



19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA

11 Número de publicación: **2 337 163**

51 Int. Cl.:
H04L 12/56 (2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Número de solicitud europea: **05760912 .5**

96 Fecha de presentación : **29.06.2005**

97 Número de publicación de la solicitud: **1776807**

97 Fecha de publicación de la solicitud: **25.04.2007**

54 Título: **Método y dispositivo para acceder a los datos de una memoria de mensajes de un módulo de comunicación.**

30 Prioridad: **05.08.2004 DE 10 2004 038 213**

45 Fecha de publicación de la mención BOPI:
21.04.2010

45 Fecha de la publicación del folleto de la patente:
21.04.2010

73 Titular/es: **ROBERT BOSCH GmbH**
Postfach 30 02 20
70442 Stuttgart, DE

72 Inventor/es: **Hartwich, Florian;**
Horst, Christian y
Bailer, Franz

74 Agente: **Carvajal y Urquijo, Isabel**

ES 2 337 163 T3

Aviso: En el plazo de nueve meses a contar desde la fecha de publicación en el Boletín europeo de patentes, de la mención de concesión de la patente europea, cualquier persona podrá oponerse ante la Oficina Europea de Patentes a la patente concedida. La oposición deberá formularse por escrito y estar motivada; sólo se considerará como formulada una vez que se haya realizado el pago de la tasa de oposición (art. 99.1 del Convenio sobre concesión de Patentes Europeas).

DESCRIPCIÓN

Método y dispositivo para acceder a los datos de una memoria de mensajes de un módulo de comunicación.

5

Estado actual de la técnica

La presente invención hace referencia a un método y un dispositivo para acceder a los datos de una memoria de mensajes de un módulo de comunicación a través de la entrada y la salida de datos a o desde la memoria de mensajes de un módulo de comunicación conforme a la reivindicación independiente.

La conexión en red de dispositivos de control, sensores y actuadores con ayuda de un sistema de comunicación y de un sistema ómnibus, por consiguiente, de un enlace de comunicación, ha aumentado en forma drástica en los últimos años, tanto en la construcción de vehículos modernos como también en la construcción de máquinas, especialmente en el sector de las máquinas-herramienta, y en la automatización. A través de la distribución de funciones en varios dispositivos de control pueden obtenerse efectos sinérgicos. Se habla así de sistemas distribuidos. La comunicación entre diferentes estaciones tiene lugar cada vez más frecuentemente mediante un sistema ómnibus, o mediante un enlace de comunicación. La circulación de la información en el sistema ómnibus, los mecanismos de entrada y salida, así como también el control de errores están regidos por un protocolo. Dentro de este área es conocido el protocolo CAN, así como también los protocolos TTCAN y FlexRay que sirven de base actualmente a la especificación del protocolo FlexRay. Flexray es un sistema ómnibus rápido, determinista y tolerante a fallos, en especial para la utilización en un vehículo. El protocolo FlexRay trabaja conforme al acceso múltiple por división de tiempo (TDMA), en el cual los componentes -las partes- de los mensajes a transmitir son adjudicados en ranuras de tiempo, donde éstas tienen una entrada exclusiva al enlace de comunicación. Un traslado similar tiene lugar en TTCAN. Las ranuras de tiempo se repiten en un ciclo determinado, de manera que el instante en el que se transmite el mensaje mediante el bus, puede predecirse con exactitud, y el acceso al bus se produce determinísticamente. Para aprovechar óptimamente el ancho de banda para la transmisión del mensaje en el sistema ómnibus, FlexRay subdivide el ciclo en una parte dinámica y una parte estática. Las ranuras de tiempo determinadas se encuentran así en la parte estática al comienzo de un ciclo de bus. En la parte dinámica las ranuras de tiempo se adjudican dinámicamente. De esta forma se posibilita entonces el exclusivo acceso al bus cada vez sólo por un corto tiempo, conocido como minislots (minirranuras de tiempo). Sólo cuando un acceso al bus tiene lugar dentro de una minirranura, se alarga el intervalo durante el tiempo necesario. De esta manera, el ancho de banda se utiliza sólo cuando efectivamente se le necesita. De este modo FlexRay comunica mediante dos líneas separadas con una velocidad máxima de 10 MB. Ambos canales corresponden a la capa física, en especial al modelo de capas del OSI (Open System Architecture). Estos canales sirven efectivamente para lograr una transmisión de mensajes redundante y tolerante a fallos, por lo cual se duplicaría entonces la velocidad. FlexRay puede, sin embargo, ser activado también con velocidades más bajas. Para realizar funciones sincrónicas y optimizar el ancho de banda mediante pequeñas distancias entre los mensajes, los componentes distribuidos en la red de comunicación FlexRay, los participantes, requieren de una base de tiempo común, conocida como tiempo global.

Para lograr la sincronización de tiempos se transmiten colas de mensajes a la parte estática del ciclo, de manera tal que con la ayuda de un algoritmo especial correspondiente a la especificación FlexRay, el tiempo local de un componente es corregido, logrando así que todos los relojes marchen sincrónicamente en una secuencia de tiempo global. Esta sincronización es similar en la red de comunicación TTCAN.

Un nodo de red FlexRay o un participante FlexRay o procesador central contiene un procesador participante o procesador central, un controlador FlexRay o controlador de comunicación así como una control del bus contiene un controlador del bus. Con ello el procesador central o procesador participante envía y procesa los datos transmitidos por el controlador de comunicación FlexRay. Para la comunicación en una red FlexRay pueden configurarse mensajes u objetos de mensajes, con por ejemplo, hasta 254 bytes de datos. Para transmitir estos mensajes u objetos de mensajes entre la capa física, por tanto entre el enlace de comunicación y el procesador central, se emplea un módulo de comunicación, en particular un controlador de comunicación.

Los accesos a la memoria de mensajes de un módulo de comunicación, en especial de un controlador de comunicación FlexRay se producen tanto a través de la unidad del procesador central, por tanto del procesador CPU, como también mediante módulos de interfaces en la capa física. El acceso al procesador CPU comprende así la escritura y la lectura de datos de configuración, de datos de estado y de datos que propiamente se transmiten. El acceso del procesador CPU a la capa física comprende la lectura de mensajes enviados, el almacenamiento de mensajes recibidos, asegurando en ambos casos la integridad de los mensajes mediante medidas adecuadas.

La US 2003/0174708 A1 presenta una memoria de alta velocidad con una estructura modular. En ella se prevee un almacenamiento temporal en particular de mensajes, para garantizar por completo su recepción. En condiciones atípicas es posible además que el espacio previsto no sea suficiente.

Es objeto de la presente invención el optimizar la transmisión de datos entre el procesador CPU o procesador participante y la memoria de mensajes con respecto a la velocidad de transmisión y a la integridad de los datos.

Ventajas de la invención

Para alcanzar este objeto la invención se basa en un método y en un dispositivo para acceder a datos de una memoria de mensajes, mediante la entrada y la salida de datos en o de la memoria de mensajes, con lo cual la memoria se conecta a una disposición de búfers (espacios de memoria).

La disposición de búfers comprende un primer búfer, el cual en forma conveniente se encuentra dividido en una sub memoria búfer y una memoria espejo para la sub memoria búfer, con lo cual se ejecutan los siguientes pasos: Entrada de datos en la memoria del búfer correspondiente e intercambio del acceso en la sub memoria búfer y la memoria espejo, de manera que los siguientes datos puedan ser ingresados en la memoria espejo, mientras que los datos ingresados ya son devueltos de la sub memoria búfer en la dirección de transmisión prevista.

Esto significa que, de manera ventajosa, se prevee un búfer para el acceso a datos de una memoria de mensajes, el cual contiene una sub memoria búfer y una memoria espejo, así como también los medios a través de los cuales se intercambia el acceso a la sub memoria búfer y a la memoria espejo. La invención presentada es ventajosa, ya que mediante este intercambio, la velocidad de transmisión puede aumentar de forma evidente y mediante la bipartición del búfer es posible una mutua lectura y escritura, de manera que durante la transferencia de datos se suprime un acceso simultáneo a celdas de memoria iguales, asegurando así la integridad de los datos.

En una presentación ventajosa de la invención se instala, para acceder a los datos de una memoria de mensajes de un módulo de comunicación, un búfer con una memoria de entrada y una memoria de salida, con lo cual la memoria de entrada contiene una primera sub memoria búfer y una primera memoria espejo y la memoria de salida contiene una segunda sub memoria búfer y una segunda memoria espejo. A su vez se prevee un primer medio para intercambiar el acceso de la primera sub memoria búfer y la primera memoria espejo, y un segundo medio para para intercambiar el acceso de la segunda sub memoria búfer y la segunda memoria espejo.

Esto significa que la invención describe un dispositivo para la transferencia de datos entre un procesador CPU y una memoria de mensajes a través del intercalado de un búfer, en particular de una memoria de entrada para la escritura y una memoria de salida para la lectura para desacoplar los accesos de las unidades de transmisión y de recepción. De esta forma el administrador o gestor del mensaje dirige y controla todos los accesos relativos al mensaje, asegurando así la integridad de los mensajes almacenados.

De manera aún más ventajosa se indica el intercambio de la sub memoria búfer y la memoria espejo a través de un identificador o se produce el intercambio a través del identificador en caso de una ejecución especial.

Convenientemente, los datos son transmitidos en mensajes y los mensajes son provistos de un identificador de mensajes, gracias al que tiene lugar el intercambio de acceso de la sub memoria búfer y la memoria espejo, cuando el identificador de mensaje es inscrito en un registro prefijado.

Es particularmente ventajoso el que los datos en un mensaje se encuentren como mínimo divididos en primeros datos o datos de configuración, datos de estado y segundos datos o los datos propiamente a transmitir, con lo cual son transmitidos, dependiendo de al menos un identificador de datos, o sólo los primeros datos o sólo los segundos datos o los primeros y los segundos datos.

Convenientemente, el acceso a los datos puede indicarse mediante un identificador de acceso, el cual se inscribe en un registro. De manera ventajosa, se transmite una información desde el indicador de acceso mismo o desde su posición, con respecto a qué sub memoria búfer o memoria espejo se tiene acceso en ese momento preciso.

De manera conveniente se sustituyen en el registro, durante el intercambio y al menos parcialmente, las inscripciones de los registros, en especial con respecto a los mencionados identificadores.

Para realizar el intercambio de acceso de la sub memoria búfer y la memoria espejo puede, convenientemente, reverse un identificador conmutable.

Además se puede, convenientemente, inscribir en un registro un identificador de inicio para al menos una dirección de transmisión, cuando los datos deben ser inscritos en una memoria.

Gracias a la presente invención, mediante la disposición de búferes y de las memorias de entrada y salida que posee, así como también del intercambio de acceso, se asegura que el procesador CPU pueda acceder sin interrupciones a la memoria de mensajes pese a los períodos de latencia internos del módulo. A través del aprovechamiento óptimo de la memoria y de la ausencia de interrupciones, aumenta la velocidad de transmisión, así como también se garantiza al mismo tiempo la integridad de los datos mediante la presente invención.

Otras ventajas y ejecuciones ventajosas resultan de las características de las reivindicaciones, así como de la descripción de ejemplos de ejecución.

Dibujos

La presente invención se explica en detalle mediante las siguientes figuras de los dibujos.

5 La figura 1 muestra, mediante una representación esquemática, el módulo de comunicación y su conexión a la capa física, por tanto, el enlace de comunicación y el participante del procesador o participante de la comunicación.

La figura 2 presenta en detalle, en una forma de ejecución especial, el módulo de comunicación de la figura 1, así como su conexión.

10 En la figura 3 se representa la estructura de la memoria.

Las figuras 4 a 6 describen esquemáticamente la arquitectura y el proceso del acceso de datos en dirección de los participantes hacia la memoria.

15 Las figuras 7 a 9 describen esquemáticamente la arquitectura y el proceso del acceso de datos en dirección de la memoria hacia los participantes.

20 En la figura 10 se representan esquemáticamente el gestor del mensaje y las allí contenidas máquinas de estado finito.

La figura 11 muestra nuevamente en forma esquemática los componentes del módulo de comunicación, así como los participantes y las correspondientes rutas de datos, dirigidas por el gestor de mensajes.

25 La figura 12 describe la división de accesos con respecto a las rutas de datos de la figura 11.

La invención se explica en detalle mediante los siguientes ejemplos de ejecución.

Ejemplos de ejecución

30 La figura 1 muestra esquemáticamente un módulo de comunicación 100 para la conexión de un participante o procesador 102 a un enlace de comunicación FlexRay 101, por tanto a la capa física de FlexRay. Para esto, el enlace de comunicación FlexRay 100 se encuentra conectado, mediante una conexión 107, con el participante o procesador participante 102 y mediante una conexión 106, al enlace de comunicación 101. Para lograr una conexión sin problemas con respecto a los tiempos de transmisión y a la integridad de los datos, se diferencian en forma esquemática esencialmente tres disposiciones en el módulo de comunicación FlexRay. De este modo, una primera disposición 105 se utiliza para la memoria, en especial como portapapeles, al menos una parte de los mensajes a transmitir. Entre el participante 102 y esta primera disposición 105 se encuentra conectada una segunda conexión 104 mediante las conexiones 107 y 108. Por tanto, se encuentra, entre el participante 101 y la primera disposición 105 una tercera disposición 103, enlazada mediante las conexiones 106 y 109, por el cual se puede obtener una entrada y salida de datos muy flexible como parte de los mensajes, en particular de mensajes FlexRay a o desde la primera disposición 105, garantizando la integridad de los datos con una velocidad óptima.

45 En la figura 2 se presenta detalladamente este módulo de comunicación 100 en una forma de ejecución preferente. Igualmente detalladas se encuentran las conexiones correspondientes 106 a 109. La segunda disposición 104 contiene un búfer de memoria de entrada o búfer de memoria de ingreso (Input Búfer IBF) 201, un búfer de memoria de salida 202 (Output Búfer OBF), así como una interfaz compuesta de dos partes 203 y 204, siendo la parte del módulo 203 independiente del participante y la parte del módulo 204 específica del participante. La parte del módulo 204 específica del participante (Customer CPU Interface CIF) conecta un controlador CPU 102 específico del participante, un participante específico del cliente con el módulo de comunicación FlexRay. Para ello se encuentra prevista una línea de datos bidireccional 216, una línea de dirección 217, así como también una entrada de control 218. Del mismo modo se encuentra prevista una salida de corte o de interrupción 219. La parte del módulo 204 específica del participante se encuentra conectada a una parte del módulo 203 independiente del participante (Generic CPU Interface, GIF), o sea que el módulo de comunicación FlexRay o el módulo IP de FlexRay dispone de una interfaz del CPU genérica, por tanto de uso general, a la que se pueden conectar, mediante la correspondiente parte del módulo específica del participante (Customer CPU Interface CIF), una gran cantidad de diferentes participantes específicos del cliente del controlador CPU. De esta forma sólo presenta variaciones el módulo 204 dependiente del participante, lo que se traduce en un esfuerzo significativamente menor.

60 El módulo del búfer de memoria de entrada o memoria de ingreso 201 y el módulo del búfer memoria de salida 202 pueden estar conformados en un módulo de memoria o en memorias separadas. De esta manera el búfer de memoria de entrada 201 sirve para el almacenamiento de mensajes para la transmisión de la memoria de mensajes 200. Para ello el módulo de la memoria de entrada se encuentra conformado, preferentemente, de manera tal que pueda almacenar dos mensajes completos compuestos respectivamente por un segmento de encabezado o segmento Header, en especial con datos de configuración, y un segmento de datos o segmento Payload (segmento de carga útil de datos de información datos y no de control). Para ello el módulo de la memoria de entrada consta de dos partes (sub memoria búfer y memoria espejo), a través de las cuales, mediante una escritura alternante de ambas partes del búfer de memoria de entrada, así como a través del cambio de acceso, se puede acelerar la transmisión entre el procesador

ES 2 337 163 T3

CPU participante 102 y la memoria de mensajes 200. Del mismo modo sirve el búfer de memoria de salida o búfer de memoria de devolución (Output Búfer OBF) para el almacenamiento de mensajes en la transmisión de la memoria de mensajes 200 al procesador CPU participante 102. Para ello el búfer de memoria de salida 202 se encuentra a su vez conformado de manera tal, que puedan ser almacenados dos mensajes completos compuestos por un segmento de encabezado, en especial con datos de configuración y un segmento de datos o segmento Payload. También aquí el búfer de memoria de salida 202 se encuentra dividido en dos partes, una sub memoria búfer y una memoria espejo, a través de las cuales, mediante una escritura alternada de ambas partes, así como a través del cambio de acceso, se puede acelerar la transmisión entre el procesador CPU participante 102 y la memoria de mensajes 200. Esta segunda disposición 104, compuesta por los bloques 201 a 204, se encuentra, tal como se representa, conectada a la primera disposición 105.

La disposición 105 está compuesta por un gestor de mensajes 200 (Message Handler MHD) y una memoria de mensajes 300 (Message RAM). El gestor de mensajes controla, por tanto dirige, la transferencia de datos entre el búfer de memoria de entrada 201, así como también del búfer de memoria de salida 202 y la memoria de mensajes 300. En igual medida controla, por tanto dirige la transferencia de datos en la dirección inversa a través de la disposición 103. La memoria se encuentra, preferentemente, constituida como single-ported RAM (RAM de acceso sencillo, no simultáneo de escritura y lectura). Esta memoria RAM almacena los mensajes u objetos del mensaje, o sea los mensajes propiamente dichos, conjuntamente con datos de configuración y datos de estado. La estructura exacta de la memoria 300 es representada con detalle en la figura 3.

La tercera disposición 103 está compuesta por los bloques 205 a 208. En correspondencia con los dos canales de la capa física de FlexRay, esta disposición 103 se encuentra dividida en dos rutas de datos con dos direcciones de datos en cada una. Esto se pone de manifiesto claramente mediante las conexiones 213 y 214, en las cuales se presentan ambas direcciones para el canal A, RxA y TxA para recepción (RxA) y transmisión (TxA), así como para el canal B, RxB y TxB. La conexión 215 se refiere a una óptima entrada de control bidireccional. El enlace de la tercera disposición 103 se produce mediante una primera memoria búfer 205 para el canal B y una segunda memoria búfer 206 para el canal A. Ambas memorias búfer (búfer no residente RAM's: RAM A y RAM B) sirven como memorias intermedias para la transmisión de datos desde y hacia la primera disposición 105. En correspondencia con los dos canales, estas dos memorias búfer 205 y 206 se encuentran conectadas respectivamente con interfaces del módulo 207 y 208, las cuales contienen el controlador del protocolo FlexRay o controlador protocolo bus, el cual consta de un registro de desplazamiento de recepción y transmisión y del protocolo FlexRay para máquinas de estado finito. Ambas memorias búfer 205 y 206 sirven como memorias intermedias para la transmisión de datos entre los registros de desplazamiento de la interfaz del módulo o del controlador del protocolo FlexRay 207 y 208 y la memoria de mensajes 300. También aquí se almacenan de manera ventajosa, a través de cada memoria búfer 205 o 206, los campos de datos, por tanto el segmento Payload o el segmento de datos de información de carga útil de dos mensajes FlexRay.

Posteriormente representada como 209, se encuentra, en el módulo de comunicación 100, la unidad de tiempo global (Global Time Unit GTU), la cual es competente para la representación del intervalo de tiempo global en FlexRay, por tanto de los microciclos PT y de los macrociclos MT. Del mismo modo se encuentra reglamentada, mediante la unidad de tiempo global 209, la sincronización de tiempo tolerante a fallos del contador de ciclos (Cycle Counter) y el control del ritmo de ciclos en el segmento estático y dinámico de FlexRay.

Como bloque 210 se encuentra representado el sistema de control universal (System Universal Control SUC), a través del cual se controlan y dirigen los modos de operación del control de comunicación FlexRay. Entre ellos se cuentan el wakeup (la activación), el startup (arranque), la reintegración o integración, la operación normal (normal operation) y la Operación pasiva (passive operation).

El bloque 211 muestra la red y la gestión de fallos (Network- und Error Management NEM), tal como se describe en el protocolo de especificación v2.0 FlexRay. Por último, el bloque 212 muestra el control de interrupción (Interrupt Control INT), el cual gestiona las marcas de la interrupción con respecto a fallos y al estatus (status and error interrupt flags) y controla, por tanto dirige, las salidas de interrupción 219 hacia el CPU participante 102. El bloque 212 contiene además un contador o temporizador para generar las interrupciones de tiempo o las interrupciones del contador.

Para la comunicación dentro de una red FlexRay pueden configurarse objetos de mensaje, o sea mensajes de búfer (Message Búfer) de hasta 254 bytes de datos. El búfer de memoria de mensajes 300 es especialmente una memoria RAM de mensajes (Message RAM) que puede almacenar, por ejemplo, hasta un máximo de 64 objetos de mensaje. Todas las funciones que atañen al tratamiento, así como a la gestión de los mensajes en sí mismos, se implementan en el gestor de mensajes o administrador de mensajes. Éstas son, por ejemplo, el filtrado de aceptación, la transferencia de mensajes entre ambos bloques del controlador del protocolo FlexRay 207 y 208 y el búfer de memoria de mensajes 300, por tanto de la memoria RAM, así como el control de la secuencia de transmisión y la provisión de datos de configuración, por tanto de datos de estado.

Una CPU externo, o sea un procesador externo del participante 102, puede acceder en forma directa, mediante la interfaz del participante, con la parte específica del participante 204, al registro del módulo de comunicación FlexRay. De esta manera se utiliza una gran cantidad de registros. Estos registros se constituyen para configurar y dirigir el controlador del protocolo FlexRay, por tanto las interfaces del módulo 207 y 208 del gestor de mensajes (administrador de mensajes MHD) 200, la unidad de tiempo global 209 (Global Time Unit GTU), el sistema de control universal (System Universal Control SUC) 210, la red y la gestión de fallos (Network and Error Management NEM) 211, el

ES 2 337 163 T3

control de interrupción (Interrupt Control INT) 212, así como el acceso a la RAM de mensajes, por tanto al búfer de memoria de mensajes 300 y de este modo consignar el correspondiente estado. Partes de este registro, como mínimo, se abordan en detalle en las figuras 4 a 6 y 7 a 9. Un módulo de configuración FlexRay como el descrito, conforme a la invención, posibilita la conversión simple del protocolo de especificación v2.0 FlexRay, mediante el cual fácilmente se puede generar un ASIC (Circuito Integrado de Aplicaciones Específicas) o un microcontrolador con la funcionalidad FlexRay correspondiente integrada.

En la figura 3 se describe detalladamente la división de la memoria de mensajes. Para lograr la funcionalidad exigida de un controlador de comunicación FlexRay se precisa una memoria de mensajes para la relocalización de los búfer de mensajes a transmitir (Transmit búfer) así como el almacenamiento de mensajes recibidos sin fallos (Receive Búfer). Un protocolo FlexRay permite mensajes con un área de datos, por tanto de un área Payload de 0 a 254 bytes. De esta forma, en la figura 2 se representa la memoria de mensajes como parte del módulo de comunicación FlexRay 100. El método que se describe a continuación, así como la correspondiente memoria, describen el almacenamiento de mensajes a transmitir así como de mensajes recibidos, especialmente mediante el uso de una Random Access Memory (RAM), con lo cual es posible, a través del mecanismo conforme a la invención, almacenar una cantidad variable de mensajes en una memoria de mensajes de tamaño previsto. Al mismo tiempo, la cantidad de mensajes almacenables depende del tamaño de las áreas de datos de cada mensaje, por lo cual en primer lugar se minimiza el tamaño precisado para el almacenamiento sin reducir el tamaño de las áreas de datos y en segundo lugar se aprovecha la memoria en forma óptima. A continuación se describe en detalle esta división variable de una memoria de mensajes basada en RAM en especial para un controlador de comunicación FlexRay.

Para la implementación se tomará, a modo de ejemplo, una memoria de mensajes con una longitud de palabra predeterminada de n bits, por ejemplo 8, 16, 32, etc; así como una profundidad de almacenamiento de palabras m (m , n equivalen a números naturales). De esta manera se divide el búfer de memoria de mensajes 300 en dos segmentos, un segmento Header (de cabecera) o segmento de inicio HS y un segmento de datos DT (Payload Section, Payload Segment). Por mensaje se delinea un área inicial HB y un área de datos DB. Para los mensajes de $0,1$ bits k (k como número natural), se delimitan segmentos Header (de cabecera) o segmentos de inicio HBO, HB1 a HBk y áreas de datos DBO, DB1 a DBk. Dentro de un mensaje, se diferencia así, entre datos primarios y datos secundarios, con lo cual los datos primarios, datos de configuración o datos de estado se corresponden con respecto al mensaje FlexRay y se clasifican respectivamente en un área de inicio HB (HB0, HB1, ..., HBk). Los datos secundarios, los cuales equivalen a los datos propiamente dichos que deben ser transmitidos, se clasifican respectivamente en un área de inicio DB (DB0, DB1, ..., DBk). De este modo surge, para los datos primarios por cada mensaje, una primera amplitud de datos (medida en bits, bytes o por palabras almacenadas) y, para los datos secundarios de un mensaje, una segunda amplitud de datos (también medida en bits, bytes o por palabras almacenadas), en la cual la segunda amplitud de datos por mensaje puede ser diferente. La división entre el segmento de inicio HS y el segmento de datos DS es variable en la memoria de mensajes 300, esto significa, que no existe ningún límite previsto entre las áreas. La división entre el segmento de inicio HS y el segmento de datos DS es, conforme a la invención, dependiente de la cantidad k de mensajes, así como también de la segunda amplitud de datos, por tanto de la amplitud de los datos propiamente dichos, de un mensaje, por tanto de todos los mensajes k en conjunto. Conforme a la invención, a los datos de configuración KDO, KD1 hasta KDK del respectivo mensaje, se le adjunta en forma directa un elemento indicador o puntero de datos DPO, DP1 hasta DPK. En la conformación especial se le adjunta a cada área de inicio de HB0, HB1 a HBk una cantidad fija de palabras almacenadas, en este caso dos, de manera que siempre un dato de configuración KD (KDO, KD1, ..., KDK) y un elemento indicador DP (DPO, DP1, ..., DPK) se clasifican conjuntamente en un área de inicio HB. A este segmento de inicio HS con él área de inicio HB, cuyo tamaño, por tanto amplitud de datos, depende de la cantidad k de mensajes a almacenar, se conecta un segmento de datos DS para el almacenamiento de los mensajes propiamente dichos D0, D1 a Dk. Este segmento de datos (o sección de datos) DS depende de los datos de mensaje clasificados, en este caso, por ejemplo, en DBO seis palabras, en DB1 una palabra, y en DBk dos palabras. Los respectivos elementos indicadores DPO, DP1 a DPK muestran con esto, siempre al comienzo, o sea, en la dirección inicial del área de datos correspondiente DB0, DB1 a DBk, en los cuales los datos D0, D1 a Dk del respectivo mensaje se clasifican de 0, 1 a k . Por esto la división de la memoria de mensajes entre el segmento de inicio HS y el segmento de datos DS es variable y depende de la cantidad de mensajes en sí mismos, así como de la respectiva amplitud de los datos de un mensaje, por tanto de la amplitud total de la segunda amplitud de datos. Si se configuran pocos mensajes, se disminuye el segmento inicial y al área que se libera en la memoria de mensajes puede ser utilizada como suplemento del segmento de datos DS para el almacenamiento de datos. Mediante esta variabilidad se puede garantizar un óptimo aprovechamiento, con lo cual también es posible la utilización de memorias más pequeñas. El segmento de datos FDS, en especial su tamaño, igualmente dependiente de la cantidad k de mensajes almacenados y de la respectiva amplitud de datos de los mensajes es, por tanto, mínimo y puede incluso ser igual a 0.

Junto a la utilización de elementos indicadores es también posible clasificar los datos primarios y secundarios, por tanto los datos de configuración KD (KDO, KD1, ..., KDK) y los datos propiamente dichos D (D=, D1, ..., Dk) en una secuencia prevista, de manera tal que la secuencia del área de inicio HBO hasta HBk en el segmento inicial HS y la secuencia del área de datos DBO hasta DBk en el segmento de datos DS respectivamente, son idénticas. Así se podría incluso, eventualmente, renunciar a un elemento indicador.

En una conformación especial, se le adjunta a la memoria de mensajes un generador corrector, en especial un elemento generador de bits de paridad y un verificador corrector, en especial un elemento verificador de bits de paridad, para garantizar la corrección de los datos almacenados en HS y DS, pudiendo clasificarse, por palabra almacenada o por área (HB o DB), por una suma de control, en especial como una verificación de bits de paridad. También se

ES 2 337 163 T3

puede pensar en otros indicadores de control, por ejemplo, en CRC (Cyclic redundancy check -control de redundancia cíclica-) o en indicadores de una mayor potencia, como ECC (Error Code Correction -código corrector de errores-). Comparada con una división fija de la memoria de mensajes, se presentan las siguientes ventajas:

- 5 el usuario puede decidir durante la programación si desea utilizar una gran cantidad de mensajes con un campo de datos pequeño o una pequeña cantidad de mensajes con un gran campo de datos. Durante la configuración de mensajes con áreas de datos de diferentes tamaños se aprovecha el espacio de memoria presente de forma óptima. El usuario tiene la posibilidad de utilizar el área de almacenamiento de datos en común para diferentes mensajes.
- 10 Durante la implementación del controlador de comunicación a una conexión integrada se puede adecuar el tamaño de la memoria de mensajes, mediante una adaptación de la profundidad de memoria de la memoria utilizada, a las necesidades de la aplicación, sin modificar otras funciones del comunicador de control.

15 A continuación se describe en detalle, mediante las figuras 4 a 6 y 7 a 9 el acceso del procesador CPU, por tanto de la escritura y la lectura de datos de configuración, o sea, de datos de estado y de los datos propiamente dichos, a través de la colocación de la memoria búfer 201 y 202. Por este medio se intenta lograr un desacoplamiento con respecto a la transmisión de datos, que pueda asegurar la integridad de los datos y al mismo tiempo, que garantice una alta velocidad de transmisión. El control de estas operaciones tiene lugar mediante el gestor de mensajes 200, el cual será descrito más adelante en las figuras 10, 11 y 12.

20 En las figuras 4, 5 y 6 se explican en detalle, en primer lugar, los accesos de escritura de la memoria de mensajes 300 a través del procesador CPU del participante 102 mediante la memoria de acceso 201. La figura 4 muestra además nuevamente el módulo de comunicación 100, donde, para mayor claridad, sólo se muestran las partes aquí relevantes del módulo de comunicación 100. Este módulo es, en primer lugar, el gestor de mensajes 200, responsable del control de los resultados, así como dos registros de control 403 y 404, los cuales pueden ser colocados, tal como se representa, 25 por fuera del gestor de mensajes 200 en el módulo de comunicación 100, pero que también pueden estar contenidos dentro del gestor de mensajes 200. 403 representa el registro de peticiones de entrada (Input Búfer Command Request Register) y 404 el registro de enmascaramiento de entrada (Input Búfer Command Mask Register). Los accesos de escritura del procesador CPU 102 se producen en la memoria de mensajes 300 por medio de una memoria de acceso 30 interconectada 201 (Input Búfer). Esta memoria de acceso interconectada 201 se exhibe dividida, por tanto duplicada, a saber en una sub memoria búfer 400 y una memoria espejo 401 perteneciente a la sub memoria búfer. Con esto, puede tener lugar, como se describe a continuación, un acceso continuo del controlador CPU 102 a los mensajes, por tanto a los objetos del mensaje y a los respectivos datos de la memoria de mensajes, y puede asegurarse la integridad de los datos y una transmisión rápida. El control de los accesos se produce mediante el registro de peticiones de entrada 35 403 y el registro de enmascaramiento de entrada 404. En el registro 403 se representan con los números 0 a 31, aquí a modo de ejemplo, los respectivos espacios bit en 403 para un ancho de 32 bits. Esto mismo es válido para el registro 404 y los espacios bit 0 a 31.

40 Conforme a la invención y a modo de ejemplo, los espacios bit 0 a 5, 15, 16 a 21 y 31 del registro 403 mantienen, con respecto al control de desarrollo, una función especial. De esta manera se puede inscribir en los espacios bits 0 a 5 del registro 403 un identificador IBRH (Input Búfer Request Host) como identificador de mensaje. De la misma manera, se puede inscribir un identificador IBRS (Input Búfer Request Shadow) en los espacios bit del registro 403. Del mismo modo se pueden inscribir en el espacio de registro 15 de 403, IBSYH y en el espacio de registro 31 de 403, IBSYS como identificadores de acceso. Los espacios 0 a 2 del registro 404 se encuentran también marcados, 45 donde en 0 y 1, junto con LSHS (Load Header Section Host -segmento de cabecera de la carga útil de datos-) y LDSH (Load Data Section Host -segmento de la carga útil de datos-) se inscriben otros identificadores como identificadores de datos. Estos identificadores de datos se encuentran aquí conformados de la manera más simple, o sea conformados como un bit. En el espacio bit 2 del registro 404 se inscribe un identificador inicial con STXRH (Set Transmission X Request Host -configuración de transmisión a petición del sistema principal-). A continuación se describe el desarrollo 50 del acceso de escritura a la memoria de mensajes mediante la memoria de acceso.

El procesador CPU 102 escribe los datos de los mensajes a transferir en el búfer de memoria de acceso 201. De este modo, el procesador CPU 102 puede escribir sólo los datos de configuración e inicio KD de un mensaje para el segmento inicial HS de la memoria de mensajes o sólo los datos propiamente dichos, los datos a transmitir de un 55 mensaje D para el segmento de datos DS de la memoria de mensajes o escribir ambos. Qué parte de un mensaje, por tanto qué datos de configuración deben ser transmitidos, se determina por medio de identificadores de datos LSHS y LDSH en el registro de enmascaramiento de entrada 404. Así se determina, mediante LSHS, si los datos de inicio, por tanto las configuraciones de datos KD, deben ser transmitidas o si los datos D deben ser los transmitidos. Debido a que la memoria de acceso 201 se encuentra conformada por dos partes, o sea, por una parte de la memoria búfer 60 400 y una memoria espejo 401 perteneciente a la memoria búfer 400 y a que debe producirse un acceso alternado, se prevén, como opuestos a LSHS y LDSH, otros dos identificadores de área, relativos a la memoria espejo 401. Estos identificadores de área en los espacios bit 16 y 17 del registro 404 se caracterizan como LHSS (Load Header Section Shadow -espejado del segmento de cabecera de la carga útil de datos-) y LDSS (Load Data Section Shadow -espejado del segmento de la carga útil de datos-). A través de estos se dirige el proceso de transmisión con respecto a la memoria espejo 401. 65

Si se dispone del bit de inicio, por tanto el identificador de inicio STX-RH en el espacio bit 2 del registro de enmascaramiento de entrada 404, se asienta automáticamente, de acuerdo a la transferencia efectuada de los respectivos

ES 2 337 163 T3

datos de configuración y/o de los datos propiamente dichos a transmitir en la memoria de mensajes 300, una petición de transmisión (Transmission Request) para el objeto de mensaje correspondiente. Esto significa, que mediante este identificador de inicio STXRH, se dirige y en especial se da inicio, a la transmisión automática de un objeto de mensaje a traducir.

5 El opuesto correspondiente aquí en la memoria espejo es el identificador de inicio STXRS (SetTransmission X Request Shadow -espejado de la configuración de transmisión a petición del sistema principal-), el cual, a modo de ejemplo, está contenido en el espacio bit 18 del registro de enmascaramiento de entrada 404 y también aquí, de la manera más simple, se constituye como un bit. La función de STXRS es análoga a la función de STXRH, sólo que es
10 respecto a la memoria espejo 1.

Cuando el procesador CPU 102 intercambia el identificador de mensajes, en especial el número de los objetos de mensaje, en la memoria 300, en la cual se deben transmitir los datos del búfer de memoria de acceso 201, en los espacios bit 0 a 5 del registro de peticiones de acceso 403, por consiguiente escritos conforme a IBRH, se intercambian
15 la sub memoria búfer 400 de la memoria de entrada 401 y la a ella perteneciente memoria espejo 401. Así como también el respectivo acceso del procesador CPU 102 y la memoria de mensajes 300 se intercambian en ambas memorias parciales 400 y 401, tal como se indica mediante la flecha semicircular. De este modo, se da inicio, por ejemplo, a la transmisión de datos hacia la memoria de mensajes 300. La transmisión de datos hacia la memoria de mensajes 300 se produce en sí misma desde la memoria espejo 401. Al mismo tiempo, se intercambian las áreas de registro IBRH y IBRS. De la misma forma se intercambian LSHS y LDSH en oposición a LHSS y LDSS. STXRH
20 con STXRS se intercambian en igual medida. IBRS indica con esto la identificación del mensaje, de este modo el número del objeto de mensaje para una transmisión. Por tanto se inicia una transferencia desde la memoria espejo. De este modo, la transferencia se refiere a cuál área dentro de una memoria de mensajes se obtiene como último dato (KD y/o D) de la memoria espejo. Mediante el identificador (1 bit nuevamente a modo de ejemplo) IBSYS (Input Búfer Busy Shadow -entrada de búfer espejado ocupado-) en el espacio bit 31 del registro de peticiones de acceso 403 se indica si en ese momento se está produciendo una transmisión con participación de la memoria 401. De esta manera, se transmite desde la memoria espejo 201, a modo de ejemplo, en IBSYS=1 y en IBSYS=0. Este bit IBSYS se dispone, a modo de ejemplo, mediante la escritura de IBRH como espacio bit 0 a 5 en el registro 403, para señalar que se ha iniciado una transferencia entre la memoria espejo 401 y la memoria de mensajes 300. Al finalizar dicha
25 transmisión de datos hacia la memoria de mensajes 300, IBSYS se coloca nuevamente en su lugar.

Mientras se está realizando la transferencia desde la memoria espejo 401, el procesador CPU 102 puede escribir los siguientes datos a transferir en la memoria de acceso, por tanto en la sub memoria búfer 400. Mediante la ayuda de otro identificador de acceso, por ejemplo, IBSYH (Input Búfer Busy Host -entrada de búfer espejado ocupado del sistema principal-), en el espacio de bit 15 del registro 403, se puede perfeccionar aún más la identificación. Si el
35 el procesador CPU 102 escribe IBRH, o sea los espacios de bits 0 a 5 del registro 403 durante la realización de una transmisión entre la memoria espejo 401 y la memoria de mensajes 300, entonces IBSYS=1, de esta manera IBSYH se coloca en el registro de peticiones de acceso 403. Tan pronto como concluye la transferencia, por tanto la transmisión que se está realizando, se da inicio a la transferencia requerida y el bit IBSYH se coloca nuevamente en su lugar. El bit IBSYS permanece todo el tiempo marcado, para indicar que los datos son transferidos a la memoria de mensajes. Todos los bits utilizados como ejemplos de ejecución, pueden conformarse con más de un bit como indicadores. La separación de un bit es ventajosa ya que permite una economización de recursos durante el procesamiento.

El mecanismo así descrito le permite al procesador CPU 102 transferir datos en forma continua, a los objetos de
45 mensaje compuestos por un área de inicio HB y un área de datos DB que se encuentran en la memoria de mensajes, siempre y cuando la velocidad de acceso del procesador CPU 102 en la memoria de entrada sea menor o igual al índice de transferencia de datos interno del módulo IP FlexRay del módulo de comunicación 100.

En las figuras 7, 8 y 9 se explican en detalle los accesos de lectura en la memoria de mensajes 300 a través del procesador CPU 102 o participante CPU 102, mediante el búfer de memoria de salida o búfer de memoria de devolución
50 202. La figura 7 muestra, a su vez, nuevamente el módulo de comunicación 100, donde, para mayor claridad, sólo se muestran las partes aquí relevantes del módulo de comunicación 100. Este módulo es, en primer lugar, el gestor de mensajes 200, responsable del control de los resultados, así como dos registros de control 403 y 404, los cuales pueden ser colocados, tal como se representa, por fuera del gestor de mensajes 200 en el módulo de comunicación 100, pero
55 que también pueden estar contenidos dentro del gestor de mensajes 200. 703 representa el registro de peticiones del búfer de salida (Output Búfer Command Request Register) y 704 el registro de enmascaramiento del búfer de salida (Output Búfer Command Mask Register). Los accesos de lectura del procesador CPU 102 se producen en la memoria de mensajes 300 por medio de un búfer de memoria de salida interconectada 202 (Output Búfer). Este búfer de memoria de acceso interconectado 202 se exhibe dividido, por tanto duplicado, a saber en un búfer de sub memoria 701 y un búfer de memoria espejo 700 perteneciente a la sub memoria búfer. Con esto, puede tener lugar, como se describe a continuación, un acceso continuo del controlador CPU 102 a los mensajes, por tanto a los objetos del mensaje y a los respectivos datos de la memoria de mensajes, y puede asegurarse la integridad de los datos y una transmisión rápida en la dirección contraria de la memoria de mensajes hacia el procesador. El control de los accesos se produce mediante el registro de peticiones de salida 703 y el registro de enmascaramiento de salida 704. En el registro 703 se representan
60 con los números 0 a 31, aquí a modo de ejemplo, los respectivos espacios bit en 703 para un ancho de 32 bits. Esto mismo es válido para el registro 704 y los espacios bit 0 a 31.

ES 2 337 163 T3

Conforme a la invención y a modo de ejemplo, los espacios bit 0 a 5, 8 y 9, 15, 16 a 21 del registro 703 mantienen, con respecto al control de desarrollo del acceso de lectura, una función especial. De esta manera se puede inscribir en los espacios bits 0 a 5 del registro 703 un identificador OBRS (Output Búfer Request Shadow) como identificador de mensaje. De la misma manera, se puede inscribir un identificador del sistema principal de salida del búfer OBRH (Output Búfer Request Host -petición de salida del búfer del sistema principal-) en los espacios bit 16 a 21 del registro 703. Los espacios 0 a 1 del registro de salida 704 se encuentran también marcados, donde en 0 y 1, y junto con RDSS (Read Data Section Shadow -espejado del segmento de lectura de datos-) y RHSS (Read Header Section Shadow -espejado del segmento de lectura de la cabecera-) se inscriben otros identificadores como identificadores de datos. Otros identificadores de datos, con RDSH y RSHS, se prevén en los espacios bit 16 y 17. Estos indicadores de datos se conforman de un bit respectivamente, aquí también en la forma más simple, a modo de ejemplo. En el espacio bit 9 del registro 703 se inscribe un indicador de inicio REQ. Posteriormente se prevén un identificador conmutable VIEW, el cual se inscribe, a modo de ejemplo, en el espacio bit 8 del registro 703.

El procesador CPU 102 requiere los datos de un objeto de mensajes de la memoria de mensajes 300, escribiendo la identificación del mensaje deseado, en especial del número del objeto de mensaje deseado, conforme a OBRS, en los espacios bit 0 a 5 del registro 703. De este modo, también aquí el procesador CPU 102 puede, como en la dirección contraria, leer sólo los datos de configuración e inicio KD de un mensaje de un área de inicio o sólo los datos propiamente dichos de un mensaje D del área de datos o leer ambos. Qué parte de los datos del área de inicio o del área de datos puede ser transmitida, se determina, en forma comparable a la dirección contraria, a través de RHSS y RDSS. Esto significa que RHSS indica si los datos de inicio deben ser leídos y RDSS indica si los datos propiamente dichos deben ser leídos.

Un indicador de inicio sirve para dar comienzo a la transmisión de la memoria de mensajes a la memoria espejo 700. Esto significa que se utiliza como indicador, en el caso más simple, un bit, y, mediante la disposición del bit REQ en el espacio bit 9 en el registro de peticiones de salida 703, se da inicio a la transmisión desde la memoria de mensajes 300 hacia la memoria espejo 700. La transmisión que se está realizando es mostrada en el registro 703, mediante un bit OBSYS, aquí nuevamente tratándose del caso más simple. Para evitar colisiones es ventajoso que se coloque el bit REQ sólo cuando el bit OBSYS no esté colocado, o sea, cuando no se esté realizando una transmisión. Aquí se produce entonces también la transferencia de mensajes entre la memoria de mensajes 300 y la memoria espejo 700. La secuencia propiamente dicha podría ser controlada, en forma comparable a la dirección contraria, tal como se describe en las figuras 4, 5, y 6 (registro complementario de ocupación) y efectuarse en una variación a través de un identificador complementario, o sea, un identificador conmutable VIEW en el espacio bit 8 del registro 703. Esto significa que al finalizar la transmisión se coloca nuevamente el bit OBSYS y, mediante la colocación del bit VIEW en el registro de peticiones de salida 703, se intercambian la sub memoria búfer 701 con su equivalente memoria espejo 700, así como los accesos se intercambian y el procesador CPU 102 puede leer el objeto del mensaje requerido por la memoria de mensajes, o sea, el mensaje correspondiente. De esta manera se intercambian las celdas de registro OBRS y OBRH, aquí también en forma comparable a la dirección de transmisión contraria de las figuras 4 a 6. En igual medida se intercambian RHSS y RDSS en oposición a RSHS y RDSH. Como mecanismo de protección puede aquí también preverse, que el bit VIEW sólo se coloque cuando OBSYS ya no se encuentre posicionado, por tanto, cuando no tenga lugar ninguna transmisión.

Con lo mencionado, se producen accesos de lectura del procesador CPU 102 a la memoria de mensajes 300 mediante un búfer de memoria de salida interconectada 202. Este búfer de memoria de salida se muestra, al igual que el búfer de memoria de entrada, dividido, por tanto duplicado, para garantizar un acceso continuo del controlador CPU 102 a los mensajes, por tanto a los objetos del mensaje y a los respectivos datos de la memoria de mensajes, logrando así mantener la integridad de los datos y una transmisión rápida.

Mediante la utilización de los búfer de memorias de entrada y salida descritos, se asegura que un controlador CPU pueda acceder sin interrupciones a la memoria de mensajes, pese a los tiempos de latencia internos del módulo.

Para asegurar la integridad de los datos, se efectúa la transmisión de los datos, en especial la transmisión en el módulo de comunicación 100, mediante el gestor de mensajes 200 (Message Handler MHD). Con este fin se representa en la figura 10 el gestor de mensajes 200. El gestor de mensajes puede ser representado en su funcionalidad, mediante varias máquinas de estado o máquinas automáticas de estado, o sea, mediante las llamadas máquinas de estado finito (FSM). Para esto se prevén como mínimo tres máquinas de estado y, en una forma de ejecución especial, cuatro máquinas de estado finito. Una primera máquina de estado finito es la IOBF-FSM, designada con el número 501 (Input/Output Búfer State Machine búfer de máquina de estado infinito de entrada/salida). Esta IOBF-FSM podría estar dividida, de acuerdo a cada dirección de transmisión con respecto al búfer de memoria de entrada o al búfer de memoria de salida, en dos búferes de máquinas de estado finito IBF-FSM (Input Búfer FSM) y OBF-FSM (Output Búfer FSM), con lo cual se podría pensar en como máximo cinco máquinas automáticas de estado (IBF-FSM, OBF-FSM, TBF1-FSM, TBF2-FSM, AFSM). Prever una IOBF-FSM común es, sin embargo, preferente. Una, al menos, segunda máquina de estado finito se divide, durante el desarrollo del ejemplo preferente de ejecución, en dos bloques 502 y 503 y sirve a los dos canales A y B con respecto a las memorias 205 y 206, tal como se describe en la Fig. 2. Para ello puede preverse una máquina de estado finito para servir a los dos canales A y B o, como en la forma preferente, una máquina de estado finito TBF1-FSM, (Transient Búfer 1 -búfer transitorio 1- (206, RAM A) State Machine -de máquina de estado-) denominada como 502 para el canal A y para el canal B una TBF2-FSM (Transient Búfer 2 -búfer transitorio 2- (205, RAM B) State Machine -de máquina de estado-), denominada como 503.

ES 2 337 163 T3

Para controlar el acceso de las tres máquinas de estado finito 501-503 en el ejemplo de ejecución preferente, se emplea una máquina de estado finito árbitro, conocida como AFSM, denominada como 500. Los datos (KD y/o D) se transmiten en el módulo de comunicación, mediante un temporizador, adaptado o generado a través de un medio temporizador, como, por ejemplo, un VCO (Voltage controlled oscillator -voltaje controlado por oscilador-), un resonador de cuarzo, etc. El temporizador T puede ser generado en el módulo o desde afuera, por ejemplo, ser determinado como temporizador bus. Esta máquina de estado finito árbitro AFSM 500 permite a las tres máquinas de estado finito 501-503 acceder, en forma alterna, a la memoria de mensajes, en especial respectivamente durante un período elemental T. Esto significa, que el tiempo disponible se divide en correspondencia con las peticiones de acceso de cada máquina de estado automática 501, 502, 503. Si se verifica una petición de acceso de una sola máquina de estado finito, ésta obtiene el 100% del tiempo de acceso, o sea, todos los temporizadores T. Si se verifica una petición de acceso de dos máquinas de estado automáticas, cada máquina de estado finito obtiene 50% del tiempo de acceso. Por último, si se verifica una petición de acceso de tres máquinas de estado automáticas, cada una de las máquinas obtiene 1/3 del tiempo de acceso. De esta forma se aprovecha en forma óptima el ancho de banda que se encuentra disponible.

La primera máquina de estado finito 501, o sea IOBF-FSM realiza en caso necesario las siguientes acciones:

- Transferencia de datos desde la memoria de entrada 201 hacia el objeto de mensaje seleccionado en la memoria de mensajes 300.
- Transferencia del objeto de mensaje seleccionado en la memoria de mensajes 300 hacia al búfer de memoria de salida 202.

La máquina de estado para el canal A 502, o sea TBF1FSM, realiza las siguientes acciones:

- Transferencia del objeto de mensaje seleccionado en la memoria de mensajes 300 hacia la memoria búfer 206 del canal A.
- Transferencia de datos desde la memoria búfer 206 hacia el objeto de mensaje seleccionado en la memoria de mensajes 300.
- Búsqueda de los objetos de mensaje adecuados en la memoria de mensajes, donde se busca, durante la recepción del objeto de mensaje (Receive Búfer) hasta el almacenamiento, un mensaje recepcionado en el canal A en el marco de un filtrado de aceptación y durante la transmisión el primer objeto de mensaje a transmitir en el canal A (Transmit Búfer).

Análoga a la anterior es la acción de TBF2-FSM, o sea de la máquina de estado finita para el canal A en el bloque 53. Ésta dirige la transferencia de datos del objeto de mensaje seleccionado en la memoria de mensajes 300 hacia la memoria búfer 205 por el canal B y de los datos de transferencia de la memoria búfer 205 hacia el objeto de mensaje seleccionado en la memoria de mensajes 300. También la función de búsqueda es análoga a TBF1-FSM, tras un objeto de mensaje adecuado en la memoria de mensajes (Receive Búfer -búfer de recepción-) hasta el almacenamiento, un mensaje recepcionado en el canal B en el marco de un filtrado de aceptación y durante la transmisión del primer objeto de mensaje a transmitir en el canal B (Transmit Búfer -búfer de transmisión).

En la figura 11 se representan nuevamente el desarrollo y las rutas de transmisión. Las tres máquinas de estado 501-503 dirigen las respectivas transmisiones de datos entre cada parte. De este modo se denomina nuevamente al procesador CPU como 102, al búfer de memoria de acceso como 201 y al búfer de memoria de salida como 202. Como 300 se denomina a la memoria de mensajes y ambas memorias búfer para el canal A y canal B como 206 y 205. Los elementos de interfaz 207 y 208 son asimismo representados. La primera máquina de estado automática IOBF-FSM 501 dirige la transferencia de datos Z1A y Z1B, desde la memoria de acceso 201 a la memoria de mensajes 300 y de la memoria de mensajes 300 al búfer de memoria de salida 202. La transmisión de datos se produce mediante buses de datos con una amplitud de palabra de, por ejemplo, 32 bits, donde también cualquier otra cantidad de bits es posible. Esto mismo es válido para la transmisión Z2 entre la memoria de mensajes y la memoria búfer 206. Esta transmisión de datos se dirige a través de TBF1-FSM, o sea de la máquina de estado para el canal A 502. La transferencia Z3 entre la memoria de mensajes 300 y la memoria búfer 205 se dirige a través de la máquina de estado automática TBF2-FSM 503. También en este caso la transmisión de datos se produce mediante buses de datos con una amplitud de palabra de, por ejemplo, 32 bits, donde también cualquier otra cantidad de bits es posible. Por lo general, la transferencia de un objeto de mensaje completo por medio de las llamadas rutas de datos requiere varios períodos del temporizador T. Por consiguiente se produce una división del tiempo de transmisión con respecto a los períodos del temporizador T a través del árbitro AFSM 500. En la figura 11 se representan las rutas de datos entre los componentes de almacenamiento controlados por el administrador de mensajes. Para garantizar la integridad de los objetos de mensaje almacenados en la memoria de mensajes, se deben intercambiar al mismo tiempo, de manera ventajosa, simultáneamente datos que se encuentran en una de las rutas representadas Z1A, Z1B, así como Z2 y Z3 al mismo tiempo.

ES 2 337 163 T3

En la representación 12 se muestra mediante un ejemplo, cómo se dividen los temporizadores de sistema T que se encuentran a disposición, en tres máquinas de estado automáticas requeridas. En la fase 1 se verifican peticiones de acceso de la máquina de estado automática 501 y la máquina de estado automática 502, lo que significa que el tiempo total se divide respectivamente a la mitad en ambas máquinas de estado automáticas. Con respecto a los períodos del temporizador en la fase 1, esto significa, que la máquina de estado automática 501 en los períodos del temporizador T1 y T3 logra el acceso, mientras que la máquina de estado automática 502 lo logra en los períodos del temporizador T2 y T4. En la fase 2 se verifica el acceso sólo a través de la máquina de estado automática 501, de manera que los tres períodos, o sea el 100% del tiempo de acceso de T5 a T7 en IOBF-FSM se suprime. En la fase 3 se verifican peticiones de acceso de las tres máquinas de estado automáticas 501 a 503, de manera que se produce una tripartición del tiempo total de acceso. El árbitro AFSM divide el tiempo de acceso, por ejemplo, de manera tal que logren el acceso, la máquina de estado automática 501 en los períodos del temporizador T8 y T11, la máquina de estado automática 502 en los períodos del temporizador T9 y T12 y la máquina de estado automática 503 en los períodos del temporizador T10 y T13. En la fase 4, por último, se produce el acceso a través de dos máquinas de estado automáticas, 502 y 503, en los dos canales A y B del módulo de comunicación, de manera que se produce una división de acceso de los períodos del temporizador T14 y T16 en la máquina de estado automática 502 y de los períodos del temporizador T15 y T17 en la máquina de estado automática 503.

La máquina de estado automática árbitro AFSM 500 se ocupa de dividir el acceso de las máquinas de estado automáticas, en forma de períodos alternados, dado el caso, de que más de una de las tres máquinas de estado automáticas presente una petición para el acceso a la memoria de mensajes 300. Esta forma de procedimiento asegura la integridad de los objetos de mensaje que se encuentran en la memoria de mensajes, por tanto la integridad de los datos. Si por ejemplo, el procesador CPU 102 desea leer un objeto de mensaje mediante el búfer de memoria de salida, mientras que en ese mismo momento se escribe un mensaje recepcionado en ese objeto de mensaje, se leerá o el estado antiguo o el estado nuevo, dependiendo de cual petición se presentó en primer lugar, de manera tal que no se produzca una colisión de los accesos en la misma memoria de mensajes.

El procedimiento descrito le permite al procesador CPU 102 en actividad, leer o escribir cualquier objeto de mensaje en la memoria de mensajes, sin que el objeto de mensaje seleccionado para la duración del acceso del procesador CPU 102 sea cerrado (Búfer Locking -cierre de búfer-) por la participación en el intercambio de datos en los dos canales del bus FlexRay. Al mismo tiempo, se asegura la integridad de los datos depositados en la memoria de mensajes mediante la jerarquización de los accesos y, a su vez, se incrementa la velocidad de transmisión mediante el aprovechamiento del ancho de banda en su totalidad.

35

40

45

50

55

60

65

ES 2 337 163 T3

REIVINDICACIONES

5 1. Método para el acceso de datos de una memoria de mensajes (300), para lo cual se prevee una primera memoria búfer (201, 202), **caracterizado** porque, la primera memoria búfer se encuentra dividida en una sub memoria búfer (400, 701) y una memoria espejo (401, 700) para la sub memoria búfer, por lo cual durante un acceso a los datos se ejecutan los siguientes pasos:

- entrada de datos en la sub memoria búfer e
- 10 - intercambio del acceso de la sub memoria búfer y de la memoria espejo, de manera que
- los siguientes datos puedan ser ingresados en la memoria espejo, mientras que los datos ingresados puedan al mismo tiempo ser extraídos de la sub memoria búfer.

15 2. Método conforme a la reivindicación 1, **caracterizado** porque, la memoria de mensajes es parte de un módulo de comunicación (100), por lo cual se pueden ingresar o devolver datos hacia o desde la memoria de mensajes (300), por lo que la memoria de mensajes (300) se encuentra conectada a una disposición de memoria búfer (201 y 202) y los datos se pueden transmitir en una primera dirección de transmisión hacia la memoria de mensajes y en una segunda dirección de transmisión desde la memoria de mensajes, por lo cual se preveen una primera y una segunda memoria búfer, por lo cual la disposición de memoria búfer presenta una primera memoria búfer como una memoria de acceso (201), en la primera dirección de transmisión, y en la segunda dirección de transmisión, la segunda memoria búfer se presenta como un búfer de memoria de salida (202) y la memoria de acceso y el búfer de memoria de salida se dividen respectivamente en una sub memoria búfer (400, 701) y una memoria espejo (401, 700) para la sub memoria búfer, por lo que en cada transmisión se ejecutan los siguientes pasos:

- entrada de datos en la respectiva sub memoria búfer e
- 30 - intercambio del acceso de sub memoria búfer y memoria espejo, de manera que
- los siguientes datos puedan ser ingresados en la memoria espejo, mientras que los datos ingresados puedan ser devueltos de la sub memoria búfer en la dirección prevista.

35 3. Método conforme a la reivindicación 2, **caracterizado** porque se marca el intercambio de acceso de la sub memoria búfer y la memoria espejo mediante un indicador.

40 4. Método conforme a la reivindicación 2, **caracterizado** porque, el intercambio de acceso de la sub memoria búfer y la memoria espejo se produce mediante un indicador.

45 5. Método conforme a la reivindicación 2, **caracterizado** porque se transmiten los datos en mensajes y los mensajes son provistos de un identificador de mensajes (IBRH, IBRS, OBRH, OBRS), por lo cual se produce el intercambio de acceso de la sub memoria búfer y la memoria espejo, cuando el indicador de mensaje se inscribe en un registro predeterminado (403, 404, 703, 704).

50 6. Método conforme a la reivindicación 2, **caracterizado** porque se transmiten los datos en mensajes y los datos se dividen en un mensaje, por lo menos en datos primarios (KD) y datos secundarios (D), por lo que, dependiendo de al menos un identificador de datos (LHSH, LDSH, LDSS, LHSS, RHSS, RDSS, RDSH, RSHH), sólo se transmiten los datos primarios (KD) o los datos secundarios (D).

7. Método conforme a la reivindicación 2, **caracterizado** porque, el acceso a los datos, se marca mediante un indicador (IBSYS, IBSYH, OBSYS), el cual se inscribe en un registro.

55 8. Método conforme a la reivindicación 5, **caracterizado** porque, una información puede ser transmitida desde el indicador mismo (IBSYS, IBSYH, OBSYS) o desde su posición en el registro y puede acceder a la sub memoria búfer o a la memoria espejo.

60 9. Método conforme a las reivindicaciones 5 ó 6, **caracterizado** porque, durante el intercambio de accesos, por lo menos una parte de las inscripciones de los registros, en especial los indicadores, se intercambian en el registro.

10. Método conforme a la reivindicación 2, **caracterizado** porque, el intercambio de accesos de la sub memoria búfer y de la memoria espejo se inician mediante un identificador conmutable (VIEW).

65 11. Método conforme a la reivindicación 2, **caracterizado** porque, un indicador de inicio (STXRH, REQ) se inscribe en un registro, para al menos una dirección de transmisión, cuando los datos deban escribirse en una memoria.

ES 2 337 163 T3

12. Disposición de búfer de memoria para el acceso a datos de una memoria de mensajes (300) de un módulo de comunicación (100) **caracterizado** porque, la primera memoria búfer (201, 202) contiene una sub memoria búfer (400, 701) y una memoria espejo (401, 700) de la sub memoria búfer, por lo que se prevén los medios, a través de los cuales se intercambia el acceso de la sub memoria búfer (400, 701) y de la memoria espejo (401, 700).

5

13. Disposición de búfer de memoria conforme a la reivindicación 12, **caracterizado** porque, la disposición de búfer de memoria presenta una primera memoria búfer y una segunda memoria búfer y la disposición de búfer de memoria se divide en un búfer de memoria de entrada (201) y un búfer de memoria de salida (202) y el búfer de memoria de entrada (201) contiene una primera sub memoria búfer (400) y una primera memoria espejo (401) de la sub memoria búfer y el búfer de memoria de salida (202) contiene una segunda sub memoria búfer (701) y una segunda memoria espejo (700), por lo que se prevén los medios primarios, a través de los cuales se intercambia el acceso a la primera sub memoria búfer (400) y la primera memoria espejo (401) y se prevén los medios secundarios, a través de los cuales se intercambia el acceso a la segunda sub memoria búfer (701) y la segunda memoria espejo (700).

10

15

14. Módulo de comunicación con una disposición de memoria búfer conforme a la reivindicación 12.

15. Módulo de comunicación con una disposición de memoria búfer conforme a la reivindicación 13.

20

25

30

35

40

45

50

55

60

65

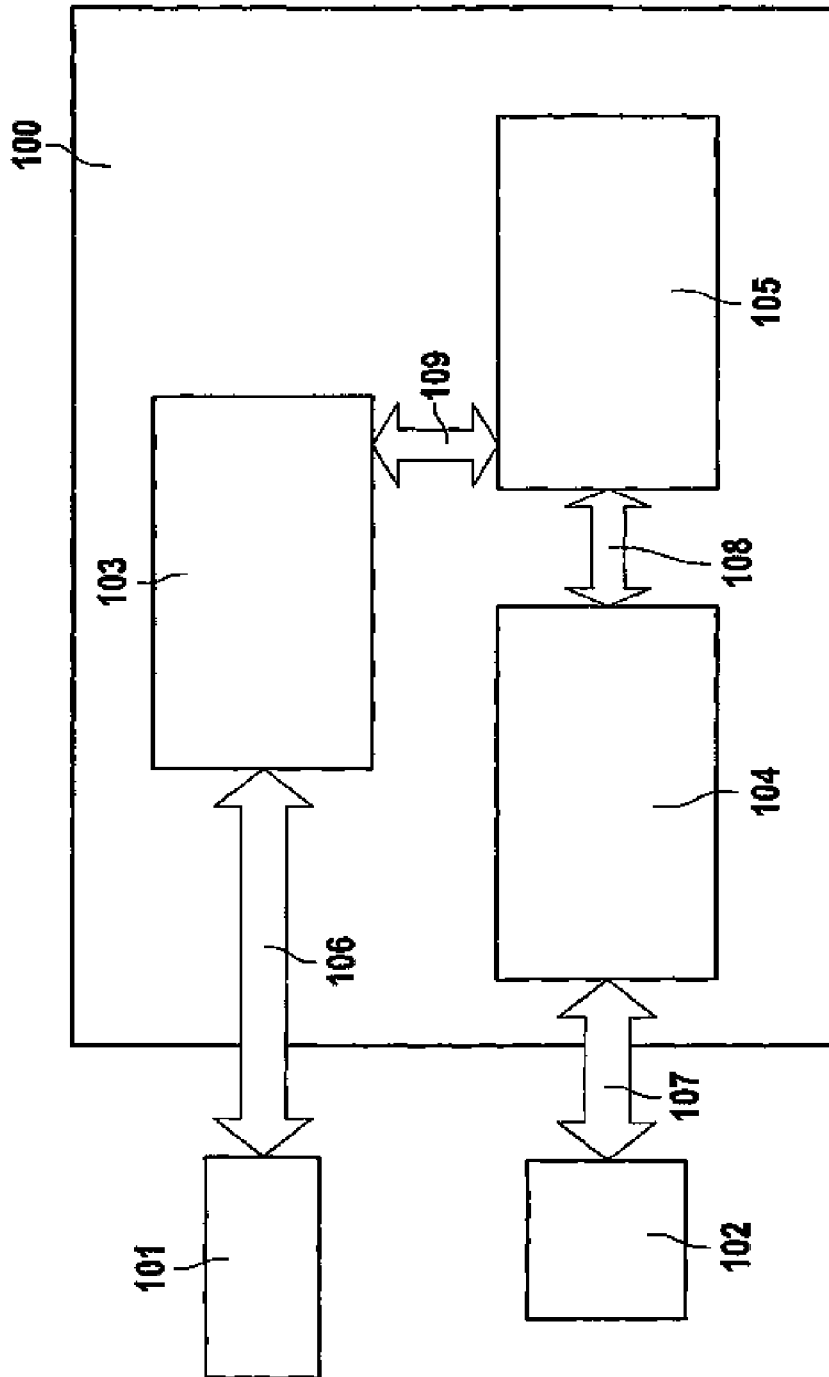


Fig. 1

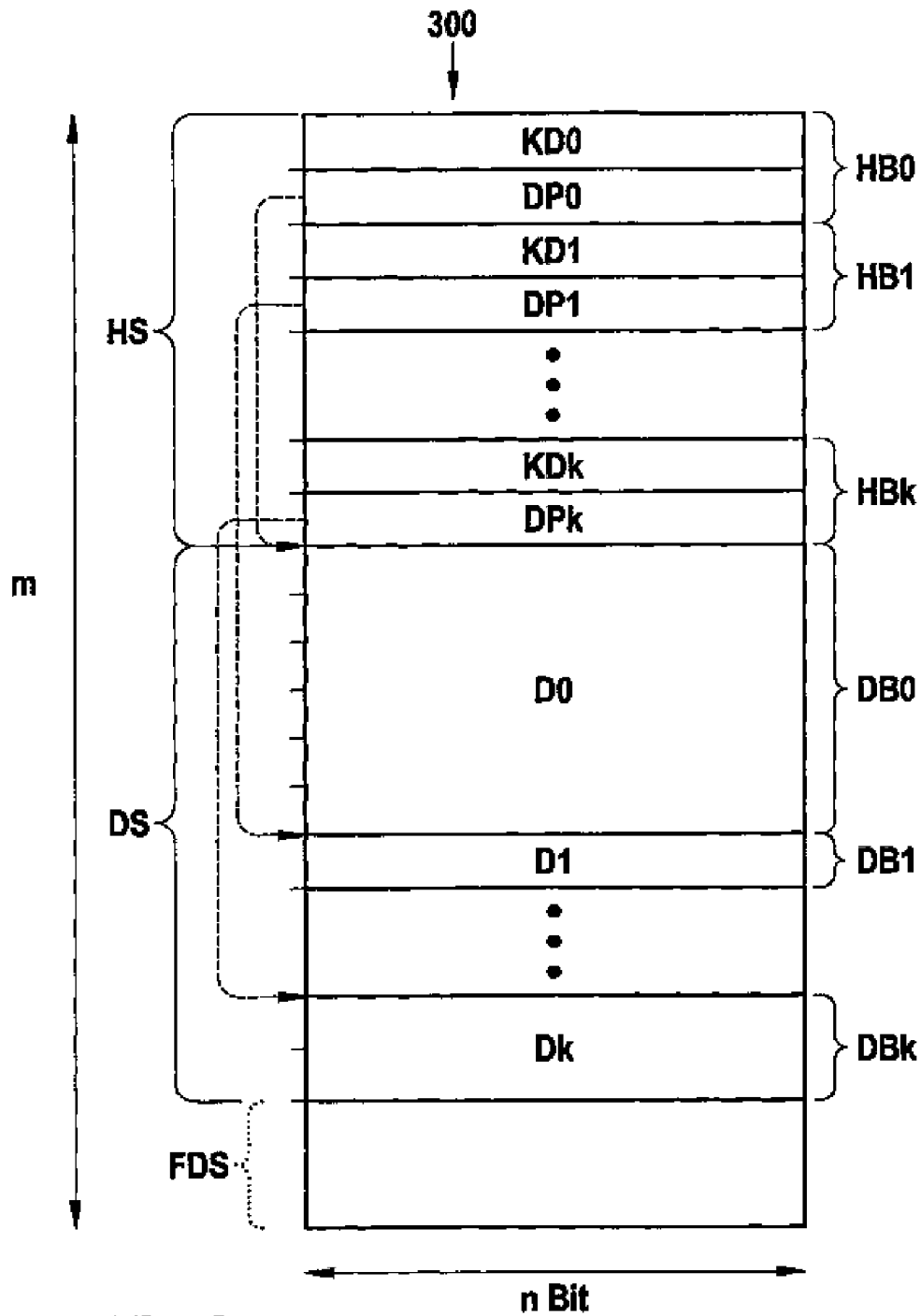


Fig. 3

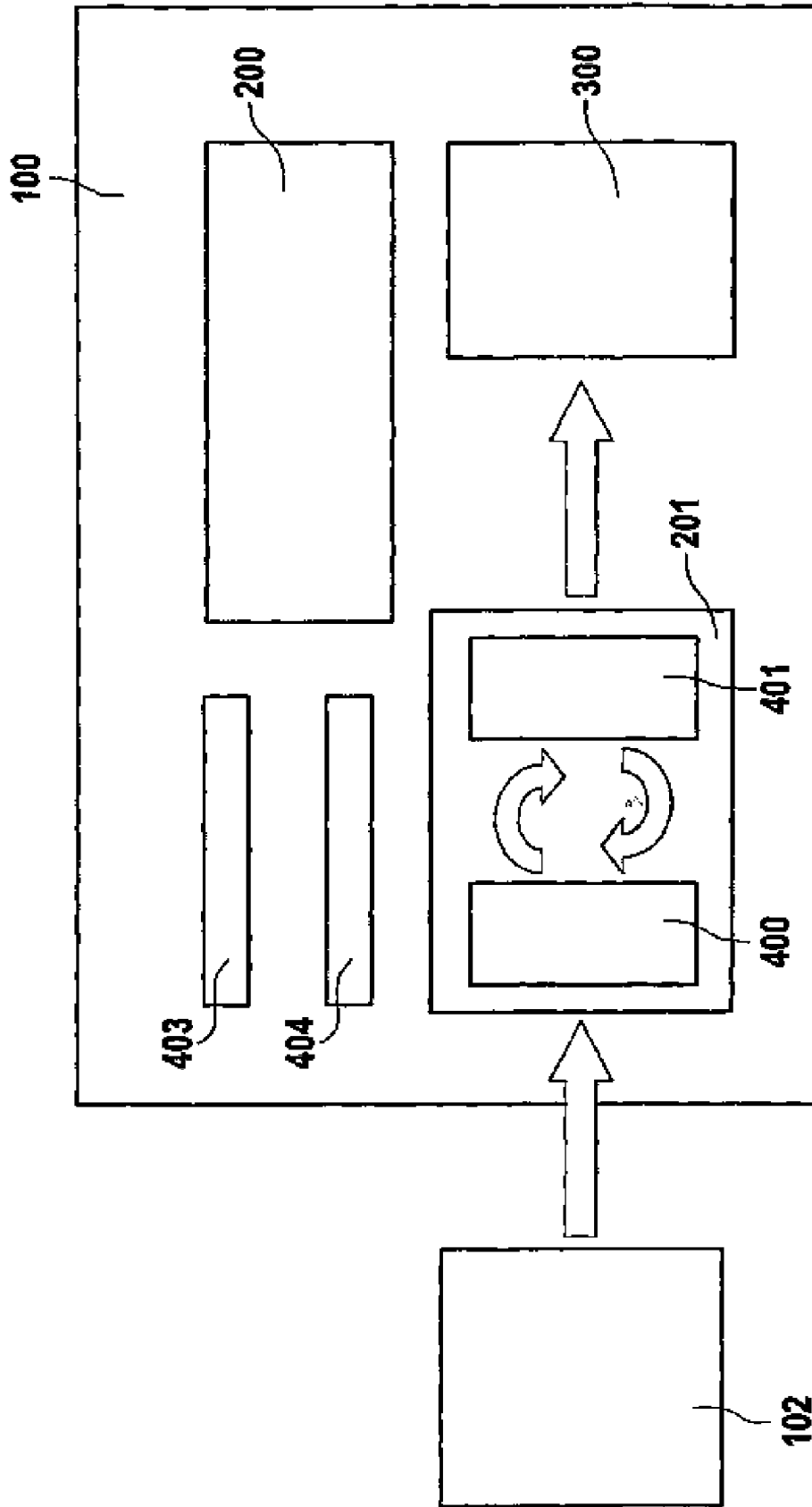


Fig. 4

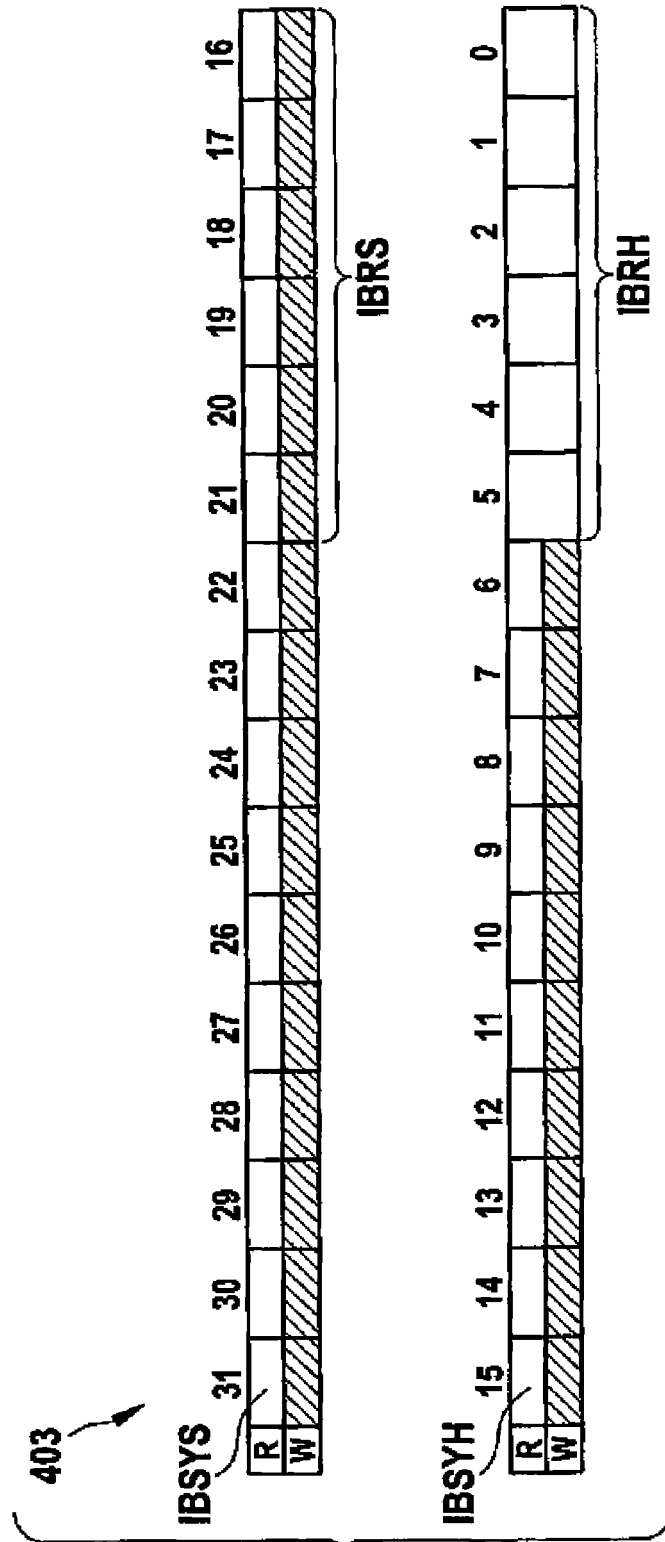
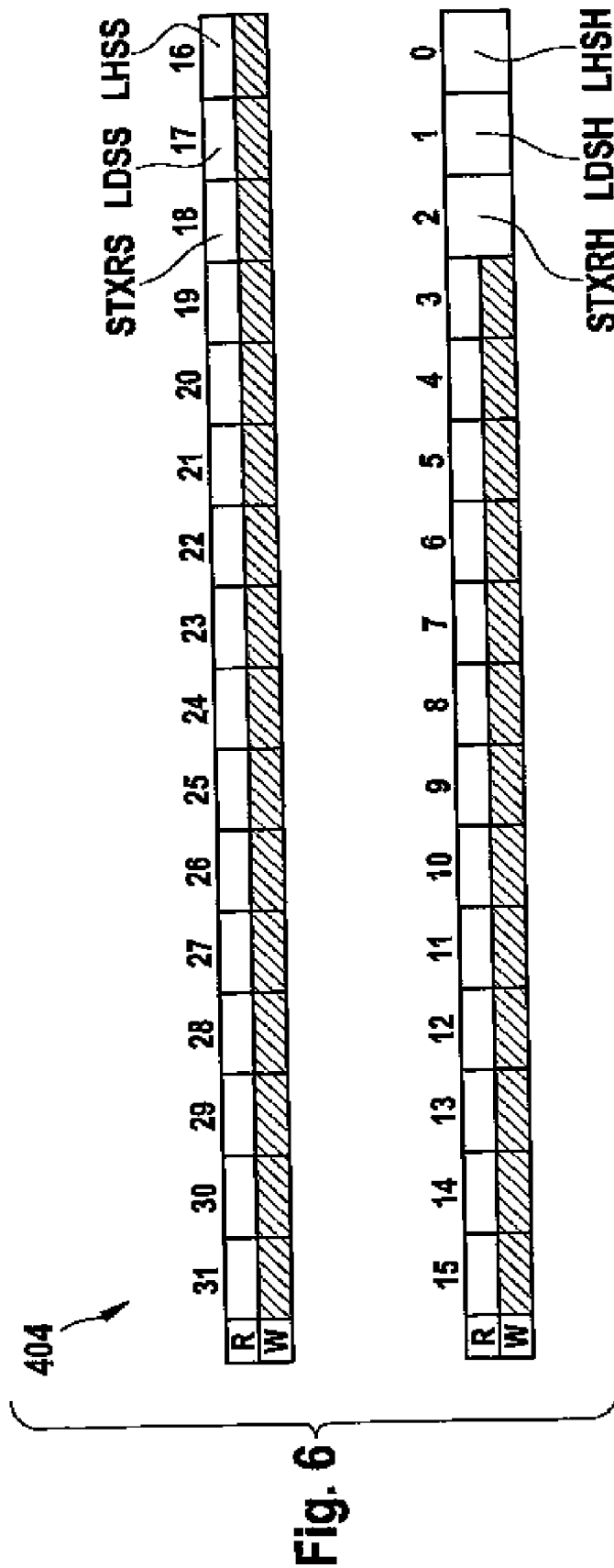


Fig. 5



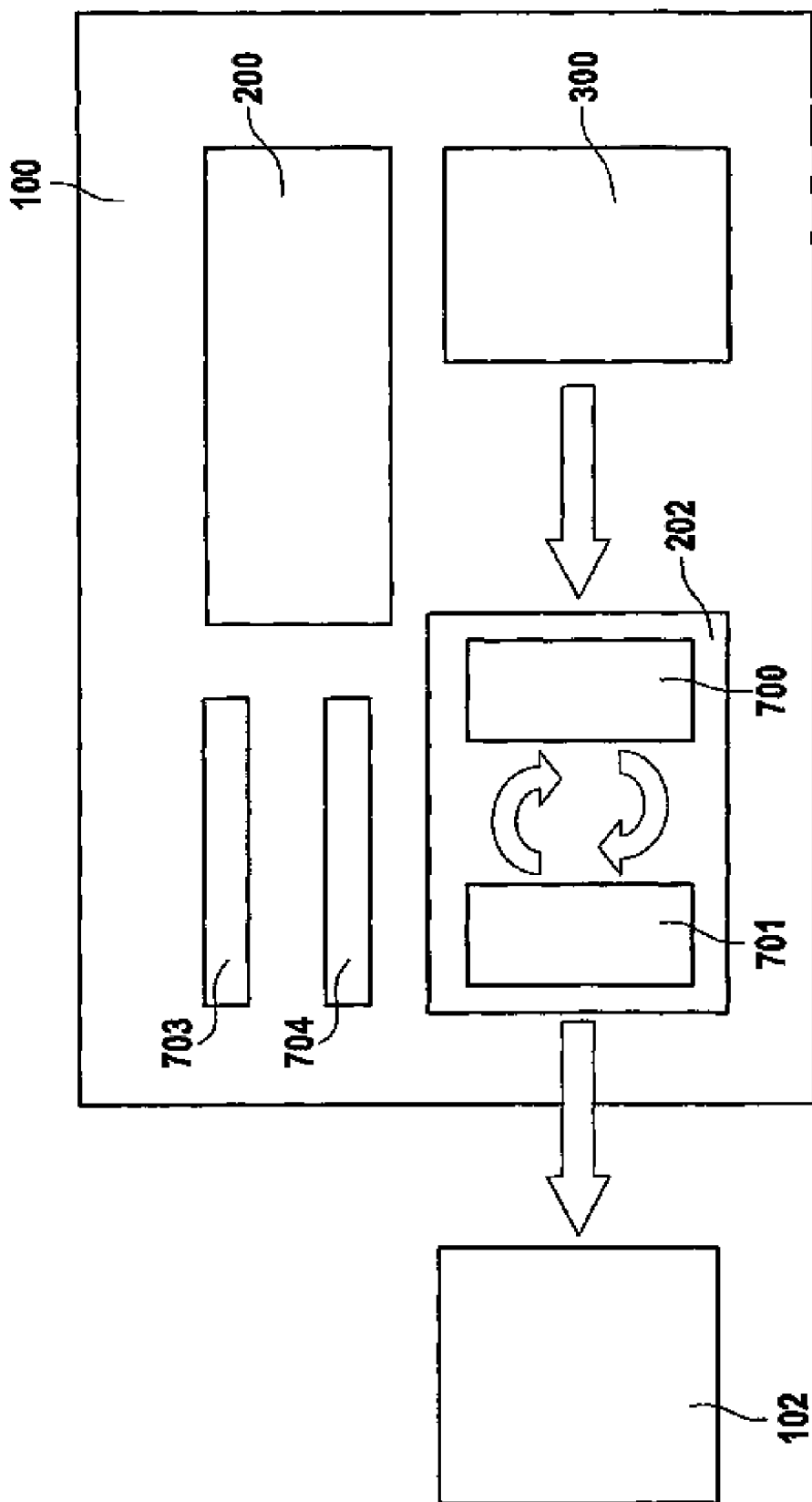


Fig. 7

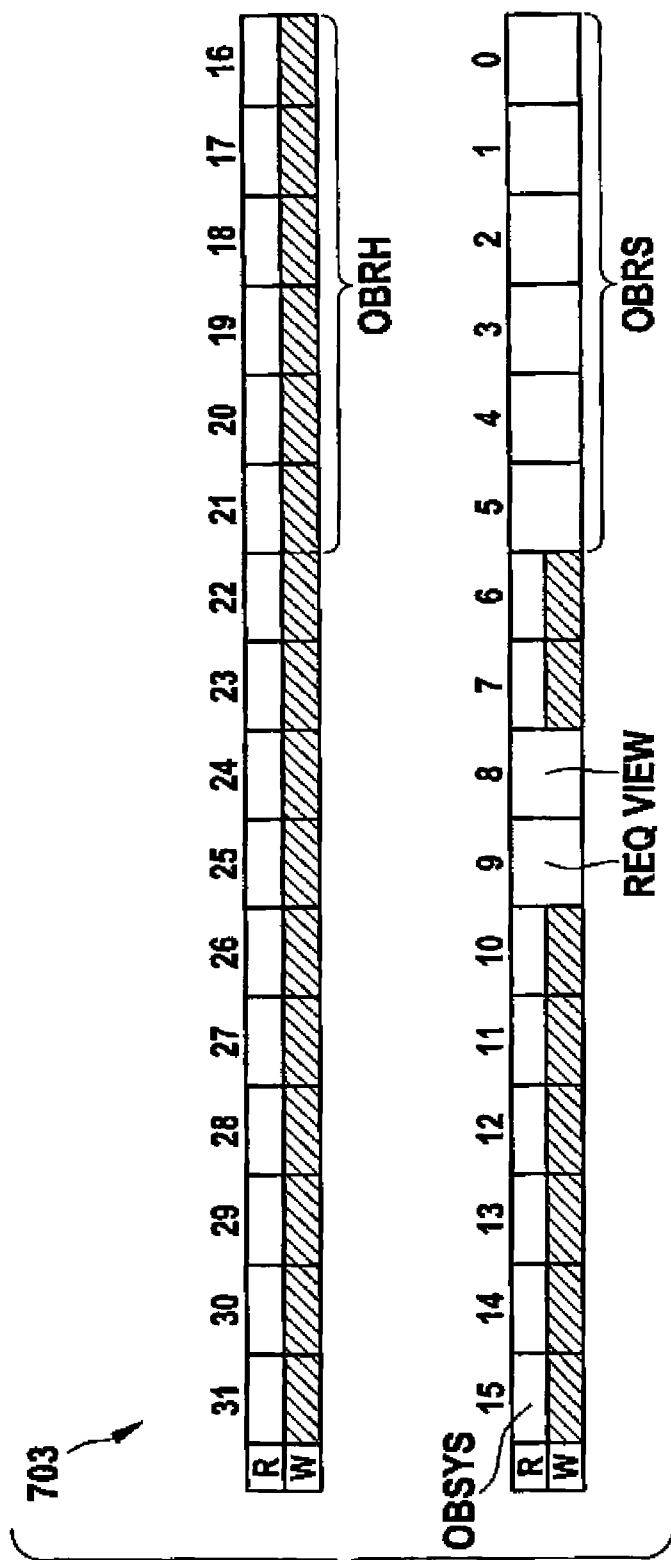


Fig. 8

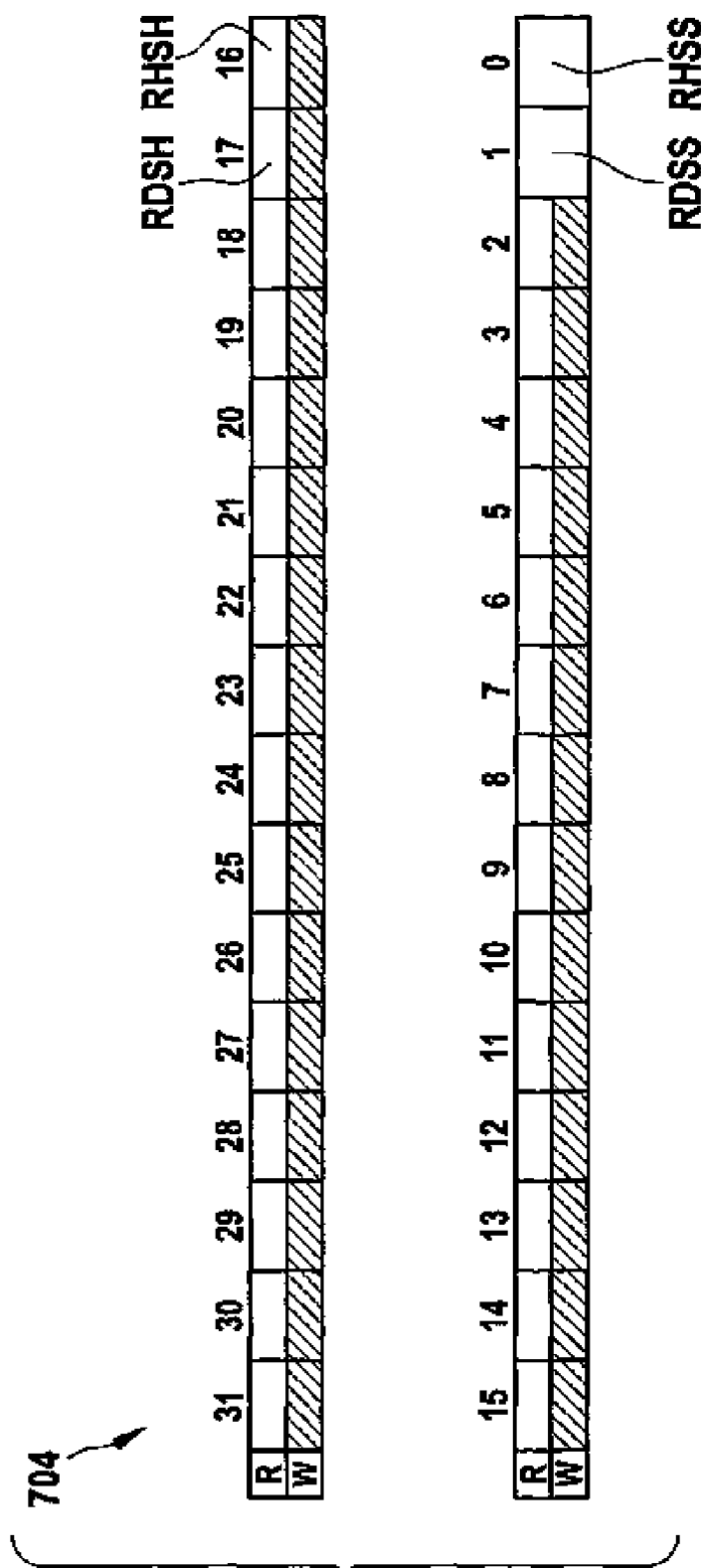


Fig. 9

Fig. 10

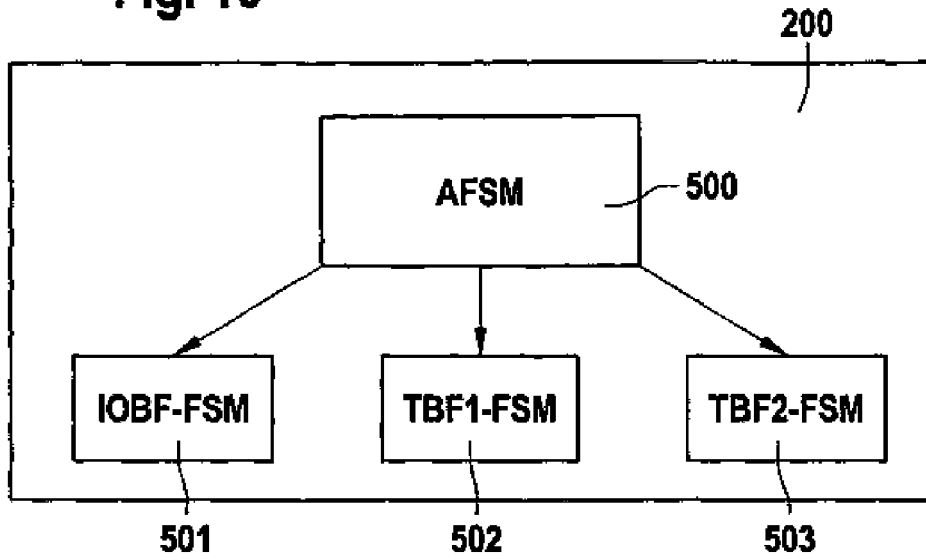
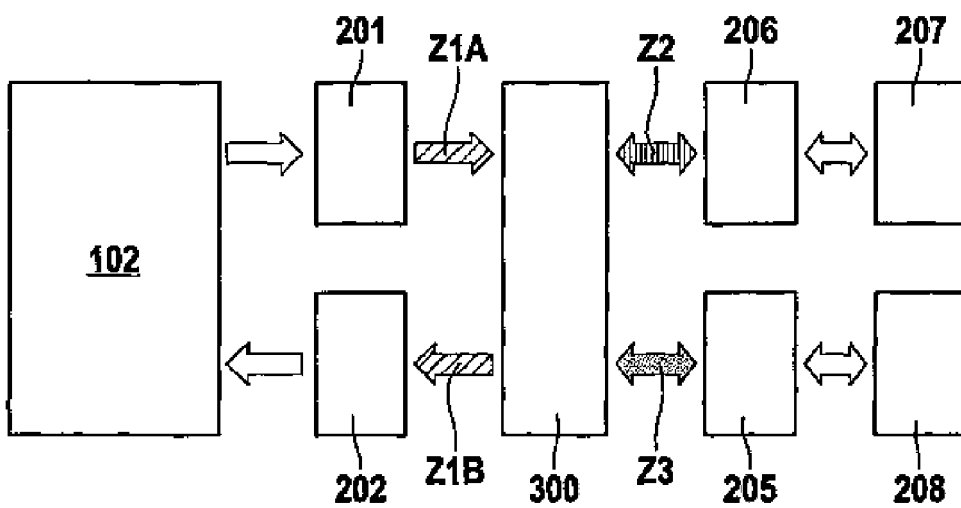


Fig. 11



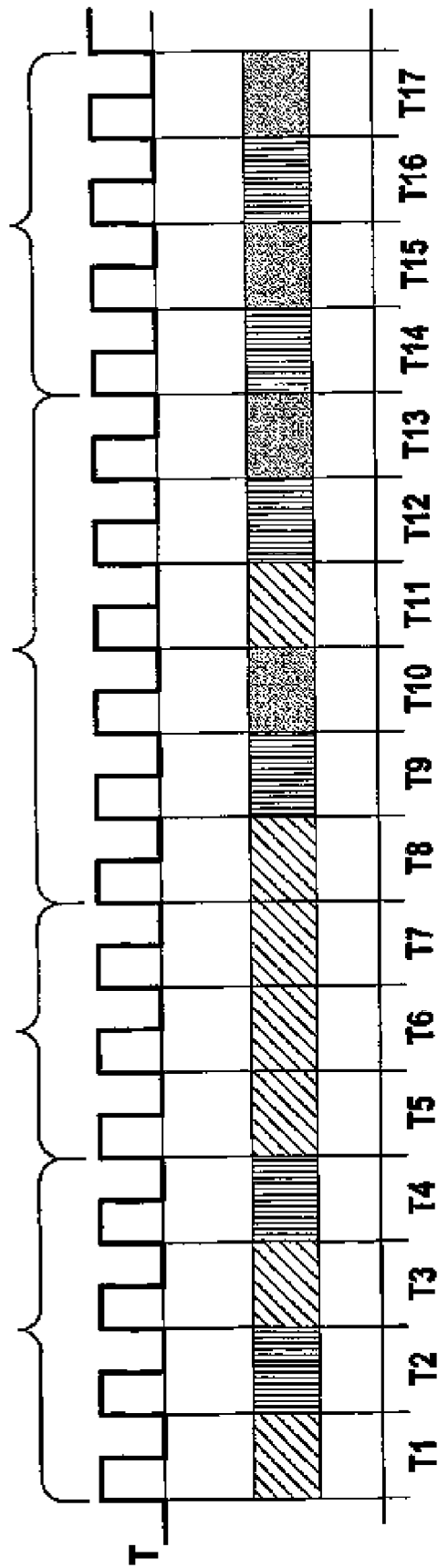


Fig. 12