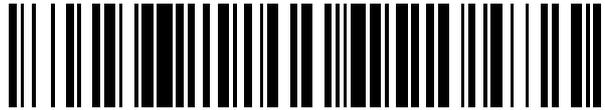


19



OFICINA ESPAÑOLA DE
PATENTES Y MARCAS

ESPAÑA



11 Número de publicación: **2 446 590**

51 Int. Cl.:

G06F 9/44

(2006.01)

12

TRADUCCIÓN DE PATENTE EUROPEA

T3

96 Fecha de presentación y número de la solicitud europea: **07.04.2009 E 09730478 (6)**

97 Fecha y número de publicación de la concesión europea: **20.11.2013 EP 2266028**

54 Título: **Procedimiento y dispositivo para implementación de un protocolo de comunicación en una unidad de control, especialmente para aplicaciones vehiculares**

30 Prioridad:

08.04.2008 EP 08425235

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

10.03.2014

73 Titular/es:

**IVECO S.P.A. (100.0%)
Via Puglia 35
10156 Torino, IT**

72 Inventor/es:

CORTESE, DEMETRIO

74 Agente/Representante:

RUO, Alessandro

ES 2 446 590 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

Procedimiento y dispositivo para implementación de un protocolo de comunicación en una unidad de control, especialmente para aplicaciones vehiculares.

Campo de la Invención

[0001] La presente invención se refiere a un procedimiento y un dispositivo para la implementación de un protocolo de comunicación en una unidad de control, especialmente para aplicaciones vehiculares.

Descripción de la técnica anterior

[0002] La realización de unidades de control en el campo vehicular, y en general donde sean necesarios altos rendimientos y una alta fiabilidad, es un punto crítico en el proceso de producción, debido a la necesidad de un plazo de realización eficiente.

[0003] Cuando tienen que integrarse más unidades de control en una red de comunicación adecuada para gestionar procesos diferentes y paralelos, existe una probabilidad muy alta de que se produzcan algunos errores durante la realización de las unidades de control. En consecuencia, los periodos de prueba son particularmente prolongados y los costes son altos.

[0004] La gestión de la comunicación en las unidades de control, puede definirse en general como constituida por dos procesos fundamentales: el primero se gestiona por el dispositivo de interfaz denominado "Controlador CAN", cuando el protocolo de comunicación es del tipo CAN conocido en la técnica, y se refiere a la interconexión de bajo nivel con la red de comunicación vehicular en cuanto a la transmisión y recepción de los mensajes; el segundo, definido como intérprete/manipulador de los mensajes, tiene que interpretar los mensajes recibidos, extrayendo la información para facilitarla al Software de aplicación, y tiene que compilar los mensajes que se van a transmitir, partiendo de la información facilitada por el software de aplicación.

[0005] En la actualidad, el desarrollo del denominado intérprete/manipulador necesita la intervención de dos profesionales, concretamente el ingeniero del sistema, que define las especificaciones y las funcionalidades de los mensajes que la unidad de control tiene que gestionar, y el ingeniero de software, que traduce en el lenguaje de software apropiado las especificaciones formuladas por el ingeniero de sistemas.

[0006] En particular, el ingeniero de sistemas escribe las especificaciones de acuerdo con la estandarización del mensaje definida por el fabricante para el que trabaja, por ejemplo, de acuerdo con el protocolo J1939 del tipo conocido, mientras que el ingeniero de software tiene que implementar dicha especificación considerando las características del hardware de la unidad de control específica a programar. Por este motivo, cuando la unidad de control cambia, es necesario implementar de nuevo las funciones de nivel de la aplicación que realizan las llamadas a la función de bajo nivel, con el fin de obtener el empaquetado y desempaqueado correcto de los mensajes.

[0007] Además, cada fabricante, aunque respetando los estándares de comunicación, ha de desarrollar los mensajes CAN del propietario y una gestión específica para la identificación de los fallos y su ulterior recuperación.

[0008] Por ejemplo, en un mensaje CAN, la información se dispone en serie en una palabra de bits y cada mensaje puede tener una longitud variable, dependiendo también de cómo quiere personalizar el mensaje el fabricante, y puede comprender uno o más campos de información.

[0009] En los últimos años, el uso de software de aplicación a base de modelos comerciales ha simplificado de forma considerable la redacción de códigos de programación, pero el tiempo necesario para el desarrollo y la alta probabilidad de error son todavía un grave problema. En particular, el paquete MatLab® con la aplicación de software Simulink® de Math-Works® es un entorno de desarrollo visual que, por medio de la interconexión gráfica de bloques, permite definir el diagrama de flujo funcional y la función de transferencia de cualquier sistema, conociéndose en la técnica que la función de transferencia puede estudiarse como un filtro, y después el paquete convierte todo en código de programación.

[0010] La figura 1 de la técnica anterior muestra un ejemplo de diagrama de bloques creado en el entorno Simulink®. En general a la izquierda están las entradas, y a la derecha están la salida o salidas, de acuerdo con la trayectoria indicada por las flechas de conexión entre los bloques.

[0011] Los elementos gráficos ovalados a la izquierda generalmente indican las entradas, mientras que el rectángulo superior a la izquierda indica un parámetro de trabajo modificable del filtro, mientras que el óvalo a la derecha indica la salida del filtro.

[0012] Esta salida puede ser un parámetro de trabajo o una variable en relación a una magnitud física.

5 [0013] A pesar de esta simplificación considerable, la interacción con este entorno de desarrollo para la programación de unidades de control no es adecuada para profesionales que no sean particularmente expertos. Por este motivo, Está destinado a utilizarse básicamente por el ingeniero de software sin resolver los problemas en relación a la redacción de las especificaciones funcionales de acuerdo con los estándares corporativos y su ulterior implementación de acuerdo con la unidad de control específica que se va a programar.

10 [0014] Por lo tanto, los problemas técnicos relacionados con la complejidad de las operaciones de implementación de un protocolo de comunicación en una unidad de control, y en segundo lugar a la redacción de la documentación técnica que puede conducir a interpretaciones erróneas, que provoca una pérdida de tiempo y de dinero, están aún sin resolver. El documento US6931574 muestra analizadores de protocolo de red para capturar, analizar y mostrar gráficamente información sobre tramas de datos que se transmiten por una red. El documento EP 1551142 permite la traducción de mensajes de un primer a un segundo protocolo.

Resumen de la invención

20 [0015] El fin de la presente invención es proporcionar un procedimiento y un dispositivo para la implementación de un protocolo de comunicación en una unidad de control, especialmente para aplicación vehicular, capaz de resolver los problemas que se han descrito anteriormente, especialmente en cuanto a la minimización del tiempo de desarrollo y de la probabilidad de error durante las operaciones de programación de la unidad de control, además de la facilitación de las operaciones de interconexión entre el ingeniero y el sistema de procesamiento.

25 [0016] Un objeto de la presente invención es un procedimiento para programar un procedimiento de alto nivel de una unidad de control de un vehículo, de acuerdo con la reivindicación 1.

30 [0017] Otro objeto de la presente invención es un dispositivo para programar un procedimiento de alto nivel de una unidad de control de un vehículo, adecuado para realizar el procedimiento que se ha descrito anteriormente, de acuerdo con la reivindicación 10. Dicho dispositivo comprende al menos una interfaz gráfica que permite la selección de al menos un mensaje de entrada, la filtración de este mensaje de acuerdo con al menos una especificación funciona, la generación automática de una equivalencia del modelo de bloques con respecto al mensaje que se va a enviar y la conversión automática del modelo en lenguaje de programación.
35 Las reivindicaciones dependientes describen las realizaciones preferidas de la invención, y son una parte integrante de esta descripción.

Breve descripción de las figuras

40 [0018] Características y ventajas adicionales de la invención serán evidentes a la luz de una descripción detallada de una realización preferida, pero no exclusiva, de un procedimiento y dispositivo para la implementación de un protocolo de comunicación en una unidad de control, que se muestra con la ayuda de los dibujos que se adjuntan a la presente, que son meramente ilustrativos y no limitativos, en los que:

45 La figura 1 de la técnica anterior muestra un ejemplo de diagrama de bloques del tipo basado en modelos generados en el entorno Simulink®;
la figura 2 muestra un modelo generado de acuerdo con el procedimiento que es objeto de la presente invención;
la figura 3 muestra un ejemplo de un menú gráfico principal de un dispositivo adecuado para la implementación del procedimiento de acuerdo con la presente invención;
50 la figura 4 muestra un ejemplo de una primera ventana de diálogo que permite seleccionar algunos parámetros operativos de un filtro;
la figura 5 muestra un ejemplo de una segunda interfaz gráfica que permite crear o modificar un mensaje, la configuración de los filtros que pueden aplicarse a las señales correspondientes a los datos contenidos en el mensaje importado.

55 [0019] En los dibujos, los mismos números y letras de referencia se usan para identificar los mismos elementos o componentes.

Descripción detallada de una realización preferida de la invención

60 [0020] De acuerdo con la presente invención, existe una primera etapa preliminar en la que los siguientes datos se cargan en una base de datos:

- un conjunto de filtros en un formato basado en modelos adecuados para manipular variables físicas y parámetros operativos de un dispositivo, tal como un motor, o un detector, un accionador o una unidad de aire acondicionado, etc., de acuerdo las especificaciones del fabricante o el implementador,
- un conjunto de mensajes CAN predefinidos de acuerdo con las especificaciones del fabricante o el implementador, comprendiendo cada mensaje CAN una o más palabras de bits que representan una magnitud física o un parámetro de trabajo de dicho dispositivo, motor, etc.

[0021] En una segunda etapa preliminar, dicha base de datos se carga con un primer conjunto de etiquetas, en una relación uno a uno con cada mensaje CAN, y con un segundo conjunto de etiquetas, en una relación uno a uno con cada variable o parámetro contenido en dichos mensajes CAN, y con un tercer conjunto de etiquetas, en una relación uno a uno con cada filtro en un formato basado en modelos.

[0022] Después de hacer esto, es posible:

- seleccionar un mensaje CAN, adecuado para recibirse y reconocerse por la unidad de control, en una lista de mensajes CAN de acuerdo con su etiqueta,
- ver, por medio de las etiquetas relativas, la variable o variables y/o el parámetro o parámetros operativos contenidos en dicho mensaje,
- rellamar a partir de una lista de filtros a uno o más filtros, por ejemplo, en un formato basado en modelos, de acuerdo con las etiquetas relativas,
- asignar a dicho filtro una o más entradas y uno o más parámetros operativos entre los contenidos en dicho mensaje.

[0023] Para cada mensaje es posible seleccionar los filtros que trabajan en paralelo en la misma variable o en diferentes variables contenidas en dicho mensaje.

[0024] Después, se genera un mensaje de salida, formalizado de acuerdo con las especificaciones CAN del fabricante o del implementador, comprendiendo la serialización bit a bit, concretamente la sucesión de todas las salidas de la totalidad de los filtros seleccionados configurados de acuerdo con las variables o con los parámetros operativos contenidos en el mensaje.

[0025] Además, puede añadirse de forma manual un parámetro operativo de un filtro seleccionado, o una variable en un mensaje a enviar, por el operador.

[0026] Por ejemplo, si hay tres filtros con tres salidas y el mensaje a enviar tiene cuatro variables, el valor de una de estas cuatro variables contenidas en el mensaje puede añadirse a mano.

[0027] La figura 2, que es meramente ilustrativa y no limitativa, muestra en funcionamiento de forma más clara.

[0028] La figura 2 muestra cinco ramificaciones, cada una de ellas está ocupada por un bloque de filtro (filtro), en el interior de cada una de ellas hay una entrada definida por una variable física contenida en un mensaje CAN. Las cinco ramificaciones convergen en un bloque de serialización, también denominado MUX o PACKING, donde las salidas de cada filtro se ponen en espera una después de la otra, después se disponen en serie y se incluyen en un mensaje CAN de salida.

[0029] Cada uno de dichos filtros está, al menos parcialmente, vacío, los programadores lo denominan "plantilla", concretamente es un filtro cuyos bloques y la interconexión entre ellos se define, pero en el que al menos una entrada y/o un parámetro operativo no está definido. Ventajosamente, el presente procedimiento hace más fácil y rápida la programación de una unidad de control que trabaja en una red de comunicación, por ejemplo en base al protocolo CAN, ya que es fácil crear una relación entre dichos filtros vacíos y los mensajes y, por lo tanto, implementar de forma apropiada los filtros.

[0030] Por este motivo, caso por caso, un filtro en un formato basado en modelos, definido completamente en cuanto a entradas y/o parámetros operativos, se genera automáticamente partiendo de la selección de su plantilla correspondiente y de los parámetros operativos y/o variables de entrada y/o variables operativas.

[0031] Finalmente, el mensaje CAN se genera automáticamente de acuerdo con las especificaciones impuestas por el fabricante o el implementador, por ejemplo, por medio de las bibliotecas apropiadas, de forma análoga a lo que sucede en una placa Ethernet.

[0032] En resumen, el procedimiento de acuerdo con la presente invención puede aplicarse en un sistema que comprende al menos una unidad de procesamiento, y comprende las siguientes etapas:

- una primera selección de un primer mensaje de una lista de mensajes, adecuado para recibirse y procesarse por la unidad de control, conteniendo este mensaje al menos un parámetro y/o un valor de una magnitud física,
- 5 - una visualización de la variable o variables y/o un parámetro o parámetros operativos contenidos en dicho primer mensaje,
- una segunda selección de uno o más filtros en un formato basado en modelos de una lista de filtros, comprendiendo dicho filtro al menos una entrada y/o un parámetro operativo,
- una asignación a dicha entrada y/o parámetro operativo de dicho parámetro o valor de magnitud física contenido en dicho mensaje,
- 10 - una generación de un segundo mensaje, adecuado para enviarse por la unidad de control, que comprende al menos una salida de dicho filtro, como resultado de la operación realizada por dicho filtro, estando dichas listas memorizadas en una base de datos.

15 **[0033]** Estas etapas pueden realizarse fácilmente, puesto que cada selección y manipulación está guiada por las etiquetas descriptivas con respecto a cada mensaje y a cada filtro y/o variable/parámetro contenido en dicho mensaje.

20 **[0034]** Además, la asignación de la entrada/parámetros operativos de cada filtro puede realizarse fácilmente por medio de los menús desplegables que se generan automáticamente de acuerdo con el número de variables disponible en el espacio de trabajo.

25 **[0035]** Por lo tanto, en el espacio de trabajo están presentes todas las variables en relación a uno o más mensajes seleccionados, y también las variables definidas por el operario, si las hubiera, y las variables y parámetros generados por el sistema operativo de la propia unidad de control.

30 **[0036]** Por lo tanto, la selección de un mensaje de una base de datos pre-existente permite traducir los parámetros contenidos en éste en un lenguaje de alto nivel, de manera que puedan entenderse por un operador.

35 **[0037]** Por lo tanto, el procedimiento de importación es adecuado para convertir una palabra de bits contenida en un mensaje, por ejemplo un mensaje CAN, de acuerdo con su posición en el mensaje y al tipo de mensaje, en un parámetro operativo o en una variable en relación a una magnitud física, con el fin de gestionar y controlar, por ejemplo, el estado de un accionador o detector particular, o al menos a un valor análogo específico en relación con una magnitud física, tal como la temperatura o la presión.

40 **[0038]** En particular, el mensaje, compuesto por un conjunto de palabras de bits, se representa, por ejemplo en una pantalla, como un conjunto de una o más variables representadas por sus etiquetas relativas que tienen un significado explicativo, lo que hace más comprensible la naturaleza del mensaje y su contexto de uso.

45 **[0039]** Por ejemplo, en el caso de la programación de una unidad de control para la gestión de un motor endotérmico, la palabra de bits contenida en un mensaje CAN que se refiere a la temperatura del motor se muestra automáticamente en la pantalla con un campo editable o no editable y con una etiqueta descriptiva de su significado.

50 **[0040]** Por lo tanto, si hay una primera base de datos que contiene la lista de los mensajes y sus especificaciones relativas usadas por un fabricante para el intercambio de datos en los sistemas, entonces es posible seleccionar el mensaje, por ejemplo, de un primer menú desplegable. Después de eso, los campos editables o no editables se muestran con sus etiquetas descriptivas relativas que se refieren a todas las palabras de bits contenidas en el mensaje. Dicha etapa preliminar para cargar los mensajes CAN en dicha base de datos puede realizarse por medio de un analizador de mensajes, por ejemplo, uno denominado analizador CAN, que usa el protocolo CAN, para crear dicha base de datos.

55 **[0041]** Sin embargo, puede crearse un nuevo mensaje en cualquier momento por el operador por medio de una interfaz gráfica apropiada.

60 **[0042]** La figura 3 muestra una interfaz de trabajo, en la que arriba a la izquierda hay un botón "Crear marco de mensaje" que abre una segunda interfaz que se muestra en la figura 4, la cual permite crear un nuevo mensaje.

[0043] Por lo tanto, para programar una unidad de control, siempre de forma gráfica, de manera que se comporte de cierto modo cuando llega un determinado mensaje, es posible seleccionar, por ejemplo por medio de un segundo menú desplegable, al menos un filtro, para manipular y filtrar la información contenida en el mensaje recibido y/o para añadir el resultado de un hilo a nivel de aplicación implementado por la unidad de control.

[0044] Las características de cada filtro se especifican por medio de un diagrama de bloques relacionado del tipo basado en modelos, por ejemplo en el entorno Simulink®. Los diagramas de bloques están contenidos en una

segunda base de datos pre-existente y están asociados a una etiqueta. Por este motivo, la generación automática del modelo basado en modelos comprende la configuración de los parámetros operativos del modelo con respecto al filtro seleccionado por medio de una ventana de diálogo, que comprende las etiquetas que explican cada parámetro.

5 **[0045]** Por lo tanto, la etapa de generación automática del modelo del tipo basado en modelos comprende la utilización de los parámetros operativos de un modelo pre-cargado en la segunda base de datos.

10 **[0046]** Por lo tanto, la generación automática comprende la aplicación de parámetros predeterminados a un modelo del tipo basado en modelos asociado a un filtro que opera de acuerdo con las variables y los parámetros que se han fijado previamente.

15 **[0047]** El acoplamiento entre un mensaje recibido y al menos un filtro puede ser automático asociando en dicha base de datos un mensaje a al menos un filtro y a un filtro al menos un mensaje y/o dejando que el operador elija libremente.

20 **[0048]** Con respecto al ejemplo de un dispositivo para la aplicación del procedimiento, con referencia a la figura 3, una primera interfaz gráfica comprende un botón de comando denominado "Importar Datos de BD" que permite la activación del procedimiento de importación de datos. Dicho comando activa una ventana de diálogo, como se muestra en la figura 4, que contiene algunos botones de selección que pueden usarse para seleccionar los mensajes a importar, tanto los mensajes recibidos como transmitidos.

[0049] Una vez que se completa la selección, la primera interfaz gráfica se mostrará de nuevo, permitiendo realizar las siguientes operaciones:

- 25 - la entrada manual de datos en un campo del mensaje seleccionado por medio de un teclado o en general por medio de interfaces hombre-máquina;
- 30 - la selección de un filtro aplicable al mensaje, por medio de un botón de comando denominado "Abrir Biblioteca de Filtros de Señal" que abre una segunda ventana de diálogo, mostrada en el óvalo en la figura 3, denominada "Seleccionar un Filtro".

[0050] La primera interfaz gráfica comprende un botón de comando adicional denominado "Config. Filtro" que activa una segunda interfaz gráfica mostrada en la figura 5, que permite configurar, uno a uno, los filtros aplicables a las señales correspondientes a los datos contenidos en el mensaje importado.

35 **[0051]** Finalmente, la segunda ventana de diálogo comprende un botón de comando adicional denominado "Aplicar Filtro" que permite generar automáticamente un modelo basado en modelos en el entorno Simulink®, equivalente al mensaje a enviar, de forma análoga a lo que se muestra en la figura 2.

40 **[0052]** Después, es posible convertir en lenguaje C, u otro lenguaje de programación equivalente, el modelo generado.

45 **[0053]** De forma ventajosa, es posible añadir a la base de datos no sólo las etiquetas sino también las descripciones detalladas de los filtros vacíos (plantillas), por ejemplo nombres y cantidades en relación a los parámetros operativos o a las entradas, por lo tanto, dichas descripciones sirven como borradores o plantillas predefinidas. De esta forma, cuando el operador asigna un parámetro o una entrada a un filtro usando el presente dispositivo, automáticamente modifica el borrador de la descripción del filtro, reemplazando un nombre genérico, por ejemplo, VAR1, ya presente en el borrador, con la etiqueta en relación a la variable asignada al filtro, por ejemplo TEMPERATURA, en cualquier momento el operador se refiere a una entrada o a un parámetro operativo del filtro vacío.

50 **[0054]** Por lo tanto, es posible generar automáticamente un documento técnico que tiene una estructura coherente con las operaciones que el operador realiza, ya que dichos campos vacíos se mejoran, una y otra vez, con la información específica añadida en las primeras tres etapas del procedimiento.

55 **[0055]** Por este motivo, cuando un operador crea un nuevo filtro, el sistema le obliga a escribir una descripción detallada del filtro, asignando nombres genéricos a las entradas, por ejemplo, VAR1, VAR2, etc.. Cuando una variable o parámetro de entrada o parámetro operativo se asigna al filtro, reemplaza todos los casos de VAR1, por ejemplo con TEMPERATURA.

60 **[0056]** De este modo es posible generar un documento técnico formateado de acuerdo con un esquema convencional y hacer referencia a mensajes y filtros contenidos en la base de datos respectiva y, en algunos casos, acompañados por su documentación respectiva. Dicho documento es perfectamente comprensible por un ingeniero de sistemas o por un ingeniero de software.

5 **[0057]** De forma ventajosa, ya que el proceso de programación puede realizarse con el software de aplicación Simulink®, soportado por el entorno Matlab®, es posible, gracias a las funciones de Matlab®, obtener ficheros, por ejemplo en lenguaje C, de los diagramas de flujo que se han mencionado anteriormente. Esto elimina cualquier posibilidad de error de programación humano, y reduce considerablemente la actividad del ingeniero de software, conservando la calidad del código de programación muy alta.

10 **[0058]** Además, puesto que se realiza una programación de tan alto nivel por medio de un entorno gráfico basado en modelos, es posible compilar el resultado de la programación en lenguaje C, o en lenguaje máquina o en otro lenguaje, por medio de las bibliotecas específicas, también denominadas API, proporcionadas por el fabricante de la unidad de control. De este modo, cuando el tipo de unidad de control cambia, las operaciones para adaptar el programa se reducen extremadamente.

15 **[0059]** El presente procedimiento y el dispositivo relacionado pueden emplearse de forma ventajosa en cualquier entorno de desarrollo similar a Matlab® y Simulink®, además, el programa de comunicación implementado puede ser cualquiera.

20 **[0060]** En particular, la presente invención puede aplicarse en el campo aeroespacial, donde el protocolo de comunicación entre las unidades de procesamiento es el identificado por el número 1553.

25 **[0061]** Por lo tanto, la presente invención puede incorporarse ventajosamente por medio de un programa informático, que comprende medios de código de programa que realizan una o más etapas de dicho procedimiento, cuando dicho programa se ejecuta en un ordenador. Por este motivo, el alcance de la presente patente pretende cubrir también dicho programa informático y el medio legible por ordenador que comprende un mensaje grabado, comprendiendo dicho medio legible por ordenador los medios de código de programa para realizar una o más etapas de dicho procedimiento, cuando dicho programa se ejecuta en un ordenador.

30 **[0062]** Será evidente para el experto en la técnica que pueden concebirse y llevarse a la práctica otras realizaciones alternativas y equivalentes de la invención sin apartarse del alcance de la invención.

[0063] A partir de la descripción que se ha expuesto anteriormente será posible para el experto en la técnica desarrollar la invención sin la necesidad de describir detalles de construcción adicionales.

REIVINDICACIONES

- 5 **1.** Procedimiento para programar un procedimiento de alto nivel de una unidad de control de un vehículo, comprendiendo el vehículo una red CAN y un dispositivo, recibiendo y procesando la unidad de control primeros mensajes CAN y generando y enviando segundos mensajes CAN, comprendiendo el procedimiento las siguientes etapas:
- 10 - asociación de un primer mensaje CAN contenido en una lista de mensajes CAN a una primera etiqueta, conteniendo dicho primer mensaje CAN un parámetro y/o un valor de una magnitud física de dicho dispositivo,
 - asociación de un filtro en un formato basado en modelos contenido en una lista de filtros a una segunda etiqueta,
 - asociación de cada valor relativo a una magnitud física o parámetro operativo de dicho dispositivo a una tercera etiqueta,
 - 15 - visualización de dicha primera, segunda y tercera etiquetas,
 - primera selección de un primer mensaje CAN de dicha lista de mensajes CAN para recibirse y procesarse por la unidad de control,
 - segunda selección de uno o más filtros en un formato basado en modelos de dicha lista de filtros, comprendiendo dicho uno o más filtros una entrada y/o un segundo parámetro operativo,
 - 20 - asignación de dicho primer parámetro o valor del magnitud física contenido en dicho primer mensaje CAN a dicha entrada y/o segundo parámetro operativo del filtro,
 - generación de un segundo mensaje CAN para enviarse por la unidad de control, comprendiendo al menos una salida de dicho filtro como resultado de la operación realizada por dicho filtro,
 - serialización bit a bit como una sucesión de todas las salidas de la totalidad de uno o más filtros seleccionados,
 - 25
- estando dichas listas y dichas etiquetas almacenadas en una base de datos.
- 30 **2.** Procedimiento de acuerdo con la reivindicación 1, que comprende adicionalmente la etapa de generar una equivalencia de lenguaje de programación con respecto a dicho filtro en formato basado en modelos y compilar dicha equivalencia de lenguaje de programación para ejecutarse en dicha unidad de control por medio de bibliotecas proporcionadas por el fabricante de la unidad de control.
- 35 **3.** Procedimiento de acuerdo con las reivindicaciones anteriores, que comprende adicionalmente una etapa preliminar de introducir dichos filtros y dichos mensajes CAN en dicha base de datos.
- 4.** Procedimiento de acuerdo con las reivindicaciones anteriores, que comprende adicionalmente una etapa preliminar de introducir dichos primeros mensajes CAN y dichos filtros en dicha base de datos.
- 40 **5.** Procedimiento de acuerdo con la reivindicación 4, en el que dicha introducción de dichos primeros mensajes CAN se realiza por medio de un analizador CAN.
- 45 **6.** Procedimiento de acuerdo con las reivindicaciones anteriores, en el que una relación 1 a 1 o una relación 1 a muchos entre un primer mensaje CAN y uno o más filtros se almacena, de manera que dicha segunda selección pueda realizarse en un subconjunto de dicha lista de filtros.
- 7.** Procedimiento de acuerdo con las reivindicaciones anteriores, en el que al menos una información en dicho segundo mensaje CAN se añade de forma manual.
- 50 **8.** Procedimiento de acuerdo con las reivindicaciones anteriores, que comprende adicionalmente una etapa de introducir en dicha base de datos una descripción técnica para cada filtro, comprendiendo al menos un nombre genérico de una entrada.
- 55 **9.** Procedimiento de acuerdo con la reivindicación 8, en el que dicho nombre genérico de una entrada se reemplaza por la etiqueta de una variable o de un parámetro contenido en dicho primer mensaje CAN, cuando dicha variable se asigna a la entrada del filtro, a la que dicho nombre genérico se refiere.
- 60 **10.** Dispositivo para programar un procedimiento de alto nivel de una unidad de control de un vehículo que comprende un dispositivo, recibiendo y procesando la unidad de control de primeros mensajes CAN y generando y enviando segundos mensajes CAN, comprendiendo medios para realizar el procedimiento de acuerdo con cualquiera de las reivindicaciones anteriores.
- 11.** Dispositivo de acuerdo con la reivindicación 10, en el que dichos medios comprenden al menos una base de

datos, una interfaz gráfica dotada de medios de entrada de datos, medios de comando, medios de selección, permitiendo la selección de al menos un mensaje CAN de entrada, el filtrado de dicho mensaje CAN de acuerdo con al menos una especificación funcional, la generación automática de una equivalencia del modelo de bloques con respecto al segundo mensaje CAN que se enviará y la conversión automática del modelo en lenguaje de programación.

- 5
- 12.** Programa informático que comprende medios de código de programa adecuados para realizar las etapas de cualquier reivindicación de 1 a 9, cuando dicho programa está funcionando en un ordenador.
- 10
- 13.** Medios legibles por ordenador que comprenden un programa grabado, comprendiendo dichos medios legibles por ordenador medios de código de programa adecuados para realizar las etapas de acuerdo con las reivindicaciones de 1 a 9, cuando dicho programa está funcionando en un ordenador.

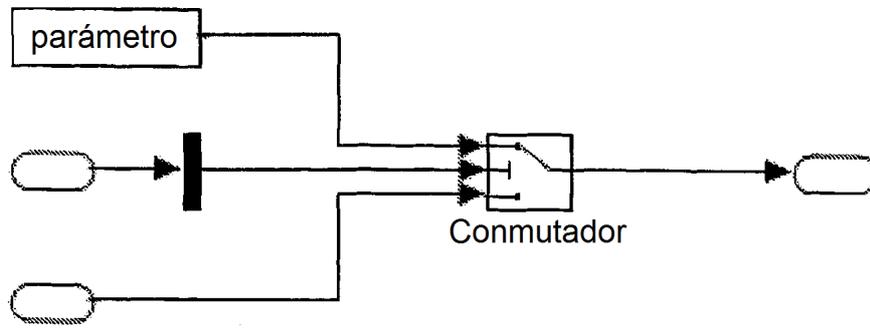


Fig. 1

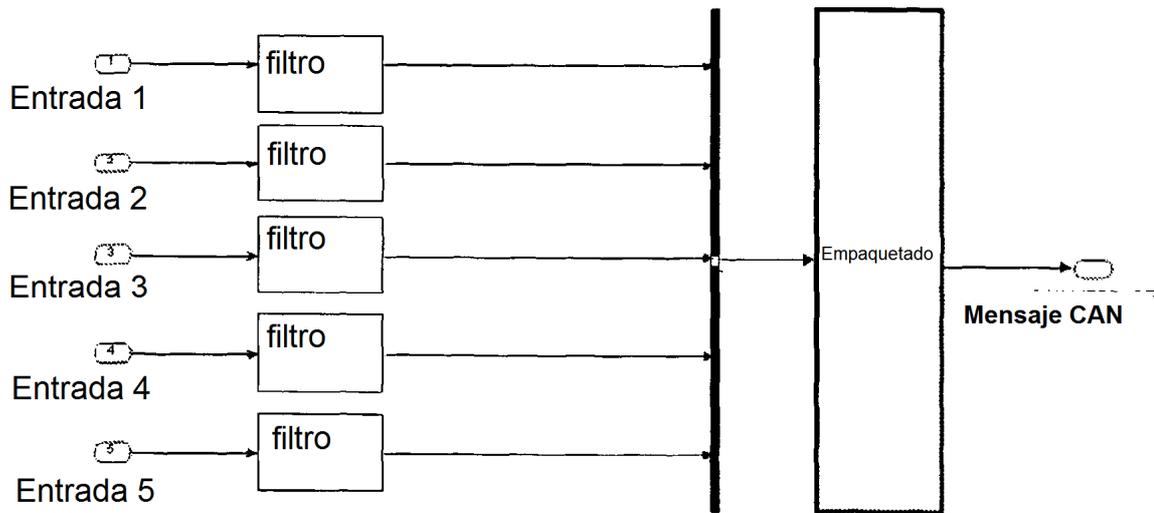


Fig. 2

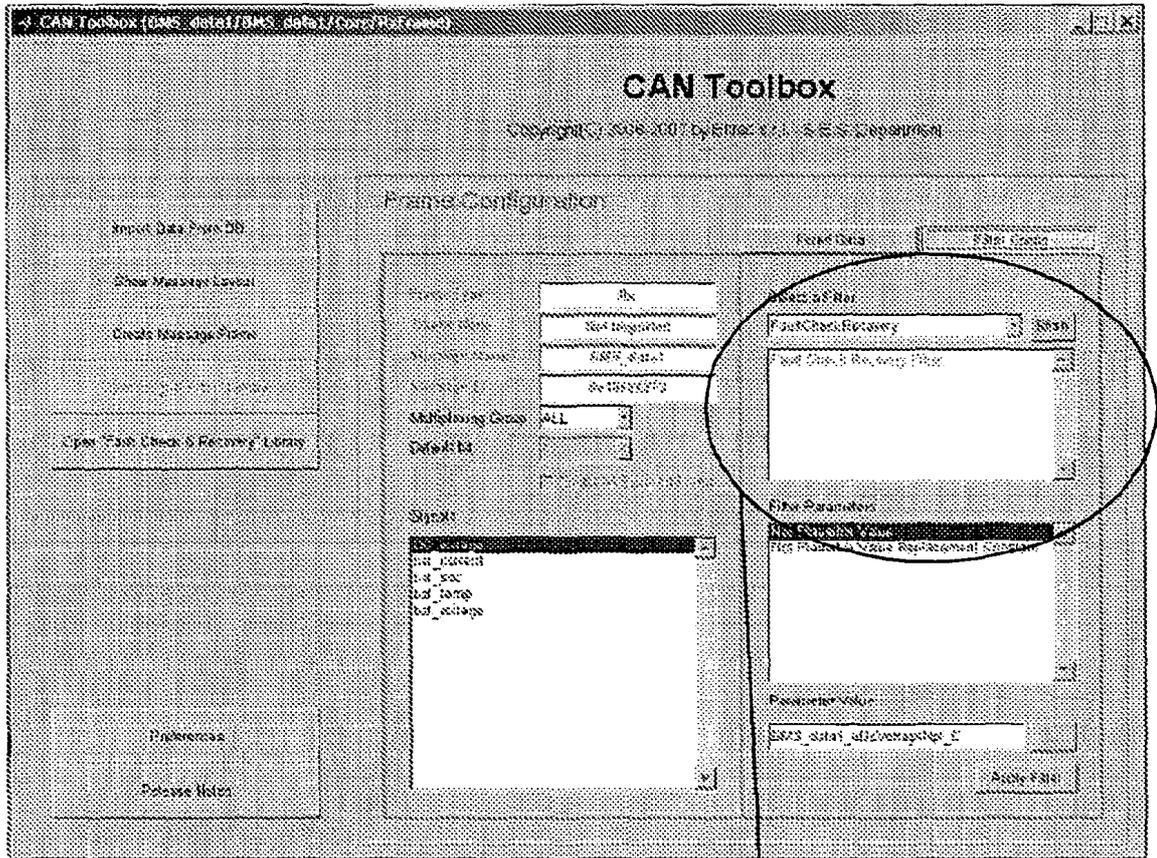


Fig. 3

Seleccionar un Filtro

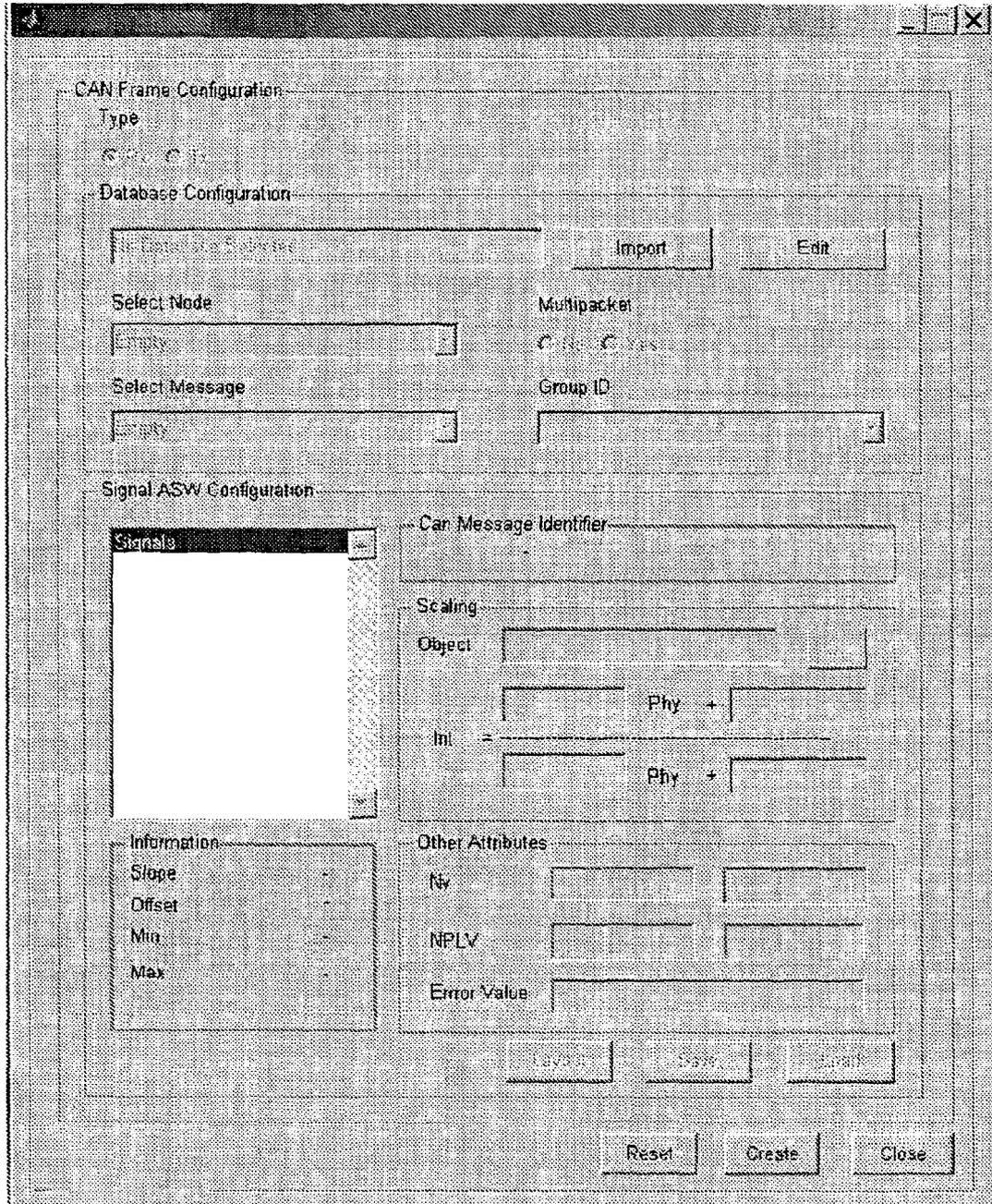


Fig. 4

Input configuration:

Object Configuration

Action Mode

Select Existing Object
 Update Current Object
 Create New Object
 Reset Info

Existing Object

Object Info (edit mode)

Name	VehicleSpeed	Object Type	TYPE_SIGNAL_INPUT
Source	SPLM	Responsibility	
Description	vehicle speed		
Phys Val	0	Phys Min	0
		Phys Max	300
Precision	0.1	Units	Km/h
Storage	Default		...
Scaling	RConv_VehicleSpeed		...

Fig. 5