



# Manual de la Comunicación CANopen

Convertidor de Frecuencia

Serie: CFW-11

0899.5748 S/1



# Sumario

<b>A RESPECTO DEL MANUAL.....</b>	<b>5</b>
ABREVIACIONES Y DEFINICIONES .....	5
REPRESENTACIÓN NUMÉRICA.....	5
DOCUMENTOS .....	5
<b>1 INTRODUCCIÓN A LA COMUNICACIÓN CANOPEN .....</b>	<b>6</b>
1.1 CAN.....	6
1.1.1 <i>Frame de Datos</i> .....	6
1.1.2 <i>Frame Remoto</i> .....	6
1.1.3 <i>Acceso a la Red</i> .....	6
1.1.4 <i>Control de Errores</i> .....	6
1.1.5 <i>CAN y CANopen</i> .....	7
1.2 CARACTERÍSTICAS DE LA RED CANOPEN .....	7
1.3 MEDIO FÍSICO .....	7
1.4 DIRECCIÓN EN LA RED CANOPEN .....	7
1.5 ACCESO A LOS DATOS.....	7
1.6 TRANSMISIÓN DE DATOS .....	8
1.7 OBJETOS RESPONSABLES POR LA COMUNICACIÓN – COBS .....	8
1.8 COB-ID .....	9
1.9 ARCHIVO EDS.....	9
<b>2 KITS OPCIONALES.....</b>	<b>10</b>
2.1 INTERFACES CAN .....	10
2.1.1 <i>Kit CAN-01</i> .....	10
2.1.2 <i>Kit CAN/RS485-01</i> .....	10
2.1.3 <i>Terminales del Conector</i> .....	10
2.1.4 <i>Fuente de Alimentación</i> .....	11
2.1.5 <i>Resistor de Terminación</i> .....	11
2.1.6 <i>Tasa de Comunicación</i> .....	11
2.1.7 <i>Conexión del Convertidor en la Red CAN</i> .....	11
<b>3 PARAMETRIZACIÓN DEL CONVERTIDOR.....</b>	<b>13</b>
3.1 SÍMBOLOS PARA DESCRIPCIÓN DE LAS PROPIEDADES .....	13
P0220 – SELECCIÓN FUENTE LOCAL/REMOTO .....	13
P0221 – SELECCIÓN REFERENCIA LOCAL .....	13
P0222 – SELECCIÓN REFERENCIA REMOTA .....	13
P0223 – SELECCIÓN GIRO LOCAL.....	13
P0224 – SELECCIÓN GIRA/PARA LOCAL.....	13
P0225 – SELECCIÓN JOG LOCAL .....	13
P0226 – SELECCIÓN GIRO REMOTO.....	13
P0227 – SELECCIÓN GIRA/PARA REMOTO .....	13
P0228 – SELECCIÓN JOG REMOTO .....	13
P0313 – ACCIÓN PARA ERROR DE COMUNICACIÓN .....	13
P0680 – ESTADO LÓGICO.....	14
P0681 – VELOCIDAD DEL MOTOR EN 13 BITS.....	15
P0684 – PALABRA DE CONTROL VÍA CANOPEN/DEVICENET .....	15
P0685 – REFERENCIA DE VELOCIDAD VÍA CANOPEN/DEVICENET.....	16
P0695 – VALOR PARA LAS SALIDAS DIGITALES .....	17
P0696 – VALOR 1 PARA SALIDAS ANALÓGICAS.....	17
P0697 – VALOR 2 PARA SALIDAS ANALÓGICAS.....	17
P0698 – VALOR 3 PARA SALIDAS ANALÓGICAS.....	17
P0699 – VALOR 4 PARA SALIDAS ANALÓGICAS.....	17
P0700 – PROTOCOLO CAN .....	18
P0701 – DIRECCIÓN CAN .....	18
P0702 – TASA DE COMUNICACIÓN CAN .....	19
P0703 – RESET DE <i>BUS OFF</i> .....	19
P0705 – ESTADO DEL CONTROLADOR CAN.....	20

P0706 – CONTADOR DE TELEGRAMAS CAN RECIBIDOS .....	20
P0707 – CONTADOR DE TELEGRAMAS CAN TRANSMITIDOS .....	21
P0708 – CONTADOR DE ERRORES DE BUS OFF .....	21
P0709 – CONTADOR DE MENSAJES CAN PERDIDAS .....	21
P0721 – ESTADO DE LA COMUNICACIÓN CANOPEN .....	22
P0722 – ESTADO DEL NUDO CANOPEN .....	22
<b>4 DICCIONARIO DE OBJETOS .....</b>	<b>23</b>
4.1 ESTRUCTURA DEL DICCIONARIO .....	23
4.2 TIPOS DE DATOS .....	23
4.2.1 <i>Tipos Básicos</i> .....	23
4.2.2 <i>Tipos Compuestos</i> .....	24
4.2.3 <i>Tipos Extendidos</i> .....	24
4.3 <i>COMMUNICATION PROFILE</i> – OBJETOS PARA COMUNICACIÓN .....	25
4.4 <i>MANUFACTURER SPECIFIC</i> – OBJETOS ESPECÍFICOS DEL CFW-11 .....	25
4.5 <i>DEVICE PROFILE</i> – OBJETOS COMUNES PARA DRIVES .....	26
<b>5 DESCRIPCIÓN DE LOS OBJETOS DE COMUNICACIÓN .....</b>	<b>27</b>
5.1 OBJETOS DE IDENTIFICACIÓN .....	27
5.1.1 <i>Objeto 1000h – Device Type</i> .....	27
5.1.2 <i>Objeto 1001h – Error Register</i> .....	27
5.1.3 <i>Objeto 1018h – Identity Object</i> .....	28
5.2 SERVICE DATA OBJECTS – SDOS .....	28
5.2.1 <i>Objeto 1200h – Servidor SDO</i> .....	29
5.2.2 <i>Funcionamiento de los SDOs</i> .....	29
5.3 PROCESS DATA OBJECTS – PDOS .....	30
5.3.1 <i>Objetos mapeables para los PDOs</i> .....	31
5.3.2 <i>PDOs de recepción</i> .....	32
5.3.3 <i>PDOs de transmisión</i> .....	35
5.4 EMERGENCY OBJECT – EMCY.....	38
5.5 SYNCHRONIZATION OBJECT – SYNC .....	38
5.6 NETWORK MANAGEMENT – NMT .....	39
5.6.1 <i>Control de los estados del esclavo</i> .....	39
5.6.2 <i>Control de Errores – Node Guarding</i> .....	40
5.6.3 <i>Control de Errores – Heartbeat</i> .....	42
5.7 PROCEDIMIENTO DE INICIALIZACIÓN .....	44
<b>6 DESCRIPCIÓN DE LOS OBJETOS PARA DRIVES.....</b>	<b>45</b>
6.1 <i>DEVICE CONTROL</i> – OBJETOS PARA CONTROL DEL DRIVE .....	46
6.1.1 <i>Objeto 6040h – Controlword</i> .....	47
6.1.2 <i>Objeto 6041h – Statusword</i> .....	48
6.1.3 <i>Objeto 6060h – Modes of operation</i> .....	49
6.1.4 <i>Objeto 6061h – Modes of operation display</i> .....	49
6.2 <i>VELOCITY MODE</i> – OBJETOS PARA CONTROL DEL DRIVE.....	50
6.2.1 <i>Bits de control y estado</i> .....	50
6.2.2 <i>Objeto 6042h – vl target velocity</i> .....	50
6.2.3 <i>Objeto 6043h – vl velocity demand</i> .....	50
6.2.4 <i>Objeto 6044h – vl control effort</i> .....	51
6.2.5 <i>Objeto 6046h – vl velocity min max amount</i> .....	51
6.2.6 <i>Objeto 6048h – vl velocity acceleration</i> .....	51
6.2.7 <i>Objeto 6049h – vl velocity deceleration</i> .....	52
<b>7 FALLAS Y ALARMAS RELACIONADAS CON LA COMUNICACIÓN CANOPEN .....</b>	<b>53</b>
A133 – SIN ALIMENTACIÓN EN LA INTERFACE CAN .....	53
A134 – <i>BUS OFF</i> .....	53
A135 – ERROR DE <i>NODE GUARDING</i> .....	53

## A RESPECTO DEL MANUAL

Este manual suministra la descripción necesaria para la operación del convertidor de frecuencia CFW-11 utilizando el protocolo CANopen. Este manual debe ser utilizado en conjunto con el manual del usuario del CFW-11.

## ABREVIACIONES Y DEFINICIONES

CAN	Controller Area Network
CiA	CAN in Automation
COB	Communication Object
COB-ID	Communication Object Identifier
SDO	Service Data Object
PDO	Process Data Object
RPDO	Receive PDO
TPDO	Transmit PDO
NMT	Network Management Object
ro	Read only (solamente 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' después del número.

## DOCUMENTOS

El protocolo CANopen para CFW-11 fue desarrollado basado en las siguientes especificaciones y documentos:

Documento	Versión	Fuente
CAN Specification	2.0	CiA
CiA DS 301 CANopen Application Layer and Communication Profile	4.02	CiA
CiA DRP 303-1 Cabling and Connector Pin Assignment	1.1.1	CiA
CiA DSP 306 Electronic Data Sheet Specification for CANopen	1.1	CiA
CiA DSP 402 Device Profile Drives and Motion Control	2.0	CiA

Para obtener esta documentación, se debe consultar la CiA, que actualmente es la organización que mantiene, divulga y actualiza las informaciones relativas a la red CANopen.

# 1 INTRODUCCIÓN A LA COMUNICACIÓN CANOPEN

Para la operación del convertidor de frecuencia CFW-11 en red CANopen, es necesario conocer el modo como la comunicación es hecha. Para eso, este ítem trae una descripción general del funcionamiento del protocolo CANopen, conteniendo las funciones utilizadas por el CFW-11. Para una descripción detallada del protocolo, consulte la documentación CANopen indicada en el ítem anterior.

## 1.1 CAN

La red CANopen es una red basada en CAN, lo que significa decir que ella utiliza telegramas CAN para intercambiar 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)<sup>1</sup>. En esta camada, son definidos los diferentes tipos de telegramas (*frames*), el modo de detección de errores, validación y determinación 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 bits<sup>2</sup> (*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. Funciona como una requisición para que otro dispositivo de la red transmita el frame de datos deseado.

### 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 prioritario. 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 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:

- ☑ **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 elle para de actuar en la red al detectar que un otro dispositivo envió un telegrama con error.
- ☑ **Bus Off:** por último, tenemos el estado de *bus off*, en el cual el dispositivo no irá más enviar o no irá más recibir telegramas.

---

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

<sup>2</sup> La especifica CAN 2.0 define dos tipos de frames de datos: *standard* (11 bits) y *extended* (29 bits). Para el protocolo CANopen del CFW-11, solamente frames *standard* son aceptos.

### **1.1.5 CAN y CANopen**

Solamente la definición de como detectar errores, crear y transmitir un frame no es suficiente para definir un significado para los datos que son enviados vía red. Es necesario que se tenga una especificación que indique como el identificador y los datos deben ser montados y como las informaciones deben ser intercambiadas. De esto modo los elementos de la red pueden interpretar correctamente los datos que son transmitidos. En esto sentido, la especificación CANopen 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, J1939, etc., que también 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 CANOPEN**

Por utilizar un bus CAN como modo de transmisión de telegramas, todos los dispositivos de la red CANopen poseen los mismos derechos de acceso a la red, donde la prioridad del identificador es responsable por solucionar problemas de conflicto cuando accesos simultáneos ocurren. Eso trae el beneficio de posibilitar la comunicación directa entre esclavos de la red, además del hecho de que los datos pueden ser dispuestos de manera más optimizada, sin la necesidad de un maestro que haga el control de toda la comunicación, haciendo acceso cíclico a todos los dispositivos de la red para la actualización de los datos.

Otra característica importante es la utilización del modelo productor / consumidor para la transmisión de datos. Eso significar decir que un mensaje que transita en la red no posee una dirección fija en la red como destino. Este mensaje posee un identificador que indica cual el dato que ella está transportando. Cualquier elemento de la red que necesite utilizarse de esta información para la suya lógica de operación, podrá consumirla y, por lo tanto, un mismo mensaje puede ser utilizada por varios elementos de la red al mismo tiempo.

## **1.3 MEDIO FÍSICO**

El medio físico para la transmisión de las señales en una red CANopen es especificado por la norma ISO 11898. Ella define como bus de transmisión un par de cables tranzado con señal eléctrico diferencial.

El convertidor de frecuencia CFW-11 utiliza todavía un circuito de interface con la red aislada. La fuente de alimentación para la interface CANopen es compartida con las entradas y salidas digitales y analógicas presente en la tarjeta de control del CFW-11. El componente responsable por la transmisión y recepción de las señales es denominado transceiver, que sigue el especificado por la ISO 11898.

## **1.4 DIRECCIÓN EN LA RED CANOPEN**

Toda la red CANopen debe poseer un maestro, responsable por servicios de gestión de la red, y también puede poseer un conjunto de hasta 127 esclavos. Cada dispositivo de la red también puede ser llamado de nudo. Todo esclavo en una red CANopen es identificado a través de su dirección, o Node-ID, que debe ser único para cada esclavo de la red, y puede cambiar de 1 hasta 127.

El CFW-11 no posee funciones que implementan los servicios de gestión de red, y por lo tanto elle debe ser utilizado en conjunto con algún equipamiento que posea tales servicios, en general un maestro de la red CANopen.

## **1.5 ACCESO A LOS DATOS**

Cada esclavo de la red CANopen posee un listado, denominado diccionario de objetos, que contiene todos los datos que son accesibles vía red. Cada objeto de este listado es identificado a través de un índice, y durante la configuración del equipamiento e intercambio de mensajes, este índice es utilizado para identificar lo que esta siendo transmitido.

Una descripción más detallada de como el diccionario de objetos está estructurado es presentado en el ítem 6.

## 1.6 TRANSMISIÓN DE DATOS

La transmisión de datos numéricos a través de telegramas CANopen es hecha utilizando la representación hexadecimal del número, y enviando el byte menos significativo del dato primer.

Ejemplo: transmisión de un entero con señal de 32 bits (12345678h = 305419896 decimal), más un entero con señal de 16 bits (FF00h = -256 decimal), en un frame CAN.

identificador	6 bytes de datos					
11 bits	Entero 32 bits				Entero 16 bits	
	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5
	78h	56h	34h	12h	00h	FFh

## 1.7 OBJETOS RESPONSABLES POR LA COMUNICACIÓN – COBS

Existe un determinado conjunto de objetos que son responsables por la comunicación entre los dispositivos de la red. Estos objetos están divididos de acuerdo con los tipos de datos y el modo como son enviados o recibidos por un dispositivo. El CFW-11 soporta los siguientes objetos de comunicación (COBs):

Tipo de Objeto	Descripción
<i>Service Data Object</i> (SDO)	Los SDOs son objetos responsables por el acceso directo al diccionario de objetos de un dispositivo. A través de mensajes utilizando los SDOs, es posible indicar explícitamente (a través del índice del objeto), cual el dato que está siendo manipulado. Existen dos tipos de SDOs: Cliente SDO, responsable por hacer una requisición de lectura o de escrita para un dispositivo de la red, y el Servidor SDO, responsable por atender esta requisición. Como los SDOs son utilizados generalmente para configuración de un nudo de la red, son menos prioritarios que otros tipos de mensajes. Solamente un SDO del tipo servidor está disponible para el CFW-11.
<i>Process Data Object</i> (PDO)	Los PDOs son utilizados para acceder datos del equipamiento sin la necesidad de indicar explícitamente cual es el objeto del diccionario que está siendo accedido. Para eso, es necesario configurar previamente cuales son los datos que el PDO estará transmitiendo (mapeo de los datos). También existen dos tipos de PDOs: PDO de recepción y PDO de transmisión. PDOs usualmente son utilizados para transmisión y recepción de datos utilizados durante la operación del dispositivo, y por eso son más prioritarios que los SDOs.
<i>Emergency Object</i> (EMCY)	Este objeto es responsable por el envío de mensajes para indicar la ocurrencia de errores en el dispositivo. Cuando un error ocurre en un determinado dispositivo (Productor EMCY), este puede enviar un mensaje para la red. Caso algún dispositivo de la red se encuentre monitoreando este mensaje (Consumidor EMCY), es posible programar para que una acción sea tomada (deshabilitar demás dispositivos de la red, reset de errores, etc.). El CFW-11 posee solo la funcionalidad de productor EMCY.
<i>Synchronization Object</i> (SYNC)	En la red CANopen es posible programar un dispositivo (Productor SYNC) para enviar, periódicamente, un mensaje de sincronización para todos los dispositivos de la red. Estos dispositivos (Consumidores SYNC) pueden entonces, por ejemplo, enviar un determinado dato que necesita estar disponible periódicamente. El CFW-11 posee la función de consumidor SYNC.
<i>Network Management</i> (NMT)	Toda la red CANopen precisa tener un maestro que haga el control de los demás dispositivos de la red (esclavos). Este maestro será responsable por un conjunto de servicios que controlan la comunicación de los esclavos y su estado en la red CANopen. Los esclavos son responsables por recibir los comandos enviados por el maestro y ejecutar las acciones solicitadas. El CFW-11 opera como un esclavo de la red CANopen, y dispone dos tipos de servicios que el maestro puede utilizar: servicio de control del dispositivo, donde el maestro controla el estado de cada esclavo en la red, y servicios de control de errores ( <i>Node Guarding</i> ), donde el esclavo envía mensajes periódicos para el maestro para informar que la conexión está activa.

*Tabla 1.1 – Tipos de Objetos de Comunicación (COBs)*

Toda la comunicación del convertidor con la red es hecha utilizándose estos objetos, y los datos que pueden ser accedidos son los existentes en el diccionario de objetos del dispositivo. La descripción del funcionamiento de cada COB es hecha en el ítem 7.



## 1.8 COB-ID

Un telegrama de la red CANopen siempre es transmitido por un objeto de comunicación (COB). Todo COB posee un identificador que indica el tipo de dato que está siendo transportado. Este identificador, llamado de COB-ID, posee un tamaño de 11 bits, y es transmitido en el campo identificador de un telegrama CAN. Elle puede ser subdividido en dos partes:

Código da Función				Dirección del nudo						
bit 10	bit 9	bit 8	bit 7	bit 6	bit 5	bit 4	bit 3	bit 2	bit 1	bit 0

- Código de la función: indica el tipo de objeto que está siendo transmitido.
- Dirección del nudo: indica con cual dispositivo de la red el telegrama está vinculado.

A seguir es presentada una tabla con los valores padrones para los diferentes objetos de comunicación disponibles en el CFW-11. Es necesario observar que el valor padrón del objeto depende del enderezo del esclavo, con excepción de los COB-IDs para NMT y SYNC, que son comunes para todos los elementos de la red. Estos valores también pueden ser alterados durante la etapa de configuración del dispositivo.

COB	Código de la Función (bits 10 – 7)	COB-ID Resultante (función + dirección)
NMT	0000	0
SYNC	0001	128 (80h)
EMCY	0001	129 – 255 (81h – FFh)
PDO1 (tx)	0011	385 – 511 (181h – 1FFh)
PDO1 (rx)	0100	513 – 639 (201h – 27Fh)
PDO2 (tx)	0101	641 – 767 (281h – 2FFh)
PDO2 (rx)	0110	769 – 895 (301h – 37Fh)
PDO3 (tx)	0111	897 – 1023 (381h – 3FFh)
PDO3 (rx)	1000	1025 – 1151 (401h – 47Fh)
PDO4 (tx)	1001	1153 – 1279 (481h – 4FFh)
PDO4 (rx)	1010	1281 – 1407 (501h – 57Fh)
SDO (tx)	1011	1409 – 1535 (581h – 5FFh)
SDO (rx)	1100	1537 – 1663 (601h – 67Fh)
Node Guarding/ Heartbeat	1110	1793 – 1919 (701h – 77Fh)

*Tabla 1.2 – COB-ID para los diferentes objetos*

## 1.9 ARCHIVO EDS

Cada dispositivo en una red CANopen posee un archivo de configuración EDS, que contiene las informaciones sobre el funcionamiento del dispositivo en la red CANopen, bien como la descripción de todos los objetos existentes para comunicación. En general este archivo es utilizado por un maestro o software de configuración, para programación de los dispositivos presentes en la red CANopen.

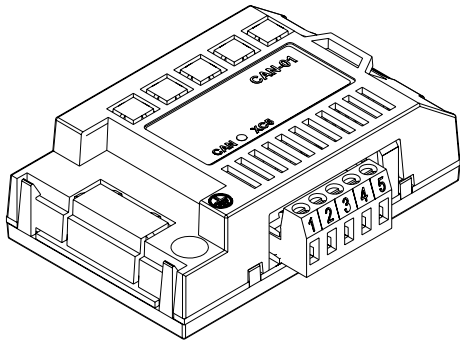
El archivo de configuración EDS para el CFW-11 es suministrado en conjunto con el producto, y también puede ser obtenido a través del sitio web <http://www.weg.net>. Es necesario observar la versión de software del convertidor, para utilizar un archivo EDS que sea compatible con esta versión.

## 2 KITS OPCIONALES

Para posibilitar la comunicación CANopen en el convertidor de frecuencia CFW-11, es necesario utilizar un de los kits para comunicación CAN descritos a seguir. Informaciones a respecto de la instalación de estos módulos en el convertidor pueden ser obtenidas en el guía rápido que acompaña el kit.

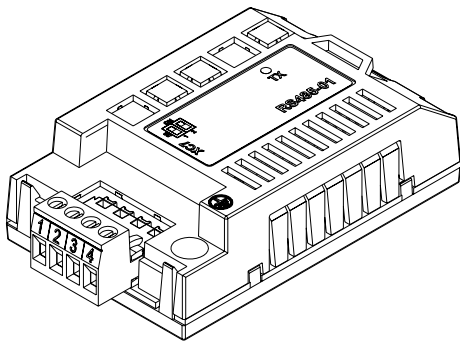
### 2.1 INTERFACES CAN

#### 2.1.1 Kit CAN-01



- ☑ Ítem WEG: 417107436.
- ☑ Formado por el módulo de comunicación CAN (figura al lado) más un guía rápido de montaje.
- ☑ Interface aislada galvanicamente y con señal diferencial, atribuyendo mayor robustez contra interferencia electromagnética.
- ☑ Alimentación externa de 24V.
- ☑ Permite la conexión de hasta 32 dispositivos en el mismo segmento. Una cantidad mayor de dispositivos puede ser conectada y con el uso de repetidores<sup>3</sup>.
- ☑ Longitud máximo del bus de 1000 metros.

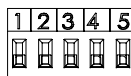
#### 2.1.2 Kit CAN/RS485-01



- ☑ Ítem WEG: 417107435.
- ☑ Formado por el módulo de comunicación CAN/RS485 (figura al lado) más un guía rápido de montaje.
- ☑ Posee las mismas características de la interface CAN-01, más una interface RS485, para aplicaciones donde sea necesario la operación en conjunto de las dos interfaces.

#### 2.1.3 Terminales del Conector

El módulo para comunicación CAN posee un conector *plug-in* de 5 vías (XC5) con los siguientes terminales:



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

*Tabla 2.1 – Terminales del conector XC5 para interface CAN*

<sup>3</sup> El número límite de equipamientos que pueden ser conectados en la red también depende del protocolo utilizado.

## 2.1.4 Fuente de Alimentación

La interface CAN para el CFW-11 necesita de una tensión de alimentación externa entre los terminales 1 y 5 del conector de la red. Para evitar problemas de diferencia de potencial (tensión) entre los dispositivos de la red, es recomendado que la red sea alimentada en uno solo punto, y la señal de alimentación sea llevada a todos los dispositivos a través de cable. Caso sea necesaria más de una fuente de alimentación, estas deben estar referenciadas al mismo punto. Los datos para consumo individual de corriente y de tensión de entrada son presentados en la tabla que sigue.

Tensión de alimentación ( $V_{CC}$ )		
Mínimo	Máximo	Recomendado
11	30	24
Corriente (mA)		
Mínimo	Máximo	Medio
20	50	30

*Tabla 2.2 – Características de la alimentación para interface CAN*

Los módulos de interface CAN poseen un LED en el color verde para señalar que la interface está alimentada.

## 2.1.5 Resistor de Terminación

Las extremidades del bus CAN deben poseer un resistor de terminación en el valor de  $120 \Omega / 0.25 W$ , conectando las señales CAN\_H y CAN\_L caso el drive sea el primero o el último elemento del segmento.

## 2.1.6 Tasa de Comunicación

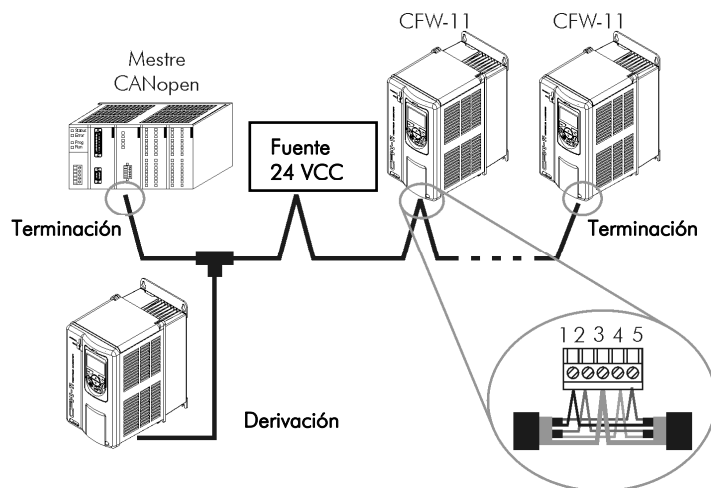
La tasa de comunicación (*baud rate*) que puede ser utilizada por un equipamiento en la red CANopen depende de la longitud del cable utilizado en la instalación. La tabla que sigue presenta las tasas de comunicación disponibles para el CFW-11, y la longitud máxima de cable que puede ser utilizado en la instalación; de acuerdo con el recomendado por la CiA.

Tasa de comunicación	Longitud del cable
1 Mbit/s	40 m
500 Kbit/s	100 m
250 Kbit/s	250 m
125 Kbit/s	500 m
100 Kbit/s	600 m
50 Kbit/s	1000 m
20 Kbit/s	1000 m
10 Kbit/s	1000 m

*Tabla 2.3 – Tasas de comunicación soportadas y longitud de la instalación*

## 2.1.7 Conexión del Convertidor en la Red CAN

Para interconectar los diversos nudos 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 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 2.1 – CFW-11 en red CANopen*

El cable para la conexión de las señales CAN\_L y CAN\_H debe tener impedancia característica de aproximadamente 120 Ohm, y un retraso máximo de propagación de la señal de 5ns/m. Otras características dependen de la longitud del cable, que debe estar de acuerdo con la tabla que sigue.

Longitud del cable (m)	Resistencia por metro (mOhm/m)	Área del conductor (mm <sup>2</sup> )
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

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.

### 3 PARAMETRIZACIÓN DEL CONVERTIDOR

A seguir serán presentados solo los parámetros del convertidor de frecuencia CFW-11 que poseen relación con la COMUNICACIÓN CANOPEN.

#### 3.1 SÍMBOLOS PARA DESCRIPCIÓN DE LAS PROPIEDADES

RO	Parámetro solamente de lectura.
CFG	Parámetro solamente modificado con el motor parado.
Net	Parámetro visible a través de la HMI si el convertidor posee interface de red instalada – RS232, RS485, CAN, Anybus-CC, Profibus – o si la interface USB se encuentra conectada.
Serial	Parámetro visible a través de la HMI si el convertidor posee interface RS232 o RS485 instalada.
CAN	Parámetro visible a través de la HMI si el convertidor posee interface CAN instalada.

#### 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 local y remoto del convertidor de frecuencia CFW-11. Para que el convertidor de frecuencia sea controlado a través de la interface Anybus-CC, se debe seleccionar una de las opciones 'Anybus-CC' disponibles en los parámetros.

La descripción detallada de estos parámetros se encuentra en el Manual de programación del CFW-11.

#### P0313 – Acción para Error de Comunicación

<b>Rango de Valores:</b>	0 = Inactivo 1 = Para por Rampa 2 = Deshabilita General 3 = Va para Local	<b>Padrón:</b> 0
--------------------------	--	------------------

**Propiedades:** CFG y Net

**Grupos de Acceso vía HMI:**

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 111 Estados/Comandos

#### Descripción:

Este parámetro permite seleccionar cual la acción que debe ejecutada el convertidor de frecuencia, caso un error de comunicación sea detectado.

Opciones	Descripción
0 = Inactivo	Ninguna acción es realizada, convertidor de frecuencia 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 convertidor de frecuencia es deshabilitado general, y el motor para por inercia.
3 = Va para Local	El convertidor es comandado para el modo local.

Tabla 3.1 – Valores para el parámetro P0313

Para la interface CAN utilizando el protocolo CANopen, son considerados errores de comunicación los siguientes eventos:

- Alarma A133: sin alimentación en la interface CAN.
- Alarma A134: *bus off*
- Alarma A135: error de comunicación CANopen (*Node Guarding*)

La descripción de estas alarmas es hecha en el ítem 7.

Las acciones descritas en este parámetro son ejecutadas a través de la escritura automática de los respectivos bits en el parámetro de control vía CAN – P0684. Para que la acción ejecutada tenga efecto, es necesario que el convertidor de frecuencia se encuentre programado para tener control vía interface CAN. Esta programación es hecha a través de los parámetros P0220 hasta P0228.

## P0680 – Estado Lógico

**Rango de Valores:** 0000h – FFFFh

**Padrón:** -

**Propiedades:** RO

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 111 Estados/Comandos

### Descripción:

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

Bits	15	14	13	12	11	10	9	8	7	6	5 a 0
<b>Función</b>	En Falla	Subtensión	Manual/ Automático	LOC/REM	JOG	Sentido de Giro	Habilitado General	Rampa Habilitada	En Alarma	En Modo de Configuración	Reservado

Bits	Valores
Bits 0 a 5	Reservado
Bit 6 En Modo de Configuración	0: convertidor de frecuencia operando normalmente. 1: convertidor en modo de configuración. Indica una condición especial en la cual el convertidor no puede ser habilitado: <ul style="list-style-type: none"> <li>• Ejecutando rutina de autoajuste.</li> <li>• Ejecutando rutina de puesta en marcha orientado.</li> <li>• Ejecutando función <i>copy</i> de la HMI.</li> <li>• Ejecutando rutina auto orientada de la tarjeta de memoria flash.</li> <li>• Posee incompatibilidad de parametrización.</li> </ul> <b>Obs.:</b> Es posible obtener la descripción exacta del modo especial de operación en el parámetro P0692.
Bit 7 En Alarma	0: convertidor no está en el estado de alarma. 1: convertidor está en el estado de alarma. <b>Obs.:</b> el número de la alarma puede ser leído a través del parámetro P0048 – Alarma Actual.
Bit 8 Rampa Habilitada	0: convertidor está ejecutando rampa de aceleración o desaceleración, o alcanzo la consigna de velocidad. 1: no posee rampa habilitada.
Bit 9 Habilitado General	0: convertidor está deshabilitado general. 1: convertidor está habilitado general y listo para girar el motor.
Bit 10 Sentido de Giro	0: motor girando en el sentido antihorario. 1: motor girando en el sentido horario.
Bit 11 JOG	0: función JOG inactiva. 1: función JOG activa.
Bit 12 LOC/REM	0: convertidor en modo local. 1: convertidor en modo remoto.
Bit 13 Manual/ Automático	0: en modo manual (función PID). 1: en modo automático (función PID).
Bit 14 Subtensión	0: sin subtensión. 1: con subtensión.
Bit 15 En Falla	0: convertidor de frecuencia no está en el estado de falla. 1: alguna falla registrada por el convertidor. <b>Obs.:</b> el número de la falla puede ser leído a través del parámetro P0049 – Falla Actual.

**Tabla 3.2** – Funciones de los bits para el parámetro P0680

## P0681 – Velocidad del Motor en 13 bits

**Rango de Valores:** -32768...32767

**Padrón:** -

**Propiedades:** RO

### Grupos de Acceso vía HMI

01 GRUPOS PARÁMETROS

└ 49 Comunicación

└└ 111 Estados/Comandos

### Descripción:

Permite el monitoreo de la velocidad del motor. Esta palabra utiliza resolución de 13 bits para representar la rotación nominal del motor:

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

Valores de velocidad en "rpm" intermedios o superiores pueden ser obtenidos utilizando esta escala. Por ejemplo, para un motor de 4 polos y 1800 "rpm" de rotación nominal, caso el valor leído sea 2048 (0800h), para obtener el valor en "rpm" se debe calcular:

- 8192 – 1800 rpm
  - 2048 – velocidad en rpm
- $$\text{velocidad en rpm} = \frac{1800 \times 2048}{8192}$$

velocidad en rpm = 450 rpm

Valores negativos para este parámetro indican que el motor esta girando en el sentido antihorario.

## P0684 – Palabra de Control vía CANopen/DeviceNet

**Rango de Valores:** 0000h – FFFFh

**Padrón:** 0000h

**Propiedades:** CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS

└ 49 Comunicación

└└ 111 Estados/Comandos

### Descripción:

Palabra de comando del convertidor vía interface serial. Este parámetro solamente puede ser modificado vía interface CAN (protocolos CANopen o DeviceNet). Para las demás fuentes (HMI, USB, Serial, etc.) se comporta como un parámetro solamente de lectura.

Para que los comandos escritos en este parámetro sean ejecutados, es necesario que el convertidor de frecuencia se encuentre programado para recibir el control vía CAN. Esta programación es hecha a través de los parámetros P0220 hasta P0228.

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

Bits	15 a 8	7	6 e 5	4	3	2	1	0
Función	Reservado	Reset de Fallas	Reservado	LOC/REM	JOG	Sentido de Giro	Habilita General	Habilita Rampa

Bits	Valores
Bit 0 Habilita Rampa	0: ejecuta comando de parada por rampa de desaceleración. 1: gira el motor de acuerdo con la rampa de aceleración hasta alcanzar el valor de la consigna de velocidad.
Bit 1 Habilita General	0: deshabilita general el convertidor, interrumpiendo la alimentación para el motor. 1: habilita general el convertidor, permitiendo la operación del motor.
Bit 2 Sentido de Giro	0: girar motor en el sentido opuesto al de la referencia. 1: girar 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: convertidor va para el modo local. 1: convertidor va para el modo remoto.
Bits 5 e 6	Reservado
Bit 7 Reset de Fallas	0: sin función. 1: si se encuentra en el estado de falla, ejecuta el "reset" del convertidor.
Bits 8 a 15	Reservado

*Tabla 3.3 – Funciones de los bits para el parámetro P0684.*

## P0685 – Referencia de Velocidad vía CANopen/DeviceNet

**Rango de Valores:** -32768...32767

**Propiedades:** CAN

**Grupos de Acceso vía HMI:**

01 GRUPOS PARÁMETROS

└ 49 Comunicación

└└ 111 Estados/Comandos

### Descripción:

Permite programar la consigna (referencia) de velocidad para el convertidor vía interface serial. Este parámetro solamente puede ser modificado vía interface CAN (protocolos CANopen o DeviceNet). Para las demás fuentes (HMI, USB, Serial, etc.) se comporta como un parámetro solamente de lectura.

Para que la referencia escrita en este parámetro sea utilizada, es necesario que el convertidor se encuentre programado para utilizar la referencia de velocidad vía serial. Esta programación es hecha en los parámetros P0221 y P0222.

Esta palabra utiliza resolución de 13 bits para representar la rotación nominal del motor:

- P0683 = 0000h (0 decimal) → consigna de velocidad = 0 rpm
- P0683 = 2000h (8192 decimal) → consigna de velocidad = rotación nominal

Valores de referencias intermedias o superiores pueden ser programados utilizando esta escala. Por ejemplo, para un motor de 4 polos y 1800 rpm de rotación nominal, caso se desea una referencia de 900 rpm, se debe calcular:

$$\begin{aligned} \text{Consigna en 13 bit} &= \frac{900 \times 8192}{1800} \\ \text{Consigna en 13 bit} &= 4096 \end{aligned}$$

Referencia en 13 bit = 4096 (valor correspondiente a 900 rpm en la escala en 13 bits).

Este parámetro también acepta valores negativos para permitir inversión del sentido de giro del eje del motor. El sentido de rotación de la consigna, sin embargo, depende también del valor del bit 2 de la palabra de control – P0682:

- Bit 2 = 0 y P0683 > 0: consigna para el sentido horario
- Bit 2 = 0 y P0683 < 0: consigna para el sentido antihorario
- Bit 2 = 1 y P0683 > 0: consigna para el sentido antihorario
- Bit 2 = 1 y P0683 < 0: consigna para el sentido horario



## P0695 – Valor para las Salidas Digitales

**Rango de Valores:** 0000h – FFFFh

**Padrão:** 0000h

**Propiedades:** Net

**Grupos de Acceso vía HMI:**

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 111 Estados/Comandos

### Descripción:

Posibilita el control de las salidas digitales a través de las interfaces de red (Serial, USB, 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 necesario que su función sea programada para "Contenido P0695", en los parámetros P0275 a P0280.

Bits	15 a 5	4	3	2	1	0
Función	Reservado	Valor para DO5	Valor para DO4	Valor para DO3 (RL3)	Valor para DO2 (RL2)	Valor para DO1 (RL1)

Bits	Valores
Bit 0 Valor para DO1 (RL1)	0: salida DO1 abierta. 1: salida DO1 cerrada.
Bit 1 Valor para DO2 (RL2)	0: salida DO2 abierta. 1: salida DO2 cerrada.
Bit 2 Valor para DO3 (RL3)	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

*Tabla 3.4 – Funciones de los bits para el parámetro P0695*

## P0696 – Valor 1 para Salidas Analógicas

## P0697 – Valor 2 para Salidas Analógicas

## P0698 – Valor 3 para Salidas Analógicas

## P0699 – Valor 4 para Salidas Analógicas

**Rango de Valores:** -32769...32767

**Padrón:** 0

**Propiedades:** Net

**Grupos de Acceso vía HMI:**

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 111 Estados/Comandos

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

## Descripción:

El valor escrito en estos parámetros es utilizado como valor para la salida analógica, desde que la función de salida analógica deseada sea programada para "Contenido P0696/7/8/9", en los parámetros P0251, P0254, P0257 o P0260.

El valor debe ser escrito en una escala de 15 bits (7FFFh = 32767)<sup>4</sup> 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 P0697/8/9. Por ejemplo, se desea hacer el control del valor de la salida analógica 1 a través de la interface CAN. En este caso se debe proceder la siguiente programación:

- Elegir un de los parámetros P0696 a P0699 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 de la interface CAN, escribir en el P0696 el valor deseado para la salida analógica 1, entre 0 y 100 %, de acuerdo con la escala del parámetro.



### ¡NOTA!

Caso la salida analógica sea programada para operar de -10V hasta 10V, valores negativos para estos parámetros deben ser utilizados para comandar la salida con valores negativos de tensión, o sea, -32769 hasta 32767 representa una variación de -10V hasta 10V en la salida analógica.

## P0700 – Protocolo CAN

**Rango de Valores:** 1 = CANopen  
2 = DeviceNet

**Padrón:** 2

**Propiedades:** CFG y CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 112 CANopen/DeviceNet

### Descripción:

Permite seleccionar el protocolo deseado para la interface CAN. Para habilitar el protocolo CANopen, es necesario programar este parámetro con la opción '1 = CANopen'.

Caso este parámetro sea modificado, él solamente es válido luego del convertidor ser apagado y encendido nuevamente.

## P0701 – Dirección CAN

**Rango de Valores:** 0 a 127

**Padrón:** 63

**Propiedades:** CFG y CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 112 CANopen/DeviceNet

<sup>4</sup> Para la resolución real de la salida, consulte el manual del CFW-11.

### Descripción:

Permite programar la dirección utilizada para comunicación CAN del convertidor. 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 sea modificado, este solamente es válido luego que el convertidor es apagado y encendido nuevamente.

### P0702 – Tasa de Comunicación CAN

<b>Rango de Valores:</b>	1 = 800 Kbit/s 2 = 500 Kbit/s 3 = 250 kbit/s 4 = 125 Kbit/s 5 = 100 kbit/s 6 = 50 Kbit/s 7 = 20 Kbit/s 8 = 10 Kbit/s	<b>Padrón</b>
--------------------------	---	---------------

**Propiedades:** CFG y CAN

#### Grupos de Acceso vía HMI:

- 01 GRUPOS PARÁMETROS
  - L 49 Comunicación
  - L 112 CANOpen/DeviceNet

### Descripción:

Permite programar el valor deseado para la tasa de comunicación de la interface CAN, en bits por segundo. Esta tasa debe ser la misma para todos los equipamientos conectados en la red.

### P0703 – Reset de *Bus Off*

<b>Rango de Valores:</b>	0 = Manual 1 = Automático	<b>Padrón:</b> 1
--------------------------	------------------------------	------------------

**Propiedades:** CFG y CAN

#### Grupos de Acceso vía HMI:

- 01 GRUPOS PARÁMETROS
  - L 49 Comunicación
  - L 112 CANOpen/DeviceNet

### Descripción:

Permite programar cual es el comportamiento del convertidor al detectar un error de *bus off* en la interface CAN:

Opciones	Descripción
0 = Reset Manual	Caso ocurra <i>bus off</i> , será indicado en la HMI la alarma A133, la acción programada en el parámetro P0313 será ejecutada y la comunicación será deshabilitada. Para que el convertidor vuelva a comunicarse a través de la interface CAN, será necesario apagar y encender 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 indicación de la alarma en la HMI y el convertidor no ejecutará la acción descrita en el P0313.

Tabla 3.5 – Valores para el parámetro P0703

## P0705 – Estado del Controlador CAN

**Rango de Valores:** 0 = Deshabilitado  
1 = Reservado  
2 = Interface CAN activa  
3 = *Warning*  
4 = *Error Passive*  
5 = *Bus Off*  
6 = Sin alimentación

**Padrón:** -

**Propiedades:** RO

**Grupos de Acceso vía HMI:**

01 GRUPOS PARÁMETROS  
L 49 Comunicación  
L 112 CANopen/DeviceNet

**Descripción:**

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

Opciones	Descripción
0 = Inactivo	Interface CAN inactiva. Ocurre cuando el convertidor no posee tarjeta de interface CAN instalada.
1 = Reservado	
2 = Interface CAN activa	Interface CAN activa y sin errores.
3 = <i>Warning</i>	Controlador CAN alcanzó el estado de <i>warning</i> <sup>5</sup> .
4 = <i>Error Passive</i>	Controlador CAN alcanzó el estado de <i>error passive</i> <sup>4</sup> .
5 = <i>Bus Off</i>	Controlador CAN alcanzó el estado de <i>bus off</i> .
6 = Sin alimentación	Interface CAN no posee alimentación entre los terminales 1 y 5 del conector.

Tabla 3.6 – Valores do parámetro P0705

## P0706 – Contador de Telegramas CAN Recibidos

**Rango de Valores:** 0 a 65535

**Padrón:** -

**Propiedades:** RO y CAN

**Grupos de Acceso vía HMI:**

01 GRUPOS PARÁMETROS  
L 49 Comunicación  
L 112 CANopen/DeviceNet

**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 convertidor fuera apagado, hecho el reset o al alcanzar el límite máximo del parámetro.

<sup>5</sup> Consultar descripción en el ítem 1.1.4.

## P0707 – Contador de Telegramas CAN Transmitidos

**Rango de Valores:** 0 a 65535

**Padrón:** -

**Propiedades:** RO y CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 112 CANopen/DeviceNet

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 convertidor fuera apagado, 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 y CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 112 CANopen/DeviceNet

### Descripción:

Contador cíclico que indica el número veces que el convertidor entro en estado de *bus off* en la red CAN. Este contador es puesto a cero siempre que el convertidor fuera apagado, 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 y CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS

L 49 Comunicación

L 112 CANopen/DeviceNet

### Descripción:

Contador cíclico que indica el número de mensajes recibidas por la interface CAN, más que no podrán ser procesadas por el convertidor. 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 convertidor fuera apagado, hecho el reset o al alcanzar el límite máximo del parámetro.

## P0721 – Estado de la Comunicación CANopen

<b>Rango de Velocidad:</b>	0 = Deshabilitado 1 = Reservado 2 = Comunicación Habilitada 3 = Control de Errores Habilitado 4 = Error de <i>Guarding</i> 5 = Error de <i>Heartbeat</i>	<b>Padrón:</b> -
----------------------------	---	------------------

**Propiedades:** RO y CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS
L 49 Comunicación
L 112 CANopen/DeviceNet

#### Descripción:

Indica el estado de la tarjeta con relación a la red CANopen, informando si el protocolo fue habilitado y si el servicio de control de errores está activo (*Node Guarding* o *Heartbeat*).

## P0722 – Estado del Nudo CANopen

<b>Rango de Valores:</b>	0 = Deshabilitado 1 = Inicialización 2 = Parado 3 = Operacional 4 = Preoperacional	<b>Padrón:</b> -
--------------------------	--	------------------

**Propiedades:** RO y CAN

### Grupos de Acceso vía HMI:

01 GRUPOS PARÁMETROS
L 49 Comunicación
L 112 CANopen/DeviceNet

#### Descripción:

El convertidor CFW-11 opera como esclavo de la red CANopen, y como tal posee una máquina de estados que hace el control de su comportamiento con relación a la comunicación. Este parámetro indica en cual estado se encuentra el dispositivo.

## 4 DICCIONARIO DE OBJETOS

El diccionario de objetos es un listado con los diversos datos del equipamiento que son accedidos a través de la red CANopen. Un objeto de este listado es identificado a través de un índice de 16 bits, y es basado en este listado que todo el intercambio de datos entre los dispositivos es efectuado.

El documento CiA DS 301 define un conjunto mínimo de objetos que todo el esclavo de la red CANopen debe poseer. Los objetos disponibles en este listado son agrupados de acuerdo con el tipo de función que ellos ejecuta. Los objetos son dispuestos en el diccionario de la siguiente manera:

Índice	Objetos	Descripción
0001h – 0360h	Definición de los tipos de datos	Utilizado como referencia para los tipos de datos soportados por el sistema.
1000h – 1FFFh	Objetos de comunicación	Son objetos comunes a todos los dispositivos CANopen. Contiene informaciones generales a respecto del equipo y también datos para la configuración de la comunicación.
2000h – 5FFFh	Objetos específicos del fabricante	En este rango, cada fabricante de equipos CANopen es libre para definir cuales datos estos objetos irán representar.
6000h – 9FFFh	Objetos estandarizados para dispositivos	Este rango es reservado para objetos que describen el comportamiento de equipos similares, independiente del fabricante.

Tabla 4.1 – Agrupamientos del diccionario de objetos.

Demás índices no referenciados en este listado son reservados para uso futuro.

### 4.1 ESTRUCTURA DEL DICCIONARIO

La estructura general del diccionario de objetos posee el siguiente formato:

Índice	Objeto	Nombre	Tipo	Acceso
--------	--------	--------	------	--------

- ☑ **Índice:** indica directamente el índice del objeto en el diccionario.
- ☑ **Objeto:** describe que información el índice almacena (variables simples, *array*, *record*, etc.)
- ☑ **Nombre:** contiene el nombre del objeto para facilitar su identificación.
- ☑ **Tipo:** indica directamente el tipo de dato almacenado. Para variables simples, este tipo puede ser un entero, un *float*, etc. Para *arrays*, indica el tipo del dato contenido en el *array*. Para *records*, indica el formato del *record*, de acuerdo con los tipos descriptos en la primera parte del diccionario de objetos (índices 0001h – 0360h).
- ☑ **Acceso:** informa si el objeto en cuestión está accesible solamente para lectura (ro), para lectura y escrita (rw), o es una constante (const).

Para objetos del tipo *array* o *records*, todavía es necesario un subíndice, que no es descrito en la estructura del diccionario.

### 4.2 TIPOS DE DATOS

La primera parte del diccionario de objetos (índices 0001h – 0360h) describe los tipos de datos que pueden ser accedidos en un dispositivo en la red CANopen. Estos pueden ser tipos básicos, como enteros y *floats*, o tipos compuestos, formados por un conjunto de entradas, como *records* y *arrays*.

#### 4.2.1 Tipos Básicos

Los siguientes tipos básicos de datos son soportados:

- ☑ **Enteros con señal:** existen tres tipos de enteros con señal soportados por el CFW-11, INTEGER8, INTEGER16 e INTEGER32, que representan, respectivamente, enteros con 8, 16 y 32 bits de datos. Enteros con señal son calculados utilizando complemento de dos, y durante la transmisión, siempre el byte menos significativo es transmitido por primero en un telegrama CAN.
- ☑ **Enteros sin señal:** existen tres tipos de enteros sin señal soportados por el CFW-11, UNSIGNED8, UNSIGNED16 y UNSIGNED32, que representan, respectivamente, enteros con 8, 16 y 32 bits de datos. También durante la transmisión, siempre el byte menos significativo es transmitido por primero.

## 4.2.2 Tipos Compuestos

Es posible formar nuevos tipos de datos a través del agrupamiento de tipos básicos en listas (*arrays* – formados por un único tipo de dato) y estructuras (*records* – formado por diversos tipos de datos). En este caso, cada ítem de este tipo es identificado a través de un subíndice. Los tipos compuestos utilizados por el convertidor son listados abajo.

PDO COMM PARAMETER: este *record* define las informaciones necesarias para configurar un PDO para la comunicación CANopen. El contenido y forma como cada campo es utilizado son detallados en el ítem 5.3.

Subíndice	Descripción de la entrada	Tipo
00h	Número de entradas soportadas en este <i>record</i>	UNSIGNED8
01h	COB-ID	UNSIGNED32
02h	<i>Transmission type</i>	UNSIGNED8
03h	<i>Inhibit time</i>	UNSIGNED16
04h	Reservado	UNSIGNED8
05h	<i>Event timer</i>	UNSIGNED16

**Tabla 4.2** – *Record para configuración de los PDOs*

PDO MAPPING: este *record* define como planificar los datos que serán transmitidos por un PDO durante la comunicación CANopen. El contenido y forma como cada campo es utilizado son detallados en el ítem 5.3.

Subíndice	Descripción de la entrada	Tipo
00h	Número de objetos planificado en el PDO	UNSIGNED8
01h	1o objeto planificado	UNSIGNED32
02h	2o objeto planificado	UNSIGNED32
	...	
	...	
	...	
40h	64o objeto planificado	UNSIGNED32

**Tabla 4.3** – *Record para planificado de los datos de un PDO*

SDO PARAMETER: este *record* define las informaciones necesarias para configurar un SDO para la comunicación CANopen. El contenido y la forma como cada campo es utilizado son detallados en el ítem 5.2.

Subíndice	Descripción de la entrada	Tipo
00h	Número de entradas soportadas en este record	UNSIGNED8
01h	COB-ID cliente → servidor	UNSIGNED32
02h	COB-ID servidor → cliente	UNSIGNED32
03h	Node-ID del cliente/servidor	UNSIGNED8

**Tabla 4.4** – *Record para configuración de los SDOs.*

IDENTITY: este *record* es utilizado para describir el tipo de dispositivo presente en la red.

Subíndice	Descripción de la entrada	Tipo
00h	Número de entradas soportadas en este <i>record</i>	UNSIGNED8
01h	<i>Vendor-ID</i>	UNSIGNED32
02h	<i>Product Code</i>	UNSIGNED32
03h	<i>Revision Number</i>	UNSIGNED32
04h	<i>Serial Number</i>	UNSIGNED32

**Tabla 4.5** – *Record para identificación del dispositivo.*

## 4.2.3 Tipos Extendidos

○ CFW-11 no posee tipos extendidos.



### 4.3 COMMUNICATION PROFILE – OBJETOS PARA COMUNICACIÓN

Los índices de 1000h hasta 1FFFh corresponden, en el diccionario de objetos, la parte responsable por las configuraciones de la comunicación en la red CANopen. Estos objetos son comunes a todos los dispositivos, sin embargo solamente algunos son obligatorios. A seguir es presentado uno listado con los objetos de este rango soportados por el convertidor.

Índice	Objeto	Nombre	Tipo	Acceso
1000h	VAR	device type	UNSIGNED32	ro
1001h	VAR	error register	UNSIGNED8	ro
1005h	VAR	COB-ID SYNC	UNSIGNED32	rw
100Ch	VAR	guard time	UNSIGNED16	rw
100Dh	VAR	life time factor	UNSIGNED8	rw
1014h	VAR	COB-ID EMCY	UNSIGNED32	ro
1016h	ARRAY	Consumer heartbeat time	UNSIGNED32	rw
1017h	VAR	Producer heartbeat time	UNSIGNED16	rw
1018h	RECORD	Identity Object	Identity	ro
Server SDO Parameter				
1200h	RECORD	1st Server SDO parameter	SDO Parameter	ro
Receive PDO Communication Parameter				
1400h	RECORD	1st receive PDO Parameter	PDO CommPar	rw
1401h	RECORD	2nd receive PDO Parameter	PDO CommPar	rw
1402h	RECORD	3rd receive PDO Parameter	PDO CommPar	rw
1403h	RECORD	4th receive PDO Parameter	PDO CommPar	rw
Receive PDO Mapping Parameter				
1600h	RECORD	1st receive PDO mapping	PDO Mapping	rw
1601h	RECORD	2nd receive PDO mapping	PDO Mapping	rw
1602h	RECORD	3rd receive PDO mapping	PDO Mapping	rw
1603h	RECORD	4th receive PDO mapping	PDO Mapping	rw
Transmit PDO Communication Parameter				
1800h	RECORD	1st transmit PDO Parameter	PDO CommPar	rw
1801h	RECORD	2nd transmit PDO Parameter	PDO CommPar	rw
1802h	RECORD	3rd transmit PDO Parameter	PDO CommPar	rw
1803h	RECORD	4th transmit PDO Parameter	PDO CommPar	rw
Transmit PDO Mapping Parameter				
1A00h	RECORD	1st transmit PDO mapping	PDO Mapping	rw
1A01h	RECORD	2nd transmit PDO mapping	PDO Mapping	rw
1A02h	RECORD	3rd transmit PDO mapping	PDO Mapping	rw
1A03h	RECORD	4th transmit PDO mapping	PDO Mapping	rw

*Tabla 4.6 – Listado de objetos del CFW-11 – Communication Profile*

Demás objetos no presentados en este listado no son utilizados por el CFW-11, o entonces están en rangos reservados del diccionario.

### 4.4 MANUFACTURER SPECIFIC – OBJETOS ESPECÍFICOS DEL CFW-11

En los índices de 2000h hasta 5FFFh, cada fabricante es libre para definir cuales objetos estarán presentes, el tipo y la función de cada objeto. En el caso del CFW-11, en este rango de objetos fue proporcionado todo el listado de parámetros. A través de estos parámetros es posible operar el CFW-11, ejecutando cualquier función que el convertidor pueda realizar. Los parámetros fueran dispuestos a partir del índice 2000h, y con el número del parámetro sumado a este índice para obtener su posición en el diccionario. La tabla que sigue presenta como están distribuidos los parámetros en el diccionario de objetos.

Índice	Objeto	Nombre	Tipo	Acceso
2000h	VAR	P0000 – Parámetro de acceso	INTEGER16	rw
2001h	VAR	P0001 – Referencia de Velocidad	INTEGER16	ro
2002h	VAR	P0002 – Velocidad del motor	INTEGER16	ro
2003h	VAR	P0003 – Corriente del motor	INTEGER16	ro
2004h	VAR	P0004 – Tensión CC	INTEGER16	ro
...	...	...	...	...
2064h	VAR	P0100 – Tiempo de aceleración	INTEGER16	rw
2065h	VAR	P0101 – Tiempo de desaceleración	INTEGER16	rw
...	...	...	...	...

*Tabla 4.7 – Listado de objetos del CFW-11 – Manufacturer Specific*

Para el listado completo y una descripción detallada de los parámetros, consulte el manual del CFW-11. Es necesario reconocer la operación del convertidor a través de los parámetros para que se pueda programar correctamente su operación vía red CANopen.

#### **4.5 DEVICE PROFILE – OBJETOS COMUNES PARA DRIVES**

La documentación CANopen también incluyó propuestas para la estandarización de determinados tipos de dispositivos. En el caso del CFW-11, él sigue el descrito por la *CiA DPS 402 – Device Profile Drives and Motion Control*. Este documento describe un conjunto de objetos que deben ser comunes para drives, independiente del fabricante. Eso facilita la interoperabilidad entre dispositivos con la misma función (como convertidores de frecuencia), pues tanto los datos cuanto el comportamiento del dispositivo están dispuestos de una forma estándar.

Para estos objetos fueran reservados los índices de 6000h hasta 9FFFh. Es posible operar el convertidor a través de la red CANopen, tanto a través de los parámetros (ubicados a partir del índice 2000h) cuanto a través de estos objetos estandarizados. Para una descripción detallada de cuales objetos están disponibles en este rango del diccionario de objetos, consulte la sección 6.

## 5 DESCRIPCIÓN DE LOS OBJETOS DE COMUNICACIÓN

En esta sección se describe detalladamente cada un de los objetos mencionados en la Tabla 4.6, además de mencionar también el funcionamiento de los objetos de comunicación (COBs) referenciados en el ítem 1.7. Es necesario conocer como estos objetos son operados para utilizar las funciones disponibles para la comunicación del CFW-11.

### 5.1 OBJETOS DE IDENTIFICACIÓN

Existe un conjunto de objetos en el diccionario que son utilizados para la identificación del equipamiento, sin embargo no poseen influencia en su comportamiento en la red CANopen.

#### 5.1.1 Objeto 1000h – *Device Type*

Este objeto suministra un código en 32 bits que describe el tipo de objeto y su funcionalidad.

Índice	1000h
Nombre	Device type
Objeto	VAR
Tipo	UNSIGNED32
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	0001.0192h

Este código puede ser dividido en dos partes: 16 bits inferiores, describiendo el tipo de perfil (*profile*) que el dispositivo utiliza, y 16 bits superiores, indicando una función específica, de acuerdo con el perfil especificado. Para el CFW-11, estos valores son respectivamente 0192h (sigue el especificado por el documento *CiA 402 - Device Profile Drives and Motion Control*), y 0001h (posee funcionalidades de convertidor de frecuencia).

#### 5.1.2 Objeto 1001h – *Error Register*

Este objeto indica la ocurrencia o no de error en el dispositivo. El tipo de error registrado para el CFW-11 es descrito conforme la Tabla 5.1.

Índice	1001h
Nombre	Error register
Objeto	VAR
Tipo	UNSIGNED8
Acceso	ro
Mapeable	Sí
Rango	UNSIGNED8
Valor Padrón	0

Bit	Significado
0	Error genérico
1	Corriente
2	Tensión
3	Temperatura
4	Comunicación
5	Reservado (siempre 0)
6	Reservado (siempre 0)
7	Específico del fabricante

*Tabla 5.1 – Estructura del objeto Error Register*

Caso el dispositivo presente algún error, el bit equivalente debe ser activado. El primer bit (error genérico) deberá ser activado en cualquier situación de error.

### 5.1.3 Objeto 1018h – *Identity Object*

Trae informaciones generales a respecto del dispositivo.

Índice	1018h
Nombre	Identity object
Objeto	Record
Tipo	Identity

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	4

Subíndice	1
Descripción	Vendor ID
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	0000.0123h

Subíndice	2
Descripción	Código del producto
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	0000.0400h

Subíndice	3
Descripción	Número de la revisión
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	De acuerdo con la versión de firmware del equipamiento

Subíndice	4
Descripción	Número serial
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	Diferente para cada CFW-11

El Vendor ID es un número que identifica el fabricante junto a la CiA. El código del producto es definido por el fabricante de acuerdo con el tipo de producto. El número de la revisión representa la versión de firmware del equipamiento. El subíndice 4 es un número serial único para cada convertidor de frecuencia CFW-11 en red CANopen.

## 5.2 SERVICE DATA OBJECTS – SDOS

Los SDOS son responsables por el acceso directo al diccionario de objetos de un determinado dispositivo en la red. Ellos son utilizados para la configuración y, por lo tanto, poseen baja prioridad, ya que no deben ser utilizados para comunicar datos necesarios para la operación del dispositivo.

Existen dos tipos de SDOS: cliente y servidor. Básicamente, la comunicación inicia con el cliente (usualmente el maestro de la red) haciendo una requisición de lectura (*upload*) o escrita (*download*) para un servidor, y este contesta al que fue solicitado.

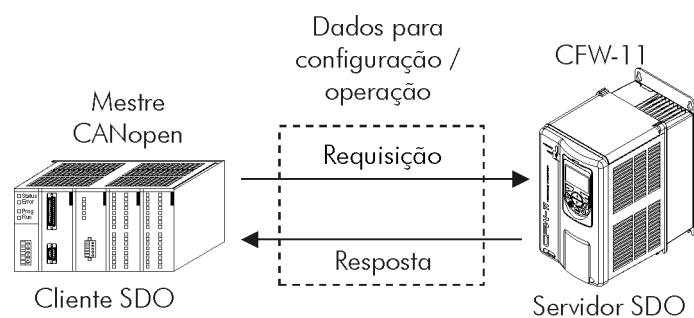


Figura 5.1 – Comunicação entre Cliente y Servidor SDO

## 5.2.1 Objeto 1200h – Servidor SDO

El CFW-11 posee un único SDO del tipo servidor, que posibilita el acceso a todo el su diccionario de objetos. A través de el, un cliente SDO puede configurar la comunicación, parámetros y modos de operación del convertidor. Todo el servidor SDO posee un objeto, del tipo SDO\_PARAMETER (consultar ítem 4.2.2), para la su configuración, poseyendo la siguiente estructura:

Índice	1200h
Nombre	Server SDO Parameter
Objeto	Record
Tipo	SDO Parameter

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	2

Subíndice	1
Descripción	COB-ID Cliente - Servidor (rx)
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	600h + Node-ID

Subíndice	2
Descripción	COB-ID Servidor - Cliente (tx)
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	580h + Node-ID

## 5.2.2 Funcionamiento de los SDOs

Un telegrama enviado por un SDO posee 8 bytes de tamaño, con la siguiente estructura:

Identificador	8 bytes de datos							
	Comando	Índice		Subíndice	Datos del objeto			
11 bits	byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7

El identificador depende del sentido de la transmisión (rx o tx) y de la dirección (o Node-ID) del servidor destino. Por ejemplo, un cliente que hace una requisición para un servidor cuyo Node-ID es 1, debe enviar un mensaje con el identificador igual a 601h. El servidor irá recibir esto mensaje y contestar con un telegrama cuyo COB-ID es igual a 581h.

El código del comando depende del tipo de función utilizada. Para las transmisiones de un cliente para un servidor, pueden ser utilizados los siguientes comandos:

Comando	Función	Descripción	Datos del objeto
22h	<i>Download</i>	Escrita en objeto	Indefinido
23h	<i>Download</i>	Escrita en objeto	4 bytes
2Bh	<i>Download</i>	Escrita en objeto	2 bytes
2Fh	<i>Download</i>	Escrita en objeto	1 byte
40h	<i>Upload</i>	Lectura de objeto	No utilizado
60h o 70h	<i>Upload segment</i>	Lectura segmentada	No utilizado

**Tabla 5.2** – Código de los comandos para cliente SDO

Al hacer la requisición, el cliente indicará a través de su COB-ID, cual es la dirección del esclavo para el cual esta requisición se destina. Solamente un esclavo (usando su respectivo servidor SDO) podrá contestar para el cliente el telegrama recibido. El telegrama de respuesta poseerá también la misma estructura del telegrama de requisición, sin embargo los comandos serán diferentes:

Comando	Función	Descripción	Datos del objeto
60h	<i>Download</i>	Respuesta para escrita en objeto	No utilizado
43h	<i>Upload</i>	Respuesta para lectura de objeto	4 bytes
4Bh	<i>Upload</i>	Respuesta para lectura de objeto	2 bytes
4Fh	<i>Upload</i>	Respuesta para lectura de objeto	1 byte
41h	<i>Upload segment</i>	Inicia respuesta segmentada para lectura	4 bytes
01h ... 0Dh	<i>Upload segment</i>	Último segmento de datos para lectura	8 ... 2 bytes

**Tabla 5.3** – Código de los comandos para servidor SDO.

Para lecturas que involucran hasta cuatro bytes de datos, un único mensaje puede ser transmitido por el servidor; para lectura de una cantidad mayor de bytes, es necesario que cliente y servidor intercambien múltiples telegramas.

Un telegrama solamente es completo luego de la confirmación del servidor para la requisición hecha por el cliente. Caso algún error sea detectado durante el intercambio de telegramas (por ejemplo, no hay respuesta del servidor), el cliente podrá abortar el proceso con un mensaje de aviso con el código del comando igual a 80h.



### ¡NOTA!

Cuando el SDO es utilizado para escrita en los objetos que representan los parámetros del CFW-11 (objetos a partir del índice 2000h), este valor es guardado en la memoria no volátil del convertidor de frecuencia. De esta forma, después de apagado o hecho el reset del equipamiento, los valores configurados no son perdidos. Para los demás objetos, estos valores no son guardados automáticamente, de manera que es necesario rescribir los valores deseados.

Ejemplo: un cliente SDO solicita para un CFW-11 en la dirección 1, la lectura del objeto identificado por el índice 2000h, subíndice 0 (cero), que representa un entero de 16 bits. El telegrama del maestro posee la siguiente forma:

Identificador	Comando	Índice		Subíndice		Datos			
601h	40h	00h	20h	00h	00h	00h	00h	00h	00h

El CFW-11 contesta a la requisición, indicando que el valor para el referido objeto es igual a 999<sup>6</sup>:

Identificador	Comando	Índice		Subíndice		Datos			
581h	4Bh	00h	20h	00h	E7	03h	00h	00h	00h

## 5.3 PROCESS DATA OBJECTS – PDOS

Los PDOS son utilizados para enviar y recibir datos utilizados durante la operación del dispositivo, que muchas veces precisan ser transmitidos de forma rápida y eficiente. Por eso, ellos poseen una prioridad mayor del que los SDOs.

En los PDOS, solamente los datos son transmitidos en el telegrama (índices y subíndices son omitidos), y de esta forma es posible hacer una transmisión más eficiente, con mayor volumen de datos en un único telegrama. Sin embargo es necesario configurar previamente el que está siendo transmitido por el PDO, de forma que, mismo sin la indicación del índice y subíndice, sea posible saber el contenido del telegrama.

<sup>6</sup> No olvidar que cualquier dato del tipo entero, el orden de transferencia de los bytes va del menos significativo hasta el más significativo.

Existen dos tipos de PDOs, los PDOs de recepción y los PDOs de transmisión. Los PDOs de transmisión son responsables por enviar datos para la red, mientras que los PDOs de recepción se quedan responsables por recibir y tratar estos datos. De esta forma es posible que haya comunicación entre esclavos de la red CANopen, desde que sea configurado un esclavo para transmitir una información, y un o más esclavos para recibir esta información.

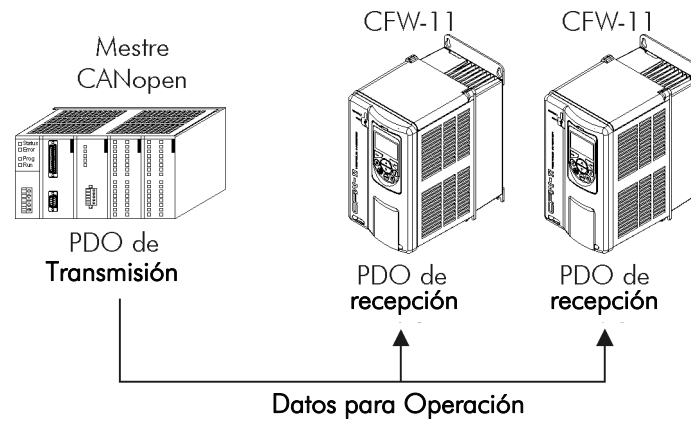


Figura 5.2 – Comunicación utilizando PDOs



**¡NOTA!**

PDOs solamente pueden ser transmitidos o recibidos cuando el dispositivo está en el estado operacional. La

Figura 5.5 presenta los estados disponibles para un nudo de la red CANopen.

**5.3.1 Objetos mapeables para los PDOs**

Para un objeto poder ser transmitido a través de un PDO, es necesario que él sea mapeable para el contenido del PDO. En la descripción de los objetos de comunicación (1000h – 1FFFh), el campo “Mapeable” informa esta condición. Usualmente, solo informaciones necesarias para la operación del dispositivo son mapeables, como comandos para habilitación, status del dispositivo, referencias, etc. Informaciones para configuración del dispositivo no son accedidas a través de PDOs, e caso sea necesario accederlas vía red se debe utilizar los SDOs.

Para los objetos específicos del CFW-11 (2000h – 5FFFh), la tabla que sigue presenta algunos objetos mapeables para los PDOs. Parámetros con acceso solo para lectura (ro) pueden ser utilizados solo por PDOs de transmisión, mientras que los demás parámetros pueden ser utilizados solo por PDOs de recepción. El archivo EDS del CFW-11 trae un listado de todos los objetos disponibles para el convertidor, informando si el objeto es mapeable o no.

Índice	Objeto	Nombre	Tipo	Acceso
2002h	VAR	P0002 – Velocidad del Motor	UNSIGNED16	ro
2003h	VAR	P0003 – Corriente del Motor	UNSIGNED16	ro
2004h	VAR	P0004 – Tensión Bus CC (Ud)	UNSIGNED16	ro
2005h	VAR	P0005 – Frecuencia del Motor	UNSIGNED16	ro
2006h	VAR	P0006 – Estado del Convertidor	UNSIGNED16	ro
2007h	VAR	P0007 – Tensión de Salida	UNSIGNED16	ro
2009h	VAR	P0009 – Par en el Motor	INTEGER16	ro
200Ah	VAR	P0010 – Potencia de Salida	UNSIGNED16	ro
200Ch	VAR	P0012 – Estado DI1 a DI8	UNSIGNED16	ro
200Dh	VAR	P0013 – Estado DO1 a RL3	UNSIGNED16	ro
2012h	VAR	P0018 – Valor de AI1	INTEGER16	ro
2013h	VAR	P0019 – Valor de AI2	INTEGER16	ro
2014h	VAR	P0020 – Valor de AI3	INTEGER16	ro
2015h	VAR	P0021 – Valor de AI4	INTEGER16	ro
2064h	VAR	P0100 – Tiempo Aceleración	UNSIGNED16	rw
2065h	VAR	P0101 – Tiempo Desaceleración	UNSIGNED16	rw
22A8h	VAR	P0680 – Estado Lógico	UNSIGNED16	ro
22A9h	VAR	P0681 – Velocidad 13 bits	UNSIGNED16	ro
22ACh	VAR	P0684 – Control CANopen/DNet	UNSIGNED16	rw
22ADh	VAR	P0685 – Ref. Vel. CANop./DNet	UNSIGNED16	rw

Tabla 5.4 – Ejemplos de parámetros mapeables para PDOs

### 5.3.2 PDOs de recepción

Los PDOs de recepción, o RPDOs, son responsables por recibir datos que otros dispositivos envían para la red CANopen. El CFW-11 posee 4 PDOs de recepción, cada uno pudiendo recibir hasta 8 bytes de datos. Cada RPDO posee dos parámetros para su configuración, un PDO\_COMM\_PARAMETER y un PDO\_MAPPING, conforme presentado a seguir.

#### PDO\_COMM\_PARAMETER

Índice	1400h
Nombre	Receive PDO communication parameter
Objeto	Record
Tipo	PDO COMM PARAMETER

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	2

Subíndice	1
Descripción	COB-ID usado por el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1400h: 200h + Node-ID 1401h: 300h + Node-ID 1402h: 400h + Node-ID 1403h: 500h + Node-ID

Subíndice	2
Descripción	Tipo de transmisión
Acceso	rw
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	254

El subíndice 1 contiene el COB-ID del PDO de recepción. Siempre que un mensaje es enviado para la red, este objeto irá leer cual es el COB-ID de este mensaje, e caso este sea igual a valor de este campo, el mensaje será recibida por el dispositivo. Este campo es formado por un UNSIGNED32 con la siguiente estructura:

Bit	Valor	Descripción
31 (MSB)	0	PDO está habilitado
	1	PDO está deshabilitado
30	0	RTR permitido
29	0	Tamaño del identificador = 11 bits
28 – 11	0	No utilizado, siempre 0
10 – 0 (LSB)	X	COB-ID de 11 bits

**Tabla 5.5** – Descripción del COB-ID

El bit 31 permite habilitar o deshabilitar el PDO. Los bits 30 y 29, que deben ser mantenidos en 0 (cero), indican respectivamente que el PDO acepta frames remotos (RTR frames) y que utiliza identificador de 11 bits. Como el convertidor de frecuencia CFW-11 no utiliza identificadores de 29 bits, los bits de 28 hasta 11 deben ser mantenidos en 0 (cero), mientras que los bits de 10 hasta 0 (cero) son usados para configurar el COB-ID para el PDO.



El subíndice 2 indica el tipo de transmisión de este objeto, de acuerdo con la tabla que sigue.

Tipo de transmisión	Transmisión de PDOs				
	<i>Cíclico</i>	<i>Acíclico</i>	<i>Sincrónico</i>	<i>Asíncrono</i>	<i>RTR</i>
0		•	•		
1 – 240	•		•		
241 – 251	Reservado				
252			•		•
253				•	•
254				•	
255				•	

*Tabla 5.6 – Descripción del tipo de transmisión*

- ☑ **Valores 0 – 240:** cualquier RPDOs programado en este rango posee el mismo funcionamiento. Al detectar un mensaje, él irá recibir los datos, sin embargo no actualizará los valores recibidos hasta detectar el próximo telegrama SYNC.
- ☑ **Valores 252 y 253:** no permitido para PDOs de recepción.
- ☑ **Valores 254 y 255:** indica que no posee relación con el objeto de sincronización. Al recibir unos mensajes, sus valores serán actualizados inmediatamente.

## **PDO\_MAPPING**

Índice	1600h
Nombre	Receive PDO mapping
Objeto	Record
Tipo	PDO MAPPING

Subíndice	0
Descripción	Número de objetos mapeados
Acceso	ro
Mapeable	No
Rango	0 = deshabilitado 1 ... 4 = número de objetos mapeados
Valor Padrón	1600h: 1 1601h: 2 1602h: 2 1603h: 2

Subíndice	1
Descripción	1º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1600h: 6040.0010h 1601h: 6040.0010h 1602h: 6040.0010h 1603h: 22AC.0010h

Subíndice	2
Descripción	2º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1600h: 0 1601h: 6060.0008h 1602h: 6042.0010h 1603h: 22AD.0010h

Subíndice	3
Descripción	3º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1600h: 0 1601h: 0 1602h: 0 1603h: 0

Subíndice	4
Descripción	4º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1600h: 0 1601h: 0 1602h: 0 1603h: 0

Este parámetro indica los objetos mapeados en los PDOs de recepción del CFW-11. Para cada RPDO, es posible mapear hasta cuatro objetos diferentes, desde que el tamaño total no ultrapase ocho bytes. El mapeado de un objeto es hecho indicando su índice, subíndice<sup>7</sup> y tamaño (en bits) en un campo UNSIGNED32, con el siguiente formato:

UNSIGNED32		
Índice (16 bits)	Subíndice (8 bits)	Tamaño del objeto (8 bits)

Por ejemplo, analizando el mapeado padrón del PDO de recepción, tenemos:

- ☑ **Subíndice 0 = 2:** el RPDO posee dos objetos mapeados.
- ☑ **Subíndice 1 = 22AC.0010h:** el primero objeto mapeado posee índice igual a 22ACh, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0684 del convertidor, que representa la palabra de control vía CANopen.
- ☑ **Subíndice 2 = 22AD.0010h:** el segundo objeto mapeado posee índice igual a 22ADh, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0685 del convertidor, que representa la referencia de velocidad.

De esta forma, siempre que este PDO recibir un telegrama, él va saber que el telegrama deberá contener cuatro bytes de datos, con el contenido para la palabra de control y referencia de velocidad para el CFW-11. Es posible modificar este mapeado, modificando la cantidad o el número de los objetos mapeados. Recordar que en el máximo pueden ser mapeados hasta 4 objetos o 8 bytes.



### ¡NOTA!

- ☑ Para poder modificar los objetos mapeados en un PDO, primero es necesario escribir el valor 0 (cero) en el subíndice 0 (cero). De este modo, los valores de los subíndices 1 hasta 4 pueden ser modificados. Después de hecho el mapeado deseado, se debe escribir nuevamente en el subíndice 0 (cero) el número de objetos que fueran mapeados, habilitando nuevamente el PDO.
- ☑ Para agilizar la actualización de los datos vía PDO, los valores recibidos a través de estos objetos no son guardados en la memoria no volátil del convertidor. De este modo, luego de un comando para apagar o de reset del equipamiento, los objetos modificados por un RPDO vuelven para el su valor padrón.
- ☑ No olvidar que los PDOs solamente pueden ser recibidos caso el CFW-11 se encuentre en el estado operacional.

<sup>7</sup> Caso o objeto seja do tipo VAR e não possua sub-índice, deve ser indicado o valor 0 (zero) para o sub-índice.

### 5.3.3 PDOs de transmisión

Los PDOs de transmisión, o TPDOs, como el nombre dice, son responsables por transmitir datos para la red CANopen. El CFW-11 posee 4 PDOs de transmisión, cada uno pudiendo transmitir hasta 8 bytes de datos. De modo semejante a los RPDOs, cada TPDO posee dos parámetros para su configuración, un PDO\_COMM\_PARAMETER y un PDO\_MAPPING, conforme presentado a seguir.

#### PDO\_COMM\_PARAMETER

Índice	1800h
Nombre	Transmit PDO Parameter
Objeto	Record
Tipo	PDO COMM PARAMETER

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	5

Subíndice	1
Descripción	COB-ID usado por el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1800h: 180h + Node-ID 1801h: 280h + Node-ID 1802h: 380h + Node-ID 1803h: 480h + Node-ID

Subíndice	2
Descripción	Tipo de transmisión
Acceso	rw
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	254

Subíndice	3
Descripción	Tiempo entre transmisiones
Acceso	rw
Mapeable	No
Rango	UNSIGNED16
Valor Padrón	-

Subíndice	4
Descripción	Reservado
Acceso	rw
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	-

Subíndice	5
Descripción	Temporizador de eventos
Acceso	rw
Mapeable	No
Rango	0 = deshabilitado UNSIGNED16
Valor Padrón	0

El subíndice 1 contiene el COB-ID del PDO de transmisión. Siempre que este PDO envíe un mensaje para la red, el identificador de este mensaje será este COB-ID. La estructura de este campo es descrita en la Tabla 5.5.

El subíndice 2 indica el tipo de transmisión de este objeto, que sigue descrito por la Tabla 5.6. Sin embargo, su funcionamiento es distinto para PDOs de transmisión:

- Valor 0:** indica que la transmisión debe ocurrir inmediatamente luego de la recepción de un telegrama SYNC, más no periódicamente.
- Valores 1 – 240:** el PDO debe ser transmitido a cada telegrama SYNC detectado (u ocurrencias múltiples de SYNC, de acuerdo con el número elegido entre 1 e 240).
- Valor 252:** indica que el contenido del mensaje debe ser actualizado (más no enviado), luego de la recepción de un telegrama SYNC. El envío del mensaje debe ser hecho luego de la recepción de un frame remoto (RTR frame).
- Valor 253:** el PDO debe actualizar y enviar un mensaje así que recibir un frame remoto.
- Valores 254:** el objeto debe ser transmitido de acuerdo con el timer programado en el subíndice 5.
- Valores 255:** el objeto es transmitido automáticamente cuando el valor de algún de los objetos mapeados en este PDO fuera modificado. Funciona por modificación de estado (*Change Of State*). Este tipo también permite que el PDO sea transmitido de acuerdo con el timer programado en el subíndice 5.

En el subíndice 3 es posible programar un tiempo mínimo (en múltiplos de 100us) que debe transcurrir para que, después de transmitido un telegrama, un nuevo telegrama pueda ser enviado por este PDO. El valor 0 (cero) deshabilita esta función.

El subíndice 5 contiene un valor para habilitar un temporizador para el envío automático de un PDO. De este modo, siempre que un PDO es configurado para el tipo asíncrono, es posible programar el valor de este temporizador (en múltiplos de 1ms), para que el PDO sea transmitido periódicamente en el tiempo programado.



**¡NOTA!**

- Se debe observar el tiempo programado en este temporizador, de acuerdo con la tasa de transmisión utilizada. Tiempos muy pequeños (próximos al tiempo de transmisión del telegrama) pueden monopolizar el bus, causando la retransmisión indefinida del PDO e impidiendo que otros objetos menos prioritarios posan transmitir sus datos.
- El tiempo mínimo permitido para esta función en el CFW-11 es 2ms.
- Es importante observar el tiempo entre transmisiones programado en el subíndice 3 principalmente cuando el PDO es programado con el valor 255 en el subíndice 2 (*Change Of State*).
- No olvidar que los PDOs solamente pueden ser transmitidos caso el esclavo se encuentra en el estado operacional.

**PDO\_MAPPING**

Índice	1A00h
Nombre	Transmit PDO mapping
Objeto	Record
Tipo	PDO MAPPING

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	0 = deshabilitado 1 ... 4 = número de objetos mapeados
Valor Padrón	1A00h: 1 1A01h: 2 1A02h: 2 1A03h: 2

Subíndice	1
Descripción	1º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1A00h: 6041.0010h 1A01h: 6041.0010h 1A02h: 6041.0010h 1A03h: 22A8.0010h

Subíndice	2
Descripción	2º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1A00h: 0 1A01h: 6061.0008h 1A02h: 6044.0010h 1A03h: 22A9.0010h

Subíndice	3
Descripción	3º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1A00h: 0 1A01h: 0 1A02h: 0 1A03h: 0

Subíndice	4
Descripción	4º objeto mapeado en el PDO
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	1A00h: 0 1A01h: 0 1A02h: 0 1A03h: 0

El PDO MAPPING para la transmisión funciona de modo semejante que para la recepción, sin embargo en este caso son definidos los datos que serán transmitidos por el PDO. Cada objeto mapeado debe ser colocado en el listado conforme presentado a seguir:

<i>UNSIGNED32</i>		
Índice (16 bits)	Subíndice (8 bits)	Tamaño del objeto (8 bits)

Por ejemplo, analizando el mapeado padrón del cuarto PDO de transmisión, tenemos:

- Subíndice 0 = 2:** este PDO posee dos objetos mapeados.
- Subíndice 1 = 22A8.0010h:** el primero objeto mapeado posee índice igual a 22A8h, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0680 del convertidor, que representa el estado del convertidor.
- Subíndice 2 = 22A9.0010h:** el segundo objeto mapeado posee índice igual a 22A9h, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0681 del convertidor, que representa la velocidad del motor.

De este modo, siempre que este PDO fuera transmitir sus datos, él va elaborar su telegrama conteniendo cuatro bytes de datos, con los valores de los parámetros P0680 y P0681. Es posible modificar este mapeado, alterando la cantidad o el número de los parámetros mapeados. Recordar que en el máximo pueden ser mapeados 4 objetos o 8 bytes.



### ¡NOTA!

Para poder modificar los objetos mapeados en un PDO, primero es necesario escribir el valor 0 (cero) en el subíndice 0 (cero). De este modo, los valores de los subíndices 1 hasta 4 pueden ser modificados. Después de hecho el mapeado deseado, se debe escribir nuevamente en el subíndice 0 (cero) el número de objetos que fueran mapeados, habilitando nuevamente el PDO.

## 5.4 EMERGENCY OBJECT – EMCY

El objeto de emergencia (EMCY) es utilizado para señalar la ocurrencia de un error en el dispositivo. Siempre que un error ocurra en el CFW-11 (cortocircuito, sobretensión, falla en la comunicación, etc.), este objeto irá a enviar un mensaje de emergencia para la red. Este mensaje podrá ser interpretado por un consumidor EMCY (usualmente el maestro de la red), que podrá tomar una acción de acuerdo con el programado para la aplicación, como hacer el reset de error o deshabilitar los demás dispositivos de la red.

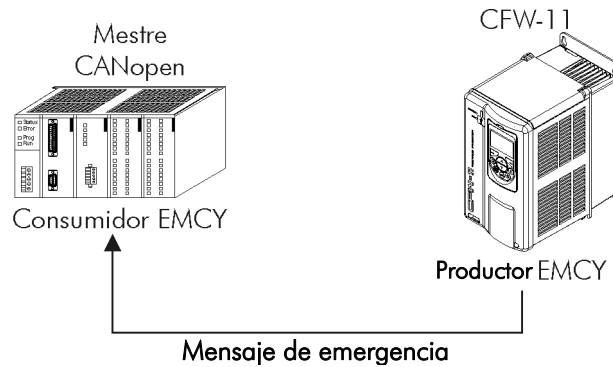


Figura 5.3 - EMCY

Al transmitir un mensaje, tres informaciones son enviadas en los ocho bytes de datos del telegrama: el código de error CiA, el objeto 1001h (*error register*) y el código de la falla del CFW-11. El telegrama posee entonces la siguiente estructura:

byte 0	byte 1	byte 2	byte 3	byte 4	byte 5	byte 6	byte 7
Código del error CiA		Objeto 1001h Error Register	Código del Error CFW-11	Reservado			

En el CFW-11, este objeto está siempre activo y, por lo tanto, siempre reportará para la red la ocurrencia de errores en el convertidor. Existe un único parámetro que posibilita leer cual es el COB-ID del objeto, o sea, cual es el identificador del telegrama para los mensajes de error enviado.

Índice	1014h
Nombre	COB-ID emergency message
Objeto	VAR
Tipo	UNSIGNED32
Acceso	ro
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	80h + Node-ID

## 5.5 SYNCHRONIZATION OBJECT – SYNC

Este objeto es transmitido con el objetivo de permitir la sincronización de eventos entre los dispositivos de la red CANopen. Elle es transmitido por un productor SYNC, y los dispositivos que detectan su transmisión son denominados consumidores SYNC.

El CFW-11 posee la función de consumidor SYNC y, por lo tanto, puede programar sus PDOs para que sean sincrónicos. Como presentado en la Tabla 5.6, PDOs sincrónicos son aquellos relacionados con el objeto de sincronización, y por lo tanto pueden ser programados para que sean transmitidos o actualizados con base en este objeto.

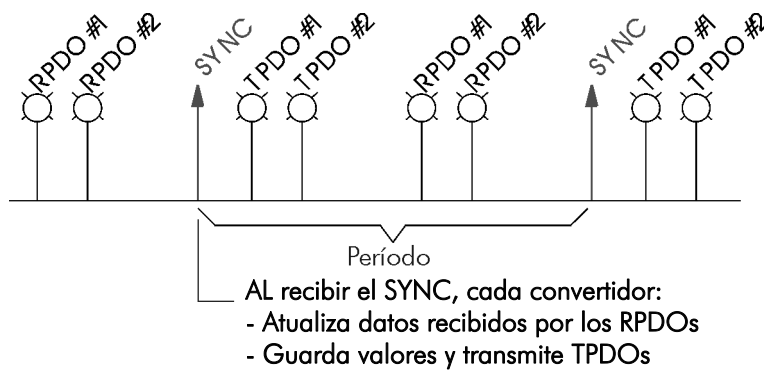


Figura 5.4 – SYNC

El mensaje SYNC transmitido por el productor no posee dato alguno en su campo de datos, pues su objetivo es suministrar una base de tiempo para los demás objetos. En el CFW-11, existe un objeto para la configuración del COB-ID del consumidor SYNC.

Índice	1015h
Nombre	COB-ID SYNC
Objeto	VAR
Tipo	UNSIGNED32

Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	80h



**¡NOTA!**

Se debe observar el tiempo programado en el productor para el período de los telegramas SYNC, de acuerdo con la tasa de transmisión utilizada y el número de PDOs sincrónicos a ser transmitidos. Es necesario que se tenga tiempo suficiente para la transmisión de estos objetos, y también es recomendado que se tenga holgura para posibilitar el envío de mensajes asíncronas, como EMCY, PDOs asíncronos y SDOs.

## 5.6 NETWORK MANAGEMENT – NMT

El objeto de gestión de la red es responsable por un conjunto de servicios que controlan la comunicación del dispositivo en la red CANopen. Para el CFW-11 están disponibles los servicios de control del nudo u de control de errores (utilizando *Node Guarding* o *Heartbeat*).

### 5.6.1 Control de los estados del esclavo

Con relación a la comunicación, un dispositivo de la red CANopen puede ser descrito por la siguiente máquina de estados:

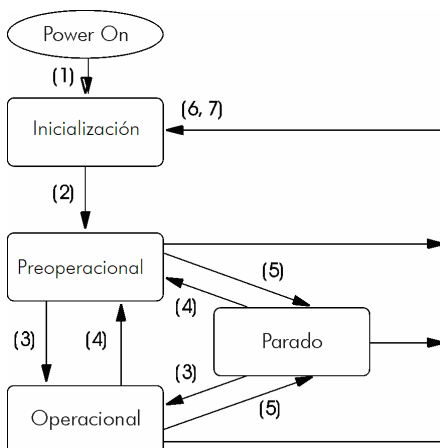


Figura 5.5 – Diagrama de estados del nudo CANopen

Transición	Descripción
1	Dispositivo es encendido y empieza la inicialización (automático).
2	Inicialización concluida, va para el estado preoperacional (automático).
3	Recibe comando Start Node para entrar en el estado operacional.
4	Recibe comando Enter Pre-Operational, y va para el estado preoperacional.
5	Recibe comando Stop Node para entrar en el estado parado.
6	Recibe comando Reset Node, donde ejecuta el reset completo del dispositivo.
7	Recibe comando Reset Communication, donde reinicializa el valor de los objetos y la comunicación CANopen del dispositivo.

*Tabla 5.7 – Descripción de las transiciones*

Durante la inicialización, es definido el Node-ID, creados los objetos y configurada la interface con la red CAN. No es posible se comunicar con el dispositivo en esta etapa, que es concluida automáticamente. En el final de esta etapa, el esclavo envía para la red un telegrama del objeto Boot-up, utilizado solo para indicar que la inicialización fue concluida y que el esclavo entro en el estado preoperacional. Este telegrama posee identificador 700h + Node-ID, y solo un byte de datos con valor igual a 0 (cero).

En el estado preoperacional, ya es posible se comunicar con el esclavo, sin embargo los PDOs todavía no están disponibles para operación. En el estado operacional, todos los objetos están disponibles, mientras que en el estado parado, solo el objeto NMT puede recibir o transmitir telegramas para la red. La tabla que sigue presenta los objetos disponibles para cada estado.

	Inicialización	Preoperacional	Operacional	Parado
PDO			•	
SDO		•	•	
SYNC		•	•	
EMCY		•	•	
Boot-up	•			
NMT		•	•	•

*Tabla 5.8 – Objetos accesibles en cada estado*

Esta máquina de estados es controlada por el maestro de la red, que envía para cada esclavo, comandos para que sea ejecutado la transición de estados deseado. Estos telegramas no poseen confirmación, lo que significa que el esclavo solo recibe el telegrama sin retornar respuesta para el maestro. Los telegramas recibidos poseen la siguiente estructura:

Identificador	byte 1	byte 2
00h	Código del comando	Node-ID destino

Código del comando	Node-ID destino
1 = <i>START node</i> (transición 3)	0 = Todos los esclavos
2 = <i>STOP node</i> (transición 4)	1 ... 127 = Esclavo específico
128 = <i>Enter pre-operational</i> (transición 5)	
129 = <i>Reset node</i> (transición 6)	
130 = <i>Reset communication</i> (transición 7)	

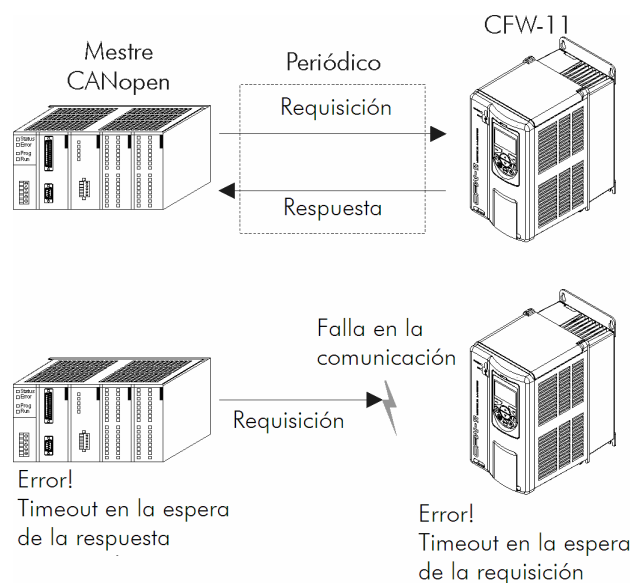
*Tabla 5.9 – Comandos para la transición de estados*

Las transiciones indicadas en el código del comando equivalen a las transiciones de estado ejecutadas por el nudo luego de recibir el comando (conforme Figura 5.5). El comando *Reset node* hace con que el CFW-11 ejecute un reset completo del dispositivo, mientras que el comando *Reset communication* hace con que el dispositivo reinicialice solo os objetos relativos a la comunicación CANopen.

### 5.6.2 Control de Errores – Node Guarding

Este servicio es utilizado para posibilitar el monitoreo de la comunicación con la red CANopen, tanto por el maestro cuanto por el esclavo. En este tipo de servicio, el maestro envía telegramas periódicos para el esclavo, que contesta el telegrama recibido. Caso ocurra algún error que interrumpa la comunicación, será posible identificar este error, pues tanto el maestro cuanto el esclavo serán notificados por el *timeout* en la ejecución de este servicio. Los eventos de error son llamados de *Node Guarding* para el maestro, y de *Life Guarding* para el esclavo.





**Figura 5.6** – Servicio de control de errores – Node Guarding

Para el servicio de *Node Guarding*, existen dos objetos del diccionario para configuración de los tiempos para detección de errores de comunicación:

Índice	100Ch
Nombre	Guard Time
Objeto	VAR
Tipo	UNSIGNED16
Acceso	rw
Mapeable	No
Rango	UNSIGNED16
Valor Padrón	0
Índice	100Dh
Nombre	Life Time Factor
Objeto	VAR
Tipo	UNSIGNED8
Acceso	rw
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	0

El objeto 100Ch permite programar el tiempo necesario (en milisegundos) para que una ocurrencia de falla sea detectada, caso el CFW-11 no reciba ninguno telegrama del maestro. El objeto 100Dh indica cuantas fallas en secuencia son necesarias hasta que se considere que ocurrió realmente un error de comunicación. Por lo tanto, la multiplicación de estos dos valores suministrará el tiempo total necesario para detección de errores de comunicación utilizando este objeto. El valor 0 (cero) deshabilita esta función.

Una vez configurado, el CFW-11 empieza a contar estos tiempos a partir del primero telegrama *Node Guarding* recibido del maestro de la red. El telegrama del maestro es del tipo remoto, no poseyendo bytes de datos. El identificador es igual a 700h + Node-ID del esclavo destino. Ya el telegrama de respuesta del esclavo posee 1 byte de datos con la siguiente estructura:

Identificador	byte 1	
	bit 7	bit 6 ... bit 0
700h + Node-ID	Toggle	Estado del esclavo

Este telegrama posee un único byte de datos. Este byte contiene, en los siete bits menos significativos, un valor para indicar el estado del esclavo (4 = Parado, 5 = Operacional y 127 = Preoperacional), y en el octavo bit, un valor que debe ser modificado a cada telegrama enviado por el esclavo (*toggle bit*).

Caso el CFW-11 detecte un error utilizando este mecanismo, él irá automáticamente para el estado preoperacional y indicará alarma A135 en su HMI. La ocurrencia de este error también puede ser observada a través del parámetro P0721. Es posible también programar el convertidor para tomar una acción cuando este error ocurra, a través del parámetro P0313. Consulte el ítem 3 para la descripción detallada de los parámetros.

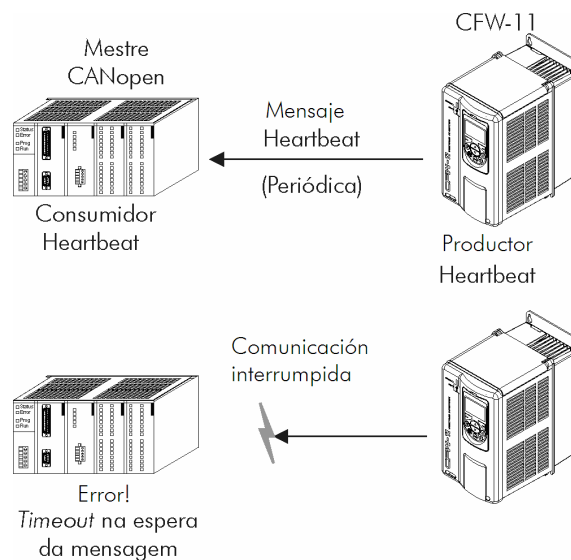


**¡NOTA!**

- ☑ Este objeto está activo mismo en el estado parado (mirar
- ☑ Tabla 5.8).
- ☑ El valor 0 (cero) en un de los dos objetos deshabilita esta función.
- ☑ Después de detectado el error, caso el servicio sea habilitado más una vez, la indicación del error es retirada de la HMI.
- ☑ El valor mínimo acepto para el CFW-11 es de 2 ms, más llevándose en cuenta la tasa de transmisión y el número de puntos en la red, los tiempos programados para esa función deben ser coherentes, de manera que haya tiempo suficiente para transmisión de los telegramas y también para que el resto de la comunicación pueda ser procesada.
- ☑ Para cada esclavo, solamente un de los servicios – *Heartbeat* o *Node Guarding* – puede ser habilitado.

**5.6.3 Control de Errores – *Heartbeat***

La detección de errores a través del mecanismo de *heartbeat* es hecha utilizando dos tipos de objetos: el productor *heartbeat* y el consumidor *heartbeat*. El productor es responsable por enviar telegramas periódicos para la red, simulando un batido del corazón, indicando que la comunicación está activa y sin errores. Un o más consumidores pueden monitorear estos telegramas periódicos y, caso estos telegramas dejen de ocurrir, significa que algún problema de comunicación ha ocurrido.



**Figura 5.7** – Servicio de control de errores – *Heartbeat*

Un mismo dispositivo de la red puede ser productor y consumidor de mensajes *heartbeat*. Por ejemplo, el maestro de la red puede consumir mensajes enviadas por un esclavo, permitiendo detectar problemas de comunicación con el esclavo, y al mismo tiempo el esclavo puede consumir mensajes *heartbeat* enviadas por el maestro, también posibilitando al esclavo detectar fallas en la comunicación con el maestro.

El CFW-11 posee los servicios de productor y consumidor *heartbeat*. Como consumidor, es posible programar hasta 4 diferentes productores para que sean monitoreados por el convertidor:

Índice	1016h
Nombre	Consumer Heartbeat Time
Objeto	ARRAY
Tipo	UNSIGNED32

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	-
Valor Padrón	4

Subíndices	1 – 4
Descripción	Consumer Heartbeat Time 1 – 4
Acceso	rw
Mapeable	No
Rango	UNSIGNED32
Valor Padrón	0

En los subíndices de 1 hasta 4, es posible programar el consumidor escribiendo un valor en el siguiente formato:

<i>UNSIGNED32</i>		
Reservado (8 bits)	Node-ID (8 bits)	Heartbeat time (16 bits)

- Node-ID:** permite programar el Node-ID del productor *heartbeat* el cual se desea monitorear.
- Heartbeat time:** permite programar el tiempo, en múltiplos de 1 milisegundo, hasta la detección de error, caso ningún mensaje del productor sea recibida. El valor 0 (cero) en este campo deshabilita el consumidor.

Después de configurado, el consumidor *heartbeat* inicia el monitoreo luego del primero telegrama enviado por el productor. Caso sea detectado error por el hecho del consumidor dejar de recibir mensajes del productos *heartbeat*, el convertidor irá automáticamente para el estado preoperacional e indicará alarma A135 en su HMI. La ocurrencia de este error también puede ser observada a través del parámetro P0721.

Como productor, el CFW-11 posee un objeto para configuración de este servicio:

Índice	1017h
Nombre	Producer Heartbeat Time
Objeto	VAR
Tipo	UNSIGNED16

Acceso	rw
Mapeable	No
Rango	UNSIGNED8
Valor Padrón	0

El objeto 1017h permite programar el tiempo en milisegundos en el cual el productor envíe un telegrama *heartbeat* para la red. Una vez programado, el convertidor inicia la transmisión de mensajes con el siguiente formato:

<i>Identificador</i>	<i>byte 1</i>	
	bit 7	bit 6 ... bit 0
700h + Node-ID	Siempre 0	Estado del esclavo

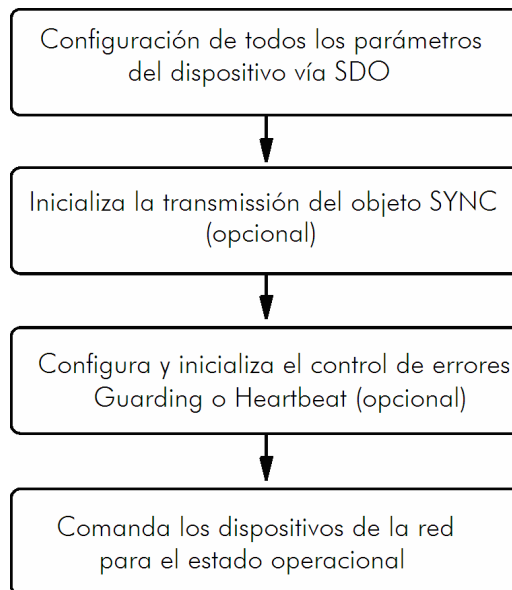


### ¡NOTA!

- Este objeto está activo mismo en el estado parado (ver tabla 5.8).
- El valor 0 (cero) en un de los objetos deshabilita esta función.
- Después de detectado el error, caso el servicio sea habilitado más una vez, la indicación del error es quitada de la HMI.
- El valor de tiempo programado para el consumidor debe ser mayor del que el programado para el respectivo productor. Recomendase inclusive programar el consumidor con valores múltiplos del utilizado para el productor.
- Para cada esclavo, solamente un de los servicios – *Heartbeat* o *Node Guarding* – puede ser habilitado.

## 5.7 PROCEDIMIENTO DE INICIALIZACIÓN

Una vez conocido el funcionamiento de los objetos disponibles para el convertidor de frecuencia CFW-11, es necesario ahora programar los diferentes objetos para operaren en conjunto en la red. De modo general, el procedimiento para la inicialización de los objetos en una red CANopen sigue es siguiente diagrama de flujo:



*Figura 5.8 – Diagrama de flujo del proceso de inicialización*

Es necesario observar que los objetos de comunicación del CFW-11 (1000h hasta 1FFFh) no son almacenados en la memoria no volátil. De este modo, siempre que fuera hecho el reset o apagado el equipo, es necesario rehacer la parametrización de los objetos de comunicación. Para los objetos específicos del fabricante (a partir de 2000h, que representan los parámetros), estos son almacenados en la memoria no volátil y, por lo tanto, pueden ser parametrizados una sola vez.

## 6 DESCRIPCIÓN DE LOS OBJETOS PARA DRIVES

En este ítem serán descritos los objetos comunes para drives, definidos por la especificación CANopen, en el documento CiA DSP 402. Los objetos mencionados aquí poseen descripción y operación semejantes, independiente del fabricante del drive. Eso facilita la interoperabilidad e intercambiabilidad entre diferentes dispositivos.

La Figura 6.1 presenta un diagrama con la arquitectura lógica de funcionamiento de un drive a través de la red CANopen, con los diferentes modos de operación definidos en esta especificación. Cada modo de operación posee un conjunto de objetos que permite la configuración y operación del drive en la red.

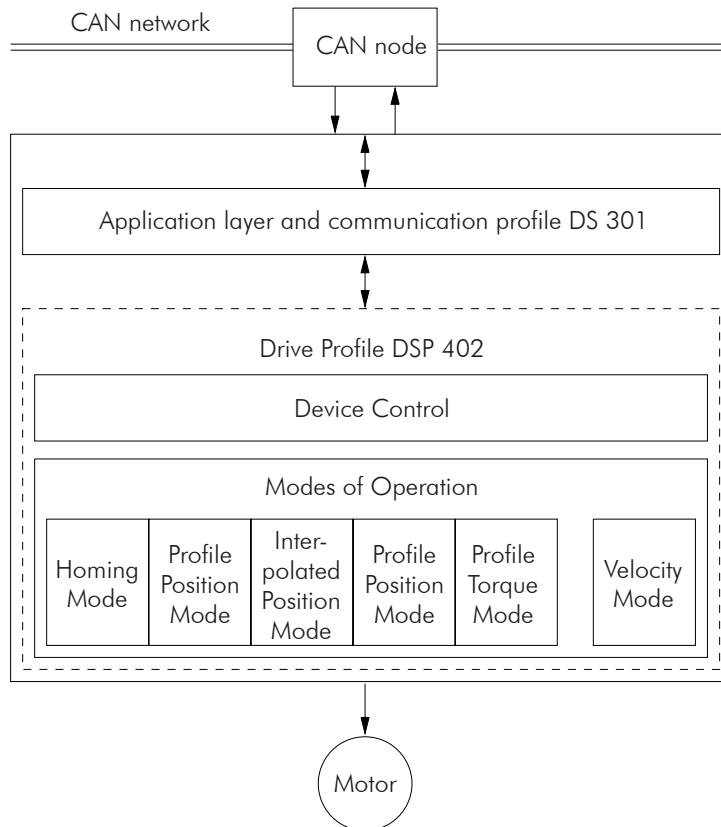


Figura 6.1 – Arquitectura de comunicación para un drive en la red CANopen

Para el CFW-11, solamente el *Velocity Mode* es soportado. La Tabla 6.1 presenta el listado de los objetos disponibles para el CFW-11, divididos de acuerdo con los diferentes modos de operación del convertidor.

Índice	Objeto	Nombre	Tipo	Acceso	Mapeable
Device Control					
6040h	VAR	Controlword	UNSIGNED16	rw	Sí
6041h	VAR	Statusword	UNSIGNED16	ro	Sí
6060h	VAR	Modes of Operation	INTEGER8	rw	Sí
6061h	VAR	Modes of Operation Display	INTEGER8	ro	Sí
Velocity Mode					
6042h	VAR	vl target velocity	INTEGER16	rw	Sí
6043h	VAR	vl velocity demand	INTEGER16	ro	Sí
6044h	VAR	vl control effort	INTEGER16	ro	Sí
6046h	ARRAY	vl velocity min max amount	UNSIGNED32	rw	Sí
6048h	RECORD	vl velocity acceleration	vl vel. accel. decl. record	rw	Sí
6049h	RECORD	vl velocity deceleration	vl vel. accel. decl. record	rw	Sí

Tabla 6.1 – Listado de objetos del CFW-11 – Drive Profile

Siempre que un objeto de este listado fuera leído o escrito, el CFW-11 irá mapear sus funciones en los parámetros del convertidor. De este modo, a operar el sistema a través de estos objetos, el valor de los parámetros puede ser modificado de acuerdo con la función utilizada. En los ítems siguientes, es hecho una descripción detallada de

cada un de estos objetos, donde son indicados los parámetros del convertidor que son utilizados para ejecutar las funciones de estos objetos.

## 6.1 DEVICE CONTROL – OBJETOS PARA CONTROL DEL DRIVE

Todo el drive que opera en una red CANopen siguiendo la especificación DSP 402 debe estar de acuerdo con el descrito en la siguiente máquina de estados:

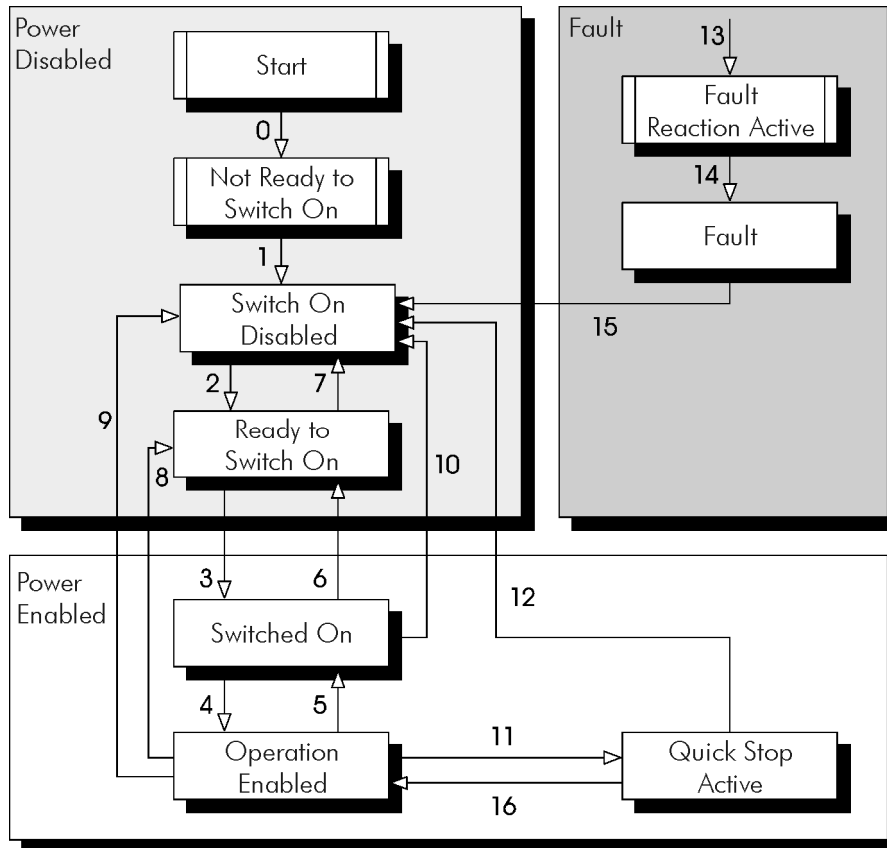


Figura 6.2 – Máquina de estado para drives

Descripción de los estados:

- ☑ **Not ready to switch on:** el convertidor está inicializando, no puede ser comandado.
- ☑ **Switch on disabled:** inicialización completa, convertidor puede recibir comandos.
- ☑ **Ready to switch on:** comando para permitir la alimentación del drive fue recibido.
- ☑ **Switched on:** comando para energizar la potencia del drive fue recibido.
- ☑ **Operation enabled:** el drive está habilitado, sendo controlado de acuerdo con el modo de operación programado. Potencia está sendo aplicada al motor.
- ☑ **Quick stop active:** durante la operación, el comando de *quick stop* fue recibido. Potencia está sendo aplicada al motor.
- ☑ **Fault reaction active:** una falla ocurrió y el drive está ejecutando la acción relativa al tipo de error.
- ☑ **Fault:** drive con error. Función deshabilitada, sin potencia sendo aplicada en el motor.



### ¡NOTA!

El convertidor de frecuencia CFW-11 no posee llave para bloqueo / habilitación de la alimentación de potencia del equipamiento. De esto modo, los estados descritos en el grupo *Power disabled* fueran implementados por cuestiones de compatibilidad con la máquina de estados descrita, sin embargo la alimentación de potencia del dispositivo permanece activa mismo en estos estados.

Descripción de las transiciones:

- ☑ **Transición 0:** el drive es encendido e inicia el procedimiento de inicialización. La alimentación de potencia del drive está activa.
- ☑ **Transición 1:** Inicialización completa (automático).
- ☑ **Transición 2:** Comando "Shutdown" recibido. Es hecha la transición de estados, más ninguna acción es tomada por el CFW-11.
- ☑ **Transición 3:** Comando "Switch on" recibido. Es hecha la transición de estados, más ninguna acción es tomada por el CFW-11.
- ☑ **Transición 4:** Comando "Enable operation" recibido. El drive es habilitado. Semejante a activar el bit 1 de la palabra de control vía CAN del convertidor – P0684.
- ☑ **Transición 5:** Comando "Disable operation" recibido. El drive es deshabilitado. Semejante a poner a cero el bit 1 de la palabra de control vía CAN del convertidor – P0684.
- ☑ **Transición 6:** Comando "Shutdown" recibido. Es hecha la transición de estados, más ninguna acción es tomada por el CFW-11.
- ☑ **Transición 7:** Comandos "Quick stop" y "Disable voltage" recibidos. Es hecha la transición de estados, más ninguna acción es tomada por el CFW-11.
- ☑ **Transición 8:** Comando "Shutdown" recibido. Durante la operación del drive este es deshabilitado, bloqueando la alimentación para el motor. Semejante a poner a cero el bit 1 de la palabra de control vía CAN del convertidor – P0684.
- ☑ **Transición 9:** Comando "Shutdown" recibido. Durante la operación del drive este es deshabilitado, bloqueando la alimentación para el motor. Semejante a poner a cero el bit 1 de la palabra de control vía CAN del convertidor – P0684.
- ☑ **Transición 10:** Comando "Quick stop" o "Disable voltage" recibido. Es hecha la transición de estados, más ninguna acción es tomada por el CFW-11.
- ☑ **Transición 11:** Comando "Quick stop" recibido. Convertidor ejecuta la función de parada por rampa. Semejante a poner a cero el bit 0 de la palabra de control vía CAN del convertidor – P0684.
- ☑ **Transición 12:** Comando "Disable voltage" recibido. El drive es deshabilitado. Semejante a poner a cero el bit 1 de la palabra de control vía CAN del convertidor – P0684..
- ☑ **Transición 13:** Error es detectado y el drive es deshabilitado.
- ☑ **Transición 14:** Después de deshabilitar el drive, él va para el estado de error (automático).
- ☑ **Transición 15:** Comando "Fault reset" recibido. El convertidor ejecuta el reset del error y retorna para el estado deshabilitado y sin falla.
- ☑ **Transición 16:** Comando "Enable operation" recibido. Convertidor ejecuta la función de arranque por rampa. Semejante a activar el bit 0 de la palabra de control vía CAN del convertidor – P0684.

Esta máquina de estado es controlada por el objeto 6040h, y los estados pueden ser monitoreados por el objeto 6041h. Ambos los objetos son presentados a seguir.

### 6.1.1 Objeto 6040h – Controlword

Hace el control del estado del convertidor.

Índice	6040h
Nombre	Controlword
Objeto	VAR
Tipo	UNSIGNED16
Parámetros utilizados	P0684
Acceso	rw
Mapeable	Sí
Rango	UNSIGNED16
Valor Padrón	-

Los bits de esta palabra poseen las siguientes funciones:

15 – 9	8	7	6 – 4	3	2	1	0
Reservado	Halt	Fault reset	Operation mode specific	Enable operation	Quick stop	Enable voltage	Switch on

Los bits 0, 1, 2, 3 y 7 permiten hacer el control de la máquina de estados del drive. Los comandos para transición de estados son hechos a través de las combinaciones de bits indicados en la tabla 26. Los bits marcados con 'x' son irrelevantes para la ejecución del comando.

Comando	Bits de la palabra de control					Transiciones
	Fault reset	Enable operation	Quick stop	Enable voltage	Switch on	
Shutdown	0	x	1	1	0	2, 6, 8
Switch on	0	0	1	1	1	3
Disable voltage	0	x	x	0	x	7, 9, 10, 12
Quick stop	0	x	0	1	x	7, 10, 11
Disable operation	0	0	1	1	1	5
Enable operation	0	1	1	1	1	4, 16
Fault reset	0 → 1	x	x	x	x	15

*Tabla 6.2 – Comandos de la palabra de control*

Los bits 4, 5, 6 y 8 poseen diferentes funciones de acuerdo con el modo de operación utilizado. La descripción detallada de la función de estos bits para el modo velocidad (velocity mode) está descrita en el ítem 6.2.1.

### 6.1.2 Objeto 6041h – Statusword

Indica el estado actual del CFW-11.

Índice	6041h
Nombre	Statusword
Objeto	VAR
Tipo	UNSIGNED16
Parámetros utilizados	P0680
Acceso	ro
Mapeable	Sí
Rango	UNSIGNED16
Valor Padrón	-

Los bits de esta palabra poseen las siguientes funciones:

Bit	Descripción
0	Ready to switch on
1	Switched on
2	Operation enabled
3	Fault
4	Voltage enabled
5	Quick stop
6	Switch on disabled
7	Warning
8	Reservado
9	Remote
10	Target reached
11	Internal limit active
12 – 13	Operation mode specific
14 – 15	Reservado

*Tabla 6.3 – Función de los bits de la palabra de estado (Statusword)*

En esta palabra, los bits 0, 1, 2, 3, 5 y 6 indican el estado del dispositivo de acuerdo con la máquina de estados presentada en la

Figura 6.2. La

Tabla 6.4 describe las combinaciones de estos bits para indicación de los estados. Los bits marcados con 'x' son irrelevantes para la indicación del estado.



Valor (binario)	Estado
xxxx xxxx x0xx 0000	Not ready to switch on
xxxx xxxx x1xx 0000	Switch on disabled
xxxx xxxx x01x 0001	Ready to switch on
xxxx xxxx x01x 0011	Switched on
xxxx xxxx x01x 0111	Operation enabled
xxxx xxxx x00x 0111	Quick stop active
xxxx xxxx x0xx 1111	Fault reaction active
xxxx xxxx x0xx 1000	Fault

**Tabla 6.4** – Estados del drive indicados a través de la palabra de estado

Demás bits indican una condición específica para el drive.

- ☑ **Bit 4 – Voltage enabled:** indica que la potencia del drive está sendo alimentada.
- ☑ **Bit 7 – Warning:** No utilizado para el CFW-11.
- ☑ **Bit 9 – Remote:** indica cuando el drive está en el modo remoto y acepta comandos vía red CANopen<sup>8</sup>. Representa el valor del bit 4 de la palabra de estado – P0680.
- ☑ **Bit 10 – Target reached:** indica cuando el drive está operando en el valor de la referencia, que depende del modo de operación utilizado. También es colocado en 1 cuando las funciones “quick stop” o “halt” son accionadas.
- ☑ **Bit 11 – Internal limit active:** indica cuando el drive está operando en un valor límite interno, como límite de corriente.
- ☑ **Bits 12 e 13 – Operation mode specific:** depende del modo de operación del drive.

### 6.1.3 Objeto 6060h – Modes of operation

Permite programar el modo de operación del CFW-11.

Índice	6060h
Nombre	Modes of operation
Objeto	VAR
Tipo	INTEGER8
Parámetros utilizados	-

Acceso	rw
Mapeable	Sí
Rango	INTEGER8
Valor Padrón	-

El único modo soportado por el convertidor CFW-11 es el “Velocity Mode”, representado por el valor 2.

### 6.1.4 Objeto 6061h – Modes of operation display

Indica el modo de operación del CFW-11.

Índice	6061h
Nombre	Modes of operation display
Objeto	VAR
Tipo	INTEGER8
Parámetros utilizados	-

Acceso	rw
Mapeable	Sí
Rango	INTEGER8
Valor Padrón	-

El único modo soportado por el convertidor CFW-11 es el “Velocity Mode”, representado por el valor 2.

<sup>8</sup> Depende de la programación del convertidor de frecuencia.

## 6.2 VELOCITY MODE – OBJETOS PARA CONTROL DEL DRIVE

Este modo de operación permite el control del convertidor de modo simple, disponiendo funciones del tipo:

- Cálculo del valor de referencia.
- Captura y monitoreo de la velocidad.
- Limitación de velocidad.
- Rampas de velocidad, dentro otras funciones.

Estas funciones son ejecutadas con base en un conjunto de objetos para configuración de este modo de operación.

### 6.2.1 Bits de control y estado

Los bits 4, 5, 6 y 8 de la palabra de control (objeto 6040h – Controlword) poseen las siguientes funciones en el modo velocidad:

Bit	Nombre	Valor	Descripción
4	<i>rfg enable</i> (habilita rampa)	0	Deshabilita Rampa – bit 0 del P0684 = 0
		1	Habilita Rampa – bit 0 del P0684 = 1
5	Reservado		
6	Reservado		
8	<i>Halt</i>	0	Habilita Rampa – bit 0 del P0684 = 1
		1	Deshabilita Rampa – bit 0 del P0684 = 0

Para que el motor gire de acuerdo con la rampa de aceleración, es necesario que el bits 4 sea activado.

Para la palabra de status, los bits específicos del modo de operación (bits 12 y 13) son reservados para uso futuro.

### 6.2.2 Objeto 6042h – *vl target velocity*

Permite programar el valor de la referencia de velocidad para el convertidor, en rpm:

Índice	6042h
Nombre	<i>vl target velocity</i>
Objeto	VAR
Tipo	INTEGER16
Parámetros utilizados	P0684, P0685
Acceso	rw
Mapeable	Sí
Rango	INTEGER16
Valor Padrón	0

El objeto *vl target velocity* permite la escritura de valores negativos para indicar referencia de velocidad para el motor en el sentido de giro antihorario. Esta modificación en el sentido de giro es ejecutada a través de la escritura del bit 2 en la palabra de comando vía CAN – P0684.

### 6.2.3 Objeto 6043h – *vl velocity demand*

Indica el valor de la referencia de velocidad luego de la rampa, en rpm:

Índice	6043h
Nombre	<i>vl velocity demand</i>
Objeto	VAR
Tipo	INTEGER16
Parámetros utilizados	-
Acceso	ro
Mapeable	Sí
Rango	INTEGER16
Valor Padrón	-

### 6.2.4 Objeto 6044h – vl control effort

Indica el valor de la velocidad de acuerdo con el medido el motor, en rpm. Para los modos de control sin realimentación, este objeto posee el mismo valor que el objeto 6043h.

Índice	6044h
Nombre	vl control effort
Objeto	VAR
Tipo	INTEGER16
Parámetros utilizados	P0681

Acceso	ro
Mapeable	Sí
Rango	INTEGER16
Valor Padrón	-

### 6.2.5 Objeto 6046h – vl velocity min max amount

Permite programar el valor de la velocidad mínima y máxima para el convertidor. Solamente valores positivos son aceptos, sin embargo los valores programados también son válidos para el sentido de giro antihorario. Los valores son escritos en "rpm".

Índice	6046h
Nombre	vl velocity min max amount
Objeto	ARRAY
Tipo	UNSIGNED32
Parámetros utilizados	P0133, P0134

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	2
Valor Padrón	2

Subíndice	1
Descripción	vl velocity min amount
Acceso	rw
Mapeable	Sí
Rango	UNSIGNED32
Valor Padrón	-

Subíndice	2
Descripción	vl velocity max amount
Acceso	rw
Mapeable	Sí
Rango	UNSIGNED32
Valor Padrón	-

### 6.2.6 Objeto 6048h – vl velocity acceleration

Permite programar la rampa de aceleración del convertidor.

Índice	6048h
Nombre	vl velocity acceleration
Objeto	RECORD
Tipo	Vl velocity acceleration deceleration record
Parámetros utilizados	P100

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	2
Valor Padrón	2

Subíndice	1
Descripción	Delta speed
Acceso	rw
Mapeable	Sí
Rango	UNSIGNED32
Valor Padrón	-

Subíndice	2
Descripción	Delta time
Acceso	rw
Mapeable	Sí
Rango	UNSIGNED32
Valor Padrón	-

El valor de la aceleración es calculado dividiendo la velocidad programada en el subíndice 1 por el tiempo programado en el subíndice 2. Los valores programados deben respetar el rango de valores para el parámetro P100.

### 6.2.7 Objeto 6049h – *vl velocity deceleration*

Permite programar la rampa de desaceleración del convertidor.

Índice	6049h
Nombre	vl velocity deceleration
Objeto	RECORD
Tipo	VI velocity acceleration deceleration record
Parámetros utilizados	P100

Subíndice	0
Descripción	Número del último subíndice
Acceso	ro
Mapeable	No
Rango	2
Valor Padrón	2

Subíndice	1
Descripción	Delta speed
Acceso	rw
Mapeable	Sí
Rango	UNSIGNED32
Valor Padrón	-

Subíndice	2
Descripción	Delta time
Acceso	rw
Mapeable	Sí
Rango	UNSIGNED32
Valor Padrón	-

El valor de la desaceleración es calculada dividiendo la velocidad programada en el subíndice 1 por el tiempo programado en el subíndice 2. Los valores programados deben respetar el rango de valores para el parámetro P101.

## 7 FALLAS Y ALARMAS RELACIONADAS CON LA COMUNICACIÓN CANOPEN

### A133 – Sin alimentación en la Interface CAN

#### Descripción:

Indica que la interface CAN no posee alimentación entre los terminales 1 y 5 del conector.

#### Actuación:

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

Si se detecta la falta de alimentación en la interface CAN, la comunicación es deshabilitada y será presentado A133 en la HMI del convertidor y este ejecutará la acción programada en el P0313. Caso la alimentación del circuito sea restablecida, la indicación de alarma será retirada de la HMI y la comunicación CAN será reiniciada.

#### Posibles Causas/Corrección:

- Medir si hay tensión entre los terminales 1 y 5 del conector de la interface CAN, conforme indicado en la
- Tabla 2.2.
- Verificar si los cables de alimentación no están cambiados o invertidos.
- Verificar problemas de contacto en el cable o en el conector de la interface CAN.

### A134 – Bus Off

#### Descripción:

Detectado error de *bus off* en la interface CAN.

#### Actuación:

Caso el número de errores de recepción o transmisión detectados por la interface CAN sea muy alto<sup>9</sup>, el controlador CAN puede ser llevado al estado de *bus off*, donde él interrumpe la comunicación y deshabilita la interface CAN.

Caso ocurra error de *bus off*, la comunicación CAN es deshabilitada y será presentado A134 en la HMI del convertidor y este ejecutará la acción programada en el P0313. Para que la comunicación sea restablecida, es necesario apagar y encender nuevamente el convertidor, o interrumpir y restablecer nuevamente la alimentación de la interface CAN, para que la comunicación sea reiniciada.

#### Posibles Causas/Corrección:

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

### A135 – Error de Node Guarding

#### Descripción:

Control de errores de la comunicación CANopen detecto error de comunicación utilizando el mecanismo de *guarding*.

#### Actuación:

---

<sup>9</sup> Para mayores informaciones a respecto de la detección de errores, consultar la especificación CAN.

El control de errores debe ser habilitado por el maestro de la red CANopen. Utilizando el *Node Guarding*, el maestro intercambia mensajes periódicos con el esclavo (en este caso, el CFW-11), en un período predeterminado. Caso la comunicación sea interrumpida por algún motivo, tanto el maestro cuanto el esclavo podrán detectar error en la comunicación por el *timeout* en el intercambio de estos mensajes.

Caso ocurra error de "*guarding*" para la comunicación CANopen, será señalado A135 en la HMI del convertidor y este ejecutará la acción programada en el P0313. Caso este control de errores sea habilitado nuevamente, la indicación de alarma será retirada de la HMI.

**Posibles Causas/Corrección:**

- Verificar los tiempos programados en el maestro y en el esclavo para intercambio de los mensajes. Para evitar problemas debido al retraso en la transmisión y diferencias en el conteo de los tiempos, recomendase que los valores programados para intercambio de los mensajes en el maestro sea un poco menor que los tiempos programados para detección de errores por el esclavo.
- Verificar si el maestro está enviando los telegramas de "*guarding*" en el tiempo programado.
- Verificar problemas en la comunicación que puedan ocasionar pérdida de telegramas o retrasos en la transmisión.

