Technical Information

YS-net
Personal Computer Communication Functions

YS100 SERIES

Contents

YOKOGAWA ◆
Yokogawa Electric Corporation

2-9-32, Nakacho, Musashino-shi, Tokyo, 180 Japan
Tel.: 81-422-52-5622 Telex: 02822-327 YEW MTJ Fax.: 81-422-52-9802

Nov. 1995 (YG)

◆ PREFACE

This manual describes the methods and procedures when using YS-net personal computer (PC) communication functions.

The set-up and installation procedure of hardware and software is described in the manual YSS50 YS-net Parameter Definition File, IM 1B7C8-04E.

● Configuration of Document

This technical information document is composed of two chapters and two appendixes.

Chapter 1 introduces the overview and functions of YS-net PC communications, required equipment, and model and suffix codes.

Chapter 2 describes the communication statements in YS-net PC communication functions.

Appendix 1 lists the YS100 instrument data that can be accessed via communication.

Appendix 2 lists the error codes that may appear during the communication.

● Intended Readers

This document is addressed to engineers who will configure the communication system between YS100 instruments and a PC. Readers should be well aware of the fundamentals of YS100 instruments, the fundamentals and operations of an IBM PC/AT PC, the fundamentals and operations of Microsoft Windows 3.1 (the operating system in a PC), and the application software to be used.

● Trademarks and Licensed Software

- **Microsoft** and **MS-DOS** are registered trademarks of Microsoft Corporation, U. S. A.
- **Windows** is a trademark of Microsoft Corporation, U. S. A.
- **Echelon**, **LonTalk**; and **LonManager** are registered trademarks of Echelon Corporation, U. S. A.
- Other company and product names used throughout this document are trademarks or registered trademarks.

● Revision Record

November 1995 First Edition

Table 0.1 YS100 Series Document Map

Document Class	Document No.	Title	Usage (◎: Essential, ○: For Reference)				
			Programming for YS170	Engineering for function selections and parameter settings	Tuning	Normal Operation	Installation and Maintenance
Technical Information	TI 1B7A1-01E	YS100 SERIES Information	○	○		○	
	TI 1B7C0-01E Note 2	YS100 SERIES Intelligent Self-tuning Controllers			◎		◎
	TI 1B7C1-01E	YS150, YS170 Single-loop Controller Control Functions	◎	◎	◎	◎	
	TI 1B7C2-03E Note 3	YS170 Programmable Functions	◎		○		
	TI 1B7C8-03E Note 1	YS100 SERIES Communication Functions		◎		◎	
	TI 1B7C8-04E Note 5	YS-net Peer-to-peer Communication Functions		◎			
	TI 1B7C8-05E Note 5	YS-net Personal Computer Communication Functions		◎		◎	
Instruction Manual	IM 1B7C1-01E	YS150 Single-loop Multi-function Controller YS170 Single-loop Programmable Controller	○	◎	◎	◎	◎
	IM 1B7C8-01E	YSS10 YS100 SERIES Programming Package	◎				
	IM 1B7C8-03E Note 1	YS100 SERIES RS-485 Communication Functions (/A31) DCS-LCS Communication Functions (/A32)		◎		◎	◎
	IM 1B7C8-04E Note 5	YSS50 YS-net Parameter Definition File		◎			◎
	IM 1B7D2-01E	YS131 Indicator with Alarm		◎	◎	◎	◎
	IM 1B7D3-01E	YS135 Auto/Manual Station for SV Setting		◎	◎	◎	◎
	IM 1B7D4-01E	YS136 Auto/Manual Station for MV Setting		◎	◎	◎	◎
	IM 1B7D5-01E Note 4	YS110 Standby Manual Station				◎	

Note 1: Only when used with supervisory communication functions

Note 2: Only when using self-tuning functions

Note 3: Only for YS170 programmable controllers

Note 4: The YS110 can be a standby station only for the YS150, YS170, or YS136

Note 5: Only when using YS-net communication functions

1. OVERVIEW

This chapter describes the following:

- Overview of YS-net
- Functions and specifications of PC communication functions
- Configuration of YS-net
- Model and suffix codes

1.1 Overview of YS-net

YS-net is not a general-purpose communications network but, rather, is a network used exclusively for YS series instruments. In comparison with the RS-485 communication functions which have been offered, YS-net thus requires less engineering work and works at a faster speed. It allows communications using the DDE function in a PC. And communications with general Supervisory Control And Data Acquisition (SCADA) software through the DDE is easy. There are two functions in YS-net.

- Peer-to-peer communication
- PC communication

These functions are available with the following YS100 instruments.

- YS170 single-loop programmable controller (simply referred to as the YS170 controller)
- YS150 single-loop multi-function controller (simply referred to as the YS150 controller)

The YS-net communication option card (Option Code /A33) must be installed in each controller, however. The peer-to-peer communication, which is available only for YS170 controllers, is the function that allows data for computation and control to be exchanged between YS170 controllers via YS-net. The PC connection function allows a PC to access the YS instruments via YS-net. Monitoring, operation, and settings like PID tuning can be done at a PC with this function.

There are three types of YS-net operations.

(1) Peer-to-peer communication only
(2) PC communication only
(3) Peer-to-peer communication and PC communication together at the same time

The table below shows the engineering tasks required to operate YS-net depending on the operation type and mode you choose.

	Engineering
(1) Peer-to-peer communication only	Programming of user program for YS170
(2) PC communication only	Preparation of application software on the PC
(3) Peer-to-peer communication and PC communication	Both tasks above

This document covers the functions, specifications, and installation procedures for operation (2), "PC communication only". For operation (1), refer to *YS-net Peer-to-peer Communication Functions*, TI 1B7C8-04E. Operation (3) is the simultaneous use of operations (1) and (2). There are no restrictions when running both, so refer to both technical information documents.

1.2 Personal Computer Communication Functions

The PC communication function allows up to sixteen YS170 or YS150 controllers and one PC to be connected to YS-net. Specifying the call name corresponding to the communication address set in each controller, a Microsoft Windows-based application program on a PC can retrieve controller data such as the PV, SV, or MV.

Specifications of PC Communication

Communication speed:	78.125 kbps
Communication medium:	Twisted-pair cable (AWG20 or AWG22 cable)
Connection:	Daisy-chain connection
Distance:	Up to 1000 m

Number of YS instruments that can be connected: up to 16 on the following models

- YS170 controller
- YS150 controller

(Either model must be equipped with the communication card referred to as option /A33.)

Data update period on the PC: 16 data items (see below) from 16 controllers per second

The data items discussed here are as follows: Measured value, setpoint value, manipulated output value, high-limit alarm setpoint, low-limit alarm setpoint, deviation alarm setpoint, velocity alarm setpoint, output high-limit setpoint, output low-limit setpoint, proportional band, integral time, derivative time, operation mode, lamp status, system alarm status, process alarm status
(Note: Data items other than these can be communicated.)

1.3 Configuration of YS-net

Figure 1.1 shows the system configuration when YS-net is used for PC communication. Figure 1.2 shows the software configuration for a PC. Among these components, Yokogawa provides the following:

- YS100 series instruments, YS150 and YS170
- YSS50 YS-net parameter definition file

The components supplied by others are as follows:

- IBM PC/AT-compatible personal computer (one)
- YS-net communication board (from Echelon)
- YS-net DDE server (from Echelon)
- Microsoft Windows-based application software which has the DDE function

In respect to the hardware, the system is composed of up to sixteen YS150 or YS170 controllers and an IBM PC/AT-compatible PC. Install the YS-net communication board in the I/O slot of the PC.

With respect to the software, use MS-DOS and Microsoft Windows for its OS. The application on Microsoft Windows can read and set YS instrument data via the DDE server without a specific program. The DDE server refers to the YSS50 parameter definition file to communicate with a YS150 or YS170 controller via the YS-net communication board.

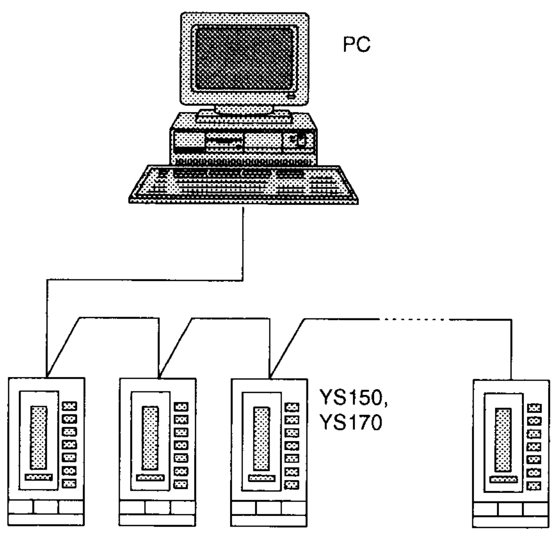


Figure 1.1 Hardware System Configuration of YS-net for PC Communication

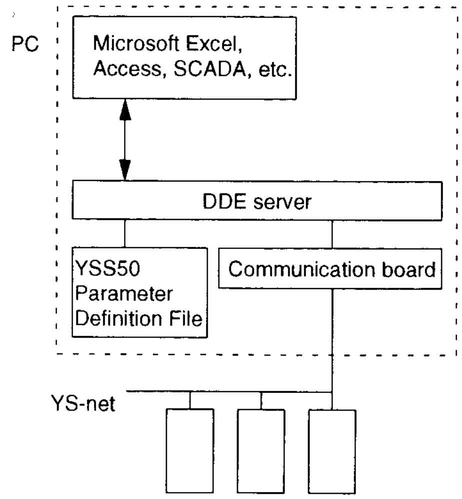


Figure 1.2 Software System Configuration of YS-net for PC Communication

Table 1.1 YS-net Components for PC Communication

PC Note 1	An IBM PC/AT-compatible PC An ISA slot that can be used for a full-size extension board Floppy drive for 3.5-inch, 1.44-MB floppy disks CPU: 486DX2, 66 MHz or greater Memory: 8 MB or more HD: 6 MB or more of free space DOS Version 5.0 or later, Windows 3.1 or later
YS-net communication board Note 1	Supplied by Echelon (1)73100-01 PCLTA PC LonTalk Adapter with communication driver software and instruction manual (2)77010 TPM/XF-78 Modular Transceiver To be installed at (1).
YS-net DDE server	Supplied by Echelon 33000 LonManager DDE Server with instruction manual
Application software	Commercially available Microsoft Windows compatible software which has the DDE function (for instsnce, Microsoft Excel and various SCADA software)

Note 1: An IBM PC/AT-compatible notebook can be used, too. In this case, use the 73000 SLTA/2 Serial LonTalk Adapter supplied by Echelon instead of the YS-net communication board. This is an external adapter and is attached to the RS-232C port of the notebook PC. In this document there is no further description of how to use this adapter.

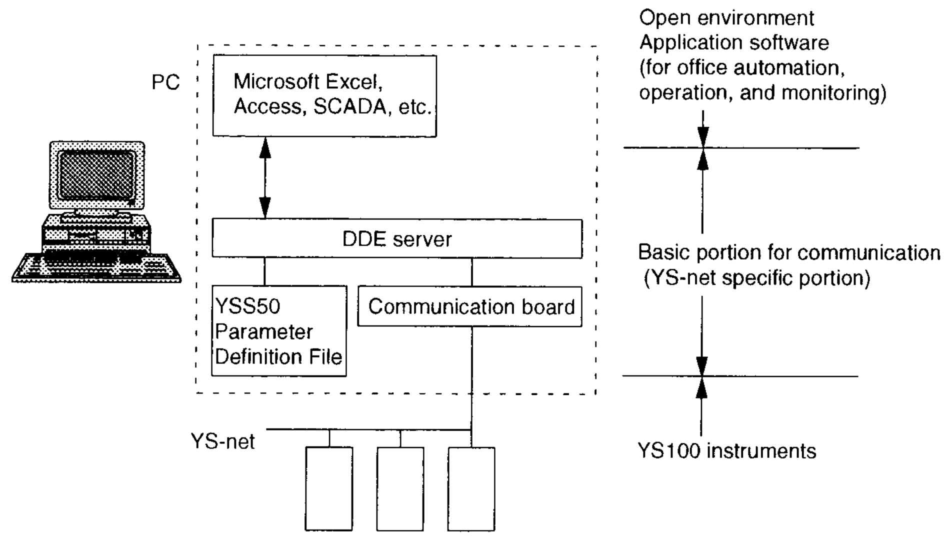
1.4 Model and Suffix Codes

Model	Suffix Code			Option Code	Description
YS170 YS150					Single-loop programmable controller Single-loop multi-function controller
	-0				General purpose
		1			Always 1
			1		Power supply: 100 V
			2		Power supply: 220 V
				/A33	With YS-net communication

Model	Suffix Code		Description
YSS50			YS-net parameter definition file
	-000		Always 000

2. DDE COMMUNICATION WITH PC

This chapter describes the data communication method between YS100 series controllers and a PC using Dynamic Data Exchange. To describe the communication texts, we use Microsoft Excel 5.0 in this document.



2.1 DDE Interface Function

The YS-net PC communication functions use DDE communication software (called the DDE server). If the application software has the DDE client function (or just the DDE function), simply requesting the DDE server to get and send YS100 instrument data executes communications with YS100 instruments. This section describes how to access the DDE server from application software.

2.1.1 Communication Type

There are two types of communications with YS100 instruments via a DDE server. Use either one depending on your need.

Periodic communication: Exclusively used for data acquisition and automatically collects frequently required data on a periodic basis.

Message communication: Used to collect and set any of the available data from the YS100 instruments and is further divided into simplified and standard communications.

It is generally useful to employ periodic communication for data acquisition and the simplified message communication for data setting. To learn how to write communication text, read the corresponding sections that follow.

Communication Type		Data Acquisition	Data Setting	Section
Periodic communication		Periodic data acquisition Simple communication procedure Restricted to the major data required in normal operation	Unavailable (Use message communication.)	2.2
Message communication		Can handle all communication-accessible data.		2.3
	Simplified	Gets data that cannot be handled by periodic communication	Simple communication text No confirmation message at data setting	2.3.2 2.3.3
	Standard	Not available (Use simplified communication.)	Has a confirmation message at data setting. Multiple data items can be set at one time.	2.3.4

2.1.2 DDE Access Code

When getting and sending DDE server data from the application software, specify the location and name of the data by parameters in the DDE communication command: **Application**, **Topic**, and **Item**.

- **Application**

This is the name of the application in the DDE server and is always **LMSRVR1** for YS-net.

- **Topic**

A DDE server always has at least one Topic. For YS-net, the Topics are as follows.

For periodic communication: **NETVAR**

For message communication: **MSGTAG**

- **Item**

Each Topic in a DDE server always has at least one Item. The Item depends on the communication type and is referred to in the corresponding descriptions that follow.

NOTE

When describing DDE access code (Application, Topic and Item), you can use both upper case and lower case characters.
There is no difference between using upper case and lower case characters.

2.2 Periodic Communication

In periodic communication, the application software specifies Application-Topic-Item to collect the data required. In this communication, accessible data are restricted to frequently required data items. To collect other data items, use message communication. Since this communication runs periodically, this is not suitable to data setting. Use message communication for data setting. If a communication error occurs during periodic communication, the data values at that runtime cannot be gotten.

2.2.1 DDE Access Code

The configuration of the DDE access code in periodic communication is shown in the table below. The Application and Topic are fixed. The Item is determined by the data item to be gotten.

	DDE Access Code
Application	LMSRVR1
Topic	NETVAR
Item	Instrument name.parameter name.field name

The description for the Item is described below.

Instrument name: Fixed to ys1 to ys16 according to the communication address 1 to 16 assigned to each YS100 instrument.

Parameter name: Generic title of the set of YS100 instrument data

Examples:

lpo[0]: denotes the data items belonging to loop 1 in the YS100 instrument.

lpo[1]: denotes the data items belonging to loop 2 in the YS100 instrument.

There are some other parameter names. See Section 2.2.4.

Field name: Name of YS100 instrument data. See Section 2.2.4.

For instance, to specify the PV of loop 1 for the instrument at YS-net address 1, write "ys1.lpo[0].pv".

NOTE

The subscript of the parameter originates with zero. Thus, Loop 1 is described as **lpo[0]**, and loop 2, **lpo[1]**.

2.2.2 Data Acquisition Method

There are two ways to acquire data described as below.

(1) Data Acquisition Equation

In Microsoft Excel, simply describing the equation as follows in the cell where you want to view the data value executes data acquisition. The communication command is composed of the Application, Topic, Item, and suffix codes.

Examples of data acquisition equations:

=LMSRVR1|NETVAR ! 'ys1.lpo[0].pv -r0' /100	Gets the measured value (PV) of loop 1 of the instrument whose instrument name is YS1
=LMSRVR1|NETVAR ! 'ys8.cx[2].a[3] -r0' /40.96	Gets the value of the CX2 peer-to-peer communication resister of the instrument whose instrument name is YS8
=LMSRVR1|NETVAR ! 'ys3.ys_status.load -r0'	Gets the value of the loading ratio of the instrument whose instrument name is YS3

=: Denotes, in Microsoft Excel, that the strings that follow are an equation.

-r0: Number of times to be retried upon a communication error

/100: Microsoft Excel standard equation. Denotes division by 100. The measured value pv is in the form of 0-10000 for the value of 0-100% (See Table 2.1). Division by 100 is thus used to give the form of 0-100%.

/40.96: Microsoft Excel standard equation. Denotes division by 40.96. The peer-to-peer communication data cx[2]. a[3] is in the form of 0-4096 for the value of 0-100% (See Table 2.2). Division by 100 is thus used to give the form of 0-100%.

A |, ! (exclamation point), and ' (single quotation mark) are delimiters.

If the communication attempt succeeds, the corresponding data values in the YS100 instrument appears in the cell where you described the equation.

For the equation to get PV shown above, if the value is 54.3%, the cell appears as 54.3.

(2) Data Acquisition Macro

The following describe the method to perform data acquisition using the macro function of Microsoft Excel. The meanings of the parameters are the same as the aforementioned examples. In this example, the SV and PV are viewed in cells (3,C) and (4,C) on worksheet 1, respectively. For each command in the Macro (e.g., DDEInitiate and DDERequest), see Section 2.3.

```
Sub YSNET1()          Example of a periodic communication macro
YsnetChannel = DDEInitiate("lmsrvr1", "netvar")
Worksheets("sheet1").Cells(3, 3).Value = DDERequest(YsnetChannel, "ys1.lpo[0].SV -r0")
Worksheets("sheet1").Cells(4, 3).Value = DDERequest(YsnetChannel, "ys1.lpo[0].PV -r0")
DDETerminate (YsnetChannel)
End Sub
```

Where **Sub YSNET1**: **Sub** denotes a macro and **YSNET1** is an arbitrary macro name.

YsnetChannel: A parameter expressed by an arbitrary character string which identifies the YS-net channel

DDEInitiate, **DDERequest**, **DDETerminate**: DDE messages

2.2.3 Communication Performance

Communication performance for periodic communication depends on the number of data to acquire and the performance of the PC. The relationship between the data-acquisition period and number of communication data items are discussed here on the assumption of the PC environment as shown. Communication performance mainly depends on the speed of the YS-net communication line; it depends much less on CPU performance of the PC. [Note] The number of YS100 instruments on the YS-net does not affect the communication performance so much, either.

If you set too short a scanning period ("Poll Interval" in the DDE server) for the number of data to acquire, data acquisition takes priority and the DDE server automatically expands its scanning period. This action prevents data from being missed even when a very short scanning period is set.

Guideline on Number of Data Acquired and Scanning Period

1 to 16 parameter names:	1 second
17 to 32 parameter names:	2 seconds
33 to 48 parameter names:	3 seconds

Increase by one second for every 16 parameter names increased.
Note that "parameter names" here correspond to what you see in the table 2.1, 2.2 and 2.3.

Environment When Monitoring Communication Performances

CPU = 486DX2/66 MHz; Main memory = 16 MB
The DDE server and Microsoft Excel is the only applications running on the PC and data acquisition is attempted using Microsoft Excel.

TIP

Relationship Between Communication Performance and Performance of CPU in PC

As mentioned above, CPU performance does not affect communication performance. However, if the application software you use is 'heavy,' a rather slow-performing CPU may suspend YS-net communication in order to run the application itself. Therefore, it is recommended that you use a CPU with performance as high as possible among the ones recommended by the application software.

2.2.4 Communication-Accessible Data

The table below lists the data items accessible by periodic communication. These are restricted to the *loop parameters*, which are required during normal operation, such as the measured value, setpoint, manipulated output value, and PID parameters.

Table 2.1 List of Loop Parameters

Parameter Name	Field Name	Description	Data Format
lpo[0]	**no**	Communication counter of loop 1	An integer. 0 to 255. Increased by 1 every 200 mS when normal. (Can be used to check the fail of the instrument.)
	ls	Operation mode of loop 1	1 = MAN; 2 = AUT; 3 = CAS; 5 = DDC; 6 = SPC; 9 = BUM; 10 = BUA
	lamp	Lamp status of loop 1	The statuses of lamps are represented by 0/1 statuses of 8 bits integer where 1 = on and 0 = off. Bit 7 (most significant bit) for **ALM** lamp; bit 2 for **C** lamp; bit 1 for **A** lamp; bit 0 for **M** lamp; other bits are not used.
	sys_alm	System alarm on loop 1	The alarm statuses are represented by 0/1 statuses of 8 bits integer where 1 = alarm and 0 = normal. Bit 7 (MSB) for input signal over-range; bit 6 for current-output open; bit 3 for computation over-range; bit 2 for "RAM contents volatilized"; other bits are not used.
	prc_alm	Process alarm on loop 1	The alarm statuses are represented by 0/1 statuses of 8 bits integer where 1 = alarm and 0 = normal. Bit 7 (MSB) for high-limit alarm; bit 6 for low-limit alarm; bit 5 for deviation alarm; bit 4 for velocity alarm; other bits are not used.
	pv	Measured value of loop 1	100% is displayed as 10000.
	sv	Setpoint value of loop 1	Same as above.
	mv	Manipulated output value of loop 1 Note 1	Same as above.
	ph	High-limit alarm setpoint of loop 1	Same as above.
	pl	Low-limit alarm setpoint of loop 1	Same as above.
	dl	Deviation alarm setpoint of loop 1	Same as above.
	vl	Velocity alarm setpoint of loop 1	Same as above.
	mh	Output-high-limit alarm setpoint of loop 1	Same as above.
	ml	Output-low-limit alarm setpoint of loop 1	Same as above.
	pb	Proportional band of loop 1	2.0-999.9% is displayed as 20-9999.
	ti	Integral time of loop 1	1-9999 s is displayed as 1-9999.
	td	Derivative time of loop 1	0-9999 s is displayed as 0-9999.
lop[1]	**no**	Communication counter of loop 2	Same as the **no** for loop 1.
	ls	Operation mode of loop 2	Same as the **ls** for loop 1.
	lamp	Lamp status of loop 2	Same as the **lamp** for loop 1.
	sys_alm	System alarm on loop 2	Same as the **sys_alm** for loop 1.
	prc_alm	Process alarm on loop 2	Same as the **prc_alm** for loop 1.
	pv	Measured value of loop 2	100% is displayed as 10000.
	sv	Setpoint value of loop 2	Same as above.
	mv	Manipulated output value of loop 2 Note 1	Same as above.
	ph	High-limit alarm setpoint of loop 2	Same as above.
	pl	Low-limit alarm setpoint of loop 2	Same as above.
	dl	Deviation alarm setpoint of loop 2	Same as above.
	vl	Velocity alarm setpoint of loop 2	Same as above.
	mh	Output-high-limit alarm setpoint of loop 2	Same as above.
	ml	Output-low-limit alarm setpoint of loop 2	Same as above.
	pb	Proportional band of loop 2	2.0-999.9% is displayed as 20-9999.
	ti	Integral time of loop 2	1-9999 s is displayed as 1-9999.
	td	Derivative time of loop 2	0-9999 s is displayed as 0-9999.

Note 1: The values of **lpo[0].mv** and **lpo[1].mv** are as below.

When BSC (single loop control function) for YS170 or Single Loop Mode for YS150;
lpo[0].mv = MV1
lpo[1].mv = MV2 (only for YS170)

When CSC (cascade control function) for YS170 or Cascade Mode for YS150;
lpo[0].mv = Internal MV output of loop1 (This value is to be set to SV2, the setpoint variable of loop2.)
lpo[1].mv = MV1, the final manipulated variable of the cascade

When SSC (selector control function) for YS170 or Selector Mode for YS150;
lpo[0].mv = **lpo[1].mv** = MV1, the final manipulated variable of the selector

The table below lists the data items accessible also by peer-to-peer communication. Not only the data in send-and-receive YS170 controllers (addresses 1 to 4), but also the data in receive-only YS170's (addresses 5 to 16) can be accessed.

Table 2.2 Parameters for peer-to-peer Communication

Parameter Name	Field	Description (Register Name in YS170)	Data Format
cy	a[0]	CY01 (peer-to-peer communication output 1)	Register data in YS170. 0-100% is displayed as an integer of 0-4096.
	a[1]	CY02 (peer-to-peer communication output 2)	
	a[2]	CY03 (peer-to-peer communication output 3)	
	a[3]	CY04 (peer-to-peer communication output 4)	
cx[0]	a[0]	CX01 (peer-to-peer communication input 1 from YS-net address 1 instrument)	Same as above.
	a[1]	CX02 (peer-to-peer communication input 2 from YS-net address 1 instrument)	
	a[2]	CX03 (peer-to-peer communication input 3 from YS-net address 1 instrument)	
	a[3]	CX04 (peer-to-peer communication input 4 from YS-net address 1 instrument)	
cx[1]	a[0]	CX05 (peer-to-peer communication input 1 from YS-net address 2 instrument)	Same as above.
	a[1]	CX06 (peer-to-peer communication input 2 from YS-net address 2 instrument)	
	a[2]	CX07 (peer-to-peer communication input 3 from YS-net address 2 instrument)	
	a[3]	CX08 (peer-to-peer communication input 4 from YS-net address 2 instrument)	
cx[2]	a[0]	CX09 (peer-to-peer communication input 1 from YS-net address 3 instrument)	Same as above.
	a[1]	CX10 (peer-to-peer communication input 2 from YS-net address 3 instrument)	
	a[2]	CX11 (peer-to-peer communication input 3 from YS-net address 3 instrument)	
	a[3]	CX12 (peer-to-peer communication input 4 from YS-net address 3 instrument)	
cx[3]	a[0]	CX13 (peer-to-peer communication input 1 from YS-net address 4 instrument)	Same as above.
	a[1]	CX14 (peer-to-peer communication input 2 from YS-net address 4 instrument)	
	a[2]	CX15 (peer-to-peer communication input 3 from YS-net address 4 instrument)	
	a[3]	CX16 (peer-to-peer communication input 4 from YS-net address 4 instrument)	

The table below lists the status data which represent the operating status and self-diagnosis data.

Table 2.3 Status Data

Parameter name	Field name	Data Format
ys_status	**adrs**	YS-net address for the instrument. 1 to 16.
	ctl	0 = stop; 1 = run; 2 = test run
	load	Loading ratio (0 to 100%)
	fil_status	0 = EEPROM* normally written; -128 = EEPROM while being written; 1 to 15 = EEPROM write abnormal
	eeprom_status	0 = normal; other than zero = abnormal

* EEPROM is an erasable memory chip that can retain the memory contents even during a power failure.

2.3 Message Communication

With message communication, a PC can get and send all accessible data (approximately 400 kinds of data items) from and to YS100 instruments. When you use message communication, communication is performed once when the communication program is activated. You can choose standard or simplified message communication.

Simplified: Omits a communication confirmation message when setting data. The communication text is simple.

Standard: A communication confirmation message intervenes when setting data. Used when the success of communication should be confirmed.

For the DDE access code for message communication, see the following descriptions for each type of communication.

2.3.1 DDE Communication Command

The following shows an example of the communication commands in Microsoft Excel (the communication command is called "method" in Microsoft Excel) when the application software use message communication to access the YS-net DDE server. For details on the commands, refer to the Microsoft Excel manual.

The following five DDE commands are used. To exchange data, fill in the proper information — Application, Topic, Item — in the DDE communication command descriptions.

DDEInitiate	Start the DDE
DDERequest	Get a character string from another application
DDEPoke	Deliver a character string to another application
DDEExecute	Execute a command in another application (not used in YS-net communications.)
DDETerminate	Terminate the DDE

● **DDEInitiate** - Start the DDE

Connects another application that has DDE server function and opens a DDE channel.

Format: **channelNumber=DDEInitiate(application,topicname)**

Parameters:
- **channelNumber**: Specifies the variable in which the DDE channel connected by **DDEInitiate** is to be stored
- **application**: Specifies the **Application** to be connected. Fixed at **LMSRVR1** for YS-net.
- **topicname**: Specifies the **Topic** in the application. Fixed at **MSGTAG** for YS-net message communication.

● **DDERequest** - Get a character string from another application

Gets a character string or numeric from the application that was connected by **DDEInitiate**.

Format: **variable=DDERequest(channelNumber,item)**

Parameters:
- **variable**: The variable to which a retrieved value or string is to be stored
- **channelNumber**: Enter the channel number specified when invoking **DDEInitiate**.
- **item**: Specifies the **Item**.

● **DDEPoke** - Deliver a character string to another application

Delivers a character string or numeric to an application that was connected by **DDEInitiate**.

Format: **DDEPoke(channelNumber,item,text)**

Parameters: **channelNumber**: Enter the channel number specified when invoking **DDEInitiate**.

item: Specifies the **Item**.

text: Specifies the character string to be delivered. Or specifies the cell address where the character string is.

● **DDETerminate** - Terminate the DDE

Terminates the DDE.

Format: **DDETerminate(channelNumber)**

Parameters: **channelNumber**: Enter the channel number specified when invoking **DDEInitiate**.

2.3.2 Setting Data by Simplified Message Communication

The following describe the procedure to send a data value using simplified message communication.

(1) DDE Access Code

The configuration of the DDE access code in message communication is shown in the table below. The Application and Topic are fixed. In Item, specify the instrument name to which the data are to be sent. The **-r2** appended to the Item description denotes the number of times to be retried if communication fails.

	DDE Access Code
Application	LMSRVR1
Topic	MSGTAG
Item	Instrument name.msg_out.2 -t ys_string -r2

(2) Example of Macro Program

To send a data value to a YS100 instrument with Microsoft Excel, generate a macro program for communication and run it. In the program, specify Application and Topic on the second line, and Item and the cell address (Cells(1,1) = cell (1,A) in this example) where the data setting message is stored, on the third line. In this example, the data setting message stored in cell (1,A) is to be delivered to the YS100 instrument. The data setting message must thus be written in cell (1,A) prior to running this macro.

```
Sub YSNET2()
YsnetChannel = DDEInitiate("lmsrvr1", "msgtag")
DDEPoke YsnetChannel, "ys1.msg_out.2 -t ys_string -r 2", Cells(1,1)
DDETerminate (YsnetChannel)
End Sub
```

Where **Sub YSNET2**: **Sub** denotes a macro and **YSNET2** is an arbitrary macro name.

YsnetChannel: A parameter expressed by an arbitrary character string which identifies the YS-net channel

DDEInitiate, **DDEPoke**, **DDETerminate**: DDE messages

Underscores (_), quotation marks (""), spaces, and periods (.) in the macro description must be written precisely.

(3) Message Format

The format of transmission message for data setting written in the specified cell (cell (1,A) in the example above) is as below.

Transmission message (PC →YS100) : **PARAM DATA**

For the actual **PARAM** and the corresponding **DATA**, see the appendix 1.
PARAM is the data type in the YS100 instrument.
DATA is the data value to be set. Enter the value or character string to be set.
Enter a space between the **PARAM** and **DATA**.

Example: To set the SV1 setpoint to 75.0%, enter the following in the cell.
SV1 75.0

2.3.3 Getting Data by Simplified Message Communication

The following describe the procedure to get a data value using simplified message communication. When getting a data value using simplified message communication, the application software first sends a data request message to the YS100 instrument. The instrument then returns the corresponding data value and the application software receives it by a data reception statement.

(1) DDE Access Code

The configuration of the DDE access code in message communication is shown in the table below. In Item, specify the instrument name from which the data are to be gotten. Note that the Item description differs between the message transmission and the message reception.

	DDE Access Code for Message Transmission	DDE Access Code for Message Reception
Application	LMSRVR1	LMSRVR1
Topic	MSGTAG	MSGTAG
Item	Instrument name.msg_out.1 -t ys_string -s UNACKD	Instrument name.msg_in.1 -t string

(2) Example of Macro Program

The following shows an example of macro programs to get data from a YS100 instrument to the Microsoft Excel. In the program, specify Application and Topic on the second line, Item and the cell address (Cells(1,1) = cell (1,A) in this example) where the data getting message must be written prior to running this macro, on the fourth line. Execute the data reception statement at least 600 ms after execution of the data request statement. The cell address to store the response is described on the tenth line as Cells(2,1) meaning cell (2,A). If a communication error occurs while getting data, there will be no response from the YS100 instrument. Design the application to retry communication in such a case. Set the interval until the next attempt to at least 600 ms.

```
Sub YSNET3()                          'Message data retrieval macro
YsnetChannel = DDEInitiate("lmsrvr1", "msgtag")
YSdata = DDERequest(YsnetChannel, "ys1.msg_in.1 -t string")
                                                    'Initialize the reception item.
DDEPoke YsnetChannel, "ys1.msg_out.1 -t ys_string -s UNACKD", Cells(1, 1)
                                                    'Request for data
      countT = 0                      'Set the waiting time for communication.
      Do While countT < 50            'This counter setpoint depends on the CPU.
      countT = countT + 1
      Loop
YSdata = DDERequest(YsnetChannel, "ys1.msg_in.1 -t string")
                                                    'Retrieves the response.
Worksheets("sheet1").Cells(2,1).Value = YSdata
DDETerminate (YsnetChannel)
End Sub
```

Where **Sub YSNET3**: **Sub** denotes a macro and **YSNET3** is an arbitrary macro name.
YsnetChannel: A parameter expressed by an arbitrary character string which identifies the YS-net channel
DDEInitiate, DDEPoke, DDERequest, DDETerminate: DDE messages

(3) Message Format

The formats of the transmission message and the reception message in the specified cells (cell (1,A) and cell (2.A) in the example above) are as below.

Transmission message (PC → YS100):	**PARAM**	
Reception message (YS100 → PC):	**PARAM**	**DATA**

For the actual **PARAM** and the corresponding **DATA**, see the appendix 1. **PARAM** is the data type in the YS100 instrument. The reception message is always composed of forty characters, in which the data values (**DATA**) are filled from the end.

Example: To get the PV1 measured value, enter that data type in the cell. After communication is established, the reception message is viewed in the other cell.

Transmission message:	**PV1**	
Reception message:	**PV1**	**54.3**

2.3.4 Standard Message Communication

The following describe the procedure to use standard message communication. First, send a communication message from the application software to the YS100 instrument. If the communication attempt succeeds, the instrument returns a response message. The application software reads this response message and performs the consequent processing. Like this, a sent message always corresponds to a response message; thus, standard message communication is useful when you want to confirm the data value after setting. Standard message communication can, of course, be used for getting a data value, too.

(1) DDE Access Code

The configuration of the DDE access code in message communication is shown in the table below. In Item, specify the instrument name. Note that the Item description differs between the message transmission and the message reception.

	DDE Access Code for Message Transmission	DDE Access Code for Message Reception
Application	LMSRVR1	LMSRVR1
Topic	MSGTAG	MSGTAG
Item	Instrument name.msg_out.3 -t ys_string -s UNACKD	Instrument name.msg_in.3 -t string

(2) Example of Macro Program

The following shows an example of a macro program for standard message communication. This macro can be used for both getting and sending data, depending on the transmission message. In the macro, specify the Application and Topic on the second line, and the cell address (Cells(1,1) = cell (1,A) in this example) where the transmission message (Item and data value to be set) is stored, on the fourth line. Invoke the statement to retrieve the response at least 600 ms after invocation of the statement for sending the message. The cell address to store the response is described on the tenth line as Cells(2,1) meaning cell (2,A). If a communication error occurs while getting data, there will be no response from the YS100 instrument. Design the application to retry communication in such a case. Set the interval until the next attempt to at least 600 ms.

```
Sub YSNET4()
YsnetChannel = DDEInitiate("lmsrvr1", "msgtag")
YSdata = DDERequest(YsnetChannel, "ys1.msg_in.3 -t string")
                                                  'Initialize reception item.
DDEPoke YsnetChannel, "ys1.msg_out.3 -t ys_string -s UNACKD", Cells(1, 1)
                                                  'Send message.
    countT = 0              'Set waiting time for communication.
    Do While countT < 50    'This counter setpoint depends on the CPU.
    countT = countT + 1
    Loop
YSdata = DDERequest(YsnetChannel, "ys1.msg_in.3 -t string")
                                                  'Retrieves the response.
Worksheets("sheet1").Cells(2,1).Value = YSdata
DDETerminate (YsnetChannel)
```

Where **Sub YSNET4**: **Sub** denotes a macro and **YSNET4** is an arbitrary macro name.

YsnetChannel: A parameter expressed by an arbitrary character string which identifies the YS-net channel

DDEInitiate, **DDEPoke**, **DDERequest**, **DDETerminate**: DDE messages

(3) Message Format

The formats of the messages written and received in the specified cells (cell (1,A) and cell (2,A) in the example above) are as below.

Transmission message (PC → YS100):	**DP nn mm PARAM1 DATA1 . . . . PARAMmm DATAmm**
Reception message (YS100 → PC):	**DP nn mm DATA1 . . . . DATAmm**

The maximum length of both transmission and reception message is thirty-one characters. **DP** is the command to set data to YS100 instrument. Always set **nn** to 00. **mm** is the number of communication data items. Always specify **mm** with two characters like 01 or 02. For the actual **PARAM1**, ... and the corresponding **DATA1**, ..., see the appendix 1. The configuration of the messages are the same as those in RS-485 communication for YS100 instrument. For details, see *YS100 SERIES RS-485 Communication Functions*, TI 1B7C8-03E.

Example: To change the operation mode of the address 1 instrument to AUTO and set the SV1 setpoint to 75.0%, enter and send the following message. After communication is established, the response as shown in the lower row of the following messages returns.

Transmission message:	**DP1 00 02 SV1 75.0 LS1 AUT**
Reception message:	**DP1 00 02 75.0 AUT**

Appendix 1. COMMUNICATION DATA LIST FOR MESSAGE COMMUNICATION

This appendix describes the types and ranges of data accessed by way of message communication. The lists below are the same as those for YS100 Series RS-485 communication functions.

NOTE

1. Data Type (**PARAM**) for message communication always must be written with upper case characters.
2. The data listed and described in this appendix are not supported by periodic communication. For periodic communication, see Section 2.2.

TableA1.1 Communication Data List (1)

Data Type (PARAM)	Name	Write-enabled	Unit	Range for Displaying (DATA)	Remarks	Mode	
						Multi-func.	Pro-gram
PV1, PV2	Process variable	—	%	-6.3 to 106.3		✓	✓
SV1, SV2	Setpoint	OK	%	-6.3 to 106.3	When operation mode is MAN, AUTO, SPC, DDC, BUM, or BUA mode, setting is enabled.	✓	✓
CSV1, CSV2	Cascade setpoint	—	%	-6.3 to 106.3		✓	✓
DV1, DV2	Deviation	—	%	-6.3 to 106.3		✓	✓
MV1, MV2	Manipulated variable	OK	%	-6.3 to 106.3	When operation mode is MAN, DDC, or BUM mode, setting is enabled. MV2 is valid only for independent 2-loop control. In cascade or selector mode, MV1 is the same as MV2.	✓	✓
PRCA	Process alarm	—	—	$X_1X_2X_3X_4X_5X_6X_7X_8$ Xn = 0 (Normal) Xn = 1 (Fail)	X_1 = Hgh limit alarm 1 X_2 = Low limit alarm 1 X_3 = Deviation alarm 1 X_4 = Velocity alarm 1 X_5 = High limit alarn 2 X_6 = Low limit alarm 2 X_7 = Deviation alarm 2 X_8 = Velocity alarm 2 Example: When high limit alarm 1 occurs. 10000000	✓	✓
SYSA	System alarm	—	—	$X_1X_2X_3X_4X_5X_6X_7X_8$ Xn = 0 (Normal) Xn = 1 (Fail)	X_1 = Input open X_2 = Output current open X_3 = Invalid data X_4 = Invalid data X_5 = Computed over flow X_6 = Erased RAM memory X_7 = Erased EEPROM memory X_8 = Invalid data Example: When current output is open. 01000000	✓	✓
LS1	Operation mode 1	OK	—	MAN, AUT, CAS, SPC, DDC, BUM, BUA	Can't set to BUM or BUA modes	✓	✓
LS2	Operation mode 2	OK	—	Refer to TIP 1		✓	✓
SLS1, SLS2	Sub action mode	—	—	$X_1X_2X_3X_4X_5X_6X_7X_8$ Xn = 0 (OFF) Xn = 1 (ON)	X_1 = Output tracking switch X_2 = Preset output switch $X_3 = X_4 = X_5 = X_6 = X_7 = X_8$ = Invalid data Example: When output trakcing switch is ON 10000000	✓	✓
X01 to X05	Analog input register	—	%	-25.0 to 125.0	In multi-function mode, X01 to X04 are valid.	✓	✓
Y01 to Y06	Analog output register	—	%	-25.0 to 125.0	In multi-function mode, Y01 to Y03 are valid.	✓	✓
DI01 to DI06	Status input register	—	—	0, 1 (0 = OFF, 1 = ON)	In multi-function mode, only DI1 is valid.	✓	✓
DO01 to DO16	Status output register	—	—	0, 1 (0 = OFF, 1 = ON)	In multi-function mode, DO1 to DO5 are valid.	✓	✓
P01 to P30	Computation parameter register	OK	%	-800.0 to 800.0		—	✓
T01 to T30	Temporary storage register	—	%	-800.0 to 800.0		—	✓
K01 to K30	Constant register	—	%	-800.0 to 800.0		—	✓

TableA1.1 Communication Data List (2)

Data Type (PARAM)	Name	Write-enabled	Unit	Range for Displaying (DATA)	Remarks	Mode: Multi-func.	Mode: Pro-gram
PB1, PB2	Proportional band	OK	%	2.0 to 999.9		✓	✓
TI1, TI2	Integral time	OK	sec.	1 to 9999		✓	✓
TD1, TD2	Derivative time	OK	sec.	0 to 9999		✓	✓
SFA1, SFA2	SVF constant A	OK	—	0.000 to 1.000		✓	✓
SFB1, SFB2	SVF constant B	OK	—	0.000 to 1.000		✓	✓
AG1, AG2	Variable gain	—	—	-8.000 to 8.000		—	✓
GG1, GG2	Nonlinear control gain	OK	—	0.000 to 1.000		✓	✓
GW1, GW2	Nonlinear control dead band width	OK	—	0.0 to 100.0		✓	✓
STM1, STM2	Sample control period	OK	sec.	0 to 9999		—	✓
SWD1, SWD2	Sample control time	OK	sec.	0 to 9999		—	✓
BD1, BD2	Batch deviation setpoint	OK	—	0.0 to 100.0		—	✓
BB1, BB2	Batch bias width	OK	—	0.0 to 100.0		—	✓
BL1, BL2	Batch lockup width	OK	—	0.0 to 100.0		—	✓
MR1, MR2	Manual reset	OK	%	-6.3 to 106.3		✓	✓
RB1, RB2	Reset bias	OK	%	0.0 to 106.3		✓	✓
DM1, DM2	Input compensation	—	%	-100.0 to 100.0		—	✓
FF1, FF2	Output compensation	—	%	-100.0 to 200.0		✓	✓
TRK1, TRK2	Tracking output	—	%	-6.3 to 106.3		✓	✓
PMV1, PMV2	Preset output	OK	%	-6.3 to 106.3		✓	✓
EXT	Selector control external input	—	%	-6.3 to 106.3		—	✓
SSW	Selector switch position	—	—	-8.000 to 8.000	Refer to TIP 2	✓	✓
PH1, PH2	High limit alarm setpoint	OK	%	-6.3 to 106.3		✓	✓
PL1, PL2	Low limit alarm setpoint	OK	%	-6.3 to 106.3		✓	✓
DL1, DL2	Deviation limit alarm setpoint	OK	%	0.0 to 106.3		✓	✓
VL1, VL2	Velocity limit alarm setpoint	OK	%	0.0 to 106.3		✓	✓
VT1, VT2	Velocity limit alarm time duration setpoint	OK	sec.	1 to 9999		✓	✓
MH1, MH2	Manipulated variable high limit setpoint	OK	%	-6.3 to 106.3		✓	✓
ML1, ML2	Manipulated variable low limit setpoint	OK	%	-6.3 to 106.3		✓	✓
STC	STC mode	OK	—	0, 1, 2, 3	0 = OFF, 1 = CALC, 2 = ON, 3 = ATSTUP	✓	✓
OD	On-demand command	OK	—	0, 1	0 = OFF, 1 = ON	✓	✓
IP1, IP2	Process type	OK	—	0, 1	0 = STATIC, 1 = DYNAM	✓	✓
TR1, TR2	Process response time	OK	sec.	4 to 9999		✓	✓
NB1, NB2	Noise width	OK	%	0.0 to 20.0		✓	✓
OS1, OS2	Desired control type	OK	—	0, 1, 2, 3	0 = ZERO, 1 = MIN, 2 = MED, 3 = MAX	✓	✓
MI1, MI2	Output signal width	OK	%	0.0 to 20.0		✓	✓
PMX1, PMX2	Proportional band high limit value	OK	%	2.0 to 999.9		✓	✓
PMN1, PMN2	Proportional band low limit value	OK	%	2.0 to 999.9		✓	✓
IMX1, IMX2	Integral time high limit value	OK	sec.	1 to 9999		✓	✓
IMN1, IMN2	Integral time low limit value	OK	sec.	1 to 9999		✓	✓
DMX1, DMX2	Derivative time high limit value	OK	sec.	0 to 9999		✓	✓
PA1, PA2	Proportional band calculated value	—	%	2.0 to 999.9		✓	✓
IA1, IA2	Integral time calculated value	—	sec.	1 to 9999		✓	✓
DA1, DA2	Derivative time calculated value	—	sec.	0 to 9999		✓	✓
CR1, CR2	Estimation accuracy error	—	%	0.00 to 99.99		✓	✓
RT1, RT2	Signal ratio	—	—	0.000 to 9.999		✓	✓
LM1, LM2	Equivalent dead time	—	sec.	0 to 9999		✓	✓
TM1, TM2	Equivalent time constant	—	sec.	0 to 9999		✓	✓
GM1, GM2	Equivalent process gain	—	—	0.000 to 9.999		✓	✓
STCA	STC alarm	—	—	$X_1X_2X_3X_4X_5X_6X_7X_8X_9$ $X_{10}X_{11}X_{12}X_{13}X_{14}X_{15}X_{16}$ Xn = 0 (Normal) Xn = 1(Fail)	$X_1 = X_2 = X_3 = X_4 = X_5 = X_6$ = Invalid data X_7 = SYSALM, X_8 = PVOVR, X_9 = MVLMT X_{10} = OPERR, X_{11} = IDERR, X_{12} = PWRDWN, X_{13} = PBLMT, X_{14} = TILMT, X_{15} = TDLMT, X_{16} = RTALM Example: When SYSALM occurs, 0000001000000000	✓	✓

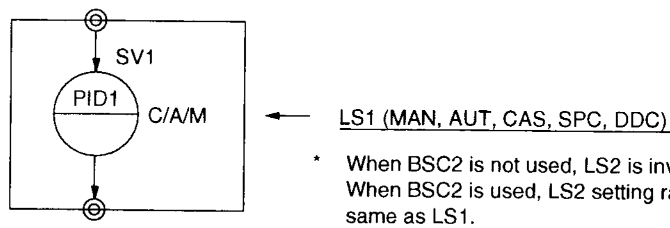
TableA1.1 Communication Data List (3)

Data Type (PARAM)	Name	Write-enabled	Unit	Range for Displaying (DATA)	Remarks	Mode	
						Multi-func.	Pro-gram
FXO1n, FXO2n	10-segment line-segment function output	OK	%	0.0 to 100.0	n = 01 to 11	✓	✓
GXI1n, GXI2n	Arbitrary line-segment function input	OK	%	-25.0 to 125.0	n = 01 to 11	—	✓
GXO1n, GXO2n	Arbitrary line-segment function output	OK	%	-25.0 to 125.0	n = 01 to 11	—	✓
PGT1n, PGT2n	Program function time set	OK	sec.	0 to 9999	n = 01 to 10	—	✓
PGO1n, PGO2n	Program function output set	OK	%	-25.0 to 125.0	n = 01 to 10	—	✓
PPID1, PPID2	Preset PID switch register	—	—	Refer to TIP 3		—	✓
PPBm	Preset PID proportional band	OK	%	2.0 to 999.9	m = 01 to 08	—	✓
PTIm	Preset PID integral time	OK	sec.	1 to 9999	m = 01 to 08	—	✓
PTDm	Preset PID derivative time	OK	sec.	0 to 9999	m = 01 to 08	—	✓
PLC1, PLC2	PV low cut square-root point	OK	%	0.0 to 100.0		✓	—
PLG1, PLG2	PV first-order lag time constant	OK	sec.	0.0 to 800.0		✓	—
CLC1, CLC2	SV low cut square-root point	OK	%	0.0 to 100.0		✓	—
CLG1, CLG2	SV first-order lag time constant	OK	sec.	0.0 to 800.0		✓	—
CGN1, CGN2	SV computational gain	OK	—	-8.000 to 8.000		✓	—
CBI1, CBI2	SV computational input bias	OK	%	-106.3 to 106.3		✓	—
CBO1, CBO2	SV computational output bias	OK	%	-800.0 to 800.0		✓	—
FLG	Output compensation first-order lag time constant	OK	sec.	0.0 to 800.0		✓	—
FGN	Computation gain	OK	—	-8.000 to 8.000		✓	—
FBI	Computation input bias	OK	%	-106.3 to 106.3		✓	—
FBO	Computation output bias	OK	%	-800.0 to 800.0		✓	—
TLG	Tracking first-order lag time constant	OK	sec.	0.0 to 800.0		✓	—
PSR1, PSR2	PV square-root switch	—	—	0, 1	0 = OFF, 1 = ON	✓	—
PFX1, PFX2	PV line-segment function switch	—	—	0, 1	0 = OFF, 1 = ON	✓	—
CSR1, CSR2	SV square-root switch	—	—	0, 1	0 = OFF, 1 = ON	✓	—
CSW1, CSW2	SV line-segment function switch	—	—	0, 1	0 = OFF, 1 = ON	✓	—
FSW	Output compensation computation switch	—	—	0, 1	0 = OFF, 1 = ON	✓	—
FON	Output compensation switch	—	—	0, 1	0 = OFF, 1 = ON	✓	—
SCH1, SCH2	Range high limit	—	—	-9999 to 9999		✓	✓
SCL1, SCL2	Range low limit	—	—	-9999 to 9999		✓	✓
SCDP1, SCDP2	Decimal point position	—	—	1, 2, 3, 4		✓	✓
CNT1, CNT2	Control module set information	—	—	Refer to TIP 4		✓	✓
SYS1, SYS2	System set information 1 or 2	—	—	Refer to TIP 5, 6		✓	✓
ID	Insturument identify	—	—	YS150, YS170		✓	✓

TIP 1

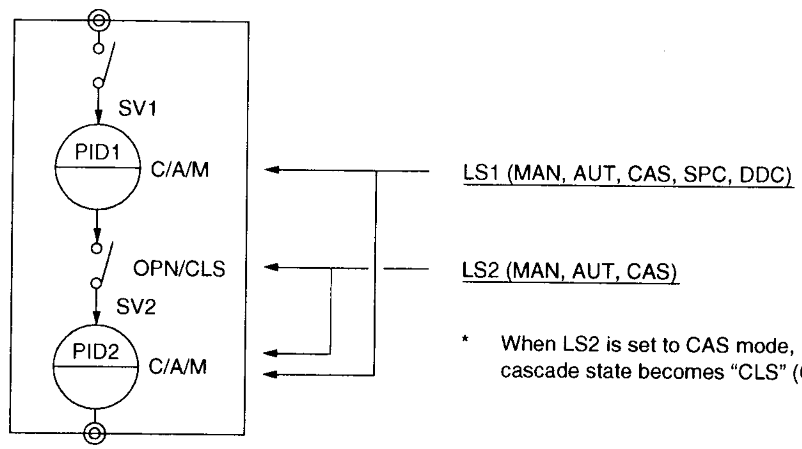
LS1 or LS2 Setting Range for Each Control Module

- For Single Loop Control
  (Single mode or when BSC 1 is used in program mode)



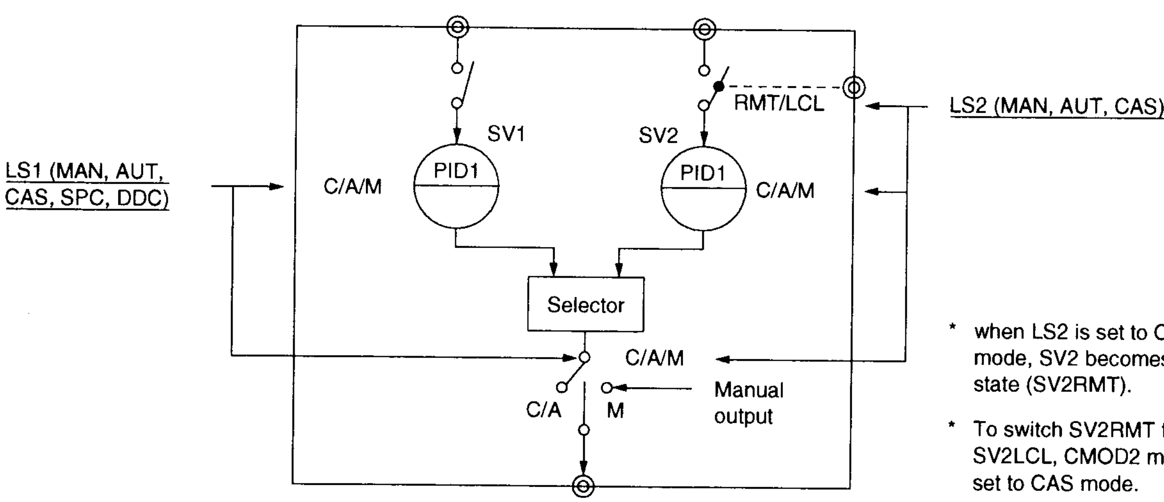
* When BSC2 is not used, LS2 is invalid.
When BSC2 is used, LS2 setting range is the same as LS1.

- For Cascade Control
  (Cascade mode or when CSC is used in program mode)



* When LS2 is set to CAS mode, internal cascade state becomes "CLS" (CLOSE).

- For Selector Control
  (Selector mode or when SSC is used in program mode)



* when LS2 is set to CAS mode, SV2 becomes remote state (SV2RMT).

* To switch SV2RMT to SV2LCL, CMOD2 must be set to CAS mode.

TIP 2

SSW Value

Data: -8.000 to 8.000

SSW Value	Selector Action
SSW < 0.500	Autoselector
0.500 < = SSW < 1.500	No. 1 Loop
1.500 < = SSW < 2.500	No. 2 Loop
2.500 < = SSW < 3.500	External signal
3.500 < = SSW	Slave

TIP 3

Switch Register for Preset PID (PPID1, PPID2)

Data: -800.0 to 800.0

Set Value to Resister	Preset PID
PPID < 0.0	Maintain present value
0.0 < = PPID < 10.0	PID Set 1 (PPB1, PTI1, PTD1)
10.0 < = PPID < 20.0	PID Set 2 (PPB2, PTI2, PTD2)
20.0 < = PPID < 30.0	PID Set 3 (PPB3, PTI3, PTD3)
30.0 < = PPID < 40.0	PID Set 4 (PPB4, PTI4, PTD4)
40.0 < = PPID < 50.0	PID Set 5 (PPB5, PTI5, PTD5)
50.0 < = PPID < 60.0	PID Set 6 (PPB6, PTI6, PTD6)
60.0 < = PPID < 70.0	PID Set 7 (PPB7, PTI7, PTD7)
70.0 < = PPID	PID Set 8 (PPB8, PTI8, PTD8)

TIP 4

Control Module Setting Information (CNT1, CNT2)

Data Type: Y1, Y2, Y3, Y4, Y5, Y6, Y7, Y8

Item \ Data		0	1	2	3
Y1	Control Type	Standard PID	Sample PI	Batch PID	Propotional (PD)
Y2	Control Computation	PV Propotional Type	PV Derivative Type	Adjustable Filter Type	———
Y3	Control Action Direction	Reverse	Direct	———	———
Y4	Open Valve Direction	C–O	O–C	———	———
Y5	C Mode	C Mode off	Cascade	Computer	———
Y6	Backup Mode	Manual Mode	Auto Mode	———	———
Y7	Engineering Unit	Max. of 6 alphanumeric digits			
Y8	Tag Number	Max. of 12 alphanumeric digits			

Note : [——] is invalid data.

TIP 5

System Setting Information 1 (SYS1)

Data Type: Y1, Y2, Y3

Item \ Data		0	1	2	3
Y1	Control Mode	Programmable	Multifunction Type Single-Loop	Multifunction Type Cascade	Multifunction Type Selector
Y2	Recovery from Power Failure	TIM1	AUT	TIM2	———
Y3	Settable by Communication	Enabled	Inhibited	———	———

Note : [——] is invalid data.

△ TIP 6

System Setting Information 2 (SYS2)

Data Type: Y1, Y2, Y3, Y4, Y5, Y6

Item	Data →	0	1	2	3	4	5	6	7
Y1	PF Key Definition	None	STC ON/OFF	——	——	——	——	——	——
Y2	DI1 Function Definition	None	A/M Toggle	C/A Toggle	Preset MV	Tracking Toggle	STC Toggle	Internal CAS Toggle	Selector Secondary Loop C/A Toggle
Y3	DI1 Set Active	Open for ON	Closed for ON	——	——	——	——	——	——
Y4	Alarm Output Status	Open	Closed	——	——	——	——	——	——
Y5	SV Tracking Definition	None	SV	PV	——	——	——	——	——
Y6	Selector Definition	Low Selector	High Selector	——	——	——	——	——	——

Note : [——] is invalid data.

Appendix 2. ERROR CODE FOR MESSAGE COMMUNICATION

This appendix describes the error codes that may appear as a response from YS100 instruments when using message communication.

Table A2.1 Error Codes

Error Codes	Contents
@011	Invalid command (but instrument address is valid)
@031	Invalid number of parameters (Not numeric, or over 3 digits long)
@032	The number of parameters is smaller than 01 or larger than 16
@033	The designated number of parameters differs from the actual number
@041	Invalid parameters name
@051	Invalid setting value

Subject to chanoe withut notice.
Printed in Japan , 601(YG)
FD No. TI 1B7C8-05E