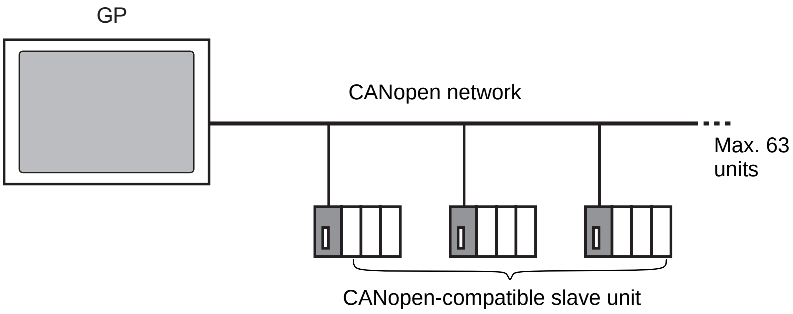
30.7 Controlling External I/O with CANopen

30.7.1 Summary

Model AGP-*****-CA1M supports the CANopen master driver. You can control remote external I/O by connecting a CANopen-compatible slave unit to the GP.



For the GP unit, an open network with maximum 63 units of connected CANopen-compatible slave unit can be constructed.

■ CAN Specification

- CANopen specification is defined by CiA and can be viewed on the CiA Web site. http://www.can-cia.org

NOTE
- For details on CANopen specification or its basic structure, refer to the CiA Web site.

■ CANopen Master Driver

- Uses DS301V4.02, DSP302V3.2, and DS405V2.0 profiles. DS301 is a profile used for the application layer and for communication. DSP302 is a framework for the CANopen Manager and programmable CANopen devices. DS405V2.0 is a profile for IEC61131-3 programmable devices.
- Supports 11 bit COB-ID (CAN2.0A). Does not support 29 bit COB-ID (CAN2.0B).
- Communication is carried out with a PDO packet as the unit.
- Does not support flying master*1.
- Network configuration is saved in a concise DCF file*2. This concise DCF file is transferred to the GP when transferring a project.

*1 Flying Master is a feature that allows the flying master to dynamically determine the master in a network when there are multiple devices that can used as the CANopen master.

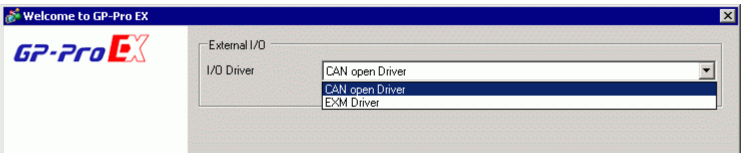
*2 Concise DCF is a device setting file in binary data format. The CANopen network is configured by downloading this data to the NMT master. NMT master is a feature that controls the actions of a slave node. Only one exists in any CANopen network; the node with this NMT master feature becomes the CANopen master.

30.7.2 Setting Procedure

1 In Model Settings, select AGP-*****-CA1M/LT.

NOTE

- When using LT models, select "CANopen Driver" for the I/O driver.

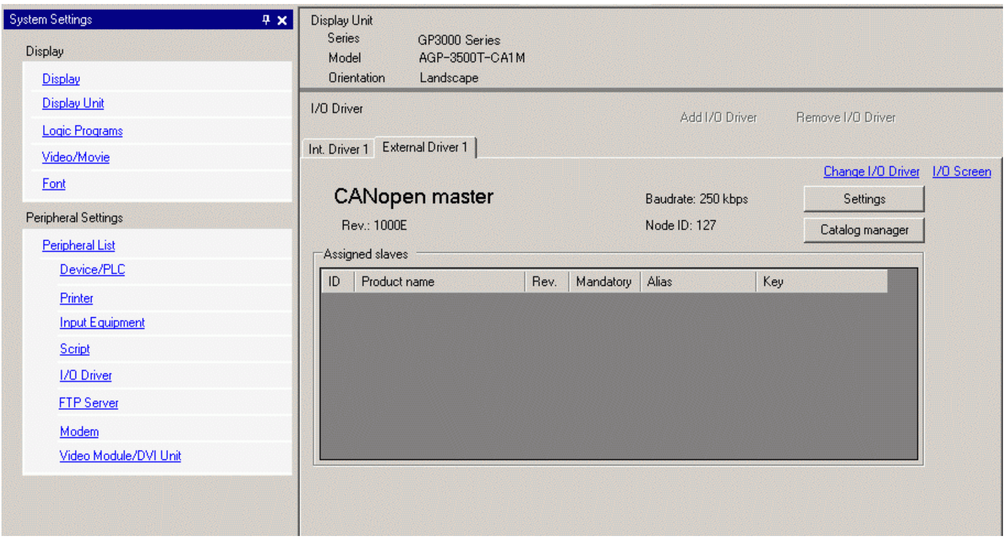


Welcome to GP-Pro EX

External I/O

I/O Driver: CAN open Driver

2 Open [I/O Driver] in System Settings.



Display Unit

Series: GP3000 Series

Model: AGP-3500T-CA1M

Orientation: Landscape

I/O Driver

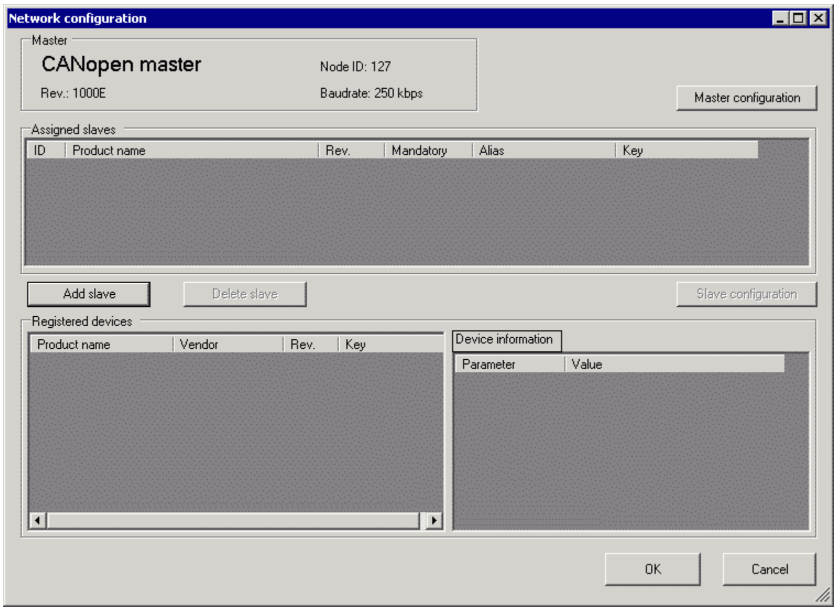
CANopen master

Rev.: 1000E

Baudrate: 250 kbps

Node ID: 127

3 Click [Settings] and the following dialog box appears.



Network configuration

Master

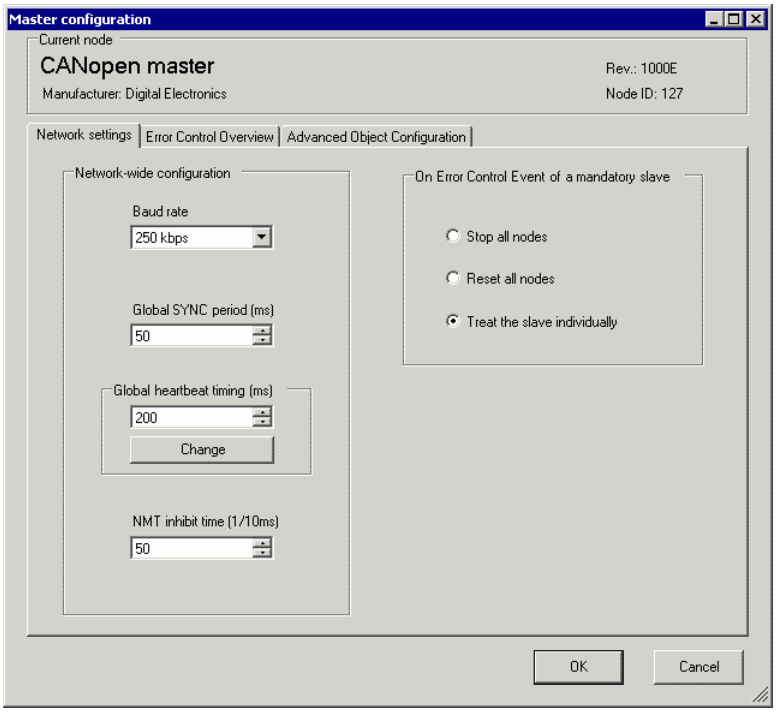
CANopen master

Rev.: 1000E

Node ID: 127

Baudrate: 250 kbps

4 Click [Master Configuration] and the following dialog box appears. Configure all of the CANopen network settings, such as Baud rate, SYNC sending period, and master object settings. Click [OK] to enable the settings and the dialog box closes.



Master configuration

Current node

CANopen master

Manufacturer: Digital Electronics

Rev.: 1000E

Node ID: 127

Network-wide configuration

Baud rate: 250 kbps

Global SYNC period (ms): 50

Global heartbeat timing (ms): 200

NMT inhibit time (1/10ms): 50

On Error Control Event of a mandatory slave

○ Stop all nodes

○ Reset all nodes

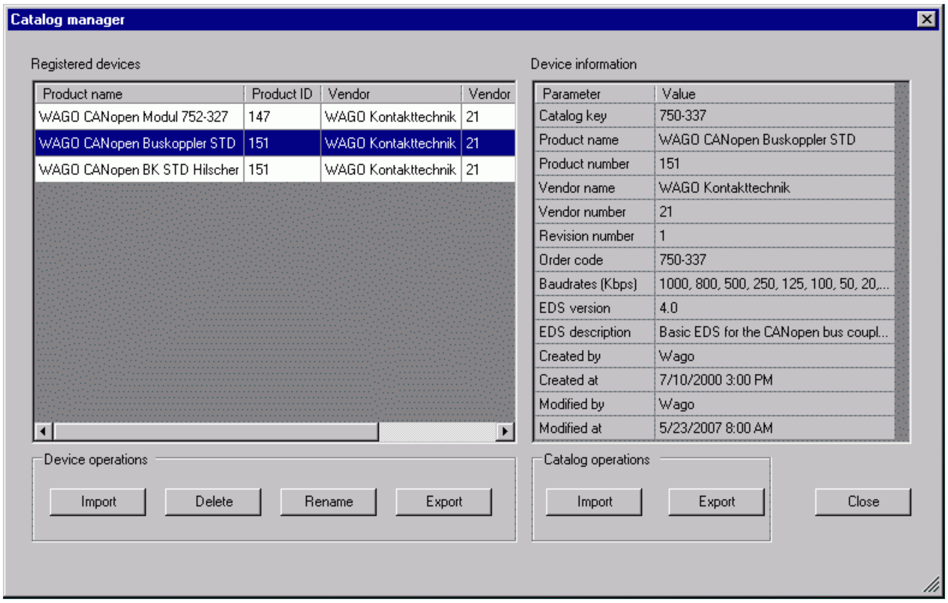
● Treat the slave individually

NOTE

- Set the slave baud rate on the slave itself.

5 Click [OK] in the [Network Configuration] dialog box.

6 Next, add a slave to the CANopen network. Click [Catalog Manager] and the following dialog box appears.
Click [Import] in the [Device operation] section and specify the EDS file of the CANopen-compatible slave unit. Click [Close].



Catalog manager

Registered devices

Product name	Product ID	Vendor	Vendor
WAGO CANopen Modul 752-327	147	WAGO Kontakttechnik	21
WAGO CANopen Buskoppler STD	151	WAGO Kontakttechnik	21
WAGO CANopen BK STD Hilscher	151	WAGO Kontakttechnik	21

Device information

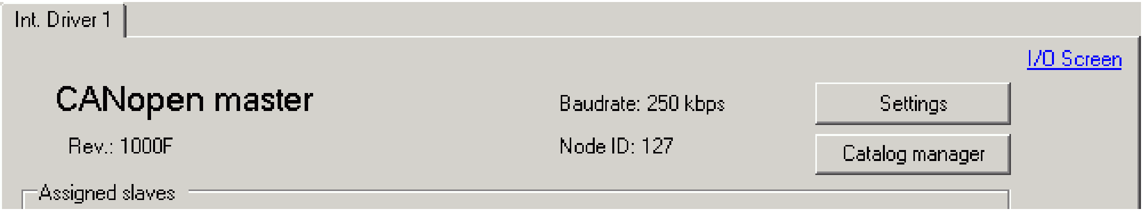
Parameter	Value
Catalog key	750-337
Product name	WAGO CANopen Buskoppler STD
Product number	151
Vendor name	WAGO Kontakttechnik
Vendor number	21
Revision number	1
Order code	750-337
Baudrates (Kbps)	1000, 800, 500, 250, 125, 100, 50, 20,...
EDS version	4.0
EDS description	Basic EDS for the CANopen bus coupl...
Created by	Wago
Created at	7/10/2000 3:00 PM
Modified by	Wago
Modified at	5/23/2007 8:00 AM

NOTE

- EDS files and connection methods for the models that have been confirmed to function by us are listed on the Pro-face support site "Otasuke Pro!" (http://www.pro-face.com/otasuke/).
- [Import/Export] in the [Catalog] section is necessary to open the created project file or transfer it to other PCs.

☞ "30.7.6 [I/O Driver] Settings Guide" (page 30-170)

7 Click [Settings].



Int. Driver 1

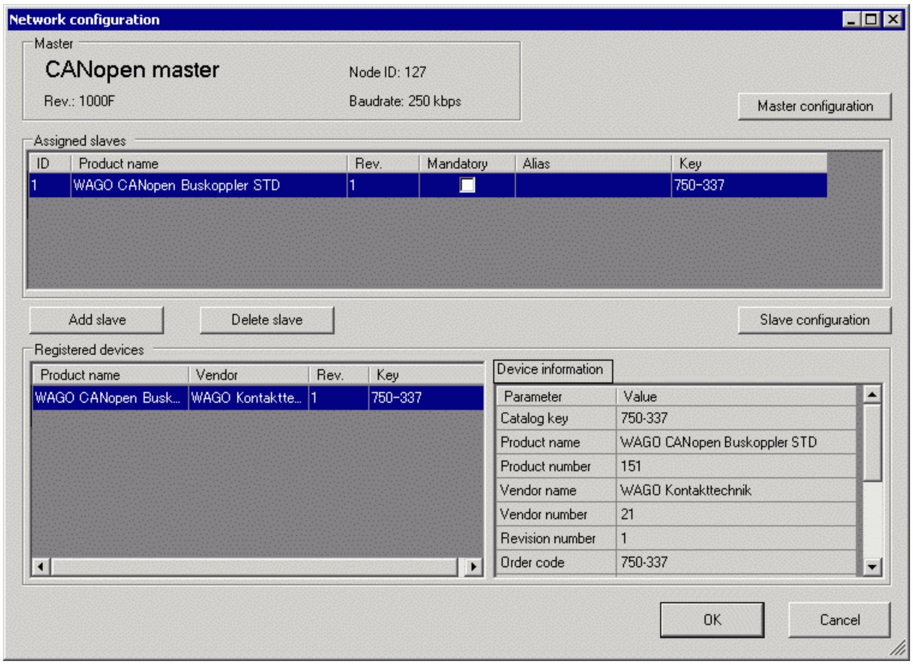
CANopen master

Rev.: 1000F

Baudrate: 250 kbps

Node ID: 127

8 Select the above mentioned EDS file in [Registered devices], and click [Add Slave].



Master

CANopen master

Node ID: 127

Rev.: 1000F

Baudrate: 250 kbps

Assigned slaves

ID	Product name	Rev.	Mandatory	Alias	Key
1	WAGO CANopen Buskoppler STD	1	□		750-337

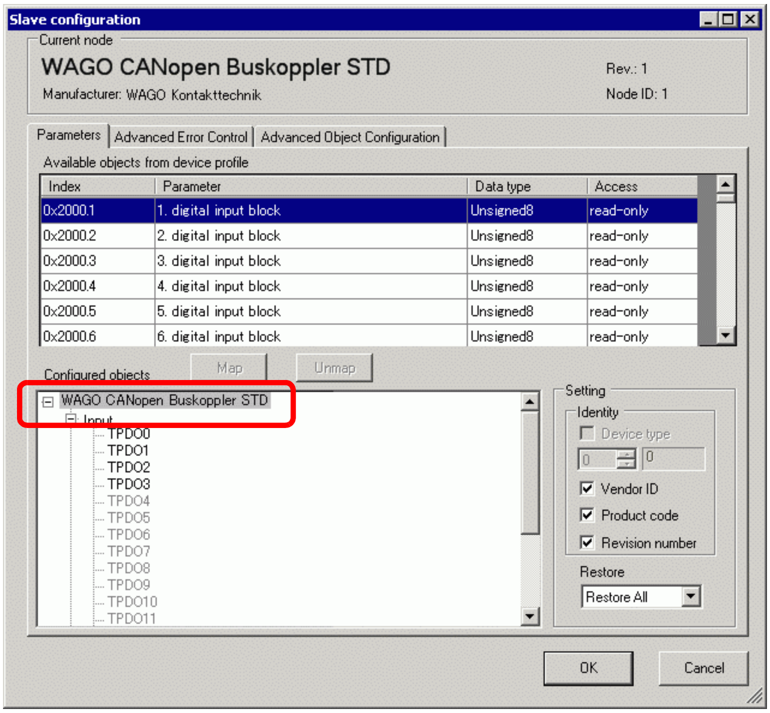
Registered devices

Product name	Vendor	Rev.	Key
WAGO CANopen Busk...	WAGO Kontaktte...	1	750-337

Device information

Parameter	Value
Catalog key	750-337
Product name	WAGO CANopen Buskoppler STD
Product number	151
Vendor name	WAGO Kontakttechnik
Vendor number	21
Revision number	1
Order code	750-337

9 Click [Slave Configuration] with the added slave unit selected and the following dialog box appears. According to the function you want to use, configure the communication parameter settings and set actions and values of objects to be used. Click [OK] to enable the settings and the dialog box closes.



Current node

WAGO CANopen Buskoppler STD

Rev.: 1

Manufacturer: WAGO Kontakttechnik

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x2000.1	1. digital input block	Unsigned8	read-only
0x2000.2	2. digital input block	Unsigned8	read-only
0x2000.3	3. digital input block	Unsigned8	read-only
0x2000.4	4. digital input block	Unsigned8	read-only
0x2000.5	5. digital input block	Unsigned8	read-only
0x2000.6	6. digital input block	Unsigned8	read-only

Configured objects

```
⊟ WAGO CANopen Buskoppler STD
  ⊟ Input
      TPDO0
      TPDO1
      TPDO2
      TPDO3
      TPDO4
      TPDO5
      TPDO6
      TPDO7
      TPDO8
      TPDO9
      TPDO10
      TPDO11
```

Setting

Identity

□ Device type

0 0

☑ Vendor ID

☑ Product code

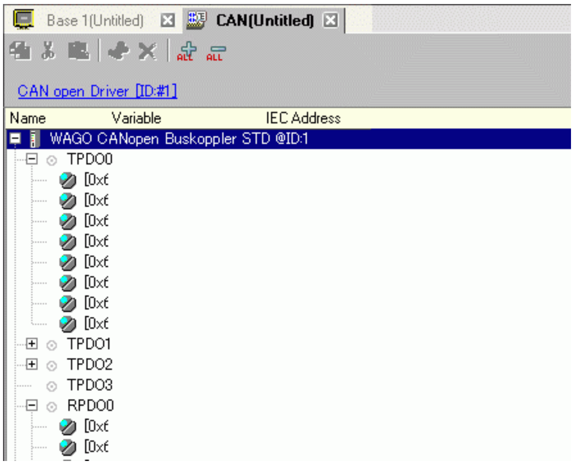
☑ Revision number

Restore

Restore All

10 Click [I/O Screen] in the [I/O Driver Settings] screen or select [I/O Screen] in the [Screen List] window on Work Space to allocate a variable to each of the mapped objects. For how to assign variables, refer to the following.

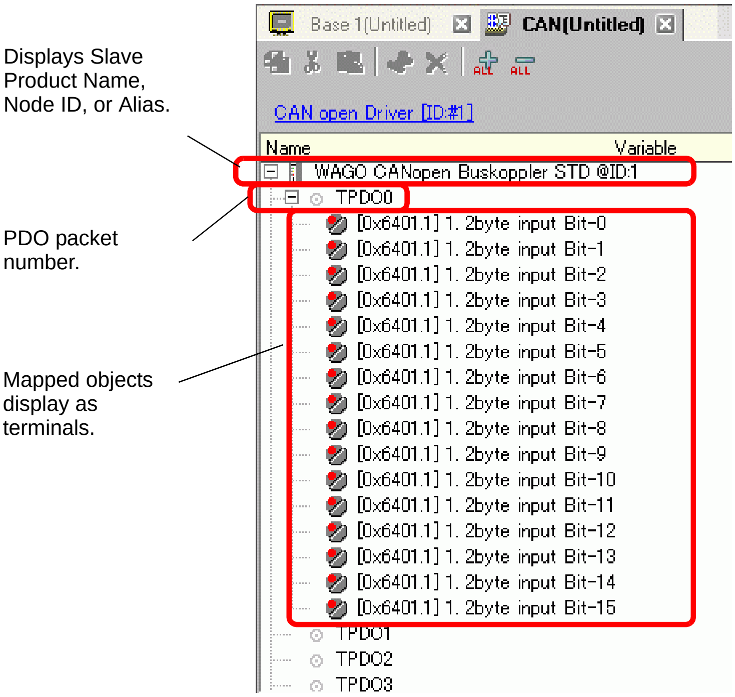
☞ "30.7.3 Allocating I/O" (page 30-155)



11 Create a Logic Screen and a Base Screen to access the allocated variables and transfer them to the GP.

30.7.3 Allocating I/O

Objects set in [I/O Driver Settings] are reflected on the I/O Screen as terminals (I/O terminals). By allocating variables to terminals, I/O can be controlled.



Displays Slave Product Name, Node ID, or Alias.

PDO packet number.

Mapped objects display as terminals.

Base 1(Untitled) CAN(Untitled)

CAN open Driver [ID:#1]

Name Variable

- WAGO CANopen Buskoppler STD @ID:1
  - TPDO0
    - [0x6401.1] 1. 2byte input Bit-0
    - [0x6401.1] 1. 2byte input Bit-1
    - [0x6401.1] 1. 2byte input Bit-2
    - [0x6401.1] 1. 2byte input Bit-3
    - [0x6401.1] 1. 2byte input Bit-4
    - [0x6401.1] 1. 2byte input Bit-5
    - [0x6401.1] 1. 2byte input Bit-6
    - [0x6401.1] 1. 2byte input Bit-7
    - [0x6401.1] 1. 2byte input Bit-8
    - [0x6401.1] 1. 2byte input Bit-9
    - [0x6401.1] 1. 2byte input Bit-10
    - [0x6401.1] 1. 2byte input Bit-11
    - [0x6401.1] 1. 2byte input Bit-12
    - [0x6401.1] 1. 2byte input Bit-13
    - [0x6401.1] 1. 2byte input Bit-14
    - [0x6401.1] 1. 2byte input Bit-15
  - TPDO1
  - TPDO2
  - TPDO3

NOTE

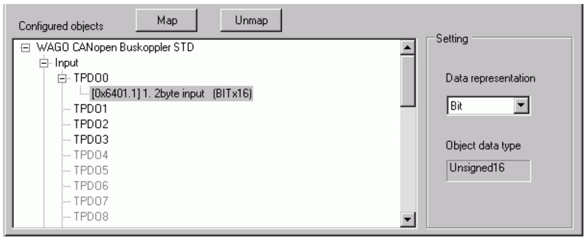
- When mapping TPDO4 or above and RPDO4 or above, the total slave settings are limited to 64.

■ I/O Screen Display

The terminal display on the I/O Screen differs depending on the type (such as, Bit, Byte, Word, Dword) of each object specified in the [Slave Configuration] dialog box in [I/O Driver Settings].

Example 1: Setting the Unsigned16 object [1. 2byte input (BITx16)] to the [Bit] display.

I/O Driver Settings

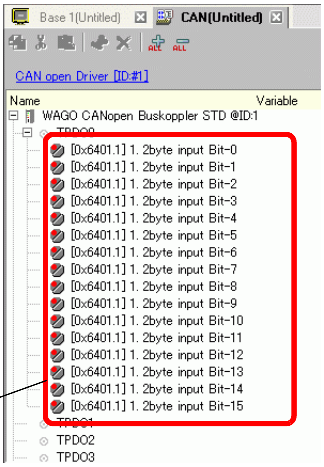


Configured objects　Map　Unmap

- WAGO CANopen Buskoppler STD
  - Input
    - TPDO0
      - [0x6401.1] 1. 2byte input (BITx16)
    - TPDO1
    - TPDO2
    - TPDO3
    - TPDO4
    - TPDO5
    - TPDO6
    - TPDO7
    - TPDO8

Setting
Data representation: Bit
Object data type: Unsigned16

I/O Screen



Base 1(Untitled)　CAN(Untitled)

CAN open Driver [ID:#1]

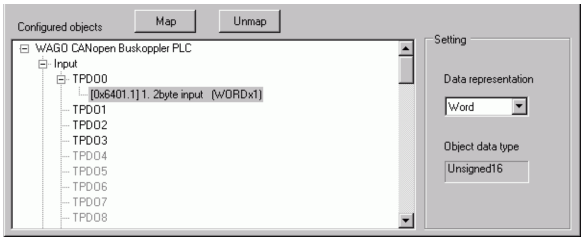
Name　Variable

- WAGO CANopen Buskoppler STD @ID:1
  - TPDO0
    - [0x6401.1] 1. 2byte input Bit-0
    - [0x6401.1] 1. 2byte input Bit-1
    - [0x6401.1] 1. 2byte input Bit-2
    - [0x6401.1] 1. 2byte input Bit-3
    - [0x6401.1] 1. 2byte input Bit-4
    - [0x6401.1] 1. 2byte input Bit-5
    - [0x6401.1] 1. 2byte input Bit-6
    - [0x6401.1] 1. 2byte input Bit-7
    - [0x6401.1] 1. 2byte input Bit-8
    - [0x6401.1] 1. 2byte input Bit-9
    - [0x6401.1] 1. 2byte input Bit-10
    - [0x6401.1] 1. 2byte input Bit-11
    - [0x6401.1] 1. 2byte input Bit-12
    - [0x6401.1] 1. 2byte input Bit-13
    - [0x6401.1] 1. 2byte input Bit-14
    - [0x6401.1] 1. 2byte input Bit-15
  - TPDO1
  - TPDO2
  - TPDO3

An Unsigned16 object displays as a 16 bit terminal.

Example 2: Setting the Unsigned16 object [1. 2byte input (BITx16)] to the [Word] display.

I/O Driver Settings

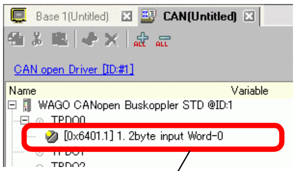


Configured objects　Map　Unmap

- WAGO CANopen Buskoppler PLC
  - Input
    - TPDO0
      - [0x6401.1] 1. 2byte input (WORDx1)
    - TPDO1
    - TPDO2
    - TPDO3
    - TPDO4
    - TPDO5
    - TPDO6
    - TPDO7
    - TPDO8

Setting
Data representation: Word
Object data type: Unsigned16

I/O Screen



Base 1(Untitled)　CAN(Untitled)

CAN open Driver [ID:#1]

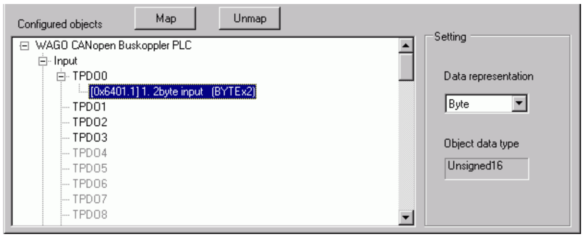
Name　Variable

- WAGO CANopen Buskoppler STD @ID:1
  - TPDO0
    - [0x6401.1] 1. 2byte input Word-0
  - TPDO1
  - TPDO2

Unsigned16 displays as a 1 word terminal.

Example 3: Setting the Unsigned16 object [1. 2byte input (BITx16)] to the [Byte] display.

I/O Driver Settings

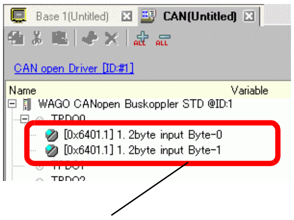


Configured objects　Map　Unmap

- WAGO CANopen Buskoppler PLC
  - Input
    - TPDO0
      - [0x6401.1] 1. 2byte input (BYTEx2)
    - TPDO1
    - TPDO2
    - TPDO3
    - TPDO4
    - TPDO5
    - TPDO6
    - TPDO7
    - TPDO8

Setting
Data representation: Byte
Object data type: Unsigned16

I/O Screen



Base 1(Untitled)　CAN(Untitled)

CAN open Driver [ID:#1]

Name　Variable

- WAGO CANopen Buskoppler STD @ID:1
  - TPDO0
    - [0x6401.1] 1. 2byte input Byte-0
    - [0x6401.1] 1. 2byte input Byte-1
  - TPDO1
  - TPDO2

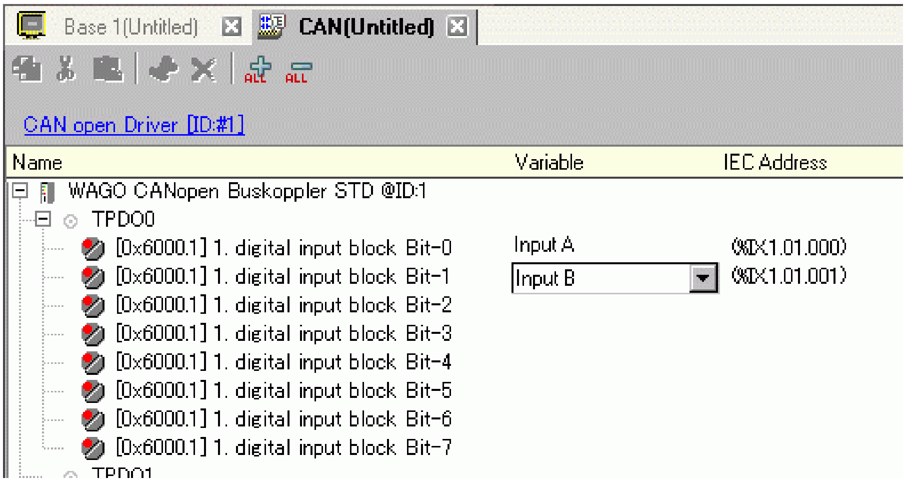
Unsigned16 displays as a 2 byte terminal.

NOTE

- In [1. 2byte input (BITx16)], Unsigned16 bit data the lower bytes are allocated to the "[Byte-0] terminal and higher bytes to the [Byte-1] terminal. The remaining bits of the assigned variables cannot be used.

■ Mapping Variables

Mapping a variable per terminal of objects. To enter a variable, double-click the [Variable] column of the terminal you wish to allocate.



CAN open Driver [ID:#1]

Name	Variable	IEC Address
WAGO CANopen Buskoppler STD @ID:1		
TPDO0		
[0x6000.1] 1. digital input block Bit-0	Input A	(%IX1.01.000)
[0x6000.1] 1. digital input block Bit-1	Input B	(%IX1.01.001)
[0x6000.1] 1. digital input block Bit-2		
[0x6000.1] 1. digital input block Bit-3		
[0x6000.1] 1. digital input block Bit-4		
[0x6000.1] 1. digital input block Bit-5		
[0x6000.1] 1. digital input block Bit-6		
[0x6000.1] 1. digital input block Bit-7		
TPDO1		

[IEC Address] appears automatically after entering a variable.

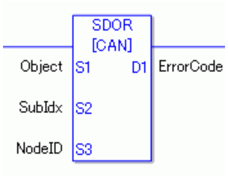
Data Type	Input	Output
Bit	IX	QX
Byte	IB	QB
Word	IW	QW
Dword	ID	QD

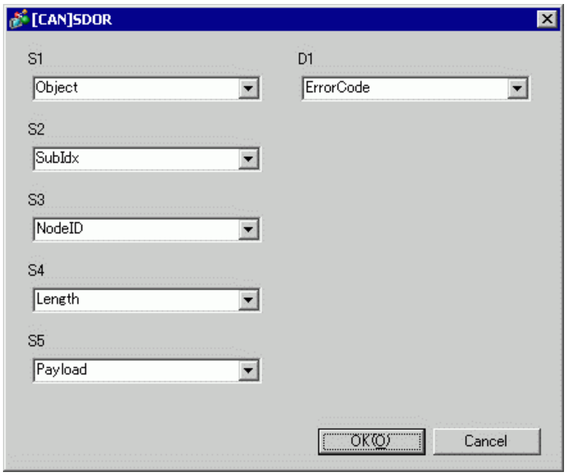
30.7.4 Using I/O Driver Instructions

You can change set values of objects and check the values while the program is running if you use I/O Driver Instructions on Logic Screen.

◆ SDOR

Data is read from the object dictionary of the specified node.
Double-click the inserted instruction and the following dialog box appears.





[CAN]SDOR

S1: Object
S2: SubIdx
S3: NodeID
S4: Length
S5: Payload
D1: ErrorCode

NOTE

- Do not run two or more SDO instructions (SDOR, SDOW) at the same time. You can only run one SDO instruction at a time.
- If the network state is not READY (such as RESET state), SDOR or SDOW instructions are not accepted and error code 146 displays.

Operand Settings

S1: Specify index of the object.
S2: Specify sub index of the object.
S3: Specify node ID.
S4: Specify length (byte number) of SDO access.
S5: Specify where to store read data and abort code.

For the lower 16 bits, if you specify an array element in the system variable #L_IOMasterDrv[0] to [255], that becomes the start address from which the number of bytes specified in parameter S4 is written in the variable.
For the higher 16 bits, if you specify an array element in the system variable

#L_IOMasterDrv[0] to [255] with the most significant bit ON, the abort code is written in the specified variable.

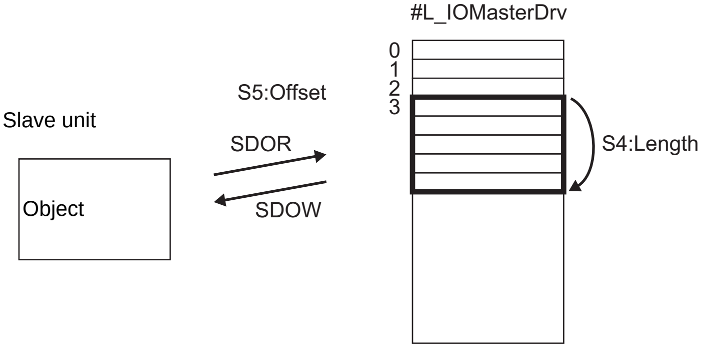
High word			Low word	
31	**30**	**16**	**15**	**0**
Abort code is stored in #L_IOMasterDrv[ ]. 0:Abort code is not stored in #L_IOMasterDrv[ ]. However, if the error includes the abort code, the abort code is stored in D1 (error code). 1:Abort code is stored in #L_IOMasterDrv[ ]. The following error code is stored in D1 (error code).	Offset of #L_IOMasterDrv[ ] to store abort code. This is ignored when bit 31 is 0.		Offset of #L_IOMasterDrv[ ] to store data.	

D1: Specify the variable where error codes are stored if an instruction does not act as expected.

Error Code

0: Normal
1: Error in operand number
111: Error in set up value for operand S4 or S5
120: Error in SDO communication or error in set up value for operand S1, S2, or S3
123: Timeout error
140: SDO protocol error
141: SDO send overflow
142: SDO master setting error
143: SDO access error
144: SDO receive timeout
145: SDO operand error
146: SDO master status error
147: SDO master status stopped
148: SDO abort error

Example: S4=20, S5=3

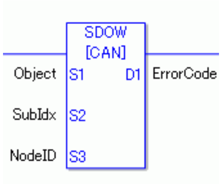


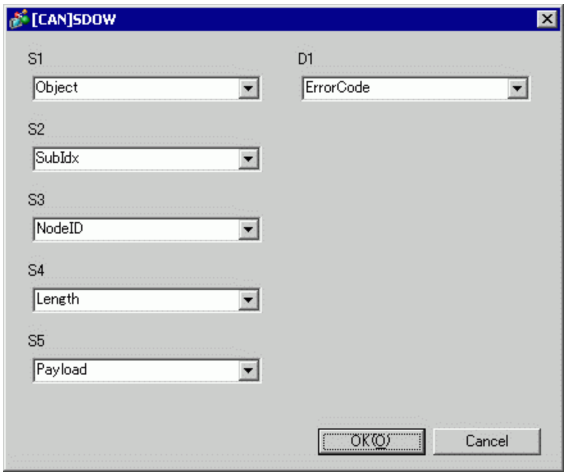
Abort Code

0503 0000h: Toggle bit was not changed.
0504 0000h: SDO protocol timeout
0504 0001h: Invalid or unknown Client/Server command
0504 0002h: Invalid block size (block mode)
0504 0003h: Invalid sequence number (block mode)
0504 0004h: CRC error (block mode)
0504 0005h: insufficient memory
0601 0000h: access to unsupported objects
0601 0001h: Read access to write-only objects
0601 0002h: Write access to read-only objects
0602 0000h: Object does not exist in the object dictionary
0604 0041h: Cannot map PDO objects
0604 0042h: The number and length of objects to map exceed the PDO length
0604 0043h: Incompatibility of general parameters
0604 0047h: General internal incompatibility of the device
0606 0000h: Access failure due to a hardware error
0607 0010h: Data type mismatch. Length of the service parameter does not match.
0607 0012h: Data type mismatch. Length of the service parameter is too long.
0607 0013h: Data type mismatch. Length of the service parameter is too short.
0609 0011h: Sub index does not exist
0609 0030h: Parameter value out of range (for write access)
0609 0031h: Written parameter value is too large.
0609 0032h: Written parameter value is too small.
0609 0036h: Max. value is smaller than Min. value
060A 0023h: Resource cannot be used
0800 0000h: General error
0800 0020h: Data cannot be transferred or stored by the application
0800 0021h: Data cannot be transferred or stored by the application due to local control
0800 0022h: Data cannot be transferred or stored by the application in the current device state.
0800 0023h: Dynamic creation of the object dictionary failed or the object dictionary does not exist
0800 0024h: Valid data does not exist

◆ SDOW

Data is written in the object dictionary of the specified node.
Double-click the inserted instruction and the following dialog box appears.





S1: Object

D1: ErrorCode

S2: SubIdx

S3: NodeID

S4: Length

S5: Payload

NOTE

- Do not run two or more SDO instructions (SDOR, SDOW) at the same time. You can only run one SDO instruction at a time.
- If the network state is not READY (such as RESET state), SDOR or SDOW instructions are not accepted and error code 146 displays.

Operand Settings

S1: Specify index of the object.
S2: Specify sub index of the object.
S3: Specify node ID.
S4: Specify length (byte number) of SDO access.
S5: Specify where to store data to be written and abort code.

For the lower 16 bits, if you specify an array element in the system variable #L_IOMasterDrv[0] to [255], that becomes the start address from which the data with the number of bytes specified in parameter S4 is written in the specified object.
For the higher 16 bits, if you specify an array element in the system variable #L_IOMasterDrv[0] to [255] with the most significant bit ON, the abort code is written in the specified variable.

NOTE

- For details on the abort code, refer to the "SDOR Command" section.

D1:Specify the variable where error codes are stored if an instruction does not act as expected.

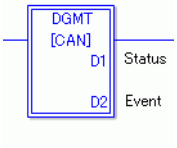
Error Code

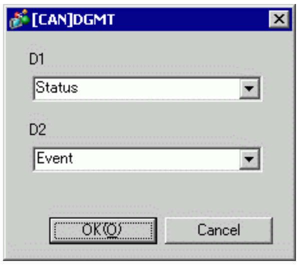
0:	Normal
1:	Error in operand number
111:	Error in set up value for operand S4 or S5
120:	Error in SDO communication or error in set up value for operand S1, S2, or S3
123:	Timeout error
140:	SDO protocol error
141:	SDO send overflow
142:	SDO master setting error
143:	SDO access error
144:	SDO receive timeout
145:	SDO operand error
146:	SDO master status error
147:	SDO master status stopped
148:	SDO abort error

◆ DGMT

State on the master is read.

Double-click the inserted instruction and the following dialog box appears.




[CAN]DGMT

D1: Status

D2: Event

Operand Settings

D1: Specify the variable to store status.

D2: Specify the variable to store information on an event.

- Status

31 … 16	15 … 8	7 … 0
Unused (0 fixed)	Communication status	Master status

Bit	Description		Details
0 to 7	0x00	Initializing	In the process of initializing.
	0x01	Reset request	Change to reset.
	0x40	Reset the object	Resetting
	0x60	Slave checking	Check slave mapping
	0x61	Resetting network	Reset all the nodes
	0x62	Network standby	Waits for a specified amount of time while the communication command can be reset.
	0x64	Initializing each slave	Initialize each slave in the network.
	0x70	Module defect	A deficiency exists for the module.
	0x8x[*1]	Clearing	Scan the network. The state in which firmware can be started.
	0x90	Fatal error	Fatal error occurred in the network. Firmware will be reset.
	0xAx [*1]	Operating	The network is operating
	0xCx [*1]	Stopped	The network is stopped.
	0xEx [*1]	Preparing for operation	The network is in pre-operating state.
8	Queue overrun (RXLP)		Overrun of receiving queue with low priority[*2] has occurred.
9	Controller overrun		CAN controller overrun has occurred
10	Controller bus off		CAN controller bus off has occurred
11	Controller error occurred		Error has occurred in CAN controller
12	Controller error recovered		CAN controller has recovered from error state
13	Queue overrun (TXLP)		Overrun of sending queue with low priority [*2] has occurred.
14	Queue overrun (RXHP)		Overrun of receiving queue of high priority[*3] has occurred.
15	Queue overrun (TXHP)		Overrun of sending queue with high priority [*3] has occurred.

*1 The following states are shown according to the state of lower 4 bits.

Bit 0: Error bit of the optional slave or the slave that is not set.
0: No error
1: Error in 1 or more optional slaves or the slave that is not set.

Bit 1: Error bit of mandatory slave
0: No error
1: Error in 1 or more mandatory slaves

Bit 2: Common action bit
0: No active slaves
1: At least one active slave

Bit 3: Action bit of CANopen master module
0: Not active
1: Active

*2 Queue with low priority is used for heartbeat, node guard and SOD transfer.

*3 Queue with high priority is for messages of TPDO, NMT command, SYNC and EMCY.

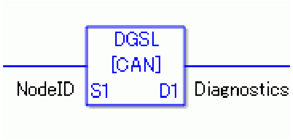
- Event

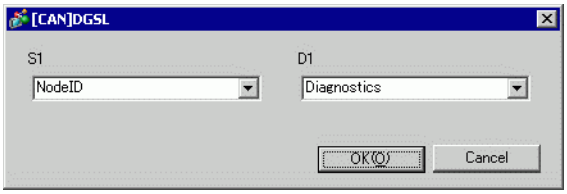
31 16	15 0
Unused (0 fixed)	Event information

Bit	Description
0	Network communication error
1	Node ID error
2	Error controlling event of mandatory slave
3	Identification error of mandatory slave or incomplete concise DCF
4	Identification error of optional slave
5, 6	Reserved
7	Set if there are changes to bits in the bit list
8	Requested feature is unsupported
9	Data byte number of received RxPDO is too small
10	Incomplete concise DCF
11	Overrun of application specific SDO queue
12	Reserved
13	Master alone
14	Change the state of the network with NMT command
15	Change the state of the slave with the NMT command

◆ DGSL

State on the slave is read.





Operand Settings

S1: Specify node ID.

D1: Specify the variable to store slave information.

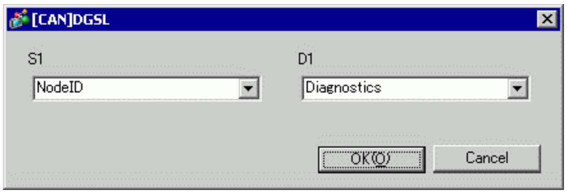
- Slave information

Bit	Description	Details
0	Mapping	0: Slave is not assigned to master. 1: Slave is assigned to master.
1	Configuration	Not set as slave. Startup is disabled. 1: Set as slave. Startup enabled.
2	Defect	Configured Slave Unconfigured Slave
3	Emergency message (EMCY)	Slave has not sent the emergency message Slave has sent the emergency message.
4	Operating	Slave is not operating. 1: Slave is operating.
5	Stopped	Slave is not stopped Slave is stopped
6	Preparing for operation	Slave is not preparing for operation. Slave is preparing for operation
7	Inconsistent Concise DCF for one or more slaves	Normal concise DCF file 1: Incorrect concise DCF file
8	Concise DCF mismatch for one or more slaves	Objects of DCF and slave are matched. 1: Objects of DCF and slave are mismatched.
9	Identity error	0: Slave information of the slave is normal. 1: Slave information of the slave has an error.
10 to 15	Reserved	—

◆ Setting Method

1 Open Logic Screen (MAIN or SUB) and right-click the rung to select [Insert Instructions (I)]. Click the icon to select [9. I/O Driver Instructions], point to [CAN], and click [DGSL] (instruction to read the state on slave side).

2 Mapping variables to operands. Double-click the inserted instruction.
The following dialog box appears. Map variables and click [OK].



[CAN]DGSL

S1: NodeID

D1: Diagnostics

30.7.5 PDO mapping on the slave and method of setting objects

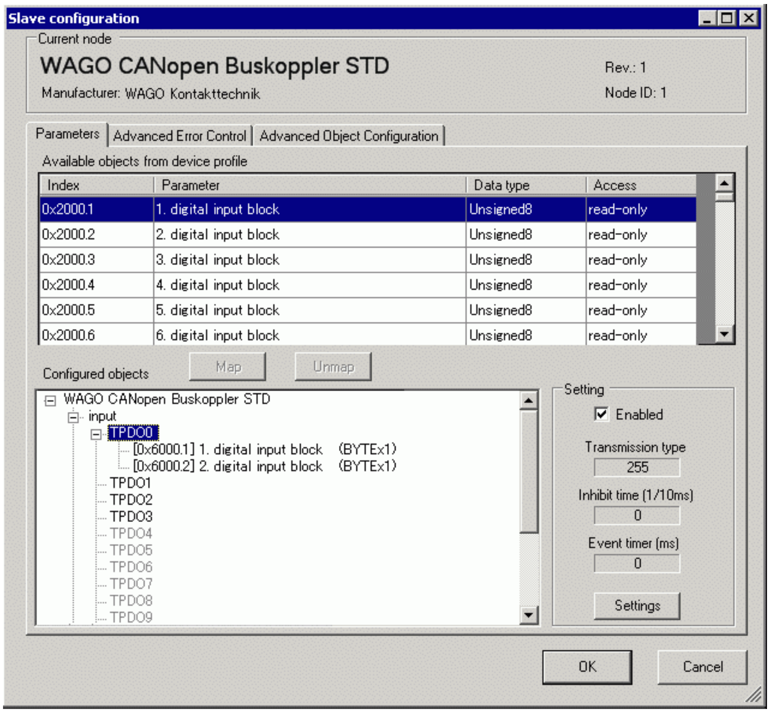
PDO mapping on the slave and the method of setting objects will be explained below.

Settings in GP-Pro EX

1 Click [Settings] from System window [I/O Driver Settings] and add a slave unit to the network.

☞ "30.7.2 Setting Procedure" (page 30-150)

2 Click [Slave Configuration] with the added slave unit selected and the following dialog box appears. In the [Parameters] tab, expand the [Configured objects] tree to open [TPDO0].



Slave configuration

Current node

WAGO CANopen Buskoppler STD

Manufacturer: WAGO Kontakttechnik

Rev.: 1

Node ID: 1

Parameters | Advanced Error Control | Advanced Object Configuration

Available objects from device profile

Index	Parameter	Data type	Access
0x2000.1	1. digital input block	Unsigned8	read-only
0x2000.2	2. digital input block	Unsigned8	read-only
0x2000.3	3. digital input block	Unsigned8	read-only
0x2000.4	4. digital input block	Unsigned8	read-only
0x2000.5	5. digital input block	Unsigned8	read-only
0x2000.6	6. digital input block	Unsigned8	read-only

Configured objects Map Unmap

```
⊟ WAGO CANopen Buskoppler STD
  ⊟ input
    ⊟ TPDO0
        [0x6000.1] 1. digital input block   (BYTEx1)
        [0x6000.2] 2. digital input block   (BYTEx1)
      TPDO1
      TPDO2
      TPDO3
      TPDO4
      TPDO5
      TPDO6
      TPDO7
      TPDO8
      TPDO9
```

Setting

☑ Enabled

Transmission type: 255

Inhibit time (1/10ms): 0

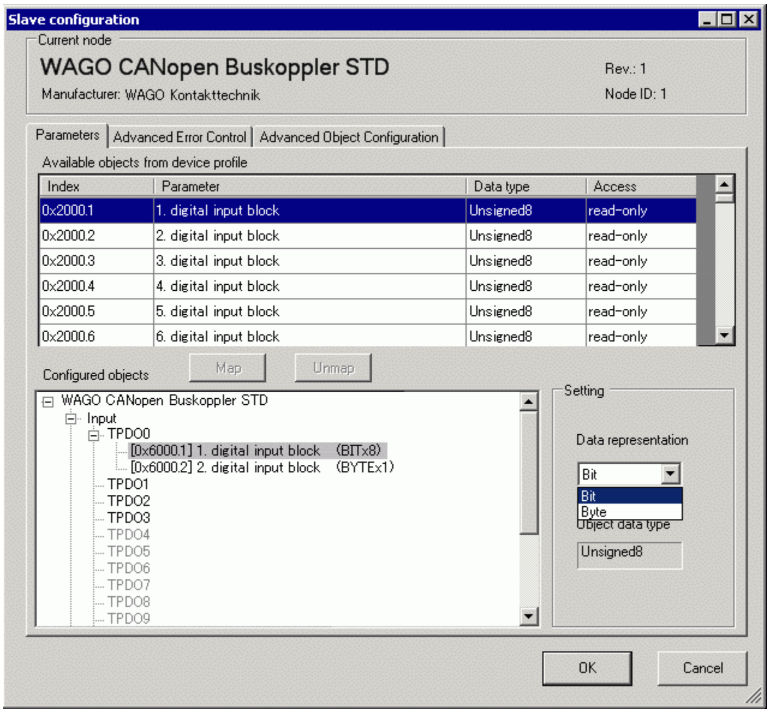
Event timer (ms): 0

Settings

OK Cancel

3 Check whether [0x6000.1] and [0x6000.2] have been allocated.

4 To change the data type of the object in TPDO, select the object and then select the [Data representation] in [Settings]. (For example, Bit)



Current node

WAGO CANopen Buskoppler STD

Manufacturer: WAGO Kontakttechnik

Rev.: 1

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x2000.1	1. digital input block	Unsigned8	read-only
0x2000.2	2. digital input block	Unsigned8	read-only
0x2000.3	3. digital input block	Unsigned8	read-only
0x2000.4	4. digital input block	Unsigned8	read-only
0x2000.5	5. digital input block	Unsigned8	read-only
0x2000.6	6. digital input block	Unsigned8	read-only

Configured objects

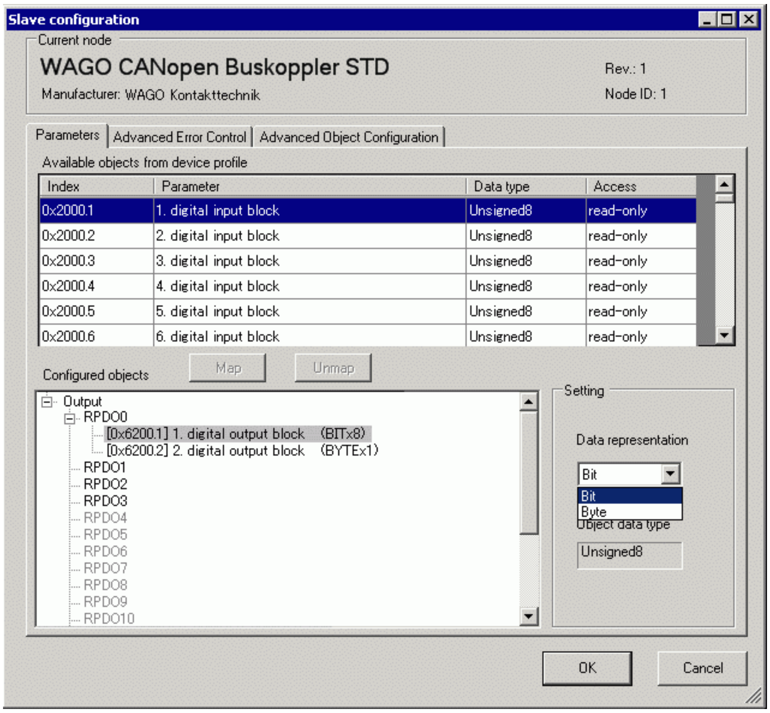
```
WAGO CANopen Buskoppler STD
  Input
    TPDO0
      [0x6000.1] 1. digital input block   (BITx8)
      [0x6000.2] 2. digital input block   (BYTEx1)
    TPDO1
    TPDO2
    TPDO3
    TPDO4
    TPDO5
    TPDO6
    TPDO7
    TPDO8
    TPDO9
```

Setting

Data representation: Bit (Bit / Byte)

Object data type: Unsigned8

5 Similarly, open RPDO0 and check whether [0x6200.1] and [0x6200.2] have been allocated. To change the data type of the object in RPDO, select the object and then and select [Data representation] in [Settings]. (For example, Bit)



Current node

WAGO CANopen Buskoppler STD

Manufacturer: WAGO Kontakttechnik

Rev.: 1

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x2000.1	1. digital input block	Unsigned8	read-only
0x2000.2	2. digital input block	Unsigned8	read-only
0x2000.3	3. digital input block	Unsigned8	read-only
0x2000.4	4. digital input block	Unsigned8	read-only
0x2000.5	5. digital input block	Unsigned8	read-only
0x2000.6	6. digital input block	Unsigned8	read-only

Configured objects

```
Output
  RPDO0
    [0x6200.1] 1. digital output block   (BITx8)
    [0x6200.2] 2. digital output block   (BYTEx1)
  RPDO1
  RPDO2
  RPDO3
  RPDO4
  RPDO5
  RPDO6
  RPDO7
  RPDO8
  RPDO9
  RPDO10
```

Setting

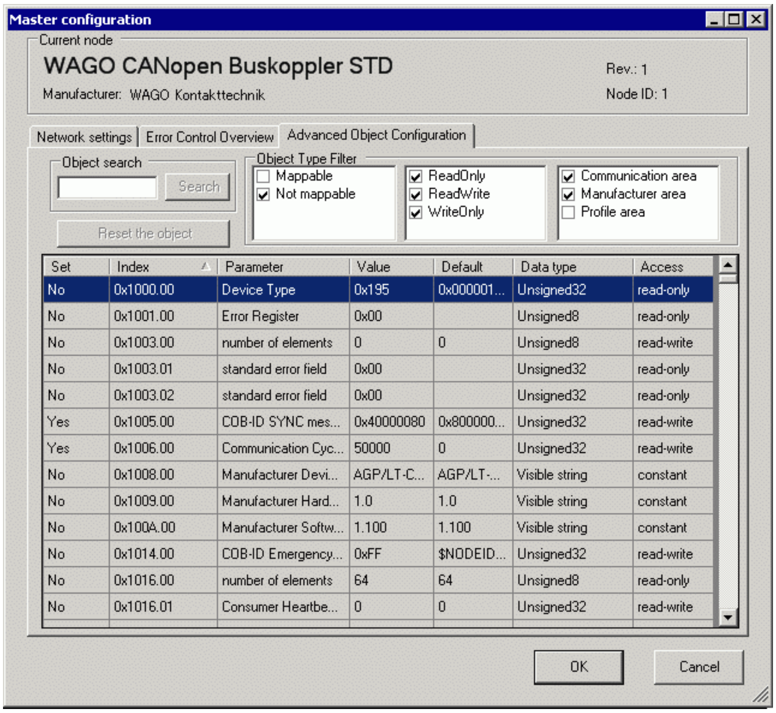
Data representation: Bit (Bit / Byte)

Object data type: Unsigned8

NOTE

- Objects related to input data from CANopen slave to GP (CANopen master) are mapped to TPDO, and objects related to output data from GP (CANopen master) to CANopen slave are mapped to RPDO.

6 Open the [Advanced Object Configuration] tab and configure detailed settings of the object.



WAGO CANopen Buskoppler STD

Rev.: 1

Manufacturer: WAGO Kontakttechnik

Node ID: 1

□ Mappable
☑ Not mappable

☑ ReadOnly
☑ ReadWrite
☑ WriteOnly

☑ Communication area
☑ Manufacturer area
□ Profile area

Set	Index	Parameter	Value	Default	Data type	Access
No	0x1000.00	Device Type	0x195	0x000001...	Unsigned32	read-only
No	0x1001.00	Error Register	0x00		Unsigned8	read-only
No	0x1003.00	number of elements	0	0	Unsigned8	read-write
No	0x1003.01	standard error field	0x00		Unsigned32	read-only
No	0x1003.02	standard error field	0x00		Unsigned32	read-only
Yes	0x1005.00	COB-ID SYNC mes...	0x40000080	0x800000...	Unsigned32	read-write
Yes	0x1006.00	Communication Cyc...	50000	0	Unsigned32	read-write
No	0x1008.00	Manufacturer Devi...	AGP/LT-C...	AGP/LT-...	Visible string	constant
No	0x1009.00	Manufacturer Hard...	1.0	1.0	Visible string	constant
No	0x100A.00	Manufacturer Softw...	1.100	1.100	Visible string	constant
No	0x1014.00	COB-ID Emergency...	0xFF	$NODEID...	Unsigned32	read-write
No	0x1016.00	number of elements	64	64	Unsigned8	read-only
No	0x1016.01	Consumer Heartbe...	0	0	Unsigned32	read-write

7 Click [I/O Screen] in [I/O Driver Settings] or select [I/O Screen] in the [Screen List] window on Work Space to assign a variable to each of the mapped objects. For how to assign variables, refer to the following.

☞ "30.7.3 Allocating I/O" (page 30-155)

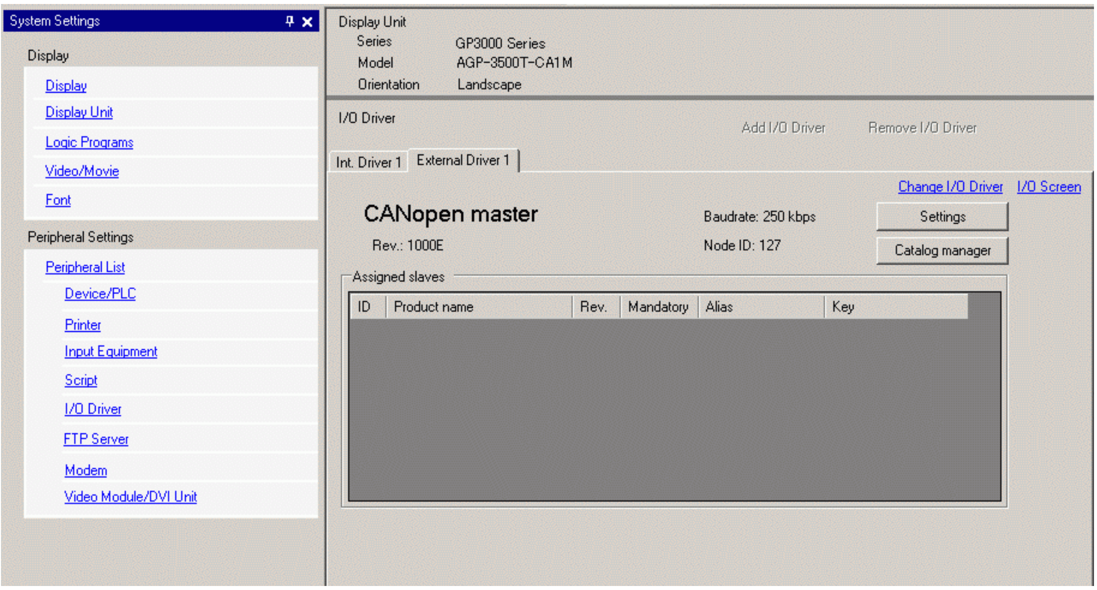
8 Create a Logic Screen and a Base Screen to access the allocated variables and transfer them to the GP.

NOTE

- To check set values, use SDOR instructions.

30.7.6 [I/O Driver] Settings Guide

When AGP-*****-CA1M/LT is selected in Display Settings, click [I/O Driver] in System Settings to open the following screen.



Display Unit
Series GP3000 Series
Model AGP-3500T-CA1M
Orientation Landscape

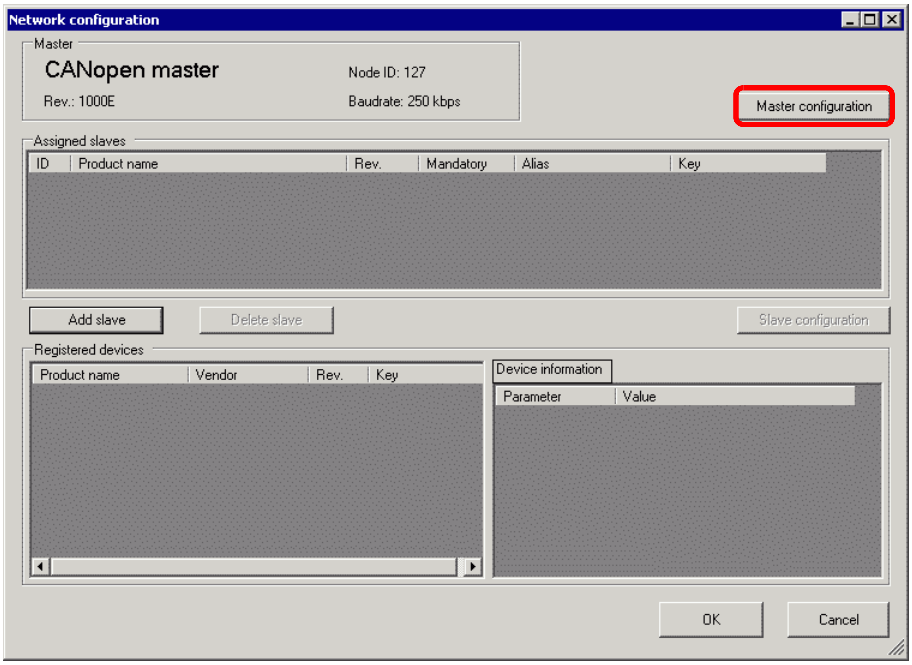
I/O Driver

CANopen master
Rev.: 1000E
Baudrate: 250 kbps
Node ID: 127

Setting		Description
Revision		Displays revision number for EDS file of CANopen master.
Baud rate		Displays baud rate of CANopen master.
Node ID		Displays node ID of CANopen master.
I/O Screen		Go to I/O Screen.
Settings		Displays [Network Configuration] dialog box. ☞ " ■ Network Configuration Dialog Box" (page 30-171)
Catalog Manager		Displays the [Catalog Manager] dialog box. ☞ " ■ Catalog Manager Dialog Box" (page 30-188)
Assigned slaves		Slave devices added to CANopen network displays. The details of [Network Configuration] and [Slave Configuration] dialog boxes display in the list.
	ID	Node ID of the slave in CANopen network.
	Product Name	Displays the product name of the slave.
	Revision	Displays the slave's EDS file revision number.*1
	On Error Control Event of a Mandatory Slave	Displays whether the slave is mandatory in the network.
	Key	Displays the device registration key of the slave. The device registration key will be the name of the EDS file without extension.

*1 EDS file is a text file in ASCII format that describes specification (example, useful features and objects) of the device. It is required to register and set the device to the network.

■ Network Configuration Dialog Box



Setting		Description
Master Configuration		Displays Master Configuration dialog box. ☞ " ◆ Master Configuration Dialog Box" (page 30-173)
Assigned slaves		Slave devices added to CANopen network displays. The details of [Network Configuration] and [Slave Configuration] dialog boxes display in the list.
	ID	Node ID of the slave in CANopen network.
	Product Name	Displays the product name of the slave.
	Revision	Displays the slave's EDS file revision number.*1
	On Error Control Event of a Mandatory Slave	Defines whether the slave is mandatory in the network. Select this option to turn ON bit 3 in object 1F81h.
	Alias	Defines the comment set for the slave.
	Key	Displays the device registration key of the slave. The device registration key will be the name of the EDS file without extension.
Add Slave		Add the slave selected in [EDS List] to [Assigned slaves]. You can change Node ID (1 to 63) of the added slave. You can also enter comments within 18 characters. In order to add a slave manufactured by another company, you need to import the EDS file using [Catalog Manager]. ☞ " ■ Catalog Manager Dialog Box" (page 30-188)

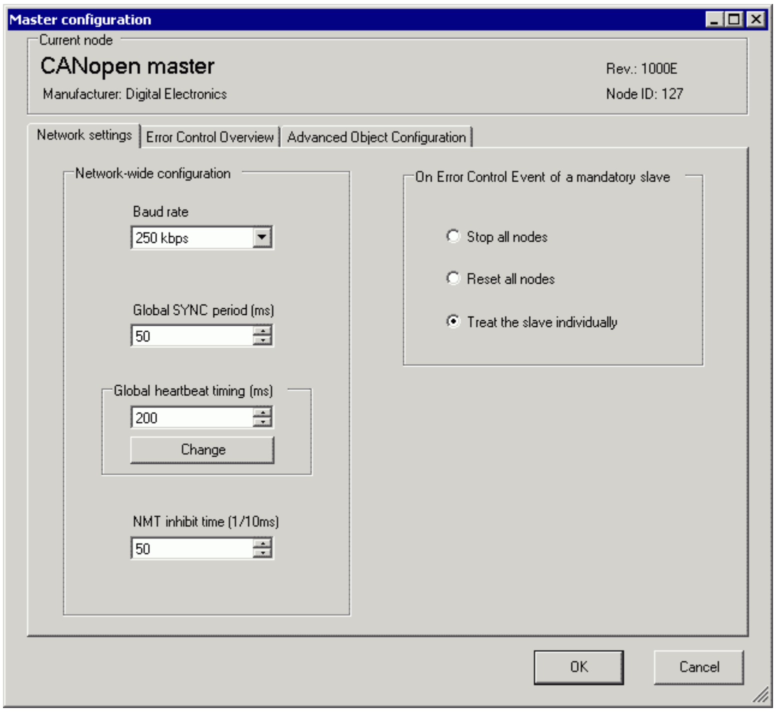
Continued

Setting		Description
Delete Slave		Deletes the slave selected in [Assigned slaves] from the network.
Slave Configuration		Displays Slave Configuration dialog box. ☞" ◆ Slave Configuration Dialog Box" (page 30-179)
Registered EDS		Displays available slaves in the list.
	Product Name	Displays the product name of the slave.
	Vendor Name	Displays the vendor name of slave.
	Revision	Displays the revision number for EDS file of the slave.
	Key	Displays the device registration key of the slave. The device registration key will be the name of the EDS file without extension.
Device Information		Information on the currently selected EDS displays.

*1 EDS file is a text file in ASCII format that describes specification (example, useful features and objects) of the device. It is required to register and set the device to the network.

◆ Master Configuration Dialog Box

- Master Configuration



Current node

CANopen master

Manufacturer: Digital Electronics

Rev.: 1000E

Node ID: 127

Network-wide configuration

Baud rate: 250 kbps

Global SYNC period (ms): 50

Global heartbeat timing (ms): 200

NMT inhibit time (1/10ms): 50

On Error Control Event of a mandatory slave

○ Stop all nodes

○ Reset all nodes

● Treat the slave individually

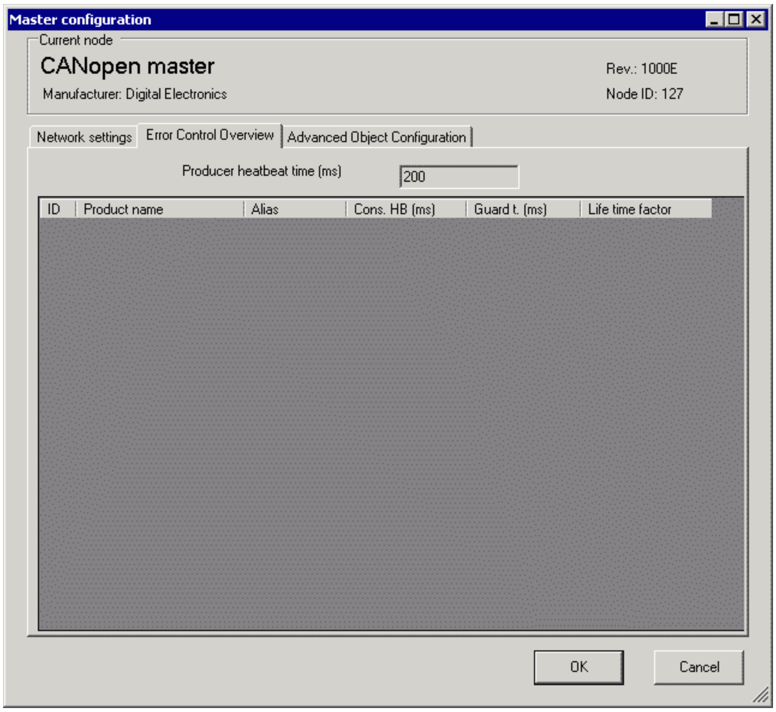
<table>
<tr><th colspan="2">Setting</th><th>Description</th></tr>
<tr><td colspan="2">Network-wide configuration</td><td>Configures CANopen network settings.</td></tr>
<tr><td rowspan="2"></td><td>Baud rate</td><td>Selects corresponding baud rate from [50kbps], [125kbps], [250kbps], [500kbps], [800kbps] and [1000kbps].
<table><tr><th>Baud rate (bps)</th><th>Wire length (m)</th></tr><tr><td>50K</td><td>1000</td></tr><tr><td>125K</td><td>500</td></tr><tr><td>250K</td><td>250</td></tr><tr><td>500K</td><td>100</td></tr><tr><td>800K</td><td>25</td></tr><tr><td>1000K</td><td>4</td></tr></table>
NOTE
• Set the slave baud rate on the slave itself.</td></tr>
<tr><td>Global SYNC period</td><td>Defines the frequency of the SYNC signal sent by the slave unit. The set up range is 0, or 3 to 32767. The set up value is stored in object 1006h.</td></tr>
</table>

Continued

<table>
<tr><th colspan="2">Setting</th><th>Description</th></tr>
<tr><td rowspan="2">Network Configuration</td><td>Global Heartbeat Timing</td><td>Set the heartbeat time sent from slave to master (or from master to slave). Unit is in milliseconds (ms). The value can be from 50 to 21844. Press [Apply to All Slaves] and the value of object 1016h (consumer heartbeat time) for CANopen master changes to the automatically optimized value. The value of object 1017h (producer heartbeat time) is overwritten by the value set here. Object 1017h (producer heartbeat time) for all slave objects is overwritten with the value set here, and the value of object 1016h (consumer heartbeat time) changes to the automatically optimized value.
NOTE
• To individually specify heartbeat time on slave side, it can be set in the [Slave Configuration] dialog box.</td></tr>
<tr><td>NMT Inhibit Time</td><td>Set the lagged time after the master input one NMT command into the network until the next NMT command is issued.
"NMT Inhibit Time" is set in 100 microseconds. The value can be from 0 to 32767.
You can disable this feature by setting 0.</td></tr>
<tr><td colspan="2">On Error Control Event of a Mandatory Slave</td><td>Select the action when an error occurs in the slave set in [Mandatory Slave] from [Stop All Slaves], [Reset All Slaves] and [Prioritize slaves individually]. When this item is set, bit 4 and 6 of the object 1F80h for CANopen master is as follows:
<table>
<tr><th></th><th>4 Bit</th><th>6 Bit</th></tr>
<tr><td>Stop all slaves</td><td></td><td>ON</td></tr>
<tr><td>Reset all slaves</td><td>ON</td><td>OFF</td></tr>
<tr><td>Treat the slave individually</td><td>OFF</td><td>OFF</td></tr>
</table>
</td></tr>
</table>

- Error Control Overview

Displays the state of each slave. No settings are allowed.



Master configuration

Current node

CANopen master

Manufacturer: Digital Electronics

Rev.: 1000E

Node ID: 127

Producer heatbeat time (ms) 200

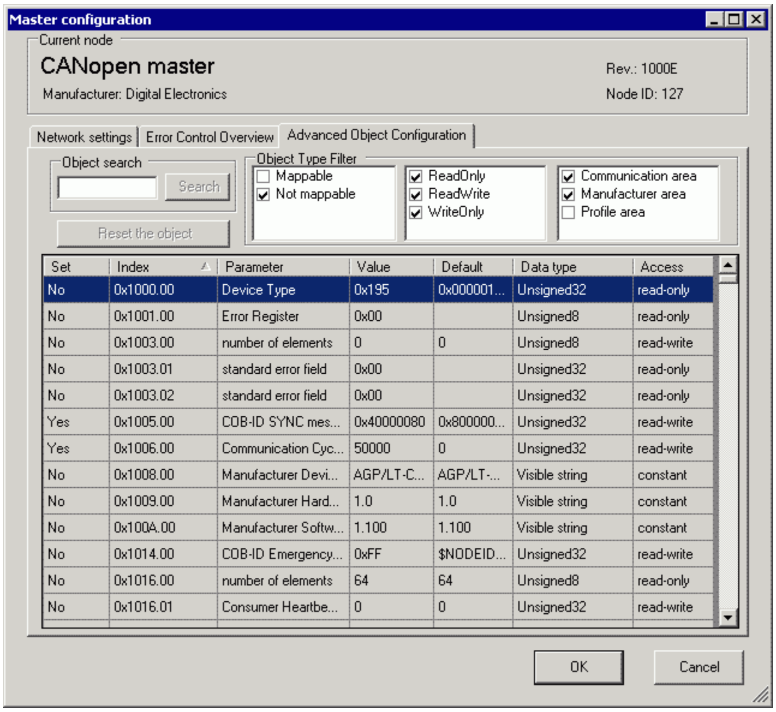
ID	Product name	Alias	Cons. HB (ms)	Guard t. (ms)	Life time factor

Setting	Description
Producer Heartbeat Time	Displays the sending period of the heartbeat set in [Global Heartbeat Time] in [Network Settings] tab. Unit is in milliseconds (ms). It is stored in the object 1017h for CANopen master.
ID	Node ID in CANopen network.
Product Name	Display EDS product name.
Alias	Displays the alias assigned to the slave.
Consumer Heartbeat	Display consumer heartbeat set in each slave. It is stored in the object 1016h for slave.
Guard Time	When [Use Node Guard] is selected in the [Advanced Error Control] tab in [Slave Configuration], displays [Guard Time] that is set. It is stored in the object 100Ch on slave side.
Life time factor	When [Use Node Guard] is selected in [Advanced Error Control] tab in [Slave Configuration], displays [Life time factor] that is set. It is stored in the object 100Dh on slave side.

- Advanced Object Configuration

You can change the values of read-write and write-only objects. The values you can change are objects that are writable and not grayed out. The values set here are written in the objects during startup.

NOTE
- To use Advanced Object Configuration, you must have sufficient knowledge about CANopen.



Current node

CANopen master

Manufacturer: Digital Electronics

Rev.: 1000E

Node ID: 127

Object search

Object Type Filter

□ Mappable
☑ Not mappable

☑ ReadOnly
☑ ReadWrite
☑ WriteOnly

☑ Communication area
☑ Manufacturer area
□ Profile area

Set	Index	Parameter	Value	Default	Data type	Access
No	0x1000.00	Device Type	0x195	0x000001...	Unsigned32	read-only
No	0x1001.00	Error Register	0x00		Unsigned8	read-only
No	0x1003.00	number of elements	0	0	Unsigned8	read-write
No	0x1003.01	standard error field	0x00		Unsigned32	read-only
No	0x1003.02	standard error field	0x00		Unsigned32	read-only
Yes	0x1005.00	COB-ID SYNC mes...	0x40000080	0x800000...	Unsigned32	read-write
Yes	0x1006.00	Communication Cyc...	50000	0	Unsigned32	read-write
No	0x1008.00	Manufacturer Devi...	AGP/LT-C...	AGP/LT-...	Visible string	constant
No	0x1009.00	Manufacturer Hard...	1.0	1.0	Visible string	constant
No	0x100A.00	Manufacturer Softw...	1.100	1.100	Visible string	constant
No	0x1014.00	COB-ID Emergency...	0xFF	$NODEID...	Unsigned32	read-write
No	0x1016.00	number of elements	64	64	Unsigned8	read-only
No	0x1016.01	Consumer Heartbe...	0	0	Unsigned32	read-write

Setting	Description
Object Search	Enter the object you want to search, or enter the parameter text string. Click [Search]. The application searches for the corresponding object in the object list.
Reset the object	Returns the set up value of the object selected in the [Object List] to its initial value.

Continued

Setting		Description
Object Type Filter		Select the extracting conditions you want to display in [Available objects from device profile] from the following three categories: • PDO mapping PDO mapping enabled PDO mapping disabled • Access Read only Write only Read/Write enabled • Data Area Commutation Area (1000h to 1FFFh) Manufacturer Area (2000h to 5FFFh) Profile Area (6000h to 9FFFh)
Available objects from device profile		Display the list of objects according to the conditions selected in [Object type filter].
	Enable	When the "Set Value" changes, this check box is automatically selected; you can confirm that the object has changed. When the check box is cleared, the object returns to its initial value.
	Index	Display index and sub index of the object. The sub index appears after the comma. For example, for "0x1003.2", "0x1003" is index and "2" is sub index.
	Parameter	Display parameter name of the object.
	Value	The value of the object can be changed. The number with "0x" indicates a hexadecimal, otherwise without, it indicates a decimal number.
	Default	Display initial value of the object.

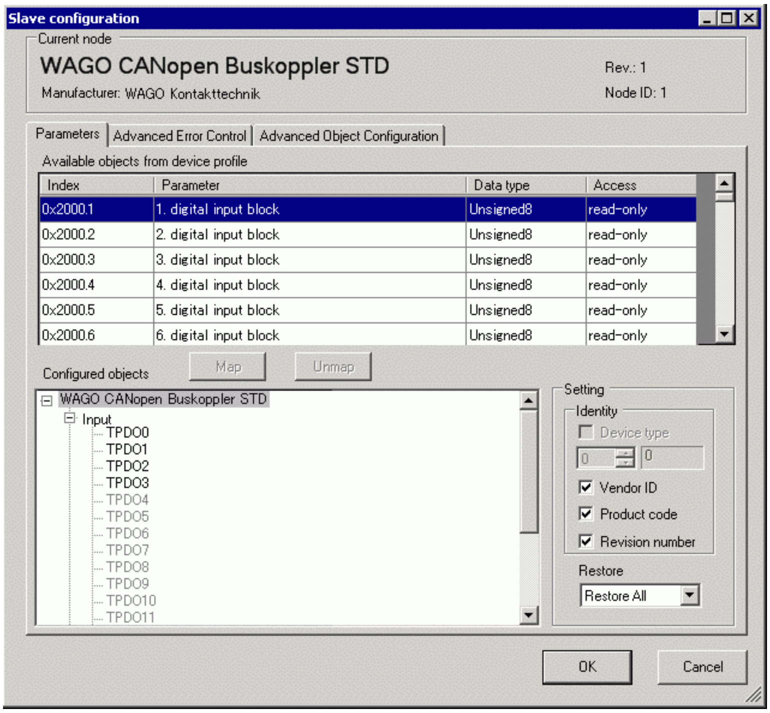
Continued

Setting		Description
Available objects from device profile	Data Type	Display data type of the object. The following data types are included. • Boolean (Single bit value) • Integer8 (Integer with 8 bit code) • Integer16 (Integer with 16 bit code) • Integer24 (Integer with 24 bit code) • Integer32 (Integer with 32 bit code) • Integer40 (Integer with 40 bit code) • Integer48 (Integer with 48 bit code) • Integer56 (Integer with 56 bit code) • Integer64 (Integer with 64 bit code) • Unsigned8 (Integer without 8 bit code) • Unsigned16 (Integer without 16 bit code) • Unsigned24 (Integer without 24 bit code) • Unsigned32 (Integer without 32 bit code) • Unsigned40 (Integer without 40 bit code) • Unsigned48 (Integer without 48 bit code) • Unsigned56 (Integer without 56 bit code) • Unsigned64 (Integer without 64 bit code) • Float (32 bit single accuracy fixed decimal point) • Float64 (64 bit single accuracy fixed decimal point) • Visible String (Text string including ASCII text) • Octet string (Array of integer without 8 bit code) • Unicode string (Array of integer without 16 bit code) • Bit string (Array of single bit) • Time of day (48 bit value indicating time and date) • Time difference (48 bit value indicating time) • Domain (Application specific data block) • Reserved (Reserved type)
	Access	Displays access method of the objects. The following types are included. • readonly (Read only) • writeonly (Write only) • readwrite (Read/Write) • constant (Constant)

◆ Slave Configuration Dialog Box

Configure detailed settings of the slave selected in [Slave Configuration].

- Parameters



Current node
WAGO CANopen Buskoppler STD
Manufacturer: WAGO Kontakttechnik
Rev.: 1
Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x2000.1	1. digital input block	Unsigned8	read-only
0x2000.2	2. digital input block	Unsigned8	read-only
0x2000.3	3. digital input block	Unsigned8	read-only
0x2000.4	4. digital input block	Unsigned8	read-only
0x2000.5	5. digital input block	Unsigned8	read-only
0x2000.6	6. digital input block	Unsigned8	read-only

Configured objects

```
⊟ WAGO CANopen Buskoppler STD
   ⊟ Input
      TPDO0
      TPDO1
      TPDO2
      TPDO3
      TPDO4
      TPDO5
      TPDO6
      TPDO7
      TPDO8
      TPDO9
      TPDO10
      TPDO11
```

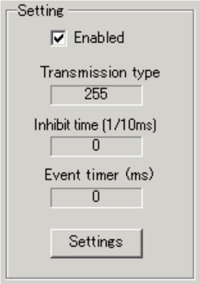
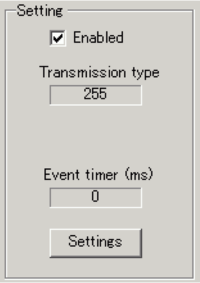
Setting
Identity
□ Device type
0 0
☑ Vendor ID
☑ Product code
☑ Revision number
Restore
Restore All

Setting		Description
Available objects from device profile		Display the list of objects that can be mapped to PDO.
	Index	Display index and sub index of the object. The sub index appears after the comma. For example, for "0x1003.2", "0x1003" is index and "2" is sub index.
	Parameter	Display parameter name of the object.

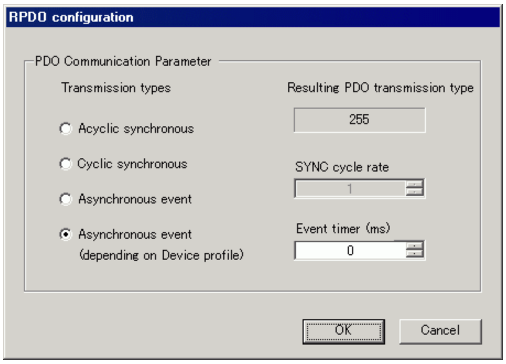
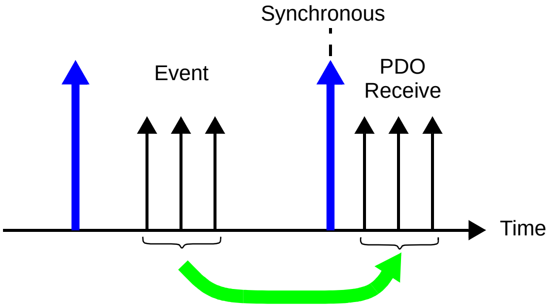
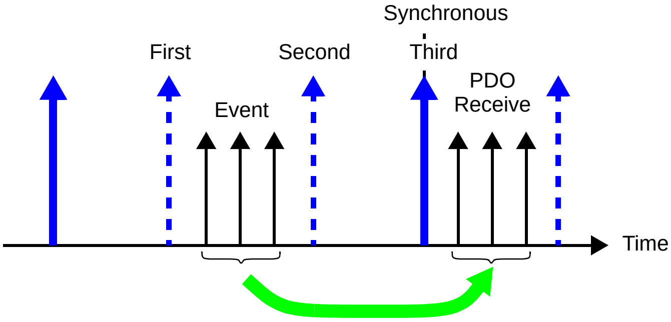
Continued

Setting		Description
Available objects from device profile	Data Type	Display data type of the object. The following data types are included. • Boolean (Single bit value) • Integer8 (Integer with 8 bit code) • Integer16 (Integer with 16 bit code) • Integer24 (Integer with 24 bit code) • Integer32 (Integer with 32 bit code) • Integer40 (Integer with 40 bit code) • Integer48 (Integer with 48 bit code) • Integer56 (Integer with 56 bit code) • Integer64 (Integer with 64 bit code) • Unsigned8 (Integer without 8 bit code) • Unsigned16 (Integer without 16 bit code) • Unsigned24 (Integer without 24 bit code) • Unsigned32 (Integer without 32 bit code) • Unsigned40 (Integer without 40 bit code) • Unsigned48 (Integer without 48 bit code) • Unsigned56 (Integer without 56 bit code) • Unsigned64 (Integer without 64 bit code) • Float (32 bit single accuracy fixed decimal point) • Float64 (64 bit single accuracy fixed decimal point) • Visible String (Text string including ASCII text) • Octet string (Array of integer without 8 bit code) • Unicode string (Array of integer without 16 bit code) • Bit string (Array of single bit) • Time of day (48 bit value indicating time and date) • Time difference (48 bit value indicating time) • Domain (Application specific data block) • Reserved (Reserved type)
	Access	Displays access method of the objects. The following types are included. • readonly (Read only) • writeonly (Write only) • readwrite (Read/Write) • constant (Constant)
Map		Map the object selected in [Available objects from device profile] to [Configured objects] tree.
Unmap		Remove the object mapped to [Configured objects] tree.
Configured objects		Mapped objects per slave displays in tree configuration. Map the object. TPDO PDO sent from slave to master. When data is input from the external I/O that's connected to the selected slave, map the object here. RPDO PDO sent from master to slave. When data is output from the external I/O that's connected to the selected slave, map the object here.

Continued

Setting			Description
Settings			Configure detailed settings of the item selected in [Configured objects]. Setting contents vary from when: the slave is selected on the tree, TPDO or RPDO is selected, or the mapped object is selected.
	Verification		Defines whether or not to run the device type reference. Select this option to check whether the structure matches the set up value in object 1000h. If 1000h is zero, reference does not take place.
		Vendor ID	Displays the vendor ID of slave.
		Product Code	Displays the product code of the slave.
		Revision	Displays the revision number for EDS file of the slave.
		Restore Parameters	Return the parameters to the initial value when restarting the network. You can set so as to restore the parameters for each slave. • None Parameters are not restored. • All All parameters are restored. • Communication only Parameters in the area shown in [Commutation Area] of [Slave Configuration/Extended] are restored. • Application only Parameters in the area shown in [Manufacturer Area] and Profile Area] of [Slave Configuration/Extended] are restored.
	When TPDO/ RPDO Is Selected		Set PDO enabled/disabled and set [Transmission Type], [Inhibit Time] and [Event Timer].  Setting ☑ Enabled Transmission type: 255 Inhibit time (1/10ms): 0 Event timer (ms): 0  Setting ☑ Enabled Transmission type: 255 Event timer (ms): 0 NOTE • When PDO is disabled, when you re-open the project, all the information set up in the PDO is initialized to its initial values.
		Enable	Select the check box to enable PDO.
		Transmission Type	Display the transmission type for PDO in [Change Settings].
		Inhibit Time	Display time when continuously transmitting PDO is inhibited only for TPDO.Set in [Change Settings]. The unit is 100 µs.
		Event Timer	Displays transmission interval in which PDO is continuously sent. Set in [Change Settings]. The unit is in milliseconds (ms).

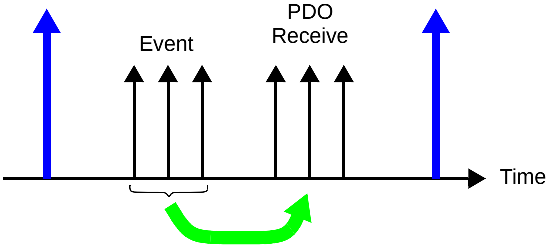
Continued

Setting			Description
Settings	When TPDO/RPDO Is Selected	Change Settings	Click to display the following screen. Different screens are displayed for RPDO and TPDO. When RPDO is selected  RPDO configuration PDO Communication Parameter Transmission types ○ Acyclic synchronous ○ Cyclic synchronous ○ Asynchronous event ● Asynchronous event (depending on Device profile) Resulting PDO transmission type: 255 SYNC cycle rate: 1 Event timer (ms): 0 OK Cancel • Acyclic synchronous (non cyclic) PDO is received when an event occurs as in the case in which 255/ 254 is set, but its timing is after the next SYNC signal flows into the network. In addition, if no event occurs, PDO is not received even when the SYNC signal has flown to the network.  • Cyclic synchronous (cyclic) In synchronization with the SYNC signal flowing into the network, PDO is received based on the value set to the SYNC cyclic number. For example, when 1 is set, PDO is received for every SYNC object. When 3 is set, PDO is received each time that three SYNC signals flow into the network. The following figure shows the case in which 3 is set.  Synchronous First Second Third Event PDO Receive Time

Continued

<table>
<tr><th colspan="3">Setting</th><th>Description</th></tr>
<tr><td>Settings</td><td>When TPDO/RPDO Is Selected</td><td>Change Settings</td><td>

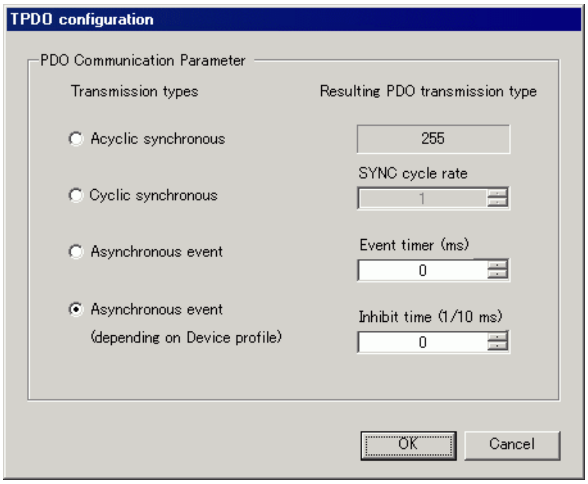
- Asynchronous event (standard event)/Determined by (standard event) device profile
PDO is received immediately when an event occurred regardless of the SYNC signal. For 254, this is defined by the manufacturer. For 255, this is defined in the device profile.



The information of PDO that has been set is reflected for the following objects:

<table>
<tr><th>Object number</th><th>SubIndex</th><th>Item Name</th></tr>
<tr><td>0x1400 + PDO number</td><td>0</td><td>Number of RPDO</td></tr>
<tr><td></td><td>1</td><td>COB ID</td></tr>
<tr><td></td><td>2</td><td>Transmission Type</td></tr>
<tr><td></td><td>3</td><td>Inhibit Time</td></tr>
<tr><td></td><td>5</td><td>Event Timer</td></tr>
</table>

TPDO



TPDO configuration

PDO Communication Parameter

Transmission types

○ Acyclic synchronous

○ Cyclic synchronous

○ Asynchronous event

● Asynchronous event (depending on Device profile)

Resulting PDO transmission type: 255

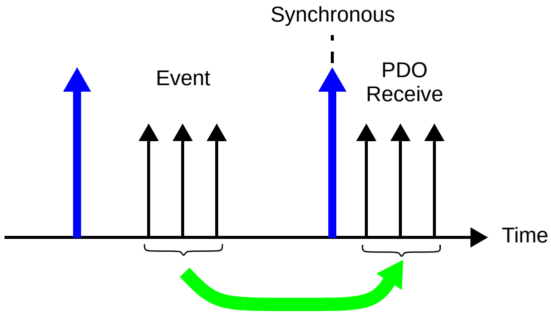
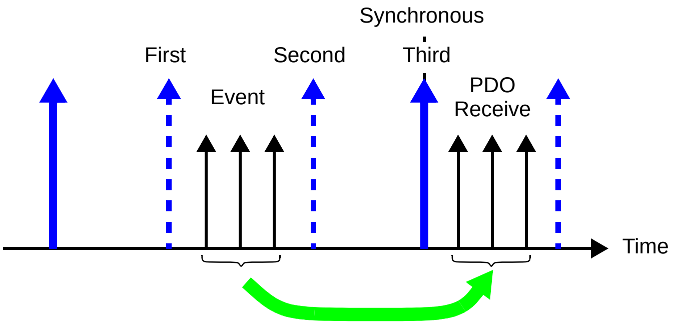
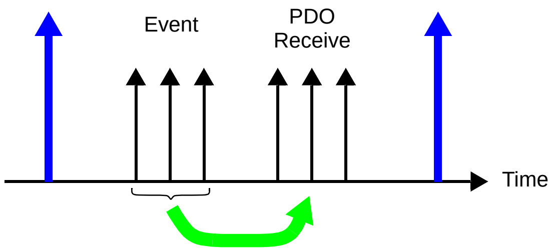
SYNC cycle rate: 1

Event timer (ms): 0

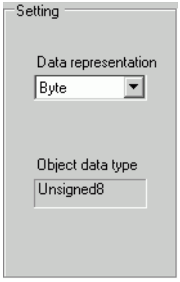
Inhibit time (1/10 ms): 0

</td></tr>
</table>

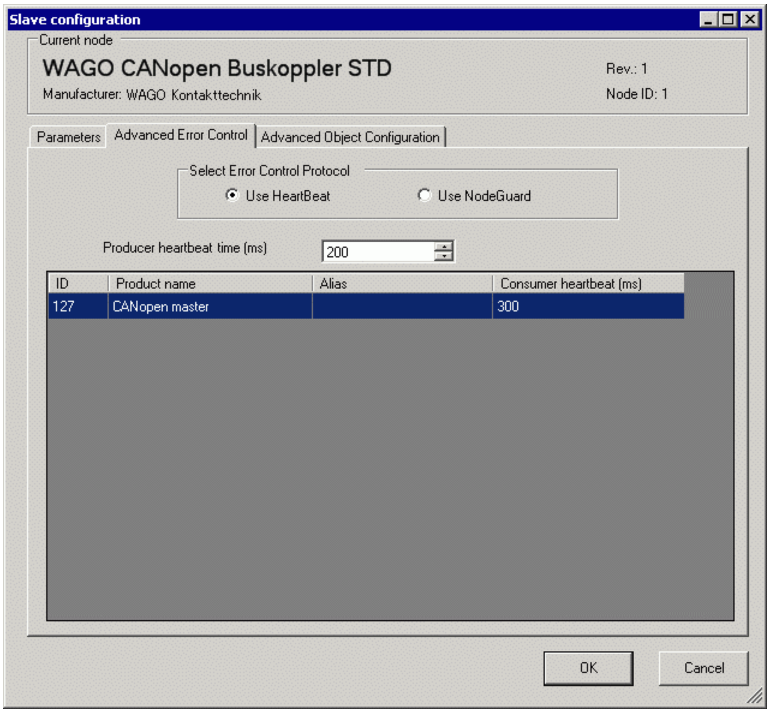
Continued

Setting			Description
Settings	When TPDO/RPDO Is Selected	Change Settings	• Acyclic synchronous (non cyclic) PDO is sent when an event occurs as in the case in which 255/254 is set, but its timing is after the next SYNC signal flows into the network. In addition, if no event occurs, PDO is not sent even when the SYNC signal has flown to the network.  • Cyclic synchronous (cyclic) In synchronization with the SYNC signal flowing into the network, PDO is sent based on the value set to the SYNC cyclic number. For example, when 1 is set, PDO is sent for every SYNC object. When 3 is set, PDO is sent each time that three SYNC signals flow into the network. The following figure shows the case in which 3 is set.  Synchronous First Second Third Event PDO Receive Time • Asynchronous event (standard event)/Determined by (standard event) device profile PDO is sent immediately when an event occurred regardless of the SYNC signal. For 254, this is defined by the manufacturer. For 255, this is defined in the device profile. 

Continued

<table>
<tr><th colspan="2">Setting</th><th>Description</th></tr>
<tr><td>Settings</td><td>When the object is selected</td><td>The information of PDO that has been set is reflected for the following objects:
<table>
<tr><td>Object number</td><td>SubIndex</td><td>Item Name</td></tr>
<tr><td rowspan="5">0x1800 + PDO number</td><td>0</td><td>Number of TPDO</td></tr>
<tr><td>1</td><td>COB ID</td></tr>
<tr><td>2</td><td>Transmission Type</td></tr>
<tr><td>3</td><td>Inhibit Time</td></tr>
<tr><td>5</td><td>Event Timer</td></tr>
</table>
Configure how to work with the mapped object. Display on I/O Screen varies depending on the type set here.
☞ "30.7.3 Allocating I/O ■ I/O Screen Display" (page 30-156)
• Type
Select from "Bit", "Byte", "Word" and "Dword" (Double Word).
• Data Type
Data type of the selected object displays.

Setting
Data representation: Byte
Object data type: Unsigned8
</td></tr>
</table>

- Advanced Error Control (when heartbeat is selected)



Slave configuration

Current node

WAGO CANopen Buskoppler STD

Manufacturer: WAGO Kontakttechnik

Rev.: 1

Node ID: 1

Select Error Control Protocol

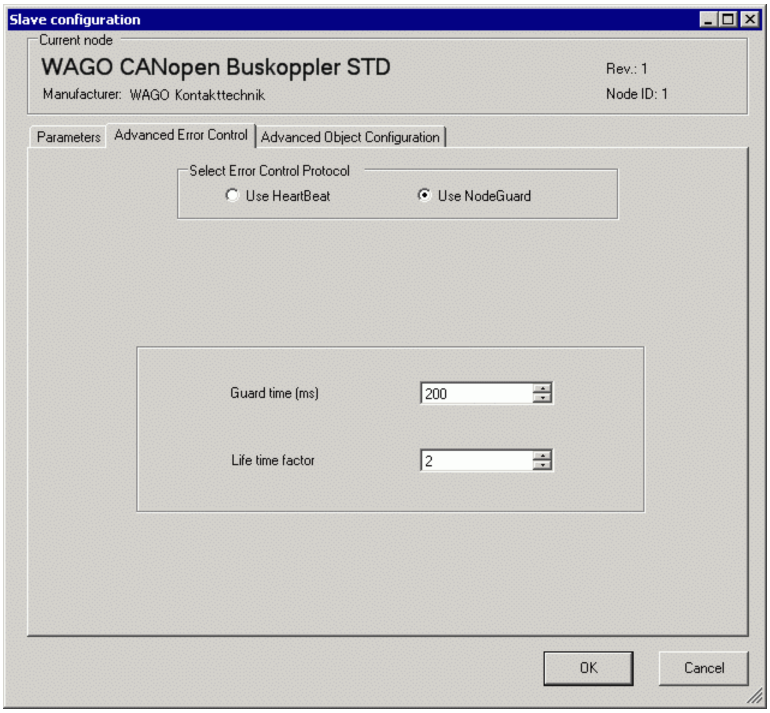
● Use HeartBeat ○ Use NodeGuard

Producer heartbeat time (ms): 200

ID	Product name	Alias	Consumer heartbeat (ms)
127	CANopen master		300

Setting	Description
Producer Heartbeat Time	Set the transmission cycle of heartbeat. Unit is in milliseconds (ms). The value can be from 50 to 21844. It is stored in the object 1017h on slave side.
ID	Displays Node ID of CANopen master.
Product Name	Displays the product name of CANopen master.
Alias	Displays alias of CANopen master.
Consumer Heartbeat	Consumer heartbeat of CANopen master displays. Stored in the object 1016h on master side.

- Advanced Error Control (when no guard is selected)



Slave configuration

Current node

WAGO CANopen Buskoppler STD

Manufacturer: WAGO Kontakttechnik

Rev.: 1

Node ID: 1

Select Error Control Protocol

○ Use HeartBeat ● Use NodeGuard

Guard time (ms): 200

Life time factor: 2

Setting	Description
Guard Time	When NMT master polls the slaves, sets up the frequency for when slaves receive polling requests. Unit is in milliseconds (ms). The set up range is 0, or 50 to 32767. Heartbeat is used when set to zero. It is stored in object 100Ch.
Life time factor	Set the time to monitor errors when NMT master polls the slaves. An error occurs if it is not polled when the time which is the value set in [Guard Time] multiplied by the value set here has passed. The set up range is 0, or 2 to 255. It is stored in object 100Dh.

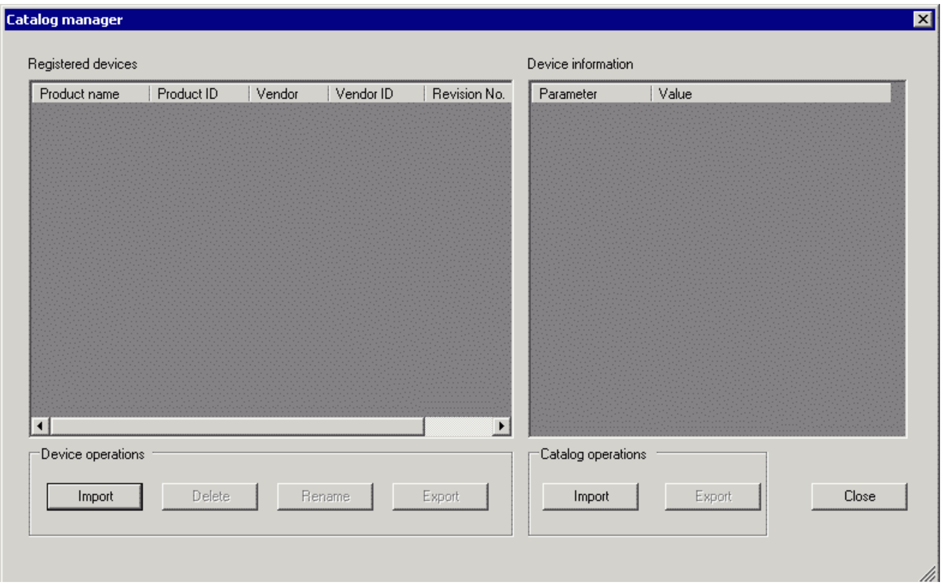
- Extended

You can change the values of read-write and write-only objects for access method. The values you can change are objects that are writable and not grayed out. The values set here are written in the objects on slave side on startup.

Detail settings are the same as those of [Advanced Object Configuration] tab in [Master Configuration] dialog box.

☞ 30.7.6 [I/O Driver] Settings Guide ◆ Master Configuration Dialog Box •Advanced Object Configuration 30-176

■ Catalog Manager Dialog Box



Setting		Description
Registered Devices		Displays a list of the registered slaves In order to register a slave, import the EDS file using [Import] in the [Device] section.
	Product Name	Displays the product name of the slave.
	Product Code	Displays the product code of the slave.
	Vendor Name	Displays the vendor name of slave.
	Vendor ID	Displays the vendor ID of slave.
	Revision	Displays the revision number for EDS file of the slave.
	Key	Displays the device registration key of the slave. The device registration key will be the name of the EDS file without extension.
Device		Slave can be registered or deleted. NOTE • In order to edit/transfer the project file (PRX) in the environment other than the PC where the file is created, you need to import the device information (EDS file) again.
	Import	Imports an EDS file to register a slave.
	Delete	Deletes the EDS file specified in [Registered Devices] to delete the registered slave. Note that the device that is currently used in the open project cannot be deleted.
	Change Key	Changes [Key].
	Export	Exports an EDS file.

Continued

Setting		Description
Catalog		In order to edit/transfer the project file (PRX) in the environment other than the PC where the file is created, you need to import the device information for each slave again. You can collectively import/export device information of multiple slaves.
	Import	Imports a CAT file in which device information of multiple slaves is contained.
	Export	Exports device information of multiple slaves as a CAT file.

30.7.7 Error information

The system variable #L_IOStatus stores error information in the bottom 8 bits.

#L_IOStatus

H	Reserved (0 Fixed)								
L	Major Fault	0	0	0	0	0	0	0	Error Code

Major Fault
"1" is set when it detects a malfunction that needs to stop the logic.

■ Error code

	Error Code	Error Message	Description		Solution
Project date related error	000	No error	—	Initialization error	—
	001	Illegal Instruction(s)	I/O Driver Instruction Error		Check the set value of operand for I/O Driver Instructions and re-transfer the project.
	002	Too many instructions	Number of used I/O Driver Instructions is greater than 16		Check the number of used I/O Driver Instructions and re-transfer the project.
	003	Driver ID error	The driver/unit results in an error while registering, and they have not been registered.		The project file might not have been sent properly. Transfer the project file again.
	004	Repeat driver setting	The driver is registered twice.		
	005	Setting level value error	The driver is not correct.		
	006	Data obtaining address error	The driver information is in correct. The controller information is not correct.		
	007	Driver not registered	Driver is not registered		
	008	PDO Settings Error Initialization error	PDO Settings Error Initialization error		Confirm PDO Settings and re-transfer the project.
	009	Invalid terminal type	Invalid terminal type		The project file might not have been sent properly. Transfer the project file again.
	010	Concise DCF not found	There is no concise DCF		
	011	Invalid Concise DCF	Invalid concise DCF		
	012	F/W File Read Error	Invalid firmware		
	013	Inappropriate firmware	Failure in downloading set value		Reset AGP. If the problem is unresolved, please contact your support center.

Continued

	Error Code	Error Message	Description		Solution
Project date related error	014	Setting value error	Error code is 0 when error flag is set	Initialization error	Reset AGP. If the problem is unresolved, please contact your support center.
H/W related error	050	I/O board ID different	The connected I/O board is not correct.	Hardware error	Display type may be different. Confirm the display type and transfer the project file again.
	051	CANopen unit initialization error	The I/O board initialization fails.		The project file might not have been sent properly. Transfer the project file again. If the problem is still not solved, there may be a problem with the hardware. Contact your support center.

Continued

	Error Code	Error Message	Description		Solution
Application related error	100	Fatal error: Bus off	Bus failure (such as noise failure), hardware failure, abnormal baud rate settings, etc.	Fatal master error	Confirm the connection state of the communication cable and check the baud rate settings of all the slaves to re-transfer the project. If the problem is still not solved, there may be a problem with the hardware. Contact your support center.
	101	Fatal error: high priority receive queue overrun	Excessive amount of received data such as PDO, NMT, and SYNC.		Review the network settings by decreasing PDO.
	102	Fatal error: high priority transmit queue overrun	Excessive amount of sending data such as PDO, NMT and SYNC.		
	103	Fatal error: low priority receive queue overrun	Excessive amount of receiving data such as heartbeat, node guard, and SDO.		Review the network settings by increasing node guard intervals.
	104	Fatal error: low priority transmit queue overrun	Excessive amount of transmitting data such as heartbeat, node guard, and SDO.		
	105	Fatal error: CAN controller overrun	Data loss due to excessive amount of data such as PDO and SDO.		Review the network settings by decreasing PDO and SDO.
	106	Fatal error: duplicate node ID detected	More than two identical node IDs are used.		Confirm whether node ID is duplicated and re-transfer the project.
	107	Unsupported slave error	Unsupported slave exists		Review the slave configuration.

Continued

	Error Code	Error Message	Description		Solution
Application related error	108	Fatal error: invalid Concise DCF	Network configuration is wrong	Fatal master error	Please download concise DCF again. If the problem is unresolved, check all the EDS files and review the network settings.
	109	Mandatory slave error	Concise DCF file of the mandatory slave is invalid		Check the EDS file and confirm whether it matches the hardware configuration.
	110	Master fatal error	Fatal error has occurred in CANopen master.		Reset the display.
	111	Invalid value in instruction	An I/O driver instruction has run with an operand value that is out of range.	Runtime error	Please run each I/O driver with values inside specified ranges.
	112	No mandatory slave	There is a problem with the defined mandatory slave.		Confirm whether the mandatory slave is correctly connected and that the power is on.
	120	SDO communication error	Slave object information specified in SDO communication is invalid, or communication partner is unavailable for communication. *Excluding error codes 140 to 148.		Confirm whether the specified information of the SDO communication is correct. If correct, confirm the communication state of the network or reset the network.
	121	CANopen module timeout (Input)	I/O firmware is not updating input data.		I/O firmware is not operating properly. Reset the display.
	122	CANopen module timeout (Output)	I/O firmware does not update output data.		
	123	CANopen module interface is busy	I/O firmware is not running SDO communication.		

Continued

	Error Code	Error Message	Description		Solution
Application related error	130	Incorrect PDO received	Invalid size for the received PDO	Nonfatal error	Confirm the EDS file.
	131	SDO queue overrun	Excessive SDO communication data size		Make SDO communication data size smaller.
	132	Master alone	Slave does not exist		Confirm whether slave is correctly connected and that the power is on.
	140	SDO protocol error	SDO server (slave) protocol error		Check the abort code. Check the object dictionary of the slave.
	141	SDO send overflow	Overflow of sending queue with low priority (see error 104)		See error 104. Fatal error: Overrun of receiving queue with low priority
	142	SDO master setting error	AGP is not set as a CANopen master. Therefore, SDO sending is not performed.		Please check the network settings. Check the network status from the offline menu. Reset AGP.
	143	SDO access error	Object specified with the SDO instruction is accessed with another service via SDO.		Check whether only AGP is CANopen master through the network. (multiple masters are not supported.) Check whether only one SDO instruction is executed at the same time.
	144	SDO receive timeout	SDO request is not responded to within the SDO timeout period.		Check the abort code. Check whether the node ID exists in the network. Check whether the object exists in the object dictionary of the slave. Check whether the sub index exists.

Continued

	Error Code	Error Message	Description		Solution
Application related error	145	SDO operand error	SDO instruction parameter error	Nonfatal error	Check the abort code. Check the parameter of the SDO instruction. Check whether the object is read-only or write-only.
	146	SDO master status error	SDO transmission cannot be performed due to the status of AGP		Check the status of AGP with DGMT instruction or offline menu.
	147	SDO master status stopped	SDO sending cannot be performed because AGP is in the STOP mode or changed to the STOP mode during SDO transmission		Change the AGP to RUN mode.
	148	SDO abort error	SDO transmission was aborted by the SDO server (slave)		Check the abort code. Check the status of the slave.
	150	Identity error	The set slave is different from the connected slave.	Slave error	Check the EDS file and network settings to ensure that the node and EDS file are correct. You can find the node ID of the slave using an offline function.
	151	Optional slave error	Incorrect configuration of the optional slave		Confirm slave configuration in the network settings and re-transfer the project.
	152	Unexpected state for one or more mandatory slaves	Mismatched state of the Mandatory slave and that of the network		
	153	Abnormal Slave	Abnormal Slave		The project file might not have been sent properly. Transfer the project file again.
	154	Inconsistent Concise DCF for one or more slaves	CDCF is mismatched with the object dictionary of slave.		

Continued

	Error Code	Error Message	Description		Solution
Application related error	155	Concise DCF mismatch for one or more slaves	Slave configuration and the object directory are mismatched.	Slave error	Check whether correct EDS file is used. Using an offline function, you can find node ID of slave.
	156	Identity error for one or more slaves	One or multiple set slaves do not match the connected slave.		Check the EDS file and network settings to ensure that the node and EDS file are correct. You can find the node ID of the slave using an offline function.

30.7.8 Restrictions

- If a project with a different I/O Driver version is transferred, it takes time to start the GP.
- If there are many PDO data, the number of processes executing the logic increases, so scanning may take longer than the set scan time.
- Depending on the I/O driver settings, when the CANopen communication cable is disconnected, the CANopen network is reset upon restoration.
- The number of I/O bit points you can set up is 512 (input bits: 256 points, output bits: 256 points). The number of I/O integer points you can set up is 128 (input integer: 64 points, output integer: 64 points).
- Please do not set values outside the valid range. Even if you set up a value outside the range, an error may not display.
- If you open the network setup dialog box, despite not making any changes, a save project message may display when you close the project.
- After mapping I/O, deleting objects, disabling PDO, or deleting the slave will not cancel the I/O mapping.
- When using the CANopen driver, the undo operation is not available in the associated I/O screen or I/O driver settings.
- While the LT unit is powered ON, such as when it is running, do not install or remove the LT unit to/from the CANopen unit.