



IFD9502

DeviceNet Slave Communication Module Application Manual



<http://www.delta.com.tw/industrialautomation>

Warning

- ✓ Please read this instruction carefully before use and follow this instruction to operate the device in order to prevent damages on the device or injuries to staff.
- ✓ Switch off the power before wiring.
- ✓ IFD9502 is an OPEN TYPE device and therefore should be installed in an enclosure free of airborne dust, humidity, electric shock and vibration. The enclosure should prevent non-maintenance staff from operating the device (e.g. key or specific tools are required for operating the enclosure) in case danger and damage on the device may occur.
- ✓ IFD9502 is to be used for controlling the operating machine and equipment. In order not to damage it, only qualified professional staff familiar with the structure and operation of IFD9502 can install, operate, wire and maintain it.
- ✓ DO NOT connect input AC power supply to any of the I/O terminals; otherwise serious damage may occur. Check all the wirings again before switching on the power and DO NOT touch any terminal when the power is switched on. Make sure the ground terminal ⊕ is correctly grounded in order to prevent electromagnetic interference.

Table of Contents

1	INTRODUCTION.....	3
1.1	IFD9502 Brief	3
1.2	Features	3
2	COMPONENTS	4
2.1	Product Profile and Outline	4
2.2	DeviceNet Connector	4
2.3	Address Setup Rotary Switch.....	5
2.4	Function Setup DIP Switch.....	5
2.5	Communication ports on IFD9502.....	6
3	FUNCTIONS OF IFD9502	7
3.1	When IFD9502 is connected to Delta VFD series AC motor drive	7
3.2	When IFD9502 is connected to Delta DVP series programmable logic controller	10
3.3	When IFD9502 is connected to Delta DTA temperature controller.....	13
3.4	When IFD9502 is connected to Delta ASD-A series servo drive.....	16
3.5	When IFD9502 is connected to Delta DOP-A series human machine interface	19
3.6	When IFD9502 is connected to custom equipment.....	22
4	LED INDICATORS & TROUBLE-SHOOTING.....	26
4.1	Network Status LED	26
4.2	Module Status LED.....	26
4.3	Scan Port LED.....	26
5	DEVICENET OBJECTS IFD9502 SUPPORTS	27

5.1	DeviceNet Object.....	27
6	CONNECTION OF IFD9502 WITH OTHER EQUIPMENT	38
6.1	Communication Wirings When Connected to Equipment Through PORT 1.....	38
6.2	Communication Wirings When Connected to Equipment Through PORT 2.....	39

1 Introduction

1. To make sure that you are able to correctly install and operate IFD9502, please read this chapter carefully before starting to use IFD9502 and keep this handy for your quick reference.
2. This chapter only provides introductory information and guidelines on IFD9502. Details of DeviceNet protocol are not included. For more information on DeviceNet protocol, please refer to relevant references or literatures.
3. IFD9502 is defined as DeviceNet slave communication module to be used on the connection between DeviceNet network and Delta programmable logic controller, Delta AC motor drive, Delta servo drive, Delta temperature controller and Delta human machine interface. In addition, the custom function of IFD9502 allows the custom equipment with Modbus protocol to connect to DeviceNet network.

1.1 IFD9502 Brief

1. MODULE STATUS indicator and NETWORK STATUS indicator display the connection status between IFD9502 and DeviceNet. SCAN PORT indicator displays the connection status between IFD9502 and the equipments. For more details on LED indicators, see 12.4.
2. IFD9502 sets up its node address in DeviceNet by two rotary switches. For more details on the switches, see 12.2.
3. Functions of DIP switches: selecting equipments connected to IFD9502, selecting communication port of IFD9502, setting up the baud rate between IFD9502 and the master. For more details on DIP switches, see 12.2.
4. DeviceNet interface connects IFD9502 to DeviceNet network. For more details, see 12.2.
5. The communication ports allows IFD9502 to connect with Delta programmable logic controller, Delta AC motor drive, Delta temperature controller, Delta servo drive, Delta human machine interface and equipment with Modbus protocol. For more details, see 12.2.

1.2 Features

1. Supports Group 2 only servers
2. Supports explicit connection in the pre-defined master/slave connection group
3. Supports polling
4. Supports EDS files in DeviceNet network configuration tools
5. Specifications:

DeviceNet connector

Type	Removable connector (5.08mm)
Transmission method	CAN
Transmission cable	2 communication cables, 2 power cables and 1 shielded cable
Electrical isolation	500V DC

Communication

Message type	I/O polling
	Explicit
Baud rate	125 k bps (bit/sec)
	250 k bps (bit/sec)
	500 k bps (bit/sec)

Electrical specification

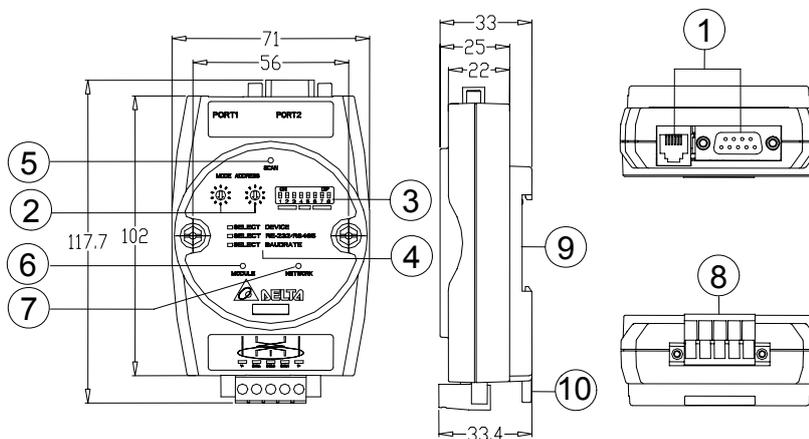
DeviceNet voltage	11 ~ 25V DC (Network power input connector)
Safety standard	Under EN50178 standard
Certifications	CE certified and UL certified

Environment

Operation temperature	-4 ~ 122°F (-10 ~ 50°C)
Storage temperature	-4 ~ 140°F (-20 ~ 60°C)
Humidity	<90% (under normal pressure)
Altitude	Max.: 1,000m
Shock/vibration immunity	0.5G 9 ~ 200Hz

2 Components

2.1 Product Profile and Outline



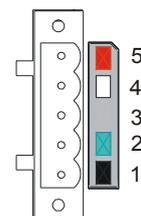
Unit: mm

1. Communication ports	6. MS (Module Status) indicator
2. Address setup rotary switches	7. NS (Network Status) indicator
3. Function setup DIP switches	8. DeviceNet connector
4. Descriptions for DIP switches	9. DIN rail
5. SP (Scan Port) indicator	10. DIN rail clip

2.2 DeviceNet Connector

To connect with DeviceNet network, you can use the connector enclosed with IFD9502 or any connectors you can buy in the store for wiring.

PIN	Signal	Color	Description
1	V-	Black	0V DC
2	CAN_L	Blue	Signal-
3	SHIELD	-	Shielded cable
4	CAN_H	White	Signal+
5	V+	Red	24V DC



2.3 Address Setup Rotary Switch

The two rotary switches SW1 and SW2 set up the node address on DeviceNet in decimal form. Setup range: 00 ~ 63 (64 ~ 99 are forbidden).



1. Example:

If you need to set the node address of IFD9502 as 26, simply switch the corresponding rotary switch of $X10^1$ to “2” and the corresponding rotary switch of $X10^0$ to “6”.

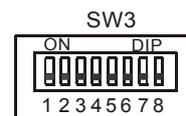
Address setting	Description
0 ~ 63	Valid DeviceNet node address
64 ~ 99	Invalid DeviceNet node address

2. Note:

The changed values on SW1 or SW2 are only valid when IFD9502 is re-powered. When IFD9502 is operating, changing the set value of node address will be invalid.

2.4 Function Setup DIP Switch

The DIP switch SW3 is to be used on the equipment connected to IFD9502, the selection of communication ports and setting up the baud rate of IFD9502 and the master in DeviceNet.



1. Selecting equipment connected to IFD9502

PIN 3	PIN 2	PIN 1	Equipment
Off	Off	On	AC motor drive
Off	On	Off	Programmable logic controller
Off	On	On	Temperature controller
On	Off	Off	Servo drive
On	Off	On	Human machine interface
On	On	Off	Custom equipment
On	On	On	Configuration mode

- Example

If the equipment connected to IFD9502 is Delta servo drive, you only need to switch PIN 3 in SW3 to “On” and PIN 1 and PIN 2 to “Off” and re-power IFD9502.

- Note:

The changed setting of DIP switch is only valid when IFD9502 is re-powered. When IFD9502 is operating, changing the setting of DIP switch will be invalid.

2. Selecting IFD9502 communication mode

PIN 5	PIN 4	Communication mode
Off	Off	RS-485
On	On	RS-232
Off	On	Incorrect setting
On	Off	

- Note

The changed setting of communication mode is only valid when IFD9502 is re-powered. When IFD9502 is operating, changing the setting of communication mode will be invalid.

3. Setting up baud rate

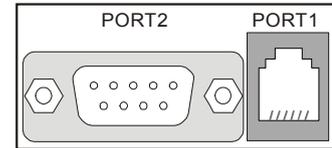
PIN 8	PIN 7	PIN 6	Baud rate of slave device
Reserved	Off	Off	125k bps
	Off	On	250k bps
	On	Off	500k bps
	On	On	Auto baud rate detection

- Note:

The changed setting of the baud rate of DeviceNet is only valid when IFD9502 is re-powered. When IFD9502 is operating, changing the baud rate will be invalid.

2.5 Communication ports on IFD9502

The communication ports on IFD9502 are used for the connection with the equipment (Delta programmable logic controller, Delta AC motor drive, Delta temperature controller, Delta servo drive, Delta human machine interface and custom equipment).



1. PORT 1 PIN definition

PORT 1 sketch	PIN	Description
	1	N.C.
	2	GND
	3	DATA-
	4	DATA+
	5	N.C.
	6	N.C.

- Note:

PORT 1 supports RS-485 communication only.

2. PORT 2 PIN definition

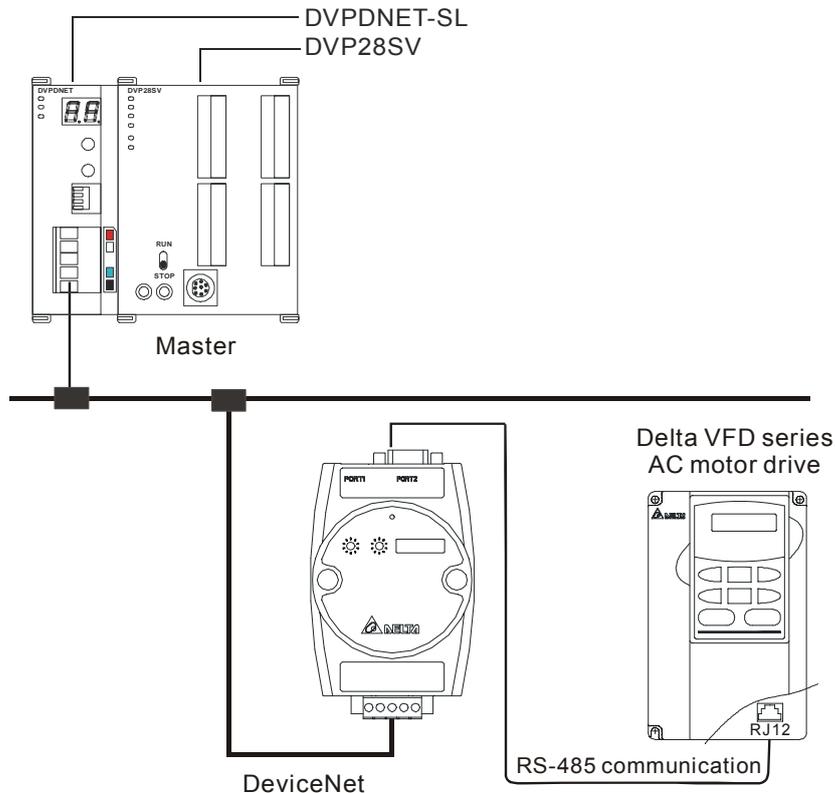
PORT 2 sketch	PIN	RS-232	RS-485
	1	N.C.	N.C.
	2	RXD	N.C.
	3	TXD	DATA-
	4	N.C.	N.C.
	5	GND	N.C.
	6	N.C.	N.C.
	7	N.C.	N.C.
	8	N.C.	DATA+
	9	N.C.	N.C.

- Note

PORT 2 supports RS-232 and RS-485 communication only.

3 Functions of IFD9502

3.1 When IFD9502 is connected to Delta VFD series AC motor drive



1. Baud rate and the setting of communication format
 Before connecting Delta AC motor drive to the BUS, first set up the node address of the AC motor drive as 01, baud rate as 38,400bps and communication format as 8, N, 2; RTU (the format is fixed; other formats will be invalid).
2. Modifying parameters in AC motor drive in the configuration software
 When the configuration software is used for modifying a parameter in AC motor drive, IFD9502 will read the maximum value, minimum value and read/write attribute of the parameter and decide if the value modified by the user falls within the modifiable range and whether to modify the parameter.
3. Reading and modifying parameters
 - IFD9502 allows you to inquire Delta AC motor drive by explicit messages. The format of the inquiry is shown in the table below.

Byte position	Data written into AC motor drive	Data read from AC motor drive
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [0]+Service code [0x10]	R/R [0]+Service code [0x0E]
2	Class ID [0x0F]	Class ID [0x0F]
3	Instance ID LSB	Instance ID LSB
4	Instance ID MSB	Instance ID MSB
5	Attribute ID	Attribute ID
6	Service data LSB	N/A
7	Service data MSB	N/A

- Format of messages responded

Byte position	Data written into AC motor drive	Data read from AC motor drive
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [1]+Service code [0x10]	R/R [1]+Service code [0x0E]
2	N/A	Response data LSB
3	N/A	Response data MSB

- Note

When modifying parameters in the parameter table of the device, make sure that the parameter allows you to modify it before you modify it.

4. I/O data mapping (Default)

- AC motor drive → DeviceNet master

Node address of AC motor drive	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'2101	-		Status of command control						-		LED status of AC motor drive					
H'2102	Displaying frequency command															

- DeviceNet master → AC motor drive

Node address of AC motor drive	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'2000	Control command of AC motor drive															
H'2001	Frequency command of AC motor drive															

- Explanation

- When the equipment connected to IFD9502 is an AC motor drive, the length of data downloaded from IFD9502 to the AC motor drive is preset as 2 words at addresses H'2000 and H'2001. IFD9502 is able to download maximum 8 words to AC motor drive. The length of data uploaded to IFD9502 from AC motor drive is preset as 2 words at addresses H'2101 and H'2102 and IFD9502 is able to be uploaded maximum 8 words.
- The length of data transmitted from IFD9502 to the AC motor drive is preset as 2 words. If you need to extend the length, you have to first set up Class 0x95>>Instance1>>Attribute2 as the destination value and next set up Attribute11 ~ Attribute 18. The length of data transmitted from the AC motor drive to IFD9502 is preset as 2 words. If you need to extend the length, you have to first set up Class 0x95>>Instance1>>Attribute3 as the destination value and next set up Attribute25 ~ Attribute32.
- Length of I/O data to be exchanged and address for I/O mapping can be modified through changing Class 0x95 as listed below. The modification will be valid after IFD9502 is re-powered. Maximum 16 words are allowed for I/O data exchange.
- If you are to return the I/O mapping to default setting, change Attribute1 of Instance1 of Class 0x95 into H'0001 and re-power IFD9502. Please note that doing so can only recover the current I/O data mapping in the AC motor drive.
- Class 0x95 Data Config
Instance0:

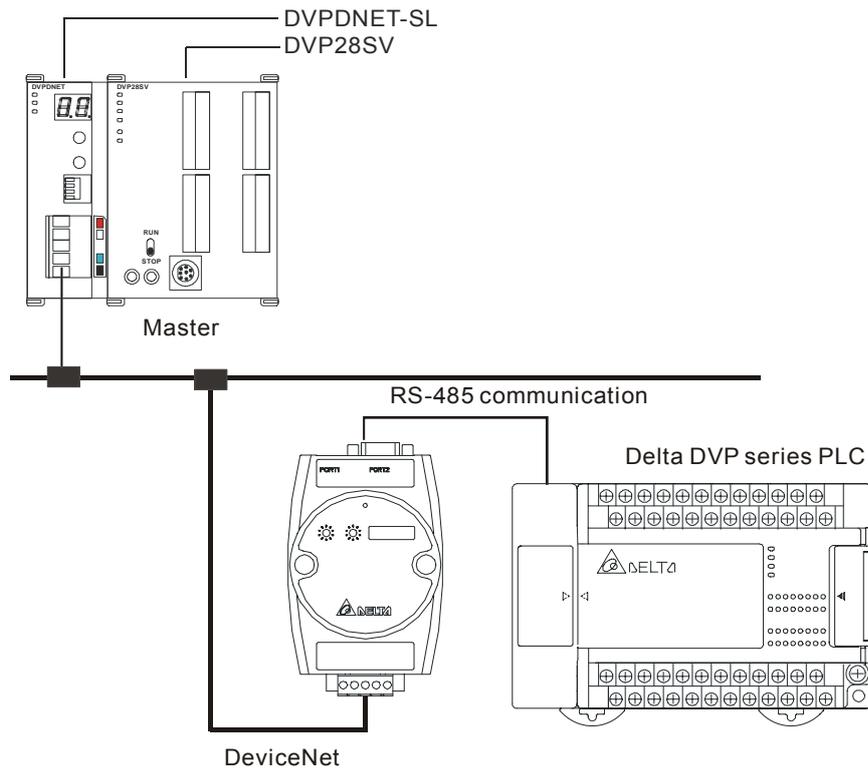
Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

Instance1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Reset parameter	USINT	H'0000
2	Get/Set	Length of input data	USINT	2 words
3	Get/Set	Length of output data	USINT	2 words
11	Get/Set	Data_in [0]	UINT	H'2000
12	Get/Set	Data_in [1]	UINT	H'2001
13	Get/Set	Data_in [2]	UINT	H'FFFF
14	Get/Set	Data_in [3]	UINT	H'FFFF
15	Get/Set	Data_in [4]	UINT	H'FFFF
16	Get/Set	Data_in [5]	UINT	H'FFFF
17	Get/Set	Data_in [6]	UINT	H'FFFF
18	Get/Set	Data_in [7]	UINT	H'FFFF
25	Get/Set	Data_out [0]	UINT	H'2101
26	Get/Set	Data_out [1]	UINT	H'2102
27	Get/Set	Data_out [2]	UINT	H'FFFF
28	Get/Set	Data_out [3]	UINT	H'FFFF
29	Get/Set	Data_out [4]	UINT	H'FFFF
30	Get/Set	Data_out [5]	UINT	H'FFFF
31	Get/Set	Data_out [6]	UINT	H'FFFF
32	Get/Set	Data_out [7]	UINT	H'FFFF

In the table above, Data_in refers to the data transmitted from DeviceNet master to AC motor drive, and Data_out refers to the data transmitted from the AC motor drive to DeviceNet master.

3.2 When IFD9502 is connected to Delta DVP series programmable logic controller



1. Baud rate and the setting of communication format

Before connecting the PLC to bus, set the node address of PLC as 01 and the communication format as 115,200 bps; 7, E, 1; ASCII (the format is fixed; other formats will be invalid).

2. Class 0x96 parameter

- Instance0

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance1 through 9

Instance ID	Description
1	S device of PLC
2	X device of PLC
3	Y device of PLC
4	M device of PLC
5	T device of PLC (bit device)
6	T device of PLC (word device)
7	C device of PLC (bit device)
8	C device of PLC (word device)
9	D device of PLC

All Instances in the table do not support any Attribute.

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x32	No	Yes	Read PLC data
0x33	No	Yes	Write PLC data

3. Format of inquiries and messages responded

- IFD9502 allows you to inquire PLC by explicit messages. The format of inquiry is shown in the table below.

Byte position	Data read from PLC	Data written into PLC
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [0]+Service code [0x32]	R/R [0]+Service code [0x33]
2	Class ID [0x96]	Class ID [0x96]
3	Instance ID	Instance ID
4	Position address LSB	Position Address LSB
5	Position address MSB	Position Address MSB
6	Number LSB	Service data LSB
7	Number MSB	Service data MSB

- Explanation

- “Position address” refers to the No. of the device in the register of PLC. For example, the position address of M0 is 0 and that of D100 is 100.
- “Number” refers to the number of PLCs. When the read device is a word device, “Number” refers to the number of words read; when the read device is a bit device, “Number” refers to the number of bits read.
- Bit devices, S, X, Y and M in DVP-PLC allow the reading of maximum 16 bits and writing in of only 1 bit (either 0 or 1). Devices T, C and D in DVP-PLC allow the reading of maximum 3 words and writing in of 1 word. See the table below.

Device in PLC	Max. length to be read allowed	Max. length to be written in allowed
S	16 bits	1 bit
X	16 bits	Not support
Y	16 bits	1 bit
M	16 bits	1 bit
T (bit device)	16 bits	1 bit
C (bit device)	16 bits	1 bit
T (word device)	3 words	1 word
C (word device)	3 words	1 word
D (word device)	3 words	1 word

- Format of messages responded

Byte position	Data read from PLC	Data written into PLC
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [1]+Service code [0x32]	R/R [1]+Service code [0x33]
2 ~ 7	Response data	N/A

- If error occurs during the communication, IFD9502 will send the error code to the master. See the table below for the definitions of error codes.

Error code		Definition
Byte 1	Byte 2	
0x08	0xFF	Service not supported (Invalid service code)
0x16	0xFF	Object polled does not exist (Illegal instance ID or class ID)
0x20	0x01	Communication instruction is illegal
0x20	0x02	The register address is illegal
0x20	0x03	The register No. in PLC exceeds the range
0x20	0x04	Cannot respond to the inquiry
0x20	0x07	IFD9502 and PLC communication error

4. I/O data mapping (default)

- PLC → DeviceNet master

Project	Start address of I/O data mapping	Default length of I/O data mapping	Max. Length of I/O data mapping	Unit	Description
PLC→DeviceNet	D408 (H'1198)	6	128	word	(M/8)+(D*2)<=Max size (256 bytes)
	M256 (H'0900)	10	256	bit	

- DeviceNet master → PLC

Project	Start address of I/O data mapping	Default length of I/O data mapping	Max. Length of I/O data mapping	Unit	Description
DeviceNet→PLC	D500 (H'11F4)	6	128	word	(M/8)+(D*2)<=Max size (256 bytes)
	M512 (H'0A00)	10	256	bit	

- Explanation

- When the equipment connected to IFD9502 is PLC, the length of data downloaded from IFD9502 to PLC is preset as 7 words at addresses D500 ~ D505 and M512 ~ M521. IFD9502 is able to download maximum 128 words to the word device D, T, and C and 256 bits to bit devices M, Y, and S in PLC, but the total data length downloaded to M and D should be no more than 256 bytes. The length of data uploaded from the PLC to IFD9502 is preset as 7 words at addresses D408 ~ D413 and M256 ~ M265. IFD9502 is able to be uploaded maximum 128 words from the word device D, T, and C and 256 bits to bit devices M, Y, and S in PLC but the total data length uploaded from M and D should be no more than 256 bytes.
- The length of I/O data to be exchanged and address for mapping can be modified through changing Class 0x97 as listed below. The modification will be valid after IFD9502 is re-powered. Maximum 16 words are allowed for I/O data exchange.
- If you are to return the I/O mapping to default setting, change Attribute1 of Instance1 of Class 0x97 into H0001 and re-power IFD9502. Please note that doing so can only recover

the current I/O data mapping in the PLC.

■ Class 0x97 Data Config

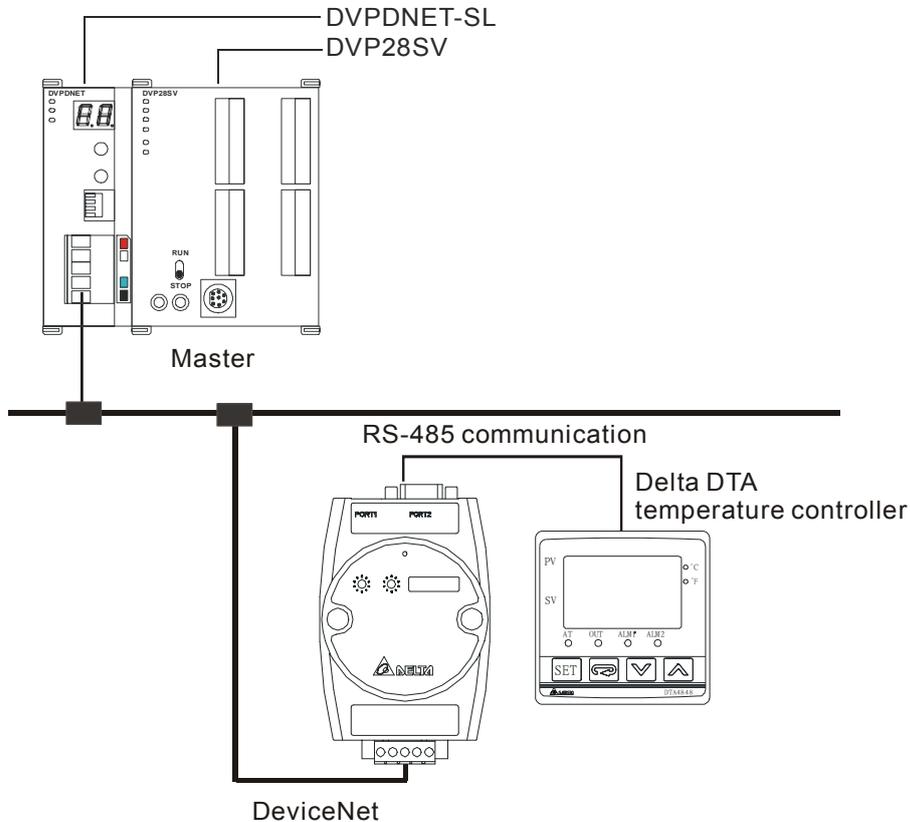
Instance0:

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

Instance1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Reset parameter	USINT	H'0000
2	Get/Set	M_dlen_in	UINT	A bits
3	Get/Set	D_dlen_in	UINT	6words
4	Get/Set	M_dlen_out	UINT	A bits
5	Get/Set	D_dlen_out	UINT	6words
6	Get/Set	M_in_start_adr	UINT	H'0100
7	Get/Set	D_in_start_adr	UINT	H'0198
8	Get/Set	M_out_start_adr	UINT	H'0200
9	Get/Set	D_out_start_adr	UINT	H'01F4
10	Get/Set	Comm Timeout	UINT	H'0020

3.3 When IFD9502 is connected to Delta DTA temperature controller



1. Baud rate and the setting of communication format

Before connecting Delta temperature controller to bus, set the node address of the temperature controller as 01 and the communication format as 38,400 bps; 7, E, 1; ASCII (the format is fixed; other formats will be

invalid). Before the communication with DTA series temperature controller, you have to set the content of H'471A as H'0001 to allow the write-in of communication.

2. Reading and modifying parameters

- IFD9502 allows you to inquire Delta temperature controller by explicit messages. The format of inquiry is shown in the table below:

Byte position	Data written into temperature controller	Data read from temperature controller
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [0]+Service code [0x10]	R/R [0]+Service code [0x0E]
2	Class ID [0x98]	Class ID [0x98]
3	Instance ID LSB	Instance ID LSB
4	Instance ID MSB	Instance ID MSB
5	Attribute ID	Attribute ID
6	Service data LSB	N/A
7	Service data MSB	N/A

- Format of messages responded:

Byte position	Data written into temperature controller	Data read from temperature controller
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [1]+Service code [0x10]	R/R [1]+Service code [0x0E]
2	N/A	Response data LSB
3	N/A	Response data MSB

3. I/O data mapping (Default)

- Temperature controller → DeviceNet master

Node address of AC motor drive	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'4700	Present temperature value (PV)															
H'4719	RUN/STOP															

- DeviceNet master → temperature controller

Node address of AC motor drive	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'4701	Set temperature value (SV)															
H'4719	RUN/STOP															

- Explanation

- When the equipment connected to IFD9502 is DTA series temperature controller, the length of data downloaded from IFD9502 to DTA is preset as 2 words at addresses H'4701 and H'4719. IFD9502 is able to download maximum 8 words. The length of data uploaded to IFD9502 from DTA is preset as 2 words at addresses H'4700 and H'4719. IFD9502 is able to be uploaded maximum 8 words.
- Length of I/O data to be exchanged and address for mapping can be modified through changing Class 0x99 as listed below. The modification will be valid after IFD9502 is re-powered. Maximum 16 words are allowed for I/O data exchange.
- If you are to return the I/O mapping to default setting, change Attribute1 of Instance1 of Class 0x99

into H'0001 and re-power IFD9502. Please note that doing so can only recover the current I/O data mapping in PLC.

■ Class 0x99 Data Config

Instance0:

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

Instance1 (DTA):

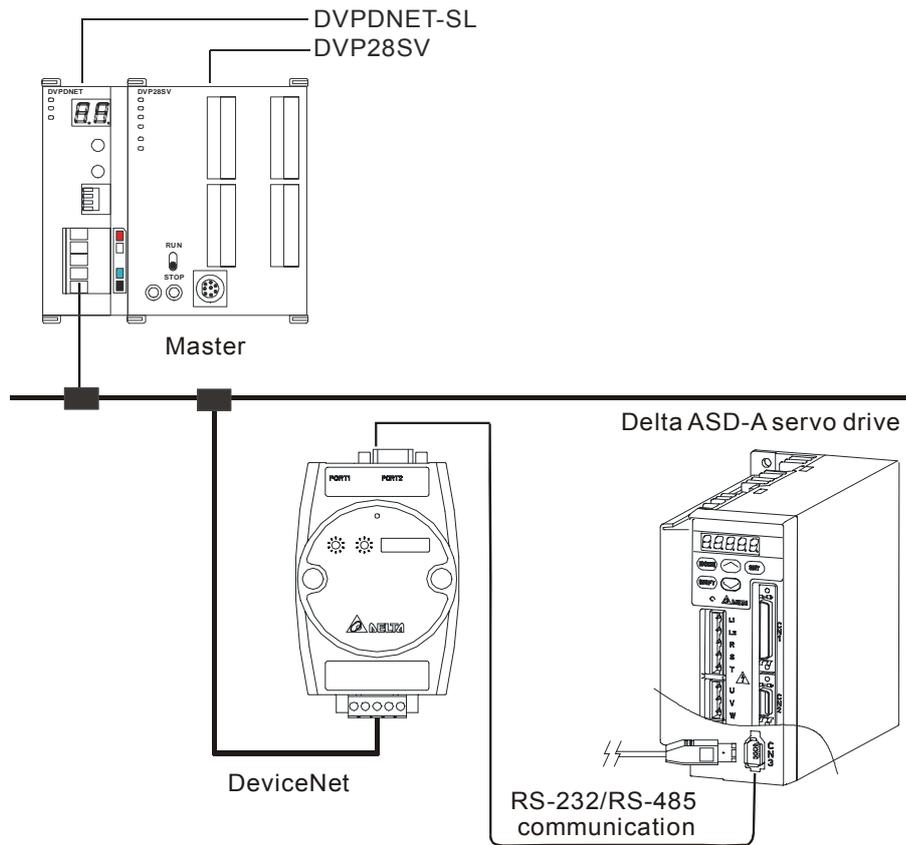
Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Reset parameter	USINT	H'0000
2	Get/Set	Input data length	USINT	2 words
3	Get/Set	Output data length	USINT	2 words
11	Get/Set	Data_in [0]	UINT	H'4701
12	Get/Set	Data_in [1]	UINT	H'4719 (for DTA)
13	Get/Set	Data_in [2]	UINT	H'FFFF
14	Get/Set	Data_in [3]	UINT	H'FFFF
15	Get/Set	Data_in [4]	UINT	H'FFFF
16	Get/Set	Data_in [5]	UINT	H'FFFF
17	Get/Set	Data_in [6]	UINT	H'FFFF
18	Get/Set	Data_in [7]	UINT	H'FFFF
25	Get/Set	Data_out [0]	UINT	H'4700
26	Get/Set	Data_out [1]	UINT	H'4719 (for DTA)
27	Get/Set	Data_out [2]	UINT	H'FFFF
28	Get/Set	Data_out [3]	UINT	H'FFFF
29	Get/Set	Data_out [4]	UINT	H'FFFF
30	Get/Set	Data_out [5]	UINT	H'FFFF
31	Get/Set	Data_out [6]	UINT	H'FFFF
32	Get/Set	Data_out [7]	UINT	H'FFFF

Data_in refers to the data transmitted from DeviceNet master to DTA. Data_out refers to the data transmitted from DTA to DeviceNet master.

Common Services

Service Code	Implemented for		Service name
	Class	Instance	
0x05	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_single
0x10	No	Yes	Set_Attribute_single

3.4 When IFD9502 is connected to Delta ASD-A series servo drive



1. Baud rate and the setting of communication format

Before connecting Delta servo drive to bus, set the node address of the servo drive as 01 and the communication format as 115,200; 7, E, 1; ASCII (the format is fixed; other formats will be invalid). Assume the equipment connected to IFD9502 is ASDA-A0121LA, you have to configure the following parameters before connecting the servo drive to IFD9502.

- P3-00 = H'0001 (set node address to 1)
- P3-01 = H'0005 (set baud rate to 115,200 bps)
- P3-02 = H'0001 (set communication format as ASCII,7,E,1)
- P3-05 = H'0002 (select RS-485 communication)
- P3-05 = H'0000 (select RS-232 communication)
- P3-06 = H'00FF (set DI1 ~ DI8 to valid communication control)

2. Reading and modifying parameters

- IFD9502 allows you to inquire Delta servo drive by explicit messages. The format of inquiry is shown in the table below.

Byte position	Data written into servo drive	Data read from servo drive
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [0]+Service code [0x10]	R/R [0]+Service code [0x0E]
2	Class ID [0x9A]	Class ID [0x9A]
3	Instance ID LSB	Instance ID LSB
4	Instance ID MSB	Instance ID MSB
5	Attribute ID	Attribute ID

Byte position	Data written into servo drive	Data read from servo drive
6	Service Data LSB	N/A
7	Service Data MSB	N/A

- Format of messages responded

Byte position	Data written into servo drive	Data read from servo drive
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [1]+Service code [0x10]	R/R [1]+Service code [0x0E]
2	N/A	Response data LSB
3	N/A	Response data MSB

- Note

When modifying parameters in the parameter table of Delta ASD-A servo drive, make sure that the parameter allows you to modify it before you modify it.

3. I/O data mapping (Default)

- Servo drive → DeviceNet master

Node address of servo drive	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'0409	-								Digital output DO1 ~ DO5							

- DeviceNet Master → Servo drive

Node address of servo drive	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'0407	-								Digital input DI1 ~ DI8							

- Explanation:

- When the equipment connected to IFD9502 is a servo drive, the length of data downloaded from IFD9502 to the servo drive is preset as 1 word at addresses H'0407 (corresponding address of DI1 ~ DI8). IFD9502 is able to download maximum 16 words. The length of data uploaded to IFD9502 from the servo drive is preset as 1 word at addresses H'0409 (corresponding address of DO1 ~ DO5). IFD9502 is able to be uploaded maximum 16 words.
- The length of I/O data to be exchanged and address for mapping can be modified through changing Class9B s listed below. The modification will be valid after IFD9502 is re-powered. Maximum 32 words are allowed for I/O data exchange.
- If you are to return the I/O mapping to default setting, change Attribute1 of Instance1 of Class 0x9B into H'0001 and re-power IFD9502. Please note that doing so can only recover the current I/O data mapping in servo drive.
- Class 0x9B Data Config

Instance0:

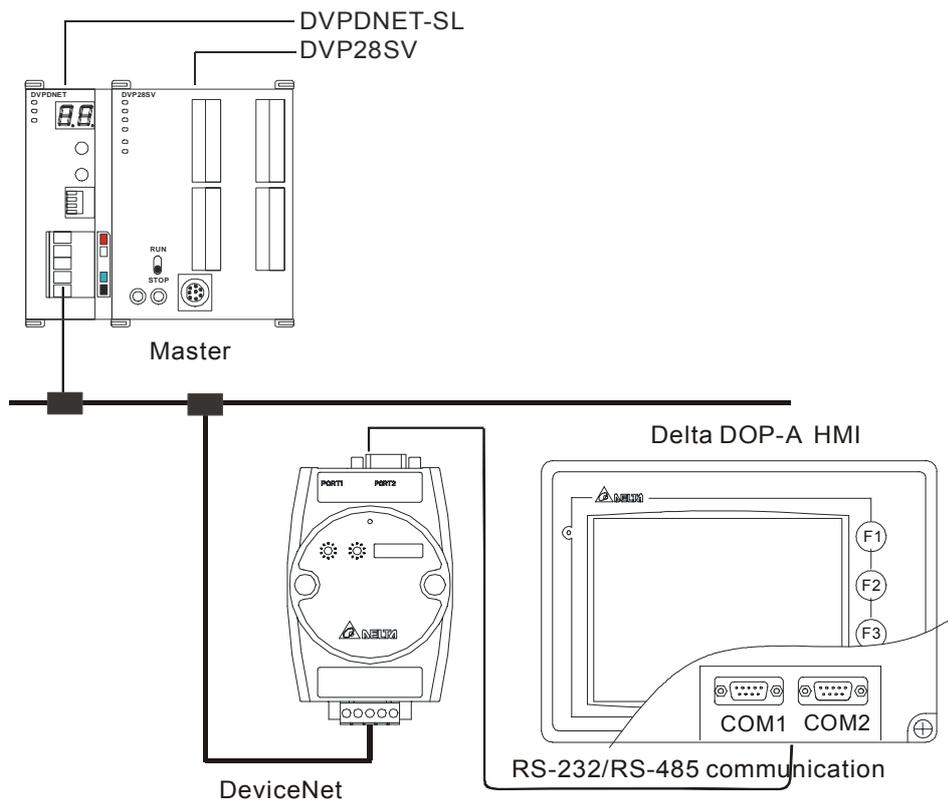
Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

Instance1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Reset parameter	USINT	H'0000
2	Get/Set	Input data length	USINT	1 words
3	Get/Set	Output data length	USINT	1 words
11	Get/Set	Data_in [0]	UINT	H'0407
12	Get/Set	Data_in [1]	UINT	H'FFFF
13	Get/Set	Data_in [2]	UINT	H'FFFF
14	Get/Set	Data_in [3]	UINT	H'FFFF
15	Get/Set	Data_in [4]	UINT	H'FFFF
16	Get/Set	Data_in [5]	UINT	H'FFFF
17	Get/Set	Data_in [6]	UINT	H'FFFF
18	Get/Set	Data_in [7]	UINT	H'FFFF
19	Get/Set	Data_in [8]	UINT	H'FFFF
20	Get/Set	Data_in [9]	UINT	H'FFFF
21	Get/Set	Data_in [10]	UINT	H'FFFF
22	Get/Set	Data_in [11]	UINT	H'FFFF
23	Get/Set	Data_in [12]	UINT	H'FFFF
24	Get/Set	Data_in [13]	UINT	H'FFFF
25	Get/Set	Data_in [14]	UINT	H'FFFF
26	Get/Set	Data_in [15]	UINT	H'FFFF
31	Get/Set	Data_out [0]	UINT	H'0409
32	Get/Set	Data_out [1]	UINT	H'FFFF
33	Get/Set	Data_out [2]	UINT	H'FFFF
34	Get/Set	Data_out [3]	UINT	H'FFFF
35	Get/Set	Data_out [4]	UINT	H'FFFF
36	Get/Set	Data_out [5]	UINT	H'FFFF
37	Get/Set	Data_out [6]	UINT	H'FFFF
38	Get/Set	Data_out [7]	UINT	H'FFFF
39	Get/Set	Data_out [8]	UINT	H'FFFF
40	Get/Set	Data_out [9]	UINT	H'FFFF
41	Get/Set	Data_out [10]	UINT	H'FFFF
42	Get/Set	Data_out [11]	UINT	H'FFFF
43	Get/Set	Data_out [12]	UINT	H'FFFF
44	Get/Set	Data_out [13]	UINT	H'FFFF
45	Get/Set	Data_out [14]	UINT	H'FFFF
46	Get/Set	Data_out [15]	UINT	H'FFFF

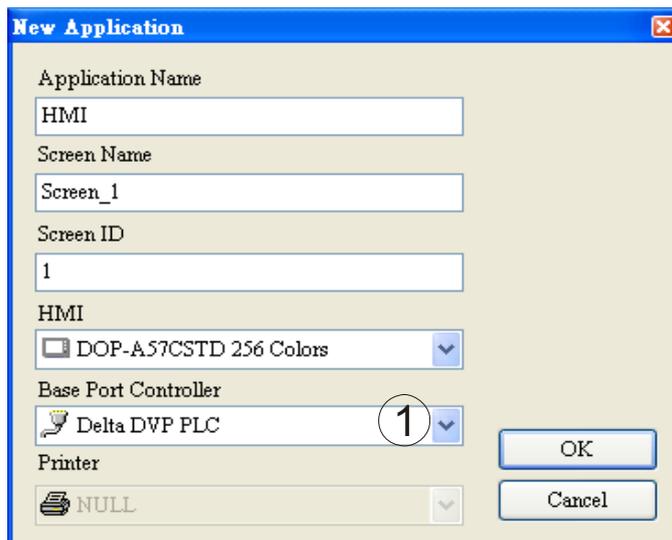
Data_in refers to data transmitted from DeviceNet master to servo drive. Data_out refers to data transmitted from the servo drive to DeviceNet master.

3.5 When IFD9502 is connected to Delta DOP-A series human machine interface



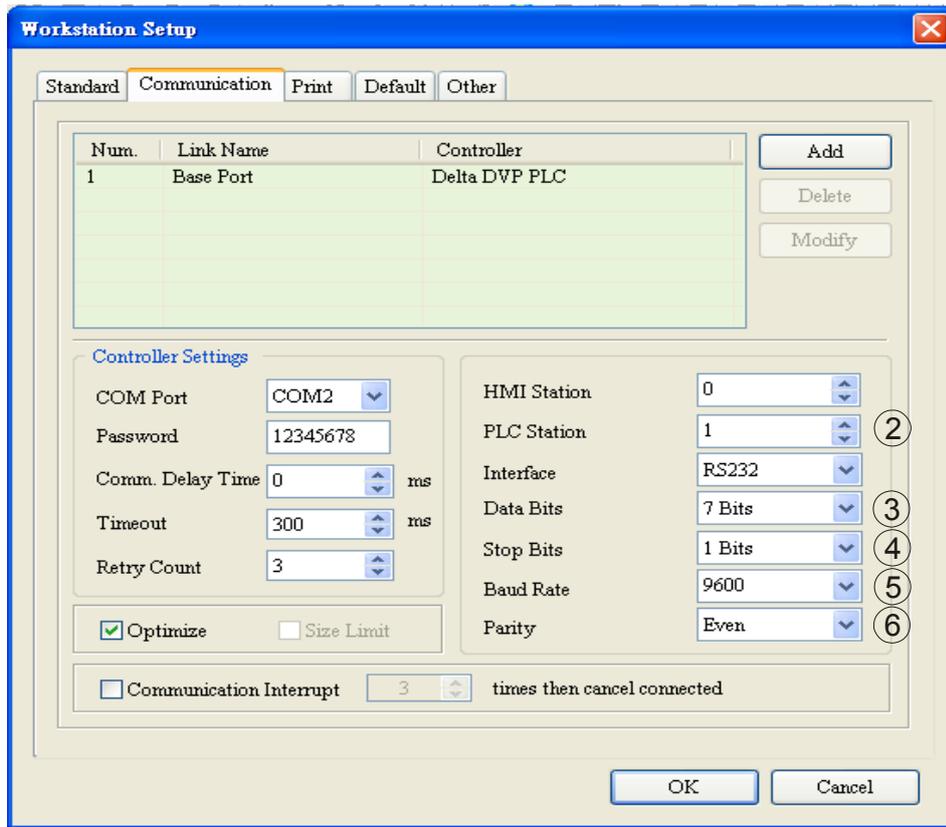
1. Baud rate and the setting of communication format

- Before connecting the Delta HMI to bus, set the node address of HMI as 01 and the communication format as 115,200; 7, E, 1; ASCII (the format is fixed; other formats will be invalid).
- When IFD9502 is connected to Delta DOP series HMI, please set up the baud rate and format following the procedure listed below.
 - Open software Screen Editor and select "File>>New". You will see the dialog box as below.

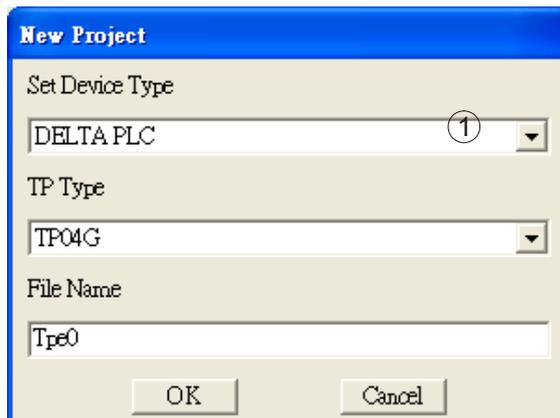


- Follow ① and set the Base Port Controller to "Delta DVP PLC". Click "OK" to create a new file.
- Select "Options>>Configuration>>Communication" and you will see the dialog box as

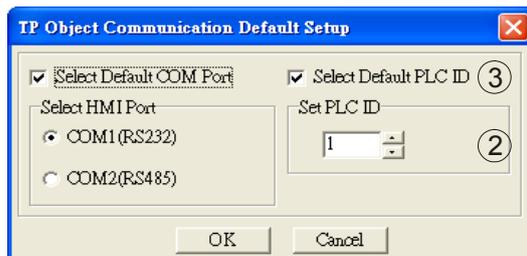
below.



- Follow ② ~ ⑥ and set the PLC Station to 1 and communication format to 115,200, 7, E, 1, ASCII. Click "OK".
- When IFD9502 is connected to TP04/TP02, please set up the baud rate and format following the procedure listed below.
 - Open TPEditor and select "File>>New". You will see the dialog box as below.



- Follow ① and set the Device Type to "Delta PLC". Click "OK" to create a new file.
- Select "Tools>>TP Object Communication Default Setting" and you will see the dialog box as below.



- Follow ② ③ and set PLC ID to “1”. Check “Select Default PLC ID” and click “OK” to complete the setting.

2. I/O data mapping (Default)

- DOP series HMI writes IFD9502

Node address of IFD9502	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D0	DOP writes register of IFD9502															
D1	DOP writes register of IFD9502															
D2	DOP writes register of IFD9502															
D3	DOP writes register of IFD9502															

- IFD9502 → DeviceNet master

Node address of IFD9502	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D0	Content of register D0															
D1	Content of register D1															
D2	Content of register D2															
D3	Content of register D3															

- DOP series HMI reads IFD9502

Node address of IFD9502	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D32	DOP reads register of IFD9502															
D33	DOP reads register of IFD9502															
D34	DOP reads register of IFD9502															
D35	DOP reads register of IFD9502															

- DeviceNet master → IFD9502

Node address of IFD9502	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
D32	Content of register D32															
D33	Content of register D33															
D34	Content of register D34															
D35	Content of register D35															

- Explanation

- When the equipment connected to IFD9502 is DOP series HMI, the length of data written to IFD9502 from DOP is preset as 4 words from the preset devices D0 ~ D3. The maximum length of data written from DOP series HMI to device D of IFD9502 is 32 words. The length of data in IFD9502 read by DOP series HMI is preset as 4 words from the preset devices D32 ~ D35. The maximum length of data in device D of IFD9502 read by DOP series HMI is

32 words.

- The length of I/O data to be exchanged can be modified through changing Class9C as listed below, but the start device of I/O mapping cannot be changed. The modification will be valid after IFD9502 is re-powered. Maximum 64 words are allowed for I/O data exchange.
- If you are to return the I/O mapping to default setting, change Attribute1 of Instance1 of Class 0x9C into H'0001 and re-power IFD9502. Please note that doing so can only recover the current I/O data mapping in HMI.
- Class 0x9C Data Config

Instance0:

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

Instance1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Reset parameter	USINT	H'0000
2	Get/Set	Input data length	USINT	4 words
3	Get/Set	Output data length	USINT	4 words

3.6 When IFD9502 is connected to custom equipment

In the custom mode, IFD9502 can still establish EXP/IO connection when there is no equipment connected to it.

1. Settings of the equipment connected to IFD9502 in custom mode

Before connecting the custom equipment to bus through IFD9502, you have to configure and check the following parameters.

- Configuration of the equipment: RS-485 or RS-232 communication
- Node address of the equipment: 01
- Baud rate of the equipment: 19,200bps
- Communication format of the equipment: 8, N, 2, RTU
- According to the actual need, configure the I/O mapping between IFD9502 and the equipment in the configuration software.

2. Reading and modifying parameters in the custom equipment

- IFD9502 allows you to inquire the custom equipment by explicit messages (fits in Modbus protocol). The format of inquiry is shown in the table below.

Byte position	Data written into custom equipment	Data read from custom equipment
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [0]+Service code [0x10]	R/R [0]+Service code [0x0E]
2	Class ID [0x9E]	Class ID [0x9E]
3	Instance ID LSB	Instance ID LSB
4	Instance ID MSB	Instance ID MSB

Byte position	Data written into custom equipment	Data read from custom equipment
5	Attribute ID	Attribute ID
6	Service Data LSB	N/A
7	Service Data MSB	N/A

- Format of messages responded:

Byte position	Data written into custom equipment	Data read from custom equipment
0	Frag [0]+XID+MAC ID	Frag [0]+XID+MAC ID
1	R/R [1]+Service code [0x10]	R/R [1]+Service code [0x0E]
2	N/A	Response data LSB
3	N/A	Response data MSB

3. Class 0x9E Custom_Parameter

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance 1: Parameter Instance 1 through N

Parameter attribute

Attribute ID	Access rule	Name	Data type
1	Get/Set	Parameter Value	-

Common Services

Service Code	Implemented for		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

- Explanation

- When you need to use DeviceNet Config software to read the parameters in the custom equipment, you have to first make sure that the equipment supports the Modbus function code, 0x03 (read), 0x06 (write) and so on.
- You have to acquire the Modbus address of the parameter before reading the parameter.
- How to read: Plus 0x0001 in Modbus address as Instance value and 0x0001 as Attribute value.
- Assume you read the Modbus address of the custom equipment as 0x011E, use DeviceNet Config software and read Instance = 0x011E + 0x0001 and Attribute = 0x01.

4. I/O data mapping (Default)

- Custom equipment → DeviceNet master

Node address of custom equipment	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'2101	Relevant to the equipment connected															
H'2102	Relevant to the equipment connected															

- DeviceNet master → custom equipment

Node address of custom equipment	b15	b14	b13	b12	b11	b10	b9	b8	b7	b6	b5	b4	b3	b2	b1	b0
H'2000	Relevant to the equipment connected															
H'2001	Relevant to the equipment connected															

- Explanation

- When the equipment connected to IFD9502 is custom equipment, the length of data downloaded from IFD9502 to the equipment is preset as 2 words at addresses H'2101 and H'2102. IFD9502 is able to download maximum 4 words. The length of data uploaded to IFD9502 from the equipment is preset as 2 words at addresses H'2000 and H'2001. IFD9502 is able to be uploaded maximum 4 words. The I/O mapping can be modified by changing Class9F.
- The length of I/O data to be exchanged and address for mapping can be modified through changing Class9F as listed below. The modification will be valid after IFD9502 is re-powered. Maximum 8 words are allowed for I/O data exchange.
- If you are to return the I/O mapping to default setting, change Attribute1 of Instance1 of Class 0x9F into H'0001 and re-power IFD9502. Please note that doing so can only recover the current I/O data mapping in the custom equipment.
- Class 0x9F Custom Data Config

Instance0:

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

Instance1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Reset parameter	USINT	H'0000
2	Get/Set	Input data length	USINT	2 words
3	Get/Set	Output data length	USINT	2 words
9	Get/Set	Data_in [0]	UINT	H'2000
10	Get/Set	Data_in [1]	UINT	H'2001
11	Get/Set	Data_in [2]	UINT	H'FFFF
12	Get/Set	Data_in [3]	UINT	H'FFFF
17	Get/Set	Data_out [0]	UINT	H'2101
18	Get/Set	Data_out [1]	UINT	H'2102
19	Get/Set	Data_out [2]	UINT	H'FFFF
20	Get/Set	Data_out [3]	UINT	H'FFFF

If IFD9502 fails to connect to DeviceNet by explicit messages due to the modification on the I/O mapping between IFD9502 and the equipment connected to it (e.g. dlen_in, dlen_out, data_in, data_out, etc), the I/O mapping between IFD9502 and the equipment connected to it will not be able to return to default setting through explicit messages.

- In the custom mode, DNAIFD9502 can still establish EXP/IO connection when there is no equipment connected to it and the I/O mapping of IFD9502 can be returned to default

setting.

Instance2 EEPROM read/write:

Attribute ID (decimal)	Access rule	Name	Default
30	Get/Set	AMD reset parameter	H'0000
71	Get/Set	PLC reset parameter	H'0000
90	Get/Set	DTA reset parameter	H'0000
150	Get/Set	ASDA reset parameter	H'0000
224	Get/Set	DOP reset parameter	H'0000
200	Get/Set	Custom reset parameter	H'0000

In the custom mode, if you are to return the I/O mapping of AC motor drive to default setting, change Attribute30 of Instance2 of Class 0x9F into H'0001 and re-power IFD9502.

In the custom mode, if you are to return the I/O mapping of PLC to default setting, change Attribute71 of Instance2 of Class 0x9F into H'0001 and re-power IFD9502.

In the custom mode, if you are to return the I/O mapping of temperature controller to default setting, change Attribute90 of Instance2 of Class 0x9F into H'0001 and re-power IFD9502.

In the custom mode, if you are to return the I/O mapping of servo drive to default setting, change Attribute150 of Instance2 of Class 0x9F into H'0001 and re-power IFD9502.

In the custom mode, if you are to return the I/O mapping of HMI to default setting, change Attribute224 of Instance2 of Class 0x9F into H'0001 and re-power IFD9502.

In the custom mode, if you are to return the I/O mapping of the custom equipment to default setting, change Attribute200 of Instance2 of Class 0x9F into H'0001 and re-power IFD9502.

Common Services:

Service Code	Implemented for		Service name
	Class	Instance	
005	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

Instance3 Modbus baud rate setting

Attribute ID (decimal)	Access rule	Name	Range (Note 1)	Default value (Note 1)
1	Get/Set	AMD baud rate setting	1 ~ 2	2
2	Get/Set	PLC baud rate setting	1 ~ 4	4
3	Get/Set	DT baud rate setting	1 ~ 2	2
4	Get/Set	ASDA baud rate setting	1 ~ 4	4
5	Get/Set	DOP baud rate setting	1 ~ 4	4
6	Get/Set	CUSTOM baud rate setting	1 ~ 4	1

Note 1: The baud rates of the codes are in the table below.

Code	Baud rate
1	19,200 bps
2	38,400 bps
3	57,600 bps
4	115,200 bps

4 LED Indicators & Trouble-shooting

There are 3 LED indicators on IFD9502, Network Status LED, Module Status LED and Scan Port LED, for displaying the connection status of the communication.

4.1 Network Status LED

LED status	Indication	How to deal with it?
Off	No power; Duplicating ID has not completed	<ol style="list-style-type: none"> 1. Check the power of IFD9502 and see if the connection is normal. 2. Check if the node communication on the BUS is normal. 3. Make sure at least 1 node is normally communicating with the network through IFD9502.
Green light flashes	Online, but not connected to DeviceNet	--
Green light on	Online, and connected to DeviceNet normally	--
Red light flashes	Online, but I/O connection time-out occurs.	--
Red light on	Network failure. Failed ID duplication or Bus-off.	<ol style="list-style-type: none"> 1. Make sure all node addresses on the BUS are not repeated. 2. Check if the network installation is normal. 3. Check if the baud rate of IFD9502 is consistent with that of the BUS. 4. Check if the station No. of IFD9502 is valid. 5. Check if your choice of switch on IFD9502 is consistent with the actual connected equipment. 6. Check if IFD9502 is correctly wired with the equipment.

4.2 Module Status LED

LED status	Indication	How to deal with it?
Off	No power; off-line	Check the power of IFD9502 and see if the connection is normal.
Green light flashes	Waiting for I/O data, no I/O data or PLC program is being edited.	IFD9502 has passed operation test and is waiting for I/O data
Green light on	I/O operation in progress normally.	--
Red light flashes	Configuration problem; IFD9502 is not connected to equipment	<ol style="list-style-type: none"> 1. Reset parameters in IFD9502. 2. Check if IFD9502 is correctly wired with the equipment.
Red light on	EEPROM is damaged, or other hardware error.	Send back to factory for repair.

4.3 Scan Port LED

LED status	Indication	How to deal with it?
Off	No power	Check the power of IFD9502 and see if the connection is normal.
Green light flashes	DNA-02 is reading the default value in the equipment. IFD9502 obtains the parameters from the equipment and	

LED status	Indication	How to deal with it?
	initializes some of the attributes.	
Green light on	Communication between DNA-02 and the equipment is normal.	
Red light flashes	CRC check fails, or the equipment sends back error information.	1. Check if the communication format of the equipment is correctly set up. 2. Check carefully if the installation is correct.
Red light on	Connection fails, or no connection.	1. Check if IFD9502 is correctly connected with the equipment. 2. Restart the connection and make sure the communication cable meets the specification.

5 DeviceNet Objects IFD9502 Supports

This section provides details of the objects, instances and attributes supported by IFD9502 in DeviceNet network.

5.1 DeviceNet Object

1. Object Classes

Class	Object
0x01	Identity
0x02	Message router
0x03	DeviceNet
0x05	Connection
0x0F	Parameter
0x93	Base Object
0x95	AMD config Object
0x96	PLC parameter
0x97	PLC config Object
0x98	DT parameter
0x99	DT config
0x9A	ASDA paramter
0x9B	ASDA config

2. Class 0x01 Identity

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT
2	Get	MaxInstance	UINT
3	Get	NumberOfInstances	UINT
6	Get	MaxIdClass	UINT
7	Get	MaxIdInstance	UINT

- Instance 1: Drive Instance

Attribute ID	Access rule	Name	Data type
1	Get	Vendor ID	UINT

Attribute ID	Access rule	Name	Data type
2	Get	DeviceType	UINT
3	Get	ProductCode	UINT
4	Get	Revision MajRev MinRev	USINT USINT
5	Get	Status	WORD
6	Get	Sn	UDINT
7	Get	ProdName StrLen ASCIIstr	USINT STRING

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x05	No	Yes	Reset
0x0e	Yes	Yes	Get_Attribute_Single
0x10	Yes	No	Find_Next_Object_Instance

3. Class 0x02 Message router

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT
6	Get	MaxIdClass	UINT
7	Get	MaxIdInstance	UINT

- Instance 1:

Attribute ID	Access rule	Name	Data type
2	Get	NumAvailable	UINT
3	Get	NumActive	UINT

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x0e	Yes	Yes	Get_Attribute_Single

4. Class 0x03 DeviceNet

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance 1: Drive Instance

Attribute ID	Access rule	Name	Data type
1	Get	MAC ID	USINT
2	Get	BaudRate	USINT

Attribute ID	Access rule	Name	Data type
3	Get/Set	BusofInterrupt	BOOL
4	Get/Set	BusofCounter	USINT
5	Get	AllocationInfo AllocationChioce MasterNodeAddress	BYTE USINT
6	Get	MACIDSwitchChanged	BOOL
7	Get	BaudRateSwitchChanged	BOOL
8	Get	MACIDSwitchValue	USINT
9	Get	BaudRateSwitchValue	USINT

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single
0x4B	No	Yes	Allocate_Master/Slave_Connection_Set
0x4C	No	Yes	Release_Master/Slave_Connection_Set

5. Class 0x05 Connection

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance 1: Master/Slave Explicit Message Connection

Attribute ID	Access rule	Name	Data type
1	Get	State	USINT
2	Get	InstanceType	USINT
3	Get	TransportClassTrigger	USINT
4	Get	ProducedConnectionId	UINT
5	Get	ConsumedConnectionId	UINT
6	Get	InitialCommCharacteristics	BYTE
7	Get	ProducedConnectionSize	UINT
8	Get	ConsumedConnectionSize	UINT
9	Get/Set	ExpectedPackedRate	UINT
12	Get/Set	WatchdogTimeoutAction	USINT
13	Get	Produced Connection Path Length	USINT
14	Get	Produced Connection Path	EPATH
15	Get	Consumed Connection Path Length	USINT
16	Get	Consumed Connection Path	EPATH

- Instance 2: Polled I/O Connection

Attribute ID	Access rule	Name	Data type
1	Get	State	USINT
2	Get	InstanceType	USINT
3	Get	TransportClassTrigger	USINT
4	Get	ProducedConnectionId	UINT
5	Get	ConsumedConnectionId	UINT
6	Get	InitialCommCharacteristics	BYTE
7	Get	ProducedConnectionSize	UINT
8	Get	ConsumedConnectionSize	UINT
9	Get/Set	ExpectedPackedRate	UINT
12	Get/Set	WatchdogTimeoutAction	USINT
13	Get	Produced Connection Path Length	USINT
14	Get	Produced Connection Path	EPATH
15	Get	Consumed Connection Path Length	USINT
16	Get	Consumed Connection Path	EPATH

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x05	No	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

6. Class 0x93 Base object

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Parameter: Instance 1

Attribute ID	Access rule	Name	Data type	Default
1	Get	Sfversion	UINT	1100
2	Get/Set	LossDNTreat	USINT	1
3	Get/Set	LossSPTreat	USINT	1
4	Get/Set	MODtime (0...255)	USINT	50
11	Get	USErrCord1	USINT	0
12	Get	USErrCord2	USINT	0
13	Get	USErrCord3	USINT	0
14	Get	USErrCord4	USINT	0
15	Get	USErrCord5	USINT	0

7. Class 0x0F AMD_Parameter

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT
2	Get	MaxInstance	UINT
8	Get	ParaClassDescriptor	WORD
9	Get	ConfAssemblyInst	UINT
10	Get	NativeLanguage	USINT

- Parameter: Instance 1 ~ Instance 216

Attribute ID	Access rule	Name	Data type
1	Get/Set	Parameter Value	-
2	Get	Link Path Size	USINT
Attribute ID	Access rule	Name	Data type
3	Get	Link Path	-
4	Get	Descriptor	WORD
5	Get	Data Type	USINT
6	Get	Data Size	USINT

- Common Services

Service code	Implemented for		Service Name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

8. Class 0x95 AMD_DataConf

- Instance0:

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Reset Parameter	USINT	H'0066
2	Get/Set	dlen_in	USINT	2
3	Get/Set	dlen_out	USINT	2
11	Get/Set	Data_in [0]	UINT	H'2000
12	Get/Set	Data_in [1]	UINT	H'2001H
13	Get/Set	Data_in [2]	UINT	H'FFFF
14	Get/Set	Data_in [3]	UINT	H'FFFF
15	Get/Set	Data_in [4]	UINT	H'FFFF
16	Get/Set	Data_in [5]	UINT	H'FFFF
17	Get/Set	Data_in [6]	UINT	H'FFFF
18	Get/Set	Data_in [7]	UINT	H'FFFF

Attribute ID	Access rule	Name	Data type	Default
25	Get/Set	Data_out [0]	UINT	H'2101
26	Get/Set	Data_out [1]	UINT	H'2102
27	Get/Set	Data_out [2]	UINT	H'FFFF
28	Get/Set	Data_out [3]	UINT	H'FFFF
29	Get/Set	Data_out [4]	UINT	H'FFFF
30	Get/Set	Data_out [5]	UINT	H'FFFF
31	Get/Set	Data_out [6]	UINT	H'FFFF
32	Get/Set	Data_out [7]	UINT	H'FFFF

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x05	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

9. Class 0x96 PLC_Access

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance 1 through 9

Instance ID	Description
1	S device of PLC
2	X device of PLC
3	Y device of PLC
4	M device of PLC
5	T device of PLC (bit device)
6	T device of PLC (word device)
7	C device of PLC (bit device)
8	C device of PLC (word device)
9	D device of PLC

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x32	No	Yes	Get_PLC_Data
0x33	No	Yes	Set_PLC_Data

- All instances in Class 0x96 do not support any Attribute.

10. Class 0x97 DVP PLC_Config

- Class attributes

Attribute ID	Access rule	Name	Data type	Default
1	Get	Revision	UINT	-

- Instance 1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/Set	Rest Parameter	USINT	H'0066
2	Get/Set	M_dlen_in	UINT	H'000A
3	Get/Set	D_dlen_in	UINT	H'0006
4	Get/Set	M_dlen_out	UINT	H'000A
5	Get/Set	D_dlen_out	UINT	H'0006
6	Get/Set	M_in_start_adr	UINT	H'0100
7	Get/Set	D_in_start_adr	UINT	H'0198
8	Get/Set	M_out_start_adr	UINT	H'0200
9	Get/Set	D_out_start_adr	UINT	H'01F4
10	Get/Set	Comm Timeout	UINT	H'0020

- Common Services

Service code	Implemented for		Service Name
	Class	Instance	
0X05	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

11. Class 0x98 DT_Parameter

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Parameter: Instance 1 ~ Instance 63

Attribute ID	Access rule	Name	Data type
1	Get/Set	Parameter Value	-
2	Get	Link Path Size	USINT

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

12. Class 0x99 DT_DataConfig

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance 1

Attribute ID	Access rule	Name	Data type	Default
1	Get	Reset parameter	UINT	H'0066
2	Get/Set	dlen_in	USINT	2
3	Get/Set	dlen_out	USINT	2
11	Get/Set	data_in [0]	UINT	H'4701
12	Get/Set	data_in [1]	UINT	H'4719
13	Get/Set	data_in [2]	UINT	H'FFFF
14	Get/Set	data_in [3]	UINT	H'FFFF
15	Get/Set	data_in [4]	UINT	H'FFFF
16	Get/Set	data_in [5]	UINT	H'FFFF
17	Get/Set	data_in [6]	UINT	H'FFFF
18	Get/Set	data_in [7]	UINT	H'FFFF
25	Get/Set	data_out [0]	UINT	H'4700
26	Get/Set	data_out [1]	UINT	H'4719
27	Get/Set	data_out [2]	UINT	H'FFFF
28	Get/Set	data_out [3]	UINT	H'FFFF
29	Get/Set	data_out [4]	UINT	H'FFFF
30	Get/Set	data_out [5]	UINT	H'FFFF
31	Get/Set	data_out [6]	UINT	H'FFFF
32	Get/Set	data_out [7]	UINT	H'FFFF

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0X05	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

13. Class 0x9A ASDA_Parameter

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Parameter: Instance 1 ~ Instance 138

Attribute ID	Access rule	Name	Data type
1	Get/Set	Parameter Value	-

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

14. Class 0x9B ASDA_DataConfig

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance 1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/set	resetparameter	UINT	H'0066
2	Get/Set	dlen_in	USINT	1
3	Get/Set	dlen_out	USINT	1
11	Get/Set	data_in [0]	UINT	H'0407
12	Get/Set	data_in [1]	UINT	H'FFFF
13	Get/Set	data_in [2]	UINT	H'FFFF
14	Get/Set	data_in [3]	UINT	H'FFFF
15	Get/Set	data_in [4]	UINT	H'FFFF
16	Get/Set	data_in [5]	UINT	H'FFFF
17	Get/Set	data_in [6]	UINT	H'FFFF
18	Get/Set	data_in [7]	UINT	H'FFFF
19	Get/Set	data_in [8]	UINT	H'FFFF
20	Get/Set	data_in [9]	UINT	H'FFFF
21	Get/Set	data_in [10]	UINT	H'FFFF
22	Get/Set	data_in [11]	UINT	H'FFFF
23	Get/Set	data_in [12]	UINT	H'FFFF
24	Get/Set	data_in [13]	UINT	H'FFFF
25	Get/Set	data_in [14]	UINT	H'FFFF
26	Get/Set	data_in [15]	UINT	H'FFFF
31	Get/Set	data_out [0]	UINT	H'0409
32	Get/Set	data_out [1]	UINT	H'FFFF
33	Get/Set	data_out [2]	UINT	H'FFFF
34	Get/Set	data_out [3]	UINT	H'FFFF
35	Get/Set	data_out [4]	UINT	H'FFFF
36	Get/Set	data_out [5]	UINT	H'FFFF
37	Get/Set	data_out [6]	UINT	H'FFFF
38	Get/Set	data_out [7]	UINT	H'FFFF
39	Get/Set	data_out [8]	UINT	H'FFFF
40	Get/Set	data_out [9]	UINT	H'FFFF
41	Get/Set	data_out [10]	UINT	H'FFFF
42	Get/Set	data_out [11]	UINT	H'FFFF
43	Get/Set	data_out [12]	UINT	H'FFFF
44	Get/Set	data_out [13]	UINT	H'FFFF
45	Get/Set	data_out [14]	UINT	H'FFFF

Attribute ID	Access rule	Name	Data type	Default
46	Get/Set	data_out [15]	UINT	H'FFFF

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0X05	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

15. Class 0x9C DOP_DataConfig

- Class attributes

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance 1:

Attribute ID	Access rule	Name	Data type	Default
1	Get/set	resetparameter	UINT	66H
2	Get/Set	dlen_in	USINT	4
3	Get/Set	dlen_out	USINT	4

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0X05	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

16. Class 0x9E Custom_Parameter

- Class attributes

Attribute ID	Access rule	Name	Data Type
1	Get	Revision	UINT

- Parameter: Instance 1 ~ Instance N

Attribute ID	Access rule	Name	Data Type
1	Get/Set	Parameter Value	-

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

17. Class 0x9F Custom Data Config

- Instance0:

Attribute ID	Access rule	Name	Data type
1	Get	Revision	UINT

- Instance1:

Attribute ID	Access rule	Name	Data Type	Default
1	Get/Set	resetparameter	USINT	H'0066
2	Get/Set	dlen_in	USINT	2 words
3	Get/Set	dlen_out	USINT	2 words
9	Get/Set	Data_in [0]	UINT	H'2000
10	Get/Set	Data_in [1]	UINT	H'2001
11	Get/Set	Data_in [2]	UINT	H'FFFF
12	Get/Set	Data_in [3]	UINT	H'FFFF
17	Get/Set	Data_out [0]	UINT	H'2101
18	Get/Set	Data_out [1]	UINT	H'2102
19	Get/Set	Data_out [2]	UINT	H'FFFF
20	Get/Set	Data_out [3]	UINT	H'FFFF

- Instance 2 EEPROM read/write

Attribute ID (decimal)	Access rule	Name	Data type
30	Get/Set	AMD reset parameter	H'0066
71	Get/Set	PLC reset parameter	H'0066
90	Get/Set	DTA reset parameter	H'0066
150	Get/Set	ASDA reset parameter	H'0066
	Get/Set	DOP reset parameter	H'0066
200	Get/Set	Custom reset parameter	H'0066

- Instance 3 Modbus baud rate setting

Attribute ID(decimal)	Access rule	Name	Range	Default
1	Get/Set	AMD baud rate setting	1 ~ 2	2
2	Get/Set	PLC baud rate setting	1 ~ 4	4
3	Get/Set	DT baud rate setting	1 ~ 2	2
4	Get/Set	ASDA baud rate setting	1 ~ 4	4
5	Get/Set	DOP baud rate setting	1 ~ 4	4
6	Get/Set	CUSTOM baud rate setting	1 ~ 4	1

- Common Services

Service code	Implemented for		Service name
	Class	Instance	
005	Yes	Yes	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

- By RS-485 communication

