e-Front runners

MICREX-SX series

Guide to MICREX-SX

~ Using simulator function of D300winV3 ~

Preface

Thank you for purchasing Fuji Electric Programmable Controller MICREX-SX Series.
This document is a guide to MICREX-SX series conforming to IEC61131-3, international standards of programmable controllers. This guide is designed to help you acquire the basic knowledge about MICREX-SX series and master the basic operations of loader software D300winV3, by actually using the simulator function of D300winV3.
In addition to this guide, please read the corresponding user's manuals listed below, which will deepen your understanding.

Title	Manual No.	Contents
User's Manual Instructions, MICREX-SX series	FEH200	Explains the memory, language and system definitions of the MICREX-SX series.
User's Manual Hardware, MICREX-SX series SPH	FEH201	Explains the system configuration, thehardware specifications and operations of modules in the MICREX-SX series.
User's Manual D300win <Reference>, MICREX-SX series	FEH257	Explains the menu and icon of D300winV3 and all of the operations of D300winV3
User's Manual D300win <LD/FBD Editor Operations> MICREX-SX series	FEH257-1	Explains the operating instruction of the LD/FBD editor which is added to D300winV3 as new function.
Guide to Constructing a Redundant MICREX-SX System MICREX-SX series	FEH253-2	Explains the method of MICREX-SX redundancy system to improve system reliability.
Guide to ST Language MICREX-SX series	FEH253-3	Self-study guide for high-level language (ST language) used for FA.

* User's manuals listed above can be downloaded from the following Fuji Electric FA Components & Systems Co., Ltd. site. In addition to them, Fuji site offers various manuals and technical documents associated with MICREX-SX.
URL http://www.fujielectric.co.jp/fcs/eng/

Notes

1. The contents of this manual (including specifications) are subject to change without prior notice.
2. If you find any ambiguous or incorrect descriptions in this manual, please write them down (along with the manual No. shown on the cover) and contact FUJI.

Revision

*The manual No. is printed at the bottom right of the cover of this manual.

Printed on	*Manual No.	Revision contents
May 2006	FEH253-1	First edition

Contents

Preface

Revision

Contents

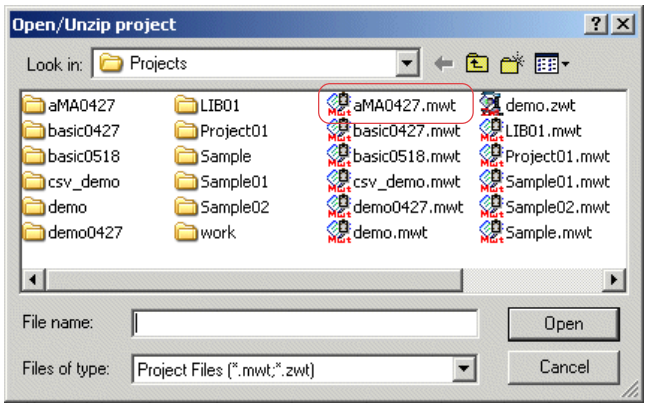
Section 1 Creating a New Project

1-1 Project Name

When you newly create an application program using D300win, give a “project name” to it first.
(Name a new project with the [Save Project As / Zip Project As] command. As the first step in creating a new project or changing the project, form a habit of naming a project.)
A project name can be up to 24 characters.

[FYI]

By prefixing the name of a project under development with “a”, you can open it with ease.
When multiple projects exist in the [Open/Unzip project] dialog as shown below, a project given the prefix “a” is displayed at the top of the list of projects.



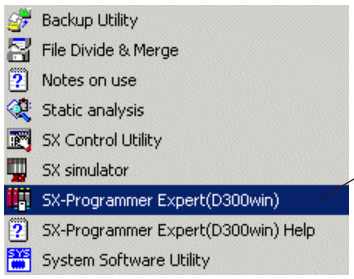
* Projects are sorted in the order of numbers → alphabets. Naturally, project names beginning with a number are displayed at the top of the list. However, it is advised not to prefix project names with a number because there are some restrictions; variable names cannot begin with a number, for example.

Section 1 Creating a New Project

1-1 Project Name

<Operational procedure>

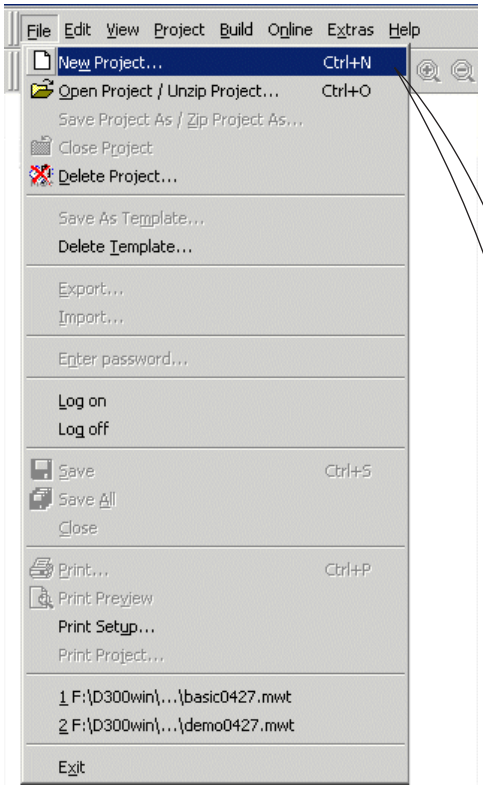
◆ Click [SX-Programmer Expert(D300win)] in the [SX-Programmer Expert(D300win)] menu in the Windows Start menu to start up D300win.

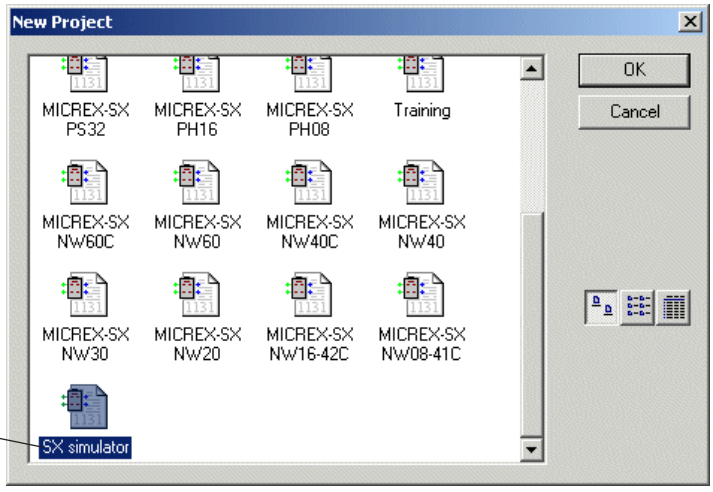


Click here to start up D300win.

Note: If there is a project that was opened last time, D300win is activated with the project opened.

◆ After D300win is started up, create a new project first. Click the [New Project] command in the [File] menu on the menu bar to display the [New Project] dialog.





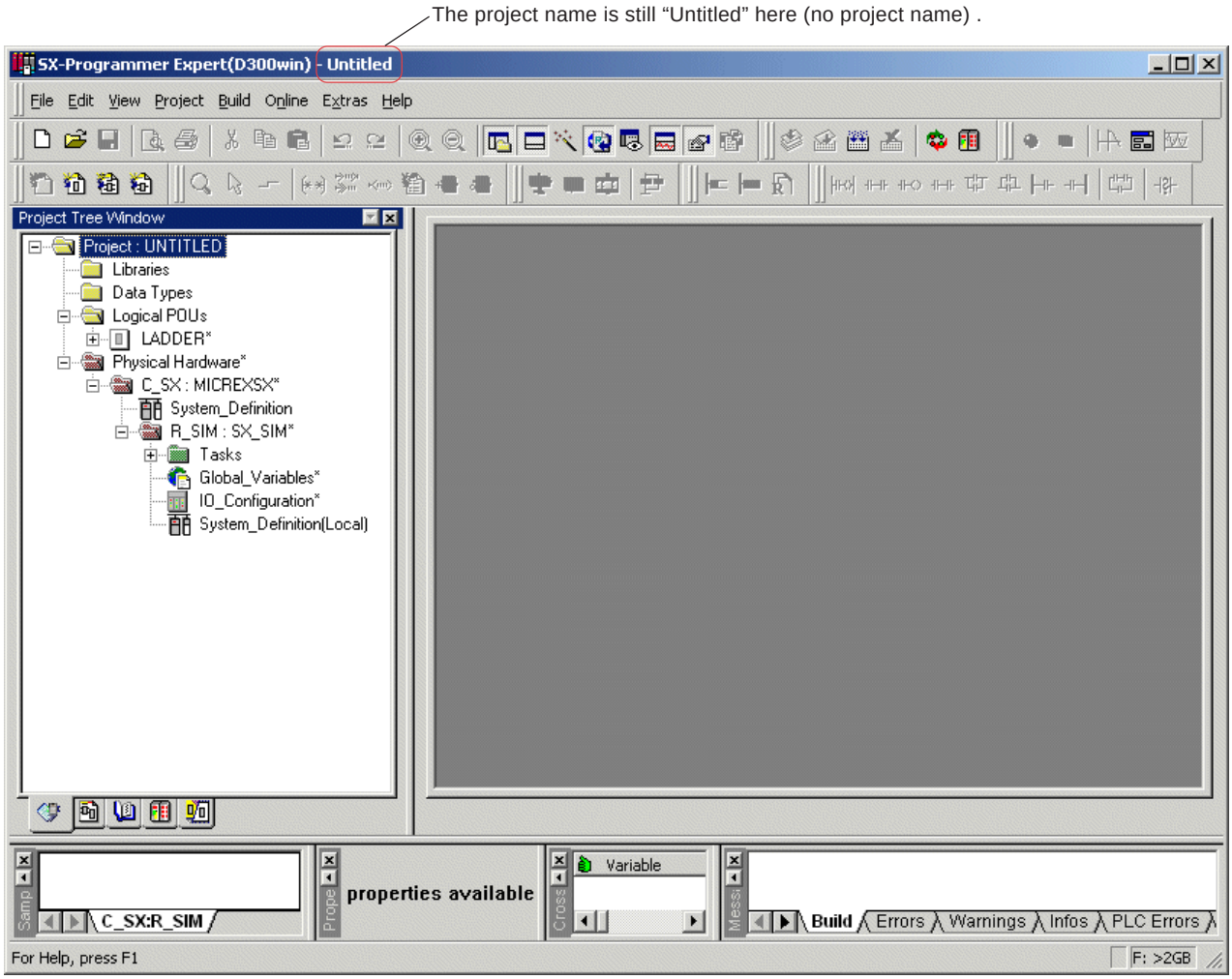
Select “SX simulator”.

Note: For using the SX simulator, it is necessary to select the SX simulator by “Custom setup” when installing D300win. When you select “Typical setup”, the SX simulator is not installed.

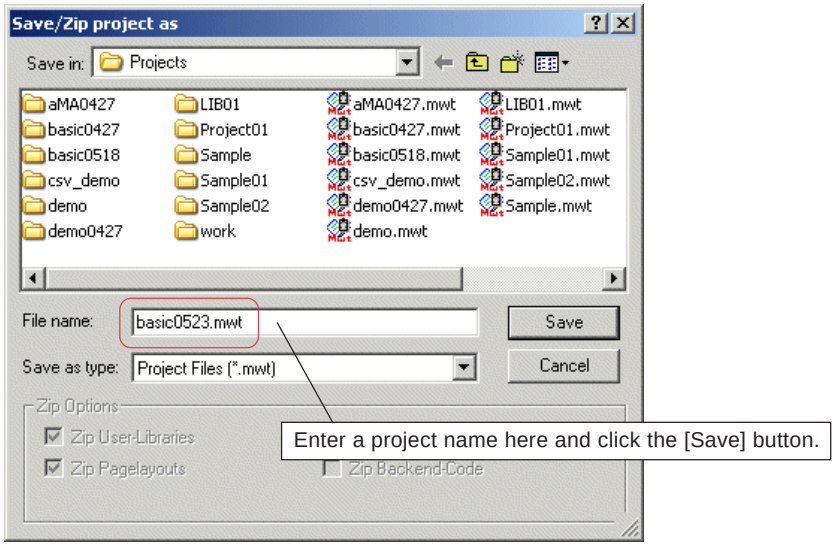
Section 1 Creating a New Project

1-1 Project Name

◆ On the [New Project] dialog, select “SX simulator” and click the [OK] button.
A new project “Untitled” is displayed.



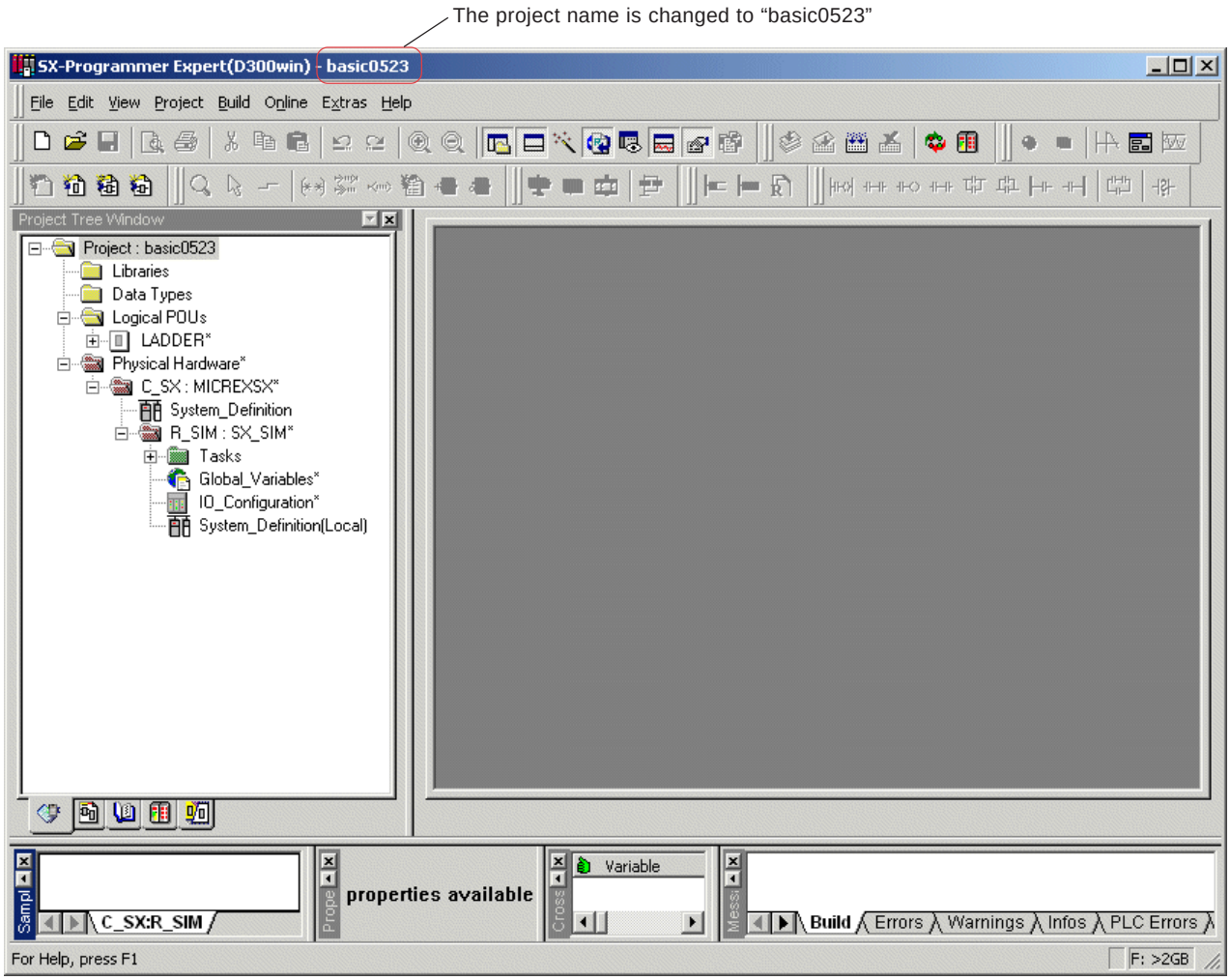
◆ Click the [Save Project As / Zip Project As...] command in the [File] menu on the menu bar of D300win to display the [Save/Zip project as] dialog.



Section 1 Creating a New Project

1-1 Project Name

◆ The project name is set.

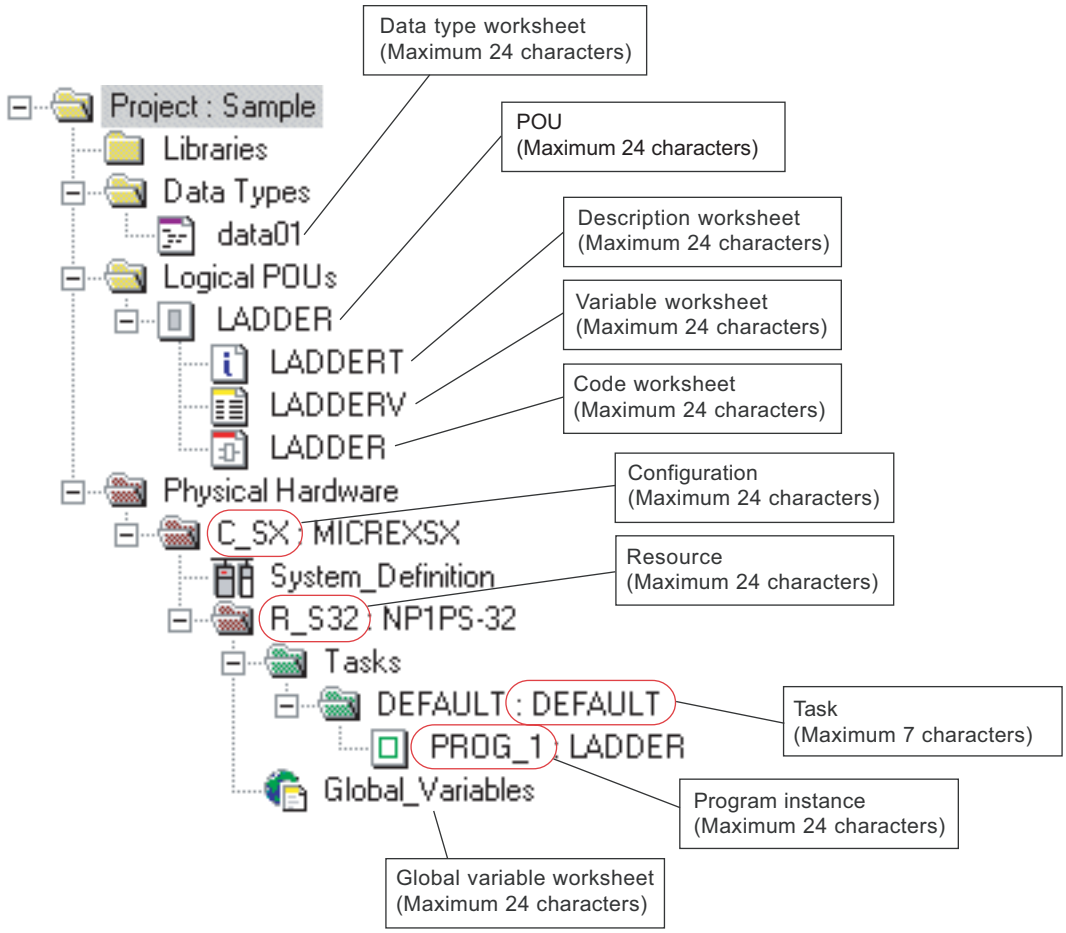


Section 1 Creating a New Project

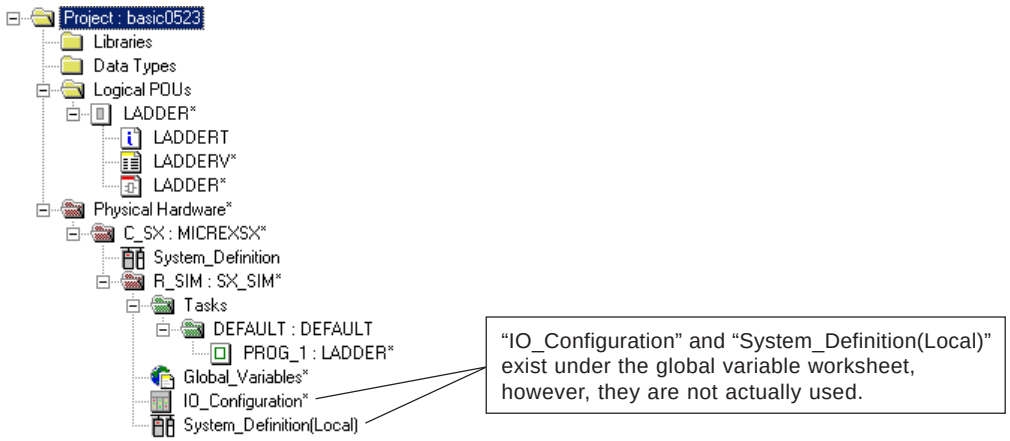
1-1 Project Name

<Structure of project tree (SPH)>

The project tree that is displayed on the D300win screen is structured as follows. Individual folder name or worksheet name is restricted as shown below:



Note: The project tree for the SX simulator is slightly different from the figure shown above. However, you can create a program in much the same manner.

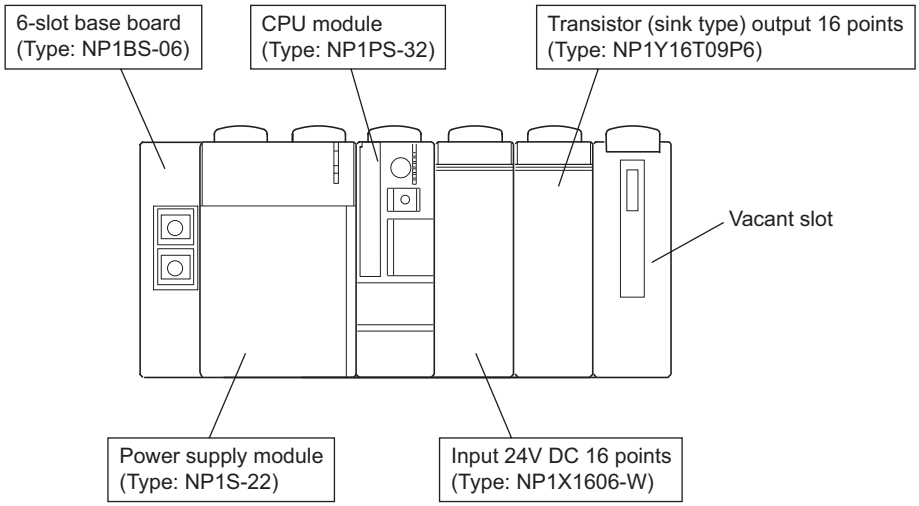


Section 1 Creating a New Project

1-2 System Configuration Definition

For MICREX-SX series, it is necessary to register a base board and modules such as power supply, CPU and I/O to be used so as to match the actual hardware configuration. This paragraph explains how to define system configuration, using the system configuration shown below as an example.

<System configuration>



<Setting items necessary for system configuration definition>

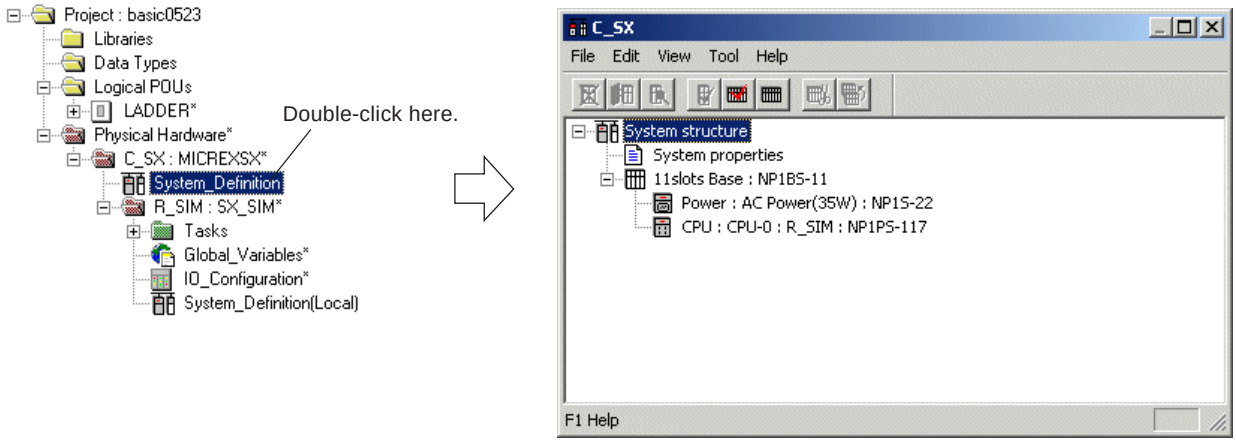
- Registration of modules to be used → See “Section 1-2-1”.
- I/O group setting → See “Section 1-2-2”.
- Tact time setting in the system properties → See “Section 1-2-3”.

1-2-1 Registering modules

Register modules so as to match the system configuration above.

<Registration procedure>

◆ Double-click “System_Definition” in the project tree to display the system configuration definition screen.

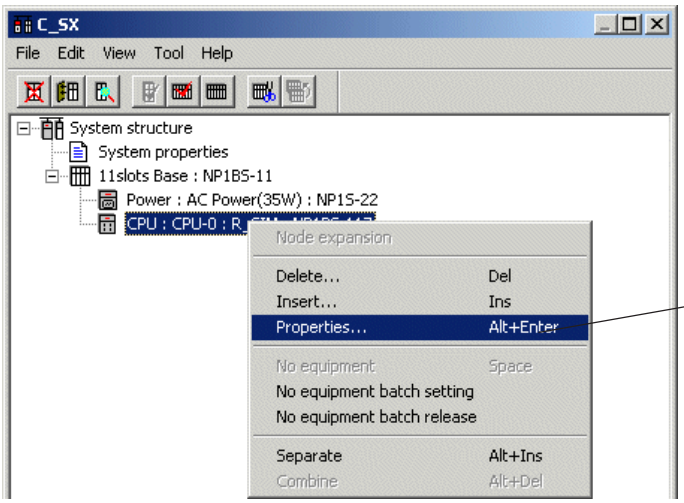


* Initially, an 11-slot base board, AC input power supply module (35W) and NP1PS-117 (the target CPU for CPU (simulator)) are registered.

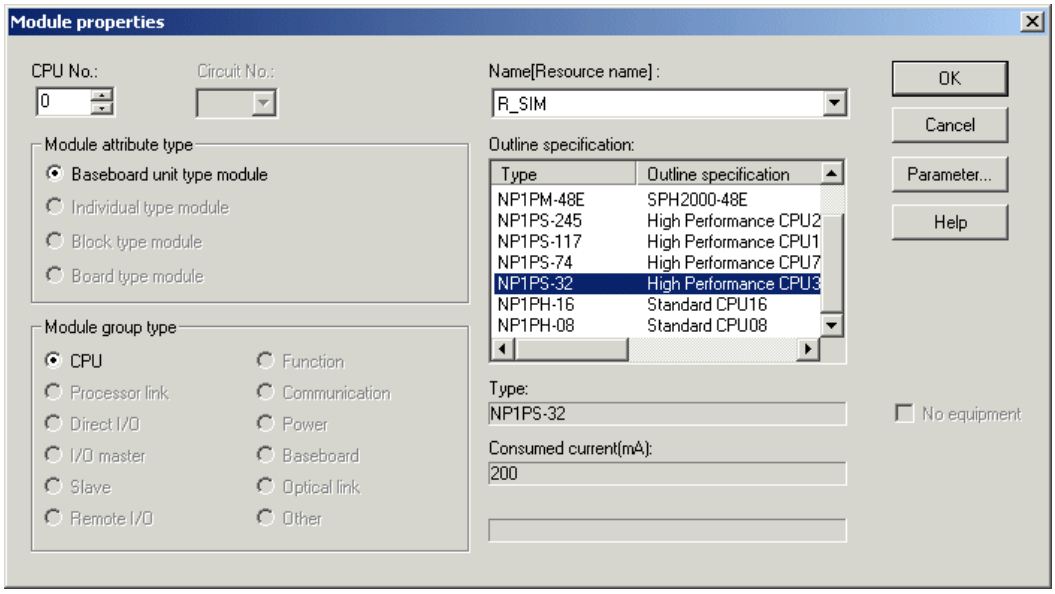
Section 1 Creating a New Project

1-2 System Configuration Definition

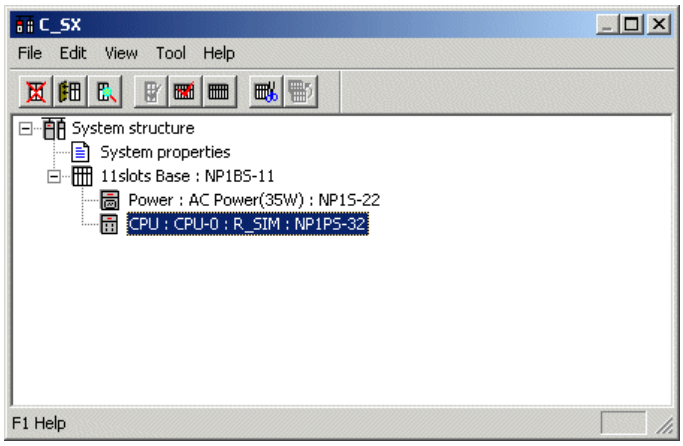
◆ First, change the target CPU. A high-performance CPU of 117K steps (type: NP1PS-117) is registered by default. In this example, change it to a high-performance CPU of 32K steps (type: NP1PS-32).
Right-click the CPU and then execute the [Properties...] command. The [Module properties] dialog is displayed.



◆ Select “NP1PS-32 High-performance CPU32” from the [Outline specification] list box on the [Module properties] dialog.



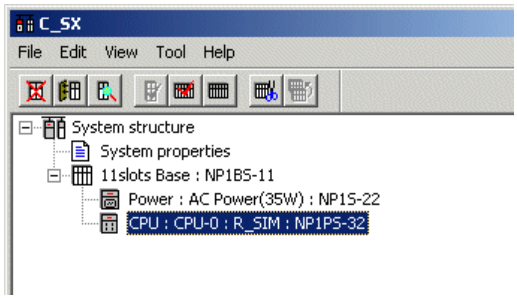
After selecting the CPU, click the [OK] button. The target CPU is changed to NP1PS-32.

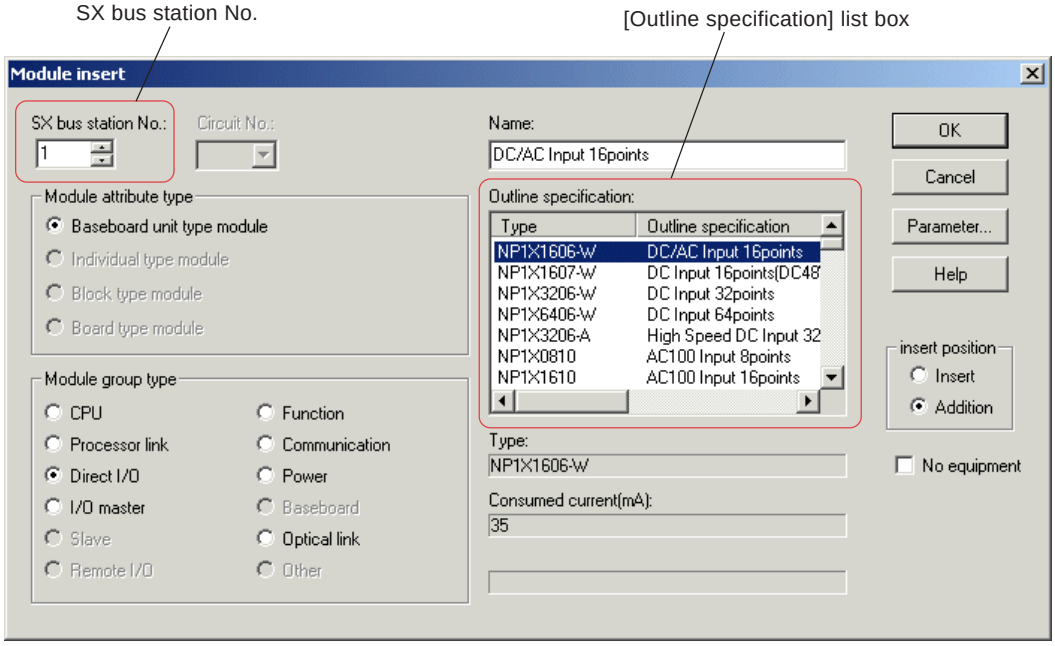


Section 1 Creating a New Project

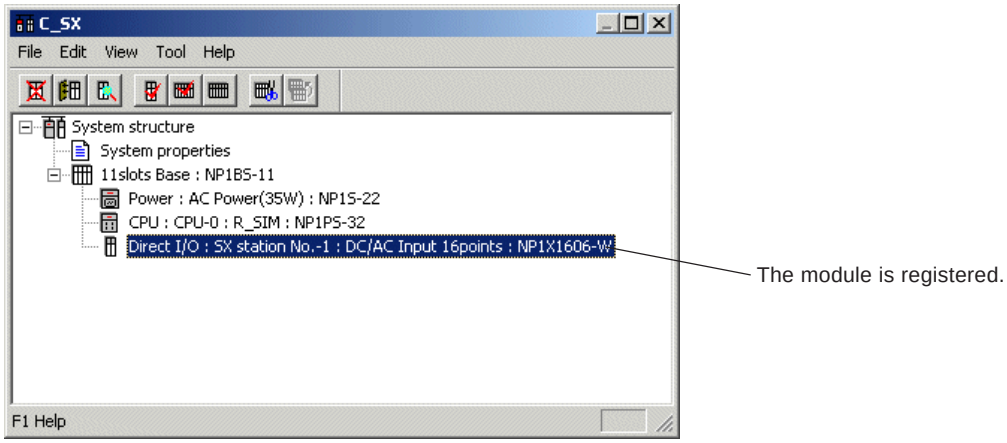
1-2 System Configuration Definition

◆ Modules must be registered in their mounted order. Register the 16-point DC input module (type: NP1X1606-W) so as to match the system configuration on page 1-6.
Select the CPU and then click the [Insert] button or press the <Insert> key of the personal computer. The [Module insert] dialog is displayed.



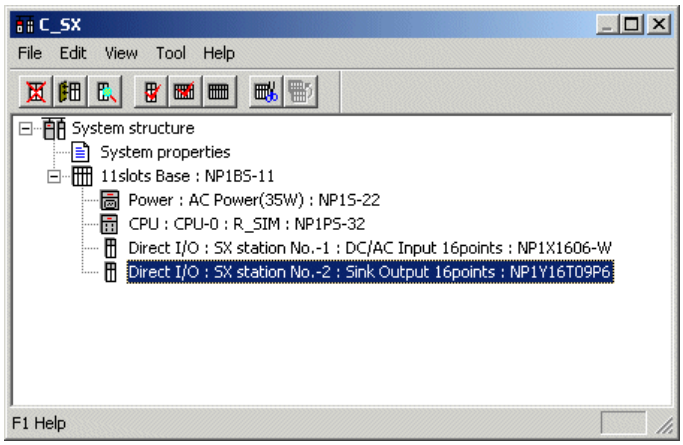


◆ On the [Module insert] dialog, register a module and its SX bus station No. The SX bus station numbers are usually assigned in order of 1, 2, 3... from the right side of the CPU module.
Select a module group type from the [Module group type] box and a module from the [Outline specification] list box. After selecting a module and checking the SX bus station No., click the [OK] button.

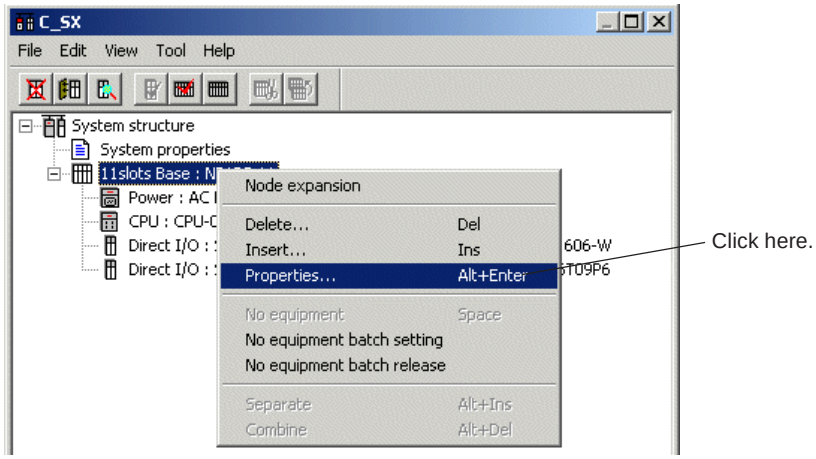


1-2 System Configuration Definition

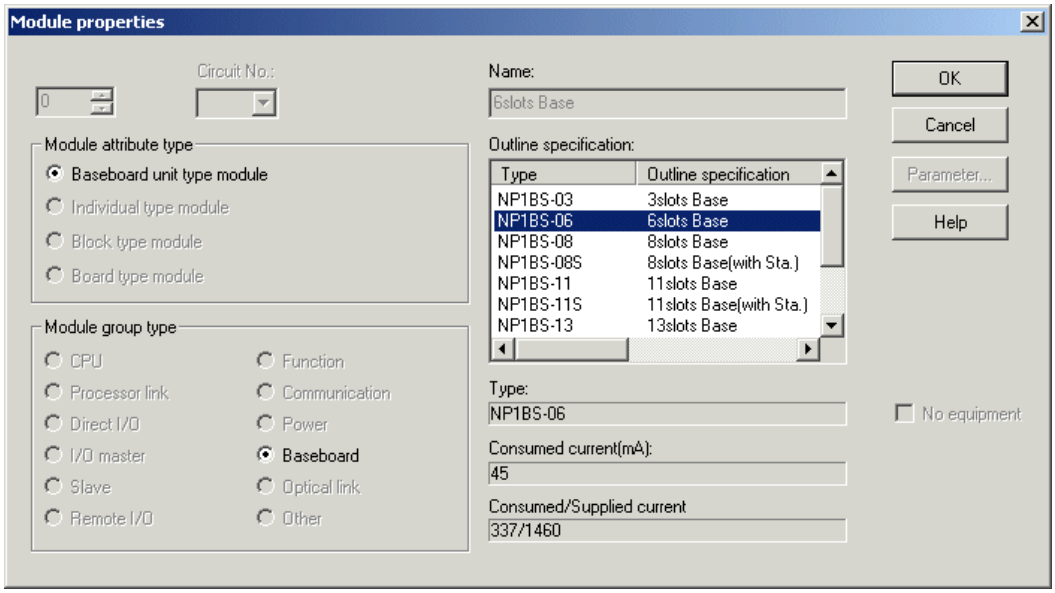
◆ In the same manner, register a 16-point transistor (sink type) output module (type: NP1Y16T09P6).



◆ Next, change the properties of the base board. An 11-slot base board is registered by default. Change it to a 6-slot base board so as to match the system configuration shown on page 1-6.
Right-click the baseboard and then execute the [Properties...] command.



The [Module properties] dialog is displayed.
Select “6slots Base” from the [Outline specification] list box and click the [OK] button.



Section 1 Creating a New Project

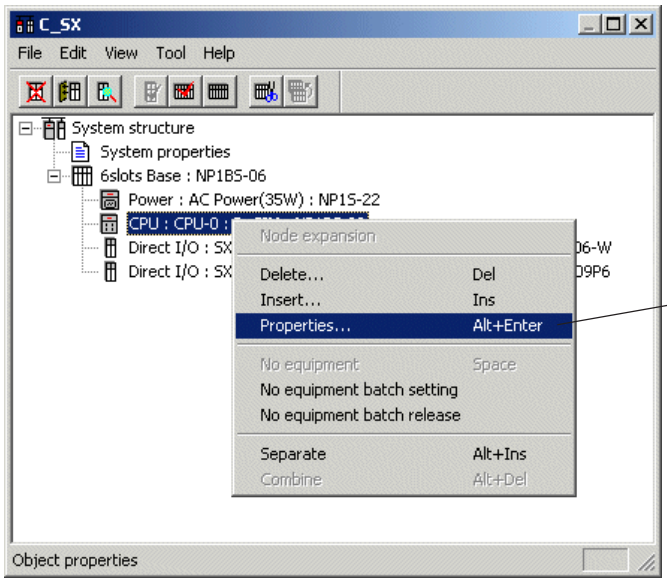
1-2 System Configuration Definition

1-2-2 I/O group setting

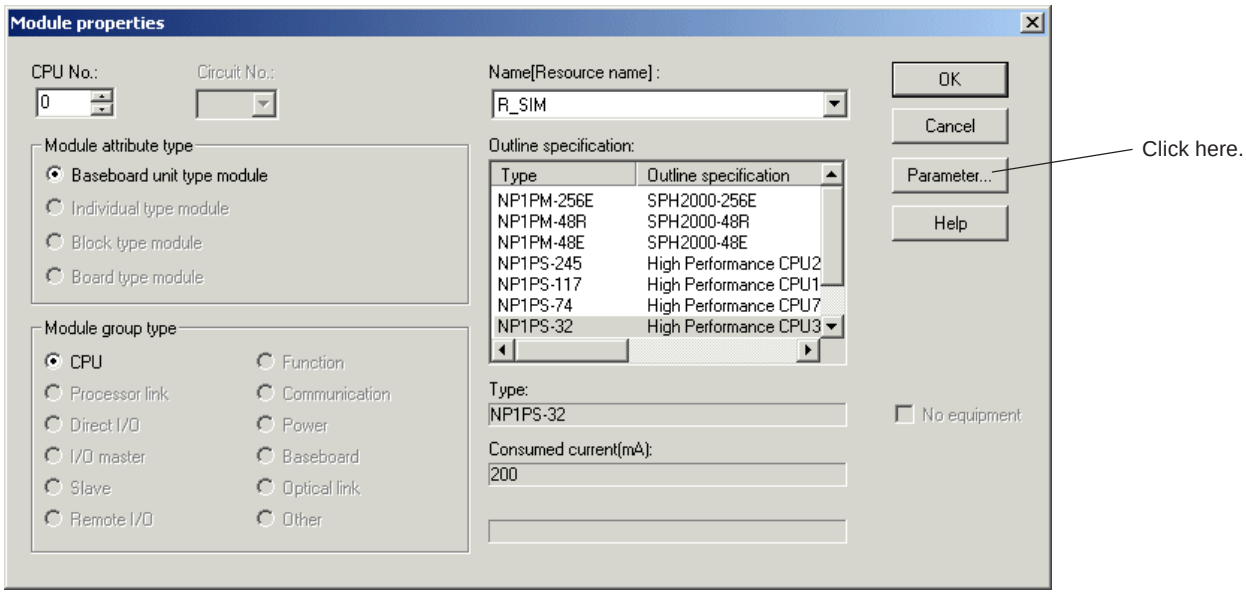
In this process, it is set which task of which CPU controls the I/O modules (modules that have input/output memory) registered in “System_Definition”. Without this setting, it is not possible to use addresses of I/O modules in application programs.

<Setting procedure>

◆ Right-click the CPU on the system configuration definition screen and then execute the [Properties...] command.



The [Module properties] dialog for the CPU is displayed.

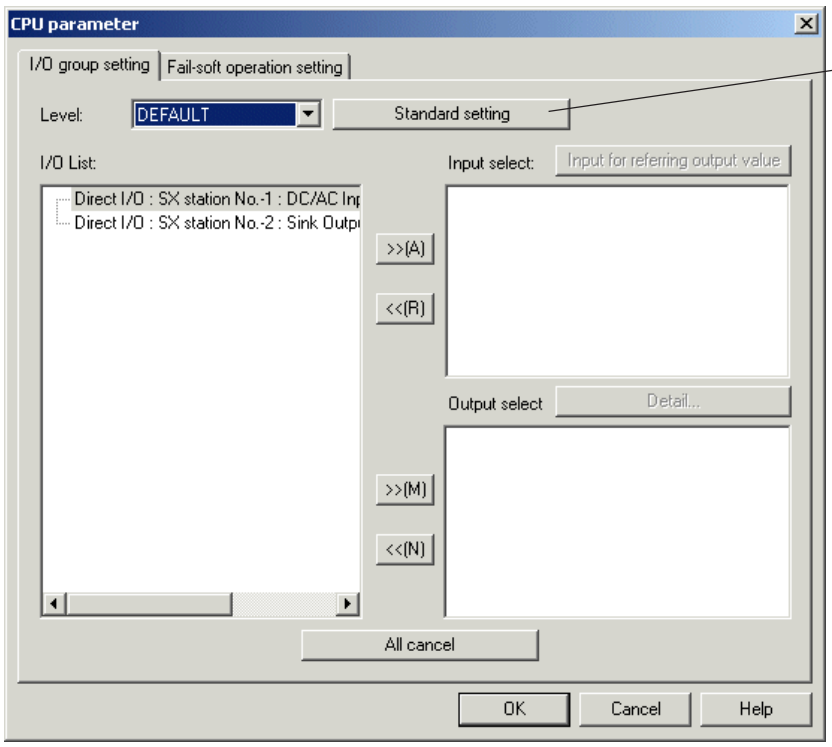


◆ Click the [Parameter...] button on the [Module properties] dialog. The [CPU parameter] dialog is displayed. (shown on the following page)

Section 1 Creating a New Project

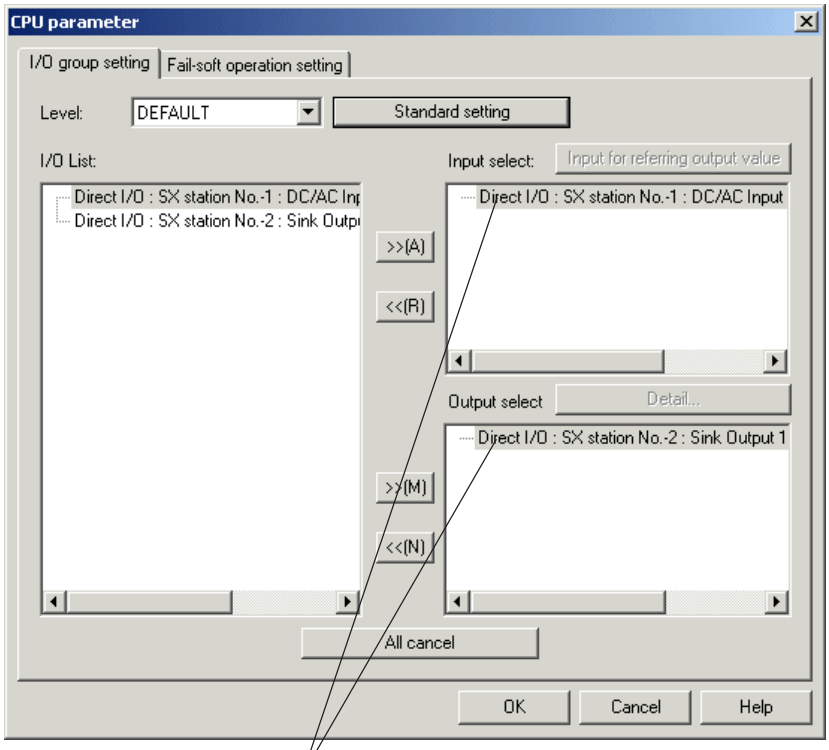
1-2 System Configuration Definition

◆ Register the I/O modules to each task of the CPU. In this example, the “Standard setting” function is used. (All the I/O modules are assigned to the “DEFAULT” task.) Click the [Standard setting] button.



Click here.

* For the tasks, refer to “User’s Manual Instructions FEH200”.



Input modules are registered to the Input select and output modules to the Output select.

◆ After determining the setting, click the [OK] button.

Section 1 Creating a New Project

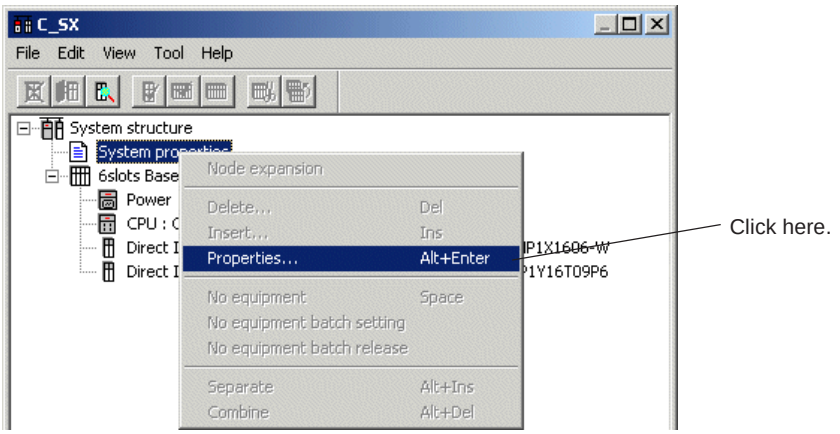
1-2 System Configuration Definition

1-2-3 Tact time setting

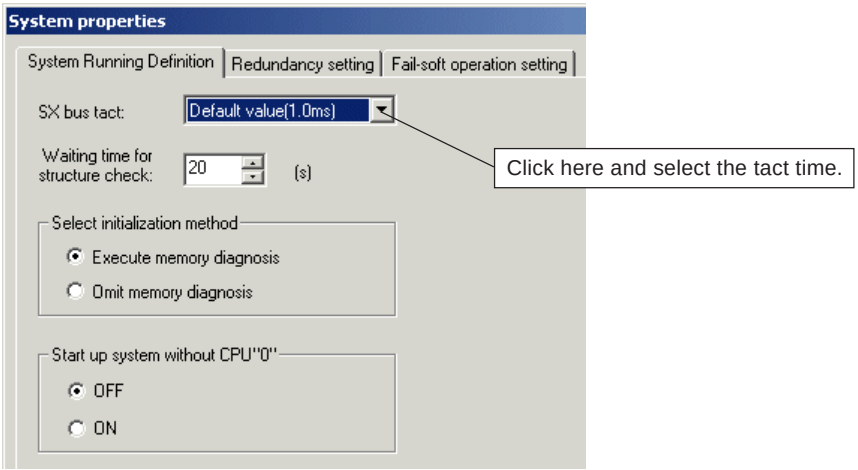
In one configuration of MICREX-SX, I/O refresh and activation of tasks (program execution) are performed in synchronization with the tact time. Default is 1ms, however, it is necessary to calculate a tact time according to the system configuration. The example system in this section is a single-CPU system configured with one 16-point input module and one 16-point output module. Therefore, the tact time can be set from 0.5ms.

<Setting procedure>

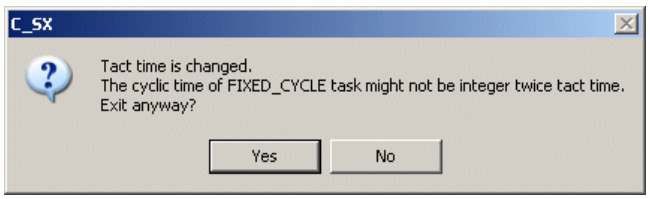
◆ Right-click “system properties” on the system configuration definition screen and then execute the [Properties...] command.



◆ The [System properties] dialog is displayed. Set “SX bus tact” on the [System Running Definition] tab window.



◆ In this example, select 1ms. Clicking the [OK] button displays the confirmation window shown below. Click the [Yes] button to complete the tact time setting.

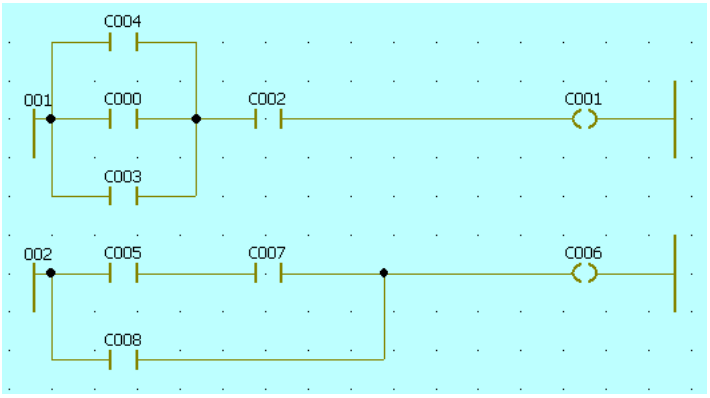


Note: The SX simulator operates at 20ms (fixed) even if 1ms is selected here.

Section 1 Creating a New Project

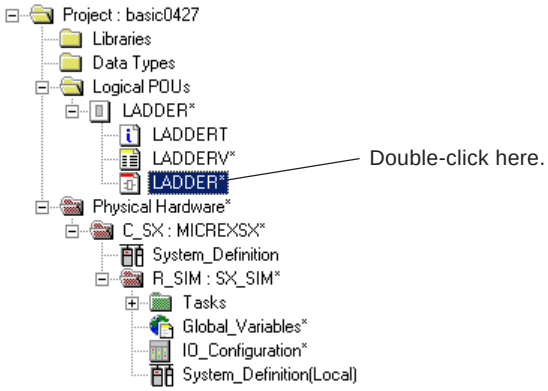
1-3 Practice Inserting a Contact or a Coil

Let's master the operation of D300win so that you can create an LD circuit shown below with ease.

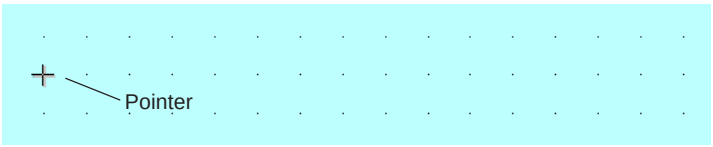


<Operational procedure>

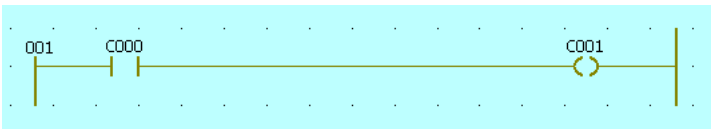
◆ Double-click the "LADDER" code worksheet in the project tree to display the code worksheet.



◆ The code worksheet of POU "LADDER" is displayed. Click an arbitrary point on the worksheet to place the pointer where you want to paste a circuit .

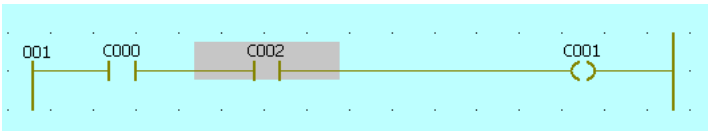


◆ Click button on the tool bar. A circuit consisting of a pair of one contact and one coil is created.

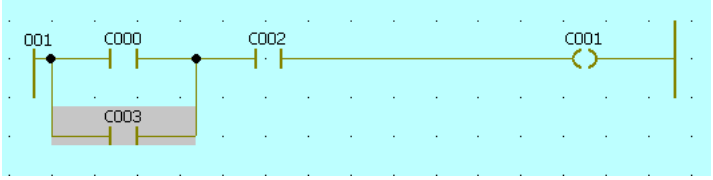


Note: Symbols above contacts and coils are "variable names" that are temporarily given. On an actual program, rename them meaningfully.

◆ Select contact "C000" and click button. A contact is inserted on the right of contact "C000".



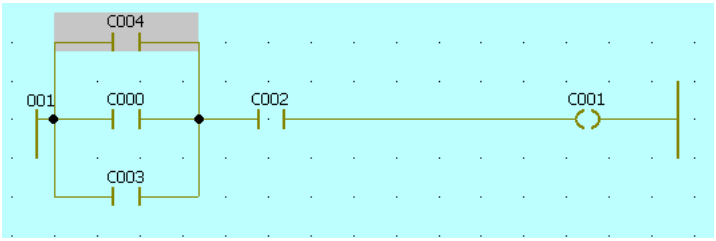
◆ Select contact "C000" and click button. A contact is inserted below contact "C000".



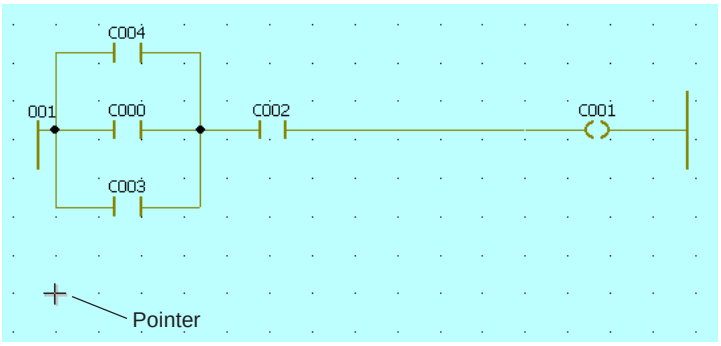
Section 1 Creating a New Project

1-3 Practice Inserting a Contact or a Coil

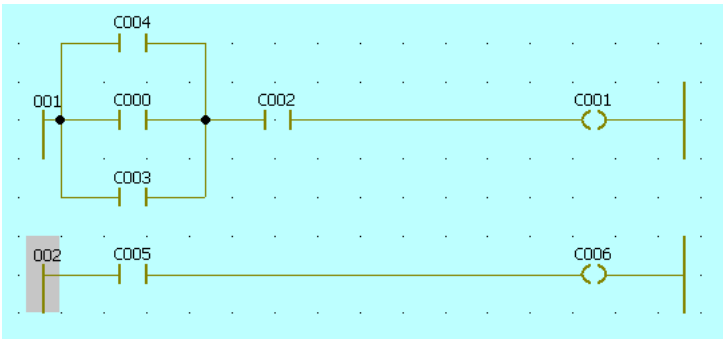
◆ Select contact “C000” and click button. A contact is inserted above contact “C000”.



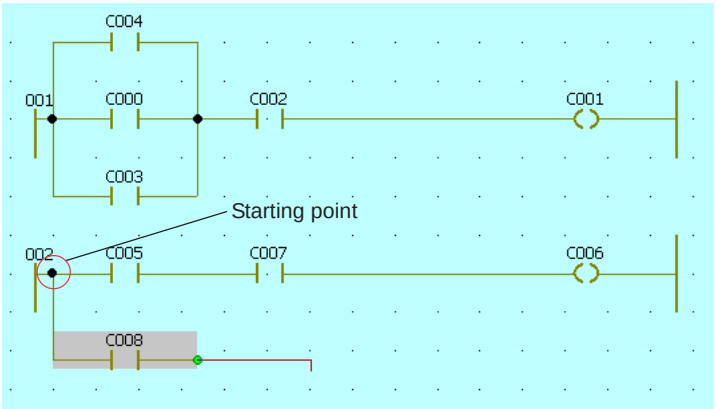
◆ To create a next circuit, click an arbitrary point on the worksheet to place the pointer.



◆ Click button. A circuit consisting of a pair of one contact and one coil is created.



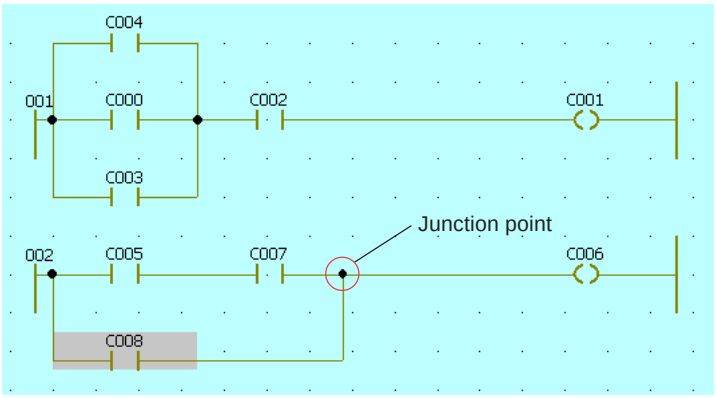
◆ Click button to insert a contact on the right of contact “C005”. Click button, and the pointer for LD branch edit mode is added to the cursor. Click the left of contact “C005” (the starting point of OR contact) and then a point where you want to paste a new object (contact).



Section 1 Creating a New Project

1-3 Practice Inserting a Contact or a Coil

◆ Move the cursor and click an arbitrary junction point.



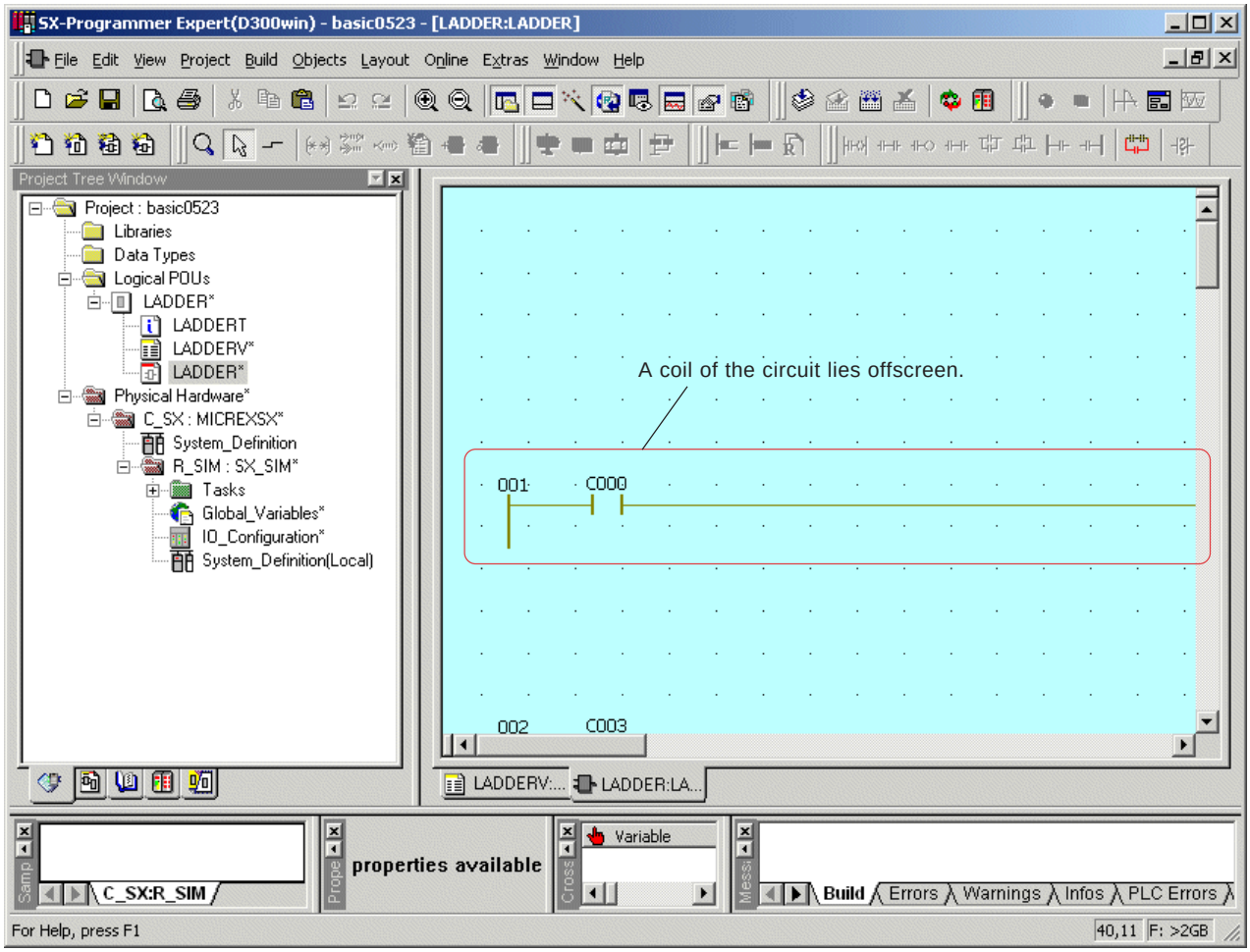
* For more information about the operation, refer to “Section 4” of User’s Manual D300win Reference (FEH257).

Section 1 Creating a New Project

1-3 Practice Inserting a Contact or a Coil

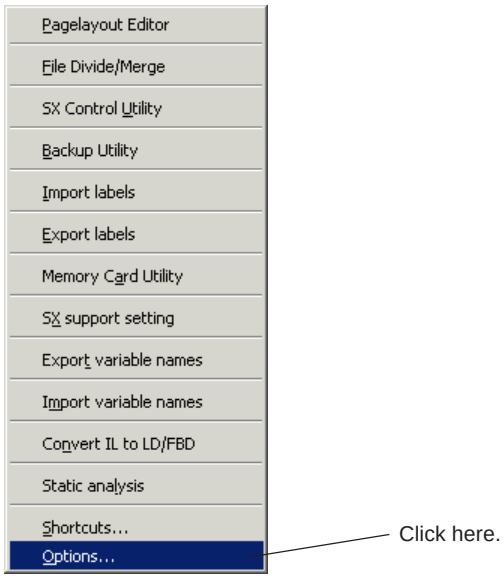
[Tip]

When you click button, a circuit consisting of a pair of one contact and one coil is created. If a part of the circuit lies offscreen, the size of the circuit displayed by clicking button can be changed.



<Operational procedure>

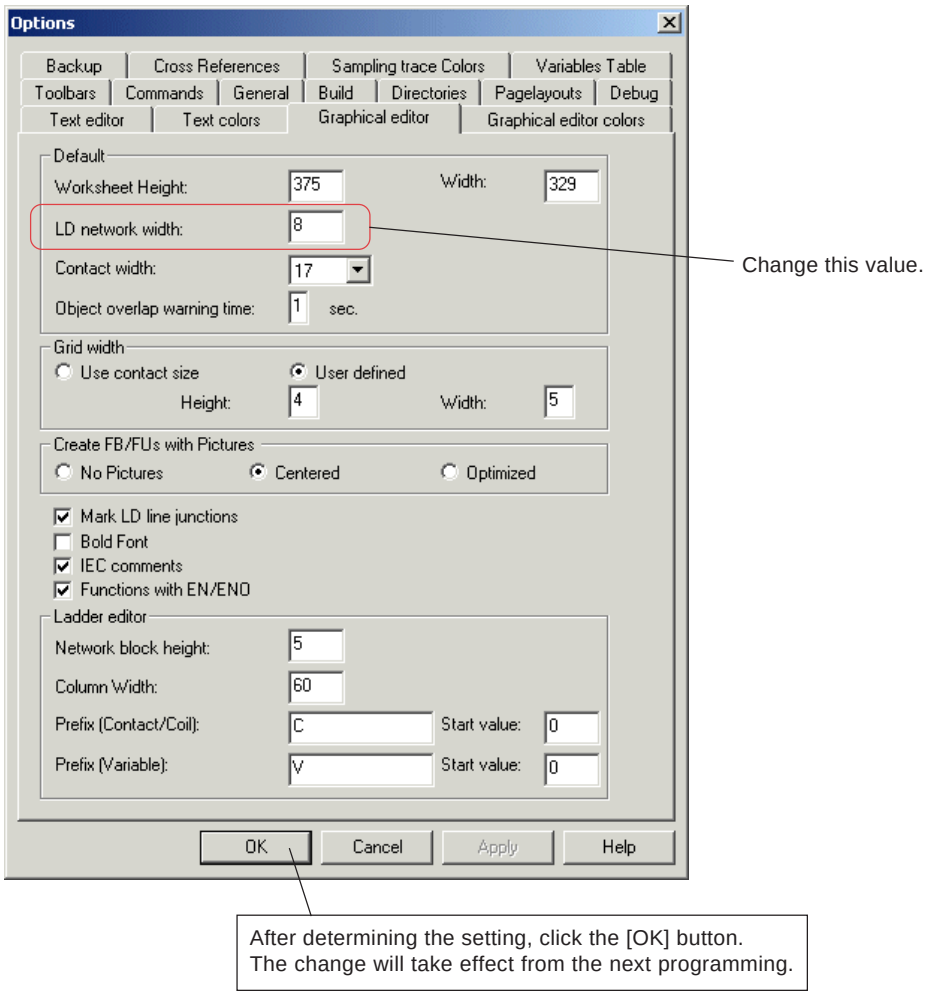
◆ Click the [Options] command in the [Extras] menu to display the [Options] dialog.



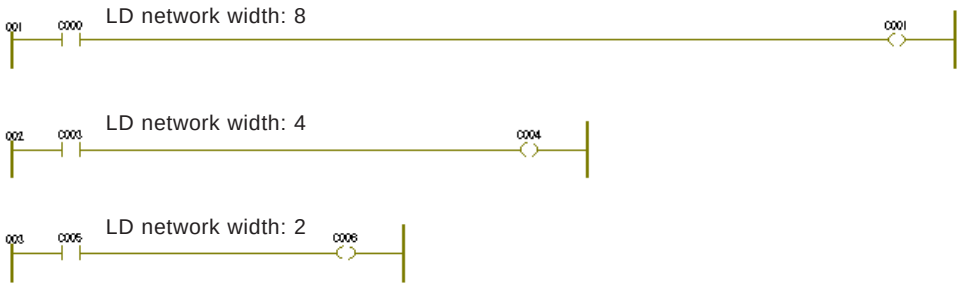
Section 1 Creating a New Project

1-3 Practice Inserting a Contact or a Coil

◆ Click the [Graphical editor] tab and change the value of “LD network width” in the [Default] box. Default value is “8”. Although there is no particular unit such as “mm”, set an appropriate value with reference to “8”. For example, if you change it to “4”, the circuit will be displayed in half the size of default.



<Set value and circuit size>

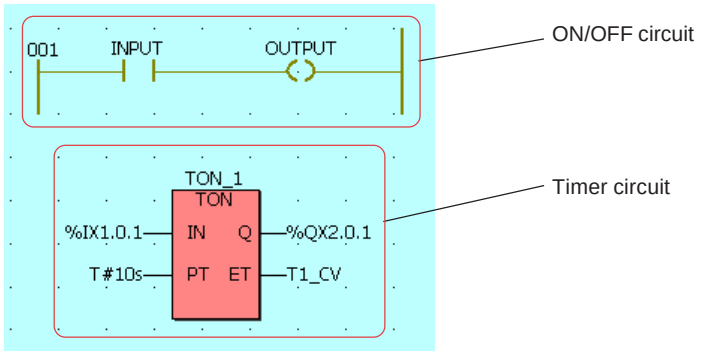


Section 1 Creating a New Project

1-4 ON/OFF Circuit and Timer Circuit

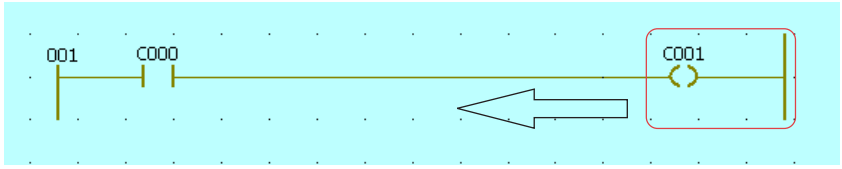
This paragraph explains how to create an ON/OFF circuit and timer circuit on the POU “LADDER” code worksheet.

<A circuit to be created>

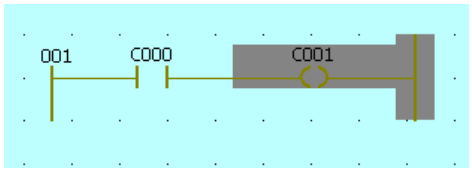


<Operational procedure>

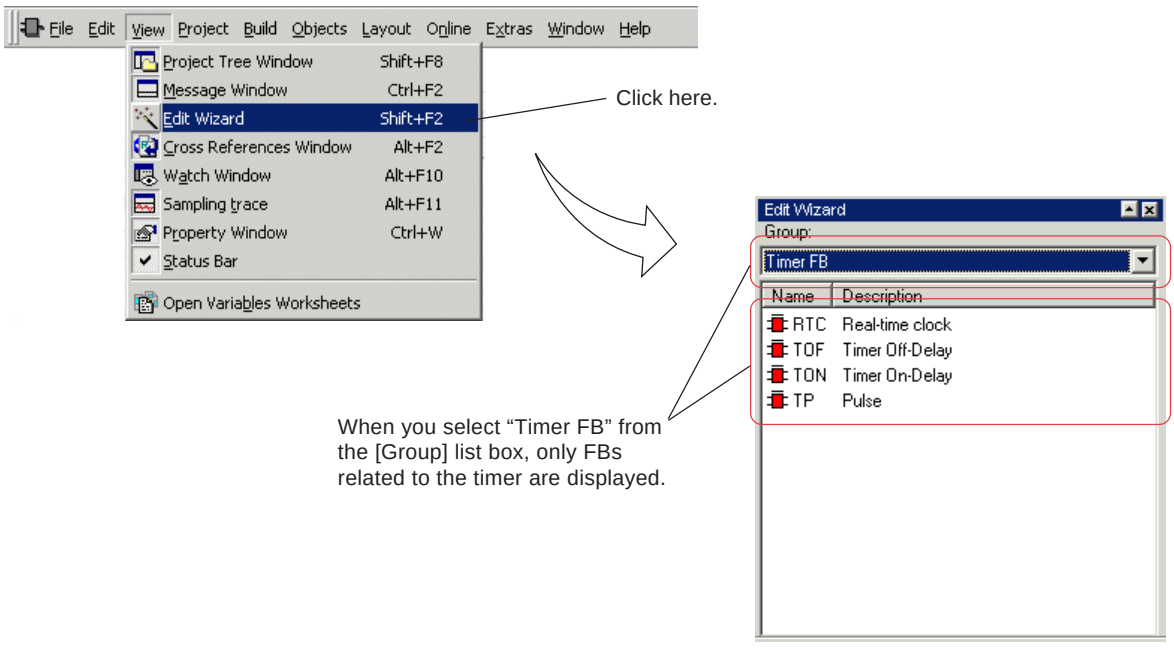
◆ Create an ON/OFF circuit. Move the cursor and click an arbitrary point on the worksheet to specify the starting point of the circuit. Then, click button. An ON/OFF circuit is created.



◆ On the graphical editor, objects such as contacts and coils can be freely laid out; they can be moved to any position. When you select coil “C001” and the right power rail and drag them to the left, the circuit can be smaller as shown below.



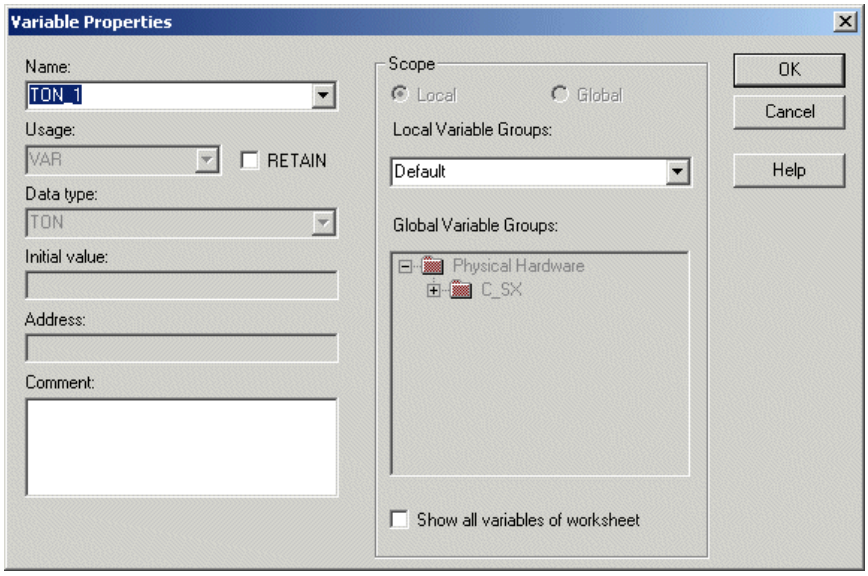
◆ Then, insert a timer circuit. Place the pointer at an arbitrary point and execute the [Edit Wizard] command in the [View] menu. The [Edit Wizard] window is displayed.

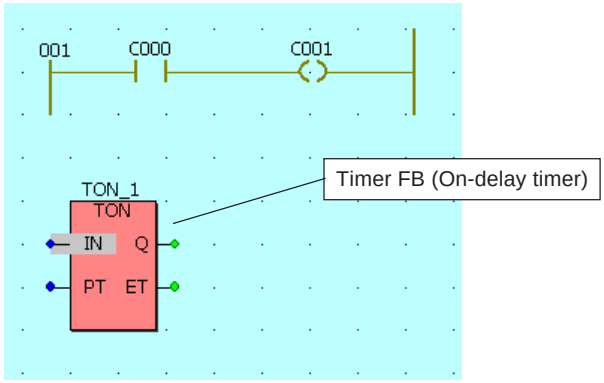


Section 1 Creating a New Project

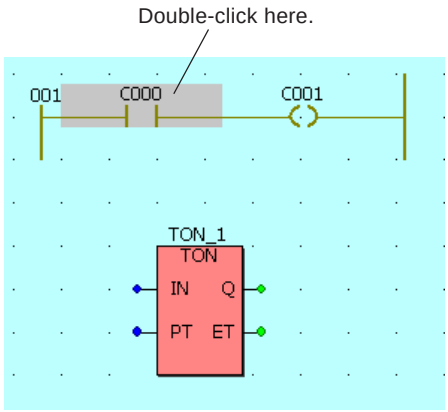
1-4 ON/OFF Circuit and Timer Circuit

◆ Double-click an FB to be pasted to display the [Variable Properties] dialog. Clicking the [OK] button pastes a timer FB on the code worksheet.





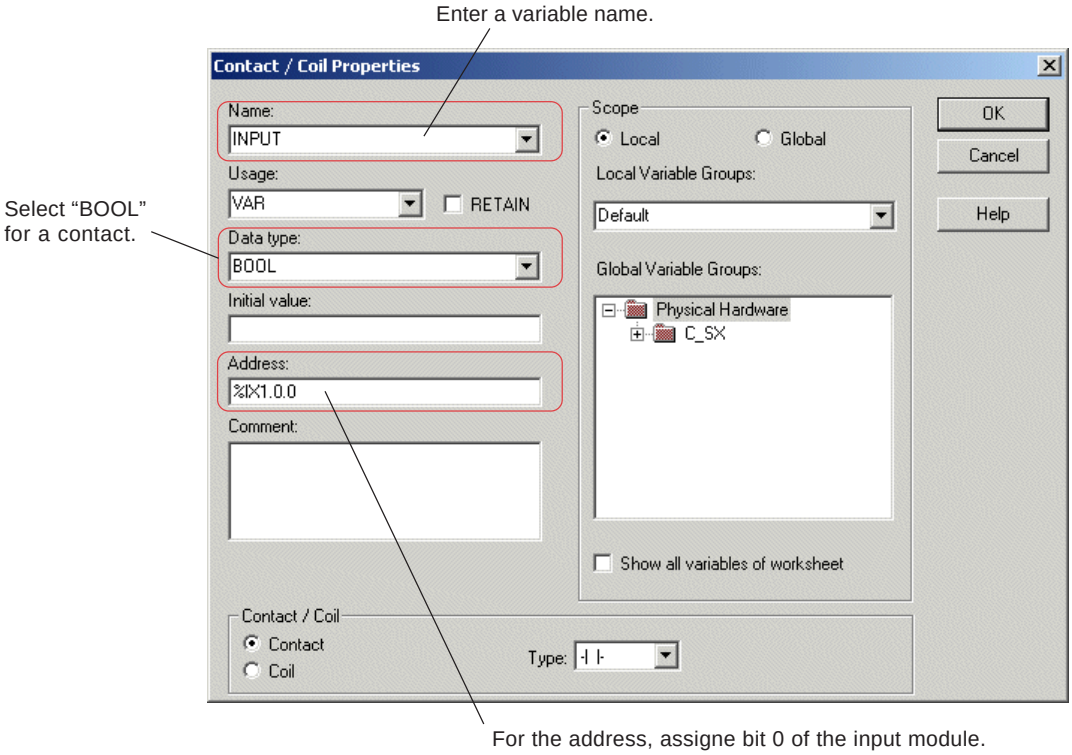
◆ Assign variables to the contact and coil of the ON/OFF circuit. First, double-click contact “C000” to display the [Contact / Coil Properties] dialog.



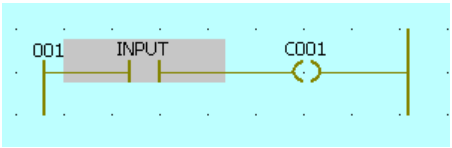
Section 1 Creating a New Project

1-4 ON/OFF Circuit and Timer Circuit

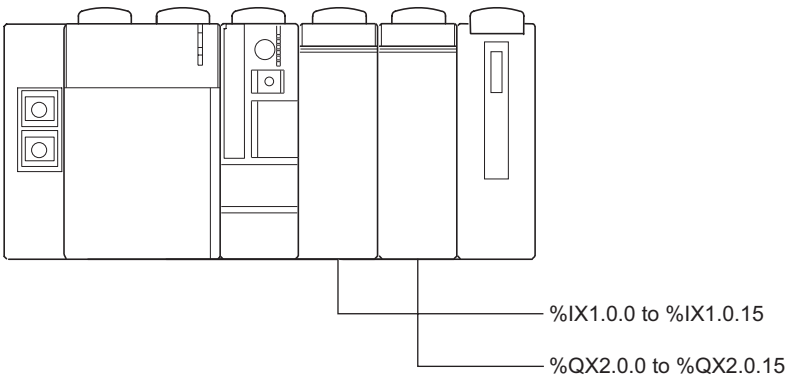
◆ On the [Contact / Coil Properties] dialog, enter a variable name and specify a real address assigned to the variable.



◆ After setting all necessary items, click the [OK] button. The variable “INPUT” is assigned to the contact.

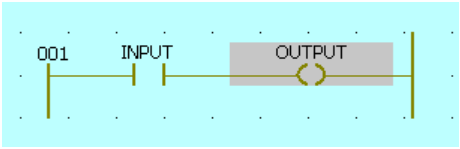


<I/O Address assignment>



* For more information about I/O address assignment, refer to “MICREX-SX User’s Manual Instructions (FEH200)”.

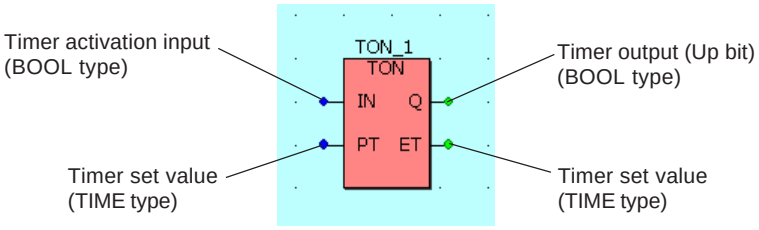
◆ In the same manner, assign a variable “OUTPUT” and address “%QX2.0.0” to the coil.



Section 1 Creating a New Project

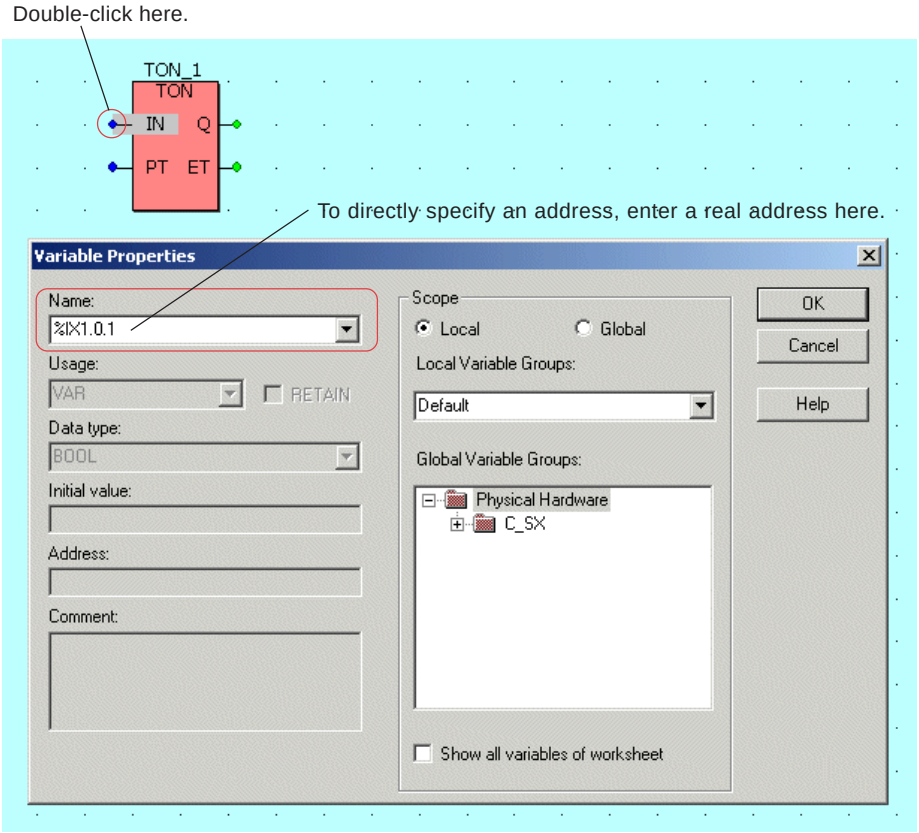
1-4 ON/OFF Circuit and Timer Circuit

◆ Next, connect a variable, real address, and constant (set value of timer) to each terminal of the timer FB.
For function blocks and functions, terminals on the left are input and the ones on the right are output.
Connect the following to each terminal of the timer FB.

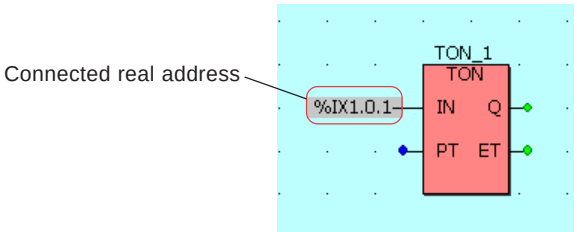


* For more information about the timer FB, refer to “MICREX-SX User’s Manual Instructions (FEH200)”.

◆ Double-click the “IN” terminal of the timer FB to display the “Variable Properties” dialog.



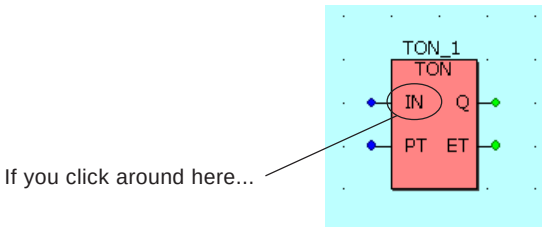
◆ Directly enter an address (%IX1.0.1) in the “Name” box and click the [OK] button. The address is connected to the terminal as shown below.



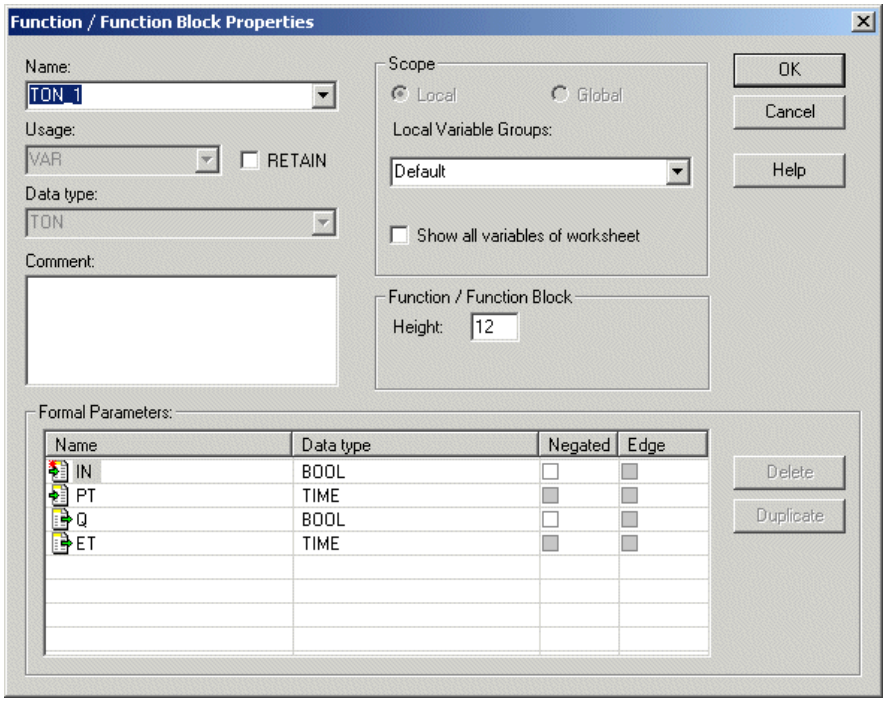
Section 1 Creating a New Project

1-4 ON/OFF Circuit and Timer Circuit

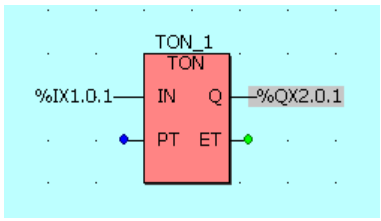
Note: If you click other positions on the terminal, a different dialog is displayed.



The [Function / Function Block Properties] dialog is displayed.



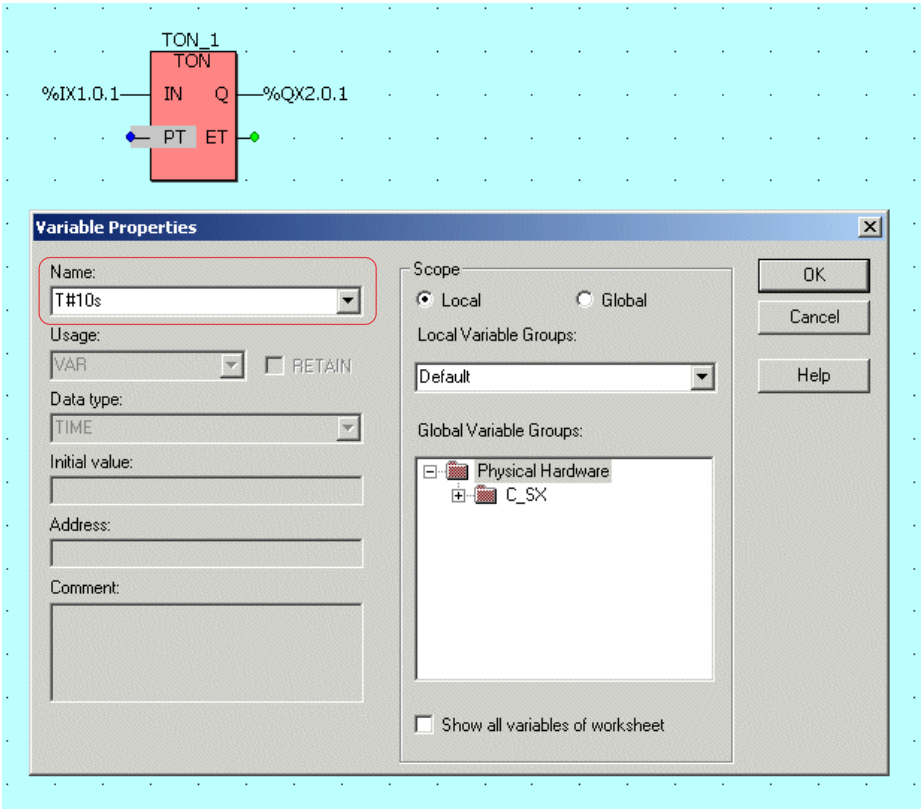
◆ In the same manner, assign a real address (%QX2.0.2) to the “OUT” terminal of the timer FB.



Section 1 Creating a New Project

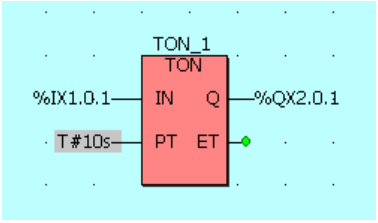
1-4 ON/OFF Circuit and Timer Circuit

◆ Connect a timer set value (constant) to the timer set value terminal “PT”. Double-click “PT” terminal and directly enter a timer value in the “Name” box on the [Variable Properties] dialog.



A “TIME-type” variable or constant can be connected to the timer set value terminal. In this example, enter “T#10s” (Set value: 10 seconds). For more information about how to describe constants of each data type, refer to “User’s Manual Instructions (FEH200)”.

◆ After determining the setting, click the [OK] button. The set value is connected as shown below.



<Data types of variables and constants dealt with in MICREX-SX>

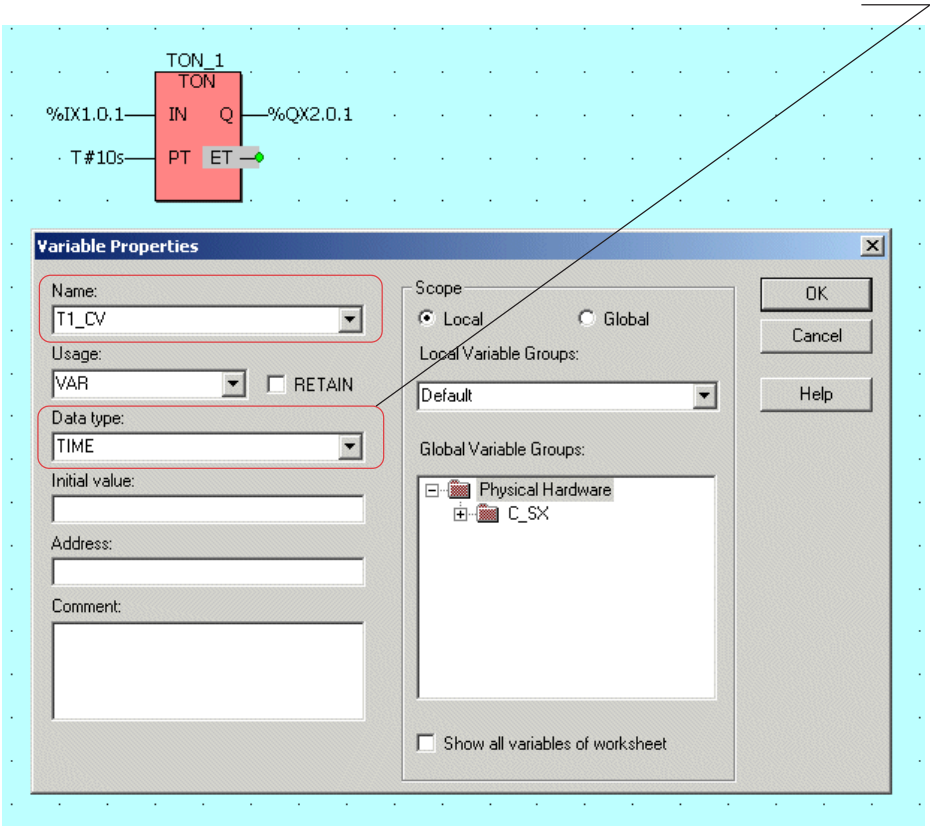
As you know, MICREX-SX series of programmable controllers conform to IEC161131-3. Under IEC, it is necessary to specify data types for all data dealt with in applications. For example, the data type of timer set values and current values must be “TIME” type. MICREX-SX supports the following 13 (indicated in red) of basic data types defined in IEC. For more information, refer to “User’s Manual Instructions (FEH200)”.

Basic data type (elementary)
- ANY_NUM (number type)
  - ANY_REAL (real type) → REAL
  - ANY_INT (integer type) → INT, DINT, UINT, UDINT
- ANY_BIT (bit type) → BOOL, WORD, DWORD
- Character string type → STRING
- ANY_DATE (date type) → DT, DATE, TOD
- Time type → TIME

Section 1 Creating a New Project

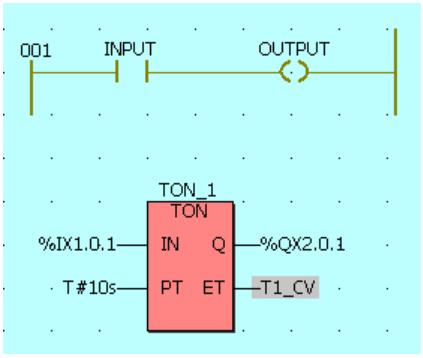
1-4 ON/OFF Circuit and Timer Circuit

◆ Connect a variable “T1_CV” to the timer current value terminal “ET”. Double-click “ET” terminal to display the [Variable Properties] dialog. Enter a variable name “T1_CV” in the name box and select “TIME” for the data type.



◆ After setting all necessary items, click the [OK] button. The variable is connected to the terminal. The code worksheet and variable worksheet become as shown below.

<Code worksheet>



<Variable worksheet>

Variable	Data type	Usage	Comment	Address
Default				
TON_1	TON	VAR		
INPUT	BOOL	VAR		%IX1.0.0
OUTPUT	BOOL	VAR		%QX2.0.0
T1_CV	TIME	VAR		

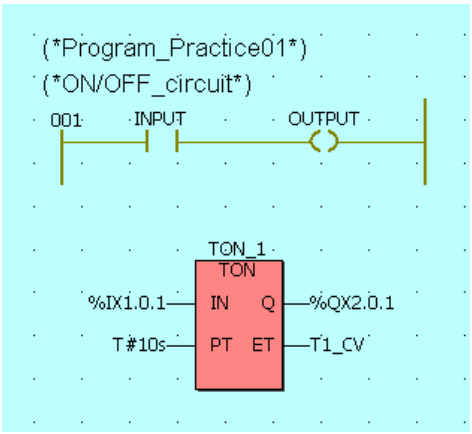
Section 1 Creating a New Project

1-5 Inserting a Comment

This paragraph explains how to insert a comment on the created code worksheet and variable worksheet. On the graphical editor (code worksheet of LD language), comments and program codes can be freely laid out.

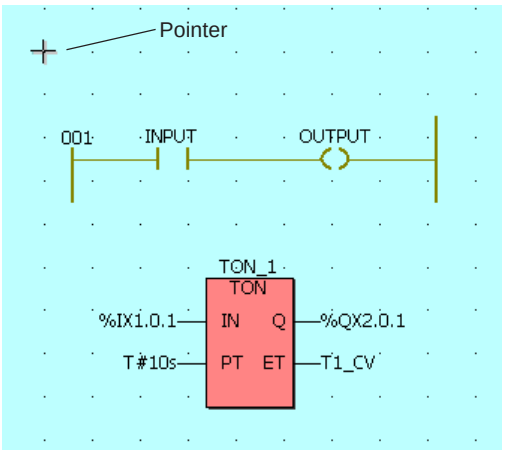
1-5-1 Inserting a comment on the code worksheet

Insert a comment on the code worksheet as shown below.



<Operational procedure>

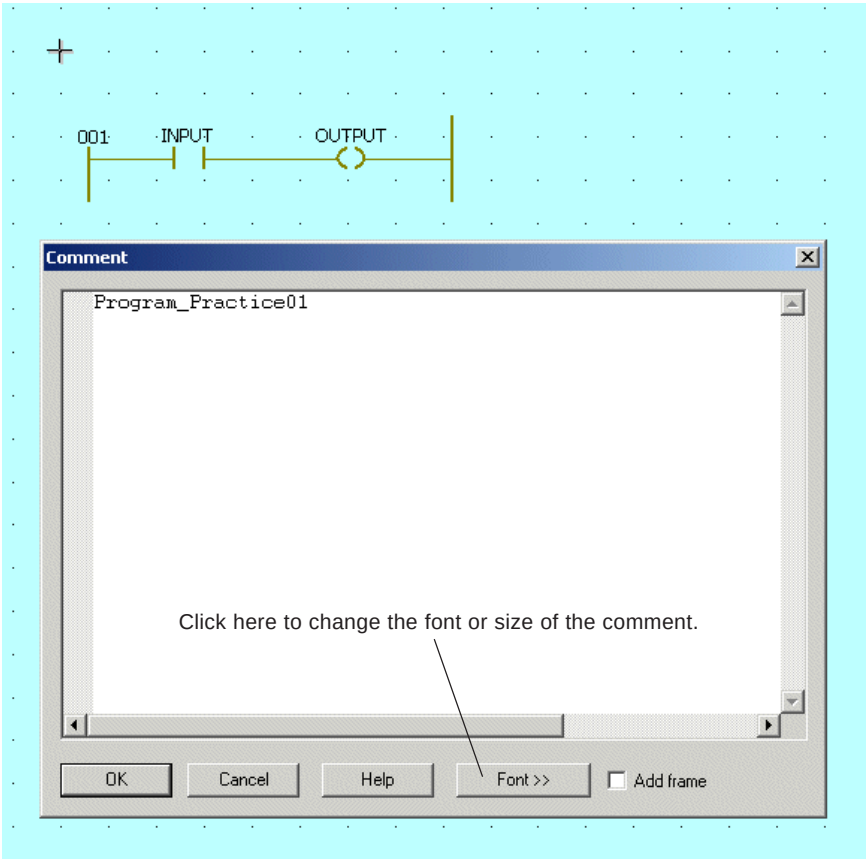
◆ Place the pointer at an arbitrary point on the code worksheet where you want to insert a comment. You can move inserted comments later.



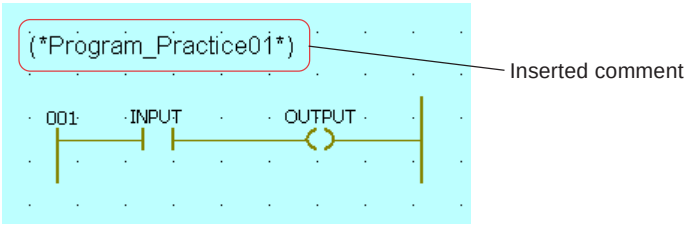
Section 1 Creating a New Project

1-5 Inserting a Comment

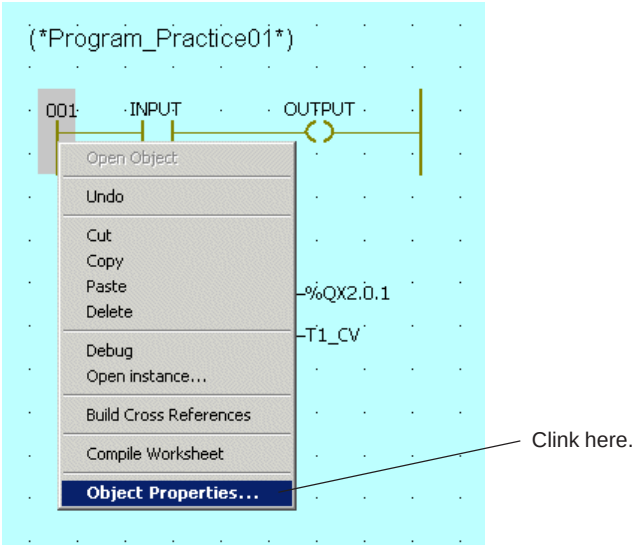
◆ Click button to display the [Comment] dialog. On this dialog box, enter a comment.



◆ After entering a comment, click the [OK] button. The comment is inserted at the position where the pointer was placed.



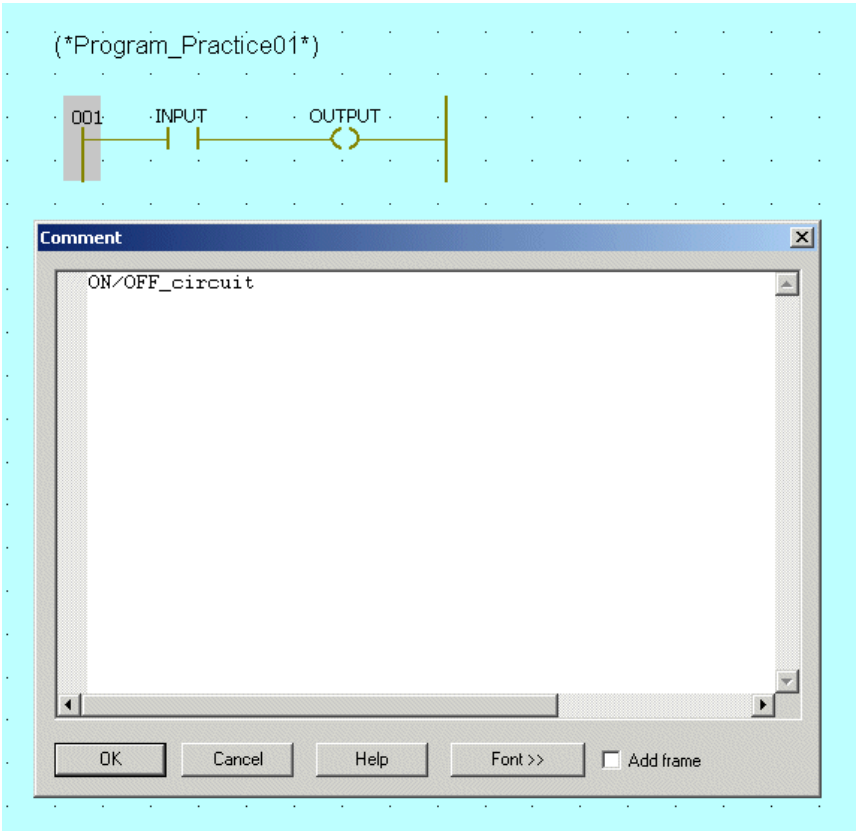
◆ Next, add a comment to the ON/OFF circuit. Right-click the left power rail and then execute the [Object Properties...] command. The [Comment] dialog is displayed.

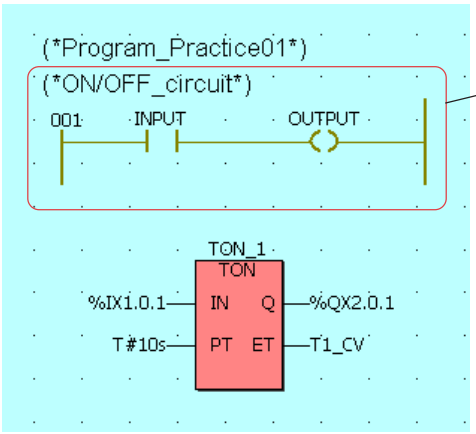


Section 1 Creating a New Project

1-5 Inserting a Comment

◆ After entering a comment, click the [OK] button.





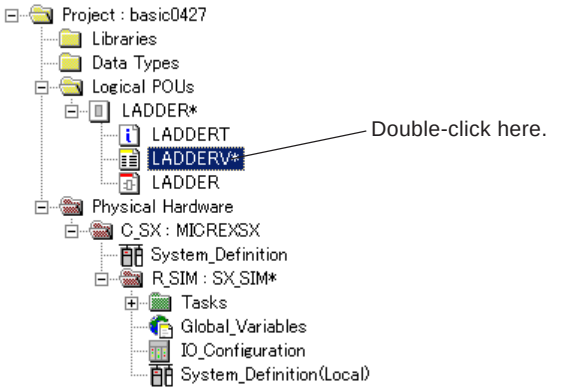
"(*ON/OFF_circuit*)" is a comment that is added to the ON/OFF circuit. Therefore, if the circuit is moved, the comment is also moved.

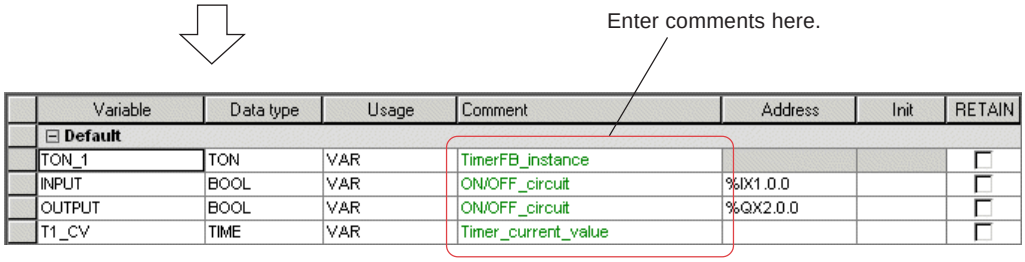
Section 1 Creating a New Project

1-5 Inserting a Comment

1-5-2 Inserting a comment on the variable worksheet

◆ Double-click the “LADDER” variable worksheet in the project tree to display the variable worksheet. Then, enter a comment in the comment box on the variable worksheet.





	Variable	Data type	Usage	Comment	Address	Init	RETAIN
	Default						
	TON_1	TON	VAR	TimerFB_instance			
	INPUT	BOOL	VAR	ON/OFF_circuit	%IX1.0.0		
	OUTPUT	BOOL	VAR	ON/OFF_circuit	%QX2.0.0		
	T1_CV	TIME	VAR	Timer_current_value			

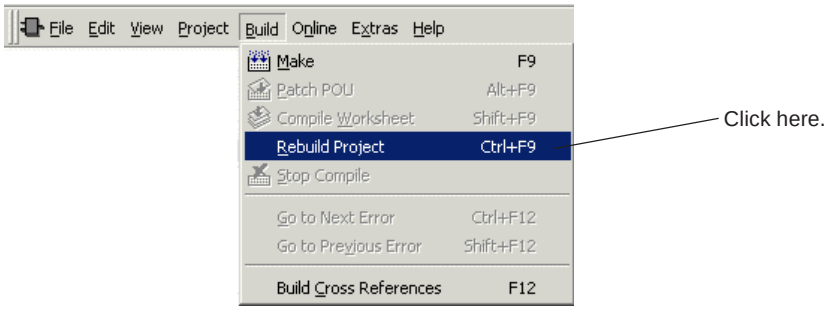
Section 1 Creating a New Project

1-6 Compiling a Project

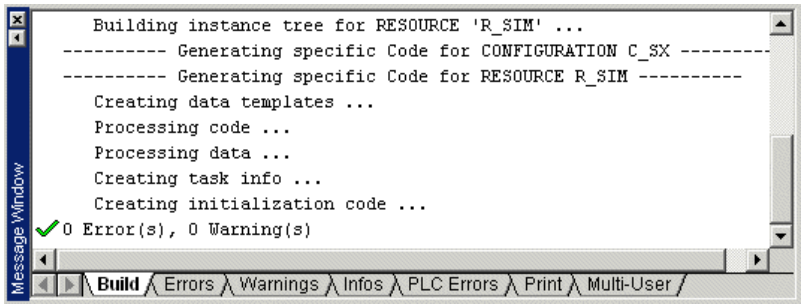
When you complete a project, compile it. By compilation, only data necessary for program execution are encoded.

<Operational procedure>

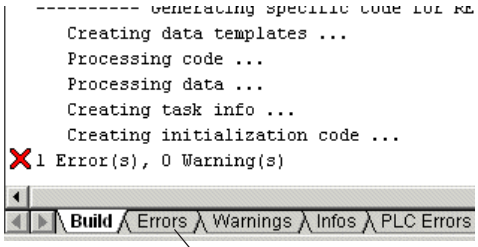
◆ Execute the [Rebuild Project] command in the [Build] menu on the D300win menu bar.

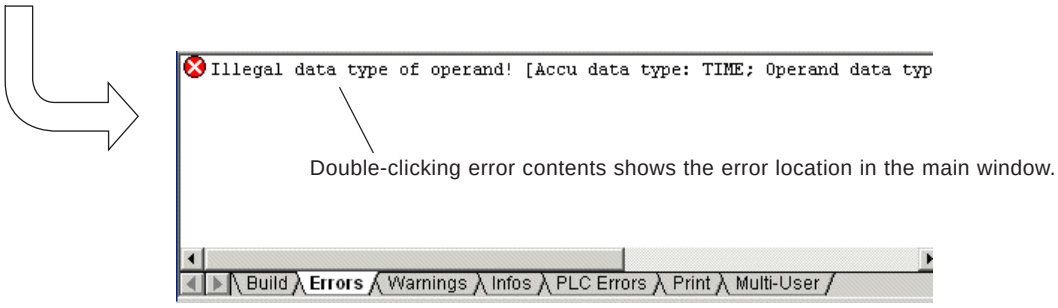


◆ Compilation of the project starts. Results of compilation are displayed in the “Message window”.



* When an error is found as a result of compilation, you need to correct it and compile the project until no error is found.





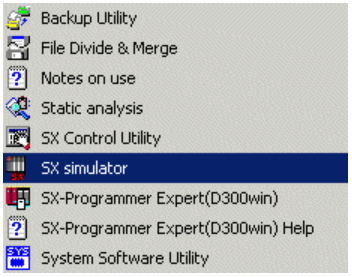
Section 1 Creating a New Project

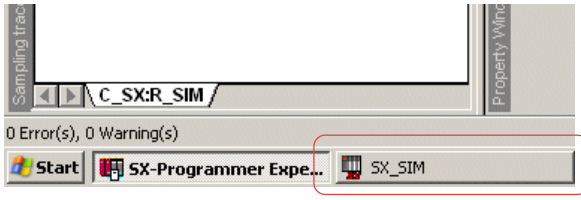
1-7 Downloading a Project

Download the project to the SX simulator after compilation completes.

<Operational procedure>

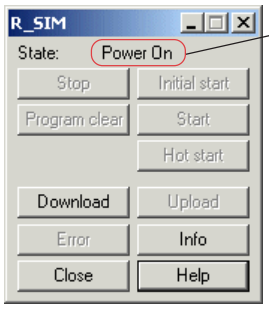
◆ Start the “SX simulator” first. Click the “SX simulator” in the D300win program menu to start the SX simulator.



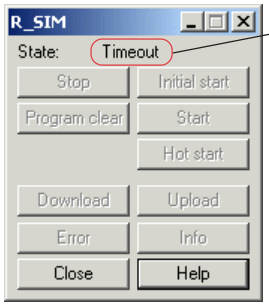


The SX simulator appears on the task bar.

◆ Click [Project Control Dialog] button to display the [R_SIM] dialog.



When the SX simulator is running, “Power On” is displayed in the state field.

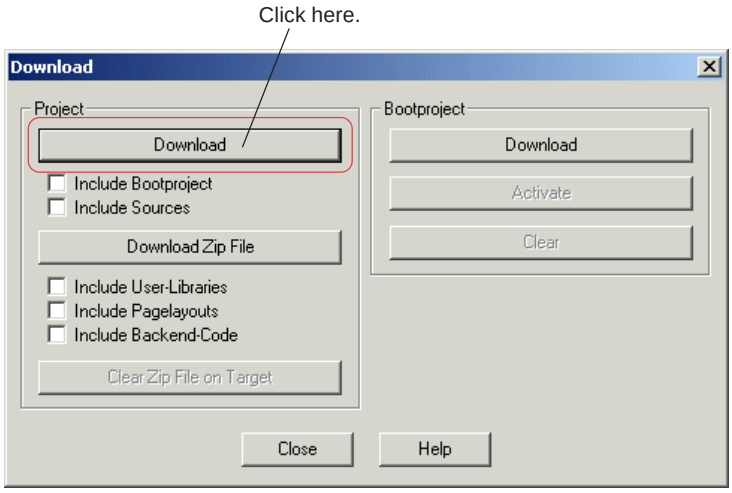


When the SX simulator is not running, “Timeout” is displayed in the state field.

Section 1 Creating a New Project

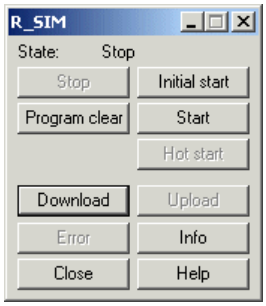
1-7 Downloading a Project

◆ Click the [Download] button on the [R_SIM] dialog to display the [Download] dialog. Clicking the [Download] button in the [Project] box starts downloading.



◆ When downloading is completed, the [R_SIM] dialog becomes as shown below. Clicking the [Start] button activates the project downloaded to the SX simulator.

<When simulator stops>



<When simulator is running>

Section 1 Creating a New Project

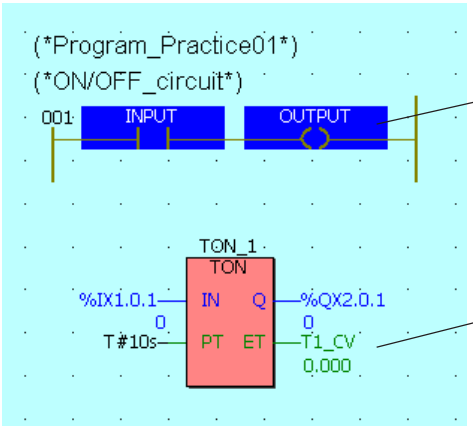
1-8 Monitoring and Checking Operation

Monitor the operating status of the project downloaded to the SX simulator.

1-8-1 Monitoring and checking operation on the code worksheet

<Operational procedure>

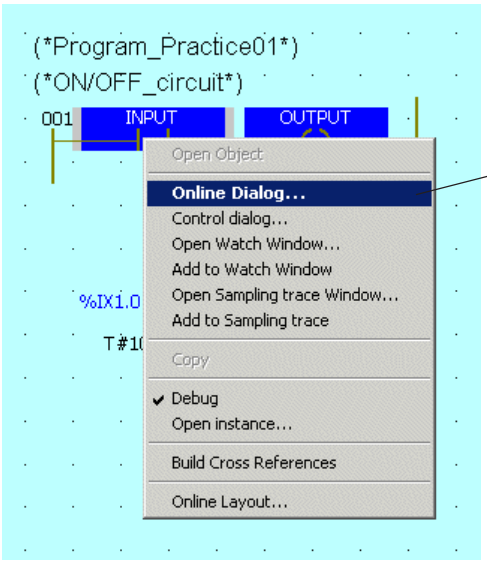
◆ Display the code worksheet and click the [Debug on/off] button to enter monitor mode.



BOOL type data is shown in blue when it is ON and in red when OFF.

Data other than BOOL type such as a current timer value is shown in green.

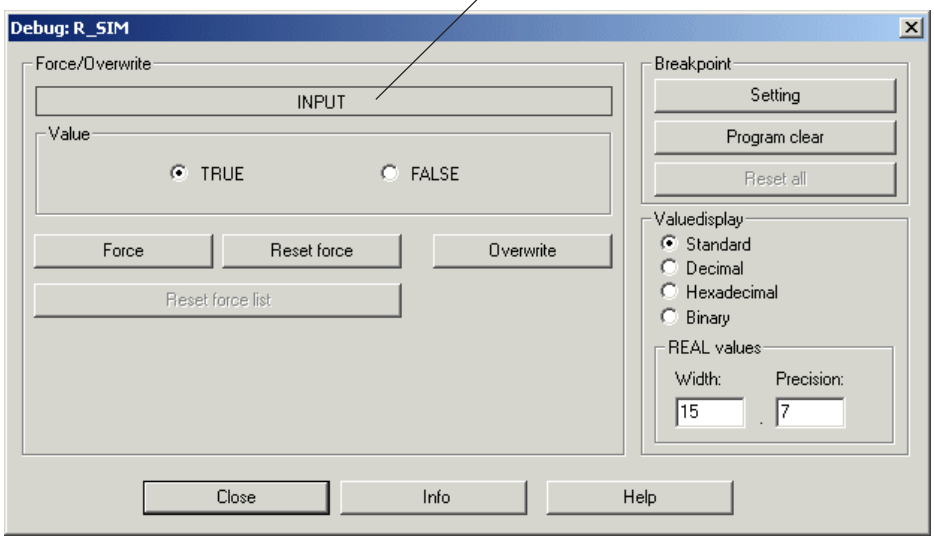
◆ Turn ON the variable “INPUT” to check the operation of the circuit. Right-click the variable “INPUT” and execute the [Online Dialog...] command. The [Debug] dialog is displayed.



Click here.

[Debug] dialog

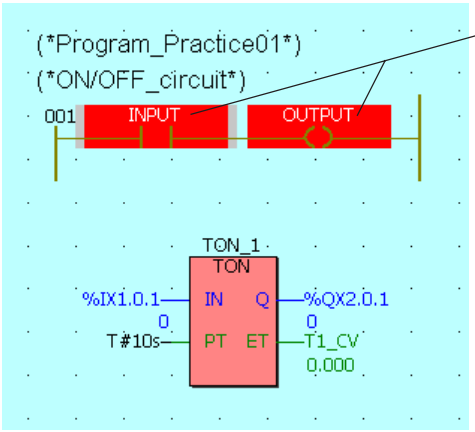
The target variable is indicated here.



Section 1 Creating a New Project

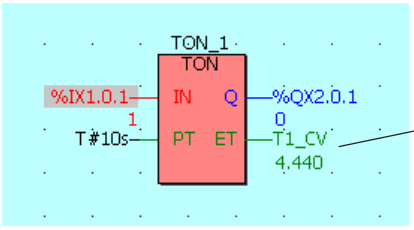
1-8 Monitoring and Checking Operation

◆ Check to see that TRUE is selected for the “Value” on the [Debug] dialog and then click the [Overwrite] button. The variable “INPUT” is turned ON and the monitor screen becomes as shown below:



If INPUT is turned ON (FALSE → TRUE), the symbols of the contacts are shown in red and OUTPUT is also turned ON (shown in red).

◆ Next, monitor the operation of the timer circuit. In the same manner, turn ON “%IX1.0.1”. The timer is activated.



You can monitor the current value.

1-8-2 Monitoring on the variable worksheet

If you display the variable worksheet in monitor mode (double-click the variable worksheet in the project tree), you can monitor operations on it.

Variable	Online value	Data type	Usage	Comment	Address	Init	RETAIN
Default							
TON_1		TON	VAR	TimerFB_instance			☐
INPUT	TRUE	BOOL	VAR	ON/OFF_network	%IX1.0.0		☐
OUTPUT	TRUE	BOOL	VAR	ON/OFF_network	%QX2.0.0		☐
T1_CV	10.000	TIME	VAR	Timer_current_value			☐

Note: You cannot monitor “IN” terminal (timer activation input) and “Q” timer output terminal of the timer circuit on the variable worksheet because addresses are directly specified for them.

Section 2 Inserting a POU

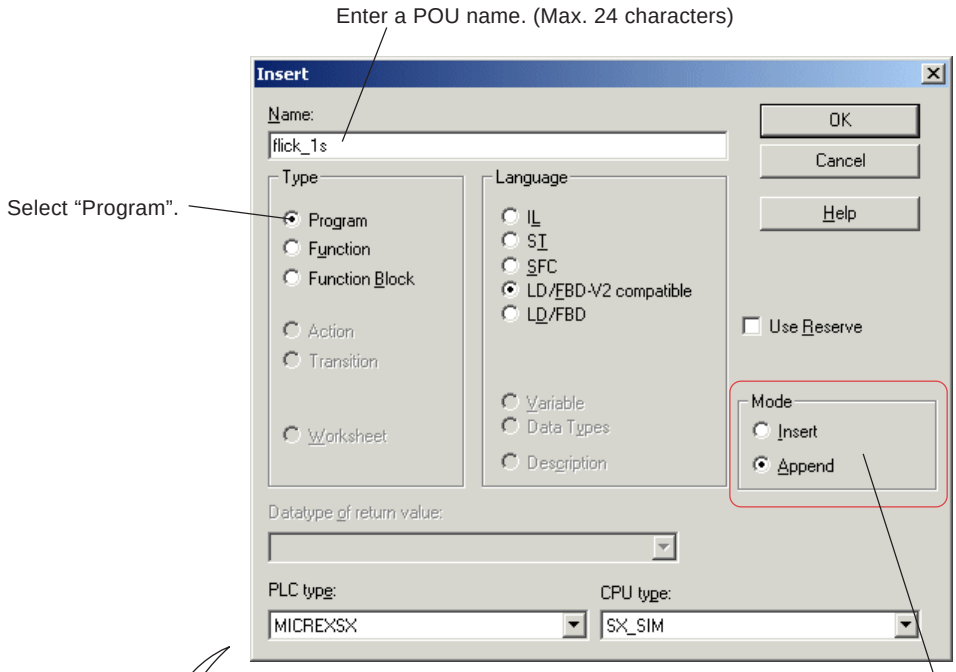
This section explains how to insert a POU into the project tree and create a program. One of the features of MICREX-SX is that programs are structured so that efficiency of programming and debug can be improved. To program one control dividing each function into units of POUs (Program Organization Unit) is the first step toward structured programming.

2-1 Inserting a POU in the Project Tree

Insert a new project in the current project tree.

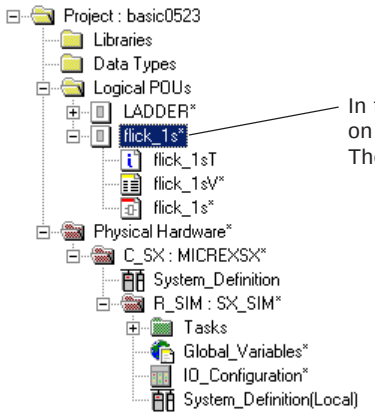
<Operational procedure>

◆ Click the POU “LADDER” and press the <Insert> key. The [Insert] dialog is displayed.



When “Append” is selected, a new POU is added in the rear of the POU “LADDER”, when “Insert” is selected, in the front of the POU “LADDER”.

After setting all necessary items, click the [OK] button. A POU “flick_1s” is inserted as shown below.

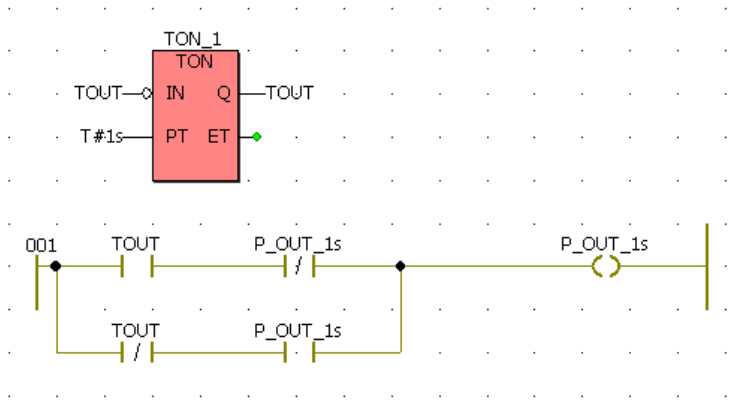


In this example, a “flicker” circuit which turns on and off at one-second intervals is created. Therefore, the POU is named “flick_1s”.

Section 2 Inserting a POU

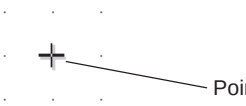
2-2 Creating a Flicker Circuit

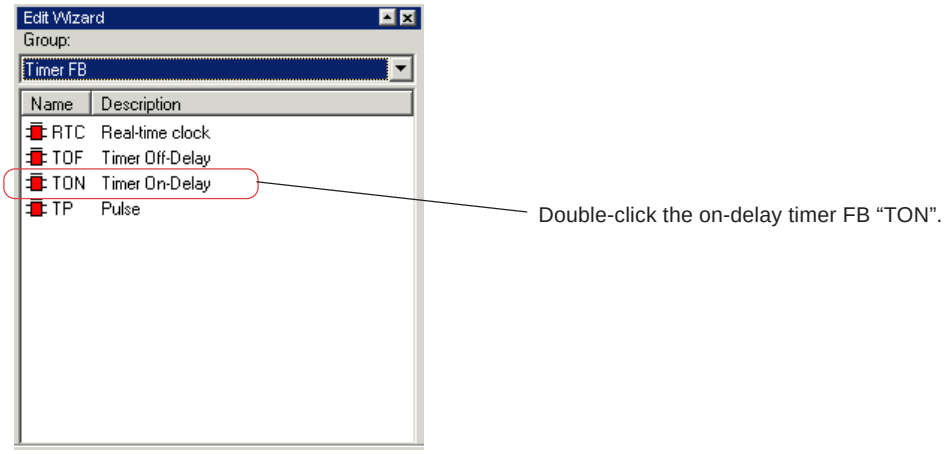
Create a flicker circuit which turns on and off at one-second intervals. As a general manner, the circuit is created by combining a timer circuit and flip-flop circuit. In this section, create a circuit shown below:

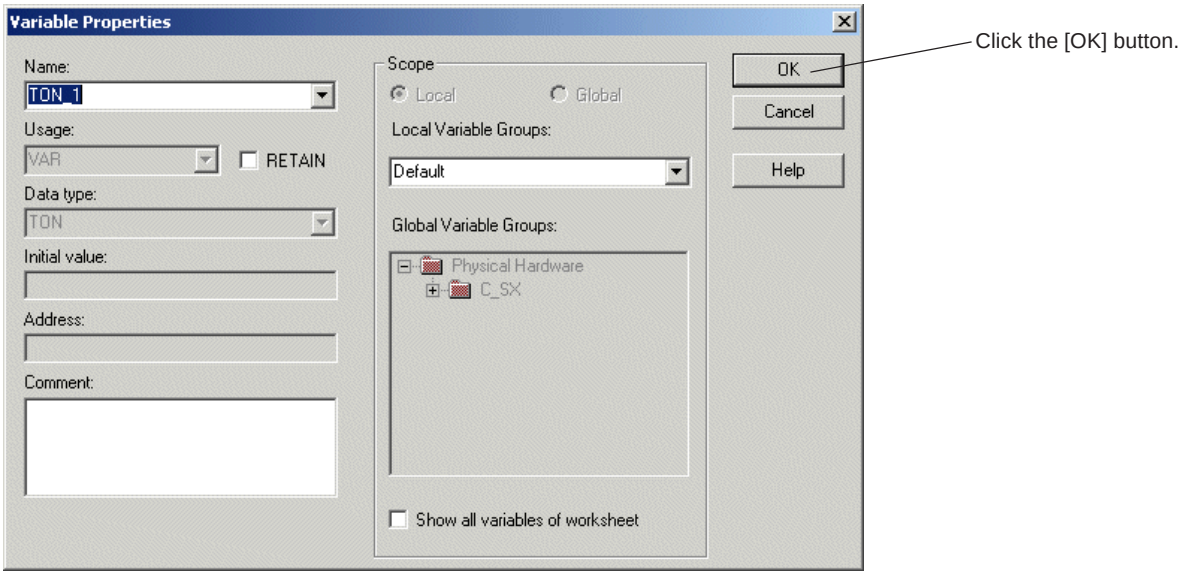


<Operational procedure>

◆ Create a timer FB on the code worksheet of POU “flick_1s”. Click an arbitrary point on the code worksheet to place the pointer there and paste a timer FB.



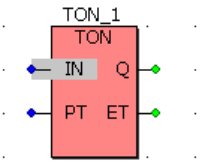




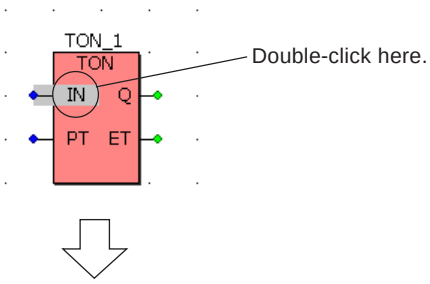
Section 2 Inserting a POU

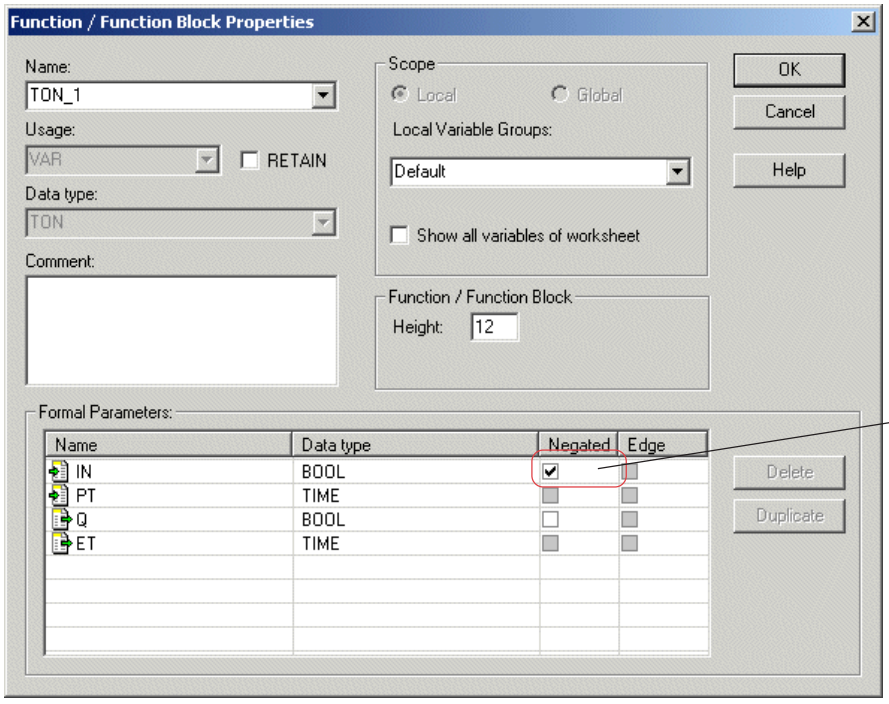
2-2 Creating a Flicker Circuit

A timer FB is pasted on the code worksheet.



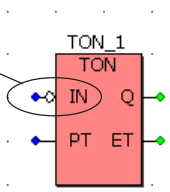
◆ Connect necessary objects (variable or timer set value) to the terminals of the timer FB. Double-click the IN terminal of the timer FB to display the [Variable Properties] dialog.





Set this checkbox to ON and click the [OK] button.

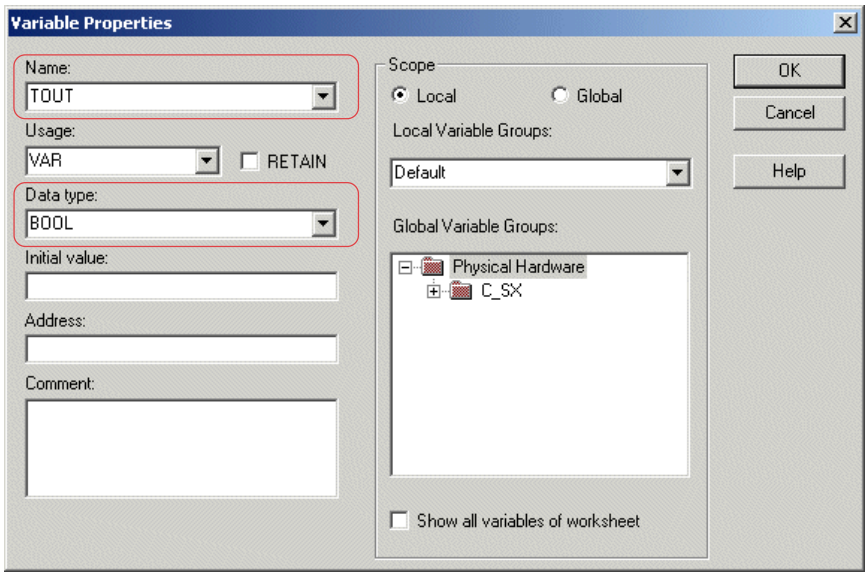
The IN terminal is changed into a negated terminal.



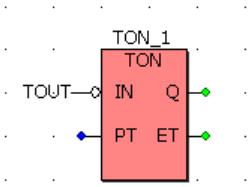
Section 2 Inserting a POU

2-2 Creating a Flicker Circuit

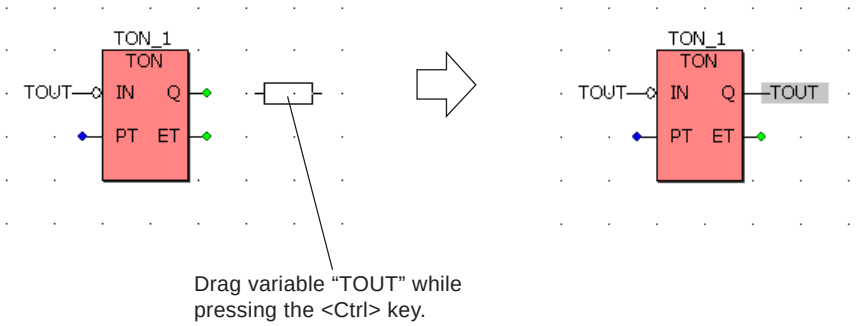
◆ Connect variable “TOUT” to “IN” and “Q” terminals of the timer. Double-clicking the IN terminal of the timer FB (the outside of the FB) displays the [Variable Properties] dialog. Enter “TOUT” in the name box, select “BOOL” for the data type and then click the[OK] button.



◆ Variable “TOUT” is connected to the IN terminal of the timer as shown below.



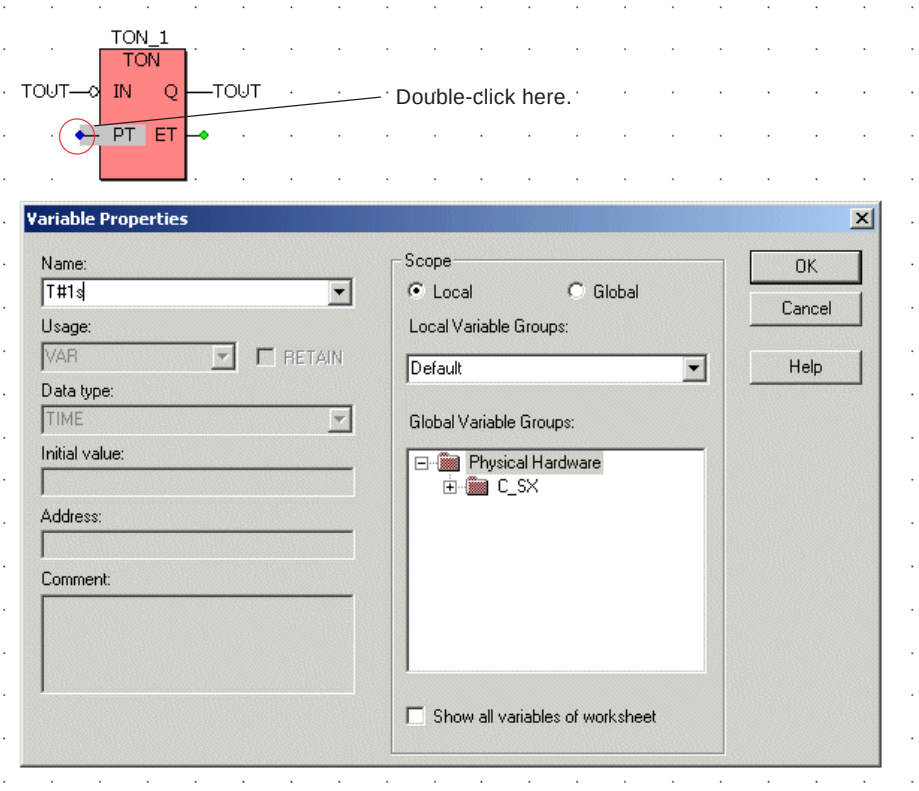
◆ Select variable “TOUT” connected to the IN terminal, drag it while pressing the <Ctrl> key and connect it to the Q terminal

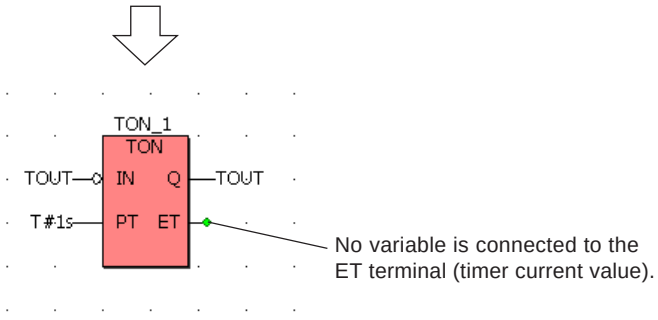


Section 2 Inserting a POU

2-2 Creating a Flicker Circuit

◆ Enter a timer set value into the PT terminal. Double-click the PT terminal and directly enter a constant in the name box on the [Variable Properties] dialog. After entering it, click the [OK] button.



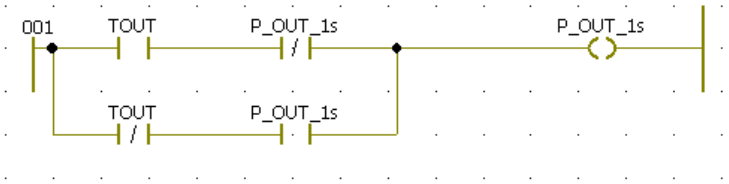


* The above circuit is a pulse output circuit in which the time-out signal itself turns OFF the timer activation input so that the time-out signal (TOUT) outputs one pulse each cycle of the timer set value.

Section 2 Inserting a POU

2-2 Creating a Flicker Circuit

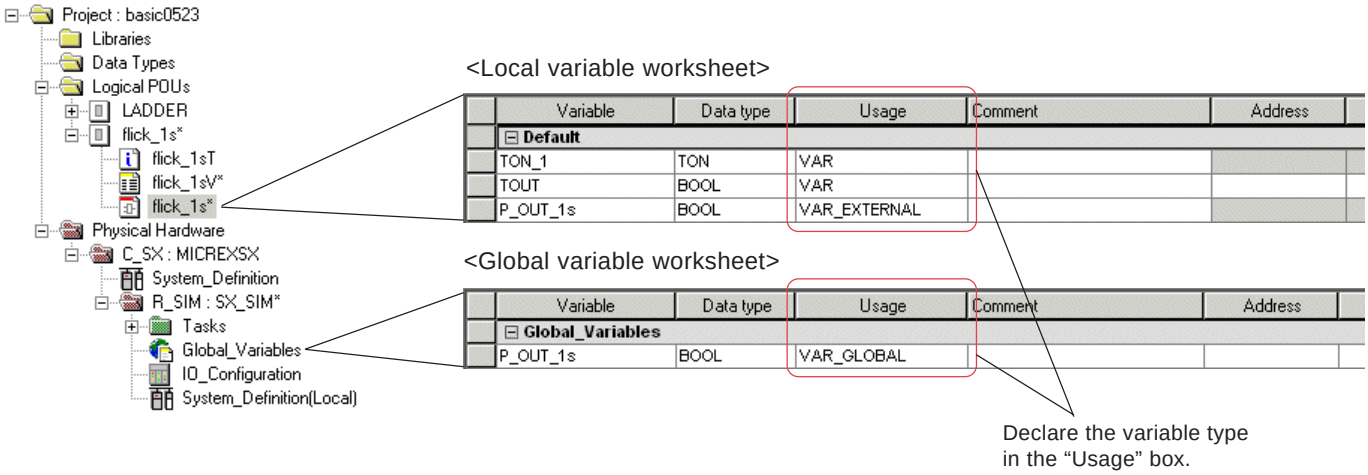
◆ Next, create a flip-flop circuit. By using TOUT for the input signal of the flip-flop, a flicker circuit that outputs ON/OFF signals at one-second intervals can be created.



* Variable “P_OUT_1s” to become the output of the flicker must be a global variable.

<Variable type>

There are two types of variables: local variables and global variables. Local variables can be used only in a POU in which the variable is defined, whereas global variables can be used in any POU in the same project.
The worksheet of POU “flick_1s” is as shown below:



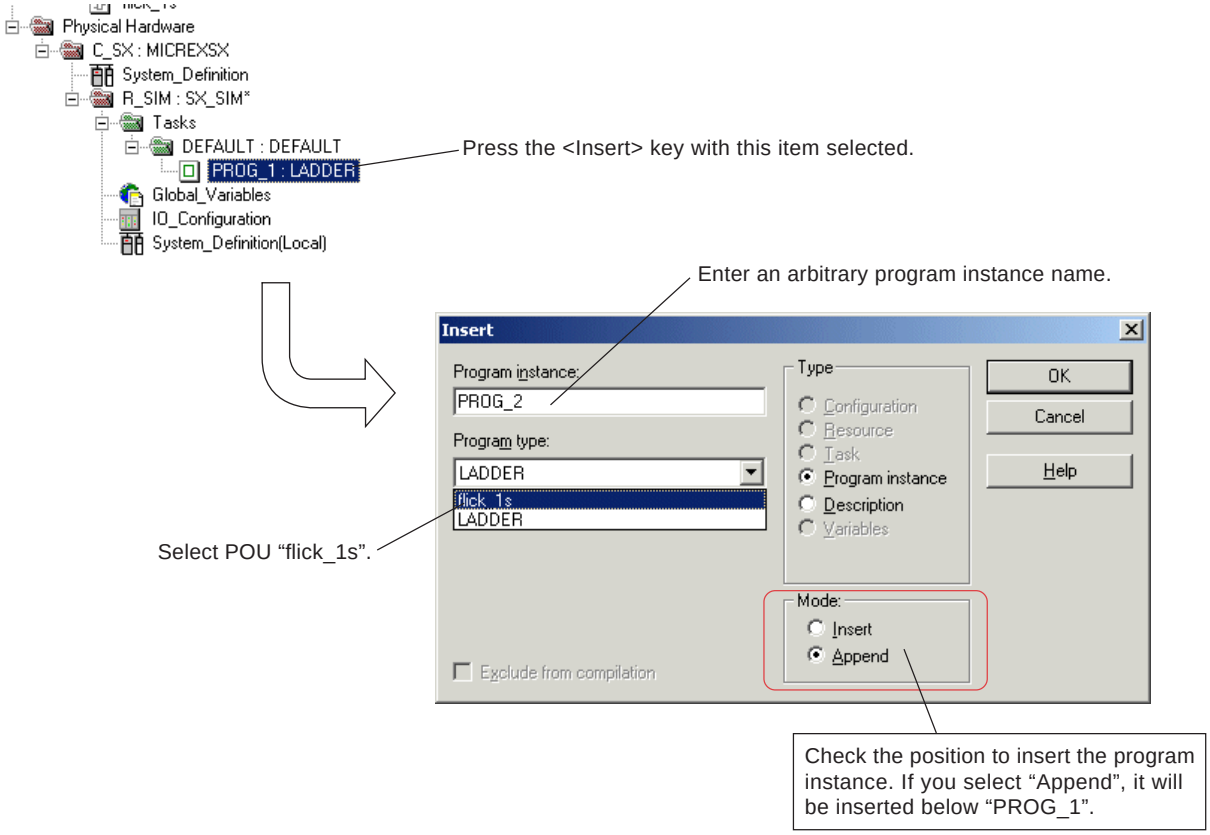
Section 2 Inserting a POU

2-3 Assigning a POU to a Task

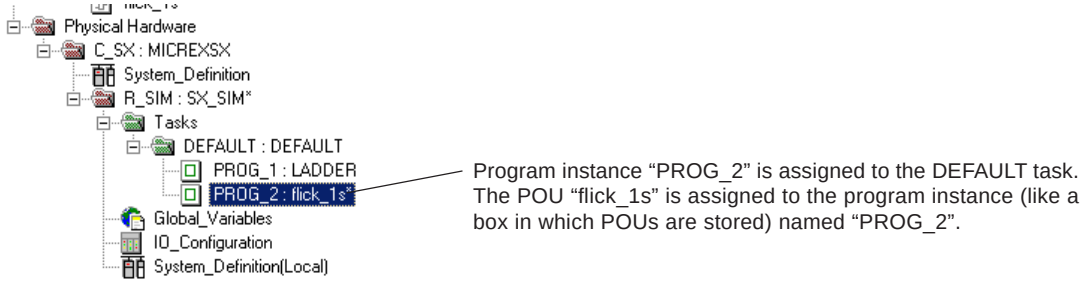
Assign the created POU “flick_1s” to a task. The created POU needs to be assigned to a task and compiled.

<Operational procedure>

◆ Select “PROG_1” under the DEFAULT task in the project tree and press the <Insert> key of the personal computer. The [Insert] dialog is displayed.



◆ After setting all necessary items on the [Insert] dialog, click the [OK] button. The POU “flick_1s” is assigned to the task as shown below.



<Order of POU execution>

POUs assigned to the same task are executed in order in which they are assigned. For the example above, PROG_1:LADDER is assigned first and then PROG_2:flick_1s. Therefore, POU “LADDER” is executed first and then POU “flick_1s”.
When assigning a POU to a task, be aware of the assigning position (order).

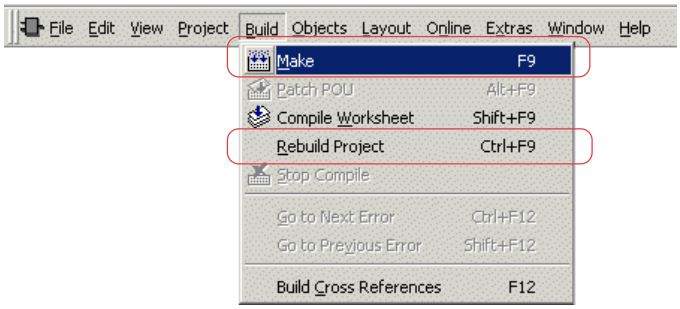
Section 2 Inserting a POU

2-4 Checking Operation of the Created POU

Check the operation of the created POU “flick_1s”. After compiling the project and downloading it to the SX simulator, check its operation by using the monitoring function.

<Operational procedure>

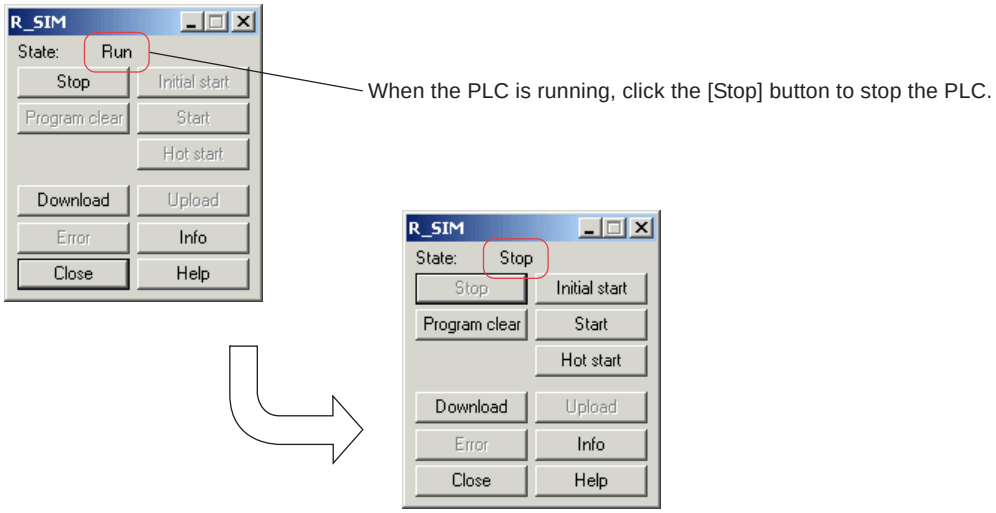
◆ Compile the project by executing the [Make] or [Rebuild Project] command in the [Build] menu.



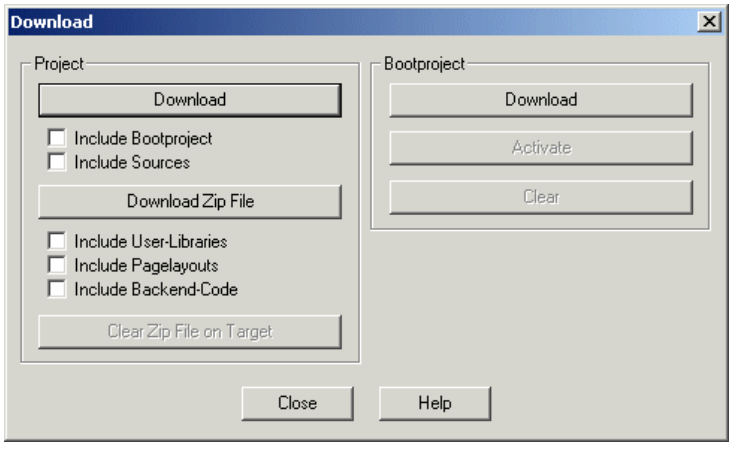
*** “Make” and “Rebuild Project”**

The [Make] command does not compile for a POU that has already been complied and not changed. Therefore, the time for compilation can be shortened. The [Rebuild Project] command compiles the whole project again. Although it takes time compared with [Make], messages such as warning are always displayed. You are recommended to compile the project again with the [Rebuild Project] command when you complete creating it.
Try to create a project that contains no warning.

◆ When compilation is completed, download the project to the SX simulator.
Click the [Project Control Dialog] button to display the [R_SIM] dialog.



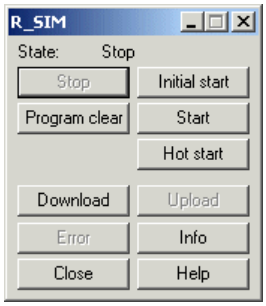
◆ After the PLC is stopped, click the [Download] button to display the [Download] dialog.



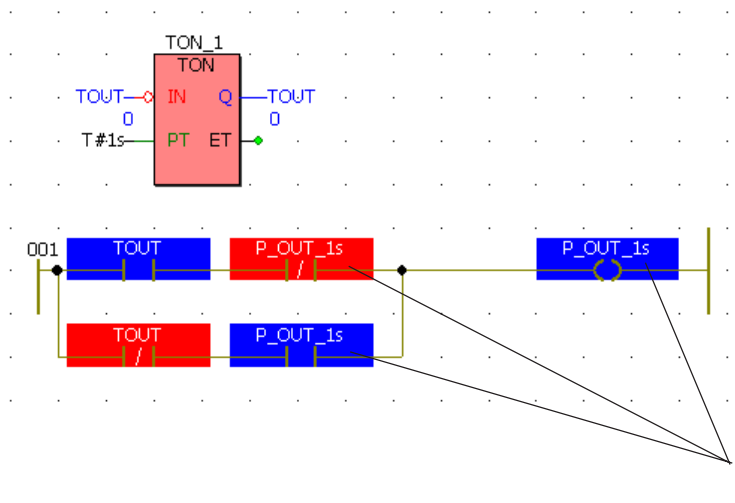
Section 2 Inserting a POU

2-4 Checking Operation of the Created POU

◆ Clicking the [Download] button downloads the project to the SX simulator and displays the [R_SIM] dialog. Click the [Start] button to activate the program.



◆ Display the code worksheet and click the [Debug on/off] button to monitor the operation.



Check to see that variable “P_OUT_1s” turns ON and OFF at one-second intervals.

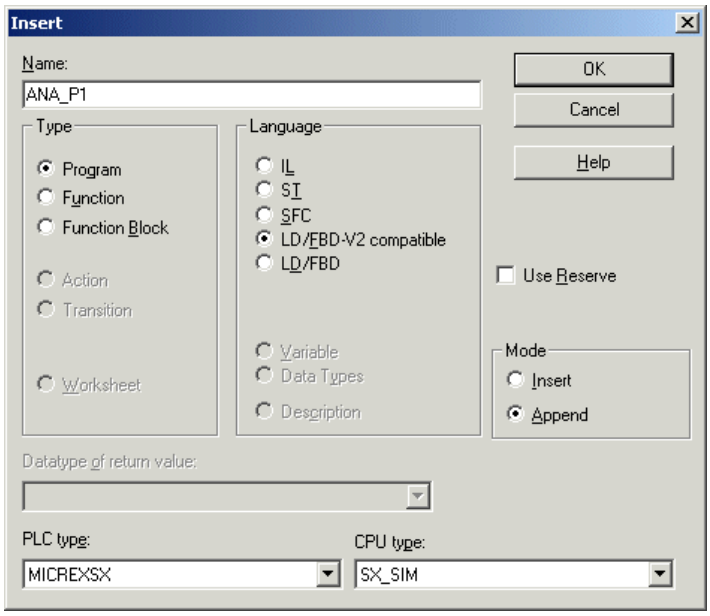
Section 2 Inserting a POU

2-5 Creating an Annunciator Circuit

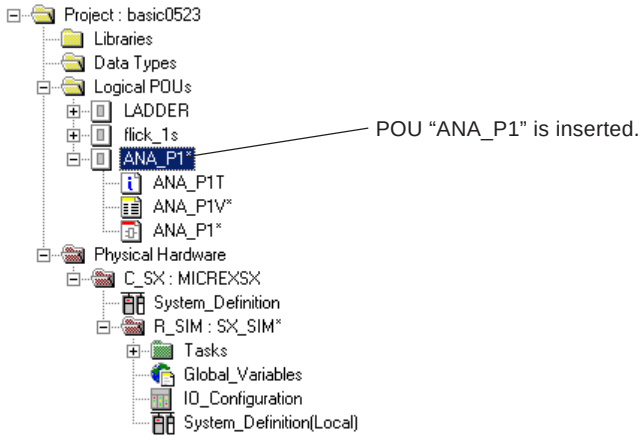
In the same manner as in POU “flick_1s”, create a POU of annunciator (warning) operation.

<Operational procedure>

◆ Click POU “flick_1s” in the project tree and press the <Insert> key to display the [Insert] dialog.



◆ After setting as shown above, click the [OK] button.

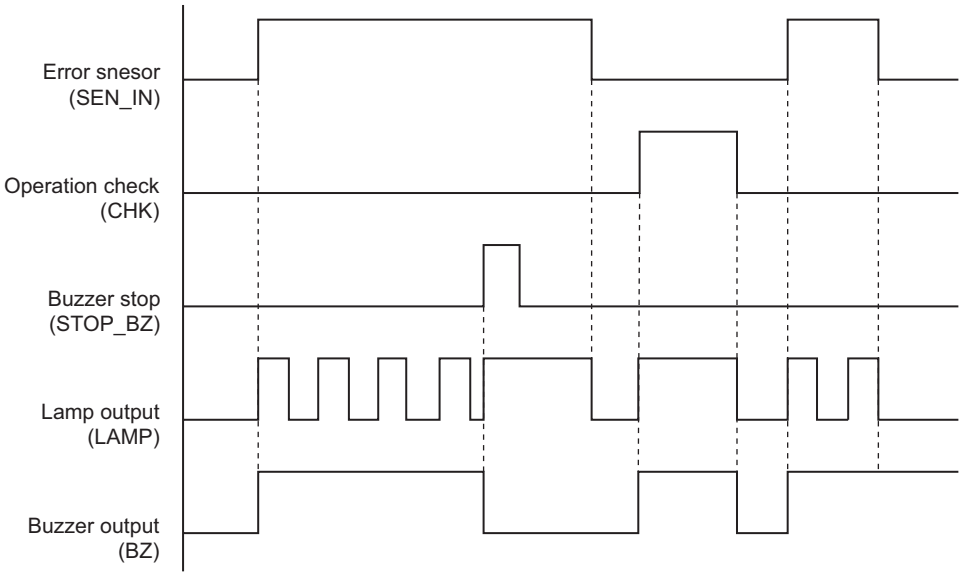


Section 2 Inserting a POU

2-5 Creating an Annunciator Circuit

<Operation of annunciator>

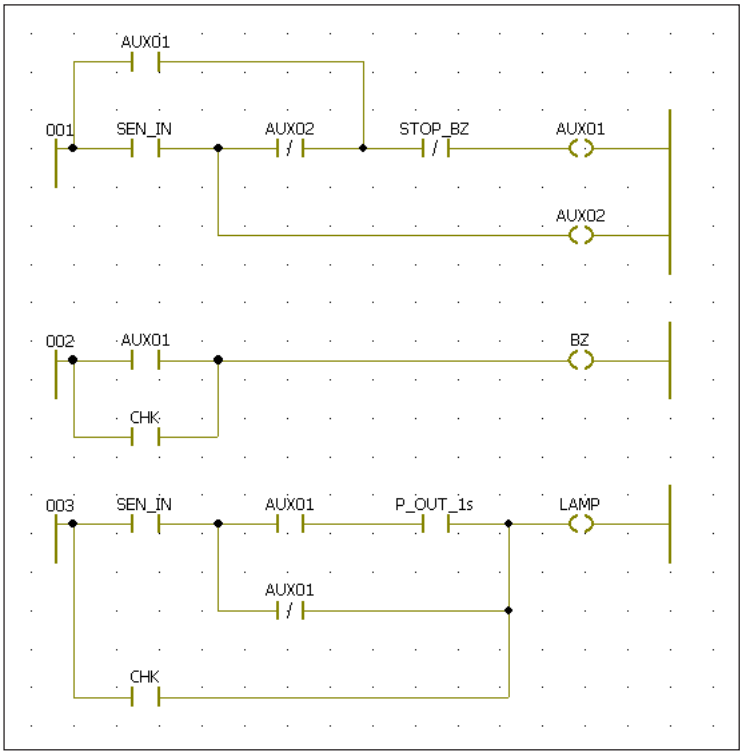
The annunciator circuit has various operation patterns. In this example, create a circuit that operates in a pattern shown in the time chart below.



* Alphanumerics in parentheses are variable names.

◆ The code worksheet and variable worksheet become as shown below.

<Code work sheet>



<Variable worksheet>

	Variable	Data type	Usage	Comment	Address	Init	RETAIN
	⊟ **Default**						
	AUX02	BOOL	VAR				☐
	AUX01	BOOL	VAR				☐
	P_OUT_1s	BOOL	VAR_EXTERNAL				☐
	SEN_IN	BOOL	VAR		%IX1.0.7		☐
	STOP_BZ	BOOL	VAR		%IX1.0.8		☐
	CHK	BOOL	VAR		%IX1.0.9		☐
	BZ	BOOL	VAR		%QX2.0.8		☐
	LAMP	BOOL	VAR		%QX2.0.9		☐

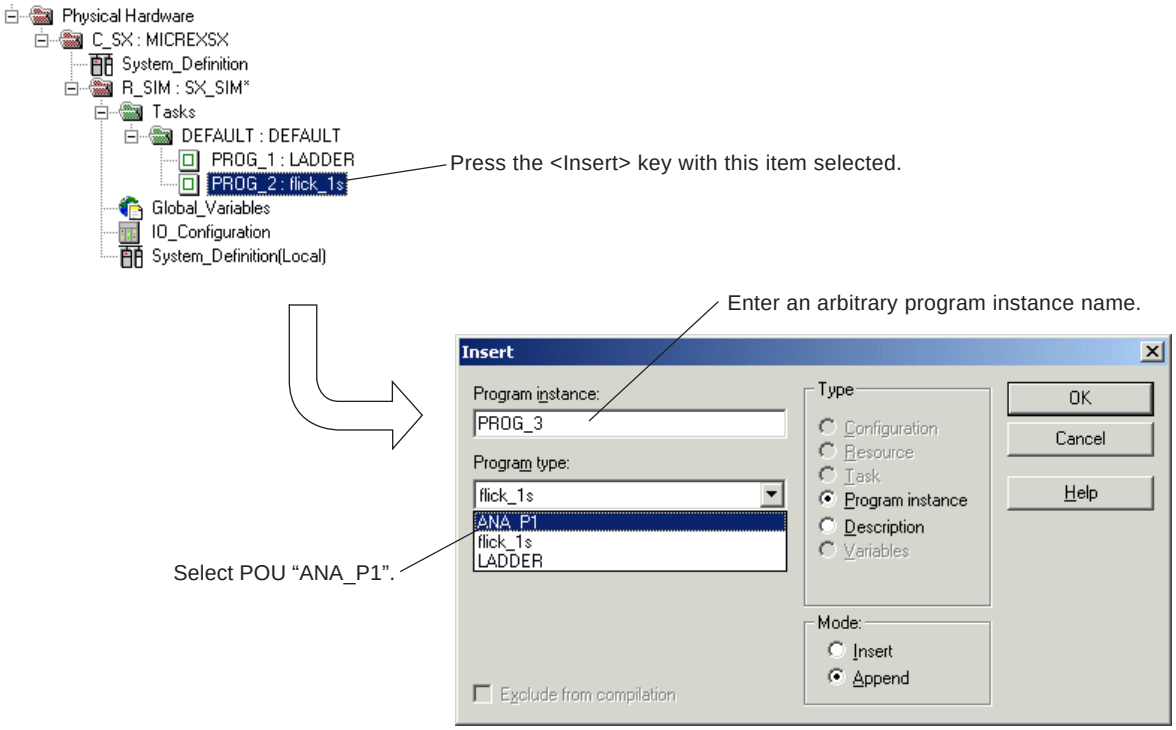
Section 2 Inserting a POU

2-6 Checking Operation

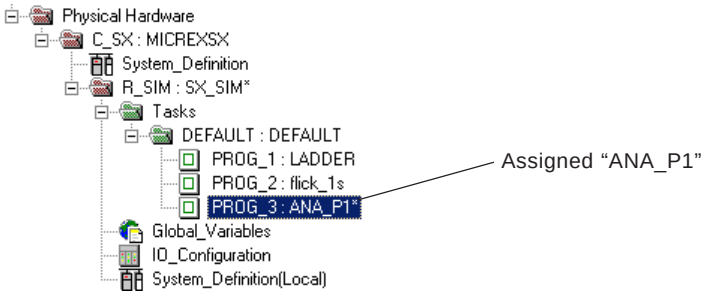
After POU “ANA_P1” is created, assign the POU to a task, compile and download it to the SX simulator. Then, monitor and check the operation on D300win.

<Operational procedure>

◆ Select “PROG_2” under the DEFAULT task in the project tree and press the <Insert> key of the personal computer. The [Insert] dialog is displayed.



◆ After setting all necessary items on the [Insert] dialog, click the [OK] button. The POU “ANA_P1” is assigned to the task as shown below.



◆ Next, compile the project with the [Make] or [Rebuild Project] command in the [Build] menu. After compilation is completed, download the project to the SX simulator.

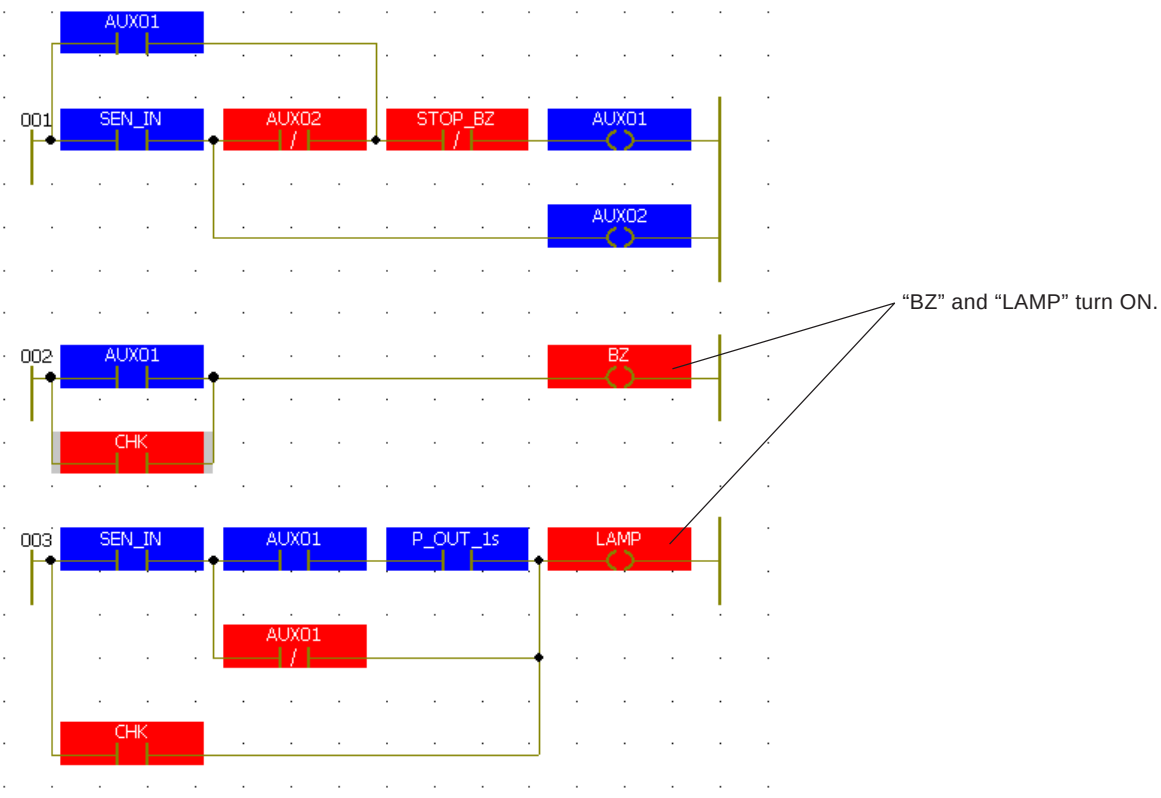
◆ After the project is downloaded, display the code worksheet and click the [Debug on/off] button to monitor the operation.

Section 2 Inserting a POU

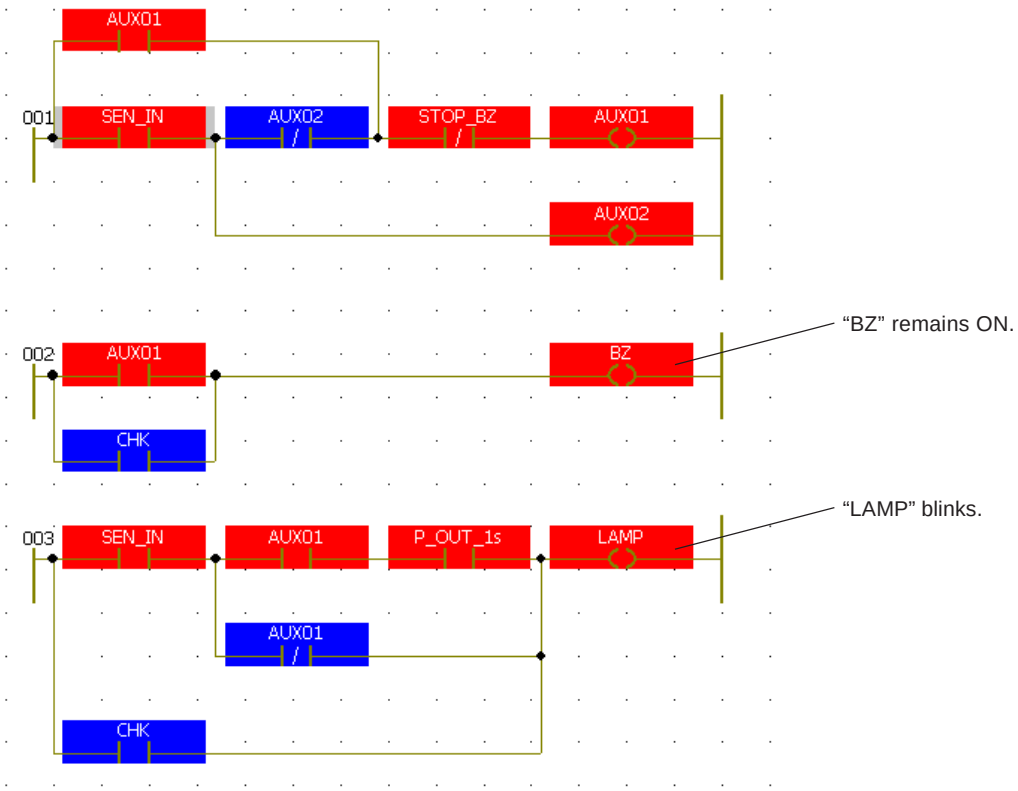
2-6 Checking Operation

<Operation of annunciator circuit>

1) If “CHK” is turned ON...



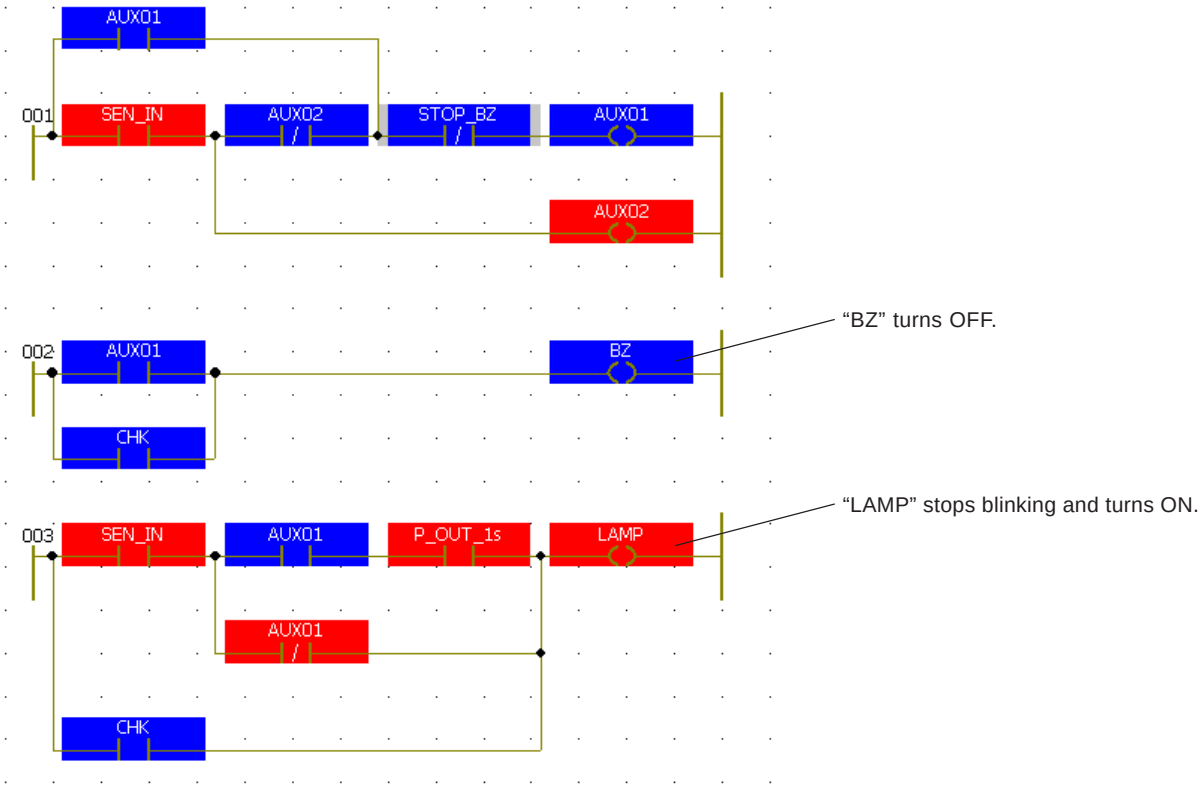
2) If “CHK” is turned OFF and “SET_IN” is turned ON...



Section 2 Inserting a POU

2-6 Checking Operation

3) If “STOP_BZ” is turned ON from the state of 2)...



“BZ” turns OFF.

“LAMP” stops blinking and turns ON.

* By turning ON and OFF each input element on the online dialog, check the operation of the circuit.

Section 3 Creating a User FB

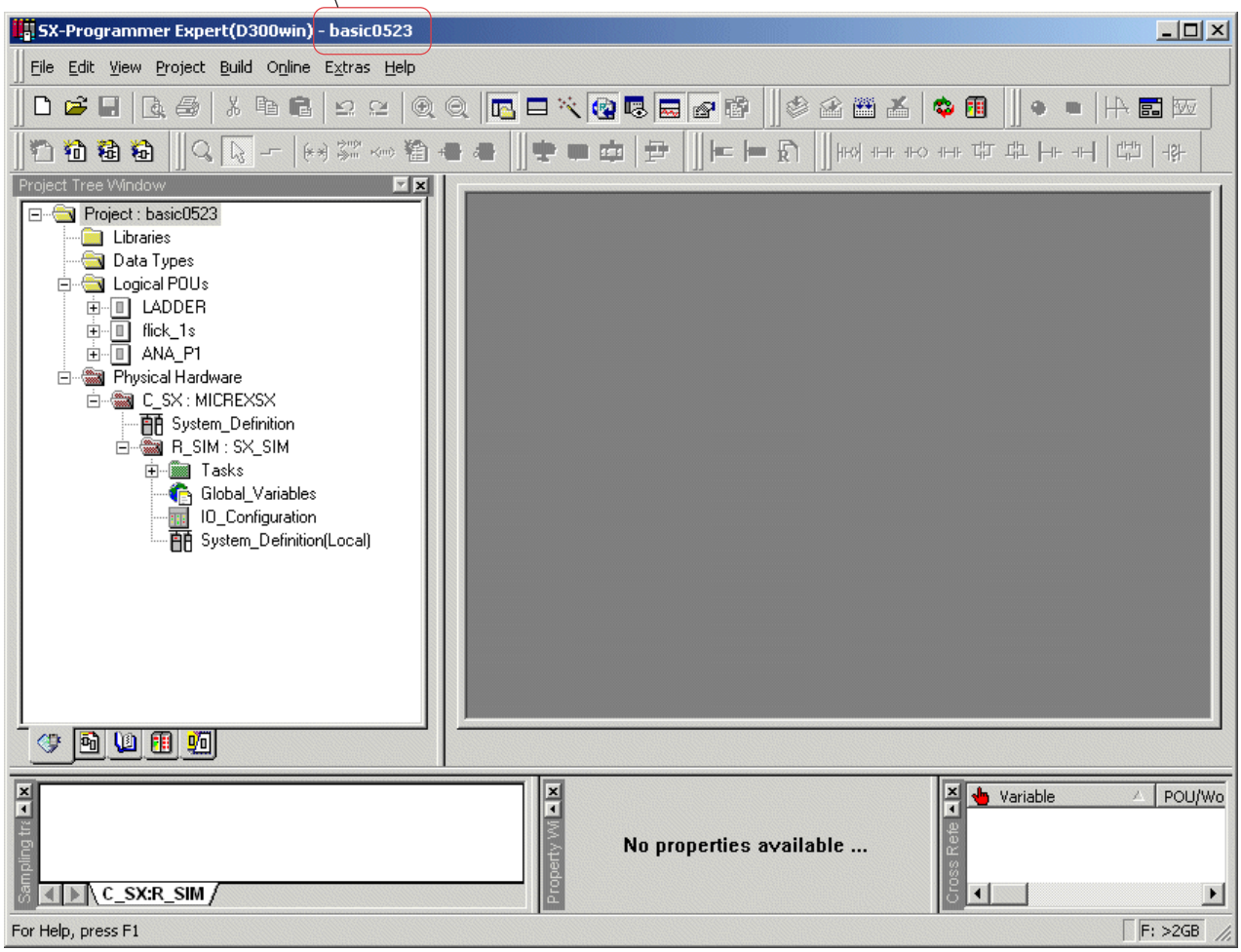
This section explains how to make the POU “flick_1s” and “ANA_P1” created in Section 2 into function blocks.

3-1 Creating a Library Project

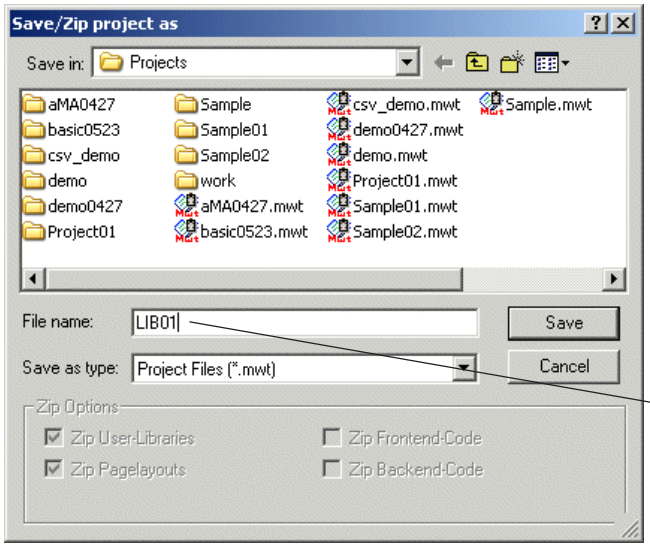
Prepare a library project (a project in which user FBs are collected) from the current project (created in Section 2) by using the [Save Project As / Zip Project As...] command.

<Operational procedure>

◆ Open the project created in Section 2.



◆ Execute the [Save Project As / Zip Project As...] command to display the [Save/Zip project as] dialog. Enter a file name (project name) and click the [Save] button.

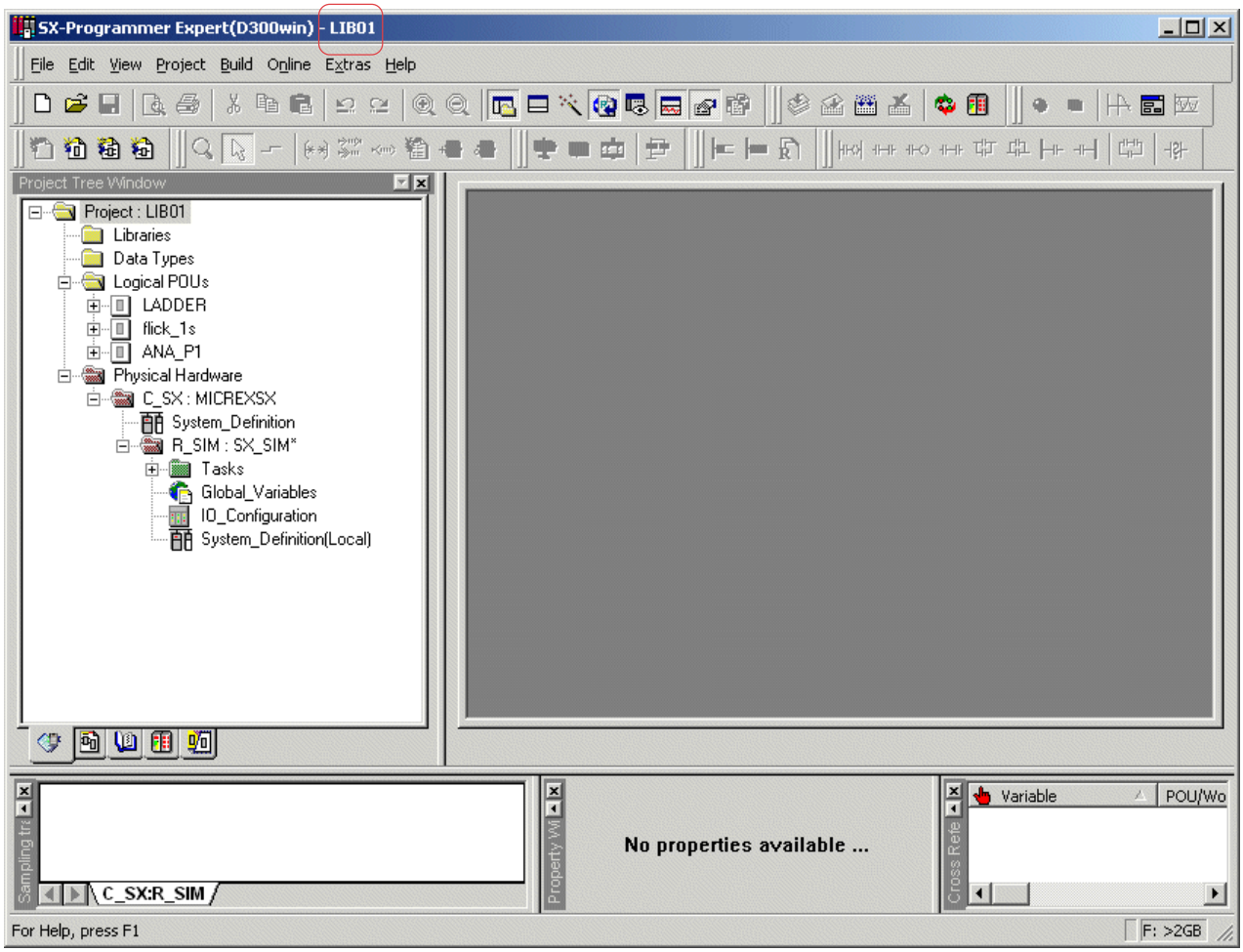


Enter a project name here.
In this example, “LIB01” is entered.

Section 3 Creating a User FB

3-1 Creating a Library Project

◆ The currently opened project is changed to “LIB01”. The source project remains as is.
The new project has exactly the same contents as the source project.



* How to change this project and make POU “flick_1s” and “ANA_P1” into FBs is explained below. They are still programs here.

Section 3 Creating a User FB

3-2 Changing a POU

Change a POU currently created as a “program” to an FB (function block).

<Operational procedure>

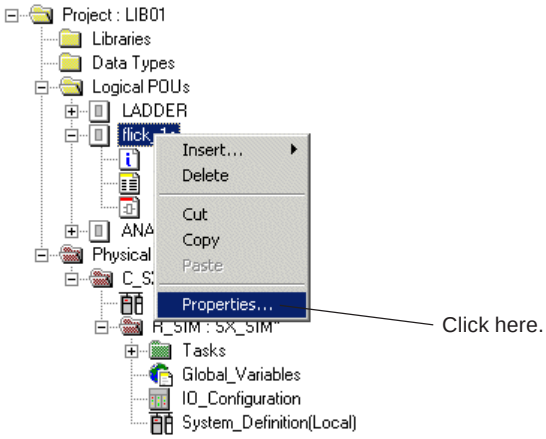
◆ First, delete addresses (AT specification) assigned to variables in the POU that is to be made into FB. Open the variable worksheet of POU “ANA_P1” and delete the addresses.

	Variable	Data type	Usage	Comment	Address	Init	RETAIN
	Default						
	AUX02	BOOL	VAR				
	AUX01	BOOL	VAR				
	P_OUT_1s	BOOL	VAR_EXTERNAL				
	SEN_IN	BOOL	VAR		%IX1.0.7		
	STOP_BZ	BOOL	VAR		%IX1.0.8		
	CHK	BOOL	VAR		%IX1.0.9		
	BZ	BOOL	VAR		%QX2.0.8		
	LAMP	BOOL	VAR		%QX2.0.9		

Delete these addresses.

	Variable	Data type	Usage	Comment	Address	Init	RETAIN
	Default						
	AUX02	BOOL	VAR				
	AUX01	BOOL	VAR				
	P_OUT_1s	BOOL	VAR_EXTERNAL				
	SEN_IN	BOOL	VAR				
	STOP_BZ	BOOL	VAR				
	CHK	BOOL	VAR				
	BZ	BOOL	VAR				
	LAMP	BOOL	VAR				

◆ Then, change the properties of the POU. Right-click the POU in the project tree and then execute the [Properties] command. The [Properties...] dialog is displayed. (in this example, POU “flick_1s”).

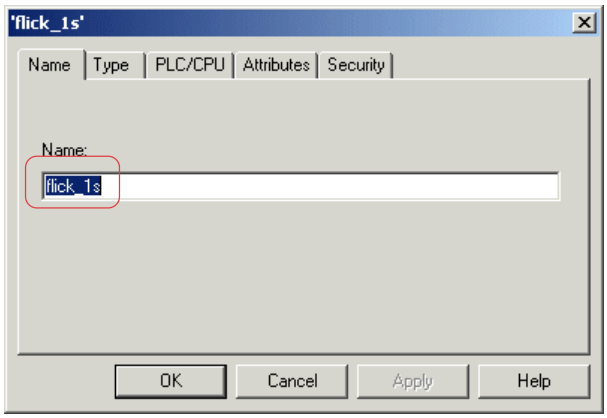


Section 3 Creating a User FB

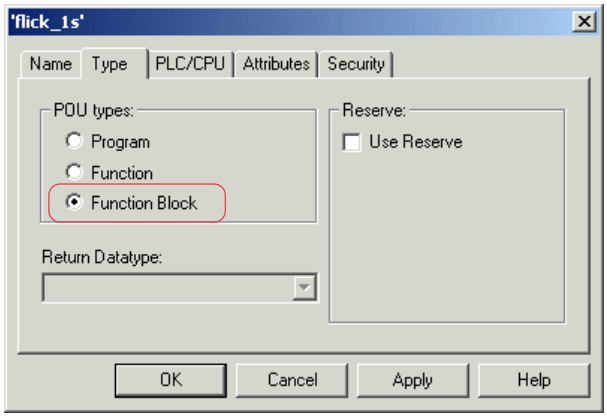
3-2 Changing a POU

◆ By clicking a corresponding tab, you can change the active tab window.

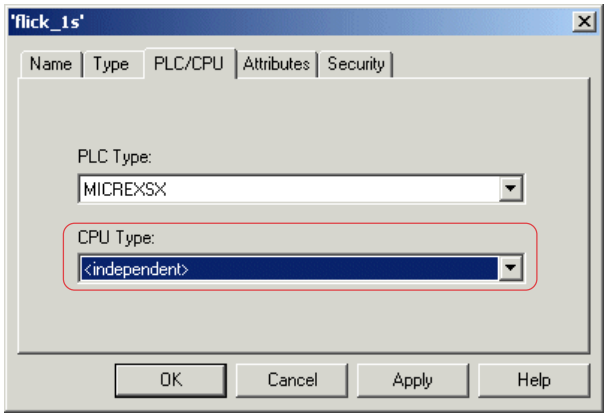
<[Name] tab window>
A POU name entered here becomes the FB name as is.



<[Type] tab window>
Select “Function block” for “Type”.



<[PLC/CPU] tab window>
Select “<independent>” for “CPU Type”.



<< FYI >> About FB name

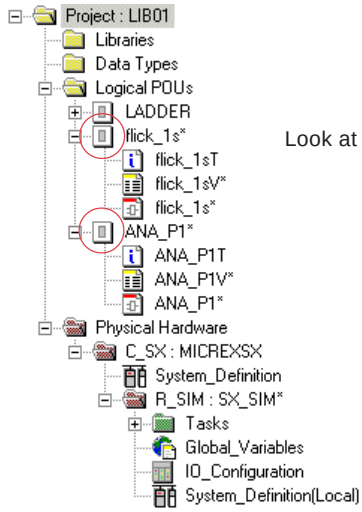
Since the POU name becomes the FB name as is, you need to give a name with care. FB names should be determined such that the user can imagine how the FB works from the name.

Section 3 Creating a User FB

3-2 Changing a POU

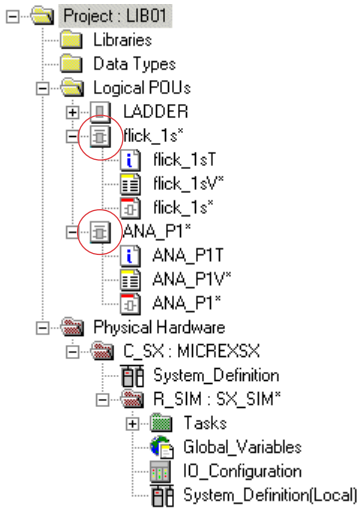
◆ After setting all necessary items, click the [OK] button. The POU is changed into “FB” type. Icons for the FB type is as shown below:

<“Program” type POU>



Look at the shapes of the icons!

<“FB” type POU>



◆ Next, to determine terminals of the FB, change the variable worksheet.

<Variable worksheet of FB “flick_1s”>

FB “flick_1s” has no input terminal. Only an output terminal that outputs ON/OFF at one-second intervals is contained.

Variable	Data type	Usage	Comment	Address	Init	RETAIN
Default						
TON_1	TON	VAR				☐
TOUT	BOOL	VAR				☐
P_OUT_1s	BOOL	VAR_EXTERNAL				☐

Note: P_OUT_1s to become an output terminal is a global variable on the source project. The output terminal is not a variable, therefore, delete the declaration of the global variable.

Variable	Data type	Usage	Comment
Global_Variables			
P_OUT_1s	BOOL	VAR_GLOBAL	

Variable	Data type	Usage	Comment	Address	Init	RETAIN
Default						
TON_1	TON	VAR				☐
TOUT	BOOL	VAR				☐
P_OUT_1s	BOOL	VAR_OUTPUT				☐

Select “VAR_OUTPUT” for the usage.

Section 3 Creating a User FB

3-2 Changing a POU

<Variable worksheet of FB “ANA_P1”>

In FB “ANA_P1”, variables that were assigned to input addresses and used as global variables are changed into input terminals, and variables assigned to output addresses are changed into output variables.

<Variable worksheet of the source project>

Variable	Data type	Usage	Comment	Address
Default				
AUX02	BOOL	VAR		
AUX01	BOOL	VAR		
P_OUT_1s	BOOL	VAR_EXTERNAL		
SEN_IN	BOOL	VAR		%IX1.0.7
STOP_BZ	BOOL	VAR		%IX1.0.8
CHK	BOOL	VAR		%IX1.0.9
BZ	BOOL	VAR		%QX2.0.8
LAMP	BOOL	VAR		%QX2.0.9

→ Input terminals

→ Output terminals

* These address have already been deleted.

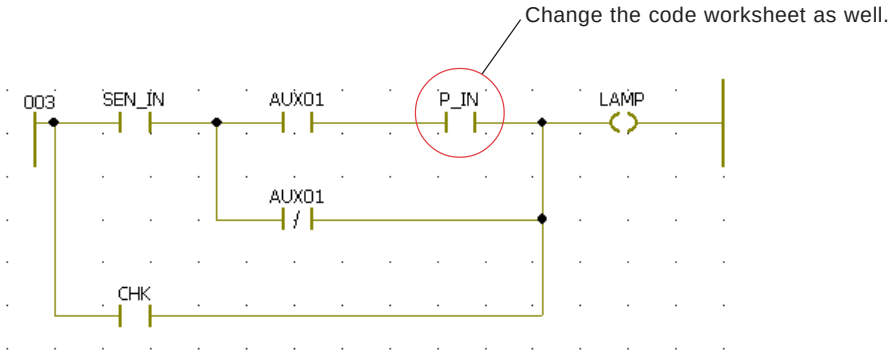
<Variable worksheet of the current project “LIB01”>

Variable	Data type	Usage	Comment	Address	Init	RETAIN
Default						
AUX02	BOOL	VAR				☐
AUX01	BOOL	VAR				☐
SEN_IN	BOOL	VAR_INPUT				☐
STOP_BZ	BOOL	VAR_INPUT				☐
CHK	BOOL	VAR_INPUT				☐
P_OUT_1s	BOOL	VAR_INPUT				☐
BZ	BOOL	VAR_OUTPUT				☐
LAMP	BOOL	VAR_OUTPUT				☐

When “ANA_P1” is regarded as an FB, “P_OUT_1s” is not a suitable name for the input terminal. Change it to “P_IN” in this example.

Variable	Data type	Usage	Comment	Address	Init	RETAIN
Default						
AUX02	BOOL	VAR				☐
AUX01	BOOL	VAR				☐
SEN_IN	BOOL	VAR_INPUT				☐
STOP_BZ	BOOL	VAR_INPUT				☐
CHK	BOOL	VAR_INPUT				☐
P_IN	BOOL	VAR_INPUT				☐
BZ	BOOL	VAR_OUTPUT				☐
LAMP	BOOL	VAR_OUTPUT				☐

<Code worksheet of the current project “LIB01”>



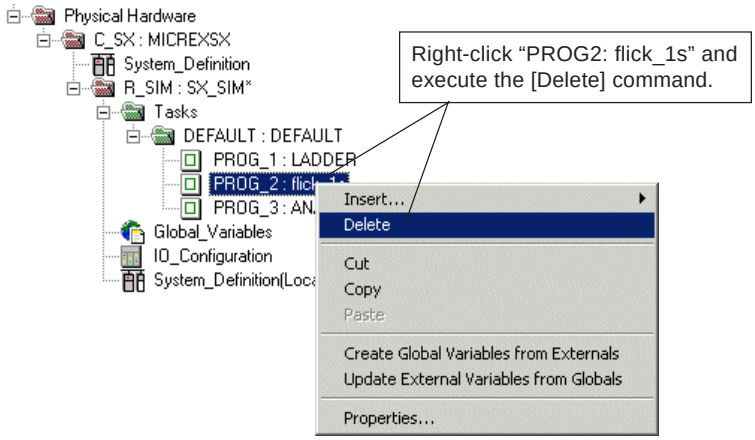
Section 3 Creating a User FB

3-3 Changing the Task

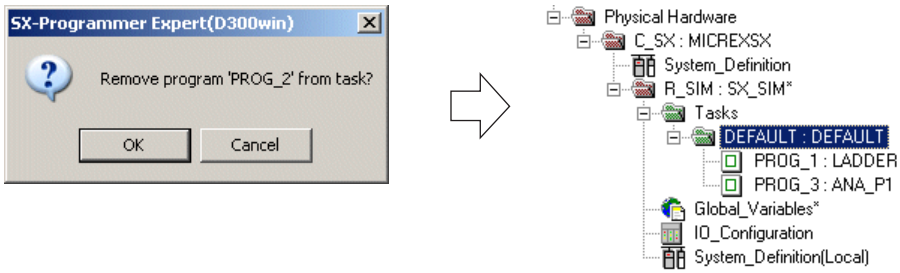
The POUs created as “programs” have already been made into FBs. Then, delete POU “flick_1s” and POU “ANA_P1” assigned to the task as programs.

<Operational procedure>

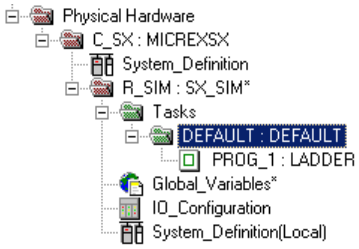
◆ Delete the POUs “flick_1s” and “ANA_P1” assigned to the task.
First, delete the program instance of POU “flick_1s”.



◆ The confirmation window shown below appears. Clicking the [OK] button deletes “PROG2: flick_1s” from the default task.



◆ In the same manner, delete “PROG_3: ANA_P1” as well.



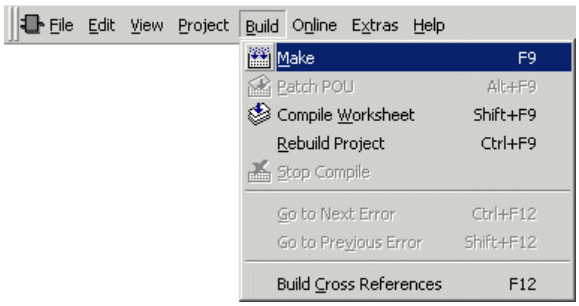
Section 3 Creating a User FB

3-4 Compiling and Checking an FB

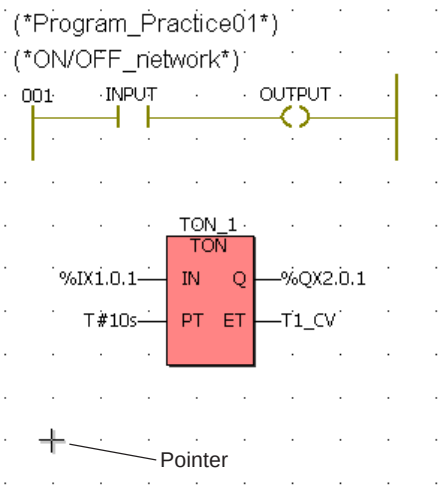
After you change the POUs to FBs and delete them from the task, compile the project. When compilation is completed, assign the created FBs to the POU “LADDER” and check the operation.

<Operational procedure>

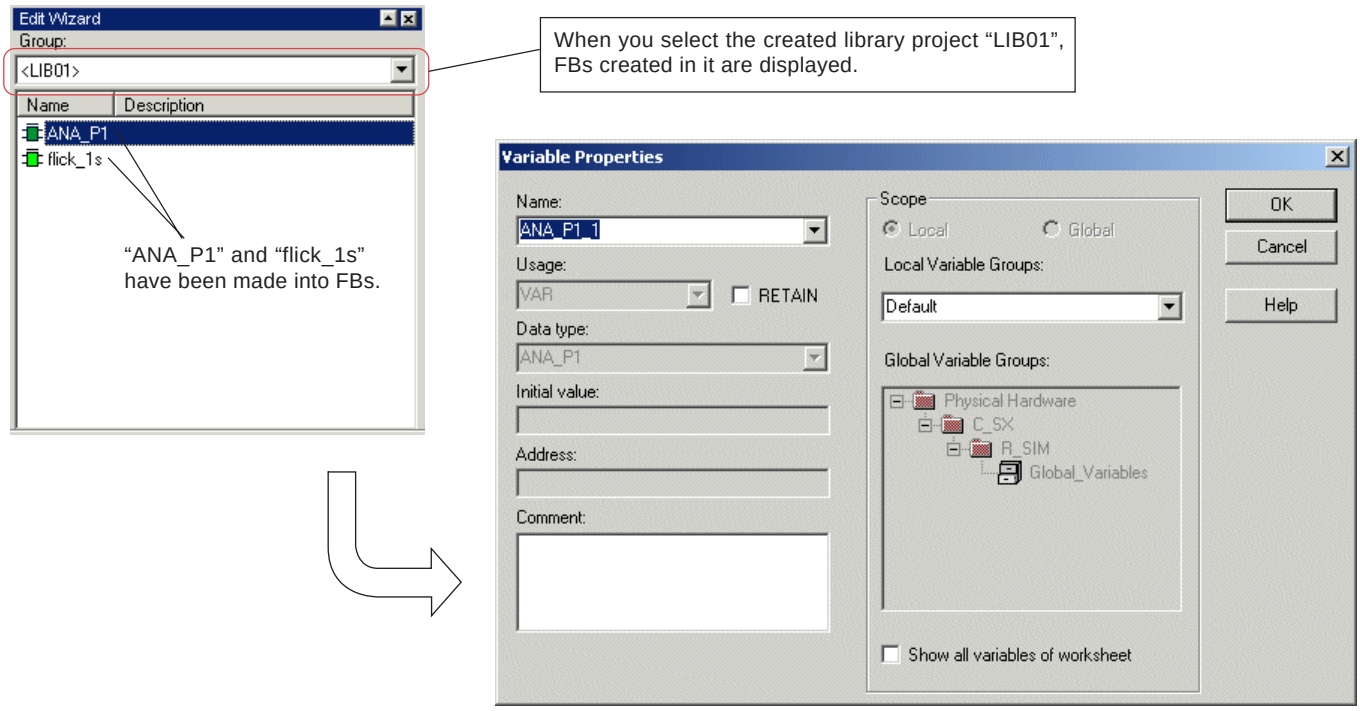
◆ Execute the [Make] command in the [Build] menu to compile the project.



◆ When compilation is completed, try to use the created FBs on POU “LADDER” of “program” type.
Open the code worksheet of POU “LADDER” and paste the FBs.
Place the pointer at an arbitrary point and execute the [Edit wizard] command in the [View] menu to display the [Edit wizard] window.



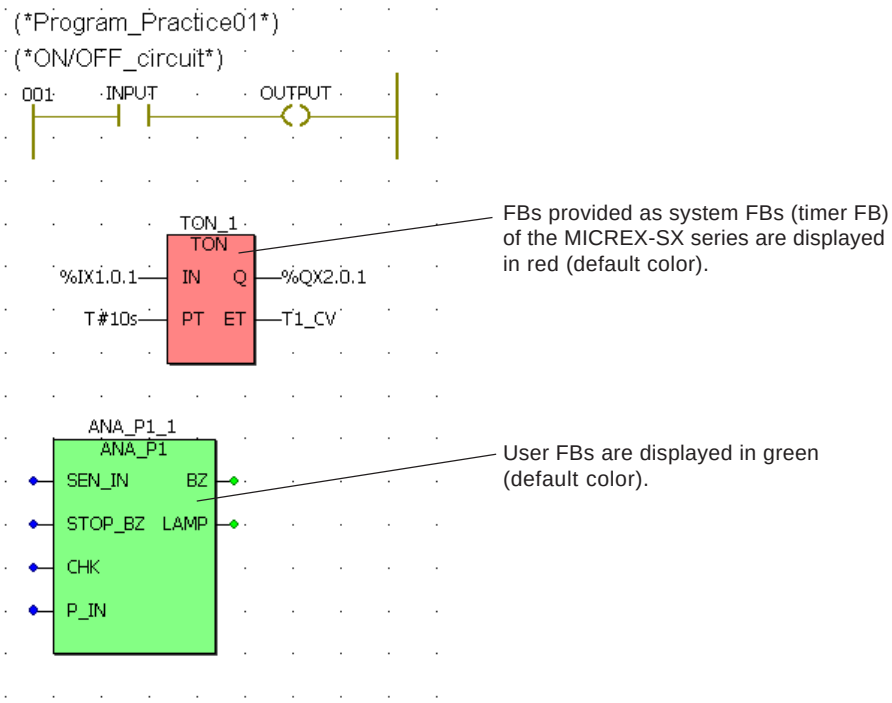
◆ Double-click the FB to be pasted on the [Edit wizard] window to display the [Variable Properties] dialog.



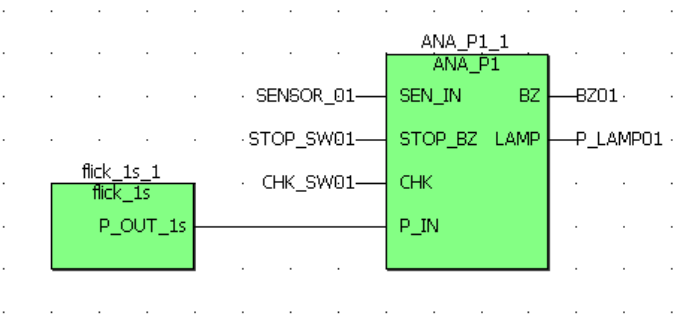
Section 3 Creating a User FB

3-4 Compiling and Checking an FB

◆ After selecting the FB to be used and click the [OK] button. The FB is pasted on the worksheet as shown below.



◆ In the same manner, paste the FB “flick_1s” on the code worksheet and create the following circuit.



* The circuit above is created by changing the flicker and annunciator circuits created in the Section 2 to FBs. Therefore, it operates in the same manner naturally. The circuit above shows you which variables are input or output at a glance. The MICREX-SX enables you to make frequently used circuits into FBs and reuse them with ease.

Section 3 Creating a User FB

3-4 Compiling and Checking an FB

◆ The local variable worksheet and global variable worksheet of POU “LADDER” are as shown below.

<Local variable worksheet>

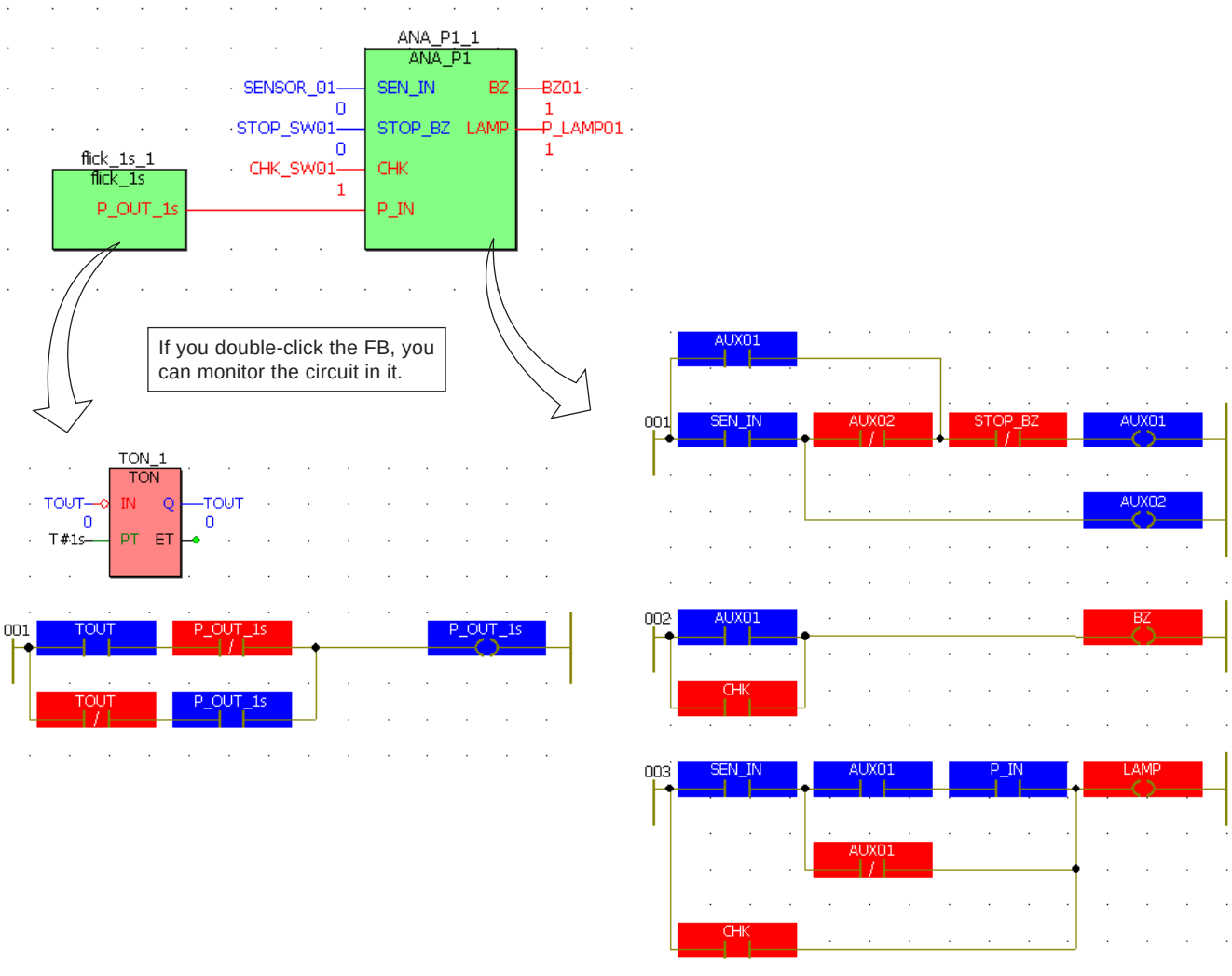
Variable	Data type	Usage	Comment	Address
Default				
TON_1	TON	VAR	TimerFB_instance	
INPUT	BOOL	VAR	ON/OFF_circuit	%IX1.0.0
OUTPUT	BOOL	VAR	ON/OFF_circuit	%QX2.0.0
T1_CV	TIME	VAR	Timer_current_value	
flick_1s_1	flick_1s	VAR	UserFB_FBinstance	
ANA_P1_1	ANA_P1	VAR	UserFB_FBinstance	
SENSOR_01	BOOL	VAR_EXTERNAL		
STOP_SW01	BOOL	VAR_EXTERNAL		
CHK_SW01	BOOL	VAR_EXTERNAL		
BZ01	BOOL	VAR_EXTERNAL		
P_LAMP01	BOOL	VAR_EXTERNAL		

<Global variable worksheet>

Variable	Data type	Usage	Comment	Address
Global_Variables				
SENSOR_01	BOOL	VAR_GLOBAL		
STOP_SW01	BOOL	VAR_GLOBAL		
CHK_SW01	BOOL	VAR_GLOBAL		
BZ01	BOOL	VAR_GLOBAL		
P_LAMP01	BOOL	VAR_GLOBAL		

The variables connected to the terminals of the FB are changed into global variables.

◆ Compile and download the project to the SX simulator to check the operation on it.



Fuji Electric FA Components & Systems Co., Ltd.

Mitsui Sumitomo Bank Ningyo-cho Bldg.,
5-7, Nihonbashi Odemma-cho, Chuo-ku, Tokyo 103-0011, Japan
Phone: +81-3-5847-8011
Fax: +81-3-5847-8172
URL http://www.fujielectric.co.jp/fcs/eng/