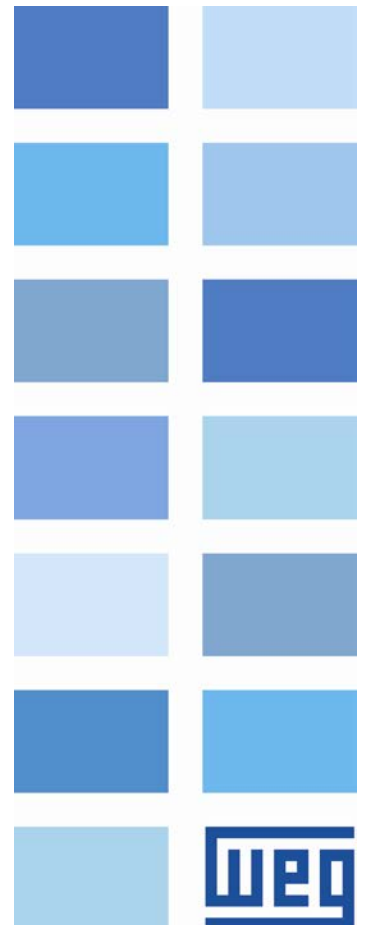


DeviceNet

MW500

Manual del Usuario





Manual del Usuario DeviceNet

Serie: MW500

Idioma: Español

N ° del Documento: 10002252999 / 00

Fecha de la Publicación: 06/2013

CONTENIDOS

CONTENIDOS.....	3
A RESPECTO DEL MANUAL.....	5
ABREVIACIONES Y DEFINICIONES.....	5
REPRESENTACIÓN NUMÉRICA.....	5
DOCUMENTOS.....	5
1 INTRODUCCIÓN A LA COMUNICACIÓN DEVICENET	6
1.1 CAN.....	6
1.1.1 Frame de Datos	6
1.1.2 Frame Remoto.....	6
1.1.3 Acceso a la Red.....	6
1.1.4 Control de Errores.....	6
1.1.5 CAN y DeviceNet.....	7
1.2 CARACTERÍSTICAS DE LA RED DEVICENET.....	7
1.3 MEDIO FÍSICO.....	7
1.3.1 Camada de Enlace de Datos.....	8
1.3.2 Camada de Transporte y Red.....	8
1.3.3 Camada de Aplicación – Protocolo CIP.....	9
1.3.4 Archivo de Configuración EDS.....	9
1.3.5 Modos de Comunicación.....	10
1.3.6 Conjunto de Conexiones Predefinidas Maestro/Esclavo.....	10
2 ACCESORIO PARA COMUNICACIÓN DEVICENET	11
2.1 KIT CAN-01.....	11
2.2 TERMINALES DEL CONECTOR.....	11
2.3 FUENTE DE ALIMENTACIÓN.....	11
2.4 SEÑALIZACIONES.....	11
2.5 CONFIGURACIÓN DEL MÓDULO	12
2.6 ACCESO A LOS PARÁMETROS.....	12
3 INSTALACIÓN DE LA RED DEVICENET.....	13
3.1 TASA DE COMUNICACIÓN.....	13
3.2 DIRECCIÓN EN LA RED DEVICENET	13
3.3 RESISTOR DE TERMINACIÓN	13
3.4 CABLE.....	13
3.5 CONEXIÓN CON LA RED	14
4 PARAMETRIZACIÓN.....	15
4.1 SÍMBOLOS PARA DESCRIPCIÓN DE LAS PROPIEDADES.....	15
P0105 – SELECCIÓN 1ª/2ª RAMPA	15
P0220 – SELECCIÓN FUENTE LOCAL/REMOTO	15
P0221 – SELECCIÓN REFERENCIA LOCAL.....	15
P0222 – SELECCIÓN REFERENCIA REMOTA.....	15
P0223 – SELECCIÓN GIRO LOCAL.....	15
P0224 – SELECCIÓN GIRA/PARA LOCAL.....	15
P0225 – SELECCIÓN JOG LOCAL	15
P0226 – SELECCIÓN GIRO REMOTO.....	15
P0227 – SELECCIÓN GIRA/PARA REMOTO	15
P0228 – SELECCIÓN JOG REMOTO.....	15
P0313 – ACCIÓN PARA ERROR DE COMUNICACIÓN.....	15
P0680 – ESTADO LÓGICO	16

P0681 – VELOCIDAD DEL MOTOR EN 13 BITS.....	17
P0684 – PALABRA DE CONTROL VÍA DEVICENET.....	18
P0685 – REFERENCIA DE VELOCIDAD VÍA DEVICENET	19
P0695 – VALOR PARA LAS SALIDAS DIGITALES.....	19
P0696 – VALOR 1 PARA SALIDAS ANALÓGICAS.....	20
P0697 – VALOR 2 PARA SALIDAS ANALÓGICAS.....	20
P0698 – VALOR 3 PARA SALIDAS ANALÓGICAS.....	20
P0700 – PROTOCOLO CAN.....	21
P0701 – DIRECCIÓN CAN.....	21
P0702 – TASA DE COMUNICACIÓN CAN.....	21
P0703 – RESET DE BUS OFF.....	22
P0705 – ESTADO DEL CONTROLADOR CAN	22
P0706 – CONTADOR DE TELEGRAMAS CAN RECIBIDOS.....	23
P0707 – CONTADOR DE TELEGRAMAS CAN TRANSMITIDOS	23
P0708 – CONTADOR DE ERRORES DE BUS OFF	23
P0709 – CONTADOR DE MENSAJES CAN PERDIDAS.....	23
P0710 – INSTANCIAS DE I/O DEVICENET.....	24
P0711 – LECTURA #3 DEVICENET.....	27
P0712 – LECTURA #4 DEVICENET.....	27
P0713 – LECTURA #5 DEVICENET.....	27
P0714 – LECTURA #6 DEVICENET.....	27
P0715 – ESCRITA #3 DEVICENET.....	27
P0716 – ESCRITA #4 DEVICENET.....	27
P0717 – ESCRITA #5 DEVICENET.....	27
P0718 – ESCRITA #6 DEVICENET.....	27
P0719 – ESTADO DE LA RED DEVICENET	28
P0720 – ESTADO DEL MAESTRO DEVICENET	28
5 CLASES DE OBJETOS SOPORTADAS.....	29
5.1 CLASE IDENTITY (01H).....	29
5.2 CLASE MESSAGE ROUTER (02H).....	29
5.3 CLASE DEVICENET (03H).....	29
5.4 CLASE ASSEMBLY (04H)	29
5.5 CLASE CONNECTION (05H)	30
5.5.1 Instancia 1: Explicit Message.....	30
5.5.2 Instancia 2: Polled.....	30
5.5.3 Instancia 4: Change of State/Cyclic.....	31
5.6 CLASE MOTOR DATA (28H)	31
5.7 CLASE CONTROL SUPERVISOR (29H).....	31
5.8 CLASE AC/DC DRIVE (2AH).....	32
5.9 CLASE ACKNOWLEDGE HANDLER (2BH).....	32
5.10 CLASES ESPECÍFICAS DEL FABRICANTE.....	32
6 FALLAS Y ALARMAS RELACIONADAS CON LA COMUNICACIÓN DEVICENET ..	34
A133/F233 – SIN ALIMENTACIÓN EN EL INTERFAZ CAN.....	34
A134/F234 – BUS OFF	34
A136/F236 – MAESTRO EN IDLE.....	34
A137/F237 – TIMEOUT EN LA CONEXIÓN DEVICENET	35

A RESPECTO DEL MANUAL

Este manual suministra la descripción necesaria para la operación del convertidor de frecuencia MW500 utilizando el protocolo DeviceNet. Este manual debe ser utilizado en conjunto con el manual del usuario del MW500.

ABREVIACIONES Y DEFINICIONES

ASCII	American Standard Code for Information Interchange
CAN	Controller Area Network
CiA	CAN in Automation
CIP	Common Industrial Protocol
PLC	Programmable Logic Controller
HMI	Human-Machine Interface
ODVA	Open DeviceNet Vendor Association
ro	Read only (solamente de lectura)
rw	Read/write (lectura y escrita)

REPRESENTACIÓN NUMÉRICA

Números decimales son representados a través de dígitos sin sufijo. Números hexadecimales son representados con la letra 'h' luego del número.

DOCUMENTOS

El protocolo DeviceNet fue desarrollado con base en las siguientes especificaciones y documentos:

Documento	Versión	Fuente
CAN Specification	2.0	CiA
Volume One Common Industrial Protocol (CIP) Specification	3.2	ODVA
Volume Three DeviceNet Adaptation of CIP	1.4	ODVA

Para obtener esta documentación, de debe consultar la ODVA, que actualmente es la organización que mantiene, promociona y actualiza las informaciones relativas a la red DeviceNet.

1 INTRODUCCIÓN A LA COMUNICACIÓN DEVICENET

Para la operación en red DeviceNet, es necesario conocer el modo como la comunicación es hecha. Para es, este ítem trae una descripción general del funcionamiento del protocolo DeviceNet, conteniendo las funciones utilizadas por el MW500. Para una descripción detallada, consulte la especificación del protocolo.

1.1 CAN

La red DeviceNet es una red basada en CAN, lo que significa decir que ella utiliza telegramas CAN para intercambios de datos en la red.

El protocolo CAN es un protocolo de comunicación serial que describe los servicios de la camada 2 del modelo ISO/OSI (camada de enlace de datos)¹. En esta camada, son definidos los distintos tipos de telegramas (frames), el modo de detección de errores, validación y arbitración de los mensajes.

1.1.1 Frame de Datos

Los datos en una red CAN son transmitidos a través de un frame de datos. Este tipo de frame es formado principalmente por un campo identificador de 11 bits² (arbitration field), y un campo de datos (data field), que puede contener hasta 8 bytes de datos.

Identificador	8 bytes de datos							
11 bits	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7

1.1.2 Frame Remoto

Además del frame de datos, existe también el frame remoto (RTR frame). Este tipo de frame no posee campo de datos, solo el identificador. Elle funciona como una requisición para que otro dispositivo de la red transmita el frame de datos deseado. El protocolo DeviceNet no utiliza este tipo de frame.

1.1.3 Acceso a la Red

En una red CAN, cualquier elemento de la red puede intentar transmitir un frame para la red en un determinado instante. Caso dos elementos intenten acceder la red al mismo tiempo, conseguirá transmitir aquello que enviar el mensaje más prioritaria. La prioridad del mensaje es definida por el identificador del frame CAN, cuanto menor el valor de este identificador, mayor es la prioridad del mensaje. El telegrama con el identificador "0" (cero) corresponde al telegrama más prioritario.

1.1.4 Control de Errores

La especificación CAN define diversos mecanismos para el control de errores, lo que la torna una red muy confiable y con un índice muy bajo de errores de transmisión que no son detectados. Cada dispositivo de la red debe ser capaz de identificar la ocurrencia de estos errores, e informar los demás elementos que un error fue detectado.

Un dispositivo de la red CAN posee contadores internos que son incrementados toda vez que un error de transmisión o recepción es detectado, y decrementado cuando un telegrama es enviado o recibido con suceso. Caso ocurra una cantidad considerable de errores, el dispositivo puede ser llevado para los siguientes estados:

- **Error Active:** los contadores internos de error están en un nivel bajo y el dispositivo opera normalmente en la red CAN. Puede enviar y recibir telegramas y actuar en la red CAN caso detecte algún error en la transmisión de telegramas.
- **Warning:** cuando ese contador pasa de un determinado límite, el dispositivo entra en el estado de *warning*, significando la ocurrencia de una elevada tasa de errores.
- **Error Passive:** cuando este valor ultrapasa un límite mayor, elle entra en el estado de *error passive*, donde es dispositivo para de actuar en la red al detectar que otro dispositivo ha enviado un telegrama con error.

¹ En la especificación del protocolo CAN, es referenciada a la normativa ISO 11898 como definición de la camada 1 de este modelo (camada física).

² La especificación CAN 2.0 define dos tipos de frames de datos: *standard* (11bits) y *extended* (29 bits). Para esta implementación, solamente frames *standard* son aceptos.

- **Bus Off:** por último, tenemos el estado de *bus off*, en el cual el dispositivo no irá más enviar o recibir telegramas. El dispositivo opera como si estuviera desconectado de la red.

1.1.5 CAN y DeviceNet

Solamente la definición de como detectar errores, crear y transmitir un frame no son suficientes para definir un significado para los datos que son enviados vía red. Es necesario que haya una especificación que indique como el identificador y los datos deben ser montados y como las informaciones deben ser intercambiadas. De este modo los elementos de la red pueden interpretar correctamente los datos que son transmitidos. En este sentido, la especificación DeviceNet define justamente como intercambiar datos entre los equipamientos y como cada dispositivo debe interpretar estos datos.

Existen diversos otros protocolos basados en CAN, como DeviceNet, CANopen, J1939, etc., que utilizan frames CAN para la comunicación. Sin embargo estos protocolos no pueden operar en conjunto en la misma red.

1.2 CARACTERÍSTICAS DE LA RED DEVICENET

Presentado en 1994, DeviceNet es una implementación del protocolo *Common Industrial Protocol* (CIP) para redes de comunicación industrial. Desarrollado originalmente por la Allen-Bradley, donde ha tenido su tecnología transferida para la ODVA que, desde entonces, mantiene, divulga y promueve el DeviceNet y otras redes basadas en el protocolo CIP³. Además de eso utiliza el protocolo *Controller Area Network* (CAN) para enlace de datos y acceso al medio, capas 2 y 1 del modelo OSI/ISO, respectivamente.

Utilizado principalmente en la interconexión de controladores industriales y dispositivos de entrada/salida (I/O), el protocolo sigue el modelo productor-consumidor, soporta múltiples modos de comunicación y posee prioridad entre mensajes.

Es un sistema que puede ser configurado para operar tanto en una arquitectura maestro-esclavo cuanto en una arquitectura distribuida punto a punto. Además de eso, define dos tipos de mensajes, I/O (datos de proceso) y *explicit* (configuración y parametrización). Posee también mecanismos de detección de dirección dobles y aislamiento de los nudos en caso de fallas críticas.

Una red DeviceNet puede contener hasta 64 dispositivos, con dirección de 0 a 63. Cualquier uno de estos puede ser utilizado. No hay cualquier restricción, sin embargo se debe evitar el 63, pues esta dirección es común de ser utilizada para fines de pruebas (comisionamiento / puesta en marcha).

1.3 MEDIO FÍSICO

DeviceNet usa una topología de red del tipo tronco/derivaciones que permite que tanto el cableado de la señal cuanto de alimentación se encuentre presentes en el mismo cable. Esta alimentación, suministrada por una fuente conectada directamente en la red, alimenta los transceivers CAN de los nudos, y posee las siguientes características:

- 24 Vdc;
- Salida DC aislada de la entrada AC;
- Capacidad de corriente compatible con los equipamientos instalados.

El tamaño total de la red varía de acuerdo con la tasa de transmisión utilizada, conforme presentado en la tabla 1.1.

Tabla 1.1: Tamaño de la red x Tasa de transmisión

Tasa de transmisión	Tamaño de la red	Derivaciones	
		Máximo	Total
125 kbps	500 m	6 m	156 m
250 kbps	250 m		78 m
500 kbps	100 m		39 m

³ CIP representa, en la realidad, una familia de redes. DeviceNet, EtherNet/IP y ControlNet utilizan CIP en la camada de aplicación. La diferencia entre ellos está primordialmente en las capas de enlace de datos y física.

Para evitar reflexiones de la señal en la línea, se recomienda la instalación de resistores de terminación en las extremidades de la red, pues la falta de estos puede provocar errores intermitentes. Este resistor debe poseer las siguientes características, conforme especificación del protocolo:

- 121 Ω ;
- 0,25 W;
- 1% de tolerancia.

En DeviceNet, diversos tipos de conectores pueden ser utilizados, tanto sellados cuanto abiertos. La definición del tipo a ser utilizado dependerá de la aplicación y del ambiente de operación del equipamiento. El MW500 utiliza un conector del tipo *plug-in* de 5 vías cuyos terminales está presentado en la sección 2. Para una descripción completa de los conectores utilizados por el DeviceNet consulte la especificación del protocolo.

1.3.1 Camada de Enlace de Datos

La camada de enlace de datos del DeviceNet es definida por la especificación del CAN, el cual define dos estados posibles; dominante (nivel lógico 0) y recesivo (nivel lógico 1). Un nudo puede llevar la red al estado dominante si transmitir alguna información. Así, el bus solamente estará en el estado recesivo si no se tiene nudos transmisores en el estado dominante.

CAN utiliza el CSMA/NBA para acceder el medio físico. Eso significa que un nudo, antes de transmitir, debe comprobar si el bus está libre. Caso este, entonces él podrá iniciar la transmisión de su telegrama. Caso no este, deberá aguardar. Si más de un nudo acceder la red simultáneamente, un mecanismo basado en prioridad de mensaje entrará en acción para decidir cual de ellos tendrá prioridad sobre los otros. Este mecanismo es no destructivo, o sea, el mensaje es preservado mismo que ocurran choques entre dos o más telegramas.

CAN define cuatro tipos de telegramas (*data*, *remote*, *overload*, *error*). De estos telegramas, DeviceNet utiliza solo el frame de datos (*data frame*) y el frame de errores (*error frame*).

Datos son trasladados utilizándose el frame de datos. La estructura de este frame es presentado en la figura 1.1.

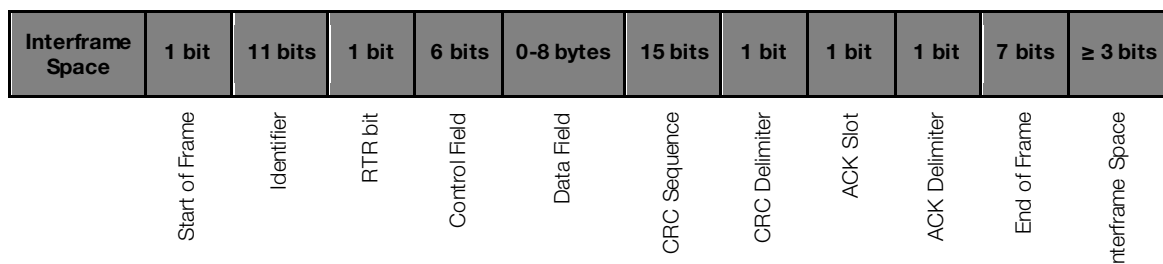


Figura 1.1: Frame de datos CAN

Ya los errores son señalados a través del frame de errores. CAN posee una verificación y un confinamiento de errores bastante robusto. Eso garantiza que un nudo con problemas no perjudique la comunicación en la red.

Para una descripción completa de los errores, consulte la especificación del CAN.

1.3.2 Camada de Transporte y Red

DeviceNet requiere que una conexión sea establecida antes de haber intercambio de datos con el dispositivo. Para establecer esta conexión, cada nudo DeviceNet debe implementar el *Unconnected Message Manager* (UCMM) o el *Group 2 Unconnected Port*. Estos dos mecanismos de posicionamiento utilizan mensajes del tipo explicit para establecer la conexión, que a seguir será utilizada para el intercambio de datos de proceso entre un nudo y otro. Este intercambio de datos utiliza mensajes del tipo I/O (consultar ítem 1.3.5).

Los telegramas DeviceNet son clasificados en grupos, le cual definen funciones y prioridades específicas. Estos telegramas utilizan el campo identificador (11 bits) del frame de datos CAN para identificar únicamente cada uno de los mensajes, garantizando así el mecanismo de prioridades CAN.

Un nudo DeviceNet puede ser cliente, servidor o los dos. Además de eso, clientes y servidores pueden ser productores y/o consumidores de mensajes. En un típico nudo cliente, por ejemplo, su conexión producirá requisiciones y consumirá respuestas. Otras conexiones de clientes o servidores solo consumirán mensajes. O sea, el protocolo prevé diversas posibilidades de conexión entre los dispositivos.

El protocolo dispone también de un recurso para detección de nudos con direcciones (Mac ID) duplicados. Evite que ocurra direcciones duplicadas, en general, es más eficiente que intentar ubicarlas después.

1.3.3 Camada de Aplicación – Protocolo CIP

DeviceNet utiliza el *Common Industrial Protocol (CIP)* en la camada de aplicación. Se trata de un protocolo estrictamente orientado a objetos, también es utilizado por el ControlNet y por el EtherNet/IP. O sea, él es independiente del medio físico y de la camada de enlace de datos. La figura 1.2 presenta la estructura de este protocolo.

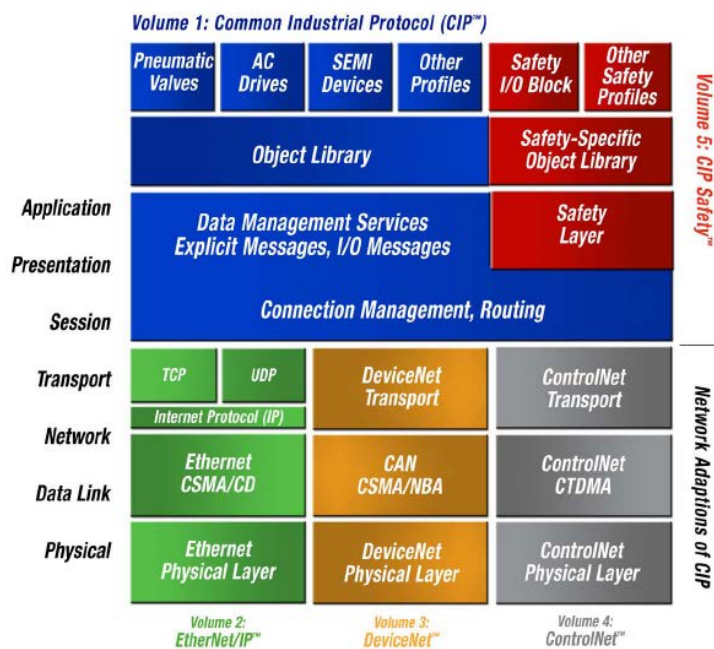


Figura 1.2: Estructura en capas del protocolo CIP

CIP tiene dos objetivos principales:

- Transporte de datos de control de los dispositivos de I/O.
- Transporte de informaciones de configuración y diagnóstico del sistema que esta siendo controlado.

Un nudo (maestro o esclavo) DeviceNet es entonces modelado por un conjunto de objetos CIP, los cuales encapsulan datos y servicios y determinan así su comportamiento.

Existen objetos obligatorios (todo dispositivo debe contener) y objetos opcionales. Objetos opcionales son aquellos que moldean el dispositivo de acuerdo con la categoría (llamado de perfil) que pertenecen, tales como: AC/DC Drive, lector de código de barras o válvula neumática. Por sierran diferentes, cada un de estos contendrá un conjunto también diferente de objetos.

Para más informaciones, consulte la especificación del DeviceNet. Ella presenta un listado completo de los perfiles de dispositivos ya padronizados por la ODVA, bien como los objetos que la componen.

1.3.4 Archivo de Configuración EDS

Cada dispositivo en una red DeviceNet posee un archivo de configuración EDS, que contiene las informaciones sobre el funcionamiento del dispositivo y debe ser registrado en el software de configuración, para programación de los dispositivos presentes en la red DeviceNet.

El archivo de configuración EDS es suministrado en conjunto con el producto, y también puede ser obtenido a través del sitio web <http://www.weg.net>. Es necesario observar la versión de software del equipamiento, para utilizar un archivo EDS que sea compatible con esta versión.

1.3.5 Modos de Comunicación

El protocolo DeviceNet posee dos tipos básicos de mensajes, I/O y explicit. Cada uno de ellos es adecuado a un determinado tipo de dato, conforme presentado abajo:

- I/O: tipo de telegrama síncrono dedicado al traslado de datos prioritarios entre un productor y un o más consumidores. Se dividen de acuerdo con el método de intercambio de datos. Los principales son:
 - *Polled*: método de comunicación en que el maestro envía un telegrama a cada uno de los esclavos de su lista (*scan list*). Así que recibe la solicitud, el esclavo contesta prontamente la solicitud del maestro. Este proceso es repetido hasta que todos sean consultados, reiniciando el ciclo.
 - *Bit-strobe*: método de comunicación donde el maestro envía para la red un telegrama conteniendo 8 bytes de datos. Cada bit de estos 8 bytes representa un esclavo que, cuando es llamado, contesta de acuerdo con el programado.
 - *Change of State*: método de comunicación donde el intercambio de datos entre maestro y esclavo ocurre solo cuando se tiene cambios en los valores monitoreados/controlados, hasta un cierto límite de tiempo. Cuando este límite es alcanzado, la transmisión y recepción ocurrirán mismo que no se tenga ocurrido alteraciones. La configuración de esta variable de tiempo es hecha en el programa de configuración de la red.
 - *Cyclic*: otro método de comunicación muy semejante al anterior. La única diferencia se queda por cuenta de la producción y del consumo de mensajes. En este tipo, todo intercambio de datos ocurre en intervalos regulares de tiempo, independiente de haberes sido modificados o no. Este período también es ajustado en el software de configuración de red.
- Explicit: tipo de telegrama de uso general y no prioritario. Utilizado principalmente en tareas asíncronas tales como parametrización y configuración do equipamiento.



¡NOTA!

El convertidor de frecuencia MW500 no disponiza el método de comunicación *Bit-strobe*.

1.3.6 Conjunto de Conexiones Predefinidas Maestro/Esclavo

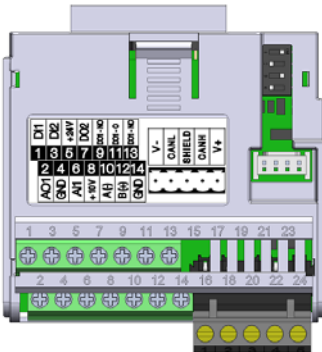
DeviceNet utiliza fundamentalmente un modelo de mensajes punto a punto. Sin embargo, es bastante común utilizar un esquema predefinido de comunicación basado en el mecanismo maestro/esclavo.

Esta estructura utiliza un movimiento simplificado de mensajes del tipo I/O muy común en aplicaciones de control. Las ventajas de este método están en los requisitos necesarios para ejecutarlo, en general menores si comparados al UCMM. Hasta mismo dispositivos simples con recursos limitados (memoria, procesador de 8 bits) son capaces de ejecutar el protocolo.

2 ACCESORIO PARA COMUNICACIÓN DEVICENET

Para posibilitar la comunicación DeviceNet en el producto, es necesario utilizar el kit para comunicación CAN descrito a seguir. Informaciones a respecto de la instalación de estos módulos pueden ser obtenidas en el guía rápido que acompaña el kit.

2.1 KIT CAN-01



- Ítem WEG: 11593087.
- Formado por el módulo de comunicación CAN (figura al lado) más un guía rápido de montaje.
- Interfaz aislada galvánicamente y con señal diferencial, atribuyendo mayor robustez contra la interferencia electromagnética.
- Alimentación externa de 24V.
- Permite la conexión de hasta 64 dispositivos en el mismo segmento. Una cantidad mayor de dispositivos puede ser conectado con el uso de repetidores⁴.
- Longitud máxima del bus de 1000 metros.

2.2 TERMINALES DEL CONECTOR

El módulo para la comunicación CAN posee un conector *plug-in* de 5 vías con los siguientes terminales:

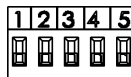


Tabla 2.1: Terminales del conector para interfaz CAN

Terminal	Nombre	Función
1	V-	Polo negativo de la fuente de alimentación
2	CAN_L	Señal de comunicación CAN_L
3	Shield	Blindaje del cable
4	CAN_H	Señal de comunicación CAN_H
5	V+	Polo positivo de la fuente de alimentación



¡NOTA!

Se recomienda conectar a tierra el pin GND del módulo CFW500-CCAN. Esto es necesario para que la pantalla del cable tenga una conexión con la tierra.

2.3 FUENTE DE ALIMENTACIÓN

El interfaz CAN necesita de una tensión de alimentación externa entre los terminales 1 y 5 del conector de la red. Los datos para consumo individual y tensión de entrada son presentados en la tabla que sigue.

Tabla 2.2: Características de la alimentación para interfase CAN

Tensión de alimentación (V_{cc})		
Mínimo	Máximo	Sugerido
11	30	24
Corriente (mA)		
Típico		Máximo
30		50

2.4 SEÑALIZACIONES

Indicaciones de alarmas, fallas y estados de la comunicación son realizadas a través de la HMI y de los parámetros del producto.

⁴ El número limite de equipos que pueden ser conectados en la red también depende del protocolo utilizado.

2.5 CONFIGURACIÓN DEL MÓDULO

Para configurar el módulo DeviceNet proceder los pasos indicados abajo:

- Con el convertidor de frecuencia sin alimentación de corriente instale el módulo plug-in CFW500-CCAN.
- Certifíquese de que él está correctamente encajado y fijado por los tornillos.
- Energice el convertidor de frecuencia.
- Verifique el contenido del parámetro P0027 y verifique si el accesorio fue correctamente reconocido. Consulte el guía rápido de instalación si necesario.
- Ajuste el protocolo CAN para DeviceNet a través del parámetro P0700.
- Ajuste la dirección del convertidor de frecuencia en la red a través del parámetro P0701.
 - Valores válidos: 0 a 63.
- Ajuste la tasa de comunicación en el P0702. Valores válidos:
 - 0 = Auto
 - 1 = Auto
 - 2 = 500kbps
 - 3 = 250kbps
 - 4 = 125kbps
 - 5 = Auto
 - 6 = Auto
 - 7 = Auto
 - 8 = Auto
- En el parámetro P0710 configure la instancia de I/O más adecuada a la aplicación (esta elección tendrá impacto en la cantidad de palabras intercambiadas con el maestro de la red). Exactamente esta misma cantidad de palabras deberá ser ajustada en el maestro de la red. Por fin, programe un valor distinto de 0 para los parámetros P0711 a P0718.
 - Valores válidos: 0 a 1199.
- Interrumpa la alimentación de corriente y alimente nuevamente el MW500 para que los cambios tengan efecto.
- Conecte el cable de red en el módulo.
- Registre el archivo de configuración (archivo EDS) en el software de configuración de la red.
- Inserte el MW500 en la *scan list* del maestro.
- En el *software* de configuración de la red elija un método para intercambio de datos con el maestro, o sea, *polled*, *change of state* o *cyclic*. El módulo DeviceNet del MW500 soporta todos estos tipos de datos de I/O, además del *explicit* (datos acíclicos).
- Si todo se encuentra correctamente configurado, el parámetro P0719 señalará el estado 'Online, Conectado'. Observe también el parámetro que señala el estado del maestro de la red, P0720. Solamente habrá intercambio efectivo de datos cuando el estado del maestro es *Run*.

2.6 ACCESO A LOS PARÁMETROS

Luego de registro del archivo EDS en el software de configuración de red, el usuario tendrá acceso al listado completa de los parámetros del equipamiento, los cuales pueden ser accedidos vía *explicit messages*. Eso significa que es posible hacer la parametrización y la configuración del drive a través del software de configuración de red.

Para detalles de utilización de esto recurso, consulte la documentación del software de programación del maestro de la red (PLC, PC, etc.).

3 INSTALACIÓN DE LA RED DEVICENET

La red DeviceNet, como varias redes de comunicación industriales, por el hecho de ser aplicada muchas veces en ambientes agresivos y con alta exposición a la interferencia electromagnética, exige ciertos cuidados que deben ser aplicados para garantizar una baja tasa de errores de comunicación durante su operación. A seguir son presentadas recomendaciones para realizar la conexión del producto en esta red.

3.1 TASA DE COMUNICACIÓN

Equipamientos con interfaz DeviceNet en general permiten configurar la tasa de comunicación deseada, pudiendo variar de 125Kbit/s hasta 500Kbit/s. La tasa de comunicación (*baud rate*) que puede ser utilizada por un equipamiento depende de la longitud del cable utilizado en la instalación. La tabla que sigue presenta las tasas de comunicación y la longitud máxima de cable que puede ser utilizado en la instalación, de acuerdo con el recomendado por la ODVA⁵.

Tabla 3.1: Tasas de comunicación soportadas y longitud máxima de cable

Tasa de comunicación	Longitud del cable
500 Kbit/s	100 m
250 Kbit/s	250 m
125 Kbit/s	500 m

Todos los equipamientos de la red deben programarse para utilizar la misma tasa de comunicación. Para el convertidor de frecuencia MW500, la configuración de la tasa de comunicación es hecha a través del parámetro P0702.

3.2 DIRECCIÓN EN LA RED DEVICENET

Cada dispositivo de la red DeviceNet precisa tener una dirección, o MAC ID, entre 0 y 63. Esta dirección debe ser única para cada equipamiento. Para el convertidor de frecuencia MW500, la configuración de la dirección es hecha a través del parámetro P0701.

3.3 RESISTOR DE TERMINACIÓN

La utilización de resistencias de terminación en las extremidades del bus CAN es fundamental para evitar reflexión de línea, que puede perjudicar la señal transmitida y ocasionar errores en la comunicación. Las extremidades del bus CAN deben poseer un resistor de terminación en el valor de $121\Omega / 0.25W$, conectando las señales CAN_H y CAN_L.

3.4 CABLE

Para la conexión de las señales CAN_L y CAN_H se debe utilizar par tranzado con blindaje. La tabla a seguir presenta las características recomendadas para el cable.

Tabla 3.2: Propiedades del cable para red DeviceNet

Longitud del cable (m)	Resistencia por metro (mOhm/m)	Área del conductor (mm ²)
0 ... 40	70	0.25 ... 0.34
40 ... 300	<60	0.34 ... 0.60
300 ... 600	<40	0.50 ... 0.60
600 ... 1000	<26	0.75 ... 0.80

También es necesaria la utilización de un par tranzado adicional para llevar la alimentación de 24Vcc para los equipamientos que necesitan de esta señal. Se recomienda usar un cable certificado para red DeviceNet.

⁵ Diferentes productos pueden tener diferentes longitudes máximas de cable permitida para la instalación.

3.5 CONEXIÓN CON LA RED

Para interconectar los diversos nodos de la red, se recomienda la conexión del equipamiento directamente a partir de la línea principal, sin la utilización de derivaciones. Durante la instalación de los cables, se debe evitar su disposición cerca de los cables de potencia, pues debido a la interferencia electromagnética, eso facilita la ocurrencia de errores durante la transmisión. Para evitar problemas de circulación de corriente por diferencia de potencial entre distintos puntos de puesta a tierra, es necesario que todos los dispositivos estén conectados en el mismo punto de tierra.

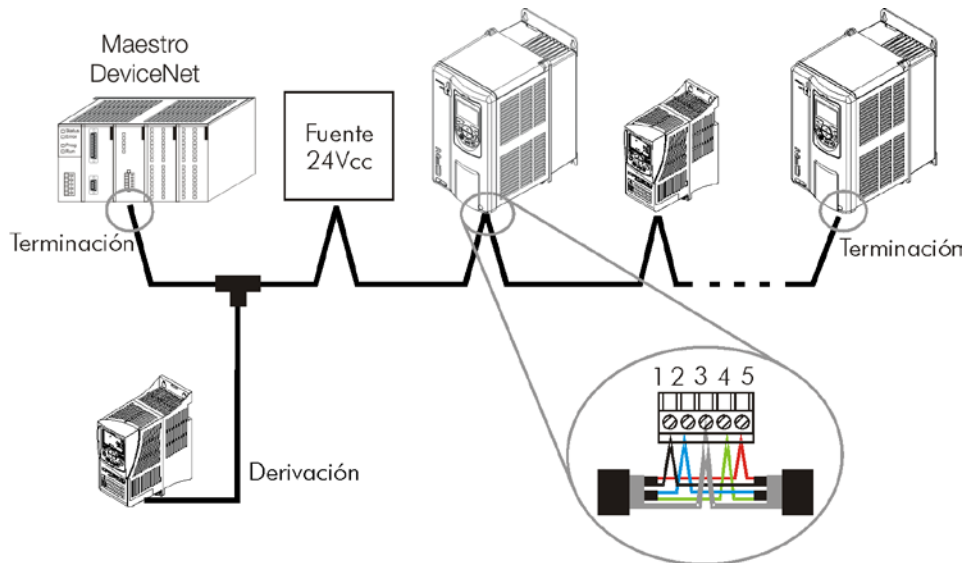


Figura 3.1: Ejemplo de instalación en red DeviceNet

Para evitar problemas de diferencia de tensión en la alimentación entre los dispositivos de la red, es recomendado que la red sea alimentada en apenas un punto, y la señal de alimentación sea llevada a todos los dispositivos a través del cable. Caso sea necesaria más de una fuente de alimentación, éstas deben estar referenciadas al mismo punto.

El número máximo de dispositivos conectados en un único segmento de la red es limitado en 64. Repetidores pueden ser utilizados para conectar un número mayor de dispositivos.

4 PARAMETRIZACIÓN

A seguir serán presentados solamente los parámetros del convertidor de frecuencia MW500 que poseen relación directa con la comunicación DeviceNet.

4.1 SÍMBOLOS PARA DESCRIPCIÓN DE LAS PROPIEDADES

RO	Parámetro solamente de lectura
CFG	Parámetro solamente modificado con el motor parado
CAN	Parámetro visible a través de la IHM si el producto poseer interfaz CAN instalada

P0105 – SELECCIÓN 1ª/2ª RAMPA

P0220 – SELECCIÓN FUENTE LOCAL/REMOTO

P0221 – SELECCIÓN REFERENCIA LOCAL

P0222 – SELECCIÓN REFERENCIA REMOTA

P0223 – SELECCIÓN GIRO LOCAL

P0224 – SELECCIÓN GIRA/PARA LOCAL

P0225 – SELECCIÓN JOG LOCAL

P0226 – SELECCIÓN GIRO REMOTO

P0227 – SELECCIÓN GIRA/PARA REMOTO

P0228 – SELECCIÓN JOG REMOTO

Estos parámetros son utilizados en la configuración de la fuente de los comandos para los modos de operación local y remota del convertidor de frecuencia MW500. Para que el equipo sea controlado a través de la interfaz DeviceNet, se debe seleccionar una de las opciones 'CANopen/DeviceNet/Profibus DP' disponibles en los parámetros.

La descripción detallada de estos parámetros se encuentra en el manual de programación del convertidor de frecuencia MW500.

P0313 – ACCIÓN PARA ERROR DE COMUNICACIÓN

Rango de	0 = Inactivo	Padrón: 1
Valores:	1 = Para por Rampa	
	2 = Deshabilita General	
	3 = Va para modo Local	
	4 = Va para modo Local y mantiene comandos y referencia	
	5 = Causa Falla	
Propiedades:	CFG	
Grupo de acceso vía HMI:	NET	

Descripción:

Este parámetro permite seleccionar cual es la acción que debe ser ejecutada por el equipo, caso este sea controlado vía red y un error de comunicación sea detectado.

Tabla 4.1: Valores de lo parámetro P0313

Opciones	Descripción
0 = Inactivo	Ninguna acción es tomada, el equipo permanece en el estado actual.
1 = Para por Rampa	El comando de parada por rampa es ejecutado, y el motor para de acuerdo con la rampa de desaceleración programada.
2 = Deshabilita General	El equipo es deshabilitado general, y el motor para por inercia.
3 = Va para modo Local	El equipo es comandado para el modo local.
4 = Va para modo Local y mantiene comandos y referencia	El equipo es comandado para el modo local, más los comandos de habilita y de referencia de velocidad recibidos vía red son mantenidos en modo local, desde que el equipo sea programado para utilizar, en modo local, comandos vía HMI o 3 "wire start stop", y la referencia de velocidad vía HMI o potenciómetro electrónico.
5 = Causa Falla	En el lugar de alarma, un error de comunicación causa una falla en el convertidor de frecuencia; siendo necesario hacer el reset de fallas en el convertidor de frecuencia para que el mismo regrese a su operación normal.

Se considera errores de comunicación los siguientes eventos:

Comunicación CANopen/DeviceNet

- Alarma A133/Falla F233: sin alimentación en la interfaz CAN.
- Alarma A134/Falla F234: *bus off*.
- Alarma A135/Falla F235: error de comunicación CANopen (*Node Guarding/Heartbeat*).
- Alarma A136/Falla F236: maestro de la red DeviceNet en modo *Idle*.
- Alarma A137/Falla F237: ocurrió *timeout* en una o más conexiones I/O DeviceNet.

Las acciones descritas en este parámetro son ejecutadas a través de la escrita automática de los respectivos bits en el parámetro de control de la interfaz de red que corresponde a la falla detectada. De esta forma, para que los comandos escritos en este parámetro tengan efecto, es necesario que el equipo se encuentre programado para ser controlado vía la interfaz de red utilizada (a excepción de la opción "Causa Falla", que bloquea el equipo aunque el mismo no sea controlado vía red). Esta programación es hecha a través de los parámetros P0220 hasta P0228.

P0680 – ESTADO LÓGICO

Rango de 0000h a FFFFh

Padrón: -

Valores:

Propiedades: RO

Grupo de acceso vía HMI: NET

Descripción:

Permite el monitoreo del estado del convertidor de frecuencia. Cada bit representa un estado:

Bits	15	14	13	12	11	10	9	8	7	6	5	4	3 a 0
Función	En Falla	Manual/ Automático	Subtensión	LOC/REM	JOG	Sentido de Giro	Habilitado General	Motor Girando	En Alarma	En modo de configuración	Segunda Rampa	Parada Rápida Activa	Reservado

Tabla 4.2: Función de los bits para el parámetro P0680

Bits	Valores
Bits 0 a 3	Reservado.
Bit 4 Parada Rápida Activa	0: Drive no posee comando de parada rápida activa. 1: Drive está ejecutando el comando de parada rápida.
Bit 5 Segunda Rampa	0: Drive configurado para rampa de aceleración y de desaceleración del motor vía primera rampa; valores programados en los parámetros P0100 y P0101. 1: Drive configurado para rampa de aceleración y de desaceleración del motor vía segunda rampa; valores programados en los parámetros P0102 y P0103.
Bit 6 En Modo de Configuración	0: Drive operando normalmente. 1: Drive en modo de configuración. Indica una condición especial en la cual el drive no puede ser habilitado: Ejecutando la rutina de autoajuste. Ejecutando la rutina de puesta en marcha (start-up) orientada. Ejecutando la función copy de la HMI. Ejecutando la rutina auto-guiada de la tarjeta de memoria flash. Posee incompatibilidad de parametrización. Sin alimentación en el circuito de potencia del drive.
Bit 7 En Alarma	0: Drive no está en el estado de alarma. 1: Drive está en el estado de alarma. Observación: el número de la alarma puede ser leído a través del parámetro P0048 – Alarma Actual.
Bit 8 Motor Girando	0: Motor está parado. 1: Drive está girando el eje del motor a la velocidad de referencia, o ejecutando rampa de aceleración o desaceleración.
Bit 9 Habilitado General	0: Drive está deshabilitado general. 1: Drive está habilitado general y listo para girar el eje del motor.
Bit 10 Sentido de Giro	0: Motor girando en el sentido reverso. 1: Motor girando en el sentido directo.
Bit 11 JOG	0: Función JOG inactiva. 1: Función JOG activa.
Bit 12 LOC/REM	0: Drive en modo local. 1: Drive en modo remoto.
Bit 13 Subtensión	0: Sin subtensión. 1: Con subtensión.
Bit 14 Manual/ Automático	0: En modo manual (función PID). 1: En modo automático (función PID).
Bit 15 En Falla	0: Drive no está en el estado de falla. 1: Algún falla registrado por el drive. Observación: El número del falla puede ser leído a través del parámetro P0049 – Falla Actual.

P0681 – VELOCIDAD DEL MOTOR EN 13 BITS
Rango de - 32768 a 32767

Padrón: -

Valores:
Propiedades: RO

Grupo de acceso vía HMI: NET

Descripción:

Permite monitorear la velocidad del motor. Esta palabra utiliza resolución de 13 bits con señal para representar la frecuencia nominal (P0403) del motor:

- P0681 = 0000h (0 decimal) → velocidad del motor = 0
- P0681 = 2000h (8192 decimal) → velocidad del motor = frecuencia nominal

Valores de velocidad intermediarios o superiores pueden ser obtenidos utilizando esta escala. Por ejemplo, 60 Hz de frecuencia nominal, caso el valor leído sea 2048 (0800h), para obtener el valor en Hz se debe calcular:

8192 => 60 Hz 2048 => Frecuencia en Hz

Frecuencia en Hz = $\frac{60 \times 2048}{8192}$
--

Frecuencia en Hz = 30 Hz

Valores negativos para este parámetro indican motor girando en el sentido reverso.


¡NOTA!

Los valores transmitidos a través de la red presentan una limitación en la escala utilizada, que permite indicar una velocidad máxima de 4 veces la velocidad de sincronismo del motor, con saturación en 32767 (o -32768).

P0684 – PALABRA DE CONTROL VÍA DEVICENET

Rango de	0000h a FFFFh	Padrón: 0000h
Valores:		
Propiedades:	-	
Grupo de acceso vía HMI:	NET	

Descripción:

Palabra de comando del **convertidor de frecuencia** vía interfaz DeviceNet. Este parámetro solamente puede ser modificado vía interfaz CANopen/DeviceNet/Profibus DP. Para las demás fuentes (HMI, etc.) ele se comporta como un parámetro solamente de lectura.

Para que los comandos escritos en este parámetro sean ejecutados, es necesario que el equipo se encuentre programado para ser controlado vía CANopen/DeviceNet/Profibus DP. Esta programación es hecha a través de los parámetros P0105 y P0220 hasta P0228.

Cada bit de esta palabra representa un comando que puede ser ejecutado en el equipo.

Bits	15 a 8	7	6	5	4	3	2	1	0
Función	Reservado	Reset de Fallas	Parada Rápida	Utiliza Segunda Rampa	LOC/REM	JOG	Sentido de Giro	Habilita General	Gira/Para

Tabla 4.3: Función de los bits para el parámetro P0684

Bits	Valores
Bit 0 Gira/Para	0: Para el eje del motor por rampa de desaceleración. 1: Gira el eje del motor de acuerdo con la rampa de aceleración hasta alcanzar el valor de la referencia de velocidad.
Bit 1 Habilita General	0: Deshabilita general el drive de frecuencia, interrumpiendo la alimentación para el motor. 1: Habilita general el drive, permitiendo la operación del motor.
Bit 2 Sentido de Giro	0: Girar el eje del motor en el sentido opuesto al de la referencia. 1: Girar el eje del motor en el sentido indicado en la referencia.
Bit 3 JOG	0: Deshabilita la función JOG. 1: Habilita la función JOG.
Bit 4 LOC/REM	0: Drive va para el modo local. 1: Drive va para el modo remoto.
Bit 5 Utiliza Segunda Rampa	0: Drive configurado para rampa de aceleración y de desaceleración del motor vía primera rampa; valores programados en los parámetros P0100 y P0101. 1: Drive configurado para rampa de aceleración y de desaceleración del motor vía segunda rampa; valores programados en los parámetros P0102 y P0103.
Bit 6 Parada Rápida	0: No ejecuta el comando de parada rápida. 1: Ejecuta el comando de parada rápida. Observación: cuando el tipo de control (P0202) es V/f o VVW no se recomienda la utilización de esta función.
Bit 7 Reset de Fallas	0: Sin función. 1: Si en estado de falla, ejecuta el reset del drive.
Bits 8 a 15	Reservado.

P0685 – REFERENCIA DE VELOCIDAD VÍA DEVICENET

Rango de	-32768 a 32767	Padrón: 0
Valores:		
Propiedades:	-	
Grupo de acceso vía HMI:	NET	

Descripción:

Permite programar la referencia de velocidad para el motor vía interfaz DeviceNet. Este parámetro solamente puede ser modificado vía interfaz CANopen/DeviceNet/Profibus DP. Para las demás fuentes (HMI, etc.) elle se comporta como un parámetro solamente de lectura.

Para que la referencia escrita en este parámetro sea utilizada, es necesario que el equipo se encuentre programado para utilizar la referencia de velocidad vía CANopen/DeviceNet/Profibus DP. Esta programación es hecha a través de los parámetros P0221 y P0222.

Esta palabra utiliza resolución de 13 bits con señal para representar la frecuencia nominal (P0403) del motor:

- P0685 = 0000h (0 decimal) → velocidad del motor = 0
- P0685 = 2000h (8192 decimal) → velocidad del motor = frecuencia nominal (P0403)

Valores de referencias intermedias o superiores pueden ser programados utilizando esta escala. Por ejemplo, 60 Hz de frecuencia nominal, caso se dese una referencia de 30 Hz, se debe calcular:

60 Hz => 8192 30 Hz => Referencia en 13 bits

Referencia en 13 bits = $\frac{60 \times 8192}{30}$

Referencia en 13 bits = 4096	=> Valor correspondiente a 30 Hz en la escala de 13 bits
------------------------------	--

Este parámetro también acepta valores negativos para cambiar el sentido de la rotación del motor. El sentido de la rotación de la referencia, sin embargo, depende también del valor del bit 2 de la palabra de control – P0684:

- Bit 2 = 1 y P0685 > 0: referencia para el sentido directo
- Bit 2 = 1 y P0685 < 0: referencia para el sentido reverso
- Bit 2 = 0 y P0685 > 0: referencia para el sentido reverso
- Bit 2 = 0 y P0685 < 0: referencia para el sentido directo


¡NOTA!

Los valores transmitidos por la red tienen una limitación debido a la escala utilizada, permitiendo que sea programada una velocidad máxima de referencia de 4 veces la velocidad síncrona del motor.

P0695 – VALOR PARA LAS SALIDAS DIGITALES

Rango de	0000h a 001Fh	Padrón: 0000h
Valores:		
Propiedades:	-	
Grupo de acceso vía HMI:	NET	

Descripción:

Posibilita el control de las salidas digitales a través de la interfaz de red (Serial, CAN, etc.). Este parámetro no puede ser modificado a través de la HMI.

Cada bit de este parámetro corresponde al valor deseado para una salida digital. Para que la salida digital correspondiente pueda ser controlada de acuerdo con este contenido, es necesaria que su función sea programada para “Contenido P0695”, en los parámetros P0275 a P0279.

Bits	15 a 5	4	3	2	1	0
Función	Reservado	Valor para DO5	Valor para DO4	Valor para DO3	Valor para DO2	Valor para DO1

Tabla 4.4: Función de los bits para el parámetro P0695

Bits	Valores
Bit 0 Valor para DO1	0: salida DO1 abierta. 1: salida DO1 cerrada.
Bit 1 Valor para DO2	0: salida DO2 abierta. 1: salida DO2 cerrada.
Bit 2 Valor para DO3	0: salida DO3 abierta. 1: salida DO3 cerrada.
Bit 3 Valor para DO4	0: salida DO4 abierta. 1: salida DO4 cerrada.
Bit 4 Valor para DO5	0: salida DO5 abierta. 1: salida DO5 cerrada.
Bits 5 a 15	Reservado.



¡NOTA!

Algunas de las salidas digitales pueden no estar disponibles dependiendo del módulo plug-in utilizado.

P0696 – VALOR 1 PARA SALIDAS ANALÓGICAS

P0697 – VALOR 2 PARA SALIDAS ANALÓGICAS

P0698 – VALOR 3 PARA SALIDAS ANALÓGICAS

Rango de Valores:	-32768 a 32767	Padrón: 0
Propiedades:	-	
Grupo de acceso vía HMI:	NET	

Descripción:

Posibilita el control de las salidas analógicas a través del interfaz de red (Serial, CAN, etc.). Estos parámetros no pueden ser modificados a través de la HMI.

El valor escrito en estos parámetros es utilizado como valor para la salida analógica, desde que la función de la salida analógica deseada sea programada para “Contenido P0696 / P0697”, en los parámetros P0251, P0254, P0257.

El valor debe ser escrito en una escala de 15 bits (7FFFh = 32767)⁶ para representar 100 % del valor deseado para la salida, o sea:

- P0696 = 0000h (0 decimal) → valor para la salida analógica = 0 %
- P0696 = 7FFFh (32767 decimal) → valor para la salida analógica = 100 %

En este ejemplo fue presentado el parámetro P0696, más la misma escala es utilizada para los parámetros P0697 / P0698. Por ejemplo, se desea controlar el valor de la salida analógica 1 a través del serial. En este caso se debe proceder la siguiente programación:

- Elegir uno de los parámetros P0696, P0697, P0698 para ser el valor utilizado por la salida analógica 1. En este ejemplo, vamos elegir el P0696.
- Programar, en la función de la salida analógica 1 (P0254), la opción “Contenido P0696”.
- A través del interfaz de red, escribir en el P0696 el valor deseado para la salida analógica 1, entre 0 y 100 %, de acuerdo con la escala del parámetro.

⁶ Para la resolución real de la salida, consulte el manual del producto.


¡NOTA!

Para el convertidor de frecuencia MW500, la salida analógica 3 representa la salida en frecuencia (FO).

P0700 – PROTOCOLO CAN

Rango de	1 = CANopen	Padrón: 2
Valores:	2 = DeviceNet	
Propiedades:	CFG	
Grupo de acceso vía HMI:	NET	

Descripción:

Permite seleccionar el protocolo deseado para el interfaz CAN. Caso este parámetro fuera alterado, la alteración tendrá efecto solamente si la interfaz CAN estuviera sin alimentación, en *autobaud* o luego que el equipamiento fuera desconectado y conectado nuevamente.

P0701 – DIRECCIÓN CAN

Rango de	0 a 127	Padrón: 63
Valores:		
Propiedades:	CFG	
Grupo de acceso vía HMI:	NET	

Descripción:

Permite programar la dirección utilizada para comunicación CAN del dispositivo. Es necesario que cada equipamiento de la red posea una dirección distinta de las demás. Las direcciones válidas para este parámetro dependen del protocolo programado en el P0700:

- P0700 = 1 (CANopen) → direcciones válidas: 1 a 127.
- P0700 = 2 (DeviceNet) → direcciones válidas: 0 a 63.

Caso este parámetro fuera alterado, la alteración tendrá efecto solamente si la interfaz CAN estuviera sin alimentación, en *autobaud* o luego que el equipamiento fuera desconectado y conectado nuevamente.

P0702 – TASA DE COMUNICACIÓN CAN

Rango de	0 = 1 Mbit/s / <i>Autobaud</i>	Padrón: 0
Valores:	1 = 800 Kbit/s / <i>Autobaud</i>	
	2 = 500 Kbit/s	
	3 = 250 Kbit/s	
	4 = 125 Kbit/s	
	5 = 100 Kbit/s / <i>Autobaud</i>	
	6 = 50 Kbit/s / <i>Autobaud</i>	
	7 = 20 Kbit/s / <i>Autobaud</i>	
	8 = 10 Kbit/s / <i>Autobaud</i>	
Propiedades:	CFG	
Grupo de acceso vía HMI:	NET	

Descripción:

Permite programar el valor deseado para la tasa de comunicación del interfaz CAN, en bits por segundo. Esta tasa debe ser la misma para todos los equipamientos conectados en la red. Las tasas de comunicación soportadas para el dispositivo dependen del protocolo programado en el P0700:

- P0700 = 1 (CANopen): se puede utilizar cualquier tasa indicada en este parámetro, mas no posee la función de detección automática de tasa – *autobaud*.
- P0700 = 2 (DeviceNet): solamente las tasas de 500, 250 y 125 Kbit/s son soportadas. Demás opciones habilitan la función de detección automática de tasa – *autobaud*.

Caso este parámetro fuera alterado, la alteración tendrá efecto solamente si la interfaz CAN estuviera sin alimentación o luego que el equipamiento fuera desconectado y conectado nuevamente.

Luego de una detección con suceso, el parámetro de la tasa de comunicación (P0702) modificase automáticamente para la tasa seleccionada. Para ejecutar nuevamente la función de *autobaud*, es necesario modificar el parámetro P0702 para una de las opciones '*Autobaud*'.

P0703 – RESET DE BUS OFF

Rango de	0 = Manual	Padrón: 0
Valores:	1 = Automático	
Propiedades:	CFG	
Grupo de acceso vía HMI:	NET	

Descripción:

Permite programar cual es el comportamiento del convertidor al detectar un error de *bus off* en el interfaz CAN.

Tabla 4.5: Opciones para el parámetro P0703

Opción	Descripción
0 = Reset Manual	Caso ocurra <i>bus off</i> , será señalado en el HMI la alarma A134/F234, la acción programada en el parámetro P0313 será ejecutada y la comunicación será deshabilitada. Para que el convertidor vuelva a se comunicar a través del interfaz CAN, será necesario desenergizar y energizar nuevamente el convertidor.
1 = Reset Automático	Caso ocurra <i>bus off</i> , la comunicación será reiniciada automáticamente y el error será ignorado. En este caso, no será hecha la señalización de alarma en el HMI y el convertidor no ejecutará la acción descrita en el P0313.

P0705 – ESTADO DEL CONTROLADOR CAN

Rango de	0 = Deshabilitado	Padrón: -
Valores:	1 = <i>Autobaud</i> 2 = Interfaz CAN activa 3 = <i>Warning</i> 4 = <i>Error Passive</i> 5 = <i>Bus Off</i> 6 = Sin alimentación	
Propiedades:	RO	
Grupo de acceso vía HMI:	NET	

Descripción:

Permite identificar si la tarjeta de interfaz CAN está debidamente instalada, y si la comunicación presenta errores.

Tabla 4.6: Valores para el parámetro P0705

Valor	Descripción
0 = Deshabilitado	Interfaz CAN inactiva. Ocurre cuando el convertidor no posee tarjeta de interfaz CAN instalada.
1 = <i>Autobaud</i>	Ejecutando función para detección automática de la tasa de comunicación (apenas para el protocolo DeviceNet).
2 = Interfaz CAN activa	Interfaz CAN activa y sin errores.
3 = <i>Warning</i>	Controlador CAN alcanzo el estado de <i>warning</i> .
4 = <i>Error Passive</i>	Controlador CAN alcanzo el estado de <i>error passive</i> .
5 = <i>Bus Off</i>	Controlador CAN alcanzo el estado de <i>bus off</i> .
6 = Sin alimentación	Interfaz CAN no posee alimentación entre los terminales 1 y 5 del conector.

P0706 – CONTADOR DE TELEGRAMAS CAN RECIBIDOS

Rango de Valores:	0 a 65535	Padrón: -
Propiedades:	RO	
Grupo de acceso vía HMI:	NET	

Descripción:

Este parámetro funciona como un contador cíclico, que es incrementado toda vez que un telegrama CAN es recibido. Suministra un retorno para el operador si el dispositivo está consiguiendo comunicarse con la red. Este contador es puesto a cero siempre que el equipo es desenergizado, hecho el reset o al alcanzar el límite máximo del parámetro.

P0707 – CONTADOR DE TELEGRAMAS CAN TRANSMITIDOS

Rango de Valores:	0 a 65535	Padrón: -
Propiedades:	RO	
Grupo de acceso vía HMI:	NET	

Descripción:

Este parámetro funciona como un contador cíclico, que es incrementado toda vez que un telegrama CAN es transmitido. Suministra un retorno para el operador si el dispositivo está consiguiendo comunicarse con la red. Este contador es puesto a cero siempre que el equipo fuera desenergizado, hecho el reset o al alcanzar el límite máximo del parámetro.

P0708 – CONTADOR DE ERRORES DE BUS OFF

Rango de Valores:	0 a 65535	Padrón: -
Propiedades:	RO	
Grupo de acceso vía HMI:	NET	

Descripción:

Contador cíclico que indica el número de veces que el equipo ha entrado en el estado de *bus off* en la red CAN. Este contador es puesto a cero siempre que el equipo es desenergizado, hecho el reset o al alcanzar el límite máximo del parámetro.

P0709 – CONTADOR DE MENSAJES CAN PERDIDAS

Rango de Valores:	0 a 65535	Padrón: -
Propiedades:	RO	
Grupo de acceso vía HMI:	NET	

Descripción:

Contador cíclico que indica el número de mensajes recibidas por la interfaz CAN, más que no podrán ser procesadas por el equipo. Caso el número de mensajes perdidos sea incrementado con frecuencia, recomendase disminuir la tasa de comunicación utilizada para la red CAN. Este contador es puesto a cero siempre que el equipo es desenergizado, hecho el reset o al alcanzar el límite máximo del parámetro.

P0710 – INSTANCIAS DE I/O DEVICENET

Rango de Valores:	0 = ODVA Basic Speed (2 palabras) 1 = ODVA Extended Speed (2 palabras) 2 = Especific.Fab 2W (2 palabras) 3 = Especific.Fab 3W (3 palabras) 4 = Especific.Fab 4W (4 palabras) 5 = Especific.Fab 5W (5 palabras) 6 = Especific.Fab 6W (6 palabras)	Padrón: 0
Propiedades:	CFG, CAN	
Grupo de acceso vía HMI:	NET	

Descripción:

Permite seleccionar la instancia de la clase *Assembly* para comunicación do tipo I/O.

El convertidor de frecuencia MW500 posee siete opciones de ajustes. Dos de ellas siguen el padrón definido en el perfil *AC/DC Drive Profile* de la ODVA. As otras cinco, representan palabras específicas del convertidor de frecuencia MW500. Las tablas presentadas en seguida describen cada una de estas palabras de control y de monitoreo.


¡NOTA!

Caso este parámetro sea modificado, elle solamente será válido luego que el producto es desenergizado y energizado nuevamente.

0 = Formato de los datos para las instancias *ODVA Basic Speed* (2 palabras):

Llamada de *Basic Speed*, estas instancias representan lo más sencillo interfaz de operación de un equipamiento segundo el perfil *AC/DC Device Profile*. El mapa de los datos es presentado abajo.

Monitoreo (Entrada)

Instancia	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
70	0						Running1		Faulted
	1	-							
	2	Speed Actual (low byte)							
	3	Speed Actual (high byte)							

Control (Salida)

Instancia	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
20	0						Fault Reset		Run Fwd
	1	-							
	2	Speed Reference (low byte)							
	3	Speed Reference (high byte)							

1 = Formato de los datos para las instancias *ODVA Extended Speed* (2 palabras):

Llamada de *Extended Speed*, estas instancias representan un interfaz un poco más mejorada de operación del equipamiento que sigue el perfil *AC/DC Device Profile*. El mapa de los datos es presentado abajo.

Monitoreo (Entrada)

Instancia	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
71	0	At Reference	Ref. from Net	Ctrl from Net	Ready	Running2 (Rev)	Running1 (Fwd)	Warning	Faulted
	1	Drive State							
	2	Speed Actual (low byte)							
	3	Speed Actual (high byte)							

Control (Salida)

Instancia	Byte	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Bit 0
21	0		NetRef	NetCtrl			Fault Reset	Run Rev	Run Fwd
	1	-							
	2	Speed Reference (low byte)							
	3	Speed Reference (high byte)							

La tabla abajo presenta el significado de los datos para las instancias 20/70 y 21/71.

Monitoreo:

Bits (Byte 0)	Valores
Bit 0 Faulted	0: Convertidor no está en estado de falla. 1: Alguna falla registrada por el convertidor. Obs.: El número de la falla puede ser leído a través del parámetro P0049 – Falla Actual.
Bit 1 Warning	0: Convertidor no está en estado de alarma. 1: Algún alarma registrado por el convertidor. Obs.: El número del alarma puede ser leído a través del parámetro P0048 – Alarma Actual.
Bit 2 Running1 (Fwd)	0: El eje del motor no está girando en el sentido horario. 1: El eje del motor girando en el sentido horario.
Bit 3 Running2 (Rev)	0: El eje del motor no está girando en el sentido antihorario. 1: El eje del motor girando en el sentido antihorario.
Bit 4 Ready	0: Convertidor no está listo para operar. 1: Convertidor listo para operar (estados Ready, Enabled o Stopping).
Bit 5 Ctrl from Net	0: Drive controlado de modo local. 1: Drive controlado de modo remoto.
Bit 6 Ref. from Net	0: Referencia de velocidad no está siendo enviada vía red DeviceNet. 1: Indica referencia de velocidad siendo enviada vía red DeviceNet.
Bit 7 At Reference	0: Convertidor todavía no alcanzó la velocidad programada. 1: Convertidor alcanzó la velocidad programada.

- Byte 1 indica el estado del drive:
 - 0 = Non-Existent
 - 1 = Startup
 - 2 = Not_Ready
 - 3 = Ready
 - 4 = Enabled
 - 5 = Stopping
 - 6 = Fault_Stop
 - 7 = Faulted
- Bytes 2 (*low*) y 3 (*high*) representan la velocidad real del motor en RPM.

Control:

Bits (Byte 0)	Valores
Bit 0 Run Fwd	0: Para el motor. 1: Gira el eje del motor en el sentido horario.
Bit 1 Run Rev	0: Para el motor. 1: Gira el eje del motor en el sentido antihorario.
Bit 2 Fault Reset ⁷	0: Sin función. 1: Si en estado de falla, ejecuta el reset del convertidor.
Bits 3 e 4	Reservado.
Bit 5 NetCtrl ⁸	0: Selecciona el modo local. 1: Selecciona el modo remoto.
Bit 6 NetRef	0: Referencia de velocidad no está siendo enviada vía red. 1: Envío de la referencia de velocidad vía red.
Bit 7	Reservado.

- Bytes 2 (*low*) y 3 (*high*) representan la referencia de velocidad del motor en RPM.

2 = Formato de los datos para las instancias *Manufacturer Specific 2W* (2 palabras):

3 = Formato de los datos para las instancias *Manufacturer Specific 3W* (3 palabras):

4 = Formato de los datos para las instancias *Manufacturer Specific 4W* (4 palabras):

5 = Formato de los datos para las instancias *Manufacturer Specific 5W* (5 palabras):

6 = Formato de los datos para las instancias *Manufacturer Specific 6W* (6 palabras):

Llamada de *Manufacturer Specific*, estas instancias representan el más sencillo interfaz de operación del equipamiento segundo el perfil del convertidor de frecuencia MW500. El mapa de los datos es presentado abajo. Además de las palabras de comando y de monitoreo control y estado, referencia y valor actual de la velocidad, estas permiten programar hasta 4 parámetros del propio equipamiento para lectura y/o escrita vía red, a través de los parámetros P0711 hasta P0718.

Monitoreo (Entrada)

Instancia	Palabras de 16 bits (word)	Función	Opción del P0710
100	#1	Palabra de Estado (P0680)	2 3 4 5 6
	#2	Velocidad Actual (P0681)	
	#3	Lectura #3 DeviceNet	
	#4	Lectura #4 DeviceNet	
	#5	Lectura #5 DeviceNet	
	#6	Lectura #6 DeviceNet	

Control (Salida)

Instancia	Palabras de 16 bits (word)	Función	Opción del P0710
100	#1	Palabra de Controle (P0684)	2 3 4 5 6
	#2	Referencia de Velocidad (P0685)	
	#3	Escrita #3 DeviceNet	
	#4	Escrita #4 DeviceNet	
	#5	Escrita #5 DeviceNet	
	#6	Escrita #6 DeviceNet	

⁷ Luego del reset, el drive irá para el estado offline.

⁸ Para que esta acción sea ejecutada, los parámetros P220 – P228 deben estar correctamente programados.

P0711 – LECTURA #3 DEVICENET**P0712 – LECTURA #4 DEVICENET****P0713 – LECTURA #5 DEVICENET****P0714 – LECTURA #6 DEVICENET**

Rango de	0 a 1199	Padrón: 0 (deshabilitado)
Valores:		
Propiedades:	CFG, CAN	
Grupo de acceso vía HMI:	NET	

Descripción:

Estos parámetros permiten programar el contenido de las palabras 3 a 6 de entrada (input: esclavo envía para el maestro). Utilizando estos parámetros, es posible programar el número de otro parámetro cuyo contenido debe ser disponibilizado en el área de entrada del maestro de la red.

Por ejemplo, caso se desee leer del drive la corriente del motor en amperes, se debe programar en alguno de los parámetros el valor 3, pues el parámetro P0003 es el parámetro que contiene esta información. Vale recordar que el valor leído de cualquier parámetro es representado con una palabra de 16 bits. Mismo que el parámetro posea resolución decimal, el valor es transmitido sin la indicación de los decimales. Por ejemplo, si el parámetro P0003 posee el valor 4,7 A, el valor suministrado vía red será 47.

Estos parámetros son utilizados solamente si el drive fuera programado en el parámetro P0710 para utilizar las opciones 3 hasta 6. De acuerdo con la opción seleccionada, son disponibilizadas hasta 6 palabras para lectura por el maestro de la red.

Las dos primeras palabras de entrada son fijas.

**¡NOTA!**

El valor 0 (cero) deshabilita la escritura en la palabra. La cantidad de palabras de entrada, por otro lado, permanece siempre igual a lo que fuera programado en el parámetro P0710.

P0715 – ESCRITA #3 DEVICENET**P0716 – ESCRITA #4 DEVICENET****P0717 – ESCRITA #5 DEVICENET****P0718 – ESCRITA #6 DEVICENET**

Rango de	0 a 1199	Padrón: 0 (deshabilitado)
Valores:		
Propiedades:	CFG, CAN	
Grupo de acceso vía HMI:	NET	

Descripción:

Estos parámetros permiten programar el contenido de las palabras 3 a 6 de salida (output: maestro envía para el esclavo). Utilizando estos parámetros, es posible programar el número de otro parámetro cuyo contenido debe ser disponibilizado en el área de salida del maestro de la red.

Por ejemplo, caso se desee escribir en el drive la rampa de aceleración, se debe programar en alguno de los parámetros el valor 100, pues el parámetro P0100 es el parámetro donde esta información es programada. Vale recordar que el valor escrito de cualquier parámetro es representado con una palabra de 16 bits. Aunque el parámetro posea resolución decimal, el valor es transmitido sin la indicación de los decimales. Por ejemplo, caso se desee programar el parámetro P0100 con el valor 5,0s, el valor escrito vía red deberá ser 50.

Estos parámetros son utilizados solamente si el drive fuera programado en el parámetro P0710 para utilizar las opciones 3 hasta 6. De acuerdo con la opción seleccionada, son disponibilizadas hasta 6 palabras para escritura por el maestro de la red.

Las dos primeras palabras de salida son fijas.


¡NOTA!

El valor 0 (cero) deshabilita la escritura en la palabra. La cantidad de palabras de salida, por otro lado, permanece siempre igual a lo que fue programado en el parámetro P0710.

P0719 – ESTADO DE LA RED DEVICENET

Rango de Valores:	0 = Offline 1 = Online, No Conectado 2 = Online, Conectado 3 = Conexión Caduco 4 = Falla en la Conexión 5 = Auto-baud	Padrón: -
Propiedades:	RO, CAN	
Grupo de acceso vía HMI:	NET	

Descripción:

Indica el estado de la red DeviceNet. La tabla que sigue presenta una breve descripción de estos estados.

Estado	Descripción
Offline	Sin alimentación o no online. Comunicación no puede ser establecida.
Online, No Conectado	Dispositivo online, más no conectado. Esclavo completó con suceso el procedimiento de verificación del MacID. Eso significa que la tasa de comunicación configurada está correcta (o fue detectada correctamente en el caso de la utilización del autobaud) y que no hay otros nodos en la red con la misma dirección. Sin embargo, en esta etapa, todavía no hay comunicación con el maestro.
Online, Conectado	Dispositivo operacional y en condiciones normales. Maestro colocó un conjunto de conexiones del tipo I/O con el esclavo. En esta etapa ocurre efectivamente el intercambio de datos a través de conexiones del tipo I/O.
Conexión Caduco	Una o más conexiones del tipo I/O caducaran.
Falla en la Conexión	Indica que el esclavo no puede entrar en la red debido a problemas de dirección o entonces debido a la ocurrencia de bus off. Verifique si la dirección configurada ya no está siendo utilizada por otro equipamiento, si la tasa de comunicación elegida está correcta o si existen problemas en la instalación.
Auto-baud	Equipamiento ejecutando rutina del mecanismo de autobaud.

P0720 – ESTADO DEL MAESTRO DEVICENET

Rango de Valores:	0 = Run 1 = Idle	Padrón: -
Propiedades:	RO, CAN	
Grupo de acceso vía HMI:	NET	

Descripción:

Indica el estado del maestro de la red DeviceNet. Este puede estar en el modo de operación (*Run*) o en el modo de configuración (*Idle*).

Cuando en *Run*, telegramas de lectura y escrita son procesados y actualizados normalmente por el maestro. Cuando en *Idle*, solamente telegramas de lectura de los esclavos son actualizados por el maestro. La escrita, en este caso, se queda deshabilitada.

Cuando la comunicación estuviera deshabilitada este parámetro no representa el estado real del maestro.

5 CLASES DE OBJETOS SOPORTADAS

Todo dispositivo DeviceNet es modelado por un conjunto de objetos. Son ellos los responsables por definir que función, determinado equipamiento tendrá. O sea, de acuerdo con los objetos implementados, este equipamiento podrá ser un adaptador de comunicación, un drive AC/DC, un sensor fotoeléctrico, etc.. Objetos obligatorios y opcionales son definidos en cada uno de estos perfiles de dispositivos (Device Profile). El convertidor de frecuencia MW500 soporta todas las clases obligatorias del perfil AC/DC Device Profile. Soporta también clases específicas del fabricante. Detalles de cada una de ellas son presentados en las secciones a seguir.

5.1 CLASE IDENTITY (01H)

Suministra informaciones generales sobre la identidad del dispositivo, tales como VendorID, Product Name, Serial Number, etc. Están implementados los siguientes atributos:

Tabla 5.1: Atributos de la instancia de la Clase Identity

Atributo	Método	Nombre	Padrón	Descripción
1	Get	Vendor ID	355h	Identificador del fabricante
2	Get	Product Type	2h	Tipo del producto
3	Get	Product Code		Código del producto
4	Get	Vendor Revision		Revisión del firmware
5	Get	Status		Estado actual del dispositivo
6	Get	Serial Number		Número serial
7	Get	Product Name	MW500	Nombre del producto

5.2 CLASE MESSAGE ROUTER (02H)

Suministra informaciones sobre el objeto roteador de mensajes del tipo explicit. En el MW500, esta clase no tiene cualquier atributo implementado.

5.3 CLASE DEVICENET (03H)

Responsable por mantener la configuración y el estado de las conexiones físicas del nodo DeviceNet. Están implementados los siguientes atributos:

Tabla 5.2: Atributos de la Clase DeviceNet

Atributo	Método	Nombre	Mín./Max	Padrón	Descripción
1	Get	Revision	1-65535		Revisión de la definición del Objeto de Clase DeviceNet sobre cual la implementación fue basada

Tabla 5.3: Atributos de la instancia de la Clase DeviceNet

Atributo	Método	Nombre	Mín./Max	Padrón	Descripción
1	Get/Set	Mac ID	0-63	63	Dirección del nodo
2	Get/Set	Baud rate	0-2	0	Tasa de transmisión
3	Get/Set	Bus-Off Interrupt	0-1	1	Reset de bus-off
4	Get/Set	Bus-Off Counter	0-255		Contador de bus-off
5	Get	Allocation Information			Información sobre el allocation byte

5.4 CLASE ASSEMBLY (04H)

Clase cuya función es juntar diversos atributos en una sola conexión. En el MW500 solamente el atributo Data (3) está implementado (Tabla 5.4).

Tabla 5.4: Atributos de las instancias de la Clase Assembly

Atributos	Método	Nombre	Descripción
3	Get/Set	Data	Datos de la instancia

En el MW500, la clase Assembly contiene las siguientes instancias:

Tabla 5.5: *Instancias de la Clase Assembly*

Instancias	Tamaño	Descripción
20	2 palabras	ODVA Basic Speed Control Output
70	2 palabras	ODVA Basic Speed Control Input
21	2 palabras	ODVA Extended Speed Control Output
71	2 palabras	ODVA Extended Speed Control Input
100	2 – 6 palabras	Especifica de lo Fabricante Salida
150	2 – 6 palabras	Especifica de lo Fabricante Entrada

5.5 CLASE CONNECTION (05H)

Instancia conexiones del tipo I/O y explicit. Están implementados los siguientes atributos:

5.5.1 Instancia 1: Explicit Message

Tabla 5.6: *Clase Connection – Instancia 1: Explicit Message*

Atributo	Método	Nombre	Descripción
1	Get	State	Estado del objeto
2	Get	Instance Type	I/O o explicit
3	Get	Transport Class Trigger	Define comportamiento de la conexión
4	Get	Produced Connection ID	Identificador CAN de transmisión
5	Get	Consumed Connection ID	Identificador CAN de recepción
6	Get	Initial Comm. Charac.	Define el grupo de mensajes asociado a esta conexión
7	Get	Produced Connection Size	Tamaño en bytes de esta conexión de transmisión
8	Get	Consumed Connection Size	Tamaño en bytes de esta conexión de recepción
9	Get/Set	Expected Packet Rate	Define valores de tiempo utilizado internamente
12	Get	Watchdog Timeout Action	Define como tratar contador Inactivity/Watchdog
13	Get	Produced Connection Path Length	Número de bytes de la conexión productora
14	Get	Produced Connection Path	Camino de los objetos productores de datos
15	Get	Consumed Connection Path Length	Número de bytes de la conexión consumidora
16	Get	Consumed Connection Path	Camino de los objetos consumidores de datos
17	Get/Set	Production Inhibit Time	Define el tiempo mínimo para nueva producción de datos

5.5.2 Instancia 2: Polled

Tabla 5.7: *Clase Connection – Instancia 2: Polled*

Atributo	Método	Nombre	Descripción
1	Get	State	Estado del objeto
2	Get	Instance Type	I/O o explicit
3	Get	Transport Class Trigger	Define comportamiento de la conexión
4	Get	Produced Connection ID	Identificador CAN de transmisión
5	Get	Consumed Connection ID	Identificador CAN de recepción
6	Get	Initial Comm. Charac.	Define el grupo de mensajes asociado a esta conexión
7	Get	Produced Connection Size	Tamaño en bytes de esta conexión de transmisión
8	Get	Consumed Connection Size	Tamaño en bytes de esta conexión de recepción
9	Get/Set	Expected Packet Rate	Define valores de tiempo utilizado internamente
12	Get	Watchdog Timeout Action	Define como tratar contador Inactivity/Watchdog
13	Get	Produced Connection Path Length	Número de bytes de la conexión productora
14	Get	Produced Connection Path	Camino de los objetos productores de datos
15	Get	Consumed Connection Path Length	Número de bytes de la conexión consumidora
16	Get	Consumed Connection Path	Camino de los objetos consumidores de datos
17	Get/Set	Production Inhibit Time	Define el tiempo mínimo para nueva producción de datos

5.5.3 Instancia 4: Change of State/Cyclic

Tabla 5.8: Clase Connection – Instance 4: Change of State/Cyclic

Atributo	Método	Nombre	Descripción
1	Get	State	Estado del objeto
2	Get	Instance Type	I/O o explicit
3	Get	Transport Class Trigger	Define comportamiento de la conexión
4	Get	Produced Connection ID	Identificador CAN de transmisión
5	Get	Consumed Connection ID	Identificador CAN de recepción
6	Get	Initial Comm. Charac.	Define el grupo de mensajes asociado a esta conexión
7	Get	Produced Connection Size	Tamaño en bytes de esta conexión de transmisión
8	Get	Consumed Connection Size	Tamaño en bytes de esta conexión de recepción
9	Get/Set	Expected Packet Rate	Define valores de tiempo utilizado internamente
12	Get	Watchdog Timeout Action	Define como tratar contador Inactivity/Watchdog
13	Get	Produced Connection Path Length	Número de bytes de la conexión productora
14	Get	Produced Connection Path	Camino de los objetos productores de datos
15	Get	Consumed Connection Path Length	Número de bytes de la conexión consumidora
16	Get	Consumed Connection Path	Camino de los objetos consumidores de datos
17	Get/Set	Production Inhibit Time	Define el tiempo mínimo para nueva producción de datos

5.6 CLASE MOTOR DATA (28H)

Clase que almacena datos de placa del motor conectado al producto. Están implementados los siguientes atributos:

Tabla 5.9: Atributos de la Clase Motor Data

Atributo	Método	Nombre	Mín./Max	Descripción
1	Get	Revision	1-65535	Revisión de la definición del Objeto de Clase Motor Data sobre cual la implementación foi basada
2	Get	Max Instance		Número máximo de instancias

Tabla 5.10: Atributos de la instancia de la Clase Motor Data

Atributo	Método	Nombre	Mín./Max	Unidade	Padrón	Descripción
3	Get	Motor Type	0-10		7	0 = Non Standard Motor 1 = PM DC Motor 2 = FC DC Motor 3 = PM Synchronous Motor 4 = FC Synchronous Motor 5 = Switched Reluctance Motor 6 = Wound Rotor Induction Motor 7 = Squirrel Cage Induction Motor 8 = Stepper Motor 9 = Sinusoidal PM BL Motor 10 = Trapezoidal PM BL Motor
6	Get/Set	Rated Current	0-999.9	100mA		Corriente nominal
7	Get/Set	Rated Voltage	0-600	V		Tensión nominal

5.7 CLASE CONTROL SUPERVISOR (29H)

Responsable por modelar funciones de control del drive. Están implementados los siguientes atributos:

Tabla 5.11: Atributos de la Clase Control Supervisor

Atributo	Método	Nombre	Mín./Max	Descripción
1	Get	Revision	1-65535	Revisión de la definición del Objeto de Clase Control Supervisor sobre cual la implementación fue basada
2	Get	Max Instance		Número máximo de instancias

Tabla 5.12: Atributos de la instancia de la Clase Control Supervisor

Atributo	Método	Nombre	Mín./Max	Padrón	Descripción
3	Get/Set	Run1	0-1		Run Fwd
4	Get/Set	Run2	0-1		Run Rev
5	Get/Set	NetCtrl	0-1	0	0 = Control local 1 = Control vía red
6	Get	State	0-7		0 = Vendor specific 1 = Startup 2 = Not Ready 3 = Ready 4 = Enabled 5 = Stopping 6 = Fault Stop 7 = Fault
7	Get	Running1	0-1	0	0 = Otro estado 1 = (Enabled y Run1) o (Stopping y Running1) o (Fault Stop y Running1)
8	Get	Running2	0-1	0	0 = Otro estado 1 = (Enabled y Run2) o (Stopping y Running2) o (Fault Stop y Running2)
9	Get	Ready	0-1	0	0 = Otro estado 1 = Ready o Enabled o Stopping
10	Get	Faulted	0-1	0	0 = Sin fallos 1 = Fallo ocurrido
11	Get	Warning	0	0	0 = Sin warnings
12	Get/Set	Fault Reset	0-1	0	0 = Sin acción 0 -> 1 = Reset de errores
15	Get	Ctrl from Net	0-1	0	0 = Control es local 1 = Control es vía red

5.8 CLASE AC/DC DRIVE (2AH)

Contiene informaciones específicas de un AC/DC Drive tales como modo de operación y escalas de velocidad y torque. Están implementados los siguientes atributos:

Tabla 5.13: Atributos de la Clase AC/DC Drive

Atributo	Método	Nombre	Mín./Max	Descripción
1	Get	Revision	1-65535	Revisión de la definición del Objeto de Clase AC/DC Drive sobre cual la implementación fue basada
2	Get	Max Instance		Número máximo de instancias

Tabla 5.14: Atributos de la instancia de la Clase AC/DC Drive

Atributo	Método	Nombre	Mín./Max	Padrón	Descripción
4	Get/Set	NetRef	0-1	0	0 = Referencia local 1 = Referencia vía red
6	Get	DriveMode	1-2	2	1 = Speed control (open loop) 2 = Speed control (closed loop)
7	Get	Speed Actual	0-9999		Velocidad real (mejor aproximación)
8	Get/Set	Speed Ref	0-9999	0	Referencia de velocidad


¡NOTA!

El MW500 operará en modo de velocidad independiente del contenido del atributo DriveMode.

5.9 CLASE ACKNOWLEDGE HANDLER (2BH)

La función de esta clase es controlar la recepción de mensajes de reconocimiento (acknowledge).

Tabla 5.15: Atributos de la instancia de la Clase Acknowledge Handler

Atributo	Método	Nombre
1	Get/Set	Acknowledge Timer
2	Get	Retry Limit
3	Get	COS Production Connection Instance

5.10 CLASES ESPECÍFICAS DEL FABRICANTE

Las clases específicas del fabricante son utilizadas para definir todos los parámetros del MW500. Ellas permiten que el usuario lea y escriba en cualquier parámetro a través de la red. Para esto, mensajes DeviceNet del tipo

explicit son utilizadas. Hay rangos separadas para cada grupo de parámetros, conforme presentado en la Tabla 5.16:

Tabla 5.16: Clases Específicas del Fabricante

Clase	Nombre	Rango de valores
Clase 100 (64h)	VENDOR CLASS F1	Parámetros 000-099
Clase 101 (65h)	VENDOR CLASS F2	Parámetros 100-199
Clase 102 (66h)	VENDOR CLASS F3	Parámetros 200-299
Clase 103 (67h)	VENDOR CLASS F4	Parámetros 300-399
Clase 104 (68h)	VENDOR CLASS F5	Parámetros 400-499
Clase 105 (69h)	VENDOR CLASS F6	Parámetros 500-599
Clase 106 (6Ah)	VENDOR CLASS F7	Parámetros 600-699
Clase 107 (6Bh)	VENDOR CLASS F8	Parámetros 700-799
Clase 108 (6Ch)	VENDOR CLASS F9	Parámetros 800-899
Clase 109 (6Dh)	VENDOR CLASS F10	Parámetros 900-999
Clase 110 (6Eh)	VENDOR CLASS F11	Parámetros 1000-1099

Tabla 5.17: Parámetros das Clases específicas del fabricante

Parámetro	Clase	Instancia	Atributo
P0000	Clase 100 (64h)	1	100
P0001	Clase 100 (64h)	1	101
P0002	Clase 100 (64h)	1	102
...
P0100	Clase 101 (65h)	1	100
P0101	Clase 101 (65h)	1	101
P0102	Clase 101 (65h)	1	102
...
P0200	Clase 102 (66h)	1	100
P0201	Clase 102 (66h)	1	101
P0202	Clase 102 (66h)	1	102
...
P0300	Clase 103 (67h)	1	100
P0301	Clase 103 (67h)	1	101
P0302	Clase 103 (67h)	1	102
...



¡NOTA!

- Para estas clases, el MW500 utiliza solamente la instancia 1.
- También para estas clases, los parámetros son accedidos adicionando el valor decimal 100 a los dígitos de la decena de cualquier parámetro. Este nuevo número es llamado de atributo.

Por ejemplo:

Parámetro 23: Clase 64h, instancia 1, atributo 123. Este camino da acceso al P0023.
 Parámetro 100: Clase 65h, instancia 1, atributo 100. Este camino da acceso al P0100.
 Parámetro 202: Clase 66h, instancia 1, atributo 102. Este camino da acceso al P0202.

6 FALLAS Y ALARMAS RELACIONADAS CON LA COMUNICACIÓN DEVICENET

A133/F233 – SIN ALIMENTACIÓN EN EL INTERFAZ CAN

Descripción:

Indica que el interfaz CAN no posee alimentación entre los terminales 1 y 5 del conector.

Actuación:

Para que sea posible enviar y recibir telegramas a través del interfaz CAN, es necesario suministrar alimentación externa para el circuito de interfaz.

Si la interfaz CAN estuviera alimentada y si fuera detectada la falta de alimentación en el interfaz CAN, será señalizada a través del HMI el mensaje de alarma A133 – o falla F233; dependiendo de la programación hecha en el parámetro P0313. Si la alimentación del circuito es restablecida, la comunicación CAN será reiniciada. Para la condición de alarma, la señalización desaparecerá automáticamente del HMI en el momento que la alimentación del circuito es reestablecida.

Posibles Causas/Corrección:

- Medir si hay tensión entre los terminales 1 y 5 del conector del interfaz CAN.
- Verificar si los cables de alimentación no están cambiados o invertidos.
- Verificar problemas de contacto en el cable o en el conector de la interfaz CAN.

A134/F234 – BUS OFF

Descripción:

Detectado error de *bus off* en el interfaz CAN.

Actuación:

Caso el número de errores de recepción o transmisión detectados por el interfaz CAN sea muy alto⁹, el controlador CAN puede ser llevado al estado de *bus off*, donde él interrumpe la comunicación y deshabilita la interfaz CAN.

En este caso será señalizada a través del HMI el mensaje de alarma A134 – o falla F234, dependiendo de la programación hecha en el parámetro P0313. Para que la comunicación sea restablecida es necesario interrumpir y regresar nuevamente la alimentación para el producto o interrumpir y regresar nuevamente la alimentación del interfaz CAN; para que la comunicación sea reiniciada.

Posibles Causas/Corrección:

- Verificar cortocircuito en los cables de transmisión del circuito CAN.
- Verificar si los cables no están cambiados o invertidos.
- Verificar si todos los dispositivos de la red utilizan la misma tasa de comunicación.
- Verificar si resistores de terminación con valores correctos fueran colocados solamente en los extremos del bus principal.
- Verificar si la instalación de la red CAN fue hecha de manera adecuada.

A136/F236 – MAESTRO EN IDLE

Descripción:

Alarma que señala que el maestro de la red DeviceNet está en modo *Idle*.

Actuación:

Actúa cuando el MW500 detecta que el maestro de la red fue para el modo *Idle*. En este modo, solamente las variables leídas del esclavo continúan siendo actualizadas en la memoria del maestro. Ninguno de los comandos enviados al esclavo es procesado.

⁹ Para más informaciones a respecto de la detección de errores, consultar la especificación CAN.

En este caso será señalizado a través del HMI el mensaje de alarma A136 – o falla F236 dependiendo de la programación hecha en el parámetro P0313. Para la condición de alarma, caso el maestro regrese nuevamente para el modo *Run* (estado normal de operación del equipamiento), la señalización de alarma desaparecerá automáticamente del HMI.

Posibles Causas/Corrección:

Ajuste la llave que comanda el modo de operación del maestro para ejecución (Run) o entonces el bit correspondiente en la palabra de configuración del software del maestro. En caso de dudas, consulte la documentación del maestro en uso.

A137/F237 – TIMEOUT EN LA CONEXIÓN DEVICENET

Descripción:

Alarma que señala que una o más conexiones I/O DeviceNet caducaran.

Actuación:

Ocurre cuando, por algún motivo, luego que iniciada la comunicación cíclica del maestro con el producto, esta comunicación fuera interrumpida.

En este caso será señalizada a través del HMI el mensaje de alarma A137 – o falla F237 dependiendo de la programación hecha en el parámetro P0313. Para la condición de alarma, caso la conexión con el maestro sea restablecida, la señalización de alarma desaparecerá automáticamente del HMI.

Posibles Causas/Corrección:

- Verificar el estado del maestro de la red.
- Verificar instalación de la red, cable roto o falla/mal contacto en las conexiones con la red.



WEG Drives & Controls - Automação LTDA.
Jaraguá do Sul - SC - Brasil
Teléfono 55 (47) 3276-4000 - Fax 55 (47) 3276-4020
São Paulo - SP - Brasil
Teléfono 55 (11) 5053-2300 - Fax 55 (11) 5052-4212
automacao@weg.net
www.weg.net