DEVICENET SLAVE MODULE

Introduction

This manual describes DeviceNet slave module FC3A-SX5DS1 used with the OpenNet Controller to interface with the DeviceNet network.

Read this chapter to understand the DeviceNet system setup and the slave module specifications. For general information about safety precautions, installation, wiring, and dimensions, see the OpenNet Controller user's manual EM333.

OpenNet Interface Module

The OpenNet Controller can be linked to three major open networks; INTERBUS, DeviceNet, and LONWORKS. For communication through these networks, OpenNet interface modules are available. Mounting the DeviceNet slave module beside the OpenNet Controller CPU module makes a slave station used as an I/O terminal in a DeviceNet network. The slave station can transfer I/O data to and from the master station just as an ordinary I/O module in a distributed network.

DeviceNet Slave Module Features

Since the FC3A-SX5DS1 module conforms to the DeviceNet specifications, the OpenNet Controller can be linked to DeviceNet networks consisting of DeviceNet compliant products from many different vendors, such as I/O terminals, sensors, drives, operator interfaces, and barcode readers.

The transmit/receive data quantity can be selected from 0 through 8 bytes (64 bits) in 1-byte increments. One DeviceNet slave module enables the OpenNet Controller CPU module to transmit 64 bits and receive 64 bits at the maximum to and from the DeviceNet master station.

About DeviceNet

DeviceNet has been originally developed by Allen-Bradley of the USA as a network for sensors, actuators, and other discrete devices, and later the specifications have been opened. Now, major automotive manufactures and various industries employ the DeviceNet networks.

DeviceNet Features

The network configuration is based on the bus system. The basic network consists of a trunkline-dropline topology, and multi-drop or daisy-chain configuration is also possible.

The DeviceNet protocol is based on CAN (Controller Area Network) which has been widely used for networks on automobiles, making it possible to configure reliable networks with high noise immunity.

Transmission Distance and Nodes

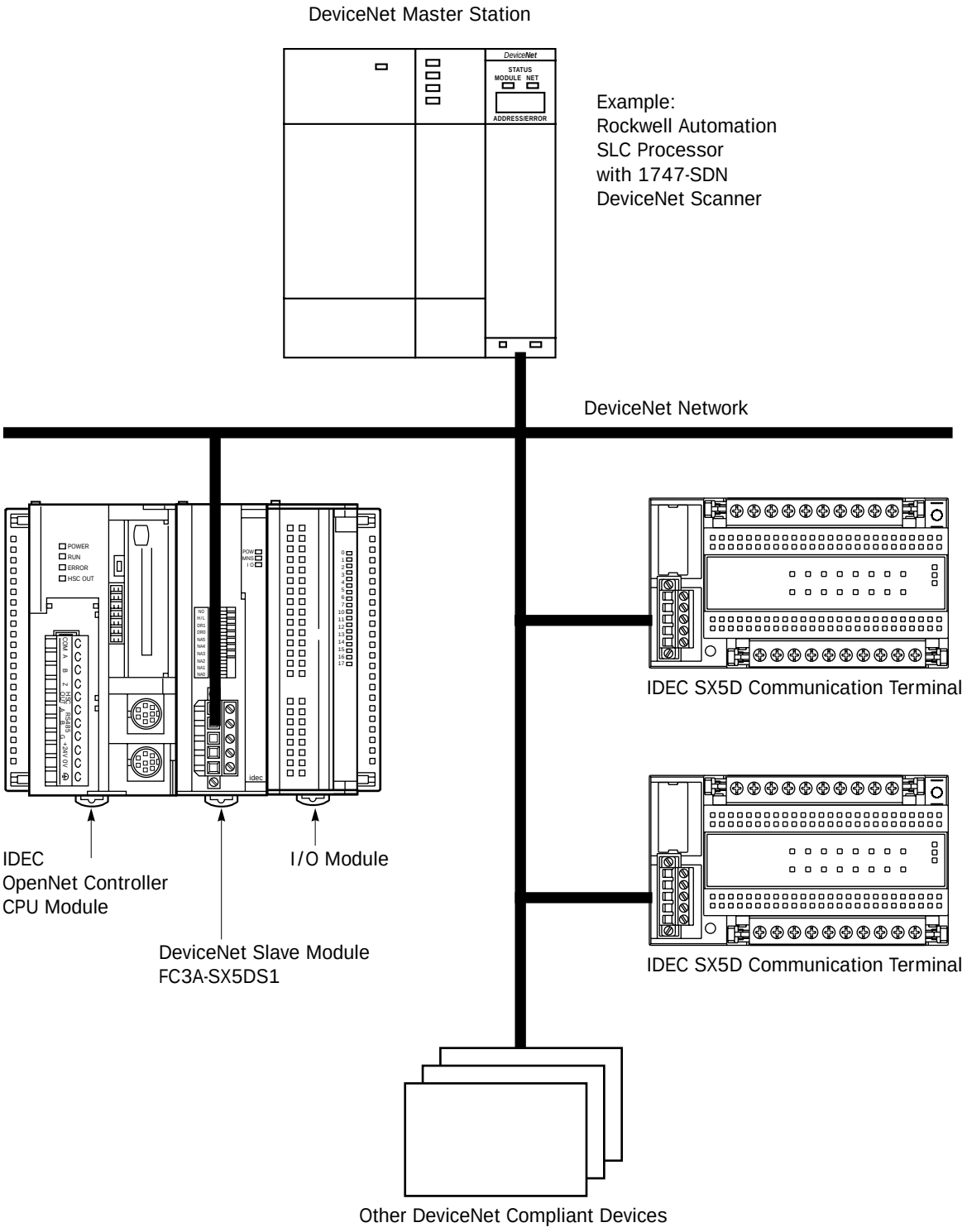
The maximum transmission distance is 500 meters when using a thick trunk cable at a data rate of 125k baud, and the maximum quantity of nodes is 64 including a master station.

System Setup

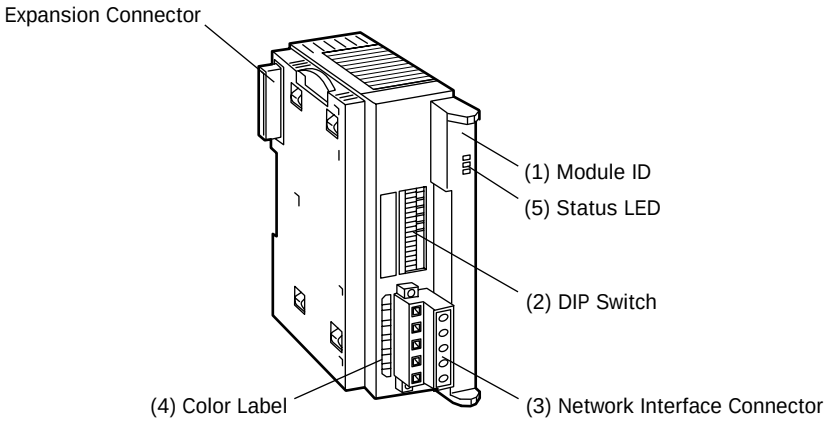
Various DeviceNet compliant devices can be connected to the DeviceNet network.

The DeviceNet network requires a DeviceNet master module available from other vendors. The OpenNet Controller can make up a slave station by installing the FC3A-SX5DS1 module to the right of the OpenNet Controller CPU module.

A maximum of seven OpenNet interface modules and analog I/O modules can be mounted with one OpenNet Controller CPU module.



Parts Description



OpenNet Interface Module for DeviceNet

Module Name	DeviceNet Slave Module
Type No.	FC3A-SX5DS1

(1) Module ID FC3A-SX5DS1 indicates the DeviceNet slave module ID.

(2) DIP Switch 10-pole DIP switch for setting node address, data rate, and output hold/load off

(3) Network Interface Connector For connecting the DeviceNet communication cable

(4) Color Label A five-color label is located beside the connector on the FC3A-SX5DS1 module. Connect each of the five different-color wires of the DeviceNet communication cable to the terminal of a matching color.

Label and Wire Insulation Color	Name
Black	V–
Blue	CAN_L
Bare Wire	Drain
White	CAN_H
Red	V+

(5) Status LED Indicates operating status

Indicator	Status		Description
POW (power)	—	OFF	Module power OFF
	Green	ON	Module power ON
MNS (module/network status)	—	OFF	Power OFF or Dup_MAC_ID test not completed
	Green	Flash	Normal operation (communication not established)
		ON	Normal operation (communication established)
	Red	Flash	Minor fault (temporary network error)
		ON	Critical fault
IO (I/O status)	—	OFF	I/O inactive (communication not established)
	Green	Flash	I/O idle (communication established)
		ON	I/O active
	Red	ON	I/O error or communication interrupted

Specifications

General Specifications

Communication Interface Power Voltage Range	11 to 25V DC
Current Draw	Approx. 25 mA
Isolation	Between control circuit and communication terminal: Photocoupler isolated
Insulation Resistance	Between communication terminal and FG: 10 MΩ minimum (500V DC megger)
Dielectric Strength	Between communication terminal and FG: 1000V AC, 1 minute (10 mA maximum)
Vibration Resistance	10 to 57 Hz, amplitude 0.075 mm; 57 to 150 Hz, acceleration 9.8 m/sec^2 (1G); 10 sweep cycles each in 3 axes (total 80 minutes) (IEC1131)
Shock Resistance	147 m/sec^2 (15G), 11 msec, 3 shocks each in 3 axes (IEC1131)
Altitude	Operation: 0 to 2000m Transportation: 0 to 3000m
Operating Temperature	0 to +55°C (no freezing)
Operating Humidity	30 to 90% RH (no condensation)
Storage Temperature	–25 to +75°C
Storage Humidity	30 to 90% RH (no condensation)
Corrosion Immunity	Free from corrosive gases
Mounting	Snap-on mount on 35-mm DIN rail
Weight (approx.)	180g

Communication Specifications

- **Data Rate and Transmission Distance**

Data Rate	Max. Cable Distance for 100% Thick Cable	Max. Cable Distance for 100% Thin Cable	Max. Drop Line Length	Max. Total Drop Line Length
500k baud	100m	100m	6m	39m
250k baud	250m	100m	6m	78m
125k baud	500m	100m	6m	156m

- **Maximum Number of Stations in the Network** 64 stations (including a master)

- **Communication Data Length**
  Transmit: 0 to 8 bytes (selectable in 1-byte increments)
  Receive: 0 to 8 bytes (selectable in 1-byte increments)

- **Network Interface Connector**
  In the module: MSTB2.5/5-GF-5.08AU (made by Phoenix Contact)
  To the cable: FRONT-MSTB2.5/5-STF-5.08AU (made by Phoenix Contact)

- **Communication Cable (Special DeviceNet Cable)**

Thick Cable Type No.	Thin Cable Type No.	Maker
1485C-P1A50	1485C-P1-C150	Rockwell Automation For details about cables, consult Rockwell Automation.

- **Terminator**
  Terminators must be connected to both ends of the DeviceNet network. When setting up a network, either connect commercially-available terminators at both ends of the network or connect the following resistor to the branch taps at both ends of the network.

  Metal film resistor: 121Ω, ±1%, 1/4W

Wiring DeviceNet Cable

Precautions for Wiring

- Do not run the network cable in parallel with or near power lines, and keep the network cable away from noise sources.
- Power down the DeviceNet slave module before starting wiring. Make sure of correct wiring before powering up the DeviceNet slave module.
- Use the special DeviceNet cable for connecting the network.
- A five-color label is located beside the connector on the DeviceNet slave module. Connect each of the five different-color wires of the cable to the terminal of a matching color.
- When using thick cables, only one wire can be connected a terminal of the network interface connector. To connect two wires of thick cables, use a device tap.
- Tighten the mounting screws of the network interface connector to a recommended torque of 0.3 to 0.5 N-m.
- Tighten the terminal screws of the network interface connector to a recommended torque of 0.5 to 0.6 N-m.
- Either connect commercially-available terminators at both ends of the network or connect the following resistor to the branch taps at both ends of the network. Connect the terminator between the CAN_H (white) and CAN_L (blue) lines.

  Metal film resistor: 121Ω, ±1%, 1/4W

Ferrules, Crimping Tool, and Screwdriver for Phoenix Terminal Blocks

The screw terminal block of the network interface connector can be wired with or without using ferrules on the end of the cable. Applicable ferrules for the terminal block and crimping tool for the ferrules are listed below. The screwdriver is used to tighten the screw terminals on the DeviceNet slave module. These ferrules, crimping tool, and screwdriver are made by Phoenix Contact and are available from Phoenix Contact.

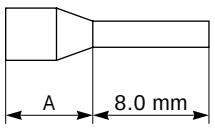
Type numbers of the ferrules, crimping tool, and screwdriver listed below are the type numbers of Phoenix Contact. When ordering these products from Phoenix Contact, specify the Order No. and quantity listed below.

DeviceNet slave modules are connected to the network using special DeviceNet thick or thin cables, each cable consisting of three different sizes of wires listed below.

• Ferrule Order No.

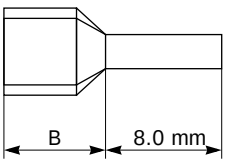
Applicable Wire Size		For 1-wire connection		For 2-wire connection		Pcs./Pkt.
mm^2	AWG	Phoenix Type	Order No.	Phoenix Type	Order No.	
0.25	24	AI 0,25-8 YE	32 00 85 2	—	—	100
0.5	20	AI 0,5-8 WH	32 00 01 4	AI-TWIN 2 x 0,5-8 WH	32 00 93 3	100
0.75	18	AI 0,75-8 GY	32 00 51 9	AI-TWIN 2 x 0,75-8 GY	32 00 80 7	100
1.0	18	AI 1-8 RD	32 00 03 0	AI-TWIN 2 x 1-8 RD	32 00 81 0	100
1.5	16	AI 1,5-8 BK	32 00 04 3	AI-TWIN 2 x 1,5-8 BK	32 00 82 3	100
2.5	14	AI 2,5-8 BU	32 00 52 2	—	—	100

For 1-wire Connection



Ferrule	Dimension A
AI 0,25-8 YE	4.5 mm
AI 0,5-8 WH AI 0,75-8 GY AI 1-8 RD AI 1,5-8 BK AI 2,5-8 BU	6.0 mm

For 2-wire connection



Ferrule	Dimension B
AI-TWIN 2 x 0,5-8 WH AI-TWIN 2 x 0,75-8 GY AI-TWIN 2 x 1-8 RD	7.0 mm
AI-TWIN 2 x 1,5-8 BK	8.0 mm

• Crimping Tool and Screwdriver Order No.

Tool Name	Phoenix Type	Order No.	Pcs./Pkt.
Crimping Tool	CRIMPFOX UD 6	12 04 43 6	1
Screwdriver	SZS 0,6 x 2,5	12 05 04 0	10

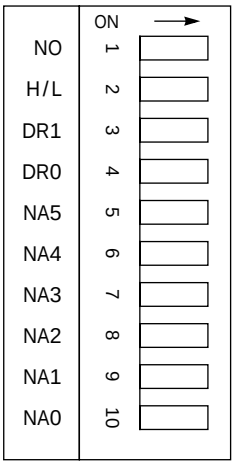
DIP Switch Settings

DIP switches are inside the protective lid. After setting the DIP switches, replace the lid into position.

All DIP switches are set to off before shipping from factory.

Set the DIP switches to select the node address (MAC ID: media access control identifier), data rate, and output hold/load off.

Do not set the DIP switches to the “Selection Prohibited” positions.



Node Address (MAC ID)

Node Address	NA0	NA1	NA2	NA3	NA4	NA5	Node Address	NA0	NA1	NA2	NA3	NA4	NA5
0	OFF	OFF	OFF	OFF	OFF	OFF	32	OFF	OFF	OFF	OFF	OFF	ON
1	ON	OFF	OFF	OFF	OFF	OFF	33	ON	OFF	OFF	OFF	OFF	ON
2	OFF	ON	OFF	OFF	OFF	OFF	34	OFF	ON	OFF	OFF	OFF	ON
3	ON	ON	OFF	OFF	OFF	OFF	35	ON	ON	OFF	OFF	OFF	ON
4	OFF	OFF	ON	OFF	OFF	OFF	36	OFF	OFF	ON	OFF	OFF	ON
5	ON	OFF	ON	OFF	OFF	OFF	37	ON	OFF	ON	OFF	OFF	ON
6	OFF	ON	ON	OFF	OFF	OFF	38	OFF	ON	ON	OFF	OFF	ON
7	ON	ON	ON	OFF	OFF	OFF	39	ON	ON	ON	OFF	OFF	ON
8	OFF	OFF	OFF	ON	OFF	OFF	40	OFF	OFF	OFF	ON	OFF	ON
9	ON	OFF	OFF	ON	OFF	OFF	41	ON	OFF	OFF	ON	OFF	ON
10	OFF	ON	OFF	ON	OFF	OFF	42	OFF	ON	OFF	ON	OFF	ON
11	ON	ON	OFF	ON	OFF	OFF	43	ON	ON	OFF	ON	OFF	ON
12	OFF	OFF	ON	ON	OFF	OFF	44	OFF	OFF	ON	ON	OFF	ON
13	ON	OFF	ON	ON	OFF	OFF	45	ON	OFF	ON	ON	OFF	ON
14	OFF	ON	ON	ON	OFF	OFF	46	OFF	ON	ON	ON	OFF	ON
15	ON	ON	ON	ON	OFF	OFF	47	ON	ON	ON	ON	OFF	ON
16	OFF	OFF	OFF	OFF	ON	OFF	48	OFF	OFF	OFF	OFF	ON	ON
17	ON	OFF	OFF	OFF	ON	OFF	49	ON	OFF	OFF	OFF	ON	ON
18	OFF	ON	OFF	OFF	ON	OFF	50	OFF	ON	OFF	OFF	ON	ON
19	ON	ON	OFF	OFF	ON	OFF	51	ON	ON	OFF	OFF	ON	ON
20	OFF	OFF	ON	OFF	ON	OFF	52	OFF	OFF	ON	OFF	ON	ON
21	ON	OFF	ON	OFF	ON	OFF	53	ON	OFF	ON	OFF	ON	ON
22	OFF	ON	ON	OFF	ON	OFF	54	OFF	ON	ON	OFF	ON	ON
23	ON	ON	ON	OFF	ON	OFF	55	ON	ON	ON	OFF	ON	ON
24	OFF	OFF	OFF	ON	ON	OFF	56	OFF	OFF	OFF	ON	ON	ON
25	ON	OFF	OFF	ON	ON	OFF	57	ON	OFF	OFF	ON	ON	ON
26	OFF	ON	OFF	ON	ON	OFF	58	OFF	ON	OFF	ON	ON	ON
27	ON	ON	OFF	ON	ON	OFF	59	ON	ON	OFF	ON	ON	ON
28	OFF	OFF	ON	ON	ON	OFF	60	OFF	OFF	ON	ON	ON	ON
29	ON	OFF	ON	ON	ON	OFF	61	ON	OFF	ON	ON	ON	ON
30	OFF	ON	ON	ON	ON	OFF	62	OFF	ON	ON	ON	ON	ON
31	ON	ON	ON	ON	ON	OFF	63	ON	ON	ON	ON	ON	ON

Data Rate

Data Rate	DR0	DR1
125k baud	OFF	OFF
250k baud	ON	OFF
500k baud	OFF	ON
(Selection Prohibited)	ON	ON

Output Hold or Load Off

Output/Load	H/L
LOAD OFF	OFF
HOLD	ON

System Reserve

System Reserve	NO
Fixed to	OFF
(Selection Prohibited)	ON

Link Register Allocation in the CPU Module

DeviceNet communication data is stored to link registers in the OpenNet Controller CPU module and the data is communicated through the DeviceNet slave module.

Since seven functional modules including the DeviceNet slave module can be mounted with one OpenNet Controller CPU module, link registers are allocated depending on the position where the DeviceNet slave module is mounted.

Link Register Allocation Numbers

Allocation Number	Area	Function	Description	R/W
L*00	Data area	Receive data	Stores received data from the network	Read
L*01	Data area	Receive data	Stores received data from the network	Read
L*02	Data area	Receive data	Stores received data from the network	Read
L*03	Data area	Receive data	Stores received data from the network	Read
L*04	Data area	Transmit data	Stores transmit data for the network	Write
L*05	Data area	Transmit data	Stores transmit data for the network	Write
L*06	Data area	Transmit data	Stores transmit data for the network	Write
L*07	Data area	Transmit data	Stores transmit data for the network	Write
L*12	Status area	Error data	Stores various error codes	Read
L*13	Status area	I/O counts	Stores the byte counts of transmit/receive data	Read
L*14	Status area	Connection status	Stores the allocation choice byte	Read
L*24	Reserved area	Software version	Stores the system software version	Read

Note: A number 1 through 7 comes in place of * depending on the position where the functional module such as OpenNet interface module or analog I/O module is mounted. Consequently, operand numbers are automatically allocated to each functional module in the order of increasing distance from the CPU module, starting with L100, L200, L300, through L700.

Error Data (Status Area) L*12

L*12	b15	b14: unused	b13	b12-b9: unused	b8	b7-b0: unused

When an error occurs, the MNS or IO LED on the DeviceNet slave module goes on or flashes depending on the error and a corresponding bit in the link register goes on. The status LED goes off when the cause of the error is removed. The error data bit remains on until the CPU is powered up again or reset.

b15 (initialization error)
This bit goes on when the CPU module fails to acknowledge the completion of initialization for communication with the DeviceNet slave module.

b13 (I/O error)
This bit goes on when an error occurs during communication through the CPU bus.

b8 (communication fault)
This bit goes on when a communication fault is detected.

I/O Counts (Status Area) L*13

L*13	b15-b12: transmit bytes	b11-b8: receive bytes	b7-b0: unused

This link register stores the transmit and receive byte counts selected in the Function Area Setting > Open Bus in WindLDR.

Connection Status (Status Area) L*14

L*14	b15-b8: allocation choice	b7-b0: unused

This link register stores the data of the allocation choice byte.

Soft Version (Reserved Area) L*24

L*24	b15-b12: major revision	b11-b8: minor revision	b7-b0: unused

This link register stores the system software version number. [Example] Version 1.3 — 1: major revision, 3: minor revision

Function Area Setting for DeviceNet Slave Station

The quantity of transmit/receive data for DeviceNet network communication is specified using the Function Area Setting in WindLDR. The OpenNet Controller CPU module at the DeviceNet slave station recognizes functional modules automatically at power-up and exchanges data with the DeviceNet master station through the link registers allocated to each slave station (node).

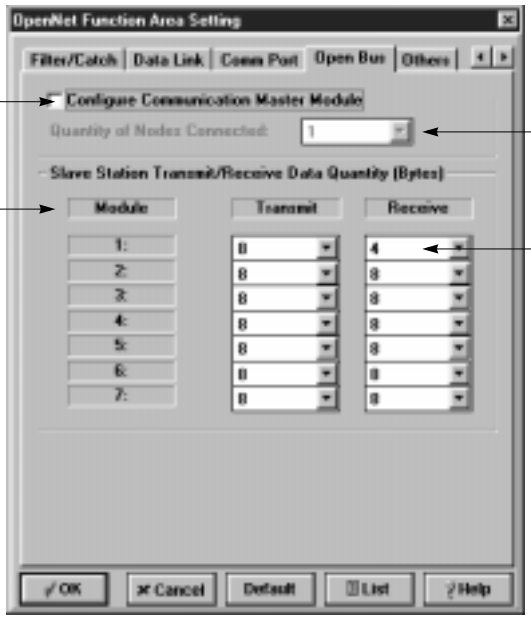
Since these settings relate to the user program, the user program must be downloaded to the OpenNet Controller after changing any of these settings.

Programming WindLDR

1. From the WindLDR menu bar, select **Configure** > **Function Area Settings**. The Function Area Setting dialog appears.
2. Select the **Open Bus** tab.

Configure Communication Master Module Check Box
Check this box only when the remote I/O master module is used.

Slave Station Transmit / Receive Data Quantity (Bytes)
When using OpenNet interface modules for DeviceNet, INTERBUS, or LONWORKS, specify the data bytes to communicate through each OpenNet interface module.



Quantity of Nodes Connected
When using the remote I/O master module, specify the quantity of nodes from 1 through 32.

Transmit / Receive Bytes 0 to 8 (default: 8 bytes)
This value determines the data quantity 0 through 8 bytes to communicate with the DeviceNet or other master module.
For the example on the next page, select 8 transmit bytes and 4 receive bytes for Module 1.

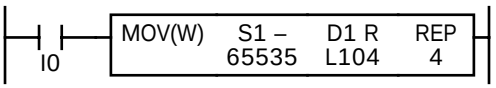
3. Select transmit and receive data bytes for the slave module position 1 through 7 where the slave module is mounted.
4. Click the **OK** button and download the user program to the OpenNet Controller.

Programming a User Program on WindLDR

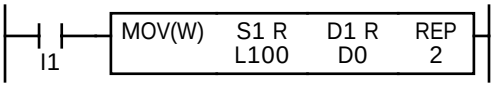
The OpenNet interface module exchanges data between the open network and the link registers in the CPU module allocated to the OpenNet interface module, depending on the slot where the OpenNet interface module is mounted.

To create a communication program for an OpenNet interface module, first determine the slot number where the OpenNet interface module is mounted, and make a program to write data to link registers allocated to transmit data and to read data from link registers allocated to receive data.

Example: When an OpenNet interface module is mounted in the first slot of all functional modules



65535 → L104 through L107
When input I0 is on, constant 65535 (FFFFh) designated by source operand S1 is moved to four link registers L104 through L107 designated by destination operand D1. All 64 bits (8 bytes) in link registers L104 through L107 are turned on. Since link registers L104 through L107 are for transmit data, the data is transmitted to the network.



L100·L101 → D0·D1
When input I1 is on, 32-bit (4-byte) data in two link registers L100 and L101 designated by source operand S1 is moved to data registers D0 and D1 designated by destination operand D1. Since link registers L100 and L101 are for receive data, communication data is read from the network to L100 and L101, then moved to data registers D0 and D1.

Starting Operation

Set up the OpenNet Controller CPU and DeviceNet slave modules, and download the user program to the CPU module. Connect the DeviceNet slave module to the DeviceNet network using DeviceNet cables. Power up the CPU module and start the CPU module to run, then DeviceNet communication starts. The delay until the communication starts after power-up depends on the size of the user program and the system setup.

While the CPU is stopped, data exchange between the CPU and DeviceNet slave modules is halted, but communication with the DeviceNet network continues.

Data exchange between the CPU and DeviceNet slave modules is asynchronous with the user program scanning in the CPU module.

Transmission Time

The response time of the DeviceNet network varies greatly depending on such factors as the quantity of nodes, data bytes, and DeviceNet system setup. To determine the accurate response time, confirm the response time on the actual network system.

The following example describes a response time in a DeviceNet system comprising IDEC SX5D series communication terminals.

Example: DeviceNet Transmission Time

- **System Setup**

  PLC: 1747-L532 (SLC5/03 CPU made by Rockwell Automation)

  Master: 1747-SDN (SLC500 DeviceNet Scanner Module made by Rockwell Automation)

  Slave: SX5D-SBM16K (8pt transistor source input / 8pt transistor sink output)
  SX5D-SBM16P (8pt transistor sink input / 8pt transistor protect source output)
  SX5D-SBR08 (8pt relay output)

  Data Rate: 125k baud

  Operation Mode: Communication according to the scan list in the master

- **System Operation (Data Flow)**

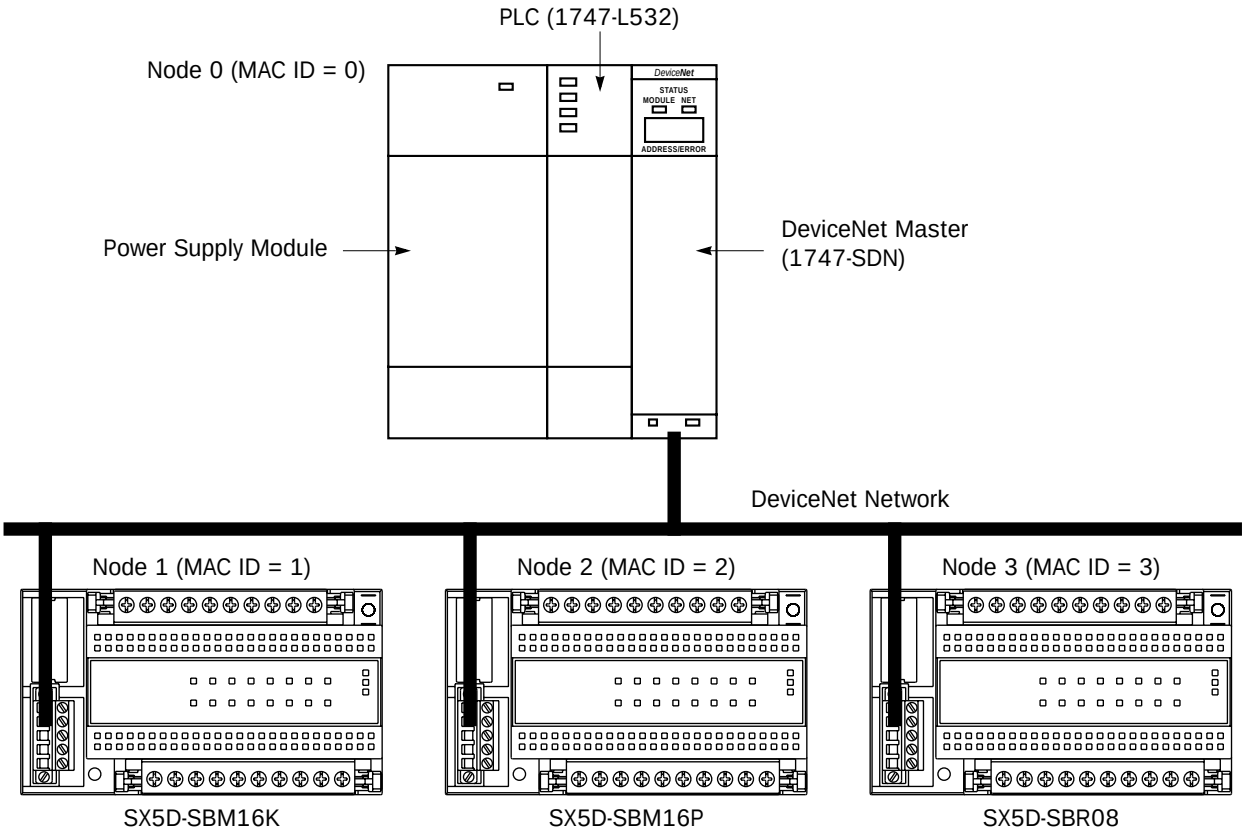
  (1) SX5D-SBM16K sends 8-input data to the master, and the master sends 8-output data to SX5D-SBM16K.
  (2) SX5D-SBM16P sends 8-input data to the master, and the master sends 8-output data to SX5D-SBM16P.
  (3) SX5D-SBM16K sends 8-input data to the master, and the master sends 8-output data to SX5D-SBR08.

- **Calculating the Response Time**

  Response time = Input processing time (slave) + Communication time (slave to master) + Data processing time (master and PLC) + Communication time (master to slave) + Output processing time (slave)

- **Measured Value of Response Time**

  SX5D-SBM16K Input ON/OFF → SX5D-SBM16K Output ON/OFF response time = Approx. 18 msec



Troubleshooting in DeviceNet Operation

Three LED indicators are provided on the DeviceNet slave module. When a trouble occurs during DeviceNet communication, these status LEDs go on or flash depending on the error. When the LEDs go on or flash, locate the error referring to the table described below.

Probable Causes for Network Errors

When a trouble occurs during DeviceNet communication, the following causes are suspected.

- Strong external noise
- The power voltage to the DeviceNet slave module has dropped below the minimum operating voltage (at least momentarily).
- Use of a faulty communication line, incorrect cable, or transmission over the rated distance
- Improper terminator

DeviceNet Master Module fails to recognize the slave module

Status LEDs on Slave Module			Cause	Action
POW	MNS	IO		
OFF	OFF	OFF	Power is not supplied to the OpenNet Controller CPU module	Supply 24V DC to the OpenNet Controller CPU module
				Plug in the expansion connector correctly
Green ON	OFF	OFF	Power is not supplied to the DeviceNet interface	Plug in the communication connector correctly
				Connect the DeviceNet power lines red (V+) and black (V–) correctly
				Supply 11-25V DC to the DeviceNet power line
Green ON	OFF	Green ON	Master is not found	Plug in the communication connector correctly
				Set the data rate correctly using DIP switches
				Set the data rate of the master station correctly
				Make sure that network wiring is correct in the entire DeviceNet network, without short circuit or disconnection
				Connect terminators (121Ω) at both ends of the network
Green ON	Red ON	Green ON	Physical communication trouble or duplicate MAC ID exists in the network	Plug in the communication connector correctly
				Set the data rate correctly using DIP switches
				Set the MAC ID correctly using DIP switches
				Make sure that nodes with duplicate MAC ID does not exist in the same network
				Make sure that network wiring is correct in the entire DeviceNet network, without short circuit or disconnection
				Connect terminators (121Ω) at both ends of the network
Green ON	Green Flash	Green ON	Slave operates normally, but is not recognized by the master	Supply power to the DeviceNet master
				Make sure of correct settings for the master
				Plug in the communication connector correctly
				Set the data rate correctly using DIP switches
				Set the MAC ID correctly using DIP switches
				Make sure that network wiring is correct in the entire DeviceNet network
				Connect terminators (121Ω) at both ends of the network

Communication error occurs

Status LEDs on Slave Module			Cause	Action
POW	MNS	IO		
OFF	OFF	OFF	Power is not supplied to the OpenNet Controller CPU module	Supply 24V DC to the OpenNet Controller CPU module
				Plug in the expansion connector correctly
Green ON	Red ON	Green ON	Physical communication trouble exists in the network	Plug in the communication connector correctly
				Make sure that network wiring is correct in the entire DeviceNet network, without short circuit or disconnection
				Make sure that the network is not affected by noise
Green ON	Red ON	Green ON or Red Flash	Data from the master does not arrive	Make sure that the master is operating
				Plug in the communication connector correctly
				Make sure that network wiring is correct in the entire DeviceNet network, without short circuit or disconnection
				Make sure that the network is not affected by noise
Green ON	Green Flash	Green ON	Communication with the master is not established	Make sure of correct settings for the master
				Make sure that the slave is not stopped by power-down or other causes (if automatic recovery is enabled at the master, communication resumes when power is restored at the slave)
				Plug in the communication connector correctly
				Make sure that network wiring is correct in the entire DeviceNet network, without short circuit or disconnection
				Supply 11-25V DC to the DeviceNet power line

OpenNet Controller link registers cannot receive data from the network correctly

Status LEDs on Slave Module			Cause	Action
POW	MNS	IO		
ON or OFF	ON or OFF	ON or OFF	Incorrect setting or communication error	Make sure of correct settings for the master
				Set the transmit/receive bytes in Function Area Settings correctly
				Make sure of correct link register numbers
				See “DeviceNet Master Module fails to recognize the slave module” and “Communication error occurs” described above

OpenNet Controller link registers cannot send out data to the network correctly

Status LEDs on Slave Module			Cause	Action
POW	MNS	IO		
ON or OFF	ON or OFF	ON or OFF	Incorrect setting or communication error	Make sure of correct settings for the master
				See “DeviceNet Master Module fails to recognize the slave module” and “Communication error occurs” described above