

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 867 929**

51 Int. Cl.:

G05B 19/414 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **09.10.2018** **E 18199464 (1)**

97 Fecha y número de publicación de la concesión europea: **24.03.2021** **EP 3561622**

54 Título: **Controlador integrado para control de movimiento y control de motor**

30 Prioridad:

25.04.2018 CN 201810380436

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

21.10.2021

73 Titular/es:

RTIMEMAN MOTION CONTROL CO., LTD.
(100.0%)

Room 1102, Building 2, No.188 Yizhou Rd., Xuhui District
Shanghai, CN

72 Inventor/es:

HE, YAN;
GONG, SHAOQIU;
ZHANG, SHUGUO;
LI, RUIQIN;
QIAN, JIN;
TANG, WENBIN;
WANG, QICHAO;
FENG, YUN;
HU, YUEJIN y
FAN, DONGPING

74 Agente/Representante:

CURELL SUÑOL, S.L.P.

ES 2 867 929 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

Controlador integrado para control de movimiento y control de motor

5 **Campo técnico**

La presente invención se refiere en general a un controlador y, más particularmente, a un controlador integrado para control de movimiento y control de motor.

10 **Técnica anterior**

15 El control de movimiento y el control de motor son tecnologías nucleares en el campo de la automatización industrial. El controlador de movimiento y el controlador de motor son dos dispositivos de control críticos comunes utilizados en aparatos automáticos que realizan operaciones complicadas. Los controladores integrados para el control de movimiento y el control de motor pueden aplicarse también en otros campos tales como el campo de los vehículos aéreos no tripulados.

20 Convencionalmente, la mayoría de los sistemas automatizados industriales, como robots o máquinas herramienta a gran escala adoptan la arquitectura de control distribuido. En una arquitectura de control distribuido, un controlador de movimiento trabaja junto con múltiples controladores de motor. La transmisión de datos entre el controlador de movimiento y el controlador de motor y entre los controladores de motor se consigue a través de buses como buses de campo.

25 Dicha arquitectura de control distribuido presenta una pluralidad de desventajas conocidas. Por ejemplo, el excesivo hardware puede llevar a elevados costes de hardware y la ocupación de espacios grandes. Además, el protocolo de comunicación de bus es vulnerable a interferencias y afronta problemas de cuellos de botella en volúmenes de transmisión de datos y tasas de transmisión de datos.

30 En consecuencia, el concepto de "integración de accionamiento y control" es propuesto por la industria para obtener un controlador industrial que integra las funciones del controlador de movimiento y el controlador de motor. Sin embargo, el controlador industrial integrado convencional de accionamiento y control es meramente una simple combinación física del controlador de movimiento y el controlador de motor. Como resultado, la mejora en la cooperación de los dos controladores es limitada.

35 El documento US 2018/065244 A1 divulga un dispositivo de control que incluye una interfaz para emitir un valor de orden hacia un accionador de motor adaptado para accionar un motor, una parte de almacenamiento adaptada para almacenar una o más órdenes para especificar un comportamiento del motor accionado por el accionador de motor, y una parte de procesamiento que incluye un primer circuito aritmético y un segundo circuito aritmético.

40 El documento US 2003/230998 A1 divulga un sistema de control distribuido por robot que incluye un controlador principal que emite una primera orden y una pluralidad de controladores de comunicación conectados al controlador principal a modo de estructura de árbol.

45 El documento WO 2014/110748 A1 divulga un controlador de movimiento que incluye unos medios de compartición de información y una pluralidad de módulos de función.

50 El documento US 2013/307459 A1 divulga un generador de perfil que puede calcular un perfil de movimiento de curva ST que incluye una referencia de fluctuación que varía continuamente a lo largo del tiempo durante por lo menos uno de los segmentos de perfil de movimiento.

Breve resumen de la divulgación

55 En consecuencia, un objetivo de la presente invención es proporcionar un controlador integrado para el control de movimiento y el control de motor, que pueden mejorar de manera fiable y estable la tasa de interacción entre la parte de control de movimiento y la parte de control de motor.

60 Con el fin de resolver el problema técnico anterior, la presente invención proporciona un controlador integrado para el control de movimiento y el control de motor que comprende un primer procesador, un segundo procesador, una memoria caché y una memoria compartida. El primer procesador está configurado para ejecutar un sistema operativo y realizar por lo menos el control de movimiento, en el que el control de movimiento comprende calcular valores establecidos de pares cinemáticos de un objeto controlado en cada momento. El segundo procesador está configurado para realizar por lo menos el control de motor y normalmente no para ejecutar el sistema operativo, en el que el control de motor comprende controlar la rotación de motores según los valores establecidos, que comprende por lo menos cualquiera o una combinación de entre control de bucle de corriente de motor, control de bucle de velocidad de motor y control de bucle de posición de motor. La memoria caché está acoplada al primer procesador y al segundo procesador. La memoria compartida es un bloque de memoria cacheable dedicada con

5 direcciones fijas especificadas por el sistema operativo. La memoria compartida mapea sobre la memoria caché. El primer procesador y el segundo procesador están configurados para compartir la memoria compartida y, en consecuencia, realizar una transmisión de datos a través de la memoria caché durante los periodos de control de movimiento y control de motor. El primer procesador, el segundo procesador y la memoria caché están integrados en un mismo chip.

10 En una forma de realización de la presente invención, antes del control de movimiento y del control de motor, el controlador integrado está configurado para realizar las siguientes operaciones: iniciar el primer procesador y el segundo procesador por multiprocesamiento simétrico; ejecutar el sistema operativo en el primer procesador y el segundo procesador; desconectar el segundo procesador y mantener el sistema operativo ejecutándose individualmente en el primer procesador; reiniciar y reconfigurar el segundo procesador para hacer que el segundo procesador no ejecute normalmente el sistema operativo.

15 En una forma de realización de la presente invención, el primer procesador está configurado para escribir datos de control de movimiento en la memoria caché, el segundo procesador está configurado para leer los datos de control de movimiento de la memoria caché, comprendiendo los datos de control de movimiento datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza para los pares cinemáticos o los motores.

20 En una forma de realización de la invención, el segundo procesador está configurado para escribir datos de realimentación en la memoria caché, el primer procesador está configurado para leer la realimentación de la memoria caché, comprendiendo los datos de realimentación datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza de los pares cinemáticos o los motores.

25 La presente invención proporciona también un controlador integrado para control de movimiento y control de motor que comprende un primer procesador, un segundo procesador, una memoria caché, una memoria compartida y un dispositivo lógico programable. El primer procesador está configurado para ejecutar un sistema operativo y realizar por lo menos un control de movimiento, comprendiendo el control de movimiento calcular valores establecidos de pares cinemáticos de un objeto controlado en cada momento. El segundo procesador está configurado para realizar por lo menos un control de motor y normalmente no para ejecutar el sistema operativo, comprendiendo el control de motor controlar la rotación de los motores según los valores establecidos, que comprenden por lo menos cualquiera o una combinación de entre control de bucle de corriente de motor, control de bucle de velocidad de motor y control de bucle de posición de motor. La memoria caché está acoplada al primer procesador y al segundo procesador. La memoria compartida es un bloque de memoria cacheable dedicado con direcciones fijas especificadas por el sistema operativo. El dispositivo lógico programable está acoplado con el segundo procesador y está configurado para realizar el control de motor junto con el segundo procesador. La memoria compartida mapea sobre la memoria caché. El primer procesador y el segundo procesador están configurados para compartir la memoria compartida y, en consecuencia, realizar una transmisión de datos a través de la memoria caché en los periodos de control de movimiento y control de motor. El primer procesador, el segundo procesador y la memoria caché están integrados en un mismo chip.

45 En una forma de realización de la presente invención, el primer procesador está configurado para escribir datos de control de movimiento en la memoria caché, el segundo procesador está configurado para leer los datos de control de movimiento de la memoria caché, comprendiendo los datos de control de movimiento datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza para los pares cinemáticos o los motores.

50 En una forma de realización de la invención, el segundo procesador está configurado para escribir datos de realimentación en la memoria caché, el primer procesador está configurado para leer la realimentación de la memoria caché, comprendiendo los datos de realimentación datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza de los pares cinemáticos o los motores.

55 En una forma de realización de la invención, un reloj del dispositivo lógico programable y un reloj del segundo procesador están sincronizados. Cuando el segundo procesador lee datos en el dispositivo lógico programable o los escribe en este, el dispositivo lógico programable ancla los datos leídos o escritos.

60 En una forma de realización de la presente invención, antes del control de movimiento y del control de motor, el controlador está configurado para realizar las siguientes operaciones: iniciar el primer procesador y el segundo procesador por multiprocesamiento simétrico; ejecutar el sistema operativo en el primer procesador y el segundo procesador; desconectar el segundo procesador y mantener el sistema operativo ejecutándose individualmente en el primer procesador; reiniciar y reconfigurar el segundo procesador para hacer que el segundo procesador no ejecute normalmente el sistema operativo.

65 La presente invención proporciona también un controlador integrado para control de movimiento y control de motor que comprende un primer procesador, un segundo procesador, una memoria caché, una memoria compartida y un dispositivo lógico programable. El primer procesador está configurado para ejecutar un sistema operativo y realizar

por lo menos un control de movimiento, comprendiendo el control de movimiento calcular valores establecidos de pares cinemáticos de un objeto controlado en cada momento. El segundo procesador está configurado para realizar por lo menos un control de motor y normalmente no para ejecutar el sistema operativo, comprendiendo el control de motor controlar la rotación de los motores según los valores establecidos, que comprenden por lo menos cualquiera o una combinación de entre control de bucle de corriente de motor, control de bucle de velocidad de motor y control de bucle de posición de motor. La memoria caché está acoplada con el primer procesador y el segundo procesador. La memoria compartida es un bloque de memoria cacheable dedicado con direcciones fijas especificadas por el sistema operativo. El dispositivo lógico programable está acoplado con el segundo procesador y está configurado para realizar el control de motor junto con el segundo procesador. La memoria compartida mapea sobre la memoria caché. El primer procesador y el segundo procesador están configurados para compartir la memoria compartida y, en consecuencia, realizar una transmisión de datos a través de la memoria caché en los periodos de control de movimiento y control de motor. El primer procesador, el segundo procesador, la memoria caché y el dispositivo lógico programable están integrados en un mismo chip.

En una forma de realización de la presente invención, el primer procesador está configurado para escribir datos de control de movimiento en la memoria caché, el segundo procesador está configurado para leer los datos de control de movimiento de la memoria caché, comprendiendo los datos de control de movimiento datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza para los pares cinemáticos o los motores.

En una forma de realización de la invención, el segundo procesador está configurado para escribir datos de realimentación en la memoria caché, el primer procesador está configurado para leer la realimentación de la memoria caché, comprendiendo los datos de realimentación, datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza de los pares cinemáticos o los motores.

En una forma de realización de la presente invención, antes del control de movimiento y del control de motor, el controlador está configurado para realizar las siguientes operaciones: iniciar el primer procesador y el segundo procesador por multiprocesamiento simétrico; ejecutar el sistema operativo en el primer procesador y el segundo procesador; desconectar el segundo procesador y mantener el sistema operativo ejecutándose individualmente en el primer procesador; reiniciar y reconfigurar el segundo procesador para hacer que el segundo procesador no ejecute normalmente el sistema operativo.

En comparación con la tecnología convencional, la presente invención proporciona un controlador con altas prestaciones en tiempo real a través de una memoria caché compartida y una configuración especial del sistema. Además, puesto que el sistema de integración de accionamiento y control está incorporado en un solo chip, la transmisión de datos se realiza en el interior del chip sin interferencia externa, lo que asegura una transmisión de datos apropiada y fiable. Además, el sistema de integración de accionamiento y control reduce también la redundancia del hardware. Puesto que el sistema de integración de accionamiento y control puede materializarse utilizando un único chip SOC con elementos de circuito periféricos necesarios, se reduce también la ocupación del hardware.

Breve descripción de los dibujos

Con el fin de que las soluciones técnicas de la presente invención puedan entenderse más completamente, las formas de realización de la presente invención se describirán ahora en detalle en lo que sigue con referencia a los dibujos que se acompañan. Principalmente, los dibujos se utilizan sustancialmente para ilustrar las formas de realización de la invención y no deberán utilizarse como limitación de la misma.

La figura 1 es un diagrama de bloques de un controlador según una forma de realización de la presente invención;

La figura 2 es un diagrama de circuito de un controlador según una primera forma de realización de la presente invención;

La figura 3 es un diagrama que ilustra el funcionamiento del controlador según la primera forma de realización de la presente invención;

La figura 4 es un diagrama de circuito de un controlador de acuerdo con una segunda forma de realización de la presente invención;

La figura 5 es un diagrama que ilustra el funcionamiento del controlador de acuerdo con la segunda forma de realización de la presente invención;

La figura 6 es un diagrama de circuito de un controlador de acuerdo con una tercera forma de realización de la presente invención;

La figura 7 es un diagrama de flujo que ilustra un procedimiento para realizar el control de movimiento y el control de motor por un controlador integrado según una forma de realización de la presente invención;

5 La figura 8 es un diagrama que ilustra el primer procesador y el segundo procesador cargando códigos según una forma de realización de la presente invención;

La figura 9 es un diagrama que ilustra ciclos de sincronización y actualización del segundo procesador y el dispositivo lógico programable según una forma de realización de la presente invención;

10 La figura 10 es un diagrama que ilustra registros de los dispositivos lógicos programables según una forma de realización de la presente invención.

Descripción detallada de las formas de realización

15 Con el fin de que los objetivos, características y ventajas de la presente invención puedan entenderse más completamente, las formas de realización de la presente invención se describirán a continuación en detalle en lo que sigue haciendo referencia a los dibujos adjuntos.

20 Como se utilizan en la presente memoria, las formas singulares “un”, “una”, “el” y “la” están destinadas a incluir también las formas plurales a menos que el contexto indique claramente otra cosa. Se entenderá además que los términos “comprende” y/o “que comprende”, cuando se utilizan en esta memoria, especifican la presencia de características, números enteros, etapas, operaciones, elementos y/o componentes expuestos, pero no impiden la presencia o adición de una o más características, números enteros, etapas, operaciones, elementos, componentes y/o grupos adicionales de los mismos.

25 Asimismo, los términos “acoplar”, “conectar” están destinados a designar un acoplamiento o conexión directo o indirecto. En consecuencia, si un dispositivo está acoplado a otro dispositivo, esa conexión puede ser a través de una conexión directa o a través de una conexión indirecta por medio de otros componentes y conexiones, a menos que el contexto indique claramente otra cosa. El término “y/o” está destinado a indicar cualquiera o todas las combinaciones de uno o más componentes enumerados.

30 Las formas de realización de la presente invención describen un controlador que presenta una función de control de movimiento y una función de control de motor. La figura 1 es un diagrama de bloques del controlador según una forma de realización de la presente invención. Se entiende por los expertos en la materia que el control de movimiento se refiere al cálculo de movimientos diana de pares cinemáticos de un objeto controlado en cada momento en aplicaciones específicas. En la presente memoria, el par cinemático se refiere a una conexión móvil entre dos cuerpos del objeto controlado en contacto entre ellos que impone restricciones sobre su movimiento relativo tal como una junta. El control de movimiento pretende obtener la relación entre los valores establecidos de pares cinemáticos del objeto controlado y el tiempo, así como generar un flujo correspondiente de datos “valores establecidos de pares cinemáticos en función del tiempo” (información de datos), a pesar de los cambios en las etapas de cálculo específicas, cálculo de dianas, cálculo de parámetros y variables en diferentes aplicaciones. Los valores establecidos de pares cinemáticos pueden ser uno o más de entre la posición, la velocidad, la aceleración, la fuerza y el momento de fuerza de los pares cinemáticos.

45 El control de movimiento controla la rotación de los motores según los “valores establecidos de pares cinemáticos”. En una forma de realización de la presente invención, los valores establecidos de pares cinemáticos se convierten en valores dados de motores de juntas del objeto controlado. Los valores dados de motores pueden ser uno o más de entre el ángulo rotacional, velocidad rotacional, par del motor. La transición de los valores dados de motores desde los valores establecidos de pares cinemáticos puede realizarse por la parte de control de movimiento o la parte de control de motor, que no está limitada en la presente memoria. En consecuencia, los valores establecidos de pares cinemáticos se transmiten a la parte de control de motor puntualmente en momentos predeterminados de tal manera que la parte de control de motor pueda controlar en consecuencia la rotación de los motores. Cuando los motores de todos los ejes alcanzan sus valores dados correspondientes de motor dentro del tiempo requerido, el movimiento resultante del objeto controlado se consigue como se esperaba. Se observa que, en la presente invención, se espera que la parte de control de motor controle múltiples ejes simultánea y rápidamente para asegurar la sincronización multieje y el movimiento resultante apropiado. Además, se espera también que la parte de control de motor consiga una elevada frecuencia en la actualización y ejecución de datos de manera que se cumplan los requisitos de la precisión de movimiento y la continuidad de movimiento. Como se muestra en la figura 1, en una forma de realización de la presente invención, el controlador integrado 10 comprende un controlador de movimiento 11 y un controlador de motor 12. El controlador de movimiento 11 puede realizar la función de control de movimiento antes mencionada, el controlador de motor 12 puede realizar la función de control de motor antes mencionada. Una consola portátil es un ordenador principal para el controlador de movimiento 11, para enviar o editar instrucciones de tarea de aplicación al controlador de movimiento. El controlador de movimiento recibe las instrucciones procedentes de la consola portátil y realiza la planificación de la trayectoria de movimiento según las instrucciones, y transmite entonces los datos de control de movimiento, tales como los valores dados de motores de múltiples ejes al controlador de motor 12. El controlador de motor 12 produce una señal de control (como señal

PWM) según los valores dados y emite la señal de control hacia un accionador de motor 13 por el que se accionan los motores.

5 En las formas de realización de la presente invención, el controlador 10 puede utilizarse en diversas aplicaciones tales como robots eléctricos, máquinas herramienta de control numérico, aeronaves eléctricas de tipo multirroto, electromóviles, prótesis mecánicas, manos mecánicas, vehículos móviles eléctricos, etc., para realizar el control de movimiento y el control de motor para aparatos accionados por motor coordinados de múltiples ejes.

10 En las formas de realización de la presente invención, la cantidad de ejes de motor no está limitada y pueden ser 6 u 8, o más o menos.

15 Actualmente, en un sistema convencional de automatización de múltiples ejes, la transmisión de datos entre el controlador de movimiento y el controlador de motor, y entre los controladores de motor se consigue a través de buses archivados. En la forma de realización de la presente invención, se mejora dicho protocolo de comunicación entre el controlador de movimiento y los controladores de motor.

Primera forma de realización

20 La figura 2 es un diagrama de circuito de un controlador según una primera forma de realización de la presente invención. Como se muestra en la figura 2, el controlador integrado 20 para el control de movimiento y el control de motor comprende un primer núcleo de procesamiento 21, un segundo núcleo de procesamiento 22, una memoria caché 23 y una memoria compartida 30. El primer núcleo de procesamiento 21 está configurado para ejecutar un sistema operativo y realizar por lo menos el control de movimiento. El segundo núcleo de procesamiento 22 está configurado para realizar el control de motor y normalmente no ejecutar el sistema operativo. En la presente memoria, "normalmente" se define como la mayor parte del tiempo. La memoria caché, como es conocido por los expertos en la materia, es una memoria entre un procesador y una memoria principal. La memoria caché funciona más rápido que la memoria principal a una velocidad casi próxima al procesador. La memoria caché 23 puede ser una memoria caché Nivel 2 (memoria caché L2) para un procesador. Se entiende que la memoria caché 23 puede ser de otros niveles. La memoria caché 23 está acoplada al primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22. La memoria compartida 30 es un bloque de memoria cacheable dedicado con direcciones fijas, que es especificado por el sistema operativo ejecutado en el primer núcleo de procesamiento 21. La memoria compartida 30 mapea sobre la memoria caché 23. El primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 están configurados para compartir la memoria compartida 30 y, en consecuencia, realizar una transmisión de datos a través de la memoria caché 23 durante los periodos del control de movimiento y el control de motor. El primer procesador, el segundo procesador y la memoria caché están integrados en un mismo chip. Por ejemplo, la memoria compartida 30 mapea sobre la memoria caché 23 a través de un mapeo de direcciones. Cuando la memoria compartida 30 mapea sobre la memoria caché 23, la transmisión de datos se realiza en la memoria caché 23 que mejora la tasa de transmisión en 8-10 veces en comparación con el intercambio directo de datos en una memoria física.

40 En el contexto de la presente invención, el núcleo de procesamiento es una CPU. La CPU puede contener también una memoria caché, como una memoria caché Nivel 1 (memoria caché L1). Múltiples CPU pueden acoplarse a la memoria caché L2 para realizar la comunicación entre ellas.

45 En la arquitectura de procesador actual, múltiples núcleos de procesamiento que comparten la misma memoria caché están usualmente integrados en un mismo chip de procesamiento. Por tanto, en la forma de realización, el primer núcleo de procesamiento 21, el segundo núcleo de procesamiento 22 y la memoria caché 23 están integrados en un mismo chip, tal como un chip de procesamiento.

50 En la forma de realización, el primer núcleo de procesamiento 21 ejecuta el sistema operativo y realiza por lo menos el control de movimiento. El sistema operativo puede supervisar el funcionamiento de todo el controlador. El sistema operativo puede realizar también tareas de aplicación específicas, como la tarea de control de movimiento. La tarea de control de movimiento pretende calcular movimientos diana de pares cinemáticos de un objeto controlado en cada momento. La implementación del control de movimiento puede ser diversa. Por ejemplo, según los requisitos de aplicación, diferentes algoritmos de planificación de trayectorias y el software de análisis de tareas pueden desarrollarse para realizar diversas planificaciones de trayectorias y conversiones de bloques de unión. Los algoritmos específicos para dichas planificaciones de trayectorias y conversiones de bloques de unión pueden ser también ampliamente diferentes según las aplicaciones. Por ejemplo, pueden aplicarse algoritmos cinemáticos directos y algoritmos cinemáticos inversos en robots industriales; pueden aplicarse algoritmos de control de vuelo de UAV en UAV. Es obvio para los expertos en la materia la implementación de la función de control de movimiento cuando se requiera.

En la forma de realización, el sistema operativo es un sistema Linux.

65 El primer núcleo de procesamiento 21 calcula uno o más de entre la posición, la velocidad, la aceleración, la fuerza y el momento de fuerza de los pares cinemáticos del objeto controlado por el controlador 20 en cada momento,

que se toman como valores establecidos de pares cinemáticos. En la forma de realización, el primer núcleo de procesamiento 21 convierte también los valores establecidos de pares cinemáticos en valores dados de motores de las juntas del objeto controlado en cada momento. Los valores dados de motores incluyen la posición, la velocidad, la aceleración, la fuerza, el momento de fuerza de los motores o la combinación de los mismos. El primer núcleo de procesamiento 21 emite los valores dados de motores hacia el segundo núcleo de procesamiento 22 en tiempo predeterminado. Alternativamente, en otras formas de realización, el primer núcleo de procesamiento 21 puede no realizar la conversión de valores establecidos de pares cinemáticos en valores dados de motores. El primer núcleo de procesamiento 21 emite directamente los valores establecidos de pares cinemáticos hacia el segundo núcleo de procesamiento 22, a continuación, el segundo núcleo de procesamiento 22 los convierte en los valores dados de motores y realiza el control de motor correspondiente.

En la forma de realización, el segundo núcleo de procesamiento 22 realiza el control de motor sin ejecutar normalmente el mismo sistema operativo que el primer núcleo de procesamiento 21 o sin ejecutar ningún sistema operativo. Según su asignación, el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 funcionan normalmente en una arquitectura de multiprocesamiento asimétrico. El segundo núcleo de procesamiento 22 controla el accionador de motor según los valores dados de motores para accionarlos de manera que cumplan los requisitos rápida y establemente. El control de motor implica el control de bucle de corriente, el control de bucle de velocidad y/o el control de bucle de posición. El control de bucle diferente puede implementarse según diferentes valores dados de motores. En general, cuando el valor dado es un valor posicional, se requiere que se calculen los tres bucles; cuando el valor dado es un valor relacionado con la velocidad (valor de velocidad o valor de aceleración), se requiere que se calculen el bucle de corriente y el bucle de velocidad; cuando el valor dado es un valor de fuerza o valor de momento de fuerza, se requiere que se calcule el bucle de corriente. En consecuencia, el segundo núcleo de procesamiento 22 realiza alternativamente el control de bucle de corriente, o la combinación del control de bucle de corriente y el control de bucle de velocidad, o la combinación del control de bucle de corriente, el control de bucle de velocidad y el control de bucle de posición, para conseguir la tarea de control de motor. Además, puede modificarse la asignación de las tareas entre el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22. Por ejemplo, el primer núcleo de procesamiento 21 realiza el control de bucle de posición, el segundo núcleo de procesamiento 22 realiza el control de bucle de velocidad y el control de bucle de corriente; o el primer núcleo de procesamiento 21 realiza el control de bucle de posición y el control de bucle de velocidad, el segundo núcleo de procesamiento 22 realiza el control de bucle de corriente. La asignación de tareas se determina según los requisitos de la tasa de cálculo de bucle y las prestaciones de la plataforma de hardware.

El segundo núcleo de procesamiento 22 calcula la corriente por la que el motor puede alcanzar la posición, la velocidad, la aceleración, la fuerza o el momento de fuerza requeridos de los valores dados de motores y emite una señal de accionamiento (como una señal PWM) hacia dispositivos de potencia tales como dispositivos IGBT, dispositivos IPM según los resultados calculados para accionar los motores.

En una forma de realización, el segundo núcleo de procesamiento 22 ejecuta un sistema operativo diferente al del primer núcleo de procesamiento 21. Por ejemplo, el segundo núcleo de procesamiento 22 ejecuta un sistema operativo más optimizado. En otra forma de realización, el segundo núcleo de procesamiento 22 no ejecuta ningún sistema operativo.

En algunas formas de realización, el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 son núcleos ARM. Por ejemplo, cada núcleo de procesamiento es un ARM Cortex-A9 MPCore, 4000 MIPS.

Convencionalmente, en una arquitectura de multiprocesamiento asimétrico, los núcleos de procesamiento no comparten una memoria caché. A diferencia de la técnica anterior, en la forma de realización de la presente invención está previsto compartir una memoria compartida a través de la memoria caché en una arquitectura de multiprocesamiento asimétrico. Aunque el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 utilizan la arquitectura de multiprocesamiento asimétrico, comparten todavía la memoria caché 23 y comparten en consecuencia la memoria compartida a través de la memoria caché.

Con el fin de implementar dicha compartición, el controlador está configurado para realizar las operaciones siguientes antes del control de movimiento y el control de motor:

En primer lugar, iniciar un sistema operativo Linux en el primer procesador y el segundo procesador por multiprocesamiento simétrico. Definir que un bloque de memoria sea la memoria compartida, asignar la memoria compartida para que sea una memoria I/O de tal manera que se mantengan las direcciones de la memoria compartida en lugar de asignarse a procesos del sistema operativo y la memoria compartida está configurada para ser cacheable. Desconectar el segundo procesador a través del sistema operativo. En este momento, el primer núcleo de procesamiento ejecuta el sistema operativo individualmente y utiliza la memoria caché L2. El primer núcleo de procesamiento es capaz de acceder a la memoria compartida de una manera cacheable. Seguidamente, reiniciar y reconfigurar el segundo procesador para hacer que el segundo procesador no ejecute normalmente el sistema operativo. El segundo núcleo de procesamiento está reconfigurado para utilizar la memoria caché L2 y acceder a la memoria compartida de una manera cacheable.

Gracias a los medios anteriores, los núcleos de procesamiento son capaces de compartir la memoria compartida en una arquitectura de multiprocesamiento asimétrico.

5 A partir de lo anterior, el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 están configurados para transferir datos a través de la memoria caché 23 (memoria caché L2) durante los periodos de control de movimiento y control de motor. Específicamente, durante el periodo de control de movimiento, el primer núcleo de procesamiento 21 escribe unos datos de control de movimiento (como valores establecidos de pares cinemáticos o valores dados de motor) en la memoria caché 23, mientras que el segundo núcleo de procesamiento 10 22 lee los datos de control de movimiento de la memoria caché 23. Como se menciona anteriormente, en las formas de realización de la presente invención, los datos de control de movimiento comprenden datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza para los pares cinemáticos o los motores, y se determinan según la asignación de tareas entre el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22. Correspondientemente, el segundo núcleo de procesamiento 15 22 escribe datos de realimentación en la memoria caché 23, mientras que el primer núcleo de procesamiento 21 lee la realimentación de la memoria caché 23. Los datos de realimentación comprenden datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza de los pares cinemáticos o los motores. Cuando los datos de control de movimiento son datos de posición, entonces los datos de realimentación comprenden generalmente datos de posición, y pueden comprender además datos relacionados con la velocidad, 20 datos de fuerza y datos de momento de fuerza. Cuando los datos de control de movimiento son datos relacionados con la velocidad, entonces los datos de realimentación comprenden datos relacionados con la velocidad y pueden comprender además datos de posición, datos de fuerza y datos de momento de fuerza. Se entiende que los datos de realimentación pueden no relacionarse tampoco con los datos de control de movimiento. Por ejemplo, los datos de realimentación pueden comprender normalmente uno o más de entre datos de posición, datos de velocidad, 25 datos de aceleración, datos de fuerza y datos de momento de fuerza. En algunas aplicaciones, los datos de realimentación pueden ser datos de posición o datos de orientación del objeto controlado (como la orientación o la velocidad de un UAV o electromóvil).

30 Para un sistema de control de un aparato de múltiples ejes que requiere una trayectoria precisa y regular, se espera que la parte de control de movimiento del sistema de control pueda transmitir de forma fiable los valores establecidos de pares cinemáticos o valores dados de motores a la parte de control de motor en un periodo de tiempo corto y consistente. En la forma de realización, el primer núcleo de procesamiento y el segundo núcleo de procesamiento trabajan en una arquitectura de multiprocesamiento asimétrico de manera que el segundo núcleo de procesamiento para el control de motor pueda ejecutar un sistema operativo simple diferente al del primer 35 núcleo de procesamiento o no ejecutar un sistema operativo mientras los dos núcleos de procesamiento comparten la memoria caché, asegurando así una transmisión de datos de alta velocidad con altas prestaciones en tiempo real (altas prestaciones en tiempo real significa baja latencia y baja fluctuación).

40 La figura 3 es un diagrama que ilustra el funcionamiento del controlador según la primera forma de realización de la presente invención. Como se muestra en la figura 3, el primer núcleo de procesamiento 21 genera y emite los datos de control de movimiento hacia el segundo núcleo de procesamiento 22. El segundo núcleo de procesamiento 22 realiza el control de motor según los datos de control de movimiento y emite una señal de accionamiento hacia el accionador de motor 31, el accionador de motor 31 emite corriente para controlar la rotación del motor 32. Los primeros datos de realimentación recogidos del motor 32 se transmiten al segundo núcleo de 45 procesamiento 22. El segundo núcleo de procesamiento 22 emite segundos datos de realimentación hacia el primer núcleo de procesamiento 21. Los primeros datos de realimentación pueden comprender uno o más de entre datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y datos de momento de fuerza. La segunda realimentación puede obtenerse completamente a partir de los primeros datos de realimentación o puede no obtenerse totalmente a partir de los primeros datos de realimentación. Por ejemplo, los segundos datos de realimentación pueden comprender datos generados por el segundo núcleo de procesamiento 22. Como se muestra en la figura 3, la comunicación entre el primer núcleo de procesamiento 21 y el segundo núcleo de 50 procesamiento 22 se materializa a través de la memoria caché 23 como se muestra en la figura 2. Cuando se requiera, los primeros datos de realimentación o los segundos datos de realimentación pueden no ser necesarios en control de bucle local o control de bucle completo, lo que no afectará a la función y las prestaciones básicas de todo el sistema de control.

55 Puesto que acceder a una memoria caché es mucho más rápido que acceder a una memoria principal, se mejora ampliamente la velocidad de funcionamiento del primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22. Particularmente, el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 están integrados en un único chip SOC que presenta una excelente capacidad de procesamiento de datos, pudiendo materializarse así una transmisión de datos de alta velocidad dentro del chip SOC. En la forma de realización que emplea dos núcleos ARM-Contex A9 como el primer y el segundo núcleos de procesamiento, el controlador funciona con una latencia media de 19 μ s y una latencia máxima de 59 μ s. El controlador de movimiento y el controlador de motor pueden transmitir datos a una tasa de 10 kb (aproximadamente, volumen de datos para 60 motores de 8 ejes) por microsegundo. El bucle de posición se actualiza cada 200 microsegundos, mientras la fluctuación de datos está por debajo de 1 microsegundo, esto es, la tasa de fluctuación es de 0,5%.

En una forma de realización, el controlador 20 comprende además una memoria principal que contiene la memoria compartida 30.

5 La presente invención proporciona también un procedimiento para cambiar el multiprocesamiento simétrico al multiprocesamiento asimétrico, de manera que se implemente la “compartición de la memoria compartida en una arquitectura de multiprocesamiento asimétrico” antes mencionada.

10 Según el procedimiento, el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 arrancan y ejecutan el sistema operativo Linux bajo un modo de multiprocesamiento simétrico. El primer y segundo núcleos de procesamiento ejecutan el sistema operativo. Sin embargo, dicho estado de funcionamiento no es normal y solo ocurre cuando el controlador está encendido. A continuación, se especifica que un bloque de memoria sea una memoria compartida y se asigna para que sea una memoria I/O. Como resultado, las direcciones de la memoria compartida se mantienen sin asignarse a otros procesos por el sistema operativo, y la memoria compartida se especifica como cacheable. A continuación, el sistema operativo desconecta uno de los núcleos de procesamiento, tal como el segundo núcleo de procesamiento 22. En este momento, el primer núcleo de procesamiento ejecuta individualmente el sistema operativo y realiza un acceso a la memoria caché L2. El primer núcleo de procesamiento realiza también un acceso cacheable a la memoria compartida. Mientras tanto, el segundo núcleo de procesamiento 22 no ejecuta un sistema operativo diferente en comparación con el primer núcleo de procesamiento 21 ni ejecuta ningún sistema operativo. Sin embargo, el sistema operativo supervisará la memoria caché y la memoria compartida en un modo de multiprocesamiento simétrico puesto que el sistema operativo no es consciente del reajuste del segundo núcleo de procesamiento 22. Por tanto, el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 comparten la memoria caché, así como la memoria compartida en una arquitectura asimétrica. Códigos específicos para las operaciones anteriores pueden almacenarse en una memoria no volátil (tal como una tarjeta SD, EMMC o diversas memorias flash) y ejecutarse por el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 cuando el controlador 20 está encendido.

La figura 7 es un diagrama de flujo que ilustra un procedimiento para realizar el control de movimiento y el control de motor por un controlador integrado según una forma de realización de la presente invención. Como se muestra en la figura 7, el procedimiento comprende las etapas siguientes:

35 S701, arrancar el sistema operativo Linux en el primer procesador y el segundo procesador en un modo de multiprocesamiento simétrico. En la forma de realización, el primer procesador es el primer núcleo de procesamiento 21, el segundo procesador es el segundo núcleo de procesamiento 22. Especificar un bloque de memoria como la memoria compartida y asignar la memoria compartida para que sea una memoria I/O, de tal manera que las direcciones de la memoria compartida se mantengan sin asignarse a otros procesos por el sistema operativo y la memoria compartida se especifica como cacheable.

40 S702, desconectar el segundo procesador por medio del sistema operativo. En este momento, el primer procesador ejecuta individualmente el sistema operativo y realiza un acceso a la memoria caché L2. El primer núcleo de procesamiento realiza también un acceso cacheable a la memoria compartida.

45 S703, reiniciar y reasignar el segundo procesador para hacer que el segundo procesador no ejecute normalmente el sistema operativo. Reasignar el segundo procesador para hacer que realice un acceso a la memoria caché L2 y realice un acceso cacheable a la memoria compartida. Ahora, el sistema se convierte en una arquitectura de multiprocesamiento asimétrico. Como se menciona anteriormente, el segundo procesador puede ejecutar otro sistema operativo, tal como un sistema operativo optimizado en comparación con el primer procesador, o el segundo procesador puede no ejecutar ningún sistema operativo.

50 S704, durante los periodos de control de movimiento por el primer procesador y de control de motor por el segundo procesador, intercambiando datos el primer procesador y el segundo procesador a través de la memoria caché L2.

La figura 8 es un diagrama que ilustra el primer procesador y el segundo procesador cargando códigos según una forma de realización de la presente invención. Como se muestra en la figura 8, los códigos informáticos que se ejecutan por el primer procesador y el segundo procesador (primer núcleo de procesamiento y segundo núcleo de procesamiento) se almacenan en una memoria no volátil 800. Bajo el modo de multiprocesamiento simétrico, el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 inicia la carga de códigos a partir de la dirección A1. Las direcciones A1 a A2 almacenan códigos informáticos para el sistema operativo y el control de movimiento. Además, las direcciones A1 a A5 pueden almacenar también códigos informáticos para desconectar, reiniciar y reasignar el segundo núcleo de procesamiento 22. Después de que el primer núcleo de procesamiento 21 y el segundo núcleo de procesamiento 22 ejecuten el sistema operativo, el segundo núcleo de procesamiento 22 se reasigna e inicia la carga de códigos desde la dirección A6. Las direcciones A6 a A8 pueden almacenar códigos informáticos para el control de motor por el segundo núcleo de procesamiento 22.

65 El control de bucle de velocidad y el control de bucle de corriente de la tarea de control de motor requieren ambas elevadas características en tiempo real para asegurar la rotación regular y continua del motor. Sin embargo,

el sistema operativo interrumpe siempre el procesamiento en curso de vez en cuando para tratar con diferentes tareas, lo que provocará la latencia del envío de los resultados de cálculo desde el control de motor durante la interrupción si el primer núcleo de procesamiento realiza el control de bucle de velocidad y el control de bucle de corriente, afectando así a la característica en tiempo real. Por tanto, en una forma de realización, el segundo núcleo de procesamiento realiza el control de bucle de velocidad y el control de bucle de corriente sin ejecutar el sistema operativo. Además, el segundo núcleo de procesamiento permite siempre la interrupción, esto es, el segundo núcleo de procesamiento no necesita realizar la habilitación/deshabilitación de la respuesta de interrupción.

Además, el segundo núcleo de procesamiento 22 puede utilizar una cola sin bloqueo para asegurar la consistencia de los datos.

El controlador de la forma de realización puede utilizarse en diversos sistemas de control tales como robots eléctricos, máquinas herramienta de control numérico, aeronaves de tipo multirrotor eléctrico, electromóviles, prótesis mecánicas, manos mecánicas, vehículos móviles eléctricos, etc.

El controlador de la forma de realización adopta un sistema de integración de accionamiento y control que reduce la redundancia de hardware. Además, puesto que el sistema de integración de accionamiento y control puede materializarse utilizando un único chip SOC con elementos de circuito periféricos necesarios, se reduce también la ocupación de hardware.

Además, puesto que el sistema de integración de accionamiento y control está incorporado en un único chip, se realiza una transmisión de datos en el interior del chip sin interferencia interna, lo que asegura una transmisión de datos apropiada y fiable.

Además, el sistema de integración de accionamiento y control incorporado en un único chip puede asegurar altas prestaciones en tiempo real. El SOC proporciona alta velocidad de funcionamiento y alta velocidad de comunicación entre el primer y segundo procesadores a través de la memoria caché. El sistema de integración de accionamiento y control de chip único puede utilizarse en aplicaciones más complicadas como aparatos móviles, tales como vehículos tripulados y UVA, etc.

Segunda forma de realización

La figura 4 es un diagrama de circuito de un controlador de acuerdo con una segunda forma de realización de la presente invención. Como se muestra en la figura 4, el controlador integrado 40 para control de movimiento y control de motor comprende un primer núcleo de procesamiento 41, un segundo núcleo de procesamiento 42, una memoria caché 43, un dispositivo lógico programable 44 y una memoria compartida 50. El primer núcleo de procesamiento 41 está configurado para ejecutar un sistema operativo y realizar por lo menos el control de movimiento. El segundo núcleo de procesamiento 42 está configurado para realizar el control de motor y normalmente no ejecutar el sistema operativo. La memoria caché, como se conoce por los expertos en la materia, es una memoria entre un procesador y una memoria principal. La memoria caché funciona más rápido que la memoria principal a una velocidad casi próxima al procesador. La memoria caché 43 puede ser una memoria caché Nivel 2 (memoria caché L2) para un procesador. Se entiende que la memoria caché 43 puede ser de otros niveles. La memoria caché 43 está acoplada con el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42. La memoria compartida 50 es un bloque de memoria cacheable dedicado con direcciones fijas, que se especifica por el sistema operativo ejecutado en el primer núcleo de procesamiento 41. La memoria compartida 50 mapea sobre la memoria caché 43. El primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 están configurados para compartir la memoria compartida 50 y, en consecuencia, realizar una transmisión de datos a través de la memoria caché 43 durante los periodos de control de movimiento y control de motor. Por ejemplo, la memoria compartida 50 mapea sobre la memoria caché 43 a través del mapeo de direcciones. Cuando la memoria compartida 50 mapea sobre la memoria caché 43, se realiza una transmisión de datos en la memoria caché 43 que mejora la tasa de transmisión en 8-10 veces en comparación con el intercambio de datos directo en una memoria física. El dispositivo lógico programable 44 está acoplado con el segundo núcleo de procesamiento 42 para realizar el control de motor junto con el segundo núcleo de procesamiento 42.

En la arquitectura de procesador actual, múltiples núcleos de procesamiento que comparten la misma memoria caché están integrados usualmente en un mismo chip de procesamiento. Por tanto, en la forma de realización, el primer núcleo de procesamiento 41, el segundo núcleo de procesamiento 42 y la memoria caché 43 están integrados en un mismo chip, tal como un chip de procesamiento.

En la forma de realización, el primer núcleo de procesamiento 41 ejecuta el sistema operativo y realiza por lo menos el control de movimiento. El sistema operativo puede supervisar el funcionamiento de todo el controlador. El sistema operativo puede realizar también tareas de aplicación específicas, como la tarea de control de movimiento. La tarea de control de movimiento pretende calcular movimientos diana de pares cinemáticos de un objeto controlado en cada momento. La implementación del control de movimiento puede ser de varias formas. Por ejemplo, según los requisitos de aplicación, los diferentes algoritmos de planificación de trayectoria y el software de análisis de tarea pueden desarrollarse para realizar diversas planificaciones de trayectorias y

conversiones de bloque de unión. Los algoritmos específicos para dichas planificaciones de trayectorias y conversiones de bloque de unión pueden ser también ampliamente diferentes según las aplicaciones. Por ejemplo, los algoritmos cinemáticos directos y los algoritmos cinemáticos inversos pueden aplicarse en robots industriales; pueden aplicarse algoritmos de control de vuelo de UAV en UAV. Es obvio para los expertos en la materia implementar la función de control de movimiento cuando se requiera.

El primer núcleo de procesamiento 41 calcula uno o más de entre la posición, la velocidad, la aceleración, la fuerza y el momento de fuerza de los pares cinemáticos del objeto controlado por el controlador 40 en cada momento, que se toman como valores establecidos de pares cinemáticos. En la forma de realización, el primer núcleo de procesamiento 41 convierte también los valores establecidos de pares cinemáticos en valores dados de motores de las juntas del objeto controlado en cada momento. Los valores dados de motores incluyen posición, velocidad, aceleración, fuerza, momento de fuerza de los motores o su combinación. El primer núcleo de procesamiento 41 emite los valores dados de motores hacia el segundo núcleo de procesamiento 42 en un tiempo predeterminado. Alternativamente, en otras formas de realización, el primer núcleo de procesamiento 41 puede no realizar la conversión de valores establecidos de pares cinemáticos a valores dados de los motores. El primer núcleo de procesamiento 41 emite directamente los valores establecidos de pares cinemáticos hacia el segundo núcleo de procesamiento 42, a continuación, el segundo núcleo de procesamiento 42 los convierte en los valores dados de motores y realiza el control de motor correspondiente.

A diferencia de la primera forma de realización, el dispositivo lógico programable 44 se introduce en el controlador según la forma de realización. El dispositivo lógico programable tiene una capacidad de cálculo paralela mayor lo que es una gran ventaja en el control de múltiples ejes. El dispositivo lógico programable 44 está integrado junto con el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 en un mismo chip. El dispositivo lógico programable 44 está acoplado con el segundo núcleo de procesamiento 42 para funcionar junto con este. En la forma de realización, el dispositivo lógico programable 44 está acoplado con el segundo núcleo de procesamiento 42 a través de una interfaz. Algunos chips SOC existentes, como Cyclone V proporcionados por Altera Corporation, ofrecen dicho interfaz. En diversas formas de realización, el dispositivo lógico programable puede ser un FPGA.

En la forma de realización, el segundo núcleo de procesamiento 42 normalmente no ejecuta el mismo sistema operativo con el primer núcleo de procesamiento 41. Por el contrario, el segundo núcleo de procesamiento 42 y el dispositivo lógico programable 44 constituyen conjuntamente el controlador de motor para realizar la tarea de control de motor. Según su asignación, el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 funcionan normalmente en una arquitectura de multiprocesamiento asimétrico. El controlador de motor controla el accionador de motor según los valores dados de los motores para accionar los motores a fin de que cumplan de manera rápida y estable los requisitos. El control de motor implica el control de bucle de corriente, el control de bucle de velocidad y/o el control de bucle de posición. Pueden implementarse diferentes controles de bucle según diferentes valores dados de motores. En general, cuando el valor dado es un valor posicional se requiere que se calculen los tres bucles; cuando el valor dado es un valor relacionado con la velocidad (valor de velocidad o valor de aceleración), se requiere que se calculen el bucle de corriente y el bucle de velocidad; cuando el valor dado es un valor de fuerza o valor de momento de fuerza, se requiere que se calcule el bucle de corriente. En consecuencia, el controlador de motor realiza alternativamente el control de bucle de corriente, o la combinación del control de bucle de corriente y el control de bucle de velocidad, o la combinación del control de bucle de corriente, el control de bucle de velocidad y el control de bucle de posición para lograr la tarea de control de motor. Además, la asignación de tareas entre el primer núcleo de procesamiento 41 y el controlador de motor puede modificarse. Por ejemplo, el primer núcleo de procesamiento 41 realiza el control de bucle de posición, el controlador de motor realiza el control de bucle de velocidad y el control de bucle de corriente; o el primer núcleo de procesamiento 41 realiza el control de bucle de posición y el control de bucle de velocidad, el controlador de motor realiza el control de bucle de corriente. La asignación de tareas se determina según los requisitos de la tasa de cálculo de bucle y las prestaciones de la plataforma de hardware.

El controlador de motor calcula la corriente por la que el motor puede alcanzar la posición, la velocidad, la aceleración, la fuerza o el momento de fuerza requeridos de los valores dados de motores y emite una señal de accionamiento (como la señal PWM) hacia dispositivos de potencia tales como dispositivos IGBT, dispositivos IPM según los resultados calculados para accionar los motores.

En una forma de realización, el segundo núcleo de procesamiento 42 ejecuta un sistema operativo diferente al del primer núcleo de procesamiento 41. Por ejemplo, el segundo núcleo de procesamiento 42 ejecuta un sistema operativo más optimizado. En otra forma de realización, el segundo núcleo de procesamiento 22 no ejecuta ningún sistema operativo.

En diversas formas de realización, la tarea de control de movimiento se asigna entre el segundo núcleo de procesamiento 42 y el dispositivo lógico programable 44 que constituyen el controlador de motor como un todo. Cuando el segundo núcleo de procesamiento 42 y el dispositivo lógico programable 44 funcionan conjuntamente para realizar el control de bucle de posición, el control de bucle de velocidad y el control de bucle de corriente, la asignación de tareas puede ser como sigue: el segundo núcleo de procesamiento 42 realiza el control de bucle de

posición mientras el dispositivo lógico programable 44 realiza el control de bucle de velocidad y el control de bucle de corriente; o el segundo núcleo de procesamiento 42 realiza el control de bucle de posición y el control de bucle de velocidad mientras el dispositivo lógico programable 44 realiza el control de bucle de corriente. Incluso con una configuración avanzada de hardware, el dispositivo lógico programable 44 realiza individualmente el control de bucle de posición, el control de bucle de velocidad y el control de bucle de corriente. Cuando la combinación del segundo núcleo de procesamiento 42 y el dispositivo lógico programable 44 realizan solo parte del control de los tres bucles, tales como el control de bucle de velocidad y el control de bucle de corriente, la asignación de tareas entre el segundo núcleo de procesamiento 42 y el dispositivo lógico programable 44 puede ajustarse correspondientemente. Por ejemplo, el segundo núcleo de procesamiento 42 realiza el control de bucle de velocidad y el dispositivo lógico programable 44 realiza el control de bucle de corriente.

Para un aparato o sistema de múltiples ejes que requiera una trayectoria precisa y regular, se desea que todos los ejes sean controlados simultáneamente y de forma síncrona, de manera que se asegure un movimiento resultante apropiado. En consecuencia, cuando los valores establecidos de pares cinemáticos o valores dados de los motores se calculan y se transmiten al controlador de motor dentro de un periodo predeterminado, se espera que la parte de control de movimiento transmita de forma fiable los valores establecidos de pares cinemáticos o valores dados de los motores a la parte de control de motor en un periodo de tiempo corto y consistente.

En la forma de realización, se proporciona un procedimiento de "compartición de la memoria compartida en una arquitectura de multiprocesamiento asimétrico". El primer núcleo de procesamiento y el segundo núcleo de procesamiento funcionan en una arquitectura de multiprocesamiento asimétrico de manera que el segundo núcleo de procesamiento para el control de motor puede ejecutar un sistema operativo simple diferente al del primer núcleo de procesamiento o no ejecutar un sistema operativo mientras los dos núcleos de procesamiento comparten la memoria caché, asegurando así una transmisión de datos de alta velocidad con elevadas prestaciones en tiempo real (elevadas prestaciones en tiempo real significa baja latencia y baja fluctuación). En otras palabras, aunque el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 utilizan una arquitectura de multiprocesamiento asimétrico, comparten la memoria caché 43 y comparten además la memoria compartida a través de la memoria caché.

El procedimiento comprende las etapas siguientes:

Antes de realizar el control de movimiento y el control de motor, arrancar el sistema operativo Linux en el primer procesador y el segundo procesador en un modo de multiprocesamiento simétrico. Especificar un bloque de memoria como la memoria compartida y asignar la memoria compartida para que sea una memoria I/O, de tal manera que las direcciones de la memoria compartida se mantengan sin asignarse a otros procesos por el sistema operativo y la memoria compartida se especifica como cacheable.

Desconectar el segundo procesador por medio del sistema operativo.

En este momento, el primer procesador ejecuta individualmente el sistema operativo y realiza un acceso a la memoria caché L2. El primer procesador realiza también un acceso cacheable a la memoria compartida.

Reiniciar y reasignar el segundo procesador para hacer que el segundo procesador normalmente no ejecute el sistema operativo. Reasignar el segundo procesador para hacer que realice un acceso a la memoria caché L2 y realice un acceso cacheable a la memoria compartida.

Por los medios anteriores, los núcleos de procesamiento son capaces de compartir la memoria compartida en una arquitectura de multiprocesamiento asimétrico. Por lo anterior, el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 están configurados para transferir datos a través de la memoria caché 43 durante los periodos de control de movimiento y control de motor. Específicamente, durante el periodo de control de movimiento, el primer núcleo de procesamiento 41 escribe los datos de control de movimiento (como los valores establecidos de pares cinemáticos o los valores dados de motor) en la memoria caché 43, mientras que el segundo núcleo de procesamiento 42 lee los datos de control de movimiento de la memoria caché 43. Como se menciona anteriormente, en las formas de realización de la presente invención, los datos de control de movimiento comprenden datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza para los pares cinemáticos o los motores, y se determinan según la asignación de tareas entre el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42. Correspondientemente, el segundo núcleo de procesamiento 42 escribe los datos de realimentación en la memoria caché 43, mientras el primer núcleo de procesamiento 41 lee la realimentación de la memoria caché 43. Los datos de realimentación comprenden datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza de los pares cinemáticos o los motores. Cuando los datos de control de movimiento son datos de posición, entonces los datos de realimentación comprenden generalmente datos de posición y pueden comprender además datos de velocidad, datos de aceleración, datos de fuerza y datos de momento de fuerza. Cuando los datos de control de movimiento son datos de velocidad, entonces los datos de realimentación comprenden datos relacionados con la velocidad, y pueden comprender además datos de posición, datos de fuerza y datos de momento de fuerza. Se entiende que los datos de realimentación pueden no relacionarse tampoco con los datos de control de movimiento.

Por ejemplo, los datos de realimentación pueden comprender normalmente uno o más de entre datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y datos de momento de fuerza. En algunas aplicaciones, los datos de realimentación pueden ser datos de posición o datos de orientación del objeto controlado (como la orientación o la velocidad de un UAV o electromóvil).

5

La figura 5 es un diagrama que ilustra el funcionamiento del controlador de acuerdo con la segunda forma de realización de la presente invención. Como se muestra en la figura 5, el primer núcleo de procesamiento 41 genera y emite los datos de control de movimiento hacia el segundo núcleo de procesamiento 42. El segundo núcleo de procesamiento 42 y el dispositivo lógico programable 44 generan y emiten la señal de accionamiento hacia el accionador de motor 51 según los datos de control de movimiento. El accionador de motor 51 emite corriente para controlar la rotación del motor 52. En la forma de realización, si el segundo núcleo de procesamiento 42 realiza el control de bucle de posición y el control de bucle de velocidad, entonces el dispositivo lógico programable 44 realiza el control de bucle de corriente. Los primeros datos de realimentación recogidos del motor 52 se transmiten al segundo núcleo de procesamiento 52. El segundo núcleo de procesamiento 42 emite segundos datos de realimentación hacia el primer núcleo de procesamiento 41. La primera realimentación puede comprender uno o más de entre datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y datos de momento de fuerza. La segunda realimentación puede completarse a partir de los primeros datos de realimentación o puede no completarse totalmente a partir de los primeros datos de realimentación. Por ejemplo, los segundos datos de realimentación pueden comprender datos generados por el segundo núcleo de procesamiento 42. Como se muestra en la figura 3, la comunicación entre el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 se materializa a través de la memoria caché 43, como se muestra en la figura 4. Cuando se requiera, los primeros datos de realimentación o los segundos datos de realimentación pueden no ser necesarios en un control de bucle local o un control de bucle completo, lo que no afectará a la función básica y las prestaciones de todo el sistema de control.

10

15

20

25

Puesto que el acceso a una memoria caché es mucho más rápido que el acceso a una memoria principal, la velocidad de funcionamiento del primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 se mejora ampliamente. En particular, el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 están integrados en un único chip SOC, que originalmente tiene en el mismo un dispositivo lógico programable. Para un aparato o sistema de múltiples ejes que requiere una trayectoria precisa y regular, se desea que todos los ejes estén controlados simultáneamente y de manera síncrona, de modo que se asegure un movimiento resultante apropiado. En consecuencia, el controlador de motor utiliza una CPU (el segundo procesador) y un FPGA (dispositivo lógico programable) para materializar el control de motor de 8 ejes que adopta completamente las ventajas de una mayor capacidad de cálculo paralelo del FPGA (dispositivo lógico programable).

30

35

La presente invención proporciona también un procedimiento para cambiar el multiprocesamiento simétrico a multiprocesamiento asimétrico, de manera que se implemente la antes mencionada "compartición de la memoria compartida en una arquitectura de multiprocesamiento asimétrico".

40

El procedimiento puede referirse a la descripción en la primera forma de realización y la figura 7 que no se detalla en la presente memoria.

45

En algunas formas de realización, el dispositivo lógico programable 44 puede compartir también la memoria compartida 50 con el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42, lo que no se detalla en la presente memoria.

50

La fluctuación de datos acaecida durante el periodo en que el dispositivo lógico programable 44 realimenta los datos de velocidad y los datos de posición al segundo núcleo de procesamiento 42 puede provocar un cambio de corriente abrupto en el control de bucle de velocidad, que degradará las prestaciones del motor. Puesto que el dispositivo lógico programable 44 y el sistema de procesamiento (que comprende el primer y segundo núcleos de procesamiento 41, 42) son dos elementos de cálculo independientes, son problemas críticos que deben resolverse la manera en que se mantiene la sincronización de datos entre ellos y la forma en que se proporciona realimentación de datos continua y completa desde el dispositivo lógico programable 44.

55

En una forma de realización, como se muestra en la figura 9, los dos elementos de cálculo, el sistema de procesamiento y el dispositivo lógico programable 44 utilizan un mismo reloj externo (oscilación de cristal) para asegurar que ambos estén sincronizados por reloj con una misma base de tiempo. Por tanto, el periodo de tiempo para que el sistema de procesamiento lea datos del dispositivo lógico programable 44 es un cierto tiempo relativo, y se introduce un tiempo escalonado entre el ciclo de actualización de datos del dispositivo lógico programable 44 y el ciclo de actualización de datos del sistema de procesamiento. Por tanto, el FPGA actualiza datos en un cierto tiempo, y el sistema de procesamiento lee datos después de dicho cierto tiempo.

60

65

En una forma de realización, toda la señal de reloj interna del dispositivo lógico programable 44 está dividida (por ejemplo, por un divisor de frecuencia PLL) de una misma fuente de reloj. En consecuencia, todos los datos del dispositivo lógico programable 44 se transmiten de forma síncrona. Además, el procesador (el segundo núcleo de procesamiento 42), el bus, el dispositivo lógico programable 44 utilizan el mismo reloj.

En una forma de realización, como se muestra en la figura 10, el dispositivo lógico programable 44 comprende dos registros 101, 102 para anclar los datos de corriente cuando el primer núcleo de procesamiento 41 y el segundo núcleo de procesamiento 42 leen los datos, de manera que se evite la pérdida de datos debido a la metaestabilidad acaecida en un diseño asíncrono. Esto se denomina también "registro en la sombra". Como se muestra en la figura 10, el dispositivo lógico programable 44 realiza internamente un cálculo matemático y almacena los datos de resultado en el registro 102. Seguidamente, los datos de resultado se almacenan adicionalmente en el registro 101. A través de dicho tictac del reloj, el sistema de procesamiento puede leer datos completos y correctos del dispositivo lógico programable. Durante el periodo de tiempo en que el sistema de procesamiento lee los datos, puede impedirse la actualización de datos en los registros del dispositivo lógico programable. En la forma de realización, los registros 101, 102 pueden ser unos biestables.

Tercera forma de realización

La figura 6 es un diagrama de circuito de un controlador según una tercera forma de realización de la presente invención. Como se muestra en la figura 6, el controlador integrado 60 para el control de movimiento y el control de motor comprende una primera combinación de núcleo de procesamiento 61, una segunda combinación de núcleo de procesamiento 62, un dispositivo lógico programable 63, una memoria caché (no mostrada) y una memoria compartida (no mostrada). La primera combinación de núcleo de procesamiento 61 está configurada para ejecutar un sistema operativo y realizar por lo menos el control de movimiento. La segunda combinación de núcleo de procesamiento 62 está configurada para realizar el control de motor y normalmente para no ejecutar el sistema operativo. La memoria caché está acoplada con la primera combinación de núcleo de procesamiento 61 y la segunda combinación de núcleo de procesamiento 62. La memoria compartida mapea sobre la memoria caché. La primera combinación de núcleo de procesamiento 61 y la segunda combinación de núcleo de procesamiento 62 están configuradas para compartir la memoria compartida y, en consecuencia, realizar una transmisión de datos a través de la memoria caché durante los periodos de control de movimiento y control de motor. A diferencia de las formas de realización antes mencionadas, la primera combinación de núcleo de procesamiento 61, hecha funcionar como el primer procesador, comprende núcleos de procesamiento 1 a $N^{\text{ésimo}}$ la segunda combinación de núcleo de procesamiento 62, hecha funcionar como el segundo procesador, comprende $N+1^{\text{ésimo}}$ a $N+K^{\text{ésimo}}$ núcleos de procesamiento. En la presente memoria, N y K son ambos números enteros positivos.

El dispositivo lógico programable 63 está acoplado con la segunda combinación de núcleo de procesamiento 62 para realizar el control de motor junto con la segunda combinación de núcleo de procesamiento 62. Haciendo referencia a la primera forma de realización, el dispositivo lógico programable 63 puede omitirse también, la segunda combinación de núcleo de procesamiento 62 puede realizar individualmente el control de motor.

Para la primera combinación de núcleo de procesamiento 61, el sistema operativo y la tarea de control de movimiento puede asignarse entre los múltiples núcleos de procesamiento. Para la segunda combinación de núcleo de procesamiento 62, la tarea de control de motor puede asignarse entre los múltiples núcleos de procesamiento.

Otros detalles de la forma de realización pueden referirse a la primera y segunda formas de realización, lo que se omite en la presente memoria.

REIVINDICACIONES

1. Controlador integrado (40) para control de movimiento y control de motor, comprendiendo el controlador integrado (40):
- 5 un primer procesador (41) configurado para ejecutar un sistema operativo y realizar por lo menos un control de movimiento, en el que el control de movimiento comprende calcular unos valores establecidos de pares cinemáticos de un objeto controlado en cada momento;
- 10 un segundo procesador (42) configurado para realizar por lo menos un control de motor, en el que el control de motor comprende controlar la rotación de motores según los valores establecidos, que comprende por lo menos cualquiera o una combinación de entre un control de bucle de corriente de motor, un control de bucle de velocidad de motor, y un control de bucle de posición de motor; en el que, el segundo procesador (42) ejecuta el mismo sistema operativo que el primer procesador (41) en el inicio, a continuación, el segundo procesador (42) se desconecta y se reinicia y se reconfigura para ejecutar un sistema operativo diferente al del primer procesador (41) o no ejecutar ningún sistema operativo después del inicio;
- 15 una memoria caché (43) acoplada con el primer procesador y el segundo procesador;
- 20 una memoria compartida (50), que es un bloque de memoria cacheable dedicado con direcciones fijas especificadas por el sistema operativo ejecutado en el primer procesador; en el que la memoria compartida (50) mapea sobre la memoria caché;
- 25 el primer procesador (41) y el segundo procesador (42) están configurados para compartir la memoria compartida (50) y, en consecuencia, realizar una transmisión de datos a través de la memoria caché durante los periodos de control de movimiento y control de motor;
- 30 el primer procesador (41), el segundo procesador (42) y la memoria caché (43) están integrados en un mismo chip.
2. Controlador integrado (40) según la reivindicación 1, caracterizado por que el controlador integrado (40) comprende asimismo un dispositivo lógico programable (44) que está acoplado con el segundo procesador (42) y configurado para realizar el control de motor junto con el segundo procesador (42).
- 35 3. Controlador integrado (40) según la reivindicación 2, caracterizado por que el primer procesador (41), el segundo procesador (42), la memoria caché (43) y el dispositivo lógico programable (44) están integrados en el mismo chip.
- 40 4. Controlador integrado (40) según una de las reivindicaciones anteriores, caracterizado por que antes del control de movimiento y el control de motor, el controlador integrado (40) está configurado para realizar las siguientes operaciones:
- 45 iniciar el primer procesador (41) y el segundo procesador (42) por multiprocesamiento simétrico;
- ejecutar el sistema operativo en el primer procesador (41) y el segundo procesador (42);
- desconectar el segundo procesador (42) y mantener el sistema operativo ejecutándose individualmente en el primer procesador (41);
- 50 reiniciar y reconfigurar el segundo procesador (42) para hacer que el segundo procesador (42) no ejecute el sistema operativo después del inicio.
- 55 5. Controlador integrado (40) según una de las reivindicaciones anteriores, caracterizado por que el primer procesador (41) está configurado para escribir unos datos de control de movimiento en la memoria caché (43) el segundo procesador (42) está configurado para leer los datos de control de movimiento de la memoria caché (43), comprendiendo los datos de control de movimiento datos de posición, datos de velocidad, datos de aceleración, datos de fuerza, y/o datos de momento de fuerza para los pares cinemáticos o los motores.
- 60 6. Controlador integrado (40) según una de las reivindicaciones anteriores, caracterizado por que el segundo procesador (42) está configurado para escribir unos datos de realimentación en la memoria caché (43), el primer procesador (41) está configurado para leer la realimentación de la memoria caché (43), comprendiendo los datos de realimentación unos datos de posición, datos de velocidad, datos de aceleración, datos de fuerza y/o datos de momento de fuerza de los pares cinemáticos o los motores.
- 65 7. Controlador integrado (40) según la reivindicación 2 o la reivindicación 3, caracterizado por que un reloj del

dispositivo lógico programable (44) y un reloj del segundo procesador (42) están sincronizados; cuando el segundo procesador (42) lee datos del dispositivo lógico programable (44) o los escribe en el mismo, el dispositivo lógico programable (44) ancla los datos leídos o escritos.

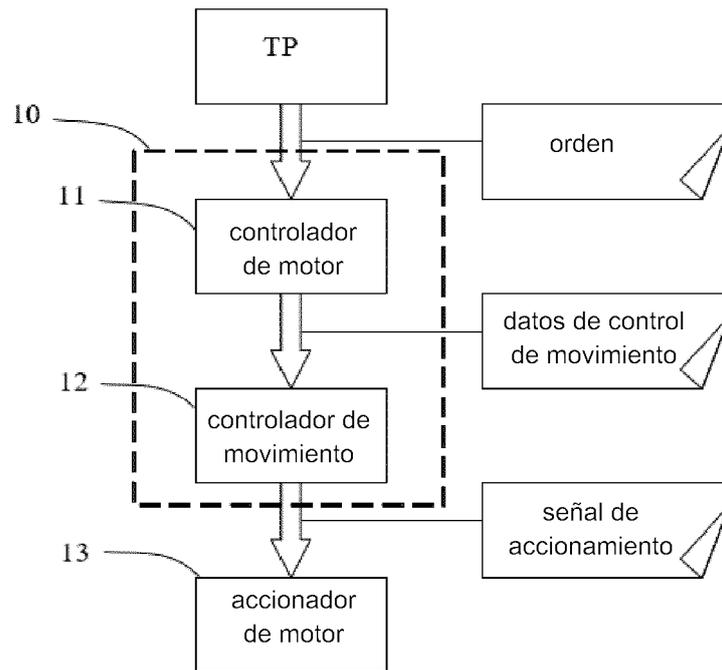


FIG. 1

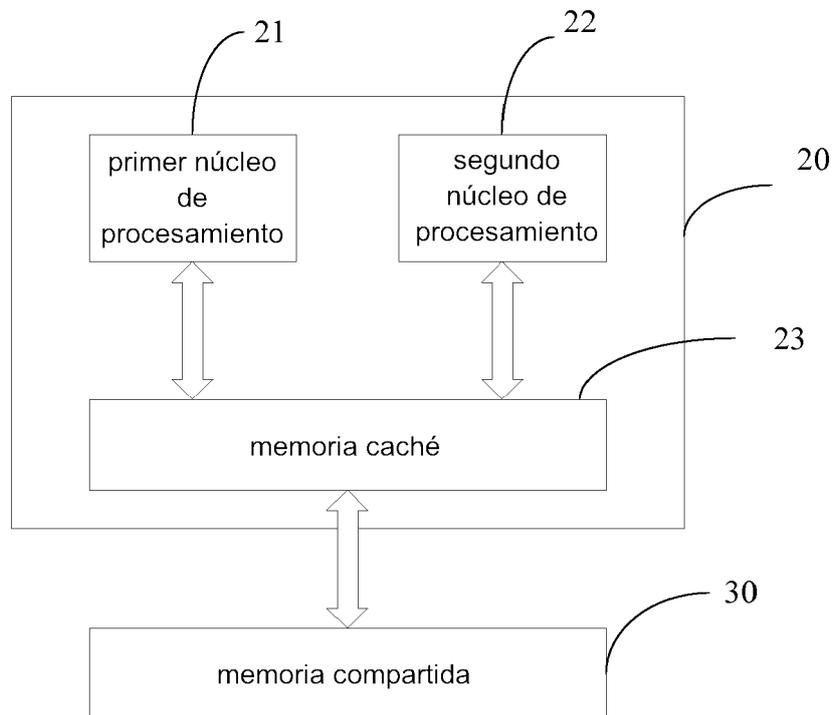


FIG.2

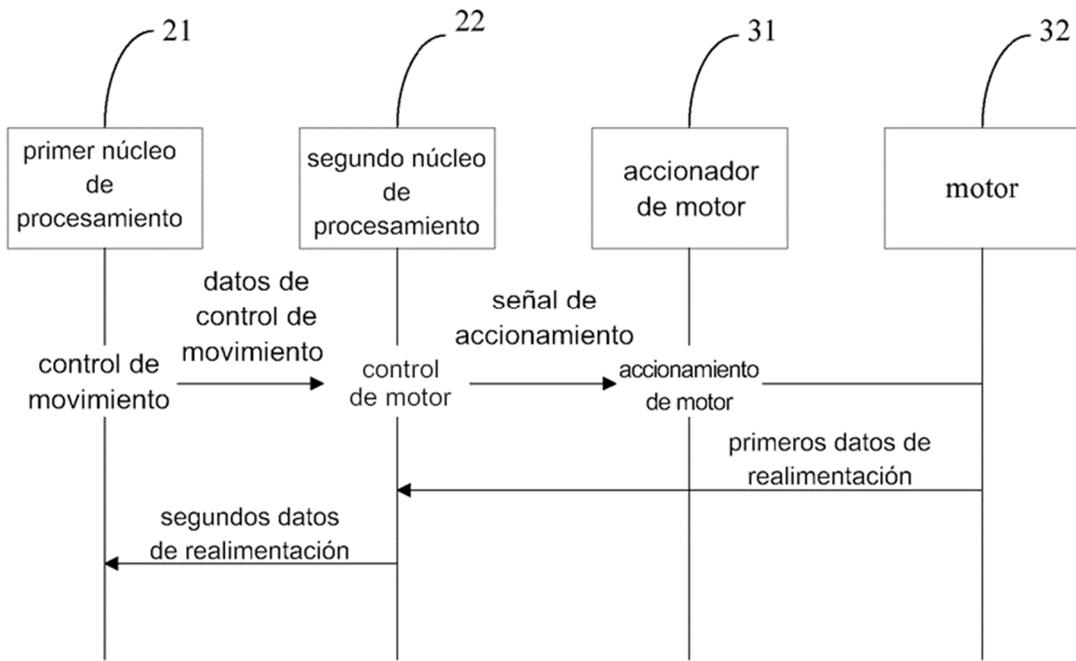


FIG.3

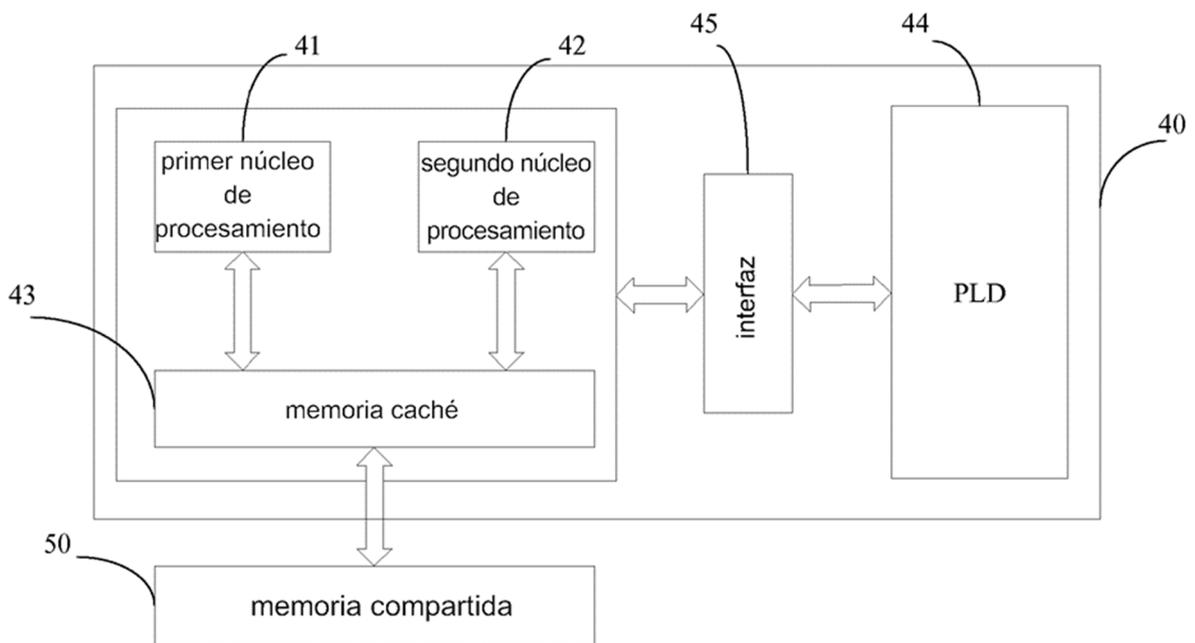


FIG. 4

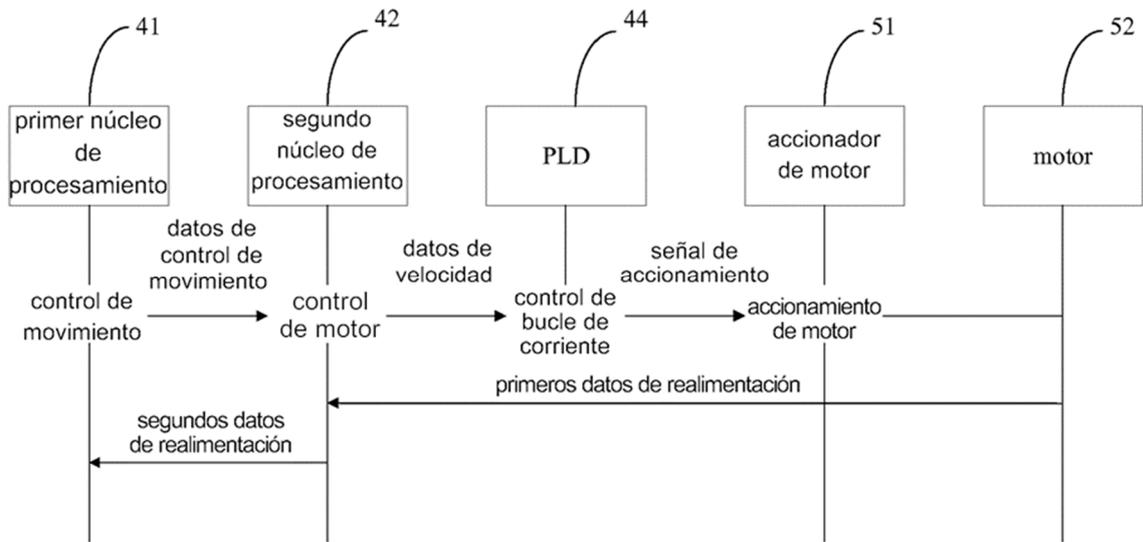


FIG. 5

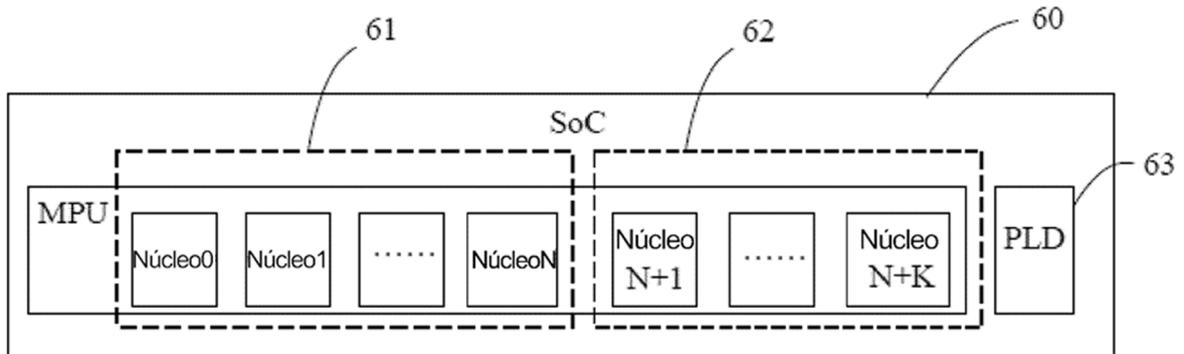


FIG. 6

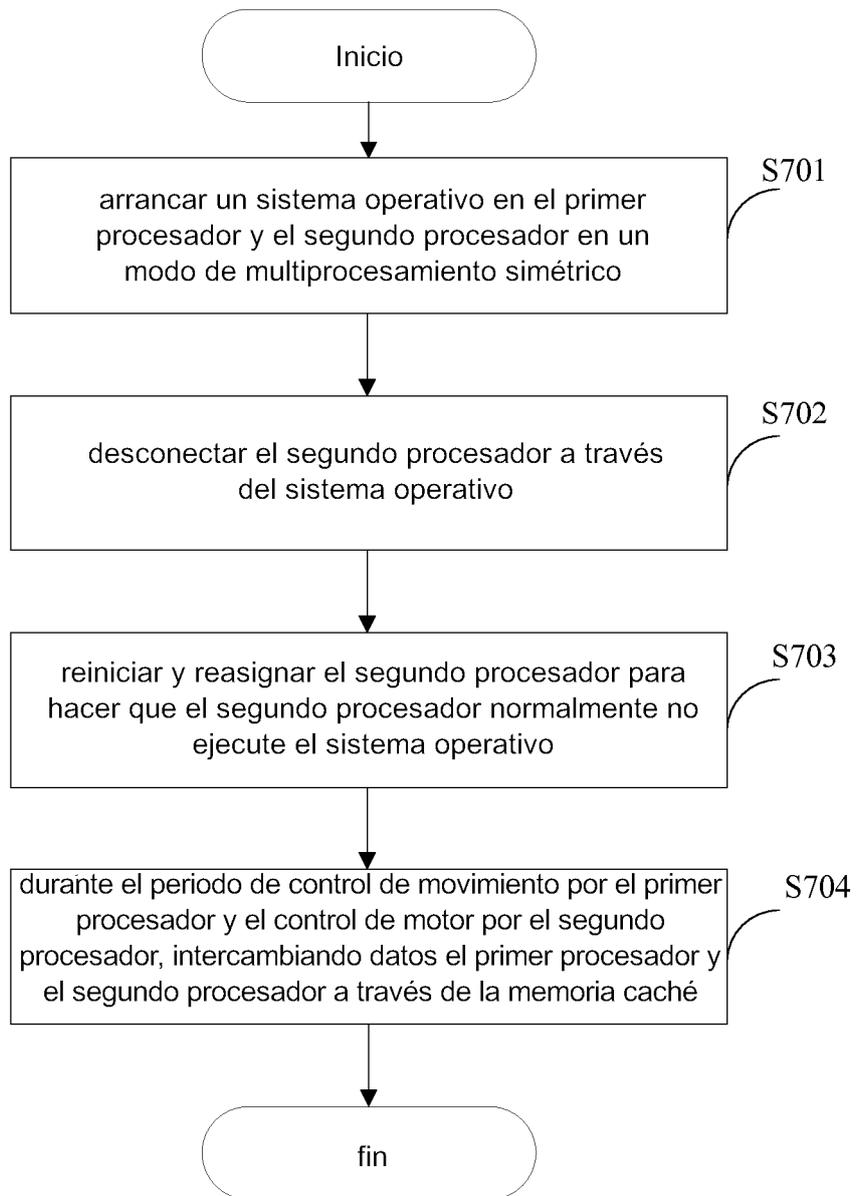


FIG. 7

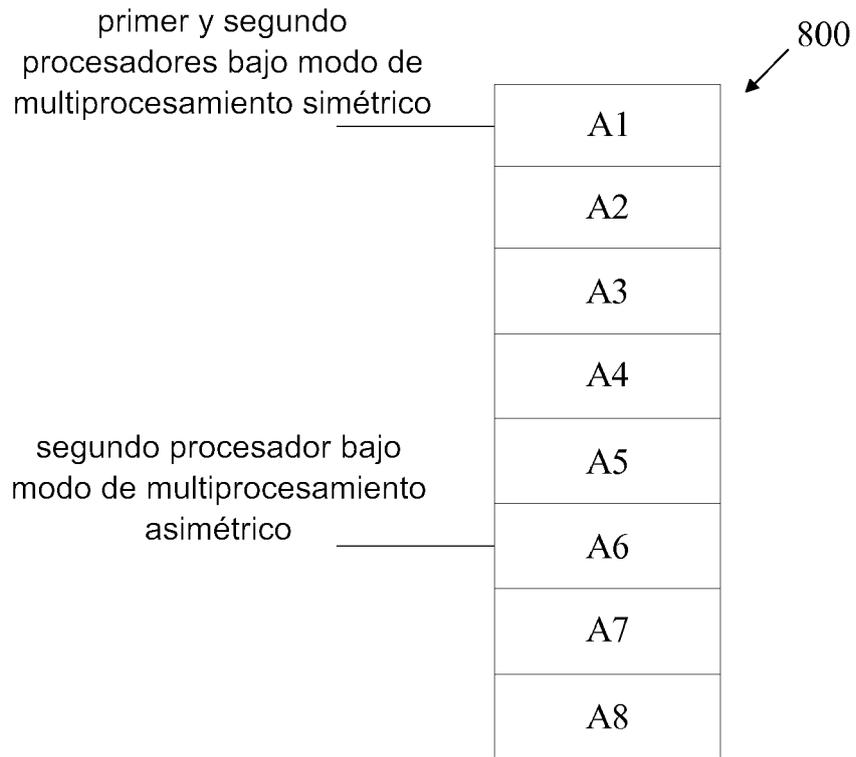


FIG. 8

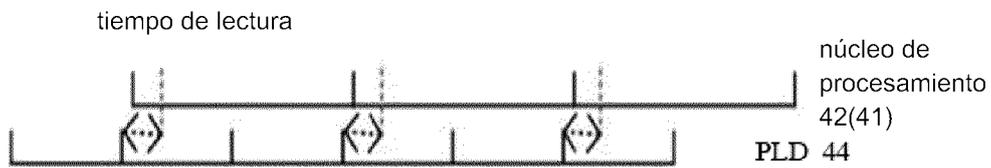


FIG. 9

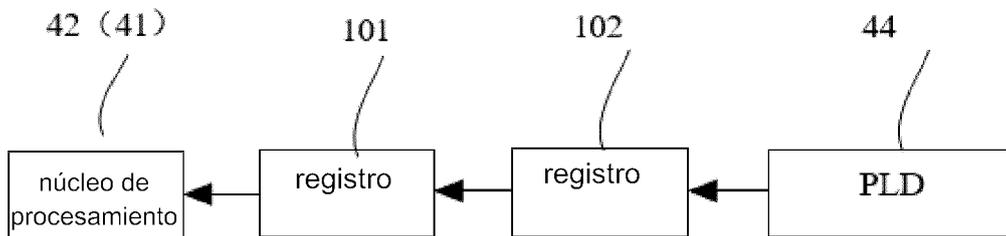


FIG. 10