

Changes for the Better

 **mitsubishi**
PROGRAMMABLE CONTROLLERS
MELSEC-F

USER'S MANUAL

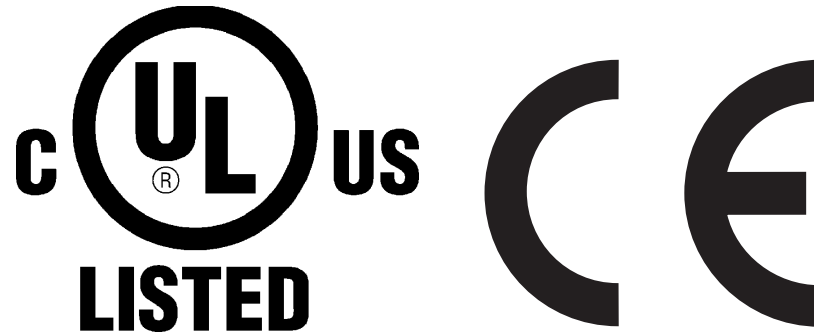
FX_{2N}-64DNET DeviceNet Interface Block

FX_{2N}



Foreword

- This manual contains text, diagrams and explanations which will guide the reader in the correct installation and operation of the FX2N-64DNET DeviceNet Interface Block. It should be read and understood before attempting to install or use the unit.
- Further information can be found in the hardware manual for connecting main unit, FX Series Programming Manual and manual of DeviceNet master units.
- If in doubt at any stage of the installation of FX2N-64DNET DeviceNet Interface Block always consult a professional electrical engineer who is qualified and trained to the local and national standards which apply to the installation site.
- If in doubt about the operation or use of FX2N-64DNET DeviceNet Interface Block please consult the nearest Mitsubishi Electric distributor.
- This manual is subject to change without notice.



FX₂N-64DNET DeviceNet Interface Block

User's Manual

Manual number : JY992D86301

Manual revision : C

Date : March 2006

This manual confers no industrial property rights or any rights of any other kind, nor does it confer any patent licenses. Mitsubishi Electric Corporation cannot be held responsible for any problems involving industrial property rights which may occur as a result of using the contents noted in this manual.

Guidelines for the Safety of the User and Protection of the FX2N-64DNET DeviceNet Interface Block.

This manual provides information for the use of the FX2N-64DNET DeviceNet Interface Block. The manual has been written to be used by trained and competent personnel. The definition of such a person or persons is as follows:

- a) Any engineer who is responsible for the planning, design and construction of automatic equipment using the product associated with this manual, should be of a competent nature, trained and qualified to the local and national standards required to fulfill that role. These engineers should be fully aware of all aspects of safety with regards to automated equipment.
- b) Any commissioning or service engineer must be of a competent nature, trained and qualified to the local and national standards required to fulfill that job. These engineers should also be trained in the use and maintenance of the completed product. This includes being completely familiar with all associated documentation for said product. All maintenance should be carried out in accordance with established safety practices.
- c) All operators of the completed equipment should be trained to use that product in a safe and coordinated manner in compliance to established safety practices. The operators should also be familiar with documentation which is connected with the actual operation of the completed equipment.

Note : The term ‘completed equipment’ refers to a third party constructed device which contains or uses the product associated with this manual.

Note's on the Symbols Used in this Manual

At various times through out this manual certain symbols will be used to highlight points of information which are intended to ensure the users personal safety and protect the integrity of equipment. Whenever any of the following symbols are encountered its associated note must be read and understood. Each of the symbols used will now be listed with a brief description of its meaning.

Hardware Warnings



1) Indicates that the identified danger **WILL** cause physical and property damage.



2) Indicates that the identified danger could **POSSIBLY** cause physical and property damage.



3) Indicates a point of further interest or further explanation.

Software Warnings



4) Indicates special care must be taken when using this element of software.



5) Indicates a special point which the user of the associate software element should be aware.



6) Indicates a point of interest or further explanation.

- Under no circumstances will Mitsubishi Electric be liable responsible for any consequential damage that may arise as a result of the installation or use of this equipment.
- All examples and diagrams shown in this manual are intended only as an aid to understanding the text, not to guarantee operation. Mitsubishi Electric will accept no responsibility for actual use of the product based on these illustrative examples.
- Owing to the very great variety in possible application of this equipment, you must satisfy yourself as to its suitability for your specific application.

Table of Contents

| | |
|--|-----|
| Guideline | iii |
| 1. Introduction..... | 1-1 |
| 1.1 Features of the 64DNET | 1-1 |
| 1.2 External Dimensions and Each Part Name | 1-2 |
| 1.3 System Configuration | 1-4 |
| 1.4 Applicable PLC | 1-5 |
| 2. Wiring and Mounting | 2-1 |
| 2.1 Mounting | 2-1 |
| 2.1.1 Arrangements..... | 2-1 |
| 2.1.2 Mounting..... | 2-2 |
| 2.2 Wiring | 2-3 |
| 2.2.1 Caution | 2-3 |
| 2.2.2 Wiring | 2-4 |
| 2.2.3 Terminating Resistor | 2-4 |
| 3. Specifications | 3-1 |
| 3.1 General Specifications..... | 3-1 |
| 3.2 Power Supply Specifications | 3-1 |
| 3.3 Performance Specifications..... | 3-2 |

| | |
|---|------|
| 4. Allocation of Buffer Memories (BFMs)..... | 4-1 |
| 4.1 Buffer Memories (BFM) Lists | 4-2 |
| 4.2 DeviceNet LINK Enable <BFM #0> (Read/Write)..... | 4-8 |
| 4.3 Module Status about Connection <BFM #1> (Read Only) | 4-9 |
| 4.4 Module Status about DeviceNet <BFM #2> (Read Only)..... | 4-10 |
| 4.5 Number of DeviceNet UCMM Connections <BFM #3> (Read/Write)..... | 4-11 |
| 4.6 DeviceNet Bus-off Counter <BFM #4> (Read/Write)..... | 4-11 |
| 4.7 Number of DeviceNet Retries <BFM #5> (Read/Write)..... | 4-12 |
| 4.8 FROM/TO WDT Setting <BFM #6> (Read/Write)..... | 4-13 |
| 4.9 Mode Control Flags <BFM #7> (Read/Write) | 4-14 |
| 4.10 UCMM Client Connection Setting <BFM #8 ~ #11> (Read/Write)..... | 4-15 |
| 4.11 UCMM Client Sending Trigger <BFM #12 ~ #15> (Read/Write) | 4-16 |
| 4.12 UCMM Client Receiving Status <BFM #16 ~ #19> (Read Only)..... | 4-17 |
| 4.13 UCMM Server Receiving Status <BFM #20 ~ #23> (Read Only)..... | 4-18 |
| 4.14 Error Status <BFM #26 ~ #29> (Read Only) | 4-19 |
| 4.15 Module ID Code <BFM #30> (Read Only) | 4-19 |
| 4.16 UCMM Client WDT switch <BFM #32 ~ #35> (Read/Write)..... | 4-20 |
| 4.17 UCMM Client Connection Status <BFM #36 ~ #39> (Read Only)..... | 4-21 |
| 4.18 UCMM Client Communication Status <BFM #40 ~ #43> (Read Only)..... | 4-22 |
| 4.19 UCMM Server Communication Status <BFM #44 ~ #47> (Read Only) | 4-23 |
| 4.20 Baud Rate <BFM #48> (Read Only) | 4-24 |
| 4.21 Master/Slave Communication Data Area <BFM #100 ~ #133>..... | 4-25 |
| 4.21.1 Send Data/Receive Data Area <BFM #100 ~ #131> (Read/Write)..... | 4-25 |
| 4.21.2 Number of Link Points <BFM #132> (Read/Write)..... | 4-27 |
| 4.21.3 MAC ID <BFM #133> (Read Only)..... | 4-27 |

| | |
|---|------|
| 4.22 UCMM Client Communication Data Area <BFM #10000 ~ #16336> | 4-28 |
| 4.22.1 Send Data/Receive Data Area <BFM #1□□00 ^{*1} ~ #1□□31 ^{*1} > (Read/Write)..... | 4-28 |
| 4.22.2 Length of Assembly Data <BFM #1□□32 ^{*1} > (Read/Write)..... | 4-30 |
| 4.22.3 Length of Sending Common Service Data <BFM #1□□33 ^{*1} > (Read/Write)..... | 4-31 |
| 4.22.4 Length of Receiving Data <BFM #1□□34 ^{*1} > (Read Only)..... | 4-31 |
| 4.22.5 Group Selection <BFM #1□□35 ^{*1} > (Read/Write) | 4-31 |
| 4.22.6 Class ID and Instance ID <BFM #1□□36 ^{*1} >..... | 4-32 |
| 4.22.7 Attribute ID and Service Code <BFM #1□□37 ^{*1} > | 4-32 |
| 4.23 UCMM Server Communication Data Area <BFM #20000 ~ #26336> | 4-33 |
| 4.23.1 Receive Data Area <BFM #2□□00 ^{*1} ~ #2□□31 ^{*1} > (Read Only)..... | 4-33 |
| 4.23.2 Length of Assembly Data <BFM #2□□32 ^{*1} > (Read/Write)..... | 4-35 |
| 4.23.3 Length of Sending Common Service Data <BFM #2□□33 ^{*1} > (Read Only)..... | 4-35 |
| 4.23.4 Length of Receiving Data <BFM #2□□34 ^{*1} > (Read Only)..... | 4-36 |
| 4.23.5 Group Select <BFM #2□□35 ^{*1} > (Read Only) | 4-36 |
| 4.23.6 Class ID and Instance ID <BFM #2□□36 ^{*1} > (Read only) | 4-37 |
| 4.23.7 Attribute ID and Service Code <BFM #2□□37 ^{*1} > (Read only) | 4-37 |
| | |
| 5. DIP Switch Setting..... | 5-1 |
| 5.1 Setting DIP Switch | 5-1 |
| 5.1.1 Mode Setting (DIP Switch: SW0, SW1)..... | 5-1 |
| 5.1.2 Baud Rate Setting (DIP Switch: DR0, DR1)..... | 5-2 |
| 5.1.3 MAC ID Setting (DIP Switches: NA0 ~ NA5)..... | 5-3 |
| | |
| 6. Client/Sever Communication by UCMM Connection..... | 6-1 |
| 6.1 Information of UCMM Connection by 64DNET..... | 6-1 |
| 6.2 How to Use UCMM Communication | 6-2 |

| | |
|--|------|
| 7. Example Program..... | 7-1 |
| 7.1 System Configuration | 7-1 |
| 7.2 Operation | 7-2 |
| 7.3 Example Program..... | 7-3 |
| 7.3.1 Initial Setting..... | 7-4 |
| 7.3.2 Checking Error Status | 7-8 |
| 7.3.3 Input Data and Output Data for Master/Slave I/O Connection | 7-9 |
| 7.3.4 Input Data and Output Data for UCMM Connection..... | 7-10 |
| | |
| 8. Diagnostics..... | 8-1 |
| 8.1 Preliminary Checks..... | 8-1 |
| 8.2 LED Check | 8-2 |
| 8.3 Error Code | 8-5 |
| 8.3.1 64DNET Error Code (BFM #7 Bit 1 is ON)..... | 8-5 |
| 8.3.2 DeviceNet Error Code (BFM #7 Bit 1 is OFF)..... | 8-18 |
| | |
| Appendix A | |
| EDS (Electric Data Sheet)..... | A-1 |

Appendix B

| | |
|--|------|
| Class ID, Instance ID, Attribute ID and Service Code | B-1 |
| B-1: Identity (0x01) | B-1 |
| B-2: Message Router (0x02) | B-2 |
| B-3: DeviceNet (0x03) | B-3 |
| B-4: Assembly (0x04) | B-5 |
| B-5: Connection (0x05) | B-6 |
| B-6: Acknowledge Handler Object (2B Hex) | B-12 |

Appendix C

| | |
|--|-----|
| List of Further Information Manual | C-1 |
|--|-----|

1. Introduction

1.1 Features of the 64DNET

The FX_{2N}-64DNET DeviceNet Interface Block (hereafter called “64DNET”) can be used to connect FX_{0N} and FX_{2N} series programmable controller to a DeviceNet network (hereafter called “DeviceNet”). The 64DNET is a slave (Group 2) on DeviceNet.

- The 64DNET passed the conformance test with A-14 for DeviceNet. (The 64DNET conforms to the DeviceNet specifications Volume 1 release 2.0 and Volume 2 release 2.0.)
- The 64DNET supports the following communication.
 - Master/Slave communication (using Master/Slave I/O connection).
 - Client/Server communication (using UCMM connection).
The 64DNET can use UCMM connection when the connection target is another 64DNET.
- Communication Data Length;
 - Master/Slave communication will both send and receive 64 bytes of data to/from the master.
 - Client/Sever communication will both send and receive 64 bytes of data to/from each 64DNET.
- The communication method for I/O connection supports “polling”, “cyclic” and “change of state”.
- The MAC ID and baud rate is adjusted by DIP switches. See chapter 5.
- The communication between the connected main unit and the internal buffer memory of the 64DNET is handled by FROM/TO instructions.

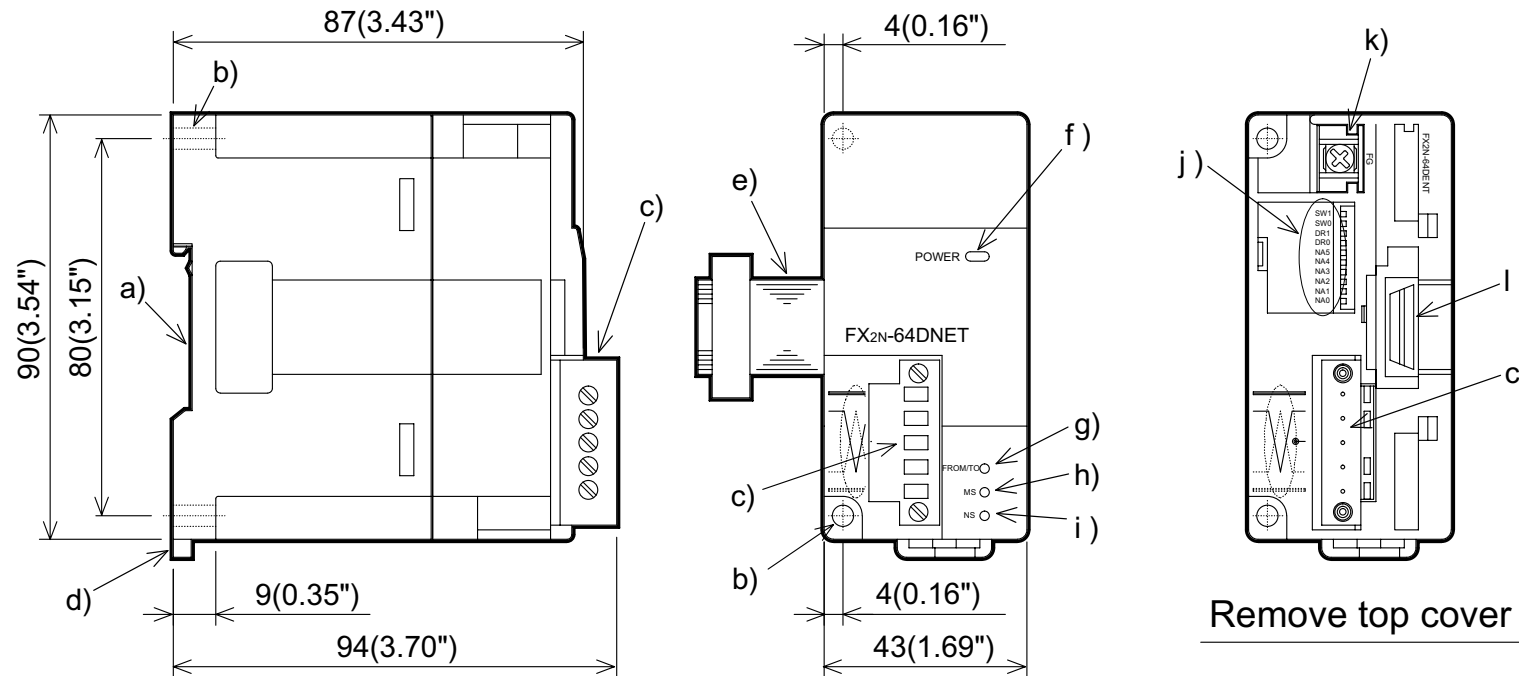
DeviceNet is a registered trademark of the Open DeviceNet Vendor Association, Inc.

1.2 External Dimensions and Each Part Name

Dimensions: mm (inches) MASS (Weight):Approx. 0.2 kg (0.44 lbs)

Accessory: Special block No. label

Figure 1.1: External Dimensions

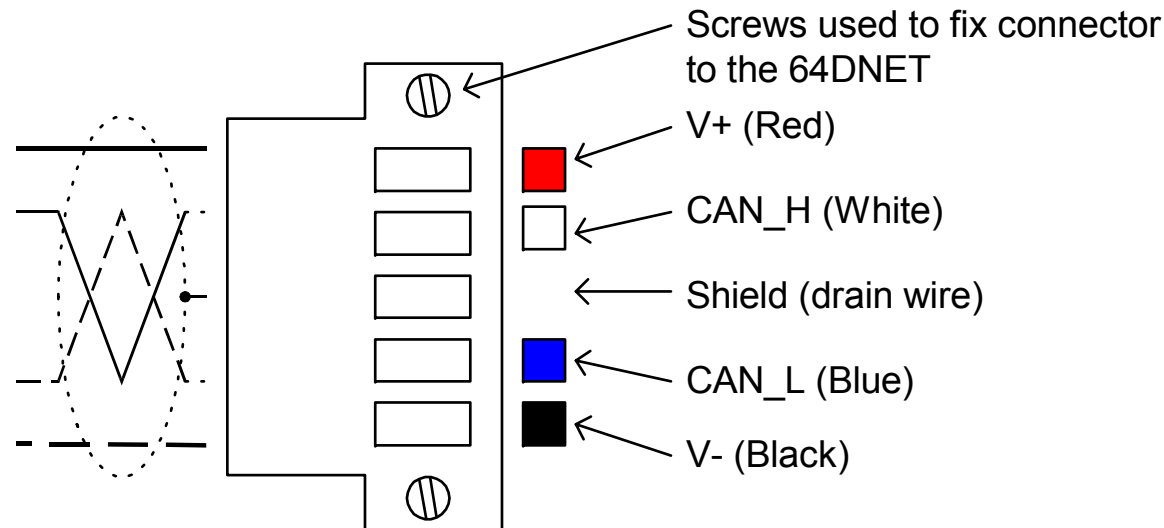


a) Groove for mounting DIN rail (DIN46277) <DIN rail width: 35mm (1.38)">

b) Direct mounting hole (2- ϕ 4.5 (0.18"))

c) Connector for connecting DeviceNet communication cable

Figure 1.2: Connector layout



d) Hook for mounting DIN rail

e) Extension cable

f) POWER LED:

g) FROM/TO LED

h) MS LED (Module Status LED)

i) NS LED (Network Status LED)

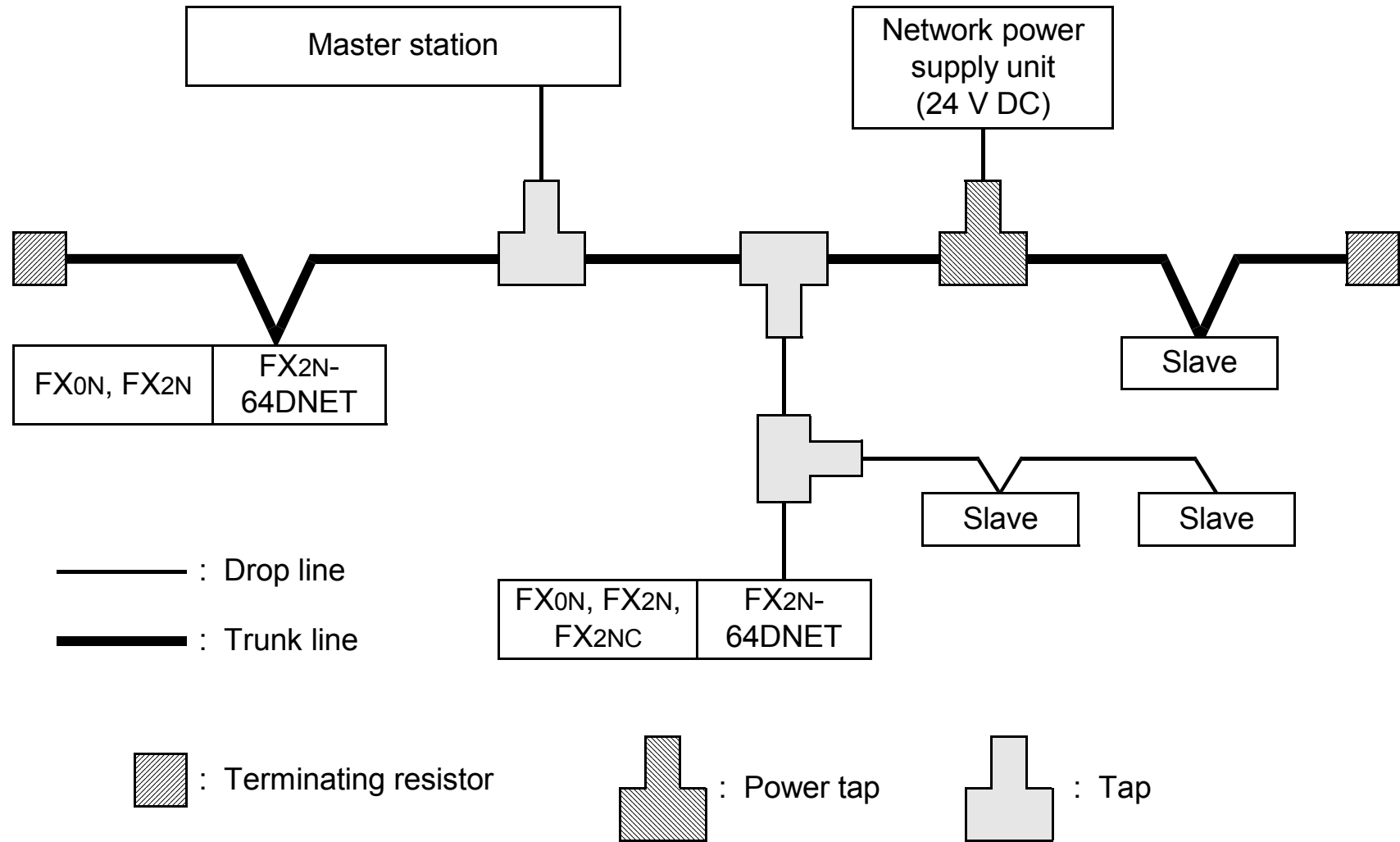
j) DIP switch for setting the MAC ID, baud rate and operation mode. See chapter 5.

k) FG Terminal (screw terminal: M3 (0.12"))

l) Connector for connecting extension cable

1.3 System Configuration

Figure 1.3: System Configuration



1.4 Applicable PLC

Table 1.1: Agreement PLC

| Programmable Controllers | Description |
|-----------------------------|--------------|
| FX ₀ N series | All products |
| FX ₂ N series | |

2. Wiring and Mounting

2.1 Mounting

2.1.1 Arrangements

The 64DNET connects on the right side of connected main unit or extension unit/block (including special function blocks). For further information of mounting arrangements, refer to the hardware manual of the connected main unit.

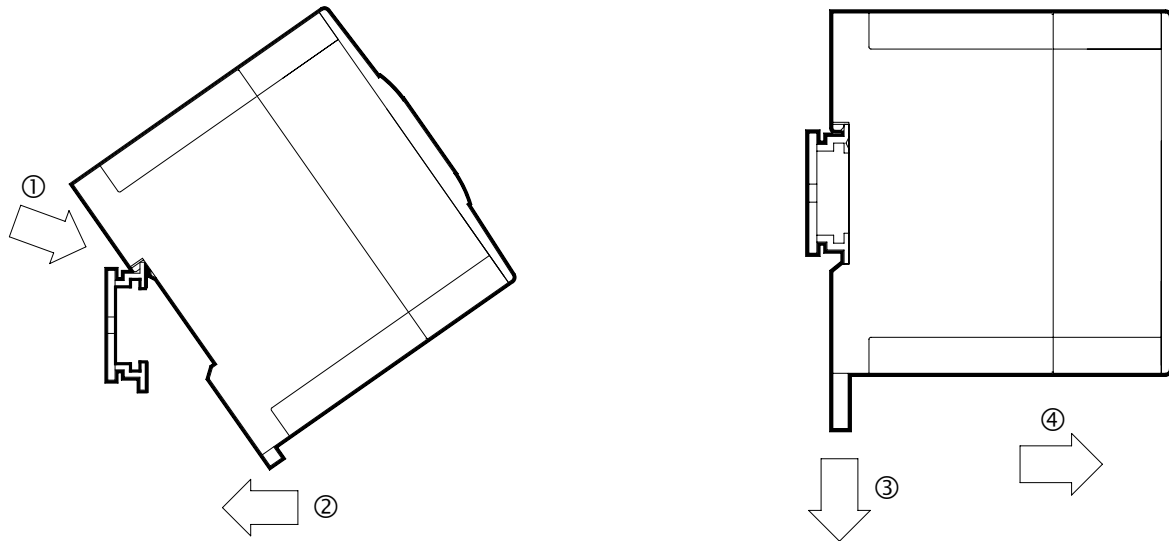
2.1.2 Mounting

The mounting method of the 64DNET can be DIN rail mounting or direct wall mounting.

1) DIN rail mounting

- Align the upper side of the DIN rail mounting groove of the 64DNET with a DIN rail*1 (①), and push it on the DIN rail(②). See Figure 2.1.
- When removing the 64DNET from the DIN rail, the hook for DIN rail is pulled (③), and the 64DNET is removed (④). See Figure 2.1.

Figure 2.1: Attach to DIN Rail and Remove from DIN Rail



*1 Uses DIN 46277 <35mm (1.38")>

2) Direct mounting to back walls

The 64DNET can be mounted with the M4 screw by using the direct mounting hole. However, an interval space between each unit of 1 ~ 2 mm is necessary.

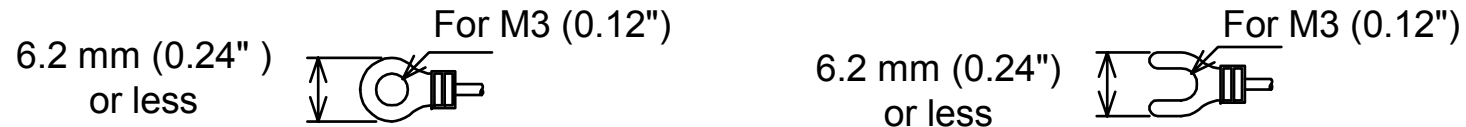
2.2 Wiring

2.2.1 Caution



- 1) Do not lay signal cable near to high voltage power cable or house them in the same trunking duct. Effects of noise or surge induction may occur. Keep signal cables a safe distance of more than 100 mm (3.94") from these power cables.
- 2) The terminal screws of the 64DNET are M3 (0.12"), therefore the crimp style terminal (see drawing) suitable for use with these screws should be fitted to the cable for wiring.

Figure 2.2: Crimp Terminals

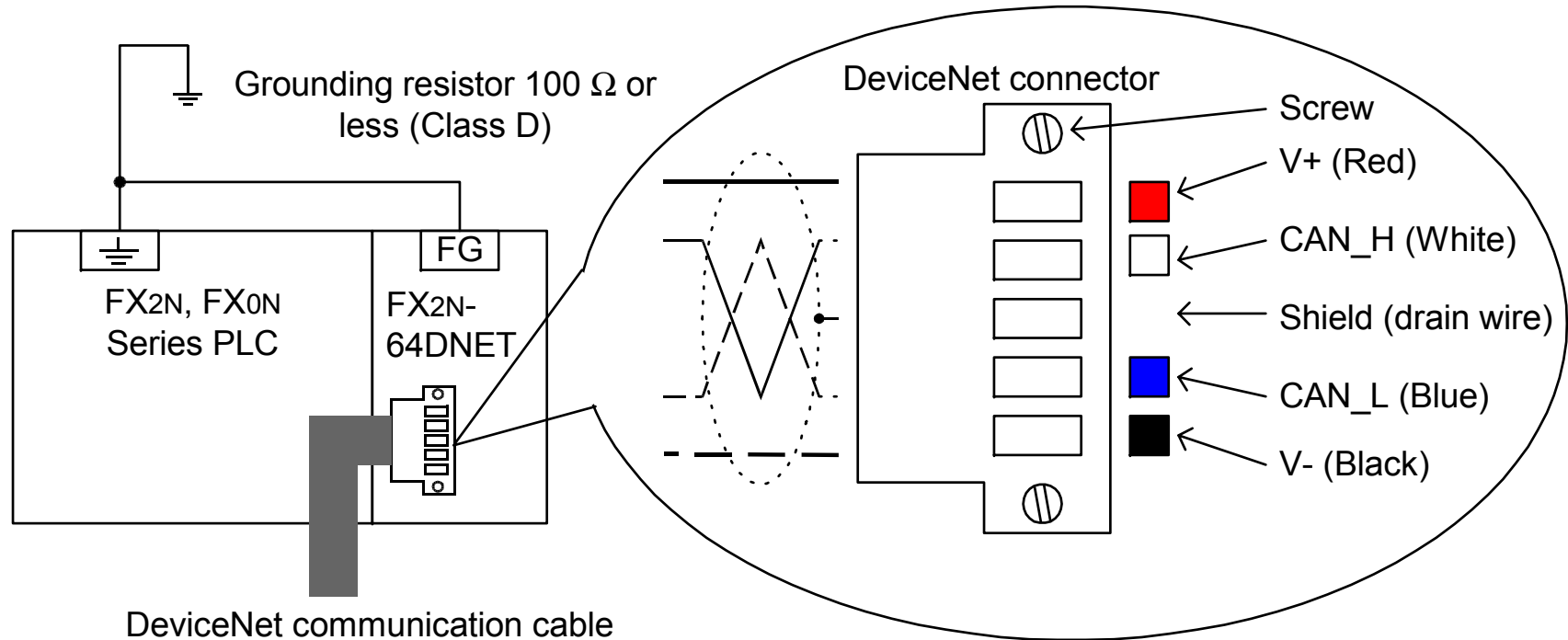


- 3) The terminal tightening torque is 0.5 ~ 0.8 N·m. Tighten securely to avoid malfunction.
- 4) Cut off all phases of power source before installation or performing wiring work in order to avoid electric shock or damage of product.
- 5) Replace the provided terminal cover before supplying power and operating the unit after installation or wiring work in order to avoid electric shock.

2.2.2 Wiring

Connects Each color code of DeviceNet communication cable to same color part in DeviceNet connector. Please fix this connector to the 64DNET by the screws provided.

Figure 2.3: Wiring



2.2.3 Terminating Resistor

The units at each end of the DeviceNet must have a terminating resistor. However, the 64DNET does not have a terminating resistance built-in.

3. Specifications

3.1 General Specifications

Table 3.1: General Specifications

| Items | Description |
|---|---|
| General specifications excluding Dielectric Withstand Voltage | Same as those of the main unit |
| Dielectric Withstand Voltage | 500 V AC > 1 min, tested between the DC power line in the DeviceNet connector and ground |
| Complies with | EN50325 (conformance tested with A-14) UL508 |

3.2 Power Supply Specifications

Table 3.2: Power Supply Specifications

| Items | Description |
|--------------------------------|---|
| Power Supply For Communication | 50 mA at 24 V DC supplied from V+, V- terminals |
| Internal Power Supply | 120 mA at 5 V DC supplied via extension cable |

3.3 Performance Specifications

Table 3.3: Performance Specifications

| Items | | | Specifications | |
|----------------------|----------------------------------|--|--|---|
| Node Type | | | G2 server | |
| Station Numbers | | | 0 ~ 63 | |
| Transmission Cable | | | Shielded twisted-pair cable conforming to DeviceNet specifications | |
| MAC ID | | | 0 ~ 63 | |
| Supported Baud Rates | | | 125 kbps, 250 kbps, 500 kbps | |
| Maximum Cable Length | | | See Note 1. | |
| Communication Data | Explicit Connection | Predefined Master/Slave Connection Set | Number of Connection | 1 (Group 2) |
| | | | Transmission Time-out | 2,000 ms (ACK time-out) |
| | UCMM Client/Server Connection *1 | Number of Connection | 62/62 (Group 1, 3) <recommend group 3> | |
| | | | Communication Data Length | Sending: Maximum 64 bytes /1 connection Receiving: Maximum 64 bytes/1 connection |

*1 The 64DNET can use UCMM connection when the connection target is another 64DNET.

Table 3.3: Performance Specifications

| Items | | Specifications |
|-------------------------------|----------------|--|
| Communication Data | | Supported Communication Type |
| | I/O Connection | Communication Data Length |
| | | Polling, Cyclic, Change of State Sending: Maximum 64 bytes (Fragmentation is allowed.) Receiving: Maximum 64 bytes (Fragmentation is allowed.) |
| Applicable PLC | | FX2N Series, FX0N Series |
| Communication with PLC | | FROM/TO instruction |
| Number of Occupied I/O Points | | 8 points taken from the programmable controller extension bus (can be either input or output) |
| Module ID Code | | K 7090 |
| LED Indicators | POWER LED | |
| | FROM/TO LED | |
| | MS LED | |
| | NS LED | |
| | | Lit when 5 V DC power supplied from main unit or extension unit. Lit when 64DNET is accessed from main unit. See Note 2. See Note 3. |

Note 1; Maximum Cable Length**Table 3.4: Maximum Cable Length**

| Using Baud Rate | Trunk Line Maximum Cable Length | Drop Line | |
|-----------------|---|----------------|----------------------|
| | | Maximum Length | Maximum Total Length |
| 125 kbps | Thick cable length + 5 × thin cable length ≤ 500 m (1,640') | 6 m (20') | 156 m (512') |
| 250 kbps | Thick cable length + 2.5 × thin cable length ≤ 250 m (820') | | 78 m (23') |
| 500 kbps | Thick cable length + thin cable length ≤ 100 m (328') | | 39 m (128') |

Note 2; MS LED Status (Module Status LED)**Table 3.5: MS LED Status (Module Status LED)**

| MS LED Status | 64DNET Status | Description |
|---|---------------------|---|
| Not lit | Power OFF | The power supply is insufficient. |
| Lit Green | Normal Operation | The 64DNET is in normally operation mode. |
| Flashing Green at 0.5 s intervals | Stand-by | The 64DNET is in stand-by status or setting error. |
| Flashing Red at 0.5 s intervals | Minor Fault | A recoverable error has occurred. |
| Lit Red | Unrecoverable Fault | An unrecoverable error has occurred. Please contact a service representative. |
| Flashing Green and Red at 0.5 s intervals | Self Test | The 64DNET is in self testing mode. |

Note 3; NS LED Network Status**Table 3.6: NS LED Status (Network Status LED)**

| MS LED Status | 64DNET Status | Description |
|--|-----------------------------------|--|
| Not lit | Not On-line | The 64DNET is not on-line. <ul style="list-style-type: none"> • The power supply voltage is abnormal. • Dup_MAC_ID test is not finished. |
| Lit Green | Link OK | The 64DNET is on-line and has established the connection. |
| Flashing Green at 0.5 s intervals | On-line Status Without Connection | The 64DNET is on-line but has not the established the connection. |
| Flashing Red at 0.5 s intervals | Communication Time-out | The 64DNET I/O connection is in time-out status. |
| Lit Red | Critical link Failure | Communication is not possible. |
| Flashing Green and Red at 0.25 s intervals | Communication Fault | The 64DNET is the Communication Faulted for detecting the Network Accessing Error. After this module will receive the Identify Communication Faulted Request. |

**Note:**

When the power supply is turned on, MS LED and NS LED quickly flashes green, then red, and are turned off. This operation is normal.

4. Allocation of Buffer Memories (BFMs)



Caution;

1) Do not access the “Not used” section of the buffer memory (BFM #24, #25, #31, #50 ~ #99, #134 ~ #999, #1□□38^{*1} ~ #1□□99^{*1}, #16338 ~ #19999, #2□□38^{*1} ~ #2□□99^{*1}) by the FROM/TO instruction. There is a possibility to cause abnormal operation of the 64DNET if accessing these buffer memories.

*1 □□ has a value of 0 ~ 62.

2) Do not write to (access by TO instruction) the “Read only” section of the buffer memory (BFM #1, #2, #30, #16 ~ #23, #26, #27, #30, #36 ~ #49, #133, #2□□00^{*1} ~ #2□□31^{*1}, #2□□33^{*1} ~ #2□□37^{*1}) in the programmable controller. It is not possible to operate the 64DNET by writing to (accessing by TO instruction) these buffer memories.

*1 □□ has a value of 0 ~ 63.



Note:

If the sending data and receiving data buffers have the same buffer memory addresses (BFM #100 ~ #1031, #1□□00^{*1} ~ #1□□31^{*1}) for use with FROM and TO instructions, this means it is not possible to check the buffer memory data using a FROM instruction because this instruction only reads receive buffer data.

*1 □□ has a value of 0 ~ 63.

4.1 Buffer Memories (BFM) Lists

Table 4.1: BFM Lists

| BFM #No. | Description | | Default Value | FROM/TO Instruction (Read/Write) |
|--------------|--------------------------------------|------------------------|---------------|----------------------------------|
| | FROM Instruction (Read) | TO Instruction (Write) | | |
| BFM #0 | DeviceNet LINK enable | | K0 | Read /Write |
| BFM #1 | Module status about connection | | — | Read Only |
| BFM #2 | Module status about DeviceNet | | | |
| BFM #3 | Number of DeviceNet UCMM connections | | K0808 | Read /Write |
| BFM #4 | DeviceNet bus-off counter | | K0 | |
| BFM #5 | Number of DeviceNet retries | | K2 | |
| BFM #6 | FROM/TO WDT (watchdog timer) setting | | K20 | |
| BFM #7 | Module control flag | | K0 | |
| BFM #8 ~ 11 | UCMM client connection status | | | |
| BFM #12 ~ 15 | UCMM client sending trigger | | | |
| BFM #16 ~ 19 | UCMM client receiving status | | | |
| BFM #20 ~ 23 | UCMM server receiving status | | | Read Only |
| BFM #24,25 | Not used | | | |

Table 4.1: BFM Lists

| BFM #No. | Description | | Default Value | FROM/TO Instruction (Read/Write) |
|-----------------|-------------------------------------|------------------------|---------------|----------------------------------|
| | FROM Instruction (Read) | TO Instruction (Write) | | |
| BFM #26 | Error flag | | K0 | Read Only |
| BFM #27 | Error code | | | |
| BFM #28 | Error flag latch | | | Read/Write |
| BFM #29 | Error code latch | | | |
| BFM #30 | Module ID code | | K7090 | Read only |
| BFM #31 | Not used | | | |
| BFM #32 ~ 35 | UCMM client WDT switch | | K0 | Read/Write |
| BFM #36 ~ 39 | UCMM client connection status | | | Read only |
| BFM #40 ~ 43 | UCMM client communication status | | | |
| BFM #44 ~ 47 | UCMM server communication status | | | |
| BFM #48 | Baud rate | | | |
| BFM #49 | DIP switch status | | | H BF |
| BFM #50 ~ 99 | Not used | | | |
| BFM #100 ~ 131 | Master/Slave receive data | Master/Slave send data | K0 | Read/Write |
| BFM #132 | Number of link points for DeviceNet | | H0404 | |
| BFM #133 | DeviceNet MAC ID | | K63 | Read only |
| BFM #134 ~ 9999 | Not used | | | |

Table 4.1: BFM Lists

| BFM #No. | Description | | Default Value | FROM/TO Instruction (Read/Write) | |
|--------------------|-------------------------|---|---------------|----------------------------------|------------|
| | FROM Instruction (Read) | TO Instruction (Write) | | | |
| BFM #10000 ~ 10031 | UCMM Client 0 | Receive data | Send data | K0 | Read/Write |
| BFM #10032 | | Length of assembly data | | H0404 | |
| BFM #10033 | | Length of sending common service data | | K0 | Read only |
| BFM #10034 | | Length of received data (assembly or common service data) | | | |
| BFM #10035 | | Group select | | K3 | Read/Write |
| BFM #10036 | | DeviceNet class ID and instance ID | | K0 | |
| BFM #10037 | | DeviceNet attribute ID and service ID | | | |
| BFM #10038 ~ 10099 | Not used | | | | |
| BFM #10100 ~ 10131 | UCMM Client 1 | Receive data | Send data | K0 | Read/Write |
| BFM #10132 | | Length of assembly data | | H0404 | |
| BFM #10133 | | Length of sending common service data | | K0 | Read only |
| BFM #10134 | | Length of received data (assembly or common service data) | | | |
| BFM #10135 | | Group selection | | K3 | Read/Write |
| BFM #10136 | | DeviceNet class ID and instance ID | | H466 | |
| BFM #10137 | | DeviceNet attribute ID and service ID | | H300 | |
| BFM #10138 ~ 10199 | Not used | | | | |

Table 4.1: BFM Lists

| BFM #No. | Description | | Default Value | FROM/TO Instruction (Read/Write) | |
|--------------------|-------------------------|---|---------------|----------------------------------|------------|
| | FROM Instruction (Read) | TO Instruction (Write) | | | |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ | |
| BFM #16300 ~ 16331 | UCMM Client 63 | Receive data | Send data | K0 | Read/Write |
| BFM #16332 | | Length of assembly data | | H0404 | |
| BFM #16333 | | Length of sending common service data | | K0 | Read only |
| BFM #16334 | | Length of received data (assembly or common service data) | | | |
| BFM #16335 | | Group selection | | K3 | Read/Write |
| BFM #16336 | | DeviceNet class ID and instance ID | | H466 | |
| BFM #16337 | | DeviceNet attribute ID and service ID | | H300 | |
| BFM #16338 ~ 19999 | | Not used | | | |
| BFM #20000 ~ 20031 | UCMM Server 0 | Receive data | | K0 | Read only |
| BFM #20032 | | Length of assembly data | | H0404 | Read/Write |
| BFM #20033 | | Length of sending common service data | | K0 | Read only |
| BFM #20034 | | Length of received data (assembly or common service data) | | | |
| BFM #20035 | | Group selection | | K3 | |
| BFM #20036 | | DeviceNet class ID and instance ID | | H466 | |
| BFM #20037 | | DeviceNet attribute ID and service ID | | H300 | |

Table 4.1: BFM Lists

| BFM #No. | Description | | Default Value | FROM/TO Instruction (Read/Write) |
|--------------------|-------------------------|---|---------------|----------------------------------|
| | FROM Instruction (Read) | TO Instruction (Write) | | |
| BFM #20038 ~ 20099 | Not used | | | |
| BFM #20100 ~ 20131 | UCMM Server 1 | Receive data | K0 | Read only |
| BFM #20132 | | Length of assembly data | H0404 | Read/Write |
| BFM #20133 | | Length of sending common service data | K0 | Read only |
| BFM #20134 | | Length of received data (assembly or common service data) | | |
| BFM #20135 | | Group selection | K3 | |
| BFM #20136 | | DeviceNet class ID and instance ID | H466 | |
| BFM #20137 | | DeviceNet attribute ID and service ID | H300 | |
| BFM #20138 ~ 20199 | | Not used | | |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |

Table 4.1: BFM Lists

| BFM #No. | Description | | Default Value | FROM/TO Instruction (Read/Write) |
|--------------------|-------------------------|---|---------------|----------------------------------|
| | FROM Instruction (Read) | TO Instruction (Write) | | |
| BFM #26300 ~ 26331 | UCMM Server 63 | Receive data | K0 | Read only |
| BFM #26332 | | Length of assembly data | H0404 | Read/Write |
| BFM #26333 | | Length of sending common service data | K0 | Read only |
| BFM #26334 | | Length of received data (assembly or common service data) | | |
| BFM #26335 | | Group selection | K3 | |
| BFM #26336 | | DeviceNet class ID and instance ID | H466 | |
| BFM #26337 | | DeviceNet attribute ID and service ID | H300 | |

4.2 DeviceNet LINK Enable <BFM #0> (Read/Write)

This buffer memory is used to enable the 64DNET on DeviceNet.

Table 4.2: DeviceNet LINK Enable <BFM #0>

| Value | Description |
|-----------|--------------------------------------|
| K0 | The 64DNET is off-line on DeviceNet. |
| K1 | The 64DNET is on-line on DeviceNet. |
| Otherwise | Not used |

4.3 Module Status about Connection <BFM #1> (Read Only)

This buffer memory is used for checking connection status of the 64DNET.
The connection status is allocated as follows.

Table 4.3: Module Status about Connection <BFM #1>

| Bit No. | Name | Description | |
|------------|--|---|---|
| | | ON (1) | OFF (0) |
| Bit 0 | Master/Slave I/O Connection | The 64DNET connects to the master station by I/O connection. | The 64DNET does not connect to the master station by I/O Connection. |
| Bit 1 | Master/Slave Explicit Messaging Connection | The 64DNET connects to the master station by Explicit Connection. | The 64DNET does not connect to the master station by Explicit Connection. |
| Bit 2 | UCMM Connection | The 64DNET connects to the other node by the UCMM Connection. | The 64DNET does not connect to the other node by the UCMM Connection. |
| Bit 3 ~ 15 | Not used | | |

4.4 Module Status about DeviceNet <BFM #2> (Read Only)

This buffer memory is used for checking the DeviceNet link status of the 64DNET. The value of this buffer memory's mean shows bellow table.

Table 4.4: Module Status about DeviceNet <BFM #2>

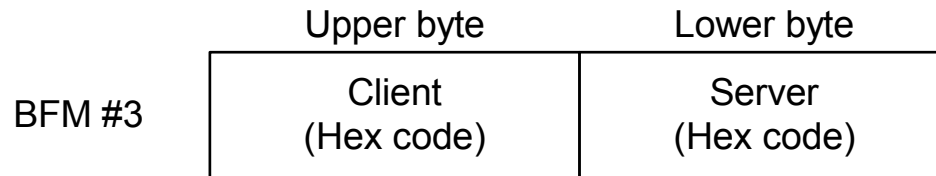
| Value | 64DNET Status | MS LED Status | Description |
|-------|-------------------------------|--|---|
| K0 | Power off | Not lit | The power supply is insufficient. |
| K10 | Device Self test | Flashing Green and Red at 0.5 s interval | The 64DNET is in self-testing status. |
| K30 | Device operational (On-line) | Lit Green | The 64DNET is on-line in normally operation. |
| K31 | Device operational (Off-line) | | The 64DNET is off-line in normally operation. |
| K50 | Unrecoverable Fault | Lit Red | An unrecoverable error has occurred. Please contact a service representative. |

4.5 Number of DeviceNet UCMM Connections <BFM #3> (Read/Write)

This buffer memory is used for setting or checking the number of the UCMM Connections on the DeviceNet.

The setting Range of client is 00 ~ 40 hex, and the setting range of server is 01 ~ 40 hex. Default value is 0808 hex (client: Max. 8 connection, server: Max. 8 connection)

Figure 4.1: Number of DeviceNet UCMM Connections <BFM #3>



4.6 DeviceNet Bus-off Counter <BFM #4> (Read/Write)

This buffer memory is used for checking the number of times the bus-off has occurred on the DeviceNet. To reset a bus off counter, this buffer memory writes and has permitted only “K0”. Default value is “K0”.



Note:

If any increase in this value, 64DNET does not have stable communication on DeviceNet. Please check wiring and settings.

4.7 Number of DeviceNet Retries <BFM #5> (Read/Write)

This buffer memory is used for setting number of DeviceNet retries on the UCMM connection. Default value is K2. This setting range is 0 ~ 255.



Note:

This process is performed only if the UCMM client WDT switch is ON. For the UCMM client WDT switch information, refer to section 4.16.

4.8 FROM/TO WDT Setting <BFM #6> (Read/Write)

This buffer memory is used for setting the FROM/TO WDT. This WDT is adjusted in 10 ms steps. Default value is 20 ($20 \times 10 \text{ ms} = 200 \text{ ms}$). When this value is 0, the watchdog timer is disabled. When all BFMs (BFM #0 ~ #26337) are not accessed by FROM/TO instructions, the watchdog timer begins operating.



Caution:

1) Using the Master/Slave communication (Master/Slave connection)

If the watchdog timer reaches the set time ($\text{BFM \#6} \times 10 \text{ ms}$), the 64DNET is in WDT time-out status. At this time, the 64DNET will store error codes (BFM #27, #29), set error flags (BFM #26, #28), and all the send data from the slave to the master station is cleared. However, 64DNET does not stop all data exchange to the master station. The 64DNET will send "K0" to the master station.

Please the PLC access the BFMs (one or more) each time as in the example program (see section 6.3).

2) Using the Client/Server communication (UCMM connection)

In this case, the connection of the UCMM connection follows the UCMM Client WDT switch setting (BFM #32 ~ #35). For UCMM Client WDT switch information, refer to section 4.16.

4.9 Mode Control Flags <BFM #7> (Read/Write)

This buffer memory is used for changing mode of the receiving data flag and contents of error code. For error code, refer to chapter 9.

Table 4.5: Mode Control Flags

| Bit No. | Name | Description | |
|------------|-----------------|---|---|
| | | OFF (0) | ON (1) |
| Bit 0 | Receiving mode | See Note | |
| Bit 1 | Error code type | Upper byte: general error code Lower byte: additional error code | Upper byte: service type Lower byte: detailed error code |
| Bit 2 ~ 15 | Not used | | |

Note:

When this bit 1 is turned ON, 64DNET does not receive new data from other nodes to which the receiving data flag (the UCMM client receiving status and the UCMM server receiving status) is turned ON. When the reception flag is turned OFF, new reception data can be received.

When this bit 1 is turned OFF, 64DNET will receive new data from other node to which the receiving data flag is turned ON. If the PLC does not read data from BFMs of receiving data area, this data will be replaced by new data.

4.10 UCMM Client Connection Setting <BFM #8 ~ #11> (Read/Write)

These buffer memories are used for connecting to the server by the UCMM connection. Turn On MAC ID server bit to connect the 64DNET as a client via the UCMM connection. The allocation of the bits is as shown in the table below.

Table 4.6: UCMM Client Connection Setting <BFM #8 ~ #11>

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #8 | Bit 0 | 0 |
| | Bit 1 | 1 |
| | Bit 2 | 2 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 15 |
| BFM #9 | Bit 0 | 16 |
| | Bit 1 | 17 |
| | Bit 2 | 18 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 31 |

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #10 | Bit 0 | 32 |
| | Bit 1 | 33 |
| | Bit 2 | 34 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 47 |
| BFM #11 | Bit 0 | 18 |
| | Bit 1 | 19 |
| | Bit 2 | 50 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 63 |



Caution:

If the connection is cut while sending or receiving data, the 64DNET will not finish sending/receiving the interrupted data. the data may not be correct or complete.

4.11 UCMM Client Sending Trigger <BFM #12 ~ #15> (Read/Write)

These buffer memories are used for sending data to the server to which 64DNET is connected by the UCMM connection of the client.

When sending data (BFM #1□□00^{*1} ~ #1□□31^{*1}), the bit which corresponds to the server destination is turned OFF → ON. After the data (BFM #1□□00^{*1} ~ #1□□31^{*1}) is sent, this bit is automatically turned OFF. The allocation of the bits is as shown in the table below.

For BFM #1□□00^{*1} ~ #1□□31^{*1}, refer to section 4.23.

Table 4.7: UCMM Client Sending Trigger <BFM #12 ~ #15>

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #12 | Bit 0 | 0 |
| | Bit 1 | 1 |
| | Bit 2 | 2 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 15 |
| BFM #13 | Bit 0 | 16 |
| | Bit 1 | 17 |
| | Bit 2 | 18 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 31 |

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #14 | Bit 0 | 32 |
| | Bit 1 | 33 |
| | Bit 2 | 34 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 47 |
| BFM #15 | Bit 0 | 18 |
| | Bit 1 | 19 |
| | Bit 2 | 50 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 63 |

*1 The □□ is entered value of 0 ~ 63.

4.12 UCMM Client Receiving Status <BFM #16 ~ #19> (Read Only)

These buffer memories are used for reading received data from the server to which 64DNET is connected by the UCMM connection of the client.

When receiving data from server, the bit which corresponds to the server of the destination is turned OFF → ON. After the PLC reads data from BFMs (BFM #1□□00^{*1} ~ #1□□31^{*1}) in the received data area for the server, this bit automatically is turned OFF. The allocation of the bits is as shown in the table below. For BFM #1□□00^{*1} ~ #1□□31^{*1}, refer to section 4.23.

Table 4.8: UCMM Client Receiving Status <BFM # 16~ #19>

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #16 | Bit 0 | 0 |
| | Bit 1 | 1 |
| | Bit 2 | 2 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 15 |
| BFM #17 | Bit 0 | 16 |
| | Bit 1 | 17 |
| | Bit 2 | 18 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 31 |
| BFM #18 | Bit 0 | 32 |
| | Bit 1 | 33 |
| | Bit 2 | 34 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 47 |
| BFM #19 | Bit 0 | 18 |
| | Bit 1 | 19 |
| | Bit 2 | 50 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 63 |

*1 The □□ is entered value of 0 ~ 63.

4.13 UCMM Server Receiving Status <BFM #20 ~ #23> (Read Only)

These buffer memories are used for reading received data from the client to which 64DNET (server) is connected by the UCMM connection.

When receiving data from the client, the bit which corresponds to the server is turned OFF → ON. After the PLC reads data from BFMs (BFM #2□□00^{*1} ~ #2□□31^{*1}) in the received data area for the client, this bit automatically is turned OFF. The allocation of the bits is as shown in the table below. For BFM #1□□00^{*1} ~ #1□□31^{*1}, refer to section 4.24.

Table 4.9: UCMM Client Connection Setting

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #20 | Bit 0 | 0 |
| | Bit 1 | 1 |
| | Bit 2 | 2 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 15 |
| BFM #21 | Bit 0 | 16 |
| | Bit 1 | 17 |
| | Bit 2 | 18 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 31 |

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #22 | Bit 0 | 32 |
| | Bit 1 | 33 |
| | Bit 2 | 34 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 47 |
| BFM #23 | Bit 0 | 18 |
| | Bit 1 | 19 |
| | Bit 2 | 50 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 63 |

*2 The □□ is entered value of 0 ~ 63.

4.14 Error Status <BFM #26 ~ #29> (Read Only)

These buffer memories are used for checking error status and error codes.

If an error occurs, the 64DNET stores "K1" to BFM #26 and #28. At the same time, the 64DNET stores the error code to BFM #27 and #29. However, if the error automatically recovers, BFM #26 and #27 will be cleared by 64DNET. BFM #28 and #29 are not cleared. BFM #28 and #29 can be cleared by the PLC when writing "K0" to these BFMs by the TO instruction. For error codes, refer to chapter 8.

Table 4.10: Error Status <BFM #26, #28>

| BFM No. | Name | Description | | |
|---------|--------------------|---|---|------------|
| | | K0 | K1 | K2 or more |
| BFM #26 | Present error flag | Normal operation (BFM #27 is cleared by 64DNET) | Occurred Error (error code is stored BFM #27) | Not used |
| BFM #28 | Error flag latch | Normal operation | Occurred Error (error code is stored BFM #29) | |

4.15 Module ID Code <BFM #30> (Read Only)

The identification code for a 64DNET is read by using the FROM instruction. The identification code for the 64DNET is "K7090". By reading this identification code, the user may create built-in checking routines to check whether the physical position of 64DNET matches to that software.

4.16 UCMM Client WDT switch <BFM #32 ~ #35> (Read/Write)

These buffer memories are used for setting either to cut off the connection or not cut off the connection, when the time-out occurs in the Explicit Message.

If the MAC ID bit is ON, an Explicit error time out does not cut off the connection to the server. The allocation of the bits is as shown in the table below.

Table 4.11: UCMM Client WDT Switch <BFM #32 ~ #35>

| BFM No. | Bit No. | MAC ID | Description | | | |
|---------|---------|--------|---|---|---|---|
| | | | ON (1) | OFF (0) | | |
| BFM #32 | Bit 0 | 0 | Does not cut off connection for a server when time-out occurs in Explicit Message | Cuts off connection for a server when time-out occurs in Explicit Message | | |
| | Bit 1 | 1 | | | | |
| | Bit 2 | 2 | | | | |
| | : | : | | | | |
| | : | : | | | | |
| Bit 15 | 15 | | | | | |
| BFM #33 | Bit 0 | 16 | | | Does not cut off connection for a server when time-out occurs in Explicit Message | Cuts off connection for a server when time-out occurs in Explicit Message |
| | Bit 1 | 17 | | | | |
| | Bit 2 | 18 | | | | |
| | : | : | | | | |
| | : | : | | | | |
| Bit 15 | 31 | | | | | |
| BFM #34 | Bit 0 | 32 | Does not cut off connection for a server when time-out occurs in Explicit Message | Cuts off connection for a server when time-out occurs in Explicit Message | | |
| | Bit 1 | 33 | | | | |
| | Bit 2 | 34 | | | | |
| | : | : | | | | |
| | : | : | | | | |
| Bit 15 | 47 | | | | | |
| BFM #35 | Bit 0 | 18 | | | Does not cut off connection for a server when time-out occurs in Explicit Message | Cuts off connection for a server when time-out occurs in Explicit Message |
| | Bit 1 | 19 | | | | |
| | Bit 2 | 50 | | | | |
| | : | : | | | | |
| | : | : | | | | |
| Bit 15 | 63 | | | | | |

4.17 UCMM Client Connection Status <BFM #36 ~ #39> (Read Only)

These buffer memories are used for checking if the server has server been connected. The bit which corresponds to the MAC ID of the server to which 64DNET has connected previously as a client is turned ON. The allocation of the bits is as shown in the table below.

Table 4.12: UCMM Client Connection Status <BFM #36 ~ #39>

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #36 | Bit 0 | 0 |
| | Bit 1 | 1 |
| | Bit 2 | 2 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 15 |
| BFM #37 | Bit 0 | 16 |
| | Bit 1 | 17 |
| | Bit 2 | 18 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 31 |

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #38 | Bit 0 | 32 |
| | Bit 1 | 33 |
| | Bit 2 | 34 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 47 |
| BFM #39 | Bit 0 | 18 |
| | Bit 1 | 19 |
| | Bit 2 | 50 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 63 |

4.18 UCMM Client Communication Status <BFM #40 ~ #43> (Read Only)

These buffer memories are used to check the status of correct server communication. The bit which corresponds to the MAC ID of the server to which 64DNET communicates correctly as a client is turned ON. The allocation of the bit is as shown in the table below.

Table 4.13: UCMM Client Communication Status <BFM #40 ~ #43>

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #40 | Bit 0 | 0 |
| | Bit 1 | 1 |
| | Bit 2 | 2 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 15 |
| BFM #41 | Bit 0 | 16 |
| | Bit 1 | 17 |
| | Bit 2 | 18 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 31 |

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #42 | Bit 0 | 32 |
| | Bit 1 | 33 |
| | Bit 2 | 34 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 47 |
| BFM #43 | Bit 0 | 18 |
| | Bit 1 | 19 |
| | Bit 2 | 50 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 63 |

4.19 UCMM Server Communication Status <BFM #44 ~ #47> (Read Only)

These buffer memories are used for checking the status of client. connection
The bit which corresponds to the MAC ID of the client to which 64DNET is connected by client is turned ON. The allocation of the bits is as shown in the table below.

Table 4.14: UCMM Server Communication Status <BFM #44 ~ #47>

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #44 | Bit 0 | 0 |
| | Bit 1 | 1 |
| | Bit 2 | 2 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 15 |
| BFM #45 | Bit 0 | 16 |
| | Bit 1 | 17 |
| | Bit 2 | 18 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 31 |

| BFM No. | Bit No. | MAC ID |
|---------|---------|--------|
| BFM #46 | Bit 0 | 32 |
| | Bit 1 | 33 |
| | Bit 2 | 34 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 47 |
| BFM #47 | Bit 0 | 18 |
| | Bit 1 | 19 |
| | Bit 2 | 50 |
| | ⋮ | ⋮ |
| | ⋮ | ⋮ |
| | Bit 15 | 63 |

4.20 Baud Rate <BFM #48> (Read Only)

This buffer memory is used for checking the baud rate.

This value is the baud rate. When 64DNET starts, the value set with DIP switches is stored in this buffer memory. For setting the baud rate, refer to subsection 5.1.2.



Note:

The baud rate cannot be changed by the DIP switches when the 64DNET is ON. Please turn 64DNET off to change the DIP switch positions.

4.21 Master/Slave Communication Data Area <BFM #100 ~ #133>

This area is used for I/O connection.

4.21.1 Send Data/Receive Data Area <BFM #100 ~ #131> (Read/Write)

These buffer memories are used to send/receive data to/from the master.

For note, see next page.

Table 4.15: Send Data/Receive Data Area <BFM #100 ~ #131>

| BFM No. | Description | | | |
|----------|----------------------------------|------------|---------------------------------------|------------|
| | Send Data (using TO Instruction) | | Receive Data (using FROM Instruction) | |
| | Upper Byte | Lower Byte | Upper Byte | Lower Byte |
| BFM #100 | 2nd byte | 1st byte | 2nd byte | 1st byte |
| BFM #101 | 4th byte | 3rd byte | 4th byte | 3rd byte |
| BFM #102 | 6th byte | 5th byte | 6th byte | 5th byte |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| BFM #131 | 64th byte | 63th byte | 64th byte | 63th byte |

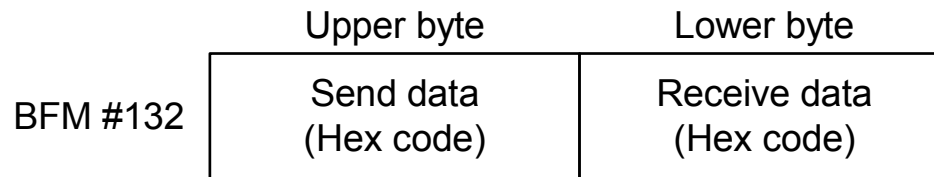
**Note:**

- When accessing the send data/receive data by TO/FROM instruction, write/read all data corresponding to the number of points set in Number of Link Points (BFM #132). The value set in Number of Link Points (BFM #132) is the amount of data that will be refreshed in the send or receive buffers.
If data less than the amount listed in BFM #132 is written/read the send or receive buffers will not be refreshed.
- The sending data and receiving data buffers have the same buffer memory addresses (BFM #100 ~ #131) for use with FROM and TO instructions. This means it is not possible to check the buffer memory data using a FROM instruction because this instruction only reads receive buffer data.
- Buffer Memories (BFM #100 ~ #131) are only refreshed for the number of bytes decided by BFM #132.
Therefore, when less data is received from the master compared to the previous time, some data remains from the prior message.

4.21.2 Number of Link Points <BFM #132> (Read/Write)

This buffer memory is used for setting number of link points of the I/O connection. The 64DNET can exchange maximum 64 bytes to the master. Default value is H0404 (Hex code). The setting range of the upper and lower bytes is H01 to H40.

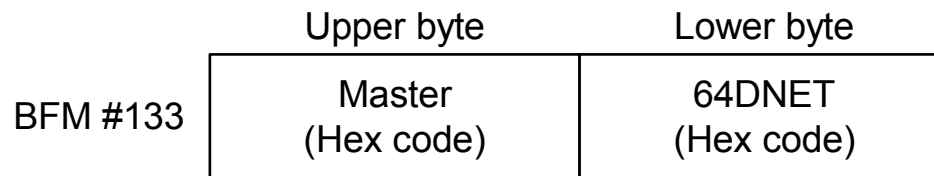
Figure 4.2: Number of Points <BFM #132>



4.21.3 MAC ID <BFM #133> (Read Only)

This buffer memory is used for checking the MAC ID of master and 64DNET I/O connection. The upper byte is the MAC ID of the master. The lower byte is the MAC ID of the 64DNET. The MAC ID of 64DNET is stored to the lower byte during start up.

Figure 4.3: MAC ID <BFM #133>



4.22 UCMM Client Communication Data Area <BFM #10000 ~ #16336>

This area is used for the UCMM connection when 64DNET is a client.

4.22.1 Send Data/Receive Data Area <BFM #1□□00^{*1} ~ #1□□31^{*1}> (Read/Write)

These buffer memories are used to send/receive data to/from the server.
For note, see next page.

Table 4.16: Send Data/Receive Data Area <BFM #1□□00^{*1} ~ #1□□31^{*1}>

| BFM No. | Description | | | |
|--------------------------|----------------------------------|------------|---------------------------------------|------------|
| | Send Data (using TO Instruction) | | Receive Data (using FROM Instruction) | |
| | Upper Byte | Lower Byte | Upper Byte | Lower Byte |
| BFM #1□□00 ^{*1} | 2nd byte | 1st byte | 2nd byte | 1st byte |
| BFM #1□□01 ^{*1} | 4th byte | 3rd byte | 4th byte | 3rd byte |
| BFM #1□□02 ^{*1} | 6th byte | 5th byte | 6th byte | 5th byte |
| ⋮ | ⋮ | ⋮ | ⋮ | ⋮ |
| BFM #1□□31 ^{*1} | 64th byte | 63th byte | 64th byte | 63th byte |

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

**Note:**

- The 64DNET can use UCMM connection when the connection target is another 64DNET.
- When accessing the send data by TO instruction, write all data corresponding to the number of points set in Length of Assembly Data (BFM #1□□32^{*1}). The value set in Length of Assembly Data is the amount of data that will be refreshed in the send buffers. If a data amount less than the amount listed in BFM #1□□32^{*1} is written the send buffers will not be refreshed.
- When accessing the receive data by FROM instruction, read all data corresponding to the number of points set in Length of Assembly Data (BFM #1□□34^{*1}). The value set in Length of Assembly Data is the amount of data that will be refreshed in the receive buffers. If a data amount less than the amount listed in BFM #1□□34^{*1} is read the receive buffers will not be refreshed.
- The send data and receive data buffers have the same buffer memory addresses (BFM #1□□00^{*1} ~ #1□□31^{*}) for use with FROM and TO instructions. This means it is not possible to check the buffer memory data using a FROM instruction because this instruction only reads receive buffer data.
- Buffer Memories (BFM #1□□00^{*1} ~ #1□□31^{*}) are only refreshed for the number of bytes decided by BFM #1□□32. Therefore, when less data is received from the master compared to the previous time, some data remains from the prior message.

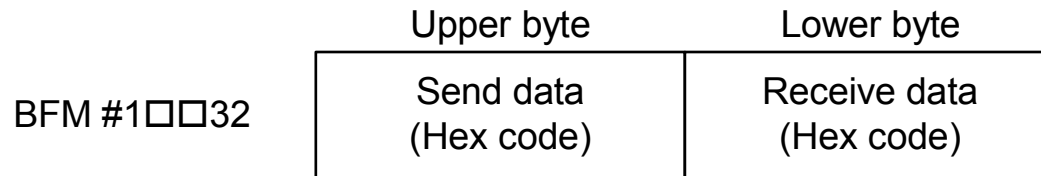
*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

- When 64DNET client accesses the 64DNET sever using GET service on UCMM connection, client will get the following data from server.
 - When client accesses server using GET service after SET service, client will get the last set data SET service.
 - When client accesses server using GET service before any SET service, client will get the default data (“K0”).

4.22.2 Length of Assembly Data <BFM #1□□32*1> (Read/Write)

These buffer memories are used for setting length of assembly data about the connected to the server. The 64DNET can exchange maximum 64 bytes to the server. Default value is H0404 (Hex code). The setting range of the upper and lower bytes is H01 to H40.

Figure 4.4: Length of Assembly Data <BFM #1□□32*1>



*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.22.3 Length of Sending Common Service Data <BFM #1□□33*1> (Read/Write)

These buffer memories are used for setting length of sending common service data to the server. Default value is K0.

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.22.4 Length of Receiving Data <BFM #1□□34*1> (Read Only)

These buffer memories are used for checking the length of data received from the server. This value is given in number of bytes received.

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.22.5 Group Selection <BFM #1□□35*1> (Read/Write)

These buffer memories are used for setting the connection group. The default value is K3.

Table 4.17: Group Setting <BFM #1□□35*1>

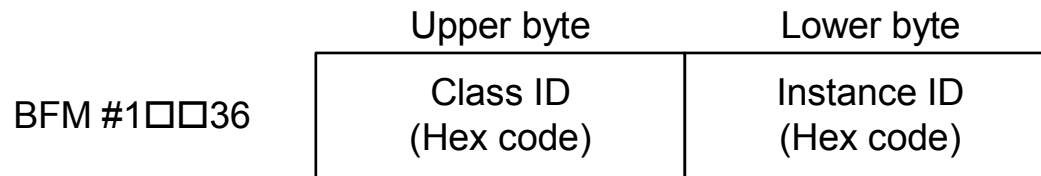
| Setting Value | Reading Value | Description |
|---------------|---------------|-------------|
| 0 | Not used | |
| 1 | 0 | Group 1 |
| 2 | Not used | |
| 3 | 3 | Group 3 |
| 4 or more | Not used | |

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.22.6 Class ID and Instance ID <BFM #1□□36*1>

These buffer memories are used for setting the Class ID and the Instance ID. For Class ID and Instance ID, refer to Appendix B.

Figure 4.5: Class ID and Instance ID <BFM #1□□36*1>

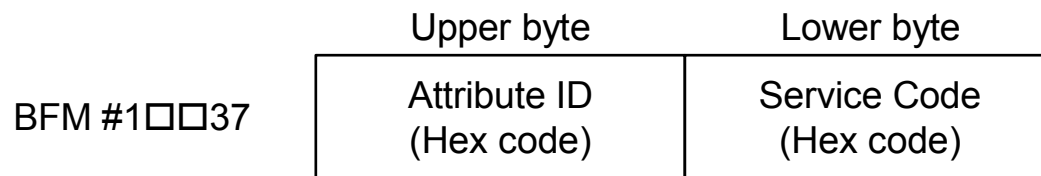


*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.22.7 Attribute ID and Service Code <BFM #1□□37*1>

These buffer memories are used for setting the Class ID and the Instance ID. For Class ID and Instance ID, refer to Appendix B.

Figure 4.6: Attribute ID and Service Code <BFM #1□□37*1>



*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.23 UCMM Server Communication Data Area <BFM #20000 ~ #26336>

This area is used for the UCMM connection when 64DNET is a server.

4.23.1 Receive Data Area <BFM #2□□00^{*1} ~ #2□□31^{*1}> (Read Only)

These buffer memories are used to receive data from the client.
For note, see next page.

Table 4.18: Receive Data Area <BFM #2□□00^{*1} ~ #2□□31^{*1}>

| BFM No. | Description | |
|--------------------------|-------------|------------|
| | Upper Byte | Lower Byte |
| BFM #2□□00 ^{*1} | 2nd byte | 1st byte |
| BFM #2□□01 ^{*1} | 4th byte | 3rd byte |
| BFM #2□□02 ^{*1} | 6th byte | 5th byte |
| ⋮ | ⋮ | ⋮ |
| BFM #2□□31 ^{*1} | 64th byte | 63th byte |

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the client.

**Note:**

- The 64DNET can use UCMM connection when the connection target is another 64DNET.
- When accessing the receive data by FROM instruction, read all data corresponding to the number of points set in Length of Assembly Data (BFM #2□□34^{*1}). The value set in Length of Assembly Data is the amount of data that will be refreshed in the send or receive buffers.
If a data amount less than the amount listed in BFM #2□□34^{*1} is read the receive buffers will not be refreshed.
- Do not write to (access by TO instruction) the “Read only” section of the buffer memory (BFM #2□□00^{*1} ~ #2□□31^{*1}, #2□□33^{*1} ~ #2□□37^{*1}) in the programmable controller. It is not possible to operate the 64DNET by writing to (accessing by TO instruction) these buffer memories.
- Buffer Memories (BFM #2□□00^{*1} ~ #2□□31^{*}) are only refreshed for the number of bytes decided by BFM #2□□32.
Therefore, when less data is received from the master compared to the previous time, some data remains from the prior message.

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the client.

4.23.2 Length of Assembly Data <BFM #2□□32*1> (Read/Write)

These buffer memories are used for setting the length of assembly data from the client. The 64DNET can exchange maximum 64 bytes to the client. Default value is H0404 (Hex code). The setting range of the upper and lower bytes is H01 to H40.

Figure 4.7: Length of Assembly Data <BFM #2□□32*1>

| | Upper byte | Lower byte |
|------------|-------------------------------|----------------------------|
| BFM #2□□32 | Not Used (Must be 00 hex.) | Receive data (Hex code) |

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the client.

4.23.3 Length of Sending Common Service Data <BFM #2□□33*1> (Read Only)

These buffer memories are used for setting the length of the sending common service data to the server from client. The default value is K0.

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the client

4.23.4 Length of Receiving Data <BFM #2□□34*1> (Read Only)

These buffer memories are used for checking the length of the data received from the client. This value is given in number of bytes received.

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.23.5 Group Select <BFM #2□□35*1> (Read Only)

These buffer memories are used for checking the connection group. The default value is K3.

Table 4.19: Group Setting <BFM #2□□35*1>

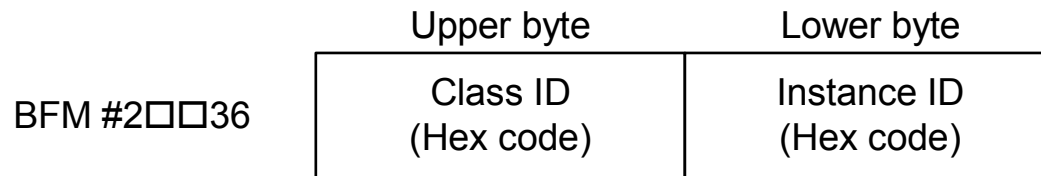
| Value | Description |
|-----------|-------------|
| 0 | Group 1 |
| 1, 2 | Not used |
| 3 | Group 3 |
| 4 or more | Not used |

*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.23.6 Class ID and Instance ID <BFM #2□□36*1> (Read only)

These buffer memories are used for checking the Class ID and the Instance ID. For Class ID and Instance ID, refer to Appendix B.

Figure 4.8: Class ID and Instance ID <BFM #2□□36*1>

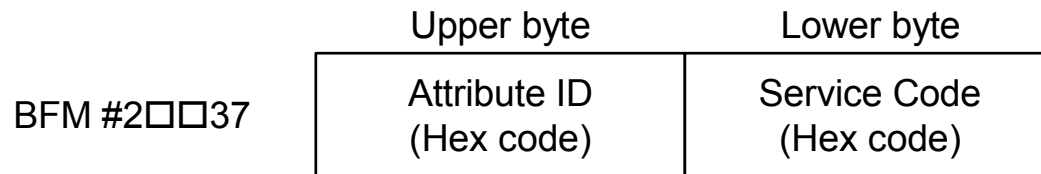


*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

4.23.7 Attribute ID and Service Code <BFM #2□□37*1> (Read only)

These buffer memories are used for checking the Attribute ID and the Service Code. For the Attribute ID and the Service Code, refer to Appendix B.

Figure 4.9: Attribute ID and Service Code <BFM #2□□37*1>

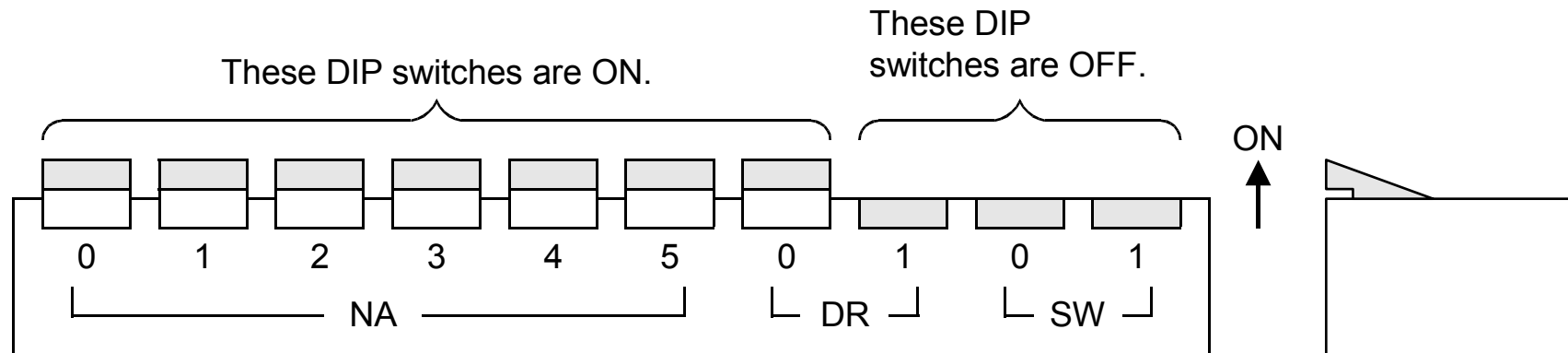


*1 □□ has a value of 0 ~ 63. This value is the MAC ID of the server.

5. DIP Switch Setting

5.1 Setting DIP Switch

Figure 5.1: Allocation DIP Switch



5.1.1 Mode Setting (DIP Switch: SW0, SW1)

Operation mode is decided by the DIP switches SW0 and SW1.

Default setting: Normal operation mode (SW0 =OFF, SW1 = OFF)

Table 5.1: Mode Setting (DIP Switches: SW0, SW1)

| SW1 | SW0 | Using Mode |
|-----|-----|------------------|
| OFF | OFF | Normal operation |
| OFF | ON | Not used |
| ON | OFF | |
| ON | ON | |

5.1.2 Baud Rate Setting (DIP Switch: DR0, DR1)

Operation mode is decided by the DIP switches DR0 and DR1.

Default setting: 500 kbps (DR0 = OFF, DR1 = ON)

Table 5.2: Baud Rate Setting (DIP Switches: DR0, DR1)

| DR1 | DR0 | Setting Baud Rate |
|-----|-----|-------------------|
| OFF | OFF | 125 kbps |
| OFF | ON | 250 kbps |
| ON | OFF | 500 kbps |
| ON | ON | Not used |

5.1.3 MAC ID Setting (DIP Switches: NA0 ~ NA5)

The MAC ID is decided by the sum total value of the DIP switches NA0 ~ NA5. The MAC ID setting value range is 0 ~ 63. Default setting: 63 (NA0 ~ NA5 are all ON)

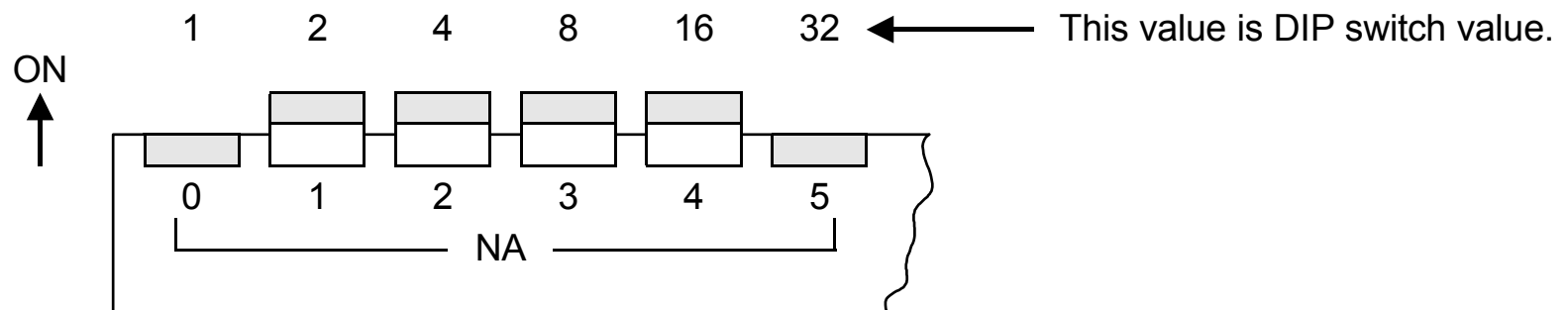
Table 5.3: DIP Switch Value

| DIP Switch | Value | |
|------------|-------|-----|
| | ON | OFF |
| NA0 | 1 | 0 |
| NA1 | 2 | |
| NA2 | 4 | |
| NA3 | 8 | |
| NA4 | 16 | |
| NA5 | 32 | |

Example Setting

If the MAC ID of the 64 DNET is set to “30”, the DIP switches are as shown below.

Figure 5.2: Example Setting of MAC ID



6. Client/Sever Communication by UCMM Connection

6.1 Information of UCMM Connection by 64DNET

The UCMM connection is client/server communication. The 64DNET can use UCMM connection when the connection target is another 64DNET. The direction of connection is from client to server. The 64DNET becomes a client of maximum of 62 servers (64DNETs), and a server for a maximum of 62 clients (64DNETs). Between two 64DNET, UCMM connection of both directions client → server and server ← client can be performed.

For supported objects, refer to Appendix B.



Note: Supported UCMM Connection by 64DNET

When the 64DNET is a client, it can send data or request data from the server. If the 64DNET sends a data request to the server, it will receive data from the server. When the 64DNET is the server, it can receive and request data from client. If the 64DNET receives a request from a client, it will send an answer data to the client.

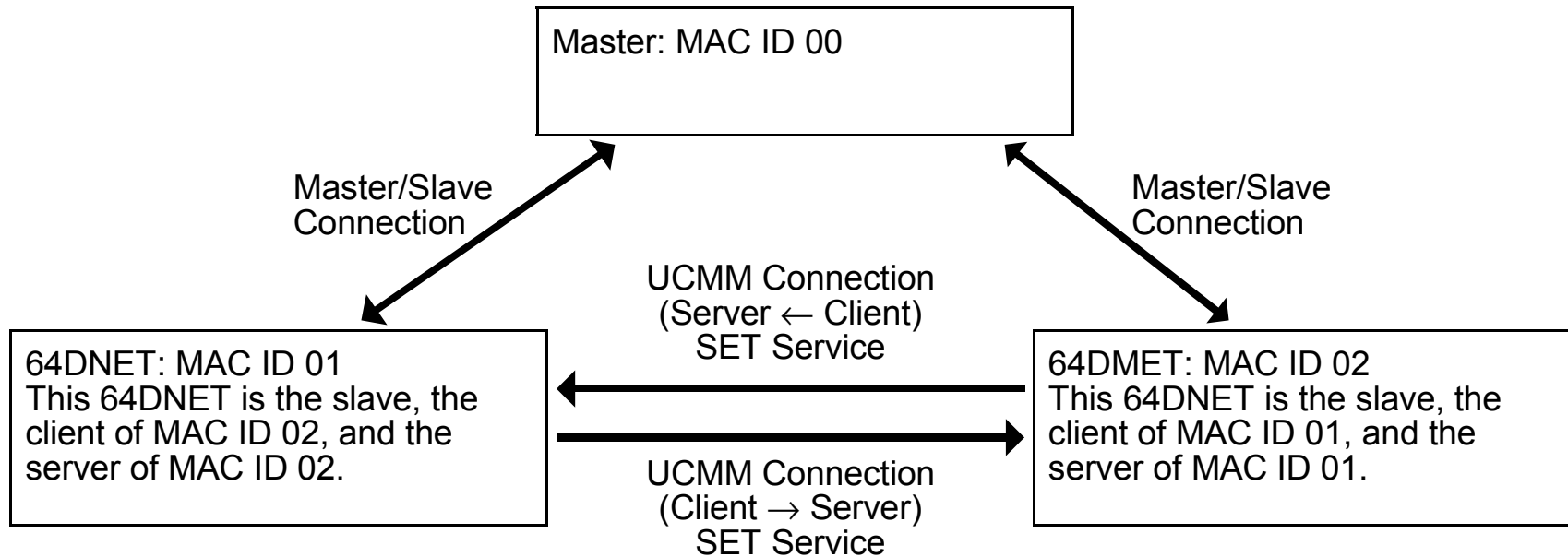
Table 6.1: UCMM Connection Flags and Status

| Flags | Client | Server |
|------------------------|--------|--------|
| Connection status | ✓ | |
| Sending trigger flag | ✓ | |
| Receiving status | ✓ | ✓ |
| UCMM client WDT switch | ✓ | |
| Communication status | ✓ | ✓ |

6.2 How to Use UCMM Communication

When using UCMM communication, the connection for each node is shown below.
 For note, see next page.

Figure 6.1: UCMM Communication



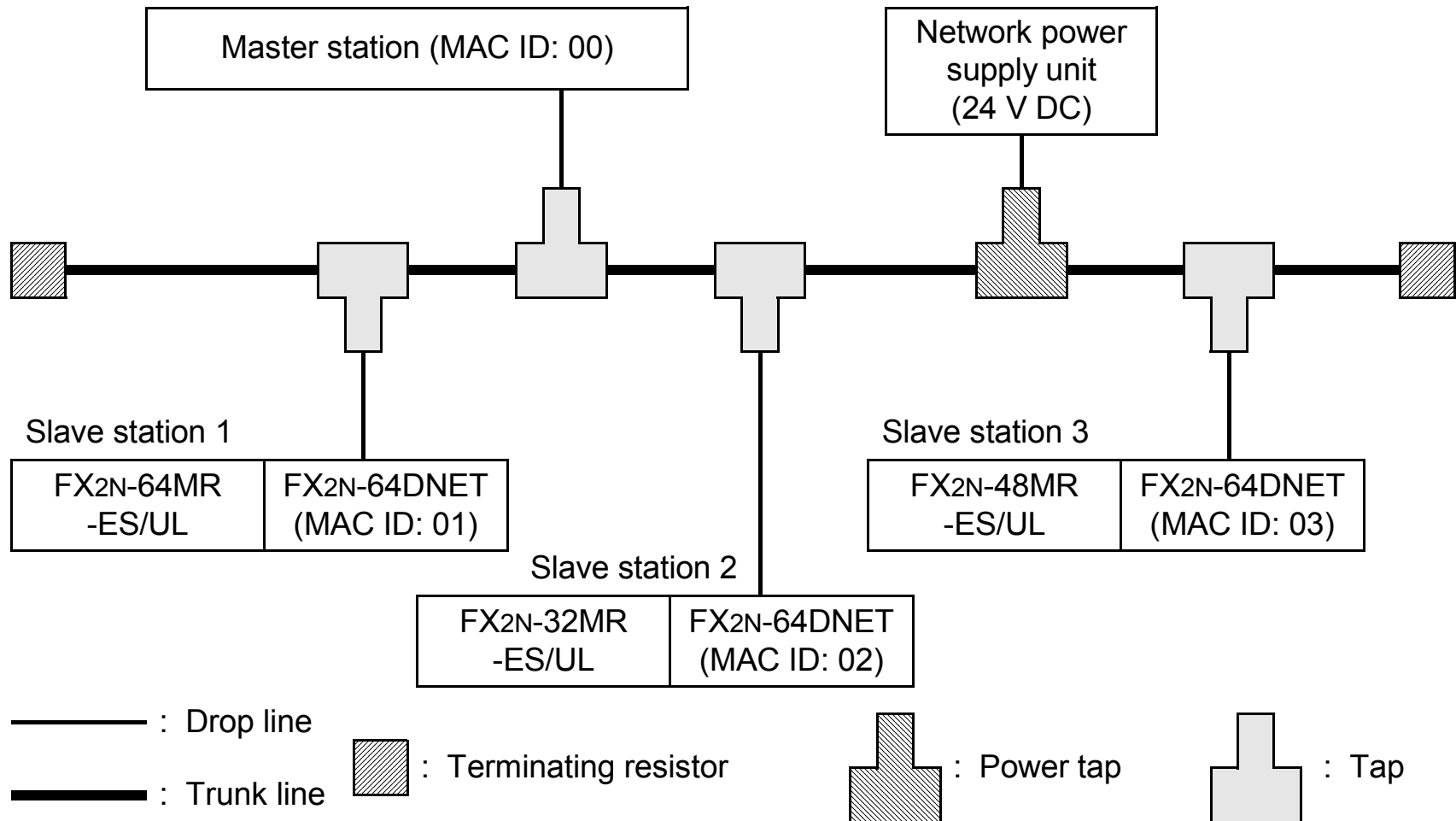
**Note:**

- The 64DNET can use UCMM connection when the connection target is another 64DNET. The 64DNET must know the object information (Classes ID, Instance ID, and Attribute ID and service code) of the target 64DNET.
For the object information of 64DNET, refer to Appendix B.
- When using UCMM connection, please choose group 3.
- When 64DNET client accesses the 64DNET server using GET service on UCMM connection, client will get the following data from server.
 - When client accesses server using GET service after SET service, client will get the last set data SET service.
 - When client accesses server using SET service before any SET service, client will get the default data (“K0”).

7. Example Program

7.1 System Configuration

Figure 7.1: System Configuration for Example Program



7.2 Operation

This operation is written for Slave Station 1.

1) Master/Slave I/O connection:

- Slave station 1 exchanges 64/64 bytes data to the master station (MAC ID 00) by the Master/Slave I/O connection.
- Received data from the master station is stored in D150 ~ D181 in slave station 1.
- D100 ~ D131 in slave station 1 are sent to the master station.

2) The UCMM connection when slave station 1 (MAC ID 01) is the client:

- Slave station 1 exchange 64/64 bytes data to slave station 2 (server, MAC ID 02) by the UCMM connection.
- Received data from the server is stored in D500 ~ D531 in slave station 1.
- When X000 is turned ON, slave station 1 connects to slave station 2 by the UCMM connection.

3) The UCMM connection when the slave station 1 (MAC ID 01) is the server:

- The slave station 1 can receive 64 bytes data from slave station 3 (client, MAC ID 03) by the UCMM connection.
- Data received data from slave station 3 is stored in D600 ~ D631 in slave station 1.

7.3 Example Program

This example program is written about slave station 1.



Note:

When many FROM/TO instructions are executed in the same scan, the PLC might have a watchdog timer error. In this case, add a watchdog timer reset (FNC07 WDT) instruction with each FROM/TO instruction that sends and receives data.

7.3.1 Initial Setting

Figure 7.2: Initial Setting

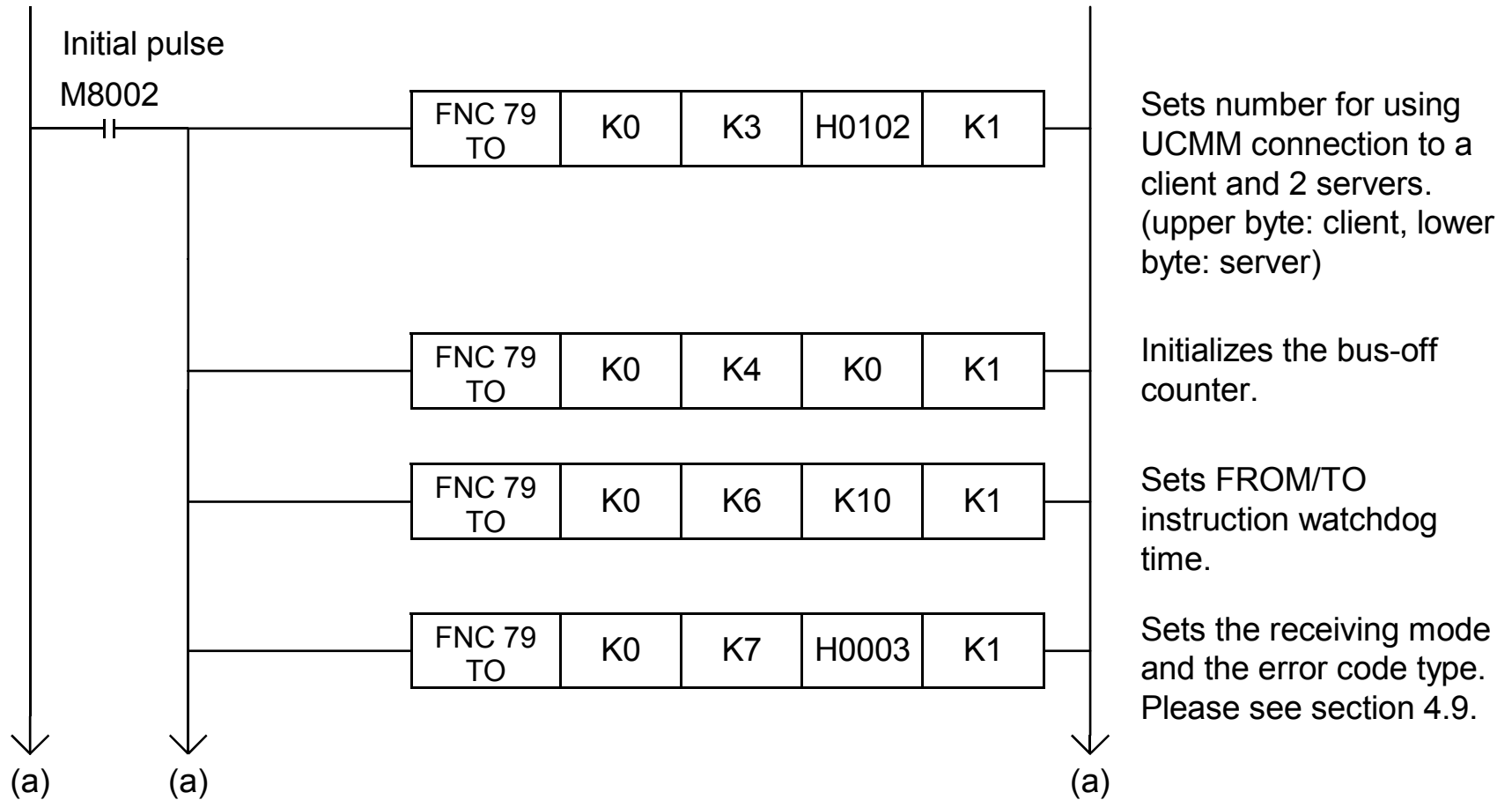


Figure 7.2: Initial Setting

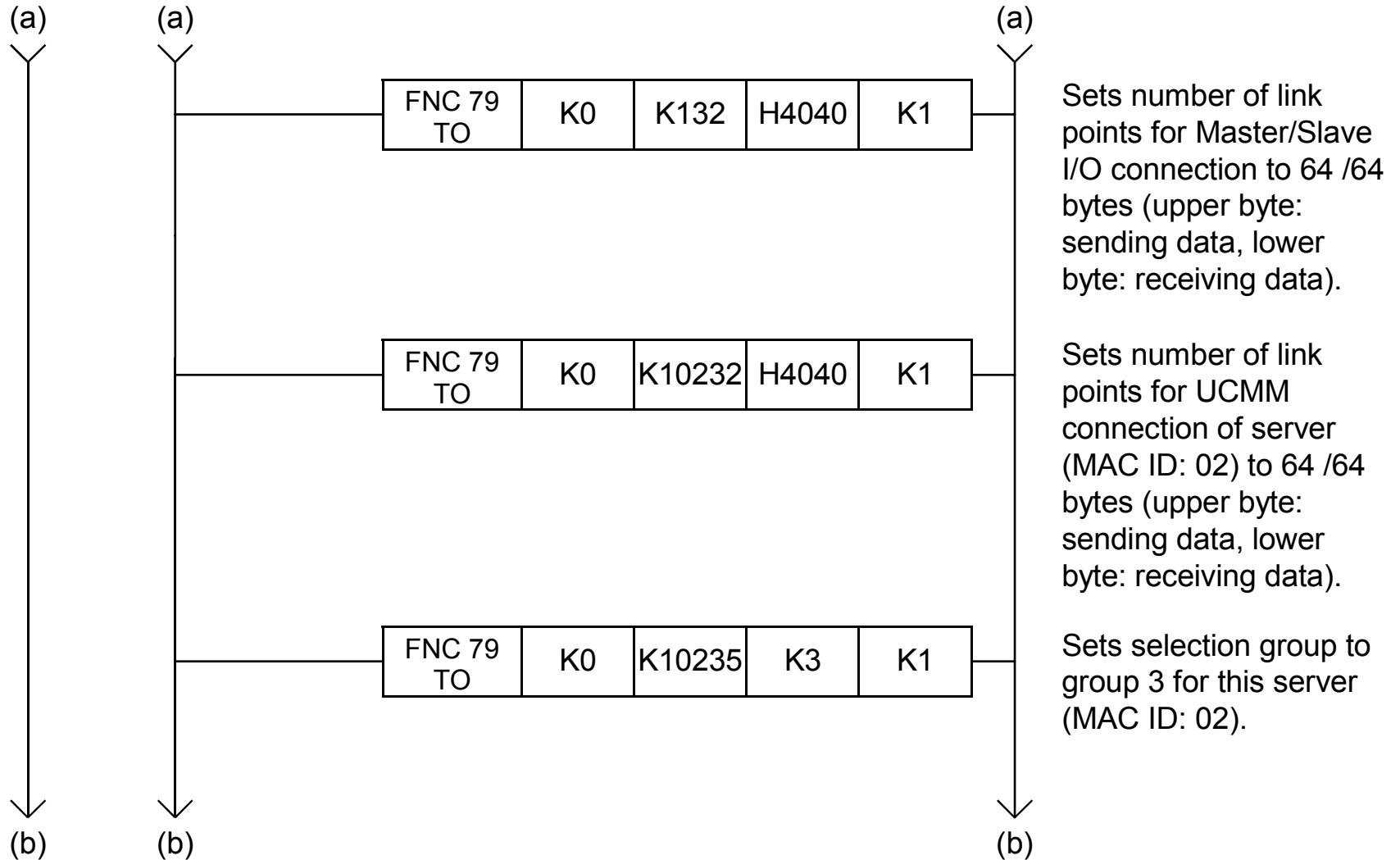


Figure 7.2: Initial Setting

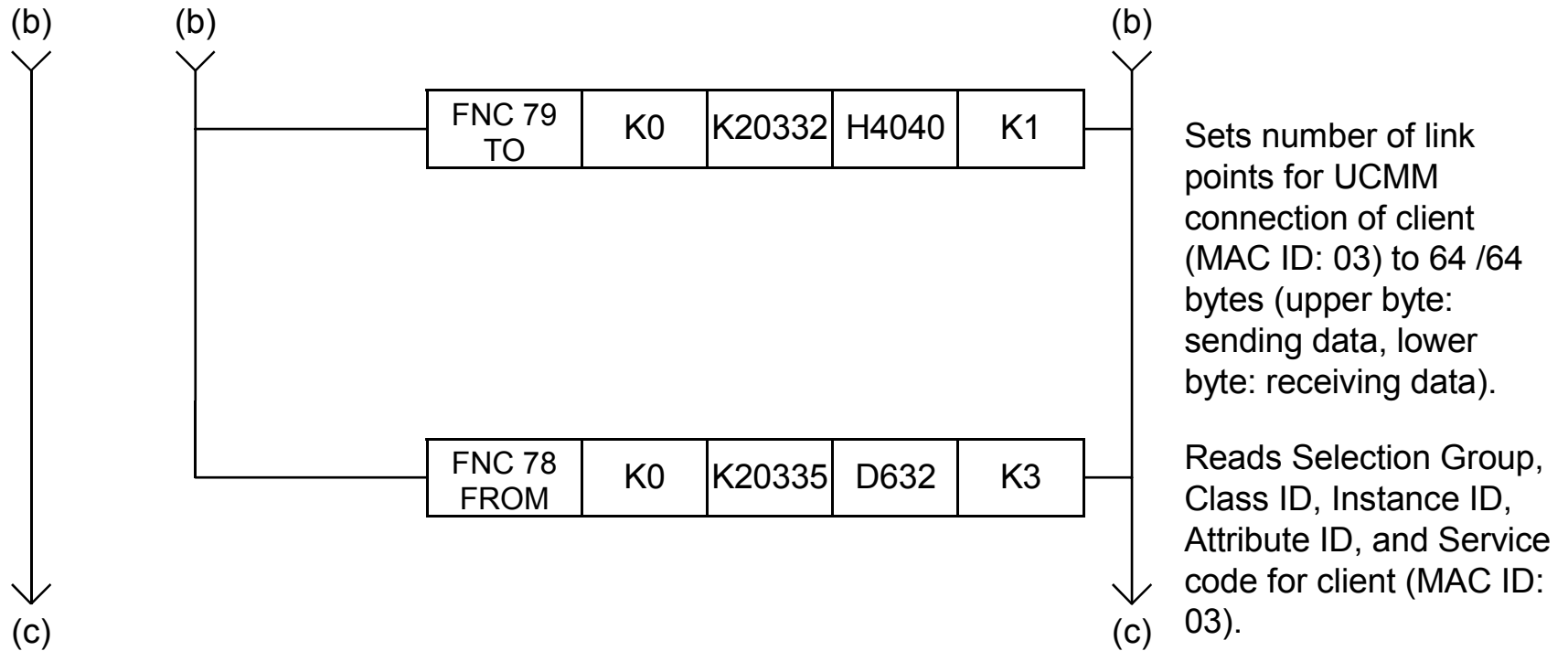
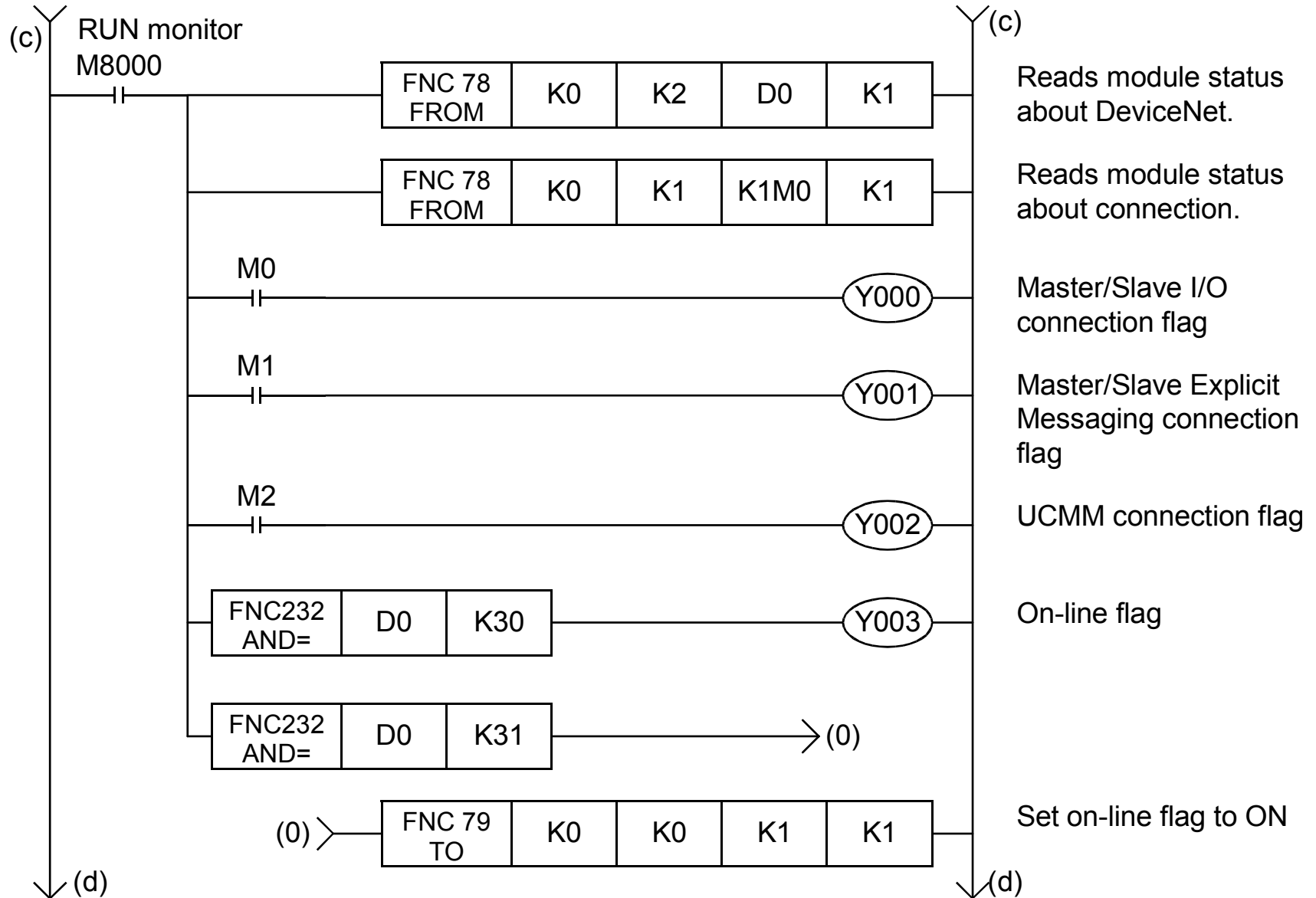
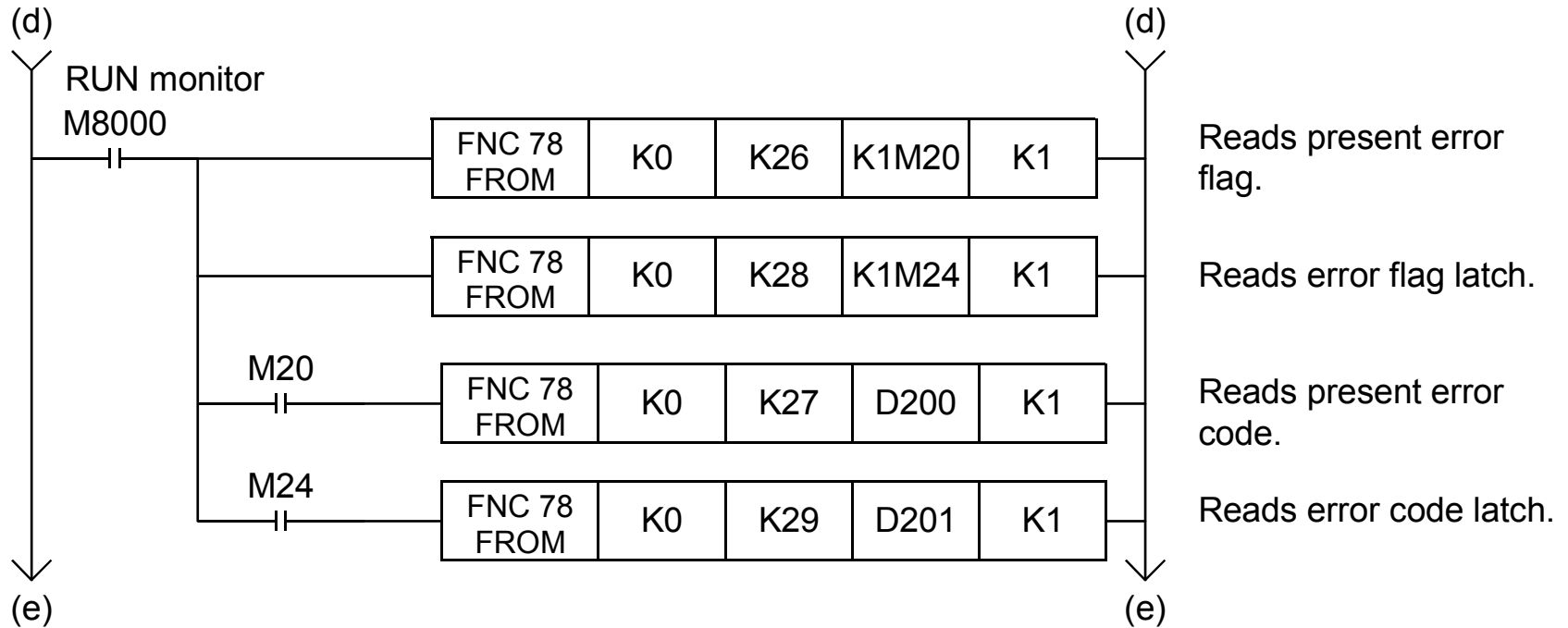


Figure 7.2: Initial Setting



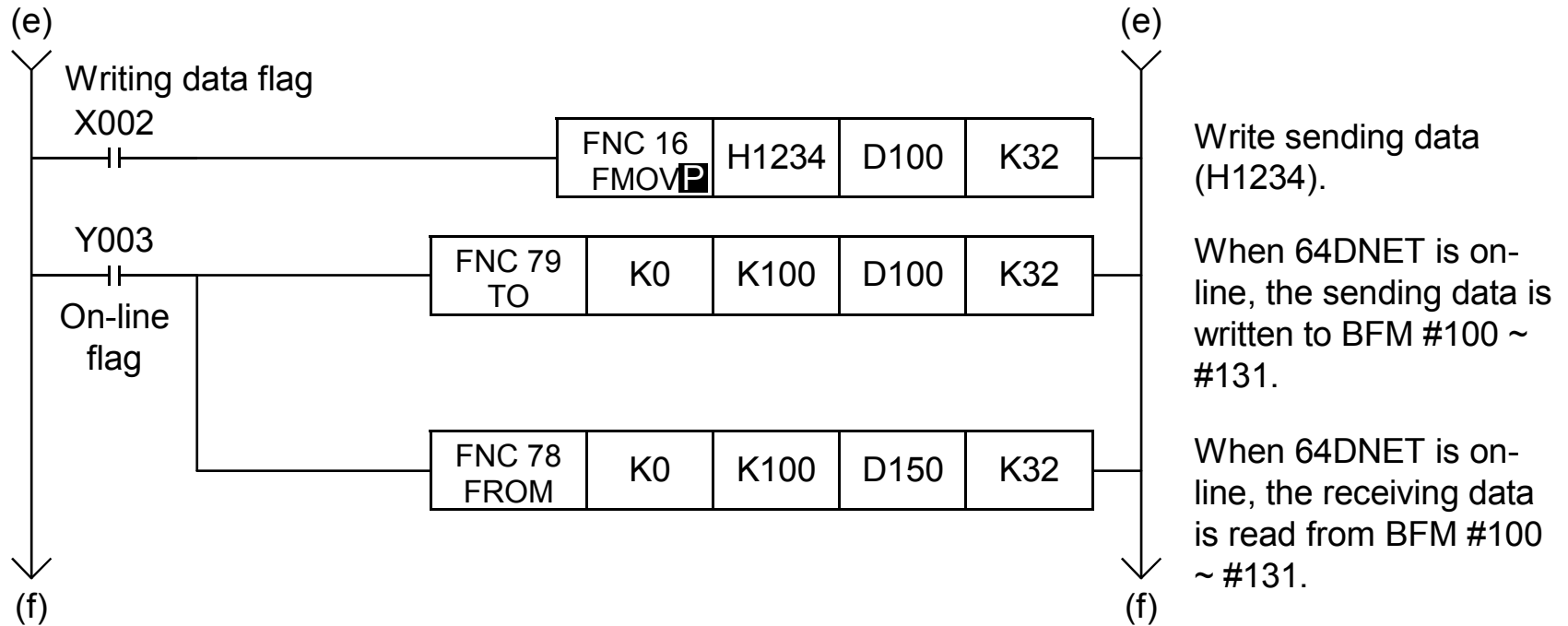
7.3.2 Checking Error Status

Figure 7.3: Checking Error Status



7.3.3 Input Data and Output Data for Master/Slave I/O Connection

Figure 7.4: Input Data and Output Data for Master/Slave I/O Connection



7.3.4 Input Data and Output Data for UCMM Connection

1) Using client

Figure 7.5: UCMM Connection (Client)

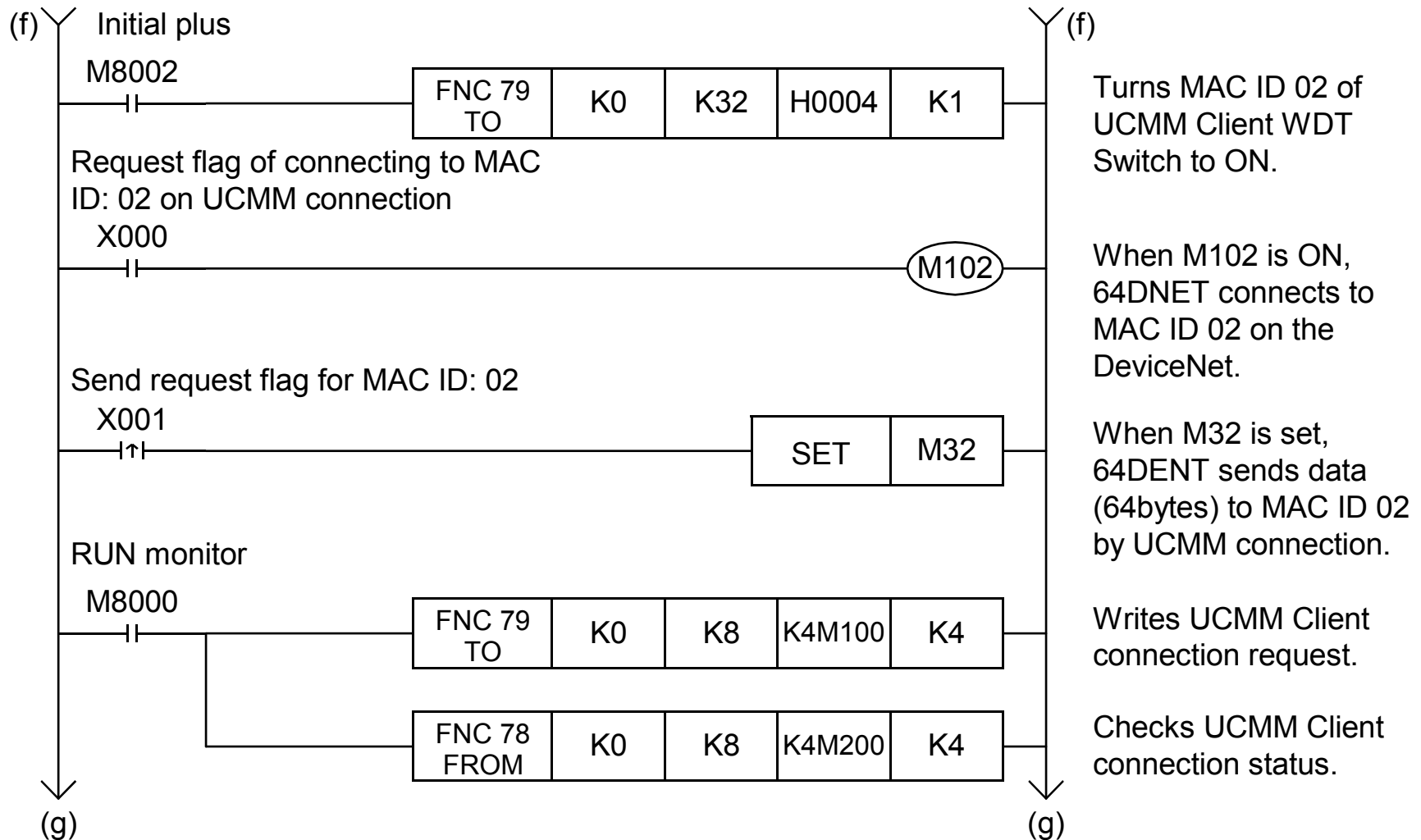


Figure 7.5: UCMM Connection (Client)

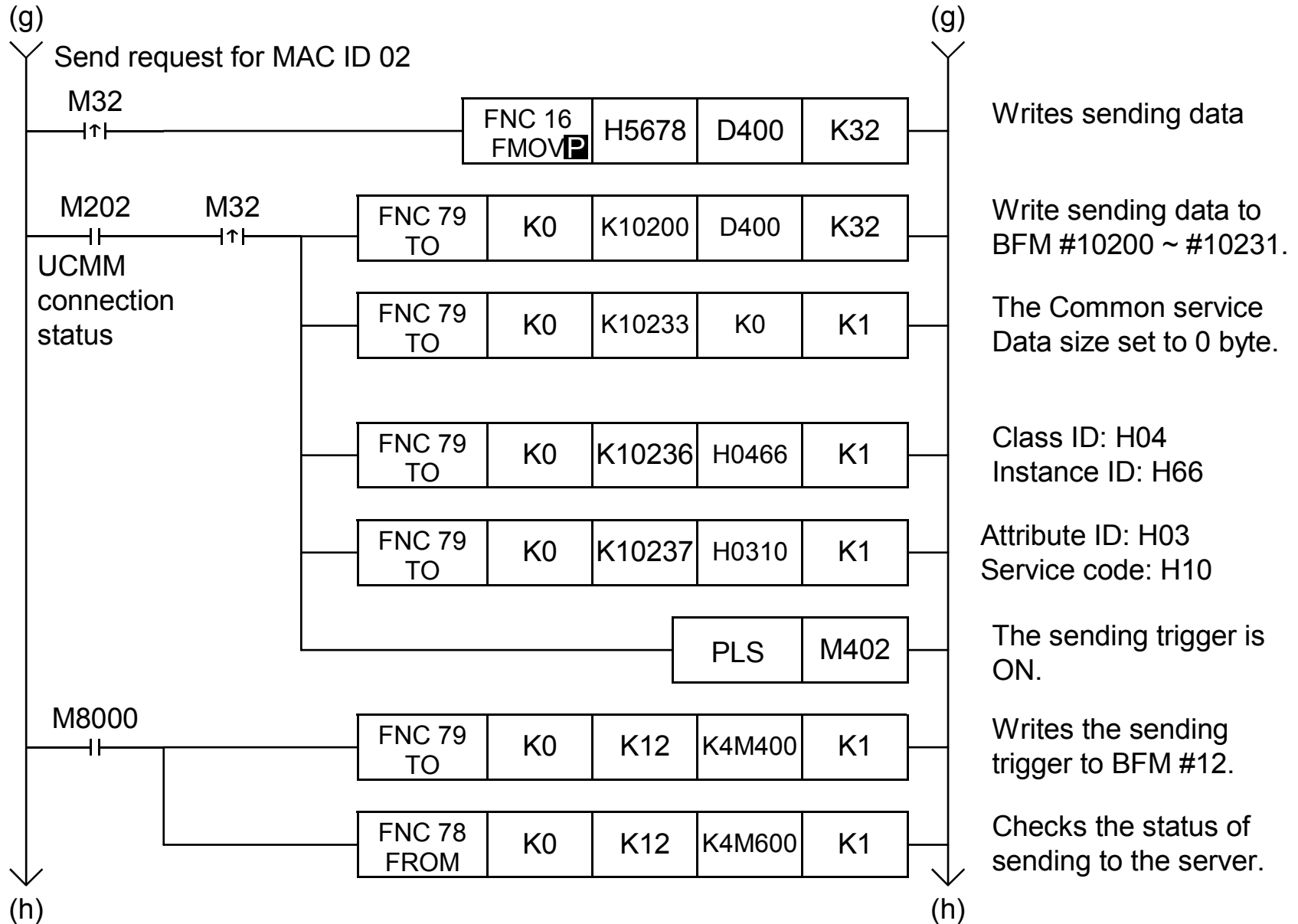
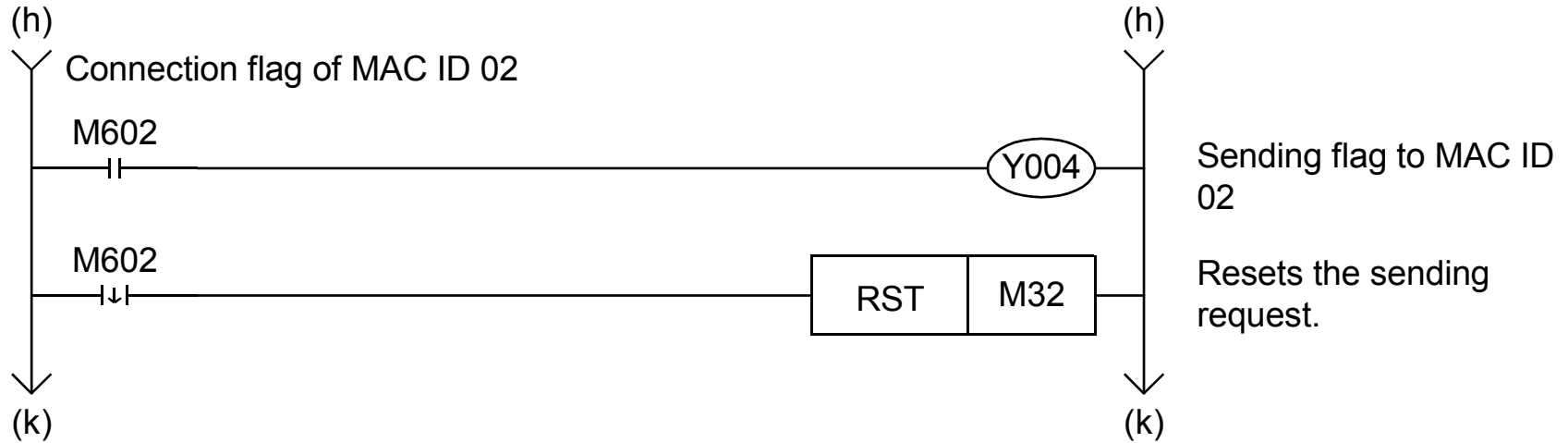


Figure 7.5: UCMM Connection (Client)



2) Using server

Figure 7.5: UCMM Connection (Client)

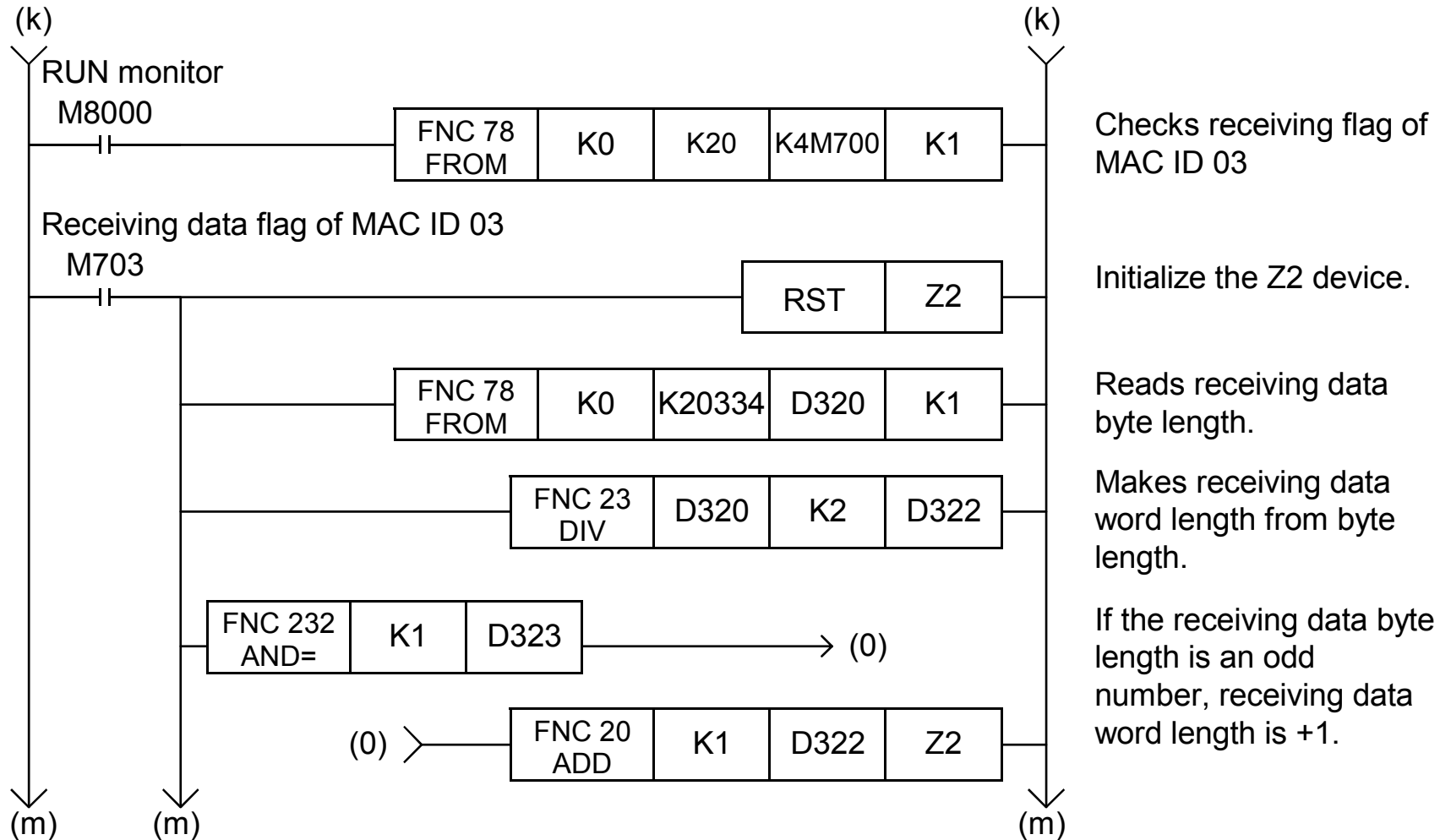
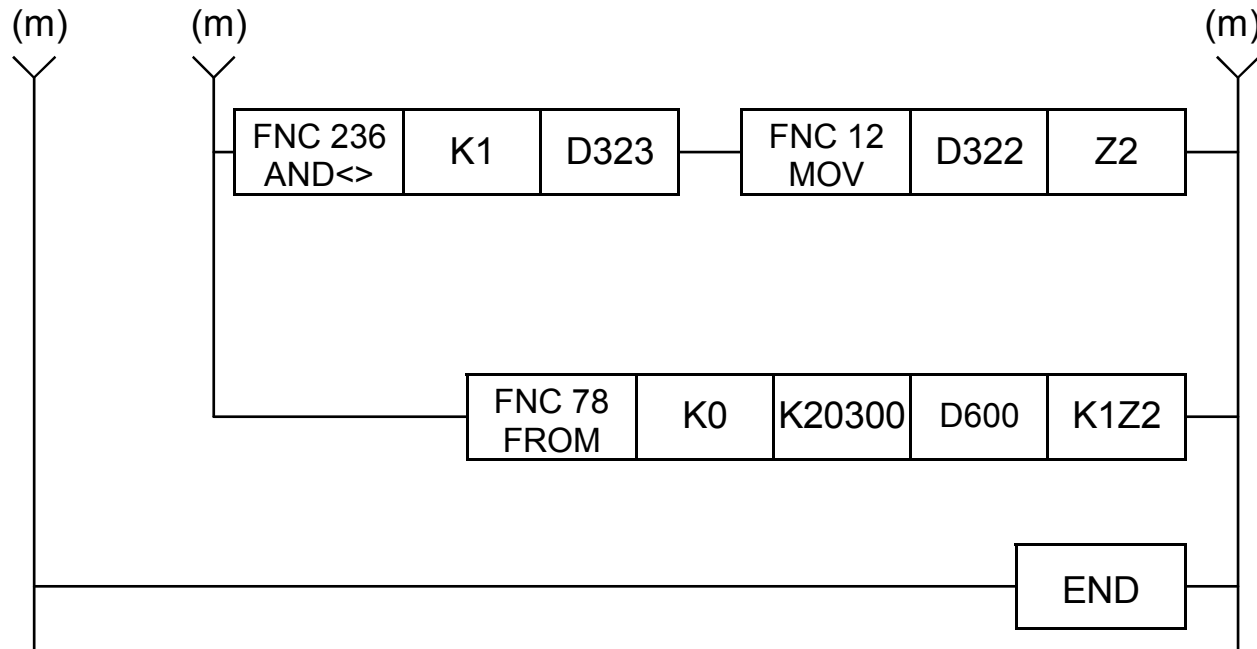


Figure 7.6: UCMM Connection (Server)



If the receiving data byte length is an even number, receiving data word length is the same as byte length.

Reads the receiving data from BFM #20300 ~ #20331 area.

8. Diagnostics

8.1 Preliminary Checks

- 1) Check "POWER LED".
If "POWER LED" is OFF, check the connection of the 64DNET extension cable to PLC, and check whether the 5V DC voltage source capacity is sufficient.
- 2) Check "MS LED" and "NS LED".
If both LEDs are not lit green, check the wiring for the DeviceNet, the module status in BFM #2, the setting of the Master and the error code in BFM #27, #29.
For module status, refer to section 4.4. For error code, subsection 7.2.2. For wiring, refer to chapter 2 and manual of the Master. For the setting of the Master, refer to manual of the Master.
- 3) Check whether 24 V DC voltage source capacity of DeviceNet suffices.
Supply current and voltage.
- 4) Check baud rate.
If the baud rate set in 64DNET is not the same as other station, change to matching values.
When changing setting value, change is valid after PLC power ON.
For the baud rate of 64DNET, see BFM #48 in 64DNET.
- 5) Check the MAC ID of 64DNET.
Please do not use the same MAC ID for more than one node on the DeviceNet.

8.2 LED Check

Check the status of the LED's for the 64DNET as follows.

1) POWER LED check

Table 8.1: POWER LED Check

| Status | Description |
|-----------|---|
| Lit | The extension cable is properly connected. And 5 V DC is properly supplied from PLC. |
| Otherwise | Check the connection of the 64DNET extension cable to the PLC, and check whether the 5V DC voltage source capacity of PLC suffices. |

2) FROM/TO LED check

Table 8.2: FROM/TO Check

| Status | Description |
|-----------|---|
| Lit. | BFM is accessed by FROM/TO instruction. |
| Otherwise | BFM is not accessed by FROM/TO instruction. |

3) MS LED check

Table 8.3: MS LED (Module Status LED) Check

| MS LED Status | 64DNET Status | Description |
|---|---------------------|---|
| Not lit | Power OFF | Check whether 24 V DC voltage source has sufficient ampere capacity. Provide sufficient current. |
| Lit Green | Normal Operation | The 64DNET is in normal operation mode. |
| Flashing Green at 0.5 s intervals | Stand-by | The 64DNET is in stand-by status. |
| Flashing Red at 0.5 s intervals | Minor Fault | A recoverable error has occurred. |
| Lit Red | Unrecoverable Fault | An unrecoverable error has occurred. Please contact a service representative. |
| Flashing Green and Red at 0.5 s intervals | Self Test | The 64DNET is in self test mode. |

4) NS LED

Table 8.4: NS LED (Network Status LED) Check

| MS LED Status | 64DNET Status | Description |
|--|-----------------------------------|--|
| Not lit | Not On-line | The 64DNET is not on-line. <ul style="list-style-type: none"> The power supply voltage is incorrect for DeviceNet. When the value in BFM #0 is K0, 64DNET is off-line. In this case, write K1 to BFM #0. At this time, 64DNET will be on-line. |
| Lit Green | Link OK | The 64DNET is on-line and has established the connection. |
| Flashing Green at 0.5 s intervals | On-line Status Without Connection | The 64DNET is on-line but has not established the connection. Check the setting of the Master, and check setting of the UCMM connection in 64DNET. |
| Flashing Red at 0.5 s intervals | Communication Time-out | The 64DNET I/O connection is in time-out status. Check the setting of the Master. |
| Lit Red | Critical link Failure | Communication is not possible. Check wiring, and check the MAC ID of 64DNET to be used by other node. If the same, change the MAC ID of 64DNET or other node. |
| Flashing Green and Red at 0.25 s intervals | Communication Fault | The 64DNET is in the Communication Faulted status for detecting the Network Accessing Error. After this module will receive the Identify Communication Faulted Request. Check error code. For error code, refer to subsection 7.2.2. |

8.3 Error Code

Error code of 64DNET is 64DNET error code and DeviceNet error code.



Note:

If error code is for UCMM connection, check the UCMM client connection status (BFM #36 ~ #39), UCMM client communication status (BFM #40 ~ #43) and UCMM server communication status (BFM #44 ~ #47). The source of the error must be checked in the individual device buffer memories.

8.3.1 64DNET Error Code (BFM #7 Bit 1 is ON)

This error code is service type (upper byte) and detailed error code (lower byte).

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|---|---|
| 01 | | Critical fault connection time-out | |
| | 01 | The first occurrence of a duplicate MAC ID was received. | Change the MAC ID setting using the DIP switch with respect to the other devices in the network as to not repeat previously used numbers. |
| | 02 | The second occurrence of duplicate MAC ID was received. | |
| | 03 | DR1 and DR2 of DIP switch are set to 1 (This setting is the "Not used" position). | Set the baud rate correctly. |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|---|---|
| 01 | 04 | The receive buffer for the CAN interrupt overflowed. | Reduce the number of data of sending and receiving, reduce the communication frequency or lower the baud rate. |
| | 05 | CAN Controller is in the bus-off state. | Check system configuration on the DeviceNet. |
| | 06 | The Explicit message connection time-out occurred. | Allocate the Explicit Message Connection from the Master or the client again. |
| | 11 | Duplicate MAC ID response message was received. | Change the MAC ID setting using the DIP switch with respect to the other devices in the network as to not repeat previously used numbers. |
| | 12 | Duplicate MAC ID request was received, the MAC ID is in transmission time-out waiting status. | |
| | 13 | The Polling connection time-out occurred. | Allocate the Master/Slave I/O Connection from the Master again. |
| 14 | The COS/Cyclic connection time-out occurred. | | |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|---|--|
| 02 | Allocate_Master/Slave_Connection_Set service error (An error with the connection the Allocate request) | | |
| | 02 | The Allocate request was received from a different Master. | Check the settings of the Master |
| | 06 | 64DNET cannot support connections other than the Explicit message, Polling, and COS/Cyclic in the Allocate request. | |
| | 07 | The Allocate request bit is not properly set. | Check the setting of the Master to order the Allocate request. |
| | 08 | The specified connection has already been allocated. | Allocate only once for each connection. |
| | 09 | The Reserve Bit of allocator MAC ID does not turn OFF. | Check packet data for the Allocate request from the Master. |
| | 10 | Allocation Choice Bit is not specified. | Check the setting of Master to order the Allocate request. |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|---|--|---|
| 02 | 11 | The Ack Suppress Bit is specified but the COS/Cyclic bit is not specified. | Check the setting of Master to order the Allocate request. |
| | 12 | Tries to establish both COS and Cyclic transmission method for Master/Slave connection. | |
| | 13 | The Polling connection cannot be established because the Instance ID has already been used by the COS/Cyclic connection. | |
| 03 | Release_Master/Slave_Connection_Set service error (An error with the open connection request) | | |
| | 06 | 64DNET cannot support connections other than the Explicit message, Polling, and COS/Cyclic in the Allocate request. | Connections other than the Explicit message, Polling, and COS/Cyclic are not supported. |
| | 07 | The connection for the Release request is done is not specified. | Specify the connection for the Release request. |
| | 08 | The specified connection has already been released. | Use only one Release request per connection. |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|---|---|--|
| 04 | Get_Attribute_Single service error (An error with the read request of the Master) | | |
| | 03 | An invalid value for the Instance ID and the Attribute ID of the Get service command was specified. | Check the packet data of the Get service request from the Master and correctly specify the Attribute ID value. |
| 05 | Set_Attribute_Single error (An error with the write request of the Master) | | |
| | 02 | The Set service is not supported by the specified Attribute ID. | Do not use the Set service request with the specified Attribute ID. |
| | 03 | The Attribute ID is outside the range. | Correctly specify the Attribute ID. |
| | 04 | The attribute value written in the Set service is outside the range. | |
| 06 | Reset service error (An error with the reset request of the Master) | | |
| | 01 | The state of the connection is contradicted. | Do not reset except when in time-out. |
| | 02 | The parameter of the Reset service is outside the range. | Correctly specify parameter. |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|--|--|
| 07 | I/O Polling request receive error | | |
| | 01 | Middle or final packet was received before the first packet data. | Check the parameter settings of the Master with regard to the Master/Slave I/O communication, and status of the DeviceNet. |
| | 02 | Sequence error for receiving the I/O data. | |
| | 03 | Receives more data than the maximum number of I/O byte data. | |
| 08 | Explicit message reception error | | |
| | 01 | The message fragment send and receive are operating concurrently. | Check the system configuration on the DeviceNet. |
| | 02 | Middle or final packet was received before the first packet data. | Check the parameter setting of the Master and the system configuration on the DeviceNet. |
| | 03 | The same message fragment was received twice rather than receiving the complete message. | |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|--|--|
| 08 | 04 | Sequence error for receiving a message fragment. | Check the parameter setting of the Master and the system configuration on the DeviceNet. |
| | 05 | Receives message larger than maximum number of message bytes allowed. | |
| | 06 | A message is received before the send message has finished transmission. | Check the system configuration on the DeviceNet. |
| A1 | DeviceNet object error | | |
| | 01 | An invalid service request for the DeviceNet Object Class was received. | Check the packet data for the Get service request from the Master or other node. |
| | 02 | An invalid request for the DeviceNet Object Instance was received. | |
| | 03 | Instance ID which does not exist in the DeviceNet Object Instance was specified. | |
| | 05 | An invalid Class ID for the DeviceNet Object Instance was specified. | |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|---|--|
| A2 | Connection object error | | |
| | 01 | The Connection Object Class was accessed. | Check the packet data for the Get service request from the Master. |
| | 02 | An invalid Service request for the Connection Object Instance was received. | |
| | 03 | Instance ID which does not exist in the Connection Object Instance was specified. | |
| | 04 | An invalid Attribute ID for the Connection Object Class was specified. | |
| A3 | Identity (ID) object error | | |
| | 01 | The Object Class ID was accessed. | Check the packet data for the Get service request from the Master. |
| | 02 | An invalid Service request for the Object Instance ID was received. | |
| | 03 | A value which does not exist in the Object Instance ID was specified. | |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|---|--|
| A5 | Assembly object error | | |
| | 01 | The Assembly Object Class was accessed. | Check the packet data for the Get service request from the Master. |
| | 02 | A value which does not exist in the Assembly Object Instance ID was specified. | |
| A6 | Grup2 Only UCMM message error | | |
| | 03 | The value which does not support to the Class ID or the Instance ID for the UCMM message is used. | Check the packet data for the Get service request from the Master. |
| B0 | Peer-to-peer message error | | |
| | 01 | The Service code received through the UCMM connection is not supported. | Check the client setting for the UCMM connection with regard to the Service code. |
| | 02 | The specified Selection Group for the UCMM connection from client is not supported. | Check the client setting for the UCMM connection with regard to the Selection Group. |
| 03 | The specified value of the Selection Group in the client is invalid. | | |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|--|--|
| B0 | 04 | The 64DNET is already connected to the number of devices specified in BFM #3. | Check the setting value in the BFM #3. The 64DNET cannot be connected to more devices than the number in BFM #3. |
| | 06 | The message ID received by the Open_Explicit_Messaging Connection Request is invalid in the specified message group. | Check the setting of the client with regard to the Source Message ID. |
| | 07 | The message ID received by the Open_Explicit_Messaging_Connection_Request already is used in the group 1 or group 3. | Check the setting of the client, and configuration of the DeviceNet. |
| | 08 | The Instance ID received by the Close_Connection_Request is not supported. | Check the setting of the client about Instance ID. |
| | 09 | There is no response for the Open Explicit Messaging Connection. | Check the power supply of the server and the connection. |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|--|--|
| B1 | Acknowledge Handler Object service error | | |
| | 01 | The Acknowledge Handler Object was accessed. | Check the packet data from the Master. |
| | 02 | The received Service request does not support on the Acknowledge Handler Object. | |
| | 03 | The specified Instance ID is invalidated on the Acknowledge Handler Object. | |
| | 04 | The specified Attribute ID is not supported in the Acknowledge Handler Object. | |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|--|--|---|
| B3 | BFM error | | |
| | 01 | The server does not have an empty buffer for receiving data. | Put the server into the state that data from the client can be received. |
| | 02 | The UCMM client sending trigger turns ON while data is still being sent to the server. | Check the PLC program with regard to BFM #12 ~ #15. Do not turn UCMM Client Sending Trigger for the same server to ON with data sent to the server. |
| | 03 | The 64DNET enters the Connection established request to itself. | Check the PLC program with regard to BFM #8 ~ #11. |
| | 04 | The 64DNET enters the UCMM client sending trigger to itself. | Check the PLC program with regard to BFM #12 ~ #15. |
| | 05 | FROM/TO WDT error | Check the PLC program. The PLC should access the BFMs (one or more) on every scan. |

Table 8.5: 64DNET Error Code

| Service Type (Upper Byte) | Detailed error code (Lower Byte) | Description | Check Points |
|------------------------------|---|--|---|
| B4 | Hardware error of 64DNET | | |
| | 00 | EEPROM error | Check the program of PLC about that accessing BFM of “Not used”, and please contact a service representative. |
| FF | Others | | |
| | 01 | The specified service does not support on the UCMM message. | Check the packet data from the Master or the client. |
| | 02 | The message router object was accessed. | |
| | 03 | The specified Class ID is invalidated. | |
| 04 | The network power supply is not supplied. | Check the connection and the network power supply. Supply current and voltage. | |

8.3.2 DeviceNet Error Code (BFM #7 Bit 1 is OFF)

This error code is general error code (upper byte) and additional error code (lower byte).

Table 8.6: DeviceNet Error Code

| General Error Code | Additional Error Code | Description | Check Points |
|--------------------|-------------------------|---|--|
| 02 | Resource unavailable | | Check the setting of all Masters. |
| | 01 | The Open Request Group Select is incorrect. | |
| | 02 | Invalid allocation/release choice parameter. | |
| | 04 | A necessary resource for Predefined Master/Slave connection set cannot be used. | |
| 08 | Service not supported | | Check the specification of client and server for the UCMM connection, and setting of the Master. |
| | FF | 64DNET does not support this service. | |
| 09 | Invalid attribute value | | Check the specification of client and server for the UCMM connection, and setting of the Master |
| | 02 | Allocation request is incorrect. | |
| | FF | Set_Attribute_Single service error | |

Table 8.6: DeviceNet Error Code

| General Error Code | Additional Error Code | Description | Check Points |
|--------------------|-------------------------|---|--|
| 13 | Not enough data | | |
| | FF | Not enough data | Check the length of send data. The send data is too small. |
| 14 | Attribute not supported | | |
| | FF | 64DNET does not support this attribute. | Check attribute ID in program for 64DNET and the setting of connected client and server. |
| 15 | Too much data | | |
| | FF | Too much data | Check the length of send data. The send data is too large. |
| 16 | Object does not exist | | |
| | FF | 64DNET does not have this object. | Check object (class ID, instance ID, attribute ID and service code) in program for 64DNET, and the setting of connected client and server. |
| 0B | Reserved by DeviceNet | | |
| | 02 | Allocate/Release request for Poll, COS and Cyclic is incorrect. | Check the parameter of Master. |

Table 8.6: DeviceNet Error Code

| General Error Code | Additional Error Code | Description | Check Points |
|--------------------|-----------------------|---|--|
| 0C | Reserved by DeviceNet | | |
| | 01 | The MAC ID is incorrect | Check wiring. |
| | FF | Set_Attribute_Single service error or Rest service error (This object does not support Set_Attribute_Single.) | Check class ID, instance ID, attribute ID and service code in program for 64DNET, and setting of connected client and server |
| 20 | Invalid parameter | | |
| | FF | Rest service error | Check wiring, and class ID, instance ID, attribute ID and service code of Master |

Appendix A: EDS (Electric Data Sheet)

The EDS (Electric Data Sheet) of 64DNET shows bellow.

Figure A-1: EDS of 64DNET

\$ File Description Section

[File]

```
DescText      = "FX2N 32 word Input / 32 Word Output Slave Unit";
CreateDate    = 01-06-2000;
CreateTime    = 16:14:50;
ModDate       = 01-06-2000;
ModTime       = 16:14:50;
Revision      = 1.0;
```

\$ Device Description Section

[Device]

```
VendCode    = 161;
VendName    = "Mitsubishi Electric Corporation";
ProdType    = 12;
ProdTypeStr = "Communication Adapter";
ProdCode    = 3;
MajRev      = 1;
MinRev      = 1;
ProdName    = "FX2N-64DNET";
Catalog     = "Users' Manual (JY992D86301)";
```

\$ I/O Characteristics Section

[IO_Info]

| | | |
|------------|-----------|-------------------------------------|
| Default | = 0x0001; | \$ Bit mapped |
| | | \$ Bit 0 = Poll |
| | | \$ Bit 1 = Strobe |
| | | \$ Bit 2 = Change of State |
| | | \$ Bit 3 = Cyclic |
| | | \$ Bit 4-15 = Reserved |
| PollInfo | = 0x000D, | \$ Combine Cyclic,COS I/O with Poll |
| | 1, | \$ Default Input = Input1 |
| | 1; | \$ Default Output = Output1 |
| COSInfo | = 0x000D, | \$ Combine Poll,Cyclic I/O with COS |
| | 1, | \$ Default Input = Input1 |
| | 1; | \$ Default Output = Output1 |
| CyclicInfo | = 0x000D, | \$ Combine Poll,COS I/O with Cyclic |
| | 1, | \$ Default Input = Input1 |
| | 1; | \$ Default Output = Output1 |

```

Input1      =
             0x04,          $ Size in Bytes
             0,             $ All bits significant
             0x000D,        $ Polled , COS or Cyclic Connection
             "Input Data",  $ Name String
             6,             $ Path size
             "20 04 24 64 30 03", $ IO Assy Obj Inst 64 Attr 3
             "";           $ help string

Output1     =
             0x04,          $ Size in Bytes
             0,             $ All bits significant
             0x000D,        $ Polled ,COS or Cyclic Connection
             "Output Data", $ Name String
             6,             $ Path size
             "20 04 24 65 30 03", $ IO Assy Obj Inst 65 Attr 3
             "";           $ help string

```

\$ Parameter Class Section

\$ [ParamClass]

\$ Parameter Section

\$ [Params]

\$ Parameter Enumeration Section

\$ [EnumPar]

\$ Parameter Groups Section

\$ [Groups]

\$ End

Appendix B: Class ID, Instance ID, Attribute ID and Service Code

B-1: Identity (0x01)

| Class | |
|-----------|-------------|
| Attribute | Not support |
| Service | Not support |

| Instance (01 Hex) | | | | | |
|-------------------|----|----------------|-----|-----|------------------------------|
| Attribute | ID | Description | GET | SET | Value |
| | 1 | Vendor ID | ✓ | | 161 |
| | 2 | Device type | ✓ | | 12 |
| | 3 | Product code | ✓ | | 3 |
| | 4 | Revision | ✓ | | 1H |
| | | Major revision | | | |
| | | Minor revision | | | |
| | 5 | Status | ✓ | | DeviceNet spesifica- tion |

| Service | DeviceNet service | | Parameter option |
|---------|-------------------|----------------------|------------------|
| | 0E | Get_Attribute_Single | |
| | 10 | Get_Attribute_Signal | |
| | 4B | Reset | |
| | 4C | Get_Attribute_Single | |

| | |
|---|--|
| Vendor-specific additional specifications | |
|---|--|

B-2: Message Router (0x02)

| | | |
|-----------|-------------|--|
| Class | | |
| Attribute | Not support | |
| Service | Not support | |

| | | |
|-----------|-------------|--|
| Instance | | |
| Attribute | Not support | |
| Service | Not support | |

| | |
|---|--|
| Vendor-specific additional specifications | |
|---|--|

B-3: DeviceNet (0x03)

| Class | | | | | |
|-----------|-------------------|----------------------|------------------|-----|-------|
| Attribute | ID | Description | GET | SET | Value |
| | 1 | Revision | ✓ | | 02 |
| Service | DeviceNet service | | Parameter option | | |
| | 0E | Get_Attribute_Single | | | |

| Instance | | | | | |
|-----------|----|-----------------------|-----|-----|--|
| Attribute | ID | Description | GET | SET | Setting Length |
| | 1 | MAC ID | ✓ | | 0 ~ 63 |
| | 2 | Device type | ✓ | | 0: 125kbps 1: 250kbps 2: 500kbps |
| | 3 | Baud rate | ✓ | ✓ | 0 |
| | 4 | BOI | ✓ | ✓ | 0 ~ 255 |
| | 5 | Bus-off counter | ✓ | | DeviceNet specifica- tion |
| | | Alloction information | | | |
| | | Master's MAC ID | | | |

| Service | DeviceNet service | | Parameter option |
|---|-------------------|-----------------------------|------------------|
| | 0E | Get_Attribute_Single | |
| | 10 | Get_Attribute_Signal | |
| | 48 | Allocate M/S Connection Set | |
| | 4C | Release M/S Connection Set | |
| Vendor-specific additional specifications | | | |

B-4: Assembly (0x04)

| | |
|-----------|-------------|
| Class | |
| Attribute | Not support |
| Service | Not support |

| | | | | | |
|---|-------------------|----------------------|------------------|-----|-------|
| Instance (64 Hex: input, 65 Hex output, 66 Hex: UCMM server, 67 Hex: UCMM client) | | | | | |
| Attribute Service | ID | Description | GET | SET | Value |
| | 3 | data | ✓ | ✓ | |
| | DeviceNet service | | Parameter option | | |
| | 0E | Get_Attribute_Single | | | |
| | 10 | Set_Attribute_Signal | | | |

| | |
|---|--|
| Vendor-specific additional specifications | |
|---|--|

B-5: Connection (0x05)

| | |
|---------------------------|-------------|
| Class | |
| Attribute | Not support |
| Service | Not support |
| Maximum active connection | 128 |

| | | | | | |
|---|-------------------------------------|-------------------------|-----|----------------------------|------------------------------------|
| Instance (01 Hex, 0A ~ 49 Hex, 64 ~ A3 Hex, 67 Hex: Explicit Message) | | | | | |
| Section | Information | | | Number of maximum Instance | |
| Instance Type | Explicit Message | | | | |
| Production Trigger | Change of State/Application Trigger | | | | |
| Transport Type | Server | | | | |
| Transport Class | 3 | | | | |
| Attribute | ID | Description | GET | SET | Value |
| | 1 | State | ✓ | | DeviceNet specification |
| | 2 | Instance type | ✓ | | 00 Hex (Explicit Message) |
| | 3 | Transport class trigger | ✓ | | 23 Hex (Client) 83 Hex (server) |

| | | | | |
|----|-------------------------------------|---|---|------------------------------|
| 4 | Produceed connection ID | ✓ | | DeviceNet specifica- tion |
| 5 | Consumed connection ID | ✓ | | DeviceNet specifica- tion |
| 6 | Initial comm. Characteris- tics | ✓ | | 21 Hex |
| 7 | Produceed connection size | ✓ | | FFFF Hex |
| 8 | Consumed connection size | ✓ | | FFFF Hex |
| 9 | Expected packet rate | ✓ | ✓ | 2500 ms |
| 12 | Watchdog time-out Action | ✓ | ✓ | 1 (Auto delete) |
| 13 | Produceed connection path length | ✓ | | 0 |
| 14 | Produceed connection path | ✓ | | NULL |
| 15 | Consumed connection path length | ✓ | | 0 |
| 16 | Consumed connection path | ✓ | | NULL |
| 17 | Produce_Inhibit_Time | ✓ | | 0 |

| | | | |
|---------|-------------------|----------------------|------------------|
| Service | DeviceNet service | | Parameter option |
| | 05 | Reset | |
| | 0E | Get_Attribute_Single | |
| | 10 | Set_Attribute_Single | |

| | | | | | |
|-----------------------------|-------------|-------------------------|-----|----------------------------|------------------------------|
| Instance (02 Hex: I/O POLL) | | | | | |
| Section | Information | | | Number of maximum Instance | |
| Instance Type | Polled I/O | | | 1 | |
| Production Trigger | Cyclic | | | | |
| Transport Type | Server | | | | |
| Transport Class | 2 | | | | |
| Attribute | ID | Description | GET | SET | Value |
| | 1 | State | ✓ | | DeviceNet specifica- tion |
| | 2 | Instance type | ✓ | | 01 Hex (I/O Message) |
| | 3 | Transport class trigger | ✓ | | 82 Hex (server, class 2) |
| | 4 | Produceed connection ID | ✓ | | DeviceNet specifica- tion |
| | 5 | Consumed connection ID | ✓ | | DeviceNet specifica- tion |

| | | | | | |
|---------|-------------------|---------------------------------|------------------|---|-----------------------|
| | 6 | Initial comm. Characteristics | ✓ | | 01 Hex |
| | 7 | Produced connection size | ✓ | ✓ | BFM setting |
| | 8 | Consumed connection size | ✓ | ✓ | BFM setting |
| | 9 | Expected packet rate | ✓ | ✓ | 0 ms |
| | 12 | Watchdog time-out Action | ✓ | ✓ | 0 (time-out) |
| | 13 | Produced connection path length | ✓ | | 6 |
| | 14 | Produced connection path | ✓ | | 20 04 24 65 30 03 Hex |
| | 15 | Consumed connection path length | ✓ | | 6 |
| | 16 | Consumed connection path | ✓ | ✓ | 20 04 24 65 30 03 Hex |
| | 17 | Produce_Inhibit_Time | ✓ | | 0 |
| Service | DeviceNet service | | Parameter option | | |
| | 0E | Reset | | | |
| | 10 | Get_Attribute_Single | | | |
| | | Set_Attribute_Single | | | |

| Instance (04 Hex: I/O COS) | | | | | |
|----------------------------|-------------|------------------------------------|----------------------------|-----|------------------------------|
| Section | Information | | Number of maximum Instance | | |
| Instance Type | Polled I/O | | 2 | | |
| Production Trigger | COS | | | | |
| Transport Type | Server | | | | |
| Transport Class | 2 | | | | |
| Attribute | ID | Description | GET | SET | Value |
| | 1 | State | ✓ | | DeviceNet specifica- tion |
| | 2 | Instance type | ✓ | | 01 Hex (I/O Message) |
| | 3 | Transport class trigger | ✓ | | |
| | 4 | Produceed connection ID | ✓ | | DeviceNet specifica- tion |
| | 5 | Consumed connection ID | ✓ | | DeviceNet specifica- tion |
| | 6 | Initial comm. Characteris- tics | ✓ | | 01 Hex |
| | 7 | Produceed connection size | ✓ | | BFM setting |
| | 8 | Consumed connection size | ✓ | | 0 |
| | 9 | Expected packet rate | ✓ | ✓ | 0 ms |

| | | | | | |
|---|-------------------|---------------------------------|------------------|---|-----------------------|
| | 12 | Watchdog time-out Action | ✓ | | 0 (time-out |
| | 13 | Produced connection path length | ✓ | | 6 |
| | 14 | Produced connection path | ✓ | ✓ | 20 04 24 65 30 03 Hex |
| | 15 | Consumed connection path length | ✓ | | No ACK: 0 |
| | | | | | ACK: 4 |
| | 16 | Consumed connection path | ✓ | ✓ | No ACK: NULL |
| | | | | | ACK: 20 2B 24 01 |
| | 17 | Produce_Inhibit_Time | ✓ | | 0 |
| Service | DeviceNet service | | Parameter option | | |
| | 0E | Reset | | | |
| | 10 | Get_Attribute_Single | | | |
| | | Set_Attribute_Single | | | |
| Vendor-specific additional specifications | | | | | |

B-6: Acknowledge Handler Object (2B Hex)

| | |
|-----------|-------------|
| Class | |
| Attribute | Not support |
| Service | Not support |

| Instance (01 Hex) | | | | | |
|-------------------|-------------------|-----------------------------------|------------------|-----|------------------------|
| Attribute | ID | Description | GET | SET | Value |
| | 1 | Acknowledge Timer | ✓ | ✓ | 16 |
| | 2 | Retry Limit | ✓ | ✓ | 1 |
| | 3 | COS Producing Connection Instance | ✓ | | Connection Instance ID |
| Service | DeviceNet service | | Parameter option | | |
| | 0E | Get_Attribute_Single | | | |
| | 10 | Get_Attribute_Signal | | | |

| | |
|---|--|
| Vendor-specific additional specifications | |
|---|--|

Appendix C: List of Further Information Manual

For further information manual about FX Series, refer to following table. For DeviceNet master unit and other slave units, refer to each unit manual.

Table C-1: List of Further Information Manual

| Manual Name | Manual No. | Description |
|---|-------------|--|
| FX _{2N} Hardware Manual | JY992D66301 | This manual contains hardware explanations of wiring, installation and specifications for FX _{2N} Series programmable controllers. |
| FX ₀ /FX _{0N} Hardware Manual | JY992D47501 | This manual contains hardware explanations of wiring, installation and specifications for FX ₀ and FX _{0N} Series programmable controllers. |
| FX Programming Manual | JY992D48301 | This manual contains instruction explanations for the FX ₀ , FX _{0S} , FX _{0N} , FX, FX _{2C} , FX _{2N} and FX _{2NC} Series programmable controllers. |
| FX Programming Manual II | JY992D88101 | This manual contains instruction explanations for the FX _{1S} , FX _{1N} , FX _{2N} and FX _{2NC} Series programmable controllers. |

USER'S MANUAL

FX₂N-64DNET DeviceNet Interface Block



HEAD OFFICE: TOKYO BUILDING, 2-7-3 MARUNOUCHI, CHIYODA-KU, TOKYO 100-8310, JAPAN
HIMEJI WORKS: 840, CHIYODA CHO, HIMEJI, JAPAN

JY992D86301C
(MEE0603)

Effective Mar. 2006
Specification are subject
to change without notice.