(19) OFICINA ESPAÑOLA DE PATENTES Y MARCAS

ESPAÑA

(11) Número de publicación: **2 369 762**

(51) Int. Cl.:
G05B 11/01 (2006.01)

(12) TRADUCCIÓN DE PATENTE EUROPEA T3

(96) Número de solicitud europea: **06840250 .2**
(96) Fecha de presentación : **13.12.2006**
(97) Número de publicación de la solicitud: **1963930**
(97) Fecha de publicación de la solicitud: **03.09.2008**

(54) Título: **Sistema y método para implementar el control y detección de sincronización de tiempo en un sistema instrumentado de seguridad.**

(30) Prioridad: **20.12.2005 US 752272 P**

(45) Fecha de publicación de la mención BOPI:
05.12.2011

(45) Fecha de la publicación del folleto de la patente:
05.12.2011

(73) Titular/es: **FIELDBUS FOUNDATION**
9005 Mountain Ridge Drive
Bowie Building, Suite 190
Austin, Texas 78759, US

(72) Inventor/es: **Duffy, Joseph D.;**
Ramachandran, Ram y
Gabler, John Carl

(74) Agente: **Martín Santos, Victoria Sofía**

DESCRIPCIÓN

Sistema y método para implementar el control y detección de sincronización de tiempo en un sistema instrumentado de seguridad

CAMPO TÉCNICO

El campo técnico, al que las diversas realizaciones se refieren es la arquitectura del sistema de control. Más particularmente, el campo técnico se refiere a sistemas y métodos para controlar las funciones y la operación de sistemas instrumentados de seguridad de sistemas de control automáticos. Incluso más particularmente, ciertas realizaciones se refieren a sistemas y métodos para controlar sistemas instrumentados de seguridad en el contexto de un sistema de control automático que vincula los controladores de dispositivo por medio de una red de control para facilitar el control de procesos industriales, de fabricación y otros.

ANTECEDENTES

Los procesos y sistemas complejos de implementación industriales, de fabricación, petroquímicos y de otras “industrias de automatización” han migrado de arquitecturas de propiedad, centralizadas a arquitecturas abiertas, descentralizadas para facilitar la automatización de tales procesos y sistemas. Las arquitecturas descentralizadas implementan normalmente sistemas y redes de control de bus de campo en el que el control se distribuye entre los diversos dispositivos dentro de la red y/o sistema. Ejemplos de arquitecturas de bus de campo abiertas, interoperables y descentralizadas incluyen el bus de campo FOUNDATION de Fieldbus Foundation (Austin, TX), PROFIBUS de PROFIBUS International (Karlsruhe, Alemania); LonWorks de Echelon Corporation (San José, CA), Ethernet Industrial, Ethernet de alta velocidad y otros tipos de arquitecturas de redes. Tales redes, independientemente de la configuración real, las velocidades de transmisión de datos soportadas por las mismas o similares se refieren en lo sucesivo colectivamente como “Arquitecturas de bus de campo”.

La demanda para sistemas de bus de campo de control distribuidos abiertos e interoperables a menudo se lleva por proveedores y usuarios de equipos. Los proveedores prefieren normalmente Arquitecturas de bus de campo debido a que les permite vender sus productos y servicios a más usuarios, en lugar de hacerlo sólo a usuarios que operan con un sistema patentado específico. Los usuarios desean estabilizar las Arquitecturas de bus de campo, por ejemplo, debido a que a menudo les ofrece seleccionar los dispositivos y/o servicios de campo de bus a partir de múltiples usuarios en lugar de sólo dispositivos diseñados específicamente para un sistema patentado.

Muchos sectores de la industria de automatización tienen también una necesidad para sistemas de “seguridad” especiales para posibilitar la seguridad del personal de planta y para evitar el daño a equipos debido a eventos inesperados. Estos sistemas de “seguridad” especiales se denominan colectivamente “Sistemas Instrumentados de Seguridad” (SIS). Los usuarios y proveedores a menudo requieren sistemas SIS para cumplir con los estándares de seguridad internacionales tales como el Comité Electrotécnico Internacional (IEC) 61508 (sistemas de seguridad eléctrica funcional/electrónico/relacionados con seguridad electrónicos programables) y IEC 61511 (seguridad funcional: sistemas instrumentados de seguridad para el sector industrial de procesos).

Por tanto, los usuarios y proveedores de los dispositivos y sistemas SIS tienen una necesidad para una Arquitectura de bus campo de SIS abierta, interoperable (en lo sucesivo un “campo de bus SIS”) que le permita a lo usuarios soportar y/o proporcionar el control SIS utilizando Arquitecturas de bus de campo existentes. De forma deseable, un bus de campo SIS es directamente compatible con las Arquitectura de bus de campo existentes y no requiere una modificación de protocolos de comunicación existentes, bloques de función y/o otros aspectos de red.

El documento US 2004/0230323 muestra una solución de control SIS para una Arquitectura de bus de campo que comprende un canal negro en el que la solución de control SIS es capaz de determinar si al menos un mensaje comunicado a través de un medio de transmisión no se ha colocado en cola controlando las señales de sincronización de tiempo. El problema se basa en cómo detectar los errores colocados cola controlando las señales de sincronización del tiempo. Este problema se soluciona mediante el aparato de la reivindicación 1.

BREVE DESCRIPCIÓN DE LAS FIGURAS REPRESENTADAS

La Figura 1 es una visa global de un sistema de control extendido que se puede utilizar en conjunto con una o más realizaciones para soportar un bus de campo SIS.

La Figura 2 muestra el modelo de comunicación estratificado de Interconexión de Sistemas Abiertos en comparación con el modelo de comunicación descrito en la presente memoria.

La Figura 3 ilustra una realización de los componentes físicos de un dispositivo de campo.

La Figura 4 resume las relaciones de comunicación virtual proporcionadas por la Subcapa de Acceso al Bus de Campo.

La Figura 5 ilustra dos dispositivos interconectados por medio de servicios de comunicación uno con componentes relacionados con seguridad y uno sin componentes relacionados con seguridad.

La Figura 6 ilustra un diccionario de objetos.

Las Figura 7A y 7B ilustran los dispositivos de comunicación virtuales dentro del modelo de comunicación para usarse en los dispositivos de seguridad y sin seguridad.

Las Figuras 8A y 8B ilustran una estructura de aplicación del bloque de función dentro de un dispositivo de campo que contiene componentes relacionados con seguridad y sin seguridad.

Las Figuras 9A y 9B ilustran dispositivos externos interconectados en un bus con dispositivos de campo para implementaciones de seguridad y sin seguridad.

La Figura 10 ilustra el plano de un objeto de directorio del diccionario de objetos.

La Figura 11 ilustra ejemplos de parámetros interconectados para un único bucle para componentes relacionados con seguridad y sin seguridad.

La Figura 12 ilustra una realización de una arquitectura de sistema.

La Figura 13 ilustra un bloque de función relacionado con seguridad con entradas configurables por el usuario, salidas configurables por el usuario y un algoritmo configurable por el usuario.

La Figura 14 ilustra una aplicación que utiliza bloques de función relacionados con seguridad y sin seguridad estándares y bloques de función flexibles.

La Figura 15 es un diagrama de bloques que ilustra un ejemplo de una aplicación utilizando bloques de función estándares, flexibles y los FFB.

Las Figuras 16A, 16B y 16C son un diagrama de flujo que ilustra una metodología por la que se pueden comunicar datos, utilizando una topología editor-suscriptor, y autentificada para los bloques de función relacionados con seguridad y bloques de función flexibles relacionados con seguridad.

Las Figuras 16D, 16E y 16F son un diagrama de flujo que ilustra otras metodologías por las que se pueden comunicar datos, utilizando una topología editor-suscriptor y autentificada para bloques de función relacionados con seguridad y bloques de función flexibles relacionados con seguridad.

Las Figuras 17A, 17B y 17C son un diagrama de flujo que ilustra una metodología por la que se pueden comunicar datos, que utiliza una topología cliente-servidor, y autentificada para los bloques de función relacionados con seguridad y los bloques de función flexibles relacionados con seguridad.

Las Figuras 17D, 17E y 17F son un diagrama de flujo que ilustra una metodología por la que se pueden comunicar datos, que utiliza una topología cliente-servidor, y autentificada para los bloques de función relacionados con seguridad y los bloques de función flexibles relacionados con seguridad.

DESCRIPCIÓN DETALLADA

Como se ha descrito en la presente memoria, diversos sistemas, componentes y métodos ejemplares (en los sucesivo denominados, colectivamente “sistemas”) se describen para utilizar los sistemas SIS en Arquitecturas de bus de campo. Los diversos sistemas expuestos en la presente memoria se pueden utilizar en diversos tipos y formas de dispositivos SIS así como diversos tipos de Arquitecturas de bus de campo. Adicionalmente, diversos sistemas se pueden utilizar de forma deseable en diversas Arquitecturas de bus de campo sin cambios significantes y preferiblemente de ninguno de los protocolos, metodologías u otros procesos utilizados actualmente en Arquitecturas de bus de campo para comunicar la información no SIS a través de la red para la notificación a y/o la autorización mediante dispositivos de bus de campo compatibles con la red.

Más particularmente, una implementación SIS puede incluir un aparato configurado para operar en un sistema de control abierto que incluye: una memoria, que incluye un sistema de gestión de datos; uno o más elementos SIS; un procesador, conectado de forma que pueda operar a la memoria; una unidad de fijación del medio, que traslada los mensajes de entrada y los mensajes de salida entre el procesador y un medio de transmisión; y un protocolo de seguridad o relacionado con seguridad extendido (“SISRP”), el cual proporciona el nivel deseado de seguridad necesario para una implementación SIS particular. El sistema de gestión de datos puede incluir la información programada del sistema que el procesador ejecuta de forma deseable según se especifica por la programación del sistema.

Una implementación SIS en una Arquitectura de bus de campo puede permitir, por ejemplo, la interoperabilidad entre una pluralidad de dispositivos, al menos uno de los cuales incluye un componente SIS (“SISC”), tal como un bloque de

recurso, un bloque de función, un bloque transductor o un objeto de enlace, y una unidad de fijación del medio, conectada de forma que pueda operar al SISC. En una realización de este tipo, los bloques de recurso identifican únicamente cada dispositivo, el bloque de función procesa los parámetros para producir un mensaje de salida y la unidad de fijación del medio traduce el mensaje o los mensajes de entrada recibidos, por ejemplo, de un medio de transmisión al dispositivo SIS y de los mensajes de salida del dispositivo SIS al medio de transmisión. Una realización de este tipo se puede considerar como una implementación del bloque de función.

Se puede usar un aparato que deseablemente incluya: una capa de usuario, que incluya un SISC encapsulado para proporcionar funcionalidad; una capa física, que traduzca los mensajes desde un medio de transmisión en un formato adecuado para usarse mediante la capa de usuario y de la capa de usuario en una señal para la transmisión en el medio de transmisión; y una pila de comunicaciones, conectada a la capa de usuario y a la capa física. La pila de comunicaciones puede incluir una capa de enlace de datos y una capa de aplicación. La capa de enlace de datos controla la transmisión de mensajes en el medio de transmisión. La capa de aplicación le permite a la capa de usuario comunicarse a través del medio de transmisión.

De forma similar, una implementación SIS en una Arquitectura de bus de campo puede permitir la interoperabilidad entre una pluralidad de dispositivos, en el que al menos un dispositivo incluye un bloque de recurso, un objeto de enlace y una unidad de fijación del medio, conectada de forma que pueda operar al objeto de enlace. Los bloques de recurso se pueden adaptar para identificar únicamente cada dispositivo, el objeto de enlace se puede adaptar para recibir los parámetros procesados y producir un mensaje de salida, y la unidad de fijación del medio puede traducir un mensaje de entrada de un medio de transmisión al objeto de enlace y el mensaje de salida del objeto de enlace al medio de transmisión.

Se puede utilizar un aparato que incluya: una capa de usuario, que incluya un o más SISC encapsulado o encapsulados para proporcionar funcionalidad; una capa física, que traduzca los mensajes desde un medio de transmisión en un formato adecuado para la capa de usuario y desde la capa de usuario en una señal para la transmisión en el medio de transmisión; y una pila de comunicaciones, conectada a la capa de usuario y a la capa física. La pila de comunicaciones incluye una capa de enlace de datos y una capa de aplicación. La capa de enlace de datos controla la trasmisión de los mensajes en el medio de transmisión. La capa de aplicación le permite a la capa de usuario comunicarse a través del medio de transmisión.

Del mismo modo, se puede usar una memoria para almacenar datos para el acceso mediante un marco de aplicación que opera en el dispositivo dentro de un sistema de control. La memoria incluye una estructura de datos almacenada en la memoria, incluyendo la estructura de datos uno o más SICS, tal como un bloque de recurso, que hace que las características específicas de los componentes físicos del dispositivo puedan leerse electrónicamente, un bloque de función de encapsulado y/o al menos un bloque transductor. El bloque de función incluye programa y parámetros configurados por el usuario final y el al menos un bloque transductor controla el acceso al bloque de función.

Por tanto, debe apreciarse que diversos dispositivos y sistemas se pueden utilizar y que tales dispositivos y sistemas pueden utilizar los SISC y/o un SISRP para incorporar los dispositivos SIS en las Arquitecturas de bus de campo. Sin embargo, con propósitos de claridad y de simplicidad, la descripción siguiente se refiere principalmente a una realización de una Arquitectura de bus de campo, en concreto, una que utiliza los bloques de función para proporcionar una estructura general para especificar diferentes tipos de funciones de dispositivo en tanto utiliza una red de comunicaciones de bus de campo común. Sin embargo, se pueden utilizar otras arquitecturas y realizaciones en conjunto con los sistemas descritos en la presente memoria.

Como se conoce y se aprecia normalmente, una implementación de bloque de función de una Arquitectura de bus de campo define los componentes internos de una aplicación o porción de la misma, implementada por un dispositivo para proporcionar la operación del sistema. Las aplicaciones del bloque de función especifican cómo cada aplicación, o porción de la misma, está en la interfaz con cada una de las demás aplicaciones en el sistema para proporcionar la interoperabilidad estándar entre dispositivos.

Una implementación de una implementación de bloque de función de una Arquitectura de bus de campo se ha especificado como las especificaciones el bus de campo FOUNDATION como se ha proporcionado por Fieldbus Fountadion de Austin, Texas. Como se conoce y se aprecia normalmente, el bus de campo FOUNDATIONS especifica un bus de campo de menor velocidad (H1) optimizado para el control de procesos y una estructura principal de bus de campo de Ethernet a Alta Velocidad (HSE) para el control de alto rendimiento, la integración del subsistema y la integración de sistemas de gestión de información. El sistema de control puede soportar una diversidad de dispositivos de campo, incluyendo sensores y accionadores, o dispositivos de campo de alta velocidad, tales como los controles de celdas, motores, unidades lógicas y entrada/salida remota (I/O). Puesto que el bus de campo FOUNDATION es una arquitectura de control distribuida abierta e interoperable, los dispositivos de control a partir de los diferentes vendedores interoperan en el bus de campo H1 o HSE y comparten las funciones de control (por ejemplo, el control se distribuye en dispositivos de bus de campo). La distribución del control en los dispositivos de bus de campo a menudo reduce los costes de la instalación del sistema debido a la necesidad de ordenadores de control centralizados y los subsistemas I/O se reducen o eliminan. Adicionalmente, la distribución de control en dispositivos de bus de campo a menudo reduce los costes de operación y de mantenimiento del sistema debido a que los bloques de función estándares en los

dispositivos proporcionan más información acerca de las mediciones de proceso y del estado del dispositivo. En este contexto se expone en la presente memoria esta realización para proporcionar el control de bus de campo de dispositivos SIS. Debe apreciarse, sin embargo, que otras implementaciones se pueden usar, incluyendo aquellas compatibles con y/o incorporadas en otras Arquitecturas de campo de bus.

En particular, una Arquitecturas de bus de campo FOUNDATION está provista de una arquitectura de sistema de control con seguridad de comunicación extendida y adicional. Tal seguridad de comunicación se proporciona además de y/o "por encima" de la seguridad proporcionada mediante los sistemas de comunicación existentes en tanto utiliza bloques de función relacionados con seguridad nuevos, que son compatibles con un marco de bloques de función existente, tal como el marco proporcionado por las especificaciones del bus de campo FOUNDATION. Debe apreciarse que el nuevo sistema elimina normalmente y/o reduce significativamente, la necesidad del control personalizado costoso y difícil para mantener el soporte lógico y los dispositivos de entrada/salida "I/O" especiales para las aplicaciones SIS. Los bloques de función relacionados con seguridad descritos en la presente memoria con referencia a esta realización y realizaciones similares, se refieren colectivamente en la presente memoria como Bloques de Función de Sistemas Instrumentados de Seguridad ("SISFB") - una realización de un SISC.

VIZUALIZACIÓN GLOBAL DEL SISTEMA SISFB

Como se muestra en la Figura 1, un dispositivo de campo (que puede contener uno o más SISC) es uno que opera en un sistema de control de Arquitectura de bus de campo y que se clasifica en general como un programador activo de enlace compatible no SIS 100, un programador activo de enlace compatible SIS 100', un maestro de enlace compatible (no) SIS 105/105', o un dispositivo básico compatible (no) SIS 110/110'. Como se describe en mayor detalle a continuación, los componentes SIS y no SIS en los en los dispositivos SIS (es decir, programadores activos de enlace, maestros de enlace y dispositivos básicos) son sustancialmente similares, pero, los componentes SIS utilizan una seguridad extendida adicional o SISRP para posibilitar que las comunicaciones entre los SISC sean seguras y no se interrumpan, modifiquen o de lo contrario se degraden. Por tanto, cuando se incluyen dispositivos SIS en un sistema de control de Arquitectura de bus de campo, tales dispositivos de campo SIS pueden clasificarse además como un programador activo de enlace SIS 100', un maestro de enlace SIS 105' o un dispositivo básico SIS 110'. Los SISC pueden comunicarse con otros SISC y/o sin SISC (por ejemplo, con propósitos de reportar y otros usos) utilizando las Arquitecturas de bus de campo existentes tales como un bus 120 y/o 120'.

Independientemente de si un dispositivo de campo incluye un SISC, un dispositivo de campo se clasifica en base a sus capacidades y responsabilidades de control. Por ejemplo, un dispositivo de campo se clasifica como un programador activo de enlace 100/100' si actúa como el controlador de red de un bus 120/120'. Un dispositivo de campo se clasifica como un maestro de enlace 105/105' si es capaz de actuar como el controlador de red o el programador activo de enlace, pero que no ha asumido esa responsabilidad. Un dispositivo básico 110/110' no es capaz de actuar como el controlador de red.

Los dispositivos de campo se acoplan o se conectan electrónicamente mediante un medio de transmisión 120/120' que pueden ser alambres de entrada y salida individuales o una variedad de configuraciones de buses. Como se muestra en la Figura 1, se puede utilizar una configuración de bus por la cual ambos dispositivo de campo con SISC y sin SISC se pueden conectar. El índice de rendimiento del bus puede variar. Unos pocos de los buses ejemplares son el bus H1 31,25 kbit/s y el bus HSE 100Mbit/s. Sin embargo, otras velocidades de transferencia de datos y configuraciones de bus se pueden usar de forma adecuada. Tales realizaciones pueden utilizar otras configuraciones de red de alta velocidad o de baja velocidad y pueden o no utilizar bloques de función.

Las velocidades de transferencia de datos de bus en general no dependen de si el dispositivo de campo SISC y/o no SISC se conectan al sistema de control. Los puentes 130 (Figura 1) y los buses 120/120' se pueden reemplazar adecuadamente por otras configuraciones del sistema. Por ejemplo, una realización puede utilizar dispositivos HSE que se conectan a una topología estrella por conmutadores de Ethernet. Otras realizaciones de sistema y/o de red también se pueden utilizar.

Sin embargo, en una realización, el bus H1 se usa en general para aplicaciones de control de procesos, tales como temperatura, nivel y control de flujo. El bus HSE se utiliza generalmente para aplicaciones de alta velocidad. Los dispositivos que operan en buses HSE normalmente se autoenergizan o extraen la energía de un bus de energía separado en el cable de bus de campo (es decir, cable de 4 alambres), sin embargo, también se pueden energizar directamente a partir del bus de campo.

Como se muestra en la Figura 1, los varios dispositivos maestros de enlace 105/105' pueden operar en el bus o en los buses 120/120'. Cuando estos dispositivos maestros de enlaces 105/105' se activan, estos dispositivos maestros de enlace (SIS) 105/105' apuestan por la responsabilidad de convertirse en el programador activo de enlace 100/100'. El dispositivo maestro de enlace 105/105' que se convierte en el programador activo de enlace 100/105' es el dispositivo con la dirección de red más baja. Como alternativa, un dispositivo particular puede ser el maestro de enlace "preferido". En cuyo caso, cuando se activa el sistema el maestro de enlace 105/105' con la dirección de red más baja asumiría las responsabilidades del programador activo de enlace 100/100'. Después, el maestro de enlace "preferido" 105/105' enviaría un mensaje al programador activo de enlace 100/100' dirigiéndolo al control de transferencia. Tras la recepción del mensaje, el programador activo de enlace 100/100' transferiría el control al maestro de enlace preferido 105/105'.

Además, cuando se conectan los dispositivos SIS al sistema de control, por ejemplo, en el bus 120 ó 120', el maestro de enlace designado como el programador activo de enlace es deseablemente compatible con SIS. Como se muestra en la Figura 1, el bus 120' se controla de forma adecuada por un programador activo de enlace SIS 100' debido a la existencia del dispositivo básico SIS 110' en la red 120'. Como se describirá en más detalle a continuación en la presente memoria, los SISC en los dispositivos de campo SIS 110' se configuran normalmente para aceptar entradas e instrucciones a partir de los SISC en otros dispositivos SIS y no a partir de no SISC en dispositivos SIS o no SIS. Los no SISC en los dispositivos SIS o no SIS, sin embargo, pueden aceptar normalmente entradas e instrucciones a partir de componentes SISC o no SISC. Por tanto, cuando se incluyen los SISC en una red, el programador maestro de enlace designado incluye deseablemente cualquier SISC necesario.

Cuando existen múltiples maestros de vínculos 105/105' en un bus 120/120', existe una variedad de formas de conducir el proceso de adjudicación. Por ejemplo, un tipo de proceso de adjudicación se muestra en la Patente de Estados Unidos Nº 5.526.358 publicada el 11 de Junio de 1996 que se incorpora aquí por referencia en su totalidad. El proceso de adjudicación se puede conducir también si el programador activo de enlace 100/100' que controla un bus 120/120' funciona incorrectamente o se retira. Además, cuando los dispositivos SIS 105' están sobre un bus, el programador activo de enlace se seleccionará a partir de un maestro de enlace SIS disponible 105'.

El sistema de control puede incluir también un puente 130 para interconectar los buses individuales y crear redes más grandes. La comunicación entre los buses individuales se puede controlar a través de uno o más puestos de operarios 150. Además, en una realización SIS, los puestos de operarios son deseablemente compatibles con SIS.

Además, un maestro de enlace 105/105' contiene deseablemente las mismas capacidades de control que un programador activo de enlace 100/100'. Por tanto, las capacidades de ambos se describen además en la presente memoria con referencia a un maestro de enlace. Más particularmente, un maestro de enlace 105/105' incorpora un interfaz de programa que comprende las siguientes tres capas: (1) una capa física, (2) una pila de comunicaciones y (3) una capa de usuario. Cuando los SISC se utilizan en dispositivos de campo, la capa de usuario incluye y utiliza además un SISRP o interfaz, como se describe con mayor detalle en la presente memoria a continuación. De lo contrario, los componentes SIS y no SIS compatibles utilizan capas físicas comunes, pilas de comunicaciones y capas de usuario.

Con referencia ahora a la Figura 2, la capa física (PHY) 200 y la pila de comunicaciones 205 se derivan del modelo de interconexión de sistemas abiertos (OSI). La capa física (PHY) 200 es deseablemente la misma que la capa OSI 1, y la pila de comunicaciones 205 corresponde generalmente a las capas OSI 2 y 7. La capa de usuario 235 no se define por el modelo OSI. Como alternativa, la capa física 200 y la pila de comunicaciones 205 se pueden derivar de una diversidad de estándares de redes diferentes, tales como el Protocolo de Control de Transmisión/Protocolo Internet (TCP/IP), UNIX y otras. Una descripción detallada de cada una de las tres capas se presenta a continuación. De forma deseable, tanto para implementaciones no SIS como SIS, la PHY 200 y la pila son las mismas. Los elementos comunes en estas capas posibilitan que los dispositivos SIS se conecten a las arquitecturas de bus de campo existentes sin necesitar cambios en los protocolos de comunicaciones utilizados actualmente por los dispositivos no SIS. Como tal, la capa física 200 y la pila de comunicaciones 205 se describen en la presente memoria con respecto a una Arquitectura de bus de campo común o genérico (es decir, no SIS).

CAPA FÍSICA

Como se muestra en las Figuras 1 y 2, la capa física 200 recibe mensajes de la pila de comunicaciones 205 y convierte los mensajes en señales físicas en el medio de transmisión 120/120' y viceversa. La capa física 200 se puede definir mediante estándares aprobados por la Comisión Electrotécnica Internacional (IEC) y la Sociedad Internacional de Medición y de Control (ISA). Para más información acerca de la capa física 200, véase el documento de ISA S50.02-1992 y el documento de IEC 1158-2, ambos de los cuales se incorporan en la presente memoria por referencia en sus totalidades. Debe apreciarse, sin embargo, que la capa física se puede definir también mediante otros estándares normalmente conocidos en la técnica.

Los mensajes se pueden codificar utilizando, por ejemplo, la técnica Manchester Biphase-L bastante conocida y la señal de reloj se puede introducir en el flujo de datos en serie. Nuevamente, otros esquemas de codificación se pueden utilizar, según se desee y/o se especifique por los estándares de implementación y de conexiones en red que se utilicen en cualquier realización particular. Los elementos físicos requeridos para instalar los mensajes de entrada del bus 120/120' y los mensajes de salida de un procesador dentro del dispositivo se denominan generalmente como la unidad de fijación del medio, tal como un adaptador de red. Después que la capa física 200 traduce un mensaje de entrada del bus 120/120', lo envía a la pila de comunicaciones 205. La pila de comunicaciones 205 se describe a continuación.

PILA DE COMUNICACIONES

La Figura 2 muestra una pila de comunicaciones 205 que incluye la capa de enlace de datos 210, la subcapa de acceso al bus de campo 220 y la especificación de mensaje de bus de campo 230. Estas capas (205, 210, 220 y 230) son comúnmente deseables tanto para los SISC como para los no SISC. También, para al menos una realización, la capa de enlace de datos es la misma que la capa OSI 2, mientras que la subcapa de acceso al bus de campo 220 y la especificación de mensaje de bus de campo 230 son subcapas dentro de la capa de aplicación OSI, capa OSI 7. La pila

de comunicaciones 205 no utiliza las capas 3-6. Las capas de la pila de comunicaciones 205 se describen a continuación.

Capa de Enlace de Datos

La capa de enlace de datos 210 controla transmisión de mensajes en el bus 120/120' de un programador activo de enlace 100/100', el dispositivo maestro de enlace 105/105' o dispositivo básico 110/110' en base a las instrucciones recibidas a partir de un controlador de red o del programador activo de enlace 100/100'. La capa de enlace de datos 220 puede ser un subconjunto de los estándares de la capa de enlace de datos IEC e ISA.

El programador activo de enlace 100/100' controla la capa de enlace de datos 210 de acuerdo con un programa de red almacenado en una memoria. La programación de red es una lista de tiempos de transmisión para la memoria intermedia de datos dentro del sistema. Las memorias intermedias de datos almacenan los datos recogidos por los dispositivos de campo. Por ejemplo, si el dispositivo de campo es un termómetro, la memoria de datos almacena la temperatura, y tras el comando, expone la lectura de temperatura en el bus 120/120'. Adicionalmente, el programador activo de enlace 100/100' mantiene una "lista viva" transmitiendo periódicamente un mensaje de paso de testigo. Utilizando la "lista viva", el programador activo de enlace puede identificar todos los dispositivos de campo que operan en el sistema. Cualquier dispositivo de campo que responda apropiadamente al paso de testigo se mantiene en la lista viva. Si un dispositivo de campo falla en responder al paso de testigo después de un número predeterminado de intentos, el dispositivo se retira de la lista viva. Puesto que múltiples redes se pueden conectar mediante puentes o de cualquier otra forma, en una realización, cada programación activa de enlace 100/100' mantiene deseablemente una "lista viva" para aquellos componentes en cada red respectiva.

Nuevos dispositivos se pueden añadir también a la lista viva. El programador activo de enlace 100/100' envía periódicamente mensajes de nodos de sondeo para las direcciones de red no mencionadas en la lista viva. Si un dispositivo de campo está presente en la dirección de red y recibe un mensaje de nodo de sondeo, el dispositivo de campo regresa inmediatamente un mensaje de respuesta de sondeo. Si el dispositivo de campo responde con un mensaje de respuesta de sondeo, el programador activo de enlace 100/100' añade el dispositivo de campo a la lista viva y confirma la adición del dispositivo de campo enviándole al dispositivo de campo un mensaje de activación de nodo.

Siempre que se añada o se retire un dispositivo de campo de la lista viva, el programador activo de enlace 100/100' difunde los cambios de la lista viva a todos los dispositivos de campo. Esto permite que cada dispositivo de campo mantenga una copia actual de la lista viva.

El programador activo de enlace 100/100' programa también las comunicaciones de otros dispositivos de campo que operan en el sistema. El programador activo de enlace 100/100' coordina el tiempo de cada comunicación emitiendo mensajes de datos compilados en tiempos programados. Tras la recepción del mensaje de dato compilado, el dispositivo de campo requerido difunde o publica sus datos a los otros dispositivos de campo que operan en el sistema. Para asegurar la sincronización apropiada, el programador activo de enlace 100/100' difunde también periódicamente un mensaje de distribución de tiempo en el bus 120/120' de manera que todos los dispositivos de campo tienen exactamente el mismo tiempo de enlace de datos. El mensaje de distribución de tiempo es un mensaje que incluye el tiempo de enlace de datos. El tiempo de enlace de datos es el tiempo del sistema del programador activo de enlace 100/100'. Cuando se recibe el mensaje de distribución de tiempo por los maestros de enlace 105/105 en un bus dado, los maestros de enlace 105/105' reajustan o recalibran sus tiempos de sistema individuales al tiempo de enlace de datos.

Las operaciones restantes se realizan entre los mensajes programados o intercambios de datos. El programador activo de enlace 100/100' otorga permisos a los otros dispositivos de campo para usar el bus 120/120' emitiendo un mensaje de paso de testigo a un dispositivo individual. Cuando el dispositivo de campo individual recibe el paso de testigo, se le permite al dispositivo de campo enviar mensajes hasta que el dispositivo de campo termine de enviar mensajes o hasta que haya expirado el tiempo máximo de permanencia con el testigo, lo que sea más corto. El tiempo de permanencia de testigo es la cantidad de tiempo que el dispositivo puede enviar mensajes después de recibir el paso de testigo. Este método de gestión de control se denomina normalmente, control del paso de testigo. Una diversidad de técnicas para implementar el control del paso de testigo es bastante conocida para aquellos expertos en la materia.

Para controlar los intercambios de datos cada dispositivo/componente incluye preferiblemente un cierre de entrada 240, el procesador 250, la memoria 255, parámetros contenidos 257 y un cierre de salida 260, y una unidad de fijación del medio 612, como se muestra en las Figuras 3 y 8. El cierre de entrada 240 y el cierre de salida 260 protegen el los valores de parámetros del acceso escrito u otras interfaces externas durante la ejecución de un bloque. El procesador 250 procesa la ejecución de los bloques almacenados, así como los algoritmos y programas dentro de los bloques. Los parámetros de introducidos y los parámetros contenidos 257 se almacenan en una memoria 255. La memoria es preferiblemente EEPROM o FLASHROM para permitir la programación del dispositivo sin el peligro de perder los datos debido a las pérdidas de energía. En realizaciones alternativas, la memoria 255 puede ser ROM, RAM, o EEPROM.

Subcapa de Acceso al Bus de Campo

Con referencia nuevamente a la Figura 2, la siguiente capa en la pila de comunicaciones 205 es la subcapa de acceso al bus de campo 220. La subcapa de acceso al bus de campo 220 utiliza los intercambios de datos programados y no programados de la capa de enlace de datos 210 para proporcionar un servicio para una especificación de mensaje del bus de campo 230. Una vez más, ésta subcapa 220 es deseablemente la misma tanto en los dispositivos compatibles SIS como no SIS. Los servicios proporcionados por la subcapa de acceso al bus de campo 220 es la dirección eficiente de mensajes enviados normalmente. Algunos ejemplos de los servicios de la subcapa de acceso al bus de campo se denominan relaciones de comunicación virtuales (VCR). La Figura 4 muestra tres capas de VCR: cliente/servidor 251, distribución de informe 252 y editor/suscriptor 254. Otras VCR pueden existir, sin embargo, en otras implementaciones.

La VCR cliente/servidor 251 se utiliza para los mensajes de operario, tales como los tipos de mensajes mencionados en la Figura 4. Específicamente, las VCR cliente/servidor 251 son comunicaciones uno a uno iniciadas por el usuario, no programadas, en cola entre los dispositivos de campo y/o los componentes dentro de los dispositivos de campo, incluyendo los SISC. En cola significa que los mensajes se envían y se reciben en el orden en que los mensajes fueron presentados para la transmisión sin sobreescritura del mensaje anterior. Un dispositivo de campo puede enviar un mensaje solicitando el intercambio de datos cuando el dispositivo de campo recibe un mensaje de paso o testigo del programador activo de enlace 100/100'. El dispositivo solicitante se denomina el cliente. El dispositivo que recibe la solicitud se denomina el servidor. El servidor responde cuando recibe un mensaje de paso de testigo del programador activo de enlace 100/100'. Como se menciona en mayor detalle en lo que sigue a continuación, cuando se implican los SISC en los intercambios de datos cliente/servidor, se utilizan las técnicas de verificación de mensaje y emisor adicional, tales como aquellas expuestas en el SISRP, para asegurar que se consiga el Nivel de Integridad de Seguridad apropiado ("SIL").

Las VCR de distribución de informe 252 se utilizan para notificaciones de eventos, tales como notificaciones de alarma para las consolas de los operarios e informes de tendencia. Especialmente, las VCR de distribución de informe son comunicaciones uno a muchos, iniciadas por el usuario, no programadas, en cola. Las VCR de distribución de informes 252 le permiten al dispositivo enviar un mensaje a una dirección común, tal como "TODAS LAS CONSOLAS DE OPERARIOS". De forma deseable, las VCR de distribución del informe para los SISC y los no SISC son idénticas. Debe apreciarse que las consideraciones SIL no surgen normalmente con respecto a los eventos de informe debido a los salvavidas integrados que existen en los componentes compatibles SIL, en concreto, en su pre-programación para conseguir ciertas funciones o acciones de "seguridad" cuando se detecta una condición errónea.

Las VCR editor/suscriptor 254 se utilizan para publicar datos. Específicamente, las VCR editor/suscriptor 254 son comunicaciones uno a muchos tamponadas. Tamponadas significa que sólo la última versión del dato se mantiene dentro de la red. Nuevos datos sobrescriben datos anteriores. Por ejemplo, un dispositivo de campo puede publicar o difundir mensajes a otros dispositivos de campo en un bus 120/120' cuando el dispositivo de campo recibe un mensaje de datos compilados del programador activo de enlace 100/100'. La VCR editor/suscriptor 254 se utiliza en los dispositivos de campo para la publicación programada de entradas y salidas de bloque de función de capa de usuario. La publicación de entradas y salidas de bloque de función de capa de usuario se describe más adelante. Como se describe con mayor detalle a continuación en la presente memoria, cuando se implican los SISC en el intercambio de datos editor/suscriptor, se utilizan técnicas de verificación de mensaje y de editor adicionales, tales como aquellas proporcionadas en el SISRP para asegurar que se consiga el Nivel de Integridad de Seguridad apropiado ("SIL").

Especificación del Mensaje de Bus de Campo ("FMS")

Otra capa en la pila de comunicaciones 205 es la especificación de mensaje del bus de campo ("FMS") 230. El FMS 230 permite la función de las aplicaciones del bloque para enviar esos mensajes entre sí utilizando un conjunto convencional de formato de mensajes. El FMS 230 describe los servicios de comunicación 270, los formatos de mensaje y el comportamiento del protocolo necesario para construir un mensaje para la capa de usuario 240, como se ilustra en la Figura 5. El formato de los FMS se define mediante un lenguaje de descripción con sintaxis formal denominado Notación de Sintaxis Abstracta 1 desarrollada por el Comité Consultivo de Telégrafos y Teléfonos Internacionales. De otro modo, el formato de los FMS se define utilizando los lenguajes descriptivos de mensaje normalmente conocidos.

Los datos que se comunican a través del bus 120/120' se pueden describir mediante una descripción de objeto. Como se ilustra en la Figura 6, las descripciones de objeto 280 se recogen juntas en una estructura denominada diccionario de objetos 281. Las descripciones de objetos 280 se identifican mediante un número índice 285. Un número índice es una referencia cruzada con la ubicación en la que se almacena una descripción de objeto particular en la memoria. El cero índice 287, denominado el encabezado del diccionario de objetos, proporciona una descripción del propio diccionario y define el primer índice para las descripciones de objetos de la aplicación de bloque de función 440.

Por ejemplo, los números índices 1-255 pueden definir tipos datos convencionales, tales como Booleanos, enteros, punto flotante, cadena de bits y estructuras de datos, que se utilizan para construir todas las descripciones de objetos 280. Los números índices por encima del número índice 255 son referencias cruzadas con las descripciones de objetos de la capa de usuario 280.

Los servicios de comunicación 270, mostrados en la Figura 5, proporcionan una forma estándar para las capas de usuario 235/235', tanto compatible con SIS como con no SIS, para comunicarse a través del bus de campo. Algunos ejemplos de los servicios de comunicación 270 son el servicio de gestión de contexto, el servicio del diccionario de

objetos y el acceso variable. Los servicios de gestión de contexto se utilizan para establecer y liberar las relaciones de comunicación virtuales con un dispositivo de campo virtual. El servicio del diccionario de objetos permite a la capa de usuario 235/235' acceder y cambiar las descripciones de objetos en un dispositivo de campo virtual. Los servicios de acceso variable permiten a la capa de usuario 235/235' acceder y cambiar variables asociadas con una descripción de objeto.

Además, los servicios de comunicación 270 posibilitan que la especificación del mensaje de bus de campo 230 se comunique con los dispositivos de bus de campo virtuales 310, 400 en la capa de usuario 235/235'. Como se muestra en la Figura 7A, un dispositivo de campo tendrá al menos dos dispositivos de campo virtuales, una red y un dispositivo de campo virtual de gestión del sistema 310 y un dispositivo de campo virtual de usuario 400.

El dispositivo de campo de gestión virtual de red y sistema 310 almacena típicamente los datos de gestión de red 320 y los datos de gestión del sistema 330. Los datos de gestión de red incluyen una porción de descripciones de objetos 322 de la base de información de gestión de red (NMIB) y una porción de datos objeto NMIB 325. Los datos de gestión del sistema 330 incluyen una porción de descripciones de objetos (SMIB) de la base de información de gestión del sistema 332, y una porción de datos objeto SMIB 335. El dispositivo de campo virtual de usuario 400 incluye los datos objeto de bloque 327 que incluye la descripción de objeto del bloque 326.

Las descripciones de objetos en base a información de gestión de sistemas y de red 322, 335 describen el formato de sistema y de red para los datos objetos en base a la información de gestión del sistema y de red 325, 332.

Los perfiles de comunicaciones convencionales se pueden utilizar también para permitir que los dispositivos de campo se comuniquen y trabajen juntos en el mismo medio de transmisión 120/120'. Los perfiles de comunicación utilizados en las aplicaciones de bloque de función 440 se pueden definir como dependientes en las categorías o clases de dispositivos de campo. También, para configurar y mantener los dispositivos de campo y su aplicación de bloque de función, se utiliza a menudo un formato de archivo común.

Como se muestra en la Figura 7B, se considera la combinación del bus 120/120', la unidad de fijación del medio 612 (Figuras 8A y 8B), la capa física 200, la capa de enlace de datos 210 y la pila de comunicación 205 para formar un "canal negro" (como se ha ilustrado con los bloques rallados) 207. El canal negro 207 proporciona una red de comunicaciones y de interconexiones estándar entre los SISC y los no SISC sin necesitar adiciones, supresiones o modificaciones de los protocolos ni de configuraciones de comunicaciones que se utilizan actualmente para soportar las comunicaciones entre los no SISC en uno o más dispositivos de campo que utilizan una Arquitectura de bus de campo.

CAPA DE USUARIO

La capa de usuario 235 procesa la información recogida por el dispositivo de campo que opera en el sistema. Como se muestra en la Figura 2, la capa de usuario 235 es una capa adicional añadida al modelo OSI. Como se muestra en la figura 7A, la capa de usuario se compone generalmente de una aplicación de la gestión de red y de sistema 430 y al menos una aplicación de bloque de función 440. Cada uno con su propio dispositivo de campo virtual descrito anteriormente.

La aplicación de bloque de función 440 define la funcionalidad del dispositivo de campo. Una aplicación de bloque de función 440 incluye uno o más recursos 500/500', como se muestra en la Figura 8A para un dispositivo de campo que tiene uno o más de los no SISC, y como se muestra en la Figura 8B para un dispositivo de campo que tiene uno o más de los SISC. Un dispositivo de campo puede contener recursos que incluyen SISC y no SISC. Un recurso 500/500' es una subdivisión lógica dentro de la estructura del soporte lógico y/o elementos físicos de un dispositivo. Un recurso 500/500' tiene control independiente de su operación, y su definición se puede modificar sin afectar los recursos relacionados.

Adicionalmente, en los componentes SIS, una subcapa de SIS, que proporciona el SISRP, 328, como se muestra en la Figura 7A, se incluye en la aplicación de bloque de función 440. Esta subcapa/protocolo 328 se describe en mayor detalle en lo sucesivo en la presente memoria.

Con referencia ahora las Figuras 8A y 8B, tanto los recursos relacionados no SISC 500 como los recursos relacionados SISC 500' se pueden construir de bloques de objetos, tales como: un bloque de recurso 510 o un bloque de recurso SIS ("SISRB") 510', bloque traductor 520 o bloque traductor SIS ("SISTB") 520', bloque traductor 530 o bloque traductor SIS ("SISFB") 530', objetos de tendencia 560, objetos de visualización 565, objeto de enlace 570 y/o objeto de enlace SIS 570', objetos de alerta 585, tiempo de sistema 601, programaciones de bloque de función 602 y tráfico de red. El tráfico de red incluye el tráfico programado y no programado. En un dispositivo SIS, el recurso debería contener uno o más SISFB y SISTB. Debe apreciarse que los SISRB, SISTB y SISFB son unos pocos ejemplos de los SIS. Adicionalmente, el recurso SIS debería diseñarse para detectar los fallos que ocurren fuera del recurso. A continuación, se proporciona una breve descripción de los bloques objetos utilizados en al menos una realización.

Un bloque de función 530 representa las funciones de automatización básicas realizadas por un recurso, tales como una entrada análoga, salida análoga, o proporcional/derivativa (PD), o cualquier otra función requerida para los dispositivos

de control de proceso o de fabricación. Los bloques de función 530 se diseñan para ser independientes en lo posible de los específicos dispositivos de entrada/salida y de la red.

En los dispositivos SIS, el SISFB 530' se utiliza normalmente para publicar datos a otro SISFB así como a bloques de función no SIS diseñados para usarse en aplicaciones de proceso. Sin embargo, un SISFB puede suscribir a datos publicados por otro SISFB para asegurar la conformidad con un SIL dado. Cada SISFB se puede identificar mediante un único número de perfil. En una realización, tal número de perfil se especifica mediante el bus de campo Foundation. Además, un SISFB posibilita la distribución del control SIS dentro y entre componentes de bus de campo conectados a una Arquitectura de bus de campo. En ciertas realizaciones, los SISFB se limitan a un conjunto determinado. Un de tales conjuntos de SISFB puede incluir una entrada análoga, salida análoga, entrada discreta, salida discreta, asignación análoga, asignación discreta, cambio de bloqueo, y lógica. Similarmente, otras realizaciones pueden proporcionarse para los SISFB que se proporcionan en una o todas de las tres clasificaciones tales como los bloques de función de entrada, los bloques de función de salida y los bloques de función de control.

Cada bloque de función 530/530' utiliza un parámetro de entrada de acuerdo con un algoritmo específico y conjunto interno de parámetros contenidos. Los parámetros de entrada son parámetros estructurados compuestos de un valor de campo y un estado de campo. El tipo de datos especificado para los parámetros de entrada es dependiente del tipo de dato de su valor de campo. El estado de campo es idéntico para todos los parámetros de entrada. Los parámetros contenidos se pueden utilizar para proporcionar valores al algoritmo de bloqueo. Los valores de los parámetros contenidos pueden ajustarse por el fabricante o como parte de la configuración. En general, los parámetros contenidos pueden ajustarse también durante el funcionamiento. Los parámetros de entrada y los parámetros contenidos se procesan de acuerdo con el algoritmo específico para producir parámetros de salida. Los parámetros de salida están disponibles para usarse dentro del mismo bloque de función 530/530' o por otros bloques de función 530/530'.

Los bloques transductores 520/520' pueden pre-procesar y post-procesar los datos entre los bloques de función 530/530' y dispositivos de campo, tales como sensores, accionadores y conmutadores. Los bloques transductores 520/520' pueden controlar el acceso a los dispositivos de entrada/salida a través de una interfaz independiente de dispositivo utilizada mediante los bloques de función 530/530'. Los bloques transductores 520/520' pueden realizar también funciones, tales como la calibración y la linearización. Los SISTB se diseñan deseablemente mediante un único número de perfil, tal como aquel asignado por el bus de campo Foundation. Además, los bloques transductores 520 pueden recibir entradas de no SISFB y/o SISFB. Sin embargo, los SISTB 520' pueden sólo deseablemente recibir entradas de otros SISC, tales como los SISFB, los bloques de recurso SIS y/o otros bloques SIS compatibles.

También, puesto que un SISFB no puede asumir que un núcleo de gestión del sistema subyacente ("SMK") está libre de fallos, ya que el SMK es parte del canal negro, el SISFB puede incluir un parámetro que funcione como un temporizador de vigilancia. Tal temporizador de vigilancia ayuda de forma adecuada en la detección de errores durante la programación de los bloques, surgiendo tales errores del canal negro que programa incorrectamente un bloque de función. Por ejemplo, un parámetro de "período de ejecución" se puede utilizar como un temporizador de vigilancia. Tal parámetro puede escribirse de forma deseada mediante dispositivos de configuración SIS compatibles. Adicionalmente, cada SISFB de salida puede controlar su ejecución y reajustar el temporizador de vigilancia (parámetro de período de ejecución) a cada momento en que se ejecute el bloque. Más particularmente, si el temporizador de vigilancia se vence o se actualiza a una velocidad demasiado rápida, alguna o todas las salidas del SISC o de los SISC afectados se pueden ajustar en un estado seguro. Debe apreciarse que un "estado seguro" es normalmente un componente e implementación específica.

Como se muestra además en las Figuras 8A y 8B, un recurso 500/500' también incluye normalmente uno o más objetos de enlace 570/570'. En un dispositivo no SIS, un objeto de enlace 570 intercambia datos entre los bloques de función 530 dentro de un recurso 500 o entre recursos. Los datos intercambiados por el objeto de enlace 570 pueden incluir datos o eventos del proceso. Además, el objeto de enlace 570 puede intercambiar datos de informe de tendencia o datos de notificación de alertas.

En un dispositivo SIS, se utilizan los objetos de enlace SIS ("SISLO"). Además de proporcionar las funciones y capacidades mencionadas anteriormente, los SISLO se extienden a objetos de enlace que utilizan un protocolo relacionado con seguridad extendido, tal como el SISRP, que incluye parámetros que especifican la correlación entre dos SISC, por ejemplo, entre un SISFB, un SISRB o un SISTB y un huésped, independientemente de si los SISFB se ubican en un huésped o en otro dispositivo o componente de campo. Debe apreciarse que otra correlación entre SISFB, SISTB, SISRB, huéspedes y no SISC puede proporcionarse según se necesite. Tales parámetros de correlación del objeto de enlace posibilitan a un suscriptor detectar errores que se pueden inducir mediante un canal negro subyacente. Los SISC pueden comunicarse entre sí utilizando el SISRP. El SISRP se describe con mayor detalle en lo que sigue a continuación.

Un bloque de recurso 510 posibilita que las características específicas de los elementos físicos de un dispositivo sean accesibles a la red. Los bloques de recurso 510 aíslan los bloques de función 530 de los elementos físicos del recurso incluyendo un conjunto de parámetros de elementos físicos independientes de implementación. En los componentes que contienen uno o más SISC, el bloque de recurso debería ser un bloque de recurso SIS ("SISRB") 510' puesto que, como se ha mencionado anteriormente, los SISC se pueden configurar (según se desea o sea necesario) para suscribir

la información proporcionada sólo mediante otro de los SISC. Los SISRB pueden diseñarse mediante un único número de perfil, tal como uno asignado por el bus de campo Foundation. También, los SISRB pueden incluir un parámetro, por ejemplo “SIL_LEVEL_SUP-PORTED”, que especifica el máximo nivel SIL de una aplicación en la que se puede utilizar el componente.

Los objetos de visualización 565 y los objetos de tendencia 560 proporcionan un acceso eficaz a los datos de parámetro dentro de una aplicación de bloque de función 440. Los objetos de visualización 565 permiten acceder a los grupos de parámetros ejecutando un solo requisito de comunicación. Los objetos de tendencia 560 permiten una recogida de muestras de parámetros que tienen que reportarse en una sola transferencia de comunicaciones. Por ejemplo, los objetos de visualización y tendencia asociados con los SISC pueden comunicarse utilizando mecanismos de comunicación normales o clientes/servidor, editor-suscriptor “SIS” u otros.

Los objetos de alerta 585 soportan el informe de los eventos a un dispositivo de interfaz y otros dispositivos de campo. Tras la detección de un evento significante, un bloque de función 530/530’ puede enviar un mensaje de alerta utilizando un objeto de alerta 585. Por ejemplo, un evento significante puede ser un evento que afecte la operación del sistema. Los bloques de función pueden reportar sus propios errores, alertar a los operarios acerca de problemas en una base “de tiempo real”, según se desee.

El tiempo del sistema 601 se proporciona mediante la gestión del sistema con respecto a las aplicaciones de bloque de función (es decir, uno o más recursos) 440 para usarse en operaciones de sincronización entre dispositivos de campo. Cada dispositivo 100/100’, 105/105’, 110/110’ mantiene su propio tiempo de sistema 601. Cada dispositivo 100/100’, 105/105’, 110/110’ utiliza su tiempo de sistema para controlar la ejecución de sus bloques de función internos. La representación del tiempo en alarmas, eventos e información de tendencia puede basarse en el tiempo del sistema 601 mantenido por cada dispositivo.

La gestión del sistema coordina la ejecución de los bloques de función 530/530’ de acuerdo con una programación del sistema. La programación del sistema es una lista de tiempos de ejecución para bloques de función dentro de un dispositivo. Adicionalmente, la ejecución de un bloque de función 530/530’ puede también invocarse con la terminación de la ejecución de otro bloque de función 530/530’. La gestión del sistema se describe con más detalle más adelante.

Marco de aplicación

Una vez que los componentes (es decir, los bloques y los objetos) se implementan, se completan o conectan mediante un marco de aplicación. El marco de aplicación coordina la comunicación entre los componentes internamente y externamente. La comunicación interna significa la comunicación entre los bloques de función 530/530’, independientemente de si están en el mismo dispositivo de campo. La comunicación externa significa la comunicación entre dispositivo de campo con bloques de función 530/530’ y de dispositivos de campo sin bloques de función. Idealmente, la conexión de estos bloques mediante el marco de aplicación da como resultado un sistema modular que permite que la funcionalidad de una aplicación sea más extensible y portátil. La funcionalidad es extensible el sentido adicional en que la funcionalidad puede fácilmente añadirse a una función o a un componente existente. La funcionalidad es portátil en el sentido en que la funcionalidad puede fácilmente moverse de una ubicación, dispositivo o componente en un sistema a otro, o incluso de un sistema a otro.

La Figura 9 muestra algunos ejemplos de las comunicaciones externas. Específicamente, la Figura 9A muestra la comunicación de dispositivos de campo 620 y de un dispositivo monitor 650, un dispositivo temporal 660 y un dispositivo de interfaz 670. A diferencia del dispositivo de campo 620, los otros dispositivos 650, 660, 670 contienen aplicaciones que no se implementan como bloques de función. El dispositivo de control 650 se conecta al marco de aplicación, pero no tiene una dirección de red. Un dispositivo de control controla las comunicaciones de la red (por ejemplo, una herramienta de diagnóstico puede ser un dispositivo monitor). Un dispositivo temporal 660 soporta los diagnósticos y ajusta los valores de parámetros. Un dispositivo de interfaz 670 proporciona una interfaz de operario, aplicaciones de control y/o soporte de configuración y de diagnóstico.

De forma similar, la Figura 9B muestra un ejemplo de las comunicaciones externas que se pueden soportar dentro de un subsistema de control SIS 910, tal como uno que puede existir en un bus 120’, y un subsistema de control no SIS 920, tal como uno que podría existir en un bus 120. En este ejemplo ilustrativo, el subsistema de control SIS incluye dos dispositivos SIS, el dispositivo SIS A 930 y el dispositivo SIS B 940, un subsistema de control no SIS 920 incluye dos dispositivos no SIS, el dispositivo C 950 y el dispositivo D 960. Una aplicación SIS A 970 se implementa entre los dispositivos A y B, 930/940, utilizando los SISC. Adicionalmente, una aplicación no SIS B 980 se implementa entre los dispositivos C y D 950/960, utilizando, entre otras cosas, los no SISC tales como los bloques de función estándares, los bloques transductores, los bloques de recurso y los objetos de enlace. Las comunicaciones entre los dispositivos SIS A y B 930/940 y lo dispositivos no SIS C y D 950/960 también se soporta utilizando los bloques de función, transductores y de recursos comunes, objetos de enlace, objeto de visualización, objeto de alerta y otros. Como se ha mencionado anteriormente, el dispositivo SIS A (o B) puede publicar información al dispositivo SIS B (o A) y a los dispositivos no SIS C y D. Sin embargo, los dispositivos no SIS sólo pueden publicar información a dispositivos no SIS. De forma deseable, cuando una implementación tal como aquella mostrada en la Figura 9B se implementa primero, los SISC en lo dispositivos/componentes de bus de campo se muestran y se registran con una facilidad de muestreo adecuada, tal

como una proporcionada por el bus de campo Foundation, para asegurar la interoperabilidad entre los dispositivos de bus de campo SIS y no SIS que utilizan los bloques de función estándares y/o los SISC.

Además, el acceso escrito a los SISC (por ejemplo, SISFB, SISTB y SISRB) puede restringirse a una lista de dispositivos de interfaz. La lista se preconfigura deseablemente en el dispositivo mediante un sistema de configuración. Los SISC sólo otorgan acceso por escrito a esta lista de dispositivos. El acceso escrito para los SISC pueden también "bloquearse" o de forma similar configurarse para evitar el cambio de cualquier parámetro relacionado con seguridad en tanto el dispositivo SIS o el SISC está en línea o de lo contrario en una condición no deseable (por ejemplo, durante ciertos procedimientos de mantenimiento potencialmente arriesgados para el humano o el equipo). Deseablemente, una alerta se genera a un operario en cualquier momento que se cambie un estado escrito. También, cada conexión entre los SISC se identifica por una Identificación de Conexión ("CK"). Las CK se describen con mayor detalle en lo sucesivo la presente memoria. Además, la Unidad de Datos de Protocolo ("PDU") que contiene los datos escritos o los de a un SISC dado se extiende para incluir un autenticador, tal como una Comprobación de Redundancia Cíclica ("CRC"). Una CRC de 32 bit se puede usar para autenticar la validez de los datos comunicados sobre un canal negro calculando la CRC utilizando los datos transmitidos, una identificación de conexión y otra información, como se describe con mayor detalle a continuación en la presente memoria. Las CRC y el cálculo de las CRC son bastante conocidos en la técnica. Adicionalmente, debe apreciarse que las CRC iguales o mayores que 32 bits se pueden utilizar en conjunto con las diversas realizaciones para autenticar las transferencias de datos en SIL 1, SIL 2, SIL 3 y/o niveles más altos.

La CRC-32 puede utilizarse para detectar y proteger mensajes inválidos, fallos de dirección, tales como enmascaramiento y expansión de un mensaje. La CRC-32 puede utilizarse también para proteger otros errores normalmente conocidos en las técnicas o condiciones de mensajería inválidas.

Además de interacciones externas e internas, una diversidad de otras posibles interacciones es bastante conocida para un experto en la materia. Por ejemplo, pueden existir interacciones con aplicaciones de configuración, interacciones con aplicación de interfaz humana, interacciones con otras aplicaciones de control, interacciones para el establecimiento de enlaces de bloque de función, interacciones con otros recursos, interacciones con gestión del sistema y muchas otras.

Estructura de Aplicación de Bloque de Función

Como se ha mencionado anteriormente, una aplicación de bloque de función 440, tanto para dispositivos SIS como no SIS, define la funcionalidad del dispositivo de campo, e incluye uno o más recursos 500/500'. Un recurso es una subdivisión lógica dentro de la estructura de soporte lógico y/o del soporte físico del dispositivo. Aunque no se muestra, las aplicaciones de bloque de función 440 se implementan generalmente utilizando múltiples recursos. Como se muestra en las Figura 8A y 8B, los recursos 500/500' que constituyen una aplicación de bloque de función 440 pueden modelarse como un conjunto de SISC (es decir, bloques u objetos) coordinados para ejecutar un conjunto relacionado de operaciones.

Un bloque es una unidad de procesamiento lógica del soporte lógico que comprende una copia nombrada de la estructura de datos del bloque y del parámetro especificada por un tipo de función. Una copia nombrada de bloque es una unidad de procesamiento del soporte lógico encapsulada, tal como un algoritmo o programa informático. El bloque se encapsula para crear un sistema modular con flexibilidad para actualizaciones y mejoras. La unidad de procesamiento del soporte lógico puede incluir un programa informático y parámetros. La unidad de soporte lógico se diseña para ser independiente de otros bloques y realizar una función que puede utilizarse en muchas otras aplicaciones de bloque de función.

Un bloque es identificable por su clase o subclase. La clase de un bloque indica sus parámetros, y cómo los parámetros afectan la ejecución de la unidad de procesamiento del soporte lógico. Una clase de bloque especifica los atributos comunes compartidos para todas las instancias de la clase, incluyendo los elementos de bloque (por ejemplo, eventos de entrada y de salida, parámetros contenidos y función común) y asociación con la función de recurso (por ejemplo, notificadores de alarma y servicios de bloque de función). Cada subclase de bloque asume todos los parámetros especificados por la clase, así como los parámetros adicionales atribuidos a la subclase.

Las clases de bloque se pueden clasificar como elementarías o compuestas. Una clase de bloque compuesta es una cuyo algoritmo requiere la invocación de funciones y/o bloques de componente del bloque compuesto. Un bloque elementario tiene un algoritmo fijo y no requiere el uso de funciones de componente o bloques de función. Los ejemplos específicos de los bloques elementarios y compuestos se describen con más detalle más adelante.

Soporte Físico de Aplicación de Bloque de Función

Cada dispositivo poder contener al menos una aplicación de bloque de función 440. Para ejecutar la aplicación de bloque de función 440, un dispositivo contiene normalmente un cierre de entrada 240, procesador 250, memoria 255, cierre de salida 260, y control de ejecución 265, como se muestra la Figura 3, así como la pila de comunicaciones 205 y una unidad de fijación del medio 612, como se muestra en las Figuras 8A y 8B.

La unidad de fijación del medio 612, tal como un adaptador de red, recibe las señales de otros dispositivos sobre el medio de transmisión 120/120' y traduce las señales en un mensaje para el procesador 250. Por ejemplo, la unidad de

fijación del medio 612 convierte o traduce un mensaje del procesador 250 en una señal para la transmisión a través del medio de transmisión 120/120', o una señal del medio de transmisión 120/120' en un mensaje para el procesador 250.

El cierre de entrada 240, el procesador 250, la memoria 255, y el cierre de salida 260 son para ejecutar los bloques transductores, los bloques de función y del bloque de recurso dentro de una aplicación de bloque de función. Específicamente, el cierre de entrada 240 recibe y mantiene los parámetros de entrada. Estos parámetros de entrada pueden ser constantes o recibirse de otros bloques de función. El procesador 250 ejecuta o procesa un programa informático o algoritmo en base a estos parámetros de entrada y cualquiera de los parámetros contenidos o almacenados. Estos parámetros se describen con mayor detalle a continuación. El procesador 250 puede ser, por ejemplo, un microprocesador o una matriz lógica programable. Cualquier programa o parámetro informático usados por el procesador 250 pueden almacenarse la memoria 255, que es preferiblemente EEPROM o FLASHROM. La funcionalidad de la aplicación de bloque de función 440 puede limitarse, sin embargo, por el tamaño de la memoria 255 y la velocidad del procesamiento del procesador 250. La salida del procesador 250 se envía a un cierre de salida 260.

El cierre de entrada 240 y el cierre de salida 260 son responsables para proteger los valores de parámetro de inferencias externas, tales como acceso escrito, en tanto se ejecuta el procesador 250. En otras palabras, una vez que el procesador 250 que está procesando las entradas, el cierre de entrada 240 y el cierre de salida 260 mantienen las entradas y las salidas constantes hasta que se completa el procesamiento.

Parámetros

Los parámetros definen las entradas, las salidas y los datos utilizados para controlar la operación de bloque. Los parámetros son accesibles a través de la red.

Un parámetro de entrada obtiene su valor a partir de una fuente externa al bloque. Un parámetro de entrada puede vincularse a un parámetro de salida de otro bloque dentro de su recurso 500/500' o dentro de otro dispositivo. Un parámetro de entrada es una variable de entrada o constante que se procesa mediante el algoritmo o programa de un bloque de función 530/530'.

Un parámetro de salida es un parámetro que puede enlazarse a parámetro de entrada de uno o más bloques. Los parámetros de salida contienen tanto valores como atributos de estado. El atributo de estado de salida indica la calidad del valor de parámetro generado.

Un parámetro contenido es un parámetro cuyo valor se configura, calcula, o ajusta mediante un operario o dispositivo de mayor nivel. En una realización, un parámetro contenido no puede vincularse a otro bloque de función de entrada o de salida, y por lo tanto no puede contener el atributo de estado.

IDENTIFICADORES DE PARÁMETRO

Cada parámetro puede caracterizarse por sus identificadores, almacenamiento, uso y relación con otros parámetros. Cada parámetro puede caracterizarse mediante más de un identificador. Por ejemplo, un parámetro dentro de un bloque se identifica únicamente por su identificación de dispositivo de parámetro, y un parámetro dentro de un sistema se identifica únicamente mediante su identificación y etiqueta de dispositivo. Las etiquetas pueden proporcionar una única referencia simbólica a cada bloque dentro del sistema.

El tipo de dato para un parámetro se especifica mediante sus índices de tipo de datos. El índice de tipo de datos es un índice del diccionario de objetos del tipo de datos. El índice de tipo de datos especifica la sintaxis independiente de máquina del parámetro. Normalmente, la sintaxis independiente de máquina del parámetro es una sintaxis abstracta. La capa de usuario 235 codifica/decodifica los datos de acuerdo con la regla de sintaxis de transferencia en la especificación de mensaje de bus de campo 230. Adicionalmente, una diversidad de otros parámetros puede también almacenarse en el diccionario de objetos 281 y referenciarse por su número índice del diccionario de objetos.

ALMACENAMIENTO DE PARÁMETROS

Los atributos de parámetros pueden clasificarse como dinámicos, estáticos o no volátiles. Los parámetros dinámicos son valores calculados mediante el algoritmo de bloque y por lo tanto no necesitan restaurarse después de un fallo de energía.

Los atributos estáticos son un valor específico, configurado que debe restaurarse después de un fallo de energía. Un dispositivo de interfaz 670 o un dispositivo temporal 660 puede escribir los atributos de parámetros estáticos en una base infrecuente. Los atributos de parámetros estáticos pueden seguirse mediante un dispositivo de configuración.

Los atributos de parámetros no volátiles se escriben sobre una base frecuente y el último valor guardado debe restaurarse mediante un dispositivo después de un fallo de energía. Puesto que los valores de estos atributos de parámetros se cambian constantemente, los valores pueden seguirse mediante un dispositivo de configuración.

Relaciones de parámetros

La ejecución de un bloque implica parámetros de entrada, parámetros de salida, parámetros contenidos y el algoritmo o programa informático almacenado por el bloque. El tiempo de ejecución de un algoritmo de bloque se identifica como un atributo de bloque. La duración del tiempo de ejecución es independiente de la implementación del soporte físico y del soporte lógico.

En los bloques simples, los parámetros de entrada se reciben antes de la ejecución de bloque. Cuando el bloque comienza la ejecución, los valores de entrada se introducen rápidamente para evitar que los mismos se actualicen en tanto se utilizan por el algoritmo.

Sin embargo, antes que se procesen estos parámetros de entrada, los parámetros de entrada se utilizan para determinar si el algoritmo puede conseguir el modo deseado. Una aplicación de bloque de función puede conseguir una diversidad de modos, tales como fuera de servicio (O/S), inicialización manual (Iman), intervención local (LO) apagado manual (Man), automático (Auto), cascada (Cas), cascada remota (Rcas) y modo de salida remoto (Rout). El fuera de servicio, la inicialialización manual, los modos de intervención local se describen a continuación. Para los SISFB, deseablemente, para los bloques de función de entrada, tales como los modos de soporte de entrada análoga o de entrada discreta O/S y Auto. De forma similar, para los bloques de función de salida, se pueden soportar los modos O/S, Cas y LO.

Cuando un bloque está en el modo OS, el bloque no está siendo evaluado, y la salida se mantiene en su último valor.

Cuando un bloque está en el modo IMan, la salida del bloque está ajustándose en respuesta del estado de parámetro de entrada de cálculo retroactivo. Cuando el estado indica que no existe trayectoria al elemento de salida final, entonces los bloques de control inicializan la proporción de una transferencia sin interrupciones cuando la condición se aclara. El parámetro de salida de cálculo retroactivo se soporta mediante todos los bloques de función de clase de salida y de control. El punto de referencia puede mantenerse u, opcionalmente, inicializar el valor de parámetro variable de proceso.

El modo LO aplica para los bloques de control y de salida que soportan un parámetro de entrada de seguimiento. El modo LO puede posibilitarse mediante un conmutador de bloqueo en el dispositivo o mediante una diversidad de otras formas. En el modo LO, la salida del bloque está siendo ajustada para seguir el valor del parámetro de entrada del seguimiento. El punto de referencia puede mantenerse u, opcionalmente, inicializar el valor de parámetro variable de proceso.

La determinación de si el bloque es capaz de conseguir el modo deseado se realiza comparando el atributo de modo real y el atributo de modo diana. El atributo de modo real refleja el modo de operación que está siendo capaz de conseguir el bloque. El atributo de modo diana indica qué modo de operación se desea para el bloque. El modo diana se ajusta normalmente mediante una aplicación de control o mediante un operario a través de una aplicación de interfaz humana.

Una vez que se determina el modo real, la ejecución del bloque progresa y la salida se genera. Si se detectan condiciones de alerta, se actualizan parámetros de salida de alarma y de evento para informar mediante un objeto de alerta. Cuando se completa la ejecución, las salidas se introducen haciéndolas disponibles para el acceso externo. Antes de introducirse, sólo los valores anteriores están disponibles para el acceso externo.

Componentes de Recurso

Como se ha mencionado anteriormente, una aplicación de bloque de función 440 contiene uno más recursos, y un recurso 500/500' incluye uno o más bloques. Un bloque se puede identificar por su clase o su subclase. La clase de un bloque indica sus parámetros, y cómo estos parámetros afectan a la ejecución de su algoritmo o programa. La Sección de Componentes de Recurso proporciona los modelos formales para las clases preferidas. Las clases preferidas incluyen una clase de recurso, clase de objeto de directorio, clase de objeto de bloque, clase objeto de parámetro, clase de objeto de enlace, clase de objeto de alerta, clase de objeto de tendencia, clase de objeto de visualización, clase de objeto de dominio, clase de objeto de invocación de programa y clase de objeto de acción. En realizaciones alternativas, alguien experto en la materia podría definir un sistema con más, menos o diferentes clases. Nuevamente, el objeto de dominio, los objetos de invocación de programa y los objetos de acción no se soportan en el SISC.

Clase de Recurso

La clase de recurso definida en una realización preferida especifica los atributos descriptivos de recurso. El diccionario de objetos de cada recurso contiene una descripción de los componentes contenidos dentro del recurso. La clase de recurso indica los siguientes atributos: nombre de recurso, nombre del vendedor, nombre del modelo, revisión, estado lógico, estado físico, diccionario de objetos y, en los dispositivos SIS, el nivel SIL soportado por el dispositivo.

El nombre del vendedor identifica al vendedor del soporte lógico y/o soporte físico asociado con el recurso. El nombre de modelo especifica el modelo del soporte lógico y/o del soporte físico asociado con el recurso. El atributo de revisión es el nivel de revisión del soporte lógico y/o del soporte físico asociado con el recurso. El atributo de estado lógico contiene información acerca de la funcionalidad de comunicación asociada con el recurso. El atributo de estado físico proporciona un amplio resumen del componente del soporte físico asociado con el recurso. El diccionario de objetos

contiene los atributos de un objeto de directorio del diccionario de objetos, del bloque de recurso y otros objetos específicos del proceso de aplicación del bloque de función 440. Cada uno de estos atributos es accesible a través de la especificación de mensaje de bus de campo 230.

Estos atributos y los atributos definidos para cualquier clase o subclase solo son ejemplos de los atributos que podrían utilizarse. En realizaciones alternativas, la clase de recurso o cualquier otra clase o subclase podría incluir más, menos o diferentes atributos. Este concepto aplica para todas las clases y subclases descritas en esta memoria descriptiva.

Objeto de directorio

Un objeto de directorio actúa como una guía de otros bloques y objetos dentro de un recurso o aplicación de bloque de función 440. El objeto de directorio contiene una lista de referencia de los otros bloques de objetos que constituyen un recurso o aplicación de bloque de función 440. Esta información puede leerse mediante un dispositivo de interfaz o dispositivo temporal que desee acceder a los objetos en el diccionario de objetos. La clase de objeto de directorio se define como incluyendo los siguientes atributos: identificación del miembro; índice de inicio del directorio de objeto estático; tipo de datos; entradas del subíndice; longitud de datos; uso; almacenamiento; lista de valores válidos; valor inicial e identificación de artículo.

El atributo de identificación de miembro es el único número que identifica la función del directorio. El índice es el índice del objeto de directorio en el diccionario de objetos. Los diferentes tipos de datos incluyen el tipo Meta o el tipo nombre. El tipo meta indica el tipo de objeto. El tipo nombre específica el nombre de tipo de datos del objeto. Las entradas de subíndice permiten que los atributos de un objeto de directorio se asocien individualmente a través del servicio de lectura y de escritura. El atributo de longitud de datos especifica el número de bits reservados para representar los valores de subíndice en el directorio. El atributo de uso indica que éste es un objeto contenido y que puede no referenciarse mediante los objetos de enlace para la conexión con los parámetros del bloque de función. El atributo de almacenamiento indica si el parámetro está almacenado en la memoria estática. La lista de valores válidos especifica los valores permitidos para los atributos de subíndice del objeto de directorio. El valor inicial especifica el valor inicial asignado a los atributos de subíndice del objeto de directorio, y la identificación de artículo identifica la descripción del objeto.

Objeto de Bloque

La clase de objeto de bloque especifica las características comunes a los bloques de función, bloques transductores y bloques de recurso. En el diccionario de objetos, los parámetros siguen continuamente después del objeto de bloque, teniendo cada uno un índice. La clase de objeto de bloque se define mediante los siguientes atributos: identificación del miembro; índice de bloque; tipos de datos; subíndice, longitud de datos; uso; almacenamiento; listas de parámetros; listas de valores válidos e identificación de artículo. La identificación de miembro identifica la función del bloque. El índice de bloque es el índice del objeto de bloque en el diccionario de objetos. El tipo de datos incluye el tipo Meta y el tipo nombre. El tipo Meta indica el tipo de objeto. El tipo nombre específica el nombre de la estructura de datos del bloque. El subíndice incluye atributos, tales como etiqueta de bloque, identificación de miembro, identificación de artículo, revisión, perfil, revisión de perfil, tiempo de ejecución, periodo de ejecución, número de parámetros, próximo bloque en ejecutarse, observaciones de inicio, número de objeto en 3 visualizaciones y número de objeto en 4 visualizaciones. El atributo de longitud de datos es igual a 62. La lista de parámetros incluye revisión estática, descripción de etiqueta, estrategia, tecla de alerta, modo y error de bloque. Los atributos restantes se han descrito anteriormente.

Las tres subclases de la clase de objeto de bloques usadas en una realización preferida son los objetos de bloque de recursos, los objetos de bloque transductores y los objetos de bloque de función.

OBJETO DE BLOQUE DE RECURSO

El objeto de bloque de recurso define las características específicas del soporte físico de su recurso asociado. Debido a que el objeto de bloque de recurso es una subclase del modelo de objeto de bloque, el objeto de bloque de recurso asume la lista de parámetros atribuidos al objeto de bloque, así como sus propios atributos adicionales. Los atributos adicionales en la subclase de bloque de recurso son: estado de recurso, prueba, recurso, parámetros contenidos adicionales; tiempo de ejecución=0, periodo de ejecución=0 y próximo bloque en ejecutarse=0.

Un bloque de recurso aísla los bloques de función del soporte físico conteniendo un conjunto de parámetros de soporte físico independientes de implementación. El bloque de recurso se especifica por el fabricante; y todos sus parámetros se definen como contenidos.

OBJETOS DE BLOQUES TRANSDUCTORES

Los objetos de bloques transductores se definen para desacoplar los bloques de función de las funciones I/O locales requeridas para leer el sensor físico y el comando físico. El bloque transductor se puede ejecutar tan frecuentemente como sea necesario para obtener datos a partir de los sensores sin interrumpir los bloques de función que utilizan los

datos. Los bloques de función aíslan también el bloque de función de las características de especificación del fabricante de un dispositivo I/O.

El objeto de bloque transductor es una subclase del objeto de bloque y asume todos los atributos de la clase de bloque. Los atributos adicionales de la subclase de bloque transductor son: parámetros contenidos adicionales; tiempo de ejecución=0; periodo de ejecución=0 y próximo bloque en ejecutarse=0.

OBJETOS DE BLOQUES DE FUNCIÓN

Los bloques de función representan las funciones de automatización básicas realizadas por un recurso, tales como una entrada análoga o discreta; los bloques de función son los medios de definición primarios que monitorean y controlan una aplicación de bloque de función. Se diseñan para ser independientes en lo posible de las especificaciones de los dispositivos I/O y de la red. Los bloques de función trabajan procesando parámetros de entrada y entradas de los bloques transductores (u otros bloques de función) de acuerdo con un algoritmo especificado y un conjunto interno de parámetros contendidos. Los mismos producen también parámetros de salidas y salida a los bloques transductores o la entrada de otros bloques de función.

En base al algoritmo de procesamiento, puede proporcionarse una función de monitoreo, cálculo o control deseada. Los resultados de la ejecución del bloque de función pueden reflejarse en la salida a un bloque transductor o a uno o más parámetros de salida que pueden vincularse a otros bloques de función o directamente al soporte físico del dispositivo.

Los bloques de función son una subclase de la clase objeto. Los atributos adicionales definidos en la subclase de bloque de función son el subíndice del tiempo de ejecución, el periodo de ejecución, el número de parámetros, próximo bloque en ejecutarse y parámetros adicionales.

El atributo de subíndice define los atributos de un objeto que puede accederse individualmente a través de servicios de lectura y de escritura utilizando el número subíndice con el número índice de objeto. Los números subíndices se definen en base al tipo Meta.

El parámetro de tiempo de ejecución del objeto de bloque de función denota el tiempo requerido para que se ejecute un bloque de función. El tiempo de ejecución puede dividirse en tres componentes: pre-procesamiento (es decir, introducción de valores de parámetros); ejecución; y post-procesamiento (es decir, se actualizan los valores de salida de bloque, alarma y parámetros de tendencia asociados).

Para proporcionar un comportamiento consistente, el algoritmo de bloque ejecutado durante la ejecución de componente se divide en las siguientes etapas. Primero, el algoritmo determina el atributo de modo real del parámetro de modo. Este cálculo estará basado en el modo diana y en el estado de los atributos de las entradas como se ha descrito anteriormente. Segundo, el algoritmo calcula el punto de referencia, si el punto de referencia se define para el bloque de función, el cálculo del punto de referencia estará basado en el modo real, los parámetros de entrada del punto de referencia, tales como cascada y cascada remota, y cualquier estado de entrada de trayectoria hacia atrás. También, el valor del parámetro controlado, variable de proceso, puede utilizarse para seguir el punto de referencia. El punto de referencia resultante se muestra en un parámetro del punto de referencia. Un ejemplo del punto de referencia es la temperatura de ajuste de un termostato (por ejemplo, 72ºC). En otros ejemplos, el punto de referencia cambiará frecuentemente.

Tercero, el algoritmo ejecuta el control o cálculo del algoritmo para determinar el valor y estado de los parámetros de salida. Las condiciones que determinan el atributo de estado en los parámetros de salida. Los atributos de valores de los parámetros de entrada del bloque y los parámetros contenidos, el modo real y el punto de referencia de trabajo se utilizan en este algoritmo. En general, el cálculo del modo real y el uso del modo real en el algoritmo se toman en cuenta para el estado de las entradas críticas.

Cuarto, la fase de ejecución calcula los parámetros de salida. Esta etapa aplica solo a bloques de salida, bloques de control y bloques de cálculo designados para usarse en la trayectoria de cascada.

El periodo de ejecución de un bloque de función se programa típicamente en una base periódica: el periodo de ejecución se especifica por el usuario en base a los requisitos de control y de monitorización específicos para una aplicación. Los servicios de gestión de sistema coordinan la ejecución del bloque de función. La base de información de gestión, que incluye la programación del sistema, se almacena en su propio recurso en el dispositivo. El periodo de ejecución del bloque de función se especifica para un bloque en el tiempo de la capa de enlace de datos. A través de la capacidad de programación proporcionada por la gestión del sistema, es posible realizar en fases o etapas la ejecución de los bloques en un dispositivo en el que sus periodos de tiempo de ejecución son los mismos o son múltiplos enteros entre sí. La gestión del sistema se describe con mayor detalle a continuación.

El atributo de “número de parámetro” dentro del objeto del bloque de función es el número total de objetos parámetros asociados con el bloque de función, incluyendo el objeto de bloque.

El atributo “próximo bloque en ejecutarse” del objeto del bloque de función especifica el próximo bloque de función dentro de un dispositivo a ejecutarse para conseguir mínimo retraso en la ejecución dentro de un dispositivo. Si no existe próximo bloque de función, entonces el próximo bloque en ejecutarse es 0. Por lo tanto, cuando múltiples bloques de función necesitan ejecutarse en serie dentro de un dispositivo, un usuario puede especificar el primer bloque de función en ejecutarse en la cadena. A través del atributo del próximo bloque en ejecutarse, el orden de ejecución puede predeterminarse.

El atributo “lista de parámetros” del objeto del bloque de función realiza la lista de los parámetros de entrada, salida y contenidos dentro de un bloque de función.

En base a los parámetros comunes y al comportamiento, una realización preferida define también las subclases correspondientes con la subclase del bloque de función, incluyendo: bloque de función de entrada; bloque de función de salida; bloque de función de control y bloque de función de cálculo.

La subclase de bloque de función de entrada recibe mediciones o valores físicos del bloque transductor. La subclase del bloque de función de entrada incluye un parámetro de simulación por el que pueden intervenirse el valor y el estado del transductor. Los otros parámetros del bloque de función de entrada incluyen preferiblemente: variables de proceso; salida primaria; número de canal; y parámetros adicionales.

La subclase del bloque de función de salida actúa tras la entrada de los otros bloques de función y reenvía sus resultados a un bloque transductor de salida. La subclase de bloque de función de salida soporta el parámetro de salida de cálculo retroactivo y el parámetro simulado. Los atributos del bloque de función de salida adicionales son: punto de referencia, parámetro simulado, entrada cascada; salida de cálculo retroactivo; cascada remota dentro; cascada remota fuera y número de canal.

La subclase del bloque de función de control actúa tras las entradas de los otros bloques de función para producir valores que se envían a los otros bloques de función de control o de salida. Los atributos adicionales para el bloque de función de control son: salida primaria; cálculo trasero; variables de proceso; punto de referencia; entrada primaria; entrada cascada; cascada remota dentro; salida remota dentro; salida de cálculo retroactivo; cascada remota fuera; salida remota fuera y parámetros adicionales. Los parámetros del bloque de función de cálculo adicional son: entrada de cálculo retroactivo; salida de cálculo retroactivo y parámetros adicionales.

Objetos de Parámetros

Los objetos de parámetro se definen para permitir que se tenga acceso a los atributos del bloque de función, del bloque transductor y del bloque de recurso a través del bus. Los atributos definidos en el modelo de objeto de parámetros básicos son: identificación del miembro; índice de parámetro; índice relativo; tipo de datos; subíndice; longitud de datos; unidades; uso; almacenamiento; lista de valores válidos; valor inicial e identificación de artículo. No todos los parámetros mencionados en la lista se requieren en un bloque particular. Adicionalmente, varias subclases de la clase de objeto de parámetro pueden identificarse incluyendo los objetos de parámetros de salida, los objetos de parámetros de entrada y los objetos de parámetros contenidos.

Objetos de Enlace

Los objetos de enlace 570/570’ proporcionan la correlación entre los recursos y la información intercambiada por medio de una red de comunicación como se ilustra en las Figuras 8A y 8B. Los datos y eventos del proceso a intercambiarse entre de los bloques de función dentro de un recurso o entre recursos se pueden definir a través de objetos de enlace. Además, el intercambio de comunicación para el soporte de tendencias y alertas puede identificarse utilizando objetos de enlace.

Los objetos de enlace 570/570’ se definen en dispositivos de campo asociados con el proceso de aplicaciones de bloque de función. Los objetos de enlace 570/570’, referenciando el VCR apropiado, pueden usarse para acceder, distribuir o intercambiar objetos individuales. Además, los objetos de enlace definen la asociación entre los parámetros de entrada y de salida, e informes de tendencias que deben recibir los dispositivos de interfaz.

En las implementaciones SIS, se utiliza una seguridad/SISRP extendida. El SISRP proporciona autenticación de comunicaciones entre los SISC de tal manera que puede conseguirse el SIL-3 y el SIL-2. En particular el SISRP protege contra errores que puedan surgir durante el uso del canal negro. Tales errores puede incluir: transmisión de fallos de bit, tales como cuando un solo bit o múltiples bits en un mensaje cambian el estado del canal negro; retransmisión, cuando el canal negro retransmite inadvertidamente un mensaje; omisión, cuando el canal negro pierde un mensaje o mensajes; inserción/expansión, cuando un mensaje se genera erróneamente y/o se inserta o expande en el canal negro; orden errada, cuando el canal negro suministra mensajes en el orden errado; retraso, cuando el canal negro retrasa la transmisión o recepción de un mensaje durante un periodo de tiempo; enmascaramiento, cuando el canal negro suministra mensajes al punto final errado o a múltiples dispositivos que tienen la misma dirección de red; falta en cola, cuando el canal negro retrasa un mensaje por más de la velocidad de transmisión pero por menos del tiempo de retraso necesario para causar una parada; errores de comunicación y de programación de bloque de función; errores de gestión de sistema y de configuración y otros.

Un mecanismo por el que el SISRP protege contra los errores mencionados anteriormente es utilizando una secuencia numérica. Un número de dieciséis (16) bits puede utilizarse para cada conexión VCR para identificar una secuencia de mensajes enviados entre un SISC de envío y un SISC de recepción. El SISC de recepción mantiene un número índice correspondiente. Cuando se opera apropiadamente, el número de índice se actualiza tanto en el emisor como en el receptor con cada mensaje enviado. Cuando el número máximo de mensajes se recibe, los números se distorsionan y empiezan a contarse desde cero (0) nuevamente. Asumiendo una velocidad de transmisión de mensajes de un mensaje por cada diez milisegundos, que se puede conseguir utilizando una conexión de Ethernet de Alta Velocidad, los números de secuencia deberían distorsionarse cada 655 segundos. A velocidades de transmisión de mensajes menores, los números de secuencia se distorsionarían menos frecuentemente.

Si el número de secuencia enviado y el número de secuencia esperado (es decir, el número de secuencia esperado en el componente de recepción) no coinciden, entonces los datos se consideran “sin interés” o no usables. Si una condición de datos sin interés de este tipo se repite a sí misma un número configurable de veces dado entonces un enlace entre los componentes puede establecerse como deficiente. Tales números de secuencia y el enlace necesitarán, en general, después reajustarse antes que se reanuden las comunicaciones entre los componentes efectuados. Además, en una relación editor-suscriptor, el número de secuencias se reajusta deseablemente tanto en el emisor como en el receptor cuando sea que dos mensajes consecutivos correctos se hayan recibido. Para una relación cliente-servidor, si los números de secuencia están fuera de sincronización, entonces la conexión se aborta y los números de secuencia se reajustan tras la conexión que se está restableciendo. Sin embargo, en otras realizaciones, los números de secuencia pueden reajustarse usando otros procesos y técnicas.

Además, utilizando un número de secuencia en el SISRP se proporciona protección para la retransmisión, orden errada y errores de inserción/expansión. Adicionalmente, las identificaciones de secuencia se utilizan también en conjunto con las CK en el SISRP para protegerse contra errores de enmascaramiento.

Como se ha mencionado anteriormente para implementaciones SIS, las identificaciones de conexión (“CK”) se utilizan de forma deseada. Tales CK son una parte del SISRP y se proporcionan normalmente como un parámetro en los SISLO. La CK es una única identificación que se asigna mediante un sistema de configuración para la conexión entre un dispositivo de interfaz y los SISC (es decir, SISFB, SISTB y SISRB) para una conexión cliente-servidor. También, una única CK se asigna para cada conexión editor-suscriptor, en el que todos los suscriptores tienen un editor dado que se configura deseablemente para usar la identificación del editor. La utilización de la CK y de los números de secuencia en conexiones cliente-servidor y editor-suscriptor se describe además con mayor detalle en lo sucesivo en la presente memoria.

También, para implementaciones SIS, los SISLO pueden incluir un parámetro de Acceso SIS. Este parámetro, cuando se ajusta, especifica que los informes de lectura y escritura se procesan utilizando el SISRP extendido.

Objetos de Alerta

Los objetos de alerta se utilizan para comunicar mensajes de notificación cuando se detectan alarmas o eventos. Un evento en una ocurrencia instantánea que es significante para la ejecución del bloque de programación y para la visualización operativa de una aplicación de bloque de función 440. Una alarma es la detección de un bloque que deja un estado particular. La clase de objeto de alarma permite que las alarmas y eventos se informen a un dispositivo responsable para la gestión de alarmas.

En base al tipo de información de alarma y de evento que puede informarse mediante los bloques, la realización preferida designa tres subclases de objetos de alarma. Estas son alertas análogas, alertas discretas y alertas de actualización. Las alarmas análogas se utilizan para informar alarmas o eventos cuyos valores se asocian como un punto flotante. Las alarmas discretas se utilizan para reportar alarmas o eventos cuyos valores asociados son discretos. Las alarmas de actualización se utilizan para reportar un cambio en los datos estáticos de un bloque.

Objetos de tendencias

Los objetos de tendencias soportan las gestión y el control de los bloques de función proporcionando visibilidad en una información histórica para revisar sus comportamientos. En base al tipo de información recogida, una realización preferida define tres subclases de objetos de tendencia. Estas subclases son la subclase flotante de tendencia, la subclase discreta de tendencia y la subclase de cadena de bits de tendencia. La clase flotante de tendencia recoge los valores y el estado de una entrada puntual flotante y de parámetros de salida. La subclase discreta de tendencia recoge los valores y el estado de parámetros de entrada y de salida discretos. La subclase de cadena de bits de tendencia recoge los valores y el estado de los parámetros de entrada y de salida de cadena de bits.

Objetos de Visualización

Los objetos de visualización soportan la gestión y el control de los bloques de función proporcionando “visibilidad” dentro de sus configuraciones y operaciones. En otras palabras, los objetos de visualización le permiten al usuario monitorear u “observar” los datos asociados con la operación, diagnóstico y configuración del sistema, la aplicación de bloques de funciones 440 o recursos 500. En una realización, existen cuatro subclases de las clases de objetos de visualización.

Estas subclases son visualización 1, visualización 2, visualización 3 y visualización 4. La visualización 1 permite el acceso a los valores parámetros de operación dinámicos. La visualización 2 permite el acceso a los valores parámetros de operación estáticos. La visualización 3 permite el acceso a todos los valores de parámetros dinámicos. La visualización 4 permite el acceso a los otros valores de parámetros estáticos.

Objeto de Dominio

Para un dispositivo no SIS, un objeto de dominio 580 soporta los servicios de descarga que pueden usarse para cargar datos de un cliente dentro de un dominio del servidor. Los datos pueden transmitirse desde el dominio del servidor hasta un cliente a través del servicio de carga de dominio. Los objetos de dominio son parte de la memoria. Los mismos contienen programas o datos. Los dominios con códigos y datos se combinan en un programa ejecutable utilizando un objeto de invocación de programa.

Otros Objetos

Para un dispositivo no SIS, un objeto de invocación de programa 590 proporciona servicios para vincular los dominios con un programa, para iniciar este programa, detenerlo y suprimirlo. Los objetos de acción pueden opcionalmente soportarse mediante un recurso en un dispositivo no SIS. A través de un objeto de acción, un bloque u objeto individual dentro de un recurso puede suprimirse en el dispositivo no SIS. Para los dispositivos SIS, los objetos de acción no se soportan normalmente debido a que es indeseablemente común suprimir bloques u objetos críticos importantes de seguridad.

Bloque de Función – Correlación

Para la implementación de una aplicación de bloque de función 440, la aplicación de bloque de función 440 se correlaciona en el dispositivo de campo virtual de la especificación de mensaje de bus de campo 230, como se muestra en la Figura 7A. Los objetos de campo virtual que son las herramientas preferidas durante la descripción de una aplicación de bloque de función 440 son: objetos variables; objetos de gestión de eventos; objetos de dominio (solo en dispositivos no SIS); y objetos de invocación de programa (solo en dispositivos no SIS).

Los objetos variables son un tipo de parámetro de bloque. Otros tipos de parámetros de bloque son simples, de matriz o registro. Los objetos de registro soportan los objetos de tendencia, de acción y de enlace. Agrupar la información para el acceso puede realizarse utilizando objetos de visualización variables.

Los objetos de notificación de eventos se utilizan para una notificación de alarma y de evento. Los objetos de dominio, que preferiblemente no están disponibles en los dispositivos SIS, son un programa informático que puede cargarse en una memoria utilizando los servicios de descarga de dominio. Los servicios de invocación de programa, que preferiblemente no están disponibles en los dispositivos SIS, pueden controlar la inicialización de la aplicación de bloque de función. Tales dispositivos incluyen; iniciar, detener y reajustar.

La tabla a continuación se utiliza para mostrar cómo el modelo de aplicación del bloque de función puede correlacionarse directamente con los objetos definidos en el diccionario de objetos.

Modelo de Bloque de Función		**Correlación con FMS**	
Recurso		VFD	
Directorio		Objeto del Directorio	
Bloque		Registro	Matriz
	-Parámetros		
	-Visualizaciones		Variables, Matriz y Registros Simples
			Listas de Variables
Objeto de Enlace		Registro	
Objeto de Alerta		Evento	
Objeto de Tendencia		Registro	
Invocación de Programa		Invocación de Programa	
Dominio		Dominio	

Modelo de Bloque de Función		Correlación con FMS	
Acción		Registro	

Para coordinar la correlación de los modelos de bloque de función con el diccionario de objetos, el lenguaje de descripción de dispositivo (descrito con más detalle más adelante) puede utilizarse para describir el bloque de función y los parámetros de bloque de soporte utilizados mediante la herramienta de configuración. Una descripción de este tipo se conoce como una “descripción de dispositivo”. En muchos casos, la “descripción de dispositivo” se utiliza en los puestos de configuración y de interfaz. Sin embargo, en algunos casos, todo o parte de la descripción del dispositivo puede almacenarse en el dispositivo de campo. Cuando la descripción del dispositivo se almacena en el dispositivo de campo, puede residir en su propio diccionario de objetos en un recurso separado de aquél utilizado para la aplicación del bloque de función. Para acceder a la información de descripción de dispositivo, cada bloque mantiene un número de referencia de descripción de dispositivo asociado.

El dispositivo de campo virtual recoge los bloques y los objetos mencionados anteriormente en un diccionario de objetos. Dentro del diccionario de objetos, cada bloque u objeto se abarca por un número índice y se identifica mediante una descripción de objeto. Las descripciones de objetos contienen generalmente un índice, código de objeto, y otros atributos de objetos adicionales, y referencias específicas del sistema con respecto al objeto real.

Números índices

Los números índices pueden agruparse de acuerdo con sus tipos o estructura de datos, o si el objeto es estático o dinámico. Los índices de objeto 1-255 pueden reservarse para usar normalmente tipos de datos y estructuras de datos. Como se muestra en la tabla a continuación, los índices 1-14 y 21 se definen como tipos de datos y la especificación de mensaje de bus de campo 230, y los índices 64-86 son estructuras de datos utilizadas normalmente, que hacen referencia a la definición de objetos de registro. Estos índices son los mismos que los números índices 285 mostrados en la Figura 6. La Figura 10 ilustra cómo estos números índices pueden agruparse también por si el objeto es estático o dinámico.

Índice	Tipo	Nam
1	Datos	Booleano
2	Datos	Entero 8
3	Datos	Entero 16
4	Datos	Entero 32
5	Datos	Sin signo 8
6	Datos	Sin signo 16
7	Datos	Sin signo 32
8	Datos	Punto Flotante
9	Datos	Cadena Visible
10	Datos	Cadena de Octeto
11	Datos	Fecha
12	Datos	Hora del Día
13	Datos	Diferencia de Tiempo
14	Datos	Cadena de Bit
21	Datos	Valor de tiempo
64	Estructura	Bloque
65	Estructura	Valor y Estado - Flotante
66	Estructura	Valor y Estado - Discreto

Índice	Tipo	Nam
67	Estructura	Valor y Estado - Cadena de Bits
68	Estructura	En aumento
69	Estructura	Modo
70	Estructura	Permisos de Acceso
71	Estructura	Alarma-Flotante
72	Estructura	Alarma-Discreta
73	Estructura	Evento-Actualización
74	Estructura	Alarma-Resumen
75	Estructura	Alerta-Análoga
76	Estructura	Alerta-Discreta
77	Estructura	Alerta-Actualización
78	Estructura	Tendencia-Flotante
79	Estructura	Tendencia-Discreta
80	Estructura	Tendencia-Cadena de Bits
81	Estructura	Enlace FB
82	Estructura	Simulado-Flotante
83	Estructura	Simulado-Discreto
84	Estructura	Simulado-Cadena de Bits
85	Estructura	Ensayo
86	Estructura	Acción-Representar/Suprimir

Todas las descripciones de objetos en el diccionario de objetos distintas a las descripciones de datos y estructuras de datos pueden soportar extensiones. Por ejemplo, el número índice de una descripción de objeto distinto a un tipo o estructura de datos puede cambiarse sin afectar a los otros objetos. Además, la descripción del objeto puede mejorarse también o actualizarse sin afectar a los otros objetos.

Diccionario de objetos

El diccionario de objetos se define para actuar como una guía de la información dentro de una aplicación de bloque de función 440. El diccionario de objetos 281 es una lista de referencias a los objetos que constituyen esa aplicación de bloque de función. Esta información puede leerse mediante un dispositivo de interfaz que desee acceder a los objetos en el diccionario de objetos.

El objeto de directorio del diccionario de objetos 282 se definirá como el primer índice en el diccionario de objetos estático (S-OD) 700, mostrado en la Figura 10. El punto de inicio del diccionario de objetos estático se define mediante la descripción del objeto del diccionario de objetos que reside en el Índice Cero. Además, la descripción del diccionario de objetos identifica el índice de inicio, la longitud de la lista dinámica o de la lista de variables (DV-OD) 710 y la lista dinámica de la invocación de programas (DP-OD) 720 asociada con los objetos de visualización y los objetos de invocación de programas.

El directorio puede construirse lógicamente concatenando los objetos de directorio y consiste en un encabezado seguido por las entradas de directorio. Una matriz desfasada se especifica desde el inicio del directorio lógico. El directorio lógico puede pensarse como una sola matriz compuesta de todos los casos de objeto de directorio. El encabezado se presenta normalmente en el primer objeto de directorio. Los bloques que residen en un recurso se identifican en el diccionario de objetos mediante el objeto de directorio. Cada instancia de un bloque de recurso 510, bloque de función 530 o bloque transductor 520 consiste en un objeto de bloque y parámetros asociados. El objeto de bloque referencia su objeto de visualización asociado 565.

El objeto de bloque es la identificación principal utilizada en la referenciación de una instancia de un bloque. Define la etiqueta de bloque, el tiempo de ejecución, el perfil y el número de parámetros de bloque. También, define la ubicación

de inicio y el número de objetos de visualización para este bloque. Los parámetros de un bloque se ubican continuamente en el diccionario de objetos que sigue al objeto de bloque. Los valores de parámetro de bloque pueden accederse a través de estos objetos de parámetro. También, los objetos de parámetro de bloque se restringen generalmente a los parámetros variables simples, parámetros de serie y parámetros de registro.

En una realización, varias estructuras de datos se han estandarizado para el proceso de aplicación del bloque de función.

Subfunciones Comunes

Esta sección contiene descripciones de subfunciones comunes para muchos bloques. Una función de control de procesos tiene los siguientes elementos: (1) una o más entradas; (2) una o más salidas; (3) información de escalada; (4) un selector de modo; (5) un algoritmo seleccionado; (6) un conjunto de parámetros de datos visibles; y (7) un conjunto de datos internos. Cada uno de estos elementos representa bien sea datos estáticos o datos dinámicos. Los datos estáticos son datos que se cambian muy pocas veces, en tanto que los datos dinámicos pueden cambiarse con cada evaluación de bloque.

Cada instancia de un bloque se procesa de acuerdo con la selección de algoritmo en el momento determinado mediante una ejecución de bloque y un programador de comunicación combinados. La información de programación contenida a menudo en los parámetros de un bloque es el periodo de ejecución y el tiempo de ejecución máximo.

Conexiones

Una entrada de bloque contiene los datos leídos de las salidas de otros bloques. Si un bloque no recibe una entrada de otro bloque, una entrada constante puede introducirse. La permanencia del valor depende del tipo de memoria para almacenarlo. El tipo de memoria usado depende de los parámetros. Por ejemplo, la memoria volátil es suficiente para un parámetro que cambia frecuentemente. La memoria no volátil puede utilizarse, por ejemplo, para puntos de referencia. Las salidas de bloque contienen el resultado de la evaluación de bloque, o una entrada de operario si el modo es manual.

Tanto las entradas como las salidas comprenden un valor de campo y un estado de campo. El valor de campo contiene atributos de calidad, subestados y límites. Es el mismo para todas las entradas y salidas.

Las clases de bloques de función de entrada y de salida deben intercambiar datos con el dispositivo de soporte físico y están completamente bajo el control del fabricante quien escribe el código del dispositivo, ya que estos datos nunca van a través del sistema de comunicación. Como alternativa, muchos bloques proporcionan parámetros que pueden escribirse o leerse mediante dispositivos remotos operando una aplicación de control de patentada. Para conducir un intercambio de este tipo, el dispositivo remoto debe ejecutar un algoritmo de utilización de protocolo de enlace antes de la escritura o el bloque puede ignorar la entrada.

Simulación

Los bloques de función de clase de entrada y de salida pueden tener un parámetro de simulación, que tiene un par de valores de estados y un conmutador habilitado. Este parámetro actúa como un conmutador en la interfaz entre un bloque de función y el bloque transductor asociado o canal de soporte físico. Para las entradas, el valor del transductor y el estado se reciben del bloque transductor o canal de soporte físico si se deshabilita el conmutador. Cuando el conmutador habilitado está encendido el parámetro de simulación y los valores de estado se reciben del bloque de función, y se ignora el bloque transductor o canal de entrada.

Para las salidas, el valor simulado y el valor de estado se convierten en el valor y en el estado de lectura cuando está encendido el conmutador habilitado, y el conmutador transductor se ignora.

La información de escalamiento se utiliza con dos propósitos. Los dispositivos de visualización utilizan el intervalo de gráficos de barra y la tendencia. Los bloques de control utilizan el intervalo como el porcentaje de separación, de manera que la constante de calibración permanece sin valor.

Además en los SISC, cuando se activa el bloque escrito en un recurso, deseablemente las capacidades de simulación se desactivan.

Modos

Los bloques pueden tener también un parámetro de modo que determina el recurso de los datos que tienen que usarse para la entrada y salida del bloque. Los bloques se configuran deseablemente para permitir el modo fuera de servicio (O/S) y pueden soportar al menos un modo diferente.

Los modos permitidos se aplican al modo diana. Una respuesta escrita al modo diana se rechaza si no coincide con la lista permitida. Un dispositivo de configuración no debería permitir un modo que no está soportado. Si el modo real no está restringido en el modo permitido, debido a que se adquieren algunos otros modos para la inicialización.

Tabla de Acceso a Parámetros

Una tabla de acceso existe normalmente para cada bloque. El propósito de esta tabla es definir la posición relativa de los parámetros dentro de cada bloque, y definir los contenidos de las vistas estándares de los parámetros.

Los parámetros de bloque que necesitan comunicarse a través del bus pueden variar dentro de la aplicación. Para permitir las aplicaciones entre los diversos bloques de función, se pueden seleccionar ajustes predefinidos de variables para cada bloque de función. Los parámetros incluidos en estos ajustes predefinidos de los bloques de función se especifican en las vistas de la tabla de acceso a parámetros. Si los parámetros se añaden a un bloque, estos parámetros se añaden después de todos los parámetros estándares.

La tabla de acceso a parámetros proporciona lo siguiente: (1) el orden en el que aparecen secuencialmente los parámetros en el diccionario de objetos en relación con la ubicación del objeto del bloque asociado; (2) una lista de parámetros asociada con la dirección de bloque de función en la tabla; y (3) ajustes de los parámetros predefinidos. Los ajustes de parámetros predefinidos incluyen de la visualización 1 a la visualización 4.

La visualización 1 es el ajuste de parámetros de operación dinámica. El ajuste de parámetros de operación dinámica incluye información requerida por el operario para observar en control del proceso, observar las condiciones de alarma y ajustar las dianas de operación.

La visualización 2 es el ajuste de parámetro de operación estática. El ajuste de parámetro de operación estática incluye la información que puede requerirse para la visualización por un operario con información dinámica. Esta información se lee una vez que primero se captura la visualización asociada, y se actualiza si cambia el código de actualización estático.

La visualización 3 es el ajuste de todos los parámetros dinámicos. El ajuste de todos los parámetros dinámicos incluye información que está cambiando en valor y que puede necesitar referenciarse en una visualización detallada.

La visualización 4 es el ajuste de parámetros estático. El ajuste de parámetros estático incluye información que se referencia normalmente durante la configuración o mantenimiento y que tiene un valor específico a menos que se haya cambiado por un operario o por un técnico instrumentista.

Los parámetros asociados con cada bloque se mencionan por separado en tablas de acceso. Los primeros seis índices son idénticos, formando un encabezado estándar para todos los bloques de función estándares y extendidos. Los índices restantes son para los parámetros de núcleo de la función y para los parámetros menos utilizados. Finalmente, están los parámetros requeridos para el procesamiento de alarma, seguido por los registros de alarma.

Otras Subfunciones Comunes

Además de las funciones comunes descritas anteriormente, existen muchas otras. Estas otras subfunciones pueden incluir, por ejemplo: estado; cálculo retroactivo (BKCAL), cascada (CAS); seguimiento de salida (TRK); desviación o proporción de equilibrio (BIAS o SP); manipulación segura de fallos (FSAFE); manipulación de estado de cascada deficiente; valores inválidos; parámetros; alarmas e inicialización y reinicio.

Componentes de Recursos

Como se ha mencionado anteriormente, un dispositivo incluye normalmente una o más aplicaciones de bloque de función 440. Una aplicación de bloque de función 440 incluye uno o más recursos 500/500', y un recurso 500/500' incluye uno o más bloques/objetos. Cada recurso tiene un bloque de recurso.

Adicionalmente, cada bloque de recurso contiene normalmente datos que se especifican para el soporte físico que está asociado con el recurso. El dato en el bloque de recurso se modela según los parámetros contenidos, por lo que no existen enlaces a este bloque.

Cada aplicación de bloque de función también contiene normalmente al menos un bloque de función. En general, existen diez bloques de función que, cuando se combinan, pueden proporcionar la vasta mayoría de funciones para el equipo de fabricación en un sistema de control de procesos. Estos bloques son: entrada analógica; salida analógica; sesgo; selector de control; entrada discreta; salida discreta; cargador manual, proporcional/derivativo; proporcional/integral/derivativo; y proporción. En estos dispositivos SIS, algunos pero generalmente no todos estos tipos bloques de función pueden o no soportarse. Por ejemplo, los SISFB disponibles pueden limitarse a entrada analógica, a salida analógica, entrada discreta y salida discreta. Del mismo modo, la asignación analógica, asignación digital y otros bloques se pueden soportar. Por tanto, debe apreciarse que alguno, todos, o ninguno u otros de los bloques de función identificados anteriormente pueden usarse y que algunos de estos bloques de función pueden incluir los SISFB o los no SISFB.

Además, diecinueve bloques de función estándares se utilizan normalmente para realizar funciones más complejas, incluyendo bloques de función avanzados, bloques de cálculo y bloques auxiliares. Estos diecinueve bloques de función incluyen normalmente: entrada de impulso; salida análoga compleja; salida discreta compleja; salida por etapa

proporcional/integral/ derivativa; control de dispositivo; generador de rampa de punto de referencia; separador; selector de entrada; caracterizador de señal, avance retraso; tiempo muerto; aritmética; cálculo; integrador; temporizador, alarma análoga; alarma discreta; interfaz humano análogo e interfaz de humano discreta. Estos bloques abarcan requisitos adicionales tanto para un bus de campo de baja velocidad como para uno de alta velocidad.

Los bloques transductores estándares se pueden utilizar también según se desee y/o se ha necesario.

Ejemplos de dos aplicaciones diana, un control manual 760 y una función relacionada con seguridad 770, que utiliza los bloques de función que se muestran en la Figura 11. El control manual 760 consiste en un bloque de función de entrada análoga 762, un cargador manual 764 y un bloque de función de salida análoga 768. La función relacionada con seguridad 770 consiste en una pluralidad de las entradas análogas SIS ("SISAI") 772, un votante análogo SIS ("SISAVTR") 774 y una salida digital SIS ("SISDO") 778.

Descripciones de Dispositivos (DD)

Como se muestra en la Figura 12, los procesos de aplicación de bloque de función pueden almacenar también descripciones de dispositivos (DD) 860. Para extender la interoperabilidad de la red, las descripciones de dispositivos 860 se utilizan además de los parámetros de bloques de función estándares. La descripción de dispositivos 860 se extiende a las descripciones de cada objeto en el dispositivo de campo virtual.

Las descripciones de dispositivos 860 proporcionan la información necesaria para que un sistema de control interprete el significado de los datos en el dispositivo de campo virtual, incluyendo las funciones de interfaz humana, tales como la calibración y diagnósticos.

La descripción de dispositivo se puede escribir en cualquier lenguaje de programación estándar, tal como C, C++ o SmallTalk.

GESTIÓN DEL SISTEMA

Durante el funcionamiento, los bloques de función ejecutan, en intervalos definidos de forma precisa y en una secuencia apropiada, la correcta operación del sistema de control. La gestión de sistema sincroniza la ejecución de los bloques de función y una comunicación de los parámetros del bloque de función en el bus. La gestión del sistema manipula también otros elementos importantes tales como la publicación de la hora del día a todos los dispositivos, la asignación automática de direcciones de dispositivos y la búsqueda de nombres o etiquetas de parámetros en el bus de campo.

La información de configuración necesaria para la gestión del sistema, tal como la programación del sistema, se describe mediante las descripciones de objetos. La información de configuración se almacena en el dispositivo de campo virtual de gestión de red y de sistema 310, como se muestra en la Figura 7A. El dispositivo de campo virtual de gestión de red y de sistema 310 proporciona acceso a la base de información de gestión de sistema (SIMB) 330 y también a la base de información de gestión de red (NMIB) 320. Las programaciones del sistema se pueden introducir manualmente o construirse utilizando una herramienta de construcción de programación. Una herramienta de construcción de programación se utiliza para generar un bloque de función y programaciones del programador activo de enlace se utilizan para generar las programaciones del sistema y de la red.

En base a la programación del sistema, se controla la gestión del sistema cuando se ejecutan los bloques de función. Esto asegura que cada bloque de función se ejecute en un momento apropiado en relación con otros bloques de función del sistema. Para un sistema de control verdaderamente distribuido, las actividades de los dispositivos y sus bloques de función tienen también que sincronizarse con aquellos de los otros dispositivos de la red. La coordinación de la ejecución de los bloques en diferentes dispositivos se controla mediante un gestor de red que utiliza una programación de red. La Gestión de Red se describe con mayor detalle más adelante.

Las programaciones de sistema y de red contienen el tiempo de inicio compensado desde el comienzo del tiempo de inicio de programación de enlace absoluto. El tiempo de inicio de programación de enlace absoluto se conoce por todos los dispositivos en el bus de campo.

La gestión de sistema tiene también un editor de tiempo que, en un programador activo de enlace 100, envía periódicamente la sincronización de la aplicación de red a todos los dispositivos de campo. El tiempo de programación de enlace de datos se muestrea y se envía con el mensaje de reloj de aplicación de manera que los dispositivos de recepción pueden ajustar su tiempo de aplicación local. Entre mensajes de sincronización, el tiempo de reloj de aplicación o de sistema se mantiene independientemente en cada dispositivo de campo en base a su propio reloj de sistema. El reloj de sistema en cada dispositivo de campo inicia la ejecución de la programación de sistema para ese dispositivo, no para el reloj de enlace de datos, a menos que el dispositivo de campo sea el programador activo de enlace 100. La sincronización de reloj de sistema permite que los dispositivos de campo representen datos a través de toda una red. Si existen editores de reloj del sistema de refuerzo en el bus, un editor de refuerzo se convertirá en el reloj de enlace de datos si el editor de tiempo activo actual fallara.

La gestión de sistema también asigna automáticamente direcciones de red únicas para cada dispositivo de campo. Normalmente, cada dispositivo de campo, excepto una memoria temporal, tendría una única red de dirección y una etiqueta física. A los dispositivos temporales no se les asignan normalmente etiquetas o direcciones permanentes. Los dispositivos temporales simplemente unen la red en una de las cuatro de direcciones de visitante de enlace de datos reservadas para ellos en la especificación de protocolo de capa de enlace de datos.

La función de gestión de sistema responsable de etiquetar y asignar la dirección de enlace de datos se refiere como el maestro de configuración. Este se co-ubica normalmente con el programador activo de enlace 100, aunque no se requiere, por lo que puede monitorear la lista viva para la adición de nuevos dispositivos. Cuando se añade un dispositivo en una dirección de red predeterminada, el maestro de configuración verifica que un núcleo de gestión de sistema para el dispositivo de campo no tiene una etiqueta física y le asigna una utilizando el protocolo de núcleo de gestión de sistema 810. Una vez asignada, el núcleo de gestión de sistema se mueve al estado inicializado. En este estado, está listo para asignarse a una dirección de red en la red operativa. Un núcleo de gestión de sistema se describe con más detalle más adelante.

La secuencia para asignar una dirección de red a un nuevo dispositivo de campo es como sigue: (1) se asigna una etiqueta física a un nuevo dispositivo por medio del dispositivo de configuración; (2) la gestión del sistema pregunta al dispositivo de campo por su dirección de red predeterminada de etiqueta de dispositivo físico; (3) la gestión de sistema utiliza la etiqueta de dispositivo físico para consultar la nueva dirección de red en la tabla de configuración; y (4) la gestión de sistema envía un mensaje de dirección compilado especial al dispositivo que fuerza al dispositivo a asumir la dirección de red. La secuencia de estas etapas se repite para todos los dispositivos que entran a la red en una dirección predeterminada.

La Figura 12 muestra una relación representativa entre cierta gestión de sistema y otros componentes de comunicación y de aplicación. En particular, la Figura 12 muestra las relaciones entre la gestión del sistema y la aplicación de bloques de función 440, los objetos de bloque de función 850, las descripciones de dispositivos (DD) 860 y las descripciones de objetos (OD) 280. La gestión de sistema puede utilizar la especificación de mensaje de bus de campo 230 para acceder remotamente a la información de gestión dentro de un dispositivo de campo. La gestión de sistema puede acceder también a la pila de comunicaciones 205 para realizar sus otras funciones.

Una sola entidad de gestión de sistema existe normalmente en cada maestro de enlace 105/105' o en el programador activo de enlace 100. Esta entidad comprende una base de información de gestión de sistema 830 (SMIB), un diccionario de objetos 280 y un núcleo de gestión de sistema 800.

El núcleo de gestión de sistema 800 proporciona un conjunto de funciones coordinadas y sincronizadas de red. Para promover la coordinación y sincronización de estos bloques a través de la red, se utiliza un modelo gestor/agente. En una realización preferida, el núcleo de gestión de sistema 800 asume el papel de un agente y responde a las instrucciones recibidas por la gestión del sistema. Un protocolo de la gestión del sistema se utiliza normalmente para definir las comunicaciones entre los gestores y los agentes.

La información que se utiliza para controlar la operación de la gestión del sistema puede organizarse como objetos almacenados en el SMIB 830. Al SMIB 830 se le accede mediante la red a través del dispositivo de campo virtual de gestión de sistema y de red 310. El SMIB 830 contiene los parámetros de configuración y de operación para un dispositivo. Ejemplos de los objetos incluidos en el SMIB 830 son: identificación del dispositivo, etiqueta física del dispositivo, lista de dispositivos de campo virtuales, objeto de tiempo, objeto de programación, y estado de la configuración.

La gestión del sistema permite que se accedan a los objetos de SMIB utilizando los servicios de aplicación de la aplicación de mensaje de bus de campo, tales como lectura, escritura, etc. El acceso al SMIB permite que aplicaciones remotas obtengan la información de la gestión del dispositivo, bien sea antes o durante de la operación de la red. El dispositivo de campo virtual de gestión se comparte con el agente de la gestión de red 880 del dispositivo y también proporciona, por tanto, el acceso a los objetos de agente de la gestión de red.

GESTION DE RED

La Figura 12 muestra también la relación arquitectónica entre la gestión de red y los otros componentes de comunicación y de aplicación en un dispositivo. Cada dispositivo contiene un solo agente de gestión de red 880 y las entidades de gestión de capa (LME) 875 para sus protocolos (uno para cada capa). Cada red tiene al lo menos un gestor de red que coordina la gestión de red de todo el sistema. La gestión de red proporciona las capacidades de: cargar una lista virtual de relaciones de comunicaciones; configurar la pila de comunicación 205; cargar la programación de red; controlar el rendimiento; y controlar la detección de fallos.

El gestor de red es responsable de mantener la operación de la red de acuerdo con las políticas definidas para el mismo por el gestor del sistema. El gestor de la red hace cumplir las políticas de la gestión del sistema controlando el estado de la pila de comunicaciones 205 en cada dispositivo, y toma la acción cuando es necesario. El gestor de red realiza estas tareas procesando la información y los informes producidos por los agentes de la gestión de red 880, y recomendando a los agentes para realizar los servicios solicitados por medio de la especificación de mensaje del bus de campo 230.

El agente de la gestión de red 880 es responsable de proporcionarle al gestor de red una interfaz de de especificación de mensaje de bus de capo 230 para gestionar los objetos de la pila de comunicaciones 205. Interno al dispositivo, el agente de la gestión de red 880 correlaciona las solicitudes de servicios de la especificación de mensaje del bus de campo con los objetos que mantiene para la pila de comunicaciones 205 como un conjunto, los objetos mantenidos por las LME.

Las LME 875 proporcionan la capacidad de gestión de un protocolo de capa, tal como la capa física (PHY) 200, la capa de enlace de datos (DLL) 210, la subcapa de acceso al bus de campo (FAS) 220 o la especificación de mensaje de bus de campo (FMS) 230 (como se ha mostrado en la Figura 2). Las LME 875 proporcionan al agente de gestión de red 880 una interfaz local de los objetos gestionados del protocolo. Todos los accesos de red a las LME y sus objetos se proporcionan por el agente de gestión de red 880.

La NMIB 895 contiene las NMIB 320 en el dispositivo de campo virtual de gestión de sistema y de red (VFD) 310. La NMIB contiene también objetos utilizados para definir la gestión de configuración, el rendimiento de la gestión y la gestión de fallos. Los objetos se acceden mediante los gestores de red que utilizan servicios especificación de mensaje de bus de campo. Los objetos usados para los objetos de gestión de red se designan de forma similar a los bloques de función descritos anteriormente.

BLOQUES DE FUNCIÓN FLEXIBLES

Para simplificar y con propósitos ilustrativos, los bloques de función flexibles ("FFB") se describen haciendo referencia principalmente a las realizaciones ejemplares. Sin embargo, debe apreciarse que los FFB y los FFB relacionados con seguridad ("SIS-FFB") se pueden utilizar en otras implementaciones y diseños que utilizan otras arquitecturas de control distribuidas. Además, debe apreciarse que los principios descritos en la presente memoria como se han aplicado a los FFB y/o a los SIS-FFB pueden ser aplicables también a otras implementaciones orientadas a bloque, Arquitecturas de bus de campo y otros sistemas de control de procesos.

Marco del Bloque de Función

Con referencia a las Figuras 8A y 8B, el sistema abierto descrito anteriormente proporciona un marco para y una descripción detallada de bloque de función 530/530'. Con referencia a la Figura 11, el sistema abierto descrito anteriormente proporciona un marco para y una descripción de la interconexión de las entradas y salidas del bloque de función para proporcionar una solución de aplicación.

Con referencia a la Figura 12, descrita anteriormente existen descripciones de dispositivos (DD) 860 que se pueden utilizar para describir los parámetros de entrada y de salida de un bloque de función. La DD 860 proporciona la información necesaria para que un sistema de control interprete el significado de los datos del bloque de función, incluyendo las funciones de interfaz humana, tales como la calibración y el diagnóstico. Como se ha mencionado anteriormente, la descripción de dispositivos puede escribirse en cualquier lenguaje de programación estandarizado, tal como C, C++ o SmallTalk, o un lenguaje de descripción de dispositivos diseñado personalizado.

Bloque de Función Flexible - Entrada/Salida y Algoritmo/Programa Configurado por Usuario Final

Como se ha mostrado en la Figura 13, una implementación de un SIS-FFB 1350 puede incluir entrada o entradas SIS-FFB configurables por el usuario final 1351, salida o salidas SIS-FFB configurables por el usuario final 1352 y un algoritmo (programa) SIS-FFB configurable por el usuario final 1353. El usuario final 1300 crea el SIS-FFB 1350, que configura la entrada o las entradas 1351, la salida o las salidas 1352 y el algoritmo 1353 de acuerdo con las necesidades de una aplicación particular y de acuerdo con los requisitos de seguridad particulares. Como se ha descrito anteriormente para un SIS-FB el parámetro o parámetros de entrada SIS-FFB 1351 definen la entrada o las entradas que se definen por el SIS-FFB 1350 y el parámetro o los parámetros de salida SIS-FFB 1352 definen la salida o las salidas que se generan por el SIS-FFB 1350 después que la entrada o las entradas se procesan por un algoritmo 1353, como se ha especificado por el SIS-FFB 1350. La Herramienta de Configuración SIS-FFB 1301 crea una descripción de dispositivo (DD) SIS-FFB 1360 que coincide con el SIS-FFB 1350 configurado por el usuario final. La Herramienta de Configuración SIS-FFB 1301 crea preferiblemente el SIS-FFB 1350 generando archivos de datos y archivos de código que definen el SIS-FFB 1350 en base a la entrada o entradas 1351, salida o salidas 1352 y el algoritmo 1353 configurados por el usuario y generando una descripción de dispositivo coincidente. Como alternativa, el usuario final 1300 (o un programador) puede generar archivos de datos y de código que definen el SIS-FFB 1350 y la descripción de dispositivo coincidente.

El usuario final crea el SIS-FFB 1350 y un SIS-FFB DD coincidente 860 ejecutando la Herramienta de Configuración SIS-FFB 1301. La SIS-FFB 860 posibilita que las aplicaciones de interfaz humana tales como la interfaz de operario, sintonización, calibración y diagnóstico se usen con el SIS-FFB 1350.

Puesto que el SIS-FFB 1350 opera en un entorno de bloque de función, las entradas y salidas SIS-FFB configuradas por el usuario final pueden interconectarse para solucionar problemas de control de aplicación específicos complejos tales como el control discreto/híbrido/lote y de PLC. Cualquier combinación de los bloques (estandarizados y flexibles, SIS y no SIS) puede generalmente utilizarse para solucionar cualquier problema de aplicación particular. También es

aparente que la interconexión de bloques estandarizados y flexibles es aplicable a conexiones de alta velocidad, tales como HSE, y conexiones de velocidades más bajas. Como tales los FFB y/o los SIS-FFB son en general comunicaciones de protocolos y de configuración independientes y pueden operar en cualquier diversidad de canales de comunicación.

Con referencia a la Figura 14, se ilustran dos dispositivos de campo 620 en un bus 120' que controlan un proceso. Como se ha mostrado, existen dos aplicaciones, la Aplicación A y la Aplicación B, ejecutándose por dos dispositivos de campo 620. La primera aplicación, la Aplicación A, es una aplicación no distribuida ejecutada por el primero de los dispositivos de campo 620. La aplicación A se construye mediante una combinación de SISFB y SIS-FFB interconectados (por ejemplo, SIS-FFB 1350). La segunda aplicación, la Aplicación B es una aplicación distribuida ejecutada por ambos de los dispositivos de campo 620. La Aplicación B se construye también mediante una combinación de los SISFB y de los SIS-FFB interconectados (por ejemplo, SIS-FFB 1350). Como se ha ilustrado por la Figura 14, el SIS-FFB supera la limitación de la entrada/salida no configurable por el usuario final y los bloques de función estandarizados no configurables el usuario final. Las aplicaciones distribuidas y no distribuidas 1360 en los dispositivos de campo 620 en el bus 120 pueden construirse utilizando cualquier combinación de los SISFB y SIS-FFB 1350. Debe apreciarse que en ciertas realizaciones, reducciones importantes en la instalación del sistema de control de planta, costes de operación y de mantenimiento pueden conseguirse utilizando los FFB y/o los SIS-FFB.

La figura 15 es un diagrama de bloque que ilustra un ejemplo de una aplicación compleja construida usando una combinación de bloques de función estándares y de los FFB. La figura 15 es un ejemplo del control de matriz multivariate para una planta del tratamiento de gas implementado utilizando el FFB-MVMC 954. Los dispositivos/componentes de campo (por ejemplo, PI 1, TI 1, TI 2, TI-3, AI 1, AI 2, FIC 1, FIC 2, LIC 1) mostrados en la figura 15 incluyen preferiblemente bloques de función estándares. Debe apreciarse, que aunque no se muestran en la figura 15, los SISFB y/o los SIS-FFB puedan ser utilizados en una implementación de este tipo tanto como particulares necesidades relacionadas con seguridad puedan requerir.

PROTOCOLO RELACIONADO CON SEGURIDAD EXTENDIDO

Como se ha descrito anteriormente, un SISRP puede utilizarse para autenticar y asegurar que las comunicaciones entre los SISC no se hayan interrumpido. En una realización, el SISRP utiliza números de secuencia y CK para validar y autenticar los mensajes.

Comunicaciones Editor-Suscriptor

Cuando se han conseguido las comunicaciones editor-suscriptor, al menos una de las diversas realizaciones facilita las comunicaciones seguras mediante el proceso mostrado en la Figura 16. Sin embargo, antes de describir la realización ilustrada en detalle debe apreciarse que, en general, una CK se asocia con cada enlace entre los SISFB y/o y los SIS-FFB. Como se ha descrito anteriormente, estas CK se generan mediante el sistema de configuración y se almacenan deseablemente en el recurso como parte de los objetos de enlace editor-suscriptor. Cuando los datos tienen que publicarse a un suscriptor, las comunicaciones incluyen deseablemente los parámetros de salida (es decir, los datos) incluyendo información de valor y de estado, un número de secuencia (como se ha descrito anteriormente) y autenticadores (por ejemplo, un CRC-32). Una realización de un proceso para generar el autenticador se muestra en la Figura 16A.

Como se muestra en la Figura 16A, el proceso para generar un autenticador para una realización que incluye obtener la información utilizada para generar el autenticador (Operación 1602). Esta operación incluye identificar el editor y el suscriptor con el fin de especificar la conexión a través de la cual los datos tienen que comunicarse (Operación 1604). Esta operación implica también obtener los datos (Operación 1606), obtener el próximo número de secuencia utilizado a través de la conexión identificada (Operación 1608), obtener la CK asociada con la conexión específica (Operación 1610), y obtener el índice de objeto utilizado para identificar el parámetro en un FBAP al que pertenecen los datos que tienen que comunicarse (Operación 1612). En una realización, la identificación de conexión incluye cuatro bytes (4) de información, el número de secuencia incluye dos bytes (2), el índice de objeto incluye dos bytes (2) y los datos incluyen cualquiera de dos (2) a ciento veinte (120) bytes de información. Más aún, debe apreciarse que en otras realizaciones, otras longitudes de datos y/o de información se pueden utilizar.

Una vez que se obtiene y se almacena adecuadamente la información deseada y necesaria (por ejemplo, en una RAM u otro) para usarse por el procesador del dispositivo de publicación, el proceso continúa con la disposición de la información obtenida en una secuencia deseada utilizada para generar una Unidad de Datos de Protocolo Virtual ("VPDU"), que puede usarse posteriormente para generar el autenticador. Como se muestra en la Operación 1614, se muestra una secuencia que puede utilizarse para generar una VPDU. Debe de apreciarse que se pueden utilizar otras secuencias, como implementaciones particulares requeridas o específicas. Normalmente, sin embargo, la secuencia utilizada para disponer la información y generar la VPDU debería estandarizarse de manera que cualquier SISC (por ejemplo un SISFB o SIS-FFB) pueda autenticar los datos recibidos de y proporcionados a cualquier otro SISFB/SIS-FFB. Por tanto, la secuencia mostrada en la Figura 16 es una secuencia de VPDU preferida, pero no obligatoria. Debe apreciarse que esta secuencia es "virtual" debido a que no se comunica sobre el canal negro con el dispositivo/componente de suscripción.

Usando la secuencia de la información generada en la Operación 1614, el proceso continúa con la generación del autenticador (“GA”) (Operación 1616). Debe apreciarse que cualquiera de muchos otros procesos de generación de autenticadores bastante conocidos puede utilizarse. De forma deseable, un autenticador escogido cumple con los requisitos de seguridad SIL-3 y/o SIL-2. En una realización, los algoritmos CRC-32 se utilizan para generar un autenticador CRC-32. En otras realizaciones, el CRC-64 y otros algoritmos se pueden utilizar para generar el autenticador.

Con referencia ahora a la Figura 16B, el proceso mostrado en la Figura 16A continúa con la generación de un ensamblaje de la PDU Real (“APDU”), es decir, los datos y la información que tienen que comunicarse sobre el canal negro desde el editor hasta el suscriptor (Operación 1618). Como se muestra en la Figura 16B, la APDU se ensambla en la siguiente secuencia: los datos 1616, el número de secuencia 1608 y el autenticador (GA) 1620 (es decir, el resultado generado en la Operación 1616). Debe apreciarse, que otras secuencias se pueden utilizar en otras realizaciones. Sin embargo, con propósitos de comunicaciones estandarizadas en una Arquitectura de bus de campo abierto o arquitectura similar, es deseable que todas las APDU utilicen las mismas secuencias de datos. La APDU se comunica después a través del canal negro hasta el suscriptor (Operación 1622). Tras la recepción de la APDU, el suscriptor extrae y almacena adecuadamente el autenticador recibido (“RA”) 1620’. Debe apreciarse que el GA 1620, así como los datos 1606 y/o el número de secuencia 1608 pueden interrumpirse durante la transmisión a través del canal negro. El suscriptor procede después con la autenticación de la unidad de datos de protocolo recibida o real (“APDU”) y determina si los datos en la APDU se han interrumpido o son, de lo contrario, erróneos debido a cualquiera de las consideraciones de seguridad mencionadas anteriormente (por ejemplo, retransmisiones, omisiones, falsificación de bits, enmascaramiento y similares). La autenticación de la APDU procede en el suscriptor generando una unidad de datos de protocolo esperada (“EPDU”) (Operación 1623). Como se ha mostrado, para generar la EPDU, el suscriptor dispone información esperada y los datos y la información recibida en la misma secuencia que se usó en la operación 1614 para generar la VPDU. Más específicamente, el suscriptor obtiene, de su memoria o de otro, una identificación de conexión esperada 1624 y un índice de objeto esperado 1626. Esto se combina, como se ha mostrado, con el número de secuencia recibido (“RSN”) 1608’ y los datos recibidos 1606’. En la Operación 1628, el suscriptor calcula después el valor autenticador esperado (“EA") usando, preferiblemente, los mismos algoritmos (por ejemplo, CRC-32) que se han utilizado por el editor en la Operación 1616 para generar el GA (1620).

Con referencia ahora a la Figura 16C, durante la Operación 1630, el EA se compara después con el RA.

Si el EA no es el mismo que el RA, entonces los datos se han distorsionado de alguna forma durante la transmisión a través del canal negro, se han enviado erróneamente por el editor, o se han procesado erróneamente por el suscriptor. La APDU se rechaza por el suscriptor y se detiene el procesamiento de la APDU (Operación 1632). Además, si un número sucesivo de las verificaciones de autenticador falla más de un número de veces de un umbral máximo, entonces el enlace entre el editor y el suscriptor se identifica deseablemente como “deficiente” y las PDU futuras no se procesan utilizando el enlace editor-suscriptor “deficiente” hasta que se resuelve la condición “deficiente”.

Con referencia nuevamente a la Operación 1630, si el EA es el mismo que el RA, entonces los datos y la información comunicados en la APDU se consideran autenticadas. Sin embargo, puesto que es posible que los datos que se envíen a un editor, sin la distorsión a través del canal negro, en una secuencia incorrecta, de forma deseable, el proceso continúa con el suscriptor verificando que los datos recibidos del editor estén en una secuencia esperada y correcta. Esta verificación de secuencia puede conseguirse, por ejemplo, mediante el suscriptor que obtiene el número de secuencia esperado de su VCR (Operación 1634) y comparando el número de secuencia recibido (“RSN”) con el número de secuencia esperado (“RSN”) (Operación 1636). Si el RSN no es igual al ESN, entonces se detiene el procesamiento del APDU (Operación 1632) y los datos se descartan. Si los números de secuencia son los mismos, entonces el procesamiento mensaje/datos continúa utilizando el mensaje no SIS de rutina que procesa rutinas y procedimientos. Adicionalmente, el suscriptor establece un parámetro, el último número de secuencia recibido (“LRSN”) variable, igual al RSN y deseablemente restablece un parámetro del “stalecount” a un valor de cero.

Como se ha mencionado anteriormente, si los números de secuencia son incorrectos un número dado de veces (es decir, un “stalecount” excede un umbral preestablecido), entonces el estado de la conexión entre el editor y el suscriptor asociados se ajusta para ser “deficiente” y no se aceptarán futuras PDU entre el editor y el suscriptor hasta que se pre-establezca el estado de conexión. Para recuperar o restaurar una conexión “deficiente” entre un editor y un suscriptor debido a un excedente del umbral de “stalecount”, para una realización, el proceso continúa, normalmente después de una pausa y de un reajuste manual o automático del enlace, cuando se recibe un autenticador válido. Puesto que el número de secuencia enviado por el editor, el RSN y el LRSN en un suscriptor son la mayoría de las veces diferentes, se calcula la diferencia entre el RSN y el LRSN. Si esta diferencia excede el umbral para el stalecount, entonces se ajusta el LRSN con respecto al RNS, el stalecount se ajusta para ser cero (0) y se descarta la PDU. La próxima PDU recibida se procesa nuevamente de acuerdo con los procedimientos descritos en las Figuras 16A - C y las reanudaciones de los procesamientos de datos normales proporcionados al autenticador y los números de secuencia son menos que el umbral del stalecount.

Aunque las operaciones mostradas en las Figuras 16A -16C exponen una realización para autenticar los datos comunicados a través de un canal negro cumplen ciertos requisitos de seguridad, debe apreciarse que las operaciones, secuencias de datos, algoritmos de autenticación y similares pueden cambiarse, añadirse y/o suprimirse. Por ejemplo,

las operaciones 1630-1636 pueden suprimirse del procesamiento. Tales operaciones pueden considerarse como opcionales si el RSN utilizado para generar la EPDU en la Operación 1623 se reemplaza con un ESN. Utilizando un flujo de procesos de este tipo, se anticipa que el EA no sería igual al RA si cualquiera de los bits en la información recibida se interrumpiera o de lo contrario fuese erróneo. Sin embargo, un flujo de proceso de este tipo puede no ser capaz de identificar más precisamente qué tipo de errores están ocurriendo a través del canal negro, por ejemplo, si el error es debido a problemas de número de secuencia, interrupción de datos reales recibidos, problemas de la identificación de conexión u otros. Como tal, debe apreciarse que diversos flujos de procesos, algoritmos y operaciones, PDU y similares se pueden utilizar para autenticar los datos comunicados entre un editor y un suscriptor a través de un canal negro.

Las Figuras 16D, 16E y 16F muestran el diagrama de flujo de otra realización que puede utilizarse para asegurar las comunicaciones editor-suscriptor. Esta realización utiliza datos replicados para proporcionar seguridad adicional de la integridad de datos. Con referencia a la Figura 16D, las operaciones 1650 - 1666 son sustancialmente las mismas que aquellas de las Operaciones 1600 - 1616 (mostradas en la Figura 16A) y proceden como se han descrito anteriormente. Una vez que el proceso de generación de autenticación (Operación 1666) se completa, el proceso continúa con la generación o ensamblaje de una PDU Real Extendida (“EAPDU”) que tiene que comunicarse a través del canal negro desde el editor hasta el suscriptor (Operación 1668). Como se muestra en la Figura 16E, la EAPDU se ensambla en la siguiente secuencia: los datos 1656, el número de secuencia 1658, el autenticador 1670, los datos 1656, el número de secuencia 1658 y el autenticador 1670. Es decir, la secuencia de la EAPDU contiene datos replicados: RD1 1692 que consiste en los datos 1656, el número de secuencia 1658 y el autenticador 1670 y el RD2 1694 que consiste en los datos 1656, el número de secuencia 1658 y el autenticador 1670. La EAPDU se comunica después a través del canal negro hasta el suscriptor (Operación 1672, como se ha descrito anteriormente).

Después de la recepción de la EAPDU, el suscriptor procede con la autenticación de la EAPDU y determina si los datos en la EAPDU se han distorsionado o de lo contrario son erróneos debido a cualquiera de las consideraciones de seguridad mencionadas anteriormente. Como se ha mostrado en la Figura 16, la autenticación de la EAPDU procede en el suscriptor comparando, en una base bit por bit, el RD1 recibido 1692 con el RD2 recibido 1694 (Operación 1696). Si el RD1 no es idéntico al RD2, entonces los datos son inválidos, la EAPDU se rechaza por el suscriptor, el procesamiento de la EAPDU se detiene (Operación 1682) y se incrementa el stalecount. De lo contrario, la autenticación de la EAPDU continúa con el suscriptor generando una unidad de datos de protocolo extendida esperada (“EEPDU”) (Operación 1673). La EEPDU se genera mediante el suscriptor que obtiene una identificación de conexión esperada 1674 y un índice de objeto esperado 1676 como se ha descrito anteriormente. Esto se combina, como se ha mostrado, con el número de secuencia recibido 1658’ y los datos recibidos 1656’ (que pueden obtenerse de cualquiera del RD1 o del RD2). El suscriptor calcula después el valor de autenticador esperado (“EA”) (Operación 1678) como se ha descrito anteriormente. El EA se compara después con el autenticador recibido (“RA”) que se ha extraído adecuadamente de cualquiera de RD1 o RD2. Si el EA no es el mismo que el RA, los datos son inválidos. La EAPDU se rechaza por el suscriptor, se detiene el procesamiento de la EAPDU (Operación 1682) y se incrementa el stalecount.

Con referencia nuevamente a la Operación 1680, si el EA es el mismo que el RA, entonces continúa el proceso de autenticación con el suscriptor que verifica que los datos recibidos del editor estén en la secuencia esperada y correcta comparando el número de secuencia recibido (“RSN”) con el número de secuencia esperado (“ESN”) obtenido de su VCR (Operación 1684). Si el RSN no es igual al ESN, entonces el procesamiento de la EAPDU se detiene (Operación 1682), los datos se descartan y se incrementa el stalecount. Si los números de secuencia son los mismos, entonces el procesamiento de mensaje/datos continúa utilizando las rutinas y procedimientos de procesamiento de mensaje no SIS rutinario. Adicionalmente, el suscriptor ajusta el LRSN para ser igual al RSN y reajusta deseablemente el stalecount a un valor de cero.

Si los datos 1656, el autenticador 1670 o los números de secuencia son incorrectos un número dado de veces (es decir, que el stalecount excede un umbral predeterminado), entonces el estado de la conexión entre el editor y el suscriptor asociados se establece como “deficiente” y no se aceptarán PDU futuras entre el editor y el suscriptor hasta que el estado de conexión se restablezca como se ha descrito anteriormente.

Además, debe apreciarse que autenticando los datos comunicados a través del canal negro para usarse mediante los componentes relacionados con seguridad, las diversas realizaciones de la invención descritas en la presente memoria se pueden utilizar también para verificar el estado de operación de un canal negro para dispositivos no relacionados con seguridad. En particular, si errores repetidos ocurren a través de un canal negro entre los dispositivos relacionados con seguridad, la probabilidad de errores para los dispositivos no relacionados con seguridad se aumenta probablemente. Como tal, monitoreando la velocidad y el número de fallos de autenticación para las comunicaciones relacionadas con seguridad, uno puede monitorear indirectamente también el estado global del canal negro para todas las comunicaciones.

Comunicaciones Cliente-Servidor

Además de soportar las comunicaciones editor-suscriptor, se pueden soportar las comunicaciones cliente-servidor entre los SISC. Como se aprecia normalmente, las comunicaciones cliente-servidor implican normalmente operaciones de lectura y operaciones de escritura. Los procesos, sistemas y dispositivos para implementar los procesos de lectura y de escritura relacionados con seguridad se describen en lo sucesivo a continuación.

Lectura

En general, una lectura utilizada para leer datos utilizados en un dispositivo SIS (es decir, uno con los SISC) es similar, y en ciertas realizaciones idéntico, a las lecturas utilizadas en dispositivos no SIS (es decir, aquellos sin los SISC).

Como se muestra en la Figura 17A, el proceso por el cual un componente responde a una solicitud de lectura normalmente trae una respuesta del servidor (es decir, el componente que proporciona la información requerida al solicitante/cliente) de una solicitud para obtener o "leer" ciertos bloques (es decir, valores de salida) proporcionados por el dispositivo. Tales bloques se identifican normalmente mediante un número índice u otro identificador (Operación 1702). Como se aprecia normalmente, un dispositivo puede ser capaz de arrojar muchos bloques, algunos de los cuales pueden ser SISC. Como tal, el proceso continúa adecuadamente con la determinación de si la lectura implica un bloque relacionado con seguridad (Operación 1704).

Si la lectura no implica un SISC, entonces el procesamiento continúa deseablemente utilizando los no SISRP estándares (Operación 1706).

Con referencia nuevamente a la Operación 1704, si la lectura implica un SISC, por ejemplo, uno que incluye un parámetro de seguridad tal como SIS_Access o similares, entonces el procesamiento continúa utilizando el SISRP. Como se ha descrito anteriormente con respecto a las comunicaciones editor-suscriptor, el SISRP, en al menos una realización, proporciona el cálculo y la transmisión de un autenticador a través de un canal negro. Las comunicaciones cliente-servidor, que implican los SISC, un autenticador se utiliza en la transmisión de la información deseada. Como se muestra en las Operaciones 1708-1744 (Figuras 17A - C), el proceso para generar un autenticador, transmitir los datos de lectura requeridos y autenticar los datos transmitidos es sustancialmente el mismo que aquel utilizado en la transmisión de datos para las comunicaciones editor-suscriptor. Sin embargo, un subíndice opcional y adicional 1717 puede incluirse en la información utilizada para generar y verificar el autenticador. Además, si el número de secuencia recibido no es igual al número de secuencia esperado entonces, para al menos una realización, se descarta la PDU.

Las Figuras 17D - 17F muestran el diagrama de flujo de otra realización del SISRP que proporciona la réplica de datos transmitidos a través de un canal negro en las comunicaciones cliente-servidor entre los SISC. Como se muestra en las Operaciones 1750-1799 (Figuras 17D - F) el proceso para generar un autenticador, replicar los datos, transmitir los datos de lectura requeridos y autenticar los datos transmitidos es sustancialmente el mismo que el utilizado en la transmisión de los datos extendidos para las comunicaciones editor-suscriptor.

Escritura

Cuando se desean procesos de escritura para los SISC, se utiliza de forma deseada el SISRP, como se ha descrito anteriormente. Para asegurar que el receptor de los datos y de la información "de escritura" verifica primero los datos y la información recibida, el SISRP incluye un autenticador y un número de secuencia en cualquiera de las comunicaciones. Este proceso puede conseguirse, por ejemplo, comparando la longitud de datos que tiene que escribirse (como se ha especificado, por ejemplo, en la FMS) frente a la longitud de la cadena de datos recibida en realidad. Si tal cadena de datos no tiene una longitud válida, la cadena de datos se descarta. Si la cadena de datos tiene una longitud válida, entonces se valida el autenticador, preferiblemente utilizando los procesos descritos anteriormente en la memoria con respecto a las comunicaciones editor-suscriptor. Además, una vez que se valida el autenticador, la operación de escritura procede después. Sin embargo, si el bloque de recurso asociado con el bloque que recibe la solicitud de escritura (es decir, el servidor) no está en un estado OSS o en un estado Manual, entonces deseablemente la solicitud de escritura se descarta. Tiene que apreciarse que en general es indeseable escribir parámetros utilizados por los SISC cuando se está utilizando un recurso relacionado con seguridad.

Adicionalmente, bajo ciertas condiciones, puede ser deseable escribir datos en un SISLO. Puesto que no se proporcionan los números de secuencia para los SISLO, la escritura a un SISLO procede adecuadamente con el cálculo de un autenticador sin incluir un número de secuencia. De lo contrario, el procesamiento de "escritura" para los SISLO procede como se ha descrito anteriormente en la presente memoria para los SISC.

ELEMENTOS DE SEGURIDAD ADICIONALES

Detección de Error de Autenticador

Además del uso del SISRP y otros elementos y funciones descritos anteriormente en la presente memoria, se puede proporcionar también la detección de error de autenticador. Por ejemplo, el número de errores de autenticador que se generan durante un periodo de tiempo pueden monitorearse. Para componentes utilizados en aplicaciones SIL-3, deseablemente un índice de error mayor que uno (1) durante cada 140 minutos da como resultado que el componente se configure en un estado seguro a fallos con respecto a sus bloques de salida. De forma similar, en una aplicación SIL-2, un umbral de un error (1) durante cada catorce (14) minutos da como resultado la configuración del componente en un estado seguro a fallos. Por último, para las aplicaciones SIL-1, el umbral de error es deseablemente uno (1) cada 1,4 minutos. Otros índices de error pueden utilizarse para otras implementaciones de seguridad, según se desee. Normalmente, la detección de error de autenticador no es necesaria si se utiliza el método de datos de réplica descrito en la presente memoria.

Bloque Transductor de Diagnósticos

Uno o más bloques transductores de diagnósticos pueden utilizarse en una Arquitectura de bus de campo de bloque de función o arquitectura similar. Sin embargo, normalmente solo un bloque transductor de diagnóstico existe por dispositivo SIS. El bloque transductor de diagnóstico incluye generalmente un temporizador y un conjunto de contadores de pistas que monitorean todas las comunicaciones hasta/desde el dispositivo (es decir, los VCR) para errores. Los contadores pueden incluir aquellos para los autenticadores deficientes, el tiempo en el que el último autenticador deficiente se ha recibido, el número de números de secuencias deficientes, el tiempo en el que el último número de secuencia deficiente se ha recibido y el tiempo desde que el último error se ha comunicado a través del canal negro.

Monitor de Integridad de Canal Negro

Un monitor de integridad de canal negro también puede utilizarse. Este monitor verifica que el índice de errores no detectados (es decir, errores que no se han detectado por los dispositivos de monitorización de canal negro sino que se detectan debido a números de secuencias deficientes o autenticadores inválidos) del canal negro no exceda un límite dado (es decir, predeterminado o ajustado en tiempo real). Si el número de errores excede el umbral, deseablemente, este monitor termina las conexiones del canal negro en las que surgen los errores. De forma deseable, tales conexiones se reanudan tras la intervención del operario, se reconectan y reinician. Las reconexiones o reinicios automatizados o semiautomatizad0s pueden conseguirse también.

Además, la implementación de la unidad de protocolo real extendida descrita anteriormente con referencia a las Figuras 16D - 16F y 17D - 17F puede utilizarse con o sin el monitor de integridad de canal negro.

Monitorización y Detección del Retraso en Cola

La monitorización del número de secuencia puede conseguirse. Por ejemplo, en una conexión editor-suscriptor, un mensaje puede publicarse con cualquier macro-ciclo. Para ayudar en la detección de los retrasos en cola, al inicio de la conexión editor-suscriptor, el editor comunica un número de secuencia al suscriptor para la conexión. Después, el suscriptor aumenta el número de secuencia localmente con cada macro-ciclo y lo compara con el número de secuencia recibido, si la diferencia excede la varianza tolerable máxima, entonces la seguridad de fallos u otras acciones apropiadas pueden activarse por el suscriptor u otro.

Monitorización y Detección de la Sincronización de Tiempo

La sincronización de tiempo a través del canal negro puede conseguirse también. Por ejemplo, las diversas realizaciones pueden incluir un Monitor de Sincronización de Tiempo de Capa SIS (“el Monitor de Tiempo”). Tal monitor incluye deseablemente un cristal con al menos 100 ppm de precisión y una resolución de al menos 100 micro segundos. Sin embargo, se pueden utilizar otras precisiones y resoluciones en otras realizaciones.

El Monitor de Tiempo se ejecuta después que el canal negro ha procesado una PDU de distribución de tiempo recibida. El Monitor de Tiempo determina si el sentido entre un tiempo de canal negro y un tiempo de la capa SIS es mayor que una cantidad predeterminada. En una realización, la tendencia permisible puede calcularse como sigue:

Tendencia Permisible = $(TLCTD_n - TLCTD_{n-1}) * 2 * 10^{-4}$

en la que $TLCTD_n$ es el tiempo del reloj de la capa SIS después del procesamiento de la distribución de tiempo actual (“TD”); y $TLCTD_{n-1}$ es el tiempo del reloj de la capa SIS después del procesamiento del TD anterior.

Además, la tendencia real puede calcularse como sigue:

Tendencia Real = $(TTD_n - TTD_{n-1}) - (TLCTD_n - TLCTD_{n-1})$

en la que TTD_n es el Tiempo de Enlace de Datos (“DL-Tiempo”) después del procesamiento de la TD real y el TTD_{n-1} es el DL-Tiempo después del procesamiento de la TD anterior.

Adicionalmente, el Monitor de Tiempo ajusta una bandera que indica la tendencia excesiva cuando la tendencia real es mayor que la tendencia permisible. Tal bandera puede establecerse en un parámetro de error de sincronización de canal trasero del Bloque de Recurso. Del mismo modo, si un TD válido no se recibe dentro de un periodo de tiempo dado, por ejemplo, dentro de seis (6) periodos de distribución de tiempo consecutivos, el bloque de recurso puede también establecer el parámetro de error de sincronización del canal negro. Cuando el parámetro de error de sincronización del canal negro se establece, el monitor de tiempo instruirá al canal negro para enviar periódicamente secuencias de tiempo obligado (“CT”) con el fin de resincronizar el tiempo una vez cada periodo TD. Cuando el parámetro de error de sincronización del canal negro se ajusta y ocurre un fallo en un bloque de recurso, la capa de enlace de datos envía un reajuste de manera que tanto errores de fallos como se sincronización de tiempo pueden clarificarse simultáneamente. Por tanto, tiene que apreciarse que cada capa SIS puede incluir un Monitor de Tiempo que monitoree si los recursos están en sincronización y si no, instruya a las capas de enlace de datos para tomar las acciones apropiadas.

Como se ha descrito anteriormente en la presente memoria, las diversas realizaciones proporcionan sistemas y procesos para comunicar datos entre los sistemas SIS proporcionando uno o más SISC utilizando una Arquitectura de bus de campo y Arquitecturas similares y sistemas. Aunque la presente invención se ha descrito con referencia a ciertas arquitecturas, bloques de función, procesos y estructuras de datos y similares debe apreciarse que la presente invención no está limitada y tiene que interpretarse.

REIVINDICACIONES

1. Un aparato que comprende: un procesador configurado para determinar si al menos un mensaje comunicado a través de un medio de transmisión no se ha puesto en cola controlando la sincronización del tiempo y comparando una tendencia real con una tendencia permisible, en el que la tendencia permisible se calcula multiplicando una diferencia de tiempo entre un primer reloj de canal negro determinado después del procesamiento de un mensaje de distribución de tiempo anterior y un segundo reloj de la capa de seguridad determinado después del procesamiento de un mensaje de distribución de tiempo actual por una constante; y una unidad de fijación de medio, que comunica los mensajes de distribución de tiempo y los mensajes de salida entre el procesador y un medio de transmisión.

2. El aparato de la reivindicación 1, que comprende además al menos un bloque de función relacionado con seguridad.

3. El aparato de la reivindicación 2, en el que al menos un bloque de función relacionado con seguridad recibe datos de entrada análogos o es un Bloque de Función del Sistema Instrumentado de Seguridad.

4. El aparato de la reivindicación 1, que comprende además una memoria adaptada para almacenar al menos un bloque de función no relacionado con seguridad.

5. El aparato de la reivindicación 1, que comprende además una memoria configurada para almacenar una pluralidad de bloques de función, que incluye un bloque de función no relacionado con seguridad y un bloque de función relacionado con seguridad.

6. El aparato de la reivindicación 5, en el que el bloque de función no relacionado con seguridad y el bloque de función relacionado con seguridad se interconectan para comunicar datos sólo desde el bloque de función relacionado con seguridad hasta el bloque de función no relacionado con seguridad.

7. El aparato de la reivindicación 1, en el que el medio de transmisión incluye un bus digital.

8. El aparato de la reivindicación 6, en el que el medio de transmisión incluye una red Ethernet de Alta Velocidad.

9. El aparato de la reivindicación 1, en el que el procesador se adapta para usarse con al menos uno de:

un bloque de recurso relacionado con seguridad;

un primer bloque transductor relacionado con seguridad; y

un segundo bloque transductor relacionado con seguridad.

10. El aparato de la reivindicación 9, en el que el bloque de recurso relacionado con seguridad aísla un bloque de función relacionado con seguridad del soporte físico.

11. El aparato de la reivindicación 9, en el que el primer bloque transductor relacionado con seguridad desconecta una entrada de un bloque de función relacionado con seguridad y un segundo transductor relacionado con seguridad desconecta una salida del bloque de función relacionado con seguridad.

12. El aparato de la reivindicación 1, en el que el mensaje de salida es al menos consistente con SIL-1 o al menos consistente con SIL-2 y/o al menos consistente con SIL-3.

13. El aparato de la reivindicación 1, en el que el procesador genera una unidad de datos de protocolo virtual antes de la comunicación del mensaje de salida.

14. El aparato de la reivindicación 1, en el que el medio de transmisión comprende además un canal negro o una Ethernet de alta velocidad.

15. El aparato de la reivindicación 1, que comprende además un bloque transductor de diagnóstico.

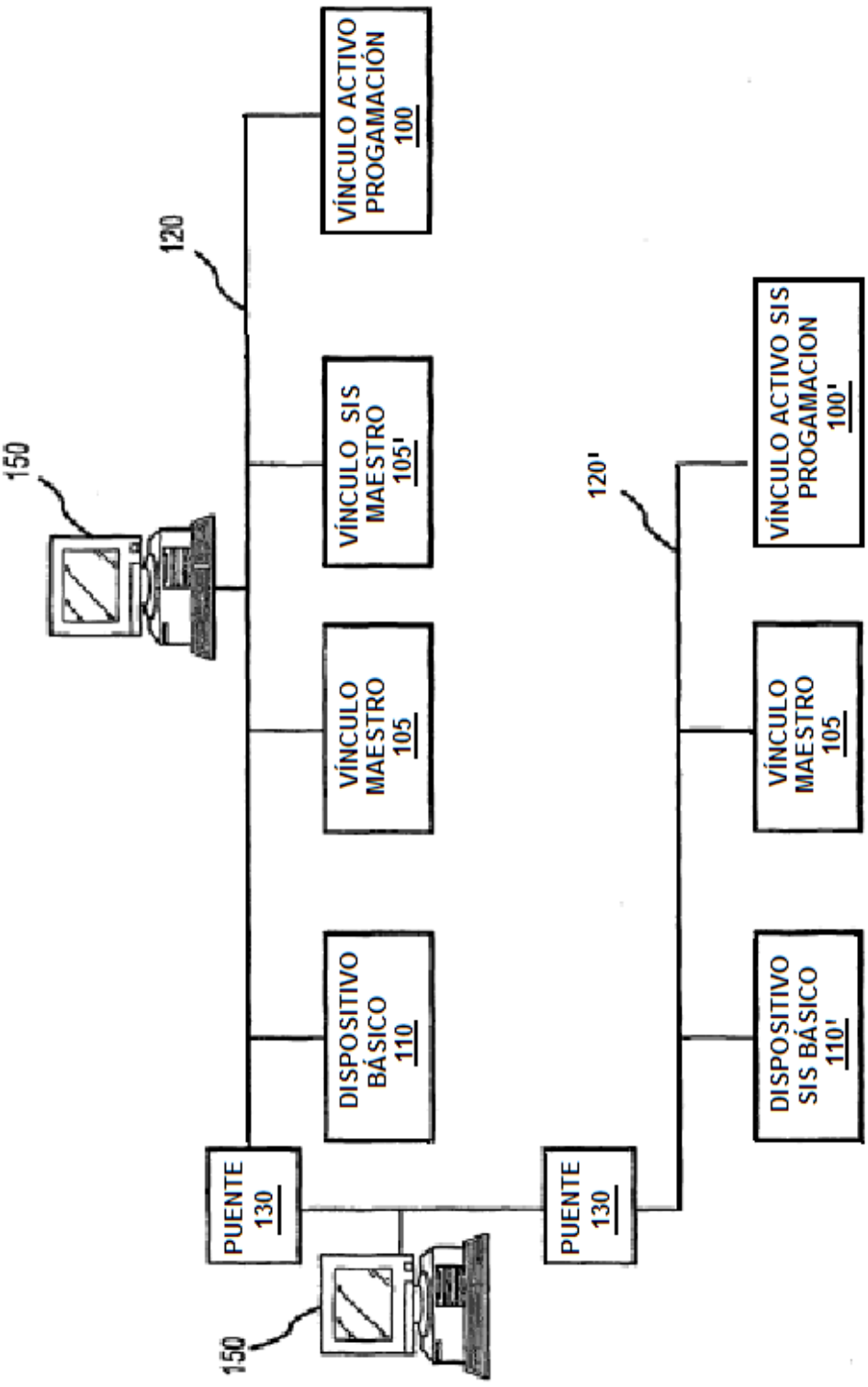

150
120
VÍNCULO ACTIVO PROGAMACIÓN 100
VÍNCULO SIS MAESTRO 105'
VÍNCULO MAESTRO 105
DISPOSITIVO BÁSICO 110
PUENTE 130
150
PUENTE 130
120'
VÍNCULO ACTIVO SIS PROGAMACION 100'
VÍNCULO MAESTRO 105
DISPOSITIVO SIS BÁSICO 110'

FIG.1

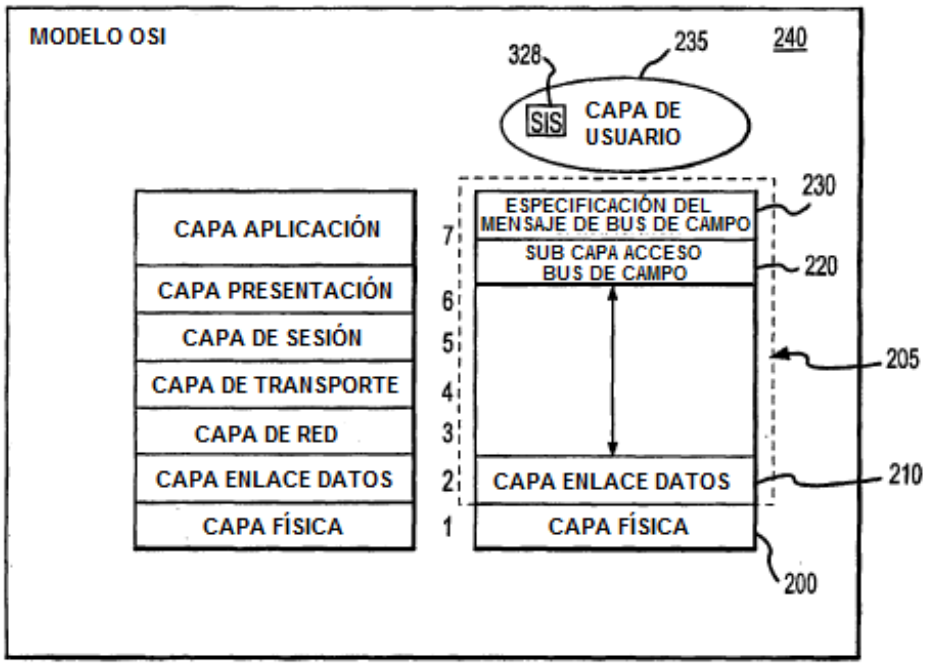

MODELO OSI
328
235
240
SIS
CAPA DE USUARIO
CAPA APLICACIÓN
CAPA PRESENTACIÓN
CAPA DE SESIÓN
CAPA DE TRANSPORTE
CAPA DE RED
CAPA ENLACE DATOS
CAPA FÍSICA
7
6
5
4
3
2
1
ESPECIFICACIÓN DEL MENSAJE DE BUS DE CAMPO
SUB CAPA ACCESO BUS DE CAMPO
CAPA ENLACE DATOS
CAPA FÍSICA
230
220
205
210
200

FIG.2

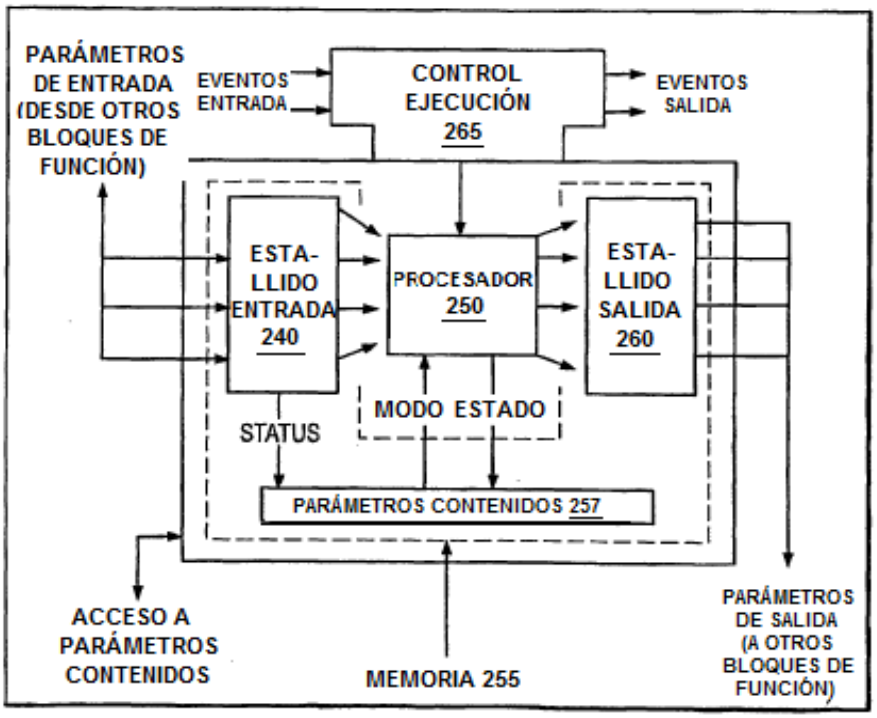

PARÁMETROS DE ENTRADA (DESDE OTROS BLOQUES DE FUNCIÓN)
EVENTOS ENTRADA
CONTROL EJECUCIÓN 265
EVENTOS SALIDA
ESTA-LLIDO ENTRADA 240
PROCESADOR 250
ESTA-LLIDO SALIDA 260
MODO ESTADO
STATUS
PARÁMETROS CONTENIDOS 257
ACCESO A PARÁMETROS CONTENIDOS
MEMORIA 255
PARÁMETROS DE SALIDA (A OTROS BLOQUES DE FUNCIÓN)

FIG.3

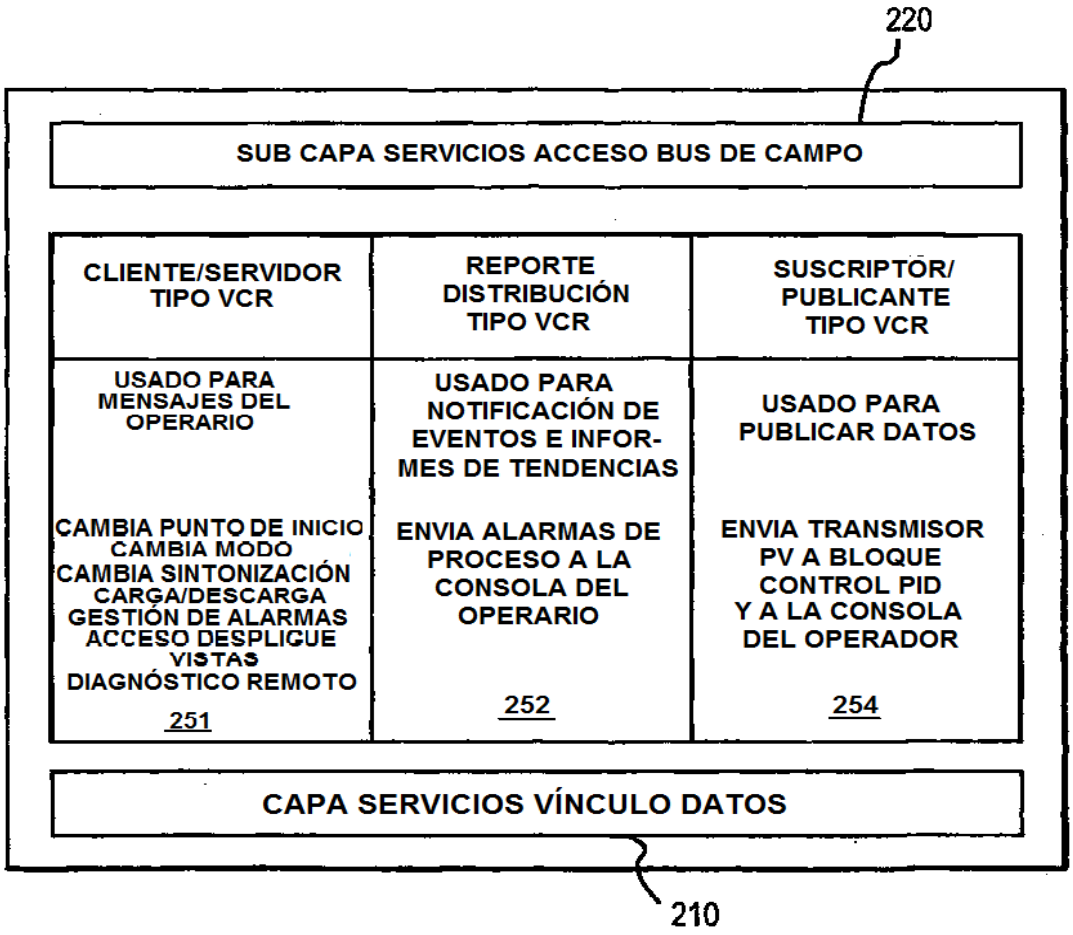

220
SUB CAPA SERVICIOS ACCESO BUS DE CAMPO
CLIENTE/SERVIDOR TIPO VCR
REPORTE DISTRIBUCIÓN TIPO VCR
SUSCRIPTOR/ PUBLICANTE TIPO VCR
USADO PARA MENSAJES DEL OPERARIO
USADO PARA NOTIFICACIÓN DE EVENTOS E INFOR-MES DE TENDENCIAS
USADO PARA PUBLICAR DATOS
CAMBIA PUNTO DE INICIO
CAMBIA MODO
CAMBIA SINTONIZACIÓN
CARGA/DESCARGA
GESTIÓN DE ALARMAS
ACCESO DESPLIGUE VISTAS
DIAGNÓSTICO REMOTO
ENVIA ALARMAS DE PROCESO A LA CONSOLA DEL OPERARIO
ENVIA TRANSMISOR PV A BLOQUE CONTROL PID Y A LA CONSOLA DEL OPERADOR
251
252
254
CAPA SERVICIOS VÍNCULO DATOS
210

FIG.4

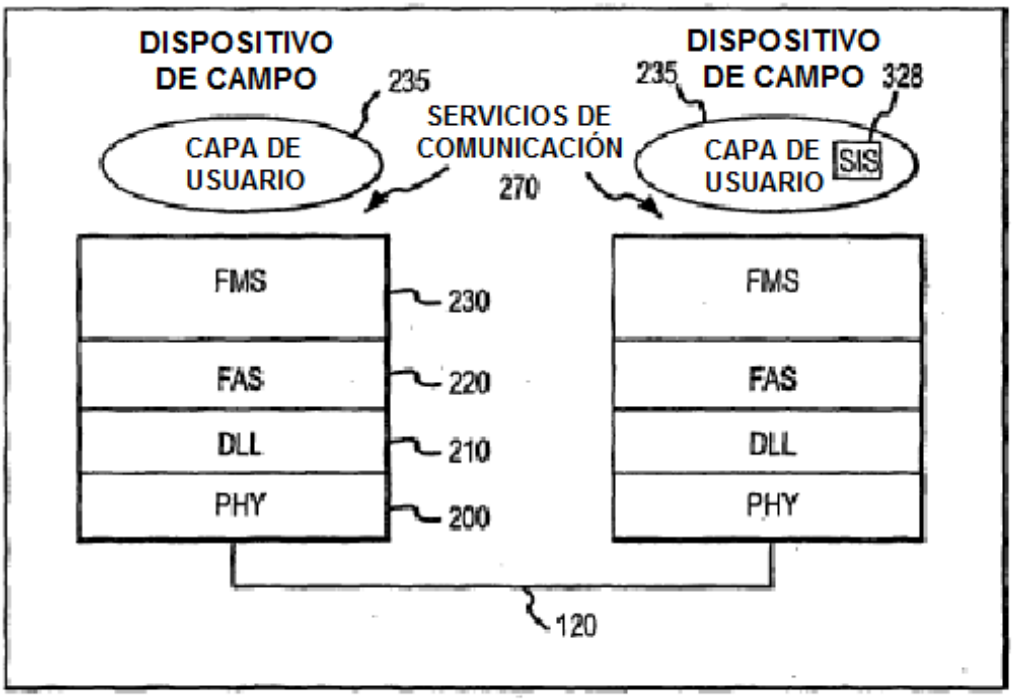

DISPOSITIVO DE CAMPO
235
CAPA DE USUARIO
SERVICIOS DE COMUNICACIÓN
270
DISPOSITIVO DE CAMPO
235
328
CAPA DE USUARIO
SIS
FMS
FAS
DLL
PHY
230
220
210
200
FMS
FAS
DLL
PHY
120

FIG.5

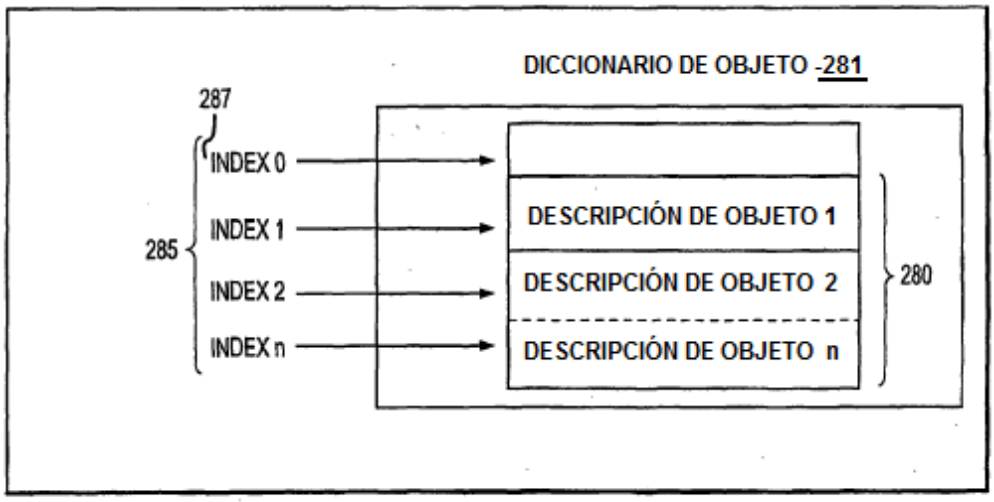

DICCIONARIO DE OBJETO -281
287
INDEX 0
INDEX 1
INDEX 2
INDEX n
285
DESCRIPCIÓN DE OBJETO 1
DESCRIPCIÓN DE OBJETO 2
DESCRIPCIÓN DE OBJETO n
280

FIG.6

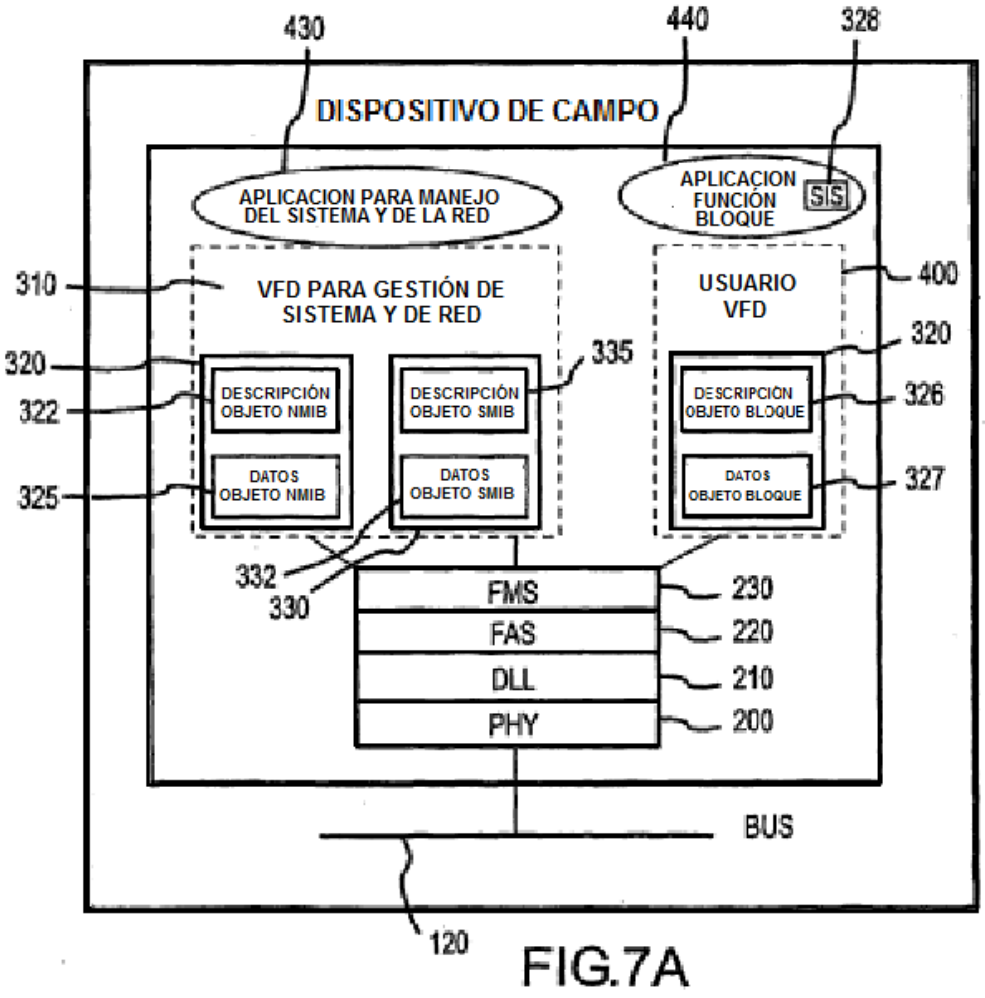

430
440
328
DISPOSITIVO DE CAMPO
APLICACION PARA MANEJO DEL SISTEMA Y DE LA RED
APLICACION FUNCIÓN BLOQUE
SIS
310
VFD PARA GESTIÓN DE SISTEMA Y DE RED
USUARIO VFD
400
320
322
325
DESCRIPCIÓN OBJETO NMIB
DATOS OBJETO NMIB
DESCRIPCIÓN OBJETO SMIB
DATOS OBJETO SMIB
335
DESCRIPCIÓN OBJETO BLOQUE
DATOS OBJETO BLOQUE
326
327
332
330
FMS
FAS
DLL
PHY
230
220
210
200
BUS
120

FIG.7A

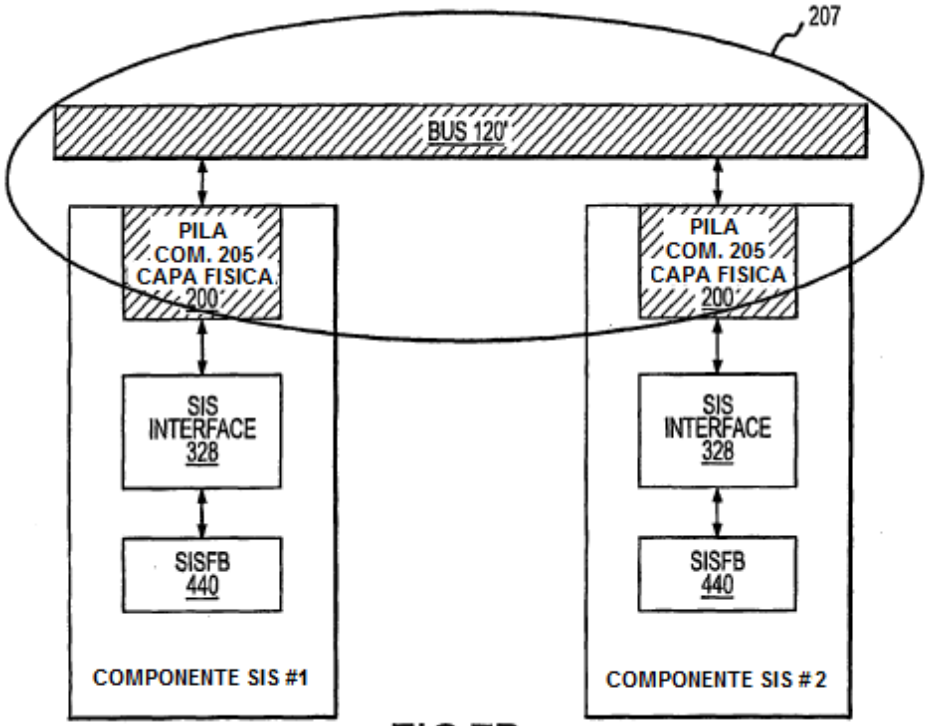

207
BUS 120'
PILA
COM. 205
CAPA FISICA
200'
SIS
INTERFACE
328
SISFB
440
COMPONENTE SIS #1
PILA
COM. 205
CAPA FISICA
200'
SIS
INTERFACE
328
SISFB
440
COMPONENTE SIS #2

FIG.7B

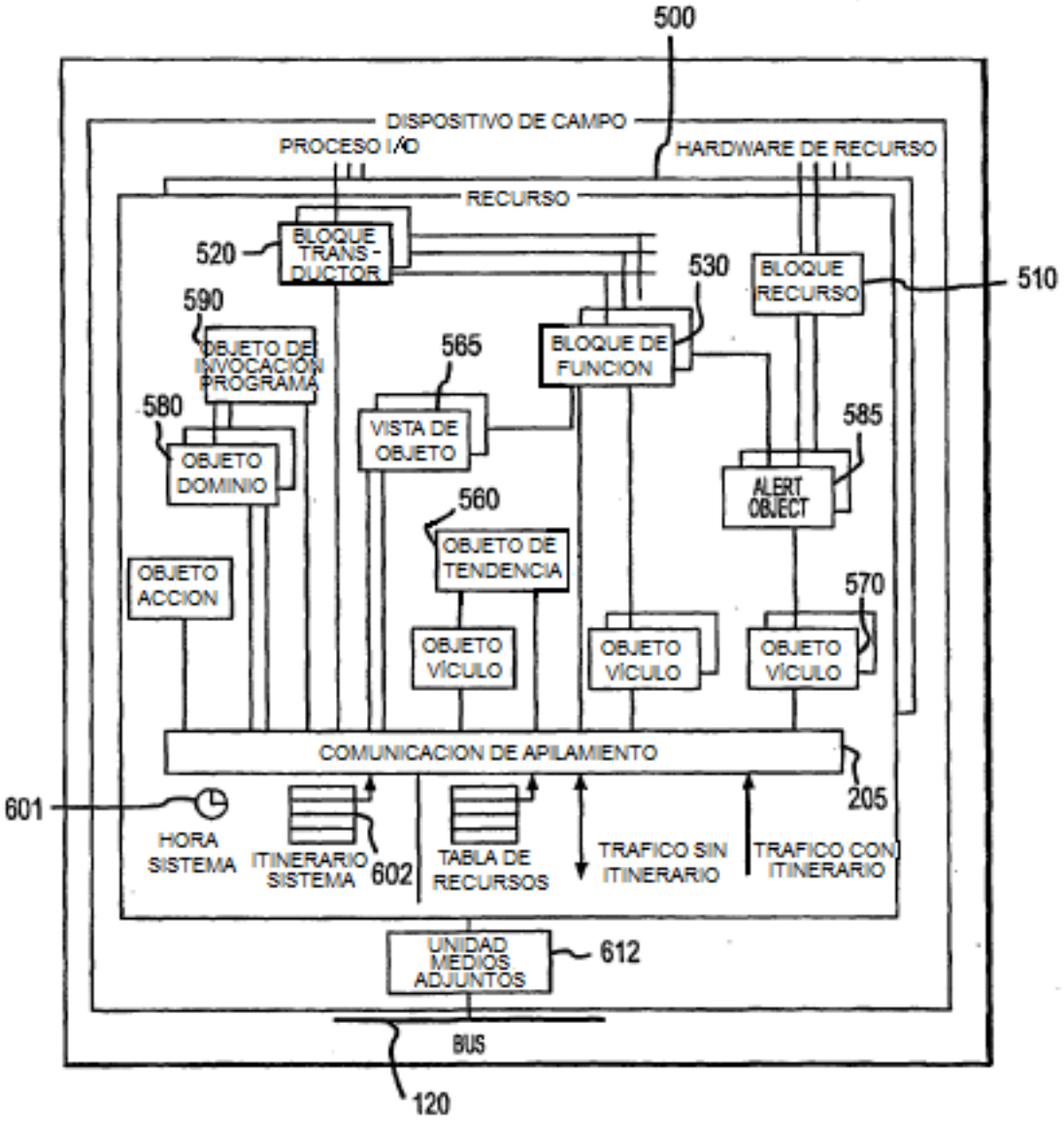

500
DISPOSITIVO DE CAMPO
PROCESO I/O
HARDWARE DE RECURSO
RECURSO
520
BLOQUE TRANS-DUCTOR
530
BLOQUE RECURSO
510
590
OBJETO DE INVOCACIÓN PROGRAMA
565
BLOQUE DE FUNCION
580
VISTA DE OBJETO
585
OBJETO DOMINIO
ALERT OBJECT
560
OBJETO DE TENDENCIA
OBJETO ACCION
570
OBJETO VÍCULO
OBJETO VÍCULO
OBJETO VÍCULO
COMUNICACION DE APILAMIENTO
205
601
HORA SISTEMA
ITINERARIO SISTEMA
602
TABLA DE RECURSOS
TRAFICO SIN ITINERARIO
TRAFICO CON ITINERARIO
UNIDAD MEDIOS ADJUNTOS
612
BUS
120

FIG.8A

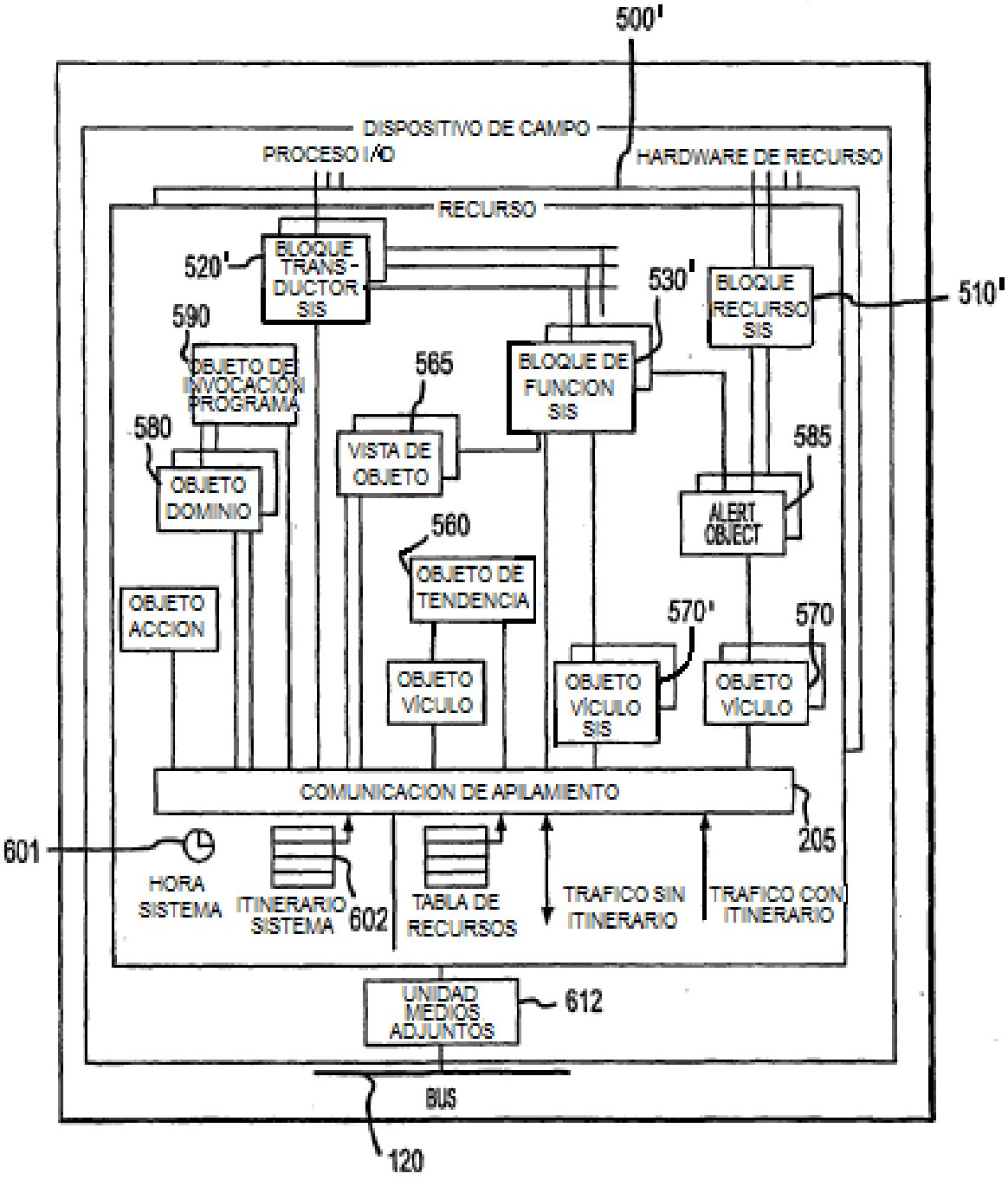

500'
DISPOSITIVO DE CAMPO
PROCESO I/O
HARDWARE DE RECURSO
RECURSO
520'
BLOQUE TRANS-DUCTOR SIS
530'
BLOQUE RECURSO SIS
510'
590
OBJETO DE INVOCACIÓN PROGRAMA
565
VISTA DE OBJETO
BLOQUE DE FUNCION SIS
580
585
OBJETO DOMINIO
ALERT OBJECT
560
OBJETO DE TENDENCIA
OBJETO ACCION
570'
570
OBJETO VÍCULO
OBJETO VÍCULO SIS
OBJETO VÍCULO
COMUNICACION DE APILAMIENTO
205
601
HORA SISTEMA
ITINERARIO SISTEMA
602
TABLA DE RECURSOS
TRAFICO SIN ITINERARIO
TRAFICO CON ITINERARIO
UNIDAD MEDIOS ADJUNTOS
612
BUS
120

FIG.8B

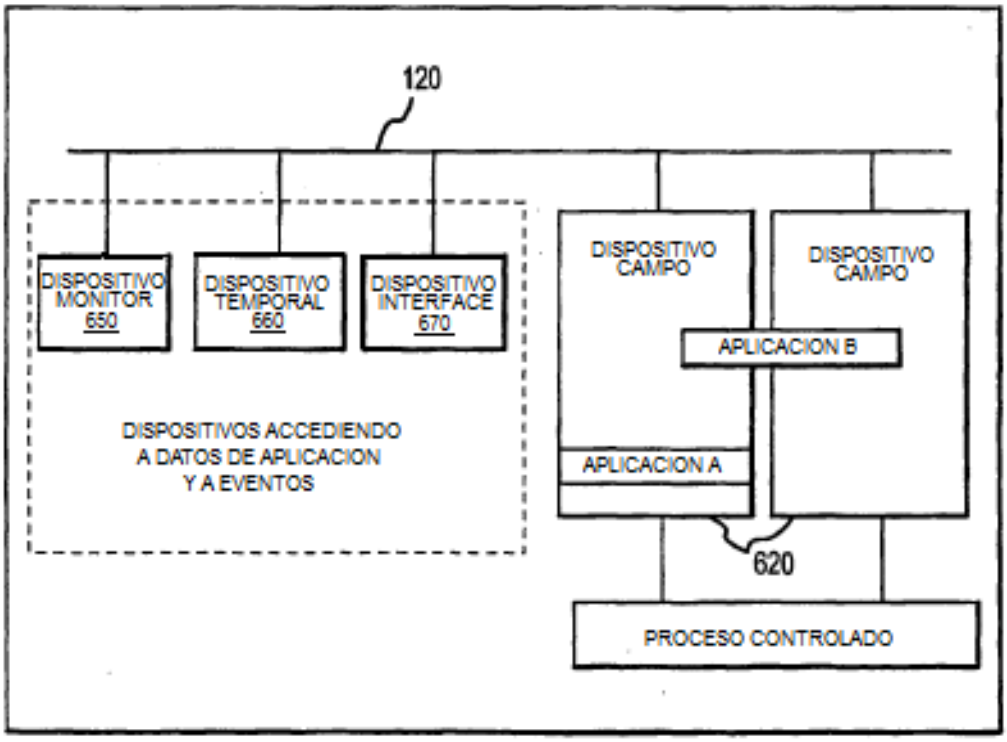

120
DISPOSITIVO MONITOR 650
DISPOSITIVO TEMPORAL 660
DISPOSITIVO INTERFACE 670
DISPOSITIVOS ACCEDIENDO A DATOS DE APLICACION Y A EVENTOS
DISPOSITIVO CAMPO
DISPOSITIVO CAMPO
APLICACION B
APLICACION A
620
PROCESO CONTROLADO

FIG.9A

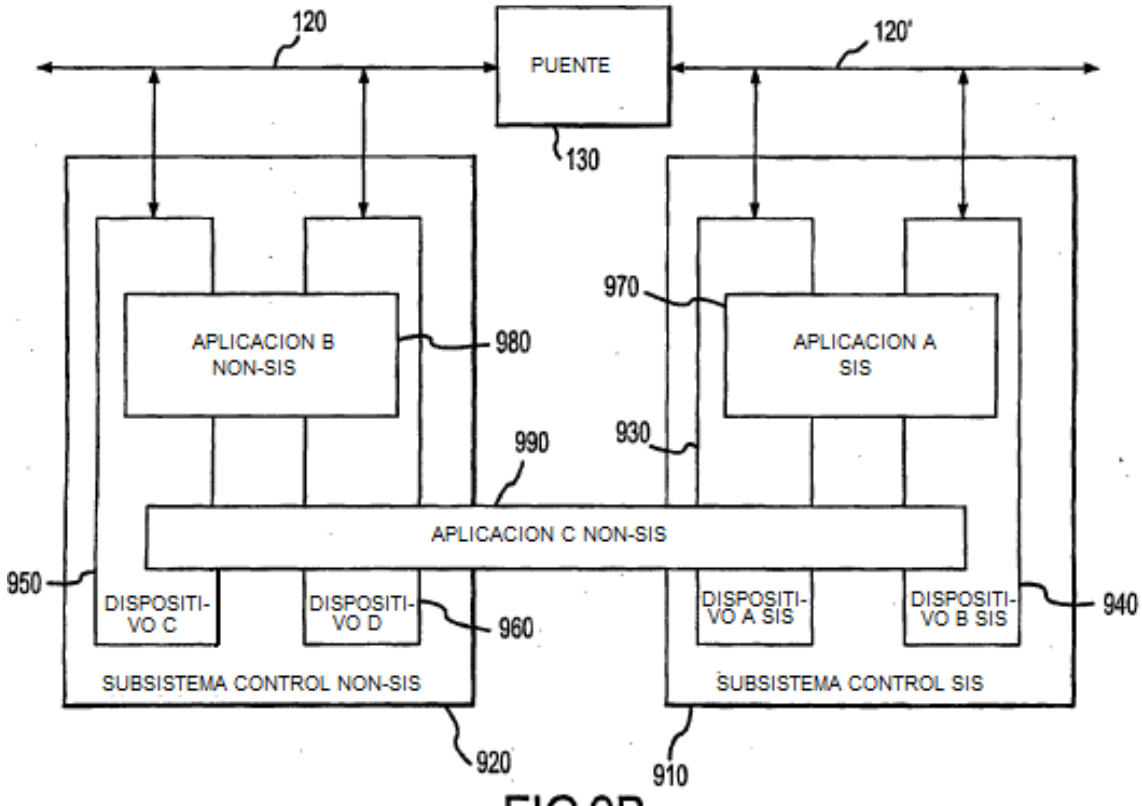

120
PUENTE
120'
130
970
APLICACION B NON-SIS
980
APLICACION A SIS
930
990
APLICACION C NON-SIS
950
DISPOSITI-VO C
DISPOSITI-VO D
960
DISPOSITI-VO A SIS
DISPOSITI-VO B SIS
940
SUBSISTEMA CONTROL NON-SIS
SUBSISTEMA CONTROL SIS
920
910

FIG.9B

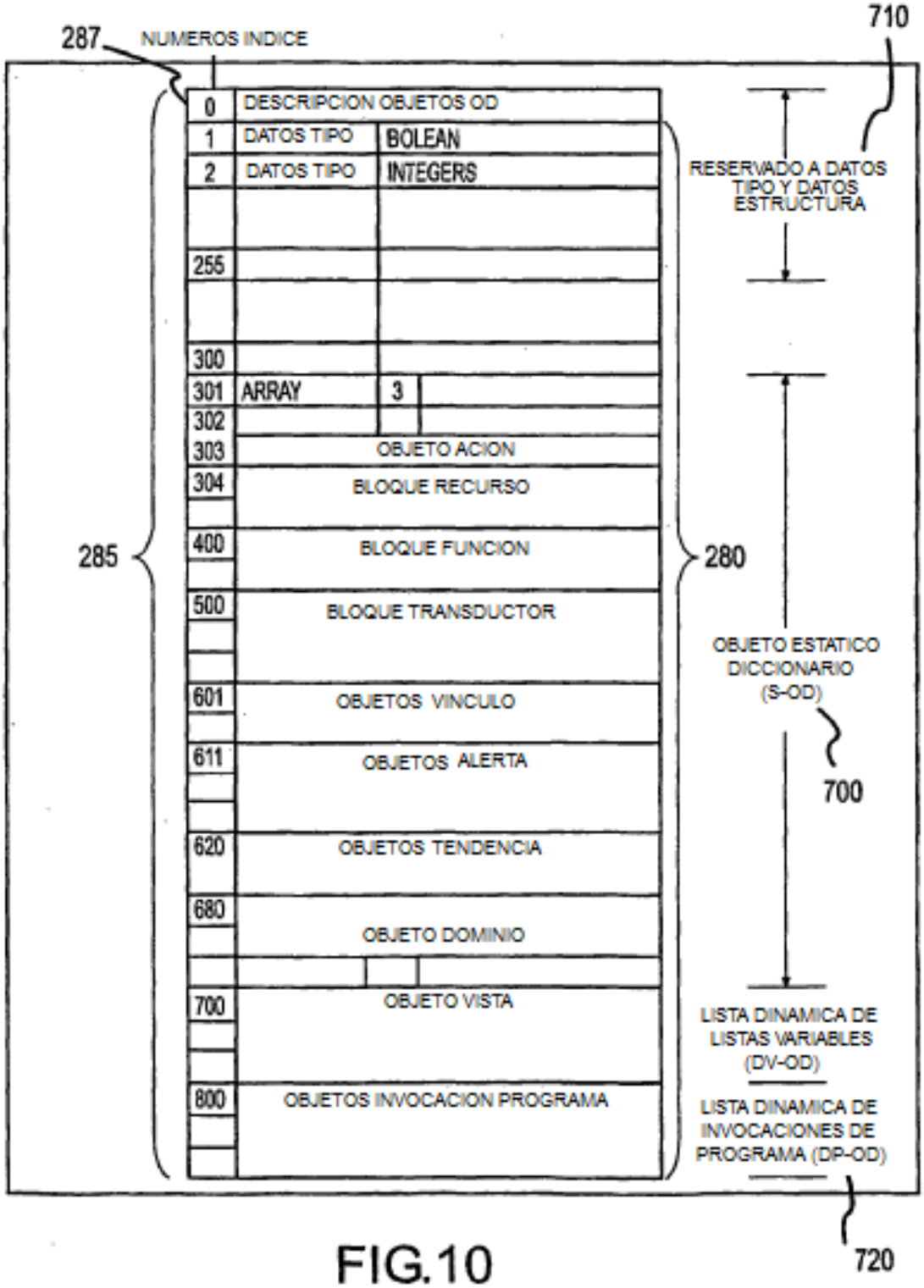

287
NUMEROS INDICE
710
0
DESCRIPCION OBJETOS OD
1
DATOS TIPO
BOLEAN
2
DATOS TIPO
INTEGERS
255
300
301
ARRAY
3
302
303
OBJETO ACION
304
BLOQUE RECURSO
400
BLOQUE FUNCION
500
BLOQUE TRANSDUCTOR
601
OBJETOS VINCULO
611
OBJETOS ALERTA
620
OBJETOS TENDENCIA
680
OBJETO DOMINIO
700
OBJETO VISTA
800
OBJETOS INVOCACION PROGRAMA
285
280
RESERVADO A DATOS TIPO Y DATOS ESTRUCTURA
OBJETO ESTATICO DICCIONARIO (S-OD)
700
LISTA DINAMICA DE LISTAS VARIABLES (DV-OD)
LISTA DINAMICA DE INVOCACIONES DE PROGRAMA (DP-OD)
720

FIG.10

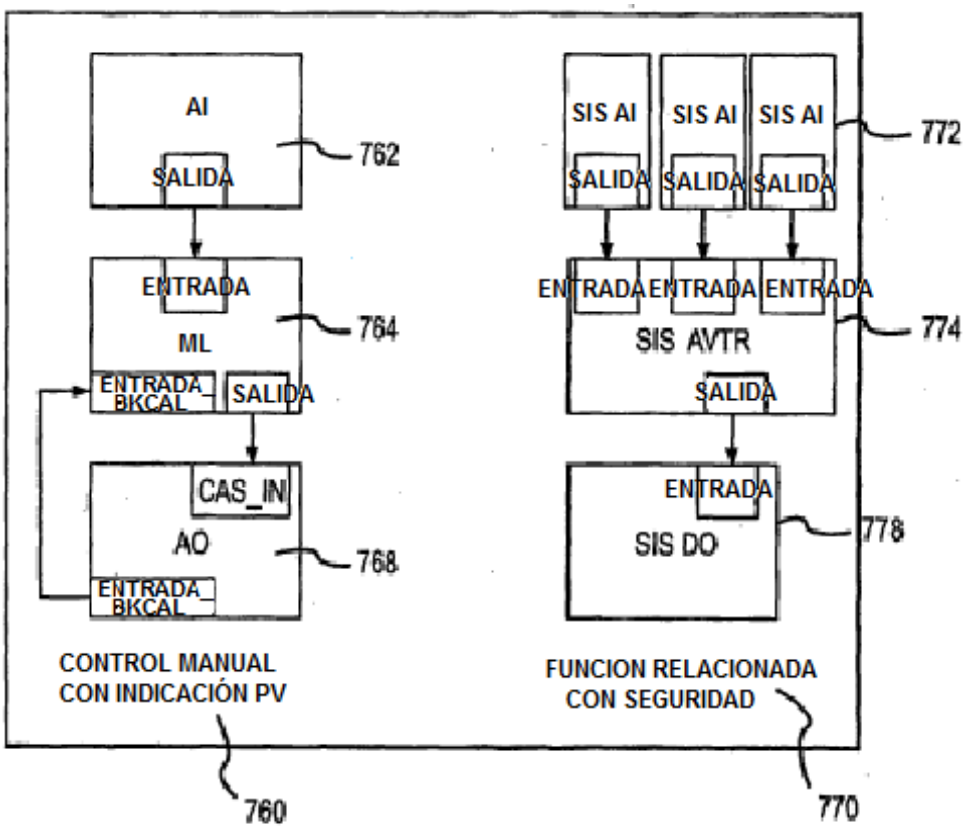

AI
SALIDA
762
ENTRADA
ML
764
ENTRADA BKCAL
SALIDA
CAS_IN
AO
768
ENTRADA BKCAL
SIS AI
SIS AI
SIS AI
772
SALIDA
SALIDA
SALIDA
ENTRADA
ENTRADA
ENTRADA
SIS AVTR
774
SALIDA
ENTRADA
SIS DO
778
CONTROL MANUAL
CON INDICACIÓN PV
760
FUNCION RELACIONADA
CON SEGURIDAD
770

FIG.11

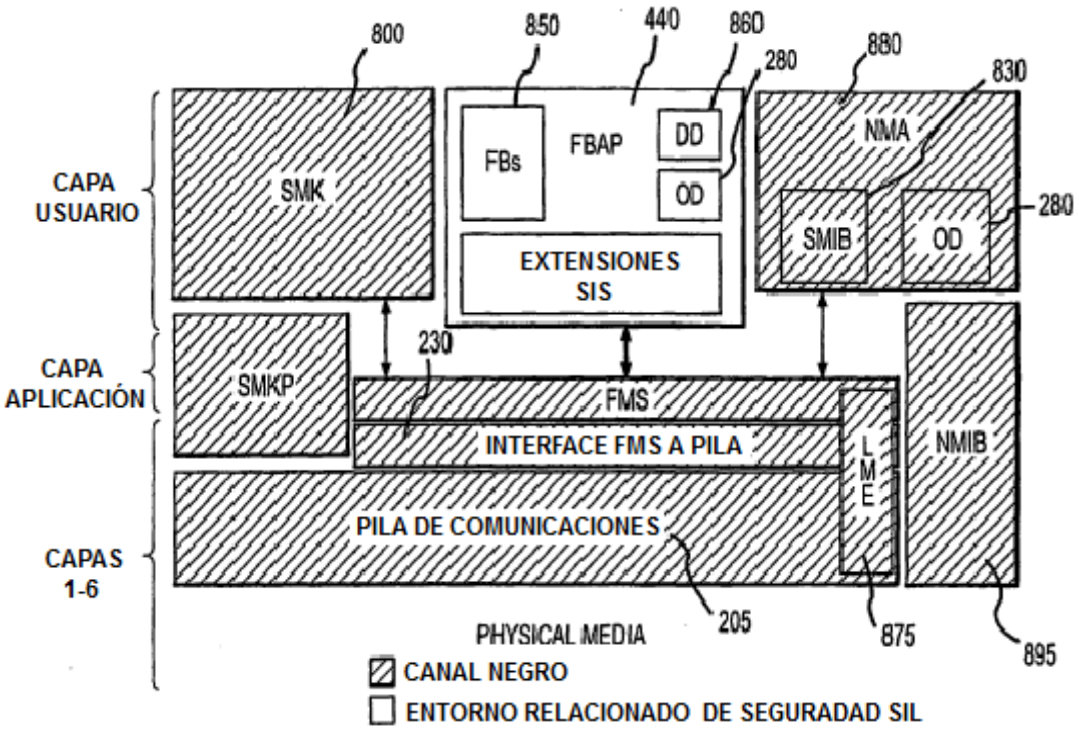

800
850
440
860
280
880
830
CAPA
USUARIO
SMK
FBs
FBAP
DD
OD
EXTENSIONES
SIS
NMA
SMIB
OD
280
230
CAPA
APLICACIÓN
SMKP
FMS
INTERFACE FMS A PILA
L
M
E
NMIB
CAPAS
1-6
PILA DE COMUNICACIONES
205
875
895
PHYSICAL MEDIA
CANAL NEGRO
ENTORNO RELACIONADO DE SEGURADAD SIL

FIG.12

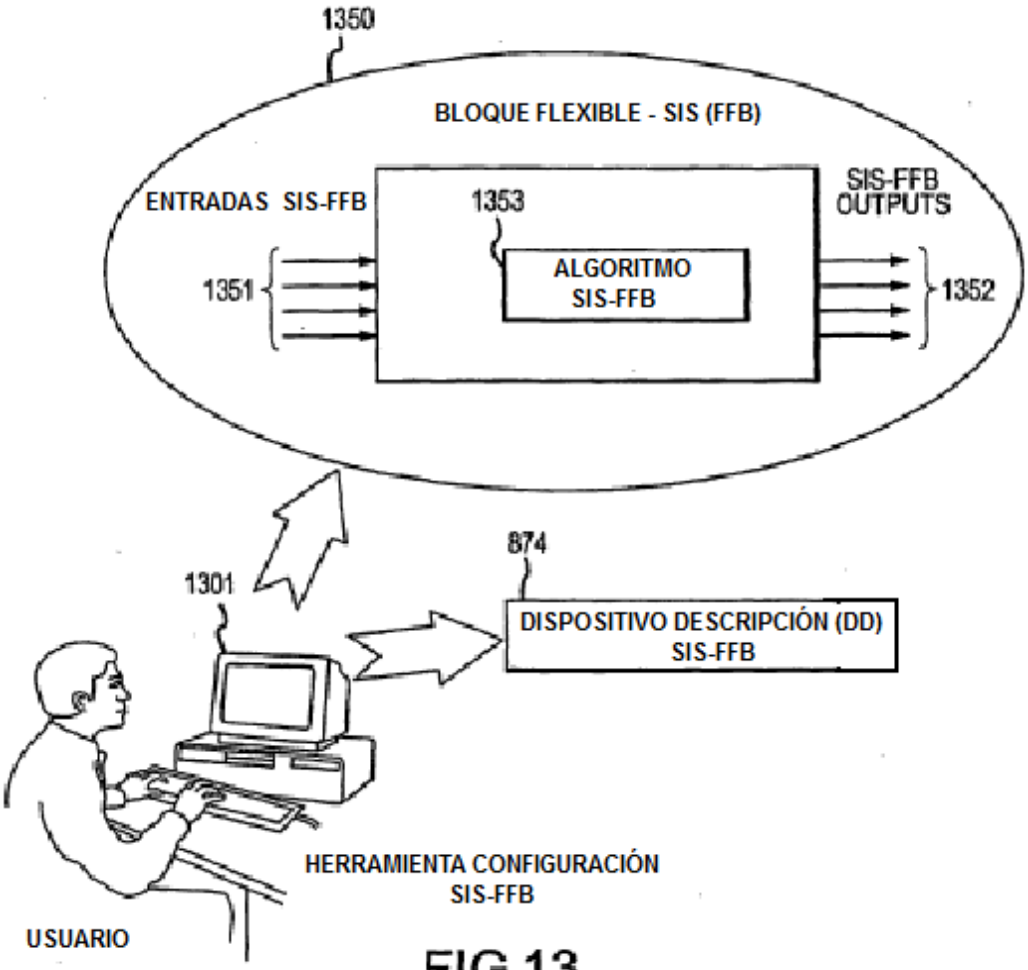

1350
BLOQUE FLEXIBLE - SIS (FFB)
ENTRADAS SIS-FFB
1353
SIS-FFB OUTPUTS
1351
ALGORITMO SIS-FFB
1352
874
1301
DISPOSITIVO DESCRIPCIÓN (DD) SIS-FFB
HERRAMIENTA CONFIGURACIÓN SIS-FFB
USUARIO

FIG.13

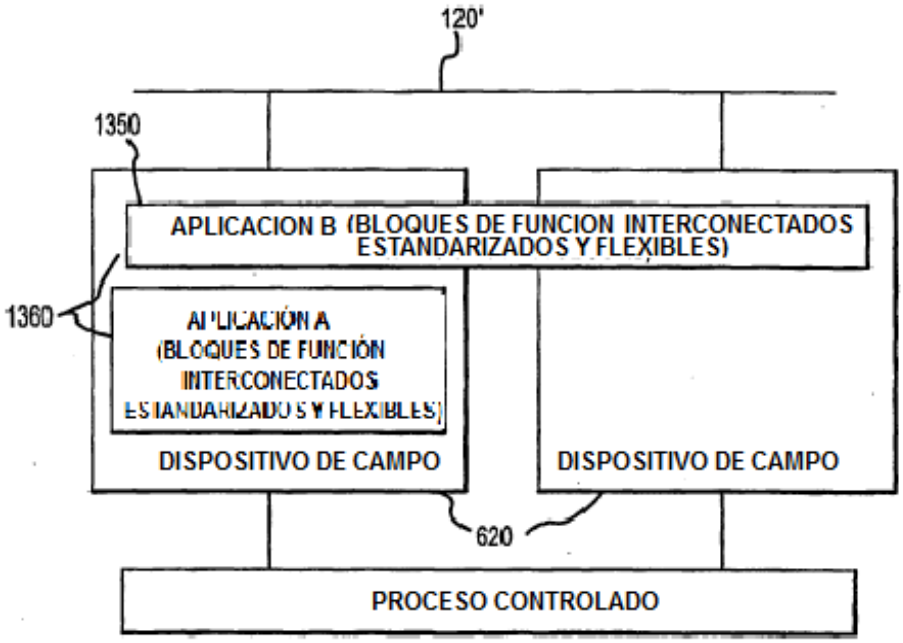

120'
1350
APLICACION B (BLOQUES DE FUNCION INTERCONECTADOS ESTANDARIZADOS Y FLEXIBLES)
1360
APLICACIÓN A (BLOQUES DE FUNCIÓN INTERCONECTADOS ESTANDARIZADOS Y FLEXIBLES)
DISPOSITIVO DE CAMPO
DISPOSITIVO DE CAMPO
620
PROCESO CONTROLADO

FIG.14

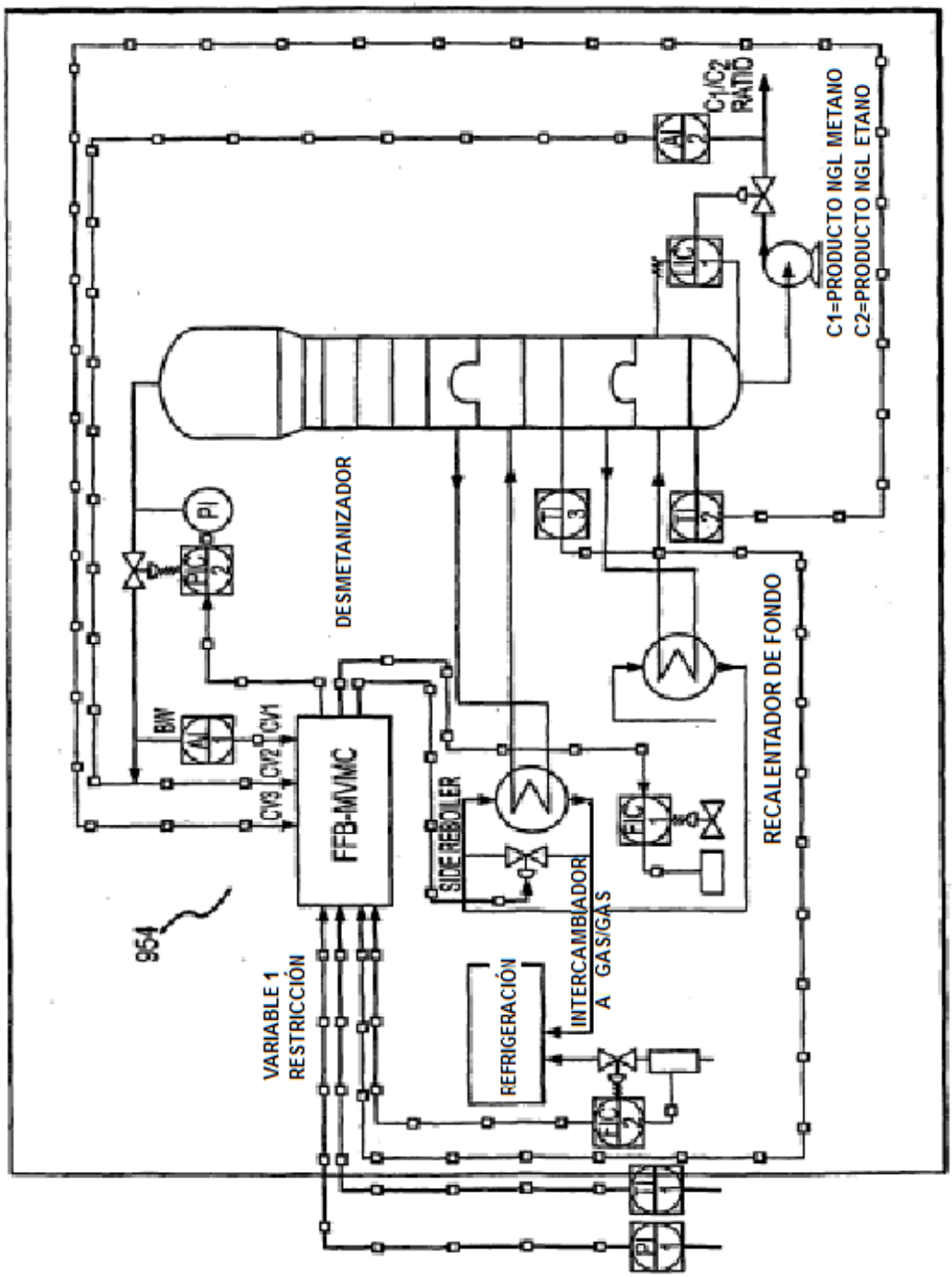

C1/C2 RATIO
C1=PRODUCTO NGL METANO
C2=PRODUCTO NGL ETANO
DESMETANIZADOR
RECALENTADOR DE FONDO
FFB-MVMC
SIDE REBOILER
INTERCAMBIADOR A GAS/GAS
REFRIGERACIÓN
VARIABLE 1 RESTRICCIÓN
954
BW
CV1
CV2
CV3

FIG.15

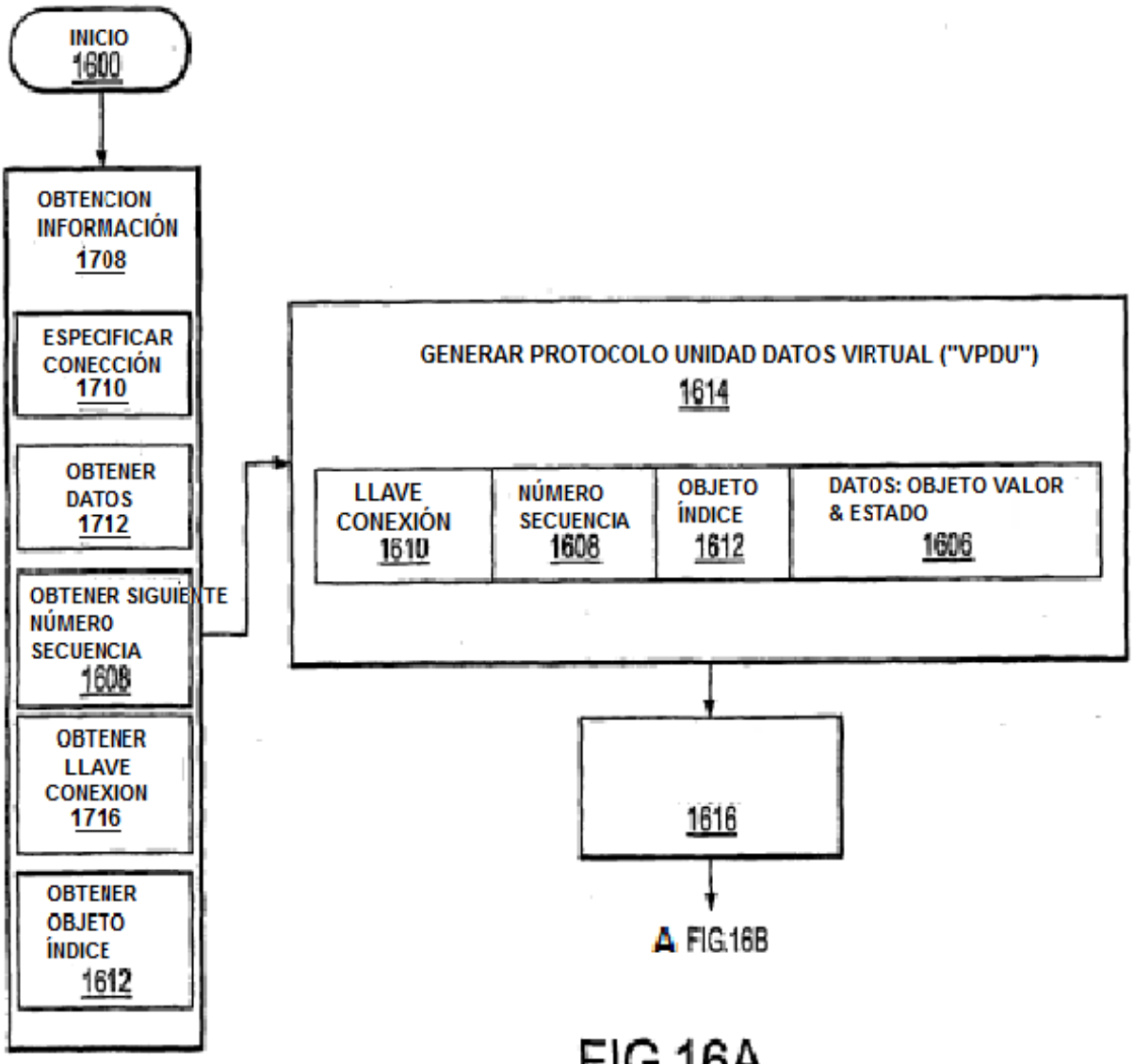

INICIO
1600
OBTENCION INFORMACIÓN
1708
ESPECIFICAR CONECCIÓN
1710
OBTENER DATOS
1712
OBTENER SIGUIENTE NÚMERO SECUENCIA
1608
OBTENER LLAVE CONEXION
1716
OBTENER OBJETO ÍNDICE
1612
GENERAR PROTOCOLO UNIDAD DATOS VIRTUAL ("VPDU")
1614
LLAVE CONEXIÓN
1610
NÚMERO SECUENCIA
1608
OBJETO ÍNDICE
1612
DATOS: OBJETO VALOR & ESTADO
1606
1616
A FIG.16B

FIG.16A

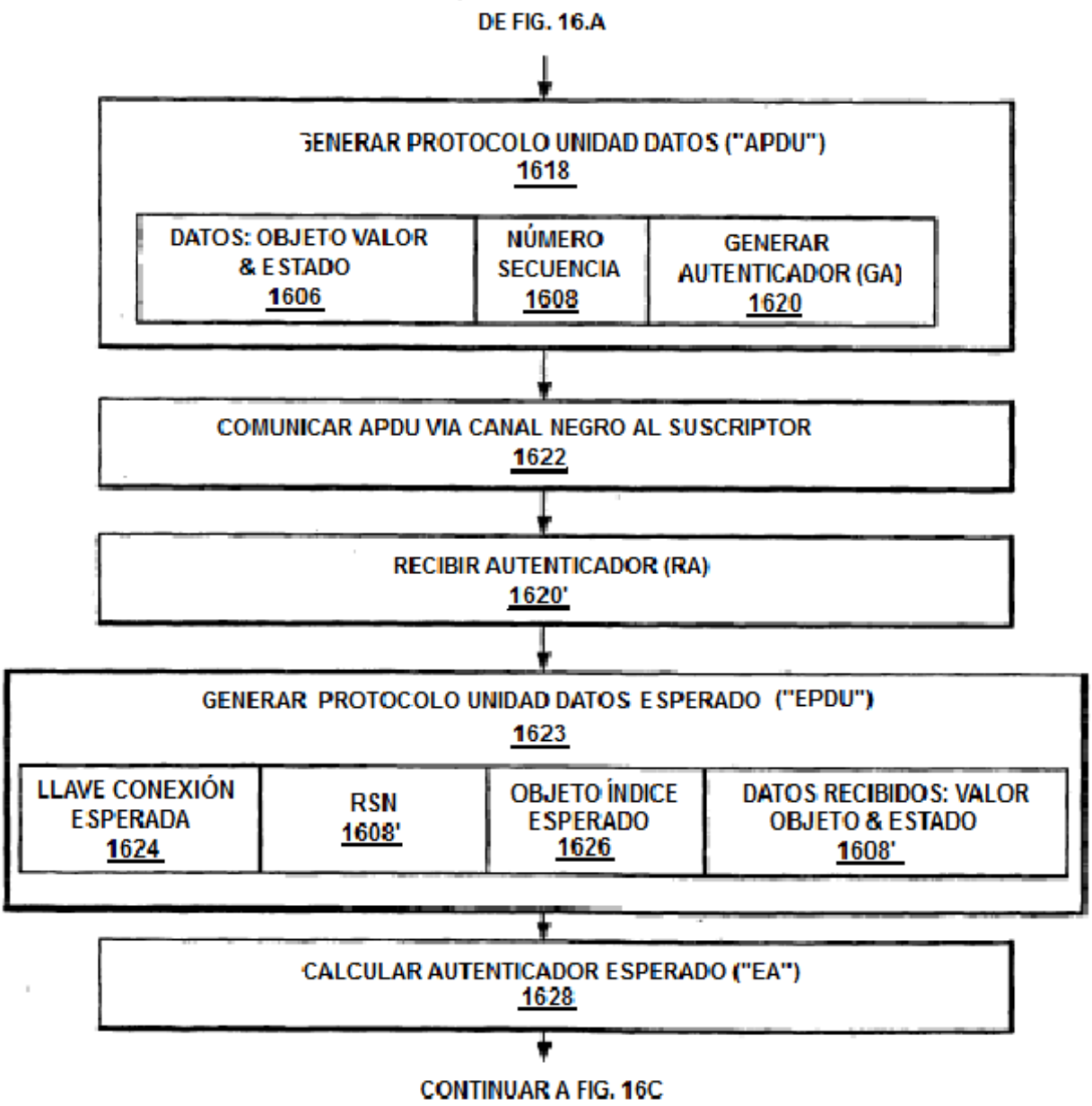

DE FIG. 16.A
GENERAR PROTOCOLO UNIDAD DATOS ("APDU")
1618
DATOS: OBJETO VALOR & ESTADO
1606
NÚMERO SECUENCIA
1608
GENERAR AUTENTICADOR (GA)
1620
COMUNICAR APDU VIA CANAL NEGRO AL SUSCRIPTOR
1622
RECIBIR AUTENTICADOR (RA)
1620'
GENERAR PROTOCOLO UNIDAD DATOS ESPERADO ("EPDU")
1623
LLAVE CONEXIÓN ESPERADA
1624
RSN
1608'
OBJETO ÍNDICE ESPERADO
1626
DATOS RECIBIDOS: VALOR OBJETO & ESTADO
1608'
CALCULAR AUTENTICADOR ESPERADO ("EA")
1628
CONTINUAR A FIG. 16C

FIG.16B

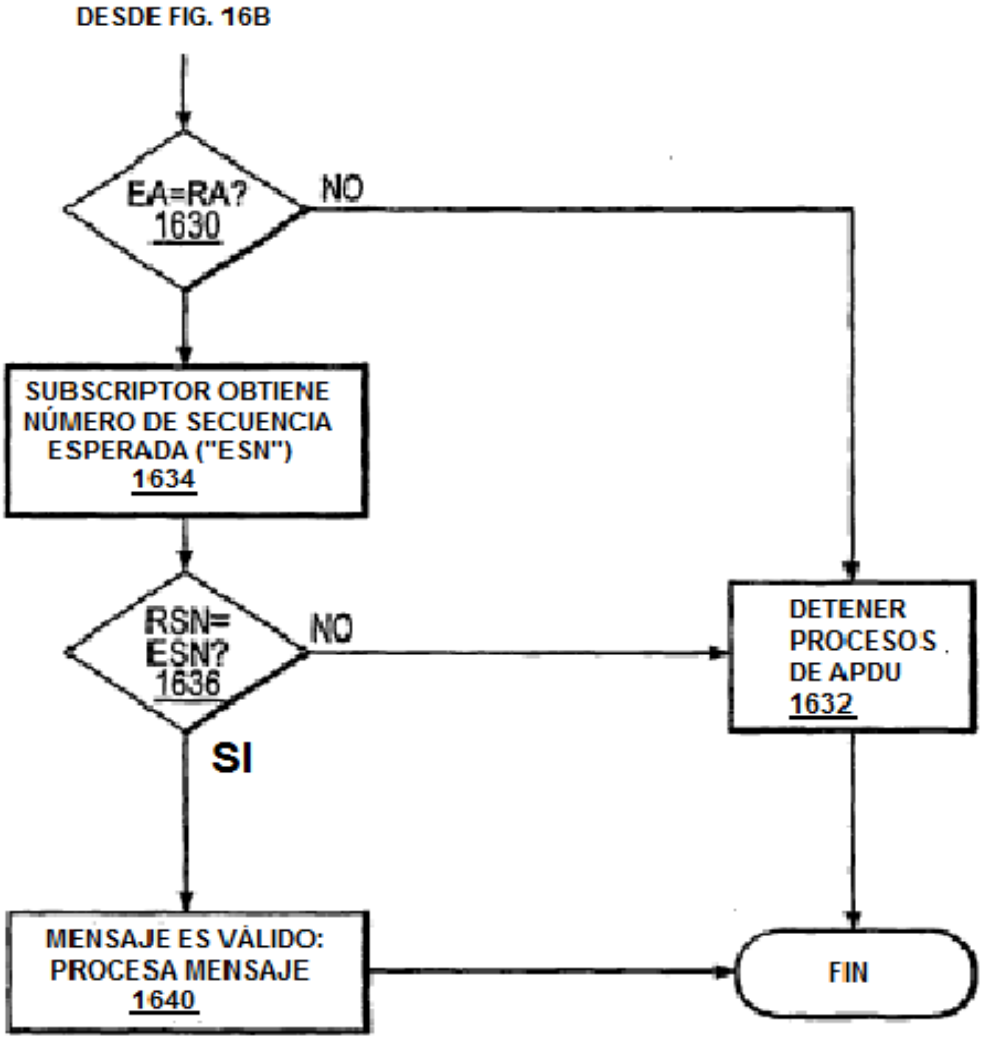

DESDE FIG. 16B
EA=RA?
1630
NO
SUBSCRIPTOR OBTIENE
NÚMERO DE SECUENCIA
ESPERADA ("ESN")
1634
RSN=
ESN?
1636
NO
SI
DETENER
PROCESOS
DE APDU
1632
MENSAJE ES VÁLIDO:
PROCESA MENSAJE
1640
FIN

FIG.16C

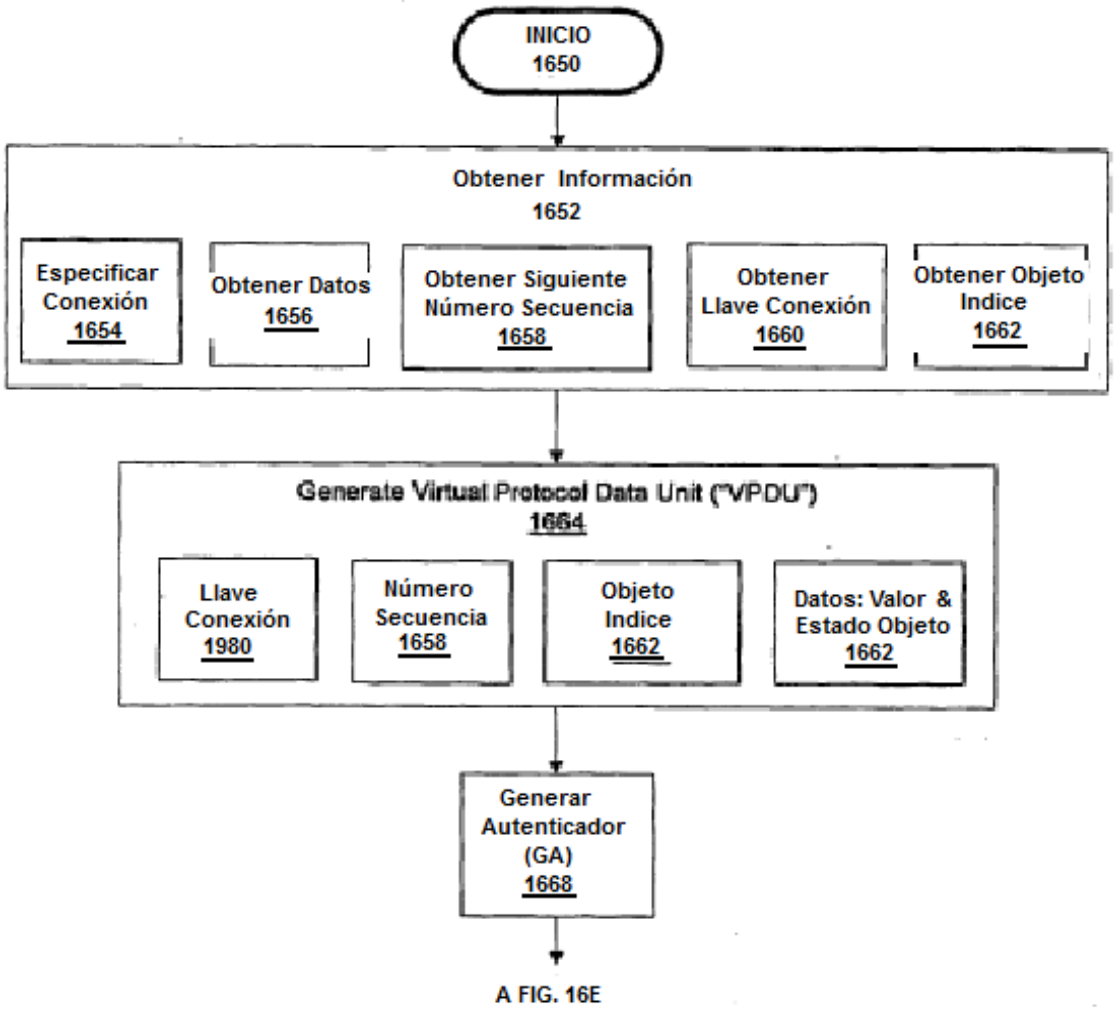
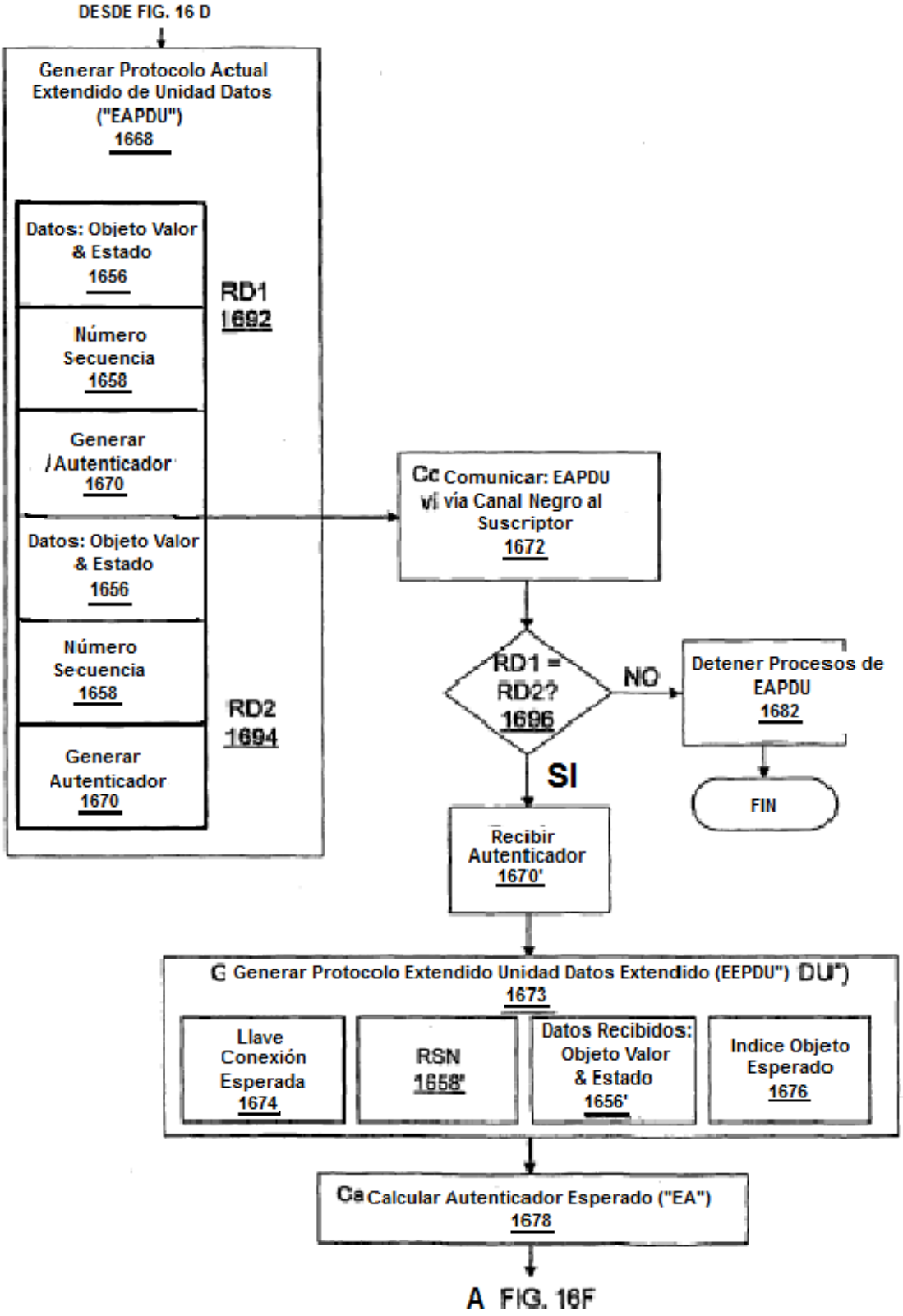

INICIO
1650
Obtener Información
1652
Especificar Conexión
1654
Obtener Datos
1656
Obtener Siguiente Número Secuencia
1658
Obtener Llave Conexión
1660
Obtener Objeto Indice
1662
Generate Virtual Protocol Data Unit ("VPDU")
1664
Llave Conexión
1980
Número Secuencia
1658
Objeto Indice
1662
Datos: Valor & Estado Objeto
1662
Generar Autenticador (GA)
1668
A FIG. 16E

FIG. 16D


DESDE FIG. 16 D
Generar Protocolo Actual Extendido de Unidad Datos ("EAPDU")
1668
Datos: Objeto Valor & Estado
1656
Número Secuencia
1658
Generar /Autenticador
1670
RD1
1692
Datos: Objeto Valor & Estado
1656
Número Secuencia
1658
Generar Autenticador
1670
RD2
1694
Cc Comunicar: EAPDU vi vía Canal Negro al Suscriptor
1672
RD1 = RD2?
1696
NO
Detener Procesos de EAPDU
1682
FIN
SI
Recibir Autenticador
1670'
G Generar Protocolo Extendido Unidad Datos Extendido (EEPDU") DU")
1673
Llave Conexión Esperada
1674
RSN
1658'
Datos Recibidos: Objeto Valor & Estado
1656'
Indice Objeto Esperado
1676
Ca Calcular Autenticador Esperado ("EA")
1678
A FIG. 16F

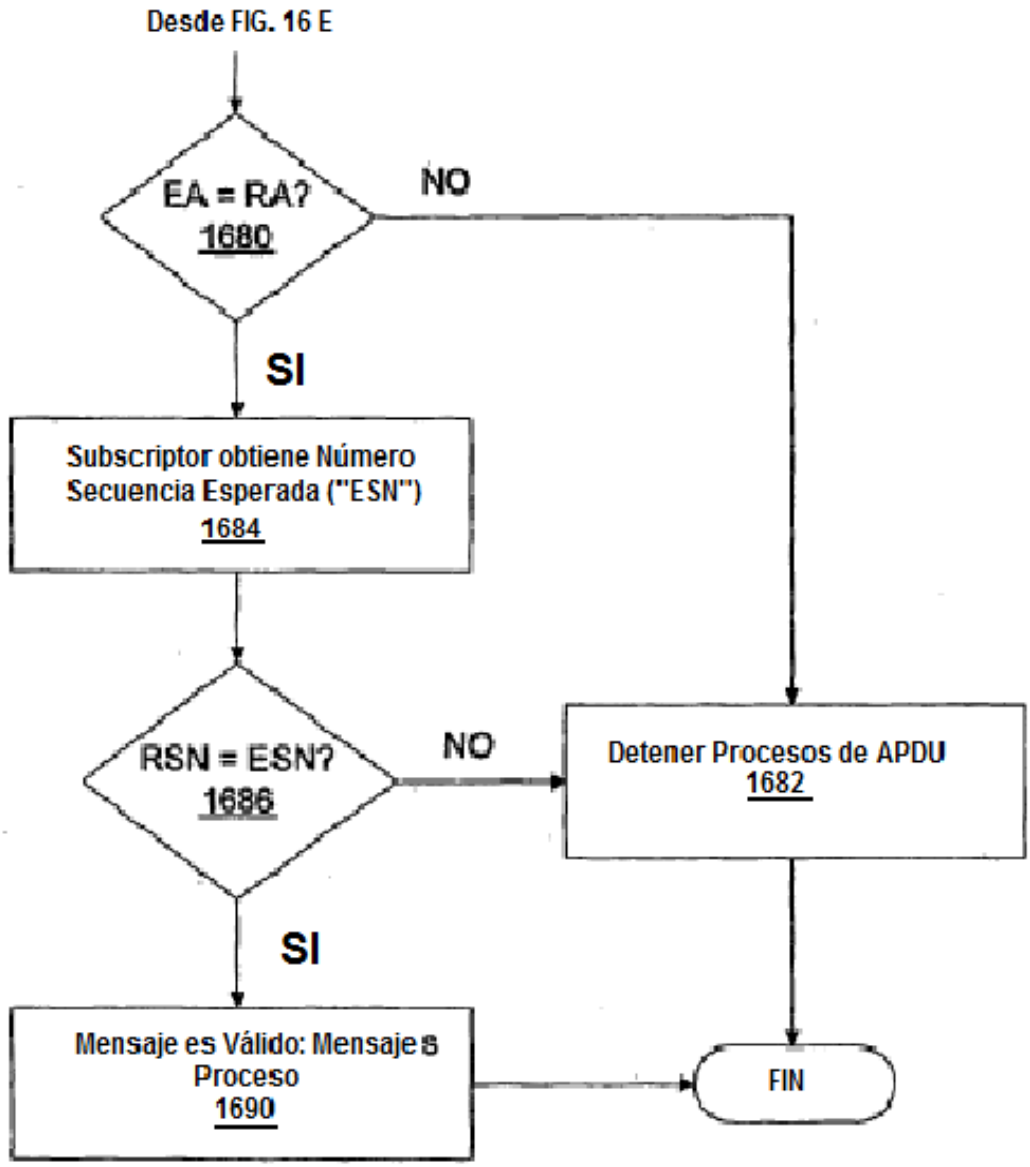

Desde FIG. 16 E
EA = RA?
1680
NO
SI
Subscriptor obtiene Número Secuencia Esperada ("ESN")
1684
RSN = ESN?
1686
NO
Detener Procesos de APDU
1682
SI
Mensaje es Válido: Mensaje s Proceso
1690
FIN

FIG. 16F

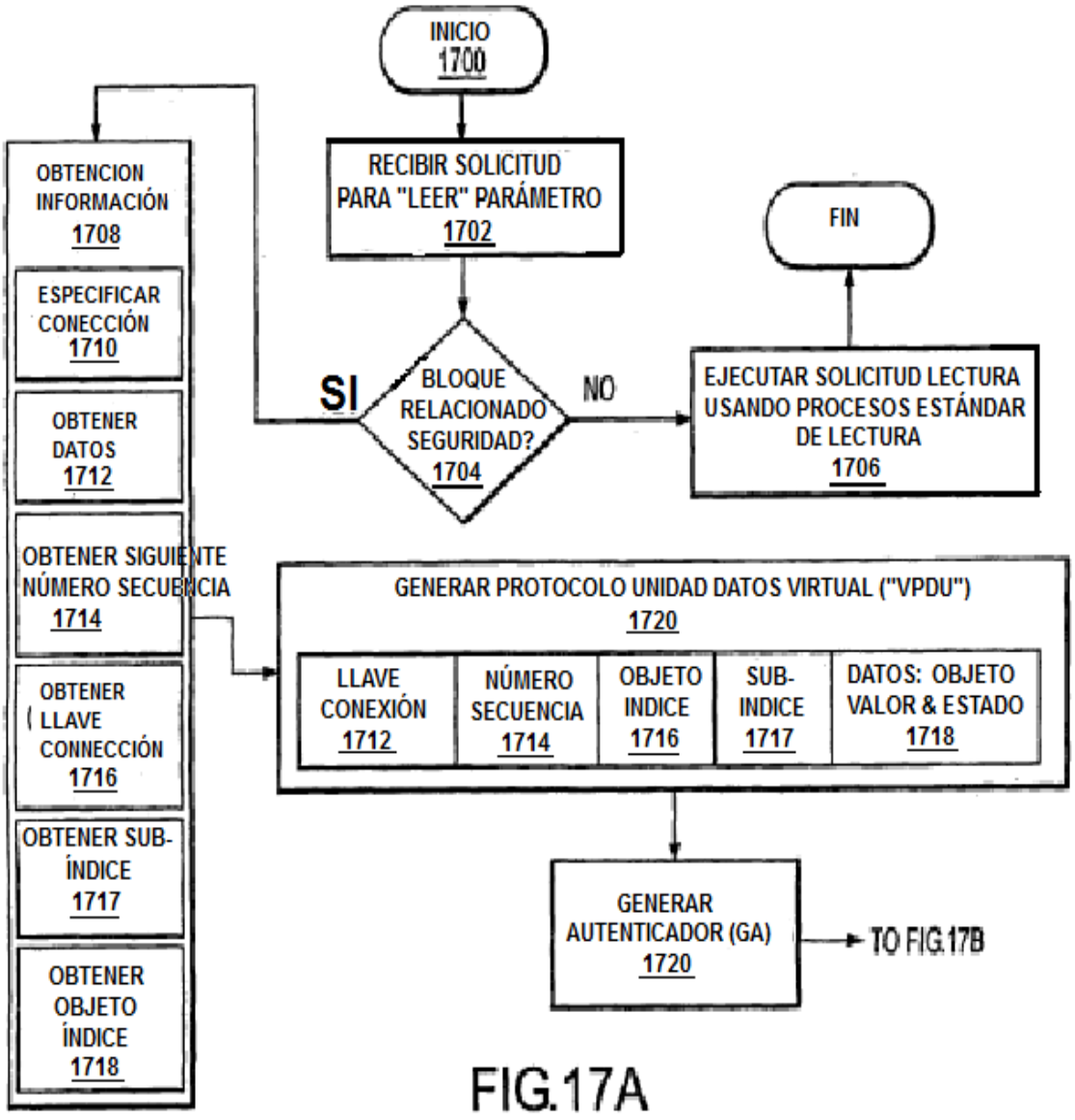

INICIO
1700
RECIBIR SOLICITUD PARA "LEER" PARÁMETRO
1702
BLOQUE RELACIONADO SEGURIDAD?
1704
SI
NO
EJECUTAR SOLICITUD LECTURA USANDO PROCESOS ESTÁNDAR DE LECTURA
1706
FIN
OBTENCION INFORMACIÓN
1708
ESPECIFICAR CONECCIÓN
1710
OBTENER DATOS
1712
OBTENER SIGUIENTE NÚMERO SECUENCIA
1714
OBTENER LLAVE CONNECCIÓN
1716
OBTENER SUB-ÍNDICE
1717
OBTENER OBJETO ÍNDICE
1718
GENERAR PROTOCOLO UNIDAD DATOS VIRTUAL ("VPDU")
1720
LLAVE CONEXIÓN
1712
NÚMERO SECUENCIA
1714
OBJETO INDICE
1716
SUB-INDICE
1717
DATOS: OBJETO VALOR & ESTADO
1718
GENERAR AUTENTICADOR (GA)
1720
TO FIG.17B

FIG.17A

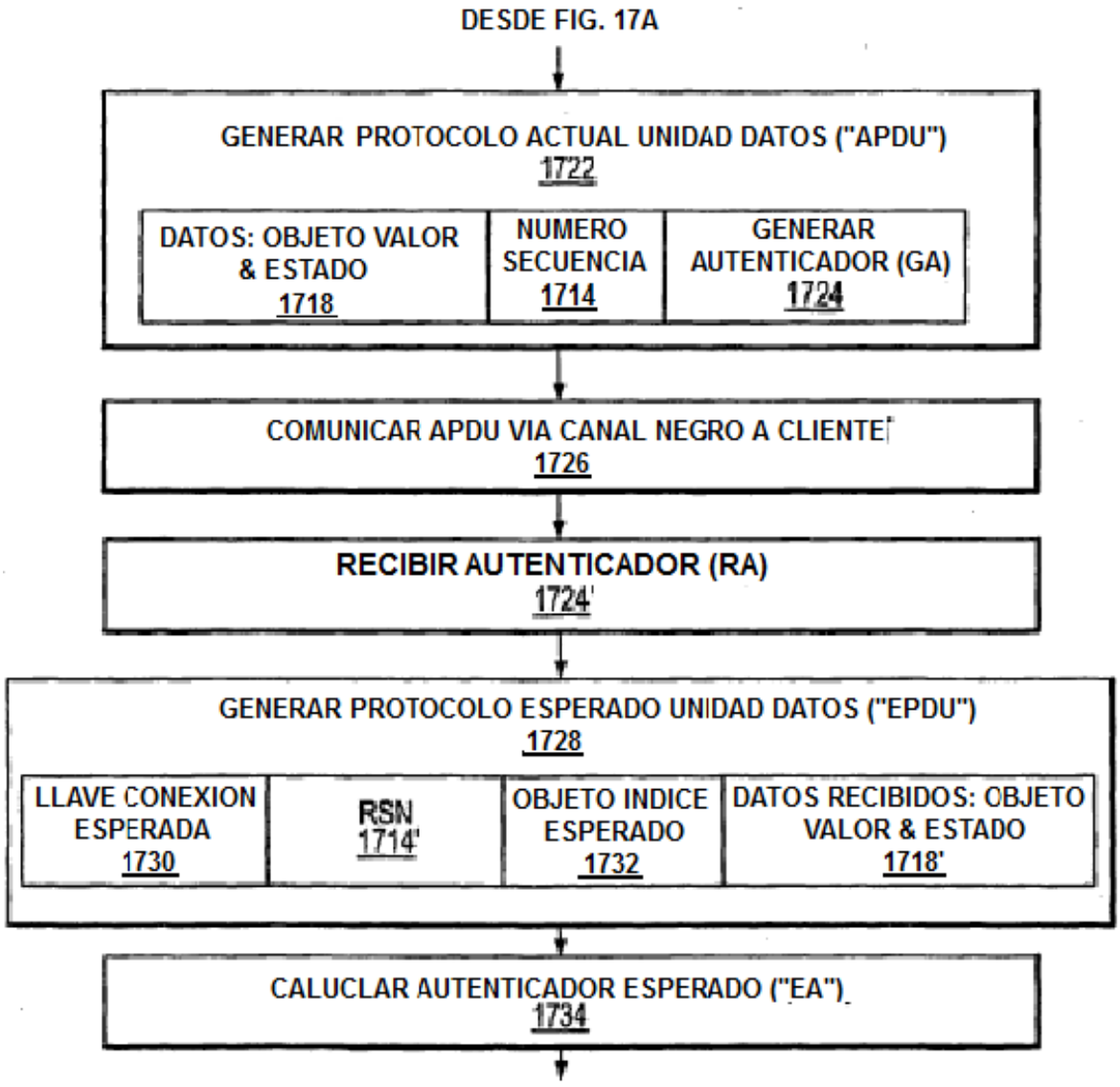

DESDE FIG. 17A
GENERAR PROTOCOLO ACTUAL UNIDAD DATOS ("APDU")
1722
DATOS: OBJETO VALOR & ESTADO
1718
NUMERO SECUENCIA
1714
GENERAR AUTENTICADOR (GA)
1724
COMUNICAR APDU VIA CANAL NEGRO A CLIENTE
1726
RECIBIR AUTENTICADOR (RA)
1724'
GENERAR PROTOCOLO ESPERADO UNIDAD DATOS ("EPDU")
1728
LLAVE CONEXION ESPERADA
1730
RSN
1714'
OBJETO INDICE ESPERADO
1732
DATOS RECIBIDOS: OBJETO VALOR & ESTADO
1718'
CALUCLAR AUTENTICADOR ESPERADO ("EA")
1734
CONTINUAR A FIG. 17C

FIG.17B

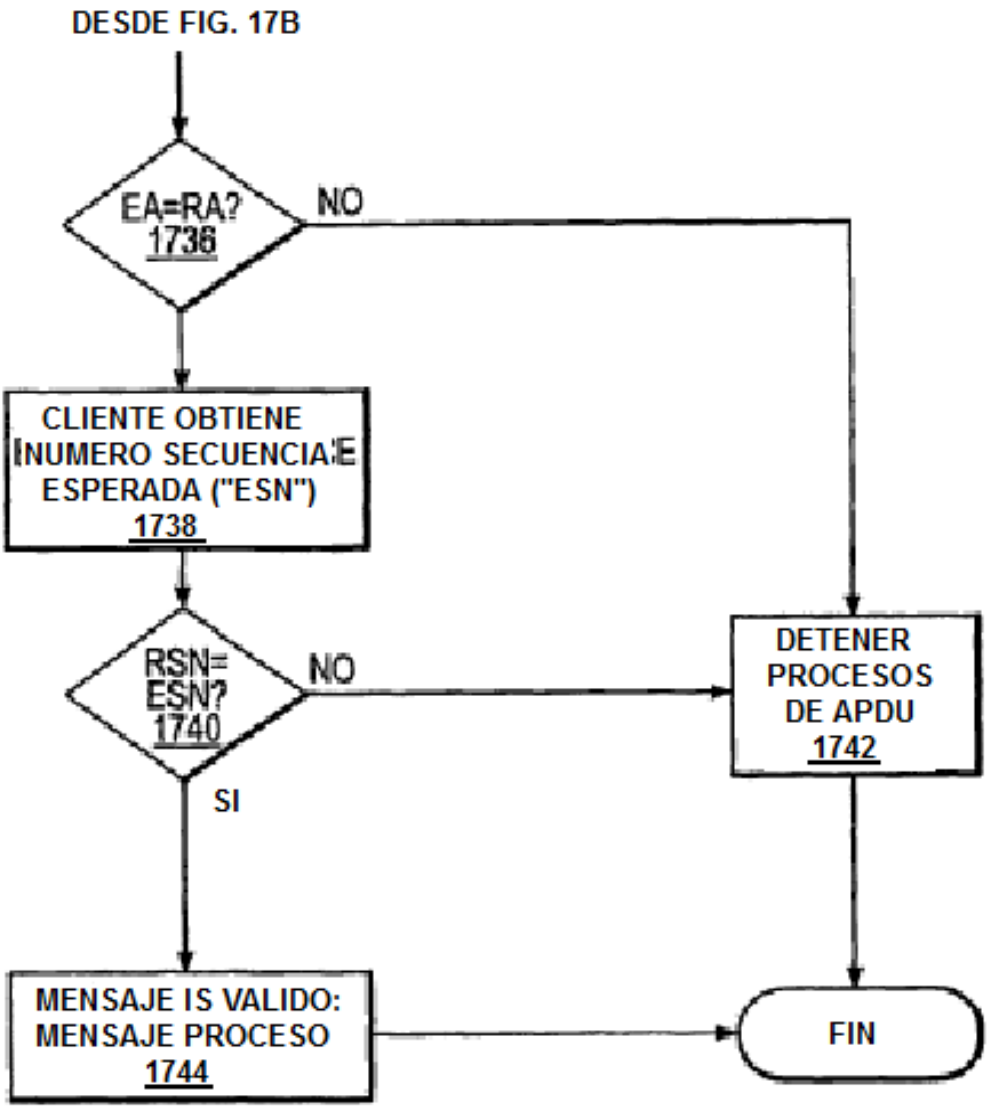

DESDE FIG. 17B
EA=RA?
1736
NO
CLIENTE OBTIENE NUMERO SECUENCIA ESPERADA ("ESN")
1738
RSN= ESN?
1740
NO
SI
DETENER PROCESOS DE APDU
1742
MENSAJE IS VALIDO: MENSAJE PROCESO
1744
FIN

FIG.17C

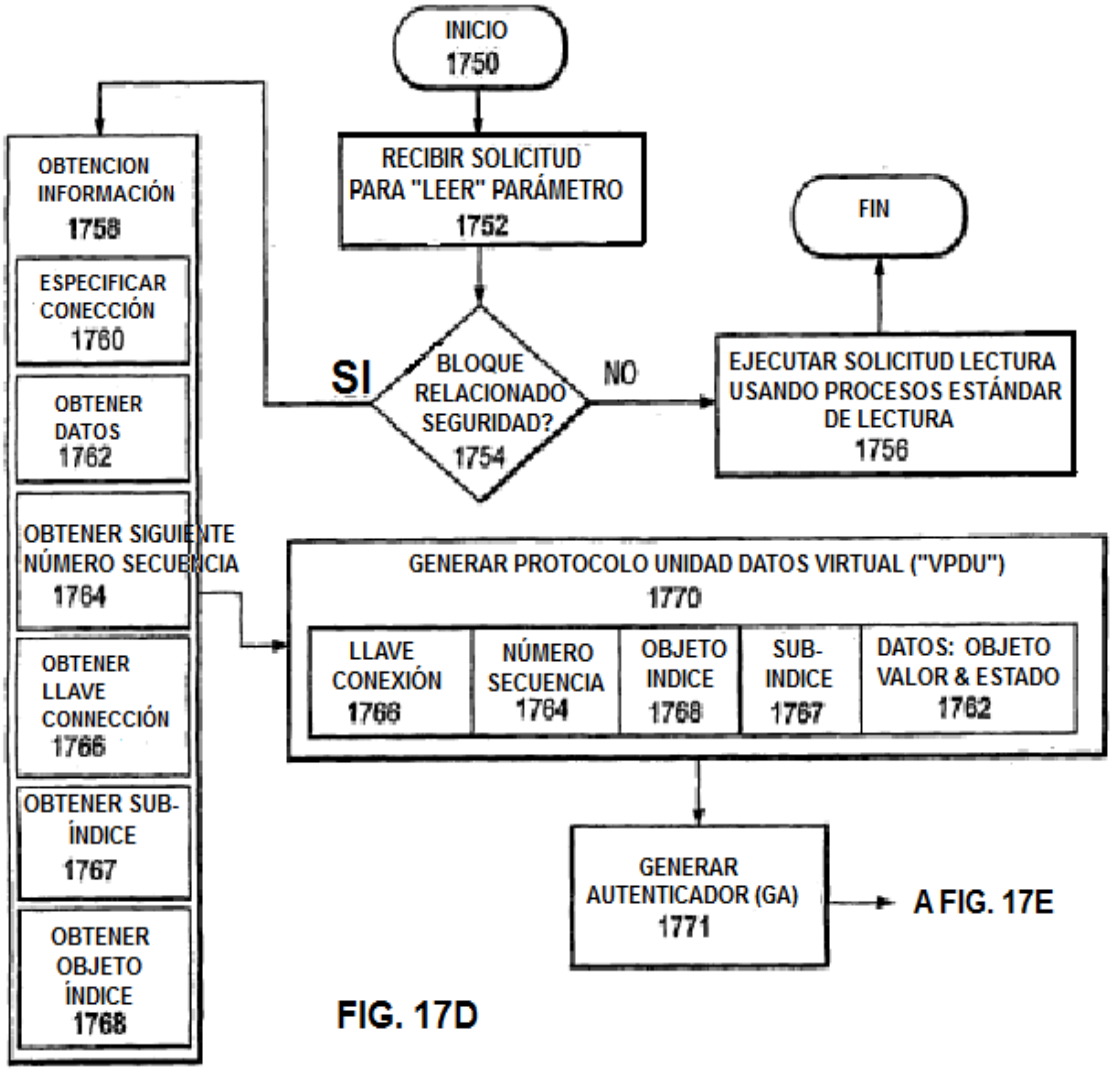
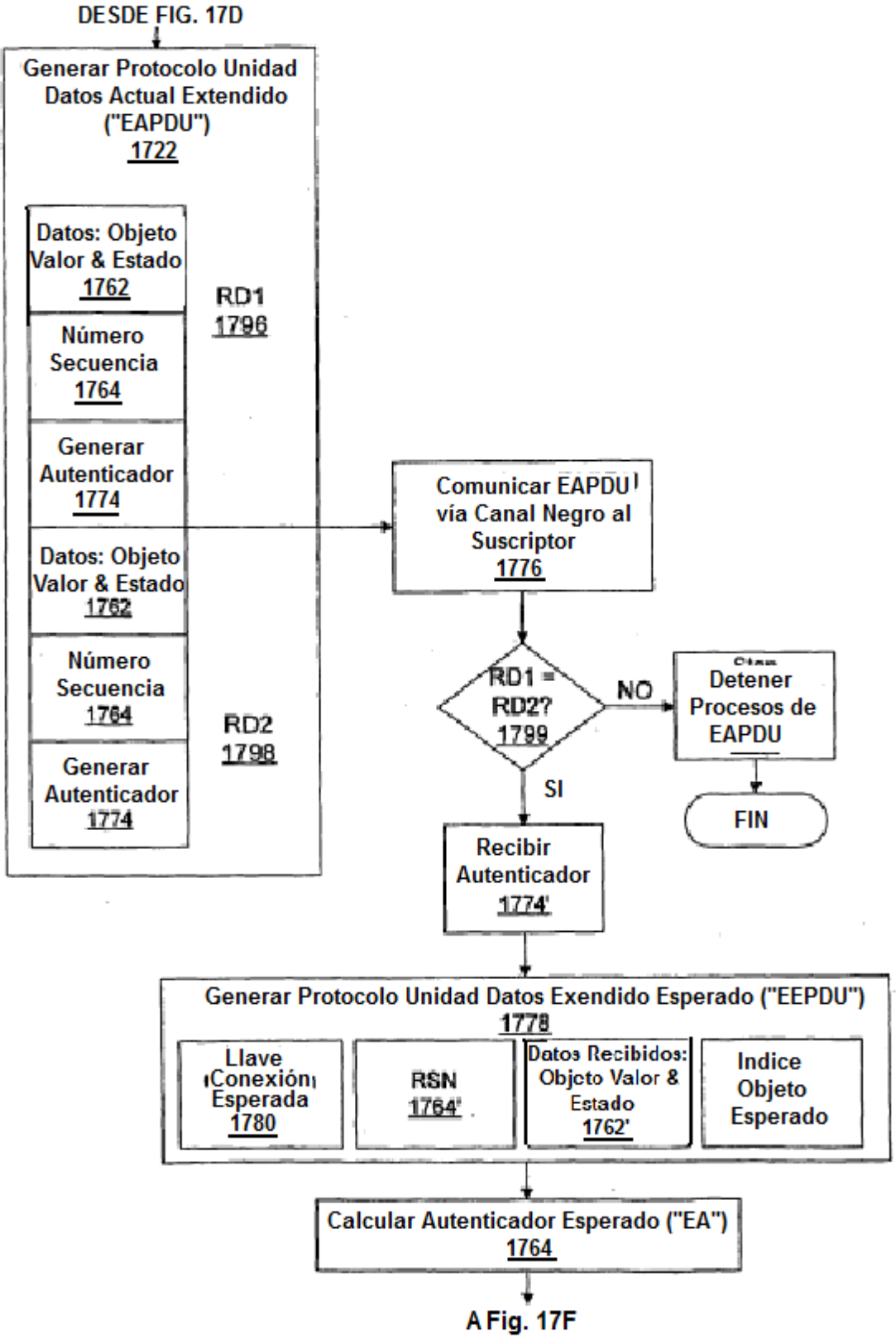

INICIO
1750
RECIBIR SOLICITUD PARA "LEER" PARÁMETRO
1752
BLOQUE RELACIONADO SEGURIDAD?
1754
SI
NO
EJECUTAR SOLICITUD LECTURA USANDO PROCESOS ESTÁNDAR DE LECTURA
1756
FIN
OBTENCION INFORMACIÓN
1758
ESPECIFICAR CONECCIÓN
1760
OBTENER DATOS
1762
OBTENER SIGUIENTE NÚMERO SECUENCIA
1764
OBTENER LLAVE CONNECCIÓN
1766
OBTENER SUB-ÍNDICE
1767
OBTENER OBJETO ÍNDICE
1768
GENERAR PROTOCOLO UNIDAD DATOS VIRTUAL ("VPDU")
1770
LLAVE CONEXIÓN
1766
NÚMERO SECUENCIA
1764
OBJETO INDICE
1768
SUB-INDICE
1767
DATOS: OBJETO VALOR & ESTADO
1762
GENERAR AUTENTICADOR (GA)
1771
A FIG. 17E

FIG. 17D


DESDE FIG. 17D
Generar Protocolo Unidad Datos Actual Extendido ("EAPDU")
1722
Datos: Objeto Valor & Estado
1762
Número Secuencia
1764
Generar Autenticador
1774
RD1
1796
Datos: Objeto Valor & Estado
1762
Número Secuencia
1764
Generar Autenticador
1774
RD2
1798
Comunicar EAPDU vía Canal Negro al Suscriptor
1776
RD1 = RD2?
1799
NO
Detener Procesos de EAPDU
FIN
SI
Recibir Autenticador
1774'
Generar Protocolo Unidad Datos Exendido Esperado ("EEPDU")
1778
Llave Conexión Esperada
1780
RSN
1764'
Datos Recibidos: Objeto Valor & Estado
1762'
Indice Objeto Esperado
Calcular Autenticador Esperado ("EA")
1764
A Fig. 17F

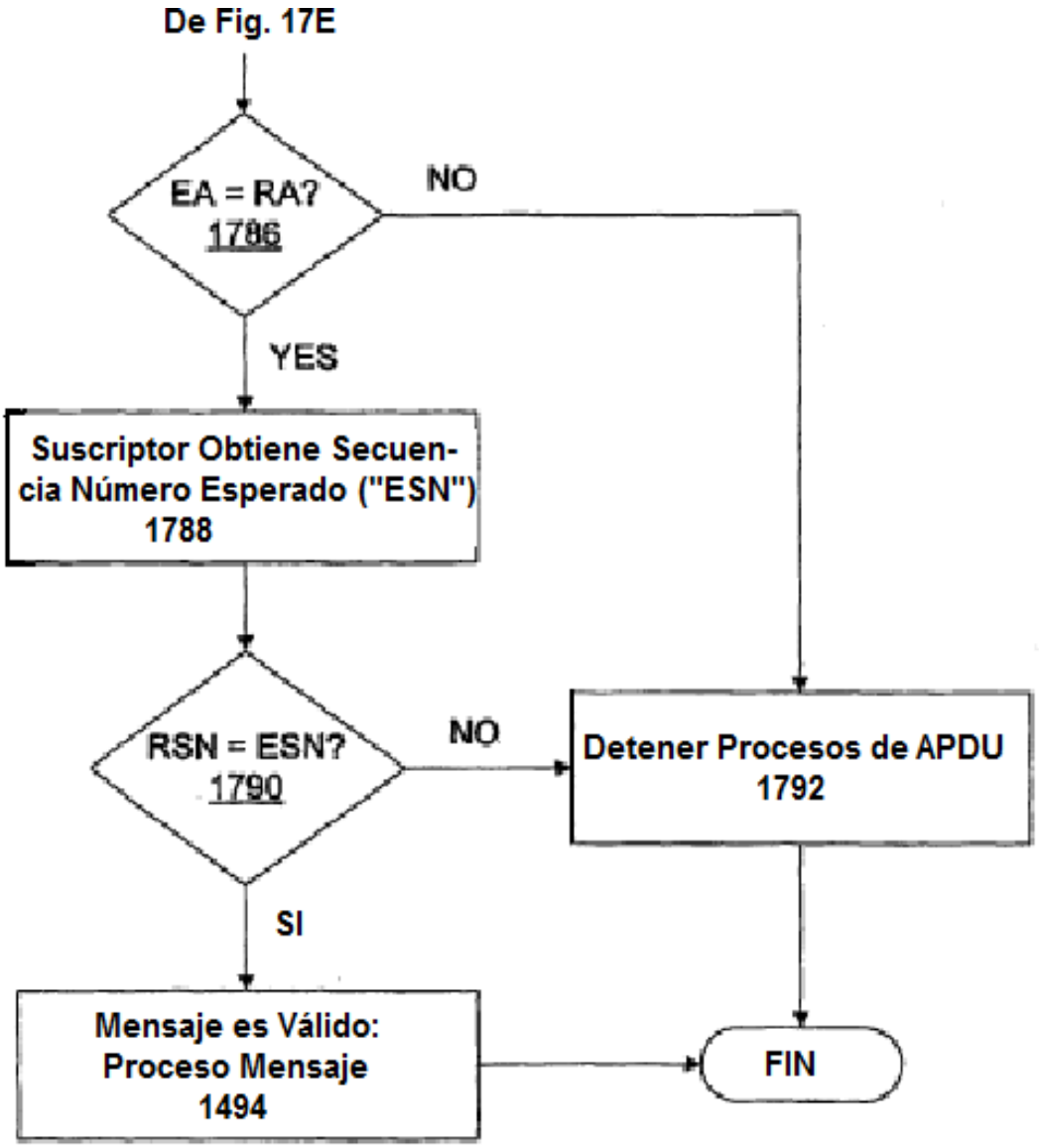

De Fig. 17E
EA = RA?
1786
NO
YES
Suscriptor Obtiene Secuencia Número Esperado ("ESN")
1788
RSN = ESN?
1790
NO
Detener Procesos de APDU
1792
SI
Mensaje es Válido:
Proceso Mensaje
1494
FIN

FIG. 17F