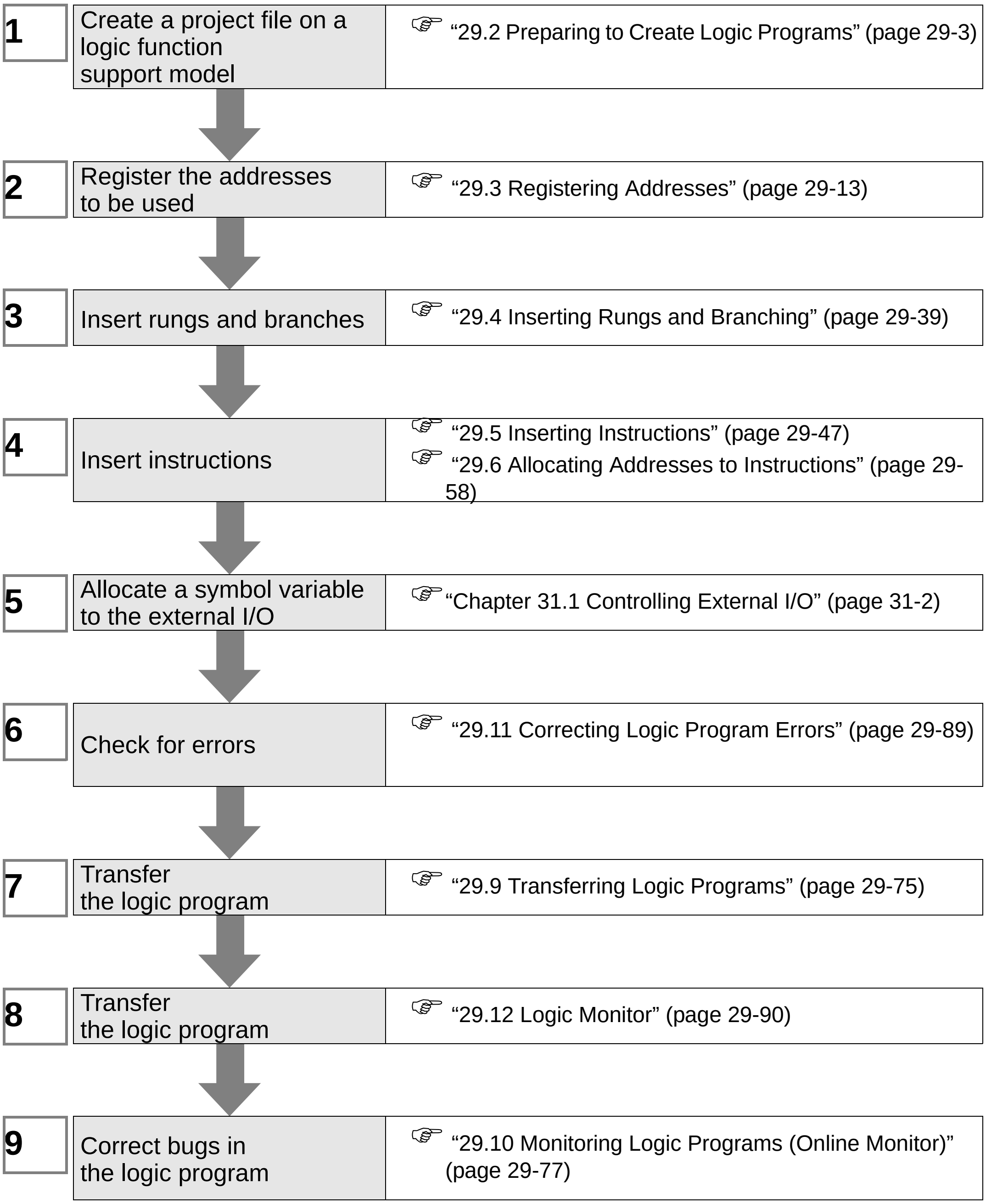
29 Logic Programming

This chapter describes the addresses that you can use in GP-Pro EX and GP, and how to create Logic Programs using the GP-Pro EX Logic Functions. Start by reading "29.1 Logic Programming Steps" (page 29-2) and then turn to the relevant sections.

29.1 Logic Programming Steps

Use the following steps to create logic programs.



1	Create a project file on a logic function support model	☞ “29.2 Preparing to Create Logic Programs” (page 29-3)
2	Register the addresses to be used	☞ “29.3 Registering Addresses” (page 29-13)
3	Insert rungs and branches	☞ “29.4 Inserting Rungs and Branching” (page 29-39)
4	Insert instructions	☞ “29.5 Inserting Instructions” (page 29-47) ☞ “29.6 Allocating Addresses to Instructions” (page 29-58)
5	Allocate a symbol variable to the external I/O	☞“Chapter 31.1 Controlling External I/O” (page 31-2)
6	Check for errors	☞ “29.11 Correcting Logic Program Errors” (page 29-89)
7	Transfer the logic program	☞ “29.9 Transferring Logic Programs” (page 29-75)
8	Transfer the logic program	☞ “29.12 Logic Monitor” (page 29-90)
9	Correct bugs in the logic program	☞ “29.10 Monitoring Logic Programs (Online Monitor)” (page 29-77)

NOTE

- For the complete development flow, please see the following.
  ☞ “4.2 Creating HMI Displays and Logic Programs” (page 4-3)

29.2 Preparing to Create Logic Programs

NOTE

- The procedures to start/end GP-Pro EX and save project files are the same as the procedures to create a new screen.
  ☞"Chapter 5 Start to Finish" (page 5-1)

29.2.1 Using Logic Functions

■ If your model supports logic functions

To enable logic programming settings, select a model that supports logic functions in [display type] when you create a new project file.

NOTE

☞ "1.3 List of Supported Functions by Device" (page 1-5)

■ If your model does not support logic functions

When you create a project file and in [Display Unit] 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 if it does not support logic functions.

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

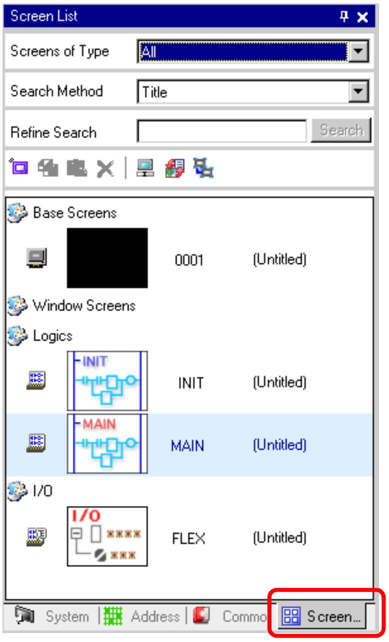
Logic programs consist of the following three types of logic.

Logic Type	Logic Name	Description
Initialize Logic	**INIT**	The logic is run only once when the GP starts up. You can create only one INIT program in a project file. Start label: "INT START" End Label: "INIT END"
Main Logic	**MAIN**	The logic program is run after the initialize logic has been run. 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 to 32) End Label: "SUB-** RETURN"(**01 to 32)

- The Initialize Logic and Main Logic are created in advance.
- You can add a subroutine with the [New Screen] command.
- The total number of rungs in all programs, excluding the Start Label/End Label must be less than 5000.

■ Logic Display

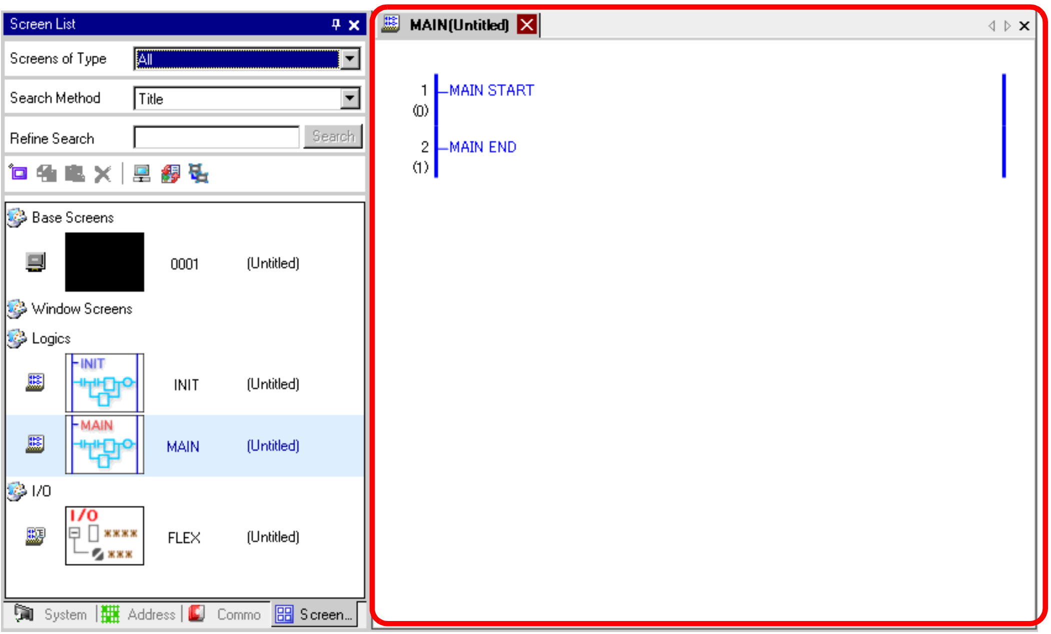
1 Click the [Screen List] tab to open the [Screen List] Window.



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 (G)].

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

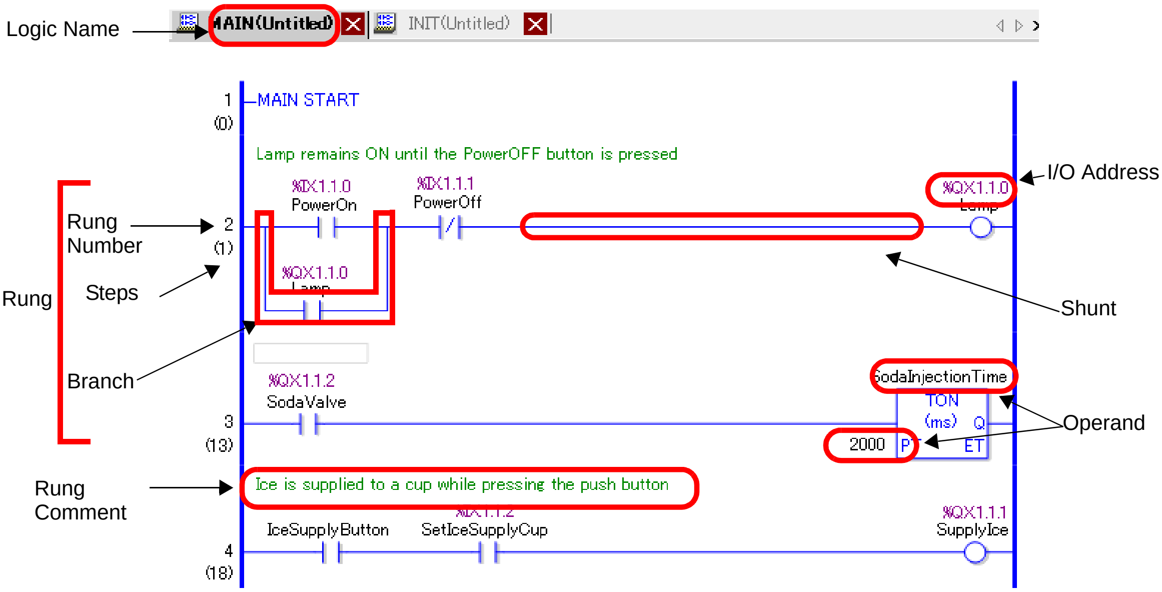


NOTE

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

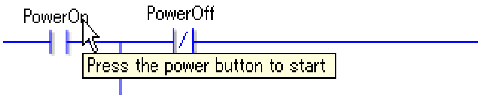
29.2.3 Logic Screens

The following explains the names of the basic elements in the Logic.



Lamp remains ON until the PowerOFF button is pressed

Ice is supplied to a cup while pressing the push button

Item	Description
Logic Name	The name of the logic screen is displayed. Click the tab to switch screens.
Rung	Consists of zero or more instructions or one label. The maximum number of instructions per rung is 99; the maximum number of branches is 25.
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	This indicates the horizontal shaft connecting the left power bar to the right power bar.(connect line)
Branch	Executes the logic program by connecting to rungs in parallel. ☞ “29.4.2 Inserting and Deleting Branches” (page 29-45)
Operand	Indicates the constants allocated to the instructions. ☞ “29.6.1 Operand Settings” (page 29-58)
I/O Address	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	Displayed when a rung has a comment. ☞ “29.7.2 Adding Rung Comments” (page 29-67)
Variable Comment	Displayed as a tool tip when the pointer points to a variable with a comment. ☞ “29.7.3 Symbol Variable Comments” (page 29-69)  Press the power button to start

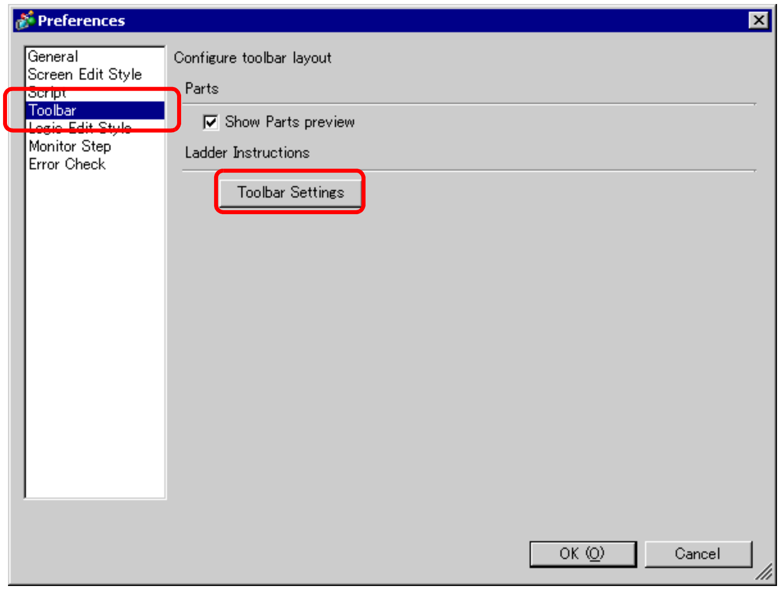
29.2.4 Customizing the Toolbar

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

■ Setup Procedures

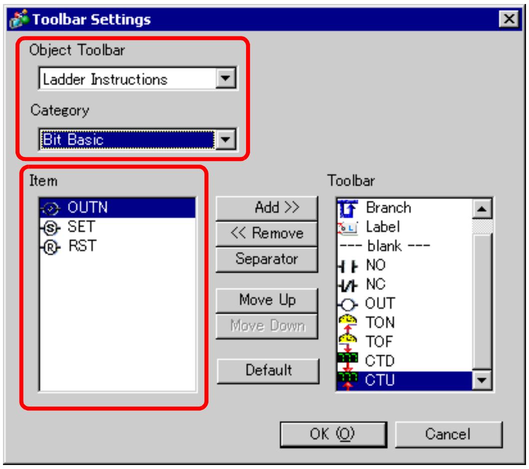
1 On the [View (V)] menu, click [Preferences O)]. The [Preferences] dialog box appears.

2 Select [Toolbar], then click [Toolbar Settings]. The [Toolbar Settings] dialog box appears.



☑ Show Parts preview

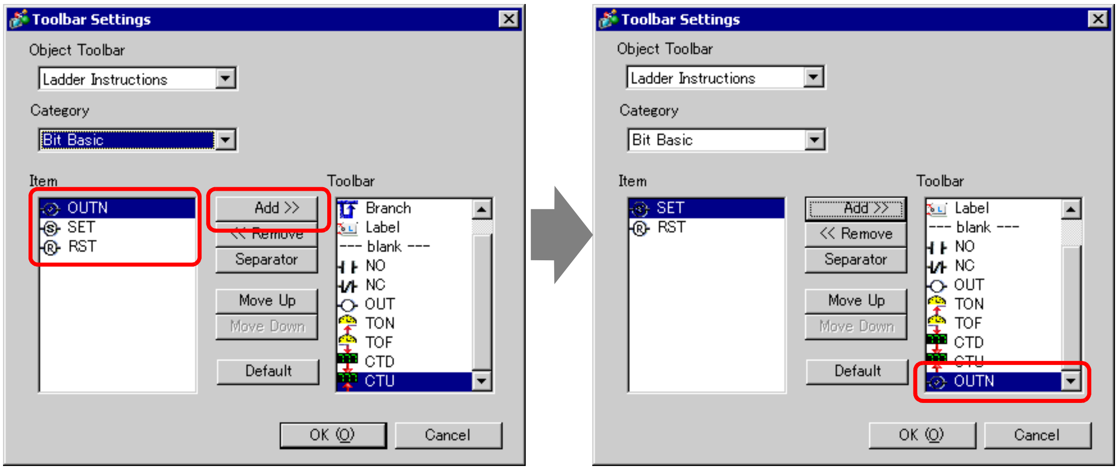
3 In [Object Toolbar], select [Ladder Instructions]. Select the category of the command you want to place on the toolbar. Icons in the selected category appear in [Item].



Object Toolbar: Ladder Instructions

Category: Bit Basic

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



Toolbar Settings

Object Toolbar: Ladder Instructions

Category: Bit Basic

5 Click [OK] to close the [Toolbar Settings] dialog box, then click [OK] to close the [Preferences] 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 Ladder Instructions" (page 30-1)

Category		Feature	Command	Icon
Basic Instructions	Bit Basics	Normally Open	NO	-| |-
		Normally Closed	NC	-|/|-
		Out	OUT	-( )-
		Negative Out	OUTN	-(/)-
		Set	SET	-(S)-
		Reset	RST	-(R)-
	Pulse Basic	Positive Transition	PT	-|↑|-
		Negative Transition	NT	-|↓|-
Basic Instructions	Program Control	Jump	JMP	J L ↷
		Jump to Subroutine	JSR	M S ↷
		Return	RET	S M ↷
		For (Repeat number of times)	FOR	FOR ↓↓↓
		Next	NEXT	↑↑↑ NEXT
		Inverse	INV	⟳
		Exit	EXIT	🚪
		Power Bar Control	PBC	⇄
		Power Bar Reset	PBR	⇄
		Logic Wait Instruction	LWA	✋

Continued

Category		Feature	Command	Icon
Operation Instruction	Arithmetic Operation	Add	ADD	+
		Subtract	SUB	−
		Multiplication	MUL	×
		Division	DIV	/
		Modulation	MOD	%
		Increment	INC	+1
		Decrement	DEC	−1
	Time Operation	Time Addition	JADD	+
		Time Subtraction	JSUB	−
	Logical Operation	Logical AND	AND	AND
		Logical OR	OR	OR
		Logical XOR	XOR	XOR
		Logical NOT	NOT	NOT
	Transfer	Move (Copy)	MOV	→
		Block Move (Block Copy)	BLMV	
		Fill Move	FLMV	→
		Exchange	XCH	⇄
	Shift	Shift Left	SHL	
		Shift Right	SHR	
		Arithmetic Shift Left	SAL	
		Arithmetic Shift Right	SAR	
Operation Instruction	Rotation	Rotate Left	ROL	
		Rotate Right	ROR	
		Rotate Left with Carry Over	RCL	
		Rotate Right with Carry Over	RCR	

Continued

Category		Feature	Command	Icon
Compare Instruction	Arithmetic Compare	Equal	EQ	=
		Greater Than	GT	>
		Greater than Or Equal To	GE	>=
		Less Than	LT	<
		Less Than Or Equal To	LE	<=
		Not Equal	NE	≠
	Time Compare	Time Compare Equal	JEQ	H:M =
		Time Compare Greater Than	JGT	H:M >
		Time Compare Greater Than Or Equal To	JGE	H:M >=
		Time Compare Less Than	JLT	H:M <
		Time Compare Less Than Or Equal To	JLE	H:M <=
		Time Compare Not Equal	JNE	H:M ≠
	Date Compare	Date Compare Equal	NEQ	Y:M =
		Date Compare Greater Than	NGT	Y:M >
		Date Compare Greater Than Or Equal To	NGE	Y:M >=
		Date Compare Less Than	NLT	Y:M <
		Date Compare Less Than Or Equal To	NLE	Y:M <=
		Date Compare Not Equal	NNE	Y:M ≠
Timer Instruction	-	On Delay Timer	TON	↑
		Off Delay Timer	TOF	↓
		Pulse Timer	TP	↑↑
		Duration On Delay Timer	TONA	+↑
		Duration 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 ↓ BCD
		BIN Convert	BIN	BCD ↓ Bin
		Encode	ENCO	001 ↓ Bin
		Decode	DECO	Bin ↓ 001
		Convert to Radian	RAD	DEG ↓ RAD
		Convert Degree	DEG	RAD ↓ DEG
		Scale	SCL	
	Type Convert	Convert Integer -> Float	I2F	INT ↓ FLOAT
		Convert Integer -> Real	I2R	INT ↓ REAL
		Convert Float -> Integer	F2I	FLOAT ↓ INT
		Convert Float -> Real	F2R	FLOAT ↓ REAL
		Convert Real -> Integer	R2I	REAL ↓ INT
		Convert Real -> Float	R2F	REAL ↓ FLOAT
		Convert to Seconds	H2S	Y:M ↓ H:M
		Convert Seconds to Time	S2H	H:M ↓ Y:M
Function Instruction	Calculation Function	Sum	SUM	
		Average	AVE	
		Square Root	SQRT	
		Bit Count	BCNT	
		PID	PID	
	Trigonometric Function	Sine	SIN	SIN
		Cosine	COS	COS
		Tangent	TAN	TAN
		Arc Sine	ASIN	SIN -1
		Arc Cosine	ACOS	COS -1
		Arc Tangent	ATAN	TAN -1
		Cotangent	COT	TAN 1
	Other Function	Exponential	EXP	EXP
		Logarithm	LN	LN
		Log Base 10	LG10	LG10

Continued

Category		Feature	Command	Icon
R/W Instruction	Time Read/ Write	Read Time	JRD	H:M
		Set Time	JSET	H:M
	Date Read/ Write	Read Date	NRD	Y:M
		Set Date	NSET	Y:M

29.3 Registering Addresses

29.3.1 Usable Addresses

On the GP-Pro EX, you can use (Device Address) of the connection device/PLC and the addresses of the GP data storage area.
These addresses can be used in two ways, as follows.
For a device address (Device Address), use the original addresses of device/PLC or GP, e.g. [PLC1]X00100 or [#INTERNAL]LS0100.
For a symbol variable, you can assign a name to the device/PLC or GP address, such as "sales_quantity" or "stock_quantity".

■ Device Address

◆ 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.
  For example, [PLC1]X00100

NOTE ☞ "29.3.4 Using External Addresses" (page 29-35)

◆ 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 29-36)

- LS Area
  This contains free user areas and an area for operating the GP.
  You can use this area only when communication with the device/PLC is established through a direct access system.
  For example, .[#INTERNAL]LS0100

- User Area
  You may use all areas as you like, up to 30,000 Words.
  For example, [#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.
  For example, [#MEMLINK]0100

■ Symbol Variable

There are two kinds of Symbol Variables.

- Symbol
  Names applied to device addresses are called symbols.
  You can manage all addresses using names, and even use these names when defining the address in parts and other objects.
  Allocatable device address: Bit address and Word address

- Variable
  Items allocated to the GP internal area are called "variables."
  There are two ways to register variables as follows.

  Variable Format: Allows you to name each variable.

NOTE

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

Address format: Uses the device address as the name. Use this format when there are too many addresses to name.

NOTE

☞ "29.3.3 Using Symbol Variables with Fixed Addresses (Address Format)" (page 29-30)

■ 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 29-38)

☞ "A.6 System Variables" (page A-84)

■ Methods of Registering Variables

Before you create a logic program, you should define all the addresses that will be used in the logic program.
There are two methods of registering variables.

◆ Variable Format

Using this method, you can register symbol variables that have descriptive names associated with a device address. The symbol variable can save you time if you ever have to change the device address. Just make the device address change in one location, not in every place it's used.

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 creating a new program. You cannot change the format after you create a program.

◆ Address Format

This allows you to use only an already selected address. It does not allow you to register variables using other names of your choice to delete or change them.

NOTE

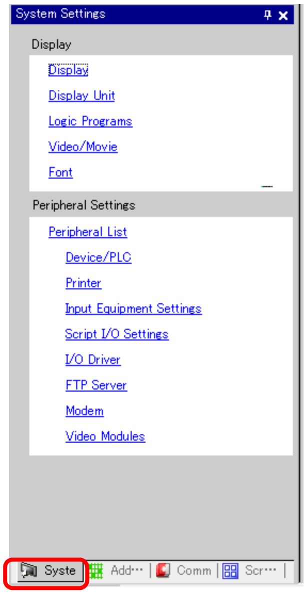
- You can change the method from [Address Format] to [Variable Format] even during logic programming. However you cannot change back from [Variable Format] to [Address Format].
- Even if the number of addresses in the logic program exceeds the number of variables, you can change the format from [Address Format] to [Variable Format], which causes addresses in the logic program to change to variables. Addresses available to the logic program but unused do not change into variables.

■ Choosing the Method of Registering Variables

NOTE

- Please refer to the settings guide for details.
  ☞ "29.14 Settings Guide" (page 29-136)

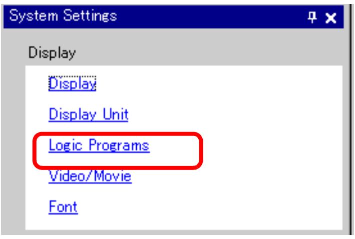
1 Select the [System Settings] tab to display [System Settings].



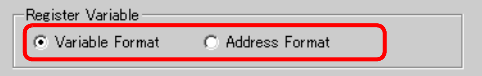
NOTE

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

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



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


Register Variable

● Variable Format ○ Address Format

■ Retentive Settings

Retentive Settings allow you to specify the variable points to keep or clear between sessions. Please note that [Variable Format] and [Address Format] have different functions.

◆ Variable Format

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

◆ Address Format

For each address, specify the retentive/volatile range. With this setting, all the addresses within the keep area will be kept, and the addresses within the clear area will not be kept. Please note that all the variables allocated to I/O (X, Y, I, Q) will be set to volatile. You cannot change the settings to retain variables between sessions. Also, you can only choose the retentive setting for PID Variables (U).

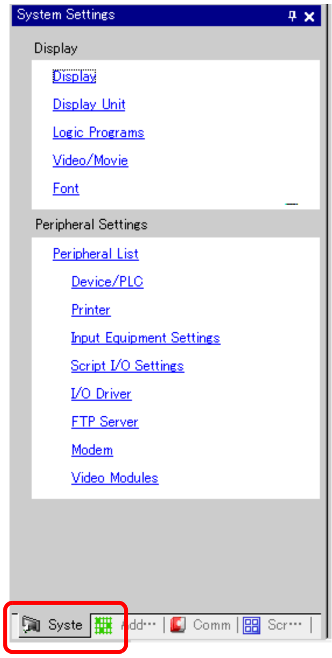
◆ Default Settings for retentive/volatile Numbers

Symbol Variable	Default Settings		For Address Format
	Retentive	Volatile	
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

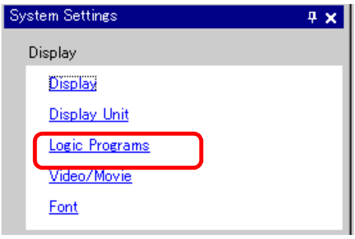
NOTE
- Please refer to the settings guide for details.
  ☞ "29.14 Settings Guide" (page 29-136)

1 Select the [System Settings] tab to display [System Settings].

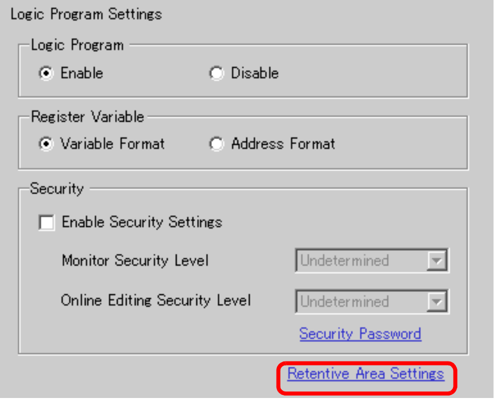


NOTE
- If the [System Settings] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings (S)].

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



3 Click [Retentive Settings] to open the [Retentive Settings] dialog box.



Logic Program Settings

Logic Program: ● Enable ○ Disable

Register Variable: ● Variable Format ○ Address Format

Security

□ Enable Security Settings

Monitor Security Level: Undetermined

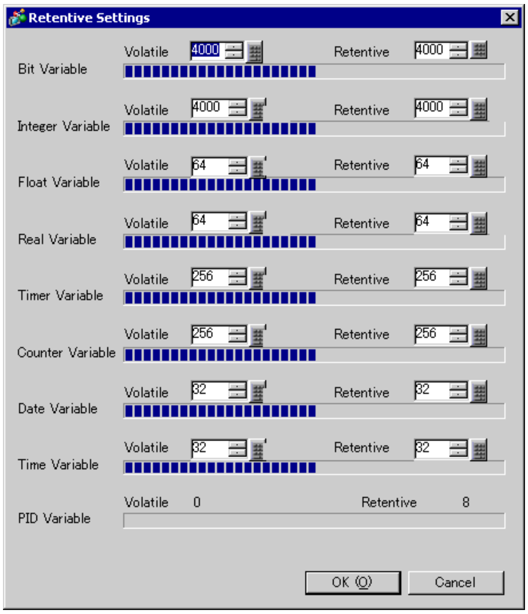
Online Editing Security Level: Undetermined

Security Password

Retentive Area Settings

4 Specify the points for each symbol variable.

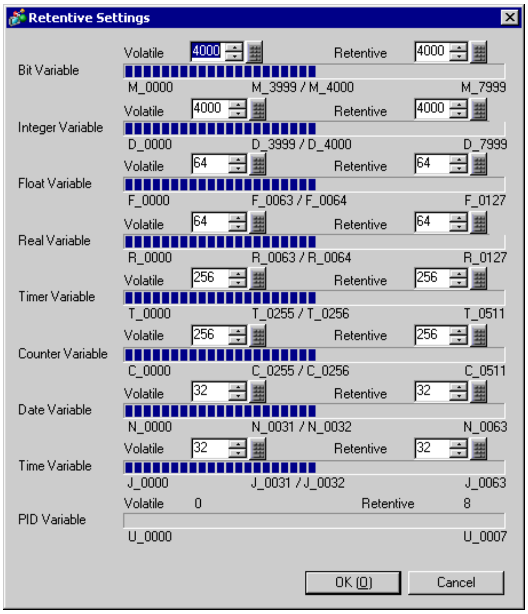
Variable Format



Retentive Settings

	Volatile	Retentive
Bit Variable	4000	4000
Integer Variable	4000	4000
Float Variable	64	64
Real Variable	64	64
Timer Variable	256	256
Counter Variable	256	256
Date Variable	32	32
Time Variable	32	32
PID Variable	0	8

Address Format



Retentive Settings

	Volatile	Retentive			
Bit Variable	4000	4000	M_0000	M_3999 / M_4000	M_7999
Integer Variable	4000	4000	D_0000	D_3999 / D_4000	D_7999
Float Variable	64	64	F_0000	F_0063 / F_0064	F_0127
Real Variable	64	64	R_0000	R_0063 / R_0064	R_0127
Timer Variable	256	256	T_0000	T_0255 / T_0256	T_0511
Counter Variable	256	256	C_0000	C_0255 / C_0256	C_0511
Date Variable	32	32	N_0000	N_0031 / N_0032	N_0063
Time Variable	32	32	J_0000	J_0031 / J_0032	J_0063
PID Variable	0	8	U_0000		U_0007

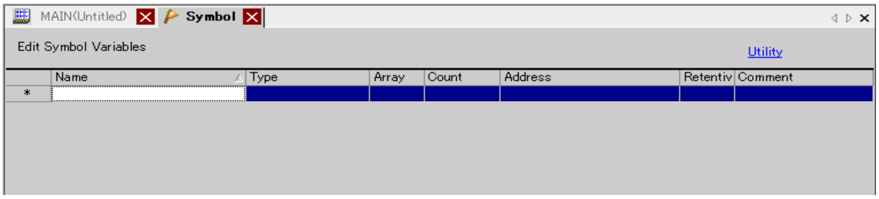
29.3.2 Using Symbol Variables with Arbitrary Names (Variable Format)

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.

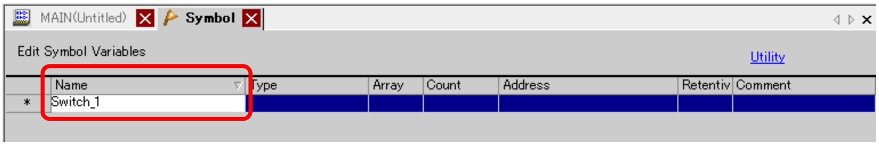
■ Registering Symbol Variables

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



	Name	Type	Array	Count	Address	Retentiv	Comment
*							

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

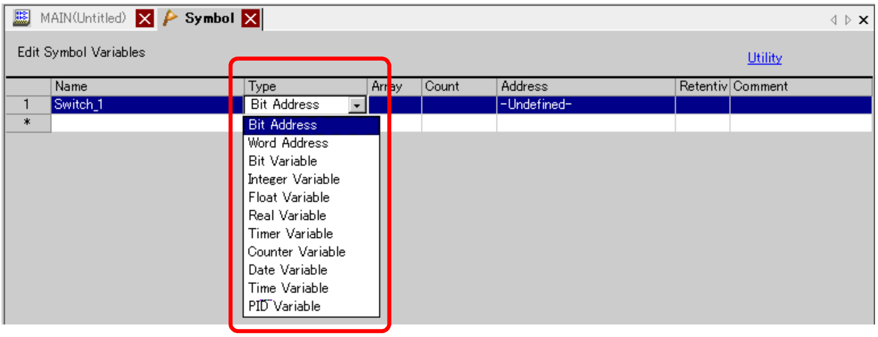


	Name	Type	Array	Count	Address	Retentiv	Comment
*	Switch_1						

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.

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



	Name	Type	Array	Count	Address	Retentiv	Comment
1	Switch_1	Bit Address			-Undefined-		
*							

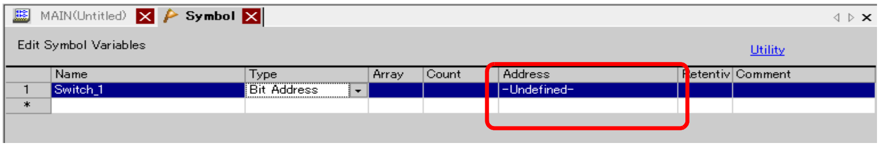
- If you selected [Bit Address] or [Word Address], you can specify the address. Proceed to step 4.

- 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, proceed to step 6.
- If you selected [Timer Variable], [Counter Variable], [Time Variable], or [Date Variable], proceed to step 6.
- If you selected [PID Variable], the [Retentive] check box must always be selected. Proceed to step 7.

NOTE

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

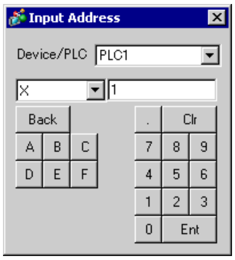
4 Specify the address on the [Address] column. Proceed to step 8.


MAIN(Untitled) Symbol

Edit Symbol Variables   Utility

	Name	Type	Array	Count	Address	Retentiv	Comment
1	Switch_1	Bit Address			-Undefined-		
*							

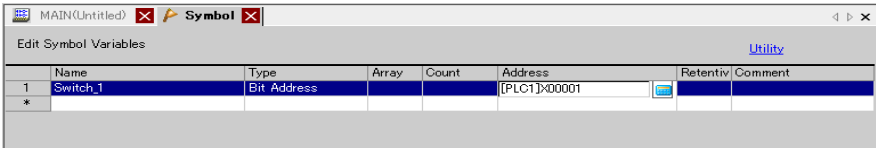
Click [-Undefined-] to display a keyboard for inputting the address.


Input Address

Device/PLC: PLC1

X 1

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

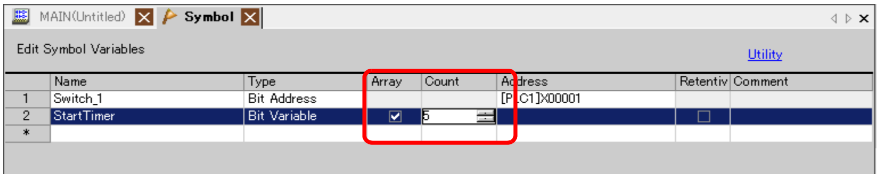

MAIN(Untitled) Symbol

Edit Symbol Variables   Utility

	Name	Type	Array	Count	Address	Retentiv	Comment
1	Switch_1	Bit Address			[PLC1]X00001		
*							

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. Proceed to step 6.


MAIN(Untitled) Symbol

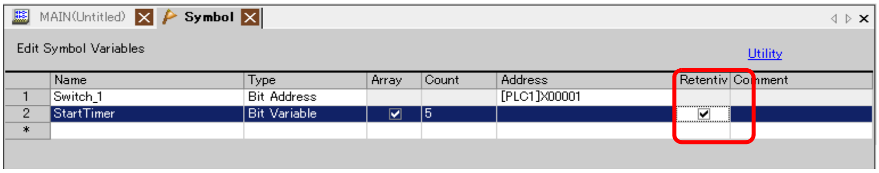
Edit Symbol Variables   Utility

	Name	Type	Array	Count	Address	Retentiv	Comment
1	Switch_1	Bit Address			[PLC1]X00001		
2	StartTimer	Bit Variable	☑	5		□	
*							

NOTE

- For details on the arrays, refer to the following.
  ☞ “ ■ Arrays and Array Sizes” (page 29-25)

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



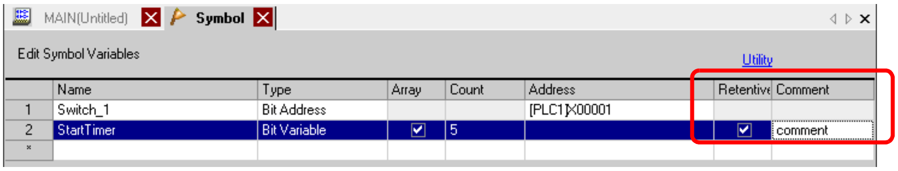
Edit Symbol Variables

	Name	Type	Array	Count	Address	Retentiv	Comment
1	Switch_1	Bit Address			[PLC1]X00001		
2	StartTimer	Bit Variable	☑	5		☑	
*							

NOTE

- For details on the retentive/volatile settings, refer to the following.
  ☞ " ■ Retentive" (page 29-25)

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



Edit Symbol Variables

	Name	Type	Array	Count	Address	Retentive	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 29-69)

8 The registration is complete.

NOTE

- You can make changes to registered variables only when they are used in the logic program. You can delete variables that are not used on any screen.
  To delete variables, select the symbol variable and either click ✕ or press the [Delete] key.
- 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 5-49)

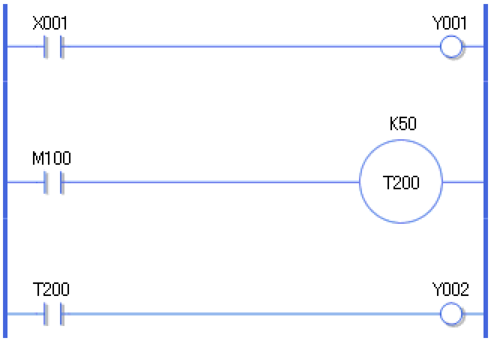
■ Variable name

On the GP-Pro EX, you can name variables and use them in the logic program. For most PLCs, data storage areas are handled as device addresses in registers named by the PLC manufacturer.

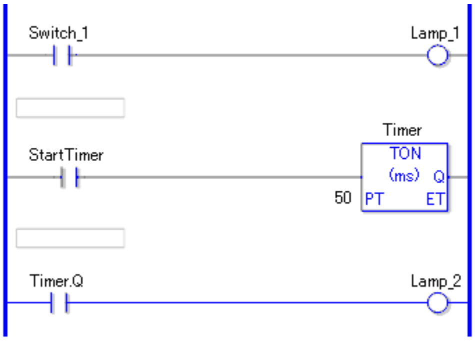
For example,

	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

For models by other manufacturers



For the GP-Pro EX



The above program description is an image.

■ Variable Type

There are nine variable types: Bit, Integer, Float, Real, Timer, Counter, Time, Date, and PID.

◆ Bit Variable

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

◆ Integer Variable

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

◆ Float Variable

Variable with a 32-bit length that has a floating point value of ±1.175494351e-38 - ±3.402823466e+38 and 0. You may use up to 7 decimal places.

◆ Real Variable

Variable with a 64-bit length that has a floating point value of ±2.2250738585072014e-308 - ±1.7976931348623158e+308 and 0. You may use up to 15 decimal places.

◆ Timer Variable

Use timer variables to enable timer instructions.
Timer variables consist of the following five special variables.
For details on ladder instructions, refer to “Chapter 30 Ladder Instructions” (page 30-1) .

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

- Even if the clear setting has been selected for timer variables, the special variable timer PT (value) is kept.
  ☞ “ ■ Retentive” (page 29-25)

◆ Counter Variable

Use counter variables to enable counter instructions.
Counter variables consist of the following seven special variables.
For details on ladder instructions, refer to “Chapter 30 Ladder Instructions” (page 30-1) .

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.
- Even if the clear setting has been selected for counter variables, the special variable counter PV (value) is kept.
  ☞ “ ■ Retentive” (page 29-25)

◆ Date/Time Variable

Use date/time variables to enable date/time instructions.

Date/time variables consist of the following three special variables.

For details on ladder instructions, refer to “Chapter 30 Ladder Instructions” (page 30-1) .

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

Use time variables to enable time instructions.

The time variable consists of the following three special variables.

For details on ladder instructions, refer to “Chapter 30 Ladder Instructions” (page 30-1) .

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

Use PID variables to enable PID instructions.

PID variables consist of the following eleven special variables.

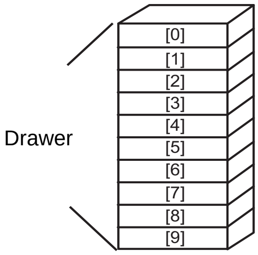
For details on ladder instructions, refer to “Chapter 30 Ladder Instructions” (page 30-1) .

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 Deadband 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 Deadband Range Flag	Bit
IF	Integral Range Processing Flag	Bit

■ Arrays and Array Sizes

You can specify arrays for bit, integer, float, and real variables.
You can set up a maximum 4,096 array elements for bit and integer variables. You can set up a maximum 128 array elements for float and real variables.

Arrays provide you with a method to set up multiple elements with the same data type, in a single variable, all at once.



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.

■ Retentive

If variables are set to retentive, they are stored in backup SRAM and retain their values when the unit is shut down.
These values are kept until the backup battery runs out, which causes these variables to revert to their default value as defined in GP-Pro EX. When shutting down or resetting the GP, the latest values are copied to SRAM. Downloading a logic program will initialize variables with their default values set up in GP-Pro EX, unless you select the Retentive Transfer check box.

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

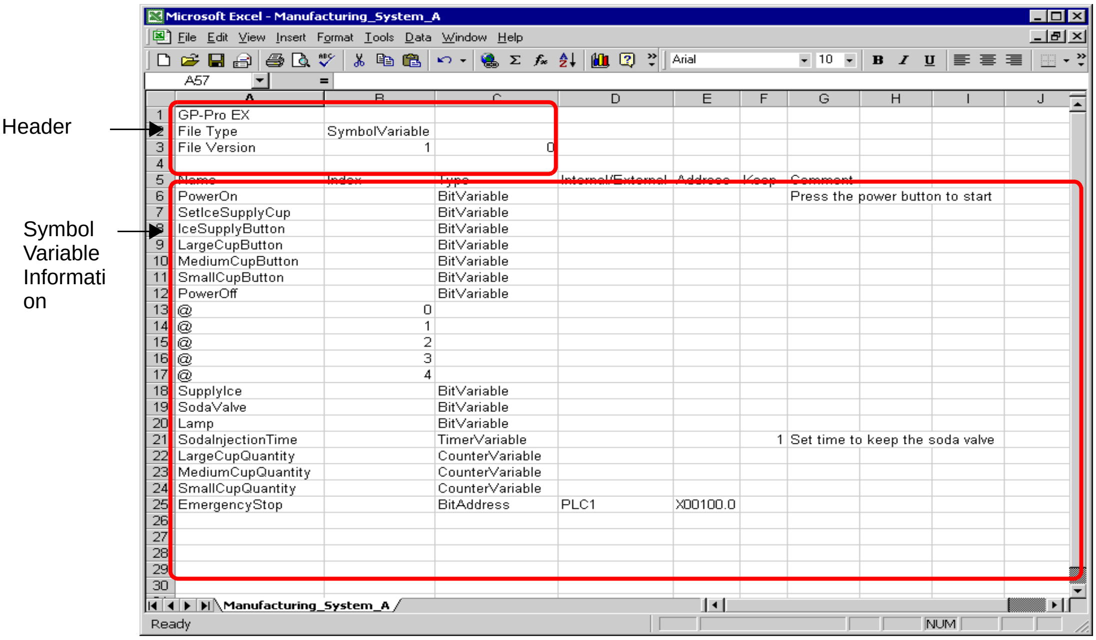
- Retentive Settings allow you to specify the variable points to retentive/ volatile.
  ☞ “ ■ Retentive Settings” (page 29-16)

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


Header
Symbol Variable Information
Microsoft Excel - Manufacturing_System_A
File Edit View Insert Format Tools Data Window Help
Arial
10
A57
=
Manufacturing_System_A
Ready
NUM

	A	B	C	D	E	F	G
1	GP-Pro EX						
2	File Type	SymbolVariable					
3	File Version	1	0				
4							
5	Name	Index	Type	Internal/External	Address	Keep	Comment
6	PowerOn		BitVariable				Press the power button to start
7	SetIceSupplyCup		BitVariable				
8	IceSupplyButton		BitVariable				
9	LargeCupButton		BitVariable				
10	MediumCupButton		BitVariable				
11	SmallCupButton		BitVariable				
12	PowerOff		BitVariable				
13	@	0					
14	@	1					
15	@	2					
16	@	3					
17	@	4					
18	SupplyIce		BitVariable				
19	SodaValve		BitVariable				
20	Lamp		BitVariable				
21	SodaInjectionTime		TimerVariable			1	Set time to keep the soda valve
22	LargeCupQuantity		CounterVariable				
23	MediumCupQuantity		CounterVariable				
24	SmallCupQuantity		CounterVariable				
25	EmergencyStop		BitAddress	PLC1	X00100.0		

- 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: Symbol Variable (Please do not edit.)
  File Version: The file version is saved. (Please do not edit.)

NOTE
- When creating new symbol variables in a CSV file, use the above formats, including 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 29-25)

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.
  ☞ " ■ Variable 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 29-13)

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 29-13)

Retentive: Input the retentive/volatile settings.

Settings	Value
Retentive	1
Volatile	0

NOTE

- For the volatile setting, you can omit the "0" value.
- When [Register Variable] is specified as [Address Format], it is not used.
- For details on the retentive/volatile settings, refer to the following.
  ☞ " ■ Retentive" (page 29-25)

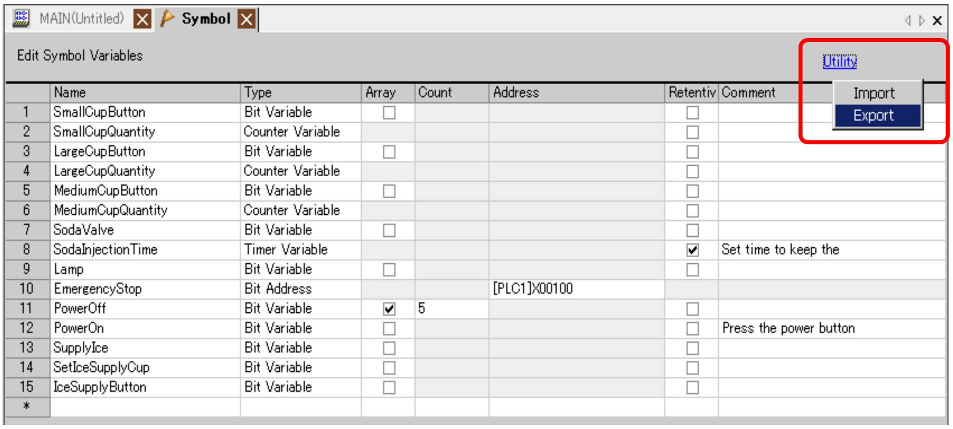
Comment: Input the comment.

NOTE

- For details on comments, refer to the following.
  ☞ "29.7 Comment Input" (page 29-65)

◆ Export Procedures

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

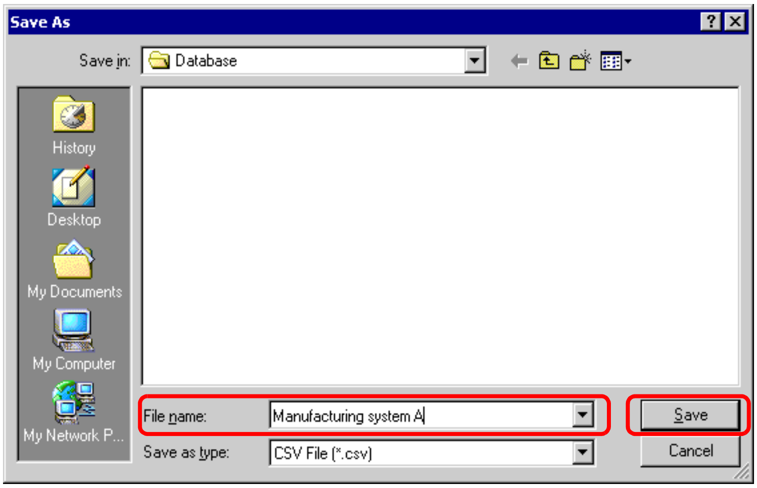


	Name	Type	Array	Count	Address	Retentiv	Comment
1	SmallCupButton	Bit Variable	□			□	
2	SmallCupQuantity	Counter Variable				□	
3	LargeCupButton	Bit Variable	□			□	
4	LargeCupQuantity	Counter Variable				□	
5	MediumCupButton	Bit Variable	□			□	
6	MediumCupQuantity	Counter Variable				□	
7	SodaValve	Bit Variable	□			□	
8	SodaInjectionTime	Timer Variable				☑	Set time to keep the
9	Lamp	Bit Variable	□			□	
10	EmergencyStop	Bit Address			[PLC1]X00100		
11	PowerOff	Bit Variable	☑	5		□	
12	PowerOn	Bit Variable	□			□	Press the power button
13	SupplyIce	Bit Variable	□			□	
14	SetIceSupplyCup	Bit Variable	□			□	
15	IceSupplyButton	Bit Variable	□			□	
*							

NOTE

- You cannot import/export system variables.

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

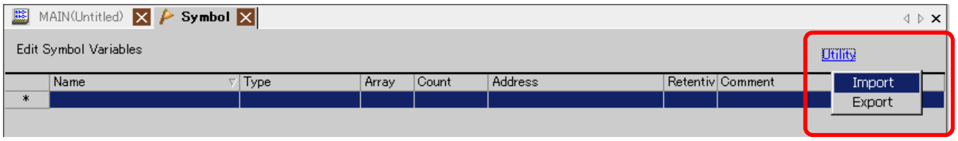


File name: Manufacturing system A

Save as type: CSV File (*.csv)

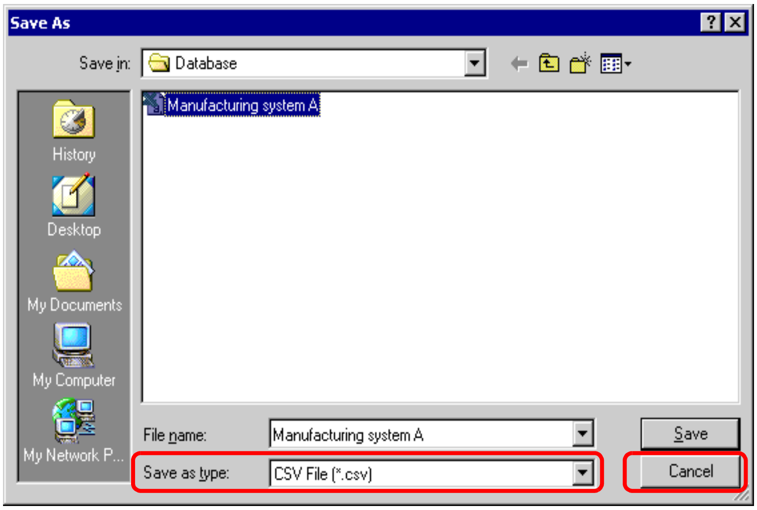
◆ Import Procedures

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



Name	Type	Array	Count	Address	Retentiv	Comment
*						

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



File name: Manufacturing system A

Save as type: CSV File (*.csv)

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)

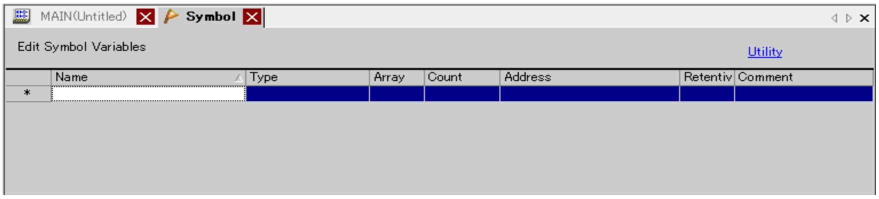
The following explains the symbol variables that you can use when [Register Variable] is set to [Address Format].

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

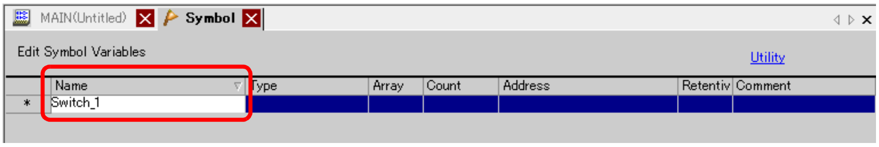
■ Registering Symbol Variables

You can name Bit Addresses and Word Addresses as you like.

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



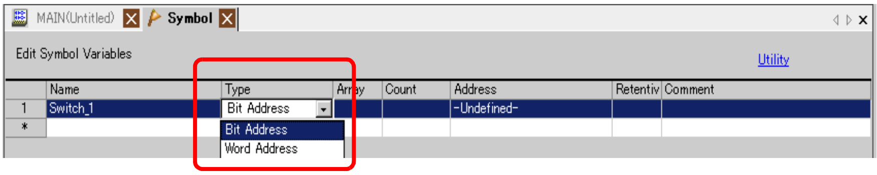
2 Double-click a cell in the [Name] column to enter a name.



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.

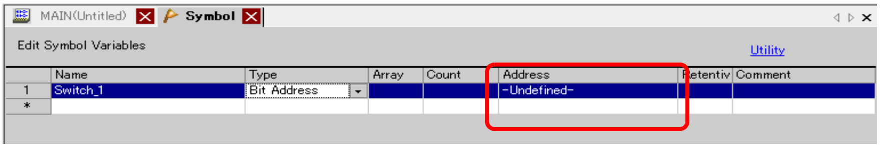
3 Click the cell in the [Type] column and select either [Bit Address] or [Word Address] type.



NOTE

- For details on the types of variables, refer to the following.
  ☞ " ■ Variable Type" (page 29-22)

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

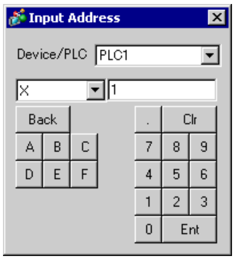


MAIN(Untitled) Symbol

Edit Symbol Variables Utility

	Name	Type	Array	Count	Address	Retentiv	Comment
1	Switch_1	Bit Address			-Undefined-		
*							

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

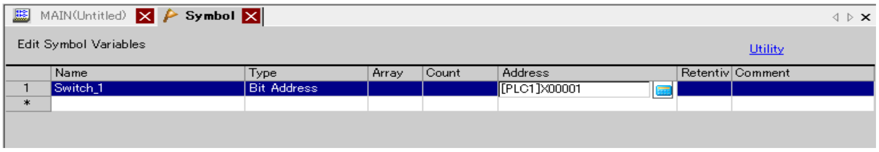


Input Address

Device/PLC PLC1

X 1

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



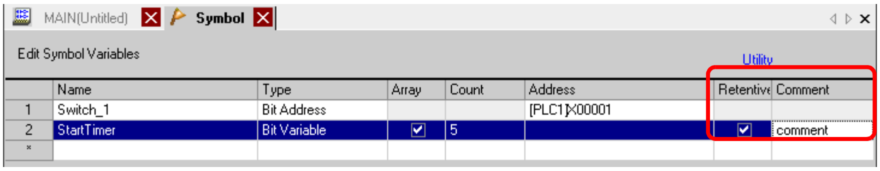
MAIN(Untitled) Symbol

Edit Symbol Variables Utility

	Name	Type	Array	Count	Address	Retentiv	Comment
1	Switch_1	Bit Address			[PLC1]X00001		
*							

The address is finalized.

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



MAIN(Untitled) Symbol

Edit Symbol Variables Utility

	Name	Type	Array	Count	Address	Retentive	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 29-69)

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 ✕ 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 5-49)

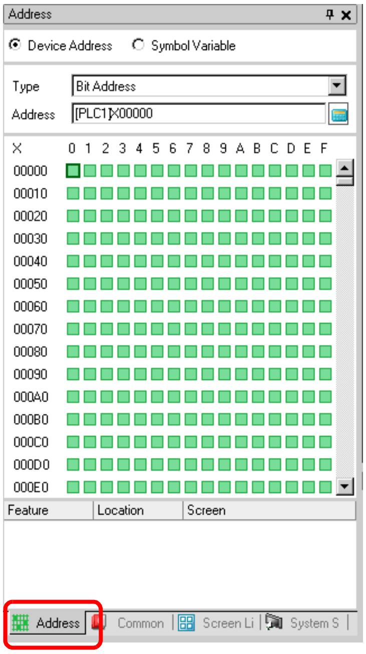
■ Logic Address Display

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.
The address is displayed as X_0100 on a logic program. For example, .[#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] window, and specify the address for the logic program instruction and the part placed on the screen.

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



Address

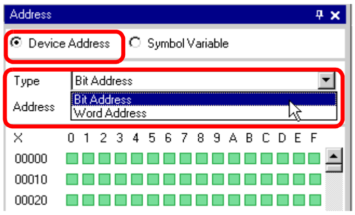
● Device Address ○ Symbol Variable

Type: Bit Address

Address: [PLC1]X00000

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

2 Select [Device Address], and in [Type], select [Bit Address] or [Word Address].



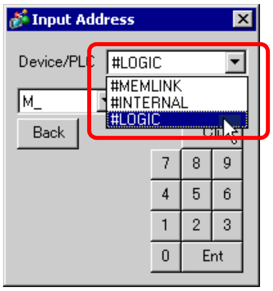
Address

● Device Address ○ Symbol Variable

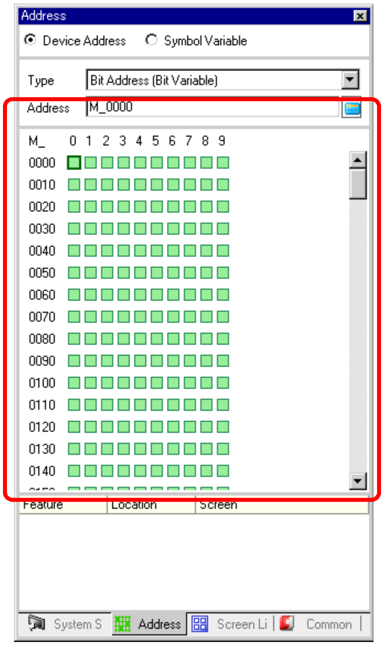
Type: Bit Address

3 Click the icon 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 29-62)

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 Using the Direct Access Method" (page A-4)

■ External Device Address

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

◆ [Symbol Variable] Window

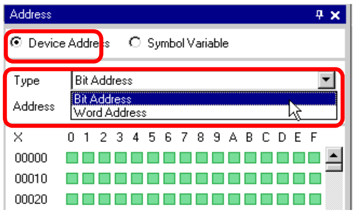
Click the cell in the [Address] column and click .

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)

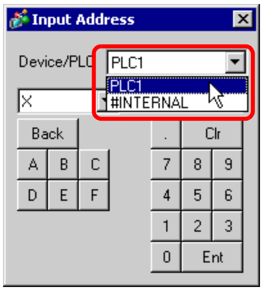
◆ [Address] Window

1 Select [Device Address], and in [Type], select [Bit Address] or [Word Address].



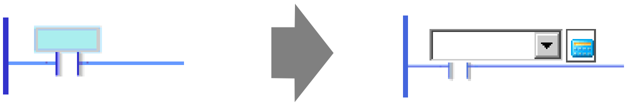
2 Click the icon to display the [Input Address] dialog box.

3 In [Device/PLC] (for example, PLC1) and input the address of the model (for example, X00000).

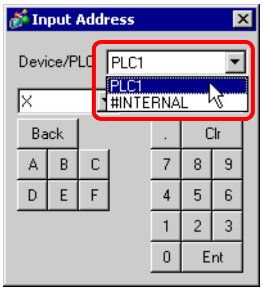


◆ Logic

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



2 In [Device/PLC] (for example, PLC1) and input the address of the model (for example, 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 Using the Direct Access Method" (page A-4)

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 Using the Memory Link Method with Unsupported Devices/PLCs" (page A-6)

■ Internal Address

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

◆ [Symbol Variable] Window

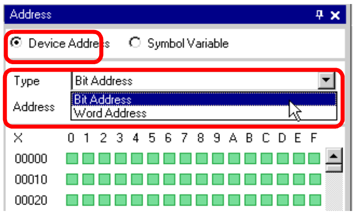
Click the cell in the [Address] column and click .

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] Window

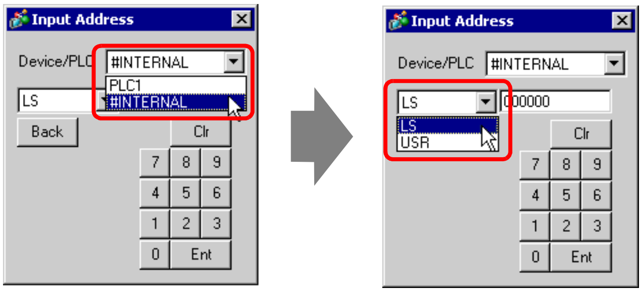
1 Select [Device Address], and in [Type], select [Bit Address] or [Word Address].



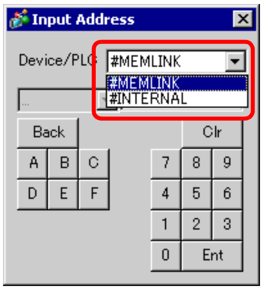
2 Click the icon to display the [Input Address] dialog box.

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 (For example, :. LS0000)

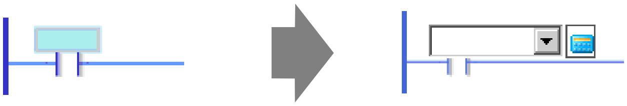


System Area for Memory Link
Connection Device Name (MEMLINK)
Address (for example, 0000)



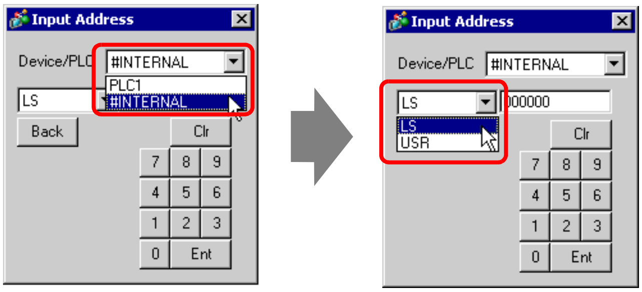
◆ Logic

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

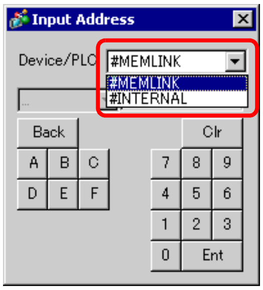


2 In [Device/PLC], select the connection device name and input the address of the model.

LS Area or User Area
Connection Device Name (INTERNAL)
Address (For example, :. LS0000)



System Area for Memory Link
Connection Device Name (MEMLINK)
Address (for example, 0000)



29.3.6 System Variables

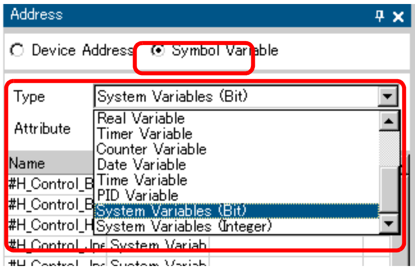
There are two kinds of system variables. One is used for logic and the other for screens. Users cannot edit the variables, for example register the names of their choice, or change or delete addresses.
Also, there are two types of system variables. One is an integer-type and the other is a bit-type.

NOTE ☞ "A.6 System Variables" (page A-84)

■ System Variable Settings

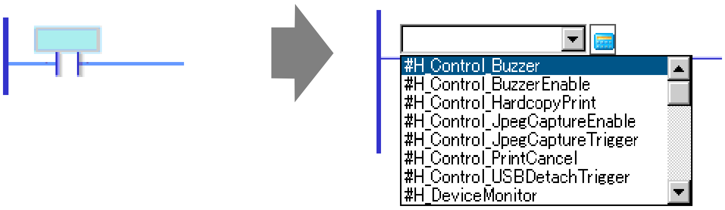
◆ [Address] Window

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 type.



◆ Logic

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



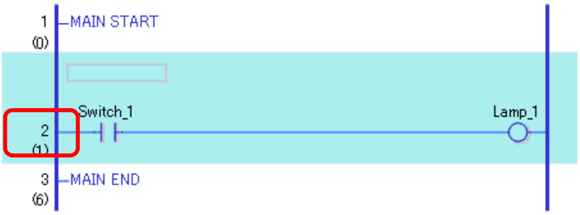
29.4 Inserting Rungs and Branching

The following explains how to edit rungs in the logic.

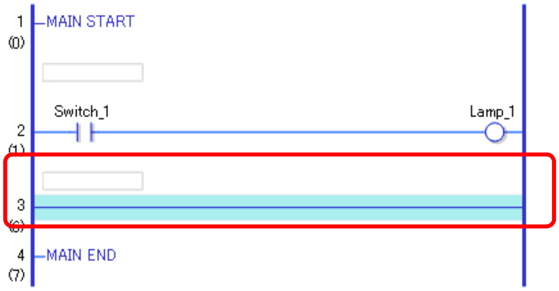
29.4.1 Editing Rungs

■ Inserting Rungs

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



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

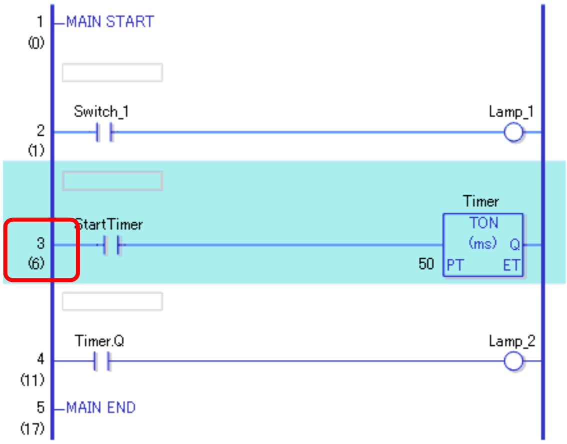


NOTE

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

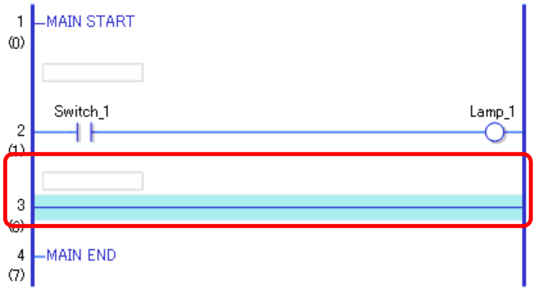
■ Delete Rung

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 29-44)

2 ✕ Click the icon.
The selected rung is deleted.

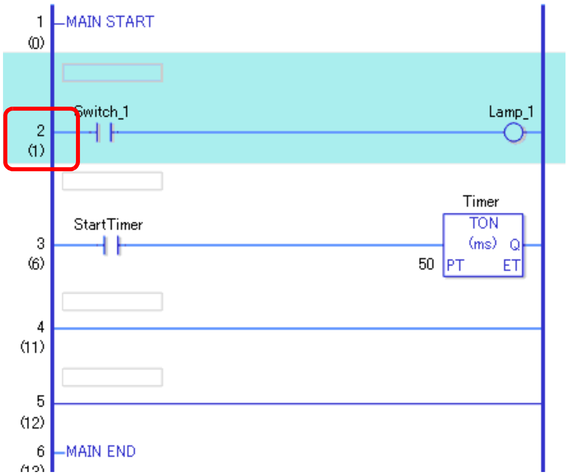


NOTE • You can also delete a rung in either of the following ways.
•Right-click and then click [Delete (D)].
•Press DELETE.

■ 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 29-44)

2 Click the icon.
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.

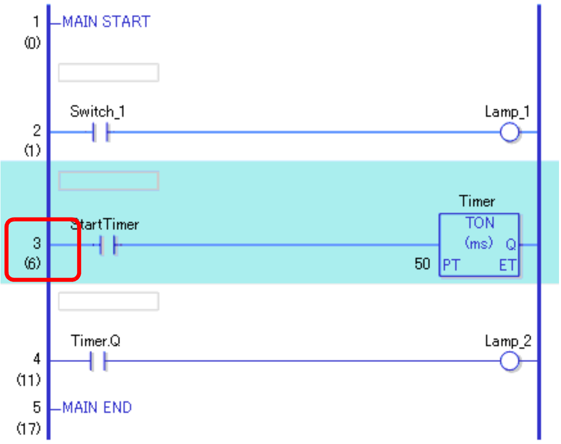
3 Paste the copied rung in the desired location.

NOTE ☞ " ■ Pasting Rungs" (page 29-43)

■ 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.

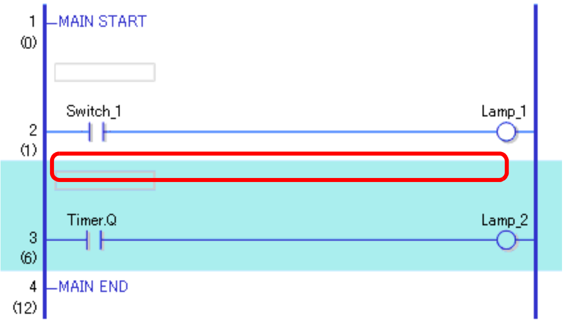


NOTE

- You can select a range to cut more than one rung.
  ☞ "■ Selecting Multiple Rungs" (page 29-44)

2 ✂ Click the icon.

The cut rungs are copied to the clipboard and the selected rungs are deleted.



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.

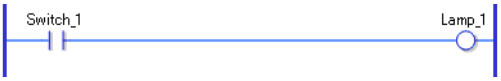
NOTE

☞ "■ Pasting Rungs" (page 29-43)

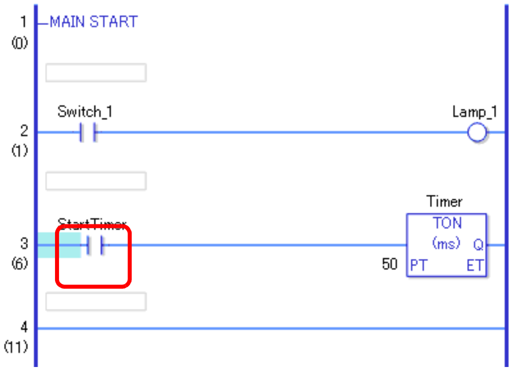
■ Pasting Rungs

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

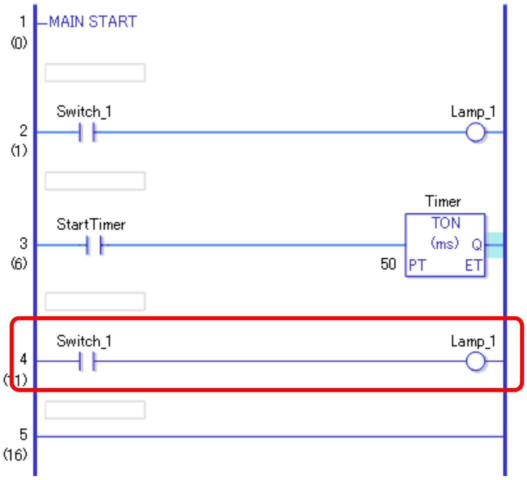


1 Select a part (power bar, instruction, etc.) immediately above where you want to insert the rung.



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

2 📋 Click the icon.
The rung is pasted below the selected rung.



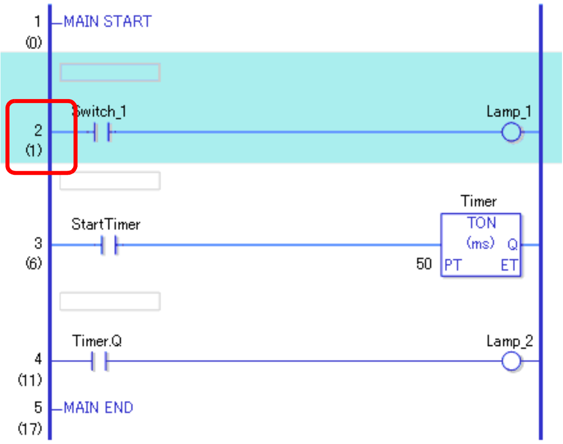
NOTE

- 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 29-58)
  ☞ “29.7 Comment Input” (page 29-65)

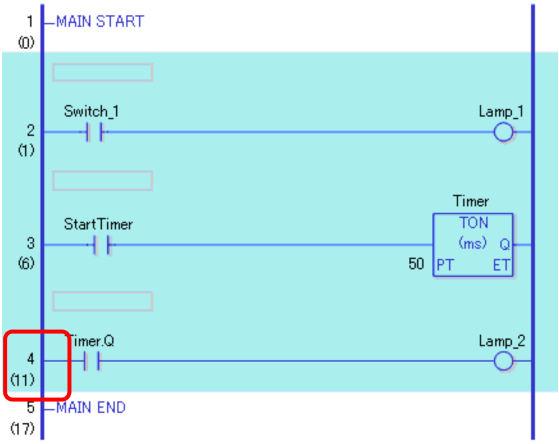
■ Selecting Multiple Rungs

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

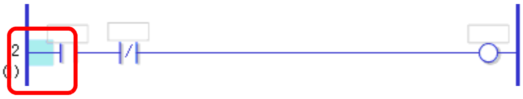
- You can also select a rung range in the following way.
  •While holding down SHIFT, press the [↑] key or the [↓] key to select the last rung number of the range that you want to select.
  •Press CTRL+A to select all rungs. Note that the first rung's start label and the last rung's end label will not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

29.4.2 Inserting and Deleting Branches

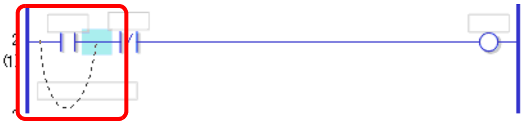
■ Inserting Branches

The following explains how to insert a branch.
Here, a branch will be inserted between a 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).



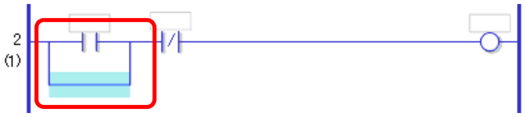
2 Click the icon. A dashed line is drawn between the start and end points of the branch.



NOTE

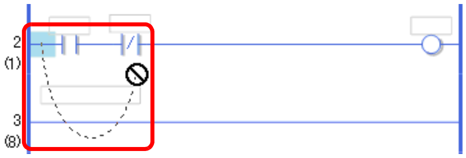
- You can also insert a branch in either of the following ways.
•Right-click and then click [Insert Branch (B)].
•Press CTRL+B.

3 Press the LEFT ARROW or RIGHT ARROW key to determine the final position, and then press ENTER. The branch will be inserted.

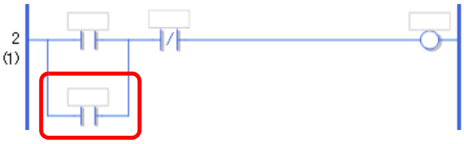


NOTE

- 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 pointer changes back from ⃠ to ↖ and the branch will be inserted.
The branch end is not valid if the pointer is shown as a ⃠. If this symbol shows, the branch will not be inserted after you release the left mouse button.



4 Insert an instruction in the branch.



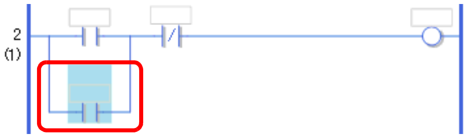
NOTE

☞ " ■ Inserting Instructions" (page 29-47)

■ Deleting Branches

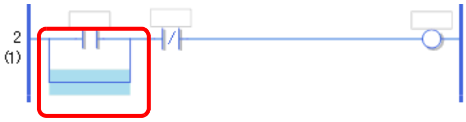
To delete branches, you must delete the instructions before deleting the branches.

1 Delete the instruction.

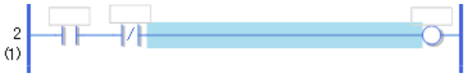


NOTE ☞ " ■ Deleting Instructions" (page 29-49)

2 Select the branch that you want to delete.



3 ✕ Click the icon.The branch is deleted.



NOTE

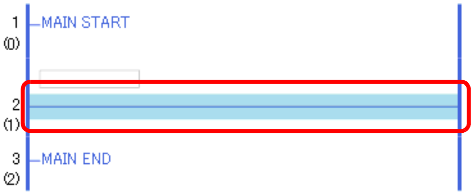
- You can also delete a branch in either of the following ways.
  •Right-click and then click [Delete (D)].
  •Press DELETE.

29.5 Inserting Instructions

29.5.1 Editing Instructions

■ Inserting Instructions

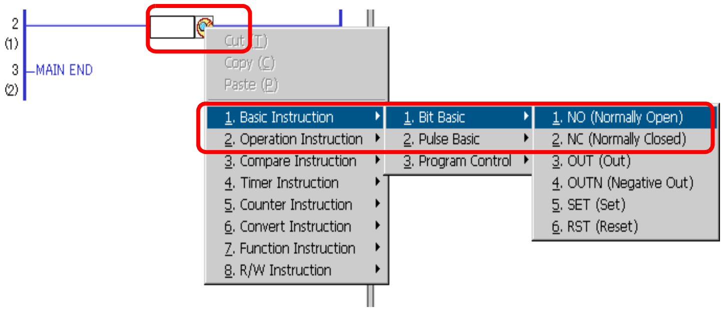
1 Select where you want to insert the instruction and, on the [Logic (L)] menu, click [Insert Instruction (I)].



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 29-6)

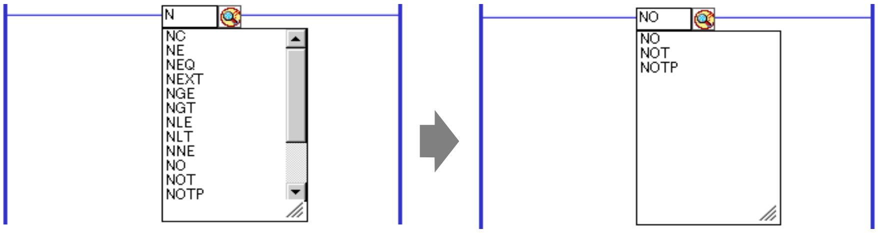
2 Click the icon to select the instruction.



NOTE

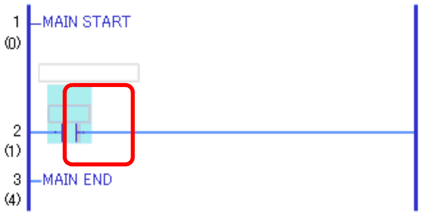
- You can also type the instruction directly into the text box. Every time you enter a character, possible instructions for the input text are displayed.

Enter "N" **Enter "O"**



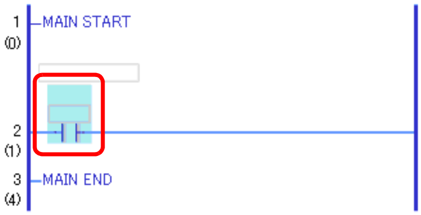
- On the [View (V)] menu, select [Preferences (O)]. The [Preferences] window opens. In the [Preferences] window, select [Logic Edit Style]. Select the [Set up operands when adding instructions] check box. The operand input box will appear when the instruction is inserted.
  ☞ “29.6.1 Operand Settings” (page 29-58)

3 The instruction is inserted.

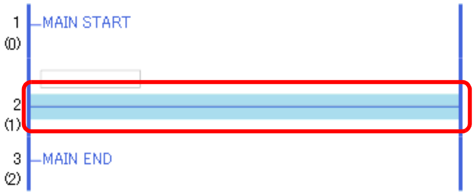


■ Deleting Instructions

1 Select the instruction that you want to delete.



2 ✕ Click the icon.The instruction is deleted.



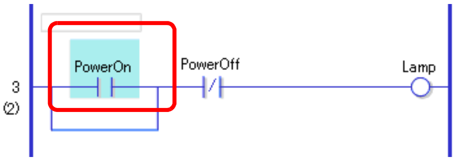
NOTE

- 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 icon.
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.

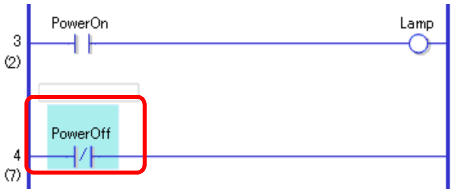
NOTE

☞ "■ Pasting Instructions" (page 29-51)

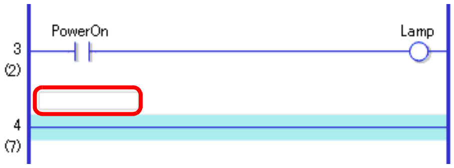
■ Cutting Instructions

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 icon.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

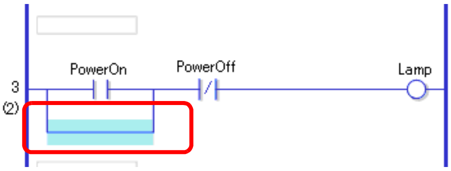
☞ “ ■ Pasting Instructions” (page 29-51)

■ 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.

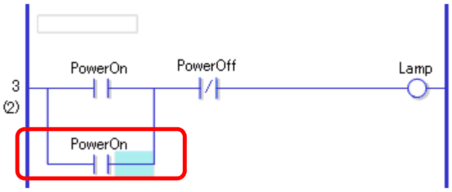
An instruction copied to the clipboard

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 icon. 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 29-58)

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 the label specified in the JMP instruction and continues to execute the logic program, without returning to the original JMP instruction.

NOTE

- For details on JSR instructions and JMP instructions, refer to the explanation for the relevant instruction.
  ☞ "Chapter 30 Ladder Instructions" (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

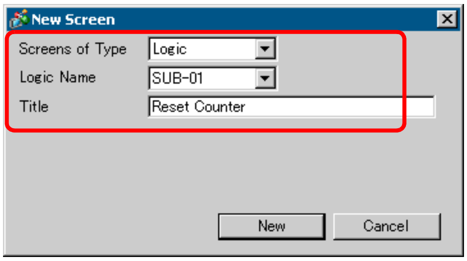
1 On the normal toolbar or the [Screen List] window, click ▣ . The [New Screen] dialog box appears.

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 (G)].

2 In [Screens of Type], select [Logic] and in [Logic Name], select the subroutine name (for example, SUB-01).

Input the title when necessary. You can input up to 30 characters.



New Screen

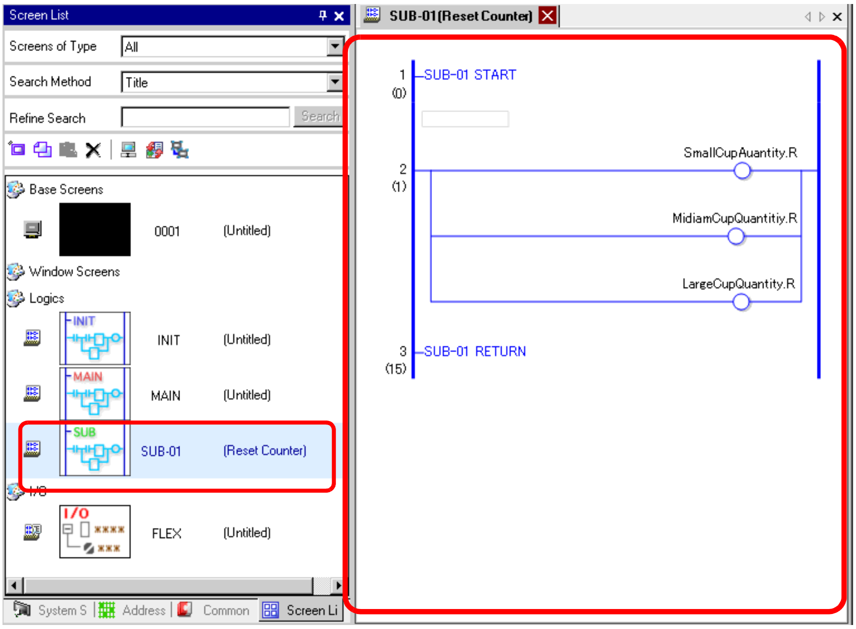
Screens of Type: Logic

Logic Name: SUB-01

Title: Reset Counter

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

4 Create the subroutine.



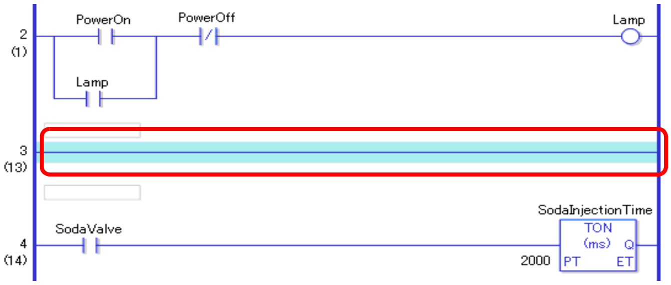
NOTE

- To add more than one subroutine to a logic program, repeat procedures 1 to 5 to create the desired number of subroutine 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.

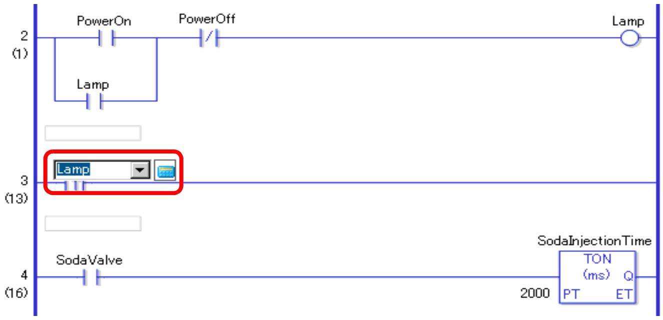
1 Select the 2nd rung to enter the rung.



NOTE

- For how to enter a rung, refer to the following.
  ☞ " ■ Inserting Rungs" (page 29-39)

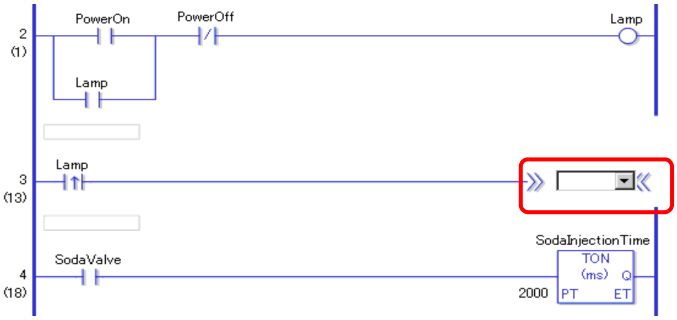
2 Insert a PT instruction in the 3rd rung (Positive Transition) and assign the symbol variable "lamp" to the PT instruction.



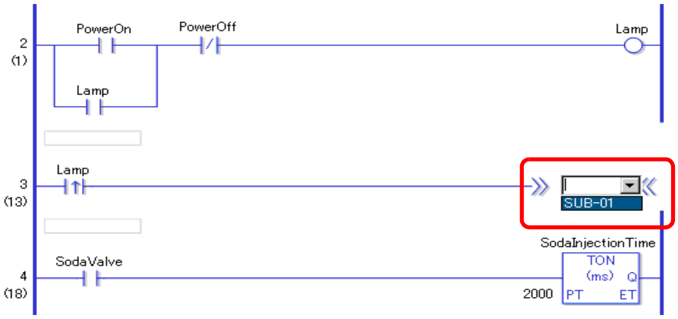
NOTE

- For how to enter an instruction, refer to the following.
  ☞ " ■ Inserting Instructions" (page 29-47)
- For the operand settings, refer to the following.
  ☞ "29.6.1 Operand Settings" (page 29-58)
- For details on an instruction, refer to the explanation of the relevant instruction.
  ☞"Chapter 30 Ladder Instructions" (page 30-1)

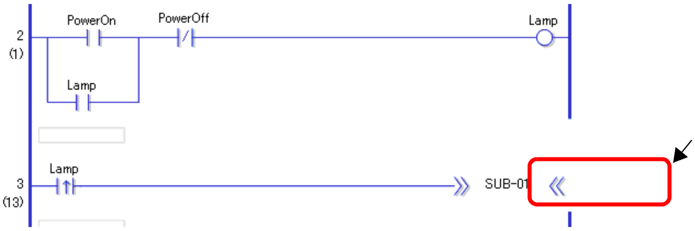
3 Insert the JSR instruction to the right of the PT instruction.



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



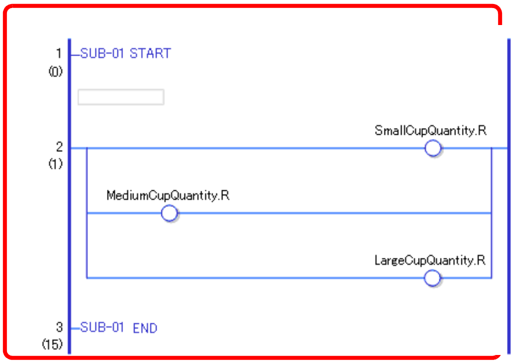
When the "Lamp" ON is detected, the operation jumps to the subroutine program [SUB-01]. After the subroutine program [SUB-01] is executed, the main logic program [MAIN] resumes in the 4th rung.



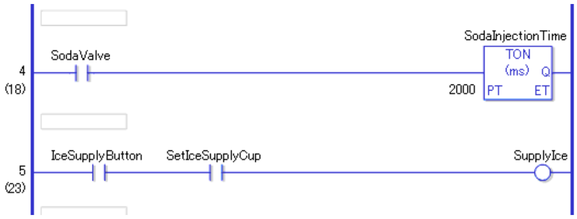
JSR Instruction

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.



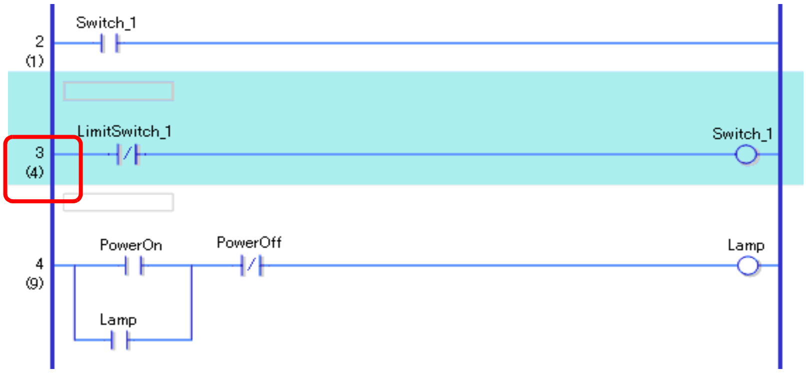
■ Inserting Labels

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.

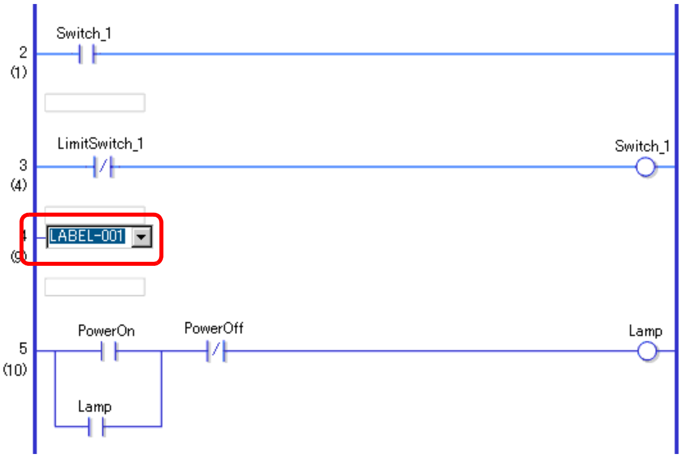
Here, for example, a [LABEL-001] label is inserted in the jump instruction, and the operation jumps to the 3rd rung upon execution of the JMP instruction when the "Switch 1" in the 2nd rung turns on.

◆ Inserting a Label

1 Select the 2nd rung.



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



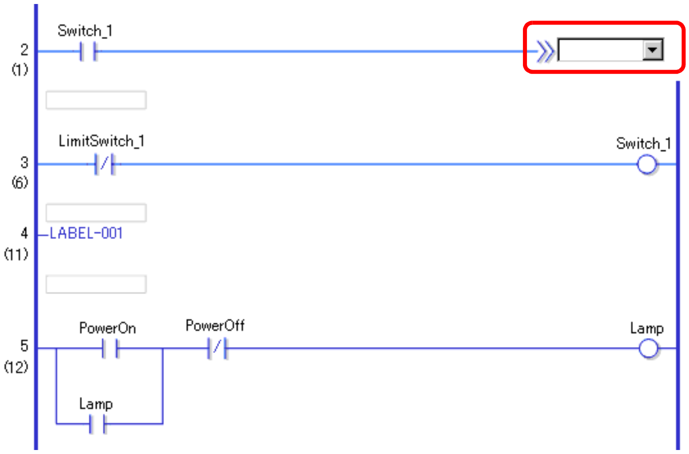
NOTE

- 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 (for example, LABEL-001).

◆ Inserting a JMP Instruction

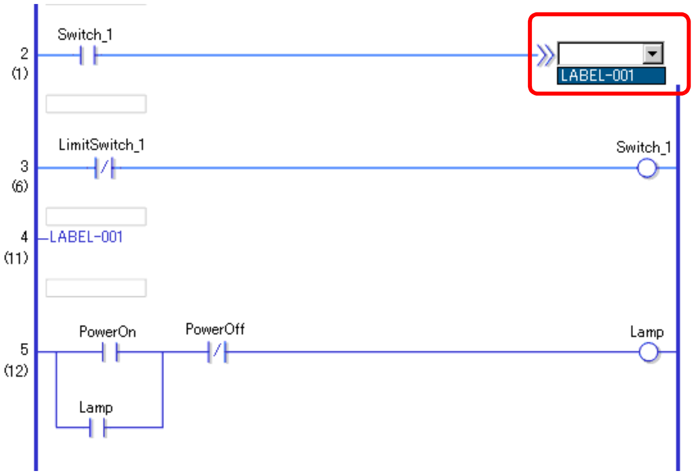
1 Insert a JMP instruction to the right of "Switch 1" of a NO instruction (Normally Open) in the 2nd rung.



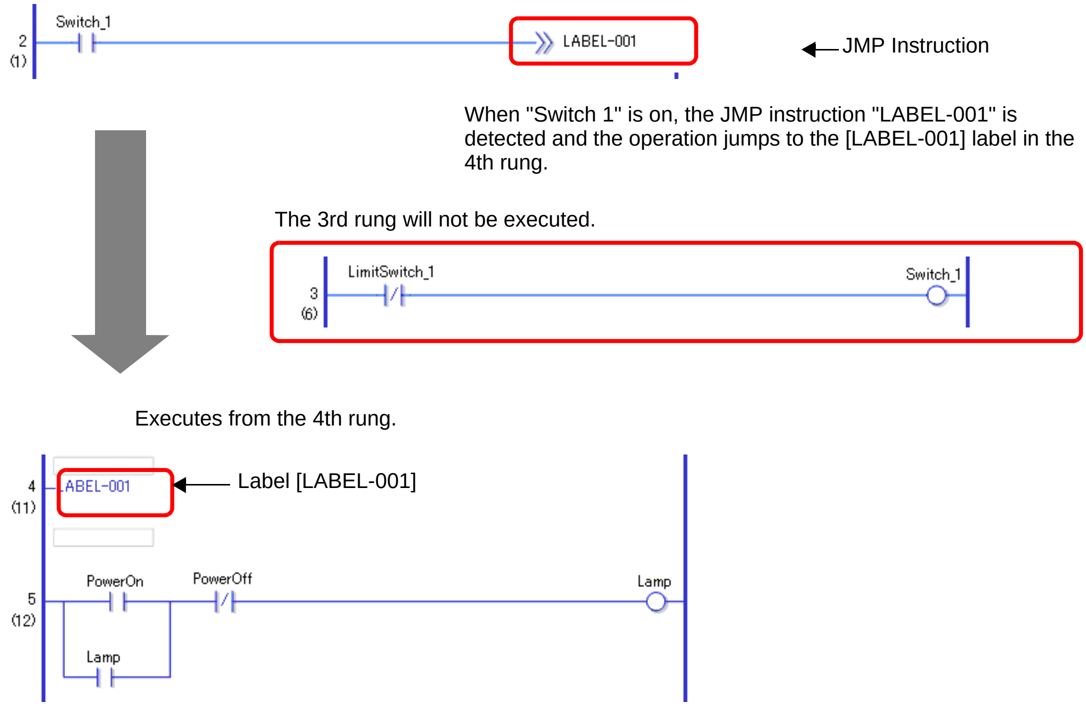
NOTE

- For how to enter an instruction, refer to the following.
  ☞ " ■ Inserting Instructions" (page 29-47)

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.



JMP Instruction

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.

Label [LABEL-001]

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 Ladder Instructions" (page 30-1)

29.6.1 Operand Settings

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

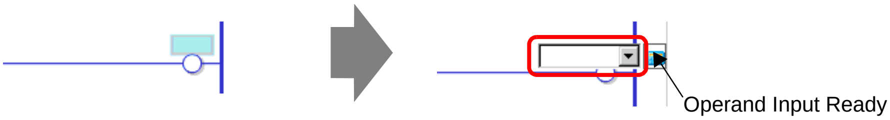
NOTE

- In the [Preferences] window, select [Logic Edit Style]. Select the [Set up operands when adding instructions] check box. The operand input box will appear when the instruction is inserted.
  ☞ "5.14.7 [Preferences] Settings Guide ■ Logic Edit Style" (page 5-152)
- You can change the operand details using the property window.
  ☞ "29.13.5 Using Reference Features to Search Logic Programs" (page 29-127)

■ 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).

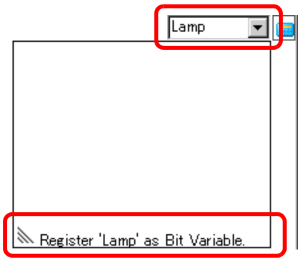
1 Double-click the OUT instruction (Out) operand. A text box will be displayed and the operand input box will appear.



NOTE

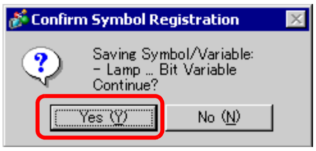
- You can display the operand input box in either of the following ways.
  •Right-click the operand and then click [Edit (E)].
  •Select the operand and press ENTER.

2 Type "lamp" in the text box and press ENTER.
A message appears: "Register 'lamp' as bit variable."


Lamp

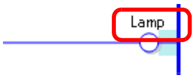
Register 'Lamp' as Bit Variable.

3 Press ENTER. The [Confirm Symbol Registration] dialog box appears. Click [Yes].


Confirm Symbol Registration

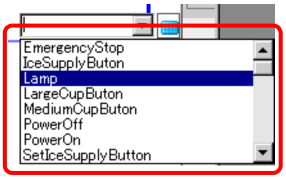
Saving Symbol/Variable:
- Lamp ... Bit Variable
Continue?

4 The symbol variable type necessary for the instruction will be allocated.
In this case, a "bit variable" type is allocated.
GP-Pro EX automatically assigns the type necessary for the new symbol variable created for the instruction.



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.

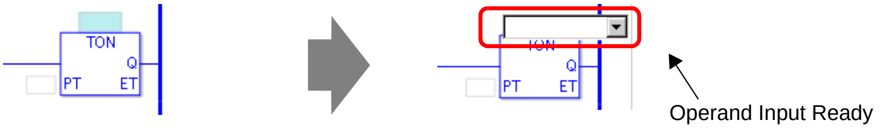


- To directly enter the address, click ▦.

■ Setting Operands, Advanced

Advanced instructions have more than one operand.
Here, the operand settings procedures for a TON instruction (On Delay Timer) are explained as an example. A symbol variable "soda injection time" is allocated to an operand and a setting time [setting time (PT)] for the timer output to turn on is allocated to another operand.

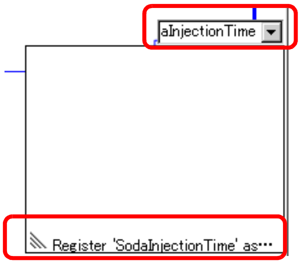
1 Double-click the TON instruction (On Delay Timer) operand. A text box is displayed for entering the operand.



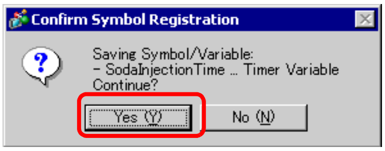
NOTE

- 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.

2 Input "soda injection time" in the text box and press [Enter] to confirm.
A message appears: "Register 'soda injection time' as timer variable."



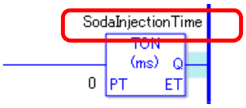
3 Press ENTER. The [Confirm Symbol Registration] dialog box appears. Click [Yes].



Confirm Symbol Registration

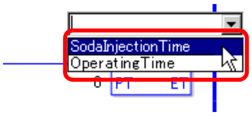
Saving Symbol/Variable:
- SodaInjectionTime ... Timer Variable
Continue?

4 The symbol variable type necessary for the instruction will be allocated. In this case, a "timer variable" type is allocated.

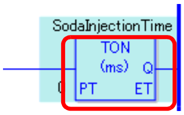


NOTE

- 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.



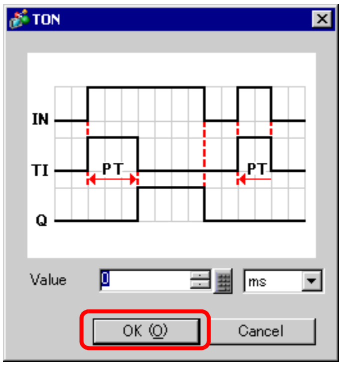
5 The default value "0" is input in the setting value [Setting Time (PT)]. Double-click the instruction to change the [Setting Time (PT)].



NOTE

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

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

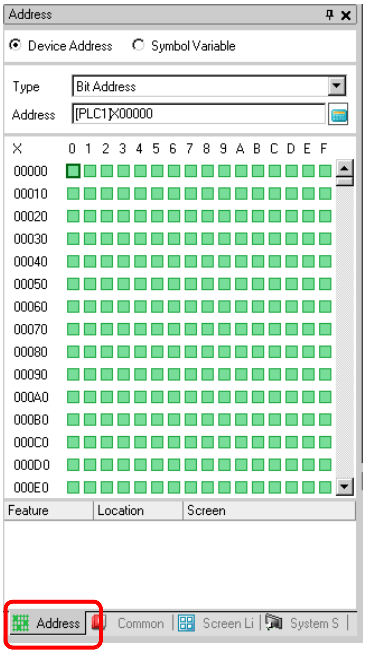


Value 0 ms

■ Operand Settings Using Drag & Drop

When the symbol variable has already been registered for the instruction, you can drag the variable from the [Address] 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.

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



● Device Address ○ Symbol Variable

Type: Bit Address

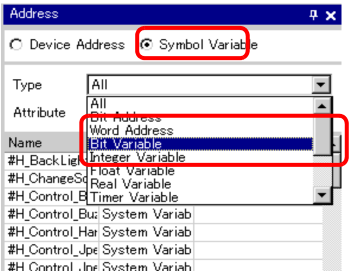
Address: [PLC1]X00000

X	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
00000	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00010	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00020	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00030	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00040	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00050	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00060	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00070	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00080	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
00090	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
000A0	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
000B0	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
000C0	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
000D0	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩
000E0	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩	🟩

NOTE

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

2 In [Type], select [Symbol Variable] and [Bit Variable].



Address

○ Device Address ● Symbol Variable

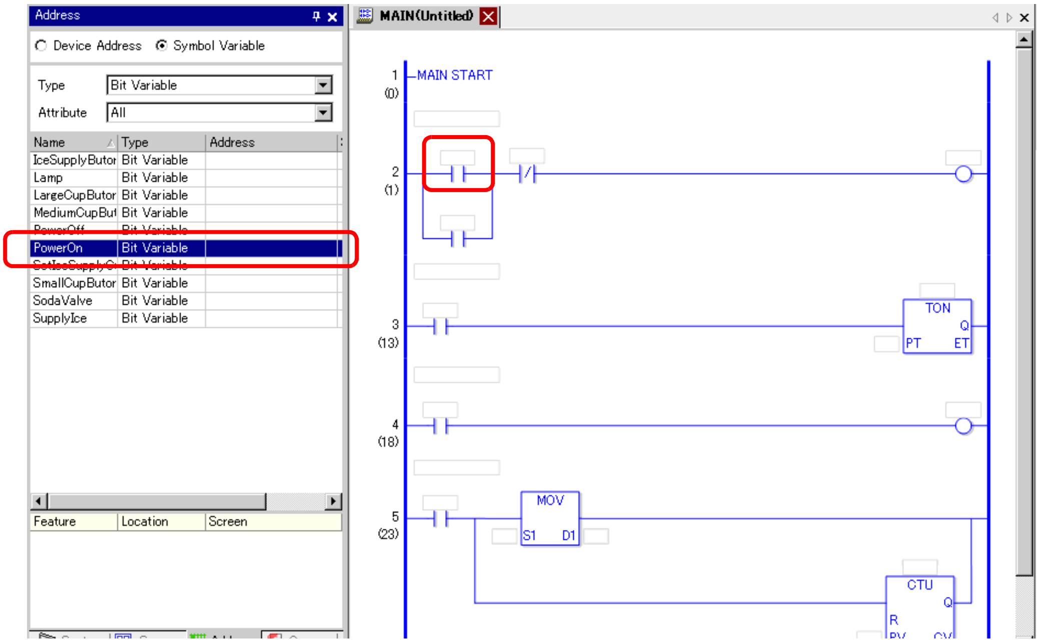
Type: All

Attribute

All
Bit Address
Word Address
Bit Variable
Integer Variable
Float Variable
Real Variable
Timer Variable

Name	
#H_BackLigl	
#H_ChangeSc	
#H_Control_B	
#H_Control_Bu:	System Variab
#H_Control_Har	System Variab
#H_Control_Jpe	System Variab
#H_Control_Jpe	System Variab

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



Address

○ Device Address ● Symbol Variable

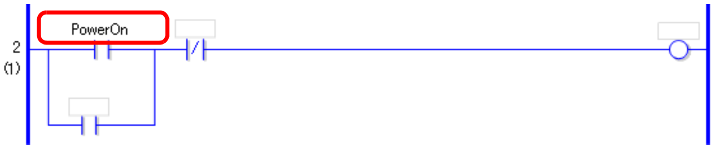
Type: Bit Variable

Attribute: All

Name	Type	Address
IceSupplyButor	Bit Variable	
Lamp	Bit Variable	
LargeCupButor	Bit Variable	
MediumCupBut	Bit Variable	
PowerOff	Bit Variable	
PowerOn	Bit Variable	
SetIceSupplyCi	Bit Variable	
SmallCupButor	Bit Variable	
SodaValve	Bit Variable	
SupplyIce	Bit Variable	

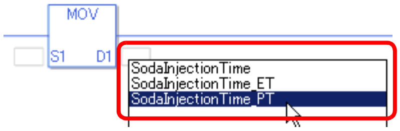
Feature	Location	Screen

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



NOTE

- All the possible variables are displayed, whether they be variable types [Bit Variable], [Integer Variable], [Float Variable], or [Real Variable] set up as arrays, or structure variables [Timer Variable], [Counter Variable], [Time Variable], [Date Variable], or [PID Variable] which consist of several other variables. Select the variable from the displayed list of variables.

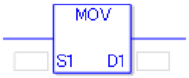


SodaInjectionTime
SodaInjectionTime_ET
SodaInjectionTime_PT

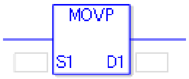
■ Pulse Settings

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

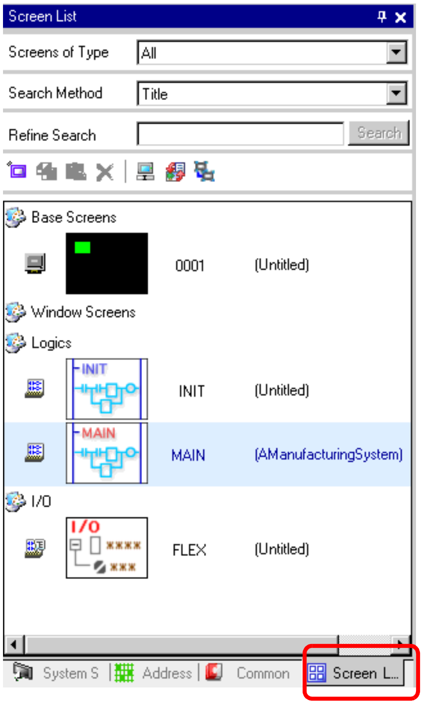
- To cancel the pulse setting, right-click to and then click [Remove Pulse].

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

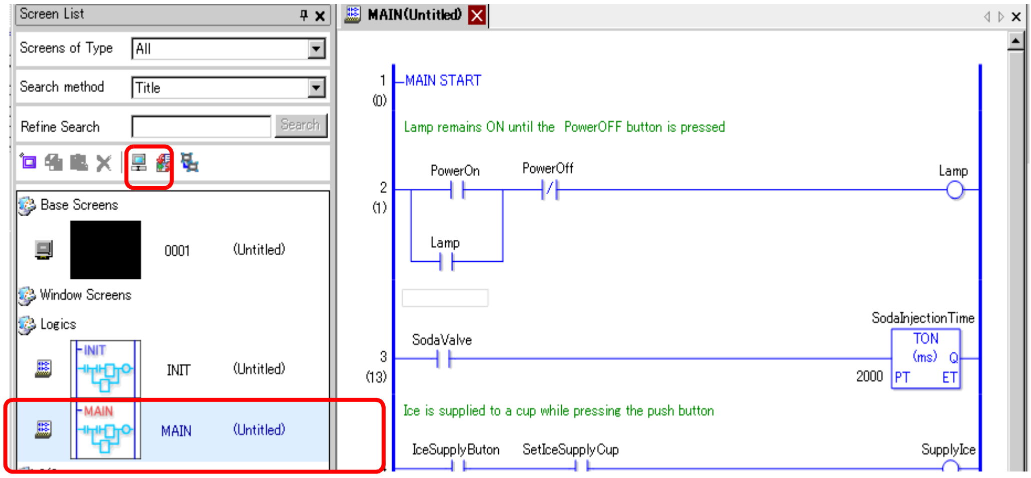
1 Click the [Screen List] tab to open the [Screen List] window.



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 (G)].

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



Screens of Type: All

Search method: Title

Refine Search

0001	(Untitled)
INIT	(Untitled)
MAIN	(Untitled)

1 (0) MAIN START

Lamp remains ON until the PowerOFF button is pressed

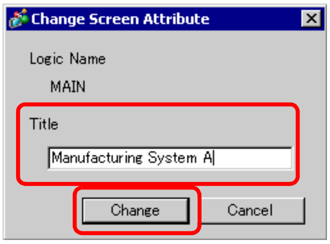
Ice is supplied to a cup while pressing the push button

NOTE

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

---

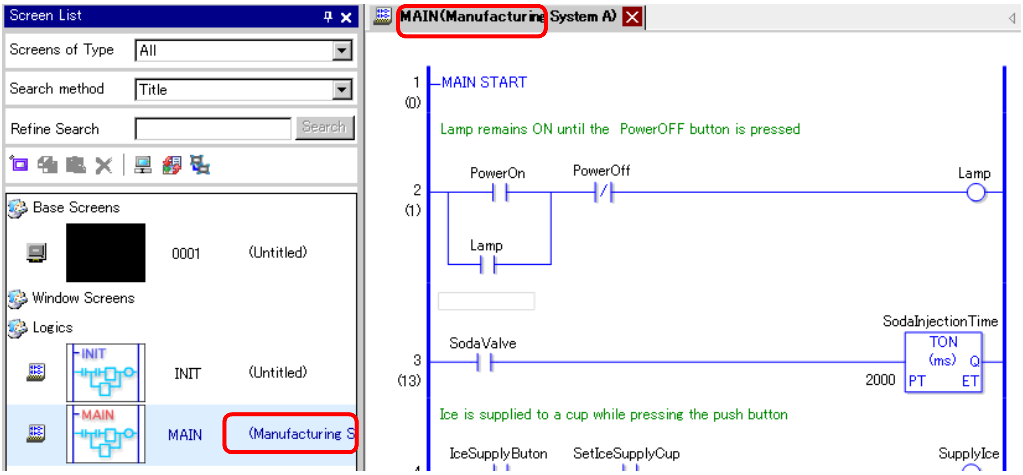
3 Enter the title and click [Change]. You can input up to 30 characters.



Logic Name: MAIN

Title: Manufacturing System A

4 The title is displayed to the right of the logic screen and in the screen tab.



Screens of Type: All

Search method: Title

Refine Search

0001	(Untitled)
INIT	(Untitled)
MAIN	(Manufacturing S

1 (0) MAIN START

Lamp remains ON until the PowerOFF button is pressed

Ice is supplied to a cup while pressing the push button

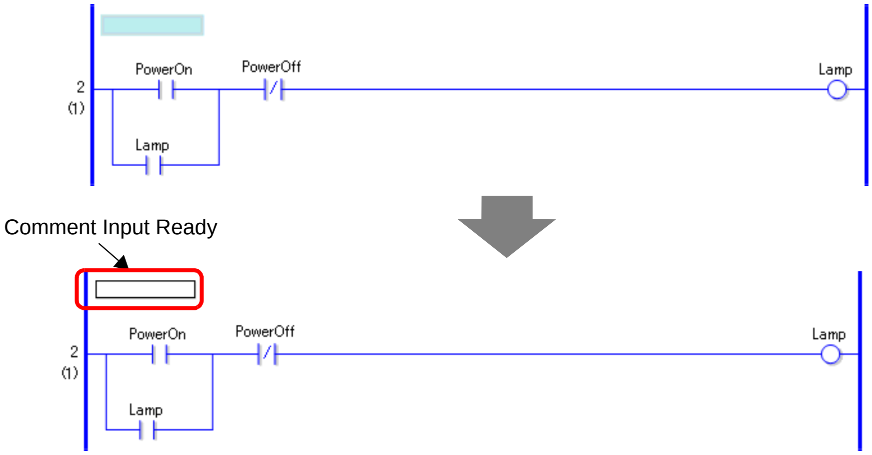
NOTE

- You can also add and change the titles in [Properties].
  ☞ “29.13.5 Using Reference Features to Search Logic Programs” (page 29-127)

29.7.2 Adding Rung Comments

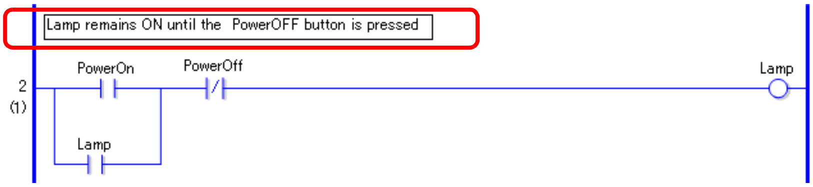
You can add comments to each rung in a logic program.

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



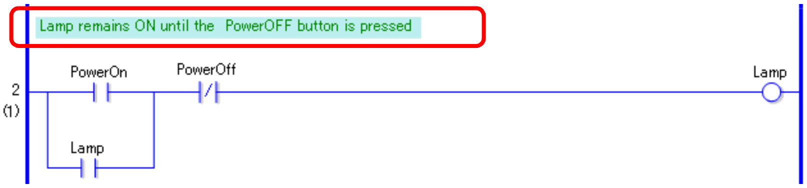
NOTE •You can also right-click the part for rung comments and then click [Edit] for comment input.

2 Enter a rung comment up to 128 characters long.


Lamp remains ON until the PowerOFF button is pressed

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

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



NOTE

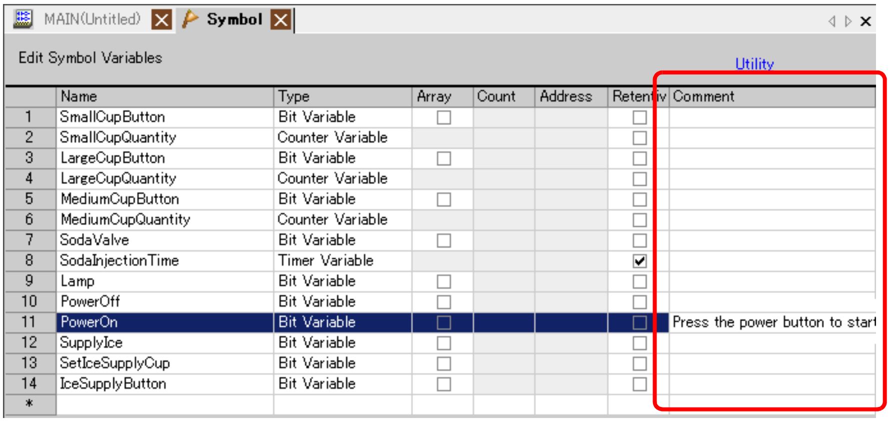
- You can also add and change rung comments in [Properties].
  ☞ "29.13.5 Using Reference Features to Search Logic Programs" (page 29-127)
- You can display a list of rung comments and edit the comments in the [Comment List] window.
  ☞ "29.7.4 [Comment List] Window" (page 29-70)

IMPORTANT

- Define the number of comments you can have in the project in the [Project Information] dialog box, [Logic Memory] area. Do not store comments that exceed the [Logic Memory].
  ☞ "29.13.2 Checking the Size for Creating Programs" (page 29-112)
- 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

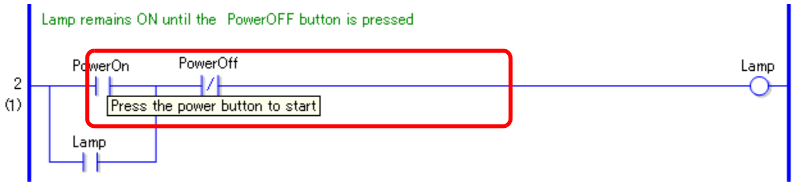
For variable comments, in the [Edit Symbol Variables] window type comments up to 32 characters long in the [Comment] field.



	Name	Type	Array	Count	Address	Retentiv	Comment
1	SmallCupButton	Bit Variable	□			□	
2	SmallCupQuantity	Counter Variable				□	
3	LargeCupButton	Bit Variable	□			□	
4	LargeCupQuantity	Counter Variable				□	
5	MediumCupButton	Bit Variable	□			□	
6	MediumCupQuantity	Counter Variable				□	
7	SodaValve	Bit Variable	□			□	
8	SodaInjectionTime	Timer Variable				☑	
9	Lamp	Bit Variable	□			□	
10	PowerOff	Bit Variable	□			□	
11	PowerOn	Bit Variable	□			□	Press the power button to start
12	SupplyIce	Bit Variable	□			□	
13	SetIceSupplyCup	Bit Variable	□			□	
14	IceSupplyButton	Bit Variable	□			□	
*							

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].
  ☞ "29.13.5 Using Reference Features to Search Logic Programs" (page 29-127)
- •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 29-70)
- •Usually, the symbol variable comments are not displayed in the Logic. Point to the symbol variable allocated to the instruction. The comment will be displayed as a tool tip.


Lamp remains ON until the PowerOFF button is pressed
2
(1)
PowerOn
PowerOff
Lamp
Press the power button to start
Lamp

IMPORTANT

- Define the number of symbol variable comments you can have in the project in the [Project Information] dialog box, [Logic Memory] area. Do not store comments that exceed the [Logic Memory].
  ☞ "29.13.2 Checking the Size for Creating Programs" (page 29-112)
- You cannot edit the symbol variable comment during online monitoring.

29.7.4 [Comment List] Window

Use the [Comment List] to view variable, system variable, and rung comments in the logic program.

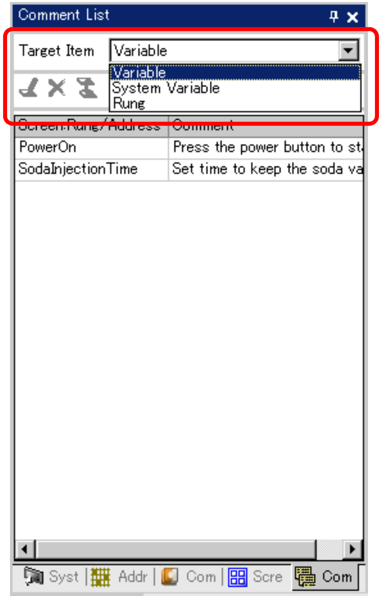
NOTE

- Please refer to the settings guide for details.
  ☞ “29.14 Settings Guide” (page 29-136)

■ Using the [Comment List] Window

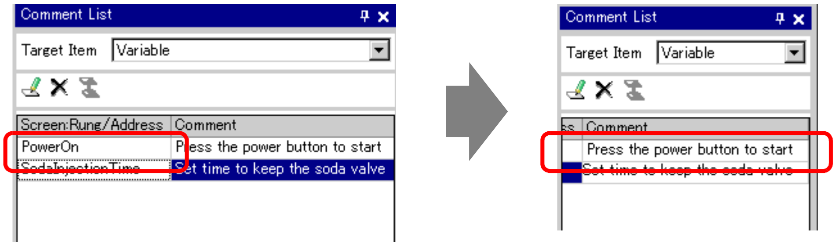
1 On the [View (V)] menu, point to [Work Space (W)] and then click [Comment List (C)]. The [Comment List] window opens.

2 In [Target Item], select the type of comment type you want to display in the list.



Screen:Rung/Address	Comment
PowerOn	Press the power button to st
SodaInjectionTime	Set time to keep the soda va

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

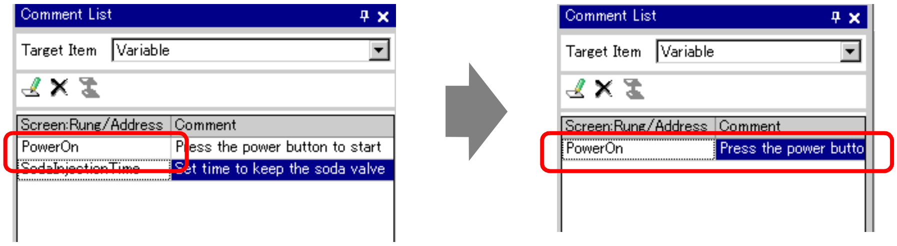


Comment List
Target Item
Variable
Screen:Rung/Address
Comment
PowerOn
Press the power button to start
SodaInjectionTime
Set time to keep the soda valve
Comment List
Target Item
Variable
Comment
Press the power button to start
Set time to keep the soda valve

NOTE

- 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 ✕.



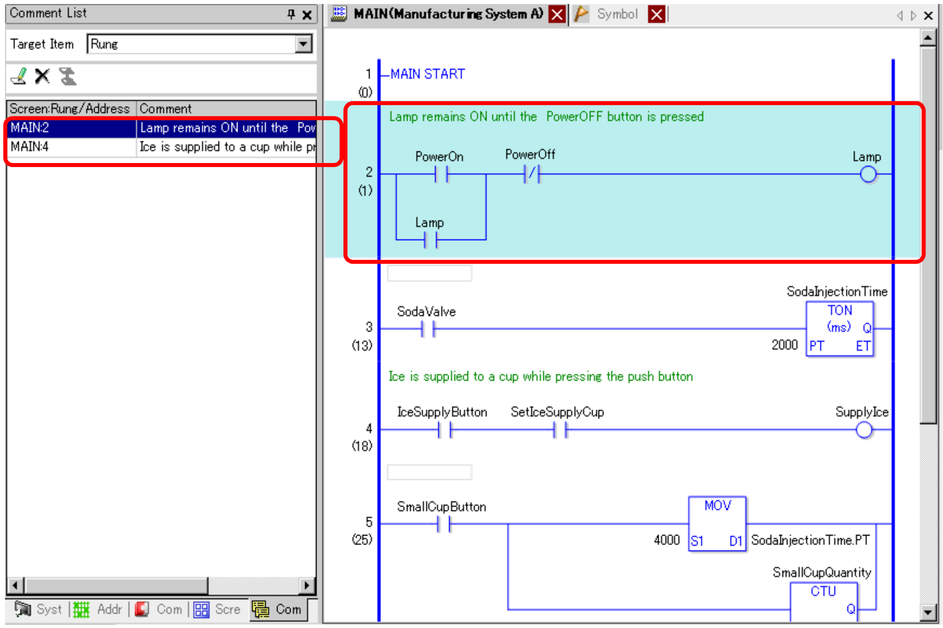
Screen:Rung/Address	Comment
PowerOn	Press the power button to start
SodaInjectionTime	Set time to keep the soda valve

Screen:Rung/Address	Comment
PowerOn	Press the power butto

NOTE

- 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 in the Logic with the comment you want to delete.

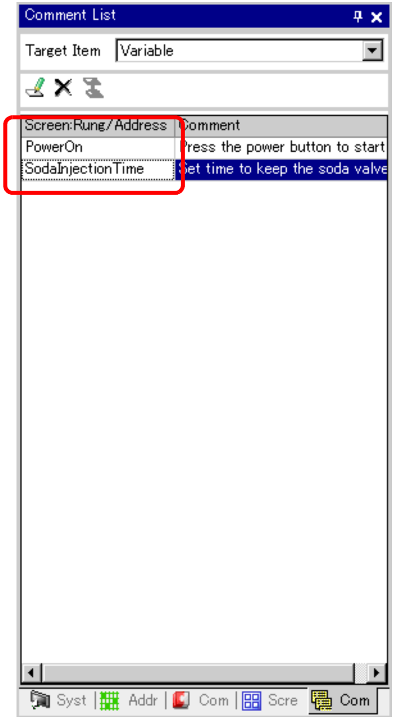


Screen:Rung/Address	Comment
MAIN:2	Lamp remains ON until the Pow
MAIN:4	Ice is supplied to a cup while pr

Lamp remains ON until the PowerOFF button is pressed

Ice is supplied to a cup while pressing the push button

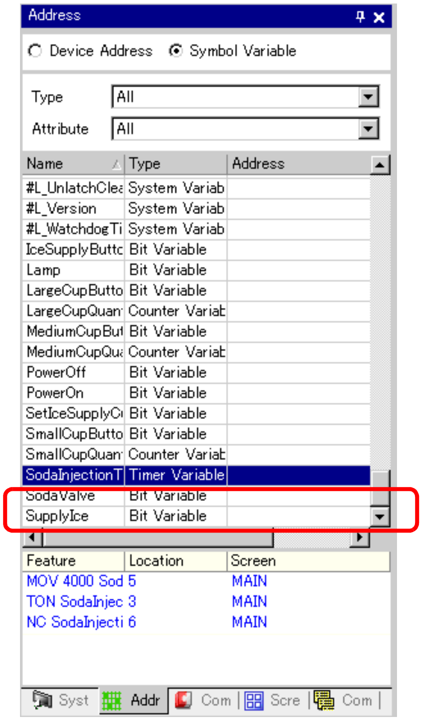
6 When [Target Item] is [Variable] or [System Variable], double-click the cell in the [Screen: Rung/Address] column. This displays the [Address] windows and selects the relevant symbol variable or system variable.



Comment List

Target Item: Variable

Screen:Rung/Address	Comment
PowerOn	Press the power button to start
SodaInjectionTime	Set time to keep the soda valve



Address

○ Device Address ● Symbol Variable

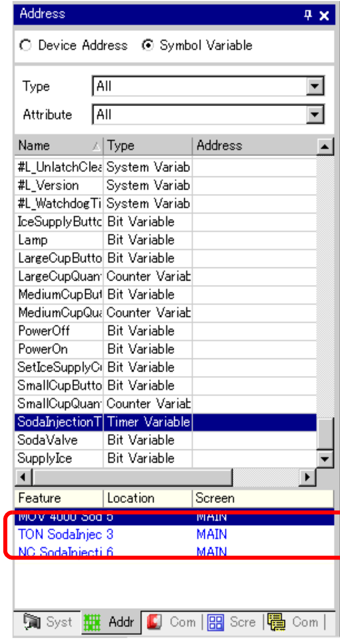
Type: All

Attribute: All

Name	Type	Address
#L_UnlatchClea	System Variab	
#L_Version	System Variab	
#L_WatchdogTi	System Variab	
IceSupplyButto	Bit Variable	
Lamp	Bit Variable	
LargeCupButto	Bit Variable	
LargeCupQuan	Counter Variab	
MediumCupBut	Bit Variable	
MediumCupQua	Counter Variab	
PowerOff	Bit Variable	
PowerOn	Bit Variable	
SetIceSupplyCu	Bit Variable	
SmallCupButto	Bit Variable	
SmallCupQuan	Counter Variab	
SodaInjectionT	Timer Variable	
SodaValve	Bit Variable	
SupplyIce	Bit Variable	

Feature	Location	Screen
MOV 4000 Sod	5	MAIN
TON SodaInjec	3	MAIN
NC SodaInjecti	6	MAIN

7 Select the lower part of the [Address] window to select the target variable on the logic screen.



Address

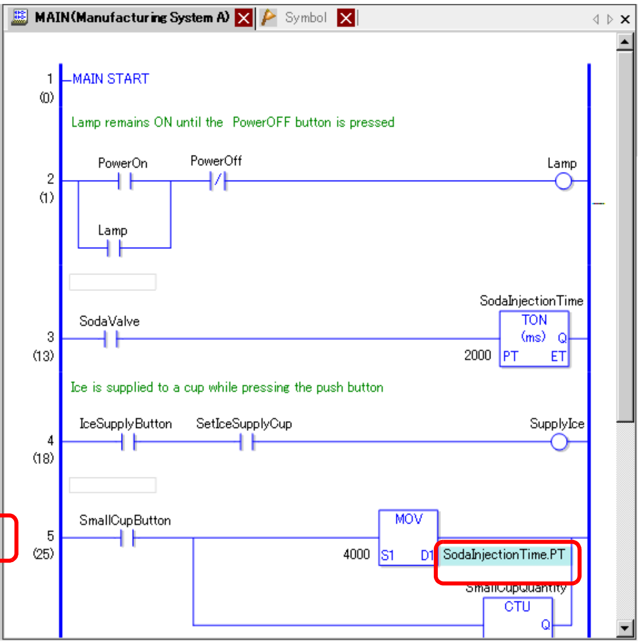
○ Device Address ● Symbol Variable

Type: All

Attribute: All

Name	Type	Address
#L_UnlatchClea	System Variab	
#L_Version	System Variab	
#L_WatchdogTi	System Variab	
IceSupplyButto	Bit Variable	
Lamp	Bit Variable	
LargeCupButto	Bit Variable	
LargeCupQuan	Counter Variab	
MediumCupBut	Bit Variable	
MediumCupQua	Counter Variab	
PowerOff	Bit Variable	
PowerOn	Bit Variable	
SetIceSupplyCu	Bit Variable	
SmallCupButto	Bit Variable	
SmallCupQuan	Counter Variab	
SodaInjectionT	Timer Variable	
SodaValve	Bit Variable	
SupplyIce	Bit Variable	

Feature	Location	Screen
MOV 4000 Sod	5	MAIN
TON SodaInjec	3	MAIN
NC SodaInjecti	6	MAIN



MAIN(Manufacturing System A)

Symbol

1 (0) MAIN START

Lamp remains ON until the PowerOFF button is pressed

Ice is supplied to a cup while pressing the push button

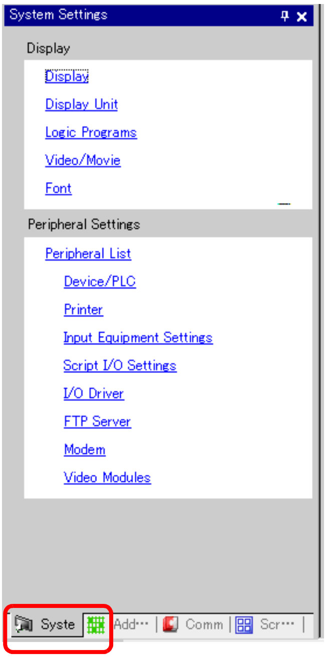
29.8 Logic Operations with Power ON

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 29-136)
- You can configure the logic screen settings in offline mode.

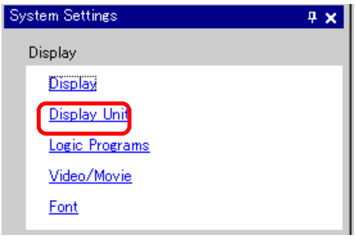
1 Click the [System Settings] tab to open [System Settings].



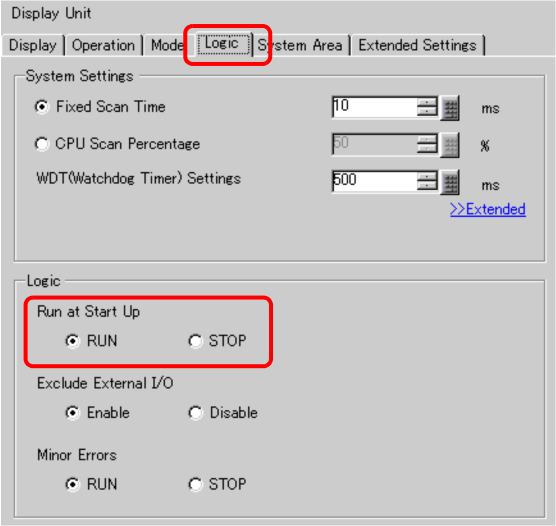
NOTE

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

2 Select [Display Unit] from [Display].



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



Display Unit

Display | Operation | Mode | Logic | System Area | Extended Settings

System Settings

● Fixed Scan Time 10 ms

○ CPU Scan Percentage 50 %

WDT(Watchdog Timer) Settings 500 ms

>>Extended

Logic

Run at Start Up

● RUN ○ STOP

Exclude External I/O

● Enable ○ Disable

Minor Errors

● RUN ○ STOP

29.9 Transferring Logic Programs

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

NOTE

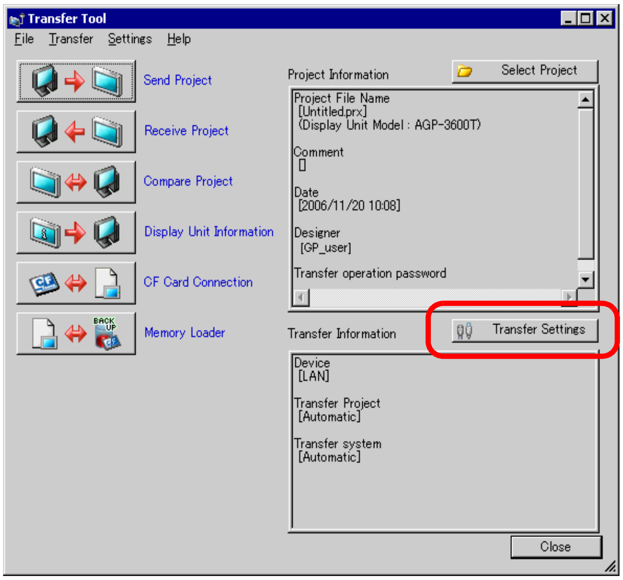
- For details on how to transfer, refer to the following.
  ☞ "Chapter 33 Transferring Data" (page 33-1)
- The logic program is checked for errors when transferring and saving. If any error is found in the check, transfer to the GP fails.
  ☞ "29.11 Correcting Logic Program Errors" (page 29-89)
  ☞ "33.9 Checking Errors" (page 33-54)

■ Retentive Transfer

When using the transfer tool to download a project with the same name as on the GP, and project transfer is set to [Auto] and the Retain retentive variables check box is selected, you can transfer the project while retaining GP variable values backed up to SRAM. Variable values cannot be retained at download if the system settings are different, the project on the GP is damaged, the Retain retentive variables check box is not selected, or transfer is set to Forced. Here, if the keep transfer check box is not selected or the compulsory transfer check box is selected, you cannot transfer the file by keeping the current value.
If the [Retain retentive variables] check box is cleared, the variable value is cleared to 0 even if in the [Symbol Variable] common settings [Retentive] is selected. Select the [Retain retentive variables] check box to retain values of variables that use the [Retentive] setting.

1 Connect the GP to your PC.

2 On the state toolbar, click the transfer project icon to start the transfer tool and click [Transfer Settings].



Project Information

Project File Name
[Untitled.prx]
(Display Unit Model : AGP-3600T)

Comment
[]

Date
[2006/11/20 10:08]

Designer
[GP_user]

Transfer operation password

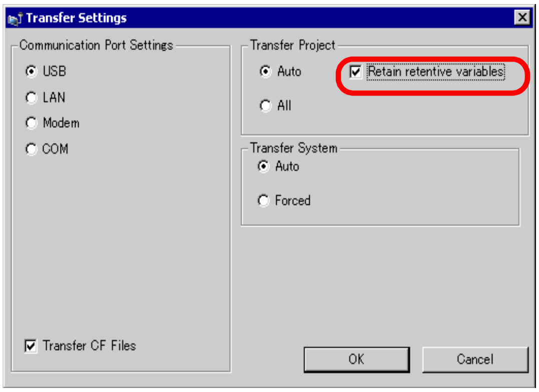
Transfer Information

Device
[LAN]

Transfer Project
[Automatic]

Transfer system
[Automatic]

3 The [Transfer Settings] dialog box appears. Select the [Retain Retentive Variables] check box in [Transfer Project] and click [OK].



Transfer Settings

Communication Port Settings
- ● USB
- ○ LAN
- ○ Modem
- ○ COM

☑ Transfer CF Files

Transfer Project
- ● Auto
- ○ All

☑ Retain retentive variables

Transfer System
- ● Auto
- ○ Forced

4 In the Transfer Tool, click [Send Project] to start transferring to the GP.

29.10 Monitoring Logic Programs (Online Monitor)

You can monitor from your PC the GP logic program while it is running.
In the [Watch List] window, you can confirm the symbol variable ON/OFF state or device values. In the [PID Monitor] window, you can also make adjustments to the PID instruction values while monitoring. These features make it useful for debugging.
The Online Monitor can be run simultaneously with a computer connected via USB and a computer connected via Ethernet (LAN).

NOTE

- AGP3301S, AGP3301L, and AGP3302B do not support online monitor function.
- In [Preferences], you can configure the communication settings and monitor the settings with online monitoring.
  ☞ "5.14.7 [Preferences] Settings Guide ■ Monitor Step" (page 5-153)

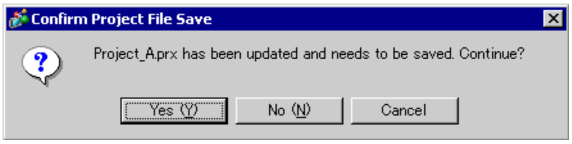
29.10.1 Online Monitoring Procedures

1 To your PC, connect the GP that you want to monitor.

2 In GP-Pro EX, from the state toolbar, click the monitor icon .

NOTE

- If you click the monitor icon in the middle of editing the project, the [Confirm Project File Save] dialog box appears.

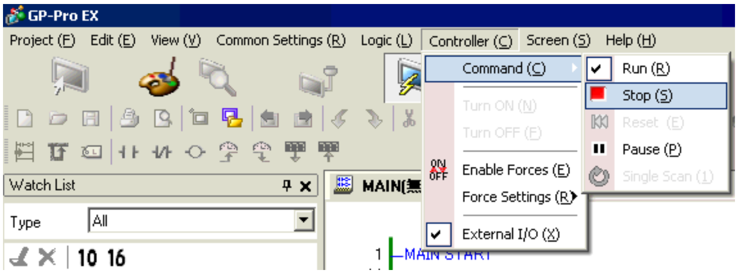


Confirm Project File Save

Project_A.prx has been updated and needs to be saved. Continue?

  Click [Yes] to save the edited logic program. Upon saving, the logic program performs an error check. If any errors are detected, monitor mode will not start. Instead, an error message will be displayed. Click [OK], correct the error, and then try again. If there is no error, monitor mode starts.
  Click [No] and any edits to the logic program are discarded and monitor mode starts.
  Click [Cancel] to return to the editor without saving any changes.
- If a monitor password has been set up, enter the password.
  ☞ "29.13.4 Enhancing Security" (page 29-125)

3 The lines in the logic program turn green to show the operational status of the logic program. To stop the logic program, select [Command] from the [Controller] menu and then click [Stop] or [Pause]. (Stopping the logic program changes the front LED of the GP unit from constant green to flashing green.)



29.10.2 Monitor/Change the Current Value of Symbol Variables

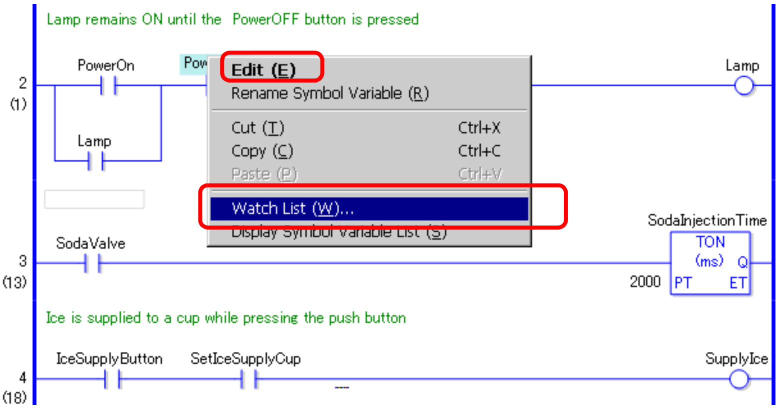
You can monitor the symbol variable ON/OFF and the device values within the logic program registered in the [Watch List] window.

NOTE

- Please refer to the settings guide for details.
  ☞ “29.14.2 [Work Space] Settings Guide ■ Watch List Window” (page 29-142)

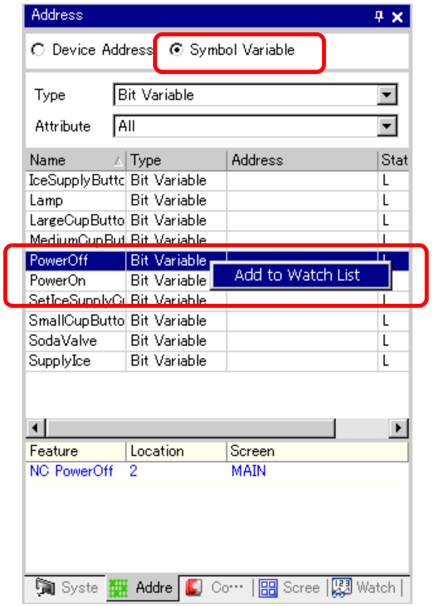
1 On the [View (V)] menu, point to [Work Space (W)] and then click [Watch List (W)]. The [Watch List] window appears.

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



NOTE

- To add a variable to the Watch List, drag the symbol variable to monitor and drop it into the Watch List.
- You can also add the variable by selecting [Symbol Variable] in the [Address] window, right-clicking the symbol variable that you want to monitor, and clicking [Add to Watch List].



○ Device Address ● Symbol Variable

Type: Bit Variable

Attribute: All

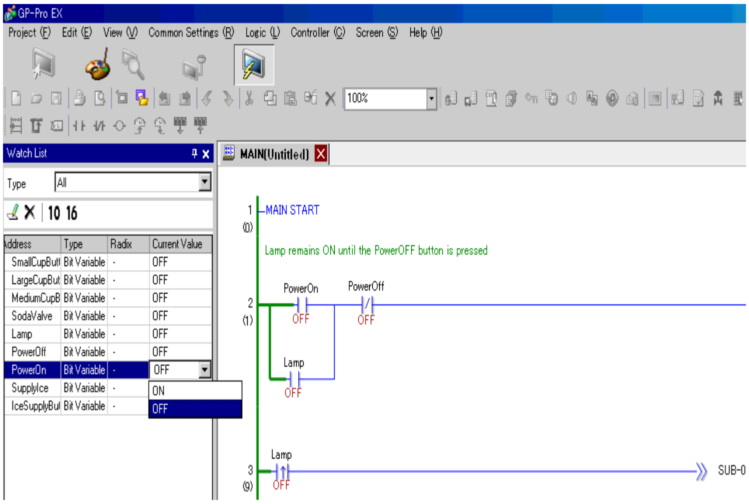
Name	Type	Address	Stat
IceSupplyButtc	Bit Variable		L
Lamp	Bit Variable		L
LargeCupButto	Bit Variable		L
MediumCupBut	Bit Variable		L
PowerOff	Bit Variable		L
PowerOn	Bit Variable		
SetIceSupplyCu	Bit Variable		L
SmallCupButto	Bit Variable		L
SodaValve	Bit Variable		L
SupplyIce	Bit Variable		L

Feature	Location	Screen
NC PowerOff	2	MAIN

3 In the [Watch List]'s Type list, select the data type of symbol variables to monitor.

4 Transfer the project to the GP.

5 From the GP-Pro EX State toolbar, click the Monitor icon to shift to Monitor Step.


GP-Pro EX
Project (F) Edit (E) View (V) Common Settings (R) Logic (L) Controller (C) Screen (S) Help (H)
100%
Watch List
MAIN(Untitled)
10 16
1
(0)
MAIN START
Lamp remains ON until the PowerOFF button is pressed
2
(1)
PowerOn
OFF
PowerOff
OFF
Lamp
OFF
3
(9)
Lamp
OFF
SUB-0

Type: All

Address	Type	Radix	Current Value
SmallCupButt	Bit Variable	-	OFF
LargeCupBut	Bit Variable	-	OFF
MediumCupB	Bit Variable	-	OFF
SodaValve	Bit Variable	-	OFF
Lamp	Bit Variable	-	OFF
PowerOff	Bit Variable	-	OFF
PowerOn	Bit Variable	-	OFF
SupplyIce	Bit Variable	-	ON
IceSupplyBut	Bit Variable	-	OFF

6 Monitor the current value of registered symbol variables in the [Watch List]. You can check how operations are affected when you change the current value of each address. To check operations when the [Type] is [Bit Variable], you can right-click to select [Force ON] or [Force OFF].

7 To exit the online monitor and return to the editor, on the State toolbar click the [Edit] icon .

NOTE

- When you specify [Force ON] or [Force OFF], the symbol variable keeps its ON or OFF state until you remove forces.
- You can add addresses to the [Watch List] while in Monitor Step. However, when you exit Monitor Step and return to the editor, the registered contents are discarded. To avoid this situation, transfer the project after adding to the Watch List.

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.
  ☞ "29.14.2 [Work Space] Settings Guide ■ PID Monitor Window" (page 29-138)

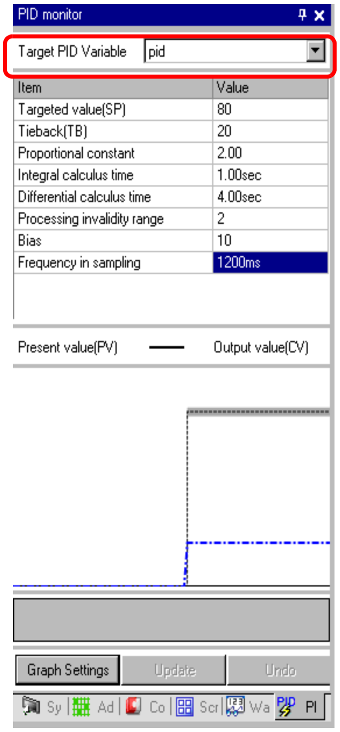
1 Run the online monitor.

NOTE

☞ "29.10.1 Online Monitoring Procedures" (page 29-77)

2 On the [View (V)] menu, point to [Work Space (W)] and then click [PID Monitor (M)]. The [PID monitor] window opens.

3 In [Target PID Variable], select the PID variable that you want to monitor. The PID instruction setting is displayed in a graph.



Target PID Variable: pid

Item	Value
Targeted value(SP)	80
Tieback(TB)	20
Proportional constant	2.00
Integral calculus time	1.00sec
Differential calculus time	4.00sec
Processing invalidity range	2
Bias	10
Frequency in sampling	1200ms

4 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	Displays PID variables 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 FormatU_****.KP	Set the (×1000) value.
Integral Calculus Time	Variable Format****.IT Address FormatU_****.IT	Set the (×1000) value.
Differential Calculus Time	Variable Format****.DT Address FormatU_****.DT	Set the (×1000) value.
Processing Deadband Range	Variable Format****.PA Address FormatU_****.PA	
Bias	Variable Format****.BA Address FormatU_****.BA	
Frequency in Sampling	Variable Format****.ST Address FormatU_****.ST	

NOTE

- Click [Update] to update the graph.
- Click [Graph] to change the graph display’s Upper Limit, Lower Limit, and display Width settings.

29.10.4 Editing the Logic Program When Monitoring (Online Edit)

While running online monitoring, you can edit the logic program. With Online Edit, you can do the following.

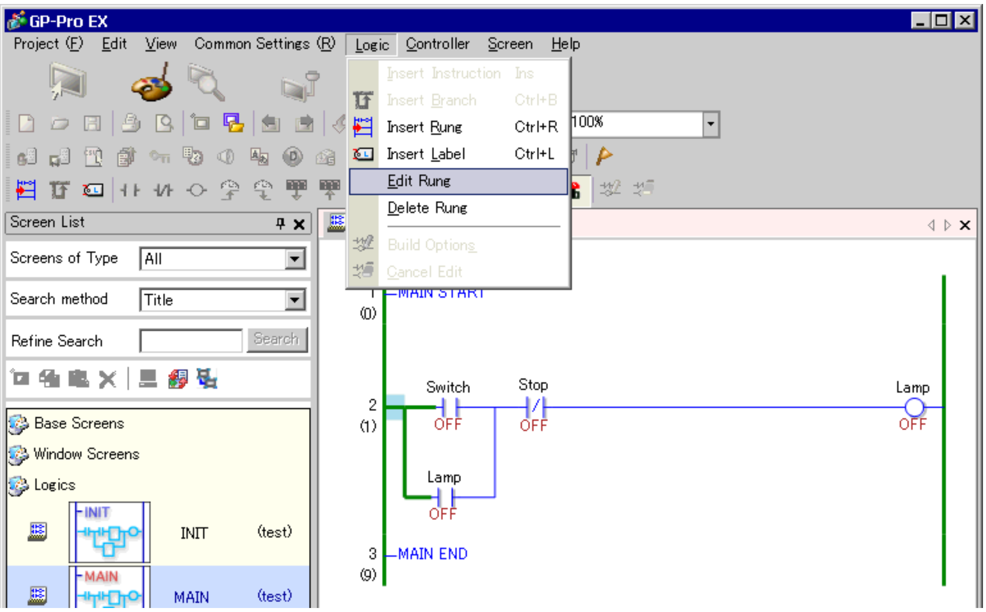
- Insert/Delete Rungs
- Insert/Delete Branching
- Insert/Delete Instructions
- Edit Operands
- Insert/Delete Labels

IMPORTANT

- With Online Edit, you cannot create a new variable. Allocate existing variables when adding instructions.
- After editing, an error check is performed in the logic program. If any errors are found, the transfer will not occur.

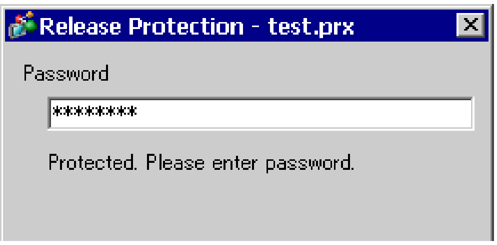
■ Editing Procedures

In Online Edit, you can edit only a single rung at a time. Click [Edit Rung] in the [Logic] menu to edit the selected rung. Click [OK] to transfer the edited logic program to the GP. Click [Cancel] to cancel editing and go back to the online monitor.



NOTE

- Each time an online edit is performed, 1 is added to the system variable (#L_EditCount) showing the number of edits.
  For details on system variables, see "A.6 System Variables" (page A-84)
- If a password has been set for online editing, the [Password Confirmation] dialog box is displayed before starting editing.
- To set the password, refer to "29.14.1 [Logic Programs] Setting Guide" (page 29-136) .



Release Protection - test.prx

Password

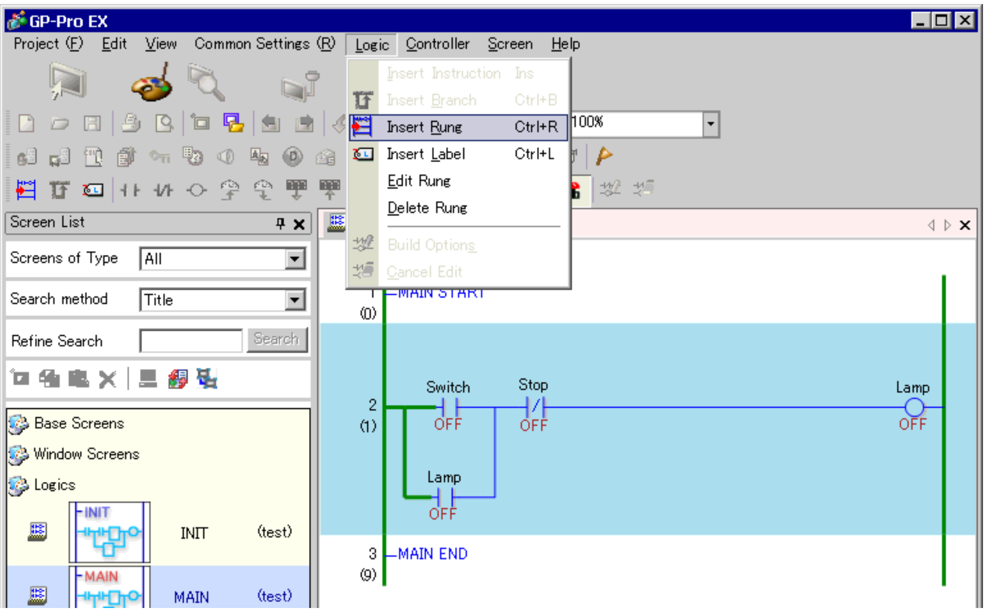
****

Protected. Please enter password.

◆ Inserting/Deleting Rungs

A rung is inserted one down from the rung you selected. To insert, select a rung when the online monitor is on, and click [Insert Rung] under the [Logic] menu.

To delete a rung, select the rung to be deleted and click [Delete] on the [Edit] menu.



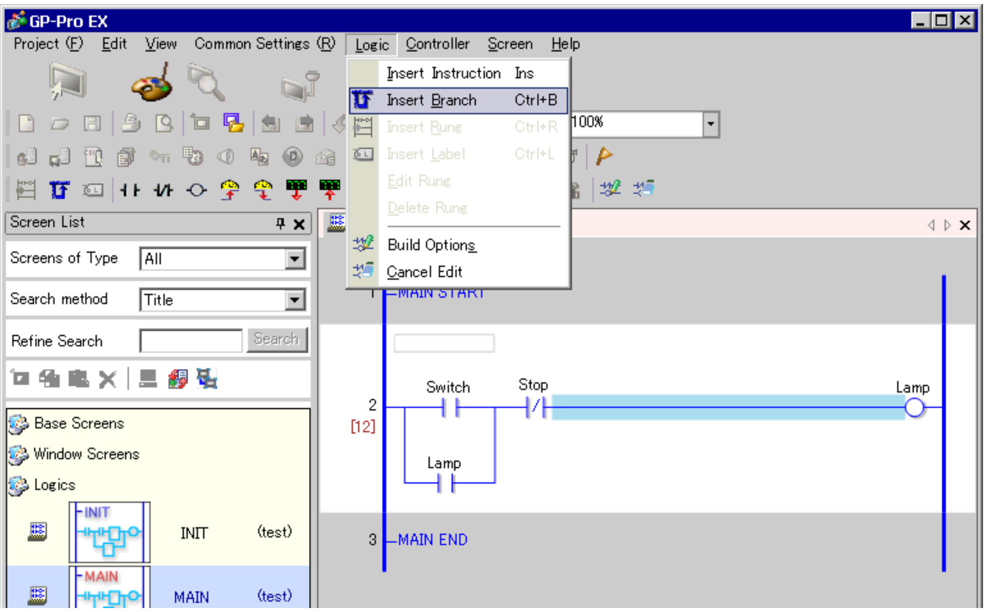
NOTE

- Upon deleting a rung, the [Transfer Logic] dialog box is displayed and the modified logic program will be transferred to the GP. You do not need to go to the [Logic] menu and click [OK].
- Alternatively, you can insert/delete rungs from the [Edit] menu or right click the menu.

◆ Inserting/Deleting Branching

Select the point where you want to insert a branch and click [Insert Branch] on the [Logic] menu.

To delete, go to the [Edit] menu. Then click [Delete]

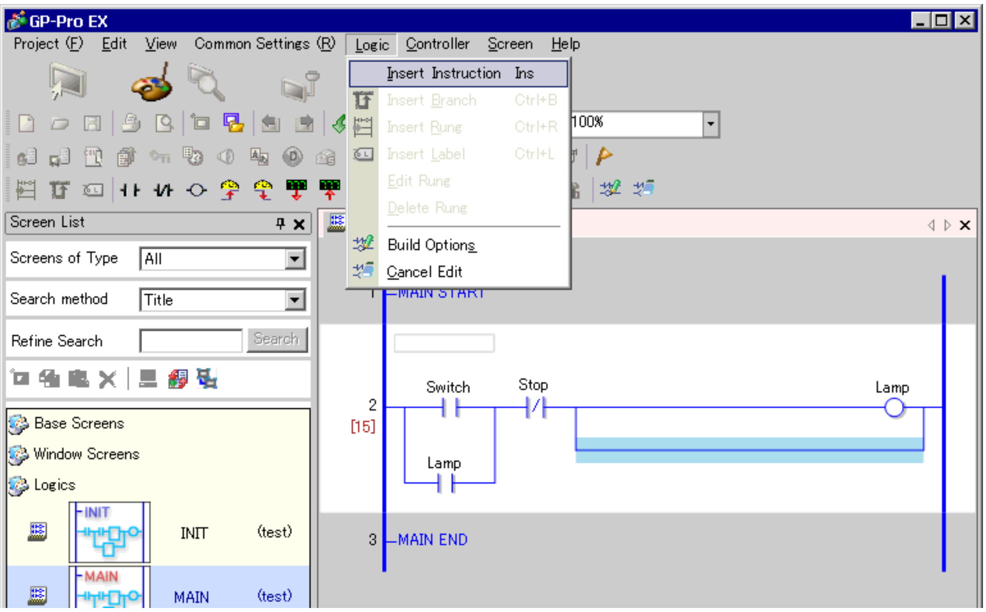


NOTE

- Alternatively, you can insert/delete branches from the [Edit] menu or right click the menu.

◆ Inserting/Deleting Instructions

Select the point where you want to insert an instruction and click [Insert Instruction] on the [Logic] menu.
To delete, go to the [Edit] menu. Then click [Delete].



IMPORTANT

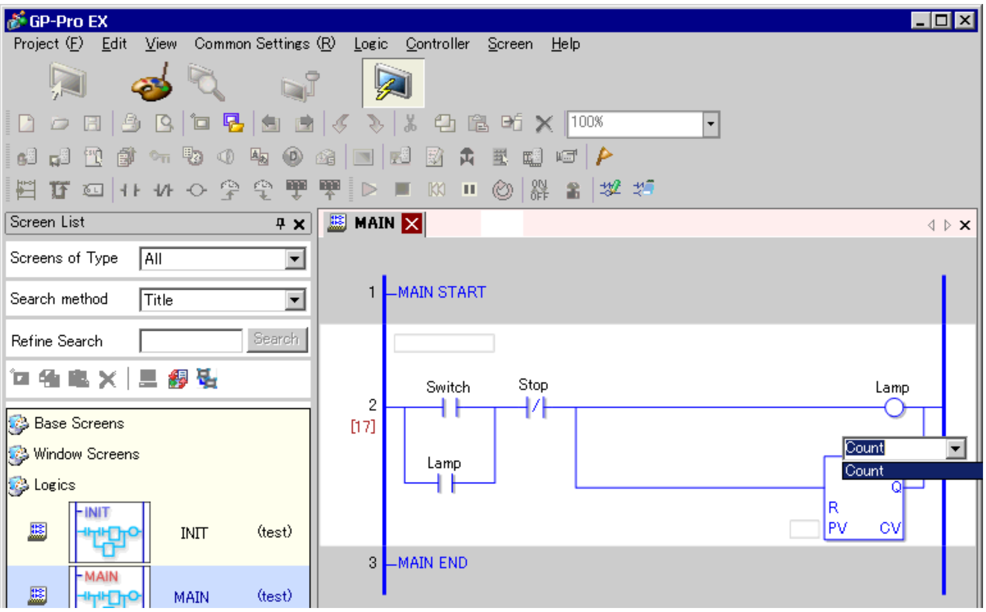
- With Online Edit, you cannot create a new variable. Allocate existing variables when adding instructions.

NOTE

- Alternatively, you can insert/delete instructions from the [Edit] menu or right click the menu.

◆ Editing Operands

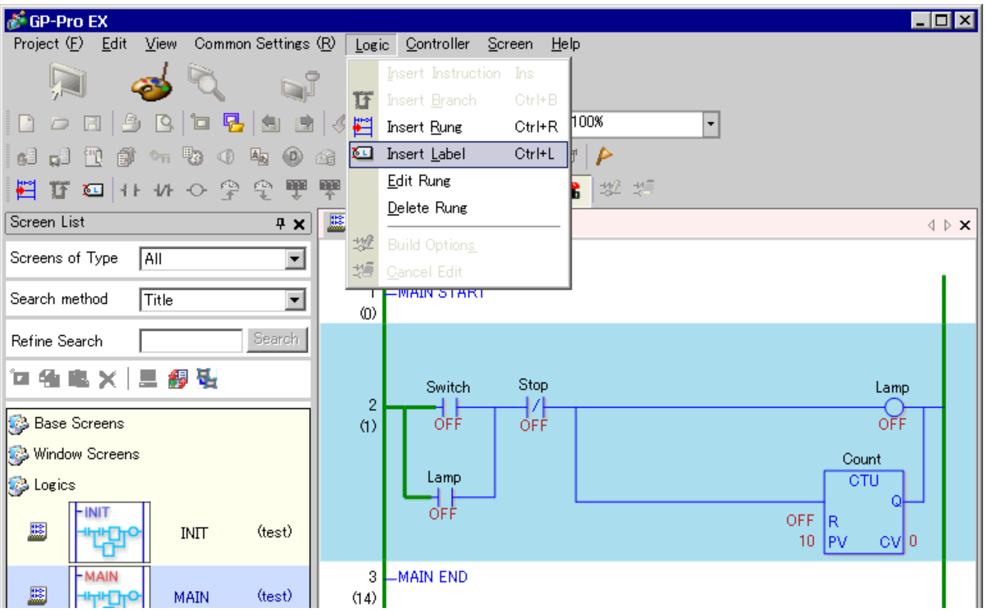
Select the operand to be edited, then select [Cut/Copy/Paste/Delete] on the [Edit] menu.



◆ Inserting/Deleting Labels

A label is inserted one down from the rung you select. To insert, select a rung when online monitor is on, and click [Insert Label] on the [Logic] menu.

To delete, select the label to be deleted, then click [Delete] on the [Edit] menu.



NOTE

- Upon deleting a label, the [Transfer Logic] dialog box is displayed and the modified logic program will be transferred to the GP. You do not need to go to the [Logic] menu and click [OK].

■ Restrictions

- When editing online, you can only edit a rung at a time. You cannot edit multiple rungs at a time.
- Scan time delay

  Upon clicking [OK], the modified logic program will be loaded to the GP. At that time, a scan time delay may occur only once.

  ExampleThe following delay may occur when adding 339 steps (8 timer instructions, 8 counter instructions) to the running 10000 steps logic program:

  GP-3300 Series: Approx. 8.1ms

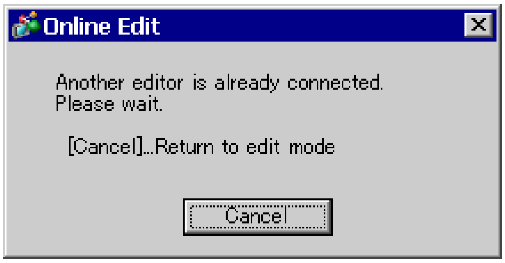
  GP-3400/3500/3600/3700 Series: Approx. 2.9ms

  To avoid #L_WatchdogTime error caused by delay, the Watchdog Time is ignored for the one scan mentioned above.

  When you end online editing, the Watchdog Time settings will be enabled.
- Monitoring from Multiple Editors

  Using the online monitor, you can connect two computers at the same time, one using a USB connection and the other using an Ethernet connection.

  When one of the computers makes an online editing connection, it forces termination of the other computer's online monitor connection. The following dialog box appears to indicate that online editing is used and the online monitor will stop.



Online Edit

Another editor is already connected.
Please wait.

[Cancel]...Return to edit mode

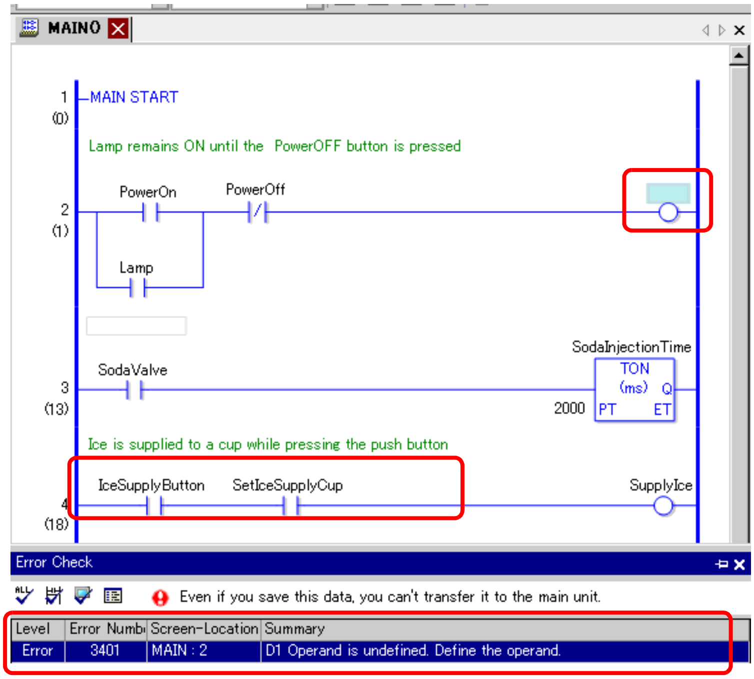
29.11 Correcting Logic Program Errors

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.
  ☞ "33.9 Checking Errors" (page 33-54)

If errors are found in the error check, a list of errors is displayed. Refer to [Level], [Error Number], [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.



Even if you save this data, you can't transfer it to the main unit.

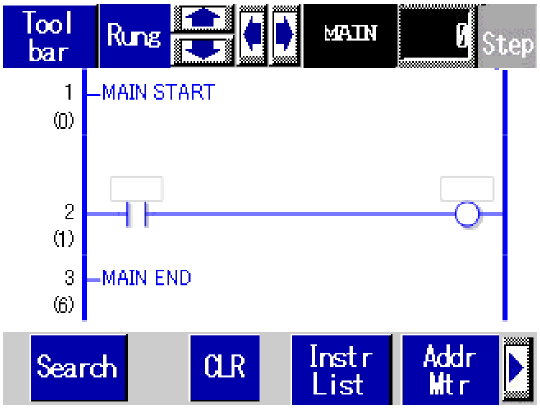
Level	Error Numb	Screen-Location	Summary
Error	3401	MAIN : 2	D1 Operand is undefined. Define the operand.

NOTE

- In [Preferences], you can select the settings to display duplicate coil warnings during error checking.
  ☞ "5.14.7 [Preferences] Settings Guide ■ Error Check" (page 5-154)

29.12 Logic Monitor

Logic monitoring is the function that displays the logic program on the GP screen. It allows you to check the operations of the logic program with only the GP unit without using a personal computer.
The program is executed without stopping even during logic monitoring.



29.12.1 Starting and Ending the Logic Monitor

■ Triggered Method

There are 3 ways to start the logic monitor.

- Start up with Parts
  Logic monitoring begins when you turn on the first bit of the #L system variable (#L_LogicMonitor) using a switch part.
  Turn off the first bit to display the screen before logic monitoring begins. Specify #L_LogicMonitor.X[1] for the address. The #L_LogicMonitor.X[0] is for monitoring addresses. The address monitoring screen appears when the system variable specification bit is turned on, such as during logic monitoring.
- 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

- You can not start up multiple monitors at the same time. When bits 0 and 1 are turned on at the same time, the address monitor turns on and bit 1 turns off.
- Once the monitors have started up, it is easy to switch monitors. Bit 0 or bit 1 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 the logic monitor on AGP-3302B or AGP-3301. The address monitor starts 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] in each of the logic monitoring and address monitoring screens.

NOTE

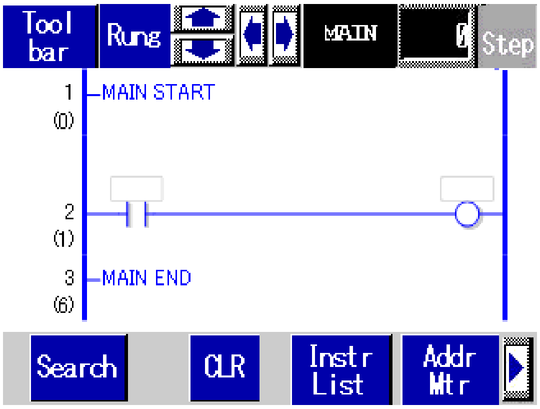
- If not changing the screens, click [Back to Previous] to end.
- Please note that if there is no screen to go back to, such as when you start up the logic monitor when the initial screen was not on, you will not be able to end the logic monitor
- When the logic monitor and address monitor end, #L_LogicMonitor is Zero-cleared.

29.12.2 Logic Monitor Functions

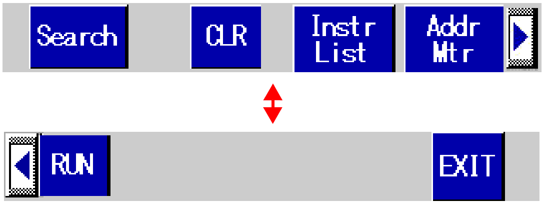
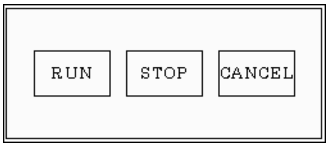
The following explains the logic monitor features.

■ Logic Monitor

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

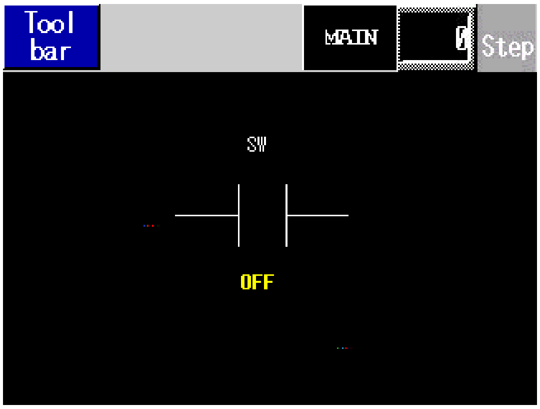


The logic monitor has the following features.

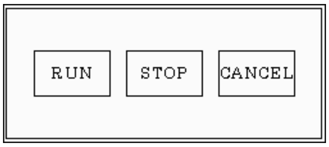
Feature	Details
Scroll ▲▼ ◀ ▶	Scroll the logic using [Rung] or [Column]. Rung: Scroll the logic using rungs. Column: Scroll the instructions one by one without the logic. For landscape, you can use only the [Column] scroll.
Enlarge Monitor	Touch the displayed instruction to enlarge the monitor. ☞ " ■ Enlarge Monitor" (page 29-93)
Logic Name Display MAIN	Display the logic names being monitored. The names to be displayed are [INIT], [MAIN], [ERRH], and [SUB-01]-[SUB-32].
Step 0 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 Band 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 EXIT	End the monitor.
RUN/STOP RUN	Switch the logic RUN/STOP. Click to display the screen below. Use the buttons to run and stop the logic. 
Address Monitor Addr Mtr	Switch to the address monitor. ☞ " ■ Address Monitor" (page 29-94)
Ladder Instructions Instr List	Switch to the instruction list. ☞ " ■ Ladder Instructions" (page 29-96)
Search Search	Search the variables and instructions specified in the instruction list. ☞ " ■ Search" (page 29-97)

■ Enlarge Monitor

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

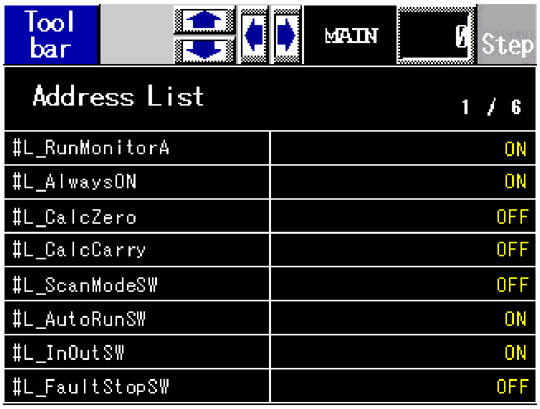


The enlarged monitor has the following features.

Feature		Details
Tool Band	Tool bar	Switch the tool bar display/hide at the bottom of the screen. Page 1 Search Address HEX Logic ▶ ↕ Page 2 ◀ RUN EXIT ▶Click /◀ to switch Page 1 with Page 2.
	End EXIT	End the monitor.
	RUN/STOP RUN	Switch the logic RUN/STOP. Click to display the screen below. Use the buttons to run and stop the logic.  RUN STOP CANCEL
	Logic Logic	Switch to the logic monitor. ☞ “■ Logic Monitor” (page 29-91)
	DEC/HEX HEX	Switch the display of the current operand value between decimal/hexadecimal.

■ Address Monitor

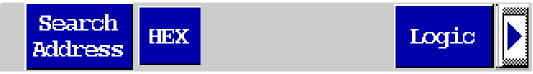
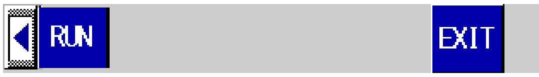
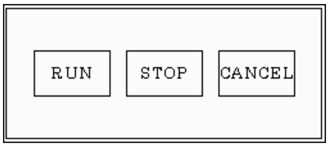
Monitor the address used in the logic. You can check the variable name and the current value. In the address format, the logic address is monitored.



Address List 1 / 6

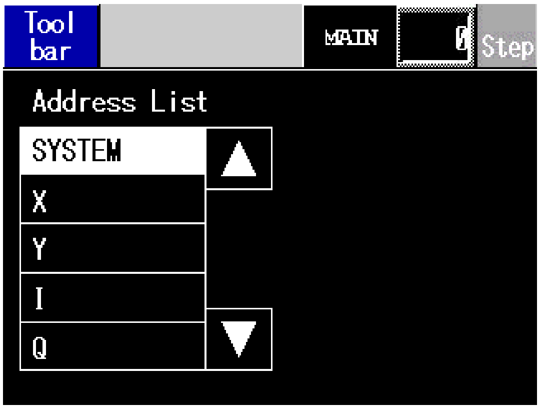
#L_RunMonitorA	ON
#L_AlwaysON	ON
#L_CalcZero	OFF
#L_CalcCarry	OFF
#L_ScanModeSW	OFF
#L_AutoRunSW	ON
#L_InOutSW	ON
#L_FaultStopSW	OFF

The address monitor has the following features.

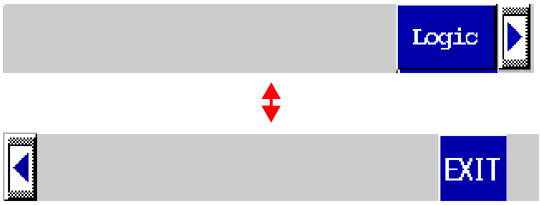
Feature		Details
Scroll	⇧⇩ ◀ ▶	Scroll the list. Up/Down: Scroll per page. Right/Left: Jump to the top/last page.
Tool Band	Tool bar	Switch the tool bar display/hide at the bottom of the screen. Page 1  Page 2  ▶Click / ◀ to switch Page 1 with Page 2.
End	EXIT	End the monitor.
RUN/STOP	RUN	Switch the logic RUN/STOP. Click to display the screen below. Use the buttons to run and stop the logic. 
DEC/HEX	HEX	Switch the display of the current operand value between decimal/hexadecimal.
Address Search	Search Address	Switch to the address search. ☞ “ ■ Address Search” (page 29-95)

■ Address Search

Select the Address Type to display in the address monitor. You can check the values stored in each address. You can use it only in address format.

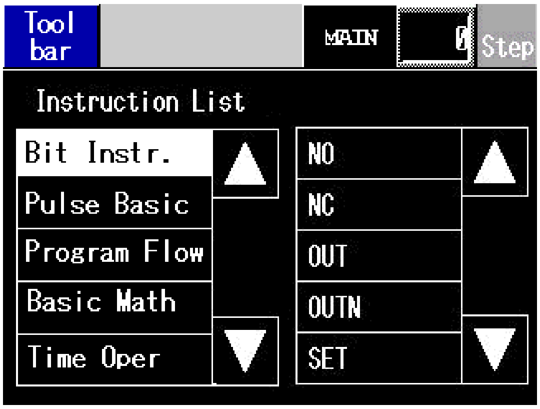


The address search has the following features.

Feature		Details
Tool Band	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	EXIT	End the monitor.
Logic	Logic	Switch to the logic monitor. ☞ “■ Logic Monitor” (page 29-91)

■ Ladder Instructions

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

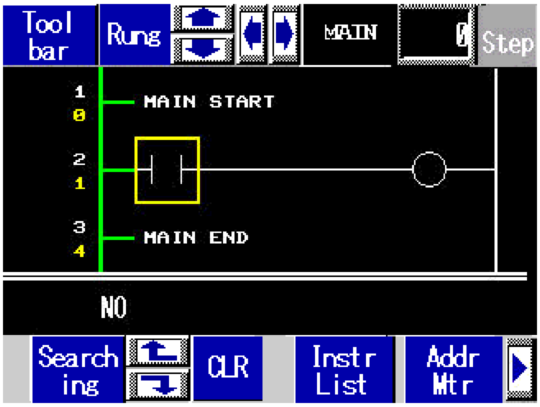


The instruction list has the following features.

Feature		Details
Tool Band Tool bar		Switch the tool bar display/hide at the bottom of the screen. Page 1 Search Address Logic ▶ ↕ Page 2 ◀ EXIT ▶Click /◀ to switch Page 1 with Page 2.
	End EXIT	End the monitor.
	Logic Logic	Switch to the logic monitor. ☞ “ ■ Logic Monitor” (page 29-91)

■ 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.
- For the numeric display of actual number variables, the value displayed on the screen may not match the internal value.

29.13 Useful Features of 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, a logic screen (e.g. MAIN) and a drawing screen (e.g. Base 1) are open in the editing area tile vertically.

NOTE

- To display the screens tile-vertically in the work space, on the [View (V)] menu, point to [Editing Area (B)] and select [Tile Vertically] or click ◫.
- You can start up multiple instances of GP-Pro EX 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.

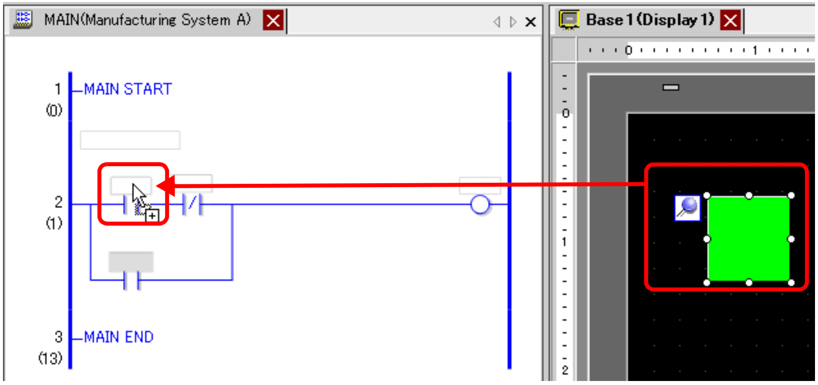
IMPORTANT

- If you drag and drop from another project, the allocated symbol variables may overlap. If you drag and drop different types of symbol variables, if the address you have set is for drawing parts, they will be undefined. Please note that in logic program, the type will be changed to match the target project. We suggest you make sure the symbol variable names do not overlap when you drag and drop.
  ☞ "29.11 Correcting Logic Program Errors" (page 29-89)
  ☞ "29.9 Transferring Logic Programs" (page 29-75)
- You cannot drag and drop between projects created in different versions.

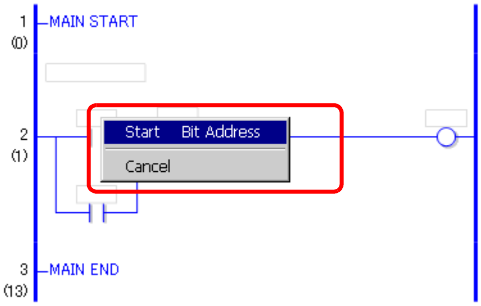
■ Allocating Symbol Variables to Instructions from Parts

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

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



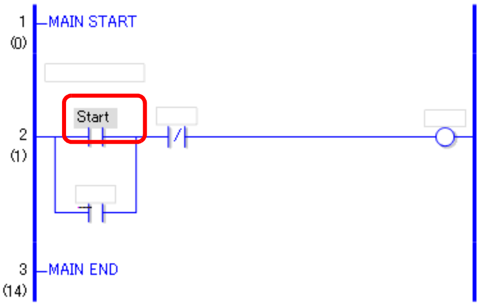
2 Select the symbol variable.



NOTE

- Symbol variables that can be allocated differ depending on parts.
  ☞ " ◆ Drag & Drop Parts" (page 29-111)
- 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 pointer is displayed with 🛇, you cannot allocate symbol variables.

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



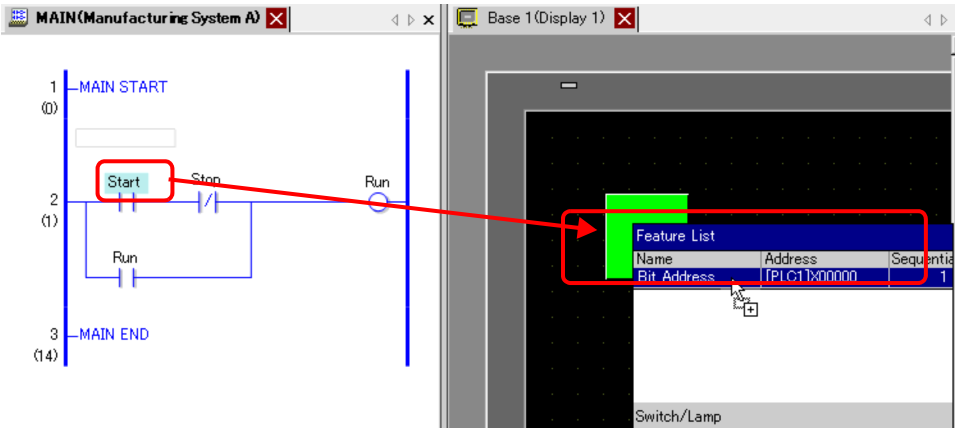
NOTE

- You can allocate symbol variables in the [Address] window.
  ☞ “ ■ Operand Settings Using Drag & Drop” (page 29-62)

■ Allocating Symbol Variables to Parts from Instructions

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

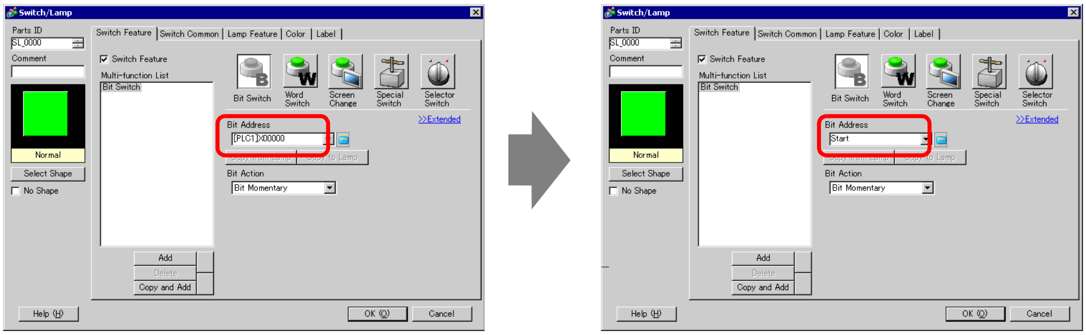
1 Click the instruction operand in the Logic. Drag it to the part on the drawing screen to which you want to allocate it. When the feature list is displayed, point to the feature to which you want to allocate it. Where 🛇 changes to ↖, release the left mouse button.



NOTE

- Symbol variables you can be allocated differ depending on instructions.
  ☞ “◆ Instructions that can be moved by a Drag & Drop Operation” (page 29-105)
- 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 pointer is displayed with 🛇, you cannot allocate symbol variables.

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



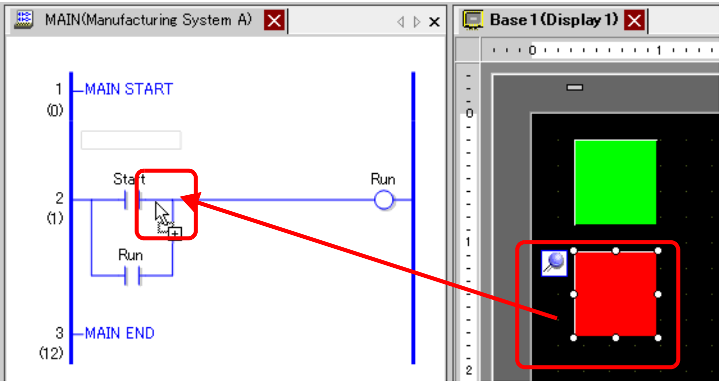
Bit Address [PLC1]X00000

Bit Address Start

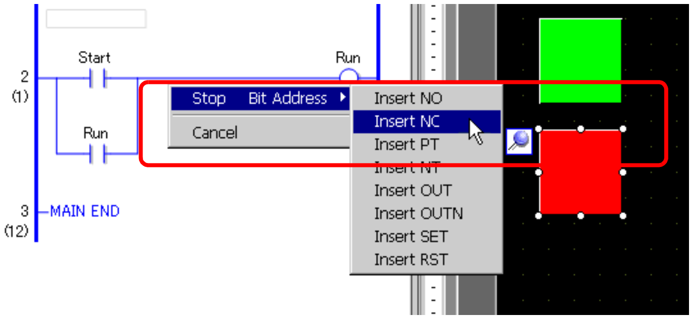
■ Inserting New Instructions from Parts

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 in the Logic. Where the pointer changes from ⃠ to the insert pointer, release the left mouse button.



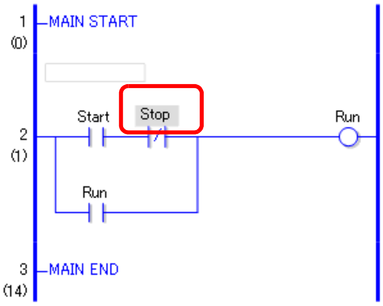
2 Select the symbol variable and then select the instruction that you want to insert.



NOTE

- Instructions that can be inserted differ depending on the parts.
  ☞ “ ◆ Drag & Drop Parts” (page 29-111)
- 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 pointer is displayed with ⃠, you cannot insert instructions.

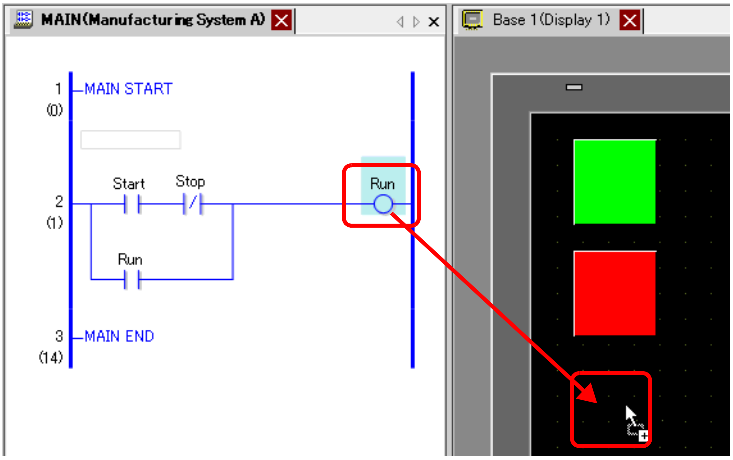
3 The instruction to which the symbol variable of the part is allocated is inserted.



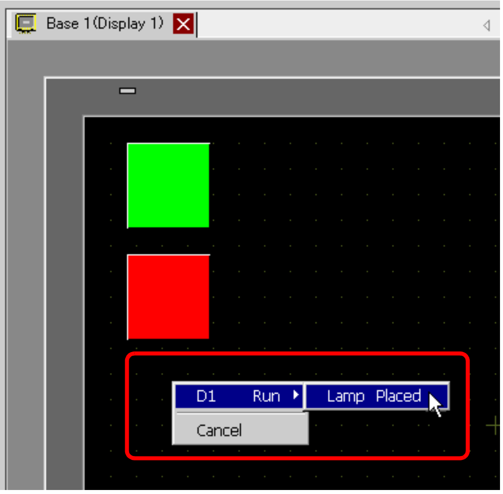
■ Placing New Parts from Instructions

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

1 Click the instruction in the Logic. Drag the instruction to where you want to place it on the drawing screen. Where ⊘ changes to ↖, release the left mouse button.



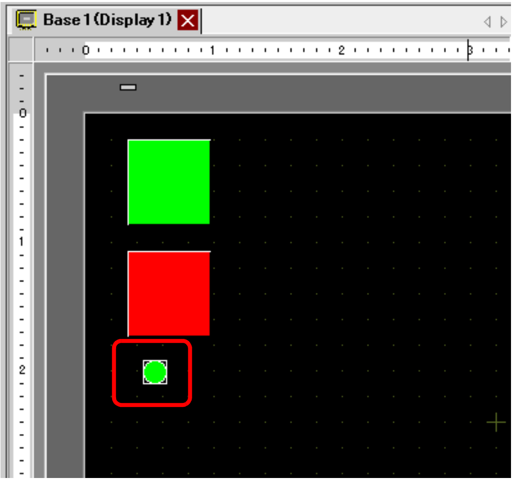
2 Select the operand and then select the part that you want to place.



NOTE

- Parts that can be placed differ depending on the instruction.
  ☞ " ◆ Instructions that can be moved by a Drag & Drop Operation" (page 29-105)
- 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 pointer 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 in the Logic and drop on the drawing screen to allocate or reallocate the symbol variables.

IMPORTANT

- Do not place real variables or instructions that have real variables as parts in operands because they cannot be displayed properly.

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 Switch
				Bit Variable	
	OUT, OUTN, SET, RST, PTO, NTO	1	D1	Bit Address	• Lamp
				Bit Variable	
	JMP, JSR	1	–	–	–
	RET, EXIT	–	–	–	–
	FOR	1	S1	Word Address	• Data Display
				Integer Variable	
	NEXT	–	–	–	–
	PBC	2	S1	–	–
			D1	Bit Variable	• Lamp
	PBR	1	S1	–	–

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	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	
				Real Variable	
			S2	Word Address	
				Integer Variable	
				Float Variable	
				Real Variable	
			D1	Word Address	
				Integer Variable	
				Float Variable	
				Real Variable	
	MOD	3	S1	Word Address	• Word Switch • Data Display
				Integer Variable	
			S2	Word Address	
				Integer Variable	
			D1	Word Address	
				Integer Variable	
	JADD, JSUB	3	S1	Time Variable (.HR .MIN	[Special Variable: Integer Variable] • Word Switch • Data Display
			S2	Time Variable (.HR .MIN	
			D1	Time Variable (.HR .MIN	
	AND, OR, XOR	3	S1	Word Address	• 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	• Word Switch • Data Display
				Integer Variable	
			D1	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	BLMV	3	S1	Bit Address (Array)	• Word Switch • Data 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	• 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	• Word Switch • Data Display
				Integer Variable	
			D2	Word Address	
				Integer Variable	
	ROL, ROR, RCL, RCR, SHL, SHR, SAL, SAR	3	S1	Word Address	• Word Switch • Data Display
				Integer Variable	
			S2	Word Address	
				Integer Variable	
			D1	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	
Compare Instruction	EQ, GT, GE, LT, LE, NE	2	S1	Word Address	• Word Switch • Data Display (You cannot select the Word switch for float variables and real variables.)
				Integer Variable	
				Float Variable	
				Real Variable	
			S2	Word Address	
				Integer Variable	
				Float Variable	
				Real Variable	
	JEQ, JGT, JGE, JLT, JLE, JNE	2	S1	Time Variable (.HR .MIN .SEC)	[Special Variable: Integer Variable] • Word Switch • Data Display
			S2	Time Variable (.HR .MIN .SEC)	
	NEQ, NGT, NGE, NLT, NLE, NNE	2	S1	Date Variable (.YR .MO .DAY)	[Special Variable: Integer Variable] • Word Switch • Data Display
			S2	Date Variable (.YR .MO .DAY)	
Timer Instruction	TON, TOF, TP, TONA, TOFA	1	Special Variable	Timer Variable (.ET .PT)	[Special Variable: Integer Variable] • Word Switch • Data Display
				Timer Variable (.Q .TI .R)	[Special Variable: Bit Variable] • Bit Switch • Lamp
Counter Instruction	CTU, CTD, CTUD	1	Special Variable	Counter Variable (.PV .CV)	[Special Variable: Integer Variable] • Word Switch • Data Display
				Counter Variable (.Q .QD .QU .UP .R)	[Special Variable: Bit Variable] • Bit Switch • Lamp

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	
Convert Instruction	BCD, BINENCO, DECO	2	S1	Word Address	• Word Switch • Data Display
				Integer Variable	
			D1	Word Address	
				Integer Variable	
	RAD, DEG, SCL	2	S1	Word Address	• Word Switch • Data Display (You cannot select the Word switch for float variables and real variables.)
				Integer Variable	
				Float Variable	
				Real Variable	
			D1	Word Address	
				Integer Variable	
				Float Variable	
				Real Variable	
	I2F, I2R, F2I, F2R, R2I, R2F, H2S, S2H	2	S1	Type	
				–	
			D1	Type	
Function Instruction	SUM, AVE	3	S1	Integer Variable (Array)	• Word Switch • Data Display (You cannot select S1.)
				Float Variable (Array)	
				Real Variable (Array)	
			S2	Integer Variable	
			D1	Integer Variable	
				Float Variable	
				Real Variable	
	SQRT	2	S1	Float Variable	• 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)	

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	
Function Instruction	PID	5	Special Variable	PID Variable (.KP .TR .TD .PA .BA .ST)	[Special Variable: Integer Variable] • Word Switch • Data Display
				PID Variable (.Q .UO .TO .PF .IF)	[Special Variable: Bit Variable] • Bit Switch • Lamp
			S1	Word Address	• 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	• 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] • Word Switch • Data Display
	NRD, NSET	1	D1	Date Variable (.YR .MO .DAY)	[Special Variable: Integer Variable] • Word Switch • Data Display

◆ Drag & Drop Parts

The following are operands that you can allocate symbol variables or instructions that you can insert, by dragging and dropping parts from the drawing screen to the logic screen.

Drag & 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, SET, RST, PTO, NTO	
		Bit Variable		
	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, SET, RST, PTO, NTO	
		Bit Variable		
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.

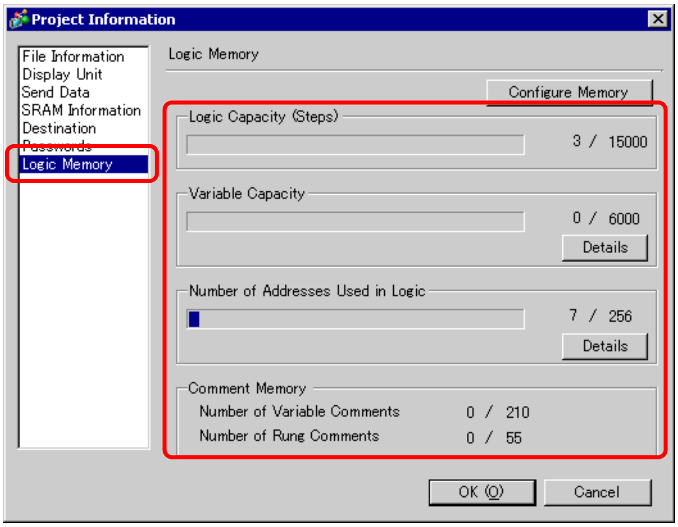
NOTE

- The logic capacity is calculated as the sum of the logic and the comments in the logic. The variable capacity is the sum of the variables and the variable comments.

■ Confirming the Logic Memory

1 Select the [Project (F)] menu, [Information (I)], then [Project Information (I)].
The [Project Information] dialog box is displayed.

2 Click [Logic Memory] to check [Logic Capacity], [Variable Capacity], [Number of Addresses used in Logic] and [Comment Memory].



Logic Capacity (Steps) 3 / 15000

Variable Capacity 0 / 6000

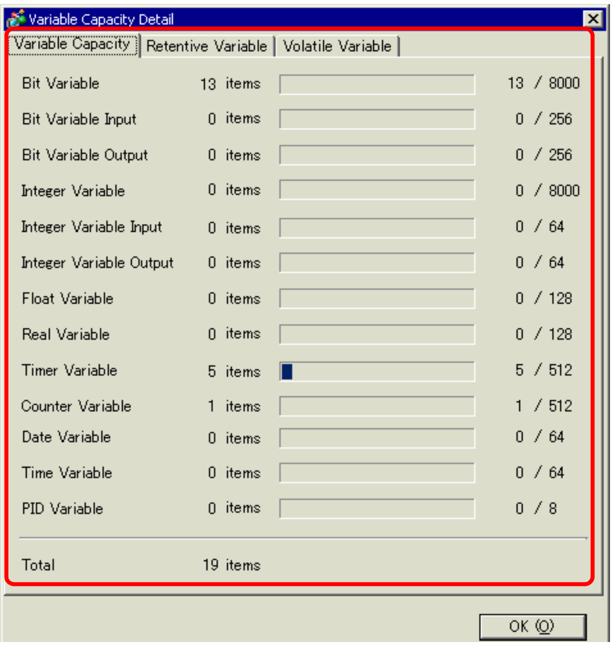
Number of Addresses Used in Logic 7 / 256

Comment Memory

Number of Variable Comments 0 / 210

Number of Rung Comments 0 / 55

3 From [Variable Capacity], click [Details] to check the current number, assignable number and the current sum for each symbol variable.



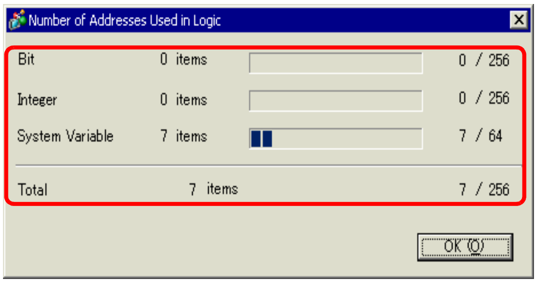
Variable Capacity Detail

Bit Variable	13 items	13 / 8000
Bit Variable Input	0 items	0 / 256
Bit Variable Output	0 items	0 / 256
Integer Variable	0 items	0 / 8000
Integer Variable Input	0 items	0 / 64
Integer Variable Output	0 items	0 / 64
Float Variable	0 items	0 / 128
Real Variable	0 items	0 / 128
Timer Variable	5 items	5 / 512
Counter Variable	1 items	1 / 512
Date Variable	0 items	0 / 64
Time Variable	0 items	0 / 64
PID Variable	0 items	0 / 8
Total	19 items	

NOTE

- You can choose to display [Retentive Variable]/[Volatile Variable] by clicking the tab.

4 Click the [Number of Addresses used in Logic] details to check the current number, the configurable number of [Bit], [Integer] and [System Variable] and the total number.



Number of Addresses Used in Logic

Bit	0 items	0 / 256
Integer	0 items	0 / 256
System Variable	7 items	7 / 64
Total	7 items	7 / 256

NOTE

- [Number of Addresses Used in Logic] is the number of external addresses of [PLC1]****and internal addresses of [USER]. Please note that the number of addresses that can be used in a logic program is limited.

■ Configure Memory

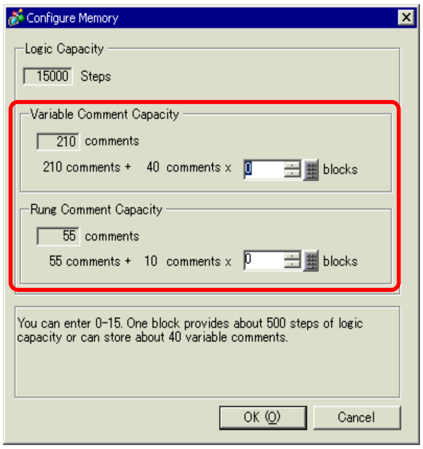
You can specify the upper capacity limit for the symbol variable comments and rung comments.

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.

1 In the [Project Information] dialog box, click [Configure Memory]. The [Configure Memory] dialog box appears.

2 Specify the upper comment memory limit for the symbol variable and rung comment capacity within the range of 0-15.



Configure Memory

Logic Capacity

15000 Steps

Variable Comment Capacity

210 comments

210 comments + 40 comments x 0 blocks

Rung Comment Capacity

55 comments

55 comments + 10 comments x 0 blocks

You can enter 0-15. One block provides about 500 steps of logic capacity or can store about 40 variable comments.

■ 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	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	96KByte
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	64KByte
Number of NT/PT Instructions	Unlimited (depends on the number of programs)	None
Number of Constants	Unlimited (depends on the number of programs)	None
Number of Compulsory Changes	Unlimited (depends on the number of programs)	None
Array Size	4096 arrays	None
Number of Variables	9000 Symbol Variables 6000 Variables	1MByte
Variable Name	32 Characters	
Number of Variable Comments	210 (Default value)	16KByte →14KByte
Number of Variable Comment Characters	32 Characters	
Number of Rung Comments	55 comments (initial value)	16KByte →14KByte
Characters in Rung Comments	128 characters	
Number of Program Comments	34	8KBytes
Characters in Program Comments	32 Characters	
Number of Nests	25 Nests	Stack: 16 (32)

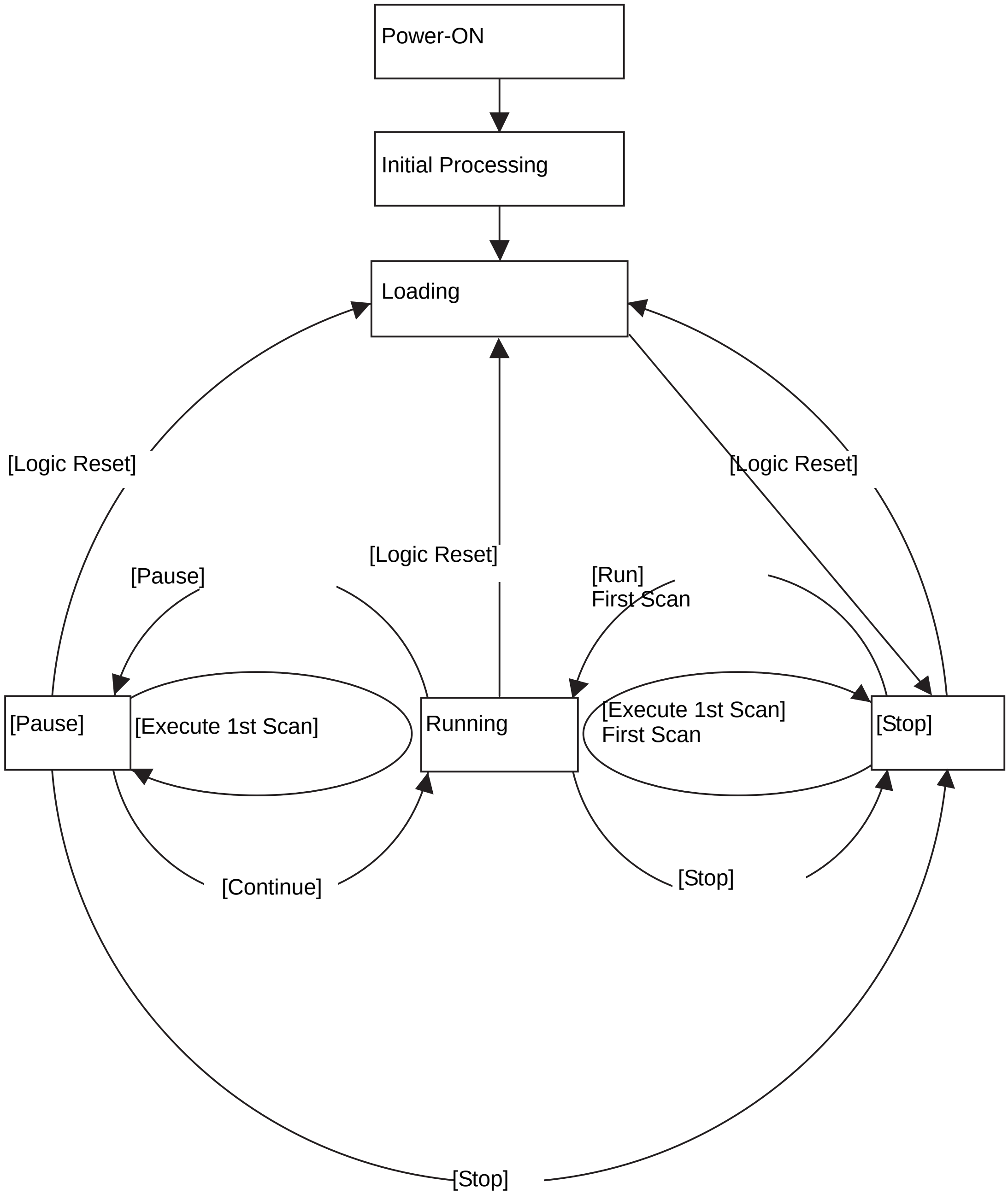
29.13.3 Adjusting Logic Scan Time

The following provides an overview of logic functions and the scan time when the logic program is run. The steps to set the scan time are also provided.

■ 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 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 retentive 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 variable is initialized with the initial value for logic features in 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.

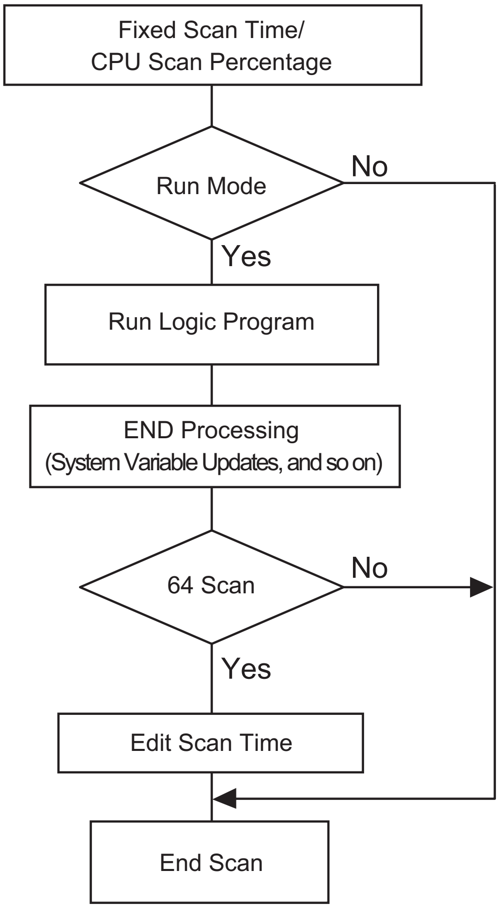
- First Scan
  Reads I/O, runs the initialized logic program, and then writes the I/O.

- 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 available model GP33∗∗ Series: Models other than 30: 50)

- CPU Scan Percentage Mode

  Scan Time = (#L_AvgLogicTime × 100) ÷ #L_PercentAlloc

NOTE

- For the details of #L_AvgLogicTime, #L_PercentAlloc, refer to the following.

  ☞ “A.6 System Variables” (page A-84)

- Scan Time Error

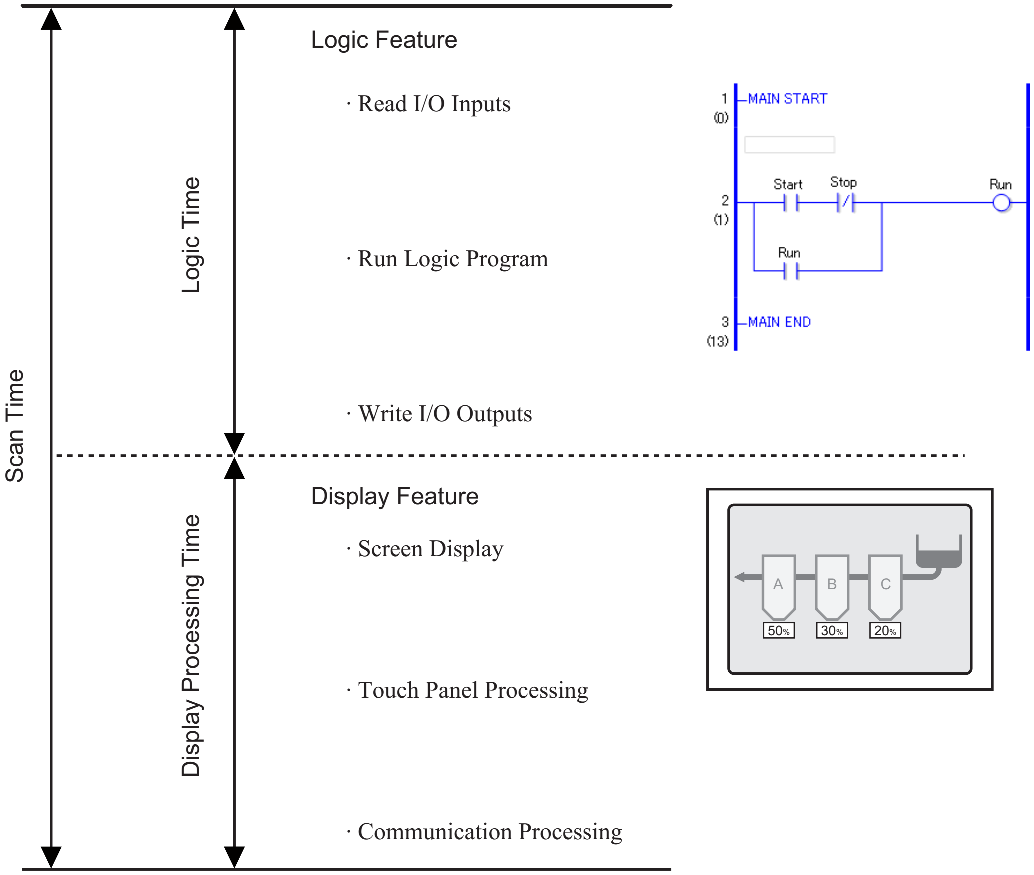
  The following error is found in the logic scan time.

Model	Error
AGP3000 Series	10%

* Communicating via Ethernet or MPI may affect the scan time.
For details, refer to “29.15 Restrictions” (page 29-144) .

■ 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

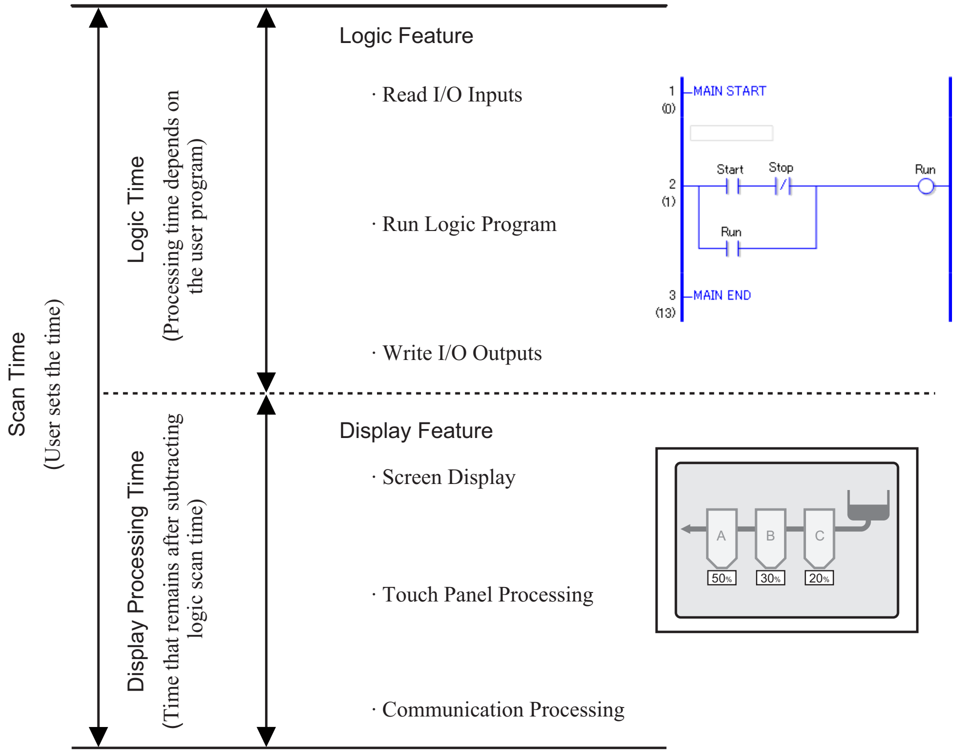
- Updating the device/PLC addresses depends on Address Refresh, and is not affected by the fixed scan time or CPU scan percentage.
  For details on address refresh, refer to “ ■ Address Refresh” (page 29-123) .

  ☞ “ ◆ Fixed Scan Time” (page 29-120)

  ☞ “ ◆ CPU Scan Percentage” (page 29-121)

◆ 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

For example, 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 adjusted to be twice as long as the logic time.
  For example) When the fixed scan time is set to 50ms:
  And the logic time is 30ms, the scan time is 60ms.

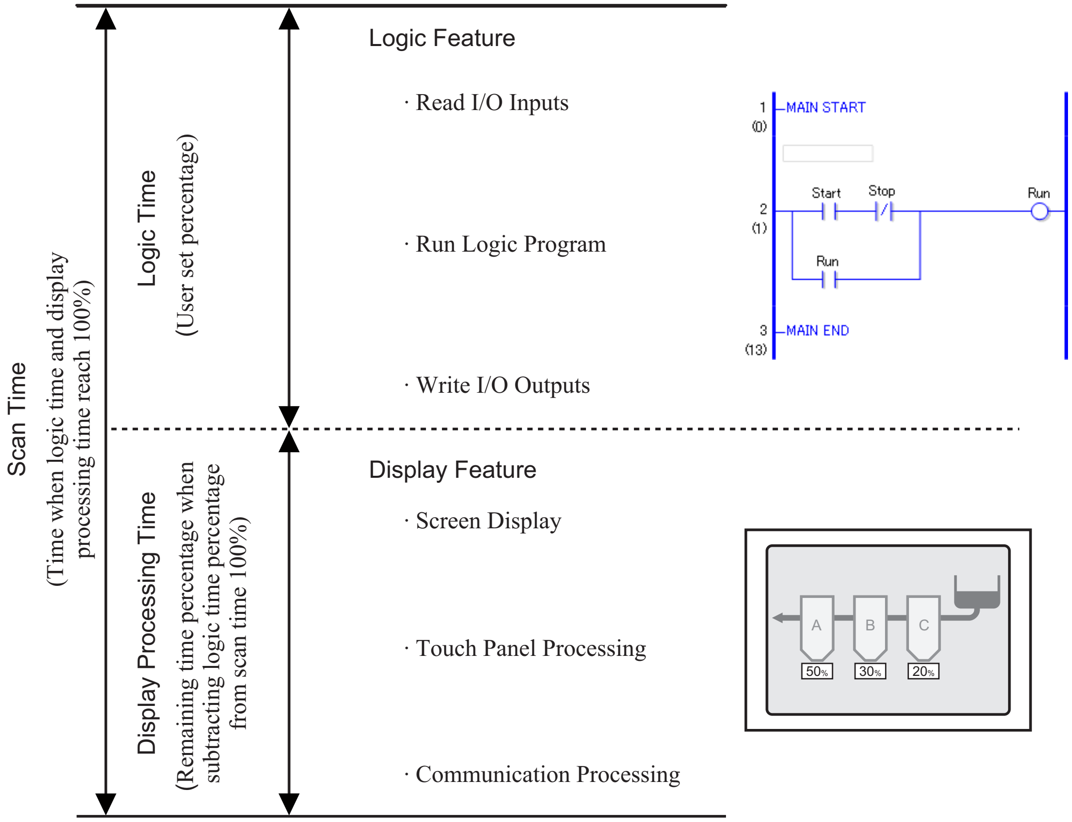
NOTE

- Adjust the setting time based on the #L_AvgScanTime value after testing the operation on the GP.
  ☞ “A.6 System Variables” (page A-84)

◆ 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 (%)

For example, If 40% is specified for the CPU scan percentage and the logic executing time is 20 ms,

Scan Time = (20 ÷ 40) × 100 = 50 ms

Display Processing Time = 50 ms – 20 ms = 30 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

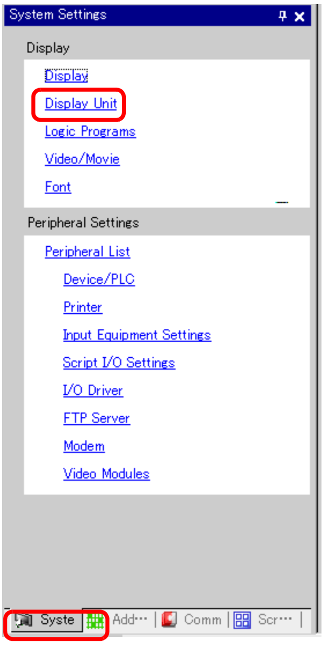
- Specify the scan time value for the CPU scan percentage by 1 ms increments.
- The processing time per instruction in the logic program remains unchanged.
- You cannot specify a CPU scan percentage larger than 50%.
- If 50% is specified for the CPU scan percentage, the display processing time and logic 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.
  ☞ "5.14.6 [System Settings] Setting Guide ◆ Logic" (page 5-134)

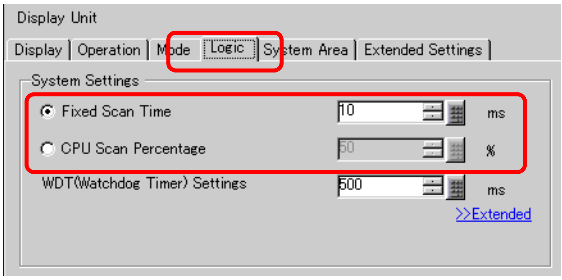
1 In the [System Settings], click [Display Unit].



NOTE

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

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



Display Unit

System Settings

● Fixed Scan Time 10 ms

○ CPU Scan Percentage 50 %

WDT(Watchdog Timer) Settings 500 ms

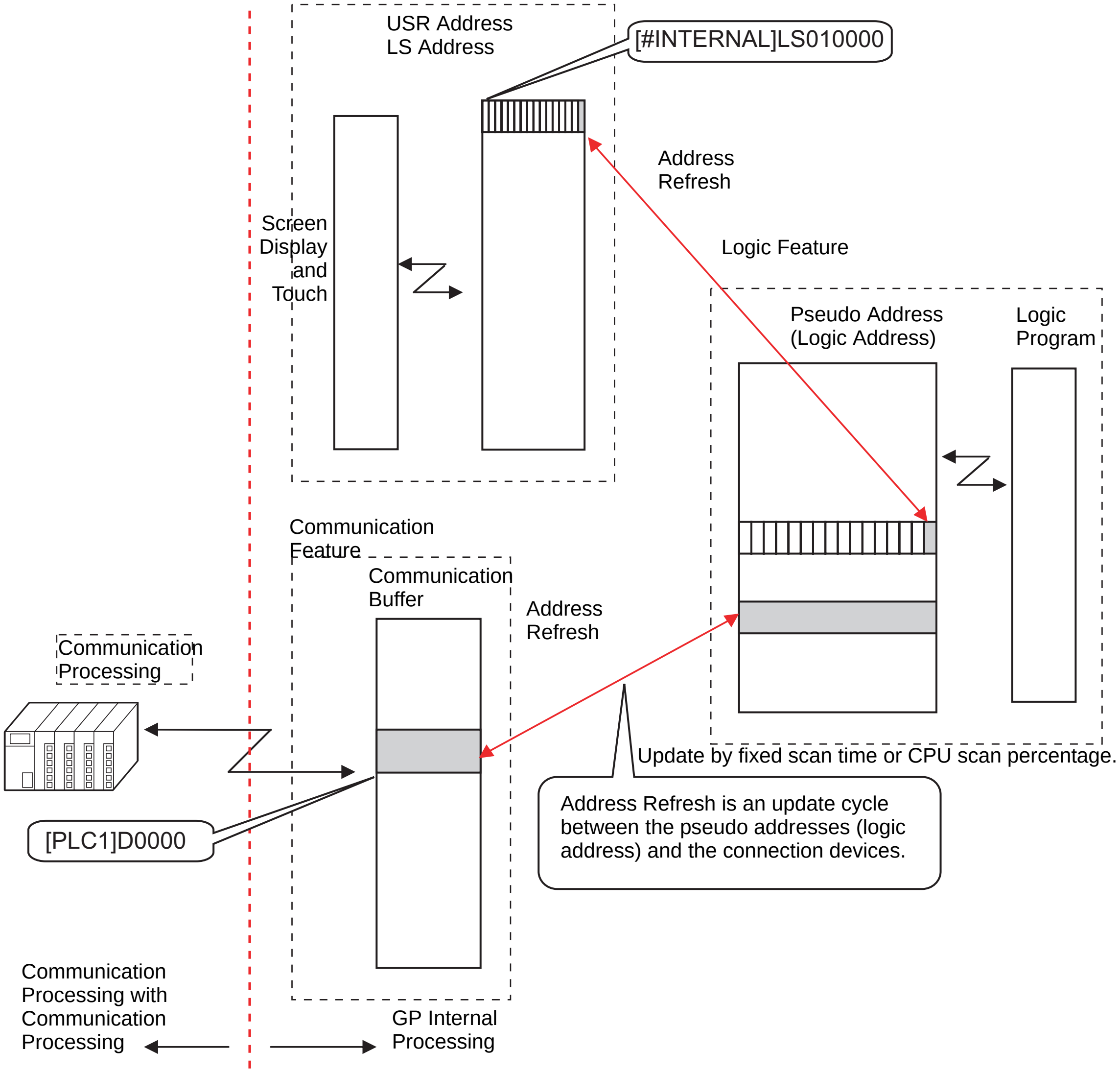
■ Address Refresh

◆ 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.

You can specify the update interval depending on the user system.


USR Address
LS Address
[#INTERNAL]LS010000
Address Refresh
Screen Display and Touch
Logic Feature
Pseudo Address (Logic Address)
Logic Program
Communication Feature
Communication Buffer
Address Refresh
Communication Processing
[PLC1]D0000
Update by fixed scan time or CPU scan percentage.
Address Refresh is an update cycle between the pseudo addresses (logic address) and the connection devices.
Communication Processing with Communication Processing
GP Internal Processing

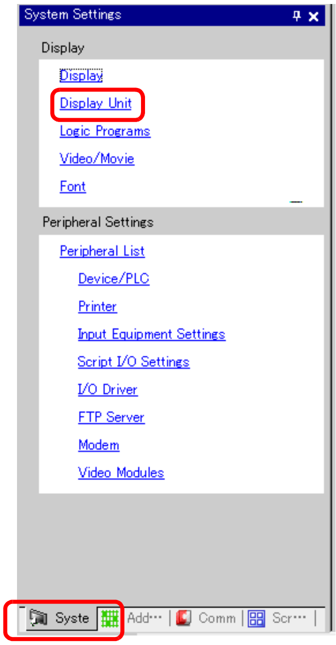
◆ Setup Procedure

You can choose the address refresh update from fast, medium or slow.

IMPORTANT
- Update interval is not a fixed value as it is affected by the user system. The actual update interval is stored in (#L_AddressRefreshTime). Adjust the system variable to select fast, medium or slow for the update interval.
- The update speed of the screen may be affected because the address refresh update interval is shorter.

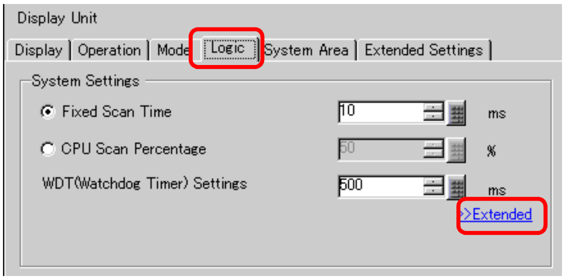
NOTE
- Please refer to the settings guide for details.
  ☞ "5.14.6 [System Settings] Setting Guide ◆ Logic" (page 5-134)

1 In the [System Settings], click [Display Unit].



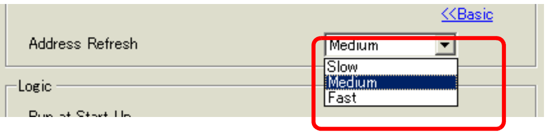
NOTE
- If the [System Settings] tab is not displayed in the work space, on the [View (V)] menu point to [Work Space (W)] and then click [System Settings (S)].

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



System Settings
- ● Fixed Scan Time 10 ms
- ○ CPU Scan Percentage 50 %
- WDT(Watchdog Timer) Settings 500 ms

3 In [Address Refresh], select the speed.

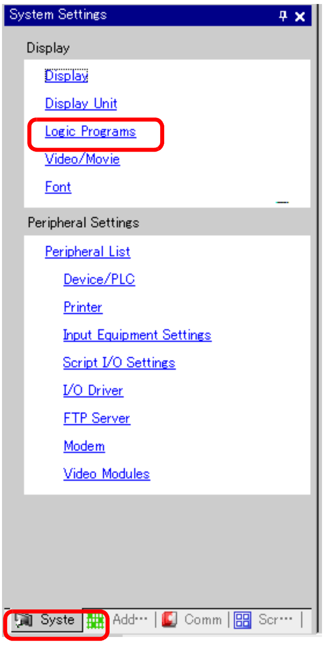


29.13.4 Enhancing Security

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

■ Setup Procedure

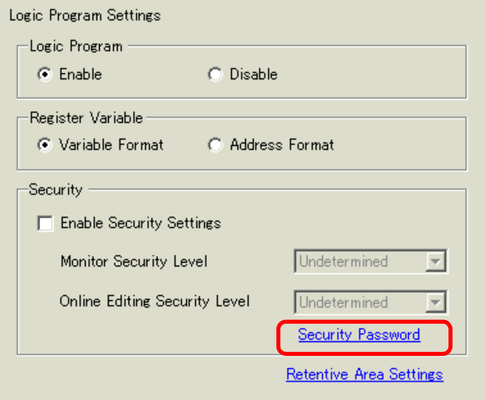
1 In the [System Settings], click [Logic Programs].



NOTE

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

2 Click [Security Password].



Logic Program Settings

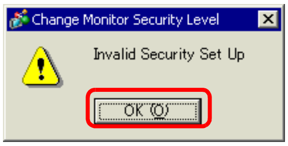
Logic Program
● Enable ○ Disable

Register Variable
● Variable Format ○ Address Format

Security
□ Enable Security Settings
Monitor Security Level: Undetermined
Online Editing Security Level: Undetermined

NOTE

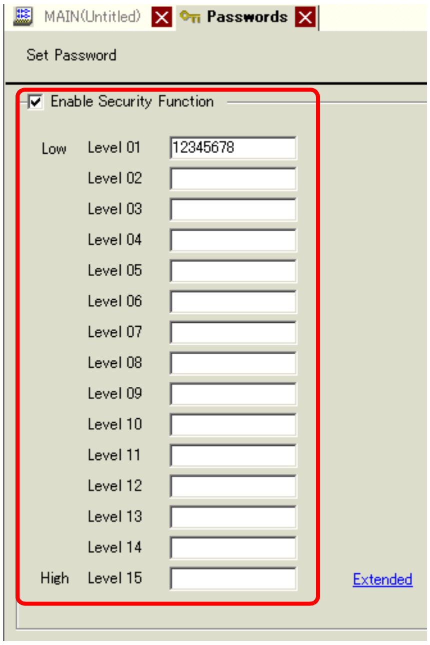
- If you select a [Monitor Security Level] without setting up a security password, the following dialog box appears. Click [OK] and set up a security password.



Change Monitor Security Level

Invalid Security Set Up

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



Set Password

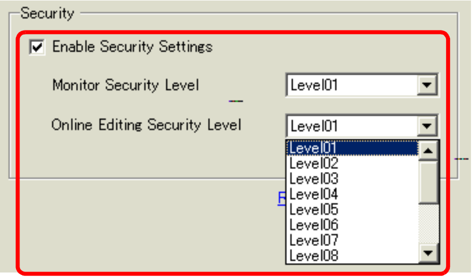
☑ Enable Security Function

Low	Level 01	12345678
	Level 02	
	Level 03	
	Level 04	
	Level 05	
	Level 06	
	Level 07	
	Level 08	
	Level 09	
	Level 10	
	Level 11	
	Level 12	
	Level 13	
	Level 14	
High	Level 15	

NOTE

- You can configure advanced security feature settings. For details about the settings, refer to the settings guide.
  ☞ "22.5.1 Password Settings" (page 22-9)

4 In the [Security] area select the [Enable Security Settings] check box, and then define the [Monitor Security Level] and [Online Editing Security Level].



Security

☑ Enable Security Settings

Monitor Security Level: Level01

Online Editing Security Level: Level01

29.13.5 Using Reference Features to Search Logic Programs

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

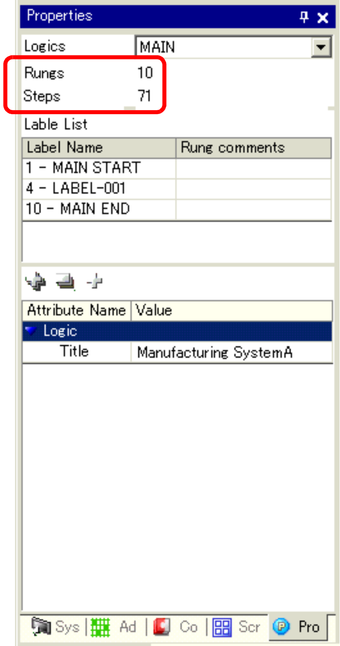
NOTE

- Please refer to the settings guide for details.
  ☞ "5.14.5 [Work Space] Settings Guide ■ Screen List" (page 5-103)

■ [Properties] Display

1 In [Logic], open the logic screen you want to display.

2 Select [View(V)], [Work Space(W)], and click [Properties(P)].
The total numbers of rungs and steps are displayed.



Properties

Logics: MAIN

Rungs: 10

Steps: 71

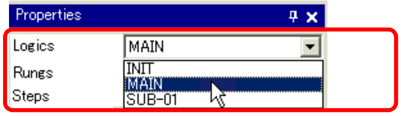
Lable List

Label Name	Rung comments
1 - MAIN START	
4 - LABEL-001	
10 - MAIN END	

Attribute Name	Value
Logic	
Title	Manufacturing SystemA

NOTE

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



Properties

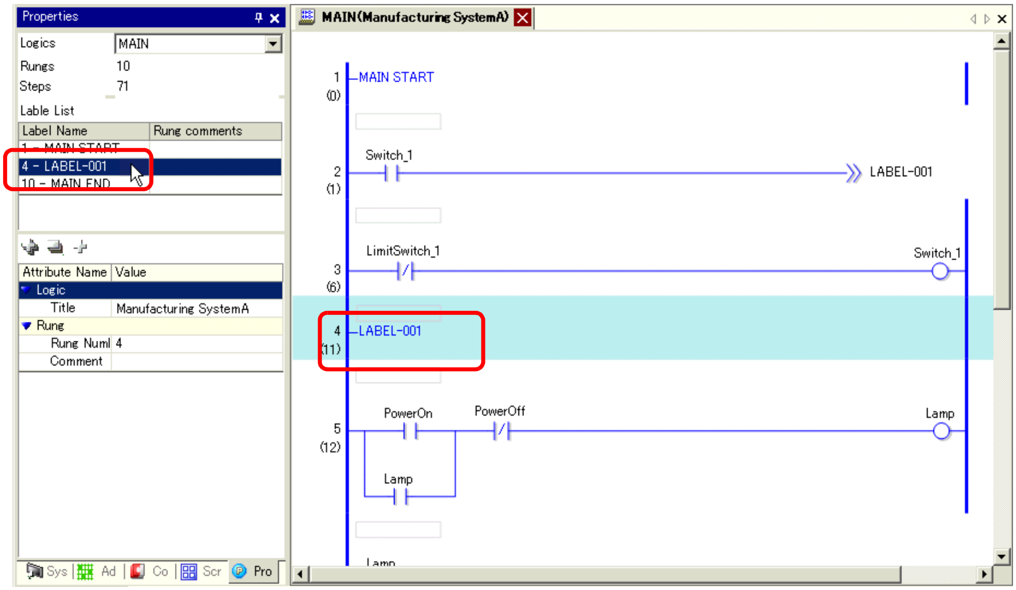
Logics: MAIN

Rungs

Steps

■ Properties Search

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



Properties

Logics: MAIN

Rungs: 10

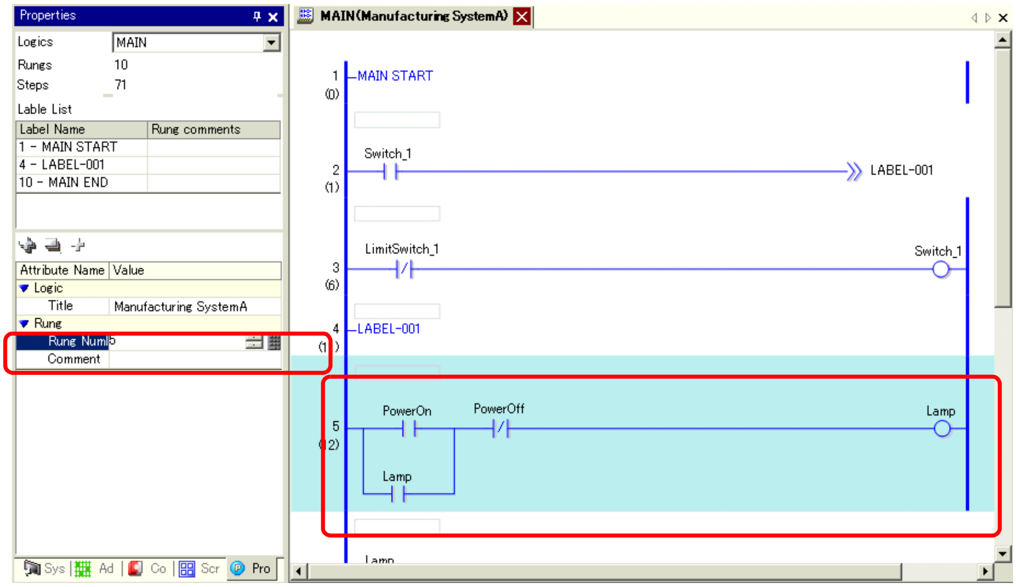
Steps: 71

Lable List

Label Name	Rung comments
1 - MAIN START	
4 - LABEL-001	
10 - MAIN END	

Attribute Name	Value
Logic	
Title	Manufacturing SystemA
Rung	
Rung Numl	4
Comment	

- Click [Rung Number] and enter a number to move the cursor to that rung.



Properties

Logics: MAIN

Rungs: 10

Steps: 71

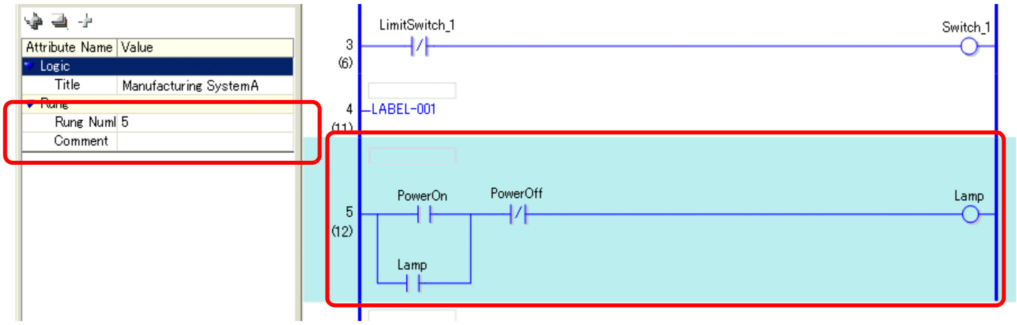
Lable List

Label Name	Rung comments
1 - MAIN START	
4 - LABEL-001	
10 - MAIN END	

Attribute Name	Value
Logic	
Title	Manufacturing SystemA
Rung	
Rung Numl	5
Comment	

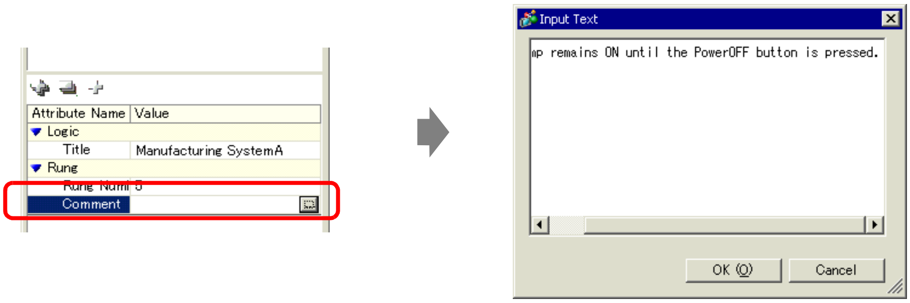
■ Display and Edit the Logic Program Information in Program Windows

- Select a rung in the logic program to display [Rung Number] and [Comment] for the selected rung.



Attribute Name	Value
Logic	
Title	Manufacturing SystemA
Rung	
Rung Numl	5
Comment	

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

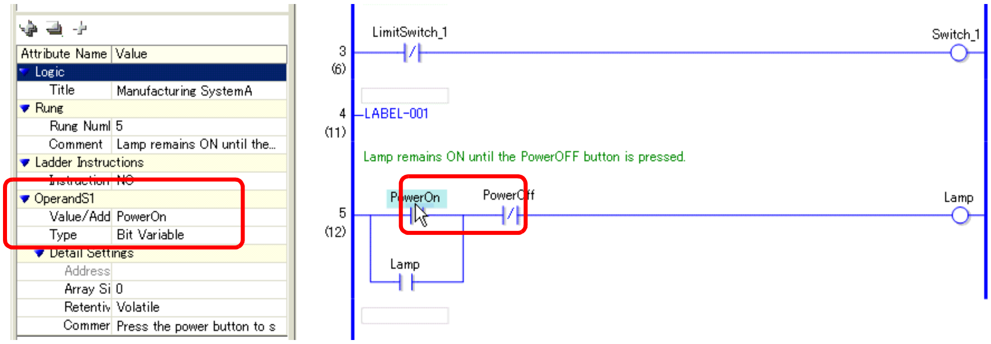


Attribute Name	Value
Logic	
Title	Manufacturing SystemA
Rung	
Rung Numl	5
Comment	

Input Text

mp remains ON until the PowerOFF button is pressed.

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



Attribute Name	Value
Logic	
Title	Manufacturing SystemA
Rung	
Rung Numl	5
Comment	Lamp remains ON until the...
Ladder Instructions	
Instruction	NO
OperandS1	
Value/Add	PowerOn
Type	Bit Variable
Detail Settings	
Address	
Array Si	0
Retentiv	Volatile
Commer	Press the power button to s

Lamp remains ON until the PowerOFF button is pressed.

NOTE

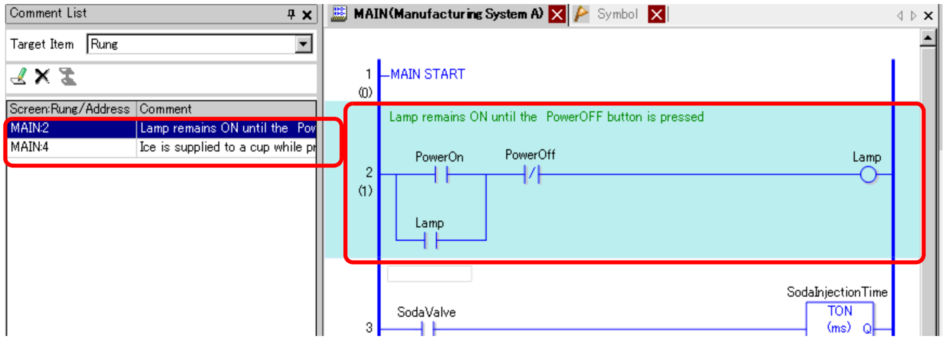
- If the detailed settings are not displayed, click ⇩ (or ▸ located to the left of [Detail Settings]) to display [Address], [Array Size], [Retentive], and [Comment]. The settings that you can edit differ depending on [Type].

■ Searching Method for the Comment List

NOTE

- For how to display the Comment List, refer to the following.
  ☞ "29.7.4 [Comment List] Window" (page 29-70)

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



Comment List

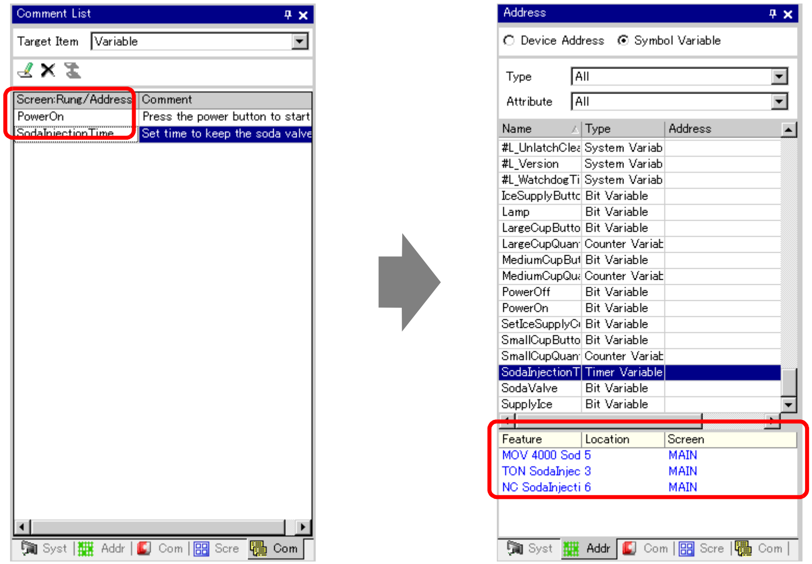
Target Item: Rung

Screen:Rung/Address	Comment
MAIN:2	Lamp remains ON until the Pow
MAIN:4	Ice is supplied to a cup while pr

MAIN(Manufacturing System A)

Lamp remains ON until the PowerOFF button is pressed

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



Comment List

Target Item: Variable

Screen:Rung/Address	Comment
PowerOn	Press the power button to start
SodaInjectionTime	Set time to keep the soda valve

Address

○ Device Address ● Symbol Variable

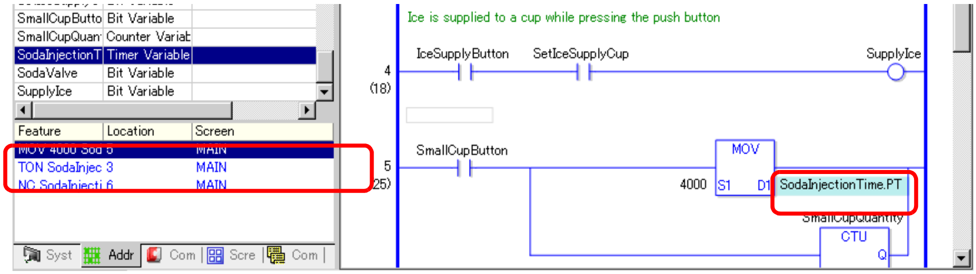
Type: All

Attribute: All

Name	Type	Address
#L_UnlatchClea	System Variab	
#L_Version	System Variab	
#L_WatchdogTi	System Variab	
IceSupplyButto	Bit Variable	
Lamp	Bit Variable	
LargeCupButto	Bit Variable	
LargeCupQuan	Counter Variab	
MediumCupBut	Bit Variable	
MediumCupQu	Counter Variab	
PowerOff	Bit Variable	
PowerOn	Bit Variable	
SetIceSupplyC	Bit Variable	
SmallCupButto	Bit Variable	
SmallCupQuan	Counter Variab	
SodaInjectionT	Timer Variable	
SodaValve	Bit Variable	
SupplyIce	Bit Variable	

Feature	Location	Screen
MOV 4000 Sod	5	MAIN
TON SodaInjec	3	MAIN
NC SodaInjecti	6	MAIN

Select the lower part of the [Address] window to select the target variable on the logic screen.



SmallCupButto	Bit Variable	
SmallCupQuan	Counter Variab	
SodaInjectionT	Timer Variable	
SodaValve	Bit Variable	
SupplyIce	Bit Variable	

Feature	Location	Screen
MOV 4000 Sod	5	MAIN
TON SodaInjec	3	MAIN
NC SodaInjecti	6	MAIN

Ice is supplied to a cup while pressing the push button

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

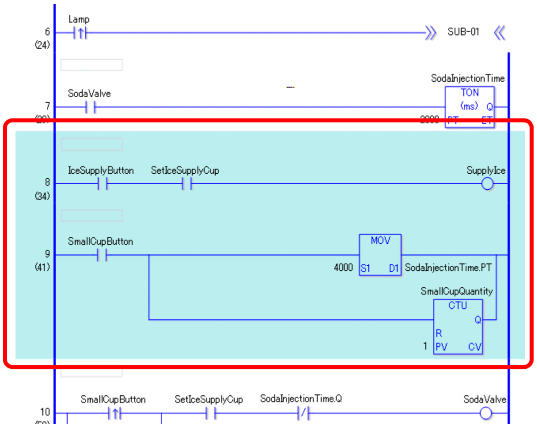
- Logic parts are saved in the specified folder. As a result, only the computer where logic parts have been registered or saved can load the logic parts.

■ Registering Logic Parts

◆ Registering Rung Parts

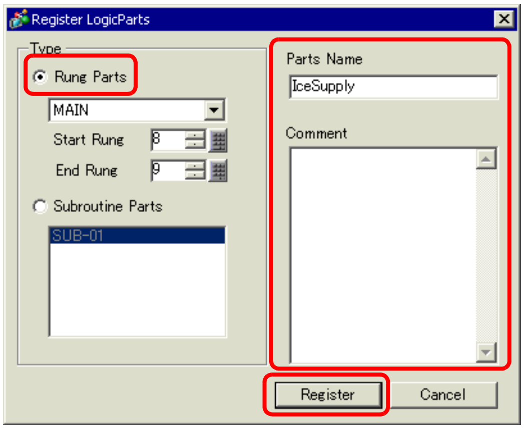
Registering a part of a rung as a logic part.

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



2 Select [Logic (L)], [Save Parts (S)]. The [Register Logic Parts] is displayed.

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


Register LogicParts

Type

● Rung Parts

MAIN

Start Rung 8

End Rung 9

○ Subroutine Parts

SUB-01

Parts Name IceSupply

Comment

NOTE

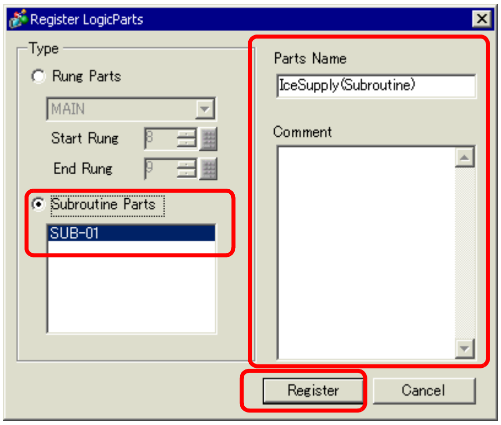
- If the [Register Logic Parts] dialog box appears with no rung selected, you can select a logic program and input [Start Rung] and [End Rung] to specify the range.

◆ Registering Subroutine Parts

Registering a subroutine program as a logic part.

1 Select [Logic (L)], [Save Parts (S)]. The [Register Logic Parts] is displayed.

2 In [Type], select [Subroutine Parts] and the subroutine name. Enter the [Parts Name] and [Comment] and click [Register].



Register LogicParts

Type

○ Rung Parts

MAIN

Start Rung 8

End Rung 9

● Subroutine Parts

SUB-01

Parts Name: IceSupply(Subroutine)

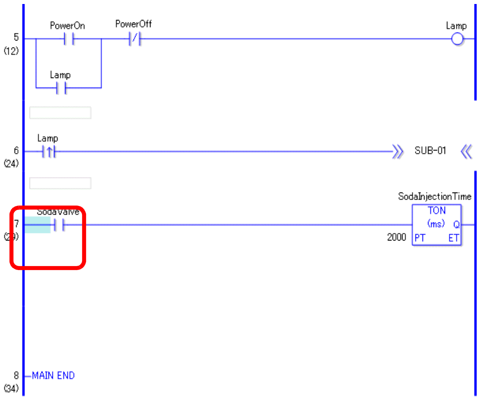
Comment

■ Calling Logic Parts

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

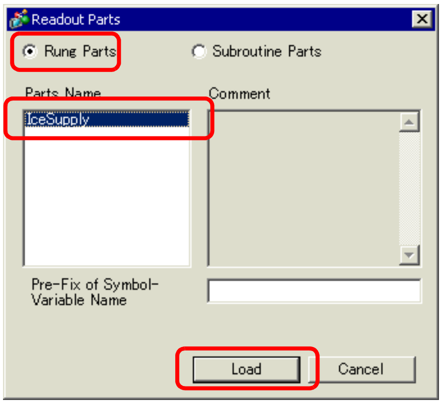
◆ Inserting Rung Parts in Logic Programs

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 Select [Logic (L)], [Load Parts (P)]. The [Load Parts] dialog box is displayed.

3 Select [Rung Parts] and choose the rungs parts to be inserted from [Parts Name] and click [Load].

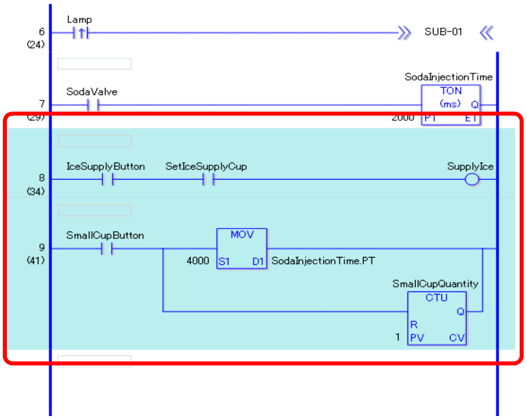


NOTE

- You can avoid overlapping the symbol variable names by inputting text in [Pre-Fix of Symbol-Variable Name].
  If a logic program is loaded when the symbol variables are overlapped, the variable type may be changed.
  Is that case, the input text is added in front of the symbol variable name of the rung parts, then inserted into the logic program you are editing.
  For example, when entering "ALine" in [Pre-Fix of Symbol-Variable Name]:

	Rung Parts	**After Insertion of the Logic Program**
Symbol Variable Name	Switch1	ALineSwitch1

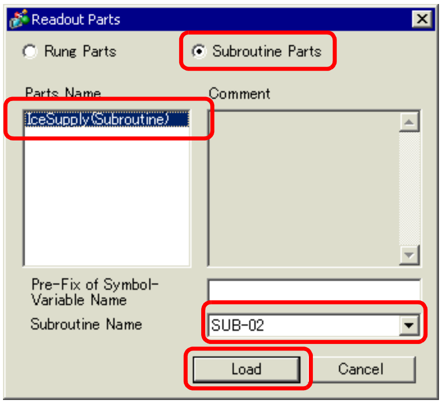
4 The rung part is inserted.



◆ Add Subroutine Parts

1 Select [Logic (L)], [Load Parts (P)]. The [Load Parts] dialog box is displayed.

2 Select [Subroutine Parts], then select [Parts Names], [Subroutine Name], then click [Load].



Readout Parts

○ Rung Parts ● Subroutine Parts

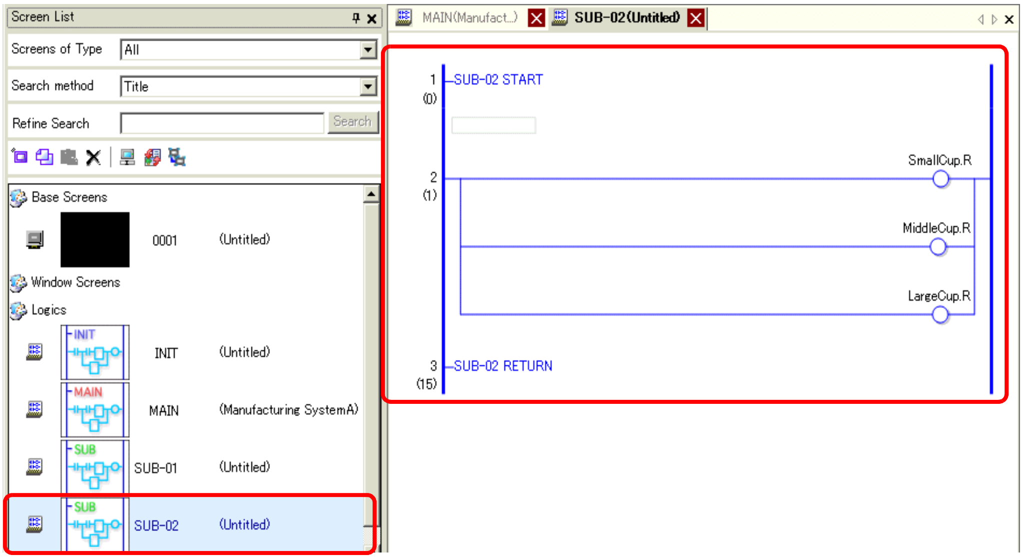
Parts Name: IceSupply(Subroutine)

Comment

Pre-Fix of Symbol-Variable Name

Subroutine Name: SUB-02

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



Screen List

Screens of Type: All

Search method: Title

Refine Search

Base Screens		
	0001	(Untitled)
Window Screens		
Logics		
INIT	INIT	(Untitled)
MAIN	MAIN	(Manufacturing SystemA)
SUB	SUB-01	(Untitled)
SUB	SUB-02	(Untitled)

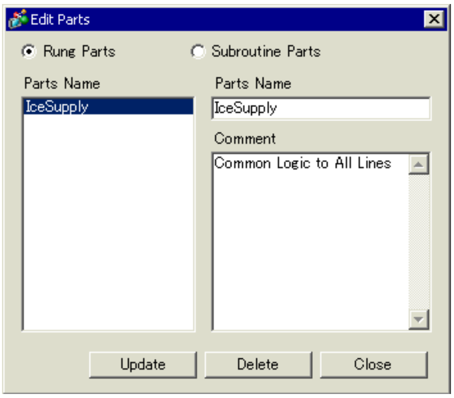
■ Editing Logic Parts

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

1 From the [Logic (L)] menu, select [Edit Parts (E)]. The [Edit Parts] dialog box is displayed.

2 Select [Rung Parts] to edit rung parts, and select [Subroutine Parts] to edit subroutine parts. Then click the [Parts Name] you want to edit.

3 To edit [Parts Name] or [Comment], modify the text in [Parts Name] or [Comment] and then click [Update].


Edit Parts

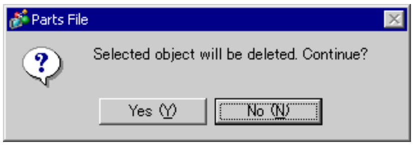
● Rung Parts ○ Subroutine Parts

Parts Name: IceSupply

Parts Name: IceSupply

Comment: Common Logic to All Lines

To delete parts, click [Delete]. When the following dialog box appears, click [Yes].


Parts File

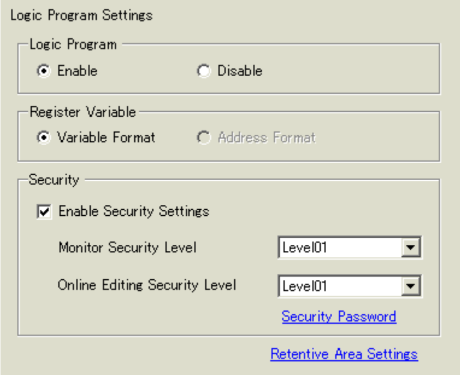
Selected object will be deleted. Continue?

29.14 Settings Guide

29.14.1 [Logic Programs] Setting Guide

NOTE

- You can define the logic scan time settings in the System Settings [Display Unit]'s [Logic] tab.
  ☞ "5.14.6 [System Settings] Setting Guide ◆ Logic" (page 5-134)



Logic Program Settings

Logic Program
● Enable ○ Disable

Register Variable
● Variable Format ○ Address Format

Security
☑ Enable Security Settings
Monitor Security Level: Level01
Online Editing Security Level: Level01
Security Password

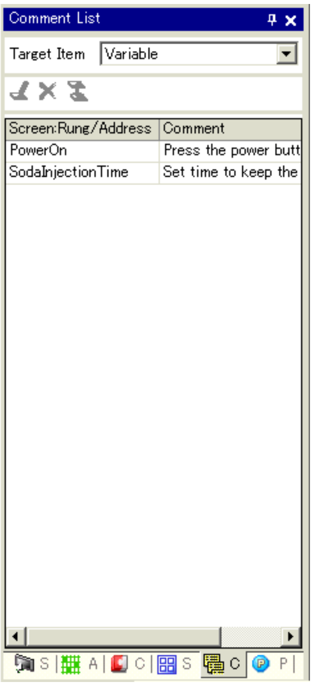
Retentive Area Settings

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 29-30)
Security		Sets up the password for online monitoring. To use this feature, from the [Common Settings (R)] menu, point to [Security] and select [Security Password]. In the [Set Password] page, select [Enable Security Function.
	Enable Security Settings	Check to enable the Security Settings.
	Monitor Security Level	Select the level of monitoring security. The settings range from [Level 01] to [Level 15.]
	Online Editing Security Level	Select a level for online edit security. The settings range from [Undefined], [Level 01]-[Level 15].
	Security Passwords	Click [Security Password] to switch to [Set Password]. Set a password for each security level. ☞ "22.5.1 Password Settings" (page 22-9)
Retentive Settings		Click [Retentive Settings]. The [Retentive Settings] dialog box appears. The retentive/volatile points can be specified for symbol variables in [Variable Format]. The retentive/volatile ranges can be specified for symbol variables in [Address Format]. ☞ " ■ Retentive Settings" (page 29-16)

29.14.2 [Work Space] Settings Guide

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

■ Comment List Window



Comment List

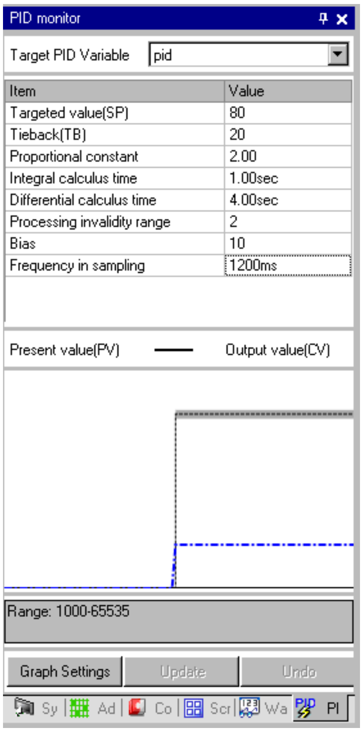
Target Item: Variable

Screen:Rung/Address	Comment
PowerOn	Press the power butt
SodaInjectionTime	Set time to keep the

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 appears only in [Address Format], selected in [Register Variable]. You can specify addresses and add logic addresses. ☞ “ ■ Logic Address Display” (page 29-33)
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] window and the relevant variable will be selected. The logic name and rung number are displayed in [Rungs]. Double-click to select the target rung in the logic program.
Comment		The comment for the selected rung is displayed. Double click to edit [Variable] and [Rung].

■ PID Monitor Window

In Monitor Step, for PID instructions used on the Logic screen, you can check the operation of PID values, and even change each parameter.



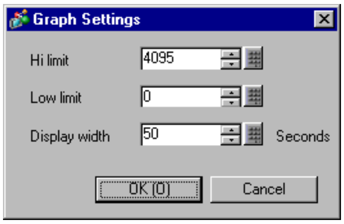
PID monitor

Target PID Variable: pid

Item	Value
Targeted value(SP)	80
Tieback(TB)	20
Proportional constant	2.00
Integral calculus time	1.00sec
Differential calculus time	4.00sec
Processing invalidity range	2
Bias	10
Frequency in sampling	1200ms

Present value(PV) —— Output value(CV)

Range: 1000-65535

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	You can specify the details of the graph. Click and the settings dialog box appears.  Graph Settings Hi limit: 4095 Low limit: 0 Display width: 50 Seconds OK (O) Cancel
Displayed Items	Select the check box to display [Current Value], [Target Value], [Output Value], [Output Invalidity Range], or [Output Range].
Graph Display Range	Specify [Upper Limit], [Lower Limit] and [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

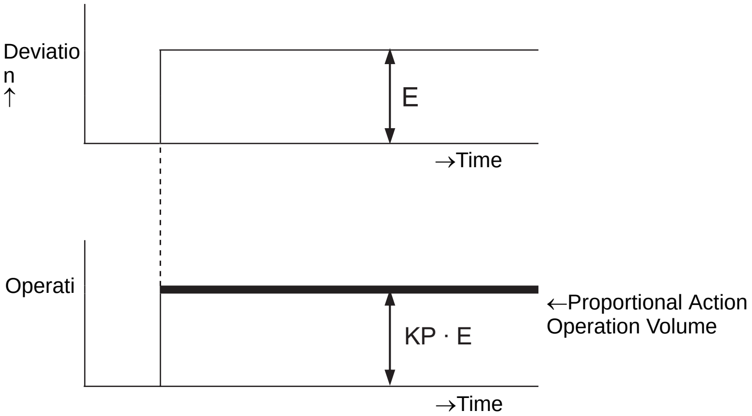
Item	Value
Targeted Value (SP)	Specify the target value. Enter values for the minimum and maximum output. The range depends on the PID instruction output settings. For details, refer to the PID instructions. ☞ “Chapter 30 Ladder Instructions” (page 30-1)
Tieback (TB)	Specify a value for output during power off. The range for input depends on the PID instruction output settings. For details, refer to the PID instructions. ☞ “Chapter 30 Ladder Instructions” (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 Operation (P)” (page 29-140)
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 29-140)
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 29-141)
Processing Deadband Range	Specifies the range in which PID instructions are not operated. The deviation in the settings range is "0", and the processing deadband range is based on ± from the Set Point. 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 65535 milliseconds (ms).

◆ Proportional Operation (P)

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 · E (KP is the proportional gain.)

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



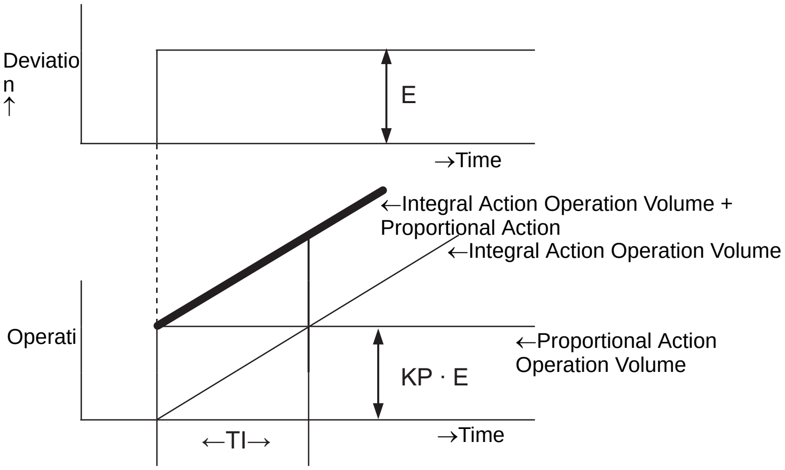
The operation volume changes within the range of 0-4095 (initial value). As KP increases, the operation volume proportionate to the deviation increases and the correcting action strengthens, and this causes offset (residual deviation).

◆ 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.

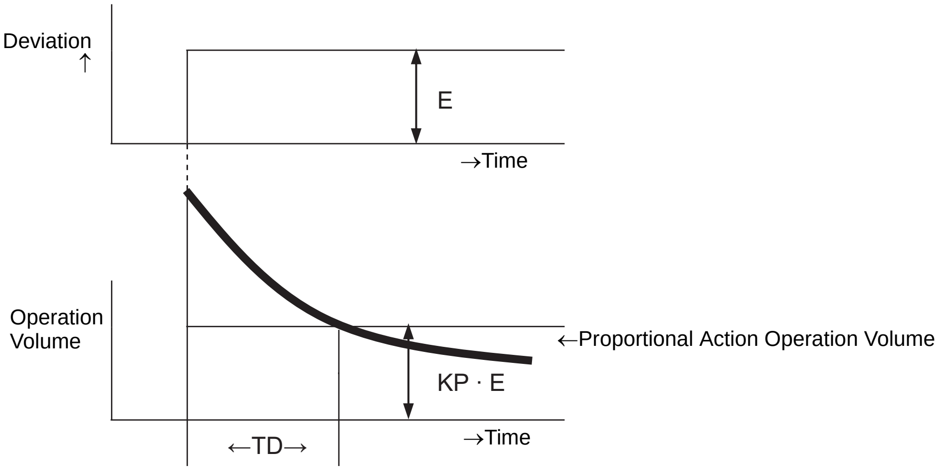
If the deviation is fixed, the integral action is as follows.



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)

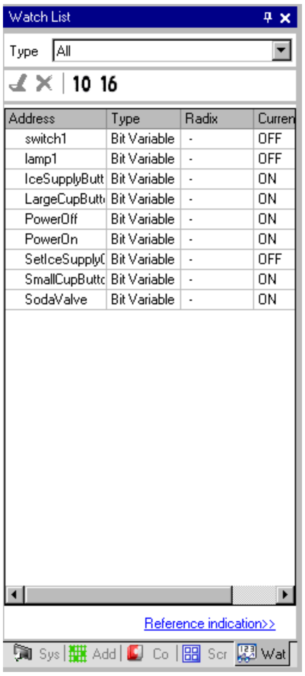
Add the operation volume (output value) proportionate to any deviation (deviation between the target value and current value) to eliminate deviation. Doing so prevents the control target from drastically changing due to an external disturbance.
Once deviation has occurred in the derivative action, the 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 Window

In Monitor Step, the Watch List displays the current value of symbol variables. The Watch List is useful for debugging purposes because you can edit values of symbol variables in the list.



Address	Type	Radix	Curren
switch1	Bit Variable	-	OFF
lamp1	Bit Variable	-	OFF
IceSupplyButt	Bit Variable	-	ON
LargeCupButt	Bit Variable	-	ON
PowerOff	Bit Variable	-	ON
PowerOn	Bit Variable	-	ON
SetIceSupply(	Bit Variable	-	OFF
SmallCupButtc	Bit Variable	-	ON
SodaValve	Bit Variable	-	ON

Setting		Description
Type		Select the type of symbol variable or system variable registered in the [Watch List] window. For how to register, refer to the following. ☞ "29.10.2 Monitor/Change the Current Value of Symbol Variables" (page 29-79)
Operation Button	Edit ✎	In Monitor Step, you can edit the display format and value of registered symbol variables.
	Delete ✕	In Monitor Step, you can remove symbol variables from the watch list.
	Decimal 10	In Monitor Step, changes the display to decimal format.
	Hexadecimal 16	In Monitor Step, changes 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		In Monitor Step, the current value added to the watch list is displayed. If the type is [Bit Variable], right-click and then click [ON], [OFF], [Force ON], or [Force OFF]. If the type is [Integer Variable], [Float Variable], or [Real Variable], right-click to input the value.

Setting	Description
Example Display	In Monitor Step, 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 a logic program is "enabled," a maximum 6% delay may occur temporarily.
- When communicating a large volume of data (for example sequential address = 960 Words) on a PLC over Ethernet (for example Mitsubishi Electric's Q Series), a maximum 30% delay may occur.
- When sending and receiving data using AGP Ethernet, take the scan time delay into consideration.
- When data is communicated with a PLC (for example Mitsubishi Q Series) that has several Ethernet connections, a max 100% delay may occur.
- When a large volume of data (for example 10KBytes) is communicated using Pro-Server EX (our product), a maximum 100% scan time delay may occur. When accessing memory for a large volume of data (for example, 10KBytes) with the Pro-Server EX, take the scan time delay into consideration.
- When data is communicated with the MPI protocol, a maximum 30% delay may occur.

■ AGP-3400/3500/3600/3750 Series

- When communicating a large volume of data (sequential address = 960 Words) on a PLC that uses Ethernet (Mitsubishi Electric's Q Series), a maximum 15% delay may occur. When sending and receiving data with AGP Ethernet, take the scan time delay into consideration.
- When a large volume of data (for example 10KBytes) is communicated using Pro-Server EX (our product), a maximum 20% scan time delay may occur. When accessing memory for a large volume of data (for example, 10KBytes) with the Pro-Server EX, take the scan time delay into consideration.
- When data is communicated with MPI protocol, a max of 15% delay may occur.
- When movies are recorded or played on an FTP server that has multimedia functions, a max of 15% delay may occur.
- No scan time can be guaranteed when a program is being uploaded.
- No scan time can be guaranteed when a CF card being read.
- When an error arises in the logic or the I/O driver, the scan time is delayed by approximately 10ms.
- When many devices are connected to a LAN, the scan time may be delayed.

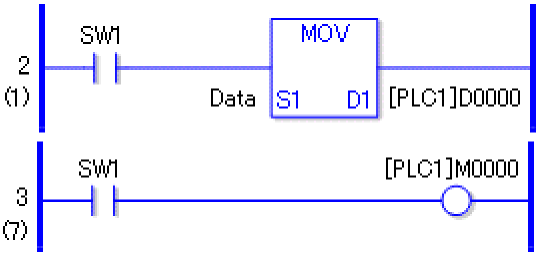
  It is recommended to physically separate LAN into a control system LAN and an information system LAN, etc.

  To be more specific, have two LAN cards ready on a PC and configure the control system LAN that AGP belongs to on one card and configure the information system LAN on the other. In other words, separate the LAN into two groups on the PC.
- When movies are recorded/played using the multimedia function while the logic is in use, movie recording/playback may stop.
- Please note that data updates between a device/PLC and the logic program are not synchronized when the device/PLC address (excluding internal addresses) is used in the

logic program. The data value is sometimes undefined when the logic program starts, and is not updated until communication with the device/PLC is established. Check that the device/PLC address data has been read before using it in the logic program.
For example, A special relay (always on) is used for a device/PLC.
Connection Device: Special relay (always on)
Logic program: Use the special relay in the logic program and check that the special relay is on before using the device/PLC address. If there are several devices/PLCs, a different relay is required for each device.

- If the logic program includes operations for writing to the device/PLC address, the following phenomenon may occur, depending on the communication speed and the number of device/PLC addresses to write to:
  - When the GP starts up, parts set up with the device/PLC address do not display.
  - Change screen operations take extra time.
  - Writing to the device/PLC fails.

  This phenomenon occurs due to excess write operations from the logic program to the device/PLC.
  To avoid this problem, do the following:
  - Increase the transmission speed with the device/PLC.
  - Adjust the number of addresses to write to the device/PLC. Adjust the number of addresses used to write to the device/PLC. You can use operand (D) to specify device/PLC addresses.



The number of addresses that you can reference is as follows.

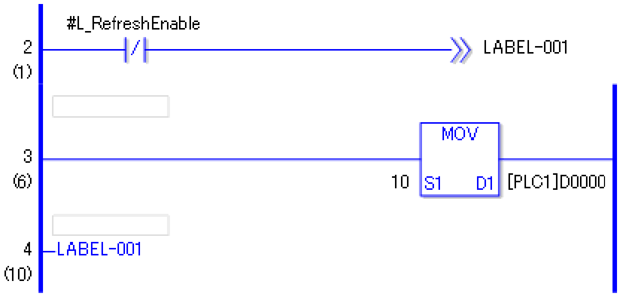
For example, when updating data in the logic every 10ms, in the following system use a maximum value of 20.

Device/PLC: MELSEC FX

Communication speed: 115200bps

Address updates: Medium speed

  - By using the system variable "#L_RefreshEnable" in the logic program, you can control write operations to the device/PLC so that they occur only after communication with the device/PLC has been confirmed.


2
(1)
#L_RefreshEnable
LABEL-001
3
(6)
MOV
10
S1
D1
[PLC1]D0000
4
(10)
LABEL-001