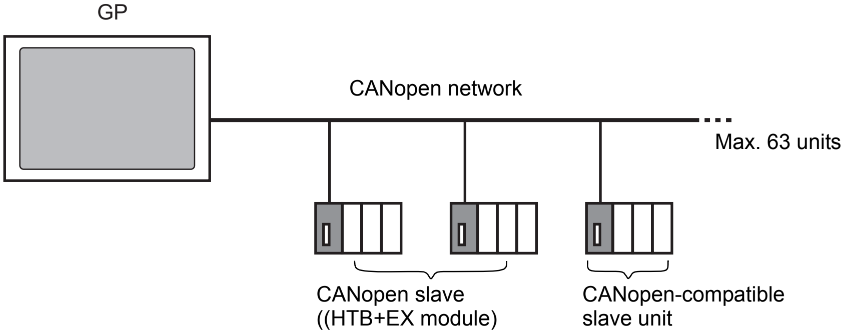
30.7 Controlling External I/O with CANopen

30.7.1 Summary

The Model AGP-*****-CA1M/LT has a built-in CANopen master driver. By connecting the CANopen slave unit "Hybrid Terminal Block" (model: HTB-1C0DM9LP; hereafter referred to as "HTB") manufactured by Digital Electronics Corporation or a CANopen-compatible slave unit to this GP, external I/O in a remote location can be controlled.
Moreover, by attaching an EX module to the HTB, I/O can be extended.



For one GP unit, an open network can be constructed, connecting with maximum of 63 HTB or CANopen compatible slave units. A maximum of seven EX modules can be attached to each HTB unit.

■ 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).
- For HTB, the communication is carried out in PDO packet units. The entire system can support up to 252 units of RxPDO and up to 252 units of TxPDO. When mapping to TPDO4+ and RPDO4+, together with all the slave settings, it is limited to a maximum of 64 units.
- 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.

■ HTB Features

With Digital Electronics Corporation's CANopen slave unit, the following features can be used.

Feature		Points	Description	Browse to
Standard I/O				
	Standard Input	12 points	-	☞ "30.7.5 HTB Standard Input/Output" (page 30-176)
	Standard Output	8 points	-	

NOTE

- To extend I/O by using an EX module, refer to the following.
  ☞ "30.7.6 Expanding I/O Using EX Module" (page 30-183)

*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.

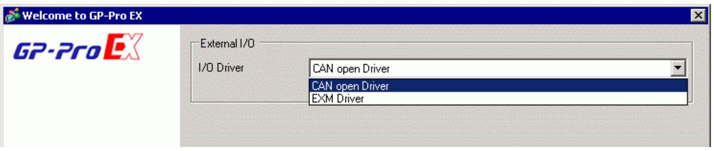
*2 Concise DCF is a device setting file in binary data format. The CANopen network is configured by downloading this data to the NMT master. NMT master is a feature that controls the actions of a slave node. Only one exists in any CANopen network; the node with this NMT master feature becomes the CANopen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

30.7.2 Setup Procedure

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

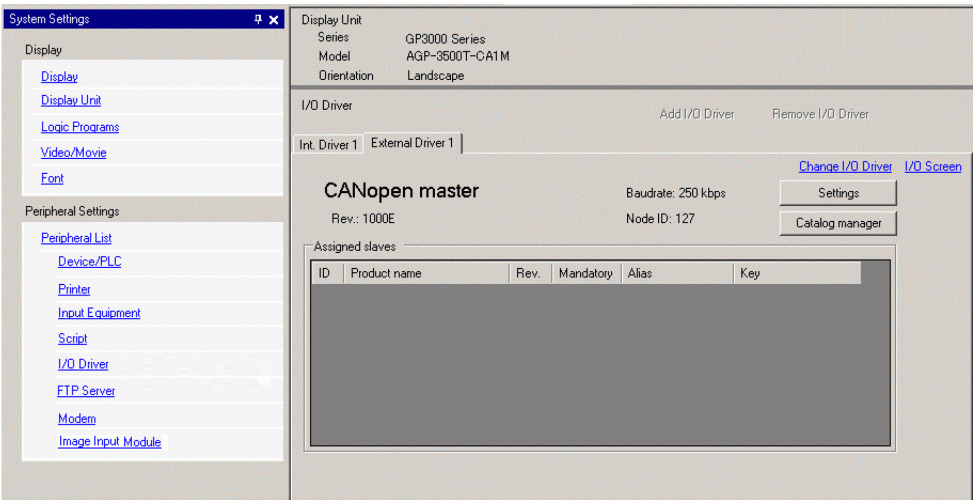
NOTE

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



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

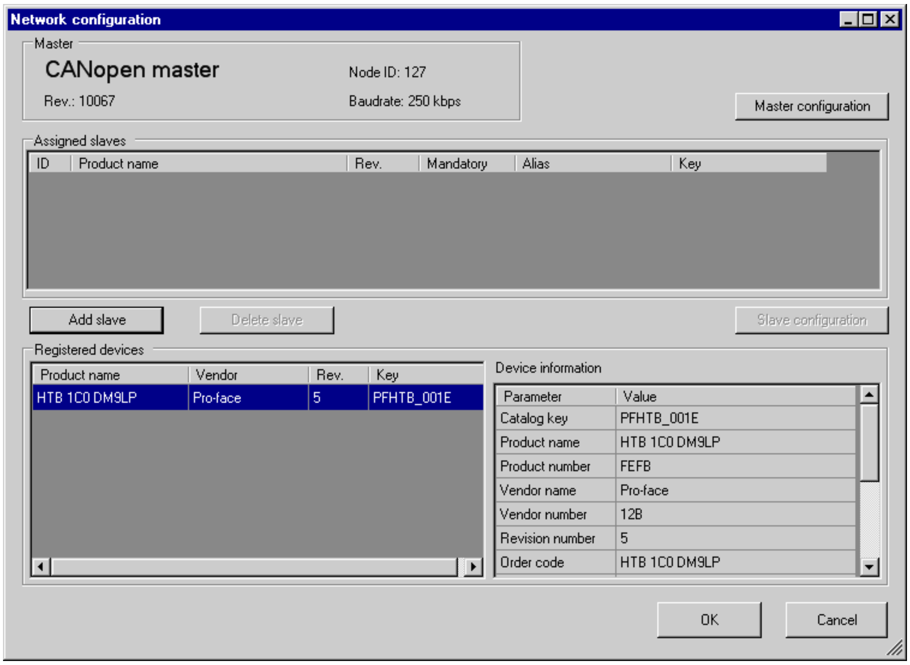
I/O Driver

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

Assigned slaves

ID	Product name	Rev.	Mandatory	Alias	Key

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



Master
CANopen master
Rev.: 10067
Node ID: 127
Baudrate: 250 kbps

Assigned slaves

ID	Product name	Rev.	Mandatory	Alias	Key

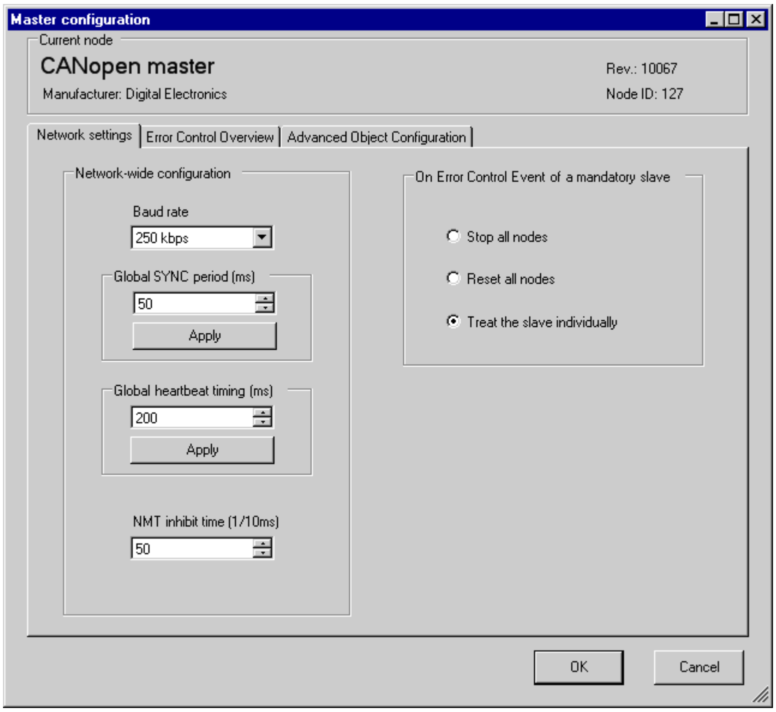
Registered devices

Product name	Vendor	Rev.	Key
HTB 1C0 DM9LP	Pro-face	5	PFHTB_001E

Device information

Parameter	Value
Catalog key	PFHTB_001E
Product name	HTB 1C0 DM9LP
Product number	FEFB
Vendor name	Pro-face
Vendor number	128
Revision number	5
Order code	HTB 1C0 DM9LP

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.: 10067

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

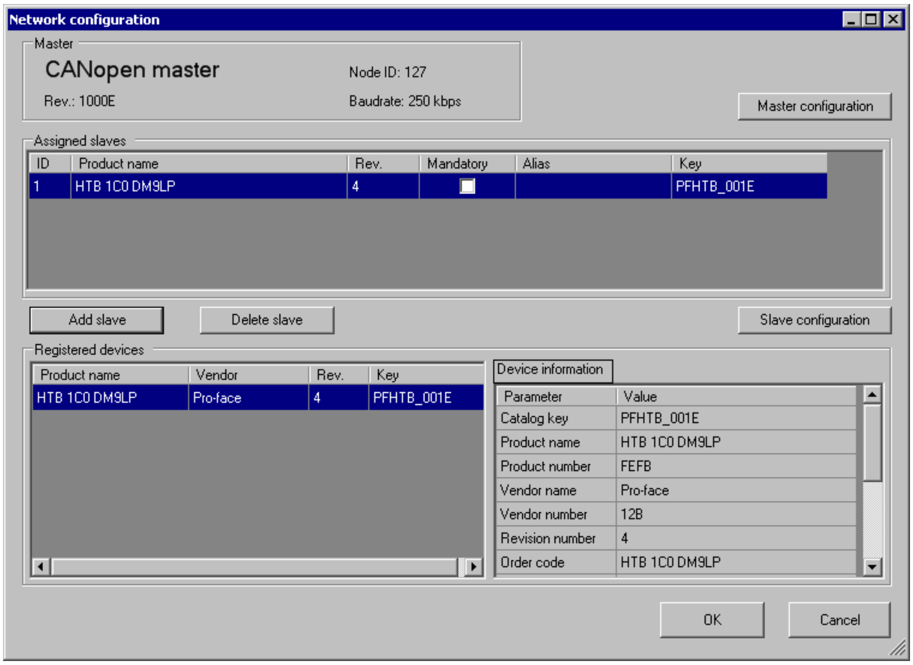
○ Reset all nodes

● Treat the slave individually

NOTE

- Set the slave (HTB) baud rate on the HTB unit itself.

5 Next, add the HTB (slave) to the CANopen network. Select EDS file in HTB in [EDS List] and click [Add Slave].



Network configuration

Master

CANopen master

Node ID: 127

Rev.: 1000E

Baudrate: 250 kbps

Assigned slaves

ID	Product name	Rev.	Mandatory	Alias	Key
1	HTB 1C0 DM9LP	4	□		PFHTB_001E

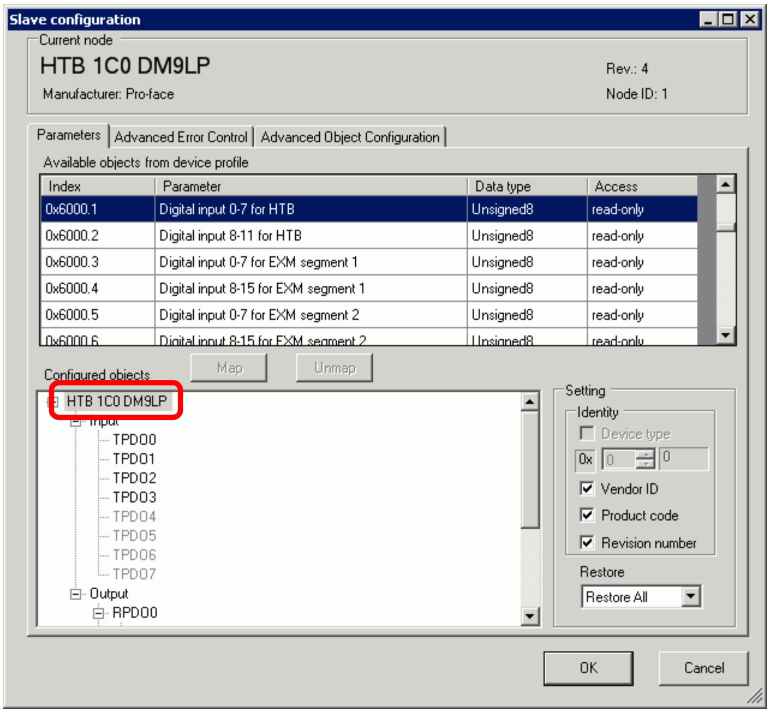
Registered devices

Product name	Vendor	Rev.	Key
HTB 1C0 DM9LP	Pro-face	4	PFHTB_001E

Device information

Parameter	Value
Catalog key	PFHTB_001E
Product name	HTB 1C0 DM9LP
Product number	FEFB
Vendor name	Pro-face
Vendor number	12B
Revision number	4
Order code	HTB 1C0 DM9LP

6 Click [Slave Configuration]. The added slave unit being 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.



Slave configuration

Current node

HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Parameters | Advanced Error Control | Advanced Object Configuration

Available objects from device profile

Index	Parameter	Data type	Access
0x6000.1	Digital input 0-7 for HTB	Unsigned8	read-only
0x6000.2	Digital input 8-11 for HTB	Unsigned8	read-only
0x6000.3	Digital input 0-7 for EXM segment 1	Unsigned8	read-only
0x6000.4	Digital input 8-15 for EXM segment 1	Unsigned8	read-only
0x6000.5	Digital input 0-7 for EXM segment 2	Unsigned8	read-only
0x6000.6	Digital input 8-15 for EXM segment 2	Unsigned8	read-only

Configured objects

Setting

Identity

□ Device type

0x 0 0

☑ Vendor ID

☑ Product code

☑ Revision number

Restore

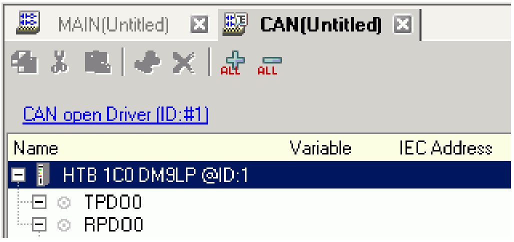
Restore All

NOTE

- When using an EX module, use a sub index object that corresponds to the attachment position of the EX module.
  ☞ "30.7.6 Expanding I/O Using EX Module" (page 30-183)

7 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 information on how to assign variables, refer to the following.

☞ "30.7.3 Allocating I/O (Common)" (page 30-163)



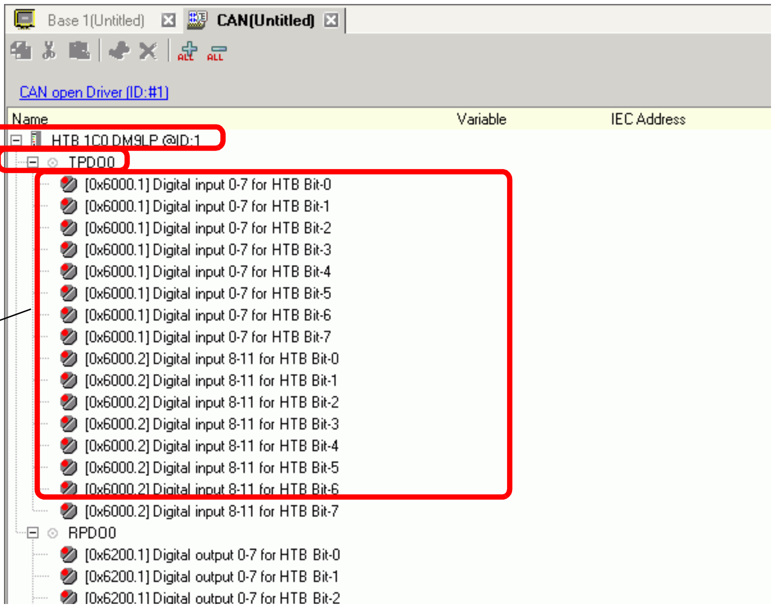
CAN open Driver (ID:#1)

Name	Variable	IEC Address
HTB 1C0 DM9LP @ID:1		
TPDO0		
RPDO0		

8 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 (Common)

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.



```
Base 1(Untitled) ☒   CAN(Untitled) ☒

CAN open Driver [ID:#1]

Name                                                   Variable        IEC Address
⊟ HTB 1C0 DM9LP @ID:1
  ⊟ TPDO0
      [0x6000.1] Digital input 0-7 for HTB Bit-0
      [0x6000.1] Digital input 0-7 for HTB Bit-1
      [0x6000.1] Digital input 0-7 for HTB Bit-2
      [0x6000.1] Digital input 0-7 for HTB Bit-3
      [0x6000.1] Digital input 0-7 for HTB Bit-4
      [0x6000.1] Digital input 0-7 for HTB Bit-5
      [0x6000.1] Digital input 0-7 for HTB Bit-6
      [0x6000.1] Digital input 0-7 for HTB Bit-7
      [0x6000.2] Digital input 8-11 for HTB Bit-0
      [0x6000.2] Digital input 8-11 for HTB Bit-1
      [0x6000.2] Digital input 8-11 for HTB Bit-2
      [0x6000.2] Digital input 8-11 for HTB Bit-3
      [0x6000.2] Digital input 8-11 for HTB Bit-4
      [0x6000.2] Digital input 8-11 for HTB Bit-5
      [0x6000.2] Digital input 8-11 for HTB Bit-6
      [0x6000.2] Digital input 8-11 for HTB Bit-7
  ⊟ RPDO0
      [0x6200.1] Digital output 0-7 for HTB Bit-0
      [0x6200.1] Digital output 0-7 for HTB Bit-1
      [0x6200.1] Digital output 0-7 for HTB Bit-2
```

Displays Slave Product Name, Node ID, or Alias.

PDO packet number.

Mapped objects display as terminals.

NOTE

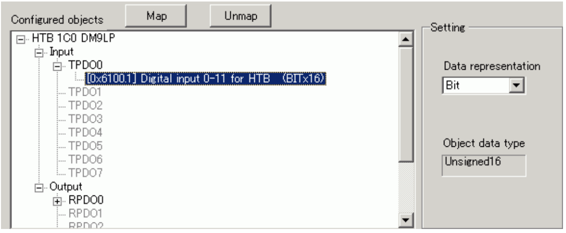
- When mapping TPDO4 or above and RPDO4 or above, the total slave settings are limited to 64.

■ 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 Settings] dialog box in [I/O Driver Settings].

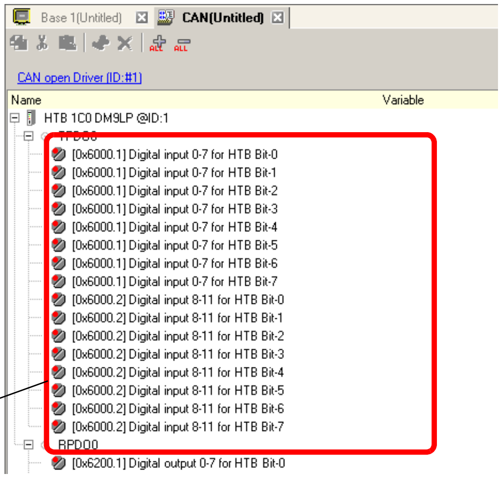
Example 1: Set the Unsigned16 object "Digital input 0-11 for HTB" to "Bit" display

I/O Driver Settings



```
Configured objects   [Map]   [Unmap]                       Setting
⊟ HTB 1C0 DM9LP
  ⊟ Input                                                   Data representation
    ⊟ TPDO0                                                 Bit ▼
        [0x6100.1] Digital input 0-11 for HTB   (BITx16)
      TPDO1
      TPDO2
      TPDO3                                                 Object data type
      TPDO4                                                 Unsigned16
      TPDO5
      TPDO6
      TPDO7
  ⊟ Output
    ⊞ RPDO0
      RPDO1
      RPDO2
```

I/O Screen

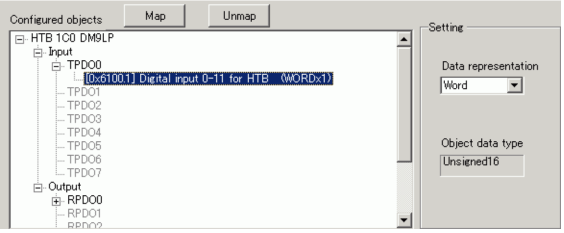


```
Base 1(Untitled) ☒   CAN(Untitled) ☒

CAN open Driver [ID:#1]

Name                                                   Variable
⊟ HTB 1C0 DM9LP @ID:1
  ⊟ TPDO0
      [0x6000.1] Digital input 0-7 for HTB Bit-0
      [0x6000.1] Digital input 0-7 for HTB Bit-1
      [0x6000.1] Digital input 0-7 for HTB Bit-2
      [0x6000.1] Digital input 0-7 for HTB Bit-3
      [0x6000.1] Digital input 0-7 for HTB Bit-4
      [0x6000.1] Digital input 0-7 for HTB Bit-5
      [0x6000.1] Digital input 0-7 for HTB Bit-6
      [0x6000.1] Digital input 0-7 for HTB Bit-7
      [0x6000.2] Digital input 8-11 for HTB Bit-0
      [0x6000.2] Digital input 8-11 for HTB Bit-1
      [0x6000.2] Digital input 8-11 for HTB Bit-2
      [0x6000.2] Digital input 8-11 for HTB Bit-3
      [0x6000.2] Digital input 8-11 for HTB Bit-4
      [0x6000.2] Digital input 8-11 for HTB Bit-5
      [0x6000.2] Digital input 8-11 for HTB Bit-6
      [0x6000.2] Digital input 8-11 for HTB Bit-7
  ⊟ RPDO0
      [0x6200.1] Digital output 0-7 for HTB Bit-0
```

An Unsigned16 object displays as a 16 bit terminal.

Example 2: Set Unsigned16 object "Digital input 0-11 for HTB" as "Word" display settings.

I/O Driver Settings



Configured objects　Map　Unmap

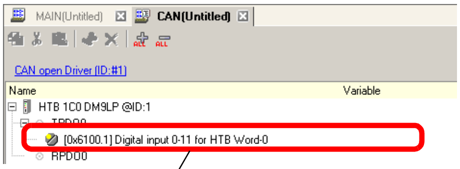
- HTB 1C0 DM9LP
  - Input
    - TPDO0
      - [0x6100.1] Digital input 0-11 for HTB (WORDx1)
    - TPDO1
    - TPDO2
    - TPDO3
    - TPDO4
    - TPDO5
    - TPDO6
    - TPDO7
  - Output
    - RPDO0
    - RPDO1
    - RPDO2

Setting

Data representation: Word

Object data type: Unsigned16

I/O Screen



MAIN(Untitled)　CAN(Untitled)

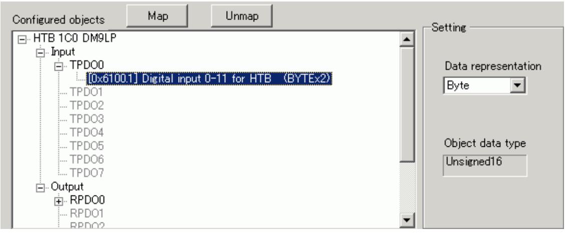
CAN open Driver [ID:#1]

Name	Variable
HTB 1C0 DM9LP @ID:1	
TPDO0	
[0x6100.1] Digital input 0-11 for HTB Word-0	
RPDO0	

Unsigned16 displays as a 1 word terminal.

Example 3: Set the Unsigned16 object "Digital input 0-11 for HTB" to "Byte" display settings.

I/O Driver Settings



Configured objects　Map　Unmap

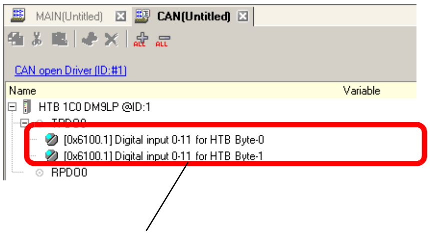
- HTB 1C0 DM9LP
  - Input
    - TPDO0
      - [0x6100.1] Digital input 0-11 for HTB (BYTEx2)
    - TPDO1
    - TPDO2
    - TPDO3
    - TPDO4
    - TPDO5
    - TPDO6
    - TPDO7
  - Output
    - RPDO0
    - RPDO1
    - RPDO2

Setting

Data representation: Byte

Object data type: Unsigned16

I/O Screen



MAIN(Untitled)　CAN(Untitled)

CAN open Driver [ID:#1]

Name	Variable
HTB 1C0 DM9LP @ID:1	
TPDO0	
[0x6100.1] Digital input 0-11 for HTB Byte-0	
[0x6100.1] Digital input 0-11 for HTB Byte-1	
RPDO0	

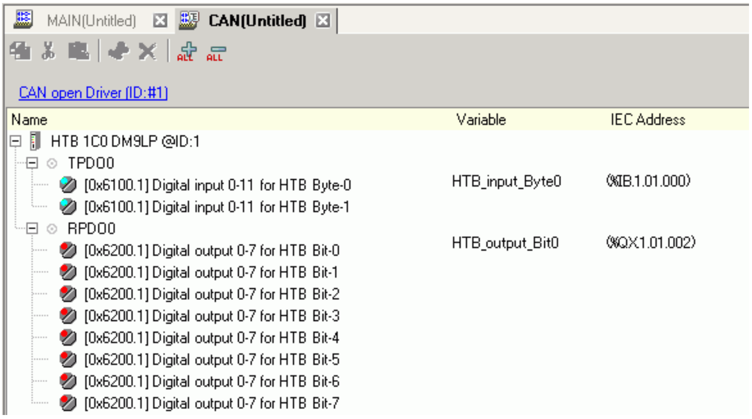
Unsigned16 displays as a 2 byte terminal.

NOTE

- In "Digital input 0-11 for HTB", unsigned16 bit data 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 on each object's terminal. To enter a variable, double-click the [Variable] column of the terminal you wish to allocate.



Name	Variable	IEC Address
HTB 1C0 DM9LP @ID:1		
TPDO0		
[0x6100.1] Digital input 0-11 for HTB Byte-0	HTB_input_Byte0	(%IB.1.01.000)
[0x6100.1] Digital input 0-11 for HTB Byte-1		
RPDO0		
[0x6200.1] Digital output 0-7 for HTB Bit-0	HTB_output_Bit0	(%QX1.01.002)
[0x6200.1] Digital output 0-7 for HTB Bit-1		
[0x6200.1] Digital output 0-7 for HTB Bit-2		
[0x6200.1] Digital output 0-7 for HTB Bit-3		
[0x6200.1] Digital output 0-7 for HTB Bit-4		
[0x6200.1] Digital output 0-7 for HTB Bit-5		
[0x6200.1] Digital output 0-7 for HTB Bit-6		
[0x6200.1] Digital output 0-7 for HTB Bit-7		

[IEC Address] appears automatically after entering a variable.

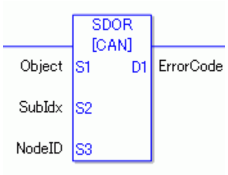
Data Type	Input	Output
Bit	IX	QX
Byte	IB	QB
Word	IW	QW
Dword	ID	QD

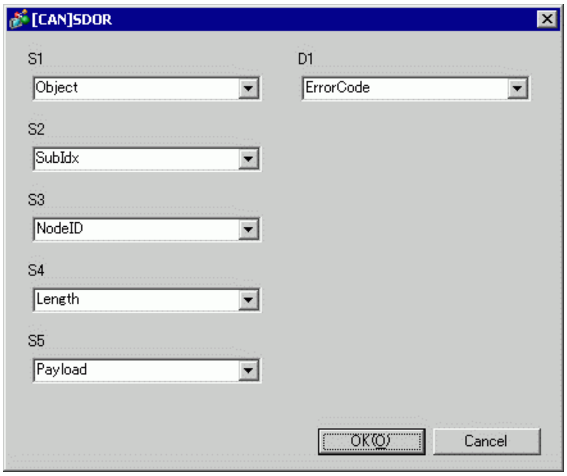
30.7.4 Using I/O Driver Instructions

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.)

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

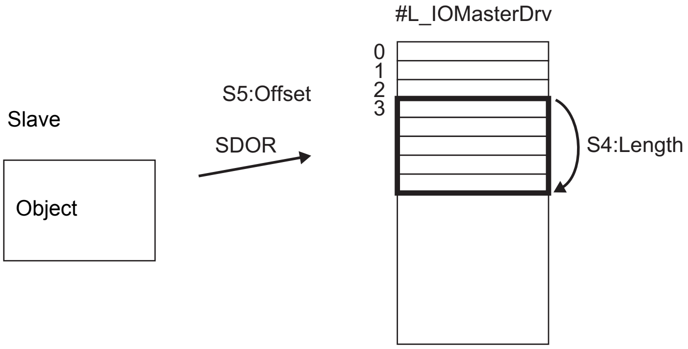
High word			**Low word**	
31	30	16	15	0
Abort code is stored in #L_IOMasterDrv[ ]. 0:Abort code is not stored in #L_IOMasterDrv[ ]. However, if the error includes the abort code, the abort code is stored in D1 (error code). 1:Abort code is stored in #L_IOMasterDrv[ ]. The following error code is stored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

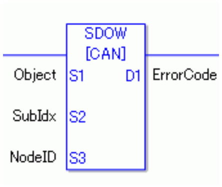


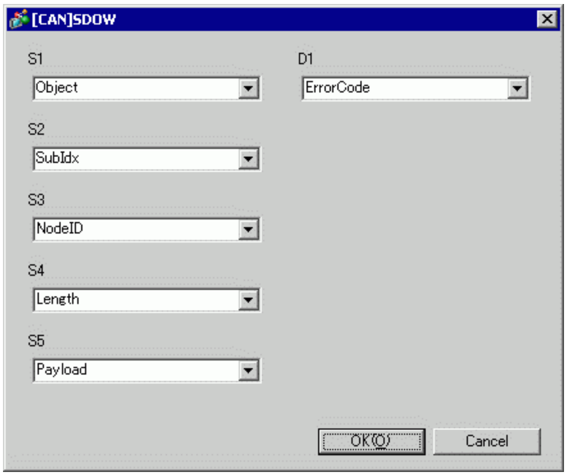
Abort Code

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
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.)

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 save the data to be written and the abort code.

For the lower 16 bits, if you specify an array element in the system variable #L_IOMasterDrv[0] to [255], the location 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 abort codes, see the following:
☞ " ◆ SDOR" (page 30-166)

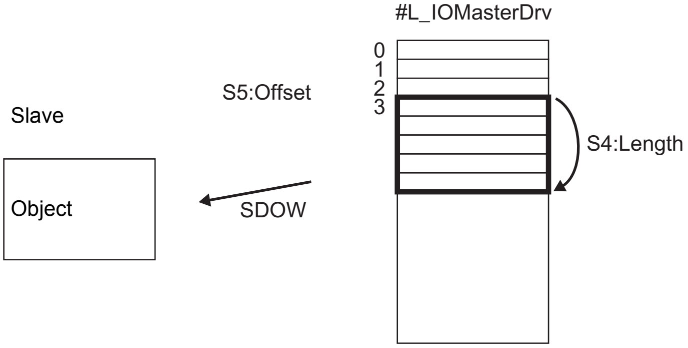
D1:Specify the variable where error codes are stored if an instruction does not act as expected.

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. However, this is ignored when the bit 31 is 0.		Offset #L_IOMasterDrv where data to be written is stored	

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

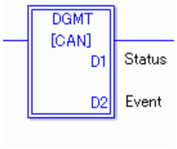
Example: S4=20, S5=3

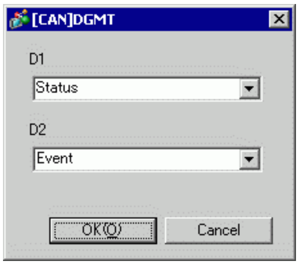


◆ DGMT

Reads the state on the master.

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





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	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.

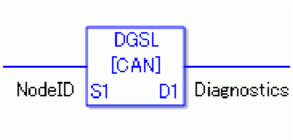
- Event

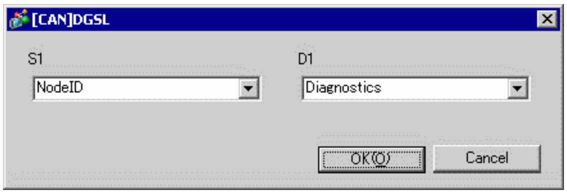
31															16	15															0
Unused (0 fixed)																Event information															

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 to 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.

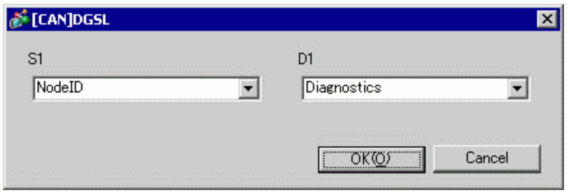
- Slave information

Bit	Description	Introduction
0	Mapping	0: Slave is not assigned to master. 1: Slave is assigned to master.
1	Configuration	0: Not set as slave. Startup is disabled. 1: Set as slave. Startup enabled.
2	Defect	0: Configured Slave. 1: Unconfigured Slave.
3	Emergency message (EMCY)	0: Slave has not sent the emergency message. 1: Slave has sent the emergency message.
4	Operating	0: Slave is not operating. 1: Slave is operating.
5	Stopped	0: Slave is not stopped. Slave is stopped.
6	Preparing for operation	0: Slave is not preparing for operation. 1: Slave is preparing for operation.
7	Inconsistent Concise DCF for one or more slaves	0: Normal concise DCF file. 1: Incorrect concise DCF file.
8	Concise DCF mismatch for one or more slaves	0: 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

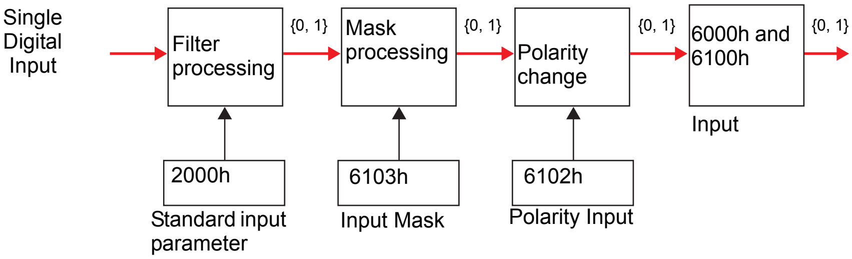
30.7.5 HTB Standard Input/Output

■ Standard Input

Input bits are read into the sub indexes 1 and 2 of object 6000H (8 bit) and sub index 1 of object 6100h (16 bit) in 8 bit and 16 bit units. The following parameters can be changed.

- Standard input parameter (object 2000h)
- Polarity output (object 6102h)
- Input mask (object 6103h)

Object 2000h is used to set input I0 to I7 as digital input (or diagnostic input).

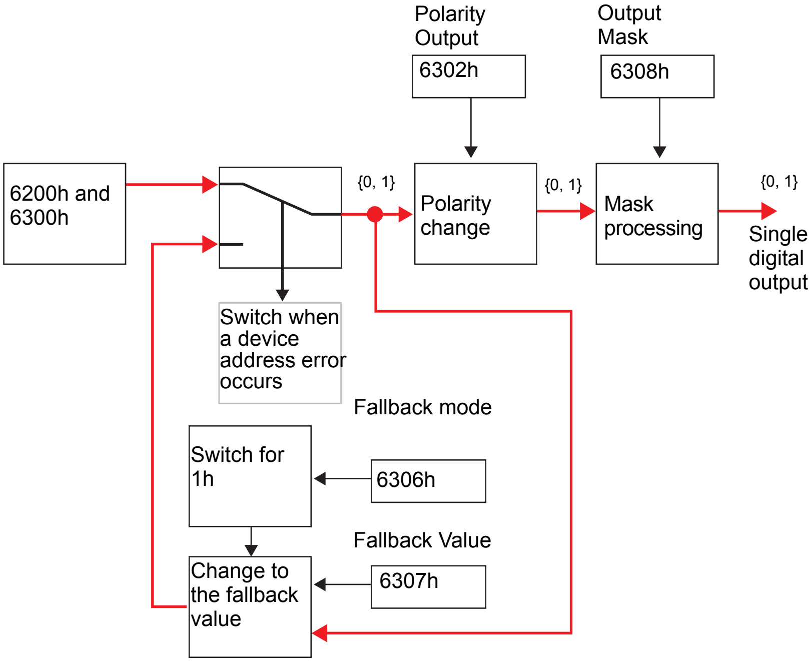


■ Standard Output

Output bits are written into sub index 1 of object 6200H (8 bit) and sub index 1 of object 6300h (16 bit) in 8 bit and 16 bit units. The following parameters can be changed.

- Polarity output (object 6302h)
- Output mask (object 6308h)

When an error such as failed communication with the GP occurs, it switches to the fallback mode.



■ Setup Procedure

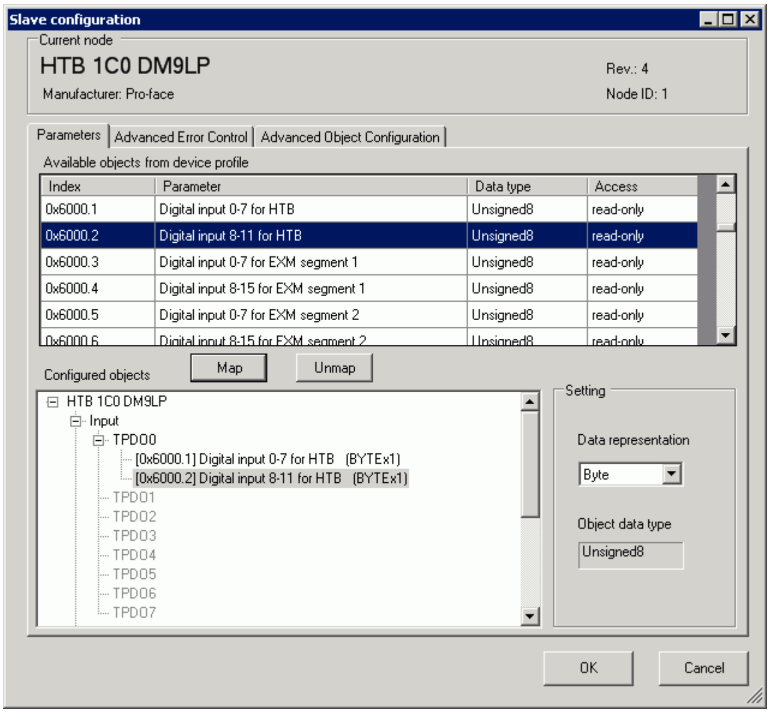
◆ Standard I/O

This section explains PDO mapping in slaves and how to configure objects for cases in which 12-point input and 8-point output are used.

Settings in GP-Pro EX

1 In [IO Driver Settings] click [Settings] and add HTB to the network.
☞ "30.7.2 Setup Procedure" (page 30-159)

2 Click [Slave Settings] while the additional HTB is selected, and the following dialog box appears.



Current node

HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x6000.1	Digital input 0-7 for HTB	Unsigned8	read-only
0x6000.2	Digital input 8-11 for HTB	Unsigned8	read-only
0x6000.3	Digital input 0-7 for EXM segment 1	Unsigned8	read-only
0x6000.4	Digital input 8-15 for EXM segment 1	Unsigned8	read-only
0x6000.5	Digital input 0-7 for EXM segment 2	Unsigned8	read-only
0x6000.6	Digital input 8-15 for EXM segment 2	Unsigned8	read-only

Configured objects

```
⊟ HTB 1C0 DM9LP
  ⊟ Input
    ⊟ TPDO0
        [0x6000.1] Digital input 0-7 for HTB   (BYTEx1)
        [0x6000.2] Digital input 8-11 for HTB   (BYTEx1)
      TPDO1
      TPDO2
      TPDO3
      TPDO4
      TPDO5
      TPDO6
      TPDO7
```

Setting

Data representation: Byte

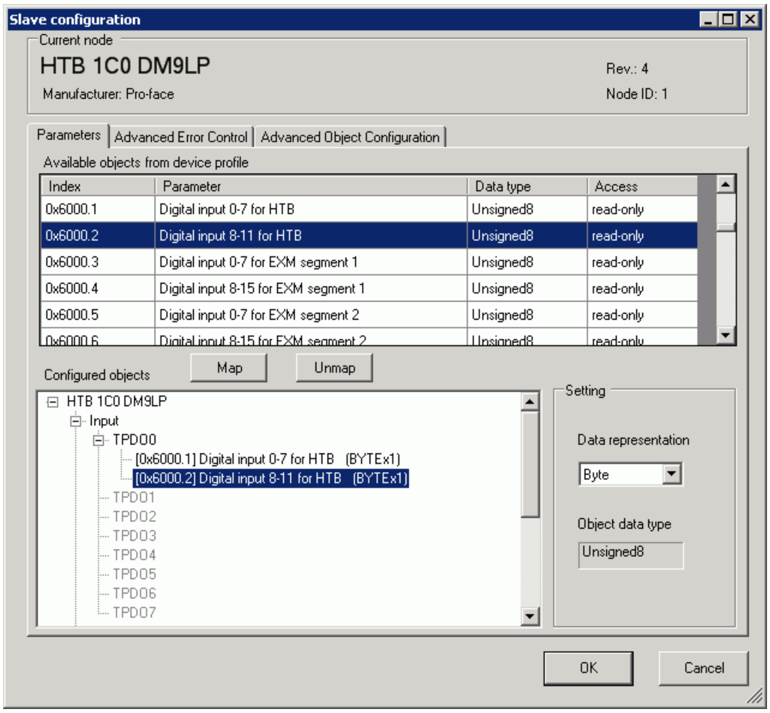
Object data type: Unsigned8

3 While [TPDO0] is selected, select [0x6000.1] and then click the [Map] button.

4 Map [0x6000.2] to [TPDO0].

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

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



HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x6000.1	Digital input 0-7 for HTB	Unsigned8	read-only
0x6000.2	Digital input 8-11 for HTB	Unsigned8	read-only
0x6000.3	Digital input 0-7 for EXM segment 1	Unsigned8	read-only
0x6000.4	Digital input 8-15 for EXM segment 1	Unsigned8	read-only
0x6000.5	Digital input 0-7 for EXM segment 2	Unsigned8	read-only
0x6000.6	Digital input 8-15 for EXM segment 2	Unsigned8	read-only

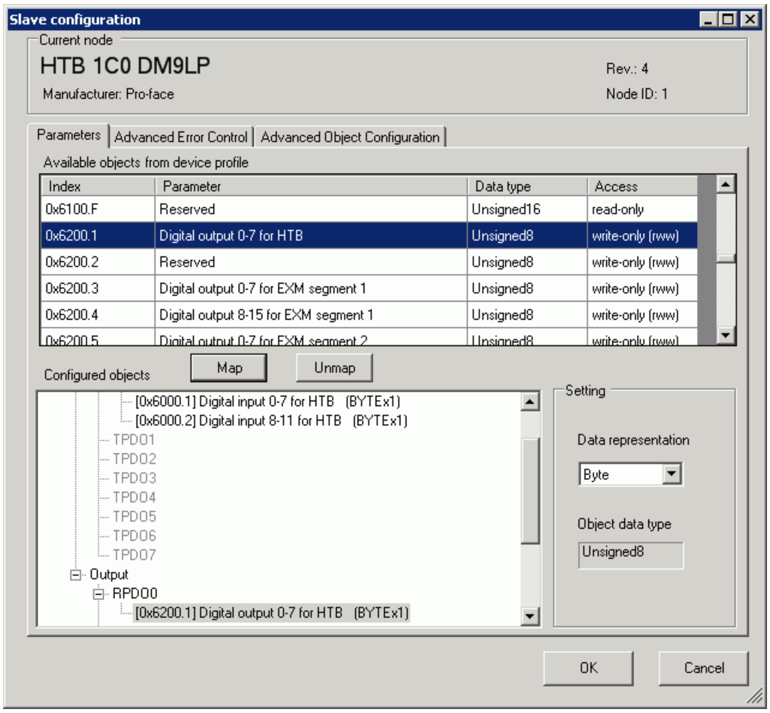
Configured objects

```
HTB 1C0 DM9LP
  Input
    TPDO0
      [0x6000.1] Digital input 0-7 for HTB   (BYTEx1)
      [0x6000.2] Digital input 8-11 for HTB   (BYTEx1)
    TPDO1
    TPDO2
    TPDO3
    TPDO4
    TPDO5
    TPDO6
    TPDO7
```

Data representation: Byte

Object data type: Unsigned8

7 Open the PRDO0 and map [0x6200.1] to [PRDO0].
To change the data type of the object in RPDO, select the object and then and select [Data representation] in [Settings]. For example, Bit



HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x6100.F	Reserved	Unsigned16	read-only
0x6200.1	Digital output 0-7 for HTB	Unsigned8	write-only (rww)
0x6200.2	Reserved	Unsigned8	write-only (rww)
0x6200.3	Digital output 0-7 for EXM segment 1	Unsigned8	write-only (rww)
0x6200.4	Digital output 8-15 for EXM segment 1	Unsigned8	write-only (rww)
0x6200.5	Digital output 0-7 for EXM segment 2	Unsigned8	write-only (rww)

Configured objects

```
      [0x6000.1] Digital input 0-7 for HTB   (BYTEx1)
      [0x6000.2] Digital input 8-11 for HTB   (BYTEx1)
    TPDO1
    TPDO2
    TPDO3
    TPDO4
    TPDO5
    TPDO6
    TPDO7
  Output
    RPDO0
      [0x6200.1] Digital output 0-7 for HTB   (BYTEx1)
```

Data representation: Byte

Object data type: Unsigned8

NOTE

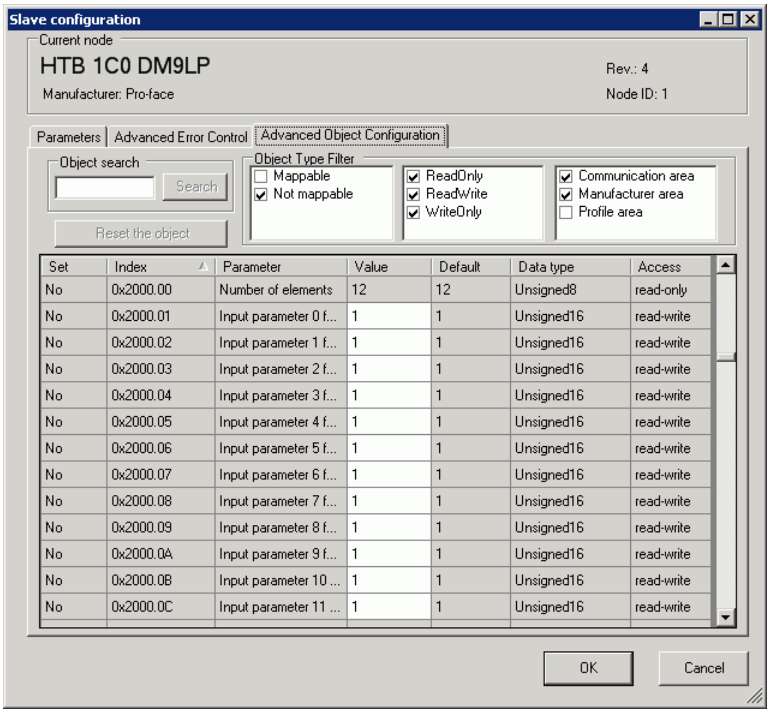
- Map the input data from HTB (CANopen slave) to the GP (CANopen master) for TPDO, and map the object relating to the output data from the GP to HTB for RPDO.
- When mapping TPDO4 or above and RPDO4 or above, the total slave settings are limited to 64.

8 Open the [Advanced Object Configuration] tab to configure the object values. Set the input filter (0x2000.1 to 0x2000.C), the fallback mode (0x6306.1) and the fallback value (0x6307.1) as required and click [OK] to close the dialog.

NOTE

- For details of objects, refer to the following.

☞ "30.7.10 List of HTB Objects" (page 30-232)



Slave configuration

Current node

HTB 1C0 DM9LP

Rev.: 4

Manufacturer: Pro-face

Node ID: 1

□ Mappable
☑ Not mappable
☑ ReadOnly
☑ ReadWrite
☑ WriteOnly
☑ Communication area
☑ Manufacturer area
□ Profile area

Set	Index	Parameter	Value	Default	Data type	Access
No	0x2000.00	Number of elements	12	12	Unsigned8	read-only
No	0x2000.01	Input parameter 0 f...	1	1	Unsigned16	read-write
No	0x2000.02	Input parameter 1 f...	1	1	Unsigned16	read-write
No	0x2000.03	Input parameter 2 f...	1	1	Unsigned16	read-write
No	0x2000.04	Input parameter 3 f...	1	1	Unsigned16	read-write
No	0x2000.05	Input parameter 4 f...	1	1	Unsigned16	read-write
No	0x2000.06	Input parameter 5 f...	1	1	Unsigned16	read-write
No	0x2000.07	Input parameter 6 f...	1	1	Unsigned16	read-write
No	0x2000.08	Input parameter 7 f...	1	1	Unsigned16	read-write
No	0x2000.09	Input parameter 8 f...	1	1	Unsigned16	read-write
No	0x2000.0A	Input parameter 9 f...	1	1	Unsigned16	read-write
No	0x2000.0B	Input parameter 10 ...	1	1	Unsigned16	read-write
No	0x2000.0C	Input parameter 11 ...	1	1	Unsigned16	read-write

9 Click [I/O Screen] in [I/O Driver Settings] or select [I/O Screen] in the [Screen List] window on Workspace to assign a variable to each of the mapped objects. For information on how to assign variables, refer to the following.
☞ "30.7.3 Allocating I/O (Common)" (page 30-163)

10 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.

■ Details of Objects Used in Standard I/O

The following table shows I/O read/write actions of an HTB.

Object (Hex)	Sub index	Bit	Description	Parameter
6000	1	0 ... 7	Input 0 ... Input 7	8 bit word
	2	0 ... 3	Input 8 ... Input 11	8 bit word
		4 to 7	Unused	
6100	1	0 ... 11	Input 0 ... Input 11	16 bit word
		12 to 15	Unused	
6200	1	0 ... 7	Output 0 ... Output 7	8 bit word
6300	1	0 ... 7	Output 0 ... Output 7	16 bit word
		8 to 15	Unused	

■ Details of I/O Setting Objects

Object (Hex)	Sub index	Bit	Description	Parameter
2000	1 ... 12	-	Input filter 0 ... Input filter 11	None 3ms (initial value) 12ms
6102	1	0 ... 11	Polarity input 0 ... Polarity input 11	0: Normal input 1: Reverse input
		12 to 15	Unused	
6103	1	0 ... 11	Input mask 0 ... Input mask 11	0: Enable mask (initial value) 1: Disable mask
		12 to 15	Unused	
6302	1	0 ... 11	Polarity output 0 ... Polarity output 7	0: Normally open output (NO) (initial value) 1: Normally closed output (NC)
		8 to 15	Unused	
6306	1	0 ... 7	Fallback mode output 0*1 ... Fallback mode output 7	0: Retentive state 1: Fallback value enabled (initial value)
		8 to 15	Unused	
6307	1	0 ... 7	Fallback value output 0 ... Fallback value output 7	Fallback to 0 (initial value) Fallback to 1
		8 to 15	Unused	
6308	1	0 ... 7	Output mask 0 ... Output mask 7	0: Retentive State (initial value) 1: Value of object (Hex) 6300
		8 to 15	Unused	

*1 Fallback mode is a feature that can define the state of output when a communication error occurs (For example, turning ON whenever a communication error occurs).

NOTE

- When fallback mode (6306h) is 0, fallback value (6307h) is ignored.

30.7.6 Expanding I/O Using EX Module

A maximum of seven units of EX modules can be attached to HTB. In addition to expansion of I/O for standard input and output, analog I/O can be used.
You can use the following EX module models and features.

◆ EX Modules: Models and Features

Feature	Type	Description	Browse to
Standard Input	EXM-DDI8DT	DIO Module: 8-input sink/source	☞ " ◆ Standard Input (Model: EXM-DDI8DT)" (page 30-184)
	EXM-DDI16DT	DIO Module: 16-input sink/source	☞ " ◆ Standard Input (Model: EXM-DDI16DT)" (page 30-185)
Standard Output	EXM-DDO8UT	DIO module: 8-output sink	☞ " ◆ Standard Output (Model: EXM-DDO8UT, EXM-DDO8TT and EXM-DRA8RT)" (page 30-185)
	EXM-DDO8TT	DIO module: 8-output source	
	EXM-DRA8RT	DIO module: 8-output relay	
	EXM-DDO16UK	DIO Module: 16-output sink	☞ " ◆ Standard Output (Model: EXM-DDO16UK, EXM-DDO16TK and EXM-DRA16RT)" (page 30-186)
	EXM-DDO16TK	DIO Module: 16-output source	
	EXM-DRA16RT	DIO Module: 16-output relay	
Standard I/O	EXM-DMM8DRT	DIO Module: 4-input sink/source DIO Module: 4-output relay	☞ " ◆ Standard Input and Output (Model: EXM-DMM8DRT)" (page 30-187)
Analog Input	EXM-AMI2HT	Analog Module: 2-input voltage/current	☞ " ◆ Analog Input (Model: EXM-AMI2HT)" (page 30-194)
Analog Output	EXM-AMO1HT	Analog Module: 1-output voltage/current	☞ " ◆ Analog Output (Model: EXM-AMO1HT)" (page 30-195)
Analog I/O	EXM-AMM3HT	Analog Module: 2-input voltage/current Analog Module: 1-output voltage/current	☞ " ◆ Analog I/O (Model: EXM-AMM3HT)" (page 30-196)
	EXM-ALM3LT	Analog Module: 2-input thermocouple/PT100 Analog Module: 1-output voltage/current	☞ " ◆ Analog I/O (Model: EXM-ALM3LT)" (page 30-198)

NOTE

- While HTB is powered ON, such as when it's running, do not install or remove HTB or the EX module.

■ Standard Input/Output

The following describes CANopen objects used in DIO for EX modules. Sub index number "n" (2 or more) is determined by the position of the module connected to HTB and the object that is used.

NOTE

- For details on the relationship between sub index "n" and attached position of EX module, refer to the following:

☞ " ■ Sub Index" (page 30-203)

◆ Standard Input (Model: EXM-DDI8DT)

The DIO input module defines active state and input mask using configuration objects.

Object (Hex)	Sub index	Bit	Description	Parameter
6000	n	0 ... 7	Input 0 ... Input 7	8 bit word
6100	n	0 ... 7	Input 0 ... Input 7	16 bit word
		8 to 15	Unused	
6102	n	0 ... 7	Polarity input 0 ... Polarity input 7	0: Normal input (initial value) 1: Reverse input
		8 to 15	Unused	
6103	n	0 ... 7	Input mask 0 ... Input mask 7	0: Enable mask (initial value) 1: Disable mask
		8 to 15	Unused	

◆ Standard Input (Model: EXM-DDI16DT)

The DIO input module defines active state and input mask using configuration objects.

Object (Hex)	Sub index	Bit	Description	Parameter
6000	n	0 ... 7	Input 0 ... Input 7	8 bit word
	n+1	0 ... 7	Input 8 ... Input 15	8 bit word
6100	n	0 ... 15	Input 0 ... Input 15	16 bit word
6102	n	0 ... 15	Polarity input 0 ... Polarity input 15	0: Normal input (initial value) 1: Reverse input
6103	n	0 ... 15	Input mask 0 ... Input mask 15	0: Enable mask (initial value) 1: Disable mask

◆ Standard Output (Model: EXM-DDO8UT, EXM-DDO8TT and EXM-DRA8RT)

The DIO output module defines fallback mode and values using configuration objects.

Object (Hex)	Sub index	Bit	Description	Parameter
6200	n	0 ... 7	Output 0 ... Output 7	8 bit word
6300	n	0 ... 7	Output 0 ... Output 7	16 bit word
		8 to 15	Unused	
6302	n	0 ... 7	Polarity output 0 ... Polarity output 7	0: Normally open output (NO) (initial value) 1: Normally closed output (NC)
		8 to 15	Unused	
6306	n	0 ... 7	Fallback mode output 0 ... Fallback mode output 7	0: Retentive State 1: Fallback value enabled (initial value) (set to the set fallback value)
		8 to 15	Unused	

Continued

Object (Hex)	Sub index	Bit	Description	Parameter
6307	n	0 ... 7	Fallback value output 0 ... Fallback value output 7	0: Fallback to 0 (initial value) 1: Fallback to 1
		8 to 15	Unused	
6308	n	0 ... 7	Output mask 0 ... Output mask 7	0: Retentive State (initial value) 1: Value of the object (Hex) 6300
		8 to 15	Unused	

◆ Standard Output (Model: EXM-DDO16UK, EXM-DDO16TK and EXM-DRA16RT)

The DIO output module defines fallback mode and values using configuration objects.

Object (Hex)	Sub index	Bit	Description	Parameter
6200	n	0 ... 7	Output 0 ... Output 7	8 bit word
	n+1	0 ... 7	Output 8 ... Output 15	8 bit word
6300	n	0 ... 15	Output 0 ... Output 15	16 bit word
6302	n	0 ... 15	Polarity output 0 ... Polarity output 15	0: Normally open output (NO) (initial value) 1: Normally closed output (NC)
6306	n	0 ... 15	Fallback mode output 0 ... Fallback mode output 15	0: Retentive State 1: Fallback value enabled (initial value)
6307	n	0 ... 15	Fallback value output 0 ... Fallback value output 15	0: Fallback to 0 (default) 1: Fallback to 1
6308	n	0 ... 15	Output mask 0 ... Output mask 15	0: Retentive State (initial value) 1: Value of the object (Hex) 6300

◆ Standard Input and Output (Model: EXM-DMM8DRT)

The DIO input and output module defines fallback mode and values using configuration objects.

Standard input

Object (Hex)	Sub index	Bit	Description	Parameter
6000	n	0 ... 3	Input 0 ... Input 3	8 bit word
		4 to 7	Unused	
6100	n	0 ... 3	Input 0 ... Input 3	16 bit word
		4 to 15	Unused	
6102	n	0 ... 3	Polarity input 0 ... Polarity input 3	0: Normal input (initial value) 1: Reverse input
		4 to 15	Unused	
6103	n	0 ... 3	Input mask 0 ... Input mask 3	0: Enable mask (initial value) 1: Disable mask
		4 to 15	Unused	

Unused

Object (Hex)	**Sub index**	**Bit**	**Description**	**Parameter**
6200	n	0 ... 3	Output 0 ... Output 3	8 bit word
		4 to 7	Unused	
6300	n	0 ... 3	Output 0 ... Output 3	16 bit word
		4 to 15	Unused	

Continued

Object (Hex)	Sub index	Bit	Description	Parameter
6302	n	0 ... 3	Polarity output 0 ... Polarity output 3	0: Normally open output (NO) (initial value) 1: Normally closed output (NC)
		4 to 15	Unused	
6306	n	0 ... 3	Fallback mode output 0 ... Fallback mode output 3	0: Retentive State 1: Fallback value enabled (initial value)
		4 to 15	Unused	
6307	n	0 ... 3	Fallback value output 0 ... Fallback value output 3	0: Fallback to 0 (initial value) 1: Fallback to 1
		4 to 15	Unused	
6308	n	0 ... 3	Output mask 0 ... Output mask 3	0: Retentive State (initial value) 1: Value of the object (Hex) 6300
		4 to 15	Unused	

◆ Setup Procedure

The following information explains how to map PDO on slave side and how to set objects when adding EX module "EXM-DMM8DRT" to HTB (using 12-input points and 8-output points).

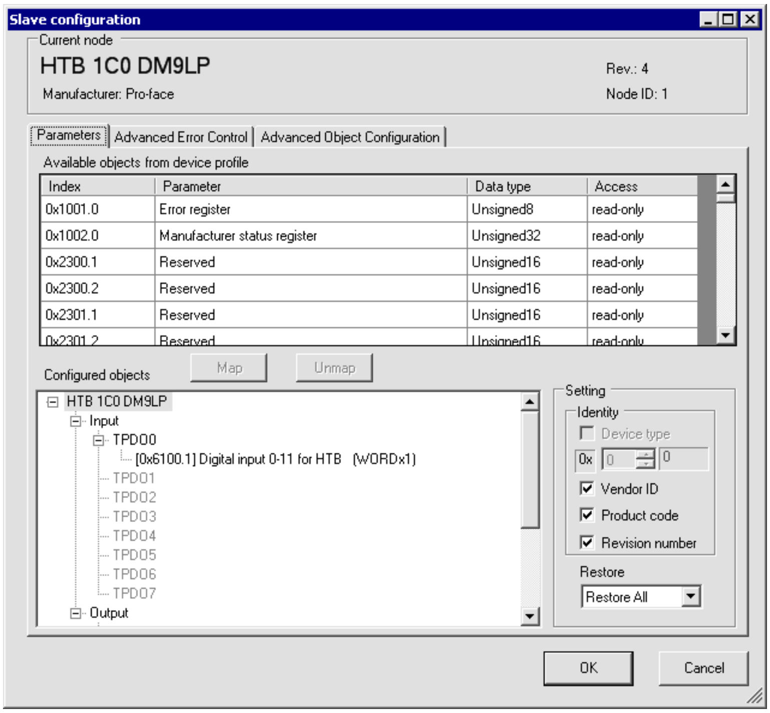
Settings in GP-Pro EX

1 In [IO Driver Settings] click [Settings] and add HTB to the network.

☞ "30.7.2 Setup Procedure" (page 30-159)

2 Set standard input and output of HTB.

☞ "30.7.5 HTB Standard Input/Output ■ Setup Procedure" (page 30-177)



Current node

HTB 1C0 DM9LP

Rev.: 4

Manufacturer: Pro-face

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x1001.0	Error register	Unsigned8	read-only
0x1002.0	Manufacturer status register	Unsigned32	read-only
0x2300.1	Reserved	Unsigned16	read-only
0x2300.2	Reserved	Unsigned16	read-only
0x2301.1	Reserved	Unsigned16	read-only
0x2301.2	Reserved	Unsigned16	read-only

Configured objects

```
⊟ HTB 1C0 DM9LP
   ⊟ Input
      ⊟ TPDO0
         [0x6100.1] Digital input 0-11 for HTB   (WORDx1)
        TPDO1
        TPDO2
        TPDO3
        TPDO4
        TPDO5
        TPDO6
        TPDO7
   ⊟ Output
```

Setting

Identity

□ Device type

0x 0 0

☑ Vendor ID

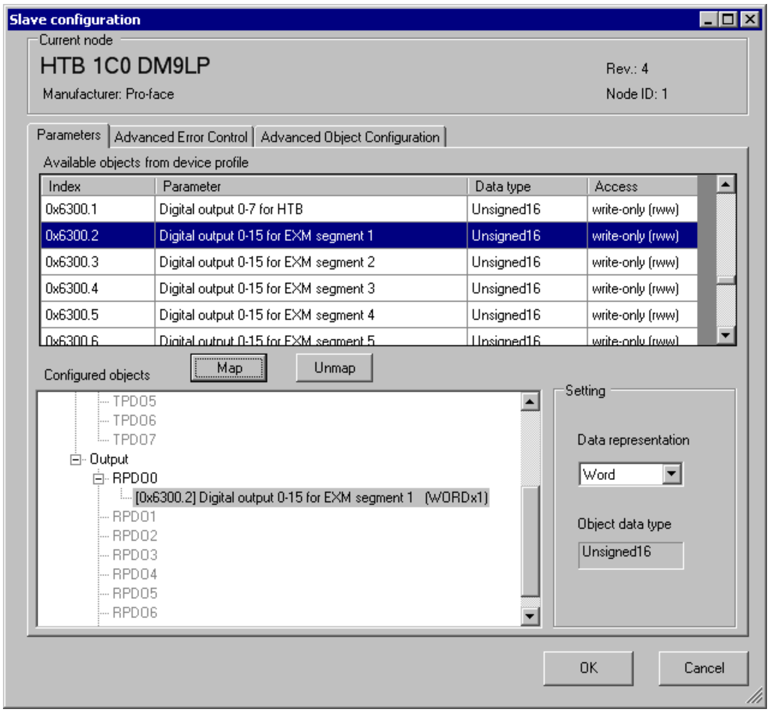
☑ Product code

☑ Revision number

Restore

Restore All

3 Set the standard I/O of EX module. Select the input value object "0x6100.2" to map to TPDO. In [Settings], select the data type. Then map "0x6300.2" to RPDO and select the data type.



Current node

HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x6300.1	Digital output 0-7 for HTB	Unsigned16	write-only (rww)
0x6300.2	Digital output 0-15 for EXM segment 1	Unsigned16	write-only (rww)
0x6300.3	Digital output 0-15 for EXM segment 2	Unsigned16	write-only (rww)
0x6300.4	Digital output 0-15 for EXM segment 3	Unsigned16	write-only (rww)
0x6300.5	Digital output 0-15 for EXM segment 4	Unsigned16	write-only (rww)
0x6300.6	Digital output 0-15 for EXM segment 5	Unsigned16	write-only (rww)

Configured objects

```
    TPDO5
    TPDO6
    TPDO7
  Output
    RPDO0
      [0x6300.2] Digital output 0-15 for EXM segment 1   (WORDx1)
    RPDO1
    RPDO2
    RPDO3
    RPDO4
    RPDO5
    RPDO6
```

Setting

Data representation: Word

Object data type: Unsigned16

NOTE

- Map the input data from HTB (CANopen slave) to the GP (CANopen master) for TPDO, and map the object relating to the output data from the GP to HTB for RPDO.
- Map error register "0x1001.1", status register "0x1002.1" and module diagnosis "0x3000.1" and "0x3000.2" to TPDO as required.

4 Open [Advanced Object Configuration] tab and configure detailed settings of the object. Set fallback mode (0x6306.1 and 0x6306.2) and fallback value (0x6307.1 and 0x6307.2) as required and click [OK] to close the dialog box.

NOTE

- For details of objects, refer to the following.
  ☞ "30.7.10 List of HTB Objects" (page 30-232)

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

☞ "30.7.3 Allocating I/O (Common)" (page 30-163)

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

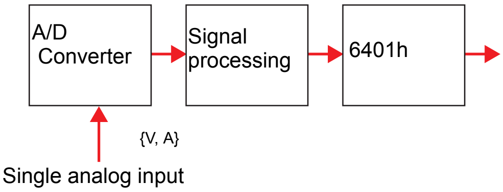
NOTE

- To check set values, use SDOR instructions.

■ Analog I/O

Analog input

Analog input is read by the unit of 16 bit word. Each channel value is stored in sub index of the object 6401h.

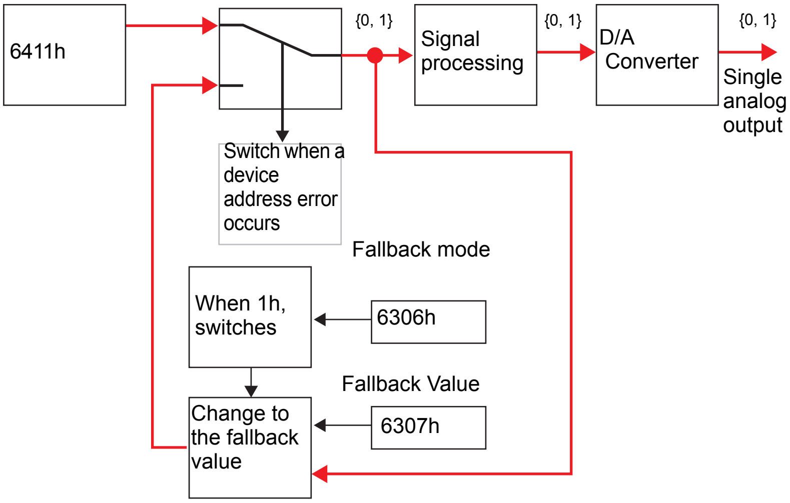


NOTE

- Data is not updated unless "PDO transmission enabled/disabled" (6423h) is set to 1 (enabled).

Analog output

Analog output is written by the unit of 16 bit word. Output bit is stored in sub index of the object 6411h. It is switched to fallback mode when an error occurs.



Analog I/O Settings - Common Objects (Model: EXM-AMI2HT, EXM-AMO1HT, EXM-AMM3HT and EXM-ALM3LT)

The following describes CANopen objects used in analog I/O for EX modules. Sub index number "n" (2 or more) is determined by the position of the module connected to HTB and the object that is used.

NOTE

- For details on the relationship between sub index "n" and attached position of EX module, refer to the following:

☞ " ■ Sub Index" (page 30-203)

EX modules that support analog have different setup objects between models. The following common values are used in all channel and registries.

Analog input

Object (Hex)	Description	Parameter
2100	Data Type	0: Unused 2: Current (4 to 20mA) 3: Voltage (0 to 10V) 5: Thermocouple (K type) 6: Thermocouple (J type) 7: Thermocouple (T type) 8: PT100
2101	Data range	0: Fixed 1: User setting 2: Celsius 3: Fahrenheit
2102	Lower limit (when 2101h = 1)	Lowest value
2103	Upper limit (when 2101h = 1)	Highest value
6401	Read value	
6422	PDO transmission channel number	Indicates the channel in which active bit caused an event.
6423	PDO transmission enabled/ disabled	0: Disable PDO transmission 1: Enable PDO transmission Transmit the command to all the channels.

Analog output

Object (Hex)	Description	Parameter
2200	Data Type	0: Unused 2: Current (4 to 20mA) 3: Voltage (0 to 10V) 5: Thermocouple (K type) 6: Thermocouple (J type) 7: Thermocouple (T type) 8: PT100
2201	Data range	0: Fixed 1: User setting 2: Celsius 3: Fahrenheit
2202	Lower limit (when 2201h = 1)	Lowest value
2203	Upper limit (when 2201h = 1)	Highest value
6411	Written value	
6443	Fallback mode	0: Fallback 1: Retentive State
6444	Fallback Value	Fallback Value

◆ Analog Input (Model: EXM-AMI2HT)

The analog input module defines input range and customizable value using configuration objects.

NOTE • Analog channels can be mixed.

Channel	Object (Hex)	Sub index	Description	Parameter
CH1	2100	n	Data Type	0: Unused 2: Current (4 to 20mA) 3: Voltage (0 to 10V)
	2101	n	Data range	0: Fixed 1: User setting
	2102	n	Lower limit (when 2101h = 1)	Lowest value
	2103	n	Upper limit (when 2101h = 1)	Highest value
	6401	n	Read value	Input
	6422	1	PDO transmission channel number	Indicates the channel in which active bit caused an event.
	6423	0	PDO transmission enabled/disabled	0: Disable PDO transmission 1: Enable PDO transmission
CH2	Same as CH1. Sub index is n+1.			

◆ Analog Output (Model: EXM-AMO1HT)

The analog output module defines output range, customizable value, and fallback value using configuration object.

Channel	Object (Hex)	Sub index	Description	Parameter
CH1	2200	n	Data Type	0: Unused 2: Current (4 to 20mA) 3: Voltage (0 to 10V)
	2201	n	Data range	0: Fixed 1: User setting
	2202	n	Lower limit (when 2101h = 1)	Lowest value
	2203	n	Upper limit (when 2101h = 1)	Highest value
	6411	n	Written value	Output
	6443	n	Fallback mode	0: Retentive State 1: Fallback
	6444	n	Fallback Value (when 6443h = 1)	Fallback Value

◆ Analog I/O (Model: EXM-AMM3HT)

The analog I/O module defines the range of I/O, customizable value, fallback mode, and fallback value using configuration objects.

NOTE • Analog channels can be mixed.

Channel	Object (Hex)	Sub index	Description	Parameter	Initial value of the parameter
CH1 (Input)	2100	n	Data Type	0: Unused 2: Current (4 to 20mA) 3: Voltage (0 to 10V)	0
	2101	n	Data range	0: Fixed 1: User setting	1
	2102	n	Lower limit (when 2101h = 1)	Lowest value	0
	2103	n	Upper limit (when 2101h = 1)	Highest value	7FFFh
	6401	n	Read value	Input	0
	6422	1	PDO transmission channel number		0
	6423	0	PDO transmission enabled/disabled	0: Disable PDO transmission 1: Enable PDO transmission	0
CH2 (Input)	Same as CH1 (Input). Sub index is n+1.				

Channel	Object (Hex)	Sub index	Description	Parameter	Initial value of the parameter
CH1 (Output)	2200	n	Data Type	0: Unused 2: Current (4 to 20mA) 3: Voltage (0 to 10V)	0
	2201	n	Data range	0: Fixed 1: User setting	1
	2202	n	Lower limit (when 2201h = 1)	Lowest value	0
	2203	n	Upper limit (when 2201h = 1)	Highest value	7FFFh
	6411	n	Written value	Output	0
	6443	n	Fallback mode	0: Retentive State 1: Fallback	1
	6444	n	Fallback Value (when 6443h = 1)	Fallback Value	0

◆ Analog I/O (Model: EXM-ALM3LT)

The temperature input module uses configuration objects to determine the thermocouple range and customizable values. The analog output module defines the output range, customizable values, fallback mode, and fallback value.

NOTE

- Analog channels can be mixed.

Channel	Object (Hex)	Sub index	Description	Parameter	Initial value of the parameter
CH1 (Input)	2100	n	Data Type	0: Unused 5: Thermocouple (K type) 6: Thermocouple (J type) 7: Thermocouple (T type) 8: PT100	0
	2101	n	Data range	2: Celsius 3: Fahrenheit	1
	6401	n	Read value	Input	0
	6422	1	PDO transmission channel number		0
	6423	0	PDO transmission enabled/disabled	0: Disable PDO transmission 1: Enable PDO transmission	0
CH2 (Input)	Same as CH1 (Input). Sub index is n+1.				

Channel	Object (Hex)	Sub index	Description	Parameter	Initial value of the parameter
CH1 (Output)	2200	n	Data Type	0: Unused 2: Current (4 to 20mA) 3: Voltage (0 to 10V)	0
	2201	n	Data range	0: Fixed 1: User setting	1
	2202	n	Lower limit (when 2201h = 1)	Lowest value	0
	2203	n	Upper limit (when 2201h = 1)	Highest value	7FFFh
	6411	n	Written value	Output	0
	6443	n	Fallback mode	0: Retentive State 1: Fallback	1
	6444	n	Fallback Value (when 6443h = 1)	Fallback Value	0

◆ Setup Procedure

The following describes how to map PDO on slave side and how to set objects when adding EX module "EXM-AMI2HT" and "EXM-AMM3HT" to HTB (using 12-input points and 8-output points).

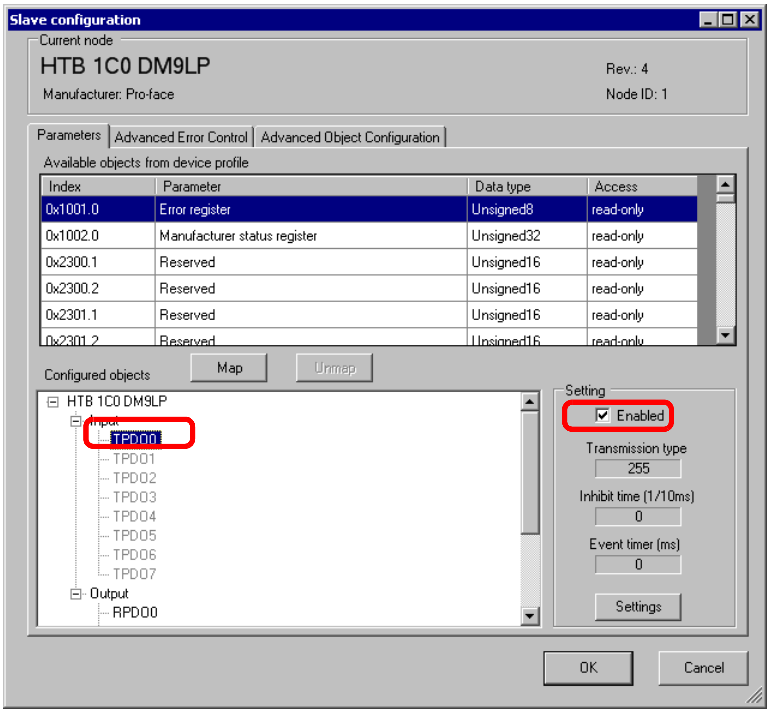
Settings in GP-Pro EX

1 In [IO Driver Settings] click [Settings] and add HTB to the network.

☞ "30.7.2 Setup Procedure" (page 30-159)

2 Set standard input and output of HTB.

☞ "30.7.5 HTB Standard Input/Output ■ Setup Procedure" (page 30-177)



Current node
HTB 1C0 DM9LP
Manufacturer: Pro-face
Rev.: 4
Node ID: 1

Available objects from device profile

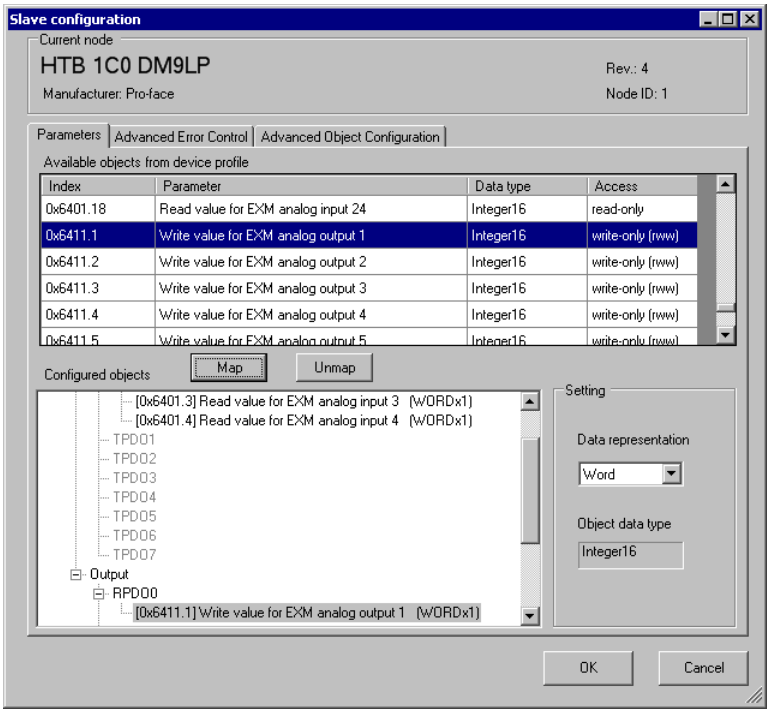
Index	Parameter	Data type	Access
0x1001.0	Error register	Unsigned8	read-only
0x1002.0	Manufacturer status register	Unsigned32	read-only
0x2300.1	Reserved	Unsigned16	read-only
0x2300.2	Reserved	Unsigned16	read-only
0x2301.1	Reserved	Unsigned16	read-only
0x2301.2	Reserved	Unsigned16	read-only

Configured objects

```
⊟ HTB 1C0 DM9LP
   ⊟ Input
      TPDO0
      TPDO1
      TPDO2
      TPDO3
      TPDO4
      TPDO5
      TPDO6
      TPDO7
   ⊟ Output
      RPDO0
```

Setting
☑ Enabled
Transmission type: 255
Inhibit time (1/10ms): 0
Event timer (ms): 0
Settings

3 Set up analog input of EX module. Select the input value object "0x6401.1 to 0x6401.4" to map to TPDO. In [Settings], select the data type. In addition, map "0x6411.1" to RPDO and select the data type.



Slave configuration

Current node

HTB 1C0 DM9LP

Rev.: 4

Manufacturer: Pro-face

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x6401.18	Read value for EXM analog input 24	Integer16	read-only
0x6411.1	Write value for EXM analog output 1	Integer16	write-only (rww)
0x6411.2	Write value for EXM analog output 2	Integer16	write-only (rww)
0x6411.3	Write value for EXM analog output 3	Integer16	write-only (rww)
0x6411.4	Write value for EXM analog output 4	Integer16	write-only (rww)
0x6411.5	Write value for EXM analog output 5	Integer16	write-only (rww)

Configured objects

```
        [0x6401.3] Read value for EXM analog input 3  (WORDx1)
        [0x6401.4] Read value for EXM analog input 4  (WORDx1)
     TPDO1
     TPDO2
     TPDO3
     TPDO4
     TPDO5
     TPDO6
     TPDO7
  Output
     RPDO0
        [0x6411.1] Write value for EXM analog output 1  (WORDx1)
```

Setting

Data representation: Word

Object data type: Integer16

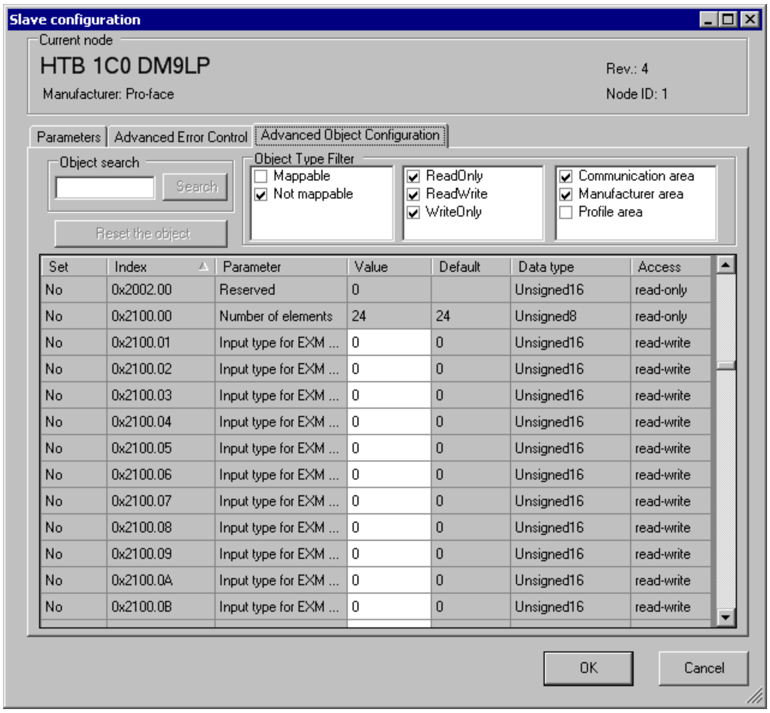
NOTE

- Map the input data from HTB (CANopen slave) to the GP (CANopen master) for TPDO, and map the object relating to the output data from the GP to HTB for RPDO.
- Map error register "0x1001.1", status register "0x1002.1", and module diagnosis "0x3000.1", "0x3000.2" and "0x3000.2" to TPDO as required.

4 Open [Advanced Object Configuration] tab and configure detailed settings of the object. Set the data type of analog input (0x2100.1 to 0x2100.4) and the data type of analog output (0x2200.1).
Set the data range (Input: 0x2101.1 to 0x2101.4, output: 0x2201.1), PDO transmission enabled/disabled (0x6423), fallback mode (0x6443.1) and fallback value (0x6444.1), and then click [OK] to close the dialog box.

NOTE

- For details of objects, refer to the following.
  ☞ "30.7.10 List of HTB Objects" (page 30-232)



Slave configuration

Current node
HTB 1C0 DM9LP
Manufacturer: Pro-face
Rev.: 4
Node ID: 1

Parameters | Advanced Error Control | Advanced Object Configuration

Object search
Search
Reset the object

Object Type Filter
□ Mappable
☑ Not mappable
☑ ReadOnly
☑ ReadWrite
☑ WriteOnly
☑ Communication area
☑ Manufacturer area
□ Profile area

Set	Index	Parameter	Value	Default	Data type	Access
No	0x2002.00	Reserved	0		Unsigned16	read-only
No	0x2100.00	Number of elements	24	24	Unsigned8	read-only
No	0x2100.01	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.02	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.03	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.04	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.05	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.06	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.07	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.08	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.09	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.0A	Input type for EXM ...	0	0	Unsigned16	read-write
No	0x2100.0B	Input type for EXM ...	0	0	Unsigned16	read-write

OK | Cancel

5 Click [I/O Screen] in [I/O Driver Settings] or select [I/O Screen] in the [Screen List] window on Workspace to assign a variable to each of the mapped objects. For information on how to assign variables, refer to the following.
☞ "30.7.3 Allocating I/O (Common)" (page 30-163)

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

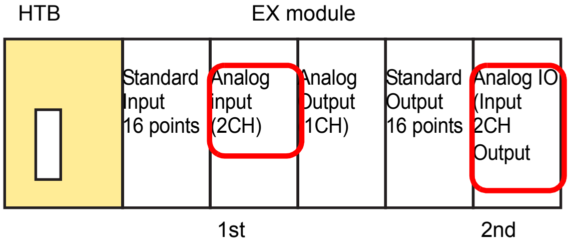
NOTE

- To check set values, use SDOR instructions.
- When changing the analog setting value with SDOW instructions, you need to use the expansion bus reset (0x3300) to stop the internal bus before making the change.

■ Sub Index

Sub index 0 stores the total number of channels used per feature, and from sub index 1 onward, each channel on the EX module is mapped, in order, beginning with the channel closest to HTB.

For example, for analog input feature

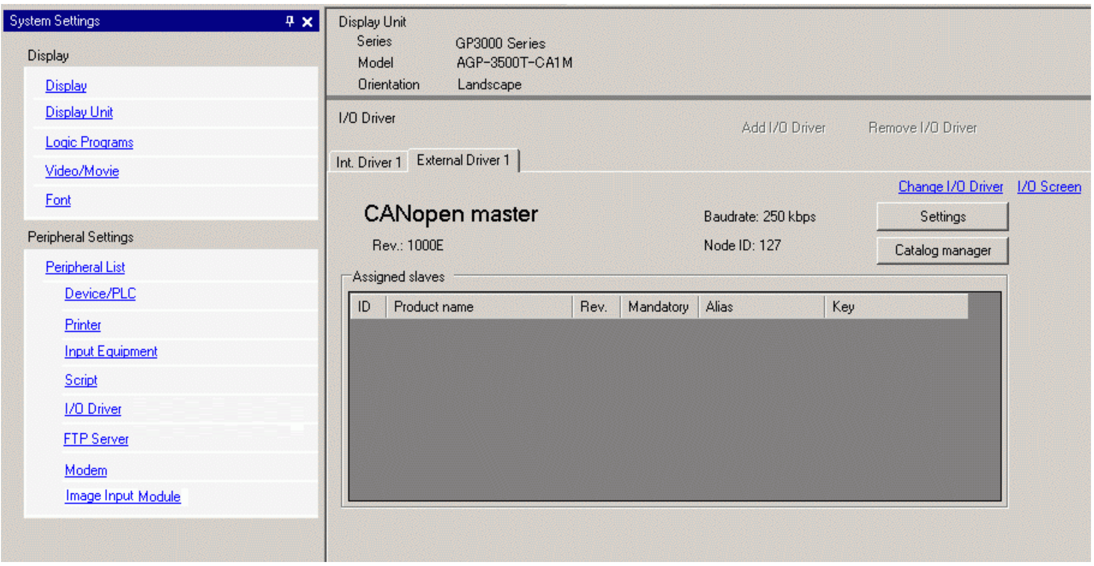


The analog input object's sub index contains the number of modules used in the analog input only, counting from those closest to HTB.

Sub index		
0	4	<- Total number of channels for analog input
1	Input data	<- CH1 data of the 1st EX module
2	Input data	<- CH2 data of the 1st EX module
3	Input data	<- CH1 data of the 2nd EX module
4	Input data	<- CH2 data for 2nd EX module

30.7.7 [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

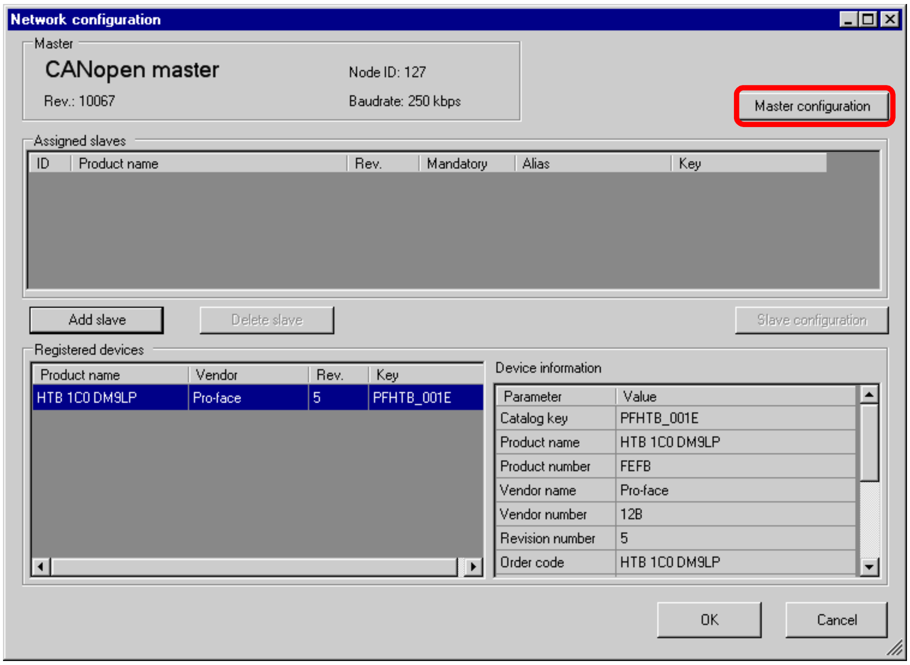
I/O Driver

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 30-205)
Catalog Manager		Displays the [Catalog Manager] dialog box. ☞ " ■ Catalog Manager Dialog Box" (page 30-222)
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
	Mandatory	Displays whether the slave is mandatory in the network.
	Alias	Displays the alias assigned to 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.

*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



CANopen master

Node ID: 127

Rev.: 10067

Baudrate: 250 kbps

ID	Product name	Rev.	Mandatory	Alias	Key

Product name	Vendor	Rev.	Key
HTB 1C0 DM9LP	Pro-face	5	PFHTB_001E

Parameter	Value
Catalog key	PFHTB_001E
Product name	HTB 1C0 DM9LP
Product number	FEFB
Vendor name	Pro-face
Vendor number	12B
Revision number	5
Order code	HTB 1C0 DM9LP

Setting		Description
Master Configuration		Displays Master Configuration dialog box. ☞ " ◆ Master Configuration Dialog Box" (page 30-207)
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.
	Rev.	Displays the slave's EDS file revision number.*1
	Mandatory	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 30-222)
Delete Slave		Deletes the slave selected in [Assigned slaves] from the network.

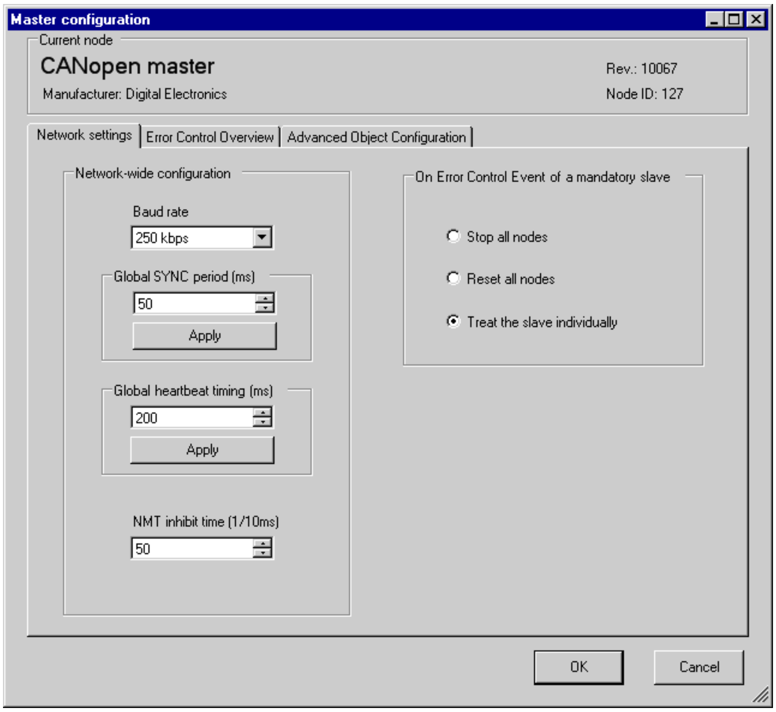
Continued

Setting		Description
Slave Configuration		Displays Slave Configuration dialog box. ☞ " ◆ Slave Configuration Dialog Box" (page 30-212)
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



Master configuration

Current node

CANopen master

Manufacturer: Digital Electronics

Rev.: 10067

Node ID: 127

Network settings | Error Control Overview | Advanced Object Configuration

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

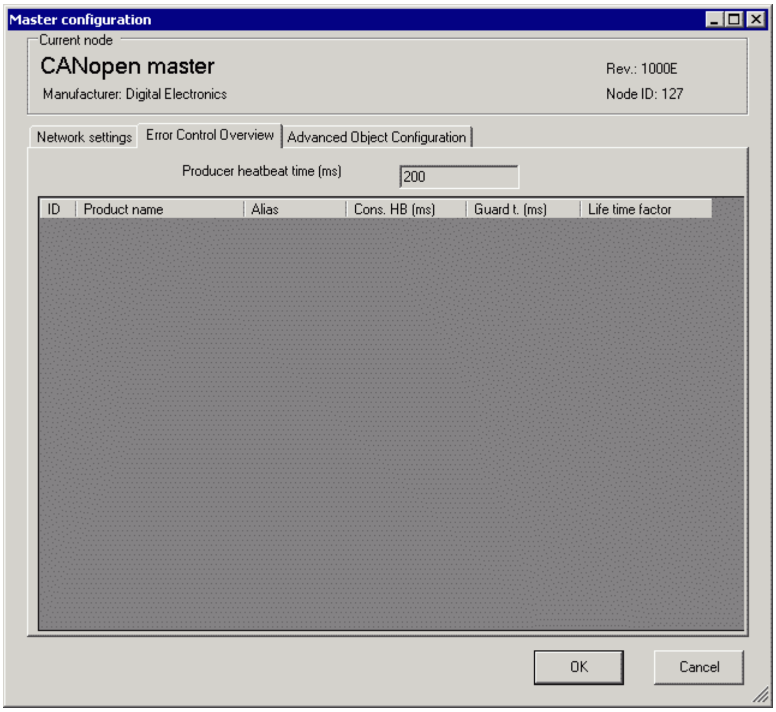
<table>
<tr><th colspan="2">Setting</th><th>Description</th></tr>
<tr><td colspan="2">Network-wide configuration</td><td>Configures CANopen network settings.</td></tr>
<tr><td></td><td>Baud rate</td><td>Selects corresponding baud rate from [50kbps], [125kbps], [250kbps], [500kbps], [800kbps] and [1000kbps].
<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 (HTB, etc.) baud rate on the slave unit itself.</td></tr>
</table>

Continued

<table>
<tr><th colspan="2">Setting</th><th>Description</th></tr>
<tr><td rowspan="3">Network Configuration</td><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.
NOTE
• If the global SYNC period of the slave side has been changed to a value other than 0ms, whenever the global SYNC period of the master side is changed, the global SYNC period of the slave side will automatically be changed while maintaining the ratio with the global SYNC period of the master side.</td></tr>
<tr><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>Bit 6</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>
NOTE
• [Handle the slave individually] sends a reset to individual slaves, rather than the reset being sent by broadcast.</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

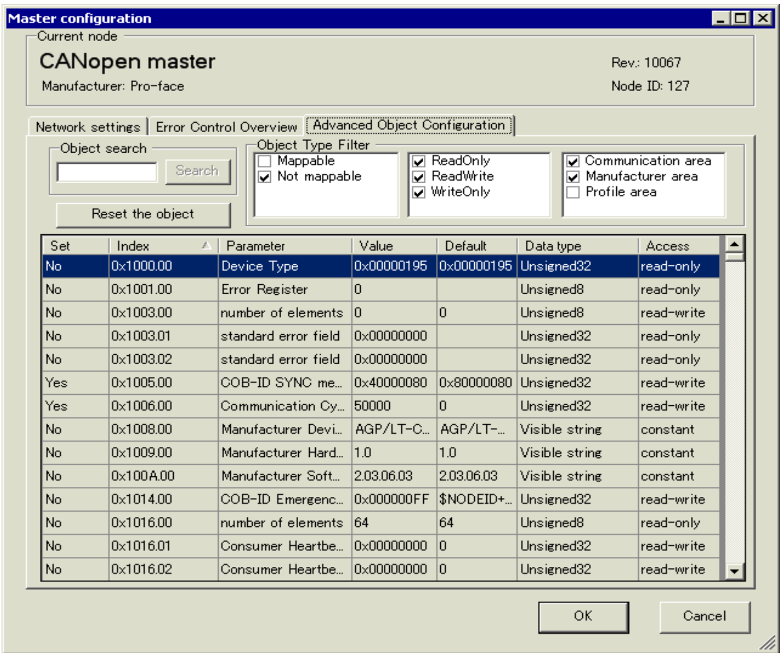
Setting	Description
Producer Heartbeat Time	Displays the sending period of the heartbeat set in [Global Heartbeat Time] in [Network Settings] tab. Unit is in milliseconds (ms). It is stored in the object 1017h for CANopen master.
ID	Node ID in CANopen network.
Product Name	Display EDS product name.
Comment	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.
Lifetime 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 of CANopen.



Master configuration

Current node

CANopen master

Manufacturer: Pro-face

Rev.: 10067

Node ID: 127

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	0x00000195	0x00000195	Unsigned32	read-only
No	0x1001.00	Error Register	0		Unsigned8	read-only
No	0x1003.00	number of elements	0	0	Unsigned8	read-write
No	0x1003.01	standard error field	0x00000000		Unsigned32	read-only
No	0x1003.02	standard error field	0x00000000		Unsigned32	read-only
Yes	0x1005.00	COB-ID SYNC me...	0x40000080	0x80000080	Unsigned32	read-write
Yes	0x1006.00	Communication Cy...	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 Soft...	2.03.06.03	2.03.06.03	Visible string	constant
No	0x1014.00	COB-ID Emergenc...	0x000000FF	$NODEID+...	Unsigned32	read-write
No	0x1016.00	number of elements	64	64	Unsigned8	read-only
No	0x1016.01	Consumer Heartbe...	0x00000000	0	Unsigned32	read-write
No	0x1016.02	Consumer Heartbe...	0x00000000	0	Unsigned32	read-write

Setting	Description
Object Search	Enter the index of the object that you wish to search or the parameter text string, and then click [Search]. The application searches for the corresponding object in the object list.
Reset	Returns the set up value of the object selected in the [Object List] to its initial value.
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)

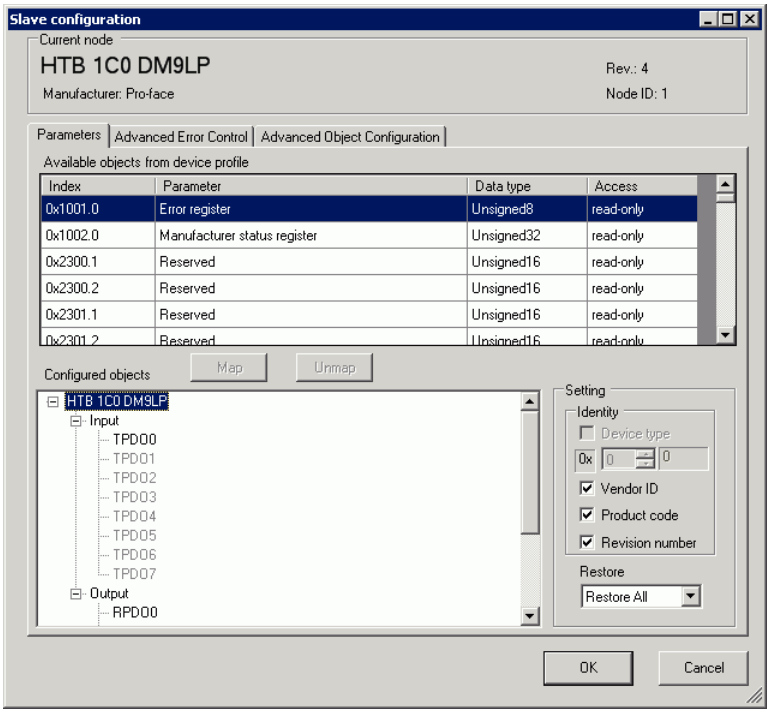
Continued

Setting		Description
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.
	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 (Reservation)
	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



HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x1001.0	Error register	Unsigned8	read-only
0x1002.0	Manufacturer status register	Unsigned32	read-only
0x2300.1	Reserved	Unsigned16	read-only
0x2300.2	Reserved	Unsigned16	read-only
0x2301.1	Reserved	Unsigned16	read-only
0x2301.2	Reserved	Unsigned16	read-only

Configured objects

```
⊟ HTB 1C0 DM9LP
   ⊟ Input
      TPD00
      TPD01
      TPD02
      TPD03
      TPD04
      TPD05
      TPD06
      TPD07
   ⊟ Output
      RPD00
```

☐ Device type

0x 0 0

☑ Vendor ID

☑ Product code

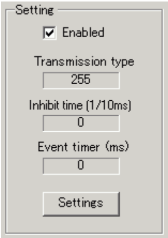
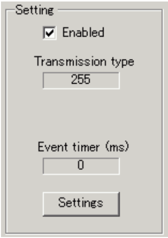
☑ Revision number

Restore: Restore All

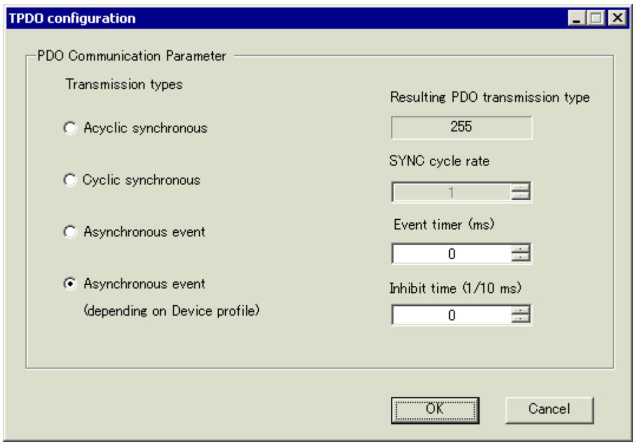
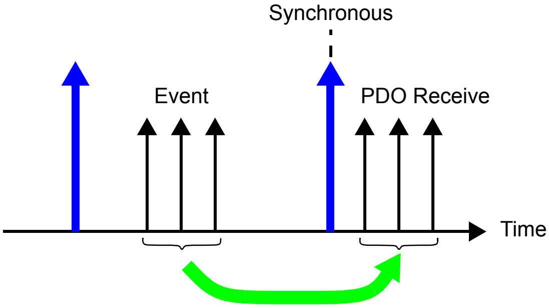
Setting		Description
Available objects from device profile		Display the list of objects that can be mapped to PDO.
	Index	Display index and sub index of the object. The sub index appears after the comma. For example, for "0x1003.2", "0x1003" is index and "2" is sub index.
	Parameter	Display parameter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

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 (Reservation)
	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.

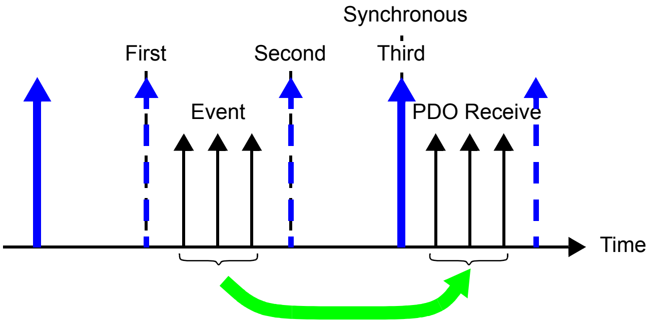
Continued

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 if 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 have not been restored. All: All parameters have been restored. Communication only: Parameters in the area shown in [Commutation Area] of [Slave Configuration/Extended] have been restored. Application only: Parameters in the area shown in [Manufacturer Area] and [Profile Area] of [Slave Configuration/Extended] have been restored. **NOTE** • Selected items that are not supported by the slave are not displayed.The EDS file of each slave defines whether the slave supports the selected item. • If the HTB is used, the application data is not recovered even if the configuration of [Application Only] has been selected. • Manufacturer-defined selected items (after sub index 4 for object 0x1011) are displayed as "extension 4" or "extension 5".
	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 Settings (When TPDO is selected)  Setting ☑ Enabled Transmission type 255 Event timer (ms) 0 Settings (When RPDO is selected) **NOTE** • If the PDO is disabled, when you re-open the project, all the information set in the PDO is initialized to its initial values. Since this initialization setting depends on the default setup on the slave, objects may be being mapped.

Continued

Setting			Description
Settings	When TPDO/RPDO Is Selected	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).
		Number of objects	Change the number of objects of Semi-Static objects to be mapped to PDO. **NOTE** • It will be displayed only if the PDO for Semi-Static objects is selected.
		Change Settings	Click to display the following screen. Different screens are displayed for RPDO and TPDO. When RPDO is selected  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 • 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. 

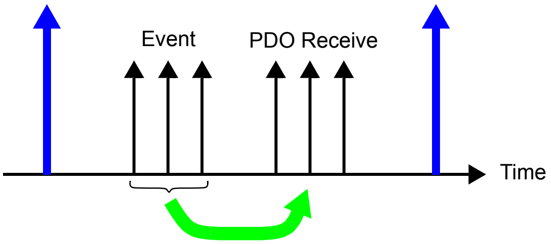
Continued

Setting			Description
Settings	When TPDO/RPDO Is Selected	Change Settings	• 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 when 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>

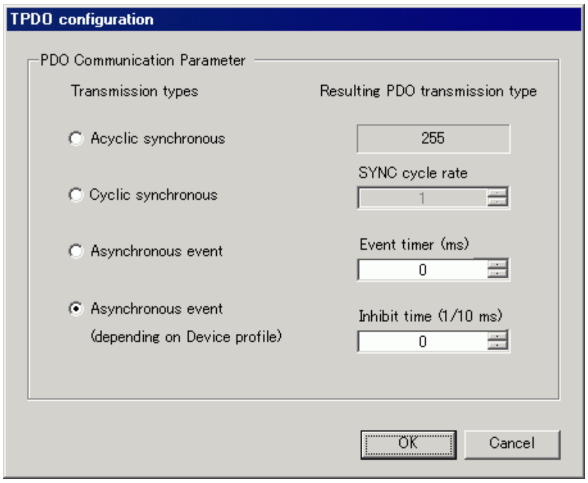
- Asynchronous event (standard event)/Determined by (standard event) device profile
  PDO is received immediately when an event occurred regardless of the SYNC signal. For 254, this is defined by the manufacturer. For 255, this is defined in the device profile.



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

<table>
<tr><th>Object number</th><th>Sub index</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

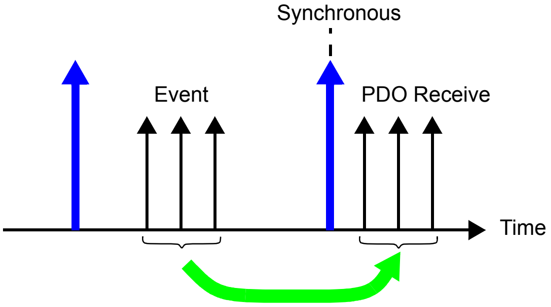
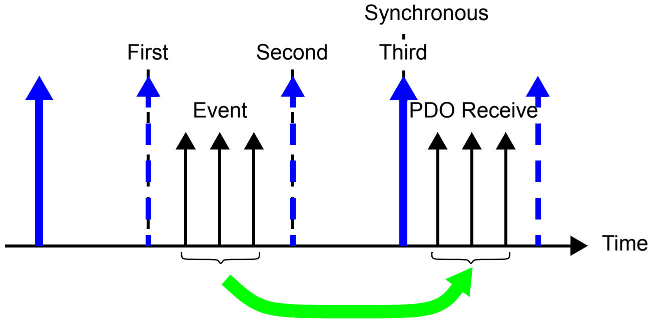
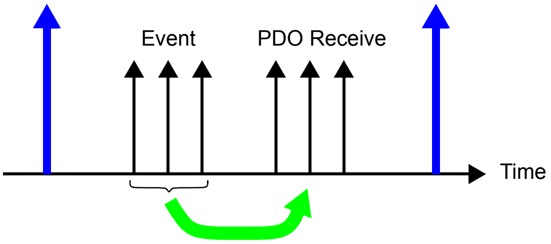
SYNC cycle rate: 1

Event timer (ms): 0

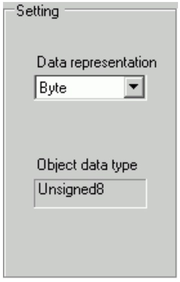
Inhibit time (1/10 ms): 0

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

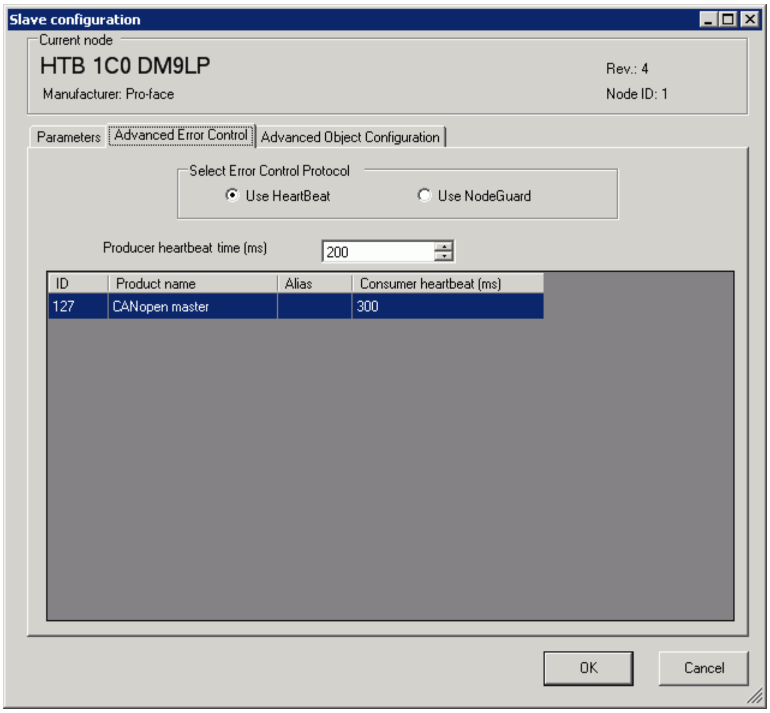
Continued

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><th>Object number</th><th>Sub index</th><th>Item Name</th></tr>
<tr><td>0x1800 + PDO number</td><td>0</td><td>Number of TPDO</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>
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 (Common) ■ I/O Screen Display" (page 30-163)
• 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

HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Select Error Control Protocol

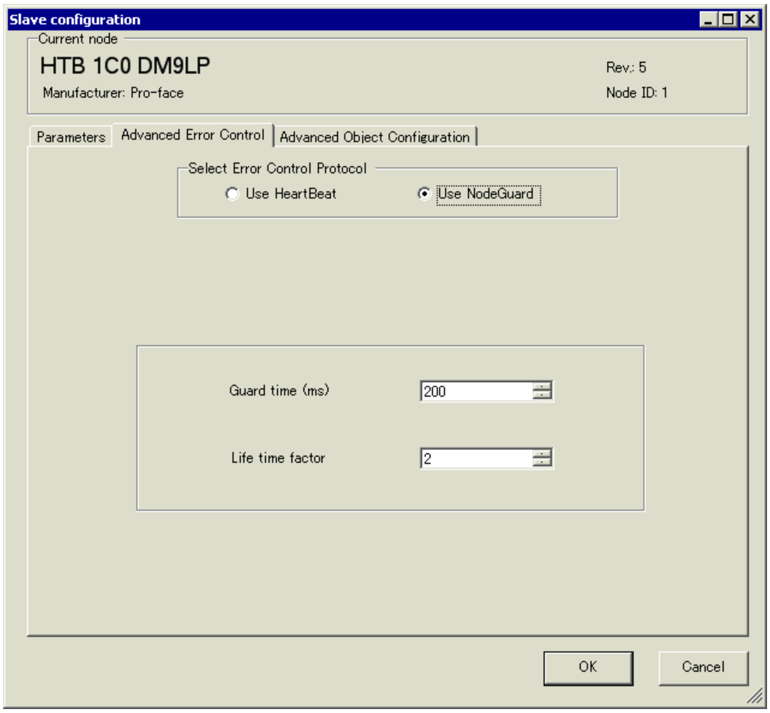
● Use HeartBeat ○ Use NodeGuard

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.
Comment	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

HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 5

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.
Lifetime 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. 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.

Display contents are the same as those of [Advanced Object Configuration] tab in [Master Configuration] dialog box.Displays the contents of the slave objects.

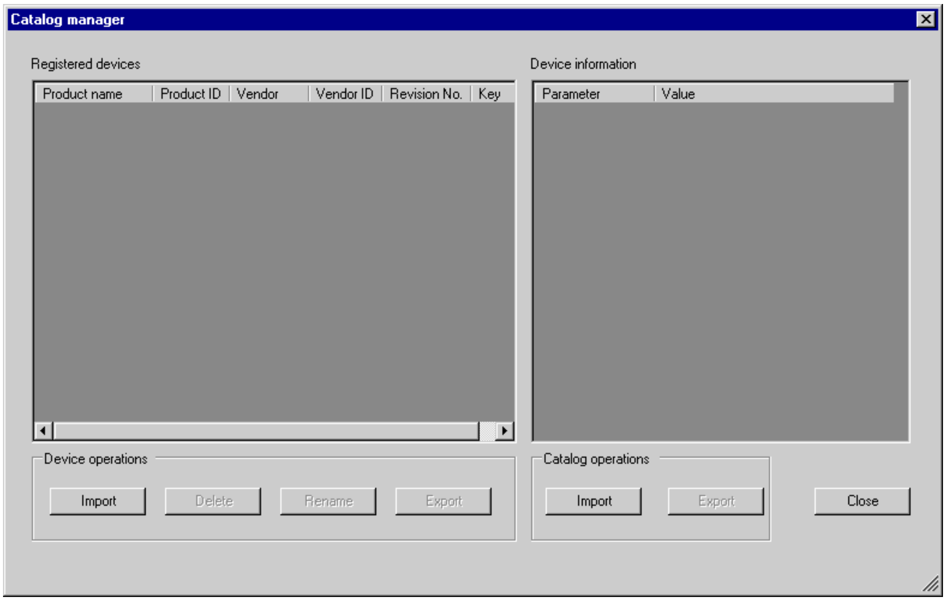
☞ "30.7.7 [I/O Driver] Settings Guide • Advanced Object Configuration" (page 30-210)

NOTE

- The object 0x1006 of the slave side (global SYNC period) is the monitor cycle for SYNC on the slave side.Please pay attention to the value of the monitor cycle for SYNC at the slave because it is related to the value of the global SYNC period on the master side.

  ☞ "30.7.7 [I/O Driver] Settings Guide ◆ Master Configuration Dialog Box" (page 30-207)
- The object 0x1016 (consumer heartbeat time) is always displayed in hexadecimals.

■ Catalog Manager Dialog Box



Setting		Description
Registered Devices		Displays a list of the registered slaves When using a slave other than HTB, 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.

Continued

Setting		Description
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. The exported ESS file has sections in which some descriptions are different from the imported original EDS file, but this may not cause any errors. NOTE • When handling the EDS file with software other than GP-Pro EX, use the original EDS file before importing.
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.

30.7.8 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.
	014	Setting value error	Error code is 0 when error flag is set		Reset AGP. If the problem is unresolved, please contact your support center.

Continued

	Error Code	Error Message	Description		Solution
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.

	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 by 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.
	145	SDO operand error	SDO instruction parameter 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.

Continued

	Error Code	Error Message	Description		Solution
Application related error	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	Slave error	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.		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.		
	155	Concise DCF mismatch for one or more slaves	Slave configuration and the object directory are mismatched.		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.

30.7.9 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.
- When mapping a variable to I/O (terminal) on the I/O screen, do not specify (use) any array variables. It will not operate normally.
- If TPDO or RPDO is set to more than the maximum of 252 in the PDO setting, the error message "Compile error" may be displayed.
  The TPDO or RPDO setting should be up to the maximum of 252.
- If the baud rate is changed, all the equipment connected to the CANopen network must be turned on again. Slave devices with auto baud rate setting must be turned on again as well.
- When turning on the master and slave separately on the CANopen network, the CANopen Slave (HTB, etc.) must be turned first, and then the CANopen master (GP, etc.) must be turned on last. Otherwise, the CANopen network may not operate properly.

■ Specific restrictions for HTB

- The number of PDO that the HTB unit can use is 16 in total. As all the objects cannot be mapped, please map only required objects to PDO.
- For HTB and EX module standard outputs, do not perform write operations from multiple objects.
- If the analog module is connected to the HTB, after the CANopen communication between the GP or the LT unit is established, error code 155 (Slave Configuration Error) may be displayed when the GP or the LT unit is reset.
- If the HTB is used, for the recovery feature for the parameter, the application data is not recovered even if the configuration of [Application Only] is selected.
- With GP-Pro EX V2.5 or higher, EDS files for the HTB are built-in. They do not need to be downloaded from “Otasuke Pro!” and imported.

30.7.10 List of HTB Objects

HTB objects are described in the following three categories.

☞ " ■ Communication Setting Objects (1000h to 1FFFh)" (page 30-232)
☞ " ■ Manufacturer Objects (2000h to 5FFFh)" (page 30-253)
☞ " ■ Standard Device Object (6000h to 9FFFh)" (page 30-266)

NOTE
- The Reserved objects may be expanded in the future, so do not use them for the following purposes:
  •Mapping to PDO
  •Object value input for setting extended objects

■ Communication Setting Objects (1000h to 1FFFh)

Object (Hex)	Parameter	Description	Reference page
1000	Device Type	Device type	☞ 30-233
1001	Error Register	Error Register	☞ 30-234
1002	Manufacturer Status Register	Status register	☞ 30-235
1003	Pre-defined Error Field (PEF)	PEF (Pre-defined Error Field)	☞ 30-236
1005	COB-ID SYNC message	SYNC COB-ID	☞ 30-237
1006	Communication Cycle Period	SYNC sending period	☞ 30-238
1008	Manufacturer Device Name	Manufacturer Device Name	☞ 30-239
100A	Manufacturer Software Version (MSV)	MSV (Manufacturer Software Version)	☞ 30-239
100C	Guard Time	Guard Time	☞ 30-239
100D	Lifetime Factor	Lifetime factor	☞ 30-239
1010	Store parameters	Storing parameters	☞ 30-240
1011	Restore Default parameters	Restoring Default Parameters	☞ 30-241
1014	COB-ID Emergency (EMCY) message	EMCY COB-ID	☞ 30-242
1016	Consumer heartbeat time	Consumer heartbeat time	☞ 30-242
1017	Producer heartbeat time	Producer Heartbeat Time	☞ 30-243
1018	Identity Object	Identity Object	☞ 30-243
1027	Module list	Module List	☞ 30-244
1200	Server SDO Parameter	Server SDO Parameter	☞ 30-244

Continued

Object (Hex)	Parameter	Description	Reference page
1400 to 1407	Receive PDO communication Parameter	RPDO communication parameter	☞ 30-245
1600 to 1607	Receive PDO Mapping Parameter	RPDO Mapping Parameter	☞ 30-247
1800 to 1807	Transmit PDO Communication Parameter	TPDO communication parameter	☞ 30-248
1A00 to 1A07	Transmit PDO Mapping Parameter	TPDO Mapping Parameter	☞ 30-252

◆ 1000h: Device Type

The last byte indicates the profile number (401 (191h) for CANopen specification I/O).
The last byte is called "additional information", which indicates feature details of the device.

<table>
<tr><td>31</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>16</td><td>15</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>4</td><td>3</td><td>2</td><td>1</td><td>0</td></tr>
<tr><td colspan="16">Additional information</td><td colspan="12">Unused</td><td>d</td><td>c</td><td>b</td><td>a</td></tr>
</table>

a: The device has standard input
b: The device has standard output
c: The device has analog input
d: The device has analog output

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 32	-	ro (read only)	X	X

◆ Object 1001h: Error Register

This object is used by the device to display an internal failure. The corresponding bit becomes active when failure is detected.

The following failures can display:

Bit	Description	Remarks
0	General error	Set to 1 when failure including invalid parameter is detected in HTB and its EX module.
1	Current	From analog input of EX module (when set for current)
2	Voltage	From analog input of EX module (when set for voltage)
3	Temp.	-
4	Communication error	CANopen error indicator. Can be supported by stack.
5	Reserved	-
6	Reserved	-
7	Manufacturer specific	Set to 1 when internal bus failure is detected between HTB and EX module.

These bits indicate boolean "OR" in case of failure in HTB and EX module.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 8	-	ro (read only)	O	X

◆ Object 1002h: Status Register

This object is stored in EMCY message.
Error code stored in the last word.
Additional information stored in the top word.

<table>
<tr><td>31</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>16</td><td>15</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td></td><td>0</td></tr>
<tr><td colspan="16">Additional information</td><td colspan="16">Error Code</td></tr>
</table>

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 32	-	ro (read only)	O	X

When each of the following bits is 0, it indicates normal and when 1, it indicates failure.

Register	Feature	Description
Bit 0 to 15	State of HTB + EX module	Bit 0 to 8: Unused Bit 9: Communication or external failure Bit 10 to 12: Unused Bit 13: Invalid configuration (There is no EX module or the configuration is invalid) Bit 14, 15: Unused
Bit 16 to 31	State of EX module	Bit 16: HTB state bit Bit 17: 1st EX module state bit Bit 18: 2nd EX module status bit Bit 19: 3rd EX module status bit Bit 21: 4th EX module status bit Bit 22: 5th EX module status bit Bit 23: 6th EX module status bit Bit 24: 7th EX module status bit Bit 25 to 31: Unused

NOTE

- EX is counted 1st, 2nd, etc. from the side closest to HTB.

◆ Object 1003h: PEF (Pre-defined Error Field)

This object is used to latch the latest failure and its properties.

- Error code is latched by the bottom two bytes. For details, refer to the error code list.
- "Additional information" is latched by the top two bytes. The top bytes of additional information (1002h, bits 16 to 23) is set up with the same information as the bottom bytes (1001h).
- The latch error number is stored in sub index 0.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index count = Total number of latched errors	UNSIGNED 8	0	rw (read-write)	X	X
1	Latest error	UNSIGNED 32	-	ro (read only)	X	X
2	2nd last error	UNSIGNED 32	-	ro (read only)	X	X
...						
10						

When a new error occurs, the displayed code moves to the sub index at the upper level. That is, error of sub index 1 goes to sub index 2, and error of sub index 2 goes to sub index 3. Afterward it is processed similarly. A maximum of 254 errors are stored.
History of error codes can be deleted by writing value 0 to sub index 0 of the object 1003h.
The error codes are not deleted from PEF even if the error is fixed.
All failures are notified by sending EMCY message ("emergency" message). When the error conditions are removed, a no-error EMCY message is sent (error code 0x0000).

Error code list

Error code (hexadecimal display)	Diagnostic message	Cause
0000	ERROR_RESET_OR_NO_ERROR	Error has disappeared
1000	GENERIC_ERROR	Internal communication error
6101	SOFTWARE_RX_QUEUE_OVERRUN	Overflow of receiving memory
6102	SOFTWARE_TX_QUEUE_OVERRUN	Overflow of sending memory
8100	COMMUNICATION	Error in sending and receiving synchronous counter. EMCY is sent when the counter is more than 96.
8120	CAN_IN_ERROR_PASSIVE_MODE	CAN controller interrupt
8130	LIFE_GUARD_ERROR	Node-Guarding error
8140	BUS_OFF	Overflow of sending buffer counter

◆ Object 1005h: SYNC COB-ID

This object stores the synchronous message ID.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 32	80h	rw (read-write)	X	O

◆ Object 1006h: Global SYNC period

This object indicates time interval between two SYNC signals. This interval is at least 10ms and the minimum increment needs to be 1ms. It must be entered with double word.
Field is set to 0 when this object is not used.
When the value 10,000 to 10,000,000 is entered, HTB needs to receive SYNC signals within this time interval. If it is not received, it is shifted to pre-operational state. Maximum tolerance is 1% of the set value. Monitoring elapsed time starts when the first SYNC signal is received.

NOTE

- If the setting is other than 0, set a larger global SYNC period than the master side's global SYNC period. When a global SYNC period smaller than the global SYNC period of the master side is set, an error message will be displayed.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 32	0	rw (read-write)	X	O

Coding switching cycle period

Threshold is shown in the following table.

Value type	Decimal value	Hexadecimal value	Synchronization interval (ms)
Standard value	0	0000 0000	-
Min	10000	0000 2710	10
-	25000	0000 61A8	25
-	250000	0003 D090	250
-	1000000	000F 4240	1000
-	5000000	004C 4B40	5000
Max	10000000	0098 9680	10000

◆ Object 1008h: Manufacturer Device Name

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	STRING	HTB1C0DM9LP	ro (read only)	X	X

◆ Object 100Ah: MSV (Manufacturer Software Version)

This object stores details of HTB firmware version in 'Vxx.yy' format.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	STRING	-	ro (read only)	X	X

◆ Object 100Ch: Guard Time

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 16	0	rw (read-write)	X	O

Guard time is stored in millisecond (ms).
Guard time x Lifetime factor (retry count) = Monitoring time

◆ Object 100Dh: Lifetime factor

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 8	0	rw (read-write)	X	O

◆ Object 1010h: Storing parameters

This object is used to store parameters of HTB and EX module in backup memory.

NOTE

- When this object stores parameters, select "All" or "Communication only" for "Restore Parameters" in the HTP slave setting. If any option other than "All" or "Communication only" is selected, it may not communicate properly.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED 8	4	ro (read only)	X	X
1	Store all the parameters	UNSIGNED 32	-	rw (read-write)	X	X
2	Store communication parameter (1000h to 1FFFh)	UNSIGNED 32	-	rw (read-write)	X	X
3	Unused	-	-	-	-	-
4	Application parameter (2000h to 9FFFh)	UNSIGNED 32	-	rw (read-write)	X	X

Action

To store parameters, it is necessary to write "save" text (6576 6173h) to the corresponding index.

	Top byte		Bottom byte	
ISO 8859 (ASCII) signature	e	v	a	s
Hexadecimal value	65h	76h	61h	73h

Information on storage feature is read from sub index. Obtained result 0000 0001h indicates parameters are restored by the module only when applicable command is received.

◆ Object 1011h: Restoring Default Parameters

This object is used to restore parameters of HTB and EX module. Restoring parameters is considered only after the power is turned on.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED 8	4	ro (read only)	X	X
1	Restore all the default parameters	UNSIGNED 32	-	rw (read-write)	X	X
2	Restore default communication parameters (1000h to 1FFFh)	UNSIGNED 32	-	rw (read-write)	X	X
3	Unused	-	-	-	-	-
4	Restore default application parameters (2000h to 9FFFh)	UNSIGNED 32	-	rw (read-write)	X	X

Action

To restore parameters, it is necessary to write "load" text (6461 6F6Ch) to the corresponding index.

	Top byte		Bottom byte	
ISO 8859 (ASCII) signature	d	a	o	l
Hexadecimal value	64h	61h	6Fh	6Ch

Information on whether default parameters of the module can be restored is read from sub index. Obtained result 0000 0001h indicates that parameters are restored only when an applicable command is received by the module.

◆ Object 1014h: COB-ID EMCY (Emergency) Message

This object stores the EMCY emergency message ID.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 32	80h + node ID	rw (read-write)	X	O

◆ Object 1016h: Consumer Heartbeat Time

This object is used when time interval set in ms for which HTB must receive heartbeat message from the monitored CANopen master. HTB is designed to monitor only one CANopen master.
The value of this objects must be greater than that of the object 1017h.
Time needs to be a multiple of 1ms.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED 8	1	ro (read only)	X	O
1	Consumer heartbeat time	UNSIGNED 32	0	rw (read-write)	X	O

Contents of variables

Contents of sub index 1 are as follows.

Bit	31 (MSB) to 24	23 to 16	15 to 0 (LSB)
Value	0h (Reserved)	Address of monitored CANopen master	Monitoring time (ms)

CANopen master is not monitored if the value of the object is 0.

◆ Object 1017h: Producer Heartbeat Time

This object is used when the time interval is set in ms for the module to generate heartbeat message.

The default monitoring method is the slave guard. Heartbeat is used when any value except 0 is written in this object.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED 16	0	rw (read-write)	X	O

◆ Object 1018h: Identity Object

This object stores the device information. It includes manufacturer's CiA identifier (vendor ID), product code, and revision number of HTB.

Revision information is divided into the following two parts.

- Major revised part (top word) indicates modification of CANopen feature.
- Minor revised part (bottom word) indicates modification of HTB feature only.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Input number	UNSIGNED 8	3h	ro (read only)	X	X
1	Vendor ID	UNSIGNED 32	12Bh	ro (read only)	X	X
2	Product code	UNSIGNED 32	FEFBh	ro (read only)	X	X
3	Revision number	UNSIGNED 32	-	ro (read only)	X	X

◆ Object 1027h: Module List

This object stores the list of EX modules connected to the bus.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Number of sub index - Number of EX modules connected	UNSIGNED 8	7	ro (read only)	X	X
1	Product code of the first module	UNSIGNED 16	-	ro (read only)	X	X
...						
7	Product code of the last module	UNSIGNED 16	-	ro (read only)	X	X

◆ Object 1200h: Server SDO Parameter

Description

This object stores the message ID for SDO communication.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED 8	2h	ro (read only)	X	X
1	COB-ID of receive SDO (client - server)	UNSIGNED 32	600h + node ID	ro (read only)	X	X
2	COB-ID of transmit SDO (server - client)	UNSIGNED 32	580h + node ID	ro (read only)	X	X

◆ Object 1400h to 1407h: RPDO communication parameter

This object stores information of received PDO.

Properties of the object

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED 8	2	ro (read only)	X	O
1	COB-ID	UNSIGNED 32	Object 1400h: 200h + node ID Object 1401h to 1403h: 80000200h + the last digit of the object * 100h + Node ID Object 1404h to 1407h: 80000000h + node ID	rw (read-write)	X	O
2	Transmission mode	UNSIGNED 8	255	rw (read-write)	X	O

Transmission mode

PDO transmission mode can be set as in the following table.

Transfer code	Transmission mode					Remarks
	Cyclic	Noncyclic	Synchronous	Asynchronous	RTR only	
0		X	X			Send PDO in the first Sync message after the event
1 to 240	X	X	X			Send PDO per x pieces of Sync message
241 to 251	Reserved					-
252 to 253	Reserved					-
254				X		Send PDO when an event occurs
255				X		Send PDO when an event occurs

NOTE

- In Modes 254 and 255, the event to trigger transmission is defined by the message producer.

◆ Object 1600h to 1607h: RPDO Mapping Parameter

This object describes the objects transferred by PDO.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED 8	-	rw (read-write)	X	O
1	First object in PDO	UNSIGNED 32	–	rw (read-write)	X	O
2	Second object in PDO	UNSIGNED 32	–	rw (read-write)	X	O
...						
8	Last object in PDO	UNSIGNED 32	-	rw (read-write)	X	O

Configuration of data fields

Each data object that is transferred is indicated in the following format.

Bit	31 (MSB) to 16	15 to 8	7 to 0 (LSB)
Summary	Index number of the transferred object	Sub index number of the transferred object	Length of the transferred object
Example	6200h	01h	08h

NOTE

- Maximum total length of the data transferred by PDO is 8 bytes.

◆ Object 1800h to 1807h: TPDO Communication Parameter

This object stores information of sent PDO.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED 8	5	ro (read only)	X	O
1	COB-ID	UNSIGNED 32	Object 1800h: 180h + node ID Object 1801h to 1803h: 80000180h + the last digit of the object * 100h + Node ID Object 1804h to 1807h: 80000000h + node ID	rw (read-write)	X	O
2	Transmission mode	UNSIGNED 8	255	rw (read-write)	X	O
3	Inhibit Time	UNSIGNED 16	0	rw (read-write)	X	O
4	Unavailable					
5	Event Timer	UNSIGNED 16	0	rw (read-write)	X	O

Transmission mode

PDO transmission mode can be set as in the following table.

Transfer code	Transmission mode					Remarks
	Cyclic	Noncyclic	Synchronous	Asynchronous	RTR only	
0		X	X			Send PDO in the first Sync message after the event
1 to 240	X		X			Send PDO per x pieces of Sync message
241 to 251	Reserved					-
252	Unused					Receive SYNC message and send PDO with remote request
253	Unused					Update the data and send PDO with remote request
254				X		Send PDO when an event occurs
255				X		Send PDO when an event occurs

NOTE

- For digital I/O and analog I/O, the event changes the value.

COB-ID configuration

Configuration of COB-ID that supports CAN2.0 is shown in the following table:

Bit number	Value	Description
31 (MSB)	0	PDO objects exist
	1	No PDO objects exist
30	-	Reserved
29	0	11-Bit ID (CAN 2.0A)
28 to 11	0	For bit 29 = 0
10 to 0 (LSB)	X	Bit of ID 10 to 0

Inhibit Time

For "Send PDO", you can enter Inhibited Time value in this 16 bit field. When the data loads, the PDO sender checks whether "Inhibited Time" has passed after the previous send. You can send a new PDO only when "Inhibited Time" passes. "Inhibited Time" helps avoid overload of the CAN bus in asynchronous sending (sending mode 255). "Inhibited Time" is in multiple of 100s of the object 1800,03 to 1807,03.
Examples of the value are shown in the following table:

Value	Event timer (ms)
0000h	0
64h	10
3E8h	100
1388h	500
2710h	1000
FFFFh	6553

Event Timer

"Event Timer" acts only in the asynchronous transmission mode (255). Temporary transmission is sent when the data changes before "Event Timer" is ended. When a value greater than 0 is written in this 16 bit field, TPDO is continuously sent after "Event Timer" ends. The value written in 1800.05 to 1805.05 corresponds to "Event Timer" in milliseconds (ms). Data is transferred without data changes.
Examples of the value are shown in the following table:

Value	Event timer (ms)
0000h	0
64h	10
3E8h	100
1388h	500
2710h	1000
FFFFh	6553

◆ Object 1A00h to 1A07h: TPDO Mapping Parameter

This object describes the objects transferred by PDO.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED8	-	rw (read-write)	X	O
1	First object in PDO	UNSIGNED32	–	rw (read-write)	X	O
2	Second object in PDO	UNSIGNED32	–	rw (read-write)	X	O
...						
8	Last object in PDO	UNSIGNED32	-	rw (read-write)	X	O

Configuration of data fields

Each data object that is transferred is indicated in the following format.

Bit	31 (MSB) to 16	15 to 8	7 to 0 (LSB)
Data	Index number of the transferred object	Sub index number of the transferred object	Length of the transferred object
Example	6200h	01h	08h

NOTE

- Maximum total length of the data transferred by PDO is 8 bytes.

■ Manufacturer Objects (2000h to 5FFFh)

Object (Hex)	Parameter	Description	Reference page
2000	**Local digital parameter**	Standard input parameter	☞ 30-254
2100	**Analog input type**	Analog Input Data Type	☞ 30-255
2101	**Analog input range**	Analog Input Data Range	☞ 30-256
2102	**Analog input minimum**	Analog Input Lower Limit	☞ 30-257
2103	**Analog input maximum**	Analog Input Upper Limit	☞ 30-258
2200	**Analog output type**	Analog Output Data Type	☞ 30-259
2201	**Analog output range**	Analog output data range	☞ 30-260
2202	**Analog output minimum**	Analog output lower limit	☞ 30-260
2203	**Analog output maximum**	Analog output upper limit	☞ 30-261
3000	**Module diagnostics**	Module Diagnosis	☞ 30-262
3200	**Parameter status**	Parameters Status	☞ 30-263
3201	**Configuration stack number**	Configuration Stack Number	☞ 30-263
3202	**Restore saved parameters**	Restoring last save parameters	☞ 30-264
3300	**Extension bus reset**	Reset Extension Bus	☞ 30-265

◆ Object 2000h: Standard Input Parameter

Used only for standard input of communication blocks.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (standard input number)	UNSIGNED 8	12	ro (read only)	X	O
1	Standard input parameter 0	UNSIGNED 16	1	rw (read-write)	X	O
...	-	-	-	-	-	-
B	Standard input parameter 11	UNSIGNED 16	1	rw (read-write)	X	O

Configurable sub index value
0: No input filter
1: Input filter 3ms
2: Input filter 12ms

◆ Object 2100h: Analog Input Data Type

This object defines the analog input data type of the EX module.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog input number)	UNSIGNED8	24	ro (read only)	X	O
1	CH1 input type	UNSIGNED16	0	rw (read-write)	X	O
...						
18	Input type of the last channel	UNSIGNED16	0	rw (read-write)	X	O

Configurable sub index value

0: Unused
2: Current (4 to 20mA)
3: Voltage (0 to 10V)
5: Thermocouple (K type)
6: Thermocouple (J type)
7: Thermocouple (T type)
8: Pt100

NOTE

- If you try to set an incompatible value (example, enter PT100 in normal analog input), the "Abort Code" is generated.

◆ Object 2101h: Analog Input Data Range

This object defines the data range of each analog input in EX module.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog input number)	UNSIGNED8	1	ro (read only)	X	O
1	Measurement unit of CH1	UNSIGNED16	1	rw (read-write)	X	O
...						
18	Measurement unit of the last channel	UNSIGNED16	1	rw (read-write)	X	O

Configurable sub index value

0: Fixed
1: User setting
2: Celsius
3: Fahrenheit

NOTE

- If you try to set an incompatible value (example, Celsius in normal analog input), the "Abort Code" is generated.

◆ Object 2102h: Analog Input Lower Limit

This object stores the minimum value of the user-defined measurement range (corresponding sub index value of object 2101h = 1).

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog input number)	UNSIGNED 8	24	ro (read only)	X	O
1	Minimum measured value of CH1	INTEGER16	0	rw (read-write)	X	O
...						
18	Minimum measured value of the last channel	INTEGER16	0	rw (read-write)	X	O

◆ Object 2103h: Analog Input Upper Limit

This object stores the maximum value of the user-defined measurement range (corresponding sub index value of object 2101h = 1: for user setting).

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog input number)	UNSIGNED 8	24	ro (read only)	X	O
1	Maximum measured value of CH1	INTEGER16	0x7FFF	rw (read-write)	X	O
...						
18	Maximum measured value of the last channel	INTEGER16	0x7FFF	rw (read-write)	X	O

◆ Object 2200h: Analog Output Data Type

This object defines data type of the analog output in the EX module.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog output number)	UNSIGNED 8	24	ro (read only)	X	O
1	Output type of CH1	UNSIGNED 16	0	rw (read-write)	X	O
...						
E	Output type of CH14	UNSIGNED 16	0	rw (read-write)	X	O

Configurable sub index value

0: Unused

2: Current (4 to 20mA)

3: Voltage (0 to 10V)

NOTE

- If you try to set an incompatible value (example, voltage (0 to 10V) in normal analog input), the "Abort Code" may be generated.

◆ Object 2201h: Analog Output Data Range

This object defines data range of each analog output in the EX module.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog output number)	UNSIGNED8	24	ro (read only)	X	O
1	Measurement unit of CH1	UNSIGNED16	1	rw (read-write)	X	O
...						
E	Minimum value of CH14	UNSIGNED16	1	rw (read-write)	X	O

Configurable sub index value

0: Fixed

1: User setting

◆ Object 2202h: Analog Output Lower Limit

This object stores the minimum value of the user-defined writing range (corresponding sub index value of object 2201h = 1).

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog output number)	UNSIGNED 8	24	ro (read only)	X	O
1	Minimum value of CH1	INTEGER16	0	rw (read-write)	X	O
...						
E	Minimum value of the last channel	INTEGER16	0	rw (read-write)	X	O

◆ Object 2203h: Analog output Upper Limit

This object stores the maximum value of the user-defined writing range (corresponding sub index value of object 2201h = 1).

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog output number)	UNSIGNED 8	24	ro (read only)	X	O
1	Maximum value of CH1	INTEGER16	0x7FFF	rw (read-write)	X	O
...						
E	Maximum value of the last channel	INTEGER16	0x7FFF	rw (read-write)	X	O

◆ Object 3000h: Module Diagnosis

The object stores the HTB and EX module specific diagnosis.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (module number)	UNSIGNED8	8	ro (read only)	X	X
1	HTB diagnosis	UNSIGNED16	-	ro (read only)	O	X
2	First EX module diagnosis	UNSIGNED16	-	ro (read only)	O	X
...		UNSIGNED16	-	ro (read only)		
8	Last EX module diagnosis	UNSIGNED16	-	ro (read only)	O	X

Mapping HTB bit

Bit 2: Hardware failure (external power failure, common in all the channels)
Bit 3: Invalid module configuration
Bit 8: Value error occurred in the last command
Bit 9: Value consistency error occurred in the last command

DIO module bit mapping

Bit 2: Hardware failure (external power failure, common in all the channels)
Bit 3: Invalid module configuration
Bit 8: Value error occurred in the last command
Bit 9: Value consistency error occurred in the last command

Analog module bit mapping

Bit 0: All channels are operating normally
Bit 1: Module is initialized for all channels
Bit 2: Hardware failure (external power failure, common in all the channels)
Bit 3: Invalid module configuration
Bit 4: Converting data of input CH1 (data is unavailable)
Bit 5: Converting data of input CH2 (data is unavailable)
Bit 6: Thermocouple input CH1 is not configured
Bit 7: Thermocouple input CH2 is not configured
Bit 8: Value error occurred in the last command
Bit 9: Value consistency error occurred in the last command
Bit 10: Analog input data CH1 has exceeded the range

Bit 11: Analog input dataCH2 has exceeded the range

Bit 12: Incorrect wiring (analog input data CH1 is less than the rage)

Bit 13: Incorrect wiring (analog input data CH2 is less than the rage)

Bit 14: Unused

Bit 15: Output channels are unavailable

◆ Object 3200h: Parameters Status

This object stores the parameters status.

Stored object values:

0: Using default parameter

1: Using stored parameter

2: Using current parameter (unsaved)

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED16	0	ro (read only)	X	X

◆ Object 3201h: Configuration Stack Number

This object stores the number of backups executed after the previous restoration of default parameter.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED16	-	ro (read only)	X	X

◆ Object 3202h: Restore Last Saved Parameters

Restore last saved parameters.

⚠WARNING

Unexpected action may occur in the equipment
When parameters are restored in "action" state, unexpected actions may occur in the equipment resulting in damage and physical injury. If it is necessary to operate, set HTB in the "pre-action" state, or ensure physical and equipment safety before attempting other operations.
Failure to follow these instructions may result in serious injury or death.

To restore parameters, you need to write the text string "load" to the corresponding index. You may need to reverse the string to "daol" or 64616F6Ch.

NOTE

- When the restoration proceeds (object 3202h is state 0), this command is ignored.
- Parameters are restored immediately.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Restore specific application parameters	UNSIGNED32	1	rw (read-write)	X	X

Restored objects are as follows.
1005H, 1006H, 100CH, 100DH, 1014H, 1016H, 1017H, 1400H to 1407H, 1600H to 1607H, 1800H to 1807H, 1A00H to 1A07H,
2000H, 2100H, 2101H, 2102H, 2103H, 2104H, 2105H, 2106H, 2200H, 2201H, 2202H, 2203H, 2303H, 2304H, 2305H, 2306H, 2307H, 2354H, 2355H, 2403H, 2404H, 2405H, 2406H, 2407H, 2408H, 2409H, 240AH, 2454H, 2455H, 2457H, 2458H, 2502H, 2503H, 2504H, 2505H, 2506H, 2507H, 2508H, 2557H,
6102H, 6103H, 6302H, 6306H, 6307H, 6308H, 6421H, 6424H, 6425H, 6426H, 6443H, 6444H

◆ Object 3300h: Reset Extension Bus

This object is used to update settings of EX module parameters.

- When this value is set to 1, the internal bus stops.
- When this value is set to 0, the internal bus is enabled if there is consistency with EX module parameters.

NOTE

- Since the method for accessing this object is set to read-write in the [Slave setting] - [Expansion setting], it is allowed to input a value; however, changing the value does not affect anything. Do not change the initial value.
- Even if the initial value is set to 1 for the object 3300h in the HTP unit, it is cleared to 0 when the HTP is booted up.

⚠WARNING

Unexpected action may occur in the equipment
When the internal bus is stopped, all of the output for EX module is set to 0 and HTB output shifts to fallback state.
When extension bus stops in "action" state, unexpected actions may occur in the equipment resulting in damage and physical injury. If it is necessary to operate, set HTB in the "pre-action" state, or ensure physical and equipment safety before attempting other operations.
Failure to follow these instructions may result in serious injury or death.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	-	UNSIGNED16	0	rw (read-write)	X	X

■ Standard Device Object (6000h to 9FFFh)

Object (Hex)	Parameter	Description	Reference page
6000	Digital Input 8 bits	Input (8 Bit)	☞ 30-267
6100	Digital Input 16 bits	Input (16 bit)	☞ 30-268
6102	Polarity inputs 16 bits	Polarity Input	☞ 30-269
6103	Filter Input 16-bits	Input Mask	☞ 30-270
6200	Digital output 8 bits	Output (8 Bit)	☞ 30-271
6300	Digital output 16 bits	Output (16 bit)	☞ 30-272
6302	Polarity Outputs 16 bits	Polarity Output	☞ 30-273
6306	Fallback mode Outputs 16 bits	Fallback Mode Output	☞ 30-274
6307	Fallback Value Output 16 bits	Fallback Value Output	☞ 30-275
6308	Output mask 16 bits	Output Mask	☞ 30-276
6401	Analog Read Input 16 bits	Analog Input	☞ 30-276
6411	Analog Write Output 16 bits	Analog Output	☞ 30-278
6421	Analog input interrupt trigger selection	Analog input trigger settings	☞ 30-278
6422	Analog Input interrupt Source	Analog input PDO transmission channel number	☞ 30-279
6423	Analog Input global Interrupt enable	Analog input enabled/disabled	☞ 30-279
6424	Analog input upper limit	Analog input thresholds high	☞ 30-279
6425	Analog input lower limit	Analog input thresholds low	☞ 30-280
6426	Analog input delta value	Analog input delta value	☞ 30-280
6443	Analog output fallback mode	Analog Output Fallback Mode	☞ 30-281
6444	Analog output fallback value	Analog Output Fallback Value	☞ 30-281

◆ Object 6000h: Input (8 Bit)

This object reports 8 bit digital input value.

Sub index management:

- Each digital EX module uses an even-numbered sub index (allocated with 16 bit word).
- EX module and its sub index are given a number in the order closer to HTB.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 8 bit input)	UNSIGNED8	30	ro (read only)	X	X
1	HTB input 0 to 7	UNSIGNED8	-	ro (read only)	O	X
2	HTB input 8 to 15	UNSIGNED8	-	ro (read only)	O	X
3	First EX module input 0 to 7	UNSIGNED8	-	ro (read only)	O	X
...						
1E	Last EX module input 8 to 15	UNSIGNED8	-	ro (read only)	O	X

◆ Object 6100h: Input (16 Bit)

This object reports 16 bit digital input value.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 16 bit input)	UNSIGNED8	15	ro (read only)	X	X
1	Read HTB input 0 to 15	UNSIGNED16	-	ro (read only)	O	X
2	Read first EX module input 0 to 15	UNSIGNED16	-	ro (read only)	O	X
...						
F	Read last EX module input 0 to 15	UNSIGNED16	-	ro (read only)	O	X

◆ Object 6102h: Polarity Input

This object defines the input polarity.

- 0 = Input is not reversed
- 1 = Input is reversed

NOTE

- The I/O state indicator LED continues to display the actual power status of the connected HTB. It is not affected by this object.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 16 bit input)	UNSIGNED8	15	ro (read only)	X	O
1	Input polarity of HTB	UNSIGNED16	0	rw (read-write)	X	O
2	Polarity input of the first EX module	UNSIGNED16	0	rw (read-write)	X	O
...						
E	Polarity input of the last EX module	UNSIGNED16	0	rw (read-write)	X	O

◆ Object 6103h: Input Mask

This object is used to set the mask for input.

- 0 = Input is read
- 1 = Input is ignored

NOTE

- The I/O state indicator LED continues to display the actual power status of the connected HTB. It is not affected by this object.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 16 bit input)	UNSIGNED8	15	ro (read only)	X	O
1	Input mask of HTB	UNSIGNED16	0	rw (read-write)	X	O
2	Input mask of the first EX module	UNSIGNED16	0	rw (read-write)	X	O
...						
E	Input mask of the last EX module	UNSIGNED16	0	rw (read-write)	X	O

◆ Object 6200h: Output (8 Bit)

This object manages the state of digital output.
Sub index management:

- Each digital EX module uses an even-numbered sub index (allocated with 16 bit word).
- EX module and its sub index are given a number in the order closer to HTB.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 8 bit output)	UNSIGNED8	30	ro (read only)	X	X
1	HTB output 0 to 7	UNSIGNED8	0	rw (read-write)	O	X
2	HTB output 8 to 15 (unused in 8 bit output)	UNSIGNED8	0	rw (read-write)	O	X
3	First EX module output 0 to 7	UNSIGNED8	0	rw (read-write)	O	X
...						
1A	Last EX module output 8 to 15	UNSIGNED8	0	rw (read-write)	O	X

◆ Object 6300h: Output (16 Bit)

This object manages the state of digital output.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 16 bit output)	UNSIGNED8	15	ro (read only)	X	X
1	HTB output 0 to 15	UNSIGNED16	0	rw (read-write)	O	X
2	Last EX module output 0 to 15	UNSIGNED16	0	rw (read-write)	O	X
...						
8	Last EX module output 0 to 15	UNSIGNED16	0	rw (read-write)	O	X

◆ Object 6302h: Polarity Output

This object defines output polarity.

NOTE

- The I/O state indicator LED continues to display the actual power status of the connected HTB. It is not affected by this object.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 16 bit output)	UNSIGNED8	15	ro (read only)	X	O
1	Polarity of HTB digital output	UNSIGNED16	0	rw (read-write)	X	O
2	Polarity of digital output in the last EX module digital output	UNSIGNED16	0	rw (read-write)	X	O
...						
8	Polarity of digital output in the last EX module	UNSIGNED16	0	rw (read-write)	X	O

Polarity

Summary of properties of this output are shown in the following table:

State	Description
1 (Output is reversed)	0 V = 1 24 V = 0
0 (Output is not reversed)	0 V = 0 24 V = 1

◆ Object 6306h: Fallback Mode Output

This object sets actions when wires are disconnected or control stops.

State	Description
0	Retain the value
1	Fallback value (defined by the object 6307h)

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 16 bit output)	UNSIGNED8	15	ro (read only)	X	O
1	HTB fallback mode	UNSIGNED16	0xFFFF	rw (read-write)	X	O
2	Fallback mode of the last EX module	UNSIGNED16	0xFFFF	rw (read-write)	X	O
...						
8	Fallback mode of the last EX module	UNSIGNED16	0xFFFF	rw (read-write)	X	O

◆ Object 6307h: Fallback Value Output

This object indicates fallback value applied by the output if an internal or communication failure occurs and when the object 6306h is set to 1 for the corresponding bit.
If the object 6308h is 0, fallback does not run.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub-index number (number of 16 bit outputs)	UNSIGNED8	15	ro (read only)	X	O
1	HTB fallback value	UNSIGNED16	0	rw (read-write)	X	O
2	Fallback value of the first EX module	UNSIGNED16	0	rw (read-write)	X	O
...						
8	Fallback value of the last EX module	UNSIGNED16	0	rw (read-write)	X	O

◆ Object 6308h: Output Mask

This object sets the mask for output.

State	Description
0	Holds the current output value
1	Approve writing to output (corresponds to the value of the object 6200h or 6300h)

NOTE

- The I/O status indicator LED continues to display the actual power status of a connected HTB. It is not affected by this object.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (number of 16 bit output)	UNSIGNED8	15	ro (read only)	X	O
1	HTB output mask	UNSIGNED16	0xFFFF	rw (read-write)	X	O
2	Output mask of the first EX module	UNSIGNED16	0xFFFF	rw (read-write)	X	O
...						
8	Output mask of the last EX module	UNSIGNED16	0xFFFF	rw (read-write)	X	O

◆ Object 6401h: Analog Input

This object reports the analog input value.

Sub index management:

- One sub index is used in each channel.
- EX module and its sub index are given a number in the order closer to HTB.

NOTE

- These rules are applied to all the objects related to analog input.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Analog channel number	UNSIGNED8	24	ro (read only)	X	X
1	Analog input value of CH1 in the first analog module	UNSIGNED16	0	ro (read only)	O	X
...						
18	Analog input value of the last analog module	UNSIGNED16	0	ro (read only)	O	X

◆ Object 6411h: Analog Output

This object writes the value of analog output.

Sub index management:

- One sub index is used in each channel.
- EX module and its sub index are given a number in the order closer to HTB.

NOTE • These rules are applied to all the objects related to analog output.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Analog channel number	UNSIGNED8	24	ro (read only)	X	X
1	Write analog output value of CH1 in the first analog module CH1.	INTEGER 16	0	rw (read-write)	O	X
...						
18	Write the last analog output value of the last analog module	INTEGER 16	0	rw (read-write)	O	X

◆ Object 6421h: Analog input trigger settings

This object sets trigger events of analog input.

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Analog channel number	UNSIGNED8	24	ro (read only)	X	O
1	Trigger events of the first analog input module	UNSIGNED8	7	rw (read-write)	X	O
...						
18	Trigger events of the last analog input module	UNSIGNED8	7	rw (read-write)	X	O

◆ Object 6422h: Analog Input PDO Transmission Channel Number

This object stores the PDO transmission channel number.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED8	1	ro (read only)	X	X
1	Channel number that generates PDO transmission	UNSIGNED32	-	ro (read only)	O	X

◆ Object 6423h: Analog input enabled/disabled

This object enables trigger events of analog input. If the value is FALSE, PDO is not transmitted.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
-	-	BOOLEAN	FALSE	rw (read-write)	X	X

◆ Object 6424h: Analog input thresholds high

This object inputs the maximum analog input thresholds.

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Analog channel number	UNSIGNED 8	24	ro (read only)	X	O
1	Maximum threshold of the first analog input module	INTEGER32	0	rw (read-write)	X	O
...						
18	Maximum threshold of the last analog input module	INTEGER32	0	rw (read-write)	X	O

◆ Object 6425h: Analog input thresholds low

This object inputs the minimum analog input thresholds.

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Analog channel number	UNSIGNED8	24	ro (read only)	X	O
1	Minimum threshold of the first analog input module	INTEGER32	0	rw (read-write)	X	O
...						
18	Minimum threshold of the last analog input module	INTEGER32	0	rw (read-write)	X	O

◆ Object 6426h: Analog input delta value

This object inputs a delta value of the analog input module.

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Analog channel number	UNSIGNED8	24	ro (read only)	X	O
1	Delta value of the first analog input module	UNSIGNED32	0	rw (read-write)	X	O
...						
18	Delta value of the last analog input module	UNSIGNED32	0	rw (read-write)	X	O

◆ Object 6443h: Analog Output Fallback Mode

This object instructs the fallback mode applied by the output when an internal or communication failure occurs.

0: Retain the value

1: Fallback Value

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number (analog input number)	UNSIGNED8	24	ro (read only)	X	O
1	Fallback mode of CH1	UNSIGNED8	1	rw (read-write)	X	O
...						
E	Fallback mode of the last channel	UNSIGNED8	1	rw (read-write)	X	O

◆ Object 6444h: Analog Output Fallback Value

This object indicates fallback value applied by the output if an internal or communication failure occurs and when the object 6443h is set to 1 for the corresponding sub index 1.

Properties

Object properties:

Sub index	Description	Data Type	Default	Access	PDO mapping	Backup
0	Sub index number	UNSIGNED8	24	ro (read only)	X	O
1	Fallback value of CH1	INTEGER32	0	rw (read-write)	X	O
...						
E	Fallback value of the last channel	INTEGER32	0	rw (read-write)	X	O

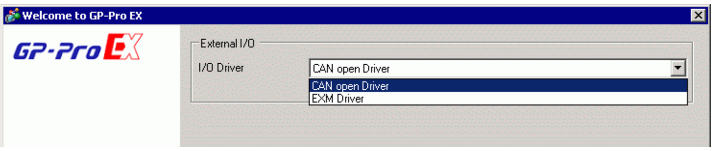
30.7.11 To connect third-party slaves

■ Setup Procedure

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

NOTE

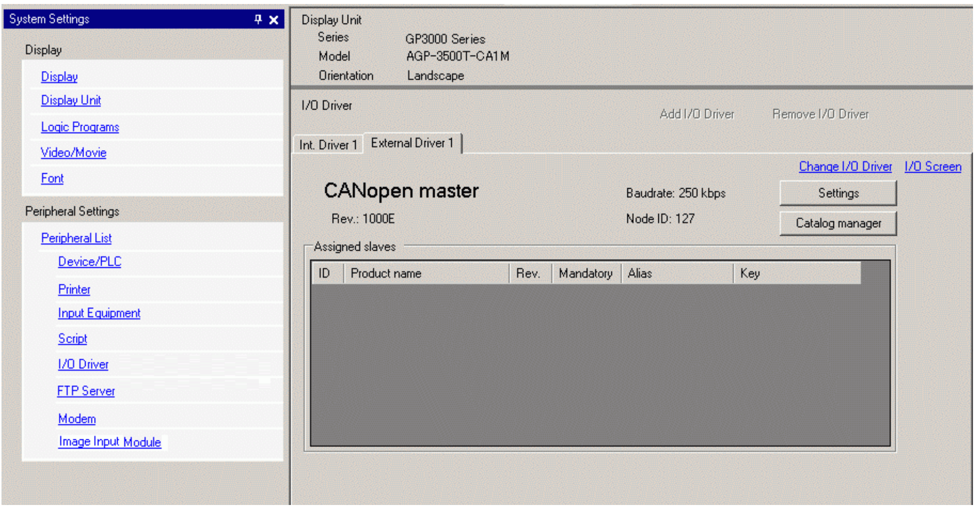
- When using LT models, select "CANopen Driver" for the I/O driver.



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

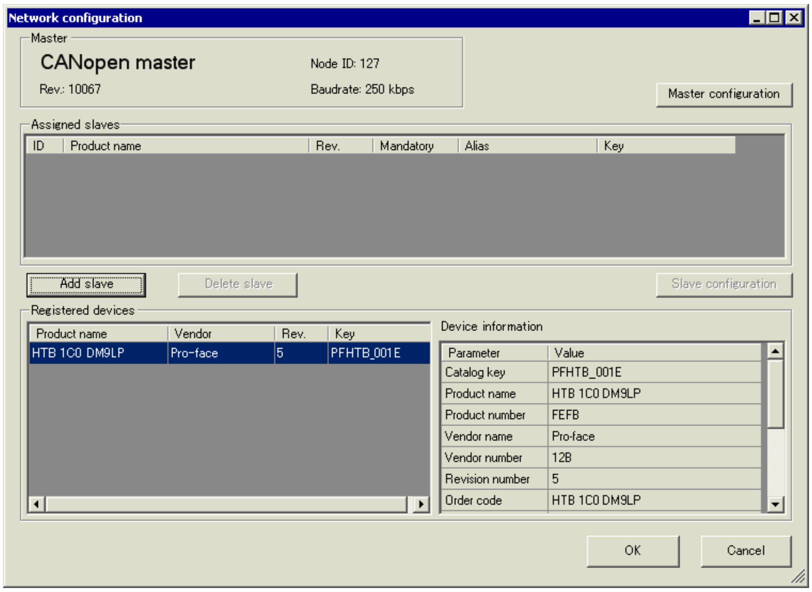
Baudrate: 250 kbps

Node ID: 127

Assigned slaves

ID	Product name	Rev.	Mandatory	Alias	Key

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



Master

CANopen master

Rev.: 10067

Node ID: 127

Baudrate: 250 kbps

Assigned slaves

ID	Product name	Rev.	Mandatory	Alias	Key

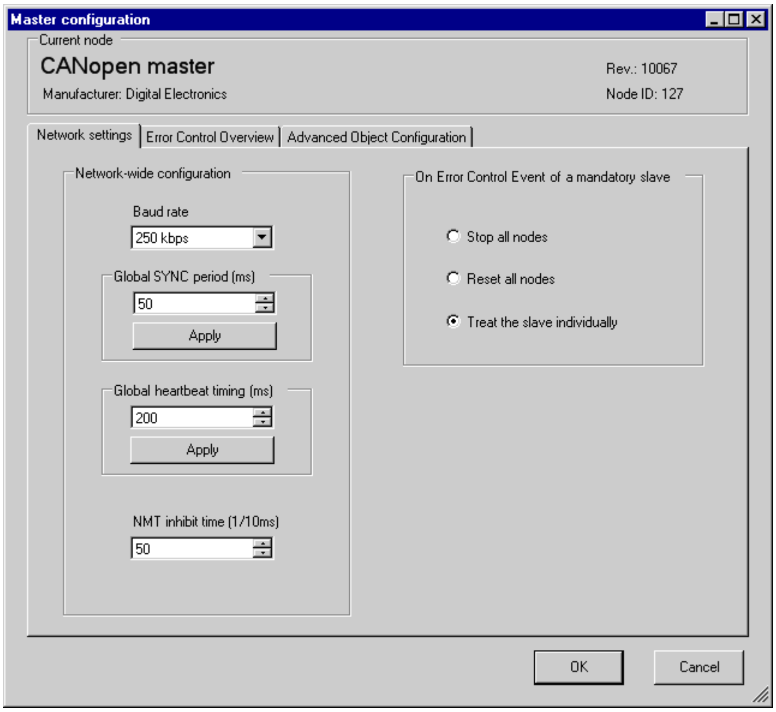
Registered devices

Product name	Vendor	Rev.	Key
HTB 1C0 DM9LP	Pro-face	5	PFHTB_001E

Device information

Parameter	Value
Catalog key	PFHTB_001E
Product name	HTB 1C0 DM9LP
Product number	FEFB
Vendor name	Pro-face
Vendor number	12B
Revision number	5
Order code	HTB 1C0 DM9LP

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.: 10067

Node ID: 127

Network settings | Error Control Overview | Advanced Object Configuration

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

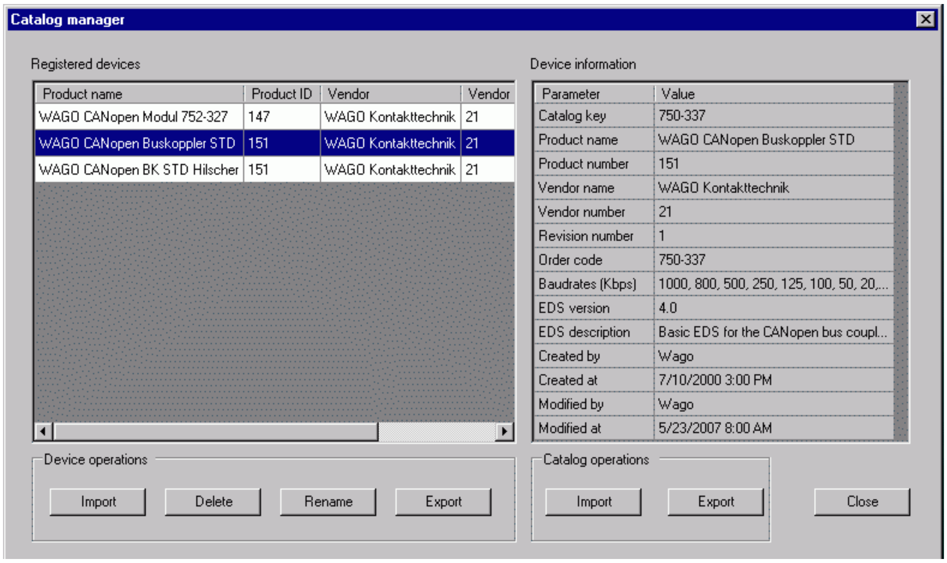
● Treat the slave individually

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

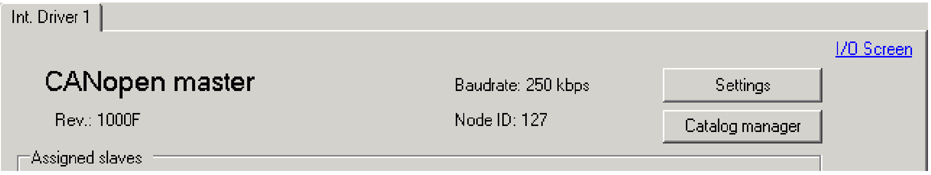
Parameter	Value
Catalog key	750-337
Product name	WAGO CANopen Buskoppler STD
Product number	151
Vendor name	WAGO Kontakttechnik
Vendor number	21
Revision number	1
Order code	750-337
Baudrates (Kbps)	1000, 800, 500, 250, 125, 100, 50, 20,...
EDS version	4.0
EDS description	Basic EDS for the CANopen bus 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.

☞ "30.7.7 [I/O Driver] Settings Guide" (page 30-204)

7 Click [Settings].


Int. Driver 1

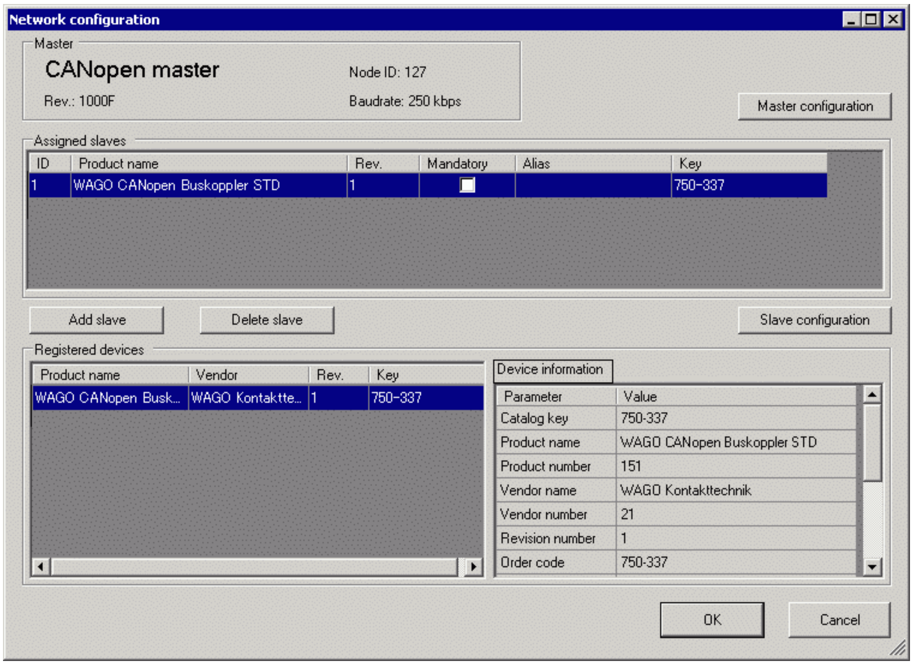
CANopen master

Rev.: 1000F

Baudrate: 250 kbps

Node ID: 127

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



Network configuration

Master

CANopen master

Rev.: 1000F

Node ID: 127

Baudrate: 250 kbps

Assigned slaves

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

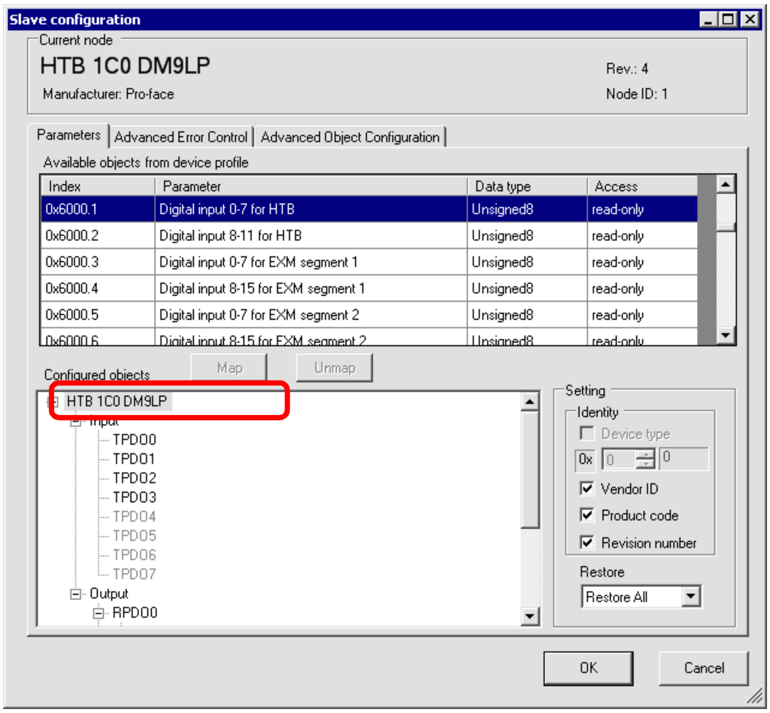
Registered devices

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.



Slave configuration

Current node

HTB 1C0 DM9LP

Manufacturer: Pro-face

Rev.: 4

Node ID: 1

Available objects from device profile

Index	Parameter	Data type	Access
0x6000.1	Digital input 0-7 for HTB	Unsigned8	read-only
0x6000.2	Digital input 8-11 for HTB	Unsigned8	read-only
0x6000.3	Digital input 0-7 for EXM segment 1	Unsigned8	read-only
0x6000.4	Digital input 8-15 for EXM segment 1	Unsigned8	read-only
0x6000.5	Digital input 0-7 for EXM segment 2	Unsigned8	read-only
0x6000.6	Digital input 8-15 for EXM segment 2	Unsigned8	read-only

Configured objects

```
HTB 1C0 DM9LP
  Input
    TPDO0
    TPDO1
    TPDO2
    TPDO3
    TPDO4
    TPDO5
    TPDO6
    TPDO7
  Output
    RPDO0
```

Setting

Identity

□ Device type

0x 0 0

☑ Vendor ID

☑ Product code

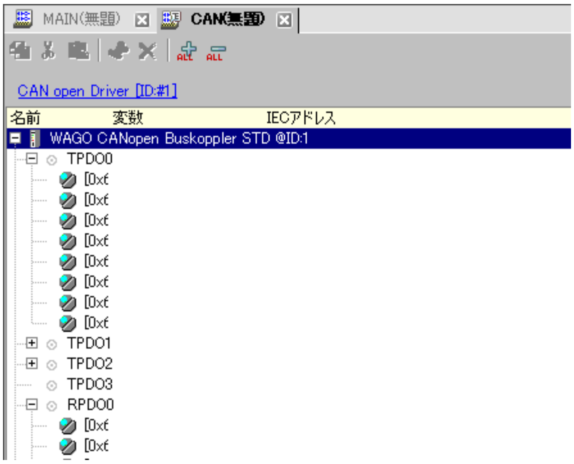
☑ Revision number

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 information on how to assign variables, refer to the following.

☞ "30.7.3 Allocating I/O (Common)" (page 30-163)



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