

Estudio del comportamiento de una red CAN basada en el 82C200 de Phillips y en el 82527 de Intel

Arturo Pascual Granged
Grupo de Aguas Residuales
José Luis Sevillano Ramos
Gabriel Jimenez Moreno
Antón Civit-Balcells
Facultad de Informática y Estadística

Resumen

El protocolo CAN nace como protocolo de red de procesadores en el campo de la automoción. Hoy en día, este protocolo se está introduciendo dentro de otros campos de aplicación. El objetivo de este artículo consiste en poner de manifiesto la dependencia respecto de la arquitectura del chip de comunicaciones en cuanto a la ocupación del canal por estación para distintos valores de la carga. La intención de los autores consiste en crear una base sólida para la evaluación de esta red con vistas a su utilización no solo en el campo de la automoción sino para cualquier aplicación en tiempo real que conlleve la utilización de una red de procesadores.

1. Introducción.

Tradicionalmente, las comunicaciones en tiempo real se han limitado por dos factores. Por un lado, el protocolo debe proporcionar una cota superior a los retrasos por mensaje, incluyendo mecanismos adicionales como el de control de errores o mecanismos de prioridad. Por otro lado, el coste de la instalación debe ser mínimo. Hoy en día, la fabricación flexible se suele organizar en pequeñas celdas, por lo que, la mejor solución pasa por una red en tiempo real pequeña y de bajo costo.

Las redes del tipo Ethernet no son válidas ya que, aunque su coste es muy bajo debido a su gran utilización en otras aplicaciones, no acota los tiempos de retraso en los mensajes. El proyecto MARS alcanzó este objetivo usando una red del tipo Ethernet con un protocolo TDMA. Las redes MAP ("Manufacturing Automation Protocols"), basados en técnicas de paso de testigo ('token') con un sistema de prioridades muy complejo, tienen un campo de aplicación muy restringido, ya que su complejidad, y por tanto su lentitud, así como su elevado coste, limita su campo de aplicaciones. Desde el punto de vista de los protocolos de acceso al medio, las redes tipo DAMA o de asignación en función de la demanda son muy recomendables en aplicaciones en tiempo real. CSMA/DCR[1] (resolución determinista de colisiones) y mecanismos de arbitración del bus como el de

FutureBus o MultiBus II [2] son ejemplos de sistemas DAMA disponibles comercialmente. En cualquier caso, los sistemas de comunicación en tiempo real más comunes están basados en buses serie bien de forma sincrónica, bien de manera asíncrona, incluyendo los buses del tipo SPI o I2C.

Recientemente, CAN (Controlled Area Network [3]), un protocolo desarrollado para reducir el cableado en los coches, está siendo implantado de manera satisfactoria en controles industriales. El coste por nodo es relativamente pequeño, y está implementado en distintos controladores comerciales (Phillips 82C200, Intel 82527,...), procesadores (Motorola HC05, Intel '196, ...) así como en un gran número de sensores, actuadores,...etc. También, y dentro de este tipo de protocolo, se puede incluir el protocolo VAN (Vehicle Area Network) cuya única modificación respecto al protocolo CAN consiste en que este usa una codificación tipo Manchester. El protocolo CAN incluye todas las características de un protocolo para aplicaciones en tiempo real.

Muy pocos son los estudios que se pueden encontrar de este protocolo. En este trabajo intentamos poner de manifiesto la dependencia que va a tener la arquitectura del controlador en el comportamiento de los mensajes en la red. Los datos que aquí vamos a presentar están basados en distintas simulaciones realizadas con modelados de los controladores de Phillips 82C200 y de Intel 82527. El efecto del controlador ya fue descrito por Tindell [6], aunque de una forma cualitativa. En este estudio se pretende poner de manifiesto con más detalle mediante las simulaciones.

2. El protocolo CAN.

CAN es una red multimaster que opera a una velocidad máxima de 1Mbps. La longitud del canal está limitada de forma que se pueda llevar a cabo la sincronización entre las estaciones. Actualmente, dicha longitud se ve limitada no solo por el retraso de propagación asociado al cable de conexión, sino también por los propios tiempos de retraso asociados a los controladores y a los transceptores, e incluso a la

tolerancia de los osciladores [4]. Como valores típicos, la longitud máxima del cable para una red trabajando a 1Mbps será de 50 m.

Los mensajes transmitidos por la red son de distintos tipos. En nuestro caso, hemos considerado que los mensajes van a ser únicamente del tipo de datos ("data frames"). Así, un mensaje va a consistir de 47 bits de cabecera donde se incluyen el identificador, ACK, CRC, ...etc, y un campo de datos de entre 0 y 8 bytes. El identificador no solo va a servir como un filtro a la hora de recibir los mensajes sino que también será el que marque la prioridad del mensaje.

Cuando una estación tiene un mensaje para transmitir, espera a que el canal esté libre y comienza a transmitir (como ocurre con el protocolo CSMA 1-persistente). Se comienza transmitiendo un bit de sincronización y acto seguido se transmite el identificador del mensaje a la vez que se monitoriza el canal. Si dos o más estaciones intentan transmitir simultáneamente, el canal se comporta como una gran puerta AND para todas las estaciones y, por lo tanto, la estación que transmita con más baja prioridad perderá la arbitración ya que emitirá un "1" y monitorizará un "0". En ese instante, la estación dejará de transmitir y esperará a que el canal esté nuevamente libre. De esta manera, la arbitración se realizará de forma que no se destruya el mensaje de la estación ganadora, no teniéndolo que repetir (como ocurre con los otros protocolos). De esta manera se asegura que todo el tiempo que este ocupado el canal será con transmisiones útiles. Aún así, la utilización máxima del canal será del 97.7% debido al espacio entre mensajes y de toda esta ocupación solamente el 58% es de utilización con mensajes debido a las cabeceras de mensaje (incluso en el mejor de los casos, con mensajes de 8 bytes). De hecho, este valor suele ser menor debido al "bit-stuffing" que introduce hasta 19 bits para el control de errores.

3. Los controladores de CAN.

Dentro del mercado existe una gama relativamente amplia de controladores de red CAN. Su misión consiste en hacer de puente entre el procesador y la red. Su funcionamiento generalmente se basa en ocupar diversas posiciones de memoria o de E/S del procesador donde éste deja los diversos mensajes y donde están los diversos registros de configuración y status del controlador. El trabajo fundamental del controlador de comunicaciones es la de añadir las cabeceras a los mensajes, prepararlos para transmitir y monitorizar el canal para transmitirlos. A su vez, este dispositivo es el encargado de retirar las cabeceras a los mensajes recibidos y controlar y corregir sus posibles errores. La filosofía general de estos controladores es la de liberar al máximo al procesador del trabajo de control de la red para que pueda ocuparse de otras tareas. La intercomunicación entre el procesador y el host puede realizarse de formas

alternativas, aunque depende del controlador particular que empleemos y de sus características.

Dentro de la variedad de controladores que existen en el mercado, los dos ejemplos que hemos tomado como referencia en nuestro estudio han sido el 82C200 de Phillips y el 82527 de Intel debido a que, en ambos casos, son muy extendidos y, a su vez tienen una gran compatibilidad con una extensa gama de procesadores existentes en el mercado.

3.1 El controlador 82C200 de Phillips.

El controlador 82C200 de Phillips es un controlador de comunicaciones para CAN diseñado para automoción, entornos industriales en general y entornos agresivos. El 82C200 tiene todos los elementos necesarios para una alta realización de comunicaciones. Su misión consiste en el control completo de las capas física y de enlace de datos. Desde el punto de vista de la conexión con el procesador, este chip proporciona dos modos de funcionamiento seleccionables por hardware: un modo "Little Ending" y un modo "Big Ending". Estos dos modos de funcionamiento hacen que sea compatible con casi cualquier procesador. Desde el punto de vista interno, el 82C200 tiene un mapa de memoria muy compacto de 31 bytes. Los diez primeros bytes y el último forman los registros de control y status. Los diez siguientes forman un buffer de transmisión de mensajes (los dos primeros bytes llevan la información del identificador, un código de longitud, y un código de mensaje requerido, y los otros ocho bytes son para el mensaje en sí). Los últimos diez bytes (sin contar el último del mapa) están duplicados en dos bancos de memoria que se usan como buffer de recepción de mensajes con la misma estructura que el buffer de emisión. Desde el punto de vista de nuestro estudio, la existencia de este buffer de transmisión de mensajes es la gran novedad que se añade a nuestras simulaciones.

3.2 El controlador 82527 de Intel.

El controlador 82527 de Intel es un controlador de comunicaciones serie capaz de realizar comunicaciones acorde al protocolo CAN. Soporta las especificaciones de los standard CAN1.0 y CAN2.0 (con ampliación en el tamaño de Identificadores). El 82527 es un controlador diseñado para trabajar en entornos de automoción y ambientes agresivos controlando las capas físicas y de enlace de datos. Desde el punto de vista externo, este controlador dispone de cinco modos de interconexión con un procesador: Un modo de 8 bits multiplexados de datos y direcciones "Little Ending", un modo de 16 bits multiplexados de datos y direcciones "Little Ending", un modo de 8 bits multiplexados de datos y direcciones "Big Ending", un modo de 8 bits no multiplexados y un modo de interconexión serie. Estos cinco modos de conexión lo hace más flexible que el 82C200 de

Phillips a la hora de enlazarlo con algún procesador. Como módulo de entrada-salida es más completo ya que, además del interfaz CAN, dispone de dos puertas de E/S de 8 bits (una para el caso de 16 bits multiplexado). Desde el punto de vista interno, el 82527 dispone de un mapa de memoria de 256 bytes. Dentro de este mapa de memoria, mayor que el correspondiente al 82C200, el 82527 tiene 14 buffers de transmisión y recepción configurables por software, y 1 buffer de recepción fijo, con 8 bytes de memoria por buffer. Por un lado, estos buffers de emisión y recepción dotan a este controlador de una gran potencia aunque, por otro lado, introduce una complejidad añadida a la hora de programarlo.

4. Simulación de CAN.

Los simuladores fueron escritos con ayuda de la librería SMPL siguiendo un modelo similar al expuesto en [5] para la red Ethernet con las modificaciones detalladas en [7]. La simulación de cada uno de los controladores de comunicaciones introduce parámetros propios a cada uno de los simuladores aunque las diferencias esenciales con respecto al modelo general expuesto en [7] son:

- La asignación de identificadores ya no es única por estación. Una de las modificaciones que se introducen consiste en que cada estación va a tener asignados un grupo de identificadores y no solo uno.
- La generación de mensajes se realiza de manera independiente a las estaciones. Mediante una generación de números aleatorios que sigue una distribución exponencial, se van generando mensajes que se van asignando, mediante una distribución uniforme discreta, a identificadores y no a estaciones.
- Hay que tener en cuenta que cada uno de los controladores dispone de buffers de transmisión de datos.

A la hora de generar los mensajes, se ha mantenido la posibilidad de que el tamaño del mensaje no sea fijo como ocurre con el protocolo CAN usado para automoción. La utilización de mensajes de tamaños distintos cubre la posibilidad de distintos mensajes por identificador. En aplicaciones de control, cada estación va a poder transmitir distintos tipos de información asociada a una misma tarea y, por tanto, a un mismo identificador.

Dentro de la simulación, las modificaciones más sustanciales han sido las realizadas a la hora de tener en cuenta la arquitectura interna de los controladores. Para el caso del 82C200, cuando se genera un mensaje, se comprueba si el buffer interno de la estación está vacío. Si es así, el mensaje se introduce en el buffer y se prepara para la transmisión. Caso de que el buffer no esté vacío, se comparan los identificadores y se deja en el buffer el mensaje más prioritario.

Para el caso del 82527, debido a que el buffer de transmisión puede llegar a tener el tamaño de 14 mensajes, cuando se genera un mensaje, este se

introduce en el buffer de forma ordenada por identificador. Cabe destacar que la ordenación del buffer habrá que hacerla vía software desde el procesador, ya que el controlador no tiene ninguna forma de ordenar los mensajes.

En ambos casos, teniendo en cuenta que las velocidades de los procesadores es del orden de unos 20 Mhz o mayor, y la velocidad máxima de transferencia en el canal será de 1Mbps, no se han considerado los retrasos asociados a estas operaciones de "puesta en buffer".

Otro punto a tener en cuenta es la distribución de los identificadores entre las estaciones. Hemos considerado conveniente el número de cuatro identificadores por estación para poner de manifiesto el efecto del buffer. Los resultados dependen fuertemente de la distribución de los identificadores en las estaciones. Así, hemos tenido en cuenta dos configuraciones distintas:

- Configuración "ordenada": Los identificadores van a quedar distribuidos de forma que los cuatro más prioritarios quedarán asociados a la primera estación, los cuatro siguientes a la segunda, ..., etc.

- Configuración "desordenada": En este caso, a la primera estación le corresponderá el identificador 1, $n+1$, $2n+1$ y $3n+1$, a la segunda estación le corresponderán los identificadores 2, $n+2$, $2n+2$ y $3n+2$, ..., y así para el resto de las estaciones, donde n representa el número de estaciones en la red.

El número de estaciones elegido para este estudio es de diez con cuatro identificadores cada una. Se han realizado algunas simulaciones con un número mayor de estaciones y con una asignación mayor de identificadores por estación pero los resultados cualitativos no difieren de los aquí expuestos.

Los simuladores emplean el método de "batch means" con una longitud 10000 muestreos que dan un intervalo de confianza del 95% con menos de un 10% de error.

5. Análisis de resultados

La primera simulación que se puso en marcha fue la correspondiente al 82C200. En la figura 1 se muestra el reparto de carga correspondiente a cada identificador para una configuración desordenada. En ella se observa como, para valores pequeños de la carga, la utilización del canal es uniforme por identificador. Conforme se va aumentando la carga, se ve un aumento de la ocupación del canal por los identificadores más prioritarios en detrimento de los menos prioritarios. Este resultado era de esperar por la propia asignación de prioridades. Para valores más altos de la carga, del orden de 1.7, se comienza a detectar el efecto del buffer interno de la estación. A partir de estos valores se empieza a detectar como la disminución de la utilización por identificador ya no es gradual, empezándose a detectar "mesetas" en la gráfica. La primera meseta corresponde al grupo de los primeros identificadores por estación, la segunda meseta corresponde a los segundos identificadores de

Analizamos este comportamiento para una configuración ordenada y se obtuvieron unos resultados muy espectaculares (fig 2). El resultado para valores pequeños de la carga siguen siendo los esperados. Cuando aumentamos la utilización, se empieza a dentear la gráfica por la parte correspondiente a los identificadores menos prioritarios. Para valores de la carga del orden de 1.0, la gráfica muestra una forma completa de dientes de sierra que se acentúa cuando tenemos cargas cercanas al calor 2.0. Aquí se pone de manifiesto como el acceso a la red no solo va a depender del identificador del mensaje sino también de como estén distribuidos esos identificadores por estación y de la existencia de ese buffer de mensajes. De esta forma, se puede observar y cuantificar la existencia de la inversión de prioridades que Tindell introdujo como una posibilidad y de forma intuitiva. El acceso a la red del identificador ultimo de una estación será menor que el siguiente, menos prioritario, por el hecho de éste ocupa la posición de identificador prioritario de la siguiente estación.

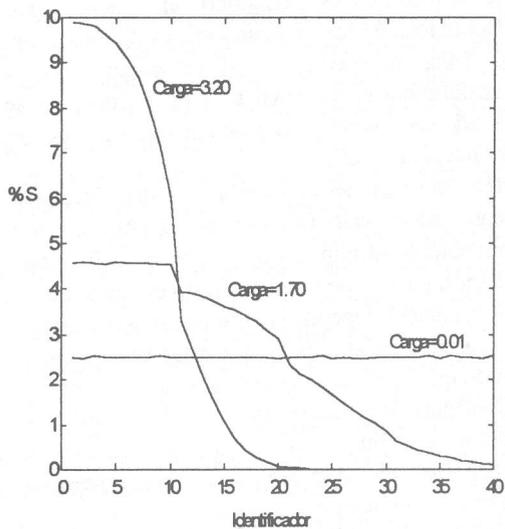


Figura 1: 82C200, configuración desordenada.

cada estación, Dentro de cada meseta, la disminución de la utilización es gradual, pero entre ellas existe un salto apreciable que pone de manifiesto la competición por el buffer. Para cargas muy altas, del orden de 3, se puede observar como, los segundos identificadores tienen poco acceso a la red y el tercer identificador de cada estación no accede a la red.

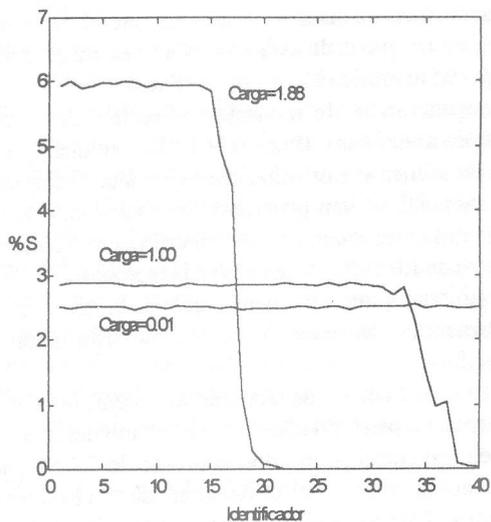


Figura 3: 82527, configuración ordenada.

Para el caso de las simulaciones con el 82527 con la configuración ordenada (fig. 3), se puede observar como esos dientes de sierra desaparecen. En este caso el efecto de buffer es menor sobre la utilización por identificador. En este caso, la utilización se distribuye uniformemente entre los identificadores hasta valores de la carga del orden de 1.0. Para valores mayores de la carga se puede observar como la utilización por identificador decrece casi uniformemente para todos los identificadores menos prioritarios de todas las estaciones. En este caso se puede observar como el efecto del buffer consigue suavizar el comportamiento

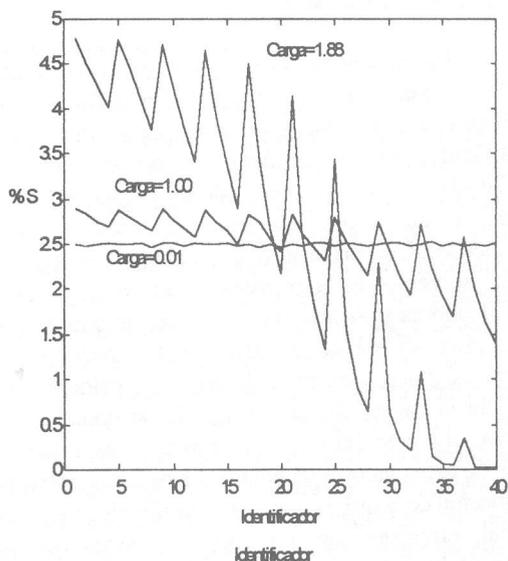


Figura 2: 82C200: Configuración ordenada.

de la red. Hay que tener en cuenta que el buffer se reordena internamente independientemente de los controladores de comunicaciones.

Para el caso de una configuración desordenada, se puede ver que el comportamiento de la red sigue siendo suave. Para este caso, seguimos teniendo el comportamiento normal para valores pequeños de la carga. Cuando ese valor va aumentando, se puede observar como la disminución de la utilización es uniforme. Cabe destacar que desaparecen esas fluctuaciones que existían en la zona donde la utilización va disminuyendo, siendo en este caso mucho mas suave.

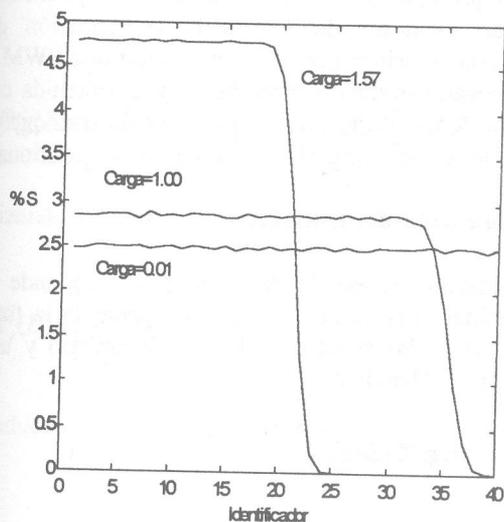


Figura 4: 82527, configuración desordenada.

6. Conclusiones.

Como se puede deducir de este trabajo, la elección del controlador va a resultar fundamental en el comportamiento de una red CAN. Para redes con un nivel bajo de carga, ambos controladores van a tener unas prestaciones similares, pero en este escrito se pone de manifiesto como el 82527 de Intel va a poder responder de manera mas efectiva a los posibles picos de carga que puedan aparecer sobre la red, mientras que el 82C200 de Phillips no consigue absorberlos de manera efectiva.

También se puede observar, a raíz de los datos expuestos, que la asignación de los identificadores unida al efecto de ese buffer puede acarrear desviaciones apreciables respecto del funcionamiento ideal de la red. El efecto conjunto de un buffer pequeño y una configuración ordenada lleva, como se ha demostrado a la existencia de inversión de prioridades,

es decir, a que una estación menos prioritaria tenga mejor acceso al canal que una estación mas prioritaria. Desde el punto de vista de la red, se puede deducir que el controlador de Intel va a estar en mejores condiciones que el controlador de Phillips para funcionamiento en carga altas. Sin embargo, el control del buffer hace que el controlador de Phillips sea mas cómodo para su implantación. En cualquier caso, el controlador 82527 de Intel se ha dejado de fabricar aunque se sigue implantando en algunos microcontroladores de la gama '196 y 8051.

Agradecimientos: Deseamos agradecer al Profesor J.J. Serrano de la Universidad Politécnica de Valencia por su ayuda. Este trabajo ha sido financiado por el proyecto CICYT TAP 93-0443 y por el Grupo de Tratamiento de Aguas Residuales.

7. Referencias.

- [1] J.L. Sevillano, G. Jimenez, A. Civit "Performance Analysis of CSMA Networks with Deterministic Collision Resolution". Telecom System (aceptado para publicación).
- [2] J.L. Sevillano et al. "Performance evaluation of Multibus II based communication system for multiple robot controller". Proc. 12th IASTED Int. Conf. Applaid Informatics. Annecy, France. 1994.
- [3] "Road Vehicles - interchange of digital information Controller Area Network (CAN) for high speed communication" ISO 11898, 1993.
- [4] P. Buehring "Bit timing parameters for CAN networks" Phillips Report KIE 07/91 ME, 1991.
- [5] M.H. MacDougall, "Simulating Computer Systems". The MIT Press, 1987.
- [6] K.W. Tindel, H. Hanson, A.J. Wellings, "Analysing Real-Time Communications: Controlled Area Network (CAN)". Tech. Report. Dept. Computer Science, Univ. of York.
- [7] J.L. Sevillano, A. Pascual, G. Jimenez, F. Díaz del Rio, A. Civit-Balcell, "A performance Study of CAN in Manufacturing Cells". Proc. of 14th IASTED Int. Conf. Applaid Informatics. Innsbruck, Austria. 1996.