INST.№ INE-350

CHINO

LT830 SERIES
DIGITAL INDICATING CONTROLLER
COMMUNICATIONS INTERFACES

INSTRUCTIONS

Retain this manual apart from the instrument and in an easily

Please make sure that this manual is handed to the final user of the instrument.

CHINO

Contents

INTRODUCTION

This instruction manual describes the specifications and operation built-in communications interfaces (RS-485) of the LT830 series Digital Indicating Controllers.
Be sure to confirm the model code of the LT830 controllers you purchased as this determines the required option.

LT830□□000－□□□

└Event output／Communications IF／CT

0: None
1: 2 points of Event output
2: 2 points of Event output+ Communications IF+CT

1. Other Instruction Manual to be consulted.

This instruction manual is for the communications interface only. For the running and operation, please refer to the following instruction manuals.

1. LT830 series Digital Indicating Controllers (Manual No. LT8-11-□)
2. SC8-10 Line Converter (Manual No. SC8-10-□)
   * Also refer to the instruction manual of the computer being used.

2. Caution Display ⚠ Caution

This manual contains explanation of precautions. Observe these precautions when operating and handling the communications interfaces, otherwise the instrument may be damaged, resulting in a deterioration in its performance, or operation failures may also occur.

⚠ Caution

(1) The right is reserved to change the contents of this manual at any time without notice.

(2) The contents of this manual have been prepared professionally. However, if you have any questions, or notice of error or an omission of descriptions found on this manual, please contact your nearest CHINO sales agent.

(3) CHINO Corporation is not responsible for any results influenced by the operation of this communications interface, irrespective of item (2) above.

1 GENERAL

There are one types of communication interfaces RS-485 available between the LT830 controllers and personal computers (PCs).
PCs can be used to receive measured data from the LT830 controllers, program different parameters and issue control commands. The number of LT830 controllers that are connectable to a PC is up to 31 for the RS-485.

1.1 RS-232C Communications Interface

The RS-232C is the data communications standard being set and issued by EIA (Electronic Industries Association) in the USA and JIS C 6361 in Japan.
This standard is a basic interface between MODEM and connected data terminal units, and specifies electrical and mechanical specifications only. Most of the RS-232C communications interface is being used for personal computers and industrial instruments do not completely conform to this standard at present, and have different signal wire numbers, connectors to those specified in the standard. Also, since this standard does not specify any software parts, or so-called [data transmission procedures], units having the RS-232C communications interface can not be interconnected with each other unconditionally. With these reasons, users must survey and check the specifications and transmission procedures in advance of units being connected. However, if the counter unit is for a personal computers or similar device which can optionally program the specifications, then all the units can be combined by having proper programs prepared by a program designer.
In addition, when a RS-232C standard is investigated, the method of having JIS C 6361 referred to is the easiest.

1.2 RS-485 Communications Interface

The RS-485 communications interface can communicate with multiple LT830 controllers (up to 31 sets) in parallel by signals conforming to RS-485.
There are few personal computers which provide RS-485 communications interface. However, since these communications interfaces are characterized with serial communications, these are easily connectable to the personal computer having the RS-232C communications interface by using an RS-485 ⇔ RS-232C signal converting unit. A line converter (Model SC8-10: sold separately) is available for RS-485 ⇔ RS-232C signal conversion at CHINO.

2 COMMUNICATIONS PROTOCOL

LT830 controller has a MODOBUS communication protocol.

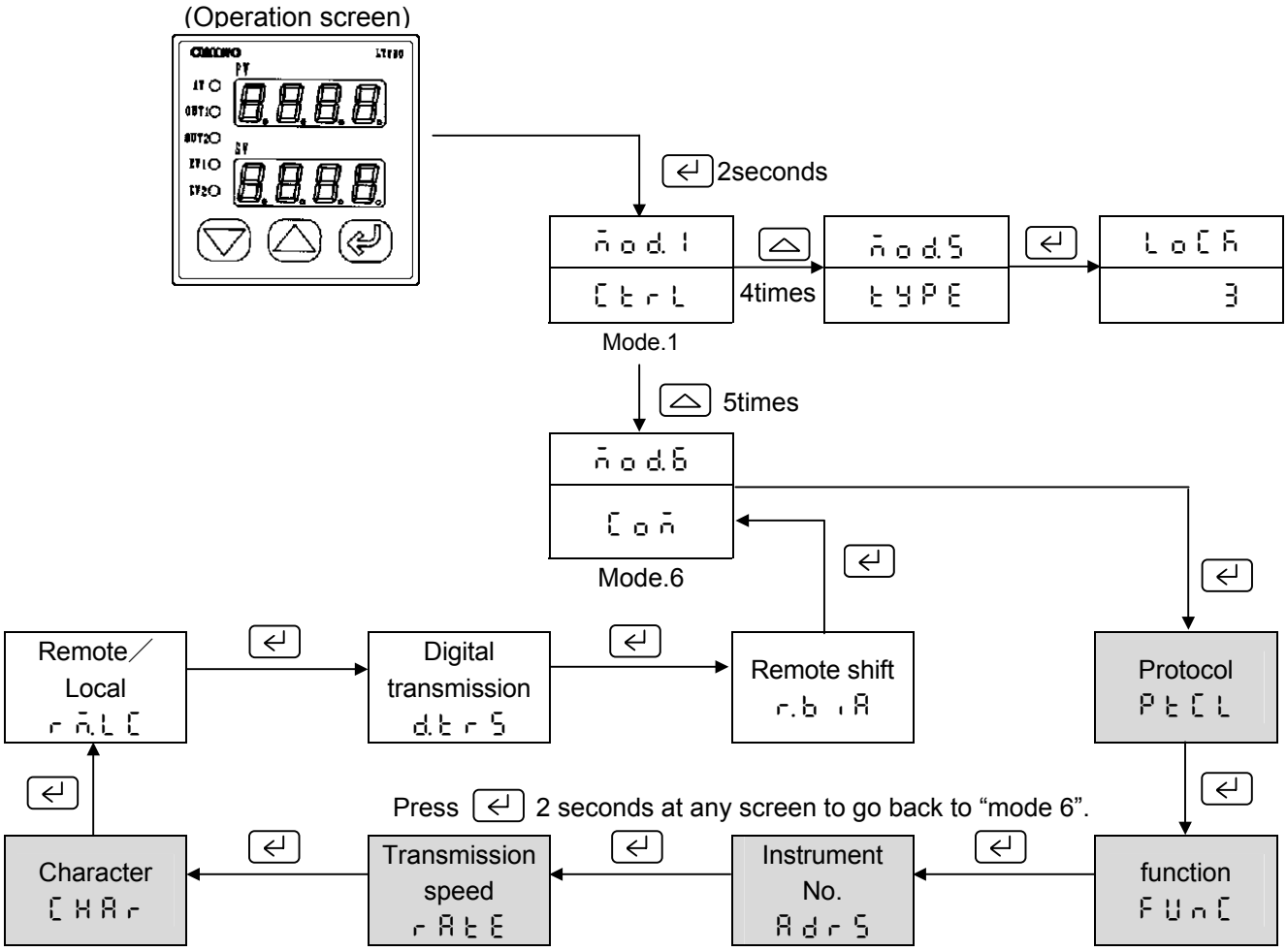
MODBUS Protocol (MODBUS is the registered trademark of Schneider Automation Inc.)
MODBUS Protocol has RTU mode and ASCII mode that can be selected by key programming.
MODBUS protocol provides the function of transmitting measured data as well as the programming and operating function.

3 COMMUNICATIONS SPECIFICATIONS

* Communication system: Half-duplex start-stop synchronization (polling selecting system)
* Protocol: MODBUS protocol
* Transmission speed: 19200, 9600 bps selectable
* Start bit: 1 bit
* Data length: 7 bits (ASCII)/8 bits (RUT/ASCII) selectable
* Parity bit: Even / Odd / Disabled selectable
* Stop bit: 1 bit / 2 bits selectable
* Transmission code: Binary (RTU) / ASCII (ASCII) (depending on protocol)
* Error check: LRC (ASCII)/CRC-16 (RTU) Depending on protocol
* Data transmission procedure: None
* Signals in use: Transmitted and received data only (no control signal in use)

4 SETTING OF COMMUNICATIONS PARAMETERS

Follow the flow chart and set up 5 items such as “Protocol”, “Communications function”, “Instrument No.”, “Transmission speed” and “Character”.



When you set the parameter (writing), set the key lock at first.

LT830 controller is always ready for communication. It responds at anytime to data requests from the personal computer. However, when you set the parameter or control the instrument from the personal computer, you need to set key lock (lock 3) on LT830 at first. Key lock will be set at the front key on LT830 or through personal computer communication. If you try to set the parameter or operate instrument from the personal computer at Non-lock, Lock1, Lock 2 LT830 will show error code 12H. （See paragraph 6.6）

4.1 Setting of Protocol (PtCL)

(1) Press [↵] to see [PtCL].
(2) Choose the protocol by pressing [△] and [▽], then press [↵] to register.

Kind	Protocol	Default
rtU	MODBUS RTU	rtU
ASC	MODBUS ASCII	

*When you change the protocol, the communication function will become initial value.

4.2 Setting of Communications Function (FUnC)

(1) Press [↵], then you will see [FUnC]
(2) Choose the communications function by pressing [△] and [▽], then press [↵] to register.

Kind	Function	Protocol	Details	Default
Com	Upper Communication	MODBUS	Set up the upper communication when you communicate with the personal computer.	Com
trS.2	Communications Transmission		By setting up "trs.2" at communications transmission, the transmitting date at LT830 will be sent to LT830 that was set up with [CoM]. It also transmits SV No. and Run/Ready information. (See **7** Communications Transmission and Communications Remote) PV: Transmit the measured data (default). SV: Transmit the set up value. Type of transmission data can be set at "mode 7", which is digital transmission.	

4.3 Setting of Instrument No. (AdrS)

From one to a few LT830, which communicate with personal computer, will be set the instrument Number that dose not fell on another LT830

(1) Press [↵] to display [AdrS]

(2) Choose the instrument number (from 1 to 99) by pressing [△] or [▽] , then press [↵] to register.

CAUTION

The instrument number should be from 1 to 99, and should not fell on another LT830.（default 1）

4.4 Setting of Transmission speed (rAtE)

Operate LT830 and personal computer by the same Transmission speed.
(Usually, the default 9600 bps is fine.)

(1) Press [↵] to display see [rAtE]

(2) Choose the Transmission speed by pressing [△] or [▽] , then press [↵] to register Transmission speed : 9600, 19200 (19.2k) bps (default is 9600 bps.)

4.5 Setting of Character (CHAr)

(1) Press [↵] to display [CHAr]

(2) Choose the character by pressing [△] or [▽] , then press [↵] to register.

Kind	Bit length	Parity	Stop bit	Default value
1	7bit	Even	1	5
2			2	
3		Odd	1	
4			2	
5	8bit	None	1	
6			2	
7		Even	1	
8			2	
9		Odd	1	
10			2	

*If the protocol is MODBUS RTU, 8 bit length is the only choice.

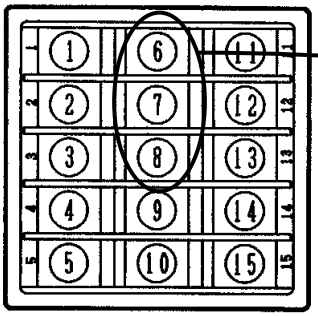
5 CONNECTIONS

5.1 Connection Precautions

5.1.1 Communication Terminals

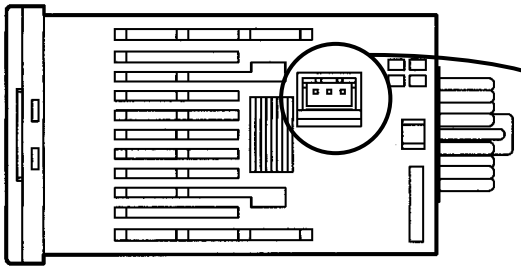
Disposition of terminals are different from each communications interface.

◇ Terminal block type



№	RS-485
⑥	SA
⑦	SB
⑧	SG

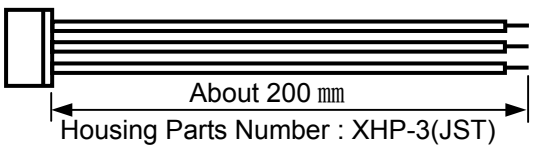
◇ Socket type



RS-485		
■	■	■
SG	SB	SA

* The communications cable is included with the socket

【Connector】
LT830 side



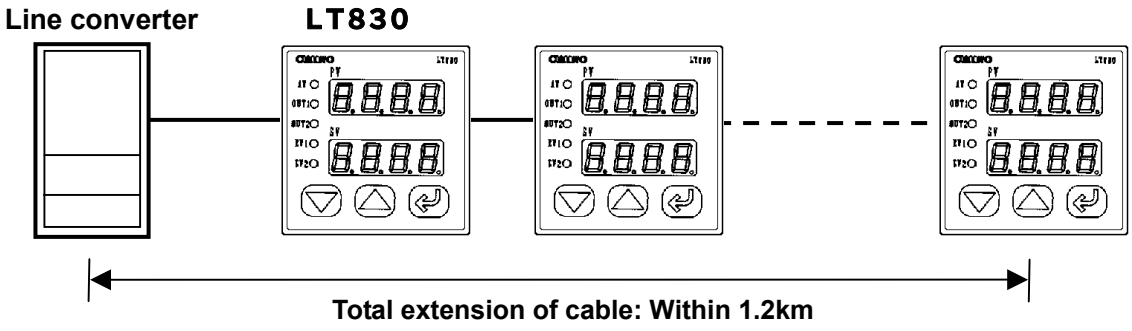
Green:SG
White:SB
Black:SA

The cable end is solder-plated.
Please prepare an extendable cable at your end.

Housing Parts Number : XHP-3(JST)
Recommendation wire size : AWG30 ~ 22

5.1.2 Total extension of RS-485 communications cable is up to 1.2km.

The wiring interval between each instrument is option, but the total extension distance of cable is within 1.2km.
(Line converter ⟺ the final end of LT830 controllers)

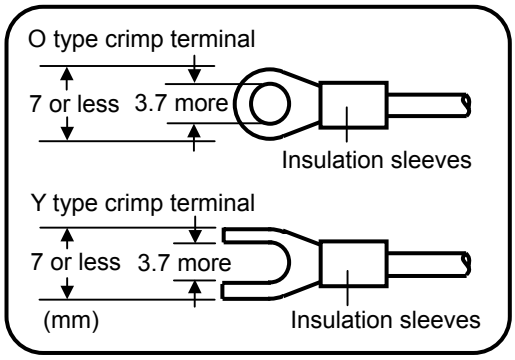


5.1.3 Noise preventive terminals

Separate the communications cable from drive power cables and other communications cable more than 50cm so as not to be affected by noises.

5.1.4 Crimp style terminals

Falling off of connections is one of communications failures. Terminate the communications cable with an [O] or [Y] type crimp style terminal having an insulation sleeve. (The terminal screws of LT830 controllers and line converters are M3.5mm.)



5.1.5 Mount an insulation resistor

For using the RS-485 communications interface, mount a 100 Ω resistor to the LT830 controller connected at the final end. (For details, see Section 5.3)

[A general metal film resistor can be used. The resistor (sold separately) is available at CHINO.]

5.1.6 Number of LT230 controllers connectable

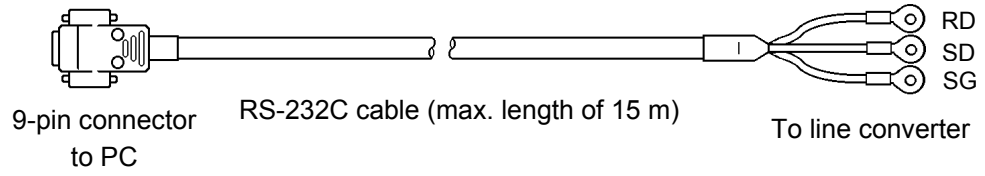
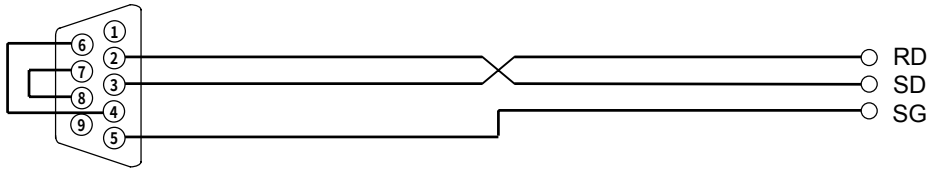
For RS-485: Up to 31 sets

5.2 Communications Cables

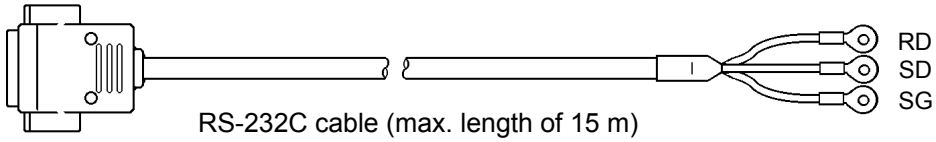
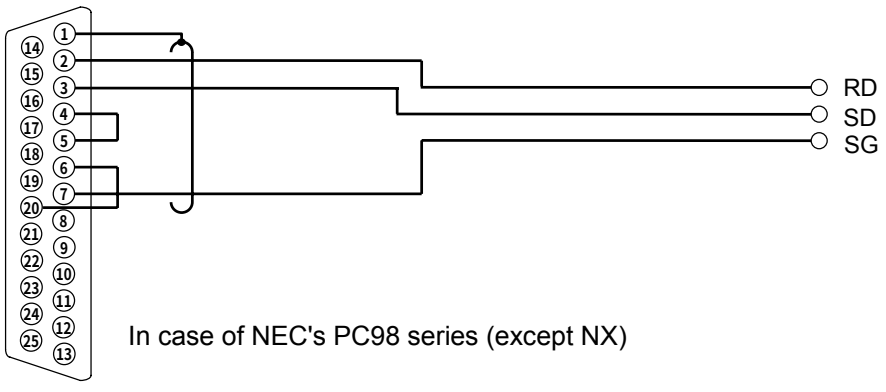
Make ready cables dedicated to communications before performing connection. Dedicated communications cables (sold separately) are available at CHINO.

5.2.1 Communications cables for RS-232SC（Between PC and line converter）

(1) Connection between PC (with 9 pin-terminal) and line converter.

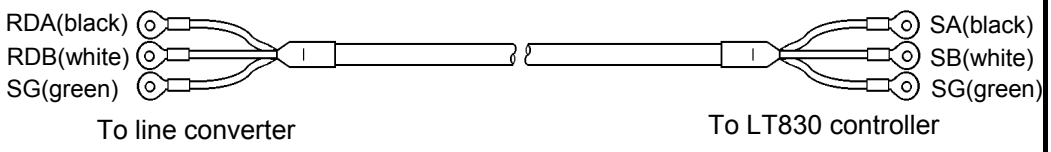
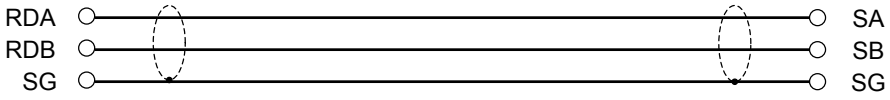
Cable	9-pin connector ⟷ RS-232C cable with O-shaped crimp terminal
Style	
Internal wiring	
Type code	**RZ-CRS6**□□ Cable length of 1 to 15 m (To be specified)

(2) Connection between PC (with 25 pin-terminal) and line converter.

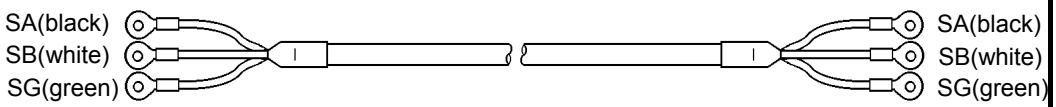
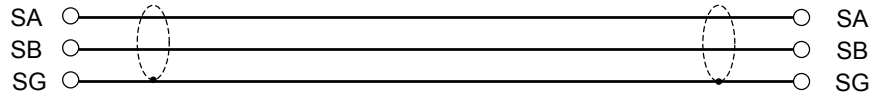
Cable	25-pin connector ⟷ RS-232C cable with O-shaped crimp terminal
Style	
Internal wiring	
Type code	**RZ-CRS2**□□ Cable length of 1 to 15 m (To be specified)

5.2.2 Communications cables for RS-485

(1) Connection between line converter and LT830 controller

Cable	O-shaped crimp terminal ⟷ RS-485 cable with O-shaped crimp terminal (for line converter)
Style	 To line converter　　To LT830 controller The cable consists of a twisted dual-core CVVS wires with SG (signal grounding) wire at both ends. Cut off the SG wire on the line converter side because this has no SG terminal.
Internal wiring	
Type code	**RZ-LEC** □□□ (for line converter) Cable length of 1 to 200 m (To be specified)

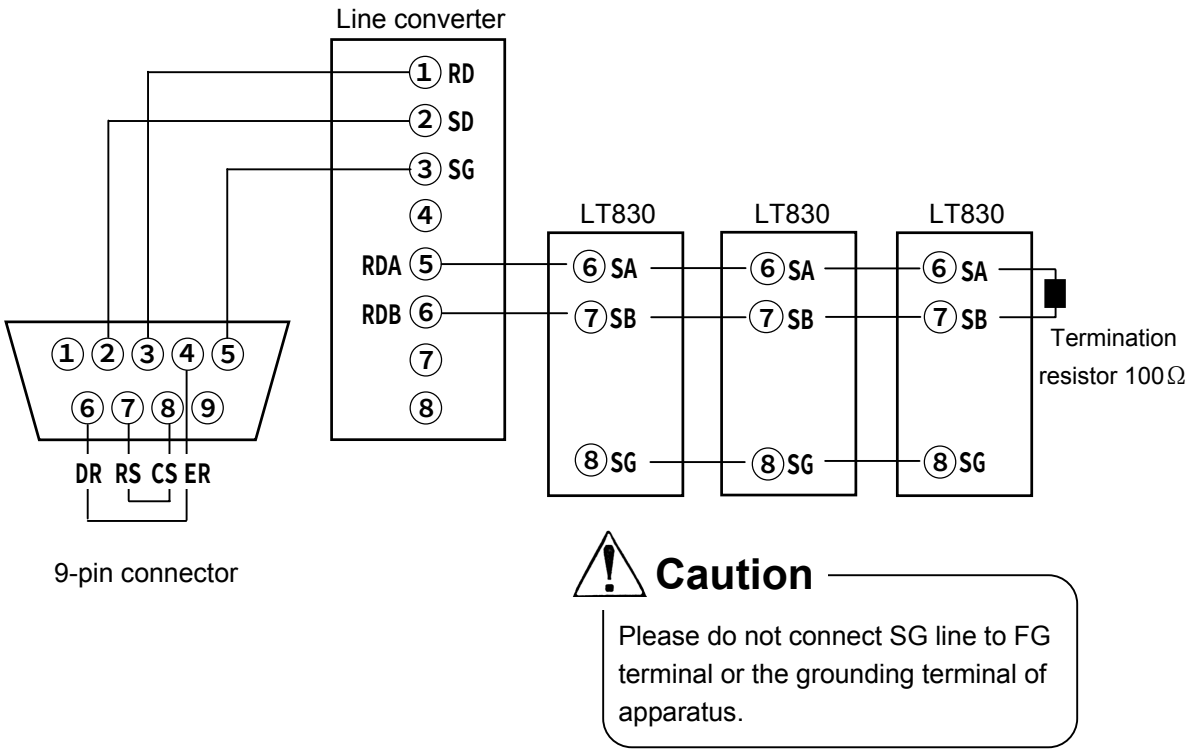
(2) Connection between LT830 controller and LT230 controller

Cable	O-shaped crimp terminal ⟷ RS-485 cable with O-shaped crimp terminal (for parallel connection)
Style	 To LT830 controller　　To LT830 controller The cable consists of a twisted dual-core CVVS wires with SG (signal grounding) wire at both ends.
Internal wiring	
Type code	**RZ-CSS1Z2(0.2m) or RZ-LEC** □□□ (For parallel connection) Cable length of 1 to 200 m (To be specified)

5.3 RS-485 Connections

This paragraph describes the method of connecting the RS-485 communications interface to the personal computer by using the line converter (Model SC8-10: sold separately). Since the line converter and the personal computer use three control signal of Send, Receive and Signal ground only. Since general personal computers are controlled by control signals, the computer does not function by only connecting three signal cables without wiring processing inside the connectors. Wiring processing depends upon the control signals being controlled by the personal computer. For details, read the instruction manual for the personal computer used.
[please refer to the following instruction manual for line converter (Model SC8-10: sold separately).]

In case of RS-485



Caution

Please do not connect SG line to FG terminal or the grounding terminal of apparatus.

6 MODBUS PROTOCOL

Basic Procedures of Communications and Precautions

⚠ CAUTION

1. When you set the parameter (writing), set the key lock at first.

LT830 controller is always ready for communication. It responds at anytime to data requests from the personal computer. However, when you set the parameter or control the instrument from the personal computer, you need to set key lock (lock 3) on LT830 at first. Key lock will be set at the front key on LT830 or through personal computer communication. If you try to set the parameter or operate instrument from the personal computer at Non-lock, Lock1, Lock 2 LT830 will show error code 12H.（See paragraph 6.6）

2. Take care of command re-transmission as there is no control signal line in use.

Since the LT830 controllers' serial interfaces communicate freely without using any control line, a reception failure may occur under some conditions. Exercise care when resending a command.

3. Don't disconnect or short any cables or instruments constituting the serial interface, or turn the power on or off during communications.

Don't disconnect or short any cables or instruments constituting the serial interface, or turn the power on or off during communications, or the operation may stop or lead to a malfunction. When this happens, all the components of the serial interface must be reset to repeat the operation from the beginning.

4. Send the next command after making sure that the communications drive has been turned off.

For RS-485 communications interface, multiple instruments are connected to the same communications line, only one instrument, of which instrument No. is specified by the PC, drives the communications line. The communications drive is turned off at a certain time (approx. 5 msec) after sending the last character so that all the characters are safely received by the personal computer. If the PC sends a command to the next unit before the communications drive is turned off, signals interfere with each other resulting in some communication failure. Exercise caution when you use a high-speed PC.

6.1 Message Transmission Modes

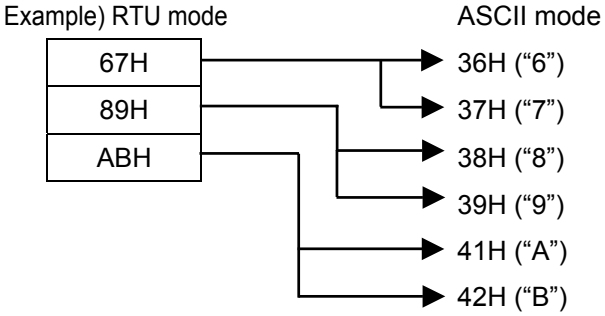
There are two modes of message transmission, RTU (Remote Terminal Unit) and ASCII, which can be selected by key programming.

(Table 1 Comparison between RTU and ASCII modes)

Item		RTU mode	ASCII mode
Interface		RS-485	
Interface(At the time of line converter use)		RS-232C, RS-422A	
Communications system		Half-duplex start-stop synchronization	
Transmission speed		9600, 19200bps	
Character code		Binary	ASCII
Error check	Vertical	Parity	
	Horizontal	CRC-16	LRC
Character Configuration	Start bit	1 bit	
	Data bit	8 bits	7 bits, 8 bits
	Parity bit	Disabled, even, odd	Disabled, even, odd
	Stop bit	1, 2 bit	
Message start code		None	: (Colon)
Message stop code		None	CR, LF
Data time interval		28 bit-time or less	1 second or less

6.1.1 Transmitted data

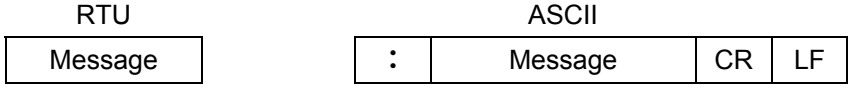
The RTU-mode data is transmitted in binary numbers. In ASCII mode, the 8-bit binary data of RTU is separated into higher-order 4 bits and lower-order 4 bits and both are turned into characters (0 - 9, A - F).



Length of the RTU-mode message is half that of an ASCII-mode message, ensuring a more efficient transmission.

6.1.2 Message frame configuration

The RTU-mode data consists only of a message section.
The ASCII mode data consists of a start character [: (colon, 3AH)], a message and a stop character [(CR (carriage return, 0DH) + LF (line feed, 0AH)].



The ASCII mode has the advantage of easier troubleshooting because its message has a start character [:].

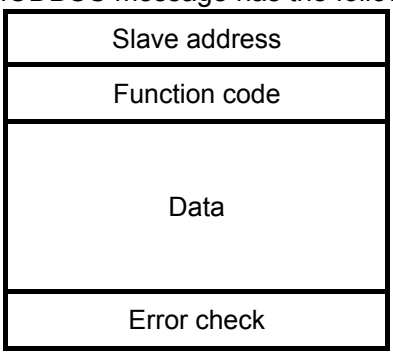
6.2 Data Time Interval

In RTU mode: Below 28 bit-time (2.9 msec. at 9600 bps, 1.4 msec. at 19200 bps)
In ASCII mode: Below 1 second

When sending a message, keep the time interval of data constituting one message not longer than the time specified above. When the time interval of data is longer than the above, the receiver unit (i.e., this controller) recognizes that the data transmission from the sending unit is complete, so that the subsequent data is processed as an abnormally received message.
While the message characters must be consecutively sent in RTU mode, the ASCII modes allows for a maximum interval of 1 second between characters, making it possible to use a master unit (PC) with a relatively slow processing speed.

6.3 Message Configuration

The MODBUS message has the following configuration in both RTU and ASCII modes.



6.3.1 Slave address

The slave address can be programmed in advance by key operation within a range between 1 and 99. The master unit usually communicates with one slave unit. While messages from the master unit are received commonly by all the units in connection, only the slave unit corresponding to the slave address included in the command message responds to the message sent.
The slave address “0” is used for a message from the master unit addressed to all the slave units (broadcast message). The slave units do not send a response back to the master unit.

6.3.2 Function code

Function codes refer to the functions to be executed by the slave units. The data is generally classified as follows. Refer to the reference table for details.

(1) Digital parameters : AT Start
(2) Digital input data : Event status.
(3) Analog parameters : Information on various parameters. Numerical values should be kept within the 16-bit range between –32768 and 32767 (see the reference table for details).
(4) Analog input data : Information on measured data. Numerical values within the range of 16-bits are delivered as an output.

(Table 2. Function code table)

Code	Functions	Unit	MODBUS original functions (ref.)
01	Read digital (ON/OFF) parameter	1 bit	Read coil status
02	Read digital input data	1 bit	Read input relay status
03	Read analog parameter	16 bit	Read hold register contents
04	Read analog input data	16 bit	Read input register contents
05	Write digital parameter	1 bit	Change single coil status
06	Write analog parameter	16 bit	Write into single hold register
08	Send received data (for diagnosis)		Loop-back test
15	Write two or more digital parameters		Change multiple coils status
16	Write two or more analog parameters		Write into multiple hold registers

6.3.3 Data section

Data configurations depend on the function codes. A master request consists of the code number of the data to be read or written (Relative No. to be calculated from the Reference No. described below) and the number of data pieces. Response from slave units consists of data responsive to the request. Every MODBUS basic data consists of 16-bit integers, with or without codes depending on individual data. It is thus configured as integers with their decimal places assigned to separate addresses, or normalized with the upper and lower limits specified by the scale with fixed decimal places. The LT830 controllers employ the system of assigning the decimal places to separate addresses.

6.3.4 Reference Nos.

Data in the LT830 controllers have “Reference No. ” assigned to each of them which is required for reading and writing the data. The data in the LT830 controllers are classified into “Digital parameter”, “Digital input data”, “Analog input data”, “Analog parameter” depending on their type. The Nos. In the message are designated by the “Relative Nos. ” corresponding to the Reference Nos.

(Table 3. Reference Nos. and Relative Nos.)

Data type	Reference No.	Relative No.	MODBUS original (for reference)
Digital parameter	1 to 10000	Reference No. – 1	Coil
Digital input data	10001 to 20000	Reference No. – 10001	Input relay
Analog input data	30001 to 40000	Reference No. – 30001	Input register
Analog parameter	40001 to 50000	Reference No. – 40001	Hold register

Example) “The Relative No” of the measured value (PV) at “Reference No. 30101” is “100”.

(Table 4. Quick search table for LT830-Reference Nos.)

Data type	Parameters	Reference No	Corresponding function code	Reference table
Digital parameter	AT start	101	01(READ) 05(WRITE)	Section 6.7.3 （P.33）
Digital input data	Error status Event status	10004 to 10120	02(READ)	Section 6.7.4 （P.33）
Analog input data	Measured data(PV/SV/MV) Run parameter Event status	30101 to 30142	04(READ)	Section 6.7.2 (P.32 to 33)
Analog parameter	Set up parameter Linear decimal point	40008		Section 6.7.1 (P.30)
	One kind parameter Control output at Ready SV rise ramp SV fall ramp PV start	40114 to 40119		(P.30)
	No.1/No.2parameter SV EV1/EV2 setting P/I/D Output limiter L Output limiter H Variation limiter	40201 to 40211	03(READ) 06(WRITE) 16(WRITE)	(P.30 to 31)
	Instrument Operation Key lock Run/ready Remote SV setting	49501 to 49512		(P.31 to 32)

6.3.5 Error check

Error check for transmission frames is different between the transmission modes.
RTU mode: CRC-16
ASCII mode: LRC

6.3.5.1 Calculation of CRC-16

In the CRC system, the information to be transmitted is divided by a generating polynomial, the resulting remainder being added to the end of the data. The generation polynomial is as follows.

$$1 + X^2 + X^{15} + X^{16}$$

The data from its slave address to its end is calculated in the following procedure.

1) Initialize the CRC-16 data (assumed as X) (= FFFFH)
2) Exclusive logical sum (EX–OR) between data 1 and X $\rightarrow$ X
3) Shift X one bit to the right $\rightarrow$ X
4) When a carry is generated, take A001H and EX–OR. If not, go to 5). $\rightarrow$ X
5) Repeat 3) and 4) until shifting 8 times.
6) EX–OR between the next data and X $\rightarrow$ X
7) Same as 3) to 5)
8) Repeat up to the last data
9) Create a message in the sequence from lower to upper orders of the calculated 16-bit data (X).

Example) Since CRC-16 is 1241H for the data | 02H | 07H | , the error check data will be | 41H | 12H |.

Example1) C programming

```
/***** CRC-16 PROGRAM *****/
#include <stdio.h>
#include <conio.h>

void main(void)
{
        unsigned int    iLoopCnt;
        unsigned short  usData;
        unsigned short  usCrcData;
        unsigned short  usErrChkData;
        int             iDummy;

        usCrcData = 0xffff;

        printf("Hex DATA ([q] = END) >\n");
        while( scanf("%x",&usData) != 0 ){
                usCrcData =  usData ^ usCrcData;

                for( iLoopCnt = 0 ; iLoopCnt < 8 ; iLoopCnt++ ){
                        if( usCrcData & 0x0001 ){
                                usCrcData = usCrcData >> 1;

                                usCrcData = usCrcData ^ 0xa001;
                        }else
                                usCrcData = usCrcData >> 1;
                        }
                }
        }

        printf( "CRC-16 DATA = %xH \n", usCrcData );

        usErrChkData = ( usCrcData >> 8) | ( usCrcData << 8 );
        printf( "CRC-16 ENDIAN DATA = %xH", usErrChkData );

        iDummy = getch();
}
```

Example2) Microsoft Visual Basic.NET

```
'***** CRC-16 PROGRAM *****
Function fncCalcCrc(ByVal sCharData As String) As String

    Dim intLoopCnt1 As Integer
    Dim intLoopCnt2 As Integer
    Dim intLoopCnt3 As Integer
    Dim intSdChrCnt As Integer
    Dim intCrc16Data As Integer
    Dim intCarryFlg As Integer
    Dim strWorkBuf As String
    Dim int2ByteImage(16) As Integer
    Dim strCRC As String


    intSdChrCnt = Len(sCharData) / 2
    intCrc16Data = &HFFFF


    For intLoopCnt1 = 1 To intSdChrCnt
        intCrc16Data = intCrc16Data Xor Val("&H" + Mid(sCharData, 2 * intLoopCnt1 - 1, 2))
        For intLoopCnt2 = 0 To 7
            For intLoopCnt3 = 0 To 15
                If (intCrc16Data And (2 ^ intLoopCnt3)) = 0 Then
                    int2ByteImage(intLoopCnt3) = 0
                Else
                    int2ByteImage(intLoopCnt3) = 1
                End If
            Next intLoopCnt3
            intCarryFlg = int2ByteImage(0)
            For intLoopCnt3 = 1 To 15
                int2ByteImage(intLoopCnt3 - 1) = int2ByteImage(intLoopCnt3)
            Next intLoopCnt3
            int2ByteImage(15) = 0

            intCrc16Data = 0
            For intLoopCnt3 = 0 To 15
                intCrc16Data = intCrc16Data + ((2 ^ intLoopCnt3) * int2ByteImage(intLoopCnt3))
            Next intLoopCnt3

            If intCarryFlg = 1 Then
                intCrc16Data = intCrc16Data Xor &HA001
            End If
        Next intLoopCnt2
    Next intLoopCnt1


    strWorkBuf = "0000"
    Mid(strWorkBuf, 5 - Len(Hex(intCrc16Data))) = Hex(intCrc16Data)
```

```
    strCRC = Mid(strWorkBuf, 3, 2) & Mid(strWorkBuf, 1, 2)
    '   (example)1241H-->'41H''12H'

    Return strCRC

End Function
```

6.3.5.2 Calculation of LRC

The data from its slave address to its end is calculated in the following procedure.

1) Create a message in RTU mode.
2) Add the start (slave address) to end of the data. → X
3) Complement X (bit reverse) → X
4) Add 1 (X = X + 1)
5) Add X as an LRC to the end of the message.
6) Convert the whole data to ASCII characters.

Example) For the data | 02H | 07H | , LRC is F7H which will be | 02H | 07H | F7H | as a binary message, so that the ASCII message will be | 30H | 32H | 30H | 37H | 46H | 37H | .

Example1) C programming

```
/***** LRC PROGRAM *****/
#include <stdio.h>
#include <conio.h>

void main(void)
{
        unsigned short   usData;
        unsigned short   usLrcData;
        int              iDummy;

        usLrcData = 0;

        printf("Hex DATA ([q] = END) >\n");
        while( scanf("%x",&usData) != 0 ){
                usLrcData += usData;
                usLrcData = usLrcData & 0xff;
        }

        usLrcData = usLrcData ^ 0xff;

        usLrcData = usLrcData++;
        usLrcData = usLrcData & 0xff;

        printf( "LRC-16 DATA = %xH \n", usLrcData );

        iDummy = getch();
}
```

Example2) Microsoft Visual Basic.NET

```
'***** LRC PROGRAM *****

Function fncCalcLrc(ByVal sCharData As String) As String

    Dim intLoop As Integer
    Dim intStrSize As Integer
    Dim intLRC As Integer
    Dim sRtnLrc As String

    intStrSize = Len(sCharData) / 2

    intLRC = 0

    For intLoop = 1 To intStrSize
        intLRC = intLRC + Val("&H" + Mid(sCharData, 2 * intLoop - 1, 2))
    Next intLoop
    sRtnLrc = Hex(&HFF - intLRC + 1)
    sRtnLrc = Microsoft.VisualBasic.Right("0" & sRtnLrc, 2)

    Return sRtnLrc

End Function
```

6.3.6 Precautions on data processing

(1) Since the measured data and decimal places are assigned to separate numbers, it is necessary to use both parts of the information when playing back the data. The decimal place of each data is showed on the reference table. There are some types of data, which are fixed decimal place, followed by each measuring range, (See paragraph 6.8), followed by linear decimal place setting. Pay attention to the decimal place, when you replay the data.

(2) Since data is accessible (changeable) one by one, care must be taken when programming related data, for instance when initializing related data by changing the range number. Processing details are given in the Reference No. list.

(3) Read or write the data within the range of Reference Nos. specified. If data is written on any non-specified Reference No., it is likely to affect the proper operation of the instruments.

(4) While it is possible to write data on two or more discreet Reference Nos., a start number with Reference No. not specified will result in an error (error No. 02H).

(5) When reading two or more Reference Nos., the data with non-specified Reference No. becomes “0”.

(6) When an error is detected during writing on two or more Reference Nos., all the programming becomes invalid.

6.4 Creating a Message

A message consists of (1) Slave address, (2) Function code, (3) Data section and (4) Error check code. (See Section 6.3)

The message readable or writable at one time is within the following range.

function code	Number of data pieces	
	ASCII	RTU
01	1	1
02	121	121
03	13	26
04	13	26
15	1	1
16	11	26

How to create a message will be described by an example given below.
Example) Reading a measured data for LT830 controller with “slave address 02”.

6.4.1 RTU mode message

(1) Slave address : 02 (02H)

(2) Function code : 04 (04H)

The data type is “Read analog input data (read input register contents) ”. When the function code is “04”, specify the “data's Relative No. by 2 bytes” and the “number of data pieces by 2 bytes” to be read from the data section. (See Section 6.5. See Section 6.5.4 for "Function code: 04".)

* It is necessary to make sure of the number of bytes of data.

(3) Data section :

Starting Relative No. 100 (00H | 64H) and Number of data pieces 2 (00H | 02H)

Measured data (analog input data) are stored in Reference Nos. “30001 to 40000” (See Table 3 in Section 6.3.4). The reference table shows that the Measured data (PV) is stored in “30101” and the PV states in “30102”. (See Section 6.7. See Section 6.7.2 for reading the measured data.)

The relative No. of the starting “Reference No. 30101” is 30101 - 30001 = 100 that can be expressed by 2 bytes “ 00H | 64H ”.(See Table 3 of Section 6.3)

The number of data pieces to be read is “2” of the Measured data (PV) and the PV status which can be expressed by “ 00H | 02H ” in 2 bytes

(4) Error check: 2730H calculated with CRC-16 (30H | 27H)

Error check in RTU mode is calculated with CRC-16. (See Section 6.3.5.1)

The data in the core message is:

| 02H | 04H | 00H | 64H | 00H | 02H | according to (1) to (3), whose CRC-16 is 2730H.

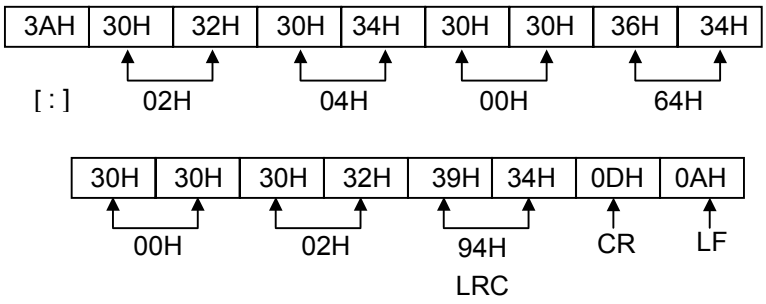
Error check data is therefore | 30H | 27H | .

(5) Message: | 02H | 04H | 00H | 64H | 00H | 02H | 30H | 27H |

Create a message according to the message configuration. (See Section 6.3)

6.4.2 Message in ASCII mode

Calculate the error check LRC from the core message. (See Section 6.4.1 (4)). LRC is 94H (See Section 6.5.3.2). Each data in the core message is converted to ASCII code. LRC is also converted to ASCII code to be added to the core message. Add a message starting character “ : ” and “CR” and “LF” to the end of the message.



6.5 Function Code

Responses by function code are given below. (See Table 2. Function code table in Section 6.3.2)
Note) See Section 6.6 for responses in abnormal status.

6.5.1 Read digital parameter (read coil status)

[Function code: 01 (01H)]
The specified number of “digital (ON/OFF) parameters” are read out consecutively commencing with the designated Reference No. For ON/OFF data, 8 Reference Nos. are placed in each data (1 byte) sequentially in number to constitute the response message data. The LSB (D0 side) of each data becomes the digital data with the smallest number. If the number of Reference No. is anything other than a multiple of 8, an unnecessary bit becomes 0.

Example) Reading 1 Reference Nos. 101of digital parameters for the slave unit 2.

Reference No	101
Data	OFF

AT stop

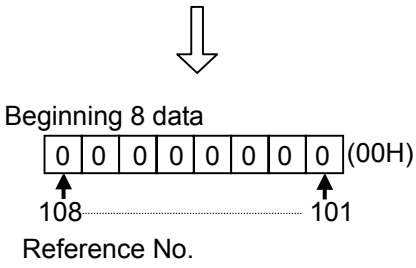
(RTU mode)

Master→Instruments

Slave address	02H
Function code	01H
Start No. (H)	00H
Start No. (L)	64H
Number of Reference No. (H)	00H
Number of Reference No. (L)	01H
CRC (L)	BCH
CRC (H)	26H

Instruments→Master (normal)

Slave address	02H
Function code	01H
No. of data	01H
1st 8 data	00H
CRC (L)	51H
CRC (H)	CCH



(Error check in ASCII mode)
The error check: CRC (L) and CRC (H) will be as follows.

LRC	98H

LRC	FCH

Note) Start No. (Relative No.) is given by “Reference No. - 1”.
(Decimal number 100 (=101 – 1) ⟶ Hexadecimal 64H)
Note) No. of data is the number of data bytes.
(which is different from the requested number of Reference No. In the example given above, the requested number of Reference No. is1 and the number of data is 1).

6.5.2 Read digital input data (read input relay status)

[Function code. 02 (02H)]
The specified number of “digital (ON/OFF) input data” are read out consecutively commencing with the designated Reference No. For ON/OFF data, 8 Reference Nos. data are placed in one data (1 byte) sequentially in number to constitute the response message data. The LSB (on DO side) of each data is a digital data with the smallest number. If the number of Reference No. read is anything other than a multiple of 8, an unnecessary bit becomes 0. An example of response message is the same as in “Function code 01”, though its start number (Relative No.) is “Reference No. –10001”.

6.5.3 Read analog set value (read hold register contents)

[Function code. 03 (03H)]

The specified number of “analog parameters (2 bytes: 16 bits)” is read out consecutively commencing with the designated Reference No. The data is split into higher-order 8 bits and lower-order 8 bits arranged sequentially in number to constitute a data of response message

Example) Reading P/I/D of the slave unit 2.

(Reading 3 Reference Nos. from 40206 to 40208 of analog parameters for the slave unit 2)

Reference No.	40206	40207	40208
Data	30 (001EH)	120 (0078H)	20 (0014H)

← Example: P=3.0%,I=120Sec,D=20Sec.

(RTU mode)

Master→Instruments

Slave address	02H
Function code	03H
Start No. (H)	00H
Start No. (L)	CDH
Number of Reference No. (H)	00H
Number of Reference No. (L)	03H
CRC (L)	94H
CRC (H)	07H

Instruments→Master (normal)

Slave address	02H
Function code	03H
No. of data	06H
P data (H)	00H
P data (L)	1EH
I data (H)	00H
I data (L)	78H
D data (H)	00H
D data (L)	14H
CRC (L)	1DH
CRC (H)	91H

(Error check in ASCII mode)

LRC	2BH

LRC	4DH

Note) Start No. (Relative No.) is given by “Reference No. - 40001”.
(Decimal number 205 (=40206–40001)⟶ Hexadecimal CDH)

Note) No. of data is the number of data bytes. (which is different from the requested number of data. In the example given above, the requested number of reference No. is 3 and the number of data is 6).

Note) The number of data of a message receivable at one time (that can be transmitted from the controller) is limited. (See Section 6.4).

6.5.4 Read analog input data (read input register contents)

[Function code. 04 (04H)]

The specified number of “analog input (2 bytes: 16 bits)” are read out consecutively commencing with the designated Reference No. The data is split into higher-order 8 bits and lower-order 8 bits arranged sequentially in number to constitute a data of response message. The response example is the same as in “Function code 03”, though its start number (Relative No.) is “Reference No. – 30001”.

6.5.5 Write digital parameter (Change single coil status)

[Function code: 05 (05H)]

A digital parameter with specified numbers is brought into specified status (ON/OFF).

Example) Executing the AT on the slave unit 2 (Turn on Reference No. 101 of digital parameter for the slave unit 2)

(RTU mode)

Master→Instruments		Instruments→ Master (normal)	
Slave address	02H	Slave address	02H
Function code	05H	Function code	05H
Parameter No. (H)	00H	Parameter No. (H)	00H
Parameter No. (L)	64H	Parameter No. (L)	64H
Programming status (H)	FFH	Programming status (H)	FFH
Programming status (L)	00H	Programming status (L)	00H
CRC (L)	CDH	CRC (L)	CDH
CRC (H)	D6H	CRC (H)	D6H

(Error check in ASCII mode)

LRC	96H	LRC	96H

Note) The response is the same as command message in the case of a normal response.
Note) Parameter No. (Relative No.) is given by “Reference No. – 1”.
(Decimal number 100 (=101–1) ⟶ Hexadecimal 64H)
Note) Upon execution, program “FF00H”. The program “0000H” is for terminating the AT command.
Note) When the slave address is programmed to “0”, all the slave units execute this command, although no response is received from any of them.

6.5.6 Write analog parameter (write into a single hold register)

[Function code: 06 (06H)]

An analog parameter with specified numbers is brought into a specified value.

Example) Programming the variation limiter of the slave unit 2 to 50.0 %.
(Program Reference No. 40211 of analog parameter to “500” for the slave unit 2.)

(RTU mode)

Master→Instruments		Instruments→Master (normal)	
Slave address	02H	Slave address	02H
Function code	06H	Function code	06H
Parameter No. (H)	00H	Parameter No. (H)	00H
Parameter No. (L)	D2H	Parameter No. (L)	D2H
Programming status (H)	01H	Programming status (H)	01H
Programming status (L)	F4H	Programming status (L)	F4H
CRC (L)	29H	CRC (L)	29H
CRC (H)	D7H	CRC (H)	D7H

(Error check in ASCII mode)

LRC	31H	LRC	31H

Note) The response is the same as command message in case of normal response.
Note) Parameter No. (Relative No.) is given by “Reference No. – 40001”. (Decimal number 210 (=40211 – 40001) ⟶ Hexadecimal D2H)
Note) When the slave address is programmed to “0”, all the slave units execute this command, though with no response received from any of them.

6.5.7 Loop back test

[Function code: 08 (08H)]

Check transmission between master and slave units. Response is made according to a specified diagnosis code. With the diagnosis code fixed at “0000H”, the LT830 controller performs a “return check” of unaltered received data transmissions.

Example) Executing “Loop back test” on the slave unit 2.

(RTU mode)

Master→Instruments

Slave address		02H
Function code		08H
Diagnosis code (H)	Fixed	00H
Diagnosis code (L)		00H
Arbitrary data		*
Arbitrary data		*
CRC (L)		*
CRC (H)		*

Instruments→ Master (normal)

Slave address		02H
Function code		08H
Diagnosis code (H)	Fixed	00H
Diagnosis code (L)		00H
Received data		*
Received data		*
CRC (L)		*
CRC (H)		*

6.5.8 Write multiple analog parameters (Change multiple coils status)

[Function code: 15 (0FH)]

According to the specified number, programs the specified number of digital parameters into the specified status (OFF or ON). Every 8 specified numbers which are OFF or ON form a single data unit (byte). The LSB of each data unit (D0) is the digital data of the smallest numbered data. When the number of specified numbers is not a multiple of 8, the unnecessary bits are ignored.

Example) Executing the AT on the slave unit 2.

(Program the data of the digital parameters with reference numbers 101 of the slave unit 2 as shown below.)

Reference No.	101
Data	ON

AT start

(RTU mode)

Master→Instruments

Slave address	02H
Function code	0FH
Start No. (H)	00H
Start No. (L)	64H
Number of Reference No. (H)	00H
Number of Reference No. (L)	01H
Number of data	01H
1st 8 data	01H
CRC (L)	DEH
CRC (H)	8AH

Instruments→Master (normal)

Slave address	02H
Function code	0FH
Start No. (H)	00H
Start No. (L)	64H
Number of Reference No. (H)	00H
Number of Reference No. (L)	01H
CRC (L)	D5H
CRC (H)	E7H

(Error check in ASCII mode)

LRC	88H

LRC	8AH

Note) Start No. (Relative value) is given by “Reference No. – 1”. (Decimal number 100 (=101 – 1) → Hexadecimal 64H)

Note) When the slave address is programmed to “0”, all the slave units execute this command, although no response is received from any of them.

Note) The number of message data transmittable (receivable by this controller) at one time is limited. (See Section 6.4)

6.5.9 Write multiple analog parameters (write into multiple hold registers)

[Function code: 16 (10H)]

A specified number of analog parameters from designated numbers are programmed to specified values. The data is split into higher-order 8 bits and lower-order 8 bits to be sent sequentially in number.

Example) Programming the P/I/D of the slave unit 2 to P=12.0%,I=90Sec,D=25Sec.
(Program 3 Reference Nos. from 40206 to 40208 of analog parameters for the slave unit 2)

Reference No.	40206	40207	40208
Data	120 (0078H)	90 (005AH)	25 (0019H)

(RTU mode)

Master→Instruments

Slave address	02H
Function code	10H
Start No. (H)	00H
Start No. (L)	CDH
Number of Reference No. (H)	00H
Number of Reference No. (L)	03H
Number of data	06H
1st data (H)	00H
1st data (L)	78H
2nd data (H)	00H
2nd data (L)	5AH
3rd data (H)	00H
3rd data (L)	19H
CRC (L)	36H
CRC (H)	56H

Instruments→Master (normal)

Slave address	02H
Function code	10H
Start No. (H)	00H
Start No. (L)	CDH
Number of Reference No. (H)	00H
Number of Reference No. (L)	03H
CRC (L)	11H
CRC (H)	C4H

(Error check in ASCII mode)

LRC	2DH

LRC	1EH

Note) Start No. (Relative value) is given by “Reference No. – 40001”. (Decimal number 205 (=40206 – 40001) → Hexadecimal CDH)

Note) When the slave address is programmed to “0”, all the slave units execute this command, although no response is received from any of them.

Note) The number of message data transmittable (receivable by this controller) at one time is limited. (See Section 6.4)

6.6 Processing in Abnormal Status

The following response is given when any problem is found in the content of a message from the master unit.

6.6.1 Case of no response

The message is ignored with no response given when

(1) A transmission error (overrun, framing, parity , CRC or LRC) is detected in the message;

(2) The slave address in the message is not the receiver's own address;

(3) Data interval in messages is too long;

28 bits or more in RTU mode

1 second or more in ASCII mode

(4) Transmission parameters are not consistent with those of the receiver;

(5) The bytes of the received message exceeds 64.

Note) When the slave address is “0” in the write function, the message is executed unless any error is detected in it, but with no response given to it. Since no response is given also when the above error is detected in the message, whether it is normal or abnormal can not be judged by the response from this controller when the slave address is “0”.

6.6.2 Response error message

If the following failure is detected in a message from the master unit with no error specified in Section 6.6.1, the code indicating the error is responded as an “error message”.

The error message format is as follows.

Slave address
Function code + 80H
Error code
CRC(L)
CRC(H)

Function code	Function code + 80H
01	81H
02	82H
03	83H
04	84H
05	85H
06	86H
08	88H
15	8FH
16	90H

Error codes are as follows.

Error code	Description
01H	Function code failure When receiving an unspecified function code
02H	Relative No. (Reference No.) failure When the start No. or parameter No. received is not the specified number.
03H	Data pieces failure The number of data pieces to be transmitted in response to the message received exceeds a specified number. (See Section 6.4)
11H	Not in the programming range ● When the number is set beyond the range of the reference table. ● When L/H of the output limit is set reversed.
12H	Programming disabled ● When you try to set or operate at unlock, lock 1, lock 2. ● When AT is set to start at “Ready”. ● When AT is positioned at 2 controls. ● When you write at the linear decimal point. ● When you write at the remote SV in a local condition (DI=OFF).

6.7 Table of LT830-Reference

6.7.1 Analog parameters

(1) Setup Parameter

FNC code ……Applicable function code,R/W……R:READ, W:WRITE

Reference No.	FNC code	R/W	DATA	Setting range (communication range)	Default	Details
40008	03	R	Linear decimal point (SV DOT)	0 to 3	1	Reading is Okay, but not writing. It reads fixed number according to the each range at temperature input. It reads linear decimal point at linear input.

(2) One kind parameter

FNC code ……Applicable function code,R/W……R:READ, W:WRITE

Reference No.	FNC code	R/W	DATA	Setting range (communication range)	Default	Details
40114	03 06 16	R W W	Control output at Ready	–5.0 to 105.0 (–50 to 1050)	0.0%	Set the output value at ready condition.
40116 40117	03 06 16	R W W	SV rise ramp SV fall ramp	0 to 9999	0	0= no inclining Decimal place: Temperature input = Set by the each range Linear input = Set linear decimal point
40119	03 06 16	R W W	PV start ON/OFF	OFF/ON (0/1)	OFF (0)	Choose [On] to start or [OFF] not to start inclining by PV. 0=OFF 1=ON

(3)No.1/No.2 Parameter

FNC code ……Applicable function code,R/W……R:READ,W:WRITE

Reference No.	FNC code	R/W	DATA	Setting range (communication range)	Default	Details
40201	03 06 16	R W W	SV	–1999 to 9999 (SV limit Min. to SV limit Max.)	0	Decimal place: Temperature input = Set by the each range Linear input = Set linear decimal point
40202 40203	03 06 16	R W W	EV1 setting EV2 setting	–1999 to 9999		When the event mode is an absolute value, deviation value, or fail: Decimal place: Temperature input = Set by the each range Linear input = Set linear decimal point
				0 to 9999 (0 to 9999)		When the event mode is a deviation High/low limits: Decimal place: Temperature input = Set by the each range Linear input = Set linear decimal point
				0.0 to 50.0 (0 to 500)		When the event mode is in a heater disconnection.

FNC code ……Applicable function code,R/W……R:READ,W:WRITE

Reference No.	FNC code	R/W	DATA	Setting range (communication range)	Default	Details
40202 40203					4000 –1999 4000 –1999 4000 0.0 —	When the event mode is changed, it will become default. Event mode= Absolute high limit Event mode= Absolute low limit Event mode= Deviation high limit Event mode= Deviation low limit Event mode= Deviation high/low limit Event mode= Heater disconnection Event mode= NON
40206	03 06 16	R W W	P	0.0 to 999.9 (0 to 9999)	3.0 %	0.0 = 2 position (On/off)
40207	03 06 16	R W W	I	0 to 9999	120 Sec	0 = infinity
40208	03 06 16	R W W	D	0 to 9999	20 Sec	0 = off
40209 40210	03 06 16	R W W	Output limiter L Output limiter H	–5.0 to 100.0 (–50 to 1000) 0.0 to 105.0 (0 to 1050)	0.0% 100.0%	It should be set L< H.
40211	03 06 16	R W W	Variation limiter	0.1 to 100.0 (1 to 1000)	100.0%	

(4)Instrument Operatopn

FNC code ……Applicable function code,R/W……R:READ,W:WRITE

Reference No.	FNC code	R/W	DATA	Setting range (communic range)	Default	Details
49501	03 06	R W	Keylock	UNLoCK/LoCK1/LoCK2 /LoCK3 0/1/2/3	LoCK1 (1)	0=Unlock 1=Keylock1 2=Keylock2 3=Keylock3
49510	03 06 16	R W W	Run/ready	Run/Ready (0/1)	RUN (0)	(Note) It can be switched by front keys or and external signals. The practical control follows to the final operation. (Communication, front control and external signals)

FNC code ……Applicable function code,R/W……R:READ,W:WRITE

Reference No.	FNC code	R/W	DATA	Setting range (communication range)	Default	Details
49512	03 06 16	R W W	Remote SV Setting	–1999to9999 (SV limit MIN to SV limit MAX)		When the communication function is COM, remote SV can be set by communication. R/L is switched by the external signals. In a Local condition, Writing is not accepted. Decimal place: Temperature input = Set by the each range Linear input = Set linear decimal point

6.7.2 Analog input data (only reading)

FNC code ……Applicable function code,R/W……R:READ, W:WRITE

Reference No.	FNC code	R/W	DATA	Details
30101	04	R	Measured value (PV)	Decimal place: Temperature input = Set by the each range Linear input = Set linear decimal point When PV is + over range 32767 When PV is – under range –32768
30102	04	R	PV status	0 = normal 1 = +Over range 2 = –Under range 4 = Input circuit hard is error.
30103	04	R	Setting value(SV)	Current adopted SV (fixed SV , inclining SV and remote SV etc.) Decimal place: Temperature input = Set by the each range Linear input = Set linear decimal point
30104	04	R	SV status	0= fixed value 1= remote SV 2= inclining working
30105	04	R	Control output value(MV1)	–50 to 1050=–5.0 to 105.0%
30106	04	R	MV1 status	0= auto 2= during AT execution output 3= control output at Ready
30107	04	R	Control output value(MV2) (cooling output)	–50 to 1050=-5.0 to 105.0% (*Invalid value can be read when there is no control output 2.)
30108	04	R	MV2 status (cooling output)	0= auto 3= control output at Ready (*Invalid value can be read when there is no control output 2.)

FNC code ……Applicable function code,R/W……R:READ, W:WRITE

Reference No.	FNC code	R/W	DATA	Details
30142	04	R	Event status	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 EV2 EV1 Event OFF→ 0 0 0 0 Event ON→ 0 1 0 1 during standby, event OFF → 1 0 1 0

6.7.3 Digital parameters

FNC code ……Applicable function code,R/W……R:READ, W:WRITE

Reference No.	FNC code	R/W	DATA	Setting range (communication range)	Default	Details
101	01 05	R W	Auto-tuning	Stop / Start (0/1=[FF00h]) in[ ], FNC code is at 05.	Stop (0)	The most appropriate PID constants will be calculated automatically and be set by the instrument by itself. 0=AT stop or End 1=AT start or AT Execution During “Ready”, cannot be execute. During 2-position control, cannot be execute

6.7.4 Digital input data (only reading)

FNC code ……Applicable function code,R/W……R:READ, W:WRITE

Reference No.	FNC code	R/W	DATA	Details
10004	02	R	RJ error	0= normal 1=RJ error occurrence
10117	02	R	Event1status	0= event OFF 1= event ON 0= during stand by,event OFF
10118			Event1status	0 0 1
10119			Event2status	0= event OFF 1= event ON 0= during stand by,event OFF
10120			Event2status	0 0 1

6.8 Table of Input types & SV_Dot

(Table 5. List of input types.)

Input type		Input type(ºC)			SV DOT
T/C	B	0	to	1820	0
	R	0	to	1760	0
	S	0	to	1760	0
	N	0	to	1300	0
	K1	–200	to	1370	0
	K2	–199.9	to	500.0	0
	E	–199.9	to	700.0	1
	J	–199.9	to	900.0	1
	T	–199.9	to	400.0	1
RTD	PT1001	–199.9	to	850.0	1
	PT1002	–199.9	to	200.0	1
Input type		Range			
DC voltage	5V	0.000	to	5.000	0 to 3

6.9 Remote SV Setting Through Communication

6.9.1 Digital communications between LT and LT controllers (Between CHINO's controller and LT)

Set the communication function at communications transmission [t r S] and communications remote [r e M], then digital communication can be done between LT and LT.（Details: See 7 Communications Transmission and Communications Remote）

In this case, LT8 that is set at communications remote will receive all types of remitted data (PV, SV) as remote SV.

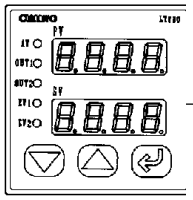
6.9.2 Personal Computer Communication + Communications Remote Functions

There will be the case that it needs to receive the remote SV in a connecting condition of personal computer and LT8 (communication function is CoM.). In this case, keep the communication function [CoM] and write SV data on reference no.49512 (remote SV), and it will become to receive remote SV with receiving measuring data between personal computer and LT8 and sending & receiving of parameter. This case of receiving data of remote SV works as remote SV data that is same as communications remote (6.9.1).

7 Communications Transmission and Communications Remote

7.1 General

Digital communications is available not only between LT8 controllers and personal computers but also between a LT8 controller and other LT8 controllers (LT8 controller and other CHINO's controllers). These functions are called "Communications Transmission" and "Communications Remote".
By programming a LT8 controller to Communication Transmission as a master unit and other LT8 controllers to Communications Remote as slave units, the SV of the slave unit (up to 31 sets) can be programmed through communications. This function can be utilized for using multiple LT8 controllers in the same condition. The master or slave designation of the LT8 controllers can be programmed by key operation.



When the communications function of the Mode 6 is programmed to "trS (Private protocol)" or "trS.2 (MODBUS)", LT8 controller is programmed as a master unit (Communications Transmission).

When the communications function of the Mode 6 is programmed to "rEM (in case a master LT8 controller is programmed to "trS")"or "CoM (in case a master LT8 controller is programmed to "trS.2")", LT8 controller is programmed as a slave unit (Communications Remote).

[Communications Functions Programs and Details of Transmission Data]

Mode 6, Communications Functions Programming (Master) → (Slave)	Details of Transmission Data
(trS. 2) → (CoM) [LT8] (N.B.) Slave unit has to be programmed to Key Lock 3 to receive data.	• Sending (Master) and receiving (Slave) Remote SV data and Execution SV No. and Run/Ready. • MODBUS protocol

7.2 Communications Specifications

* Communication system : Start-stop synchronization system
* Transmission speed : 19200, 9600 bps (selectable)
* Start bit : 1 bit
* Data length : 7bits (ASCII mode) or 8 bits (RTU mode / ASCII mode)
* Parity bit : Even or Odd or None
* Stop bit : 1 bit or 2 bit
* Character code : ASCII (ASCII mode) or Binary (RTU mode)
* Error check : CRC-16 or LRC (depending on transmission mode)
* Signals in use : Sending and receiving data only (no control signal in use)

7.3 Programming Communications Transmission

Following parameters should be programmed for the LT8 controller programmed to the Communication Transmission.

1) Program the protocol “PtCL”. (Refer to Section 7.5.1)
2) Program the communications function “FunC”. (Refer to Section 7.5.2)
3) Program the transmission speed “rAtE”. (Refer to Section 7.5.3)
4) Program the character “CHAr”. (Refer to Section 7.5.4)
5) Program the remote / local type “rM.LC”.(Refer to Section 7.5.5)
6) Program the digital transmission type “d.trS” (Refer to Section 7.5.6)

7.4 Programming Communications Remote

When Communications Remote is programmed, SV data is received with digital communications instead of analog communications. In this case, SV data receiving is only executed digitally. Functions are the same as for Remote/Local (optional) functions. However, remote scale function is not provided to the Communications Remote function. By combining a master LT8 controller (Communications Transmission) and slave LT8 controllers (Communications Remote), digital remote control and zone control will be available.

(1) Following parameters should be programmed for slave LT8 controllers (Communications Remote).
   1) Program the protocol “PtCL”. (Refer to section 7.5.1)
   2) Program the communications function “FunC”. (Refer to section 7.5.2)
   3) Program the transmission speed “rAtE”. (Refer to section 7.5.3)
   4) Program the character “CHAr”. (Refer to Section 7.5.4)
   5) Program the remote shift “r.biA” if necessary. (Refer to section 7.5.7)
(2) When the communications function of the Mode 6 is programmed to “CoM (in case a master LT8 controller is programmed to “trS.2”)”, LT8 controller is programmed to Communications Remote (slave).
(3) Refer to section 7.5 for the programming of communications parameters.
(4) Switching between Remote and Local can be done from the Local/Remote of the Mode 6.
(5) Control is executed by SV on local condition until the first remote SV data is received since remote condition has been ready. The same function is executed when the power is turned on.

7.5 Programming Communications Transmission/Remote Parameters

Parameters related to Communications Transmission/Remote are provided to the Mode 6. Program the “Communications Function” , “Transmission Speed” , “Character” , “Digital Transmission Type” and “Remote Shift” by following the flowchart on page 3.
“Remote Shift” is only programmed for the Communications Remote.

7.5.1 Programming Protocol (PtCL)

(1) Press [↵] , then you will see [PtCL] .
(2) Choose the protocol by pressing [△] and [▽] , then press [↵] to register.

rtU: MODBUS RTU
ASC: MODBUS ASCII

*If a protocol is changed, communication function and character will become an initial value.

7.5.2 Programming Communications Functions (FunC)

(1) Press [↵] , then you will see [FUnC] .
(2) Choose the “trs.2” by pressing [△] and [▽] , then press [↵] to register.

CoM: Programmed for communications with a personal computer (Default CoM)
trS.2: Programmed for communication transmission with MODBUS (on a master unit)

7.5.3 Programming Transmission Speed (rAtE)

LT8 controllers to be used as a master unit (Communications Transmission) and slave units (Communications Remote) have to be programmed at same transmission speed.

(1) Press [↵] , then you will see [rAtE] .
(2) Choose the transmission speed by pressing [△] and [▽], then press [↵] to register.

Transmission speed: 9600, 19200 (19.2k) bps

7.5.4 Programming of Character (CHAr)

(1) Press [↵] , then you will see [CHAr] .

(2) Choose the character by pressing [△] and [▽] , then press [↵] to register.

Kind	Bit length	Parity	Stop bit	Default value
1	7bit	Even	1	5
2			2	
3		Odd	1	
4			2	
5	8bit	None	1	
6			2	
7		Even	1	
8			2	
9		Odd	1	
10			2	

* If the protocol is MODBUS RTU, 8 bit length is the only choice.

7.5.5 Programming Remote / local Type (rM.LC)

(1) Press [↵] , then you will see [rM.LC] .

(2) Choose the signal by pressing [△] and [▽] , then press [↵] to register.

7.5.6 Programming Digital Transmission Type (d. trS)

Transmission Type Programming	Meanings
PV	Transmits a measured value. (Default)
SV	Transmits a setpoint value.

(1) Press [↵] , then you will see [d.trS] .

(2) Choose the “Transmission Type” by pressing [△] and [▽] , then press [↵] to register.

7.5.7 Programming Remote Shift (r. biA)

The shift width of the SV data received by Communications Remote can be adjusted.

Arbitrary shift width can be programmed for each slave unit programmed to the Communications Remote. Program it as necessary.

(1) Press [↵] , then you will see [r.biA]

(2) To program the desired shift width with [△] and [▽], then press [↵] to register.

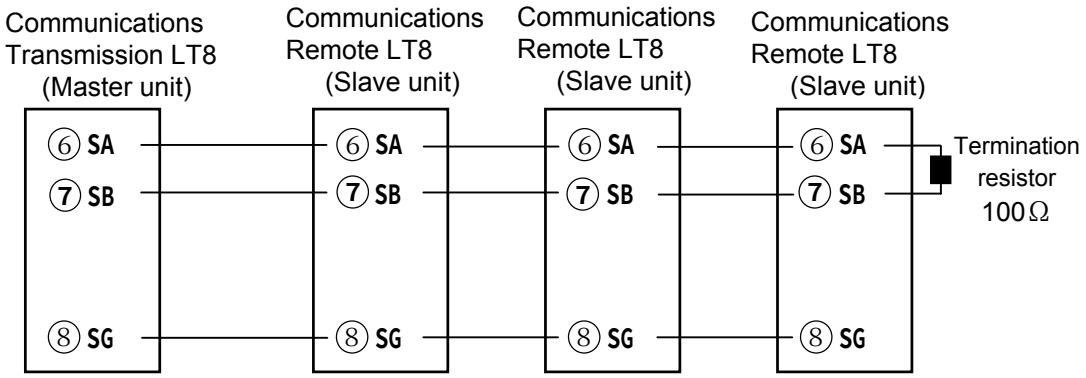
Shift variable width: –199.9 to 999.9 (Default: 0.0)

Caution

LT8 controllers to be used as a master unit (Communications Transmission) and slave units (Communications Remote) have to be programmed at the same transmission speed.

7.6 Communication transmission , Connections of Communications Remote

[RS-485]



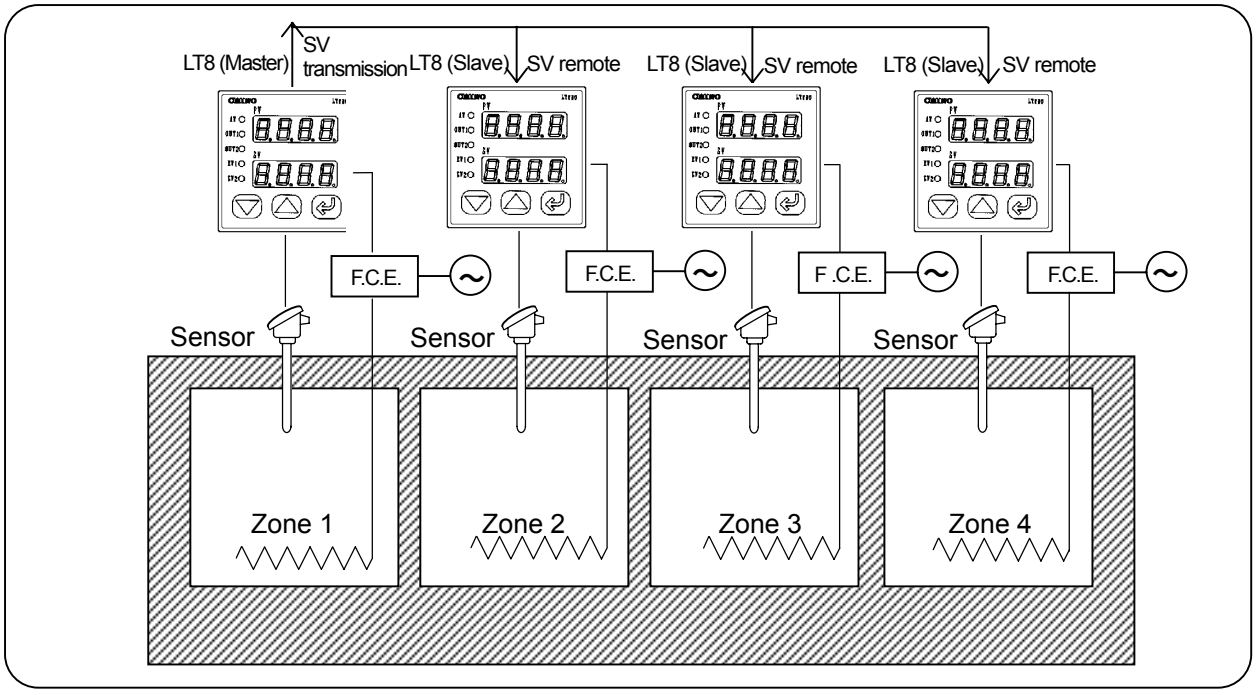
*Refer to Section 7.5.5 for remote change to slave unit.

7. 7 The example of temperature control adapting Communication transmission and Remote

7.7.1 Temperature Control for Multi Zone

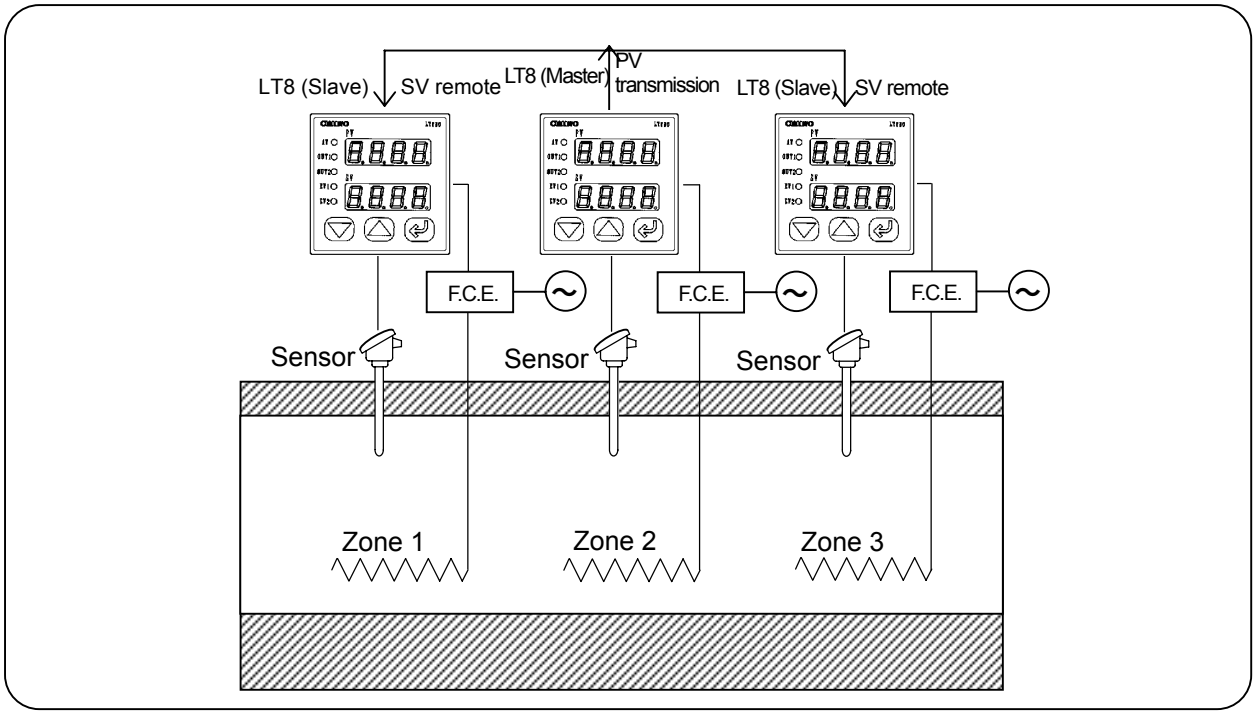
A master LT8 controller sends SV with the communications transmission and slave LT8 controllers receive it with the communications remote.

Since there is no error of an analog part, accurate remote control can be performed. With the remote shift function, temperature slope can be programmed at multiple zones.



7.7.2 Zone Control In A Soaking Pit

A master LT8 controller located in the middle sends PV by the communications transmission, and slave LT8 controllers located at both sides receive the PV as SV by the communications remote. This results in a control with superior characteristics in soaking.



CHINO

CHINO CORPORATION

32-8, KUMANO-CHO, ITABASHI-KU, TOKYO 173-8632

Telephone: 81-3-3956-2171
Facsimile: 81-3-3956-0915

Printed in Japan ( )