FUJI ELECTRIC

MICREX-SX *series*

SPH

USER'S MANUAL

STANDARD EXTENDED FB for GENERAL PURPOSE COMMUNICATION MODULE

FEH255

Preface

This User's Manual explains the specifications and operating methods for the expansion FBs that are provided as standard for communicating data between MICREX-SX Series general purpose communication module (NP1L-RS1/2/4) and external devices.
Read this manual carefully to ensure correct operation.
When using the module or peripheral devices, read the corresponding user's manuals listed below, together with this manual.

Title	Manual No.	Contents
User's Manual Instruction, MICREX-SX series SPH	FEH200	Explains the memory, language and system definitions of the MICREX-SX series.
User's Manual Hardware, MICREX-SX series SPH	FEH201	Explains the system configurations and the hardware specifications of modules in the MICREX-SX series.
User's Manual General Purpose Communication Module, MICREX-SX Series SPH	FEH225	Explains the memory, language and system definitions of the MICREX-SX series.
User's Manual D300win V2 <Reference>, MICREX-SX Series	FEH254	Explains the menus and icons of D300win V2 and all of the operations of D300win V2.

Notes

1. This manual may not be reproduced in whole or part in any form without prior written approval by the manufacturer.
2. The contents of this manual (including specifications) are subject to change without prior notice.
3. If you find any ambiguous or incorrect descriptions in this manual, please write them down (along with the manual No.shown on the cover) and contact FUJI.

Safety Precautions

Be sure to read the “Safety Precautions” thoroughly before using the module.
Here, the safety precaution items are classified into “Caution.”

⚠ Caution : Incorrect handling of the device may result in minor injury or physical damage.

Even some items indicated by “Caution” may also result in a serious accident.
Both safety instruction categories provide important information. Be sure to strictly observe these instructions.

⚠ Caution

- ◊ Do not use one found damaged or deformed when unpacked, otherwise, failure or erratic operation might be caused.
- ◊ Do not shock the product by dropping or tipping it over, otherwise, it might be damaged or troubled.
- ◊ Do not play back the CD-ROM supplied with the product using an ordinary audio CD player, otherwise, you may get your auditory sense or audio device damaged due to a loud sound.
- ◊ Engage the CD-ROM or loader connector in a correct orientation, otherwise, an erratic operation might occur.
- ◊ Sufficiently make sure of safety before program change, forced output, starting, stopping or anything else during a run. The wrong operation might break or cause machine problems.
- ◊ Be sure to keep within the software operating environment described in this manual when using this product, otherwise, an erratic operation or a failure might be caused.
- ◊ When operating the D300win or a personal computer, be sure to select a stable place so that they won’t be dropped. Otherwise, a failure might be caused.

Revision

***Manual No. is shown on the cover.**

Printed on	*Manual No.	Revision contents
Jul. 2002	FEH255	First edition

Contents

Preface

Safety Precautions

Revision

Contents

Contents

Section 1 General

Page

Section 1 General

1-1 List of Standard Expansion FBs for General Purpose Communication Module

Communication program for MICREX-SX series to communicate data with external devices via the general purpose communication module is made using the function blocks dedicated to communication. These communication function blocks include general non-procedural FB as well as the communication FBs dedicated to communication with specific external devices. The following FBs are prepared for communication purpose:

The following standard expansion FBs are prepared for the general purpose communication module. The standard expansion FBs are included in the programming support tool “D300win”.

Type	FB name	FB Overview
Non-procedural FB	_C_free	Non-procedural FB Send: 512 words Receive: 512 words
	_Cfr252	Non-procedural FB Send: 252 words Receive: 252 words
	_Cfr128	Non-procedural FB Send: 128 words Receive: 128 words
	_Cfr64	Non-procedural FB Send: 64 words Receive: 64 words
	_Cfr32	Non-procedural FB Send: 32 words Receive: 32 words
	_Cfrpr	Non-procedure FB which built into communication protocol Send: 512 words Receive: 512 words The program capacity is reduced by performing a part of the communication processing with the module.
	_Cfrp2	Non-procedure FB which built into communication protocol Send: 512 words Receive: 512 words The program capacity is reduced by performing a part of the communication processing with the module.In addition, communication processing can be mabe high-speed by using two SX bus message ports.
For FUJI Inverter	_CfdFRN	For FUJI Inverter FRENIC series
	_CfdFVR	For FUJI Inverter FVR-C11 series (FGI-BUS)
	_Cfvrpr	For FUJI Inverter FVR-C11 series (FGI-BUS) The program capacity is reduced by performing a part of the communication processing with the module.
For FUJI heater	_CfdPYX	For FUJI heater PYX series and PYH series
For FUJI bar code	_CfdPK	For FUJI bar code reader PK2 series
MODBUS	_C_modm	MODBUS procedure FB communicates data with MODBUS slave statins, making MICREX-SX the master station.

Note: For _Cfrp2, _CfdFRN, _CfdFVR, _Cfvrpr, _CfdPYX, and _CfdPK, refer to “MICREX-SX USER’S MANUAL GENERAL PURPOSE COMMUNICATION MODULE (FEH225).”

1-2 General Purpose Communication Package for Factory Automation Machine

In addition to standard expansion FBs for the general purpose communication module, the general purpose communication package for Factory Automation machine (NP4H-COMFV2) is prepared.

Procedure	FB name	Device
Temperature controller	_CrkREX	RIKA KOGYOU CO., LTD. REX-F, REX-D, FAREX-SR series
	_Com AX	OMRON Corporation. Digital temperature controller E5AX, E5XJ series
	_ComCK	OMRON Corporation. Digital temperature controller E5CK series
	_CymSDC	Yamatake Corporation. Digitronic temperature controller SDC40A/40G series
ID system	_ComV6	OMRON Corporation. V600 series
	_CshDS	SHARP MANUFACTURING SYSTEM CORPORATION. Microwave ID plate system DS series
	_CymWAM	Yamatake Corporation. Code distinguish ID system WAM120 series
	_CizFP	IDEC IZUMI CORPORATION. Data carrier system FP1A series
Bar code reader	_CtkTCD	THOKEN Co., Ltd. TCD8200/8500, TLMS-3200RV series
	_CndBCC	NIPPON ELECTRIC INDUSTRY CO., LTD. BCC2600 series, BL500, BL700
	_CkyBL	KEYENCE CORPORATION. BL180 series
	_CizDS	IZUMI DATA LOGIC CO., LTD. Bar code reader DS series
SECS	_C_SECS	SECS procedure semiconductor manufacturing system (for SECS-I only)
FANUC	_CDCN2	FANUC CNC machines
Serial Printer	_Cprint	Commercially available serial printers

Section 2 Installation

Page

Section 2 Installation

2-1 Personal Computer Environment Necessary for Standard Expansion FBs

To use standard expansion FBs, the operating environment of D300win plus the hard disk capacity necessary to install the expansion FBs that you want to use are required as personal computer operating environment.

Item		Specification
Hardware		IBM-AT compatible
CPU		Intel Pentium 233 MHz or higher
Hard disk		Min. 1 unit with 220 MB or greater free space (100 MB for D300win, 120 MB for expansion FBs)
External storage/ interface	Floppy disk drive	Min. 1 unit; medium: 2HD 1.25M/1.44M bytes, 3.5 inch
	CD-ROM drive	Min. 1 unit (quadruple speed or higher device is recommended), medium: ISO09660 format
Memory capacity		Min. 128 MB
Keyboard		106 Japanese (AO1) keyboard (CTRL + alphanumeric) or 101 English keyboard
Mouse		USB mouse, bus mouse or PS2 mouse
Display (resolution)		Min. 800x600 dots (1024x768 or higher is recommended)
Communication interface		COM port (RS-232C)
Necessary OS (see note)		WindowsNT 4.0 SP3 or higher, Windows95/98/ME/2000 (Japanese version or English version)

Note: OS depends on which version of D300win you use.
Adapted to Windows95/98: All V2 versions of D300win
Adapted to Windows2000: V2.2.0.0 or later versions of D300win
Adapted to WindowsME: V2.2.2.0 or later versions of D300win

2-2 Installation Procedure of Standard Extended FB

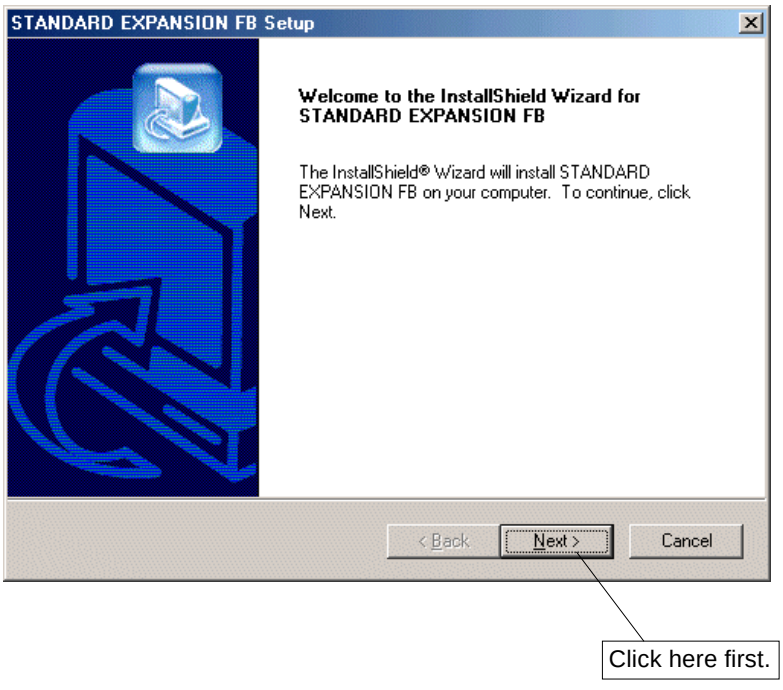
The standard extended FB is included in the Programming Support Tool D300win System Software Package (CD-ROM).

<Operating Procedure>

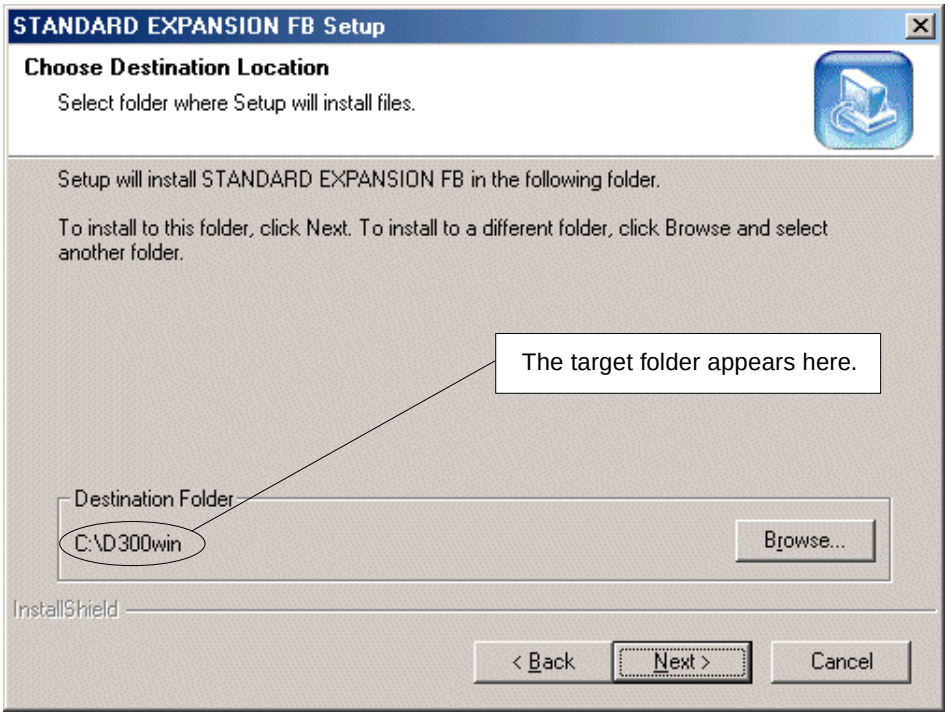
(1) Activate Windows and then insert the product CD-ROM into the CD-ROM drive. The following dialog appears.

Note: If the installation dialog box does not appear, perform the following operations.
Click [Start] and then [Run], enter: \autorun.exe, then click the [OK] button.

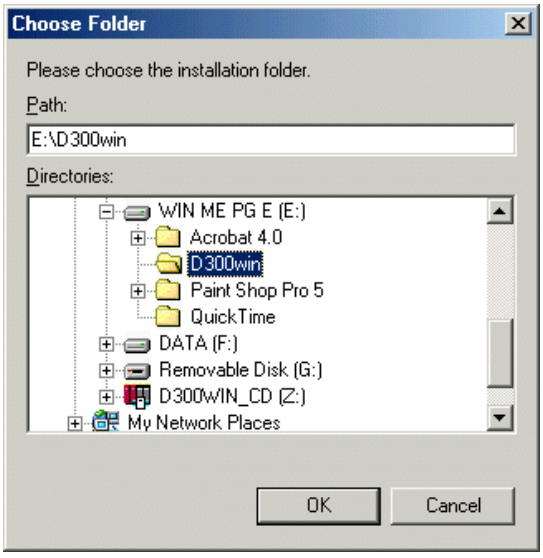
(2) When you click [Standard Expansion FB Setup], the following [Welcome] dialog box appears.



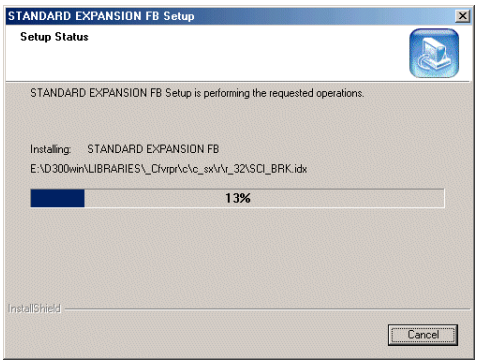
(3) When you click the [Next>] button in the [Welcome] dialog box, the [Choose Destination Locarion] dialog box appears.



To change the target folder, click the [Browse...] button. The following [Choose Folder] dialog box appears. Specify a folder and then click the [OK] button. Control returns to the [Choose Destination Location] dialog box.

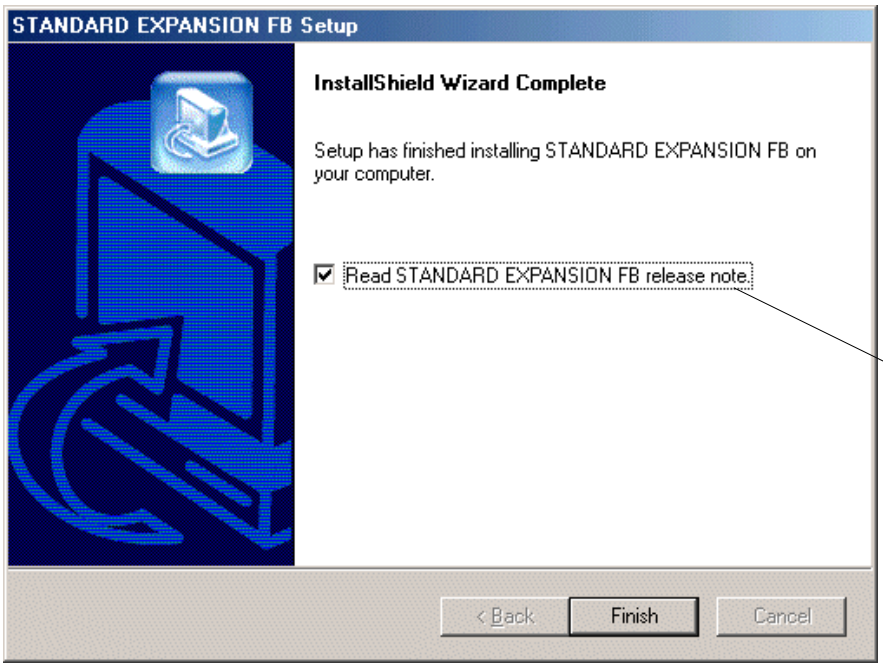


(4) When you click the [Next>] button in the [Choose Destination Location] dialog box, the installation processing is started.



2-2 Installation Procedure of Standard Extended FB

(5) When installation is completed, the following [InstallShield Wizard Complete] dialog box appears. Click the [Finish] button to complete the installation procedure.



When you check here and then click the [Finish] button, the product information of the standard extended FB appears.

<Example of release note>

README.TXT - Notepad

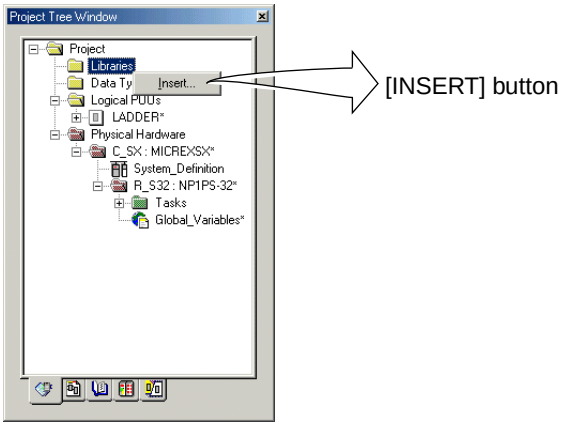
File Edit Search Help

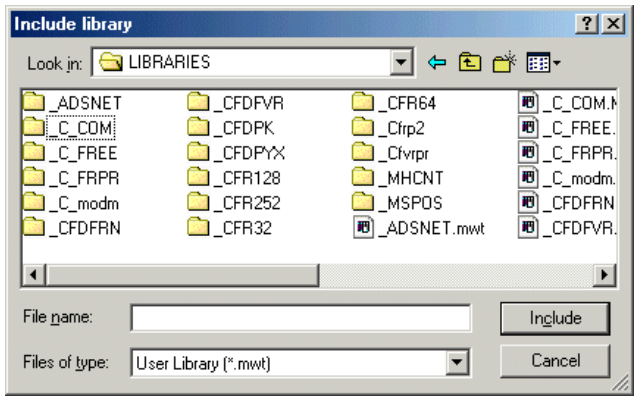
```
                    STANDARD EXPANSION FB RELEASE NOTE  V2.0.2.0E

Thank you for using this FB. This is an update description.

V2.0.2.0E
_ADSNET  library         :V2.0.0.0 added
_C_modm  library         :V2.0.0.0 added
_C_free  library         :V2.0.0.2 correction for module power-down
_CFr32   library         :V2.0.0.3 correction for module power-down
_CFr64   library         :V2.0.0.3 correction for module power-down
_CFr128  library         :V2.0.0.3 correction for module power-down
_CFr252  library         :V2.0.0.3 correction for module power-down
_C_frpr  library         :V2.0.0.3 correction for module power-down
                                   and warning 'SND_TMR un-used'
_CFrp2   library         :V2.0.0.2 correction for module power-down
_CFvrpr  library         :V2.0.0.3 correction for module power-down
_CFdFRN  library         :V2.0.0.2 correction for module power-down
_CFdFVR  library         :V2.0.0.2 correction for module power-down
_CFdPK   library         :V2.0.0.2 correction for module power-down
_CFdPYX  library         :V2.0.0.2 correction for module power-down
---------+---------+---------+---------+---------+---------+---------+---------
V2.0.1.0E
_CFrp2  library          :V2.0.0.1 added
_CFvrpr library          :V2.0.0.2 added
---------+---------+---------+---------+---------+---------+---------+---------
V2.0.0.2E
_C_COM  library          :V2.0.0.1 comment change
_C_free library          :V2.0.0.1 bug fixed  for bit length7 BCC,comment change
_CFr32  library          :V2.0.0.1 bug fixed  for bit length7 BCC,comment change
_CFr64  library          :V2.0.0.1 bug fixed  for bit length7 BCC,comment change
_CFr128 library          :V2.0.0.1 bug fixed  for bit length7 BCC,comment change
_CFr252 library          :V2.0.0.1 bug fixed  for bit length7 BCC,comment change
_C_frpr library          :V2.0.0.2 new FB
_CFdFRN library          :V2.0.0.1 comment change
_CFdFVR library          :V2.0.0.1 comment change
_CFdPK  library          :V2.0.0.1 comment change
_CFdPYX library          :V2.0.0.1 bug fixed  for MAX. word length
_MHCNT  library          :V2.0.0.0 no change
_MSPOS  library          :V2.0.0.0 no change
---------+---------+---------+---------+---------+---------+---------+---------

V2.0.0.1E
This is the original.
```

Individual communication FBs included in the standard expansion FB package can be used when registered in a library under a created project.

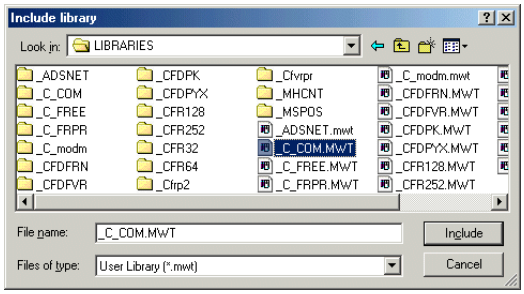
<Operation>

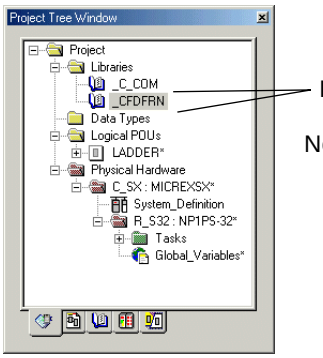
(1) Right-click “Libraries” in the project tree and then left-click the [Insert...] button. Then the [Include library] dialog box will appear on the screen.





(2) Select an FB (file name) that you want to register in the library, and left-click the [Insert] button. The project of the selected expansion FB is registered in the library.



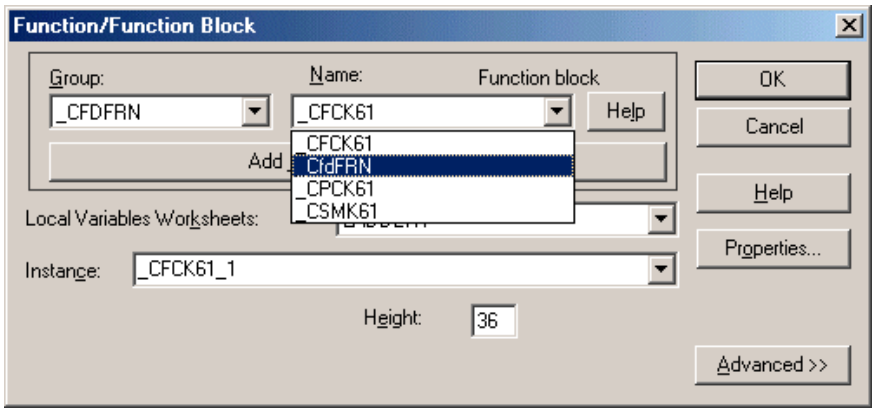


Note: “_C_COM” is necessary to use a standard expansion FB for communication.

2-3 Registration to Library

(3) FBs registered in the library can be selected from the [Function/Function Block] dialog box or from the Edit wizard.

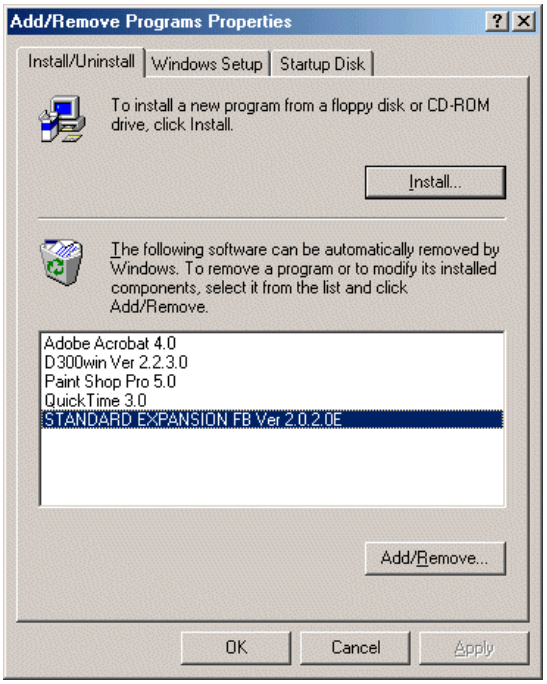
<[Function/Function Block] dialog box>



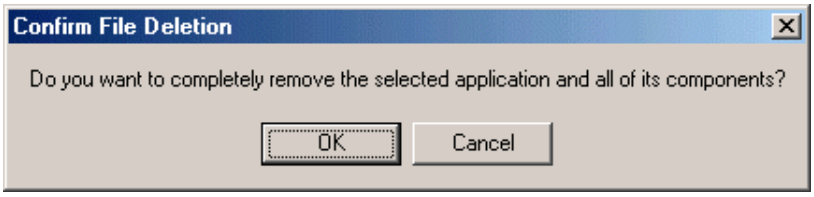
<Edit wizard>

<Operation>

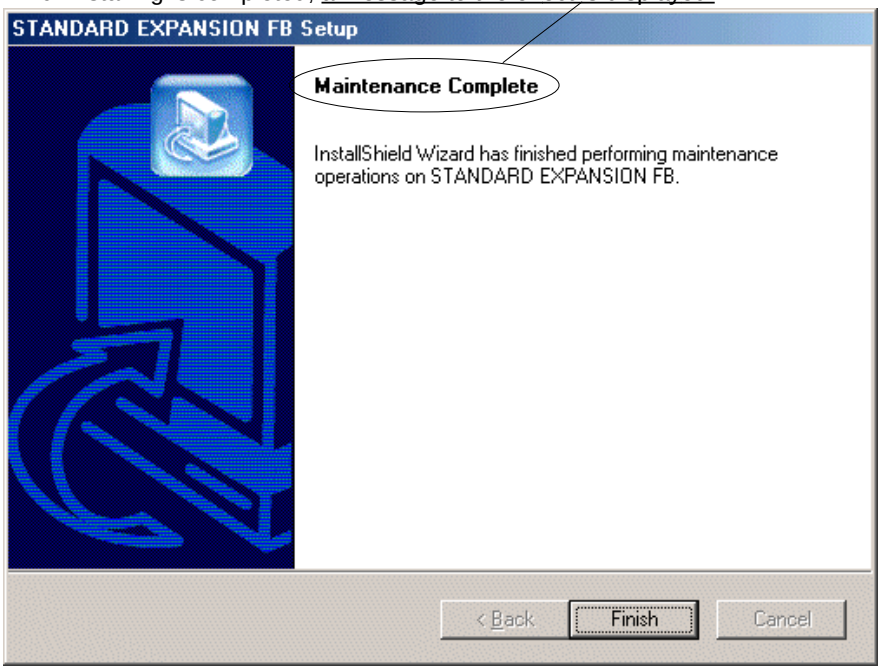
(1) Activate “Add/delete application” from the Control Panel window. The figure at left shows an example of Window98 screen image.



(2) Select “Standard Expansion FBs V2” and click the [Add/Remove...] button (the [Change/Remove...] button for Windows2000). The confirmation screen for file deleting is displayed.



(3) Clicking the [OK] button on the confirmation screen starts uninstalling the “Standard Expansion FBs V2” file. When uninstalling is completed, a message to the effect is displayed.



Section 3 Specification for Communication FB

Page

"_Cfrp2" Section 3 Specification for Communication FB

3-1 Communication Protocol Included Non-procedural FB "_Cfrp2"

3-1-1 General

For communication between MICREX-SX and external devices via RS-232C or RS-485 (NP1L-RS1/2/4), the "non-procedural" FB included in D300win is used. The non-procedural FB communicates data between the CPU module and external devices using the start-stop synchronization type non-procedural transmission protocol. 7 types of non-procedural FB are prepared, as shown below. Of these, the "_Cfrp2" FB performs communication at high speed by integrating part of the communication protocol in the general purpose communication module and using two ports for communicating SX bus messages.

FB Name	Specification Overview	
_C_free	Non-procedural FB Send: 512 words, Receive: 512 words	Note 1
_Cfr252	Non-procedural FB Send: 252 words, Receive: 252 words	Note 1
_Cfr128	Non-procedural FB Send: 128 words, Receive: 128 words	Note 1
_Cfr64	Non-procedural FB Send: 64 words, Receive: 64 words	Note 1
_Cfr32	Non-procedural FB Send: 32 words, Receive: 32 words	Note 1
_Cfrpr	Non-procedural FB Send: 512 words, Receive: 512 words A part of communication processing is processed by the firmware in the module.	Note 1 and 2
_Cfrp2	Non-procedural FB Send: 512 words, Receive: 512 words A part of communication processing is processed by the firmware in the module and two message boards are used to increase the processing speed.	Note 2

Notes:
1) For more information of non-procedural FBs other than "_Cfrp2", refer to the user's manual for the general purpose communication module (FEH225).
2) The communication protocol included non-procedural FB performs part of communication processing in the firmware in the module. Therefore, to use this FB, V2536 or later version (V2535 or later version for "_Cfrpr") of general purpose communication module is necessary.

The "_Cfrp2" FB performs the following operations in combination with the firmware in the module.

1) Initialization of communication ports
Initializes RS-232C and RS-485 ports (sets transmission speed, data bit length, parity bits, etc., so as to match the external device.)

2) Data sending/receiving function
Sends the data from an application program in the CPU of MICREX-SX series via RS1 to an external device, or outputs the data received via the general purpose communication module to an application program.

3) Monitoring of transmission condition
Monitors the condition of data transmission and, if abnormal, outputs error information.

<FB operating conditions>

System configuration:
One FB is necessary for each external device which is connected by RS-232C or RS-485 to one general purpose communication module. However, when RS-232C/RS-485 signal conversion is enabled (the mode selection switch is set to 4) for the general purpose communication module, each FB is used by two channels.

Memory capacity:
Program area = 1287 steps
Data memory capacity = Instance memory for user FBs (122 words)
+ Standard memory (1608 words)
+ Instance memory for system FBs (74 words)

Notes:
1) Above figures for memory capacity includes the area for the main body of non-procedural FB and those for sub-FBs that are called from the non-procedural FB.
2) Above figures for standard memory and retain memory includes the memory capacity necessary for sending/receiving data.

3-1-2 Specification for _Cfrp2

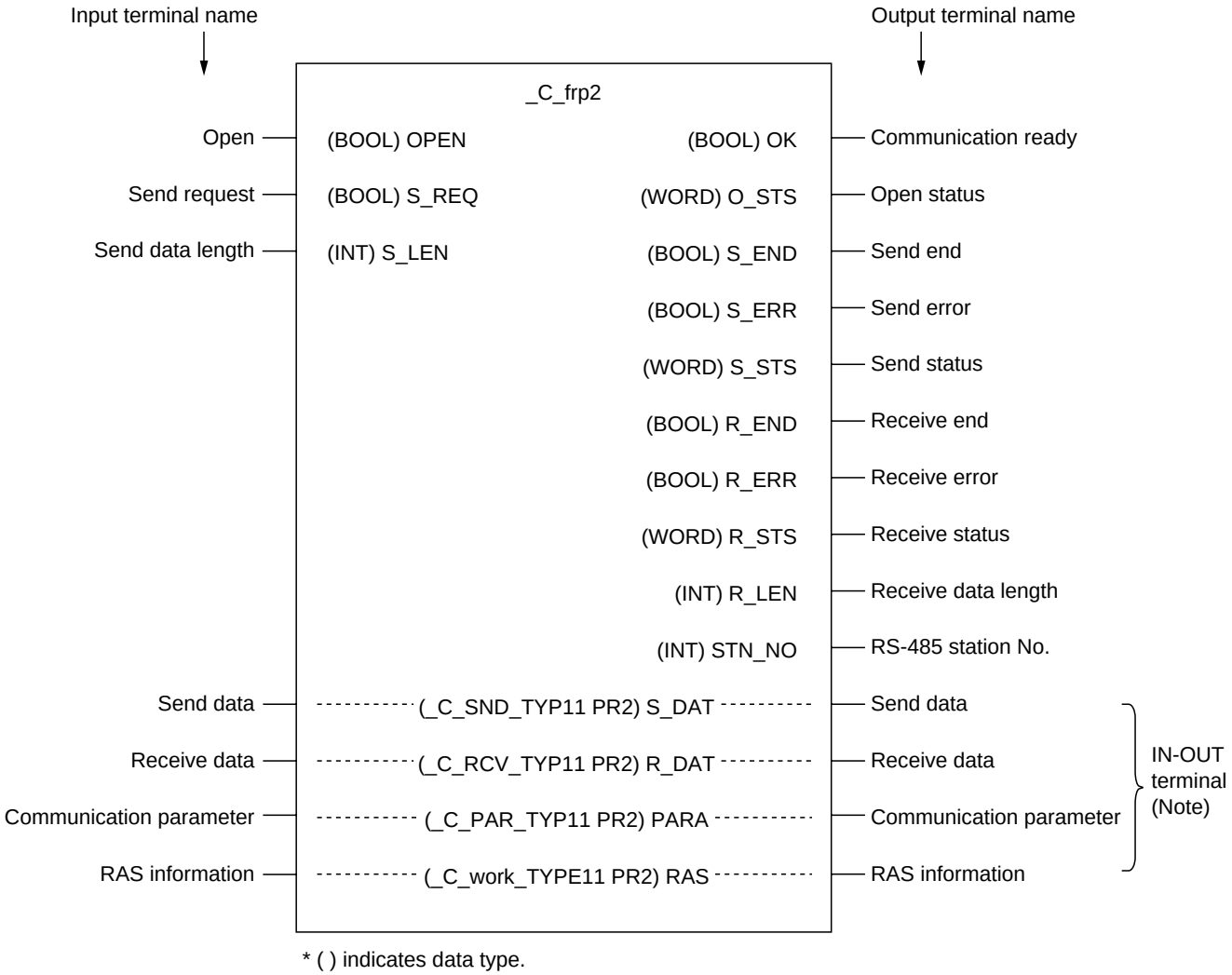
(1) Communication specification

Item		Specification
Transmission speed		300/600/1200/2400/4800/9600/19200/38400/57600/76800/115200 bps (Note)
Data bits		Selected from 7/8 bits
Parity bit		None/odd/even
Stop bits		Selected from 1/2 bits
DCE		Selected from DTE/DCE/modem DTE mode Modem DTE mode: Turn CD ON when receiving
Signal control	DTE mode	ER: always ON
	DCE mode	DR: always ON
Signal flow control	DTE mode	Off RS: always ON; Sending: unconditional On RS: ON during sending; Sending: when CS is ON
	DCE mode	Off CS: always ON; Sending: unconditional On CS: ON when RS is ON; Sending: when ER is ON
XON/XOFF control		Selectable
RS-485 mode		Selected from 4-wire/2-wire
Code conversion		Selected from None/ASCII conversion/ EBCDIC conversion (Convert binary data into character string variable.)
Frame detection	None	Disable frame detection function.
	Fixed length	Designate the number of receive data bytes
	Variable length	Designate start and end codes.
Horizontal parity (BCC)	None	None
	Upper order/Lower order	Calculation range designation. Designate position. Formula: Selected from add/add and invert/EOR/CRC BCC code: Selected from character code/binary code
	Lower order/Upper order	
Transmission timer value		x10ms

Note: The non-procedural FBs that can select the transmission speed of 300, 600, 76800 and 115200 bps are "_Cfrpr" and "_Cfrp2".
V2535 or later version of general purpose communication module can use the "_Cfrpr" FB; V2536 or later version of general purpose interface module can use the "_Cfrp2" FB.

3-1 Communication Protocol Included Non-procedural FB "_Cfrp2"

(2) FB format



* ( ) indicates data type.

Note: In general, FBs are used with only the terminals connected that are necessary for communication. Be sure to connect variables to the IN-OUT terminals.

(3) Explanation for FB terminals

Terminal name	Terminal name	Data type	I/O	Description
Open	OPEN	BOOL	IN	ON: Sends the "communication parameters" to the general purpose communication module to initialize communication. When communication becomes possible, "communication ready" turns ON. OFF: Performs the processing for ending communication. "Communication ready" turns OFF.
Send request	S_REQ	BOOL	IN	Starts to send data. When sending has ended, this needs to be turned OFF by the application program.
Send data length	S_LEN	INT	IN	Designates the length of send data by the number of bytes.
Send data	S_DAT	_C_SND_TYP 11 PR2	IN_OUT	Stores send data.
Communication parameter	PARA	_C_PAR_TYP 11 PR2	IN_OUT	Stores the initialization parameters.
Communication ready	OK	BOOL	OUT	Turns ON when initialization has ended normally to indicate that the system is ready for sending/receiving data.
Open status	O_STS	WORD	OUT	2-digit hexadecimal code to indicate the result of initialization.
Send end	S_END	BOOL	OUT	Turns ON when sending has completed.
Send error	S_ERR	BOOL	OUT	Turn ON if a send error has occurred.
Send status	S_STS	WORD	OUT	Code to indicate the result of sending
Receive end	R_END	BOOL	OUT	Turns ON when receiving has completed.
Receive error	R_ERR	BOOL	OUT	Turns ON if a receive error has occurred.
Receive status	R_STS	WORD	OUT	Code to indicate the result of receiving
Receive data length	R_LEN	INT	OUT	Outputs the received data length.
Receive data	R_DAT	_C_RCV_TYP 11 PR2	IN_OUT	Stores received data.
RS-485 station No.	STN_NO	INT	OUT	The condition of the RS-485 station number setup switch of the general purpose communication module is output to this pin.
RAS information	RAS	_C_work_TYPE 11 PR2	IN_OUT	Operating information of this FB.

<Format of the variables that are connected to the "send data" terminal (S_DAT)>

The data that are connected to "S_DAT" must be laid out as follows:

<When variable name is "S_DAT">

```
VAR
    S_DAT:_C_SND_TYP11PR2;
END_VAR
```

S_DAT [0]
S_DAT [1]
⋮
S_DAT [511]

	Upper order	Lower order
	Data 2	Data 1
	Data 4	Data 3
	⋮	⋮
	Data 1024	Data 1023

* The size of send data is maximum 1024 bytes including start code, end code and BCC when code conversion is not to be performed.

<Format of the variables that are connected to the "Receive data" terminal (R_DAT)>

The data that are connected to "R_DAT" must be laid out as follows:

<When variable name is "R_DAT">

```
VAR
    R_DAT:_C_RCV_TYP11PR2;
END_VAR
```

R_DAT [0]
R_DAT [1]
⋮
R_DAT [511]

	Upper order	Lower order
	Data 2	Data 1
	Data 4	Data 3
	⋮	⋮
	Data 1024	Data 1023

* The size of receive data is maximum 1024 bytes including start code, end code and BCC when code conversion is not to be performed.

3-1-3 Initialization

(1) Initialization parameters

In order to initialize each individual port of the general purpose communication module, it is necessary to set values which match the communication specifications of the object external device to the individual communication parameters. The table below shows the setting contents of these parameters.

No.	Item		Description
0	General purpose communication module station No.		Sets the station number of the general purpose communication module on the SX bus.
1	Port No.		Designates an interface port on general purpose communication module. 0: RS-232C port 1: RS-485 port
2	Message port No. 1		Designates a port No. for sending/receiving messages to/from a general purpose communication module (1 to 127). Note: Avoid designating a port No. which is already used for sending/receiving messages to/from other module.
3	Message port No. 2		Designates a port No. for sending/receiving messages to/from a general purpose communication module (1 to 127). Note: Be careful not to duplicately set this port with message port No. 1 or other message sending or receiving port numbers.
4	Transmission speed		Designates the transmission speed. 0: 1200 1: 2400 2: 4800 3: 9600 4: 19200 5: 38400 6: 57600 7: 76800 8: 115200 90: 300 91: 600 bps
5	Data bit		Designates the data bit length. When "7" is selected, 7 bits make up one data; when "8," 8 bits make up one data. 0: 7 bits 1: 8 bits
6	Parity bit		This is a bit which is added to data for the purpose of error detection. Designate the proper one to match the setting of the device at the other end. 0: None 1: Odd 2: Even
7	Stop bit		This bit indicates the end of data. Designate the proper one to match the setting of the device at the other end. 0: 1 bit 2: 2 bits
8	DCE designation		When signal line control is not performed, there is no difference in functions between DCE and DTE modes. The RS-232C of the general purpose communication module is of DTE specification, however, it can be used as a DCE specification interface when signal lines are converted as follows: No. 7 pin (RS) to CS No. 8 pin (CS) to RS No. 6 pin (DR) to ER No. 4 pin (ER) to DR 0: DTE 1: DCE 2: Modem DTE
9	ER/DR signal control		0: None
10	Signal flow control	DTE mode	0: Off RS: always ON; Sending: unconditional 1: On RS: ON while sending; Sending: when CS is ON
		DCE mode	0: Off CS: always ON; Sending: unconditional 1: On CS: ON when RS is ON; Sending: when ER is ON
11	XON/XOFF control		Because communication between sender and receiver is performed asynchronously, flow control may be necessary. The receiver sends an XOFF signal to indicate that it cannot receive data for a while and then sends an XON signal to cancel the XOFF condition. To use XON/OFF control, the device at the other end must also have this function. 0: NO 1: YES
12	RS-485 mode		Selects 4-wire or 2-wire for RS-485. 0: 4-wire 1: 2-wire
13	Reserved		Not used
14	Reserved		
15	Reserved		
16	Code conversion		Converts binary data into a character string variable. 0: None 1: ASCII conversion 2: EBCDIC conversion

No.	Item	Description
17	Frame detection	Designates data receiving method 0: None Receiving completes when data is received. 1: Variable length Receiving completes when the data in a range between start and end codes is received. 2: Fixed length Receiving completes when receive data reaches the specified number of receive data bytes.
18	Number of receive data bytes	Designates the number of receive data bytes when Fixed length is selected for frame detection. When Variable length is selected, set this item to 0.
19	Number of start code bytes	Designates the number of start code bytes when "Variable length" is selected.
20	Start code 1	Designates start code when Variable length is selected.
21	Start code 2	
22	Start code 3	
23	Start code 4	
24	Start code 5	
25	Number of end code bytes	Designates the number of end code bytes when Variable length is selected.
26	End code 1	Designates end code when Variable length is selected.
27	End code 2	
28	End code 3	
29	End code 4	
30	End code 5	
31	BCC designation	Sets whether or not to add horizontal parity for checking text data transmission error. 0: None 1: Set in the order of upper- and lower-order byte [Upper byte of BCC][Lower byte of BCC] 2: Set in the order of lower- and upper-order byte [Lower byte of BCC][Upper byte of BCC]
32	Calculation range, position	Sets the calculation range and BCC position. ◄——► : Calculation range 0: Calculates the text part and enters it ahead of the end code. [Start code] [TEXT] [BCC] [End code] (Note) 1: Calculates text part and end code and inserts them behind the end code. [Start code] [TEXT] [End code] [BCC] 2: Calculates start code and text part and inserts them ahead of the end code. [Start code] [TEXT] [BCC] [End code] (Note) 3: Calculates start code, text part and end code and inserts them behind the end code. [Start code] [TEXT] [End code] [BCC] Note: In this case, the BCC code type cannot be designated as binary. Specify other than CRC-16 for the BCC expression.

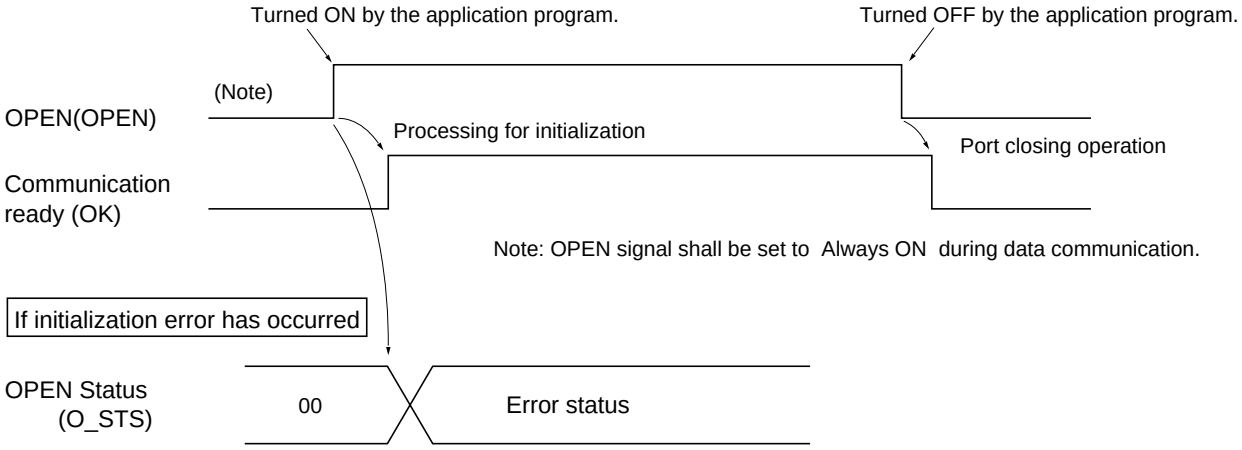
No.	Item	Description
33	BCC calculation formula	Calculation method to check for transmission error.  0: Add D1 + D2 + ... + Dn 1: Add and invert Inversion of (D1 + D2 + ... + Dn) 2: EOR D1 EOR D2 EOR ... EOR Dn 3: CRC * When CRC is selected, specify BCC Code Format to 0: Binary. CRC-16: $X^{16}+X^{15}+X^{2}+1$
34	BCC code type	Designates the code type of BCC data. 0: Binary 1: ASCII 2: EBCDIC
35	Send timer value	The timer value for monitoring transmission since the data send request has been sent from the CPU module onto the RS-232C line until sending is completed. This value can be set in 0.01-second steps. Example: For 1 second, set this value to 100 . Maximum 327.67 seconds can be set.
36 : 39	Reserved	Not used

Note 1: Parameter Nos.16 to 35 are valid even if changed after opening.
After opening, if parameter Nos.16 to 35 are changed and a setup error occurs, or if other parameters are changed, an open error results and communication "OK" is turned OFF.

Note 2: No. in the above table is element number of the array variable for communication parameter.

(2) Initialization procedure

When the OPEN terminal of FB is turned ON, communication parameters are transferred to the general purpose communication module, which executes processing for initialization. If initialization is impossible due to a communication parameter setting error or hardware error, FB outputs the corresponding error code to the OPEN Status (O_STS).

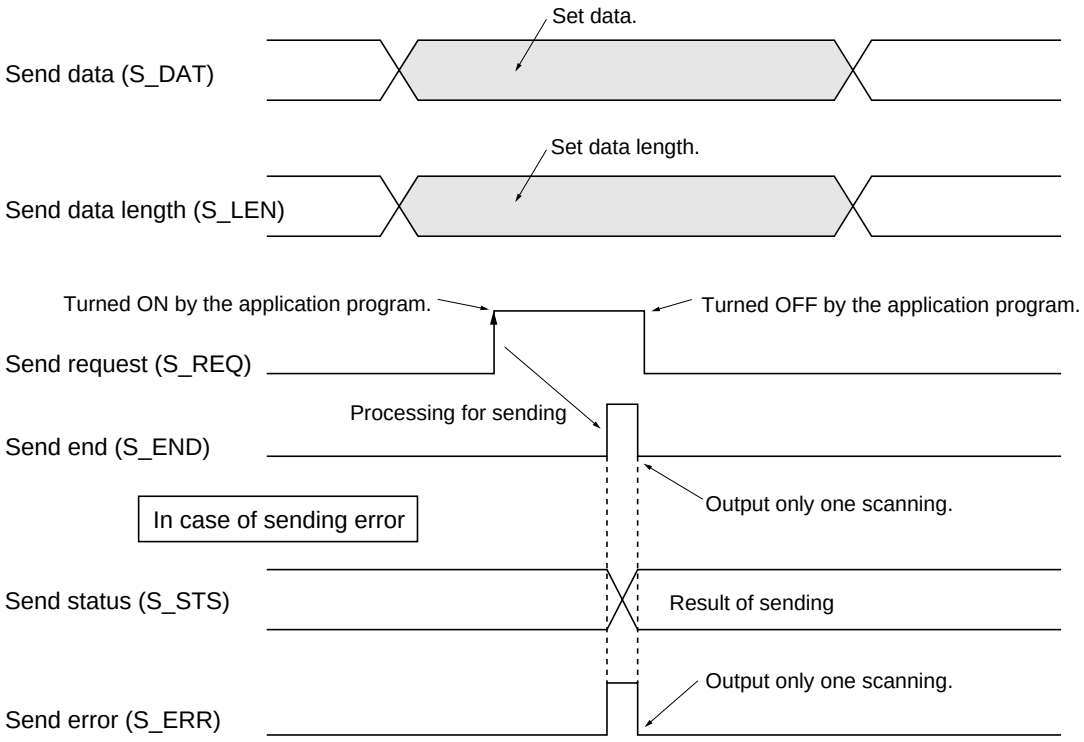


(3) OPEN status list

No.	Result of initialization	Remarks
'00'	Ended normally	
'01'	Transmission speed error	General purpose communication module detected an error.
'02'	Data bit length error	General purpose communication module detected an error.
'03'	Parity bit error	General purpose communication module detected an error.
'04'	Stop bit error	General purpose communication module detected an error.
'05'	DCE designation error	General purpose communication module detected an error.
'06'	Signal flow control setting error	General purpose communication module detected an error.
'07'	XON/XOFF control setting error	General purpose communication module detected an error.
'08'	RS-485 mode setting error	General purpose communication module detected an error.
'3F'	Parameter modification error	Parameter modification error after OPEN
'40'	Code conversion setting error	
'41'	Frame detecting method setting error	
'42'	Start code setting error	
'43'	End code setting error	
'44'	BCC code setting error	
'45'	BCC calculation range setting error	
'46'	BCC calculation formula setting error	
'47'	BCC code type setting error	
'48'	Transmission timer specify setting error	Do not specify a minus value for the send timer
'80'	General purpose communication module station No. setting error	
'81'	Channel No. setting error	
'82'	Message port No. setting error	
'93'	Open error	M_OPEN error
'94'	Open error	Initialization failed due to abnormality on general purpose communication module
'A3'	Processing impossible because in code conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.

3-1-4 Data sending

(1) Data sending procedure



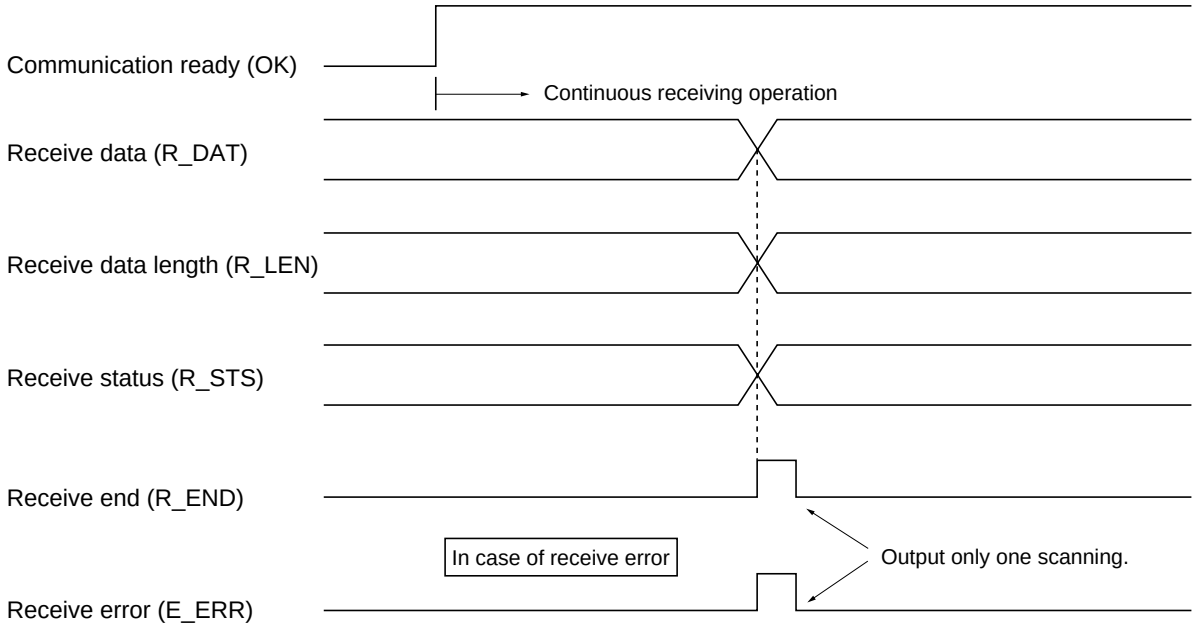
After setting send data and its data length, turn S_REQ (send request) ON by the application program. Then FB detects the rising edge to execute the processing for sending.
When sending has ended, S_END (send end) flag is turned ON (for only one pulse). If a send error has occurred, S_END (send end) and S_ERR (send error) are turned ON (for only one pulse), and the corresponding error code is output to S_STS (send status) ("00" when ended normally).

(2) Send status list

No.	Result of sending	Remarks
'00'	Normally ended	
'01'	Send buffer has overflowed.	
'40'	Data send time-out	
'90'	General purpose communication module disconnected.	Not detected
'91'	SX bus send error	
'92'	SX bus receive error	Not detected
'A0'	Object port error	Specified port No. is neither 0 nor 1.
'A3'	Processing impossible because in code conversion mode.	
'A5'	Processing impossible because in loader mode.	
'A6'	Processing impossible because self-diagnosis is being executed.	Error detection on general purpose communication module
'C2'	Send buffer has overflowed.	Error detection on general purpose communication module
'C3'	Send data size over	Error detection on general purpose communication module

3-1-5 Data receiving

(1) Data receiving procedure



While the communication ready flag is turned ON, whether the data to be received exists or not is checked continuously and, when it exists, the operation for receiving is executed. When a delimiter between data frames is detected, the received data and the number of receive data bytes are stored in R_DAT (receive data) and R_LEN (receive data length), respectively, and the R_END (receive end) flag is turned ON (for only one pulse). The result of receiving is stored in R_STS (receive status). If a receive error has occurred, R_END (receive end) and E_ERR (receive error) are turned ON (for only one pulse).

(2) Receiving status list

No.	Result of receiving	Remarks
'00'	Ended normally	
'02'	Receive buffer has overflowed.	
'42'	BCC error	
'90'	General purpose communication module disconnected.	
'91'	SX bus send error	
'92'	SX bus receive error	
'A0'	Object port error	General purpose communication module detected an error.
'A3'	Processing impossible because in code conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.
'C0'	Hardware error detected during receiving (Parity error, Franming error, etc.)	General purpose communication module detected an error.
'C1'	Receive buffer has overflowed.	General purpose communication module detected an error.

Note: Receiving conditions
The processing for receiving differs with the setting of parameter "Frame detection."
None: Receiving completes when data is received.
Fixed length: Receiving completes when receive data reaches the specified number of receive data bytes.
Variable length: Receiving completes when the data in a range between start and end codes is received.

3-1-6 RAS information

RAS information area is used as the work memory for FB operations as well as for checking the operating condition of FBs. To use RAS information from the application program, it is necessary to declare variables in the following manner:

<Example of variable declaration>

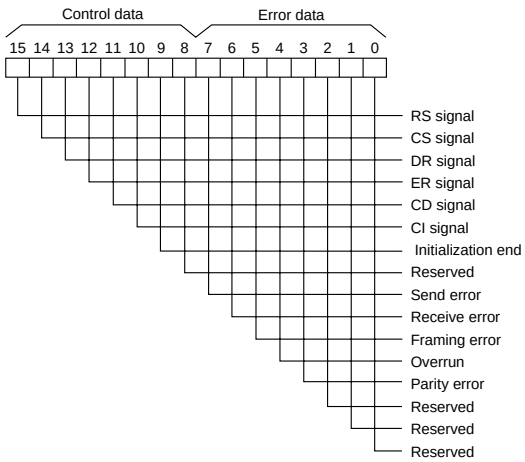
```
VAR
    RAS:     _C_work_TYPE11PR2;
END_VAR
```

<RAS information area (INT array)>

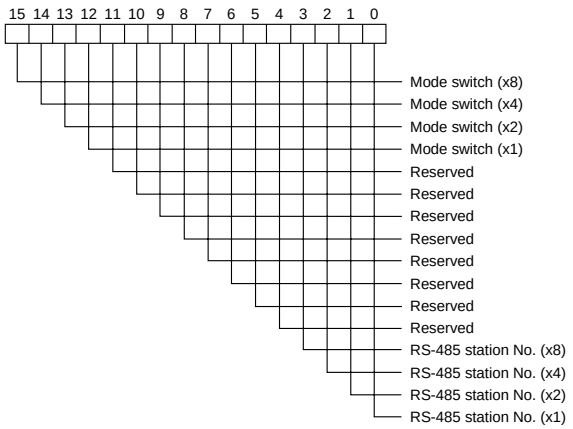
For the variable declaration shown above, the array variable in which RAS information is stored becomes "RAS". Individual element stores the following RAS information.

RAS.RAS[0]	Port status
RAS.RAS[1]	Status of general purpose communication module
RAS.RAS[2]	Send request count
RAS.RAS[3]	Send end count
RAS.RAS[4]	Receive count
RAS.RAS[5]	Frame detection count
RAS.RAS[6]	M_OPEN status
RAS.RAS[7]	M_SEND status
RAS.RAS[8]	M_RECEIVE status
RAS.RAS[9]	M_SEND error count
RAS.RAS[10]	M_RECEIVE error count
RAS.RAS[11]	Reserved
RAS.RAS[12]	Reserved
RAS.RAS[13]	Reserved
RAS.RAS[14]	Reserved
RAS.RAS[15]	Reserved
RAS.RAS[16]	Reserved
RAS.RAS[17]	Reserved
RAS.RAS[18]	Reserved
RAS.RAS[19]	Reserved

• Port status



• Status of general purpose communication module



3-2 FUJI General Purpose Inverter Procedure FB "_CfdFRN"

3-2-1 General

The "_CfdFRN" FB (function block) communicates data between the CPU module and FUJI FRENIC Series general purpose inverter using the start-stop synchronization type transmission protocol. For more information of the dedicated transmission parameters that are necessary to control the FRENIC 5000 Series general purpose inverter, refer to the manual supplied with the inverter.
This FB performs the following operations:

1) Initialization of communication port
Initializes the RS-485 port (the setting of transmission speed, data bit length, parity bit, etc.)

2) Data sending/receiving function
Sends data from an application program in the MICREX-SX series CPU module to general purpose inverter (FRENIC 5000 series) via the general purpose communication module, or outputs the data received via the general purpose communication module to an application program.

3) Transmission condition monitoring function
Monitors data transmission condition and outputs error information if abnormal.

<Connecting method of "_CfdFRN">
The connecting method between the general purpose communication module and FUJI FRENIC 5000 series general purpose inverter must be "1:N connection" by RS-485 (2-wire system).

<FB operating conditions>

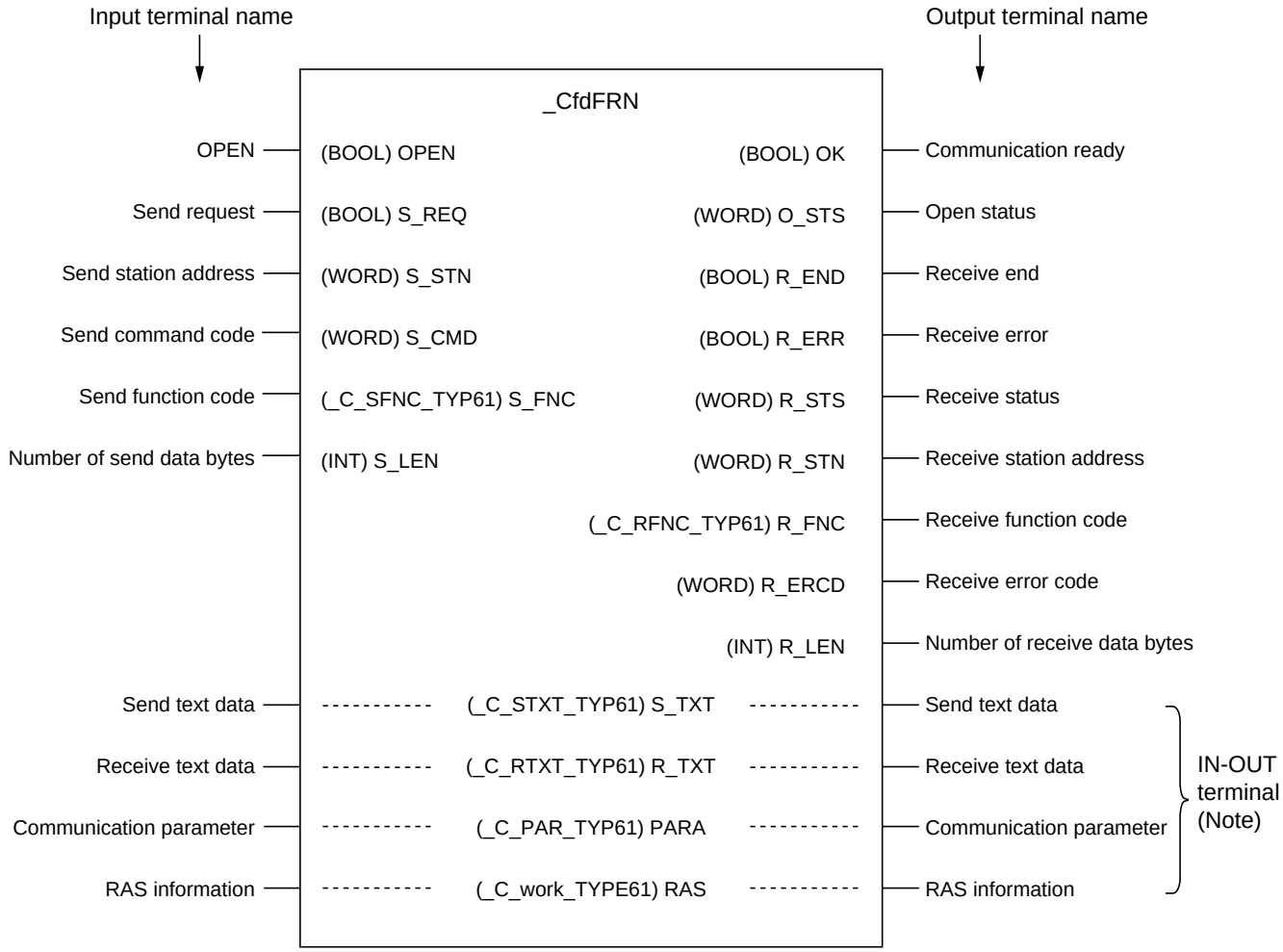
System configuration:
- One FB is necessary for each communication port (RS-485) to be used. Communication with the inverters that are connected to one RS-485 system is performed by one FB.
- FUJI general purpose inverter may not be connected to the RS-232C port.
- This FB cannot be used in the mode where RS-232C/RS-485 signal conversion is enabled for the general purpose communication module.

Memory capacity:
Program area = 2046 steps
Data memory capacity = Instance memory for user FBs (280 words)
+ Standard memory (854 words)
+ Instance memory for system FBs (90 words)

Notes:
1) Above figures for memory capacity includes the area for the main body of non-procedural FB and those for sub-FBs that are called from the non-procedural FB.
2) Above figures for standard memory and retain memory includes the memory capacity necessary for sending/receiving data.

3-2-2 Specifications for _CfdFRN

(1) FB format



* ( ) indicates data type.

Note: In general, FBs are used with only the terminals connected that are necessary for communication. Be sure to connect variables to the IN-OUT terminals.

(2) Explanation of each FB terminal

Terminal name	Variable name	Data type	I/O	Description
Open	OPEN	BOOL	IN	ON: Sends "communication parameters" to the general purpose communication module to initialize communication. When communication becomes possible, "communication ready" turns ON. OFF: Performs the processing for ending communication. "Communication ready" turns OFF.
Send request	S_REQ	BOOL	IN	Starts to send data. When sending is completed, this needs to be turned OFF by the application program.
Send station address	S_STN	WORD	IN	Stores the address of the send station.
Send command code	S_CMD	WORD	IN	Stores the send command code.
Send function code	S_FNC	_C_SFNC _TYP61	IN_OUT	Stores the send function code.
Number of send data bytes	S_LEN	INT	IN	Stores the number of send data bytes.
Send text data	S_TXT	_C_STXT _TYP61	IN_OUT	Stores the send text data.
Communication parameter	PARA	_C_PAR _TYP61	IN_OUT	Stores the initialization parameters.
Communication ready	OK	BOOL	OUT	Turns ON when initialization has ended normally to indicate that the system is ready for sending/receiving data.
Open status	O_STS	WORD	OUT	2-digit hexadecimal code to indicate the result of initialization
Receive end	R_END	BOOL	OUT	Turns ON when receive ends.
Receive error	R_ERR	BOOL	OUT	Turns ON if a receive error has occurred.
Receive status	R_STS	WORD	OUT	Receive result code
Receive station address	R_STN	WORD	OUT	Stores the address of the receive station.
Receive function code	R_FNC	_C_RFNC _TYP61	IN_OUT	Stores the receive function code.
Receive error code	R_ERC-D	WORD	OUT	Stores the receive error code.
Number of receive data bytes	R_LEN	INT	OUT	Stores the number of receive data bytes.
Receive text data	R_TXT	_C_RTXT _TYP61	IN_OUT	Stores receive text data.
RAS information	RAS	_C_work _TYPE61	IN_OUT	Stores the operating status of this FB.

3-2-3 Initialization

(1) Set values of initialization parameters

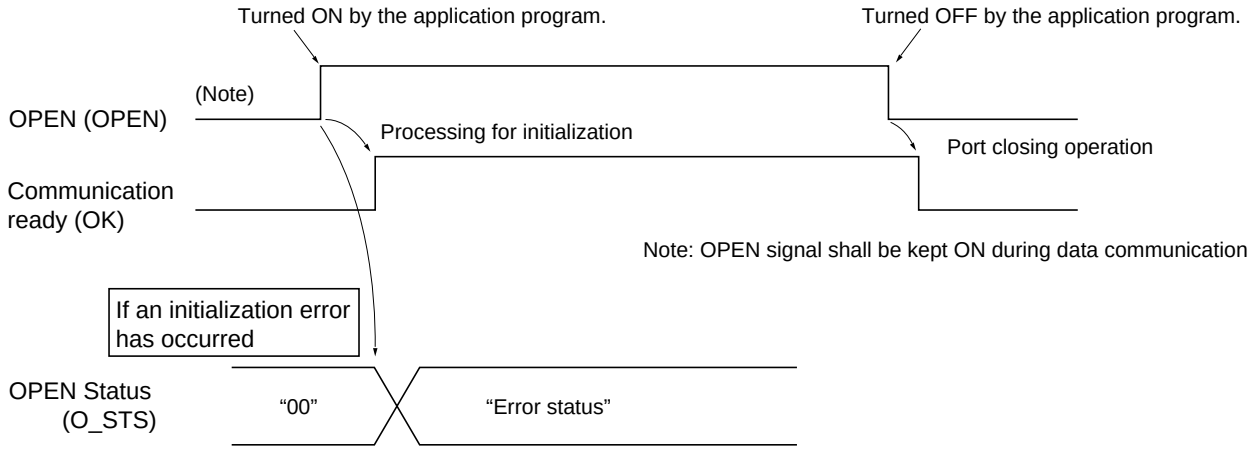
In order to initialize (RS-485) communication ports which are to be used, it is necessary to set proper values for each "communication parameter" item so as to match the communication specifications of the FVR-9 series general purpose inverter. The table below shows the setting contents of these parameters.

No.	Item	Description
0	General purpose communication module station No.	Sets the station number of the general purpose communication module on the SX bus.
1	Port No.	Designates an interface port on the general purpose communication module. 1: RS-485 port
2	Message port No.	Designates a port No. for sending/receiving message to/from a general purpose communication module.
3	Reserved	Not used
4	Transmission speed	Designates the transmission speed. 0: 1200 1: 2400 2: 4800 3: 9600 4: 19200 bps
5	Data bit	Designates the data bit length. When "7" is selected, 7 bits make up one data; when "8", 8 bits make up one data. 0: 7 bits (when ASCII code is used) 1: 8 bits (when EBCDIC code is used)
6	Parity bit	This is a bit which is added to data for the purpose of error detection. Designate the proper one to match the setting of the device at the other end. 0: None 1: Odd 2: Even
7	Stop bit	This bit indicates the end of data. Designate the proper one to match the setting of the device at the other end. 0: 1 bit 2: 2 bits
8 ⋮ 15	Reserved	Not used
16	Response monitoring timer	The timer for monitoring during the period from when the CPU module sends a send request to a device until communication ends. This value can be set in 0.01-second steps. Example: For 1 second, set this value to "100".
17	Retry count	In case of communication error, this designates how many times to retry communication.
18 ⋮ 39	Reserved	Not used

Note: No. in the above table is element number of the array variable for communication parameter.

(2) Initialization procedure

When the OPEN terminal of FB is turned ON, communication parameters are transferred to the general purpose communication module, which executes initialization. If initialization is impossible due to communication parameter setting error or hardware error, FB outputs the corresponding error code to O_STS (OPEN status).

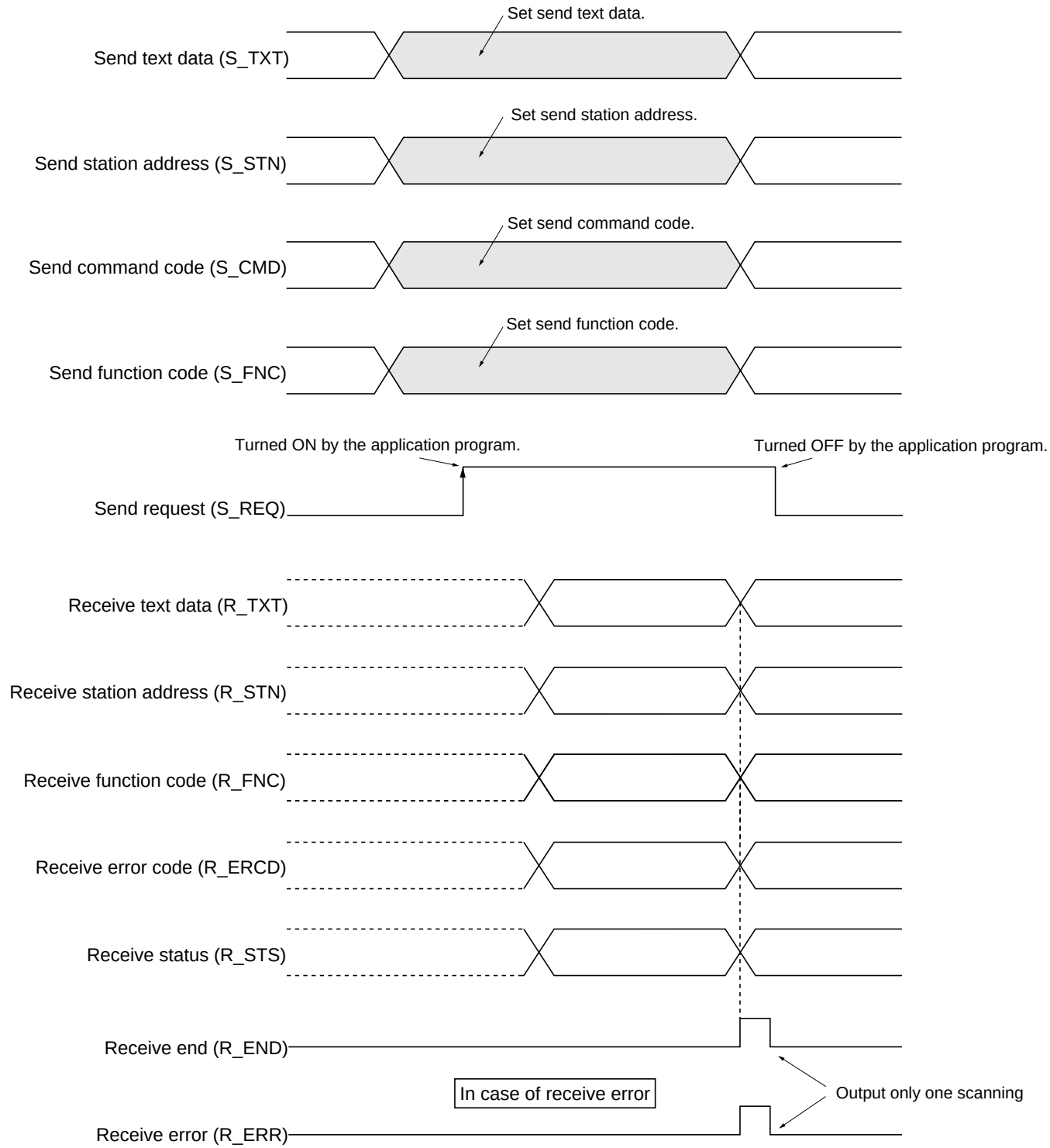


(3) OPEN status list

No.	Result of initialization	Remarks
'00'	Ended normally	
'01'	Transmission speed error	General purpose communication module detected an error.
'02'	Data bit length error	General purpose communication module detected an error.
'03'	Parity bit error	General purpose communication module detected an error.
'04'	Stop bit error	General purpose communication module detected an error.
'05'	Reserved	
'06'	Reserved	
'07'	Reserved	
'3F'	Parameter changeover error	Parameter modification after OPEN detected
'40'	Response monitoring timer value setting error	
'41'	Retry count setting error	
'42'	Reserved	
'43'	Reserved	
'80'	General purpose communication module station No. setting error	
'81'	Port No. setting error	
'82'	Message port No. setting error	
'93'	Open error	Initialization failed due to SX bus error
'94'	Open error	Initialization failed due to abnormality on general purpose communication module
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because selfdiagnosis is being executed.	General purpose communication module detected an error.

3-2-4 Data sending/receiving

(1) Data sending/receiving procedure



Send text data, send station address, send command code and send function code are set and then send request is turned ON by the application program. FB detects the rising edge to execute the processing for sending.
When the send station address is for all stations (= 90), the receive end flag is automatically turned ON the moment sending is completed (receive error flag is also turned ON if abnormal).
When the send station address is for one station (≠ 90), the system automatically enters receiving waiting mode the moment sending is completed, executing the processing for receiving. When the delimiter between frames is detected, the receive end flag is automatically turned ON (receive error flag is also turned ON if abnormal).

(2) Sending/receiving status list

No.	Result of sending/receiving	Remarks
'00'	Ended normally	
'01'	Send buffer has overflowed	
'02'	Receive buffer has overflowed	
'40'	Data send time-out	
'41'	Protocol response error	
'42'	BCC error	
'90'	General purpose communication module disconnected.	Not detected
'91'	SX bus send error	
'92'	SX bus receive error	
'A0'	Object port error	General purpose communication module detected an error.
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.
'C0'	Hardware error detected during receiving	General purpose communication module detected an error.
'C1'	Receive buffer has overflowed.	General purpose communication module detected an error.
'C2'	Send buffer has overflowed.	General purpose communication module detected an error.
'C3'	Send data size over	General purpose communication module detected an error.

3-2-5 RAS information

RAS information area is used as the work memory for FB operations as well as for checking the operating condition of FBs. To use RAS information from the application program, it is necessary to declare variables in the following manner:

<Example of variable declaration>

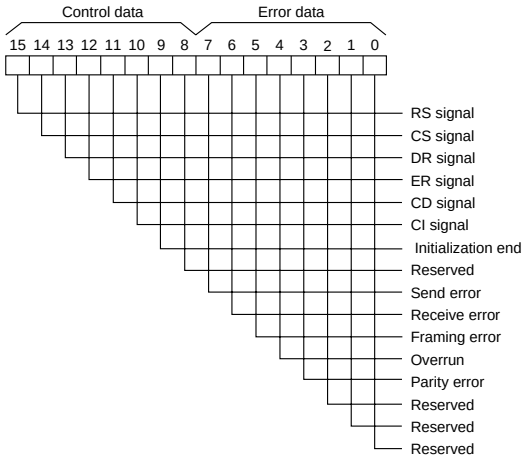
```
VAR
    RAS:    _C_work_TYPE61;
END_VAR
```

<RAS information area (INT array)>

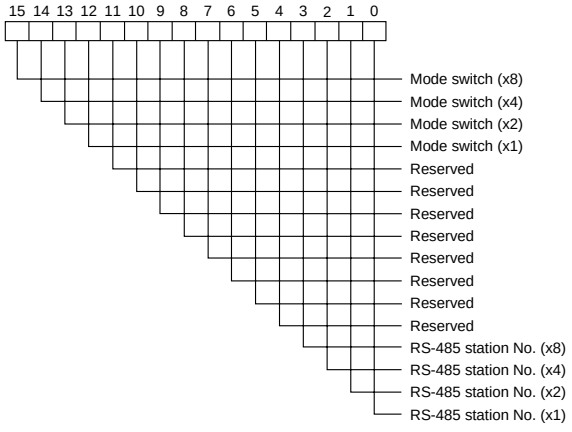
For the variable declaration shown above, the array variable in which RAS information is stored becomes "RAS". Individual element stores the following RAS information.

RAS.RAS[0]	Port status
RAS.RAS[1]	Status of general purpose communication module
RAS.RAS[2]	Send request count
RAS.RAS[3]	Send end count
RAS.RAS[4]	Receive count
RAS.RAS[5]	Frame detection count
RAS.RAS[6]	M_OPEN status
RAS.RAS[7]	M_SEND status
RAS.RAS[8]	M_RECEIVE status
RAS.RAS[9]	M_SEND error count
RAS.RAS[10]	M_RECEIVE error count
RAS.RAS[11]	Reserved
RAS.RAS[12]	Reserved
RAS.RAS[13]	Reserved
RAS.RAS[14]	Reserved
RAS.RAS[15]	Reserved
RAS.RAS[16]	Reserved
RAS.RAS[17]	Reserved
RAS.RAS[18]	Reserved
RAS.RAS[19]	Reserved

• Port status



• Status of general purpose communication module



3-3 FUJI General Purpose Inverter Standard Communication (FGI-BUS) Procedure FB "_CfdFVR"

3-3-1 General

The "_CfdFVR" FB (function block) communicates data between the CPU module and FUJI FVR Series general purpose inverter using the "FGI-BUS" transmission protocol. For more information of the dedicated transmission parameters that are necessary to control the FVR Series general purpose inverter, refer to the manual supplied with the inverter.
This FB performs the following operations:

1) Initialization of communication port
Initializes the RS-485 port (the setting of transmission speed, data bit length, parity bit, etc.)

2) Data sending/receiving function
Sends data from an application program in the MICREX-SX series CPU module to general purpose inverter (FVR) via the general purpose communication module, or outputs the data received via the general purpose communication module to an application program.

3) Transmission condition monitoring function
Monitors data transmission condition and outputs error information if abnormal.

<Connecting method of "_CfdFVR">
The connecting method between the general purpose communication module and FUJI FVR series general purpose inverter must be "1:N connection" by RS-485 (2-wire system).

<FB operating conditions>

System configuration:
- One FB is necessary for one communication port to be used (RS-485). The control of all the inverters that are connected to one RS-485 system is performed by one FB.
- FUJI general purpose inverter may not be connected to the RS-232C port.
- This FB cannot be used in the mode where RS-232C/RS-485 signal conversion is enabled for the general purpose communication module.

Memory capacity:
Program area = 2015 steps
Data memory capacity = Instance memory for user FBs (278 words)
+ Standard memory (794 words)
+ Instance memory for system FBs (90 words)

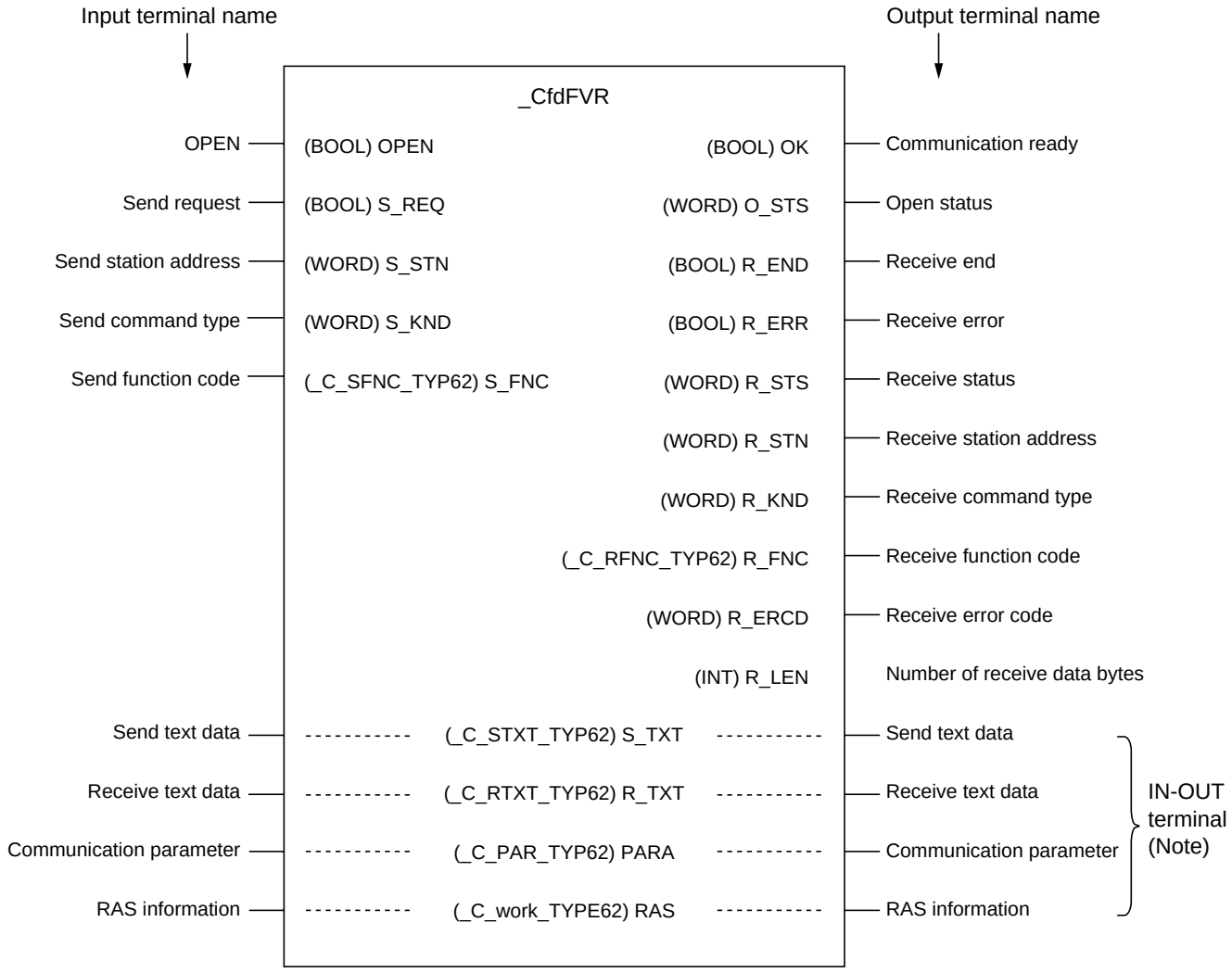
Notes:
1) Above figures for memory capacity includes the area for the main body of non-procedural FB and those for sub-FBs that are called from the non-procedural FB.
2) Above figures for standard memory and retain memory includes the memory capacity necessary for sending/receiving data.

* FGI-BUS: Fuji Electric General Inverter's Serial protocol

3-3 FUJI General Purpose Inverter Standard Communication (FGI-BUS) Procedure FB "_CfdFVR"

3-3-2 Specifications for _CfdFVR

(1) FB format



* ( ) indicates data type.

Note: In general, FBs are used with only the terminals connected that are necessary for communication. Be sure to connect variables to the IN-OUT terminals.

(2) Explanation of each FB terminal

Terminal name	Variable name	Data type	I/O	Description
Open	OPEN	BOOL	IN	ON: Sends "communication parameters" to the general purpose communication module to initialize communication. When communication becomes possible, "communication ready" turns ON. OFF: Performs the processing for ending communication. "Communication ready" turns OFF.
Send request	S_REQ	BOOL	IN	Starts to send data. When sending is completed, this needs to be turned OFF by the application program.
Send station address	S_STN	WORD	IN	Stores the address of the send station.
Send command type	S_KND	WORD	IN	Stores the send command type.
Send function code	S_FNC	_C_SFNC _TYP62	IN_OUT	Stores the send function code.
Send text data	S_TXT	_C_STXT _TYP62	IN_OUT	Stores the send text data.
Communication parameter	PARA	_C_PAR _TYP62	IN_OUT	Stores the initialization parameters.
Communication ready	OK	BOOL	OUT	Turns ON when initialization has ended normally to indicate that the system is ready for sending/receiving data.
Open status	O_STS	WORD	OUT	2-digit hexadecimal code to indicate the result of initialization
Receive end	R_END	BOOL	OUT	Turns ON when receive ends.
Receive error	R_ERR	BOOL	OUT	Turns ON if a receive error has occurred.
Receive status	R_STS	WORD	OUT	Receive result code
Receive station address	R_STN	WORD	OUT	Stores the address of the receive station.
Receive command type	R_KND	WORD	OUT	Stores the receive command type.
Receive function code	R_FNC	_C_RFNC _TYP62	IN_OUT	Stores the receive function code.
Receive error code	R_ERCD	WORD	OUT	Stores the receive error code.
Number of receive data bytes	R_LEN	INT	OUT	Stores the number of receive data bytes.
Receive text data	R_TXT	_C_RTXT _TYP62	IN_OUT	Stores receive text data.
RAS information	RAS	_C_work _TYPE62	IN_OUT	Stores the operating status of this FB.

3-3-3 Initialization

(1) Set values of initialization parameters

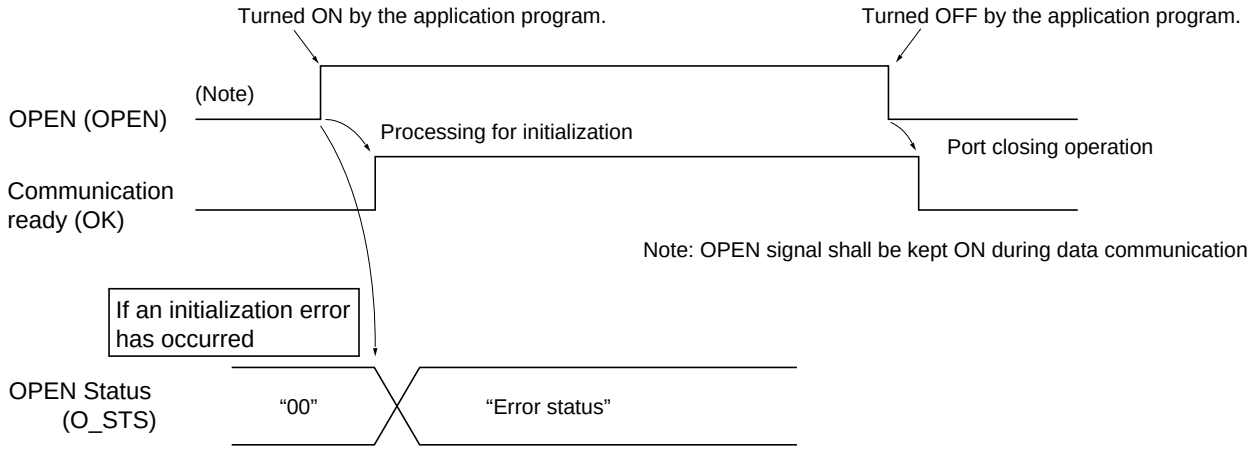
In order to initialize (RS-485) communication ports which are to be used, it is necessary to set proper values for each "communication parameter" item so as to match the communication specifications of the FVR-11 series general purpose inverter. The table below shows the setting contents of these parameters.

No.	Item	Description
0	General purpose communication module station No.	Sets the station number of the general purpose communication module on the SX bus.
1	Port No.	Designates an interface port on the general purpose communication module. 1: RS-485 port
2	Message port No.	Designates a port No. for sending/receiving message to/from a general purpose communication module.
3	Reserved	Not used
4	Transmission speed	Designates the transmission speed. 0: 1200 1: 2400 2: 4800 3: 9600 4: 19200 bps
5	Data bit	Designates the data bit length. When "7" is selected, 7 bits make up one data; when "8", 8 bits make up one data. 0: 7 bits 1: 8 bits
6	Parity bit	This is a bit which is added to data for the purpose of error detection. Designate the proper one to match the setting of the device at the other end. 0: None 1: Odd 2: Even
7	Stop bit	This bit indicates the end of data. Designate the proper one to match the setting of the device at the other end. 0: 1 bit 2: 2 bits
8 ⋮ 15	Reserved	Not used
16	Response monitoring timer	The timer for monitoring during the period from when the CPU module sends a send request to a device until communication ends. This value can be set in 0.01-second steps. Example: For 1 second, set this value to "100".
17	Retry count	In case of communication error, this designates how many times to retry communication.
18 ⋮ 39	Reserved	Not used

Note: No. in the above table is element number of the array variable for communication parameter.

(2) Initialization procedure

When the OPEN terminal of FB is turned ON, communication parameters are transferred to the general purpose communication module, which executes initialization. If initialization is impossible due to communication parameter setting error or hardware error, FB outputs the corresponding error code to O_STS (OPEN status).

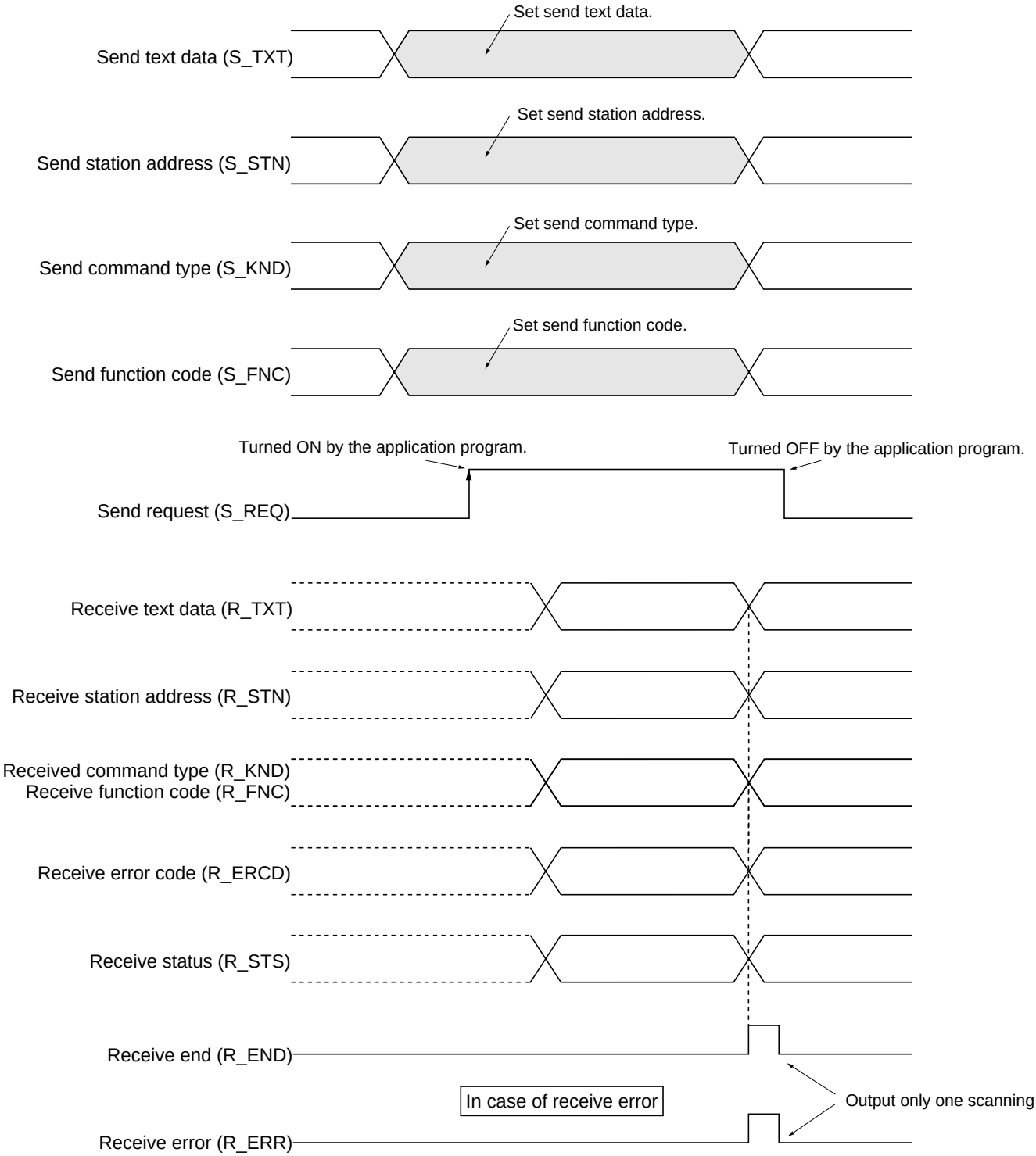


(3) OPEN status list

No.	Result of initialization	Remarks
'00'	Ended normally	
'01'	Transmission speed error	General purpose communication module detected an error.
'02'	Data bit length error	General purpose communication module detected an error.
'03'	Parity bit error	General purpose communication module detected an error.
'04'	Stop bit error	General purpose communication module detected an error.
'3F'	Parameter changeover error	Parameter modification after OPEN detected
'40'	Response monitoring timer value setting error	
'41'	Retry count setting error	
'80'	General purpose communication module station No. setting error	
'81'	Port No. setting error	
'82'	Message port No. setting error	
'93'	Open error	Initialization failed due to SX bus error
'94'	Open error	Initialization failed due to abnormality on general purpose communication module
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because selfdiagnosis is being executed.	General purpose communication module detected an error.

3-3-4 Data sending/receiving

(1) Data sending/receiving procedure



Send text data, send station address, send command code and send function code are set and then send request is turned ON by the application program. FB detects the rising edge to execute the processing for sending.
When the send station address is for all stations (= 99), the receive end flag is automatically turned ON the moment sending is completed (receive error flag is also turned ON if abnormal).
When the send station address is for one station (≠ 99), the system automatically enters receiving waiting mode the moment sending is completed, executing the processing for receiving. When the delimiter between frames is detected, the receive end flag is automatically turned ON (receive error flag is also turned ON if abnormal).

(2) Sending/receiving status list

No.	Result of sending/receiving	Remarks
'00'	Ended normally	
'01'	Send buffer has overflowed	
'02'	Receive buffer has overflowed	
'40'	Data send time-out	
'41'	Protocol response error	
'42'	BCC error	
'90'	General purpose communication module disconnected.	Not detected
'91'	SX bus send error	
'92'	SX bus receive error	
'93'	NAK recieve	
'A0'	Object port error	General purpose communication module detected an error.
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.
'C0'	Hardware error detected during receiving	General purpose communication module detected an error.
'C1'	Receive buffer has overflowed.	General purpose communication module detected an error.
'C2'	Send buffer has overflowed.	General purpose communication module detected an error.
'C3'	Send data size over	General purpose communication module detected an error.

3-3 FUJI General Purpose Inverter Standard Communication (FGI-BUS) Procedure FB "_CfdFVR"

3-3-5 RAS information

RAS information area is used as the work memory for FB operations as well as for checking the operating condition of FBs. To use RAS information from the application program, it is necessary to declare variables in the following manner:

<Example of variable declaration>

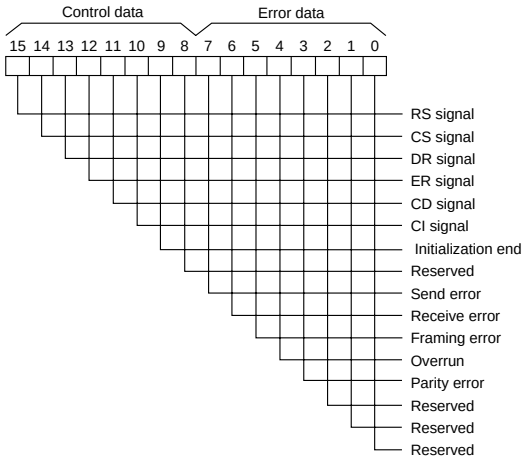
```
VAR
    RAS:     _C_work_TYPE62;
END_VAR
```

<RAS information area (INT array)>

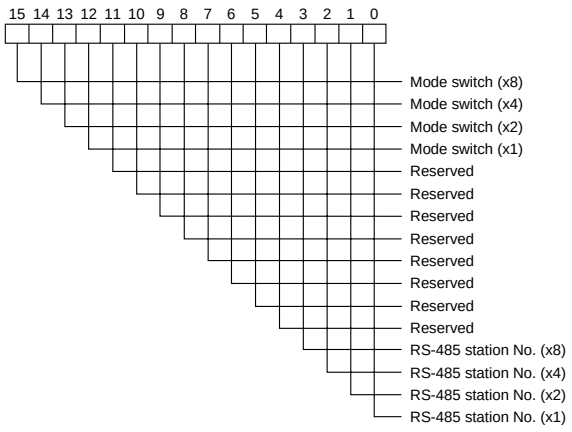
For the variable declaration shown above, the array variable in which RAS information is stored becomes "RAS". Individual element stores the following RAS information.

RAS.RAS[0]	Port status
RAS.RAS[1]	Status of general purpose communication module
RAS.RAS[2]	Send request count
RAS.RAS[3]	Send end count
RAS.RAS[4]	Receive count
RAS.RAS[5]	Frame detection count
RAS.RAS[6]	M_OPEN status
RAS.RAS[7]	M_SEND status
RAS.RAS[8]	M_RECEIVE status
RAS.RAS[9]	M_SEND error count
RAS.RAS[10]	M_RECEIVE error count
RAS.RAS[11]	Reserved
RAS.RAS[12]	Reserved
RAS.RAS[13]	Reserved
RAS.RAS[14]	Reserved
RAS.RAS[15]	Reserved
RAS.RAS[16]	Reserved
RAS.RAS[17]	Reserved
RAS.RAS[18]	Reserved
RAS.RAS[19]	Reserved

• Port status



• Status of general purpose communication module



3-4 Communication Protocol Included General Purpose Inverter Procedure FB "_Cfvrpr"

3-4-1 General

The "_Cfvrpr" FB (function block) communicates data, using "FGI-BUS", between the CPU module and FUJI FVR series general purpose inverter. The "_Cfvrpr" FB performs part of the processing under the communication protocol in the firmware in the general purpose communication module in order to reduce the number of program steps (see note).
For more information of the dedicated transmission parameters that are necessary to control FUJI FVR series general purpose inverter, refer to the manual supplied with the inverter.

Note: This FB can be used with V2536 or later version of general purpose communication module.

This FB performs the following operations:

1) Initialization of communication port
Initializes the RS-485 port (the setting of transmission speed, data bit length, parity bit, etc.)

2) Data sending/receiving function
Sends data from an application program in the MICREX-SX series CPU module to general purpose inverter (FVR) via the general purpose communication module, or outputs the data received via the general purpose communication module to an application program.

3) Transmission condition monitoring function
Monitors data transmission condition and outputs error information if abnormal.

<Connecting method of "_Cfvrpr">
The connecting method between the general purpose communication module and FUJI FVR series general purpose inverter must be "1:N connection" by RS-485 (2-wire system).

<FB operating conditions>

System configuration:
- One FB is necessary for one communication port to be used (RS-485). The control of all the inverters that are connected to one RS-485 system is performed by one FB.
- FUJI general purpose inverter may not be connected to the RS-232C port.
- This FB cannot be used in the mode where RS-232C/RS-485 signal conversion is enabled for the general purpose communication module.

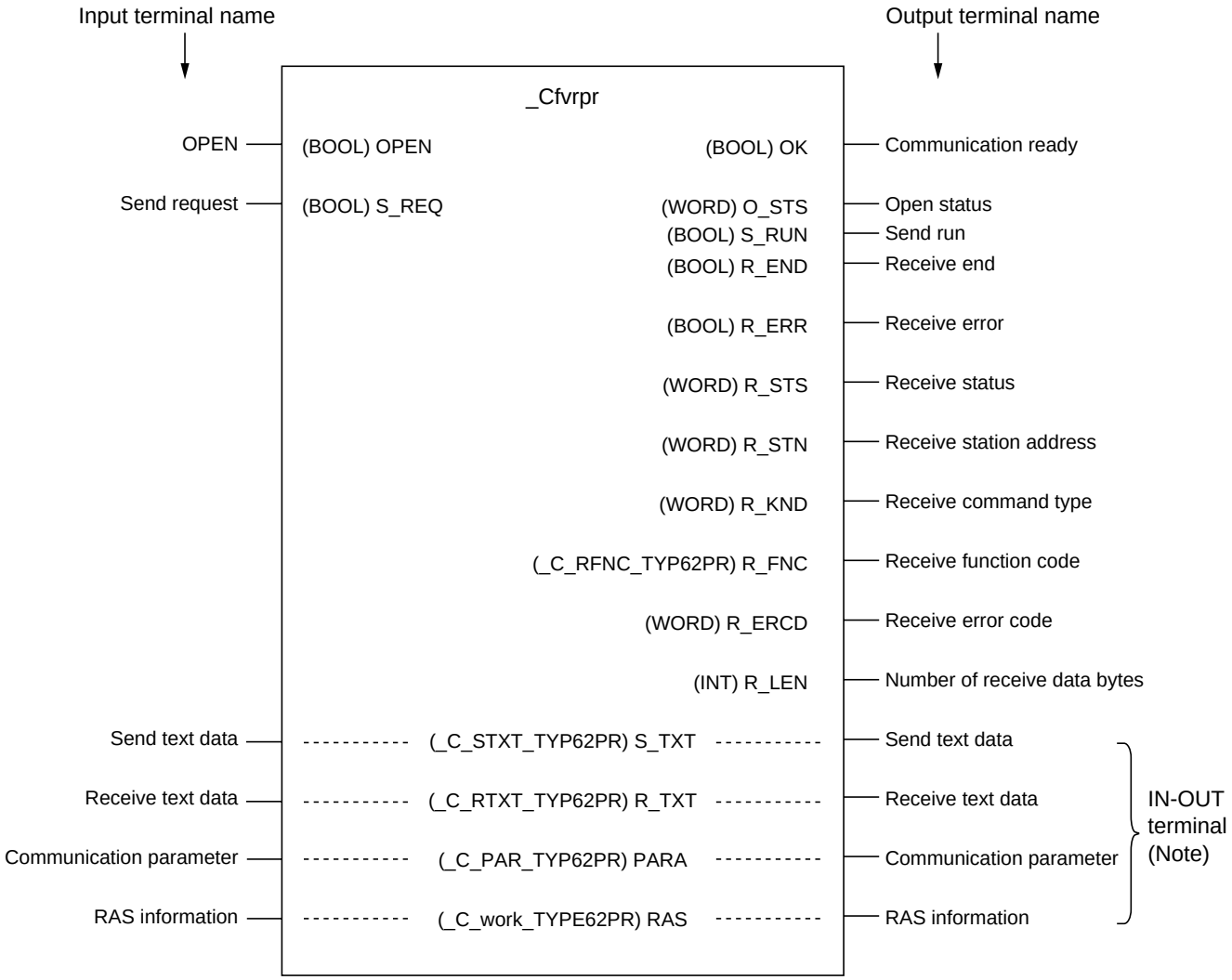
Memory capacity:
Program area = 1186 steps
Data memory capacity = Instance memory for user FBs (116 words)
+ Standard memory (826 words)
+ Instance memory for system FBs (74 words)

Notes:
1) Above figures for memory capacity includes the area for the main body of non-procedural FB and those for sub-FBs that are called from the non-procedural FB.
2) Above figures for standard memory and retain memory includes the memory capacity necessary for sending/receiving data.

3-4 Communication Protocol Included General Purpose Inverter Procedure FB "_Cfvrpr"

3-4-2 Specifications for _Cfvrpr

(1) FB format



* ( ) indicates data type.

Note: In general, FBs are used with only the terminals connected that are necessary for communication. Be sure to connect variables to the IN-OUT terminals.

(2) Explanation of each FB terminal

Terminal name	Variable name	Data type	I/O	Description
Open	OPEN	BOOL	IN	ON: Sends "communication parameters" to the general purpose communication module to initialize communication. When communication becomes possible, "communication ready" turns ON. OFF: Performs the processing for ending communication. "Communication ready" turns OFF.
Send request	S_REQ	BOOL	IN	Starts to send data. When sending is completed, this needs to be turned OFF by the application program.
Send text data	S_TXT	_C_STXT _TYP62PR	IN_OUT	Stores the send text data.
Communication parameter	PARA	_C_PAR _TYP62PR	IN_OUT	Stores the initialization parameters.
Communication ready	OK	BOOL	OUT	Turns ON when initialization has ended normally to indicate that the system is ready for sending/receiving data.
Open status	O_STS	WORD	OUT	2-digit hexadecimal code to indicate the result of initialization
Send run	S_RUN	BOOL	OUT	Turns ON while data is sent to all stations.
Receive end	R_END	BOOL	OUT	Turns ON when receive ends.
Receive error	R_ERR	BOOL	OUT	Turns ON if a receive error has occurred. R_ERR does not turn ON while NAK is received from a remote station because NAK signal is received normally. R_ERR turns ON only when data is not normally received due to no response or BCC error, for example.
Recieve status	R_STS	WORD	OUT	Code to indicate the result of receiving. In case of no response (or BCC error), sending station code is output to the receiving station address so that it can be known which station didn't respond.
Receive station address	R_STN	WORD	OUT	Stores the receiving station address that indicates from which inverter the data was sent.
Recieve command type	R_KND	WORD	OUT	Stores command type characters of received frame.
Receive function code	R_FNC	_C_RFNC _TYP62PR	IN_OUT	Stores function type character and function number of received frame.
Receive error code	R_ERC-D	WORD	OUT	Stores error code sent from inverter if NAK is received from inverter.
Number of receive data bytes	R_LEN	INT	OUT	Stores the number of bytes of received data.
Receive text data	R_TXT	_C_RTXT _TYP62PR	IN_OUT	Received text data is stored when the data of the station to which the inverter responded is received by the general purpose communication module. When multiple inverters are connected, the application program judges, based on receive station address "R_STN", from which station the data was sent.
RAS information	RAS	_C_work _TYPE62PR	IN_OUT	Stores the operating status of this FB.

3-4 Communication Protocol Included General Purpose Inverter Procedure FB "_Cfvrpr"

<Format of the variables that are connected to the "send text data" terminal (S_TXT)>
The data that are connected to "S_TXT" must be laid out as follows:

<When variable name is "S_TXT">

```
VAR
    S_TXT:   _C_SEND_TYPE62PR;
END_VAR
```

	Upper order	Lower order	
S_TXT[0]	SOH	Total number of data sending stations	Example of standard frame
S_TXT[1]	Station number (the units digit)	Station number (the tens digit)	
S_TXT[2]	Command type character	ENQ	
S_TXT[3]	Function number (the tens digit)	Function type character	
S_TXT[4]	Added special data	Function number (the units digit)	
S_TXT[5]	Data 2	Data 1	
S_TXT[6]	Data 4	Data 3	
S_TXT[7]	SOH	ETX	Example of selecting
S_TXT[8]	Station number (the units digit)	Station number (the tens digit)	
S_TXT[9]	Command type character	ENQ	
S_TXT[10]	Data 2	Data 1	
S_TXT[11]	Data 4	Data 3	
S_TXT[12]	SOH	EXT	Example of polling
S_TXT[13]	Station number (the units digit)	Station number (the tens digit)	
S_TXT[14]	Command type character	ENQ	
S_TXT[15]	00	ETX	
. . .	. . .	. . .	
S_TXT[217]	00	00	

Note: Send data is prepared for the standard frames (14 bytes each) for maximum 31 inverter stations that can be connected plus one byte (= 218 words). The data shall be set such that unused areas become 0 (zero).

* Send data is prepared for all the inverter stations that are connected, but data is sent to or received from one station at a time due to FB internal processing.

<List of receive error codes "R_ERCD">

Error code	Name	Description
4A(h)	Format error	Data exceeded the specified range.
4B(h)	Command error	Unspecified command code was sent.
4E(h)	Function code error	Inexistent function code was requested.
4F(h)	Write inhibit	Write request was issued during operation to a write inhibited function or a function that inhibits writing during operation.

Note: For more information, refer to the inverter specifications.

3-4-3 Initialization

(1) Set values of initialization parameters

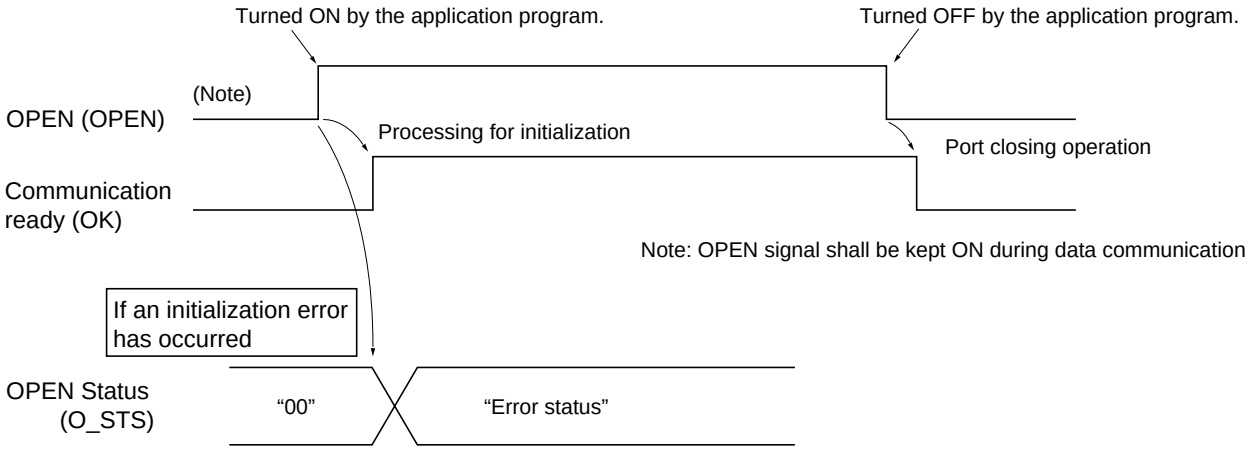
In order to initialize the ports of the general purpose communication module, it is necessary to set proper values for each "communication parameter" item so as to match the communication specifications of the corresponding external device. The table below shows the setting contents of these parameters.

No.	Item	Description
0	General purpose communication module station No.	Sets the station number of the general purpose communication module on the SX bus.
1	Port No.	Designates an interface port on the general purpose communication module. 1: RS-485 port
2	Message port No. 1	Designates a port No. for sending/receiving message to/from a general purpose communication module (1 to 127). Note: Avoide setting the same value as other message send/receive port numbers.
3	Message port No. 2	Designates a port No. for sending/receiving message to/from a general purpose communication module (1 to 127). Note: Avoide setting the same value message port No. 1 or other message send/ receive port numbers.
4	Transmission speed	Designates the transmission speed. 0: 1200 1: 2400 2: 4800 3: 9600 4: 19200 5: 38400 6: 57600 7: 76800 8: 115200 90: 300 91: 600 bps
5	Data bit	Designates the data bit length. When "7" is selected, 7 bits make up one data; when "8", 8 bits make up one data. 0: 7 bits 1: 8 bits
6	Parity bit	This is a bit which is added to data for the purpose of error detection. Designate the proper one to match the setting of the device at the other end. 0: None 1: Odd 2: Even
7	Stop bit	This bit indicates the end of data. Designate the proper one to match the setting of the device at the other end. 0: 1 bit 2: 2 bits
8 ⋮ 15	Reserved	Not used
16	Response monitoring timer	The timer for monitoring during the period from when the CPU module sends a send request to a device until communication ends. This value can be set in 0.01-second steps. Example: For 1 second, set this value to "100". Maximum 327.67 seconds can be set.
17	Retry count	Sets how many times to retry communicating. Setting range: 1 to 32767
18 ⋮ 39	Reserved	Not used

Note: Nos. 0 to 7 are the basic transmission parameters, and their setting cannot be changed during operation. The setting of parameter Nos. 16 and 17 can be changed during operation.

(2) Initialization procedure

When the OPEN terminal of FB is turned ON, communication parameters are transferred to the general purpose communication module, which executes initialization. If initialization is impossible due to communication parameter setting error or hardware error, FB outputs the corresponding error code to O_STS (OPEN status).

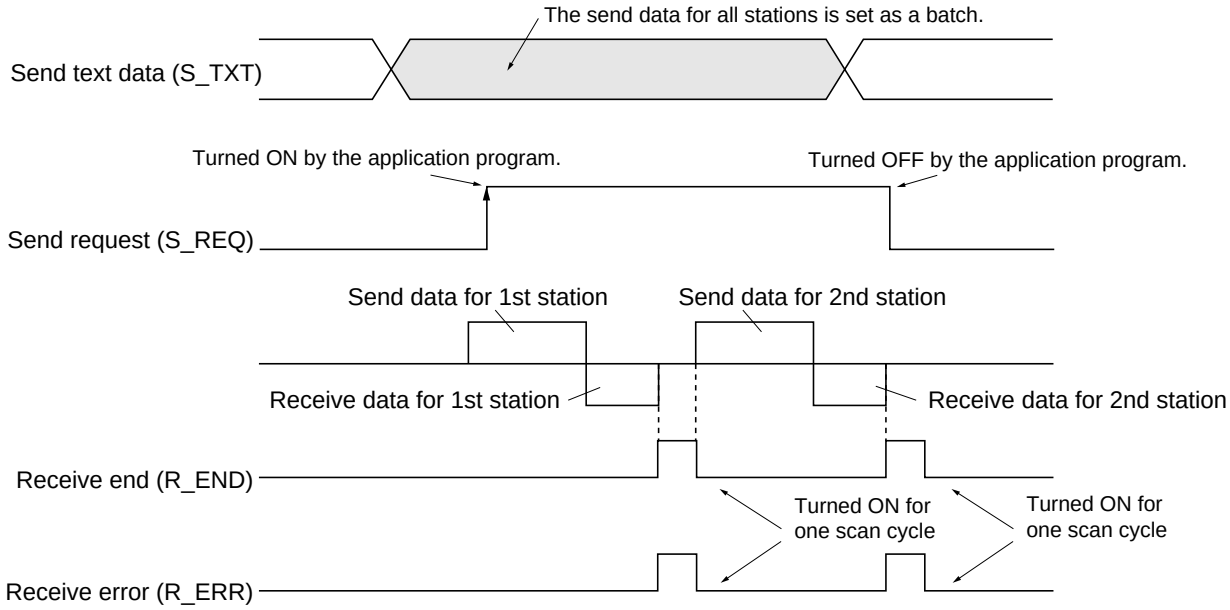


(3) OPEN status list

No.	Result of initialization	Remarks
'00'	Ended normally	
'01'	Transmission speed error	General purpose communication module detected an error.
'02'	Data bit length error	General purpose communication module detected an error.
'03'	Parity bit error	General purpose communication module detected an error.
'04'	Stop bit error	General purpose communication module detected an error.
'3F'	Parameter changeover error	Parameter modification after OPEN detected
'40'	Response monitoring timer value setting error	When negative
'41'	Retry count setting error	When negative
'80'	General purpose communication module station No. setting error	
'81'	Port No. setting error	
'82'	Message port No. setting error	
'93'	Open error	Initialization failed due to SX bus error
'94'	Open error	Initialization failed due to abnormality on general purpose communication module
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because selfdiagnosis is being exected.	General purpose communication module detected an error.

3-4-4 Data sending/receiving

(1) Data sending/receiving procedure



The send text data for all the stations is set as a batch, and send request is turned ON by the application program. The FB sends the data to the 1st station and sets receive status (R_STS), receive station address (R_STN), receive command type (R_KND) and receive function code (R_FNC) when response data is received from the inverter on the 1st station. Then receive end flag (R_END) is turned ON for one scan cycle. During the next scan cycle, communication with the 2nd station inverter is started. When response data is received from the 2nd station after the data is sent to the 2nd station, receive status (R_STS), receive station address (R_STN), receive command type (R_KND) and receive function code (R_FNC) are overwritten, and receive end flag (R_END) is turned ON again for one scan cycle.
Therefore, the setting shall be made such that the processing for receiving is started when receive end flag is turned ON.

(2) Receiving status list

No.	Result of sending/receiving	Remarks
'00'	Ended normally	
'01'	Send buffer has overflowed	
'02'	Receive buffer has overflowed	
'40'	Data send time-out	
'41'	Protocol response error	If the response of remote station could not normally be received, the address of said remote station is output to receive station address (R_STN).
'42'	BCC error	
'43'	No vaild data	
'44'	Previous send data sending	
'90'	General purpose communication module disconnected.	Not detected
'91'	SX bus send error	
'92'	SX bus receive error	
'A0'	Object port error	General purpose communication module detected an error.
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.
'C0'	Hardware error detected during receiving	General purpose communication module detected an error.
'C1'	Receive buffer has overflowed.	General purpose communication module detected an error.
'C2'	Send buffer has overflowed.	General purpose communication module detected an error.
'C3'	Send data size over	General purpose communication module detected an error.

3-4-5 RAS information

RAS information area is used as the work memory for FB operations as well as for checking the operating condition of FBs. To use RAS information from the application program, it is necessary to declare variables in the following manner:

<Example of variable declaration>

```
VAR
    RAS:    _C_work_TYPE62PR;
END_VAR
```

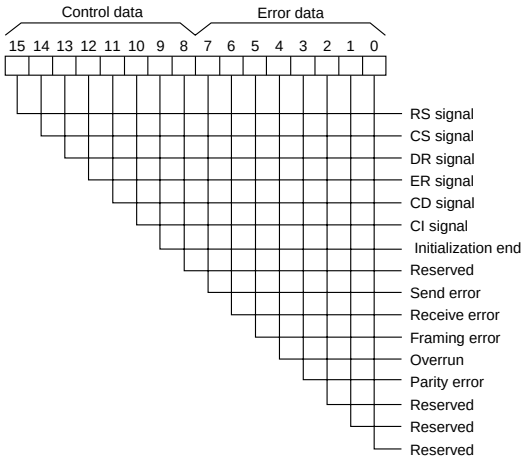
<RAS information area (INT array)>

For the variable declaration shown above, the array variable in which RAS information is stored becomes "RAS". Individual element stores the following RAS information.

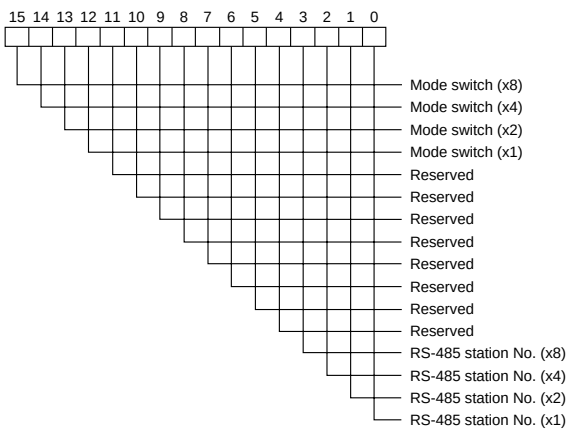
RAS.RAS[0]	Port status
RAS.RAS[1]	Status of general purpose communication module
RAS.RAS[2]	Send request count
RAS.RAS[3]	Send end count (Note)
RAS.RAS[4]	Receive count
RAS.RAS[5]	Frame detection count
RAS.RAS[6]	M_OPEN status
RAS.RAS[7]	M_SEND status
RAS.RAS[8]	M_RECEIVE status
RAS.RAS[9]	M_SEND error count
RAS.RAS[10]	M_RECEIVE error count
RAS.RAS[11]	Reserved
RAS.RAS[12]	Reserved
RAS.RAS[13]	Reserved
RAS.RAS[14]	Reserved
RAS.RAS[15]	Reserved
RAS.RAS[16]	Reserved
RAS.RAS[17]	Reserved
RAS.RAS[18]	Reserved
RAS.RAS[19]	Reserved

Note: Send end count is incremented by one when the sending of data to all stations is completed.

• Port status



• Status of general purpose communication module



3-5 FUJI Temperature Controller Procedure FB "_CfdPYX"

3-5-1 Generall

The "_CfdPYX" FB (function block) communicates data between the CPU module and FUJI PYX or PYH series temperature controller using the start-stop synchronization type transmission protocol. For more information of the dedicated transmission parameters that are necessary to control the PYX or PYH series temperature controller, refer to the manual for the corresponding temperature controller.

1) Initialization of communication port
Initializes the RS-485 port (the setting of transmission speed, data bit length, parity bit, etc.)

2) Data sending/receiving function
Sends data from an application program in the MICREX-SX series CPU module to FUJI PYX or PYH series temperature controller via the general purpose communication module, or outputs the data received via the general purpose communication module to an application program.

3) Transmission condition monitoring function
Monitors data transmission condition and outputs error information if abnormal.

<Connecting method of "_CfdPYX">
The connecting method between the general purpose communication module and FUJI PYX or PYH series temperature controller must be "1:N connection" by RS-485 (2-wire system).

<FB operating conditions>

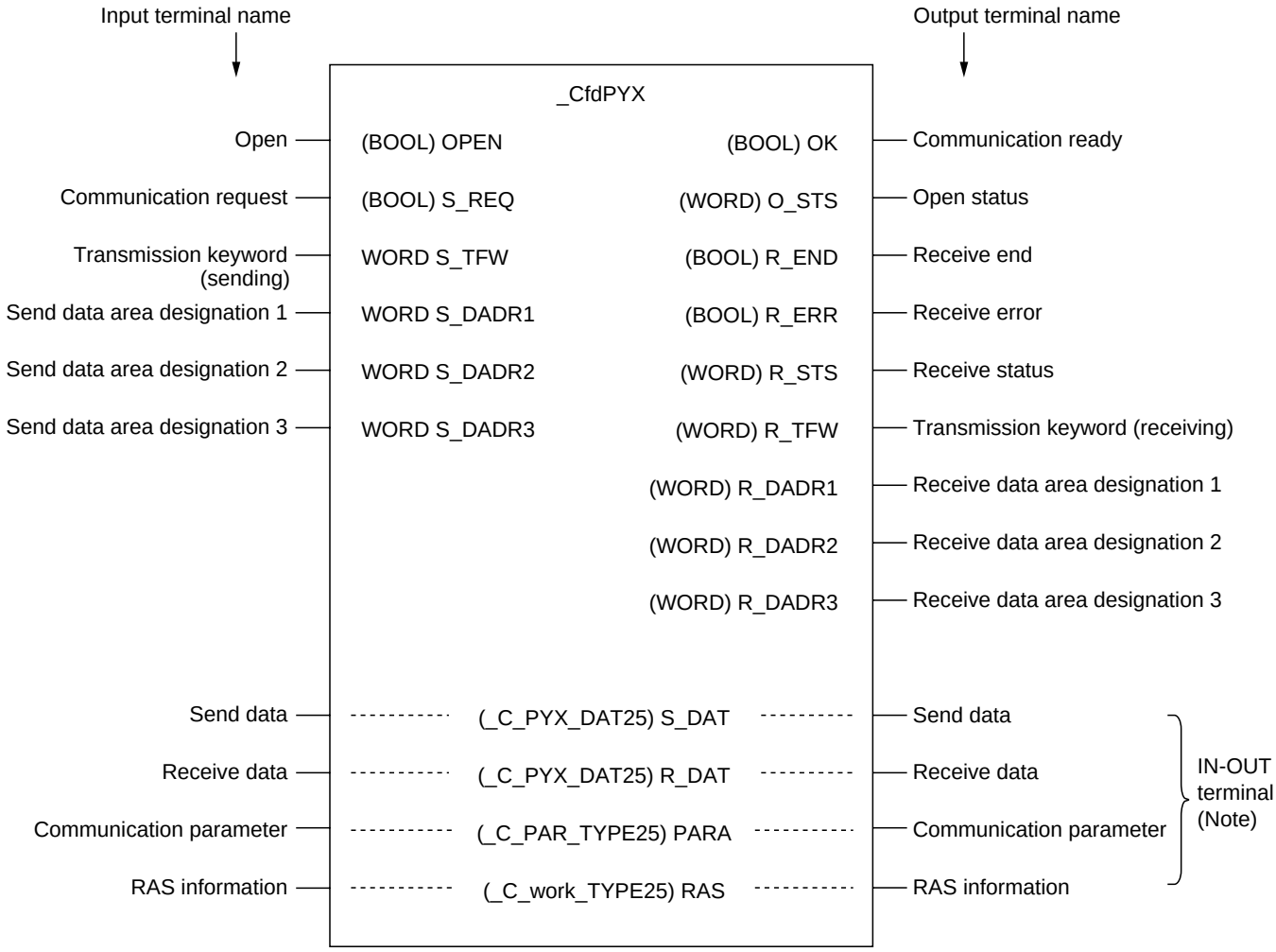
System configuration:
- One FB is necessary for each communication port (RS-485) to be used. Communication with all the temperature controllers that are connected to one RS-485 is performed by one FB.
- This FB cannot be used in the mode where RS-232C/RS-485 signal conversion is enabled for the general purpose communication module.

Memory capacity:
Program area = 1718 steps
Data memory capacity = Instance memory for user FBs (304 words)
+ Standard memory (818 words)
+ Instance memory for system FBs (98 words)

Notes:
1) Above figures for memory capacity includes the area for the main body of the temperature controller procedure FB and those for sub-FBs that are called from the temperature controller procedure FB.
2) Above figures for standard memory and retain memory includes the memory capacity necessary for sending/receiving data.

3-5-2 Specification for _CfdPYX

(1) FB format



* ( ) indicates data type.

Note: In general, FBs are used with only the terminals connected that are necessary for communication. Be sure to connect variables to the IN-OUT terminals.

(2) Explanation for FB terminals

Terminal name	Variable name	Data type	I/O	Description
Open	OPEN	BOOL	IN	ON: Sends the "communication parameters" to the general purpose communication module to initialize communication. When communication becomes possible, "communication ready" turns ON. OFF: Performs the processing for ending communication. "Communication ready" turns OFF.
Send request	S_REQ	BOOL	IN	Starts to send data. When sending has ended, this needs to be turned OFF by the application program.
Transmission keyword (sending)	S_TWF	WORD	IN	Designates transmission keyword. There are 6 types of transmission keyword: polling, selecting, control, positive acknowledge 1, positive acknowledge 2 and negative acknowledge.
Send data area designation 1	S_DADR1	WORD	IN	Designates the data area for station No., file No. and extended station No.
Send data area designation 2	S_DADR2	WORD	IN	Designates the data area for extended file No. and data length. However, when transmission keyword is "control", this becomes the command code area.
Send data area designation 3	S_DADR3	WORD	IN	Designates the data area for the start address of file, command code and negative acknowledge cause data.
Send data	S_DAT	_C_PYX_DAT25	IN_OUT	Stores send data.
Communication parameter	PARA	_C_PAR_TYP25	IN_OUT	Stores the initialization parameters.
Communication ready	OK	BOOL	OUT	Turns ON when initialization has ended normally to indicate that the system is ready for sending/receiving data.
Open status	O_STS	WORD	OUT	2-digit hexadecimal code to indicate the result of initialization.
Receive end	R_END	BOOL	OUT	Turns ON when receiving has completed.
Receive error	R_ERR	BOOL	OUT	Turns ON if a receive error has occurred.
Receive status	R_STS	WORD	OUT	Code to indicate the result of receiving
Transmission keyword (receiving)	R_TFW	WORD	OUT	Stores transmission keyword for receiving.
Receive data area designation 1	R_DADR1	WORD	OUT	Designates receive data area.
Receive data area designation 2	R_DADR2	WORD	OUT	Designates receive data area.
Receive data area designation 3	R_DADR3	WORD	OUT	Designates receive data area.
Receive data	R_DAT	_C_PYX_DAT25	IN_OUT	Stores received data.
RAS information	RAS	_C_work_TYPE25	IN_OUT	Stores the operating status of this FB.

3-5-3 Initialization

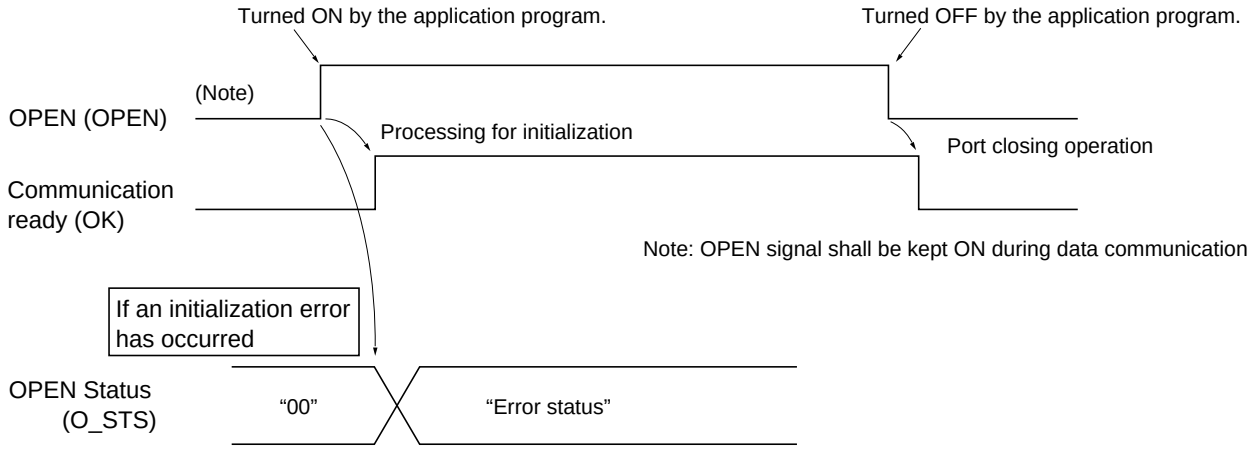
(1) Set values of initialization parameters

In order to initialize communication ports which are to be used, it is necessary to set proper values for each "communication parameter" item so as to match the communication specifications of the FUJI temperature controler. The table below shows the setting contents of these parameters.

No.	Item	Description
0	General purpose communication module station No.	Sets the station number of the general purpose communication module on the SX bus.
1	Port No.	Designates an interface port on the general purpose communication module. 0: RS-232C port 1: RS-485 port
2	Message port No.	Designates a port No. for sending/receiving message to/from a general purpose communication module.
3 ⋮ 15	Reserved	Not used. Set 0 (zero) for the data.
16	Response monitoring timer	The timer for monitoring during the period from when the CPU module sends a send request to a device until communication ends. This value can be set in 0.01-second steps. Example: For 1 second, set this value to "100".
17	Retry count	In case of communication error, this designates how many times to retry communication.
18 ⋮ 39	Reserved	Not used. Set 0 (zero) for the data.

(2) Initialization procedure

When the OPEN terminal of FB is turned ON, communication parameters are transferred to the general purpose communication module, which executes initialization. If initialization is impossible due to communication parameter setting error or hardware error, FB outputs the corresponding error code to O_STS (OPEN status).

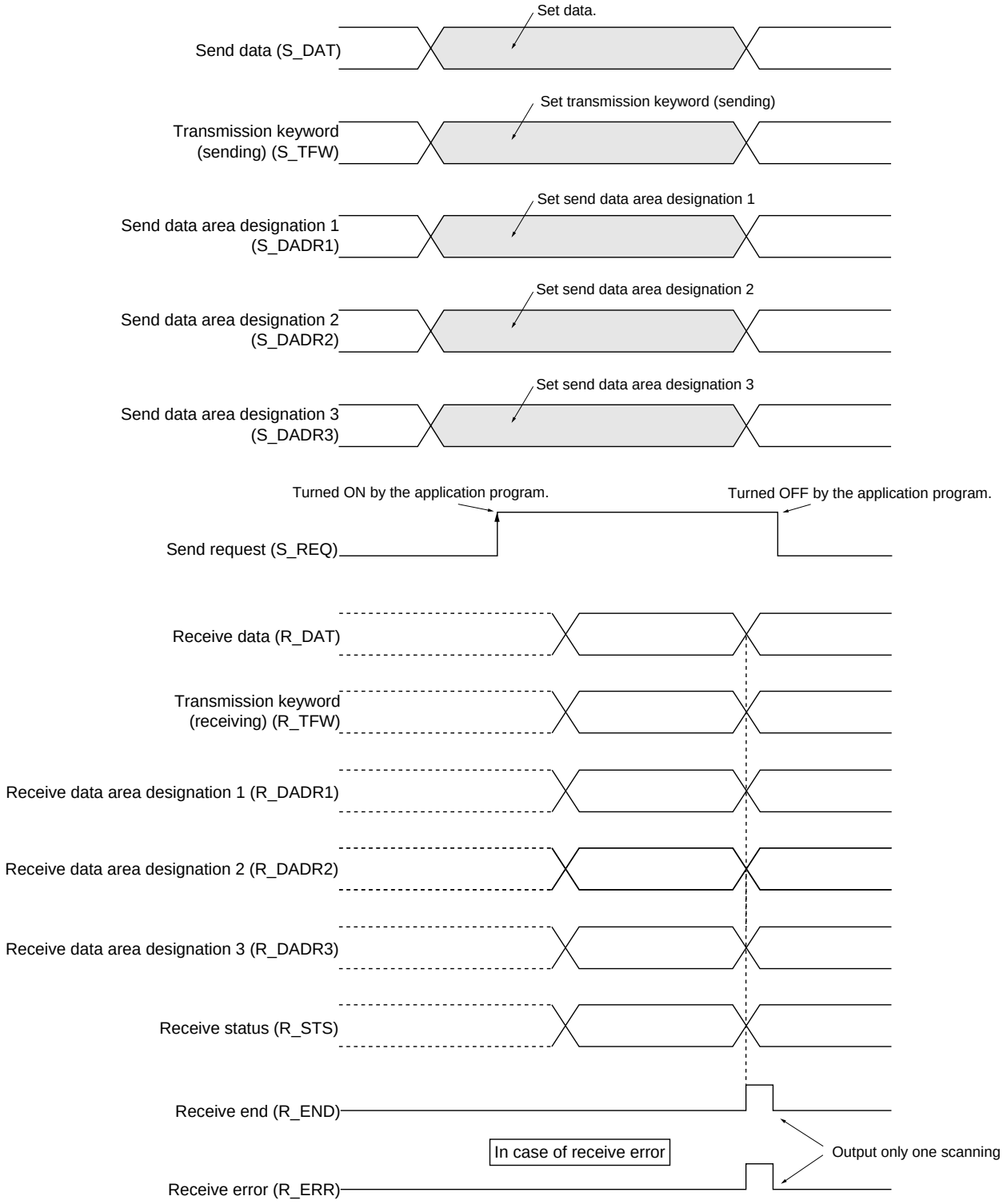


(3) Open status

No.	Result of initialization	Remarks
'00'	Normally ended	
'08'	RS-485 mode setting error	Error detected on general purpose communication module
'3F'	Parameter changeover error	Parameter modification after OPEN detected
'40'	Response monitoring timer value setting error	
'41'	Retry count setting error	
'80'	General purpose communication module station No. setting error	
'81'	Port No. setting error	
'82'	Message port No. setting error	
'93'	Open error	Initialization failed due to SX bus error
'94'	Open error	Initialization failed due to abnormality on general purpose communication module
'A3'	Processing impossible because in code conversion mode.	Error detection on general purpose communication module
'A5'	Processing impossible because in loader mode.	Error detection on general purpose communication module
'A6'	Processing impossible because self-diagnosis is being executed.	Error detection on general purpose communication module

3-5-4 Data sending/receiving

(1) Data sending/receiving procedure



Send data, transmission keyword for sending, send data area designation 1, send data area designation 2 and send data area designation 3 are set and then send request is turned ON by the application program. The FB detects the rising edge to execute the processing for sending. When the processing for sending is completed, the FB automatically comes in receiving waiting mode, executing the processing for receiving.
When the processing for sending and receiving has completed, received data, transmission keyword for receiving, receive data area designation 1, receive data area designation 2 and receive data area designation 3 are stored, and then the receive end flag is set ON (for only one pulse). In case of transmission error, both end flag and receive error flag are set ON (for only one pulse).

(2) Sending/receiving status list

No.	Result of sending/receiving	Remarks
'00'	Ended normally	
'01'	Reserved	
'02'	Send buffer has overflowed	
'40'	Data send time-out	
'41'	Protocol response error	
'42'	BCC error	
'90'	General purpose communication module disconnected.	Not detected
'91'	SX bus send error	
'92'	SX bus receive error	
'A0'	Object port error	General purpose communication module detected an error.
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.
'C0'	Hardware error detected during receiving	General purpose communication module detected an error.
'C1'	Receive buffer has overflowed.	General purpose communication module detected an error.
'C2'	Send buffer has overflowed.	General purpose communication module detected an error.
'C3'	Send data size over	General purpose communication module detected an error.

3-5-5 RAS information

RAS information area is used as the work memory for FB operations as well as for checking the operating condition of FBs. To use RAS information from the application program, it is necessary to declare variables in the following manner:

<Example of variable declaration>

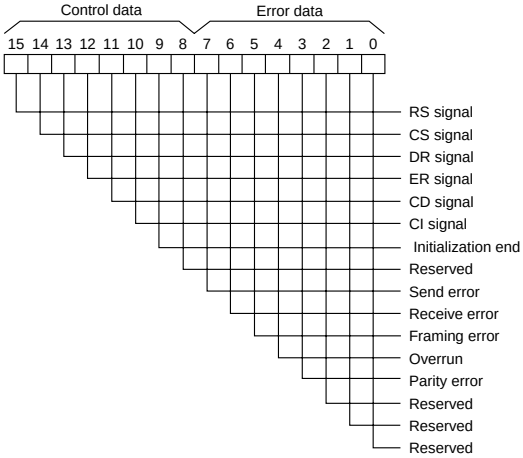
```
VAR
    RAS:    _C_work_TYPE25;
END_VAR
```

<RAS information area (INT array)>

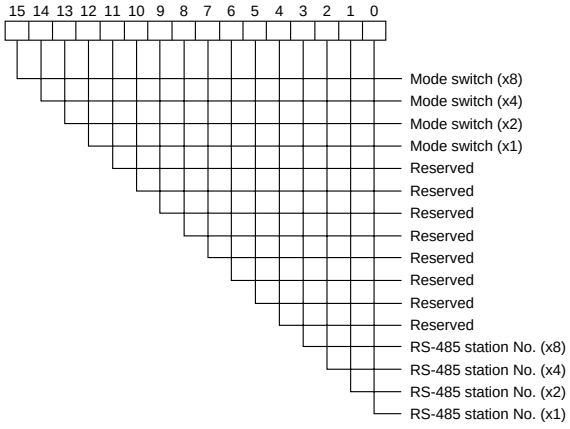
For the variable declaration shown above, the array variable in which RAS information is stored becomes “RAS”. Individual element stores the following RAS information.

RAS.RAS[0]	Port status
RAS.RAS[1]	Status of general purpose communication module
RAS.RAS[2]	Send request count
RAS.RAS[3]	Send end count
RAS.RAS[4]	Receive count
RAS.RAS[5]	Frame detection count
RAS.RAS[6]	M_OPEN status
RAS.RAS[7]	M_SEND status
RAS.RAS[8]	M_RECEIVE status
RAS.RAS[9]	M_SEND error count
RAS.RAS[10]	M_RECEIVE error count
RAS.RAS[11]	Reserved
RAS.RAS[12]	Reserved
RAS.RAS[13]	Reserved
RAS.RAS[14]	Reserved
RAS.RAS[15]	Reserved
RAS.RAS[16]	Reserved
RAS.RAS[17]	Reserved
RAS.RAS[18]	Reserved
RAS.RAS[19]	Reserved

• Port status



• Status of general purpose communication module



3-6 FUJI Bar Code Reader Procedure FB "_CfdPK"

3-6-1 General

The "_CfdPK" FB communicates data between the CPU module and FUJI PK2 series bar code reader using the "start-stop synchronization" type transmission protocol. For more information of the dedicated transmission parameters that are necessary to control FUJI bar code reader, refer to the manual supplied with the PK2 series bar code reader.

1) Initialization of communication ports

Initializes RS-232C ports (the setting of transmission speed, data bit length, parity bit, etc.)

2) Data sending/receiving function

Sends data from an application program in the MICREX-SX series CPU module to the FUJI bar code reader via the general purpose communication module, or outputs the data received via the general purpose communication module to an application program.

3) Transmission condition monitoring function

Monitors data transmission condition and outputs error information if abnormal.

<Connecting method of "_CfdPK">

The connecting method between the general purpose communication module and FUJI PK2 series bar code reader must be "1:1 connection" by RS-232C.

<FB operating conditions>

System configuration:

- One FB is necessary for each communication port (RS-232C) to be used.
- FUJI PK2 series bar code reader may not be connected to the RS-485 port.

Memory capacity:

Program area = 2907 steps

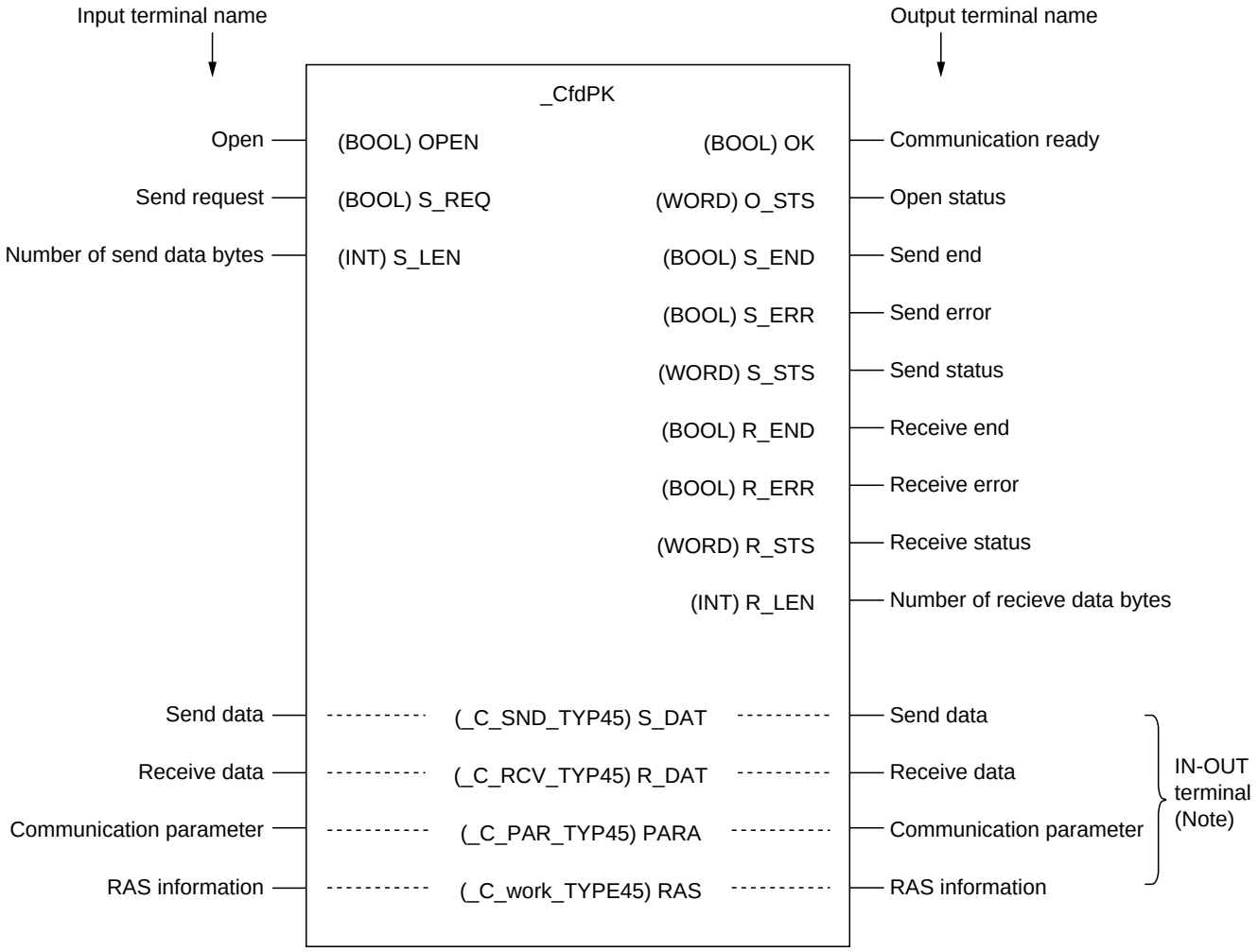
Data memory capacity = Instance memory for user FBs (352 words)
+ Standard memory (846 words)
+ Instance memory for system FBs (90 words)

Notes:

1) Above figures for memory capacity includes the area for the main body of FUJI bar code reader procedure FB and those for sub-FBs that are called from the bar code reader procedure FB.
2) Above figures for standard memory and retain memory includes the memory capacity necessary for sending/receiving data.

3-6-2 Specifiction for _CfdPK

(1) FB format



* ( ) indicates data type.

Note: In general, FBs are used with only the terminals connected that are necessary for communication. Be sure to connect variables to the IN-OUT terminals.

(2) FB terminals

Terminal name	Variable name	Data type	I/O	Description
Open	OPEN	BOOL	IN	ON: Sends the "communication parameters" to the general purpose communication module to initialize communication. When communication becomes possible, "communication ready" turns ON. OFF: Performs the processing for ending communication. "Communication ready" turns OFF.
Send request	S_REQ	BOOL	IN	Starts to send data. When sending has ended, this needs to be turned OFF by the application program.
Number of send data bytes	S_LEN	INT	IN	Stores the number of bytes of send data.
Send data	S_DAT	_C_SDAT_TYP 45	IN_OUT	Stores send data.
Communication parameter	PARA	_C_PAR_TYP45	IN_OUT	Stores the initialization parameters.
Communication ready	OK	BOOL	OUT	Turns ON when initialization has ended normally to indicate that the system is ready for sending/receiving data.
Open status	O_STS	WORD	OUT	2-digit hexadecimal code to indicate the result of initialization.
Send end	S_END	BOOL	OUT	Turns ON when sending has completed.
Send error	S_ERR	BOOL	OUT	Turn ON if a send error has occurred.
Send status	S_STS	WORD	OUT	Code to indicate the result of sending
Receive end	R_END	BOOL	OUT	Turns ON when receiving has completed.
Receive error	R_ERR	BOOL	OUT	Turns ON if a receive error has occurred.
Receive status	R_STS	WORD	OUT	Code to indicate the result of receiving
Number of recieve data bytes	R_LEN	INT	OUT	Stores the number of bytes of receive data.
Receive data	R_DAT	_C_RCV_TYP 45	IN_OUT	Stores received data.
RAS information	RAS	_C_work_TYPE 45	IN_OUT	Stores the operating status of this FB.

3-6-3 Initialization

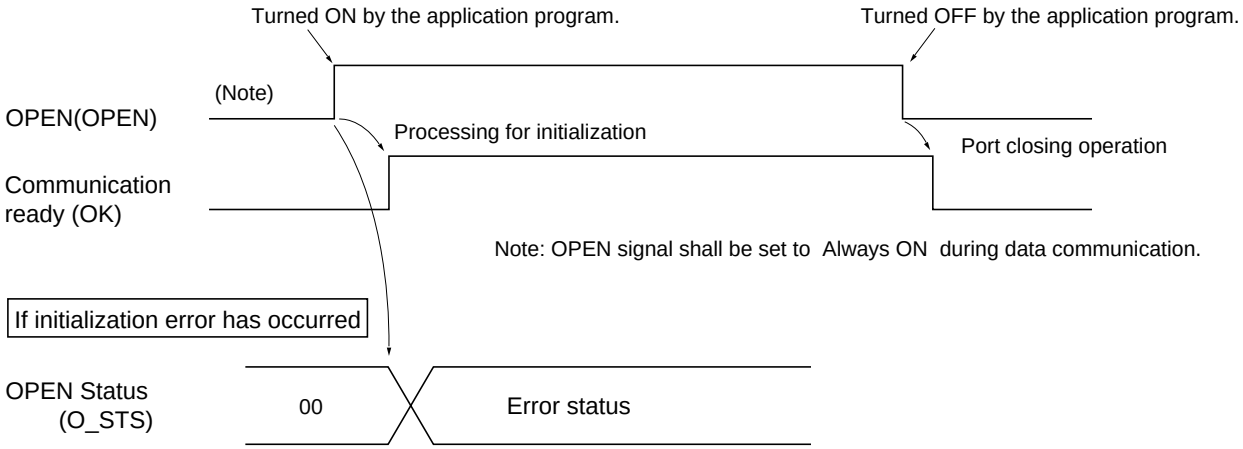
(1) Set values of initialization parameters

In order to initialize (RS-232C) communication ports which are to be used, it is necessary to set proper values for each "communication parameter" item so as to match the communication specifications of the PK2 series bar code reader. The table below shows the setting contents of these parameters.

No.	Item	Description
0	General purpose communication module station No.	Sets the station number of the general purpose communication module on the SX bus.
1	Port No.	Designates an interface port on the general purpose communication module. 0: RS-232C port
2	Message port No.	Designates a port No. for sending/receiving message to/from a general purpose communication module.
3	Reserved	Not used
4	Transmission speed	Designates the transmission speed. 0: 1200 1: 2400 2: 4800 3: 9600 4: 19200 bps
5	Data bit	Designates the data bit length. When "7" is selected, 7 bits make up one data; when "8", 8 bits make up one data. 0: 7 bits 1: 8 bits
6	Parity bit	This is a bit which is added to data for the purpose of error detection. Designate the proper one to match the setting of the device at the other end. 0: None 1: Odd 2: Even
7	Stop bit	This bit indicates the end of data. Designate the proper one to match the setting of the device at the other end. 0: 1 bit 2: 2 bits
8 ⋮ 15	Reserved	Not used
16	Response monitoring timer	The timer for monitoring during the period from when the CPU module sends a send request to a device until communication ends. This value can be set in 0.01-second steps. Example: For 1 second, set this value to "100".
17	Start code	Designates the start code of transmission command. 0: None, 1: STX
18	End code	Designates the end code of transmission command. 0: CR, 1: LF, 2: CR/LF, 3: ETX, 4: ETXCRF, 5: ETX/LF, 6: ETX/CR/LF
19	Transmission procedure	Designates transmission procedure. 0: Non-procedural Receives the end data simply as read by the bar code reader. 1: ACK/NAK procedure Sends ACK when the data read by the bar code reader is received normally; NAK if a receiving error occurred.
20	Retry count	In case of communication error, this designates how many times to retry communication.
21	BCC designation	Sets whether or not to add horizontal parity for checking text data transmission error. 0: No, 1: Yes
22 ⋮ 39	Reserved	Not used

(2) Initialization procedure

When the OPEN terminal of FB is turned ON, communication parameters are transferred to the general purpose communication module, which executes processing for initialization. If initialization is impossible due to a communication parameter setting error or hardware error, FB outputs the corresponding error code to the OPEN Status (O_STS).

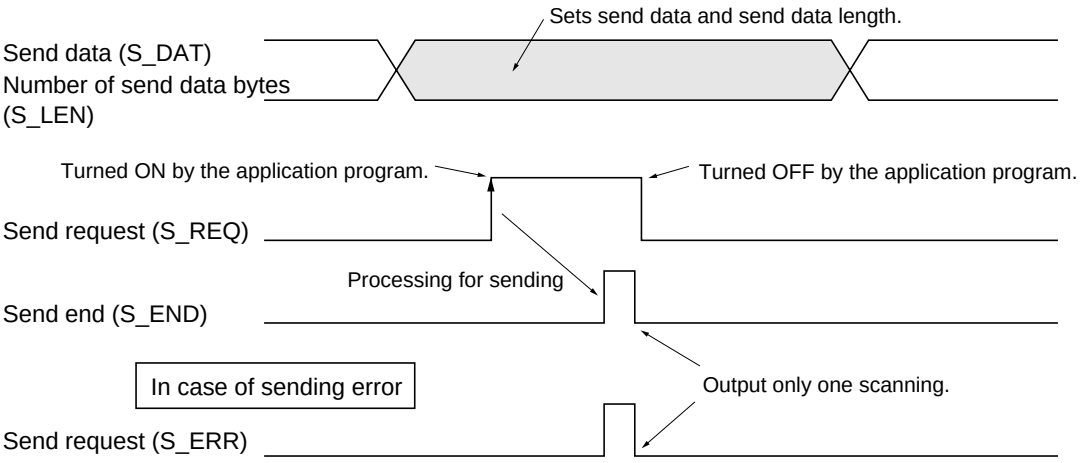


(3) OPEN status list

No.	Result of initialization	Remarks
'00'	Ended normally	
'01'	Transmission speed error	General purpose communication module detected an error.
'02'	Data bit length error	General purpose communication module detected an error.
'03'	Parity bit error	General purpose communication module detected an error.
'04'	Stop bit error	General purpose communication module detected an error.
'3F'	Parameter modification error	Parameter modification error after OPEN
'40'	Response monitoring timer value setting error	
'41'	Retry count setting error	
'42'	Start code setting error	
'43'	End code setting error	
'44'	BCC code setting error	
'45'	Retry count setting error	
'80'	General purpose communication module station No. setting error	
'81'	Port No. setting error	
'82'	Message port No. setting error	
'93'	Open error	Initialization failed due to bus error
'94'	Open error	Initialization failed due to abnormality on general purpose communication module
'A3'	Processing impossible because in code conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.

3-6-4 Data sending procedure

(1) Data sending produre



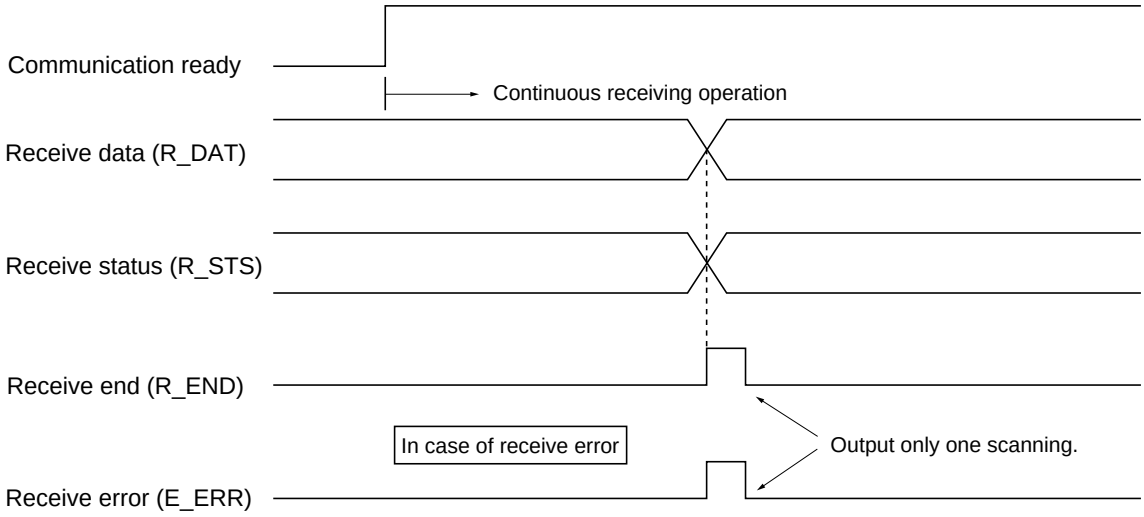
After send data is set, send request is turned ON by the application program. FB detects the rising edge to execute the processing for sending.
When sending has ended, S_END (send end) flag is turned ON (for only one pulse). If a send error has occurred, S_END (send end) and S_ERR (send error) are turned ON (for only one pulse), and the corresponding error code is output to S_STS (send status) ("00" when ended normally).

(2) Send status list

No.	Result of sending	Remarks
'00'	Normally ended	
'01'	Send buffer has overflowed.	
'42'	BCC error	
'90'	General purpose communication module disconnected.	Not detected
'91'	SX bus send error	
'92'	SX bus receive error	Not detected
'A0'	Object port error	Specified port No. is neither 0 nor 1.
'A3'	Processing impossible because in code conversion mode.	
'A5'	Processing impossible because in loader mode.	
'A6'	Processing impossible because self-diagnosis is being executed.	Error detection on general purpose communication module
'C0'	Hardware error detected during receiving	Error detection on general purpose communication module
'C1'	Receive buffer has overflowed	Error detection on general purpose communication module
'C2'	Send buffer has overflowed.	Error detection on general purpose communication module
'C3'	Send data size over	Error detection on general purpose communication module

3-6-5 Data receiving procedure

(1) Data recieving procedure



While the communication ready flag is turned ON, whether the data to be received exists or not is checked continuously and, when it exists, the operation for receiving is executed. When a delimiter between data frames is detected, the received data is stored in R_DAT (receive data), respectively, and the R_END (receive end) flag is turned ON (for only one pulse). The result of receiving is stored in R_STS (receive status). If a receive error has occurred, R_END (receive end) and E_ERR (receive error) are turned ON (for only one pulse).

(2) Receiving status list

No.	Result of receiving	Remarks
'00'	Ended normally	
'02'	Receive buffer has overflowed.	
'41'	Protocal response error	
'42'	BCC error	
'90'	General purpose communication module disconnected.	
'91'	SX bus send error	
'92'	SX bus receive error	
'A0'	Object port error	General purpose communication module detected an error.
'A3'	Processing impossible because in code conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.
'C0'	Hardware error detected during receiving (Parity error, Franming error, etc.)	General purpose communication module detected an error.
'C1'	Receive buffer has overflowed.	General purpose communication module detected an error.
'C2'	Send buffer has overflowed.	General purpose communication module detected an error.
'C3'	Number of send data over	General purpose communication module detected an error.

3-6-6 RAS information

RAS information area is used as the work memory for FB operations as well as for checking the operating condition of FBs. To use RAS information from the application program, it is necessary to declare variables in the following manner:

<Example of variable declaration>

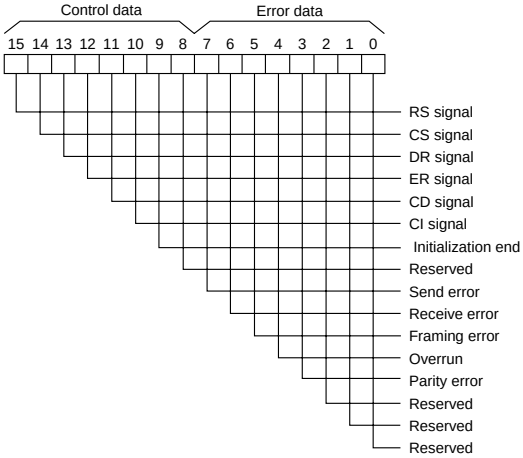
```
VAR
    RAS:    _C_work_TYPE45;
END_VAR
```

<RAS information area (INT array)>

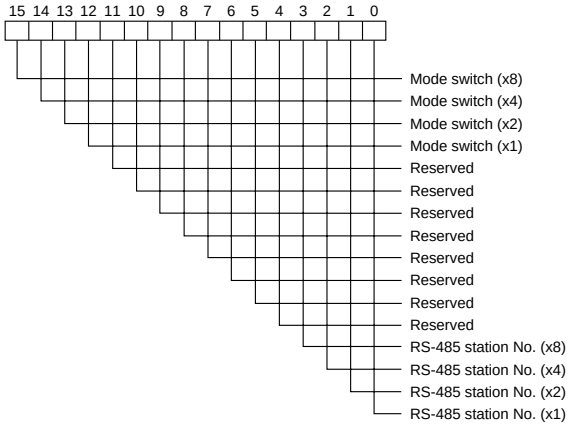
For the variable declaration shown above, the array variable in which RAS information is stored becomes “RAS”. Individual element stores the following RAS information.

RAS.RAS[0]	Port status
RAS.RAS[1]	Status of general purpose communication module
RAS.RAS[2]	Send request count
RAS.RAS[3]	Send end count
RAS.RAS[4]	Receive count
RAS.RAS[5]	Frame detection count
RAS.RAS[6]	M_OPEN status
RAS.RAS[7]	M_SEND status
RAS.RAS[8]	M_RECEIVE status
RAS.RAS[9]	M_SEND error count
RAS.RAS[10]	M_RECEIVE error count
RAS.RAS[11]	Reserved
RAS.RAS[12]	Reserved
RAS.RAS[13]	Reserved
RAS.RAS[14]	Reserved
RAS.RAS[15]	Reserved
RAS.RAS[16]	Reserved
RAS.RAS[17]	Reserved
RAS.RAS[18]	Reserved
RAS.RAS[19]	Reserved

• Port status



• Status of general purpose communication module



3-7-1 General

The "_C_modm" FB communicates data between the CPU module and FA device equipped with MODBUS interface. While there are MODBUS master and slave devices, this FB makes MICREX-SX a master device and communicates data with MODBUS slave devices. MODBUS master device may not be connected to this FB. Determine which functions to use according to the specifications for the MODBUS slave devices to be connected.

1) Initialization of communication ports

Initializes communication ports to be used (the setting of transmission speed, parity bit, etc.)

2) Data sending/receiving function

Sends data from an application program in the MICREX-SX series CPU module to MODBUS devices via the general purpose communication module, or outputs the data received via the general purpose communication module to an application program.

3) Transmission condition monitoring function

Monitors data transmission condition and outputs error information if abnormal.

4) Supported function codes

This FB supports the following functions. For the data format for individual function, refer to "3-7-2 (4) Function codes".

Function code	Function
01	Coil (output) reading
02	Input bit reading
03	Retain register reading
04	Input register reading
05	Forcible ON/OFF (1 bit)

Function code	Function
06	Retain register writing (1 word)
08	Diagnosis (see note)
0F	Forcible ON/OFF (multiple bits)
10	Retain register writing (multiple words)

Note: For more information of diagnosis, refer to the MODBUS specification.

<Connecting method of "_C_modm">

For the connecting method between the general purpose communication module and MODBUS devices, 1:N connection by RS-485 (2-wire system) and 1:1 connection by RS-232C are available.

<FB operating conditions>

System configuration:

- One FB is necessary for each communication port (RS-232C, RS-485) to be used. One FB is used to communicate data with the MODBUS devices that are connected to one port.
- This FB can be used with V2536 or later version of the general purpose communication module.

Memory capacity:

Program area = 3276 steps

Data memory capacity = Instance memory for user FBs (342 words)
+ Standard memory (2386 words)
+ Instance memory for system FBs (55 words)

Notes:

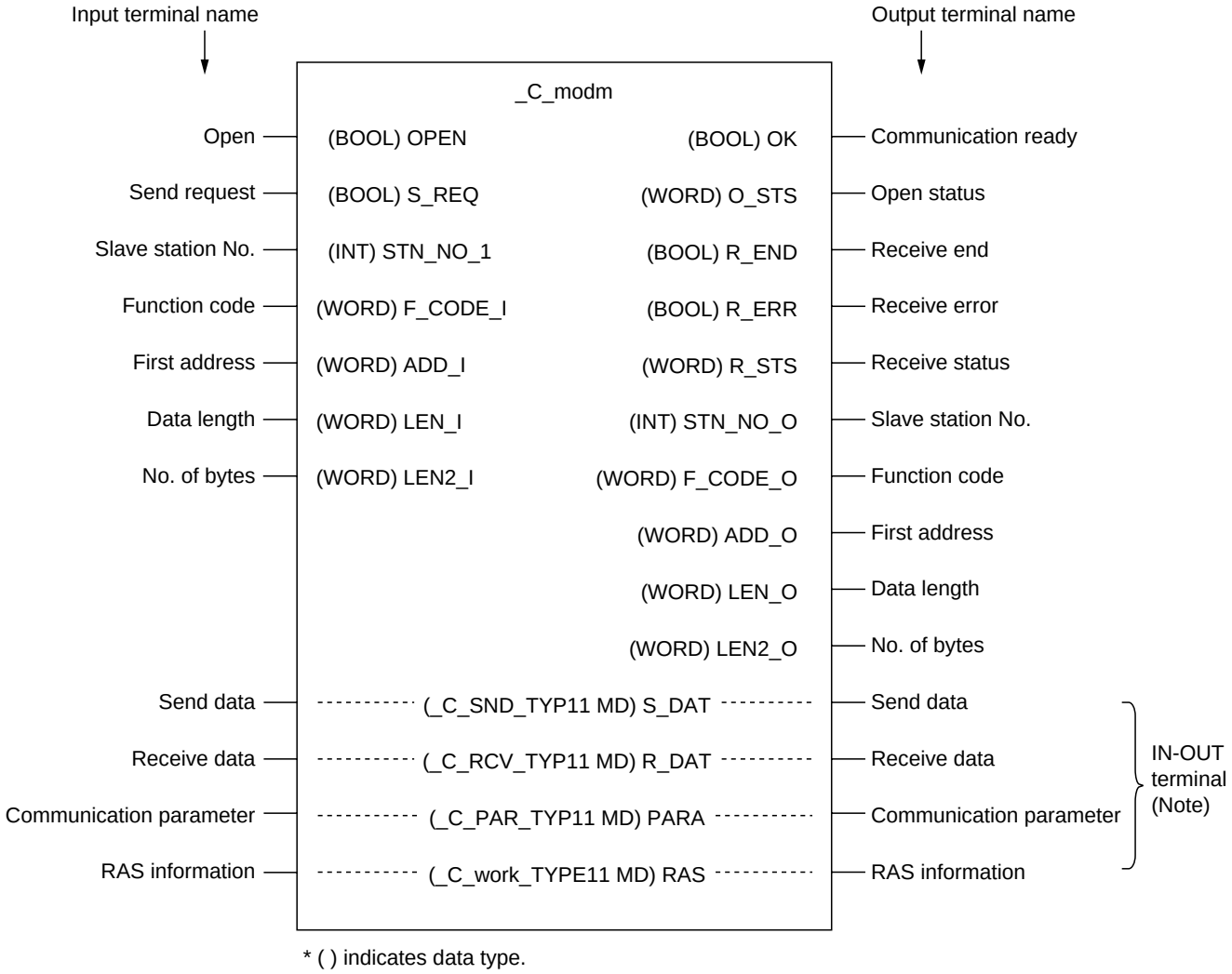
1) Above figures for memory capacity includes the area for the main body of MODBUS procedure FB and those for sub-FBs that are called from the MODBUS procedure FB.
2) Above figures for standard memory and retain memory includes the memory capacity necessary for sending/receiving data.

3-7-2 Specification for _C_modm

(1) Communication specification

Item		Specification
Transmission speed		300/600/1200/2400/4800/9600/19200/38400/57600/76800/115200 bps
Data bits		8bits (Fxation)
Parity bit		None/odd/even
Stop bits		Selected from 1/2 bits
DCE		Selected from DTE/DCE/modem DTE mode Modem DTE mode: Turn CD ON when receiving
Signal control	DTE mode	ER: always ON
	DCE mode	DR: always ON
Signal flow control	DTE mode	Off RS: always ON; Sending: unconditional On RS: ON during sending; Sending: when CS is ON
	DCE mode	Off CS: always ON; Sending: unconditional On CS: ON when RS is ON; Sending: when ER is ON
XON/XOFF control		The prerequisite of XON/XOFF flow control is that XON or XOFF does not appear on the data.
RS-485 mode		Selected from 4-wire/2-wire
Code conversion		None (Fixation)
Error check		CRC16 (Upper order/Lower order) (Fixation)
Transmission timer value		x10ms

(2) FB format



* ( ) indicates data type.

Note: In general, FBs are used with only the terminals connected that are necessary for communication. Be sure to connect variables to the IN-OUT terminals.

(3) FB terminals

Terminal name	Variable name	Data type	I/O	Description
Open	OPEN	BOOL	IN	ON: Sends the "communication parameters" to the general purpose communication module to initialize communication. When communication becomes possible, "communication ready" turns ON. OFF: Performs the processing for ending communication. "Communication ready" turns OFF.
Send request	S_REQ	BOOL	IN	Starts to send data. When sending has ended, this needs to be turned OFF by the application program.
Slave station No. (Query)	SIN_NO _I	INT	IN	Designates destination slave station No.
Function code (Query)	F_CODE _I	WORD	IN	Designates which MODBUS function to use. This code is set in the lower order byte. Example: WORD#16#0010
Start address (Query)	ADD_I	WORD	IN	Designates the start address of the data that becomes the object of reading or writing operation. For diagnosis (08), diagnosis code is input. Set data in the following format: 15 8 7 0 Upper order | Lower order
Data length (Query)	LEN_I	WORD	IN	Designates the size (the number of bits or words) of read data or write data. Some functions do not use this terminal. In such case, set this item to "0000h". Set data in the following format: 15 8 7 0 Upper order | Lower order
Number of bytes (Query)	LEN2_I	WORD	IN	Designates the number of bytes of write data when multiple bits (0F) or multiple words (10) are to be written. For other functions, set this item to "0000h". Set data in the following format: 15 8 7 0 Upper order | Lower order
Send data	S_DAT	_C_SND_TYP 11 MD	IN_OUT	Stores send data. For data reading or diagnosis, the data are all set to 0 (zero).
Communication parameter	PARA	_C_PAR_TY 11 MD	IN_OUT	Stores the initialization parameters. Refer to 3-7-3 for details.
Communication ready	OK	BOOL	OUT	Turns ON when initialization has ended normally to indicate that the system is ready for sending/receiving data.
Open status	O_STS	WORD	OUT	2-digit hexadecimal code to indicate the result of initialization.
Receive end	R_END	BOOL	OUT	Turns ON when receiving has completed.
Receive error	R_ERR	BOOL	OUT	Turns ON if a receive error has occurred.
Receive status	R_STS	WORD	OUT	Code to indicate the result of receiving
Slave station No. (Response)	SIN_NO _O	INT	OUT	Remote slave station number is output.
Function code (Response)	F_CODE _O	WORD	OUT	According to the request from this FB, the values of "function code", "start address", "data length" and "the number of bytes" that are included in the response data sent from slave device are output. They are output in the same format as is set in the input (Query).
First address (Response)	ADD_O	WORD	OUT	
Data length (Response)	LEN_O	WORD	OUT	
No. of bytes (Response)	LEN2_O	WORD	OUT	
Receive data	R_DAT	_C_RCV_TYP 11 MD	IN_OUT	Stores received data.
RAS information	RAS	_C_work_TYPE 11 MD	IN_OUT	Stores the operating status of this FB.

<Format of the variables that are connected to the send data terminal "S_DAT">
The data that is connected to S_DAT shall be laid out as follows:

<When variable name is "S_DAT">

```
VAR
    S_DAT:  _C_SND_TYP11MD;
END_VAR
```

	Upper order	Lower order
S_DAT [0]	Data 2	Data 1
S_DAT [1]	Data 4	Data 3
⋮	⋮	⋮
S_DAT [255]	Data 512	Data 511

<Format of the variables that are connected to the "receive data" terminal "R_DAT">
The data that is connected to R_DAT will be laid out as follows:

<When variable name is "R_DAT">

```
VAR
    R_DAT:  _C_RCV_TYP11MD;
END_VAR
```

	Upper order	Lower order
R_DAT [0]	Data 2	Data 1
R_DAT [1]	Data 4	Data 3
⋮	⋮	⋮
R_DAT [255]	Data 512	Data 511

(4) Function code

1) Coil reading (code: 01)/Input bit reading (code: 02)
Reads the status of the output/input bit of a slave device. The format of request message and response message is shown blow.

FB terminals	Request message	Response message	FB terminals
Slave station No. (SIN_NO_I)→	Slave station No.	Slave station No.	→Slave station No. (SIN_NO_O)
Function code (F_CODE_I)→	Function code	Function code	→Function code (F_CODE_O)
Start address (ADD_I)→	Start address (upper order)	Number of bytes of read data	→Data length (LEN_O)
	Start address (lower order)	Read data (lower order)	
Data length (LEN_I)→	Data length (upper order)	Read data (upper order)	
	Data length (lower order)	Read data (lower order)	→Receive data (R_DAT)
	CRC (lower order)	Read data (upper order)	
	CRC (upper order)	:	
		CRC (lower order)	
		CRC (upper order)	

Notes:
1) In case of abnormal response, 80hex is added to function code (F_CODE_O).
2) Function 01/02 does not use the number of bytes (LEN2_I). Set "WORD#16#0000" for the LEN2_I terminal. Similarly, all the variables that are connected to the "send data" terminal (S_DAT) shall be set to zero.
3) CRC is automatically calculated and added to the data by this FB.
4) The number of bytes of read data is stored in the upper order byte of data length (LEN_O).
5) When read data is received, its upper- and lower-order bytes are reversed as shown below:

Receive data (R_DAT) Offset address with respect to the designated start address.

+7	+6	+5	+4	+3	+2	+1	+0	+15	+14	+13	+12	+11	+10	+9	+8
+23	+22	+21	+20	+19	+18	+17	+16	+31	+30	+29	+28	+27	+26	+25	+24

2) Retain register reading (code: 03)/Input register reading (code: 04)

Reads the status of the input bit of a slave device. The format of request message and response message is shown blow.

FB terminals	Request message	Response message	FB terminals
Slave station No. (SIN_NO_I)→	Slave station No.	Slave station No.	→Slave station No. (SIN_NO_O)
Function code (F_CODE_I)→	Function code	Function code	→Function code (F_CODE_O)
Start address (ADD_I)→	Start address (upper order)	Number of bytes of read data	→Data length (LEN_O)
	Start address (lower order)	Read data (lower order)	
Data length (LEN_I)→	Data length (upper order)	Read data (upper order)	
	Data length (lower order)	Read data (lower order)	→Receive data (R_DAT)
	CRC (lower order)	Read data (upper order)	
	CRC (upper order)	:	
		CRC (lower order)	
		CRC (upper order)	

Notes:
1) In case of abnormal response, 80hex is added to function code (F_CODE_O).
2) Function 03/04 does not use the number of bytes (LEN2_I). Set "WORD#16#0000" for the LEN2_I terminal. Similarly, all the variables that are connected to the "send data" terminal (S_DAT) shall be set to zero.
3) CRC is automatically calculated and added to the data by this FB.
4) Request message specifies the number of words by data length (LEN_I); response message stores it as the number of bytes of read data in the upper order byte of data length (LEN_O).
5) Read data is received in the order of upper order byte and lower order byte, as shown below. Therefore, word data can be accessed in units of word as it is.

Recieve data (R_DAT)→	Upper order byte	Lower order byte
	Upper order byte	Lower order byte
	Upper order byte	Lower order byte
	:	:

3) Forcible ON/OFF (1 bit) (code: 05)

Forcibly sets ON/OFF a bit (coil) that is designated by a slave station. The format of request message and response message is shown below.

FB terminals	Request message	Response message	FB terminals
Slave station No. (SIN_NO_I)→	Slave station No.	Slave station No.	→Slave station No. (SIN_NO_O)
Function code (F_CODE_I)→	Function code 05h	Function code 05h	→Function code (F_CODE_O)
Start address (ADD_I)→	Coil address (upper order)	Coil address (upper order)	→Data length (LEN_O)
	Coil address (lower order)	Coil address (lower order)	
Send data (S_DAT)→	FFh (ON)/00H (OFF) (upper order)	FFh (ON)/00H (OFF) (lower order)	→Receive data (R_DAT)
	00h (lower order)	00h (lower order)	
	CRC (lower order)	CRC (lower order)	
	CRC (upper order)	CRC (upper order)	

Notes:
1) In case of abnormal response, 80hex is added to function code (F_CODE_O).
2) Function 05 does not use data length (LEN_I) or the number of bytes (LEN2_I). Set "WORD#16#0000" for these terminals.
3) CRC is automatically calculated and added to the data by this FB.
4) Forcible ON command sets WORD#16#FF00 for the start word of send data. Forcible OFF command sets WORD#16#0000.

4) Retain register writing (1 word) (code: 06)

Writes data in the register (1 word) that is designated by a slave device. The format of request message and response message is shown below:

FB terminals	Request message	Response message	FB terminals
Slave station No. (SIN_NO_I)→	Slave station No.	Slave station No.	→Slave station No. (SIN_NO_O)
Function code (F_CODE_I)→	Function code 06h	Function code 06h	→Function code (F_CODE_O)
Start address (ADD_I)→	Register address (upper order)	Register address (upper order)	→Data length (LEN_O)
	Register address (lower order)	Register address (lower order)	
Send data (S_DAT)→	Write data (upper order)	Write data (upper order)	→Receive data (R_DAT)
	Write data (lower order)	Write data (lower order)	
	CRC (lower order)	CRC (lower order)	
	CRC (upper order)	CRC (upper order)	

Notes:
1) In case of abnormal response, 80hex is added to function code (F_CODE_O).
2) Function 06 does not use the data length (LEN_I) or the number of bytes (LEN2_I). Set "WORD#16#0000" for the LEN2_I terminal.
3) CRC is automatically calculated and added to the data by this FB.
4) For send data, set the data that is to be written in the start word. Other data shall be set to all zero.

5) Forcible ON/OFF (multiple bit) (code: 0F)

Forcibly sets ON/OFF a bit (coil) that is designated by a slave station. The format of request message and response message is shown below.

FB terminals	Request message	Response message	FB terminals
Slave station No. (SIN_NO_I)→	Slave station No.	Slave station No.	→Slave station No. (SIN_NO_O)
Function code (F_CODE_I)→	Function code 0Fh	Function code 0Fh	→Function code (F_CODE_O)
Start address (ADD_I)→	Start coil address (upper order)	Start coil address (upper order)	→Start address (ADD_O)
	Start coil address (lower order)	Start coil address (lower order)	
Data length (LEN_I)→	No. of written bits (upper order)	No. of written bits (upper order)	→Data length (LEN_O)
	No. of written bits (lower order)	No. of written bits (lower order)	
Number of bytes (LEN2_I)→	Number of bytes	CRC (lower order)	
	Write data (upper order)	CRC (upper order)	
Send data (S_DAT)→	Write data (lower order)		
	:		
	CRC (lower order)		
	CRC (upper order)		

Notes:
1) In case of abnormal response, 80hex is added to function code (F_CODE_O).
2) CRC is automatically calculated and added to the data by this FB.
3) Write data shall be set with upper- and lower-order bytes reversed, as shown below:

Send data (S_DAT)/Receive data (R_DAT) Offset address with respect to the designated start address.

+7	+6	+5	+4	+3	+2	+1	+0	+15	+14	+13	+12	+11	+10	+9	+8
+23	+22	+21	+20	+19	+18	+17	+16	+31	+30	+29	+28	+27	+26	+25	+24

6) Retain register writing (multiple words) (code: 10)

Writes data in consecutive registers starting with the one designated by a slave device. The format of request message and response message is shown below:

FB terminals	Request message	Response message	FB terminals
Slave station No. (SIN_NO_I)→	Slave station No.	Slave station No.	→Slave station No. (SIN_NO_O)
Function code (F_CODE_I)→	Function code 10h	Function code 10h	→Function code (F_CODE_O)
Start address (ADD_I)→	Start address (upper order)	Start address (upper order)	→Start address (ADD_O)
	Start address (lower order)	Start address (lower order)	
Data length (LEN_I)→	No. of written words (upper order)	No. of written words (upper order)	→Data length (LEN_O)
	No. of written words (lower order)	No. of written words (lower order)	
Number of bytes (LEN2_I)→	Number of bytes	CRC (lower order)	
	Write data (upper order)	CRC (upper order)	
Send data (S_DAT)→	Write data (lower order)		
	:		
	CRC (lower order)		
	CRC (upper order)		

Notes:

1) In case of abnormal response, 80hex is added to function code (F_CODE_O).
2) CRC is automatically calculated and added to the data by this FB.
3) Write data is received in the order of upper order byte and lower order byte, as shown below. Therefore, word data can be accessed in units of word as it is.

Send data (S_DAT)→	Upper order byte	Lower order byte
	Upper order byte	Lower order byte
	Upper order byte	Lower order byte
	:	:

7) Format in case of abnormal response

In case of abnormal response, the format of response message becomes as follows. This is common to all functions.

Abnormal response	FB terminals
Slave station No.	→Slave station No. (SIN_NO_O)
Function code + 80h	→Function code (F_CODE_O)
Error code	→Lower order byte of the start word of receive data (R_DAT)
CRC (lower order)	
CRC (upper order)	

<Error code list>

01h: Function code error
02h: Address setting error
03h: Read/write data count setting error

3-7-3 Initialization

(1) Set values of initialization parameters

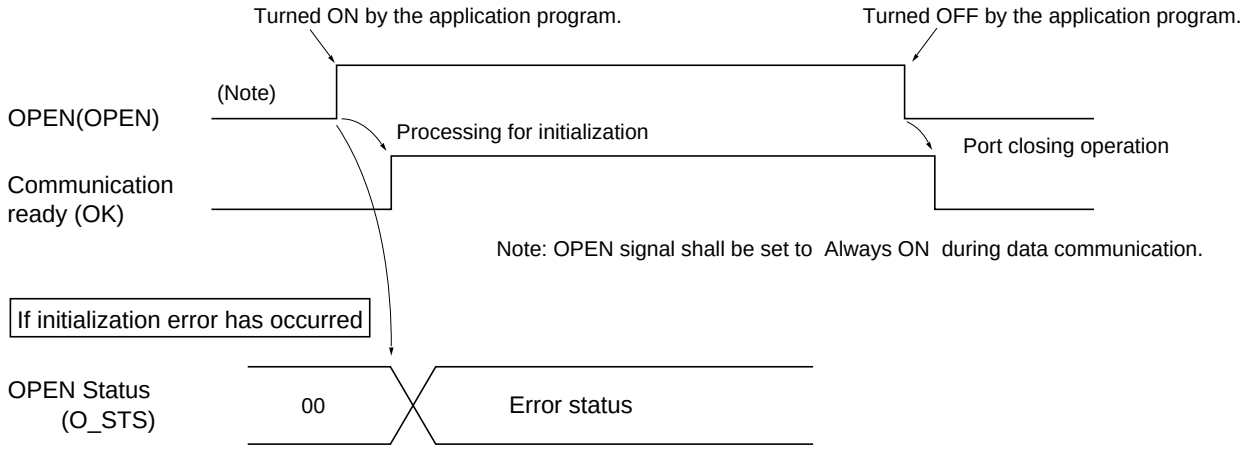
In order to initialize each individual port of the general purpose communication module, it is necessary to set values which match the communication specifications of the object external device to the individual communication parameters. The table below shows the setting contents of these parameters.

No.	Item		Description
0	General purpose communication module station No.		Sets the station number of the general purpose communication module on the SX bus.
1	Port No.		Designates an interface port on general purpose communication module. 0: RS-232C port 1: RS-485 port
2	Message port No. 1		Designates a port No. for sending/receiving messages to/from a general purpose communication module (1 to 127). Note: Avoid designating a port No. which is already used for sending/receiving messages to/from other module.
3	Reserved		Not used.
4	Transmission speed		Designates the transmission speed. 0: 1200 1: 2400 2: 4800 3: 9600 4: 19200 5: 38400 6: 57600 7: 76800 8: 115200 90: 300 91: 600 bps
5	Reserved		Not used
6	Parity bit		This is a bit which is added to data for the purpose of error detection. Designate the proper one to match the setting of the device at the other end. 0: None 1: Odd 2: Even
7	Stop bit		This bit indicates the end of data. Designate the proper one to match the setting of the device at the other end. 0: 1 bit 2: 2 bits
8	DCE designation		When signal line control is not performed, there is no difference in functions between DCE and DTE modes. The RS-232C of the general purpose communication module is of DTE specification, however, it can be used as a DCE specification interface when signal lines are converted as follows: No. 7 pin (RS) to CS No. 8 pin (CS) to RS No. 6 pin (DR) to ER No. 4 pin (ER) to DR 0: DTE 1: DCE 2: Modem DTE
9	ER/DR signal control		0: Disable initial values, 1: Enable initial values
10	Signal flow control	DTE mode	0: Off RS: always ON; Sending: unconditional 1: On RS: ON while sending; Sending: when CS is ON
		DCE mode	0: Off CS: always ON; Sending: unconditional 1: On CS: ON when RTS is ON; Sending: when ER is ON
11	XON/XOFF control		Because communication between sender and receiver is performed asynchronously, flow control may be necessary. The receiver sends an XOFF signal to indicate that it cannot receive data for a while and then sends an XON signal to cancel the XOFF condition. To use XON/OFF control, the device at the other end must also have this function. This prerequisite of XON/XOFF flow control is that XON or XOFF won't appear on the data. 0: NO 1: YES
12	RS-485 mode		Selects 4-wire or 2-wire for RS-485. 0: 4-wire 1: 2-wire
13 ⋮ 34	Reserved		Not used
35	Responce monitoring timer		This timer for monitoring during the period from when the CPU module sends a send request to a device until communication ends. This value can be set in 0.01 -second steps. Example: For 1 second, set this value to "100".
36 ⋮ 39	Reserved		Not used

Note: No. in the above table is element number of the array variables for communication parameter.

(2) Initialization procedure

When the OPEN terminal of FB is turned ON, communication parameters are transferred to the general purpose communication module, which executes processing for initialization. If initialization is impossible due to a communication parameter setting error or hardware error, FB outputs the corresponding error code to the OPEN Status (O_STS).



(3) OPEN status list

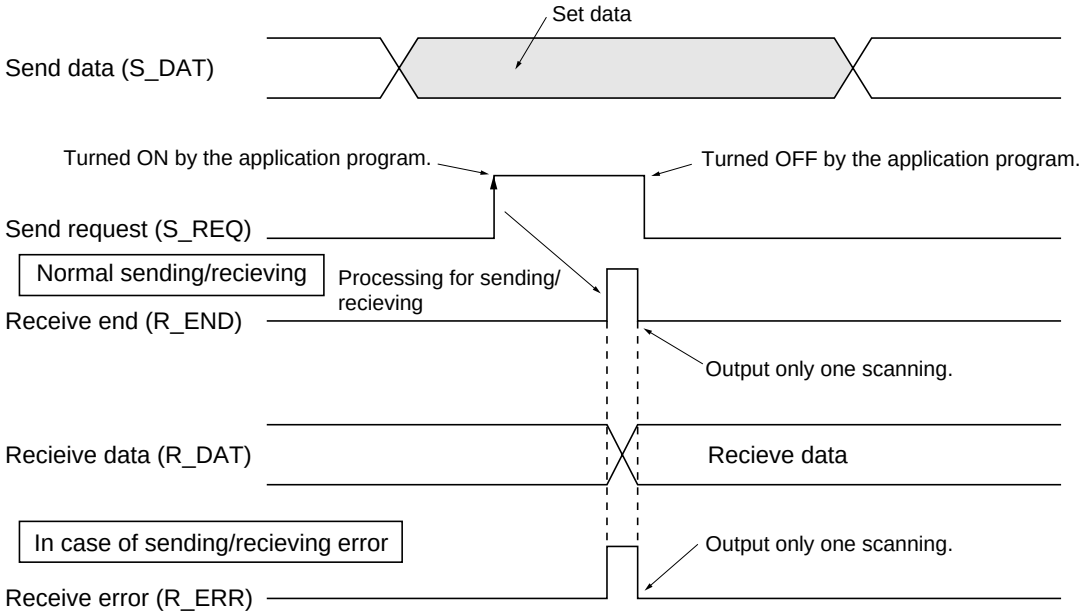
No.	Result of initialization	Remarks
'00'	Ended normally	
'01'	Transmission speed error	General purpose communication module detected an error.
'02'	Data bit length error	General purpose communication module detected an error.
'03'	Parity bit error	General purpose communication module detected an error.
'04'	Stop bit error	General purpose communication module detected an error.
'05'	DCE designation error	General purpose communication module detected an error.
'06'	Signal flow control setting error	General purpose communication module detected an error.
'07'	XON/XOFF control setting error	General purpose communication module detected an error.
'08'	RS-485 mode setting error	General purpose communication module detected an error.
'3F'	Parameter modification error	Parameter modification error after OPEN
'48'	Response monitoring timer value setting error	Specify the value of the minus.
'80'	General purpose communication module station No. setting error	
'81'	Channel No. setting error	
'82'	Message port No. setting error	
'93'	Open error	M_OPEN error
'94'	Open error	Initialization failed due to abnormality on general purpose communication module
'A3'	Processing impossible because in code conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.

3-7-4 Data communication

MODBUS protocol defines master and slave stations, and communication is performed in such manner that master station (MICREX-SX) starts communicating and slave stations respond to it. The transmission procedure is outlined below.

1) Master station sends a command message to slave station.
2) Slave station checks whether or not the station No. included in the received message coincides with its own station number and, if coincides, executes the command and then sends a response message to the master station.
3) If not coincides, the slave station discards the received message and waits for next command message.

(1) Data sending/receiving procedure



After send data is set, send request is turned ON by the application program. FB detects the rising edge to execute the processing for sending.

When sending operation ends and response message is received from the slave station, receive end flag is turned ON (for only one pulse). In case of receive error, receive end and receive error flags are turned ON (for only one pulse), and the corresponding error code is output to the receive status area ("00" when ended normally).

(2) Receiving status list

No.	Result of receiving	Remarks
'00'	Ended normally	
'10'	Receive buffer has overflowed.	
'11'	Unsupported function was sent.	
'12'	Unsupported function was received.	
'13'	Send station No. error	
'14'	Length 2 setting error	
'41'	Time up of response monitoring timer	
'42'	CRC error	
'90'	General purpose communication module disconnected.	
'91'	SX bus send error	
'92'	SX bus receive error	
'A0'	Object port error	General purpose communication module detected an error.
'A3'	Processing impossible because in conversion mode.	General purpose communication module detected an error.
'A5'	Processing impossible because in loader mode.	General purpose communication module detected an error.
'A6'	Processing impossible because self-diagnosis is being executed.	General purpose communication module detected an error.
'C0'	Hardware error detected during receiving	General purpose communication module detected an error.
'C1'	Receive buffer has overflowed.	General purpose communication module detected an error.
'C2'	Send buffer has overflowed.	General purpose communication module detected an error.
'C3'	Send data size over	General purpose communication module detected an error.

3-7-5 RAS information

RAS information area is used as the work memory for FB operations as well as for checking the operating condition of FBs. To use RAS information from the application program, it is necessary to declare variables in the following manner:

<Example of variable declaration>

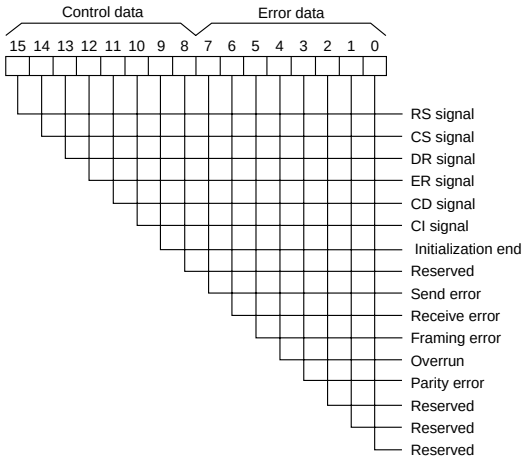
```
VAR
    RAS:     _C_work_TYPE11MD;
END_VAR
```

<RAS information area (INT array)>

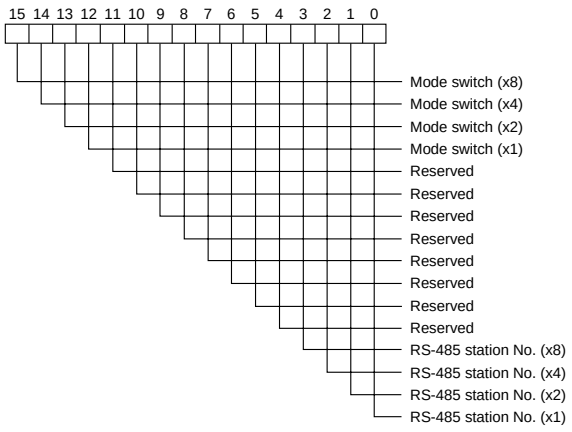
For the variable declaration shown above, the array variable in which RAS information is stored becomes "RAS". Individual element stores the following RAS information.

RAS.RAS[0]	Port status
RAS.RAS[1]	Status of general purpose communication module
RAS.RAS[2]	Send request count
RAS.RAS[3]	Send end count
RAS.RAS[4]	Receive count
RAS.RAS[5]	Frame detection count
RAS.RAS[6]	M_OPEN status
RAS.RAS[7]	M_SEND status
RAS.RAS[8]	M_RECEIVE status
RAS.RAS[9]	M_SEND error count
RAS.RAS[10]	M_RECEIVE error count
RAS.RAS[11]	Reserved
RAS.RAS[12]	Reserved
RAS.RAS[13]	Reserved
RAS.RAS[14]	Reserved
RAS.RAS[15]	Reserved
RAS.RAS[16]	Reserved
RAS.RAS[17]	Reserved
RAS.RAS[18]	Reserved
RAS.RAS[19]	Reserved

• Port status



• Status of general purpose communication module



Fuji Electric Co.,Ltd.

ED & C • Drive Systems Company

Gate City Ohsaki, East Tower, 11-2, Osaki 1-chome, Shinagawa-ku, Tokyo 141-0032, Japan
Phone: +81-3-5435-7135 ~ 8
Fax: +81-3-5435-7456 ~ 9
URL http://www.fujielectric.co.jp/kiki/

2002-7