(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **3 009 863**

(51) Int. Cl.:

G05B 9/02 (2006.01)
G06F 9/50 (2006.01)
G05B 19/406 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Fecha de presentación y número de la solicitud europea: **22.12.2022 E 22216057 (4)**

(97) Fecha y número de publicación de la concesión europea: **30.10.2024 EP 4390577**

(54) Título: **Monitorización de al menos una máquina**

(45) Fecha de publicación y mención en BOPI de la traducción de la patente:
01.04.2025

(73) Titular/es:

SICK AG (100.00%)
Erwin-Sick-Strasse 1
79183 Waldkirch, DE

(72) Inventor/es:

NEUMANN, THOMAS;
STEINKEMPER, HEIKO y
LIEGIBEL, PASCAL

(74) Agente/Representante:

DEL VALLE VALIENTE, Sonia

DESCRIPCIÓN

Monitorización de al menos una máquina

La invención se refiere a un dispositivo de seguridad y a un método para monitorizar al menos una máquina según el preámbulo de las reivindicaciones 1 y 14, respectivamente.

La ingeniería de seguridad se ocupa de la protección personal y de la prevención de accidentes con máquinas. Un dispositivo de seguridad de la categoría usa uno o más sensores para monitorizar una máquina o su entorno y conmutar la misma en un estado seguro a su debido tiempo cuando existe un peligro inminente. Una solución de ingeniería de seguridad convencional típica monitoriza un campo protegido, al que los operadores no pueden acceder durante el funcionamiento de la máquina, mediante el al menos un sensor, por ejemplo, mediante un escáner láser. Si el sensor reconoce una intrusión no autorizada en el campo protegido, por ejemplo una pierna de un operador, este desencadena una detención de emergencia de la máquina. Existen conceptos de protección alternativos, tales como la así denominada monitorización de velocidad y separación, en la que se evalúan las distancias y velocidades de los objetos detectados en el entorno y se da una respuesta en una situación arriesgada.

En la ingeniería de seguridad se requiere una fiabilidad especial y, por lo tanto, han de satisfacerse unas exigencias de seguridad altas, por ejemplo, la norma EN13849 para la seguridad de la maquinaria y la norma de maquinaria EN61496 para equipos de protección electrosensibles (ESPE). Algunas medidas típicas para este fin son una evaluación electrónica segura mediante diversos procesos electrónicos redundantes o de monitorización funcional diferentes, por ejemplo, la monitorización de la contaminación de los componentes ópticos, incluyendo una pantalla frontal. De forma algo más general, han de demostrarse unas medidas de control de defectos bien definidas de tal modo que puedan evitarse o controlarse defectos críticos posibles para la seguridad a lo largo de la cadena de señales desde el sensor a través de la evaluación hasta el inicio de la respuesta de ingeniería de seguridad.

Debido a las exigencias altas de hardware y software en la ingeniería de seguridad, hasta la fecha se han usado principalmente arquitecturas monolíticas que usan hardware desarrollado específicamente que prevé redundancias y una monitorización funcional mediante posibilidades de prueba y capacidad de múltiples canales. En consecuencia, se documenta una prueba de algoritmos correctos, por ejemplo, según las normas IEC TS 62998 e IEC-61508-3, y el proceso de desarrollo del software está sujeto a pruebas y comprobaciones estrictas y permanentes. Un ejemplo para esto es un escáner láser de seguridad tal como el conocido, por ejemplo, por primera vez a partir del documento DE 43 40 756 A1 y cuyas características principales se usan de forma generalizada hasta la fecha. Toda la función de evaluación, incluyendo la medición de tiempo de vuelo para la determinación de distancia y la detección de objetos en los campos protegidos configurados, está integrada allí. El resultado es una señal de salvaguarda binaria completamente evaluada en una salida de dos canales (OSSD, dispositivo de conmutación de señal de salida) del escáner láser que detiene la máquina en el caso de una intrusión en el campo protegido. Aunque este concepto ha demostrado ser válido, sigue siendo inflexible, debido a que los cambios prácticamente solo son posibles mediante un nuevo desarrollo de un modelo de seguimiento del escáner láser.

En algunas aplicaciones de seguridad convencionales, al menos parte de la evaluación se externaliza del sensor a un controlador programable (PLC, controlador lógico programable). Sin embargo, se requieren controladores de seguridad particulares para este fin que tengan estructuras de múltiples canales y similares para la prevención y detección de defectos. Por lo tanto, estos son caros y proporcionan comparativamente poca capacidad de memoria y capacidad de procesamiento que, por ejemplo, se ven completamente abrumadas por el procesamiento de imágenes en 3D.

Es cierto que, en principio, sería concebible el uso de controladores convencionales mientras se integraran en los procesos de monitorización funcional requeridos, pero esto apenas se usa en un entorno industrial hoy en día, debido a que ello requiere arquitecturas más complejas y un conocimiento experto. En otro orden de cosas, los controladores o PLC convencionales solo pueden programarse con ciertos lenguajes con, en parte, un área de lenguaje muy limitada. Incluso unos bloques funcionales relativamente simples requieren esfuerzos de desarrollo y recursos de tiempo de ejecución sustanciales, de tal modo que su implementación en un controlador convencional apenas puede lograrse con aplicaciones algo más complejas, particularmente con medidas de seguridad tales como redundancias.

El documento EP 3 709 106 A1 combina un controlador de seguridad con un controlador convencional en un sistema de seguridad. Los cálculos más complejos siguen correspondiendo al controlador convencional y sus resultados son validados por el controlador de seguridad. Sin embargo, el controlador de seguridad hace uso de datos seguros existentes de un sensor seguro para este fin, lo que restringe los escenarios de aplicación posibles y, adicionalmente, requiere un conocimiento experto para adquirir y, por lo tanto, validar adecuadamente datos seguros adecuados. Además, la estructura de hardware se predefine de forma fija y la aplicación se implementa específicamente de forma fija en la misma.

En un número de casos, sería deseable combinar la monitorización de seguridad con un trabajo de automatización. Por lo tanto, no solo se evitan accidentes sino que la tarea propiamente dicha de la máquina se soporta análogamente de una forma automatizada. Hasta la fecha, se han usado principalmente sistemas y sensores completamente

diferentes para este fin. Esto es debido, entre otras cosas, al hecho de que un sensor de seguridad para trabajos de automatización es demasiado caro y, a la inversa, la complejidad de un sensor de seguridad no debería sobrecargarse con funciones adicionales. El documento EP 2 053 538 B1 permite la definición de regiones de seguridad y automatización separadas para una cámara 3D. Sin embargo, esta es solo una primera etapa, debido a que, si bien es cierto que sigue usándose el mismo sensor para los dos mundos de la seguridad y la automatización, estas dos tareas vuelven a separarse claramente entre sí a continuación desde el punto de vista espacial y en el lado de implementación. La norma IEC 62998 permite la coexistencia de datos de seguridad y automatización, pero no hace ninguna propuesta de implementación específica como una norma.

Hace tiempo que existen arquitecturas mucho más flexibles fuera de la ingeniería de seguridad. Hace mucho tiempo que el enfoque monolítico ha dado paso, en un número de etapas, a conceptos más modernos. Es cierto que el despliegue tradicional anterior con hardware fijo en el que un sistema operativo coordina las aplicaciones individuales sigue estando justificado en dispositivos autónomos, pero hace mucho tiempo que ha dejado de ser satisfactorio en un mundo conectado en red. La idea básica en el desarrollo adicional fue la inclusión de capas adicionales que van abstrayéndose cada vez más del hardware específico.

Las así denominadas máquinas virtuales, donde la capa adicional se denomina hipervisor o supervisor de máquina virtual, son una primera etapa. Mientras tanto, estos enfoques también se han seguido provisionalmente en la ingeniería de seguridad. El documento EP 3 179 279 B1, por ejemplo, proporciona un entorno protegido en un sensor de seguridad para permitir que el usuario permita que sus propios módulos de programa se ejecuten en el sensor de seguridad. Sin embargo, a continuación tales módulos de programa se separan cuidadosamente de la funcionalidad de seguridad y no contribuyen con nada a la misma.

Otra abstracción se basa en los así denominados contenedores (visualización de contenedores, contenedorización). Un contenedor es, por así decirlo, una pequeña cápsula virtual para una aplicación de software que proporciona un entorno completo para su ejecución, incluyendo áreas de memoria, bibliotecas y similares. La capa de abstracción o el entorno de tiempo de ejecución asociados se denominan tiempo de ejecución de contenedor. De este modo, la aplicación de software puede desarrollarse independientemente del hardware, que puede ser prácticamente cualquier hardware en el que esta se ejecute posteriormente. Los contenedores se implementan frecuentemente con la ayuda de cargadores.

En una arquitectura de IoT (Internet de las cosas, Internet industrial de las cosas) moderna, se combina una pluralidad de contenedores que tienen las aplicaciones de software más variadas. Estos contenedores tienen que coordinarse adecuadamente, lo que a este respecto se conoce como orquestación, y para lo cual se añade una así denominada capa de orquestación como una abstracción adicional. Kubernetes ha ido estableciéndose cada vez más para la orquestación de contenedores; además, alternativas tales como un enjambre de cargadores se han dado a conocer como extensiones de cargadores, así como rkt o LXC.

El uso de arquitecturas tan modernas y de tanta abstracción en la ingeniería de seguridad ha fallado previamente debido a los importantes obstáculos de las normas de seguridad y al enfoque correspondientemente conservador en el campo de aplicación de la seguridad funcional. Indiscutiblemente, las tecnologías de contenedores se están siguiendo generalmente en el entorno industrial y hay planes, por ejemplo en la industria automotriz, para el uso de la arquitectura Kubernetes; y la fuerza aérea alemana también está siguiendo tales enfoques. Sin embargo, nada de esto está dirigido a la seguridad funcional y, por lo tanto, no soluciona los problemas mencionados.

Es cierto que también se desea una disponibilidad alta en un mundo de IoT habitual, pero esta forma de protección frente a fallos no es comparable en modo alguno a lo que requieren las normas de seguridad. Por lo tanto, las aplicaciones perimetrales o en la nube han sido inconcebibles para que la seguridad satisfaga las normas para un ingeniero de seguridad. Ello contradice el concepto generalizado de proporcionar condiciones reproducibles y de prepararse para todas las posibilidades de un mal funcionamiento que sean imaginables en estas condiciones. Una abstracción o visualización profunda proporciona una incertidumbre adicional que previamente parecía incompatible con las exigencias de seguridad.

El documento EP 4 040 034 A1 presenta un dispositivo de seguridad y un método de seguridad para monitorizar una máquina en la que la funcionalidad de seguridad puede abstraerse del hardware subyacente usando dichas tecnologías de contenedor y orquestación. Las unidades de lógica se generan, se resuelven o se asignan a otro hardware según se requiera. Esto permite grados variables de redundancia y una monitorización mutua flexible de unidades de lógica. Se proponen unidades de lógica especiales configuradas como unidades de diagnóstico para la prueba y la monitorización. Sin embargo, el documento EP 4 040 034 A1 no explica cómo es posible específicamente, con la ayuda del concepto de una unidad de diagnóstico, localizar realmente los defectos relacionados con la seguridad que tienen lugar en el sistema total.

El documento US 2021/0247743 A1 se refiere a la seguridad funcional en vehículos autónomos, robots y sistemas de control industrial.

Por lo tanto, el objeto de la invención es mejorar adicionalmente el concepto de seguridad flexible recién descrito para su implementación práctica.

Este objeto es satisfecho por un dispositivo de seguridad y por un método para monitorizar al menos una máquina según la reivindicación independiente respectiva. La máquina monitorizada o la máquina que va a salvaguardarse debería entenderse inicialmente en general; es, por ejemplo, una máquina de procesamiento, una línea de producción, un puesto de clasificación, una unidad de proceso, un robot o un vehículo en un número grande de variaciones, tales como sobre rieles o no, guiado o sin conductor, y similares. Al menos un sensor suministra datos de sensor a la máquina, es decir, datos acerca de la propia máquina, acerca de con qué interacciona la misma o acerca de su entorno. Los datos de sensor están dirigidos, al menos parcialmente, a la seguridad; son concebibles datos de sensor adicionales no dirigidos a la seguridad para funciones de automatización o funciones de comodidad. Los sensores pueden ser sensores de seguridad, pero no tienen que serlo; la seguridad solo puede asegurarse en una posición posterior.

Una unidad de procesamiento actúa como el entorno de tiempo de ejecución. La unidad de procesamiento es, por lo tanto, el elemento estructural; el entorno de tiempo de ejecución es su función. La unidad de procesamiento está conectada al menos indirectamente al sensor y a la máquina. En consecuencia, esta tiene acceso a los datos de sensor para su procesamiento, posiblemente de forma indirecta a través de otras unidades interpuestas, y puede comunicarse con la máquina y, en particular, puede influir en la misma, preferiblemente a través de un control de máquina de la máquina. La unidad de procesamiento o el entorno de tiempo de ejecución designa, como un término genérico, el hardware y el software con los que se toma una decisión acerca del requisito y, preferiblemente, también acerca del tipo de respuesta dirigida a la seguridad de la máquina con referencia a los datos de sensor.

La unidad de procesamiento comprende al menos un nodo informático. Es un dispositivo informático digital o un nodo de hardware o una parte del mismo que proporciona capacidades de procesamiento y memoria para ejecutar un bloque de funciones de software. Sin embargo, no todos los nodos informáticos tienen que ser necesariamente un módulo de hardware independiente; una pluralidad de nodos informáticos puede implementarse, por ejemplo, en el mismo dispositivo usando múltiples procesadores y, a la inversa, un nodo informático también puede agrupar diferentes recursos de hardware.

Una pluralidad de unidades de lógica se ejecutan en el nodo informático o en uno de los nodos informáticos en el funcionamiento del dispositivo de seguridad. En consecuencia, una unidad de lógica designa generalmente un bloque de funciones de software. Según la invención, al menos una unidad de lógica está configurada como una unidad de función de seguridad que realiza una evaluación relacionada con la seguridad de los datos de sensor. El objetivo de la evaluación relacionada con la seguridad es la protección personal o la prevención de accidentes en el sentido de que se determina, con referencia a los datos de sensor, si un riesgo es inminente o si se ha reconocido un evento relacionado con la seguridad. Este es, por ejemplo, el caso de la detección de una persona demasiado cerca de la máquina o en un campo protegido. Una o más unidades de lógica pueden participar en la evaluación relacionada con la seguridad. En el caso de un evento dirigido a la seguridad, preferiblemente se envía una señal de seguridad a la máquina para desencadenar allí una respuesta dirigida a la seguridad mediante la cual la máquina se conmuta a un estado seguro que elimina el riesgo o al menos lo reduce a un nivel aceptable. Al menos una unidad de lógica está configurada además como una unidad de diagnóstico. De este modo, la función de otras unidades de lógica y, en particular, de la al menos una unidad de función de seguridad se monitoriza en busca de defectos.

La invención parte de la idea básica de llevar a cabo una monitorización de estado y rendimiento de las unidades de lógica por medio de la unidad de diagnóstico. Para este fin, la al menos una unidad de función de seguridad transmite informes de estado e informes de rendimiento a la unidad de diagnóstico que se evalúan en la misma. El estado o estatus de la al menos una unidad de función de seguridad proporciona información acerca de su disponibilidad operativa y acerca de restricciones o defectos posibles en la al menos una unidad de función de seguridad. Los informes de rendimiento se refieren a la ejecución de la función de seguridad o del servicio que la al menos una unidad de función de seguridad realiza y, a partir de los mismos, puede generarse una rutina de rendimiento de las funciones o servicios de seguridad realizados. En conjunto, esto permite un diagnóstico de sistema mediante el cual puede reconocerse un mal funcionamiento del dispositivo de seguridad relacionado con la seguridad. A este respecto, la unidad de diagnóstico no requiere conocimientos especiales ni en cuanto a cómo o con qué algoritmo funciona una unidad de función de seguridad ni en cuanto a los resultados de evaluación que suministra la misma, aunque ambas cosas serían posibles de una forma complementaria. En el caso de un defecto, no puede asegurarse el funcionamiento seguro del dispositivo de seguridad, preferiblemente con consecuencias similares de una respuesta de la máquina relacionada con la seguridad que en el caso de un riesgo reconocido por un módulo de función de seguridad. Un único módulo de lógica configurado como una unidad de diagnóstico es suficiente para el diagnóstico de sistema, pero también sería concebible implementar la monitorización de estado y rendimiento en sus unidades de diagnóstico propias respectivas o distribuir la funcionalidad a lo largo de una pluralidad de módulos de lógica.

La invención tiene la ventaja de que se hace posible una arquitectura de seguridad altamente flexible que coordina u orquesta aplicaciones dirigidas a la seguridad en un entorno industrial (Internet de las cosas industrial, IIoT). A este respecto, la seguridad y la automatización pueden estar estrechamente entrelazadas. Un hardware convencional es suficiente; no se requiere ningún hardware seguro dedicado costoso. La invención es en gran medida independiente

del panorama de hardware específico siempre que haya disponibles, en conjunto, suficientes recursos informáticos y de memoria. Además, la robustez aumenta sustancialmente debido a que las unidades de lógica pueden implementarse en hardware diverso y pueden desplazarse entre nodos informáticos. El panorama del hardware o parte del panorama del hardware puede ser una nube como una aplicación concebible importante; por lo tanto, la invención combina los mundos previamente ajenos de la nube y la ingeniería de seguridad. En el marco de los conceptos nativos de la nube, existen marcos y herramientas de código abierto que también soportan aplicaciones diseñadas ampliamente, pero no proporcionan aún ninguna seguridad funcional.

El enfoque según la invención se especifica de forma radicalmente diferente a lo que es convencional en la ingeniería de seguridad. Previamente se ha predefinido una estructura de hardware fija, que típicamente se desarrolla por separado exactamente para esta función de seguridad, y la funcionalidad del software se desarrolla exactamente para esta estructura de hardware, y allí se implementa y se prueba de forma fija. Se excluye un cambio posterior en el despliegue de software y esto es aplicable aún en mayor medida por un cambio del hardware subyacente. Tales modificaciones requieren convencionalmente al menos una conversión compleja por un ingeniero de seguridad y, por regla general, un desarrollo nuevo y completo. Un producto del enfoque conservador típico en la industria y, sobre todo, en la ingeniería de seguridad es que incluso las actualizaciones de firmware o las actualizaciones de software de los sensores y controladores se llevan a cabo, en el caso más extremo, en ciclos largos y, típicamente, no se realizan en absoluto.

Los términos seguridad o seguro se usan una y otra vez en esta descripción. Preferiblemente, estos han de entenderse respectivamente en el sentido de una norma de seguridad. En consecuencia, se satisface una norma de seguridad, por ejemplo, para la seguridad de las máquinas, los equipos de protección electrosensibles o la prevención de accidentes en la protección personal o, dicho de otra forma, se observan niveles de seguridad definidos por las normas, en consecuencia, se gestionan defectos hasta un nivel de seguridad definido por la norma, en consecuencia, defectos respectivos hasta un nivel de seguridad especificado en la norma de seguridad o especificado de una forma análoga a la misma. Algunos ejemplos de tales normas de seguridad se han mencionado en la introducción, donde los niveles de seguridad se denominan, por ejemplo, clases de protección o niveles de rendimiento. La invención no se limita a una de estas normas de seguridad específicas que pueden variar en su numeración y redacción específicas a nivel regional y a lo largo del tiempo, pero no en sus principios básicos para proporcionar seguridad. El término seguridad se amplía un poco más abajo en algunas realizaciones para incluir seguridad situacional o relacionada con el contexto.

La implementación del entorno de tiempo de ejecución tiene lugar preferiblemente en Kubernetes. Allí, el entorno de tiempo de ejecución se denomina “plano de control”. Un maestro coordina las rutinas o la orquestación (capa de orquestación). Los nodos informáticos se denominan nodos en Kubernetes y tienen al menos un subnodo o cápsula en el que las unidades de lógica se ejecutan en contenedores respectivos. Kubernetes ya está al tanto de mecanismos mediante los cuales se hace una comprobación en cuanto a si una unidad de lógica sigue funcionando. Sin embargo, esta comprobación no satisface ninguna exigencia específica de seguridad y está sustancialmente restringida para obtener una señal de vida útil de vez en cuando y, posiblemente, para reiniciar un contenedor. En el presente caso no hay garantías en cuanto a cuándo tiene lugar el defecto y se ha subsanado de nuevo.

El entorno de tiempo de ejecución está configurado preferiblemente para producir y resolver unidades de lógica y para asignar las mismas a un nodo informático o para desplazar las mismas entre nodos informáticos. Esto se hace preferiblemente no solo una vez, sino también dinámicamente durante el funcionamiento y también se refiere de forma muy explícita a las unidades de lógica relacionadas con la seguridad, es decir, la al menos una unidad de función de seguridad y/o la unidad de diagnóstico. Por lo tanto, el vínculo entre el hardware y la evaluación es fluido al tiempo que se mantiene la seguridad funcional. Convencionalmente, todas las funciones de seguridad se implementan de forma fija e inmutable en hardware dedicado. Un cambio, cuando pudiera producirse en absoluto sin conversión o un nuevo desarrollo, se consideraría completamente incompatible con el concepto de seguridad subyacente. Esto ya se es aplicable a una implementación puntual y, en particular, a cambios dinámicos en el tiempo de ejecución. En cambio, previamente todo se hacía con un esfuerzo ciertamente grande y un número grande de medidas individuales complejas para que la función de seguridad hallara un entorno bien definido e inalterado al principio y a lo largo del tiempo operativo total.

El entorno de tiempo de ejecución está configurado preferiblemente para cambiar los recursos asignados a una unidad de lógica. Puede ayudar a la unidad de lógica a procesar más rápido, pero también puede liberar recursos para otras unidades de lógica. Las posibilidades particulares para proporcionar más recursos son el desplazamiento de una unidad de lógica a otro nodo informático o la generación de otro nodo informático o la generación de una instancia o copia adicional de la unidad de lógica, estando preferiblemente una unidad de lógica para esta última configurada para un rendimiento que pueda paralelizarse.

El entorno de tiempo de ejecución guarda preferiblemente la información de configuración o un archivo de configuración en las unidades de lógica almacenadas. Se mantiene o se especifica un registro de las unidades de lógica presentes con referencia a la información de configuración en cuanto a qué unidades de lógica deberían ejecutarse en qué rutina de tiempo y con qué recursos y en cuanto a cómo están posiblemente en relación entre sí.

La información de configuración está particularmente protegida preferiblemente contra la manipulación por medio de firmas o conjuntos de datos de cadena de bloques. Una manipulación de este tipo puede ser deliberada o no deliberada; en cualquier caso, la configuración de las unidades de lógica no debería cambiarse de forma inadvertida en una aplicación de seguridad.

El entorno de tiempo de ejecución tiene preferiblemente al menos una unidad maestra que se comunica con el nodo informático y lo coordina. La unidad maestra también puede tener una pluralidad de subunidades para redundancia y/o para responsabilidades distribuidas o puede ser asistida por unidades de gestión de nodo de los nodos informáticos y puede implementarse en un nodo informático separado o en un nodo informático junto con unidades de lógica.

El al menos un nodo informático tiene preferiblemente una unidad de gestión de nodo para la comunicación con otros nodos informáticos y con el entorno de tiempo de ejecución. Esta unidad de gestión de nodo es responsable de la gestión y las coordenadas del nodo informático asociado, en particular, las unidades de lógica de este nodo informático, y de la interacción con los otros nodos informáticos y con la unidad maestra. También puede asumir el trabajo de la unidad maestra en prácticamente cualquier despliegue deseado.

El al menos un nodo informático tiene preferiblemente al menos un subnodo y las unidades de lógica están asociadas a un subnodo. De este modo, los nodos informáticos se estructuran en sí mismos una vez más para combinar unidades de lógica en un subnodo. Este concepto también sigue Kubernetes en forma de cápsulas.

La al menos una unidad de lógica se implementa preferiblemente como un contenedor. A continuación, las unidades de lógica se encapsulan o se contenedorizan y son ejecutables prácticamente en cualquier hardware deseado. La relación, por lo demás habitual, entre la función de seguridad y su implementación en hardware fijo se descompone de tal modo que la flexibilidad y la estabilidad del proceso se aumentan muy considerablemente. El entorno de tiempo de ejecución coordina u orquesta los contenedores que tienen las unidades de lógica ubicadas en los mismos entre sí. Hay al menos dos capas de abstracción, por un lado, una capa de contenedor (tiempo de ejecución de contenedor) respectiva y, por otro lado, una capa de orquestación del entorno de tiempo de ejecución dispuesta por encima de la misma.

El entorno de tiempo de ejecución se implementa preferiblemente en al menos un sensor, un controlador lógico programable, un controlador de máquina, un dispositivo procesador en una red local, un dispositivo perimetral y/o en una nube. El panorama del hardware subyacente es, en otros trabajos, prácticamente el deseado, lo que es una gran ventaja del enfoque según la invención. El entorno de tiempo de ejecución funciona de forma abstracta con nodos informáticos; el hardware subyacente puede tener una composición muy heterogénea. Las arquitecturas perimetrales o en la nube, en particular, se vuelven accesibles para la ingeniería de seguridad sin tener que prescindir del consabido hardware de evaluación de los sensores o controladores (seguros) al hacer esto.

El entorno de tiempo de ejecución está configurado preferiblemente para integrar y/o para excluir nodos informáticos. Por lo tanto, el entorno de hardware puede variar; el entorno de tiempo de ejecución es capaz de ocuparse de esto y de formar nodos informáticos nuevos o adaptados. En consecuencia, es posible conectar nuevo hardware o sustituir hardware, en particular, para una sustitución ante un fallo (parcial) y para actualizar a una versión superior y para proporcionar recursos informáticos y de memoria adicionales. Las unidades de lógica pueden seguir funcionando en los nodos informáticos abstraídos del entorno de tiempo de ejecución a pesar de una configuración de hardware posible y también brutalmente cambiada.

Al menos una unidad de lógica se configura preferiblemente como una unidad de automatización que genera información relevante para el trabajo de automatización y/o una orden de control para la máquina a partir de los datos de sensor, sin que la información y la orden de control se dirijan a la seguridad. Por lo tanto, el entorno de tiempo de ejecución ayuda a otro tipo de unidad de lógica que proporciona funciones adicionales no dirigidas a la seguridad usando los datos de sensor. Con tal trabajo de automatización, no es una cuestión de protección de personas ni de prevención de accidentes y, en consecuencia, no es necesario satisfacer ninguna norma de seguridad hasta este punto. Un trabajo de automatización típico incluye controles de calidad y funcionamiento, reconocimiento de objetos para agarre y clasificación o para otras etapas de procesamiento, clasificaciones y similares. Una unidad de automatización también se beneficia de esto si el entorno de tiempo de ejecución le asigna recursos flexibles y, a continuación, monitoriza si sigue realizando su trabajo y, por ejemplo, si lo desea, vuelve a iniciar la unidad de lógica correspondiente, la desplaza a un nodo informático diferente o inicia una copia de la unidad de lógica. Sin embargo, es entonces una cuestión de disponibilidad al tiempo que se evitan los tiempos de inactividad y se soportan rutinas apropiadas que son absolutamente muy relevantes para el operador de la máquina, pero que no tienen nada que ver con la seguridad. Es concebible integrar una unidad de automatización en la monitorización de estado y rendimiento de la unidad de diagnóstico, debido a que funciones de automatización fiables pueden proporcionar, de forma similar, un valor añadido, aunque se observe de ese modo un nivel de seguridad que sea posiblemente demasiado alto en este punto.

La unidad de diagnóstico está configurada preferiblemente para determinar en una situación relativa si un mal funcionamiento está relacionado con la seguridad. Los estados de las unidades de función de seguridad existentes o de la rutina de rendimiento pueden evaluarse de forma diferente dependiendo de las circunstancias actuales. La

intrusión de una parte del cuerpo en una zona de trabajo de un robot, por ejemplo, excepcionalmente no representa un riesgo si se asegura simultáneamente que el robot permanezca instantáneamente y de forma segura en una zona de coordenadas restringida que no comprenda el punto de intrusión. Incluso es concebible en tales condiciones relacionadas con la situación que las unidades de función de seguridad y las unidades de automatización cambien dinámicamente sus funciones.

El dispositivo de seguridad tiene preferiblemente una unidad de parada que está configurada para poner la máquina en un estado seguro siguiendo las instrucciones de la unidad de diagnóstico en el caso de un mal funcionamiento relacionado con la seguridad o según las instrucciones de una unidad de función de seguridad ante el reconocimiento de una situación de riesgo con referencia a los datos de sensor evaluados. La unidad de parada o el servicio de parada se encargan por lo tanto de que la máquina esté realmente salvaguardada cuando la unidad de diagnóstico o una función de seguridad lo requiera, preferiblemente mediante una señal correspondiente a la máquina o a su controlador de máquina. Dependiendo de la situación, el estado seguro se logra, por ejemplo, mediante una ralentización, un modo de trabajo especial de la máquina, por ejemplo, con una libertad de movimiento o diversidad de movimientos restringida, una evasión o una detención. La unidad de parada puede implementarse como una unidad de lógica y puede integrarse en un diagnóstico. La unidad de parada preferiblemente recibe regularmente una señal desde la unidad de diagnóstico indicando que todo está en orden y responde igualmente a la ausencia de esta señal con una medida de salvaguardia, tal como en el caso de una demanda de salvaguardia explícita.

El entorno de tiempo de ejecución tiene un sistema de mensajes a través del cual la al menos una unidad de función de seguridad transmite informes de estado e informes de rendimiento a la unidad de diagnóstico. El sistema de mensajes está configurado con dos canales de mensajes para transmitir informes de estado e informes de rendimiento uno al lado del otro. Por lo tanto, hay dos flujos de informes para poder mantener la monitorización de estado y la monitorización de rendimiento separadas entre sí.

La al menos una unidad de función de seguridad está configurada preferiblemente para transmitir regularmente un informe de estado y/o para transmitir un informe de rendimiento de una forma por eventos para una ejecución respectiva de su función de seguridad. De este modo, el estado de la unidad de función de seguridad se monitoriza continuamente con un granulado fino que, en última instancia, se especifica según el nivel de seguridad deseado. Regularmente puede significar cíclicamente, pero es un poco más suave. Es suficiente con que el estado se conozca de nuevo, respectivamente, como muy tarde después de un período de tiempo predeterminado, pero los intervalos de tiempo entre dos informes de estado pueden fluctuar dentro de este marco. A continuación, un informe de rendimiento suministra nueva información si ha tenido lugar un rendimiento mientras tanto, de tal modo que el intercambio de informes de rendimiento puede implementarse de una forma por eventos. Debido a que la función de seguridad se basa en datos de sensor y los propios sensores proporcionan frecuentemente sus datos de forma cíclica, los eventos de evaluación pueden tener lugar cíclicamente, de tal modo que, en última instancia, la secuencia basada en eventos se vuelve, no obstante, cíclica de esta forma indirecta.

El informe de estado y/o el informe de rendimiento preferiblemente tiene(n) una información transmitida acerca de la unidad de función de seguridad de transmisión, una marca de tiempo, una secuencia y/o una suma de comprobación. De este modo, los estados y las ejecuciones pueden asociarse con la unidad de lógica correcta y pueden categorizarse en el tiempo. La secuencia pone los informes o sus contenidos en un orden. Puede asegurarse mediante una suma de comprobación o una medida comparable que el contenido del informe se ha transmitido correctamente.

La al menos una unidad de función de seguridad está configurada preferiblemente para un autodiagnóstico en el que comprueba sus propios datos, programas, resultados de procesamiento y/o la conformidad con un tiempo de sistema. La unidad de función de seguridad puede, en particular, determinar su propio estado a partir de la misma y puede comunicar el mismo en un informe de estado. Una desviación con respecto al tiempo de sistema daría como resultado discrepancias con las marcas de tiempo en los informes y, por lo tanto, posiblemente daría como resultado un diagnóstico de sistema defectuoso. El autodiagnóstico por sí solo no es suficiente para asegurar la seguridad en conjunto, debido a que la unidad de lógica en sí misma no está configurada como segura; pero un autodiagnóstico representa un módulo posible de seguridad.

La unidad de diagnóstico para la monitorización de estado está configurada preferiblemente para invocar una expectativa de estado específica para los estados de la al menos una unidad de función de seguridad, en particular, para modificar la expectativa de estado con referencia a estados, rutinas de trabajo y/o resultados de trabajo previos de las unidades de lógica, y para comparar la expectativa de estado con un estado global actual derivado de los estados de los informes de estado. Por lo tanto, la unidad de diagnóstico tiene la expectativa de estado de un sistema libre de defectos, y esta expectativa de estado puede configurarse, especificarse de otro modo, programarse de forma fija o proporcionarse en una memoria. Una expectativa de estado puede comprender obtener un informe de estado de forma completamente regular desde todas las unidades de función de seguridad existentes o que solo se informe de ciertos estados que no indican un defecto. La expectativa de estado puede adaptarse de una forma relacionada con la situación. La información de estado actual se determina a partir de los informes de estado recibidos y, en particular, se combina en un estado total para comparar el mismo con la expectativa de estado. Una desviación es una indicación de un mal funcionamiento relacionado con la seguridad. En el presente caso, todavía puede haber tolerancias con respecto a ciertas funciones de seguridad y tolerancias temporales. Una desviación que no pueda explicarse por las

tolerancias, la situación actual u otra excepción proporcionada se evalúa preferiblemente como un mal funcionamiento relacionado con la seguridad, tras lo cual la máquina se pone en estado seguro.

El informe de estado proporciona preferiblemente información acerca de si la unidad de función de seguridad que transmite el informe de estado existe, fue capaz de inicializarse por sí misma, si todos sus recursos requeridos, tales como bases de datos, códigos, bibliotecas, recursos informáticos, conexión al sensor, están disponibles y/o si está operativa. Estos son ejemplos para el contenido de un informe de estado o de estados que pueden derivarse del contenido. El informe de estado solo puede ser un resumen, de acuerdo, que incluso puede estar implícito por la mera llegada de un informe. El informe de estado preferiblemente también contiene la información general mencionada anteriormente, tal como el remitente, la marca de tiempo y la suma de comprobación.

La unidad de diagnóstico para la monitorización de rendimiento está configurada preferiblemente para invocar una expectativa de rendimiento específica para la rutina de rendimiento de la al menos una unidad de función de seguridad, en particular, para modificar la expectativa de rendimiento con referencia a estados, rutinas de trabajo y/o resultados de trabajo previos de las unidades de lógica, y para comparar la expectativa de rendimiento con la rutina de rendimiento de trabajo global actual derivada de los informes de rendimiento. Por lo tanto, la unidad de diagnóstico tiene la expectativa de rendimiento de la secuencia temporal y lógica de las ejecuciones de esa al menos una unidad de función de seguridad. Esto se compara con la secuencia de rendimiento propiamente dicha que resulta de los informes de rendimiento. Las desviaciones son indicaciones de un mal funcionamiento relacionado con la seguridad. Como ya es el caso de la monitorización de estado, no todas las desviaciones son necesariamente un mal funcionamiento y no todos los fallos son críticos para la seguridad, con la consecuencia de una respuesta de la máquina relacionada con la seguridad. Preferiblemente, hay tolerancias en la comparación y, como se ha analizado múltiples veces, posiblemente una evaluación relacionada con la situación.

La unidad de diagnóstico está configurada preferiblemente para tener en cuenta al menos uno de los siguientes criterios en una evaluación de la comparación de la expectativa de rendimiento con la rutina de rendimiento derivada de los informes de rendimiento: un orden de ejecución, la ausencia de una ejecución, una ejecución adicional, una desviación de las ejecuciones con respecto a un patrón de tiempo, una duración de ejecución demasiado corta o una duración de ejecución demasiado larga. La desviación con respecto a un patrón de tiempo puede entenderse como un caso especial particularmente relevante de ausencia o adición de un rendimiento. Un número de estos criterios no dan como resultado necesariamente una brecha de monitorización relacionada con la seguridad. Sin embargo, son una indicación de que el sistema se comporta de forma diferente a lo previsto y, si la desviación no se prevé en el concepto de seguridad y, por lo tanto, no se controla de forma segura, la máquina debería conmutarse al estado seguro como una precaución.

El informe de rendimiento tiene preferiblemente información de rendimiento acerca de la última ejecución respectiva de la función de seguridad, en particular, con un tiempo de inicio y/o una duración de ejecución. La duración de ejecución también puede transmitirse naturalmente de forma indirecta, por ejemplo, mediante un tiempo de fin. Preferiblemente, existe la información general, tal como el remitente, la marca de tiempo del informe y/o la suma de comprobación. Si una unidad de función de seguridad realiza una pluralidad de función de seguridad, puede complementarse un número de identificación correspondiente de la función de seguridad. Sin embargo, es preferible que cada unidad de función de seguridad solo sea responsable de una función de seguridad; si se requiere para una función de seguridad adicional, simplemente puede generarse una unidad de función de seguridad adicional. Un informe de rendimiento también puede incluir resultados de rendimiento; por ejemplo, para pruebas dirigidas en las que se hace una comprobación en cuanto a si datos de entrada alimentados como una prueba tales como datos de sensor o datos de sensor emulados, dan como resultado un resultado de rendimiento esperado. Sin embargo, el concepto de monitorización de estado y rendimiento es preferiblemente independiente del contenido específico, lo que a su vez no excluye que tales pruebas tengan lugar adicionalmente, siendo también susceptibles de usarse, para este fin, unidades de diagnóstico de pruebas e informes de prueba separados.

El entorno de tiempo de ejecución tiene preferiblemente un agregador que se dispone lógicamente entre la al menos una unidad de función de seguridad y la unidad de diagnóstico y que está configurado para recibir los informes de rendimiento y para generar la rutina de rendimiento a partir de los mismos con un orden y/o duración de las ejecuciones de las funciones de seguridad de la al menos una unidad de función de seguridad. De este modo, el agregador asume una tarea parcial de la monitorización de rendimiento que, alternativamente, también puede implementarse en la unidad de diagnóstico. Los informes de rendimiento individuales ya se han combinado en la rutina de rendimiento después de la agregación. El agregador funciona preferiblemente en tiempo real; en el presente caso no es solo cuestión de proporcionar datos que tienen indicaciones de cuellos de botella y similares para una optimización manual posterior, sino de una porción de la monitorización de seguridad y, por lo tanto, en última instancia, de la prevención de accidentes.

El al menos un sensor está configurado preferiblemente como un sensor optoelectrónico, en particular, una barrera de luz, un escáner de luz, una rejilla de luz, un escáner láser, un LiDAR FMCW o una cámara, como un sensor de ultrasonidos, un sensor de inercia, un sensor capacitivo, un sensor magnético, un sensor inductivo, un sensor UWB o como un sensor de parámetro de proceso, en particular, un sensor de temperatura, un sensor de flujo directo, un sensor de nivel de llenado o un sensor de presión, teniendo el dispositivo de seguridad, en particular, una pluralidad

de los mismos o diferentes sensores. Estos son algunos ejemplos para sensores que pueden suministrar datos de sensor relevantes para una aplicación de seguridad. La selección específica del sensor o sensores depende de la aplicación de seguridad respectiva. Los sensores pueden configurarse ya como sensores de seguridad. Sin embargo, alternativamente se proporciona explícitamente según la invención para lograr la seguridad solo posteriormente mediante pruebas, sistemas de sensores adicionales o redundancia (diversa), o capacidad de múltiples canales, etc., y para combinar sensores seguros y no seguros con principios de sensores iguales o diferentes entre sí. Un sensor que haya fallado, por ejemplo, no suministraría ningún dato de sensor; esto se reflejaría en los informes de estado y rendimiento de la unidad de función de seguridad responsable del sensor y, por lo tanto, sería apreciado por la unidad de diagnóstico en la monitorización de estado y rendimiento.

El método según la invención puede desarrollarse adicionalmente de una forma similar y muestra ventajas similares al hacer esto. Tales características ventajosas se describen de forma ilustrativa, pero no exclusiva, en las reivindicaciones subordinadas que dependen de las reivindicaciones independientes.

La invención se explicará con más detalle a continuación también con respecto a otras características y ventajas a modo de ejemplo con referencia a las realizaciones y al dibujo adjunto. Las Figuras del dibujo muestran en:

la Figura 1 una ilustración de visión general de un dispositivo de seguridad;

la Figura 2 una representación esquemática de un entorno de tiempo de ejecución del dispositivo de seguridad;

la Figura 3 una representación esquemática de un entorno de tiempo de ejecución a modo de ejemplo usando dos nodos informáticos;

la Figura 4 una representación esquemática de un entorno de tiempo de ejecución similar al de la Figura 3 en una realización especial que usa Kubernetes;

la Figura 5 una representación esquemática del flujo de informes doble con informes de estado y rendimiento a una unidad de diagnóstico de sistema;

la Figura 6 una representación esquemática de una monitorización de estado basándose en los informes de estado; y

la Figura 7 una representación esquemática de una monitorización de rendimiento basándose en los informes de rendimiento.

La Figura 1 muestra una representación de visión general de un dispositivo 10 de seguridad. Los términos seguridad y seguro e inseguro han de entenderse aún de tal modo que componentes, rutas de transmisión y evaluaciones correspondientes satisfacen o no satisfacen los criterios de normas de seguridad mencionadas en la introducción.

El dispositivo 10 de seguridad puede dividirse aproximadamente en tres bloques que tienen al menos una máquina 12 que va a monitorizarse, al menos un sensor 14 para generar datos de sensor de la máquina 12 monitorizada y al menos un componente 16 de hardware con recursos informáticos y de memoria para la funcionalidad de control y evaluación para evaluar los datos de sensor y desencadenar cualquier respuesta dirigida a la seguridad de la máquina 12. La máquina 12, el sensor 14 y el componente 16 de hardware se abordan a veces en singular y a veces en plural a continuación, lo que debería incluir explícitamente las otras realizaciones respectivas con solo una unidad 12, 14, 16 respectiva o una pluralidad de tales unidades 12, 14, 16.

En los márgenes se muestran ejemplos respectivos para los tres bloques. La máquina 12 preferiblemente usada industrialmente es, por ejemplo, una máquina de procesamiento, una línea de producción, una instalación de clasificación, una instalación de procesamiento, un robot o un vehículo que puede ir sobre rieles o no y que, en particular, es un carro guiado automatizado sin conductor (AGC); AGV, vehículo guiado automatizado; AMR, robot móvil autónomo).

Un escáner láser, una rejilla de luz y una cámara estereoscópica como representantes de sensores optoelectrónicos se muestran como sensores 14 ilustrativos que incluyen sensores adicionales tales como sensores de luz, barreras de luz, LiDAR FMVW o cámaras que tienen cualquier detección 2D o 3D, tal como procesos de proyección o procesos de tiempo de vuelo. Algunos ejemplos para los sensores 14 que aún no son exclusivos son los sensores UWB, los sensores de ultrasonidos, los sensores de inercia, los sensores capacitivos, magnéticos o inductivos, o sensores de parámetros de proceso tales como sensores de temperatura, sensores de flujo directo, sensores de nivel de llenado o sensores de presión. Estos sensores 14 pueden estar presentes en cualquier número deseado y pueden combinarse entre sí de cualquier forma deseada dependiendo del dispositivo 10 de seguridad.

Los componentes 16 de hardware concebibles incluyen controladores (PLC, controladores lógicos programables), un procesador en una red local, en particular, un dispositivo perimetral, o una nube separada o una nube operada por otros y, en general, cualquier hardware que proporcione recursos para el procesamiento de datos digitales.

Los tres bloques se capturan de nuevo en el interior de la Figura 1. La máquina 12 está conectada preferiblemente al dispositivo 10 de seguridad a través de su controlador 18 de máquina, siendo el controlador de máquina un controlador de robot en el caso de un robot, un controlador de vehículo en el caso de un vehículo, un controlador de proceso en una instalación de procesamiento y similares para otras máquinas 12. Los sensores 14 combinados en el interior como un bloque 20 no solo generan datos de sensor, sino que también tienen una interfaz, que no se muestra individualmente, para emitir los datos de sensor en forma bruta o (pre) procesada y, por regla general, tienen una unidad de control y evaluación separada, es decir, un componente de hardware separado para el procesamiento digital de datos.

Un entorno 22 de tiempo de ejecución es un término resumido para una unidad de procesamiento que, entre otras cosas, realiza el procesamiento de datos de los datos de sensor para obtener órdenes de control para la máquina 13 u otra información relacionada con la seguridad y adicional. El entorno 22 de tiempo de ejecución se implementa en los componentes 16 de hardware y se explicará con más detalle a continuación con referencia a las Figuras 2 a 4. El hardware en el que se ejecuta el entorno 22 de tiempo de ejecución no es fijo según la invención. La lista anterior de componentes de hardware posibles menciona algunos ejemplos que pueden combinarse según se desee. Además, el entorno 22 de tiempo de ejecución se dibuja de forma deliberada con una superposición con el controlador 18 de máquina y el bloque 20 de los sensores 14, debido a que los recursos informáticos y de memoria internos de los sensores 14 y/o de la máquina 12 también pueden ser usados por el entorno 22 de tiempo de ejecución, de nuevo en cualquier combinación deseada, incluyendo la posibilidad de que no haya ningún componente 16 de hardware adicional en absoluto fuera de la máquina 12 y los sensores 14. Se supone en lo sucesivo que los componentes 16 de hardware proporcionan los recursos de procesamiento y memoria, de tal modo que también se entiende entonces una inclusión de hardware interno de la máquina 12 y/o los sensores 14.

El dispositivo 10 de seguridad y, en particular, el entorno 22 de tiempo de ejecución proporcionan ahora funciones de seguridad y funciones de diagnóstico. Una función de seguridad acepta el flujo de información de medición y eventos con los datos de sensor sucediéndose unos a otros en el tiempo y genera resultados de evaluación correspondientes, en particular, en forma de señales de control para la máquina 12. Además, puede adquirirse información de autodiagnóstico, información de diagnóstico de un sensor 4 o información de visión general. Las funciones de diagnóstico propiamente dichas mediante las cuales se designa la monitorización de una función de seguridad en el marco de esta descripción, como se explicará en detalle a continuación con referencia a las Figuras 5 a 7, deben distinguirse de estas. Las funciones de automatización inseguras son concebibles como una opción adicional además de estas funciones relacionadas con la seguridad o funciones de automatización seguras.

El dispositivo 10 de seguridad logra una alta disponibilidad y robustez con respecto a eventos internos y externos imprevistos en el sentido de que las funciones de seguridad se realizan como servicios de los componentes 16 de hardware. La composición flexible de los componentes 16 de hardware y, preferiblemente, su conexión de red en la red local o no local o en una nube posibilitan una redundancia y una elasticidad de rendimiento, de tal modo que las interrupciones, las perturbaciones y los picos de demanda pueden abordarse de forma muy robusta. El dispositivo 10 de seguridad reconoce tan pronto como los defectos ya no pueden interceptarse y, por lo tanto, pasa a estar orientado a la seguridad y, a continuación, inicia una respuesta apropiada para la situación en la que la máquina 12 pasa a un estado seguro según se requiera. Para este fin, la máquina 12, por ejemplo, se detiene, se ralentiza, se evade o funciona en un modo no arriesgado. Una vez más, debe dejarse claro que hay dos clases de eventos que pueden desencadenar una respuesta orientada a la seguridad: por un lado, un evento que se clasifica como arriesgado y que resulta de los datos de sensor y, por otro lado, la revelación de un defecto dirigido a la seguridad.

La Figura 2 muestra una representación esquemática del entorno 22 de tiempo de ejecución. En última instancia, el objeto del entorno 22 de tiempo de ejecución es derivar una orden de control a partir de los datos de sensor, en particular, una señal de seguridad que desencadena una respuesta dirigida a la seguridad de la máquina 12. El entorno 22 de tiempo de ejecución tiene un maestro 24 y al menos un nodo informático 26. Los componentes 26 de hardware proporcionan la capacidad de procesamiento y memoria requerida para el maestro 24 y los nodos informáticos 26; el entorno 22 de tiempo de ejecución puede extenderse de forma transparente a lo largo de una pluralidad de componentes 16 de hardware. En el presente caso, un nodo informático 26 ha de entenderse de forma abstracta o virtual; no existe necesariamente una relación 1:1 entre un nodo informático 26 y un componente 16 de hardware, sino que un componente 16 de hardware puede proporcionar más bien una pluralidad de nodos 26 o, a la inversa, un nodo informático 26 puede desplegarse a lo largo de una pluralidad de componentes 16 de hardware. El despliegue se aplica de forma análoga al maestro 24.

Un nodo informático 26 tiene una o más unidades 28 de lógica. Una unidad 28 de lógica es una unidad funcional que es cerrada en sí misma, que acepta información, la coteja, la transforma, la reformula o, en general, la procesa para dar nueva información y, a continuación, la facilita a consumidores posibles como una orden de control o para su procesamiento adicional, en particular, para otras unidades 28 de lógica o para un controlador 12 de máquina. Dentro del marco de esta descripción deben distinguirse principalmente tres tipos de unidades 28 de lógica que ya se han abordado brevemente, en concreto, unidades de función de seguridad, unidades de diagnóstico y, opcionalmente, unidades de automatización que no contribuyen a la seguridad, pero sí posibilitan la integración de otros trabajos de automatización en la aplicación total.

El entorno 22 de tiempo de ejecución activa las unidades 28 de lógica requeridas respectivas y prevé su funcionamiento correcto. Para ello, asigna los recursos requeridos en los nodos informáticos 26 o componentes 26 de hardware disponibles a las respectivas unidades 28 de lógica y monitoriza la actividad y el requerimiento de recursos de todas las unidades 28 de lógica. El entorno 22 de tiempo de ejecución reconoce preferiblemente cuando una unidad 28 de lógica ya no está activa o cuando han tenido lugar interrupciones en el entorno 22 de tiempo de ejecución o en la unidad 28 de lógica. A continuación, intenta reactivar la unidad 28 de lógica y genera una nueva copia de la unidad 28 de lógica si esto no es posible para mantener por lo tanto un funcionamiento apropiado. Sin embargo, este es un mecanismo que no satisface las exigencias de seguridad funcional y solo tiene efecto si el diagnóstico de sistema que aún ha de explicarse con referencia a las Figuras 5 a 7 no ha descubierto previamente un defecto relacionado con la seguridad o, por ejemplo, durante una fase de inicialización o una fase de reinicio en la que la máquina 12 sigue de todos modos en reposo.

Las interrupciones pueden ser previstas o imprevistas. Causas ilustrativas son defectos en la infraestructura, es decir, en los componentes 16 de hardware, su sistema operativo o las conexiones de red; además de operaciones o manipulaciones incorrectas accidentales o el consumo completo de los recursos de un componente 16 de hardware. Si una unidad 28 de lógica no puede procesar toda la información requerida, en particular, la información dirigida a la seguridad, o al menos no puede procesar la misma con la suficiente rapidez, el entorno 22 de tiempo de ejecución puede preparar copias adicionales de la unidad 28 de lógica respectiva para asegurar adicionalmente el procesamiento de la información. El entorno 22 de tiempo de ejecución de esta forma prevé que la unidad 28 de lógica produzca su función con la calidad y disponibilidad esperadas. Según las observaciones en el párrafo previo, tales medidas de reparación y modificación tampoco son ninguna sustitución para el diagnóstico de sistema que aún no se ha descrito.

La Figura 3 muestra de nuevo otra realización ventajosamente completamente diferenciada del entorno 22 de tiempo de ejecución del dispositivo 10 de seguridad. El maestro 24 forma el centro de gestión y comunicación. La información de configuración o un archivo de configuración en las unidades 28 de lógica presentes se almacenan en el mismo de tal modo que el maestro 24 tiene el conocimiento requerido de la configuración, en particular, qué unidades 28 de lógica existen y deberían existir, en qué nodos informáticos 26 pueden hallarse y en qué intervalo de tiempo recibieron recursos y se invocan. El archivo de configuración se protege preferiblemente a través de firmas contra manipulaciones deliberadas y no deliberadas, por ejemplo, a través de tecnologías de cadena de bloques. En el presente caso, la ingeniería de seguridad (seguridad) une sus fuerzas de forma ventajosa con la integridad de datos (ciberseguridad), debido a que los ataques son rechazados o al menos reconocidos de esta forma, lo que podría dar como resultado consecuencias de accidentes imprevisibles.

Los nodos informáticos 26 tienen ventajosamente su propia subestructura, y las unidades ahora descritas también pueden estar presentes solo en parte. Inicialmente, los nodos informáticos 26 pueden dividirse de nuevo en subnodos 30. El número mostrado de nodos informáticos 26, teniendo cada uno dos subnodos 30, es puramente ilustrativo; puede haber tantos nodos informáticos 26, cada uno con cualquier número deseado de subnodos 30 según se requiera, siendo capaz el número de subnodos 30 de variar a lo largo de los nodos informáticos 26. Las unidades 28 de lógica se generan preferiblemente solo dentro de los subnodos 30, no ya en el nivel de los nodos informáticos 26. Preferiblemente, las unidades 28 de lógica se virtualizan, es decir, se contenedorizan, dentro de contenedores. Por lo tanto, cada subnodo 30 tiene uno o más contenedores, preferiblemente con una unidad 28 de lógica respectiva. En lugar de las unidades 38 de lógica genéricas, en la Figura 3 se muestran los tres tipos ya abordados de unidades 28 de lógica, en concreto, dos unidades 32 de función de seguridad, una unidad 34 de diagnóstico y una unidad 36 de automatización. El tipo y el número de las unidades 28 de lógica son solo un ejemplo, y el diagnóstico de sistema aún se explicará con referencia a las Figuras 5 a 7, preferiblemente gestionándolo con una sola unidad 34 de diagnóstico. La asociación de las unidades 38 de lógica con los subnodos 30 y los nodos informáticos 26 es completamente independiente de la estructura lógica y de la cooperación de las unidades 28 de lógica. Por lo tanto, no puede sacarse ninguna conclusión en absoluto acerca de la cooperación de contenido a partir de la disposición de las unidades 28 de lógica, de todos modos, solo se muestra a modo de ejemplo; cualquier redistribución deseada sería posible con una funcionalidad que sea completamente la misma; el entorno de tiempo de ejecución prevé esto.

Una unidad 38 de gestión de nodo del nodo informático 26 coordina sus subnodos 30 y las unidades 28 de lógica asignadas a este nodo informático 26. La unidad 38 de gestión de nodo se comunica además con el maestro 24 y con otros nodos informáticos 26. El trabajo de gestión del entorno 22 de tiempo de ejecución puede desplegarse prácticamente como se desee en la unidad maestra 24 y en la unidad 38 de gestión de nodo; por lo tanto, el maestro puede considerarse como implementado de una forma desplegada. Sin embargo, es ventajoso que el maestro se ocupe del trabajo global del entorno 22 de tiempo de ejecución y que cada unidad 38 de gestión de nodo se ocupe del trabajo local del nodo informático 26 respectivo. No obstante, el maestro 24 puede formarse preferiblemente en una pluralidad de componentes 16 de hardware de una forma desplegada o redundante para aumentar su seguridad frente a fallos.

El ejemplo típico para la función de seguridad de una unidad 32 de función de seguridad es la evaluación relacionada con la seguridad de datos de sensor del sensor 14. En el presente caso, los ejemplos típicos son, entre otras cosas, la monitorización de la distancia (específicamente la velocidad y la separación), la monitorización de pasajes, la monitorización de campos protegidos o la prevención de colisiones con el objetivo de una respuesta apropiada dirigida a la seguridad de la máquina 12 en un caso arriesgado. Esta es la tarea principal de la ingeniería de seguridad, debido

a que son posibles las rutas más variadas para distinguir entre una situación normal y una arriesgada dependiendo del sensor 14 y del proceso de evaluación. Las unidades 32 de función de seguridad adecuadas pueden programarse para cada una de las aplicaciones de seguridad o grupo de aplicaciones de seguridad o pueden seleccionarse de entre una agrupación de unidades 32 de función de seguridad existentes. Si el entorno de trabajo 22 genera un módulo 32 de función de seguridad, esto no significa entonces, en modo alguno, que la función de seguridad vaya a recrearse de este modo. Más bien, se hace uso de bibliotecas correspondientes o programas terminados dedicados de una forma conocida tal como por medio de soportes de datos, memorias o una conexión de red. Es concebible que una función de seguridad se ensamble y/o se configure adecuadamente de forma semiautomática o automática a partir de un kit.

Una unidad 34 de diagnóstico puede entenderse en el sentido del documento EP 4 040 034 A1 mencionado en la introducción y puede actuar como un organismo de control o puede llevar a cabo pruebas y diagnósticos de diferente complejidad. De este modo, los algoritmos seguros y las medidas de autocontrol de una unidad 32 de función de seguridad pueden sustituirse o complementarse al menos parcialmente. Para este fin, la unidad 34 de diagnóstico tiene expectativas para la salida de la unidad 32 de función de seguridad en momentos específicos, o bien en su funcionamiento regular o bien en respuesta a información de sensor artificial específica alimentada como una prueba. Según la invención, se usa una unidad 34 de diagnóstico que no prueba las unidades 32 de función de seguridad individuales o no espera un resultado de evaluación específico de las mismas, incluso esto es posible de forma complementaria, sino que realiza un diagnóstico de sistema de los módulos 32 de función de seguridad individuales implicados en la salvaguarda de la máquina 12, como se explicará a continuación con referencia a las Figuras 5 a 7.

Una unidad 36 de automatización es una unidad 28 de lógica para trabajos de automatización no relacionados con la seguridad que monitoriza los sensores 14 y las máquinas 12 o partes de las mismas, generalmente accionadores, y que controla rutinas (parciales) en función de esta información o proporciona información acerca de los mismas. Una unidad 36 de automatización es tratada, en principio, por el entorno de tiempo de ejecución como lo es cada unidad 23 de lógica, por lo tanto, preferiblemente se contenedoriza de forma similar. Los ejemplos para un trabajo de automatización incluyen una comprobación de calidad, control de variantes, reconocimiento de objetos para agarre y clasificación o para otras etapas de procesamiento, clasificaciones y similares. La delimitación de las unidades 28 de lógica dirigidas a la seguridad, es decir, a partir de una unidad 32 de función de seguridad o una unidad 34 de diagnóstico, comprende una unidad 36 de automatización que no contribuye a la prevención de accidentes, es decir, a la aplicación dirigida a la seguridad. Se desea un funcionamiento fiable y una cierta monitorización por el entorno 22 de tiempo de ejecución, pero esto contribuye a aumentar la disponibilidad y, por lo tanto, la productividad y la calidad, pero no la seguridad. Naturalmente, esta fiabilidad también puede establecerse en el sentido de que una unidad 36 de automatización se monitoriza tan cuidadosamente como una unidad 32 de función de seguridad, de tal modo que es posible, pero no absolutamente necesario.

Mediante el uso del entorno 22 de tiempo de ejecución, se vuelve posible desplegar unidades 28 de lógica para una aplicación de seguridad prácticamente de cualquier forma deseada a lo largo de un entorno, también un entorno muy heterogéneo, de los componentes 26 de hardware, incluyendo una red perimetral o una nube. El entorno 22 de tiempo de ejecución se ocupa de todos los recursos y condiciones requeridos de las unidades 28 de lógica. Invoca las unidades 28 de lógica requeridas, las finaliza o las desplaza entre los nodos informáticos 26 y los subnodos 30.

La arquitectura del entorno 22 de tiempo de ejecución permite adicionalmente una fusión perfecta de seguridad y automatización, debido a que las unidades 32 de función dirigida a la seguridad, las unidades 34 de diagnóstico y las unidades 36 de automatización pueden ejecutarse en el mismo entorno y prácticamente simultáneamente y pueden tratarse de la misma forma. En el caso de un conflicto, el entorno 22 de tiempo de ejecución preferiblemente da prioridad a la unidad de las unidades 32 de función de seguridad y a las unidades 34 de diagnóstico, por ejemplo, en el caso de escasez de recursos. Las reglas de rendimiento para la coexistencia de las unidades 28 de lógica relevantes de los tres tipos diferentes pueden tenerse en cuenta en el archivo de configuración.

La Figura 4 muestra una representación esquemática de un entorno 22 de tiempo de ejecución en una realización que usa Kubernetes. En el presente caso, el entorno 22 de tiempo de ejecución se denomina plano de control. La Figura 4 se basa en la Figura 3, en la que se ha omitido un nodo informático 26 por motivos de claridad y ahora se muestran las unidades 38 de lógica genéricas como representativas de los tres tipos posibles. El maestro 24 tiene una subestructura en Kubernetes. El maestro 24 (de Kubernetes) todavía no es responsable por sí mismo del diseño de los contenedores o las unidades 28 de lógica, sino que se encarga de las rutinas generales o de la orquestación (capa de orquestación). En consecuencia, el archivo de configuración se denomina archivo de orquestación. Además, están presentes un etcd 40 de datos para todos los datos relevantes del entorno de Kubernetes, un servidor 24 de API como una interfaz con Kubernetes y un gestor 44 de planificación y controlador que lleva a cabo la orquestación propiamente dicha.

El hardware presente se divide en nodos como nodos informáticos 26. A su vez, hay una o más de las así denominadas cápsulas, debido a que los subnodos 30 en los nodos y el contenedor que tiene los microservicios propiamente dichos están en los mismos, en este caso las unidades 28 de lógica junto con el tiempo de ejecución de contenedor asociado y, por lo tanto, todas las bibliotecas y dependencias requeridas para la unidad 28 de lógica en el tiempo de ejecución. Una unidad 38 de gestión de nodo ahora dividida en dos realiza la gestión local con un así denominado Kubelet 38a

y un intermediario 38b. El Kubelet 38a es un agente que gestiona las cápsulas y contenedores separados de los nodos. El intermediario 38b, a su vez, incluye las reglas de red para la comunicación entre los nodos y con el maestro.

Kubernetes es una opción de implementación preferida, pero en modo alguno la única, para el entorno 22 de tiempo de ejecución. Podría mencionarse un enjambre de cargadores como una alternativa adicional entre muchas. El cargador en sí no es una alternativa directa, sino más bien una herramienta para producir contenedores y, por lo tanto, es combinable con Kubernetes y un enjambre de cargadores que orquesta a continuación los contenedores.

La Figura 5 muestra una representación esquemática adicional del entorno 22 de tiempo de ejecución para ilustrar un diagnóstico de sistema mediante una monitorización de estado y una monitorización de rendimiento. Una unidad 34 de diagnóstico de sistema es responsable de esto como una realización especial de una unidad 34 de diagnóstico. Tres unidades 28 de lógica que funcionan en secuencia para evaluar los datos de un sensor 14 son en el presente caso puramente a modo de ejemplo para la monitorización. La invención no se limita a esto; puede haber cualquier número deseado de unidades 28 de lógica en cualquier conexión mutua deseada y con o sin su propia conexión a un sensor 14. Las unidades 28 de lógica son preferiblemente unidades 32 de función de seguridad. Pueden proporcionarse unidades 34 de diagnóstico adicionales que, por ejemplo, supervisan o someten a prueba unidades de función de seguridad específicas de una forma dedicada complementaria al diagnóstico de sistema. Además, también es posible integrar unidades 36 de automatización en el diagnóstico de sistema, incluso si para ello se requiriese de por sí una monitorización segura para la prevención de riesgos o la prevención de accidentes. No es el diseño específico de las unidades 28 de lógica lo que es importante a continuación de tal modo que se muestran las unidades 28 de lógica genéricas.

La unidad 34 de diagnóstico de sistema es responsable de una monitorización 46 de estado y de una monitorización 48 de rendimiento. De ello puede derivarse una evaluación final del estado seguro del sistema total. La monitorización 46 de estado se explicará posteriormente con incluso más detalle con referencia a la Figura 6; la monitorización 48 de rendimiento con referencia a la Figura 7. En la Figura 5 se muestra una única unidad 34 de diagnóstico de sistema que tiene sus propios bloques para la monitorización 46 de estado y la monitorización 48 de rendimiento. Sirve sobre todo para la comprensión del concepto; es igualmente concebible entender la monitorización 46 de estado y la monitorización de rendimiento como parte de la unidad 34 de diagnóstico de sistema o, alternativamente, desplegar la funcionalidad de una forma diferente.

Las unidades 28 de lógica se comunican con la unidad 34 de diagnóstico de sistema a lo largo de un sistema de mensajes o un sistema de transmisión de informes. El sistema de mensajes forma parte del entorno 22 de tiempo de ejecución o se implementa como complementario al mismo. Hay un flujo de informes doble a partir de los informes 50 de estatus o informes de estado de la monitorización 46 de estado que proporcionan información acerca del estado interno de la unidad 28 de lógica de envío y los informes 52 de rendimiento de la monitorización 52 de rendimiento que proporcionan información acerca de las demandas de servicio o los tiempos de ejecución del servicio de las unidades 28 de lógica de envío. En consecuencia, el sistema de mensajes se proporciona en forma doble o está configurado con dos canales de mensajes. Cada informe 50, 52 comprende preferiblemente metadatos que salvaguardan el flujo del informe. Estos metadatos, por ejemplo, comprenden información de transmisión, una marca de tiempo, información de secuencia y/o una suma de comprobación sobre el contenido del informe.

La unidad 34 de diagnóstico de sistema determina un estado global del dispositivo 10 de seguridad basándose en los informes 50 de estado obtenidos y, en consecuencia, en una declaración global acerca del procesamiento de las demandas de servicio o acerca de una rutina de tiempo de ejecución del dispositivo 10 de seguridad a partir de los informes 52 de rendimiento obtenidos. Los defectos en el dispositivo 10 de seguridad se descubren mediante una comparación con las expectativas asociadas y se inicia una respuesta apropiada relacionada con la seguridad en el caso de un defecto.

No todas las irregularidades significan inmediatamente un defecto relacionado con la seguridad. De este modo, pueden tolerarse las desviaciones durante un tiempo determinado dependiendo del nivel de seguridad o se intenta que los mecanismos de reparación vuelvan a pasar a un estado de sistema libre de defectos. Sin embargo, en el presente caso el concepto de seguridad especifica exactamente el tiempo y otros marcos en los que solo puede realizarse una observación. Además, puede haber grados de defectos que requieran medidas de salvaguardia y evaluaciones de defectos diversamente drásticas debido a la situación. Esto último da como resultado una comprensión diferenciada de la seguridad y la protección que incluye la situación actual. El fallo de un componente relacionado con la seguridad o la no ejecución de una función relacionada con la seguridad pueden no significar necesariamente aún un estado de sistema inseguro bajo ciertos requisitos, es decir, debido a la situación. Podría haber fallado, por ejemplo, un sensor 14 que monitoriza una zona de colaboración con un robot mientras el robot definitivamente no reside en esta zona, lo que a su vez puede asegurarse mediante la delimitación de coordenadas segura del propio robot. Sin embargo, tales reglas relacionadas con la situación para la evaluación de si ha de tener lugar una respuesta relacionada con la seguridad deben ser conocidas entonces, de forma similar, por la unidad 34 de diagnóstico de sistema de una forma coordinada con el concepto de seguridad.

La respuesta relacionada con la seguridad de la máquina 12 es desencadenada preferiblemente por un servicio 54 de parada. Puede ser una unidad 32 de función de seguridad adicional que puede integrarse preferiblemente en la

monitorización del sistema, al contrario de lo que se representa. El servicio 54 de parada funciona preferiblemente de una forma invertida, es decir, se espera una señal positiva desde la unidad 34 de diagnóstico de sistema y se envía a la máquina 12 para que la máquina 12 pueda funcionar. De este modo, se contiene automáticamente un fallo de la unidad 34 de diagnóstico de sistema o del servicio 54 de parada.

A pesar de su nombre, la máquina no es parada necesariamente por el servicio 54 de parada, esta es solo la medida más drástica. Dependiendo del defecto, ya puede lograrse un estado seguro mediante una desaceleración, una restricción de la velocidad y/o del espacio de trabajo, o similares. Esto tiene entonces menos efectos sobre la productividad. El servicio 54 de parada también puede ser requerido por una de las unidades 28 de lógica si se ha reconocido allí una situación de riesgo evaluando los datos de sensor. La flecha correspondiente se omitió por motivos de claridad en la Figura 5.

La Figura 6 muestra una representación esquemática de la monitorización 46 de estado basándose en los informes 50 de estado. Los informes 50 de estado se comunican preferiblemente continua o regularmente a la unidad de diagnóstico de sistema en el sensor indicando que ha de llegar un informe 50 de estado desde cada unidad 28 de lógica monitorizada como muy tarde después de un tiempo fijo de, por ejemplo, algunos milisegundos. En el presente caso es concebible un ciclo fijo o un ciclo de tiempo, pero no se requiere, por lo que son posibles fluctuaciones de tiempo dentro del marco de la duración fija.

Las unidades 28 de lógica llevan a cabo preferiblemente un autodiagnóstico antes de la transmisión de un informe 50 de estado. Este no es necesariamente el caso en cada una de las realizaciones; un informe 50 de estado puede ser solo una señal de vida o la transmisión de estados internos sin un autodiagnóstico previo o el autodiagnóstico se lleva a cabo con menos frecuencia que la que se transmiten los informes 50 de estado. El autodiagnóstico, por ejemplo, comprueba los datos y los elementos de programa almacenados en su memoria, los resultados de procesamiento y el tiempo de sistema. Los informes 50 de estado contienen, de forma correspondiente, información acerca del estado interno de la unidad 28 de lógica y proporcionan información acerca de si la unidad de lógica puede realizar su trabajo correctamente, por ejemplo, si la unidad 28 de lógica tiene todos los datos requeridos disponibles en un tiempo suficiente. Además, los informes 50 de estado comprenden preferiblemente los metadatos mencionados anteriormente.

La unidad 34 de diagnóstico de sistema interpreta el contenido de los informes 50 de estado y los asocia con las unidades 28 de lógica respectivas. Los estados individuales de las unidades 28 de lógica se combinan para formar un estado total del dispositivo 10 de seguridad desde el punto de vista de la seguridad. La unidad 34 de diagnóstico de sistema tiene una expectativa predefinida en cuanto a qué estado total asegura la seguridad en qué situación. Si esta comparación con el estado total actual muestra que esta expectativa no se ha cumplido, si bien es posible tener en cuenta las tolerancias y las adaptaciones basadas en la situación ya analizadas, es, por lo tanto, un defecto relacionado con la seguridad. Se envía un informe correspondiente al servicio 54 de parada para poner la máquina 12 en un estado seguro apropiado para el defecto.

La Figura 7 muestra una representación esquemática de la monitorización 48 de rendimiento basándose en los informes 52 de rendimiento. La primera unidad 28 de lógica dentro de un servicio genera una caracterización inequívoca de la secuencia del programa, en resumen, una secuencia a la que todas las unidades 28 de lógica implicadas de un servicio hacen referencia en su rendimiento. La secuencia se propaga a las unidades 28 de lógica directamente siguientes después de finalizar la ejecución respectiva, de tal modo que puede relacionarse con la misma en la preparación de su informe 52 de rendimiento. Un informe 52 de rendimiento comprende, preferiblemente además de los metadatos mencionados anteriormente, un tiempo de inicio y una duración de la ejecución respectiva. Otros posibles componentes de un informe 52 de rendimiento son una descripción única de lo que se llevó a cabo y de la secuencia. Los informes 52 de rendimiento se envían preferiblemente de una forma por evento en cada caso después de una ejecución completa. Debido a que los datos de sensor que están disponibles cíclicamente se evalúan frecuentemente, un sistema de informes basado en eventos también puede ser indirectamente cíclico o regular en el sentido definido anteriormente.

Un agregador 56 recopila los informes 52 de rendimiento y cambia los rendimientos en una disposición lógica y temporal o en una rutina de tiempo de ejecución con referencia a la caracterización única de la secuencia del programa. Por lo tanto, la rutina de tiempo de ejecución describe los tiempos de ejecución propiamente dichos. La unidad 34 de diagnóstico de sistema, por otro lado, tiene acceso a una expectativa 57 de tiempo de ejecución, es decir, una rutina de tiempo de ejecución esperada. Esta expectativa 58 de tiempo de ejecución es una especificación que un experto en seguridad ha fijado típicamente en relación con el concepto de seguridad, pero que aún puede ser modificada por la unidad 34 de diagnóstico de sistema dependiendo de la realización. Si la unidad 34 de diagnóstico de sistema no debiera tener acceso a la expectativa 58 de tiempo de ejecución, es al menos un defecto relacionado con la seguridad, al menos después de una tolerancia temporal, con la consecuencia de que se solicita al servicio 54 de parada que salvaguarde la máquina 12. El agregador 56 y la expectativa 58 de tiempo de ejecución se muestran por separado y se implementan preferiblemente de esta forma, pero alternativamente pueden entenderse como parte de la unidad 34 de diagnóstico de sistema.

La unidad 34 de diagnóstico de sistema compara ahora la rutina de tiempo de ejecución comunicada por el agregador 56 con la expectativa 58 de tiempo de ejecución como parte de la monitorización 48 de rendimiento para reconocer defectos lógicos y de tiempo en el procesamiento de una demanda de servicio. Ante irregularidades, pueden iniciarse etapas para la estabilización o la máquina se salvaguarda a través del servicio 54 de parada tan pronto como un defecto ya no pueda controlarse de forma inequívoca.

Algunos ejemplos para aspectos comprobados de la monitorización 48 de rendimiento son: no hay tiempo de ejecución para trabajar completamente a través de un servicio; se informó de un tiempo de ejecución adicional inesperado, o bien un tiempo de ejecución múltiple inesperado de una unidad 28 de lógica implicada en el servicio o bien un tiempo de ejecución de una unidad 28 de lógica no implicada en el servicio; un tiempo de ejecución es demasiado corto o demasiado largo y, de hecho, junto con la cuantificación para la evaluación de si es grave; el tiempo transcurrido entre los tiempos de ejecución de los tiempos de ejecución individuales de la unidad 28 de lógica. Cuáles de estas irregularidades están relacionadas con la seguridad, en qué marco y en qué situación pueden seguir tolerándose y qué medida de salvaguardia apropiada se inicia respectivamente se almacena en la expectativa 58 de tiempo de ejecución o en la unidad 34 de diagnóstico de sistema.

REIVINDICACIONES

1. Un dispositivo (10) de seguridad para monitorizar al menos una máquina (12), comprendiendo el dispositivo (10) de seguridad al menos un sensor (14) para generar datos de sensor en relación con la máquina (12) y una unidad (16, 22) de procesamiento para los datos de sensor, que está conectada al menos indirectamente al sensor (14) y a la máquina (12) y está configurada como un entorno (22) de tiempo de ejecución con al menos un nodo informático (26) y para ejecutar una pluralidad de unidades (28) de lógica en el al menos un nodo informático (26), estando configurada al menos una unidad (28) de lógica como una unidad (32) de función de seguridad para una evaluación orientada a la seguridad de los datos de sensor y estando configurada al menos una unidad (28) de lógica como una unidad (34) de diagnóstico para monitorizar la al menos una unidad (32) de función de seguridad,
caracterizado por que la al menos una unidad (32) de función de seguridad está configurada para transmitir mensajes (50) de estado y mensajes (52) de ejecución a la unidad (34) de diagnóstico, en donde mensajes (50) de estado proporcionan información acerca del estado de la al menos una unidad (32) de función de seguridad y, por lo tanto, acerca de su disponibilidad operativa y restricciones o errores posibles, y en donde un mensaje (52) de ejecución se refiere a la ejecución de una función de seguridad o un servicio que la al menos una unidad de función (32) ejecuta, a partir del cual se genera una secuencia de ejecución de las funciones o servicios de seguridad ejecutados, por que la unidad (34) de diagnóstico está configurada para detectar un mal funcionamiento relevante para la seguridad del dispositivo (10) de seguridad en una monitorización (46) de estado usando estados a partir de los mensajes (50) de estado y en una monitorización (48) de ejecución usando la secuencia de ejecución generada a partir de los mensajes (52) de ejecución, y por que el entorno (22) de tiempo de ejecución comprende un sistema de mensajes con dos canales de mensajes, a través del cual la al menos una unidad (32) de función de seguridad transmite mensajes (50) de estado y mensajes (52) de ejecución en paralelo a la unidad de diagnóstico.

2. El dispositivo (10) de seguridad según la reivindicación 1,
en donde la unidad (34) de diagnóstico está configurada para determinar, dependiendo de la situación, si un mal funcionamiento es relevante para la seguridad.

3. El dispositivo (10) de seguridad según la reivindicación 1 o 2,
que comprende una unidad (54) de apagado configurada para transferir la máquina (12) a un estado seguro ante la instrucción de la unidad (34) de diagnóstico en el caso de un mal funcionamiento relevante para la seguridad o ante la instrucción de una unidad (32) de función de seguridad cuando se detecta una situación arriesgada en función de los datos de sensor evaluados.

4. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde la al menos una unidad (32) de función de seguridad está configurada para transmitir regularmente un mensaje (50) de estado y/o para transmitir un mensaje (52) de ejecución basándose en un evento para una ejecución respectiva de su función de seguridad.

5. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde el mensaje (50) de estado y/o el mensaje (52) de ejecución comprenden preferiblemente información de transmisor acerca de la unidad (32) de función de seguridad de transmisión, una marca de tiempo, una secuencia y/o una suma de comprobación.

6. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde la al menos una unidad (32) de función de seguridad está configurada para un autodiagnóstico en el que comprueba sus propios datos, programas, resultados de procesamiento y/o cumplimiento con un tiempo de sistema.

7. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde la unidad (34) de diagnóstico para la monitorización (46) de estado está configurada para recuperar una expectativa de estado predeterminada para los estados de la al menos una unidad (32) de función de seguridad, en particular, para modificar la expectativa de estado en función de estados, secuencias operativas y/o resultados operativos previos de las unidades (28) de lógica, y para comparar la expectativa de estado con un estado global actual derivado de los estados de los mensajes (50) de estado.

8. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde el mensaje (50) de estado proporciona información en cuanto a si la unidad (32) de función de seguridad que transmite el mensaje (50) de estado existe, fue capaz de inicializarse, tiene todos los recursos requeridos disponibles y/o está lista para funcionar.

9. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde la unidad (34) de diagnóstico para la monitorización (48) de ejecución está configurada para recuperar una expectativa (58) de ejecución predeterminada para la secuencia de ejecución de la al menos una unidad (32) de función de seguridad, en particular, para modificar la expectativa (58) de ejecución en

función de estados, secuencias de trabajo y/o resultados de trabajo previos de las unidades (28) de lógica, y para comparar la expectativa (58) de ejecución con la secuencia de ejecución derivada de los mensajes (52) de ejecución.

10. El dispositivo (10) de seguridad según la reivindicación 9,
en donde la unidad (34) de diagnóstico está configurada para tener en cuenta al menos uno de los siguientes criterios cuando se evalúa la comparación de la expectativa (58) de ejecución con la secuencia de ejecución derivada de los mensajes (52) de ejecución: una secuencia de ejecución, la ausencia de una ejecución, una ejecución adicional, una desviación de las ejecuciones con respecto a una cuadrícula temporal, una duración de ejecución que es demasiado corta o una duración de ejecución que es demasiado larga.

11. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde el mensaje (52) de ejecución comprende información de ejecución acerca de la última ejecución respectiva de la función de seguridad, en particular, con tiempo de inicio y/o duración de ejecución.

12. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde el entorno (22) de tiempo de ejecución comprende un agregador (56) que se dispone lógicamente entre la al menos una unidad (32) de función de seguridad y la unidad (34) de diagnóstico y está configurado para recibir los mensajes (52) de ejecución y para generar a partir de los mismos la secuencia de ejecución con una secuencia y/o duración de las ejecuciones de las funciones de seguridad de la al menos una unidad (32) de función de seguridad, en particular, en tiempo real.

13. El dispositivo (10) de seguridad según cualquiera de las reivindicaciones anteriores,
en donde el al menos un sensor (14) está configurado como un sensor optoelectrónico, en particular, una barrera de luz, un escáner de luz, una rejilla de luz, un escáner láser, un LiDAR FMCW o una cámara, como un sensor ultrasónico, un sensor inercial, un sensor capacitivo, un sensor magnético, un sensor inductivo, un sensor UWB o como un sensor de variable de proceso, en particular, un sensor de temperatura, un sensor de flujo, un sensor de nivel o un sensor de presión, y en donde el dispositivo (10) de seguridad comprende, en particular, una pluralidad de sensores (14) idénticos o diferentes.

14. Un método implementado por ordenador para monitorizar al menos una máquina (12), en donde al menos un sensor (14) genera datos de sensor en relación con la máquina (12) y una unidad (16, 22) de procesamiento para los datos de sensor como un entorno (22) de tiempo de ejecución ejecuta una pluralidad de unidades (28) de lógica en al menos un nodo informático (26), en donde al menos una unidad (28) de lógica como una unidad (32) de función de seguridad evalúa los datos de sensor de una forma orientada a la seguridad y al menos una unidad (28) de lógica como una unidad (34) de diagnóstico monitoriza la al menos una unidad (32) de función de seguridad,
caracterizado por que la al menos una unidad (32) de función de seguridad transmite mensajes (50) de estado y mensajes (52) de ejecución a la unidad (34) de diagnóstico, en donde mensajes (50) de estado proporcionan información acerca del estado de la al menos una unidad (32) de función de seguridad y, por lo tanto, acerca de su disponibilidad operativa y restricciones o errores posibles, y en donde un mensaje (52) de ejecución se refiere a la ejecución de una función de seguridad o un servicio que la al menos una unidad de función (32) ejecuta, a partir del cual se genera una secuencia de ejecución de las funciones o servicios de seguridad ejecutados, por que la unidad (34) de diagnóstico reconoce un mal funcionamiento relevante para la seguridad en una monitorización (46) de estado usando estados a partir de los mensajes (50) de estado y en una monitorización (48) de ejecución usando una secuencia de ejecución generada a partir de los mensajes (52) de ejecución, y por que el entorno (22) de tiempo de ejecución comprende un sistema de mensajes con dos canales de mensajes, a través del cual la al menos una unidad (32) de función de seguridad transmite mensajes (50) de estado y mensajes (52) de ejecución en paralelo a la unidad de diagnóstico.

Figura 1

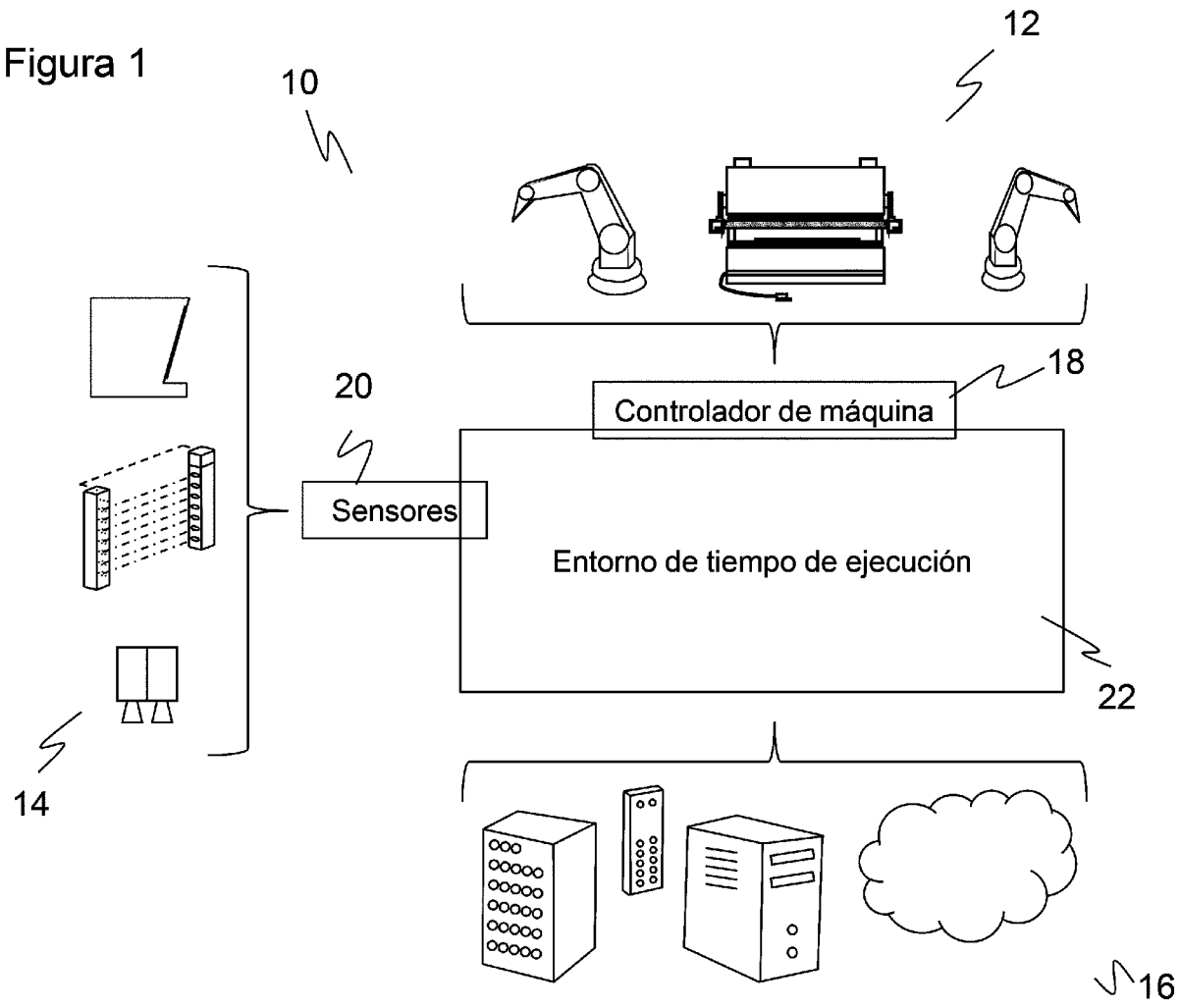


Figura 2

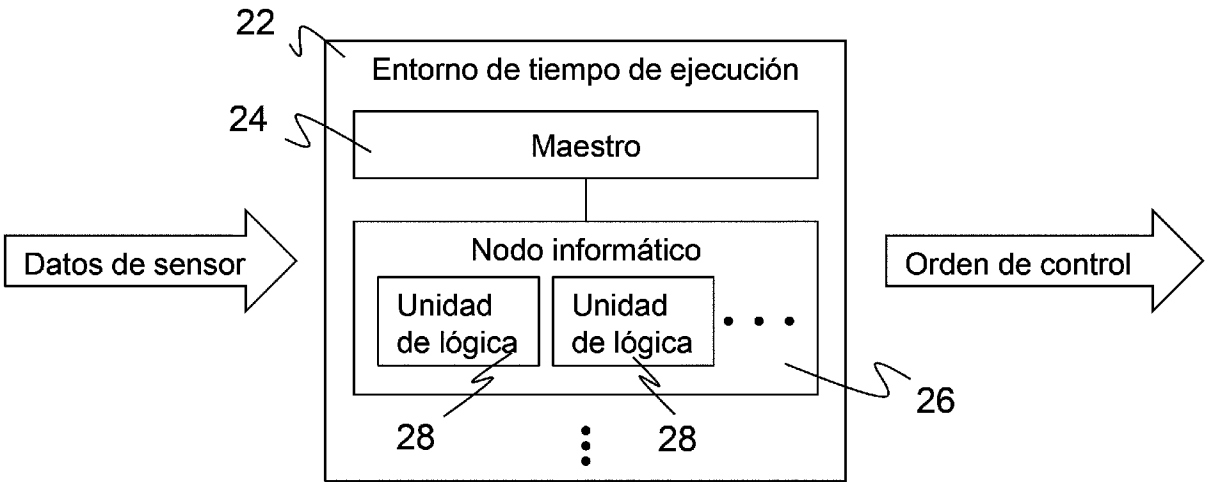


Figura 3

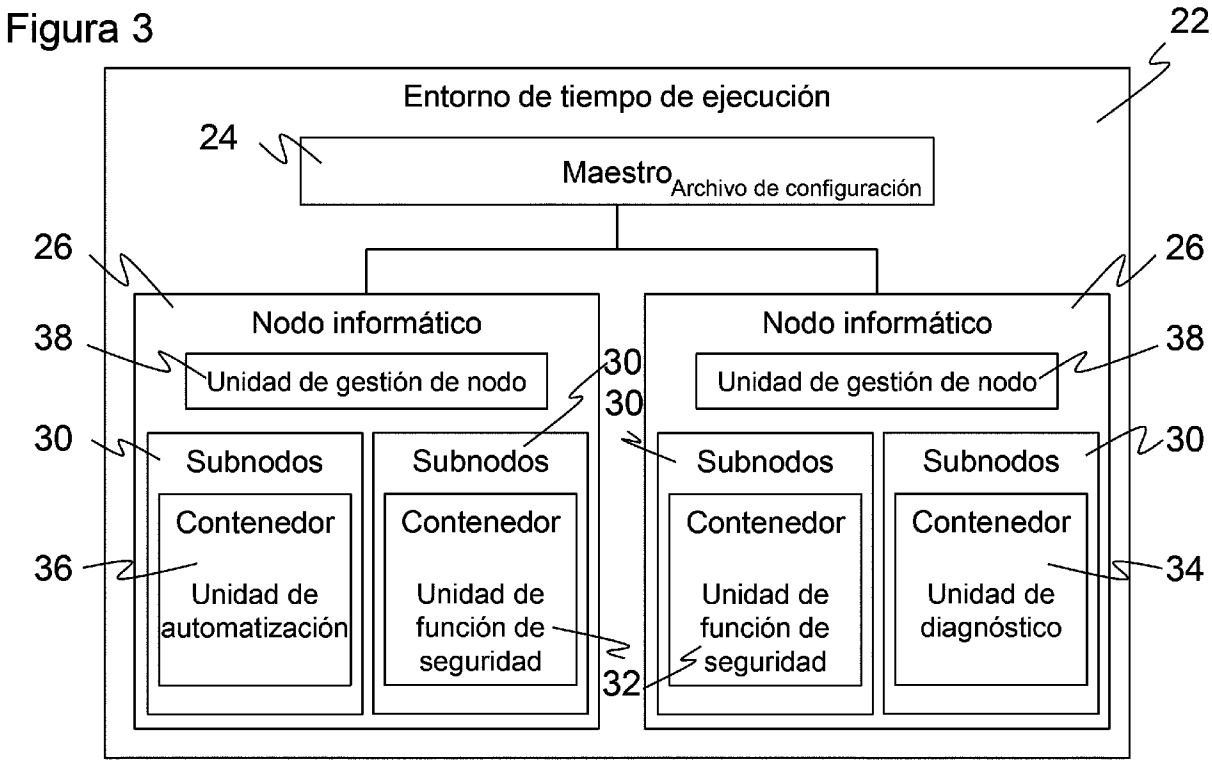


Figura 4

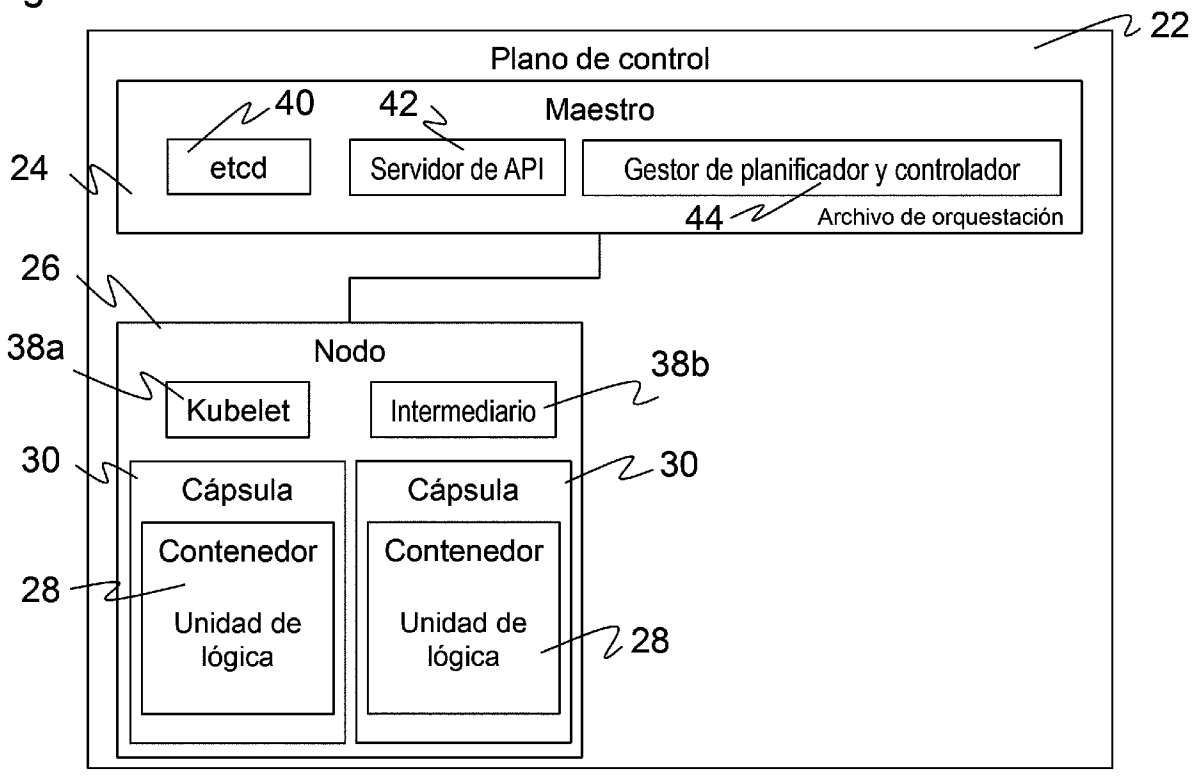


Figura 5

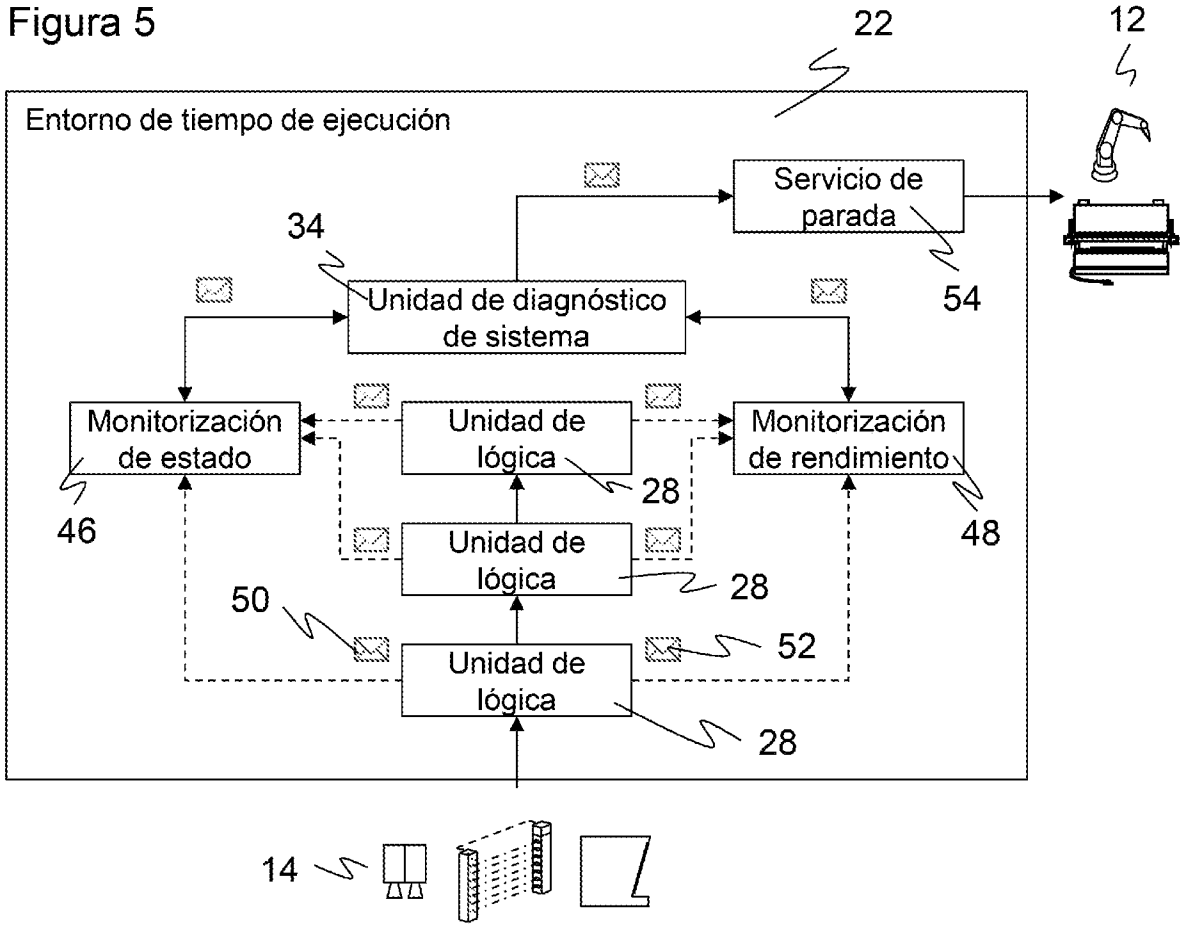


Figura 6

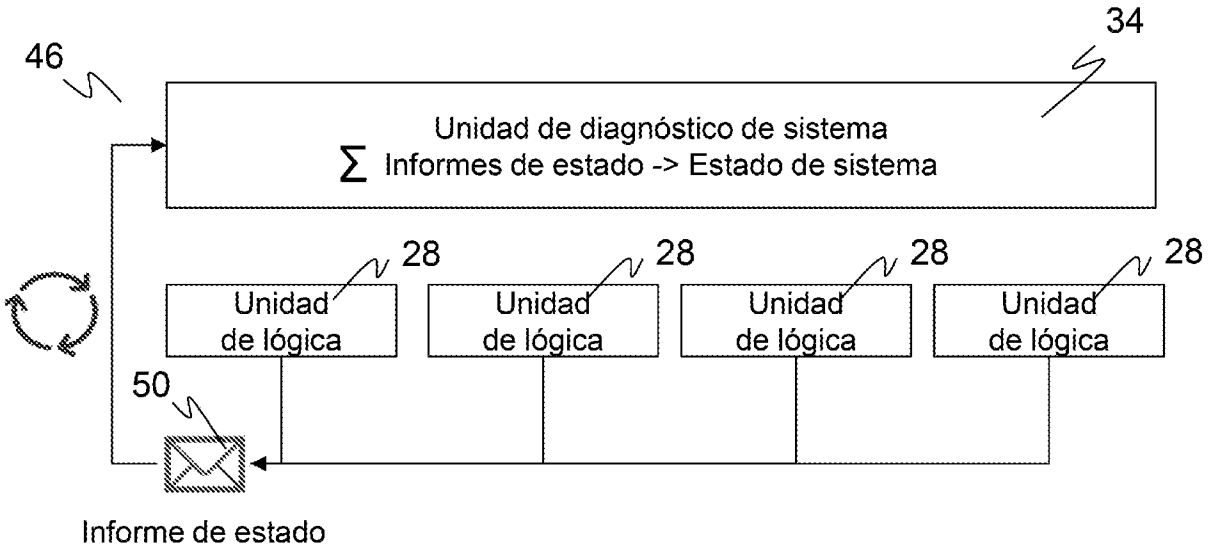
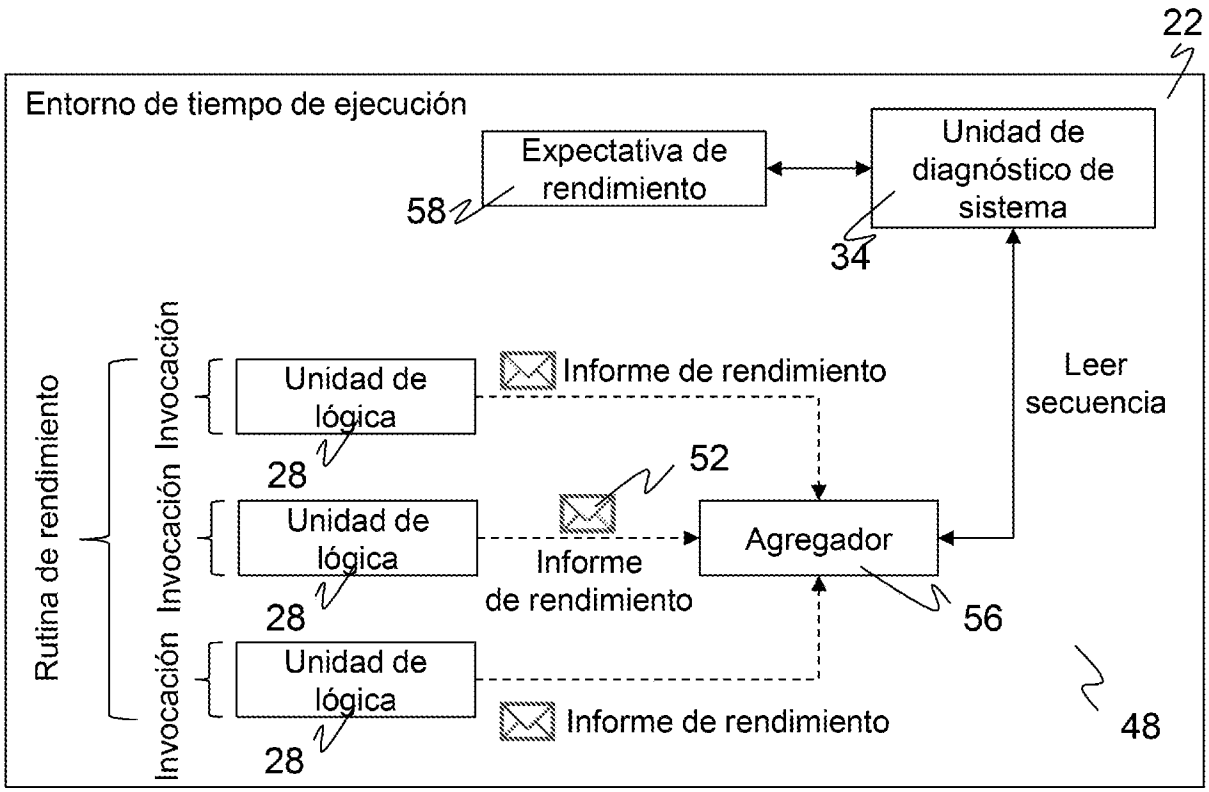


Figura 7


22
Entorno de tiempo de ejecución
Expectativa de rendimiento
58
Unidad de diagnóstico de sistema
34
Rutina de rendimiento
Invocación
Invocación
Invocación
Unidad de lógica
28
Unidad de lógica
28
Unidad de lógica
28
Informe de rendimiento
52
Informe de rendimiento
Agregador
56
Leer secuencia
Informe de rendimiento
48