

CAPÍTULO V

ASPECTOS GENERALES DEL ESTÁNDAR DE COMUNICACIÓN INDUSTRIAL DEVICENET

5.1 Introducción

La red de comunicación industrial DeviceNet es una red abierta e implementada según el modelo OSI. Posee ciertos aspectos generales que es necesario dejar en claro para el desarrollo de su estudio. Por ello es que en el presente capítulo se establecen las características de normalización, de posicionamiento y de funcionamiento de la red DeviceNet.

5.2 Antecedentes Generales

La red de campo DeviceNet es una red comunicación industrial de tipo serial que conecta a los controladores de una red con los dispositivos de entrada y salida, o I/O. Es desarrollado por la empresa Rockwell Automation como un bus de comunicación abierto y hace su aparición en 1994. Actualmente, la administración del estándar DeviceNet está a cargo de la ODVA, acrónimo de la Open DeviceNet Vendor Association (o la Asociación de Vendedores DeviceNet Abierto). La ODVA cuenta con más de 250 empresas fabricantes de dispositivos asociadas.

5.3 DeviceNet en el contexto de las estandarizaciones

La red DeviceNet está basada en el Modelo de Referencia OSI, estándar ISO/IEC 7498, que otorga una naturaleza jerárquica a la red estructurándola en 7 capas.

La forma en que las 7 capas definidas por ISO/IEC 7498 son cubiertas en DeviceNet se muestra en la figura 5.1.

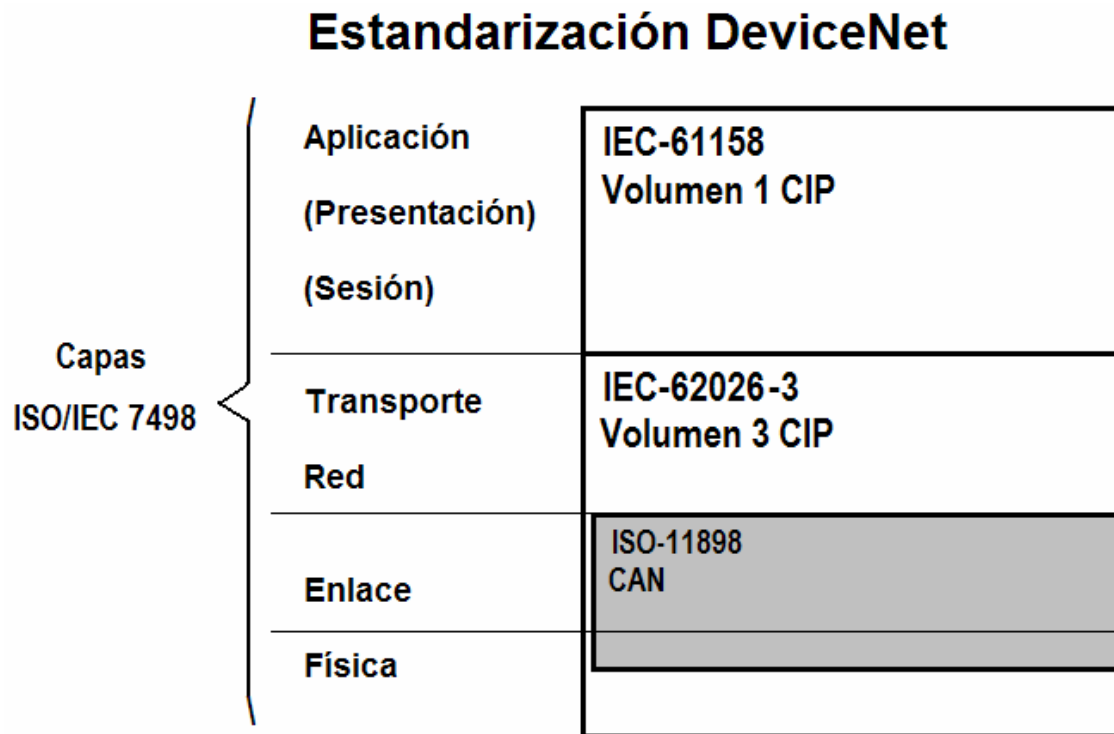


Figura 5.1. Representación de DeviceNet en el contexto de las estandarizaciones.

5.3.1 ISO 11898

El estándar ISO 11898 es la definición del protocolo de comunicación *CAN* (Controller Area Network), tanto de las versiones 2.A como 2.B del año 1991, desarrollado por la empresa BOSCH originalmente en 1984.

Éste implementa una comunicación serial para dispositivos, y como se representa en la figura 5.1 anterior, define completamente una capa de enlace y parcialmente una capa física.

Entorno a esta especificación Rockwell Automation desarrolló DeviceNet, debiendo complementar las definiciones faltantes.

5.3.2 IEC-62026-3

La norma IEC 62026-3, del año 2000, es el estándar específico para las capas #1 a #4 de la red DeviceNet. Corresponde al Volumen 3 de CIP, en donde CIP es el acrónimo de Common Industrial Protocol (o Protocolo Industrial Comun). El Volumen 3 de CIP es la adaptación de este protocolo para DeviceNet. Este volumen toma las especificaciones CAN de ISO-11898, entregando los aspectos de la capa física que esta norma no define, completando de esta forma esta capa, y adopta completamente la capa de enlace CAN sin modificarla. Por lo tanto, la norma ISO 11898 está incorporada dentro del Volumen 3 de CIP, que es lo mismo que IEC-62026-3. Luego entrega las definiciones para las capas de red y transporte.

5.3.3 IEC 61158

El estándar IEC 61158, del año 2000, es la definición para los buses de campo. Esta norma incluye los Volúmenes 1 y 4 de CIP, en la forma de IEC 61158 type 2.

El Volumen 1 de CIP especifica completamente las capas #5 a #7 del Modelo de Referencia OSI, como un protocolo común para cuatro redes industriales, que son las siguientes:

- DeviceNet
- ControlNet
- Ethernet/IP
- CompoNet

Por otra parte, el Volumen 4 de CIP define las capas #1 a #4 de la red ControlNet. Por lo tanto, IEC 61158 define completamente a ControlNet.

Por lo tanto, las capas superiores de la arquitectura de red de DeviceNet son definidas por el Volumen 1 de CIP, o que es lo mismo que una porción de IEC 61158 type 2.

5.4 DeviceNet en del contexto de la familia CIP

Como se observa en la figura 5.1, DeviceNet se compone de tres estándares internacionales. Pero dado que el protocolo CAN (ISO 11898) está contenido dentro del Volumen 3 de CIP, DeviceNet queda totalmente definida por el protocolo CIP en sus Volúmenes 1 y 3, o que es lo mismo que los estándares internacionales IEC 61158 e IEC 62026 respectivamente.

Al conjunto de redes de comunicación basadas en el protocolo CIP es llamado “Familia CIP”. La red DeviceNet fue la primera red en formar parte de esta familia.

Como se mencionó ya en la sección anterior, el Volumen 1 de CIP otorga una especificación común a las redes DeviceNet, ControlNet, Ethernet/IP y CompoNet, para las capas superiores del modelo OSI. Luego, un volumen particular de adaptación CIP para cada red en particular es definido.

La figura 5.2 representa en forma gráfica las redes que componen la familia CIP, junto a su volumen de especificación.

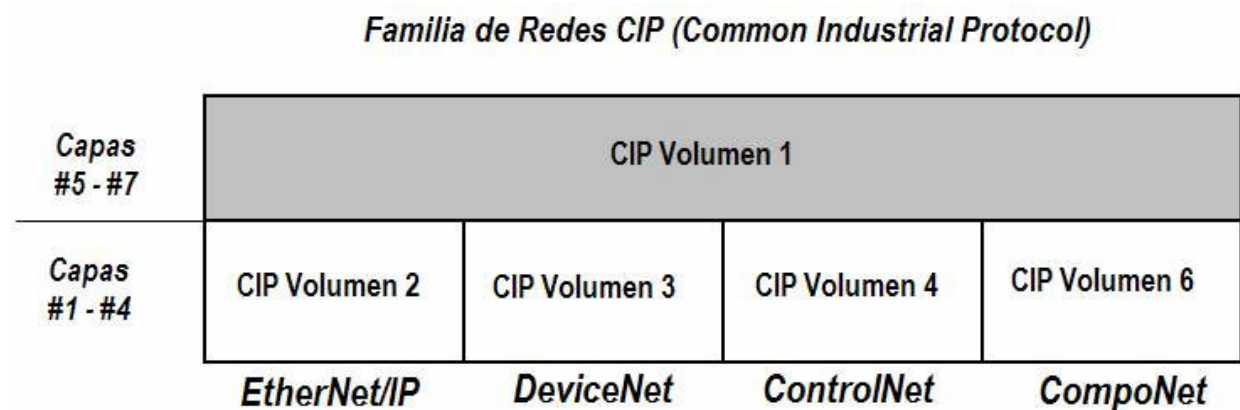


Figura 5.2. Familia de redes CIP.

5.5 Características generales de la red DeviceNet

5.5.1 Posicionamiento

La red DeviceNet se ubica entre los 2 primeros niveles de la jerarquía CIM, es decir, en los niveles de sensor/actuador y campo (y parcialmente en el nivel de control). En forma paralela, como parte de la Arquitectura NetLinx de comunicación de 3 niveles, desarrollada en conformidad al modelo CIM, impulsada por la empresa Rockwell-Automation y otras asociadas a ODVA y a ControlNet International, se ubica en su nivel de dispositivo, es decir, en su nivel 1. Esto es representado en la figura 5.3.

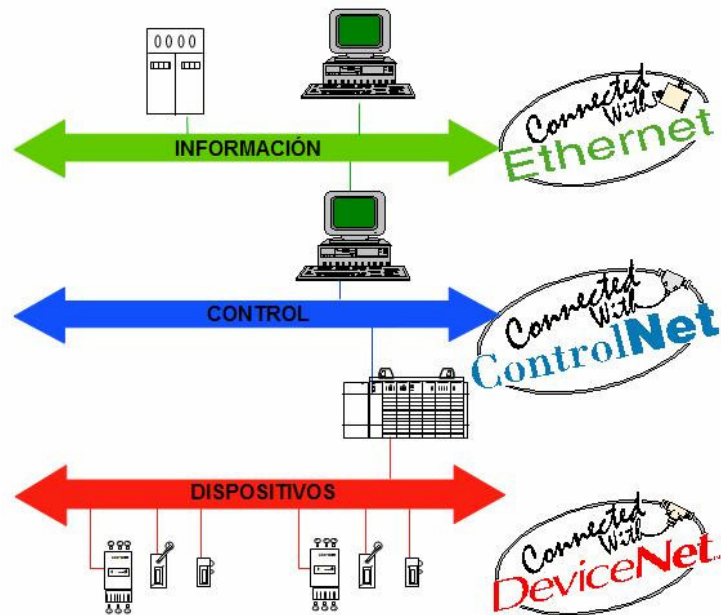


Figura 5.3. Arquitectura NetLinx de comunicación, para la integración de redes industriales.

Esta arquitectura tiene como objetivo la integración de los diferentes niveles jerárquicos de redes industriales presentes en la empresa. DeviceNet cumple su papel como la red de los dispositivos de bajo nivel, tales como sensores, botoneras y drives entre otros. También conecta dispositivos de mayor nivel tales como PLC.

5.5.2 Características de funcionamiento

El estándar de comunicación industrial DeviceNet está diseñado como un protocolo de funcionalidad media y bajo coste para el nivel inferior del bus de campo, es decir, para la conexión en red de sensores, actuadores y los controles en lazo abierto asociados.

Los tipos de dispositivo que conecta una red DeviceNet son sensores, botoneras, partidores, y PLC, entre otros.

En la figura 5.4 se muestra una típica implementación de una red DeviceNet.

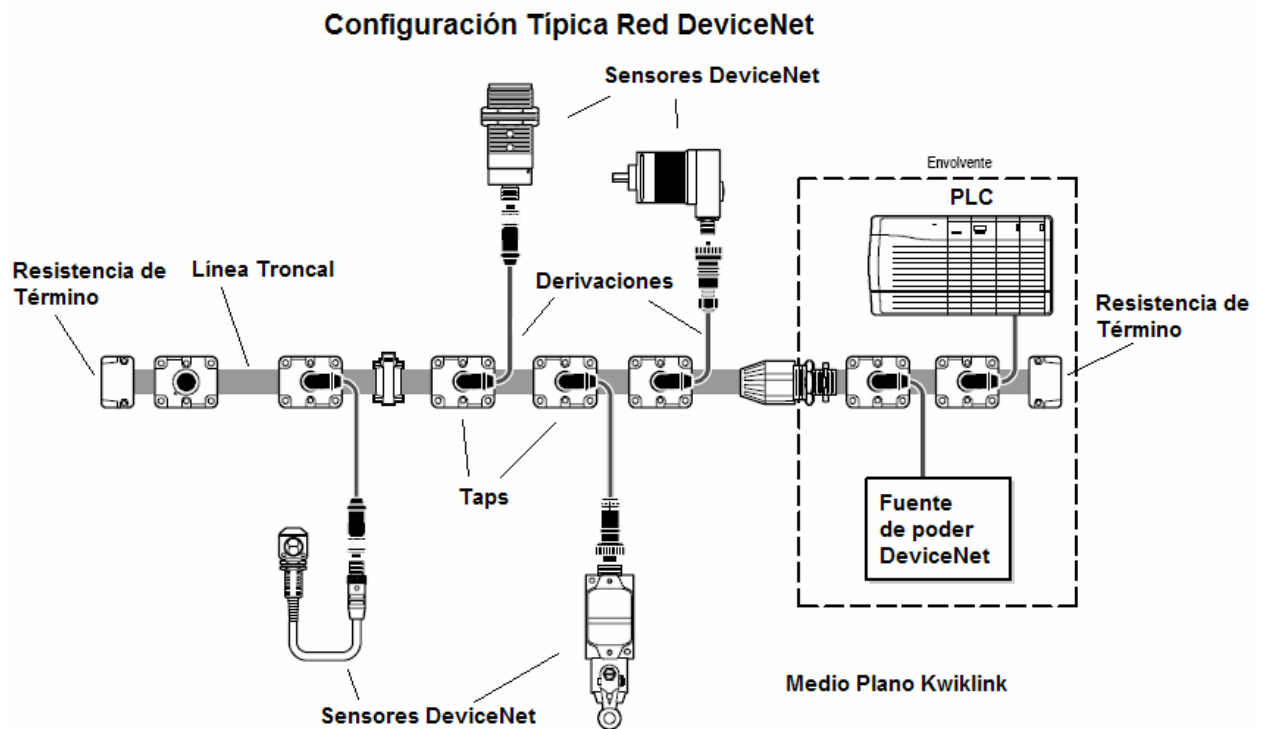


Figura 5.4. Configuración de red típica DeviceNet.

La red DeviceNet permite utilizar hasta 64 nodos con una tasa de transmisión media de 125, 250 ó 500 kbp/s. Los dispositivos pueden alimentarse a través del bus DeviceNet o disponer de su propia fuente de alimentación. En la tabla 5.1 se enlistan las principales características de la red DeviceNet.

Característica	Descripción
Topología	Línea Troncal/Derivaciones
Número de nodos	64
Longitud máxima de red	500m
Energización	24VDC sobre el mismo medio
Tasa de transmisión	125, 250 y 500 kbps (según longitud de red)
Método de Acceso al Medio	CSMA/NBA (Arbitraje de Bus)
Paquetes de Datos	0 – 8 Bytes
Método de Comunicación	Productor/Consumidor (Strobe, Poll, Cambio de estado, Cíclico)
Arquitectura de Control	Centralizada y Distribuida
Nodos Instalados (2003)	500.000 (principalmente en EE. UU.)
Otras funciones especiales	Reemplazo automático (ADR); detección automática de velocidad (Autobaud).

Tabla 5.1 Principales características de la red DeviceNet.

5.6 Test de Conformidad

La ODVA ha definido pruebas y procedimientos para la aprobación de sistemas y dispositivos DeviceNet. A través de ésta, los fabricantes de dispositivos pueden certificar sus productos en conformidad con las especificaciones del estándar DeviceNet.

La prueba de conformidad evalúan dos características que todo dispositivo debe cumplir: Interoperabilidad e Intercambiabilidad.

- **La Interoperabilidad** se refiere a que todo dispositivo DeviceNet, independientemente del fabricante, tiene la capacidad de ser configurado para operar dentro de una red DeviceNet.
- **La Intercambiabilidad** se refiere a que todo dispositivo que sigue un mismo perfil es capaz de ser sustituido por otro, independientemente del fabricante, que cumple con el mismo perfil.

A su vez, la prueba de conformidad dividida en tres partes: Prueba de Software, Prueba de Hardware y Prueba de Interoperabilidad de Sistema.

5.6.1 Prueba de Software

Verifica las funciones del protocolo DeviceNet. Dependiendo de la complejidad del dispositivo, se transmiten cientos al Dispositivo Bajo Testeo (DUT por el inglés “Device-Under-Test”). Las pruebas son adaptadas al DUT, de modo que el fabricante debe entregar una descripción formal de todas las características relevantes del DUT.

5.6.2 Prueba de Hardware

Examina la conformidad del dispositivo con las especificaciones de la capa física. Esta prueba chequea todos los requerimientos de la especificación (por ejemplo, niveles de bits, conectores, sobrevoltaje soportado, tierra y chip CAN).

5.6.3 Prueba de Interoperabilidad del Sistema

Ésta verifica que un dispositivo puede funcionar en una red con más de 60 nodos y con una variedad de Scanners de varios fabricantes.

Cuando un dispositivo ha sido aprobado en conformidad con las especificaciones DeviceNet, éste llevará un sello con la leyenda “DeviceNet Conformance Tested” (que puede traducirse al español como “Testeado en Conformidad DeviceNet”). Éstos dispositivos son publicados en el sitio web de la ODVA. En la figura 5.5 se muestra el logo de certificación.



Figura 5.5. Logo para dispositivos DeviceNet certificados.

5.7 Herramientas para la Red DeviceNet

Las herramientas para redes DeviceNet pueden dividirse en 3 grupos:

- **Herramientas para la Capa Física**
- **Herramientas de Configuración**
- **Herramientas de Monitores**

5.7.1 Herramientas para la Capa Física

Son herramientas en hardware y/o software que comprueban la integridad y cumplimiento de las especificaciones para la capa física y revisa la calidad en la transmisión de datos.

5.7.2 Herramientas de Configuración

Son herramientas software capaces de comunicarse individualmente con dispositivos para monitoreo de datos y propósitos de configuración. Estos pueden ser desde simples software operativos sobre equipos manuales hasta poderosos paquetes de software para PC para la configuración de una red completa. La mayoría de las herramientas de configuración están basadas en los Electronic Data Sheets, EDS (Hojas de Datos Electrónicas, ver sección 4.10 y anexos 1 a 4).

5.7.3 Herramientas de Monitoreo

Por lo general son paquetes de software para PC que pueden capturar y mostrar las tramas CAN. Estas pueden mostrar tanto una trama CAN pura, así como su interpretación DeviceNet.

Para una aplicación típica, se requiere sólo de una herramienta de configuración. Sin embargo, para asegurar que la red está operando apropiadamente, se recomienda el uso de herramientas de verificación de la capa física. La experiencia muestra que la gran mayoría de los problemas en una red DeviceNet son causados por una inapropiada instalación de la capa física. Las herramientas para el monitoreo de protocolo son usados principalmente para investigar problemas de interoperabilidad y asistir durante el proceso de desarrollo.

CAPÍTULO VI

ARQUITECTURA DE LA RED DE CAMPO DEVICENET

6.1 Introducción

Como se estudio en el capítulo anterior, la red de campo DeviceNet es una red de comunicación basada en el modelo de referencia ISO/OSI de capas. Por lo tanto, la especificación para DeviceNet entrega las definiciones para las distintas capas. En la figura 6.1 se muestra la composición de la arquitectura de red DeviceNet.

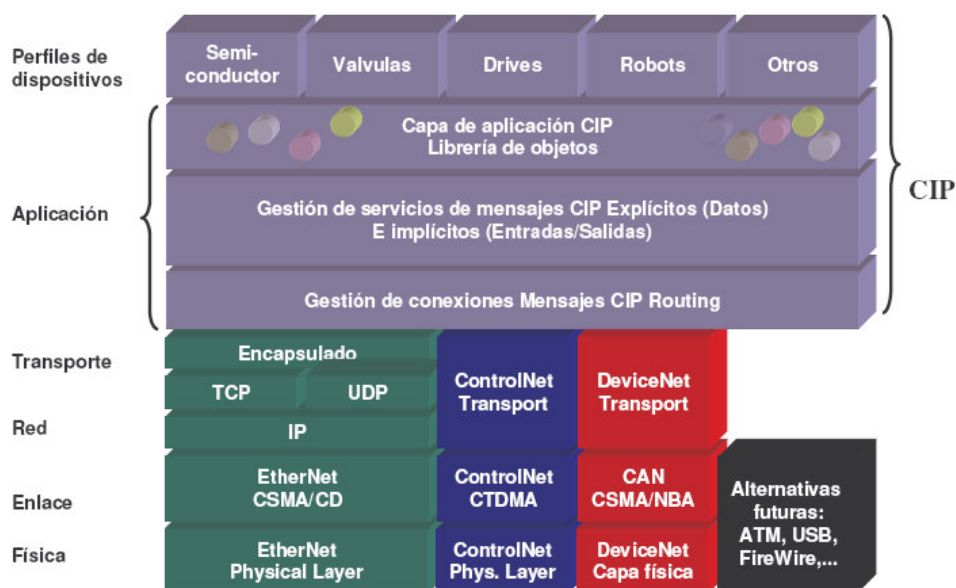


Figura 6.1. Arquitectura de red DeviceNet.

Las capas física, enlace, red y transporte son definidas bajo el Volumen 3 de CIP, el cual ha sido estandarizado como IEC-62026-3. Por otra parte, las capas de aplicación (sesión, presentación y aplicación) son definidas bajo el Volumen 1 de CIP, el cual también ha sido estandarizado como parte del IEC-61158 type 2.

En el presente capítulo se estudian las definiciones de cada capa, dadas por las especificaciones DeviceNet, entregando una descripción de las principales características.

6.2 La Capa Física de la Red de Campo DeviceNet

6.2.1 Topología

Como se estudió en el capítulo anterior, DeviceNet implementa el protocolo CAN. Por ello es que en muchas ocasiones, se hace referencia a CAN, cuando se describen características de DeviceNet. La topología básica de una implementación CAN es la *bus* o *línea troncal*. En el caso de DeviceNet, esta define una topología Trunkline-Dropline, o Línea Troncal con Derivaciones, que corresponde al bus de datos principal (Trunkline) de la que se cuelgan las distintas derivaciones (Dropline). En la figura 6.2 se muestra esquemáticamente la topología posible con DeviceNet.

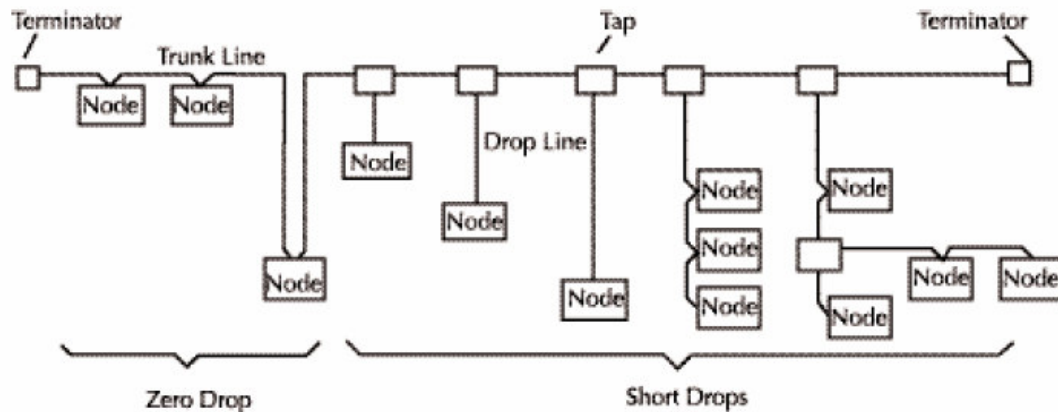


Figura 6.2. Topología Línea-Troncal/Derivaciones de DeviceNet.

6.2.2 Interdependencia de la velocidad y la longitud de la red

En la red DeviceNet existe una interdependencia entre la longitud de la red y la tasa de transmisión. Esta es debida al retardo de propagación de señal. El retardo de propagación de señal es determinado por los 2 nodos que están más alejados el uno del otro dentro del sistema. Es el tiempo que se toma la señal para viajar desde un nodo a otro que sea el más lejano (tomando en cuenta el retardo causado por la transmisión y recepción del nodo), sincronización y la señal desde el segundo nodo que viaja de regreso al primero.

Debido a esto, la relación que existe entre la longitud y la tasa de transmisión es inversamente proporcional, es decir, a mayor longitud de red, menor es la tasa de transmisión posible. Además, las longitudes de las derivaciones también afectarán la tasa de transmisión máxima.

La especificación DeviceNet, define 3 velocidades para determinados rangos de longitud de la red. En la tabla 6.1 se muestran las velocidades y longitudes para DeviceNet, según el medio físico con el que se haya implementado.

Tasa de Transmisión	Longitud Línea Troncal			Longitud Derivación	
	Cable Thick	Cable Thin	Cable Plano KwikLink	Máxima	Acumulada
125 Kbps	500 m	100 m	420 m	6 m	156 m
250 Kbps	250 m		200 m		78 m
500 Kbps	100 m		75 m		39 m

Tabla 6.1. Tasas de transmisión definidas por DeviceNet y distancias máximas según el medio físico.

6.2.3 Niveles del Bus

La especificación CAN define dos estados lógicos:

- Recesivo (1 lógico), y
- Dominante (0 lógico).

Por lo tanto, la transmisión de mensajes CAN y de la competencia por acceso al bus se basa en la habilidad para representar un valor de bit en términos de “Dominante” y “Recesivo”. Esto debe ser realizado por medios eléctricos.

Una transmisión CAN se compone de dos señales denominadas CAN_H (CAN High o CAN Alto) y CAN_L (CAN Low o CAN Bajo). Estas se transmiten simultáneamente cada una por un conductor distinto. Estas operan en modo diferencial, lo que significa que portan voltajes invertidos, para disminuir la interferencia de ruido. Los niveles de voltajes dependen del estándar utilizado. En la figura 6.3 se muestra un esquema que representa esta situación.

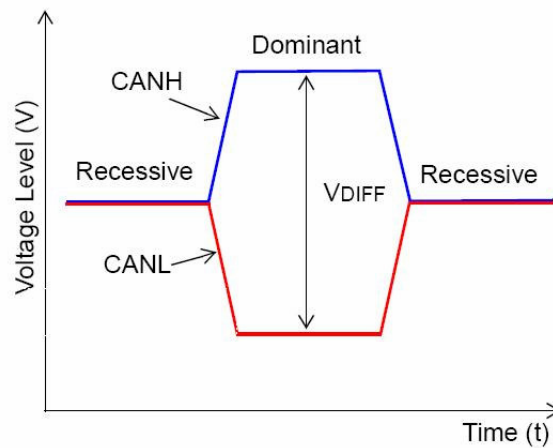


Figura 6.3. Niveles de Voltajes CAN utilizados para representar un bit dominante o uno recesivo.

Como se muestra en la figura 6.3 para la representación de un bit “Recesivo” (1 lógico) los niveles de voltaje de las señales CAN_H y CAN_L es el mismo; y para la representación de un bit “Dominante” (0 lógico) existe una diferencia de voltaje, V_{DIFF} , en donde el nivel de voltaje de CAN_H es mayor que el de CAN_L.

DeviceNet utiliza los valores definidos por ISO 11898-2, adaptada en IEC 62026-3, para los niveles de voltaje de estas señales. En la tabla 6.2 se describen los niveles de voltaje definidos por esta especificación.

Señal	Recesivo			Dominante			Unidad
	Mín.	Nominal	Máx.	Mín.	Nominal	Máx.	
CAN-High	2,0	2,5	3,0	2,75	3,5	4,5	Volt
CAN-Low	2,0	2,5	3,0	0,5	1,5	2,25	Volt

Tabla 6.2. Niveles de voltaje definidos para la señal CAN en DeviceNet

La figura 6.4 muestra esquemáticamente los valores de voltaje nominal en el bus mencionados en la tabla 6.2.

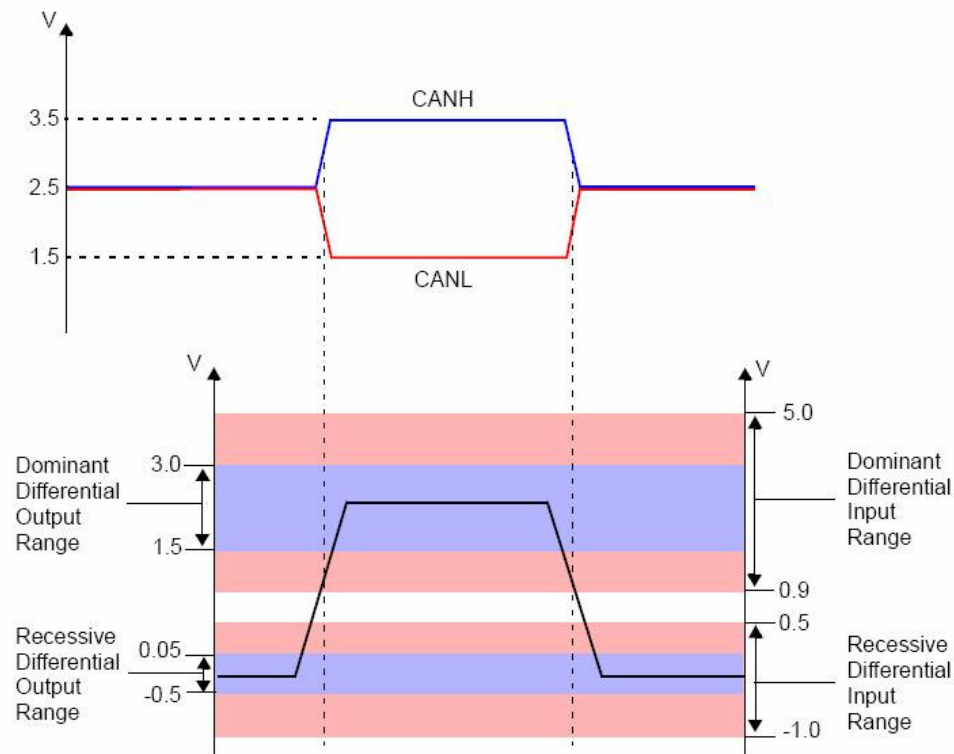


Figura 6.4. Representación esquemática de los valores nominales de voltaje de las señales del bus CAN definidos para DeviceNet.

6.2.4 Resistencia de Término

Para que un nodo pueda leer correctamente el nivel del bus es importante evitar las reflexiones de la señal en los extremos del bus. Esto se hace colocando una resistencia de término (TR) en ambos extremos del bus y evitando segmentos de término de bus innecesariamente largos.

La especificación DeviceNet define un valor de 121Ω para la resistencia de término y debe conectarse una en cada extremo del bus entre CAN_H y CAN_L.

DeviceNet define la construcción física de la resistencia de término según el tipo de cable y conector utilizado.

6.2.5 Codificación de Bit

El método de codificación de bit (bit-encoding) utilizado por DeviceNet es Non-Return-to-Zero, NRZ (no retorno a cero). Según esta codificación de bit, el nivel de señal permanece constante durante el bit-time (tiempo de bit) y así simplemente un tiempo de slot (time slot) es necesario para representar un bit (otros métodos de codificación de bit son e. g. Manchester o Modulación por Ancho de Pulso, PWM). El nivel de señal puede permanecer constante durante un periodo largo de tiempo; entonces es necesario tomar medidas para asegurar que el máximo intervalo permitido entre dos bordes de señal no sea excedido. Esto es importante para los propósitos de sincronización. Un bit de relleno (bit-stuffing) es insertando luego de 5 bits de igual valor. El receptor identifica los bits de relleno y procesa el contenido del dato original.

En la figura 6.5 se muestra el método de codificación de bit NRZ, y se compara con el método Manchester.

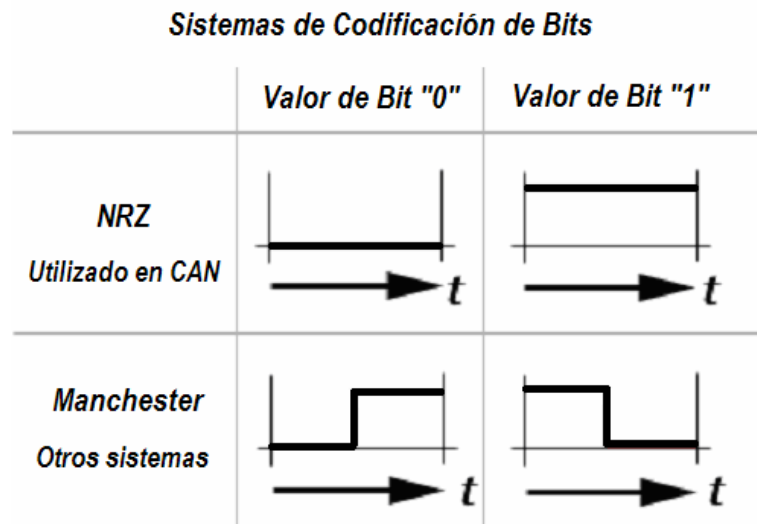


Figura 6.5. DeviceNet utiliza bit-encoding NRZ

6.2.6 Bit-Timing y Sincronización

CAN es un protocolo de transmisión síncrona. Por ello, para que el receptor esté habilitado para recibir un mensaje, se requiere de continuas resincronizaciones. Para ello, se utiliza el bit-time, o tiempo de bit, el cual se subdivide en 4 segmentos. En la figura 6.6 se representa el bit-time nominal CAN

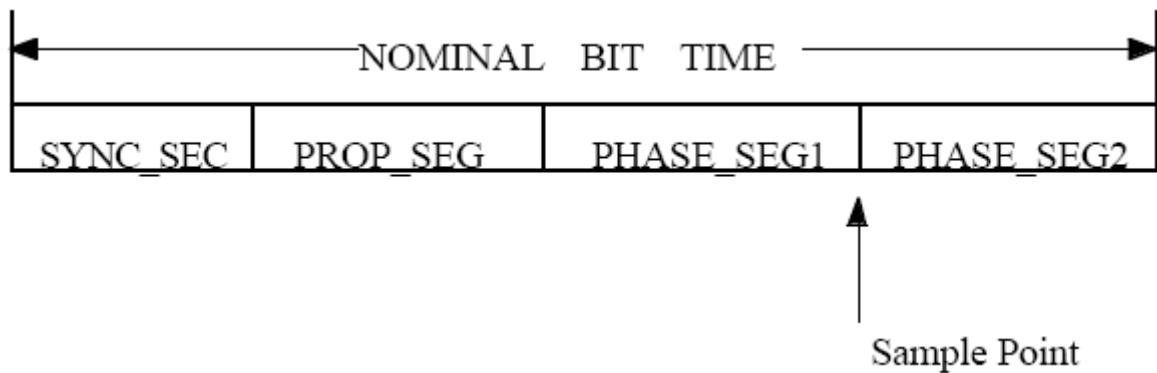


Figura 6.6. Bit-time nominal CAN y sus segmentos.

De la figura 6.6 se desprenden las siguientes definiciones:

- **Segmento de Sincronización (SYNC_SEG):** Es usada para sincronizar los nodos conectados en el bus. Se espera que un borde quede dentro de este segmento.
- **Segmento de Retardo de Propagación (PROP_SEG):** Es usado para compensar los retardos debidos a la propagación de la señal en el medio físico dentro de la red. Considera los recorridos de ida y vuelta de la señal entre un nodo transmisor y un nodo receptor.
- **Segmento de Fase 1 (PHASE_SEG1) y Segmento de Fase 2 (PHASE_SEG2):** Son usados para compensar errores de fase de borde. Estos segmentos de tiempo pueden ser alargados o acortados para lograr la resincronización.
- **Punto de Muestreo (Simple Point):** Es el momento en que el nivel del bus es leído e interpretado como un valor respectivo de bit. Se ubica al final del PHASE_SEG1.

A través de la subdivisión del bit-time en los segmentos de tiempos, se logra los propósitos de sincronización y las sucesivas resincronizaciones. Entonces, se distinguen dos tipos de sincronización: Sincronización Dura (Hard) y Resincronización por salto de anchura.

- **En la Sincronización Dura (Hard)**, el bit-time es reiniciado al término segmento de sincronización (SYNC_SEG). Por consiguiente, el borde que causó la sincronización dura, queda dentro del SYNC_SEG del bit-time reiniciado.
- **En la Resincronización por salto de anchura**, los segmentos de fase son acortados o alargados, para cambiar el punto de muestreo. En este caso, el PHASE_SEG1 es alargado, y el PHASE_SEG2 es acortado dentro del bit-time. Por tanto, estos segmentos parten con un valor inicial por sobre el cual se alargan o acortan..

6.2.7 Especificaciones del medio físico

a) Cables

La especificación DeviceNet define 3 tipos de cables: Round-Thick, Round-Thin y Flat (o plano). Los medios Round tienen 5 hilos conductores, mientras que el medio Plano tiene 4. En la figura 6.7 se muestran los hilos conductores definidos en DeviceNet.

Wire Color	Wire Identity	Usage Round	Usage Flat
white	CAN_H	signal	signal
blue	CAN_L	signal	signal
bare	drain	shield	n/a
black	V-	power	power
red	V+	power	power

Figura 6.7. Hilos de los cables especificados en DeviceNet.

Pueden observarse los hilos conductores para las señales CAN_H y CAN_L. También puede notarse que se incluye un par de hilos conductores correspondientes a V- y V+, por los cuales se entrega la energía a los distintos dispositivos conectados al bus. Estos pares están presentes en los 3 tipos de cable. Además, se define un quinto hilo conductor para los medios tipo Round (Thick y Thin) para protección eléctrica, drain.

A continuación se da una descripción de los tipos de cables definidos por DeviceNet.

a.1) Round Thick: éste cable es utilizado principalmente para la línea troncal (trunk-line) de la red DeviceNet (también puede ser utilizado para las línea de derivación o drop-line). Posee un diámetro de 12,2 mm (0,48 in). En la figura 6.8 se muestra la construcción física de este cable.

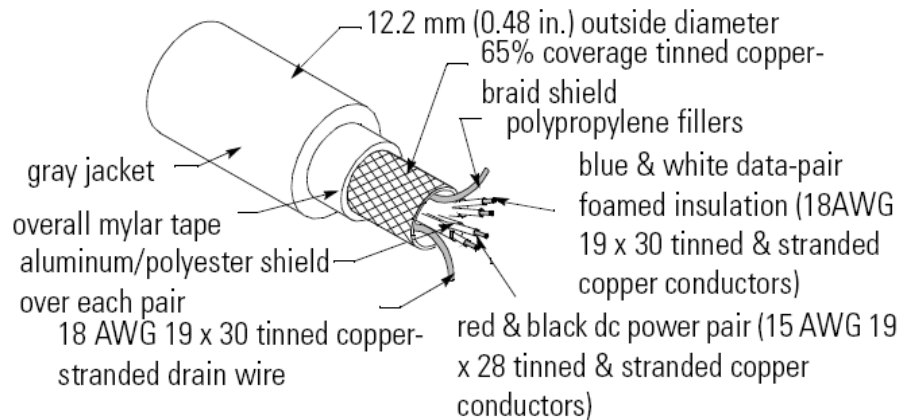


Figura 6.8. Construcción física del cable DeviceNet tipo Round Thick.

a.2) Round Thin: éste cable es utilizado principalmente para las líneas de derivación (drop-line) de la red DeviceNet (también puede ser utilizado para la línea troncal o trunk-line). Posee un diámetro de 6,9 mm (0,27 in). El diámetro menor de éste cable, le otorga mayor flexibilidad. Puede observarse en la figura 6.9 la construcción física de este cable.

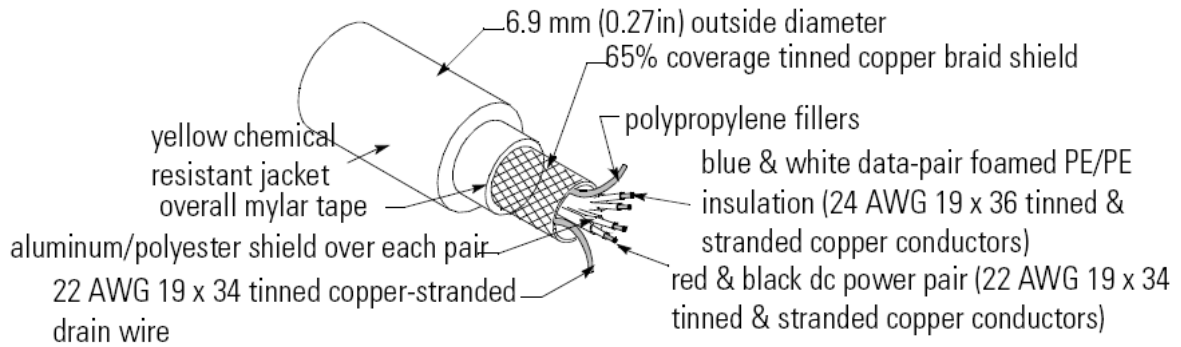


Figura 6.9. Construcción física del cable DeviceNet tipo Round Thin.

a.3) Flat (sistema KwikLink): éste cable es utilizado exclusivamente para la línea troncal (trunk-line) de la red DeviceNet. Los sistemas que utilizan este tipo de cable se denominan KwikLink. La dimensión de este cable es de 19,3 mm x 5,3 mm. No posee una longitud predeterminada. Se distinguen dos clases de cable tipo Flat:

- *Clase 1:* puede soportar hasta 8 A.
- *Clase 2:* puede soportar hasta 4 A.

En la figura 6.10 se muestran las especificaciones para el medio plano KwikLink.

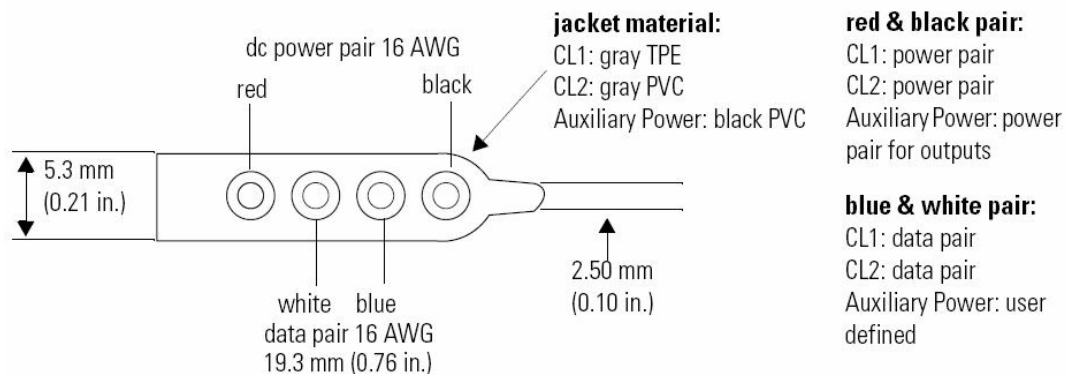


Figura 6.10. Construcción física del cable DeviceNet tipo Flat.

También, un sistema KwikLink permite el uso de un *cable para derivación sin protección (Unshielded-drop-cable)* el cual posee los 4 hilos conductores y posee un diámetro especificado por el fabricante.

b) Conectores

Los conectores DeviceNet incorporan conexión para los 5 hilos conductores definidos anteriormente. Los tipos definidos por DeviceNet se detallan a continuación.

Los conectores se dividen en dos categorías: sellados y abiertos.

b.1) Conectores sellados: estos conectores reciben ésta denominación debido a que los hilos conductores van sellados por una carcasa tipo plug. Dentro de estos conectores se definen los siguientes tipos: Mini-style y Micro-Style.

- **Mini-Style:** conector utilizado para Taps y cables tipo Round-Thick y Round-Thin.
- **Micro-Style:** conector utilizado sólo en cables Round-Thin (tiene una reducción de corriente).

En la figura 6.11 se muestra la construcción física especificada para los conectores sellados.

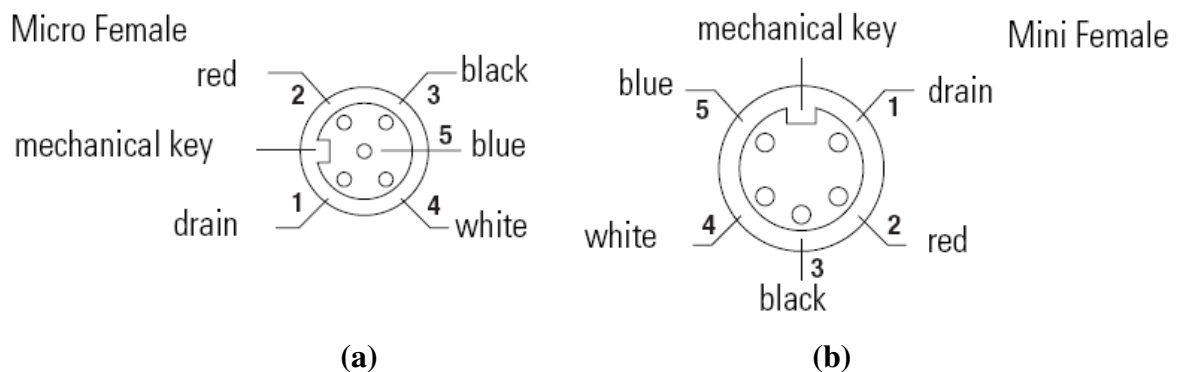


Figura 6.11. Conectores sellados DeviceNet: (a) Micro-Style, y (b) Mini-Style.

b.2) Conectores abiertos: estos conectores se denominan de esta manera debido a que los pines dejan expuestos los hilos conductores de la red. Dentro de estos conectores se definen los siguientes tipos: Plug-in y Fixed.

- **Plug-in:** utilizado para conectar los hilos conductores en un conector removible. Se definen de 5 y 10 pines (2 líneas de 5 pines en paralelo). Las dimensiones de éste conector removible también son definidas en la especificación.
- **Fixed:** utilizado para conectar los hilos conductores directamente en un terminal fijo de tornillos (regleta de conexión) de un dispositivo.

En la figura 6.12 se muestra la construcción física de conectores abiertos.

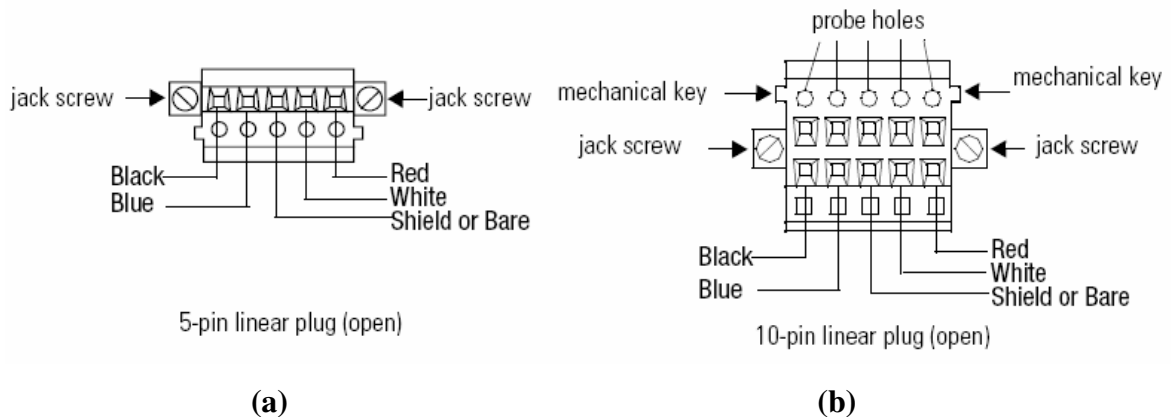


Figura 6.12. Tipos de conectores definidos en DeviceNet: (a) Conector abierto plug-in de 5 pines; (b) Conector abierto plug-in de 10 pines.

En la figura 6.13 puede observarse la utilización típica de este tipo de conectores, junto a un conector removible para una conexión a un PC.

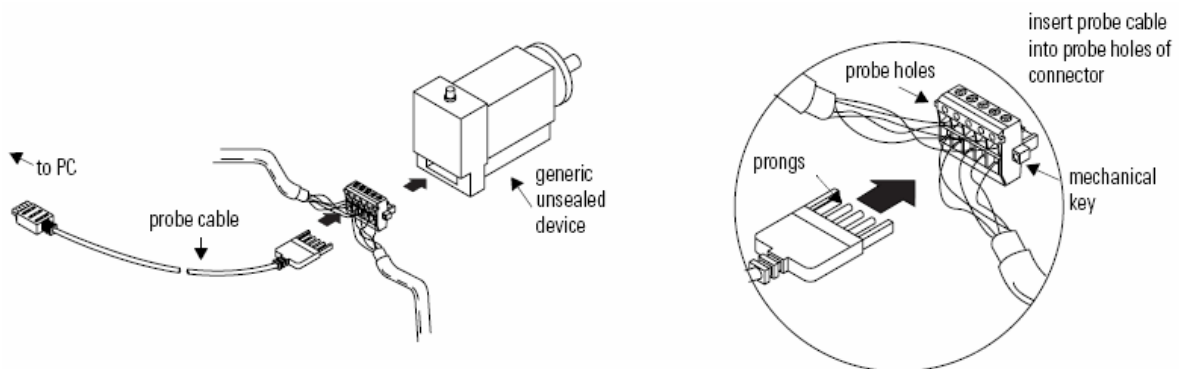


Figura 6.13. Utilización del conector abierto plug-in.

Además, se definen adaptadores de conector abierto a conector sellado. En la figura 6.14 se muestra un adaptador para un conector sellado Micro-Style.

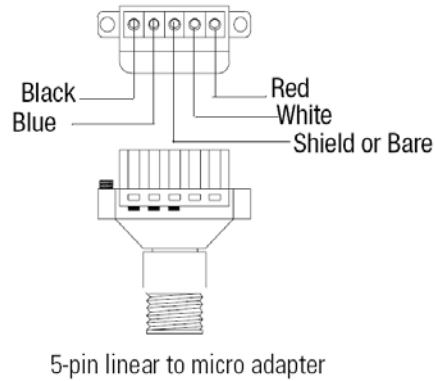


Figura 6.14. Adaptador de conector abierto a conector sellado DeviceNet.

c) Taps DeviceNet

Los taps son los puntos de la Línea Troncal (trunk-line) que cumplen alguna de las siguientes funciones:

- Puntos desde donde se conectan los nodos a la red.
- Puntos desde donde se inician las derivaciones (drop-lines).
- Puntos por los cuales se entrega la energía a la red.

La especificación DeviceNet define las características físicas de los Taps. Para cada tipo de cable son aplicables distintos tipos de Taps.

Los Taps definidos para DeviceNet son los siguientes:

c.1) T-Port : Permite la conexión de un dispositivo directamente a la línea troncal. Se utiliza sólo en los medios Round (tanto thick como thin). En la figura 6.15 se muestra la construcción física de un T-Port tipo Mini.

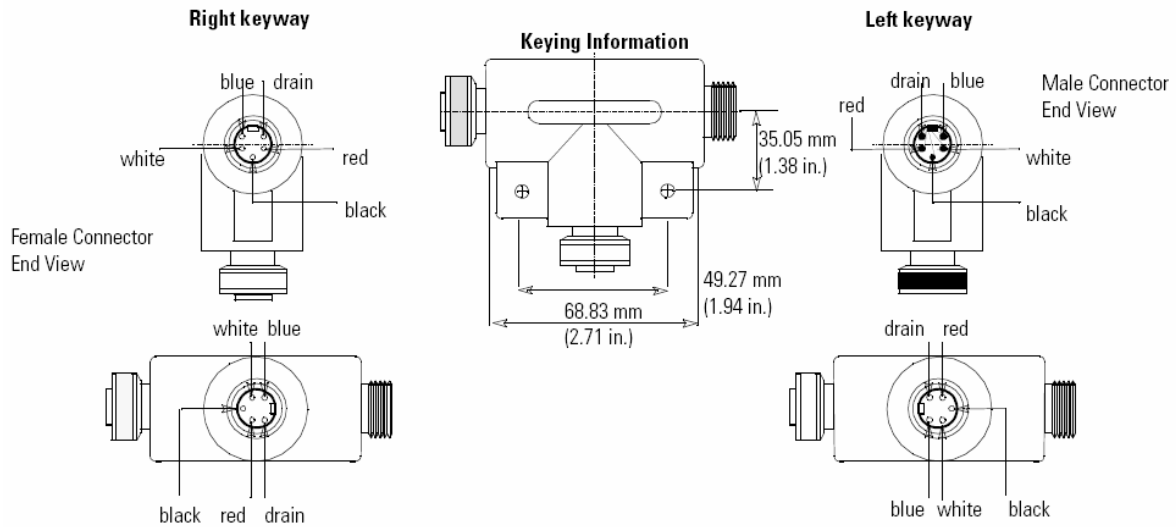


Figura 6.15. T-Port tipo Mini para cable Round-Thin DeviceNet.

c.2) DeviceBox : Caja de derivación de dispositivos para conectores sellados. Permite la conexión de hasta 8 dispositivos (o nodos) directamente a la línea troncal. En la figura 6.16 se observa la construcción física de los DeviceBox Taps.

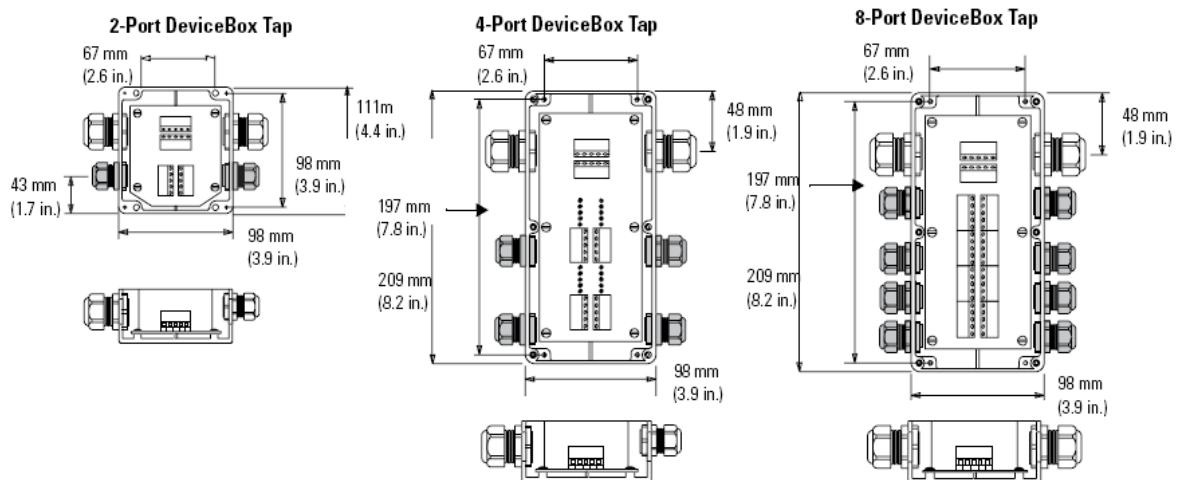


Figura 6.16. Los DeviceBox Taps permiten la conexión de dispositivos directamente a la línea troncal.

c.3) PowerTap : Permite la conexión de un fuente de poder de 24VDC para la alimentación de la red y protege contra sobrecorrientes. Se utiliza sólo en medios Round. Se puede observar en la figura 6.17 la construcción física y esquema de conexión de este Tap.

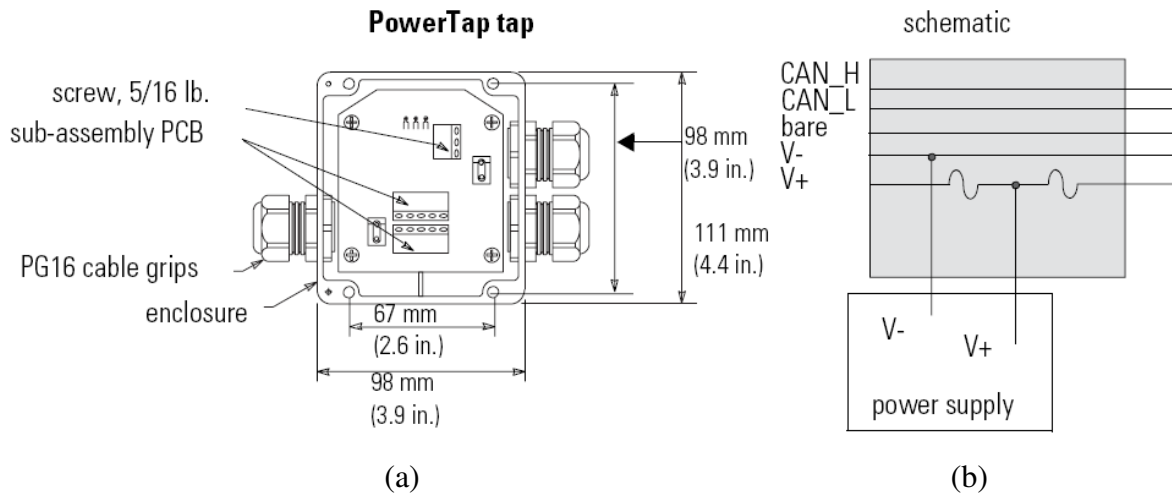


Figura 6.17. PowerTap definido para DeviceNet: (a) construcción física; (b) Esquema de conexión.

c.4) DevicePort : Caja multipuerto que permite la conexión a la línea troncal a través de una línea de derivación. Se distinguen 3 tipos de DevicePort: Micro, Mini y Thru-trunk. Permite la conexión de hasta 8 dispositivos (o nodos) a una línea de derivación. Se utiliza tanto para medios Round como Flat.

Micro y Mini DevicePort: Estos tipos de DevicePort permiten la conexión de hasta 8 dispositivos a una línea de derivación. El tipo Micro es usado en un medio Round-Thick mientras que un tipo Mini es utilizado en un medio Round-Thin. En las figuras 6.18 y 6.19 se muestran la construcción física de un DevicePort tipo Micro y de un DevicePort tipo Mini respectivamente.

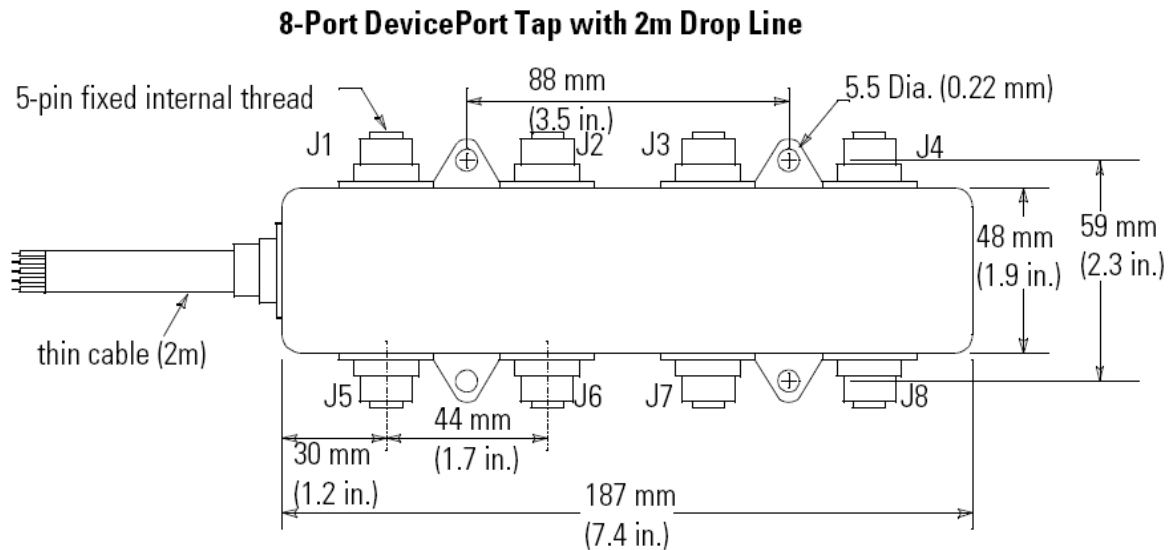


Figura 6.18. Construcción física de un Micro DevicePort DeviceNet de 8 puertos.

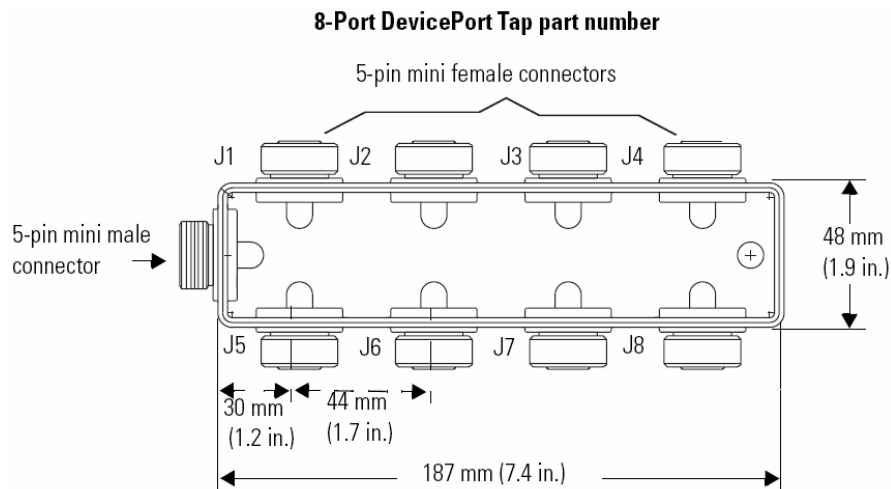


Figura 6.19. Construcción física de un Mini DevicePort DeviceNet de 8 puertos.

Thru trunk DevicePort: Caja multipuerto. A diferencia de los otros DevicePort, este es un dispositivo pasivo que permite la conexión de hasta 8 dispositivos (o nodos) directamente a la línea troncal. En la figura 6.20 puede apreciarse la construcción física de este tipo de DevicePort.

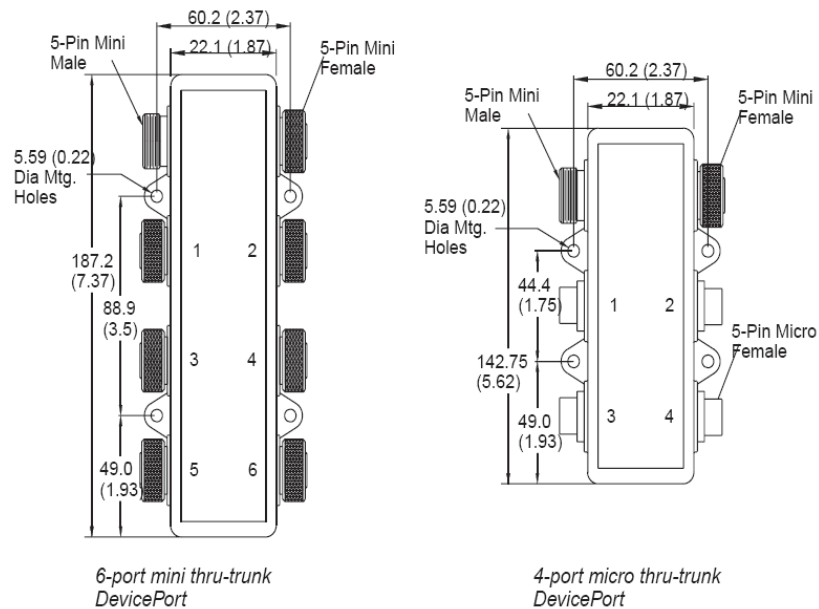


Figura 6.20. DevicePort tipo Thru trunk.

c.5) Taps open-style: o tap de estilo abierto. Estos taps dejan expuestos los hilos conductores de la red de manera que es posible conectarse a través de un conector removible o bien conectando los cables directamente a una regleta. Sólo se utilizan en sistemas KwikLink. Los tipos de estos taps son: Open-Style-Connector, Open-Style, KwikLink open-style-connector y KwikLink micro-connector. En las figuras 6.21 y 6.22 se muestran distintos Taps de estilo abierto.

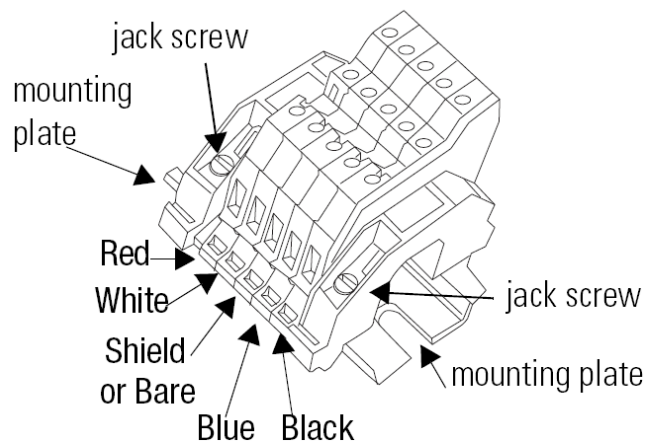


Figura 6.21. Tap de estilo abierto utilizado en medios Flat.

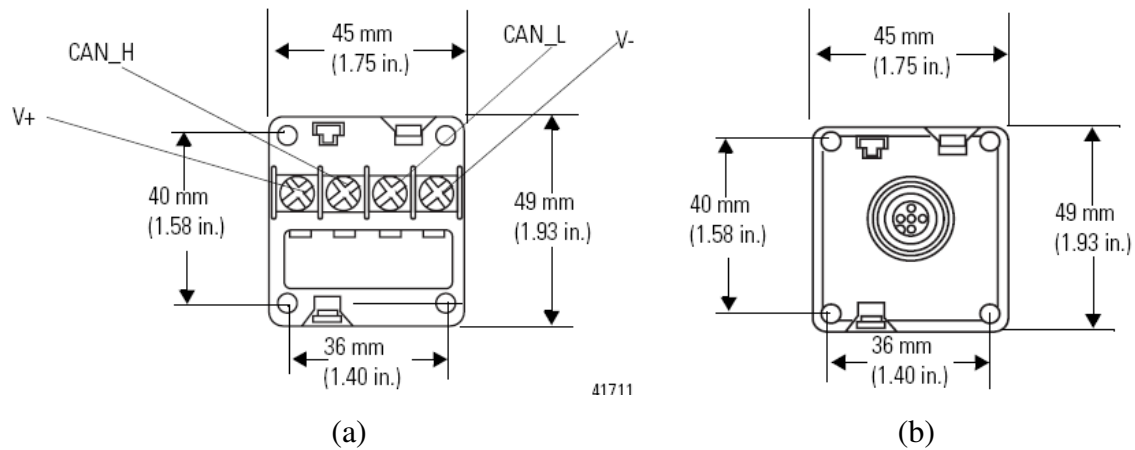


Figura 6.22. Taps KwikLink: (a) Tap KwikLink con conector de estilo abierto (con regleta), (b) Tap KwikLink con conector tipo sellado.

d) Construcción física de la Resistencia de Término

d.1) Para cable Round: Resistencia tipo sellada (Sealed) para extremos con tap de derivación tipo T-Port sellado. En la figura 6.23 se muestra la construcción física de esta resistencia de término.

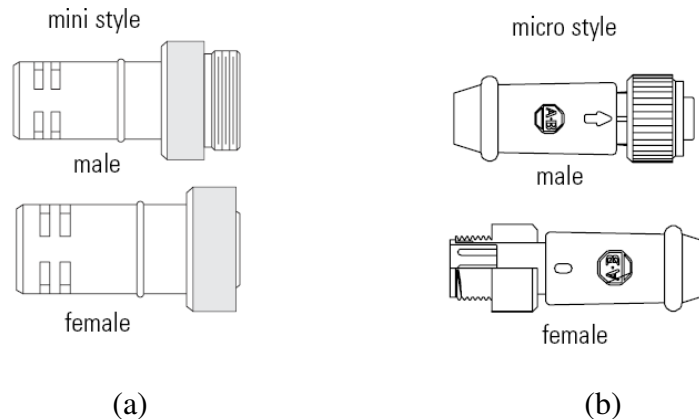


Figura 6.23. Construcción física de la resistencia de término tipo sellada (a) Mini y (b) Micro.

d.2) Para cable Flat: Se definen las resistencias de término tanto en versiones sellada como abierta. En las figuras 6.24 y 6.25 se muestran estas alternativas respectivamente.

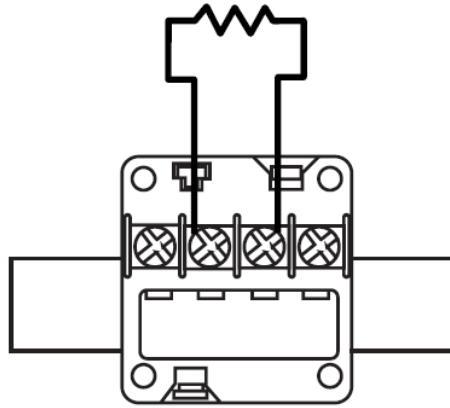


Figura 6.24. Resistencia de término open-style para un sistema *flat*.

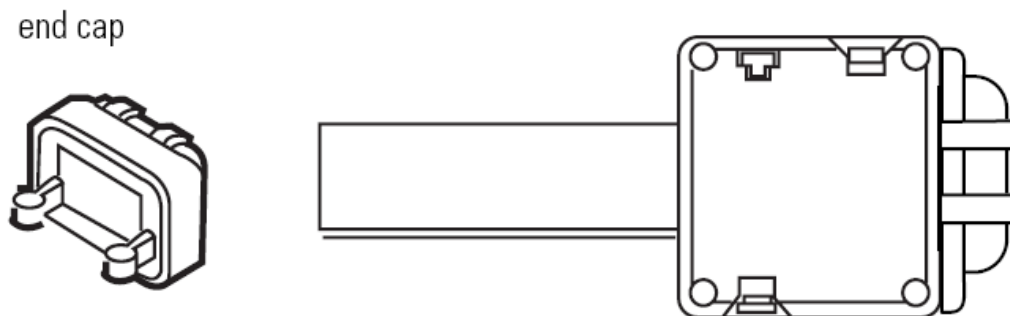


Figura 6.25. Resistencia de término sellado para un sistema *flat*.

e) Fuente de Poder

DeviceNet utiliza una fuente de +24VDC y permite la conexión de fuentes redundantes. La energización se entrega a los dispositivos a través del mismo bus a través del par de hilos V+ y V-. Además es posible insertar o desconectar nodos sin tener que desconectar la energía de la red.

En la figura 6.26 se muestra la conexión esquemática de una fuente de energía en un medio Round, tanto Thick como Thin. Puede observarse la conexión de la fuente a V+ y V- además de la conexión de la tierra de protección.

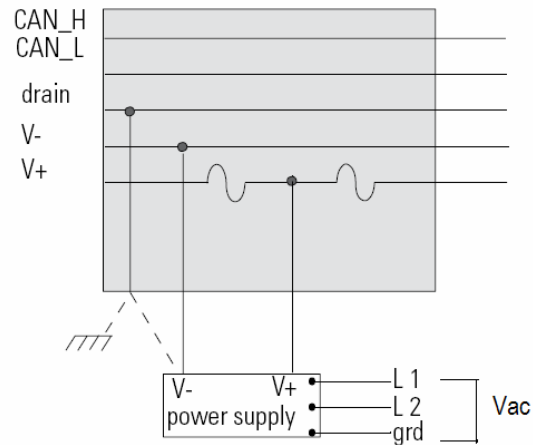


Figura 6.26. Esquema de conexión de una fuente de poder en un medio tipo Round.

En la figura 6.27 se muestra el esquema de conexión de una fuente para un medio tipo Flat (KwikLink). En esta figura puede observarse la conexión de V+ y V-, pero el medio tipo Flat no va conectado a la tierra de protección, pero sí lo está la fuente.

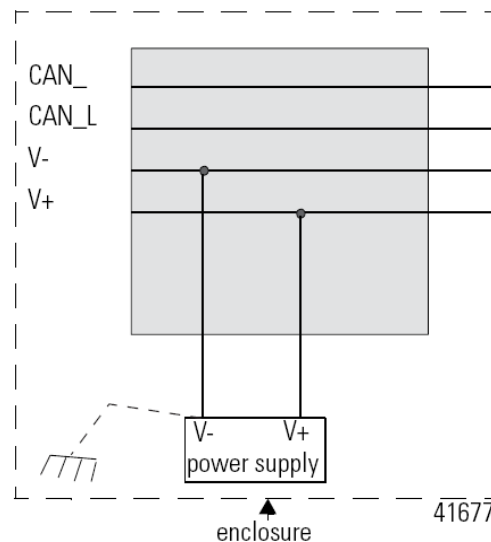


Figura 6.27. Esquema de conexión de una fuente de poder en un medio Flat (KwikLink).

e.1) Uso de Fuentes Redundantes

En el caso de usar fuentes redundantes, se deben seguir las pautas de conexión dadas en la especificación DeviceNet.

En la figura 6.28 donde se muestra el esquema de conexión para una red DeviceNet con una fuente redundante en un medio Round.

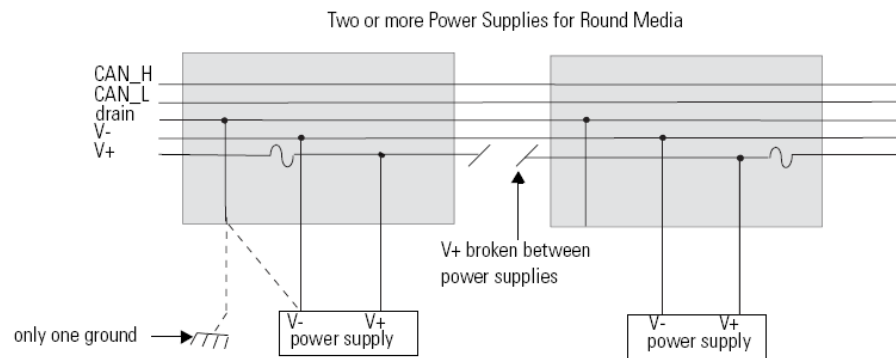


Figura 6.28. Conexión de fuentes redundantes en un medio Round.

En la figura 6.29 se muestra esta misma situación, esta vez, para un medio Flat.

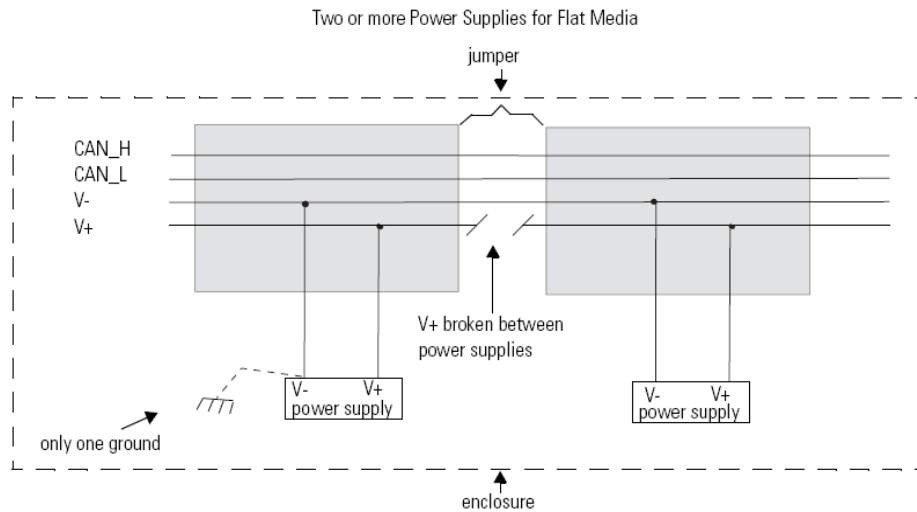


Figura 6.29. Conexión de fuentes redundantes en un medio Flat.

6.3 La Capa de Enlace de DeviceNet

6.3.1 Método de Acceso al Medio

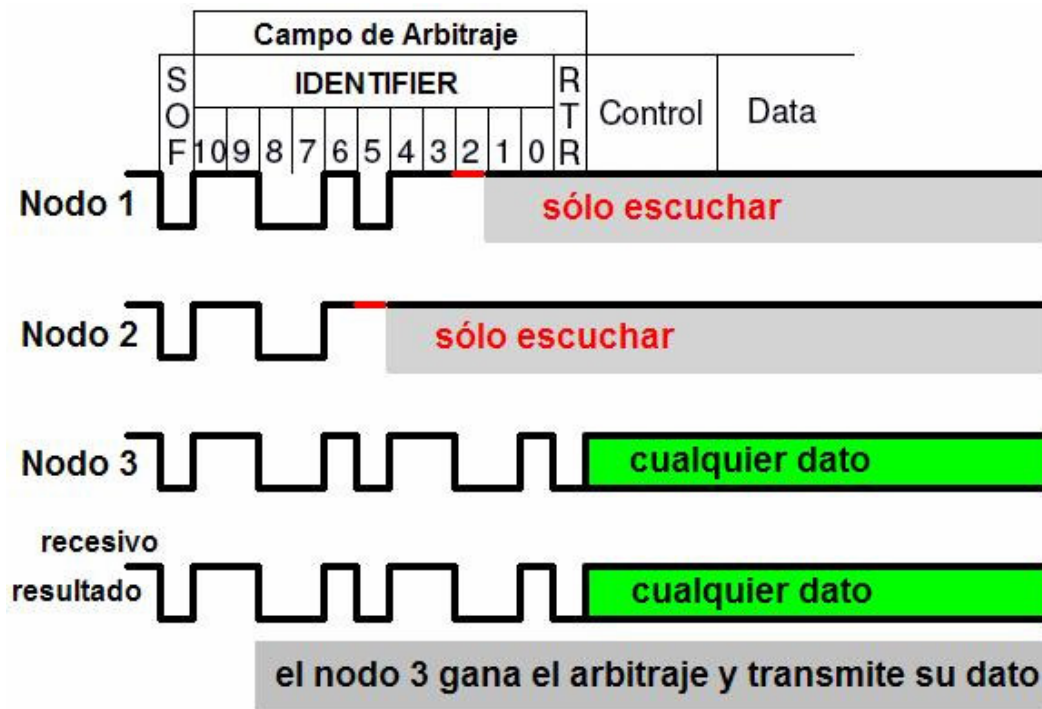
Como se ha mencionado, DeviceNet implementa el protocolo de CAN en su capa de enlace. Así, el método de acceso al medio definido por CAN consiste en un mecanismo de arbitraje de bus para evitar la pérdida de la información debida a una colisión. Este método ha sido llamado CSMA/NBA, lo cual es la sigla para Carrier Sense Media Access/Not destructive Bit-wise Arbitration, que en español puede traducirse como Acceso al Medio por Detección de Portadora/Arbitraje de Bit-inteligente no destructivo.

6.3.2 Arbitraje del bus

Cada vez que el bus está libre, cualquier unidad (nodo) puede comenzar a transmitir un mensaje. Si 2 o más unidades comienzan a transmitir un mensaje en el mismo momento, el conflicto por el acceso al bus es resuelto por el mecanismo de bit-wise arbitration (o arbitraje de bit inteligente) usando el IDENTIFIER (o identificador), contenido en la trama CAN. El mecanismo de arbitraje garantiza que ninguna información ni tiempo es perdido.

Como se mencionó en la sección 6.2.3, el protocolo CAN define dos estados para el bus: “recesivo” (1 lógico) y “dominante” (0 lógico). Los nombres de los estados lógicos tienen que ver con el proceso de arbitraje. Durante el arbitraje todo transmisor compara el nivel del bit transmitido con el nivel de bit que es detectado en el bus. Si estos niveles son iguales la unidad puede continuar enviando. Cuando un nivel “recesivo” es transmitido y un nivel “dominante” es detectado, la unidad ha perdido el arbitraje y debe retirarse sin enviar ningún bit más.

En la figura 6.30 se muestra una representación de cómo opera el mecanismo de arbitraje del bus CAN, en dónde 3 nodos comienzan a transmitir simultáneamente.



Después de perder el arbitraje, un nodo cambia modo "sólo escuchar".

Figura 6.30. Representación del mecanismo de arbitraje del bus CAN.

Se observa que los 3 nodos transmiten un IDENTIFIER que es igual hasta el bit 6. Luego de ese momento el mecanismo de arbitraje realiza las siguientes acciones:

- En el bit 5 del IDENTIFIER, el nodo 2 transmite un nivel recesivo, mientras que los nodos 1 y 3 transmiten un nivel dominante. De esta manera, el nodo 2 pierde el arbitraje al detectar que el estado del bus es recesivo cuando está transmitiendo un nivel recesivo. Por lo tanto, entra en modo "sólo escuchar" y los nodos 1 y 3 aún participan por la contienda del bus.
- Hasta el bit 3 del IDENTIFIER, los nodos 1 y 3 transmiten niveles de bit iguales. Situación que cambia en el bit 2, en donde el nodo 1 transmite un bit recesivo mientras que el nodo 3 transmite un bit dominante. Ahora el nodo 1 pierde el arbitraje frente al nodo 3 y entra en estado de sólo escuchar. El nodo 3 ha ganado el arbitraje al no haber más contendores por el bus y transmitirá su dato.

6.3.3 Definición del Transmisor/Receptor

En el contexto del protocolo de comunicación CAN, se define la unidad Transmisora o Receptora según la función que está realizando, en determinado momento.

- **Transmisor CAN:** una unidad originadora de un mensaje es llamada “Transmisor” de ese mensaje. La unidad permanece como Transmisor hasta que el bus está desocupado o bien, hasta que la unidad pierda el arbitraje.
- **Receptor CAN:** una unidad es llamada “Receptora” de un mensaje, si esta no es Transmisora de un mensaje y el bus no está desocupado.

6.3.4 Formato de la Trama CAN para DeviceNet

a) Tipos de Tramas CAN

La especificación CAN define 4 tipos de tramas, o frames, para la transmisión de datos, y son las siguientes:

- **Data Frame** (Trama de Dato): porta datos desde un transmisor a los receptores.
- **Remote Frame** (Trama Remota): es transmitido por una unidad del bus para requerir la transmisión del Data Frame con el mismo IDENTIFIER.
- **Error Frame** (Trama de Error): es transmitido por cualquier unidad que haya detectado un error en el bus.
- **Overload Frame** (Trama de Sobrecarga): es usado para proveer un retardo extra entre el precedente y el siguiente Data Frame o Remote Frame.

El protocolo DeviceNet utiliza *sólo* el **Data Frame** de CAN, por lo cual es considerado en detalle a continuación.

b) El Data Frame

El *Data Frame*, o Trama de Dato, porta los datos desde un dispositivo transmisor a uno o varios dispositivos receptores. Está compuesto de 7 diferentes campos de bits:

- **Start of Frame** (Comienzo de Trama)
- **Arbitration Frame** (Arbitraje de Trama)
- **Control Field** (Campo de Control)
- **Data Field** (Campo de Dato)
- **CRC Field** (Campo CRC)
- **ACK Field** (Campo ACK)
- **End of Frame** (Fin de Trama)

En la figura 6.31 se muestra la representación del Data Frame.

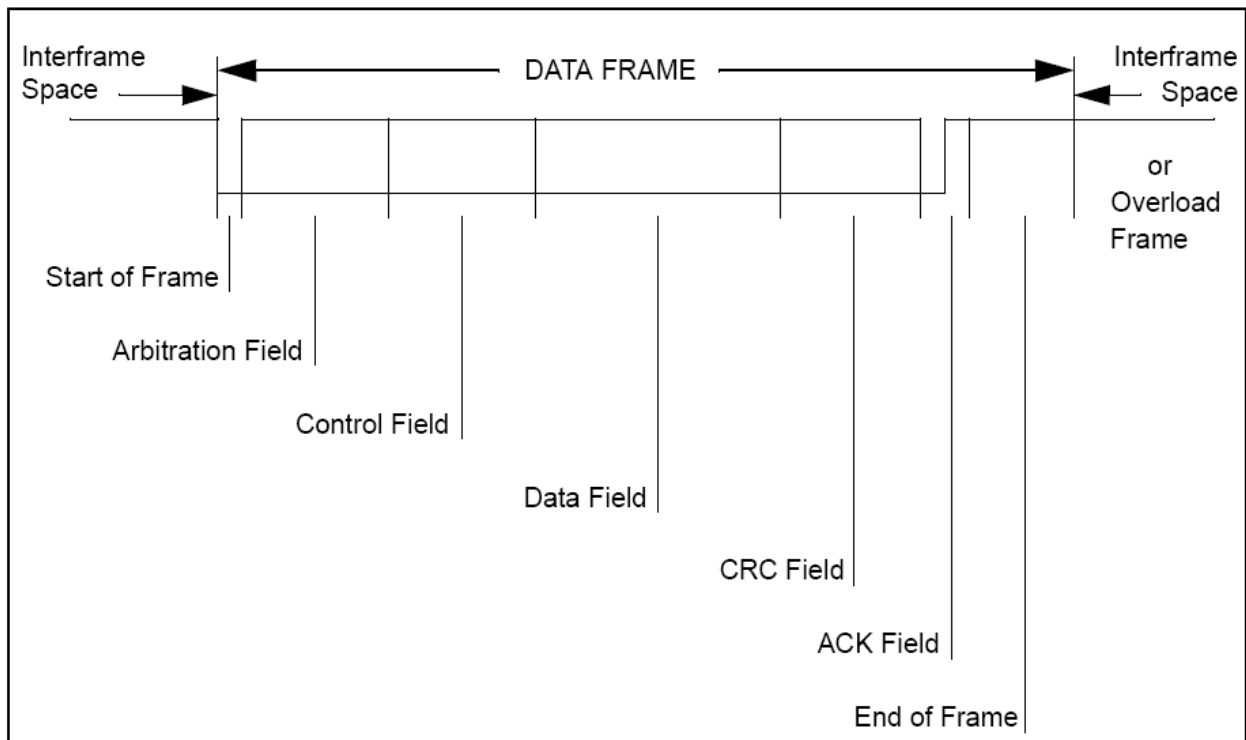


Figura 6.31. Representación del Data Frame CAN.

b.1) Campo de bit Start of Frame

El *Start of Frame* indica el comienzo del Data Frame. Consiste de un único bit “dominante”. Una estación puede comenzar a transmitir solamente cuando el bus está desocupado. Al comenzar a transmitir un nodo, éste lleva al bus a un estado dominante a través del bit de comienzo Start of Frame, por lo cual, todas las estaciones deben sincronizarse al cambio de estado producido por este para escuchar la transmisión.

En el caso de que más de una unidad comience a transmitir simultáneamente, el conflicto por el acceso al medio se resuelve mediante el arbitraje.

b.2) Campo de bit Arbitration Field

El *Arbitration Field*, o Campo de Arbitraje, es la porción del Data Frame en dónde opera el mecanismo de arbitraje para resolver el conflicto por el acceso al medio. El Arbitration Field de CAN se compone del Identifier, o Identificador, y el RTR-Bit. En la figura 6.32 puede observarse representación del Campo de Arbitraje.

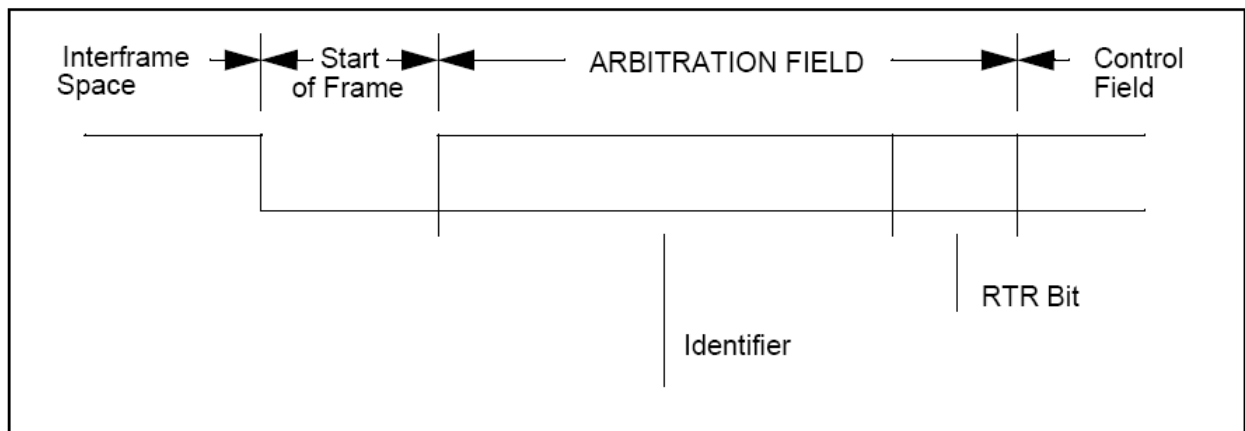


Figura 6.32. Campo de bit ARBITARTION FIELD del DATA FRAME.

b.2.1) El Identifier

El *Identifier*, o Identificador, tal como su nombre lo indica, permite identificar el mensaje y su prioridad sobre otros. Posee una longitud de 11 bits. Estos bits son transmitidos en el orden desde ID-10 hasta ID-0, tal como puede apreciarse en la figura 6.30 anterior. El bit menos significativo es ID-0. Los 7 bits más significativos, ID-10 – ID-4, no deben ser todos “recesivos”. Como se puede deducir, los mensajes con Identifier de menor valor son los que tendrán mayor prioridad, dado que comenzarán con niveles de bit dominante que les permitirán ganar el arbitraje, tal como se pudo apreciar en la sección 6.3.2.

b.2.2) El RTR Bit

El *RTR Bit* (Remote Transmisión Request Bit) se implementa para distinguir entre un Data Frame y un Remote Frame que posean un mismo Identifier. De éstos, el Data Frame tiene mayor prioridad, por lo su RTR-Bit tiene un valor dominante, mientras que el RTR-bit del Remote Frame es recesivo. Sin embargo, éste bit no es utilizado por DeviceNet, dado que sólo se utiliza el Data Frame y, por lo tanto, no necesita realizar esta distinción.

c) El Control Field

El *Control Field*, o Campo de Control, entrega la información acerca de cuántos bytes de datos son contenidos dentro del Data Field, o Campo de Dato, que le sigue después. El Control Field consiste de 6 bits, distribuidos en el Data-Length-Code, o Código de Longitud e Dato, de 4 bits y los 2 bits restantes están reservados. Los bits reservados deben ser enviados como “dominante”. En la figura 6.33 se muestra una representación de este campo.

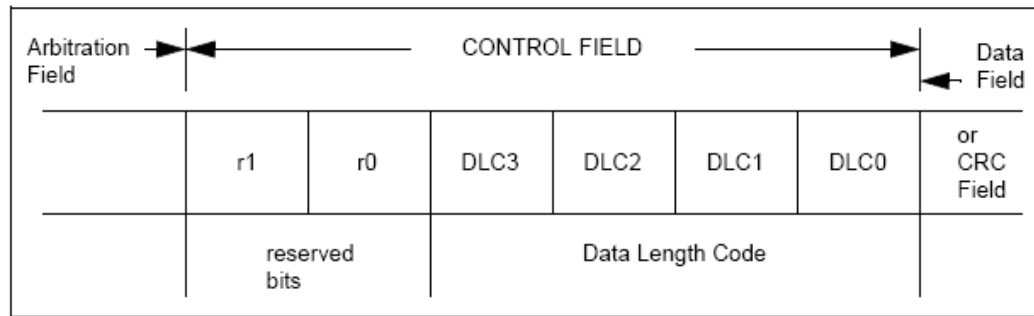


Figura 6.33. Representación del campo de bits CONTROL FIELD.

c.1) El Data Length Code

El *Data-Length-Code*, o Código de Longitud de Dato, indica el número de bytes contenidos en el Data Field. El Data-Length-Code tiene un ancho de 4 bits y es transmitido dentro del Control Field. La tabla 6.3 muestra la codificación del número de bytes de datos por el Data-Length-Code, en donde d es “dominante”, y r es “recesivo”.

Number of Data Bytes	Data Length Code			
	DLC3	DLC2	DLC1	DLC0
0	d	d	d	d
1	d	d	d	r
2	d	d	r	d
3	d	d	r	r
4	d	r	d	d
5	d	r	d	r
6	d	r	r	d
7	d	r	r	r
8	r	d	d	d

Tabla 6.3 Codificación Data-Length-Code, en donde: d es “dominante”, y r es “recesivo”.

De acuerdo con la tabla 6.3, el Data Frame puede contener entre 0 y 8 bytes de datos admisibles. Otros valores no pueden ser utilizados.

d) El Data Field

El *Data Field*, o Campo de Dato, contiene los datos a transferir. Puede contener desde 0 hasta 8 bytes, lo cual es indicado previamente en el Campo de Control. Cada byte es de 8 bits los cuales son transmitidos desde el MSB al LMS.

e) El CRC Field

El *CRC Field*, o Campo de CRC, tiene una longitud de 16 bits, divididos en la Secuencia CRC (CRC Sequence) de 15 bits, seguida por el Delimitador CRC (CRC Delimiter) de 1 bit. En la figura 6.34 se muestra la estructura del CRC Field.

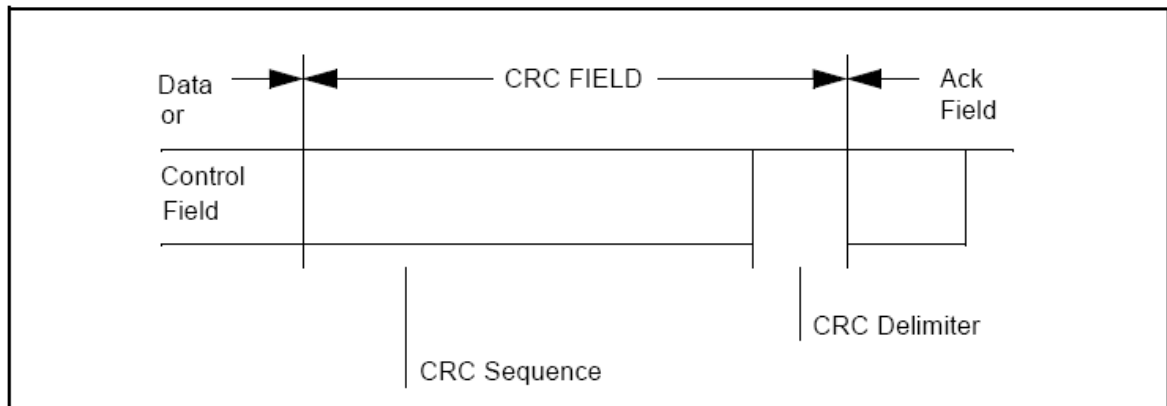


Figura 6.34. Representación de la CRC SEQUENCE junto al CRC DELIMITER.

e.1) El CRC Sequence

Corresponde a una secuencia de chequeo de trama la cual es derivada de un código de redundancia cíclica, o CRC, que opera mejor en tramas con cantidades de bit menor a 127 bits (BCH Code).

Para llevar a cabo el cálculo del CRC se define un polinomio $M(X)$, que debe ser dividido por el generador polinomial $G(X)$. Este polinomio $M(X)$ se compone de los coeficientes obtenidos de los campos de bits de la trama CAN Start of Frame, Arbitration Field, Control Field, Data Field (sin los bits de relleno), y 15 coeficientes menores con valor de 0. El generador polinomial $G(X)$ es (los coeficientes son calculados en módulo 2):

$$G(X) = X^{15} + X^{14} + X^{10} + X^8 + X^7 + X^4 + X^3 + 1.$$

El remanente de esta división polinómica es la Secuencia CRC transmitida.

e.2) El CRC Delimiter

El CRC Delimiter, o Delimitador CRC, sigue a la Secuencia CRC y consiste de un único bit “recesivo”.

f) El ACK Field

El *ACK Field*, o Campo ACK, significa el Campo de Reconocimiento. Se compone de 2 bits, el ACK Slot, o Ranura ACK, y el ACK Delimiter, o Delimitador ACK. La figura 6.35 muestra la representación del campo de bits ACK Field.

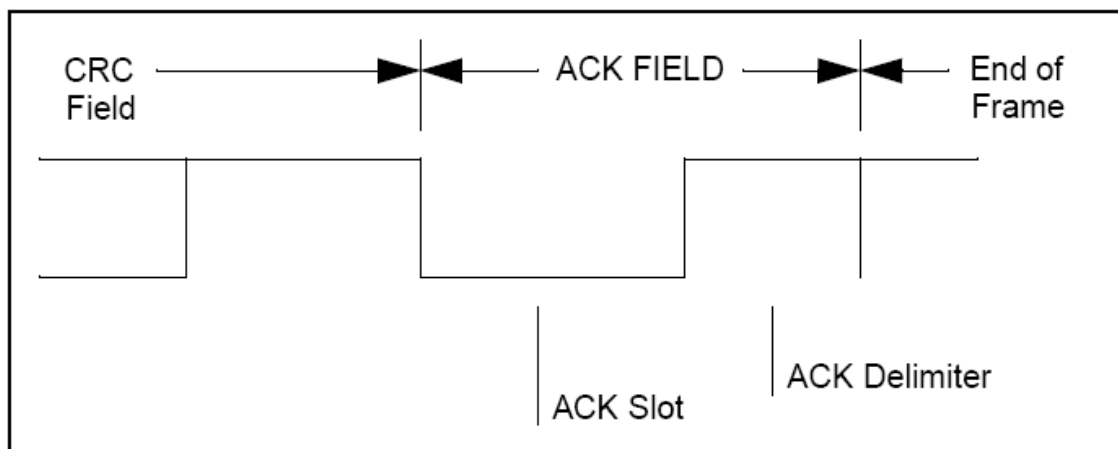


Figura 6.35. Representación del campo de bits ACK FIELD.

Cuando el nodo Transmisor envía los 2 bits del ACK Field, éstos son de niveles “recesivos”. Cuando el nodo Receptor ha recibido un mensaje correctamente, reporta esto al Transmisor por el envío de un bit “dominante” durante el ACK Slot (envía un “ACK” o reconocimiento).

f.1) El ACK Slot

Todas las estaciones habiendo recibido la Secuencia CRC reportan esto dentro del ACK SLOT sobrescribiendo el bit “recesivo” del transmisor por un bit “dominante”.

f.2) El ACK Delimiter

El ACK Delimiter es el segundo bit del ACK Field y debe ser un bit “recesivo”. Como consecuencia, el ACK Slot es rodeado por 2 bits “recesivos” (CRC Delimiter y el ACK Delimiter).

g) El End of Frame

El End of Frame, o Fin de Trama, delimita a todo Data Frame, para indicar, tal como su nombre lo indica, el fin de éste. Consiste de una secuencia de 7 bits “recesivos”.

6.3.5 Codificación

Como se mencionó en la sección 6.2.5, el método de codificación de bit es el NRZ y se utiliza la técnica de bit de relleno (bit-stuffing) cada vez que el transmisor detecta 5 bits consecutivos idénticos. Sin embargo, no se aplica relleno bits a todos los campos del Data Frame.

La técnica de relleno de bits se aplica en los siguientes campos del Data Frame:

- Start of Frame
- Arbitration Field
- Control Field
- Data Field
- CRC Sequence del CRC Field

Dado que los campos restantes son de formato fijo (CRC Delimiter, ACK Field y End of Frame), estos no son rellenados.

6.3.6 Validación de Mensaje

El momento en que un mensaje es tomado para ser validado, es diferente para el Transmisor y el Receptor del mensaje.

- **Validación del Transmisor:** El mensaje es validado por el transmisor, si no hay error hasta el final del End of Frame. Si un mensaje está dañado, será retransmitido automáticamente y de acuerdo a la priorización. Para poder competir por el acceso al bus con otros mensajes, la retransmisión debe comenzar tan pronto como el bus esté desocupado.
- **Validación del Receptor:** El mensaje es validado por los receptores, si no hay error hasta el último bit del End of Frame.

6.4 Las Capas de Red y Transporte DeviceNet

6.4.1 Utilización del CAN Identifier (CAN-ID)

Dado que DeviceNet implementa CAN en su capa de enlace, se utiliza el Identifier de 11 bits, del campo de arbitraje del Data Frame, para identificar diferentes mensajes. Por consiguiente, pueden distinguirse $2^{11} = 2048$ mensajes.

El Identifier de CAN, llamado de ahora en adelante CAN-ID, es desglosado por DeviceNet en las siguientes dos partes:

- **MAC-ID**
- **Mensaje-ID**

El MAC-ID es la dirección de nodo (ya sea fuente o destino) dentro de la red. Siempre es de 6 bits. Por lo tanto, la cantidad de nodos admitidos en una red DeviceNet es de 64.

El Mensaje-ID identifica a un tipo de mensaje.

El rango de CAN-IDs está dividido en 4 Grupos de Mensaje, variando el tamaño. En la figura 6.36 se pueden observar estos rangos.

Connection ID = CAN Identifier (bits 10:0)											Used for
10	9	8	7	6	5	4	3	2	1	0	
0	Message ID				Source MAC ID						Message Group 1
1	0	MAC ID						Message ID			Message Group 2
1	1	Message ID			Source MAC ID						Message Group 3
1	1	1	1	1	Message ID						Message Group 4
1	1	1	1	1	1	1	x	x	x	x	Invalid CAN Identifiers

Figura 6.36. Grupos de Mensajes DeviceNet.

En DeviceNet, el CAN-ID es la Connection-ID, CID (o Conexión ID). Comprende el Grupo de Mensaje ID, el Mensaje-ID dentro de este grupo y el MAC-ID del dispositivo, que puede ser dirección de fuente o destino. La definición depende del Grupo de Mensaje y del Mensaje ID.

6.4.2 Los Grupos de Mensajes DeviceNet

Los 4 Grupos de Mensajes usados en DeviceNet se describen a continuación.

a) Mensaje de Grupo 1

Tiene asignados 1024 CAN-ID (0x0000 – 0x03FF), que es el 50% de todos los IDENTIFIERS disponibles. Hasta 16 Mensajes-IDs diferentes están disponibles por dispositivos (nodo) dentro de este grupo. La prioridad de un mensaje de este grupo es primeramente determinada por el Mensaje-ID (el significado de este mensaje), y solo después por la fuente MAC-ID (dispositivo productor). Si 2 dispositivos transmiten al mismo tiempo, entonces el dispositivo con el más bajo Mensaje-ID ganará siempre el arbitraje. Sin embargo, si 2 dispositivos transmiten el mismo Mensaje-ID al mismo tiempo sobre el bus CAN, entonces el dispositivo con el menor MAC-ID ganará. Por lo tanto, los Mensajes de Grupo 1 están hechos para el intercambio de datos de proceso de alta prioridad.

b) Mensajes de Grupo 2

Tiene asignado 512 CAN-ID (0x0400 – 0x05FF). La mayoría de los Mensajes-IDs en este grupo son definidos opcionalmente a lo que normalmente es llamado el Predefined Master/Slave Connection Set (traducido al español como Conjunto de Conexión Predefinida Maestro/Esclavo). Un Mensaje-ID es definido para la gestión de red. La prioridad se determina primero por medio del MAC ID y, sólo después, por medio del Mensaje ID.

c) Mensaje de Grupo 3

Tiene asignados 448 CAN-ID (0x0600 – 0x07BF). Tiene una estructura similar a la del Grupo 1. Sin embargo, a diferencia de ese grupo, intercambia datos de proceso de baja prioridad. Además, el uso principal de este grupo es preparar Conexiones Explícitas dinámicas. Son posibles 7 Mensajes IDs por dispositivo, y 2 de estos están reservados a lo que es llamado el puerto Unconnected Message Manager, UCMM, lo cual se puede traducir al español como el puerto Gestionador de Mensajería No Conectada.

d) Mensaje de Grupo 4

Tiene asignados 48 CAN-ID (0x07C0 – 0x07EF). No incluye ninguna MAC ID, sólo Mensajes ID. Los mensajes en este grupo sólo son usados para la gestión de la red. Cuatro Mensajes-ID son normalmente asignados para los servicios del Conjunto de Conexión Offline. Los restantes 16 CAN-ID (0x07F0 – 0x07FF) son inválidos y por lo tanto no se permite su uso en sistemas DeviceNet.

Con esta asignación de CAN-ID, no pueden ser usados por otros los CAN-ID inválidos. Por consiguiente, cada dispositivo tiene exactamente:

- **16 Mensajes ID en el Grupo 1,**
- **08 Mensajes ID en el Grupo 2,**
- **07 Mensajes ID en el Grupo 3, y**
- **0 Mensajes ID en el Grupo 4.**

Una ventaja de este sistema es que los CAN-ID usados en la red siempre pueden ser claramente asignados a un dispositivo. Los dispositivos son responsables de la gestión de sus propios identificadores. Esto simplifica el diseño, la mantención y el diagnóstico del sistema DeviceNet. De esta manera, no es necesaria una herramienta central que guarde un registro de todas las asignaciones en la red.

6.4.3 Establecimiento de la Conexión

El intercambio de mensajes en una red DeviceNet está basado en conexión. Esto significa que debe establecerse primero una conexión entre los dispositivos antes de poder transmitir y recibir. Se puede comparar a una conexión del circuito telefónico. Cuando se realiza una llamada, el sistema de telefonía selecciona un camino para ésta y configura cada estación de combinación en la ruta. Con tal de que la llamada continúe, resulta un circuito virtual (CV) que permanece abierto, transportando datos o tráfico de voz. En el sistema telefónico, una llamada puede atravesar múltiples y diferentes tipos de enlaces. A través de todo esto, aparece la misma conexión en ambos lados de la comunicación: sonido en un extremo, volviéndose sonido también en el otro.

También, una conexión entrega un camino entre 2 puntos extremos. Una vez que el administrador de la conexión determina el circuito virtual, la ruta entre los puntos extremos es fijada. En la figura 6.37 se representa una transferencia de mensajes DeviceNet a través de un CV basado en conexión.

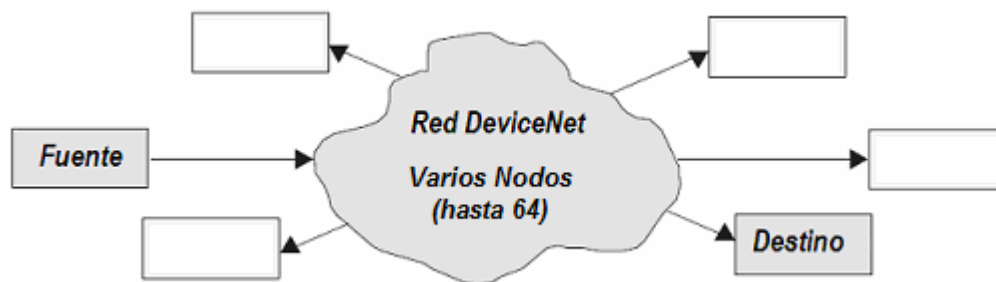


Figura 6.37. Circuito Virtual DeviceNet.

Los Communication Objects (Objetos de Comunicación), deben estar preparados para este propósito. Éstos no están disponibles inicialmente cuando un dispositivo es encendido; ellos deben ser creados primero. Los únicos puertos por los que un dispositivo DeviceNet puede ser direccionado cuando es encendido por primera vez, son el puerto Unconnected Message Manager (UCMM) o el puerto Group 2 Only Unconnected Request del Predefined Master/Slave Connection Set (que puede traducirse al español como Conjunto de Conexión Predefinida Maestro/Esclavo).

Para entender cómo funcionan estos puertos, puede ilustrarse que son como las puertas del dispositivo. Sólo una llave abrirá cada puerta. La llave apropiada para cada cerradura es la CONNECTION ID (el CAN-ID con la indicación del MAC-ID y el Mensaje-ID) del puerto seleccionado. Otras puertas en el dispositivo pueden ser abiertas sólo si la llave apropiada está disponible y otras instancias del Connection Objects están preparadas.

La comunicación a través del puerto UCMM representa un procedimiento general que debe ser agregado en todo dispositivo DeviceNet. Dispositivos con la característica de Predefined Master/Slave Connection Set y capacidades UCMM son llamados Servidores de Grupo 2. Un Servidor de Grupo 2 puede ser direccionado por uno o más conexiones de uno o más clientes.

Dado que un dispositivo con capacidades UCMM necesita de buena potencia de proceso para requerimientos de múltiples servicios de comunicación, se ha creado un método de establecimiento de comunicación simplificada e intercambio de I/O para dispositivos de bajas capacidades. Este es llamado el Predefined Master/Slave Connection Set, o Conjunto de Conexión Predefinida Maestro/Esclavo (ver la sección 6.4.4 siguiente). Este cubre 5 conexiones predefinidas que pueden ser activadas (asignadas) al acceder al dispositivo. El Predefined Master/Slave Connection Set representa un subconjunto del método de establecimiento de conexión general, y se limita a relaciones tipo Master/Slave.

Los dispositivos Slave sin capacidades de UCMM y que sólo soportan éste subconjunto son llamados *Group 2 Only Servers*. Sólo el Master que asigna esto puede direccional a un Group 2 Only Server. Todos los mensajes recibidos por este dispositivo son definidos en el Mensaje de Grupo 2.

6.4.4 El Predefined Master/Slave Connection Set

El establecimiento de una conexión vía el puerto UCMM requiere, relativamente, numerosos pasos que deben ser completados para permitir el intercambio de datos vía DeviceNet, y los dispositivos deben entregar recursos para administrar las conexiones dinámicas. Dado que

todo dispositivo puede preparar una conexión con cualquier otro dispositivo, y el MAC-ID fuente de los dispositivos es contenido en la Connection ID, el CAN-ID puede tener que ser filtrado vía software. Esto depende de sobre cuántas conexiones el dispositivos soporta, y sobre el tipo y número de screeners (filtros CAN-ID en hardware) del chip con protocolo CAN usado en la fabricación del dispositivo.

Mientras esta aproximación maximiza el uso de capacidades CAN de multicast, peer-to-peer, y Productor/Consumidor, un método más simple requiere menos recursos de CPU para dispositivos menos complejos. Ésta es la razón por el cual fue definido el Predefined Master/Slave Connection Set (Conjunto de Conexión Predefinida Maestro/Esclavo). El puerto Group 2 Only Unconnected Explicit Message (Grupo 2 Sólo Mensajería Explícita No Conectada) del Predefined Master/Slave Connection Set entrega una interfaz por un conjunto de 5 tipos de conexiones preconfiguradas en un nodo.

Las bases de este modelo es un estructura de comunicación 1:n, esto es, un dispositivo de control a “n” dispositivos I/O descentralizados. La porción central de cada sistema es conocida como el “Master” y los dispositivos descentralizados son conocidos como “Slaves”. Se permite que se conecten a la red varios Masters, pero un Slave puede ser asignado a uno solo de éstos.

Los Objetos de Conexión Predefinidos (Connection Objects Predefined) ocupan instancias 1 a 5 en el Connection Object (Class ID 0x05):

- Conexión de Mensajería Explícita (Explicit Messaging Connection):
 - Group 2 Explicit Request/Response Message (Instancia ID 1)
- Conexiones de Mensajería I/O (I/O Messaging Connections):
 - Polled I/O Connection (Instancia ID 2)
 - Bit-Strobe I/O Connection (Instancia ID 3)
 - Change of State or Cyclic I/O Connection (Instancia ID 4)
 - Multicast Polling I/O Connection (Instancia ID 5)

Los mensajes a los Slaves están definidos en los Mensajes de Grupo 2, y algunas de las respuestas de los Slaves están contenidas en los Mensajes de Grupo 1. En la figura 6.38 se muestra la distribución definida para las conexiones ID.

Connection ID = CAN Identifier (bits 10:0)											Used for
10	9	8	7	6	5	4	3	2	1	0	
0	Group 1 Message ID				Source MAC ID						Group 1 Messages
0	1	1	0	0	Source MAC ID						Slave's I/O Multicast Poll Response
0	1	1	0	1	Source MAC ID						Slave's I/O Change of State or Cyclic Message
0	1	1	1	0	Source MAC ID						Slave's I/O Bit-Strobe Response Message
0	1	1	1	1	Source MAC ID						Slave's I/O Poll Response or COS/Cyclic Ack Message
1	0	MAC ID					Group 2 Message ID				Group 2 Messages
1	0	Source MAC ID					0	0	0		Master's I/O Bit-Strobe Command Message
1	0	Source MAC ID					0	0	1		Master's I/O Multicast Poll Group ID
1	0	Destination MAC ID					0	1	0		Master's Change of State or Cyclic Acknowledge Message
1	0	Source MAC ID					0	1	1		Slave's Explicit/Unconnected Response Messages
1	0	Destination MAC ID					1	0	0		Master's Explicit Request Messages
1	0	Destination MAC ID					1	0	1		Master's I/O Poll Command/COS/Cyclic Message
1	0	Destination MAC ID					1	1	0		Group 2 Only Unconnected Explicit Request Messages
1	0	Destination MAC ID					1	1	1		Duplicate MAC ID Check Messages

Figura 6.38. Distribución de las Conexiones ID definidas en DeviceNet del Predefined Master/Slave Connection Set.

Dado que en el CAN ID de la mayoría de los mensajes, los Master productores contienen el MAC-ID destino del Slave, es imperativo que sólo un Master “hable” a cualquier Slave. En consecuencia, el Master debe asignar a un Slave su Predefined Connection Set, antes de usarlo. El DeviceNet Object, u Objeto DeviceNet, de CIP (ver sección 6.5.2) gestiona esta importante función en el dispositivo Slave. Esto permite que sólo un Master asigne su Predefined Connection Set, con lo cual se previene la duplicación de CAN-IDs.

Los dos servicios usados son llamados `Allocate_Master/Slave_Connection_Set` (Código de Servicio 0x4B), que se puede traducir al español como Conjunto de Conexión de Asignación Maestro/Esclavo, y `Release_Group_2_Identifier_Set` (Código de Servicio 0x4C), que puede ser traducido como Conjunto de Identificación de Lanzamiento Grupo 2.

Estos dos servicios siempre acceden a la instancia 1 del Objeto DeviceNet (Clase ID 0x03). En la figura 6.39 se muestra el formato del mensaje de requerimiento Allocate_Master/Slave_Connection_Set.

	Bit number								
Byte offset	7	6	5	4	3	2	1	0	
0	Frag [0]	XID	MAC ID						Message header
1	R/R [0]	Service Code [0x4B]							Message body
2...5	Class ID [0x03]								
	Instance ID [0x01]								
	Allocation Choice								
	0	0	Allocator's MAC ID						

Figura 6.39. Mensaje de requerimiento Allocate_Master/Slave_Connection_Set (Código de Servicio 0x4B).

El mensaje de requerimiento Allocate Message, mostrado en la figura anterior, tiene una Class ID de 8 bits y una Instancia ID de 8 bits. Siempre es usado este formato cuando es enviado como un Group 2 Only Unconnected Message. Puede usarse otro formato para una conexión existente siempre que haya sido acordado un formato distinto a 8/8 en el establecimiento de la conexión.

El Allocation Choice Byte (Byte de Selección de Asignación) es usado para determinar qué tipo de conexión predefinida es asignada. En la figura 6.40 se muestra el formato de este byte.

Bit number							
7	6	5	4	3	2	1	0
Reserved	Ack Suppression	Cyclic	Change of State	Multicast Polling	Bit-Strobe	Polled	Explicit Message

Figura 6.40. Formato del Allocation Choice Byte.

Las conexiones asociadas son activadas por la selección de los bits apropiados. Change of State (Change of State) y Cyclic (Cíclico) son opciones mutuamente exclusivas. La conexión Change of State/Cyclic puede ser configurada sin reconocimiento usando Supresión de Reconocimiento. Los tipos de conexión individual son descritos en más detalles abajo.

Los MAC-ID asignados contienen la dirección del nodo Master que desea asignar el Predefined Master/Slave Connection Set. El Byte 0 de este mensaje difiere de los MAC-ID asignados si este servicio ha sido pasado a un Grupo 2 Only Server vía un Group 2 Only Client (que es normalmente llamado como “proxy function”, o función proxy).

El Slave responde con un Success Message (Mensaje exitoso). Las conexiones ahora están en Estado de Configuración. Poniendo en marcha el Expected_Packet_Rate EPR (el servicio Set_Attribute_Single atribuye 9ms en el Objeto de Conexión CIP) comienza la función de monitoreo de tiempo de conexión. La conexión entonces cambia a Established State (Estado Establecido), y los Mensajes I/O (o I/O Messages) comienzan a ser transferidos por esta conexión.

Las conexiones asignadas pueden ser liberadas individualmente o colectivamente a través del servicio Release_Group_2_Identifier_Set (Código de Servicio 0x4C), usando el mismo formato que se muestra en la figura 6.39, excepto que el último byte (MAC-ID de Asignación) es omitido. La figura 6.41 muestra el formato de este servicio.

	Bit number								
Byte offset	7	6	5	4	3	2	1	0	
0	Frag [0]	XID	MAC ID						Message header
1	R/R [0]	Service Code [0x4B]						Message body	
2...5	Class ID [0x03]								
	Instance ID [0x01]								
	Allocation Choice								

Figura 6.41. Servicio Release_Group_2_Identifier_Set.

6.4.5 Conexiones I/O del Predefined Master/Slave Connection Set

a) Conexión I/O Polled

La conexión tipo Polled, o de sondeo, es usada para implementar la clásica relación Master/Slave entre una unidad de control y un dispositivo. En esta configuración, un Master puede transferir datos a un Slave usando el Poll-Request y recibir datos desde el Slave usando el Poll-Response. En la figura 6.42 se muestra el intercambio de datos entre un Master y tres Slaves en un modo Polled I/O.

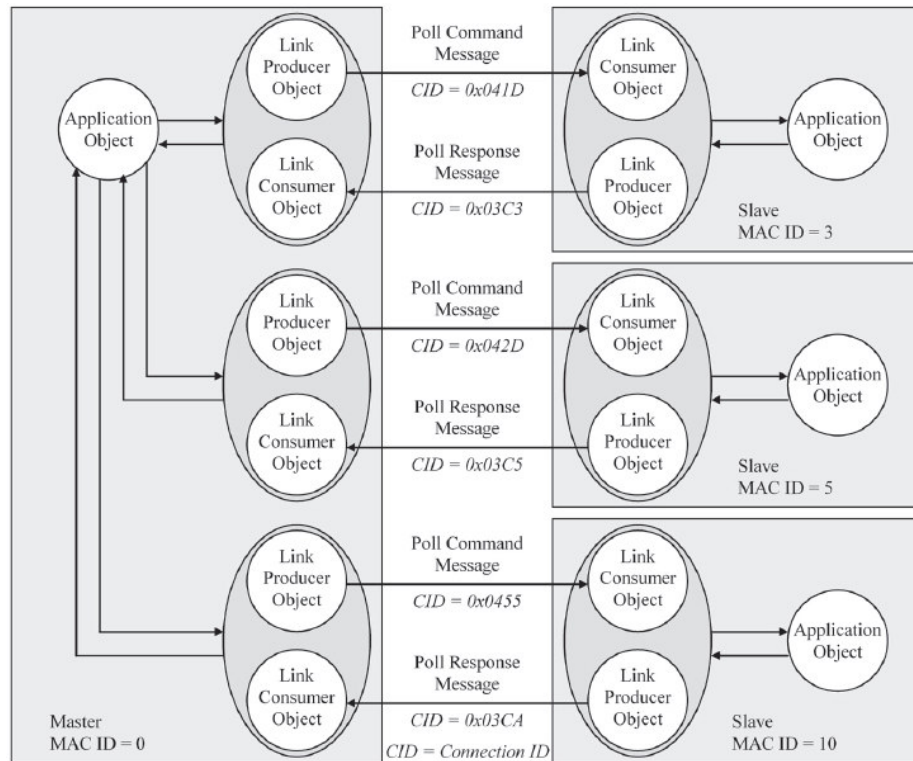


Figura 6.42. Conexiones Polled I/O en DeviceNet.

La cantidad de datos transferidos en un mensaje entre un Master y un Slave usando la conexión Polled puede ser de cualquier longitud. Si la longitud excede a 8 bytes, el protocolo de fragmentación es usado automáticamente. La conexión Polled I/O siempre es una conexión punto a punto entre un Master y un Slave. Los Slaves consumen los mensajes Poll y envían de regreso una respuesta apropiada (normalmente sus datos de entrada).

La conexión Polled está sujeta a la función de monitoreo de tiempo, que puede ser ajustada en el dispositivo. Un Comando Poll debe haber sido recibido dentro de este tiempo ($4 \times \text{EPR}$) o bien la conexión revierte a modo time-out (fuera de tiempo). Cuando una conexión está en time-out, el nodo opcionalmente puede pasar a un estado de falla, preconfigurado según lo haya definido el usuario. El Master realiza el poll en todos los Slaves normalmente mediante el método round-robin.

El tiempo de respuesta de un Slave a un Comando Poll no está definido en la especificación DeviceNet. Esto entrega flexibilidad a los dispositivos Slaves que están diseñados para aplicaciones simples; sin embargo, puede excluir al dispositivo para el uso en aplicaciones de alta velocidad.

b) Conexión Bit-Strobe I/O

La Conexión I/O Bit-Strobe es un tipo de conexión Multicast, en que el Master realiza la transmisión de mensajes a través de un comando llamado Bit-Strobe-Command. Utilizando este comando, el Master envía un mensaje multicast que alcanza a todos los Slaves asignados por la Conexión I/O Bit-Strobe. El frame enviado por el Maestro a través de este comando, siempre es de 8 bytes o de 0 bytes (si está desocupado). De estos 8 bytes, cada Slave está asignado a 1 bit. Cada Slave puede enviar de regreso hasta 8 bytes de datos en su respuesta. En la figura 6.43 se muestra el formato de dato de la conexión Bit-Strobe I/O.

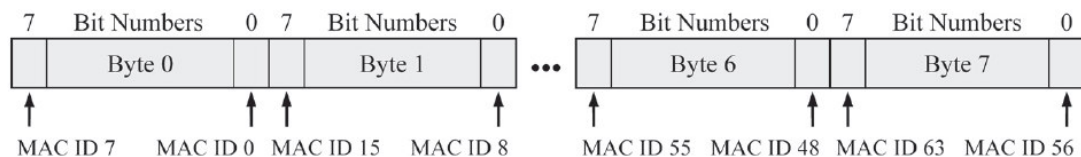


Figura 6.43. Formato de dato de la conexión Bit-Strobe I/O de DeviceNet.

En una conexión Multicast, un Master envía un mensaje sobre la red que todos los nodos Slaves ven simultáneamente, pero que sólo los nodos que estén asignados lo consumirán. En la figura 6.44 se representa una conexión Bit-Strobe como una conexión multicast.

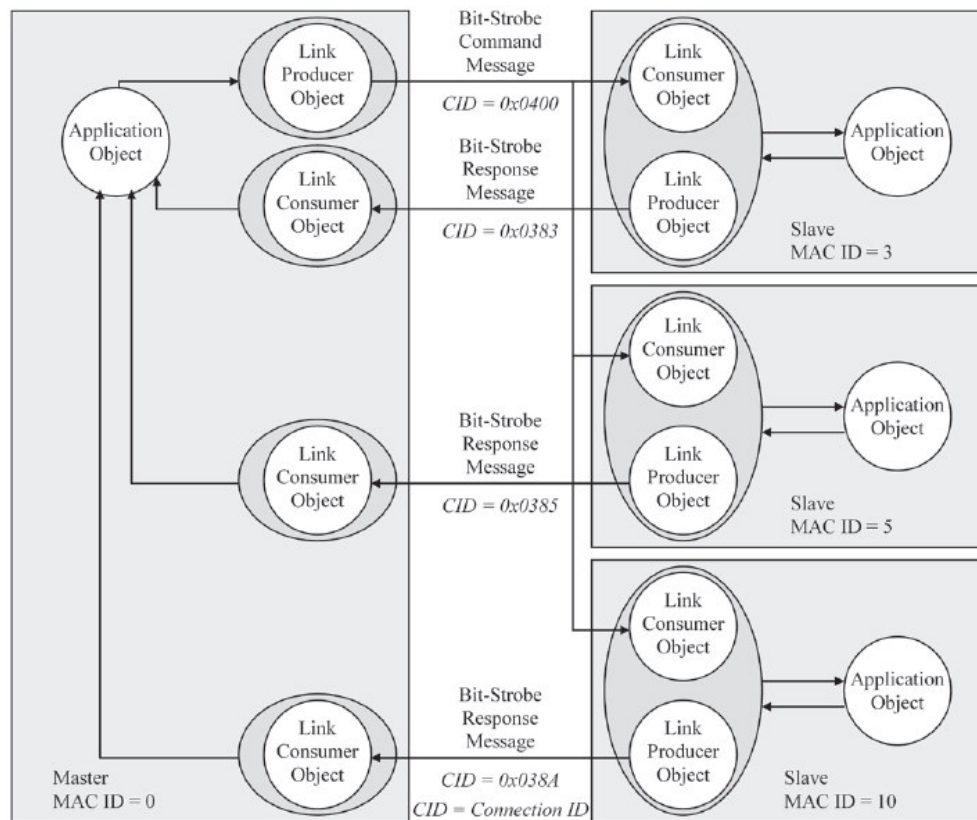


Figura 6.44. Conexión I/O Bit-Strobe de DeviceNet.

Dado que todos los dispositivos en una red reciben el Comando Bit-Strobe al mismo tiempo, ellos pueden ser sincronizados por medio de este comando. Cuando el Comando Bit-Strobe es recibido, el Slave puede consumir su bit asociado, y envía entonces una respuesta de hasta 8 bytes.

Como se desprende de la figura 6.43 anterior, el comando Bit-Strobe utiliza el MAC-ID fuente en la Conexión ID. Debido a esto los dispositivos que soportan la conexión Bit-Strobe I/O y que tienen un chip CAN con screening limitado a 8 bits del CAN-ID (11 bits), deben realizar un screening por software del CAN Identifier.

c) Conexión I/O Cambio de Estado/Cíclico (COS/Cyclic I/O)

La Conexión I/O COS/Cyclic es diferente de los otros tipos de Conexión I/O en que ambos endpoints (puntos extremos) producen sus datos independientemente. Esto puede ser llevado a cabo a través de un Cambio de Estado o por medio de un Ciclo. En el primer caso, la Conexión I/O COS reconoce que el dato del Objeto de Aplicación indicado por el Produced_Connection_Path (o Camino de Conexión Producido) ha cambiado. En el segundo caso, un tiempo de la Conexión I/O Cyclic expira y entonces dispara (*trigger*) la transferencia de mensaje de los últimos datos del Objeto de Aplicación. En la figura 6.45 se representan conexiones Conexión I/O COS/Cyclic.

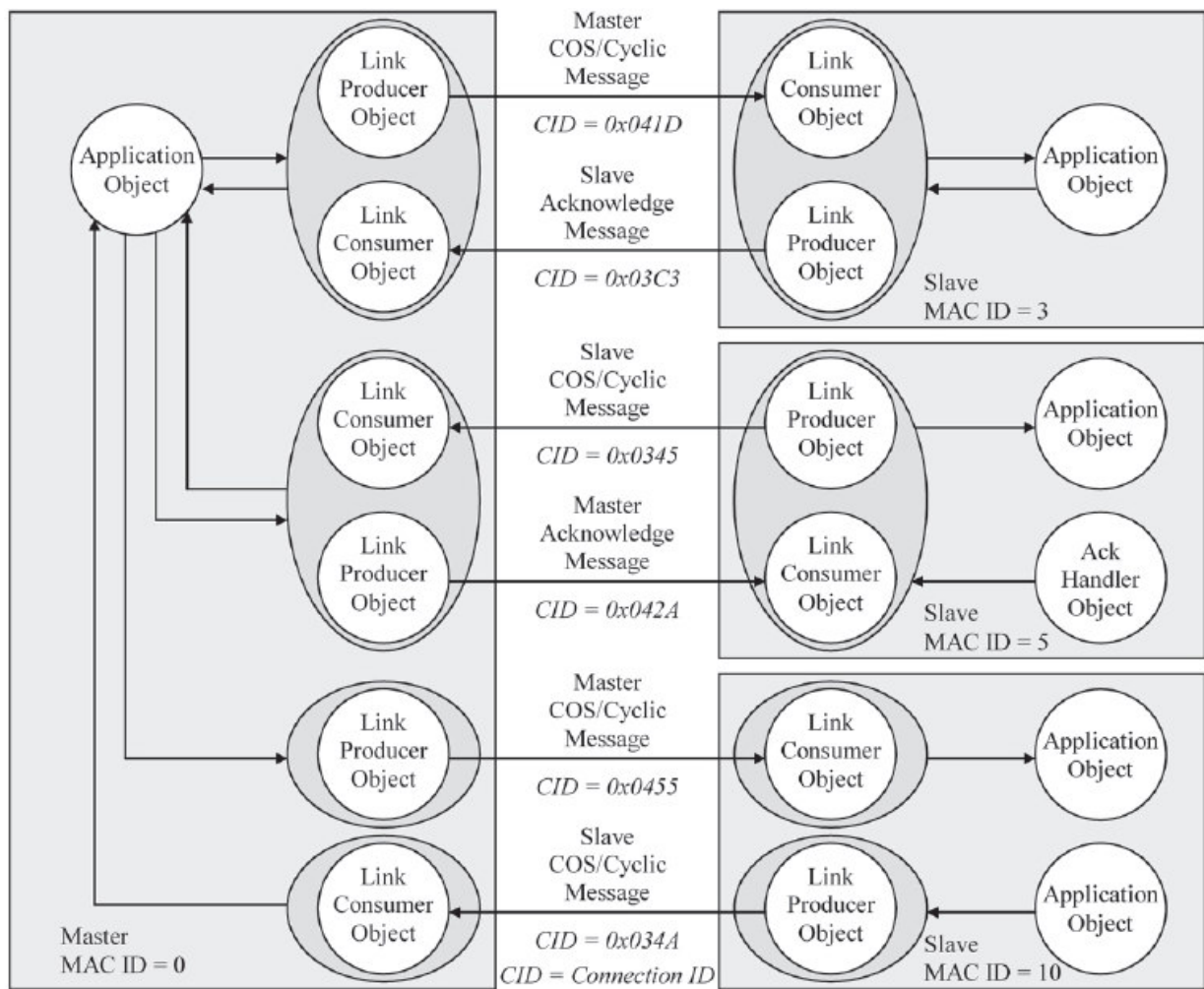


Figura 6.45. Conexiones I/O COS/Cyclic posibles en DeviceNet.

Una Conexión COS/Cyclic I/O puede ser configurada con reconocimiento (Acknowledged) o sin reconocimiento (unacknowledged).

En la configuración Acknowledged, el lado consumidor de la conexión debe definir un camino al Acknowledge Handler Object para asegurar que los reintentos, si son necesarios, son gestionados apropiadamente.

Una Conexión I/O COS/Cyclic puede ser también originada desde un Master, haciéndole aparecer al Slave como una Conexión I/O. Esto puede observarse en la figura 6.45 anterior, dado que la misma Conexión ID es emitida por el Mensaje I/O Polled del Master como es emitido por el Mensaje I/O COS/Cyclic del Master. La figura 6.46 muestra un acercamiento de la figura 6.38 en donde se aprecia esta situación.



Figura 6.46. Acercamiento de la figura 6.38 en donde se aprecia que se utiliza la misma conexión ID para la conexión poll y COS/Cyclic.

Las Conexiones COS poseen dos comportamientos adicionales. El Expected Packet Rate (EPR) es usado como un *Production Trigger*, o productor de disparo, por defecto, tal que, si el dato no tiene cambios después que el Timer-EPR ha expirado, éste será notado. Éste “Heart-beat” (o latido), como algunas veces es llamado, es utilizado para que los nodos consumidores puedan saber la diferencia entre un nodo “muerto” y uno cuyos datos no han cambiado. Las Conexiones COS poseen también una característica *Production Inhibit Timer* (Temporizador de Inhibición de Producción) que previene a un nodo de producir datos demasiados frecuentes, y que en consecuencia utilizan demasiado ancho-de-banda. El Production Inhibit Timer determina la cantidad de tiempo que el nodo debe permanecer en silencio después de producir datos a la red.

6.4.6 Secuencia típica de comienzo Master/Slave

Típicamente, la puesta en marcha de una red DeviceNet con un scanner y un conjunto de slaves (esclavos), se ejecuta como sigue:

- Todos los dispositivos corren su secuencia de auto-testeo e intentan seguir en línea. Cualquier dispositivo que use un mecanismo de auto-detección de tasa de transmisión (autobaud) de la red debe esperar con su Duplicate Node ID Message hasta que ha visto suficiente CAN frame para detectar la correcta tasa de transmisión.
- Una vez en línea, los dispositivos Slaves no harán nada hasta que sean asignados por su Master.
- Una vez en línea, un Master intentará asignar a cada Slave configurado a su lista de escaneo (scan-list) mediante la ejecución de la siguiente secuencia de mensajes:
 - Intenta abrir una conexión a los Slaves usando un UCMM Open Message.
 - Si resulta exitoso, el Master puede usar entonces esta conexión para comunicaciones adicionales con el Slave.
 - Si no resulta exitoso, el Master intentará nuevamente, después de un tiempo mínimo de 1 segundo, abrir la conexión.
 - Si otra vez no resulta exitosa la apertura de conexión, el master intentará asignar el Slave usando el Group 2 Only Unconnected Explicit Request Message (por lo menos para Mensajería Explícita o Explicit Messaging), después de un tiempo mínimo de 1 segundo.

- Si resulta exitosa, el Master puede usar entonces esta conexión para comunicaciones adicionales con el Slave.
 - Si no resulta exitosa, el Master intentará nuevamente después de un tiempo mínimo de 1 segundo.
 - Si no resulta exitosa nuevamente, el Master comenzará con el UCMM Message después de un tiempo mínimo de 1 segundo. Este proceso se repetirá indefinidamente hasta que el Master haya asignado el Slave.
- Una vez que el Master ha asignado el Slave, éste puede llevar alguna verificación para ver si es seguro comenzar la Mensajería I/O (o I/O Messaging) con el Slave. El Master también puede aplicar configuraciones adicionales para las conexiones que ha establecido, por ejemplo, configurando el Explicit Messaging Connection a “Deferred Delete”.
 - Configurando el valor o los valores de EPR que brindan las Conexiones I/O al Estado de Establecimiento (o Established State), puede comenzar la Mensajería I/O (I/O Messaging).

6.5 Las Capas Superiores de la red DeviceNet

6.5.1 Modelo de Objeto para DeviceNet

En la figura 6.47 se muestra la representación abstracta de un dispositivo de la red de campo DeviceNet basado en el modelo de objetos, en el contexto de la definición CIP.

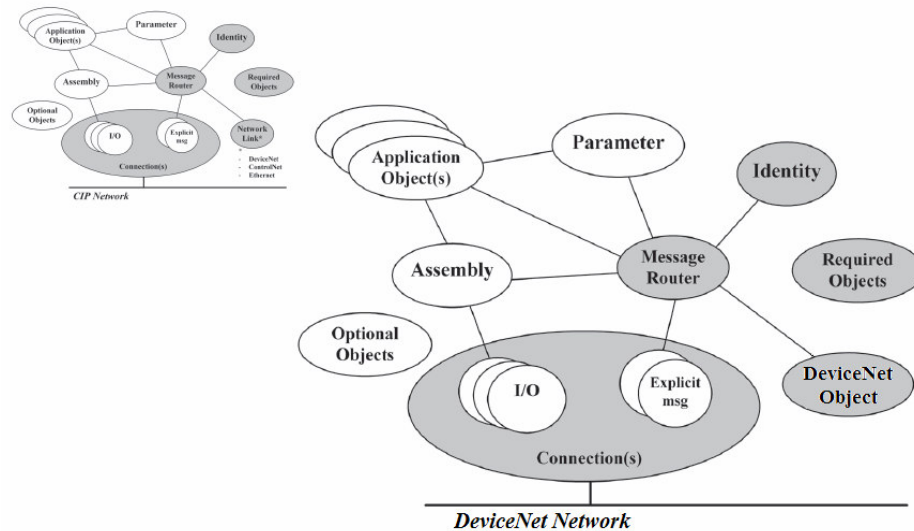


Figura 6.47. Dispositivo de Red DeviceNet basado en el Modelo de Objetos CIP.

Tal como se vio en la sección 4.7, CIP define objetos que son comunes para los dispositivos de las redes basadas en este modelo. Dentro de los objetos, algunos son de requerimiento mínimo para los dispositivos para implementar una red funcional, y como se vio en la sección 4.7.1, los objetos requeridos como mínimo por un dispositivo de red basada en CIP son:

- **Identity Object**
- **Message Router Object**
- **Object Connection o Connection Manage Object**
- **Objetos adicionales específicos de red**

Debido a la naturaleza distinta de cada red, se definen objetos adicionales específicos para cada red.

6.5.2 Objeto adicional específico de red DeviceNet-Object

Para la red DeviceNet, la especificación CIP define el objeto adicional específico de red **DeviceNet-Object**. Este objeto es necesario en todo dispositivo DeviceNet. Las instancias atributos de este objeto contienen la siguiente información para el dispositivo:

- El MAC-ID de los dispositivos de la red.
- La tasa de transmisión de la red.

Ambos atributos no son volátiles, por ejemplo, después de una interrupción de energía, el dispositivo está esperando intentar seguir en línea nuevamente con los mismos valores almacenados en esos atributos antes de de la interrupción de la energía.

El DeviceNet-Object es identificado como un objeto de *Class ID* 0x03.

La tabla 6.4 resume los objetos CIP utilizados por DeviceNet.

Objetos para DeviceNet	
Identity Object	Objetos comunes para redes CIP
Message Router	
Object Connection o Connection Manager Object	
DeviceNet Object	Objeto específico de la red DeviceNet

Tabla 6.4. Resumen de objetos utilizados por DeviceNet.

6.5.3 Perfil de dispositivo

CIP también define el perfil del dispositivo, que identifica el mínimo conjunto de objetos, opciones de configuración y los formatos de los datos I/O que un determinado dispositivo debe tener. Los dispositivos que siguen un perfil estándar tendrán los mismos datos de I/O y opciones de configuración, responderán a todos los mismos comandos y tendrán la misma conducta que otros dispositivos que siguen ese mismo perfil. Aún no se han definido perfiles específicos de la red DeviceNet. Por lo tanto, los dispositivos para la red DeviceNet pueden utilizar cualquiera de los perfiles estándar CIP.

Un subconjunto de perfiles de dispositivo estándar CIP se muestra en la tabla 6.5.

Device Profile	Device Type No.
AC Drives	02 _{hex}
Communications Adapter	0C _{hex}
Contactor	15 _{hex}
DC Drives	13 _{hex}
DC Power Generator	1F _{hex}
General Purpose Discrete I/O	07 _{hex}
Generic Device	00 _{hex}
Human-Machine Interface	18 _{hex}
Inductive Proximity Switch	05 _{hex}
Limit Switch	04 _{hex}
Mass Flow Controller	1A _{hex}
Motor Overload	03 _{hex}
Motor Starter	16 _{hex}
Photoelectric Sensor	06 _{hex}
Pneumatic Valve(s)	1B _{hex}
Position Controller	10 _{hex}
Process Control Valve	1D _{hex}
Residual Gas Analyzer	1E _{hex}
Resolver	09 _{hex}

Tabla 6.5. Perfil y Tipo de Dispositivo soportado por DeviceNet y CIP.

Por ejemplo, de acuerdo con la tabla 6.5, un dispositivo con perfil de Fotosensor es identificado en la red DeviceNet a través del código 06Hex. Por tanto, todo fabricante de dispositivos fotosensores podrá utilizar este perfil para sus propios dispositivos. Por otra parte, los fabricantes pueden crear sus propios perfiles para sus dispositivos DeviceNet. Para ello deben seguir las pautas entregadas por ODVA, a través de las especificaciones DeviceNet.

6.5.4 Configuración

La configuración de los dispositivos DeviceNet se lleva a cabo a través de los Electronics Data Sheets, EDS, que traducido al español significa Hoja de Datos Electrónica (ver sección 4.10). Todo dispositivo DeviceNet debe incluir su EDS para configuración. Por lo tanto, los softwares de configuración se basan en la información contenida en estos EDS para la configuración de los parámetros del dispositivo. Esto es transparente al usuario. En los anexos 1 y 2, se entrega el código de dos archivos EDS de dos dispositivos DeviceNet.

CAPÍTULO VII

DISPOSITIVOS PARA LA RED DE CAMPO DEVICENET

7.1 Introducción

Dado que la red de campo DeviceNet ha sido diseñada como una red para los primeros niveles de la jerarquía de automatización, ésta permite la conexión entre dispositivos industriales simples, tales como sensores y actuadores, y los interfasa con dispositivos de mayor nivel, como lo son los PLCs y computadores personales (PC). Por ello, es necesario estar familiarizado con los posibles dispositivos que se utilizarán en un proyecto con la red DeviceNet. Por ello es que el presente capítulo entrega una descripción general de los dispositivos que integran a una red DeviceNet.

7.2 Clasificación de los componentes

Los componentes de la red se clasifican según su función. En la figura 7.1 se muestra la representación de una red DeviceNet con la clasificación de los componentes de red.

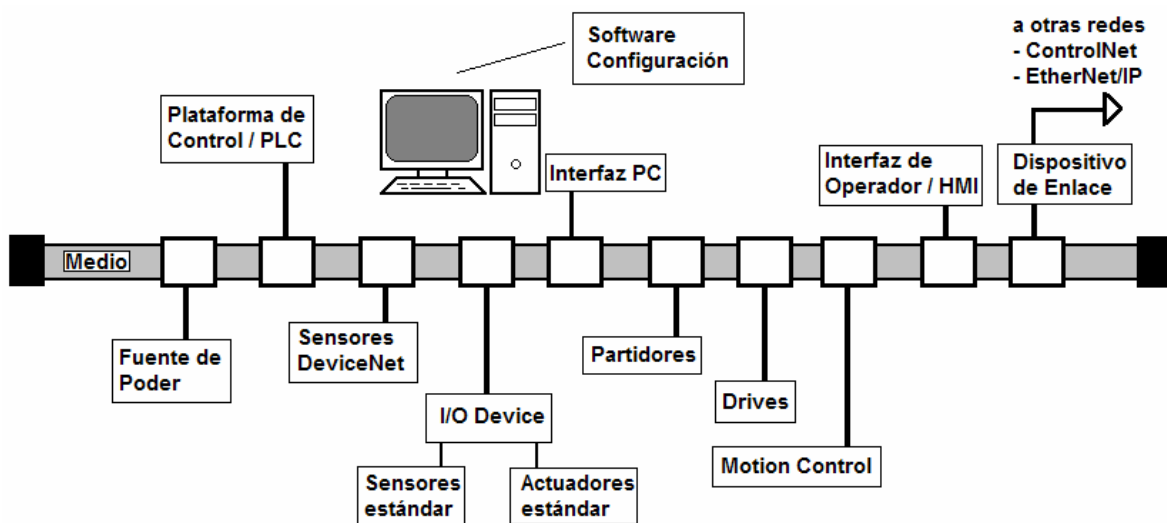


Figura 7.1. Componentes de la red DeviceNet.

Como se observa en la figura anterior, el equipamiento que compone a una red DeviceNet se puede clasificar en las siguientes clases:

- Plataforma de Control*
- Medio*
- Sensores DeviceNet
- Software de Configuración*
- Interfaz para PC*
- I/O Device
- Fuente de Poder*
- Interfaz de Operador
- Partidores
- Drives
- Motion Control
- Dispositivo de Enlace

* *dispositivos imprescindibles*

Una red DeviceNet se compondrá de las clases de dispositivos listadas. Las clases marcadas con “*” son imprescindibles en toda red DeviceNet. El resto de las clases dependerá de la aplicación particular para la cual se ha de emplear.

7.3 Plataforma de control

La plataforma de control para DeviceNet se compone de un Controlador Lógico Programable, o PLC, junto a un módulo de comunicación para soportar comunicación a una red DeviceNet. En la figura 7.2 se muestra un esquema que ilustra una plataforma de control para DeviceNet.

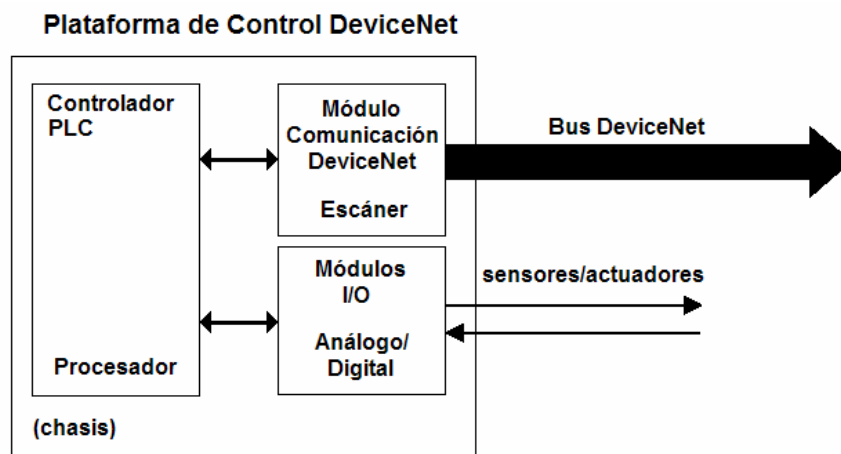


Figura 7.2. Esquema de una plataforma de control DeviceNet.

Existe una amplia gama de plataformas de control disponibles para DeviceNet, entre las que se pueden nombrar:

- ControlLogix
- CompactLogix
- PLC-5
- FlexLogix
- SLC 500

En la figura 7.3 se muestra el aspecto físico de diferentes plataformas de control para DeviceNet.

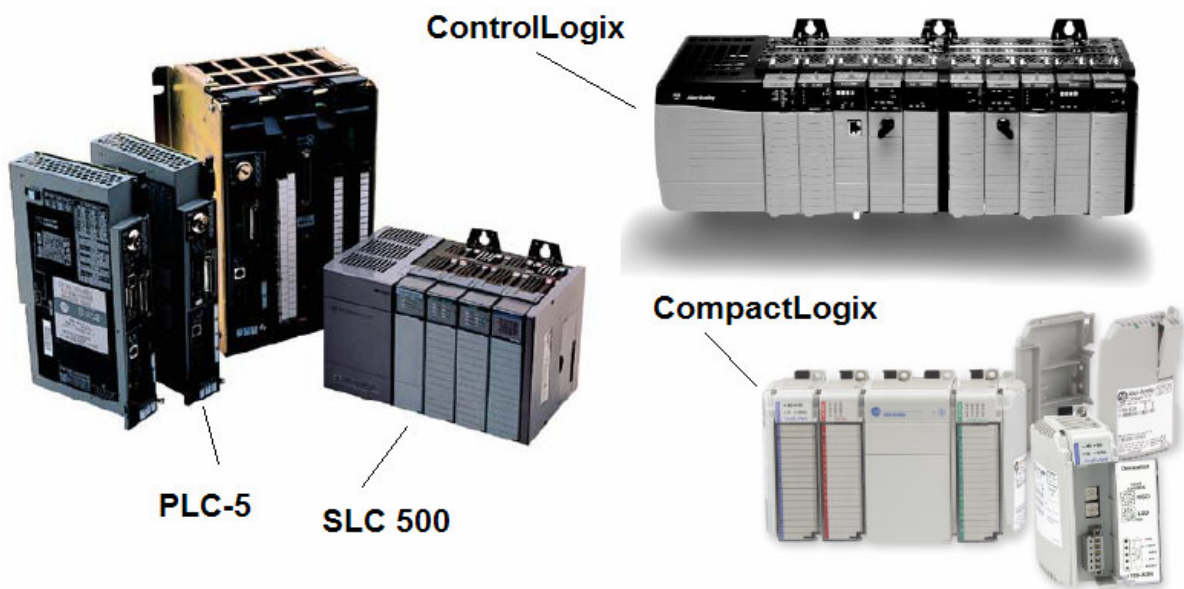


Figura 7.3. Aspecto físico de distintas plataformas de control.

Para ilustrar, a continuación se hará una breve descripción de los componentes de la plataforma ControlLogix para DeviceNet.

7.3.1 Plataforma ControlLogix

a) Controlador de la serie 1756

En la figura 7.4 se muestra el aspecto físico del controlador de la serie 1756 de Allen-Bradley para la plataforma ControlLogix



Figura 7.4. Controlador de la serie 1756 ControlLogix de Allen-Bradley.

Las principales características de este controlador son:

- **Memoria de Usuario:** desde 750Kbytes hasta 8Mbytes.
- **Puertos de comunicación:** 1 o 2 puertos serie RS-232
- **Opciones de comunicación:** DeviceNet, ControlNet, EtherNet/IP, DH+, RIO, serial, Modbus, DH-485, SynchLink
- **I/O nativo:** 1756 ControlLogix I/O
- **Lenguajes de programación:** Ladder, texto estructurado, bloque de funciones y carta de funciones secuenciales.

b) Módulo de comunicación/escáner 1756-DNB para DeviceNet

Este dispositivo permite la comunicación entre un PLC ControlLogix y una red DeviceNet. Permite también, en su función de escáner, operar como maestro de una red DeviceNet. En la figura 7.5 se muestra el módulo de comunicación DeviceNet para ControlLogix.



Figura 7.5. Escáner DeviceNet.

A continuación se enlistan las principales características de este dispositivo.

- Mensajería Explícita Peer-to-peer.
- Reemplazo automático de dispositivo (ADR).
- Tasas de transmisión: 125, 250 y 500 kbps.
- Modos Master/Slave
- Métodos de comunicación:
 - strobe
 - poll
 - cambio de estado
 - cíclico.

c) Chasis para ControlLogix

El chasis de la plataforma de control entrega los slots (o ranuras) para que sean insertadas el PLC, el módulo de comunicación y otros módulos de entrada/salida (I/O). El chasis lleva incorporada la fuente de poder que energiza a los módulos a través de los slots. En la figura 7.6 se muestra el chasis ControlLogix.

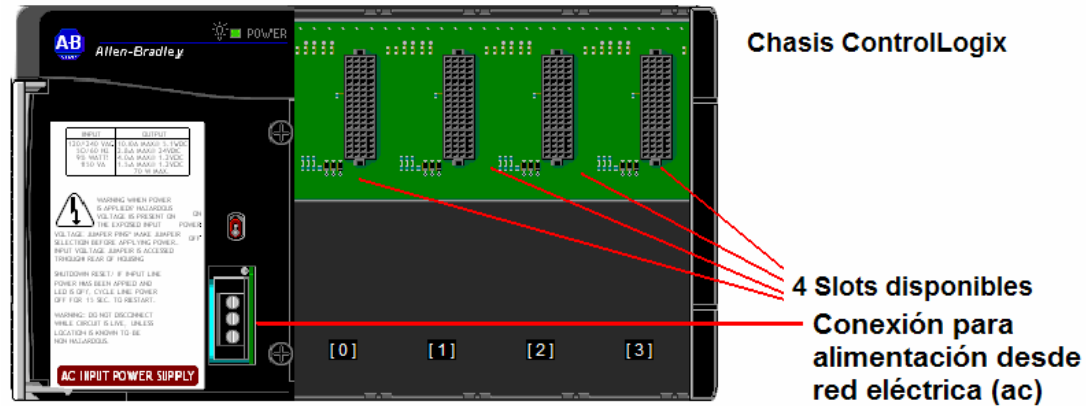


Figura 7.6. Chasis ControlLogix.

En la figura 7.7 se muestra la instalación del controlador junto al módulo de comunicación en un chasis ControlLogix. El chasis incluye la fuente de poder de 24VDC para DeviceNet.



(a)



(b)

Figura 7.7. Instalación de las tarjetas en el chasis ControlLogix (a) vista general; (b) acercamiento a las tarjetas.

7.4 Medio

Como se describió en el estudio de la capa física de DeviceNet, se definen 3 medios que son Round-Thick, Round-Thin y Plano-KwikLink. En la figura 7.8 se muestra el aspecto físico de componentes de los tres medios.



Figura 7.8. Aspecto físico de los componentes de los distintos medios.

Round-Thick: Este sistema de medio permite la máxima longitud de red para la línea troncal, la cual es de 500m. Es el medio original de DeviceNet.

Round-Thin: Este sistema de medio se utiliza en redes que no tengan una línea troncal de más de 100m, pero permite una instalación más compacta y económica para algunas instalaciones. Se utiliza como alternativa para el medio Round-Thick.

Plano-KwikLink: El medio plano-kwiklink permite la simplificación y ahorro en el diseño y montaje de la red, pero aún tiene limitaciones en relación con el medio Round-Thick respecto de la longitud de la red. Se distinguen 2 clases según su capacidad de conducción de corriente: Clase 1, hasta 4A y Clase 2, hasta 8A.

A continuación se muestra la configuración típica de una red DeviceNet implementada con cada uno de los medios.

7.4.1 Configuración Típica de red DeviceNet con medio Round-Thick

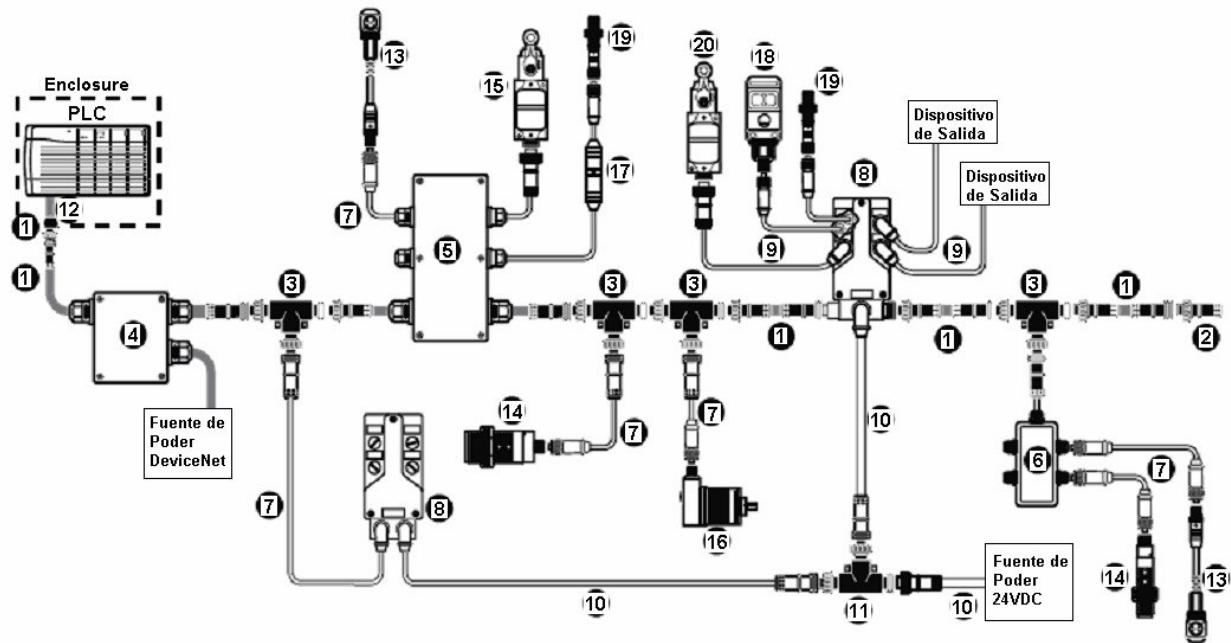


Figura 7.9. Configuración típica de una red DeviceNet con medio Round-Thick.

Claves de los dispositivos para el medio Round-Thick			
1	Thick trunk cable	11	Aux power tee
2	Terminator	12	Open terminator
3	T-port	13	DeviceNet photoelectric sensor
4	Power tap	14	DeviceNet inductive proximity switch
5	DeviceBox	15	DeviceNet limit switch
6	DevicePort	16	DeviceNet encoder
7	Thin drop cable	17	DeviceLink
8	ArmorBlock MaXum	18	Standard photoelectric sensor
9	ArmorBlock cordsets	19	Standard proximity switch
10	Aux power cable	20	Standard limit switch

Tabla 7.1. Claves de dispositivos de la figura 7.9.

7.5 Sensores DeviceNet

Son sensores con capacidades de comunicación directa con en una red DeviceNet. Los sensores DeviceNet cuentan como un nodo para la red. Pueden soportar uno o más de los métodos de comunicación establecidos para la red como son Poll, Strobe, Cambio de Estado y Cíclico. Por lo tanto, debe tomarse en cuenta este factor al escoger el sensor.

Los sensores DeviceNet, son llamados también “dispositivos inteligentes” debido a que poseen características especiales tales como autodiagnóstico, auto-detección de tasa de transmisión de la red (autobaud) y reemplazo automático de dispositivo (ADR). En la figura 7.12 se muestra el aspecto físico de típicos sensores DeviceNet.



Fotosensor



Sensor de Proximidad Inductivo



Final de carrera (Limit-Switch)

Figura 7.12. Sensores “inteligentes” DeviceNet.

En la tabla 7.4, tomada de un catálogo de dispositivos para DeviceNet, se muestran las especificaciones de estos dispositivos.

Aplicación	Sensor DeviceNet	Especificaciones
harsh environments such as breweries and food processing plants, with temperatures up to 70° C, or high-pressure washdowns and many harsh solvents	SmartSight 9000 Sensors 	• NEMA 6P, IP67, 1200 psi washdown rating • choose from micro or mini style connectors or an attached 2m CPE drop cable • I/O change-of-state and strobe messaging
material handling and packaging industries where shorter sensing distances are required	RightSight Photoelectric Sensors 	• NEMA 4X, IP67 1200 psi washdown rating • I/O change-of-state and strobe messaging
potentially corrosive environments such as metalworking, food processing and material handling industries	Bulletin 871TM Inductive Proximity Sensors 	• 1200 psi (8,270 kPa) washdown rating • available with mini or micro QD, or 2m CPE jacketed cable • I/O change-of-state and strobe messaging
applications which require heavy-duty pilot ratings, a high degree of versatility and rugged, oil-tight construction. The sensors's patented 3-way shaft seal design offers superior fluid resistance.	Bulletin 802DN Limit Switches 	• NEMA 13 and IP65 (IEC529) rating • available with mini or micro QD, or 2m CPE jacketed cable • I/O change-of-state and strobe messaging, and Devicenet connectivity is built right in.
applications which require direct connection to DeviceNet for advanced functionality with reduced wiring cost	Bulletin 842D Absolute Multi-function Rotary Encoders 	• NEMA 4, 13 and IP66 (IEC 529) rating • five-pin micro quick disconnect • I/O change-of-state and strobe messaging

Tabla 7.4 Sensores DeviceNet junto a sus características.

7.6 Herramientas de Software

Las principales herramientas de software para la red DeviceNet son:

- **Architecture Builder**
- **RSLogix5000**
- **RSNetworkx**
- **RSLink**

7.6.1 Architecture Builder

Este software es una herramienta para el diseño de redes que implementan el protocolo CIP, es decir, DeviceNet, ControlNet y EtherNet/IP. A través de esta herramienta es posible diseñar una red DeviceNet, tomando en cuenta el medio, los dispositivos y otros factores asociados. En la figura 7.13 se muestra una captura de pantalla de este software.

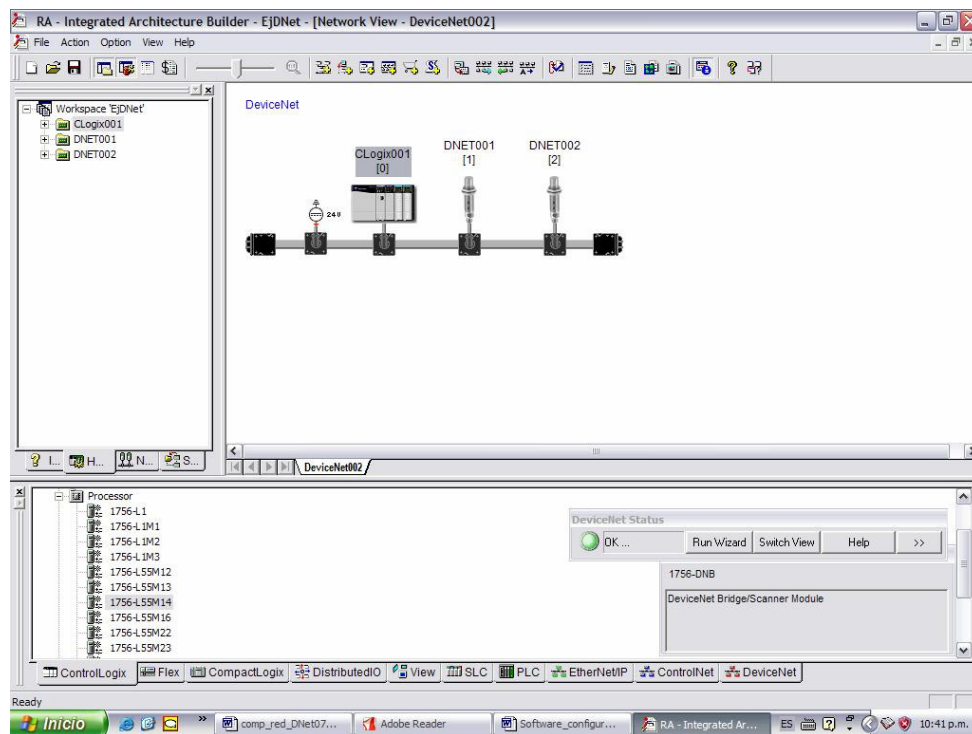


Figura 7.13. Software para el diseño de redes de la familia CIP Architecture Builder.

El programa Architecture Builder permite además la generación de informes y listas de cotización.

7.6.2 RSLogix5000

El software RSLogix5000 es la herramienta utilizada en la mantención de la plataforma de control. Este permite realizar las siguientes operaciones:

- Desarrollar los programas de aplicación para el PLC en alguno de los siguientes 4 lenguajes:
 - Ladder
 - texto estructurado
 - bloque de funciones
 - carta de funciones secuenciales
- Descargar (Download) los programas desde el disco duro del PC hacia el PLC de la plataforma de control, e. g. ControlLogix.
- Subir (Upload) un programa existente en un procesador de la plataforma de control, tal como ControlLogix, al disco duro del PC.
- Revisar y modificar los programas de ladder instalados en el procesador.
- Forzar entradas y salidas.
- Acceder a la configuración y status de operación de todos los módulos conectados al sistema.
- Examinar la tabla de datos y las estructuras de datos definidas para la aplicación.
- Diagnóstico de fallas del procesador.

En la figura 7.14 se muestra una captura de pantalla del programa RSLogix 5000.

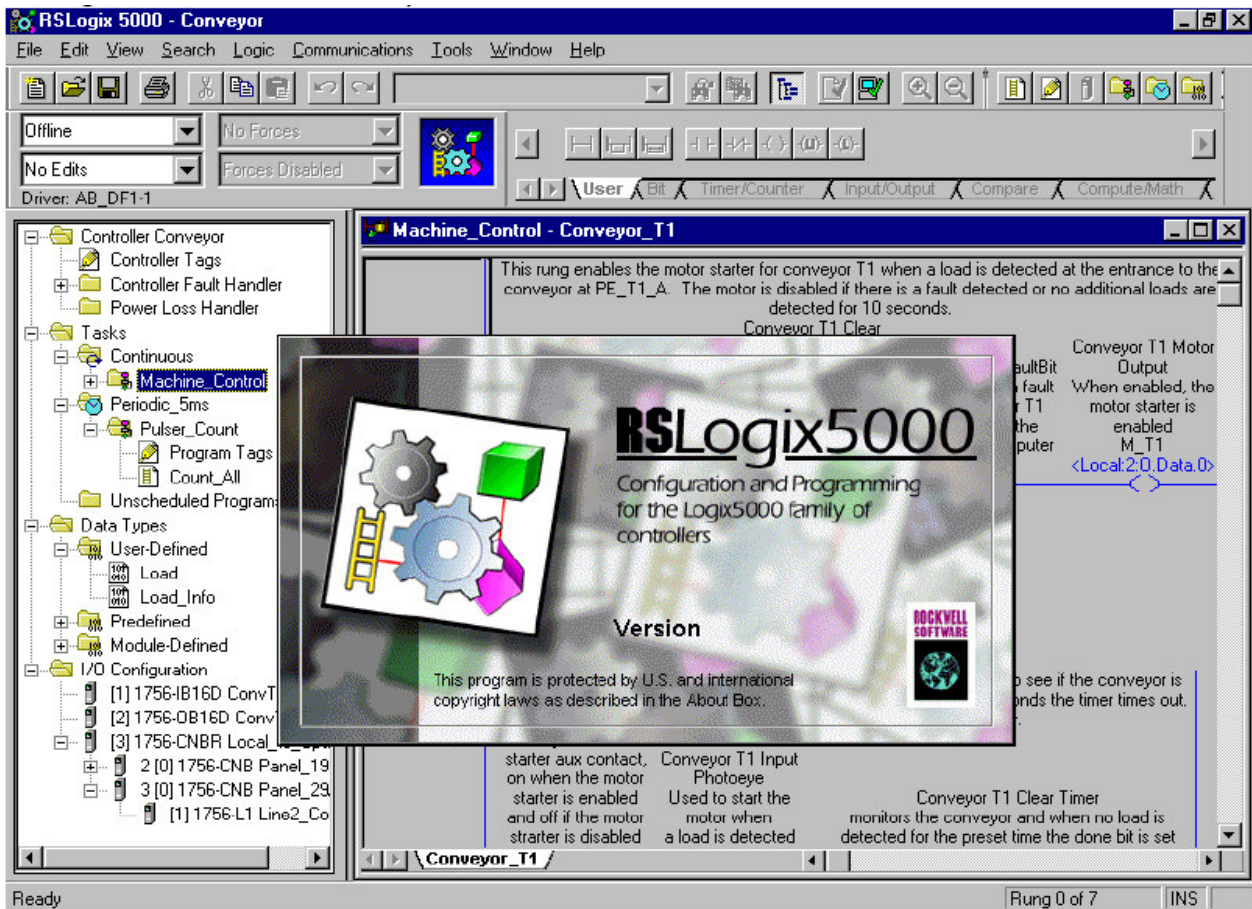


Figura 7.14. Pantalla de inicio del programa RSLogix5000 para la configuración de la plataforma de control.

La conexión entre el PC en el cual corre el programa RSLogix 5000 y el PLC de la plataforma de control, se efectúa por defecto vía RS-232. Pero, además, esta se puede efectuar a través de los módulos de comunicación de la plataforma de control, con el soporte del servidor de comunicación RSLinx (ver sección 7.8).

Por último, el PC anfitrión del programa RSLogix5000 puede ser un nodo más en la red con la conexión de una interfaz de comunicación (ver sección 7.9).

7.7 RSNetworx para DeviceNet

Este software es la herramienta de configuración para una red DeviceNet. Está desarrollado para operar en la plataforma Windows. A través de este, se crea una representación gráfica de la red, agregando y/o quitando dispositivos, y se configuran todos sus parámetros y características.

Todos los dispositivos conectados a la red DeviceNet son configurados con RSNetWorx. Entre las características de configuración se pueden destacar:

- Nombre y descripción del dispositivo.
- Dirección de nodo.
- Autobaud.
- Estado de falla.
- Método de comunicación: strobe, poll, cambio de estado y cíclico.
- ADR (reemplazo automático de dispositivo).

En la figura 7.15 se muestra una captura de pantalla de este programa.

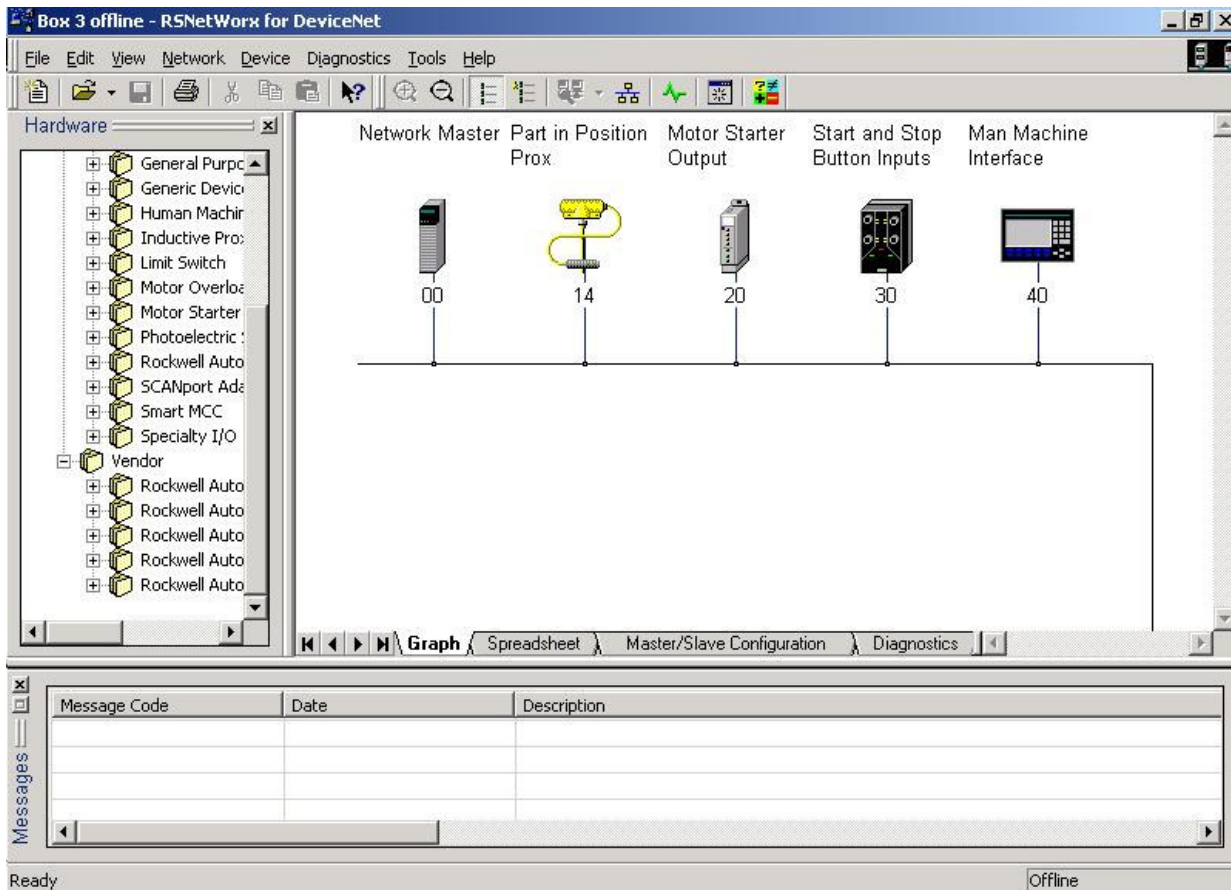


Figura 7.15. Captura de pantalla del programa RSNetWorx para DeviceNet.

La comunicación entre el PC en el cual corre el programa RSNetWorx y la red DeviceNet se realiza por los siguientes medios:

- Por puerto de red si la plataforma de control posee un módulo de comunicación EtherNet/IP.
- Por la conexión de una interfaz de comunicación en el PC anfitrión del programa RSNetWorx, con lo cual, el PC puede ser configurado como un nodo más en la red.

Por último, la conexión entre el programa RSNetWorx y la red DeviceNet es establecida por el servidor de comunicación RSLinx (ver sección 7.8).

7.8 RSLinx

Este software es un servidor de comunicación que permite el enlace entre una red de la familia CIP, y los softwares de aplicación en entorno Windows. De esta forma, una red DeviceNet es posible ser configurada por RSNetWorx, así como su plataforma de control por RSLogix, dado que el software RSLinx opera para soportar la comunicación.

Entre otras funcionalidades que entrega este programa, se pueden mencionar:

- Como servidor de comunicación, establece la comunicación entre la red y los programas RSLogix5000 y RSNetWorx.
- Subir (Upload) un programa en lenguaje Ladder desde un PLC y modificarlo.
- Navegar por los dispositivos de la red y monitorear su estado.

En la figura 7.16 se muestra una captura de pantalla de este programa.

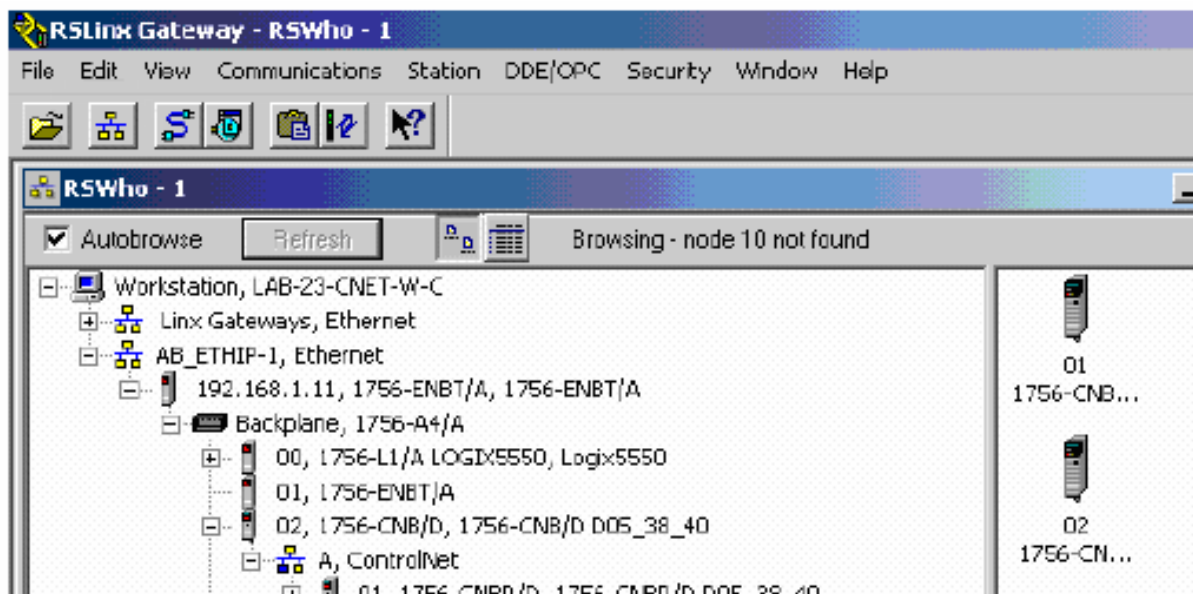
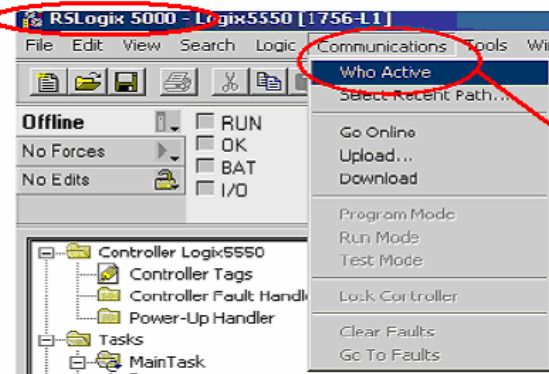


Figura 7.16. Captura de pantalla del programa RSLinx.

En programa RSLinx opera automáticamente desde los programas de configuración para la red DeviceNet, RSLogix 5000 y RSNetWorx. En la figura 7.17 se muestra cómo se establece una comunicación desde el programa RSLogix 5000 hacia la red DeviceNet, en donde RSLinx efectúa su tarea de enlazar en forma automática.

Desde el software RSLogix5000, se pueden ver los dispositivos conectados a la plataforma de control en una red CIP.



RSLinx es quien establece automáticamente la comunicación entre la red y el programa de plataforma Windows.

Figura 7.17. RSLinx establece la comunicación entre un programa de plataforma Windows y la red.

7.9 Interfaces para PC

Permiten la conexión a la red DeviceNet del PC anfitrión de los softwares de programación y configuración de la red, como un nodo más. Los tipos de interfaces para PC para la red son tarjetas PCMCIA para Notebooks, tarjetas PCI para PC de escritorio e interfaz para PC vía puerto RS-232. En la figura 7.18 se muestra el aspecto físico de estos tres tipos de interfaces.

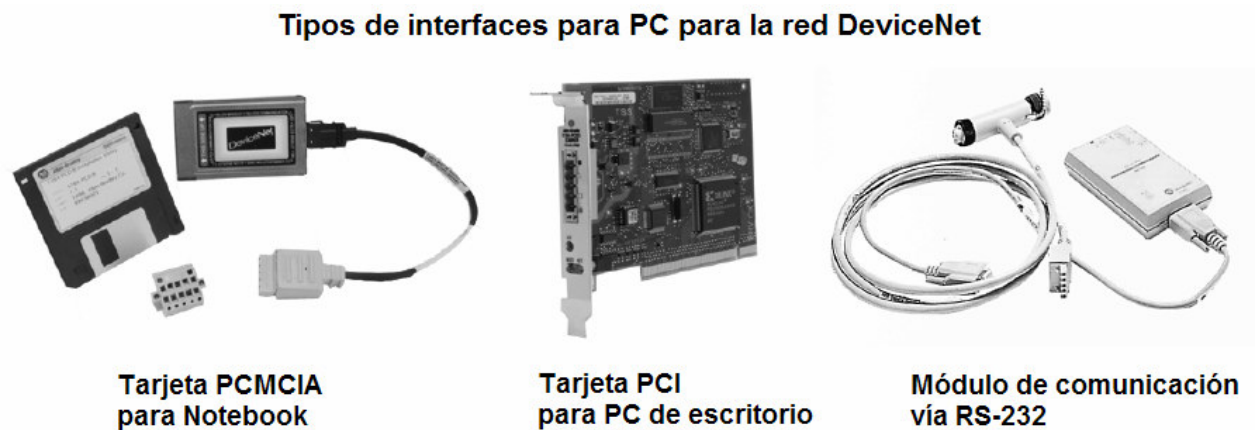


Figura 7.18. Interfaces para PC para DeviceNet.

A través de estas tarjetas, es posible programar, monitorear y configurar la red DeviceNet desde los programas RSLogix5000 y RSNetWorx. Estas tareas pertenecen al tipo de mensajería explícita (Explicit I/O). Sin embargo, no aplican las características de mensajería implícita (datos críticos).

A continuación se listan las principales características de estos dispositivos.

- Sólo mensajería explícita (configuración, programación y monitoreo).
- Tasas de transmisión: 125, 250 y 500kbps.
- Autobaud.

7.10 Dispositivo de Entrada/Salida o I/O Device

Los Dispositivos de entrada/salida, o I/O Device, son dispositivos que permiten conectar a la red DeviceNet, dispositivos de entrada salida, tales como sensores y actuadores, de tipo estándar. Un I/O Device es considerado un nodo dentro de la red, por lo cual debe asignársele una dirección de red. A través de este se conectan varios puntos I/O.

Para el caso de una entrada para un I/O Device, éste procesa la señal proveniente de un sensor estándar, la digitaliza y le otorga el formato de trama DeviceNet para transmitirla a la red como un nodo más. Dado que un I/O Device soporta la conexión de varios puntos I/O simultáneamente, en la trama DeviceNet va contenida la información para cada punto.

Para ilustrar el funcionamiento de los I/O Device, se tomará como ejemplo el módulo de 2 entradas analógicas de voltaje modelo 1734-IE2V de Allen-Bradley. Este módulo posee un riel DIN para montaje en un chasis. En la figura 7.19 se muestra el formato físico, así como el esquema de conexionado para este módulo.

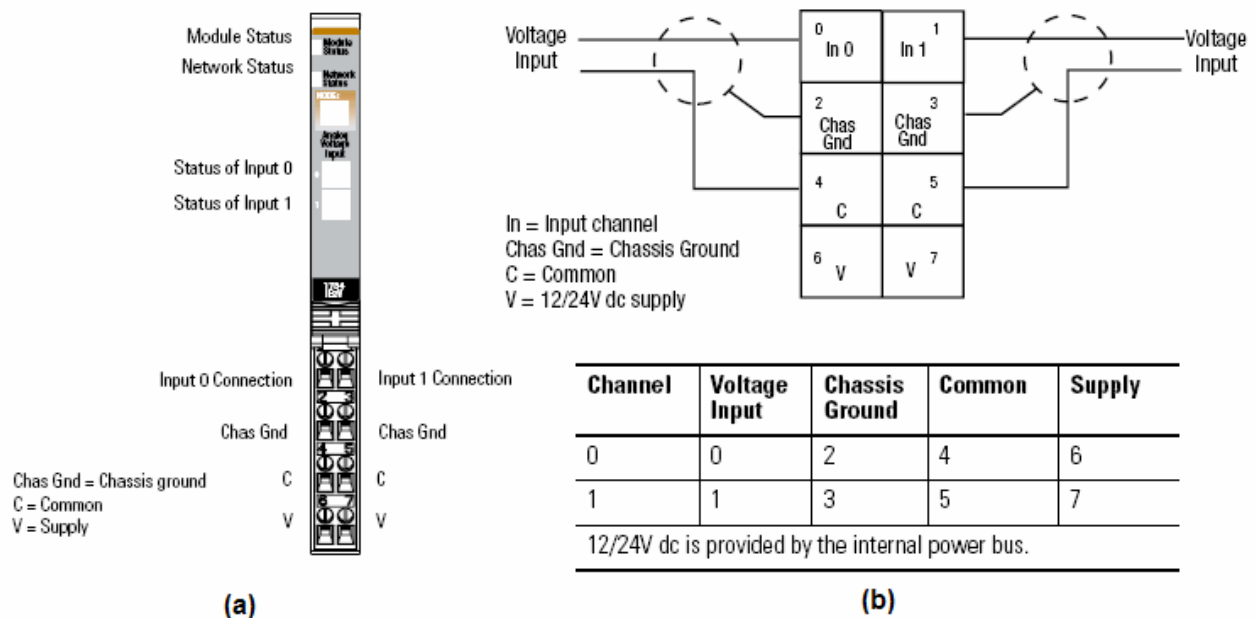


Figura 7.19. Módulo para la entrada de 2 señales de voltaje analógicas de 0-10V. (a)

Construcción física; (b) Esquema de conexionado.

Se pueden observar en la figura 7.19 las dos entradas de voltaje. La señal de voltaje de entrada, de acuerdo con su hoja de especificación, para este dispositivo puede tener un rango entre 0-10V. Esta señal puede ser provista desde un sensor estándar.

El módulo digitaliza esta señal, genera un paquete de datos DeviceNet (CAN) de 6 Bytes y lo transmite a la red, como cualquier otro nodo productor. En la figura 7.20 se muestra el paquete de datos generado por este módulo.

	15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
Produces (scanner Rx)	Input Channel 0 - High Byte								Input Channel 0 - Low Byte							
	Input Channel 1 - High Byte								Input Channel 1 - Low Byte							
	Status Byte for Channel 1								Status Byte for Channel 0							
	O R	U R	H H A	L L A	H A	L A	C M	C F	O R	U R	H H A	L L A	H A	L A	C M	C F
Consumes (scanner Tx)	No consumed data															

Where: CF = Channel Fault status; 0 = no error, 1 = fault
CM = Calibration Mode; 0 = normal, 1 = calibration mode
LA = Low Alarm; 0 = no error, 1 = fault
HA = High Alarm; 0 = no error, 1 = fault
LLA = Low/Low Alarm; 0 = no error, 1 = fault
HHA = High/High Alarm; 0 = no error, 1 = fault
UR = Underrange; 0 = no error, 1 = fault
OR = Overrange; 0 = no error, 1 = fault

Figura 7.20. Paquete de datos de 6 bytes generado por el módulo de entradas analógicas.

Puede observarse en la figura 7.20 que los dos primeros bytes corresponden a la entrada analógica 1, mientras que los siguientes 2 bytes para la segunda entrada analógica. Los siguientes dos bytes, entregan información del estado de los canales 1 y 2 respectivamente.

Una amplia gama de dispositivos para entrada/salida están disponibles para la red DeviceNet. En la figura 7.21 se muestra la construcción física de otro dispositivo típico para

entrada/salida DeviceNet. Corresponde a un dispositivo modelo 1791D CompactBlock I/O de la marca Allen-Bradley.



Figura 7.21. Típico dispositivo I/O para la conexión de sensores y actuadores estándar a la red DeviceNet.

Se puede observar en la figura el conector DeviceNet, así como la regleta de conexiones para las entradas y salidas. Además, se muestra que la configuración de la dirección de nodo se puede efectuar mecánicamente a través de dos switch incorporados.

Otras características de este dispositivo se enlistan a continuación.

- Métodos de comunicación: poll, cambio de estado y cíclico.
- Autodetección de tasa de transmisión (autobaud).
- Reemplazo de automático de dispositivo (ADR).

7.11 Fuente de Poder

Como se ha mencionado anteriormente, la red DeviceNet entrega la alimentación a dispositivos simples, a través de la misma red. El voltaje utilizado por DeviceNet es de 24VDC. La fuente de poder se conecta a los hilos conductores V- y V+ de la línea troncal para suministrar la energía. Una fuente de poder para alimentar la red debe cumplir con los requerimientos dados por la especificación DeviceNet.

En la figura 7.22 se muestra una fuente de poder diseñada específicamente para DeviceNet.

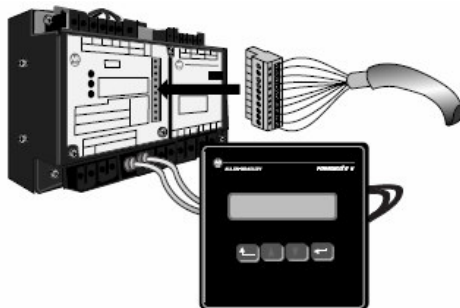


Fuente DeviceNet

- 24VDC/5,25A a 60°
- Entrada 110V/60Hz y 220/50Hz
- Indicación por Led.
- Conector estándar 5 pines DeviceNet.

Figura 7.22. Fuente de poder DeviceNet de 24VDC.

Además, existen dispositivos denominados Powermonitor del fabricante Allen-Bradley, que incluyen una fuente de poder junto a un sistema de monitoreo del estado de energía de la red. Estos dispositivos poseen comunicación con la red DeviceNet y son configurados como un nodo más. En la figura 7.23 se muestra un dispositivo Powermonitor junto a sus principales características.



Powermonitor II

- 24VDC/5A
- Entrada 110V/60Hz y 220/50Hz
- Métodos de Comunicación: poll, cambio de estado, cíclico.
- Autobaud
- Software de configuración: RSPower32.

Figura 7.23. Powermonitor junto a sus principales características.

7.12 Interfaz de Operador

Las interfaces de operador entregan al usuario la capacidad de monitorear el proceso en forma visual a través de indicadores o pantallas que entregan información de lo que está ocurriendo en la planta. Estas pueden ser simples luces indicadores o completos diagramas del proceso plasmados en monitores adaptados para las condiciones industriales. Las interfaces de operador son llamadas también Interfaces Humano Máquina, o HMI (Human- Interface-Machine).

Las interfaces de operador DeviceNet tienen la capacidad de conectarse directamente a la red y deben ser configurados como un nodo más en la red.

En la figura 7.24 se muestra el modelo de pantalla de visualización HMI Panel View de Allen-Bradley, junto a sus especificaciones.



PanelView

Pantallas gráficas

- 5,5", 6", 9", 10" y 14"
- Color y Monocromo
- Teclado, Táctil, ambos

Software de programación en Windows

- Panel Builder

Comunicación por

- Cambio de estado, cíclico, Poll

Gestión de alarmas integrada

Sustituto de elementos de panel

- pulsadores, pilotos, etc.
- máxima flexibilidad para realizar cambios frecuentes

Figura 7.24. Interfaz Humano Máquina gráfica PanelView para DeviceNet.

Existe una variedad de dispositivos interfaces de operador con capacidades de comunicación DeviceNet. En las figuras 7.25, 7.26 y 7.27 se muestran otros tipos de interfaces de operador del fabricante Allen-Bradley, junto a sus principales características.



Visualizador InView

- InView Configuration Software (Windows)
- Mensajería I/O
- Polled, Cambio de Estado, Cíclico
- Matriz de Led
- Rojo o Tricolor

Figura 7.25. Visualizador de matriz de Led para DeviceNet.



Balizas luminosas 855T

- Combinación de hasta 5 módulos luminosos/sonido
- Incandescente, LED, 6 colores
- Módulos de sonido de 1 y 2 tonos
- Montaje horizontal o vertical
- Alimentación y señal desde red
- No es necesario cableado adicional
- 16 modos de operación
- Manual, destellos, etc.
- Autobaud, temporización
- I/O polling

Figura 7.26. Baliza luminosa para DeviceNet.



RediSTATION

- Botonera con 2 pulsadores y una lámpara
- Configurable por el usuario
- Elección de la combinación de pulsadores/lámpara
- Placa suministrable individualmente
- 4 entradas digitales
- 2 salidas digitales
- IP66, 0-55°C
- Alimentación desde red
- Diagnósticos
- Polled, Strobe, Cambio de Estado, Cíclico

Figura 7.27. Botonera para ambiente industrial DeviceNet.

7.13 Partidores y protección de Motores

La red DeviceNet soporta la conexión de dispositivos partidores de motores, relays de protección, todos con mecanismos de comunicación de red DeviceNet. En la tabla 7.5 se muestran algunos ejemplos de esta clase de dispositivos junto a sus características.

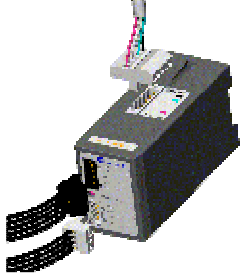
Partidor Auxiliar DeviceNet DSA	Características
	• Módulo de 2/4 entradas y 2/1 salidas para control de contactores ○ Las E/S múltiples permiten el control de arrancadores, inversores y estrella-triángulo • Alimentación desde DeviceNet • Múltiples opciones ○ entradas cc/ca ○ salidas relé/transistor • Comunicación poll, Cambio de estado y cíclico
Partidor Suave DeviceNet SMC	Características
	• Programación, monitorización y control desde DeviceNet • Diagnósticos: inversión de fases, desequilibrio de tensión, limitación de arranques. • Comunicación Poll, Strobe, cambio de estado y cíclico. • Múltiples características de arranque y paro • Protección electrónica integrada contra la sobrecarga del motor • Se puede programar localmente mediante el teclado y LCD integrados o remotamente con el software RSNetworkx
Relay de Protección E3	Características
	• Protección ○ Protección térmica ○ Pérdida de fase ○ Falla a tierra ○ Atasco - Stall (overload al arranque) ○ Atasco - Jam (overload una vez partido) ○ Baja carga ○ Desbalance de corriente • Comunicación poll, cambio de estado y cíclico.

Tabla 7.5. Dispositivos Partidores y Relay de Protección para motores en una red DeviceNet.

7.14 Drives

La red DeviceNet soporta la conexión de drives o variadores de frecuencia para la regulación de la velocidad en motores de corriente alterna. Pueden tener integrada la comunicación DeviceNet o debe incorporárseles un módulo de comunicación. En la figura 7.28 se muestra el aspecto físico típico de estos dispositivos.

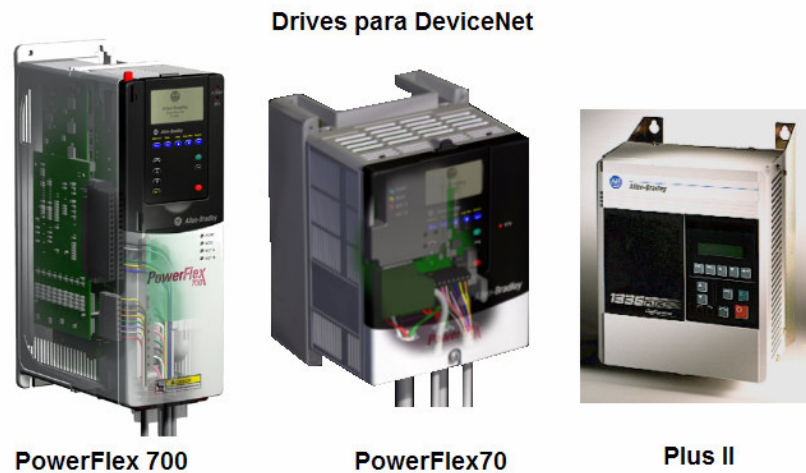


Figura 7.28. Aspecto físico típico de AC Drives para DeviceNet.

En la tabla 7.6 se enlistan las características de estos Drives.

Aplicación	Drive	Especificaciones
versatile, application-configurable features, such as volts-per-hertz or sensorless vector control, in a compact, panel-mount package	PowerFlex 70 AC Drive DeviceNet Communication Adapter	• 0.37 to 37 kW; 0.5 to 50 HP • voltage ratings of 200-240V / 380-480V / 500-600V • Volts/Hertz or Sensorless Vector control
outstanding performance in an easy-to-use drive, the PowerFlex 700 is designed for applications with requirements ranging from the simplest speed control to the most demanding torque control	PowerFlex 700 AC Drive DeviceNet Communication Adapter	• 0.37 to 132 kW, 0.5 to 200 HP • voltage ratings of 200-240V, 380-480V, 500-690V • Volts/Hertz, Sensorless Vector, and Vector Control
excellent speed control and torque performance of a powerful sensorless vector drive an extremely large feature set for any application	1336 PLUS II AC Drive DeviceNet Communication Module	• 0.37 to 448 kW, 0.5 to 600 HP • voltage ratings of 200-240V / 380-480V / 500-600V • Volts/Hertz or Sensorless Vector control

Tabla 7.6. Tipos de drives y sus especificaciones.

7.15 Motion Control

Los dispositivos para Motion Control, o Control de Movimiento, son utilizados para la regulación precisa de motores en aplicaciones de movimiento (e. g. robótica).

En la tabla 7.7 se entrega una lista de dispositivos para Motion Control junto a sus especificaciones.

Aplicación	Dispositivo	Especificaciones
to operate your digital servo drive directly from any compatible control, such as a PLC. This drive, with its powerful indexing feature, is a cost-effective solution for point-to-point applications.	Ultra3000 Digital Servo Drive, 500W Ultra3000 Indexing Drive, 500W Ultra3000 Digital Servo Drive, 1kW Ultra3000 Indexing Drive, 1kW Ultra3000 Digital Servo Drive, 2kW Ultra3000 Indexing Drive, 2kW Ultra3000 Digital Servo Drive, 3kW Ultra3000 Indexing Drive, 3kW Ultra3000 Digital Servo Drive, 7.5kW Ultra3000 Indexing Drive, 7.5kW Ultra3000 Digital Servo Drive, 15kW Ultra3000 Indexing Drive, 15kW 	I/O change-of-state, cyclic, and polling messaging
to operate your digital servo drive directly from any compatible control, such as a PLC. When communicating over the DeviceNet network, the Ultra5000 user program can dynamically adjust to changing commands and parameters from a supervisory PLC or Operator Interface.	Ultra5000 Positioning Drive, 500W Ultra5000 Positioning Drive, 1kW Ultra5000 Positioning Drive, 2kW Ultra5000 DeviceNet Expansion Kit 	I/O change-of-state, cyclic, and polling messaging
a full-featured digital ac, closed-loop servo controller that integrates multi-axis motion control and servo drive functions. One- to four-axis-field configurable.	Bulletin 1394 Digital, AC, Multi-Axis Motion Control System 	I/O change-of-state, cyclic, and polling messaging

Tabla 7.7. Dispositivos para Control de Movimiento en una red DeviceNet.

7.16 Dispositivos de Enlace

Los dispositivos de enlace, también llamados bridges, permiten la conexión de un tipo de red a otro tipo de red, al acomodar el protocolo de comunicación de la primera a uno que sea compatible con la segunda.

La red DeviceNet, dado que implementa el protocolo CIP, se enlaza con las otras redes de la familia de este protocolo, ControlNet y EtherNet/IP, permitiendo una integración de los niveles jerárquicos de comunicación.

7.16.1 Dispositivo de enlace DeviceNet a ControlNet

En la figura 7.29 se muestra el dispositivo de enlace 1788-CN2DN del fabricante Allen-Bradley para conectar DeviceNet con ControlNet.

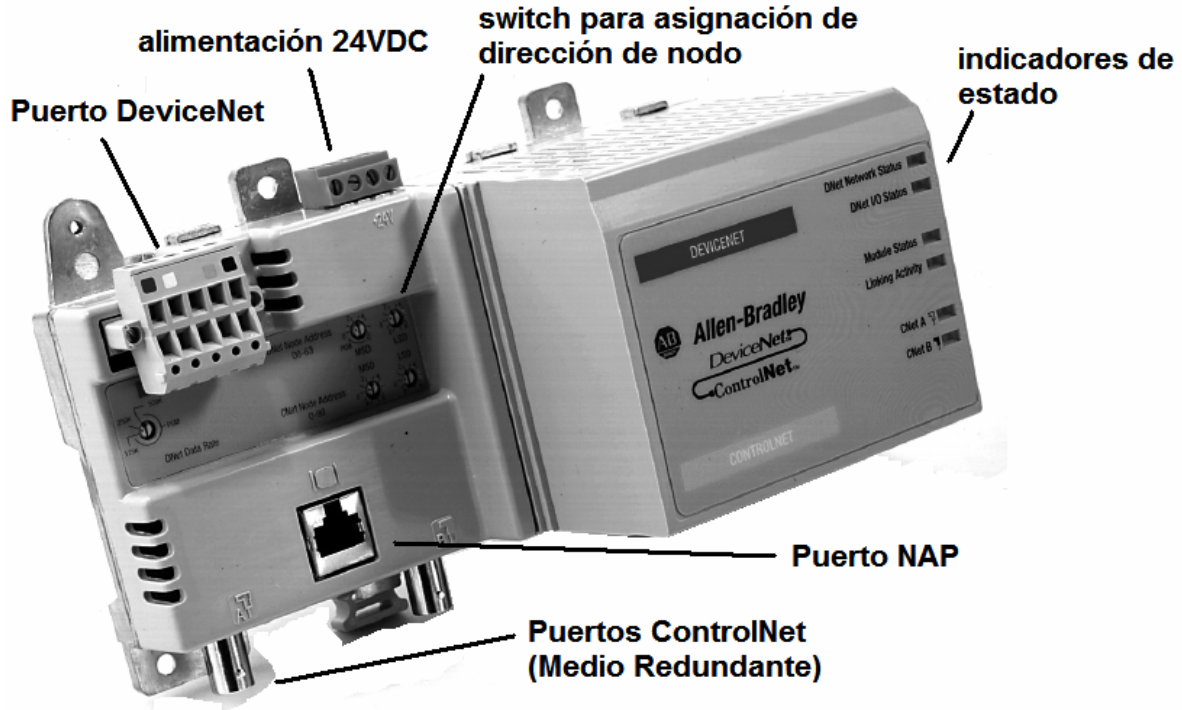


Figura 7.29. Dispositivo de enlace DeviceNet a ControlNet.

Pueden observarse el puerto DeviceNet, junto a los puertos ControlNet para medio redundante opcional. También se puede ver el puerto NAP para la configuración de la red ControlNet a través de un programador manual (ver sección 9.2.5.b2).

Otras características de este dispositivo son:

- Mensajería Explícita
- Tasas de transmisión: 125, 250 y 500 kbps
- Reemplazo automático de dispositivo (ADR)
- Master/Scanner
- Mensajería I/O: strobe, poll, cambio de estado y cíclico.

En la figura 7.30 se muestra la conexión de este dispositivo en la red.

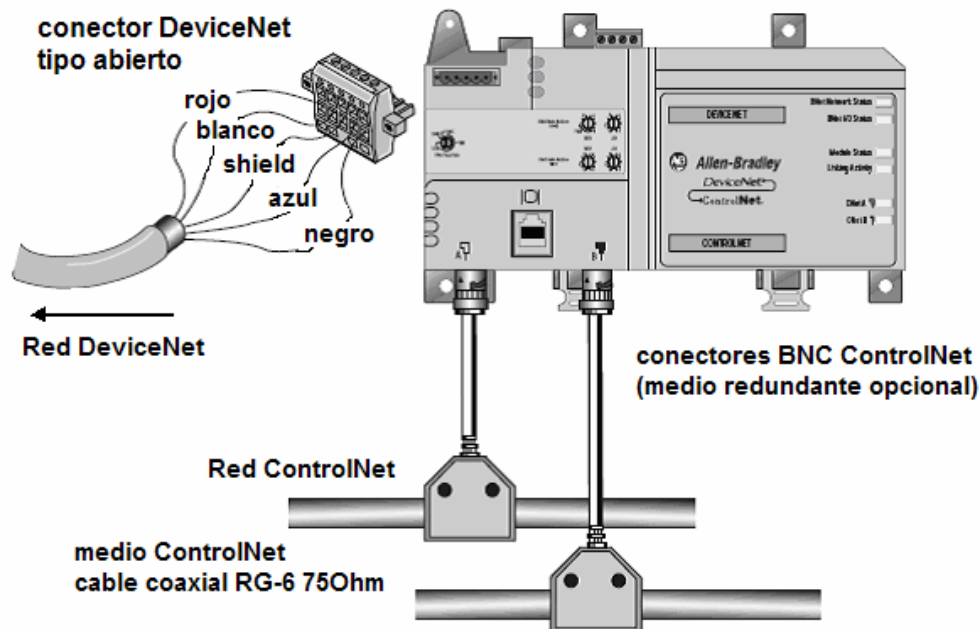


Figura 7.30. Representación conexiones DeviceNet y ControlNet del dispositivo 1788CN2DN de Allen-Bradley.

Una aplicación típica para este dispositivo es utilizar una red ControlNet como el *backbone* (o columna) para enlazar varias redes DeviceNet distribuidas.

7.16.2 Dispositivo de enlace DeviceNet a EtherNet/IP

En la figura 7.31 se muestra el dispositivo de enlace 1788-EN2DN, del fabricante Allen-Bradley, para conectar las redes EtherNet/IP y DeviceNet.

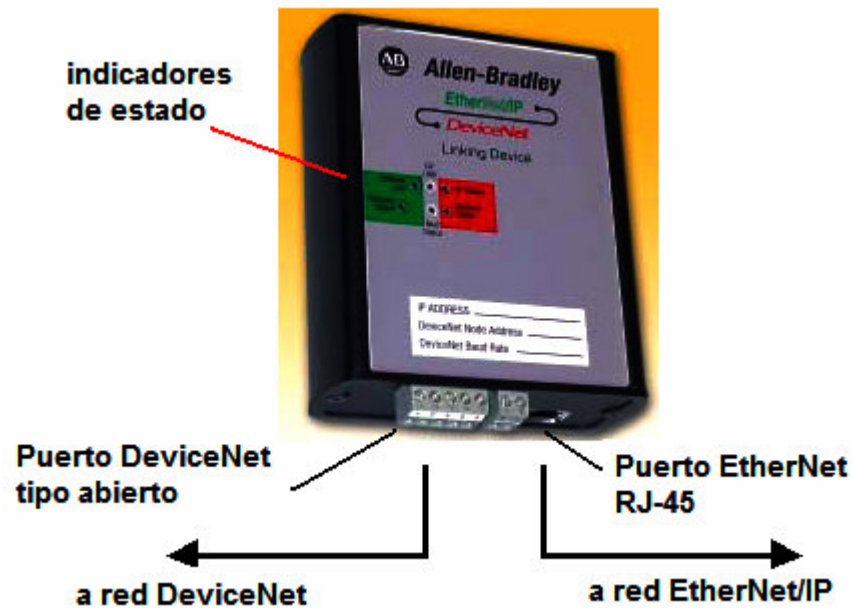


Figura 7.31. Dispositivo para el enlace entre redes EtherNet/IP y DeviceNet.

A continuación se listan las principales características de este dispositivo.

- Mensajería Explícita
- Mensajería I/O: strobe, poll, cambio de estado y cíclico.
- Master/Scanner

Una aplicación para este dispositivo es utilizar una red EtherNet/IP como el *backbone* (o columna) para enlazar varias redes DeviceNet distribuidas.