

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **3 012 397**

51 Int. Cl.:

G05B 19/05 (2006.01)

G06F 8/656 (2008.01)

G06N 20/00 (2009.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **12.04.2022 PCT/EP2022/059787**

87 Fecha y número de publicación internacional: **19.10.2023 WO23198280**

96 Fecha de presentación y número de la solicitud europea: **12.04.2022 E 22722501 (8)**

97 Fecha y número de publicación de la concesión europea: **06.11.2024 EP 4285191**

54 Título: **Sistema de computación en la nube, procedimiento y programa informático**

45 Fecha de publicación y mención en BOPI de la traducción de la patente:
09.04.2025

73 Titular/es:
**SOFTWARE DEFINED AUTOMATION GMBH
(100.00%)
Lichtenbergstr. 8
85748 Garching bei München, DE**

72 Inventor/es:
WALTI, JOSEF

74 Agente/Representante:
DE ROOIJ, Mathieu Julien

ES 3 012 397 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

Sistema de computación en la nube, procedimiento y programa informático

5 **1. Campo técnico**

La presente divulgación se refiere a un sistema de computación en la nube, así como a un procedimiento y programa informático relacionados para configurar y optimizar el rendimiento de sistemas de automatización industrial.

10 **2. Antecedentes**

Desde la invención de los controladores lógicos programables (PLC) a fines de la década de 1960, la producción automatizada de bienes a gran escala ha aumentado la riqueza y la productividad de las economías industrializadas. El impacto general del sector manufacturero industrial en el PIB es del 14 % en los países del G10, y el consumo de energía es de alrededor del 30 % de la energía de las naciones industrializadas. En las últimas décadas, los proveedores de equipos de automatización industrial han desarrollado principalmente tecnología propia, desde el bus de campo y el dispositivo de entrada/salida (E/S), pasando por el nivel de PLC hasta sistemas de software industrial empresarial. Si bien dicha integración vertical de sistemas garantiza la compatibilidad dentro de la pila, genera fuertes dependencias y disminuye la interoperabilidad entre diferentes plataformas de tecnología de automatización, problemas de seguridad y costes elevados. Además, en dichas plataformas, los sistemas de edición de código de control industrial y los dialectos de instrucciones de código de control (normalmente algunas variaciones de un intento de estandarización de los años 1990, tal como IEC 61131/3) también se mantienen en propiedad, lo que da como resultado un acoplamiento estrecho de software y hardware en los sistemas de tiempo de ejecución del PLC. Normalmente, los sistemas de edición de código PLC se limitan a aplicaciones de Windows locales y la programación del PLC debe llevarse a cabo principalmente en proximidad física al dispositivo PLC real. Por lo tanto, modificar el comportamiento de las líneas de producción industrial es fundamentalmente una tarea manual que se realiza en ordenadores individuales conectados físicamente a un PLC ubicado en el armario de una máquina de producción.

Además, la arquitectura de sistema convencional de tales plataformas de tecnología de automatización industrial centradas en hardware no permite la integración de líneas de producción en sistemas de software de empresa modernos, que normalmente comprenden sistemas basados en microservicios que implementan funciones centrales de la empresa, tales como gestión del ciclo de vida del producto (PLM), gestión de la cadena de suministro (SCM), gestión de la relación con el cliente (CRM), planificación de recursos empresariales (ERP), etc. Si bien estos sistemas de software de función central corporativa ya emplean servicios basados en la nube, los cambios en la función central de fabricación, por ejemplo, por medio de la reprogramación/reconfiguración del PLC, a menudo todavía requieren el envío de un correo electrónico al departamento de ingeniería de automatización describiendo los cambios de un sistema de producción industrial que requiere modificación.

Además, el estrecho acoplamiento de hardware y software en los controladores industriales comunes ha generado una plétora de arquitecturas de hardware específicas de cada proveedor y da como resultado el uso de circuitos integrados específicos de la aplicación (ASIC), lo que genera largos plazos de producción para los equipos de automatización industrial y vulnerabilidad a las interrupciones de la cadena de suministro. Además, la respuesta actual al tiempo de ejecución determinista es utilizar software y hardware integrados en un único PLC, lo que genera el mayor coste de procesamiento, también en términos de mantenimiento y disponibilidad.

Por lo tanto, se necesitan más mejoras en la tecnología de automatización industrial.

50 En este contexto, el documento US 2021/0356944 A1 se refiere a un sistema y procedimientos para proporcionar una gestión centralizada de un sistema de automatización definido por software (SDA). El sistema SDA descrito comprende una colección de nodos controladores y una colección de nodos de cálculo lógicamente centralizados y, sin embargo, físicamente distribuidos.

55 Además, O. Givehchi, J. Imtiaz, H. Trsek y J. Jasperneite, "Control-as-a-service from the cloud: A case study for using virtualized PLCs", 10º Taller IEEE sobre sistemas de comunicación de fábrica, 2014, se refiere al concepto de implementar un PLC como un servicio dentro de una infraestructura basada en la nube y analiza el rendimiento de dichos PLC basados en la nube con respecto a los PLC heredados.

60 El documento US 2014/0336791 A1 se refiere a un servicio de mantenimiento predictivo basado en la nube que recopila datos industriales de múltiples clientes industriales para su almacenamiento y análisis en una plataforma en la nube. Los servicios de notificación basados en la nube también notifican a las entidades de soporte técnico apropiadas para facilitar el mantenimiento proactivo y la gestión de dispositivos.

65 El documento US 2015/120012 A1 se refiere a la sincronización entre la lógica de control en un controlador de automatización y una representación de esta lógica en una red de comunicación. La lógica de control se

implementa inicialmente en un controlador automático y una representación de esta lógica de control se almacena en la red de comunicación.

5 Cuando se realiza un cambio en la representación de la lógica de control en la red, la lógica de control se vuelve a implementar en el controlador de automatización para que el operador de la lógica de control esté sincronizado con la representación.

10 El documento US 2020/103842 A1 se refiere a un sistema de ingeniería que incluye una nube. La nube incluye una herramienta de ingeniería y un dispositivo virtual. La herramienta de ingeniería está configurada para crear y depurar software para operar un dispositivo de control que controla un dispositivo de campo instalado en una planta. El dispositivo virtual está configurado para simular un estado operativo del dispositivo de control de acuerdo con el software y con una entrada simulada o una entrada al dispositivo de control. La herramienta de ingeniería depura el software en base al resultado de funcionamiento del dispositivo virtual y en la salida del dispositivo de control o la entrada simulada. Badar Amjad y otros: "Intelligent Edge Control with Deterministic-/P based Industrial Communication in Process Automation", 2019, 15ª Conferencia internacional sobre gestión de redes y servicios (CNSM), investiga cómo extender la comunicación IP determinista al dominio de la tecnología operativa (OT) para respaldar las comunicaciones Ethernet industriales en tiempo real (RTE). Los autores han integrado un sistema en tiempo de ejecución de PLC (controlador lógico programable) basado en IEC-61131-3 en una pasarela de computación frontera ("*edge computing*"). Las tramas RTE se envuelven con una cabecera UDP/IP personalizada mediante un proxy y se envían a los enrutadores deterministas. Los enrutadores reenvían paquetes con un retraso limitado de menos de 30 μ s por salto. Validamos nuestro enfoque usando una configuración de prueba experimental, un PLC virtualizado (vPLC) dentro del dispositivo frontera que controla de forma remota la aplicación PA (biorreactor) pasando a través de servidores proxy y enrutadores deterministas en una red heterogénea.

25 El documento US 2021/389968 A1 se refiere a un controlador frontera que puede utilizarse para obtener datos del dispositivo de uno o más dispositivos locales en una instalación local y para proporcionar una representación de al menos algunos de los datos del dispositivo a un servidor remoto. El controlador fronterero ("*edge controller*") puede incluir un puerto de comunicación de red, un puerto de comunicación celular y un puerto de comunicación de dispositivo. Un controlador está acoplado operativamente al puerto de comunicación de red, al puerto de comunicación celular y al puerto de comunicación del dispositivo y está configurado para recibir información de configuración e instalar la información de configuración recibida en el controlador frontera. La información de configuración instalada configura el controlador para obtener los datos del dispositivo del uno o más dispositivos locales y para enviar una representación de al menos algunos de los datos del dispositivo al servidor remoto.

35 **3. Breve explicación**

En un primer aspecto, la presente divulgación proporciona un procedimiento para configurar un controlador de un sistema de automatización industrial, que comprende obtener, mediante un sistema de computación en la nube, una configuración de controlador y un programa de controlador. El procedimiento comprende, además, generar, mediante el sistema de computación en la nube, en base a la configuración de controlador obtenida y al programa de controlador obtenido, una representación en tiempo de compilación del controlador que permite la modificación del programa de controlador y de la configuración de controlador en tiempo de compilación que comprende un tiempo en o antes de la generación del código de máquina en tiempo de ejecución para el controlador a partir de un código fuente correspondiente, y un tiempo durante el cual el código fuente se genera extrayendo el código fuente de un archivo de configuración o escribiendo el código fuente por medio de un lenguaje de programación o por medio de un entorno de desarrollo de código fuente basado en la nube; y generar, mediante el sistema de computación en la nube, un programa de controlador modificado y una configuración de controlador modificada, usando la representación en tiempo de compilación del controlador, en el que el programa de controlador define el funcionamiento en tiempo de ejecución del controlador que controla un subsistema del sistema de automatización industrial. A continuación, el sistema de computación en la nube proporciona el programa de controlador modificado y la configuración de controlador modificada al controlador que se está configurando por medio de una red. Allí, la representación en tiempo de compilación del controlador comprende una capa de persistencia que contiene configuraciones del controlador y elementos de instrucciones de control de programas del controlador, los elementos de control individuales se representan como objetos de lógica de control separados y los elementos individuales de la capa de persistencia se pueden configurar individualmente en base a instrucciones de programación y configuración recibidas por medio de una interfaz de programación.

60 Por ejemplo, partes de la configuración de controlador y/o del programa de controlador pueden obtenerse, en el sistema de computación en la nube, desde el controlador por medio de la red, desde un archivo de configuración de controlador y/o desde una biblioteca de software o un repositorio de código fuente que resida dentro o fuera del sistema de computación en la nube.

65 Por ejemplo, los archivos de proyecto de PLC generados por medio de software específico del proveedor, tal como Siemens TIA Portal, Rockwell Studio 5000, Codesys Engineering, etc., se pueden importar y usar para importar partes de la configuración de controlador y/o el programa de controlador.

Como alternativa o adicionalmente, el programa de controlador también puede obtenerse, en el sistema de computación en la nube, desde un servicio de entorno de desarrollo integrado (IDE) alojado por el sistema de computación en la nube o conectado a él por medio de una red.

5 Además, la configuración de controlador comprende uno o más de los siguientes: información sobre el tipo y la capacidad del controlador, configuración de red del controlador, información de interfaz para unidades electromecánicas y configuración de interfaz para dispositivos de E/S conectados al controlador por medio de la integración en panel posterior de sistemas de bus de campo industriales ("*Industrial Field Bus Systems*").

10 Uno de los aspectos clave de la presente divulgación es que dicha representación en tiempo de compilación del controlador de automatización industrial (también designado como sinónimo *Servicio PLCTwirt* en lo sucesivo) pueden existir en la misma infraestructura de TI/nube que otros microservicios y sistemas de software centrales de la empresa.

15 En este caso el término *tiempo de compilación* debe entenderse que cubre el tiempo en, o antes de la generación del código de máquina en tiempo de ejecución a partir de un código fuente correspondiente, así como el tiempo durante el cual se genera el código fuente, por ejemplo, el tiempo necesario para extraer el código fuente de un archivo de configuración o el tiempo necesario para escribir el código fuente por medio de un lenguaje o sistema de programación de alto nivel, tal como un IDE basado en la nube.

20 Por ejemplo, el servicio PLC Twin permite la modificación (por ejemplo, por medio de un IDE conectado a una GUI o una API) de un programa de controlador existente/obtenido, en tiempo de compilación, mientras o antes de que un ejecutable correspondiente se esté ejecutando actualmente, en tiempo de ejecución, en uno o más controladores industriales (por ejemplo, PLC, PLC virtuales, etc.) de un sistema de automatización industrial.

25 De este modo, aspectos de la presente divulgación permiten optimizar el tiempo de ejecución de los sistemas de automatización en lo que respecta al tiempo de respuesta y al coste de cálculo. La ejecución de tareas de control de automatización sensibles en tiempo real aún se puede realizar cerca de los activos industriales bajo control mientras el servicio PLC Twin abstrae las principales funciones del PLC en diferentes plataformas tecnológicas. Basado en un servicio de conectividad seguro y no intrusivo, el servicio PLC Twin abstrae completamente la lógica de control del PLC y permite la implementación automática del PLC en todo el sistema en cuestión de minutos.

30 Como se analiza con más detalle a continuación, los aspectos de la presente divulgación permiten que una arquitectura empresarial de microservicios por encima del nivel de control en tiempo real interactúe de forma nativa con la automatización industrial; en esencia, la automatización industrial se está convirtiendo en un conjunto de microservicios en la pila central de la empresa. Además, aspectos de la presente divulgación permiten construir API programáticas sobre instrucciones de control como un facilitador clave para la optimización del sistema de bucle completo.

40 Además, algunos aspectos de la presente divulgación también habilitan características de desarrollo y operaciones de CI/CD modernas, tal como IDE como servicio (IDEaaS), control de versiones de código de PLC, políticas de seguridad y acceso configurables, supervisión de la integridad del código en todos los servicios de la flota de PLC, etc.

45 Como se analiza con más detalle a continuación, el programa de controlador modificado se puede proporcionar luego a un servicio de compilador (conectado operativamente o integrado con el servicio PLC Twin) alojado por el sistema de computación en la nube, generando un ejecutable en tiempo de ejecución que luego se proporciona, por medio de una red, a uno o más servicios de PLC basados en hardware o PLC virtuales, alojados por un sistema de computación fronteriza, que controlan un subsistema de un sistema de automatización industrial. Dicho servicio de compilación puede ser un servicio de computación en la nube externo al servicio PLC Twin o integrado en él.

50 Por ejemplo, el servicio de compilador y/o el servicio PLC Twin pueden implementarse en un entorno de virtualización configurable, tal como un contenedor o microservicio que especifica políticas de acceso de seguridad y/o parámetros de utilización de recursos del sistema, además de comprender bibliotecas de software necesarias para la ejecución aislada por diferentes tipos de equipos de computación en la nube (virtuales).

55 Una vez que el programa de controlador y los parámetros de configuración se representan por medio de dicho servicio PLC Twin, se pueden utilizar interfaces de programación tales como *interfaces de programación de aplicaciones* (API) para adaptar programáticamente el código del controlador y/o los parámetros de configuración por medio de otros sistemas de software de la empresa como PLM, SCM, CRM, ERP, Analytics, AI y/o mediante una interfaz gráfica de usuario mostrada en un dispositivo terminal y conectada al sistema de computación en la nube por medio de una red. De este modo, el servicio PLC Twin constituye la base para la automatización basada en la nube de la ingeniería de automatización.

65 Así, algunos aspectos pueden comprender, además, recibir, en el sistema de computación en la nube, por medio de una interfaz de programación (por ejemplo, una API), instrucciones de programación y configuración para

modificar el programa de controlador y/o la configuración de controlador y generar, usando la representación en tiempo de compilación, el programa de controlador modificado y/o la configuración de controlador modificada en base a las instrucciones recibidas.

5 Por ejemplo, dichas instrucciones de programación y configuración podrían recibirse por medio de una API desde un segundo sistema de computación en la nube, en el que las instrucciones de programación y configuración pueden configurarse para modificar el programa de controlador y/o la configuración de controlador en base a los requisitos de gestión del ciclo de vida del producto ("*Product Lifecycle Management*"), PLM, requisitos de ejecución de fabricación, requisitos de gestión de relaciones con el cliente ("*Customer Relationship Management*"), CRM, requisitos de gestión de la cadena de suministro ("*Supply Chain Management*"), SCM, y/o requisitos de planificación de recursos empresariales ("*Enterprise Resource Planning*"), ERP.

10 En algunas implementaciones, el programa de controlador y/o las instrucciones de programación y configuración obtenidos también pueden generarse usando un servicio IDE, preferentemente en contenedores, y alojado por el sistema de computación en la nube.

15 De este modo, algunos aspectos de la presente divulgación permiten la integración de la configuración y reconfiguración del sistema de automatización industrial en los sistemas de software existentes de la empresa basados en la nube, lo que da como resultado menos tiempo de inactividad, menos costes y una mayor producción.

20 Básicamente, la capa de persistencia puede actuar como una única fuente de verdad (por ejemplo, por medio de un módulo de almacenamiento de datos basado en la nube que proporciona capacidades de control de versiones de código) para todo el código de control utilizado por los controladores (por ejemplo, PLC, PLC virtuales, etc.) de un sistema de automatización industrial, lo que permite una multitud de beneficios. Por ejemplo, un beneficio de seguridad clave es la capacidad de respaldo total de líneas de producción completas o incluso fábricas completas en caso de una incidencia de seguridad (por ejemplo, un incidente de tipo STUXnet), por ejemplo, en la capa de control. El uso de dicha capa de persistencia o funciones de persistencia similares mejora aún más la depuración y el mantenimiento del sistema por medio de capacidades de control de versiones de código a nivel de todo el sistema e independientes de la tecnología.

25 Algunos aspectos pueden comprender, además, recibir, en el sistema de computación en la nube, por medio de la red, datos de sensor asociados a un subsistema del sistema de automatización industrial, controlado por el controlador y generar, mediante el sistema de computación en la nube, instrucciones de programación y configuración para modificar el programa de controlador y/o la configuración de controlador, en base al análisis de los datos de sensor recibidos.

30 De esta manera, la presente divulgación permite una reconfiguración completa de bucle cerrado de sistemas de automatización industrial de una manera independiente de la tecnología y del proveedor y, por lo tanto, proporciona la base técnica para sistemas de producción optimizados automáticamente.

35 Si bien dicha reconfiguración de bucle cerrado de sistemas de automatización industrial funciona de manera más eficiente y segura mediante el uso de una representación en tiempo de compilación de los controladores involucrados, como se analizó anteriormente, otras implementaciones son concebibles y están cubiertas por la presente divulgación. De este modo, la presente divulgación también proporciona un procedimiento para reconfigurar un controlador de un sistema de automatización industrial, que comprende: recibir, en el sistema de computación en la nube, por medio de una red, datos de sensor asociados a un subsistema del sistema de automatización industrial, controlado por el controlador, generar, mediante el sistema de computación en la nube, un programa de controlador modificado y una configuración de controlador modificada, en base al análisis de los datos de sensor recibidos y proporcionar, mediante el sistema de computación en la nube por medio de una red, el programa de controlador modificado y la configuración de controlador modificada al controlador del sistema de automatización industrial, en el que el programa de controlador define el funcionamiento en tiempo de ejecución del controlador.

40 Algunas implementaciones pueden comprender, además, obtener, mediante el sistema de computación en la nube y en base a los datos de sensor recibidos, una métrica de calidad asociada al funcionamiento del subsistema controlado por el controlador, comparar, mediante el sistema de computación en la nube, la métrica de calidad con un requisito de funcionamiento para el sistema de automatización industrial (por ejemplo, requisitos generales de salida del sistema, frecuencia de defectos del producto, etc.) y generar, mediante el sistema de computación en la nube, las instrucciones de programación y configuración en base a la comparación de la métrica de calidad con el requisito de funcionamiento.

45 Por ejemplo, el sistema de computación en la nube puede emplear software de computación en la nube, tal como una red neuronal entrenada, para evaluar/clasificar la calidad del producto en base a los datos del sensor de imagen recibidos y comparar una frecuencia o porcentaje de productos fabricados con baja calidad con un requisito de funcionamiento recibido de otros sistemas de software de la empresa (por ejemplo, el defecto del producto debe ser inferior al 5 por ciento). Con base en esta comparación, el sistema de computación en la nube puede generar

instrucciones de programación y configuración y proporcionárselas a la representación en tiempo de compilación de uno o más controladores para cambiar el comportamiento en tiempo de ejecución de los controladores para disminuir los defectos del producto.

5 Por ejemplo, el software de computación en la nube también puede configurarse para determinar una causa probable y/o una modificación adecuada del programa de controlador para que coincida con el requisito de funcionamiento en base a los datos del sensor recibidos.

10 En algunas implementaciones, el controlador puede ser un controlador virtual implementado por un software de computación frontera ejecutado por un sistema de computación frontera conectado operativamente al sistema de computación en la nube y al sistema de automatización industrial. En dichas implementaciones, la configuración de controlador puede especificar un entorno de virtualización para ejecutar el software de computación frontera que implementa el controlador virtual y puede comprender una configuración de red de un dispositivo físico de E/S conectado a un subsistema del sistema de automatización industrial que será controlado por el controlador virtual.

15 En algunas implementaciones, el sistema de computación en la nube puede crear una instancia del controlador virtual por medio de la red en una máquina anfitriona del sistema de computación frontera administrado por un hipervisor en tiempo real usando la representación en tiempo de compilación del controlador virtual.

20 Para mejorar la seguridad, la escalabilidad y aumentar la eficiencia de los recursos de computación frontera, el código que implementa dichos PLC virtuales también se puede contenerizar como se analizó anteriormente para el servicio PLC Twin y/o el servicio del compilador.

25 De este modo, los aspectos de la presente divulgación permiten la (re)configuración de sistemas de control industrial complejos empleando una combinación de controladores virtuales y basados en hardware desde una interfaz uniforme basada en la nube que trata diferentes tipos, modelos e implementaciones de equipos de control industrial en pie de igualdad. De esta manera, las ventajas clave de la computación en la nube, la computación frontera y los PLC basados en hardware se pueden integrar de forma sinérgica en un único sistema de control de automatización industrial híbrido con seguridad, escalabilidad y flexibilidad enormemente mejoradas, manteniendo al mismo tiempo un comportamiento determinista y en tiempo real cuando sea necesario. Como se analizó anteriormente, un sistema de control de automatización industrial híbrido integrado de este tipo se puede interconectar fácilmente con otros sistemas de software centrales de la empresa, lo que da como resultado ganancias significativas en productividad, capacidades de personalización de productos y reducción del tiempo de inactividad de las instalaciones de fabricación.

35 Algunas implementaciones pueden comprender, además, la recepción, en el sistema de computación en la nube, de datos de supervisión del sistema de computación frontera que caracterizan el rendimiento del controlador virtual. De manera similar a los datos de sensor que recibe el sistema de computación en la nube, dichos datos de supervisión pueden usarse para optimizar el rendimiento del sistema de automatización industrial como se analizó anteriormente.

40 En otro aspecto, la presente divulgación también proporciona un sistema de computación en la nube para configurar controladores de un sistema de automatización industrial, comprendiendo el sistema de computación en la nube uno o más nodos de computación en la nube, proporcionando cada uno de los cuales recursos de procesamiento, memoria y red para la ejecución de software de computación en la nube, en el que los nodos de computación en la nube están configurados para recibir y transmitir, por medio de una red, datos desde y hacia los controladores y, opcionalmente, desde uno o más sensores que supervisan el sistema de automatización industrial, en el que el uno o más nodos de computación en la nube están configurados para ejecutar software de computación en la nube para configurar los controladores, por medio de la red, realizando uno de los procedimientos como se analizó anteriormente y en lo siguiente.

50 En otro aspecto, la presente divulgación también proporciona un programa informático que comprende instrucciones para llevar a cabo uno de los procedimientos analizados anteriormente y a continuación, cuando es ejecutado por dicho sistema de computación en la nube.

55 Como se analizó anteriormente, la presente divulgación también proporciona un sistema de control de automatización industrial distribuido que comprende un primer sistema de computación en la nube como se describió anteriormente, uno o más controladores conectados al sistema de computación en la nube por medio de una red y un segundo sistema de computación en la nube que se comunica con el primer sistema de computación en la nube por medio de una API.

60 En un ejemplo adicional no reivindicado, la presente divulgación proporciona un procedimiento para optimizar el rendimiento de un sistema de automatización industrial, que comprende generar, por un sistema de computación en la nube (por ejemplo, mediante el uso de un servicio PLC Twin como se analizó anteriormente), un programa de controlador modificado y/o una configuración de controlador modificada para un controlador que controla un subsistema del sistema de automatización industrial, en el que el programa de controlador define el funcionamiento

en tiempo de ejecución del controlador. El procedimiento comprende, además, proporcionar, por el sistema de computación en la nube por medio de una red, el programa de controlador modificado y/o la configuración de controlador modificada al controlador para modificar el funcionamiento en tiempo de ejecución del controlador, recibir, en el sistema de computación en la nube por medio de la red, datos de sensor para el sistema de automatización industrial, estimar, por el sistema de computación en la nube y en base a los datos de sensor recibidos, un cambio de rendimiento del sistema de automatización industrial causado por el funcionamiento en tiempo de ejecución modificada del controlador y optimizar, por el sistema de computación en la nube, el rendimiento del sistema de automatización industrial en base al cambio de rendimiento estimado.

En algunas implementaciones, optimizar el rendimiento del sistema de automatización industrial puede comprender uno o más de los siguientes: almacenar, en un subsistema de memoria, una estructura de datos que correlacione la modificación del programa de controlador y/o la configuración de controlador con el cambio estimado en el rendimiento, obtener, en base al cambio estimado en el rendimiento, una modificación posterior del programa de controlador y/o la configuración de controlador y comparar el cambio estimado en el rendimiento con una predicción obtenida de un modelo computacional del sistema de automatización industrial.

Por ejemplo, al almacenar una estructura de datos de este tipo que correlacione la modificación del programa de controlador y/o la configuración de controlador con el cambio estimado en el rendimiento, se puede generar un conjunto de entrenamiento de muestras etiquetadas, preferentemente en muchos sistemas de automatización industrial diferentes que se pueden usar como entrada para paradigmas modernos de aprendizaje automático, tal como el aprendizaje de refuerzo.

Por ejemplo, la obtención de la modificación posterior del programa de controlador y/o la configuración de controlador puede basarse en un algoritmo de búsqueda estocástica o en un aprendizaje de refuerzo.

Algunas implementaciones pueden comprender el entrenamiento de un modelo de red neuronal de aprendizaje de refuerzo profundo basado al menos en parte en un conjunto de entrenamiento que comprende correlaciones almacenadas del programa de controlador y/o las modificaciones de configuración de controlador con los cambios estimados en el rendimiento, en el que, opcionalmente, el conjunto de entrenamiento se obtiene de múltiples sistemas de automatización industrial diferentes.

De esta manera, la presente divulgación proporciona una optimización autónoma basada en IA de sistemas de automatización industrial complejos que pueden comprender una pluralidad de diferentes tecnologías, tipos y modelos de controladores.

4. Breve descripción de las figuras

A continuación se describen con más detalle diversos aspectos y detalles de implementación de la presente divulgación haciendo referencia a las figuras adjuntas. Estas figuras muestran:

Figura 1: un diagrama de bloques funcionales de un sistema de computación en la nube para configurar controladores industriales de acuerdo con algunos aspectos de la presente divulgación.

Figura 2: un diagrama de bloques funcionales de una implementación de un sistema de computación en la nube de acuerdo con algunos aspectos de la presente divulgación.

Figura 3: un diagrama de bloques funcionales de una implementación de un sistema de computación en la nube adaptado para permitir la (re)configuración de bucle cerrado de sistemas de automatización industrial de acuerdo con algunos aspectos de la presente divulgación.

Figura 4: un diagrama de proceso que ilustra un procedimiento para configurar un controlador de un sistema de automatización industrial de acuerdo con algunos aspectos de la presente divulgación.

Figura 5: un diagrama de proceso que ilustra un procedimiento para optimizar el rendimiento de un sistema de automatización industrial de acuerdo con algunos aspectos de la presente divulgación.

5. Descripción de los modos de realización ejemplares

A continuación, se describen con más detalle algunos modos de realización ejemplares de la presente divulgación, con referencia a procesos y sistemas de computación ejemplares. Naturalmente, los sistemas de computación proporcionados por la presente divulgación pueden emplear componentes de hardware estándar (por ejemplo, los nodos o servidores de computación en la nube se conectan entre sí por medio de tecnología de red cableada o inalámbrica convencional). En algunas implementaciones, también se puede emplear hardware específico de la aplicación (por ejemplo, circuitos para entrenar modelos de redes neuronales y/o circuitos para ejecutar modelos entrenados, etc.). Además, dichos sistemas de computación están configurados para ejecutar instrucciones de software (por ejemplo, recuperadas de circuitos de memoria no transitoria ubicados en el mismo lugar o remotos)

para ejecutar los procedimientos implementados por ordenador analizados en la sección anterior.

Si bien en los párrafos siguientes se describen combinaciones de rasgos característicos específicos con respecto a modos de realización ejemplares de la presente divulgación, se debe entender que no todas las características de los modos de realización analizados tienen que estar presentes para realizar la divulgación, que está definida por la materia objeto de las reivindicaciones. Los modos de realización divulgados pueden modificarse combinando ciertos rasgos característicos de un modo de realización ejemplar con uno o más rasgos característicos técnicos y funcionalmente compatibles de otros modos de realización ejemplares. En concreto, el experto en la materia entenderá que los rasgos característicos, componentes, etapas de procesamiento y/o elementos funcionales de un modo de realización ejemplar se pueden combinar con rasgos característicos, etapas de procesamiento, componentes y/o elementos funcionales técnicamente compatibles de cualquier otro modo de realización ejemplar de la presente divulgación siempre que estén cubiertos por las especificaciones proporcionadas por las reivindicaciones adjuntas.

Además, los diversos modos de realización expuestos en el presente documento pueden implementarse en hardware, software o una combinación de ambos. Por ejemplo, los diversos componentes, elementos, subsistemas, módulos, etc. de los sistemas descritos en el presente documento también pueden implementarse por medio de software específico de la aplicación que se ejecuta en equipos de procesamiento de datos y señales multipropósito, tales como servidores, nodos de cómputo, CPU, DSP y/o sistemas en un chip, SOC o componentes similares o cualquier combinación de los mismos. Algunas implementaciones también emplean componentes de hardware específicos de la aplicación, tales como circuitos integrados específicos de la aplicación, ASIC y/o conjuntos de puertas programables en campo, FPGA y/o componentes similares y/o cualquier combinación de los mismos.

Por ejemplo, los diversos (sub)sistemas de computación analizados en el presente documento pueden implementarse, al menos en parte, en equipos de procesamiento de datos multipropósito, tales como servidores de computación en la nube y/o frontera.

La figura 1 muestra un diagrama de bloques funcionales que ilustra la arquitectura del sistema, las funciones y el funcionamiento de un sistema de computación en la nube de acuerdo con un aspecto de la presente divulgación. Como se analizó en la sección 3. "Breve explicación" anterior, el sistema de computación en la nube 110 puede comprender uno o más nodos de computación en la nube 112, proporcionando cada uno de los cuales recursos de procesamiento (por ejemplo, virtualizados) 114, recursos de memoria 116 y recursos de red 118 para la ejecución distribuida basada en la nube de software de computación en la nube (no mostrado).

Los nodos de computación en la nube 112 están configurados para recibir y transmitir, por medio de una red 120 (por ejemplo, una red basada en IP tal como Internet), datos desde y hacia los controladores 132 de un sistema de automatización industrial 130 y, opcionalmente, desde y hacia uno o más sensores (no mostrados) que supervisan el funcionamiento del sistema de automatización industrial 130. El sistema de automatización industrial 130 también puede comprender (o puede estar conectado a) un equipo de computación frontera 134 (por ejemplo, uno o más nodos de computación frontera que ejecutan software de computación frontera) que puede estar configurado para alojar controladores industriales virtuales virtualizados y preferentemente contenerizados como se analiza en la sección 3 anterior. El sistema de automatización industrial 130 también puede comprender controladores industriales basados en hardware 132 tales como PLC.

Los controladores virtuales y basados en hardware del sistema de automatización industrial 130 pueden conectarse por medio de tecnología de red de automatización industrial con capacidad en tiempo real 136 a actuadores 138 y sensores (no mostrados) del sistema de automatización industrial 130. Como también se analizó anteriormente en la sección 3, los nodos de computación en la nube 112 están configurados para ejecutar software de computación en la nube para configurar los controladores del sistema de automatización industrial 130, por medio de la red 120, realizando procedimientos como los analizados anteriormente y a continuación con referencia a la figura 4.

La figura 2 muestra un diagrama de bloques funcionales que ilustra otros aspectos de la arquitectura del sistema, las funciones y el funcionamiento de un sistema de computación en la nube 110 de acuerdo con algunos aspectos de la presente divulgación. En particular, la figura 2 ilustra aspectos clave de la arquitectura de software del sistema de computación en la nube 110. Como se analiza en la sección 3 anterior, el sistema de computación en la nube 110 está configurado para generar y operar una o más representaciones en tiempo de compilación 210 para cada controlador 132 (por ejemplo, en forma conjunta o como instancias separadas) que se configurarán por medio de la red 120. Como también se analizó anteriormente, las representaciones en tiempo de compilación 210 comprenden o utilizan una capa de persistencia 212 y una capa de interfaz de programación 214. La capa de persistencia 212 está configurada para contener configuraciones de controlador y elementos de instrucciones de control de programas de controlador, en la que los elementos de control individuales se representan como objetos de lógica de control separados. Además, los elementos individuales de la capa de persistencia 212 se pueden configurar individualmente en base a las instrucciones de programación y configuración recibidas por medio de una interfaz de programación 250 mediante la capa de interfaz de programación 214.

5 Por ejemplo, el sistema de computación en la nube 110 está configurado para recibir por medio de una interfaz de programación 250 instrucciones de programación y configuración para modificar los programas del controlador y/o las configuraciones del controlador mantenidas por la capa de persistencia 212 como se analizó anteriormente, así como para generar el programa de controlador modificado y la configuración de controlador modificada en base a las instrucciones recibidas.

10 Por ejemplo, las instrucciones de programación y configuración 242 pueden recibirse por medio de una API 250 desde un segundo sistema de computación en la nube 240, tal como un sistema PLM, de ejecución de fabricación, CRM, SCM o ERP.

15 El sistema de computación en la nube 110 puede configurarse, además, para alojar un servicio de entorno de desarrollo integrado (IDE) 230 que puede proporcionar instrucciones de programa de controlador 232 (por ejemplo, en forma de bloques de elementos de lógica de control) y/o configuraciones de controlador como se analizó anteriormente.

20 Como se analizó anteriormente, el programa de controlador modificado puede entonces proporcionarse a un servicio de compilador 220 (conectado operativamente a, o integrado con el servicio PLC Twin 210) alojado por el sistema de computación en la nube 110, generando un ejecutable en tiempo de ejecución que luego se proporciona, por medio de una red 120, a uno o más PLC basados en hardware 132 o servicios de PLC virtuales 132, alojados por un sistema de computación frontera (ver la figura 1 y figura 3), que controlan un subsistema 138 (por ejemplo, sensores y/o actuadores) del sistema de automatización industrial 130. Un servicio de compilador 220 de este tipo puede implementarse como un servicio de computación en la nube externo al servicio PLC Twin 210 o integrarse en él. El servicio de compilador 220 también puede comprender varios módulos de compilador 222, 224 diferentes para compilar diferentes tipos de programas de controlador (por ejemplo, específicos del proveedor).

25 Por ejemplo, el servicio de compilador 220 y/o el servicio PLC Twin 210 pueden implementarse en/por medio de un entorno de virtualización configurable, tal como un contenedor o un microservicio que especifique, entre otros, políticas de acceso de seguridad y/o parámetros de utilización de recursos del sistema, así como comprender bibliotecas de software necesarias para la ejecución aislada del respectivo servicio por diferentes tipos de equipos de computación en la nube (virtuales).

30 Como se analizó anteriormente, el sistema de computación en la nube ilustrado en la figura 2 está así configurado para llevar a cabo los diversos procedimientos para configurar controladores industriales analizados anteriormente y con referencia a la figura 4 a continuación.

35 La figura 3 muestra un diagrama de bloques funcionales que ilustra otros aspectos de la arquitectura del sistema, las funciones y el funcionamiento de un sistema de computación en la nube 110 de acuerdo con otros aspectos de la presente divulgación. En particular, la figura 3 ilustra aspectos clave de la arquitectura de software de un sistema de computación en la nube 110 que está configurado para permitir la (re)configuración de controlador de bucle completamente cerrado como se analiza con más detalle en la sección 3 anterior.

40 Además de los elementos, rasgos característicos y funciones analizados con referencia a la figura 2, la figura 3 también ilustra sistemas de automatización industrial 130 que involucran un sistema de computación frontera 310 conectado operativamente, por ejemplo, por medio de una red 120, al sistema de computación en la nube 110. El sistema de computación frontera 310 puede comprender un módulo/servicio de gestión de controlador 312 y puede estar configurado para ejecutar software de computación frontera para implementar/alocar controladores virtuales 314 (por ejemplo, PLC virtuales).

45 En dichas configuraciones, la configuración de controlador obtenida y utilizada para generar la representación en tiempo de compilación correspondiente 210 también puede especificar un entorno de virtualización para ejecutar el software de computación frontera que implementa el controlador virtual 314, así como una configuración de red de dispositivos de E/S físicos 332 conectados a un subsistema 138 del sistema de automatización industrial 130 que será controlado por el controlador virtual 314.

50 El sistema de computación en la nube 110, por ejemplo, por medio del módulo/servicio de gestión del controlador 312, puede configurarse para crear una instancia del controlador virtual 314, por medio de la red 120, en una máquina anfitriona del sistema de computación frontera 310 administrada por un hipervisor en tiempo real para permitir un control determinista en tiempo real del subsistema 138 correspondiente.

55 El sistema de computación frontera 310 también puede comprender/alocar un módulo/servicio de gestión de sensores 316 configurado para recibir, preprocesar y enviar datos de sensor al sistema de computación en la nube 110, por medio de la red 120, como se analizó anteriormente en la sección 3.

60 En particular, el sistema de computación en la nube 110 puede configurarse para recibir, por medio de la red 120, datos de sensor asociados a un subsistema 138 del sistema de automatización industrial 130, controlado por el

controlador 132, 314 y para generar instrucciones de programación y configuración (por ejemplo, por medio de una API al IDE 230) para modificar el programa de controlador y/o la configuración de controlador, en base al análisis de los datos de sensor recibidos.

5 Los datos de sensor recibidos pueden, por ejemplo, almacenarse en un módulo de almacenamiento de datos 350 del sistema de computación en la nube 110 que también puede usarse para soportar o habilitar el funcionamiento de la capa de persistencia utilizada por las representaciones en tiempo de compilación 210.

10 El sistema de computación en la nube también puede alojar un servicio de procesamiento y visualización de datos de sensor 360 que puede implementar ciertas etapas de análisis de datos de sensor como se analizó anteriormente.

15 Por ejemplo, dicho servicio de procesamiento y visualización de datos de sensor 360 puede configurarse para obtener, en base a los datos de sensor recibidos, una métrica de calidad asociada al funcionamiento del subsistema 138 controlado por el controlador 132, 314 y para comparar la métrica de calidad con un requisito de funcionamiento para el sistema de automatización industrial 130, así como para generar (por ejemplo, directamente o por medio de una API al servicio IDE 230) instrucciones de programación y configuración, en base a la comparación de la métrica de calidad con el requisito de funcionamiento como se analiza con más detalle en la sección 3 anterior.

20 El sistema de computación en la nube 110, además, puede comprender/alojar un módulo/servicio de inteligencia artificial (IA) 370 que puede estar conectado operativamente (por ejemplo, por medio de API 390) al módulo de almacenamiento de datos 350 y/o a otros sistemas de software basados en la nube 242, 344, 346 como se analizó anteriormente. Además, el servicio de IA 370 puede interactuar con un motor de experimentación 380 alojado por el sistema de computación en la nube 110, que a su vez puede interactuar con el servicio PLC Twin 210 (por
25 ejemplo, directamente o por medio de una API al servicio IDE 230).

De esta manera, el sistema de computación en la nube 110 puede configurarse para la optimización automática basada en IA del sistema de automatización industrial 130 como se analizó con más detalle anteriormente. Específicamente, el sistema de computación en la nube 110 puede configurarse para llevar a cabo un
30 procedimiento para optimizar el rendimiento de un sistema de automatización industrial como se divulga en el presente documento y con referencia a la figura 5.

La figura 4 ilustra un procedimiento ejemplar para configurar un controlador de un sistema de automatización industrial por medio de una red (ver las figuras 1 a 3). El proceso comienza en la etapa 410, donde un sistema de
35 computación en la nube obtiene una configuración de controlador y un programa de controlador. En la etapa 420, el sistema de computación en la nube genera, en base a la configuración de controlador obtenida y el programa de controlador obtenido, una representación en tiempo de compilación del controlador que permite la modificación del programa de controlador y la configuración de controlador (por ejemplo, por medio de interfaces de programación o un IDE basado en GUI). A continuación, en la etapa 430 se genera un programa de controlador
40 modificado y una configuración de controlador modificada, usando la representación en tiempo de compilación del controlador, en la que el programa de controlador define el funcionamiento en tiempo de ejecución del controlador que controla un subsistema del sistema de automatización industrial (como se analiza con más detalle anteriormente en la sección 3 y con referencia a las Fig. 1 a 3).

45 Luego, en la etapa 440, el sistema de computación en la nube proporciona, por medio de la red, el programa de controlador modificado y la configuración de controlador modificada al controlador del sistema de automatización industrial. En la sección 3 anterior se analizan en detalle otras implementaciones de dicho procedimiento.

De acuerdo con otro aspecto de la presente divulgación, la figura 5 ilustra un procedimiento para optimizar el
50 rendimiento de un sistema de automatización industrial, que comprende las siguientes etapas:

Generación 510, por un sistema de computación en la nube (para detalles de implementación, consulte las Fig. 1 a 3), de un programa de controlador modificado y/o una configuración de controlador modificada para un controlador que controla un subsistema del sistema de automatización industrial, en el que el programa de
55 controlador define el funcionamiento en tiempo de ejecución del controlador y, proporcionar 520, por el sistema de computación en la nube por medio de una red, el programa de controlador modificado y/o la configuración de controlador modificada al controlador para modificar el funcionamiento en tiempo de ejecución del controlador.

60 El procedimiento ilustrado comprende, además, recibir 530, en el sistema de computación en la nube por medio de la red, datos de sensor para el sistema de automatización industrial y estimar 540, mediante el sistema de computación en la nube y en base a los datos de sensor recibidos, un cambio en el rendimiento del sistema de automatización industrial causado por el funcionamiento de tiempo de ejecución modificado del controlador y optimizar 540, mediante el sistema de computación en la nube, el rendimiento del sistema de automatización industrial en base al cambio estimado de rendimiento.

65 En la sección 3 anterior se analizan con más detalle otras implementaciones de dicho procedimiento.

REIVINDICACIONES

1. Un procedimiento para configurar un controlador de un sistema de automatización industrial por medio de una red, comprendiendo el procedimiento:

5

obtener (410), por un sistema de computación en la nube, una configuración de controlador y un programa de controlador;

10

generar (420), por el sistema de computación en la nube, en base a la configuración de controlador obtenida y al programa de controlador obtenido, una representación en tiempo de compilación (210) del controlador que permite la modificación del programa de controlador y de la configuración de controlador en tiempo de compilación que comprende un tiempo en o antes de la generación de código de máquina en tiempo de ejecución para el controlador a partir de un código fuente correspondiente, y un tiempo durante el cual el código fuente se genera extrayendo el código fuente de un archivo de configuración o escribiendo el código fuente por medio de un lenguaje de programación o por medio de un entorno de desarrollo de código fuente basado en la nube; y

15

generar (430), por el sistema de computación en la nube, un programa de controlador modificado y una configuración de controlador modificada, usando la representación en tiempo de compilación (210) del controlador, en el que el programa de controlador define el funcionamiento en tiempo de ejecución del controlador que controla un subsistema del sistema de automatización industrial; y

20

proporcionar (440), por el sistema de computación en la nube por medio de la red, el programa de controlador modificado y la configuración de controlador modificada al controlador del sistema de automatización industrial;

25

en el que la configuración de controlador obtenida comprende, además, uno o más de: información sobre el tipo y la capacidad del controlador, configuración de red del controlador, información de interfaz para unidades electromecánicas y configuración de interfaz para dispositivos de E/S conectados al controlador por medio de la integración en panel posterior de sistemas de bus de campo industriales,

30

en el que la representación en tiempo de compilación (210) del controlador comprende una capa de persistencia que contiene configuraciones del controlador y elementos de instrucciones de control de programas del controlador, en el que los elementos de control individuales se representan como objetos de lógica de control separados, y en el que los elementos individuales de la capa de persistencia (212) se pueden configurar individualmente en base a instrucciones de programación y configuración recibidas por medio de una interfaz de programación (250).

35

2. El procedimiento de la reivindicación 1, en el que partes de la configuración de controlador y/o del programa de controlador se obtienen, en el sistema de computación en la nube, desde el controlador, desde un archivo de configuración de controlador y/o desde una biblioteca de software o un repositorio de código fuente que reside dentro o fuera del sistema de computación en la nube.

40

3. El procedimiento de la reivindicación 1 o 2, en el que el programa de controlador se obtiene, en el sistema de computación en la nube, a partir de un servicio de entorno de desarrollo integrado alojado por el sistema de computación en la nube.

45

4. El procedimiento de cualquiera de las reivindicaciones 1 a 3, que comprende además:

50

recibir, en el sistema de computación en la nube, por medio de la interfaz de programación, instrucciones de programación y configuración para modificar el programa de controlador obtenido y/o la configuración de controlador obtenida; y

55

generar el programa de controlador modificado y/o la configuración de controlador modificada en base a las instrucciones recibidas.

5. El procedimiento de la reivindicación 4, en el que la interfaz de programación es una interfaz de programación de aplicaciones, API, y en el que las instrucciones de programación y configuración se reciben por medio de la API desde un segundo sistema de computación en la nube; y en el que las instrucciones de programación y configuración están configuradas para modificar el programa de controlador y/o la configuración de controlador en base a los requisitos de gestión del ciclo de vida del producto, PLM, requisitos de ejecución de fabricación, requisitos de gestión de relaciones con el cliente, CRM, requisitos de gestión de la cadena de suministro, SCM, y/o requisitos de planificación de recursos empresariales, ERP.

60

65

6. El procedimiento de cualquiera de las reivindicaciones 4 a 5, que comprende además:

- recibir, en el sistema de computación en la nube, por medio de la red, datos de sensor asociados a un subsistema del sistema de automatización industrial, controlado por el controlador; y
- 5 generar, mediante el sistema de computación en la nube, las instrucciones de programación y configuración para modificar el programa de controlador y/o la configuración de controlador, en base al análisis de los datos de sensor recibidos.
- 10 7. El procedimiento de la reivindicación 6, que comprende además
- obtener, mediante el sistema de computación en la nube y en base a los datos de sensor recibidos, una métrica de calidad asociada al funcionamiento del subsistema controlado por el controlador;
- 15 comparar, mediante el sistema de computación en la nube, la métrica de calidad con un requisito de funcionamiento para el sistema de automatización industrial; y
- generar, mediante el sistema de computación en la nube, las instrucciones de programación y configuración, en base a la comparación de la métrica de calidad con el requisito de funcionamiento.
- 20 8. El procedimiento de la reivindicación 7, que comprende además:
- clasificar, mediante el sistema de computación en la nube, la calidad del producto en base a los datos de sensor recibidos y comparar una frecuencia o porcentaje de defectos de producto con un requisito de funcionamiento; y/o
- 25 determinar una causa probable y/o una modificación adecuada del programa de controlador para que coincida con el requisito de funcionamiento en base a los datos de sensor recibidos.
- 30 9. El procedimiento de cualquiera de las reivindicaciones 1 a 8,
- en el que el controlador es un controlador virtual, implementado por un software de computación frontera ejecutado por un sistema de computación frontera conectado operativamente al sistema de computación en la nube y al sistema de automatización industrial; y
- 35 en el que la configuración de controlador obtenida especifica un entorno de virtualización para ejecutar el software de computación frontera implementando el controlador virtual y comprende una configuración de red de un dispositivo físico de E/S conectado a un subsistema del sistema de automatización industrial que será controlado por el controlador virtual,
- 40 que comprende además, preferentemente, crear una instancia del controlador virtual, mediante el sistema de computación en la nube por medio de la red, en una máquina anfitriona del sistema de computación frontera gestionada por un hipervisor en tiempo real.
- 45 10. El procedimiento de la reivindicación 9, que comprende además:
- recibir, en el sistema de computación en la nube, datos de supervisión del sistema de computación frontera que caracterizan el rendimiento del controlador virtual.
- 50 11. El procedimiento de cualquiera de las reivindicaciones precedentes 1 a 10, que comprende además: generar el programa de controlador obtenido y/o las instrucciones de programación y configuración usando un entorno de desarrollo integrado del sistema de computación en la nube.
- 55 12. Un sistema de computación en la nube para configurar uno o más controladores de un sistema de automatización industrial, comprendiendo el sistema de computación en la nube:
- uno o más nodos de computación en la nube, proporcionando cada uno de los cuales recursos de procesamiento, memoria y red para la ejecución de software de computación en la nube;
- 60 en el que los nodos de computación en la nube están configurados para recibir y transmitir, por medio de una red, datos desde y hacia los controladores de los sistemas de automatización industrial y, opcionalmente, desde uno o más sensores que supervisan el sistema de automatización industrial;
- 65 en el que los uno o más nodos de computación en la nube están configurados para ejecutar software de computación en la nube para configurar los controladores, por medio de la red, realizando el procedimiento de una de las reivindicaciones precedentes 1 a 11.

13. Producto de programa informático, que comprende instrucciones para llevar a cabo el procedimiento de cualquiera de las reivindicaciones precedentes 1 a 11, cuando se ejecuta por un sistema de computación en la nube.
- 5 14. Un sistema de control de automatización industrial distribuido, que comprende:
- 10 un primer sistema de computación en la nube como se especifica en la reivindicación 12;
- uno o más controladores de un sistema de automatización industrial conectados al primer sistema de computación en la nube por medio de una red; y
- un segundo sistema de computación en la nube que se comunica con el primer sistema de computación en la nube por medio de una interfaz de programación.

Figuras

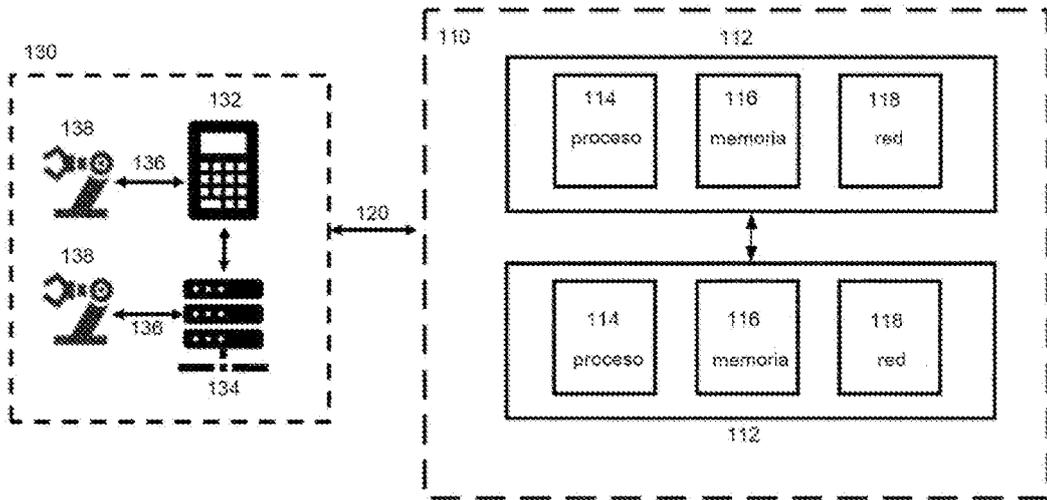


Figura 1

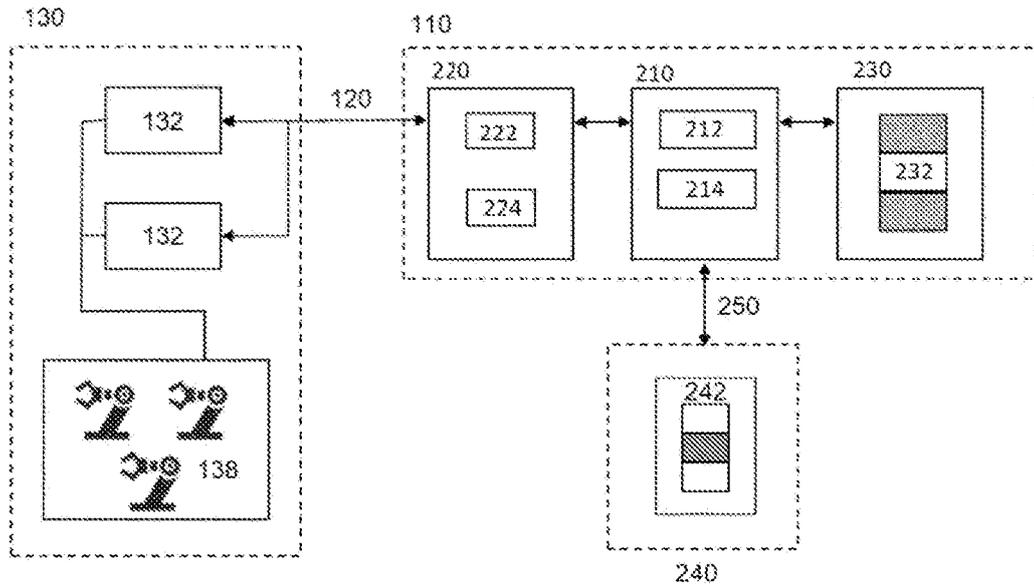


Figura 2

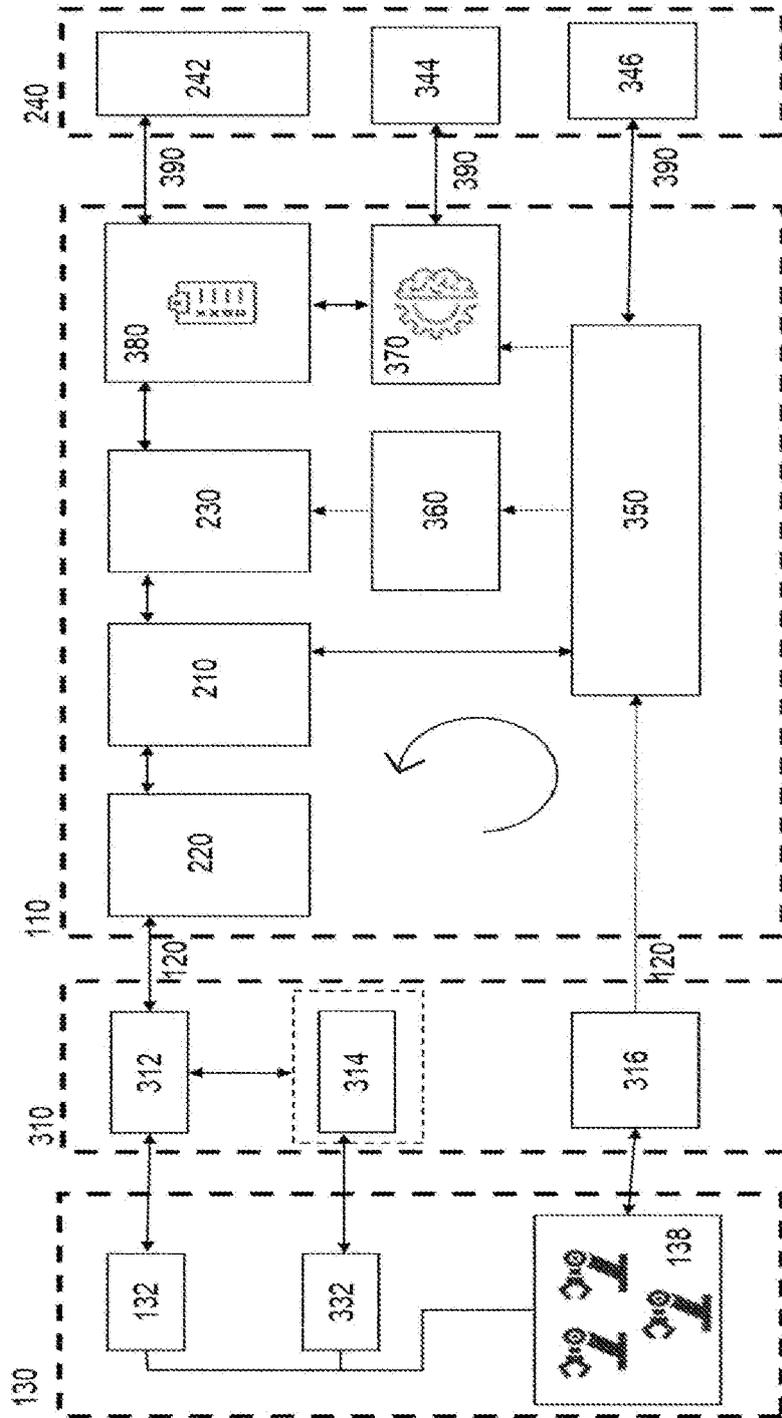


Figura 3

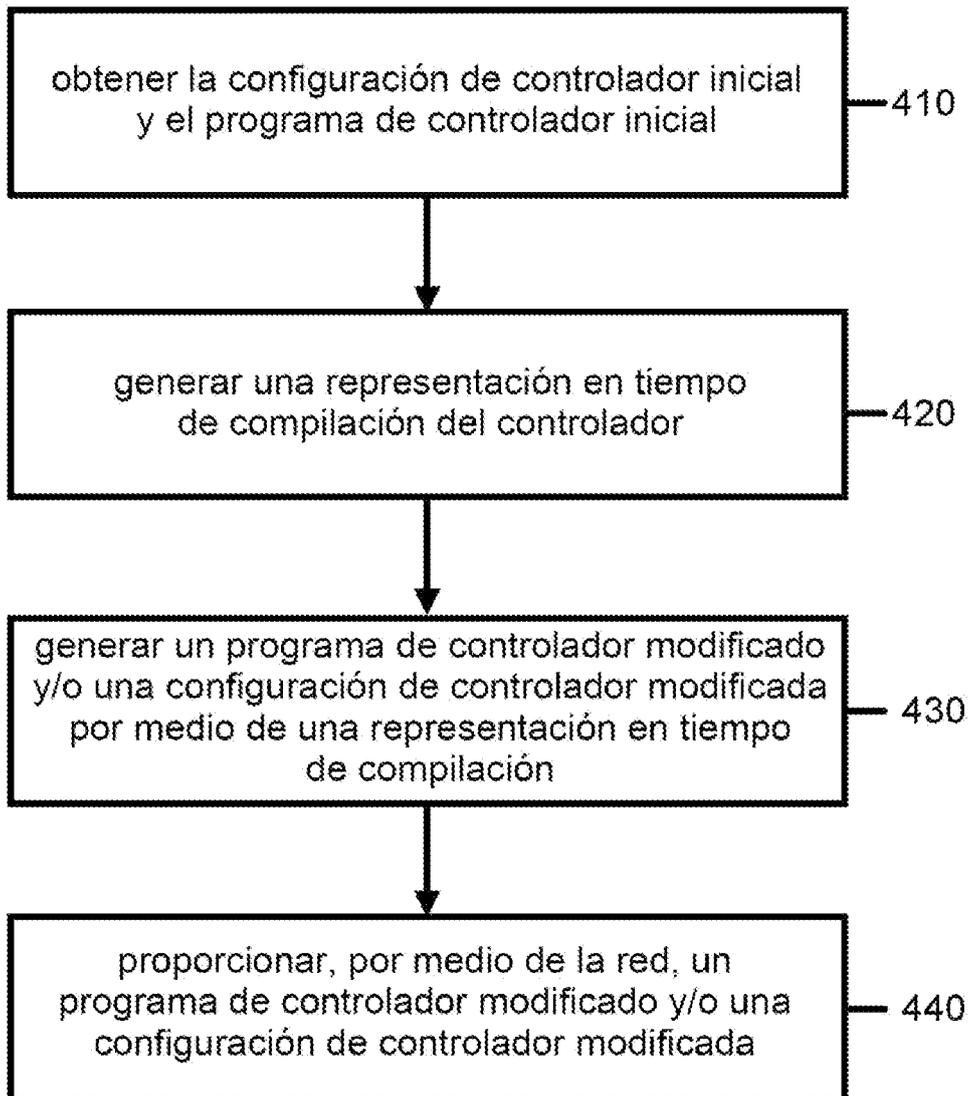


Figura 4

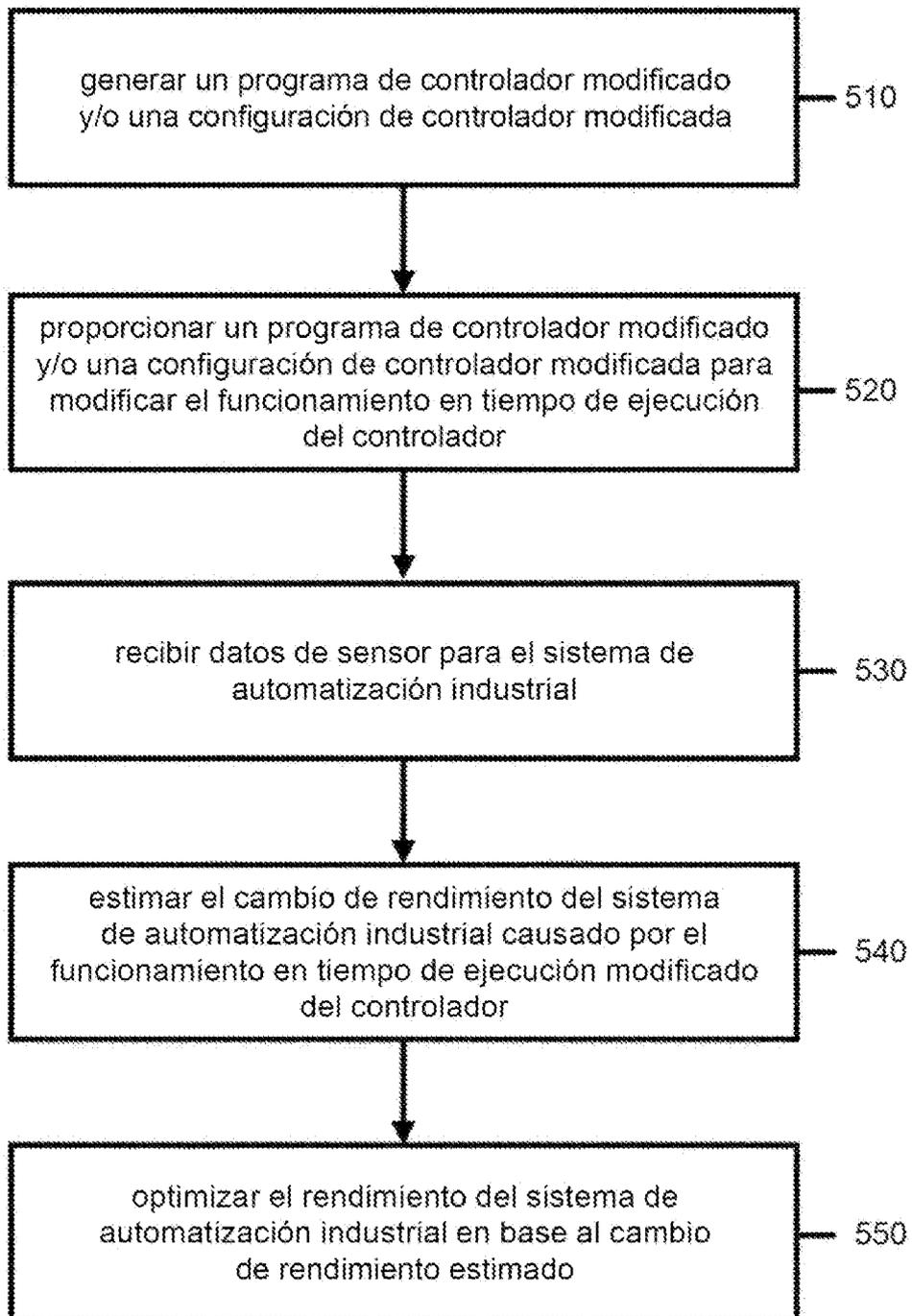


Figura 5