

19



OFICINA ESPAÑOLA DE  
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 409 882**

51 Int. Cl.:

**G06F 11/14** (2006.01)

**G06F 11/20** (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **30.12.2009** **E 09801897 (1)**

97 Fecha y número de publicación de la concesión europea: **06.03.2013** **EP 2382545**

54 Título: **Mecanismo de configuración de componentes para reinicio**

30 Prioridad:

**31.12.2008 US 347300**

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

**28.06.2013**

73 Titular/es:

**SCHNEIDER ELECTRIC USA, INC. (100.0%)**  
**1415 S. Roselle Road**  
**Palatine, Illinois 60067 , US**

72 Inventor/es:

**SUN, ENXI y**  
**DOGGETT, DAVID**

74 Agente/Representante:

**DE ELZABURU MÁRQUEZ, Alberto**

**ES 2 409 882 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**

Mecanismo de configuración de componentes para reinicio.

**Antecedentes**

- 5 Los sistemas de automatización en fábrica están siendo integrados cada vez más con las redes de comunicación. Los sistemas de control están siendo implementados en redes para monitorización y control remotos de dispositivos, procesos, etc. Los fallos de sistema que implican el controlador de modo primario que pueden apagar el sistema de control se evitan teniendo un controlador de respaldo fácilmente disponible en modo caliente/de espera activa para sustituir el controlador de modo primario que falla. Incluso con redundancia del sistema, es importante que cualquier fallo del sistema sea reparado de manera expedita a fin de reducir la probabilidad de una interrupción del sistema.
- 10 Controladores tales como los controladores de lógica programable (PLC) se han implementado en configuraciones de sistema dúplex o de respaldo donde no se puede tolerar un tiempo de inactividad de un sistema. Tal sistema de control entrega una alta fiabilidad a través de la redundancia. Generalmente, la configuración dúplex incorpora un par de PLC montados en una configuración de espera activa o caliente, donde un PLC está operando en un modo primario y el otro PLC está funcionando en un modo secundario o de espera/respaldo. El controlador primario ejecuta una aplicación explorando un programa de usuario para controlar y monitorizar una red de entrada/salida (I/O) remota. El otro controlador (secundario) actúa como el controlador en espera activa. El controlador en espera no ejecuta la aplicación y no opera los dispositivos de I/O remotos. El controlador en espera se actualiza por el controlador primario con cada exploración. El controlador en espera entonces está listo para asumir el control del sistema de control dentro de una exploración si el controlador primario falla al operar o se retira de la operación.
- 15
- 20 Los controladores primario y secundario son intercambiables y se pueden cambiar o conmutar cuando se desee. Cualquiera de los dos controladores se puede situar en el estado primario. La configuración en espera activa requiere que el controlador no primario sea situado en el modo en espera para asegurar la redundancia del sistema. Los controladores comunican continuamente uno con otro para asegurar la operatividad del sistema de control. La comunicación entre los controladores se usa para determinar si se debería iniciar un cambio de los controladores debido a un fallo del sistema o por elección de un operador.
- 25 Incluso con un controlador primario/secundario donde el controlador no operativo se puede quitar del servicio, es importante que un controlador no operativo sea reparado a fin de proporcionar una fiabilidad que a menudo se espera por el operador. Un controlador típicamente incluye una serie de paquetes de circuitos, por ejemplo, un módulo de unidad central de proceso (CPU) y el módulo de comunicación que hace de interfaz con dispositivos externos. A fin de reparar un controlador no operativo, un módulo defectuoso detectado se sustituye típicamente con un módulo operativo. No obstante, un controlador funciona en un entorno de control específico; por consiguiente, un paquete de circuitos insertado está configurado típicamente para el entorno de control.
- 30 La US7411417 describe sistemas y métodos para proporcionar técnicas para la carga de celdas de memoria de configuración en circuitos integrados, tales como dispositivos de lógica programable.
- 35 La US2004/268106 describe métodos para detectar la validez de datos de configuración y recuperar datos de configuración en un sistema informático.
- La US2007/168704 describe un método de configuración automática de conmutación por fallo basado en una interfaz gráfica de usuario desde un sistema de base de datos primario a un sistema de base de datos en espera que está acoplado al sistema de base de datos primario por una red.
- 40 En un primer aspecto de la invención, como se define en la reivindicación independiente 1 adjunta, hay proporcionado un módulo de circuito (113, 115) de un controlador de lógica programable PLC (101) que comprende: una memoria interna (304) configurada para almacenar la información de configuración almacenada; una interfaz de comunicación configurada para obtener información de configuración externa a partir de una fuente de configuración externa, dicha interfaz de comunicación que comprende: una interfaz de red (404) configurada para comunicar a través de una red de comunicación; y una interfaz de PLC (405) configurada para comunicar con un módulo de controlador del PLC. El módulo de circuito además comprende un procesador (402) configurado para: soportar un primer componente que está asociado con una configuración basada en web; y determinar un escenario de configuración y seleccionar una fuente de configuración basada en el escenario de configuración, en donde la fuente de configuración es o bien la memoria interna o bien la fuente de configuración externa, la determinación que comprende: comparar una primera comprobación de redundancia almacenada en el módulo de controlador, en donde la primera comprobación de redundancia y la segunda comprobación de redundancia están asociadas con el primer componente; y comparar un primer tiempo de configuración de módulo de circuito almacenado en el módulo de circuito con un segundo tiempo de configuración de módulo de circuito almacenado en el módulo de controlador.
- 45
- 50
- 55 En un segundo aspecto de la invención, como se define en la reivindicación independiente 9 adjunta, hay proporcionado un método para configurar un módulo de circuito (113, 115), el método que comprende: determinar un

escenario de configuración a partir de un conjunto de escenarios de configuración posibles, el conjunto que incluye un primer escenario de configuración y un segundo escenario de configuración, la determinación comprende: comparar una primera configuración de redundancia almacenada en el módulo de circuito con una segunda comprobación de redundancia almacenada en un módulo de controlador asociado, en donde la primera comprobación de redundancia y la segunda comprobación de redundancia están asociadas con un primer componente del módulo de circuito; y comparar un primer tiempo de configuración de módulo de circuito con un segundo tiempo de configuración de módulo de circuito almacenado en el módulo de controlador asociado; cuando el escenario de configuración determinado es el primer escenario de configuración, seleccionar una fuente de configuración interna; cuando el escenario de configuración determinado es el segundo escenario de configuración, seleccionar una fuente de configuración basada en web externa; y configurar el primer componente del módulo de circuito usando datos de configuración desde la fuente de configuración seleccionada.

Otros aspectos opcionales de la invención son como se describe en las reivindicaciones dependientes adjuntas.

### Breve descripción de los dibujos

Una comprensión más completa de la presente invención y las ventajas de la misma se pueden adquirir con referencia a la siguiente descripción en consideración de los dibujos anexos, en los cuales números de referencia iguales indican rasgos iguales y en donde:

La Figura 1 muestra un sistema de control según una realización de la invención.

La Figura 2 muestra un diagrama de bloques de un sistema de control de red según una realización de la invención.

La Figura 3 muestra un módulo de comunicación y un módulo de CPU de un controlador de lógica programable según una realización de la invención.

La Figura 4 muestra un diagrama de bloques de un módulo de comunicación según una realización de la invención.

La Figura 5 muestra una tabla con diferentes escenarios de configuración según una realización de la invención.

La Figura 6 muestra un diagrama de flujo para seleccionar una fuente de configuración según una realización de la invención.

La Figura 7 muestra un diagrama de flujo para configurar un componente configurado por web según una realización de la invención.

La Figura 8 muestra un diagrama de flujo para configurar un componente configurado por CPU según una realización de la invención.

### Descripción detallada

En la siguiente descripción de las diversas realizaciones, se hace referencia a los dibujos anexos los cuales forman parte de ésta, y en los que se muestra a modo de ilustración diversas realizaciones en las que la invención se puede poner en práctica. Se tiene que entender que otras realizaciones se pueden utilizar y se pueden hacer modificaciones estructurales y funcionales sin apartarse del alcance de la presente invención.

Algunos componentes del programa fijo de máquina en el módulo de comunicación están configurados por una herramienta de ingeniería, *por ejemplo*, Unity Pro, a través de un controlador de lógica programable (PLC), que almacena los datos de configuración desde la herramienta de ingeniería. Otros componentes del programa fijo de máquina en el módulo de comunicación se pueden configurar por "WEB" a través de Ethernet, por ejemplo, una página Web y un cliente FTP. Todos los datos de configuración, ya sean desde el PLC o bien desde la "WEB", están almacenados en la memoria instantánea local del módulo de comunicación.

En el caso del módulo de comunicación que se reinicia, se espera típicamente que lea los datos de configuración desde la memoria instantánea local en lugar de leer desde el PLC y/o desde la "WEB" para mejorar el rendimiento de arranque del sistema.

En el caso del módulo de comunicación que se sustituye por un módulo usado con los datos de configuración correctos para otro sistema de control, pero no para el sistema de control actual, el módulo de comunicación no debería configurar sus componentes con los datos de configuración en su memoria instantánea local.

Se necesita un mecanismo para asegurar que el módulo de comunicación se configura correctamente en ambos casos anteriores.

La Figura 1 muestra un sistema de control según una realización de la invención. Dos controladores de lógica programable (PLC) configurados idénticamente 101 y 103 comunican uno con otro a través del procesador de opción en espera situado en cada controlador de lógica programable sobre el enlace de fibra óptica 157. El enlace de fibra óptica 157 puede soportar uno de diferentes protocolos, incluyendo Ethernet y Control de Enlace de Datos

de Alto Nivel (HDLC) en la capa 2 del modelo OSI. Uno de los controladores de lógica programable actúa típicamente como el controlador primario, mientras que el otro controlador de lógica programable está en modo de espera (que corresponde al controlador secundario). El controlador en espera es capaz de tomar la propiedad de los enlaces de I/O remotos 159 y 161, que comunican con los dispositivos de I/O remotos 105 y 107.

5 Los PLC 101 y 103 típicamente incluyen una serie de módulos de circuito (paquetes de circuitos) (por ejemplo, los módulos de circuito 113 y 115) que están insertados en un alojamiento en la placa posterior. Si el PLC 101 y 103 llegan a estar no operativos, un usuario puede intercambiar el módulo de circuito defectuoso determinado con un módulo de circuito bueno que puede ser un nuevo o un viejo repuesto. Si el módulo de circuito es un repuesto viejo, típicamente ha estado configurado previamente para un PLC diferente que puede estar asociado con un entorno de control diferente. Como ejemplo, un usuario puede determinar que un módulo de comunicación está mal y lo sustituye con otra comunicación de un PLC que está fuera de línea.

Un usuario comunica y controla los controladores de lógica programable 101 y 103 desde los ordenadores 109 y 111 sobre los enlaces 151, 153, y 155. Con una realización, los enlaces 151, 153, y 155 operan según un Protocolo de Control de Transmisión/Protocolo de Internet (TCP/IP) sobre una red Ethernet.

15 La Figura 2 muestra un diagrama de bloques de un sistema de control de red según una realización de la invención. Por ejemplo, un sistema de control de red consta de un controlador de lógica de programación (PLC o módulo de CPU 202), un módulo de comunicación 201, un navegador web 203, dispositivos de I/O 204-206, y herramientas de ingeniería (no se muestran) para configuración y diagnosis. El módulo de comunicación 201 puede contener numerosos componentes de programa fijo de máquina (aplicaciones, *por ejemplo*, los componentes 207-210), donde algunos componentes están configurados desde una fuente de configuración basada en web ("WEB") (*por ejemplo*, los componentes 207 y 208) y algunos componentes se configuran desde el módulo de CPU 202 (*por ejemplo*, los componentes 209 y 210) típicamente en conjunto con una herramienta de configuración. Cuando el módulo de comunicación 201 se considera roto, el usuario puede sustituirlo con otro módulo de comunicación que puede contener datos de configuración desde la WEB pero para un sistema de control diferente. Esto puede provocar un problema de configuración del sistema con datos de configuración para otro sistema. Ejemplos de componentes que se configuran basados en web incluyen un Escáner de IO, componentes de Datos Globales, y un componente de conmutación. Ejemplos de componentes que se configuran basados en CPU incluyen un componente de servidor DHCP y de Agente SNMP. Con algunas realizaciones, cuando el módulo de comunicación 201 determina que los datos de configuración se deben obtener desde WEB, el módulo de comunicación 201 espera los datos de configuración de manera pasiva y el usuario introduce los datos de configuración. Los datos se envían al módulo de comunicación 201 a través del navegador web o el cliente FTP. Con otras realizaciones, el módulo 201 emite una petición a un dispositivo servidor remoto para descargar la configuración, en cuyo caso, la fuente de configuración basada en web es un repositorio de datos actuales para componentes basados en web y se sitúa típicamente en un servidor seguro en Internet o una intranet de empresa.

35 La Figura 3 muestra un módulo de comunicación 201 y el módulo de CPU 202 de un controlador de lógica programable según una realización de la invención. El módulo de CPU 202 puede soportar diferentes tipos de módulos de comunicación, incluyendo módulos de Ethernet de Opciones de Red (NOE) que hacen de interfaz del PLC con dispositivos de I/O remotos 105 y 107 sobre los cables 159 y 161. El módulo de comunicación 201 típicamente soporta una pluralidad de diferentes componentes (aplicaciones).

40 Si el módulo de comunicación 201 necesita reconfigurar (por ejemplo, durante un reinicio), el módulo 201 usa los datos de configuración desde la memoria instantánea 304 si el módulo 201 determina que los datos son correctos para el entorno de operación. (La memoria instantánea es una memoria de ordenador no volátil que se puede reprogramar y borrar eléctricamente.)

45 Hay varias consideraciones cuando se reconfigura (reinicia) el módulo de comunicación 201. Por ejemplo, se pueden configurar o actualizar los componentes que incluyen la configuración del escáner de I/O, los datos globales, el Protocolo de Gestión de Red Simple (SNMP), y el Protocolo de Configuración de Ordenador Principal Dinámica (DHCP). Con los sistemas tradicionales, si un componente está configurado con datos de configuración en la memoria instantánea, es posible para el componente que esté configurado con datos no válidos, provocando el fallo del sistema. Esto puede ocurrir cuando el módulo se reinicia a partir de un encendido después de un apagado durante la escritura/actualización de los datos de configuración desde un módulo de CPU o desde una fuente de configuración basada en web (no se muestra) en la memoria instantánea. También, con sistemas tradicionales, si un componente está configurado con los datos de configuración en la memoria instantánea, también es posible para el componente estar configurado con los datos de configuración que están asociados con un módulo diferente o sistema diferente, provocando el fallo del sistema.

55 La operación errónea puede ocurrir cuando un módulo de comunicación falla y se sustituye con otro módulo que tiene los datos de configuración en la memoria instantánea para un módulo o sistema diferente. Por ejemplo, un módulo de NOE se puede configurar con una tabla de exploración del dispositivo para explorar diferentes dispositivos de I/O. Si un módulo de NOE se toma de una ranura diferente, el módulo se puede configurar para un bastidor diferente y de esta manera no tendrá la tabla de exploración de I/O correcta almacenada en la memoria

- instantánea del módulo. Cuando se arranca, el módulo de NOE por consiguiente usará la tabla de exploración de IO existente para un diferente bastidor para configurar e iniciar el escáner de IO, provocando el fallo del sistema 100. A fin de evitar esta situación en sistemas tradicionales, el usuario puede reconfigurar el escáner de I/O a través de la "WEB" todas las veces que el módulo de NOE se reinicia o el módulo de CPU se reinicia. No obstante, esta acción requiere tiempo y esfuerzo por el usuario.
- También, con los sistemas tradicionales, si solamente se sustituye el módulo de CPU, el módulo de comunicación no puede configurarse a sí mismo o algunos componentes con los datos de configuración en la memoria instantánea y en su lugar indica que está esperando los datos de configuración desde la fuente web. Esta situación ocurre debido a que el programa fijo de máquina de arranque del módulo típicamente no pueden hacer una distinción entre las combinaciones de un nuevo módulo de CPU + un módulo de comunicación viejo, un viejo módulo de CPU + un nuevo módulo de comunicación, y un nuevo módulo de CPU + un nuevo módulo de comunicación con el sistema tradicional.
- Las realizaciones de la invención soportan un mecanismo de configuración de componentes durante el arranque del módulo de comunicación 201 en la condición de mezcla de las configuraciones de componentes desde la WEB y las herramientas de PC y usando respaldo de datos de configuración de la memoria instantánea. El mecanismo de configuración típicamente evita usar datos de configuración no válidos desde la memoria instantánea 304 debido a la desconexión durante la actualización de la memoria instantánea y la sustitución del módulo de comunicación. El mecanismo va a ser usado para diferentes productos de módulo de comunicación, por ejemplo, NOE (Ethernet de Opciones de Red), ETY, NOC (CIP de Opciones de Red), y ETC. Además, en adición a los módulos de comunicación, las realizaciones de la invención soportan otros tipos de módulos de circuitos en el sistema de control 100, por ejemplo, cualquier módulo que contiene datos de configuración y/o parámetros de dispositivo para otros dispositivos.
- El sistema 100 típicamente incluye el módulo (controlador) de CPU 202, el módulo de comunicación 201, y otros numerosos dispositivos. La configuración del módulo de CPU 202 puede ser a través de una herramienta de configuración 303 (Unity™ Pro que se ejecuta en un PC, *por ejemplo*, los ordenadores 109 o 111). Una vez que finaliza la configuración del módulo de CPU 202, el usuario normalmente desconecta la herramienta de configuración 303 (Unity Pro) con el módulo de CPU 202. Como se soportan por el módulo de comunicación 201, algunos componentes se configuran por el módulo de CPU 202, mientras que otros componentes se configuran por una fuente de configuración basada en web.
- Con un aspecto de la invención, un mecanismo de configuración de componentes típicamente evita la configuración de un componente que está configurado con datos de configuración no válidos cuando el módulo de comunicación 201 se reinicia (se enciende) justo después de que ocurre un apagado durante la escritura de los datos de configuración en la memoria instantánea. La capacidad puede ser seleccionable. Si un usuario desea un rendimiento del sistema mayor, el usuario puede elegir no usar este rasgo. El mecanismo de configuración de componentes típicamente evita la configuración de cualquier componente que está configurado con los datos de configuración existentes inadecuados que están asociados con un módulo diferente o un sistema diferente cuando se arranca un nuevo módulo. El programa fijo de máquina de arranque puede determinar si el reinicio es a partir de un reinicio del módulo de comunicación existente o a partir de un primer encendido de un nuevo módulo de comunicación sustituido.
- A fin de reducir el tiempo de arranque y realizar una automatización real, el módulo de comunicación 201 usa las copias locales de los datos de configuración 316, 317, y 318c almacenadas en memoria instantánea 304 para configurar sus componentes. Por lo tanto, las configuraciones de los componentes del módulo de comunicación durante el reinicio consideran no solamente los escenarios (casos) de arranque (como se muestra en la Figura 5) sino también la validez de los datos de configuración en la memoria instantánea local 304. Con algunas realizaciones, el escenario de arranque y la validez de las copias locales de los datos de configuración se determinan, donde la información de configuración está solamente en el módulo de CPU 202 y el módulo de comunicación 201. Cualquiera de los dos o ambos módulos se pueden sustituir, lo cual provoca la pérdida de la información de configuración de respaldo. Las realizaciones de la invención resuelven las ambigüedades de configuración bajo diferentes condiciones de arranque (escenarios de configuración).
- Con las realizaciones de la invención, la herramienta de configuración Unity Pro 303 proporciona casillas de comprobación etiquetadas como "CPU sustituida y configuración por primera vez" 307 y "NOE sustituido y configuración por primera vez" 308. Cuando el usuario sustituye un módulo de CPU fallido con otro módulo de CPU y configura el módulo de CPU por primera vez, el usuario comprueba la casilla de comprobación "CPU sustituida y configuración por primera vez" 307. Cuando el usuario sustituye un módulo de NOE fallido con otro módulo de NOE y configura el módulo de NOE por primera vez, el usuario comprueba la casilla de comprobación "NOE sustituido y configuración por primera vez" 308. Para cada módulo de sustitución, la casilla de comprobación correspondiente se comprueba solamente cuando se hace la configuración por primera vez. Cada vez que el usuario construye los datos de configuración, la herramienta de configuración Unity Pro 303 emite una ventana emergente para recordar al usuario fijar las casillas de comprobación 307 y 308 correctamente.

- 5 La herramienta de configuración Unity Pro 303 descarga los datos de configuración al módulo de CPU 202. Por consiguiente, el módulo de CPU 202 obtiene CPU\_Config\_CRC 309a, 309b, CPU\_Config\_Time 310a, 310b, y los valores de casilla de comprobación CPU\_new 312a, 312b y NOE\_new 311a, 311b. Los datos de configuración se almacenan en la memoria instantánea de la CPU 305 y en la RAM del sistema 306 para acceso posterior por el módulo de CPU 202. Los valores de configuración deberían ser coherentes en la memoria instantánea 305 y en la RAM 306. Por consiguiente, cada vez que el módulo de CPU 202 se reinicia, estos valores se actualizan en la RAM 306 a partir de la memoria instantánea 305; cada vez que el módulo de CPU 202 recibe nuevos datos de configuración desde la herramienta de configuración Unity Pro 303, estos valores de configuración se escriben en la memoria instantánea 305.
- 10 El código de redundancia cíclica (CRC) para los datos de configuración 316 y 317 de cada componente configurado por web se calcula siempre que los datos de configuración 316 y 317 estén configurados/actualizados desde la "WEB". (La realización ejemplar mostrada en las Figuras 2 y 3 representa un componente A y componente B.) Los valores de CRC correspondientes y el tiempo de configuración del módulo se guardan ambos en la memoria instantánea 304 (313c, 314c, y 315c del módulo de comunicación 201), la memoria instantánea 305, y la tabla de configuración del sistema 306. El tiempo de configuración del módulo (NOE\_Config\_Time 315a, 315b, 315c) se actualiza siempre que ocurre una nueva configuración sin importar si viene desde el módulo de CPU 202 o desde la WEB y siempre debería ser mayor que el CPU\_Config\_Time (310a, 310b, 310c) de la herramienta de configuración Unity Pro 303.
- 15 Tras la configuración, el módulo de comunicación 201 lee CPU\_Config\_CRC 309b, CPU\_Config\_Time 310b, CPU\_new 312b, y NOE\_new 311b a partir de la RAM del sistema 306 del módulo de CPU 202. El módulo 201 los guarda en la memoria instantánea local 304 y lee cada CRC de componente (313c, 314c) al módulo de CPU 202 (que corresponde al CRC del Componente A 313b y el CRC del Componente B 314b). El módulo de CPU 202 fija CPU\_new 312b, 312a y NOE\_new 311a, 311b a "0" después de que el módulo de comunicación (NOE) 201 los lee a fin de indicar que tanto el módulo de CPU 202 como el módulo de NOE 201 no son nuevos. Con algunas realizaciones, si el módulo de CPU no puede fijar CPU\_new 312b y NOE\_new 311b, entonces el módulo de comunicación 201 fija estos valores.
- 20 El módulo de CPU 202 y el módulo de NOE 201 están conectados a y comunican uno con otro a través del canal principal de la placa posterior 319. La herramienta de configuración 303 se ejecuta en un PC y conecta al PLC a través del puerto Modbus plus o el puerto USB en la placa del PLC directamente o a través del puerto Ethernet en la placa del módulo de NOE 201 indirectamente. En los dos primeros casos, la herramienta de configuración 303 configura directamente el PLP sin importar si el módulo de NOE 201 se ejecuta o no. En el segundo caso, la herramienta de configuración 303 configura el PLC a través de la placa del módulo de NOE 201, la cual debería tener una configuración válida y estar ejecutándose. El último caso es el usado mayoritariamente para cambiar la configuración cuando el sistema 100 ya está ejecutándose. Cuando se configura un PLC (uno nuevo o usado en otro sistema) en el sistema 100 por primera vez, el usuario debería reconfigurar el PLC primero y el sistema 100 debería reiniciarse.
- 25 La configuración del módulo de NOE 201 depende del módulo de CPU 202, que debería estar ejecutándose y tener datos de configuración válidos para el módulo de NOE 201. El módulo de NOE 201 lee la información de configuración desde el módulo de CPU 202 de manera que el módulo de NOE 201 conozca qué componentes están configurados a partir del módulo de CPU 202 y qué componentes están configurados a partir de la WEB (no se muestra).
- 30 La herramienta de configuración Unity Pro 303 soporta las casillas de comprobación 307 y 308 etiquetadas "CPU sustituida y configuración por primera vez" y "NOE sustituido y configuración por primera vez," respectivamente. Cuando el usuario sustituye un módulo de CPU fallido con otro módulo de CPU y configura el módulo de CPU por primera vez, el usuario comprueba la casilla de comprobación "CPU sustituida y configuración por primera vez" 307. Cuando el usuario sustituye un módulo de NOE fallido con otro módulo de NOE y configura el módulo de NOE por primera vez, el usuario comprueba la casilla de comprobación "NOE sustituido y configuración por primera vez" 308. Para cada módulo de sustitución, la casilla de comprobación correspondiente solamente se necesita que sea comprobada una vez cuando se hace la configuración por primera vez. Cada vez cuando el usuario construye los datos de configuración, la herramienta de configuración Unity Pro 303 emite una ventana emergente para recordar al usuario fijar las casillas de comprobación 307 y 308 correctamente.
- 35 CPU\_Config\_CRC (309a, 309b) es el CRC de los datos de configuración de la CPU y el CPU\_Config\_Time (310a, 310b) es el tiempo de construcción de los datos de configuración de la CPU. Cuando la herramienta de configuración Unity Pro 303 descarga los datos de configuración al módulo de CPU 202, también carga estas dos variables en el módulo de CPU 202 durante la configuración. También, la herramienta de configuración Unity Pro 303 carga los valores de las casillas de comprobación CPU\_new 311a, 311b y NOE\_new 312a, 312b al módulo de CPU. Estos cuatro valores se almacenan en la memoria instantánea de la CPU 305 y en el área de configuración del módulo en la RAM del sistema 306 para que acceda el módulo de CPU 202.
- 40 Estos valores deberían ser coherentes en la memoria instantánea 305 y la RAM del sistema 306 cada vez que la

CPU se reinicia o con una nueva configuración a partir de la herramienta de configuración Unity Pro 303. Estos cuatro valores de configuración en la RAM del sistema 306 se actualizan típicamente desde la memoria instantánea 305.

5 El módulo de comunicación 201 almacena los datos de configuración A 316 (que corresponden al componente A configurado por web), los datos de configuración B 317 (que corresponden al componente B configurado por web), y Copy\_CPU\_Config\_Data 318c (que corresponde a unos componentes configurados por CPU 209 y 210 como se muestra en la Figura 2). Copy\_CPU\_Config\_Data 318c es una copia local de los datos de configuración desde la CPU (que corresponde a los datos de configuración 318a, 318b). Estos datos de configuración se almacenan originalmente en el módulo de CPU 202 y se leen por el módulo de NOE 201, el cual guarda una copia de los datos de configuración. Con un reinicio posterior, el módulo de NOE 201 usa la copia local para configurar los componentes que están configurados por CPU si la copia local es válida o no está dañada.

15 La Figura 4 muestra un diagrama de bloques del módulo de comunicación 201 según una realización de la invención. El procesador 402 puede ejecutar instrucciones ejecutables por ordenador desde un medio legible por ordenador, *por ejemplo*, la memoria 406. Los medios de almacenamiento informático pueden incluir medios volátiles y no volátiles, extraíbles y no extraíbles implementados en cualquier método o tecnología para almacenamiento de información tal como instrucciones legibles por ordenador, estructuras de datos, módulos de programa u otros datos. Los medios de almacenamiento informáticos incluyen, pero no se limitan a, memoria de acceso aleatorio (RAM), memoria solamente de lectura (ROM), memoria solamente de lectura borrable electrónicamente (EEPROM), memoria instantánea u otra tecnología de memoria, CD-ROM, discos versátiles digitales (DVD) u otro almacenamiento de disco óptico, cartuchos magnéticos, cinta magnética, almacenamiento de disco magnético u otros dispositivos de almacenamiento magnético, o cualquier otro medio que se pueda usar para almacenar la información deseada y que se pueda acceder por el procesador 402.

20 El módulo de comunicación 201 típicamente realiza una pluralidad de funciones que corresponden a componentes. Por ejemplo, el módulo de comunicación 201 explora y controla los dispositivos remotos 105 y 107 a través de la interfaz de I/O 407 y por consiguiente mantiene una tabla en la memoria instantánea 304 (*por ejemplo*, datos de configuración B 317 como se muestra en la Figura 3.)

25 El procesador 402 obtiene los datos de configuración desde la memoria instantánea 304 (también mostrada en la Figura 3) para los componentes soportados cuando el procesador 402 determina que los datos de configuración en la memoria instantánea 304 son válidos. No obstante, el procesador 402 obtiene los datos de configuración a partir de la herramienta basada en web 401 para los componentes configurables por web a través de la interfaz de red 404 y obtiene los datos de configuración desde el módulo de CPU 202 para los componentes configurables por CPU a través de la interfaz de PLC 405.

30 La Figura 5 muestra la tabla 500 con diferentes escenarios de configuración según una realización de la invención. Con el escenario de arranque 501 (caso 1), el sistema de control 100 (CPU, módulo de comunicación, y dispositivos) está ejecutándose y no vienen nuevos datos de configuración desde la herramienta de configuración Unity Pro 303. El mismo módulo de comunicación se reinicia con las condiciones de recuperación de reinicio automático a partir de los errores del programa informático o los componentes físicos, el reinicio manual realizado por los usuarios, el reinicio automático forzado por comandos remotos, y el reinicio manual realizado por desarrolladores y probadores para el desarrollo, prueba, depuración, verificación, y validación.

35 Con el escenario de arranque 502 (caso 2), el sistema de control 100 está ejecutándose y el usuario cambia la configuración del sistema a través de la herramienta de configuración Unity Pro 303. No hay reinicio desde la CPU, módulo de comunicación, y dispositivos.

Con el escenario de arranque 503 (caso 3), el sistema de control 100 se detiene y entonces el usuario enciende el sistema completo. No hay nueva configuración.

40 Con el escenario de arranque 504 (caso 4), el sistema de control 100 está ejecutándose y no vienen nuevos datos de configuración desde la herramienta de configuración Unity Pro 303. Posteriormente, el módulo de comunicación falla, y el usuario lo sustituye con otro módulo de comunicación, que se reinicia automáticamente. El módulo de comunicación de sustitución puede ser completamente nuevo o uno usado. Si es un módulo usado, puede contener datos de configuración configurados por web de algunos componentes configurados por web para otro sistema de control distinto de éste. Con un aspecto de la invención, los datos de configuración previos (no para este sistema) no se usan para configurar componentes asociados del módulo de comunicación.

45 Con el escenario de arranque 505 (caso 5), el sistema de control 100 se detiene. El usuario sustituye el módulo de comunicación con otro y entonces reinicia el sistema completo. En esta situación, el módulo de comunicación puede contener datos de configuración configurados por web de componentes de configuración web asociados para otro sistema de control distinto de éste. El módulo de comunicación no se debería configurar con datos de configuración previos pero en su lugar debería esperar por los datos de configuración desde la WEB.

50 Con el escenario de arranque 506 (caso 6), el sistema de control 100 se detiene. El usuario sustituye el módulo de

CPU con uno nuevo y entonces reinicia y configura el sistema 100. En esta situación, el mismo módulo de comunicación aún contiene datos de configuración por web válidos en su memoria instantánea y se debería configurar con los datos de configuración válidos pero no espera por la configuración desde la WEB.

5 Con el escenario de arranque 507 (caso 7), el sistema de control 100 se detiene. El usuario sustituye tanto el módulo de CPU como el módulo de comunicación con unos nuevos y entonces reinicia y configura el sistema completo. En esta situación, el módulo de comunicación puede contener datos de configuración configurados por web de los componentes configurados por web asociados para otro sistema de control distinto de éste. El módulo de comunicación no se debería configurar previo a los datos de configuración pero en su lugar espera por la configuración desde la WEB.

10 La Figura 6 muestra el diagrama de flujo 600 para seleccionar una fuente de configuración según una realización de la invención. Con las realizaciones de la invención, el proceso 600 se configura por el procesador 402 como se muestra en la Figura 4. El diagrama de flujo 600 corresponde al siguiente pseudo código cuando se configuran componentes soportados por el módulo de configuración 201. El diagrama de flujo 600 procesa los siguientes datos de configuración:

15 **CPU\_Config\_CRC:** (que corresponde a 309a, 309b, 309c) el CRC de los datos de configuración descargados desde la herramienta de configuración Unity Pro 303.

**CPU\_config\_time:** (que corresponde a 310a, 310b, 310c) el tiempo de la descarga unitaria de los datos de configuración a la CPU desde la herramienta de configuración Unity Pro 303.

20 **CPU\_new:** (que corresponde a 312a, 312b) el valor que corresponde a la casilla de comprobación "CPU sustituida y configuración por primera vez" 307.

**NOE\_new:** (que corresponde a 311a, 311b) el valor que corresponde a la casilla de comprobación "NOE sustituida y configuración por primera vez" 308.

25 **NOE\_Config\_Time:** (que corresponde a 315a, 315b, 315c) el tiempo del cambio del NOE en la configuración.

**CRC:** (que corresponde a 313a, 313b, 313c y 314a, 314b, 314c) CRC para cada componente configurado por web.

**Copy\_CPU\_Config\_Data:** (que corresponde a 318c) copia local de los datos de configuración desde el módulo de CPU 202.

30 **Configuration data A:** (que corresponde a 316) copia local de los datos de configuración para el componente A desde la WEB.

En el paso 601, el proceso 600 comienza el proceso de configuración para configurar los componentes del módulo de comunicación 202. El módulo 202 se indica como NOE, aunque las realizaciones de la invención pueden soportar otros tipos de módulos de comunicación. El módulo de comunicación 202 lee el CPU\_Config\_CRC 309, CPU\_config\_time 310, CPU\_new 312, NOE\_new 311, NOE\_Config\_Time 315, y los CRC para cada componente configurado por web (por ejemplo, 313 y 314) típicamente desde la RAM del sistema 306. También el paso 601 fija tanto CPU\_new 312 como NOE\_new 311 en el módulo de CPU 202 a "0" si el módulo de CPU 202 no hace esto después de que el módulo de comunicación 202 los lea. Esto facilita al módulo de comunicación 202 obtener la información de configuración correcta cuando hay una nueva configuración pero cuando el módulo de comunicación 201 y/o el módulo de CPU 202 se reinician.

El siguiente listado de pseudo códigos indica los pasos correspondientes en el diagrama de flujo 600 y los escenarios de configuración (casos) correspondientes como se muestra en la Figura 5.

**Paso 603:**

Si( "NOE\_new ) /\* Debe haber una nueva configuración desde la herramienta de configuración Unity Pro \*/

45 {

Para cada uno de todos los componentes configurados por web,

**Paso 605:**

Si (cada CRC de componente en el módulo de CPU 202 == cada CRC de componente en el módulo de NOE 201) &&

50 Si ( NOE\_Config\_Time en el módulo de CPU == NOE\_Config\_Time en el módulo de NOE)



```

{
    /* Este es el caso 2 con la casilla de comprobación "módulo de NOE sustituido y configuración por
    primera vez" en la herramienta de configuración Unity Pro es erróneamente comprobado por el
    usuario. */
5   Paso 607 (Caso 2):
        Configure_WEB_Component_from_Flash ();    /*definición de función más adelante */
    }
    además
    {
10      /* Ahora es imposible distinguir el caso 6 y el caso 7 aquí si la comprobación del usuario de la casilla
    de comprobación "NOE sustituido y configuración por primera vez" en la herramienta de configuración
    Unity Pro es erróneamente comprobada por el usuario. Es exactamente el mismo papel diseñado de la
    casilla de comprobación "NOE sustituido y configuración por primera vez" en Unity Pro para distinguir
    el caso 6 y el caso 7. */
15   Paso 609 (Casos 6 y 7):
        No configurar los componentes con los datos en la memoria instantánea sino desde la WEB;
    }
    /* no importa qué otras condiciones son, el módulo debe leer los datos de configuración para esos
    componentes (a ser configurados desde la CPU) en situación de nueva configuración. */
20   Paso 627:
        Leer los datos de configuración desde la CPU;
        Actualizar Copy_CPU_Config_Data en la memoria instantánea local;
        Configurar esos componentes (a ser configurados desde la CPU);
    }
25   Paso 611:
    además si ("CPU_new") /* NOE_new = 0 y nueva configuración desde Unity */
    {
        Para cada uno de todos los componentes configurados por web,
30   Paso 613:
        Si ( cada CRC de componente en la CPU == cada CRC de componente en el NOE) &&
        si ( NOE_Config_Time en la CPU == NOE_Config_Time en el NOE)
        {
            /* Este es el caso 2 con la casilla de comprobación "CPU sustituida y configuración por primera vez"
            en Unity Pro es erróneamente comprobado por el usuario. */
35   Paso 607:
                Configure_WEB_Component_from_Flash ();
            }
Paso 615:
            además si ( CPU_Config_Time en la CPU <= CPU_Config_Time en el NOE))
40   {

```

*/\* El NOE debe ser nuevo y el usuario olvida comprobar la casilla de comprobación “NOE sustituido y configuración por primera vez” en Unity Pro. Caso 7 \*/*

**Paso 609 (Casos 6 y 7):**

No configurar componentes con los datos en la memoria instantánea sino desde la WEB;

5 */\* Esto evita que el módulo de NOE 201 se configure con los datos de configuración inadecuados correspondientes a un módulo o sistema diferente.\*/*

}

además

{

10 */\* Ahora es imposible distinguir el caso 6 y el caso 7 aquí sin el NOE\_new correcto, si el usuario olvida comprobar la casilla de comprobación “NOE sustituido y configuración por primera vez” en Unity Pro en el caso 7. Tendrá un problema si hay datos de configuración existentes para los componentes configurados por web. El código aquí está basado en el valor NOE\_new\*/*

**Paso 607 (Casos 6 y 7):**

15 Configure\_WEB\_Component\_from\_Flash();

}

*/\* no importa qué otras condiciones son, el módulo debe leer los datos de configuración para esos componentes (a ser configurados desde la CPU) en situación de nueva configuración. \*/*

**Paso 627:**

20 Leer los datos de configuración desde la CPU;

Actualizar Copy\_CPU\_Config\_Data en la memoria instantánea local;

Configurar esos componentes (a ser configurados desde la CPU);

}

además */\*CPU\_new = 0 && NOE\_new = 0 \*/*

25 {

*/\* caso 1, caso 2, caso 3, caso 4, y caso 5 \*/*

Para cada uno de todos los componentes configurados por web,

**Paso 617 (Casos 1, 2, 3, 4, y 5):**

Si ( cada CRC de componente en la CPU == cada CRC de componente en el NOE) &&

30 Si (“NOE\_Config\_Time” en la CPU == “NOE\_Config\_Time” en el NOE)

{

*/\* Este debe ser el caso 1, o caso 2, o caso 3\*/*

**Paso 621:**

si ( (“CPU\_Config\_CRC” en la CPU == “CPU\_Config\_CRC” en el NOE) &&

35 (“CPU\_Config\_Time” en la CPU == “CPU\_Config\_Time” en el NOE))

{

**Pasos 619 y 623 (Casos 1 y 3):**

*/\* Este debe ser el caso 1 o caso 3\*/*

Configure\_WEB\_Component\_from\_Flash ();

```

        Configure_CPU_Component_from_Flash (); /*definición de función más abajo */
    }
    además
    {
5         /* Es el caso 2 */
Pasos 619 y 627 (Caso 2):
        Configure_WEB_Component_from_Flash ();
        Leer datos de configuración desde la CPU;
        Actualizar Copy_CPU_Config_Data en la memoria instantánea local;
10        Configurar esos componentes (a ser configurados desde la CPU);
    }
}
además /* debe ser nueva CPU, o nuevo NOE, o ambos nuevos */
{
15    Si ( ! Exit_Dim_flag)
    {
        /* Exit_Dim_flag se fija cuando Unity Pro reconfigura la CPU. Debido a que el marcador no está fijado, no
        hay reinicio de la CPU y ninguna nueva configuración desde Unity. Por consiguiente, debe ser sustitución
        del NOE, caso 4 en la Figura 5*/
20 Paso 609 y 627 (Caso 4):
        Leer datos de configuración desde la CPU
        Actualizar Copy_CPU_Config_Data en la memoria instantánea local;
        Configurar esos componentes (a ser configurados desde la CPU);
        No configurar esos componentes (a ser configurados desde la WEB) con los
25        Datos en la memoria instantánea sino desde la WEB;
    }
    además
    {
        /* La CPU se reinicia o nueva configuración desde Unity */
30        /*caso 5 */
Pasos 609 y 627 (Caso 4):
        Leer datos de configuración desde la CPU;
        Actualizar Copy_CPU_Config_Data en la memoria instantánea local;
        Configurar esos componentes (a ser configurados desde la CPU);
35        No configurar esos componentes (a ser configurados desde la WEB) con los
        Datos en la memoria instantánea sino desde la WEB;
        Borrar Exit_Dim_flag

```

```

    }
  }
}

```

5 Cuando el proceso 600 ha completado la configuración, el módulo de comunicación 201 comienza a operar en el modo operacional en el paso 625.

10 Cuando el proceso 600 determina que se necesitan datos de configuración desde la WEB a fin de configurar un componente configurable por web (que corresponde al paso 609), el proceso 600 puede informar al usuario para descargar los datos desde una fuente de configuración basada en web. Con algunas realizaciones de la invención, un proceso de configuración se puede conectar automáticamente a la fuente de configuración basada en web y configurar el componente correspondiente.

15 La Figura 7 muestra el diagrama de flujo 607 (que corresponde al paso 607 como se muestra en la Figura 6) para configurar un componente configurado por web según una realización de la invención. Como se muestra con el paso 701, un usuario puede seleccionar no calcular la suma de comprobación de los datos de configuración de cada componente (*por ejemplo*, los datos de configuración 316 y 317 como se muestra en la Figura 3) y comparar el CRC calculado con el CRC almacenado (*por ejemplo*, 313c y 314c como se almacenan en la memoria instantánea 304). Los pasos correspondientes del diagrama de flujo 607 se muestran en el siguiente pseudo código.

```

/* Esta función configura los componentes configurados por web a partir de los datos de configuración en la memoria instantánea */

```

```

Configure_WEB_Component_from_Flash ()

```

```

20 {

```

**Paso 701:**

```

    Si (compute_checksum_selected)          /* rasgo seleccionable para consideración de tiempo de arranque */

```

```

    {

```

25 Para cada componente a ser configurado desde la WEB,

**Paso 703 y 705:**

```

        Calcular el CRC de sus Datos de configuración en la memoria instantánea local;

```

```

        Comparar el CRC con su Component_Config_CRC almacenado en memoria instantánea;

```

```

        Si ( igual )

```

```

30 {

```

**Paso 709:**

```

        Configurar los componentes configurables por web con sus datos de configuración correspondientes en la memoria instantánea local;

```

```

    }

```

35 además /\*no igual \*/

```

    {

```

**Paso 707:**

40 No configurar el componente configurable por web con los datos en la memoria instantánea sino desde la WEB;

```

        /* evitar usar datos no válidos debido al apagado durante la actualización de la memoria instantánea */

```

```

    }

```

```

}

```

además /\* compute\_checksum\_selected no seleccionado \*/

{

**Paso 709:**

5 Configurar los componentes configurables por web con sus datos de configuración correspondientes en la memoria instantánea local;

}

}

10 La Figura 8 muestra un diagrama de flujo 623 (que corresponde al paso 623) para configurar un componente configurado por CPU según una realización de la invención. Como se muestra con el paso 801, un usuario puede seleccionar no calcular la suma de comprobación de los datos de configuración de un componente configurable por CPU (*por ejemplo*, los datos de configuración 318 como se muestra en la Figura 3) y comparar el CRC calculado con el CRC almacenado (*por ejemplo*, 309c como está almacenado en la memoria instantánea 304). Los pasos correspondientes del diagrama de flujo 623 se muestran en el siguiente pseudo código.

15 /\* Esta función configura los componentes configurados por CPU a partir de los datos de configuración en la memoria instantánea \*/

Configure\_CPU\_Component\_from\_Flash ()

{

**Paso 801:**

20 Si (compute\_checksum\_selected) /\* rasgo seleccionable para consideración de tiempo de arranque \*/

{

**Paso 803 y 805:**

Calcular el CRC de los Copy\_CPU\_Config\_Data en la memoria instantánea local;

Comparar el CRC con CPU\_Config\_CRC;

25 Si ( igual )

{

**Paso 815:**

Configurar esos componentes (a ser configurados desde la CPU) con los datos de los

Copy\_CPU\_Config\_Data en la memoria instantánea local;

30 }

además /\*no igual \*/

{

**Paso 807:**

Leer los datos de configuración desde la CPU;

35 **Paso 809, 811, y 813:**

Actualizar los Copy\_CPU\_Config\_Data en la memoria instantánea local;

Configurar esos componentes (a ser configurados desde la CPU);

/\* evitar usar datos no válidos debido al apagado durante la actualización de la memoria instantánea \*/

}

40 además /\* compute\_checksum\_selected no seleccionado \*/

{

**Paso 815:**

Configurar esos componentes (a ser configurados desde la CPU) con los datos de los

Copy\_CPU\_Config\_Data en la memoria instantánea local;

5

}

}

10

Como se puede apreciar por un experto en la técnica, se puede utilizar un sistema informático con un medio legible por ordenador asociado que contiene instrucciones para controlar el sistema informático para implementar las realizaciones ejemplares que se describen en la presente memoria. El sistema informático puede incluir al menos un ordenador tal como un microprocesador, un procesador digital de señal, y circuitería electrónica periférica asociada.

15

Mientras que la invención se ha descrito con respecto a los ejemplos específicos que incluyen modos preferidos actualmente de llevar a cabo la invención, aquellos expertos en la técnica apreciarán que hay numerosas variaciones y permutaciones de los sistemas y técnicas descritos anteriormente que caen dentro del alcance de la invención como se expone en las reivindicaciones adjuntas.

**REIVINDICACIONES**

1. Un módulo de circuito (113, 115) de un controlador de lógica programable PLC (101) que comprende:
  - una memoria interna (304) configurada para almacenar información de configuración;
  - 5 una interfaz de comunicación configurada para obtener información de configuración externa desde una fuente de configuración externa, dicha interfaz de comunicación que comprende:
    - una interfaz de red (404) configurada para comunicar a través de una red de comunicación; y
    - una interfaz de PLC (405) configurada para comunicar con un módulo de controlador del PLC; y
  - un procesador (402) configurado para:
    - 10 soporta un primer componente que se asocia con una configuración basada en web; y
    - determinar un escenario de configuración y seleccionar una fuente de configuración basada en el escenario de configuración, en donde la fuente de configuración es o bien la memoria interna o bien la fuente de configuración externa, la determinación que comprende:
      - 15 comparar una primera comprobación de redundancia almacenada en el módulo de circuito con una segunda comprobación de redundancia almacenada en el módulo de controlador, en donde la primera comprobación de redundancia y la segunda comprobación de redundancia están asociados con el primer componente;
      - comparar un primer tiempo de configuración de módulo de circuito almacenado en el módulo de circuito con un segundo tiempo de configuración de módulo de circuito almacenado en el módulo controlador; y
      - 20 configurar el primer componente del módulo de circuito usando los datos de configuración desde la fuente de configuración seleccionada.
2. El módulo de circuito de la reivindicación 1, en donde la memoria interna comprende una memoria instantánea.
3. El módulo de circuito de la reivindicación 1 o 2, en donde el módulo de circuito comprende un módulo de comunicación (201).
4. El módulo de circuito de la reivindicación 1, 2 o 3, en donde el módulo de controlador comprende una unidad central de proceso CPU (202) y en donde el procesador soporta un segundo componente que está asociado con una configuración basada en CPU.
5. El módulo de circuito de la reivindicación 4, en donde el procesador está configurado además para:
  - 30 comparar una primera comprobación de redundancia almacenada en el módulo de circuito con una segunda comprobación de redundancia almacenada en el módulo de controlador, en donde la primera comprobación de redundancia y la segunda comprobación de redundancia están asociadas con el segundo componente; y
  - comparar un primer tiempo de configuración de CPU almacenado en el módulo de circuito con un segundo tiempo de configuración de CPU almacenado en el módulo de controlador.
6. El módulo de circuito de la reivindicación 4, en donde el procesador está configurado además para:
  - 35 en base a la comparación de las comprobaciones de redundancia y los tiempos de configuración de CPU, seleccionar la fuente de configuración o bien desde una fuente de configuración basada en CPU o bien la memoria interna.
7. El módulo de circuito de la reivindicación 5, en donde el procesador está configurado además para seleccionar la fuente de configuración además en base a un nuevo indicador de módulo.
8. El módulo de circuito de cualquier reivindicación precedente, en donde la interfaz de comunicación soporta una interfaz compatible seleccionada a partir de un grupo que consta de un canal principal serie universal USB, Modbus Plus, Ethernet, y un canal principal de placa posterior.
9. Un método para configurar un módulo de circuito (113, 115), el método que comprende:
  - 45 determinar un escenario de configuración a partir de un conjunto de escenarios de configuración posibles, el conjunto que incluye un primer escenario de configuración y un segundo escenario de configuración, la determinación que comprende:
    - comparar una primera comprobación de redundancia almacenada en el módulo de circuito con una segunda

- comprobación de redundancia almacenada en un módulo de controlador asociado, en donde la primera comprobación de redundancia y la segunda comprobación de redundancia están asociadas con un primer componente del módulo de circuito; y
- 5           comparar un primer tiempo de configuración de módulo de circuito con un segundo tiempo de configuración de módulo de circuito almacenado en el módulo de controlador asociado;
- cuando el escenario de configuración determinado es el primer escenario de configuración, seleccionar una fuente de configuración interna;
- cuando el escenario de configuración determinado es el segundo escenario de configuración, seleccionar una fuente de configuración externa basada en web; y
- 10          configurar el primer componente del módulo de circuito usando datos de configuración desde la fuente de configuración seleccionada.
- 10.** Un método para configurar un módulo de circuito según la reivindicación 9, en donde la fuente de configuración interna comprende una memoria instantánea (304).
- 15          **11.** El método de la reivindicación 9, en donde el primer componente está asociado con una configuración basada en CPU.
- 12.** El método de la reivindicación 11, que además comprende:
- comparar un primer tiempo de configuración de CPU almacenado en el módulo de circuito y un segundo tiempo de configuración de CPU almacenado en el módulo de controlador asociado.
- 13.** El método de la reivindicación 12, que además comprende:
- 20          en base a la comparación de las comprobaciones de redundancia y los tiempos de configuración de la CPU, determinar la fuente de configuración seleccionada o bien desde una fuente de configuración basada en CPU o bien la fuente interna de configuración.
- 14.** Un medio de almacenamiento legible por ordenador no transitorio que almacena instrucciones ejecutables por ordenador que, cuando se ejecutan, hace a un procesador realizar un método como se reivindica en cualquiera de las reivindicaciones 9 a 13.
- 25



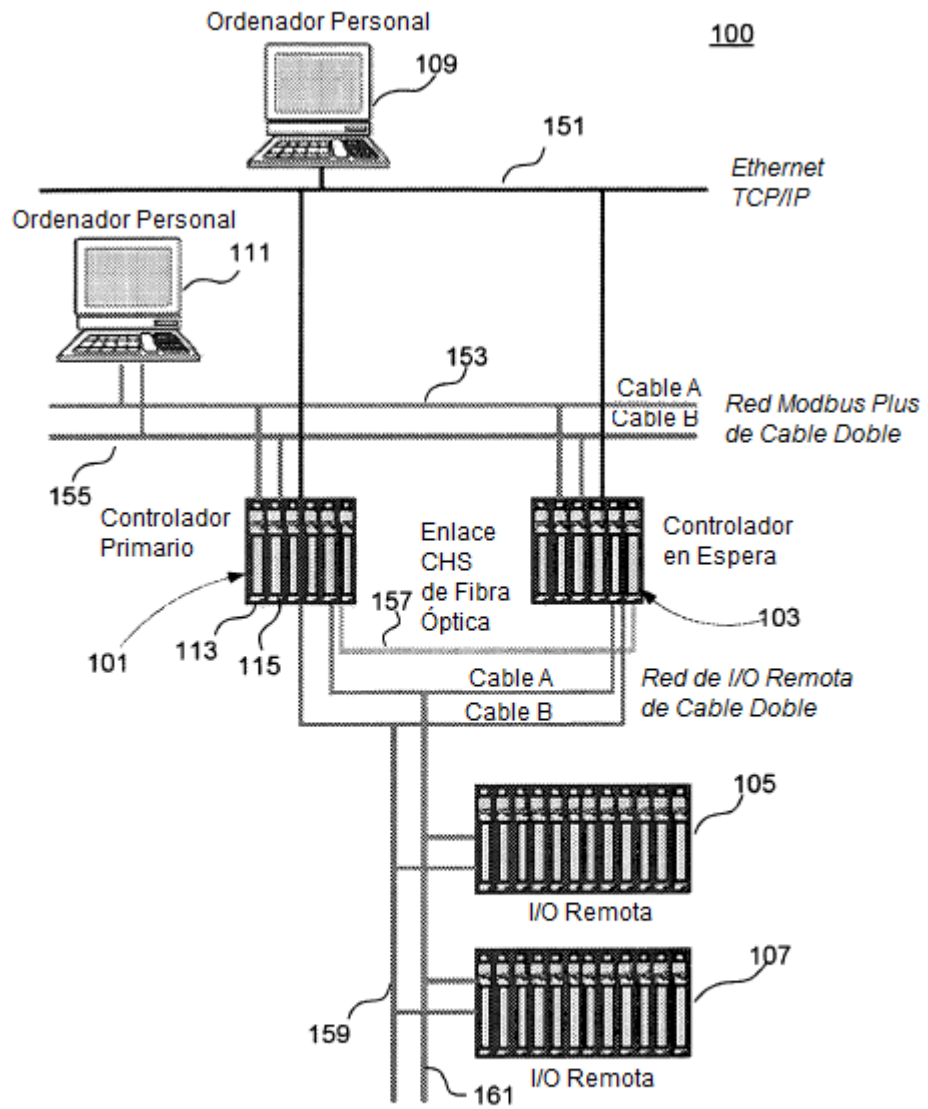


FIG. 1

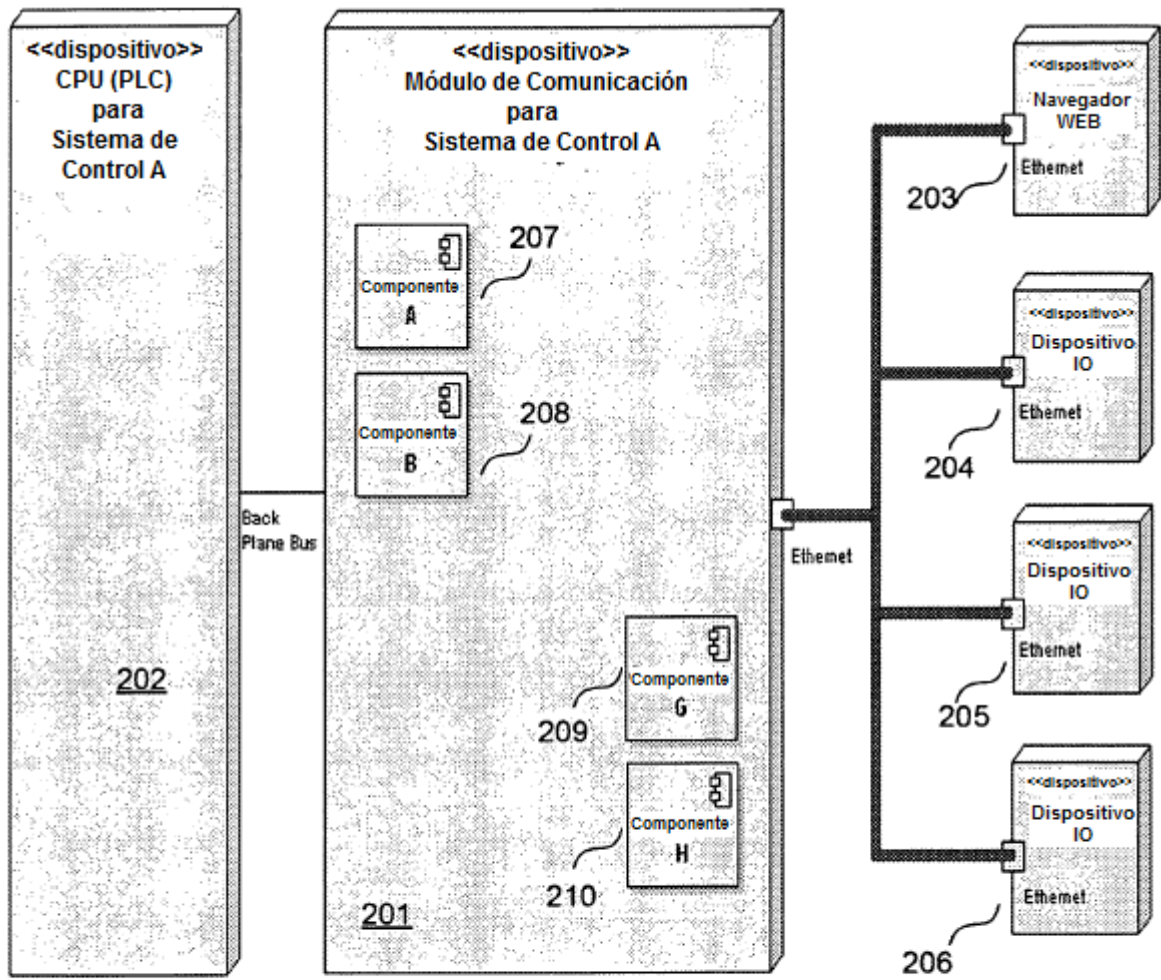


FIG. 2

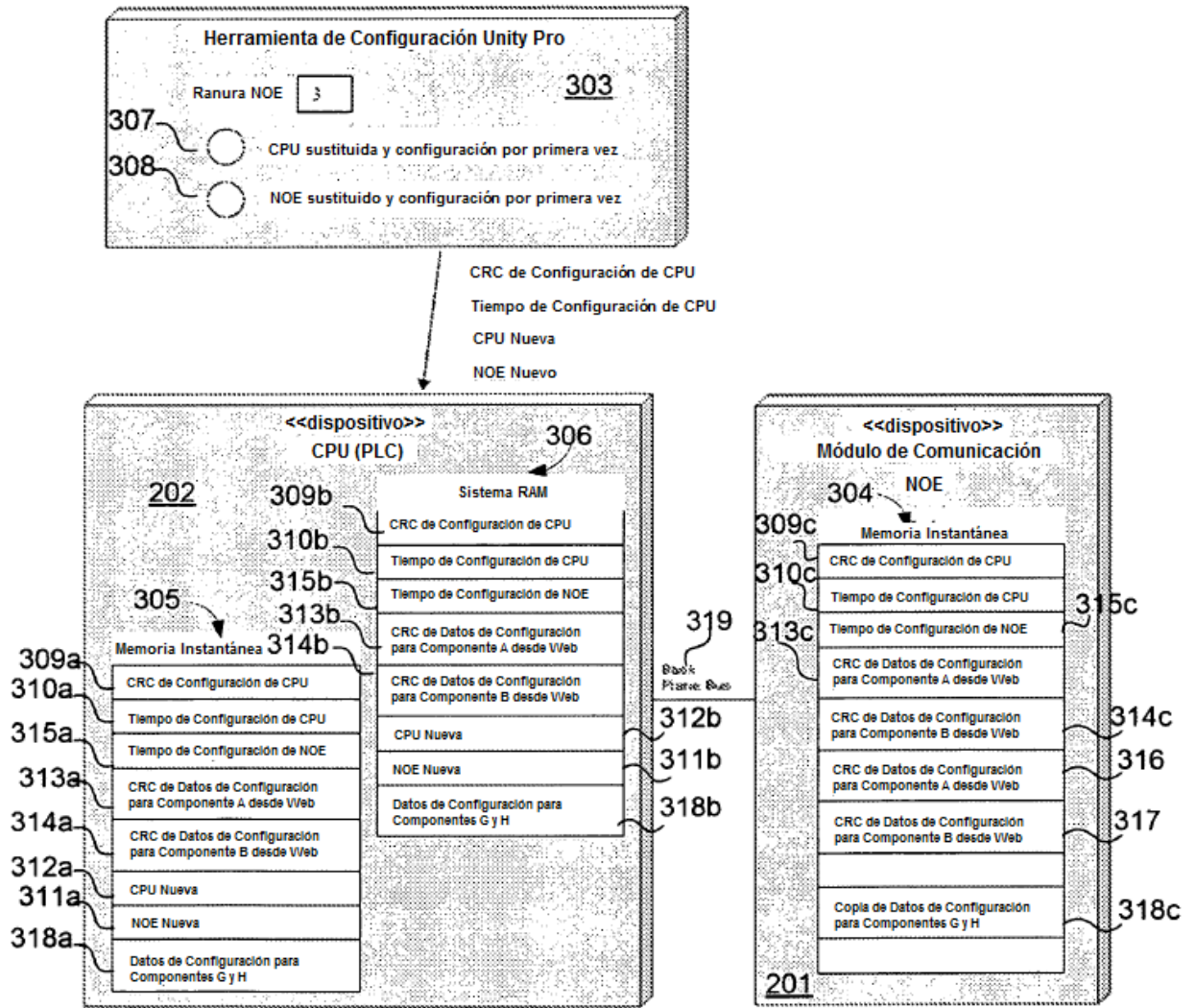
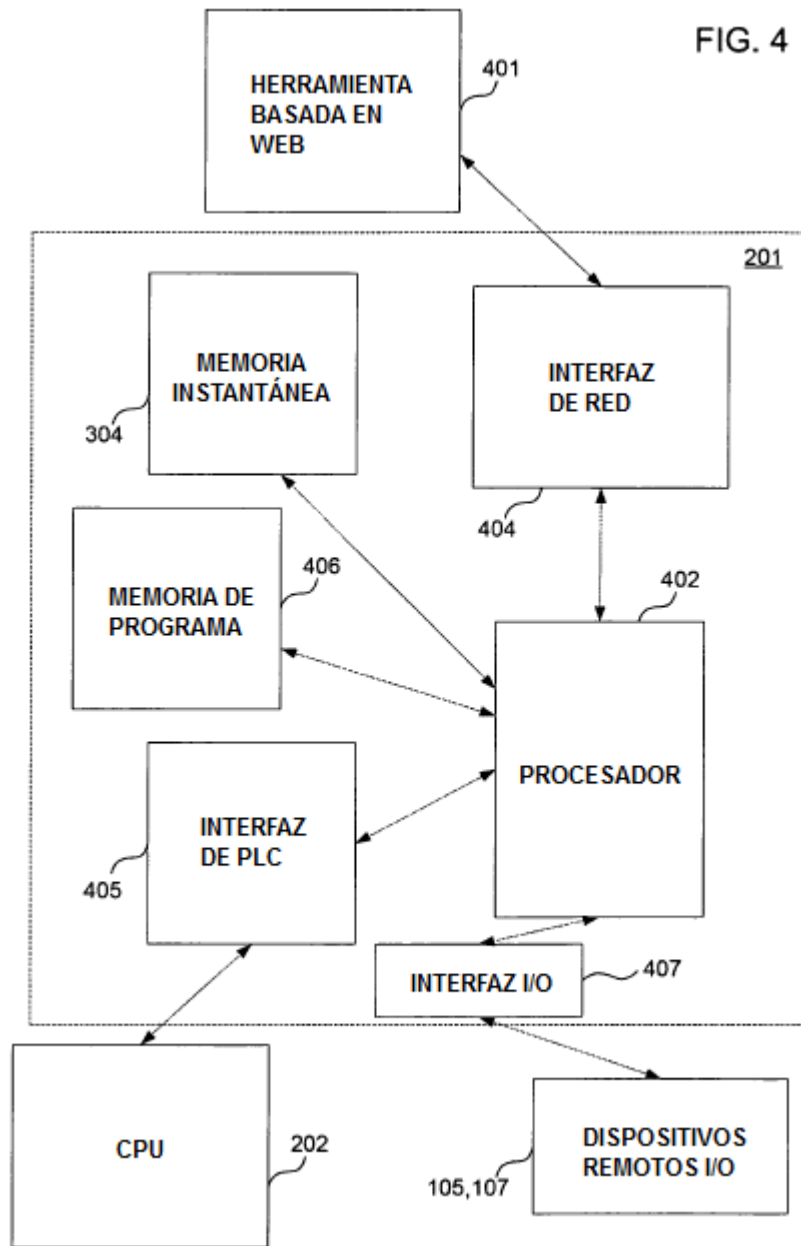


FIG. 3

FIG. 4



500

| Escenarios | CPU              |           |               | Módulo de Comunicación |           |
|------------|------------------|-----------|---------------|------------------------|-----------|
|            |                  |           |               |                        |           |
| 501        | Igual            | ENCENDIDA |               | Igual                  | reiniciar |
| 502        | Igual            | ENCENDIDA | Nueva config* | Igual                  | ENCENDIDO |
| 503        | Igual            | reiniciar |               | Igual                  | reiniciar |
| 504        | Igual            | ENCENDIDA |               | Nueva colocación       | reiniciar |
| 505        | Igual            | reiniciar |               | Nueva colocación       | reiniciar |
| 506        | Nueva colocación | reiniciar | Nueva config* | Igual                  | reiniciar |
| 507        | Nueva colocación | reiniciar | Nueva config* | Nueva colocación       | reiniciar |

FIG. 5

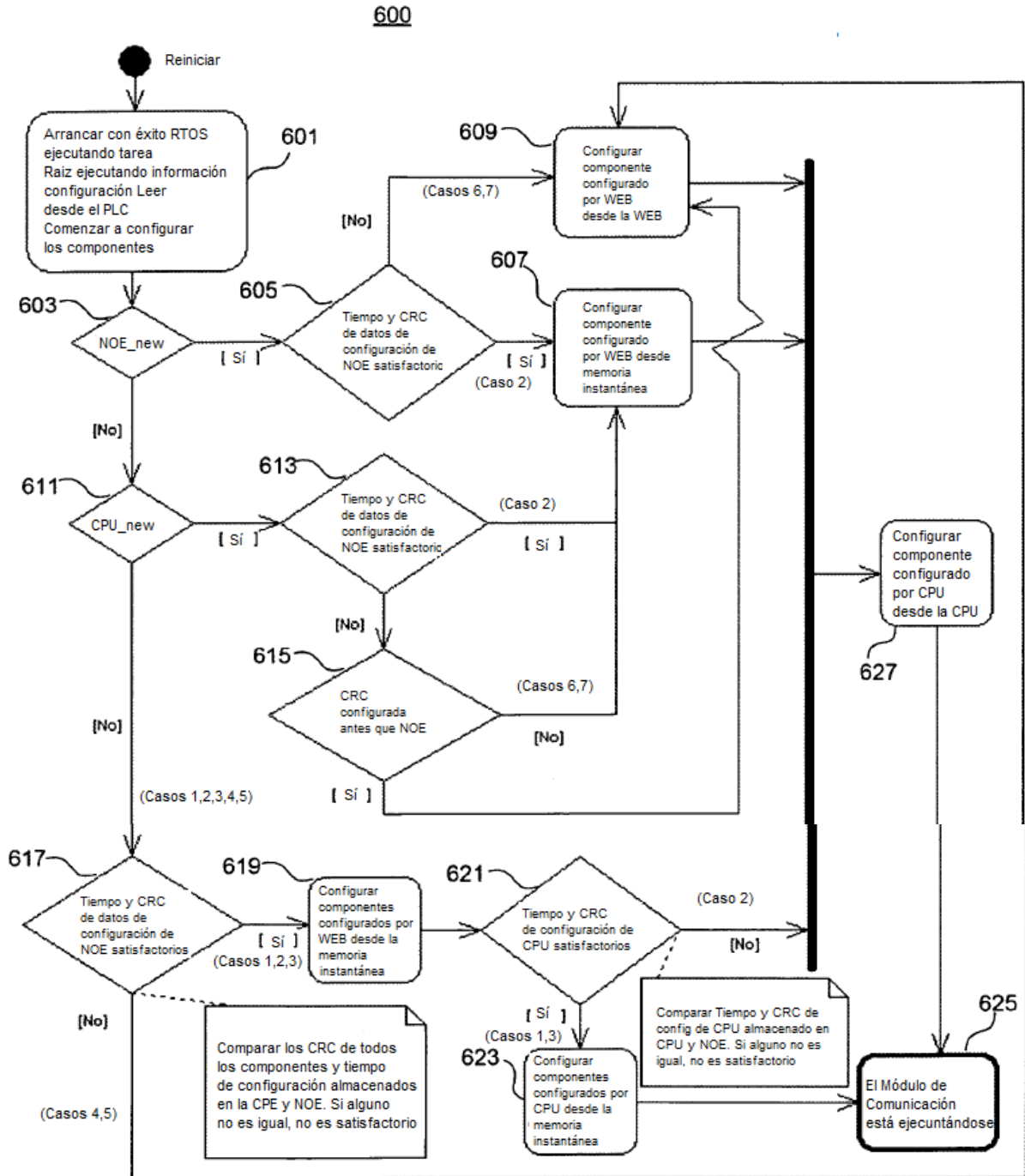


FIG. 6

607

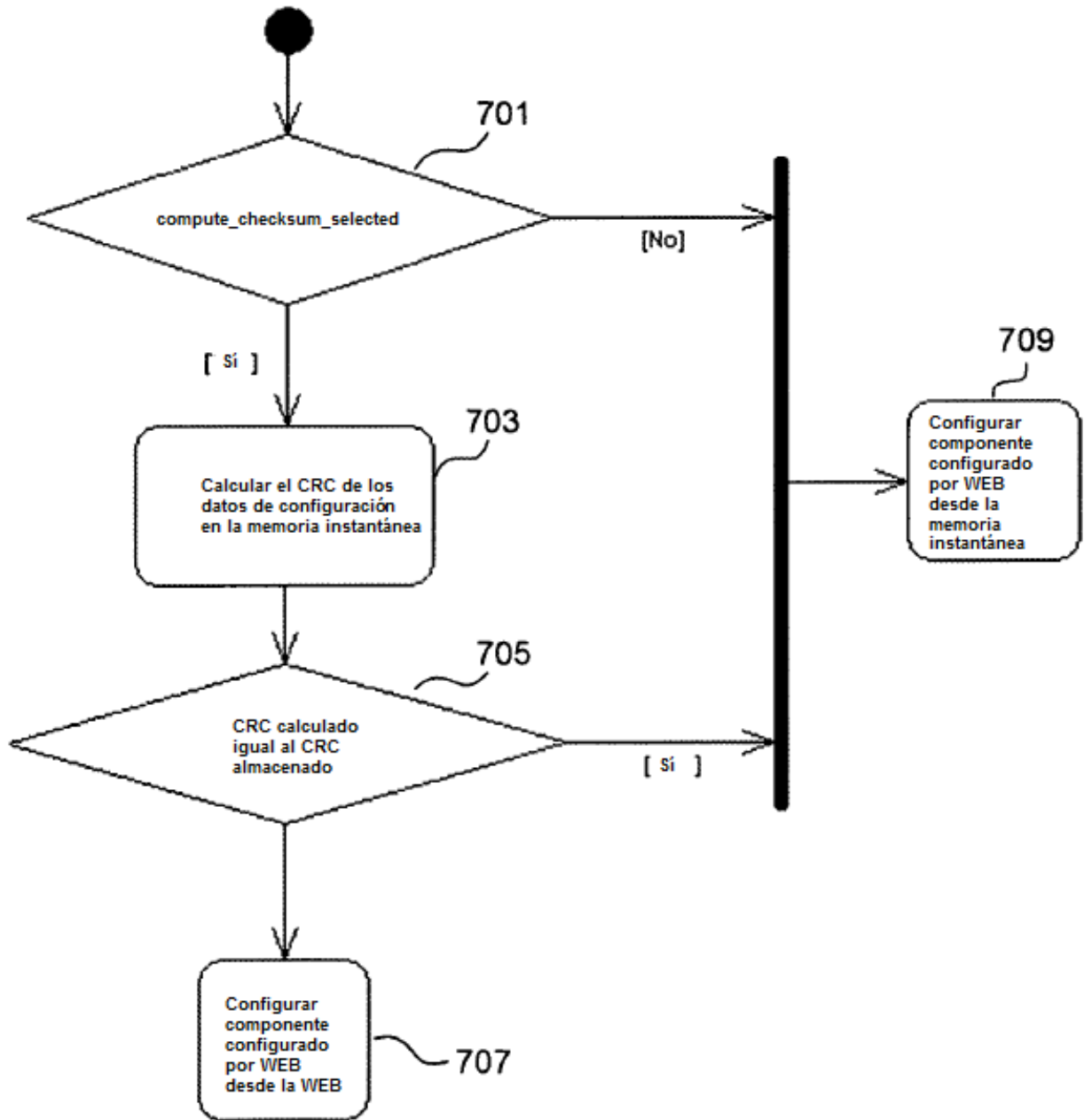


FIG. 7

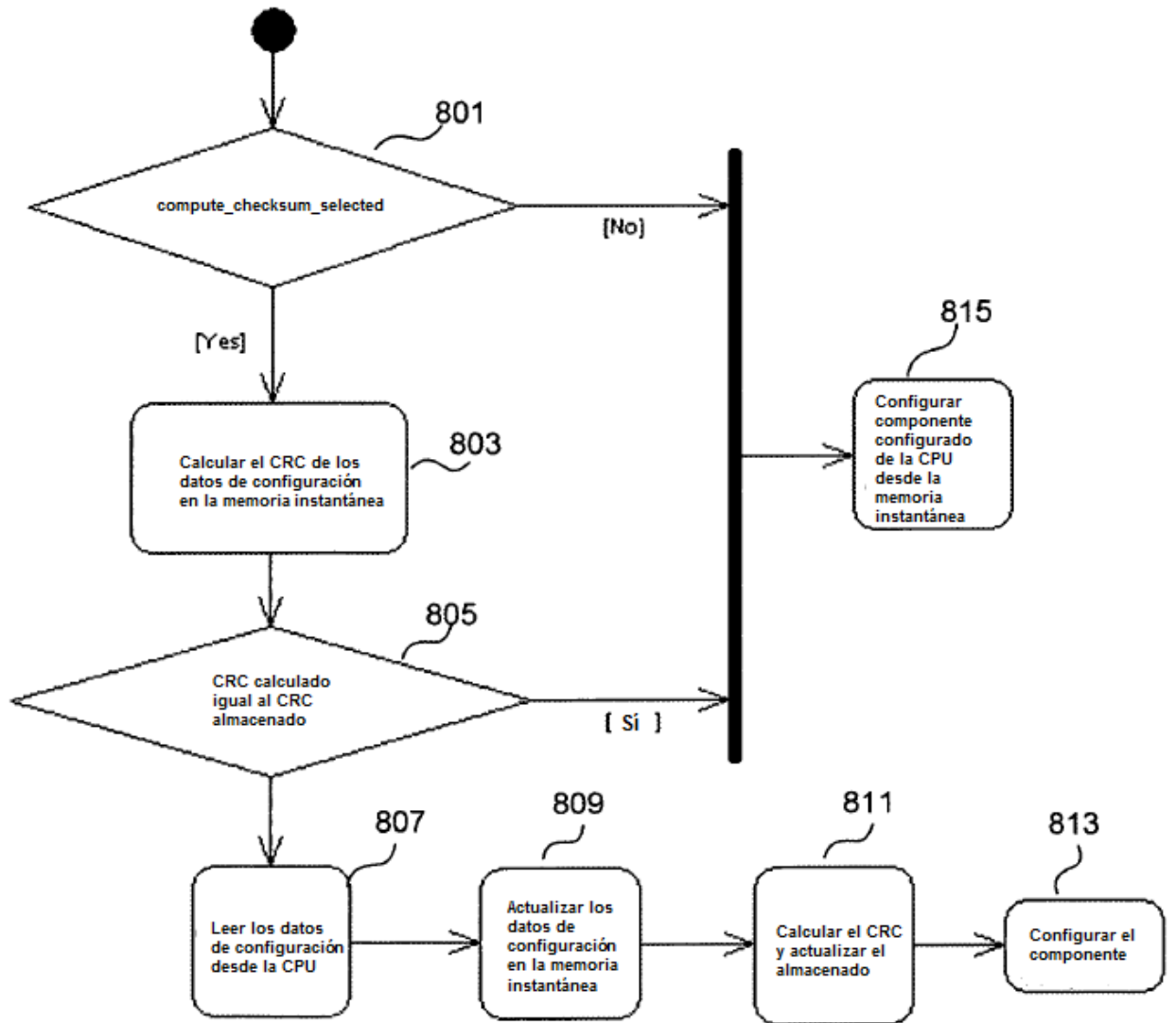


FIG. 8