

ADVANCED AND EVER-ADVANCING **MITSUBISHI ELECTRIC**

 **MITSUBISHI**  
PROGRAMMABLE CONTROLLERS  
MELSEC-F

Industrial automation

 **Elinco Group**

 European Union: [www.elinco.eu](http://www.elinco.eu)

 Russia: [www.elinc.ru](http://www.elinc.ru)

## USER'S MANUAL

RS-232C INTERFACE BLOCK FX<sub>2N</sub>-232IF

**FX<sub>2N</sub>**



**Foreword**

- This manual contains text, diagrams and explanations which will guide the reader in the correct installation and operation of the FX2N-232IF RS-232C Interface Block. It should be read and understood before attempting to install or use the unit. Further information can be found in the FX PROGRAMMING MANUAL, FX2N series hardware manuals.
- If in doubt at any stage during the installation of the FX2N-232IF RS-232C Interface Block always consult a professional electrical engineer who is qualified and trained to the local and national standards. If in doubt about the operation or use of the FX2N-232IF RS-232C Interface Block please consult the nearest Mitsubishi Electric distributor.
- This manual is subject to change without notice.



# **FX<sub>2</sub>N-232IF RS-232C INTERFACE BLOCK**

## **USER'S MANUAL**

Manual number: JY992D66701

Manual revision: A

Date : October 1997

**Guidelines for the safety of the user and protection of the FX2N-232IF  
RS-232C Interface Block**

This manual provides information for the installation and use of the FX2N-232IF RS-232C 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 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 the said product. All maintenance should be carried out in accordance with established safety practices.
- c) All operators of the compliance product 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 all 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 symbology 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 the 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 **POSSIBLY** cause physical and property damage.



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

#### Software warnings



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



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



3) Indicates a point of interest or further explanation.

**CONTENTS**

|   |            |
|---|------------|
| <b>1. INTRODUCTION</b> .....  | <b>1-1</b> |
| 1.1 Outline of Product .....  | 1-1        |
| <b>2. SPECIFICATIONS</b> .....  | <b>2-1</b> |
| 2.1 Appearance and Name of Each Portion .....                           | 2-1        |
| 2.2 General Specifications and Performance Specifications .....         | 2-2        |
| <b>3. CONNECTION AND WIRING</b> .....                                   | <b>3-1</b> |
| 3.1 Connection with the PC .....  | 3-1        |
| 3.2 Power Supply Wiring .....   | 3-2        |
| 3.3 Wiring of RS-232C Equipment .....                                   | 3-3        |
| <b>4. ALLOCATION OF BUFFER MEMORIES (BFMs)</b> .....                    | <b>4-1</b> |
| 4.1 BFM List .....  | 4-1        |
| 4.2 Details of Buffer Memories .....                                    | 4-3        |
| <b>5. TRANSMISSION PROGRAM</b> .....                                    | <b>5-1</b> |
| 5.1 Example of sending/receiving the data of 16-bit buffer length ..... | 5-1        |
| 5.2 Example of sending/receiving the data of 8-bit buffer length .....  | 5-8        |
| <b>6. APPENDIX</b> .....  | <b>6-1</b> |

## 1. INTRODUCTION

The RS-232C interface block FX<sub>2N</sub>-232IF (hereinafter referred to as "232IF") is connected to the FX<sub>2N</sub> programmable controller to realize full duplex serial data communication with another RS-232C interface such as a personal computer, bar code reader, printer, etc.

### 1.1 Outline of Product

#### Applicable PC

The 232IF can be connected as a special block of the FX<sub>2N</sub> programmable controller.

#### Control instructions

Send/receive data is received and sent and diversified control commands are manipulated using the FROM/TO instruction.

#### Number of I/O points

##### Connection method

The number of I/O points occupied is 8 in all (either input or output). However, the capacity of the 5 V power supplied from the PC is limited.

The current consumption of the 5 V power of the 232IF is 40 mA. Make sure that the total current consumption of the 5 V power including other special blocks is equivalent to or less than that available.

#### Communication method

Full duplex start-stop synchronization and non-protocol procedure are used. The communication format can be specified using the buffer memories (BFMs).

#### Send/receive buffer

The send/receive buffer can accommodate 512 bytes/256 words.

When the RS-232C interlink connection mode is used, data exceeding 512 bytes/256 words can also be received.

**ASCII/HEX conversion function**

The function to convert and send a hexadecimal numeric (0 to F) saved in the send data buffer as well as the function to convert a received ASCII code into a hexadecimal numeric (0 to F) and save it to the receive buffer are provided.

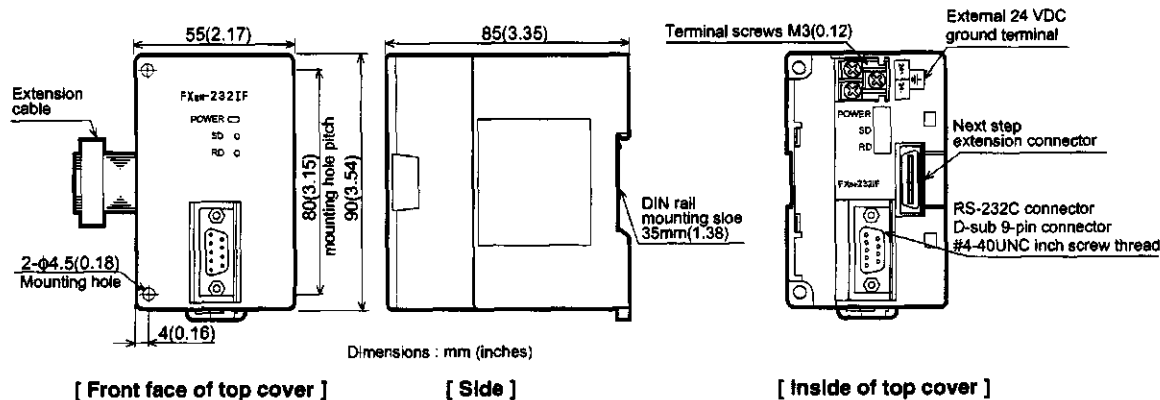


## 2. SPECIFICATIONS

### 2.1 Appearance and Name of Each Portion

Weight : Approx. 0.3 kg

Accessory : Special block No. label



- POWER LED** : Lt when both the 5 VDC power supplied from the PC basic unit and the 24 VDC power supplied from the external terminal are supplied.
- SD (TXD) LED** : Lt while data is sent to the RS-232C equipment connected to the 232IF.
- RD (RXD) LED** : Lt while data is received from the RS-232C equipment connected to the 232IF.

## 2.2 General Specifications and Performance Specifications

### General specifications

Insulating withstand voltage: 500 VAC, 1 minute between the entire external terminal and the ground terminal

Other specifications are equivalent to those of the PC basic unit.

### Performance specifications

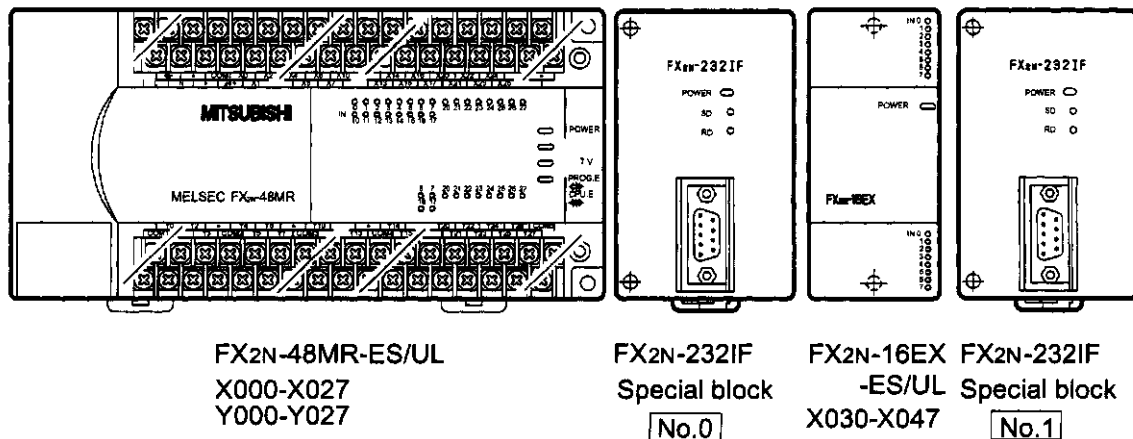
|  |  |
|--|--|
| Driving power supply   | 24 VDC $\pm$ 10%, 80 mA  |
| Current consumption  | 5 VDC, 40 mA (supplied from PC via extension cable)  |
| Transmission standard  | In accordance with RS-232C, D-Sub 9-pin connector  |
| Isolation method   | Photocoupler isolation   |
| Transmission distance  | 15 m or less   |
| Arrangement of D-sub 9-pin connector<br>Operation of each signal | <p>1 : CD (DCD) Carrier detection<br/>           2 : RD (RXD) Receive data<br/>           3 : SD (TXD) Send data<br/>           4 : ER (DTR) Data terminal ready<br/>           5 : SG (signal ground)<br/>           6 : DR (DSR) Data set ready<br/>           7 : RS (RTS) Request to send<br/>           8 : CS (CTS) Carrier to send<br/>           9 : CI (RI) Calling indicator</p> <p> <math>\rightarrow</math> : From 232IF to counterpart equipment<br/> <math>\leftarrow</math> : From counterpart equipment to 232IF<br/>           * : only shows status in BFM's.         </p> |
| Indication (LED)   | POWER, SD (TXD), RD (RXD)  |
| Communication method   | Full duplex start-stop synchronization, non-protocol procedure. Communication format is specified by buffer memories (BFMs).   |
| Transmission speed   | 300, 600, 1200, 2400, 4800, 9600, 19200  |
| Number of I/O points occupied                                    | 8 PC I/O points total (either input or output)   |
| Applicable PC  | FX <sub>2N</sub> programmable controller   |
| Communication with PC  | Communication is performed by FROM/TO instruction given by PC via buffer memories. Each of send/receive buffer accommodates 256 words.   |

### 3. CONNECTION AND WIRING

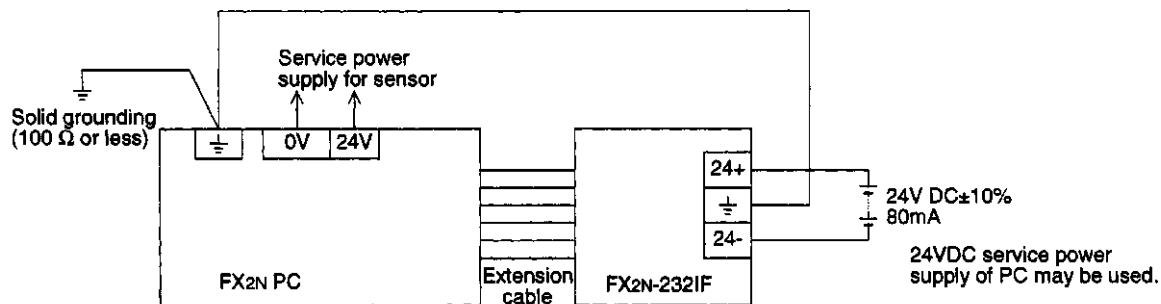
#### 3.1 Connection with the PC

##### Connecting the extension cable

The 232IF can be directly connected to the basic unit of the FX<sub>2N</sub> PC or connected on the right side of another extension block/unit. A number is assigned to each special unit/block counting from the one nearest the basic unit in the way of "No. 0, No. 1 . . . No. 7". Up to eight special units/blocks in all can be connected in principle. However, the capacity of the 5 VDC power supplied from the PC is limited. The current consumption of the 5 VDC power in the 232IF is 40 mA. Make sure that the total current consumption of the 5 VDC power supply including other special blocks is equivalent to or less than that available.



### 3.2 Power Supply Wiring



#### Wiring



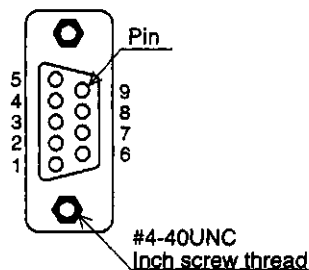
#### Handling of the crimp-style terminal

Use the crimp-style terminals of the dimensions shown on the figure on the left. Make sure that the tightening torque of the terminal is 0.5 to 0.8 N (5 to 8 kgf·cm). Tighten each terminal securely to avoid malfunction.



### 3.3 Wiring of RS-232C Equipment

Pin arrangement of communication connector



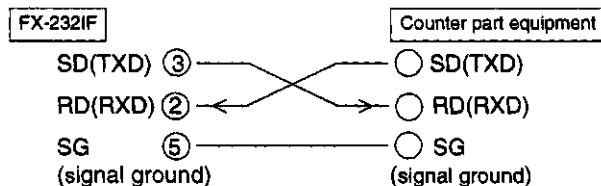
| Pin No. | Signal name | Function                               | Signal direction<br>232IF:Counterpart<br>equipment |
|---------|-------------|--|--|
| 1       | CD (DCD)    | Carrier detection                      | ←  |
| 2       | RD (RXD)    | Receive data (LED indication provided) | ←  |
| 3       | SD (TXD)    | Send data (LED indication provided)    | →  |
| 4       | ER (DTR)    | Data terminal ready                    | →  |
| 5       | SG          | Signal ground                          | —  |
| 6       | DR (DSR)    | Data set ready                         | ←  |
| 7       | RS (RTS)    | Request to send (or clear to receive)  | →  |
| 8       | CS (CTS)    | Clear to send                          | ←  |
| 9       | CI (RI)     | Calling indication (Ring indication)   | ←  |

**Connection example**

The signal wiring of the RS-232C equipment varies depending on the RS-232C specifications connected. Check the specifications of the RS-232C equipment used, then connect the signals correctly. Representative wiring examples are shown below.

**Connection with counterpart equipment of terminal specifications (when control line is not used)**

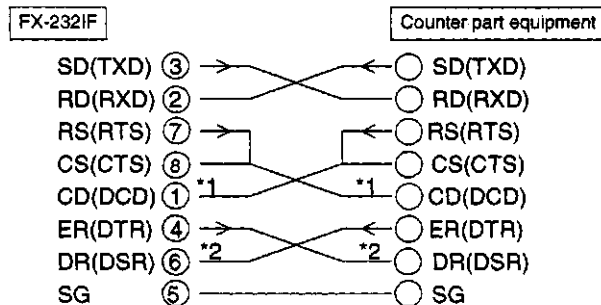
**BFM #0 communication format: b9 = 0, b8 = 0, without control line**



Communication is performed in accordance with the condition determined by the software in the 232IF and the counterpart equipment.

**Connection with counterpart equipment of terminal specifications (when control line is used)**

**Cross cable used, BFM #0 communication format: b9 = 0, b8 = 1, standard RS-232C mode**

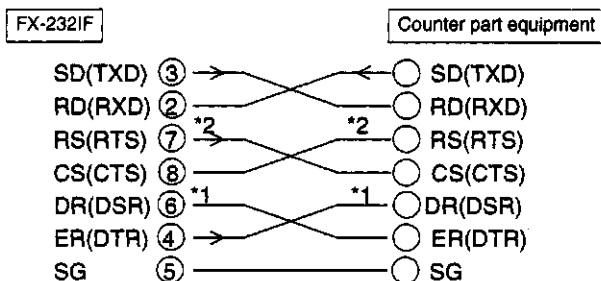


Because the carrier to send (CS) signal pin of the 232IF itself receives the request to send (RS) signal, signal transfer is performed as if the counterpart equipment is functioning.

\*1 When the CD signal is not monitored, the CD signal pin is not required to be connected. With regard to the CD signal, the 232IF only indicates the status.

\*2 The 232IF only indicates the status.

Interlink serial cross cable used, BFM #0 communication format: b9 = 1, b8 = 1, RS-232C interlink connection mode



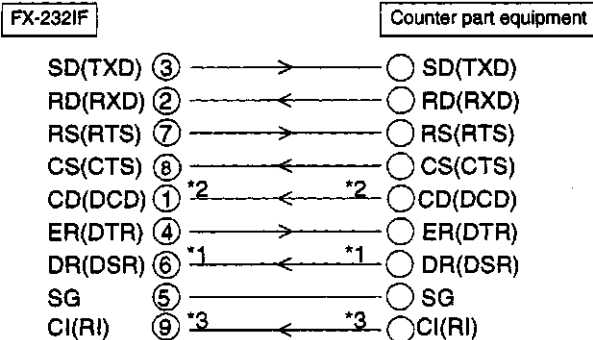
In the interlink connection mode, data exceeding 512 bytes (upper limit of the receive buffer in the 232IF) can be received.

\*1 The 232IF only indicates the status.

\*2 In this mode, the request to send (RS) signal functions as the signal to enable receive in the 232IF.

When receiving data exceeding 512 bytes (upper limit of the receive buffer in the 232IF), the 232IF sets the request to send (RS) signal to "OFF" and requests the counterpart equipment to suspend the send operation. When the data saved in the receive buffers is read by the sequence program, the remaining data can be received.

Connection with counterpart equipment of modem specifications (Control line is essential.)  
 Straight cable used, BFM #0 communication format: b9 = 0, b8 = 1, standard RS-232C mode



- \*1 The 232IF indicates the status exclusively.
- \*2 When the CD signal is not monitored, the CD signal pin is not required to be connected. With regard to the CD signal, the 232IF indicates the status exclusively.
- \*3 When the CI signal is not required, the CI signal pin is not required to be connected. With regard to the CI signal, the 232IF indicates the status exclusively.



## 4. ALLOCATION OF BUFFER MEMORIES (BFMs)

### 4.1 BFM List

The RS-232C interface block FX<sub>2N</sub>-232IF (232IF) transmits data with the PC via the buffer memories BFMs (16-bit RAM memories) in the 232IF.

FNC78 (FROM) and FNC79 (TO) instructions are used to read and write the buffer memories.

| BFM No. | Name  | Setting range  | Initial value    | R : For read<br>W:For write |
|---------|---|--|------------------|-----------------------------|
| # 0     | Communication format                                      |  | 0087H            | W                           |
| # 1     | Command   |  | 0                | W                           |
| # 2     | Receive upper limit byte count                            | 1 to 512 (when data length is 16 bits)<br>1 to 256 (when data length is 8 bits)<br>"0" is treated as "512" or "256". | 0                | W                           |
| # 3     | Receive time-out time                                     | 1 to 32,767 (X 10 ms)<br>"0" eliminates time-out time.   | 0                | W                           |
| # 4     | Send header, lower 2 bytes                                | 4 bytes max., zero suppression   | 0 (no header)    | W                           |
| # 5     | Send header, upper 2 bytes                                |  | 0                |                             |
| # 6     | Send terminator, lower 2 bytes                            | 4 bytes max., zero suppression   | 0(no terminator) | W                           |
| # 7     | Send terminator, upper 2 bytes                            |  | 0                |                             |
| # 8     | Receive header, lower 2 bytes                             | 4 bytes max., zero suppression   | 0 (no header)    | W                           |
| # 9     | Receive header, upper 2 bytes                             |  | 0                |                             |
| # 10    | Receive terminator, lower 2 bytes                         | 4 bytes max., zero suppression   | 0(no terminator) | W                           |
| # 11    | Receive terminator, upper 2 bytes                         |  | 0                |                             |
| # 12    | Receive suspension waiting time (in interlink connection) | 0 to 32,327 (X 10 ms)  | 0                | W                           |
| # 13    | Number of remaining send data                             | 0 to 512 (when data length is 16 bits)<br>0 to 256 (when data length is 8 bits)                                      | 0                | R                           |
| # 14    | Number of receive buffers                                 | 0 to 256 + 15 *1   | 0                | R                           |
| # 15    | Send sum result   |  | 0                | R                           |
| # 16    | Receive sum result  |  | 0                | R                           |

| BFM No.              | Name   | Setting range   | Initial value | R:For read<br>W:For write |
|----------------------|--|---|---------------|---------------------------|
| # 20                 | Time from CS ON to send start                                      | 0 to 32,327 (X 10 ms)   | 0             | W                         |
| # 21                 | Time from completion of actual send to RS OFF (completion flag ON) | 0 to 32,327 (X 10 ms)   | 0             | W                         |
| # 28                 | Status   |   | 0             | R                         |
| # 29                 | Error code   |   | 0             | R                         |
| # 30                 | Model code   |   | K7030         | R                         |
|                      |  |   |               |                           |
| #1000                | Send byte count  | 0 to 512 (when data length is 16 bits)<br>0 to 256 (when data length is 8 bits) | 0             | W                         |
| #1001<br>to<br>#1256 | Send buffers   |   | 0             | W                         |
|                      |  |   |               |                           |
| #2000                | Receive byte count   | 0 to 512 + 30 *1<br>0 to 256 + 15 *1  | 0             | R                         |
| #2000<br>to<br>#2256 | Receive buffers  |   | 0             | R                         |
| #2257<br>to<br>#2271 | Spare receive buffers for interlink connection mode                |   | 0             | R                         |

Note: "W: For write" can be used for read also. Undefined BFM Nos. are not allowed to be used in the program.

\*1 : Spare buffers used in the interlink connection mode

## 4.2 Details of Buffer Memories

BFM #0: Communication format

| Bit                  | Description  | 0   | 1     | Initial value        |
|----------------------|--|---|-------|----------------------|
| b0                   | Data length  | 7 bit   | 8 bit | 1 : 8 bit            |
| b1<br>b2             | Parity   | (00) : None<br>(01) : Odd<br>(11) : Even  |       | (11) : Even          |
| b3                   | Stop bit   | 1 bit   | 2 bit | 0 : 1 bit            |
| b4<br>b5<br>b6<br>b7 | Baud rate (bps)                                    | (0011) : 300<br>(0100) : 600<br>(0101) : 1200<br>(0110) : 2400<br>(0111) : 4800<br>(1000) : 9600<br>(1001) : 19200  |       | (1000) : 9600 bps    |
| b8<br>b9             | Control line                                       | (00) : Not used<br>(01) : Standard RS-232C<br>(11) : RS-232C interlink connection mode  |       | (00) : Not used      |
| b10<br>b11           | Addition of CR and LF                              | (00) : Not added<br>(01) : CR only<br>(11) : CR and LF  |       | (00) : Not added     |
| b12<br>b13           | Availability of check sum and ASCII/HEX conversion | (00) : Not available<br>(01) : ASCII/HEX conversion available<br>(10) : Check sum available<br>(11) : Check sum available, ASCII/HEX conversion available |       | (00) : Not available |
| b14                  | Send/receive buffer data length                    | 16 bit  | 8 bit | 0 : 16 bit           |
| b15                  | Undefined (disabled)                               |   |       | 0: Undefined         |

The communication format is determined on the rising edge of the send/receive enable command (BFM #1 b0). Accordingly, the setting of the communication format should be preliminary transferred using the TO instruction before BFM #1 b0 is turned on. Also, the send header and the send terminator are determined in the rising edge of the send command (BFM #1 b1). The receive header and the receive terminator are determined on the rising edge of BFM #1 b0 or on the rising edge of the receive completion reset command (BFM #1 b2). Accordingly, when only the header and the terminator exclusively are changed it is not necessary to turn BFM #1 b0 off. The change becomes valid from the next send/receive operation.

#### Setting example of communication format (hexadecimal, constant specification)

Data length : 8 bits

Parity : Odd

Stop bit : 1 bit

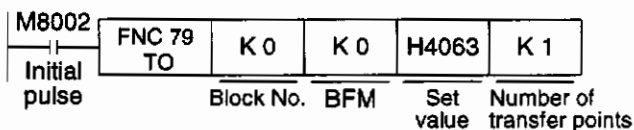
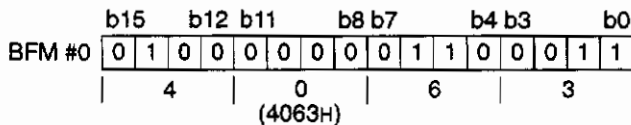
Baud rate : 2,400 bps

Control line : Not used

CR, LF : Not added

Check sum and ASCII/HEX conversion : Not available

Buffer data length : 8 bits



## Communication format list

|   |                      |      |            |     |    |    |  |
|---|----------------------|------|------------|-----|----|----|--|
| ① | <input type="text"/> | Data |            |     |    |    |  |
| ② | <input type="text"/> | Data | Terminator |     |    |    |  |
| ③ | <input type="text"/> | Data | Terminator | CR  |    |    |  |
| ④ | <input type="text"/> | Data | Terminator | CR  | LF |    |  |
| ⑤ | Header               | Data | Terminator | Sum |    |    |  |
| ⑥ | Header               | Data | Terminator | Sum | CR |    |  |
| ⑦ | Header               | Data | Terminator | Sum | CR | LF |  |
| ⑧ | <input type="text"/> | Data | CR         |     |    |    |  |
| ⑨ | <input type="text"/> | Data | CR         | LF  |    |    |  |

Select the communication format used to send/receive data in the 232IF among 9 types shown on the left.

- The header can be specified in the  portion in the communication format.
- In the communication format type ①, hexadecimal data (binary) and ASCII code can be send and received.  
In the communication format types ② to ⑨, the send/receive data should be any ASCII code except the header, the terminator, CR and LF.  
Communication can be performed using the ASCII/HEX conversion function by specifying the BFM #0 b13 and the BFM #0 b12.
- The ASCII codes available for the initial terminator are 01H to 1FH.
- In the RS-232C interlink connection mode, the communication formats ② to ⑦ are available.

● b0 to b7 (data length, parity, stop bit and baud rate):

b0 to b7 should be aligned with the communication specifications of the connected counterpart equipment.

● b9 and b8 (control line):

For examples of connecting the equipment corresponding to each setting, refer to Paragraph 3.3.

- When not used (b9 = 0, b8 = 0) is specified, communication is performed using only the SD and RD signals without using the control line.
- When standard RS-232C mode (b9 = 0, b8 = 1) is specified, a cross cable is required to connect the equipment of terminal specifications and a straight through cable is required to connect the equipment of modem specifications.
- When RS-232C interlink connection mode (b9 = 1, b8 = 1) is specified, the request to send (RS) signal functions as the signal to enable receive in the 232IF. When receiving data exceeding the receive upper limit byte count (BFM #2), the 232IF sets the request to send (RS) signal to OFF and requests the counterpart equipment to suspend the send operation.

At this time, when the data saved in the receive buffers is read to data registers in the PC using the sequence program, the remaining data can be received.

Make sure to perform the RS-232C interlink connection when specifying this mode.

● b11 and b 10 (addition of CR and LF):

Set these bits as follows.

- Not added (b11 = 0, b10 = 0)
- CR only is added. (b11 = 0, b10 = 1)
- CR and LF are added. (b11 = 1, b10 = 1)

For the CR/LF addition format, refer to the communication format list shown above.

- b13 and b12 (Availability of check sum and ASCII/HEX conversion):

Set these bits as follows.

- Neither the check sum nor the ASCII/HEX conversion is available. (b13 = 0, b12 = 0)
- The ASCII/HEX conversion only is available. (b13 = 0, b12 = 1)
- The check sum only is available. (b13 = 1, b12 = 0)
- Both the check sum and the ASCII/HEX conversion are available. (b13 = 1, b12 = 1)

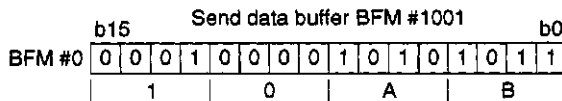
For the check sum addition format, refer to the communication format list shown above.

When execution of the ASCII/HEX conversion is specified, the hexadecimal numeric data (0 to F) inside the send buffers (BFMs #1001 to #1256) is converted into the ASCII code, then sent. The received ASCII code is converted into hexadecimal numeric data (0 to F), then saved to the receive buffers (BFMs #2001 to #2256).

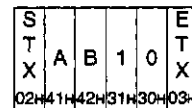
At this time, the send/receive byte count indicates the number of hexadecimal data.

#### Send format when hexadecimal data is converted into ASCII code

Example: When the send data "10ABH", the header "STX" and the terminator "ETX" are sent



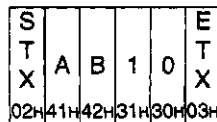
Converted into ASCII code  
before send



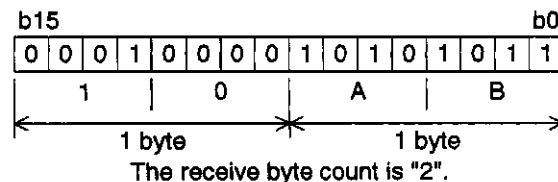
The send byte count is "4".

#### Receive format when ASCII code is converted into hexadecimal data

Example: When the receive data "10ABH", the header "STX" and the terminator "ETX" are received



Receive data buffer  
BFM #2001

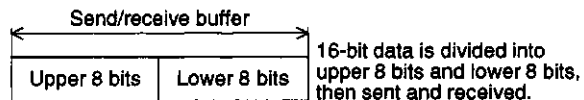




- b14 (send/receive buffer data length):

The data is treated as follows in accordance with the buffer data length.

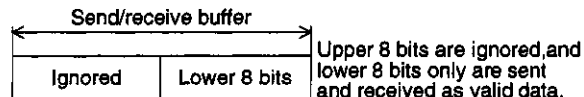
- In the case of 16 bits (b14 = 0)



Example of send buffers

|   |       |       |       |       |   |
|---|-------|-------|-------|-------|---|
| S | BFM   | BFM   | BFM   | BFM   | E |
| T | #1001 | #1001 | #1002 | #1002 | T |
| X | lower | upper | lower | upper | X |

- In the case of 8 bits (b14 = 1)



Example of send buffers

|   |       |       |       |       |   |
|---|-------|-------|-------|-------|---|
| S | BFM   | BFM   | BFM   | BFM   | E |
| T | #1001 | #1002 | #1003 | #1004 | T |
| X | lower | lower | lower | lower | X |

**BFM #1: Command**

| Bit | Description                      |
|-----|----------------------------------|
| b0  | Send/receive enable (ER ON)      |
| b1  | Send command                     |
| b2  | Receive completion reset command |
| b3  | Error reset                      |

BFM #1 gives the command for send/receive and the status information reset command to the 232IF.

● **b0 (send/receive enable):**

While b0 is turned on, the 232IF can send and receive data.

The contents of the following setting items are determined on the rising edge of b0. Make sure to set the contents using the TO instruction before setting b0 to "ON".

- BFM #0 (communication format)
- BFMs #9 and #8 (receive header)
- BFMs #11 and #10 (receive terminator)

On the rising edge of b0, the error occurrence (BFM #28 b3) and the error code (BFM #29) are cleared.

● **b1 (send command):**

On the rising edge of b1, the contents of the send buffers (BFMs #1001 to #1256) are sent to the counterpart equipment up to the send byte count (BFM #1000).

When send is completed, the send completion status (BFM #28 b0) is set. BFM #28 b0 is automatically reset when the next send command (b1) is given.

When b1 is given, the contents of the following setting items are determined.

- BFMs #5 and #4 (send header)
- BFMs #7 and #6 (send terminator)

## ● b2 (receive completion reset command):

When b2 is set to "ON", the following items are cleared.

- BFM #28 b1 (receive completion)
- BFM #2000 (receive byte count)
- BFM #2001 to #2256 (receive buffers)

When receive is completed, b2 should be set to "ON" to clear the receive completion status (BFM #28 b1). If BFM #28 b1 is not reset, the next data cannot be received.

When b2 is set to "ON", the contents of the following setting items are determined.

- BFMs #9 and #8 (receive header)
- BFMs #11 and #10 (receive terminator)

In the RS-232C interlink connection mode (BFM #0 b9 = 1, b 8 = 1), b2 functions as the receive continuation command to receive data exceeding the receive upper limit byte count (BFM #2), and clears the following items.

- BFM #28 b4 (receive suspended)
- BFM #2000 (receive byte count)
- BFMs #2001 to #2256 (receive buffers)
- BFMs #2257 to #2271 (spare receive buffers)

When b2 is set to "ON", the request to send (RS) signal is automatically set to "ON" also.

## ● b3 (error reset):

When b3 is set to "ON", the error occurrence status (BFM #28 b3) and error code (BFM #29) are cleared.

**BFM #2: Receive upper limit byte count**

Setting range 1 to 512 (when buffer data length is 16 bits)  
1 to 256 (when buffer data length is 8 bits)  
"0" is regarded as "512" or "256". The initial value is "0".

BFM #2 specifies the maximum byte count received by the 232IF.

When data is received up to the receive upper limit byte count, the receive completion status (BFM #28 b1) is set.

When the receive terminator (BFMs #11 and #10) or the receive time-out time (BFM #3) is set and the set condition is satisfied, it is regarded that receive is completed even if the data received is within the receive upper limit byte count.

**BFM #3: Receive time-out time**

Setting range 1 to 32,767 (x 10 ms)  
"0" eliminates time-out time. The initial value is "0".

BFM #3 specifies the receive data waiting time limit.

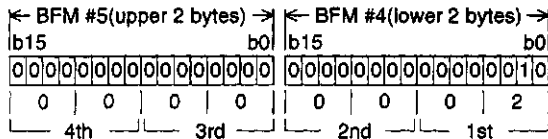
When the next data is not received within the receive time-out time starting from the receive edge of each data, the receive time-out flag (BFM #28 b2) is set to "ON", it is regarded that receive is completed, and the receive completion status (BFM #28 b1) is set.

**BFMs #5 (upper) and #4 (lower): Send header**

Setting range 4 bytes maximum, zero suppression

The initial value is "0" (not provided).

For the send data of the 232IF, 4 headers maximum can be specified. When the number of headers is less than 4, the upper "0"s are ignored (zero suppression) and not transferred.



Example:02H(STX)

- The transmission order is fourth header, third header, second header, first header when 4 headers are specified.

**BFMs #7 (upper) and #6 (lower): Send terminator**

Setting range 4 bytes maximum, zero suppression

The initial value is "0" (not provided).

For the send data of the 232IF, 4 terminators maximum can be specified. When the number of terminators is less than 4, the upper "0"s are ignored (zero suppression) and not transferred.

As the first terminator, specify an ASCII code from 01H to 1FH. (As the second to fourth terminators, any ASCII code can be specified.)

The register structure and the transmission order are equivalent to those of the send header described above.

**BFMs #9 (upper) and #8 (lower): Receive header**

Setting range 4 bytes maximum, zero suppression

The initial value is "0" (not provided).

For the receive data of the 232IF, 4 headers maximum can be specified. When the number of headers is less than 4, the upper "0"s are ignored (zero suppression).

The register structure and the transmission order are equivalent to those of the send header described above.

**BFMs #11 (upper) and #10 (lower): Receive terminator**

Setting range 4 bytes maximum, zero suppression

The initial value is "0" (not provided).

For the receive data of the 232IF, 4 terminators maximum can be specified. When the number of terminators is less than 4, the upper "0"s are ignored (zero suppression).

As the first terminator, specify an ASCII code from 01H to 1FH. (As the second to fourth terminators, any ASCII code can be specified.)

The register structure and the transmission order are equivalent to those of the send header described above.

**BFM #12: Receive suspension waiting time**

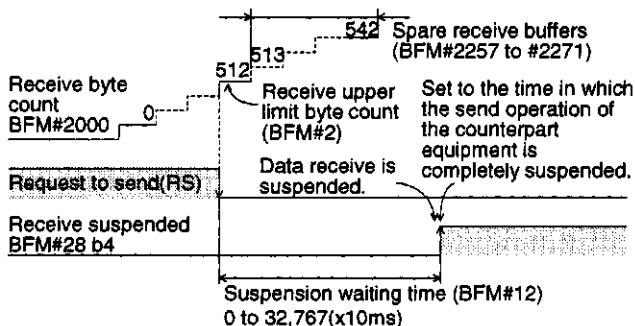
Setting range 0 to 32,767 (x 10 ms)

The initial value is 0 ms.

In the RS-232C interlink connection mode (BFM #0, b9 = 1, b8 = 1), when receiving data exceeding the receive upper limit byte count (BFM #2), the 232IF sets the request to send (RS) signal to "OFF" and requests the counterpart equipment to suspend the send operation.

BFM #12 specifies the time after the request to send (RS) signal is turned off until the receive suspended status (BFM #28 b4) is turned on. The value set to the BFM #12 should be equivalent to or more than the time after the 232IF sets the request to send (RS) signal to "OFF" until the send operation of the counterpart equipment is completely suspended.

If the time times out and the receive suspended status (BFM #28 b4) is set to "ON" before the send operation of the counterpart equipment is not suspended, the remaining data cannot be received.



**BFM #13: Number of remaining send data**

Saved value 0 to 512 (when buffer data length is 16 bits)

0 to 256 (when buffer data length is 8 bits)

The send byte count (BFM #1000) reduced by the number of data actually sent is during transmission.

**BFM #14: Number of receive buffers**Saved value 0 to 256<sup>+15</sup> (for spare receive buffers)

The number of buffers which have actually received data is saved in turn with regard to the receive buffers BFM #2001 to #2256 and the spare receive buffers BFM #2257 to #2271 for interlink connection mode.



**BFM #15: Send sum result**

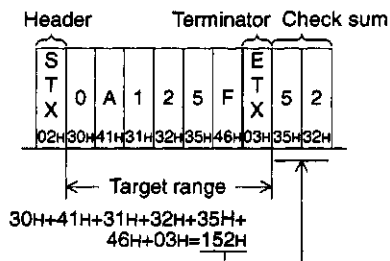
Initial value 0

The check sum value added to the send data is saved.

The sum check target range and the calculation method are shown below.

**Sum check target range and calculation method**

Example:



The total value including the terminator and excluding the header (1st byte of the header only) is calculated. Then, the lower 1 byte is converted into the ASCII code, and sent or received. The data converted into the ASCII code is placed in the order of upper digit and lower digit.

**BFM #16: Receive sum result**

Initial value: 0

The sum check value of the receive data is saved.

When the check sum added to the receive data is different from the receive sum result, "receive sum check error" occurs.

The sum check target range and the calculation method are equivalent to those for the send sum result described above.

**BFM #20: Time from CS ON to send start**

Set value 0 to 32, 767 (x 10 ms)

The initial value is 0 ms.

The time after the clear to send (CS) signal is turned on until the 232IF starts the send operation can be set.

When "control line not used" is specified, the time after the send command is given until the send operation is started is specified.

Set BFM #20 when it is required by a modem, etc.

**BFM #21: Time from completion of actual send to RS OFF (completion flag ON)**

Set value 0 to 32, 767 (x 10 ms)

The initial value is 0 ms.

The time after the 232IF sends the data until the RS signal is turned off and the send completion flag (BFM #28 b0) is turned on can be specified.

Set BFM #21 when it is required by a modem, etc.

**BFM #28: Status**

| Bit | Description        |
|-----|--------------------|
| b0  | Send completion    |
| b1  | Receive completion |
| b2  | Receive time-out   |
| b3  | Error occurrence   |
| b4  | Receive suspended  |
| b5  | Undefined          |
| b6  | Being sent         |
| b7  | Being received     |

| Bit | Description |
|-----|-------------|
| b8  | RS (RTS)    |
| b9  | ER (DTR)    |
| b10 | Undefined   |
| b11 | Undefined   |
| b12 | DR (DSR)    |
| b13 | CD (DCD)    |
| b14 | DS (CTS)    |
| b15 | CI (RI)     |

The 232IF status and the send/receive result are saved as status information. This information can be read from the PC using the FROM instruction, then utilized.

● b0 (send completion)

When send of data up to the send byte count (BFM #1000) is completed, the send completion status (b0) is set. The send completion status (b0) is automatically reset when the next send command (BFM #1 b1) is set to "ON".

● b1 (receive completion)

When receive of data up to the receive upper limit byte count (BFM #2) is completed, the receive completion status (b1) is set. If the receive terminator (BFMs #11 and #10) or the receive time-out time (BFM #3) is set, it is regarded that receive is completed when the set condition is satisfied, then the receive completion status (b1) is set in the same way.

This status is required to be reset using the sequence program. If it is not reset, the next data cannot be received. This status can be reset using the receive completion reset command (BFM #1 b2).

- b2 (receive time-out)

When the receive time-out time (BFM #3) is reached while data is received, the receive time-out status (b2) is set. At the same time, the receive completion status (b1) is also set.

This status is automatically reset when the receive completion reset command (BFM #1 b2) is executed.

- b3 (error occurrence)

When an error occurs while data is sent or received, b3 is set to "ON" and the error is saved to the error code (BFM #29).

- b4 (receive suspended)

When data exceeding the receive upper limit byte count (BFM #2) is received in the RS-232C interlink connection mode (BFM #0 b9 = 1, b8 = 1), the 232IF sets the request to send (RS) signal to "OFF", requests the counterpart equipment to suspend the send operation, then sets b4 after the receive suspension waiting time (BFM #12) has expired.

To receive the excess data in the interlink connection, the rising edge of the b4 is required to be monitored using the sequence program. The data as much as the receive byte count (BFM #2000) in the receive buffers (BFMs #2001 to #2271) or the data as much as the number of receive buffers (BFM #14) should be read to data registers in the PC and the receive completion command executed (BFM #1 b2).

- b6 (being sent)

b6 is turned on after the send command (BFM #1 b1) is given until the send completion status (BFM #28 b0) is set.

- b7 (being received)

b7 is turned on after the head data is received until the receive completion status (BFM #28 b1) is set.

- b8 (RS), b9 (ER), b12 (DR), b13 (CD), b14 (CS), b15 (CI)

These bits indicate the ON/OFF status of the control signals.

## BFM #29: Error code

| Code | Description   | Causes and countermeasures   |
|------|---|--|
| 0    | No error  | —  |
| 1    | Receive parity error, overrun error, framing error          | Communication format such as baud rate is not matched.<br>Control timing is not matched.   |
| 2    | Undefined   | —  |
| 3    | Defective receive character                                 | Receive data is not ASCII code.  |
| 4    | Receive sum check error                                     | Receive sum is not equal calculated sum result (BFM #16).  |
| 5    | Receive buffer overflow (only in interlink connection mode) | Receive byte count exceeds 512 +30 bytes.<br>Decrease receive upper byte count (BFM #2), and increase spare receive buffer area. |
| 6    | Baud rate setting error                                     | Non-existing baud rate is specified.   |
| 7    | Receive CR error  | CR is not placed in correct position.  |
| 8    | Receive LF error  | LF is not placed in correct position.  |
| 9    | Send/receive initial terminator setting error               | Initial terminator is other than 01H to 1FH.   |
| 10   | Receive terminator error                                    | Receive terminator is not placed in correct position or not matched.   |
| 11   | Undefined   | —  |
| 12   | Transmission sequence error                                 | Transmission sequence is not matched.  |

**BFM #30: Model code**

The model code of the 232IF is "K7030".

The model code is a fixed code assigned to each special extension equipment handled by the FROM/TO instruction. The PC can distinguish the equipment type by reading this code.

**BFM #1000: Send byte count**

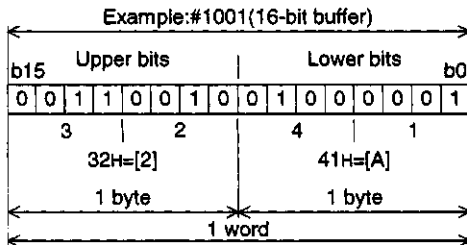
Setting range 0 to 512 (when buffer data length is 16 bits)

0 to 256 (when buffer data length is 8 bits)

The BFM #1000 specifies how many bytes out of 512 bytes/256 words in the 16-bit send buffers (BFMs #1001 to #1256) are to be sent.

**BFMs #1001 to #1256: Send buffers**

Each of them is a 16-bit buffer to save the send data, and accommodates 512 bytes/256 words.

**Send/receive buffer structure**

A numeric in the send/receive buffer is treated as hexadecimal (HEX).

**BFM #2000: Receive byte count**

Saved value 0 to  $512^{*30^{*1}}$  (when buffer data length is 16 bits)

0 to  $256^{*15^{*1}}$  (when buffer data length is 8 bits)

The byte count received from the counterpart equipment is saved.

This value is cleared by the receive completion reset command (BFM #1 b2).

\*1 Spare buffers in the interlink connection mode

**BFMs #2001 to #2256: Receive buffers**

Each of them is a 16-bit buffer to save the data received from the counterpart equipment, and accommodates 512 bytes/256 words. The buffer structure is equivalent to that of the send buffers.

The receive contents are cleared by the receive completion reset command (BFM #1 b2).

**BFMs #2257 to #2271: Spare receive buffers for interlink connection mode**

Each of them is a spare buffer for the interlink connection in the case where the data exceeding 512 bytes is received, and is used to receive the data after the request to send (RS) signal is turned off until the send operation of the counterpart equipment is suspended.

The receive contents are cleared by the receive completion reset command (BFM #1 b2).

# MEMO

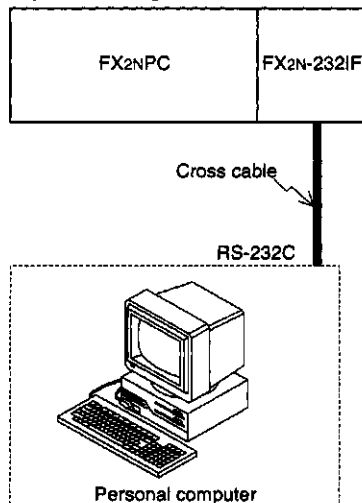


## 5. TRANSMISSION PROGRAM

### 5.1 Example of sending/receiving the data of 16-bit buffer length

This paragraph describes an example in which data of 16-bit buffer length is sent and received between the equipment of the terminal specifications. In this example, the ASCII code saved in the data registers D201 to D205 in the PC is sent to the counterpart equipment, and the data received from the counterpart equipment is saved to the data registers D301 to D304 in the PC.

#### System configuration



Setting example of buffer memories (The items not described here are set to the initial value respectively.)

**BFM #0: Communication format**

| Bit                  | Description  | Setting               |
|----------------------|--|-----------------------|
| b0                   | Data length  | (1): 8 bits           |
| b1<br>b2             | Parity   | (1, 1): Even          |
| b3                   | Stop bit   | (1): 2 bits           |
| b4<br>b5<br>b6<br>b7 | Baud rate  | (1001): 19200 bps     |
| b8<br>b9             | Control line                                       | (0, 0): Not used      |
| b10<br>b11           | Addition of CR and LF                              | (0, 0): Not added     |
| b12<br>b13           | Availability of check sum and ASCII/HEX conversion | (0, 0): Not available |
| b14                  | Send/receive buffer data length                    | (0): 16 bits          |
| b15                  | Undefined  | —                     |

← Specification item for 16-bit length

|     |       |   |   |   |    |   |   |   |   |   |   |   |   |   |   |         |
|-----|-------|---|---|---|----|---|---|---|---|---|---|---|---|---|---|---------|
| b15 | b8 b7 |   |   |   | b0 |   |   |   |   |   |   |   |   |   |   |         |
| 0   | 0     | 0 | 0 | 0 | 0  | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 1 | 1 |         |
| 0   |       |   |   | 0 |    |   |   | 9 |   |   |   | F |   |   |   | (009FH) |

**BFM #1: Command**

M0 → b0: Send/receive enable (ER ON)

M1 → b1: Send command

M2 → b2: Receive completion reset command

M3 → b3: Error reset

**BFM #2: Receive upper limit byte count**

8 bytes

**BFMs #4 to #11: Header and terminator**

BFMs #4 and #8 (send/receive header): 02H (STX)

BFMs #6 and #10 (send/receive terminator): 03H (ETX)

**BFM #28: Status**

b0 → M10: Send completion

b1 → M11: Receive completion

b2 → M12: Receive time-out

b3 → M13: Error occurrence

b4 → M14: Receive suspended

b5 → M15: Undefined

b6 → M16: Being sent

b7 → M17: Being received

b8 → M18: RS(RTS)

b9 → M19: ER(DTR)

b10 → M20: Undefined

b11 → M21: Undefined

b12 → M22: DR(DSR)

b13 → M23: CD(DCD)

b14 → M24: CS(CTS)

b15 → M25: CI(RI)

**BFM #1000: Send byte count**

9 bytes

**BFMs #1001 -: Send buffers**

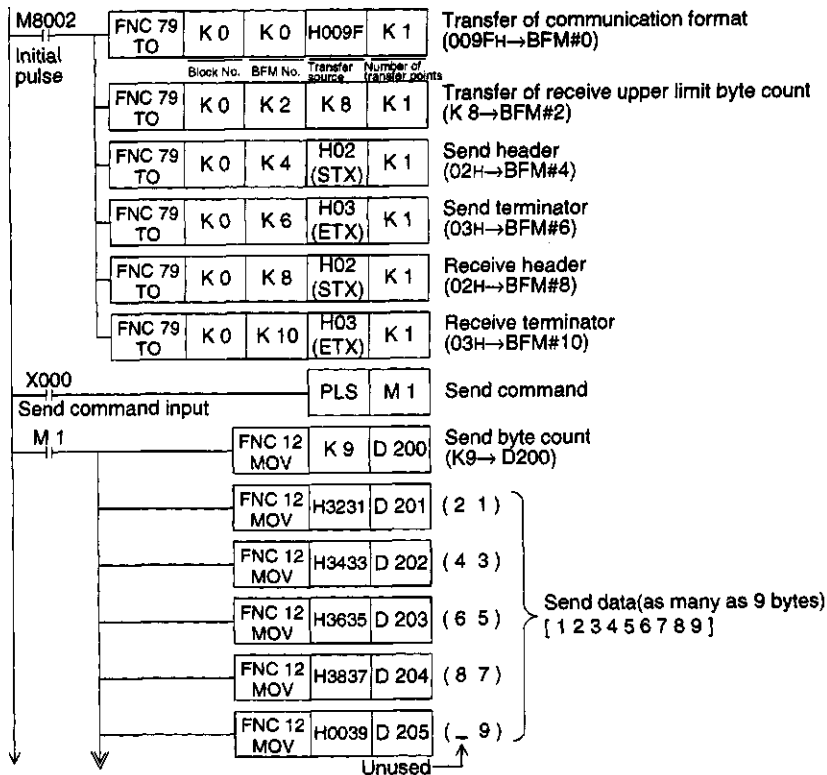
Nine-byte send data "123456789" is prepared in ASCII code in accordance with the send byte count specified above.

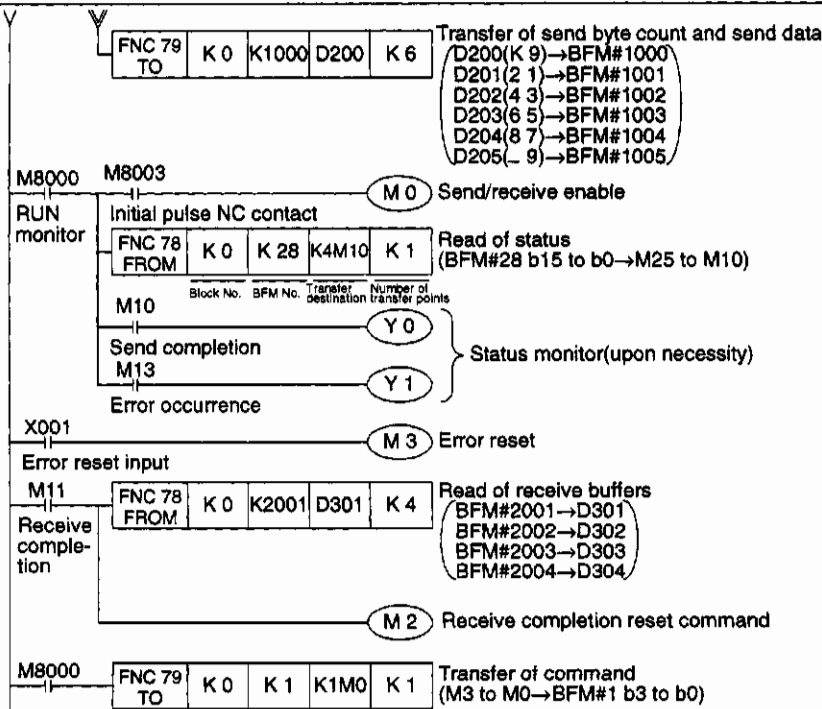
|                                      | Upper byte          | Lower byte          |
|--------------------------------------|---------------------|---------------------|
| (BFM#1001)                           | 2nd byte<br>2 (32H) | 1st byte<br>1 (31H) |
| (BFM#1002)                           | 4th byte<br>4 (34H) | 3rd byte<br>3 (33H) |
| (BFM#1003)                           | 6th byte<br>6 (36H) | 5th byte<br>5 (35H) |
| (BFM#1004)                           | 8th byte<br>8 (38H) | 7th byte<br>7 (37H) |
| 10th byte is not sent.<br>(BFM#1005) | 10th byte<br>* *    | 9th byte<br>9 (39H) |

**BFMs #2001 -: Receive buffers**

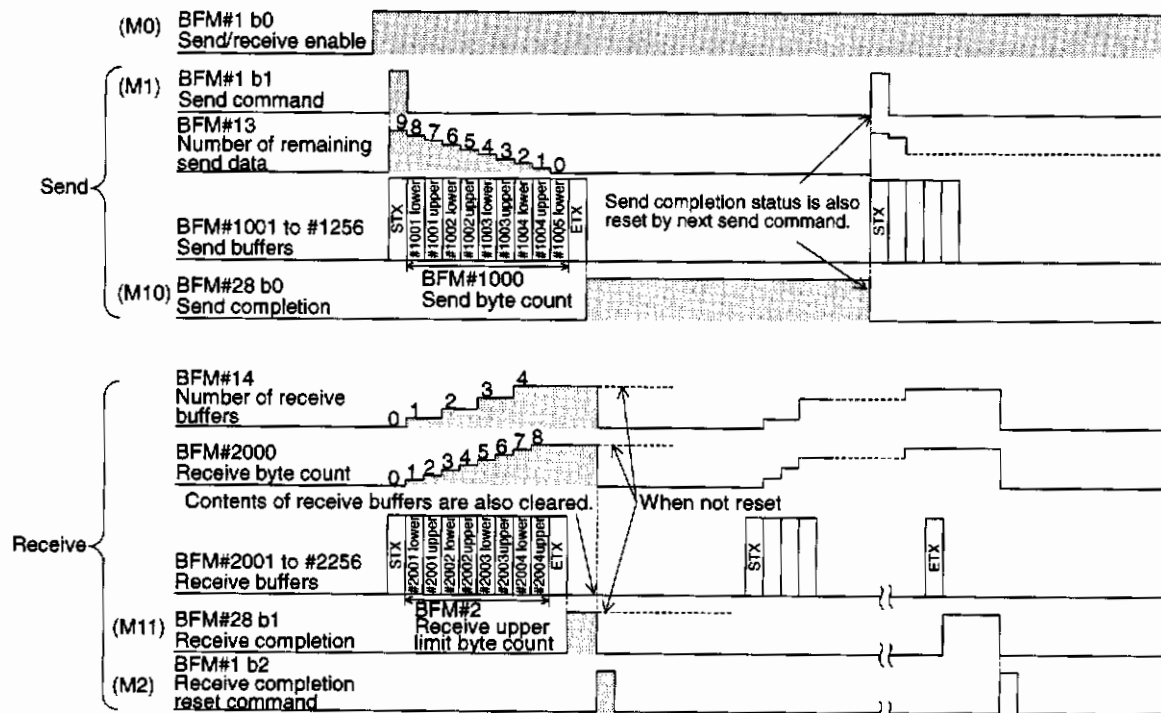
Eight-byte receive data specified in accordance with the receive upper limit byte count (BFM #2) is read to the data registers D301 to D304 in the PC.

## Example of sequence program





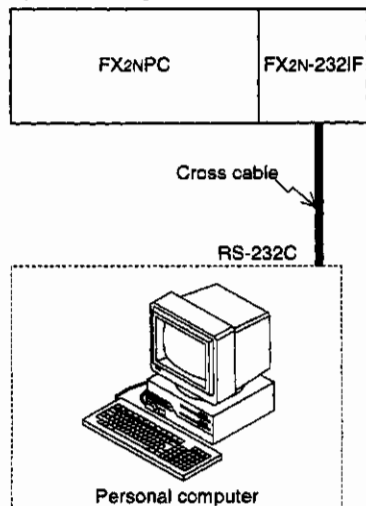
## Operation chart



## 5.2 Example of sending/receiving the data of 8-bit buffer length

This paragraph describes an example in which data of 8-bit buffer length is sent and received between the equipment of the terminal specifications. In this example, the ASCII code saved in the data registers D201 to D209 in the PC is sent to the counterpart equipment, and the data received from the counterpart equipment is saved to the data registers D301 to D308 in the PC.

### System configuration





Setting example of buffer memories (The items not described here are set to the initial value respectively.)

**BFM #0: Communication format**

| Bit                  | Description  | Setting               |
|----------------------|--|-----------------------|
| b0                   | Data length  | (1): 8 bits           |
| b1<br>b2             | Parity   | (1, 1): Even          |
| b3                   | Stop bit   | (1): 2 bits           |
| b4<br>b5<br>b6<br>b7 | Baud rate  | (1001): 19200 bps     |
| b8<br>b9             | Control line                                       | (0, 0): Not used      |
| b10<br>b11           | Addition of CR and LF                              | (0, 0): Not added     |
| b12<br>b13           | Availability of check sum and ASCII/HEX conversion | (0, 0): Not available |
| b14                  | Send/receive buffer data length                    | (1): 8 bits           |
| b15                  | Undefined  | —                     |

← Specification item for 8-bit length

|     |       |   |   |   |   |   |   |    |   |   |   |   |   |   |         |
|-----|-------|---|---|---|---|---|---|----|---|---|---|---|---|---|---------|
| b15 | b8 b7 |   |   |   |   |   |   | b0 |   |   |   |   |   |   |         |
| 0   | 1     | 0 | 0 | 0 | 0 | 0 | 0 | 1  | 0 | 0 | 1 | 1 | 1 | 1 | 1       |
| 4   |       |   |   | 0 |   |   |   | 9  |   |   |   | F |   |   | (409Fh) |

**BFM #1: Command**

M0 → b0: Send/receive enable (ER ON)  
M1 → b1: Send command  
M2 → b2: Receive completion reset command  
M3 → b3: Error reset

**BFM #2: Receive upper limit byte count**

8 bytes

**BFMs #4 to #11: Header and terminator**

BFMs #4 and #8 (send/receive header): 02H (STX)  
BFMs #6 and #10 (send/receive terminator): 03H (ETX)

**BFM #28: Status**

|                              |                      |
|------------------------------|----------------------|
| b0 → M10: Send completion    | b8 → M18: RS(RTS)    |
| b1 → M11: Receive completion | b9 → M19: ER(DTR)    |
| b2 → M12: Receive time-out   | b10 → M20: Undefined |
| b3 → M13: Error occurrence   | b11 → M21: Undefined |
| b4 → M14: Receive suspended  | b12 → M22: DR(DSR)   |
| b5 → M15: Undefined          | b13 → M23: CD(DCD)   |
| b6 → M16: Being sent         | b14 → M24: CS(CTS)   |
| b7 → M17: Being received     | b15 → M25: CI(RI)    |

**BFM #1000: Send byte count**

9 bytes

**BFMs #1001 ~: Send buffers**

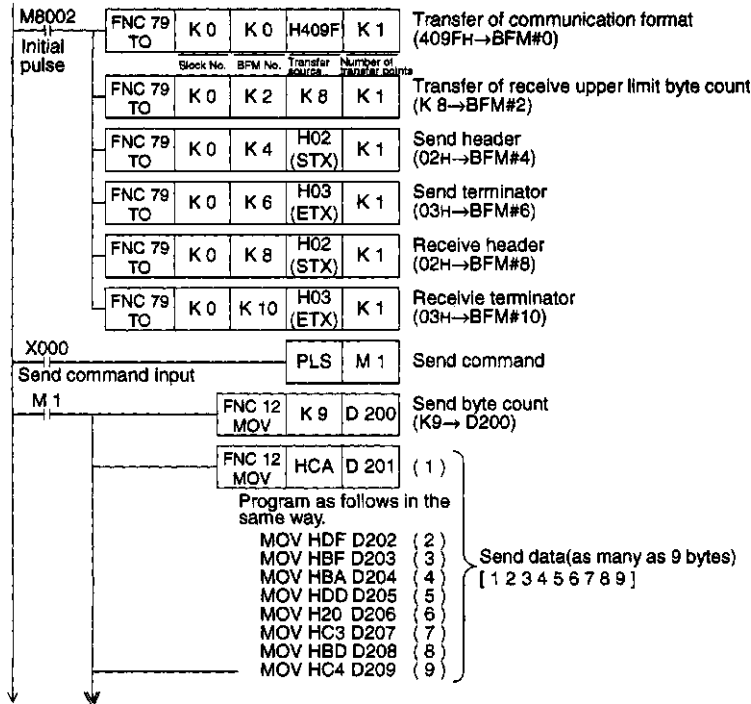
Nine-byte send data "123456789" is prepared in the ASCII code in accordance with the send byte count specified above.

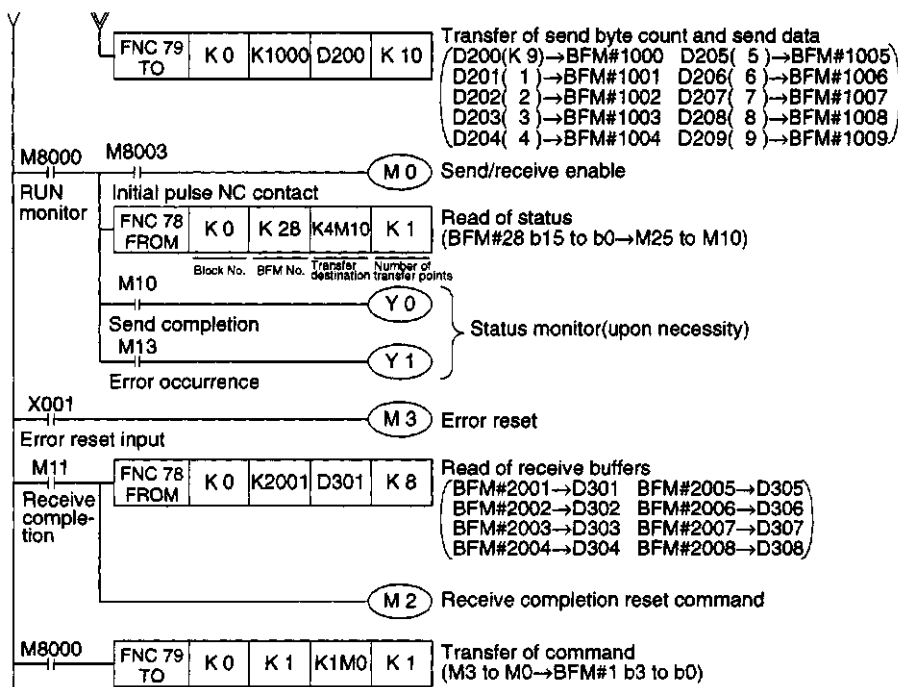
|            | Upper byte | Lower byte |        |
|------------|------------|------------|--------|
| (BFM#1001) | Ignored    | 1 (30H)    | 1 byte |
| (BFM#1002) | Ignored    | 2 (31H)    | 2 byte |
| (BFM#1003) | Ignored    | 3 (32H)    | 3 byte |
| (BFM#1004) | Ignored    | 4 (33H)    | 4 byte |
| (BFM#1005) | Ignored    | 5 (34H)    | 5 byte |
| (BFM#1006) | Ignored    | 6 (35H)    | 6 byte |
| (BFM#1007) | Ignored    | 7 (36H)    | 7 byte |
| (BFM#1008) | Ignored    | 8 (37H)    | 8 byte |
| (BFM#1009) | Ignored    | 9 (38H)    | 9 byte |

**BFMs #2001 ~: Receive buffers**

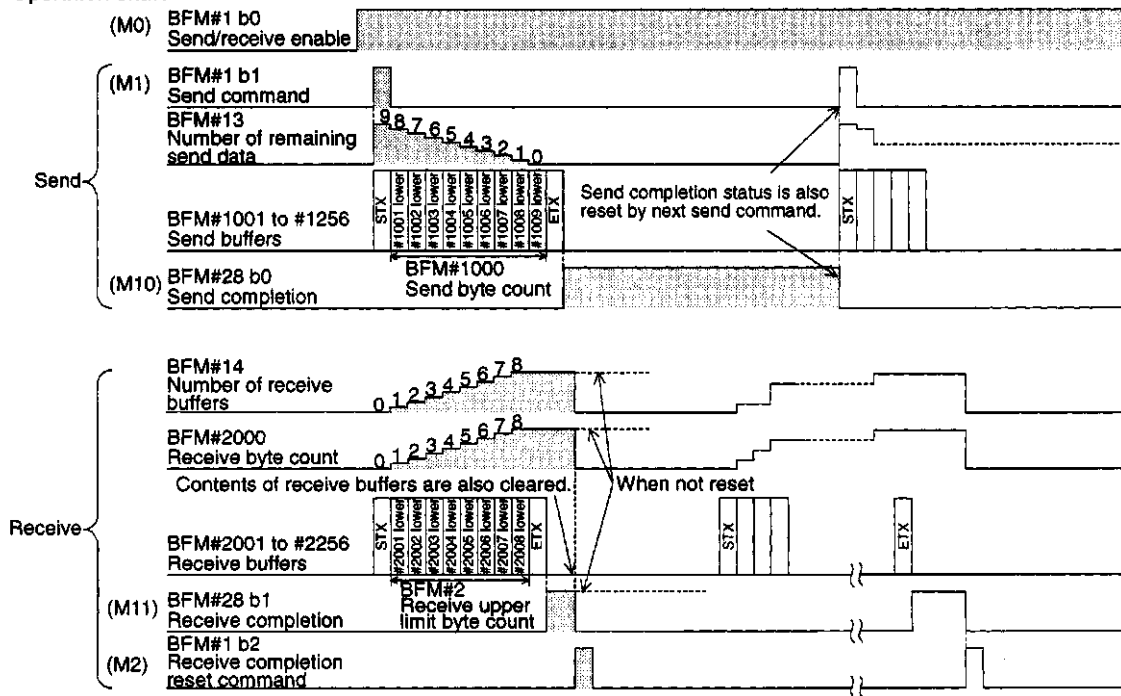
Eight-byte receive data specified in accordance with the receive upper limit byte count (BFM #2) is read to the data registers D301 to D308 in the PC.

Example of sequence program





Operation chart



## 6. APPENDIX

ASCII code table

| HEX | 0   | 1   | 2  | 3 | 4 | 5 | 6 | 7   |
|-----|-----|-----|----|---|---|---|---|-----|
| 0   |     | DLE | SP | 0 | @ | P | ` | p   |
| 1   | SOH | DC1 | !  | 1 | A | Q | a | q   |
| 2   | STX | DC2 | "  | 2 | B | R | b | r   |
| 3   | ETX | DC3 | #  | 3 | C | S | c | s   |
| 4   | EOT | DC4 | \$ | 4 | D | T | d | t   |
| 5   | ENQ | NAK | %  | 5 | E | U | e | u   |
| 6   | ACK | SYN | &  | 6 | F | V | f | v   |
| 7   | BEL | ETB | '  | 7 | G | W | g | w   |
| 8   | BS  | CAN | (  | 8 | H | X | h | x   |
| 9   | HT  | EM  | )  | 9 | I | Y | i | y   |
| A   | LF  | SUB | *  | : | J | Z | j | z   |
| B   | VT  | ESC | +  | ; | K | [ | k | {   |
| C   | FF  | FS  | ,  | < | L | \ | l |     |
| D   | CR  | GS  | -  | = | M | ] | m | }   |
| E   | SO  | RS  | .  | > | N | ^ | n | ~   |
| F   | SI  | US  | /  | ? | O | _ | o | DEL |

ASCII code table

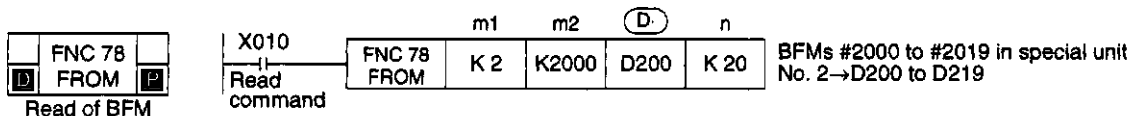
| Decimal | ASCII<br>(hexadecimal) |
|---------|------------------------|
| 0       | 30                     |
| 1       | 31                     |
| 2       | 32                     |
| 3       | 33                     |
| 4       | 34                     |
| 5       | 35                     |
| 6       | 36                     |
| 7       | 37                     |
| 8       | 38                     |
| 9       | 39                     |

| Code | ASCII<br>(hexadecimal) |
|------|------------------------|
| STX  | 02                     |
| ETX  | 03                     |

| Alphabet | ASCII<br>(hexadecimal) | Alphabet | ASCII<br>(hexadecimal) |
|----------|------------------------|----------|------------------------|
| A        | 41                     | N        | 4E                     |
| B        | 42                     | O        | 4F                     |
| C        | 43                     | P        | 50                     |
| D        | 44                     | Q        | 51                     |
| E        | 45                     | R        | 52                     |
| F        | 46                     | S        | 53                     |
| G        | 47                     | T        | 54                     |
| H        | 48                     | U        | 55                     |
| I        | 49                     | V        | 56                     |
| J        | 4A                     | W        | 57                     |
| K        | 4B                     | X        | 58                     |
| L        | 4C                     | Y        | 59                     |
| M        | 4D                     | Z        | 5A                     |



## Outline of FROM/TO command



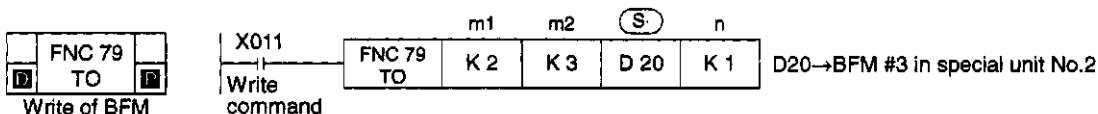
m1 : Special unit/block No. (K0 to K7 from the one nearest the basic unit)

m2 : Head specification No. in BFM (m2 = K0 to K32, 766)

**D** : Head element No. in transfer destination. Either one can be selected among T, C, D, KnM, KnY, KnS, V and Z. The element No. can be modified using the index.

n : Number of transfer points (n = K1 to K32, 767)

In the FX2N PC whose version is 2.00 or more, when the FROM/TO instruction is executed while the M8164 is turned on, the contents of the D8164 are treated as the number of transfer points (n).



m1, m2, n : Same as above

**S** : Head element No. in transfer destination. Either one can be selected among T, C, D, KnX, KnM, KnY, KnS, V, Z, K and H. The element No. can be modified using the index.

- When X010 and X011 are turned off, transfer is not executed and the data in the transfer destination is not changed.
- When a large quantity of data is read/written using the FROM/TO instruction, the watch dog timer (D8000) in the PC should be rewritten to a large value.

# MEMO

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.


# USER'S MANUAL


RS-232C INTERFACE BLOCK FX<sub>2N</sub>-232IF

 **mitsubishi electric corporation**

HEAD OFFICE: MITSUBISHI DENKI BLDG MARUNOUCHI TOKYO 100 TELEX: J24532 CABLE MELCO TOKYO  
HIMEJI WORKS: 840, CHIYODA CHO, HIMEJI, JAPAN

---

JY992D66701A  
(MEE 9710) 

Effective OCT. 1997  
Specifications are subject  
to  change without notice.