

29 | Logic Programming

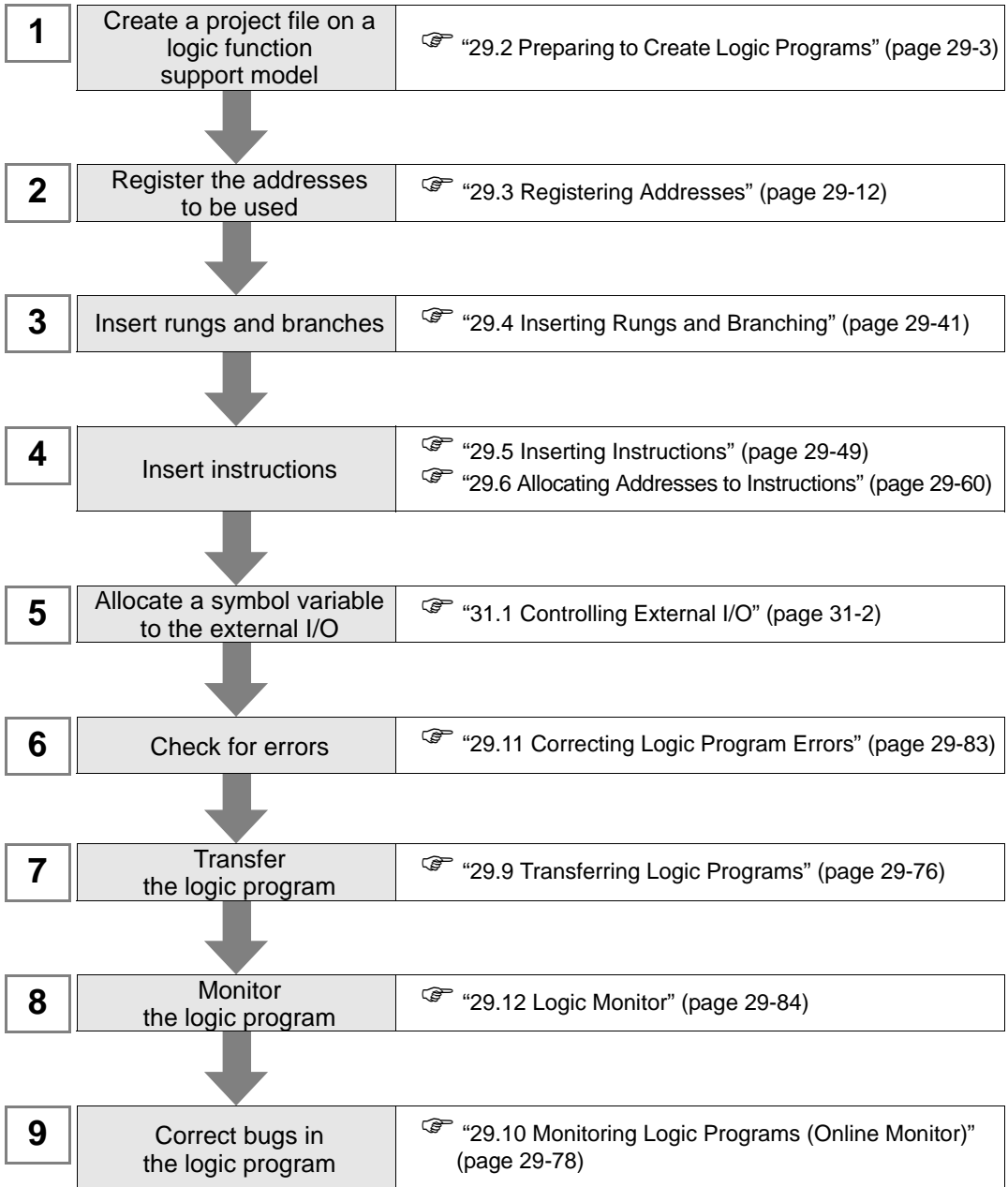
This chapter describes the addresses that you can use in GP-Pro EX and the GP, and how to program logic using the GP-Pro EX logic functions.

First read “29.1 Logic Programming Steps” (page 29-2) and then read the relevant sections.

29.1	Logic Programming Steps	29-2
29.2	Preparing to Create Logic Programs.....	29-3
29.3	Registering Addresses	29-12
29.4	Inserting Rungs and Branching.....	29-41
29.5	Inserting Instructions	29-49
29.6	Allocating Addresses to Instructions	29-60
29.7	Comment Input.....	29-66
29.8	Logic Operations with Power ON	29-74
29.9	Transferring Logic Programs.....	29-76
29.10	Monitoring Logic Programs (Online Monitor)	29-78
29.11	Correcting Logic Program Errors	29-83
29.12	Logic Monitor.....	29-84
29.13	Useful Features of the Logic Editor.....	29-93
29.14	Settings Guide.....	29-126
29.15	Restrictions	29-136

29.1 Logic Programming Steps

Use the following steps to create logic programs.



NOTE • For the complete development flow, please see the following.
☞ "4.2 Creating a Screen and Logic Program" (page 4-3)

29.2 Preparing to Create Logic Programs

-
- NOTE** • The procedure to start/end GP-Pro EX and save project files is the same as the procedure to create a new screen.
☞ “Chapter 5 Start to End” (page 5-1)
-

29.2.1 Using Logic Functions

■ If your model supports logic functions

When you create a project file and in [Display Type] select a model supporting logic functions, the logic programming settings are enabled.

-
- NOTE** ☞ “1.3 List of Supported Functions by Device” (page 1-4)
-

■ If your model does not support logic functions

When you create a project file and in [Display Type] select a model that does not support logic functions, the logic programming settings are disabled.

-
- IMPORTANT** • You can create logic programs, but you cannot transfer the programs to the GP unit.
-

-
- NOTE** • Even if you change the logic functions from [Enable] to [Disable], the logic program will not be deleted. You can also edit the logic program.
-

29.2.2 Logic Type

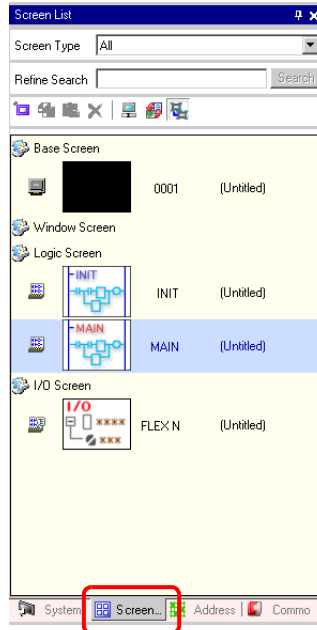
There are 3 types of logic.

Logic Type	Logic Name	Description
Initialize Logic	INIT	The logic is run only once when the GP starts up. You can create only 1 INIT program in a project file. The logic screen cannot be deleted. Start Label: "INIT START" End Label: "INIT END"
Main Logic	MAIN	The logic program is run after the initialize logic has been run. You can create only 1 MAIN program in a project file. The logic screen cannot be deleted. Start Label: "MAIN START" End Label: "MAIN END"
Subroutine	SUB-01-SUB-32	Logic is created to run the subroutine processing. You can create up to 32 subroutines in a project file. Start Label: "SUB-** START" (** = 01-32) End Label: "SUB-** RETURN" (** = 01-32)

- The Initialize Logic and Main Logic screens are created in advance.
- You can create the Subroutine logic screens on the [New Screen] menu.
- The total number of rungs in all programs, excluding the Start Label/End Label must be less than 5000.

■ Logic Screen Display

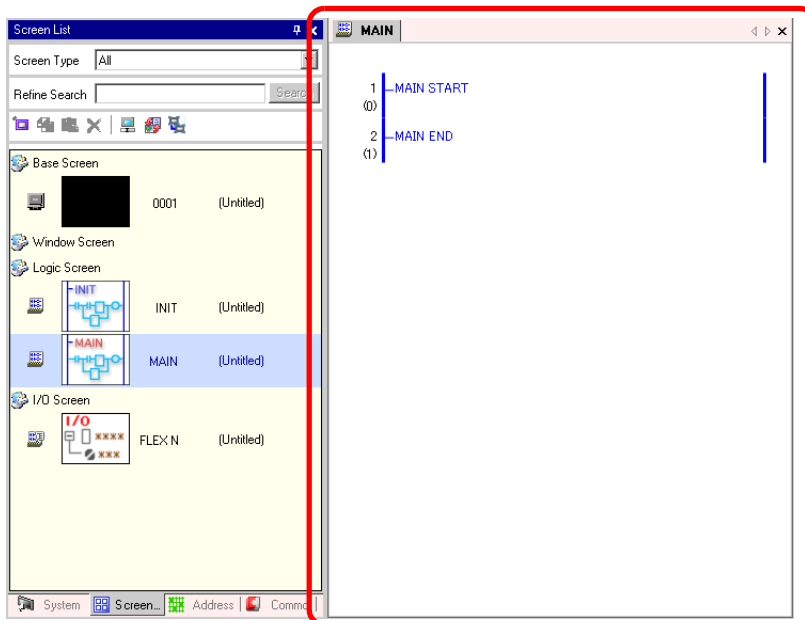
1 Click the [Screen List] tab to open the [Screen List] window.



NOTE

- If the [Screen List] tab is not displayed on the work space, on the [View (V)] menu point to [Work Space (W)] and then click [Screen List Window (G)].

2 Double-click the [MAIN] logic screen to display it in the work space.

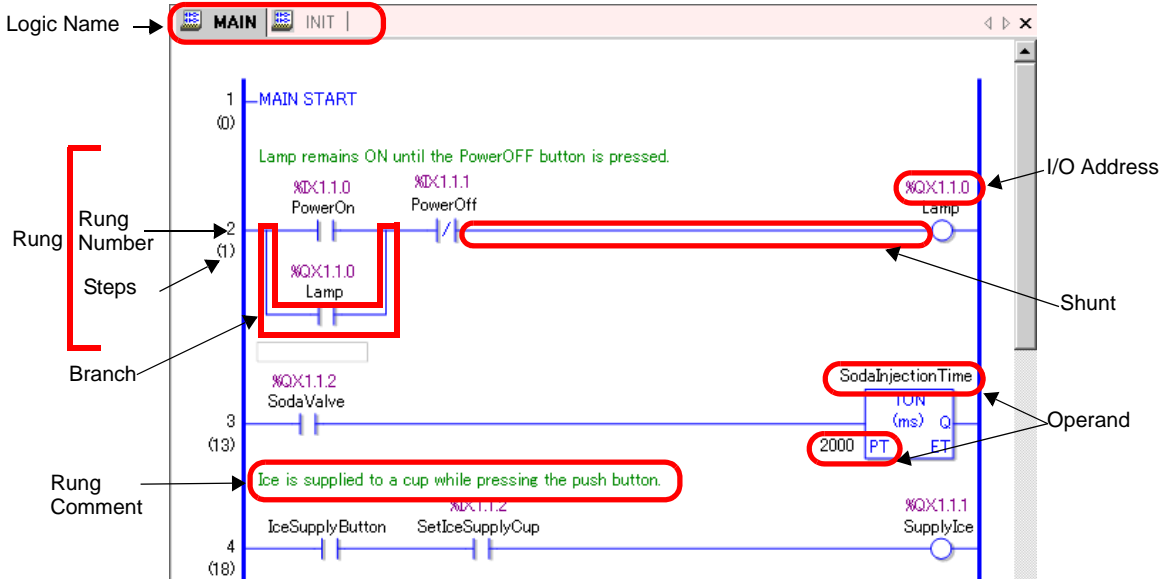


NOTE

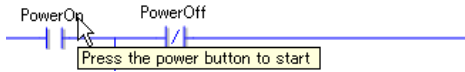
- Double-click any logic screen in the screen list to switch logic screens.

29.2.3 Logic Screens

The following explains the names of the basic elements on the logic screens.



Item	Description
Logic Name	The name of the logic screen is displayed. Click the tab to switch screens.
Rung	It consists of more than 0 instructions or 1 label. You can insert up to 99 instructions and up to 25 branches in a rung.
Rung Number	Sequential numbers with the Start Label as 1 are displayed for each rung.
Steps	Steps are the size of a logic program calculated as 6 bytes per 1 step.
Shunt	It indicates the horizontal shaft connecting the left power bar and right power bar.
Branch	This executes the logic program by connecting to rungs in parallel. ↗ "29.4.2 Inserting and Deleting Branches" (page 29-47)
Operand	This indicates the constants allocated to the instructions. ↗ "29.6.1 Operand Settings" (page 29-60)
I/O Address	This is the address value allocated to the I/O unit. The I/O address format differs depending on the allocated drivers. ↗ "Chapter 31 Controlling External I/O" (page 31-1)
Rung Comment	This is displayed when a rung has a comment. ↗ "29.7.2 Adding Rung Comments" (page 29-68)
Variable Comment	This is displayed as a tool tip when the pointer points to a variable with a comment. ↗ "29.7.3 Symbol Variable Comments" (page 29-70)

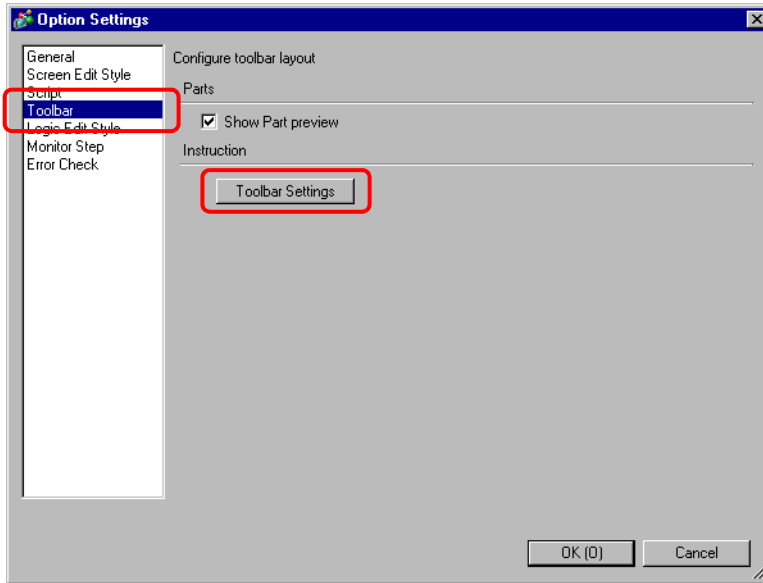


29.2.4 Customizing the Toolbar

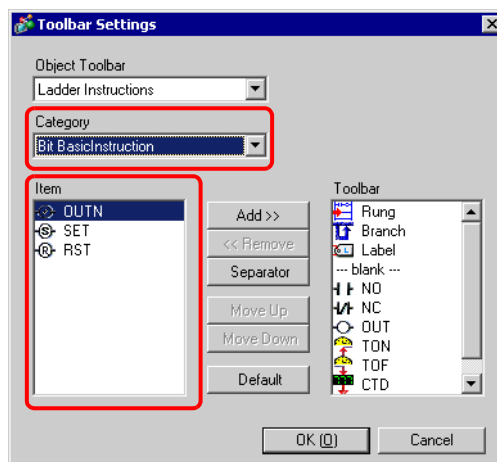
When programming, you may want to put frequently used instruction icons on the tool bar.

■ Setup Procedures

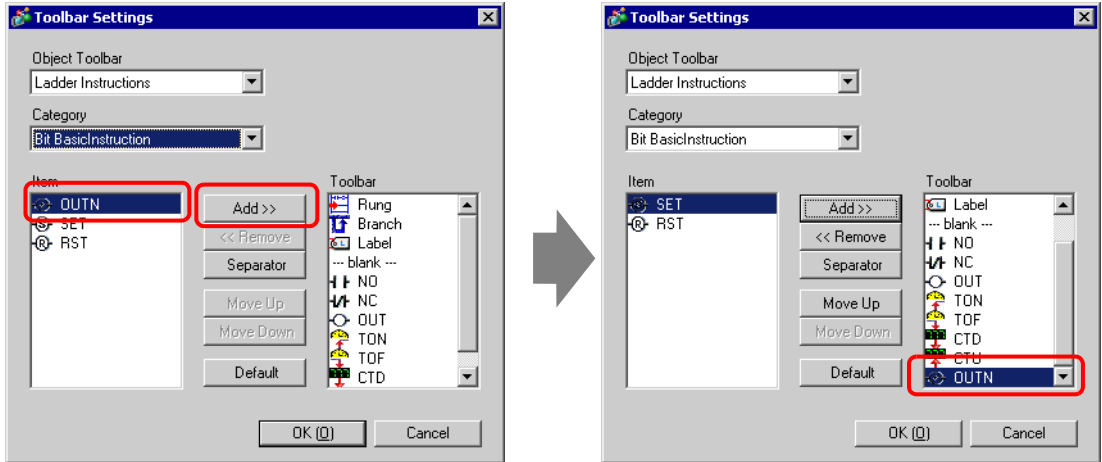
- 1 On the [View (V)] menu, click [Option Settings (O)] to display the [Option Settings] dialog box.
- 2 Select [Toolbar], then click [Toolbar Settings] to display the [Toolbar Settings] dialog box.



- 3 In [Category], select the category that you want to place on the toolbar. Instructions and their icons are displayed in [Item].



4 In [Item], select the icon and click [Add] to move the selected icon to [Toolbar].



5 Click [OK] to close the [Toolbar Settings] dialog box, then click [OK] to close the [Option Settings] dialog box. The icon will appear on the toolbar.

List of Instruction Icons

The instruction icons are as follows.































NOTE • For details on the instructions, refer to the relevant instructions.
 ☞ “Chapter 30 Instruction List” (page 30-1)

Category	Feature	Command	Icon
Basic Instructions	Normally Open	NO	↑↑
	Normally Closed	NC	↑/↓
	Out	OUT	○
	Negative Out	OUTN	○
	Set	SET	Ⓢ
	Reset	RST	Ⓡ
	Pulse Basic	Positive Transition	PT
Negative Transition		NT	↑/↓

Continued

Category		Feature	Command	Icon
Basic Instructions	Program Control	Jump	JMP	
		Jump to Subroutine	JSR	
		Return	RET	
		Repeat (Start)	FOR	
		Repeat (End)	NEXT	
		Inverse	INV	
		Exit	EXIT	
		Power Bar Control	PBC	
		Power Bar Reset	PBR	
		Logic Wait Instruction	LWA	
Operation Instruction	Arithmetic Operation	Add	ADD	
		Subtract	SUB	
		Multiplication	MUL	
		Division	DIV	
		Modulation	MOD	
		Increment	INC	
		Decrement	DEC	
	Time Operation	Time Addition	JADD	
		Time Subtraction	JSUB	
	Logical Operation	Logical AND	AND	
		Logical OR	OR	
		Logical XOR	XOR	
		Logical NOT	NOT	
	Transfer	Transfer (Copy)	MOV	
		Block Transfer (Block Copy)	BLMV	
		Multipoint Transfer (Multipoint Copy)	FLMV	
		Exchange	XCH	
	Shift	Shift Left	SHL	
		Shift Right	SHR	
		Arithmetic Shift Left	SAL	
Arithmetic Shift Right		SAR		

Continued

Category		Feature	Command	Icon
Operation Instruction	Rotation	Rotate Left	ROL	
		Rotate Right	ROR	
		Rotate Left with Carry Over	RCL	
		Rotate Right with Carry Over	RCR	
Compare Instruction	Arithmetic Compare	Comparison (=)	EQ	
		Comparison (>)	GT	
		Comparison (≥)	GE	
		Comparison (<)	LT	
		Comparison (≤)	LE	
		Comparison (≠)	NE	
	Time Compare	Time Compare (=)	JEQ	
		Time Compare (>)	JGT	
		Time Compare (≥)	JGE	
		Time Compare (<)	JLT	
		Time Compare (≤)	JLE	
		Time Compare (≠)	JNE	
	Date Compare	Date Compare (=)	NEQ	
		Date Compare (>)	NGT	
		Date Compare (≥)	NGE	
		Date Compare (<)	NLT	
		Date Compare (≤)	NLE	
		Date Compare (≠)	NNE	
Timer Instruction	—	On Delay Timer	TON	
		Off Delay Timer	TOF	
		Pulse Timer	TP	
		Accumulate On Delay Timer	TONA	
		Accumulate Off Delay Timer	TOFA	
Counter Instruction	—	Up Counter	CTU	
		Down Counter	CTD	
		Up/Down Counter	CTUD	

Continued

Category		Feature	Command	Icon
Convert Instruction	Data Convert	BCD Convert	BCD	
		BIN Convert	BIN	
		Encode	ENCO	
		Decode	DECO	
		Convert to Radian	RAD	
		Convert Degree	DEG	
		Scale	SCL	
	Type Convert	Convert Integer → Float	I2F	
		Convert Integer → Real	I2R	
		Convert Float → Integer	F2I	
		Convert Float → Real	F2R	
		Convert Real → Integer	R2I	
		Convert Real → Float	R2F	
		Convert to Seconds	H2S	
Convert Seconds to Time	S2H			
Function Instruction	Calculation Function	Sum	SUM	
		Average	AVE	
		Square Root	SQRT	
		Bit Count	BCNT	
		PID	PID	
	Trigonometric Function	Sine	SIN	
		Cosine	COS	
		Tangent	TAN	
		Arc Sine	ASIN	
		Arc Cosine	ACOS	
		Arc Tangent	ATAN	
		Cotangent	COT	
	Other Function	Exponential	EXP	
Logarithm		LN		
Log Base 10		LG10		
R/W Instruction	Time Read/Write	Read Time	JRD	
		Set Time	JSET	
	Date Read/Write	Read Date	NRD	
		Set Date	NSET	

29.3 Registering Addresses

29.3.1 Usable Addresses

On the GP-Pro EX, you can use device addresses of the connection device and the addresses of the GP data storage area.

These addresses can be used in 2 ways, as follows.

For a device address, use the connection device address or the GP address without modifications, e.g. [PLC1]X00100 or [#INTERNAL]LS0100.

For a symbol variable, you can assign a name to the connection device address or the GP address, e.g. “production quantity” or “stock quantity.”

■ Device Address

◆ Internal Address

These are temporary storage locations for saving data such as the values operated on or controlled in the GP.

NOTE

☞ “29.3.5 Using Internal Addresses” (page 29-38)

- **LS Area**
This contains free user areas, and an area for controlling the GP.
You can use this area only when communication with the connection device is established through direct access.
(e.g.) [#INTERNAL]LS0100
- **User Area**
You may use all areas as you like, up to 30,000 words.
(e.g.) [#INTERNAL]USR00100
- **Memory Link System Area**
This area acts as a medium for the host write/read request.
You can use this area only when communication with the connection device is established through a memory link.
(e.g.) [#MEMLINK]0100

◆ External Address

- **Connection Device Address**
This looks up the connection device data.
You can use this address only if communication with the connection device is established through direct access.
(e.g.) [PLC1]X00100

NOTE

☞ “29.3.4 Using External Addresses” (page 29-36)

■ Symbol Variable

There are two kinds of Symbol Variables.

- Symbol

A symbol is assigned to a device address with an arbitrary name. Each address can be managed with this name. The address of a part can also be specified with this name.

Device addresses to which symbols can be assigned: Bit address, Word address

- Variable

A variable is automatically assigned to a device inside GP-Pro EX.

There are 2 methods of registering variables as follows:

Variable format: Users can assign variables (names) as desired.

NOTE

☞ “29.3.2 Using Symbol Variables with Arbitrary Names (Variable Format)” (page 29-19)

Address format: Existing addresses are used. This method is useful when many addresses are handled, because you do not need to name addresses one by one.

NOTE

☞ “29.3.3 Using Symbol Variables with Fixed Addresses (Address Format)” (page 29-31)

■ System Variable

These variables have predefined functions. They display and control the state of the GP when a logic program is run. The system variables cannot be deleted.

NOTE

☞ “29.3.6 System Variables” (page 29-40)

☞ “A.5 System Variables” (page A-73)

■ Methods of Registering Variables

Before creating a logic program you must specify addresses which are expressed in the logic program.

There are 2 methods of registration as follows:

◆ Variable Format

With this method you can register variables with names you choose. An address is automatically allocated to the variable, saving you from having to adjust address settings and reallocate addresses, and thereby reducing programming time.

NOTE

- When creating a new project file, select [Variable Format].
 - If you selected a model that does not support logic functions, you cannot select [Address Format].
 - You can select [Address Format] only when you start creating a new program. You cannot change the method to [Address Format] during logic programming.
-

◆ Address Format

With this method, you can use only predetermined addresses. You cannot edit addresses by registering them with desired names, or by deleting or changing them.

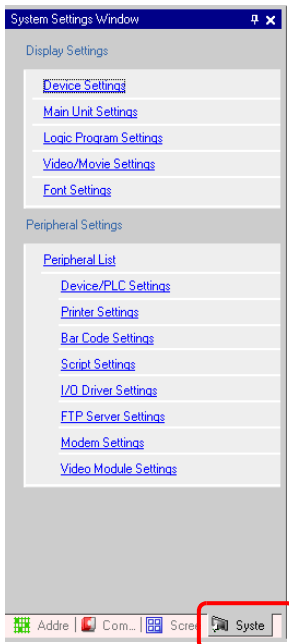
NOTE

- You can change the method from [Address Format] to [Variable Format] even during logic programming. However you cannot change back from [Variable Format] to [Address Format].
 - Even if the number of addresses used in a logic program exceeds the maximum number of variables, you can still change the method from [Address Format] to [Variable Format]. In this case, only the addresses that have been registered on the logic program are changed into variables. Note that not all addresses that can be used on the logic program are changed into variables.
-

■ Choosing the Method of Registering Variables

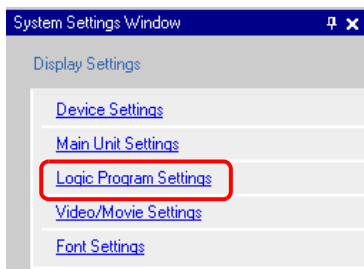
- NOTE** • Please refer to the settings guide for details.
 ☞ “29.14 Settings Guide” (page 29-126)

1 Click the [System Settings Window] tab to open the [System Settings Window].



- NOTE** • If the [System Settings Window] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings Window (S)].

2 On the [Display Settings] menu, click [Logic Program Settings].



3 In [Register Variable], select [Variable Format] or [Address Format].



■ **Keep Area Settings**

Keep Area Settings allow you to specify the variable points to keep/clear.
Please note that [Variable Format] and [Address Format] have different functions.

◆ **Variable Format**

Only the symbol variable points for keep/clear may be specified.
You can configure the keep/clear settings individually in the [Symbol Settings] window, the [Address Settings] window, or [Properties Window] after creating a symbol variable.
Please note that the clear setting is selected when creating a new variable.

◆ **Address Format**

For each symbol variable, specify the keep/clear range. With this setting, all the addresses within the keep area are kept and the addresses within the clear area are not kept.
All the symbol variables (X, Y, I, Q) allocated to the I/O are not kept (clear setting). You cannot change the setting to the keep setting.

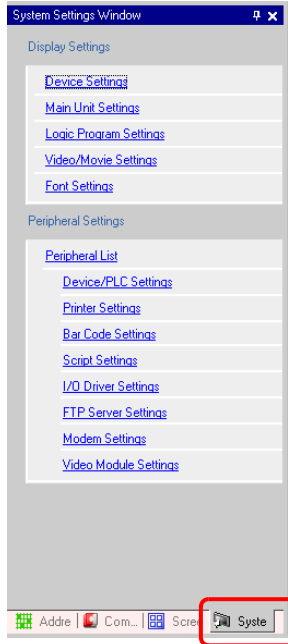
◆ **Default Settings for Keep/Clear Numbers**

Symbol Variable	Default Settings		For Address Format
	Keep	Clear	
Bit Variable	4000 pts	4000 pts	M_
Integer Variable	4000 pts	4000 pts	D_
Float Variable	64 pts	64 pts	F_
Real Variable	64 pts	64 pts	R_
Timer Variable	256 pts	256 pts	T_
Counter Variable	256 pts	256 pts	C_
Time Variable	32 pts	32 pts	J_
Date Variable	32 pts	32 pts	N_
PID Variable	8 pts	0 pts	U_

◆ Setup Procedure

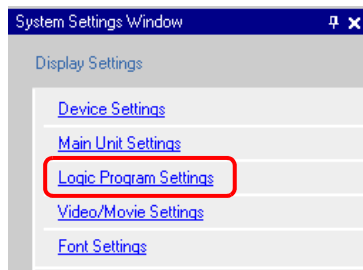
- NOTE** • Please refer to the settings guide for details.
 ☞ “29.14 Settings Guide” (page 29-126)

1 Click the [System Settings Window] tab to open the [System Settings Window].

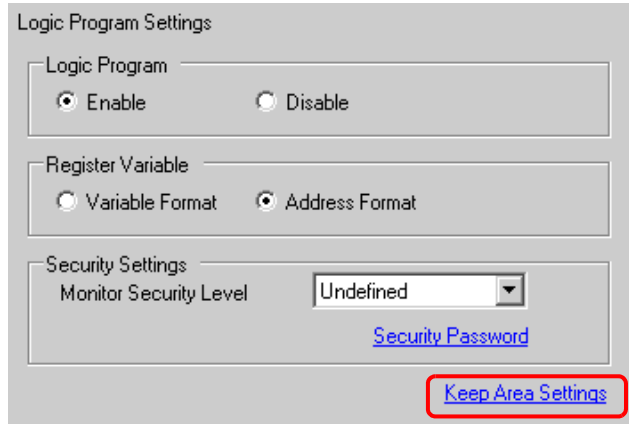


- NOTE** • If the [System Settings Window] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings Window (S)].

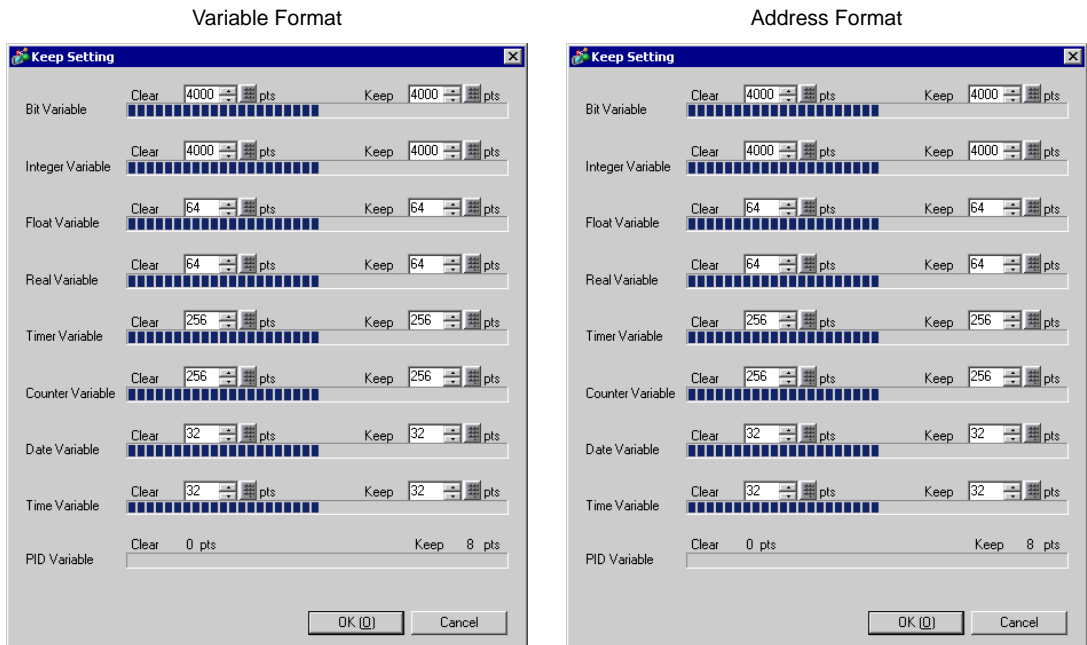
2 On the [Display Settings] menu, click [Logic Program Settings].



3 Click [Keep Area Settings] to open the [Keep Setting] dialog box.



4 Specify the points for each symbol variable.




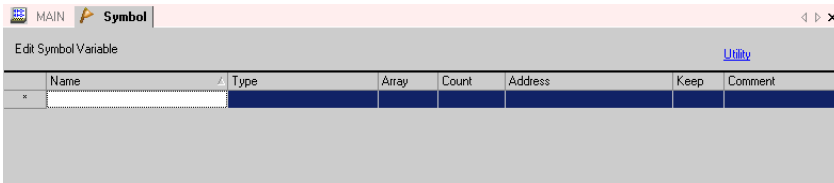
29.3.2 Using Symbol Variables with Arbitrary Names (Variable Format)

The following explains the symbol variables that you can use when [Register Variable] is set to [Variable Format].

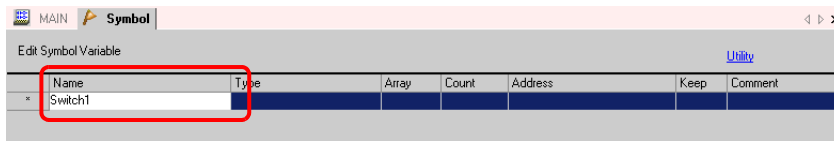
Use symbol variables that are not fixed to the hardware to create a reusable logic program.

■ Registering Symbol Variables

- 1 On the [Common Settings (R)] menu, click [Symbol Variable Settings (V)] or click  to open the [Edit Symbol Variable] window.



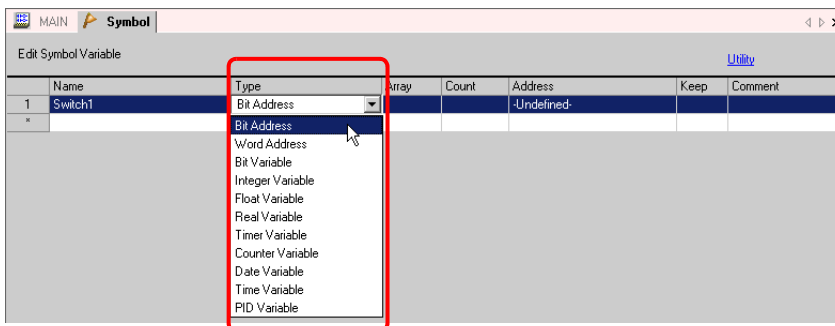
- 2 Double-click a cell in the [Name] column to column a name.



IMPORTANT

- Symbol variable names have some restrictions.
 - The maximum number of characters is 32.
 - You cannot use the following symbols.
+ - * / = % & | \ : . , # ? @ [] < > "
 - You cannot use TAB or DEL.
 - You cannot use a name starting with a single-byte number.
 - You cannot use a single-byte space.
 - You cannot leave the name blank.
 - Double-byte characters and single-byte characters are different.
 - Upper case characters and lower case characters are different.
 - You cannot name a symbol variable the same as a device address string.

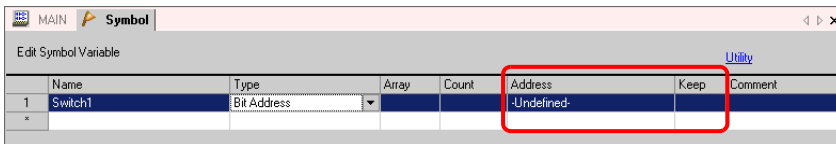
- 3 Click a cell in the [Type] column to select a type.



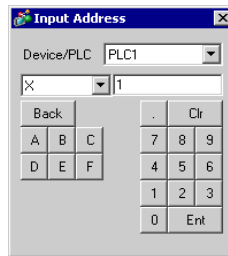
- If you selected [Bit Address] or [Word Address], you can specify the address. Then go to step 4.
- If you selected [Bit Variable], [Integer Variable], [Float Variable], or [Real Variable], you can specify the array. To specify the array, go to step 5. If not setting the array, skip to step 6.
- If you selected [Timer Variable], [Counter Variable], [Time Variable], or [Date Variable], go to step 6.
- If you selected [PID Variable], the [Keep] check box must always be selected. Then go to step 7.

NOTE • For details on the types, refer to the following.
 ☞ “ ■ Type” (page 29-22)

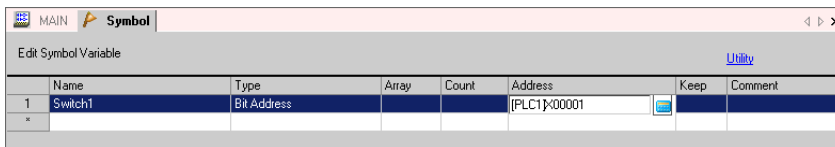
4 Specify the address on the [Address] column, then go to step 8.



Click [-Undefined-] to display a keyboard for inputting the address.

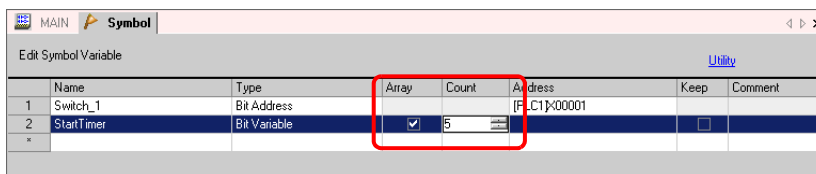


Select the connection device and device, input the address, then click “Ent”.



The address is finalized.

5 To specify the array, click a cell in the [Array] column, then select the check box to display the cell in the [Count] column. In the [Count] column, enter the array size, then go to step 6.



NOTE • For details on the arrays, refer to the following.
 ☞ “ ■ Arrays and Array Sizes” (page 29-24)

- 6 For the keep setting, click a cell in the [Keep] column and select the check box. For the clear setting, do not select the check box in the [Keep] cell.

	Name	Type	Array	Count	Address	Keep	Comment
1	Switch_1	Bit Address			[PLC1]X00001		
2	StartTimer	Bit Variable	<input checked="" type="checkbox"/>	5		<input checked="" type="checkbox"/>	comment
*							

- NOTE** • For details on the keep/clear settings, refer to the following.
 ☞ “ ■ Keep” (page 29-25)

- 7 To input a comment, click the cell in the [Comment] column and enter the comment.

	Name	Type	Array	Count	Address	Keep	Comment
1	Switch_1	Bit Address			[PLC1]X00001		
2	StartTimer	Bit Variable	<input checked="" type="checkbox"/>	5		<input checked="" type="checkbox"/>	comment
*							

- NOTE** • For details on comments, refer to the following.
 ☞ “29.7.3 Symbol Variable Comments” (page 29-70)

- 8 The registration is complete.

- NOTE** • The contents of a registered variable can be changed only when the variable is used in the logic program. A variable can be deleted only when it is not used on the screen.
- To delete, select the symbol variable and click or press DELETE.
- To register the registered symbol variable to a part placed on a new screen, refer to the following.
 ☞ “5.9 Registering Addresses with Comprehensive Names” (page 5-44)

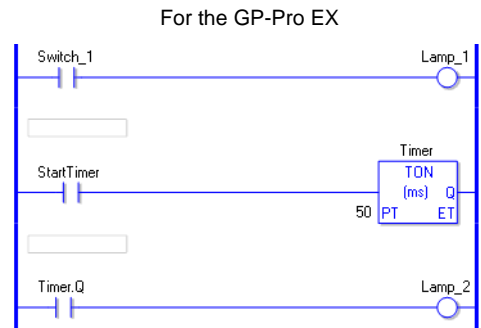
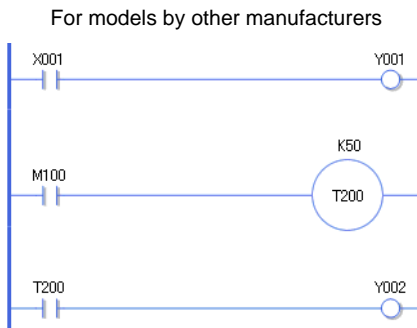
■ Symbol Variable Name

On the GP-Pro EX, you can name symbol variables and use them in the logic program without any modifications.

On general PLCs, data storage areas are handled in the same way as the device addresses under the PLC manufacturer's name.

(e.g.)

	External Input/Output	Internal Relay	Timer	Data Register
Company M	X001	M100	T200	D00001
Company O	01	1001	TIM000	DM0000
Digital Electronics Corporation	Switch1	Timer Start	Timer	Run Time



■ Type

There are 9 types of symbol variables: Bit Variable, Integer Variable, Float Variable, Real Variable, Timer Variable, Counter Variable, Time Variable, Date Variable, and PID Variable.

◆ Bit Variable

This is a variable with a 1-bit length that indicates ON/OFF with a value of either 0 (OFF) or 1 (ON).

◆ Integer Variable

This is a signed variable with a 32-bit length that has integer values of -2147483648 (16#80000000) - 2147483647 (16#7FFFFFFF).

◆ Float Variable

This is a variable with a 32-bit length that has a floating point value of $\pm 1.175494351e-38$ - $\pm 3.402823466e+38$ and 0. You may use up to 7 decimal places.


◆ Real Variable

This is a variable with a 64-bit length that has a floating point value of $\pm 2.2250738585072014e-308$ - $\pm 1.7976931348623158e+308$ and 0. You may use up to 15 decimal places.

◆ **Timer Variable**

The timer instruction uses the following 5 variables.


Special Variable	Description	Variable Settings
PT	Setting Value	32-Bit Integer
ET	Current Value	32-Bit Integer
Q	Output	Bit
TI	Time Count	Bit
R	Timer Reset	Bit

- NOTE**
- If the clear setting has been selected for the timer variable, the special variable timer PT is kept.
 “ ■ Keep” (page 29-25)

◆ **Counter Variable**

The counter instruction uses the following 7 variables.

Special Variable	Description	Variable Settings
PV	Setting Value	32-Bit Integer
CV	Current Value	32-Bit Integer
Q	Output	Bit
QD	Down Counter Output	Bit
QU	Up Counter Output	Bit
UP	Up Counter	Bit
R	Counter Reset	Bit

- NOTE**
- When scanning to reset the counter, the counter will not be updated. You must scan once to reset the counter.
 - The PV is kept if the clear setting is selected for the counter variable.
 “ ■ Keep” (page 29-25)

◆ **Date Variable**

An instruction which includes date uses the following 3 special variables.

Special Variable	Description	Variable Settings
YR	Year (0-99)	32-Bit Integer
MO	Month (1-12)	32-Bit Integer
DAY	Day (1-31)	32-Bit Integer

◆ **Time Variable**

The time instruction uses the following 3 variables.

Special Variable	Description	Variable Settings
HR	Hour (0-23)	32-Bit Integer
MIN	Minute (0-59)	32-Bit Integer
SEC	Second (0-59)	32-Bit Integer

◆ **PID Variable**

The PID instruction uses the following 11 special variables.

Special Variable	Description	Variable Settings
KP	Constant Proportion (x1000)	32-Bit Integer
TR	Integral time (x1000)	32-Bit Integer
TD	Differential time (x1000)	32-Bit Integer
PA	Processing Invalidity Range	32-Bit Integer
BA	Bias	32-Bit Integer
ST	Sampling Cycle	32-Bit Integer
Q	PID Processing Complete Flag	Bit
UO	Exceeding the Minimum Scaled Value	Bit
TO	Exceeding the Maximum Scaled Value	Bit
PF	Processing Invalidity Range Flag	Bit
IF	Integral Range Processing Flag	Bit

■ **Arrays and Array Sizes**

You can specify arrays for bit variables, integer variables, float variables, and real variables. You can specify the number of array elements up to 4,096 except float and real variables for which the number of array elements is limited to 128.

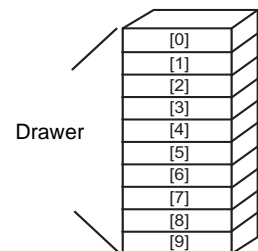
An array is a method to declare and handle multiple elements with a single variable. This allows registration of variables of the same type as a group.

Imagine the drawers of a desk or chest, for example.

A chest with an array size of 10 has 10 drawers from [0] - [9].

Each drawer is called Chest [0], Chest [1], ..., Chest [9].

Each of these drawers becomes a data register on the PLC. If 10 Chest memories are used, the array method calls the array size 10 with the symbol variable name Chest.



■ Keep

When [Keep] is selected for the symbol variables, the variables are controlled in SRAM. This allows you to keep the current value after the power is turned off.

Also, the symbol variables with [Keep] selected have default values specified by GP-Pro EX. When the power is turned off or the GP is reset, the most recent data is kept. When a logic program is downloaded, the variables are initialized to the values set with GP-Pro EX. If, however, you select [Keep Transfer] to transfer a logic program, the variables set to [Keep] are not initialized.

IMPORTANT

- Data saved in SRAM is lost when the power is turned off or the battery runs out. In such case, the values specified in GP-Pro EX are used as the default values.
-

NOTE

- Keep Area Settings allow you to specify the variable points to keep/clear.
☞ “ ■ Keep Area Settings” (page 29-16)
-

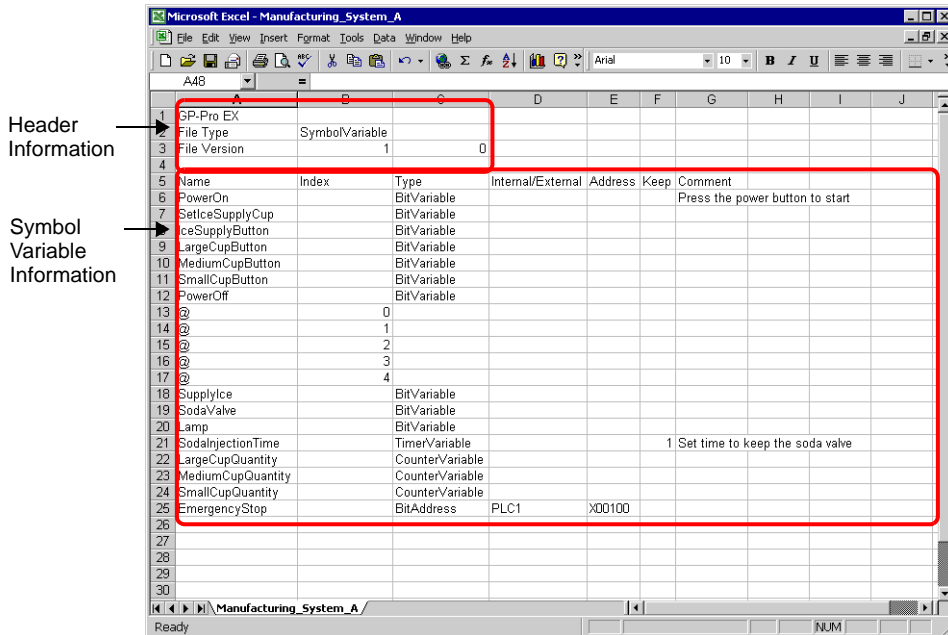
■ Importing/Exporting Symbol Variables

You can import and export a list of symbol variable settings as a CSV-format file.

The CSV format for exporting data in the symbol variable settings allows you to create or edit data using generic spreadsheet software.

◆ CSV File Format

In the [Edit Symbol Variable] window, click [Utility] and then click [Export] to output information on the symbol variable settings as a CSV-format file as follows.



- **Header Information**

The GP-Pro EX header information is attached to the exported CSV file. If any changes are made to the information, an error will occur during import. Please do not edit the information.

- GP-Pro EX : (Please do not edit.)
- File Type : SymbolVariable (Please do not edit.)
- FileVersion : The file version is saved. (Please do not edit.)

NOTE • To create new symbol variable settings in a CSV-format file, please use the above format and the header information.

- **Symbol Variable Information (Required)**

This is information on the symbol variables.

- Name : This saves the symbol variable names. To specify the array, use the “@” character for the array size starting from the next row.

NOTE • For naming restrictions, refer to the following.
 ☞ “ ■ Registering Symbol Variables” (page 29-19)

Index : Only when specifying the array, Input sequential values for the array size starting from 0.

- NOTE**
- When the symbol variable is a bit-address or word-address type, it is not used.
 - For details on arrays and array sizes, refer to the following.
☞ “ ■ Arrays and Array Sizes” (page 29-24)

Type : Input the symbol variable type using the following text.

Type	Text
Bit Address	BitAddress
Word Address	WordAddress
Bit Variable	BitVariable
Integer Variable	IntegerVariable
Float Variable	FloatVariable
Real Variable	RealVariable
Timer Variable	TimerVariable
Counter Variable	CounterVariable
Time Variable	DateVariable
Date Variable	TimeVariable
PID Variable	PidVariable

- NOTE**
- When [Register Variable] is specified as [Address Format], use bit addresses or word addresses only.
 - For details on types, refer to the following.
☞ “ ■ Type” (page 29-22)

Internal/External: Input the address category using the following text.

Category	Text	Text	Remarks
Internal Address	LS Area	#INTERNAL	You can use it only with direct access.
	User Area	#INTERNAL	
	Memory Link System Area	#MEMLINK	You can use it only with memory link.
External Address		PLC1-4	You can use it only with direct access.

- NOTE**
- When [Register Variable] is specified as [Address Format], use bit addresses or word addresses only.
 - For details on addresses, refer to the following.
☞ “29.3.1 Usable Addresses” (page 29-12)

Address : Input the address value.

- NOTE**
- When [Register Variable] is specified as [Address Format], use bit addresses or word addresses only.
 - For details on addresses, refer to the following.
 - ☞ “29.3.1 Usable Addresses” (page 29-12)

Keep : Input the keep/clear settings.

Settings	Value
Keep	1
Clear	0

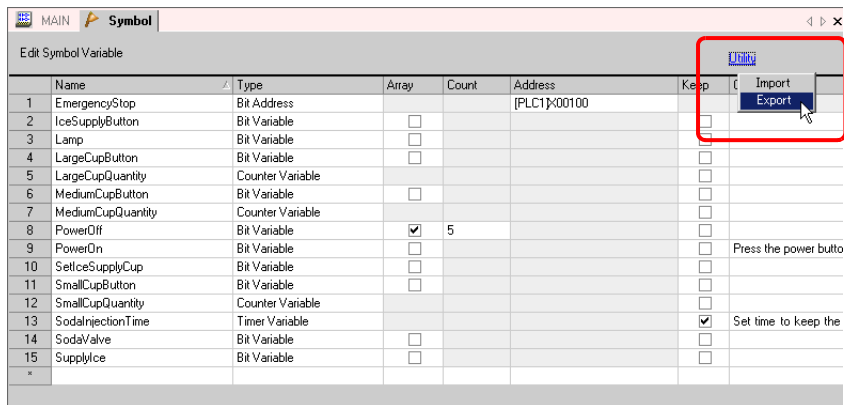
- NOTE**
- For the clear setting, you can omit the “0” value.
 - When [Register Variable] is specified as [Address Format], it is not used.
 - For details on the keep/clear settings, refer to the following.
 - ☞ “ ■ Keep” (page 29-25)

Comment : Input the comment.

- NOTE**
- For details on comments, refer to the following.
 - ☞ “29.7 Comment Input” (page 29-66)

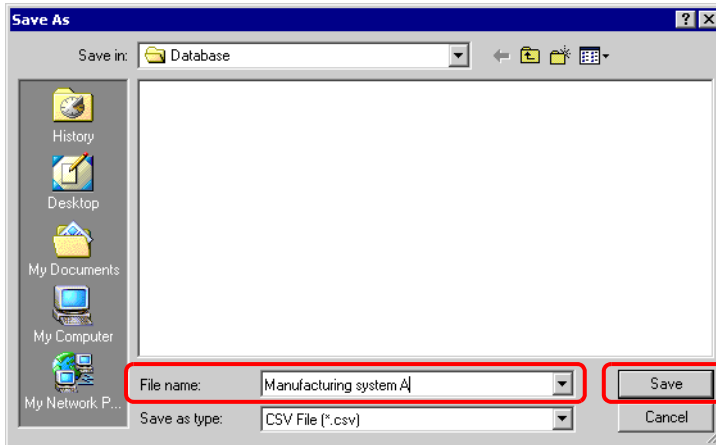
◆ Export Procedures

1 In the [Edit Symbol Variable] window, click [Utility] and then click [Export].



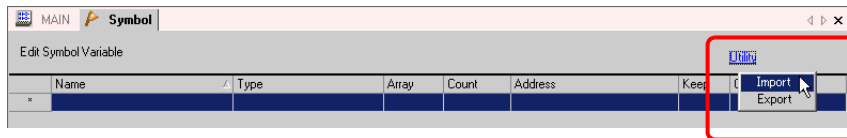
- NOTE**
- You cannot import/export system variables.

2 Specify the location to save the CSV file, enter the file name, and then click [Save].

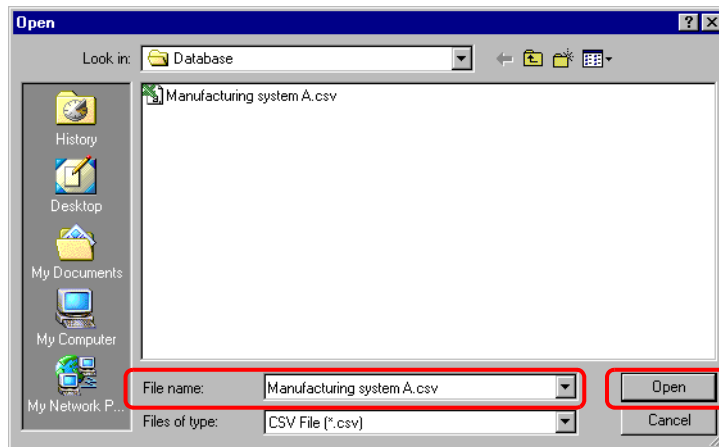


◆ Import Procedures

1 In the [Edit Symbol Variable] window, click [Utility] and then click [Import].



2 Specify the CSV file to import and then click [Open].



3 Import will be completed after error checking. If an error message is displayed, confirm the contents of the message and then click [OK].

IMPORTANT

- If the CSV file is not in the appropriate format as shown below, an error message will be displayed and import cannot be completed.
 - The name [Name] has not been input, or an inappropriate character has been used.
 - The symbol variable name already exists in the file.
 - The type [Type] has not been input, or undefined text has been used.
 - The array size has not been input, or sequential numbers have not been used.
 - The array settings have been input in a type [Type] that cannot be arrayed.
 - The keep setting "1" has been input for the bit address or word address.
 - The keep setting "1" has not been specified for the PID variable.
-


29.3.3 Using Symbol Variables with Fixed Addresses (Address Format)

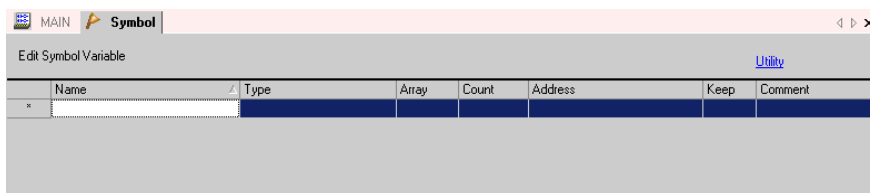
The following explains the symbol variables that you can use when [Register Variable] is set to [Address Format].

Type	Address (by the Address method)	Display	Character Size	Remarks
Bit Variable	X0000 - X0255	10 Dec	256	Input
	Y0000 - Y0255	10 Dec	256	Output
	M0000 - M7999	10 Dec	8000	Internal
Integer Variable	I0000 - I0063	10 Dec	64	Input
	Q0000 - Q0063	10 Dec	64	Output
	D0000 - D7999	10 Dec	8000	Internal
Float Variable	F0000 - F0127	10 Dec	128	Internal
Real Variable	R0000 - R0127	10 Dec	128	Internal
Timer Variable	T0000 - T0511	10 Dec	512	Internal
Counter Variable	C0000 - C0511	10 Dec	512	Internal
Date Variable	N0000 - N0063	10 Dec	64	Internal
Time Variable	J0000 - J0063	10 Dec	64	Internal
PID Variable	U0000 - U0007	10 Dec	8	Internal

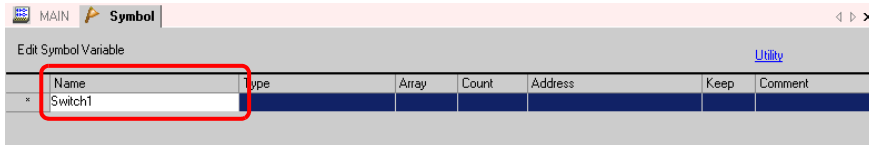
■ Registering Symbol Variables

You can name bit addresses and word addresses as you like.

- 1 On the [Common Settings (R)] menu, click [Symbol Variable Settings (V)] or click  to open the [Edit Symbol Variable] window.



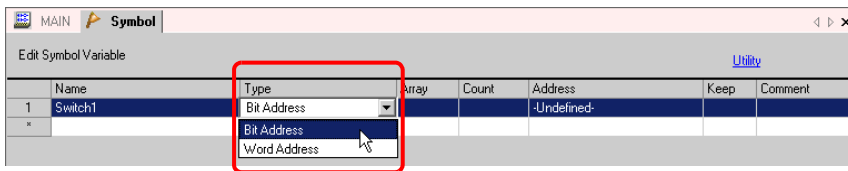
2 Double-click a cell in the [Name] column to column a name.



IMPORTANT

- Symbol variable names have some restrictions.
 - The maximum number of characters is 32.
 - You cannot use the following symbols.
+ - * / = % & | \ : . , # ? @ [] < > "
 - You cannot use TAB or DEL.
 - You cannot use a name starting with a single-byte number.
 - You cannot use a single-byte space.
 - You cannot leave the name blank.
 - Double-byte characters and single-byte characters are different.
 - Upper case characters and lower case characters are different.
 - You cannot name a symbol variable the same as a device address string.

3 Click the cell in the [Type] column and select either [Bit Address] or [Word Address] type.



NOTE

- For details on the types, refer to the following.
☞ “ ■ Type” (page 29-22)

4 Specify the address on the [Address] column, then go to step 7.

Name	Type	Array	Count	Address	Keep	Comment
1	Switch1	Bit Address		-Undefined-		

Click [-Undefined-] to display a keyboard for inputting the address.

Device/PLC: PLC1

X: 1

Back, ., Clr, A, B, C, 7, 8, 9, D, E, F, 4, 5, 6, 1, 2, 3, 0, Ent

Select the connection device and device, input the address, then click "Ent".

Name	Type	Array	Count	Address	Keep	Comment
1	Switch1	Bit Address		[PLC1]X00001		

The address is finalized.

5 To input a comment, click the cell in the [Comment] column and enter the comment.

Name	Type	Array	Count	Address	Keep	Comment
1	Switch_1	Bit Address		[PLC1]X00001		
2	StartTimer	Bit Variable	5			comment

NOTE • For details on comments, refer to the following.
 ☞ "29.7.3 Symbol Variable Comments" (page 29-70)

6 The registration is complete.

NOTE • You can only change and delete registered symbol variables that are not in use.
 To delete, select the symbol variable and click **X** or press DELETE.
 • To register the registered symbol variable to a part placed on a new screen, refer to the following.
 ☞ "5.9 Registering Addresses with Comprehensive Names" (page 5-44)

■ Logic Address Display

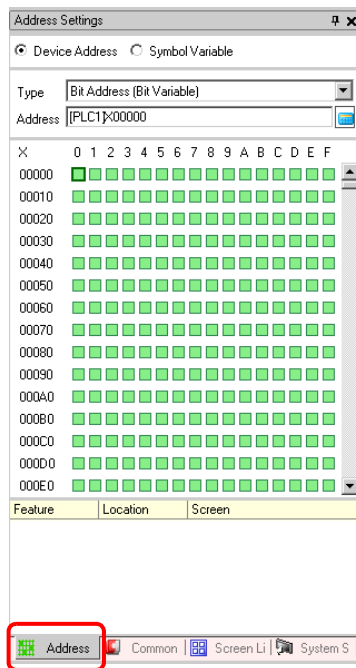
When [Register Variable] is set to [Address Format], you can use the logic addresses of bit variables and integer variables allocated within the GP-Pro EX.

(e.g.) [#LOGIC] X_0100

- NOTE** • Users cannot edit the addresses, such as register the names of their choice, or change or delete addresses.

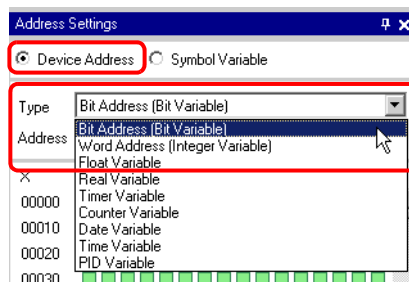
As shown below, display the logic address in the [Address Settings] window, and specify the address for the logic program instruction and the part placed on the screen.


- 1 Select the [Address Settings] tab to open the [Address Settings] window.



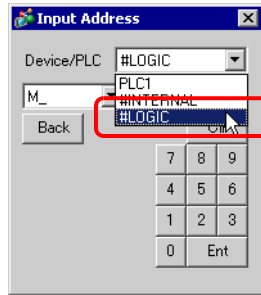
- NOTE** • If the [Address Settings] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [Address Settings Window (A)].

- 2 Select [Device Address], and in [Type], select [Bit Address (Bit Variable)] or [Word Address (Integer Variable)].

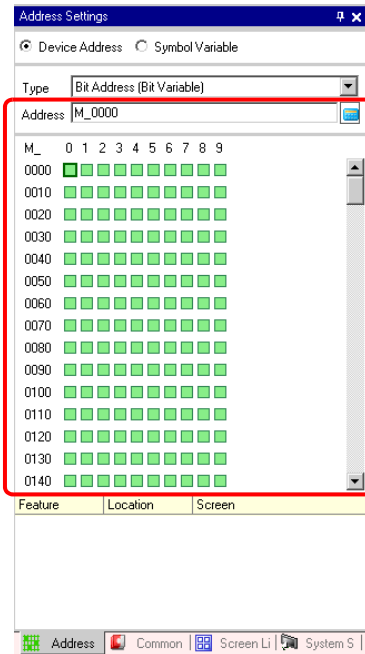


- 3 Click  to display the [Input Address] dialog box.

4 In [Device/PLC], select [#LOGIC] and the device to specify the address.



5 The logic address will be displayed. Specify the address by dragging the address to the logic program instruction or the part placed on the screen.



NOTE

☞ “ ■ Operand Settings using Drag & Drop” (page 29-63)

29.3.4 Using External Addresses

You can specify the device address if direct access is used to communicate with the connection device (PLC).


NOTE

☞ “A.1.2 Communicating with a Device/PLC without Burdening it (Direct Access Method)” (page A-3)

■ External Device Address Settings

This can be used when the symbol variable is a bit-address or word-address type.

◆ [Symbol Variable Settings] Window

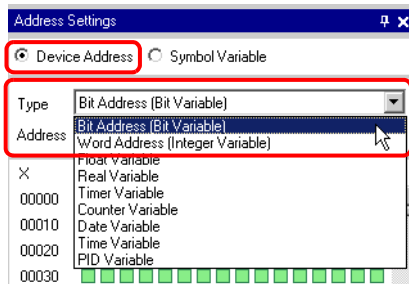
Click the cell in the [Address] column and click .


NOTE

- For the variable format, refer to the following.
☞ “■ Registering Symbol Variables” (page 29-19)
- For the address format, refer to the following.
☞ “■ Registering Symbol Variables” (page 29-31)

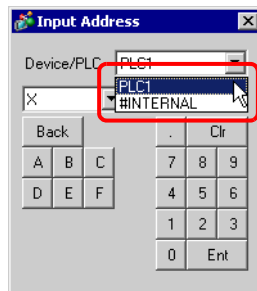
◆ [Address Settings] Window

1 Select [Device Address], and in [Type], select [Bit Address (Bit Variable)] or [Word Address (Integer Variable)].



2 Click  to display the [Input Address] dialog box.

3 In [Device/PLC], select the connection device name (e.g. PLC1) and input the address of the model (e.g. X00000).

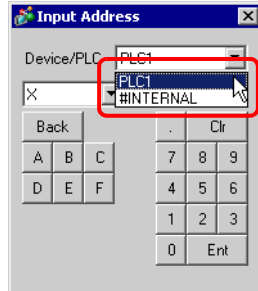


◆ Logic Screen

1 Double-click the operand and click  to display the address input box.




2 In [Device/PLC], select the connection device name (e.g. PLC1) and input the address of the model (e.g. X00000).




29.3.5 Using Internal Addresses

If direct access is used to communicate with the connection device (PLC), you can specify the addresses for the LS area and user area.

NOTE  "A.1.2 Communicating with a Device/PLC without Burdening it (Direct Access Method)" (page A-3)


When memory link is used to communicate with the connection device (PLC), you can specify the addresses for the user area and memory link system area.

NOTE  "A.1.3 Communicating with Unsupported Devices/PLCs (Memory Link Method)" (page A-5)



■ Internal Address Settings

This can be used when the symbol variable is a bit-address or word-address type.

◆ [Symbol Variable Settings] Window

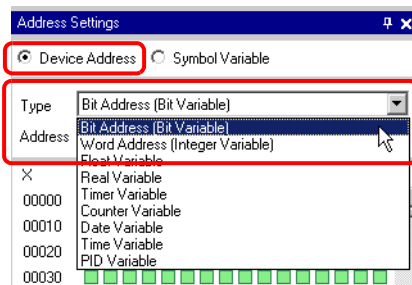
Click the cell in the [Address] column and click .


NOTE

- For the variable format, refer to the following.
 " ■ Registering Symbol Variables" (page 29-19)
- For the address format, refer to the following.
 " ■ Registering Symbol Variables" (page 29-31)

◆ [Address Settings] Window

1 Select [Device Address], and in [Type], select [Bit Address (Bit Variable)] or [Word Address (Integer Variable)].

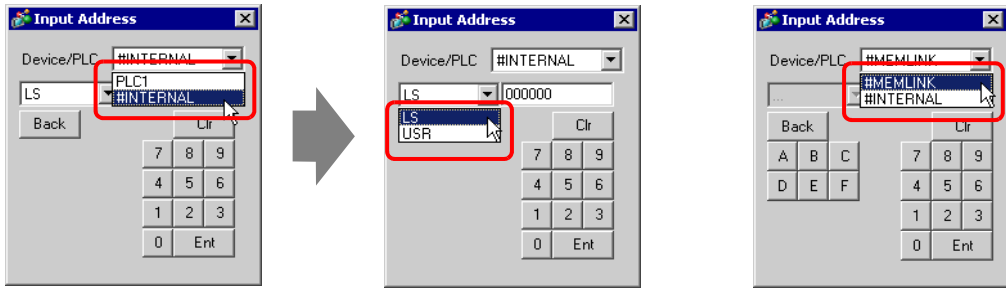


2 Click  to display the [Input Address] dialog box.

3 In [Device/PLC], select the connection device name and input the address of the model.

LS Area or User Area
 Connection Device Name (INTERNAL)
 Address (e.g. LS0000)

System Area for Memory Link
 Connection Device Name (INTERNAL)
 Address (e.g. LS0000)



◆ Logic Screen

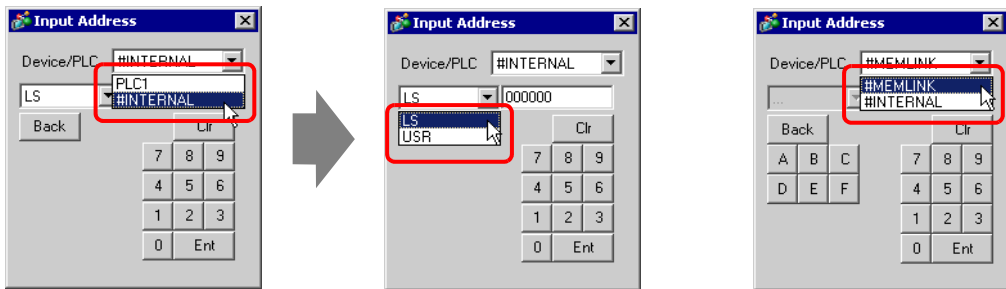
1 Double-click the operand and click  to display the address input box.



2 In [Device/PLC], select the connection device name and input the address of the model.

LS Area or User Area
 Connection Device Name (INTERNAL)
 Address (e.g. LS0000)

System Area for Memory Link
 Connection Device Name (MEMLINK)
 Address (e.g. 0000)



29.3.6 System Variables

There are two kinds of system variables. One is used for logic and the other for screens. Users cannot edit the variables, such as register the names of their choice, or change or delete addresses.

Also, there are two types of system variables. One is an integer-type and the other is a bit-type.

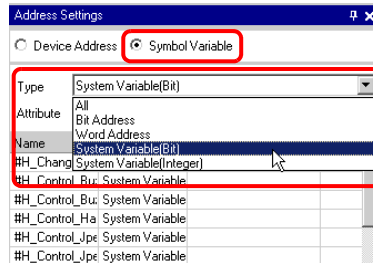
NOTE

☞ “A.5 System Variables” (page A-73)


■ System Variable Settings

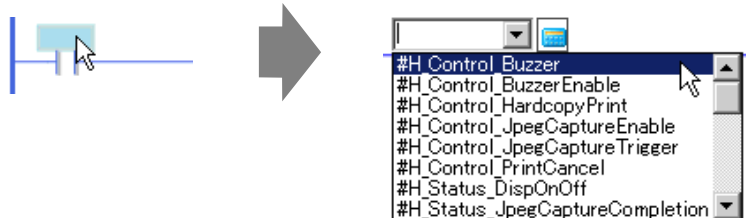
◆ [Address Settings] Window

Select [Symbol Variable] to display a list of symbol variables. Under [Type], select [System Variable (Bit)] or [System Variable (Integer)] to display the variables by types.



◆ Logic Screen

Double-click the operand and click  to select the system variable.



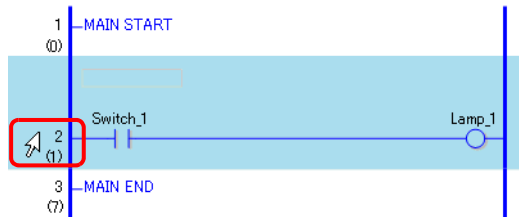
29.4 Inserting Rungs and Branching


The following explains how to edit rungs on the logic screen.

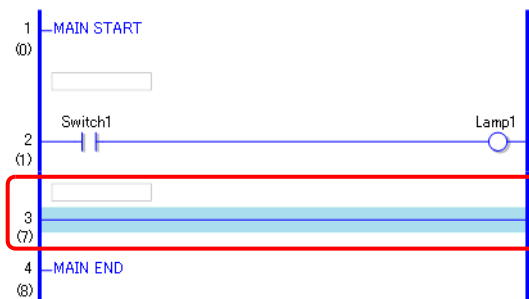
29.4.1 Editing Rungs

■ Inserting Rungs

1 Select the rung number one up from where you want to insert a rung.



2 Click  to insert a rung under the selected rung number.

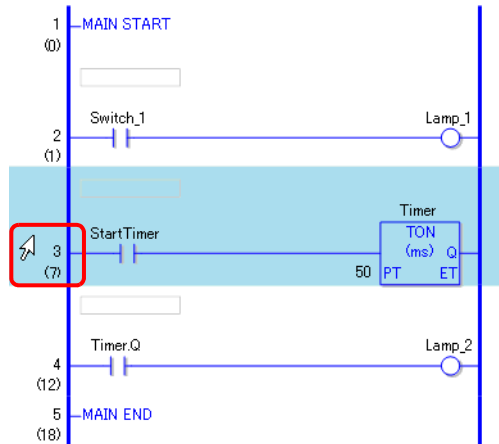


NOTE

- You can also insert a rung in any of the following ways.
 - On the [Logic (L)] menu, click [Insert Rung (R)].
 - Right-click and then click [Insert Rung (R)].
 - Press CTRL+R.

■ Deleting Rungs

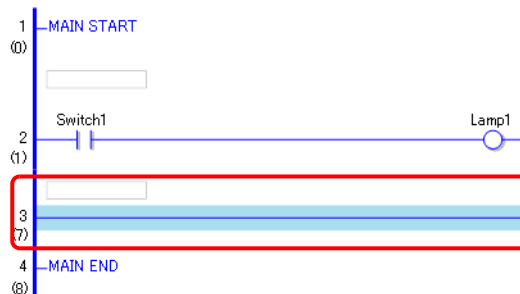
1 Select the rung that you want to delete.



-
- NOTE** • You can select a range to delete more than one rung all at once.
 ☞ “ ■ Selecting Multiple Rungs” (page 29-46)
-

2 Click .

The selected rung is deleted.

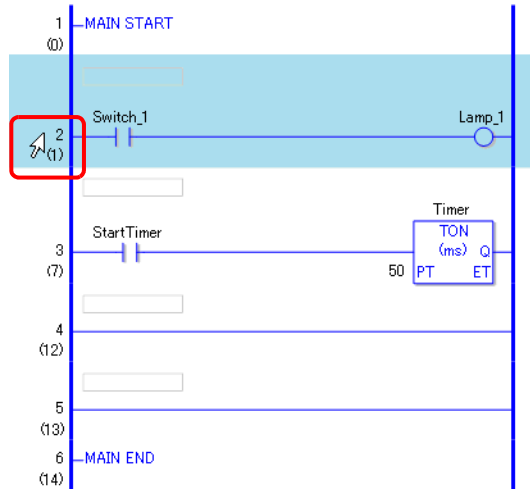


-
- NOTE** • You can also delete a rung in either of the following ways.
- Right-click and then click [Delete (D)].
 - Press DELETE.
-


■ Copying Rungs

When you want to input the same instruction sequence in more than one rung, you can save time by copying the rung already created and pasting it in the rungs.

1 Select the rung number that you want to copy.



-
- NOTE** • You can select a range to copy more than one rung at all once.
 ☞ “ ■ Selecting Multiple Rungs” (page 29-46)
-

2 Click .

The selected rung is copied to the clipboard.

-
- NOTE** • You can also copy a rung in either of the following ways.
- Right-click and then click [Copy (C)].
 - Press CTRL+C.
-

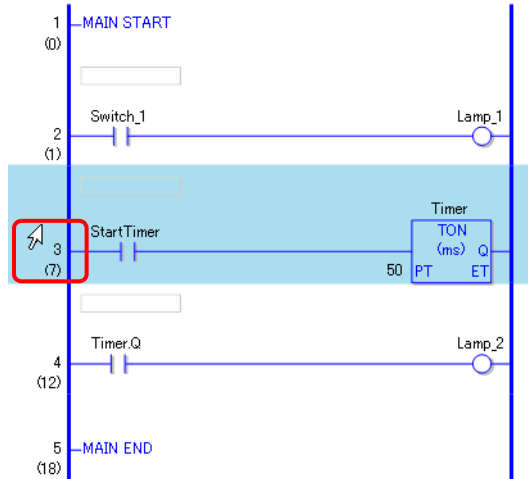
3 Paste the copied rung in the desired location.

-
- NOTE** ☞ “ ■ Pasting Rungs” (page 29-45)
-

■ Cutting Rungs

When you want to move a created rung, you can save time by cutting the rung and pasting it in the desired location.

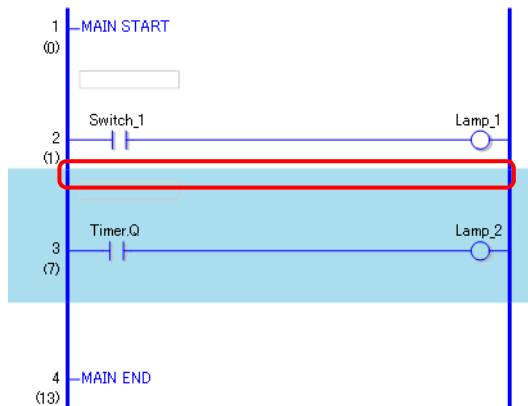
1 Select the rung number that you want to cut.



- NOTE** • You can select a range to cut more than one rung.
 ☞ “ ■ Selecting Multiple Rungs” (page 29-46)

2 Click .

The cut rung is deleted from its original location and copied to the clipboard.



- NOTE** • You can also cut a rung in either of the following ways.
- Right-click and then click [Cut (X)].
 - Press CTRL+X.

3 Paste the copied rung in the desired location.

- NOTE** ☞ “ ■ Pasting Rungs” (page 29-45)

■ Pasting Rungs

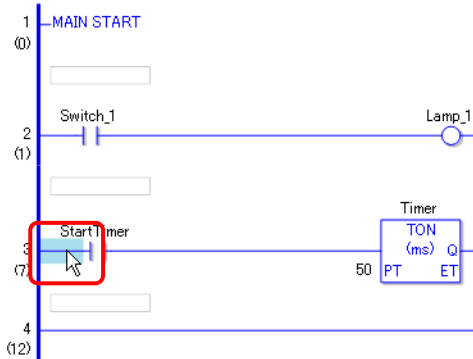
You can paste to the desired position rungs that were cut or copied.

Here, paste the copied rung between the 3rd and 4th rungs, for example.

The rung copied to the clipboard



- 1 Select a part (power bar, instruction, etc.) immediately above where you want to insert the rung.

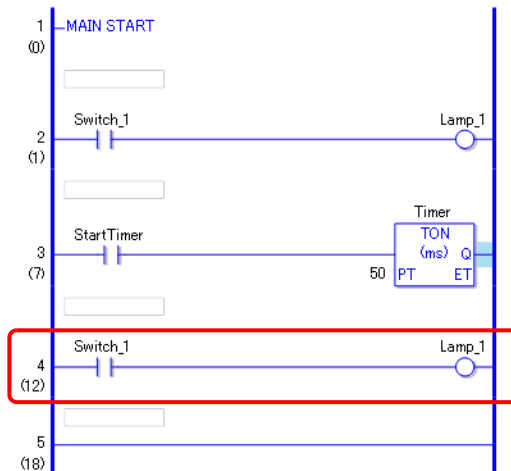


NOTE

- By clicking a rung number and selecting the entire rung, the original rung will be replaced with the copied rung.

- 2 Click .

The rung is pasted below the selected rung.



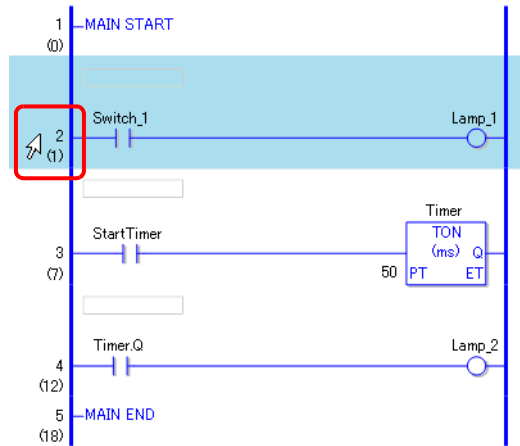
NOTE

- You can also paste a rung in either of the following ways.
 - Right-click to and then click [Paste (P)].
 - Press CTRL+V.
- When a rung is pasted, operands and rung comments in the rung instruction are also pasted. Edit the rung as required.
 - ☞ “29.6 Allocating Addresses to Instructions” (page 29-60)
 - ☞ “29.7 Comment Input” (page 29-66)

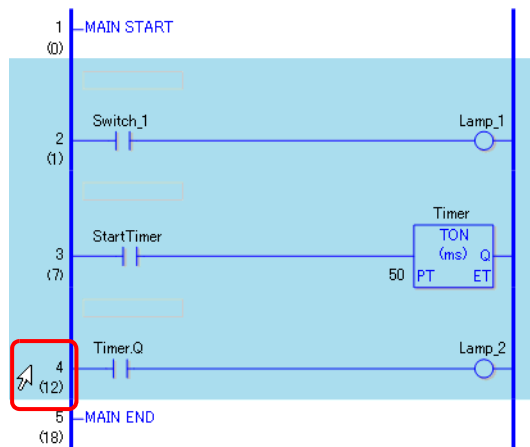
■ Selecting Multiple Rungs

You can copy and delete more than one rung by selecting a range.

1 Click the first rung number of the range that you want to select.



2 While holding down SHIFT, select the last rung number of the range. All the rungs between the two are selected.



NOTE

- You can also select a rung range in the following way.
 - While holding down SHIFT, press the [↑] key or the [↓] key to select the last rung number of the range that you want to select.
 - Press CTRL+A to select all rungs. Note that the first rung's start label and the last rung's end label will not be selected.

29.4.2 Inserting and Deleting Branches

■ Inserting Branches

The following explains how to insert a branch.

Here, a branch will be inserted between an NO instruction (Normally Open) and an NC instruction (Normally Closed) to create a self-latching logic program, for example.

1 Select where you want to start the branch.

In this case, select the left of the NO instruction (Normally Open).






2 Click . A dashed line is drawn between the start and end points of the branch.



- NOTE**
- You can also insert a branch in either of the following ways.
 - Right-click and then click [Insert Branch (B)].
 - Press CTRL+B.

3 Press the [←] or [→] key to determine the final position, and then press ENTER. The branch will be inserted.




- NOTE**
- Alternatively, you can click the start point of the branch in step 1, and drag to the right of the NO instruction (Normally Open). Release the left mouse button when the cursor changes back from  to  and the branch will be inserted. The branch end is not valid in areas where the cursor is shown as a . The branch will not be inserted after you release the left mouse button.



4 Insert an instruction in the branch.



- NOTE**  " ■ Inserting Instructions" (page 29-49)

■ Deleting Branches

To delete branches, you must delete the instructions before deleting the branches.

1 Delete the instruction.




NOTE

☞ “ ■ Deleting Instructions” (page 29-50)

2 Select the branch that you want to delete.



3 Click . The branch is deleted.



NOTE

- You can also delete a branch in either of the following ways.
 - Right-click and then click [Delete (D)].
 - Press DELETE.
-

29.5 Inserting Instructions

29.5.1 Editing Instructions


■ Inserting Instructions

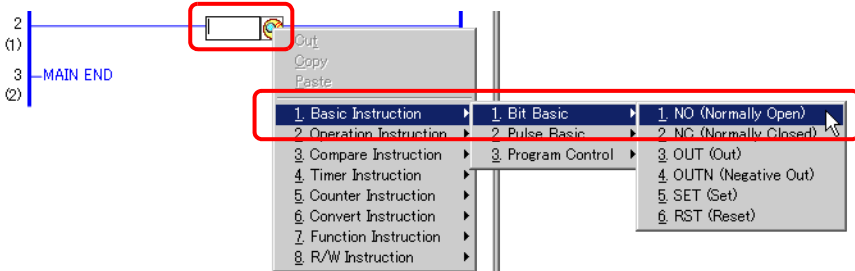
1 Select where you want to insert the instruction and, on the [Logic (L)] menu, click [Insert Instruction (I)].



NOTE

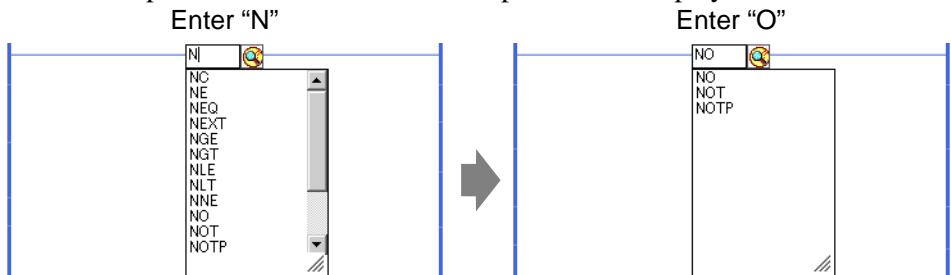
- You can also insert an instruction in the following way.
 - Double-click where you want to insert the instruction.
 - Right-click where you want to insert the instruction, and then click [Insert Instruction (I)].
 - Press INSERT.
- In the Instruction toolbar, click the instruction icon to insert the instruction immediately. You can customize instruction icons displayed in the Instruction toolbar.
 - ☞ “29.2.4 Customizing the Toolbar” (page 29-7)

2 Click  to select the instruction.



NOTE

- You can also type the instruction directly into the text box. Every time you enter a character, possible instructions for the input text are displayed.



- When [Option Settings (O)] is selected on the [View (V)] menu, the [Option Settings] window is displayed. When the [Set up operands when adding instructions] check box is selected in [Edit] under [Logic Edit Style] in the [Option Settings] window, the operand input box will appear when the instruction is inserted.

☞ “29.6.1 Operand Settings” (page 29-60)


3 The instruction is inserted.

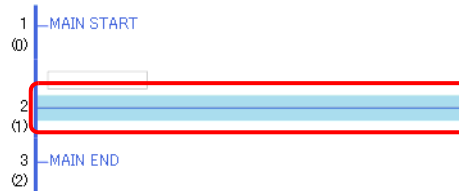


■ Deleting Instructions

1 Select the instruction that you want to delete.



2 Click . The instruction is deleted.

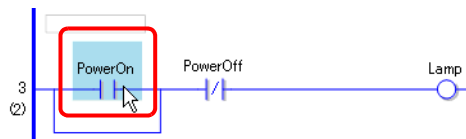


- NOTE**
- You can also delete an instruction in either of the following ways.
 - Right-click and then click [Delete (D)].
 - Press DELETE.

■ Copying Instructions

You can save time by copying the instruction and pasting it in the desired location.

1 Select the instruction that you want to copy.



2 Click .

The selected instruction is copied to the clipboard.

- NOTE**
- You can also copy an instruction in either of the following ways.
 - Right-click and then click [Copy (C)].
 - Press CTRL+C.

3 Paste the copied instruction in the desired location.

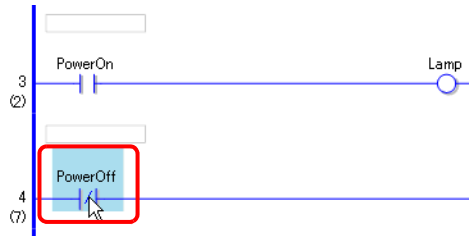
NOTE


☞ “ ■ Pasting Instructions” (page 29-52)

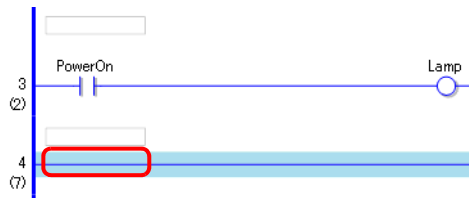
■ Cutting Instructions

When you want to move previously created instructions, you can save time by cutting the instruction and pasting it in the desired location.

1 Select the instruction that you want to cut.



2 Click . The cut instruction is deleted from its original location and copied to the clipboard.



NOTE

- You can also cut an instruction in either of the following ways.
 - Right-click and then click [Cut (X)].
 - Press CTRL+X.

3 Paste the copied instruction in the desired location.

NOTE

☞ “ ■ Pasting Instructions” (page 29-52)

■ Pasting Instructions

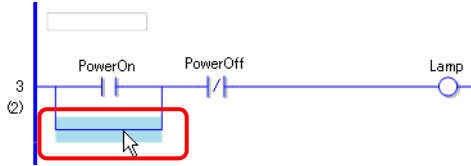
You can paste instructions that were copied/cut to the desired location.

Here, paste the copied NO instruction (Normally Open) in the branch in the 3rd rung, for example.

An instruction copied to the clipboard




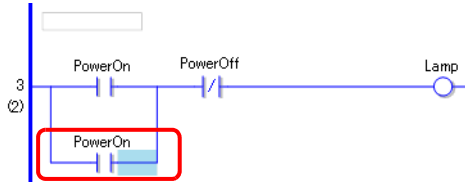
1 Select where you want to insert the instruction.



NOTE

- If you select an instruction itself, the original instruction will be replaced with the copied instruction.

2 Click . The instruction on the clipboard is pasted.



NOTE

- You can also paste an instruction in either of the following ways.
 - Right-click to and then click [Paste (P)].
 - Press CTRL+V.
- When an instruction is pasted, the operands of the instruction are also pasted. Edit the instruction as required.
 - ☞ "29.6 Allocating Addresses to Instructions" (page 29-60)

29.5.2 Subroutines and Labels

When a JSR instruction (Jump to Subroutine) or JMP instruction (Jump) is inserted, the GP jumps to the subroutine or label to execute the instruction.

Subroutines and labels have the following differences.

JSR Instruction : Executes a subroutine program with the given name and moves to the position next to the JSR instruction in the main logic program.

JMP Instruction : Jumps to a label with the name of the JMP instruction and continues to execute the logic program. It does not return to the position of the original JMP instruction and continues program execution.

-
- NOTE** • For details on JSR instructions and JMP instructions, refer to the explanation for the relevant instruction.
 ☞ “Chapter 30 Instruction List” (page 30-1)
-

■ Inserting Subroutines


Create a new subroutine screen to which the instruction will jump, and create a subroutine program on the screen.

You can insert JSR instructions anywhere in the logic program.

When the GP executes a JSR instruction, the instruction jumps to the given subroutine with the same name as itself, and the subroutine is executed.

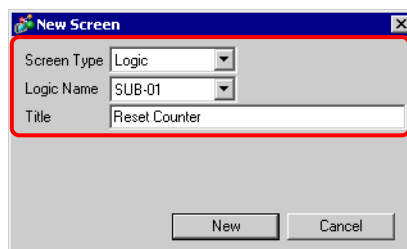
For example, a subroutine could reset counters every time the GP is turned on.

◆ Creating a Subroutine

- 1 On the normal toolbar or the [Screen List] window, click  to display the [New Screen] dialog box.

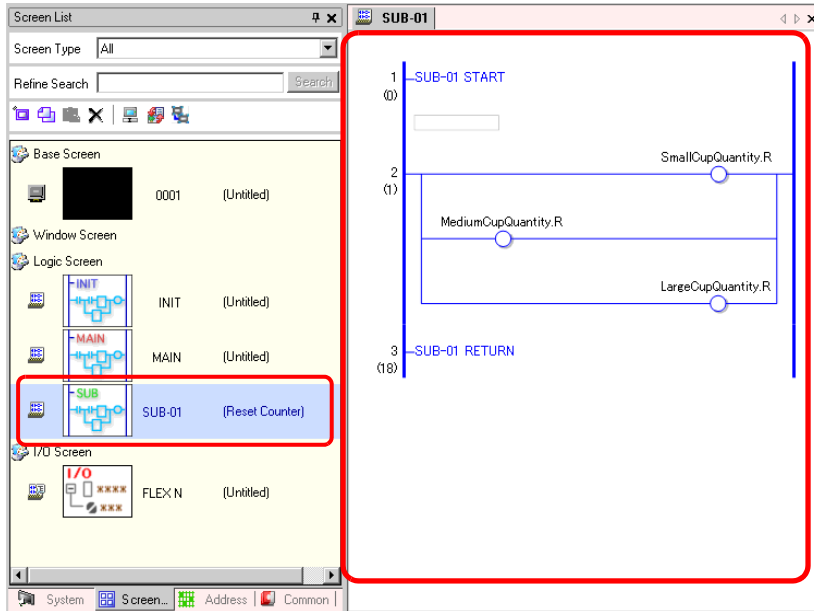
-
- NOTE** • You can display the [New Screen] dialog box in either of the following ways.
- Right-click the logic screen in the [Screen List] window and select [New Screen].
 - On the [Screen (S)] menu click [New Screen (N)].
 - If the [Screen List] tab is not displayed on the work space, on the [View (V)] menu point to [Work Space (W)] and then click [Screen List Window (G)].
-

- 2 In [Screen Type], select [Logic] and in [Logic Name], select the subroutine name (e.g. SUB-01). Input a title if desired. You can use up to 30 characters.



3 Click [New]. The subroutine screen will be displayed.

4 Create the subroutine.

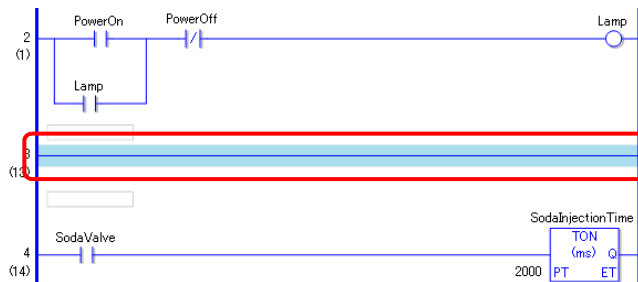


NOTE • To add more than one subroutine to a logic program, repeat steps 1-4 as many times as required to create the programs.

◆ **Inserting a JSR Instruction**

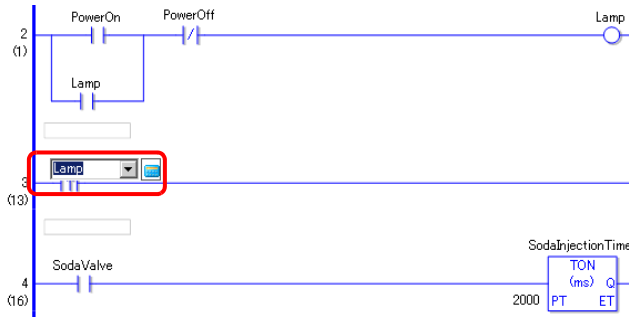
To execute the subroutine you created in a specific location in the main logic program [MAIN], you must insert a JSR instruction. Here, for example, a subroutine [SUB-01] is executed when the OUT instruction (Out) “Lamp” in the 2nd rung turns on. The JSR instruction is inserted in the 3rd rung.

1 Select the 2nd rung to enter the rung.



NOTE • For how to enter a rung, refer to the following.
 ☞ “ ■ Inserting Rungs” (page 29-41)

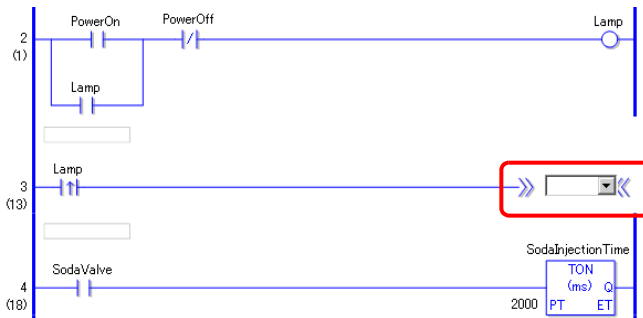
2 Insert a PT instruction in the 3rd rung (Positive Transition) and assign the symbol variable “Lamp” to the PT instruction.



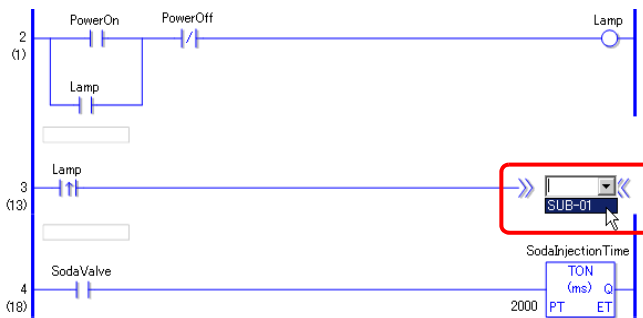
NOTE

- For how to enter an instruction, refer to the following.
 - ☞ “ ■ Inserting Instructions” (page 29-49)
- For the operand settings, refer to the following.
 - ☞ “29.6.1 Operand Settings” (page 29-60)
- For details on an instruction, refer to the explanation of the relevant instruction.
 - ☞ “Chapter 30 Instruction List” (page 30-1)

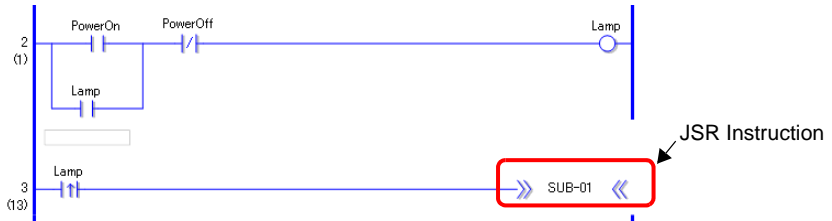
3 Insert the JSR instruction to the right of the PT instruction.



4 For the JSR instruction operand, specify the [SUB-01] subroutine.



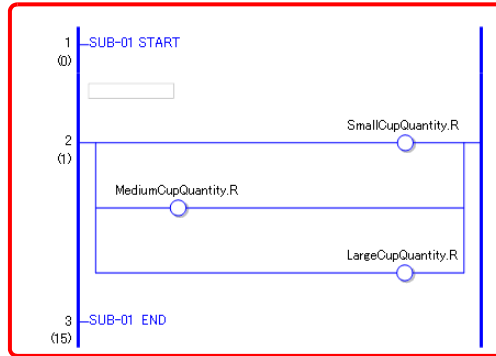
When the “Lamp” ON is detected, the operation jumps to the [SUB-01] subroutine. After the [SUB-01] subroutine is executed, the main logic program [MAIN] resumes in the 4th rung.



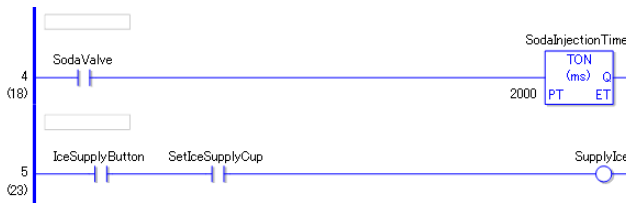
Subroutine Screen [SUB-01]



After the 3rd rung of the main program is executed, the operation jumps to the subroutine [SUB-01] upon execution of the JSR instruction.



After the [SUB-01] subroutine is executed, the operation returns to run the main program in the 4th rung.



■ Inserting Labels

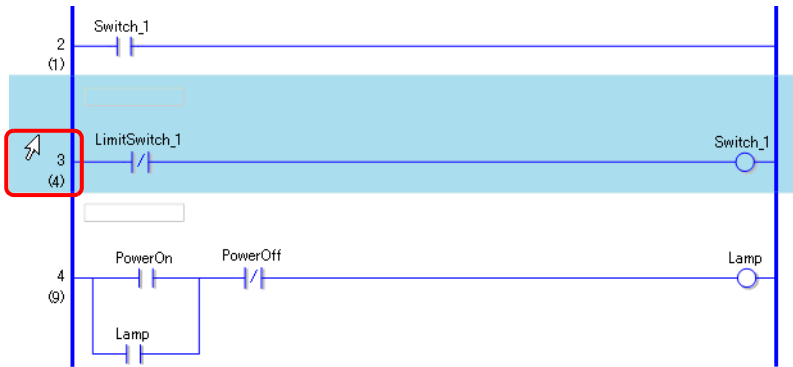
You can insert JMP instructions (Jump) and jump labels anywhere in the logic program.


When the GP executes the JMP instruction, the operation jumps to the label with the same name as the instruction, and the logic program continues to run.

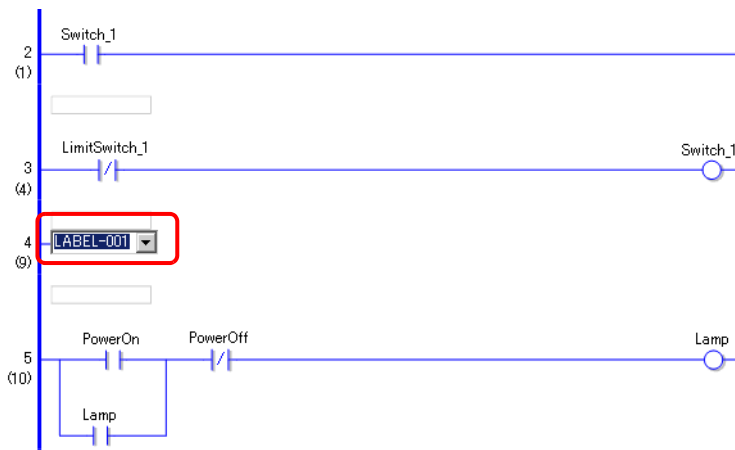
Here, for example, a [LABEL-001] label is inserted in the jump instruction, and the operation jumps to the 3rd rung upon execution of the JMP instruction when the “Switch_1” in the 2nd rung turns on.

◆ Inserting a Label

1 Select the 2nd rung.



2 Click . The label is inserted in the 3rd rung.



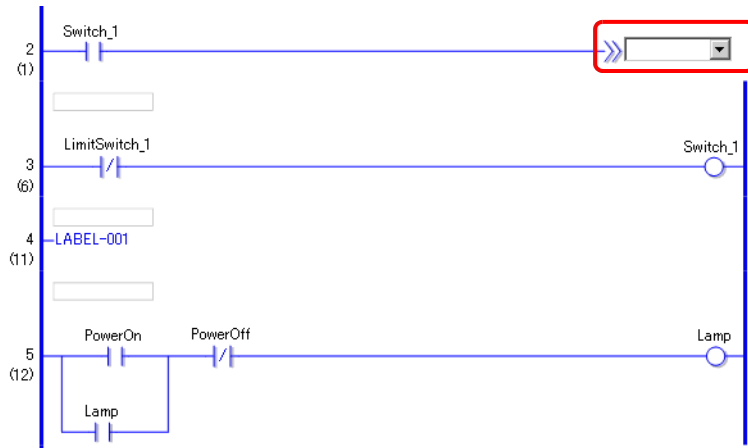
NOTE

- You can also insert a label in any of the following ways.
 - On the [Logic (I)] menu, click [Insert Label(L)].
 - Right-click and then click [Insert Label (L)].
 - Press CTRL+L.

3 Select the label name (e.g. LABEL-001).

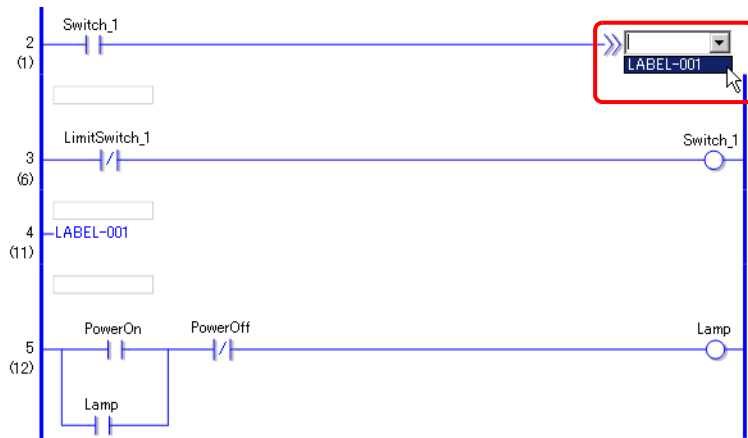
◆ **Inserting a JMP Instruction**

1 Insert a JMP instruction to the right of “Switch_1” of an NO instruction (Normally Open) in the 2nd rung.



NOTE • For how to enter an instruction, refer to the following.
 ☞ “ ■ Inserting Instructions” (page 29-49)

2 Specify an operand label [LABEL-001] for the JMP instruction operand.



When the “LABEL-001” JMP instruction is detected, the operation jumps to the [LABEL-001] label. After the [LABEL-001] label is executed, the logic program runs from the next rung.

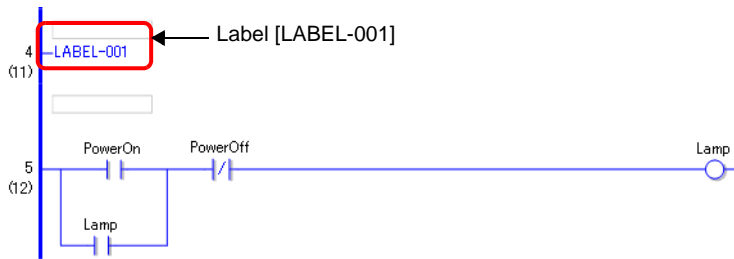


When “Switch_1” is on, the JMP instruction “LABEL-001” is detected and the operation jumps to the [LABEL-001] label in the 4th rung.

The 3rd rung will not be executed.



Executes from the 4th rung.



NOTE

- Specify the same name for the variables and labels allocated to the JMP instruction. The instruction will jump to the label with the same name.

29.6 Allocating Addresses to Instructions

- NOTE** • For details on the instructions, refer to the relevant instructions.
 ☞ “Chapter 30 Instruction List” (page 30-1)

29.6.1 Operand Settings

The following explains how to allocate values and symbol variables (addresses) to instructions.

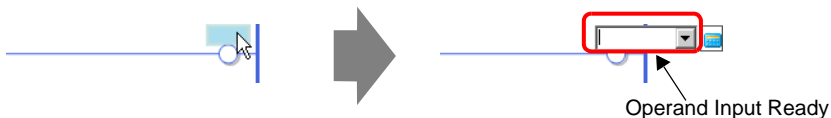
- NOTE** • When the [Set up operands when adding instructions] check box is selected under [Edit] in [Logic Edit Style] in the [Option Settings] window, the operand input box will appear when the instruction is inserted.
- ☞ “5.13.7 [Option Settings] Settings Guide ■ Logic Edit Style” (page 5-133)
- You can change the operand details using the property window.
- ☞ “29.13.5 Using Reference Features to Search Logic Programs” (page 29-118)

■ Setting Operands, Basic

When an instruction is inserted in a rung, the operand input box will appear. In the box, enter the value and symbol variable (address) to be linked to the instruction.

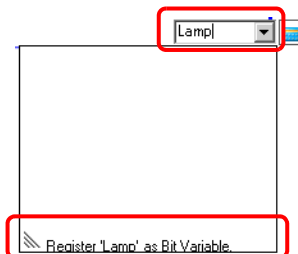
Here, for example, the symbol variable “Lamp” is allocated to an OUT instruction (Out).

- 1 Double-click the OUT instruction (Out) operand. A text box will be displayed and the operand input box will appear.

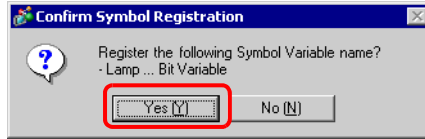


- NOTE** • You can display the operand input box in either of the following ways.
- Right-click the operand and then click [Edit (E)].
 - Select the operand and press ENTER.

- 2 Type “Lamp” in the text box and press ENTER.
 A message appears: “Register ‘Lamp’ as Bit Variable.”




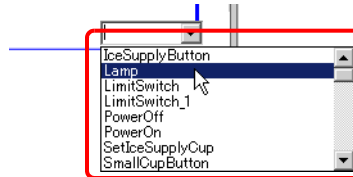
3 Press ENTER again to display the [Confirm Symbol Registration] dialog box. Click [Yes].




4 The symbol variable type necessary for the instruction will be automatically allocated. In this case, a “bit variable” type is allocated.



- NOTE**
- If symbol variables that can be allocated were previously registered, or if the system variables can be allocated, click  to display those variables. You can select and specify the displayed symbol variables and system variables.



- To directly enter the address, click .

GP-Pro EX automatically assigns the type necessary for the new symbol variable created for the instruction.

■ Setting Operands, Advanced

Advanced instructions have more than one operand.

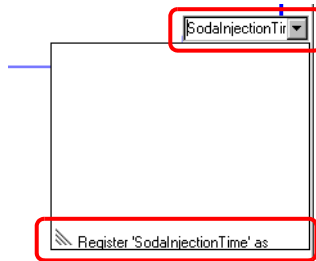
Here, for example, the operand-setting procedures for an ON instruction (On Delay Timer) are explained. A symbol variable “soda injection time” is allocated to an operand and a setting time [Setting Time (PT)] is allocated to another operand. The setting time is used when the timer output (Q) turns on.

1 Double-click the TON instruction (On Delay Timer) operand. A text box is displayed for entering the operand.

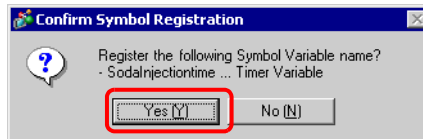


- NOTE**
- You can display the operand input box in either of the following ways.
 - Right-click the operand and then click [Edit (E)].
 - Select the operand and press ENTER.

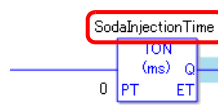
- Input “Soda Injection Time” in the text box and press ENTER to input the text.
A message appears: “Register ‘Soda Injection Time’ as Timer Variable.”




- Press ENTER again to display the [Confirm Symbol Registration] dialog box.
Click [Yes].

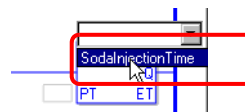


- The symbol variable type necessary for the instruction will be automatically allocated.
In this case, a “timer variable” type is allocated.

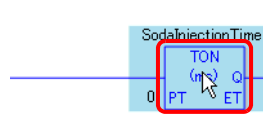


NOTE

- If symbol variables that can be allocated were previously registered, or if the system variables can be allocated, click  to display those variables. You can select and specify the displayed symbol variables and system variables.



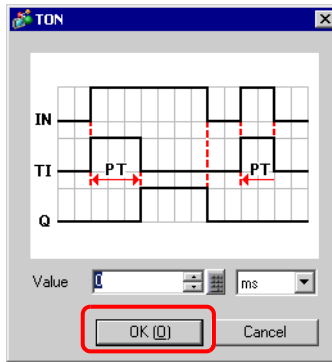
- The default value “0” is input in the setting value [Setting Time (PT)]. Double-click the instruction to change the [Setting Time (PT)].



NOTE

- Right-click and then click [Instruction Settings] to display the setting dialog box.
- If no symbol variable is input in Procedure 1, the setting dialog box will not be displayed.
- For some instructions, the setting dialog box might not be displayed.
- The setting dialog box differs depending on the instruction.

- The setting dialog box for the TON instruction (On Delay Timer) is displayed. Change the settings as necessary and click [OK].

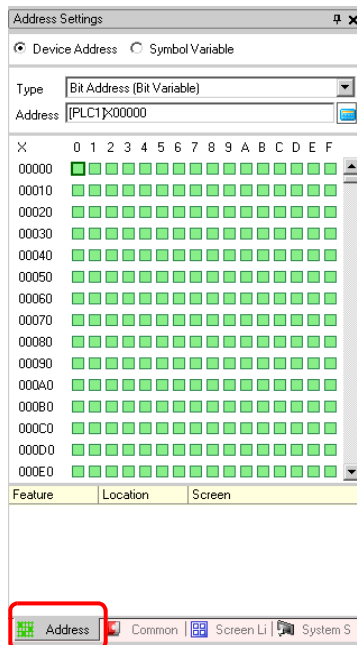


■ Operand Settings using Drag & Drop

When the symbol variable has already been registered for the instruction, you can drag the variable from the [Address Settings] window to specify the operand.

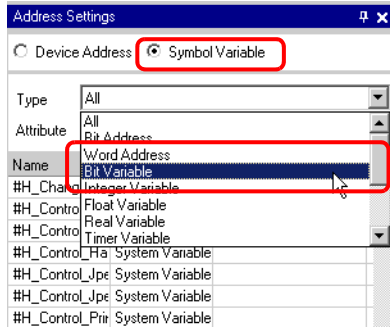
Here, an NO instruction (Normally Open) is specified for the symbol variable “power on button” of the “bit variable” type.

- Select the [Address Settings] tab to open the [Address Settings] window.

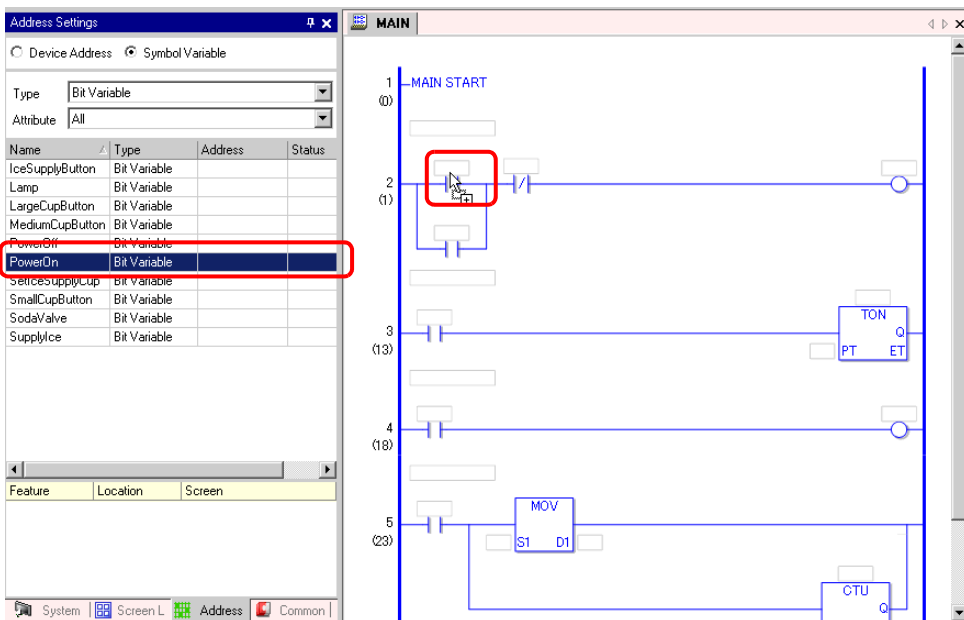


NOTE • If the [Address Settings] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [Address Settings Window (A)].

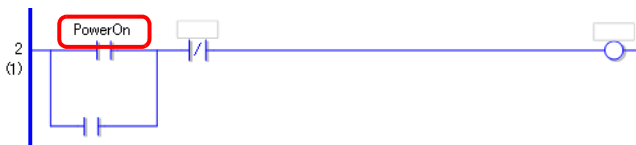
2 Select [Symbol Variable], and in [Type], select [Bit Variable].



3 Only symbol variables whose [Type] are [Bit Variable] are displayed. On the list, click [PowerOn]. Drag the variable to the instruction operand to which you want to allocate it. Release the left button when the cursor changes from to .

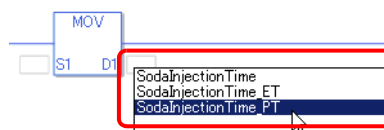


4 The symbol variable has been allocated to the instruction operand.



NOTE

- When [Array] with [Bit Variable], [Integer Variable], [Float Variable], and [Real Variable] is selected for the variable type, or when the type is [Timer Variable], possible variables are displayed for [Counter Variable], [Time Variable], [Date Variable], and [PID Variable] that consist of more than one variable. Select the relevant variable from among these.



■ Pulse Settings

You can change instructions into pulse settings as follows.

- 1 Right-click the instruction that you want to change and then click [Pulse Settings].



- 2 The instruction is changed into a pulse setting.



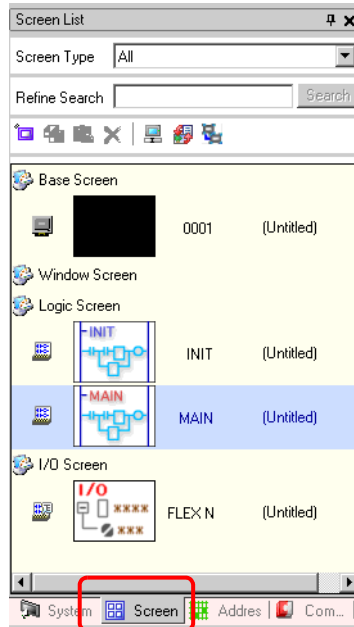
NOTE • To cancel the pulse setting, right-click to and then click [Remove Pulse].

29.7 Comment Input


GP-Pro EX allows you to add logic program titles and comments to rungs and symbols variables. Comments improve readability and are useful when debugging and making changes.

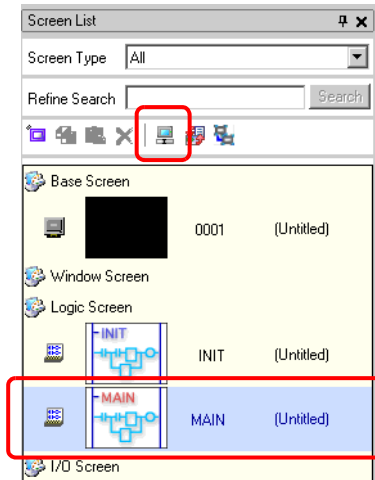
29.7.1 Adding Titles

- 1 Click the [Screen List] tab to open the [Screen List] window.



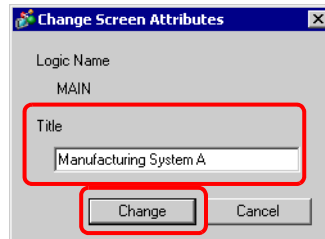
-
- NOTE** • If the [Screen List] tab is not displayed on the work space, on the [View (V)] menu point to [Work Space (W)] and then click [Screen List Window (G)].
-

2 Select the logic screen to which you want to add the title and click . The [Change Screen Attributes] dialog box is displayed.

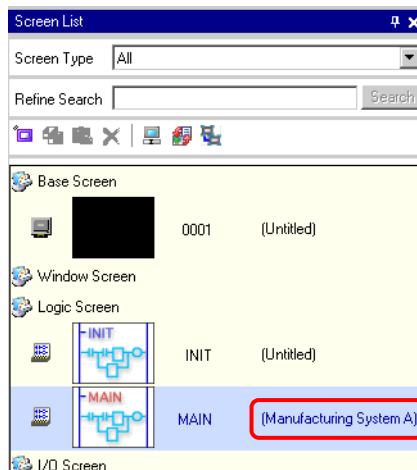



NOTE • You can also right-click the logic screen in the [Screen List] window and then click [Change Attribute] to display the [Change Screen Attributes] dialog box.

3 Enter the title in [Title] and click [Change].
You can use up to 30 characters.



4 The title is displayed on the left of the logic screen.

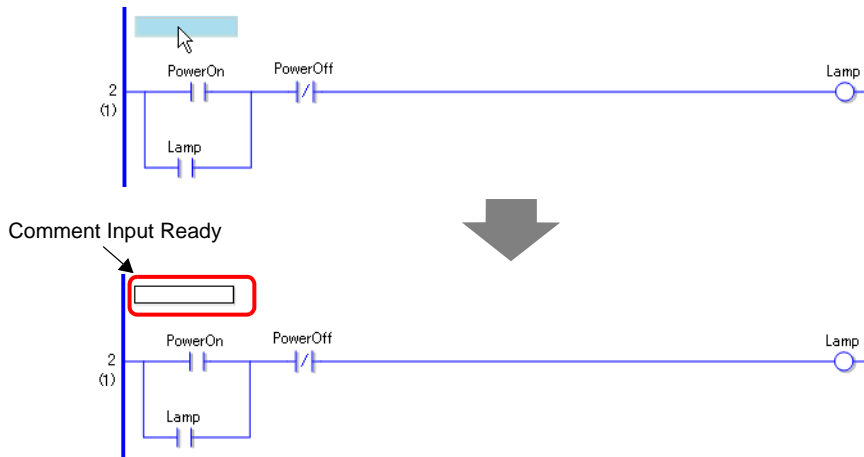


NOTE • You can also add and change the titles in [Properties Window].
 “29.13.5 Using Reference Features to Search Logic Programs” (page 29-118)

29.7.2 Adding Rung Comments

You can add comments to each rung in a logic program.

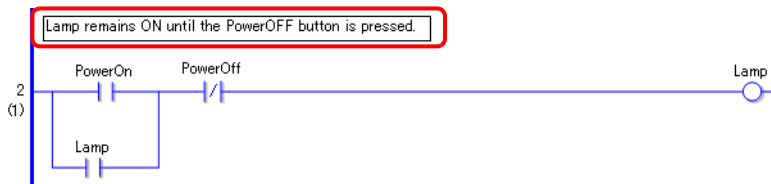
1 Double-click the part for rung comments. The comment input box will be displayed.



NOTE • You can also right-click the part for rung comments and then click [Edit] for comment input.

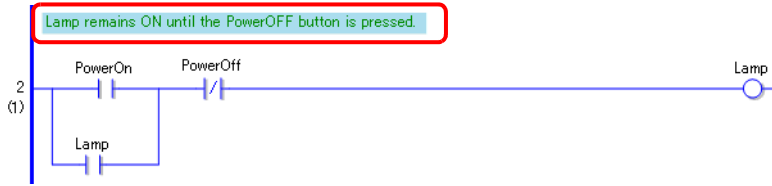
2 Input the rung comment in the text box.

You can use up to 128 characters.



NOTE • Press SHIFT+ENTER to insert a line feed.

3 Press ENTER to input the text. The rung comment has been input.

**NOTE**

- You can also add and change rung comments in [Properties Window].
 - ☞ “29.13.5 Using Reference Features to Search Logic Programs” (page 29-118)
- You can display a list of rung comments and edit the comments in the [Comment List] window.
 - ☞ “29.7.4 [Comment List] Window” (page 29-71)

IMPORTANT

- The number of rung comments depends on the comment memory specified in [Memory Usage] in the [Project Information] dialog box. You cannot create rung comments exceeding the comment memory.
 - ☞ “29.13.2 Checking the Size for Creating Programs” (page 29-103)
- You cannot edit the rung comments during online monitoring.
- You can add rung comments to normal labels but not to start or end labels.

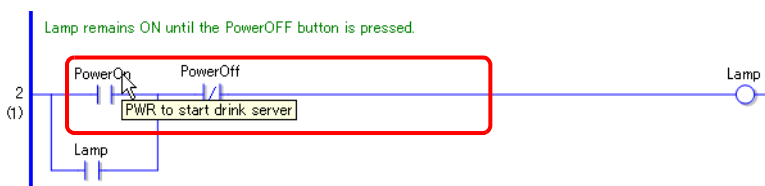
29.7.3 Symbol Variable Comments

In the [Edit Symbol Variable] window, enter the symbol variable comments in [Comment]. You can use up to 32 characters.

	Name	Type	Array	Count	Address	Keep	Comment
1	Lamp	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
2	LargeCupButton	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
3	LargeCupQuantity	Counter Variable				<input type="checkbox"/>	
4	MediumCupButton	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
5	MediumCupQuantity	Counter Variable				<input type="checkbox"/>	
6	PowerOff	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
7	PowerOn	Integer Variable	<input type="checkbox"/>			<input type="checkbox"/>	Drink server starts up.
8	SetIceSupplyCup	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
9	SmallCupButton	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
10	SmallCupQuantity	Counter Variable				<input type="checkbox"/>	
11	SodaInjectionTime	Timer Variable				<input checked="" type="checkbox"/>	
12	SodaValve	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
13	SupplyIce	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	
14	氷供給ボタン	Bit Variable	<input type="checkbox"/>			<input type="checkbox"/>	

NOTE

- For how to input the symbol variable comments, refer to the following.
 - ☞ “■ Registering Symbol Variables” (page 29-31)
- You can also add and change the symbol variable comments in [Properties Window].
 - ☞ “29.13.5 Using Reference Features to Search Logic Programs” (page 29-118)
- You can display a list of symbol variable comments and edit the comments in the [Comment List] window.
 - ☞ “29.7.4 [Comment List] Window” (page 29-71)
- Usually, the symbol variable comments are not displayed on the logic screen. Point to the symbol variable allocated to the instruction. The comment will be displayed as a tool tip.



IMPORTANT

- The number of symbol variable comments depends on the comment memory specified in [Memory Usage] in the [Project Information] dialog box. You cannot create symbol variable comments exceeding the comment memory.
 - ☞ “29.13.2 Checking the Size for Creating Programs” (page 29-103)
- You cannot edit the symbol variable comment during online monitoring.

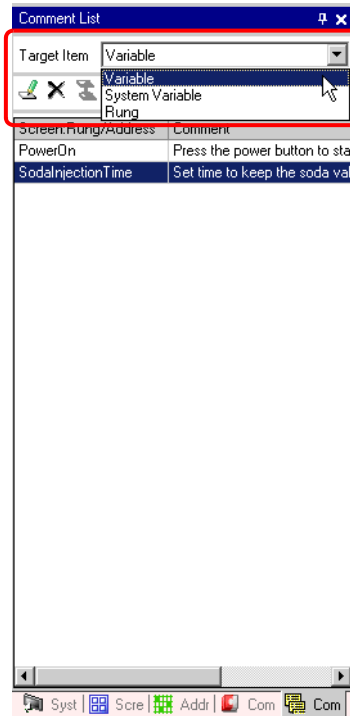
29.7.4 [Comment List] Window


The [Comment List] window displays a list of comments on the variables, system variables, and rungs in the logic program.

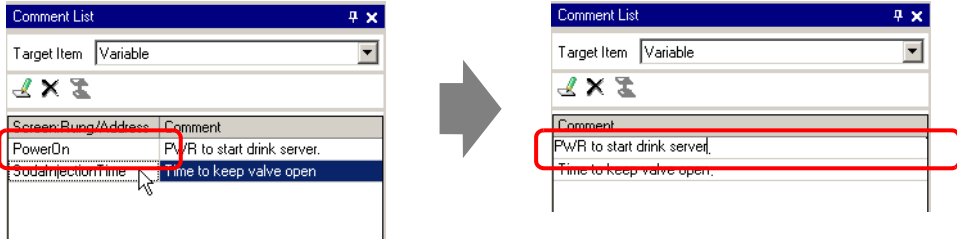
-
- NOTE** • Please refer to the settings guide for details.
 ☞ “29.14 Settings Guide” (page 29-126)
-

■ Using the [Comment List] Window

- 1 On the [View (V)] menu, point to [Work Space (W)] and then click [Comment List (C)] to display the [Comment List] window.
- 2 In [Target Item], select the type of comment type you want to display in the list.

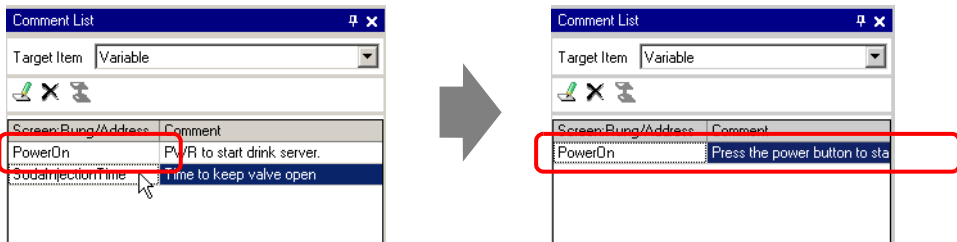


3 To edit a comment, select the comment and click .



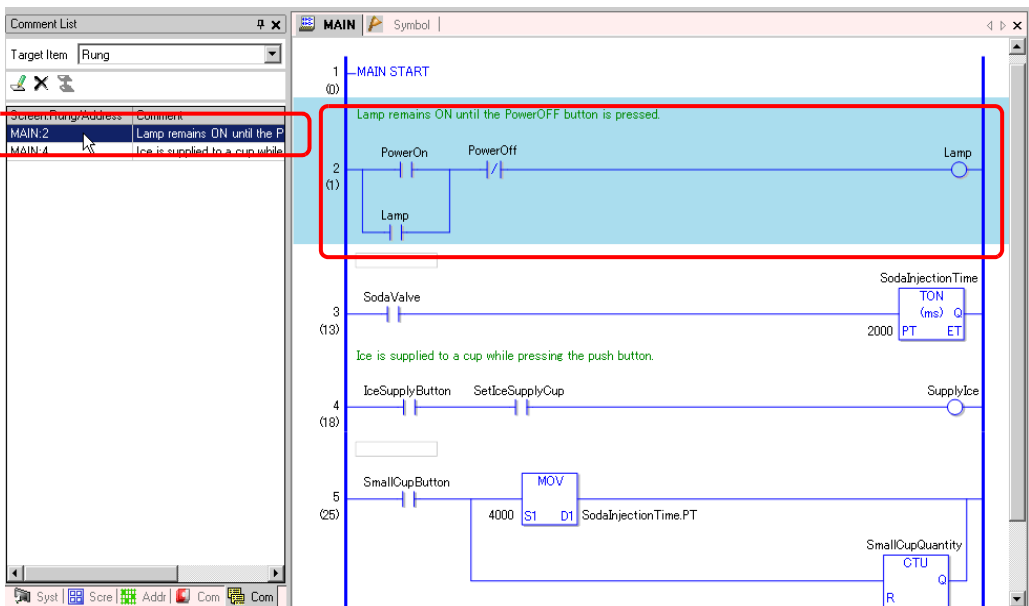
- NOTE**
- Alternatively, you can edit the comment in the following ways.
 - Double-click the comment.
 - Right-click the comment and click [Edit].

4 To delete the comment, select the comment and click .

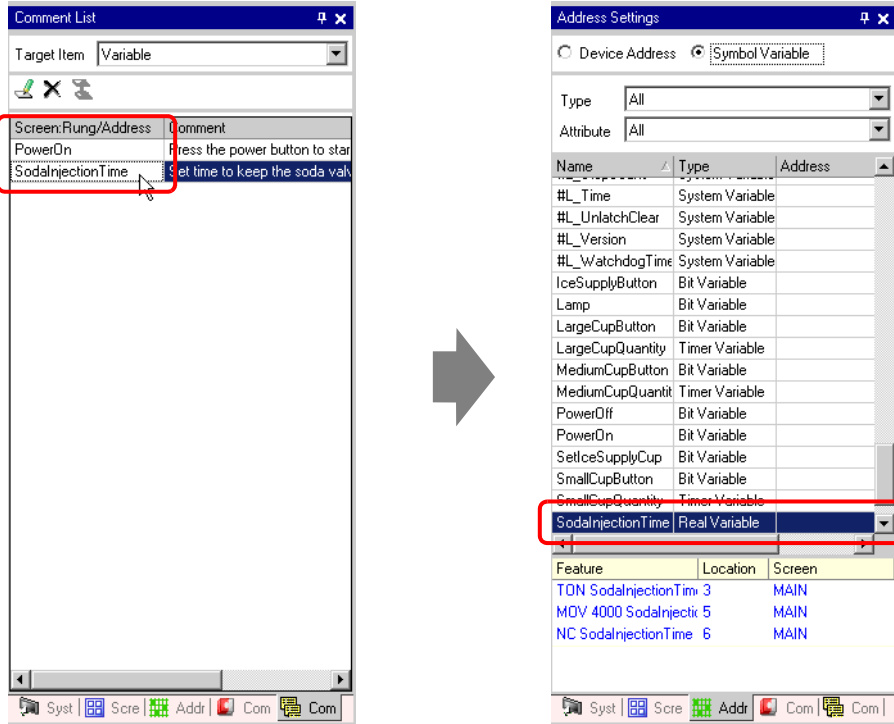


- NOTE**
- Alternatively, you can delete the comment in the following ways.
 - Double-click the comment.
 - Right-click the comment and click [Delete].
 - Select the comment and press DELETE.

5 When [Target Item] is [Rung], double-click a cell in the [Screen:Rung/Address] column to select the rung on the logic screen with the comment you want to delete.



6 When [Target Item] is [Variable] or [System Variable], double-click the cell in the [Screen:Rung/Address] column. This displays the [Address Settings] windows and selects the relevant symbol variable or system variable.



29.8 Logic Operations with Power ON

Determine whether to run or stop the logic programs when the GP is on.

NOTE

- Please refer to the settings guide for details.
 ☞ “29.14 Settings Guide” (page 29-126)
- Alternatively, you can configure the settings on the Logic Settings screen in offline mode.

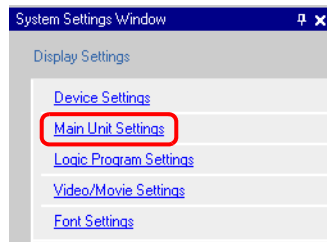
1 Click the [System Settings Window] tab to open the [System Settings Window].



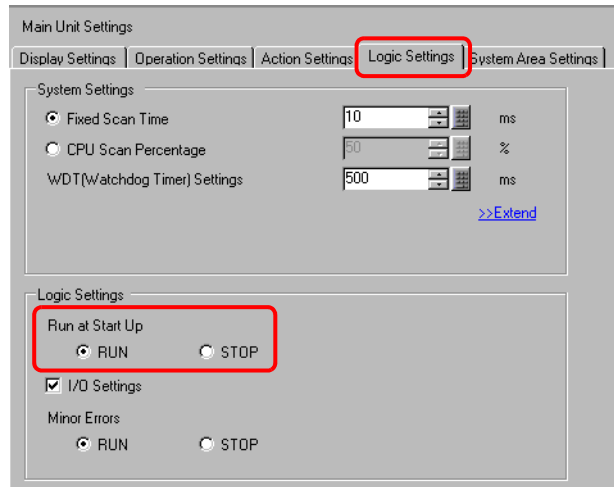
NOTE

- If the [System Settings Window] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings Window (S)].

2 Select [Main Unit Settings] from [Display Settings].



3 Click the [Logic Settings] tab and in [Run at Start Up] under [Logic Settings], select [RUN] or [STOP].



29.9 Transferring Logic Programs

Logic programs are transferred in project file format. You cannot transfer logic programs alone.


NOTE

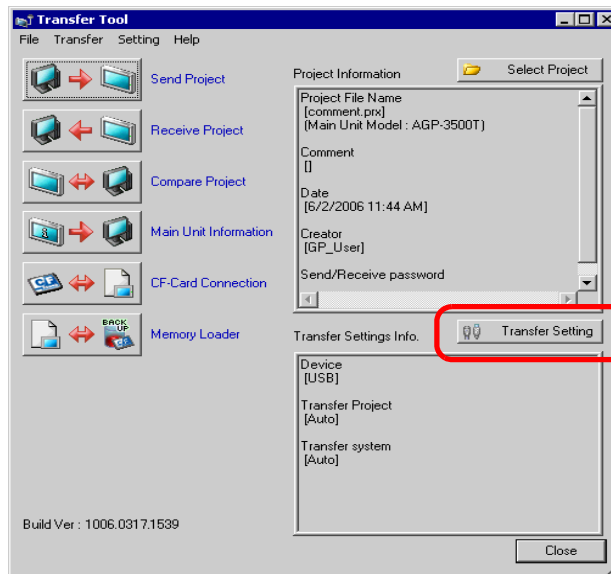
- For details on how to transfer, refer to the following.
 - ☞ “Chapter 32 Transferring Data” (page 32-1)
- The logic program is checked for errors when transferring and saving. If any error is found in the check, transfer to the GP fails.
 - ☞ “29.11 Correcting Logic Program Errors” (page 29-83)
 - ☞ “32.9 Checking Errors” (page 32-52)

■ Keep Transfer

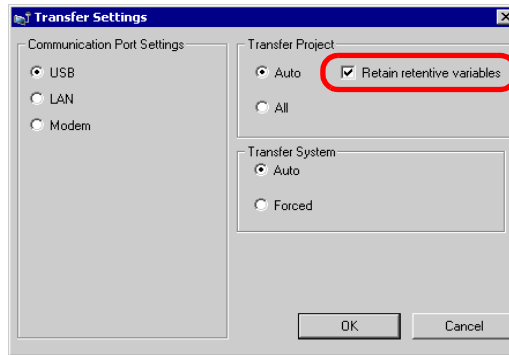
You can transfer projects when [Auto] is selected for project transfer. You can do so only when the project data on the GP and the data selected with the transfer tool match, and when in a state where the GP project data keeps the current values for the symbol variables. Even if the project data on the GP and the data selected with the transfer tool match, the keep transfer is not available if the system configuration has been changed or the project in the GP is corrupt. The keep transfer is also disabled when [Retain retentive variables] is not checked or [Forced] is checked.

When [Retain retentive variables] is not checked, the values of the variables are cleared to 0 even when [Keep] is selected in [Symbol Variable Settings] of the Common Settings. When [Retain retentive variables] is checked, the values of the variables for which [Keep] is selected is not cleared but its current value is retained.

- 1 On the state toolbar, click the transfer project icon  to start the transfer tool and click [Transfer Setting].



- 2 The [Transfer Settings] dialog box will be displayed. In [Transfer Project], select the [Retain retentive variables] check box and click [OK].




29.10 Monitoring Logic Programs (Online Monitor)

Using the online monitor, in the [Watch List] window, you can confirm the symbol variable ON/OFF or the device values within the logic program. In the [PID Monitor] window, you can also make adjustments while monitoring the setting values of the PID instruction.

NOTE • In [Option Settings], you can configure the communication settings and monitor the settings with online monitoring.

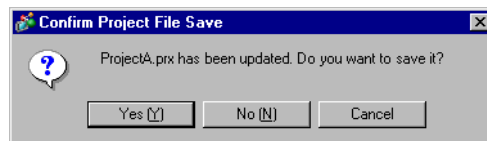
☞ “5.13.7 [Option Settings] Settings Guide ■ Monitor Step” (page 5-134)

29.10.1 Online Monitoring Procedures

1 On the state toolbar, click the monitor icon .

2 The [Confirm Project File Save] dialog box will be displayed.

- When [Yes] is selected, the logic program you are editing is saved. Upon saving, the logic program is checked for errors. If an error is found, the online monitoring will not turn on and an error message will be displayed. Click [OK], correct the error, and go back to Procedure 1.
- When [No] is selected, the logic program you are editing is discarded and the online monitoring turns on.



3 If a password has been set, enter the password.

NOTE ☞ “29.13.4 Enhancing Security” (page 29-116)

4 The GP responds with the logic program and the online monitoring starts.

IMPORTANT • You can use online monitoring only when the GP is in communication with the PC. You cannot use online monitoring when using more than one USB ports, or when communicating through both a USB port and a LAN port.

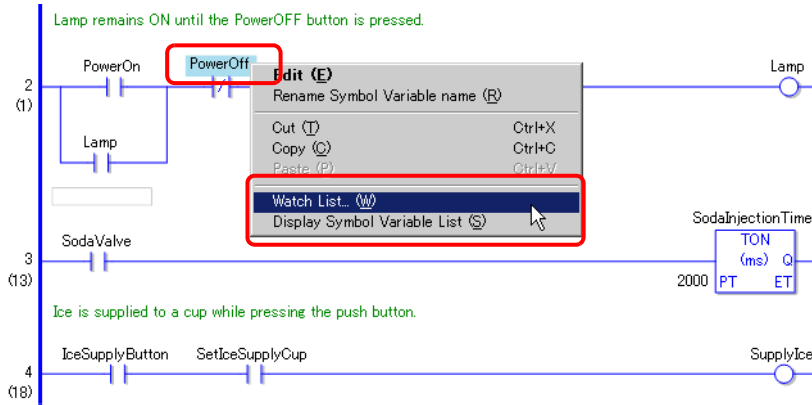
- The LAN connection does not allow you to use the online monitoring on more than one PC at the same time. The online monitoring is available to the first PC connected.

29.10.2 Monitor the Current Values of Symbol Variables

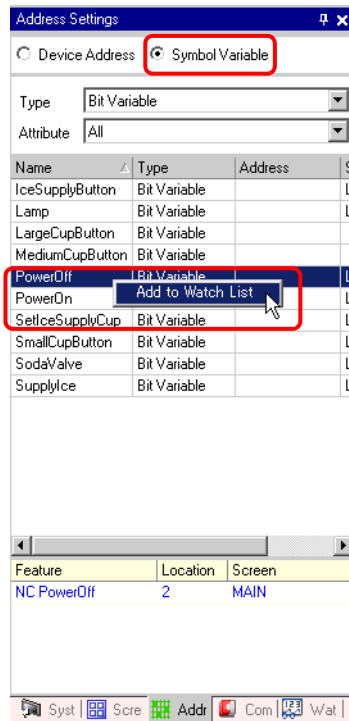
You can monitor the symbol variable ON/OFF and the device values within the logic program registered in the [Watch List] window.

- NOTE** • Please refer to the settings guide for details.
 ☞ “29.14 Settings Guide” (page 29-126)

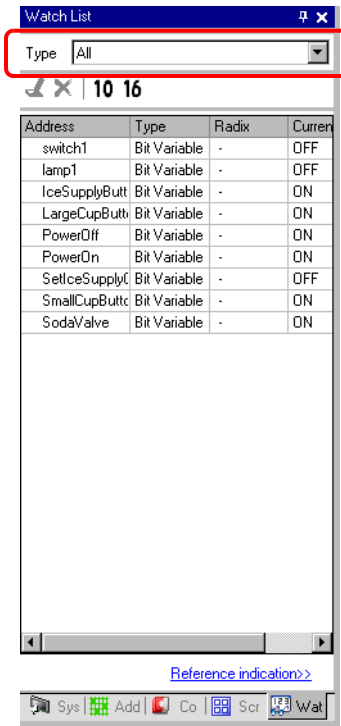
1 Right-click the symbol variable that you want to monitor within the logic program and then click [Watch List...(W)].



- NOTE** • You can also select [Symbol Variable] in the [Address Settings] window, right-click the symbol variable that you want to monitor, and then click [Add to Watch List] to add the variable.



- 2 The symbol variable is added to the [Watch List] window.
- 3 On the [View (V)] menu, point to [Work Space (W)] and then click [Watch List (W)] to display the [Watch List] window.
- 4 In [Type], select the symbol variable that you want to monitor. When the online monitoring turns on, the current values of the selected symbol variables are monitored.

**NOTE**

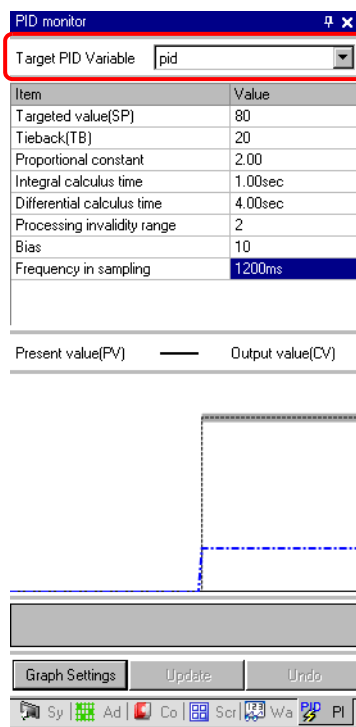
☞ "29.10.1 Online Monitoring Procedures" (page 29-78)

29.10.3 Adjusting the Setting Values for the PID Instruction

You can adjust the setting values for the PID instructions while monitoring the values in the [PID Monitor] window.

- NOTE** • Please refer to the settings guide for details.
 ☞ “29.14 Settings Guide” (page 29-126)

- 1 On the [View (V)] menu, point to [Work Space (W)] and then click [PID Monitor Window (M)] to display the [PID monitor] window.
- 2 In [Target PID Variable], select the PID variable that you want to monitor. When the online monitoring turns on, the PID instruction setting is displayed in a graph.




- NOTE** ☞ “29.10.1 Online Monitoring Procedures” (page 29-78)

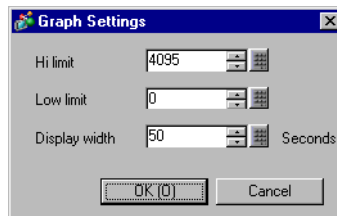
3 You can change the setting values while looking at the graph.

When the setting value is changed, the data is written to the following devices.

Items on the Screen	Save in	Remarks
Target PID Variable	None	Display the PID variable being selected.
Targeted Value (SP)	Operand S1	Can be changed only when the instruction operand is a variable.
Tieback (TB)	Operand S3	Can be changed only when the instruction operand is a variable.
Proportional Constant	Variable Format ****.KP Address Format U_****.KP	Set the (×1000) value.
Integral Calculus Time	Variable Format ****.IT Address Format U_****.IT	Set the (×1000) value.
Differential Calculus Time	Variable Format ****.DT Address Format U_****.DT	Set the (×1000) value.
Processing Invalidity Range	Variable Format ****.PA Address Format U_****.PA	
Bias	Variable Format ****.BA Address Format U_****.BA	
Frequency in Sampling	Variable Format ****.ST Address Format U_****.ST	

NOTE

- Click [Update] to update the graph.
- Click [Graph Settings] to change the graph display settings.
 "29.14 Settings Guide" (page 29-126)



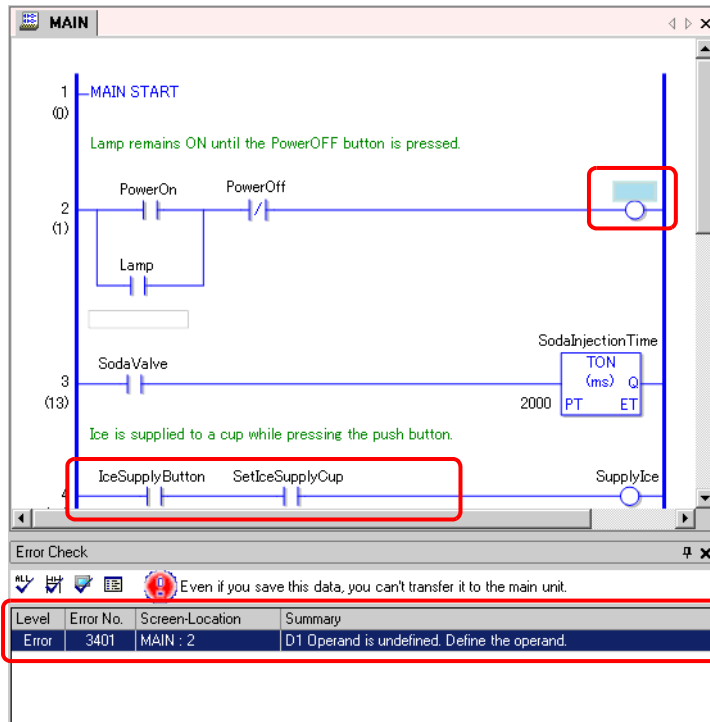
29.11 Correcting Logic Program Errors

The logic program is checked for errors when transferring and saving.

- NOTE**
- If an error is found in the check, transfer to the GP fails.
 ☞ “32.9 Checking Errors” (page 32-52)

If errors are found in the error check, a list of errors is displayed. Refer to [Level], [Error No.], [Screen-Location], [Summary] and create a correct logic program.

By displaying the logic screen and selecting the error rung, the error in the logic program will be selected. This will help you correct the error.



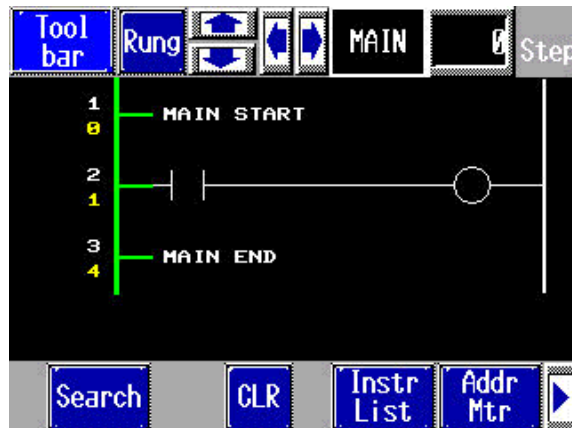
- NOTE**
- In [Option Settings], you can select the settings to display duplicate coil warnings during error checking.
 ☞ “5.13.7 [Option Settings] Settings Guide ■ Error Check” (page 5-135)

29.12 Logic Monitor

GP-Pro EX has logic monitoring functions on the GP main unit to improve the maintainability of the logic programs.

The logic monitoring function displays the logic program on the GP screen, and you can check the program operation with the GP unit itself even when the PC is not available.

The logic program runs without interruption even during the logic monitoring.



29.12.1 Starting and Ending the Logic Monitor

■ Trigger Method

There are 3 ways to start the logic monitor.

- **Start up with Parts**
Using the switch part, turn on bit 1 of #L System Variable (#L_LogicMonitor), and the logic monitor starts.
When bit 1 is turned off, the screen displayed before the logic monitor starts is displayed.
To use address, specify #L_LogicMonitor.X[1]. #L_LogicMonitor.X[0] is intended for the address monitor. Like the logic monitor, the address monitor screen appears when the specified bit of the system variable is turned on.
- **Start up with the Logic Program**
Using an instruction, turn on bits 0 (Address Monitor) and 1 (Logic Monitor) of #LSystem Variable (#L_LogicMonitor).
- **Start up with the System Menu**
On the system menu, touch [Logic Monitor] and [Address Monitor].

- | | |
|-------------|--|
| NOTE | <ul style="list-style-type: none"> • You cannot start up 2 monitors at the same time. When bits 0 and 1 are turned on at the same time, the address monitor turns on and then bit 1 turns off. • Once the monitor has started up, it is easy to switch monitors. The 0-bit and 1-bit of the system variable (#L_LogicMonitor) will not turn on/off when switching monitors. • You can start up the logic monitor when the logic functions are not being used. The address monitor is started up when you start up the logic monitor. • You cannot start up the logic monitor on AGP3302B and AGP3301. The address monitor is started up when you start up the logic monitor. |
|-------------|--|

■ End Method

There are 4 ways to end the logic monitor as follows.

- **End with Parts**
Using D-Script, turn off bits 0 and 1 of the #L system variable (#L_LogicMonitor).
(Use parts for converting data Instruction addresses such as D-Scripts.)
Since the logic monitor does not allow the user for editing, you cannot use the switch part.
- **Logic**
Use the instruction to turn off bits 0 and 1 of the system variable #L (#L_LogicMonitor) .
- **Screen Change**
When the screens change, the started-up monitor ends.
- **Monitor Screen**
Touch [End] on the screens of the logic monitor and address monitor.

NOTE

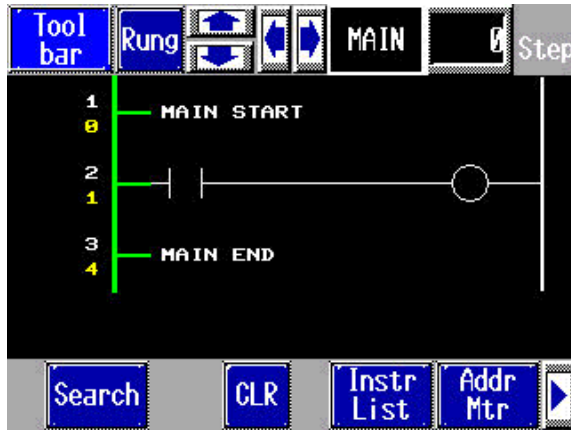
- If not changing the screens, click [Back to Previous] to end.
 - In case there is no screen to go back, such as when you start the logic monitor without the initial screen, you cannot end the logic monitor.
 - When the logic monitor and address monitor end, #L_LogicMonitor is Zero-cleared.
-

29.12.2 Logic Monitor Functions











The following explains the logic monitor features.

■ Logic Monitor









Monitors the entire logic. The logic monitor allows you to check the operational status and instruction layouts.



The logic monitor has the following features.

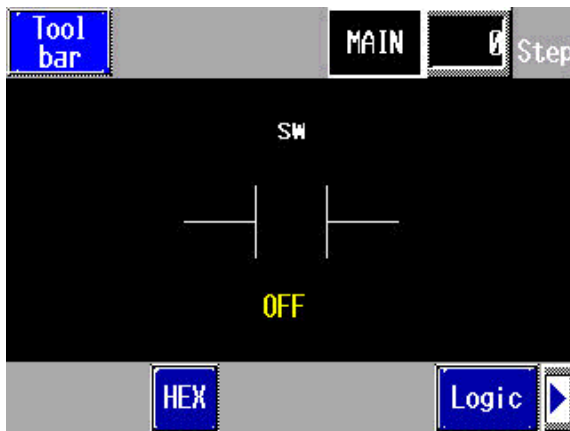
Feature	Details
Scroll 	Scroll the logic using [Rung] or [Column]. Rung: Scroll the logic using rungs. Column: Scroll the instructions one by one without the logic. For landscape, you can use only the [Column] scroll.
Enlarge Monitor	Touch the displayed instruction to enlarge the monitor.  "■ Enlarge Monitor" (page 29-88)
Logic Name Display 	Display the logic names being monitored. The names to be displayed are [INIT], [MAIN], [ERRH], and [SUB-01]-[SUB-32].
Step 	Display the top step number being monitored. When any change is made, the operation jumps to the rung with the specified step number.
Tool Bar 	Switch the tool bar display/hide at the bottom of the screen. Page 1   Page 2  Click  to switch Page 1 with Page 2.
End 	End the monitor.

Continued











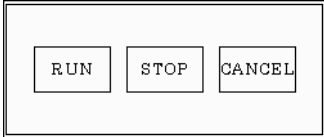



Feature		Details
RUN/STOP		<p>Switch the logic RUN/STOP. Click to display the below screen. Touch [RUN] to run and [STOP] to stop the logic.</p> 
Address Monitor		<p>Switch to the address monitor.  “ ■ Address Monitor” (page 29-89)</p>
Instruction List		<p>Switch to the instruction list.  “ ■ Instruction List” (page 29-91)</p>
Search		<p>Search the variables and instructions specified in the instruction list.  “ ■ Search” (page 29-92)</p>

■ Enlarge Monitor

Enlarge and monitor an instruction. The enlarged monitor allows you to check the operational status and the instruction operand.



The enlarged monitor has the following features.

Feature	Details
Tool Bar 	Switch the tool bar display/hide at the bottom of the screen. Page 1    Page 2   Click   to switch Page 1 with Page 2.
End 	End the monitor.
RUN/STOP 	Switch the logic RUN/STOP. Click to display the below screen. Touch [RUN] to run and [STOP] to stop the logic. 
Logic 	Switch to the logic monitor.  “ ■ Logic Monitor” (page 29-86)
DEC/HEX 	Switch the display of the current operand value between decimal/hexadecimal.

■ Address Monitor

Monitor the address used in the logic. You can check the variable name and the current value. In the address format, the logic address is monitored.

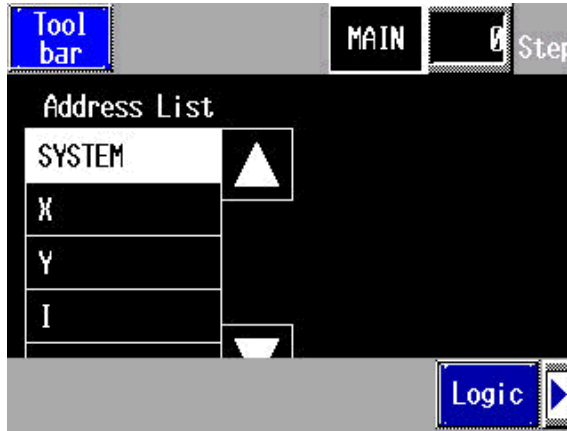


The address monitor has the following features.









Feature		Details
Scroll		Scroll the list. Up/Down: Scroll per page. Right/Left: Jump to the top/last page.
Tool Bar		Switch the tool bar display/hide at the bottom of the screen. Page 1
		↕
		Page 2
		Click to switch Page 1 with Page 2.
End		End the monitor.
RUN/STOP		Switch the logic RUN/STOP. Click to display the below screen. Touch [RUN] to run and [STOP] to stop the logic.
DEC/HEX		Switch the display of the current operand value between decimal/hexadecimal.
Address Search		Switch to the address search. ☞ “ ■ Address Search” (page 29-90)

■ Address Search

Select the address type to display in the address monitor. You can use it only in the address format.

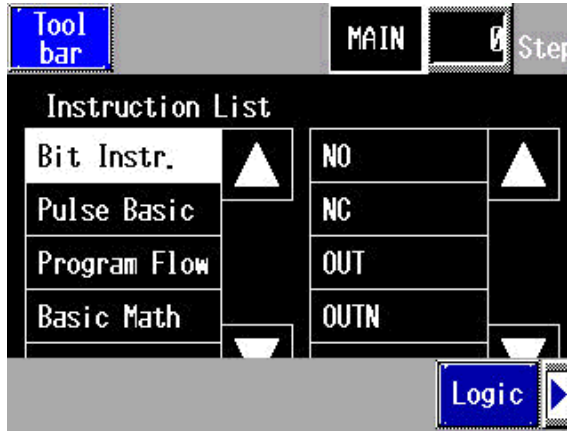


The address search has the following features.












Feature	Details
Tool Bar 	Switch the tool bar display/hide at the bottom of the screen. Page 1   Page 2  Click   to switch Page 1 with Page 2.
End 	End the monitor.
Logic 	Switch to the logic monitor. ☞ “ ■ Logic Monitor” (page 29-86)

■ Instruction List

Provides a list of instructions. Select the category to display all the lists and then select the relevant list.

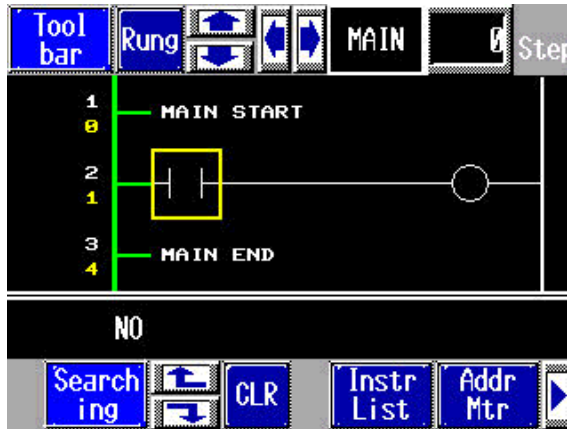


The instruction list has the following features.

Feature	Details
Tool Bar 	Switch the tool bar display/hide at the bottom of the screen. Page 1    Page 2   Click   to switch Page 1 with Page 2.
End 	End the monitor.
Logic 	Switch to the logic monitor.  “ ■ Logic Monitor” (page 29-86)

■ Search

In the address monitor and instruction list, select a variable to use as the search key. The search is conducted in the logic monitor.



The search has the following features.

Feature	Description
Variable Search	Use only variables as the search key. Select only the key variable in the address monitor.
Instruction Search	Use only instructions as the search key. Select only the key instruction in the instruction list.
Variable & Instruction Search	Use a variable and instruction as the search keys. Select the key variable in the address monitor and the key instruction in the instruction list.
Next Search	Based on the first search result, search a variable and instruction with the next closest match.
Clear Search	Clear the variable and instruction selected as the search keys.

NOTE

- You can perform a search only while search is selected. The search is terminated if you scroll the screen.
- You can use the up/down search in the next search.

■ Password

For logic with a password, you can monitor the logic after inputting the password.

Restrictions

The logic monitor has the following restrictions.

- The enlarged monitor and address monitor can display a limited number of characters.

Resolution	Pixel Size	Bit Instruction	App Instruction
QVGA	320×240	38 characters	13 characters
VGA	640×480	78 characters	33 characters
SVGA	800×600	98 characters	43 characters
XGA	1024×768	126 characters	57 characters


- This is for the landscape screen only. The portrait screen can also be displayed in landscape.

29.13 Useful Features of the Logic Editor

29.13.1 Replacing Parts in Instructions and Instructions in Parts

Drag the parts and instructions between the drawing screen and logic screen to allocate symbol variables, insert new instructions, and place new parts. This allows you to create screens and logic programs more efficiently.



Here, for example, the logic screen (e.g. MAIN) and the drawing screen (e.g. Base 1) are opened and tiled vertically in the work space.

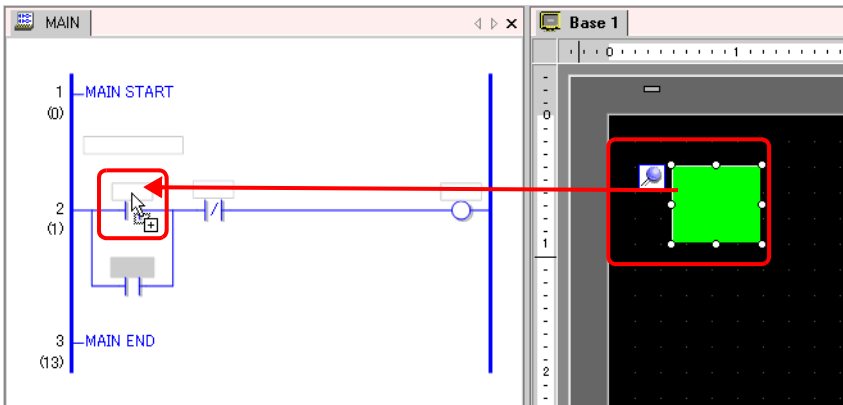
- NOTE**
- To display the screens tile-vertically in the work space, on the [View (V)] menu, point to [Screen Block (B)] and select [Tile Vertically] or click .
 - Alternatively, you can start up multiple GP-Pro EXs and drag the parts and instructions between projects from one logic screen to another, or from one drawing screen to another. You cannot drag logic from a logic screen to a drawing screen or from a drawing screen to a logic screen.

- IMPORTANT**
- If the allocated symbol variable already exists, an error will occur. Perform error checking to identify duplicated symbol variables.
 - ☞ “29.11 Correcting Logic Program Errors” (page 29-83)
 - ☞ “29.9 Transferring Logic Programs” (page 29-76)

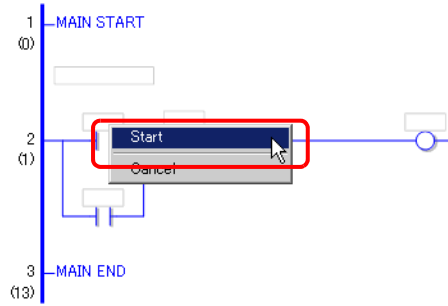
■ Allocating Symbol Variables to Instructions from Parts

You can allocate the symbol variables allocated to parts on the drawing screen to the instruction operands on the logic screen.


- 1 Click the part on the drawing screen. Drag it to the instruction operand to which you want to allocate it. Where the cursor changes from  to , release the left mouse button.



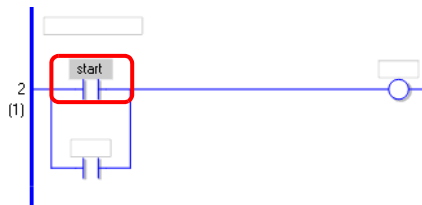
2 Select the symbol variable.



NOTE

- Symbol variables that can be allocated differ depending on parts.
☞ “◆ Drag & Drop Parts” (page 29-102)
- When more than one symbol variable is allocated to a part, the possible symbol variables are displayed.
- Click [Cancel] to cancel the symbol variable allocation.
- Where the cursor is displayed with , you cannot allocate symbol variables.

3 The symbol variable allocated to the part is allocated to the instruction operand.





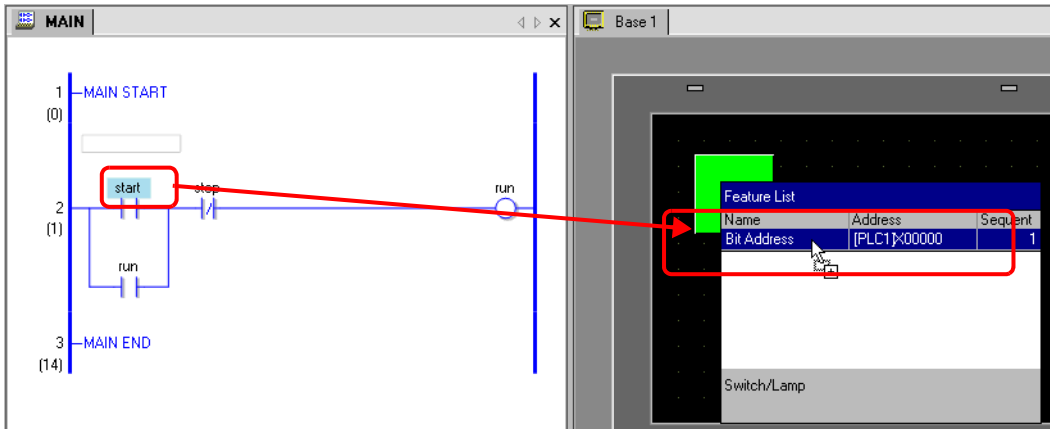
NOTE

- Alternatively, you can allocate symbol variables in the [Address Settings] window.
☞ “■ Operand Settings using Drag & Drop” (page 29-63)


■ Allocating Symbol Variables to Parts from Instructions

You can allocate symbol variables allocated to the instruction operand on the logic screen to parts on the drawing screen.

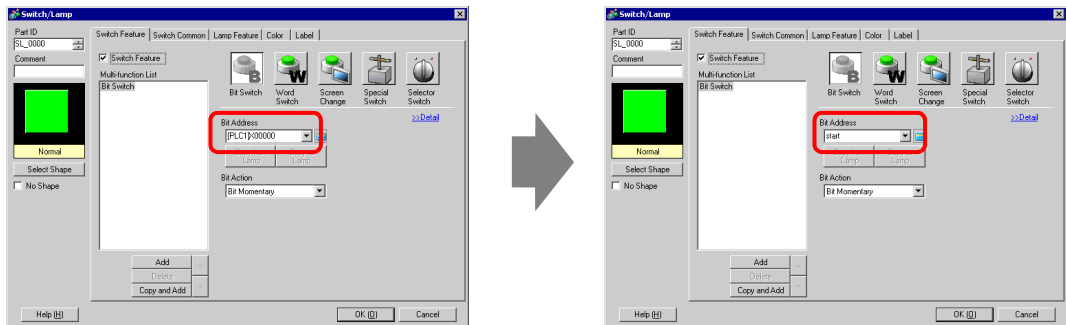
- 1 Click the instruction operand on the logic screen. Drag it to the part on the drawing screen to which you want to allocate it. When the feature list is displayed, point the cursor to the feature to which you want to allocate it. Where  changes to , release the left mouse button.



NOTE



- Symbol variables you can be allocated differ depending on instructions.
 - ☞ “ ■ Instructions and Parts that can be moved by a Drag & Drop operation” (page 29-98)
- When more than one feature is allocated to a part, the possible features are displayed.
- If you release the left button on the mouse before the rung is reverse-displayed, the symbol variable allocation is canceled.
- Where the cursor is displayed with , you cannot allocate symbol variables.

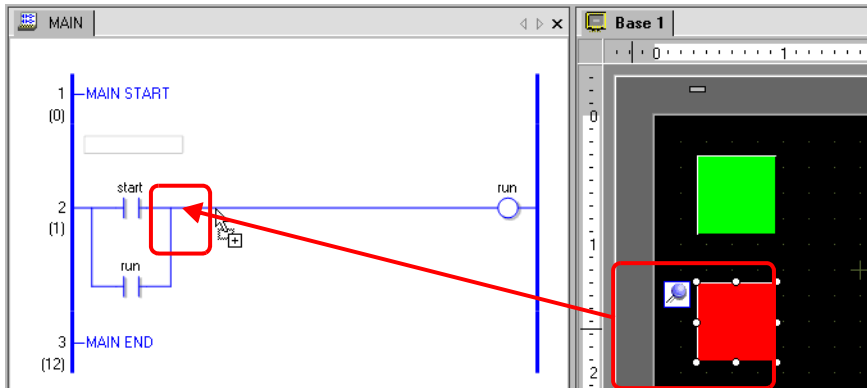
- 2 The symbol variable allocated to the instruction operand is allocated to the part.



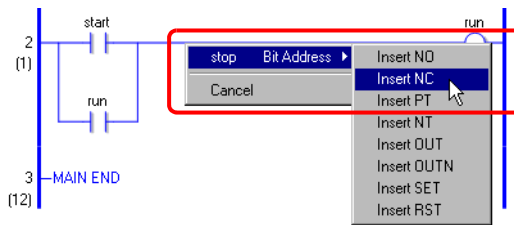
■ Inserting New Instructions from Parts

You can insert instructions by dragging the parts to the rungs or shunts of the logic program.



- 1 Click the part on the drawing screen. Drag it to where you want to insert the instruction on the logic screen. Where the cursor changes from  to , release the left mouse button.



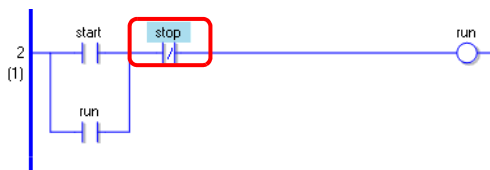
- 2 Select the symbol variable and then select the instruction that you want to insert.



NOTE



- Instructions that can be inserted differ depending on the parts.
 “◆ Drag & Drop Parts” (page 29-102)
- When more than one symbol variable is allocated to a part, the possible symbol variables are displayed.
- Click [Cancel] to cancel the symbol variable allocation.
- Where the cursor is displayed with , you cannot insert instructions.

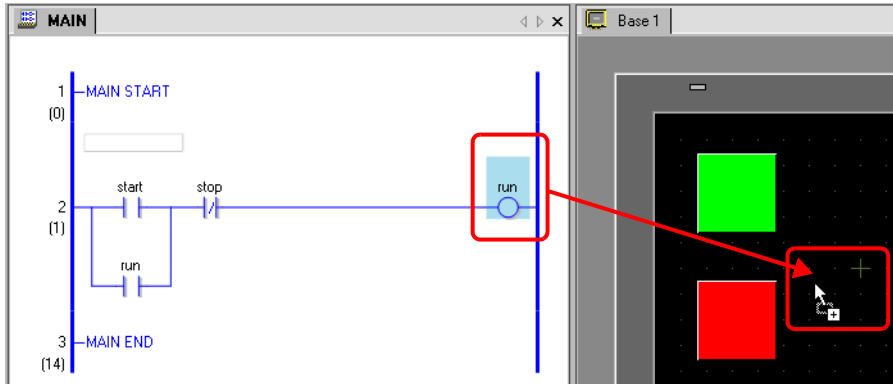
- 3 The instruction to which the symbol variable of the part is allocated is inserted.



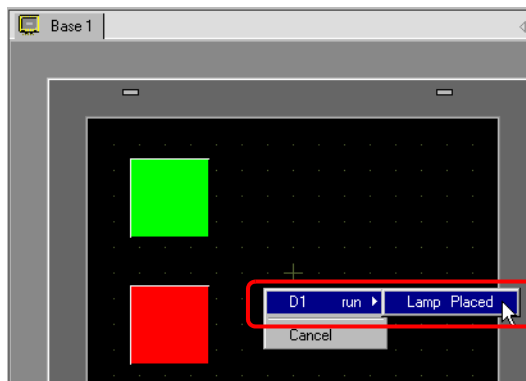
■ Placing New Parts from Instructions

You can allocate symbol variables allocated to the instruction operand on the logic screen to parts on the drawing screen.


- 1 Click the instruction on the logic screen. Drag the instruction to where you want to place it on the drawing screen. Where  changes to , release the left mouse button.



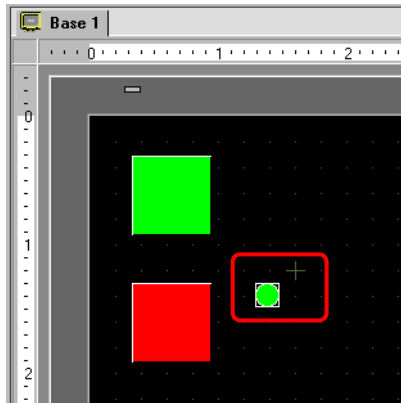
- 2 Select the operand and then select the part that you want to place.



NOTE

- Parts that can be placed differ depending on the instruction.
 - ☞ “◆ Instructions that can be moved by a Drag & Drop Operation” (page 29-98)
- When more than one part can be placed, the possible parts will be displayed.
- If you drag an instruction that cannot be placed, the action will be canceled.
- Click [Cancel] to cancel the part placement.
- Where the cursor is displayed with , you cannot place parts.

3 The part is placed with the symbol variable of the instruction. Change the size and color as necessary.



■ Instructions and Parts that can be moved by a Drag & Drop operation

◆ Instructions that can be moved by a Drag & Drop Operation

You can drag the following parts from the instructions or operands on the logic screen and drop on the drawing screen to allocate or reallocate the symbol variables.

Drag & Drop Instructions and Operands				Parts that can be allocated or parts to which a new symbol variable can be placed	
Command	Number of Operands	Symbol Variable Type			
Basic Instructions	NO, NC, PT, NT	1	S1	Bit Address Bit Variable	• Bit Switch
	OUT, OUTN, SET, RST, PTO, NTO	1	D1	Bit Address Bit Variable	• Lamp
	JMP, JSR	1	-	-	-
	RET, EXIT	-	-	-	-
	FOR	1	S1	Word Address Integer Variable	• Data Display
	NEXT	-	-	-	-
	PBC	2	S1	-	-
	PBR	1	D1	Bit Variable	• Lamp
Operation Instruction	ADD, SUB, MUL, DIV	3	S1	Word Address	• Word Switch • Data Display (You cannot select the word switch for float variables and real variables.)
				Integer Variable	
				Float Variable	
			S2	Real Variable	
				Word Address	
				Integer Variable	
		D1	Float Variable		
			Real Variable		
			Real Variable		

Continued

Drag & Drop Instructions and Operands					Parts that can be allocated or parts to which a new symbol variable can be placed
Command	Number of Operands		Symbol Variable Type		
Operation Instruction	MOD	3	S1	Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display
				Integer Variable	
			S2	Word Address	
				Integer Variable	
			D1	Word Address	
				Integer Variable	
	JADD, JSUB	3	S1	Time Variable (.HR .MIN .SEC)	[Special Variable: Integer Variable]
			S2	Time Variable (.HR .MIN .SEC)	<ul style="list-style-type: none"> • Word Switch • Data Display
			D1	Time Variable (.HR .MIN .SEC)	
	AND, OR, XOR	3	S1	Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display
				Integer Variable	
			S2	Word Address	
				Integer Variable	
			D1	Word Address	
				Integer Variable	
	NOT	2	S1	Word Address	
Integer Variable					
D1			Word Address		
			Integer Variable		
MOV	2	S1	Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display 	
			Integer Variable		
		D1	Word Address		
			Integer Variable		
BLMV	3	S1	Bit Address (Array)	<ul style="list-style-type: none"> • Word Switch • Data Display • Setting Value Display (You cannot select S1 or D1.)	
			Integer Variable (Array)		
			Float Variable (Array)		
			Real Variable (Array)		
		S2	Integer Variable		
			D1		Bit Address (Array)
					Integer Variable (Array)
					Float Variable (Array)
Real Variable (Array)					
FLMV	3	S1	Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display (You cannot select D1.)	
			Integer Variable		
			Float Variable		
			Real Variable		
		S2	Integer Variable		
			D1		Integer Variable (Array)
					Float Variable (Array)
					Real Variable (Array)
XCH	2	D1		Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display
			Integer Variable		
		D2	Word Address		
			Integer Variable		

Continued

Drag & Drop Instructions and Operands					Parts that can be allocated or parts to which a new symbol variable can be placed	
Command	Number of Operands		Symbol Variable Type			
Operation Instruction	ROL, ROR, RCL, RCR, SHL, SHR, SAL, SAR	3	S1	Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display 	
				Integer Variable		
			S2	Word Address		
				Integer Variable		
D1	Word Address					
	Integer Variable					
Compare Instruction	EQ, GT, GE, LT, LE, NE	2	S1	Word Address		<ul style="list-style-type: none"> • Word Switch • Data Display <p>(You cannot select the word switch for float variables and real variables.)</p>
				Integer Variable		
				Float Variable		
			S2	Word Address		
				Integer Variable		
				Float Variable		
	JEQ, JGT, JGE, JLT, JLE, JNE	2	S1	Time Variable (.HR .MIN .SEC)	[Special Variable: Integer Variable]	
				S2	Time Variable (.HR .MIN .SEC)	
	NEQ, NGT, NGE, NLT, NLE, NNE	2	S1	Date Variable (.YR .MO .DAY)	[Special Variable: Integer Variable]	
				S2	Date Variable (.YR .MO .DAY)	
Timer Instruction	TON, TOF, TP, TONA, TOFA	1	Special Variable	Timer Variable (.ET .PI)	[Special Variable: Integer Variable]	
				Timer Variable (.Q .TI .R)	<ul style="list-style-type: none"> • Word Switch • Data Display 	
Counter Instruction	CTU, CTD, CTUD	1	Special Variable	Counter Variable (.PV .CV)	[Special Variable: Integer Variable]	
				Counter Variable (.Q .QD .QU .UP .R)	<ul style="list-style-type: none"> • Word Switch • Data Display 	
Convert Instruction	BCD, BINENCO, DECO	2	S1	Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display 	
				Integer Variable		
			D1	Word Address		
				Integer Variable		
	RAD, DEG, SCL	2	S1	Word Address		<ul style="list-style-type: none"> • Word Switch • Data Display <p>(You cannot select the word switch for float variables and real variables.)</p>
				Integer Variable		
				Float Variable		
			D1	Real Variable		
				Word Address		
				Integer Variable		
Float Variable						
Real Variable						

Continued

Drag & Drop Instructions and Operands					Parts that can be allocated or parts to which a new symbol variable can be placed
Command		Number of Operands		Symbol Variable Type	
Conversion Instruction	I2F, I2R, F2I, F2R, R2I, R2F, H2S, S2H	2	S1	Type -	
			D1	Type	
Function Instruction	SUM, AVE	3	S1	Integer Variable (Array)	<ul style="list-style-type: none"> • Word Switch • Data Display (You cannot select S1.)
				Float Variable (Array)	
				Real Variable (Array)	
			D1	Integer Variable	
				Float Variable	
				Real Variable	
	SQRT	2	S1	Float Variable	<ul style="list-style-type: none"> • Data Display
				Real Variable	
			D1	Float Variable	
				Real Variable	
	BCNT	2	S1	Integer Variable (Array)	
				Float Variable (Array)	
				Real Variable (Array)	
			D1	Integer Variable (Array)	
Float Variable (Array)					
Real Variable (Array)					
PID	5	Special Variable	PID Variable (.KP .TR .TD .PA .BA .ST)	[Special Variable: Integer Variable] <ul style="list-style-type: none"> • Word Switch • Data Display 	
			PID Variable (.Q .UO .TO .PF .IF)	[Special Variable: Bit Variable] <ul style="list-style-type: none"> • Bit Switch • Lamp 	
		S1	Word Address	<ul style="list-style-type: none"> • Word Switch • Data Display 	
			Integer Variable		
		S2	Word Address		
			Integer Variable		
		S3	Word Address		
			Integer Variable		
		D1	Word Address		
			Integer Variable		
SIN, COS, TAN, ASIN, ACOS, ATAN, COT, EXP, LN, LG10	2	S1	Float Variable	<ul style="list-style-type: none"> • Data Display 	
			Real Variable		
		D1	Float Variable		
			Real Variable		
R/W Instruction	JRD, JSET	1	D1	Time Variable (.HR .MIN .SEC)	[Special Variable: Integer Variable] <ul style="list-style-type: none"> • Word Switch • Data Display
				Date Variable (.YR .MO .DAY)	[Special Variable: Integer Variable] <ul style="list-style-type: none"> • Word Switch • Data Display

◆ Drag & Drop Parts

Using the following operands or instructions, you can drag and drop parts and allocate or reinsert symbol variables from the drawing screen to the logic screen.

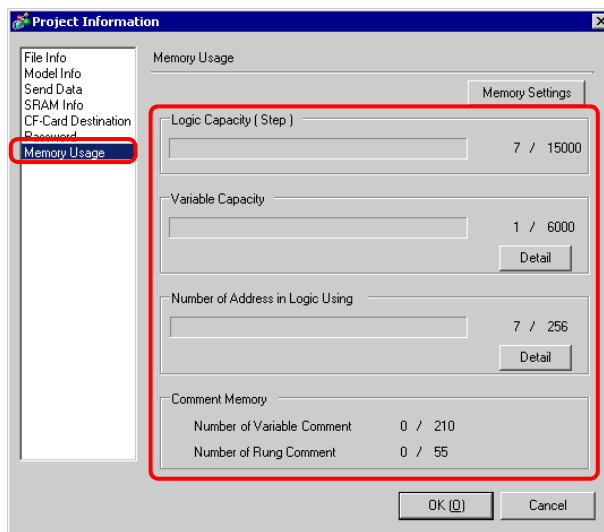
Drop & Drop Parts		Operands to which symbol variables are allocated or instructions to which symbol variables are inserted.		
Part	Symbol Variable Type	Command	Operand	
Switch/ Lamp	Bit Switch	Bit Address	NO, NC, PT, NT, OUT, OUTN,	
		Bit Variable	SET, RST, PTO, NTO	
	Word Switch	Word Address	MOV, ADD, SUB, MUL, DIV, EQ, GT, GE, LT, LE, NE	They are allocated to S1 of the instruction.
		Integer Variable		
	Screen Change	-		
	Special Switch	-		
	Selector Switch	-		
Lamp	Bit Address	NO, NC, PT, NT, OUT, OUTN,		
	Bit Variable	SET, RST, PTO, NTO		
Data Display	Data Display (Input Permit)	Word Address	MOV, ADD, SUB, MUL, DIV, EQ, GT, GE, LT, LE, NE	They are allocated to S1 of the instruction.
		Integer Variable		
		Float Variable	SIN, COS, TAN, ASIN, ACOS, ATAN, COT, EXP, LN, LG10	
	Text Display	Word Address	-	
		Integer Variable	-	
	Date/Time Display	-		
	Statistical Data Display	-		
Show Limit Value	-			

29.13.2 Checking the Size for Creating Programs

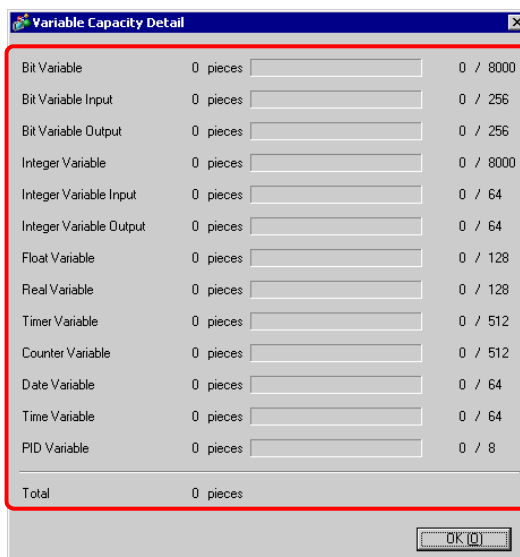
By checking the current logic capacity, symbol variable capacity, address points, comments memory of logic programs, you can prevent errors such as exceeded capacity. You can change the proportion of the logic capacity and comment memory depending on the use.

■ Memory Usage Check

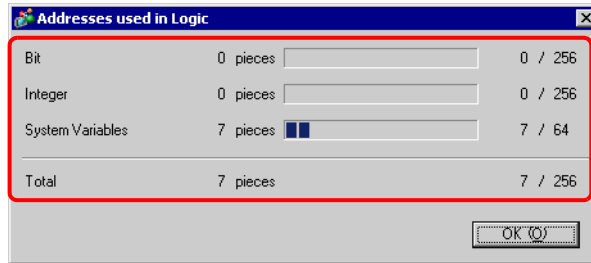
- 1 On the [Project (F)] menu, point to [Properties (I)] and click [Project Information (I)].
The [Project Information] dialog box will be displayed.
- 2 Click [Memory Usage] to check [Logic Capacity], [Variable Capacity], [Number of Address in Logic Using] and [Comment Memory].



- 3 Click [Detail] under [Variable Capacity] to check the current number, the configurable number of each symbol variable and the current total number.



- 4 Click [Detail] under [Number of Address in Logic Using] to check the current number, the configurable number of [Bit], [Integer] and [System Variables] and the total number.



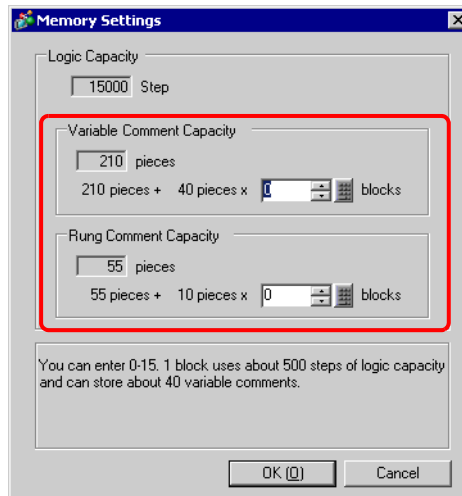
■ Memory Settings

You can specify the upper capacity limit for the symbol variable comments and rung comments.

IMPORTANT

- The size of logic program that you can create is determined based on the comment memory that you specified. When creating a logic program with many steps involved, reduce the comment blocks. When creating a logic program with many comments, increase the comment blocks. You cannot create comments exceeding the comment memory settings or steps exceeding the number determined by the comment memory.

- In the [Project Information] dialog box, click [Memory Settings]. The [Memory Settings] dialog box is displayed.
- Specify the upper comment memory limit for the symbol variable and rung comment capacity within the range of 0-15.



■ Number Restrictions on Symbol Variables

When using device addresses in logic programs, the following number restrictions will apply.

Name	Memory Size	Maximum Number for Registration	Registration Number Restrictions on the GP-Pro EX
Bit Address (Bit Variable)	64 byte	512	256
Word Address (Integer) Variable	1024 byte	256	256
System Variable	256 byte	64	64
Total Number Available	1000	256	

NOTE • Check the number restriction for the device addresses on the GP-Pro EX. If the number exceeds the restriction, an error will occur.

■ Number Restriction on Logic Programs and GP Memory Restriction

Item	Number Restriction on the GP-Pro EX	GP Memory Restriction
Number of Programs	INIT: 1, MAIN: 1, SUB: 32; Total: 34 15K Steps	96K Byte
Number of Program Rungs	5000 Rungs	
Number of Instructions per Rung	99 Instructions	
Number of Label Characters	Fixed Name	None
Number of Labels per Project	99 Instructions	
Number of Devices	28000 Devices	64K Byte
Number of NT/PT Instructions	Unlimited	None
Number of Constants	Unlimited	None
Number of Compulsory Changes	Unlimited	None
Array Size	4096 arrays	None
Number of Variables	9000 Symbol Variables 6000 Variables	1M Byte
Variable Name	32 Characters	

Continued

Item	Number Restriction on the GP-Pro EX	GP Memory Restriction
Number of Variable Comments	210 comments (initial value)	16K Byte→14K Byte
Number of Variable Comment Characters	(Single-byte Only) 32 Characters	
Number of Rung Comments	55 Comments (initial value)	16K Byte→14K Byte
Number of Characters in Rung Comments	128 Characters	
Number of Program Comments	34 Comments	8K Byte
Number of Characters in Program Comments	32 Characters	
Nest	25 Nests	Stack: 16 (32)

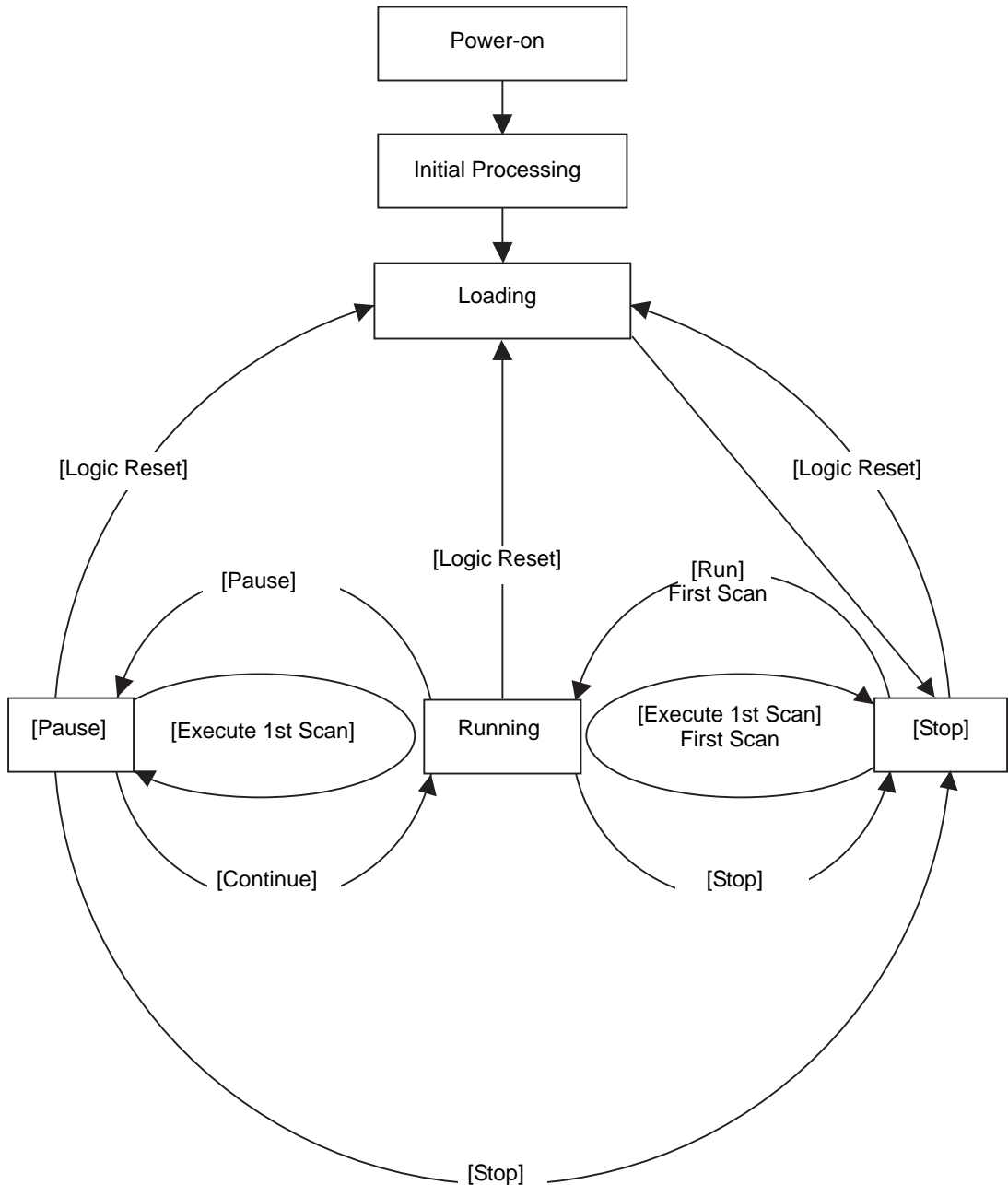
29.13.3 Adjusting Logic Scan Time

The following provides a summary of the operation of logic features and the scan time for execution of logic programs. The procedure for adjusting the scan time is also explained.

■ Logic Features

◆ Summary

The logic features operate as follows. The details are explained on the next page.



- **Initial Processing**

This is the initial state of the logic program execution engine. After the logic program execution engine has been initialized, the logic state changes to “loading.”

- **Loading**

The logic program is read from the memory. It checks whether the logic program has been loaded normally and remedies the error if not loaded normally. Once the program has been loaded normally, it will stop.

If [Run] is selected for the power-on action, the run command will be automatically executed. When changing to the “running” state, the I/O is initialized.

- **Stop**

The logic is in the paused state. Upon receiving a command ([Logic Reset], [Run], [Execute 1st Scan], [Continue], or [Pause]), the state will change accordingly.

On the [Logic Reset] command, “loading” starts and the symbol variable is initialized.

When it is a keep-type variable and the power is off or the GP logic is reset, the most recent data is maintained. However, if the logic is reset with the online monitor (mode that runs a program in the logic on GP-Pro EX) or #L_Command, the value is initialized with the value specified for the logic features of GP-Pro EX.

The [Run] command or [Execute 1st Scan] command zero-clears the clear-type variable.

The [Run] command starts “running”. The [Execute 1st Scan] command runs the logic program once.

- **First Scan**

Reads I/O, runs the initialized logic program, and then writes the I/O.

- **Running**

The logic program execution engine is running. The engine reads the I/O, runs the logic program, writes the I/O, and updates the system variables (#L_AvgLogicTime, #L_AvgScanTime, etc.).

The [Logic Reset] command starts “loading”.

The [Stop] command stops the logic.

The [Pause] command pauses the logic.

- **Pause**

The logic program execution engine is paused. To prevent the I/O watchdog time-out, I/O write/read is executed. However, since the logic program has not run, the output state remains unchanged. Upon receiving a command, the state will change accordingly.

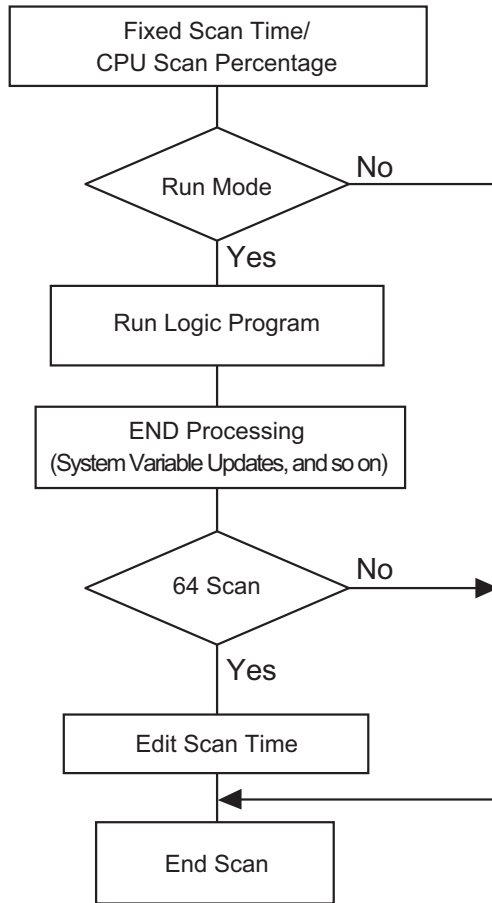
The [Logic Reset] command starts “loading”.

The [Execute 1st Scan] command runs the logic program once.

The [Stop] command stops the logic. The [Continue] command starts “running”.

◆ **Running Flow**

The scan runs as follows.



- **Scan Time Adjustment**
The scan time is adjusted every 64 scans. The scan times for the fixed scan time mode and CPU scan percentage mode are as follows.
- **Fixed Scan Time Mode**
Scan Time = (#L_AvgLogicTime × 100) ÷ 50 (Logic function support model, GP33** series: 30, Other models: 50)
- **CPU Scan Percentage Mode**
Scan Time = (#L_AvgLogicTime × 100) ÷ #L_PercentAlloc

NOTE • For the details of #L_AvgLogicTime and #L_PercentAlloc, refer to the following.
☞ “A.5 System Variables” (page A-73)

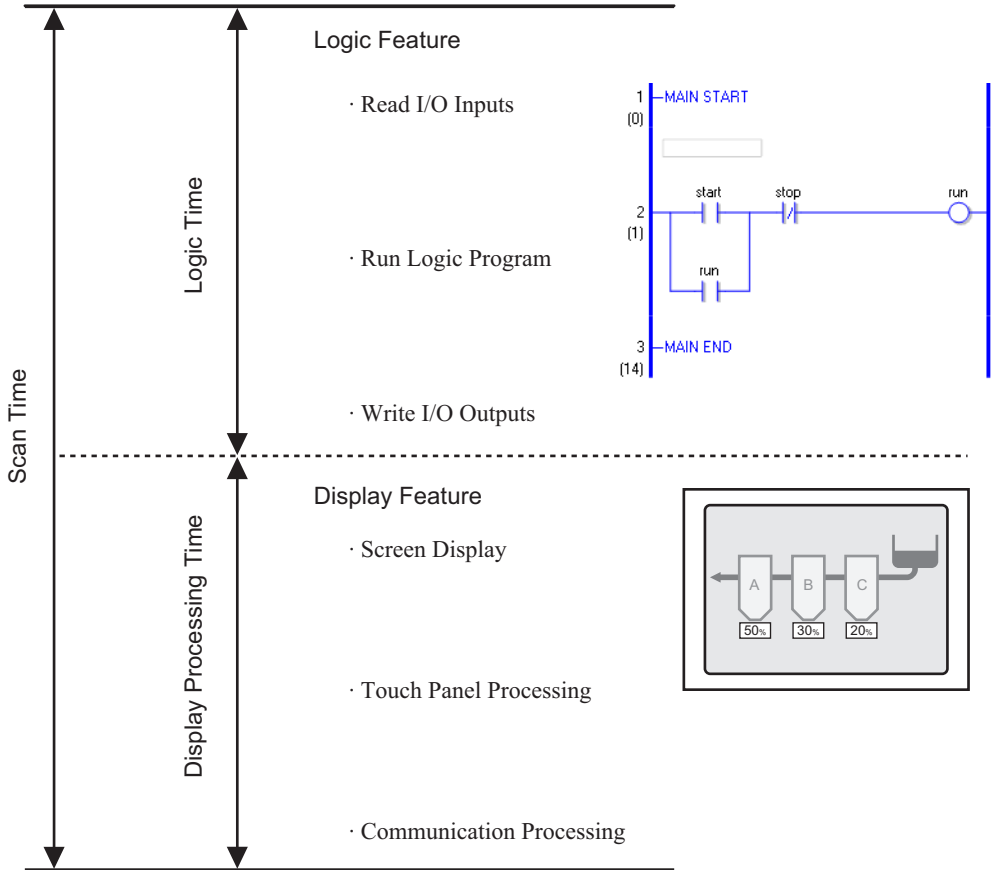
- **Scan Time Error**
The following error is found in the logic scan time.

Model	Error
AGP3000 Series	10%

*The above error is found when a PLC is used in the serial communication.

■ Logic Scan Time

The logic time includes the logic features and display and the display features (screen display, touch panel processing, communication). The logic feature runs the logic programs. Both features are as follows. The GP scan time has a fixed scan time mode and a CPU scan percentage mode.



NOTE

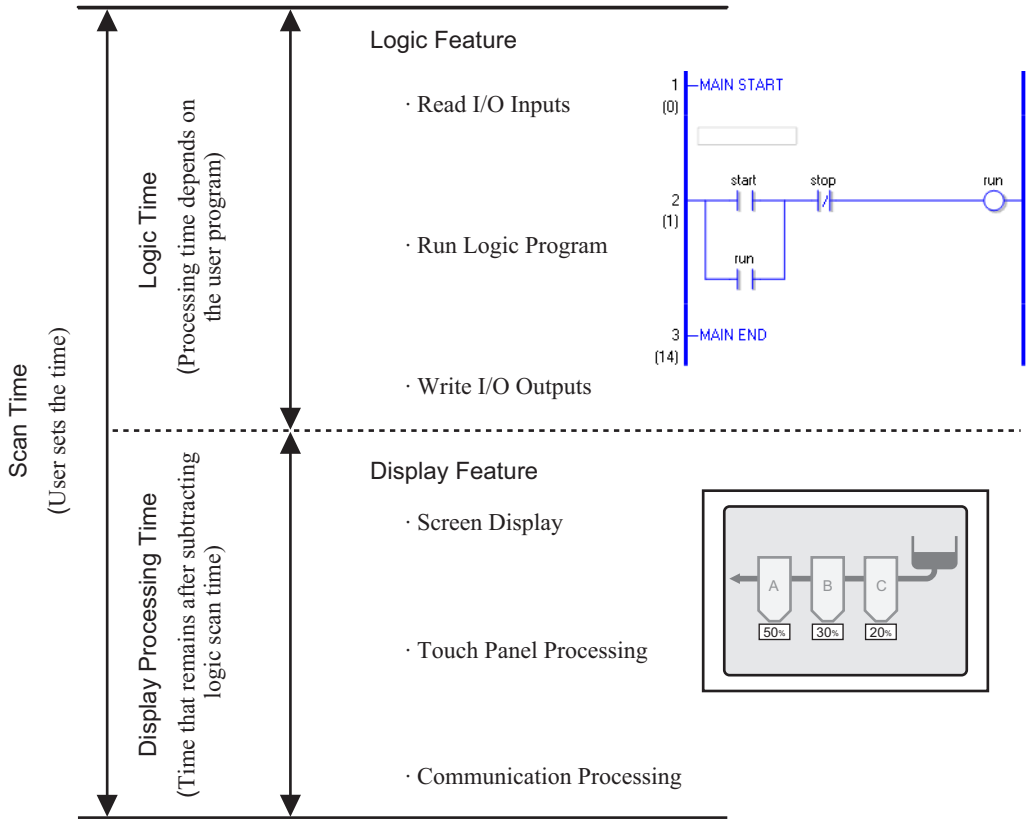
☞ “◆ Fixed Scan Time” (page 29-111)

☞ “◆ CPU Scan Percentage” (page 29-112)

◆ **Fixed Scan Time**

This mode keeps the scan time as specified.

It allows you to process a logic program in a certain cycle. It is suitable for programs that prioritize control (logic programs) and for which the screen is mainly used for monitoring (data display) with few operations required.



Display Processing Time = Setting Value for Fixed Scan Time (ms) - Logic Time

e.g.) If 50 ms is specified for the fixed scan time, and the logic executing time is 20 ms, the display processing time = 50 ms – 20 ms = 30 ms

As the logic time becomes longer, the processing time becomes shorter. For this reason, the display update speed on the GP becomes slower; however, the logic program runs continuously.

IMPORTANT

- The minimum scan time setting is 10 ms.
- For the scan setting, input 10 ms or larger by 1 ms increments.
- If the logic time exceeds the setting value for the fixed scan time; 50% for large and 30% for medium, the scan time is automatically adjusted to be twice as long as the logic time.
e.g.) When 50 ms is specified for the fixed scan time and the logic time is 30 ms, the scan time will be 60 ms.

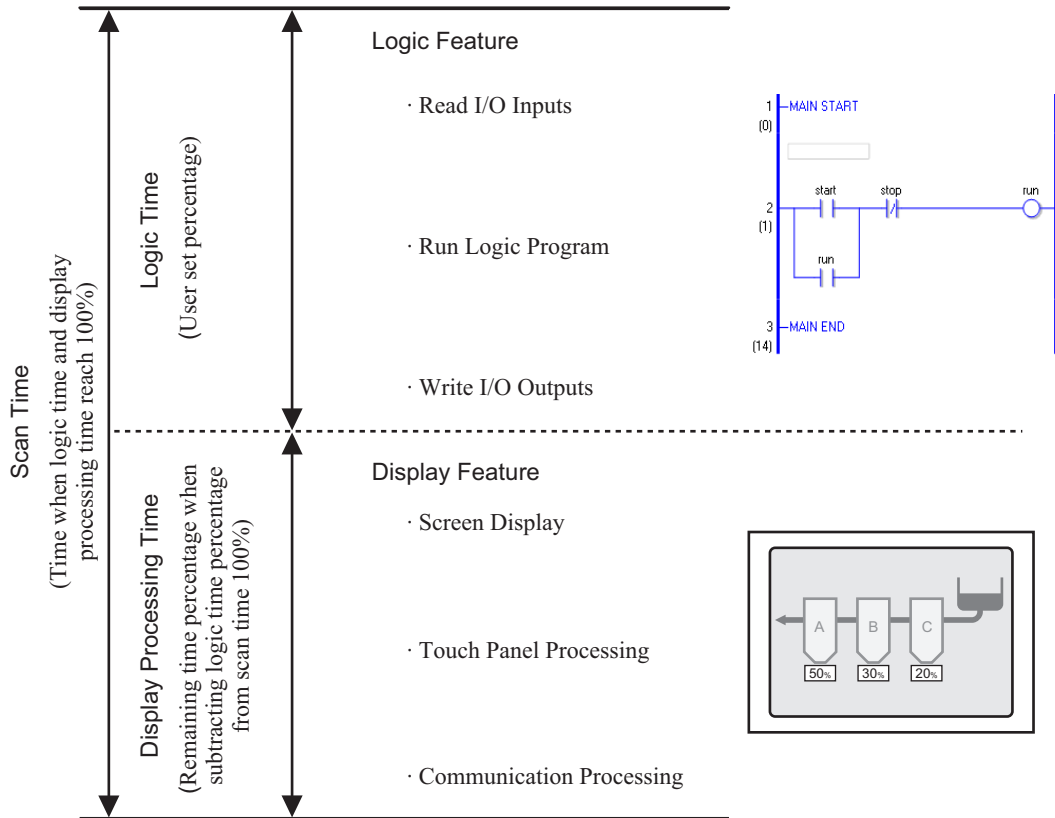
NOTE

- Adjust the setting time based on the #L_AvgScanTime value after testing the operation on the GP.
☞ “A.5 System Variables” (page A-73)

◆ CPU Scan Percentage

This mode specifies the logic time occupancy (%) during scan time and changes the scan time for operation.

The mode prevents pressure on the display processing time caused by increased logic time, and it is suitable for systems that prioritize speed in screen operations and screen switching.



Scan Time = Logic Time ÷ Setting Value for the CPU Scan Percentage (%)

e.g.) If 40% is specified for the CPU scan percentage and the logic executing time is 20 ms,

$$\text{Scan Time} = (20 \div 40) \times 100 = 50 \text{ ms}$$

$$\text{Display Processing Time} = 50 \text{ ms} - 20 \text{ ms} = 30 \text{ ms}$$

As the logic time becomes longer, the display processing time and the scan time become longer. For this reason, as the logic time becomes longer, the time allocated for the display processing becomes longer. This results in improved display update speed on the GP while slowing down the processing cycle of the logic programs.

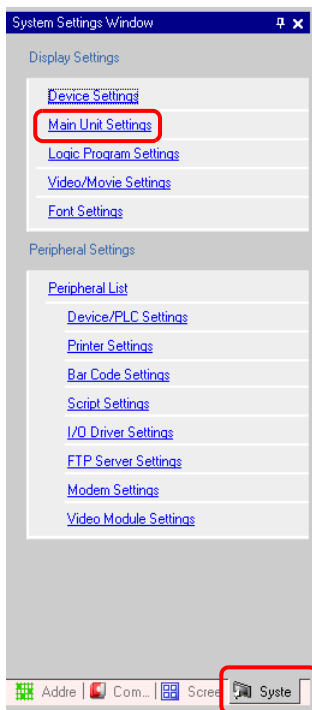
IMPORTANT

- Specify the scan time value for the CPU scan percentage by 1 ms increments.
- The processing time per instruction in the logic program remains unchanged.
- You cannot specify a CPU scan percentage larger than 50%.
- If 50% is specified for the CPU scan percentage, the display processing time and logic program processing time will be the same length. The display processing will not be prioritized.

◆ Setup Procedure

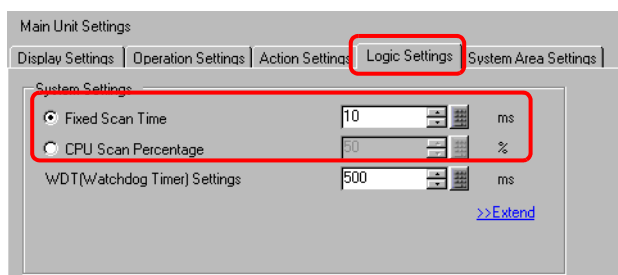
- NOTE** • Please refer to the settings guide for details.
 ☞ “ ■ [Main Unit Settings] Settings Guide” (page 29-126)

1 In the [System Settings Window], click [Main Unit Settings].



- NOTE** • If the [System Settings Window] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings Window (S)].

2 Click the [Logic Settings] tab. In [System Settings], select [Fixed Scan Time] or [CPU Scan Percentage] and enter a value for the setting.

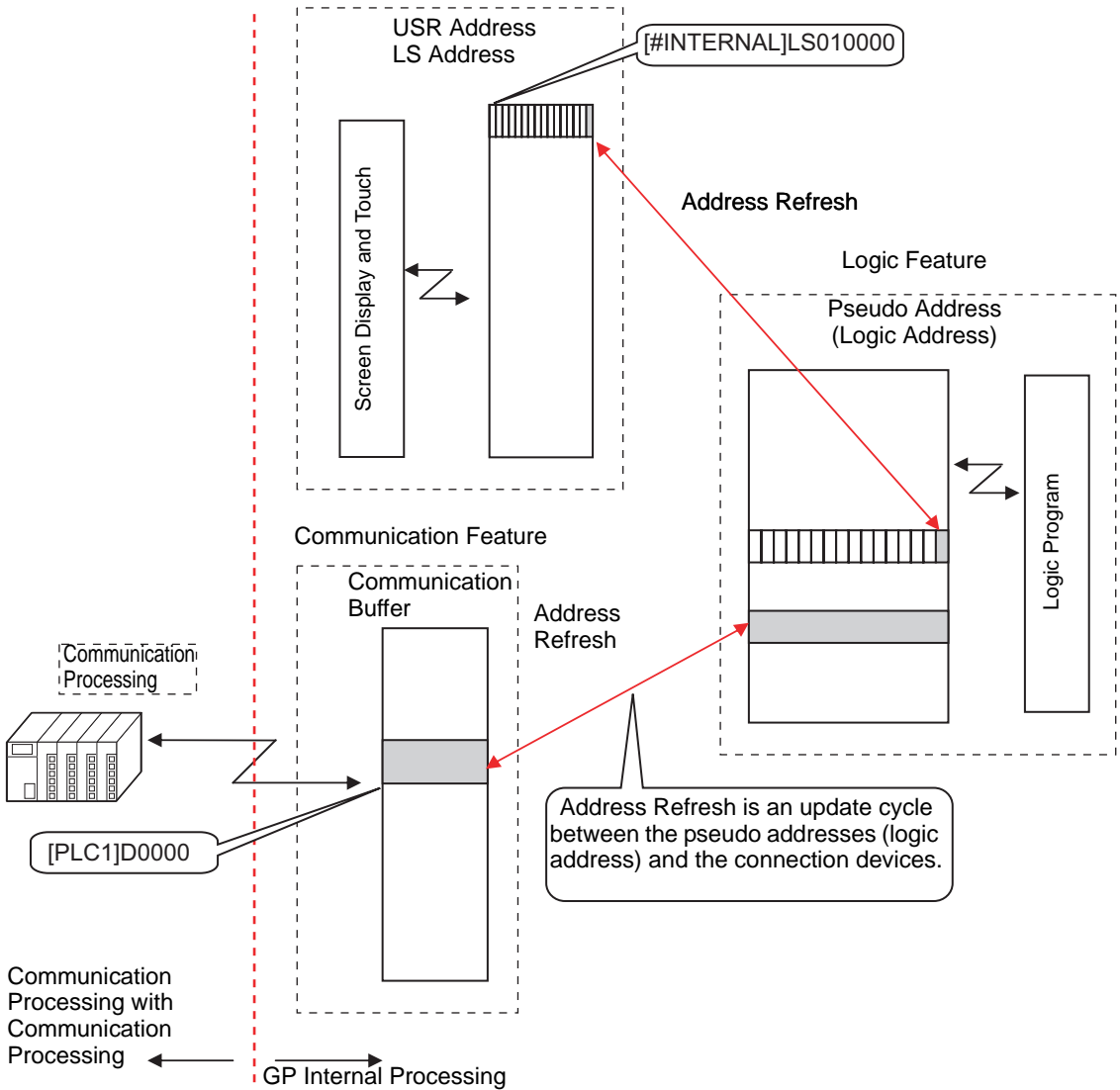


■ Address Refresh

◆ Summary


When device addresses are used in logic programs, pseudo addresses (logic addresses) of the logic features are allocated. The device addresses are updated periodically and logic programs are run through these allocated pseudo addresses.

Address Refresh is the method of updating the data between device addresses and pseudo addresses.

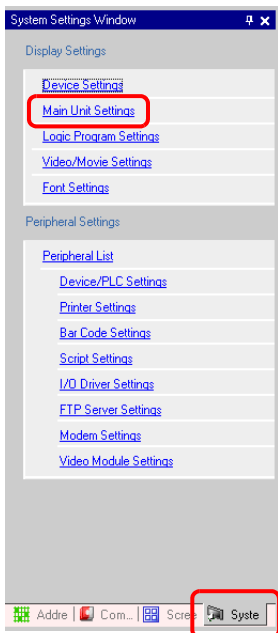


◆ Setup Procedure

NOTE

- Address Refresh can not be set up in off-line mode.
- Please refer to the settings guide for details.
 “ ■ [Main Unit Settings] Settings Guide” (page 29-126)

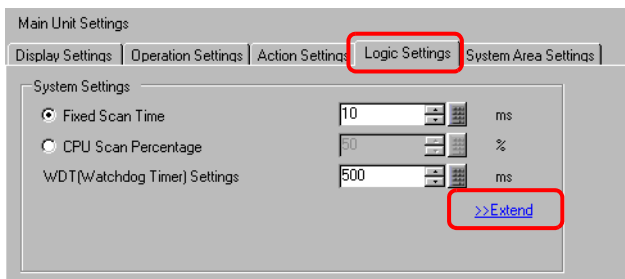
1 In the [System Settings Window], click [Main Unit Settings].



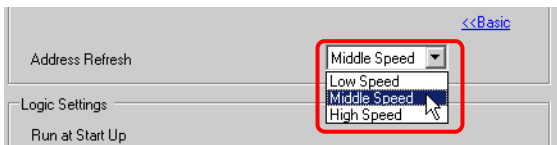
NOTE

- If the [System Settings Window] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings Window (S)].

2 Click the [Logic Settings] tab and in [System Settings], click [>>Extend].



3 In [Address Refresh], select the speed.

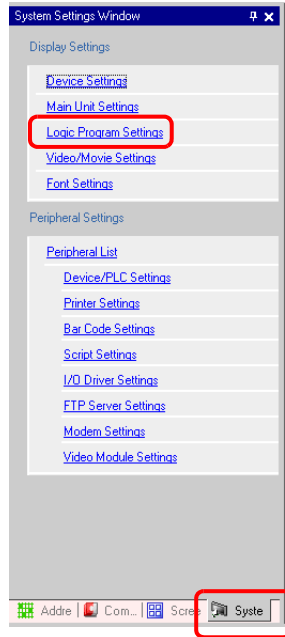


29.13.4 Enhancing Security

You can enhance security when monitoring logic programs by allowing access only to users with passwords.

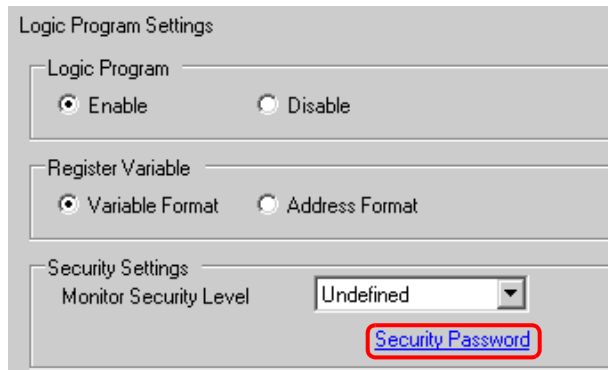
■ Setup Procedure

1 In the [System Settings Window], click [Logic Program Settings].



-
- NOTE** • If the [System Settings Window] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings Window (S)].
-

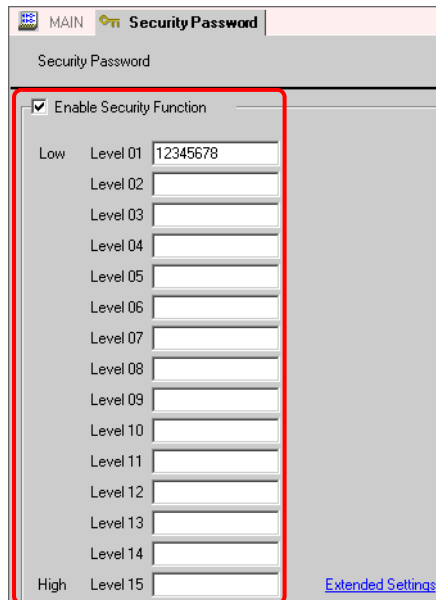
2 Click [Security Password].



- NOTE** • If you select a [Monitor Security Level] without setting up a security password, the below dialog box will be displayed. Click [OK] and set up a security password.

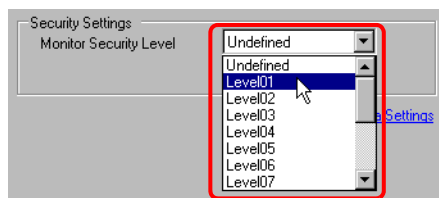


- 3 The [Security Password] dialog box appears. Select the [Enable Security Function] check box and enter the password in the relevant level.



- NOTE** • You can configure advanced security feature settings. For details about the settings, refer to the settings guide.
☞ “◆ Password Settings” (page 29-129)


- 4 In [Security Settings], select [Monitor Security Level].



29.13.5 Using Reference Features to Search Logic Programs

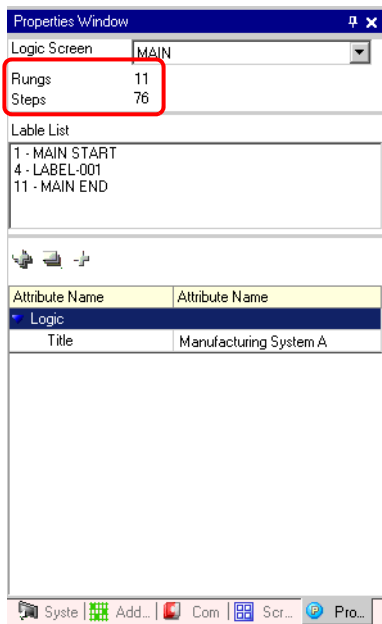
In the [Properties Window], you can search for rungs and instructions in logic programs and display the details in [Program Window]. You can edit symbol variables in [Properties Window].

NOTE

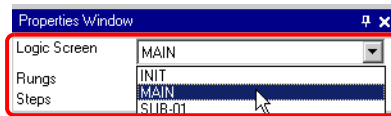
- Please refer to the settings guide for details.
 "5.13.5 [Work Space] Settings Guide ■ Screen List Window" (page 5-89)

■ [Properties Window] Display

- 1 In [Logic Screen], open the logic screen you want to display.
- 2 On the [View (V)] menu, point to [Work Space (W)] and click [Properties Window (P)].
The total numbers of rungs and steps are displayed.

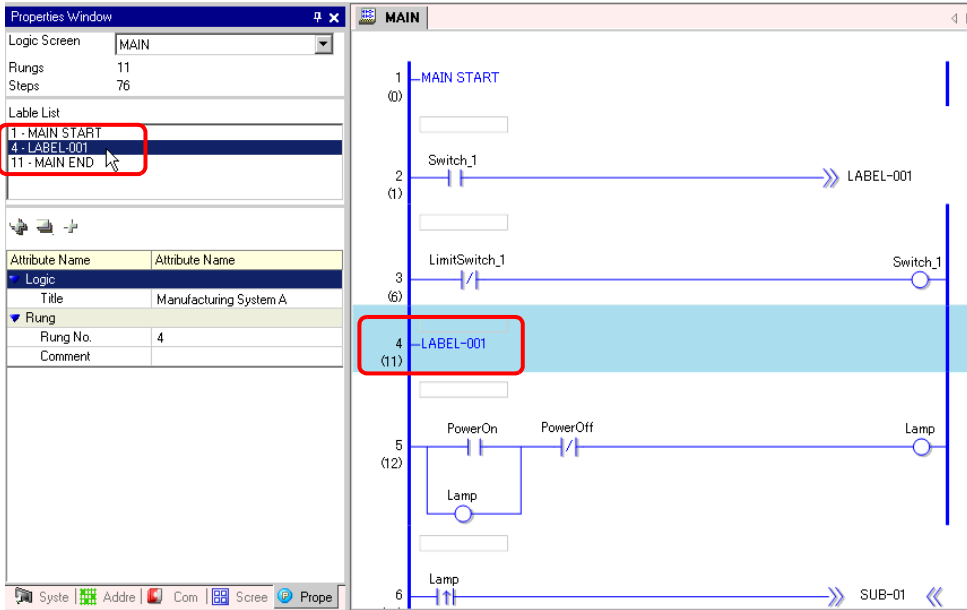

NOTE

- In [Logic Screen], you can select the logic program.

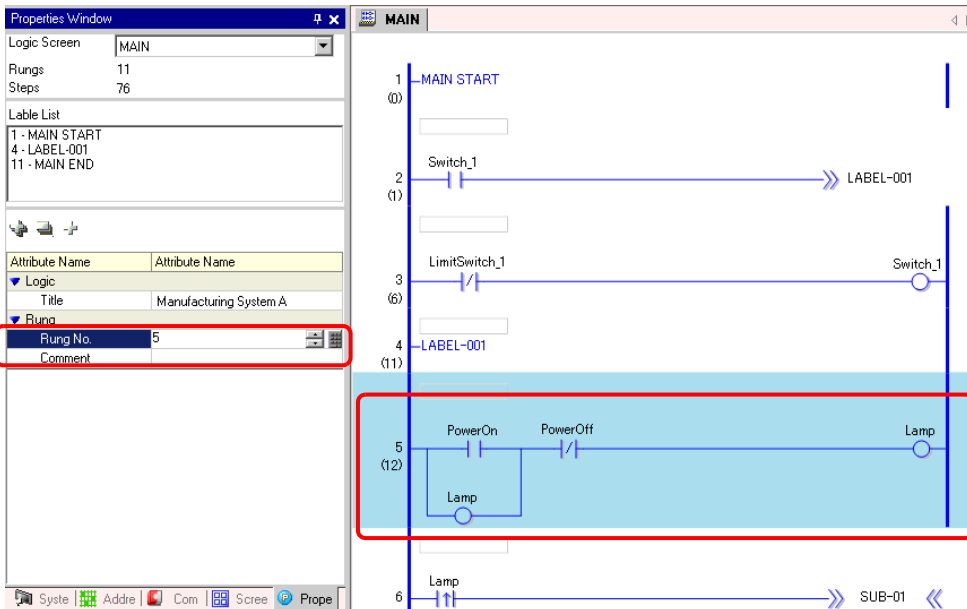


■ Properties Window Search

- In [Label List], all the logic program labels are displayed. Select a label to move the cursor to the selected logic program label.

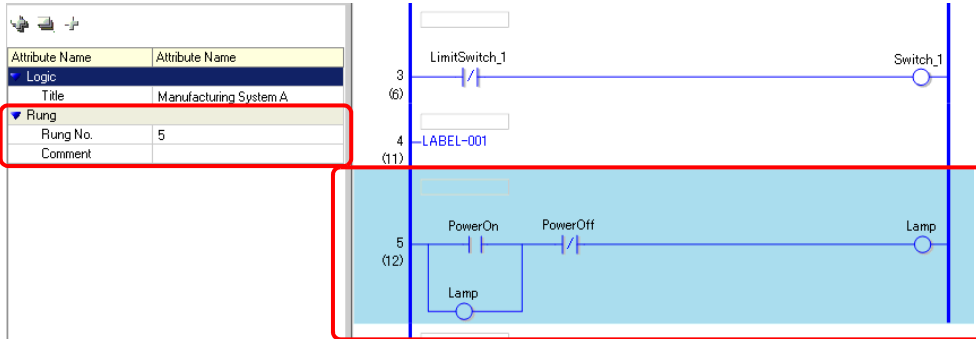



- Click [Rung No.] and enter a number to move the cursor to that rung.

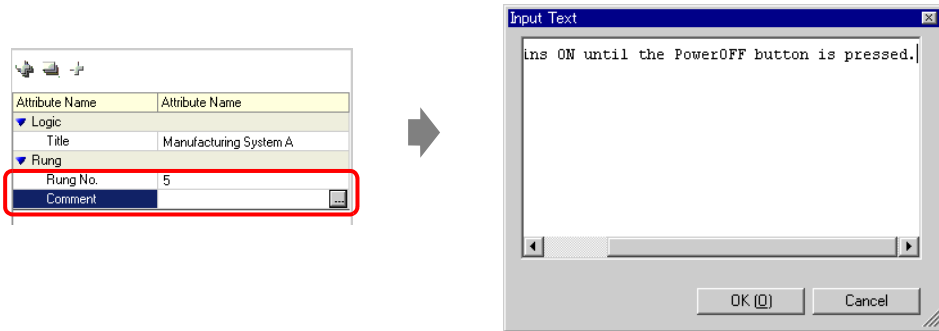


■ Display and Edit the Logic Program Information in Program Windows

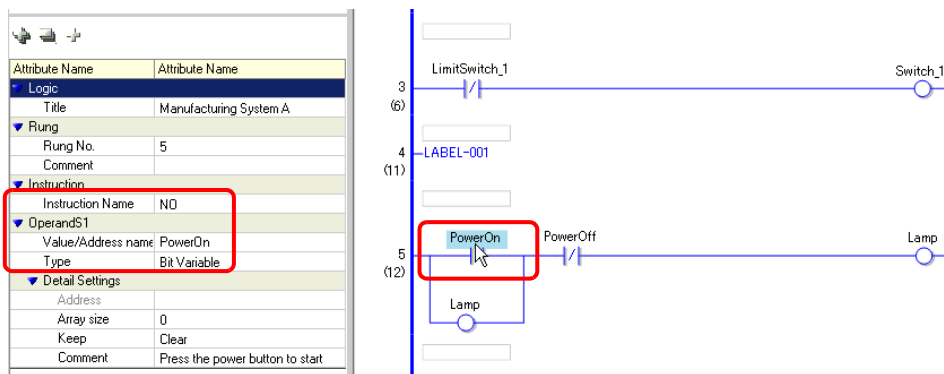
- Select a rung in the logic program to display [Rung No.] and [Comment] for the selected rung.





- Click in the box to the right of [Comment] and then  to display the [Input Text] dialog box. You can edit the comment in the dialog box.



- Select an instruction or operand in the logic program to display [Instruction Name], [Value/Address name], [Type], and the detailed settings. Click any of these to edit the settings.



NOTE • If the detailed settings are not displayed, click  (or  located to the left of [Detail Settings]) to display [Address], [Array size], [Keep], and [Comment]. The settings that you can edit differ depending on [Type].

29.13.6 Using Previously Created Logic Programs

You can register a previously created part of a logic program or a subroutine program as a logic part. You can call the logic program part from another project file as well as from the logic program that you are currently editing.

Registering frequently used logic programs saves you from creating the same program over and over and reduces your workload.

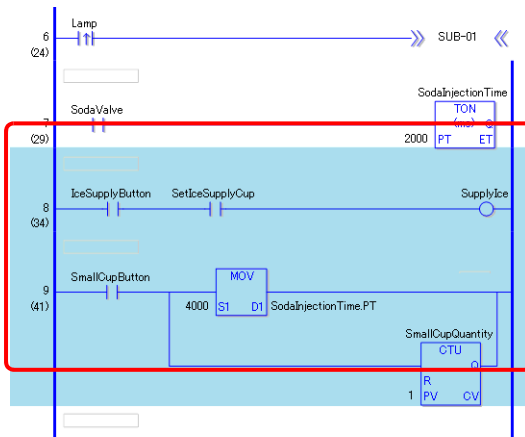
NOTE • You can call the logic parts only on the PC where the logic parts have been registered.

■ Registering Logic Parts

◆ Registering Rung Parts

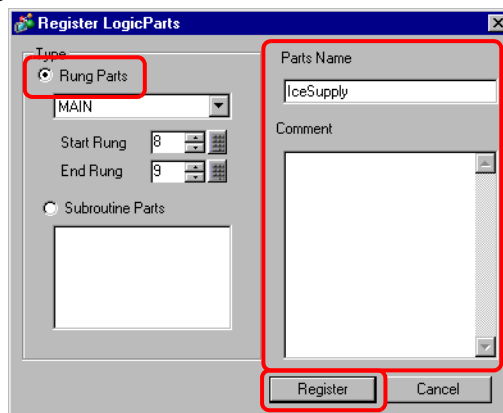
Registering a part of a rung as a logic part.

- 1 Select the range of rungs that you want to register as the part.



- 2 On the [Logic (L)] menu, select [Save Parts (S)] to display the [Register LogicParts] dialog box.

- 3 In [Type], select [Rung Parts]. Enter [Parts Name] and [Comment] and click [Register].

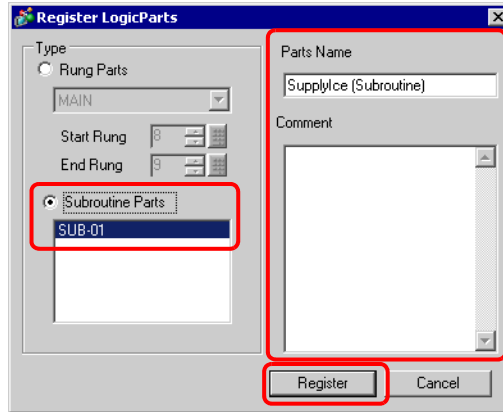


NOTE • If the [Register LogicParts] dialog box is displayed without selecting a rung, you can specify the range by selecting the logic program and entering the [Start Rung] and [End Rung].

◆ **Registering Subroutine Parts**

Registering a subroutine program as a logic part.

- 1 On the [Logic (L)] menu, select [Save Parts (S)] to display the [Register LogicParts] dialog box.
- 2 In [Type], select [Subroutine Parts] and the subroutine name. Enter the [Parts Name] and [Comment] and click [Register].

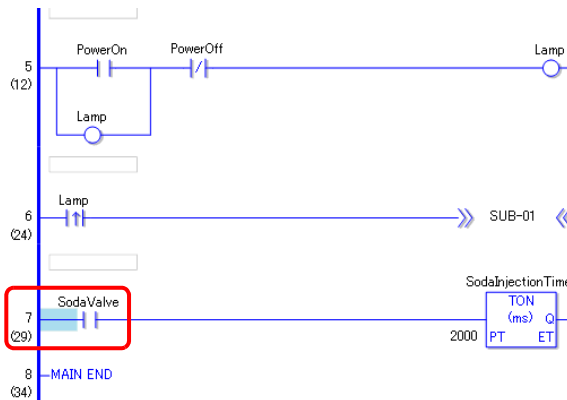


■ **Calling Logic Parts**

You can call a registered logic part and insert the part in the logic program that you are editing.

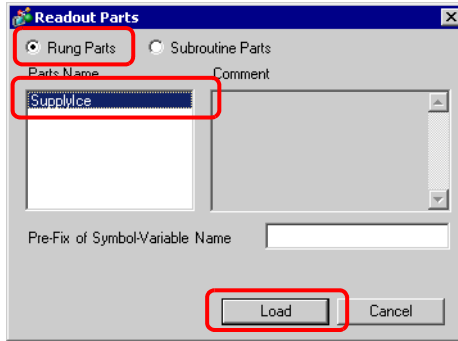
◆ **Inserting Rung Parts in Logic Programs**

- 1 Select the rung one up from where you want to insert the part or a part (power bar, instruction, etc.) of the rung.



- 2 On the [Logic (L)] menu, select [Load Parts (P)] to display the [Readout Parts] dialog box.

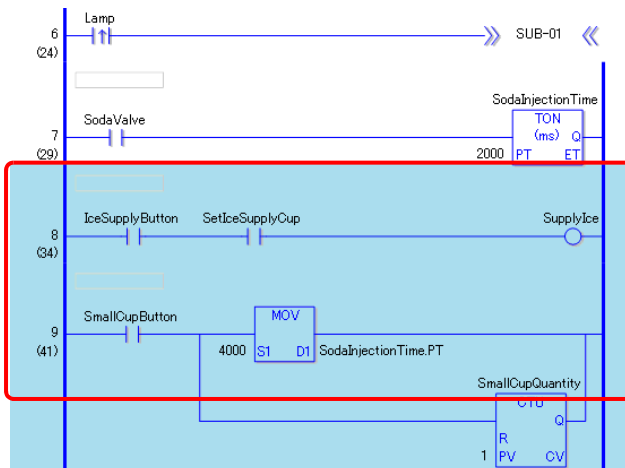
3 Select [Rung Parts], and in [Parts Name], select the part of the rung that you want to insert and click [Load].



- NOTE** • If there is a possibility of duplicating the symbol variable, enter any text in [Pre-Fix of Symbol-Variable Name] to prevent duplication of the symbol variable. The input text will be added to the front of the symbol variable name of the rung part, and the symbol variable name will be changed. The rung part will then be inserted in the logic program that you are editing.
 (e.g.) When entering “A Line” in [Pre-Fix of Symbol-Variable Name],

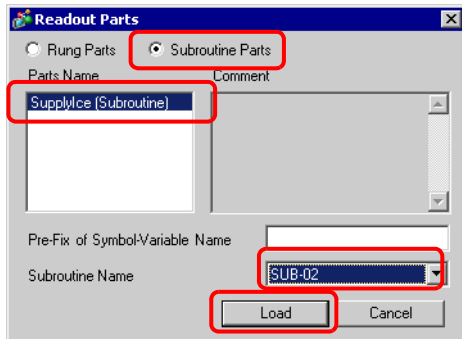
	Rung Parts	After Insertion of the Logic Program
Symbol Variable Name	Switch1	A Line Switch 1

4 The rung part is inserted.

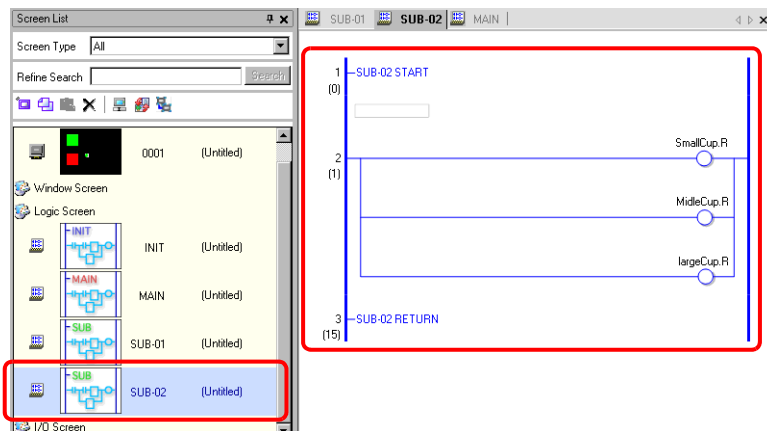


◆ Add Subroutine Parts

- 1 On the [Logic (L)] menu, select [Load Parts (P)] to display the [Readout Parts] dialog box.
- 2 Click [Subroutine Parts], and enter a [Parts Name]. Select [Subroutine Name] and click [Load].



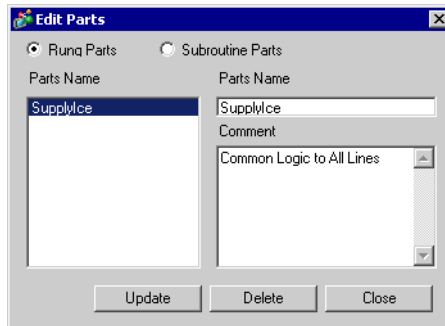
- 3 The subroutine screen with the selected subroutine name will be added.



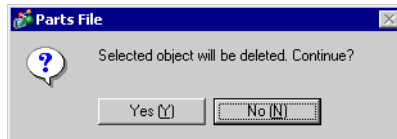
■ Editing Logic Parts

You can edit and delete part names and comments for registered logic parts.

- 1 On the [Logic (L)] menu, select [Edit Parts (E)] to display the [Edit Parts] dialog box.
- 2 Select [Rung Parts] to edit the rung part or select [Subroutine Parts] to edit the subroutine part, and then select the [Parts Name] for the part that you want to edit.
- 3 To edit [Parts Name] or [Comment], modify the text in [Parts Name] or [Comment] and then click [Update].



To delete parts, click [Delete]. When the below dialog box is displayed, click [Yes].



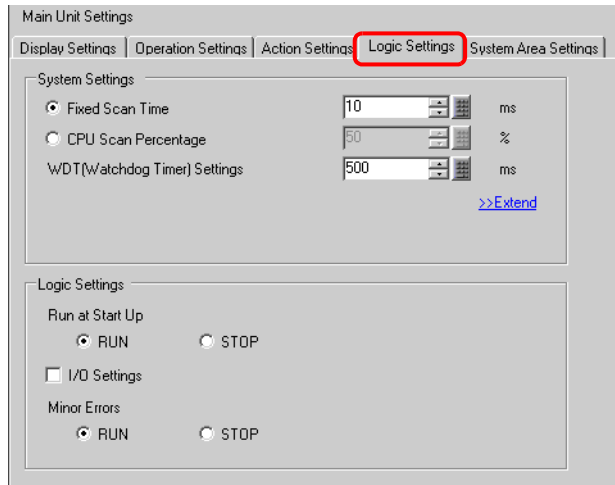
29.14 Settings Guide


29.14.1 [System Settings Window] Settings Guide for Logic Features

The following explains the logic feature settings that must be specified in [System Settings Window].


■ [Main Unit Settings] Settings Guide

◆ Logic Settings



Setting	Description
System Settings	Configure system settings for logic features.
Fixed Scan Time/CPU Scan Percentage	<p>Select the logic scan time mode.</p> <ul style="list-style-type: none"> When [Fixed Scan Time] is selected, you can specify the logic time cycle from 10 ms to 2000 ms. ☞ “◆ Fixed Scan Time” (page 29-111) When [CPU Scan Percentage] is selected, you can specify the logic time occupancy. The available settings range is 0 to 50 (Unit: %) for any model. The recommended setting values are 10 to 30 for the GP33** series (logic function support model); and 10 to 50 for other models. ☞ “◆ CPU Scan Percentage” (page 29-112)
WDT (Watchdog Timer) Settings	<p>You can configure the monitoring time for the logic scan time. An error will occur if the logic scan time exceeds the WDT (Watchdog Time). The settings range from 100 ms to 3000 ms.</p>
>>Extend/<<Basic	Click [>>Extend] to specify the [Address Refresh] speed.
Address Refresh	<p>Select the address refresh speed from [Low Speed], [Middle Speed], and [High Speed].</p>  <p>☞ “■ Address Refresh” (page 29-114)</p>

Continued

Setting	Description
Logic Settings	<p>Click [Keep Area Settings] to display the [Keep Setting] dialog box.</p> <p>The keep/clear points can be specified for symbol variables in [Variable Format]. The keep/clear ranges for symbol variables can be specified in [Address Format].</p> <p> “ ■ Keep Area Settings” (page 29-16)</p>
Run at Start Up	Select either [RUN] or [STOP] as the logic program status when the GP starts up.
I/O Settings	Determine whether or not to enable input and output to and from the I/O unit. Select the check box to enable input and output.
Minor Errors	Select whether to [RUN] or [STOP] the logic program when a minor error occurs.

■ [Logic Program Settings] Setup Guide

Logic Program Settings

Logic Program
 Enable Disable

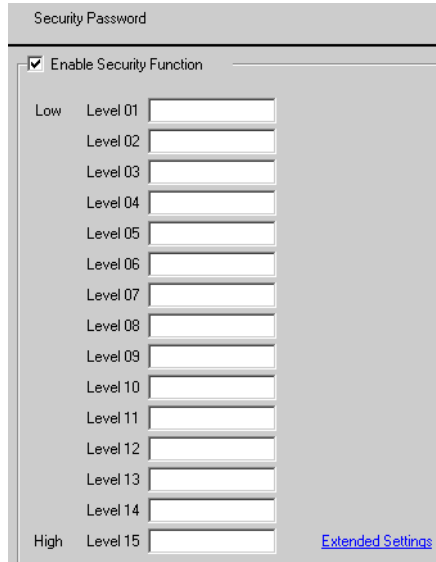
Register Variable
 Variable Format Address Format

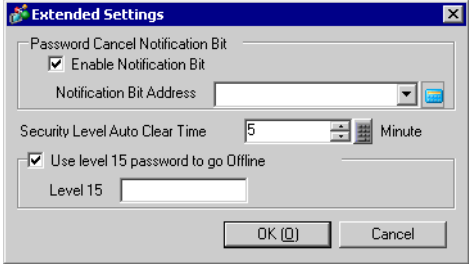
Security Settings
 Monitor Security Level: Undefined
[Security Password](#)

[Keep Area Settings](#)

Setting	Description
Logic Program	Select whether to [Enable] or [Disable] the logic features. “29.2.1 Using Logic Functions” (page 29-3)
Register Variable	Select [Variable Format] or [Address Format] to register an address. “29.3.2 Using Symbol Variables with Arbitrary Names (Variable Format)” (page 29-19) “29.3.3 Using Symbol Variables with Fixed Addresses (Address Format)” (page 29-31)
Security Settings	Set up a password during online monitoring to enhance security.
Monitor Security Level	Select the level of monitoring security. The settings range from [Undefined], [Level 01]-[Level 15].
Security Password	Click [Security Password] to display [Password Settings]. Configure the detailed settings for security features. “◆ Password Settings” (page 29-129)
Keep Area Settings	Click [Keep Area Settings] to display the [Keep Setting] dialog box. The keep/clear points can be specified for symbol variables in [Variable Format]. The keep/clear ranges can be specified for symbol variables in [Address Format]. “■ Keep Area Settings” (page 29-16)

◆ Password Settings

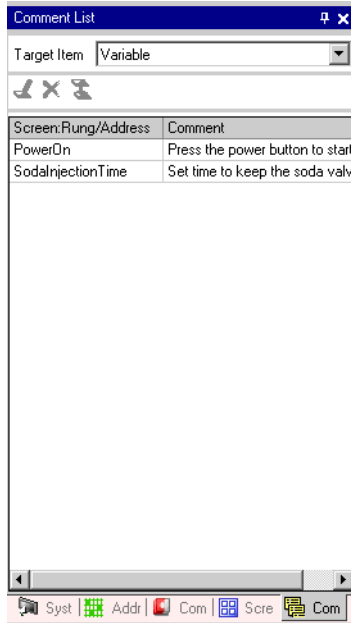






Setting	Description
Enable Security Function	Determine whether to use security features. Select the check box to set up a password.
Level	Enter a password in the relevant level with a maximum of 8 single-byte characters (numbers and letters).
Extended Settings	<p>Click [Extended Settings] to display the [Extended Settings] dialog box.</p> 
Password Cancel Notification Bit	Determine whether to use the password cancel notification bit. Select the check box to specify the notification bit address.
Security Level Auto Clear Time	Configure the security level auto clear time. The settings range from 0-60 minutes.
Use level 15 password to go Offline	Determine whether to use a level 15 password when switching to off-line mode. Select the check box to set up the password.

29.14.2 [Work Space]Setting Guide

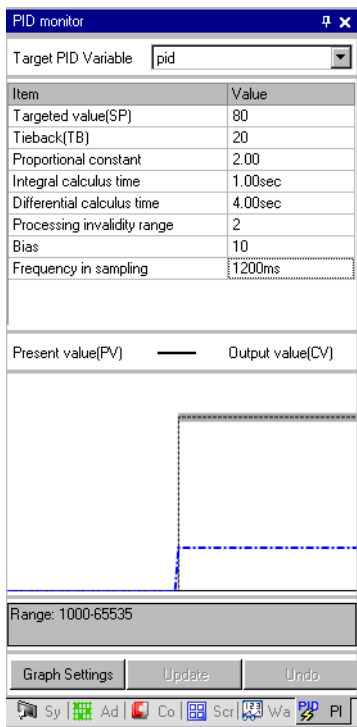
The following explains the windows displayed in the work space for using logic features.

■ Comment List



Setting		Description
Target Item		Select [Variable], [System Variable], or [Rung] to display the comments.
Operation Button	Edit 	You can edit comments in [Variable] and [Rung].
	Delete 	You can delete comments in [Variable] and [Rung].
	Add 	When you click the icon, the [Address Input] dialog box is displayed only in [Address Format], selected in [Register Variable]. You can specify addresses and add logic addresses.  “ ■ Logic Address Display” (page 29-34)
Screen: Rung/Address		The symbol variable name is displayed in [Variable]. The system variable name is displayed in [System Variable]. Double-click to switch to the [Address Settings] window and the relevant variable will be selected. The logic name and rung number of the logic screen containing the rung are displayed in [Rung]. Double-click to select the relevant rung in the logic program.
Comment		The comment for the selected rung is displayed. Double click to edit [Variable] and [Rung].

■ PID Monitor



Setting	Description				
Target PID Variable	Select the PID variable that you want to monitor.				
List of PID Adjustments	You can input values and adjust the PID while referring to the graph.				
Graph Display	The PID instruction values are displayed in a graph that can be monitored.				
Graph Settings	You can specify the details of the graph. Click to display the settings dialog box. <div data-bbox="655 1219 993 1431" data-label="Image" style="text-align: center;"> </div>				
<table border="1" style="width: 100%; border-collapse: collapse;"> <tr> <td style="width: 20%;">Displayed Items</td> <td>Select the check box to display [Current Value], [Target Value], [Output Value], [Output Invalidity Range], or [Output Range].</td> </tr> <tr> <td>Graph Display Range</td> <td>Specify [Hi limit], [Low limit] and [Display width] for the graph display range.</td> </tr> </table>	Displayed Items	Select the check box to display [Current Value], [Target Value], [Output Value], [Output Invalidity Range], or [Output Range].	Graph Display Range	Specify [Hi limit], [Low limit] and [Display width] for the graph display range.	
Displayed Items	Select the check box to display [Current Value], [Target Value], [Output Value], [Output Invalidity Range], or [Output Range].				
Graph Display Range	Specify [Hi limit], [Low limit] and [Display width] for the graph display range.				
Update	The graph must be updated with the values specified for the PID adjustment.				
Undo	Return to the state before PID adjustment values were input.				

◆ PID Adjustments

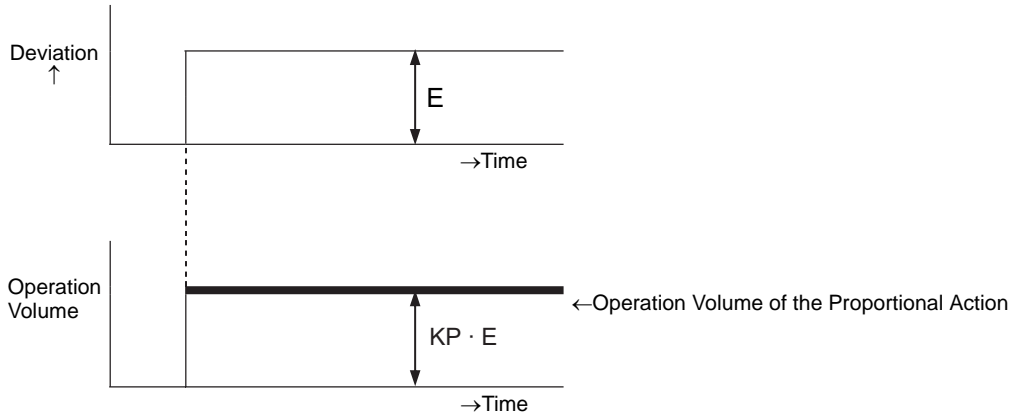
Item	Value
Targeted Value (SP)	Specify the target value. Enter values for the minimum and maximum output. The range depends on the PID instruction output settings. For details, refer to the PID instructions. ☞ “Chapter 30 Instruction List” (page 30-1)
Tieback (TB)	Specify a value for output during power off. The range for input depends on the PID instruction output settings. For details, refer to the PID instructions. ☞ “Chapter 30 Instruction List” (page 30-1)
Proportional Constant	Specify the proportion for comparison control. A larger value means that the target value will be reached sooner. A smaller value means the target value will be approached more gradually, resulting in reduced overshooting. The settings range from 0.01-1000.00. For details about proportional constants, see the next page. ☞ “◆ Proportional Action (P Action)” (page 29-133)
Integral Calculus Time	Specify the intervals between integral calculations. The settings range from 0.10 s - 3000 s. For details about integral calculus time, see the next page. ☞ “◆ Integral Action (I Action)” (page 29-133)
Differential Calculus Time	Specify the intervals between differential calculations. The settings range from 0 - 3000 s. For details about differential calculus time, see the next page. ☞ “◆ Derivative Action (D Action)” (page 29-134)
Processing Invalidity Range	Specify the range in which PID instructions are not operated. The deviation within the settings range is “0” and \pm the processing invalidity range is based on the target value. The settings range from 0 to (maximum output value – minimum output value) / 2.
Bias	The value specified here is added to the output value for operation. The settings range from the minimum output value to the maximum output value.
Frequency in Sampling	Specify the sampling frequency for the PID operation. The frequency depends on the scan time and the PID instruction is operated in the scan after the specified frequency. The settings range from operation frequency to 60000 (ms).

◆ Proportional Action (P Action)

Calculate the operation volume (output value) proportionate to the deviation (deviation between the target value and current value). The formula for the relation between deviation (E) and operation volume (CV) is as follows.

$$CV = KP \times E \text{ (The constant of proportion calls } KP \text{ the “proportional gain.”)}$$

When the deviation is fixed, the proportional action is as follows.



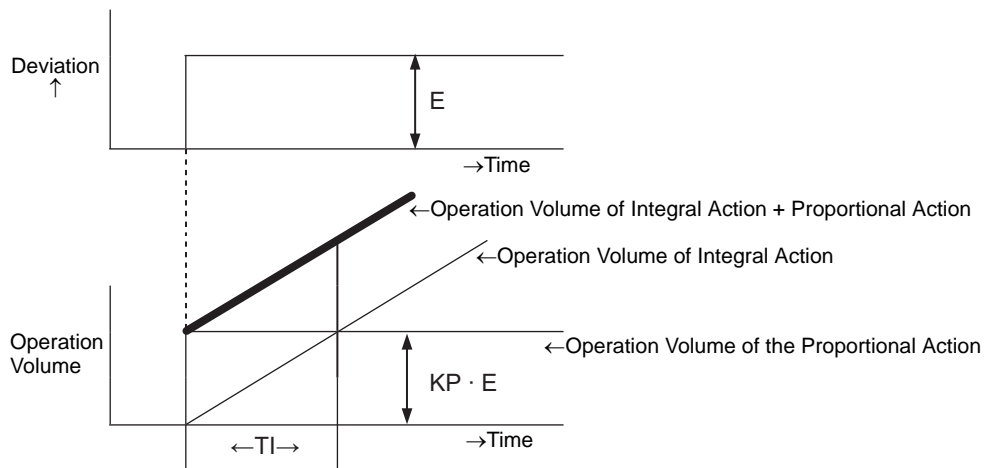
The operation volume changes within the range of 0-4095 (initial value). As KP increases, the operation volume proportionate to the deviation increases and the correcting action strengthens, and this causes offset (residual deviation).

◆ Integral Action (I Action)

Continuously change the operation volume (output value) to eliminate any deviation (deviation between the target value and current value). Doing so can eliminate the offset from the proportional action.

Once deviation is caused in the integral action, the operation volume of the action changes to the operation volume of the proportional action. The time required for the change is called the “integral calculus time.” The time is indicated as TI. A smaller TI results in a stronger integral action.

If the deviation is fixed, the integral action is as follows.



Use integral action as “PI action” combined with the proportional action or as “PID action” combined with the proportional and derivative action.” You cannot use integral action alone.

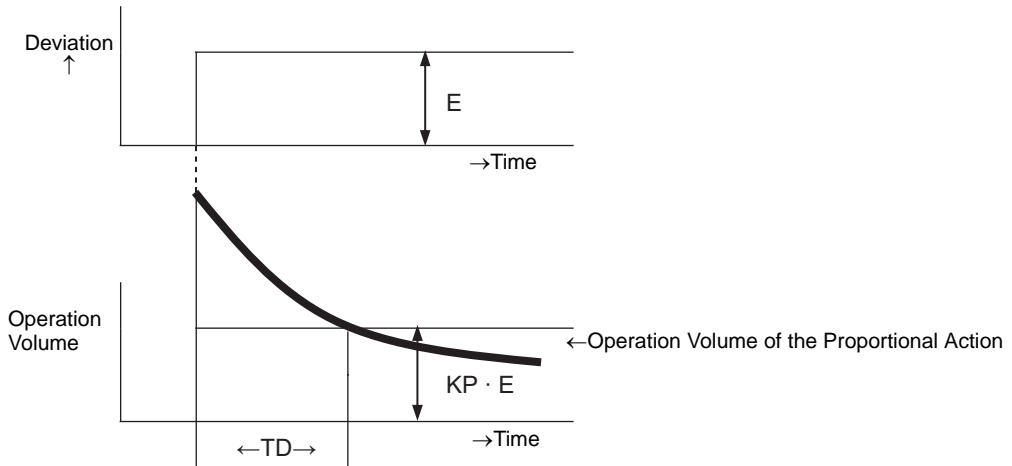
◆ Derivative Action (D Action)

Add the operation volume (output value) proportionate to any deviation (deviation between the target value and current value) to eliminate deviation. Doing so prevents the control target from drastically changing due to an external disturbance.

Once deviation has occurred in the derivative action, the operation volume of the action changes to the operation volume of the integral operation. The time required for the change is called the “differential calculus time” and is indicated as TD.

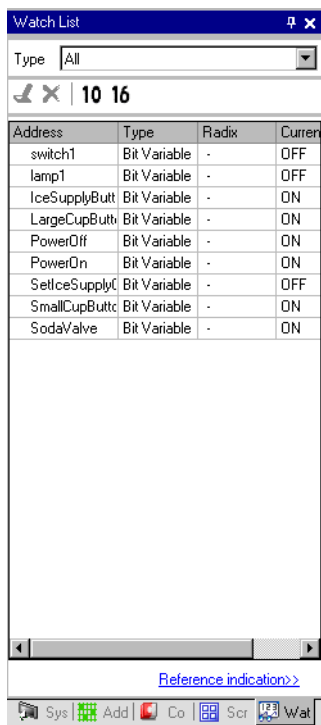
A larger TD results in stronger derivative action.

If the deviation is fixed, the derivative action is as follows.



Use derivative action as “PD action” combined with the proportional action or as “PID action” combined with the proportional action and integral action.” You cannot use derivative action alone.

■ Watch List



Setting		Description
Type		Select the type of symbol variable or system variable registered in the [Watch List] window. For how to register, please refer to the following. ☞ “29.10.2 Monitor the Current Values of Symbol Variables” (page 29-79)
Operation Button	Edit	You can edit registered symbol variables.
	Delete	You can delete registered symbol variables.
	Decimal 10	Change the display to decimal format.
	Hexadecimal 16	Change the display to hexadecimal format.
Address		The variable name added to the watch list is displayed.
Type		The variable type added to the watch list window is displayed.
Radix		The variable format added to the watch list is displayed.
Current Value		The current value added to the watch list is displayed. If the type is [Bit Variable], right-click and then click [ON] or [OFF]. If the type is [Integer Variable], [Float Variable], or [Real Variable], right-click to input the value.
Reference Indication		You can configure the settings only for the [Integer Variable] type. Select [Specify Bit], [Specify Byte], or [Specify Word]. Decimal or hexadecimal format can be specified in [Specify Byte] and [Specify Word].

29.15 Restrictions

29.15.1 Scan Time Delay

■ AGP-3300 series

- When you select to [Enable] logic programs, a maximum delay of 6% may occur temporarily.
- When you communicate large data (e.g.: Sequential addresses of 960 words) with a PLC (e.g.: Mitsubishi Q series) through Ethernet, a maximum delay of 30% may occur.
- When you transmit data through AGP's Ethernet function, you must take account of the delay in scan time.
- When you communicate with multiple PLCs (e.g.: Mitsubishi Q series) being connected (Ether), a maximum delay of 100% may occur.
- When you communicate large data (e.g.: 10K bytes) by using Pro-Server EX (product of Pro-face), a maximum delay of 100% may occur in scan time.
When accessing the memory that stores large data (e.g.: 10K bytes) by using Pro-Server EX, you must take account of the delay in scan time.
- When you communicate with MPI protocol, a maximum delay of 30% may occur.

■ AGP-3400/3500/3600/3750 series

- When you communicate large data (e.g.: Sequential addresses of 960 words) with a PLC (e.g.: Mitsubishi Q series) through Ethernet, a maximum delay of 15% may occur. When you transmit data through AGP's Ethernet function, you must take account of the delay in scan time.
 - When communicating large data (e.g.: 10K bytes) by using Pro-Server EX (a product of Pro-face), a maximum delay of 20% may occur in scan time.
When accessing the memory which stores large data (e.g.: 10K bytes) by using Pro-Server EX, you must take account of the delay in scan time.
 - When you communicate with MPI protocol, a maximum delay of 15% may occur.
 - When you record or play movies from the FTP server with the multimedia function, a maximum delay of 15% may occur.
-
- When a program is being uploaded, the scan time cannot be guaranteed.
 - When GP is reading the CF-Card, the scan time cannot be guaranteed.
 - When a logic or I/O driver error occurs, about 10ms of delay may occur in scan time.
 - When many devices are connected on the same LAN, the scan time cannot be ensured. It is recommended that you divide LAN physically, such as control LAN and data LAN.
More specifically, prepare two LAN cards for your PC, use one card to establish LAN for a control system to which AGP belongs, and use another card to establish LAN for data communication. In other words, create two groups of LAN for the PC.
 - When you use logic functions and record or play movies with the multimedia function simultaneously, the operation of recording or playing movies may be interrupted.

- When using connection device addresses (excluding internal addresses) in the logic program, note that the data refresh of the connection device and the logic program execution do not synchronize.

Since data values may not be applied at the start of logic program execution, data values are undefined until the communication with the connection device is established. When using connection device addresses in the logic program, use them after you confirm that the data in the connection device addresses has been read.

e.g.: When a special relay (normally ON) of the connection device is used

Connection device: Special relay (normally ON)

Logic program: Use the special relay in the logic program, and use the connection device address after you recognize that the special relay turned ON. When different connection devices are used, you need to handle them individually.

- Please pay attention to the number of registered connection device addresses (excluding internal addresses) used in the logic program. Writing to a connection device address affects the screen change speed depending on the communication speed and the number of registered addresses. Frequent writing to connection device addresses may cause writing errors. There is no particular problem when reading from connection devices. The following shows an example of the number of addresses for writing data to a connection device.

e.g.: When data is updated every 10 ms with a logic program, use addresses with 120 words at maximum.

Connection device: MELSEC FX

Communication speed: 115,200 bps

Address refresh: Middle Speed

Memo