

# **9: MAINTENANCE COMMUNICATION**

## **Introduction**

This chapter describes the SmartAXIS maintenance communication function.

Maintenance communication is a communication protocol dedicated for IDEC programmable controllers. It is used when WindLDR or an IDEC operator interface communicates with the SmartAXIS.

Maintenance communication of SmartAXIS is available on USB port (WindLDR only), expansion communication ports, and Ethernet port, allowing the optimum communication method to be selected for a variety of system configurations.

## **Maintenance Communication Functions**

When performing maintenance communication with the SmartAXIS, the following functions can be used:

Function	Description
Download user programs	User programs created in WindLDR can be downloaded to the SmartAXIS. See "Download Program" on page 4-7.
Upload user programs	User programs stored in the SmartAXIS can be uploaded to WindLDR.
Monitor/change device values	The user program and the device values of the SmartAXIS can be monitored and the device values can be changed using WindLDR. See "Monitor Operation" on page 4-8.
Download system software	System software can be downloaded to the SmartAXIS. See "Upgrade SmartAXIS System Software" on page A-2.

**Note:** To use maintenance communication, see "Start WindLDR" on page 4-1 and perform setup.

## **Communication Ports Used For Maintenance Communication**

SmartAXIS supports the following communication ports:

Type	USB Port (Port 1)	Expansion Communication Ports (Note)		Ethernet Port
		Port 2	Port 3	
12-I/O type	Yes	No	No	No
24-I/O type	Yes	Yes	No	Yes
40-I/O type	Yes	Yes	Yes	Yes
48-I/O type	Yes	Yes	Yes	Yes

**Note:** Communication cartridges are required in order to use expansion communication ports.

Maintenance communication supports the following functions on each communication port:

Communication Port	Standard/Option	No. of Ports	Communication Settings
USB port	Standard	1	None
Expansion communication ports	Options FT1A-PC1: EIA RS232C (Mini DIN) FT1A-PC2: EIA RS485 (Mini DIN) FT1A-PC3: EIA RS485 (Terminal block)	12-I/O type: 0 24-I/O type: 1 40-I/O type: 2 48-I/O type: 2	Function area settings See "Maintenance Communication via Expansion Communication Port" on page 9-3.
Ethernet Port	12-I/O type: - 24-I/O type: Standard 40-I/O type: Standard 48-I/O type: Standard	1	Function area settings See "Maintenance Communication via Ethernet Port" on page 9-5.

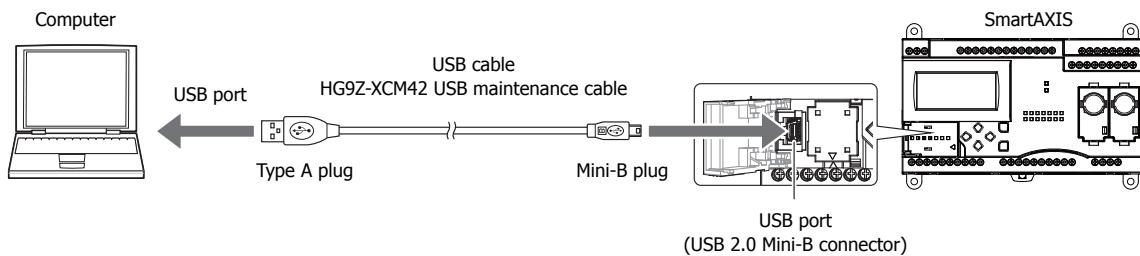
Communication Port	Maintenance Communication Functions		
	Monitor/Change Device Values	Download/Upload User Programs	Download System Software
USB port	Yes	Yes	Yes
Expansion communication ports	Yes	No	No
Ethernet port	Yes	Yes	No

## Maintenance Communication via USB Port

### Applicable SmartAXIS

FT1A-12	FT1A-24	FT1A-40	FT1A-48
X	X	X	X

Using the USB port, it is possible to connect SmartAXIS to a computer on which WindLDR is installed, and monitor and change device values, download and upload user programs, and download system software. Connect the computer and the SmartAXIS using a USB cable (recommended cable: HG9Z-XCM42).



### Maintenance Communication Specifications for the USB Port

Item	Specifications/Functions
Cable	Recommended cable: HG9Z-XCM42
Maintenance communication functions	Monitor/change device values Download/upload user programs Download system software

See the following pages for details on each maintenance communication function:

- Monitor/change device values: "Monitor Operation" on page 4-8
- Download/upload user programs: "Download Program" on page 4-7
- Download system software: "Upgrade SmartAXIS System Software" on page A-2

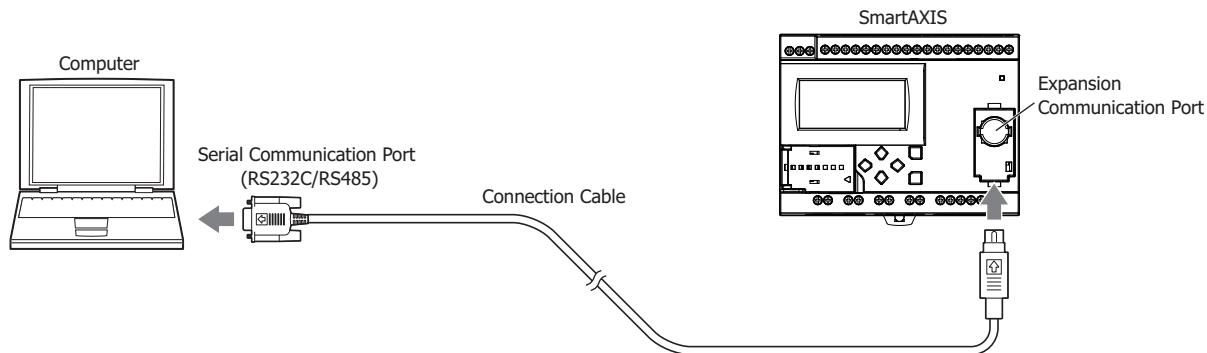
## Maintenance Communication via Expansion Communication Port

### Applicable SmartAXIS

FT1A-12	FT1A-24	FT1A-40	FT1A-48
—	X (Port 2)	X (Port 2 and 3)	X (Port 2 and 3)

By installing an RS232C or RS485 communication cartridge to a expansion communication port on the SmartAXIS, it is possible to connect SmartAXIS to a computer or an operator interface with an RS232C or RS485 port, and monitor and change device values of SmartAXIS.

For details on communication cables, see "Cables" on page A-8.



### Maintenance Communication Specifications for Expansion Communication Ports

Item	Specifications/Functions
Communication cartridge	FT1A-PC1: RS232C communication cartridge (Mini DIN type) FT1A-PC2: RS485 communication cartridge (Mini DIN type) FT1A-PC3: RS485 communication cartridge (Terminal block type) (Note)
Cable	FC2A-KC4C: RS232C communication cable (Mini DIN type) FC2A-KP1C: RS485 communication cable (Mini DIN type)
Maintenance communication functions	Monitoring/changing device values

**Note:** Use shielded twisted-pair cables.

See the following pages for details on the maintenance communication function:

- Monitor/change device values: "Monitor Operation" on page 4-8

## Programming WindLDR

Configure the settings for maintenance communication.

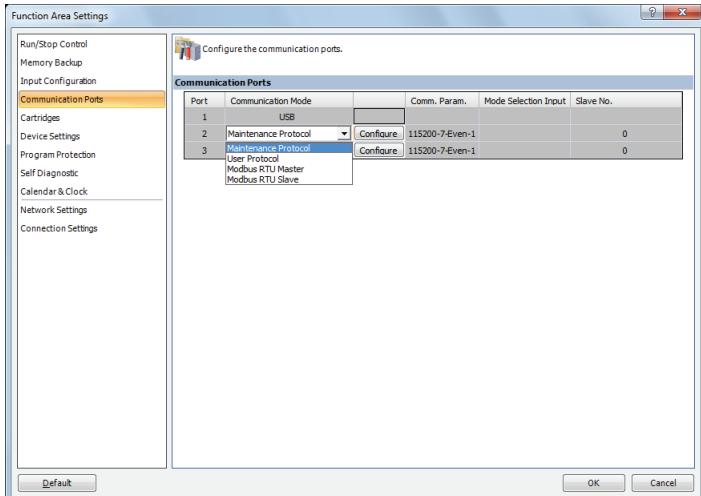
### Setting Procedure

1. From the WindLDR menu bar, select **Configuration > Comm. Ports**.  
The Function Area Settings dialog box appears.

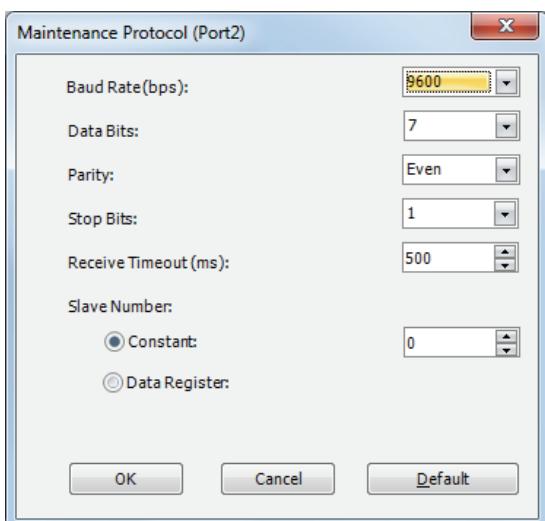
## 9: MAINTENANCE COMMUNICATION

2. In the **Communication Mode** pull-down list for Port 2, select **Maintenance Protocol**.

The Maintenance Communication (Port 2) dialog box appears.



3. Configure the parameters to match the communication settings of the computer or operator interface.



Baud Rate (bps): 115200bps (1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200)

Data Bits: 7 (7 or 8)

Parity: Even (None, Even, Odd)

Stop Bits: 1 (1 or 2)

Receive Timeout (ms): 500 (10 to 2550)

Slave Number: 0 (0 to 31)

Slave number can be specified by either a constant or a data register.

Type	Details
Constant	Set within the range of 0 to 31
Data register	Store the slave numbers 0 to 31 in the following special data registers Port 2: D8027 Port 3: D8028

### Notes:

- Values not in parentheses are the default settings.
- The following configuration cannot be selected: Data Bits: 7, Parity: None

4. Click **OK**.

Configuring the maintenance communication for the expansion communication port is now complete.

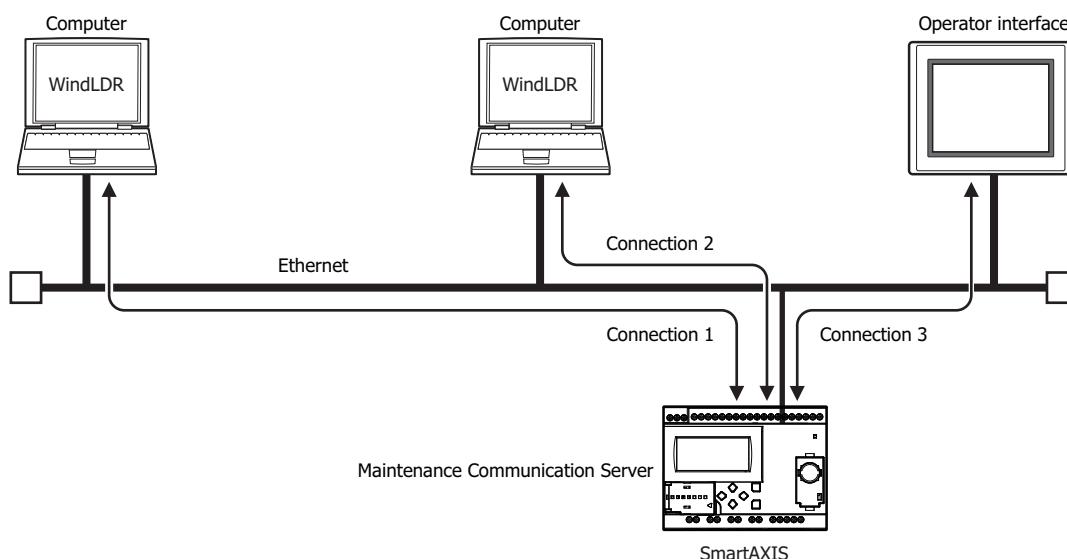
## Maintenance Communication via Ethernet Port

### Applicable SmartAXIS

FT1A-12	FT1A-24	FT1A-40	FT1A-48
—	X	X	X

Network devices such as computers or IDEC operator interfaces can communicate with SmartAXIS via Ethernet. External devices on the network can monitor or change the device values and download or upload user programs.

It is possible to use the maintenance communication server and other communications simultaneously by assigning a separate communication function such as a maintenance communication server or Modbus TCP communication to each of the three connections on the SmartAXIS.



### Maintenance Communication Specifications for the Ethernet Port

Item	Specifications/Functions
Cable	LAN cable
Maintenance communication functions	Monitor/change device values Download/upload user programs

See the following pages for details on each maintenance communication function:

- Monitor/change device values: "Monitor Operation" on page 4-8
- Download/upload user programs: "Download Program" on page 4-7

### Programming WindLDR

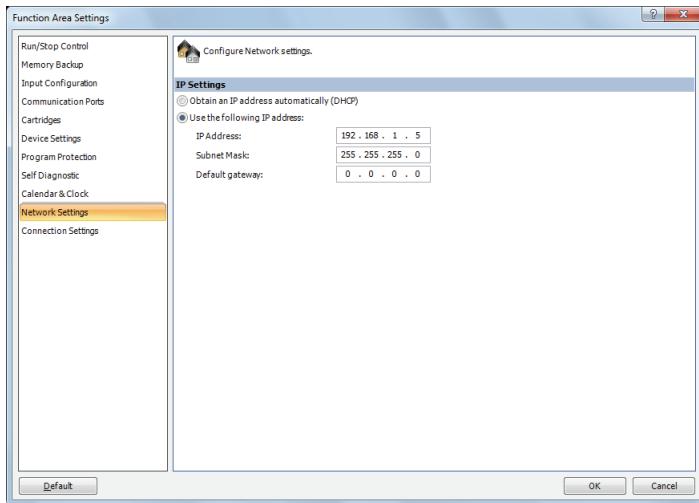
This section describes the procedures to configure the maintenance communication server for the Ethernet port and communicate with the SmartAXIS via Ethernet.

#### Configure Maintenance Communication Server

- From the WindLDR menu bar, select **Configuration > Network Settings**.

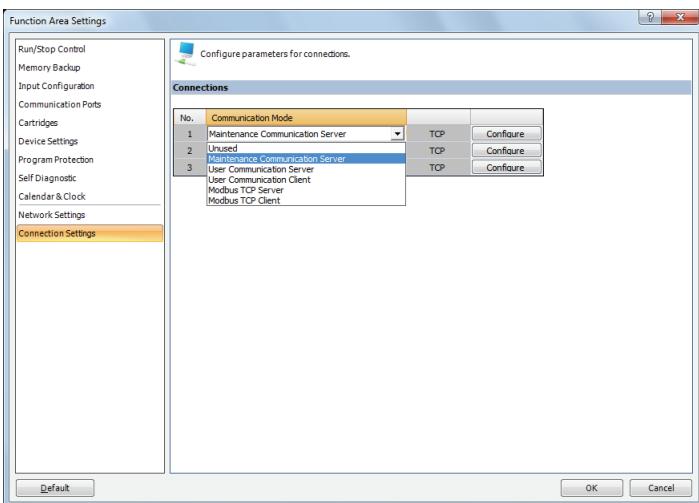
The Function Area Settings dialog box appears.

- Enter the IP address, subnet mask, and default gateway.

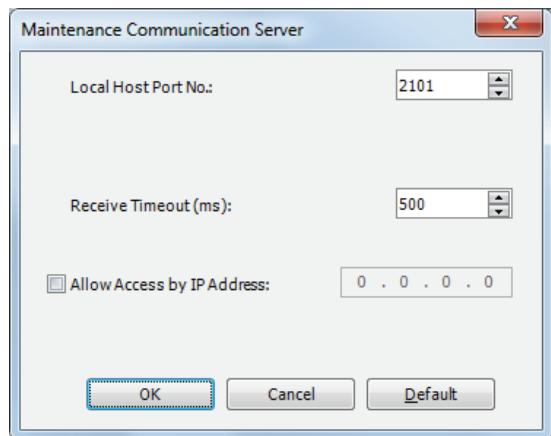


- Click **Connection Settings**.

- In the **Communication Mode** pull-down list for the connection to configure, select **Maintenance Communication Server**.  
The Maintenance Communication Server dialog box appears.



5. Set the parameters to match the communication settings of the PC or operator interface.



Local Host Port No.: 2101 (The port number that the SmartAXIS uses for the maintenance communication server)

Receive Timeout (ms): 2000 (100 to 25500ms)

Allow Access by IP Address: Disabled (Enabling this option makes it possible to prevent access from devices having any IP addresses other than the one entered.)

**Notes:**

- Settings not in parentheses are the default values.
- The number of clients that can be connected to the SmartAXIS simultaneously is one client per connection. If connections 1 to 3 are all set to the maintenance communication server, three clients can connect to the SmartAXIS at the same time.

6. Click **OK**.

This completes maintenance communications settings.

#### Download User Program and Confirm IP Addresses via USB Port

Before starting Ethernet communication, configure the function area settings and download the user program to the SmartAXIS via USB.

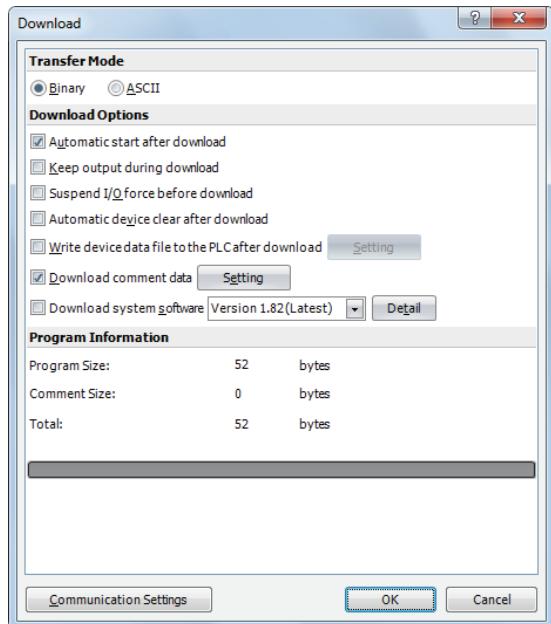
7. Connect the PC and the SmartAXIS using a USB cable.

8. From the WindLDR menu bar, select **Online > Transfer > Download**.

The Download dialog box appears.

**9. Click OK.**

The user program is downloaded to the SmartAXIS.



**10.** After the user program has been successfully downloaded, go to Monitor Mode to check the status of the SmartAXIS. Select **Online > Monitor > Monitor** from the WindLDR menu bar.

**11.** From the WindLDR menu bar, select **Online > Monitor > Batch**.

The Batch Monitor dialog box appears.

**12.** Confirm that the IP address entered in step 2 is correctly shown in D8084 to D8087.

Batch Monitor										
Device:	D (Special Data Register)		8084	Monitor Type:	DEC (W)					
<input type="checkbox"/> Comment										
	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9
D8034	0	0	0	0	0	0	1	0	0	0
D8044	0	0	0	0	0	0	0	0	0	0
D8054	0	0	0	0	0	0	0	0	0	0
D8064	0	0	0	0	0	0	0	0	0	0
D8074	0	0	0	0	255	255	255	255	255	255
D8084	192	168	1	5	255	255	0	0	0	0
D8094	0	0	0	0	0	0	0	0	0	0
D8104	0	0	0	0	0	0	0	0	0	0
D8114	0	0	0	0	0	0	0	0	0	0
D8124	0	0	0	0	0	0	0	0	0	0
D8134	0	0	0	0	0	0	0	0	0	0
D8144	0	0	0	0	0	0	0	0	0	0
D8154	0	0	0	0	0	0	0	0	0	0
D8164	0	0	0	0	0	0	0	0	0	0
D8174	0	0	0	0	0	0	0	0	0	0
D8184	0	0	0	0	0	0	0	0	0	0
D8194	0	0	0	0	0	0	0	0	0	0

### Monitor SmartAXIS via the Ethernet

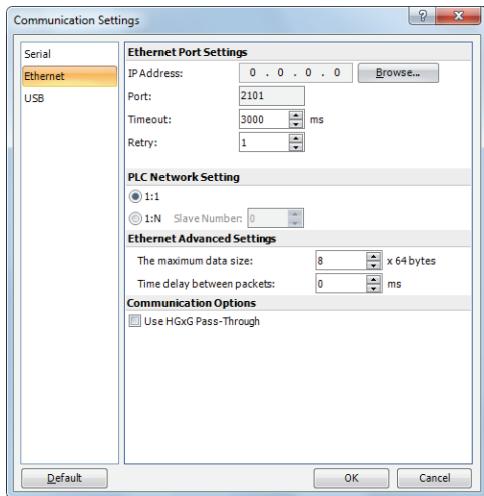
Monitor the SmartAXIS via Ethernet using WindLDR.

**13.** From the WindLDR menu bar, select **Online > Communication > Set Up**.

The Communication Settings dialog box appears.

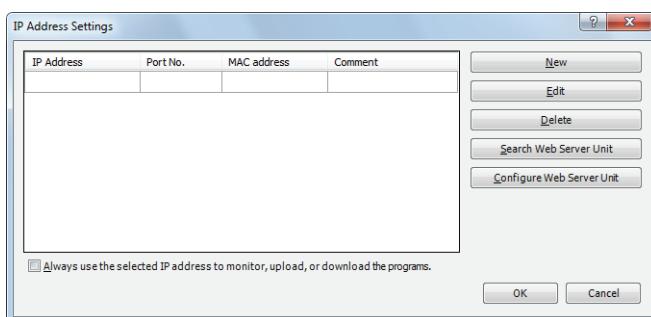
**14. Select the Ethernet tab and click Browse.**

The IP Address Settings dialog box appears.

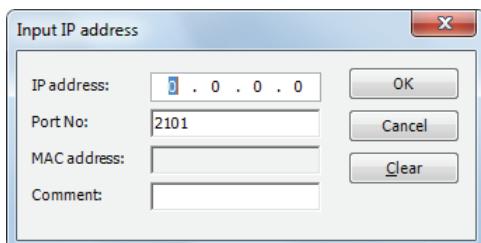


**15. Click New.**

The Input IP Address dialog box appears.



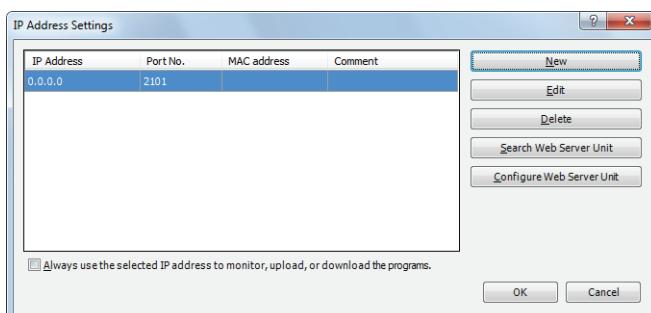
**16. Enter the IP address entered in step 2 and click OK.**



**17. From the WindLDR menu bar, select Online > Monitor > Monitor.**

The IP Address Settings dialog box appears.

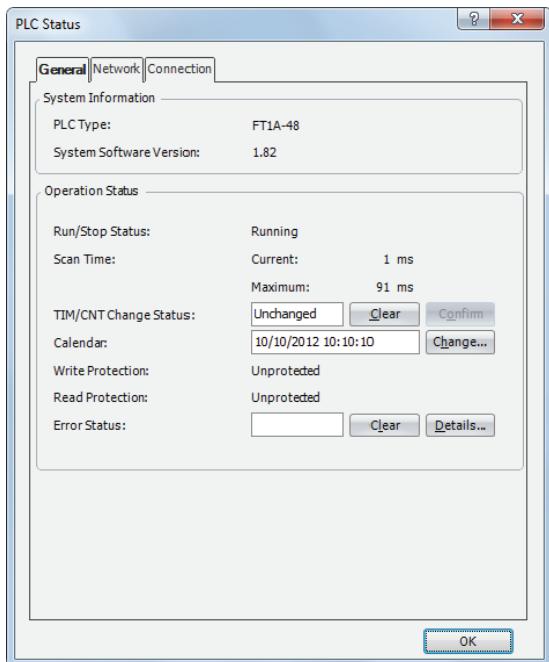
**18. Select the IP address you entered and click OK.**



19. From the WindLDR menu bar, select **Online > PLC > Status**.

The PLC Status dialog box appears.

20. Check that the SmartAXIS module type and system software version are displayed correctly.



Configuring the initial Ethernet setup for the maintenance communication server is now complete. You can download and upload user programs and monitor and change device values via Ethernet.

# **10: USER COMMUNICATION INSTRUCTIONS**

---

## **Introduction**

This chapter describes user communication that converts the specified data to the data type for the external devices connected to the SmartAXIS and sends and receives that data.

User communication allows the following two types of communication methods:

- Serial communication with an external device connected to the RS232C port or the RS485 port
- Ethernet communication with an external device connected by the Ethernet port

## **User Communication via Serial Communication**

This section describes the user communication function for communication between the SmartAXIS and external devices with an RS232C or RS485 port, such as a computer, modem, printer, or barcode reader. The SmartAXIS uses user communication instructions for transmitting and receiving communication to and from external devices.

## **User Communication Overview**

By installing a communication cartridge on the SmartAXIS expansion communication port, the SmartAXIS can communicate with two external devices simultaneously.

When using an RS485 communication cartridge, SmartAXIS modules can communicate with a maximum of 31 RS485 devices using the user communication.

User communication transmit and receive instructions can be programmed to match the communication protocol of the equipment to communicate with. Possibility of communication using the user communication mode can be determined referring to the user communication mode specifications described below.

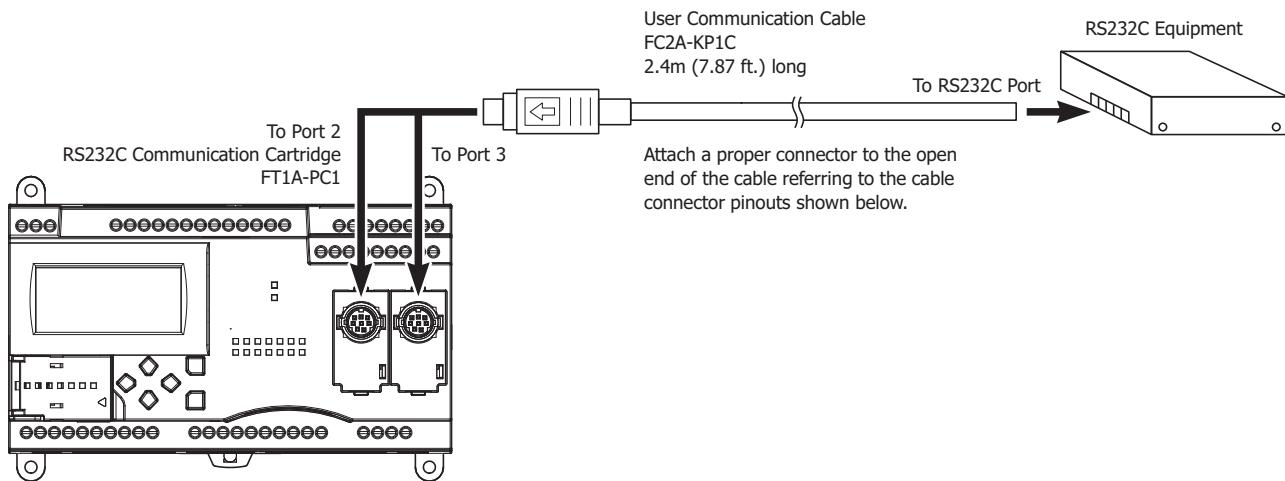
## **User Communication Mode Specifications**

Type	RS232C User Communication	RS485 User Communication
<b>Communication Port</b>	Port 2 and Port 3	Port 2 and Port 3
<b>Maximum Nodes</b>	1 per port	31 maximum
<b>Standards</b>	EIA RS232C	EIA RS485
<b>Baud Rate</b>	1200, 2400, 4800, 9600, 19200, 38400, 57600, 115200 bps (Default: 115200)	
<b>Data Bits</b>	7 or 8 bits (Default: 7) (Note)	
<b>Parity</b>	Odd, Even, None (Default: Even) (Note)	
<b>Stop Bits</b>	1 or 2 bits (Default: 1)	
<b>Receive Timeout</b>	10 to 2540ms (10ms increments) or none (Receive timeout is disabled when 2550 ms is selected.) The receive timeout has an effect when using RXD instructions.	
<b>Communication Method</b>	Start-stop synchronization system	
<b>Maximum Cable Length</b>	3m	200m
<b>Maximum Transmit Data</b>	200 bytes	
<b>Maximum Receive Data</b>	200 bytes	
<b>BCC Calculation</b>	XOR, ADD, ADD-2comp *, Modbus ASCII *, Modbus RTU * (* For calculation examples, see "BCC Calculation Examples" on page 10-28.)	

**Note:** The following configuration cannot be selected: Data Bits: 7, Parity: None

## RS232C User Communication System Setup

To connect an RS232C communication device to port 2 or 3 on the SmartAXIS, use the user communication cable FC2A-KP1C. One end of the user communication cable is not provided with a connector, and can be terminated with a proper connector to plug in to communicate with the RS232C port.



**Cable Connector Pinouts**

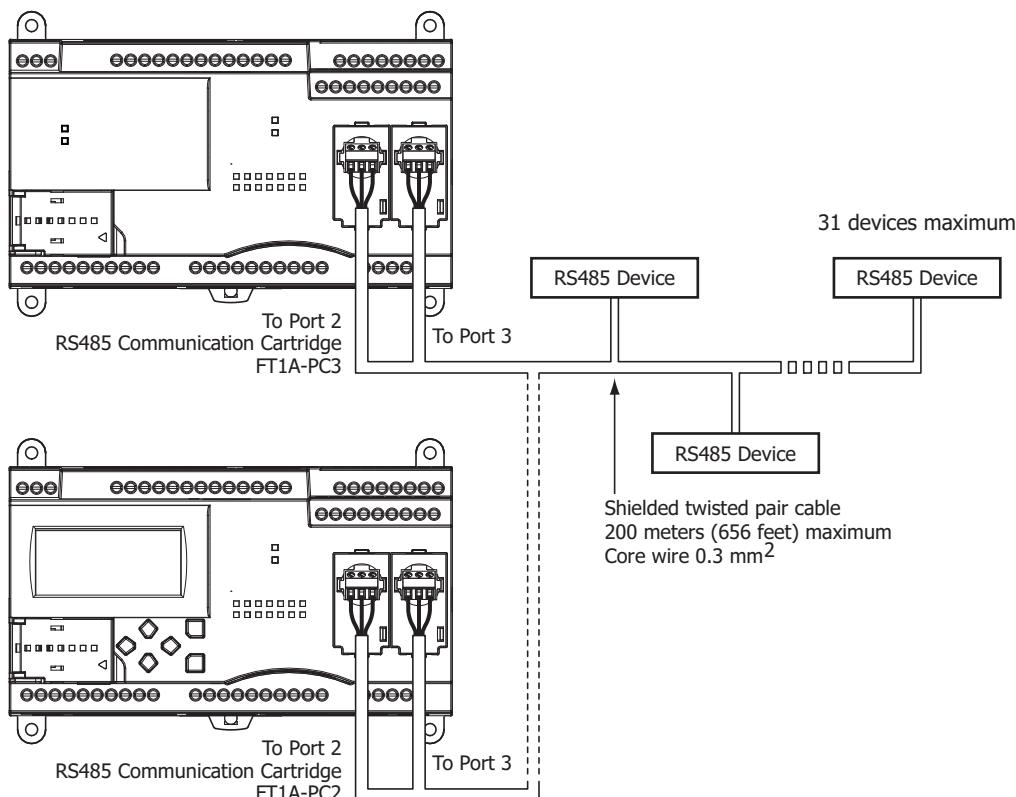
Pin	Port 2, Port 3	AWG#	Color	Signal Direction
1	RTS (request to send)	28	Black	→
2	DTR (data terminal ready)	28	Yellow	→
3	TXD (transmit data)	28	Blue	→
4	RXD (receive data)	28	Green	←
5	DSR (data set ready)	28	Brown	←
6	SG (signal ground)	28	Gray	—
7	SG (signal ground)	26	Red	—
8	NC (no connection)	26	White	—
<b>Cover</b>	—	—	Shield	—

## RS485 User Communication System Setup

Using the RS485 user communication, a maximum of 31 RS485 devices can be connected to the SmartAXIS.

When using port 2 or 3 for RS485 communication on the SmartAXIS, install the RS485 communication cartridge (FT1A-PC2 or FT1A-PC3) to the expansion communication port.

Connect RS485 device to the RS485 terminals A, B, and SG of port 2 or 3 on the SmartAXIS using a shielded twisted pair cable as shown below. When using the RS485 communication cartridge (FT1A-PC2) to the expansion communication port, use the user communication cable FC2A-KP1C. One end of the user communication cable is not provided with a connector, and can be terminated with a proper connector to plug in to communicate with the RS232C port.



**Cable Connector Pinouts**

Pin	Port 2, Port 3	AWG#	Color	Signal Direction
1	B	28	Black	→
2	A	28	Yellow	→
3	NC (no connection)	28	Blue	→
4	NC (no connection)	28	Green	←
5	NC (no connection)	28	Brown	←
6	NC (no connection)	28	Gray	—
7	SG (signal ground)	26	Red	—
8	NC (no connection)	26	White	—
Cover	—	—	Shield	—

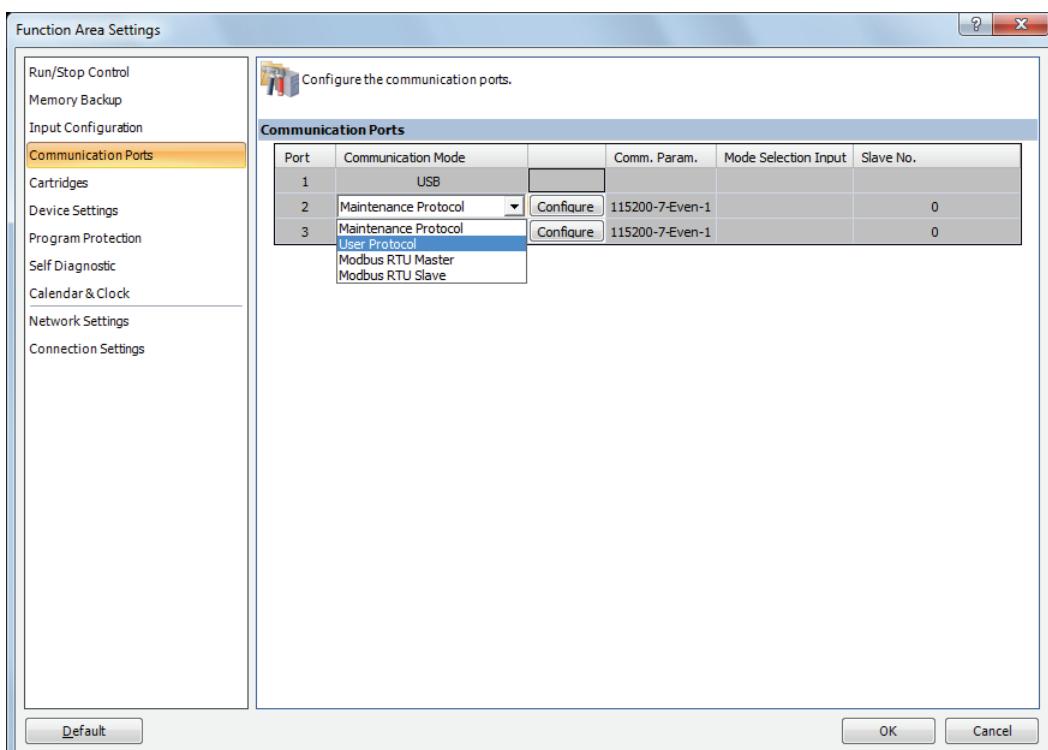
## Programming WindLDR

When using the user communication function to communicate with an external RS232C or RS485 device, set the communication parameters for the SmartAXIS to match those of the external device.

**Note:** Since communication parameters in the Function Area Settings relate to the user program, the user program must be downloaded to the SmartAXIS after changing any of these settings.

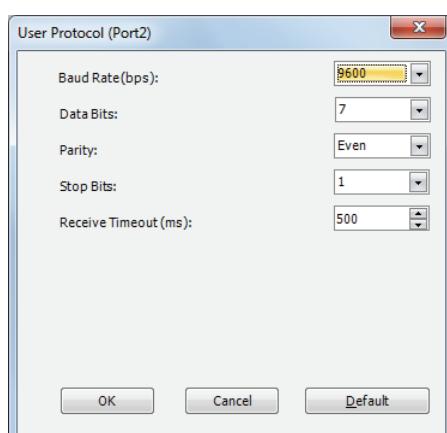
- From the WindLDR menu bar, select **Configuration > Communication Ports**.

The Function Area Settings dialog box for Communication Ports appears.



- In the Communication Mode pull-down list for Port 2 and Port 3, select **User Protocol**. (Click the **Configure** button when changing previous settings.)

The User Protocol dialog box appears.



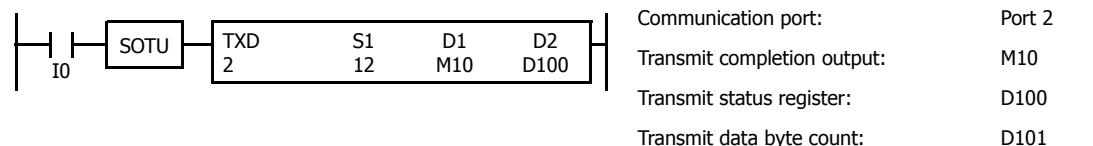
When **2550ms** is selected in the Receive Timeout box, the receive timeout function is disabled.

- Select communication parameters to the same values for the device to communicate with.
- Click the **OK** button.

## Programming TXD Instruction Using WindLDR

The following example demonstrates how to program a TXD instruction including a start delimiter, BCC, and end delimiter using WindLDR.

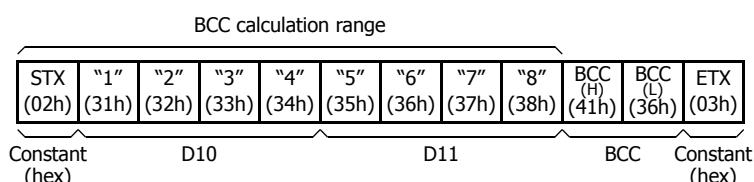
### TXD sample program:



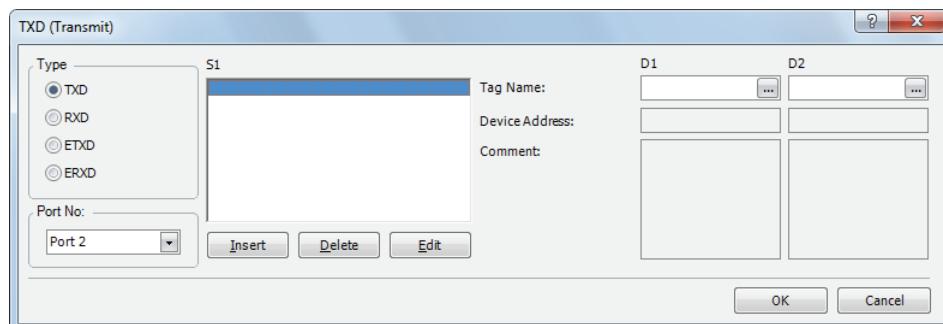
### Data register contents:

D10	<b>04D2h</b>	= 1234
D11	<b>162Eh</b>	= 5678

### Transmit data example:

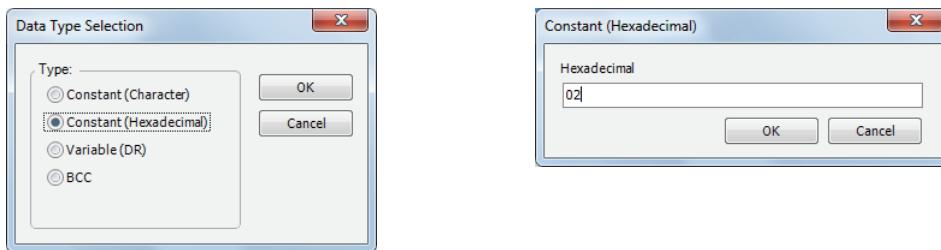


1. Start to program a TXD instruction. Move the cursor where you want to insert the TXD instruction, and type **TXD**. You can also insert the TXD instruction by clicking the User Communication icon in the menu bar and clicking where you want to insert the TXD instruction in the program edit area.  
The Transmit instruction dialog box appears.

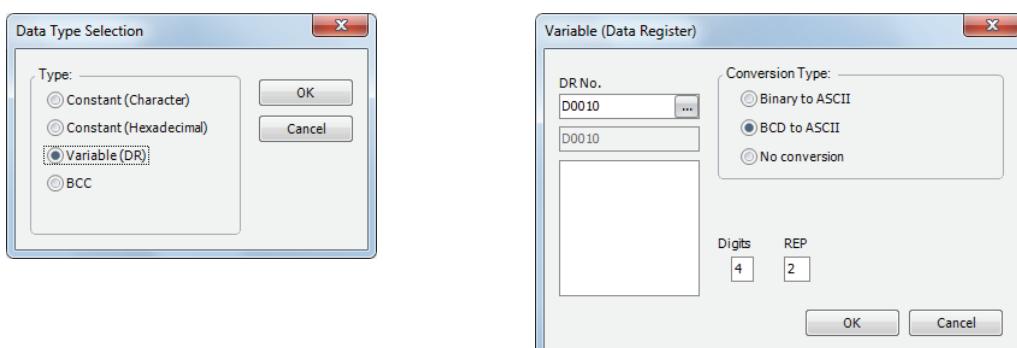


## 10: USER COMMUNICATION INSTRUCTIONS

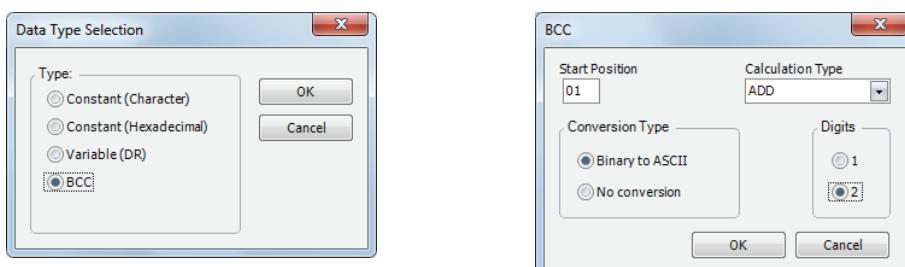
2. Check that **TXD** is selected in the Type box and select **Port 2** in the Port box. Then, click **Insert**.  
The Data Type Selection dialog box appears. You will program source device S1 using this dialog box.
3. Click **Constant (Hexadecimal)** in the Type box and click **OK**. Next, in the Constant (Hexadecimal) dialog box, type **02** to program the start delimiter STX (02h). When finished, click **OK**.



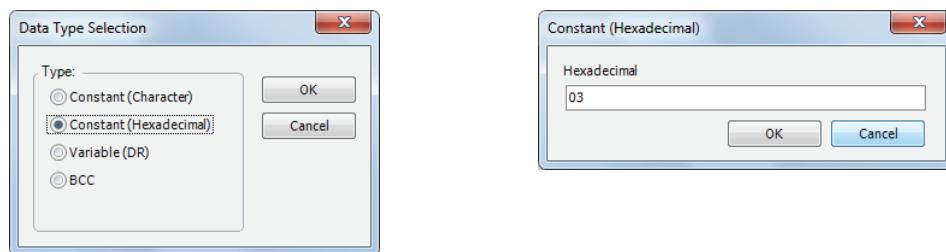
4. Since the Transmit instruction dialog box reappears, repeat the above procedure. In the Data Type Selection dialog box, click **Variable (DR)** and click **OK**. Next, in the Variable (Data Register) dialog box, type **D10** in the DR No. box and click **BCD to ASCII** to select the BCD to ASCII conversion. Enter **4** in the Digits box (4 digits) and **2** in the REP box (2 repeat cycles). When finished, click **OK**.



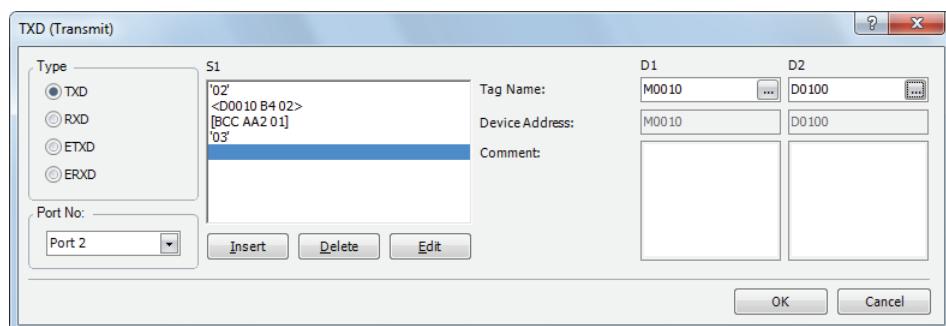
5. Again in the Data Type Selection dialog box, click **BCC** and click **OK**. Next, in the BCC dialog box, enter **1** in the Calculation Start Position box, select **ADD** for the Calculate Type, click **Binary to ASCII** for the Conversion Type, and click **2** for the Digits. When finished, click **OK**.



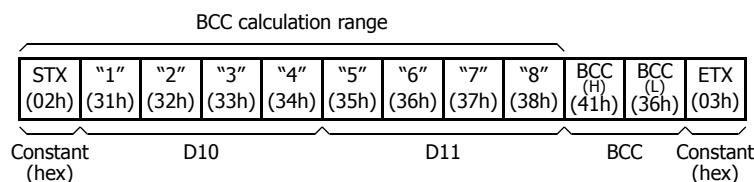
6. Once again in the Data Type Selection dialog box, click **Constant (Hexadecimal)** and click **OK**. Next, in the Constant (Hexadecimal) dialog box, type **03** to program the end delimiter ETX (03h). When finished, click **OK**.



7. In the Transmit instruction dialog box, type **M10** in the destination D1 box and type **D100** in the destination D2 box. When finished, click **OK**.



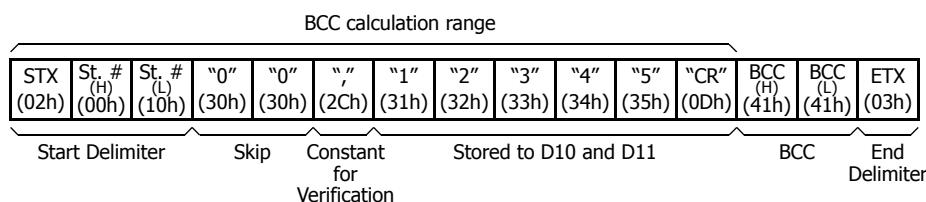
Programming of the TXD2 instruction is complete and the transmit data is specified as follows:



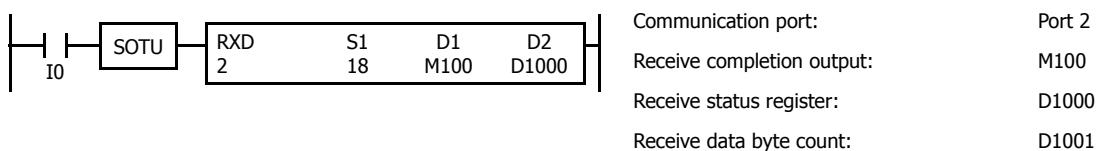
## Programming RXD Instruction Using WindLDR

The following example demonstrates how to program an RXD instruction including a start delimiter, skip, constant for verification, BCC, and end delimiter using WindLDR. Converted data is stored to data registers D10 and D11. Internal relay M100 is used as destination D1 for the receive completion output. Data register D1000 is used as destination D2 for the receive status, and data register D1001 is used to store the receive data byte count.

### Receive data example:

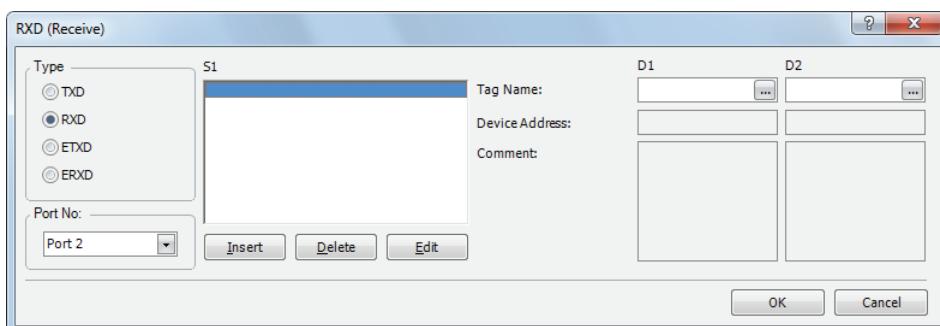


### RXD sample program:

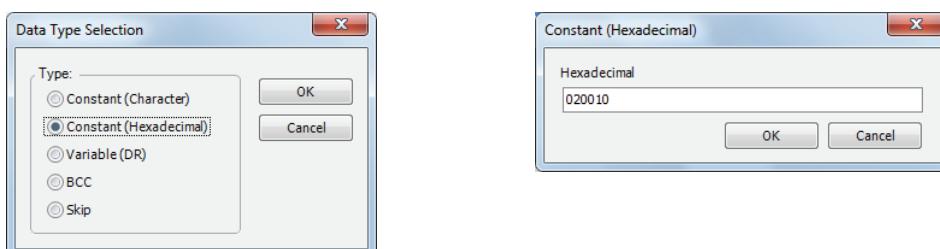


1. Start to program an RXD instruction. Move the cursor where you want to insert the RXD instruction, and type **RXD**. You can also insert the RXD instruction by clicking the User Communication icon in the menu bar and clicking where you want to insert the RXD instruction in the program edit area, then the Transmit dialog box appears. Click **RXD** to change the dialog box to the Receive dialog box.

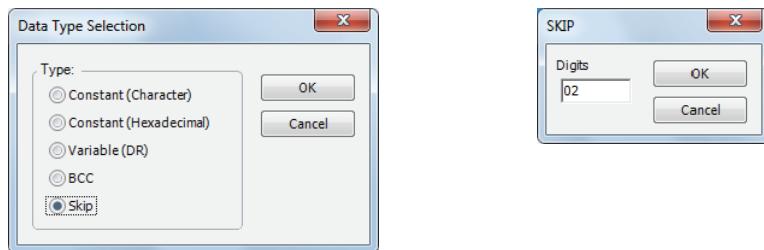
The Receive instruction dialog box appears.



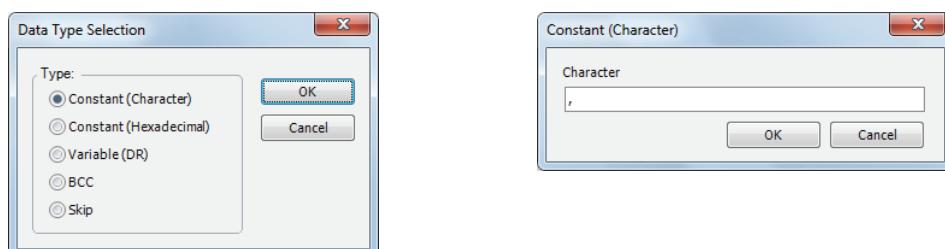
2. Check that **RXD** is selected in the Type box and select **Port 2** in the Port box. Then, click **Insert**. The Data Type Selection dialog box appears. You will program source device S1 using this dialog box.
3. Click **Constant (Hexadecimal)** in the Type box and click **OK**. Next, in the Constant (Hexadecimal) dialog box, type **020010**. When finished, click **OK**.



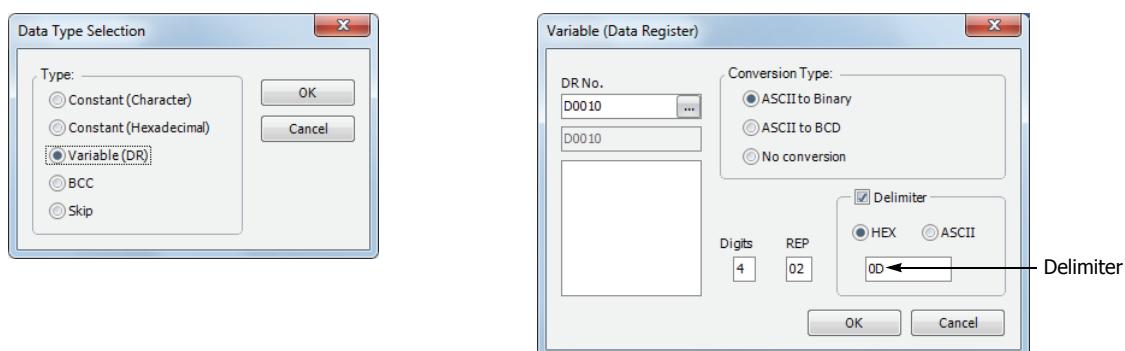
4. Since the Receive instruction dialog box reappears, repeat the above procedure. In the Data Type Selection dialog box, click **Skip** and click **OK**. Next, in the Skip dialog box, type **02** in the Digits box and click **OK**.



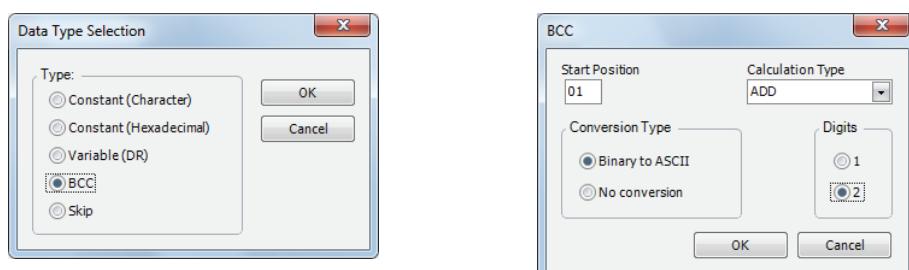
5. Again in the Data Type Selection dialog box, click **Constant (Character)** and click **OK**. Next, in the Constant (Character) dialog box, type **,** (**2Ch**) in the Character box to program a comma as a constant to verify. When finished, click **OK**.



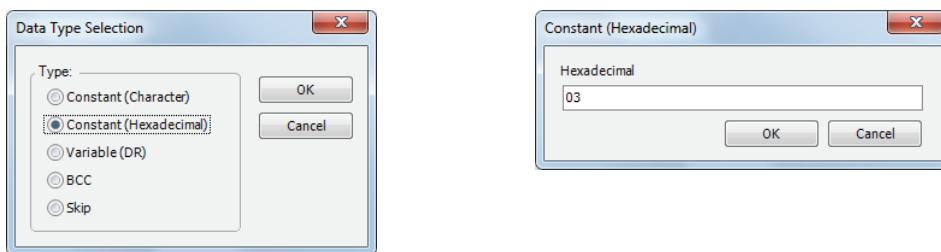
6. Again in the Data Type Selection dialog box, click **Variable (DR)** and click **OK**. Next, in the Variable (Data Register) dialog box, type **D10** in the DR No. box and click **ASCII to Binary** to select ASCII to binary conversion. Enter **4** in the Digits box (4 digits) and **2** in the REP box (2 repeat cycles). Click **Variable**, select **HEX**, and type **0D** to designate a delimiter. When finished, click **OK**.



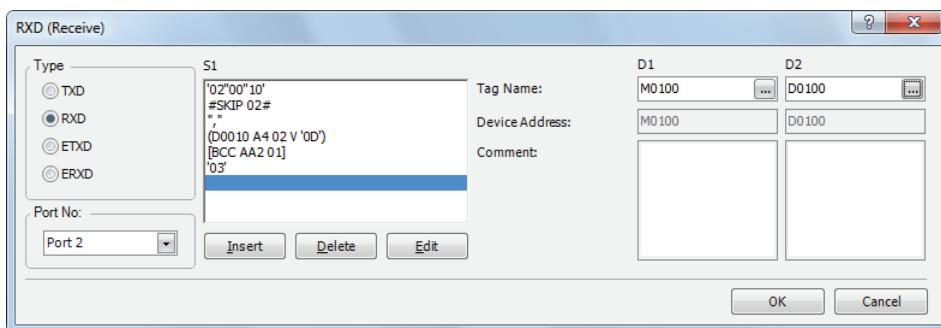
7. Again in the Data Type Selection dialog box, click **BCC** and click **OK**. Next, in the BCC dialog box, enter **1** in the Calculation Start Position box, select **ADD** for the Calculation Type, click **Binary to ASCII** for the Conversion Type, and click **2** for the Digits. When finished, click **OK**.



8. Once again in the Data Type Selection dialog box, click **Constant (Hexadecimal)** and click **OK**. Next, in the Constant (Hexadecimal) dialog box, type **03** to program the end delimiter ETX (03h). When finished, click **OK**.



9. In the Receive instruction dialog box, type **M100** in the destination D1 box and type **D1000** in the destination D2 box. When finished, click **OK**.



Programming of the RXD instruction is complete and the receive data will be stored as follows:

D10	<input type="text" value="1234h"/>	= 4660
D11	<input type="text" value="0005h"/>	= 5

## RS232C Line Control Signals

While the SmartAXIS is in user communication mode, special data registers can be used to enable or disable DSR and DTR control signal options for port 2 through port 3. The DSR and DTR control signal options cannot be used for port 1.

The RTS signal line of port 2 through port 3 remains on.

In the maintenance communication mode, DSR has no effect and DTR remains on.

### Special Data Registers for Port 2 to Port 3 RS232C Line Control Signals

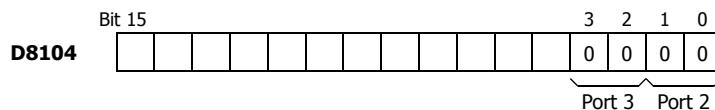
Special data registers D8104 through D8106 are allocated for RS232C line control signals.

Communication Port	DR No.	Data Register Function	Data Register Value Updated	R/W
Port 2 to Port 3	<b>D8104</b>	Control signal status	Every scan	R
	<b>D8105</b>	DSR input control signal option	When sending/receiving data	R/W
	<b>D8106</b>	DTR output control signal option	When sending/receiving data	R/W

#### Control Signal Status D8104

Special data register D8104 stores a value to show that DSR and DTR are on or off at port 2 through port 3.

The data of D8104 is updated at every END processing.



D8104 2-bit Binary Value	DTR	DSR	Description
<b>00</b>	OFF	OFF	Both DSR and DTR are off
<b>01</b>	OFF	ON	DSR is on
<b>10</b>	ON	OFF	DTR is on
<b>11</b>	ON	ON	Both DSR and DTR are on

#### DSR Control Signal Status in RUN and STOP Modes

Communication Mode	D8105 3-bit Binary Value	DSR (Input) Status	
		RUN Mode	STOP Mode
User Communication Mode	<b>000 (default)</b>	No effect	No effect (TXD/RXD disabled)
	<b>001</b>	ON: Enable TXD/RXD OFF: Disable TXD/RXD	No effect (TXD/RXD disabled)
	<b>010</b>	ON: Disable TXD/RXD OFF: Enable TXD/RXD	No effect (TXD/RXD disabled)
	<b>011</b>	ON: Enable TXD OFF: Disable TXD	No effect (TXD/RXD disabled)
	<b>100</b>	ON: Disable TXD OFF: Enable TXD	No effect (TXD/RXD disabled)
	<b>≥ 101</b>	No effect	No effect (TXD/RXD disabled)
Maintenance Mode	—	No effect	No effect

#### DTR Control Signal Status in RUN and STOP Modes

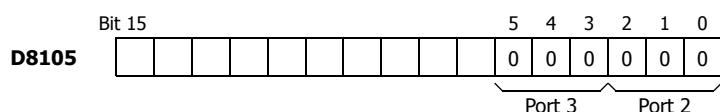
Communication Mode	D8106 2-bit Binary Value	DTR (Output) Status	
		RUN Mode	STOP Mode
User Communication Mode	<b>00 (default)</b>	ON	OFF
	<b>01</b>	OFF	OFF
	<b>10</b>	RXD enabled: ON RXD disabled: OFF	OFF
	<b>11</b>	ON	OFF
Maintenance Mode	—	ON	ON

### DSR Input Control Signal Option D8105

Special data register D8105 is used to control data flow between the SmartAXIS RS232C port 2 through port 3 and the remote terminal depending on the DSR (data set ready) signal sent from the remote terminal. The DSR signal is an input to the SmartAXIS to determine the status of the remote terminal. The remote terminal informs the SmartAXIS using DSR whether the remote terminal is ready for receiving data or is sending valid data.

The DSR control signal option can be used only for the user communication through the RS232C port 2 to port 3.

The control status of each port is allocated as shown below:



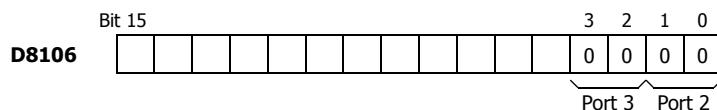
<b>D8105 3-bit Binary Value</b>	<b>Description</b>
<b>000</b>	DSR is not used for data flow control. When DSR control is not needed, set 0 to D8105.
<b>001</b>	When DSR is on, the SmartAXIS can transmit and receive data.  <p>DSR signal      ON OFF</p> <p>Transmit/receive      Impossible      Possible      Impossible</p>
<b>010</b>	When DSR is off, the SmartAXIS can transmit and receive data.  <p>DSR signal      ON OFF</p> <p>Transmit/receive      Impossible      Possible      Impossible</p>
<b>011</b>	When DSR is on, the SmartAXIS can transmit data. This function is usually called "Busy Control" and is used for controlling transmission to a remote terminal with a slow processing speed, such as a printer. When the remote terminal is busy, data input to the remote terminal is restricted.  <p>DSR signal      ON OFF</p> <p>Transmit      Impossible      Possible      Impossible</p>
<b>100</b>	When DSR is off, the SmartAXIS can transmit data.  <p>DSR signal      ON OFF</p> <p>Transmit      Impossible      Possible      Impossible</p>
<b>≥ 101</b>	Same as D8105 = 000. DSR is not used for data flow control.

## DTR Output Control Signal Option D8106

Special data register D8106 is used to control the DTR (data terminal ready) signal to indicate the SmartAXIS operating status or transmitting/receiving status.

The DTR control signal option can be used only for the user communication through the RS232C port 2 to port 3.

The control status of each port is allocated as shown below:



D8106 2-bit Binary Value	Description								
<b>00</b>	While the SmartAXIS is running, DTR is on whether the SmartAXIS is transmitting or receiving data. While the SmartAXIS is stopped, DTR remains off. Use this option to indicate the SmartAXIS operating status.  <table border="1"> <tr> <td>SmartAXIS</td> <td>Stopped</td> <td>Running</td> <td>Stopped</td> </tr> <tr> <td>DTR signal</td> <td>ON</td> <td>OFF</td> <td>OFF</td> </tr> </table>	SmartAXIS	Stopped	Running	Stopped	DTR signal	ON	OFF	OFF
SmartAXIS	Stopped	Running	Stopped						
DTR signal	ON	OFF	OFF						
<b>01</b>	Whether the SmartAXIS is running or stopped, DTR remains off.  <table border="1"> <tr> <td>SmartAXIS</td> <td>Stopped</td> <td>Running</td> <td>Stopped</td> </tr> <tr> <td>DTR signal</td> <td>ON</td> <td>OFF</td> <td>OFF</td> </tr> </table>	SmartAXIS	Stopped	Running	Stopped	DTR signal	ON	OFF	OFF
SmartAXIS	Stopped	Running	Stopped						
DTR signal	ON	OFF	OFF						
<b>10</b>	While the SmartAXIS can receive data, DTR is turned on. While the SmartAXIS can not receive data, DTR remains off. Use this option when flow control of receive data is required.  <table border="1"> <tr> <td>Receive</td> <td>Impossible</td> <td>Possible</td> <td>Impossible</td> </tr> <tr> <td>DTR signal</td> <td>ON</td> <td>OFF</td> <td>OFF</td> </tr> </table>	Receive	Impossible	Possible	Impossible	DTR signal	ON	OFF	OFF
Receive	Impossible	Possible	Impossible						
DTR signal	ON	OFF	OFF						
<b>11</b>	Same as D8106 = 00.								

## User Communication via Ethernet Communication

This section describes the Ethernet user communication. Ethernet user communication works on TCP/IP protocol. The SmartAXIS can be used as a user communication client/server. With Ethernet user communication instructions (ETXD and ERXD instructions), the SmartAXIS can exchange the data with devices on the network.

Except for the port number and the allocation of the user communication receive instruction cancel flags, Ethernet user communication instructions (ETXD and ERXD instructions) are identical to TXD and RXD instructions. For details about TXD and RXD instructions, see Chapter 25 "User Communication Instructions" in the SmartAXIS Ladder Programming Manual.

## Ethernet User Communication Overview

The SmartAXIS can be used as an Ethernet user communication client/server. It can be used simultaneously with the maintenance communication server, Modbus TCP server, and Modbus TCP client.

When using the SmartAXIS user communication client, the SmartAXIS can access and communicate with the server devices using the protocol of the server device. A maximum of three client connections of the SmartAXIS can be allocated to user communication.

User communication client functions and configuration are described in "To use the SmartAXIS as a user communication client" on page 10-15. User communication server functions and configuration are described in "User Communication Server" on page 10-19.

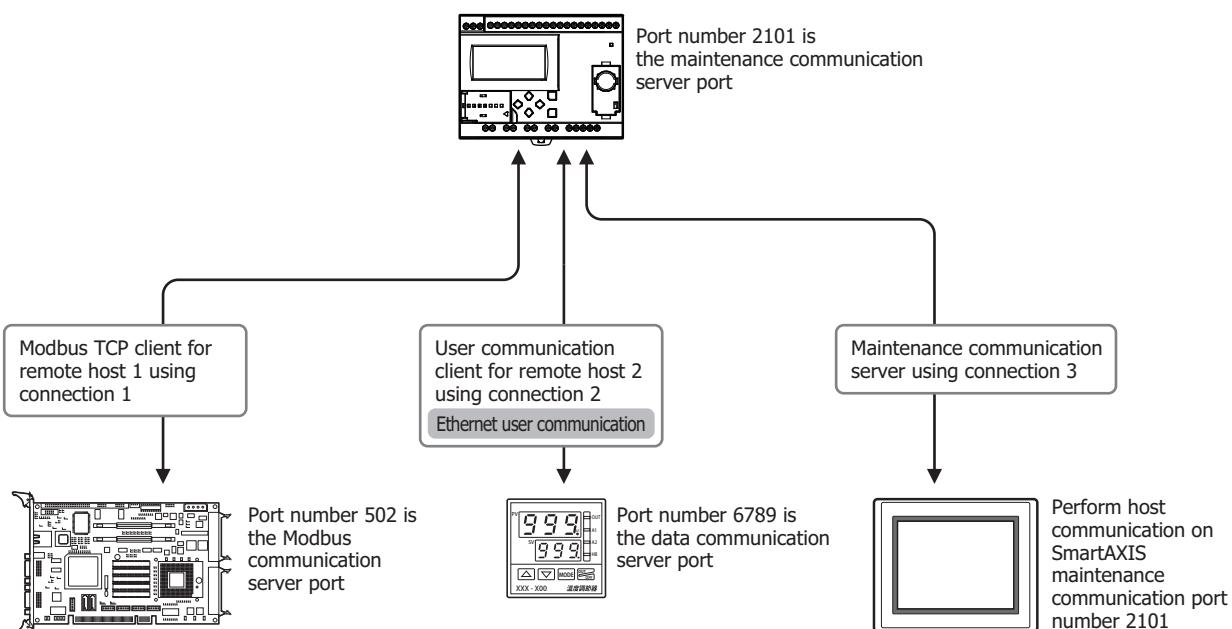
The SmartAXIS supports the TCP/IP protocol.

The SmartAXIS can send data to and receive data from devices on a network by using the ETXD (Ethernet user communication transmit) instruction and the ERXD (Ethernet user communication receive) instruction.

The SmartAXIS can be used as both an Ethernet user communication client and server.

Each of the three connections possessed by the SmartAXIS can be allocated to different types of communication. Ethernet user communication can simultaneously use the maintenance communication server, Modbus TCP server, and Modbus TCP client.

### [Ethernet communication example using three connections]



**SmartAXIS function area settings connection settings**

Connection	Communication protocol	Other settings
1	Modbus TCP client	Destination: Remote host 1
2	User communication client	Destination: Remote host 2
3	Maintenance communication server	Port number: 2101

**Remote host table**

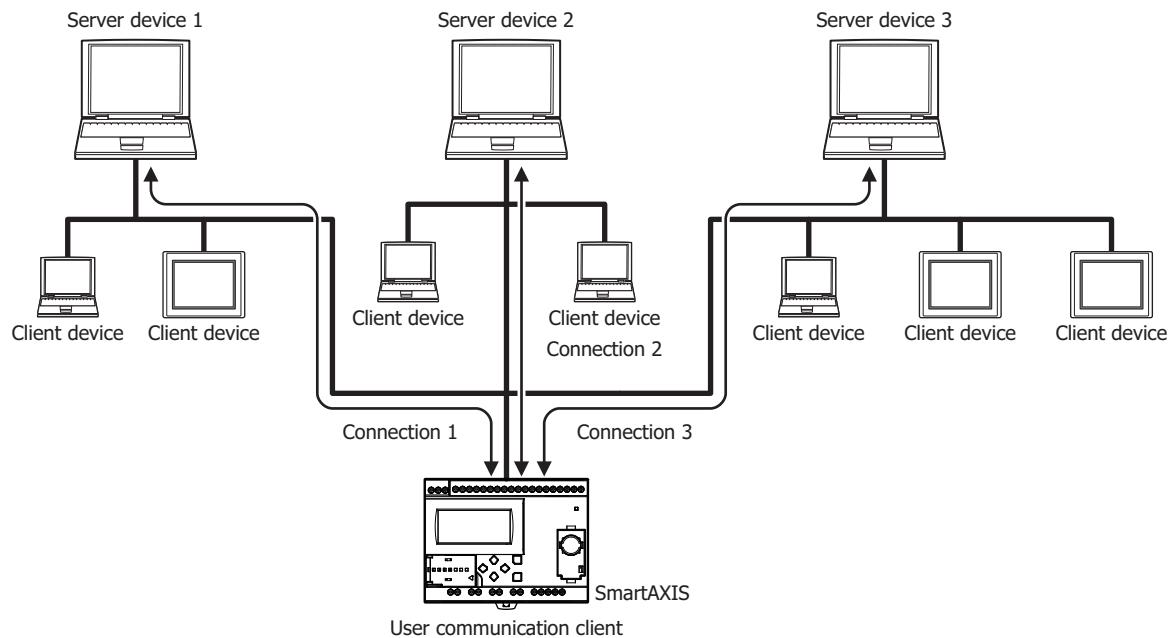
Remote host number	IP address	Port number
1	192.168.0.12	502
2	192.168.0.13	6789

## To use the SmartAXIS as a user communication client

Connect the SmartAXIS to the server device via the network and communicate with the server device using the Ethernet user communication instructions.

A maximum of three connections can be allocated to user communication clients. The SmartAXIS can simultaneously connect to and communicate with three different server devices.

### [When three connections are allocated to user communication client]

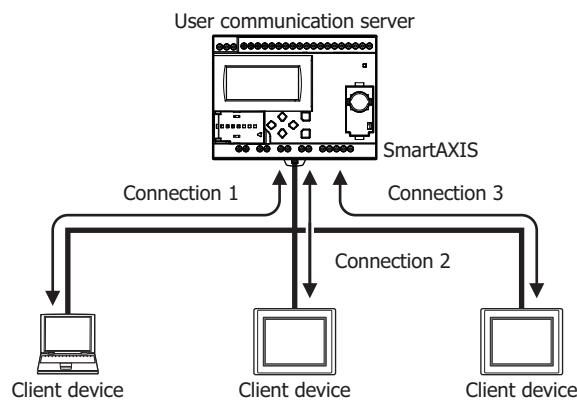


## To use the SmartAXIS as a user communication server

The client devices connect to the SmartAXIS and the SmartAXIS communicates with the client devices using Ethernet user communication instructions.

A maximum of three connections can be allocated to user communication servers. A maximum of three client devices can simultaneously connect to and communicate with the SmartAXIS.

### [When three connections are allocated to user communication server]



## User Communication Client

When a client connection is configured as the user communication client, the SmartAXIS communicates with the specified server device according to the settings configured in the ETXD and ERXD instructions that are programmed for the client connection. The remote host number and other communication settings can be configured in the Connection Settings tab in the Function Area Settings dialog box.

### Specifications (User Communication Client)

Item	User Communication Client
<b>Remote host number</b>	1 to 255
<b>Establish connection</b>	<ul style="list-style-type: none"><li>• When ETXD/ERXD Instructions are executed</li><li>• When the SmartAXIS starts to run (Note 1)</li></ul>
<b>Disconnect connection</b>	<ul style="list-style-type: none"><li>• When the SmartAXIS is stopped</li><li>• When special internal relays (M8130 to M8132) are turned on</li></ul>
<b>Number of remote hosts that the SmartAXIS can communicate simultaneously</b>	One remote host per a user communication client
<b>Receive timeout</b>	100 to 25500 ms (100 ms increments); Default: 1000 ms

**Note 1:** The settings can be configured in the Connection Settings tab in the Function Area Settings dialog box.

### Establishing/Disconnecting User Communication Client Connections

When user communication clients are configured, connections are established on TCP/IP protocol. The connections are established when ETXD/ERXD instructions are executed or when the SmartAXIS has started to run (See Note 1 above). After a connection has been established, the connection will be kept open until either the SmartAXIS is stopped or a special internal relay allocated to the connection is turned on.

Device Address	Description	Operation
<b>M8130</b>	Connection 1 Disconnect	
<b>M8131</b>	Connection 2 Disconnect	
<b>M8132</b>	Connection 3 Disconnect	When the reply is turned on, the corresponding connection is disconnected.

### User Communication Client Receive Instruction (ERXD) Cancel Flag

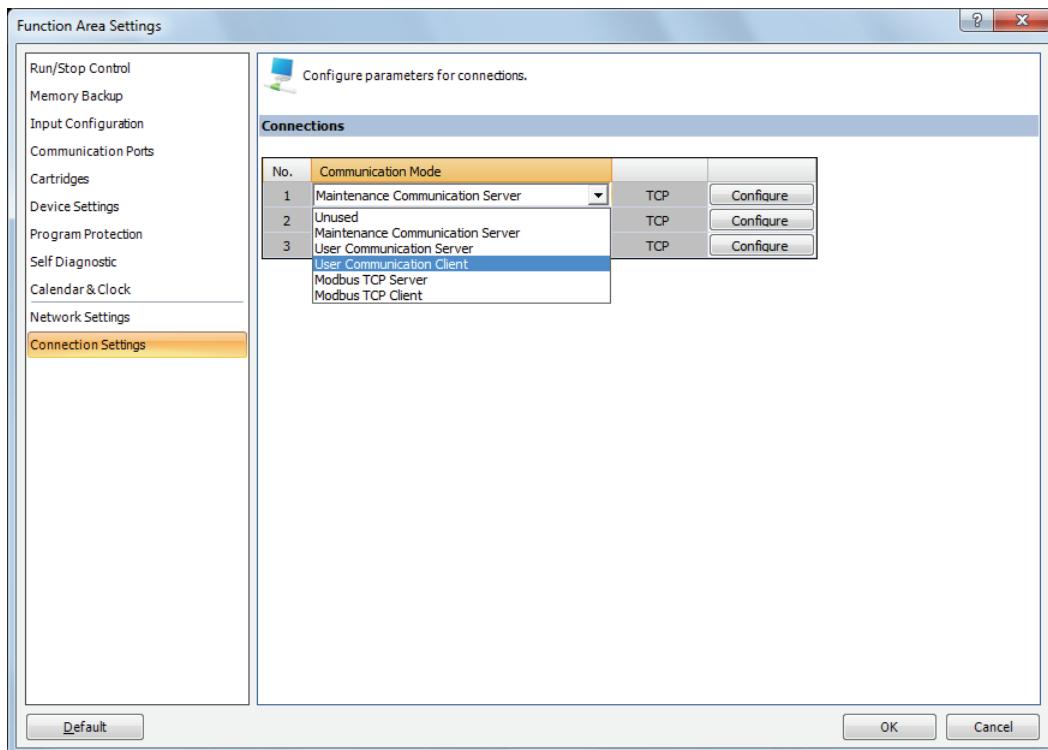
The allocation of the user communication receive instruction cancel flags for each client connection is shown in the table below. For details about the user communication receive instruction cancel flag, see Chapter 25 "User Communication Instructions" in the SmartAXIS Ladder Programming Manual.

Device Address	Description
<b>M8100</b>	User Communication Receive Instruction Cancel Flag (Connection 1)
<b>M8101</b>	User Communication Receive Instruction Cancel Flag (Connection 2)
<b>M8102</b>	User Communication Receive Instruction Cancel Flag (Connection 3)

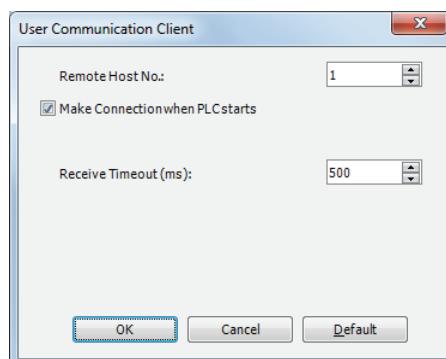
## Programming WindLDR (User Communication Client)

To use the user communication client, configure the user client communication settings in the Function Area Settings dialog box and then download the user program to the SmartAXIS.

1. Select **Configuration** from the WindLDR menu bar, and then click **Connection Settings**.  
The Function Area settings dialog box appears.
2. Select the **User Communication Client** as the communication mode for the client connection 1.



The User Communication Client dialog box appears.

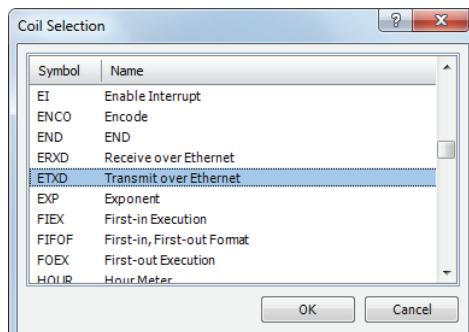


Configure the remote host number and receive timeout. If you want the SmartAXIS to establish the connection when it starts to run, select "Make Connection when PLC starts." Click **OK** button to close the dialog box.

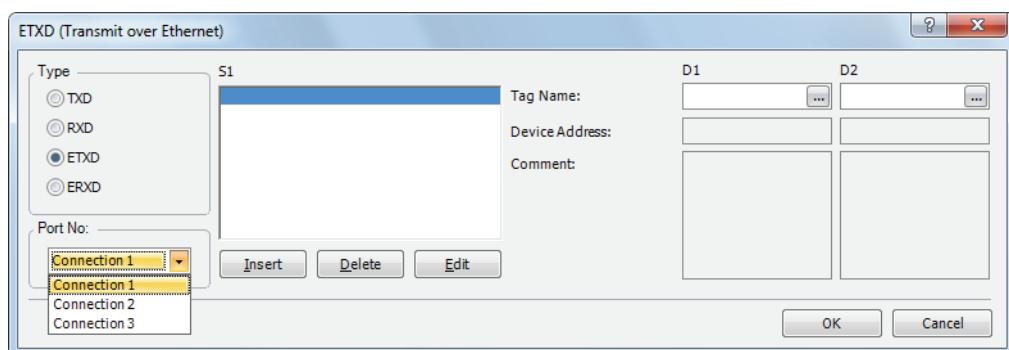
## 10: USER COMMUNICATION INSTRUCTIONS

### 3. Edit the user program.

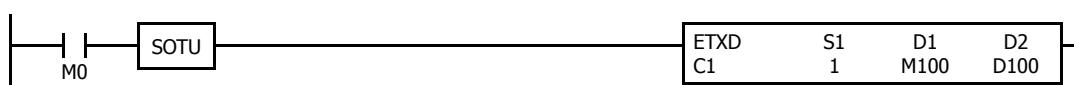
To insert Ethernet user communication instructions to the ladder editor, select the Ethernet user communication instructions (ETXD or ERXD instructions) in the Coil Selection dialog box.



The Ethernet User Communication Instruction dialog box appears.



Select ETXD (Transmit over Ethernet) to transmit data or ERXD (Receive over Ethernet) to receive data as the instruction type. Select the client connection from 1 through 3 and designate S1, D1, and D2. Click **OK** button to close the dialog box.



### 4. Download the user program.

The settings for the user communication client have been finished.

The specifications of Ethernet user communication instructions (ETXD and ERXD instructions) are identical to TXD and RXD instructions. For details about TXD and RXD instructions, see Chapter 25 "User Communication Instructions" in the SmartAXIS Ladder Programming Manual.

## User Communication Server

When a server connection is configured as the user communication server, a client device can access and communicate with the SmartAXIS. The SmartAXIS communicates with the client device according to the settings configured in the ETXD and ERXD instructions that are programmed for the server connection. The local host number and other communication settings can be configured in the **Connection Settings** tab in the Function Area Settings dialog box.

### Specifications (User Communication Server)

Item	User Communication Server
<b>Local host port number</b>	2101 to 2103 (Can be changed between 0 and 65535)
<b>Number of clients that can simultaneously communicate with the SmartAXIS</b>	One client per a user communication server
<b>Receive timeout</b>	100 to 25500 ms (100 ms increments)

#### User Communication Server Receive Instruction (ERXD) Cancel Flag

The allocation of the user communication receive instruction cancel flags for each server connection is shown in the table below. For details about the user communication receive instruction cancel flag, see Chapter 25 "User Communication Instructions" in the SmartAXIS Ladder Programming Manual.

Device Address	Description
<b>M8100</b>	User Communication Receive Instruction Cancel Flag (Connection 1)
<b>M8101</b>	User Communication Receive Instruction Cancel Flag (Connection 2)
<b>M8102</b>	User Communication Receive Instruction Cancel Flag (Connection 3)

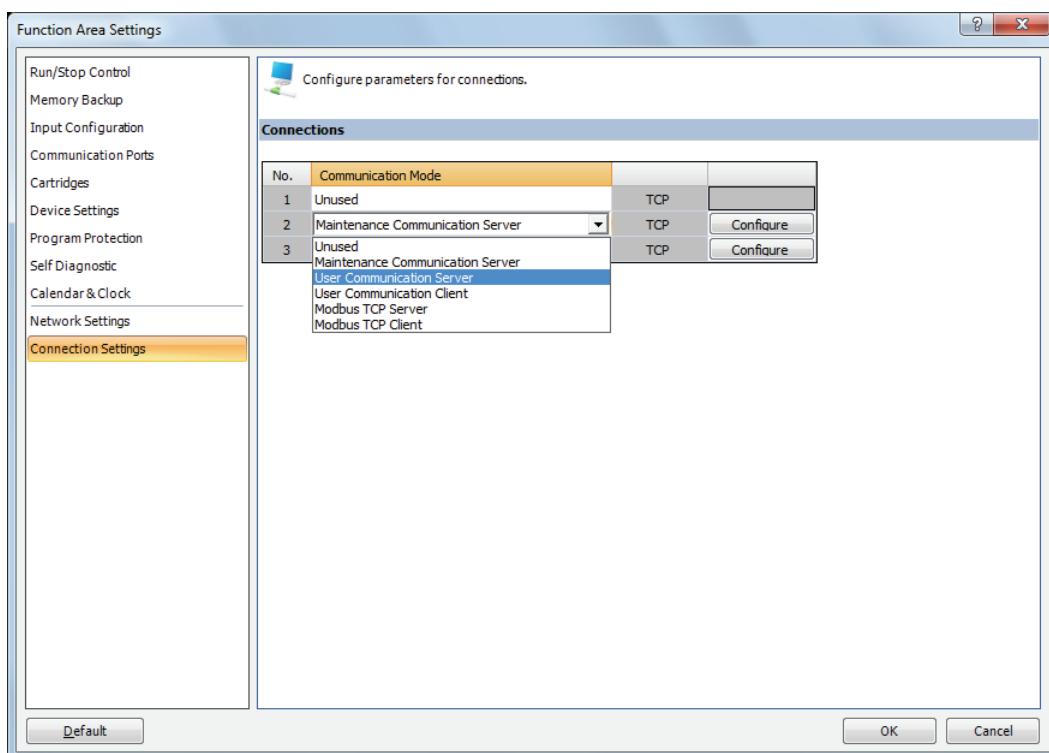
## Programming WindLDR (User Communication Server)

To use the user communication server, configure the user communication server settings in the Function Area Settings dialog box and then download the user program to the SmartAXIS.

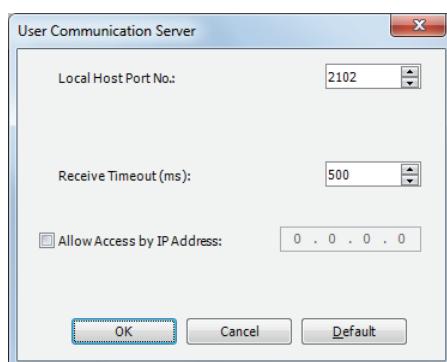
1. Select **Configuration** from the WindLDR menu bar, and then click **Connection Settings**.

The Function Area settings dialog box appears.

2. Select the **User Communication Server** as the communication mode for the server connection 1.



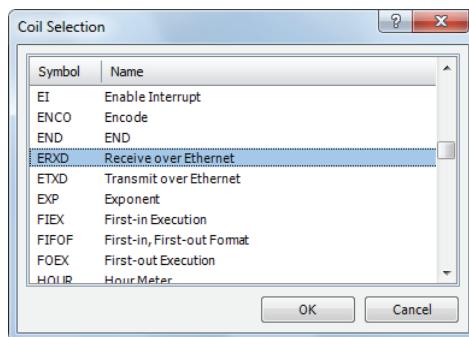
The User Communication Server dialog box appears.



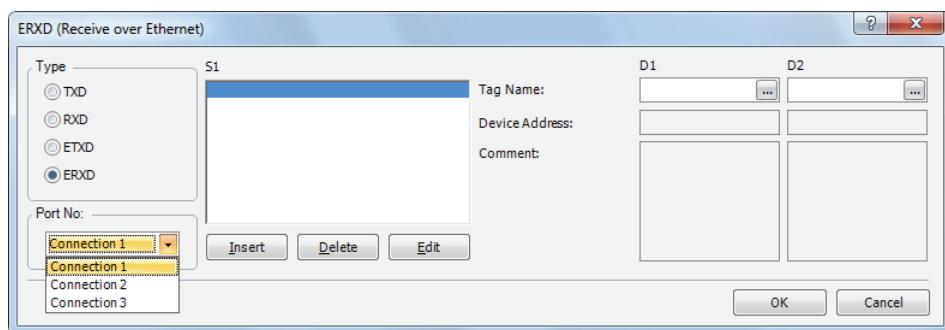
Configure the local host port number and receive timeout. If you want to restrict the access using IP address, configure the allowed IP address.

**3. Edit the user program.**

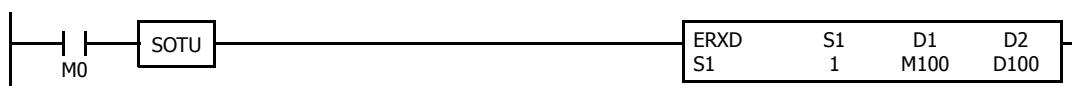
To insert Ethernet user communication instructions to the ladder editor, select the Ethernet user communication instructions (ETXD or ERXD instructions) in the Coil Selection dialog box.



The Ethernet User Communication Instruction dialog box appears.



Select **ETXD** (Transmit over Ethernet) to transmit data and **ERXD** (Receive over Ethernet) to receive data as the instruction type. Select the server connection from 1 through 8 and designate S1, D1, and D2. Click **OK** button to close the dialog box.



**4. Download the user program.**

The settings for the user communication server have been finished.

The specifications of Ethernet user communication instructions (ETXD and ERXD instructions) are identical to TXD and RXD instructions. For details about TXD and RXD instructions, see Chapter 25 "User Communication Instructions" in the SmartAXIS Ladder Programming Manual.

**User Communication Error**

When a user communication error occurs, an error code is stored in the data register designated as a transmit status in the TXD instruction or as a receive status in the RXD instruction. When multiple errors occur, the final error code overwrites all preceding errors and is stored in the status data register.

The status data register also contains transmit/receive status code. To extract a user communication error code from the status data register, divide the value by 16. The remainder is the user communication error code. See Chapter 25 "User Communication Instructions" in the SmartAXIS Ladder Programming Manual.

To correct the error, correct the user program by referring to the error causes described below:

**User Communication Error Code**

User Communication Error Code	Error Cause	Transmit/Receive Completion Output
<b>1</b>	Start inputs to more than 5 TXD instructions are on simultaneously.	Transmit completion outputs of the first 5 TXD instructions from the top of the ladder diagram are turned on.
<b>2</b>	Transmission destination busy timeout	The transmit completion output goes on.
<b>3</b>	Start inputs to more than 5 RXD instructions with a start delimiter are on simultaneously.	Among the first 5 RXD instructions from the top of the ladder diagram, receive completion outputs of RXD instructions go on if the start delimiter matches the first byte of the received data.
<b>4</b>	While a RXD instruction without a start delimiter is executed, another RXD instruction with or without a start delimiter is executed.	The receive completion output of the RXD instruction at a smaller address goes on.
<b>5</b>	While a RXD instruction with a start delimiter is executed, another RXD instruction with the same start delimiter is executed.	No effect on the receive completion output.
<b>7</b>	The first bytes of received data do not match the specified start delimiter.	No effect on the receive completion output. If incoming data with a matching start delimiter is received subsequently, the receive completion output goes on.
<b>8</b>	When ASCII to binary or ASCII to BCD conversion is specified in the receive format, any code other than 0 to 9 and A to F is received. (These codes are regarded as 0 during conversion.)	The receive completion output goes on.
<b>9</b>	BCC calculated from the RXD instruction does not match the BCC appended to the received data.	The receive completion output goes on.
<b>10</b>	Constants including the end delimiter code specified in the RXD instruction do not match the received constants.	The receive completion output goes on.
<b>11</b>	Receive timeout between characters (After receiving one byte of data, the next byte is not received in the period specified for the receive timeout value.)	The receive completion output goes on.
<b>12</b>	Overrun error (Before the receive processing is completed, the next data is received.)	The receive completion output goes off.
<b>13</b>	Framing error (Detection error of start bit or stop bit)	No effect on the completion output.
<b>14</b>	Parity check error (Error is found in the parity check.)	No effect on the completion output.
<b>15</b>	A user communication instruction was used even though the port settings or the connection settings were not set to user communication mode.	No effect on the completion output.

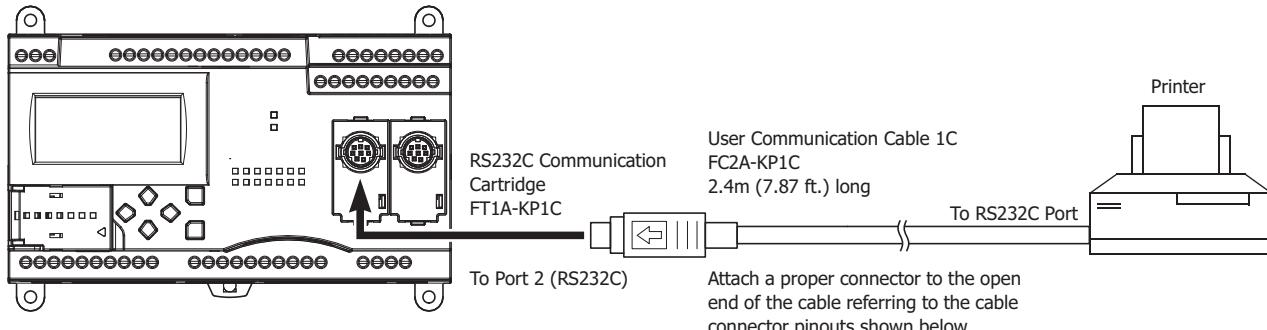
**ASCII Character Code Table**

Upper Bit Lower Bit	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>
<b>0</b>	N <sub>UL</sub>	D <sub>LE</sub>	SP	0	@	P	`	p								
Decimal	0	16	32	48	64	80	96	112	128	144	160	176	192	208	224	240
<b>1</b>	S <sub>OH</sub>	D <sub>C1</sub>	!	1	A	Q	a	q								
Decimal	1	17	33	49	65	81	97	113	129	145	161	177	193	209	225	241
<b>2</b>	S <sub>TX</sub>	D <sub>C2</sub>	"	2	B	R	b	r								
Decimal	2	18	34	50	66	82	98	114	130	146	162	178	194	210	226	242
<b>3</b>	E <sub>TX</sub>	D <sub>C3</sub>	#	3	C	S	c	s								
Decimal	3	19	35	51	67	83	99	115	131	147	163	179	195	211	227	243
<b>4</b>	E <sub>OT</sub>	D <sub>C4</sub>	\$	4	D	T	d	t								
Decimal	4	20	36	52	68	84	100	116	132	148	164	180	196	212	228	244
<b>5</b>	E <sub>NQ</sub>	N <sub>AK</sub>	%	5	E	U	e	u								
Decimal	5	21	37	53	69	85	101	117	133	149	165	181	197	213	229	245
<b>6</b>	A <sub>CK</sub>	S <sub>YN</sub>	&	6	F	V	f	v								
Decimal	6	22	38	54	70	86	102	118	134	150	166	182	198	214	230	246
<b>7</b>	B <sub>EL</sub>	E <sub>TB</sub>	'	7	G	W	g	w								
Decimal	7	23	39	55	71	87	103	119	135	151	167	183	199	215	231	247
<b>8</b>	BS	C <sub>AN</sub>	(	8	H	X	h	x								
Decimal	8	24	40	56	72	88	104	120	136	152	168	184	200	216	232	248
<b>9</b>	HT	EM	)	9	I	Y	i	y								
Decimal	9	25	41	57	73	89	105	121	137	153	169	185	201	217	233	249
<b>A</b>	LF	S <sub>UB</sub>	*	:	J	Z	j	z								
Decimal	10	26	42	58	74	90	106	122	138	154	170	186	202	218	234	250
<b>B</b>	VT	E <sub>SC</sub>	+	;	K	[	k	{								
Decimal	11	27	43	59	75	91	107	123	139	155	171	187	203	219	235	251
<b>C</b>	FF	FS	,	<	L	?										
Decimal	12	28	44	60	76	92	108	124	140	156	172	188	204	220	236	252
<b>D</b>	CR	GS	-	=	M	]	m	}								
Decimal	13	29	45	61	77	93	109	125	141	157	173	189	205	221	237	253
<b>E</b>	SO	RS	.	>	N	^	n	~								
Decimal	14	30	46	62	78	94	110	126	142	158	174	190	206	222	238	254
<b>F</b>	SI	US	/	?	O	_	o									
Decimal	15	31	47	63	79	95	111	127	143	159	175	191	207	223	239	255

## Sample Program – User Communication TXD

This example demonstrates a program to send data to a printer using the user communication TXD2 (transmit) instruction, with the optional RS232C communication cartridge installed on the port 2 connector of the 24-I/O type SmartAXIS.

### System Setup



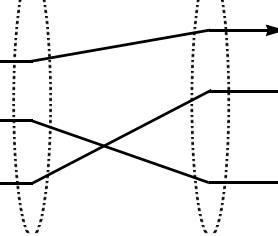
### Cable Connection and Pinouts

#### Mini DIN Connector Pinouts

Description	Color	Pin
Shield	—	<b>Cover</b>
NC No Connection	Black	<b>1</b>
NC No Connection	Yellow	<b>2</b>
TXD Transmit Data	Blue	<b>3</b>
NC No Connection	Green	<b>4</b>
DSR Data Set Ready	Brown	<b>5</b>
NC No Connection	Gray	<b>6</b>
SG Signal Ground	Red	<b>7</b>
NC No Connection	White	<b>8</b>

#### D-sub 9-pin Connector Pinouts

Pin	Description
<b>1</b>	NC No Connection
<b>2</b>	NC No Connection
<b>3</b>	DATA Receive Data
<b>4</b>	NC No Connection
<b>5</b>	GND Ground
<b>6</b>	NC No Connection
<b>7</b>	NC No Connection
<b>8</b>	BUSY Busy Signal
<b>9</b>	NC No Connection



The name of BUSY terminal differs depending on printers, such as DTR. The function of this terminal is to send a signal to remote equipment whether the printer is ready to print data or not. Since the operation of this signal may differ depending on printers, confirm the operation before connecting the cable.



- Do not connect any wiring to the NC (no connection) pins; otherwise, the SmartAXIS and the printer may not work correctly and may be damaged.

### Description of Operation

The data of counter C2 and data register D30 are printed every minute. A printout example is shown on the right.

### Programming Special Data Register

Special data register D8105 is used to monitor the BUSY signal and to control the transmission of print data.

Special DR	Value	Description
<b>D8105</b>	3 (011)	While DSR is on (not busy), the SmartAXIS sends data. While DSR is off (busy), the SmartAXIS stops data transmission. If the off duration exceeds a limit (approx. 5 sec), a transmission busy timeout error will occur, and the remaining data is not sent. The transmit status data register stores an error code. See "User Communication Error" on page 10-22 and Chapter 25 "User Communication Instructions" in the SmartAXIS Ladder Programming Manual..

### Printout Example

--- PRINT TEST ---
11H 00M
CNT2...0050
D030...3854
--- PRINT TEST ---
11H 01M
CNT2...0110
D030...2124

The SmartAXIS monitors the DSR signal to prevent the receive buffer of the printer from overflowing. For the DSR signal, see "DSR Input Control Signal Option D8105" on page 10-12.

## Setting User Communication Mode in WindLDR Function Area Settings

Since this example uses the RS232C port 2, select User Protocol for Port 2 in the Function Area Settings using WindLDR. See "Programming WindLDR" on page 10-4.

## Setting Communication Parameters

Set the communication parameters to match those of the printer. See "Programming WindLDR" on page 10-4. For details of the communication parameters of the printer, see the user's manual for the printer. An example is shown below:

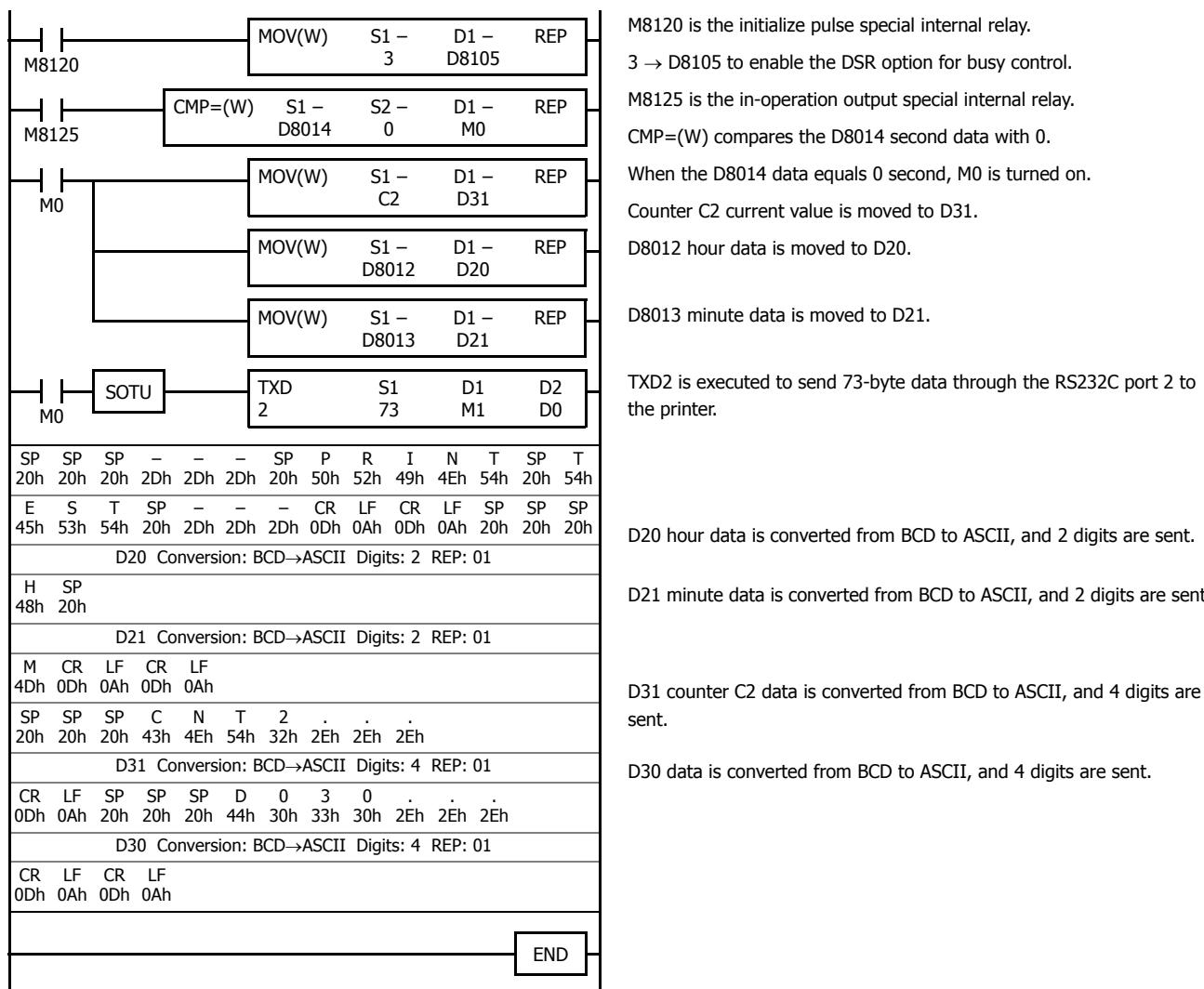
### Communication Parameters:

Baud rate:	9600 bps
Data bits:	8
Parity check:	None
Stop bits:	1

**Note:** The receive timeout value is used for the RXD instruction in the user communication mode. Since this example uses only the TXD instruction, the receive timeout value has no effect.

## Ladder Diagram

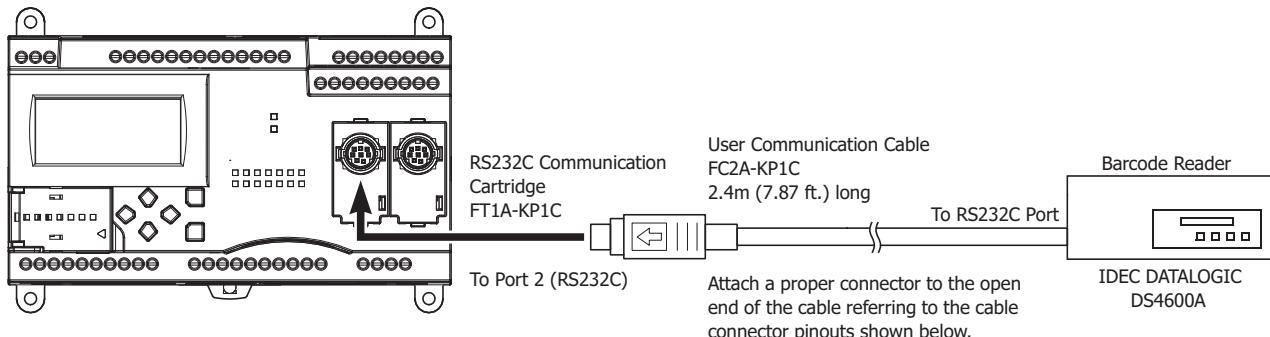
The second data stored in special data register D8014 is compared with 0 using the CMP= (compare equal to) instruction. Each time the condition is met, the TXD2 instruction is executed to send the C2 and D30 data to the printer. A counting circuit for counter C2 is omitted from this sample program.



## Sample Program – User Communication RXD

This example demonstrates a program to receive data from a barcode reader with an RS232C port using the user communication RXD2 (receive) instruction.

### System Setup

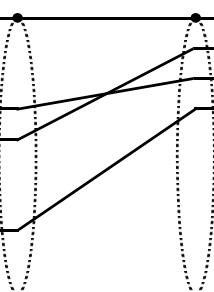


#### Mini DIN Connector Pinouts

Description	Color	Pin
Shield	—	Cover
NC	No Connection	1
NC	No Connection	2
TXD	Transmit Data	3
RXD	Receive Data	4
NC	No Connection	5
NC	No Connection	6
SG	Signal Ground	7
NC	No Connection	8

#### D-sub 25-pin Connector Pinouts

Pin	Description
1	FG Frame Ground
2	TXD1 Transmit Data
3	RXD1 Receive Data
7	GND Ground



- Do not connect any wiring to the NC (no connection) pins; otherwise, the SmartAXIS and the barcode reader may not work correctly and may be damaged.

### Description of Operation

A barcode reader is used to scan barcodes of 8 numerical digits. The scanned data is sent to the SmartAXIS through the RS232C port 2 and stored to data registers. The upper 8 digits of the data are stored to data register D20 and the lower 8 digits are stored to data register D21.

### Setting User Communication Mode in WindLDR Function Area Settings

Since this example uses the RS232C port 2, select User Protocol for Port 2 in the Function Area Settings using WindLDR. See "Programming WindLDR" on page 10-4.

### Setting Communication Parameters

Set the communication parameters to match those of the barcode reader. See "Programming WindLDR" on page 10-4. For details of the communication parameters of the barcode reader, see the user's manual for the barcode reader. An example is shown below:

#### Communication Parameters:

Baud rate:	9600bps
Data bits:	7
Parity check:	Even
Stop bits:	1

## Configuring Barcode Reader

The values shown below are an example of configuring a barcode reader. For actual settings, see the user's manual for the barcode reader.

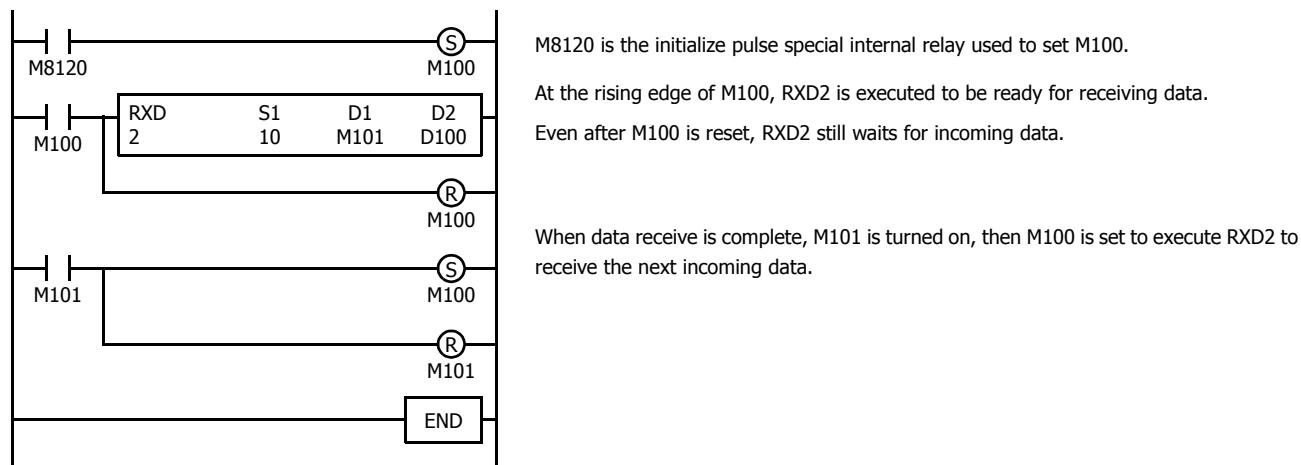
<b>Synchronization mode</b>	Auto		
<b>Read mode</b>	Single read or multiple read		
<b>Communication parameter</b>	Baud rate: Parity check:	9600bps Even	Data bits: Stop bit:
	Header: Data echo back: Output timing: Data output filter: Sub serial:	02h No Output priority 1 No No	7 1 Terminator: BCR data output: Character suppress: Main serial input:
<b>Other communication settings</b>			03h Yes No No
<b>Comparison preset mode</b>	Not used		

## Device Addresses

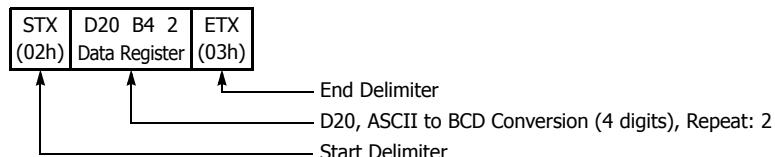
<b>M100</b>	Input to start receiving barcode data
<b>M101</b>	Receive completion output for barcode data
<b>M8120</b>	Initialize pulse special internal relay
<b>D20</b>	Store barcode data (upper 4 digits)
<b>D21</b>	Store barcode data (lower 4 digits)
<b>D100</b>	Receive status data register for barcode data
<b>D101</b>	Receive data byte count data register

## Ladder Diagram

When the SmartAXIS starts operation, the RXD2 instruction is executed to wait for incoming data. When data receive is complete, the data is stored to data registers D20 and D21. The receive completion signal is used to execute the RXD2 instruction to wait for another incoming data.



## RXD2 Data



### **BCC Calculation Examples**

The SmartAXIS can use three new BCC calculation formulas of ADD-2comp, Modbus ASCII, and Modbus RTU for transmit instructions TXD2 and TXD3 and receive instructions RXD2 and RXD3. These block check characters are calculated as described below.

#### **ADD-2comp**

Add the characters in the range from the BCC calculation start position to the byte immediately before the BCC, then invert the result bit by bit, and add 1.

1. Add the characters in the range from the BCC calculation start position to the byte immediately before the BCC.
2. Invert the result bit by bit, and add 1 (2's complement).
3. Store the result to the BCC position according to the designated conversion type (Binary to ASCII conversion or No conversion) and the designated quantity of BCC digits.

**Example:** Binary to ASCII conversion, 2 BCC digits

When the result of step **2** is 175h, the BCC will consist of 37h, 35h.

#### **Modbus ASCII — Calculating the LRC (longitudinal redundancy check)**

Calculate the BCC using LRC (longitudinal redundancy check) for the range from the BCC calculation start position to the byte immediately before the BCC.

1. Convert the ASCII characters in the range from the BCC calculation start position to the byte immediately before the BCC, in units of two characters, to make 1-byte hexadecimal data. (Example: 37h, 35h → 75h)
2. Add up the results of step **1**.
3. Invert the result bit by bit, and add 1 (2's complement).
4. Convert the lowest 1-byte data to ASCII characters. (Example: 75h → 37h, 35h)
5. Store the two digits to the BCC (LRC) position.

If the BCC calculation range consists of an odd number of bytes, the BCC calculation results in an indefinite value. Modbus protocol defines that the BCC calculation range is an even number of bytes.

#### **Modbus RTU — Calculating the CRC-16 (cyclic redundancy checksum)**

Calculate the BCC using CRC-16 (cyclic redundancy checksum) for the range from the BCC calculation start position to the byte immediately before the BCC. The generation polynomial is:  $X^{16} + X^{15} + X^2 + 1$ .

1. Take the exclusive OR (XOR) of FFFFh and the first 1-byte data at the BCC calculation start position.
2. Shift the result by 1 bit to the right. When a carry occurs, take the exclusive OR (XOR) of A001h, then go to step **3**. If not, directly go to step **3**.
3. Repeat step **2**, shifting 8 times.
4. Take the exclusive OR (XOR) of the result and the next 1-byte data.
5. Repeat step **2** through step **4** up to the byte immediately before the BCC.
6. Swap the higher and lower bytes of the result of step **5**, and store the resultant CRC-16 to the BCC (CRC) position. (Example: 1234h → 34h, 12h)

# 11: MODBUS COMMUNICATION

## Introduction

The SmartAXIS supports the Modbus communications protocols.

When a communication cartridge is installed on the expansion communication port, Modbus RTU protocol can be used. The SmartAXIS can communicate with hardware devices supporting Modbus RTU protocol via RS-232C/RS-485.

Modbus TCP protocol can also be used on the built-in Ethernet port. The SmartAXIS can communicate with hardware devices supporting Modbus TCP protocol via Ethernet.

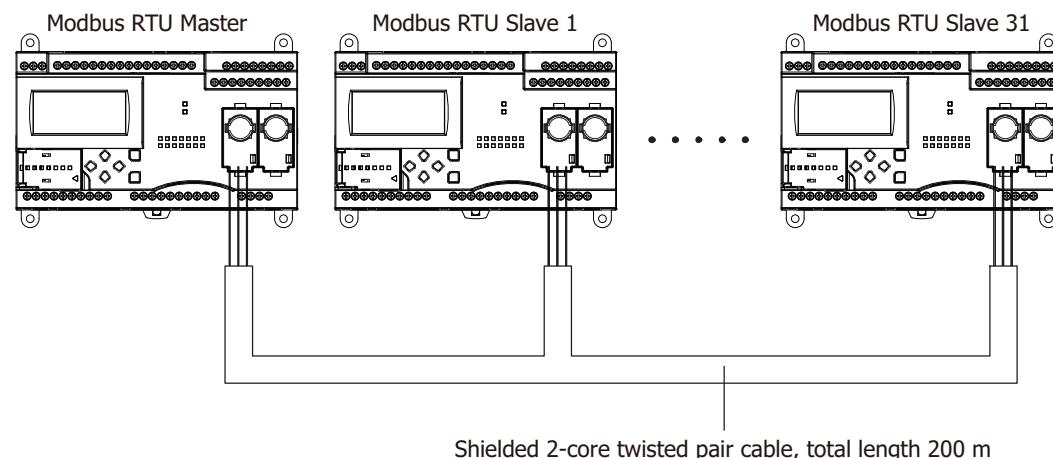
## Modbus Communication via RS-232C/RS-485

### Applicable SmartAXIS

FT1A-12	FT1A-24	FT1A-40	FT1A-48
—	X	X	X

The SmartAXIS supports the Modbus RTU protocol and can be used as the Modbus RTU master and the Modbus RTU slave. When configured as a Modbus RTU master, the SmartAXIS can monitor and change the data of Modbus RTU slave devices. When the SmartAXIS is configured as a Modbus RTU slave, the device data of the SmartAXIS can be monitored and changed from the Modbus RTU master device.

For the Modbus RTU master function and how to configure it, see "Modbus Master Communication" on page 11-2. For the Modbus RTU slave function and how to configure it, see "Modbus Slave Communication Specifications" on page 11-8.



### Notes:

- Modbus communication of the SmartAXIS does not support ASCII mode. Configure all the Modbus devices to communicate with as in RTU mode.
- The 24-I/O type has one expansion communication port. The 40- and 48-I/O types have two expansion communication ports. The 12-I/O type has no expansion communication port.

## Modbus Master Communication

When configured as a Modbus master, the SmartAXIS sends communication requests to Modbus slaves to read/write data. Each communication request is sent to a Modbus slave according to the configured request table.

Modbus master communication settings and request tables for Modbus slave stations can be programmed using the WindLDR Function Area Settings. Communication with slave stations are performed in synchronism with user program execution, and the communication data are processed at the END processing in the order of request numbers specified in the request table. When request execution devices are designated, requests are executed only when the corresponding request execution device is turned on. When request execution devices are not designated, all requests are executed continuously.

### Modbus Master Communication Specifications

Item	Description
Baud Rate (bps)	9600, 19200, 38400, 57600, 115200
Data Bits	8 bits (fixed)
Stop bits	1, 2 bits
Parity	Even, Odd, None
Slave Number	1 to 247 (0: broadcast slave number) <sup>*1</sup>
Maximum Number of Slaves	31
Receive Timeout <sup>*2</sup>	10 to 2550 ms (in increments of 10 ms)
Timeout between Characters	10 ms
Transmission Wait Time	1 to 5000 ms (in increments of 1 ms)
Retry Cycles	1 to 10

\*1: A communication request becomes the broadcast when slave number 0 is specified. The broadcast communication request is received by all Modbus RTU slaves. Modbus RTU slave does not reply to the broadcast communication. Broadcast can be used to write the same data to all Modbus RTU slaves.

\*2: Specifies the period of time before receiving a response frame from a slave.

### Modbus Master Communication Start and Stop

When request execution devices are designated in the Modbus master request table, internal relays or data register bits as many as the request quantity are allocated to execute Modbus master communication. The internal relays or data register bits are allocated in the order of requests. For example, when internal relay M0 is designated as the request execution device, M0 is allocated to request No. 1, M1 to request No. 2, and so on. To execute a request, turn on the corresponding request execution device.

When communication is completed, the request execution device turns off automatically. When it is required to send requests continuously, keep the corresponding request execution device on using a SET or OUT instruction.

When request execution devices are not designated, all requests programmed in the request table are executed continuously.

### Communication Completion and Communication Error

Modbus communication finishes when a read or write process is completed successfully or when a communication error occurs. Communication error occurs when communication failure has repeated more than the designated retry cycles or when the master station does not receive response within the designated receive timeout period. When a communication error occurs, the request is canceled and the next request is processed. When the error status data register is designated, the communication status of each request can be confirmed.

**Note:** Modbus master processes a maximum of one Modbus request per scan.

## Communication Error Data

When Error Status is configured in the Request Table from the Function Area Settings, the error data of each request can be confirmed.

<b>Use a single DR for all communication requests</b>	<b>Error data of each communication request</b>
Unchecked	Error data, the remote host number (high-order byte) and error code (low-order byte), of each request in the entire request table can be confirmed. Data registers as many as the quantity of requests are reserved for storing error data. When an error occurs for a request, error data is stored to the corresponding data register.
Checked	A single data register is shared by all requests. When an error occurs for a request, error data is stored to the data register and the old error data is overwritten.
<b>Bit Allocation</b>	
Remote Host Number (high-order byte)	1 to 255
Error Code (low-order byte)	00h: Normal completion 01h: Function code error (unsupported function code) 02h: Access destination error (address out of range, address+device quantity out of range) 03h: Device quantity error, 1-bit write data error (specified device quantity of 1-bit write is unsupported) 12h: Frame length error (frame length of transmitted request exceeds range) 13h: BCC error (BCC does not match) 14h: Slave number error (received slave number is invalid) 16h: Timeout error (timeout occurs)

## Communication Error Data of Each Request

Error data of each request in the entire request table can be confirmed. To confirm error data of each request, select to use Error Status in the Request Table from the Function Area Settings and enter the data register number.

When Use a single DR for all communication requests is not selected, starting with the data register number, data registers as many as the quantity of requests are reserved for storing error data. When an error occurs for a request, an error code is stored to a corresponding data register.

When Use a single DR for all communication requests is selected, the same data register is shared by all requests. When an error occurs for a request, an error code is stored to the data register and the old value is overwritten.

## Number of Requests in Modbus Master

The number of requests that can be programmed in a request table is shown in the table below:

<b>Port</b>	Port 2 and Port 3
<b>No. of requests</b>	255

**Note:** 8 bytes of the user program area are needed per each request.

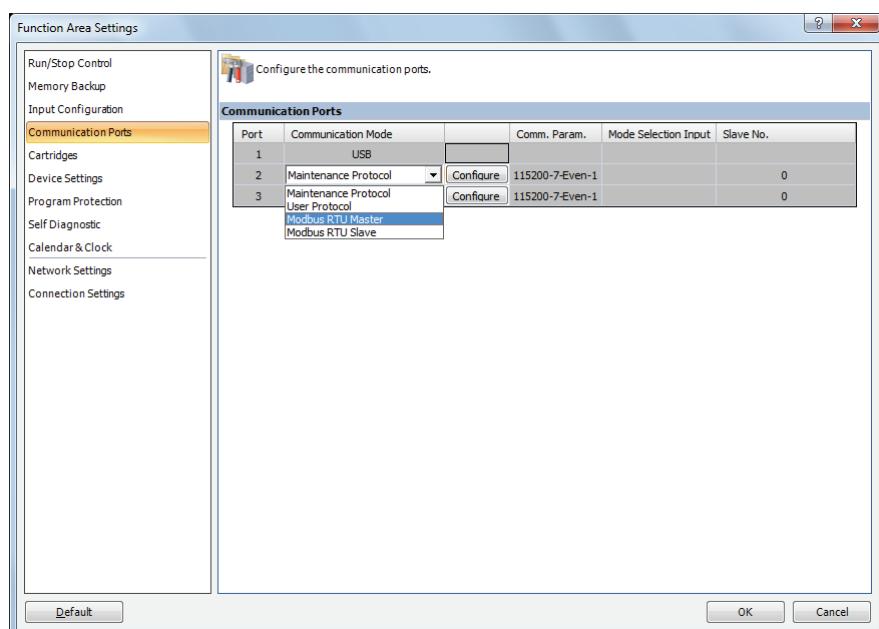
**Programming Modbus Master Using WindLDR**

Modbus master communication is programmed for Modbus RTU mode using WindLDR. Since these settings relate to the user program, the user program must be downloaded to the SmartAXIS after changing any of these settings.

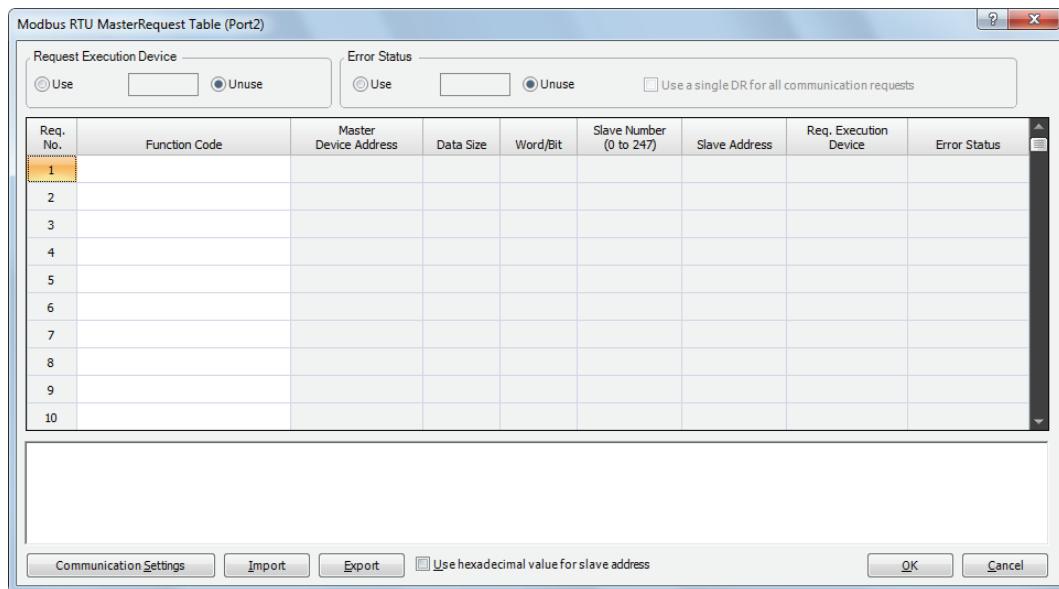
1. From the WindLDR menu bar, select **Configuration > Comm. Ports**.

The Function Area Settings dialog box for Communication Ports appears.

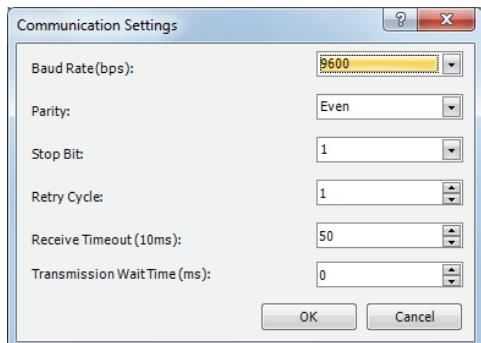
2. In the Communication Mode pull-down list for Port 2, select **Modbus RTU Master**.



3. Click the **Configure** button for Port 2. The Modbus RTU Master Request Table appears.



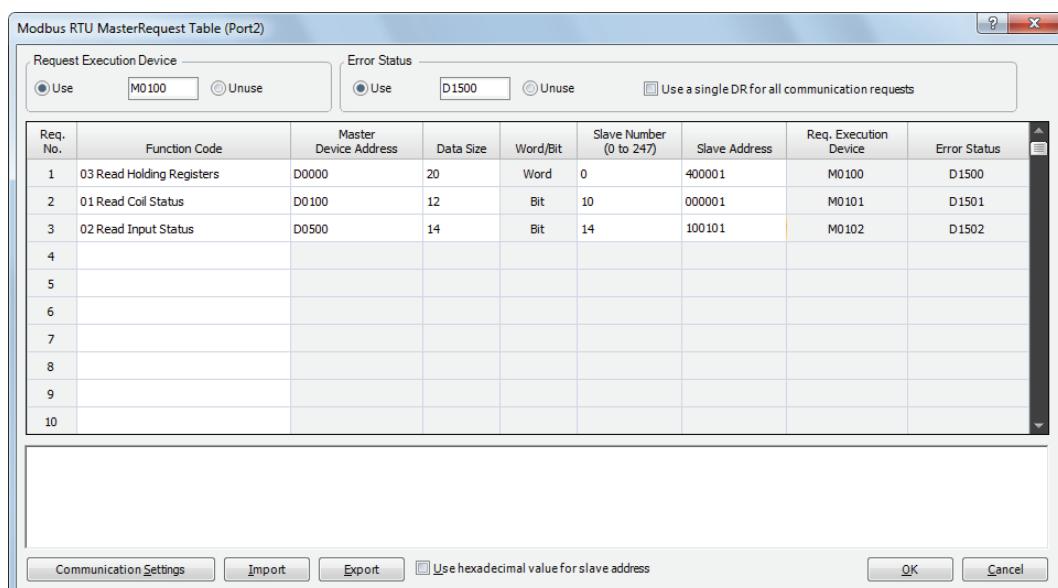
4. Click the **Communication Settings** button. The Communication Settings dialog box appears. Change settings, if required.



<b>Baud Rate (bps)</b>	9600, 19200, 38400, 57600, 115200
<b>Parity</b>	Even, Odd, None
<b>Stop Bits</b>	1 or 2
<b>Retry Cycle</b>	1 to 10
<b>Receive Timeout</b>	1 to 255 ( $\times 10$ ms)
<b>Transmission Wait Time</b>	0 to 5000 (ms)

5. Click the **OK** button to return to the Modbus RTU Master Request Table. Designate requests under the Function Code. A maximum of 255 requests can be entered in one request table.

Choose to use Request Execution Devices and Error Status data registers if necessary. When using Request Execution Devices and Error Status data registers, enter the first number of the devices.



#### Notes for Editing the Request Table

Request execution devices and error status data registers are allocated in the order of request numbers. When deleting a request or changing the order of requests, the relationship of the request to the request execution devices and error status data register is changed. If the internal relay or data register is used in the user program, the device addresses must be changed accordingly. After completing the changes, download the user program again.

6. When editing of the Master Request Table is complete, click the **OK** button to save changes.

7. Download the user program to the SmartAXIS.

Now, programming for the Modbus master is complete. Details about parameters and valid values are as follows.

**Function Code**

The Modbus RTU of the SmartAXIS supports eight function codes as listed in the table below. Supported function codes and valid slave addresses vary with each Modbus slave device to communicate with. Configure the function codes according to the specifications of the Modbus slave devices.

Function Code	Data Size	Slave Address	SmartAXIS as Modbus Slave
01 Read Coil Status	1 to 128 bits	000001 - 065535	Reads bit device statuses of Q (output), R (shift register), or M (internal relay).
02 Read Input Status	1 to 128 bits	100001 - 165535	Reads bit device statuses of I (input), T (timer contact), or C (counter contact).
03 Read Holding Registers	1 to 64 words	400001 - 465535	Reads word device data of D (data register), T (timer preset value), or C (counter preset value).
04 Read Input Registers	1 to 64 words	300001 - 365535	Reads word device data of T (timer current value) or C (counter current value).
05 Force Single Coil	1 bit	000001 - 065535	Changes a bit device status of Q (output), R (shift register), or M (internal relay).
06 Preset Single Register	1 word	400001 - 465535	Changes word device data of D (data register).
15 Force Multiple Coils	1 to 128 bits	000001 - 065535	Changes multiple bit device statuses of Q (output), R (shift register), or M (internal relay).
16 Preset Multiple Registers	1 to 64 words	400001 - 465535	Changes multiple word device data of D (data register).

**Master Device Address**

When function code 01, 02, 03, or 04 is selected to read data from Modbus slaves, designate the first data register or internal relay number to store the data received from the Modbus slave. When function code 05, 06, 15, or 16 is selected to write data to Modbus slaves, designate the first data register or internal relay number to store the data to write to the Modbus slave. Data registers and internal relays can be designated as the master device address.

**Data Size and Word/Bit**

Designate the quantity of data to read or write. The valid data size depends on the function code. When function code 01, 02, 05, or 15 is selected, designate the data size in bits. When function code 03, 04, 06, or 16 is selected, designate the data size in words. For valid data sizes, see the table above.

**Slave No.**

Designate slave numbers 0 through 247. The same slave number can be designated repeatedly for different request numbers which can be 1 through 255. In the Modbus communication, slave number 0 is used for a broadcast slave number. The broadcast can be used to write the same data to all Modbus slaves.

**Slave Address**

Designate data memory addresses of Modbus slaves. The valid slave address range depends on the function code. For valid slave addresses, see the table above. The allocations of memory addresses vary with each Modbus slave device. Refer to manuals for each Modbus slave device.

**Request Execution Device**

To use request execution devices, click the radio button for "Use" and designate the first internal relay in the Modbus RTU Master Request Table. Devices used for executing requests are automatically listed in the table. To execute a request, turn on the corresponding request execution device.

Data registers can also be designated as the Request Execution Device. When the first data register is designated as the Request Execution Device, data register bits as many as the number of requests are allocated from the least significant bit of the first data register. Data register bits assigned as the execution relays are automatically listed in the Request Table.

When request execution devices are not designated, all requests programmed in the Request Table are executed continuously.

**Error Status Data Register**

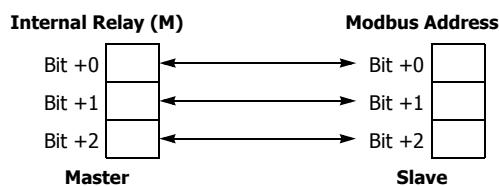
To use error status data registers, click the radio button for "Use" and designate the first data register in the Modbus RTU Master Request Table. Data registers used for storing error statuses are automatically listed in the table. When Use a single DR for all communication requests is selected, the first data register is shared by all requests.

## Processing Requests

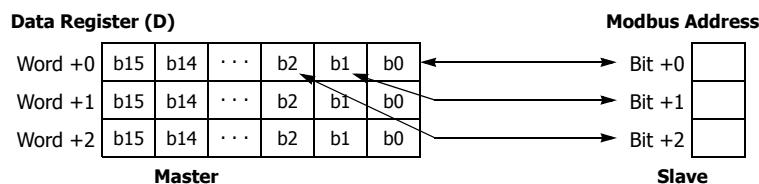
The data for Modbus communication are processed between the master and slaves as shown below.

### Bit Data at Slaves (Function Codes 01, 02, 05, and 15)

- Master Device Address: Internal Relay

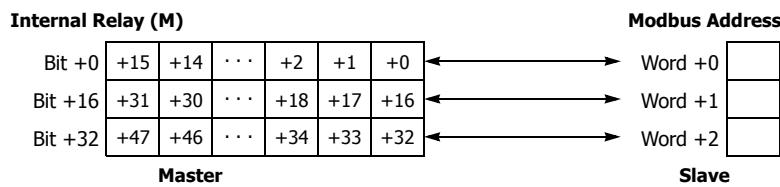


- Master Device Address: Data Register

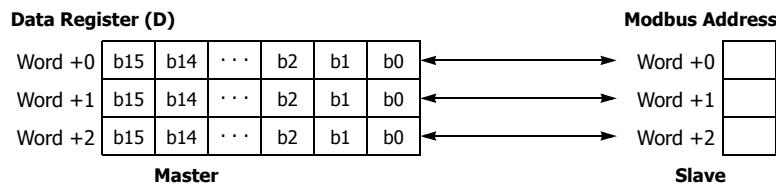


### Word Data at Slaves (Function Codes 03, 04, 06, and 16)

- Master Device Address: Internal Relay



- Master Device Address: Data Register



## Modbus Slave Communication

Modbus slave communication can be configured by selecting Modbus RTU Slave for port 2 and port 3 in the WindLDR Function Area Settings. When a Modbus slave receives a request from the Modbus master, the Modbus slave reads or writes data according to the request. The request is processed at the END processing of the user program.

Modbus slaves do not reply to the Modbus master for the broadcast requests.

## Modbus Slave Communication Specifications

Item	Description	
Baud Rate (bps)	9600, 19200, 38400, 57600, 115200	
Data Bits	8 bits (fixed)	
Stop bits	1, 2 bits	
Parity	Odd, even, none	
Slave Number	Constant	1 to 247
	Data register	The value in D8027 (port 2) or D8028 (port 3) is used
Timeout between Characters <sup>*1</sup>	1.5 characters minimum <sup>*2</sup>	
Timeout between Frames <sup>*1</sup>	3.5 characters minimum <sup>*3</sup>	

\*1: When timeout occurs, the SmartAXIS discards the received data and waits for the first frame of the next valid communication.

\*2: For communication at 19200bps or higher, the timeout between characters needs to be a minimum of 0.75ms.

\*3: For communication at 19200bps or higher, the timeout between frames needs to be a minimum of 1.75ms.

## Address Map

Modbus Device Name	Modbus Address Map (Decimal) <sup>*1</sup>	Communication Frame Address <sup>*2</sup>	SmartAXIS Device <sup>*3</sup>	Applicable Function Code
Coil (000000 and above)	000001 - 000112	0000 - 006F	Q0 - Q141	1, 5, 15
	000701 - 000828	02BC - 033B	R0 - R127	
	001001 - 002024	03E8 - 07E7	M0 - M1277	
	009001 - 009144	2328 - 23B7	M8000 - M8177	
Input Relay (100000 and above)	100001 - 100126	0000 - 007D	I0 - I155	2
	101001 - 101200	03E8 - 04AF	T0 - T199 (timer contact)	
	101501 - 101700	05DC - 06A3	C0 - C199 (counter contact)	
Input Register (300000 and above)	300001 - 300200	0000 - 00C7	T0 - T199 (timer current value)	4
	300501 - 300700	01F4 - 02BB	C0 - C199 (counter current value)	
Holding Register (400000 and above)	400001 - 400200	0000 - 07CF	D0 - D1999	3, 6, 16
	408001 - 408200	1F40 - 2007	D8000 - D8199	
	409001 - 409200	2328 - 23EF	T0 - T199 (timer preset value)	3
	409501 - 409700	251C - 25E3	C0 - C199 (counter preset value)	

\*1: Addresses generally used for Modbus communication. Calculation method of Modbus addresses for SmartAXIS devices are described below.

\*2: These 4-digit addresses are used in the communication frame. To calculate the address used in communication frame, extract lower 5 digits of the Modbus address, subtract 1 from the value, and convert the result into hexadecimal.

\*3: Access within the device range for the SmartAXIS type used.

## Calculating Modbus Addresses for SmartAXIS Devices

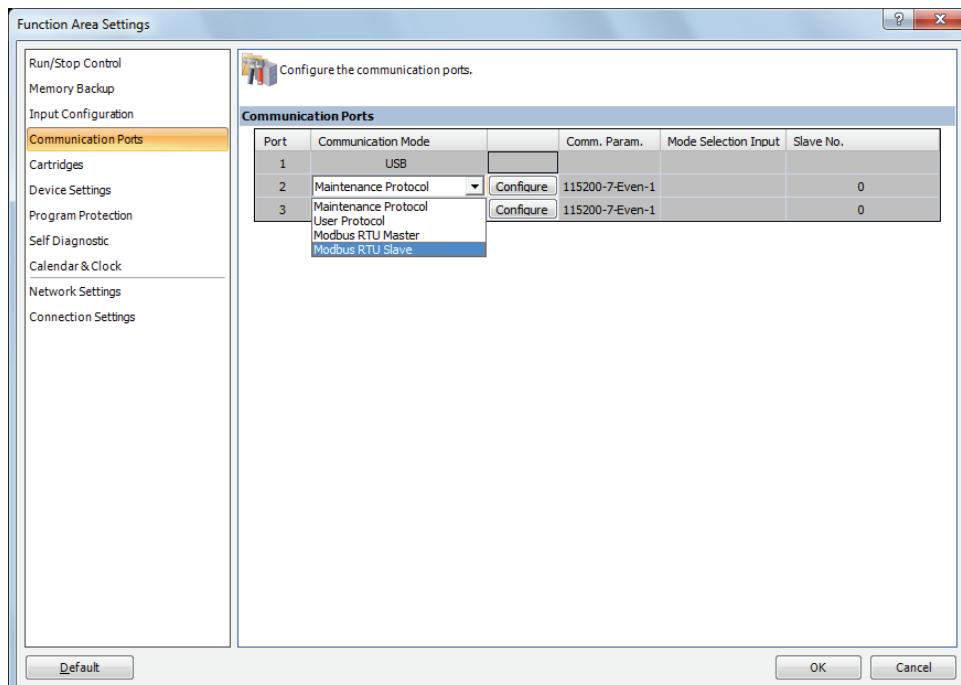
SmartAXIS Device	Calculating Modbus Address	Calculation Example
I, Q, M	M <u>XXX X</u> (2): Octal (1): Decimal ((1) - (4)) × 8 + (2) + (5) ↑ Minimum address      ↑ Offset	Example: M325 $(32 - 0) \times 8 + 5 + 1001 = 1262$ Modbus address: 1262 $1262 - 1 = 1261 = 04ED$ Communication frame address: 04ED
R, T, C, D	D <u>XXXXX</u> (3): Decimal ((3) - (4)) + (5) ↑ Minimum address      ↑ Offset	Example: D756 $(756 - 0) + 400001 = 400757$ Modbus address: 400757 Extract lower 5 digits → 757 $757 - 1 = 756 = 02F4$ Communication frame address: 02F4

Modbus Device Name	SmartAXIS Device	Minimum Address (4)	Offset (5)
Coil	Q0 - Q141	0	1
	R0 - R127	0	701
	M0 - M1277	0	1001
	M8000 - M8177	8000	9001
Input Relay	I0 - I155	0	100001
	T0 - T199 (timer contact)	0	101001
	C0 - C199 (counter contact)	0	101501
Input Register	T0 - T199 (timer current value)	0	300001
	C0 - C199 (counter current value)	0	300501
Holding Register	D0 - D1999	0	400001
	D8000 - D8199	8000	408001
	T0 - T199 (timer preset value)	0	409001
	C0 - C199 (counter preset value)	0	409501

### Programming Modbus Slave Using WindLDR

Modbus slave communication is programmed for Modbus RTU mode using WindLDR. Since these settings relate to the user program, the user program must be downloaded to the SmartAXIS after changing any of these settings.

1. From the WindLDR menu bar, select **Configuration > Comm. Ports**.  
The Function Area Settings dialog box for Communication Ports appears.
2. In the Communication Mode pull-down list for Port 2, select **Modbus RTU Slave**.



3. Click the **Configure** button. The Communication Parameters dialog box appears. Change settings, if required.

Modbus RTU Slave	
Baud Rate(bps):	9600 19200 38400 57600 115200
Data Bits:	8
Parity:	Even
Stop Bits:	1
Slave Number:	<input checked="" type="radio"/> Constant: <input type="radio"/> Data Register: 1
<input type="button" value="OK"/> <input type="button" value="Cancel"/> <input type="button" value="Default"/>	

<b>Baud Rate (bps)</b>	9600 19200 38400 57600 115200
<b>Data Bits</b>	8
<b>Parity</b>	Even, Odd, None
<b>Stop Bits</b>	1 or 2
<b>Slave Number</b>	Constant      1 to 247 Data register      The value in D8027 (port 2) or D8028 (port 3) is used

4. Click the **OK** button to save changes.
5. Download the user program to the SmartAXIS.  
Now, programming for the Modbus slave is complete. Details about parameters and valid values are as follows.

## Communication Protocol

This section describes the communication frame format used for Modbus RTU communication.

### Communication Frame Format

#### Request from Modbus Master

Idle 3.5 characters	Slave No.	Function Code	Data	CRC	Idle 3.5 characters
	1 byte	1 byte		2 bytes	

#### ACK Reply from Modbus Slave

Idle 3.5 characters	Slave No.	Function Code	Data	CRC	Idle 3.5 characters
	1 byte	1 byte		2 bytes	

#### NAK Reply from Modbus Slave

Idle 3.5 characters	Slave No.	Function Code + 80H	Error Code	CRC	Idle 3.5 characters
	1 byte	1 byte	1 byte	2 bytes	

**Note:** Note: Idle means no data flowing on the communication line.

### Communication Frame Format

RTU mode requires a minimum of 3.5-character-long idle time between frames to determine the beginning of a frame. The SmartAXIS Modbus master sends requests at idle intervals of 5ms, which can be changed in the Function Area Settings dialog box.

#### Slave No.

The SmartAXIS can be assigned slave numbers 1 through 247. In the 1:1 communication using RS232C, the same slave number must be set in the master and the SmartAXIS.

Slave No. 0 is reserved for broadcast slave number and is used to write the same data to all Modbus slaves. In this case, the Modbus slaves do not send a reply to the master.

#### Modbus Communication NG Reply Error Code

One of the following error codes is stored in NAK reply.

- 01h: Function code error (unsupported function code)
- 02h: Access destination error (address out of range, address+device quantity out of range)
- 03h: Device quantity error, 1-bit write data error

#### CRC

RTU mode uses CRC check codes.

- **Modbus RTU Mode — Calculating the CRC-16 (cyclic redundancy checksum)**

Calculate the BCC using CRC-16 for the range from the slave number to the byte immediately before the BCC. The generation polynomial is:  $X^{16} + X^{15} + X^2 + 1$ .

1. Take the exclusive OR (XOR) of FFFFh and the first 1-byte data at the slave number.
2. Shift the result by 1 bit to the right. When a carry occurs, take the exclusive OR (XOR) of A001h, then go to step 3. If not, directly go to step 3.
3. Repeat step 2, shifting 8 times.
4. Take the exclusive OR (XOR) of the result and the next 1-byte data.
5. Repeat step 2 through step 4 up to the byte immediately before the BCC.
6. Swap the higher and lower bytes of the result of step 5, and store the resultant CRC-16 to the BCC (CRC) position. (Example: 1234h → 34h, 12h)

**Communication Format**

This section describes the communication format for each function code from the slave number up to immediately before the check code.

**Function Code 01 (Read Coil Status) and Function Code 02 (Read Input Status)**

Function code 01 reads bit device statuses of Q (output), R (shift register), or M (internal relay). One through 128 consecutive bits can be read out.

Function code 02 reads bit device statuses of I (input), T (timer contact), or C (counter contact). One through 128 consecutive bits can be read out.

**Communication Frame****Request from Modbus Master**

Slave No.	Function Code	Address	No. of Bits
xxh	01h / 02h	xxxxh	xxxxh

**ACK Reply from Modbus Slave**

Slave No.	Function Code	Quantity of Data	First 8 Bits	Second 8 Bits		Last 8 Bits
xxh	01h / 02h	xxh	xxh	xxh		xxh

**NAK Reply from Modbus Slave**

Slave No.	Function Code	Error Code
xxh	81h / 82h	xxh

**Communication Example**

Purpose	Read 15 bits starting at output Q10. $Q10 \rightarrow (1 - 0) \times 8 + 0 + 1 = 9$ Modbus address: 9 $9 - 1 = 8 = 8h$ Communication frame address: 0008h
Condition	Slave No. 8 Q10 through Q27 binary data: 1234h

**• RTU Mode**

Request from Modbus Master	08 01 0008 0010 (CRC)
ACK Reply from Modbus Slave	08 01 02 34 12 (CRC)
NAK Reply from Modbus Slave	08 81 xx (CRC)

## Function Code 03 (Read Holding Registers) and Function Code 04 (Read Input Registers)

Function code 03 reads word device data of D (data register), T (timer preset value), or C (counter preset value). One through 64 consecutive words can be read out.

Function code 04 reads word device data of T (timer current value) or C (counter current value). One through 64 consecutive words can be read out.

### Communication Frame

#### Request from Modbus Master

Slave No.	Function Code	Address	No. of Words
xxh	03h / 04h	xxxxh	xxxxh

#### ACK Reply from Modbus Slave

Slave No.	Function Code	Quantity of Data	First High Byte	First Low Byte		Last Low Byte
xxh	03h / 04h	xxh	xxh	xxh		xxh

#### NAK Reply from Modbus Slave

Slave No.	Function Code	Error Code
xxh	83h / 84h	xxh

### Communication Example

Purpose	Read 2 words starting at data register D1710.  D1710 → (1710 – 0) + 400001 = 401711 Modbus address: 401711  Extract lower 5 digits → 1711 1711 – 1 = 1710 = 6AEh Communication frame address: 06AEh
Condition	Slave No. 8 D1710 data: 1234h D1711 data: 5678h

- RTU Mode

Request from Modbus Master	08 03 06AE 0002 (CRC)
ACK Reply from Modbus Slave	08 03 04 12 34 56 78 (CRC)
NAK Reply from Modbus Slave	08 83 xx (CRC)

**Function Code 05 (Force Single Coil)**

Function code 05 changes a bit device status of Q (output), R (shift register), or M (internal relay).

**Communication Frame****Request from Modbus Master**

Slave No.	Function Code	Address	OFF: 0000h ON: FF00h
xxh	05h	xxxxh	xxxxh

**ACK Reply from Modbus Slave**

Slave No.	Function Code	Address	OFF: 0000h ON: FF00h
xxh	05h	xxxxh	xxxxh

**NAK Reply from Modbus Slave**

Slave No.	Function Code	Error Code
xxh	85h	xxh

**Communication Example**

Purpose	Force internal relay M1320 on.  M1320 → (132 – 0) × 8 + 0 + 1001 = 2057 Modbus address: 2057  2057 – 1 = 2056 = 808h Communication frame address: 0808h
Condition	Slave No. 8

**• RTU Mode**

Request from Modbus Master	08 05 0808 FF00 (CRC)
ACK Reply from Modbus Slave	08 05 0808 FF00 (CRC)
NAK Reply from Modbus Slave	08 85 xx (CRC)

## Function Code 06 (Preset Single Register)

Function code 06 changes word device data of D (data register).

### Communication Frame

#### Request from Modbus Master

Slave No.	Function Code	Address	New Data
xxh	06h	xxxxh	xxxxh

#### ACK Reply from Modbus Slave

Slave No.	Function Code	Address	Acknowledge Data
xxh	06h	xxxxh	xxxxh

#### NAK Reply from Modbus Slave

Slave No.	Function Code	Error Code
xxh	86h	xxh

### Communication Example

Purpose	Write 8000 to data register D1708. D1708 → (1708 – 0) + 400001 = 401709 Modbus address: 401709 Extract lower 5 digits → 1709 1709 – 1 = 1708 = 6ACh Communication frame address: 06ACh
Condition	Slave No. 8

- RTU Mode

Request from Modbus Master	08 06 06AC 1F40 (CRC)
ACK Reply from Modbus Slave	08 06 06AC 1F40 (CRC)
NAK Reply from Modbus Slave	08 86 xx (CRC)

### Function Code 15 (Force Multiple Coils)

Function code 15 changes bit device statuses of Q (output), R (shift register), or M (internal relay). One through 128 consecutive bits can be changed.

#### Communication Frame

##### Request from Modbus Master

Slave No.	Function Code	Address	No. of Bits	Quantity of Data	First 8 Bits	Second 8 Bits		Last 8 Bits
xxh	0Fh	xxxxh	xxxxh	xxh	xxh	xxh		xxh

##### ACK Reply from Modbus Slave

Slave No.	Function Code	Address	No. of Bits
xxh	0Fh	xxxxh	xxxxh

##### NAK Reply from Modbus Slave

Slave No.	Function Code	Error Code
xxh	8Fh	xxh

#### Communication Example

<b>Purpose</b>	Write the following bit statuses to internal relays M605 through M624.							
					M605 (ON)	M606 (ON)	M607 (OFF)	
	M610 (ON)	M611 (OFF)	M612 (ON)	M613 (ON)	M614 (OFF)	M615 (OFF)	M616 (ON)	M617 (OFF)
	M620 (OFF)	M621 (OFF)	M622 (OFF)	M623 (OFF)	M624 (OFF)			
	M605 (LSB) through M614 (MSB) binary data: 6B							
	M615 (LSB) through M624 (MSB) binary data: 02							
	M605 → (60 – 0) × 8 + 5 + 1001 = 1486							
	Modbus address: 1486							
	1486 – 1 = 1485 = 5CDh							
	Communication frame address: 05CDh							
<b>Condition</b>	Slave No. 8							

- RTU Mode

<b>Request from Modbus Master</b>	08 0F 05CD 0010 02 6B 02 (CRC)
<b>ACK Reply from Modbus Slave</b>	08 0F 05CD 0010 (CRC)
<b>NAK Reply from Modbus Slave</b>	08 8F xx (CRC)

## Function Code 16 (Preset Multiple Registers)

Function code 16 changes word device data of D (data register). One through 64 consecutive words can be changed.

### Communication Frame

#### Request from Modbus Master

Slave No.	Function Code	Address	No. of Words	Quantity of Data	First High Byte	First Low Byte		Last Low Byte
xxh	10h	xxxxh	xxxxh	xxh	xxh	xxh		xxh

#### ACK Reply from Modbus Slave

Slave No.	Function Code	Address	No. of Words
xxh	10h	xxxxh	xxxxh

#### NAK Reply from Modbus Slave

Slave No.	Function Code	Error Code
xxh	90h	xxh

### Communication Example

<b>Purpose</b>	Write the following data to four data registers D1708 through D1711. D1708            D1709            D1710            D1711 (1234h)        (5678h)        (ABCDh)        (EF01h) D1708 → (1708 – 0) + 400001 = 401709 Modbus address: 401709 Extract lower 5 digits → 1709 1709 – 1 = 1708 = 6ACh Communication frame address: 06ACh
<b>Condition</b>	Slave No. 8

- RTU Mode

<b>Request from Modbus Master</b>	08 10 06AC 0004 08 12 34 56 78 AB CD EF 01 (CRC)
<b>ACK Reply from Modbus Slave</b>	08 10 06AC 0004 (CRC)
<b>NAK Reply from Modbus Slave</b>	08 90 xx (CRC)

**Modbus Communication via Ethernet Communication****Applicable SmartAXIS**

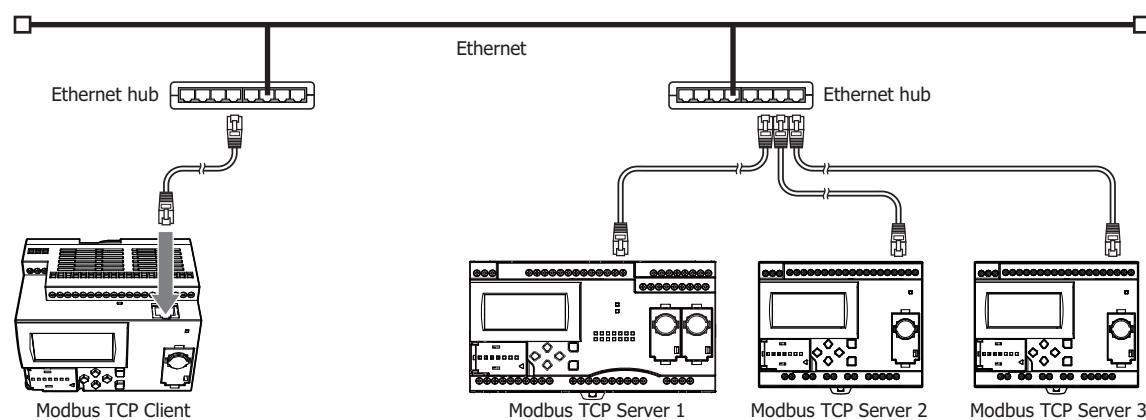
FT1A-12	FT1A-24	FT1A-40	FT1A-48
—	X	X	X

The SmartAXIS supports Modbus TCP clients and Modbus TCP servers. Connect the Ethernet port on the SmartAXIS to enable the SmartAXIS to communicate with Modbus TCP compliant devices.

When configured as a Modbus TCP client, the SmartAXIS can monitor and change the data memory of the network devices supporting Modbus TCP server. A maximum of three connections can be allocated to Modbus TCP clients. Each connection can communicate with one Modbus TCP server device.

When the SmartAXIS is configured as a Modbus TCP server, the device data of the SmartAXIS can be monitored and changed from Modbus TCP client devices. The SmartAXIS can allocate a maximum of three connections for Modbus TCP servers.

For the Modbus TCP client function and how to configure it, see "Modbus TCP Client" on page 11-19. For the Modbus TCP server function and how to configure it, see "Modbus TCP Server" on page 11-24.



## Modbus TCP Client

When configured as a Modbus TCP client, the SmartAXIS sends communication requests to a Modbus TCP server to read/write data. Each communication request is sent to a Modbus Server according to the configured request table.

Modbus TCP client communication settings and request tables for Modbus TCP servers can be configured using the WindLDR Function Area Settings. The SmartAXIS communicate with the Modbus TCP servers according to those settings.

Communication with Modbus TCP servers are performed in sync with the user program execution, and the communication data is processed at the END, in the order of request numbers specified in the request table.

### Modbus TCP Client Specifications

Parameter	Modbus TCP Client
Slave Number	1 to 247
Maximum Number of Servers	3 (one server per one connection)
Receive Timeout *1	100 to 25,500ms (in increments of 100ms)

\*1: Specifies the period of time before receiving a response frame from a server.

### Modbus TCP Client Start and Stop

When request execution devices are designated in the Modbus TCP Client request table, internal relays or data register bits (as many as the request quantity) are allocated to execute Modbus TCP Client communication. The internal relays or data register bits are allocated in the order of requests. For example, when internal relay M0 is designated as the request execution device, M0 is allocated to request No. 1, M1 to request No. 2, and so on. To execute a request, turn on the corresponding request execution device.

When communication is completed, the request execution device turns off automatically. When it is required to send requests continuously, keep the corresponding request execution device on using a SET or OUT instruction.

When request execution devices are not designated, all requests programmed in the request table are executed continuously.

### Communication Completion and Communication Error

Modbus communication finishes when a read or write process is completed successfully or when a communication error occurs.

A communication error occurs when communication failure has repeated three times. When a communication error occurs, the request is canceled and the next request is processed. When the error status data register is designated, the communication status of each request can be confirmed.

### Communication Error Data

When Error Status is configured in the Request Table from the Function Area Settings, the error data of each request can be confirmed.

Use a single DR for all communication requests	Error data of each communication request
Unchecked	Error data, the remote host number (high-order byte) and error code (low-order byte), of each request in the entire request table can be confirmed. Data registers as many as the quantity of requests are reserved for storing error data. When an error occurs for a request, error data is stored to the corresponding data register.
Checked	A single data register is shared by all requests. When an error occurs for a request, error data is stored to the data register and the old error data is overwritten.
<b>Bit Allocation</b>	
Remote Host Number (high-order byte)	1 to 255
Error Code (low-order byte)	00h: Normal completion 01h: Function code error (unsupported function code) 02h: Access destination error (address out of range, address+device quantity out of range) 03h: Device quantity error, 1-bit write data error (specified device quantity of 1-bit write is unsupported) 12h: Frame length error (frame length of transmitted request exceeds range) 13h: BCC error (BCC does not match) 14h: Slave number error (received slave number is invalid) 16h: Timeout error (timeout occurs)

### **Modbus TCP Communication Request Table**

A maximum of 255 requests can be configured in the Modbus TCP Client Request Table.

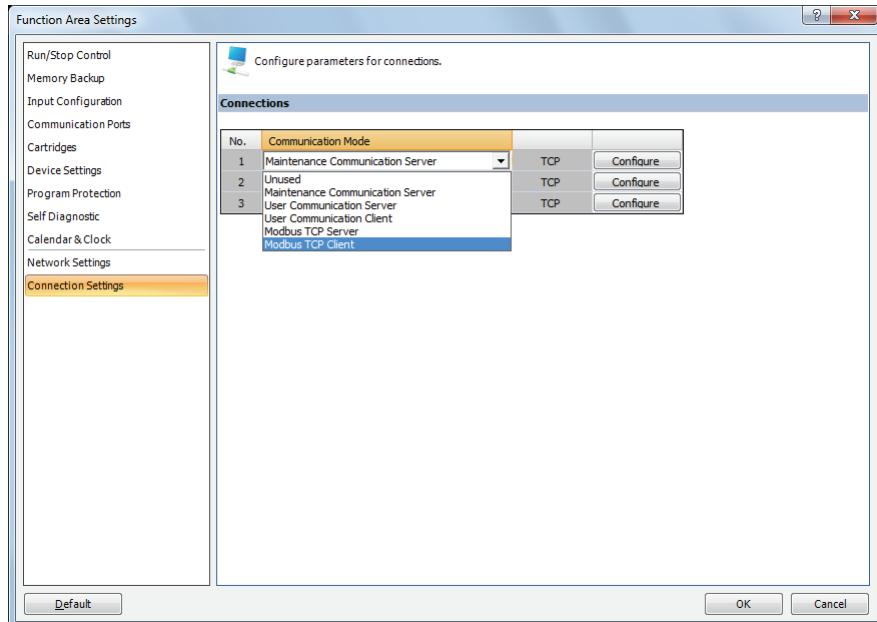
#### **Notes:**

- 10 bytes of the user program area are needed per each request.
- Request execution devices and error status data registers are allocated in the order of request numbers. When deleting a request or changing the order of requests, the relationship of the request to the request execution devices and error status data register is changed. If the allocated internal relays or data registers are used in the user program, those device addresses must be updated accordingly.

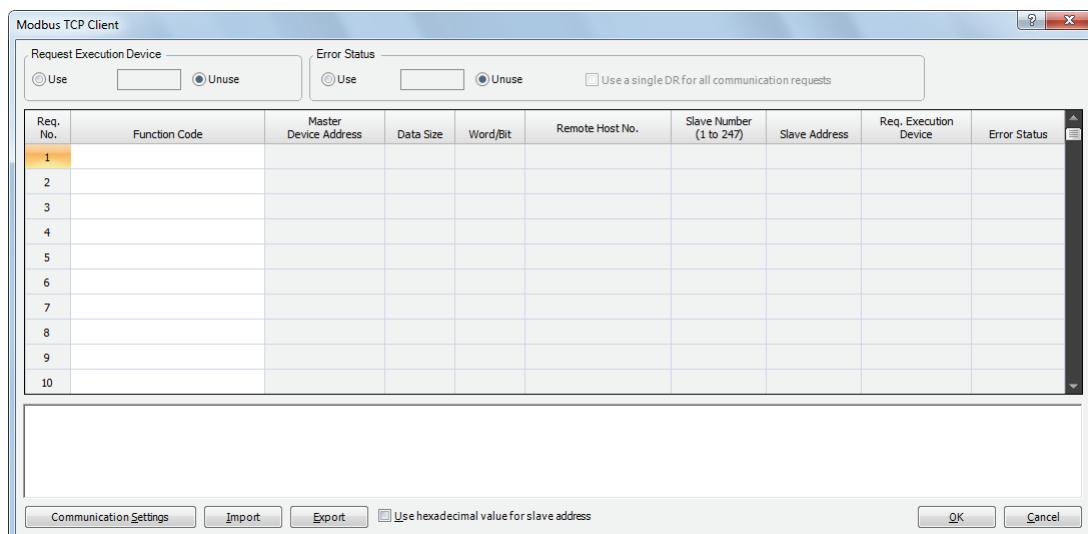
## Programming WindLDR (Modbus TCP Client)

In order to use the Modbus TCP client, configure the Modbus TCP client in the Function Area Settings dialog box and then download the user program to the SmartAXIS.

1. From the WindLDR menu bar, select **Configuration > Connection Settings**.  
The Function Area Settings dialog box appears.
2. Select **Modbus TCP Client** as the communication mode for the connection 1.

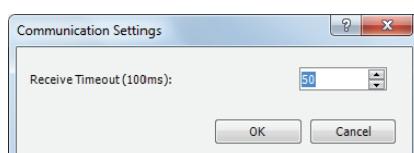


The Modbus TCP Client Request Table dialog box appears.



3. Click on the **Communication Settings** button.

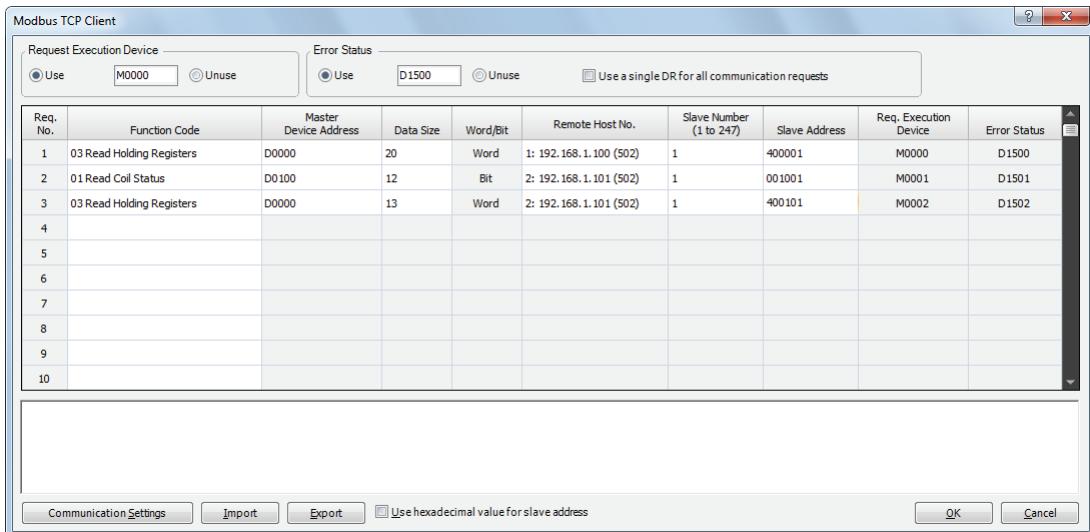
The Communication Settings dialog box appears. Configure the timeout and click **OK** button to close the Communication Settings dialog box.



### 4. Configure the requests of the Modbus TCP Client.

A maximum of 255 requests can be entered in one request table. Specify the Modbus TCP servers with the remote host numbers. After all requests are configured, click **OK** button to close the dialog box.

Choose to use Request Execution Devices and Error Status data registers if necessary. When using Request Execution Devices and Error Status data registers, enter the first number of the devices.



### 5. Download the user program to the SmartAXIS.

Programming for the Modbus TCP client is complete. Details about parameters and valid values are as follows.

#### Function Code

The Modbus TCP of the SmartAXIS supports eight function codes as listed in the table below. Supported function codes and valid slave addresses vary with each Modbus server device to communicate with. Configure the function codes according to the specifications of the Modbus server devices.

Function Code	Data Size	Slave Address	SmartAXIS as Modbus Slave
01 Read Coil Status	1 to 128 bits	000001 - 065535	Reads bit device statuses of Q (output), R (shift register), or M (internal relay).
02 Read Input Status	1 to 128 bits	100001 - 165535	Reads bit device statuses of I (input), T (timer contact), or C (counter contact).
03 Read Holding Registers	1 to 64 words	400001 - 465535	Reads word device data of D (data register), T (timer preset value), or C (counter preset value).
04 Read Input Registers	1 to 64 words	300001 - 365535	Reads word device data of T (timer current value) or C (counter current value).
05 Force Single Coil	1 bit	000001 - 065535	Changes a bit device status of Q (output), R (shift register), or M (internal relay).
06 Preset Single Register	1 word	400001 - 465535	Changes word device data of D (data register).
15 Force Multiple coils	1 to 128 bits	000001 - 065535	Changes multiple bit device statuses of Q (output), R (shift register), or M (internal relay).
16 Preset Multiple Registers	1 to 64 words	400001 - 465535	Changes multiple word device data of D (data register).

**Master Device Address**

When function code 01, 02, 03, or 04 is selected to read data from Modbus servers, designate the first data register or internal relay number to store the data received from the Modbus server. When function code 05, 06, 15, or 16 is selected to write data to Modbus servers, designate the first data register or internal relay number to store the data to write to the Modbus server. Data registers and internal relays can be designated as the master device address.

**Data Size and Word/Bit**

Designate the quantity of data to read or write. The valid data size depends on the function code. When function code 01, 02, 05, or 15 is selected, designate the data size in bits. When function code 03, 04, 06, or 16 is selected, designate the data size in words. For valid data sizes, see "Function Code" on page 11-22.

**Remote Host No.**

Designate the remote host number configured in the Remote Host List dialog box. In the Remote Host List dialog box, IP address and port number are configured for each remote host. The default port number of Modbus TCP communication is 502. If the Modbus TCP server uses a different port number, configure that port number in the Remote Host List dialog box.

**Slave No.**

Designate slave numbers 1 through 247. The same slave number can be designated repeatedly for different request numbers which can be 1 through 255. The slave number is usually not referred by Modbus TCP server. Configure the slave number if Modbus TCP server requires.

**Slave Address**

Designate data memory addresses of Modbus servers. The valid slave address range depends on the function code. For valid slave addresses, see the table above. The allocations of memory addresses vary with each Modbus server device. Refer to manuals for each Modbus server device.

**Request Execution Device**

To use request execution devices, click the radio button for "Use" and designate the first internal relay or data register in the Modbus TCP Client Request Table. Internal relays or data register bits used for executing requests are automatically listed in the table. To execute a request, turn on the corresponding request execution device.

When request execution devices are not designated, all requests programmed in the Request Table are executed continuously.

**Error Status Data Register**

To use error status data registers, click the radio button for "Use" and designate the first data register in the Modbus TCP Client Request Table. Data registers used for storing error statuses are automatically listed in the table. When Use a single DR for all communication requests is selected, the first data register is shared by all requests.

## Modbus TCP Server

When the SmartAXIS is configured as the Modbus TCP server, Modbus TCP client devices can read/write data to the SmartAXIS. When the SmartAXIS receives a valid request from a Modbus TCP client device, the data is read or written according to the request received. The communication data received from Modbus TCP clients is processed at the END processing of the user program.

### Modbus TCP Server Specifications

Parameter	Modbus TCP Server
Slave Number	Ignored
Response Time	1.5 ms
Number of Clients that can Access Simultaneously	3 (1 client per 1 connection)
Port Number	502 (can be changed between 0 and 65535)
Supported Function Code	01 Read Coil Status 02 Read Input Status 03 Read Holding Registers 04 Read Input Registers 05 Force Single Coil 06 Preset Single Register 15 Force Multiple coils 16 Preset Multiple Registers

### Address Map

Modbus TCP client can access the Modbus devices (Coil, Input Relay, Input Register, and Holding Register) of Modbus server to read or write the device data (I, Q, M, R, T, C, and D) of SmartAXIS. Refer to the following table to configure the Modbus TCP clients.

Modbus Device Name	Modbus Address Map (Decimal)	Communication Frame Address <sup>*1</sup>	SmartAXIS Device <sup>*2</sup>	Applicable Function Code
Coil (000000 and above)	000001 - 000112	0000 - 006F	Q0 - Q141	1, 5, 15
	000701 - 000828	02BC - 033B	R0 - R127	
	001001 - 002024	03E8 - 07E7	M0 - M1277	
	009001 - 009144	2328 - 23B7	M8000 - M8177	
Input Relay (100000 and above)	100001 - 100126	0000 - 007D	I0 - I155	2
	101001 - 101200	03E8 - 04AF	T0 - T199 (timer contact)	
	101501 - 101700	05DC - 06A3	C0 - C199 (counter contact)	
Input Register (300000 and above)	300001 - 300200	0000 - 00C7	T0 - T199 (timer current value)	4
	300501 - 300700	01F4 - 02BB	C0 - C199 (counter current value)	
Holding Register (400000 and above)	400001 - 400200	0000 - 07CF	D0 - D1999	3, 6, 16
	408001 - 408200	1F40 - 2007	D8000 - D8199	
	409001 - 409200	2328 - 23EF	T0 - T199 (timer preset value)	3
	409501 - 409700	251C - 25E3	C0 - C199 (counter preset value)	

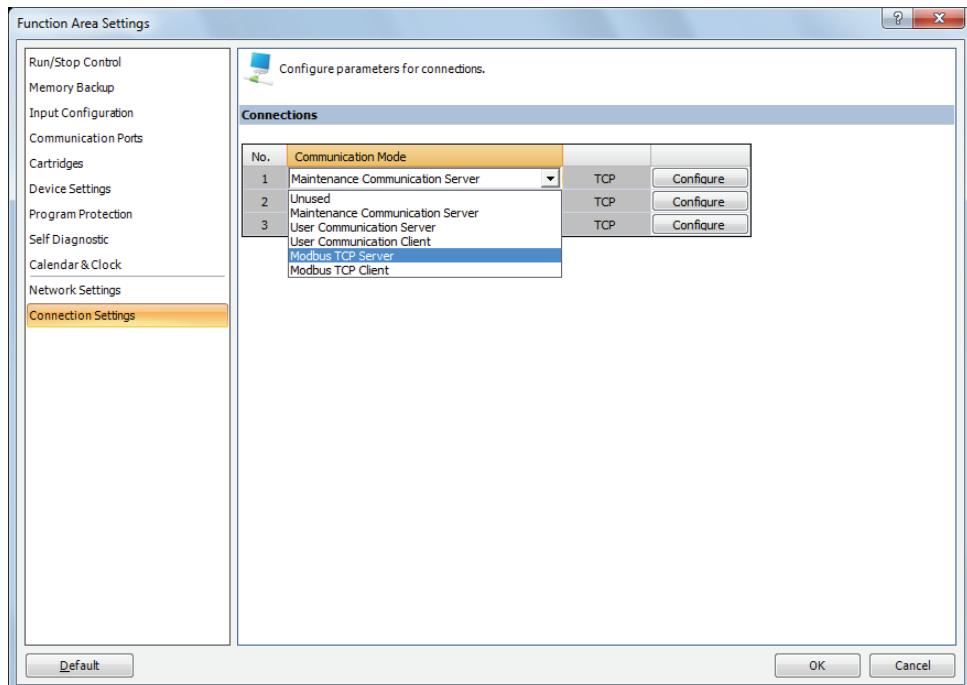
\*1: These 4-digit addresses are used in the communication frame. To calculate the address used in communication frame, extract lower 5 digits of the Modbus address, subtract 1 from the value, and convert the result into hexadecimal. For details, see "Modbus Slave Communication" on page 11-8.

\*2: Access within the device range for the SmartAXIS type used.

## Programming WindLDR (Modbus TCP Server)

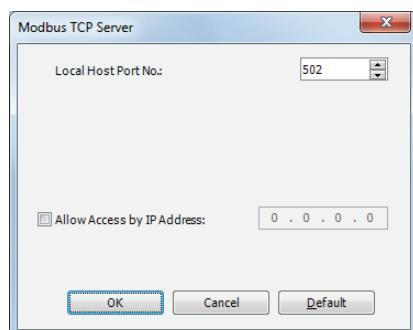
In order to use Modbus TCP server, configure the Modbus TCP server in the Function Area Settings dialog box and then download the user program to the SmartAXIS.

1. From the WindLDR menu bar, select **Configuration > Connection Settings**.  
The Function Area Settings dialog box appears.
2. Select **Modbus TCP Server** as the communication mode for connection 1.



The Modbus TCP Server dialog box appears.

3. Configure the parameters and click on **OK** button.



**Note:** For details about each parameter, see the following page.

4. Download the user program to the SmartAXIS.

Programming for the Modbus TCP server is complete.

## Modbus TCP Server Communication Settings

### Local Host Port No.

Configure the local host port number between 0 and 65,535. The same local host port number can be used with multiple connection numbers.

If the same port number is used with multiple connections, Modbus TCP clients (as many as the number of the connections) can communicate with SmartAXIS through the same port number.

### Allow Access by IP Address

You can set the IP address for a device to permit access. By setting the allowed IP address, only the device with the specified IP address can establish a connection with the SmartAXIS and communicate with the Modbus TCP server. When the same local host port number is configured in multiple connections, all the allowed IP address settings are effective. If a connection in which the allowed IP address is not configured uses the same local host port number, arbitrary access is allowed through the port.

Example 1: If two connections use the same local port number and an allowed IP address is not configured for both connections, access from a total of two clients with any IP addresses is accepted.

Example 2: If two connections use the same local port number and 192.168.1.101 and 192.168.1.102 are configured as the allowed IP addresses, access from a total of two clients whose IP addresses are 192.168.1.101 and 192.168.1.102 is accepted.

Example 3: If connection 1 and 2 use the same local port number, an allowed IP address 192.168.1.101 is configured for connection 1, and the allowed IP address is not configured for connections, access from a total of two clients with any IP addresses is accepted.

## Modbus TCP Communication Format

This section describes the communication format used for Modbus TCP client and server communication. Modbus TCP communication format starts with the Modbus TCP header followed by the RTU mode communication format without the idle 3.5 characters at both ends and CRC as shown below.

### Modbus TCP Communication Format

Transaction ID	Protocol ID	Message Length (bytes)	Unit ID	Function Code	Data
2 bytes	2 bytes	2 bytes	1 byte	1 byte	N bytes

**Modbus TCP Header**

RTU Mode Communication Format	Idle 3.5 characters	Slave No.	Function Code	Data	CRC	Idle 3.5 characters
		1 byte	1 byte	N bytes	2 bytes	

#### Transaction ID

The Modbus TCP server (slave) returns the request ID sent from the client (master) without any change. When receiving the returned request ID, the client can confirm to which request the response was returned. When confirmation is not required, designate 0 as a transaction ID.

#### Protocol ID

Designate 0 to identify Modbus TCP protocol.

#### Message Length

Designate the length of the following message in bytes.

#### Unit ID

The ID for identifying the device. Store the slave number of the Modbus TCP server. The SmartAXIS Modbus TCP server accepts and processes requests when the unit ID of the received request is not 0. When the unit ID is 0, the received request is processed as broadcast communication and no response is returned to the Modbus TCP client.

#### Function Code

Designate a function code, such as 01 (read coil status) and 02 (read input status).

#### Data

Designate required data for each function.

# 12: TROUBLESHOOTING

## Introduction

This chapter describes the procedures to determine the cause of problem and actions to be taken when any problem occurs while operating the SmartAXIS.

The SmartAXIS has self-diagnostic functions to prevent the spread of problems if any should occur. In case of any problem, follow the troubleshooting procedures to determine the cause and to correct the error.

Errors are checked in various stages. While editing a user program on WindLDR, incorrect devices and other data are rejected. User program syntax errors are found during compilation on WindLDR. When an incorrect program is downloaded to the SmartAXIS, user program syntax errors are still checked. Errors are also checked at starting and during operation of the SmartAXIS. When an error occurs, the error is reported by turning on the ERR LED on the SmartAXIS Lite and an error message can be viewed on WindLDR. For the procedure to confirm the error status and clear the error on the SmartAXIS Pro, see "Checking/Clearing Error Information" on page 6-19.

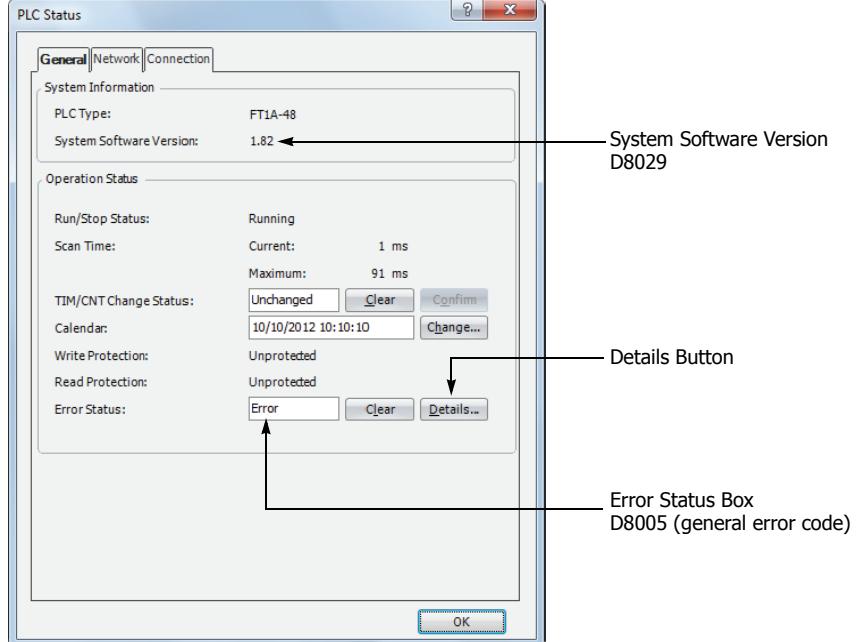
## Reading Error Data

When any error occurs during the SmartAXIS operation, "Error" is indicated and error details can be read using WindLDR.

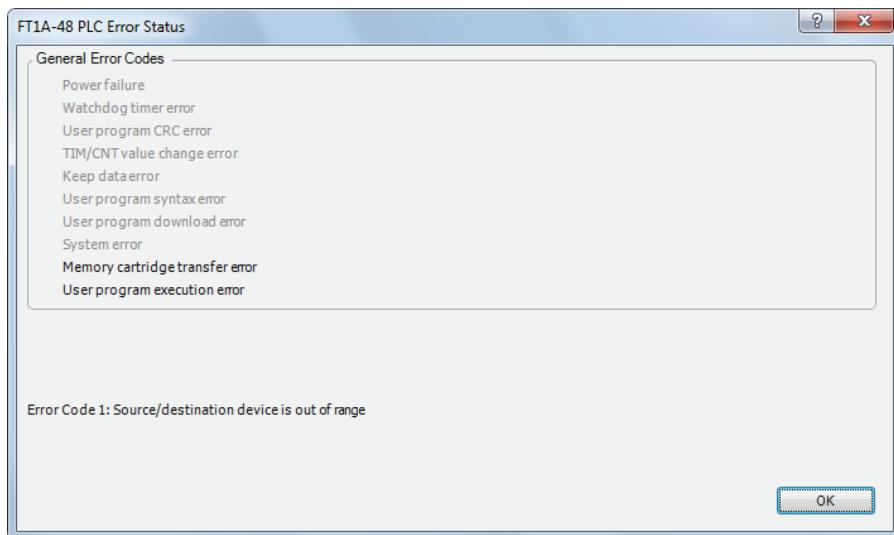
### Monitoring WindLDR

1. From the WindLDR menu bar, select **Online > Monitor > Monitor**. The monitor mode is enabled.
2. From the WindLDR menu bar, select **Online > Status**. The PLC Status dialog box appears.

When any error exists, "Error" is displayed in the error status box.



3. On the right of the Error Status in the PLC Status dialog box, click the **Details** button.  
The PLC Error Status screen appears.

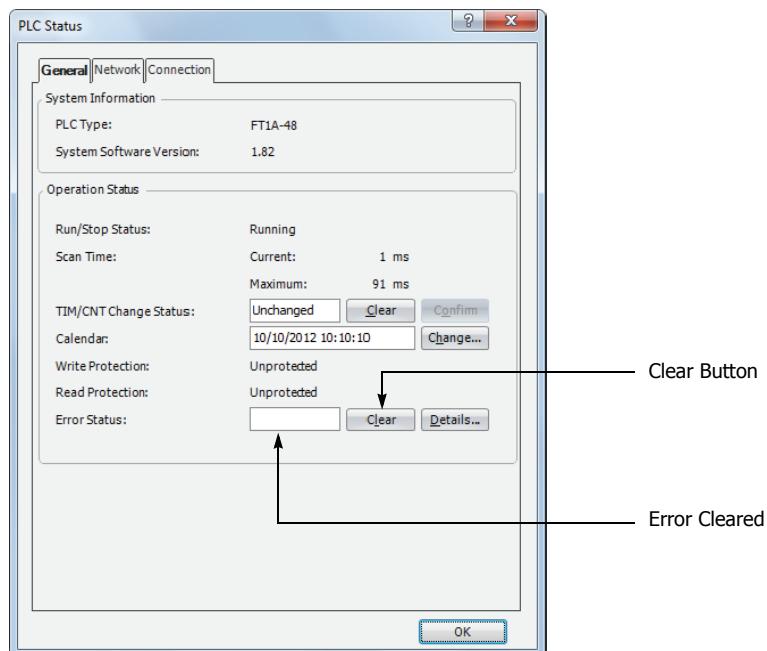


### Clearing Error Codes from WindLDR

After removing the cause of the error, clear the error code using the following procedure:

1. From the WindLDR menu bar, select **Online > Monitor > Monitor**. The monitor mode is enabled.
2. From the WindLDR menu bar, select **Online > Status**. The PLC Status dialog box appears.
3. On the right of the **Error Status** in the PLC Status dialog box, click the **Clear** button.

This procedure clears the error code from special data register D8005 (general error code), and the error is cleared from the PLC Status dialog box.

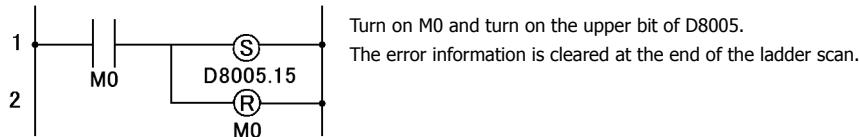


## Special Data Registers for Error Information

Two data registers are assigned to store information on errors.

<b>D8005</b>	General Error Code
<b>D8006</b>	User Program Execution Error Code

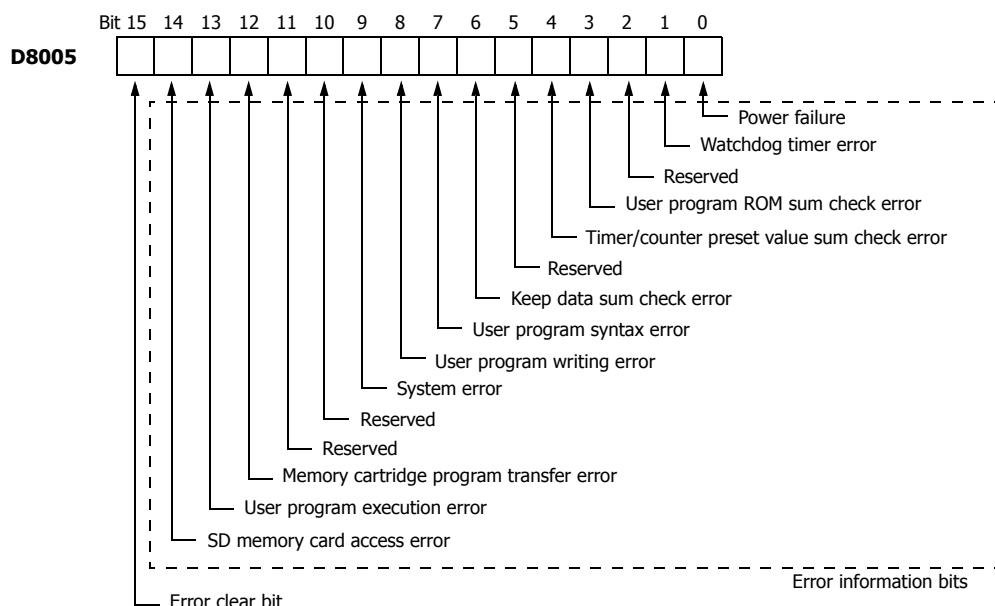
Example: This ladder program clears the error information using the error clear bit of special data register D8005.



## General Error Codes

The error code is stored in special data register D8005.

If '1' is stored in each bit of D8005, the corresponding error is occurring.



## SmartAXIS Operating Status, Output, and ERR LED during Errors

Error Items	Operating Status	Output	ERR LED	Checked at
Power failure	Stop	OFF	OFF	Any time
Watchdog timer error	Stop	OFF	ON	Any time
User program ROM sum check error	Stop	OFF	ON	Starting operation
TIM/CNT preset value sum check error	Maintained	Maintained	ON	When checking a change in timer/counter settings
Keep data error	Maintained/Stop *1	Maintained/OFF *1	OFF	Turning power on
User program syntax error	Stop	OFF	ON	Downloading user program
User program writing error	Stop	OFF	ON	Downloading user program
System error	Stop	OFF	ON	Turning power on
Memory cartridge program transfer error	Stop	OFF	ON	Turning power on
User program execution error	Maintained	Maintained	ON	Executing user program
SD memory card access error	Maintained	Maintained	ON	When an SD memory card is inserted

\*1: Operation starts to run and outputs are turned on or off according to the user program as default, but it is also possible to stop operation and turn off outputs using the Function Area Settings on WindLDR. See "Run/Stop Selection at Memory Backup Error" on page 5-5.

\*2: The SmartAXIS Pro has no error status LED.

## Error Causes and Actions

### 0001h: Power Failure

This error indicates when the power supply is lower than the specified voltage. This error is also recorded when the power is turned off. Clear the error code using WindLDR.

### 0002h: Watchdog Timer Error

The watchdog timer monitors the time required for one program cycle (scan time). When the time exceeds the allowed time, the watchdog timer indicates an error. Clear the error code using WindLDR. If this error occurs frequently, the SmartAXIS has to be replaced.

### 0008h: User Program ROM Sum Check Error

The user program stored in the SmartAXIS ROM is corrupted. Download a correct user program to the SmartAXIS, and clear the error code using WindLDR.

When a memory cartridge is installed on the SmartAXIS, the user program in the memory cartridge is checked.

### 0010h: Timer/Counter Preset Value Sum Check Error

The execution data of timer/counter preset values is corrupted. The timer/counter preset values are initialized to the values of the user program automatically. Note that changed preset values are cleared and that the original values are restored. Clear the error code using WindLDR.

### 0040h: Keep Data Sum Check Error

This error indicates that the data designated to be maintained during power failure is corrupted because of memory backup failure. Note that the "keep" data of internal relays and shift registers are cleared. Data of counters and data registers are also cleared. Clear the error code using WindLDR.

If this error occurs in a short period of power interruption after the battery has been charged as specified, the battery is defective and the SmartAXIS has to be replaced.

### 0080h: User Program Syntax Error

This error indicates that the user program has a syntax error. Correct the user program, and download the corrected user program to the SmartAXIS. The error code is cleared when a correct user program is transferred.

### 0100h: User Program Writing Error

This error indicates a failure of writing into the SmartAXIS ROM when downloading a user program. The error code is cleared when writing into the ROM is completed successfully. If this error occurs frequently, the SmartAXIS has to be replaced.

When a memory cartridge is installed on the SmartAXIS, writing into the memory cartridge is checked.

**0200h: System Error**

This error is issued when ROM is not found. When this error occurred, turn power off and on. Clear the error code using WindLDR. If this error occurs frequently, the SmartAXIS has to be replaced.

**1000h: Memory Cartridge Program Transfer Error**

This error indicates that the user program cannot be downloaded to/uploaded from SmartAXIS using the memory cartridge. The memory cartridge program transfer fails when one of the following conditions is met:

- When the user program in the SmartAXIS is password-protected and the password of the user program in the memory cartridge does not match. Configure the correct password to the user program in the memory cartridge. For details on entering the password, see "User Program Protection" on page 5-60.
- When uploading the user program from the SmartAXIS is prohibited. Upload cannot be executed.

**2000h: User Program Execution Error**

This error indicates that invalid data is found during execution of a user program. When this error occurs, special internal relay M8004 (user program execution error) is turned on.

Referring to the next page to fix and clear the error code.

**User Program Execution Error**

This error indicates that invalid data is found during execution of a user program. When this error occurs, the ERR LED and special internal relay M8004 (user program execution error) turned on. The detailed information of this error can be viewed from the error code stored in special data register D8006 (user program execution error code).

User Program Execution Error Code (D8006)	Error Details
<b>1</b>	Source/destination device exceeds range.
<b>2</b>	MUL result exceeds data type range.
<b>3</b>	DIV result exceeds data type range, or division by 0.
<b>4</b>	BCDLS has S1 or S1+1 exceeding 9999.
<b>5</b>	S1 is 10,000 or higher in the HTOB(W) instruction, or S1 is 100,000,000 or higher in the HTOB(D) instruction.
<b>6</b>	BTOH has any digit of S1 exceeding 9.
<b>7</b>	HTOA/AТОH/BTOA/ATOB has quantity of digits to convert out of range.
<b>8</b>	ATOH/ATOB has non-ASCII data for S1 through S1+4.
<b>9</b>	In the WEEK instruction, the ON time data exceeds range (hour data is 24 or higher, minute data is 60 or higher), or the OFF time data exceeds range (hour data is 25 or higher, minute data is 60 or higher).
<b>10</b>	In the YEAR instruction, the month and day data exceeds range (month data is 13 or higher, day data is 32 or higher).
<b>12</b>	CVXTY/CVYTX is executed without matching XYFS. XYFS and CVXTY/CVYTX have the same S1, but have different data types.
<b>13</b>	CVXTY/CVYTX has S2 exceeding the value specified in XYFS.
<b>14</b>	Label in LJMP, LCAL, or DJNZ is not found.
<b>18</b>	Attempt was made to execute an instruction that cannot be used in an interrupt program: SOTU, SOTD, TML, TIM, TMH, TMS, CNT, CDP, CUD, SFR, SFRN, WEEK, YEAR, MSG, TXD, RXD, DI, EI, XYFS, CVXTY, CVYTX, AVRГ, PULS, PWM, RAMP, ZRN, ARAMP, DTML, DTIM, DTMH, DTMS, TTIM, FIFO, NDSRL, HOUR, ETXD, ERXD, DLOG, and TRACE (see "Notes for Using Interrupt Inputs and Timer Interrupt:" on page 5-31).
<b>19</b>	Attempt was made to execute an instruction that is not available for the PLC.
<b>20</b>	PULS, PWM, RAMP, ZRN, or ARAMP has an invalid value in control registers.
<b>21</b>	DECO has S1 exceeding 255.
<b>22</b>	BCNT has S2 exceeding 256.
<b>23</b>	ICMP>= has S1 < S3.
<b>25</b>	BCDLS has S2 exceeding 7.
<b>26</b>	DI or EI is executed when interrupt input or timer interrupt is not programmed in the Function Area Settings.
<b>27</b>	Work area is broken when using DTML, DTIM, DTMH, DTMS, or TTIM.
<b>28</b>	S1 for trigonometric function instruction is invalid.
<b>29</b>	Result of F (float) data type instruction exceeds the data type range.
<b>30</b>	N_B for SFTL/SFTR exceeds range.
<b>31</b>	In the FIFO instruction, the FIEX instruction or the FOEX instruction was executed before the FIFO data file was registered.
<b>32</b>	TADD, TSUB, HOUR, or HTOS has invalid data for source device S1.
<b>34</b>	NDSRC has invalid data for source device S3.
<b>35</b>	In the SUM instruction, the execution result exceeds the valid range for the selected data type, or S2 data is 0.
<b>36</b>	DLOG/TRACE is executed but the capacity of CSV file exceeds 5 MB.
<b>41</b>	SD memory card is write protected.

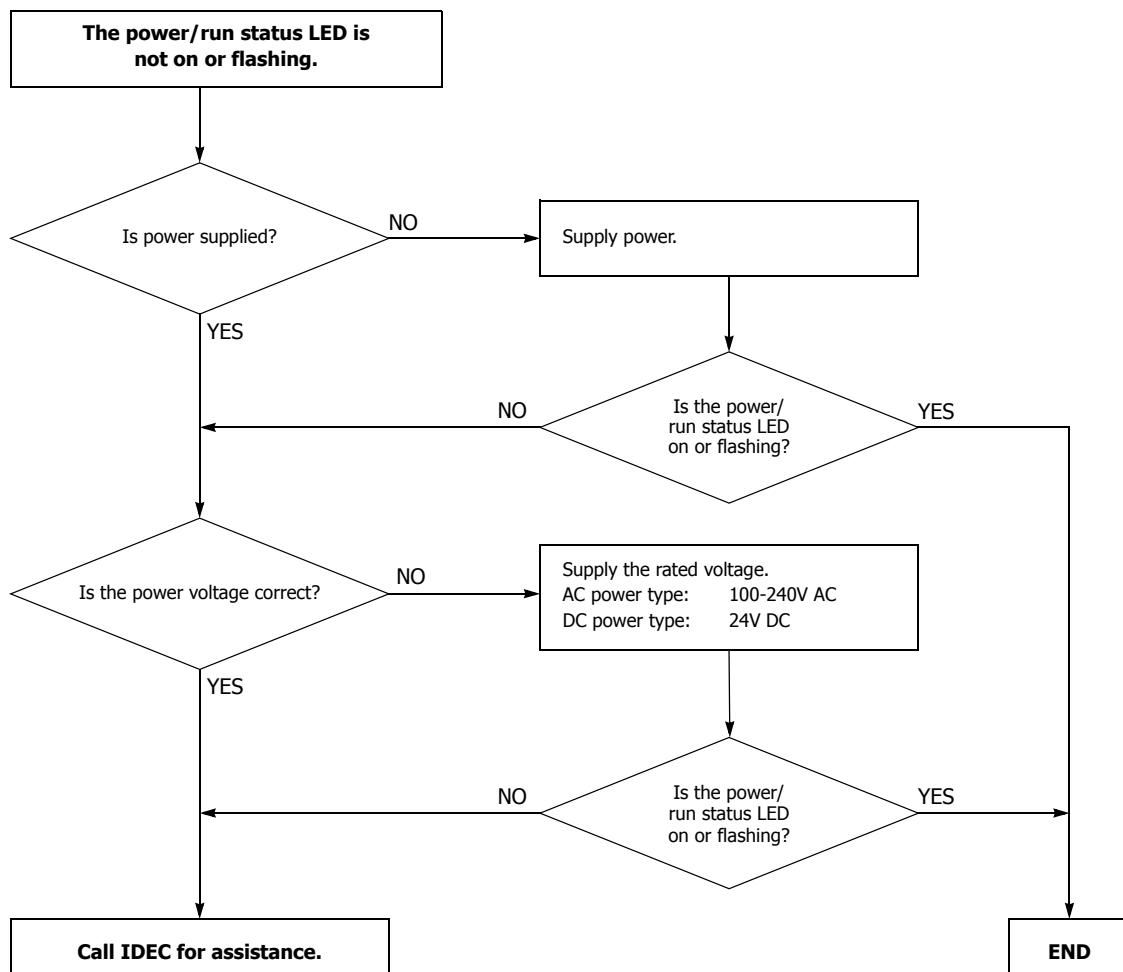
## Troubleshooting Diagrams

When one of the following problems is encountered, see the following trouble shooting diagrams:

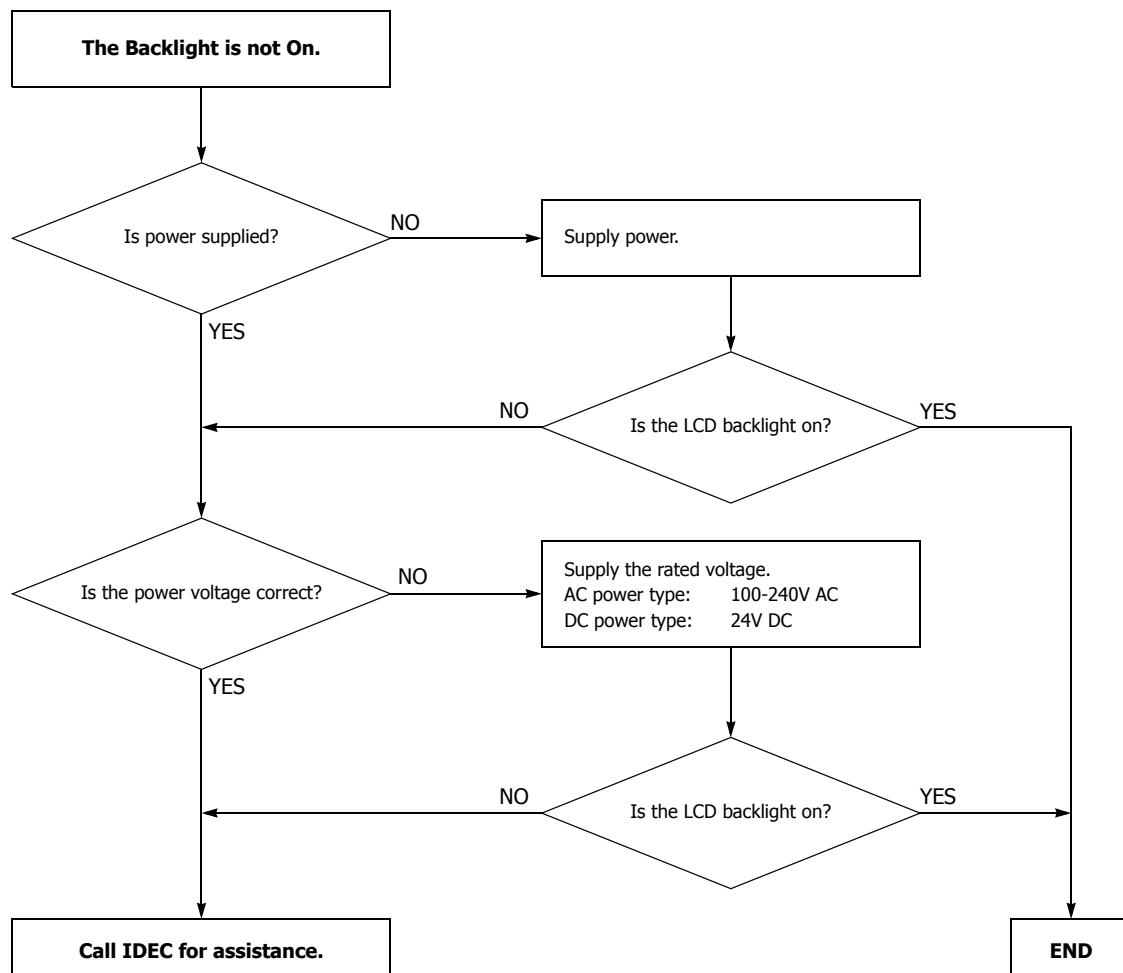
Problem	Troubleshooting Diagram
The power is not on.	Diagram 1
Operation has not started.	Diagram 2
An error has occurred.	Diagram 3
Input does not operate normally.	Diagram 4
Output does not operate normally.	Diagram 5
Communication between WindLDR on a computer and the SmartAXIS is not possible.	Diagram 6
Cannot stop or reset operation.	Diagram 7
Watchdog timer error occurs and the CPU does not run.	Diagram 8
The interrupt/catch input cannot receive short pulses.	Diagram 9
Frequency measurement does not work.	Diagram 10
Data is not transmitted at all in the user communication mode. (24-, 40-, 48-I/O types only)	Diagram 11
Data is not transmitted correctly in the user communication mode. (24-, 40-, 48-I/O types only)	Diagram 12
Data is not received at all in the user communication mode. (24-, 40-, 48-I/O types only)	Diagram 13
Data is not received correctly in the user communication mode. (24-, 40-, 48-I/O types only)	Diagram 14
Modbus master communication does not work. (24-, 40-, 48-I/O types only)	Diagram 15
Long communication cycle for Modbus RTU master communication or Modbus TCP client communication. (24-, 40-, 48-I/O types only)	Diagram 16
WindLDR and the SmartAXIS cannot communicate.	Diagram 17
WindLDR does not communicate with PLC via USB.	Diagram 18
Modbus master communication request is slow.	Diagram 19

**Troubleshooting Diagram 1****• SmartAXIS Lite - The Power/Run Status LED (PWR/RUN) is not On or Flashing**

The SmartAXIS Lite power/run status LED (PWR/RUN) is on while running and flashes while stopped.



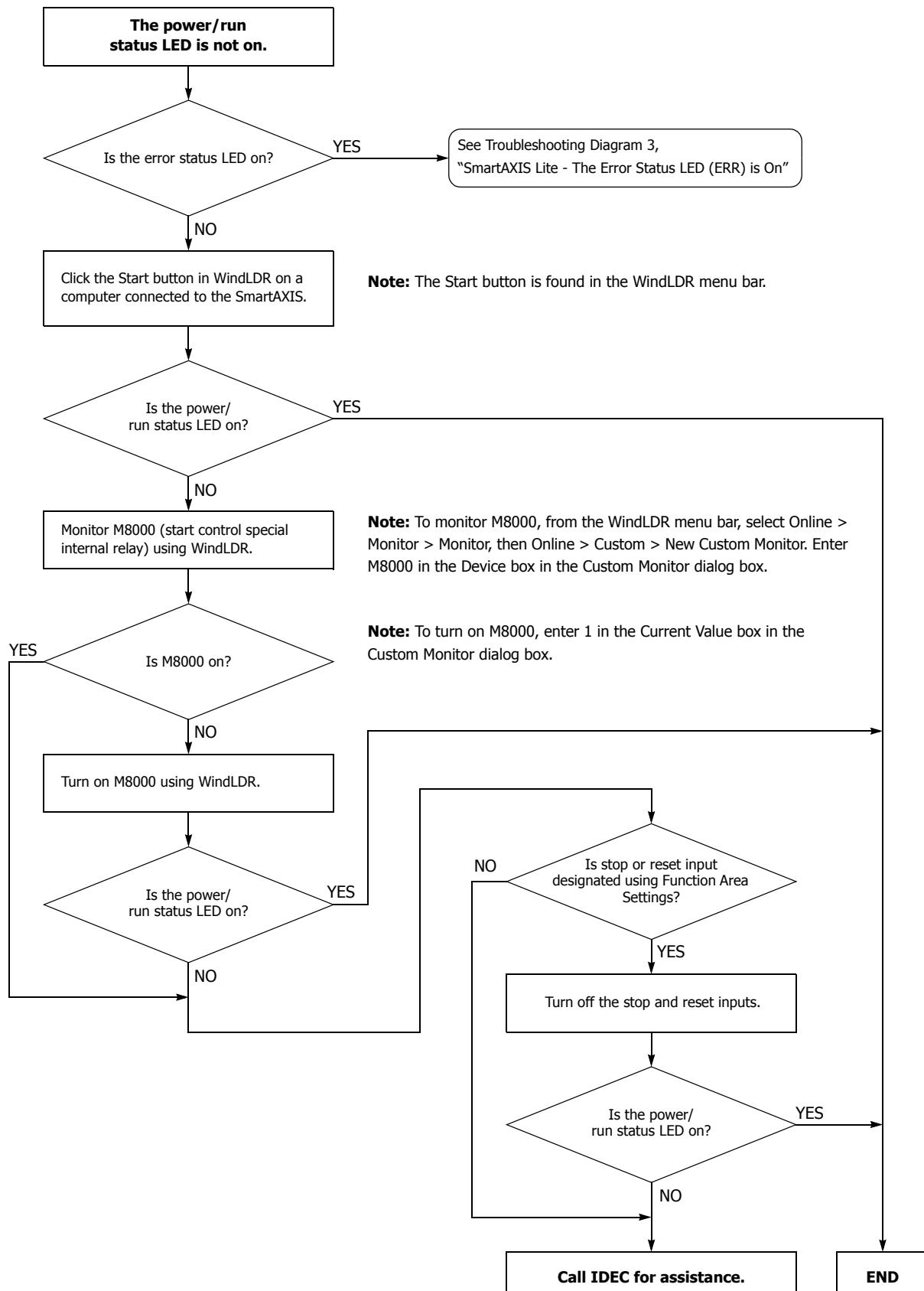
- SmartAXIS Pro - The Backlight is not On



## Troubleshooting Diagram 2

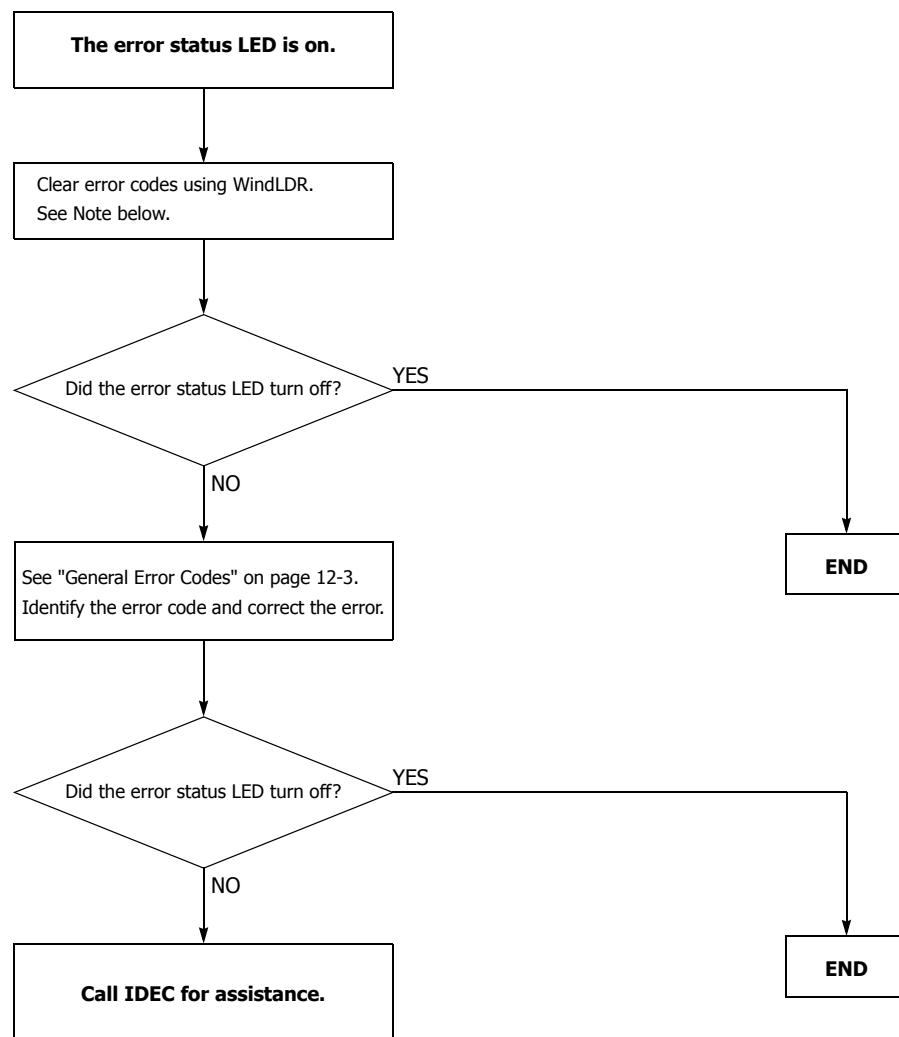
- SmartAXIS Lite - The Power/Run Status LED (PWR/RUN) is not On**

The SmartAXIS Lite power/run status LED (PWR/RUN) is on while running and flashes while stopped.



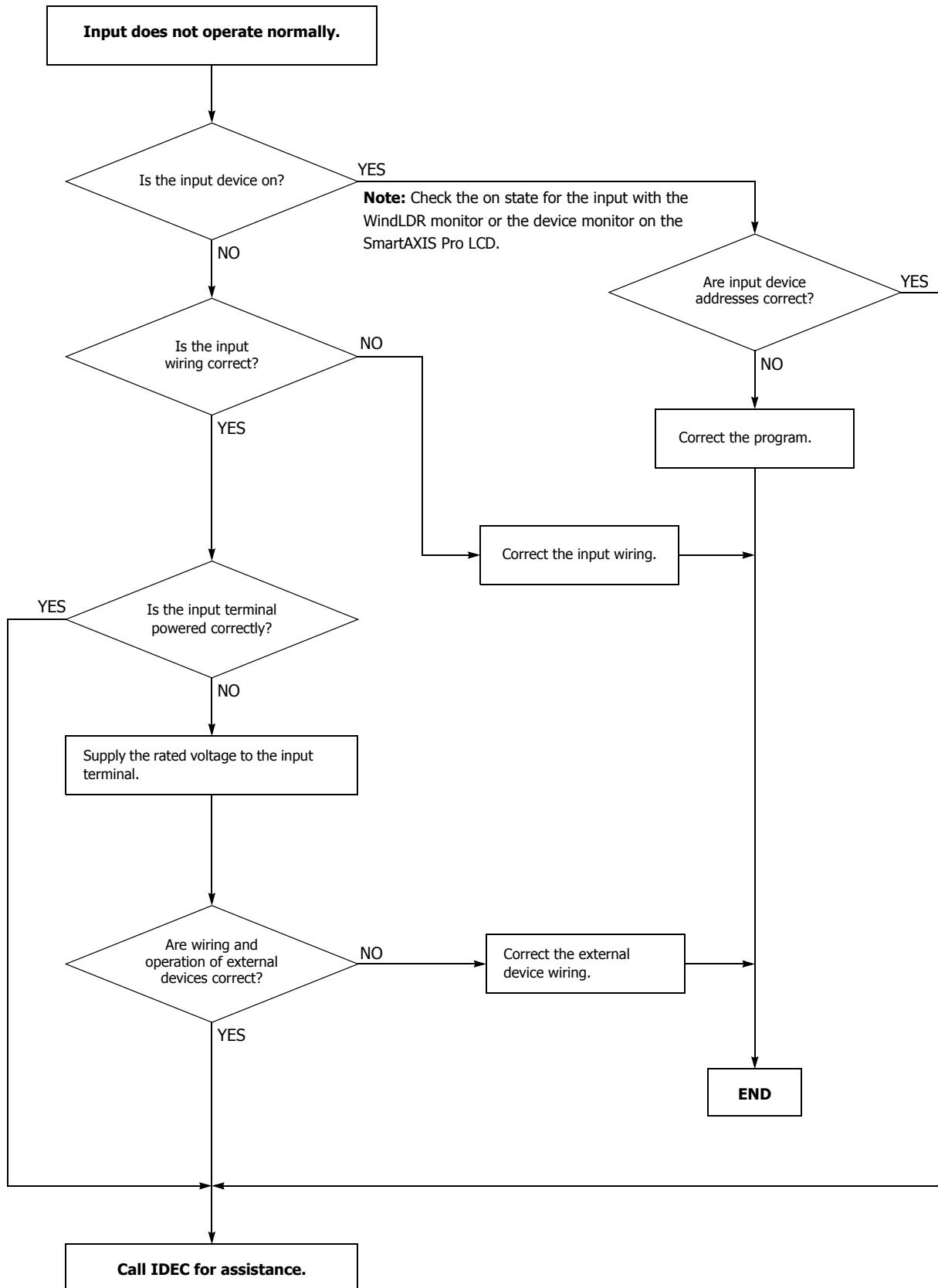
**Troubleshooting Diagram 3**

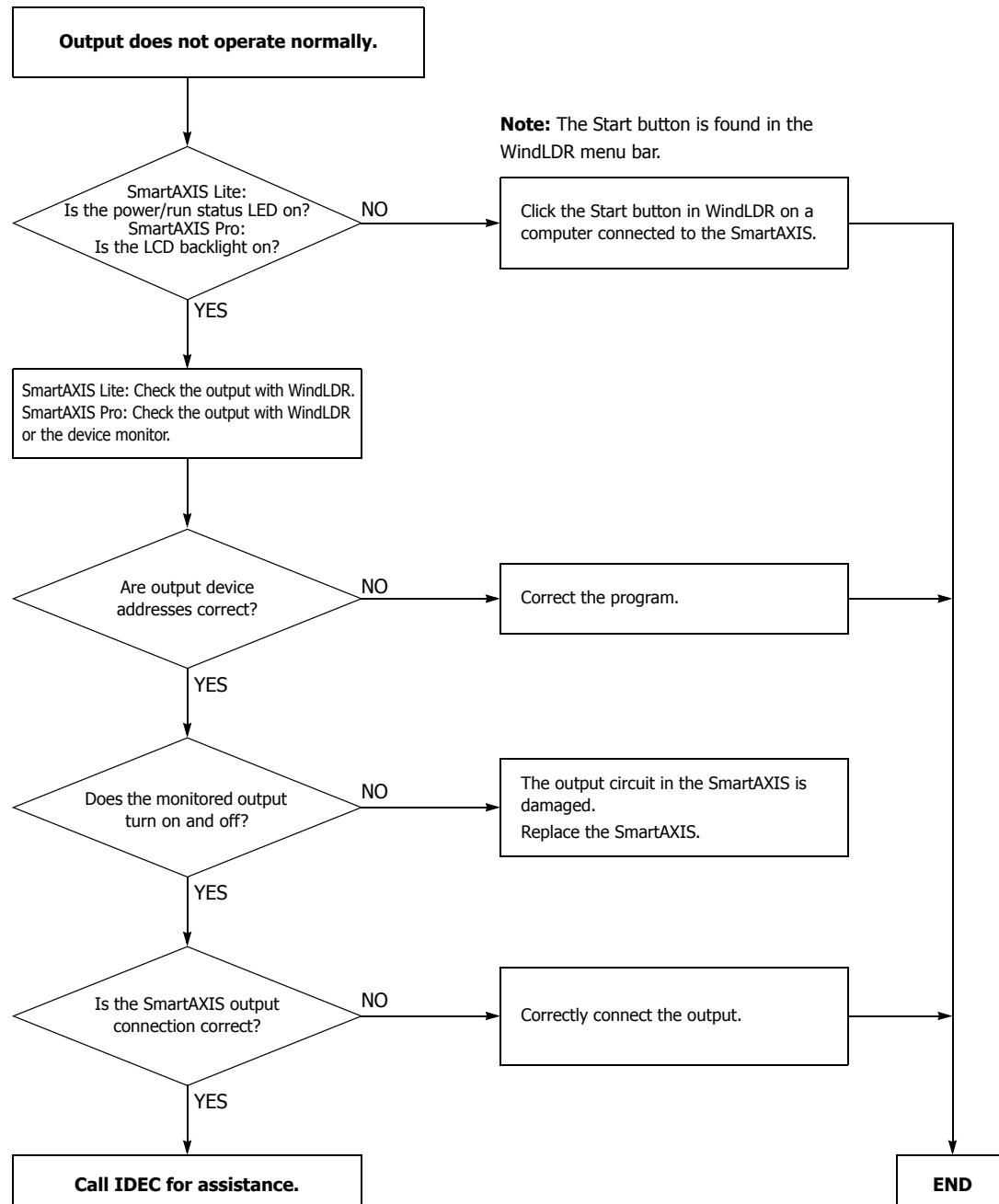
- SmartAXIS Lite - The Error Status LED (ERR) is On



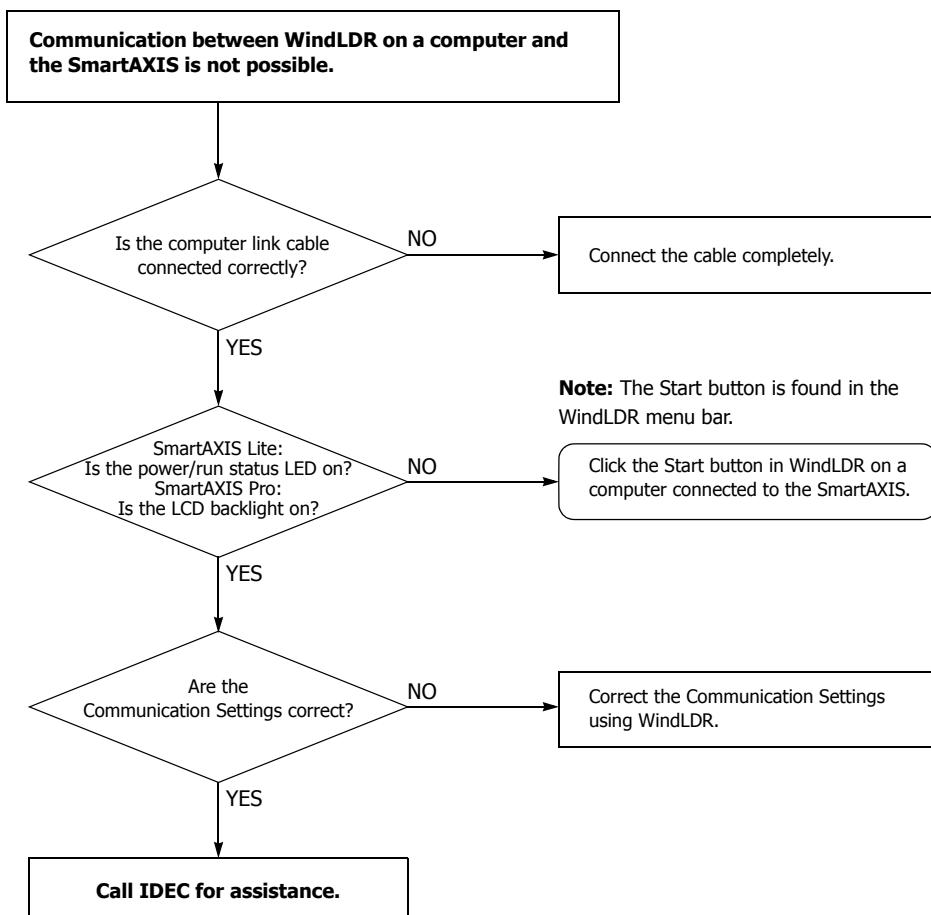
**Note:** Temporary errors can be cleared to restore normal operation by clearing error codes from WindLDR. See "Clearing Error Codes from WindLDR" on page 12-2.

**Troubleshooting Diagram 4**

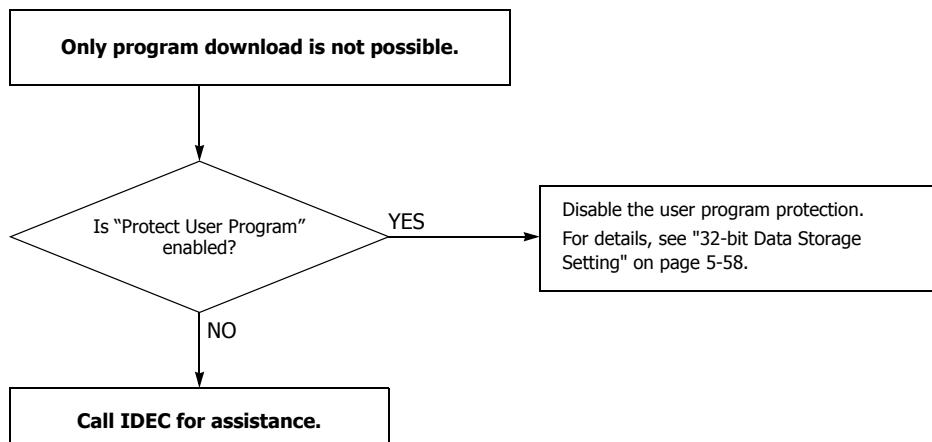


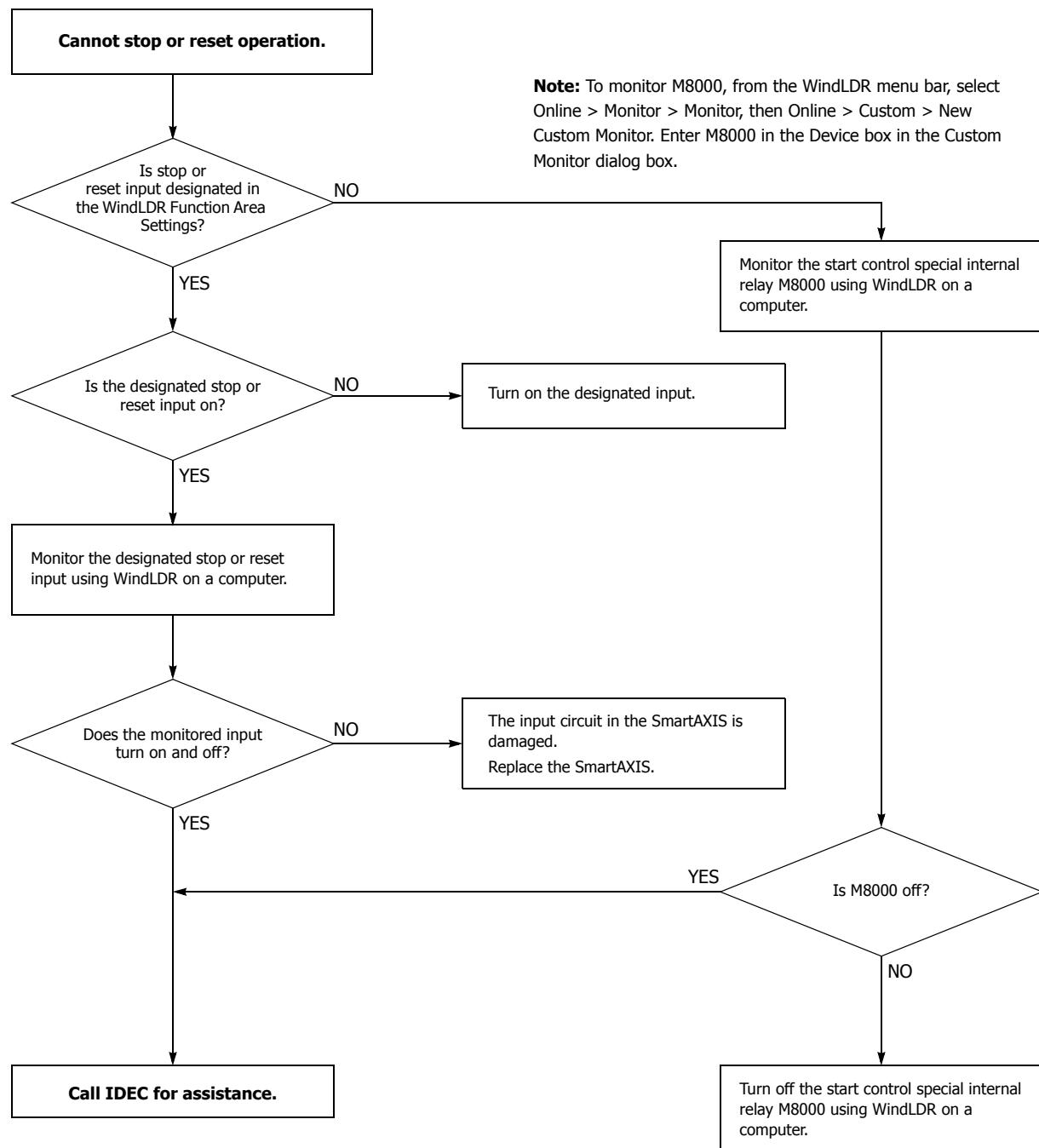
**Troubleshooting Diagram 5**

### Troubleshooting Diagram 6

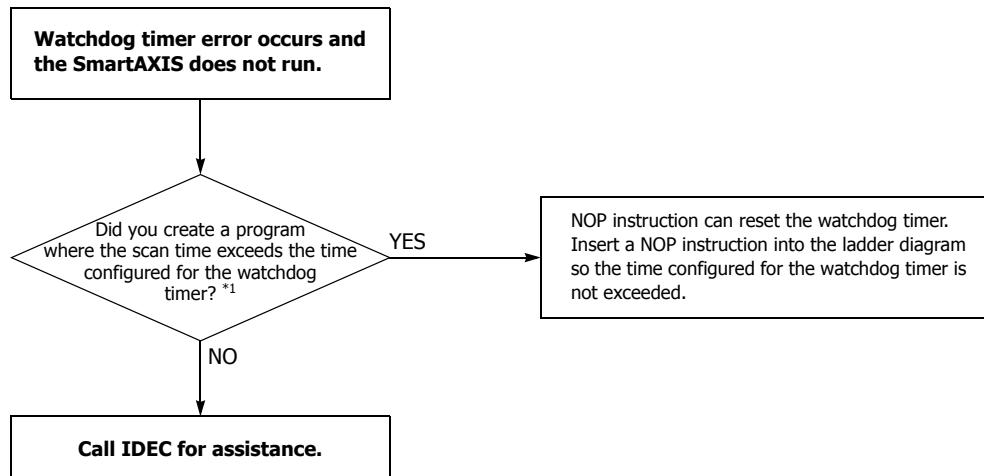


**When only program download is not possible:**

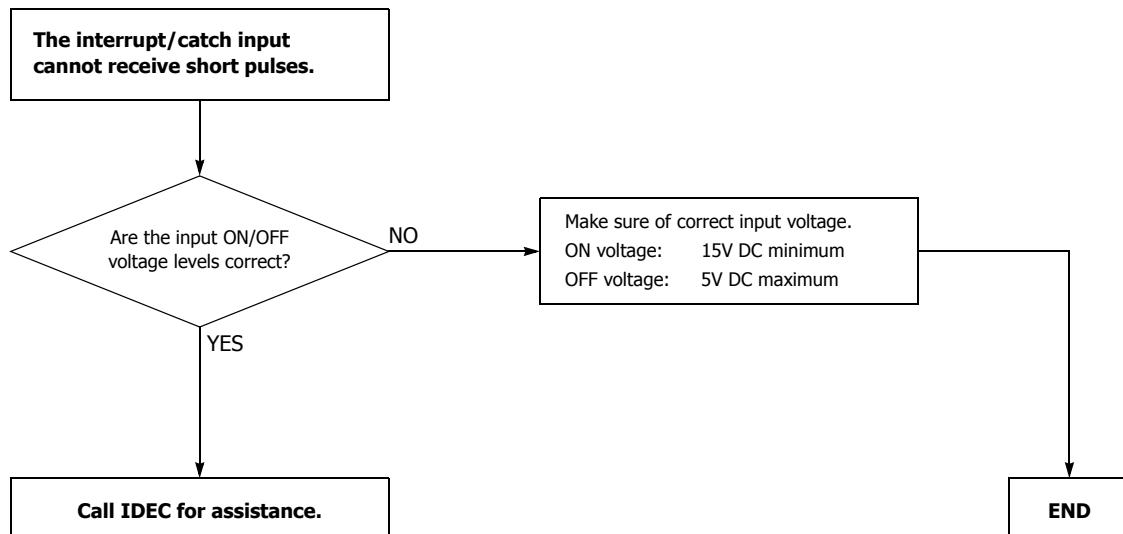


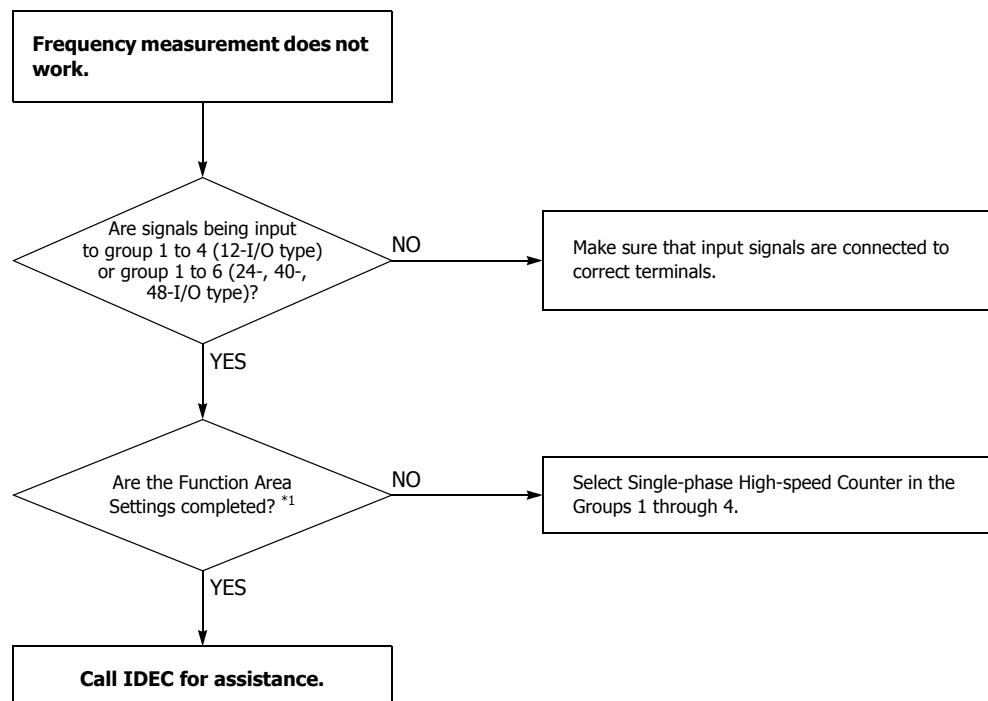
**Troubleshooting Diagram 7**

**Note:** To turn off M8000, enter 0 in the Current Value box in the Custom Monitor dialog box.

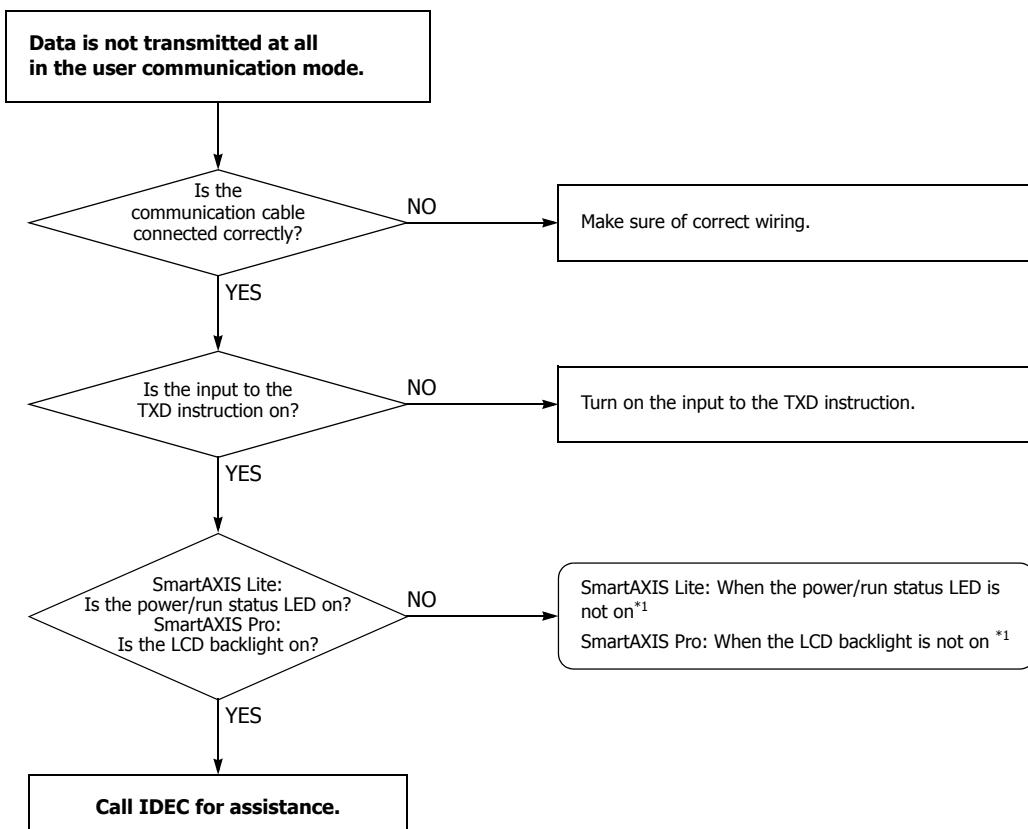
**Troubleshooting Diagram 8**

\*1 See "Watchdog Timer Setting" on page 5-62.

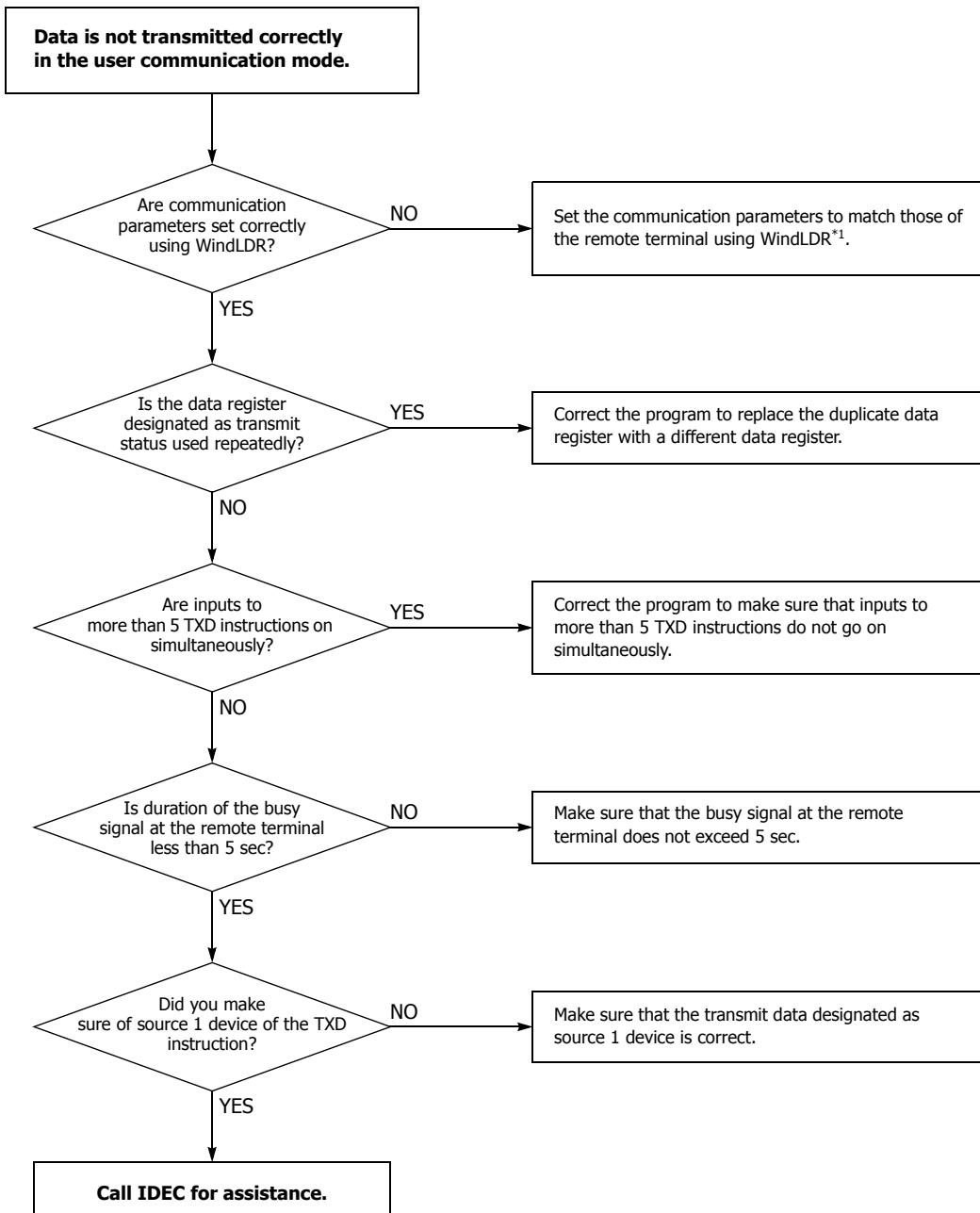
**Troubleshooting Diagram 9**

**Troubleshooting Diagram 10**

\*1 See "Frequency Measurement" on page 5-32.

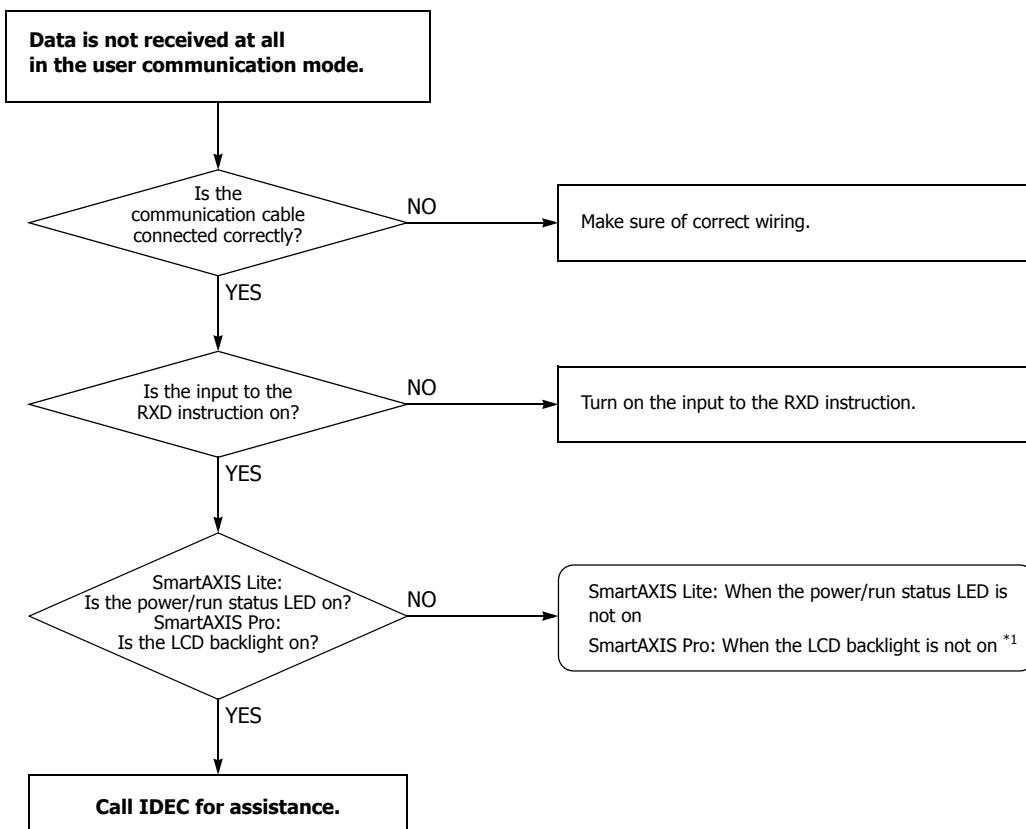
**Troubleshooting Diagram 11**

\*1 "SmartAXIS Pro - The Backlight is not On" on page 12-9 or "SmartAXIS Lite - The Power/Run Status LED (PWR/RUN) is not On" on page 12-10.

**Troubleshooting Diagram 12**

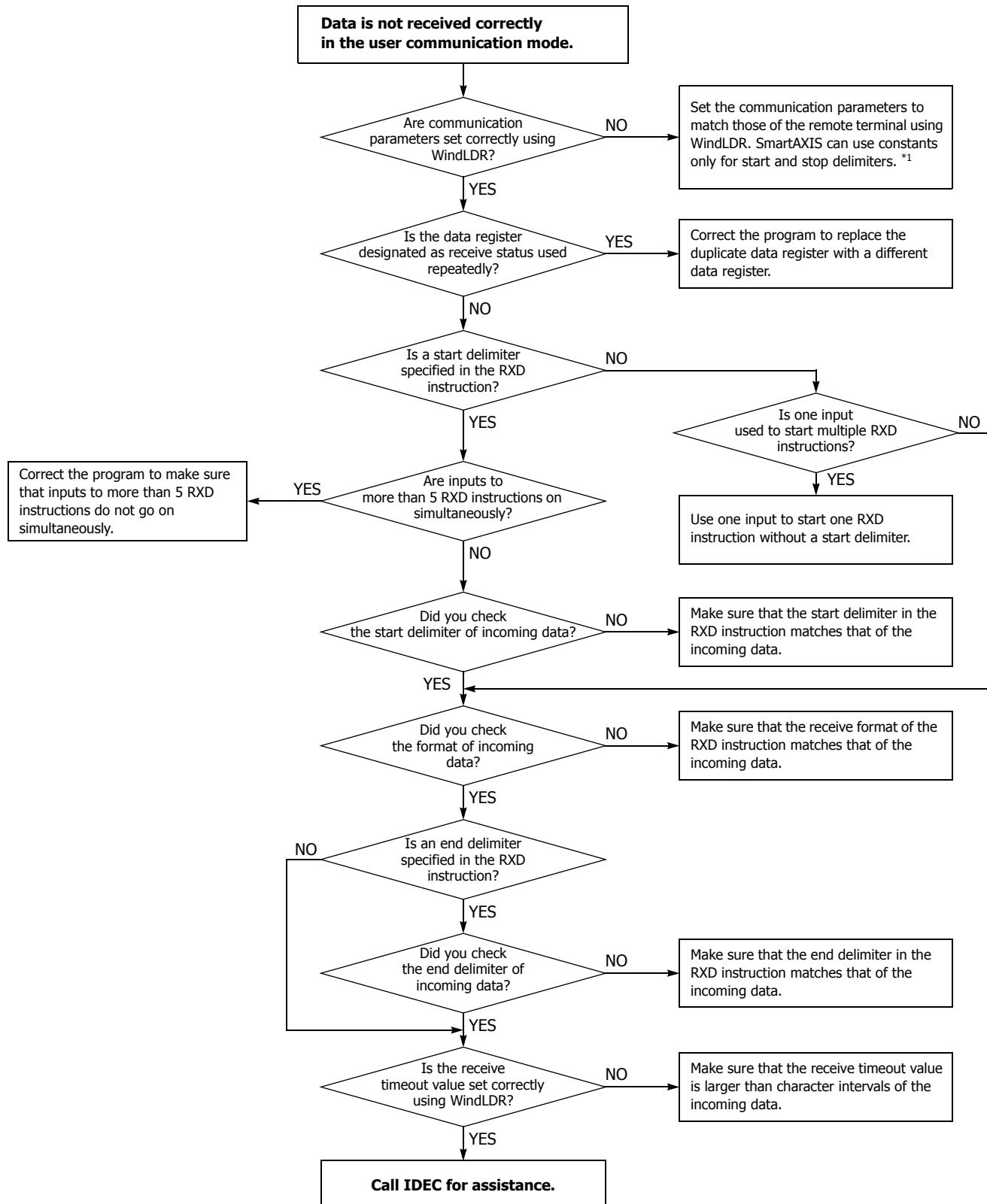
\*1 See "Programming WindLDR" on page 10-4.

If the user communication still has a problem after completing the above procedure, also perform the procedure of Diagram 11, described on the preceding page.

**Troubleshooting Diagram 13**

\*1 See "SmartAXIS Pro - The Backlight is not On" on page 12-9 or "SmartAXIS Lite - The Power/Run Status LED (PWR/RUN) is not On" on page 12-10.

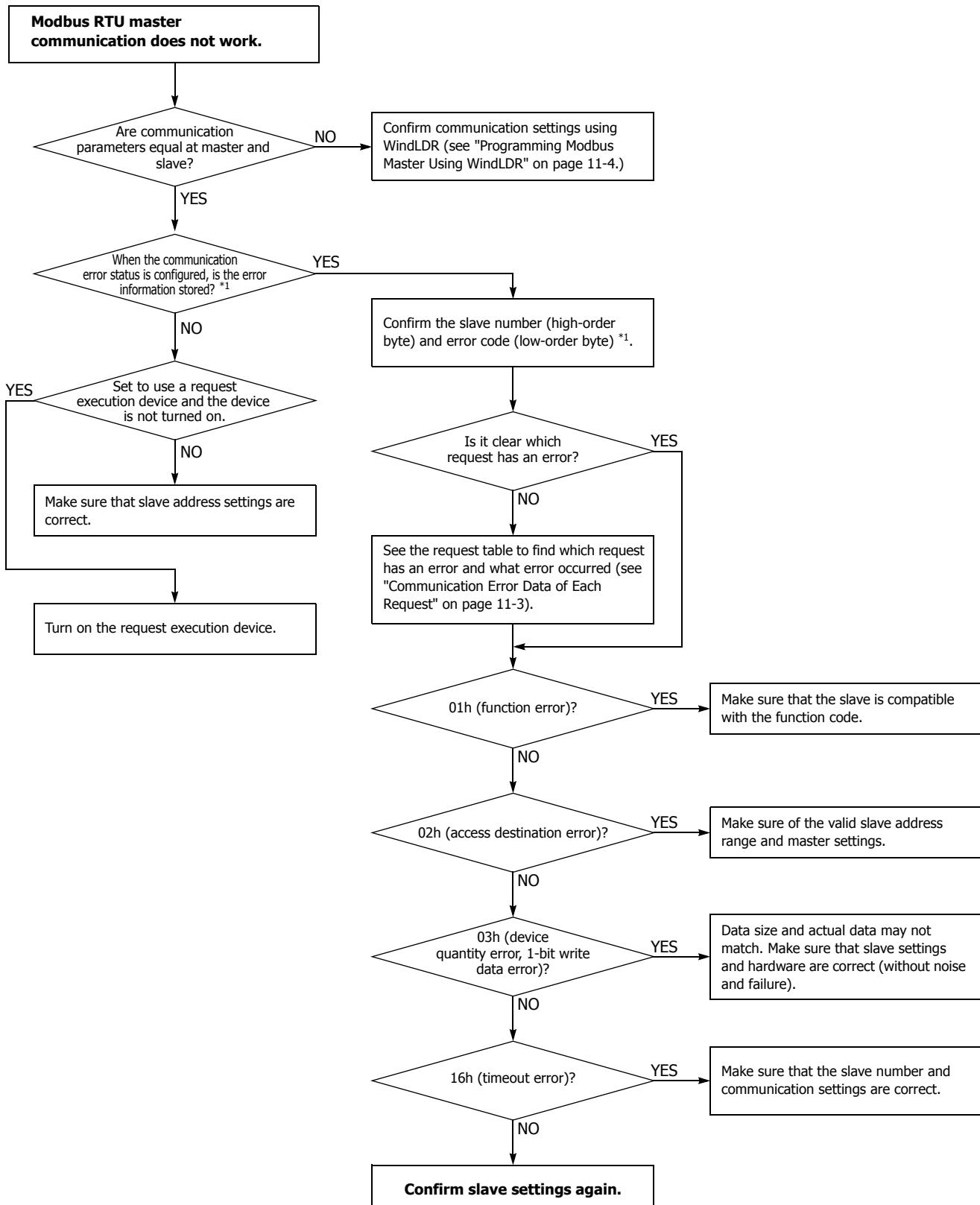
## Troubleshooting Diagram 14



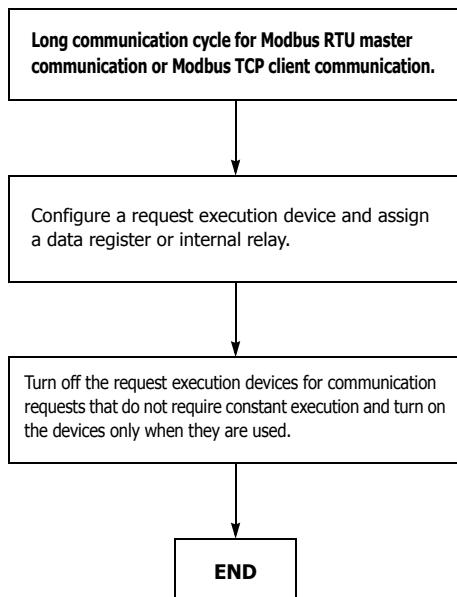
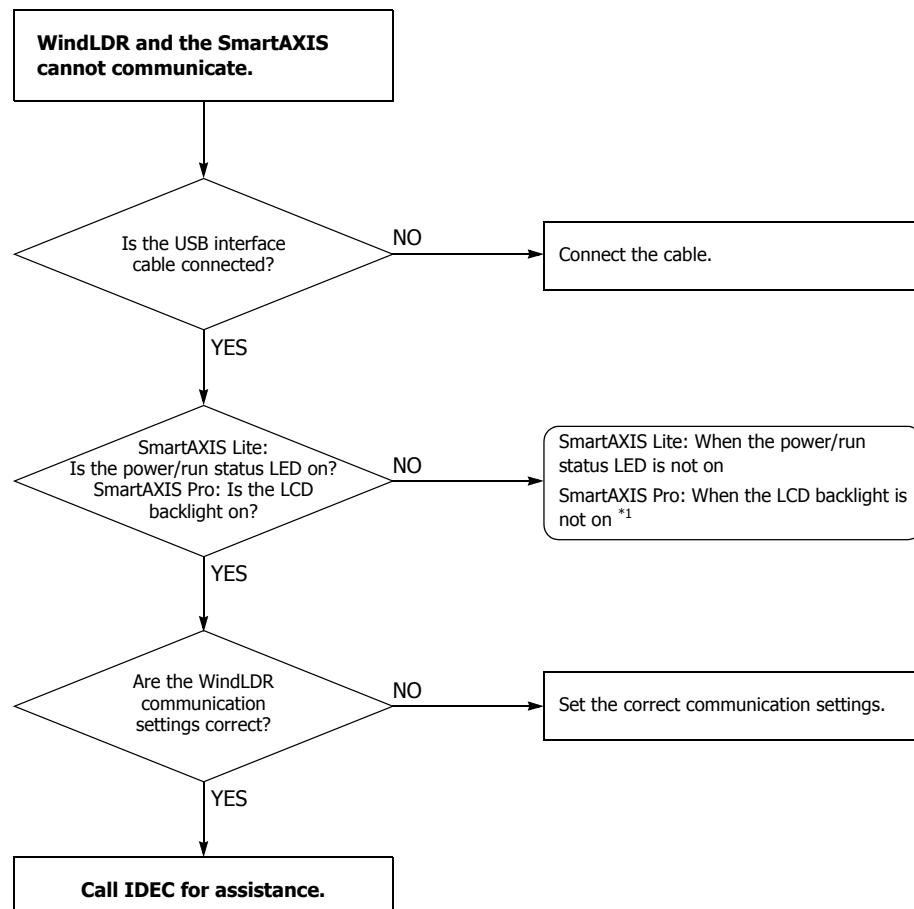
\*1 See "Programming WindLDR" on page 10-4.

If the user communication still has a problem after completing the above procedure, also perform the procedure of Diagram 11 described on the preceding page.

**Troubleshooting Diagram 15**

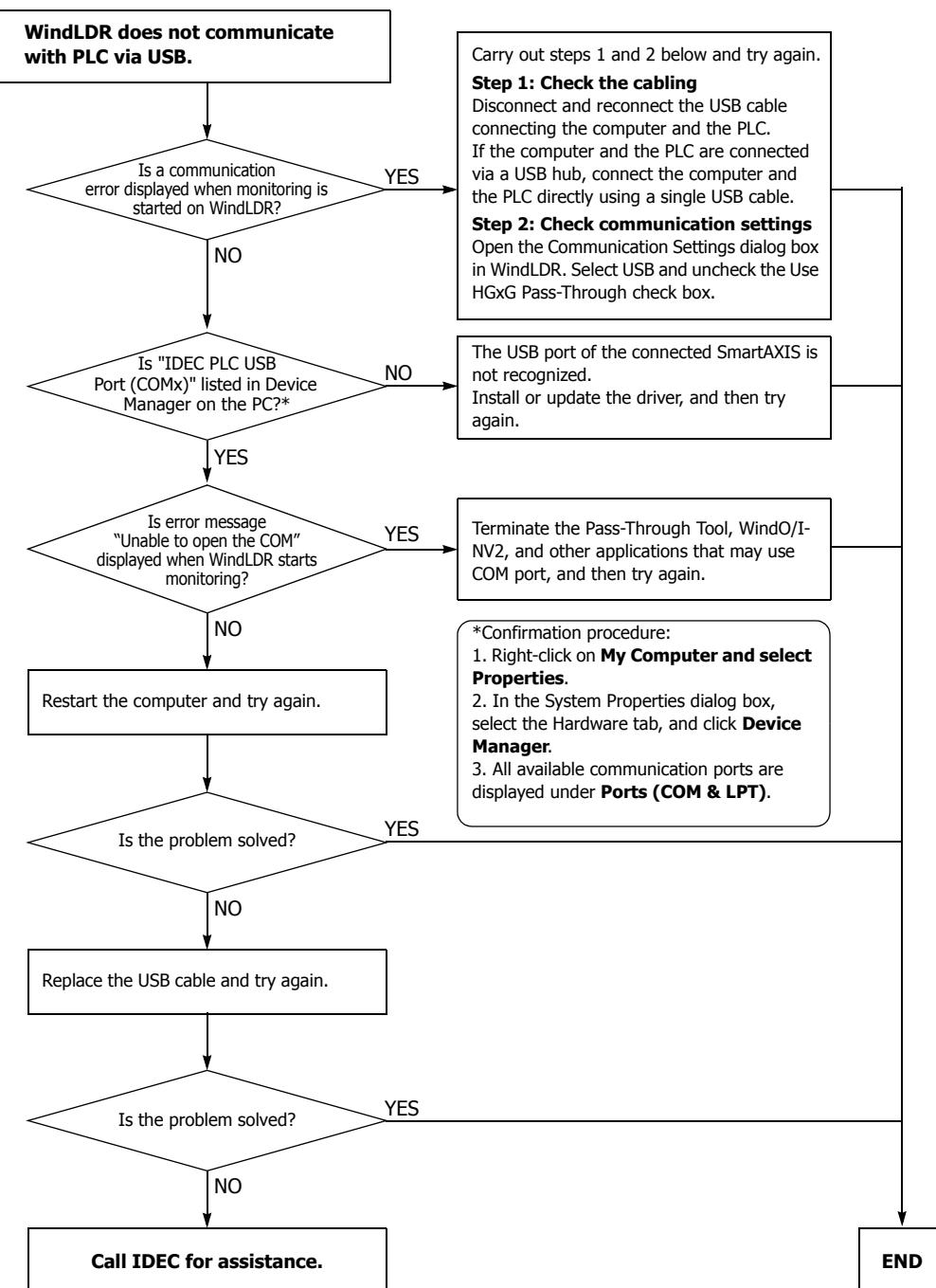


\*1 See "Communication Error Data" on page 11-3.

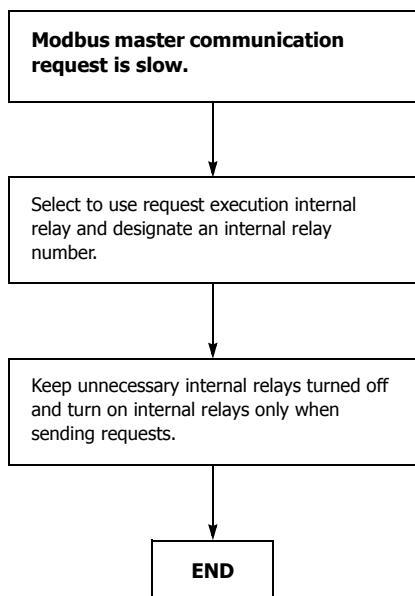
**Troubleshooting Diagram 16****Troubleshooting Diagram 17**

\*1 See "SmartAXIS Pro - The Backlight is not On" on page 12-9 or "SmartAXIS Lite - The Power/Run Status LED (PWR/RUN) is not On" on page 12-10.

**Troubleshooting Diagram 18**



**Troubleshooting Diagram 19**





# APPENDIX

---

---

## System Software

### Getting the Latest Version of the System Software

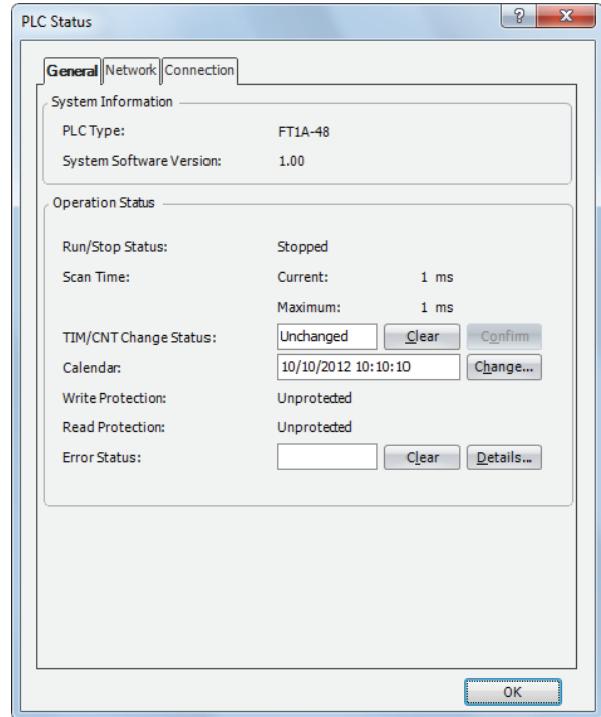
When you install or upgrade to the latest version of Automation Organizer, the latest system software is also available along with the software.

### Checking the Version

You can check the SmartAXIS system software version with the following steps.

1. Connect the computer to the SmartAXIS USB port with the USB maintenance cable (HG9Z-XCM42).
2. From the WindLDR menu bar, select **Online > Monitor > Start Monitor**.  
WindLDR enters monitor mode.
3. Select **Online > PLC > Status**.

The PLC Status dialog box is displayed.



You can check the SmartAXIS system software version with **System Software Version**.

**Note:** SmartAXIS system software downloads can be executed from WindLDR.

For how to download the system software, see "Upgrade SmartAXIS System Software" on page A-2.

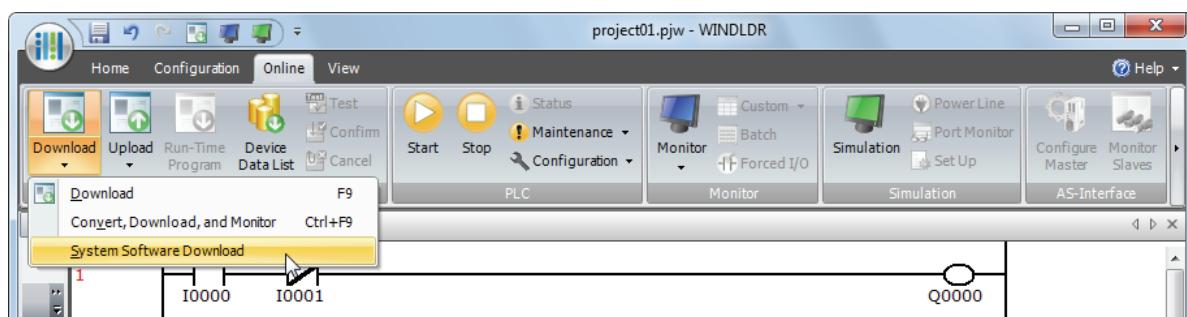
## Upgrade SmartAXIS System Software

The system software of SmartAXIS can be upgraded using WindLDR. If the system software of your SmartAXIS is old, upgrade the system software with the following procedure:

1. Connect the SmartAXIS to the computer where WindLDR is installed using USB maintenance cable (recommended cable: HG9Z-XCM42).

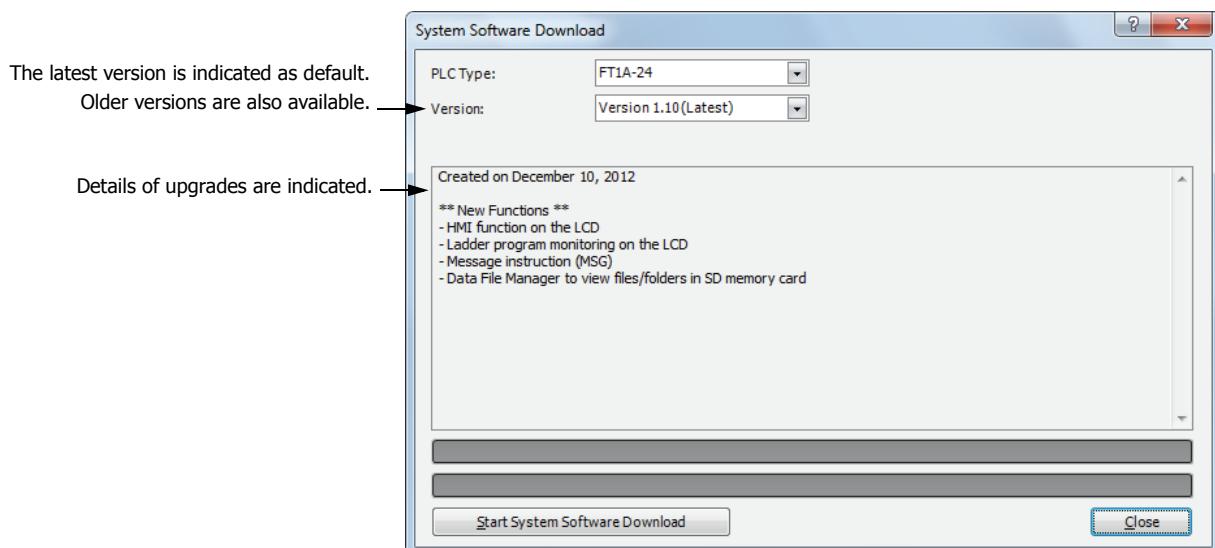
**Note:** System software cannot be downloaded via the expansion communication ports or the Ethernet port.

2. From the WindLDR menu bar, select **Online > Download > System Software Download**.



The System Software Download dialog box appears.

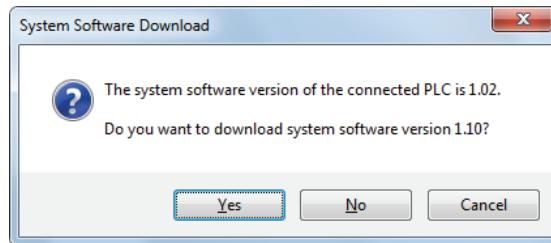
3. Select the PLC type and the system software version to download, and then click **Start System Software Download**.



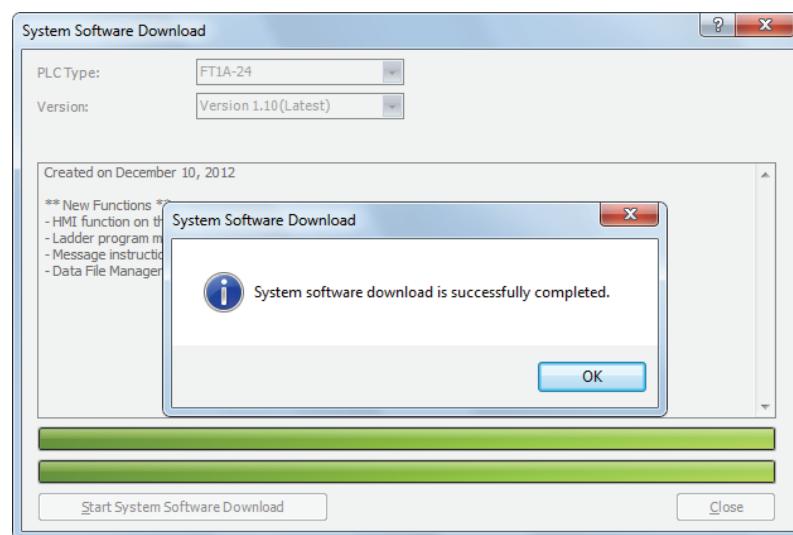
### Notes:

- If the SmartAXIS is running, it is stopped automatically before the system software download starts.
- Older system software can also be downloaded to the SmartAXIS if required.
- The system software download takes about one minute.

4. A confirmation message is displayed. Click **OK** to start downloading the system software.



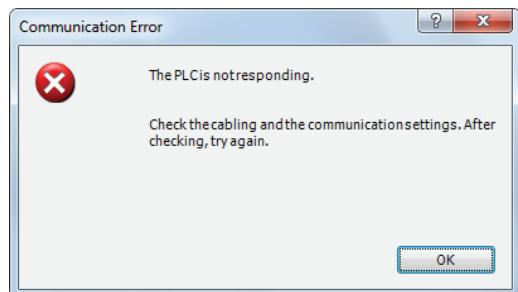
5. System software download status is shown by a progress bar. A completion message appears when the system software has been downloaded successfully. Click **OK** to close the System Software Download dialog box.



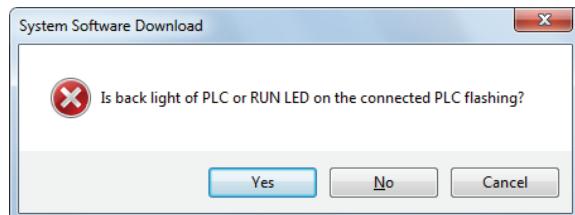
**Notes:**

- For the SmartAXIS Pro, its backlight will flash while the system software is downloading. For the SmartAXIS Lite, its error status LED will flash.
- After the system software has downloaded, the SmartAXIS will be in the STOP state. Start the SmartAXIS with WindLDR or the SmartAXIS Pro LCD/operation switches.
- The user program stored in the SmartAXIS before downloading the system software remains and is executed when the SmartAXIS is restarted. A user program execution error may occur if an older system software is downloaded to the SmartAXIS.
- If the system software download fails, the backlight on the SmartAXIS Pro will remain flashing or the error status LED on the SmartAXIS Lite will remain flashing. In this situation, turn on the SmartAXIS power again and redo the download from step 1. If the SmartAXIS Pro backlight is still flashing or the SmartAXIS Lite error status LED is still flashing even after turning on the power again, the communication error dialog box is displayed before step 4.

1. A communication error dialog box is displayed. Click **Cancel**.



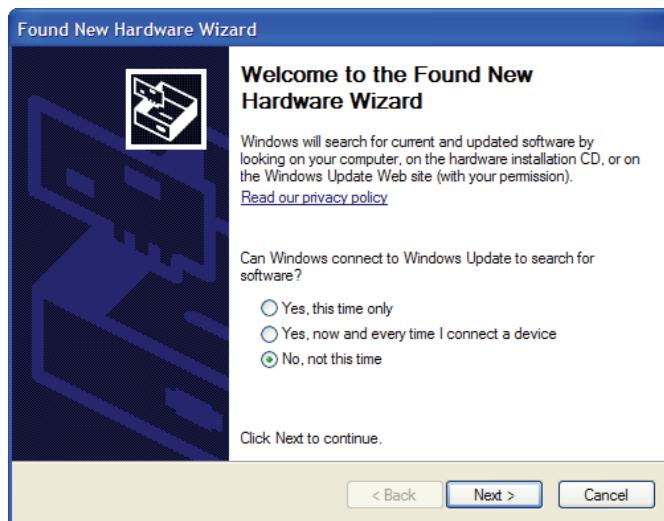
2. A dialog box prompting you to check the status of the SmartAXIS is displayed. Click **Yes**.



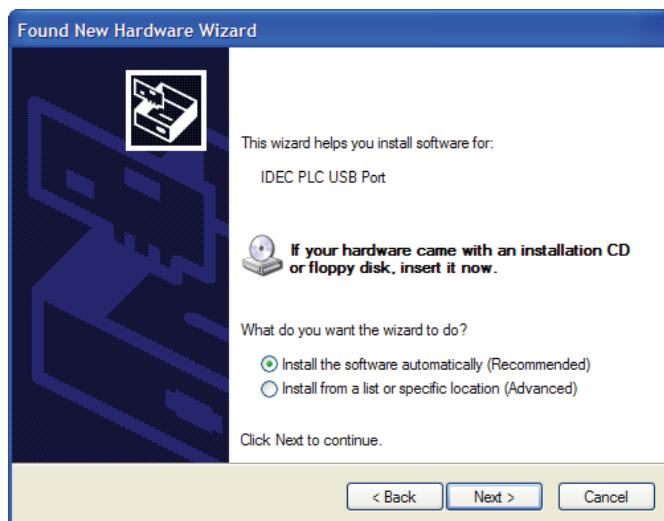
## USB Driver Installation Procedure

### Installing on Windows XP

1. Connect a USB cable to the USB port on the SmartAXIS (with power supplied) while the computer is running.
2. After the message "Found New Hardware" appears on the computer, the driver installation wizard appears. Select **No, not this time** and click **Next**.



3. Select **Install the software automatically (Recommended)** and click **Next**.



4. Click **Continue Anyway** button.



If the installation is aborted without showing the dialog box above, **Block** is selected in the Driver Signing Options in **Driver Signing** in the **System Properties** dialog box. If the installation is aborted, this option should be changed to **Warn**. After changing the option to **Warn**, reconnect the SmartAXIS and retry the installation procedure from the step 1.

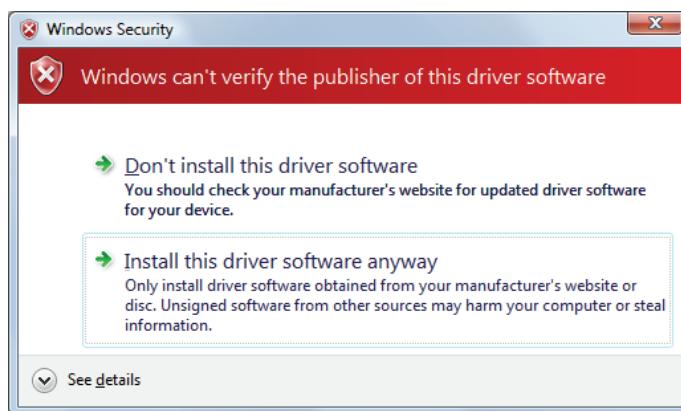
5. Installation is now complete. Click **Finish**.

## Installing on Windows Vista

1. Connect a USB cable to the USB port on the SmartAXIS (with power supplied) while the computer is running.
2. The dialog box below appears. Select **Locate and install driver software (recommended)**.



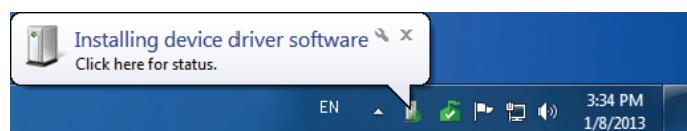
3. The User Account Control dialog box appears. Click **Continue** button.
4. The dialog box below appears. Select **Install this driver software anyway**.



5. Installation is complete once the message "Your devices are ready to use" appears in the bottom right corner of the desktop.

## Installing on Windows 7 32-bit or 64-bit Edition

1. Connect a USB cable to the USB port on the SmartAXIS (with power supplied) while the computer is running.
2. The following message appears in the bottom right corner of the desktop.



3. Installation is complete once the message "Your device is ready to use" appears in the bottom right corner of the desktop.

## Cables

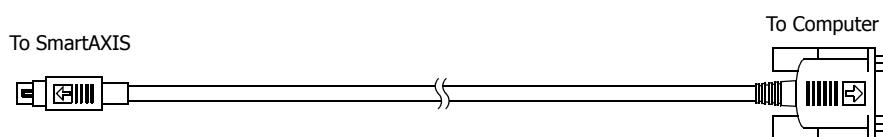
Communication cables and their connector pinouts are described in this section.

### Communication Port and Applicable Cables

Communication Port	Communication Cartridge	Applicable Cable
<b>Expansion Communication Port 2</b> <b>Expansion Communication Port 3</b>	RS232C communication cartridge FT1A-PC1 (Mini-DIN type)	FC2A-KC4C FC2A-KP1C HG9Z-XC295 FC4A-KC1C FC4A-KC2C
	RS485 communication cartridge FT1A-PC2 (Mini-DIN type)	FC2A-KP1C
	RS485 communication cartridge FT1A-PC3 (Terminal block type)	Twisted-pair shielded cable with a minimum core wire of 0.3mm <sup>2</sup> (Conductor resistance 85Ω/km maximum, shield resistance 20Ω/km maximum)

### Computer Link Cable 4C (FC2A-KC4C)

**Cable Length: 3m (9.84 feet)**

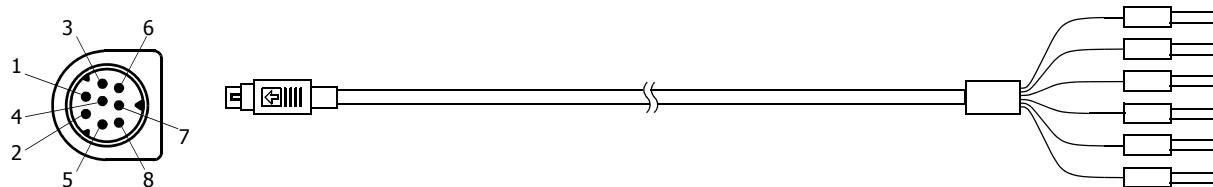


**Mini DIN Connector Pinouts**

Description	Pin
Shield	Cover
TXD Transmit Data	3
RXD Receive Data	4
RTS Request to Send	1
NC No Connection	8
DSR Data Set Ready	5
DTR Data Terminal Ready	2
SG Signal Ground	7
SG Signal Ground	6

**D-sub 9-pin Female Connector Pinouts**

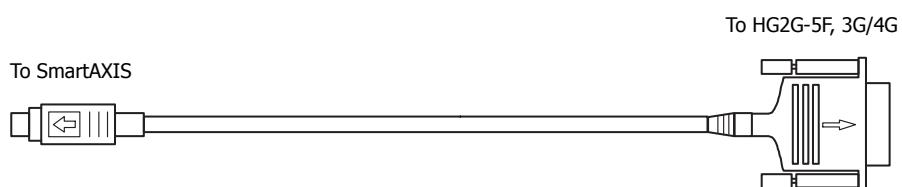
Pin	Description
Cover	FG Frame Ground
3	TXD Transmit Data
2	RXD Receive Data
6	DSR Data Set Ready
8	CTS Clear to Send
1	DCD Data Carrier Detect
4	DTR Data Terminal Ready
5	SG Signal Ground
7	NC No Connection
9	NC No Connection

**User Communication Cable 1C (FC2A-KP1C)****Cable Length: 2.5m (8.2 feet)****Mini DIN Connector Pinouts**

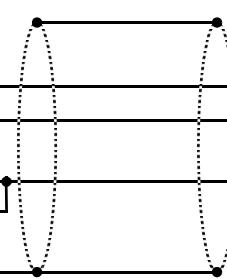
Pin	Signal line (RS232C)	Signal line (RS485)	AWG#	Color	Signal Direction
1	RTS Request to Send	B	28	Black	→
2	DTR Data Terminal Ready	A	28	Yellow	
3	TXD Transmit Data	NC No Connection	28	Blue	→
4	RXD Receive Data	NC No Connection	28	Green	
5	DSR Data Set Ready	NC No Connection	28	Brown	←
6	SG Signal Ground	NC No Connection	28	Gray	—
7	SG Signal Ground	SG Signal Ground	26	Red	—
8	NC No Connection	NC No Connection	26	White	—
Cover	—	—	—	Shield	—



- Do not connect any wiring to NC terminals, otherwise operation failure or device damage may be caused.

**O/I Communication Cable (HG9Z-XC295)****Cable Length: 5m (16.4 feet)****D-sub 9-pin Female Connector Pinouts**

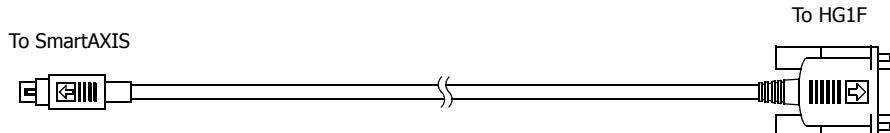
Description	Pin
RS	1
ER	2
SD	3
RD	4
DR	5
SG	6
SG	7
NC	8
Shield	Cover

**Mini DIN Connector Pinouts**

Pin	Description
2	RD
3	SD
5	SG
7	RS
8	CS
Cover	Shield

**O/I Communication Cable 1C (FC4A-KC1C)**

**Cable Length: 5m (16.4 feet)**



**Mini DIN Connector Pinouts**

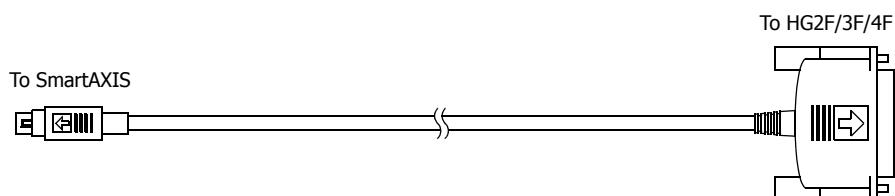
Description	Pin
NC	1
NC	2
TXD	3
RXD	4
NC	5
SG	6
SG	7
NC	8
Shield	Cover

**D-sub 9-pin Male Connector Pinouts**

Pin	Description
1	FG Frame Ground
2	TXD1 Transmit Data 1
3	RXD1 Receive Data 1
4	TXD2 Transmit Data 2
5	RXD2 Receive Data 2
6	DSR Data Set Ready
7	SG Signal Ground
8	NC No Connection
9	DTR Data Terminal Ready

**O/I Communication Cable 2C (FC4A-KC2C)**

**Cable Length: 5m (16.4 feet)**



**Mini DIN Connector Pinouts**

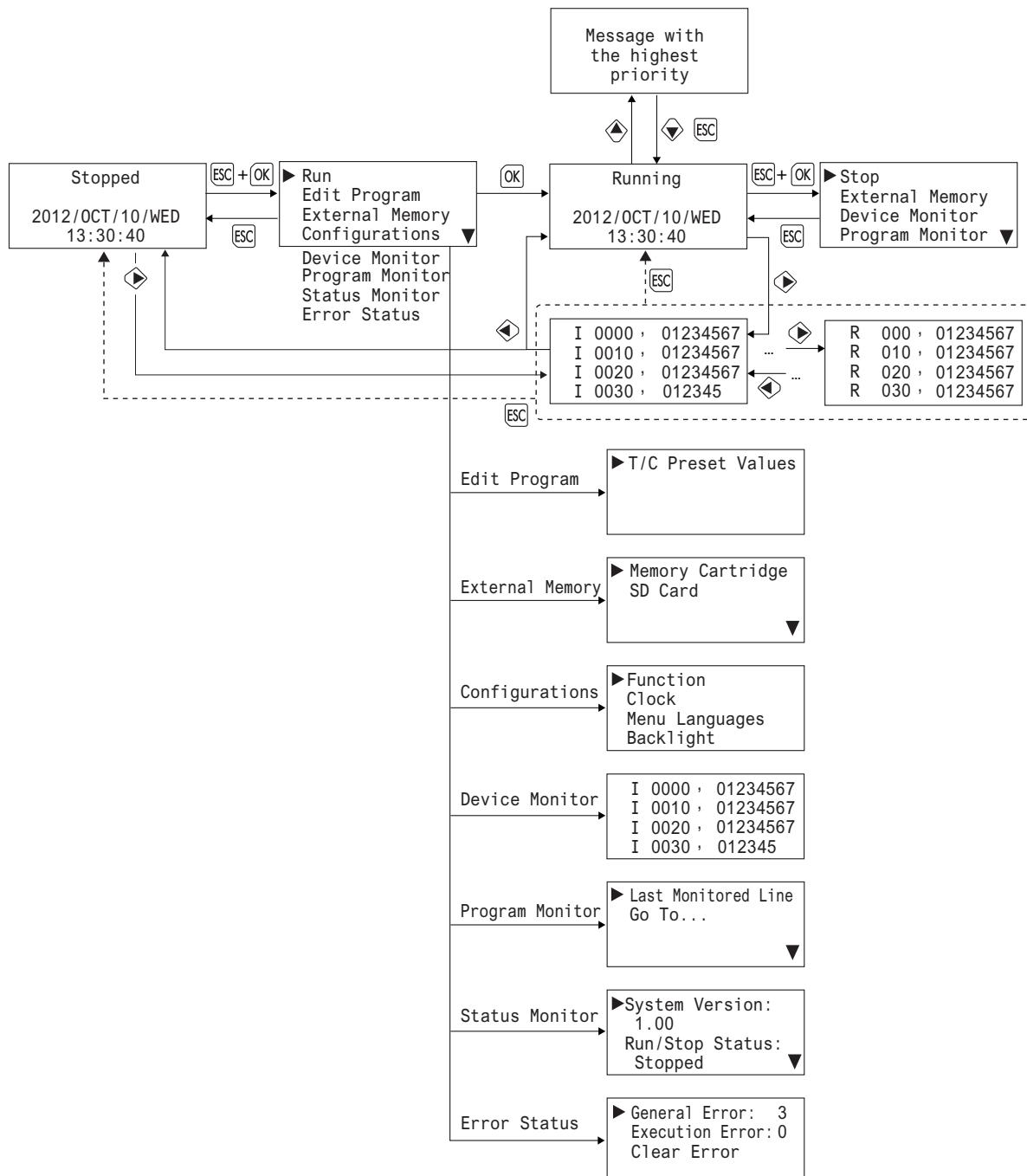
Description	Pin
NC	1
NC	2
TXD	3
RXD	4
NC	5
CMSW	6
SG	7
NC	8
Shield	Cover

**D-sub 25-pin Male Connector Pinouts**

Pin	Description
1	FG Frame Ground
2	TXD Transmit Data
3	RXD Receive Data
4	RTS Request to Send
5	CTS Clear to Send
6	DSR Data Set Ready
7	SG Signal Ground
8	DCD Data Carrier Detect
20	DTR Data Terminal Ready

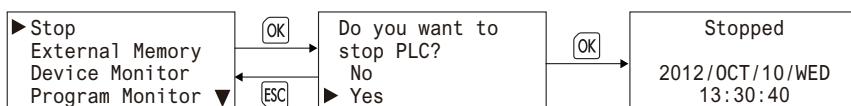
## HMI Screen Transition Diagram

### System menu overall transition diagram

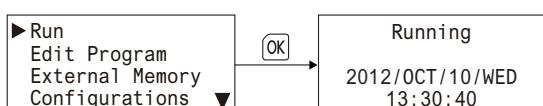


### Switching run/stop

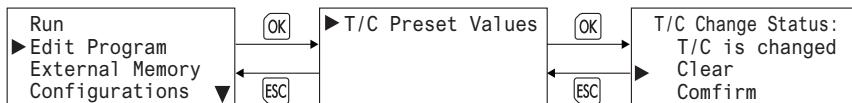
#### Running the SmartAXIS



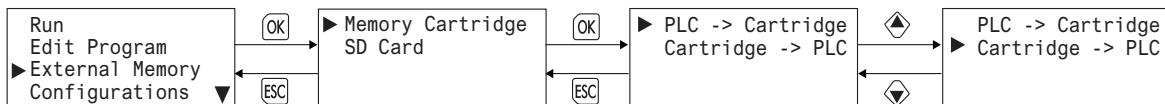
#### Stopping the SmartAXIS



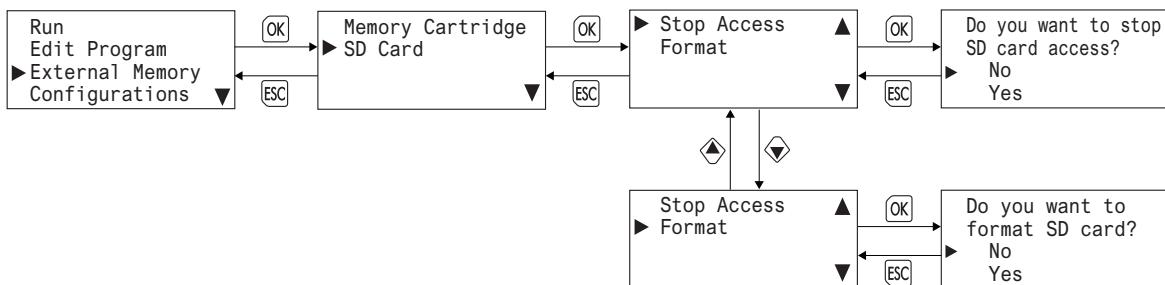
### Writing/Clearing TP (Timer Preset Values) and CP (Counter Preset Values) in the User Program



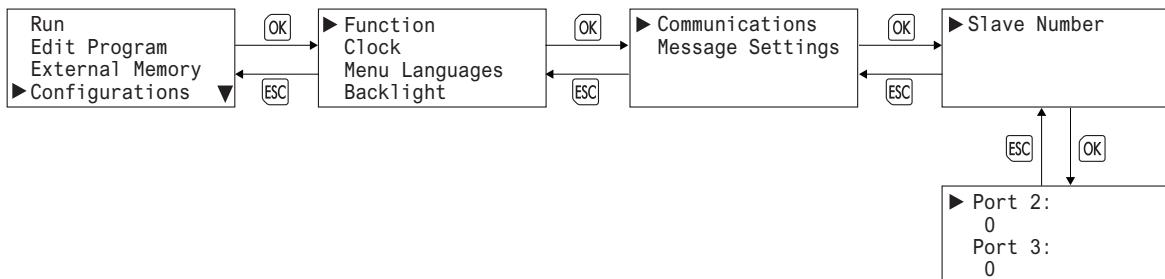
### Uploading/Downloading the User Program



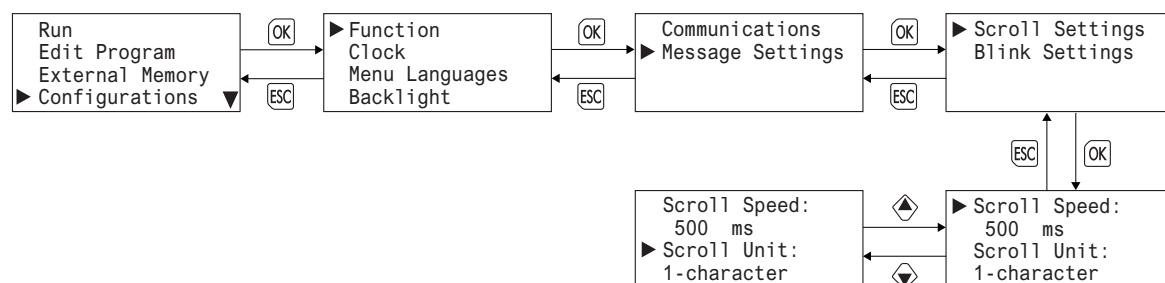
### Stopping Access to the SD Memory Card/Formatting the SD Memory Card



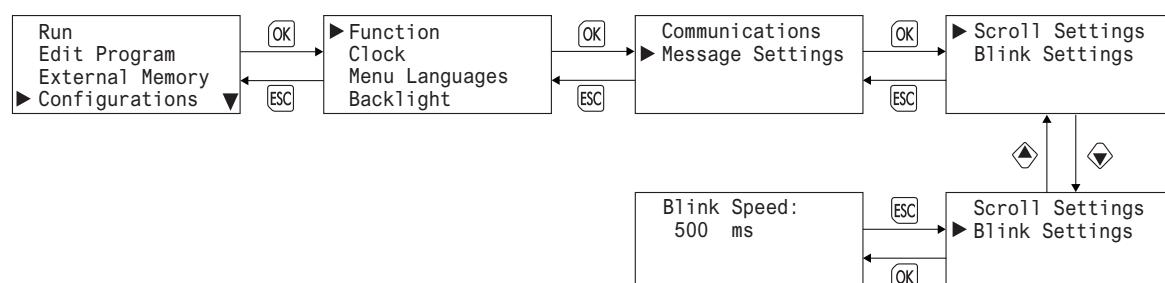
### Configuring the Slave Number



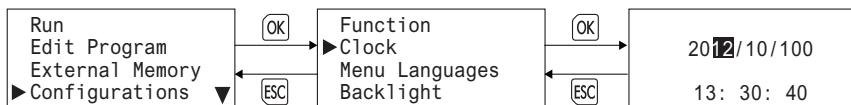
### Modifying the Scroll Speed/Configuring the Scroll Unit



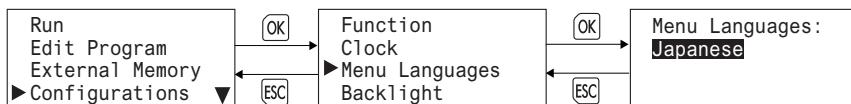
### Configuring the Blinking Speed



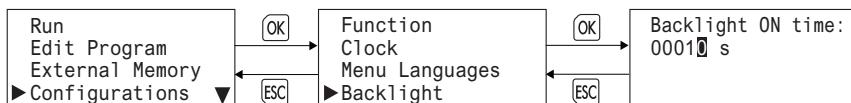
### Setting the Calender/Clock



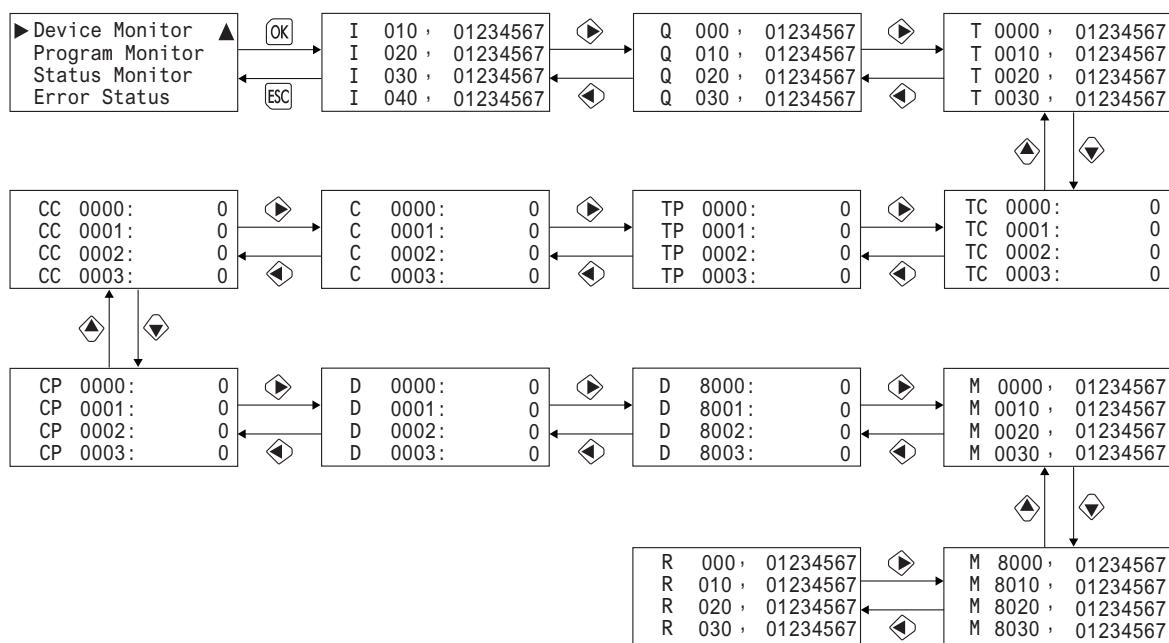
### Changing the Menu Language



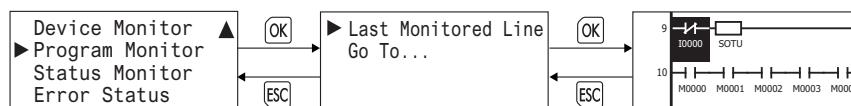
### Setting the LCD Backlight ON Time



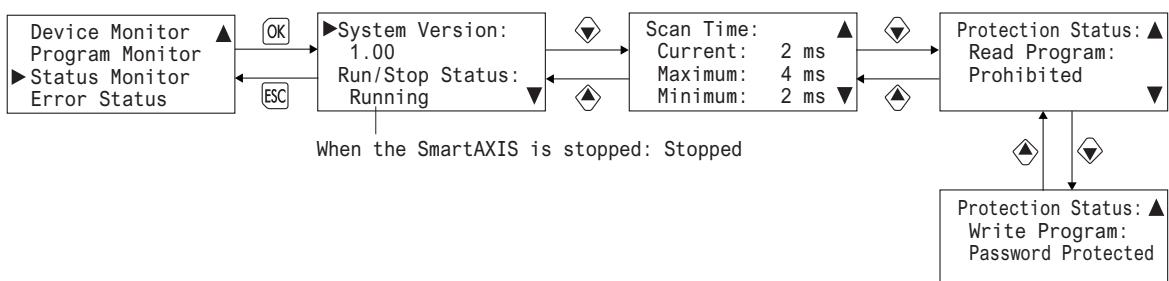
### Monitoring Device Values

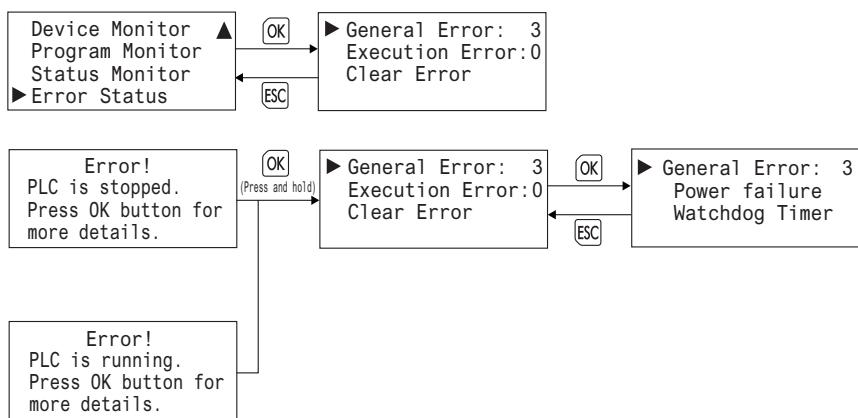


### Monitoring the User Program



### Monitoring the SmartAXIS Status



**Checking/clearing error information**

## Type List

### SmartAXIS Pro / Lite

Type	Power Voltage	I/O Points (Input Points/ Output Points)	Input Type		Output Type			Type No.
			Digital Input	Analog Input (Note)	Relay Output (10A)	Relay Output (2A)	Transistor Output	
Pro	24V DC	12 points (8/4)	6	2	4	—	—	FT1A-H12RA
		24 points (16/8)	12	4		4	—	FT1A-H24RA
		40 points (24/16)	18	6		8	4 (Sink) 4 (Source)	FT1A-H40RKA FT1A-H40RSA
		48 points (30/18)	22	8		—	18 (Sink) 18 (Source)	FT1A-H48KA FT1A-H48SA
	100-240V AC 50/60Hz	12 points (8/4)	8	—	—	—	—	FT1A-H12RC
		24 points (16/8)	16		4	4	—	FT1A-H24RC
		40 points (24/16)	24		—	12	—	FT1A-H40RC
		48 points (30/18)	30		—	—	18 (Sink) 18 (Source)	FT1A-H48KC FT1A-H48SC
Lite	24V DC	12 points (8/4)	6	2	4	—	—	FT1A-B12RA
		24 points (16/8)	12	4		4	—	FT1A-B24RA
		40 points (24/16)	18	6		8	4 (Sink) 4 (Source)	FT1A-B40RKA FT1A-B40RSA
		48 points (30/18)	22	8		—	18 (Sink) 18 (Source)	FT1A-B48KA FT1A-B48SA
	100-240V AC 50/60Hz	12 points (8/4)	8	—	—	—	—	FT1A-B12RC
		24 points (16/8)	16		4	4	—	FT1A-B24RC
		40 points (24/16)	24		—	12	—	FT1A-B40RC
		48 points (30/18)	30		—	—	18 (Sink) 18 (Source)	FT1A-B48KC FT1A-B48SC

**Note:** Shared with digital inputs

**Options**

Name	Description/specification		Type No.
SmartAXIS Pro/Lite User's Manual	Japanese	Describes product specifications, installation and wiring instructions, instructions for basic programming operations and special functions, device and instruction lists, communication functions, and troubleshooting procedures for the SmartAXIS Pro/Lite series.	FT9Y-B1377
	English		FT9Y-B1378
	Chinese (simplified)		FT9Y-B1379
	German		FT9Y-B1380
SmartAXIS Ladder Programming Manual	Japanese	Describes basic operations for ladder programming, instructions for monitoring ladders on the SmartAXIS, available devices and instruction lists, and details of each instruction.	FT9Y-B1381
	English		FT9Y-B1382
	Chinese (simplified)		FT9Y-B1383
	German		FT9Y-B1384
Application Software	Automation Organizer		SW1A-W1C
Communication Cartridge	Connection Specifications	Mini DIN	FT1A-PC1
	Standards	EIA RS232C	
	Maximum Baud Rate	115,200bps	
	Communication Functions	Maintenance communication, User communication, Modbus RTU master/slave	
	Isolation between Internal Circuit and Communication Port	Not isolated	
	Connection Specifications	Mini DIN	
	Standards	EIA RS485	
	Maximum Baud Rate	115,200bps	
	Communication Functions	Maintenance communication, User communication, Modbus RTU master/slave	
	Isolation between Internal Circuit and Communication Port	Not isolated	
End Clips	Connection Specifications	Terminal block	FT1A-PC2
	Standards	EIA RS485	
	Maximum Baud Rate	115,200bps	
	Communication Functions	Maintenance communication, User communication, Modbus RTU master/slave	
	Isolation between Internal Circuit and Communication Port	Not isolated	
Direct Mounting Hook	Used on DIN rail to fasten SmartAXIS modules (package quantity 10)	BNL6PN10	FT1A-PC3
	Direct mounting hook for 12- and 24-I/O types for maintenance, 5 pcs.	FT9Z-PSP1PN05	
	Capacity: 2GB	HG9Z-XMS2	
	Dedicated user program save memory	FT1A-PM1	
Maintenance Cable	USB maintenance cable USB A male to USB mini-B male, 2m	HG9Z-XCM42	HG9Z-XCM21
	USB extension cable (for panel attachment) USB mini-B female to USB mini-B male, 1m		

## Fonts

This section describes the fonts that can be displayed on the SmartAXIS Pro LCD.

## Supported Languages

The SmartAXIS Pro is equipped with fonts for multiple languages as standard. Messages in the language selected with the MSG instruction can be displayed on the LCD.

The menu can be displayed in English, Chinese, or Japanese, and messages can be displayed in a desired language out of the following languages: German, English, Italian, Spanish, Dutch, French, Cyrillic, Chinese, and Japanese.

## Installed Fonts List

Setting name	Character code system	Supported languages
European	ISO8859-1 (Latin-1)	English, German, Italian, Spanish, Dutch (Note), French (Note)
Japanese	Shift-JIS	Japanese (level 1)
Chinese	GB2312	Chinese (simplified)
Cyrillic	ANSI1251	Russian

**Note:** Some of the characters cannot be input.

## Character Code Table

### Using the Character Code Table

Example: Finding the character code for the character "a" in the table.

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			0	@	P		p				°	À	Ð	à	ð	
1		:	.	!	^	Q	a	q		i	±	Á	Ñ	á	ñ	
2		"	2	B	R	b	r			ø	Â	Ò	â	ò		
:		#	3	C	S	c	s			£	Ã	Ó	ã	ó		

Upper 4 bits of the code (hexadecimal)

Lower 4 bits of the code (hexadecimal)

The upper 4 bits of the code are hexadecimal 6.

The lower 4 bits of the code are hexadecimal 1.

Therefore, the character code for "a" is as follows.

"a": 61

Lower 4 bits

Upper 4 bits

**Note:** For other fonts and two-byte characters, refer to the table of the relevant code system.

**European Font (ISO 8859-1)**

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0			0	@	P	`	p				°	À	Ð	à	ã	
1		!	1	À	Q	a	q			i	±	Á	Ñ	á	ñ	
2		"	2	B	R	b	r			ø	²	Â	Ò	â	ò	
3		#	3	C	S	c	s			£	³	Ã	Ó	ã	ó	
4		\$	4	D	T	d	t			¤	¹	Ä	Ö	ä	ö	
5		%	5	E	U	e	u			¥	µ	Å	Õ	å	õ	
6		&	6	F	V	f	v			·	¶	Æ	Ö	æ	ö	
7		'	7	G	W	g	w			§	•	Ç	×	ç	÷	
8		(	8	H	X	h	x			“	,	È	Ø	è	ø	
9		)	9	I	Y	i	y			®	¹	É	Ù	é	ù	
A		*	:	J	Z	j	z			¤	º	Ê	Ó	ê	ú	
B		+	;	K	[	k	{			«	»	Ë	Ó	ë	ú	
C		,	<	L	\	l				¬	¼	Ì	Ü	ì	ü	
D		-	=	M	]	m	}				½	Í	Ý	í	ý	
E		.	>	N	^	n	~			®	¾	Ï	Þ	†	þ	
F		/	?	O	_	o				-	¸	Ï	Þ	†	þ	

**Cyrillic Font (ANSI 1251)**

	0	1	2	3	4	5	6	7	8	9	А	В	С	Д	Е	Ф
0			0	@	P	`	р	Ђ	Ђ		°	À	Ð	а	р	
1		!	1	À	Q	a	q	Ѓ	Ѓ	ў	±	Б	С	б	с	
2		"	2	B	R	b	r	,	,	ў	І	В	Т	в	т	
3		#	3	C	S	c	s	ƒ	”	Ј	і	Г	У	г	у	
4		\$	4	D	T	d	t	”	”	¤	ѓ	Д	Ф	д	ф	
5		%	5	E	U	e	u	...	...	Г	µ	Е	Х	е	х	
6		&	6	F	V	f	v	†	-	І	¶	Ж	Ц	ж	ц	
7		'	7	G	W	g	w	‡	-	§	•	З	Ч	з	ч	
8		(	8	H	X	h	x	€		Ё	ё	И	Ш	и	ш	
9		)	9	I	Y	i	y	%	™	®	ќ	Й	Щ	й	щ	
А		*	:	J	Z	j	z	љ	љ	Љ	€	е	К	њ	ќ	
В		+	;	K	[	k	{	њ	њ	Џ	»	Л	ы	л	ы	
С		,	<	L	\	l		Њ	Њ	Ћ	ј	М	њ	м	њ	
Д		-	=	M	]	m	}	Ќ	Ќ	-	Ѕ	Н	Э	н	э	
Е		.	>	N	^	n	~	Ћ	Ћ	®	ѕ	О	Ю	о	ю	
Ф		/	?	O	_	o		Џ	Џ	Ћ	†	П	ј	п	ј	

**Japanese Font (JIS X0201)**

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0				0	@	P	`	p			-	タ	ミ			
1			!	1	Â	Q	a	q		.	ア	チ	ム			
2			"	2	B	R	b	r		「	イ	ツ	メ			
3			#	3	C	S	c	s		」	ウ	テ	モ			
4			\$	4	D	T	d	t		、	エ	ト	ヤ			
5			%	5	E	U	e	u		・	オ	ナ	ユ			
6			&	6	F	V	f	v		ヲ	カ	ニ	ヨ			
7			,	7	G	W	g	w		ア	キ	ヌ	ラ			
8			(	8	H	X	h	x		イ	ク	ネ	リ			
9			)	9	I	Y	i	y		ウ	ケ	ノ	ル			
A			*	:	J	Z	j	z		エ	コ	ハ	レ			
B			+	;	K	[	k	{		オ	サ	ヒ	ロ			
C			,	<	L	¥	l			ヤ	シ	フ	ワ			
D			-	=	M	]	m	}		ュ	ス	ヘ	ン			
E			.	>	N	^	n	~		ヨ	セ	ホ	。			
F			/	?	O	_	o			ツ	ソ	マ	。			



# INDEX

---

---

- # 1:1 Maintenance Communication System 1-7  
1:1 maintenance communication system 1-7  
1:N Maintenance Communication System 1-7  
100-ms  
    clock M8122 7-7  
10-ms  
    clock M8123 7-7  
1-sec  
    clock  
        reset M8001 7-5  
2-edge count 5-12  
4-edge count 5-12
- A** AC Power Type 3-8, 3-11  
ADD-2comp 10-28  
Adding counter 5-12  
address  
    map 11-9, 11-24  
Adjust  
    function 5-65  
advanced instruction  
    applicable CPU modules 8-6  
    list 8-3  
all outputs OFF M8002 7-5  
Allow Access by IP Address 9-7, 11-26  
Analog  
    Input  
        Allocation 5-35  
        Filter 5-35  
ASCII  
    character code table 10-23
- B** Backlight 12-9  
basic  
    instructions 8-1  
Baud Rate (bps) 9-4  
BMOV/WSFT executing flag M8024 7-6  
busy  
    control 10-12  
    signal 10-24
- C** Cable  
    Computer Link  
        4C A-8  
    O/I Communication A-9  
        1C A-10  
        2C A-10  
cable 10-2, 10-3, 10-24, A-7, A-9  
    computer link 4C A-8  
    O/I communication  
        1C A-9  
        user communication 1C A-9  
Calendar & Clock 5-3  
Calendar and clock  
    data storage locations 5-66  
calendar data  
    write flag M8016 7-5  
Calendar/clock
- error 5-65  
calendar/clock  
data  
    read error flag M8014 7-5  
    write flag M8020 7-5  
    write/adjust error flag M8013 7-5  
carry  
    (Cy) and borrow (Bw) M8003 7-5  
catch input  
    ON/OFF status M8154-M8157 7-6  
Character Code Table A-17  
Checking the Version A-1  
clearing  
    error  
        codes 12-2  
client  
    user communication 10-15  
client connection  
    connected IP address 5-69  
Clock  
    Function 5-65  
clock  
data  
    adjust flag M8021 5-65, 7-5  
    write flag M8017 7-5  
Communication  
    Cartridge 2-23  
    Functions 1-6  
    Ports 1-6, 5-42  
communication  
    adapter information D8030 7-12  
    connector cover removing 3-3  
error  
    M8005 7-5  
format 11-12  
mode information (port 1 through 7) D8026 7-11  
parameters 10-25, 10-26, 11-5, 11-10  
refresh  
    port 3 through port 7 5-58  
settings 11-5  
Communication Ports and Communication Methods 5-42  
Comparison  
    Actions 5-13  
    ON status 5-19  
comparison  
result  
    1 M8151 7-7  
    2 M8150 7-7  
    3 M8152 7-7  
Comparison output reset 5-20  
computer link  
    cable 4C A-8  
confirm  
    password 5-61  
connected IP address  
    client connection 5-69  
Connection Status 5-69  
connector pinout 10-2, 10-3, 10-24, A-8, A-9, A-10

---

Constant Scan Time 5-63  
 control  
     signal  
         option DSR D8105/D8205 10-12  
         option DTR D8106/D8206 10-13  
         status D8104/D8204 10-11  
 Count direction flag 5-21  
 counter  
     high-speed 5-28  
     keep designation 5-7  
 CPU module  
     type information D8002 7-11  
 CRC-16 10-28, 11-11  
 Crimping Tool 3-13  
 CSV files 5-48  
 Current value  
     storage locations 5-19  
 cyclic redundancy checksum 10-28, 11-11

**D**

data  
     storage setting  
         from lower word 5-58  
         from upper word 5-58  
 Data Bits 9-4  
 data link  
     communication  
         prohibit flag M8006 7-5  
         connection error 12-4  
 data register  
     keep designation 5-7  
 Daylight Savings Time 5-64  
 DC Power Type 3-7, 3-11  
 default gateway D8312-D8315, D8338-D8341 7-13  
 Derating 2-13  
 details button 12-1  
 device  
     addresses 7-1  
 Device Settings 5-3  
 DHCP 5-68  
 Dimensions 2-30  
 DIN rail 3-3  
 direct  
     mounting  
         on panel surface 3-4  
 disable  
     and enable interrupts 5-30, 5-37  
 disabling protection 5-61  
 Download  
     user program 9-1  
 download  
     program 4-7  
 Downloading and uploading user programs 5-45  
 DSR  
     control signal status 10-11  
     input control signal option 7-14  
         D8105/D8205 10-12  
 DTR  
     control signal status 10-11  
     output control signal option 7-14  
         D8106/D8206 10-13  
 D-type (type 3) ground 3-7

**E**

edit user program 4-4

ERR LED  
     during errors 12-4  
 Error  
     Status LED 2-3, 12-11  
 error  
     causes and actions 12-4  
     code  
         user communication 10-22  
         user program execution 12-6  
         status box 12-1  
 ERXD 10-16  
 Ethernet 1-9  
     Communication 1-9  
     Port 2-2, 2-27  
     port 5-42  
     Status LED 2-2  
     use-defined communication 2-27  
 ethernet  
     user communication 5-71  
 ETXD 10-16  
 Expansion  
     Communication  
         Port 2-2  
             Wiring Diagram 5-43  
         communication  
             ports 5-42  
     expansion  
         communication  
             port 2-23  
                 cover 2-24  
                 port cover 2-23  
     data register  
         data writing flag M8026 7-6  
         data writing flag M8027 7-6  
     external input 5-32

**F**

falling edge of catch input 5-29  
 Ferrules 3-13  
 filter  
     input 5-58  
 Fonts A-17  
 forced  
     I/O 5-39  
     I/O status 5-39  
 Frequency Measurement 5-32  
 from  
     lower word 5-58  
     upper word 5-58  
 Function  
     Area Settings 5-2  
     List 5-1  
     Specifications 2-6  
 function  
     code 11-6  
     function code 11-24

**G**

Genera?  
     Specifications 2-4  
         DC Power Type 2-5  
 general  
     specifications  
         AC power type 2-4

- 
- H**
- High-Speed Counter 5-9
  - High-speed Counter
    - External Inputs 5-10
  - high-speed counter 5-28
    - comparison output reset M8030, M8034, M8040, M8044 7-6
    - gate input M8031, M8035, M8041, M8045 7-6
    - reset input M8032, M8036, M8042, M8046 7-6
  - High-speed counter devices 5-18
  - host name 5-71
  - HSC
    - reset input 5-21
- I**
- I/O
    - forced 5-39
    - status 5-39
  - in-operation output M8125 7-7
  - Input
    - Internal Circuit 2-9, 2-12
    - Operating Range 2-9, 2-12
    - Specifications
      - AC Power Type 2-8
      - DC Power Type 2-10
    - Terminal
      - Wiring 3-7
    - Terminals 2-1
  - input
    - filter 5-58
  - Input Configuration 5-2
  - installation
    - and wiring 3-1
    - location 3-1
  - Installed Fonts List A-17
  - instruction
    - ERXD 10-16
    - ETXD 10-16
  - instructions
    - user communication 10-1
  - internal
    - relay
      - keep designation 5-7
  - interrupt
    - input 5-30
      - I2 through I5 edge M8192?M8195 7-6
      - status M8140-M8143 7-6
  - IP address 5-71
  - IP address D8304-D8307, D8330-D8333 7-13
  - IP Addresses 9-7
- K**
- keep
    - data sum check Error 12-4
    - designation 5-7
  - key matrix
    - input 5-39
- L**
- LCD 2-2
    - Specifications 2-8
  - line
    - control signals RS232C 10-11
  - list
    - advanced instruction 8-3
    - basic instruction 8-1
    - type A-15
- M**
- Local Host Port No. 9-7, 11-26
  - longitudinal redundancy check 10-28
  - LRC 10-28
  - MAC address D8324-D8329 7-13
  - maintain outputs while CPU stopped M8025 7-6
  - maintaining catch input 5-29
  - MAINTENANCE
    - COMMUNICATION 9-1
    - Maintenance
      - Communication 1-7
        - via Ethernet Port 9-5
        - via Expansion Communication Port 9-3
        - via USB Port 9-2
      - maintenance
        - communication 9-1
    - Memory
      - Cartridge 2-25, 5-45
    - memory
      - backup error run/stop selection 5-5
      - cartridge
        - cover 2-25, 2-26
        - information D8003 7-11
        - cartridge program transfer error 12-5
    - Memory Backup 5-2
    - Modbus
      - ASCII 10-28
        - communication 11-1
      - Communication 1-8
        - via Ethernet Communication 11-18
      - communication
        - via RS-232C/RS-485 11-1
      - Communication via RS-232C/RS-485 11-1
      - master request table 11-4
      - RTU 10-28, 11-11
    - Modbus TCP
      - server specifications 11-24
    - modbus TCP
      - client 5-71, 11-19
      - client specifications 11-19
      - communication error 11-19
      - communication format 11-26
      - error status 11-23
      - function code 11-22
      - remote host number 11-23
      - request execution device 11-23
      - server 11-24
    - monitor
      - operation 4-8
    - Monitor/change
      - device values 9-1
    - monitoring
      - WindLDR 12-1
    - Mounting
      - Space 3-2
    - mounting
      - hole layout
        - for direct mounting 3-5
        - on DIN rail 3-3
        - on panel surface 3-4
    - N**
    - network settings 5-68

---

<b>O</b>	O/I communication cable 1C A-9 online edit 5-68 operating status during errors 12-4 Operation Buttons 2-2 operation basics 4-1 Operator Interface Connectivity 1-10 optional cartridge information D8031 7-12 Options 1-2, A-16 Output Delay 2-13, 2-14 Internal Circuit 2-15 Specifications 10A Relay 2-13 2A Relay 2-14 Transistor 2-15 Terminals 2-2 output during errors 12-4 Overflow 5-20	<b>Q</b>	quit WindLDR 4-8
		<b>R</b>	read program 5-61 reading error data 12-1 receive timeout 10-4 Receive Timeout (ms) 9-4, 9-7 Recommended SD Memory Card 2-28 remote host list 5-71 remote host number modbus TCP 11-23 Remote I/O 1-9 removing communication connector cover 3-3 terminal block 3-3 request table 11-5 reset input 4-10, 5-4, 5-20 status 5-21 Reset input 5-20 rising edge of catch input 5-29 rising/falling edge selection 5-28, 5-30 RS232C control signal status 7-13 DSR input control signal option 7-14 DTR output control signal option 7-14 line control signals 10-11 User Communication 1-8 RS232C/RS485 communication 1-6 RS485 Modbus Communication 1-8 RUN mode control signal status 10-11 Run/Stop Control 5-2 run/stop selection at memory backup error 5-5 at power up 5-6 rung 4-3, 4-4
<b>P</b>	Parity 9-4 password 5-61 Password protected 5-60 pinout 10-2, 10-3, 10-24, A-8, A-9, A-10 PLC Selection 4-2 PLC status 12-1, 12-2 port number 5-71 power failure 12-4 supply 3-11 wiring 3-12 Power Supply Terminals 2-1 Power/Run Status LE 2-3 Status LED 12-8, 12-10 Preset value storage locations 5-13 Preset value, reset value storage locations 5-20 programming 32-bit data storage setting using WindLDR 5-59 catch input using WindLDR 5-28 forced I/O using WindLDR 5-40 input filter using WindLDR 5-34 interrupt input using WindLDR 5-30 Modbus master using WindLDR 11-4 Modbus slave using WindLDR 11-10 RXD instruction using WindLDR 10-8 special data register 10-24 timer interrupt using WindLDR 5-37 TXD instruction using WindLDR 10-5 user communication using WindLDR 10-4 user program protection using WindLDR 5-60 Prohibited 5-60 protection user program 5-58	<b>S</b>	?ave destination for log data 5-51 Save Log Data 5-48 SD Memory Card 2-28, 5-48 Access Stop Flag 7-6 Capacity 7-12 Slot 2-2 Special Internal Relays 5-52 Status LED 2-2, 5-52 SD Memory Card Maint 5-49 Self Diagnostic 5-3 sensor power terminals 2-3 server connection (1 through 8) connected IP address D8362-D8393 7-14 user communication 10-19 Setting Clock 5-65, 5-66 shift register

- 
- keep designation 5-7  
 simulate operation 4-7  
 Single-phase high-speed counter 5-10, 5-11  
 Slave Number 9-4  
 special  
     functions 5-1  
 Special data register  
     list 5-19  
 special data register 7-8  
     for error information 12-3  
     for interrupt inputs 5-30  
     for RS232C line control signals 10-11  
     for scan time 5-63  
     for timer interrupt 5-37  
 Special internal relay  
     list 5-19  
 special internal relay 7-2  
     for catch inputs 5-28  
     for interrupt inputs 5-30  
     for timer interrupt 5-37  
 specifications  
     catch input 5-28  
     Modbus master communication 11-2  
     user communication mode 10-1  
 start  
     control M8000 7-5  
     WindLDR 4-1, 4-3  
 Start/stop  
     high-speed counter 5-19  
 start/stop  
     operation 4-9  
     schematic 4-9  
     using power supply 4-10  
     using WindLDR 4-9  
 Starting the Data File Manager 5-49  
 status  
     system 4-10  
 stop  
     input 4-10, 5-4  
 Stop Bits 9-4  
 STOP mode control signal status 10-11  
 subnet mask D8308-D8311, D8334-D8337 7-13  
 Support Languages 1-3  
 Supported Languages A-17  
 system  
     program  
         version D8029 7-12, 12-1  
     setup  
         ID quantity of inputs D8000 7-11  
         ID quantity of outputs D8001 7-11  
         RS232C user communication 10-2  
         RS485 user communication 10-3  
 software  
     upgrade A-1  
 statuses at stop, reset, and restart 4-10
- T**
- table ASCII character code 10-23  
 Terminal  
     Arrangement 2-16  
 terminal  
     block removing 3-3  
     connection 3-13  
 Terminals for Terminal Blocks 3-13
- U**
- terminals for terminal blocks 3-13  
 timer  
     interrupt  
         status M8144 7-7  
 timer/counter  
     preset value  
         sum check error 12-4  
 troubleshooting 12-1  
     diagrams 12-7  
 Two-phase high-speed counter 5-10, 5-11  
 Type  
     List 1-2  
 type  
     list A-15
- V**
- Underflow 5-20  
 Up/down selection reversible counter 5-12  
 upgrade SmartAXIS system software A-1  
 Upload  
     user programs 9-1  
 USB 4-7  
     Port 2-2  
     port 5-42  
 USB Extension Cable 3-14  
 USB Port  
     Cover 2-2  
 User  
     Communication 1-8  
     Program Passwords 5-47  
 user  
     communication  
         cable 1C A-9  
         error 10-11  
             code 10-22  
         instructions 10-1  
         receive instruction cancel flag  
             port 1 M8022 7-5  
             port 2 M8023 7-5  
             port 7 M8170 7-7  
             ports 4-6 M8145-M8147 7-6  
     system  
         setup RS232C 10-2  
         setup RS485 10-3  
 program  
     EEPROM sum check error 12-4  
     execution error 12-6  
         M8004 7-5  
     protection 5-58  
     RAM sum check error 12-4  
     syntax error 12-4  
     writing error 12-4  
     protocol 10-4  
 User Communication  
     via Ethernet Communication 10-14  
     via Serial Communication 10-1  
 user communication  
     cancel flag 10-16, 10-19  
     client 10-15  
     server 10-19
- V**
- version  
     system program D8029 7-12  
     via USB 12-24

---

**W** watchdog timer error 12-4

WindLDR

  clearing error codes 12-2

  monitoring 12-1

  programming

    catch input 5-28

    input filter 5-34

    interrupt input 5-30

  Modbus

    master 11-4

    slave 11-10

  RXD instruction 10-8

  timer interrupt 5-37

  TXD instruction 10-5

  user

    communication 10-4

    program protection 5-60

quit 4-8

start 4-1, 4-3

wiring 3-1

  power supply 3-12

write program 5-61