

Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

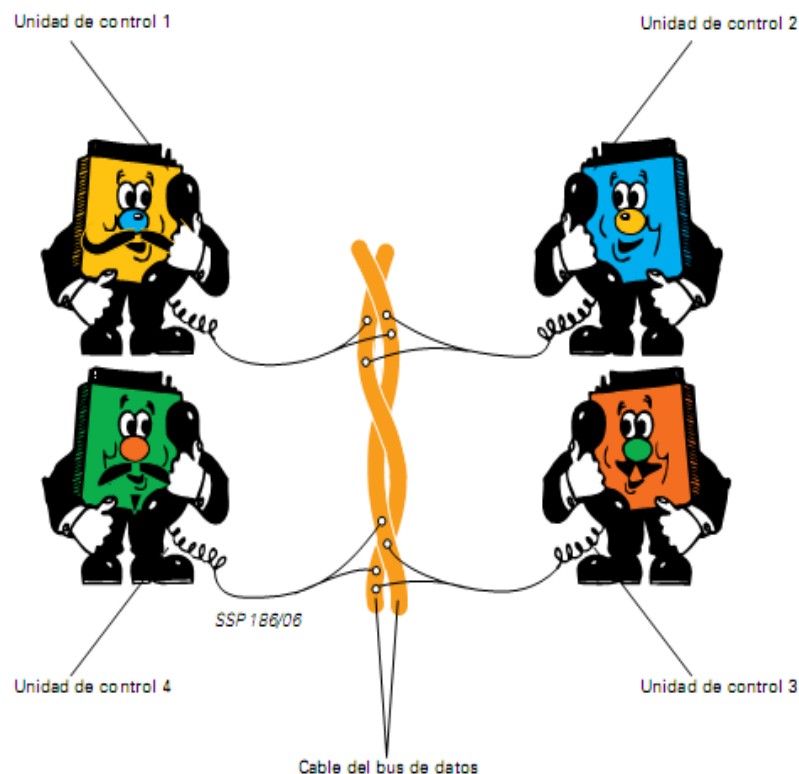
CAN BUS

Introducción a los sistemas de comunicación del vehículo

CAN (*Controller Area Network*), es un protocolo de comunicaciones desarrollado por la firma alemana Robert Bosch GmbH, basado en una topología bus para la transmisión de mensajes en ambientes distribuidos, además ofrece una solución a la gestión de la comunicación entre múltiples CPUs (unidades centrales de proceso).

El principio de la transmisión de datos

La transmisión de datos a través del CAN-Bus funciona de un modo parecido al de una conferencia telefónica. Una unidad de control modula sus datos, introduciéndolos en la red, mientras que los demás coescuchan estos datos. Para ciertos abonados resultan interesantes estos datos, en virtud de lo cual los utilizan. A otras unidades de control pueden no interesarles esos datos específicos.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

El protocolo de comunicaciones CAN proporciona los siguientes beneficios:

- Es un protocolo de comunicaciones normalizado, con lo que se simplifica y economiza la tarea de comunicar subsistemas de diferentes fabricantes sobre una red común o bus.
- El procesador anfitrión (*host*) delega la carga de comunicaciones a un periférico inteligente, por lo tanto el procesador anfitrión dispone de mayor tiempo para ejecutar sus propias tareas.
- Al ser una red multiplexada, reduce considerablemente el cableado y elimina las conexiones punto a punto.
- Para simplificar aun más la electrónica del coche se puede utilizar una subred más simple, que se conecta a la red CAN, llamada LIN.

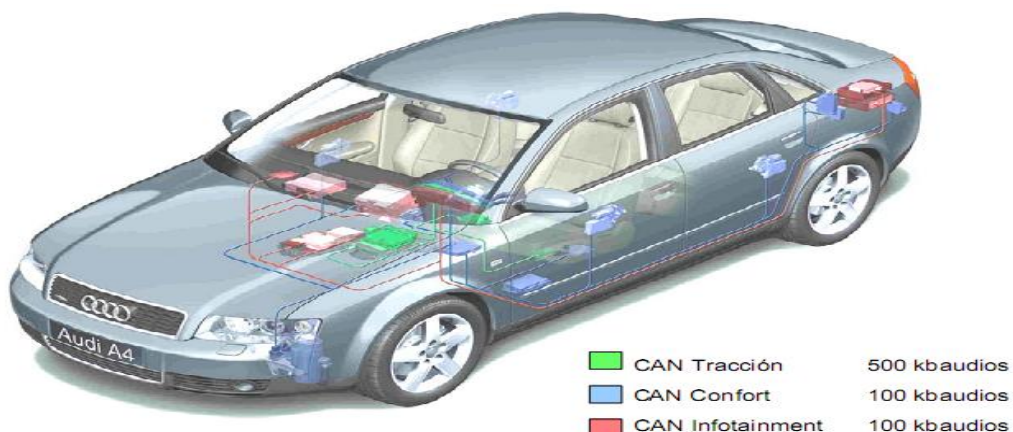
Cuanto mayor es la cantidad de información que recibe una unidad de control acerca del estado operativo del sistema global, tanto mejor puede ajustar al conjunto sus funciones específicas.

En el área de la tracción forman un sistema global:

- La unidad de control del motor
- La unidad de control para cambio automático
- La unidad de control ABS

En el área de confort constituyen un sistema global:

- La unidad de control central
- Las unidades de control de puertas



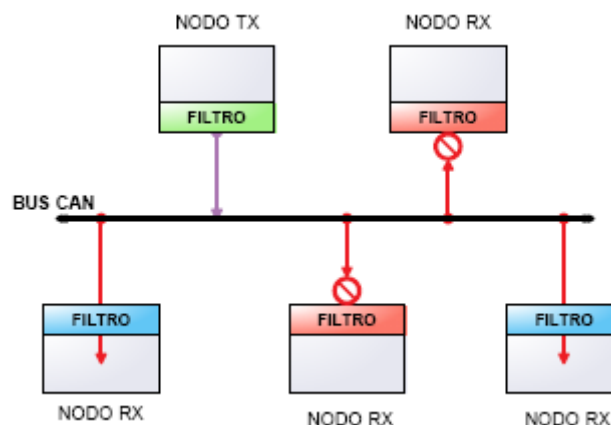
Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Principales características de CAN

CAN se basa en el modelo productor/consumidor, el cual es un concepto, o paradigma de comunicaciones de datos, que describe una relación entre un productor y uno o más consumidores. CAN es un protocolo orientado a mensajes, es decir la información que se va a intercambiar se descompone en mensajes, a los cuales se les asigna un identificador y se encapsulan en tramas para su transmisión. Cada mensaje tiene un identificador único dentro de la red, con el cual los nodos deciden aceptar o no dicho mensaje.

Dentro de sus principales características se encuentran:

- La comunicación está basada en mensajes y no en direcciones.
- Un mensaje es diferenciado por el campo llamado identificador, que no indica el destino del mensaje, pero sí describe el contenido del mismo.
- No hay un sistema de direccionamiento de los nodos en el sentido convencional. Los mensajes se envían según su prioridad.
- La prioridad entre los mensajes la define el identificador. Se trata de una prioridad para el acceso al bus.
- Es un sistema multimaestro. Cuando el bus está libre, cualquier nodo puede empezar la transmisión de un mensaje, y el mensaje con mayor prioridad gana la arbitración del bus.
- Todos los nodos CAN son capaces de transmitir y recibir datos y varios pueden acceder al bus de datos simultáneamente.
- Un nodo emisor envía el mensaje a todos los nodos de la red, cada nodo, según el identificador del mensaje, lo filtra y decide si debe procesarlo inmediatamente o descartarlo. Como consecuencia el sistema se convierte en multicast en el cual un mensaje puede estar dirigido a varios nodos al mismo tiempo.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

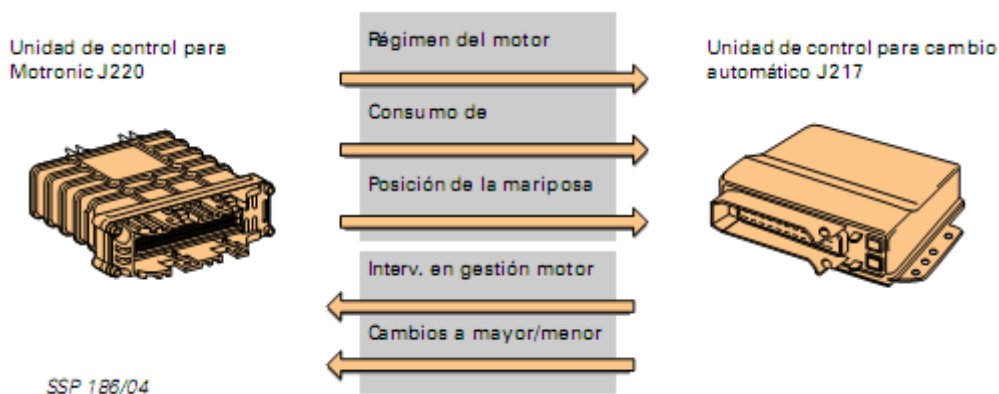
- Gran fiabilidad y robustez en la transmisión. Detecta errores, los señala, envía mensaje de error y reenviará el mensaje corrupto una vez que el bus vuelva a estar activo. Además puede operar en ambientes con condiciones extremas de ruido e interferencias gracias a que es un bus diferencial.
- Es un protocolo de comunicaciones normalizado, con lo que se simplifica y economiza la tarea de comunicar subsistemas de diferentes fabricantes sobre una red común o bus.
- Al ser una red multiplexada, reduce considerablemente el cableado y elimina las conexiones punto a punto.
- El procesador principal delega la carga de comunicaciones a un periférico inteligente (controlador), por lo tanto el procesador principal dispone de mayor tiempo para ejecutar sus propias tareas.

Transmisión de datos

En este momento las posibilidades que existen en el automóvil para una adecuada transmisión de datos o información son dos:

1. Cada información se intercambia a través de un cable propio.

La siguiente figura ilustra esta posibilidad, en la que cada información se transmite a través de un cable propio. En este caso se necesitarían 5 cables.



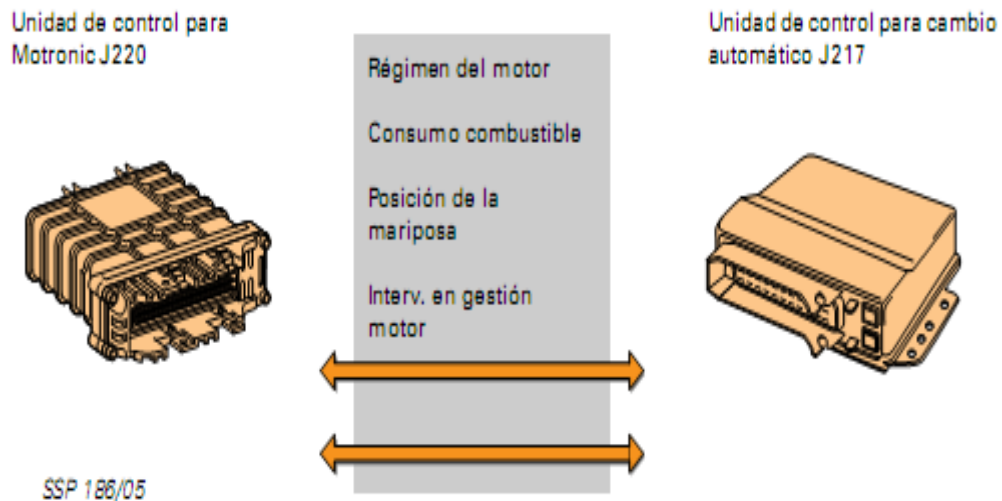
Como pueden ver para cada información se necesita un cable propio. Por lo tanto, con cada información adicional crece también la cantidad de cables y pines en las unidades de control. Por ese motivo, este tipo de transmisión de datos sólo es practicable con una cantidad limitada de informaciones a intercambiar.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

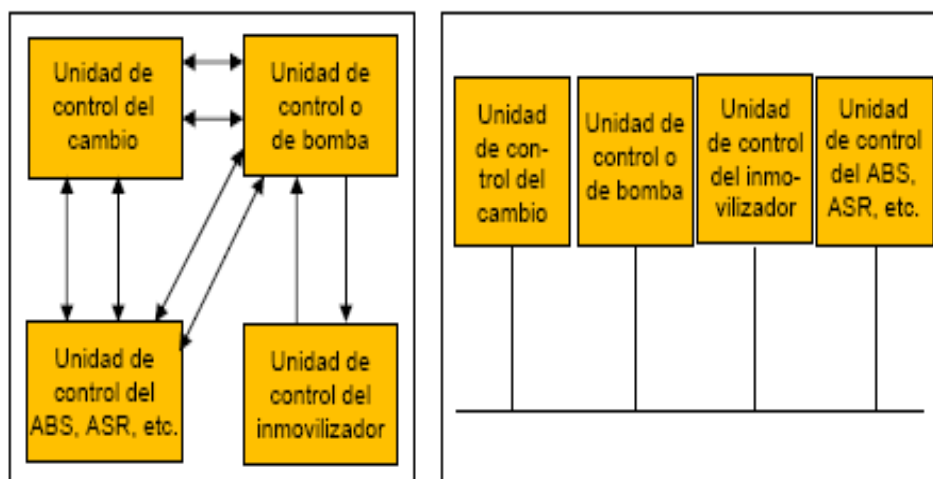
2. Toda la información se intercambia a través de dos cables como máximo, que constituyen el CAN-Bus entre las unidades de control.

Con el CAN-Bus se transmite toda la información a través de dos cables. En ambos cables bidireccionales del CAN-Bus se transmiten los mismos datos.



Con este tipo de transmisión de datos se maneja toda la información a través de dos cables. Independientemente de la cantidad de unidades de control abonadas y de la cantidad de información transmitida. Por ese motivo es conveniente transmitir los datos con un CAN-Bus cuando se intercambia una gran cantidad de información entre las unidades de control.

Transmisión convencional vs. Transmisión en serie

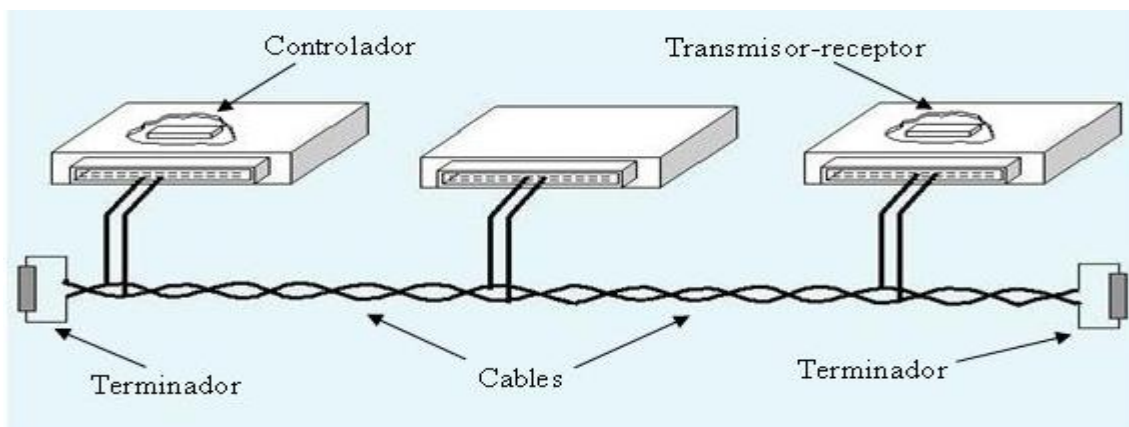


Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Componentes del CAN BUS

Consta de un controlador, un transceptor, dos elementos finales del bus y dos cables para la transmisión de datos. Con excepción de los cables del bus, todos los componentes están alojados en las unidades de control. En el funcionamiento conocido de las unidades de control no se ha modificado nada.

- **Controlador CAN** Recibe del microprocesador, en la unidad de control, los datos que han de ser transmitidos. Los acondiciona y los pasa al transceptor CAN. Asimismo recibe los datos procedentes del transceptor CAN, los acondiciona asimismo y los pasa al microprocesador en la unidad de control.
- **Transceptor CAN (Transmisor-receptor)** Es un transmisor y un receptor. Transforma los datos del controlador CAN en señales eléctricas y transmite éstas sobre los cables del CAN-Bus. Asimismo recibe los datos y los transforma para el controlador CAN.
- **Elemento final del bus de datos (Terminadores)** Es una resistencia (Casi siempre de 120Ω , aunque las resistencias de terminación no necesariamente son de aprox. 120Ω , sino que su ohmiaje depende de la topología específica del ramal de cables). Evita que los datos transmitidos sean devueltos en forma de eco de los extremos de los cables y que se falsifiquen los datos.
- **Cables del bus de datos** Funcionan de forma bidireccional y sirven para la transmisión de los datos. Se denominan con las designaciones CAN-High (señales de nivel lógico alto) y CAN-Low (señales de nivel lógico bajo).



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II**Tipos de implementación**

Teniendo en cuenta los elementos básicos que conforman el bus CAN, existen tres tipos de implementación donde la comunicación es igual para todas. La diferencia radica en los filtros de aceptación, en la capacidad de almacenamiento de las tramas, en la responsabilidad que asume el microcontrolador o el controlador CAN, etc., es decir, en el hardware del nodo. Las tres implementaciones incorporan un microcontrolador, ya que este componente representa una herramienta de hardware ideal para el desarrollo de aplicaciones con conexión CAN. Las tres distintas implementaciones son:

Basic CAN

En esta implementación existe un vínculo muy fuerte entre el controlador CAN y su microcontrolador asociado. El microcontrolador será interrumpido para tratar cada mensaje CAN que reciba. Aquí, el controlador CAN está restringido a un único buffer de mensajes. El microcontrolador es quién lleva el peso de las tareas haciendo así que el controlador CAN sea más simple y por tanto más barato. Este método es bueno para nodos encargados de manejar informaciones esporádicas, disminuyendo la ocupación del bus. Es la arquitectura más simple.

Full CAN

En este caso, el controlador CAN tiene varios buffers. Además tiene la capacidad para filtrar los tipos de mensaje que desee y puede transmitir y recibir mensajes sin ayuda del microcontrolador. En definitiva, el controlador le reduce la carga al microcontrolador. También se pueden habilitar interrupciones en el microcontrolador para notificarle la llegada de un mensaje. Este tipo de arquitectura consiste en un microcontrolador que incluya, no sólo sus características propias sino además un módulo CAN con las características de un microcontrolador CAN. El transceiver se sitúa de manera separada.

Serial Linked I/O

Los dispositivos Link Input/Output (SLIOS) son dispositivos de bajo coste y baja inteligencia. Son controladores sin capacidad de programación. Son interfaces preconfigurados que requieren de un nodo CAN programable para controlarlo y son usados para salidas y entradas lejanas del bus. Son dispositivos esclavos físicamente direccionados con jumpers o con switches DIP.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Protocolo de comunicaciones CAN

CAN fue desarrollado, inicialmente para aplicaciones en los automóviles y por lo tanto la plataforma del protocolo es resultado de las necesidades existentes en el área de la automoción. La Organización Internacional para la Estandarización (ISO, *International Organization for Standardization*) define dos tipos de redes CAN: una red de alta velocidad (hasta 1 Mbps), bajo el estándar ISO 11898-2, destinada para controlar el motor e interconectar la unidades de control electrónico (ECU); y una red de baja velocidad tolerante a fallos (menor o igual a 125 Kbps), bajo el estándar ISO 11519-2/ISO 11898-3, dedicada a la comunicación de los dispositivos electrónicos internos de un automóvil como son control de puertas, techo corredizo, luces y asientos. CAN es un protocolo de comunicaciones serie que soporta control distribuido en tiempo real con un alto nivel de seguridad y multiplexación.

El establecimiento de una red CAN para interconectar los dispositivos electrónicos internos de un vehículo tiene la finalidad de sustituir o eliminar el cableado. Las ECUs, sensores, sistemas antideslizantes, etc. se conectan mediante una red CAN a velocidades de transferencia de datos de hasta 1 Mbps.

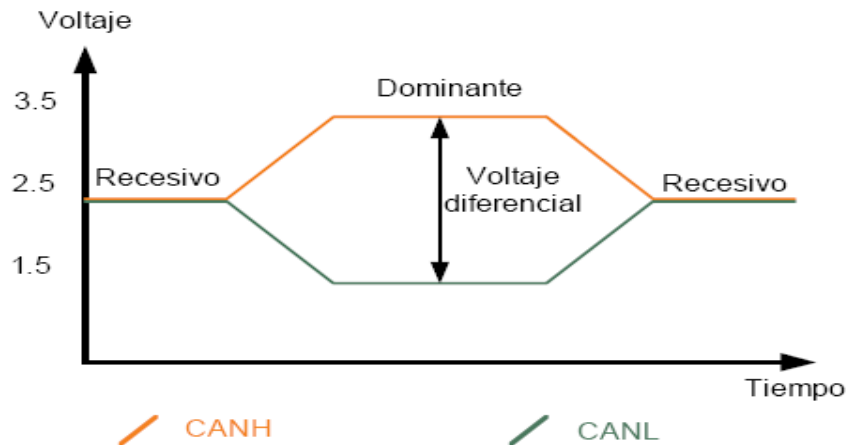
De todas las características eléctricas que define la capa física, es importante conocer los denominados niveles lógicos del bus. Al tratarse de un bus diferencial, éste está formado por dos señales y la diferencia que existe entre estas dos señales determinan el estado del bus. Por tanto, el CAN dispone de dos niveles lógicos. Normalmente en los sistemas digitales de dos niveles se conocen estos dos estados por nivel alto y nivel bajo, sin embargo en este caso se denominan nivel dominante y nivel recesivo:

Dominante. La tensión diferencial entre los pines de comunicación (CAN_H - CAN_L) ha de ser del orden de 2 V. Para conseguir esto es necesario que CAN_H tenga 3,5 V y CAN_L sea de 1,5 V (nominales). De hecho, si el voltaje de la línea CAN_H es al menos 0,9 V mayor que CAN_L, entonces ya se detectará la condición de bit dominante.

Recesivo. La tensión diferencial entre los pines de comunicación (CAN_H - CAN_L) ha de ser del orden de 0 V. Para conseguir esto es necesario que CAN_H y CAN_L tengan 2,5 V (nominales). Aunque realmente el bus detectará una condición de recesivo si el voltaje de la línea CAN_H no es más alto que el voltaje de la línea CAN_L más 0,5 V.

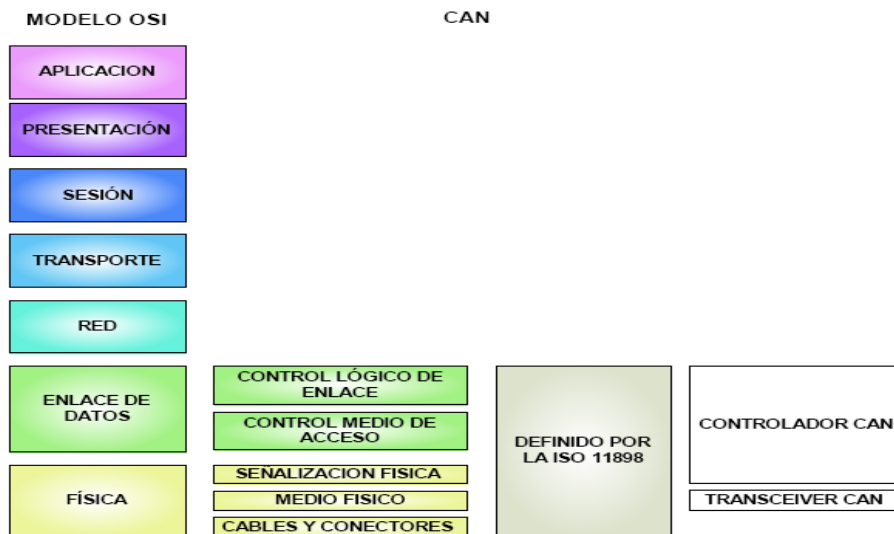


Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II



De acuerdo al modelo de referencia OSI (*Open Systems Interconnection*), la arquitectura de protocolos CAN incluye tres capas: **física**, **de enlace de datos** y **aplicación**, además de una capa especial para gestión y control del nodo llamada **capa de supervisor**.

Relación entre el Modelo OSI y CAN



Capa física: define los aspectos del medio físico para la transmisión de datos entre nodos de una red CAN, los más importantes son niveles de señal, representación, sincronización y tiempos en los que los bits se transfieren al bus. La especificación del protocolo CAN no define una capa física, sin embargo, los estándares ISO 11898 establecen las características que deben cumplir las aplicaciones para la transferencia en alta y baja velocidad.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Capa de enlace de datos: define las tareas independientes del método de acceso al medio, además debido a que una red CAN brinda soporte para procesamiento en tiempo real a todos los sistemas que la integran, el intercambio de mensajes que demanda dicho procesamiento requiere de un sistema de transmisión a frecuencias altas y retrasos mínimos. En redes multimaestro, la técnica de acceso al medio es muy importante ya que todo nodo activo tiene los derechos para controlar la red y acaparar los recursos. Por lo tanto la capa de enlace de datos define el método de acceso al medio así como los tipos de tramas para el envío de mensajes

Cuando un nodo necesita enviar información a través de una red CAN, puede ocurrir que varios nodos intenten transmitir simultáneamente. CAN resuelve lo anterior al asignar prioridades mediante el identificador de cada mensaje, donde dicha asignación se realiza durante el diseño del sistema en forma de números binarios y no puede modificarse dinámicamente. El identificador con el menor número binario es el que tiene mayor prioridad.

El método de acceso al medio utilizado es el de Acceso Múltiple por Detección de Portadora, con Detección de Colisiones y Arbitraje por Prioridad de Mensaje (CSMA/CD+AMP, *Carrier Sense Multiple Access with Collision Detection and Arbitration Message Priority*). De acuerdo con este método, los nodos en la red que necesitan transmitir información deben esperar a que el bus esté libre (detección de portadora); cuando se cumple esta condición, dichos nodos transmiten un bit de inicio (acceso múltiple). Cada nodo lee el bus bit a bit durante la transmisión de la trama y comparan el valor transmitido con el valor recibido; mientras los valores sean idénticos, el nodo continúa con la transmisión; si se detecta una diferencia en los valores de los bits, se lleva a cabo el mecanismo de arbitraje.

CAN establece dos formatos de tramas de datos (*data frame*) que difieren en la longitud del campo del identificador, las tramas estándares (*standard frame*) con un identificador de 11 bits definidas en la especificación CAN 2.0A, y las tramas extendidas (*extended frame*) con un identificador de 29 bits definidas en la especificación CAN 2.0B.

Para la transmisión y control de mensajes CAN, se definen cuatro tipos de tramas: de datos, remota (*remote frame*), de error (*error frame*) y de sobrecarga (*overload frame*). Las tramas remotas también se establecen en ambos formatos, estándar y extendido, y tanto las tramas de datos como las remotas se separan de tramas precedentes mediante espacios entre tramas (*interframe space*).

En cuanto a la detección y manejo de errores, un controlador CAN cuenta con la capacidad de detectar y manejar los errores que surjan en una red.





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Todo error detectado por un nodo, se notifica inmediatamente al resto de los nodos.

Capa de supervisor: La sustitución del cableado convencional por un sistema de bus serie presenta el problema de que un nodo defectuoso puede bloquear el funcionamiento del sistema completo. Cada nodo activo transmite una bandera de error cuando detecta algún tipo de error y puede ocasionar que un nodo defectuoso pueda acaparar el medio físico. Para eliminar este riesgo el protocolo CAN define un mecanismo autónomo para detectar y desconectar un nodo defectuoso del bus, dicho mecanismo se conoce como aislamiento de fallos.

Capa de aplicación: Existen diferentes estándares que definen la capa de aplicación; algunos son muy específicos y están relacionados con sus campos de aplicación. Los más populares que ofrece el mercado son: CAN Open, Smart Distributed System, Device Net, OSEK, J1939 o CAN Kingdom. A continuación se detallaran muy brevemente los rasgos principales de algunos de ellos:

- **CAN Open** CAN Open fue originalmente diseñado para la industria de sistemas de control, pero las redes CAN Open también son usadas para aplicaciones de campo como transporte público, equipos médicos, etc. Las especificaciones cubren el nivel de aplicación, el perfil de la comunicación, el armazón de los aparatos programables de los nodos y recomendaciones de cables y conectores. Es uno de los HLP más utilizados en las aplicaciones basadas en el bus CAN.
- **Device Net** El protocolo de comunicación Device Net es abierto y aceptado como un estándar de industria en todo el mundo. Fue diseñado para comunicar dispositivos de control inteligentes así como sensores de campo, estaciones de pulsadores, arrancadores, interfaces de operador simples, y variadores de control de velocidad. Es un sistema de CAN barato. Un aspecto curioso es que este protocolo también puede integrarse en CAN Kingdom. Es rápido y llega a soportar 3 velocidades: 125Kbps, 250Kbps y 500Kbps. Además soporta 64 nodos activos. Device Net básicamente distingue entre mensajes de procesos de mayor prioridad (I/O Mensajes) y mensajes de menor prioridad (Mensajes Explícitos). Además utiliza una técnica orientada a objetos, en el que la información está estructurada en diferentes objetos.
- **SDS (Smart Distributed System)** Smart Distributed System, es un sistema de bus para sensores y actuadores inteligentes. Es un modo



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

eficiente de conectar pequeños aparatos a un controlador máster. Todos los nodos SDS tienen asignados un número del rango (de 0 a 125) llamado dirección del aparato y debe ser único para todos los nodos del sistema. Este número es usado como base para seleccionar un conjunto de identificadores de CAN que pueden ser usados por este módulo.

- **CAN Kingdom** Este protocolo de alto nivel está diseñado principalmente para realizar sistemas de control, móviles hidráulicos, etc. Algunos denominadores comunes de estos sistemas son el alto desarrollo en tiempo real, alta demanda de seguridad. El protocolo CAN Kingdom soporta entre otras cosas, cambios dinámicos de identificadores, identificadores estándar y extendidos, un reloj global y un hardware que limita el número de módulos del sistema. En este protocolo, el nodo por defecto en la red tiene toda la información necesaria para inicializarse en el sistema. Las especificaciones de los sistemas serán los métodos de acceso al bus, la gestión de la red, las listas de mensajes y los formatos de los datos. El diseñador de sistemas es responsable del software implementado en el nodo y él puede decidir en qué condiciones los nodos serán aceptados por el sistema. En el fondo el diseñador de CAN Kingdom tiene la máxima libertad para crear su propio sistema.

El encapsulado de la trama de datos

CAN utiliza mensajes de estructura predefinida, es decir, tramas para la gestión de la comunicación. En concreto, el protocolo CAN define cuatro tipos de tramas diferentes: trama de datos, trama remota, trama de errores y trama de sobrecarga.

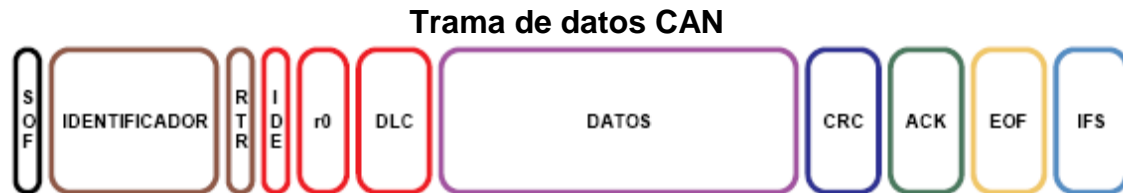
Tanto la trama de errores como la trama de sobrecarga son mensajes destinados al control de errores y se generan automáticamente por los nodos cuando detectan cualquiera de los muchos protocolos de error definidos por CAN. En cuanto a la trama remota, es básicamente una trama de datos pero sin datos, y se utiliza para solicitar la transmisión de una trama de datos. Su función es simplemente para tener también un control pero en este caso sobre la red.

Pero realmente el mensaje que es interesante conocer al detalle es la trama de datos, la trama más importante de las cuatro. En un bus CAN los nodos transmiten la información espontáneamente con tramas de datos, bien sea por un proceso cíclico o activado ante eventos en el nodo. Así que su función principal es poner información en el bus (siempre es un "Broadcast" a todos los demás nodos de la red). Para la transmisión en el bus, se crea un marco de



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II datos (Data Frame), cuya longitud abarca como máximo 130 bits (formato estándar) o 150 bits (formato extendido). De esta forma queda asegurado que el tiempo de espera hasta la siguiente transmisión, posiblemente muy urgente, se mantenga siempre corto.

La trama de datos consta de 7 campos de bits diferentes:



Start of frame (SOF): El campo de inicio de la trama está compuesto de un único bit. Se trata de un bit de sintonización dominante que indica el inicio de la transmisión. El flanco descendente de este bit es utilizado por los nodos receptores para sincronizarse entre sí.

Campo de arbitraje: Se trata del campo de identificación que permite reconocer a los nodos la prioridad del mensaje. Cuanto más bajo sea el valor del identificador más alta es la prioridad, y por lo tanto determina el orden en que van a ser introducidos los mensajes en la línea. Los bits de identificador se transmiten en orden de más significativo a menos significativo. Según sea el formato de la trama estándar (CAN 2.0-A) o extendido (CAN 2.0-B), este campo tendrá 11 ó 29 bits respectivamente. Dentro del mismo campo de arbitraje, hay un bit denominado RTR que indica si el mensaje contiene datos (0) o si se trata de una trama remota sin datos (1).

Campo de control: Este campo informa sobre las características del campo de datos. El bit IDE es el identificador del formato, es decir, si el bit IDE se transmite en nivel dominante, se trata del formato estándar y si es en nivel recesivo, es un formato extendido. Los cuatro bits que componen el campo DLC indican en binario el número de bytes de datos en el mensaje (de 0 a 8). En cuanto al bit r0, es un espacio reservado para futuras ampliaciones del bus.

Campo de datos: En este campo aparece la información del mensaje con los datos que el nodo correspondiente introduce en el bus CAN. Puede contener entre 0 y 8 bytes (de 0 a 64 bits en saltos de 8).

Cyclic Redundancy Check (CRC): Este campo tiene una longitud de 16 bits y es utilizado para averiguar si se ha recibido correctamente lo que se ha transmitido. Se produce un error de CRC cuando el resultado calculado no es

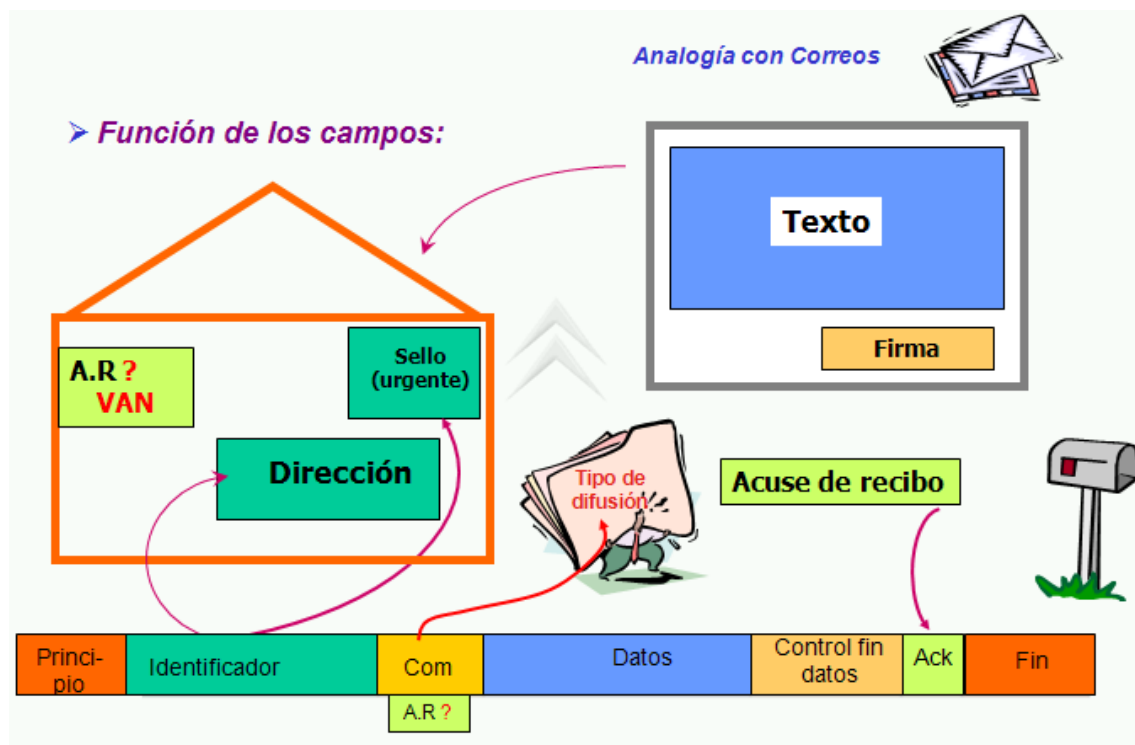


Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II el mismo que la secuencia CRC recibida. Para ello se utilizan los 15 primeros bits, mientras el último siempre es un bit recesivo que delimita el campo CRC.

Campo de reconocimiento (ACK): El campo ACK es un campo de dos bits que indica si el mensaje ha sido recibido correctamente. El nodo transmisor pone este bit como recesivo y cualquier nodo que reciba el mensaje lo pone como dominante para indicar que el mensaje ha sido recibido.

End of frame (EOF): El campo fin de trama indica el final del mensaje con una cadena de 7 bits recesivos.

Campo de intermisión (IFS): Consta de un mínimo de 3 bits recesivos que indican la separación entre dos tramas CAN seguidas. Los campos realmente importantes a tener en cuenta serían el del arbitraje (Identificador más RTR), el del control (IDE más DLC) y el de datos. Los campos CRC, ACK e IFS se detallarán más a fondo en los siguientes apartados.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Tramas de gestión y de control de errores

Como ya se ha comentado en el apartado anterior, los datos son transmitidos por medio de mensajes llamados tramas. Por otro lado y como ya se pudo ver en el mismo apartado citado anteriormente, aparte de las tramas de datos existen otras tres tramas más, las cuales están relacionadas con la gestión y control del bus. La trama de errores y la trama de sobrecarga por un lado (sirven para manipulación de errores y son generadas por los mismos nodos cuando detectan cualquiera de los muchos protocolos de error definidos por CAN); y la trama remota por otro lado (es como una trama de datos pero sin el campo datos). A continuación se va ver más información acerca de estas tramas.

Trama remota: Básicamente sirve para saber qué nodo está conectado al bus. No se mandan datos ni nada por el estilo, solo el identificador y se espera la respuesta del nodo. Primero se envía el identificador y luego el bit RTR (se encuentra dentro del campo del Identificador o Arbitraje) que es recesivo. Si al mismo tiempo se manda una trama de datos, ésta gana pues el bit RTR del identificador es dominante. El formato de esta trama es el siguiente:

SOF	Arbitraje	Control	CRC	ACK	EOF
-----	-----------	---------	-----	-----	-----

Un detalle muy importante a tener en cuenta, es que una trama remota también puede ser utilizada por un nodo para solicitar la transmisión de una trama de datos con la información asociada a un identificador dado. El nodo que disponga de la información definida por el identificador la transmitirá en una trama de datos.

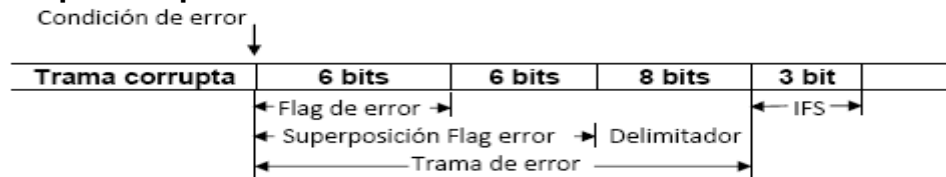
Pero cuidado, porque en un bus CAN los nodos transmiten la información espontáneamente con tramas de datos, bien sea por un proceso cíclico o activado ante eventos en el nodo. Pero por el contrario, la trama remota sólo se debe utilizar para detección de presencia de nodos o para puesta al día de información en un nodo recién incorporado a la red. Estos mensajes también pueden entrar en colisión en el bus. Entonces, como sucede siempre, el del identificador de mayor prioridad sobrevivirá y los demás serán retransmitidos lo antes posible.

Trama de error: Son generadas por cualquier nodo que detecte un error definido. Es una trama de dos campos, por un lado el Flag de error y por otro el delimitador. Éste último consiste en 8 bits recesivos consecutivos que le permite a los nodos iniciar limpiamente la transmisión. El formato de esta trama es el siguiente:



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Campos que componen la trama error



La condición o indicador de error será distinto según el estado o flag de error del nodo que detecte el error. Existen dos estados o flags de error de nodo: "Activo" y "Pasivo". El Activo consiste en seis bits dominantes consecutivos y el Pasivo consiste en seis bits recesivos consecutivos, a no ser que estén sobrescritos por otros bits dominantes de otros nodos. Si un nodo en estado de error "Activo" detecta un error en el bus interrumpe la comunicación del mensaje en proceso generando un "Indicador o Condición de error activo" que consiste, como ya se ha comentado antes, en una secuencia de 6 bits dominantes sucesivos y continuos. Esta secuencia rompe la regla de relleno de bits (La regla del relleno de bits consiste en que cada cinco bits de igual valor se introduce uno de valor inverso.) y provocará que el resto de los nodos detecten error y por lo tanto transmitan también 6 bits. Por lo tanto, la longitud de esta superposición de flags oscila entre 6 y 12 bits (dependiendo del número de nodos que detectan el error en el bus). Luego dicho transmisor coloca el delimitador de error mandando los 8 bits recesivos, espera el **IFS** (Se conoce como IFS (Inter Frame Spacing) al espacio entre tramas. Es el espacio existente entre una trama (de cualquier tipo) y la siguiente trama de datos o remota. Esta secuencia o separación predefinida también se conoce con el nombre de intermisión o íter-trama. La longitud mínima debe ser de tres bits recesivos. En un nodo en estado de error activo (nodo transmisor) esto es suficiente para reiniciar la transmisión, en cambio en el de error pasivo (nodo receptor o cualquier otro) tiene que esperar 8 bits más para iniciar una nueva transmisión o el bus permanecerá en reposo. De esta forma se asegura una ventaja en inicio de transmisión a los nodos en estado activo frente a los nodos en estado pasivo.) y reinicia la transmisión del mensaje.

Si un nodo en estado de error "Pasivo" detecta un error, el nodo transmite un "Indicador o Condición de error pasivo" seguido, de nuevo, por el campo delimitador de error. El indicador de error de tipo pasivo consiste en 6 bits recesivos seguidos y, por tanto, la trama de error para un nodo pasivo es una secuencia de 14 bits recesivos (6 del flag más 8 del delimitador). De aquí se deduce que la transmisión de una trama de error de tipo pasivo no afectará a ningún nodo en la red, excepto cuando el error es detectado por el propio nodo que está transmitiendo. En ese caso los demás nodos detectarán una violación de las reglas de relleno y transmitirán a su vez tramas de error. En este caso nuevamente la trama tendrá la misma estructura que la del error activo.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Trama de sobrecarga: Una trama de sobrecarga tiene el mismo formato que una trama de error activo. Sin embargo, la trama de sobrecarga sólo puede generarse durante el espacio entre tramas, es decir, en el IFS. De esta forma se diferencia de una trama de error, que sólo puede ser transmitida durante la transmisión de un mensaje. La trama de sobrecarga consta de dos campos, el Indicador o Flag de Sobrecarga, y el delimitador. El indicador o flag de sobrecarga consta de 6 bits dominantes que pueden ser seguidos por los generados por otros nodos, dando lugar a un máximo de 12 bits dominantes. El delimitador es de 8 bits recesivos.

¿Cuándo se genera una trama de sobrecarga?

Pueden darse tres situaciones que lo provoquen:

- Si debido a condiciones internas un nodo es incapaz de recibir un nuevo mensaje en ese instante. En este caso, la trama de sobrecarga se envía justo cuando se esperaba el primer bit del espacio de intermisión (IFS). De esta forma se consigue retrasar el inicio de transmisión del nuevo mensaje.
- Cuando un mensaje es validado por los receptores y el último bit del EOF es recibido como dominante (cuando debe ser recesivo), por tanto, este bit dominante no es considerado como error, pero la existencia de este bit ilegal puede deberse a que el receptor ha perdido la sincronización y eso requiere una reacción por parte de la red.
- Si durante la intermisión uno de los dos primeros bits es dominante, es decir, si la intermisión ha sido violada. En este caso, la trama de sobrecarga se enviará un bit después de recibir el bit dominante.

Cada nodo puede mandar hasta un máximo de 2 tramas de sobrecarga.

Detección y señalización de errores

Una vez vistas las tramas, se va proceder a ver como las utiliza el protocolo CAN para detectar los errores y señalizarlos. En el protocolo CAN se ha realizado un importante sistema de manejo de errores. Este sistema permite detectar errores en los mensajes que se transmiten para así en el caso que sea necesario se retransmitan los mensajes erróneos. Si un controlador del bus detecta un error, enviará un flag de error para avisar del mismo y así destruir el tráfico del bus. Los otros nodos posteriormente, detectarán un error debido a la violación de la regla del bit stuffing (relleno) en el flag de error y enviarán otros flags de error.

El protocolo CAN define cinco tipos de detección de errores, dos de estos modos son al nivel de bit y el resto al nivel de trama:



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

- **Error de bit:** Cuando un nodo está transmitiendo, éste monitoriza el nivel del bus y si el bit que lee en el bus no coincide con el que ha transmitido, es que se ha producido un error, así que se señala un “Error de bit”. En el siguiente tiempo de bit el nodo transmisor envía una trama de error, entonces, la trama fallida será reenviada después del espacio de intermisión.
- **Error de Stuff:** Como ya se pudo ver, este error detecta si dentro del área codificada por el método del bit stuffing, existen seis bits consecutivos del mismo nivel. El nodo que detecte esto, enviará una trama de error justo en el tiempo de bit siguiente al de detectar el sexto bit del mismo nivel. La trama fallida será retransmitida después del espacio de intermisión.
- **Error CRC:** Tanto este método para detectar errores como los dos siguientes (ACK y Forma) se realizan mediante el chequeo de la misma trama de datos a través de los campos correspondientes. El CRC es un campo de las tramas que contiene un código de redundancia cíclica el cual comprueba si hubo errores en la recepción del mensaje. Con el CRC podemos detectar errores aleatorios en hasta 5 bits o una secuencia seguida de 15 bits corruptos. Si el CRC calcula en el nodo receptor no coincide con el CRC enviado (el que contiene la trama), entonces el receptor descarta el mensaje recibido y envía una trama de error, pidiendo una retransmisión de la trama.
- **Error ACK:** Otro campo que se encuentra en las tramas. En esta ocasión se trata de dos bits que indican si el mensaje fue recibido satisfactoriamente. El nodo transmisor manda el ACK en recesivo esperando a que el nodo receptor lo sobrescriba en dominante, de lo contrario se le considera una trama corrupta y lo retransmitirá. Así, si el transmisor detecta un ACK positivo, es decir, un bit dominante durante el campo ACK, sabrá que al menos un nodo ha recibido correctamente el mensaje.
- **Error de forma:** Algunas partes de los mensajes en CAN tienen formas fijas. Estas zonas son: el delimitador de CRC, el delimitador de ACK, EOF y el espacio de intermisión (IFS). Los bits de estas zonas deben ser recesivos. Si algún controlador de CAN detecta que alguno de estos bits es dominante, genera una Trama de Error porque se ha producido un “Error de forma”, esta trama se enviará en el tiempo de bit siguiente al bit erróneo. Por tanto, como se puede ver la detección de un error por parte de un nodo se hace público al resto de nodos mediante la



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

transmisión de tramas de error. Entonces la transmisión de ese mensaje fallido es abortada y será retransmitida tan pronto como el arbitraje de la red se lo permita al nodo correspondiente.

Por otra banda, cada nodo contiene dos contadores de error: el contador de Transmisiones de Error (TEC) y el contador de Recepciones de Error (REC). Hay varias reglas que gobiernan el modo en que estos contadores deben incrementarse y/o decrementarse. En esencia, si un nodo transmite una Trama de Error su contador de Transmisiones de Error se incrementa en 8 y los nodos que reciben una Trama de Error incrementan en 1 el valor de su contador de Recepción de Error. Si un nodo transmite una trama correctamente, el TEC decrementa y si un nodo recibe una trama correctamente, su REC decrementa. Inicialmente los nodos están en estado de Error Activo. Cuando uno de sus dos contadores llega a superar el valor 127, el nodo pasa a estar en estado de Error Pasivo.

Aplicaciones y diagnóstico

Como ya se comentó anteriormente en el apartado A.1, la mayor parte de las aplicaciones que posee el CAN están concentradas en la industria automovilística donde realiza el control del motor, de la mecánica del automóvil así como los sistemas de entretenimiento. Esto es debido en parte a que el sistema CAN fue especialmente diseñado en sus orígenes con la idea de aplicarlo en el mundo de los automóviles para la transmisión de datos entre los sistemas electrónicos de control y regulación, como por ejemplo:

- Control del cambio.
- Control electrónico del motor o de la bomba de inyección.
- Sistema antibloqueo (ABS).
- Sistema de tracción antideslizante (ASR).
- Control de estabilidad (ESP).
- Regulación del momento de arrastre del motor (MSR).
- Inmovilizador.
- Ordenador de a bordo, etc.

Relacionado con la industria automovilística, el CAN se extiende al transporte público y otras máquinas móviles como aviones, helicópteros, trenes, barcos y los controles de tráfico y sistemas de información conductor/pasajero. Pero en general el CAN se aplica en cualquier sistema de control industrial: sistemas de control de plantas y maquinaria, redes entre máquinas, maquinaria agrícola, instrumental médico, sistemas de supervisión, etc. y en la automatización de edificios: control de ascensores, aire acondicionado, sistemas de calefacción y refrigeración, control de iluminación, etc. En resumen, en cualquier sistema que





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

precise control en tiempo real distribuido y con escaso flujo de datos. Por lo tanto, esto augura un gran futuro para el CAN. Incluso actualmente ya se puede considerar que el CAN a alcanzado un nivel extraordinario de madurez e implantación ya que los fabricantes y procesadores digitales de señal están incorporando controladores CAN de forma bastante generalizada (se habla de cientos de millones de nodos).

Volviendo al tema de las aplicaciones CAN en vehículos motorizados, cabe decir que existen tres campos de aplicación esenciales para estos sistemas:

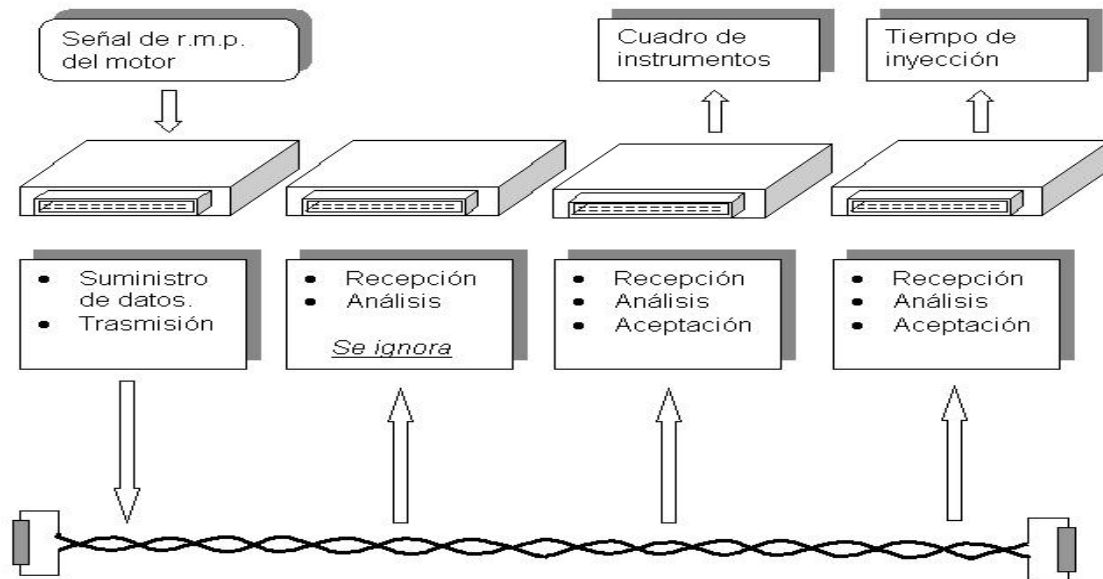
- Acoplamiento de unidades de control.
- Electrónica de la carrocería y de confort.
- Comunicación móvil.

De estos tres campos se va a documentar un poco sobre el primero: Acoplamiento de unidades de control.

Se acoplan entre sí sistemas electrónicos como el control del motor o de bomba de inyección, sistema antibloqueo ABS, sistema de tracción antideslizante ASR o regulación de la dinámica de marcha ESP, control electrónico de cambio, etc. Las unidades de control están aquí unidas como estaciones con igualdad de derechos, mediante una estructura de bus lineal. Esta estructura presenta la ventaja de que en caso de fallar una estación, el sistema bus continúa estando plenamente a disposición de las demás estaciones. En comparación con otras disposiciones lógicas (estructuras anulares o estructuras en estrella) se reduce así esencialmente la probabilidad de un fallo total. En el caso de estructuras anulares o en estrella, el fallo de una estación o de la unidad central, conduce a un fallo total. Las velocidades de transmisión típicas están entre aprox. 125 Kbps y 1 Mbps (ejemplo: la unidad de control del motor y la unidad de control de bomba en la regulación electrónica diesel comunican entre sí a 500 Kbps). Las velocidades de transmisión deben ser tan altas para poder garantizar el comportamiento de tiempo real requerido. En las siguientes figuras vemos dos diferentes ejemplos:

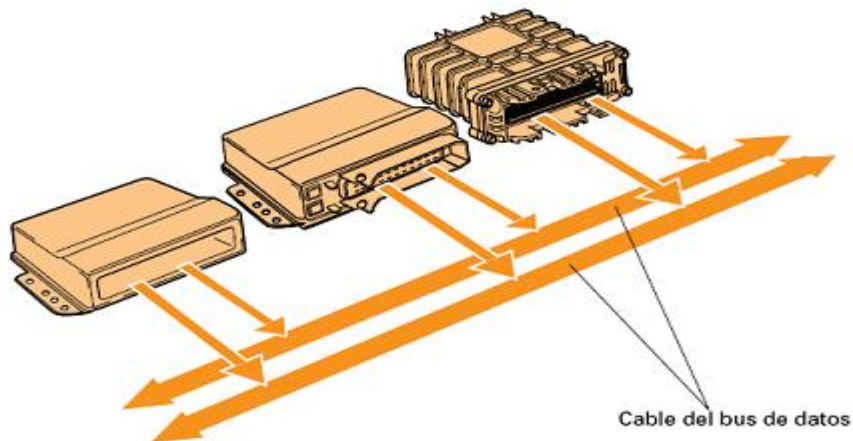


Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II



Bit con	Valor	Validación
0 voltios	0	superior
5 voltios	1	inferior

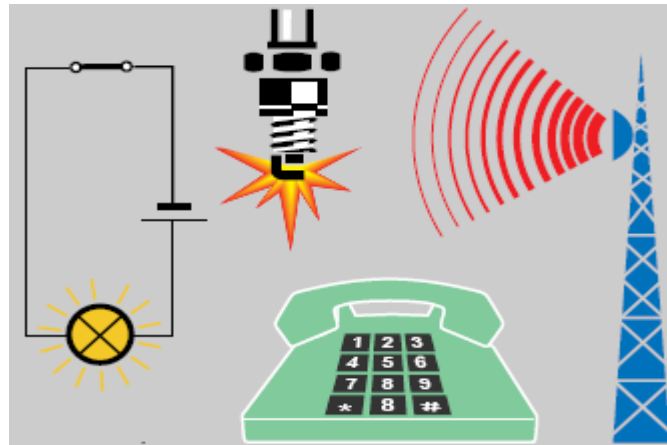
Prioridad	Protocolo de datos	Campo de estado
1	Freno I	001 1010 0000
2	Motor I	010 1000 0000
3	Cambio I	100 0100 0000



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Fuentes parásitas

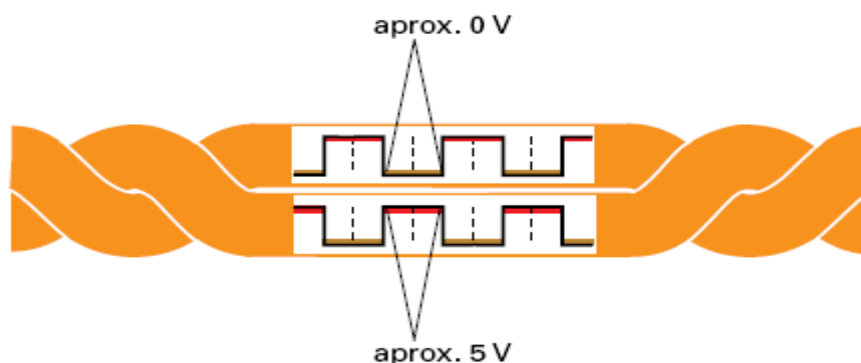
En el vehículo son fuentes parásitas los componentes en cuyo funcionamiento se producen chispas o se abren o cierran circuitos de corriente. Otras fuentes parásitas son por ejemplo teléfonos móviles y radioemisoras, o sea, todo aquello que genera ondas electromagnéticas. Estas ondas electromagnéticas pueden influir en la transmisión de datos o incluso la pueden falsificar.



Para evitar influencias parásitas sobre la transmisión de datos se procede a retorcer conjuntamente los dos alambres del bus de datos. De esa forma se evitan al mismo tiempo emisiones perturbadoras procedentes del propio cable del bus de datos.

Las tensiones en ambos cables se encuentran respectivamente contrapuestas.

Eso significa lo siguiente: Si uno de los cables del bus tiene aplicada una tensión de aprox. 0 voltios, el otro tiene una de aprox. 5 voltios y viceversa. En virtud de ello, la suma de tensiones es constante en cualquier momento y se anulan mutuamente los efectos electromagnéticos de campo de ambos cables del bus. El cable del bus está protegido contra la penetración de emisiones parásitas y tiene un comportamiento casi neutro hacia fuera.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II**Diagnóstico**

Tal como se ha explicado en este documento, el protocolo CAN dispone de una serie de mecanismos de control para el reconocimiento de anomalías. Si una estación registra una anomalía, emite entonces un "flag de error", que detiene la transmisión en curso. De esta forma se impide que otras estaciones reciban el mensaje erróneo. Gracias a este sistema de seguridad que incorpora el CAN se consigue que las probabilidades de fallo en el proceso de comunicación sean muy bajas, pero sigue siendo posible que cables, contactos y las propias unidades de mando presenten alguna disfunción.

Por ello, para el análisis de una avería, se debe tener presente que una unidad de mando averiada abonada al CAN en ningún caso impide que el sistema trabaje con normalidad. Lógicamente no será posible llevar a cabo las funciones que implican el uso de información que proporciona la unidad averiada, pero sí todas las demás. Una alternativa posible para localizar fallos en el CAN es emplear el programa informático CANalyzer (Vector Informatik GmbH) con el ordenador y con la conexión adecuada. Este programa permite visualizar el tráfico de datos en el bus CAN, indica el contenido de los mensajes y realiza la estadística de mensajes, rendimiento y fallos.

Otros sistemas de comunicación

Hoy en día existen varios protocolos de comunicaciones propietarios, es decir, definidos por un fabricante en concreto para su propio uso. Ello implica la existencia en el mercado de productos que no pueden comunicarse entre sí. Cuando deberían poder hacerlo. Por tanto, surge la necesidad de crear normas abiertas para que los diversos fabricantes utilicen los mismos protocolos y la comunicación entre sus respectivos módulos sea posible.

Clasificación SAE de los protocolos de comunicación en el Automóvil.

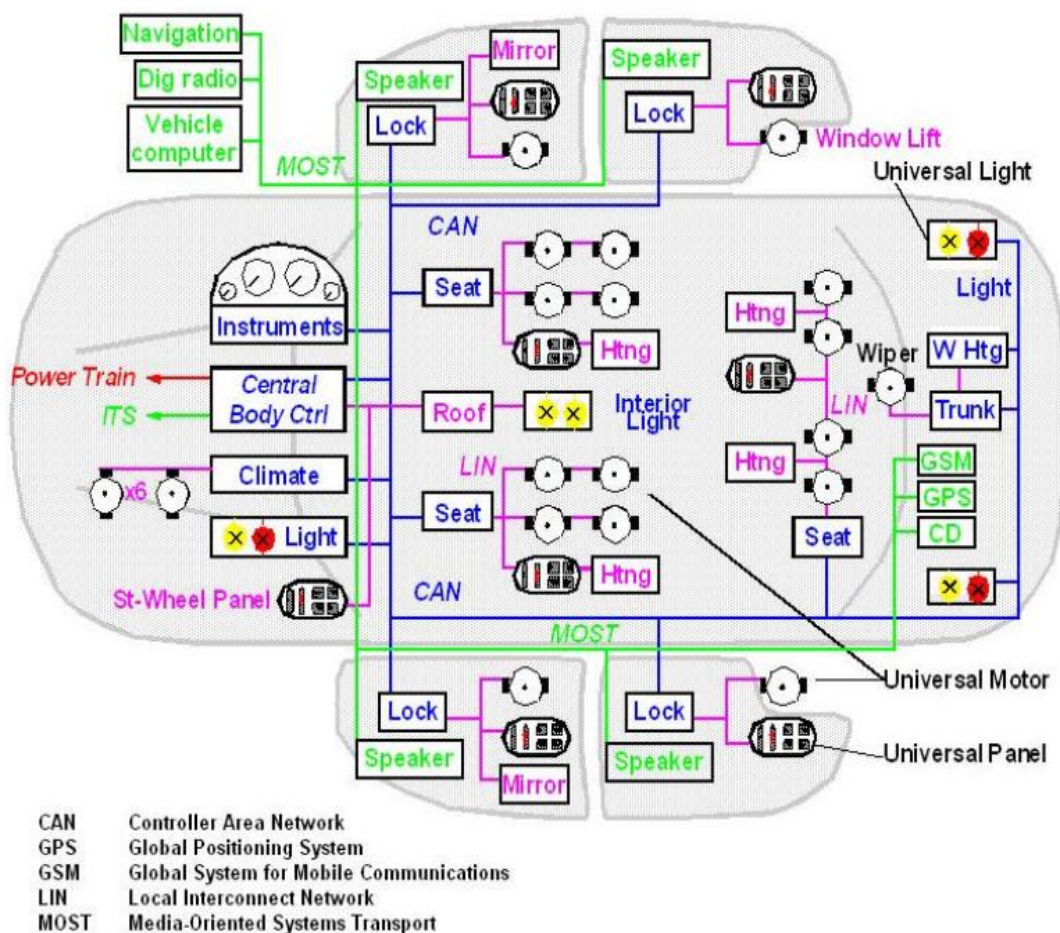
Clase	Velocidad transmisión	Aplicación
A	< 10 kbit/s	Transmisión de señales de activación poco frecuente (activación humana)
B	10 – 100 kbit/s	Transmisión de información de control en tiempo real no estricto. Aplicaciones no críticas. Presentación de datos al conductor, comunicación entre equipos electrónicos (aire acondicionado, audio, retrovisores, diagnosis, etc.)
C	100 kbit/s – 1 Mbit/s	Transmisión de información en tiempo real estricto. Aplicaciones de seguridad crítica (ABS, ASR, ESP, etc.)
D	> 1 Mbit/s	Aplicaciones multimedia (Internet, televisión, etc.) y funciones críticas en tiempo real (<i>X by wire</i>).



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

Existe una gran variedad de redes digitales para el automóvil, unas desarrolladas por fabricantes de automóviles y otras por empresas suministradoras o sociedades internacionales. Algunas de estas redes son abiertas, es decir, pueden ser empleadas por cualquiera y; otras son propietarias y sólo pueden usarse con licencia.

En un mismo vehículo puede haber diversos buses de comunicaciones, cada uno de los cuales destinado a unas funciones, según sus características. La siguiente figura muestra una posible configuración:



Sin embargo, todavía no se ha conseguido que haya un protocolo estándar empleado por todos los fabricantes de automóviles, aunque algunos están más impuestos que otros. A continuación se realiza una breve descripción de algunos protocolos desarrollados para el automóvil.





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II CCD (*Chrysler Colusión Detection*)

El protocolo CCD fue desarrollado en la década de 1980 por Chrysler. Utiliza un par trenzado como medio de transmisión y su codificación de bit es de tipo NRZ (*Non Return to Zero*). Como acceso al medio emplea el procedimiento CSMA/CA (*Carrier Sense Múltiple Access / Colisión Avoidance*) que permite el arbitraje en el acceso múltiple al bus, de forma que sólo el mensaje de más prioridad consigue ser transmitido sin ser destruido por los nodos restantes que habían accedido al bus. Este tipo de acceso al medio permite que funcione sin necesidad de un controlador central de las comunicaciones. La velocidad de transmisión es de aproximadamente 7.8 kbit/s. siendo, por tanto, de clase A. Este protocolo se aplica en el multiplexado de diversos sensores del automóvil.

SAEJ1S50

Es el estándar SAE para las clases A y B (velocidad de transmisión baja y media). Es una combinación del SCP de Ford y del Protocolo Clase 2 de General Motors y fue aprobado por la SAE en 1988 y revisado finalmente en 1994.

Existen dos versiones (al ser desarrollo de dos protocolos propietarios), cuya diferencia consiste en la codificación de bit y la velocidad de transmisión. La versión más lenta emplea una codificación VPM (*Variable Pulse Modulation*) alcanzando 10.4 kbit/s y transmite con un solo cable referido a masa. La versión más rápida usa una codificación PWM (*Pulse Width Modulation*) consiguiendo 41.6 kbit/s y transmite en modo diferencial con dos cables. Como acceso al medio emplea el procedimiento CSMA/CR (*Carrier Sense Múltiple Access / Colusión Resolution*). lo que significa que cualquier módulo puede intentar transmitir si detecta que el bus está desocupado. Si más de un módulo intenta transmitir al mismo tiempo, un proceso de arbitraje determinará cuál de ellos continuará transmitiendo y quién deberá reintentarlo después.

La principal aportación de este protocolo fue la inclusión de las respuestas de los nodos destinatarios dentro de la propia trama emitida desde el nodo origen. En concreto permite: respuesta de un byte desde un simple destinatario, respuestas concatenadas de un byte desde múltiples destinatarios y respuesta de múltiples bytes desde un simple destinatario. La utilización en series comerciales empezó en el momento de su estandarización, siendo quizás el primer protocolo en ser aplicado de forma masiva y actualmente todavía está en uso.





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II J1939

Se trata de un protocolo de alta velocidad (clase C) para soportar funciones de control en lazo cerrado en tiempo real. Su documentación cubre todas las capas del modelo OSI, con lo que está totalmente definido. Es compatible con CAN 2.0B y emplea el formato de identificador extendido de 29 bits de éste. La velocidad de transmisión es de 250 kbit/s y el medio físico es un par trenzado apantallado.

VAN (Vehicle Área Network)

Este protocolo fue desarrollado por el grupo francés Renault y PSA Peugeot-Citroen en 1988. Sin embargo, hasta 1990 no se presentó la primera versión VAN 1.0 y en 1994 la versión 2.0. La tecnología VAN está estandarizada según la especificación ISO 11519. Utiliza par trenzado como medio de transmisión, la codificación de bit es de tipo Manchester, lo que supone un menor aprovechamiento del ancho de banda. El acceso al medio se realiza por contienda en referencia al campo de dirección o identificador. La velocidad de transmisión es de hasta 1 Mbit/s. La principal aportación de este protocolo es el uso de un carácter especial de inicio de trama que le permite resincronizar el receptor a una precisión de un 1% partiendo de una diferencia entre emisor y receptor de un 20%. Al igual que CAN, también incluye un bit de reconocimiento dentro de la trama original, así como la posibilidad de encadenar una trama respuesta a continuación de la trama original, de forma parecida al J1850. VAN es utilizado en aplicaciones de control en el automóvil.

PALMNET (Protocol for Automotive Low and Medium speed NETWORK)

Protocolo desarrollado por Mazda en 1989 y puesto en práctica en un modelo de la misma marca en 1990. Utiliza par trenzado como medio de transmisión, la codificación de bit es de tipo XRZ. Con respecto a la capa de enlace, emplea la técnica CSMA/CD con arbitraje no destructivo. La velocidad de transmisión es de 24 kbit/s, lo que lo ubica en la clase B. La aportación más palpable de este protocolo es el procedimiento de confirmación multinodo al final de la trama y antes del carácter de fin de trama, denominado ANC (**Acknowledgement for Network Control**). Este método consiste en el envío al final de la trama, por parte de cada nodo conectado al bus, de una señal de reconocimiento (ACK) en una posición localizada de forma que se pueda interpretar por el nodo emisor. Este método limita el número máximo de nodos a 24.

En 1994 Mazda realizó una mejora sustancial del protocolo PALMNET, que pasó a denominarse Advanced PALMNET (**Protocol for Automotive Local área Multiplexing NETWORK**). Aunque este protocolo conserva parte del nombre de su antecesor, el significado del acrónimo cambió de acuerdo al





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II
aumento de velocidad. Advanced PALMNET especifica dos versiones: media-baja velocidad (hasta 125 kbit/s) y alta velocidad (hasta 1 Mbit/s).

TTP (Time Triggered Protocol)

El protocolo TTP fue desarrollado por la Universidad de Viena y Daimler-Benz Research para aplicaciones en tiempo real y cubrir las carencias de los protocolos de comunicación desarrollados hasta ese entonces. TTP es un protocolo activado por tiempo que proporciona:

- Coordinación de conjunto (***membership service***). Esto significa que cada nodo sabe el estado actual de cualquier otro nodo que compone el sistema distribuido.
- Servicio de sincronización de reloj tolerante a fallos (***global time-base***). Cada nodo dispone de un reloj interno, actualizado por una trama de resincronización enviada con cierta frecuencia por un nodo.
- Soporta cambios de modos de operación. Las aplicaciones en tiempo real tienen diferentes modos de operación: inicialización, normal, emergencia, etc. El protocolo de comunicaciones debe soportar cambios rápidos de un modo a otro de manera consistente en todos los nodos.
- Gestión de redundancia distribuida.
- Detección de error con pequeña latencia.
- Acceso al medio con el método TDMA (***Time-Division Multiple Access***) basada en una planificación temporal de los mensajes.
- Codificación de bit del tipo MFM (***Modified Frequency Modulation***).
- Velocidad de transmisión superior a 10 Mbit/s. con lo que es de clase D.

Las anteriores características hacen que este protocolo pueda utilizarse en cualquier aplicación crítica de tiempo real en automóvil.

De este protocolo se han realizado diversas versiones, cubriendo todas las clases SAE (A, B y C). La velocidad del bus TTP/C. se especificó en 1 Mbit/s. si bien posteriormente aparecieron versiones de velocidad superior. Actualmente está en 25 Mbit/s.

TTCAN (Time Triggered Controller Area Network)

TTCAN es una extensión del protocolo CAN estandarizada por la norma ISO 11898-4 que permite emplear CAN para mensajes disparados por tiempo (***Time Triggered***), lo que aumenta el determinismo, fiabilidad, componibilidad y sincronización sobre el bus CAN.

El porqué de TTCAN se debe a que en CAN, debido al proceso de arbitraje, el envío de un mensaje puede retrasarse si hay otro en proceso de envío o si



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

algún mensaje de mayor prioridad también compite por el control del bus. Incluso los mensajes de mayor prioridad pueden sufrir una pequeña latencia.

En TTCAN un nodo específico (**Time Master**) transmite un mensaje de referencia indicando el inicio de un ciclo de tiempo. El mensaje es reconocido por el resto de nodos gracias a su identificador! y se sincronizan con él, proporcionando así un punto de referencia temporal para la planificación de la transacción de mensajes. Este mensaje puede ser enviado periódicamente (en modo TT o **Time Triggered**) o al ocurrir en evento externo (en modo ET o **Event Triggered**).

Cada ciclo de tiempo se divide en porciones (**slots**) que son asignadas a un nodo o conjunto de nodos, o simplemente se dejan como tiempo muerto. Entonces, la transmisión de un mensaje por parte de un nodo comienza al inicio del correspondiente **slot**.

El protocolo debe asegurarse un comportamiento tolerante a fallos por parte del nodo maestro que envía el mensaje de referencia. Por ello, si éste falla, otro nodo lo reemplaza.

TTCAN dispone de dos niveles de sincronización, nivel 1 y nivel 2. El primero garantiza la operación TT de CAN basada en el mensaje de referencia del **Master**, incluyendo la redundancia de éste. El segundo nivel establece una base de tiempos sincronizada global y se realiza una corrección continua de la desviación entre los controladores de CAN.

La velocidad máxima de transmisión es 1 Mbit/s. estando ésta limitada por el acceso CSMA/CD. El número máximo de bytes de datos en la trama es de 8.

LIN (Local Interconnect Network)

Protocolo presentado el año 2000 en el congreso SAE de Detroit. LIN fue creado para ser un protocolo de bajo coste, usado en distancias cortas y a bajas velocidades de transmisión. Se diseñó para realizar funciones de clase A como control de las puertas, aire acondicionado, limpiaparabrisas, techo solar, etc. Algunas de las características más interesantes de LIN son:

- Es un protocolo de comunicaciones serie de un solo cable (**single-wire**), cuya trama está orientada a carácter, de forma que el protocolo se puede implementar basándose en un microcontrolador dotado de una UART.
- El medio de acceso está controlado por un nodo maestro, de manera que se evita la colisión de mensajes y se garantiza una latencia para la transmisión de una señal.
- Una característica particular es su mecanismo de sincronización que permite la resincronización del reloj de comunicación para la determinación del periodo de bit. sin utilizar un cuarzo o resonador cerámico.



Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

- La máxima velocidad de transmisión es de 20 kbit/s. siendo de clase B.
- Es posible añadir nodos a la red LIN sin requerirse cambios en el software o hardware. El tamaño de una red típica LIN es de 12 nodos.

La sincronización del reloj, la sencillez de la comunicación UART y el acceso al medio de un solo cable son los factores principales de la buena relación coste-eficiencia del protocolo LIN.

Volcano

Volcano es mi protocolo usado por Volvo y puede definirse como "TTP sobre CAN" ya que está basado en CAN y es determinista, pero no es TTCAN.

De acuerdo con el **Volcano Communications Concept** (Concepto de Comunicaciones Volcano). Este protocolo surge como una técnica en la que se integra CAN de tal forma que se garantiza la latencia de los mensajes. Esto se consigue especificando la latencia y periodicidad de los mensajes durante la fase de diseño. Ello permite calcular las máximas latencias de tal forma que el diseñador puede especificar la configuración de la red de tal forma que se garantice ciertos parámetros específicos evitando el arbitraje tanto como sea posible.

Esto parece implicar que el envío de mensajes es TT en vez de ET. de allí la definición de "TTP sobre CAN".

En principio este método implica que las cargas al bus CAN pueden ser mayores que empleándolo convencionalmente.

ByteFlight o Si-bus

En octubre de 1999 BMW dio a conocer el protocolo ByteFlight, también conocido como Si-bus. Fue desarrollado con el objetivo de satisfacer las necesidades de las aplicaciones de seguridad crítica: determinismo, tolerancia a fallos y alta velocidad de transmisión. Como acceso al medio emplea el método denominado FTDMA (**Flexible Time-División Múltiple Access**), consistente en una distribución cíclica de bus, dividida en dos partes. La primera dedicada a mensajes de alta prioridad, con una planificación de tipo TT mientras que la segunda se reserva a mensajes de tipo ET. los cuales se consideran de menor prioridad. Según algunos autores, este protocolo tiene, como principal cualidad la unión de las ventajas de los métodos síncronos y asíncronos, garantizando unas latencias deterministas para los mensajes de alta prioridad y un uso flexible del ancho de banda para mensajes de baja prioridad. Para reducir las EMI's (**Electromagnetic Interferentes**, Interferencias ElectroMagnéticas) utiliza fibra óptica como medio de transmisión. La velocidad de transmisión es de 10 Mbit/s.





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II FlexRay

El protocolo FlexRay fue desarrollado por el consorcio FlexRay (BMW. Daimler-Chrysler. Motorola. Philips. GM y Bosch). Es un protocolo tolerante a fallos diseñado para transmisiones de alta velocidad y aplicaciones de control avanzado, como los sistemas por cable "X-by-wire". Las principales características de este protocolo son:

- Soporta los dos paradigmas: comunicación determinista (planificación estática TT) y comunicación dinámica conducida por eventos ET.
- Sincronización de reloj tolerante a fallos a través de una base de tiempo global (**global time base**).
- Acceso al bus libre de colisiones.
- Latencia de mensajes garantizada.
- Mensajes orientados al direccionamiento vía identificadores.

Velocidad de transmisión de hasta 10 Mbit/s.

DC-BUS

En diciembre del año 2000 se constituyó la conocida como DC-bus Alliance (Audi, BMW. PSA. Infineon Technologies y YAMAR Electronics), con objeto de desarrollar un sistema de comunicación que empleara como canal de comunicación la red de alimentación de 14 V ó 42 V futura, basado en la tecnología patentada de YAMAR Electronics. Con ello se pretendía reducir los costes y simplificar el cableado de los sistemas del vehículo

Esta alianza pretende crear un nuevo estándar internacional y abierto que sea compatible con los protocolos empleados hoy en día en los automóviles y que funcione tanto con el actual sistema de 14 V como con el de 42 V que se espera implementar en un futuro próximo.

Este sistema utiliza actualmente los protocolos CAN y LIN y tiene una velocidad de transmisión de 10 kbit/s (aplicaciones de mecatrónica). 250 kbit/s (aplicaciones de telemática) y 1.7 Mbit/s (aplicaciones de multimedia).

Los beneficios de DC-BUS son:

- Ahorro de coste y peso del cableado ordinario.
- Arquitectura y modificaciones flexibles.
- Simplicidad de instalación.
- Elevada velocidad de las comunicaciones.

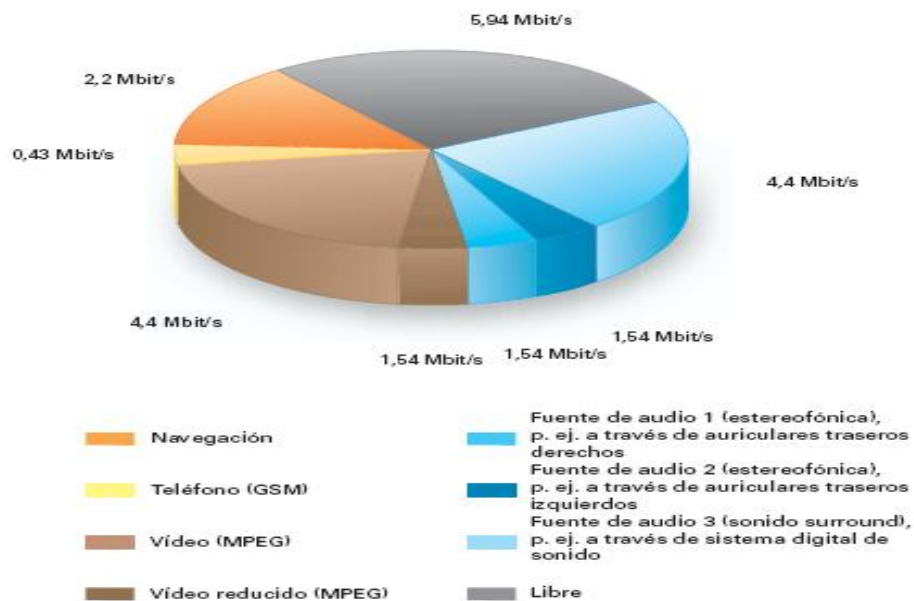


Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II MOST BUS

La denominación de este sistema de bus de datos surgió por la (Media Oriented Systems Transport (MOST) Cooperation). A esta entidad se han asociado diversos fabricantes de automóviles, sus proveedores y empresas productoras de software, con objeto de llevar a la práctica un sistema unitario para la transmisión rápida de datos.



Velocidades de transmisión de los medios





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

El término (Media Oriented Systems Transport) representa una red con transporte de datos de orientación medial. Esto, en contraste con el CAN-Bus de datos, significa que se transmiten mensajes direccionados hacia un destinatario específico. Esta técnica se implanta en vehículos Audi para la transmisión de datos en el sistema de infotainment. El sistema de infotainment ofrece una gran cantidad de medios vanguardistas destinados a **información y entretenimiento**.

Para la realización de un complejo sistema de infotainment resulta adecuada la transmisión optoelectrónica de los datos, porque con los sistemas de CAN-Bus que han venido empleando hasta ahora no se pueden transmitir los datos con la suficiente rapidez y, por tanto, tampoco en las cantidades correspondientemente necesarias.

Debido a las aplicaciones de vídeo y audio se necesitan velocidades de transmisión del orden de muchos Mbit/s (El MOST-Bus permite transmitir 21,2 Mbit/s.).

Con ayuda del MOST-Bus optoelectrónico se establece el intercambio de datos en forma digitalizada entre los componentes participantes. La transmisión de datos con ayuda de ondas luminosas, aparte de suponer una menor cantidad de cables y un menor peso, permite trabajar con una velocidad de transmisión sustancialmente mayor. En comparación con las ondas de radio, las ondas luminosas tienen longitudes muy cortas, no generan ondas electromagnéticas parásitas y son a su vez insensibles a éstas. Estos nexos permiten una alta velocidad de transmisión de los datos y un alto nivel de seguridad contra fallos e interferencias.

Bluetooth

En el mundo moderno de los negocios y en la vida privada, la comunicación e información móvil viene obteniendo una importancia creciente. Una persona suele ocupar varios aparatos móviles, tales como el teléfono móvil, el Personal Digital Assistant (PDA) o PC portátil. El intercambio de información entre los aparatos móviles sólo se podía establecer en el pasado a través de una comunicación por cable o por infrarrojos.

Estas comunicaciones no estandarizadas limitaban bastante el margen de movilidad o tenían un manejo complicado. La tecnología Bluetooth™ viene a resolver este problema. Permite enlazar los aparatos móviles de diferentes marcas a través de una comunicación por radiofrecuencia estandarizada.

Esta tecnología se implanta en los vehículos modernos para la comunicación inalámbrica entre el auricular selector para el teléfono y la unidad de control para teléfono/ telemática.

Para una fecha posterior se han previsto más posibilidades de aplicación para el usuario del vehículo:





Sincronización - Inyección Electrónica y Control Computarizado de Motor OBD II

- El montaje de un segundo auricular selector en las plazas traseras
- La conexión del PC portátil, smartphone y notepad del usuario del vehículo hacia internet para la transmisión de información y para efectos recreativos
- La recepción y transmisión de correos electrónicos del usuario mediante PC portátil o PDA
- La transmisión de direcciones y números de teléfono del PC portátil o PDA del usuario hacia el sistema de interfaz multimedia (MMI)
- La instalación del sistema de manos libres para teléfonos móviles sin adaptadores de cables adicionales
- El empleo de la tecnología Bluetooth™ en otros sistemas del vehículo (Ejemplo: mando a distancia por radiofrecuencia para la calefacción independiente)

