

CANopen

Manual CANOpen

Manual de introducción al CANOpen, bus de campo para máquinas e instalaciones



ADVERTENCIA

Los productos presentados en este manual son susceptibles de evolución en cuanto a sus características de presentación, de funcionamiento o de utilización. Su descripción en ningún momento puede revestir un aspecto contractual.

El Instituto Schneider Electric de Formación, acogerá favorablemente cualquier solicitud con fines didácticos exclusivamente, de utilización de gráficos o de aplicaciones contenidas en este manual.

Cualquier reproducción de este manual está totalmente prohibida sin la autorización expresa del Instituto Schneider Electric de Formación.

Manual de formación CANOpen

Creado: Instituto Schneider Electric de Formación

Bac de Roda 52, Edificio A – 1ª Planta

Fecha: 11 de Noviembre de 2008

Versión: 1.0

SCHNEIDER ELECTRIC ESPAÑA

Índice

1. Presentación del manual de formación	p. 7
2. Introducción a CANOpen	p. 8
2.1. CANOpen: un estándar abierto	p. 8
2.2. CAN y sus inicios	p. 9
2.3. Características generales de CANOpen	p. 11
2.4. CANOpen: transparente para Ethernet	p. 12
2.5. Insensitivo a interferencias electromagnéticas	p. 13
2.6. Transmisión fiable	p. 13
2.7. Detección de errores	p. 14
2.8. Rendimiento y flexibilidad	p. 15
2.9. Facilidad de instalación	p. 16
3. Conexionado	p. 17
3.1. Diseño e instalación del bus	p. 17
3.2. Niveles eléctricos: bus y nodos	p. 24
3.3. Hardware utilizado en la implementación del bus	p. 27
3.4. Configuración de los dispositivos	p. 32
4. Funcionamiento de CANOpen	p. 33
4.1. Principios de funcionamiento	p. 33
4.2. Tramas y protocolos CAN	p. 34
4.3. CSMA/CD+AMP	p. 35
4.4. Arbitraje del bus	p. 36
4.5. Mecanismos de detección de errores	p. 37
4.6. Estados de error	p. 38
4.7. Protocolo CANOpen	p. 40
4.8. Diccionario de objetos – Perfiles – Fichero EDS	p. 41
4.9. COB-ID – Arbitraje del bus	p. 44
4.10. Objetos de comunicación	p. 46

5. Dispositivos CANOpen	p. 57
5.1. E/S distribuidas STB	p. 57
5.2. E/S distribuidas OTB	p. 58
5.3. Variador de velocidad ATV31	p. 60
5.4. Variador de velocidad ATV61-71	p. 62
5.5. Control de movimiento Lexium 05	p. 64
5.6. Módulo para arrancador motor TeSysU	p. 66
5.7. Encoder con conexión a CANOpen	p. 68
 6. Analizador de tramas CANOpen	 p. 70
6.1. Introducción	p. 70
6.2. Instalación del sniffer de CANOpen: PEAK	p. 71
6.3. Utilización del sniffer de CANOpen: PEAK	p. 75
 7. Glosario	 p. 78

1. Presentación del manual de formación

Con el afán de ofrecer un mejor servicio, el **Instituto Schneider Electric de Formación (ISEF)** edita el Manual de introducción al bus CANOpen con el software **UNITY Pro** para la configuración y la programación de **autómatas** de marca **TELEMECANIQUE** y de gama **MODICON**, el software **Sycon** (para la configuración de bus CANOpen con PLCs Modicon Premium) y el software **Peak** (sniffer de CANOpen para visualizar tramas CANOpen)

La finalidad de este manual es la de proporcionar al usuario una herramienta que le permita tanto introducirse como ampliar sus conocimientos en la **configuración, programación de funciones y diagnóstico del bus CANOpen** con los PLCs Modicon M340 y Modicon Premium, y dispositivos comunicables en CANOpen.

Dado que el campo de aplicación de las soluciones implementadas con el bus CAN en el mundo industrial es muy amplia y que la flexibilidad que permite su **configuración y programación** es muy grande, el manual siguiente le ofrece una amplia visión de las posibilidades tanto a nivel de configuración, como de programación y depuración que tienen los autómatas programables de gama Modicon con Unity Pro.

2. Introducción a CANOpen

2.1. CANopen: un estándar abierto

CANopen es una red abierta admitida por más de 400 empresas de todo el mundo y promocionada por CAN in Automation. CANopen está normalizada en EN 50325-4 e ISO 15745-2 por la descripción de su dispositivo.



Logotipo de canopen

Schneider Electric trabaja estrechamente con CAN in Automation. Esta colaboración permite a Schneider Electric ofrecer una red verdaderamente abierta cumpliendo las normas existentes definidas por todos los miembros de CAN in Automation.

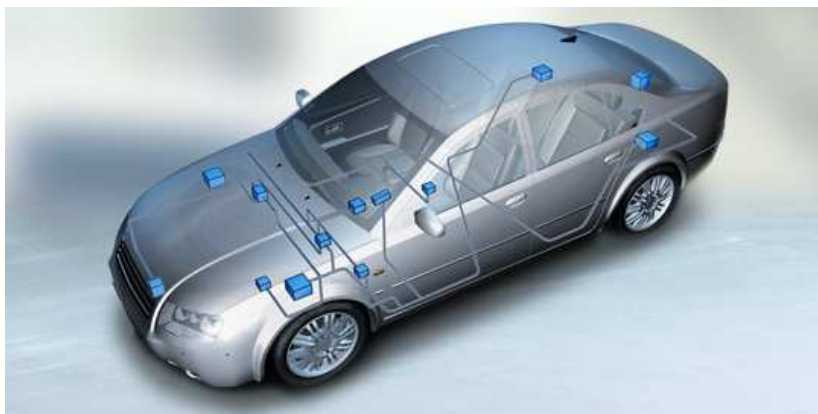
Al ser un miembro activo, Schneider Electric ajusta los requisitos del sistema y los clientes a las normas, de forma que las demás empresas también implementan el mismo comportamiento y las mismas funciones. Es la mejor forma de garantizar la interoperabilidad y apertura de nuestros productos y sistemas.

Schneider Electric está muy involucrada en los grupos de trabajo relevantes para las arquitecturas, sistemas y productos de máquinas e instalaciones. Schneider Electric tiene una posición de líder en:

- Perfil de variadores y movimiento.
- Pruebas de conformidad.

2.2. CAN y sus inicios

CANopen es un protocolo de nivel de aplicación que se basa en CAN (controller area network). Bosch diseñó CAN para aplicaciones de automoción a principios de la década de los ochenta. Su uso en el sector de la automoción ha demostrado su idoneidad en entornos difíciles.



CAN-bus en un coche

Historia:

1980: La industria del automóvil muestra la necesidad de un bus barato, de tiempo real y altamente robusto para comunicar diferentes componentes electrónicos. Can define únicamente las capas 1 y 2 del modelo ISO.

1983: Junto con diferentes Universidades Alemanas, el fabricante BOSCH desarrolla el protocolo CAN = "Controller Area Network"

1985: Los primeros chips de CAN los comercializa INTEL



Logo de intel

1986: Prototipos iniciales se presentan en Detroit (EEUU)

1987: Los mayores fabricantes de circuitos integrados (Intel, Motorola,...) ofrecen una gama completa de microcontroladores que integran CAN

1989: Primeras aplicaciones industriales

De 1987 a 1991: Aparecen diferentes organizaciones que promocionan CAN en el mundo del automóvil (SAE, OSEK), y en aplicaciones industriales (CIA = “CAN in Automation”, CAN en la automatización) <http://www.can-cia.de/>



Logo de can in automation

1992: MERCEDES usa un bus CAN a 500 kbps para enlazar 5 submódulos electrónicos en un Clase S.



Dispositivos CAN y Mercedes S (año 1992)

1993: CAN de Alta Velocidad (1 Mbps/Identificadores de 11 bits) estándar ISO 11898 = CAN 2.0 A. Publicación por la CiA de las especificaciones CAL=Can Application Layer describiendo los mecanismos de transmisión sin definir cuando y como utilizarlos.

1995: identificadores de 29 bits (CAN2.0 B) - Publicación por la CiA del perfil de comunicaciones DS-301 = CANopen

1996: CAN se usa en la mayoría de coches de gama alta europeos.

1997: 300 compañías miembros de CIA (CAN en Automatización).

2001: La CIA publica el perfil DS-304 el cual se puede usar para integrar componentes de nivel de seguridad 4 en un bus CANopen estándar (CANsafe).

2.3. Características generales de CANOpen

CANOpen es un bus de tipo serie, basado en CAN, que trabaja con dispositivos que disponen de un transceptor CAN (interfaz conexionado/señal) y un controlador CAN (parte control), tal como especifica la norma ISO11898.

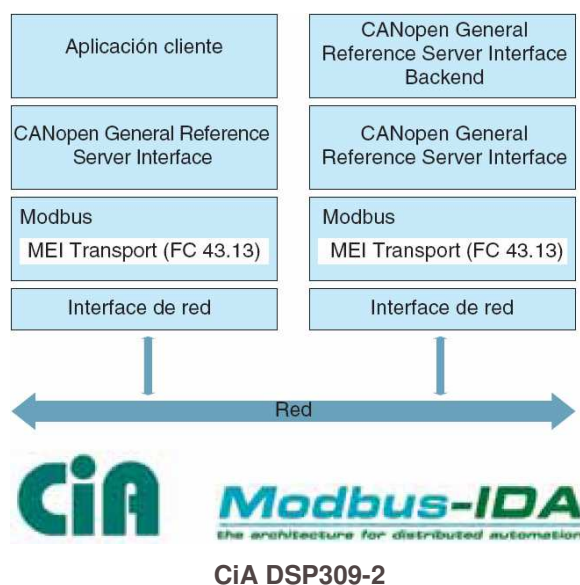
Se puede decir que CANOpen es un compuesto entre CAN y una serie de servicios de comunicación. Algunos datos:

- 2 pares trenzados (blanco, azul, negro, rojo)
- Protocolo multimaestro (Solo un maestro en bus CANOpen y dispositivos de Schneider Electric)
- Topología: en serie y/o con derivaciones
- Finales de línea (2 resistencias de 120 ohmios por segmento)
- Velocidad: 1Mbit a 40m // 50kbit a 1Km
- Muy robusto y económico
- productor/consumidor y maestro/esclavo
- 64 nodos máximo por segmento y 127 nodos en el bus.
- Funciones de comunicación

2.4. CANopen: transparente para Ethernet

CAN in Automation y Modbus-IDA han trabajado estrechamente para emitir un estándar que permite la transparencia entre CANopen y Modbus TCP/IP.

Esta colaboración derivó en la especificación CiA DSP309-2, que ofrece una definición estándar para la comunicación entre Modbus TCP/IP y una red CANopen.



En la especificación se definen los servicios de asignación de forma que los dispositivos CANopen se puedan comunicar en una red Modbus TCP/IP a través de un dispositivo gateway.

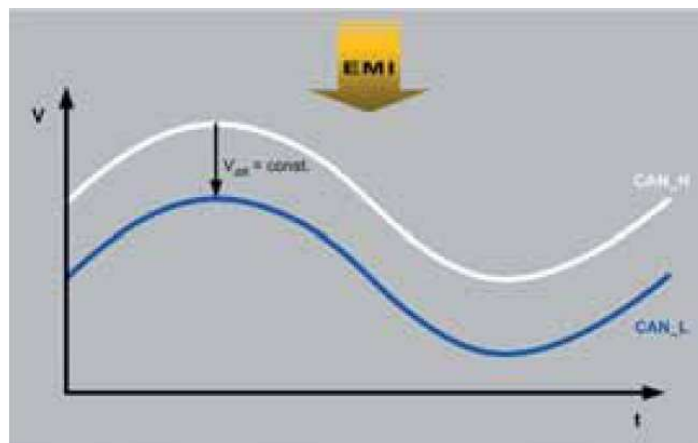
El acceso a la información de un dispositivo CANopen se admite tanto en lectura como en escritura, junto con toda una variedad de funciones de control de dispositivos.

Esta especificación es el primer estándar que permite la instalación de un estándar de comunicación abierto entre Modbus TCP/IP y CANopen. Orienta la solución de red de Schneider Electric hacia una mejor integración, diagnóstico y

configuración en aplicaciones distribuidas. Permite que las máquinas y las instalaciones se conecten perfectamente a una planta Ethernet que combine las ventajas de cada red en su propia área específica.

2.5. Insensitivo a interferencias electromagnéticas

Se debe a las altamente resistentes funciones EMI (**Electromagnetic interference**) de CANopen. Esto permite a una máquina o instalación funcionar de forma precisa incluso cuando las interferencias son elevadas. Los pequeños marcos de CANopen y la conexión "CAN ground" que ofrecen el mismo potencial para cualquier dispositivo conectado a la red, protegen frente a EMI.



Nivel de tensión CAN-H y CAN-L

2.6. Transmisión fiable

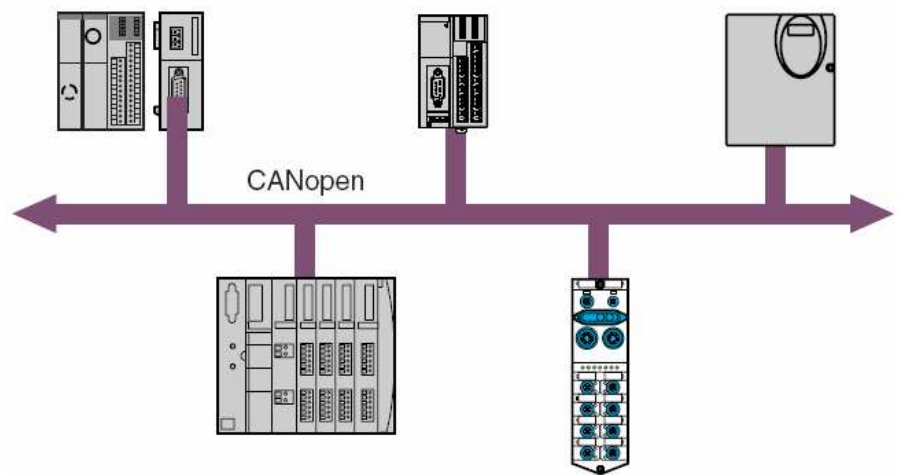
El acceso a la red de CANopen no es destructivo. Cuando cualquier dispositivo CANopen transmita sus datos, el sistema genera automáticamente y tiene en cuenta la prioridad del mensaje.

La pérdida de telegramas debido a una colisión es imposible y se evita perder tiempo hasta el siguiente estado libre de la red. CANopen ofrece una transmisión de datos totalmente fiable.

Es una de las razones por las que las redes CANopen se utilizan en equipos médicos y como base para las redes de seguridad.

2.7. Detección de errores

Las interrupciones son siempre prolongadas y costosas. CANopen está perfectamente diseñado para aumentar el tiempo de funcionamiento lo máximo posible. Con una distancia de Hamming de 6, CANopen cuenta con una detección de errores y un mecanismo de corrección óptimos. Con una probabilidad de un error no detectado cada 1.000 años, la red es la más fiable para las máquinas y las instalaciones.



Arquitectura típica

En caso de que la red presente una condición de error, el watchdog es la primera posibilidad para controlar el estado del dispositivo. Además, CANopen ofrece una clara información de diagnóstico. Cada mensaje de diagnóstico contiene el origen y el motivo del fallo, lo que permite reaccionar rápidamente y reducir

el tiempo de inactividad. La información de diagnóstico adicional se aplica para mejorar el diagnóstico de dispositivos CANopen complejos y para admitir los que se encargan de mantener la red. También se encuentra disponible un historial de fallos de ayudar a detectar fallos aleatorios.

2.8. Rendimiento y flexibilidad

El principal motivo para utilizar una red es el rendimiento y la flexibilidad para adaptar la red exactamente a los requisitos de la aplicación. CANopen ofrece una única función para adaptar la transmisión de datos.

Basado en el modelo productor/consumidor, CANopen permite la comunicación cíclica, de cambio de estado, de igual a igual y la difusión de la transmisión de datos. Esto significa que transmite los datos únicamente cuando es necesario o en una base de tiempo específica.

Los objetos de datos de proceso se pueden configurar individualmente. Los parámetros se pueden cambiar en funcionamiento.

CANopen es muy flexible y la respuesta de la red es rápida. En menos de 1 ms, se puede prestar servicio a 256 puntos de E/S digitales a 1 Mbps.

Además de esta rápida respuesta, la asignación de prioridad de los mensajes se puede modificar fácilmente. Así, CANopen ofrece las funciones necesarias para adaptar la transmisión de datos a los requisitos de la aplicación.

2.9. Facilidad de instalación

Ofrece un límite equipotencial integrado en el cable. Por lo tanto, no se necesita ningún cable adicional ni cable plano de cobre trenzado para obtener el mismo potencial en todos los dispositivos de la red. Los costes de instalación se reducen en gran medida.

Además, los chips y los componentes para CANopen se producen cantidades muy elevadas, de lo que se derivan chips y componentes con precios optimizados.

Se puede prever una ventaja de entre el 10 y el 20% en comparación con otros buses de campo de la máquina.

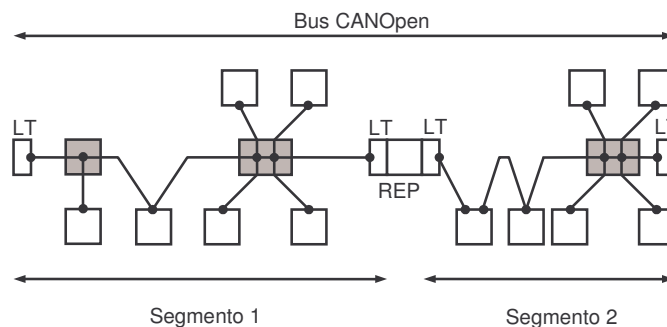
3. Conexionado

3.1 Diseño e instalación del bus

A la hora de diseñar una instalación de CanOpen es necesario conocer la siguiente información:

a. Topología del bus.

Un bus CANOpen se puede dividir en distintos segmentos. Cada segmento se puede conectar por medio de un repetidor y debe tener unas resistencias de terminación (LT) en los dos extremos.



Bus CANOpen con 2 segmentos

Tipos de conexión entre nodos:

- Por derivación (drop), mediante drops conectados a Taps de puerto simple o múltiple.
- En serie (daisy-chain), con un conector (Nodos 2, 8) o con dos conectores (Nodo 7). ATENCIÓN: El uso de dos conectores no se recomienda ya que el cambio de un equipo provocaría el corte del bus.

Para la conexión de los diferentes nodos existen cables preensamblado (RJ45, SUB-D 9 contactos, hilos libres,...), cables CANOpen estándar o flexible (de 50m, 100m o 300m), conectores, adaptadores y varios tipos de taps:



TSX CAN KCD F90T



Conector IP20 hembra SUB-D9 contactos de CANopen con ángulo a 90°. Integra un microconmutador para establecer la terminación del bus



TSX CAN KCD F180T



Conector IP20 hembra SUB-D9 contactos de CANopen con ángulo a 180°. Integra un microconmutador para establecer la terminación del bus



TSX CAN KCD F90TP



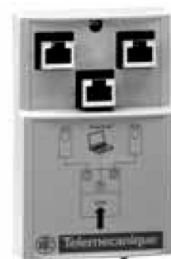
Conector IP20 hembra SUB-D9 contactos de CANopen con ángulo a 90° y conector adicional SUB-D de 9 contactos para conectar en el bus un PC. Integra un microconmutador para establecer la terminación del bus.



VW3 CAN A71



Adaptador CANopen SUB-D de 9 contactos para RJ45



VW3 CAN TAP2



Caja de conexión CANopen con 2 puertos RJ45. El tercer puerto RJ45 sirve para conectar un PC con PowerSuite. Incluye un switch para establecer la terminación del bus (switch accesible abriendo la tapa).



TSX CAN TDM4



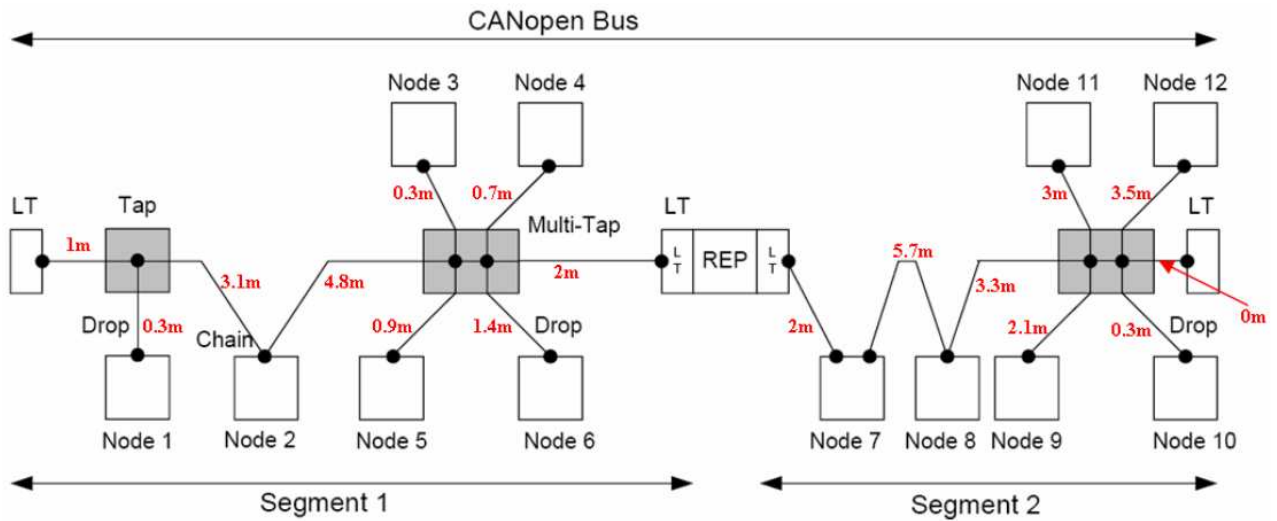
Caja de conexión CANopen con 4 puertos SUB-D de 9 contactos. Permite la conexión de 4 derivaciones en el cable de canalización, incluye un switch para establecer la terminación del bus (switch visible en la parte frontal).

También existen accesorios de conexión IP67 y componentes separados tal como conectores en Y, en T,...

Es necesario un esquema general de la topología implementada incluyendo las distancias relacionadas entre los diversos elementos, tanto de bus principal como de todas las derivaciones y segmentos existentes.

Son especialmente importantes los siguientes datos:

- Distancia máxima entre los dos nodos más alejados (tanto los más alejados del bus en general como de los más alejados en un mismo segmento).
- Verificar la longitud de cada segmento y el número de nodos conectados en él.
- Verificar la longitud de las derivaciones con respecto bus principal.
- Verificar el número de nodos (totales y de cada segmento en particular).



- Longitud total del bus (incluyendo bus principal y derivaciones de todos los segmentos): 34.4m
- Longitud total del segmento 1 (incluyendo bus principal y derivaciones del segmento): 14.5m
- Longitud total del segmento 2 (incluyendo bus principal y derivaciones del segmento): 19.9m
- Distancia máxima entre nodos del bus: 24.7m (entre nodos 1 y 12)
- Distancia máxima entre nodos de segmento 1: 9.6m (entre nodos 1 y 6)
- Distancia máxima entre nodos de segmento 2: 12.5m (entre nodos 7 y 12)

b. Restricciones y limitaciones básicas a tener en cuenta en la topología del bus:

- Limitaciones con respecto a la longitud del bus principal y de las derivaciones:

En CanOpen la prioridad de las tramas se gestionan mediante la colisión entre niveles dominante y recesivo de la línea de comunicación. Esta colisión se debe resolver durante la transmisión de 1 bit, lo cual limita el retardo en la propagación de

la señal entre nodos y esto, a su vez, limita la máxima distancia de bus admisible en una red CAN, que, como restricción principal, depende de la velocidad del bus (ratio distancia/velocidad del bus):

Veloc. de bits (kbps)	1.000	800	500	250	125	50	20	10
Longitud máx. (m)	20	40	100	250	500	1.000	2.500	5.000

NOTA: Estos valores están calculados para casos extremos, así pues en alguna documentación podemos encontrar distancias mayores, aplicables solo en caso de que el dispositivo no disponga de aislamiento galvánico.

Las derivaciones del bus crean reflexión en la señal, y, por tanto, posibles errores de comunicación; es por eso que es necesario limitar su longitud. Como norma general, las limitaciones aplicables son:

Velocidad en baudios (kbps)	1.000	800	500	250	125	50	20	10
L máx. (m)	0,3	3	5	5	5	60	150	300
ΣL máx. (m) config. estrella local	0,6	6	10	10	10	120	300	600
Interval mín. (m) $0,6 \times \Sigma L$ local	—	3,6	6	6	6	72	180	360
ΣL máx. (m) en todos los buses	1,5	15	30	60	120	300	750	1.500

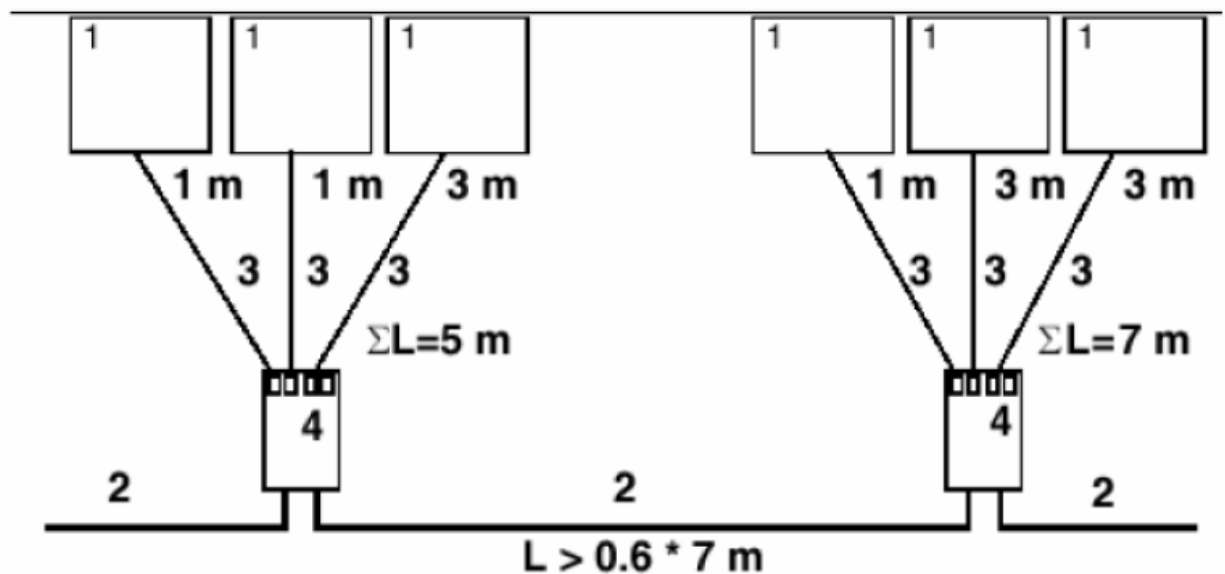
Interval mín. debe ser superior a un 60% de la suma de las longitudes de todas las derivaciones cableadas en la caja de derivación.

Donde:

- **L máx.** es la longitud máxima del cable de cada derivación,
- **ΣL máx.** es el valor máximo para la suma de la longitud de todas las derivaciones en una sola caja de derivación.
- **Interval mín.** es la distancia mínima entre las dos cajas de derivación.
- **ΣL máx.** es el valor máximo para la suma de todas las derivaciones del segmento.

Como regla general, la suma de todas las derivaciones dependientes de un mismo punto (tap) no debe exceder 2 veces la longitud de la derivación más larga (por ejemplo, si la derivación más larga es de 5m, la suma de todas las derivaciones dependientes del mismo punto no debe superar los 10m).

Ejemplo:



Donde:

- 1 son los dispositivos CANOpen.
- 2 son los cables principales de CANOpen.
- 3 son los cables de derivación.
- 4 son las cajas de derivación.

NOTA: Para casos particulares o para buses con alimentación distribuida, pueden existir diferentes restricciones que en cuanto a distancias permitidas, tanto para el bus principal como para las derivaciones (consultar el documento **CanOpen: Physical & Cabling Guide** -ver anexos: **Documentación Relacionada**-).

- Limitaciones con respecto al número de nodos permitidos en el bus.

Además de las limitaciones generales de longitud, se aplican los siguientes límites:

- Número máximo de nodos por segmento: 64.
- **NOTA:** Cuando el bus se compone de más de un segmento y se usan elementos repetidores intermedios hay que tener en cuenta nuevas restricciones que en cuanto número de nodos permitidos (consultar el documento **CanOpen: Physical & Cabling Guide** -ver anexos: **Documentación Relacionada**-)
- **NOTA:** Con los cables Telemecanique se aplican las siguientes restricciones:
 - 16 nodos en un máximo de 205 metros.
 - 32 nodos en un máximo de 185 metros.
 - 64 nodos en un máximo de 160 metros.

En el caso de que el número de equipos supere el máximo permitido para un segmento, será necesario utilizar repetidores. Los repetidores añaden un retardo de propagación en el bus, que restan longitud útil de bus. Un repetidor con un retardo de 150ns equivale a 30m de cable, disminuye la distancia disponible de bus en 15m.



Foto de un repetidor

Display	Transmit (2 green LEDs), defective segment (2 red LEDs)
CAN bus interface	ISO/IS 11898-2 with CAN choke. Terminals (CAN-High, CAN-Low, CAN-GND). CAN 1, CAN 2 and power supply are galvanic isolated against each other. CAN termination resistors are integrated (can be disabled).
Baudrate	Up to 888 kbps
Delay	200 ns (corresponds ~40 m (~120ft.) bus length)

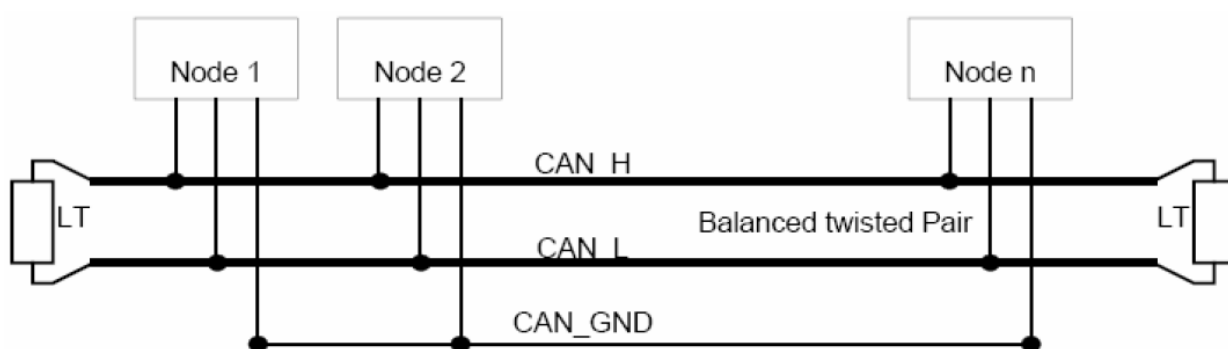
Datos técnicos del repetidor

3.2. Niveles eléctricos: bus y nodos

El bus CanOpen es bastante restrictivo en lo que se refiere a los niveles eléctricos de funcionamiento; de no respetar éstos niveles, que tienen unas tolerancias no muy grandes, fácilmente pueden aparecer problemas de comunicación.

Para verificar estos datos, basta con realizar un test del cableado físico del bus, para lo cual basta como herramienta un polímetro o téster.

Simplificándola físicamente, el esquema eléctrico de una red CanOpen es el siguiente:



LT: Line Termination

a. Test de cableado:

Idealmente la mayor parte de mediciones de este test se ha de llevar a cabo con todos los nodos del bus (tanto el maestro como los esclavos) desconectados ya que la impedancia de entrada de cada uno de ellos podría falsear las medidas.

Como mal menor, y si ésta desconexión no es posible, se puede realizar con los nodos sin tensión: si todos ellos respetan la muy

alta impedancia de entrada que se le supone a cualquier dispositivo Can, la medida resultante será muy cercana a la ideal, e igualmente válida.

Las mediciones a realizar y los resultados que se han de obtener son:

- a. Verificar la correcta conexión y el valor de los finales de línea:
 - Idealmente han de ser de 120Ω / $1/4W$ (tolerancia máxima de 5% para resistencias de $1/4W$).
- b. Verificar la resistencia entre las líneas del bus CAN_H y CAN_L:
 - Idealmente ha de ser de 60Ω
 - Si es mayor de 65Ω , comprobar la continuidad de señal en el bus.
 - Si es menor de 50Ω , comprobar que no haya cortocircuito entre las señales CAN_H y CAN_L del bus.
- c. Verificar la medida entre CAN_H y:
 - CAN_L: idealmente 60Ω
 - CAN_GND: idealmente $>1M\Omega$
 - CAN_SHLD: idealmente $>1M\Omega$
 - GND: idealmente $>1M\Omega$
- d. Verificar la medida entre CAN_L y:
 - CAN_H: idealmente 60Ω
 - CAN_GND: idealmente $>1M\Omega$
 - CAN_SHLD: idealmente $>1M\Omega$
 - GND: idealmente $>1M\Omega$
- e. Verificar que todos los dispositivos tienen su tierra correctamente conectada. Idealmente, dicha tierra ha de ser equipotencial en todo el bus y, por tanto, para cada uno de los dispositivos conectados.

(para más detalles consultar el documento **CanOpen: Physical & Cabling Guide** -ver anexos: **Documentación Relacionada**-).

f. Verificar la conexión de la malla del cable Can: dicha malla se ha de conectar correctamente en todos los elementos del bus, bien sean elementos pasivos (por ejemplo las derivaciones o taps), bien sea en los dispositivos activos o nodos del bus (para esto consultar la documentación específica de cada dispositivo para éste punto).

o **NOTA:** el correcto conexionado de la malla es especialmente importante en aquellos dispositivos que tengan un alto grado de emisión EMC (por ejemplo, los variadores de frecuencia). Es importante seguir estrictamente las recomendaciones y las restricciones en cuestión de conexionado y protección contra radiaciones radiadas/conducidas para éstos elementos.

g. Verificar los puntos de conexión la malla del bus a tierra: idealmente, se puede conectar tierra malla del bus a tierra en diversos puntos considerando que existe una equipotencialidad entre ambos; si no es así, se ha de conectar la malla a sólo un punto de tierra 'general', es decir, evitado que éste punto de tierra sea el específico de ningún elemento concreto del bus, sea activo o pasivo.

h. Verificar la medida entre CAN_V+ y CAN_GND:

- En redes sin alimentación distribuida: 0Vdc
- En redes con alimentación distribuida: 24Vdc (para más detalles consultar el documento **CanOpen: Physical & Cabling Guide** - ver anexos: **Documentación Relacionada**-)

i. Verificar el aislamiento entre CAN_H / CAN_L y CAN_GND

- Idealmente 500VACrms o 700VDC.

3.3. Hardware utilizado en la implementación del bus.

a. Cable:

A nivel de hardware, la primera prescripción que encontramos es la del cable a utilizar. Las características de éste cable han de ser:

Number of Pairs min	2	Required - shall
Pair A Conductor gauge	AWG 22	
Pair A Linear Resistance	55 Ω /km	
Pair B Conductor gauge	AWG 24	Required - shall
Pair B Linear Resistance	90 Ω /km	Required - shall
Pair B Characteristic Impedance	120 Ω	Required - shall
Pair B Capacitance	40 pF/m	
Pair B Propagation min	66% x C	Required - shall
Pair B Attenuation 500 kHz typ	1,64 dB/100m	
Pair B Attenuation 1 MHz typ	2,30 dB/100m	
Shield	Aluminum foil + tinned copper braid + drain	Required - shall
Shield Resistance	$\leq 7,6 \Omega/\text{km}$	
Capacitance Conductor / Shield	75 pF / m	
Sheath Color	Magenta RAL 4001	Required - shall
Operating Temperature	-10°C +80°C	
Overall Diameter	8 \pm 2mm	
Color pair A	Red + Black	Required - shall
Color pair B	White + Blue	Required - shall
Connection pair B white	CAN_HIGH	Required - shall
Connection pair B blue	CAN_LOW	Required - shall
Connection pair A black	CAN_GND	Required - shall
Connection pair A red	CAN_V+	Required - shall

Tabla de definición del cable

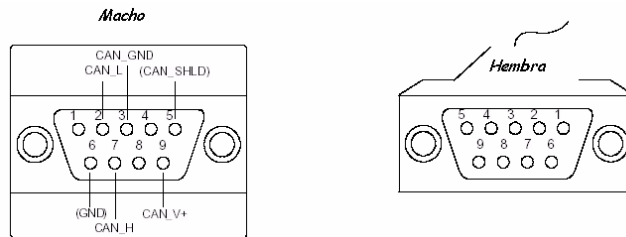
El cable de CanOpen debe tener un mínimo de dos pares (para señal); cada uno de los pares debe tener blindaje por separado para reducir las posibles perturbaciones entre los destinados a señal y los destinados a alimentación. Cada par ha de tener diferente calibre:

- Para el par correspondiente a CAN_H y CAN_L: AWG24² (0.205mm²)
- Para el par correspondiente a CAN_GND y CAN_V+: AWG22² (0.324mm²)

b. Los conectores:

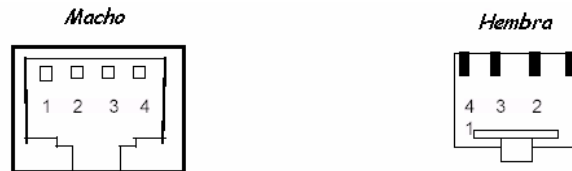
Los conectores deben cumplir con el siguiente pinout:

Para conectores SubD-9:



1:		Reservado
2:	CAN_L	Nivel bus dominante bajo
3:	CAN_GND	Línea de Tierra
4:		Reservado
5:	CAN_SHLD	Opcional malla CAN
6:	GND	Tierra opcional
7:	CAN_H	Nivel bus dominante alto
8:		Reservado
9:	CAN_V+	Opcional CAN alimentación externa positiva

Para conectores RJ10:



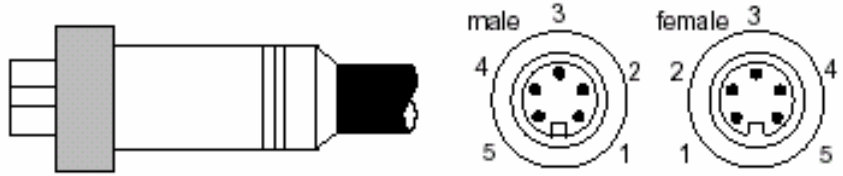
1:	CAN_V+	Línea opcional de suministro de voltaje
2:	CAN_L	Línea de bus CAN_L
3:	CAN_H	Línea de bus CAN_H
4:	CAN_GND	Línea de Tierra

Para conectores RJ45:



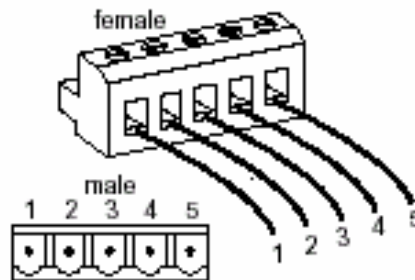
1:	CAN_H	CAN_H nivel bus (dominante alto)
2:	CAN_L	CAN_L nivel bus (dominante bajo)
3:	CAN_GND	Tierra / 0 V/V-
4:		Reservado
5:		Reservado
6:	CAN_SHLD	Opcional malla CAN
7:	CAN_GND	Tierra / 0 V/V-
8:	CAN_V+	Opcional CAN alimentación externa positiva

Para conectores mini 5 pines:



- | | | |
|----|----------|--|
| 1: | CAN_SHLD | Opcional malla CAN |
| 2: | CAN_V+ | Opcional CAN alimentación externa positiva |
| 3: | CAN_GND | Tierra / 0 V/V- |
| 4: | CAN_H | CAN_H nivel bus (dominante alto) |
| 5: | CAN_L | CAN_L nivel bus (dominante bajo) |

Para conectores abiertos:



- | | | |
|----|----------|--|
| 1: | CAN_GND | Tierra / 0 V/V- |
| 2: | CAN_L | CAN_L nivel bus (dominante bajo) |
| 3: | CAN_SHLD | Opcional malla CAN |
| 4: | CAN_H | CAN_H nivel bus (dominante alto) |
| 5: | CAN_V+ | Opcional CAN alimentación externa positiva |

c. Resto de hardware:

El resto del hardware implicado en el funcionamiento del bus es el propio de cada uno de los dispositivos conectados. Para el diagnóstico de estos elementos y la comprobación de su correcto funcionamiento, consultar los manuales específicos de hardware; básicamente, los puntos a tener en cuenta son:

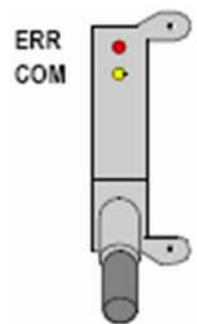
- Descripción del hardware afectado (todos los módulos), incluyendo:
 - a. referencias
 - b. número de serie (SN)

- c. versión del software (SV)
- d. versión del producto (PV)

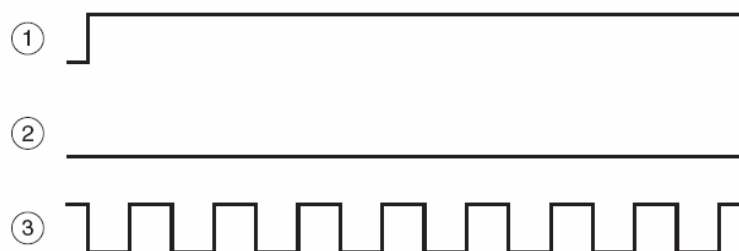
- Status de los leds de los módulos que fallan en el momento de producirse el defecto:

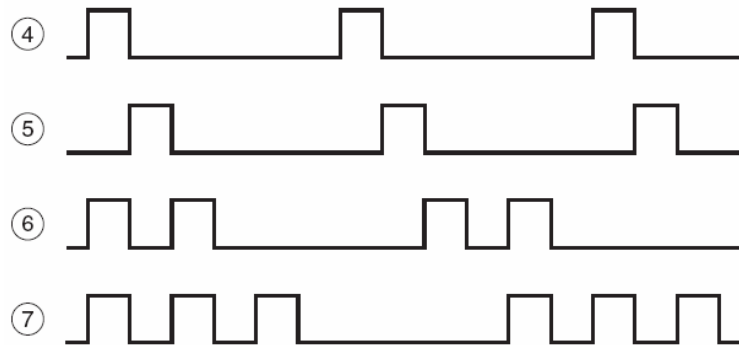
Tarjeta TSXCPP110: tarjeta PCMCIA de comunicación CANOpen para CPU de TSX Modicon Premium

ERR	COM	Significado
A	A	Tarjeta sin alimentación o transferencia de configuración activa
A	P	Tarjeta sin configurar (irregular) Tarjeta configurada y lista, bus inactivo o no hay firmware CANopen (regular)
A	S	Bus configurado y activo, no error
S	A	Error detectado, controlador bus detenido
S	P	Error en tarjeta, error de configuración, o error en sincronización entre la tarjeta y el PLC
S	S	Bus configurado y activo, no se puede conectar con al menos un abonado de bus o emite un error
S = Siempre encendido, P = Parpadeando, A = Apagado		



Variadores de velocidad (ATV31, ATV61,...), control de movimiento (Lexium05,...), E/S distribuidas (STB, OTB,...):





LED "Bus de campo RUN":

- (1) El equipo está OPERATIVO en estado NMT
- (3) El equipo está PRE-OPERATIVO en estado NMT
- (5) El equipo está PARADO en estado NMT

LED "Bus de campo ERR":

- (1) CAN es BUS-OFF, p. ej. después de 32 intentos fallidos de envío.
- (2) El equipo está en servicio
- (4) Límite de aviso alcanzado, p. ej. después de 16 intentos fallidos de envío
- (6) Se ha producido un evento de control (Node-Guarding)
- (7) El mensaje SYNC no se recibió en el periodo de tiempo configurado

Puerto CANOpen integrado en la CPU del Modicon M340:

LED	Encendido permanente	Parpadeando	Apagado
CAN RUN (verde)	Red CANopen operativa	Intermitencia rápida: AutoBaud/LSS en curso Intermitencia lenta: Red CANopen preoperativa 1 parpadeo: Red CANopen detenida 3 parpadeos: Descarga del firmware CANopen	—
CAN ERR (rojo)	Bus CANopen detenido	Intermitencia rápida: AutoBaud/LSS en curso Intermitencia lenta: Configuración de CANopen no válida 1 parpadeo: advertencia limite alcanzado 2 parpadeos: evento control errores 3 parpadeos: error de sincronización	No hay error de CANopen

- Esquema del cableado de los módulos que fallan así como referencias de los cables utilizados.
- Descripción de terceros equipos posiblemente relacionados con el defecto (características, referencias, cableado, etc).

3.4. Configuración de los dispositivos

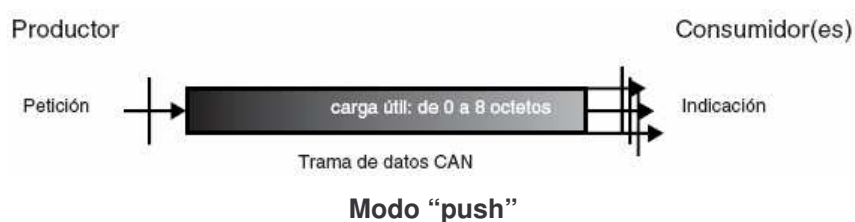
Esencialmente, los puntos a revisar en éste apartado son:

- Verificar que todos los dispositivos (tanto el maestro como los distintos esclavos) están configurados a la misma velocidad.
- Verificar que cada dispositivo tiene su propia dirección CANopen, que debe ser única (por defecto, entre 0 y 126 para cada uno de los esclavos y 127 para el maestro del bus).
- Verificar la correcta configuración de todos los dispositivos (tanto el maestro como los distintos esclavos) con las diferentes aplicaciones relacionadas (Unity, PL7, SyCon, Power Suite, etc -para todo esto, consultar la documentación de los diferentes softwares relacionados-).
- Según la configuración del bus y de cada uno de los dispositivos conectados al mismo, verificar el intercambio correcto de información entre ellos. Para ésta prueba es necesario contar con un analizador de buses Can (en nuestro caso, el dispositivo puede ser un PEAK -ver el anexo correspondiente a la configuración y a su funcionamiento-) que nos permita ver y analizar las tramas de intercambio entre todos los elementos del bus.

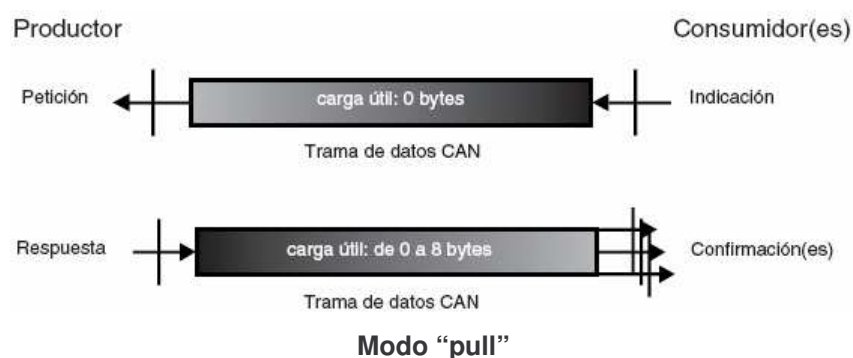
4. Funcionamiento de CANOpen

4.1. Principios de funcionamiento

En CAN se ha implementado una **arquitectura de emisión**. El emisor (editor) emite el mensaje junto a un identificador en el bus. Los receptores supervisan todo el tráfico en el bus. Si el identificador coincide con los criterios de filtro de la suscripción, el cliente lee y procesa el mensaje completo. De este modo, el receptor se convierte en suscriptor. Este es el **modo push** del modelo de editor y suscriptor.

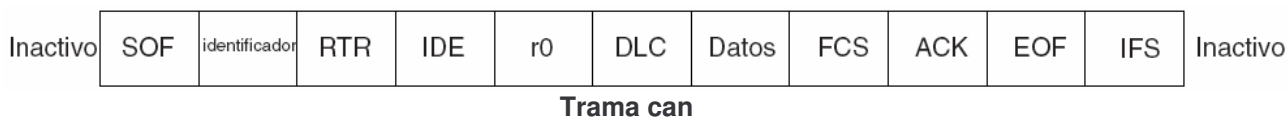


CAN también admite un **modo pull** del modelo de editor y suscriptor. Un consumidor puede activar la transmisión de un mensaje con una petición de transmisión remota. La petición de transmisión remota o RTR (remote transmission request) es una trama de CAN con la señal RTR. Cuando el productor recibe una petición de este tipo, transmite el mensaje asociado.



4.2. Tramas y protocolos CAN

Una trama CAN comienza con un bit de inicio de trama o SOF (start of frame). Le siguen once bits de identificador, desde el bit más significativo al menos significativo. El siguiente bit es el bit de petición de transmisión remota o RTR (remote transmission request) seguido de cinco bits de control y de una carga de hasta ocho bytes. Los bits de control son el identificador ampliado o IDE (extended ID), un bit de reserva y tres bits que codifican la longitud de la carga en la parte de los datos (DLC) en bytes. Una secuencia de comprobación de trama o FCS (frame check sequence) sigue a la carga útil de hasta ocho bytes. El emisor transmite un bit ACK recesivo, que sobrescribirán con un bit dominante los receptores que han recibido la trama sin errores en este momento. El bit de fin de trama o EOF (end of frame) señala el fin del mensaje. El bus debe permanecer en estado recesivo para las longitudes de bits de espacio de trama de intermisión o IFS (intermission frame space) antes de que comience la siguiente trama.



La trama de mensajes de CAN descrita anteriormente es una trama de mensajes básica.

Para las aplicaciones que necesitan un conjunto mayor de identificadores, se ha definido la trama de mensajes CAN ampliada.

La trama ampliada dispone de 18 bits de identificación adicionales en el encabezado, tras los bits de control.

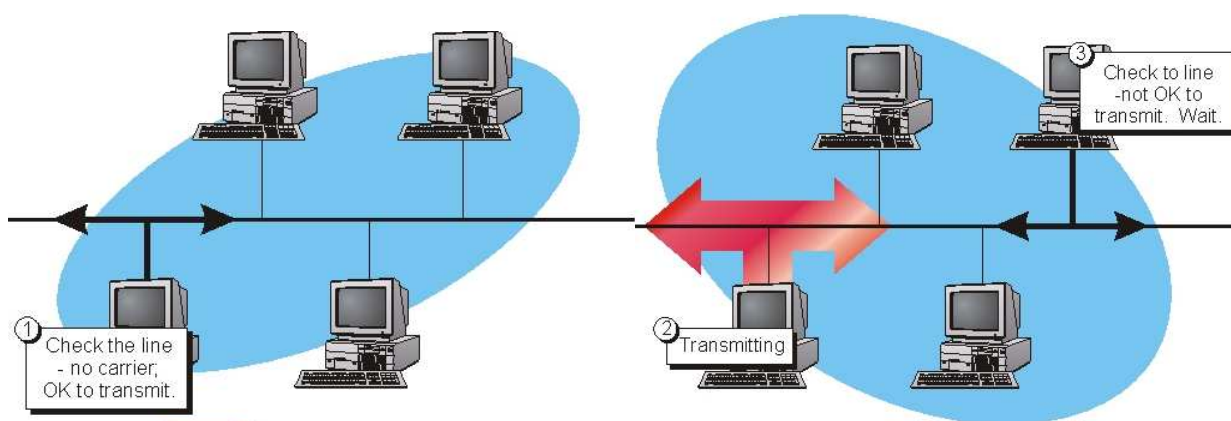
Esto amplía la gama de 2^{11} a 2^{29} identificadores diferentes. Ambos tipos de tramas pueden coexistir en el bus.

Existen varios protocolos CAN:

- **CAN Protocol Specification 2.0 A:** CAN Controller trabaja únicamente con identificadores de mensaje de 11-bit.(CAN estándar)
- **CAN Protocol Specification 2.0 B pasivo:** CAN Controller transmite sólo tramas con identificadores de 11-bit, pero chequea la recepción de tramas estándar como de tramas extendidas con identificadores de 29-bit (se envía incluso la confirmación).(CAN extendido)
- **CAN Protocol Specification 2.0 B activo:** CAN Controller puede recibir tramas estándar y extendidas.

4.3. CSMA/CD+AMP

CAN es un sistema de bus de acceso múltiple de detección de transportador, con detección de colisión y arbitraje para establecer prioridades con los mensajes (CSMA/CD+AMP). Puesto que la colisión en sí misma no se produce en CAN, a menudo se describe como acceso múltiple de detección de transportador con elusión de colisiones o CMSA/CA (carrier sense multiple access with collision avoidance).



Método CSMA/CD

4.4. Arbitraje del bus

Un problema que normalmente se produce con una arquitectura de emisión es que **diferentes nodos de la red pueden realizar envíos al mismo tiempo**.

CAN resuelve este problema con dos mecanismos:

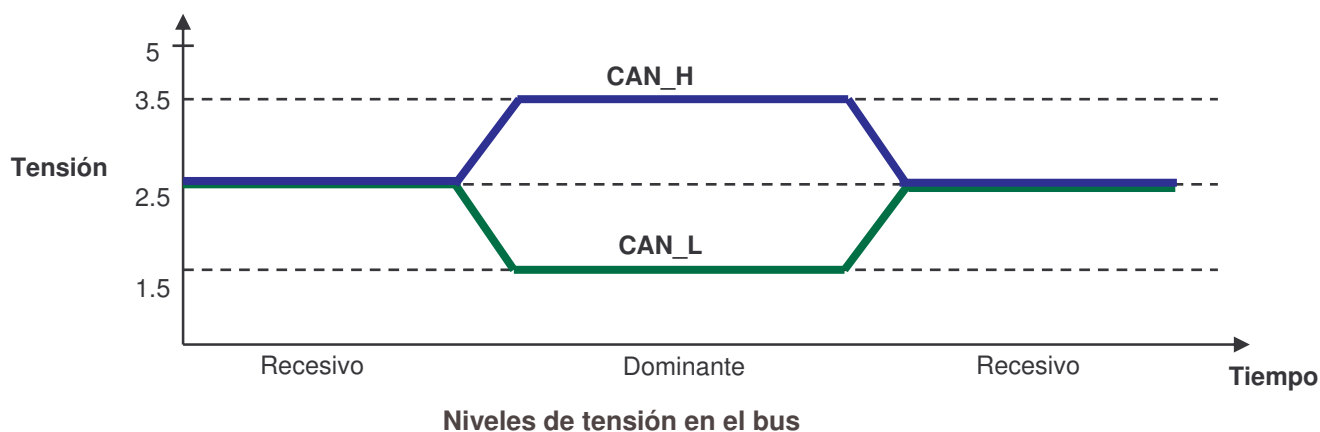
a. Estado recesivo y estado dominante:

Un emisor supervisa el medio para **comparar si otro nodo también está realizando un envío**. Si el medio está libre, el nodo comienza el envío.

La codificación de bits del medio posee un valor recesivo o dominante.

Estos estados se calculan con la diferencia de tensión entre CAN_L y CAN_H:

- Estado recesivo: $V_{CAN_H} - V_{CAN_L} = 0V$ (-0.5 V a + 50 mV)
- Estado dominante: $V_{CAN_H} - V_{CAN_L} = 2V$ (1.5 V a 3.5 V)



Si dos nodos realizan un envío al mismo tiempo, un receptor sólo verá el valor dominante. En la codificación binaria, el valor "0" es el dominante y el valor "1" es el recesivo.

Nodo A	Nodo B	Nodo C	Bus
D	D	D	D
D	D	R	D
D	R	D	D
D	R	R	D
R	D	D	D
R	D	R	D
R	R	D	D
R	R	R	R

Arbitraje del bus

b. Identificador del mensaje

No obstante, puede que varios nodos comiencen a realizar el envío al mismo tiempo. CAN resuelve este problema con un esquema de prioridades.

Cuando un nodo realiza un envío, siempre se encuentra a la escucha en el bus. Si está enviando un valor recesivo y está recibiendo un bit dominante, detiene el envío y continúa escuchando únicamente.

Este sencillo mecanismo evita las colisiones en el bus CAN. El mensaje con el identificador más bajo gana en el sistema de arbitraje del bus.

4.5. Mecanismos de detección de errores

CAN cuenta con 5 mecanismos diferentes para detectar mensajes erróneos:

- La secuencia de comprobación de tramas o FCS (frame check sequence) contiene la comprobación de redundancia cíclica o CRC (cyclic redundancy check) de la trama. El receptor además computa la CRC de la trama y compara sus propios resultados con la FCS. Si no son iguales, la trama presenta un error de CRC.

- El receptor detecta errores en la estructura de la trama. Si la trama presenta una estructura defectuosa, la trama tiene un error de formato.
- El receptor de una trama emite un bit dominante de acuse de recibo o ACK (ACK) si ha recibido una trama sin errores. Si el emisor no recibe un bit dominante de ACK, genera un error de acuse de recibo.
- El emisor también supervisa los bits en el bus. Si se envía un bit y el homólogo que recibe difiere, el emisor reconoce un error.
- CAN emplea una codificación sin retorno a cero o NZR (no return to zero) con relleno de bits. Si el emisor debe transmitir cinco bits iguales seguidos, introduce un bit invertido adicional. Con este relleno de bits, el receptor puede realizar la sincronización en el flujo de bits. El receptor elimina los bits rellenos de la trama transmitida. Si más de cinco bits cuentan con el mismo valor, el receptor reconoce un error de relleno de bits.

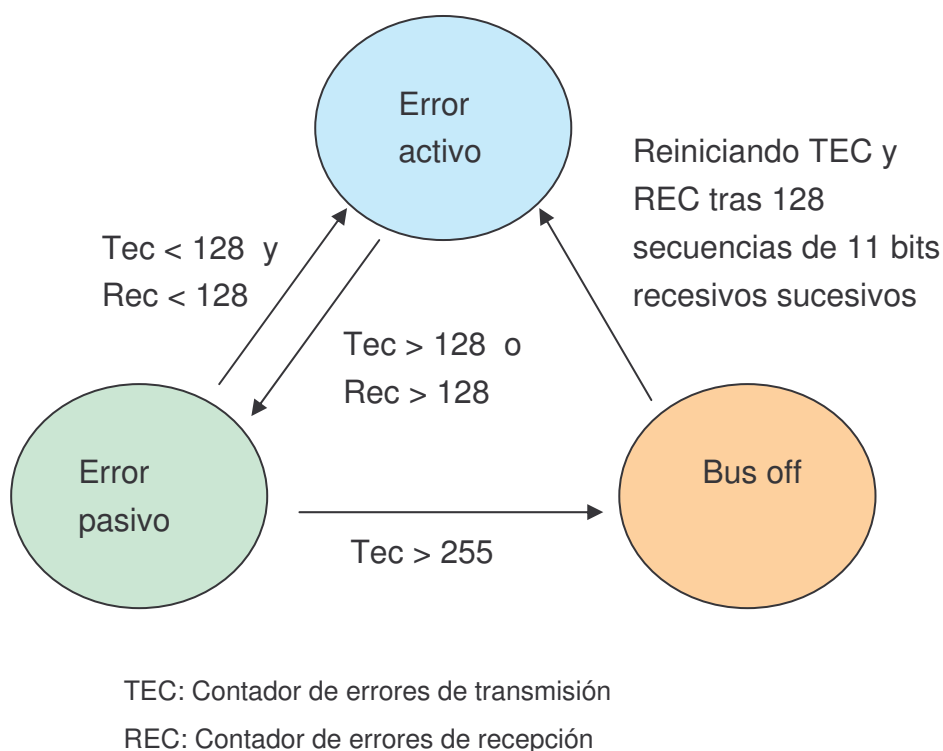
4.6. Estados de error

La trama de error consiste en dos campos: Indicador de error y Delimitador de error. El delimitador de error consta de 8 bits recesivos consecutivos y permite a los nodos reiniciar la comunicación limpiamente tras el error. El indicador de error es distinto según el estado de error del nodo que detecta el error.

Si se produce una perturbación local en un nodo del bus, la lógica de errores hará que este equipo envíe tramas de error, que provocarán errores en otros nodos. Para evitar este efecto, se han incorporado en el CAN medidas de aislamiento de nodos defectuosos.

Un nodo puede encontrarse en uno de los tres estados siguientes:

- **Error Activo:** Estado normal del nodo. Participa en la comunicación normalmente y en caso de detección de error envía una trama de error activa.
- **Error Pasivo:** Un nodo en estado de error pasivo participa de la comunicación, sin embargo tiene que recibir una secuencia adicional de bits recesivos antes de intentar transmitir.
- **Bus-OFF (aislado):** En este estado el equipo se desconecta del bus y es necesario reiniciar el equipo para que vuelva a comunicar.



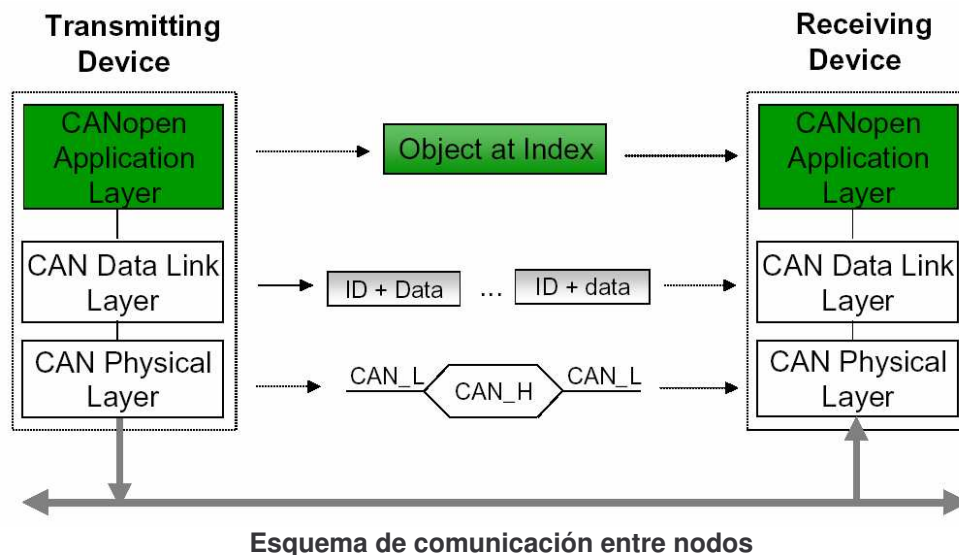
4.7. Protocolo CANOpen

El protocolo está estandarizado por CiA (www.can-cia.com) CAN en la Automatización. Es un protocolo de alto nivel de CAN. Define servicios de comunicación orientados a soluciones industriales de automatización

Cada dispositivo de un cierto tipo, sea del fabricante que sea, comunica sus funciones básicas mediante el mismo perfil. La información de cada nodo se estructura en un Diccionario de objetos.

Se trata de de comunicaciones de tipo:

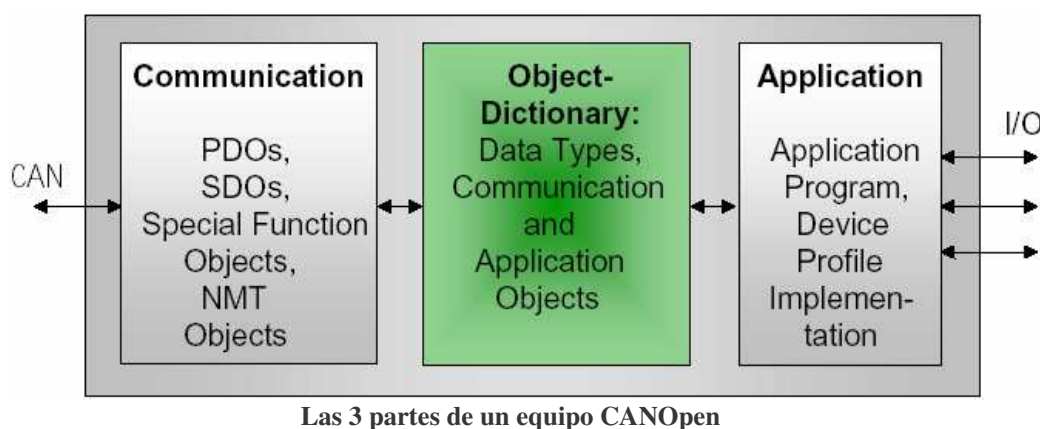
- Productor/Consumidor para datos configurados (PDO = Comunicaciones Implícitas)
- Maestro/Esclavo para datos programados (SDO = comunicaciones explícitas)
- Maestro / Esclavo para gestión de red (NMT)



Un equipo CANOpen se puede dividir en tres partes:

- Communication interface: provee los servicios de envío y recepción de objetos de comunicación por el bus
- Object dictionary: Define todos los tipos de datos, objetos de comunicación y objetos de aplicación utilizados por el equipo. Esta información se encuentra en el archivo EDS.

- Application: Contiene la funcionalidad de control del equipo y el interfase con el hardware del equipo



4.8. Diccionario de objetos – Perfiles – Fichero EDS

El diccionario de Objetos contiene todos los parámetros del equipo que son accesibles mediante comunicaciones por medio del bus. Hay una serie de parámetros comunes a todo equipo CANopen y que vienen definidos según el perfil del equipo: DS301, DS4xx. Todo equipo con perfil DS4xx contiene los parámetros del perfil DS301.

En la capa aplicación se definen los siguientes grupos de perfiles:

Profile number	Device class
DS-401	Generic I/O modules
DS-402	Drives and motion control
DS-403	Not allocated
DS-404	Measuring devices and closed loop controllers
DS-405	IEC 61131-3 programmable devices
DS-406	Encoders
DS-407	Public transportation - Passenger Information Systems
DS-408	Fluid Power Technology - Hydraulic drives and proportional valves
DS-409	Vehicle door control
DS-410	Declinometers
DS-412	Medical Devices
DS-413	Truck Gateways
DS-414	Weaving Machines
DS-415	Road Construction Machinery
DS-416	Building Door Control
DS-417	Lift Control Systems
DS-418	Battery Modules
DS-419	Battery Chargers
DS-420	Extruder Downstream Devices

Clases de dispositivos y su perfil asociado

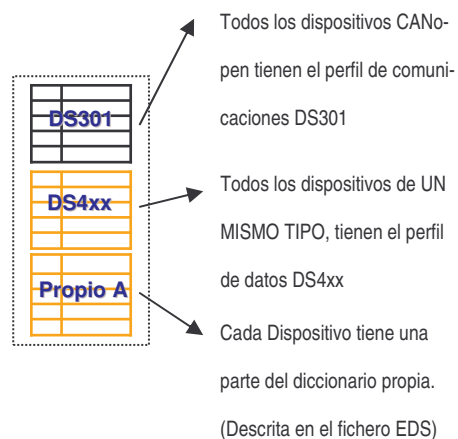
Perfiles estándar CANopen:

- CANopen communication profile (DSP-301): (perfil básico): Define una serie de parámetros necesarios para comunicar con un equipo CANopen. Todos los equipos CANopen requieren tener implementado el DSP-301. Índices asignados dentro del OD en el rango 1000h-1FFFh
- CANopen device profiles (DSP-4xx): Define la funcionalidad de un tipo de equipo en particular y como acceder a esa funcionalidad por medio del bus CAN. Los perfiles disponibles hasta ahora son: DSP 401 (módulos E/S), DSP 402 (Control de drives y Motion), DSP 403 (HMI), DSP 405 (Equipos programables IEC1131) y DSP 406 (Encoders). Índices asignados dentro del OD en el rango 6000h-9FFFh
- CANopen Application profiles: Define la funcionalidad y las relaciones de comunicación de todos los equipos CANopen del bus. Perfiles de aplicación disponibles: DS414 (maquinaria textil), DS417 (Ascensores) y DS422 (Sistemas municipales).

La parte de configuración de las comunicaciones sigue la norma DS 301

La parte de datos sigue la norma DS 4xx (xx dependiendo del tipo de dispositivo)

Existe una tercera parte propia del dispositivo (Definida en el fichero EDS)



Hay una serie de parámetros que son de libre definición por parte del fabricante.

La descripción del OD se realiza en forma de un archivo EDS (Electronic Data Sheet) en formato ASCII. Este archivo sigue una

sintaxis estricta y se puede utilizar en todos los configuradores de bus CANopen (Sycon etc.).

En los EDS se especifica el Diccionario de Objetos completo para un determinado dispositivo:

```

A58F_E.eds - Bloc de notas
Archivo Edición Formato Ver Ayuda
[FileInfo]
FileName=A58F_E.eds
FileVersion=1
FileRevision=2
Description=ATV58F Option Card
CreationTime=00:00AM
CreationDate=12-05-2000
CreatedBy=Marie-Annick Menanteau, Schneider Electric
; révision 2 du 26/07/2000 modification des valeurs par défaut des paramètres 100C et 100D
;
*****

[DeviceInfo]
VendorName=Telemecanique
ProductName=ATV58F_E
ProductVersion=1
ProductRevision=1
BaudRate_10=0
BaudRate_20=0
BaudRate_50=0
BaudRate_100=0
BaudRate_125=1
BaudRate_250=1
BaudRate_500=1
BaudRate_800=0
BaudRate_1000=1
Granularity=0x8
VendorNumber=0x0200005a
ProductNumber=0
SimpleBootUpMaster=0
ExtendedBootUpMaster=0
SimpleBootUpSlave=1
ExtendedBootUpSlave=0

```

Fichero EDS abierto con el bloc de notas

Para que el Master pueda conocer el Diccionario de Objetos de cada nodo, necesita poder acceder al EDS localmente:

- Un master que corre en un controlador “embedded” tendrá la información guardada en su memoria
- Una herramienta de configuración accederá directamente al fichero EDS

Cada objeto del diccionario se puede acceder mediante:

- Un índice de 16-bit
- Un subíndice de 8-bit.

Por ejemplo, un registro del diccionario puede tener un índice y un subíndice (ej: 6000:01).

Existen en la catalogo técnicos de los dispositivos tablas indicando el índice, subíndice y si el objeto de lectura/escritura.

Por ejemplo, para leer y modificar la rampa de aceleración y deceleración se ha de acceder a los índices/subíndices: 203C/2 y 203C/3.

Modbus address	CANopen address	Code	Read/Write	Name/Description/Possible values
64001	2262 / 2	rOt	R/WS	Direction of operation authorized Factory setting: 0 Direction of operation authorized for the RUN key on the keypad (ATV31...A only) or the RUN key on the remote display terminal. 0 = "dFr": Forward 1 = "drS": Reverse 2 = "bOt": Both directions are authorized (except for the keypad on the ATV31...A: Forward only).
9004	203C / 5	rPt	R/WS	Type of ramp Defines the shape of the acceleration and deceleration ramps. Factory setting: 0 0 = "Lin": Linear 1 = "S": S ramp 2 = "U": U ramp 3 = "CUS": Customized
9005	203C / 6	tA1	R/W	Start of CUS-type acceleration ramp rounded Unit: 1% (as % of total ramp time ACC or AC2) Factory setting: 10 Adjustment range: 0 to 100
9006	203C / 7	tA2	R/W	End of CUS-type acceleration ramp rounded Unit: 1% (as % of total ramp time ACC or AC2) Factory setting: 10 Adjustment range: 0 to 100 - tA1
9007	203C / 8	tA3	R/W	Start of CUS-type deceleration ramp rounded Unit: 1% (as % of total ramp time dEC or dE2) Factory setting: 10 Adjustment range: 0 to 100
9008	203C / 9	tA4	R/W	End of CUS-type deceleration ramp rounded Unit: 1% (as % of total ramp time dEC or dE2) Factory setting: 10 Adjustment range: 0 to 100 - tA3
9001	203C / 2	ACC	R/W	Acceleration ramp time Unit: 0.1 s Factory setting: 30 Adjustment range: 1 to 9999 Defined for accelerating between 0 and the nominal frequency FrS (page 25).
9002	203C / 3	dEC	R/W	Deceleration ramp time Unit: 0.1 s Factory setting: 30 Adjustment range: 1 to 9999 Defined for decelerating between the nominal frequency FrS (page 25) and 0. Check that the value of dEC is not too low in relation to the load to be stopped.

Tabla de objetos del variador ATV31

4.9. COB-ID – Arbitraje del bus

El COB-ID, en realidad será el Identificador del mensaje CAN en el bus CAN

El COB-ID es un número que es una combinación entre el número de nodo CANopen y el servicio de comunicaciones que se desea (4+7 bits)

El COB-ID serían los canales por los que los dispositivos envían y reciben información

Por ejemplo:

- Un PDO de un dispositivo tiene un COB-ID único
- Un dispositivo tiene dos COB-ID para los SDO (comunicaciones explícitas), uno para recibir información y otro para enviarla

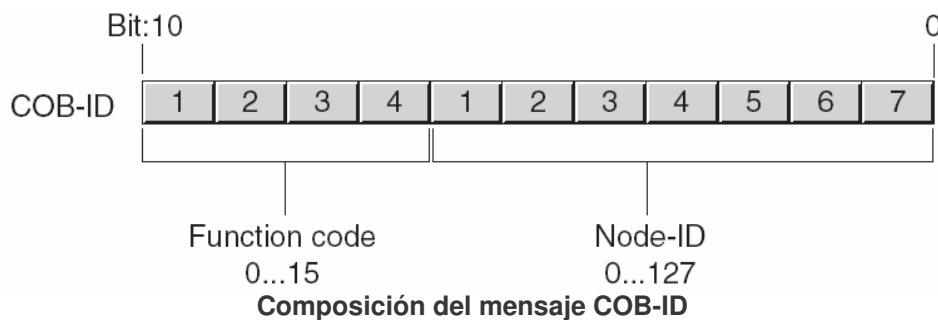
COB-ID

El identificador COB-ID (Communication Object Identifier) tiene dos funciones principales:

- Arbitraje del bus especificación de las prioridades de transmisión. El objeto de comunicación con el COB_ID más bajo es el de mayor prioridad en la red
- Identificación de los objetos de comunicación, el equipo de red reconoce el tipo de servicio que se demanda (NMT, SYNC, TxPDO, RxPDO , SDO, EMCY)

Se compone de 2 partes:

- Código de función, 4 bits
- Node-ID, 7 bits



Recordatorio: Cualquier nodo puede enviar un mensaje CAN al detectar que el bus está libre. Durante el arbitraje, cada nodo monitoriza el bus para detectar si su bit transmitido se sobrescribe por un mensaje de mayor prioridad (Nivel recesivo del bit = 1, Nivel dominante del bit = 0). Tan pronto como un nodo de transmisión detecte un bit dominante mientras transmite uno recesivo deja el bus, interrumpe la transmisión inmediatamente y comienza a recibir la trama

Definición de los identificadores:

Objetos broadcast generales			
Objeto	COB-ID Bin	COB-ID Hex	COB-ID Dec
NMT	0000	0x000	0
SYNC	0001	0x080	128
TIME STAMP	0010	0x100	256

Objetos broadcast punto a punto			
Objeto	COB-ID Bin	COB-ID Hex	COB-ID Dec
Emergencia	0001	0x081 to 0x0FF	129 to 255
PDO 1 transmisión	0011	0x181 to 0x1FF	385 to 511
PDO 1 recepción	0100	0x201 to 0x27F	513 to 639
PDO 2 transmisión	0101	0x281 to 0x2FF	641 to 767
PDO 2 recepción	0110	0x301 to 0x37F	769 to 895
PDO 3 transmisión	0111	0x381 to 0x3FF	897 to 1023
PDO 3 recepción	1000	0x401 to 0x47F	1025 to 1151
PDO 4 transmisión	1001	0x481 to 0x4FF	1153 to 1279
PDO 4 recepción	1010	0x501 to 0x57F	1281 to 1407
Servidor SDO	1011	0x581 to 0x5FF	1409 to 1535
Cliente SDO	1100	0x601 to 0x67F	1537 to 1663
NODE GUARD	1110	0x701 to 0x77F	1793 to 1919

4.10. Objetos de comunicación

CANopen define una capa de aplicación y un perfil de comunicación basado en CAN. CANopen define los siguientes objetos de comunicación (mensajes):

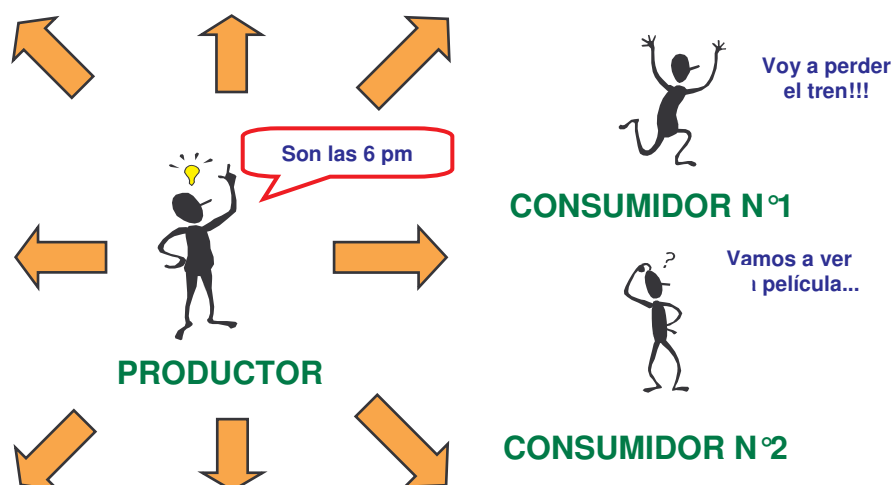
- Objeto de datos de proceso o PDO (Process data object).
- Objeto de datos de servicio o SDO (Service data object).
- Objeto de gestión de red o NMT (Network management object).
- Objeto de funciones especiales (SYNC, EMCY, TIME).

a. Objeto de datos de proceso o PDO (Process data object)

Los objetos de datos de proceso u objetos de comunicación implícita se utilizan para leer/escribir rápidamente datos de proceso, para aplicaciones de tiempo real.

La transmisión de un PDO utiliza el modelo de productor y consumidor de CAN ampliado mediante transferencias síncronas:

- El PRODUCTOR es una entidad individual que produce información.
- El CONSUMIDOR es una entidad que usa la información para hacer algo (diferentes entidades pueden usar la misma información)



Un PDO puede transportar una carga útil de 8 bytes, que es la carga máxima de una trama CAN. Por defecto, cada nodo tiene acceso a sólo 4 PDOs (es posible tener más con islas Advantys STB).

Ejemplo nº1:

Configured PDOs								
PDO name	Symbolic Name	COB-ID	I Type	I Addr.	I Len.	O Type	O Addr.	O Len.
Receive PDO1	PDO_1400	513				QB	0	6
Transmit PDO1	PDO_1800	385	IB	0	4			

PDOs configurados: 6 bytes de recepción / 4 bytes de transmisión

Ejemplo nº2:

Configured PDOs								
PDO name	Symbolic Name	COB-ID	I Type	I Addr.	I Len.	O Type	O Addr.	O Len.
Receive PDO1	PDO_1400	513				QB	0	6
Receive PDO6	PDO_1405	1664				QB	0	8
Transmit PDO1	PDO_1800	385	IB	0	4			
Transmit PDO6	PDO_1805	1665	IB	0	8			

PDOs configurados: 14 bytes de recepción / 12 bytes de transmisión

Una de las ventajas que ofrece CANopen respecto a otros buses es la posibilidad de configurar el contenido de los parámetros que se envían/reciben en un PDO. A esto se le denomina “mapeado dinámico”. Los objetos mapeables permitidos por el equipo se encuentran contenidos en el archivo EDS que se proporciona con cada equipo de CANopen.

Se seleccionan los registros que se desean leer/escribir desde el diccionario de objetos y se añaden a los PDOs de emisión y PDOs de recepción.

PDO Contents Mapping Object Index 1600			
Mappable Objects from EDS file:			
Obj.Idx.	Sub.Idx.	Parameter	Access
6041		Drivecom status register	Read
6040		Drivecom command reg.	Write
6042		Target velocity	Write
6044		Control effort	Read
6043		Speed demand	Read
603F		Error code	Read
2000	3	Reserved	Read / Write
Mapped Object dictionary			
Obj.Idx.	Sub.Idx.	Parameter	Symbolic name
6040	0	Drivecom command reg.	Object6040Idx0
6042	0	Target velocity	Object6042Idx0
2001	5	High Speed	Object2001Idx5

Mapeado dinámico desde SyCon

ATENCIÓN: No todos los PDOs de un equipo se pueden mapear. En ciertos PDOs su contenido es fijo y no puede ser cambiado por el usuario. A esto se le denomina “mapeado fijo”. Por ello es importante leer la documentación del equipo y ver cual de los PDOs del equipo ofrecen la posibilidad de ser mapeados.

En la herramienta de configuración y programación del PLC, estos datos serán accesibles en las palabras de memoria %MW y habrá que definir unas direcciones de inicio de las lecturas y escrituras y el número de palabras a leer y escribir.

Entradas		Salidas	
Cantidad de palabras (%MW)	32	<input type="radio"/> Mantener <input checked="" type="radio"/> RESETEAR	
Índice de la primera %MW	0	Cantidad de palabras (%MW)	32
		Índice de la primera %MW	32

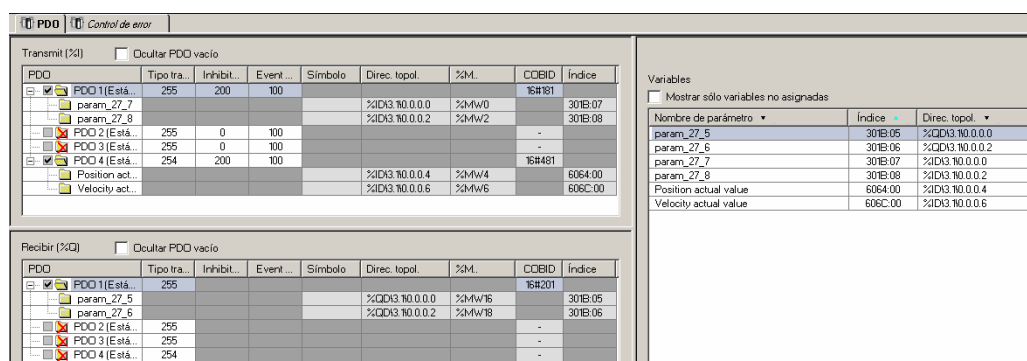
Direccionamiento %MW

Con Modicon M340:

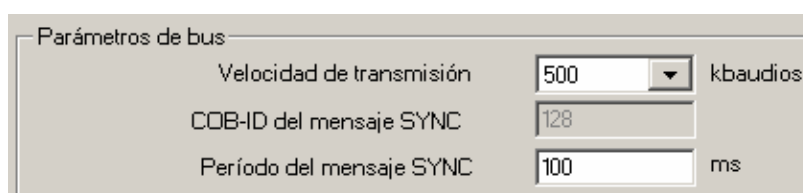
Si se utiliza Modicon M340, la configuración de los nodos y PDOs se realiza directamente mediante el interfaz gráfico arrastrando los objetos a leer y escribir en los PDO de emisión y recepción. Para ello se a de seleccionar una CPU con puerto CANOpen integrado y añadir los dispositivos desde una librería de productos (ampliable mediante una aplicación externa).



Interfaz gráfico de bus CANOpen (solo M340)



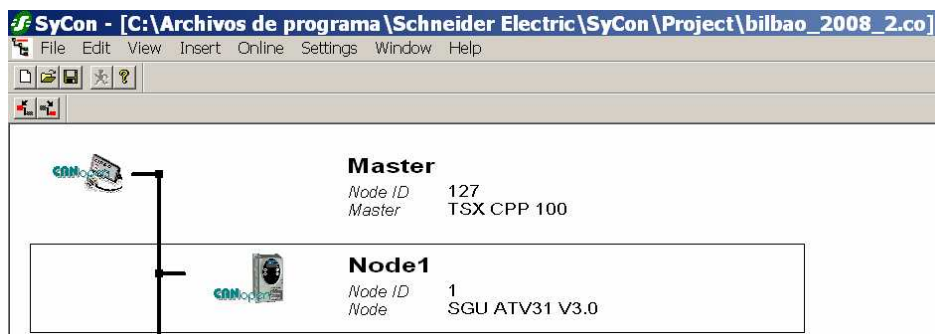
Interfaz gráfico para añadir objetos a los PDOs



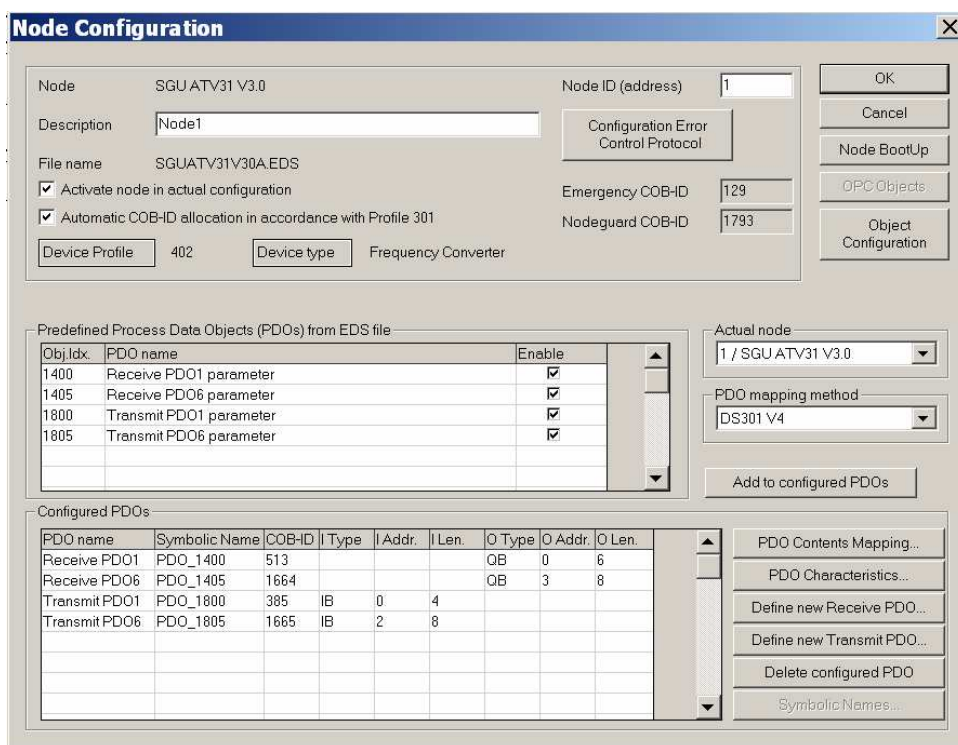
Configuración de la velocidad del bus

Modicon Modicon Premium:

Si se utiliza Modicon Premium, la configuración de los nodos y PDOs se realiza mediante SyCon, software de interfaz gráfico que permite seleccionar los objetos a leer y escribir en los PDO de emisión y recepción. Luego se ha de guardar la configuración (archivo .co) e seleccionar desde Unity Pro.



Software Sycon



Configuración de los PDOs y sus objetos de un nodo



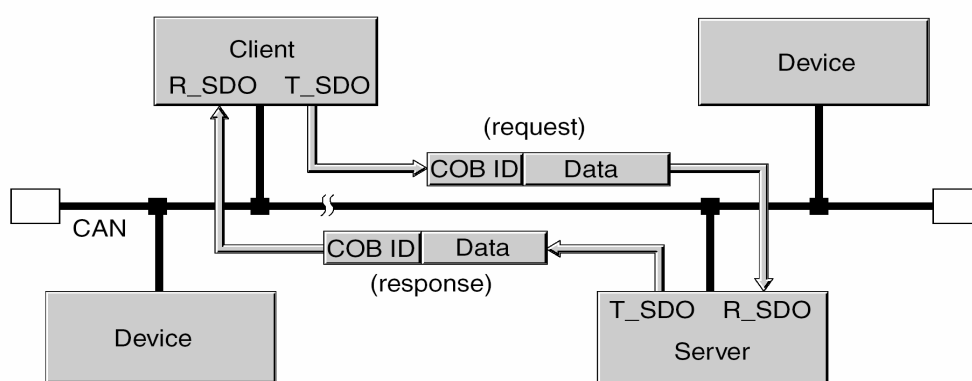
Configuración de la velocidad del bus

Selección del archivo .co y configuración de los %MW

b. Objeto de datos de servicio o SDO (Service data object)

Los objetos de datos de servicio u objetos de comunicación explícita tienen con fin la transmisión de parámetros y permiten petición de lectura y petición de escritura.

La comunicación de SDO emplea una comunicación de homólogo a homólogo, con uno de los homólogos que actúa como servidor y el otro como cliente. Se acusa el recibo de cada SDO.



Método Maestro/Esclavo

Los SDO ofrecen acceso al diccionario de objetos de los dispositivos remotos.

Un SDO no tiene límites de longitud. Si la carga útil no cabe en la trama CAN, se dividirá en varias tramas CAN.

Es posible acceder a los datos de los nodos mediante SDOs en cualquier momento y sin haber configurado nada previamente mediante:

- una sección de programa con funciones de lectura y escritura (por ejemplo: READ_VAR y WRITE_VAR),

```
(* read the node 5 SDO, index 1018, subindex 3 *)
if (%M400) then
    subindex_index := 16#00031018 ;
    %MW1052 := 50; (* timeout 5 secondes *)
    READ_VAR(ADDM('0.0.2.5'),'SDO',subindex_index,0,%MW1050:4,%MW1100:2);
    %M400:= 0;
end_if;

(* Write the node 31 SDO, index 203C, subindex 2 *)
if (%M401) then
    subindex_index := 16#0002203C;
    %MW1152 := 50; (* timeout 5 secondes *)
    %MW1153 := 2; (* length 2 bytes *)
    %MW1200 := 16#03E8; (* value of object *)
    WRITE_VAR(ADDM('0.0.2.31'),'SDO',subindex_index,0, %MW1200:1,%MW1150:4);
    %M401:= 0;
end_if;
```

Ejemplo con Modicon M340

- la pantalla de depuración de CANOpen de Unity Pro, que incluye menús para leer y escribir valores.

Lectura desde Unity Pro con Modicon M340

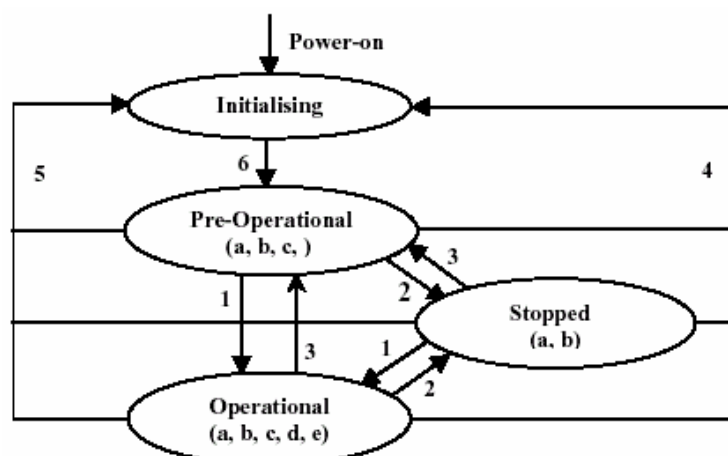
Escritura desde Unity Pro con Modicon Premium

c. Objetos de gestión de red o NMT (Network management object)

Los objetos de gestión de red o NMT (network management objects) cambian el estado o supervisan el estado de un dispositivo CANopen.

Un mensaje de NMT es un mensaje con identificador 0 de CAN. Esto otorga al mensaje NMT la máxima prioridad posible. El mensaje NMT siempre consta de una carga útil de dos bytes en la trama CAN. El segundo byte contiene el ID del nodo al que se dirige. Es posible dirigirse a todos los dispositivos con un mensaje NMT con el ID 0 de nodo reservado.

Un dispositivo CANopen comienza el estado de Inicialización tras encender la alimentación.



Transiciones realizadas por el maestro NMT:

- 1: Start_Remote_Node
- 2: Stop_Remote_Node
- 3: Enter_Pre-Operational_State
- 4: Reset_Node
- 5: Reset_Communication
- 6: Node initialisation stopped

Servicios de com. autorizados:

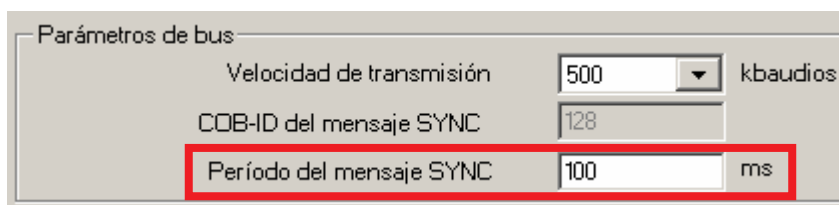
- a. NMT
- b. Node Guard
- c. SDO
- d. EMCY
- e. PDO.

El protocolo de Heartbeat para supervisar el estado del dispositivo se implementa con objetos NMT.

d. Objetos de funciones especiales (SYNC, EMCY, TIME)

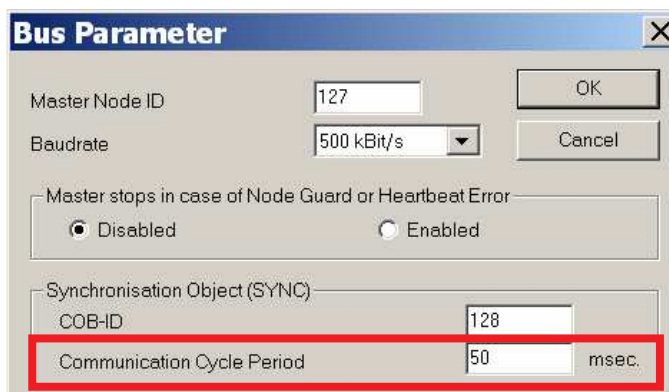
CANopen puede contar con un productor de SYNC para sincronizar las acciones de los nodos de CANopen. Un productor de SYNC emite (periódicamente) el objeto SYNC. El objeto SYNC contiene el identificador CAN 128. Esto puede introducir algo de fluctuación, debido a la prioridad de este mensaje.

Si se trabaja con Modicon M340, la configuración del periodo se hace automáticamente en Unity Pro:



Periodo del mensaje SYNC en Unity Pro Modicon M340

Si se trabaja con Modicon Premium se ha de configurarlo en el software SyCon:



Un error o fallo interno (corriente, tensión, temperatura, comunicación,...) de dispositivo puede activar un objeto de emergencia o EMCY (emergency). La reacción de los consumidores de EMCY depende de la aplicación. El estándar CANopen define varios códigos de emergencia. El objeto EMCY se transmite en una sola trama de CAN con ocho bytes.

Hay una tarea adicional para el objeto EMCY, que es el envío de un mensaje de BOOT-UP. Este mensaje informa a todos los equipos del bus que el equipo que envía el mensaje está listo para trabajar en el bus CAN. Este mensaje se envía con el COB-ID 700h+Node-ID y 1 byte de datos (00h). Este mensaje es de gran utilidad para detectar problemas de cableado o defecto del dispositivo.

Una trama CAN con el ID 256 de CAN y con una carga útil de seis bytes se puede utilizar para transmitir la hora del día a varios nodos de CANopen. Este objeto de marcaje de tiempo (TIME) contiene el valor de la hora en el tipo de datos de hora del día o TOD (Time-of-Day).

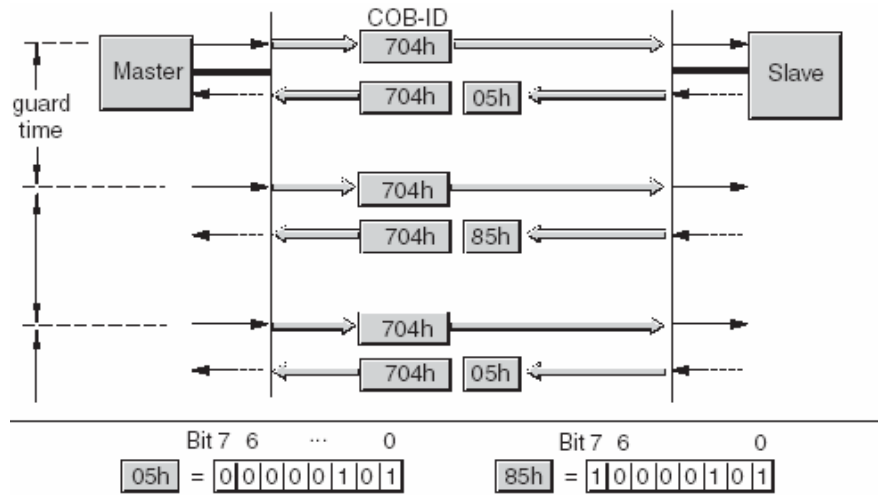
e. Mecanismos de supervisión

CANopen admite dos métodos de supervisión del estado de los dispositivos.

Un administrador de la red puede interrogar a cada dispositivo de forma regular en intervalos de tiempo configurables.

Esto se denomina supervisión o protección de nodos.

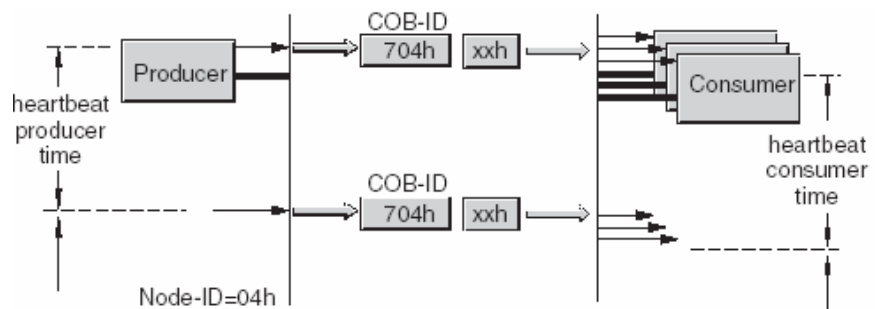
Sin embargo, la protección de nodos o método “node guarding” consume una gran cantidad de ancho de banda.



Node guarding: método maestro/esclavo

Otro mecanismo consiste en que cada dispositivo envíe regularmente un mensaje de Heartbeat.

Con este método se ahorra ancho de banda, en comparación con el protocolo de protección de nodos.



Heartbeat: método productor/consumidor

5. Dispositivos CANOpen

5.1. E/S distribuidas STB

a. introducción

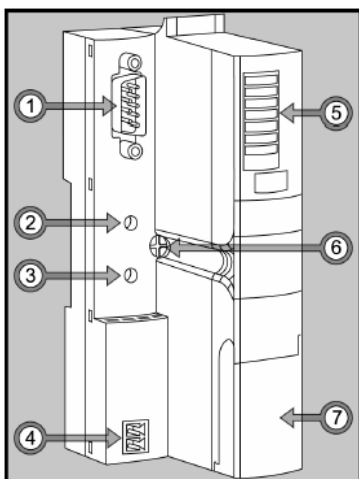


Plataforma de integración y de E/S IP20 Advantys STB:

Suficientemente flexible como para adaptarse a prácticamente cualquier aplicación, esta solución abierta es a la vez una plataforma de E/S y de integración de sistemas, con las características siguientes:

- Hasta 32 módulos repartidos en 15 metros.
- Hot-Swap, configuración automática o por software, diagnóstico integrado, acceso remoto o local en tiempo real, gestión inteligente del ciclo de vida del sistema con el software de configuración Advantys.
- Tarjeta de memoria extraíble para duplicación rápida y fácil.
- Conexión directa a equipos Schneider (Magelis, arrancadores TeSys, Tego Power, variadores Altivar o de E/S Advantys FTB), así como a muchos otros equipos.
- Conectividad con los buses de campo: Modbus, Ethernet, CANopen, Devicenet, Profibus, and Interbus.

b. Descripción física del módulo de comunicación Advantys STB

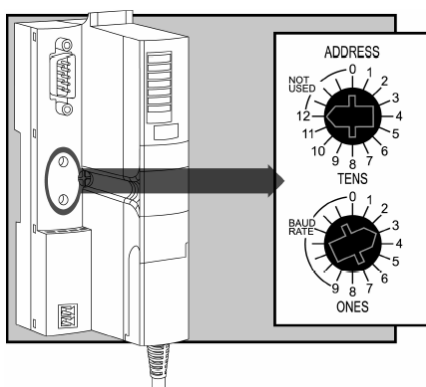


Característica	Función
1 Interface de bus de campo	Un conector SUB-D de 9 pins utilizado para conectar el NIM y el bus de la isla a un bus de campo CANopen.
2 Conmutador rotativo superior	Los dos conmutadores rotativos se utilizan juntos para especificar el ID de nodo del NIM en un bus de campo CANopen y para establecer el valor de la velocidad en baudios del bus de campo en el NIM.
3 Conmutador rotativo inferior	
4 Interface de fuente de alimentación	Un conector de dos receptáculos para conectar una fuente de alimentación externa de 24 VCC al NIM.
5 matriz de indicadores luminosos	Indicadores luminosos de color que utilizan varias secuencias para indicar visualmente el estado de funcionamiento del bus de la isla.
6 Tornillo de liberación	Mecanismo utilizado para retirar el NIM del segmento DIN. (Consulte el <i>Manual de planificación e instalación del sistema Advantys STB</i> para obtener más información.)
7 Tapa del puerto CFG	Cubierta abatible del panel frontal del NIM que cubre la interface CFG y el botón RST. El puerto CFG es sólo para actualizaciones de firmware.

c. Procedimiento para establecer la velocidad y la dirección

Los pasos a seguir son los siguientes:

1. Disminuir la alimentación de la isla.
2. Con un pequeño destornillador, establecer el conmutador rotativo inferior en cualquier posición después de 9 (BAUD RATE).
3. Decidir el valor de baudios que se empleará para las comunicaciones del bus de campo.
4. Determinar la posición del conmutador superior que corresponde a los baudios seleccionados
5. Con un pequeño destornillador, establecer el conmutador rotativo superior en la posición que corresponde al valor de baudios seleccionado.
6. Encender la isla para emplear la nueva configuración.



Posición (conmutador superior)	Velocidad en baudios
0	10.000 bits/s
1	20.000 bits/s
2	50.000 bits/s
3	125.000 bits/s
4	250.000 bits/s
5	500.000 bits/s
6	800.000 bits/s
7	1 Mbit/s

5.2. E/S distribuidas OTB

a. introducción



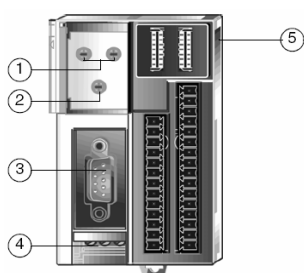
E/S distribuidas funcionales IP20 Advantys OTB:

Destinada a las máquinas sencillas y compactas, esta solución es un sistema optimizado, sencillo y económico de E/S distribuidas. Además, gracias a la compatibilidad de los módulos de extensiones Twido, existen varias posibilidades para crear grupos modulares:

- 3 módulos básicos de comunicación con 20 E/S integradas cada uno.

- Posibilidad de extensión gracias a módulos de E/S de 8 a 32 vías (hasta 264 vías).
- Gama de destino de 24 V CC.
- Amplia gama de E/S “Todo o Nada” y analógicas, comunes a los controladores programables Twido.
- Conexión de captadores/accionadores por borneros de tornillo desenchufables.
- Montaje directo en perfil DIN.
- Compacidad idónea: 20 E/S en 55 mm de ancho, conexión de bus incluida.
- Conectividad con los buses de campo: Modbus, Ethernet, CANopen.

b. Descripción física del módulo de comunicación Advantys OTB

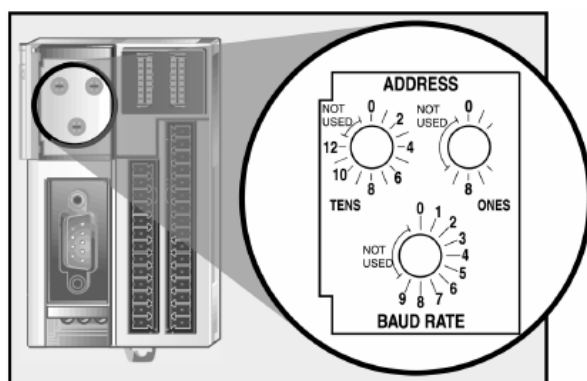


1. Dos ruedas codificadoras (ajuste de la dirección de la isla en el bus o la red).
2. Una rueda codificadora (ajuste de la velocidad de comunicación en el bus o la red).
3. Conectores para la conexión del bus CANOpen.
4. Bornes para la conexión de la alimentación 24 VCC.
5. Indicadores luminosos (estados de la comunicación y de las E/S).

c. Procedimiento para establecer la velocidad y la dirección

Los pasos a seguir son los siguientes:

1. Quitar la alimentación de la isla
2. Seleccionar una dirección de nodo
3. Ajustar los conmutadores rotativos superiores (Rueda izquierda de 0 a 12 -valor decenas-, Rueda derecha de 0 a 9 -valor unidades-)
4. Conectar la isla para implementar la nueva configuración



Position (lower encoder rate)	Baud Rate
0	10 kbits/s
1	20 kbits/s
2	50 kbits/s
3	125 kbits/s
4	250 kbits/s
5	500 kbits/s
6	800 kbits/s
7	1 Mbits/s
8	Automatic
9	Default rate (250 kbits/s)

5.3. Variador de velocidad ATV31



a. introducción

El variador Altivar 31 es un convertidor de frecuencia para motores asíncronos trifásicos de jaula. Los variadores Altivar 31 se comunican con los buses industriales Modbus y CANopen. Estos dos protocolos se integran de fábrica en el variador. Incluye funciones que se ajustan a las aplicaciones más usuales, en particular:

- Mantenimiento (cintas transportadoras pequeñas, polipastos...).
- Máquinas de envase y embalaje.
- Máquinas especiales (mezcladores, trituradores, maquinaria textil...).
- Bombas, compresores, ventiladores.

b. Descripción física del ATV31



1. 4 visualizadores de 7 segmentos
2. LED rojo "bus CC en tensión"
3. Para pasar al menú o al parámetro previo o para aumentar el valor mostrado
4. 2 indicadores de estado CANopen
5. Para salir de un menú o de un parámetro o para desechar el valor mostrado y volver al valor anterior grabado en la memoria
6. Para entrar en un menú o en un parámetro o para registrar el parámetro o el valor mostrado

c. Procedimiento para establecer la velocidad y la dirección

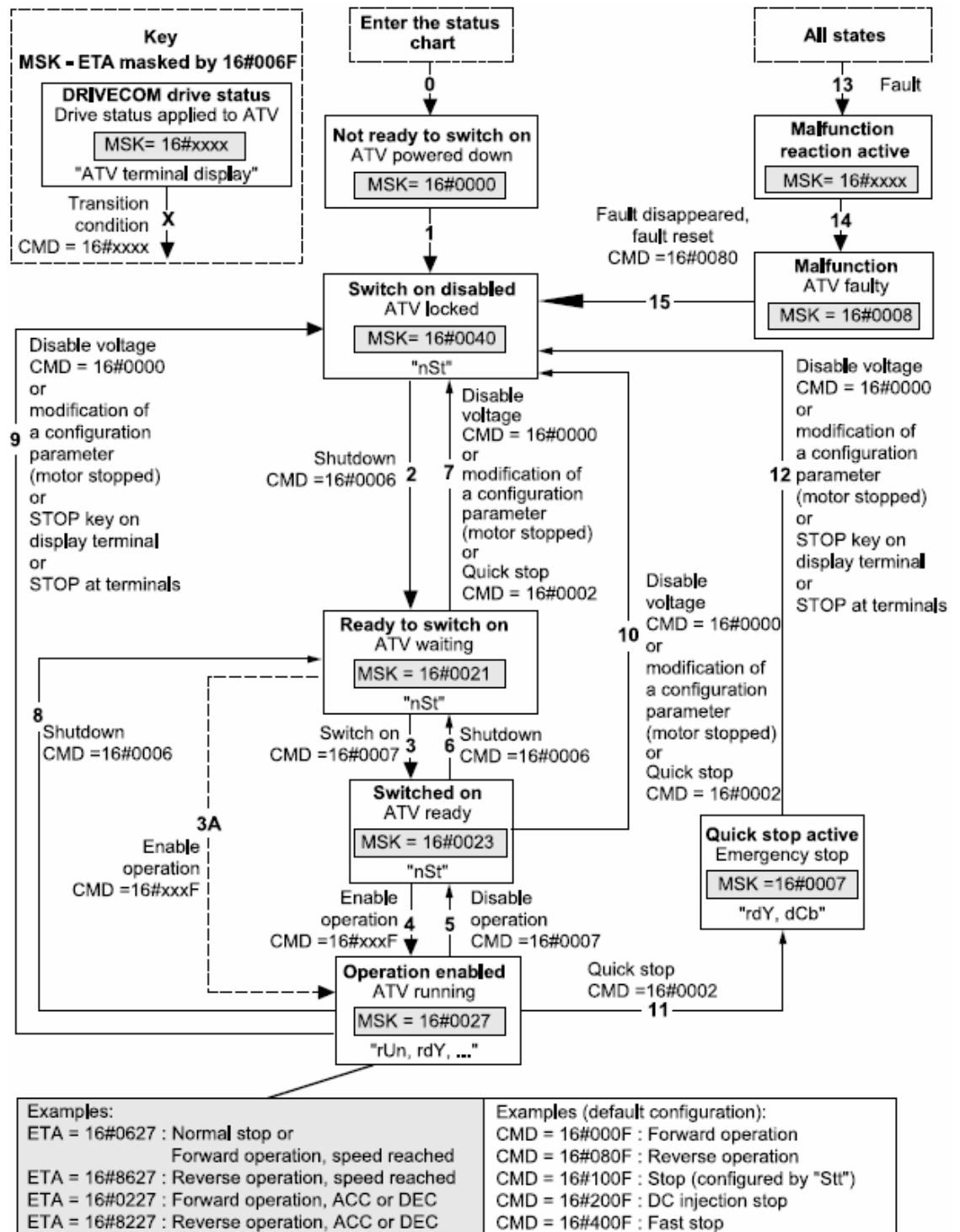
Los pasos a seguir son los siguientes:

1. Pasar a valores de fábrica desde el parámetro "FCS" que se encuentra en varios menús entre ellos "DrC".
2. Poner el número de esclavo en el menú comunicaciones "COM" opción "CoAd".
3. Poner la velocidad del transferencia del bus dentro del menú de comunicaciones "COM" en la opción "CoBd"

4. Poner que el control del variador vendrá vía bus en el menú de control "CTL" la opción de canal de referencia "FR1" ponemos el parámetro CAN y aceptamos

d. Estado del ATV31 (Status Chart)

Se trata de los diferentes estados del ATV31:



5.4. Variador de velocidad ATV61-71

a. introducción



El variador Altivar 61 es un convertidor de frecuencia para motores asíncronos trifásicos de 0,75 kW a 630 kW. Está dedicado a las aplicaciones más habituales de la gestión de fluidos en los edificios industriales y los edificios del sector terciario (HVAC "Heating Ventilation Air Conditioning"): Ventilación, Climatización, Bombeo.

La gama de variadores de velocidad Altivar 71 permite satisfacer las mayores exigencias gracias a diferentes tipos de control motor y numerosas funcionalidades integradas: Elevación, Manutención, Embalaje, Máquinas textiles, Máquinas para madera, Máquinas de proceso, Ascensores.

b. Descripción física del ATV61/71



1. Visualizador gráfico (8 líneas, 240_160 píxeles - Visualización de gráficos de barras)
2. Teclas de funciones asignables F1, F2, F3, F4 (Funciones de diálogo: acceso directo, pantallas de ayuda, navegación - Funciones de aplicación: "Local Remote", velocidad preseleccionada.)
3. Tecla "STOP/RESET": control local de parada del motor/borrado de los fallos.
4. Tecla "RUN": control local de marcha del motor.
5. Botón de navegación (Pulsación: grabación del valor en curso (ENT) - Rotación ±: incrementa o disminuye el valor, pasa a la línea siguiente o a la anterior.)
6. Tecla "FWD/REV": inversión del sentido de rotación del motor.
7. Tecla "ESC": abandono de un valor, de un parámetro o de un menú para volver a la elección anterior.

c. Procedimiento para establecer la velocidad y la dirección

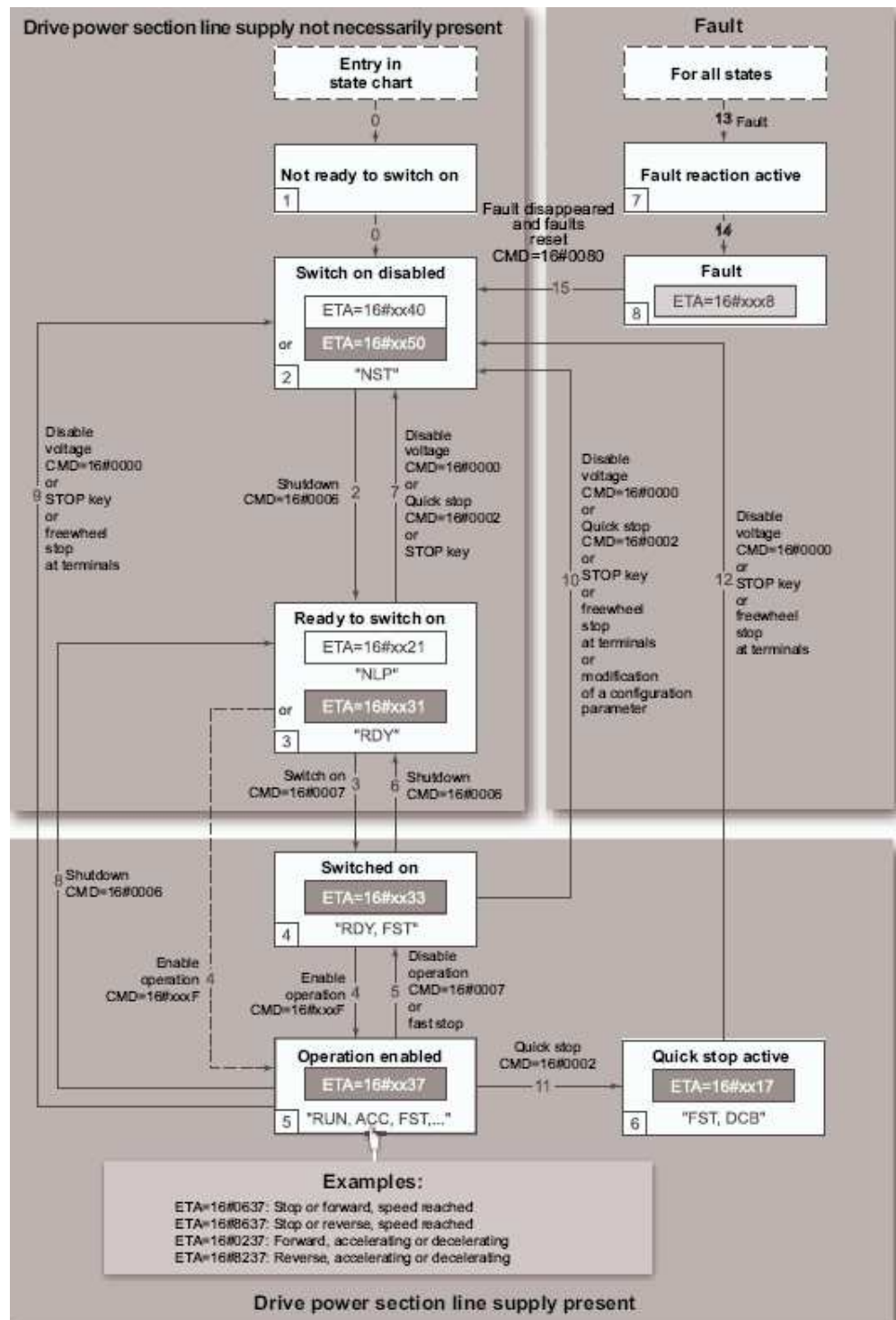
Los pasos a seguir son los siguientes:

1. Pasar a valores de fábrica desde el parámetro "FCS" que se encuentra en varios menús entre ellos "DrC".
2. Poner el número de esclavo en el menú comunicaciones "COM" opción "CoAd".
3. Poner la velocidad del transferencia del bus dentro del menú de comunicaciones "COM" en la opción "CoBd"

4. Poner que el control del variador vendrá vía bus en el menú de control "CTL" la opción de canal de referencia "FR1" ponemos el parámetro CAN y aceptamos

d. Estado del ATV61/71 (Status Chart)

Se trata de los diferentes estados del ATV61/71:



5.5. Control de movimiento Lexium 05

a. introducción



Este LXM05 es un servo accionamiento AC de aplicación universal. Los valores de consigna los controla y predetermina un controlador programable de nivel superior, p. ej. Premium.

En combinación con los servomotores seleccionados de Schneider Electric se consigue un potente y muy compacto sistema de accionamiento. En la parte frontal se encuentra la posibilidad de introducción de una parametrización sencilla (HMI) con unidad de visualización y teclas de manejo.

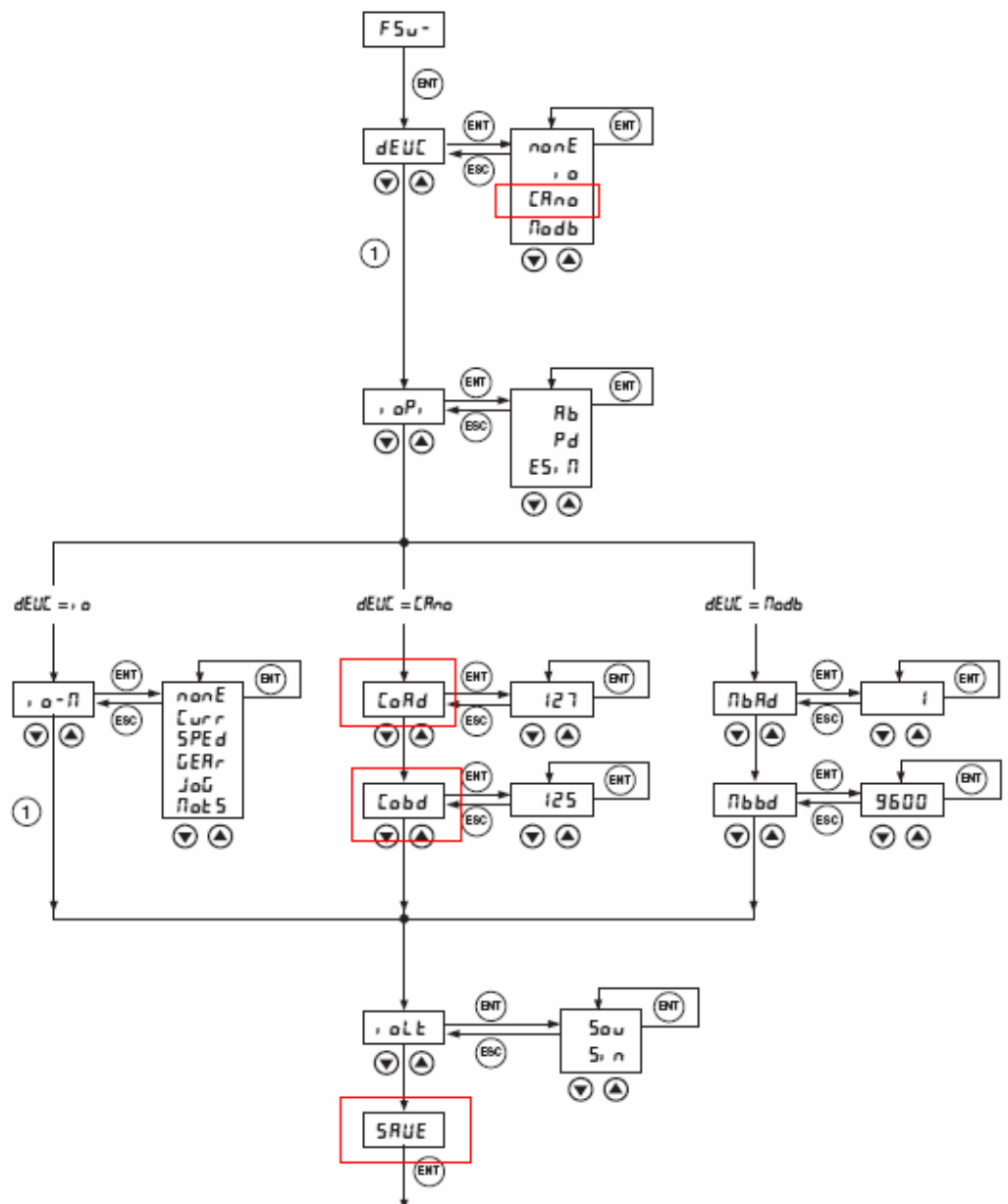
El valor de consigna se predetermina a través de:

- Bus de campo: Modbus o CANopen para posicionamientos punto a punto, control de velocidad, secuencia de movimiento así como control del par y velocidad
- Señales analógicas de ± 10 V para la regulación del par motor o para el control de velocidad. La comunicación de la posición real del motor se realiza a través de señales de encoder A/B
- Interface de posición: Señales de pulso/dirección o señales de encoder A/B para la realización de un engranaje electrónico

b. Descripción física del comunicador



- | | |
|-----|--|
| (1) | LED rojo encendido: La tensión está ajustada al bus DC |
| (2) | Indicación de estado |
| (3) | LEDs para bus de campo |
| ESC | <ul style="list-style-type: none"> • Abandonar un menú o parámetro • Retorno del valor mostrado del último valor salvado |
| ENT | <ul style="list-style-type: none"> • Seleccionar un menú o parámetro • Grabación del valor mostrado en EEPROM |
| ▲ | <ul style="list-style-type: none"> • Cambio al menú o parámetro anterior • Aumento del valor mostrado |
| ▼ | <ul style="list-style-type: none"> • Cambio al menú o parámetro siguiente • Reducción del valor mostrado |



5.6. Módulo para arrancador motor TeSysU

a. introducción

El arrancador controlador Tesys modelo U es una salida motor que realiza las siguientes funciones:

- Protección y control de motores monofásicos o trifásicos: seccionamiento de potencia, protección contra las sobrecargas y los cortocircuitos, protección contra las sobrecargas térmicas, conmutación de potencia
- Control de la aplicación: alarmas de las protecciones, supervisión de la aplicación (duración de utilización, número de disparos, valores de las corrientes de motores, etc.), históricos (registro de los 5 últimos disparos con el valor de los parámetros del motor).



Arranque motor
TeSysU

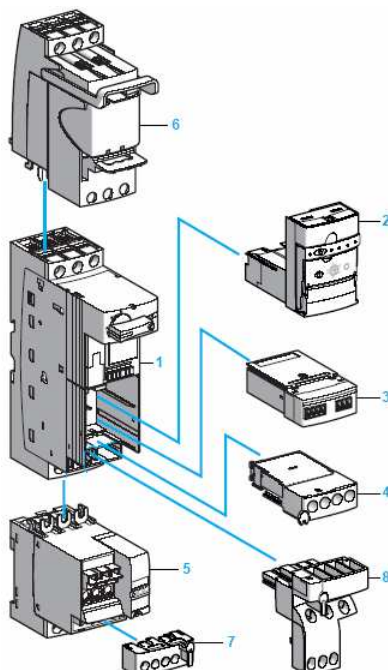


Modulo de
comunicación
CANOpen

Estas funciones se integran mediante simple fijación a una base de potencia en forma de unidad de control y de módulos de funciones. Esta personalización puede realizarse en el último momento. Los accesorios de instalación simplifican e incluso eliminan el cableado entre los diferentes elementos.

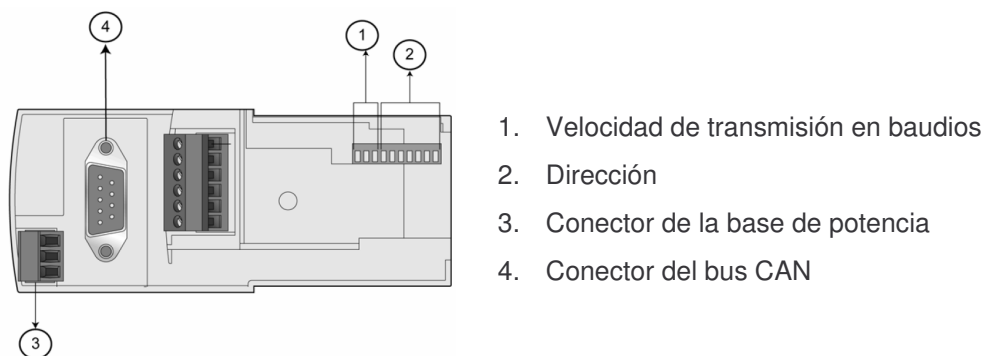
b. Descripción física del comunicador

Se puede componer del conjunto de material siguiente:



1. Base de potencia
2. Unidades de control
3. Módulo de comunicación
4. Módulos de contactos auxiliares
5. Bloque inversor
6. Limitador seccionador LUA LB
7. Borneros desenchufables
8. Sistema de precableado de control

Esta es una vista inferior de un módulo de comunicación LULC08:



c. Procedimiento para establecer la velocidad y la dirección

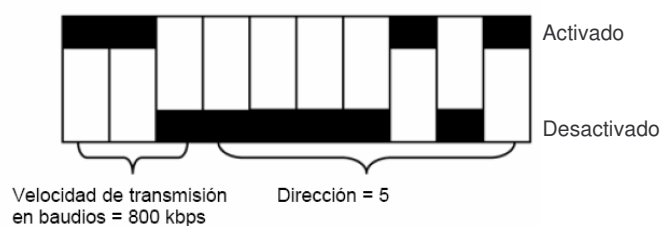
- **Velocidad de transmisión en baudios:** El sistema le permite asignar una velocidad de transmisión en baudios usando los tres interruptores situados más a la izquierda (de SW8 a SW10).

Consulte la siguiente tabla:

SW10	SW9	SW8	Velocidad de transmisión en baudios
0	0	0	10 kbps
0	0	1	20 kbps
0	1	0	50 kbps
0	1	1	125 kbps
1	0	0	250 kbps (valor predeterminado)
1	0	1	500 kbps
1	1	0	800 kbps
1	1	1	1.000 kbps

- **Dirección:** El sistema le permite asignar una dirección de 1 a 127 utilizando los siete interruptores situados más a la derecha (de SW1 a SW7). La dirección 0 (cero) no está permitida y se considera como una configuración no válida.

- **Ejemplo:**



5.7. Encoder con conexión a CANOpen

a. introducción



El codificador absoluto multivuelta Osicoder® 58 mm CANOpen se ha desarrollado para responder a las configuraciones que se encuentran en las instalaciones industriales de comunicación. Los modelos XCC 3510PS84CB y XCC3515CS84CB incluyen de fábrica los protocolos de comunicación CANOpen.

El interface CANbus integrado en el codificador giratorio absoluto incluye todas las funciones CANOpen. El protocolo de aplicación incluye la programación de las funciones adicionales siguientes: secuencia del código, resolución por vuelta, resolución global, preselección, velocidad y dirección.

Parámetros configurables:

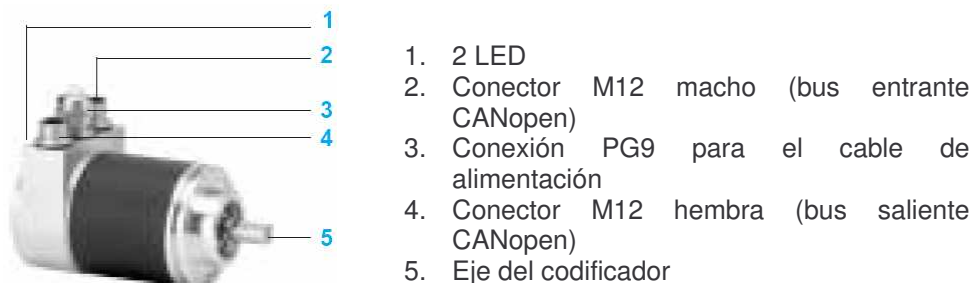
- Velocidad de la transmisión: Valor por defecto: 250 kbaudios, configurable de 10 kbaudios (distancia de 6.700 m) a 1 Mbaudio (distancia de 40 m).
- Dirección: Define la identificación del codificador en el bus, de 1 a 89. Valor por defecto: id = 1. Se define gracias a 2 ruedas codificadoras situadas en la caja.
- Resolución: Define el número de puntos por vuelta (0 a 8.191).
- Resolución global: Define el número de códigos total del codificador (0 a 33.554.431).
- Sentido: Permite definir el sentido de contaje del codificador (creciente a la derecha o a la izquierda) según su posición mecánica.
- RAX: Define el valor de su posición actual (puesta a X o puesta a la cifra).

Modos de comunicación:

- Modo Pooling: El codificador responde a las demandas del maestro. Este modo permite programar y consultar los parámetros del codificador en su posición.
- Modo Cyclic: El codificador transmite su posición cíclicamente. El período de emisión es programable de 0 a 65.535 ms.
- Modo Sync: El codificador transmite su posición bajo pedido de sincronización del maestro.

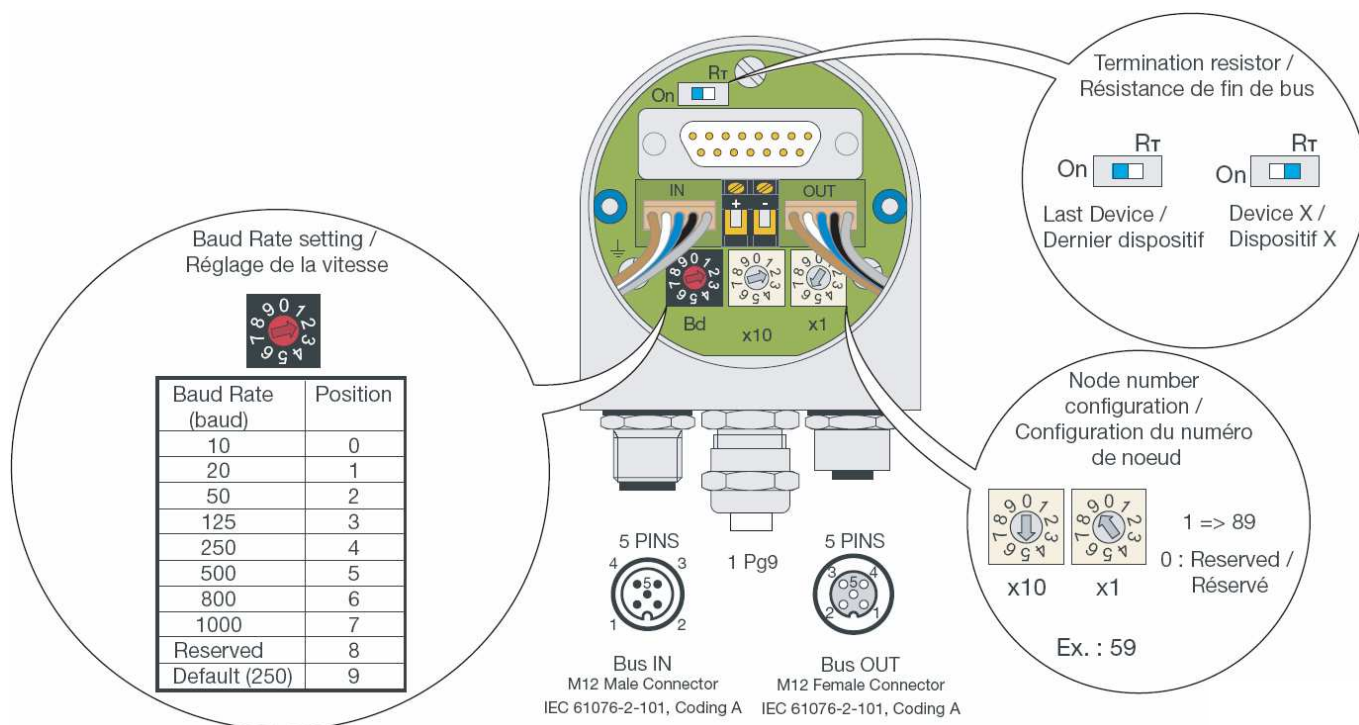
b. Descripción física de los encoders

Descripción física del encoder absoluto:



c. Procedimiento para establecer la velocidad y la dirección

El Osicoder en CANOpen permite los ajustes siguientes:



						Clamp / Borne via Pg9	
PIN	1	2	3	4	5	+	-
Description / Fonction	CAN_SHLD	(CAN_V+)	CAN_GND	CAN_H	CAN_L	24V Supply voltage	0V Supply voltage

6. Analizador de tramas CANOpen

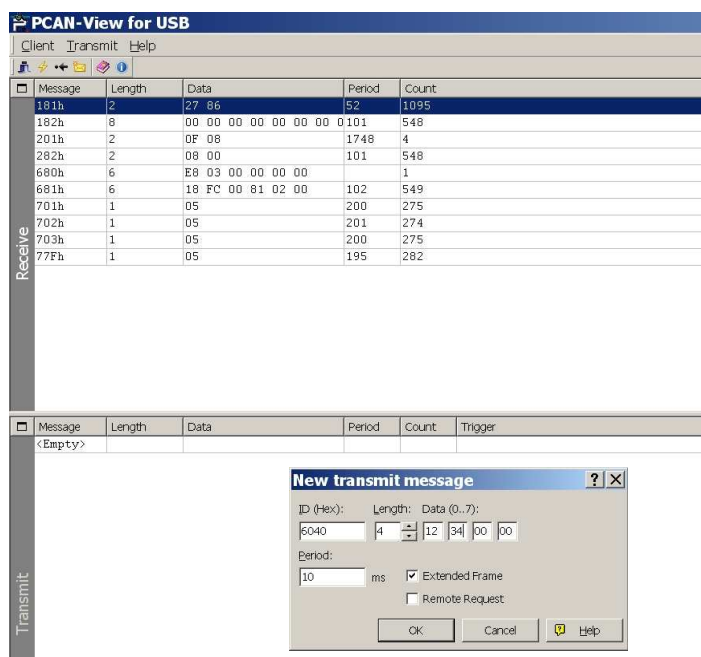
6.1. Introducción

Si se considera que el cableado / conexionado y la configuración / programación de los equipos conectados al bus son los correctos, es posible utilizar un dispositivo físico y un software para poder visualizar las tramas en el bus CANOpen.

Se trata un adaptador USB (para conectar al PC) - DB9-pines (para conectar en el bus) con posibilidad de conectar un adaptador DB9-pines – RJ45 y de un software para visualizar las tramas. Además de visualizar las tramas es posible mandar SDOs (maestro-esclavo).



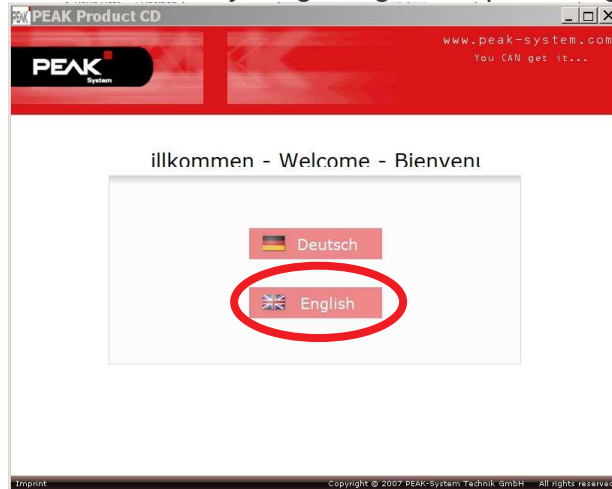
Adpatador CANOpen (USB - SubD-9pines)



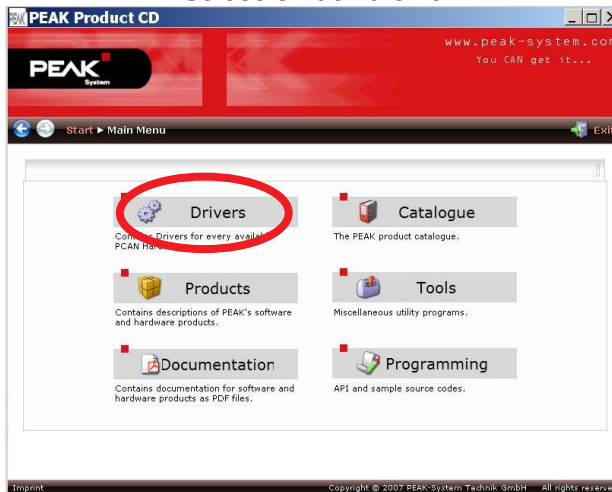
Software para visualizar/transmitir mensajes

6.2. Instalación del sniffer de CANOpen: PEAK

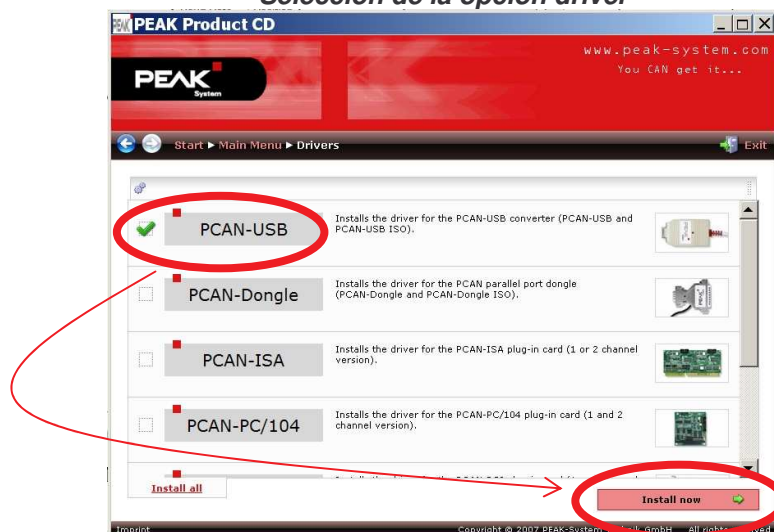
La instalación del software y driver se realiza mediante un CD. Se ha de insertar el CD en el lector y luego seguir los pasos siguientes:



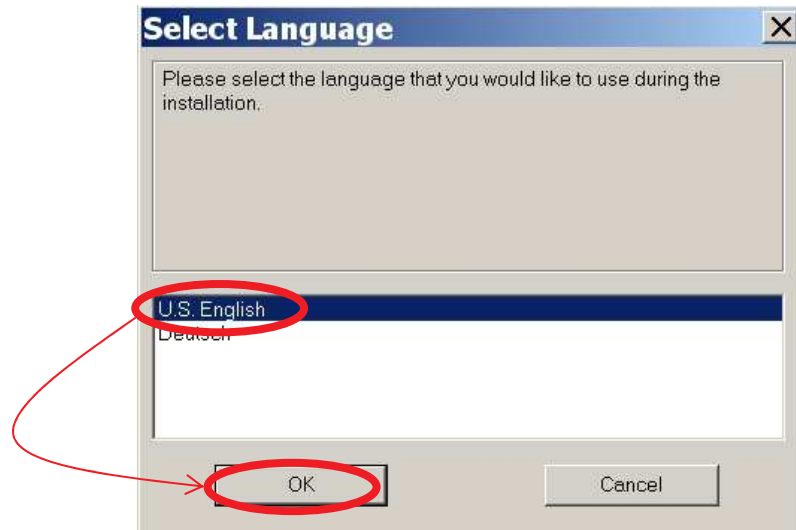
Selección del idioma



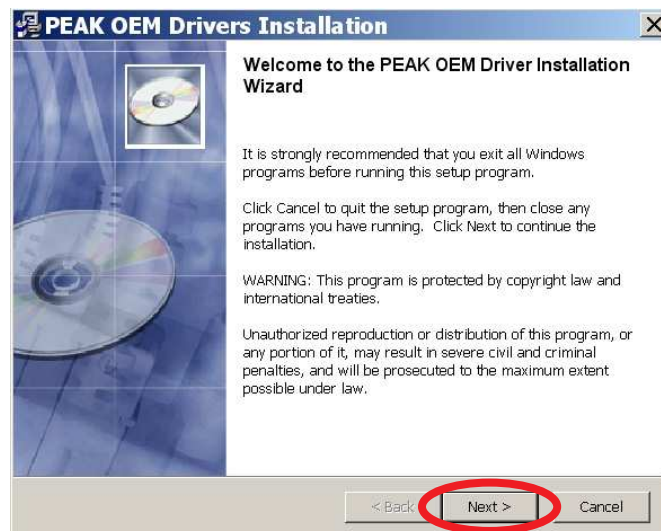
Selección de la opción driver



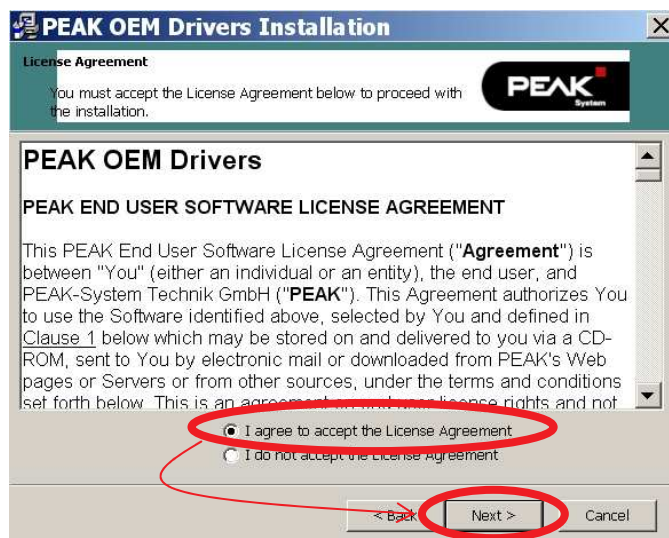
Selección del driver a instalar



Selección del idioma durante la instalación



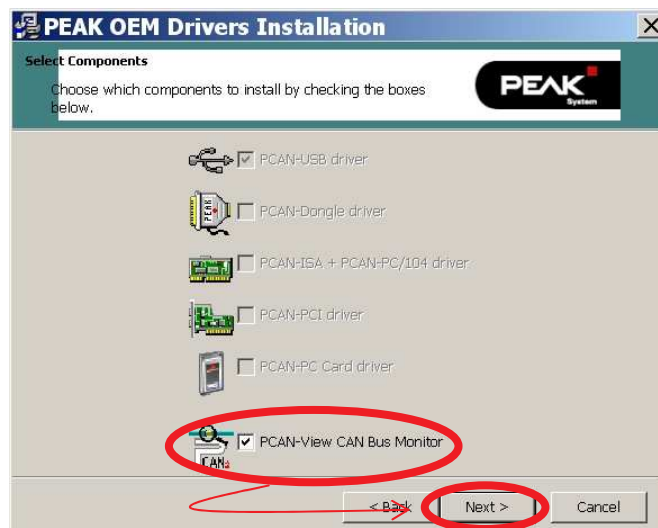
Pulsar Next para seguir la instalación



Aceptar los términos de la licencia y pulsar en Next



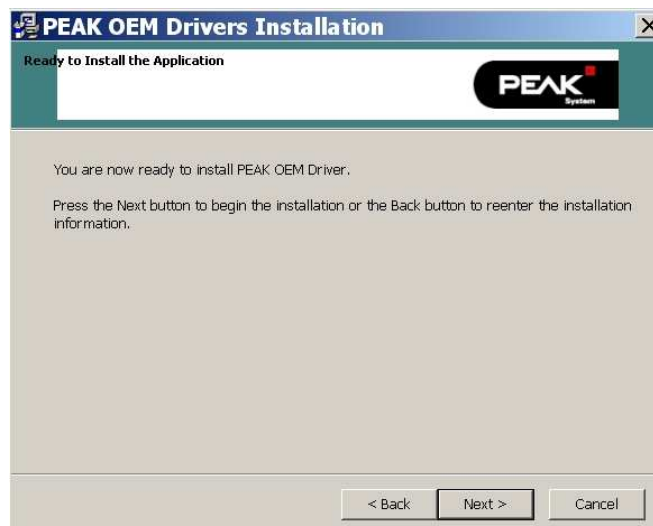
Pulsar Next para seguir la instalación



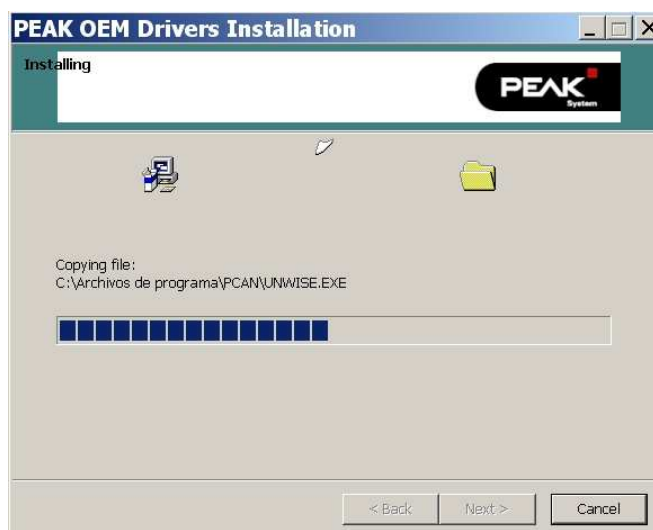
Pulsar Next para seguir la instalación



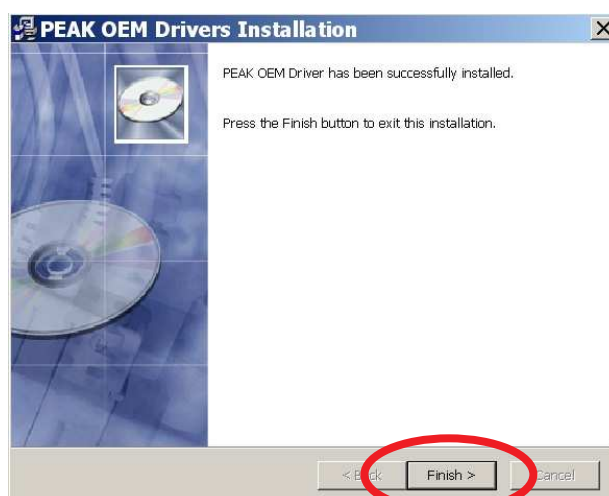
Pulsar Next para seguir la instalación



Pulsar Next para seguir la instalación



Esperar que se instale el driver y el software



Pulsar finish para seguir la instalación

Al insertar el adaptador PEAK, el ordenador tendrá que reconocer el nuevo material utilizando los drivers instalados previamente.

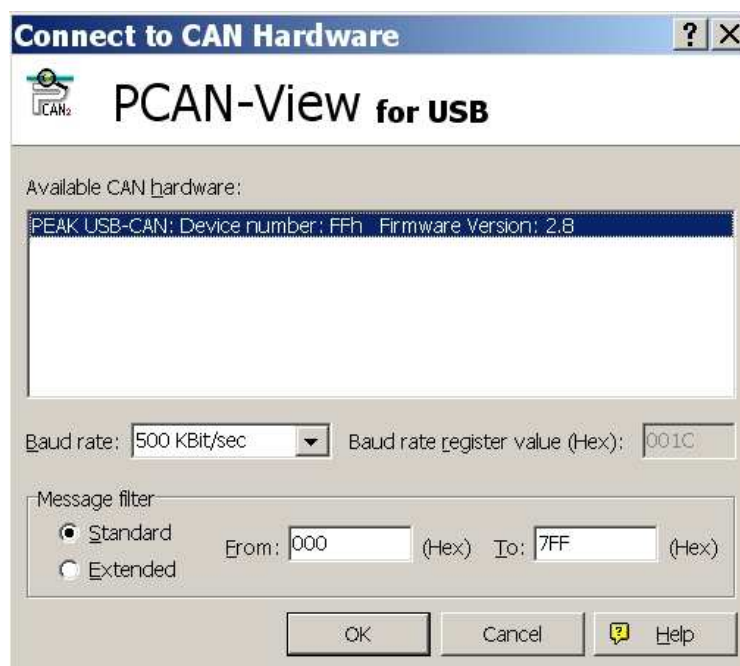
6.3. Utilización del sniffer de CANOpen: PEAK

Para iniciar el sniffer se ha de ir al menú inicio > programa > PCAN Hardware y seleccionar PCAN-View USB:



Iniciar el sniffer de CANOpen

Aparecerá una ventana en la que podemos configurar la velocidad del bus CANOpen al cual nos vamos a conectar y un filtro (corresponde a los COB-IDs posibles en el bus CANOpen)



Página de inicio

Por defecto están seleccionados la velocidad de 500kBit/sec y un filtro de tipo estándar. Además podemos ver que el sistema reconoce el adaptador ya que nos aparece en la lista de hardware CAN disponible.

Al pulsar en OK, aparece una ventana en la que podemos visualizar las tramas de CANOpen recibidas y mensajes transmitidos (desde el propio sniffer).

PCAN-View for USB						
Client Transmit Help						
<input type="checkbox"/>	Message	Length	Data	Period	Count	
Receive	181h	2	27 86	52	1095	
	182h	8	00 00 00 00 00 00 00 00	0 101	548	
	201h	2	0F 08	1748	4	
	282h	2	08 00	101	548	
	680h	6	E8 03 00 00 00 00		1	
	681h	6	18 FC 00 81 02 00	102	549	
	701h	1	05	200	275	
	702h	1	05	201	274	
	703h	1	05	200	275	
	77Fh	1	05	195	282	
Transmit	<input type="checkbox"/>	Message	Length	Data	Period	Count
		<Empty>				

Ventana principal del sniffer

En la parte superior aparecen los mensajes recibidos. La información que se puede leer es la siguiente:

- **Message:** número de COB-IDs en hexadecimal
- **Length:** longitud del mensaje
- **Data:** datos
- **Period:** periodo de refresco
- **Count:** contador de eventos

<input type="checkbox"/>	Message	Length	Data	Period	Count
Receive	181h	2	27 86	50	2222
	182h	8	00 00 00 00 00 00 00 00	0 101	1124
	201h	2	0F 08	1748	4
	282h	2	08 00	100	1124
	680h	6	E8 03 00 00 00 00		1
	681h	6	18 FC 00 81 02 00	100	1122
	701h	1	05	200	564
	702h	1	05	201	563
	703h	1	05	200	564
	77Fh	1	05	195	578

En la parte inferior aparecen los mensajes transmitidos por el sniffer. La información que se puede leer es la siguiente:

- **Message:** número de COB-IDs en hexadecimal
- **Length:** longitud del mensaje
- **Data:** datos
- **Period:** periodo de refresco
- **Count:** contador de eventos

<input type="checkbox"/>	Message	Length	Data	Period	Count	Trigger
<input checked="" type="checkbox"/>	00006040h	4	12 34 00 00	10	219	Time

Transmit

New transmit message

?

×

ID (Hex):

Length:

Data (0..7):

6040

4

12

34

00

00

Period:

10

ms

☒ Extended Frame

☐ Remote Request

OK

Cancel

?

Help

7. Glosario

%_____

%IW

Según la norma IEC, %IW indica un objeto de lenguaje de entrada analógica.

%M

Según la normativa IEC, %M indica un objeto de lenguaje de tipo bit de memoria.

%MW

Según la normativa IEC, %MW indica un objeto de lenguaje de tipo palabra de memoria.

%QW

Según la norma IEC, %QW indica un objeto de lenguaje de salida analógica.

A_____

ARRAY

Una ARRAY es una tabla de elementos del mismo tipo. La sintaxis es la siguiente: ARRAY [<rango>] OF <Tipo> Ejemplo: ARRAY [1..2] OF BOOL es una tabla de una dimensión compuesta por dos elementos de tipo BOOL. ARRAY [1..10, 1..20] OF INT es una tabla de dos dimensiones compuesta por 10 x 20 elementos de tipo INT.

B_____

BIT

Se trata de una unidad binaria para una cantidad de información que puede representar dos valores (o estados) diferentes: 0 ó 1.

BOOL

BOOL son las siglas del tipo booleano. Éste es el elemento de datos básico en computación. Una variable de tipo BOOL tiene un valor: 0

(FALSE) o 1 (TRUE). Un bit de extracción de palabras de tipo BOOL, por ejemplo: %MW10.4.

BYTE

Un BYTE es un conjunto de ocho bits. Un BYTE puede introducirse en binario o en base 8. El tipo BYTE se codifica en formato de ocho bits que, en formato hexadecimal, tiene un rango de 16#00 a 16#FF.

C_____

CADENA DE CARACTERES

Una variable de tipo STRING es una cadena de caracteres ASCII. La longitud máxima de una cadena de caracteres es de 65.534 caracteres.

CAN

Controller Area Network (Red de área del controlador). El protocolo CAN (ISO 11898) para redes de bus serie está diseñado para la interconexión de dispositivos inteligentes (de varios fabricantes) en sistemas inteligentes para aplicaciones industriales de tiempo real. Los sistemas CAN multimaestro aseguran una alta integridad de datos por medio de la aplicación de la difusión de mensajes y de mecanismos de error avanzados. CAN, diseñado originalmente para utilizarlo en automóviles, se utiliza actualmente en una amplia variedad de entornos industriales de control automático.

CANopen

CANopen es un protocolo de nivel superior que se emplea en redes de automatización. Está basado en la capa de aplicación CAN (CAL) de conformidad con la norma CiA DS 301 (EN 50325-4).

CEM

Electromagnetic Compatibility (Directiva de compatibilidad electromagnética). Los dispositivos que cumplen los requisitos de CEM pueden funcionar dentro de los límites electromagnéticos que estima el sistema sin que se produzcan errores.

COB

Communication Object: unidad de transporte en el bus CAN. Un COB se identifica mediante un identificador único codificado en 11 bits, [0,

2047]. Un COB contiene, al menos, 8 bytes de datos. La prioridad de transmisión de un COB viene determinada por su identificador; cuanto menor sea éste, mayor prioridad tiene el COB asociado.

CRC

Cyclic Redundancy Checksum: checksum de redundancia cíclica que indica que no se ha "deformado" ningún carácter durante la transmisión de la trama.

D_____

DATE

El tipo DATE codificado en BCD en un formato de 32 bits contiene la información siguiente: el año codificado en un campo de 16 bits, el mes codificado en un campo de 8 bits, el día codificado en un campo de 8 bits. El tipo DATE debe introducirse de la manera siguiente: D#<Año>-<Mes>-<Día>

DDT

DDT es la forma abreviada de «Derived Data Type» (tipo de datos derivados). Un tipo de datos derivados es un conjunto de elementos del mismo tipo (ARRAY) o de distintos tipos (estructura).

DFB

DFB (Derived Function Block). Los tipos de DFB son bloques de funciones que el usuario puede programar en ST, IL, LD o FBD.

DINT

DINT es la forma abreviada de «Double INTeger» (entero doble) (codificado en 32 bits). Los límites inferior y superior figuran a continuación: de - (2 elevado a 31) a (2 elevado a 31) - 1. Ejemplo: - 2.147.483.648, 2.147.483.647, 16#FFFFFFFF.

DT

DT es la forma abreviada de «Date and Time» (fecha y hora). El tipo DT, codificado en BCD en un formato de 64 bits, contiene esta información: el año codificado en un campo de 16 bits, el mes codificado en un campo de 8 bits, el día codificado en un campo de 8 bits, la hora codificada en un campo de 8 bits, los minutos codificados

en un campo de 8 bits, los segundos codificados en un campo de 8 bits. Nota: No se utilizan los 8 bits menos significativos. El tipo DT debe introducirse así: DT#<Año>-<Mes>-<Día>-<Hora>:<Minutos>:<Segundos>

DWORD

DWORD es la forma abreviada de «Double Word» (palabra doble). El tipo DWORD se codifica en un formato de 32 bits.

E_____

EBOOL

EBOOL (Extended Boolean). Pueden utilizarse para administrar flancos ascendentes o descendentes, así como para forzar. Una variable de tipo EBOOL ocupa un byte de memoria.

EDS

Electronic Data Sheet: archivo de descripción de cada equipo CAN (suministrado por los fabricantes). Con el software de configuración Sycon, si se desea agregar un equipo CAN al bus, es necesario seleccionar el EDS correspondiente. Los EDS están disponibles en la página web <http://www.can-cia.de> o pueden ser suministrados por los proveedores de hardware.

EF

EF es la forma abreviada de «Elementary Function» (función elemental). Se trata de un bloque, utilizado en un programa, que realiza una función de software predefinida. Una función no dispone de información sobre el estado interno. Varias llamadas de la misma función con los mismos parámetros de entrada muestran siempre los mismos valores de salida. Encontrará información sobre la forma gráfica de la llamada de la función en el «[bloque funcional (instancia)]». Al contrario que las llamadas de bloques de funciones, las llamadas de función constan únicamente de una salida a la que no se ha asignado un nombre y cuyo nombre coincide con el de la función. En FBD, cada llamada se indica mediante un [número] único a través del bloque gráfico. Este número se genera automáticamente y no puede modificarse. Coloque y configure estas funciones en el

programa para ejecutar su aplicación. También puede desarrollar otras funciones mediante el kit de desarrollo de SDKC.

EFB

Son las siglas de bloque de funciones elemental (Elementary Function Block). Se trata de un bloque que se utiliza en un programa y que realiza una función de software predefinida. Los EFB tienen estados y parámetros internos. Aún cuando las entradas sean idénticas, los valores de salida pueden ser diferentes. Por ejemplo, un contador tiene una salida que indica que se ha alcanzado el valor de preselección. Esta salida se establece en 1 cuando el valor actual es igual al valor de preselección.

EN

EN significa «ENable» (activar); se trata de una entrada de bloque facultativa. Cuando la entrada EN está activada, se establece una salida ENO automáticamente. Si $EN = 0$, el bloque no está activado, su programa interno no se ejecuta y ENO se establece en 0. Si $EN = 1$, se ejecuta el programa interno del bloque y ENO se establece en 1. Si se produce un error, ENO se establece en 0. Si la entrada EN no está conectada, se establece automáticamente en 1.

ENO

ENO significa «Error NOtification» (notificación de error); se trata de la salida asociada a la entrada facultativa EN. Si ENO se establece en 0 (porque $EN = 0$ o en caso de que se produzca un error de ejecución): el estado de las salidas de bloques de funciones permanece idéntico a aquél en el que estaban durante el último ciclo de exploración ejecutado correctamente; la(s) salida(s) de funciones, así como los procedimientos, se establecen en 0.

F

FBD

FBD son las siglas de diagrama de bloques de funciones (Function Block Diagram). FBD es un lenguaje de programación gráfico que funciona como si se tratara de un diagrama lógico. Además de los bloques lógicos simples (AND, OR, etc.), cada función o bloque de funciones del programa se representa mediante esta forma gráfica.

En cada bloque, las entradas se sitúan a la izquierda y las salidas, a la derecha. Las salidas de los bloques pueden conectarse a las entradas de otros bloques para formar expresiones complejas.

FFB

Término colectivo para EF (función elemental), EFB (bloque de funciones elemental) y DFB (bloque de funciones derivado).

G_____

H_____

HMI

Human-Machine Interface (Interface hombre-máquina). Interface del operador, generalmente gráfica, para equipos industriales.

I_____

IEC 61131-3

Normativa internacional: controles de lógica programables - Apartado 3: lenguajes de programación.

IEEE

Institute of Electrical and Electronics Engineers, Inc (Instituto de Ingeniería Eléctrica y Electrónica). Organismo internacional de normas y de evaluaciones de conformidad para todos los campos de la electrotecnia, incluidas la electricidad y la electrónica.

IL

IL son las siglas de lista de instrucciones (Instruction List). Este lenguaje consiste en una serie de instrucciones básicas. Este lenguaje es muy similar al lenguaje ensamblador utilizado en los procesadores de programa. Cada instrucción está compuesta por un código de instrucción y un operando.

Indicador LED

Light Emitting Diode (Diodo electroluminiscente). Indicador que se enciende cuando pasa por él electricidad. Señala el estado de funcionamiento del módulo de comunicaciones.

IODDT

IODDT es la forma abreviada de «Input/Output Derived Data Type» (tipo de datos derivados de E/S). El término IODDT designa un tipo de datos estructurado que representa un módulo o un canal de un módulo del PLC. Cada módulo experto posee sus propios IODDT.

Instancia DFB

Un tipo de instancia DFB se produce cuando se llama a una instancia desde un editor de lenguaje. La instancia procesa un nombre, interfaces de entrada/salida, las variables públicas y privadas se duplican (una duplicación por instancia, el código no se duplica). Un tipo DFB puede disponer de varias instancias.

Instanciar

Instanciar un objeto significa asignarle un espacio de memoria cuyo tamaño dependerá del tipo de objeto que se va a instanciar. Cuando se instancia un objeto, éste está disponible y el programa puede manipularlo.

J _____

K _____

L _____

LD

LD son las siglas de diagrama de contactos (Ladder Diagram). LD es un lenguaje de programación que representa las instrucciones que deben ejecutarse en forma de diagramas gráficos muy similares a los esquemas eléctricos (contactos, bobinas, etc.).

LT

Final de línea: Terminación del cable principal con resistencia de 120 W, que puede estar integrada en la caja de derivación o en el conector del cable.

M _____

Módulo de E/S

En un sistema de controlador programable, un módulo de E/S interactúa directamente con los sensores y actuadores de proceso/máquina. Este módulo es el componente que se monta en una base de E/S y proporciona las conexiones eléctricas entre el controlador y los dispositivos de campo. Las capacidades normales de un módulo de E/S se ofrecen en una amplia variedad de capacidades y niveles de señal.

Multidifusión

Las comunicaciones de multidifusión envían paquetes desde un solo origen a un grupo multidifusión predefinido de destinos de red, normalmente a través de un enrutador o un conmutador. El envío de mensajes a los miembros del grupo únicamente reduce el tráfico innecesario que se crea con las comunicaciones de multidifusión y elimina la necesidad de una transmisión unidifusión separada a cada destinatario. (Consulte multidifusión, unidifusión, GMRP.)

N _____

O _____

P _____

PCMCIA

Personal Computer Memory Card International Association

PDO

Process Data Object: el objeto para el intercambio de datos entre distintos elementos es CANopen. Existen los RPDO (Recieve PDO) y los TPDO (Transmit PDO).

Q _____

R _____

REAL

El tipo REAL es un tipo codificado en 32 bits.

Red (en lenguaje LD)

Una red es un conjunto de elementos gráficos interconectados. El campo de aplicación de una red es local, en relación con la unidad (sección) de organización del programa en el que se encuentra la red.

Red (con módulos de comunicación expertos)

Una red es un grupo de estaciones que se intercomunican. El término «red» se utiliza también para definir un grupo de elementos gráficos interconectados. Dicho grupo constituye una parte de un programa que puede componerse de un grupo de redes.

S_____

Sección

Módulo de programa perteneciente a una tarea que se puede escribir en el lenguaje elegido por el programador (FBD, LD, ST, IL o SFC). Una tarea puede estar compuesta por distintas secciones, y el orden de ejecución de estas secciones corresponde a su orden de creación. Este orden se puede modificar.

SDO

Service Data Object: comunicación entre pares con acceso al objeto de diccionario de un elemento de bus CANopen. existen los SSDO (Server SDO) y los CSDO (Client SDO).

SFC

Son las siglas de gráfica de función secuencial (Sequential Function Chart). SFC permite representar gráficamente y de forma estructurada el funcionamiento de un sistema de automatización secuencial. Esta descripción gráfica del comportamiento secuencial de un sistema de automatización y de las distintas situaciones resultantes se realiza utilizando símbolos gráficos simples.

ST

ST son las siglas del lenguaje de texto estructurado (Structured Text). Este lenguaje es un lenguaje de alto nivel similar a los lenguajes de programación de ordenadores. Permite estructurar series de instrucciones.

STRING

Una variable de tipo STRING es una cadena de caracteres ASCII. La longitud máxima de una cadena de caracteres es de 65.534 caracteres.

Subrutina

Módulo de programa perteneciente a una tarea (MAST, FAST) que se puede escribir en el lenguaje elegido por el programador (FBD, LD, ST, o IL). Una subrutina sólo se puede llamar desde una sección o desde otra subrutina que pertenezca a la tarea en la que se declare.

T_____

TAP (Caja de derivación)

Terminal Access Point: Caja de derivación conectada al cable principal, que permite conectar cables de derivación.

Tarea

Grupo de secciones y subrutinas ejecutadas cíclica o periódicamente si se trata de la tarea MAST, o periódicamente si se trata de la tarea FAST. Una tarea siempre tiene un nivel de prioridad y tiene asociadas entradas y salidas del PLC. Estas entradas y salidas se actualizarán en consecuencia.

Tarea principal

Tarea principal del programa. Es obligatoria y se utiliza para ejecutar un procesamiento secuencial del PLC.

TIME

El tipo TIME expresa una duración en milisegundos. Este tipo se codifica en formato de 32 bits y permite obtener periodos de 0 a (2 elevado a 32)-1 milisegundos.

TOD

TOD es la forma abreviada de «Time Of Day» (hora del día). El tipo TOD, codificado en BCD en un formato de 32 bits, contiene esta información: la hora codificada en un campo de 8 bits, los minutos codificados en un campo de 8 bits, los segundos codificados en un campo de 8 bits. Nota: No se utilizan los 8 bits menos significativos. El

tipo TOD debe introducirse de la manera siguiente:
TOD#<Hora>:<Minutos>:<Segundos>

Trama

Grupo de bits que compone un bloque binario de información. Las tramas contienen información o datos de control de red. La tecnología de red utilizada es la que determina el tamaño y la composición de una trama.

U _____

USB

Universal Serial Bus (Bus de serie universal). Un USB es una interface de hardware casi universal para la conexión de dispositivos periféricos.

V _____

Variable

Entidad de memoria del tipo BOOL, WORD, DWORD, etc., cuyos contenidos se pueden modificar desde el programa durante su ejecución.

Variable no ubicada

Una variable no ubicada es una variable cuya posición en la memoria del PLC no puede conocerse. Las variables que no tienen asignadas direcciones se consideran no ubicadas.

Variable ubicada

Una variable ubicada es una variable cuya posición en la memoria del PLC se puede conocer. Por ejemplo, la variable Water_pressure se asocia a %MW102. De este modo, Water_pressure está ubicada.

W _____

WORD

El tipo WORD se codifica en formato de 16 bits y se utiliza para procesar cadenas de bits.

X _____

Y _____

Z _____

This image shows a full page of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page, typical of notebook or legal stationery. There are no margins, text, or other markings on the page.

[illegible]

This image shows a full page of white paper with horizontal blue or grey ruling lines. The lines are evenly spaced and run across the width of the page, typical of notebook paper. There are no margins, text, or other markings on the page.

[illegible]

[illegible]