
Communications Manual

Model: TTX-700
 Designation: Module Controller

APPROVED	APPROVED	CHECKED	DESIGNED	DATE March 16, 2004	Job. No. G2745	TITLE Module Controller	
				MODEL TTX-700		DWG. No.	PAGE
				TOHO ELECTRONICS INC.		45-5488-A	01/39

Contents

1.	Before using the product.....	4
1.1	On this operation manual.....	4
1.2	What can be done with communications	4
1.3	Positioning communications (priority ranking).....	4
1.4	Setting before communications.....	4
2.	Settings regarding TOHO communications.....	5
2.1	Overview.....	5
2.2	Setting a data length.....	6
2.3	Setting a stop bit length.....	6
2.4	Setting a parity.....	6
2.5	Setting whether to conduct a BCC check.....	6
2.6	Setting a communications speed.....	6
2.7	Setting an address.....	7
2.8	Setting a response delay	7
2.9	Switching communications mode	7
3.	TOHO communications control.....	8
3.1	Communications procedure.....	8
3.2	Message types.....	8
3.3	Composition of a request message (transmitted from a high-level computer to this product).....	9
3.4	Composition of a response message (transmitted from this product to a high-level computer).....	10
3.5	Description of codes	11
3.6	Communications precautions	13
4.	Examples of TOHO communications.....	15
4.1	Examples of communications to be read.....	15
4.2	Examples of communications to be written	16
5.	Settings regarding MODBUS communications	17
5.1	Overview.....	17
5.2	Setting a data length.....	18
5.3	Setting a stop bit length.....	18
5.4	Setting a parity.....	18
5.5	Setting a communications speed.....	18
5.6	Setting an address.....	19
5.7	Setting a response delay	19
5.8	Switching communications mode	19

6	MODBUS communications control.....	20
6.1	Communications procedure.....	20
6.2	Message types.....	20
6.3	Composition of an RTU request message (transmitted from a high-level computer to this product).....	21
6.4	Composition of an RTU response message (transmitted from this product to a high-level computer).....	22
6.5	Description of RTU codes.....	23
6.6	Precautions on RTU communications.....	25
6.7	Example of CRC-16 calculations.....	26
6.8	Composition of an ASCII request message (transmitted from a high-level computer to this product).....	27
6.9	Composition of ASCII response messages (transmitted from this product to a high-level computer).....	28
6.10	Description of ASCII codes.....	30
6.11	Precautions on ASCII communications.....	32
6.12	Example of LRC calculations.....	33
7.	Specifications.....	34
7.1	Communications standard category: Compliant with EIA standard RS-485.....	34
7.2	Communications specifications.....	34
8.	Connections.....	35
8.1	Connections for the RS-485.....	35
9.	Table of ASCII codes.....	36
10.	Table of identifiers (codes).....	37

1. Before using the product

1.1 On this operation manual

This is an operation manual regarding communications with a TTX-700 (hereinafter referred to as "this product").

1.2 What can be done with communications

With this product, users can write and read items specified in "10. Table of identifiers," such as "reconfiguring, starting, or stopping items that are operable with the front keys" and "reading information displayable on the display."

However, reading and writing with ordinary commands are performed with regarding to the RAM in this product. Written data can be turned back into the values before the writing (the values stored on the EEPROM) by turning power off and on again. To store the written data on the EEPROM of this product, execute a store request message. (See "3.6. Communications precautions.")

Settings regarding options not added and other unnecessary settings cannot be read or written.

1.3 Positioning communications (priority ranking)

Data and parameters in this product can be changed with keys while in operation in the communications mode.

While this product is in operation in the RO (read-only) mode, no data or parameter setting can be changed by communications. (Provided that communications modes can be changed.)

1.4 Setting before communications

Before performing communications, this product must be set.

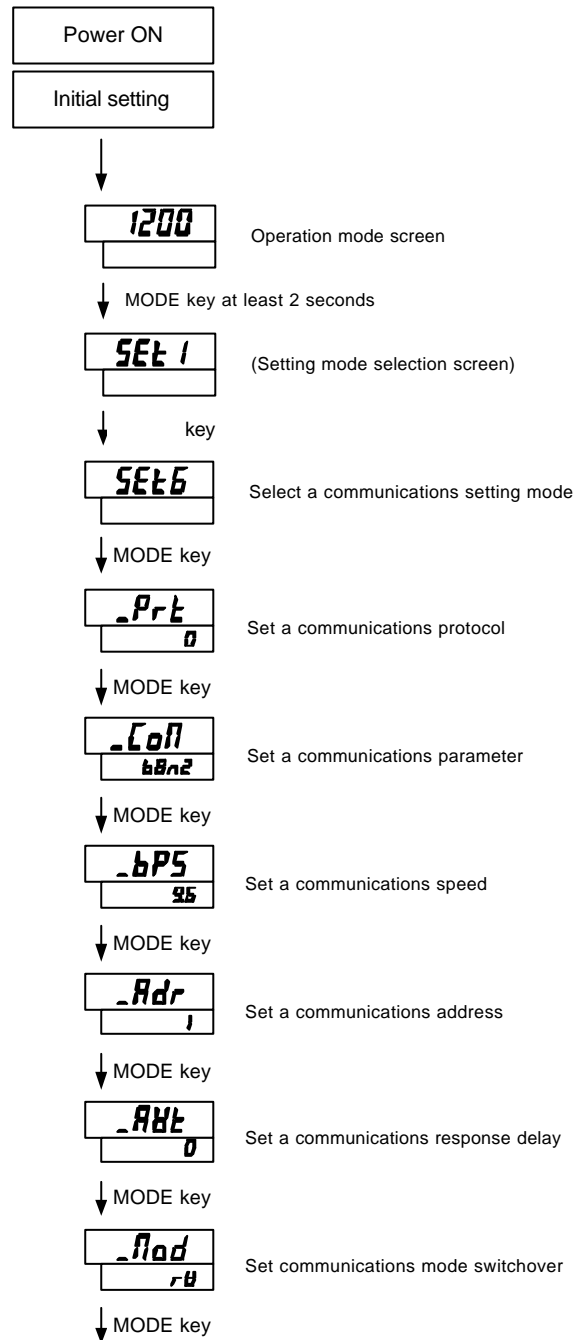
This product is compatible with the TOHO communications protocol and the MODBUS communications (RTU, ASCII).

Select a protocol with the protocol setting (~~Prt~~) on communications 1/2 setting (~~SEt5~~). For the TOHO communications protocol, see "2. Settings regarding TOHO communications." For the MODBUS communications protocol, see "5. Settings regarding MODBUS communications."

2. Settings regarding TOHO communications

2.1 Overview

Before communications is performed, initial settings must be made on this product. Enter such settings with the keys on the front panel. To switch to a series of setting screens, take the steps described below. For details, see the operation manual furnished with this product.



When the settings are over, press the MODE key at least 2 seconds to go back to the operation mode. The parameters indicated above are initial values.

	DWG.No. 45-5488-A	PAGE 06/39
--	----------------------	---------------

2.2 Setting a data length

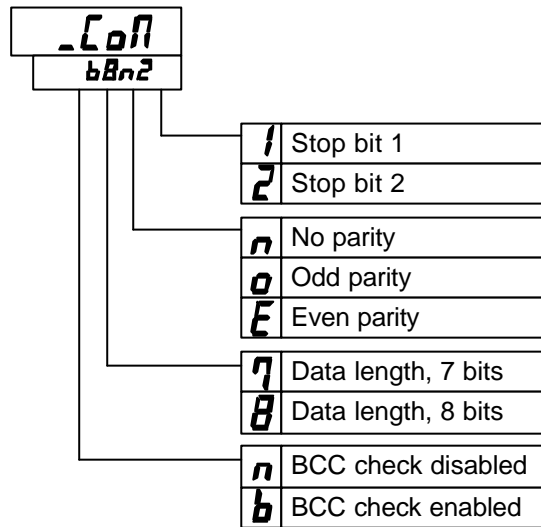
2.3 Setting a stop bit length

2.4 Setting a parity

2.5 Setting whether to conduct a BCC check

While in the "Set a communications parameter" screen on the preceding page, operate the **←** and **→** keys to make the settings.

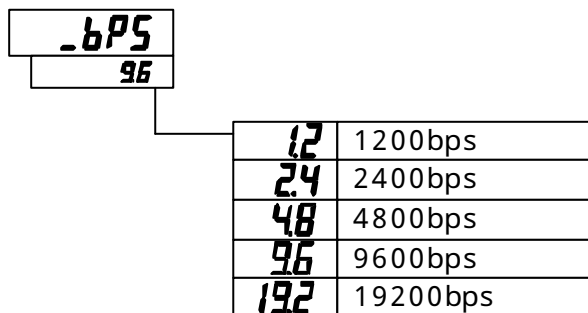
The initial value is [**b8n2**].



2.6 Setting a communications speed

While in the "Set a communications speed" screen on the preceding page, operate the **←** and **→** keys to make the settings.

The initial value is [**96**].

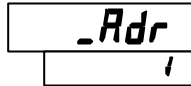


	DWG.No. 45-5488-A	PAGE 07/39
--	----------------------	---------------

2.7 Setting an address

While in the "Set a communications address" screen on the preceding page, operate the and keys to make the settings.

The initial value is 0.



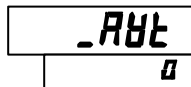
Setting range: 1 to 99 stations (It cannot be set to a 0.)

2.8 Setting a response delay

Set a time from the time when the high-level computer finished sending a "request message" until the time when it delivers the line and enters an input state.

While in the "Set a response delay" on the preceding page, operate the and keys to make the settings.

The initial value is 0.

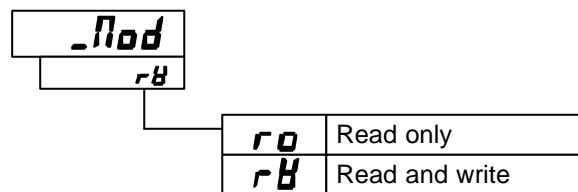


Setting range: 0 to 250ms

- * If the response delay is set to a short setting, the communications may not be conducted normally.
- * In a real operation, the processing time for this product will be added, in addition to the response delay.

2.9 Switching communications mode

While in the "Set communications mode switchover" screen on the preceding page, operate the and keys to make the settings.

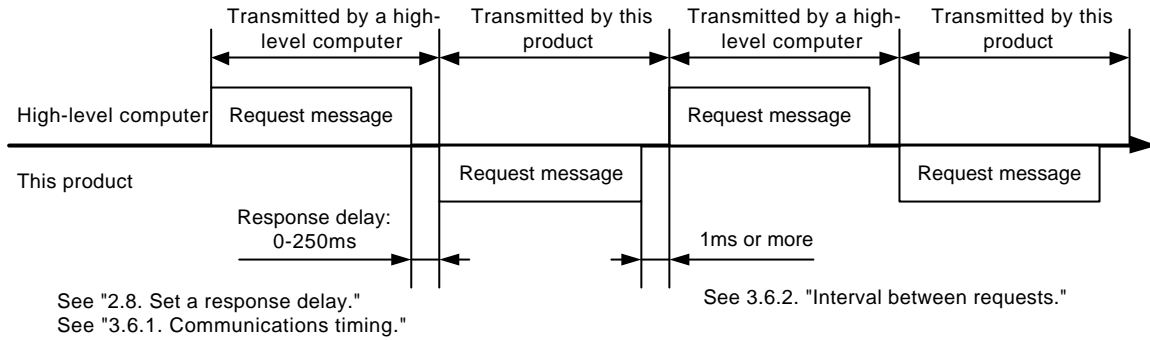


	DWG.No. 45-5488-A	PAGE 08/39
--	----------------------	---------------

3. TOHO communications control

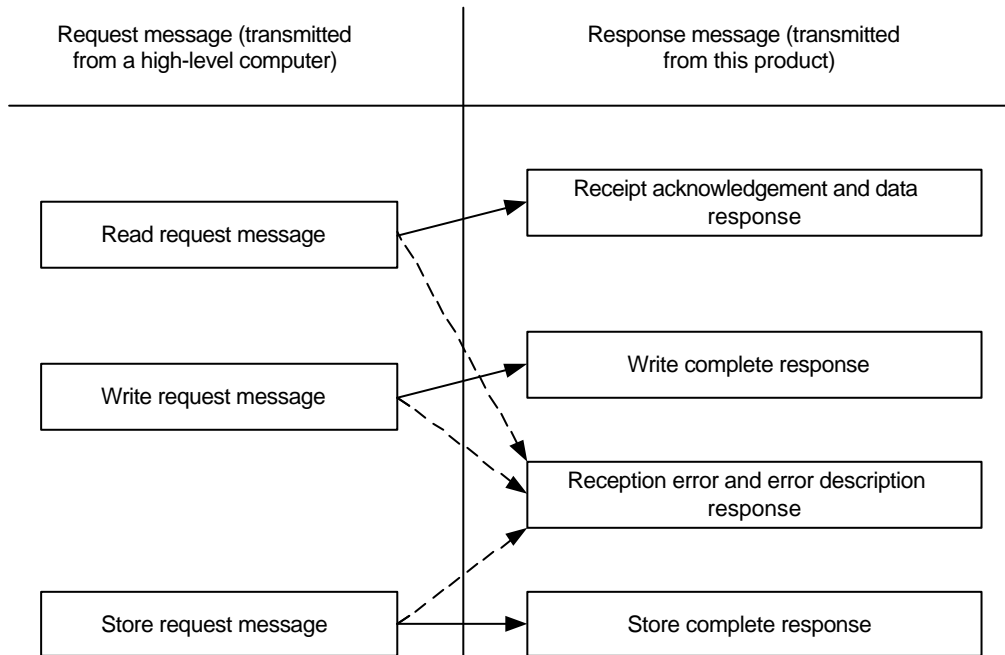
3.1 Communications procedure

This product returns a "response message" in response to a "request message" from a high-level computer. It therefore does not initiate a transmission.



3.2 Message types

- Messages are roughly divided into the following types:



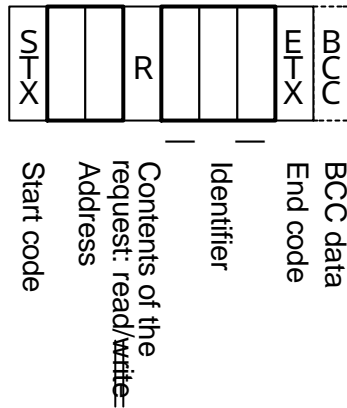
- All codes (except for BCC) from STX data to ETX are expressed in ASCII codes.
- In assembling a program for a high-level computer, see "10. Table of identifiers (codes)" and "9. Table of ASCII codes" at the end of the book.

	DWG.No. 45-5488-A	PAGE 09/39
--	----------------------	---------------

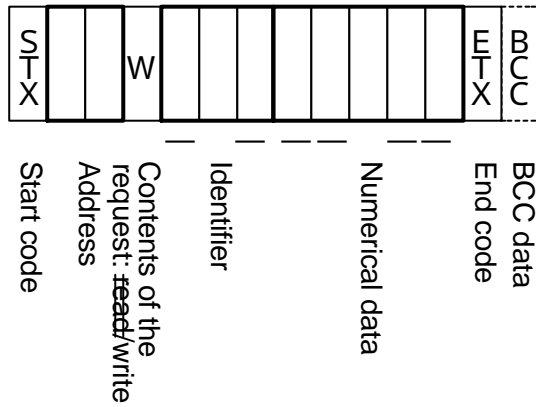
3.3 Composition of a request message (transmitted from a high-level computer to this product)

- For codes to , see "3.5. Description of codes."
- For specific examples of request messages, see "4.1. Examples of reading communications" and "4.2. Examples of writing communications."

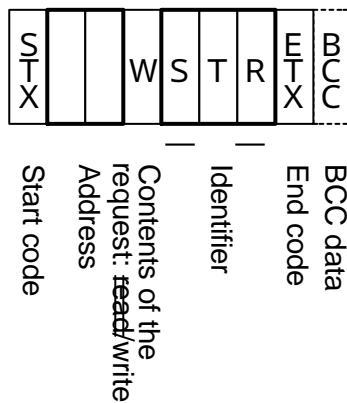
3.3.1 Composition of a read request message



3.3.2 Composition of a write request message



3.3.3 Composition of a store request message

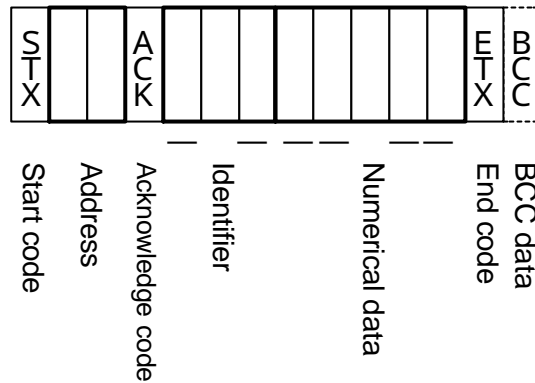


	DWG.No. 45-5488-A	PAGE 10/39
--	----------------------	---------------

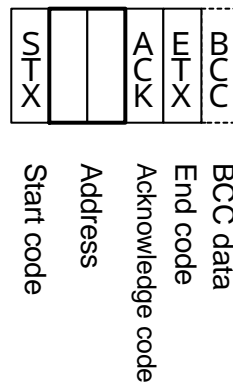
3.4 Composition of a response message (transmitted from this product to a high-level computer)

- For codes through , see "3.5. Description of codes."
- For specific examples of request messages, see "4.1. Examples of communications to be read" and "4.2. Examples of communications to be written."

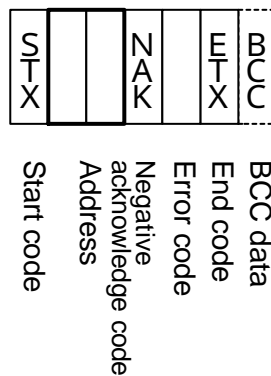
3.4.1 Response message in response to a read request message



3.4.2 Response message in response to a write/store request message



3.4.3 Response message in the case of an error



3.5 Description of codes

- The codes from STX, address to ERR as indicated below are expressed in ASCII codes.
- For the ASCII codes, see "9. Table of ASCII codes."
- For conversion to ASCII codes, see "4. Examples of TOHO communications."

STX

This code is needed for the receiver to detect the top of the message. It is affixed to the top of a character string to be sent.

Address

This is the address of the party (this product) with whom a high-level computer communicates. The address in the response message from this product indicates the sender of the response message. Note that, when CH2 is used, two addresses are occupied. (Setting ADR to 1 causes this product to occupy addresses 1 and 2.)

Contents requested

Enter a code R or W.

R: to read data from this product

W: to write or store data in this product

Identifier

An identifier is a classification code (identifier) for data to be read or written and expressed in a three-digit alphanumerical ASCII code. See "10. Table of identifiers (codes)."

Numerical data

At the time of writing or reading, 5-digit numerical data can be written.

Example: The table below indicates the significances of 5-digit numerical data 00010.

Example	Significance of the value
Proportional band (P)	→1.0%
Data (PV), etc, whose decimal point can be shifted	
When the decimal point setting (DP) is 0	10
When the decimal point setting (DP) is 0.1	1.0

If DP = 0.1, the numerical data "12000" means 1200.0. In the case of text data, it is " INP." (The is a space.)

ETX

This code is needed for the receiver to detect the end of a message. It is affixed to the end of a character string to be sent (except for BCC).

BCC

This is a check code for error detection and is the exclusive OR (EX-OR) of all characters from STX to ETX.

If the BCC check is set to "Disabled" in the communications settings in this product, this code (BCC) will not be incorporated in the response message. See "2. Settings regarding TOHO communications."

ACK

It is an acknowledge code. If a message received by this product is error-free, this code will be incorporated in the "response message" from this product and returned.

NAK

It is a negative acknowledge code. If a "request message" received by this product is error-ridden, this code will be incorporated in the "response message" from this product and returned.

If the "request message" received is error-ridden, the error contents (ERR type) will be incorporated in the "response message" from this product, following NAK.

ERR type

If a "request message" received from this product is error-ridden, the error contents (either of the numbers in the table below) will be incorporated in the "response message" from this product, following " NAK."

The error number 0 is an instrument error (memory error or A/D conversion error). It will be incorporated in the "response message" regardless of whether there is an error in the "request message."

The error number 9 is an AT error. It will therefore be incorporated in the "response message" regardless of whether there is an error in the "request message." Remove the cause of the error immediately and start the AT again.

If there are two or more errors occurring at the same time, the largest error number will be incorporated.

The table below indicates the error contents and classifications.

Error No.	Error contents in the "request message" received by this product
0	Instrument error (memory error or A/D conversion error)
1	The numerical data deviated from the "range of settings designated specifically with setting items."
2	The change of requested items is disabled or there are no items to be read.
3	An ASCII code other than the numerical data was specified in the field of numerical data. An ASCII code other than numbers and "-" was specified in the field of codes.
4	Format error
5	BCC error
6	Overrun error
7	Framing error
8	Parity error
9	A PV error occurred during AT. Or AT will not end 3 hours later.

	DWG.No. 45-5488-A	PAGE 13/39
--	----------------------	---------------

3.6 Communications precautions

3.6.1 Communications timing

Set a sufficient response delay to make sure that this product is switched over from transmission to reception with regard to a high-level computer in using an RS-485.

See the figure in "3.1. Communications procedure" and "2.8. Setting a response delay."

3.6.2 Interval between requests

In transmitting a series of "request messages" from a high-level computer, allow for an interval of 1msec or more from the reception of a "response message" from this product to a next transmission.

3.6.3 Response conditions

This product will not return a "response message" unless it receives a "request message" containing an STX and ETX (BCC).

If, therefore, the "request message" is error-ridden, this product will not return a "response message" (error reply) containing a NAK and ERR unless the conditions mentioned above are met.

Therefore transmit the necessary "request message" again if a "request message" is sent to the high-level computer but the latter does not return a "response message" at the end of an appropriate period.

The moment this product receives an STX, it clears all codes received before that.

3.6.4 Errors in address specification

This product will not respond to any "request message" that specifies an address other than that specified for itself. If, therefore, the address portion of a "request message" is error-ridden, none of the mobile units will return a "response message."

Therefore transmit the necessary "request message" again if a "request message" is sent to the high-level computer but the latter does not return a "response message" at the end of an appropriate period.

The moment this product receives an STX, it clears all codes received before that.

3.6.5 Number of digits in data and the decimal position

See "3.5. Description of codes, Numerical data."

3.6.6 Operation after receiving a store request message

This product starts to store data after correctly receiving a store request message from a high-level computer.

This product only stores data different from the contents of the EEPROM (data that is changed). The time (TW) required for storing data is within 6 seconds.

This product transmits a storage-complete reply (ACK) when the data is stored.

This product will not guarantee that the data is stored if this product is turned off during a storage operation. Do not turn off this product for 6 seconds after transmitting a store request message.

3.6.7 Operation after turning on the power

This product will not perform communications (no response) for about 4 seconds after it is turned on. Allow for a delay until communications is started after this product is turned on.

	DWG.No. 45-5488-A	PAGE 14/39
--	----------------------	---------------

3.6.8 Storing data other than a store request message

Store all parameters in the EEPROM in either of the two cases described below, even if no store request message is received.

- 1) If a parameter is changed by key operation
- 2) If auto-tuning is started and ends normally.

3.6.9 Changing the settings (SV or SV2) by communications during auto-tuning

Even if the settings (SV or SV2) used in control for auto-tuning are changed by communications, the settings (SV or SV2) will not be changed until the auto-tuning ends.

	DWG.No. 45-5488-A	PAGE 15/39
--	----------------------	---------------

4. Examples of TOHO communications

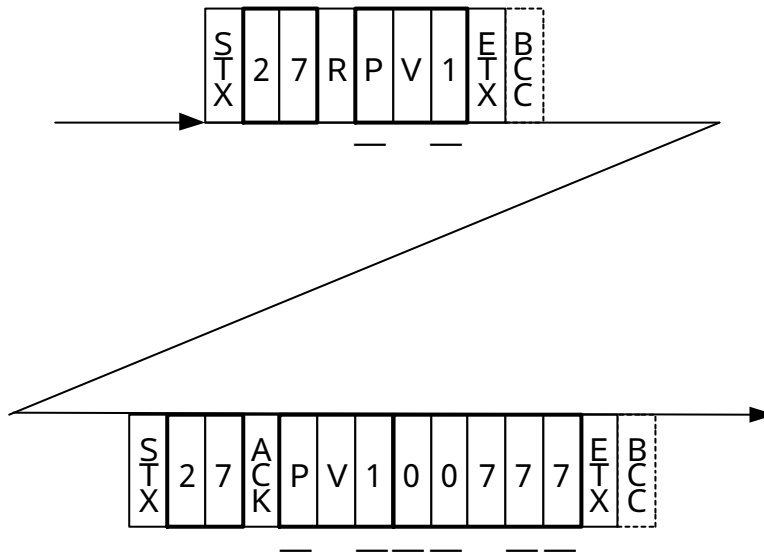
4.1 Examples of communications to be read

Example: Request message: This requests this product set at address 27 to read the PV.
(High-level computer)

In response to that,

Response message: This returns PV data (00777).

Read request message (transmitted from the high-level computer)



Code	Code, data	ASCII code, note 2)
Start code	STX	02H
Address	27	32H 37H
Request contents	R (Read)	52H
Identifier, note 1)	PV1	50H 56H 31H
Numerical data	00777	30H 30H 37H 37H 37H
End code	ETX	03H
BCC data request		61H
response		02H
Acknowledge code	ACK	06H

Note 1): See "10. Table of identifiers (codes)."

Note 2): For the ASCII codes, see "9. Table of ASCII codes."

	DWG.No. 45-5488-A	PAGE 16/39
--	----------------------	---------------

4.2 Examples of communications to be written

Example: Request message: This requests this product set at address 03 to set "the E1F setting to (High-level computer) 011" (write 011).

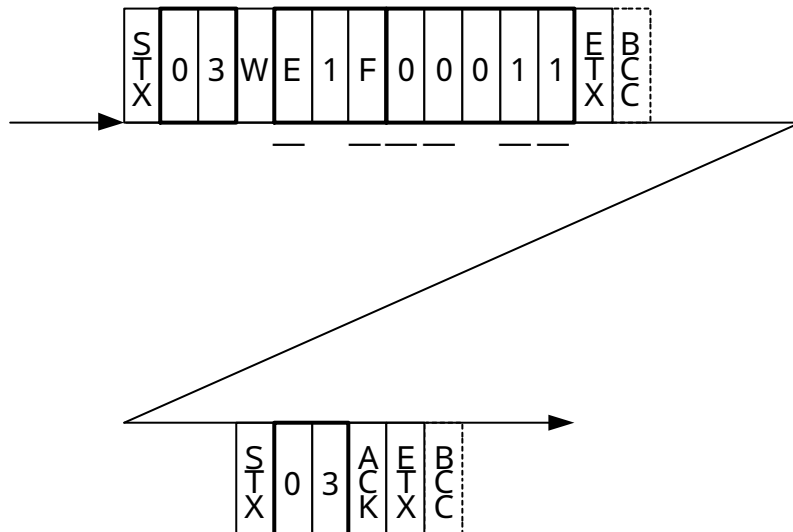
(This sets the function in event 1 to the deviation upper and lower limits + hold.)

In response to that,

Response message: This returns a notice that the request message has been received.
(This product)

*Check that it has been written by reading the data separately.

Write request message (transmitted from a high-level computer)



Code	Code, data	ASCII code, note 2)
Start code	STX	02H
Address	03	30H 33H
Request contents	W (Write)	57H
Identifier, note 1)	E1F	41H 34H 46H
Numerical data	00011	30H 30H 30H 31H 31H
End code	ETX	03H
BCC data request response		57H
		04H
Acknowledge code	ACK	06H

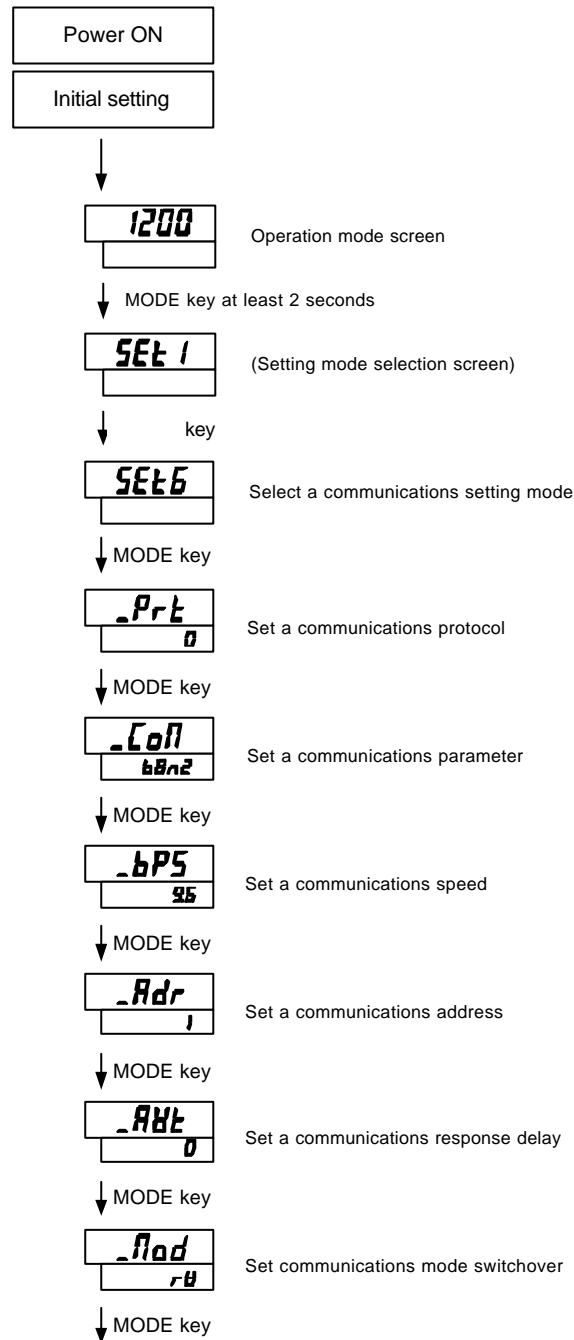
Note 1): See "10. Table of identifiers (codes)."

Note 2): For the ASCII codes, see "9. Table of ASCII codes."

5. Settings regarding MODBUS communications

5.1 Overview

Before communications is performed, initial settings must be made on this product. Enter such settings with the keys on the front panel. To switch to a series of setting screens, take the steps described below. For details, see the operation manual furnished with this product.



When the settings are over, press the MODE key at least 2 seconds to go back to the operation mode. The parameters indicated above are initial values.

	DWG.No. 45-5488-A	PAGE 18/39
--	----------------------	---------------

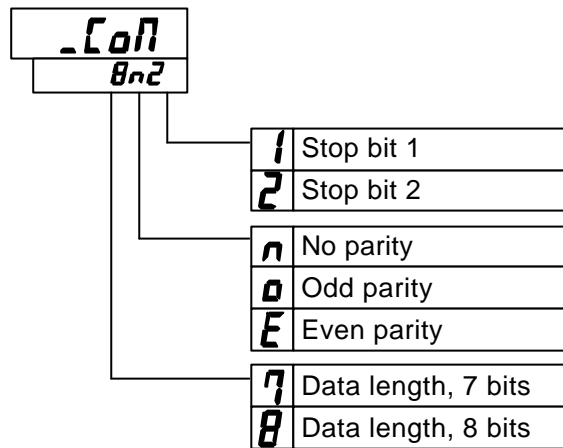
5.2 Setting a data length

5.3 Setting a stop bit length

5.4 Setting a parity

While in the "Set a communications parameter" screen on the preceding page, operate the and keys to make the settings.

The initial value is [**8n2**].

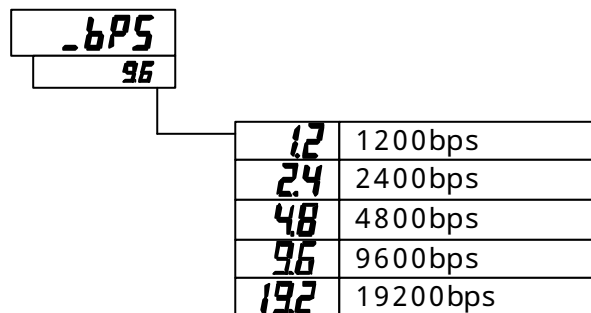


* The ASCII mode settings come only in three types: **7n2**, **7o1**, and **7E1**.
The RTU mode settings come only in three types: **8n2**, **8o1**, and **8E1**.

5.5 Setting a communications speed

While in the "Set a communications speed" screen on the preceding page, operate the and keys to make the settings.

The initial value is [**96**].



5.6 Setting an address

While in the "Set a communications address" screen on the preceding page, operate the and keys to make the settings.

The initial value is .



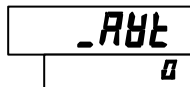
Setting range: 1 to 247 stations (It cannot be set to a 0.)

5.7 Setting a response delay

Set a time from the time when the high-level computer finished sending a "request message" until the time when it delivers the line and enters an input state.

While in the "Set a response delay" on the preceding page, operate the and keys to make the settings.

The initial value is 0.

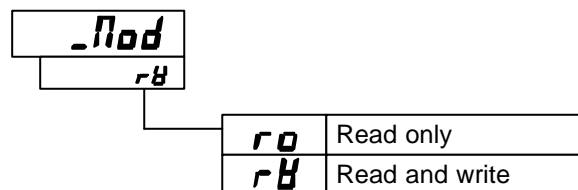


Setting range: 0 to 250ms

- * If the response delay is set to a short setting, the communications may not be conducted normally.
- * In a real operation, the processing time for this product will be added, in addition to the response delay.

5.8 Switching communications mode

While in the "Set communications mode switchover" screen on the preceding page, operate the and keys to make the settings.

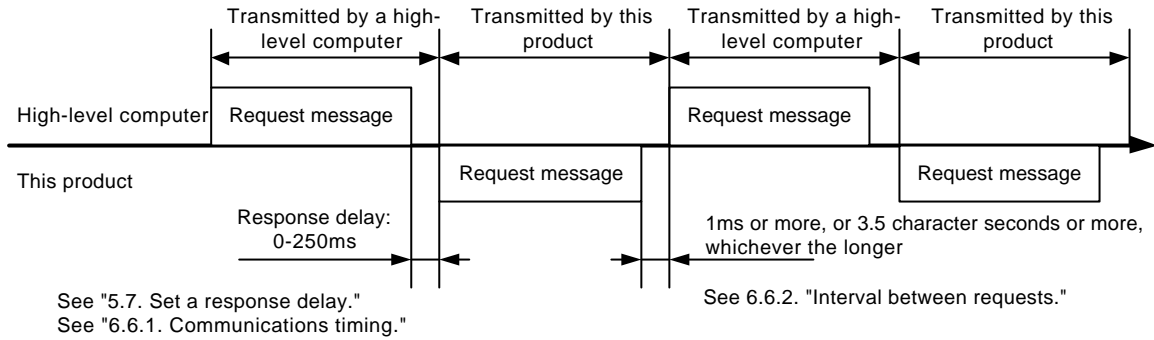


	DWG.No. 45-5488-A	PAGE 20/39
--	----------------------	---------------

6. MODBUS communications control

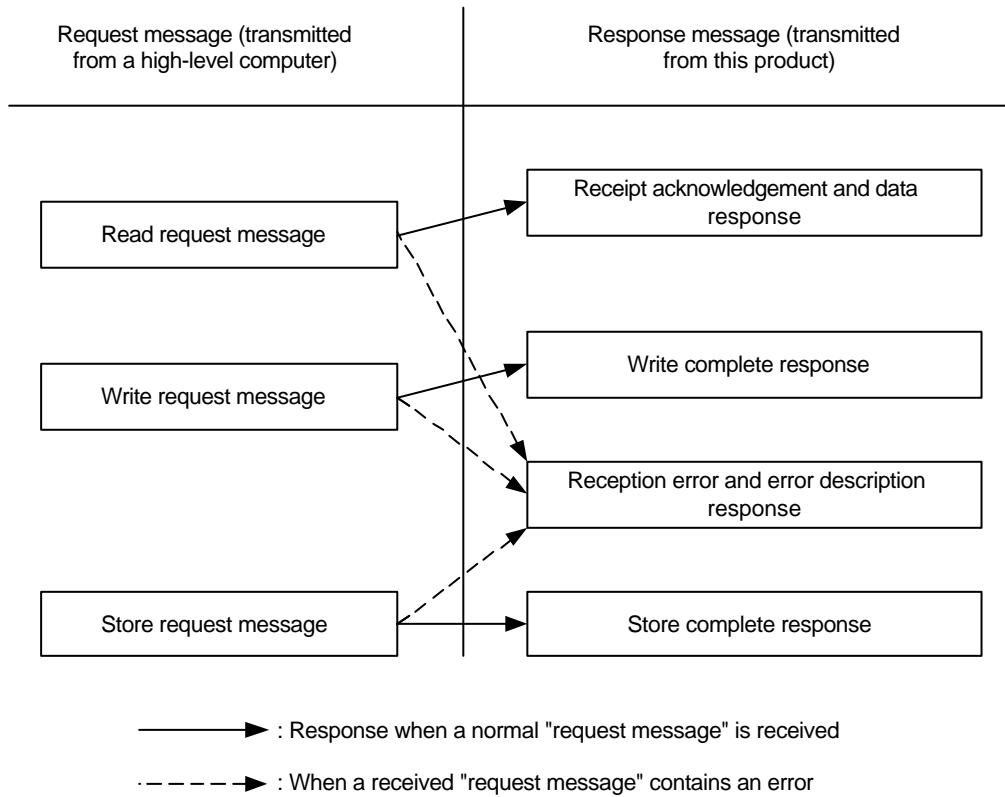
6.1 Communications procedure

This product returns a "response message" in response to a "request message" from a high-level computer. It therefore does not initiate a transmission.



6.2 Message types

- Messages are roughly divided into the following types:



- In RTU codes, the data is binary.
- In ASCII codes, all codes are expressed in ASCII codes.
- In assembling a program for a high-level computer, see "10. Table of identifiers (codes)" and "9. Table of ASCII codes" at the end of the book.

	DWG.No. 45-5488-A	PAGE 21/39
--	----------------------	---------------

6.3 Composition of an RTU request message (transmitted from a high-level computer to this product)

■ For codes a) through i), see "6.5. Description of RTU codes."

6.3.1 Composition of a read request message

a)	Slave address		1BH	
b)	Function code		03H	
c)	Register address	High level	00H	First register address
		Low level	00H	
d)	Number of registers	High level	00H	Fixed at 2
		Low level	02H	
e)	CRC-16	High level	C6H	
		Low level	31H	

6.3.2 Composition of a write request message

Multiple

a)	Slave address		03H	
b)	Function code		10H	
c)	Register address	High level	00H	First register address
		Low level	C0H	
d)	Number of registers	High level	00H	Fixed at 2
		Low level	02H	
f)	Number of data items		04H	Number of registers × 2
g)	Data for the first register (a low-level word)	High level	00H	When writing , , , and H in the data, write them in the order described on the left-hand side. (represents 1 byte.)
		Low level	6FH	
	Data for the first register + 1 (a high-level word)	High level	00H	
		Low level	00H	
e)	CRC-16	High level	C4H	
		Low level	5AH	

Single

a)	Slave address		03H	
b)	Function code		06H	
c)	Register address	High level	00H	First register address
		Low level	C0H	
g)	Data for the first register (a low-level word)	High level	00H	When writing and H in the data, write them in the order described on the left-hand side. (represents 1 byte.)
		Low level	6FH	
e)	CRC-16	High level	C4H	
		Low level	5AH	

	DWG.No. 45-5488-A	PAGE 22/39
--	----------------------	---------------

6.3.3 Composition of a store request message

a)	Slave address		03H	
b)	Function code		10H	
c)	Register address	High level	02H	First register address
		Low level	0EH	
d)	Number of registers	High level	00H	Fixed at 2
		Low level	02H	
f)	Number of data items		04H	Number of registers × 2
g)	Data for the first register (a low-level word)	High level	00H	The data about the storage of settings is arbitrary.
		Low level	00H	
g)	Data for the first register + 1 (a high-level word)	High level	00H	
		Low level	00H	
e)	CRC-16	High level	60H	
		Low level	FBH	

6.4 Composition of an RTU response message (transmitted from this product to a high-level computer)

- For codes a) through h), see "6.5. Description of RTU codes."

6.4.1 Response message for a read request message

a)	Slave address		1BH	
b)	Function code		03H	
d)	Number of data items		04H	Number of registers × 2
g)	Data for the first register (a low-level word)	High level	03H	
		Low level	09H	
g)	Data for the first register + 1 (a high-level word)	High level	00H	
		Low level	00H	
e)	CRC-16	High level	91H	
		Low level	B4H	

6.4.2 Response message for a write/store request message

Multiple

a)	Slave address		03H	
b)	Function code		10H	
c)	Register address	High level	00H	First register address
		Low level	00H	
d)	Number of registers	High level	00H	Fixed at 2
		Low level	02H	
e)	CRC-16	High level	40H	
		Low level	2AH	

	DWG.No. 45-5488-A	PAGE 23/39
--	----------------------	---------------

Single

a)	Slave address		03H	
b)	Function code		06H	
c)	Register address	High level	00H	First register address
		Low level	C0H	
g)	Data for the first register (a low-level word)	High level	00H	
		Low level	6FH	
e)	CRC-16	High level	C4H	
		Low level	5AH	

6.4.3 Response message in the case of an error

a)	Slave address		1BH	
b)	Function code		83H	← In the case of an error, the function code for the request message + 80H is entered.
h)	Error code		02H	
e)	CRC-16	High level	E1H	
		Low level	36H	

6.5 Description of RTU codes

■ The codes from a) slave address to b) function code to h) error code shown below are expressed in 8-bit binary numbers.

a) Slave address

This is the address of the party (this product) with which the high-level computer communicates. The address in the response message from this product represents the source of the response message. Note that, when CH2 is used, 2 addresses are occupied. (When the ADR is set to 1, addresses 1 and 2 are occupied.)

b) Function code

Enter a code 03H, 06H, or 10H.

03H: To read data from this product

06H: To write or store data in this product (multiple)

10H: To write or store data in this product (single)

c) Register address

The locations of the data to be read or that to be written are specified in 2 bytes.

The data is written in the holding register.

For the addresses of the commands, see "10. Table of identifiers (codes)."

d) Number of registers

This specifies the number of registers to be written in. Since this product has a fixed number of registers (which is 2), specify 0002H.

	DWG.No. 45-5488-A	PAGE 24/39
--	----------------------	---------------

e) CRC-16

This error check code is for detecting message errors. This transmits a CRC-16 (four redundancy code).

The multinomial for generating a CRC-16 used in this product is $X^{16}+X^{15}+X^2+1$.

To learn how to calculate the CRC-16, see "6.7. Example of CRC-16 calculations."

To affix an error code at the end of the message, affix the low-level byte first, then the high-level byte of the CRC.

f) Number of data

This specifies the number of registers to be read and written x 2. Since the number of registered in this product is fixed at 2, specify 04H here.

g) Data portion

This specifies data to be written in the register. The data is fixed at 4 bytes. This product will write data without the decimal point.

Example: In the case of numerical data

Example	Significance of the value
Proportional band (P) = 1.0 %	0000000AH
PV = 1200.0°C	00002EE0H
SV = -10.00°C	FFFFFFC18H

In the case of text data, write the ASCII code " INP" (is a space): 2049E50H.

h) Error code

If a message from a high-level computer is error-ridden, it will be incorporated in the "response message" from this product and returned.

The error number "04" is an instrument error (memory error or A/D conversion error, AT error). It will be incorporated in the "response message" regardless of whether there is an error in the "request message."

If there are two or more errors occurring at the same time, the largest error number will be incorporated.

The table below indicates the error contents and classifications.

Error No.	Error contents in the "request message" received by this product
01	Received an unsupported function code.
02	Received an address other than the specified one.
03	The numerical data deviated from the "range of settings designated specifically with setting items."
04	Instrument error (memory error or A/D conversion error, AT error)

	DWG.No. 45-5488-A	PAGE 25/39
--	----------------------	---------------

6.6 Precautions on RTU communications

6.6.1 Communications timing

Set a sufficient response delay to make sure that this product is switched over from transmission to reception with regard to a high-level computer in using an RS-485.

See the figure in "5.1. Overview" and "5.7. Setting a response delay."

3.6.2 Interval between requests

In transmitting a series of "request messages" from a high-level computer, allow for an interval of 1msec or more or 3.5 character minutes, whichever the longer, from the reception of a "response message" from this product to a next transmission.

6.6.3 Response conditions

If there is a time interval of 3.5 characters or more between data items constituting a "request message," this product cannot recognize it as a "request message." It will therefore not return a "response message." If, therefore, the "request message" contains an error, this product will not return a "response message" (error reply) containing an ERR unless the above conditions are met.

Therefore transmit the necessary "request message" again if a "request message" is sent to the high-level computer but the latter does not return a "response message" at the end of an appropriate period.

The moment a period of 3.5 characters or more has elapsed, it clears all codes received before that.

6.6.4 Errors in address specification

This product will not respond to any "request message" that specifies an address other than that specified for itself. If, therefore, the address portion of a "request message" is error-ridden, none of the mobile units will return a "response message."

Therefore transmit the necessary "request message" again if a "request message" is sent to the high-level computer but the latter does not return a "response message" at the end of an appropriate period.

6.6.5 Number of digits in data and the decimal position

See "6.5. Description of RTU codes, g) Data portion."

6.6.6 Operation after receiving a store request message

This product starts to store data after correctly receiving a store request message from a high-level computer.

This product only stores data different from the contents of the EEPROM (data that is changed). The time (TW) required for storing data is within 6 seconds.

This product transmits a storage-complete reply after the data is stored.

This product will not guarantee that the data is stored if this product is turned off during a storage operation. Do not turn off this product for 6 seconds after transmitting a store request message.

6.6.7 Operation after turning on the power

This product will not perform communications (no response) for about 4 seconds after it is turned on. Allow for a delay until communications is started after this product is turned on.

6.6.8 Storing data other than a store request message

This product will store all parameters in the EEPROM in either of the two cases described below, even if no store request message is received.

- 1) If a parameter is changed by key operation
- 2) If auto-tuning is started and ends normally.

6.6.9 Changing the settings (SV or SV2) by communications during auto-tuning

Even if the settings (SV or SV2) used in control for auto-tuning are changed by communications, the changes (SV or SV2) will not be changed until the auto-tuning ends.

6.7 Example of CRC-16 calculations

Following is an example of calculating CRC-16 with VisualBasic6.0.

Variables are declared as shown below.

VisualBasic6.0 cannot use code-free variables. It therefore uses code-equipped 16-bit integer variables as data. Similarly, the CRC calculation results are entered into code-equipped 32-bit integer variables.

```
Dim CRC As Long
```

```
Dim i, j, arry_count As Integer
```

```
Dim c_next, c_carry As LongDim crc_array(64) As Integer
```

Then enter calculable data into the `crc_array()`, and enter the number of data items into the `arry_count`. After that, run the following program to cause the calculation results to enter the CRC.

```
i = 0
CRC = 65535
For i = 0 To arry_count
  c_next = crc_array(i)
  CRC = (CRC Xor c_next) And 65535
  For j = 0 To 7
    c_carry = CRC And 1
    CRC = CRC ¥ 2
    If c_carry Then
      CRC = (CRC Xor &HA001) And 65535
    End If
  Next
Next
Next
```

To affix an error code to the end of the message, affix first the low-level byte and then the high-level byte of the CRC.

	DWG.No. 45-5488-A	PAGE 27/39
--	----------------------	---------------

6.8 Composition of an ASCII request message (transmitted from a high-level computer to this product)

- For the codes a) through g), see "6.10. Description of ASCII codes."

6.8.1 Composition of a read request message

a)	Start code		“.”	
b)	Slave address		“1”, “B”	
c)	Function code		“0”, “3”	
d)	Register address	High level	“0”, “0”	First register address
		Low level	“0”, “0”	
e)	Number of registers	High level	“0”, “0”	Fixed at 2
		Low level	“0”, “2”	
f)	LRC		“E”, “0”	
g)	End code		CR, LF	

6.8.2 Composition of a write request message

a)	Start code		“.”	
b)	Slave address		“0”, “3”	
c)	Function code		“1”, “0”	
d)	Register address	High level	“0”, “0”	First register address
		Low level	“C”, “0”	
e)	Number of registers	High level	“0”, “0”	Fixed at 2
		Low level	“0”, “2”	
h)	Number of data items		“0”, “4”	Number of registers × 2
i)	Data for the first register (a low-level word)	High level	“0”, “0”	When writing , , , and H in the data, write them in the order described on the left-hand side. (represents 1 byte.)
		Low level	“6”, “F”	
	Data for the first register + 1 (a high-level word)	High level	“0”, “0”	
		Low level	“0”, “0”	
f)	LRC		“E”, “0”	
g)	End code		CR, LF	

	DWG.No. 45-5488-A	PAGE 28/39
--	----------------------	---------------

6.8.3 Composition of a store request message

a)	Start code		“.”	
b)	Slave address		“0”, “3”	
c)	Function code		“1”, “0”	
d)	Register address	High level	“0”, “2”	First register address
		Low level	“0”, “E”	
e)	Number of registers	High level	“0”, “0”	Fixed at 2
		Low level	“0”, “2”	
h)	Number of data items		“0”, “4”	Number of registers × 2
i)	Data for the first register (a low-level word)	High level	“0”, “0”	The data about the storage of settings is arbitrary.
		Low level	“0”, “0”	
	Data for the first register + 1 (a high-level word)	High level	“0”, “0”	
		Low level	“0”, “0”	
f)	LRC		“D”, “7”	
g)	End code		CR, LF	

6.9 Composition of ASCII response messages (transmitted from this product to a high-level computer)

- For the codes a) through g), see "6.10. Description of ASCII codes."

6.9.1 Response message for a read request message

a)	Start code		“.”	
b)	Slave address		“1”, “B”	
c)	Function code		“0”, “3”	
h)	Number of data items		“0”, “4”	Number of registers × 2
i)	Data for the first register (a low-level word)	High level	“0”, “3”	When writing , , , and H in the data, write them in the order described on the left-hand side. (represents 1 byte.)
		Low level	“0”, “9”	
	Data for the first register + 1 (a high-level word)	High level	“0”, “0”	
		Low level	“0”, “0”	
f)	LRC		“D”, “2”	
g)	End code		CR, LF	

	DWG.No. 45-5488-A	PAGE 29/39
--	----------------------	---------------

6.9.2 Response message for a write/store request message

a)	Start code		"."	
b)	Slave address		"0", "3"	
c)	Function code		"1", "0"	
d)	Register address	High level	"0", "0"	First register address
		Low level	"0", "0"	
e)	Number of registers	High level	"0", "0"	Fixed at 2
		Low level	"0", "2"	
f)	LRC		"E", "B"	
g)	End code		CR, LF	

6.9.3 Response message in the case of an error

a)	Start code		"."	
b)	Slave address		"1", "B"	
h)	Function code		"8", "3"	← In the case of an error, the function code for the request message + 80H is entered.
j)	Error code		"0", "2"	
f)	LRC		"6", "0"	
g)	End code		CR, LF	

6.10 Description of ASCII codes

- The codes from a) start code to b) slave address to j) error type described below are expressed in ASCII codes.
- For ASCII codes, see "9. Table of ASCII codes."
- For converting to ASCII codes, see 6.8 and 6.9 "Message composition."

a) Start code

The receiver side is the code required for detecting the top of the message. It is affixed to the top of a character string to be transmitted.

b) Slave address

This is the address of the party (this product) with which the high-level computer communicates. The address in the response message from this product represents the source of the response message. Note that, when CH2 is used, 2 addresses are occupied. (When the ADR is set to 1, addresses 1 and 2 are occupied.)

c) Function code

Enter a code 03H or 10H.

03H: To read data from this product

10H: To write or store data in this product

d) Number of registers

This specifies the number of registers to be written in. Since this product has a fixed number of registers (which is 2), specify 0002H.

e) Register address

The locations of the data to be read or that to be written are specified in 2 bytes.

For the addresses of the commands, see "10. Table of identifiers (codes)."

f) LRC

LRC is an error check code for detecting message errors. An LRC is transmitted. The LRC used in this product is the 2-complement of the sum of the data portions without a carry, except for the start code and end code of the message.

The parts of the data portions expressed as a "1" and "B" are considered as "1BH."

To learn how to calculate the LRC, see "6.12. Example of LRC calculations."

If 12H is calculated as an error code, affix a "1" or "2" at the end of the message.

g) End code

This code is required for the receiver to detect the end of a message. Affix CR (0DH) and LF (0AH) at the end of a character string to be transmitted.

h) Number of data

This specifies the number of registers to be read and written x 2. Since the number of registered in this product is fixed at 2, specify 04H here.

	DWG.No. 45-5488-A	PAGE 31/39
--	----------------------	---------------

i) Data portion

This specifies data to be written in the register. The data is fixed at 4 bytes. This product will write data without the decimal point.

Example: In the case of numerical data

Example	Significance of the value
Proportional band (P) = 1.0 %	0000000AH
PV = 1200.0°C	00002EE0H
SV = -10.00°C	FFFFFFC18H

In the case of text data, write the ASCII code " INP" (is a space): 2049E50H.

j) Error code

If a message from a high-level computer is error-ridden, it will be incorporated in the "response message" from this product and returned.

The error number "04" is an instrument error (memory error or A/D conversion error, AT error). It will be incorporated in the "response message" regardless of whether there is an error in the "request message."

If there are two or more errors occurring at the same time, the largest error number will be incorporated.

The table below indicates the error contents and classifications.

Error No.	Error contents in the "request message" received by this product
01	Received an unsupported function code.
02	Received an address other than the specified one.
03	The numerical data deviated from the "range of settings designated specifically with setting items."
04	Instrument error (memory error or A/D conversion error, AT error)

	DWG.No. 45-5488-A	PAGE 32/39
--	----------------------	---------------

6.11 Precautions on ASCII communications

6.11.1 Communications timing

Set a sufficient response delay to make sure that this product is switched over from transmission to reception with regard to a high-level computer in using an RS-485.

See the figure in "5.1. Overview" and "5.7. Setting a response delay."

6.11.2 Interval between requests

In transmitting a series of "request messages" from a high-level computer, allow for an interval of 1msec or more or 3.5 character minutes, whichever the longer, from the reception of a "response message" from this product to a next transmission.

6.11.3 Response conditions

This product will not return a "response message" unless the "request message" contains a start code and end code.

If, therefore, the "request message" contains an error, this product will not return a "response message" (error reply) containing an error code unless the above conditions are met.

Therefore transmit the necessary "request message" again if a "request message" is sent to the high-level computer but the latter does not return a "response message" at the end of an appropriate period.

The moment a start code is received, this product clears all codes received before that.

6.11.4 Errors in address specification

This product will not respond to any "request message" that specifies an address other than that specified for itself. If, therefore, the address portion of a "request message" is error-ridden, none of the mobile units will return a "response message."

Therefore transmit the necessary "request message" again if a "request message" is sent to the high-level computer but the latter does not return a "response message" at the end of an appropriate period.

The moment a start is received, this product clears all codes received before that.

6.11.5 Number of digits in data and the decimal position

See "6.10. Description of ASCII codes, i) Data portion."

6.11.6 Operation after receiving a store request message

This product starts to store data after correctly receiving a store request message from a high-level computer.

This product only stores data different from the contents of the EEPROM (data that is changed). The time (TW) required for storing data is within 6 seconds.

This product transmits a storage-complete reply after the data is stored.

This product will not guarantee that the data is stored if this product is turned off during a storage operation. Do not turn off this product for 6 seconds after transmitting a store request message.

6.11.7 Operation after turning on the power

This product will not perform communications (no response) for about 4 seconds after it is turned on. Allow for a delay until communications is started after this product is turned on.

6.11.8 Storing data other than a store request message

This product will store all parameters in the EEPROM in either of the two cases described below, even if no store request message is received.

- 1) If a parameter is changed by key operation
- 2) If auto-tuning is started and ends normally.

6.11.9 Changing the settings (SV or SV2) by communications during auto-tuning

Even if the settings (SV or SV2) used in control for auto-tuning are changed by communications, the changes (SV or SV2) will not be changed until the auto-tuning ends.

6.12 Example of LRC calculations

Following is an example of calculating LRC with VisualBasic6.0.

Variables are declared as shown below.

VisualBasic6.0 cannot use code-free variables. It therefore uses code-equipped 16-bit integer variables as data. Similarly, the LRC calculation results are entered into code-equipped 16-bit integer variables.

```
Dim LRC As Integer
Dim i, arry_count As Integer

Dim lrc_arry(128) As Integer
```

Then enter calculable data into the lrc_arry(), and enter the number of data items into the arry_count. After that, run the following program to cause the calculation results to enter the LRC.

```
For i = 0 To arry_count
    LRC = (LRC + lrc_arry(i)) And &HFF
Next

LRC = ((Not LRC) + 1) And &HFF
```

If the error code is calculated as 12H as an example, affix a "1" or "2" at the end of the message.

7. Specifications

7.1 Communications standard category: Compliant with EIA standard RS-485

7.2 Communications specifications

7.2.1 Communications system

Network:..... For RS-485, multi-drop system (up to 1 pair, 31 stations)
 Direction of information:..... Half duplex
 Synchronization system:..... Asynchronous
 Transmission code:..... ASCII, 7/8 bit code, except for BBC data
 (highest-level bit = 0 in 8-bit code)

7.2.2 Interface system

Signal line:..... 2 lines for transmission and reception
 Communications speed: One speed is selected from 1,200, 4,800, 9,600, and 19,200 bps
 and this product is set to it.
 Communications distance:..... 500m maximum
 Provided that it varies somewhat depending on the cable and other
 ambient conditions.

7.2.3 TOHO communications characters

Start bit length: Fixed at 1 bit
 Stop bit length:..... Either 1 or 2 bit is selected and this product is set to it.
 Data length:..... Either 7 or 8 bit is selected and this product is set to it.
 Parity:..... No. Either odd or even is selected and this product is set to it.
 BCC check:..... Yes or no is selected and this product is set to it.
 Communications address:..... 1-99

7.2.4 MODBUS communications (RTU) characters

Start bit length: Fixed at 1 bit
 Stop bit length:..... Either 1 or 2 bit is selected and this product is set to it. (If
 parity-equipped, fixed at 1 bit.)
 Data length:..... Fixed at 8 bit.
 Parity:..... No. Either odd or even is selected and this product is set to it.
 CRC-16 check: Fixed at yes.
 Communications address:..... 1-247

7.2.5 MODBUS communications (ASCII) characters

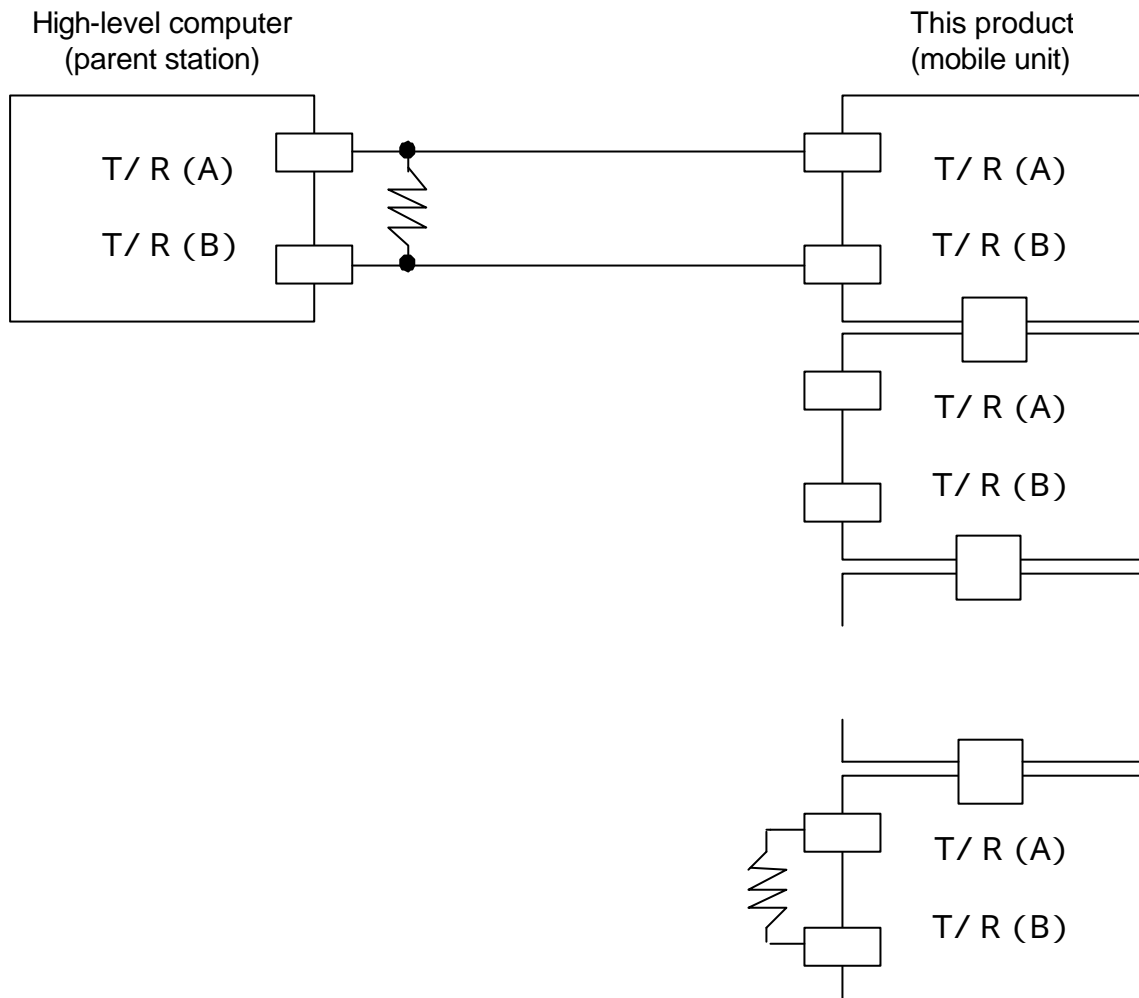
Start bit length: Fixed at 1 bit
 Stop bit length:..... Either 1 or 2 bit is selected and this product is set to it. (If
 parity-equipped, fixed at 1 bit.)
 Data length:..... Fixed at 7 bit.
 Parity:..... No. Either odd or even is selected and this product is set to it.
 LRC check: Fixed at yes.
 Communications address:..... 1-247

7.2.6 MODBUS communications (ASCII/RTU) function codes

03H (reading the contents of the holding register)
 10H (writing the contents of two or more holding registers)

8. Connections

8.1 Connections for the RS-485



Communications is such that the connections can be made with the connectors on the sides.

Install an end of line resistor at both of the farthest devices in the parent station and the mobile unit. For a resistance value, use one that matches the characteristic impedance of the cable. Provided that the synthesis is set to at least 75Ω .

9. Table of ASCII codes

ASCII code	0 0 H	0 1 H	0 2 H	0 3 H	0 4 H	0 5 H	0 6 H	0 7 H
	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL

ASCII code	0 8 H	0 9 H	0 A H	0 B H	0 C H	0 D H	0 E H	0 F H
	BS	HT	LF	VT	FF	CR	SO	SI

ASCII code	1 0 H	1 1 H	1 2 H	1 3 H	1 4 H	1 5 H	1 6 H	1 7 H
	DLE	DC1	DC2	DC3	DC4	NAK	SYM	ETB

ASCII code	1 8 H	1 9 H	1 A H	1 B H	1 C H	1 D H	1 E H	1 F H
	CAN	EM	SUB	ESC	FS	GS	RS	US

ASCII code	2 0 H	2 1 H	2 2 H	2 3 H	2 4 H	2 5 H	2 6 H	2 7 H
	λ [°] -λ	!	"	#	\$	%	&	'

ASCII code	2 8 H	2 9 H	2 A H	2 B H	2 C H	2 D H	2 E H	2 F H
	()	*	+	,	-	.	/

ASCII code	3 0 H	3 1 H	3 2 H	3 3 H	3 4 H	3 5 H	3 6 H	3 7 H
	0	1	2	3	4	5	6	7

ASCII code	3 8 H	3 9 H	3 A H	3 B H	3 C H	3 D H	3 E H	3 F H
	8	9	:	;	<	=	>	?

ASCII code	4 0 H	4 1 H	4 2 H	4 3 H	4 4 H	4 5 H	4 6 H	4 7 H
	@	A	B	C	D	E	F	G

ASCII code	4 8 H	4 9 H	4 A H	4 B H	4 C H	4 D H	4 E H	4 F H
	H	I	J	K	L	M	N	O

ASCII code	5 0 H	5 1 H	5 2 H	5 3 H	5 4 H	5 5 H	5 6 H	5 7 H
	P	Q	R	S	T	U	V	W

ASCII code	5 8 H	5 9 H	5 A H	5 B H	5 C H	5 D H	5 E H	5 F H
	X	Y	Z	[¥]	^	—

ASCII code	6 0 H	6 1 H	6 2 H	6 3 H	6 4 H	6 5 H	6 6 H	6 7 H
	'	a	b	c	d	e	f	g

ASCII code	6 8 H	6 9 H	6 A H	6 B H	6 C H	6 D H	6 E H	6 F H
	h	i	j	k	l	m	n	o

ASCII code	7 0 H	7 1 H	7 2 H	7 3 H	7 4 H	7 5 H	7 6 H	7 7 H
	p	q	r	s	t	u	v	w

ASCII code	7 8 H	7 9 H	7 A H	7 B H	7 C H	7 D H	7 E H	7 F H
	x	y	z	{		}	~	DEL

		DWG. No. 45-5488-A	PAGE 37/39
--	--	-----------------------	---------------

10. Table of identifiers (codes)

MODBUS ADR Operation mode

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
0	1	PV1		Measurement	R	Use it as a monitor for measurements (PV). When overscale: HHHH When underscale: LLLL
2	3	SV1		Control setting	R/W	R/W of the setting (SV)

MODBUS ADR Initial setting mode

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
4	5	INP	- <i>INP</i>	Set an input type	R/W	0 0 0 ? ?
6	7	PVG	- <i>PVG</i>	Set a PV corrected gain	R/W	
8	9	PVS	- <i>PVS</i>	Set a PV corrected zero point	R/W	
10	11	PDF	- <i>PDF</i>	Set a PV filter	R/W	
12	13	<input type="checkbox"/> DP	- <i>dP</i>	Set a decimal position	R/W	No decimal point : 00000 1 decimal place : 00001 2 decimal place : 00002 3 decimal place : 00003
14	15	<input type="checkbox"/> FU	- <i>FU</i>	Set a function key function	R/W	
16	17	LOC	- <i>LoC</i>	Set a key lock	R/W	

MODBUS ADR Control setting mode

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
18	19	SLH	- <i>SLH</i>	Set an SV limiter upper limit	R/W	
20	21	SLL	- <i>SLL</i>	Set an SV limiter lower limit	R/W	
22	23	<input type="checkbox"/> MD	- <i>nd</i>	Set control mode	R/W	Control execution : 00000 Manual control : 00001 Control stop : 00002 Auto-tuning in progress : 00003
24	25	CNT	- <i>Cnt</i>	Set a control type	R/W	
26	27	DIR	- <i>dir</i>	Set positive/reverse operation switchover	R/W	
28	29	MV1	- <i>Mv1</i>	Amount of output operation	R/W	W possible during manual control
30	31	TUN	- <i>tun</i>	Set a tuning type	R/W	
32	33	ATG	- <i>AtG</i>	Set an AT factor	R/W	
34	35	ATC	- <i>AtC</i>	Set an AT sensitivity	R/W	
36	37	<input type="checkbox"/> P1	- <i>P1</i>	Set a proportional band	R/W	
38	39	<input type="checkbox"/> I1	- <i>I1</i>	Set an integral time	R/W	
40	41	<input type="checkbox"/> D1	- <i>d1</i>	Set a derivative time	R/W	
42	43	<input type="checkbox"/> T1	- <i>t1</i>	Set a proportional frequency	R/W	
44	45	ARW	- <i>ArW</i>	Anti-reset windup	R/W	
46	47	MH1	- <i>MH1</i>	Set an amount-of-operation limiter upper limit	R/W	
48	49	ML1	- <i>ML1</i>	Set an amount-of-operation limiter lower limit	R/W	
50	51	PBB	- <i>Pbb</i>	Manual reset	R/W	
52	53	MV2	- <i>Mv2</i>	Amount of cooling output operation	R/W	W possible during manual control
54	55	<input type="checkbox"/> P2	- <i>P2</i>	Set a cooling power proportional band	R/W	
56	57	<input type="checkbox"/> T2	- <i>t2</i>	Set a cooling output proportional cycle	R/W	
58	59	MH2	- <i>MH2</i>	Set a cooling output amount-of-operation limiter upper limit	R/W	
60	61	ML2	- <i>ML2</i>	Set a cooling output amount-of-operation limiter lower limit	R/W	
62	63	<input type="checkbox"/> C1	- <i>C1</i>	Set a (heating output) control sensitivity	R/W	
64	65	<input type="checkbox"/> C2	- <i>C2</i>	Set a cooling output control sensitivity	R/W	
66	67	CP1	- <i>CP1</i>	Set an off-point position	R/W	
68	69	CP2	- <i>CP2</i>	Set a cooling output off-point position	R/W	
70	71	<input type="checkbox"/> DB	- <i>db</i>	Set a dead band	R/W	

		DWG. No.	PAGE
		45-5488-A	38/39

MODBUS ADR Event setting mode

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
72	73	E1F	<i>-E 1F</i>	Set an event output function	R/W	
74	75	E1H	<i>-E 1H</i>	Set an event output upper limit	R/W	
76	77	E1L	<i>-E 1L</i>	Set an event output lower limit	R/W	
78	79	E1C	<i>-E 1C</i>	Set an event output sensitivity	R/W	
80	81	E1T	<i>-E 1T</i>	Set an event output delay timer	R/W	
82	83	E1B	<i>-E 1b</i>	Set an event output special function	R/W	
84	85	E1P	<i>-E 1P</i>	Set an event output polarity	R/W	
86	87	CM1	<i>-C M 1</i>	CT monitor	R	
88	89	CT1	<i>-C t 1</i>	Set a CT abnormal current	R/W	

MODBUS ADR Timer setting mode

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
90	91	TMO	<i>-t M o</i>	Set a timer output destination	R/W	
92	93	TMF	<i>-t M F</i>	Set a timer function	R/W	
94	95	H/M	<i>-H / M</i>	Set a timer unit	R/W	
96	97	TSV	<i>-t S v</i>	Set a timer SV start tolerance	R/W	
98	99	TIM	<i>-t I M</i>	Set a timer time	R/W	
100	101	TIA	<i>-t I A</i>	Set a timer remaining time	R	

MODBUS ADR DI setting mode

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
102	103	DIF	<i>-d I F</i>	Set a DI function	R/W	
104	105	DIP	<i>-d I P</i>	Set a DI polarity	R/W	
106	107	SV2	<i>1 S v 2</i>	Set a CH1 SV2	R/W	
108	109	SV2	<i>2 S v 2</i>	Set a CH2 SV2	R/W	

MODBUS ADR Communications setting mode

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
110	111	PRT	<i>-P r t</i>	Set a communications protocol	R/W	
112	113	COM	<i>-C o m</i>	Set a communications parameter	R/W	Example: □B8N2
114	115	BPS	<i>-b P S</i>	Set a communications speed	R/W	Example: 00096 (if 9600 bps)
116	117	ADR	<i>-A d r</i>	Set a communications address	R/W	
118	119	AWT	<i>-A w t</i>	Set a communications response delay	R/W	
120	121	MOD	<i>-M o d</i>	Set communications mode switchover	R/W	RO : 0 0 0 0 0 RW : 0 0 0 0 1

MODBUS ADR Screen-less commands

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
122	123	TST		Timer start stop	R/W	
124	125	OM1		Output monitor	R	R of the output monito ①②③④⑤ ⑤ : Control output (1:ON 0:OFF) ④ : Cooling output (1:ON 0:OFF) ③ : Event output (1:ON 0:OFF)
126	127	EM1		DI monitor	R	R of DI monitor ①②③④⑤ ⑤ : DI (1: ON 0: OFF)
128	129	□AT		Start/release AT	R/W	Read/write AT start/release Start : 00001 Release : 00000 Reading during startup causes this product to replay with 00001.
130	131	STR		Store data	W	Store data

		DWG. No. 45-5488-A	PAGE 39/39
--	--	-----------------------	---------------

MODBUS ADR Identifiers used only in blind setting

Low-level W	High-level W	Identifier	Character	Name	R/W	Description
132	133	001		SET 1	L/B	Blinding enabled : 00000 Blinding disabled 00001
134	135	002		SET 2	L/B	
136	137	003		SET 3	L/B	
138	139	004		SET 4	L/B	
140	141	005		SET 5	L/B	
142	143	006		SET 6	L/B	