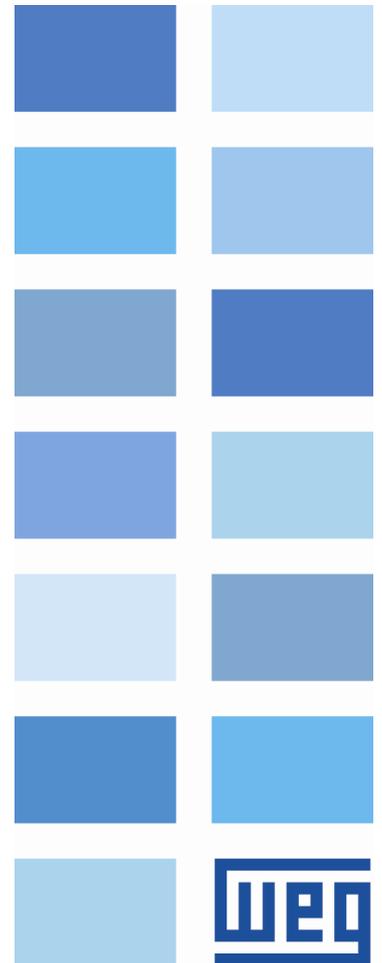


CANopen

SCW100

Manual del Usuario





Manual del Usuario de CANopen

Serie: SCW100

Idioma: Español

Documento: 10008601700 / 01

Build 221

Fecha de la Publicación: 04/2022

La información abajo describe las revisiones ocurridas en este manual.

Versión	Revisión	Descripción
-	R00	Cancelado
-	R01	Primera edición

ÍNDICE

A RESPECTO DEL MANUAL	6
ABREVIACIONES Y DEFINICIONES	6
REPRESENTACIÓN NUMÉRICA	6
DOCUMENTOS	6
1 CARACTERÍSTICAS PRINCIPALES	7
2 DESCRIPCIÓN DE LA INTERFAZ	8
2.1 CARACTERÍSTICAS DA INTERFACE CAN	8
2.2 CONECTOR	8
2.3 DIRECCIÓN	8
2.4 TASA DE COMUNICACIÓN	9
2.5 RESISTOR DE TERMINACIÓN	10
2.6 LED DE INDICACIÓN	10
3 INSTALACIÓN EN RED CANOPEN	11
3.1 TASA DE COMUNICACIÓN	11
3.2 DIRECCIÓN EN LA RED CANOPEN	11
3.3 RESISTOR DE TERMINACIÓN	11
3.4 CABLE	11
3.5 CONEXIÓN CON LA RED	12
4 S ESTADO	13
S4 COMUNICACIÓN	13
5 C CONFIGURACIÓN	15
C2 COMUNICACIÓN.....	15
6 OPERACIÓN EN LA RED CANOPEN	16
6.1 ACCESO A LOS DATOS	16
6.2 DATOS CICLICOS	16
6.3 DATOS ACICLICOS	16
6.4 OBJETOS RESPONSABLES POR LA COMUNICACIÓN - COB	16
6.5 COB-ID	17
6.6 ARCHIVO EDS	18
7 DICCIONARIO DE OBJETOS	19
7.1 ESTRUCTURA DEL DICCIONARIO	19
7.2 TIPOS DE DATOS	19
7.3 COMMUNICATION PROFILE - OBJETOS PARA COMUNICACIÓN	20
7.4 OBJETOS ESPECÍFICOS DEL FABRICANTE.....	20
8 DESCRIPCIÓN DE LOS OBJETOS DE COMUNICACIÓN	22
8.1 OBJETOS DE IDENTIFICACIÓN	22
8.1.1 Objeto 1000h - Device Type	22
8.1.2 Objeto 1001h - Error Register.....	22
8.1.3 Objeto 1018h - Identity Object.....	23
8.2 SERVICE DATA OBJECTS - SDOS	23
8.2.1 Objeto 1200h - Servidor SDO	23
8.2.2 Funcionamiento de los SDOs	24

8.3	PROCESS DATA OBJECTS - PDOS	25
8.3.1	Objetos Mapeables para los PDOs	25
8.3.2	PDOs de Recepción	26
8.3.3	PDOs de Transmisión	27
8.4	SYNCHRONIZATION OBJECT - SYNC	29
8.5	NETWORK MANAGEMENT - NMT	30
8.5.1	Control de los Estados del Esclavo	30
8.5.2	Control de Errores - Node Guarding	32
8.5.3	Control de Errores - Heartbeat	33
8.6	PROCEDIMIENTO DE INICIALIZACIÓN	35
9	PUESTA EN SERVICIO	36
9.1	INSTALAR DEL PRODUCTO EN LA RED	36
9.2	CONFIGURACIÓN DEL EQUIPO	36
9.3	CONFIGURACIÓN DEL MAESTRO	36
9.4	ESTADOS DE LA COMUNICACIÓN	37
9.5	OPERACIÓN UTILIZANDO DATOS DE PROCESO	37
9.6	ACCESO A LOS PARÁMETROS – MENSAJES ACÍCLICAS	37
10	FALLAS Y ALARMAS	38
	Apéndice A PARÁMETROS DE REFERENCIA RÁPIDA	39

A RESPECTO DEL MANUAL

Este manual provee la descripción necesaria para la operación del controlador de arranque SCW100 utilizando el protocolo CANopen. Este manual debe ser utilizado en conjunto con el manual del usuario del SCW100.

ABREVIACIONES Y DEFINICIONES

ASCII	American Standard Code for Information Interchange
CAN	Controller Area Network
CiA	CAN in Automation
CIP	Common Industrial Protocol
CRC	Cycling Redundancy Check
HMI	Human-Machine Interface
ISO	International Organization for Standardization
ODVA	Open DeviceNet Vendor Association
OSI	Open Systems Interconnection
PLC	Programmable Logic Controller
ro	Read only (solamente de lectura)
rw	Read/write (lectura y escrita)
RTR	Remote Transmission Request

REPRESENTACIÓN NUMÉRICA

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

DOCUMENTOS

El protocolo CANopen fue desarrollado con base 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 303-3 CANopen Indicator Specification	1.0	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
Planning and Installation Manual - DeviceNet Cable System	PUB00027R1	ODVA

1 CARACTERÍSTICAS PRINCIPALES

A seguir, son listadas las principales características para comunicación con el accesorio CANopen del controlador de arranque SCW100.

- Servicio de administrador de la red (NMT).
- 5 PDOs de transmisión.
- 2 PDOs de recepción.
- Consumidor Heartbeat.
- Productor Heartbeat.
- Node Guarding.
- Cliente SDO.
- Productor/consumidor SYNC.
- Es suministrado con el archivo EDS para configuración del maestro de la red.
- Pone a disposición datos acíclicos para parametrización.

2 DESCRIPCIÓN DE LA INTERFAZ

2.1 CARACTERÍSTICAS DA INTERFACE CAN

- Interfaz aislada galvánicamente y con señal diferencial, otorgando mayor robustez contra interferencia electromagnética.
- Alimentación externa de 24 V.
- Permite al equipo operar como esclavo CANopen.
- Permite la comunicación de datos para operación y para parametrización del equipo.
- Permite la comunicación usando tasas de 20 Kbit/s a 1 Mbit/s.
- Longitud máxima del embarrado de 1000 metros.

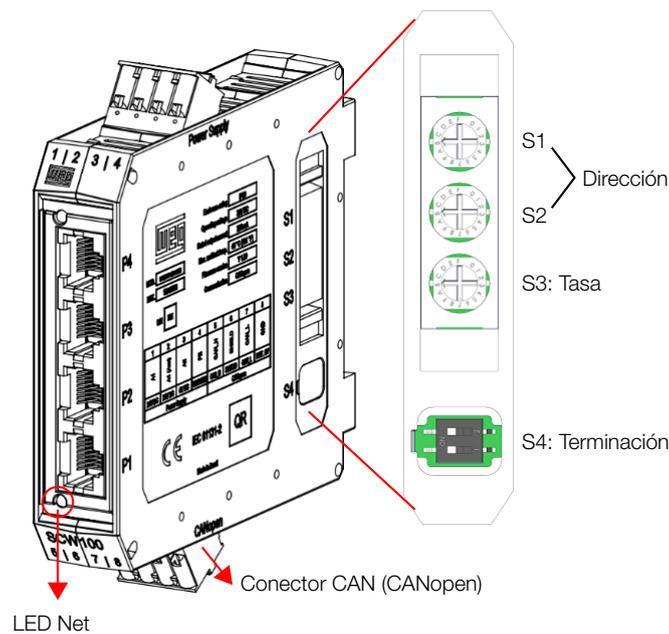


Figura 2.1: Conector, indicaciones y ajustes para el SCW100

2.2 CONECTOR

La interfaz CAN está disponible a través de un conector de 4 vías *plug-in* con los siguientes terminales:

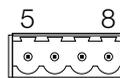


Tabla 2.1: Terminales del conector *plug-in* para CAN

Terminal	Nombre	Función
5	CAN_H	Señal de comunicación CAN_H
6	Shield	Blindaje del cable
7	CAN_L	Señal de comunicación CAN_L
8	CAN_GND	0V del circuito aislado para interfaz CAN

2.3 DIRECCIÓN

La dirección del producto en la red CANopen se configura con dos teclas hexadecimales S1 y S2 como se muestra en la figura 2.2.

- Direcciones válidas: 1 a 127 (01h a 7Fh)


¡NOTA!

Si se cambia la dirección, solo será válida después de que el producto se vuelva a encender.

La dirección seleccionada con las llaves representa un valor hexadecimal. Por lo tanto, debe convertirse a decimal, cuando sea necesario, en la parametrización del maestro CANopen.

Las llaves que configuran la dirección CANopen del esclavo se ilustran con más detalle en la figura 2.2, donde la llave S1 es la menos significativa y la llave S2 es la más significativa:

- Llave S1 = Eh
- Llave S2 = 2h

Dado que la llave S1 es la menos significativa, la dirección formada por las llaves es 2Eh, o 46 decimal.



Figura 2.2: Llaves rotativas para la configuración de la dirección

2.4 TASA DE COMUNICACIÓN

El controlador de arranque SCW100 tiene una llave hexadecimal S3 que le permite configurar la tasa de comunicación deseada. Según la posición de la llave, esta configuración se programa como se indica en la tabla 2.2.

Tabla 2.2: Configuración de la llave S3 para programar la tasa de comunicación

Ajuste de la llave	Tasa de Comunicación
0	1 Mbit/s
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


¡NOTA!

Si se cambia la configuración, solo será válida después de que el producto se vuelva a encender.

2.5 RESISTOR DE TERMINACIÓN

El producto posee la llave S4 que puede ser activada para habilitar el resistor de terminación conforme la figura 2.1. La configuración de las llaves para habilitar el resistor de terminación es presentada en la tabla 2.3.

Tabla 2.3: Configuraciones de la llave (S4) de habilitación del resistor de terminación

Ajuste de las Llaves	Opción
SW.1 = OFF y SW.2 = OFF	Terminación CAN apagada
SW.1 = ON y SW.2 = ON	Terminación CAN encendida
SW.1 = OFF y SW.2 = ON	Combinación no permitida
SW.1 = ON y SW.2 = OFF	

2.6 LED DE INDICACIÓN

El controlador de arranque SCW100 posee un LED bicolor (rojo y verde), como se muestra en la figura 2.1, las tablas a continuación muestran el comportamiento de estos LEDs según el estado de la controlador de arranque:

Tabla 2.4: LED NET - VERDE

Indicación	Estado	Descripción
Apagado	-	Sin alimentación.
Verde, un parpadeo	Stopped	El dispositivo está en estado detenido, de acuerdo con la especificación del protocolo CANopen
Verde, oscilando cada 200 ms	Pre-operational	El dispositivo está en el estado preoperacional. Los PDOS no están disponibles para la comunicación.
Verde sólido	Operational	El equipo se encuentra en el estado operacional.

Tabla 2.5: LED NET - ROJO

Indicación	Estado	Descripción
Apagado	Sin error	Equipo está operando normalmente.
Rojo, un parpadeo	Warning o Passive	La interfaz CAN está en el estado de Warning o Error Passive. Puede ocurrir, por ejemplo, si es el único equipo conectado a la red CANopen.
Rojo, dos parpadeos	Error de Node Guarding o Heartbeat	Control de errores de comunicación CANopen detectó un error de comunicación utilizando el mecanismo de guarding o heartbeat.
Rojo sólido	Error de BUS OFF	La interfaz CAN está en el estado de BUS OFF. Indica una condición operativa crítica en la red CANopen, generalmente asociada con problemas en la instalación o configuración incorrecta de la velocidad de comunicación. Es necesario apagar y encender el dispositivo para restablecer la comunicación.
Rojo, oscilando cada 50ms	CANopen no inicializado	Indica que el protocolo CANopen no se ha inicializado. Verifique que la dirección esté configurado en un valor válido (01h – 7Fh).

3 INSTALACIÓN EN RED CANOPEN

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



¡NOTA!

Recomendaciones detalladas sobre cómo realizar la instalación se pueden encontrar en el documento "Planning and Installation Manual" (ítem DOCUMENTOS).

3.1 TASA DE COMUNICACIÓN

Equipamientos con interfaz CANopen en general permiten configurar la tasa de comunicación deseada, pudiendo variar de 10 kbit/s hasta 1 Mbit/s. La tasa de comunicación (baud rate) que puede ser utilizada por un equipamiento depende de la longitud del cable utilizado en la instalación. La tabla 3.1 presenta las tasas de comunicación y la longitud máxima de cable que puede ser utilizado en la instalación, de acuerdo con el recomendado por la especificación del protocolo.

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

Tasa de Comunicación	Longitud del Cable
10 kbit/s	1000 m
20 kbit/s	1000 m
50 kbit/s	1000 m
100 kbit/s	600 m
125 kbit/s	500 m
250 kbit/s	250 m
500 kbit/s	100 m
800 kbit/s	50 m
1 Mbit/s	25 m

Todos los equipamientos de la red deben programarse para utilizar la misma tasa de comunicación.

3.2 DIRECCIÓN EN LA RED CANOPEN

Cada dispositivo de la red CANopen precisa tener una dirección, o Node-ID, entre 1 y 127. Esta dirección debe ser única para cada equipamiento.

3.3 RESISTOR DE TERMINACIÓN

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

3.4 CABLE

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

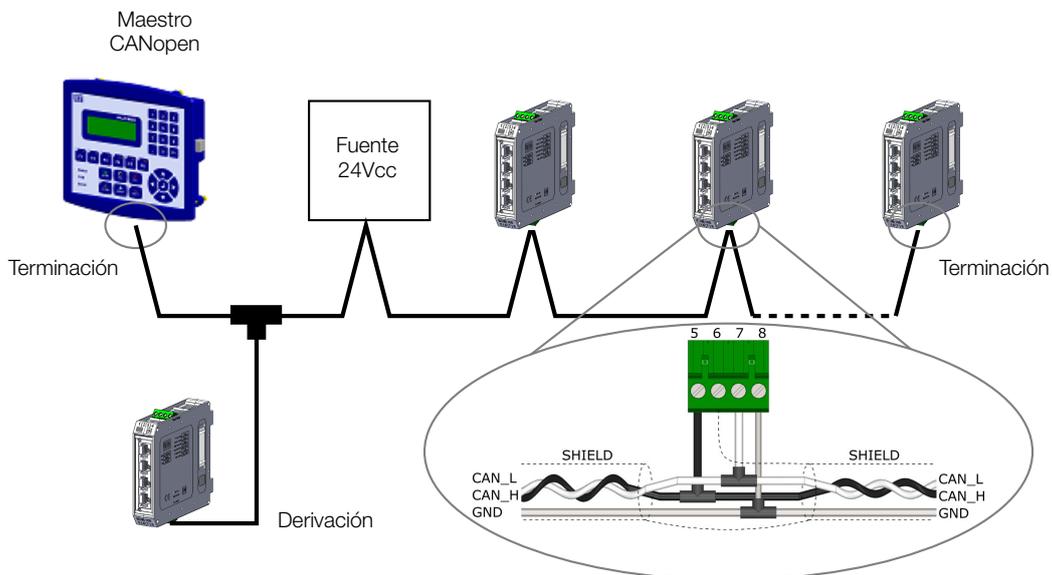
Tabla 3.2: *Propiedades del cable para red CANopen*

Longitud del Cable (m)	Resistencia por Metro (mΩ/m)	Área del Conductor (mm ²)
0 ... 40	70	0.25 ... 0.34
40 ... 300	<60	0.34 ... 0.60
300 ... 600	<40	0.50 ... 0.60
600 ... 1000	<26	0.75 ... 0.80

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

3.5 CONEXIÓN CON LA RED

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 a la interferencia electromagnética, eso facilita la ocurrencia de errores durante la transmisión.


Figura 3.1: *Ejemplo de instalación en red CANopen*

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.

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

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

4 S ESTADO

Permite acceder las variables de lectura del controlador de arranque SCW100.

S4 COMUNICACIÓN

Parámetros de lectura que muestran información sobre las interfaces de comunicación del producto.

S4 Comunicación

P0030: CAN - Dirección

Rango de valores: 1 ... 127

Estándar: 1

Descripción:

Permite ver la dirección utilizada para la comunicación CAN del dispositivo, programada usando las teclas 1 a 7 en la llave DIP S2.

S4 Comunicación

P0031: CAN - Tasa de Comunicación

Rango de valores: 0 ... 7

Estándar: 0

Descripción:

Permite ver el valor de la velocidad de comunicación de la interfaz CAN, programada en las teclas 8 a 10 en la llave DIP S2, en bits por segundo.

Indicación	Descripción
0 = 1 Mbit/s	Tasa de comunicación CAN.
1 = 800 Kbit/s	Tasa de comunicación CAN.
2 = 500 Kbit/s	Tasa de comunicación CAN.
3 = 250 Kbit/s	Tasa de comunicación CAN.
4 = 125 Kbit/s	Tasa de comunicación CAN.
5 = 100 Kbit/s	Tasa de comunicación CAN.
6 = 50 Kbit/s	Tasa de comunicación CAN.
7 = 20 Kbit/s	Tasa de comunicación CAN.

S4 Comunicación

P0033: CAN - Estado del Controlador

Rango de valores: 0 ... 5

Estándar: 0

Descripción:

Permite identificar si la interfaz CAN está activa y si la comunicación tiene errores.

Indicación	Descripción
0 = Inactivo	Interfaz CAN inactiva.
1 = Reservado	-
2 = CAN Activo	Interfaz CAN activa y sin errores.
3 = Warning	El controlador CAN alcanzó el estado de <i>warning</i> .
4 = Error Pasivo	El controlador CAN alcanzó el estado <i>error passive</i> .
5 = Bus Off	El controlador CAN alcanzó el estado de <i>bus off</i> .

S4 Comunicación

P0034: CAN - Telegramas CAN RX

Rango de valores: 0 ... 65535

Estándar: 0

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.

S4 Comunicación

P0035: CAN - Telegramas CAN TX

Rango de valores: 0 ... 65535

Estándar: 0

Descripción:

Este parámetro funciona como un contador cíclico, que es incrementado toda vez que un telegrama CAN es transmitido. Suministra un retorno para el operador si el dispositivo está consiguiendo comunicarse con la red.

S4 Comunicación

P0036: CAN - Contador Bus Off

Rango de valores: 0 ... 65535

Estándar: 0

Descripción:

Contador cíclico que indica el número de veces que el equipo ha entrado en el estado de bus off en la red CAN.

S4 Comunicación

P0037: CAN - Telegramas Perdidos

Rango de valores: 0 ... 65535

Estándar: 0

Descripción:

Contador cíclico que indica el número de mensajes recibidas por la interfaz CAN, más que no podrán ser procesadas por el equipo. Caso el número de mensajes perdidos sea incrementado con frecuencia, recomendase disminuir la tasa de comunicación utilizada para la red CAN.

S4 Comunicación

P0038: CAN - Estado de la Comunicación CANopen

Rango de valores: 0 ... 5

Estándar: 0

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*).

Indicación	Descripción
0 = Inactivo	Protocolo CANopen deshabilitado.
1 = Reservado	-
2 = Comunic. Hab.	Comunicación habilitada.
3 = Ctrl. Errors Hab.	Comunicación habilitada y control de errores habilitado (<i>Node Guarding/Heartbeat</i>).
4 = Error Guarding	Se produjo un error de <i>Node Guarding</i> .
5 = Error Heartbeat	Se produjo un error de <i>Heartbeat</i> .

S4 Comunicación

P0039: CAN - Estado del Esclavo CANopen

Rango de valores: 0 ... 4

Estándar: 0

Descripción:

Cada esclavo en la red CANopen posee una máquina de estado que controla su comportamiento en relación con la comunicación. Este parámetro indica en cual estado se encuentra el dispositivo.

Indicación	Descripción
0 = Inactivo	Protocolo CANopen deshabilitado.
1 = Inicialización	No es posible comunicarse con el dispositivo en esta etapa, que se completa automáticamente.
2 = Detenido	Solo el objeto NMT está disponible.
3 = Operacional	Todos los objetos de comunicación están disponibles.
4 = PreOperacional	Es posible comunicarse con el esclavo, pero los PDOs aún no están disponibles para la operación.

5 C CONFIGURACIÓN

Permite acceder las variables de escritura del controlador de arranque SCW100.

C2 COMUNICACIÓN

Parámetros de escritura que permiten configurar las interfaces de comunicación del producto.

C2 Comunicación

P0101: Acción para Falla en la Comunicación

Rango de valores: 0 ... 1

Estándar: 1

Descripción:

Permite configurar el modo de actuación de la protección erro de la comunicación.

Indicación	Descripción
0 = Sin Acción	No hay actuación.
1 = Apaga las Salidas	Apaga todas las salidas del producto. Es equivalente a recibir el comando de parada para los accionamientos y poner a cero la palabra de comando para las salidas digitales.

C2 Comunicación

P0032: CAN - Reset de Bus Off

Rango de valores: 0 ... 1

Estándar: 0

Descripción:

Permite programar el comportamiento del equipo al detectar un error de *bus off* en la interfaz CAN

Indicación	Descripción
0 = Manual	Si el bus se apaga, esta condición se indicará en los LED de indicación y la comunicación se deshabilitará. La acción programada en el parámetro P0101 - Acción para Falla en la Comunicación será ejecutada. Para que el equipo vuelva a se comunicar a través del interfaz CAN, será necesario desenergizar y energizar nuevamente el equipo.
1 = Automático	Caso ocurra bus off, la comunicación será reiniciada automáticamente y el error será ignorado. En este caso, no habrá indicación en los LED y la acción por error de comunicación no se realizará.

6 OPERACIÓN EN LA RED CANOPEN

6.1 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 está siendo transmitido.

6.2 DATOS CICLICOS

Los datos cíclicos son los que normalmente se utilizan para monitoreo del estado, así como para control de la operación del equipo. Para el protocolo CANopen, la Interfaz permite la comunicación de 2 PDOs de recepción y 5 PDOs de transmisión.

Es necesario que esta configuración sea realizada en el maestro de la red CANopen.

6.3 DATOS ACICLICOS

Además de los datos cíclicos, la Interfaz también pone a disposición datos acíclicos vía SDO. Utilizando este tipo de comunicación, es posible acceder a cualquier parámetro del equipo. El acceso a este tipo de dato normalmente es hecho usando instrucciones para lectura o escritura de los datos, donde se debe indicar el índice y sub-índice para el dato deseado. El ítem 7.4 describe cómo direccionar los parámetros del controlador de arranque SCW100.

6.4 OBJETOS RESPONSABLES POR LA COMUNICACIÓN - COB

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. Los siguientes objetos de comunicación (COBs) se describen en la especificación:

Tabla 6.1: Tipos de Objetos de Comunicación (COBs)

Tipo de Objeto	Descripción
Service Data Object (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.
Process Data Object (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.
Emergency Object (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.).
Synchronization Object (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.
Network Management (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 protocolo describe dos tipos de servicios: servicio de control del dispositivo, donde el maestro controla el estado de cada esclavo en la red, y servicios de control de errores (Node Guarding y Heartbeat), donde el dispositivo envía mensajes periódicas para informar que la conexión está activa.

Toda la comunicación del esclavo 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.

6.5 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 da 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. Es necesario observar que el valor padrón del objeto depende del endereço 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.

Tabla 6.2: COB-ID para los diferentes objetos

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)

6.6 ARCHIVO EDS

Cada dispositivo en una red CANopen tiene un archivo de configuración EDS, que contiene informaciones sobre el funcionamiento del dispositivo en la red. En general, este archivo es utilizado por un maestro o por un software de configuración, para programación de los dispositivos presentes en la red CANopen.

El archivo de configuración EDS está disponible en el sitio web WEG (<http://www.weg.net>). Es importante observar si el archivo de configuración EDS es compatible con la versión de firmware del controlador de arranque SCW100.

7 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 él ejecuta. Los objetos son dispuestos en el diccionario de la siguiente manera:

Tabla 7.1: Agrupamientos del diccionario de objetos

Índice	Objetos	Descripción
0001h - 025Fh	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.

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

7.1 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.

7.2 TIPOS DE DATOS

La primera parte del diccionario de objetos (índices 0001h – 025Fh) 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.

7.3 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 controlador de arranque SCW100.

Tabla 7.2: Lista de objetos – Communication Profile

Índice	Objeto	Nome	Tipo	Aceso
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
1016h	ARRAY	consume 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
Receive PDO Mapping Parameter				
1600h	RECORD	1st receive PDO mapping	PDO Mapping	rw
1601h	RECORD	2nd 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
1804h	RECORD	5th 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
1A04h	RECORD	5th transmit PDO mapping	PDO Mapping	rw

Estos objetos solamente pueden leerse y escribirse a través de la red CANopen, no están disponibles en otra interfaz de red. El maestro de la red, en general, es el equipamiento responsable por la configuración del equipamiento antes de iniciar la operación. El archivo de configuración EDS trae la lista de todos los objetos de comunicación soportados.

Para una descripción detallada de cuales objetos están disponibles en este rango del diccionario de objetos, consulte el ítem 8.

7.4 OBJETOS ESPECÍFICOS DEL FABRICANTE

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. Para el controlador de arranque SCW100, en este rango de objetos fue proporcionado todo el listado de parámetros. A través de estos parámetros es posible operar el SCW100, ejecutando cualquier función que el SCW100 pueda realizar. Los parámetros fueran dispuestos a partir del índice 2000h, y con el Net Id sumado a este índice para obtener su posición en el diccionario. Para identificar como están distribuidos los parámetros en el diccionario de objetos, consulte el ítem A.

Es necesario reconocer la operación del producto a través de los parámetros para que se pueda programar correctamente su operación vía red CANopen.

Para el listado completo y una descripción detallada de los parámetros, consulte el manual de programación del SCW100.

8 DESCRIPCIÓN DE LOS OBJETOS DE COMUNICACIÓN

En este ítem son descritos detalladamente cada uno de los objetos de comunicación disponibles para el controlador de arranque SCW100. Es necesario conocer como estos objetos son operados para utilizar las funciones disponibles para la comunicación del SCW100.

8.1 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.

8.1.1 Objeto 1000h - Device Type

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

Tabla 8.1: Objeto 1000h - Device Type

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1000h	0	Device Type	UNSIGNED32	RO	No	0

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.

8.1.2 Objeto 1001h - Error Register

Este objeto indica la ocurrencia o no de error en el dispositivo. El tipo de error registrado para el controlador de arranque es descrito conforme tabla 8.2.

Tabla 8.2: Objeto 1001h - Error Register

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1001h	0	Error register	UNSIGNED8	RO	yes	0

Tabla 8.3: Estructura del objeto Error Register

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

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.

8.1.3 Objeto 1018h - Identity Object

Trae informaciones generales a respecto del dispositivo.

Tabla 8.4: Objeto 1018h - Identity Object

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1018h	0	Número del último subíndice	UNSIGNED8	RO	No	4
	1	Vendor ID	UNSIGNED32	RO	No	0000.0123h
	2	Código do produto	UNSIGNED32	RO	No	0000.1200h
	3	Número da revisão	UNSIGNED32	RO	No	De acordo com a versão de firmware do equipamento
	4	Número serial	UNSIGNED32	RO	No	Diferente para cada SCW100

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 controlador de arranque SCW100 en red CANopen.

8.2 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.

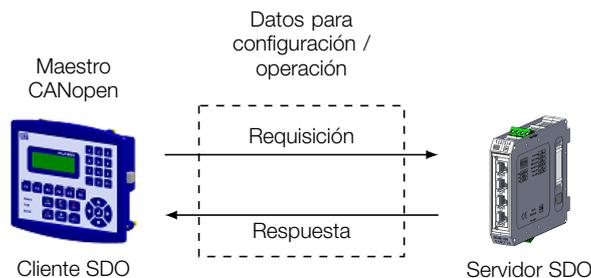


Figura 8.1: Comunicación entre cliente y servidor SDO

8.2.1 Objeto 1200h - Servidor SDO

EL controlador de arranque SCW100 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 SCW100. Todo servidor SDO posee un objeto, del tipo SDO_PARAMETER, para la su configuración, poseyendo la siguiente estructura:

Tabla 8.5: Objeto 1200h - Servidor SDO

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1200h	0	Número del último subíndice	UNSIGNED8	RO	No	2
	1	COB-ID Cliente - Servidor (rx)	UNSIGNED32	RO	No	600h + Node-ID
	2	COB-ID Servidor - Cliente (tx)	UNSIGNED32	RO	No	580h + Node-ID

8.2.2 Funcionamiento de los SDOs

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

Identificador	8 bytes de datos							
11 bits	Comando	Índice		Subíndice	Datos del objeto			
	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 este 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:

Tabla 8.6: Código de los comandos para cliente SDO

Comando	Función	Descripción	Datos del Objeto
22h	Download	Escrita en objeto	Indefinido
23h	Download	Escrita en objeto	4 byte
2Bh	Download	Escrita en objeto	2 byte
2Fh	Download	Escrita en objeto	1 byte
40h	Upload	Lectura de objeto	Não utilizado
60h ou 70h	Upload segment	Lectura segmentada	No utilizado

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:

Tabla 8.7: Código dos comandos para servidor SDO

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

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 SCW100 (objetos a partir del índice 2000h), este valor es guardado en la memoria no volátil del producto. 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 reescribir los valores deseados.

Ejemplo: un cliente SDO solicita para un esclavo 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

El esclavo contesta a la requisición, indicando que el valor para el referido objeto es igual a 999 ¹:

Identificador	Comando	Índice		Subíndice	Datos			
581h	4Bh	00h	20h	00h	E7h	03h	00h	00h

8.3 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.

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.

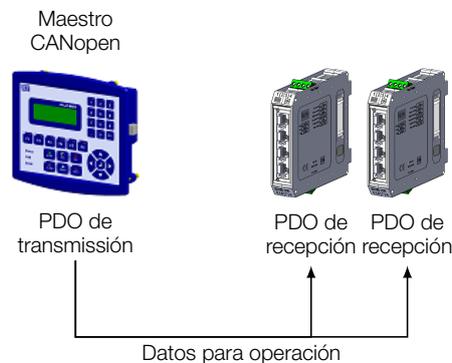


Figura 8.2: Comunicación utilizando PDOS



¡NOTA!

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

8.3.1 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.

¹No olvidar que cualquier dato del tipo entero, el orden de transferencia de los bytes va del menos significativo hasta el más significativo.

Para los objetos específicos del fabricante (2000h – 5FFFh), la tabla A presenta los 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 equipamiento trae un listado de todos los objetos disponibles, informando si el objeto es mapeable o no.

8.3.2 PDOs de Recepción

Los PDOs de recepción, o RPDOs, son responsables por recibir datos que otros dispositivos envían para la red CANopen. EL controlador de arranque SCW100 posee 2 PDOs de recepción, cada un 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	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1400h - 1401h	0	Número del último subíndice	UNSIGNED8	RO	No	2
	1	COB-ID usado por el PDO	UNSIGNED32	RW	No	200h / 300h + Node-ID
	2	Tipo de transmisión	UNSIGNED8	RW	No	254

O 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 elle 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:

Tabla 8.8: Descripción del COB-ID

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

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 SCW100 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.

Tabla 8.9: Descripción del tipo de transmisión

Tipo de transmisión	Transmisión de PDOs				
	Cíclico	Acíclico	Sincrónico	Asíncrono	RTR
0		•	•		
1 - 240	•		•		
241 - 251	Reservado				
252			•		•
253				•	•
254				•	
255				•	

- **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 e 253:** no permitido para PDOs de recepción.
- **Valores 254 e 255:** indica que no posee relación con el objeto de sincronización. Al recibir unos mensajes, sus valores serán actualizados inmediatamente.

PDO_MAPPING

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1600h - 1601h	0	Número de objetos mapeados	0 = deshabilitado 1-4=número de objetos mapeados	RO	No	0
	1 - 4	1º hasta 4º objeto mapeado no PDO	UNSIGNED32	RW	No	Indicado en el archivo EDS

Este parámetro indica los objetos mapeados en los PDOs de recepción el controlador de arranque SCW100. El valor estándar de estos objetos se indica en el archivo EDS del producto. Para cada RPDO, es posible mapear hasta 4 objetos diferentes, desde que el tamaño total no ultrapase ocho bytes. El mapeado de un objeto es hecho indicando su índice, subíndice² 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 = 3:** el RPDO posee tres objetos mapeados.
- **Subíndice 1 = 244C.0010h:** el primero objeto mapeado posee índice igual a 244Ch, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P1100 - Comando de Arranque Directo.
- **Subíndice 2 = 2451.0010h:** el segundo objeto mapeado posee índice igual a 2451h, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P1105 - Comando de Arranque Reverso.
- **Subíndice 3 = 2456.0010h:** el segundo objeto mapeado posee índice igual a 2456h, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P1110 - Comando de Parada.

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.
- No olvidar que los PDOs solamente pueden ser recibidos caso el SCW100 se encuentre en el estado operacional.

8.3.3 PDOs de Transmisión

Los PDOs de transmisión, o TPDOs, como el nombre dice, son responsables por transmitir datos para la red CANopen. EL controlador de arranque SCW100 posee 5 PDOs de transmisión, cada un pudiendo transmitir hasta

²Si el objeto es del tipo VAR y no tiene sub-índice, el valor 0 (cero) debe ser indicado para el subíndice.

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	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1800h - 1804h	0	Número del último subíndice	UNSIGNED8	RO	No	5
	1	COB-ID usado por el PDO	UNSIGNED32	RW	No	180h / 280h /380h / 480h / 580h + Node-ID
	2	Tipo de transmisión	UNSIGNED8	RW	No	254
	3	Tiempo entre transmisiones	UNSIGNED16	RW	No	-
	4	Entrada de compatibilidad	UNSIGNED8	RW	No	-
	5	Temporizador de eventos	UNSIGNED16	RW	No	0

El subíndice 1 contiene el COB-ID del PDO de transmisión. Siempre que este PDO enviar un mensaje para la red, el identificador de este mensaje será este COB-ID. La estructura de este campo es descrita en la tabla 8.8.

El subíndice 2 indica el tipo de transmisión de este objeto, que sigue descrito por la tabla 8.9. 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.
- **Valor 254:** el objeto debe ser transmitido de acuerdo con el timer programado en el subíndice 5.
- **Valor 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 100 μ s) 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 4 no tiene función y existe solo por razones de compatibilidad.

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 1 ms), 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 controlador de arranque SCW100 es 2 ms.
- 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	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1A00h - 1A04h	0	Número del último subíndice	UNSIGNED8	RO	No	0
	1 - 4	1º hasta 4º objeto mapeado no PDO	UNSIGNED32	RW	No	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:

UNSIGNED32		
Índice (16 bits)	Sub-índice (8 bits)	Tamaño del objeto (8 bits)

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

- **Subíndice 0 = 4:** este TPDO posee cuatro objetos mapeados.
- **Subíndice 1 = 20C8.0010h:** el primero objeto mapeado posee índice igual a 20C8h, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0200 - Status P1.
- **Subíndice 2 = 20C9.0010h:** el segundo objeto mapeado posee índice igual a 20C9h, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0201 - Status P1 - Contactor.
- **Subíndice 3 = 20CA.0010h:** el segundo objeto mapeado posee índice igual a 20CAh, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0202 - Status P1 - Error.
- **Subíndice 4 = 20CB.0010h:** el segundo objeto mapeado posee índice igual a 20CBh, subíndice 0 (cero), y tamaño igual a 16 bits. Este objeto corresponde al parámetro P0203 - Status P1 - Alarma.

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.

8.4 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 controlador de arranque SCW100 posee la función de consumidor SYNC y, por lo tanto, puede programar sus PDOs para que sean sincrónicos. 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.

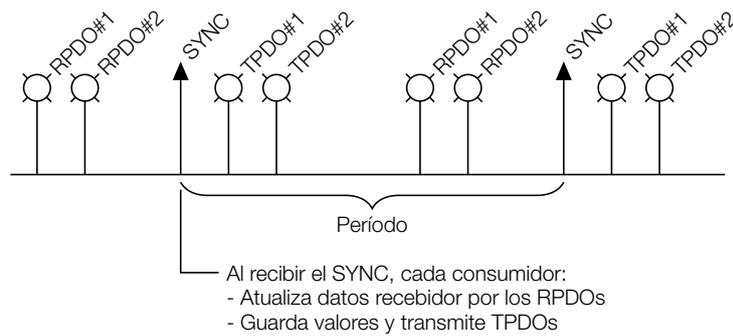


Figura 8.3: SYNC

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. El objeto siguiente está disponible para configuración del consumidor SYNC:

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1005h	0	COB-ID SYNC	UNSIGNED32	RW	No	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.

8.5 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 objeto están disponibles los servicios de control del nudo y de control de errores (utilizando Node Guarding o Heartbeat).

8.5.1 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:

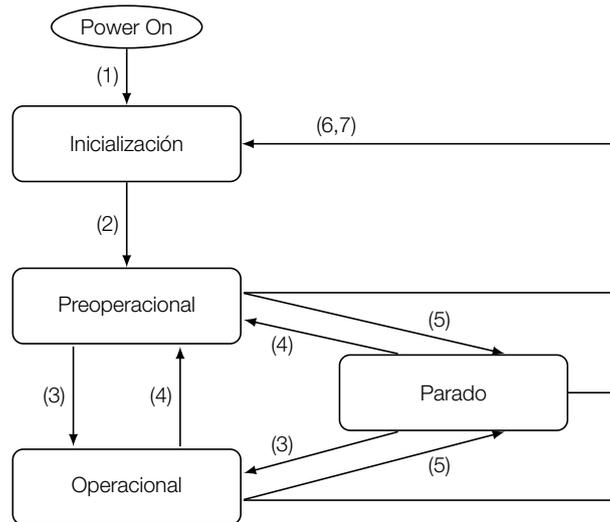

Figura 8.4: Diagrama de estados del nudo CANopen

Tabla 8.10: Descripción de las transiciones

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

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.

Tabla 8.11: Objetos accesibles en cada estado

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

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

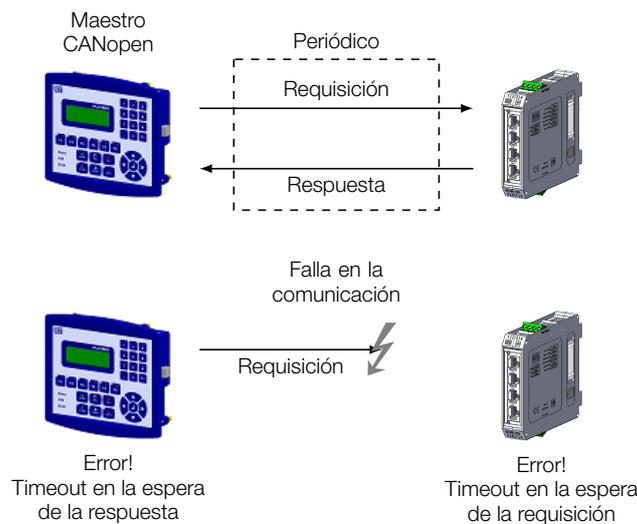
Tabla 8.12: Comandos para la transición de estados

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

Las transiciones indicadas en el código del comando equivalen a las transiciones de estado ejecutadas por el nudo luego de recibir el comando (conforme la figura 8.4). El comando Reset node hace con que el esclavo ejecute un reset completo del dispositivo, mientras que el comando Reset communication hace con que el esclavo reinicialice solo os objetos relativos a la comunicación CANopen.

8.5.2 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 8.5: 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	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
100Ch	0	Guard Time	UNSIGNED16	RW	No	0

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
100Dh	0	Life Time Factor	UNSIGNED8	RW	No	0

El objeto 100Ch permite programar el tiempo necesario (en milisegundos) para que una ocurrencia de falla sea detectada, caso el esclavo 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 esclavo 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 ... 0
700h + Node ID	Toogle	Estado do 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 controlador de arranque SCW100 detecte un error utilizando este mecanismo, elle irá automáticamente para el estado preoperacional y indicar con el LED de error.



¡NOTA!

- Este objeto está activo mismo en el estado parado (mirar tabla 8.11).
- 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.
- El valor mínimo acepto para controlador de arranque SCW100 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.

8.5.3 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.

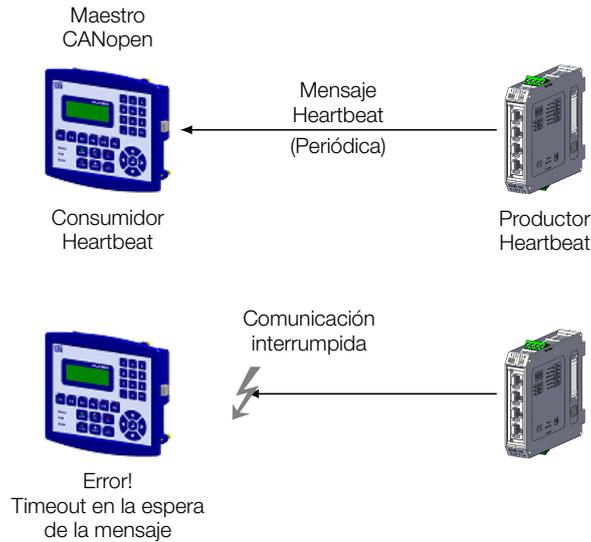


Figura 8.6: Servicio de control de errores – Heartbeat

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 controlador de arranque SCW100 posee los servicios de productor y consumidor heartbeat. Como consumidor, es posible programar diferentes productores para que sean monitoreados por el equipamiento:

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1016h	0	Número del último subíndice	UNSIGNED8	RO	No	4
	1 hasta 4	Consumer Heartbeat Time 1 hasta 4	UNSIGNED32	RW	No	0

En los subíndices de 1 hasta 2, es posible programar el consumidor escribiendo un valor en el siguiente formato:

UNSIGNED32		
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.
- 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 productor heartbeat, el dispositivo irá automáticamente para el estado preoperacional y indicar con el LED de erro.

Como productor, el controlador de arranque SCW100 posee un objeto para configuración de este servicio:

Índice	Sub-índice	Nombre	Tipo	Acceso	PDO Mapping	Valor
1017h	0	Producer Heartbeat Time	UNSIGNED16	RW	No	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 dispositivo inicia la transmisión de mensajes con el siguiente formato:

Identificador	byte 1	
	bit 7	bit 6 ... 0
700h + Node ID	Siempre 0	Estado del Esclavo


¡NOTA!

- Este objeto está activo mismo en el estado parado (mirar tabla 8.11).
- 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.
- El valor mínimo acepto para el controlador de arranque SCW100 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.

8.6 PROCEDIMIENTO DE INICIALIZACIÓN

Una vez conocido el funcionamiento de los objetos disponibles para el controlador de arranque SCW100 operando en modo esclavo, 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:

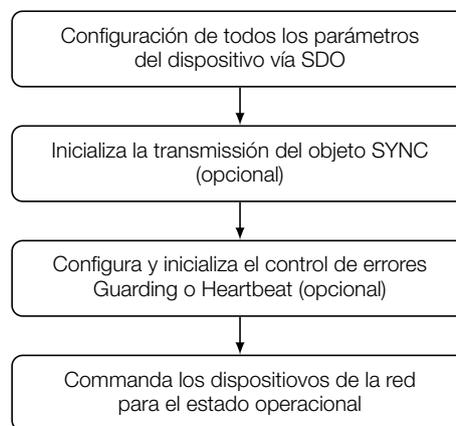


Figura 8.7: Diagrama de flujo del proceso de inicialización

Es necesario observar que los objetos de comunicación del controlador de arranque SCW100 (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.

9 PUESTA EN SERVICIO

A seguir son descritos los principales pasos para puesta en funcionamiento del controlador de arranque SCW100 en red CANopen. Los pasos descritos representan un ejemplo de uso. Consulte los capítulos específicos para detalles sobre los pasos indicados.

9.1 INSTALAR DEL PRODUCTO EN LA RED

1. Instale el controlador de arranque SCW100 en la red CANopen y hace las configuraciones necesarias para la operación conforme es descrito en el ítem 2.
2. Conecte los cables, considerando los cuidados necesarios en la instalación de la red, conforme es descrito en el ítem 3.5:
 - Utilice cable blindado.
 - Ponga a tierra adecuadamente los equipos de la red.
 - Evite el pasaje de los cables de comunicación cerca de los cables de potencia.

9.2 CONFIGURACIÓN DEL EQUIPO

1. Seguir las recomendaciones descritas en el manual del usuario para programar parámetros de ajuste del equipo.
2. Programe los ajustes de comunicación, como dirección y tasa de comunicación en el DIP switch S2.
3. Programar la acción deseada para el equipo en caso de falla en la comunicación, a través del P0101 - Acción para Falla en la Comunicación.

9.3 CONFIGURACIÓN DEL MAESTRO

La forma en la cual es hecha la configuración de la red depende en gran parte del maestro utilizado y de la herramienta de configuración. Es fundamental conocer las herramientas utilizadas para realizar esta actividad. De forma general, para realizar la configuración de la red son necesarios los siguientes pasos.

1. Cargue el archivo de configuración EDS³ para la lista de equipos en la herramienta de configuración de la red.
2. Seleccione el controlador de arranque SCW100 en la lista de equipos disponibles en el configurador de la red. Esto puede ser hecho manualmente o de forma automática, si la herramienta así lo permite.
3. Durante la configuración de la red, es necesario definir qué datos serán leídos y escritos en el controlador de arranque SCW100 configurando los PDOs de transmisión y recepción, conforme es descrito en el ítem 8.3. Entre los principales parámetros que pueden ser utilizados para control podemos citar:
4. Configurar el control de errores utilizando los servicios Node Guarding o Heartbeat, conforme es descrito en el ítem 8.5.

Si todo está correctamente configurado, el estado de la red en P0038 - CAN - Estado de la Comunicación CANopen indica 2 o 3 y el estado del nudo en P0039 - CAN - Estado del Esclavo CANopen indica 3. Es en esta condición que ocurre efectivamente la transmisión de los PDOs.

³El archivo de configuración EDS está disponible en el sitio web WEG (<http://www.weg.net>). Es importante observar si el archivo de configuración EDS es compatible con la versión de firmware del controlador de arranque SCW100.

9.4 ESTADOS DE LA COMUNICACIÓN

Una vez que la red esté montada y el maestro programado, será posible utilizar los LED Net y parámetros del equipo para identificar algunos estados relacionados a la comunicación.

- El LED Net suministran informaciones sobre el estado de la Interfaz.
- Los parámetros P0038 - CAN - Estado de la Comunicación CANopen y P0039 - CAN - Estado del Esclavo CANopen suministran informaciones sobre la comunicación CANopen.

El maestro de la red también deberá proveer informaciones sobre la comunicación con el esclavo.

9.5 OPERACIÓN UTILIZANDO DATOS DE PROCESO

Una vez que la comunicación esté establecida, los datos mapeados en los PDOs son automáticamente actualizados.

Para programar el maestro, conforme es deseado para la aplicación, es importante conocer estos parámetros.

9.6 ACCESO A LOS PARÁMETROS – MENSAJES ACÍCLICAS

Además de la comunicación cíclica utilizando los PDOs, el protocolo CANopen también define un tipo de mensaje acíclico vía SDO, utilizado principalmente en tareas asíncronas tales como parametrización y configuración del equipamiento.

El archivo de configuración EDS contiene el listado completo de los parámetros del equipamiento, los cuales pueden ser accedidos vía SDO. El ítem 7.4 describe cómo direccionar los parámetros del controlador de arranque SCW100 vía SDO.

10 FALLAS Y ALARMAS

Falla/Alarma	Descripción	Causas Más Probables
Bus Off	<p>Detectado error de bus off en el interfaz CAN. Caso el número de errores de recepción o transmisión detectados por el interfaz CAN sea muy alto, el controlador CAN puede ser llevado al estado de bus off, donde éste interrumpe la comunicación y deshabilita la interfaz CAN.</p> <p>Para que la comunicación sea restablecida es necesario interrumpir y regresar nuevamente la alimentación para el producto o interrumpir y regresar nuevamente la alimentación del interfaz CAN; para que la comunicación sea reiniciada.</p> <p>En este caso, se señalará mediante el LED Net situado en el controlador de arranque. La comunicación se restablece automáticamente si el parámetro P0032 - CAN - Reset de Bus Off tiene un valor de 1. Si el valor del parámetro P0032 - CAN - Reset de Bus Off es 0, será necesario apagar y encender la unidad remota para restablecer la comunicación.</p>	<ul style="list-style-type: none"> - Verificar cortocircuito en los cables de transmisión del circuito CAN. - Verificar si los cables no están cambiados o invertidos. - Verificar si todos los dispositivos de la red utilizan la misma tasa de comunicación. - Verificar si resistores de terminación con valores correctos fueran colocados solamente en los extremos del bus principal. - Verificar si la instalación de la red CAN fue hecha de manera adecuada.
CANopen Offline	<p>Actúa cuando el estado del nudo CANopen pasa de operacional para preoperacional.</p>	<ul style="list-style-type: none"> - Verificar el funcionamiento de los mecanismos de control de errores (Heartbeat/Node Guarding). - Verificar si el maestro está enviando los telegramas de "guarding"/"heartbeat" en el tiempo programado. - Verificar problemas en la comunicación que puedan ocasionar pérdida de telegramas o retrasos en la transmisión.

APÉNDICE A PARÁMETROS DE REFERENCIA RÁPIDA

S	Estado	S1	Información del Producto	40
		S2	Arranques	40
		S3	Errores y Alamas	41
		S4	Comunicación	42
C	Configuración	C1	Arranques	43
		C2	Comunicación	43
		C3	Contadores	43
		C4	Comandos	43

Tabla A.2

						CANopen Index	Sub-Index
S1 Estado\Información del Producto							
P0001	Versión de Firmware	0,0 a 655,35	-	ro, 16bit	1	2001h	0
P0010	Llave Rotativa S1	0 a 15	-	ro, 8bit	10	200Ah	0
P0011	Llave Rotativa S2	0 a 15	-	ro, 8bit	11	200Bh	0
P0012	Llave Rotativa S3	0 a 15	-	ro, 8bit	12	200Ch	0
P0015	Estados de las Entradas Digitales	Bit 0 = DI1 Bit 1 = DI2 Bit 2 = DI3 Bit 3 = DI4 Bit 4 = DI5 Bit 5 = DI6 Bit 6 = DI7 Bit 7 = DI8 Bit 8 = DI9 Bit 9 = DI10 Bit 10 = DI11 Bit 11 = DI12	-	ro, 16bit	15	200Fh	0
P0016	Estados de las Salidas digitales	Bit 0 = DO1 Bit 1 = DO2 Bit 2 = DO3 Bit 3 = DO4 Bit 4 = DO5 Bit 5 = DO6 Bit 6 = DO7 Bit 7 = DO8	-	ro, 16bit	16	2010h	0
P0500	Temperatura de la CPU	-100 a 100 °C	-	ro, s16bit	500	21F4h	0
S2 Estado\Arranques							
P0120	P1 Tiempo de Cierre Contactor 1	0 a 65535 ms	-	ro, 16bit	120	2078h	0
P0121	P1 Tiempo de Apertura Contactor 1	0 a 65535 ms	-	ro, 16bit	121	2079h	0
P0122	P1 Tiempo de Cierre Contactor 2	0 a 65535 ms	-	ro, 16bit	122	207Ah	0
P0123	P1 Tiempo de Apertura Contactor 2	0 a 65535 ms	-	ro, 16bit	123	207Bh	0
P0124	P2 Tiempo de Cierre Contactor 1	0 a 65535 ms	-	ro, 16bit	124	207Ch	0
P0125	P2 Tiempo de Apertura Contactor 1	0 a 65535 ms	-	ro, 16bit	125	207Dh	0
P0126	P2 Tiempo de Cierre Contactor 2	0 a 65535 ms	-	ro, 16bit	126	207Eh	0
P0127	P2 Tiempo de Apertura Contactor 2	0 a 65535 ms	-	ro, 16bit	127	207Fh	0
P0128	P3 Tiempo de Cierre Contactor 1	0 a 65535 ms	-	ro, 16bit	128	2080h	0
P0129	P3 Tiempo de Apertura Contactor 1	0 a 65535 ms	-	ro, 16bit	129	2081h	0
P0130	P3 Tiempo de Cierre Contactor 2	0 a 65535 ms	-	ro, 16bit	130	2082h	0
P0131	P3 Tiempo de Apertura Contactor 2	0 a 65535 ms	-	ro, 16bit	131	2083h	0
P0132	P4 Tiempo de Cierre Contactor 1	0 a 65535 ms	-	ro, 16bit	132	2084h	0
P0133	P4 Tiempo de Apertura Contactor 1	0 a 65535 ms	-	ro, 16bit	133	2085h	0
P0134	P4 Tiempo de Cierre Contactor 2	0 a 65535 ms	-	ro, 16bit	134	2086h	0
P0135	P4 Tiempo de Apertura Contactor 2	0 a 65535 ms	-	ro, 16bit	135	2087h	0





						CANopen Index	Sub-Index
P0150	Contador de Maniobras P1 C1	0 a 65535	-	ro, 16bit	150	2096h	0
P0151	Contador de Maniobras P1 C2	0 a 65535	-	ro, 16bit	151	2097h	0
P0152	Contador de Maniobras P2 C1	0 a 65535	-	ro, 16bit	152	2098h	0
P0153	Contador de Maniobras P2 C2	0 a 65535	-	ro, 16bit	153	2099h	0
P0154	Contador de Maniobras P3 C1	0 a 65535	-	ro, 16bit	154	209Ah	0
P0155	Contador de Maniobras P3 C2	0 a 65535	-	ro, 16bit	155	209Bh	0
P0156	Contador de Maniobras P4 C1	0 a 65535	-	ro, 16bit	156	209Ch	0
P0157	Contador de Maniobras P4 C2	0 a 65535	-	ro, 16bit	157	209Dh	0
P0200	Status P1	0 a 65535	-	ro, 16bit	200	20C8h	0
P0201	Status P1 - Contactor	0 a 1	-	ro, 16bit	201	20C9h	0
P0202	Status P1 - Error	0 a 65535	-	ro, 16bit	202	20CAh	0
P0203	Status P1 - Alarma	0 a 65535	-	ro, 16bit	203	20CBh	0
P0204	Status P2	0 a 65535	-	ro, 16bit	204	20CCh	0
P0205	Status P2 - Contactor	0 a 1	-	ro, 16bit	205	20CDh	0
P0206	Status P2 - Error	0 a 65535	-	ro, 16bit	206	20CEh	0
P0207	Status P2 - Alarma	0 a 65535	-	ro, 16bit	207	20CFh	0
P0208	Status P3	0 a 65535	-	ro, 16bit	208	20D0h	0
P0209	Status P3 - Contactor	0 a 1	-	ro, 16bit	209	20D1h	0
P0210	Status P3 - Error	0 a 65535	-	ro, 16bit	210	20D2h	0
P0211	Status P3 - Alarma	0 a 65535	-	ro, 16bit	211	20D3h	0
P0212	Status P4	0 a 65535	-	ro, 16bit	212	20D4h	0
P0213	Status P4 - Contactor	0 a 1	-	ro, 16bit	213	20D5h	0
P0214	Status P4 - Error	0 a 65535	-	ro, 16bit	214	20D6h	0
P0215	Status P4 - Alarma	0 a 65535	-	ro, 16bit	215	20D7h	0
S3 Estado\Errores y Alamas							
P0300	P1 - Último Error 1	0 a 65535	-	ro, 16bit	300	212Ch	0
P0301	P1 - Último Error 2	0 a 65535	-	ro, 16bit	301	212Dh	0
P0302	P1 - Último Error 3	0 a 65535	-	ro, 16bit	302	212Eh	0
P0303	P1 - Último Error 4	0 a 65535	-	ro, 16bit	303	212Fh	0
P0304	P1 - Último Error 5	0 a 65535	-	ro, 16bit	304	2130h	0
P0305	P2 - Último Error 1	0 a 65535	-	ro, 16bit	305	2131h	0
P0306	P2 - Último Error 2	0 a 65535	-	ro, 16bit	306	2132h	0
P0307	P2 - Último Error 3	0 a 65535	-	ro, 16bit	307	2133h	0
P0308	P2 - Último Error 4	0 a 65535	-	ro, 16bit	308	2134h	0
P0309	P2 - Último Error 5	0 a 65535	-	ro, 16bit	309	2135h	0
P0310	P3 - Último Error 1	0 a 65535	-	ro, 16bit	310	2136h	0
P0311	P3 - Último Error 2	0 a 65535	-	ro, 16bit	311	2137h	0
P0312	P3 - Último Error 3	0 a 65535	-	ro, 16bit	312	2138h	0
P0313	P3 - Último Error 4	0 a 65535	-	ro, 16bit	313	2139h	0
P0314	P3 - Último Error 5	0 a 65535	-	ro, 16bit	314	213Ah	0
P0315	P4 - Último Error 1	0 a 65535	-	ro, 16bit	315	213Bh	0
P0316	P4 - Último Error 2	0 a 65535	-	ro, 16bit	316	213Ch	0
P0317	P4 - Último Error 3	0 a 65535	-	ro, 16bit	317	213Dh	0
P0318	P4 - Último Error 4	0 a 65535	-	ro, 16bit	318	213Eh	0

						CANopen Index	Sub-Index
P0319	P4 - Último Error 5	0 a 65535	-	ro, 16bit	319	213Fh	0
P0320	P1 - Última Alarma 1	0 a 65535	-	ro, 16bit	320	2140h	0
P0321	P1 - Última Alarma 2	0 a 65535	-	ro, 16bit	321	2141h	0
P0322	P1 - Última Alarma 3	0 a 65535	-	ro, 16bit	322	2142h	0
P0323	P1 - Última Alarma 4	0 a 65535	-	ro, 16bit	323	2143h	0
P0324	P1 - Última Alarma 5	0 a 65535	-	ro, 16bit	324	2144h	0
P0325	P2 - Última Alarma 1	0 a 65535	-	ro, 16bit	325	2145h	0
P0326	P2 - Última Alarma 2	0 a 65535	-	ro, 16bit	326	2146h	0
P0327	P2 - Última Alarma 3	0 a 65535	-	ro, 16bit	327	2147h	0
P0328	P2 - Última Alarma 4	0 a 65535	-	ro, 16bit	328	2148h	0
P0329	P2 - Última Alarma 5	0 a 65535	-	ro, 16bit	329	2149h	0
P0330	P3 - Última Alarma 1	0 a 65535	-	ro, 16bit	330	214Ah	0
P0331	P3 - Última Alarma 2	0 a 65535	-	ro, 16bit	331	214Bh	0
P0332	P3 - Última Alarma 3	0 a 65535	-	ro, 16bit	332	214Ch	0
P0333	P3 - Última Alarma 4	0 a 65535	-	ro, 16bit	333	214Dh	0
P0334	P3 - Última Alarma 5	0 a 65535	-	ro, 16bit	334	214Eh	0
P0335	P4 - Última Alarma 1	0 a 65535	-	ro, 16bit	335	214Fh	0
P0336	P4 - Última Alarma 2	0 a 65535	-	ro, 16bit	336	2150h	0
P0337	P4 - Última Alarma 3	0 a 65535	-	ro, 16bit	337	2151h	0
P0338	P4 - Última Alarma 4	0 a 65535	-	ro, 16bit	338	2152h	0
P0339	P4 - Última Alarma 5	0 a 65535	-	ro, 16bit	339	2153h	0
S4 Estado\Comunicación							
P0030	CAN - Dirección	1 a 127	-	ro, 16bit	30	201Eh	0
P0031	CAN - Tasa de Comunicación	0 = 1 Mbit/s 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 = 25 Kbit/s	-	ro, enum	31	201Fh	0
P0033	CAN - Estado del Controlador	0 = Inactivo 1 = Reservado 2 = CAN Activo 3 = Warning 4 = Error Pasivo 5 = Bus Off	-	ro, enum	33	2021h	0
P0034	CAN - Telegramas CAN RX	0 a 65535	-	ro, 16bit	34	2022h	0
P0035	CAN - Telegramas CAN TX	0 a 65535	-	ro, 16bit	35	2023h	0
P0036	CAN - Contador Bus Off	0 a 65535	-	ro, 16bit	36	2024h	0
P0037	CAN - Telegramas Perdidos	0 a 65535	-	ro, 16bit	37	2025h	0
P0038	CAN - Estado de la Comunicación CANopen	0 = Inactivo 1 = Reservado	-	ro, enum	38	2026h	0

						CANopen Index	Sub-Index
P0039	CAN - Estado del Esclavo CANopen	2 = Comunic. Hab. 3 = Ctrl. Errors Hab. 4 = Error Guarding 5 = Error Heartbeat 0 = Inactivo 1 = Inicialización 2 = Detenido 3 = Operacional 4 = PreOperacional	-	ro, enum	39	2027h	0
C1 Configuración\Arranques							
P0400	P1 - Modo de Operación	0 = Arranque 1 = Transparente	0	rw, 16bit	400	2190h	0
P0401	P2 - Modo de Operación	0 = Arranque 1 = Transparente	0	rw, 16bit	401	2191h	0
P0402	P3 - Modo de Operación	0 = Arranque 1 = Transparente	0	rw, 16bit	402	2192h	0
P0403	P4 - Modo de Operación	0 = Arranque 1 = Transparente	0	rw, 16bit	403	2193h	0
P0404	P1 - Timeout Contactor	20 a 5000 ms	500 ms	rw, 16bit	404	2194h	0
P0405	P2 - Timeout Contactor	20 a 5000 ms	500 ms	rw, 16bit	405	2195h	0
P0406	P3 - Timeout Contactor	20 a 5000 ms	500 ms	rw, 16bit	406	2196h	0
P0407	P4 - Timeout Contactor	20 a 5000 ms	500 ms	rw, 16bit	407	2197h	0
P1000	Resetea Estándar de Fábrica	0 a 65535	0	rw, 16bit	1000	23E8h	0
C2 Configuración\Comunicación							
P0101	Acción para Falla en la Comunicación	0 = Sin Acción 1 = Apaga las Salidas	1	rw, enum	101	2065h	0
P0032	CAN - Reset de Bus Off	0 = Manual 1 = Automático	0	rw, enum	32	2020h	0
C3 Configuración\Contadores							
P0158	Guarda Contadores de Maniobras en la mem NV	FALSE a TRUE	FALSE	rw, bool	158	209Eh	0
P0160	Resetea Contador de Maniobras P1 C1	0 a 65535	0	rw, 16bit	160	20A0h	0
P0161	Resetea Contador de Maniobras P1 C2	0 a 65535	0	rw, 16bit	161	20A1h	0
P0162	Resetea Contador de Maniobras P2 C1	0 a 65535	0	rw, 16bit	162	20A2h	0
P0163	Resetea Contador de Maniobras P2 C2	0 a 65535	0	rw, 16bit	163	20A3h	0
P0164	Resetea Contador de Maniobras P3 C1	0 a 65535	0	rw, 16bit	164	20A4h	0
P0165	Resetea Contador de Maniobras P3 C2	0 a 65535	0	rw, 16bit	165	20A5h	0
P0166	Resetea Contador de Maniobras P4 C1	0 a 65535	0	rw, 16bit	166	20A6h	0
P0167	Resetea Contador de Maniobras P4 C2	0 a 65535	0	rw, 16bit	167	20A7h	0
C4 Configuración\Comandos							

						CANopen Index	Sub-Index
P1100	Comando de Arranque Directo	Bit 0 = Arranque 1 - directo Bit 1 = Arranque 2 - directo Bit 2 = Arranque 3 - directo Bit 3 = Arranque 4 - directo	0	rw, 16bit	1100	244Ch	0
P1105	Comando de Arranque Reverso	Bit 0 = Arranque 1 - reverso Bit 1 = Arranque 2 - reverso Bit 2 = Arranque 3 - reverso Bit 3 = Arranque 4 - reverso	0	rw, 16bit	1105	2451h	0
P1110	Comando de Parada	Bit 0 = Arranque 1 - apaga Bit 1 = Arranque 2 - apaga Bit 2 = Arranque 3 - apaga Bit 3 = Arranque 4 - apaga	0	rw, 16bit	1110	2456h	0
P1115	Comando de las Salidas Digitales	Bit 0 = DO1 Bit 1 = DO2 Bit 2 = DO3 Bit 3 = DO4 Bit 4 = DO5 Bit 5 = DO6 Bit 6 = DO7 Bit 7 = DO8	0	rw, 16bit	1115	245Bh	0

Tabla A.3

bool	Representa un bit en la que el valor 0 (cero) es falsa, el valor 1 (uno) es verdadera.
enum	Tipo enumerado (8 bits sin signo), contiene una lista de valores con la descripción de la función de cada elemento.
8bit	Entero de 8 bits sin signo, rango entre 0 y 255.
16bit	Entero de 16 bits sin signo, rango entre 0 y 65.535.
s16bit	Entero de 16 bits con signo, rango entre -32.768 y 32.767.
32bit	Entero de 32 bits sin signo, rango entre 0 y 4.294.967.295.



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