



DeviceNet (DN-02) User Manual

Table of Contents

Chapter 1 Preface	1-1
1.1 Receiving and Inspection	1-1
1.2 Using this Manual	1-1
1.3 Features.....	1-2
Chapter 2 Overview	2-1
2.1 Outline	2-1
2.2 Description.....	2-1
2.3 Network Example.....	2-2
2.4 The Distance of Trunk Line.....	2-2
2.5 The Length of Drop Cable.....	2-2
2.6 Mounting.....	2-3
Chapter 3 Quick Start	3-1
3.1 Setting Steps for Installation	3-1
3.2 EDS File Description.....	3-2
3.3 MAC ID and Baud Rate	3-2
3.4 Input/Output Data	3-3
Chapter 4 Configuration	4-1
Chapter 5 Troubleshooting	5-1
5.1 Network LED.....	5-1
5.2 Module LED	5-2
5.3 SP LED	5-2

Appendix A Installation Information	A-1
A.1 Dimension	A-1
A.2 Ambient Conditions	A-1
A.3 Product Certifications	A-2
Appendix B DeviceNet Object	B-1
Appendix C Auto EDS Generator Software	C-1

1.1 Receiving and Inspection

All Delta DN-02 have gone through rigorous quality control tests at factory prior to ship. After receiving DN-02, please check that the package includes:

- 1pcs DN-02,
- 1pcs 8cm CD,
- 1pcs communication cable (RJ-11, 6 pins),
- 1 instruction sheet.

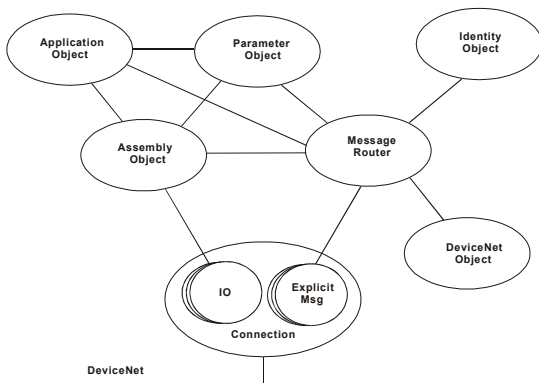
1.2 Using this Manual

- Chapter 2 describes the use DN-02 in field.
- Chapter 3 briefly describes setting for installation.
- Chapter 4 describes configure the DN-02 based upon your requirements.
- Chapter 5 describes correct actions by LED.
- Appendix A provides the information that you may need to install the DN-02.
- Appendix B defines the DeviceNet object classes, class services and attributes.
- Appendix C describes how to use the "AutoEDS software" to generate EDS file by the connected drive.
- Firmware version for each series should conform with the following table:

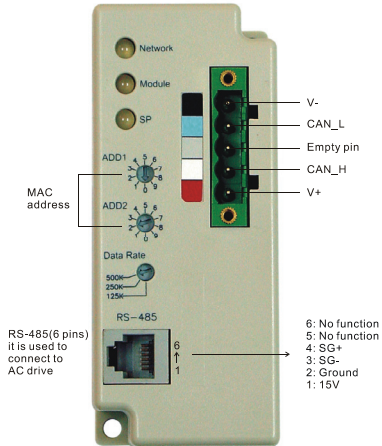
Delta AC Drive	Firmware Version
VFD-B	Version 4.00 and above
VFD-M	Version 3.00 and above
VFD-S	Version 2.50 and above

1.3 Features

- The communications adapter device acts as a gateway from DeviceNet network to MODBUS network.
- DIN rail mount.
- Defined as a DeviceNet Communications Device Profile 12 (0xC).
- There are 8 I/O addresses used to control the length by setting. The I/O mapping address can be set by the DeviceNet manage software system, such as RSNetWorx® from Rockwell. By this way, user can monitor the VFD series by the ladder diagram, such as RSLogix 500® from Rockwell, and send commands to VFD series.
- No external power needed. Power is supplied via RS-485 that is connected to VFD series. Six pins RJ-11 cable, which is packed together with DN-02, is used to connect RS-485 of VFD series to RS-485 of DN-02 for power. DN-02 starts operation once the connection finished.
- DeviceNet Group2 slave functionality including:
 - Explicit connection
 - Polled connection
- Object Model



2.1 Outline



2.2 Description

Bi colors LEDs (LED Network, LED Module and LED SP): provide users to analyze DeviceNet network and get the status of the AC motor drive to solve the problem from the Chapter 5 Troubleshooting.

MAC address: used to set the physical address in the DeviceNet network.

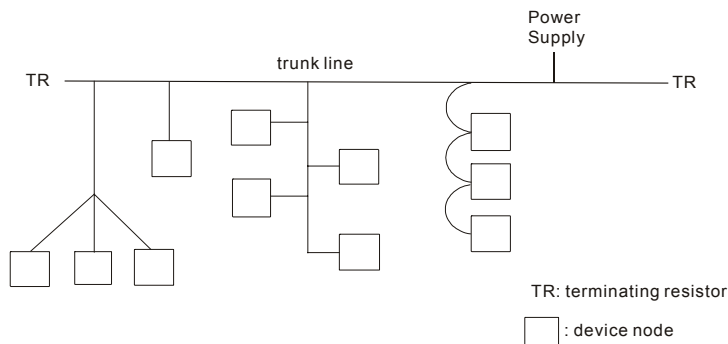
Data rate: used to set the baud rate on the DeviceNet network.

RS-485 serial port: used to connect to the AC motor drive via RJ-11. SG+ and SG- are for the signal. Besides, the power of DN-02 is provided from the AC motor drive (15V) via this port.

DeviceNet connection: 5-pin phoenix connector is used to connect DN-02 to DeviceNet network.

- ◆ **5-pin phoenix connector:**
 1. Red: V+, power supply.
 2. White: CAN_H, signal high.
 3. Bare: SHIELD, shield.
 4. Blue: CAN_L, signal low.
 5. Black: V-, common.

2.3 Network Example



2.4 The Distance of Trunk Line

The distance between any two points must not exceed the maximum cable distance allowed for the data. Refer to following table for details.

Data rate	Maximum distance (flat cable)	Maximum distance (thick cable)	Maximum distance (mid cable)	Maximum distance (thin cable)
125k bit/s	420m (1378 ft)	500m (1640 ft)	300m (984 ft)	100m (328 ft)
250k bit/s	200m (656 ft)	250m (820 ft)	250m (820 ft)	100m (328 ft)
500k bit/s	75m (246 ft)	100m (328 ft)	100m (328 ft)	100m (328 ft)

NOTE: Please refer to the "Appendix B -- DeviceNet Cable Profiles" in the DeviceNet Specifications for the wiring cable to ensure the max. distance.

2.5 The Length of Drop Cable

A drop line connects a node on the DeviceNet cable system to the DeviceNet trunk. Refer to following table for details.

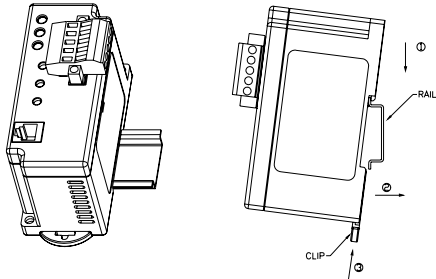
Data Rates	125 Kbps	250 Kbps	500 Kbps
Maximum Drop Length	6 m (20 ft)	6 m (20 ft)	6 m (20 ft)
Cumulative Drop Budget	156 m (512 ft)	78 m (256 ft)	39 m (128 ft)

NOTE: Please refer to the "Appendix B -- DeviceNet Cable Profiles" in the DeviceNet Specifications for the wiring cable to ensure the max. distance.

2.6 Mounting

Please mount the 35mm DIN rail to the DN-02 as shown in the following figure.

- ① Hook the top of the DIN rail to the DN-02
- ② Push the bottom of the DIN rail to the DN-02
- ③ Push the clip to lock the DIN rail



This page intentionally left blank

3.1 Setting Steps for Installation

Step 1. Setting communication format of the AC drives to 19200 RTU 8, N, 2.

Step 2. Setting frequency source of the AC drives to operate from RS-485.

Step 3. Setting operation source of the AC drives to operate from communication interface.

Note: Refer to the following table for setting step 1, step 2 and step 3.

VFD-M	VFD-S	VFD-B	Description
P00=3	P02-00=4	P02-00=4	Master frequency determined by RS-485
P01=4	P02-01=4	P02-01=4	Operating instructions determined by the RS-485 communication
P89=2	P09-01=2	P09-01=2	19200 bits/second
P92=3	P09-04=3	P09-04=3	Modbus RTU mode, 8 bits, no parity and 2 stop bits

Step 4. Using RJ11 (6 pins) to connect RS-485 of the DN-02 and the AC drive.

Step 5. Adding the EDS files (saved in the CD) into DeviceNet management software.

Step 6. Using the operation method of DeviceNet management software for DN-02 connection.

Note: Please make sure that the AC motor drive has set as above table. Or DN-02 cannot work correctly. SP LED should be Green after finishing Step 4 and indicates the DN-02 is communicating with the AC motor drive. If the SP LED is not ON, please refer to chapter 4 Troubleshooting to solve the problem.

3.2 EDS File Description

Function: it is used in the DeviceNet management software tool, such as RSNetWorx® of RA. It is used to distinguish which VFD type it connects and modify the VFD settings.

To use proper EDS file, please check the AC drives type and firmware version from the parameter first. Then you can select the correct EDS file by the following name rule of EDS file.

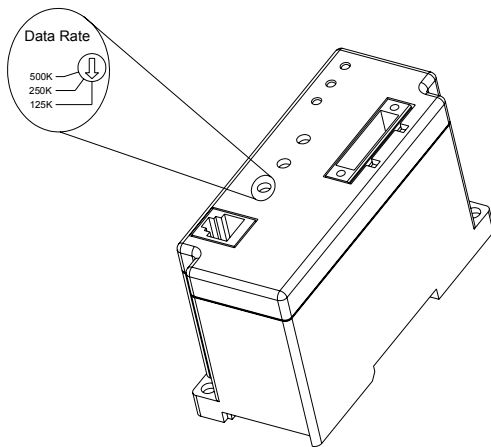
Ex: DN_S_2p52_0.eds

| | | EDS version
 | | VFD software version (V2.52)
 | VFD-S series

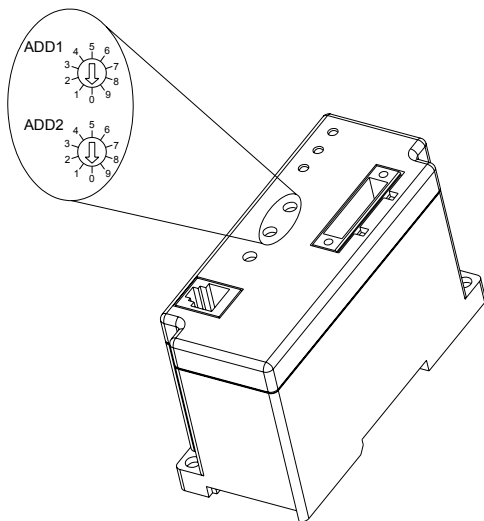
3.3 MAC ID and Baud Rate

■ Setting Baud rate

Switch value	0	1	2	Other
Baud Rate	125K	250K	500K	Not Used



- Set MAC Address: Using decimal data set the MAC address on DeviceNet network.



Address	Description
0-63	All addresses should be set within this range. Before setting address, please make sure that the address you want to use is available (NOT used by others). The address cannot be used repeatedly. Or it cannot connect to the network and the NET LED will be RED.
64-99	This range is illegal addresses and the NET LED will be RED.

Example: ADD1: 3, ADD2: 6, Data Rate: 500k.

Then the value of MAC ID is 36, not 0x36; the Baud rate is 500K.

3.4 Input/Output Data

Default settings of I/O poll message are 4 bytes input and 4 bytes output data.

The information in the following tables is an example:

- Input data from AC drivers to DeviceNet (read only)

Word	Function Description
0	Status of AC Driver, ex: VFD-M is 2101H
1	Frequency Command, ex: VFD-M is 2102H

- Output data from DeviceNet to AC drivers

Word	Function Description
0	Operate Command, Ex: VFD-M is 2000H
1	Frequency command, Ex: VFD-M is 2001H

- Refer to each AC drive user manual for setting above table.



If DN-02 has not been set before using, you can use it by connecting with DeviceNet network without any settings. In this case, DN-02 provides a default I/O setting as table above. DN-02 will use this default setting to exchange data with network in I/O message when power up. Please refer to Chapter 4 Configuration if a desire I/O setting is needed.

Chapter 4 Configuration

DN-02 also provides other functions for user to set by requirement. DN-02 supports an Object named DataConf (0x95), this object can be accessed via DeviceNet configuration or management tools. DataConf Object is defined as follows:

Class 0x95 Data Configure

■ Class Attributes

Attribute ID	Access Rule	Name	Data Type
1	Get	Revision	UINT

■ Instance 1

Attribute ID	Access Rule	Name	Data Type	Description
2	Get/Set	Config_flag	USINT	0: DN-02 will use this object for I/O message 1: DN-02 use default setting.
3	Get/Set	NetCtrl	USINT	0: controlled local. 1: controlled by network.
4	Get/Set	NetRef	USINT	0: set speed local, 1: set speed by network.
5	Get/Set	LossDNTreat	USINT	0: Ignore and continue 1: Stop according to "Stop method".
17	Get/Set	Dlen_in	USINT	Length of input data
18	Get/Set	Dlen_out	USINT	Length of output data
19	Get/Set	Data_in[0]	UINT	1 st word input data
20	Get/Set	Data_in[1]	UINT	2 nd word input data
21	Get/Set	Data_in[2]	UINT	3 rd word input data
22	Get/Set	Data_in[3]	UINT	4 th word input data
23	Get/Set	Data_in[4]	UINT	5 th word input data
24	Get/Set	Data_in[5]	UINT	6 th word input data
25	Get/Set	Data_in[6]	UINT	7 th word input data

Attribute ID	Access Rule	Name	Data Type	Description
26	Get/Set	Data_in[7]	UINT	8 th word input data
49	Get/Set	Data_out[0]	UINT	1 st word output data
50	Get/Set	Data_out[1]	UINT	2 nd word output data
51	Get/Set	Data_out[2]	UINT	3 rd word output data
52	Get/Set	Data_out[3]	UINT	4 th word output data
53	Get/Set	Data_out[4]	UINT	5 th word output data
54	Get/Set	Data_out[5]	UINT	6 th word output data
55	Get/Set	Data_out[6]	UINT	7 th word output data
56	Get/Set	Data_out[7]	UINT	8 th word output data

■ Common services

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

Explanation: User can set following variables for DataConf object by the requirement.

1. `dlen_in/dlen_out`: for user to select input/output length when communicating with DeviceNet network. Maximum length of `dlen_in/dlen_out` is 16 bytes.



After changing these two settings, the settings of Rx size and Tx size of I/O Parameters in scanner module is needed to be same as these two settings. Or there will be fault and cannot connect to DeviceNet network when running DN-02.

2. data_in1~data_in8: used to save 16 bytes data of each Modbus address of AC drive. For example, if address data in data_in1 is 0x2000, it will send the first word of data to AC drive address 0x2000. Similarly, if address data in data_out1 is 0x2101, DN-02 will send the first word of data in AC drive address 0x2101 to DeviceNet network. The length of data_in/data_out is determined by dlen_in/dlen_out. After finishing setting, the command from DN-02 to the AC motor drive won't be valid immediately.
 - *Re-power on*
 - *The modified setting will be valid via service "RESET".*



the column length is UINT, i.e. word (2 bytes). An error will occur when inputting byte for data size.

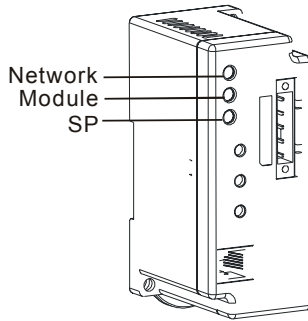
3. NetCtrl/NetRef: used to determine if DN-02 is controlled via DeviceNet network. If it is 0, DeviceNet network can't control DN-02 and user can modify control command/frequency command directly. If it is 1, DN-02 is controlled by DeviceNet network via PLC ladder diagram commands.
4. LossDNTreat: used to determine to stop the AC motor drive or not when losing connection with DeviceNet network.
 - 0: AC drive will still in current running status when losing connection with DeviceNet.
 - 1: AC drive will stop running when losing connection with DeviceNet by the AC drive's setting.
5. config_flag: factory setting is 0. It needs to be set to 1 when reset all settings to the factory settings.
6. The following table is the factory setting of DataConf Object

Attribute ID	Name	Value(Hex)
1	sfversion	
2	config_flag	0000H
3	NetCtrl	0001H
4	NetRef	0001H
5	LossDNTreat	0001H
17	dlen_in	0002H
18	dlen_out	0002H

Attribute ID	Name	Value(Hex)
19	data_in[0]	2000H
20	data_in[1]	2001H
21	data_in[2]	2002H
22	data_in[3]	0000H
23	data_in[4]	2000H
24	data_in[5]	2001H
25	data_in[6]	2002H
26	data_in[7]	0000H
49	data_out[0]	2101H
50	data_out[1]	2103H
51	data_out[2]	2104H
52	data_out[3]	010AH
53	data_out[4]	2101H
54	data_out[5]	2103H
55	data_out[6]	2104H
56	data_out[7]	010AH

Chapter 5 Troubleshooting

This chapter provides LEDs information, and corrective actions for solving problem.



5.1 Network LED

State	Indication	Corrective Actions
LED is off	No power or duplicate ID not completed.	<ol style="list-style-type: none">1. Verify that the power supply of DN-02 is connected and the power is applied to the DN-02.2. Make sure one or more nodes are communicating on the network.3. Make sure at least one other node on the network is in operation at the same time and data rate is the same as the DN-02.
Flashing Green LED	DN-02 is connected to the network but something wrong with the network configuration setting.	
Green LED	Online and connected. One or more connections established	
Flashing Red LED	Online but failed to establish connection.	

State	Indication	Corrective Actions
Red LED	Network failure. Failed duplicate ID or Bus-off.	<ol style="list-style-type: none"> 1. Ensure that all nodes have unique address. 2. If all node addresses are unique, check network for correct media installation. 3. Ensure that the address is NOT in the illegal range.

5.2 Module LED

State	Indication	Corrective Actions
LED is off	No power/not online	Ensure that the connected AC drive is powered and connected to the DN-02.
Flashing Green LED	Waiting for I/O data. No I/O, or PLC is in program mode	DN-02 has passed all operation tests and waits to transmit I/O data between the DN-02 and the AC drives.
Green LED	I/O is in running mode	
Flashing Red LED	Configuration problem. Bad CRC of DN-02 parameters or flash program.	Reset internal I/O data of the DN-02. Please refer to Data Configuration address assignment for detail.
Red LED	Hardware Failure. Failed internal or external RAM test.	Return to the factory.

5.3 SP LED

State	Indication	Corrective Actions
LED is off	No power	No power is applied to the device.
Flashing Green LED	DN-02 is reading the factory settings of the AC drives. DN-02 gets parameters information of the AC drives and initializes some Identity attributes.	
Green LED	DN-02 and the AC drives are communicating.	
Flashing Red LED	CRC check error/AC drives return error information.	To check if the setting of communication format of the AC drives is correct. Please refer to the installation settings for detail.

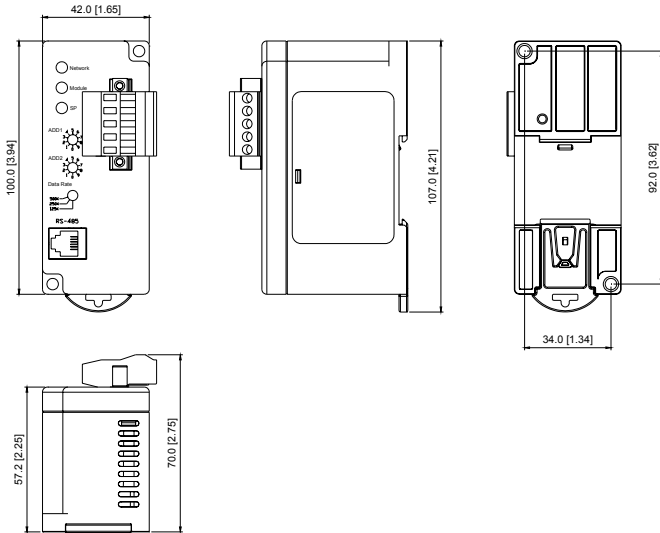
State	Indication	Corrective Actions
Red LED	Connection failure/no connection	<ol style="list-style-type: none"><li data-bbox="495 149 947 205">1. To check if the connection between the AC drive and RS-485 of DN-02 is correct.<li data-bbox="495 215 947 270">2. Re-wire the AC drive connection and ensure that the wire specification is correct.

This page intentionally left blank

Appendix A Installation Information

Appendix A provides the information that you may need to install the DN-02.

A.1 Dimension



A.2 Ambient Conditions

Operation: Temperature: -10°C to 40°C (14°F to 104°F) without dust cover.


Storage: Temperature: -20°C to 60°C (-4°F to 140°F)
Relative Humidity: Less than 90% , no condensation allowed.

Transportation: Temperature: -20°C to 60°C (-4°F to 140°F)
Relative Humidity: Less than 90% , no condensation allowed.

Pollution Degree: 2: good for a factory type environment.

A.3 Product Certifications

This product meets the following certifications.

Certifications	Description
UL	UL508C
CE	EN50178 and EN61800-3
DeviceNet Conformance Test	

Appendix B DeviceNet Object

Appendix B defines the DeviceNet object classes, class services and attributes. The following information about the DeviceNet objects can be accessed by using Explicit Messages. Refer to the DeviceNet specification for using Explicit Messages. You can get the following table information from the DN-02 through any DeviceNet management tool.

Ex. How to get the Vendor ID through Explicit Message?

Request data

	Value	Description
Service	0x0E	Get Attribute Single
Class	0x01	Identity Object
Instance	0x01	Instance 1
Attribute	0x01	Vendor ID

Response data

	Value	Description
Service	0x8E	Successful
data	0x031F	ID = 0x031F(Delta)

Object Classes

DN-02 supports following object classes and user can read or write them via the DeviceNet management tool.

Class	Object
0x01	Identity Object
0x02	Message Router Object
0x03	DeviceNet Object
0x05	Connection Object
0x0F	Parameter Access Object
0x95	DataConf Object

Class 0x01 Identity Object

This object provides identification and general information about the device.

- 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	VendorId	UINT
2	Get	DeviceType	UINT
3	Get	ProductCode	UINT
4	Get	Revision MaxRev 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
0x11	Yes	No	Find_Next_Object_Instance

Class 0x02 Message Router

The Message Router Object provides a messaging connection point to serve any object class or instance in the Client.

- 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

Class 0x03 DeviceNet Object

The DeviceNet Object provides the configuration and status of a DeviceNet port. Each DN-02 supports one DeviceNet object for a physical connection to the DeviceNet communication link.

- 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	MACID	USINT
2	Get	BaudRate	USINT
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

Class 0x05 Connection Object

The Connection Class allocates and manages the internal resources associated with both I/O and Explicit Messaging Connections. The specific instance generated by the Connection Class is called a Connection Instance or a Connection Object.

- 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

Class 0x0F Parameter Object

Use of the Parameter Object provides a known, public interface to a device's configuration data. In addition, this object also provides all the information necessary to define and describe the device's individual configuration parameters.

- 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

We can get the number of parameters of the drive that connected to the DN-02 by Instance 0 and Attribute2.

The number of instances depends on the number of parameters in the drive.

Ex:

Instance	Description
0	Class attributes
1	Drive parameter 1
2	Drive parameter 2
:	:
N	Drive parameter N

- Instance: Parameter Instance

Attribute ID	Access Rule	Name	Data Type
1	Get/Set	Parameter Value	--
2	Get	Link Path Size	USINT
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	
0X05	Yes	No	Reset
0x0E	Yes	Yes	Get_Attribute_Single
0x10	No	Yes	Set_Attribute_Single

Class 0x95 DataConf Object

■ Class attributes

Attribute ID	Access Rule	Name	Data Type
1	Get	Revision	UINT

■ Instance 1

Attribute ID	Access Rule	Name	Data Type	Description
2	Get/Set	Config_flag	USINT	0: DN-02 will use this object for I/O message 1: DN-02 use default setting.
3	Get/Set	NetCtrl	USINT	0: controlled local. 1: controlled by network.
4	Get/Set	NetRef	USINT	0: set speed local, 1: set speed by network.
5	Get/Set	LossDNTreat	USINT	0: Ignore and continue 1: Stop according to "Stop method".
17	Get/Set	dlen_in	USINT	Length of input data
18	Get/Set	Dlen_out	USINT	Length of output data
19	Get/Set	Data_in[0]	UINT	1 st word input data
20	Get/Set	Data_in[1]	UINT	2 nd word input data
21	Get/Set	Data_in[2]	UINT	3 rd word input data
22	Get/Set	Data_in[3]	UINT	4 th word input data
23	Get/Set	Data_in[4]	UINT	5 th word input data
24	Get/Set	Data_in[5]	UINT	6 th word input data
25	Get/Set	Data_in[6]	UINT	7 th word input data
26	Get/Set	Data_in[7]	UINT	8 th word input data

Attribute ID	Access Rule	Name	Data Type	Description
49	Get/Set	Data_out[0]	UINT	1 st word output data
50	Get/Set	Data_out[1]	UINT	2 nd word output data
51	Get/Set	Data_out[2]	UINT	3 rd word output data
52	Get/Set	Data_out[3]	UINT	4 th word output data
53	Get/Set	Data_out[4]	UINT	5 th word output data
54	Get/Set	Data_out[5]	UINT	6 th word output data
55	Get/Set	Data_out[6]	UINT	7 th word output data
56	Get/Set	Data_out[7]	UINT	8 th word output data

■ 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

Appendix C Auto EDS Generator Software

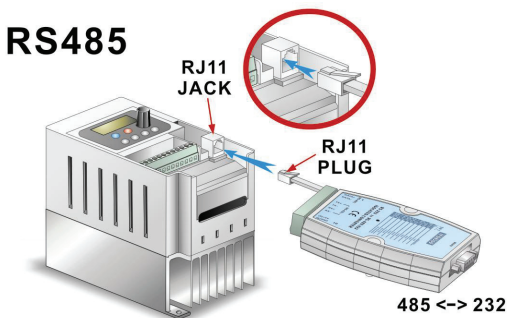
For correct display the drive that connected to the DN-02 in the DeviceNet management tool, we can generate the EDS file for the drive by using the software "Auto EDS Generator". The operation steps are shown in the following.

Step1.

Please set drive's parameters as following table.

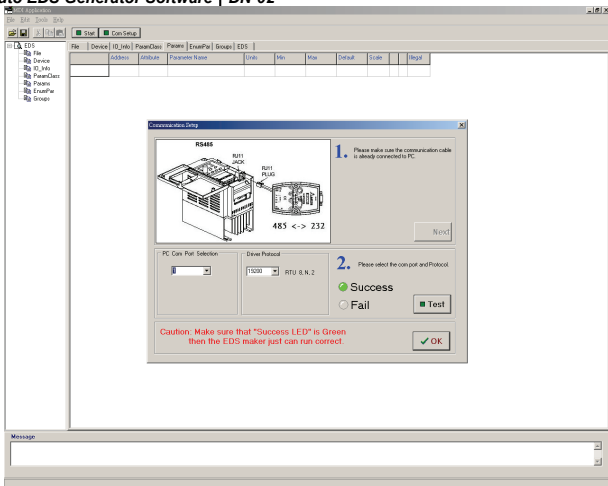
VFD-M	VFD-S	VFD-B	Description
P89=2	P09-01=2	P09-01=2	19200 bits/second
P92=3	P09-04=3	P09-04=3	RTU mode, 8 bits, no parity and 2 stop bits

Hardware Connection



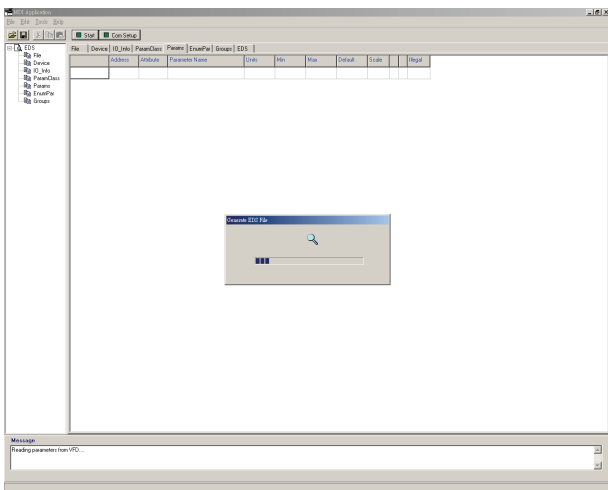
Step2.

Make sure that the hardware connection is correct and then execute the AutoEDS program. Following screen will be displayed. After successful communication test, the AutoEDS will be executed correctly.



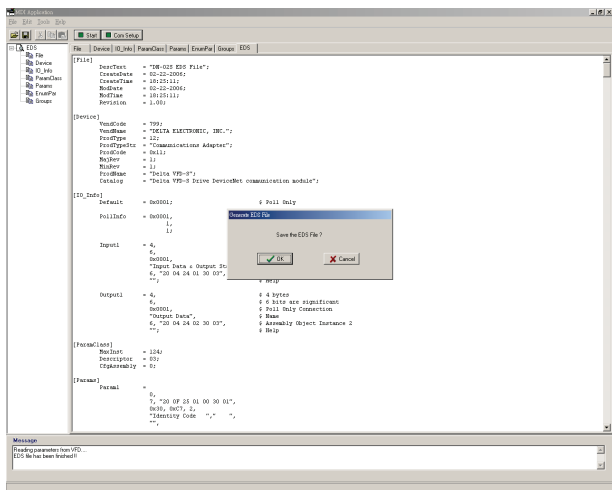
Step3.

After pressing the **【Start】** button at the upper-left corner, program will start generating the EDS file as shown in the following.



Step4.

After generating EDS file, you will get a pop-up window for saving this EDS file.



This page intentionally left blank.