

19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 913 636**

51 Int. Cl.:

G05B 19/414 (2006.01)

G06F 9/48 (2006.01)

G05B 19/05 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

86 Fecha de presentación y número de la solicitud internacional: **10.11.2017** **PCT/EP2017/078931**

87 Fecha y número de publicación internacional: **17.05.2018** **WO18087314**

96 Fecha de presentación y número de la solicitud europea: **10.11.2017** **E 17794752 (0)**

97 Fecha y número de publicación de la concesión europea: **23.03.2022** **EP 3538960**

54 Título: **Control de secuencia de módulos de programa**

30 Prioridad:

10.11.2016 LU 93299

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

03.06.2022

73 Titular/es:

PHOENIX CONTACT GMBH & CO. KG (100.0%)
Flachsmarktstrasse 8
32825 Blomberg, DE

72 Inventor/es:

HEUTGER, HENNING;
KLEINOW, ROLF;
WEICHELT, ANDREAS;
HAUSMANN, STEFAN y
FRIEBEN, JENS

74 Agente/Representante:

LOZANO GANDIA, José

ES 2 913 636 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

Control de secuencia de módulos de programa

- 5 La presente invención se encuentra en el campo de la tecnología de automatización, en particular de la tecnología de automatización industrial, y se refiere a un procedimiento para el control de secuencia de módulos de programa. Además, la presente invención se refiere a un sistema operativo en tiempo real, una unidad de control, un dispositivo de control y un producto de programa informático.
- 10 En la tecnología de automatización se puede prever un cambio, que se realiza desde el uso de lenguajes de programación clásicos al uso de lenguajes de alto nivel.
- Hasta ahora, por ejemplo, la norma IEC 61131 regula la sintaxis y la semántica de los lenguajes de programación para controladores programables. Aquí se propone un conjunto unificado de lenguajes de programación para controladores programables. Este conjunto presenta con dos lenguajes de texto (lista de instrucciones y texto estructurado) y dos lenguajes gráficos (diagrama en escalera y diagrama de bloques de funciones). Además, en la norma IEC 61131 se definen las características que simplifican la comunicación entre los controladores de programa y otros componentes de un sistema automatizado.
- 15
- 20 Mediante temas actuales tal como, por ejemplo, la Industria 4,0, el Internet de las cosas o la agenda CPS de la VDI se muestra que en el futuro las instalaciones automatizadas se caracterizarán mediante una alta proporción de integración en red y software. A este respecto, habrá un cambio de la programación clásica IEC 61131-3 a modernos lenguajes de alto nivel. Hasta el año 2018, alrededor del 70 % del código utilizado debe estar fuera de los lenguajes definidos de acuerdo con IEC. A este respecto, los requisitos existentes, tal como, por ejemplo, la capacidad de funcionamiento en tiempo real y la consistencia, se complementan con una mayor flexibilidad, más seguridad y mejor capacidad de mantenimiento.
- 25
- En consecuencia también resultan cambios técnicos en el futuro en la producción de controles lógicos programables y/o entornos de tiempo de ejecución.
- 30
- El documento EP 3 076 297 A2 describe un control que está conformado para ejecutar programas de control en paralelo en ciclos más cortos utilizando un procesador multinúcleo, en el que se evitan las colisiones de comunicación porque la secuencia temporal de la comunicación de los programas de control se realiza dependiendo de las prioridades que se asignan a los programas de control.
- 35
- En el documento WO 2009/037198 A1 se describe un sistema de control para una instalación técnica y un procedimiento para operar un sistema de control, en el que está previsto un entorno de secuencia, en el que se pueden utilizar diferentes administradores de tareas o programadores, en particular parametrizables diferentes, en el que está previsto usar, alejándose del principio establecido del uso del tiempo real como un parámetro de desencadenamiento firmemente anclado en el módulo de control, un parámetro de desencadenamiento editable o modificable por el usuario para el control de secuencia a través del módulo de control.
- 40
- El documento WO 2013/010159 A1 describe un procedimiento para la ejecución en paralelo de una aplicación de control en un procesador multinúcleo, en el que se lleva a cabo un análisis de dependencia de datos para identificar tareas que se pueden ejecutar en paralelo (D3, Resumen).
- 45
- El documento EP 2 504 738 B1 describe un procedimiento de control y un procedimiento para ejecutar varios programas en varios dispositivos de procesamiento que trabajan en paralelo. En este caso, se asigna un generador de señales de tiempo a cada uno de varios dispositivos de procesamiento que trabajan en paralelo, y durante su secuencia en el dispositivo de procesamiento asignado se ejecuta un procedimiento de control para la ejecución de programas.
- 50
- No obstante, el uso de un generador de señales de tiempo tiene la desventaja de que se puede reducir la velocidad de procesamiento de datos en un sistema operativo en tiempo real.
- 55
- Un objeto de la presente invención es especificar una solución para mejorar controles de secuencia para sistemas operativos en tiempo real.
- Este objetivo se consigue de acuerdo con la invención mediante las características de la reivindicación 1. Además, el objetivo se consigue mediante un aparato de control de acuerdo con la reivindicación 12, un producto de programa informático de acuerdo con la reivindicación 13 y un soporte de datos de acuerdo con la reivindicación 14. Otras configuraciones resultan de las reivindicaciones dependientes.
- 60
- En el procedimiento de acuerdo con la invención para el control de secuencia de módulos de programa se prevé una facilitación un equipamiento de control y una facilitación de un archivo de descripción que presenta la configuración del archivo de descripción para controlar una secuencia de programa. Además, el procedimiento comprende una facilitación de varios módulos de programa que se pueden ejecutar por un sistema operativo en tiempo real, en el que los módulos
- 65

de programa se crean usando uno o diferentes lenguajes de programación. Además, el procedimiento presenta un desencadenamiento de módulos de programa seleccionados mediante la ejecución de especificaciones en el archivo de descripción mediante el equipamiento de control.

5 El equipamiento de control se proporciona como parte del firmware de un dispositivo de control, en el que el dispositivo de control está conformado en particular como un aparato de control para controlar un sistema de automatización. Por ejemplo, el dispositivo de control sirve para leer datos de sensores y para controlar actuadores dependiendo de los datos de sensores. Con esta finalidad se ejecuta una aplicación de control en el dispositivo de control. La ejecución de la aplicación de control, es decir, la aplicación global, comprende la ejecución de varios módulos de programa. El control de
10 secuencia de los módulos de programa se realiza mediante el equipamiento de control, que desencadena los módulos de programa seleccionados con esta finalidad mediante la ejecución de especificaciones en el archivo de descripción.

El desencadenamiento de un módulo de programa designa la ejecución o el inicio de la ejecución de un módulo de programa, en el que el módulo de programa se proporciona en particular como código de programa depositado en una memoria, ejecutable por un sistema operativo en tiempo real. Para desencadenar un módulo de programa, por el
15 equipamiento de control se llama preferentemente una función correspondiente del sistema operativo. Qué módulos de programa en qué orden y dependiendo de qué evento o en qué intervalos de tiempo se desencadenan por el equipamiento de control se predetermina mediante los contenidos del archivo de descripción. Con esta finalidad, el archivo de descripción comprende, por ejemplo, información para identificar un módulo de programa e información asignado que, por
20 ejemplo, predetermina una ejecución cíclica y un tiempo de ciclo asignado. Las tareas a las que están asignadas los módulos de programa también se pueden definir ventajosamente en el archivo de descripción.

El archivo de descripción se proporciona en particular en forma de archivo de texto en una memoria de un dispositivo de control y se lee por el equipamiento de control como parte del firmware del dispositivo de control en la puesta en
25 funcionamiento o encendido del dispositivo de control. La secuencia de los módulos de programa predeterminada en el archivo de descripción se controla por el equipamiento de control mediante llamada a las funciones apropiadas del sistema operativo.

El equipamiento de control puede estar configurado como un administrador de ejecución y sincronización o un *Execution and Synchronisation Manager* (ESM).
30

Mediante el uso de un equipamiento de control semejante se posibilita una manipulación o ejecución equivalentes de programas de diferentes dominios de programación.

35 Por ejemplo, se pueden realizar módulos de programa que estén programados respectivamente al menos en uno de los lenguajes de programación de acuerdo con la norma IEC 61131, en Matlab/Simulink, C++, C# o JAVA o Python. Igualmente son posibles otros lenguajes de programación. De acuerdo con la norma IEC 61131 (edición 3.0, 2013-02) se especifican dos lenguajes de texto (textual languages), a saber, lista de instrucciones (IL) y texto estructurado (ST) y dos lenguajes gráficos (graphical languages), a saber, diagrama en escalera (LD) y diagrama de bloques de funciones (FBD).
40 Estos lenguajes de programación se pueden aplicar junto con lenguajes de programación de alto nivel en un control de secuencia.

El equipamiento de control permite una configuración sencilla de tareas similar a IEC 61131-3, que permite la integración de instancias de programa de diferentes lenguajes de programación. Para ello, el equipamiento de control utiliza una serie de procesos y subprocesos del sistema operativo para posibilitar una integración de alto rendimiento en el sistema operativo subyacente en tiempo real subyacente.
45

Por lo tanto, se puede usar un único lenguaje de programación para todos los módulos de programa. También es posible que se usen diferentes lenguajes de programación para dos o más módulos de programa.
50

Al desencadenar los módulos del programa, el archivo de descripción, que también se puede designar como archivo de configuración, ya no se direcciona. Una respuesta semejante tomaría demasiado tiempo y ralentizaría el procesamiento. Más bien, el/los archivo(s) de configuración se leen cuando en el inicio del sistema y solo se ejecutan durante la fase de procesamiento. En este caso, se prevé una ejecución de la información del archivo de descripción mediante el equipamiento de control. Por tanto, las especificaciones para la ejecución de los módulos del programa resultan mediante el contenido del archivo de descripción. Estas especificaciones se proporcionan mediante una configuración del archivo de descripción. En consecuencia, los módulos del programa se desencadenan ejecutando la configuración definida en el archivo de descripción.
55

60 La configuración definida en el archivo de descripción comprende, por ejemplo, respectivamente una identificación para los módulos de programa seleccionados, por ejemplo, un nombre bajo el cual está almacenado el módulo de programa, así como información asignada para ejecutar el módulo de programa identificado respectivamente, es decir, por ejemplo, si el módulo de programa se debe ejecutar de forma cíclica o controlada por eventos. El equipamiento de control está conformado para evaluar la información contenida en el archivo de descripción y ejecuta los módulos de programa seleccionados dependiendo de la información de configuración especificada en el archivo de descripción, en el que la
65

identificación contenida respectivamente en el archivo de descripción es adecuada para desencadenar, es decir, ejecutar el módulo de programa respectivo con la ayuda de una función del sistema operativo.

Los módulos de programa se controlan, por ejemplo, con la ayuda de una interfaz de programación, en particular con una API (API = interfaz de programación de aplicaciones). El archivo de descripción, que se puede proporcionar, por ejemplo, como un archivo XML (XML = archivo de lenguaje de marcado extensible), sirve para configurar cuándo se direccionan los programas y, por lo tanto, se ejecutan. Con la ayuda del equipamiento de control se pueden ejecutar módulos de programa y registrarse información de estado sobre la secuencia del programa, por ejemplo, tiempos de ejecución de tareas.

El propio equipamiento de control se puede inicializar y controlar por un administrador de PLC (PLC = controlador lógico programable). Esto pone el control en RUN/STOP, por ejemplo, y lleva a cabo reinicios. Durante el funcionamiento, el equipamiento de control se dispara cíclicamente por eventos (software o hardware) para procesar una lógica de ejecución.

También se pueden utilizar una pluralidad de archivos de descripción o archivos de configuración para el control de secuencia. Los otros archivos de descripción pueden contener una configuración de tareas adicionales y sus módulos de programa asignados. Por ejemplo, un archivo de descripción, por ejemplo, en forma de archivo XML, podría describir una tarea principal (Task-Main), otro archivo de descripción podría describir una tarea de control (Task-Ctrl) y otro archivo de descripción podría describir una tarea de sincronización (Task-Sync). Los archivos de configuración se pueden fusionar fácilmente en una configuración si de ello no resulta ninguna contradicción. Por esta razón, el equipamiento de control o un equipamiento de control en la función de un maestro (maestro ESM) debe conocer y leer todas las configuraciones.

En un modo de realización preferente, el procedimiento comprende el desencadenamiento múltiple de al menos uno de los módulos de programa por parte del equipamiento de control.

Para ello no es necesario direccionar el archivo de descripción, de modo que los tiempos de procesamiento se pueden mantener cortos. También se pueden asistir igualmente otros tipos de tareas o tipos de task, como eventos y tarea inactiva. Ventajosamente, el archivo de descripción o la pluralidad de archivos de descripción solo se leen para la configuración inicial, pero no durante la ejecución.

En otro modo de realización preferente del procedimiento está previsto que el equipamiento de control esté conformado para desencadenar al menos algunos de los módulos de programa, de tal manera que se puedan ejecutar en diferentes procesos gestionados por el sistema operativo en tiempo real.

De esta manera, es posible usar solo los módulos de programa que se requieren y también ejecutar procesos en paralelo para utilizar los recursos de tiempo de forma eficiente.

En un perfeccionamiento del procedimiento está previsto que el sistema operativo en tiempo real esté conformado para ejecutar los procesos y/o módulos de programa dependiendo de prioridades predeterminadas.

Mediante el establecimiento de prioridades, las tareas de control se pueden procesar de acuerdo con su importancia o urgencia.

Además, en un modo de realización del procedimiento, mediante el sistema operativo en tiempo real se puede prever un registro y evaluación del control de secuencia.

Para el registro y la evaluación también se pueden registrar puntos de marcado incorporados adicionalmente y se pueden generar estadísticas.

En otro modo de realización, el procedimiento puede prever una ampliación del control de secuencia mediante un módulo de programa adicional sin que se lleve a cabo un reinicio de un sistema de control.

Un sistema de control designa un control de secuencia en el que los módulos del programa típicamente se pueden cargar, iniciar, depurar, detener y descargar. El sistema de control se basa en el sistema operativo en tiempo real.

Es ventajoso que sea posible una ampliación del sistema de control sin que se tenga que interrumpir el sistema de control para un reinicio. De esta manera, la operación se puede seguir continuamente.

En otro modo de realización, el procedimiento puede presentar una división del control de secuencia en distintos procesos.

Una distribución semejante puede aprovechar mejor las ventajas de los diferentes procesos del sistema operativo en tiempo real, por ejemplo, los mecanismos de protección de la memoria.

Además, en otro modo de realización del procedimiento se puede prever una creación de una evaluación estadística usando información de perfilado.

En este caso, se pueden incorporar marcas o marcados en el equipamiento de control para generar la información de perfilado. Bajo la información de perfilado se entiende información para analizar el comportamiento del tiempo de ejecución del software. Por lo tanto, los datos estadísticos se pueden evaluar para mejorar el control del proceso y utilizarse con finalidades estadísticas.

En otro modo de realización, el procedimiento comprende una incorporación de al menos una marca de registro en un módulo de programa para la facilitación de un diagnóstico de aplicación.

Una marca de registro se puede entender como una ubicación en un módulo de programa que posibilita obtener información sobre una parte determinada de un módulo de programa en relación con su ejecución. En base a esta información se puede crear a continuación un diagnóstico con el fin de evaluar la aplicación ejecutada. En este caso, se puede tratar de una sola aplicación o la aplicación global de la tarea de control.

En otro modo de realización, el procedimiento comprende un uso del equipamiento de control como maestro y la facilitación de al menos otro equipamiento de control como esclavo, que está subordinado al equipamiento de control como maestro.

Entonces, el uso de un esclavo se puede utilizar ventajosamente cuando un módulo de programa se debe ejecutar en otro proceso. Si la aplicación global se ejecuta en un solo proceso, no se puede prescindir de un esclavo.

Las tareas o tasks son una unidad organizativa para ejecutar programas. Los procesos del OS y los subprocesos del OS (OS = *Operating System* o sistema operativo) se pueden utilizar para la implementación técnica de estas unidades de gestión organizativas. En este caso, es deseable posibilitar un procesamiento sincronizado de tareas a través de los límites de proceso del OS. En general, no es posible que el maestro cree subprocesos del OS en otros procesos del OS. Por esta razón, esto se puede asumir por un esclavo en cada proceso del OS involucrado. El maestro puede asumir la ejecución de los programas o módulos de programa. Si, por otro lado, solo se usa un proceso del OS, todos los programas y subprocesos se pueden ejecutar sin esclavos.

Por ejemplo, el maestro y el esclavo se pueden comunicar a través de la comunicación entre procesos si se ejecutan en el mismo dispositivo. La comunicación se implementa ventajosamente al nivel más bajo con funcionalidades básicas del sistema operativo.

Además, en otro modo de realización preferente, el procedimiento puede presentar una asignación de tareas individuales, definidas en el archivo de descripción en los módulos de programa mediante el equipamiento de control.

Por lo tanto, el equipamiento de control puede distribuir tareas individuales dentro del sistema de control para mejorar así, por ejemplo, la utilización del sistema de control. Una tarea o task o una pluralidad de las mismas puede ser un subconjunto de un proceso dentro de una tarea de control.

Un sistema en tiempo real a modo de ejemplo para el control de secuencia de módulos de programa, que comprende un equipamiento de control, presenta un archivo de descripción depositado en un dispositivo de almacenamiento, que define una secuencia de programa. Además, están previstos varios módulos de programa ejecutables por el sistema operativo en tiempo real, que se crean utilizando uno o diferentes lenguajes de programación, en los que el equipamiento de control está conformado para desencadenar los módulos de programa ejecutando especificaciones del archivo de descripción.

El sistema operativo en tiempo real (RTOS = *real-time operating system*) es un sistema operativo que es capaz de cumplir con los requisitos en tiempo real de las aplicaciones. En este caso, el cumplimiento de los tiempos de respuesta definidos en el campo de la tecnología de automatización es de particular importancia para evitar daños a una instalación instalada y controlada.

El equipamiento de control puede estar configurado en forma de un administrador de ejecución y sincronización o un *Execution and Synchronisation Manager* (ESM). El equipamiento de control puede priorizar tareas centrales de control, distribuir las a módulos de programa presentes y recibir información de los módulos de programa individuales y analizar esta información.

Además, el objetivo de la presente invención se logra mediante un dispositivo de control para el control de secuencia de módulos de programa de lenguajes de programación diferentes en un sistema operativo en tiempo real. Para ello, el dispositivo presenta un dispositivo de control de acuerdo con la invención, un procesador, una memoria y al menos una interfaz a un dispositivo de automatización.

En este caso, un dispositivo de control puede ser un aparato de control que está instalado en una instalación de automatización. En este caso, el equipamiento de control de acuerdo con la invención es parte del dispositivo de control de acuerdo con la invención. También es posible una distribución instanciadora o instanciación del equipamiento de control a varios núcleos de cálculo o núcleos de un procesador multinúcleo.

Además, el objetivo de la presente invención se consigue con un producto de programa informático. Para ello, el producto de programa informático presenta medios de código de programa para llevar a cabo el procedimiento de acuerdo con la invención, en el que el producto de programa informático se ejecuta en un sistema informático en un sistema operativo en tiempo real.

El producto de programa informático puede ser, por ejemplo, un soporte de datos que se pone a disposición en un sistema de automatización para llevar a cabo el procedimiento de acuerdo con la invención.

En conjunto, la tecnología de control propuesta ofrece una solución flexible que se puede adaptar dinámicamente a futuros desarrollos en el campo de la tecnología de automatización. Debido a su apertura, la arquitectura de control propuesta, en contraste con los sistemas cerrados hasta ahora, permite que las aplicaciones deseadas se implementen individualmente y aún se apliquen sobre una base estable. Esto se puede garantizar no solo para el grupo objetivo de programadores de lenguajes de alto nivel, sino también para instalaciones de automatización instaladas hasta ahora en el entorno de programación IEC 61131-3.

Algunos ejemplos de modo de realización de la invención se representan de forma puramente esquemática en los dibujos y se describen con más detalle a continuación. Muestra:

la fig. 1 un ejemplo de modo de realización para controlar una aplicación global en una instalación de automatización de acuerdo con la presente invención;

la fig. 2 un ejemplo de modo de realización de la presente invención de una estructura esquemática con un equipamiento de control como maestro y otros dos equipamientos de control como esclavos;

la fig. 3 un ejemplo de modo de realización de acuerdo con la presente invención de un control de una secuencia de programa mediante un equipamiento de control;

la fig. 4 un ejemplo de modo de realización de la presente invención de una estructura esquemática con un equipamiento de control.

La fig. 1 muestra un ejemplo de modo de realización para controlar una aplicación global 10 en una instalación de automatización de acuerdo con la presente invención.

En este contexto, bajo una aplicación global se entienden todos los programas o módulos de programa, tareas o tasks y ajustes de configuración que se requieren para el modo de realización en un aparato de control. A este respecto, no es relevante si el aparato de control o el control es el único o es uno de varios aparatos.

El procesamiento cíclico de programas o de módulos de programa es típico de la automatización. Típicamente no se da una retroalimentación al usuario o user en el marco de estos ciclos.

En la fig. 1 está representada una estructura esquemática de tres tareas 20, 30, 40 o tasks que ejecutan en conjunto cuatro programas o módulos de programa 21, 31, 32, 41 de los dominios IEC 61131-3, C++ y Matlab/Simulink de forma sincronizada. El módulo de programa 21 es un filtro CPP y está programado en el lenguaje de programación C++. El módulo de programa 31 es un programa de preparación IEC y está programado en uno de los lenguajes de programación de acuerdo con el estándar IEC. El módulo de programa 32 es un programa SIM-CTRL, es decir, un programa de control, y está programado en el lenguaje de programación Matlab/Simulink. El módulo de programa 41 es un programa de sincronización IEC, es decir, un programa de sincronización, y está programado en uno de los lenguajes de programación de acuerdo con el estándar IEC. En este contexto, los términos "programa" y "módulo de programa" se usan de forma intercambiable.

Después del final del procedimiento o el final del proceso del módulo de programa 30, la tarea 40 con el módulo de programa 41 se inicia motivada a través de una interfaz 33 a través de una de comunicación 34.

Las tareas 20, 30, 40 de la fig. 1 tienen diferentes prioridades para ejecutar la aplicación global 10, en la que la tarea 20 como tarea principal o *Main Task* presenta la prioridad más baja, la tarea 30 como tarea de control o *Ctrl-Task* presenta una prioridad media y la tarea 40 como tarea de evento o *Event-Task* con tareas de sincronización presenta la máxima prioridad. Además, la tarea 20 presenta un tiempo de ciclo o *Cycle Time* de 5 ms (ms = milisegundos) y la tarea 30 de 30 ms. La tarea 40 se ejecuta una vez y finaliza cuando se ha procesado el módulo de programa 41 o el programa en el módulo de programa (evento: *proc_end*).

La figura 2 muestra un ejemplo de realización de la presente invención de una estructura esquemática con un equipamiento de control 50 como maestro, un equipamiento de control 60 subordinado al maestro como primer esclavo y otro equipamiento de control 70 subordinado al maestro como segundo esclavo. En este caso, el componente maestro y los componentes esclavos 50, 60, 70 se utilizan para coordinar módulos de programa de diferentes dominios de programa en diferentes procesos. Además, en la fig. 2 están representados diferentes procesos del sistema operativo o procesos del OS 11, 12, 13, que presentan respectivamente subprocesos del OS 14, 15, 16, 17, 18.

El modo de proceder de acuerdo con la invención permite la composición de contenidos de tarea heterogéneos. A este respecto, la secuencia de procesamiento dentro de una tarea se puede seleccionar libremente, no predeterminada por el sistema, y es posible un intercambio de datos consistente entre instancias del programa.

La configuración del maestro 50 y sus tareas 20, 30, 40 se realiza mediante uno o varios archivos de descripción, aquí en forma de uno o varios archivos XML, que permiten la configuración en lugar de la programación. Por consiguiente, la función de una instalación se puede cambiar en tiempo de ejecución sin tener que compilar nuevamente las partes individuales de la aplicación. El archivo de descripción o la configuración basada en XML también le permite al usuario efectuar adaptaciones directamente en el control o utilizar sus propias herramientas y cadenas de herramientas externas.

A este respecto, la estructura del archivo de descripción, en particular en forma de un archivo XML, se puede predeterminar mediante archivos de esquema claramente definidos, que posibilita una comprobación rápida de la configuración. La especificación de tareas y las instancias de programa correspondientes en varios archivos de descripción o archivos de configuración XML permite dividir la aplicación total 10 en módulos funcionales que se pueden crear por diferentes grupos de desarrolladores o proveedores. Esto no solo facilita el desarrollo de un control de instalación complejo, sino también la ampliación gradual de una instalación en funcionamiento. Por tanto, la aplicación global 10 no debe estar disponible desde el principio, sino que se puede componer de submódulos.

Cada tarea 20, 30, 40 también se puede asignar a un núcleo de procesador, a un grupo o a todos los núcleos de procesador del control mediante la definición de parámetros en el archivo de descripción o archivo XML.

Un sistema flexible de componentes maestro y esclavo 50, 60, 70 también permite al usuario ejecutar programas a través de los límites del proceso. Este es un requisito previo ventajoso para la separación de programas de diferentes usuarios, que se ejecutan en áreas de memoria encapsuladas. Los mecanismos de protección de la memoria del sistema operativo en tiempo real subyacente pueden servir como base para esto. Por razones de seguridad es posible restringir el acceso a programas o módulos de programas correspondiente a los derechos de los usuarios.

En la fig. 2, el maestro 50 configura respectivamente un módulo de programa 21, 41 de un esclavo 60, 70 a través de conexiones activas 51, 52. En este caso, configurar se refiere a la facilitación de información de prioridad mediante el maestro 50. Además, las conexiones de comunicación 53, 54, 57, 58 están presentes en la fig. 2 para intercambiar comandos e información o retroalimentaciones entre los módulos de programa individuales 21, 31, 32, 41 y una tarea 20, 30, 40. En este caso, la información se envía y devuelve entre las instancias de maestro y esclavo o se recibe respectivamente. Los comandos son, por ejemplo, comandos de desencadenamiento ("desencadena") de la tarea 20, 40 al módulo de programa 21, 41. Se envían ("informan") retroalimentaciones, por ejemplo, desde los módulos de programa 21, 41 a las tareas 20, 40. Además, los comandos de desencadenamiento ("desencadena") también se envían desde la tarea 30 respectivamente a los respectivos módulos de programa 31, 32 a través de las conexiones de comunicación 55, 56, en la que la comunicación tiene lugar dentro de un subproceso OS.

Con el modo de proceder propuesto es posible transferir las tareas conocidas de la estandarización IEC al mundo del lenguaje de alto nivel. Por lo tanto, el equipamiento de control actualmente admite tareas cíclicas, tareas inactivas y tareas basadas en eventos, que se reproducen en los procesos y subprocesos controlados por prioridad del sistema operativo en tiempo real. Además de esta ejecución controlada por prioridad, el equipamiento de control también permite la ejecución de tareas en respuesta a la ocurrencia de un evento especial. Por lo tanto, con este sistema libremente configurable es posible mezclar instancias de programas de diferentes dominios de programación dentro de una tarea y ejecutarlos conjuntamente. Esto significa que los modelos de proceso conocidos de IEC 61131-3 también están abiertos para otros dominios.

No obstante, debido a los grados de libertad que ofrece la ejecución de tareas con el equipamiento de control 50, también aumenta la complejidad en la configuración y análisis. Por lo tanto, el equipamiento de control 50 proporciona mecanismos integrados para una elaboración de perfiles detallada en base a las ampliaciones disponibles del sistema operativo en tiempo real. El usuario obtiene por tanto la posibilidad de observar, evaluar y registrar diferentes propiedades de la ejecución de la tarea y del sistema operativo. A estas propiedades pertenecen, entre otras cosas, los tiempos de ejecución de las tareas e instancias del programa, frecuencia de los desplazamientos de tareas, latencia y jitter, aprovechamiento del sistema o la supervisión de perro guardián. Con la ayuda de búferes circulares dinámicos, esta información también se puede evaluar para análisis *postmortem* después de un caso de error.

Además, el depósito central de la información de perfilado ofrece al equipamiento de control 50 la posibilidad de llevar a cabo evaluaciones estadísticas o poner a disposición estos datos para análisis posteriores a través de servicios de servidor remoto.

La fig. 3 muestra un ejemplo de modo de realización de acuerdo con la presente invención de un control de una secuencia de programa mediante un equipamiento de control 50. Para llevar a cabo una tarea de automatización, el equipamiento de control 50 también puede acceder a instancias auxiliares, como un esclavo o varios esclavos, como se muestra en la fig. 2.

La fig. 3 muestra una representación simplificada de una coordinación o control de tareas de control mediante un equipamiento de control 50, como está representado, por ejemplo, en la fig. 2. En este caso, las tareas 20, 30, 40 descritas de la fig. 1 se controlan mediante el equipamiento de control 50. Con este control y su evaluación, se puede entender claramente cuando los programas y tareas individuales se desplazan por procesos superiores. Estos procesos superiores están reservados para el sistema operativo y están representados como procesos del OS 19 en la fig. 3.

La fig. 3 muestra el tiempo en la coordenada x y los procesos o la prioridad del proceso en la coordenada y. Por tanto, como ya se explicó en relación con la fig. 1, la tarea 20 tiene la prioridad más baja y la tarea 40 la prioridad más alta. Con RTOS está representado un subproceso de alta prioridad 90 del sistema operativo en tiempo real (RTOS) que presenta la prioridad más alta. El subproceso de alta prioridad 90 o el subproceso del OS de alta prioridad 90 es, por lo tanto, parte del sistema operativo en tiempo real.

Por tanto, la primera tarea 20 con el módulo de programa 21 se ejecuta primero en términos de tiempo, pero esto se interrumpe mediante un proceso en tiempo real 19 del sistema operativo. Luego, se realiza el procesamiento de la tarea 30 con los módulos de programa 31 y 32. A continuación, la tarea 40 se procesa entonces con el módulo de programa 41. Aquí también hay interrupciones temporales de los módulos del programa mediante los procesos en tiempo real 19. Después del procesamiento de las prioridades más altas, la tarea 20 comienza de nuevo y a continuación se ejecuta la tarea 30, de modo que en la fig. 3 se indica un procesamiento cíclico de las tareas.

En conjunto, de acuerdo con la invención, se posibilita una funcionalidad como parte de una nueva arquitectura de firmware, en base a la cual se implementan nuevos controles industriales. Esta arquitectura está implementada en primer lugar para el uso en un aparato o aparato de control. No obstante, la arquitectura está diseñada de modo que es posible una ampliación de varios aparatos. Esto es el caso, por ejemplo, en un sistema de redundancia en el que dos controles se ejecutan de forma sincronizada entre sí.

En este caso, se pretende apoyar la compleción de lenguajes de alto nivel a la programación clásica de controles. Para ello, se busca una transición suave entre IEC 61131-3 y el mundo de los lenguajes de alto nivel. Además de apoyar dichos lenguajes, el requisito previo para esto es la integración perfecta de los diferentes dominios de programación. Entre ellos figuran entre otros Matlab/Simulink, C++, C#, JAVA, Python y los lenguajes IEC 61131-3. Para tal integración perfecta, el control debería ser capaz de transferir el proceso conocido de IEC 61131-3 con instancias de programa y tareas al proceso de los otros dominios de programación. A pesar de estos requisitos más complejos para la implementación técnica se garantizan adicionalmente aspectos de seguridad de datos, usabilidad, flexibilidad y, sobre todo, rendimiento.

La fig. 4 muestra un ejemplo de modo de realización de la presente invención de una estructura esquemática con un equipamiento de control 50. En este ejemplo de modo de realización está representado un proceso de sistema operativo o proceso del OS 11, que presenta subprocesos del OS 14, 15, 16.

El modo de proceder de acuerdo con la invención permite la composición de contenidos de tarea heterogéneos. A este respecto, la secuencia de procesamiento dentro de una tarea se puede seleccionar libremente, no predeterminada por el sistema, y es posible un intercambio de datos consistente entre instancias del programa.

La configuración del equipamiento de control 50 y sus tareas 20, 30, 40 se realiza mediante uno o varios archivos de descripción, aquí en forma de uno o varios archivos XML, que permiten la configuración en lugar de la programación. Por consiguiente, la función de una instalación se puede cambiar en tiempo de ejecución sin tener que compilar nuevamente las partes individuales de la aplicación. El archivo de descripción o la configuración basada en XML también le permite al usuario efectuar adaptaciones directamente en el control o utilizar sus propias herramientas y cadenas de herramientas externas.

A este respecto, la estructura del archivo de descripción, en particular en forma de un archivo XML, se puede predeterminar mediante archivos de esquema claramente definidos, que posibilita una comprobación rápida de la configuración. La especificación de tareas y las instancias de programa correspondientes en varios archivos de descripción o archivos de configuración XML permite dividir la aplicación total 10 en módulos funcionales que se pueden crear por diferentes grupos de desarrolladores o proveedores. Esto no solo facilita el desarrollo de un control de instalación complejo, sino también la ampliación gradual de una instalación en funcionamiento. Por tanto, la aplicación global 10 no debe estar disponible desde el principio, sino que se puede componer de submódulos.

Cada tarea 20, 30, 40 también se puede asignar a un núcleo de procesador, a un grupo o a todos los núcleos de procesador del control mediante la definición de parámetros en el archivo de descripción o archivo XML. Por la tarea 30 se envían comandos de desencadenamiento ("desencadena") respectivamente a los módulos de programa 31, 32 a través de las conexiones de comunicación 55, 56, en la que la comunicación tiene lugar dentro de un subproceso del OS.

El ejemplo de modo de realización representado en la fig. 4 posibilita ventajosamente la ejecución de los módulos de programa en un único proceso del OS común, en el que se pueden crear los módulos de programa con diferentes lenguajes de programación.

Las ventajas y configuraciones especificados anteriormente en relación con el ejemplo de modo de realización representado en la fig. 2, además de aquellas que se refieren al aspecto maestro/esclavo, también se aplican al ejemplo de modo de realización representado en la fig. 2.

5 A continuación se especifica a modo de ejemplo un posible contenido de un archivo de descripción en formato XML:

```

<?xml version="1.0" encoding="UTF-8"?>
<EsmConfigurationDocument>
<EsmComponentSettings>
10 <SharedDataSettings Size="1024" />
</EsmComponentSettings>
<EsmList>
<Esm Name="ESM0" CpuAffinity="0" TimerInterval="500us"
TaskSchedulerMode="IntervalCounting" />
15 <Esm Name="ESM1" CpuAffinity="1" TimerInterval="100us"
TaskSchedulerMode="SystemClock"/>
</EsmList>
<Tasks>
<Task Name="TaskCount" CycleTime="500us" TaskType="Cyclic" TaskPriority="4" />
20 <Task Name="TaskConf" CycleTime="250ms" TaskType="Cyclic" TaskPriority="5" />
<Task Name="TaskMain" CycleTime="10ms" TaskType="Cyclic" TaskPriority="7" />
</Tasks>
<EsmTaskRelations>
<EsmTaskRelation ESMName="ESM0" TaskName="TaskCount" />
25 <EsmTaskRelation ESMName="ESM0" TaskName="TaskConf" />
<EsmTaskRelation ESMName="ESM1" TaskName="TaskMain" />
</Esm TaskRelations>
<Programs>
<Program Name="CppCounter-1" ProgramType="CppCounter"
30 ComponentName="Arp.Demonstrator.CppCounter-1" />
<Program Name="CppConfig-1" ProgramType="CppConfig"
ComponentName="Arp.Demonstrator.CppConfig-1" />
<Program Name="PizController-1" ProgramType="PizController"
ComponentName="Arp.Demonstrator.PizController-1" />
35 <Program Name="SimHeli-1" ProgramType="SimHeli"
ComponentName="Arp.Demonstrator.SimHeli-1" />
</Programs>
<TaskProgramRelations>
<TaskProgramRelation TaskName="TaskCount"
40 ProgramName="Arp.Demonstrator.CppCounter-1/CppCounter-1" Order="0" />
<TaskProgramRelation TaskName="TaskConf"
ProgramName="Arp.Demonstrator.CppConfig-1/CppConfig-1" Order="0" />
<TaskProgramRelation TaskName="TaskMain"
ProgramName="Arp.Demonstrator.PizController-1/PizController-1" Order="0" />
45 <TaskProgramRelation TaskName="TaskMain"
ProgramName="Arp.Demonstrator.SimHeli-1.SimHeli-1" Order="1" />
</TaskProgramRelations>
</EsmConfigurationDocument>

```

50 Lista de referencias

- | | |
|----|----------------------------------|
| 10 | Aplicación global |
| 11 | Proceso del OS |
| 55 | 12 Proceso del OS |
| | 13 Proceso del OS |
| 60 | 14 Subproceso del OS |
| | 15 Subproceso del OS |
| | 16 Subproceso del OS |
| 65 | 17 Subproceso del OS |
| | 18 Subproceso del OS |
| 70 | 19 Proceso del sistema operativo |

	20	Tarea o task
	21	Módulo de programa
5	30	Tarea o task
	31	Módulo de programa
	32	Módulo de programa
10	33	Fin del procedimiento
	34	Interfaz de comunicación
15	40	Tarea o task
	41	Módulo de programa
	50	Equipamiento de control o ESM
20	51	Conexión efectiva
	52	Conexión efectiva
25	53	Conexión de comunicación
	54	Conexión de comunicación
	55	Conexión de comunicación
30	56	Conexión de comunicación
	57	Conexión de comunicación
35	58	Conexión de comunicación
	60	Equipamiento de control o ESM como primer esclavo
	70	Equipamiento de control o ESM como segundo esclavo
40	80	Proceso del sistema operativo
	90	Subproceso de sistema operativo de alta prioridad

REIVINDICACIONES

1. Procedimiento para el control de secuencia de módulos de programa (21, 31, 32, 41), en el que los módulos de programa se ejecutan en un aparato de control en una instalación de automatización, y en el que los módulos de programa son parte de una aplicación de control para la ejecución en el aparato de control, y en el que el aparato de control está conformado para controlar un sistema de automatización mediante la ejecución de la aplicación de control, en el que la ejecución de la aplicación de control comprende la ejecución de los módulos de programa, que comprende
facilitación de un equipamiento de control (50), en el que el equipamiento de control forma una parte del firmware del aparato de control,
facilitación de varios módulos de programa (21, 31, 32, 41) que se pueden ejecutar por un sistema operativo en tiempo real,
facilitación de un archivo de configuración, que presenta una configuración para controlar una secuencia de programa, en el que la configuración para módulos de programa seleccionados comprende respectivamente una identificación e información asignada para ejecutar el módulo de programa identificado respectivamente, evaluación de la información contenida en el archivo de configuración mediante el equipamiento de control, y control de secuencia de los módulos de programa seleccionados mediante la ejecución de los módulos de programa seleccionados (21, 31, 32, 41) mediante el equipamiento de control (50) dependiendo de la información especificada en el archivo de configuración, en el que los contenidos del archivo de configuración predeterminan cuál de los módulos de programa seleccionados en qué orden y dependiendo de qué evento o en qué intervalos de tiempo se ejecutan por el equipamiento de control.
2. Procedimiento de acuerdo con la reivindicación 1, además que presenta ejecución múltiple de al menos uno de los módulos de programa (21, 31, 32, 41) mediante el equipamiento de control (50).
3. Procedimiento de acuerdo con la reivindicación 1 o la reivindicación 2, caracterizado por que el equipamiento de control (50) está conformado para ejecutar al menos algunos de los módulos de programa (21, 31, 32, 41), de tal manera que se pueden ejecutar en diferentes procesos gestionados por el sistema operativo en tiempo real.
4. Procedimiento de acuerdo con la reivindicación 3, caracterizado por que el sistema operativo en tiempo real está conformado para ejecutar los procesos y/o módulos de programa (21, 31, 32, 41) dependiendo de prioridades predeterminadas.
5. Procedimiento de acuerdo con la reivindicación 4, además que presenta registro y evaluación del control de secuencia mediante el sistema operativo en tiempo real.
6. Procedimiento de acuerdo con una de las reivindicaciones anteriores, además que presenta cambio del control de secuencia mediante agregación o retirada de módulos de programa sin que se lleve a cabo un reinicio de un sistema de control.
7. Procedimiento de acuerdo con una de las reivindicaciones anteriores, además que presenta división del control de secuencia en distintos procesos.

8. Procedimiento de acuerdo con una de las reivindicaciones anteriores,
además que presenta
creación de una evaluación estadística utilizando información de perfilado.
9. Procedimiento de acuerdo con una de las reivindicaciones anteriores,
además que presenta
incorporación de al menos una marca de registro en un módulo de programa (21, 31, 32, 41) para proporcionar un diagnóstico de aplicación.
10. Procedimiento de acuerdo con una de las reivindicaciones anteriores,
además que presenta
uso del equipamiento de control (50) como maestro y
facilitación de al menos otro equipamiento de control (60, 70) como esclavo, que está subordinado al equipamiento de control (50) como maestro.
11. Procedimiento de acuerdo con una de las reivindicaciones anteriores,
además que presenta
asignación de tareas individuales definidas en el archivo de configuración a módulos de programa (21, 31, 32, 41) mediante el equipamiento de control (50).
12. Aparato de control para controlar un sistema de automatización, que presenta un procesador,
una memoria,
al menos una interfaz a un dispositivo de automatización, y un firmware, que comprende un equipamiento de control (50),
en el que el equipamiento de control (50) está conformado para
evaluar la información contenida en un archivo de configuración, en el que el archivo de configuración para varios módulos de programa (21, 31, 32, 41) ejecutables por un sistema operativo en tiempo real comprende respectivamente una identificación e información asignada para ejecutar el módulo de programa identificado respectivamente, y en el que los contenidos del archivo de configuración predeterminan cuál de los módulos de programa seleccionados en qué orden y dependiendo de qué evento o en qué intervalos de tiempo se ejecutan por el equipamiento de control, y
ejecutar un control de secuencia de los módulos de programa mediante ejecución de los módulos de programa (21, 31, 32, 41) dependiendo de la información especificada en el archivo de configuración, en el que los módulos de programa se ejecutan en el aparato de control en una instalación de automatización, y en el que los módulos de programa son parte de una aplicación de control para la ejecución en el aparato de control, y en el que el archivo de configuración está depositado en la memoria del aparato de control, y en el que el aparato de control está conformado para controlar el sistema de automatización mediante la ejecución de la aplicación de control, en el que la ejecución de la aplicación de control comprende la ejecución de los módulos del programa.
13. Producto de programa informático con medios de código de programa que provocan que el aparato de control de acuerdo con la reivindicación 12 lleva a cabo el procedimiento de acuerdo con una de las reivindicaciones 1 a 11.
14. Soporte de datos, en el que está almacenado el producto de programa informático de acuerdo con la reivindicación 13.

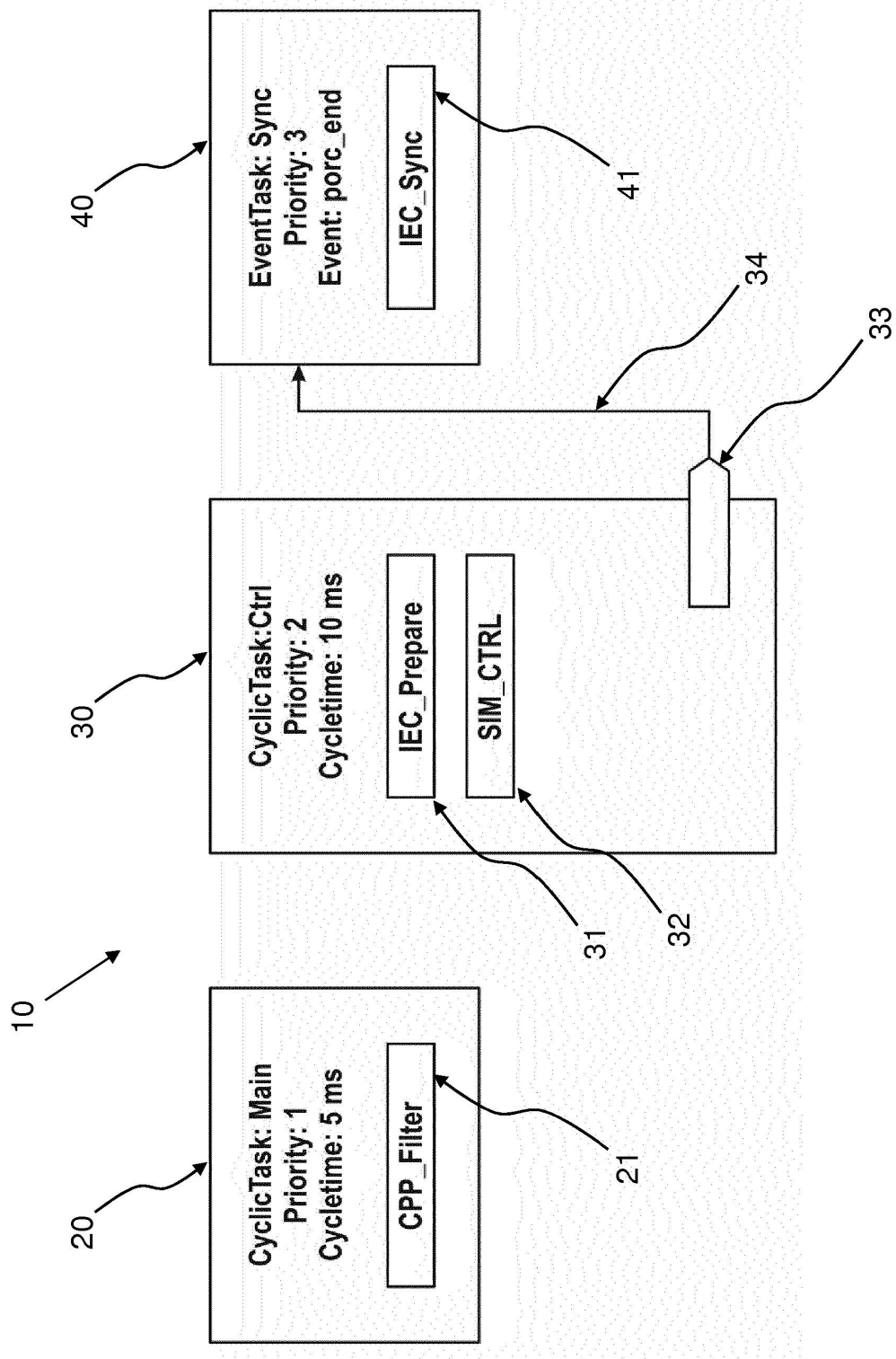


Fig. 1

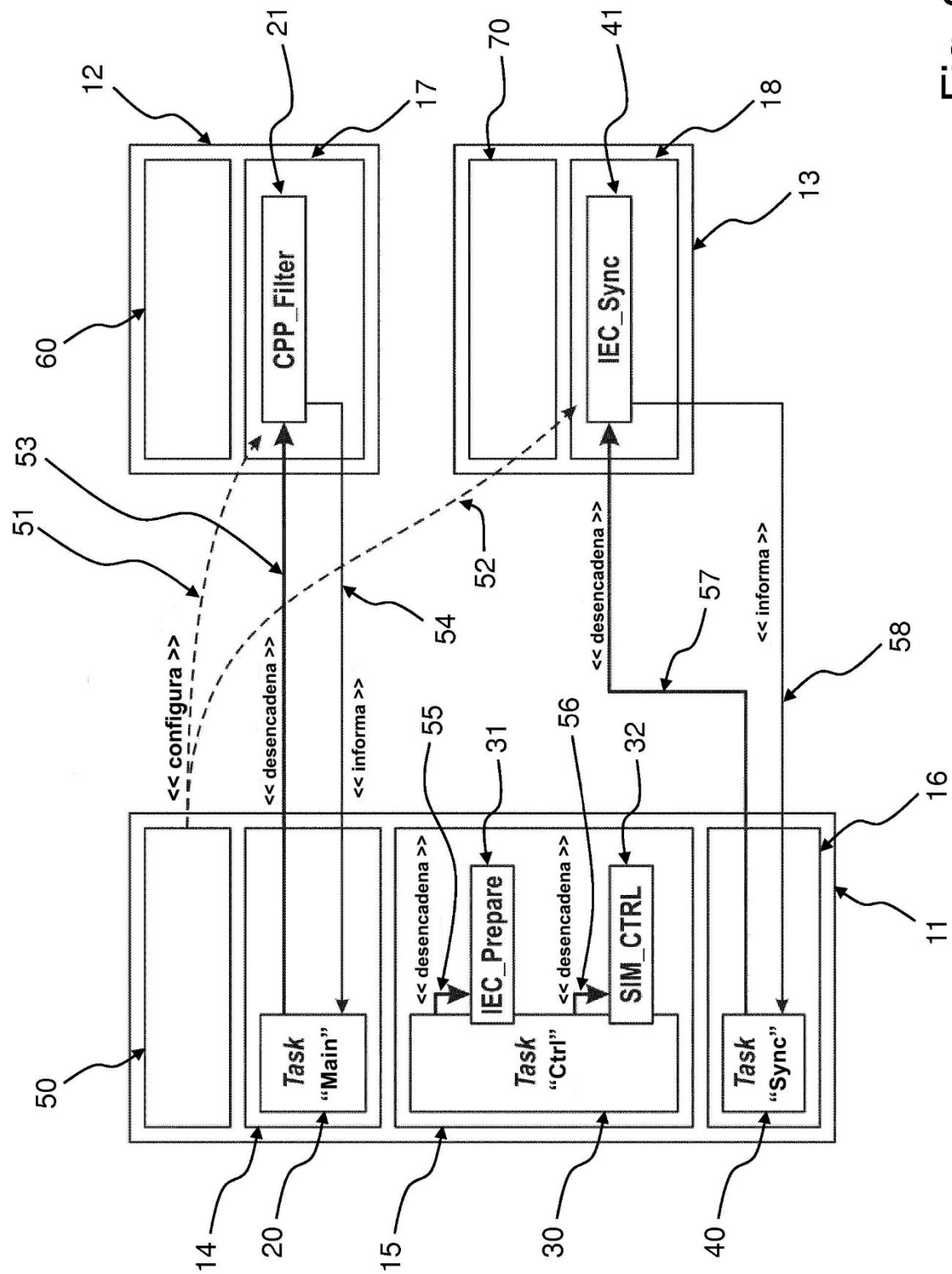


Fig. 2

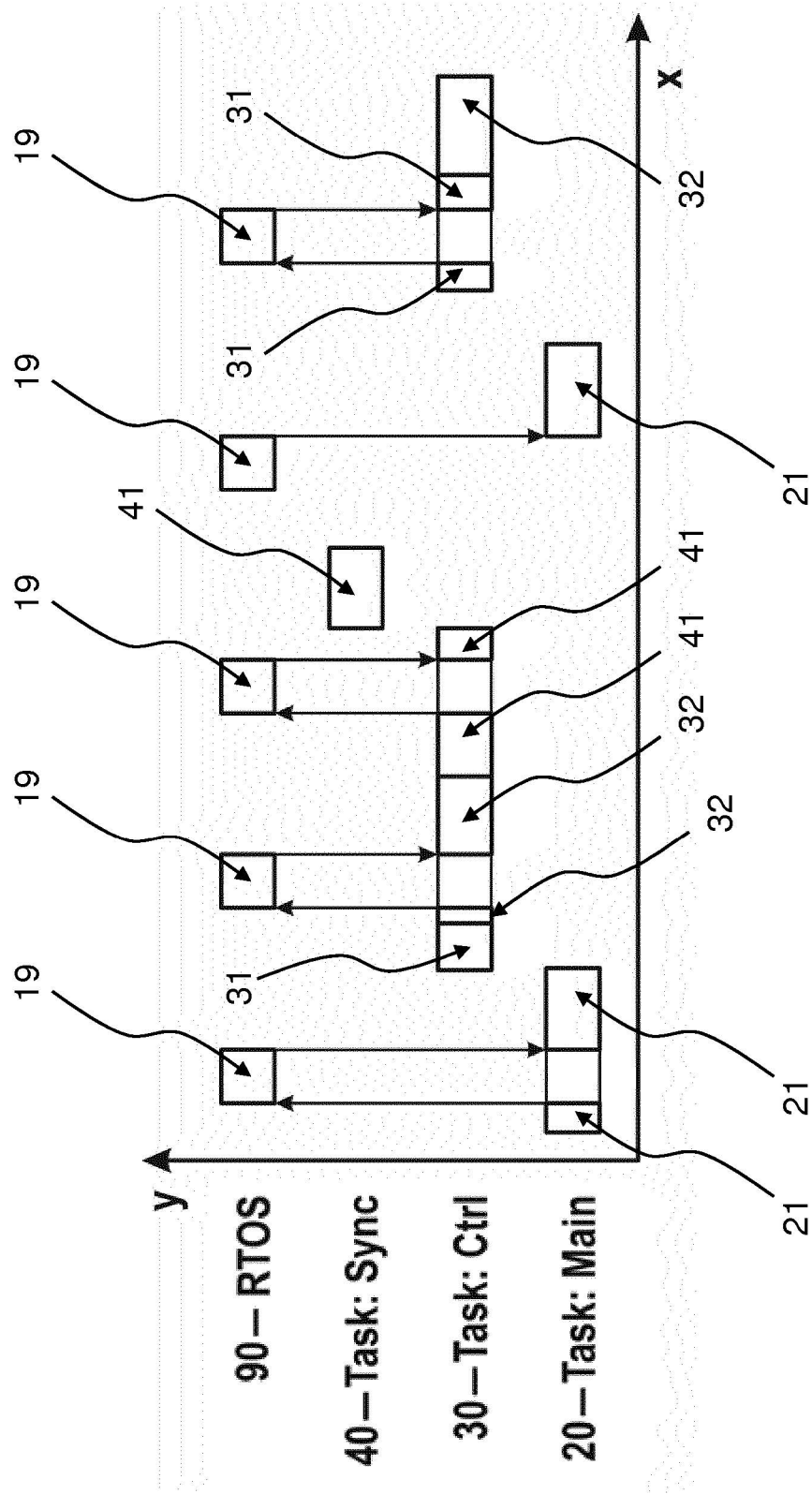


Fig. 3

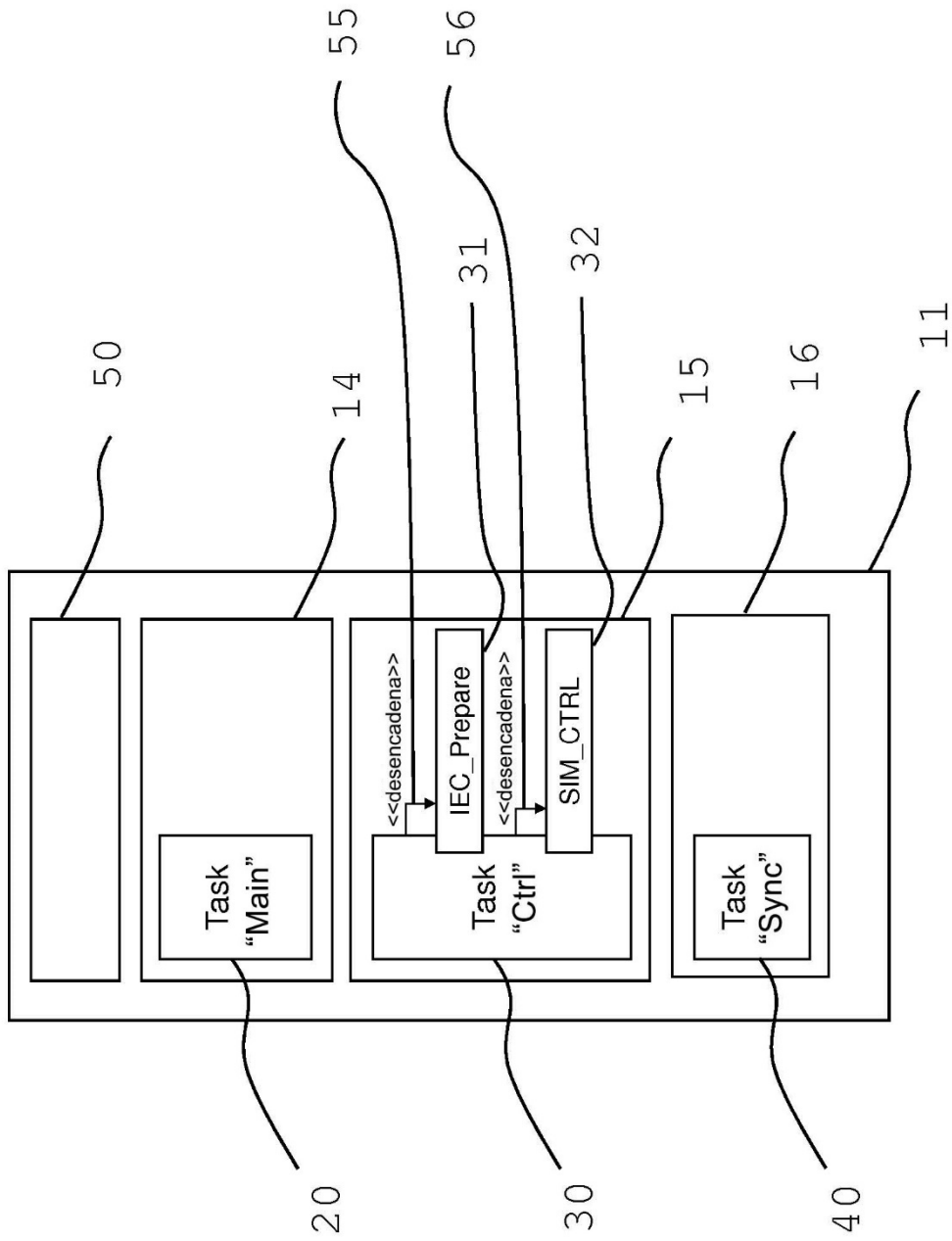


Fig. 4