

PROTOCOLOS DE COMUNICACIÓN CANBUS Y CANOPEN

Becerra duitama
Jhonatan alexander¹

Maestría en Ingeniería Mecatrónica, Universidad Militar Nueva Granada,
¹u3900273@unimilitar.edu.co

Resumen

Este artículo trató los temas más importantes sobre los protocolos de comunicación CAN bus y CANopen. Primero, se ofreció una explicación del modelo OSI/ISO. Acto seguido, se enumeraron las características principales del bus CAN. Seguidamente, se detallaron las capas existentes en los protocolos CAN bus y CANopen. Después, se profundizaron temas como la cantidad de nodos existentes, las distancias entre componentes, la velocidad de transferencia de datos, la estructura física, los paquetes de información y el modelo (maestro-esclavo). Finalmente, se enunciaron las aplicaciones principales de dichos protocolos.

Palabras clave: CANbus, CANopen, Modelo OSI, protocolo de comunicación.

Abstract

This article dealt with the most important topics about the CAN bus and CANopen communication protocols. First, it was explained that it is an OSI/ISO model. Then the main characteristics of the CAN bus were named. Next, the existing layers in the CAN bus and CANopen protocols were explained. Subsequently, subjects such as the number of existing nodes, the distances between components, the speed of data transfer, the physical structure, the information packages and the model were explored. (maestro-esclavo). Finally, the main applications of these protocols were outlined.

Keywords: CANbus, CANopen, OSI Model, communication protocol.

1. INTRODUCCIÓN

Una red de comunicaciones industriales es una red en tiempo real, implementada en un sistema productivo que se encarga de conectar distintos procesos, tales como: comando, supervisión, mantenimiento y gestión, para asegurar la correcta transmisión de información. Esta transmisión de datos se efectúa mediante unos protocolos. La comunicación de datos entre los dispositivos que forman las redes industriales, es esencial para el correcto funcionamiento de las fábricas. Estas redes industriales dependen, en gran medida, de que los protocolos de comunicación sean eficientes, eficaces y efectivos a la hora de compartir información entre los componentes electrónicos. En algunos procesos, los dispositivos deben trabajar en condiciones extremas, en donde la transmisión de los datos se complica. Por tal motivo, la comunicación debe ser altamente fiable, debido a que un paquete de información perdido puede significar un riesgo físico y económico para el proceso (Wilamowski e Irwin, 2018).

Existen diferentes protocolos de comunicación, algunos de ellos son: PROFINET, MODBUS, ASI-BUS, PROFIBUS (Guerrero y Martínez, 2010), CANopen, CANbus, entre otros. CAN, que en inglés significa *Controller Area Network* (Red de Área de Control), fue desarrollado por Bosch en 1986, con el objetivo de atender la demanda de sistemas electrónicos de control en la industria automovilística. CAN es un protocolo de comunicación en serie normalizado por la Organización Internacional de Normalización (ISO). No utiliza un método maestro-esclavo para acceder al bus. En su lugar, utiliza un método único de control de acceso al bus, llamado “arbitraje no destructivo por bits”. Es un protocolo de comunicación muy simple, altamente fiable y prioritario entre los sensores, actuadores y dispositivos inteligentes. Se aplica una técnica de productor-consumidor

para acceder al medio físico basada en el acceso múltiple con detección de colisiones (CSMA-CD) (Kumar, 2014).

Se trata de un método determinista para la resolución de colisiones o conflictos en el bus, que emplea la contención del medio de transmisión. De esta manera, el sistema aprovecha la totalidad del ancho de banda disponible. CAN se define como una red de controladores independientes que soporta el control distribuido en tiempo real con un elevado nivel de seguridad. En los vehículos automotores, los distintos controles generan datos de diversa naturaleza, lo que tradicionalmente requería el envío de múltiples líneas de bus al controlador, derivando en diversos problemas de cableado. El protocolo CAN fue desarrollado para abordar eficazmente esta problemática y se consolidó como un estándar para las redes vehiculares, extendiéndose posteriormente a numerosos campos con fines de control (Kumar, 2014).

2. MODELO OSI/ISO

Antes de explicar en qué consisten los protocolos CANopen y CAN bus, es necesario explicar qué es un modelo OSI/ISO. El modelo de interconexión de sistemas abiertos, mejor conocido como Modelo OSI (Open System Interconnection), es un modelo base que se puede tomar para desarrollar protocolos de comunicación que pueden llegar a ser normalizados. Este modelo describe la arquitectura bajo la cual se recomienda construir un protocolo. Siete capas o niveles de software componen al modelo, y su finalidad es transmitir de manera segura y sin errores la información desde un punto inicial a un punto final. En la figura 1 se puede observar la representación gráfica de las siete capas del modelo OSI, desde la capa superior (capa de aplicación), hasta la capa inferior (capa física) (Angulo Angulo y Miranda Zambrano, 2018).

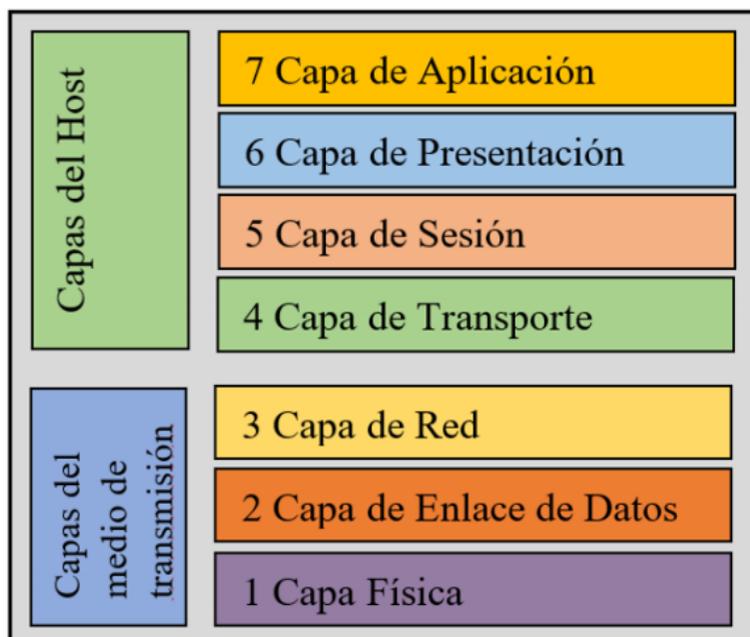


Figura 1. Modelo OSI.

Fuente: Angulo Angulo y Miranda Zambrano (2018).

La primera capa (capa física), se encarga de definir el medio por el cual se transmitirán los datos (por cables, por fibra óptica o inalámbricamente). También se establecen los materiales y las propiedades que debe tener el medio conductor y sus conectores. Finalmente, se precisan los niveles de las frecuencias de operación y las señales eléctricas.

La segunda capa gestiona el acceso de los datos al medio de transmisión. Esta capa les asigna direcciones para establecer un orden y facilitar su transmisión. También se asegura de que la información llegue de forma correcta. La capa de red define las rutas de los datos. Gracias a esta capa, se pueden definir las direcciones de dos o más nodos en una o más redes, permitiendo así la conexión entre nodos sin importar que no estén conectados físicamente.

La capa de transporte separa la información en paquetes de datos, dándoles un orden para su identificación y posterior envío. La capa de sesión se encarga de la autenticación de la información, manteniendo la comunicación

a través de una contraseña. La capa de presentación toma los datos y los traduce a un formato que facilita su lectura. Finalmente, la capa de aplicación presenta y procesa los datos de manera adecuada para que el usuario final pueda interpretarlos correctamente (Angulo Angulo y Miranda Zambrano, 2018; Oliva Alonso, 2013).

3. PROTOCOLO CAN

De acuerdo al modelo de referencia OSI, la arquitectura CAN se puede representar por dos capas: la capa física y la capa de enlace de datos (International Organization for Standardization - ISO, 2003). La figura 2 muestra las capas existentes en el protocolo CAN. Esto quiere decir que el protocolo CAN realiza el procedimiento descrito en la figura 1, en tan solo dos capas. La capa física se divide en tres partes, PLS (Señal física), PMA (medio físico de sujeción) y MDI (interfaz dependiente del medio). En la parte PLS, se codifica y decodifica la información, el tiempo de transmisión de bits y la

sincronización. En la parte PMA se encuentran las características del emisor/receptor, y en la parte MDI se localizan los conectores. Por otro lado, la capa de enlace de datos está compuesta por dos secciones: la sección LLC (Control de enlace lógico) y la sección MAC (Control de acceso al medio). La sección LLC contiene los filtros de aceptación de los datos, notificación de sobrecarga de datos y

gestión de recuperación; mientras que, en la sección MAC ocurre el empaquetamiento y el des-empaquetamiento de los datos, la codificación del marco (llenado y vaciado), la gestión del acceso a los medios, la detección de errores, la señalización de error, el reconocimiento y la serialización y des-serialización de la información (International Organization for Standardization - ISO, 2003).

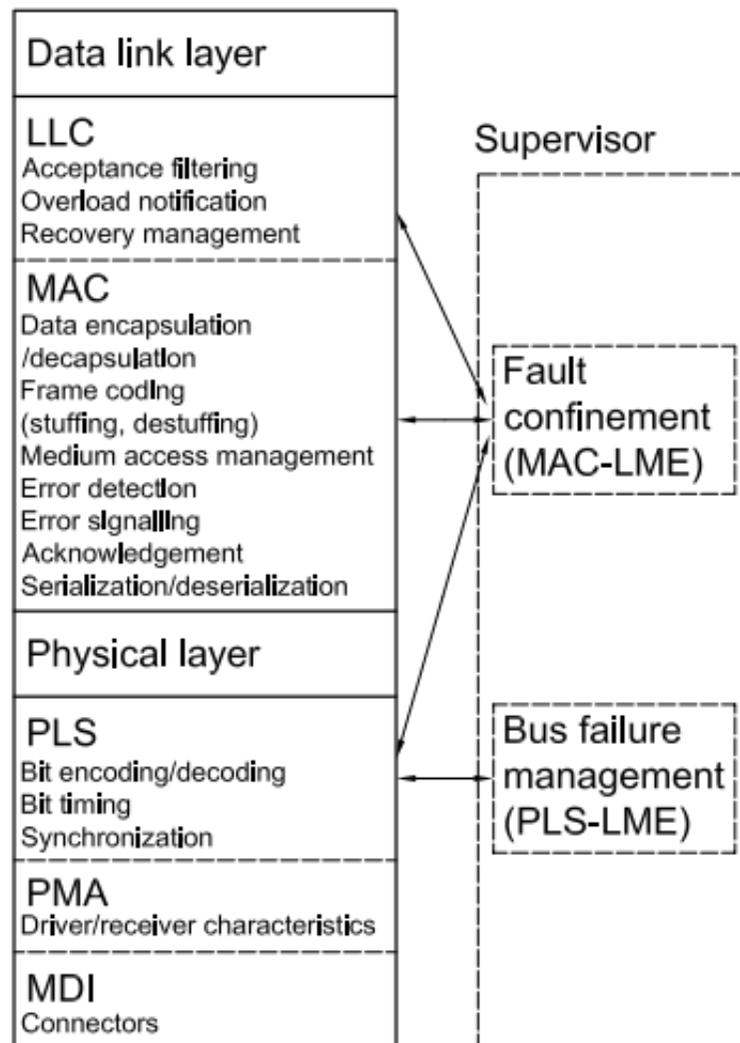


Figura 2. Modelo OSI/ISO en el protocolo CAN.

Fuente: International Organization for Standardization - ISO, (2003).

Protocolo CAN bus

El protocolo CAN bus solo incorpora la capa física y la capa de enlace de datos. En la figura 3, se muestra su modelo OSI/ISO (Thompson, 2018). La información de la capa de enlace de datos se transfiere a la capa de aplicación, sin embargo, esta capa no está definida dentro del protocolo CAN bus.

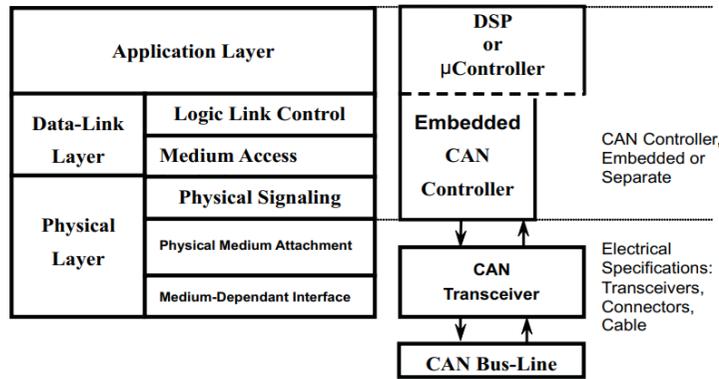


Figura 3. Modelo OSI/ISO del protocolo CAN bus.

Fuente: Thompson (2018).

Capa física

La capa física en el protocolo CAN bus define todos los aspectos físicos presentes en la capa, incluyendo terminaciones, características físicas del cable de datos y límites de tensión, teniendo en cuenta la velocidad de comunicación y el tiempo de transmisión de datos. La figura 4 muestra cómo debe ser la conexión de los nodos y la terminación estándar de los buses de comunicación. Generalmente, se utiliza esta configuración

por su simplicidad y su bajo costo de producción. El valor de la resistencia, R_L , debe ser de 120Ω (International Organization for Standardization - ISO, 2003). Del mismo modo, la capa física no aclara cuál es la cantidad máxima de dispositivos conectados a las líneas del bus. Una característica importante de este protocolo de comunicación es que los dispositivos se pueden conectar y desconectar en cualquier momento y el proceso no se ve afectado. Lo anterior depende de cómo es el cableado interno del bus (Thompson, 2018).

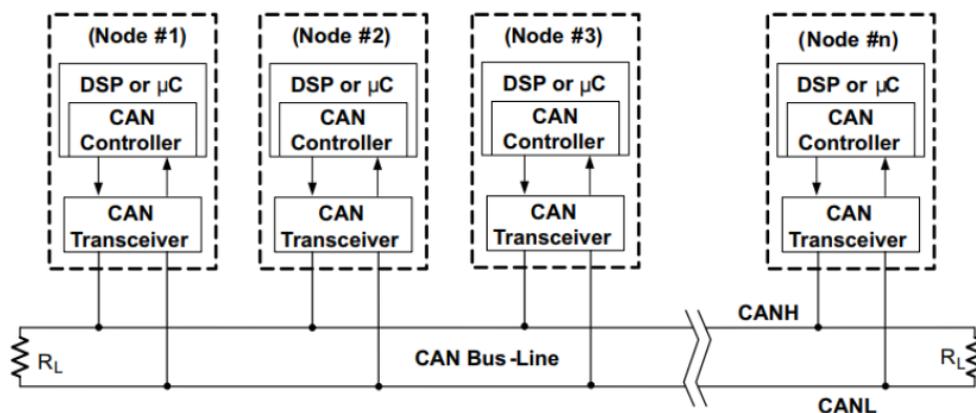


Figura 4. Línea CAN bus con conexiones terminales y nodos.

Fuente: Corrigan (2016).

La norma ISO-11898 no establece una interfaz de conexión física específica entre los dispositivos y el bus, aunque recomienda un esquema de conexiones de pines DB9 similar al ilustrado en la figura 5. No obstante, ciertos fabricantes industriales optan por implementar las líneas CAN_H y CAN_L utilizando otras configuraciones de entradas y salidas o bloques terminales de comunicación, con el objetivo de reducir la complejidad de la estructura.

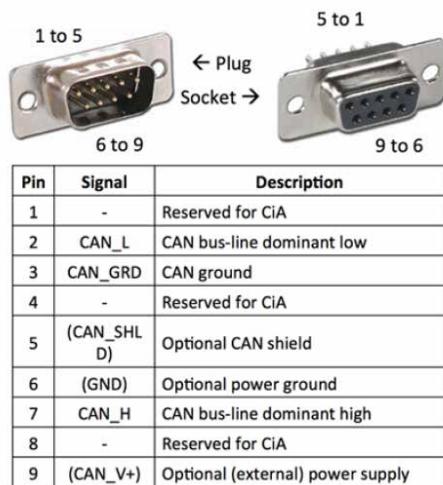


Figura 5. Conexiones de pines DB9 sugeridas.
Fuente: CAN in Automation - CiA (2018).

Las longitudes del CAN bus pueden llegar a ser de hasta 40 metros sin reducir su velocidad máxima de transmisión de datos, que es de 1 Mb/s. Sin embargo, la norma ISO-11898 permite extender los tramos del cable con una reducción aceptable en la velocidad de comunicación. La tabla 1 muestra algunas velocidades comunes y la máxima longitud del cable recomendada como se ilustra en la figura 6. Al ser un protocolo basado en macros, todos los *bits* deben atravesar toda la longitud del bus antes de que el siguiente *bit* sea transmitido (Thompson, 2018).

Velocidad del bus (kb/s)	Longitud del bus [L] (m)	Longitud del cable [l] (m)	Separación de los nodos [d] (m)
100	40	0.3	40
500	100	0.3	100
100	500	0.3	500
50	1000	0.3	1000

Tabla 1. Límites de longitud del protocolo BUS can.

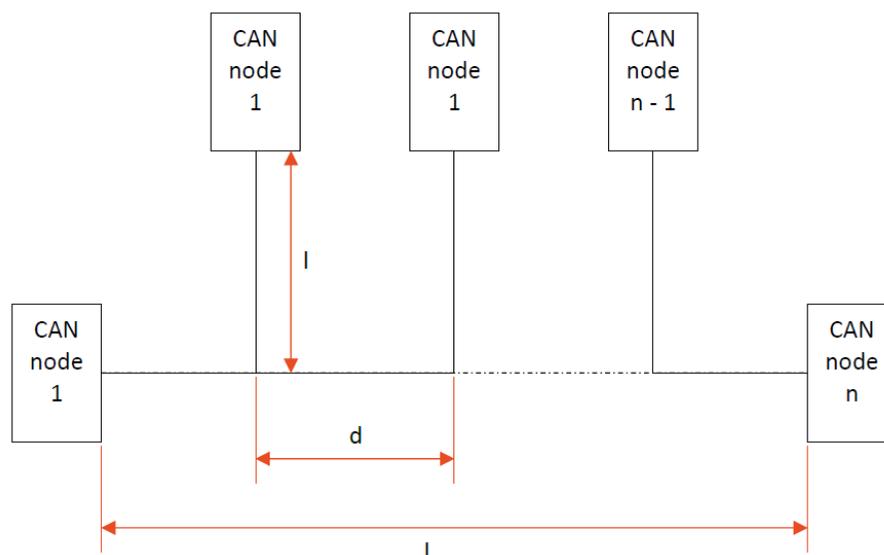


Figura 6. Dimensiones del cable para protocolo CAN bus.
Fuente: Thompson (2018).

La norma ISO 11898, en todas las formas de comunicación de datos digitales, define dos estados lógicos representados por 0 y 1. Estos, también, se definen como dominantes (0) y recesivos (1). El protocolo CAN bus es multi maestro, es decir, que no existe un maestro único que controle todo el proceso. Cada transmisor tiene una prioridad (1 o 0), y los transmisores de prioridad alta, estado dominante, tienen la capacidad de anular cualquier transmisor que produzca un estado recesivo. Lo anterior quiere decir que no hay esclavos, y la gran ventaja es que se reducen los puntos de falla (International Organization for Standardization - ISO, 2003; Thompson, 2018).

Capa de enlace de datos

La función de la capa de enlace de datos es establecer los medios de transmisión de datos en bruto a través de la capa física. El protocolo CAN bus maneja dos estructuras de comunicación. La estructura estándar maneja 11 bits, mientras que la estructura extendida maneja 29 bits. Esta información se utiliza, principalmente, para la identificación de mensajes. La figura 7a muestra la configuración estándar y en la figura 7b se puede ver la versión extendida (Corrigan, 2016).

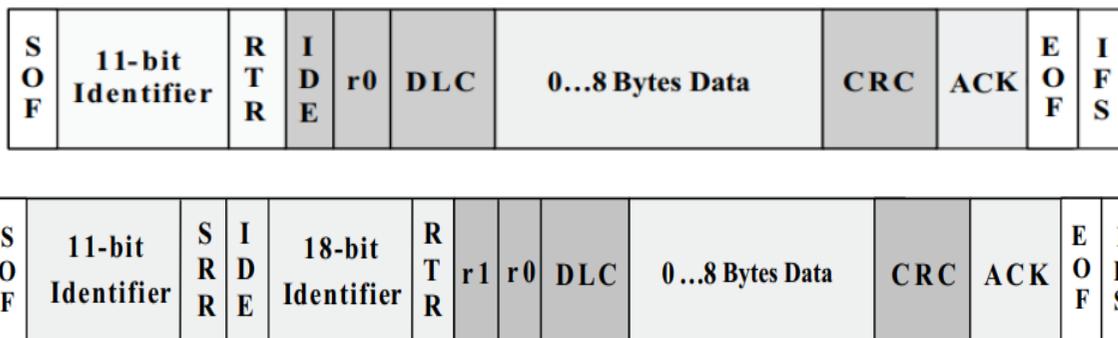


Figura 7. a) Configuración estándar (11 bits), **b)** Configuración extendida (29 bits).
Fuente: Corrigan (2016).

En la tabla 2 (Corrigan, 2016), se explican los términos utilizados en cada configuración del protocolo. Las aplicaciones de este tipo de protocolo aparecen en la industria automotriz, para la conexión de los componentes de los últimos modelos fabricados. Además, la naturaleza de las especificaciones de la capa física hace que el protocolo sea extremadamente resistente y capaz de ser implementado en entornos eléctricamente agrestes, expuestos a una alta interferencia electromagnética, con una mínima pérdida de datos y la capacidad de recuperar rápidamente los datos dañados (Markovitz y Wool, 2017).

CONFIGURACIÓN ESTÁNDAR	
Variable	Función
SOF	Bit de inicio de cuadro dominante, marca el inicio de un mensaje y sirve para sincronizar los nodos de un bus, luego de estar sin actividad.
Identifier	Establece la prioridad del mensaje. Cuánto menor sea el valor binario, mayor será su prioridad.
RTR	Bit de solicitud de transmisión remota único es dominante cuando se requiere información de otro nodo. Todos los nodos reciben la solicitud, pero el identificador (Identifier) determina el nodo especificado.
IDE	Bit de extensión de identificador dominante. Si está activo, quiere decir que se está transmitiendo un identificador CAN estándar sin extensión.
r0	Bits reservados.
DLC	Código de longitud de datos de 4 bits (DLC), contiene el número de bytes de datos que se transmiten.
Data	Se pueden transmitir hasta 64 bits de datos de aplicación.
CRC	El control de redundancia cíclica (CRC) de 16 bits (15 bits más el delimitador) contiene la suma de control (número de bits transmitidos) de los datos de aplicación precedentes para la detección de errores.
ACK	Cada nodo que recibe un mensaje preciso sobrescribe este bit recesivo del mensaje original con un bit dominante, indicando que se ha enviado un mensaje sin errores. Si un nodo receptor detecta un error y deja este bit recesivo, descarta el mensaje, y el nodo emisor repite el mensaje después de la reiteración. Esta función, de esta manera, reconoce la integridad de sus datos.
EOF	Indica el final de un mensaje y desactiva el relleno de bits, señalando un error de relleno cuando es dominante. Cuando 5 bits del mismo nivel lógico se producen sucesivamente durante el funcionamiento normal, un bit del nivel lógico opuesto se rellena en los datos.
IFS	Este espacio contiene el tiempo requerido para mover una trama recibida de manera correcta a su posición, en un área de memoria intermedia de mensajes.
CONFIGURACIÓN EXTENDIDA	
SRR	El bit de sustitución de solicitud remota (SRR) sustituye al bit RTR en la ubicación estándar del mensaje como marcador de posición en el formato extendido.
IDE	Bit recesivo en la extensión del identificador (IDE), indica que le siguen más bits de identificador. La extensión de 18 bits sigue a la IDE.
r1	Después de los bits RTR y r0, se ha agregado un bit de reserva adicional antes del bit DLC.

Tabla 2. Variables que aparecen en las configuraciones estándar y extendida.

Protocolo CANopen

El protocolo CANopen es una capa de aplicación diseñada para operar sobre redes que cumplan con la norma ISO-11898 en un entorno industrial (International Organization for Standardization - ISO, 2003). Generalmente, las capas físicas y de enlace de datos son las mismas que las del CAN bus. CANopen se basa en las propiedades de priorización de CAN bus como parte integral del diseño del protocolo. CANopen

especifica varios tipos de comunicación y los prioriza en bloques mediante el uso de los identificadores de CAN bus. Este identificador se conoce como el COB-ID en CANopen. El COB-ID se incrementa posteriormente con un NodeID, implementando así, no solo la priorización de un tipo de mensaje, sino también la priorización de las direcciones de los nodos como una característica secundaria (March *et al.*, 2019; Thompson, 2018).

Modelo de dispositivo

Un modelo de dispositivo constituye un aspecto central en CANopen e incorpora la comunicación de la capa de enlace de datos dentro de un diccionario de objetos, el cual es legible de manera organizada por la capa de aplicación. La figura 8 representa la organización general del diccionario de objetos de un dispositivo, referenciado tanto a los objetos de comunicación como a la aplicación. La interfaz de comunicación realiza la función de transmisor de datos hacia las cuatro interfaces de comunicación estándar. El diccionario de objetos ofrece una interfaz entre la aplicación y la comunicación, proporcionando acceso al objeto del dispositivo. La aplicación actúa como un puente que conecta el dispositivo esclavo CANopen con la estación maestra. El PDO proporciona un canal

de acceso directo al objeto de la aplicación del dispositivo, utilizado para la transmisión de datos de corta duración en tiempo real con alta prioridad, siguiendo el modelo de comunicación productor/consumidor. El PDO emitido por el dispositivo nodal se denomina PDO de transmisión (TPDO) del dispositivo, mientras que el PDO recibido es el PDO receptor (RPDO) del dispositivo. Cada PDO se describe mediante dos objetos en el diccionario de objetos: los parámetros de comunicación y los parámetros de mapeo. El parámetro de comunicación PDO especifica el COB-ID, el tipo de transmisión, el tiempo de inhibición y el tiempo de evento empleados; el parámetro de mapeo PDO se utiliza para establecer la relación de los datos dentro del mensaje PDO y determinar la ubicación de los datos a transmitir en el campo de datos del mensaje CAN (Ge *et al.*, 2019).

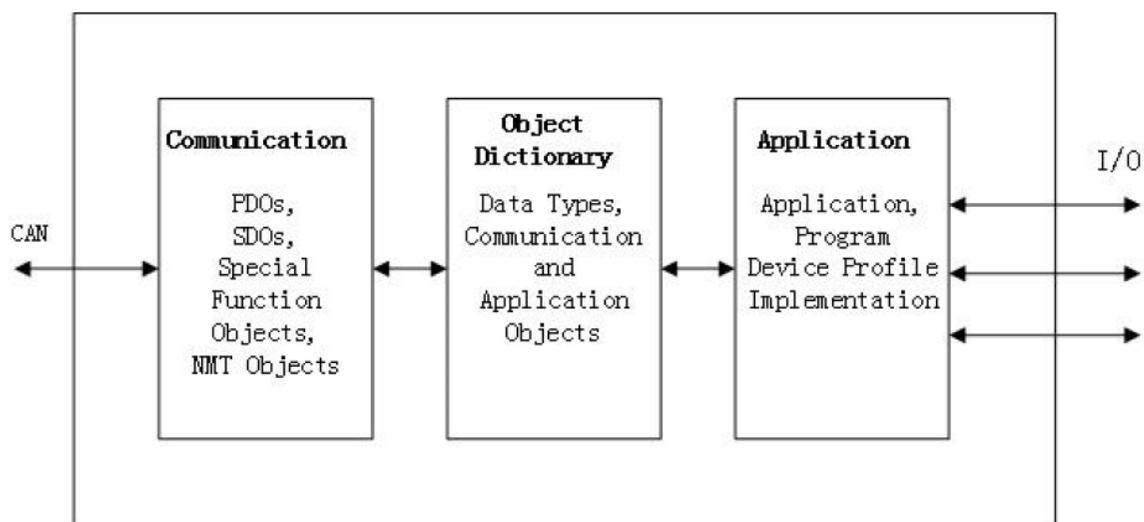


Figura 8. Modelo del dispositivo.

Fuente: Ge *et al.* (2019).

CANopen permite un máximo de 127 nodos. Algunos servicios que funcionan en la red CANopen, como el Protocolo NMT, requieren la implementación de un nodo maestro. El nodo maestro no requiere necesariamente una dirección en la red, pero esto depende de los servicios alojados en este nodo. Objeto de

datos de servicio (SDO) de comunicación, se basa en los servicios de protocolo de dominio CAL/CMS existentes; los SDO se utilizan principalmente para la parametrización y configuración del dispositivo. Comunicación de objetos de datos de proceso (PDO), es utilizado durante el funcionamiento normal de

la red para transferir datos en tiempo real con poca o ninguna sobrecarga de procesamiento de protocolos. Gestión de la red (funciones NMT): estos telegramas se utilizan para coordinar el funcionamiento del dispositivo y de la red de manera controlada y bien definida. Mensajes de formato predefinido, estos telegramas apoyan la aplicación de la sincronización y los mecanismos de notificación de errores (Ge *et al.*, 2019).

Diccionario de objetos

Un diccionario de objetos debe ser implementado en cada dispositivo para que sea compatible con CANopen. Estos se definen para un dispositivo específico por un archivo EDS que es liberado por el fabricante y usualmente disponible de forma gratuita para ser descargado o proporcionado junto con el producto en la entrega. Los archivos EDS contienen toda la información necesaria para verificar el dispositivo y lograr una comunicación exitosa. Cada entrada del archivo EDS contiene un puntero a la información descrita por medio del índice y el subíndice. Los archivos EDS se implementan en el nodo maestro.

Aplicaciones del protocolo CAN

Los automóviles modernos tienen múltiples computadoras “unidades de control electrónico” (ECUs). Estas ECUs controlan todos los aspectos del funcionamiento del auto: desde el motor, los controles de freno y dirección, hasta los sistemas de entretenimiento del coche. Las ECUs están conectadas entre sí en una red que normalmente utiliza el protocolo de bus CAN. El bus CAN es un protocolo extremadamente simple, sin ningún componente de seguridad: fue diseñado bajo el supuesto de que todas las UCE son legítimas, fiables y funcionan de acuerdo con sus especificaciones (Markovitz y Wool, 2017). Especialmente en Europa, las redes

CANopen se han utilizado en sistemas de control industrial orientados al movimiento, en el transporte público, en vehículos todoterreno (Jeep), entre otros (Chen *et al.*, 2011).

4. CONCLUSIONES

El protocolo CANopen toma varias características del protocolo CAN bus y opera bajo esas condiciones, pero, en términos generales, son muy semejantes. El protocolo CAN es un protocolo que trabaja con 3 capas del modelo OSI/ISO: la capa física, la capa de enlace de datos y la capa de aplicación. Es un protocolo multi maestro, es decir, existen elementos recesivos y dominantes. Los elementos recesivos se representan por un 1 lógico, mientras que los datos dominantes se representan por un 0 lógico. Se caracteriza por el hecho de que, a mayor distancia de operación, menor será la velocidad de transmisión de datos. La velocidad máxima de operación es 1 Mb/s. Una de las grandes ventajas de este protocolo es su simplicidad, su confiabilidad y la prioridad entre los sensores, actuadores y dispositivos inteligentes. Algunas de las aplicaciones industriales han sido orientadas a la industria automotora, en vehículos normales, todoterrenos y sistemas de transporte público.

5. REFERENCIAS

- Angulo Angulo, F. y Miranda Zambrano, M. (2018). *Medidor de velocidad de protocolo CAN*. Instituto tecnológico y de estudios superiores de occidente.
- CAN in Automation (CiA). (2018). *CAN in Automation (CiA)*. <https://www.can-cia.org/can-knowledge/can/systemdesign-can-physicallayer/>
- Chen, Z., Wang, F. y Jia, C. (2011). Research of CANopen Slave Node Based on Embedded System. *Advanced*

- Materials Research*, 220, 629-632. <https://doi.org/10.4028/www.scientific.net/AMR.219-220.629>
- Corrigan, S. (2016). *Introduction to the Controller Area Network (CAN)*. www.ti.com
- Ge, J., Gao, Y., Li, Z. y Lan, Z. (2019). *Application of CANopen in Servo Motor Control*. [ponencia]. 3rd International Conference on Computer Engineering, Information Science & Application Technology (ICCIA 2019) (Vol. 90, pp. 495-500). Atlantis Press.
- Guerrero, V. y Martínez, L. (2010). *Comunicaciones Industriales*. S. A. MARCOMBO.
- International Organization for Standardization (ISO). International Standard ISO 11898-1, 1 § (2003).
- Kumar, S. (2014). *Fieldbus and Networking in Process Automation*. Boca Raton: CRC Press.
- March, Z., Hung, C., Lee, R. C. L., Huang, B. y Yu, S. (2019). Multi-Motor Synchronous Control with CANopen. *International Journal of Computational Intelligence Systems*, 5(4), 3-7. <https://doi.org/https://doi.org/10.2991/jrnal.k.190220.006>
- Markovitz, M. y Wool, A. (2017). Field classification, modeling and anomaly detection in unknown CAN. *Vehicular Communications*, 1, 1-10. <https://doi.org/10.1016/j.vehcom.2017.02.005>
- Oliva Alonso, N. (2013). *Redes de comunicaciones industriales*. UNED.
- Thompson, S. (2018). *Application of Controller Area Network Bus and CANopen Protocol in Industrial Automation*. Murdoch University.
- Wilamowski, B. M. e Irwin, J. D. (2018). *Industrial Communication Systems* (1.ª Ed.). <https://doi.org/https://doi.org/10.1201/9781315218434>