refrescar la tensión en una o más celdas con tendencia a pérdidas en una página. Para celdas caracterizadas por ser propensas a perder tensión a lo largo de tiempo, el nivel de tensión a la que tales celdas se cargan puede aumentarse hasta casi alcanzar un umbral superior de la tensión de cada celda para compensar pérdidas de carga anticipadas en tales celdas a lo largo del tiempo. El nivel de tensión aumentado puede ser inicialmente aproximado o mayor que el umbral superior del intervalo previsto, el cual puede estar en una zona gris entre intervalos. En función de las tasas de pérdida
40 estimadas o determinadas, los datos pueden reescribirse con la suficiente frecuencia como para mantener sustancialmente las tensiones de celda dentro de un intervalo deseado. Una compensación similar puede utilizarse para compensar celdas caracterizadas por tener una variación creciente. Tales procedimientos de reescritura pueden llevarse a cabo, por ejemplo, como un proceso en segundo plano de baja prioridad que se ejecuta cuando los recursos están disponibles. Para los datos identificados como datos de valor elevado, puede planificarse que la reescritura se lleve a cabo con la suficiente frecuencia como para mantener las tensiones de celda dentro de un intervalo deseado, estando basada la frecuencia en una tasa de variación de tensión esperada y en el tamaño del intervalo de tensiones asociado con cada nivel de bits. En algunas implementaciones, la reescritura puede configurarse para llevarse a cabo más frecuentemente cuando un dispositivo portátil está acoplado a una fuente de alimentación externa, tal como una fuente de alimentación acoplada a la red eléctrica. Las operaciones de reescritura pueden llevarse a cabo en respuesta al acoplamiento a una fuente de alimentación de este tipo.
45 Además, la reescritura puede configurarse para llevarse a cabo con menos frecuencia en determinadas condiciones, tales como, por ejemplo, durante un modo de conservación de energía, durante una condición de baja batería o cuando se almacenan datos de corta duración o no críticos (por ejemplo, flujo continuo de audio/vídeo).

60 Algunos sistemas pueden implementarse como un sistema informático que puede utilizarse con implementaciones de la invención. Por ejemplo, varias implementaciones pueden incluir sistemas de circuitos digitales y/o analógicos, hardware informático, firmware, software o combinaciones de los mismos. Un aparato puede implementarse en un

- producto de programa informático realizado de manera tangible en un portador de información, por ejemplo, en un dispositivo de almacenamiento legible por máquina o en una señal propagada, para su ejecución mediante un procesador programable; y los procedimientos pueden llevarse a cabo mediante un procesador programable que ejecuta un programa de instrucciones para llevar a cabo funciones de la invención que se efectúan sobre datos de entrada y que generan una salida. La invención puede implementarse de manera ventajosa en uno o más programas informáticos que pueden ejecutarse en un sistema programable que incluye al menos un procesador programable acoplado para recibir datos e instrucciones desde, y para transmitir datos e instrucciones a, un sistema de almacenamiento de datos, al menos un dispositivo de entrada y/o al menos un dispositivo de salida. Un programa informático es un conjunto de instrucciones que puede utilizarse, directa o indirectamente, en un ordenador para llevar a cabo una determinada actividad o proporcionar un determinado resultado. Un programa informático puede escribirse en cualquier forma de lenguaje de programación, incluyendo lenguajes compilados o interpretados, y puede implantarse de cualquier forma, incluyendo un programa autónomo o un módulo, componente, subrutina u otra unidad adecuada que pueda utilizarse en un entorno informático.
- 15 Procesadores adecuados para la ejecución de un programa de instrucciones incluyen, a modo de ejemplo, microprocesadores tanto de propósito general como de propósito especial, que pueden incluir un único procesador o uno de múltiples procesadores de cualquier tipo de ordenador. Generalmente, un procesador recibirá instrucciones y datos desde una memoria de solo lectura, desde una memoria de acceso aleatorio o desde ambas. Los elementos esenciales de un ordenador son un procesador para ejecutar instrucciones y una o más memorias para almacenar instrucciones y datos. Generalmente, un ordenador incluirá además, o estará acoplado de manera operativa para comunicarse con, uno o más dispositivos de almacenamiento masivos para almacenar archivos de datos; tales dispositivos incluyen discos magnéticos, tales como discos duros internos y discos extraíbles, discos magneto-ópticos y discos ópticos. Dispositivos de almacenamiento adecuados para almacenar de manera tangible instrucciones y datos de programa informático incluyen todas las formas de memoria no volátil incluyendo, a modo de ejemplo, dispositivos de memoria de semiconductor, tales como EPROM, EEPROM y dispositivos de memoria flash, discos magnéticos, tales como discos duros internos y discos extraíbles, discos magneto-ópticos y discos CD-ROM y DVD-ROM. El procesador y la memoria pueden complementarse por, o estar incorporados en, ASIC (circuitos integrados de aplicación específica).
- 30 En algunas implementaciones, cada sistema 100 puede estar programado con la misma información o con otra similar y/o puede inicializarse con sustancialmente la misma información almacenada en memoria volátil y/o no volátil. Por ejemplo, una interfaz de datos puede estar configurada para llevar a cabo una configuración automática, una descarga automática y/o funciones de actualización automáticas cuando está acoplada a un dispositivo principal apropiado, tal como un ordenador de escritorio o un servidor.
- 35 En algunas implementaciones, una o más características de interfaz de usuario pueden configurarse de manera personalizada para llevar a cabo funciones específicas. La invención puede implementarse en un sistema informático que incluye una interfaz gráfica de usuario y/o un navegador de Internet. Para proporcionar interacción con un usuario, algunas implementaciones pueden implementarse en un ordenador que tenga un dispositivo de visualización, tal como un monitor CRT (tubo de rayos catódicos) o LCD (pantalla de cristal líquido) para mostrar información al usuario, un teclado y un dispositivo de puntero, tal como un ratón o una bola de seguimiento, mediante los cuales el usuario puede proporcionar datos de entrada al ordenador.
- 40 En varias implementaciones, el sistema 100 puede comunicarse utilizando procedimientos, equipos y técnicas de comunicación adecuados. Por ejemplo, el sistema 100 puede comunicarse con dispositivos compatibles (por ejemplo, dispositivos capaces de transferir datos a y/o desde el sistema 100) utilizando una comunicación punto a punto en la que un mensaje se transporta directamente desde el emisor hasta el receptor a través de un enlace físico dedicado (por ejemplo, un enlace de fibra óptica, un cableado de punto a punto, una cadena fragmentada). Los componentes del sistema pueden intercambiar información mediante cualquier forma o medio de comunicación de datos analógico o digital, incluyendo mensajes basados en paquetes en una red de comunicaciones. Ejemplos de redes de comunicaciones incluyen, por ejemplo, una LAN (red de área local), una WAN (red de área extensa), una MAN (red de área metropolitana), redes inalámbricas y/u ópticas, y los ordenadores y redes que forman Internet. Otras implementaciones pueden transportar mensajes mediante radiodifusión a todos o sustancialmente todos los dispositivos que están acoplados entre sí por una red de comunicaciones, por ejemplo, usando señales de radiofrecuencia (RF) omnidireccionales. Otras implementaciones adicionales pueden transportar mensajes caracterizados por una alta directividad, tales como señales de RF transmitidas usando antenas direccionales (es decir, un haz estrecho) o señales infrarrojas que pueden utilizarse opcionalmente con óptica de enfoque. Otras implementaciones adicionales son posibles utilizando interfaces apropiadas y protocolos tales como, a modo de ejemplo y de manera no limitativa, USB 2.0, Firewire, ATA/IDE, RS-232, RS-422, RS-485, 802.11 a/b/g, Wi-Fi, Ethernet, IrDA, FDDI (interfaz de datos distribuidos por fibra), redes de anillo con testigo o técnicas de multiplexación basadas en división de frecuencia, tiempo o código. Algunas implementaciones pueden incluir opcionalmente características tales como comprobación y corrección de errores (ECC) para la integridad de los

datos, o medidas de seguridad, tales como cifrado (por ejemplo, WEP) y protección mediante contraseña.

5 Se han descrito varias implementaciones de la invención. Sin embargo, debe entenderse que pueden llevarse a cabo varias modificaciones sin apartarse del alcance de la invención. Por ejemplo, pueden obtenerse resultados ventajosos si las etapas de las técnicas divulgadas se llevan a cabo en una secuencia diferente, si los componentes de los sistemas divulgadas se combinan de diferente manera o si los componentes se sustituyen o se complementan con otros componentes. Las funciones y procesos (incluyendo algoritmos) pueden llevarse a cabo en hardware, software o en una combinación de los mismos, y algunas implementaciones pueden llevarse a cabo en módulos o en hardware diferentes a los descritos.

10

REIVINDICACIONES

1. Un artículo de fabricación que comprende instrucciones legibles por máquina que, cuando se ejecutan, hacen que se lleven a cabo operaciones, **caracterizado por que** las operaciones comprenden:
 - 5 determinar si llevar a cabo una operación de mantenimiento donde la determinación de si llevar a cabo la operación de mantenimiento comprende determinar si un dispositivo principal tiene una fuente de alimentación (1210) que satisface una condición predeterminada;
 - 10 identificar información de error asociada con una página de celdas de memoria (121) en respuesta a determinar que debe llevarse a cabo una operación de mantenimiento;
 - determinar si la información de error satisface un criterio de error; y
 - 15 en respuesta a determinar que la fuente de alimentación satisface la condición predeterminada, ajustar uno o más registros de resolución (166) correspondientes a la página de celdas de memoria (121) desde una primera resolución hasta una segunda resolución, definiendo cada una de la primera y la segunda resolución una pluralidad de intervalos de tensión, correspondiendo cada intervalo de tensión a un valor de datos posible, presentando la primera resolución más intervalos de tensión que la segunda resolución.
 - 20
2. El artículo de fabricación según la reivindicación 1, en el que las operaciones comprenden además:
 - llevar a cabo una actividad de mantenimiento registrada en un registro de mantenimiento asociado con la página de celdas de memoria (121), donde la actividad de mantenimiento se selecciona a partir del grupo que consiste en: reescribir valores de datos almacenados originalmente a una resolución inferior a una resolución superior; e
 - intercambiar datos a los que se accede con más frecuencia por datos a los que se accede con menos frecuencia.
 - 30
3. El artículo de fabricación según la reivindicación 2, en el que las operaciones comprenden además actualizar códigos software de direccionamiento lógico para valores de datos reescritos en diferentes celdas de memoria física durante una o más de las actividades de mantenimiento.
- 35 4. El artículo de fabricación según cualquier reivindicación anterior, en el que la condición predeterminada comprende que el dispositivo principal reciba energía de CA.
5. El artículo de fabricación según una cualquiera de las reivindicaciones 1 a 3, en el que la condición predeterminada comprende que el dispositivo principal tenga una batería cargada a un nivel de carga predeterminado.
- 40
6. El artículo de fabricación según una cualquiera de las reivindicaciones 1 a 3, en el que la determinación de si llevar a cabo la operación de mantenimiento comprende determinar si un procesador (112) tiene un ancho de banda no utilizado que supera un umbral predeterminado.
- 45
7. Un sistema para el almacenamiento de datos, comprendiendo el sistema:
 - una pluralidad de celdas de memoria (121), donde cada celda de memoria está adaptada para recibir carga durante una operación de escritura a un nivel de tensión correspondiente a un valor de datos que tiene un número de bits especificado;
 - 50 un registro de resolución (166) asociado con la pluralidad de celdas de memoria (121), donde el registro de resolución incluye entradas que indican cada una un número de bits almacenados en una o más celdas de memoria correspondientes; **caracterizado por**
 - 55 una interfaz principal (109) adaptada para recibir señales desde un dispositivo principal que indican una condición de fuente de alimentación para el dispositivo principal; y
 - un procesador (112) adaptado para reescribir valores de datos en la pluralidad de celdas de memoria (121) y para ajustar los registros de resolución (166) para pasar de indicar un primer número de bits a indicar un segundo número de bits en respuesta a una señal que indica una condición de fuente de alimentación predeterminada.
 - 60

8. El sistema según la reivindicación 7, que comprende además:
 - 5 un código software de direccionamiento lógico para convertir las direcciones lógicas recibidas desde el dispositivo principal en direcciones físicas con el fin de acceder a los datos.
9. El sistema según la reivindicación 7 u 8, en el que la interfaz principal (109) está adaptada además para recibir comandos desde un dispositivo principal y para intercambiar datos con el dispositivo principal.
- 10 10. El sistema según una cualquiera de las reivindicaciones 7 a 9, en el que la pluralidad de celdas de memoria (121) comprende celdas de memoria flash NAND.

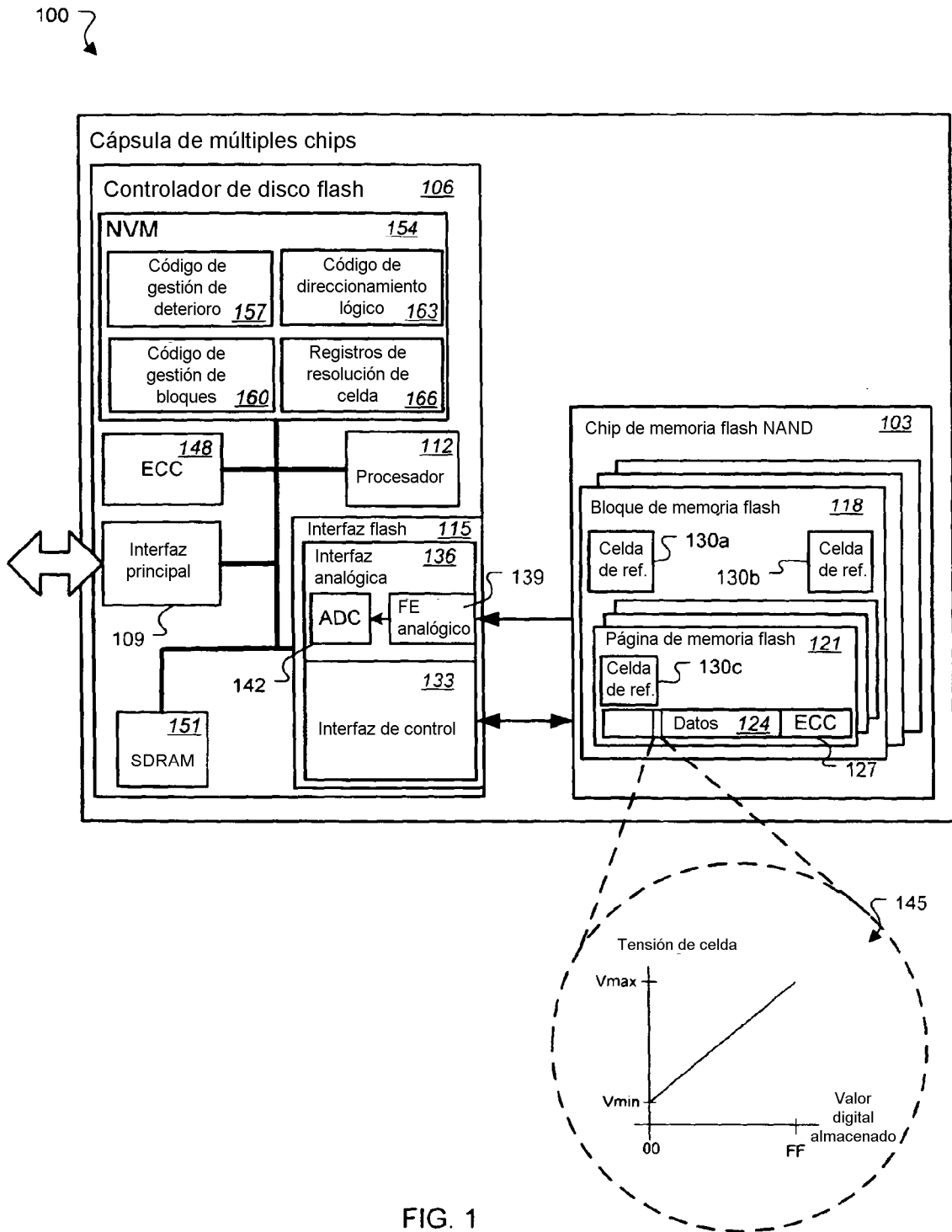


FIG. 1

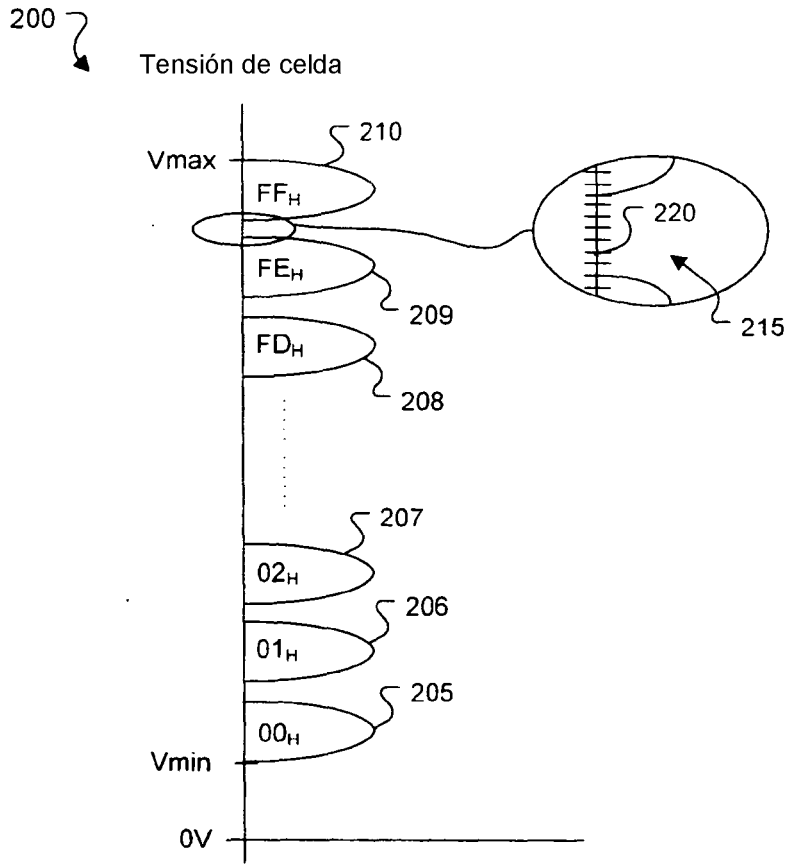


FIG. 2A

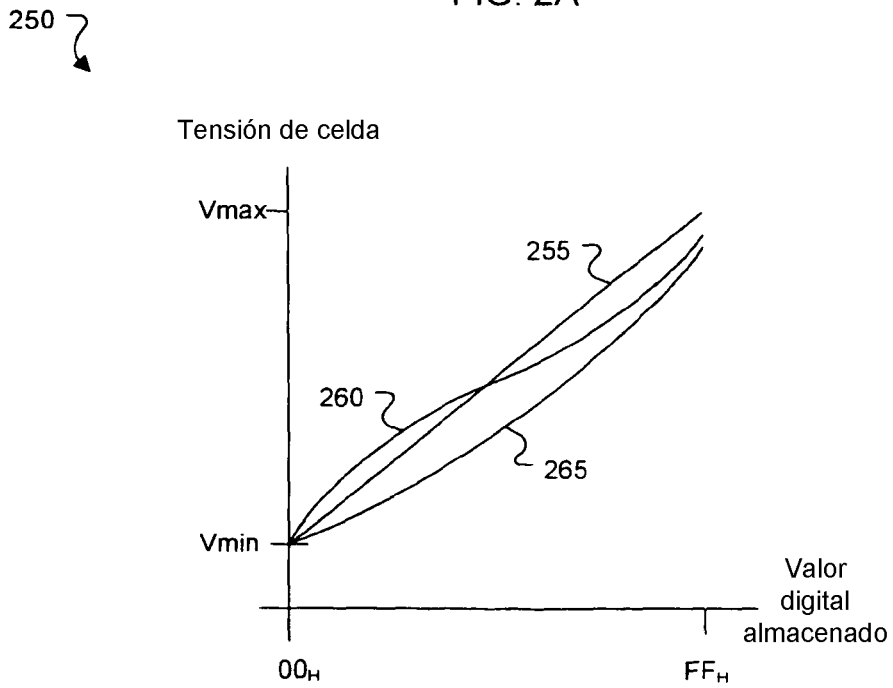


FIG. 2B

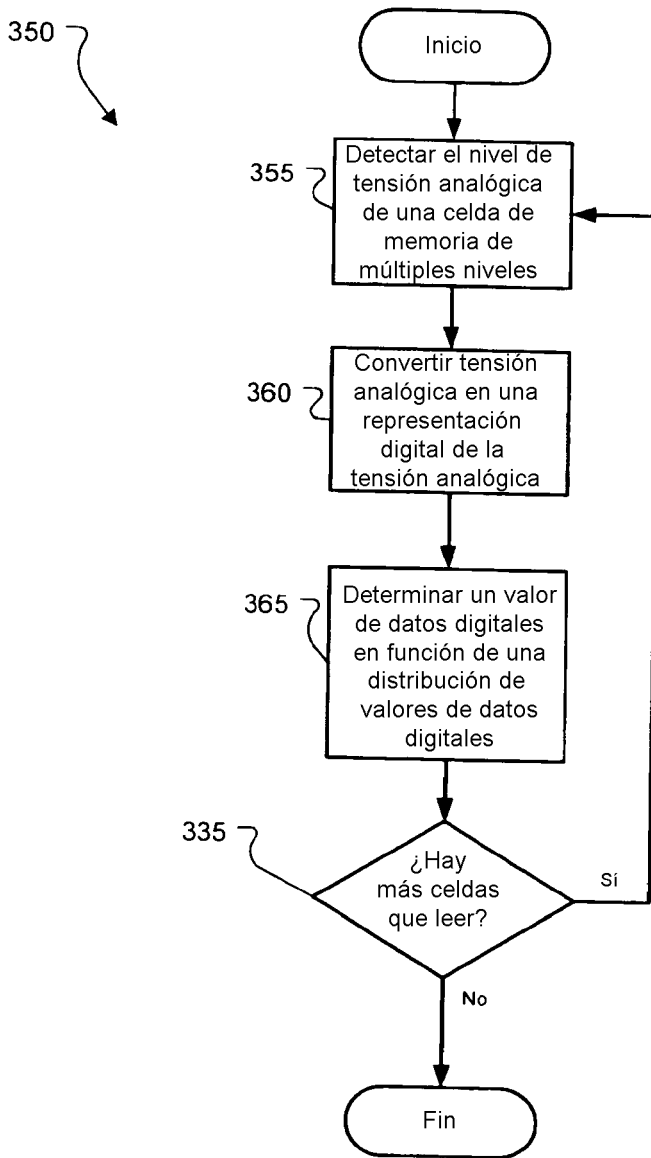


FIG. 3A

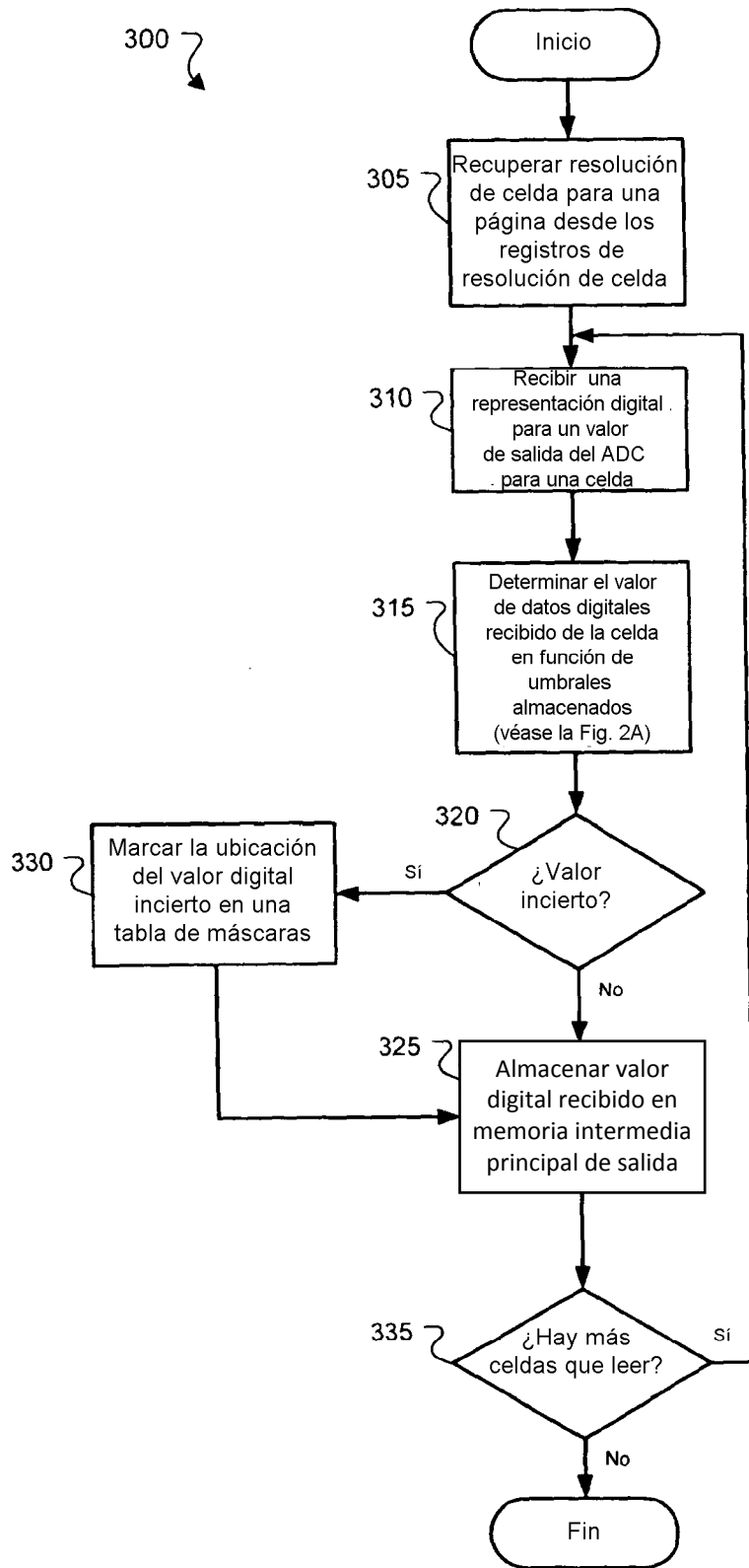


FIG. 3B

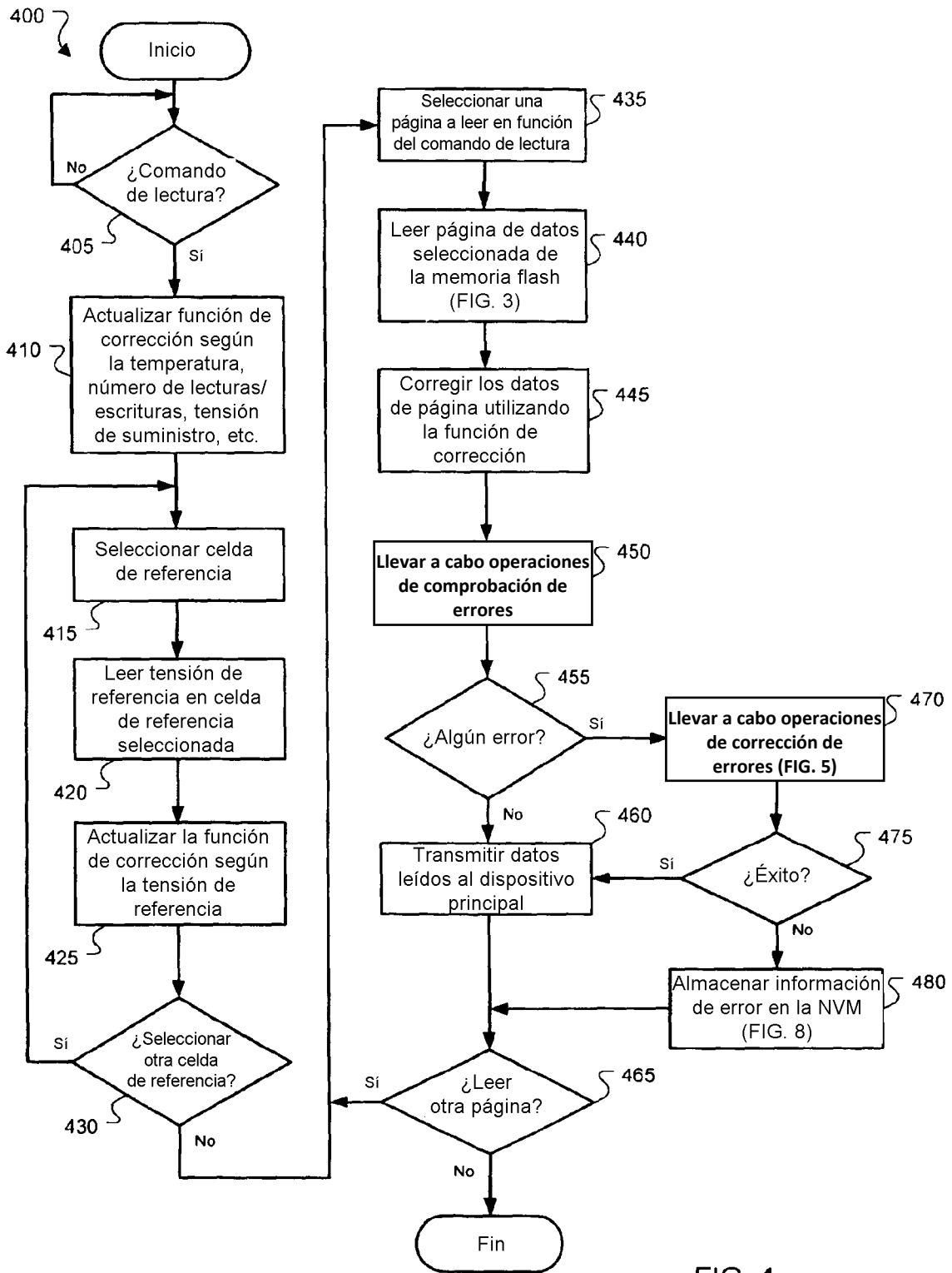


FIG. 4

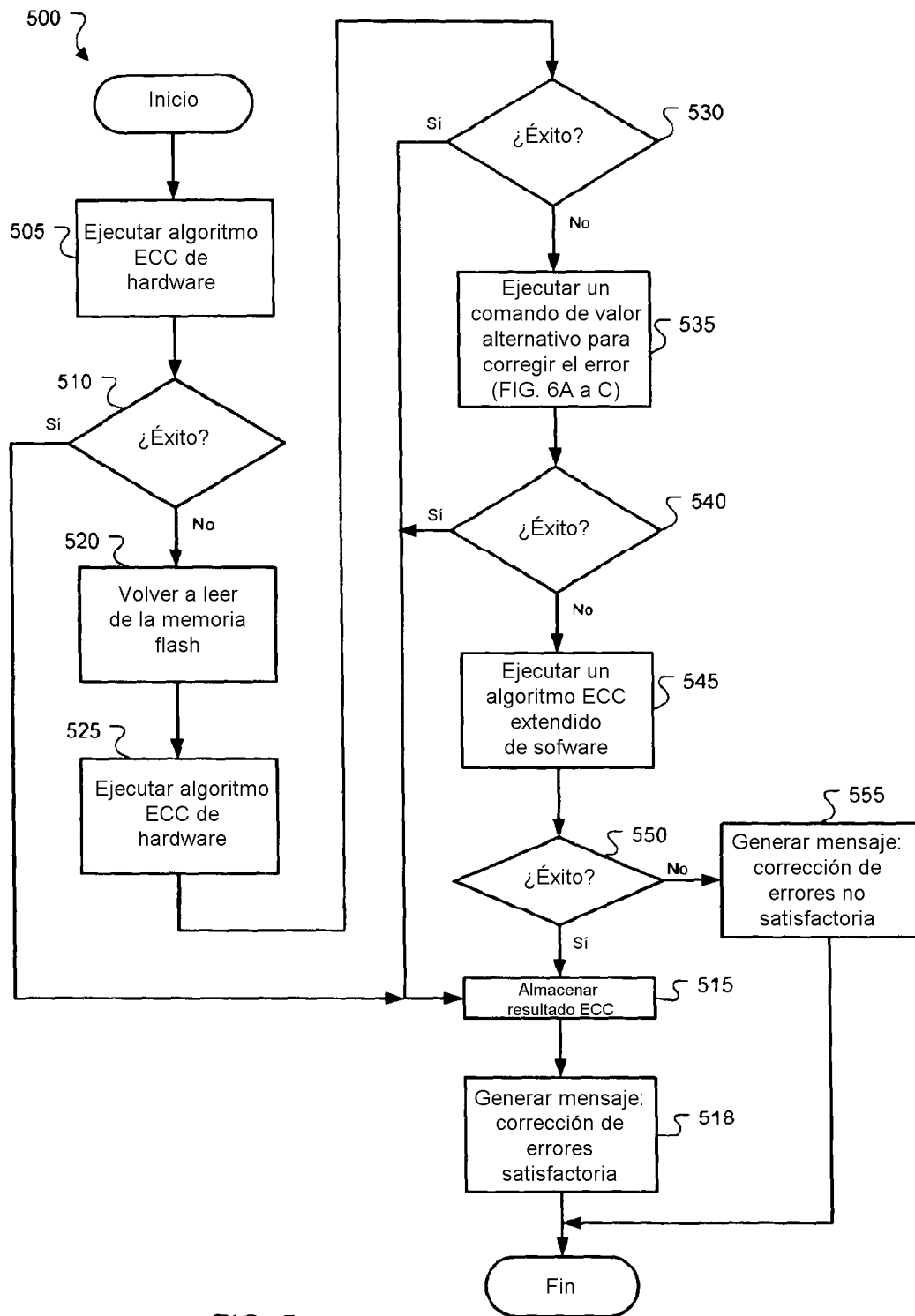


FIG. 5

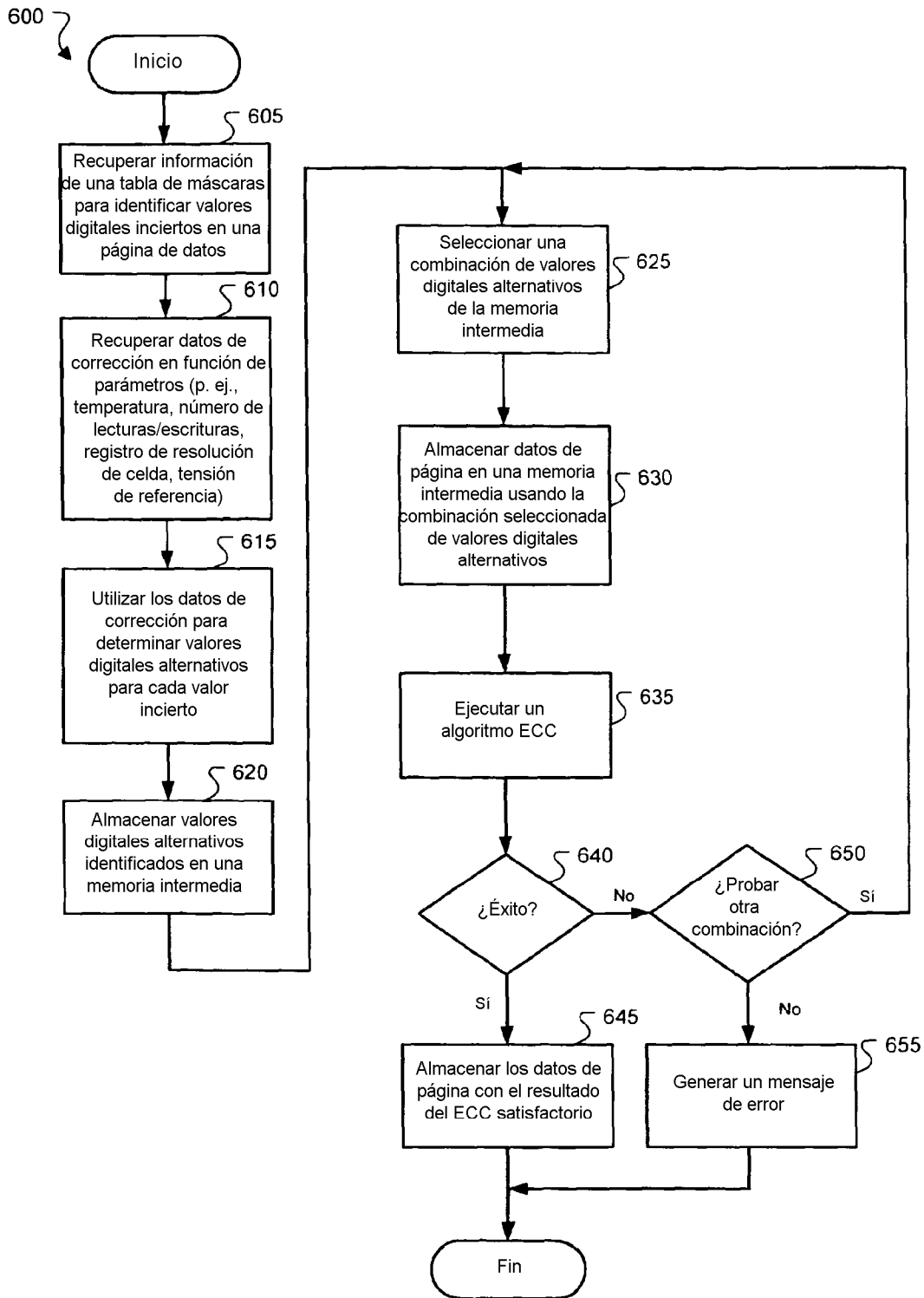


FIG. 6A

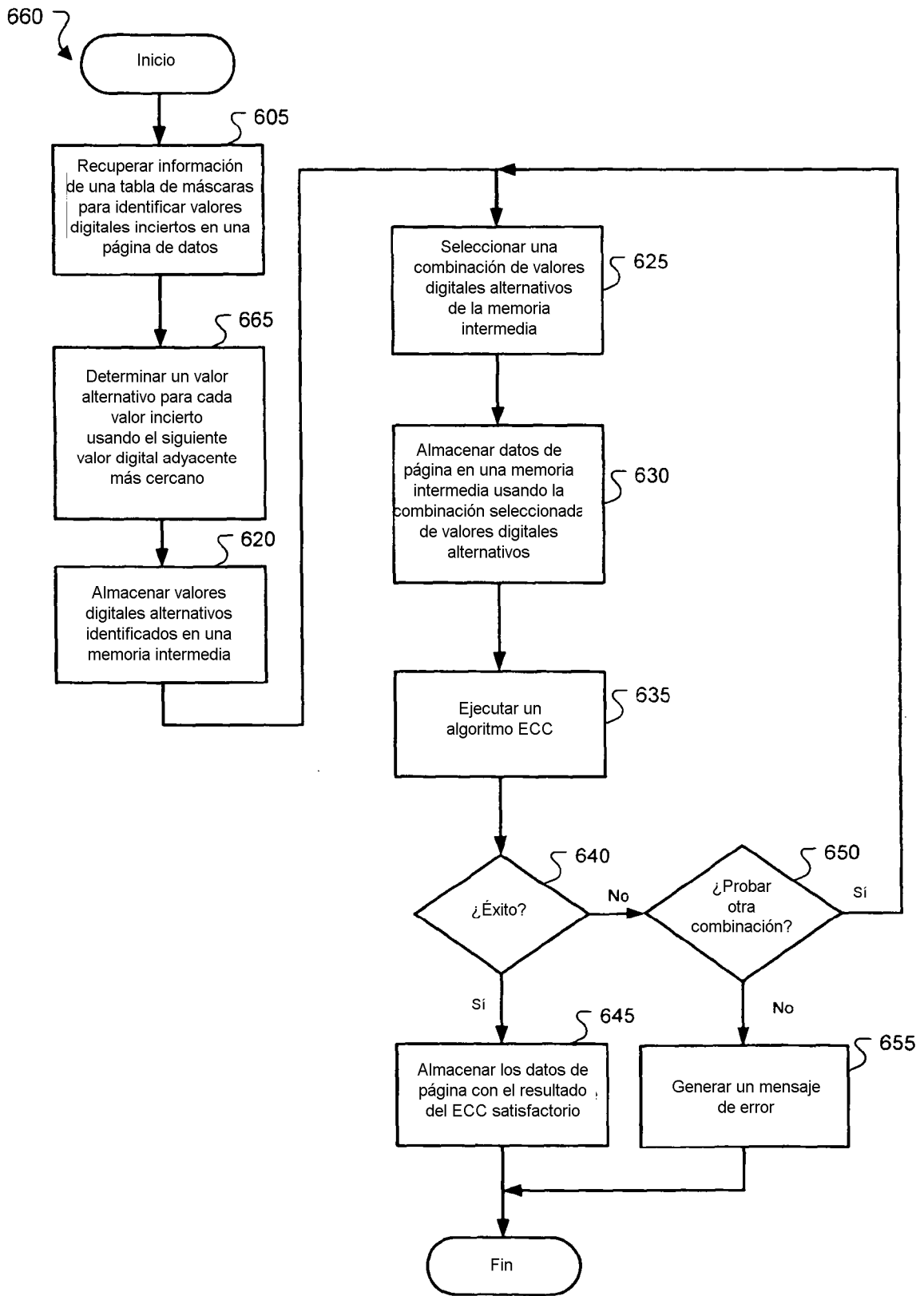


FIG. 6B

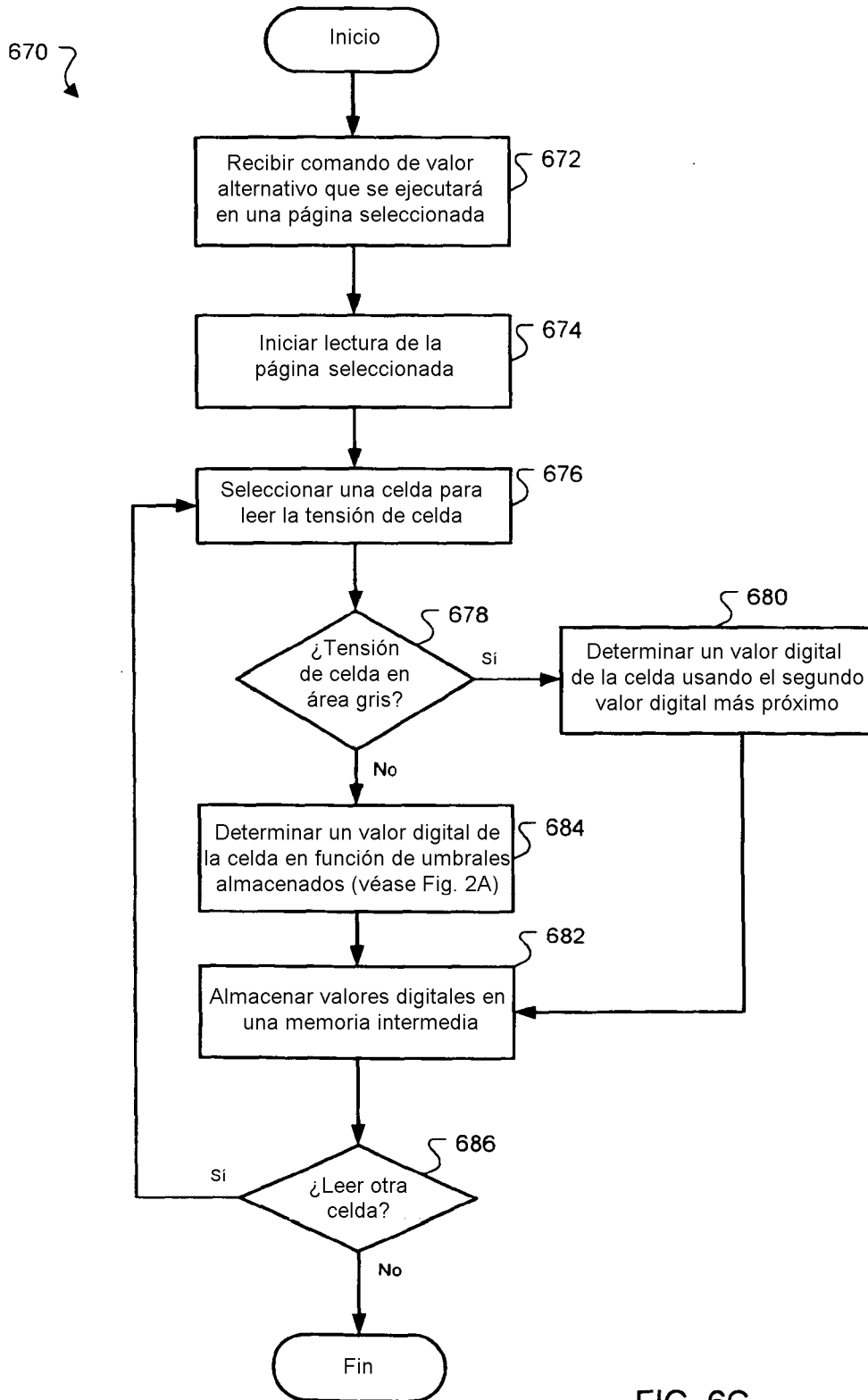


FIG. 6C

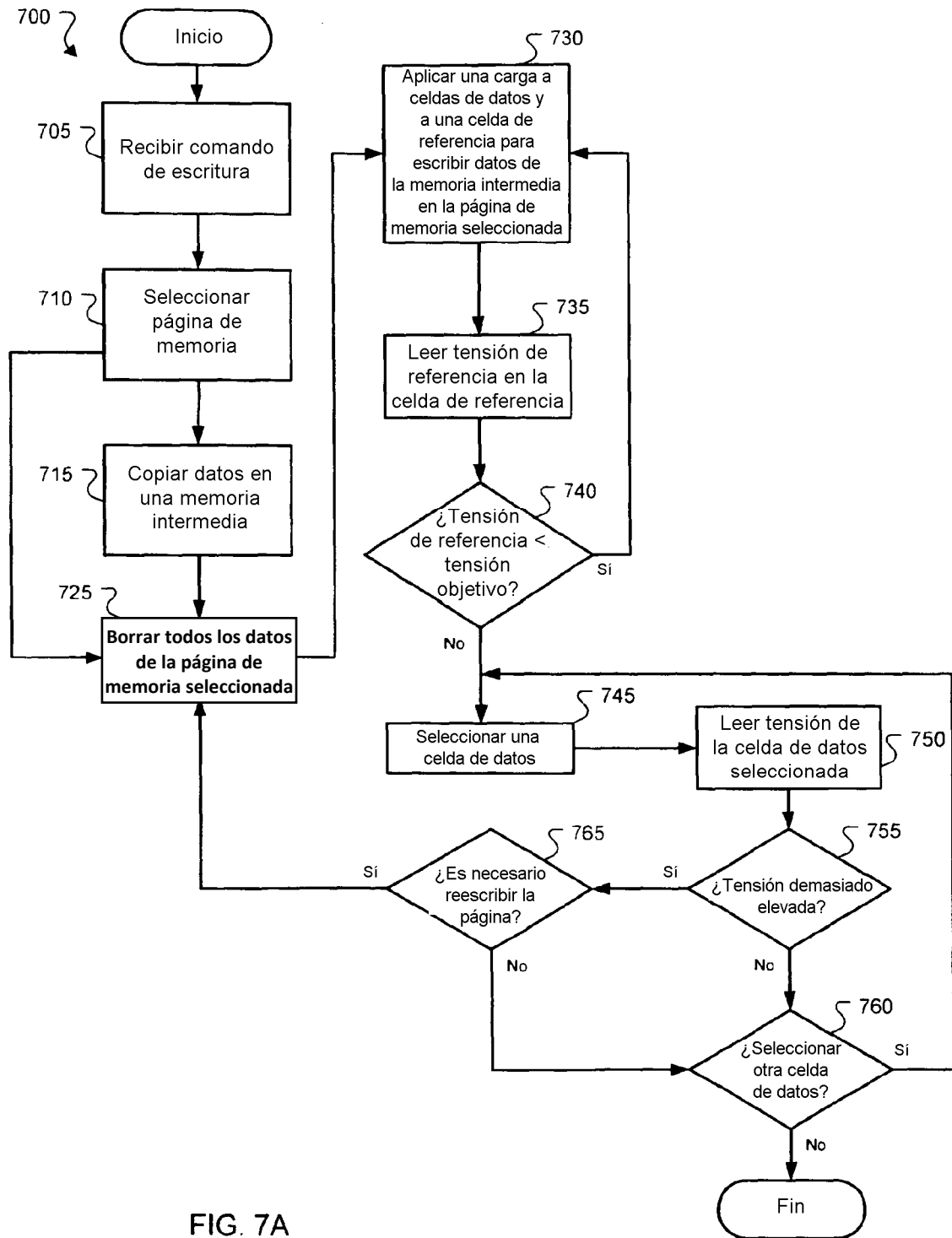


FIG. 7A

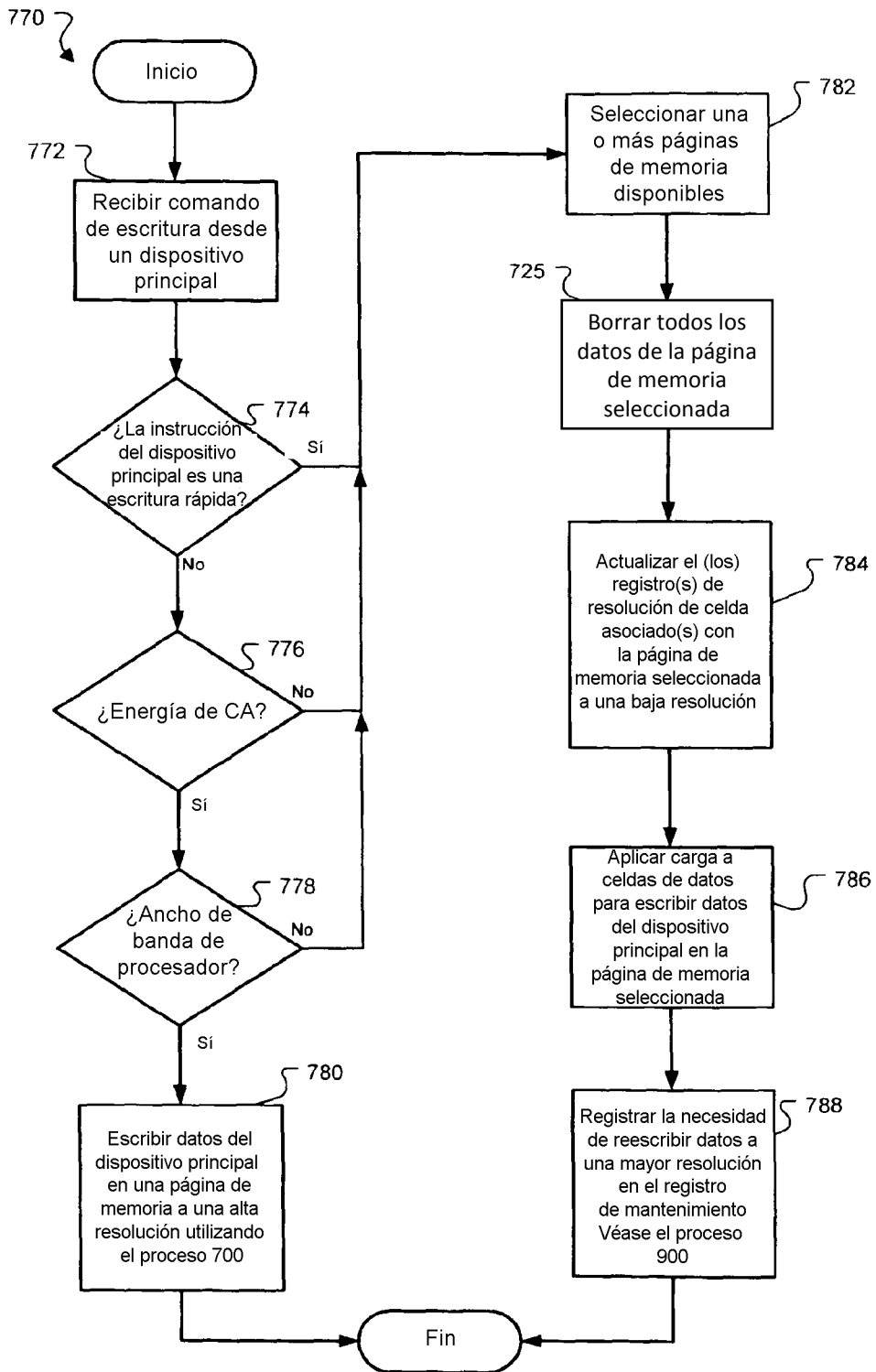


FIG. 7B

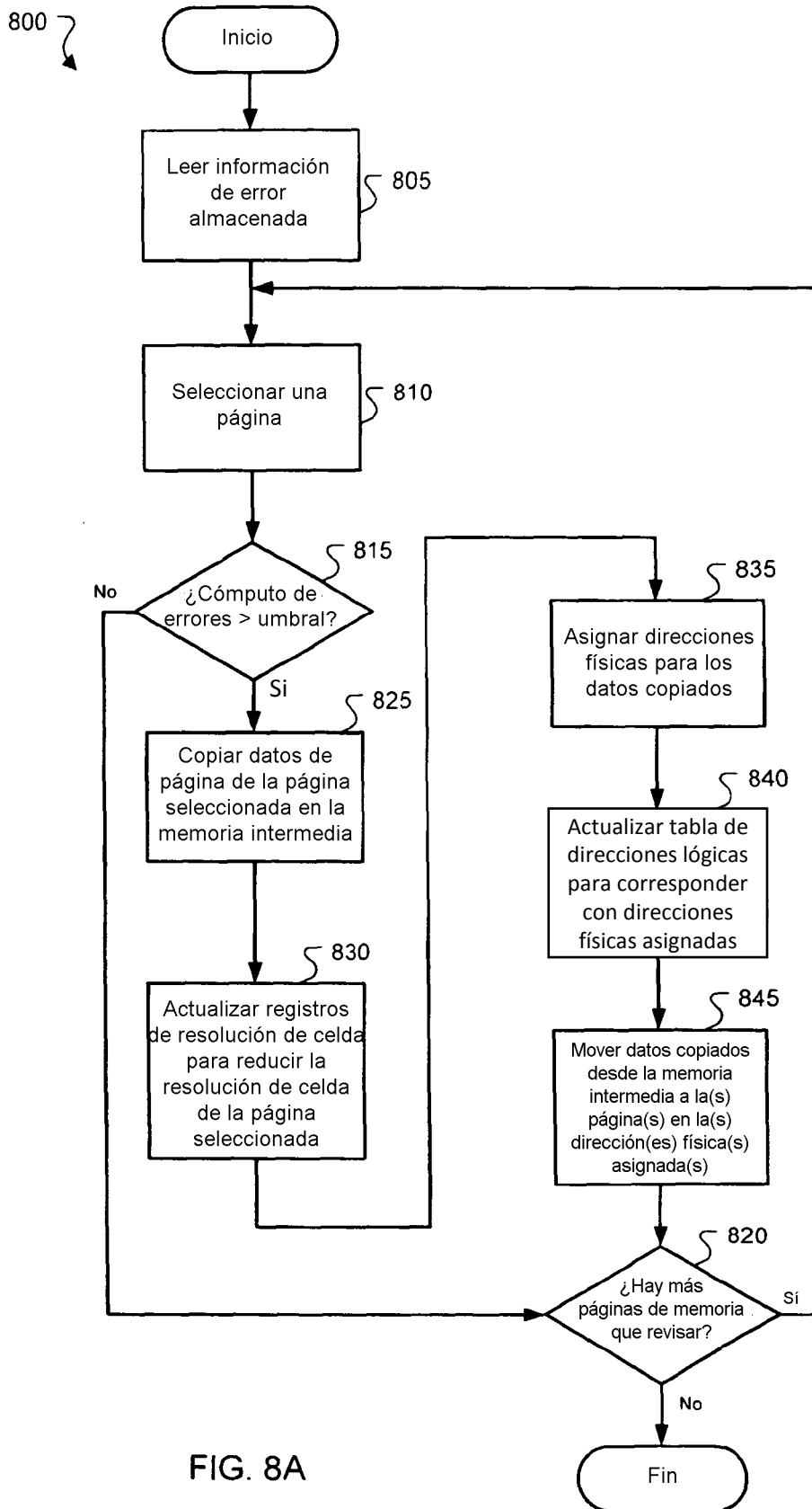
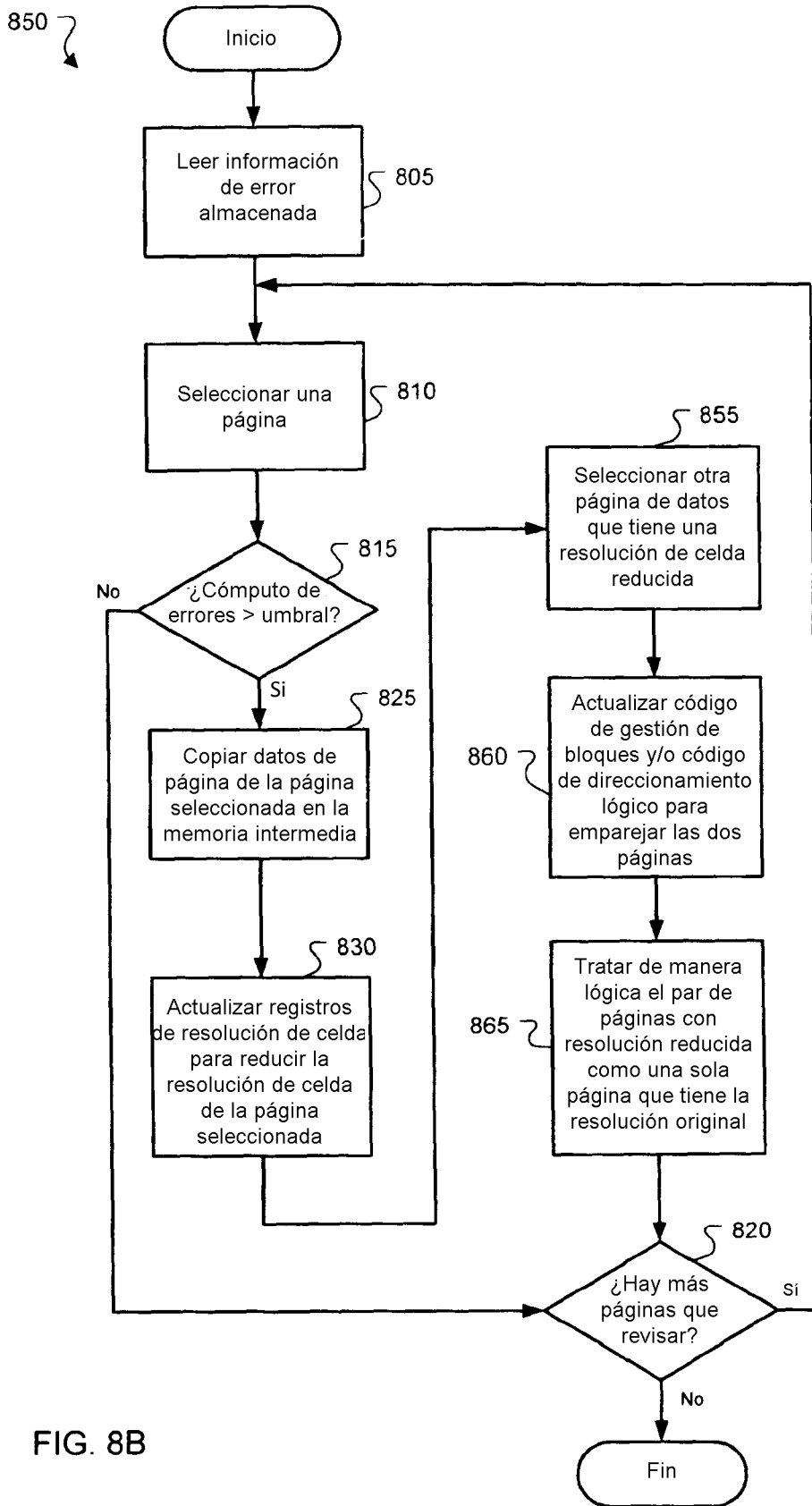
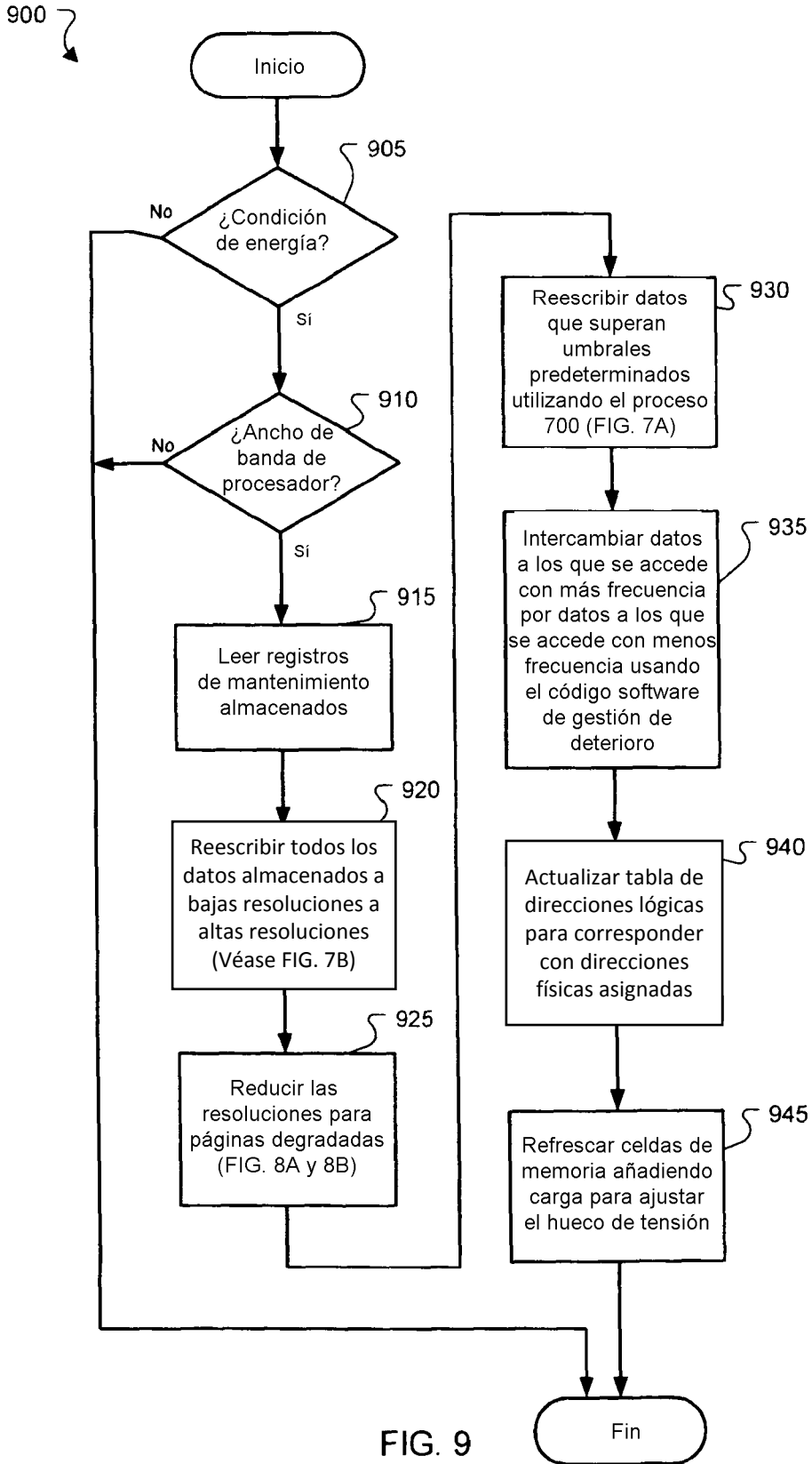


FIG. 8A





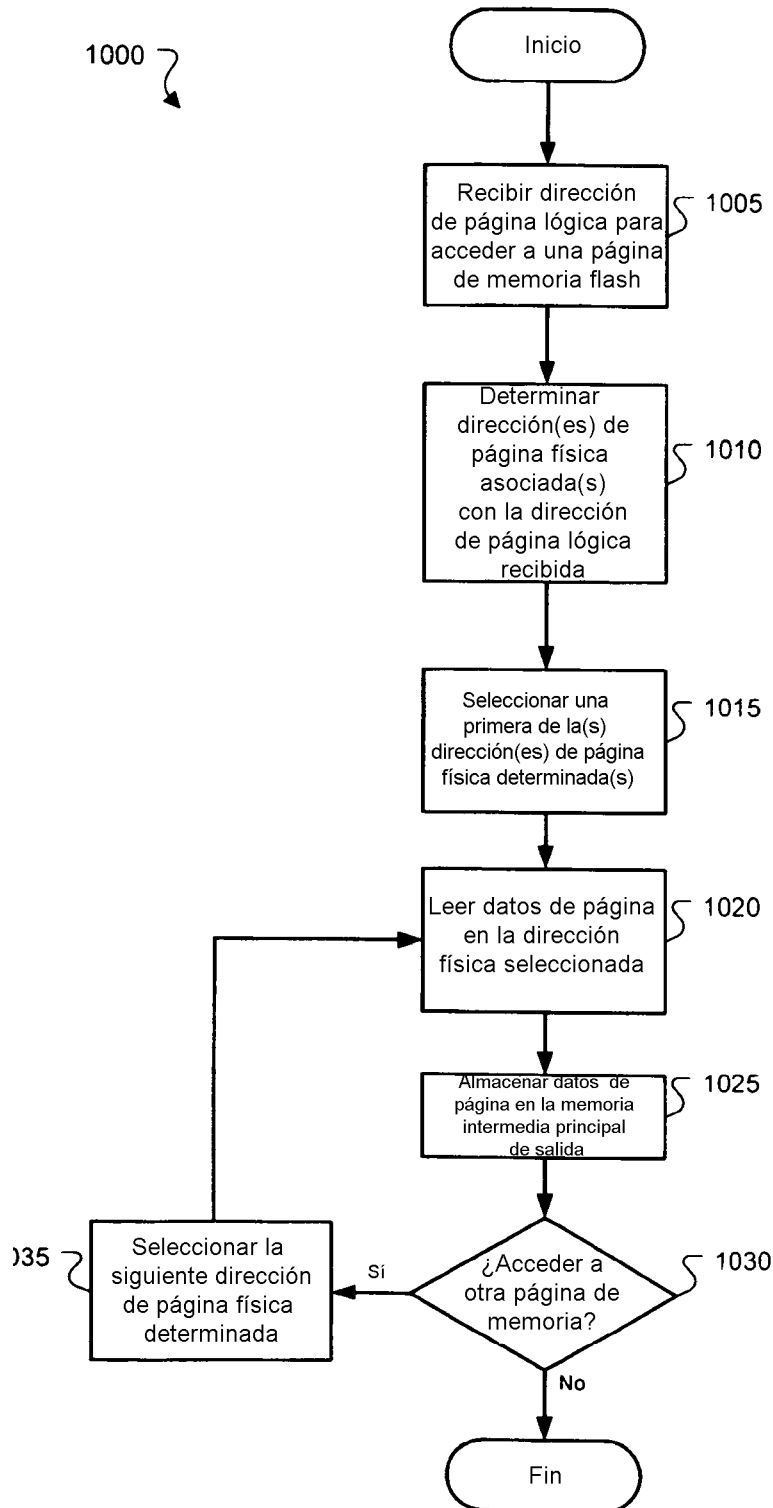


FIG. 10

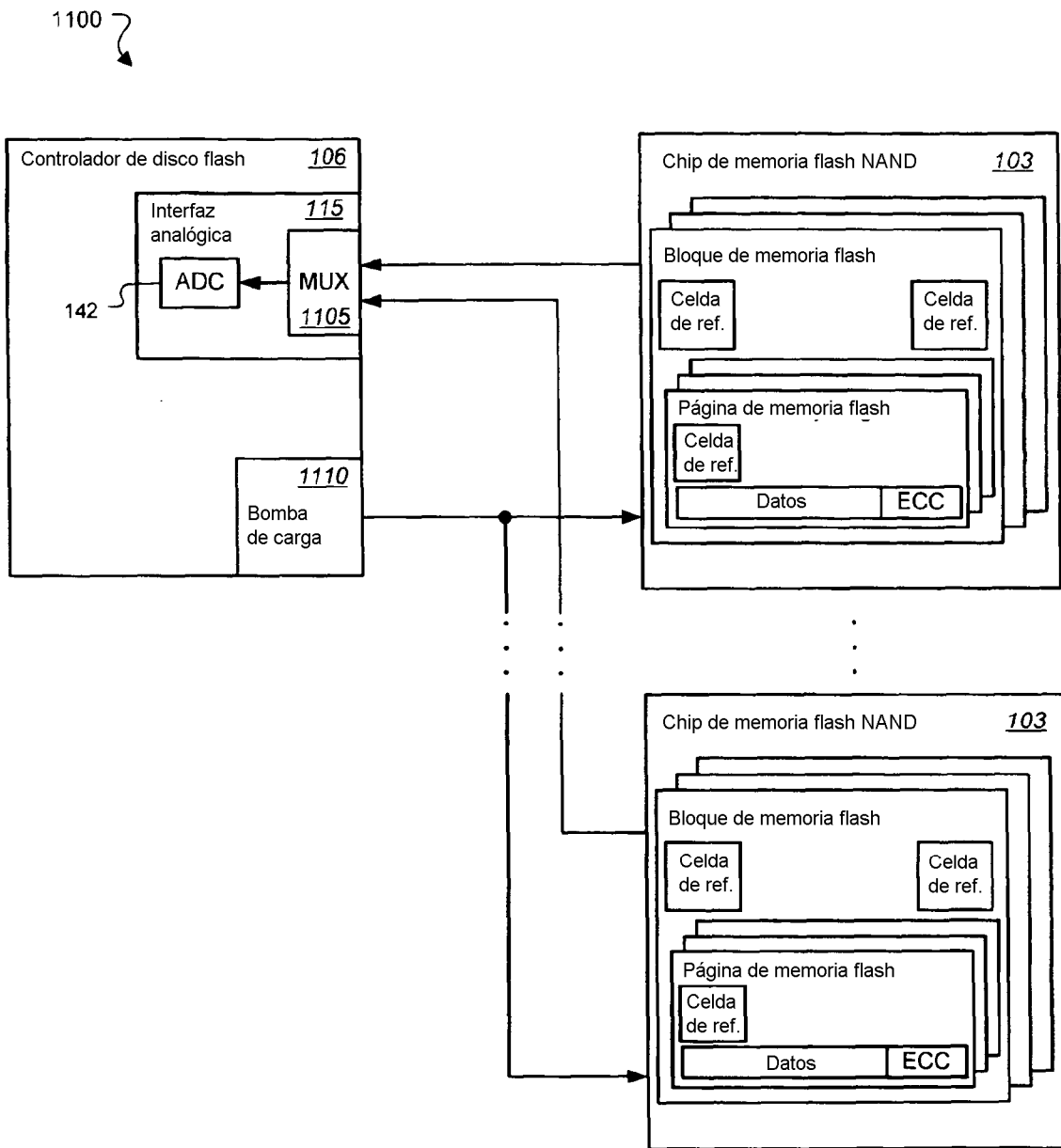


FIG. 11

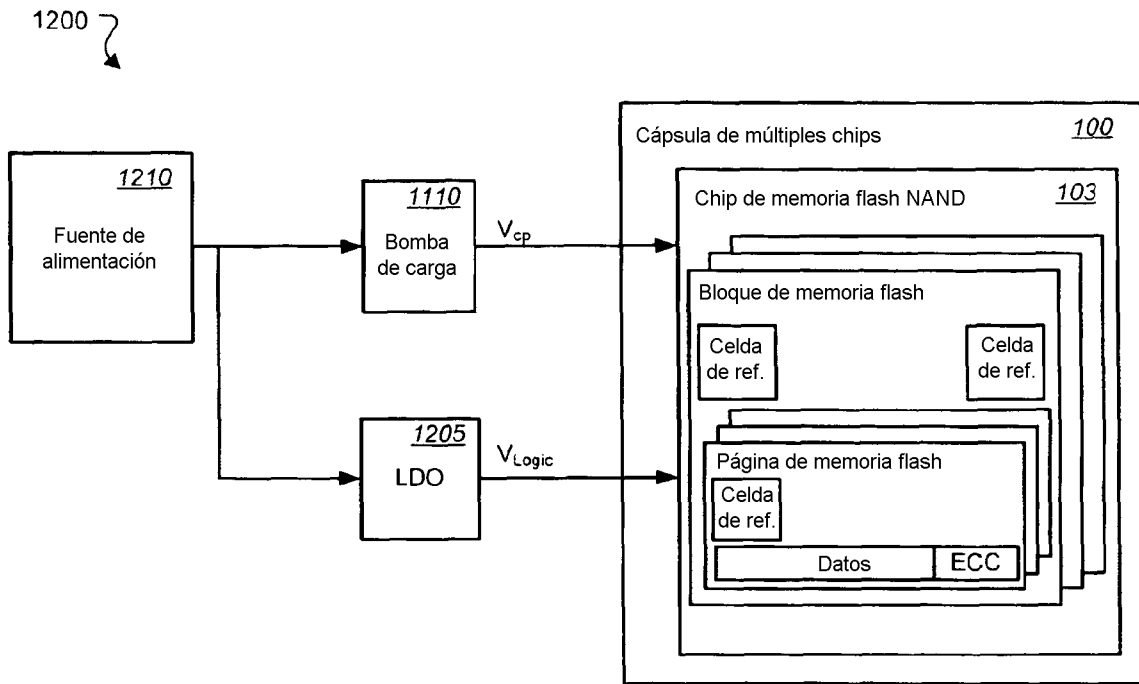


FIG. 12