Function Block
Introduction Guide

The CD-ROM of CX-Programmer has User's Manual of the PDF file.
Please read the 'Notice' and the 'Precautions' in the User's Manual before using CX-Programmer.
The 'Function Block Introduction Guide' describes the basic operation procedure to use Function Block of CX-Programmer. Refer to the Help or the User's Manual of the PDF file for detailed descriptions.
* You need Acrobat Reader 4.0 or grater versions in your PC to display the PDF file.

Contents

Chapter 1
OMRON FB Library

1. What is a Function Block?

"Function Blocks" are predefined programs (or functions) contained within a single program element that may be used in the ladder diagram. A contact element is required to start the function, but inputs and outputs are editable through parameters used in the ladder arrangement.

The functions can be reused as the same element (same memory) or occur as a new element with its own memory assigned.

Partial Ladder program for machine A
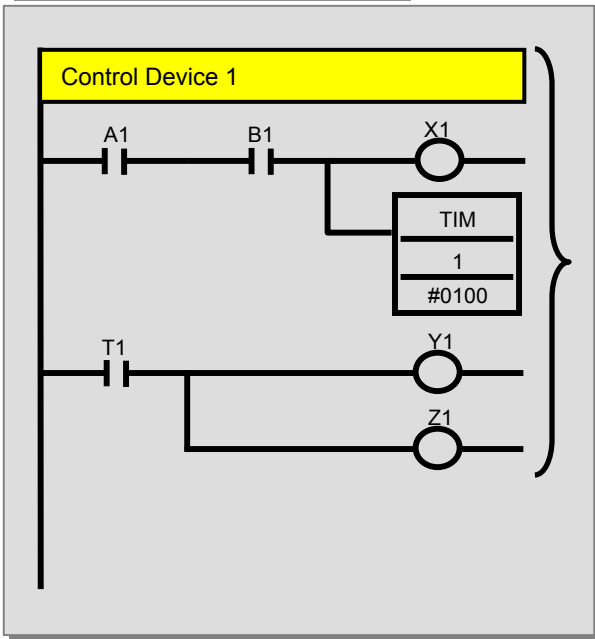
Defining Inputs and Outputs …

Input | Process (algorithm) | Output

Control Device 1

Control Device n

Produce template

Function Block definition

Allocate to Ladder program

Sets input / output parameters

Function Block Instance (invocation)

Device Control
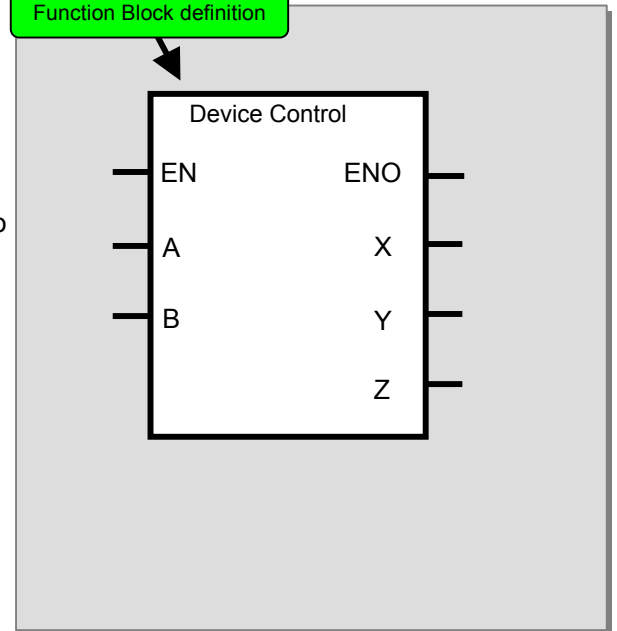
Function Block definition ...This contains the defined logic (algorithm) and I/O interface. The memory addresses are not allocated in the Function Block Definition
Function Block instance(invocation) ...This is the instruction that will call the function block instance when used by the ladder program, using the memory allocated to the instance.

2. An Example of a Function Block

The following figures describe an example of a function block for a time limit circuit, to be used in the ladder. It is possible to edit the set point of the TIM instruction to reallocate the set time for turning off the output in the ladder rung. Using the function block as shown below, it is possible to make the time limit of the circuit arbitrary by only changing one specific parameter.

Ladder diagram

By enabling the input parameter to be editable, it is possible to allow an arbitrary time limit circuit.

Timing chart

2.0 sec
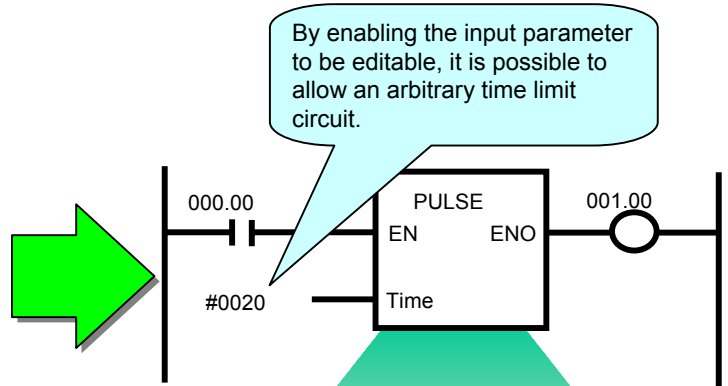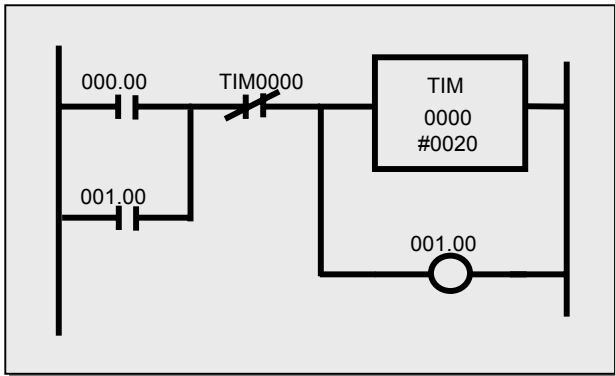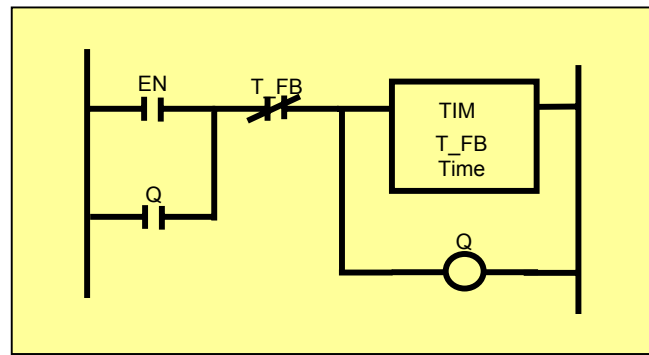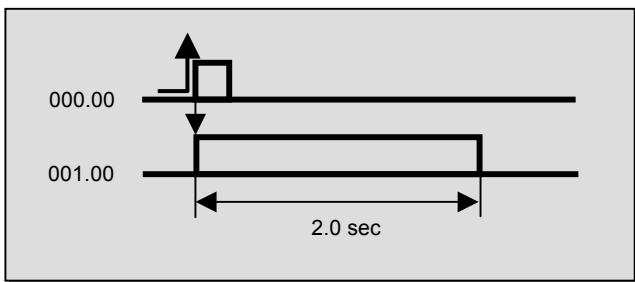
3. Overview of the OMRON FB Library

The OMRON FB Library is a collection of predefined Function Block files provided by Omron. These files are intended to be used as an aid to simplify programs, containing standard functionality for programming PLCs and Omron FA component functions.

3-1. Benefits of the OMRON FB Library

The OMRON FB Library is a collection of function block examples that aim to improve the connectivity of the units for PLCs and FA components made by Omron. Here is a list of the benefits to be gained from using the OMRON FB Library:

(1)No need to create ladder diagrams using basic functions of the PLC units and FA components
More time can be spent on bespoke programs for the external devices, rather than creating basic ladder diagrams, as these are already available.

(2)Easy to use
A functioning program is achieved by loading the function block file to perform the target functionality, then by inputting an instance (function block call instruction: invocation) to the ladder diagram program and setting addresses (parameters) for the inputs and outputs.

(3)Testing of program operation is unnecessary
Omron has tested the Function Block library. Debugging the programs for operating the unit and FA components for the PLCs is unnecessary for the user.

(4)Easy to understand
The function block has a clearly displayed name for its body and instances. A fixed name can be applied to the process.

The instance (function block call instruction: invocation) has input and output parameters. As the temporary relay and processing data is not displayed, the values of the inputs and outputs are more visible. Furthermore, as the modification of the parameters is localised, fine control during debugging etc. is easier.

Finally, as the internal processing of the function block is not displayed when the instance is used in the ladder diagram, the ladder diagram program looks simpler to the end user.

(5)Extendibility in the future
Omron will not change the interface between the ladder diagram and the function blocks. Units will operate by replacing the function block to the corresponding FB for the new unit in the event of PLC and the FA component upgrades, for higher performance or enhancements, in the future.

A fixed name can be named to the processes.

It is not necessary to create the basic communications program.

CurrentTemparatureOfXXHeater

_E5xx202_ReadPV

W100.00		(BOOL) EN	(BO
W100.01	&10	(INT) UnitSelect	(BOOL) BUSY	W101.01
	&1	(INT) PortNo	(BOOL) OK	W100.02
	&2	(INT) TCNo	(BOOL) NG	W100.03
	&2		(DINT) PV	D100

INPUT/OUTPUT data is clear.
Parameters are easy to understand and edit.
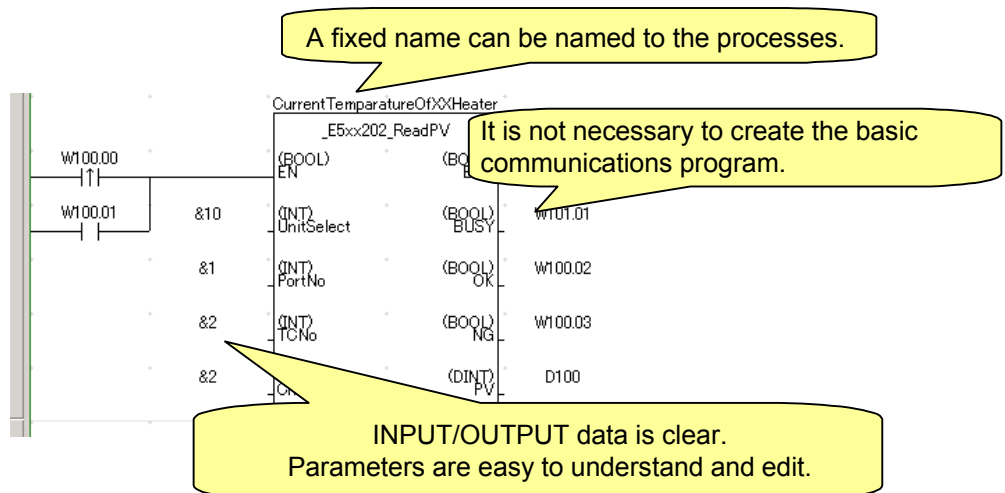
3-2-1. Example of using the OMRON FB Library - 1

Controlling the predefined components made by Omron can be easily achieved from the PLC ladder diagram.

- Ability to configure low-cost communications (RS-232C/485)
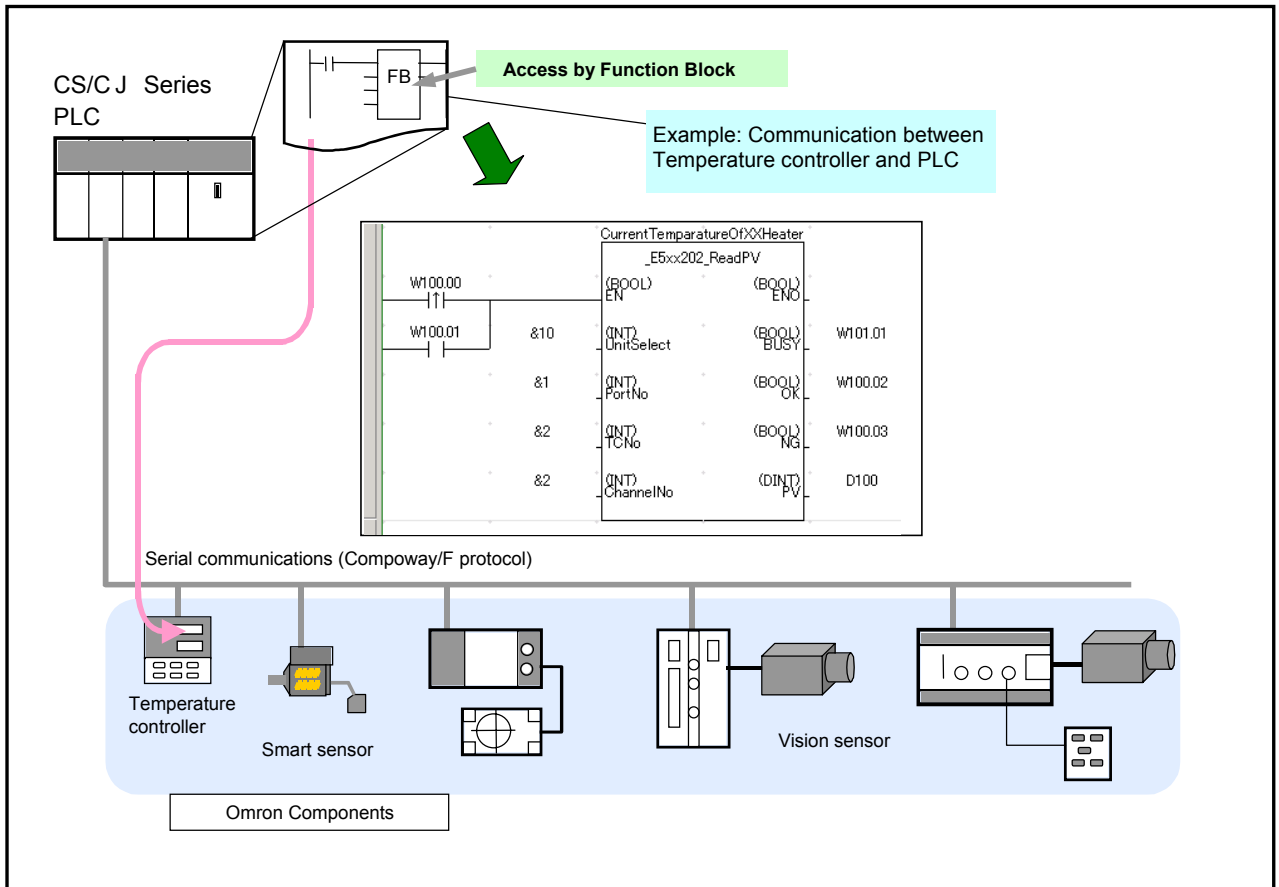
3-2-2. Example of using the OMRON FB Library - 2

High performance communications can be made by DeviceNet level.

- Ability to communicate between PLC and DeviceNet slaves easily.
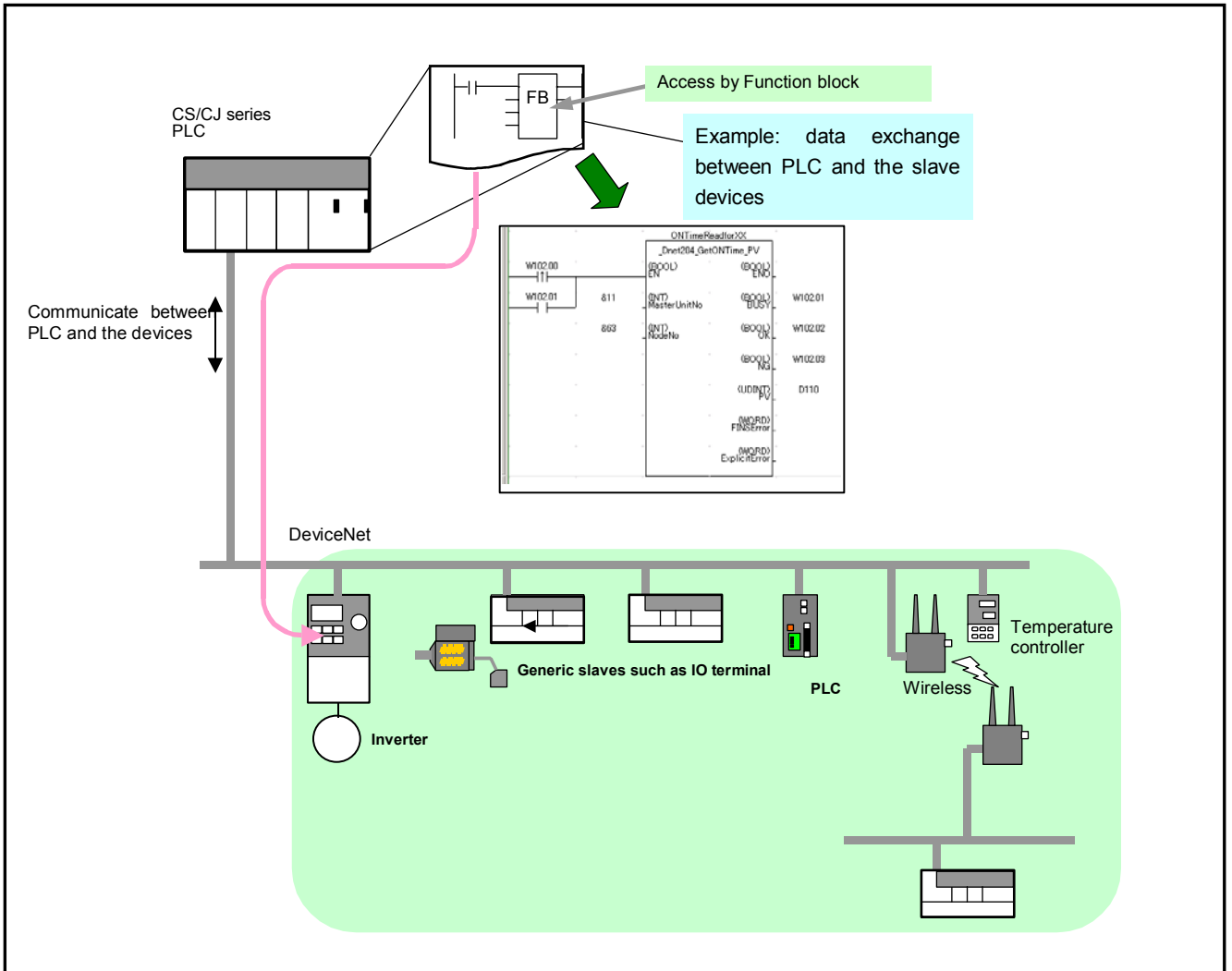
3-3. Content of the OMRON FB Library

The OMRON FB Library consist of the following:

3-3-1. OMRON FB Part Files

The OMRON FB Part file is prepared using the ladder diagram function block, for defining each function of the PLC unit and the FA component.
The files contain a program written in ladder diagram and have the extension .CXF.
The file name of the OMRON FB Part file begins with '_' (under score).
When the OMRON FB Library is installed onto a personal computer, the OMRON FB Part files are classified in the folder appropriate to each PLC Unit and FA component in the Omron Installation directory.
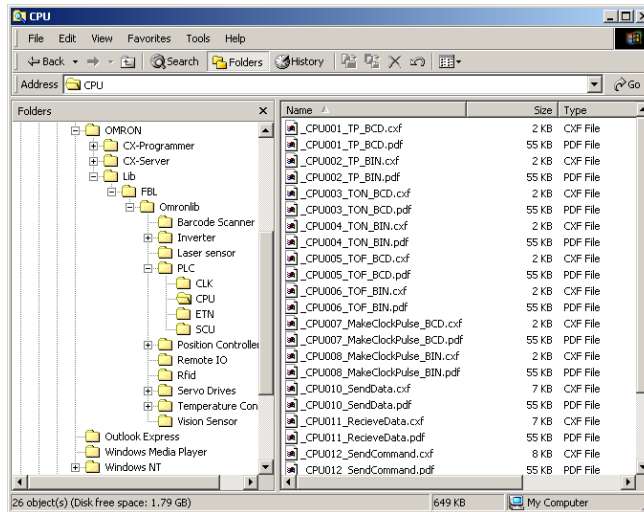
3-3-2. Library reference

The library reference describes the operation specifications of the OMRON FB Part file, and the specifications of the input and the output parameters for each. The file format for this is PDF.
When the OMRON FB Library is used, the user should select the OMRON FB Part file, set the input / output parameters, and test the program operations referring to the library reference.

V60x 200	Read Data Carrier Data _V60x200_ReadData	
FB name	_V600_ReadData	
Symbol		
File name	¥Lib¥FBL¥English¥omronlib¥RFID¥V600¥_V60x200_ReadData10.cxf	
Applicable models	CS1W-V600C11/V600C12 and CJ1W-V600C11/V600C12 ID Sensor Units	
Basic function	Reads data from a Data Carrier.	
Conditions for usage	Other • This FB cannot be executed if the ID Sensor Unit is busy. The NG Flag will turn ON if an attempt is made.	
Function description	Data is read from the specified area of the Data Carrier specified by the *Unit No.* and *Vendor No.* Up to 2048 bytes (1024 words) can be read at one time. The word designation for storing the data is specified using the area type and beginning word address. For example, for D1000, the area type is set to P_DM and the beginning word address is set to &1000.	
EN input condition	Connect EN to an OR between an upwardly differentiated condition for the start trigger and the BUSY output from the FB.	
Restrictions Input variables	• Always use an upwardly differentiated condition for EN. • If the input variables are out of range, the ENO Flag will turn OFF and the FB will not be processed. • Always specify a head number of &1 for One-Head ID Sensor Units (CS1W-V600C11 and CJ1W-V600C11).	

3-4. File Catalog and Where to Access the OMRON FB Library

3-4-1. Catalog of OMRON FB Library files

Type	Target components	Number of OMRON FB Part files (at the time of July '04)
FA components	Temperature controller, Smart sensor, ID sensor, Vision sensor, 2 dimensions bar code reader	approx. 80
PLC	CPU unit, Memory card, Special CPU IO unit (Ethernet, Controller Link, DeviceNet unit, Temperature control unit)	approx. 95
Motion control components	Position control unit Inverter Servo motor driver	approx. 30

3-4-2. CX-Programmer installation CD

CX-Programmer Ver.5.0 installation CD contains the OMRON FB Library files.
The user can select to install the OMRON FB Library during the installation of CX-Programmer Ver.5.0.

CX-Programmer Ver.5.0
Install CD
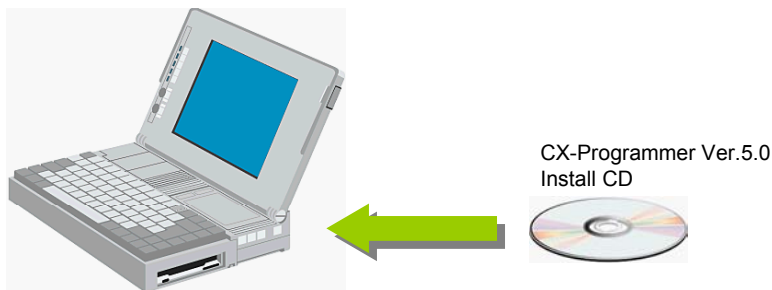
3-4-3. Accessing OMRON FB Library files from Web server

The latest version OMRON FB Library files will be provided by Omron on the Web server.
New files will be added to support new or enhanced PLC units and FA components.
The download service of the OMRON FB Library is provided as a menu of Omron Web in each country.

Web server

The internet

The internet
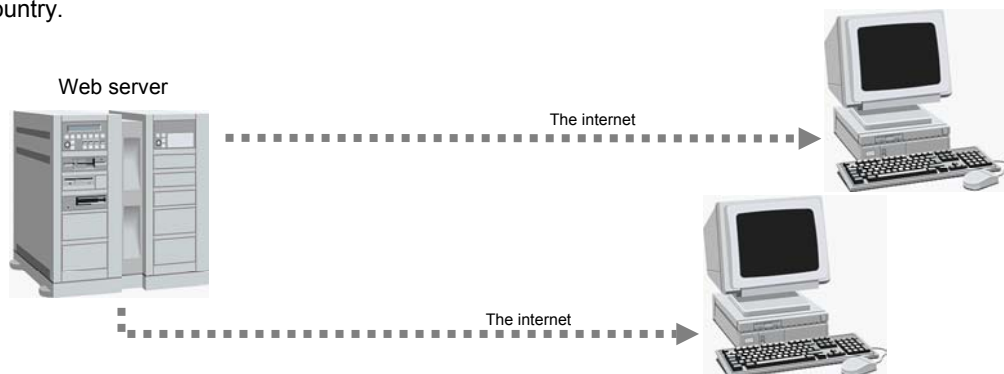
Chapter 2

How to use
the OMRON FB Library

Explanation of target Program	Opening a new project	Import FB Library	Creating a program	Program Check

1. Explanation of the target program

This chapter describes how to use OMRON FB Library using the OMRON FB Part file 'Make ON Time/OFF Time Clock Pulse in BCD'.

1-1. Application Specifications

The target application specifications are as follows :-
- Pulse is generated after PLC mode is changed to 'run' or 'monitor' mode.
- Output the pulse to address 1.00.
- On time of generated pulse is set at D100.
- Off time of generated pulse is 2 seconds.

1-2. Specifications of the OMRON FB Part file

The OMRON FB Part file 'Make ON Time/OFF Time Clock Pulse in BCD' has the following specifications:-

CPU 007	Make ON Time/OFF Time Clock Pulse in BCD _CPU007_MakeClockPulse_BCD				
Basic function	Generates a clock pulse with the specified ON time and OFF time and outputs it to ENO.				
Symbol	(symbol diagram)				
File name	¥Lib¥FBL¥English¥omronlib¥PLC¥CPU¥_CPU007_MakeClockPulse_BCD10.cxf				
Applicable models	CS1-H, CS1-H, and CJ1M CPU Units				
Conditions for usage	PLC Properties • The PV update method for timers and counters must be set to BCD in the PLC Setup. A compiling error will occur if BCD mode is not set. The mode can be set in the PLC Properties in the CX-Programmer. (PLC Properties dialog image) Shared Resources • Timers				
Function description	ENO will be OFF for the time set in *OFF time* and then will be ON for the time set in *ON time*. (timing diagram)				
EN input condition	Connect the EN input to the Always ON Flag (P_On).				
Restrictions Input variables	• If the input variables are out of range, the ENO Flag will turn OFF and the FB will not be processed. • Set the *ON time* and *OFF time* input variables to between #0000 and #9999 in BCD (100 ms units). If a setting is not within range, ENO is turned OFF.				
Application example	In the following example, bit A will be repeatedly ON for 5 s and OFF for 3 s. (example diagram)				
Related FBs	Use the correct FB for the timer/counter PV update mode set in the PLC Setup. Binary mode: Make ON Time/OFF Time Clock Pulse in Binary (_CPU008_MakeClockPulse_BIN) BCD mode: Make ON Time/OFF Time Clock Pulse in BCD (_CPU007_MakeClockPulse_BCD)				

■ Variable Tables
Input Variables

Name	Variable name	Data type	Default	Range	Description
EN	EN	BOOL			1 (ON): FB started 0 (OFF): FB not started.
ON time	OnTime	WORD		#0000 to #9999	Specify the ON time (unit: 100 ms). For example, #30 means 3 seconds.
OFF time	OffTime	WORD		#0000 to #9999	Specify the OFF time (unit: 100 ms). For example, #30 means 3 seconds.

Output Variables

Name	Variable name	Data type	Range	Description
ENO	ENO	BOOL		Turns ON for the OnTime and OFF for the OffTime.

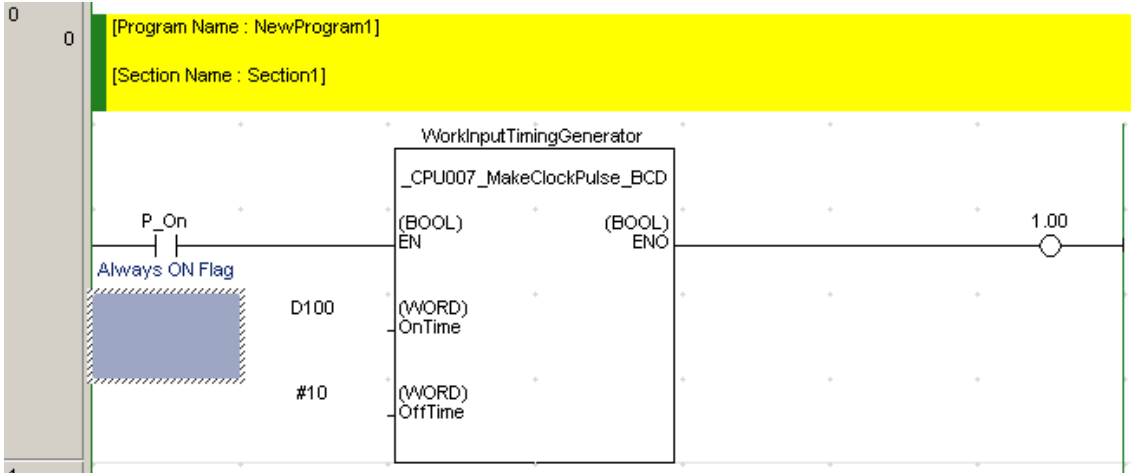| Explanation of target Program		Opening a new project		Import FB Library		Creating a program		Program Check

1-3. Input program

Create the following ladder program:-

0	0	[Program Name : NewProgram1]
		[Section Name : Section1]

```
                              WorkInputTimingGenerator
                              _CPU007_MakeClockPulse_BCD

        P_On                  (BOOL)              (BOOL)              1.00
        -| |-                 EN                    ENO               -O-
     Always ON Flag

                       D100   (WORD)
                              OnTime

                       #10    (WORD)
                              OffTime
```

[Reference] If created as a straightforward ladder diagram, the program would be as below:-
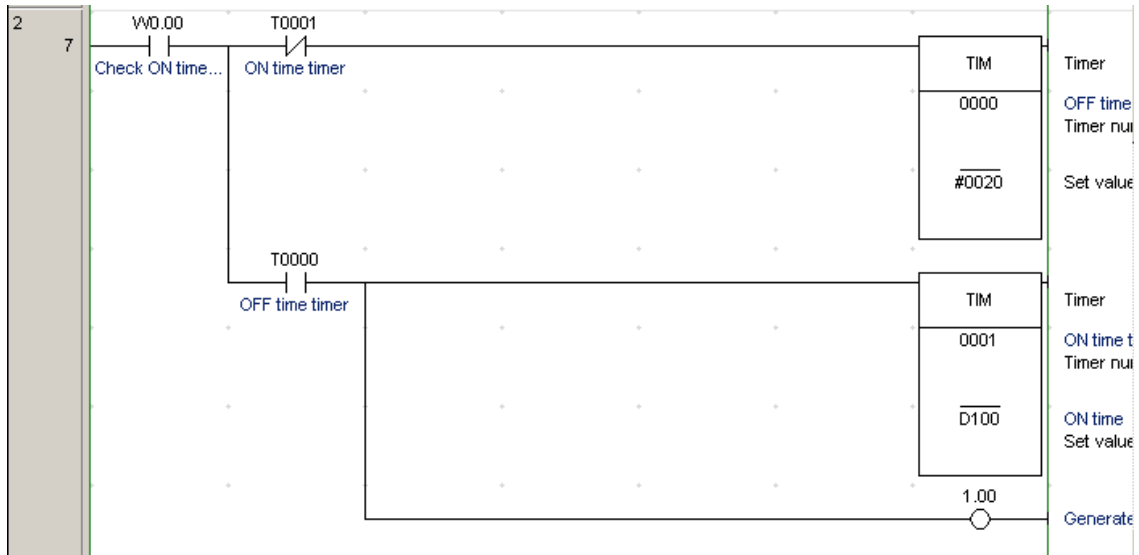
```
2    W0.00        T0001
7    -| |-        -|/|-                                        TIM       Timer
     Check ON time...  ON time timer
                                                              0000      OFF time
                                                                        Timer nu

                                                              #0020     Set value

             T0000
             -| |-                                            TIM       Timer
             OFF time timer
                                                              0001      ON time t
                                                                        Timer nu

                                                              D100      ON time
                                                                        Set value

                                                              1.00
                                                              -O-       Generate
```

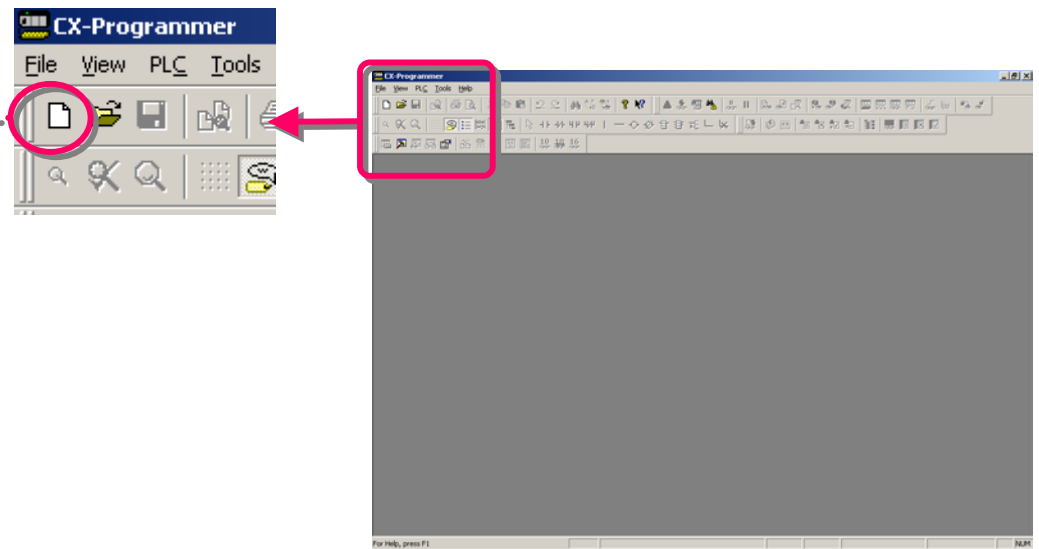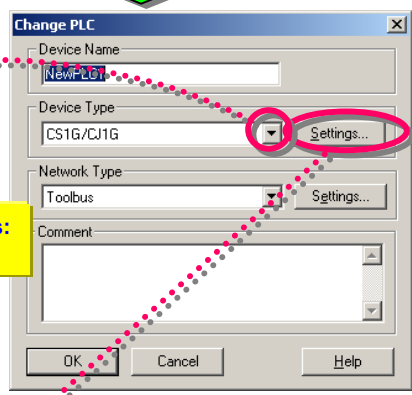| Explanation of target Program		Opening a new project		Import FB Library		Creating a program		Program Check

2. Opening a new project and setting the Device Type
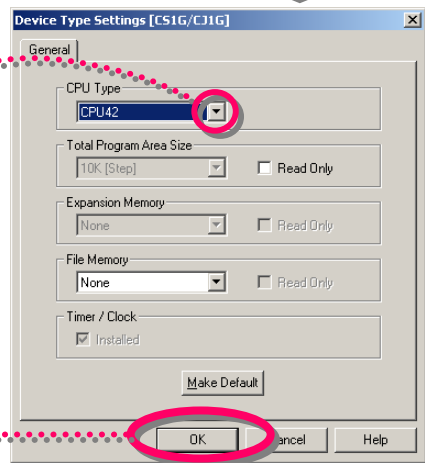
Click the toolbar button [New] in CX-Programmer.

Click ▯

Click the left mouse button.
▼

!To use Function Blocks, select the following PLCs:
CS1G-H, CS1H-H, CJ1G-H, CJ1H-H, CJ1M

Change PLC

Device Name
NewPLC1

Device Type
CS1G/CJ1G   Settings...

Network Type
Toolbus   Settings...

Comment

OK   Cancel   Help

Settings...
Click the left mouse button
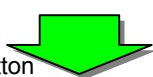
Click the left mouse button to select CPU type.
▼

Device Type Settings [CS1G/CJ1G]

General

CPU Type
CPU42 ▼

Total Program Area Size
10K [Step]   ☐ Read Only

Expansion Memory
None   ☐ Read Only

File Memory
None   ☐ Read Only

Timer / Clock
☑ Installed

Make Default

OK   Cancel   Help

OK

Click [OK] to decide the selected CPU type.

3. Main Window functions

The main window functionality is explained here.
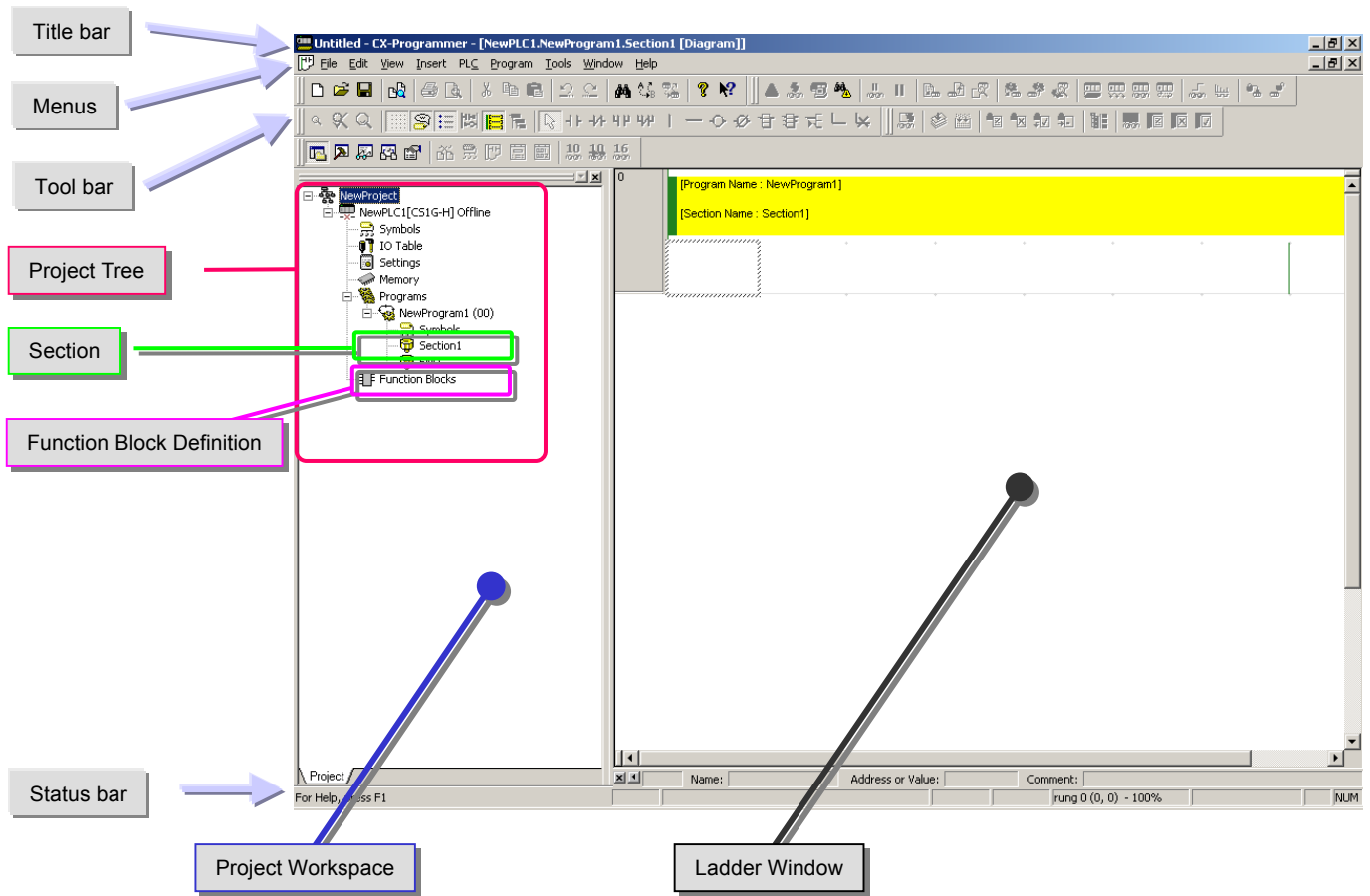

Name	Contents / Function
Title Bar	Shows the file name of saved data created in CX-Programmer.
Menus	Enables you to select menu items.
Toolbars	Enables you to select functions by clicking icons. Select [View] -> [Toolbars], display toolbars. Dragging toolbars enables you to change the display positions.
Section	Enables you to divide a program into several blocks. Each can be created and displayed separately.
Project Workspace Project Tree	Controls programs and data. Enables you to copy element data by executing Drag and Drop between different projects or from within a project.
Ladder Window	A screen for creating and editing a ladder program.
Function Block Definition	Shows Function Block definition. By selecting the icons, you can copy or delete the selected Function Block definition. - **L** is shown if the file is a OMRON FB Part file. - In the case of a User-defined Function Block **L** is shown if Ladder, **S** is shown if ST.
Status Bar	Shows information such as a PLC name, online/offline state, location of the active cell.

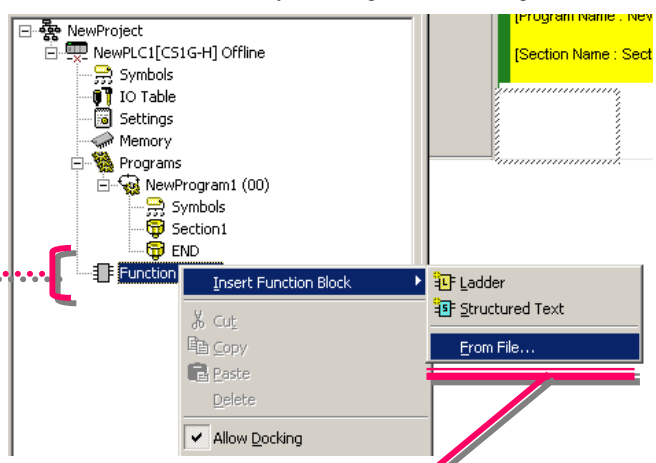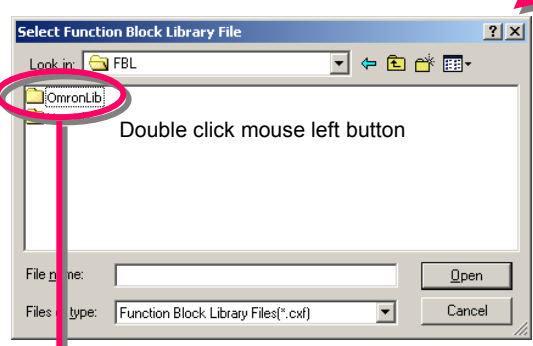| Explanation of target Program	Opening a new project	Import FB Library	Creating a program	Program Check

4. Import the OMRON FB Part file

Select Function Block definition icon from the project tree using the mouse cursor, right click.
Select Insert Function Block, then select a Library file using mouse to navigate.

Click mouse right button
-> Insert Function Block
-> Library File

Double click mouse left button.
-> [OmronLib]
-> [Programmable Controller]
-> [CPU]
Select each of the above in series.

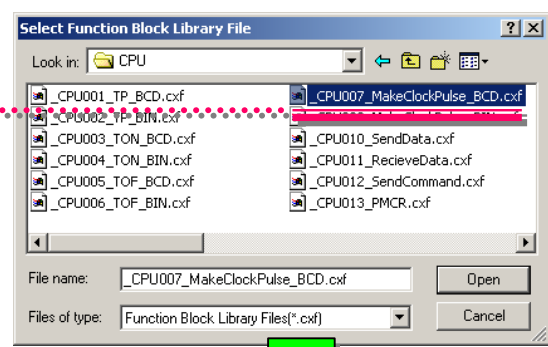Double click mouse left button

Select the necessary OMRON FB Part file in the 'Select Function Block Library' dialog.
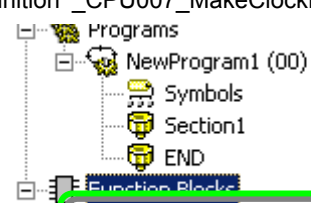
! The default path of the OMRON FB Library is C:/Program Files/Omron/Lib/FBL.

Left Click
'_CPU007_MakeClockPulse_BCD.cxf'

Left Click the [Open] button

Function Block definition '_CPU007_MakeClockPulse_BCD' is registered as part of the project file.

Function Block Definition

5. Program Creation

Confirm cursor position is at the upper left of Ladder Window to start programming.
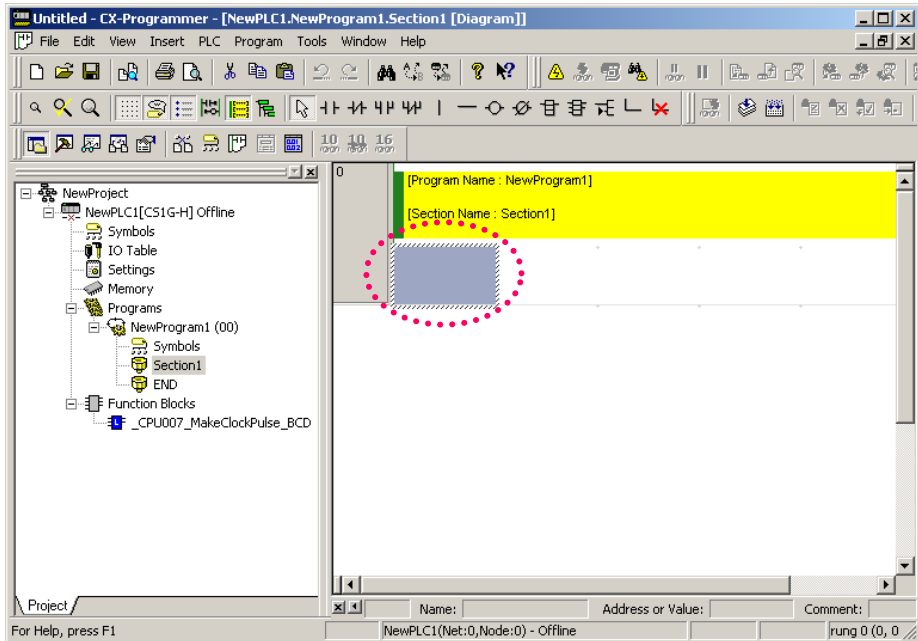
5-1. Enter a Normally Open Contact

C .... Press the [C] key on the keyboard to open the [New Contact] dialog. Use the dropdownbox to select the "P_On" symbol.

P_On ....

ENT
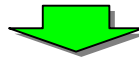
Deleting instructions
- Move the cursor to the instruction and then press the DEL key *or*
- Move the cursor to the right cell of the instruction and press the BS key.

"P_On" is a system defined symbol. Its state is always ON.
0 of the upper digit of an address is omitted when shown.
[.] (period) is displayed between a channel number and a relay number.

5-2. Entering an Instance

F ·············· Press the [F] key on the keyboard to open the [New Function Block Invocation] dialog.

Enter text to create an FB instance name.
[WorkInputTimingGenerator]

New Function Block Invocation

FB Instance: [▼]   OK
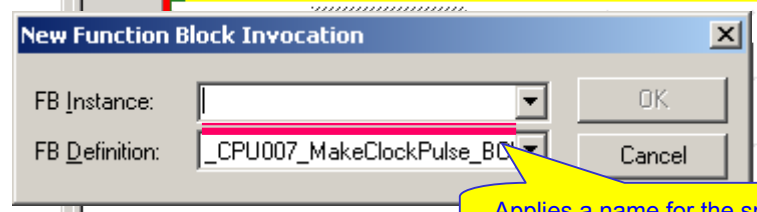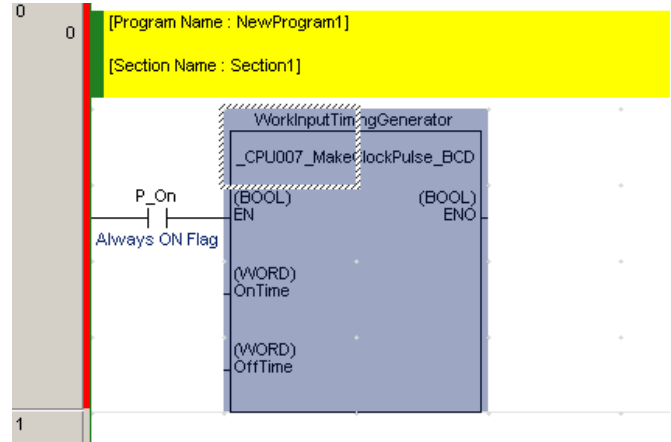
FB Definition: [_CPU007_MakeClockPulse_BC▼]   Cancel

> Applies a name for the specific process in the diagram.

ENT ·············· Shows FB instance (invocation) 'WorkInputTimingGenerator'.

```
0
   0   [Program Name : NewProgram1]

       [Section Name : Section1]

                        WorkInputTimingGenerator
                        _CPU007_MakeClockPulse_BCD
            P_On    (BOOL)              (BOOL)
            ┤ ├     EN                    ENO
       Always ON Flag

                    (WORD)
                    OnTime

                    (WORD)
                    OffTime
1
```

5-3. Entering Parameters

```
            P_On    (BOOL)
            ┤ ├     EN
       Always ON Flag

                    (WORD)
                    OnTime

                    (WORD)
                    OffTime

       OnTi
```

P or ENT
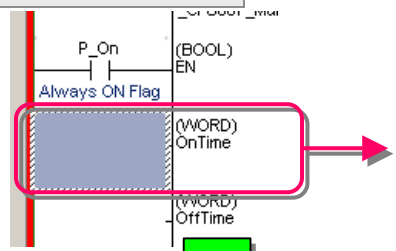
Move the cursor to the left of input parameter.

New Parameter

[d100   ▼] Detail >> OK Cancel

Enter the address.

[d100]

```
            P_On    (BOOL)
            ┤ ├     EN
       Always ON Flag

            D100    (WORD)
            OnTime  OnTime
```
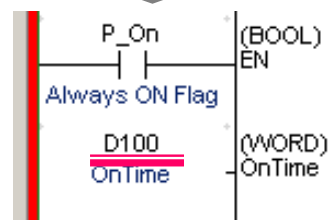
Choose an address for the input parameter 'OnTime'.

ENT

Enter the remaining parameters in the same way.

WorkInputTimingGenerator

_CPU007_MakeClockPulse_BCD

P_On	(BOOL) EN	(BOOL) ENO
Always ON Flag		
D100 OnTime	(WORD) OnTime	
	(WORD) OffTime	

New Parameter

#10 ▼ | Detail >> | OK | Cancel

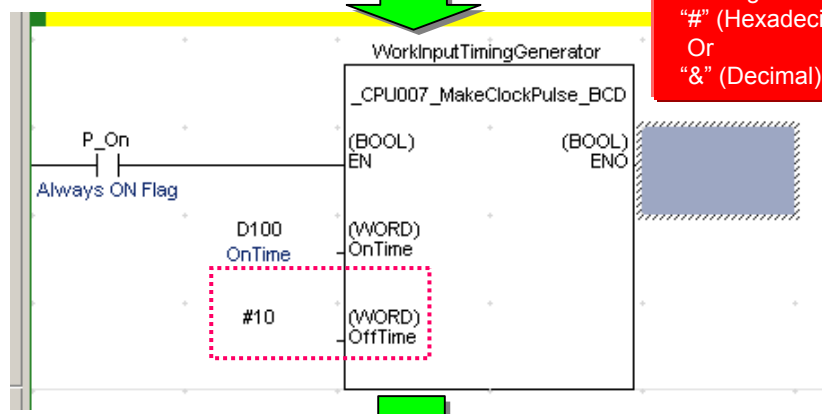P | Or | ENT

#10

ENT

O

Please add the following prefix for entering constants as parameters:
"#" (Hexadecimal/BCD)
Or
"&" (Decimal)

WorkInputTimingGenerator

_CPU007_MakeClockPulse_BCD

P_On	(BOOL) EN	(BOOL) ENO
Always ON Flag		
D100 OnTime	(WORD) OnTime	
#10	(WORD) OffTime	

-( )- New Coil

1.00 ▼ | Detail >> | OK | Cancel

1.00

ENT

[Generated Pulse]

ENT

-( )- New Edit Comment (1/1) : 1.00

1.00 | Generated pulse | OK | Cancel
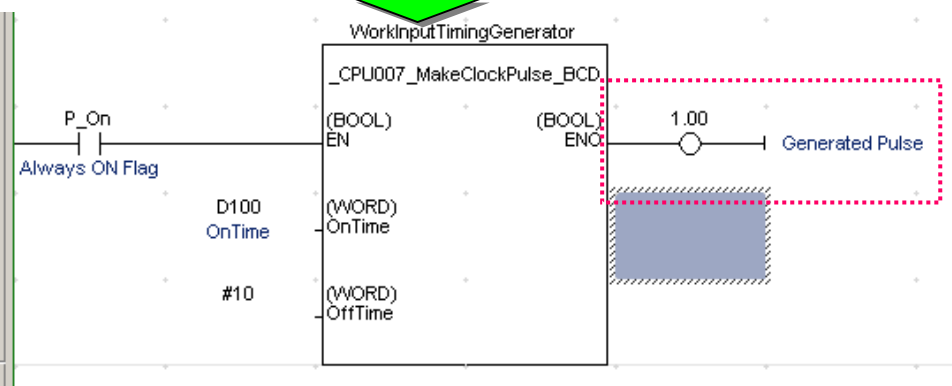
WorkInputTimingGenerator

_CPU007_MakeClockPulse_BCD

P_On	(BOOL) EN	(BOOL) ENO	1.00	Generated Pulse
Always ON Flag				
D100 OnTime	(WORD) OnTime			
#10	(WORD) OffTime			

Explanation of target Program		Opening a new project		Import FB Library		Creating a program		Program Check

6. Program Error Check (Compile)

Before program transfer, check for errors using the program compile.

Click
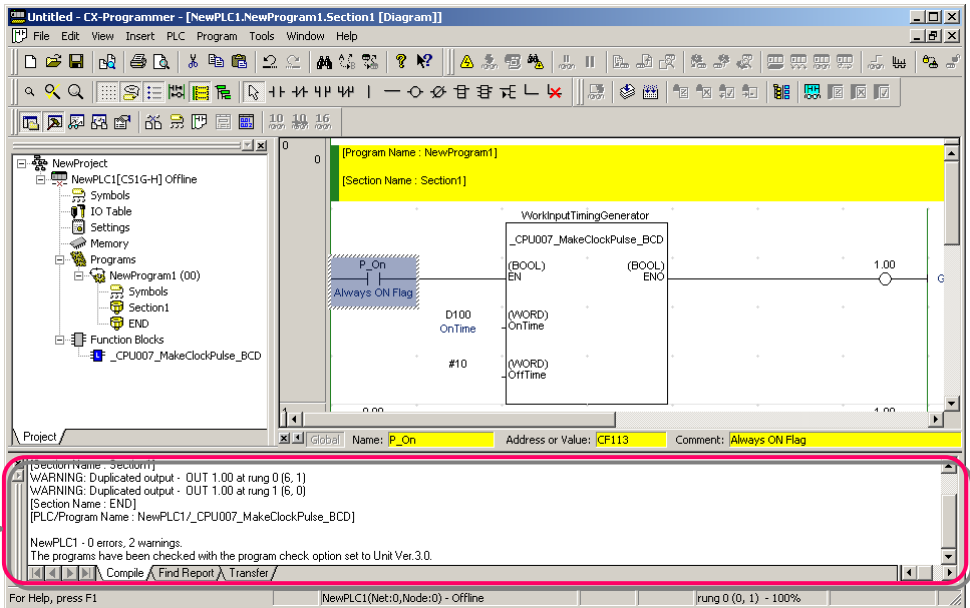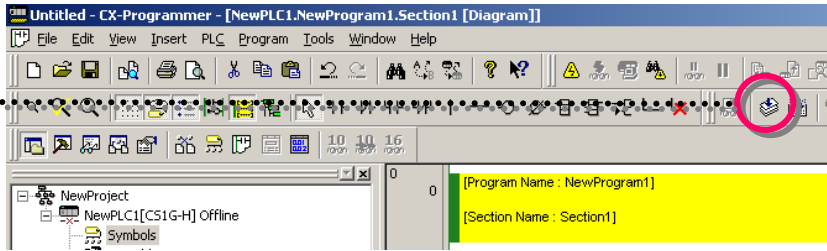
Errors and addresses are displayed in the Output Window.
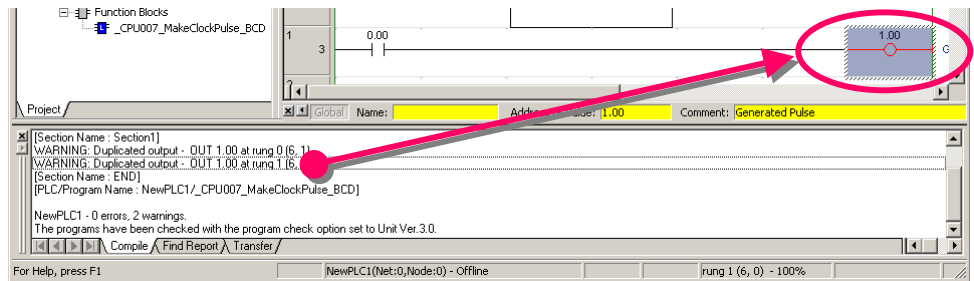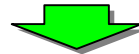
Double-click on displayed errors, and the Ladder Diagram cursor will move to the corresponding error location, displaying the error rung in red.

Modify the error.

- Output Window automatically opens at program check.
- The cursor moves to an error location by pressing J or F4 key.
- Output Window closes by pressing the ESC key.

Online to transfer	Monitoring	Online Edit

7. Going Online

CX-Programmer provides three methods of connecting, depending on usage.

⚠ Normal online. Enables you to go online with a PLC of the device type and method specified when opening a project.

🔍 Auto online. Automatically recognizes the connected PLC and enables you to go online with a PLC with one button. -> Uploads all data, such as programs, from the PLC.
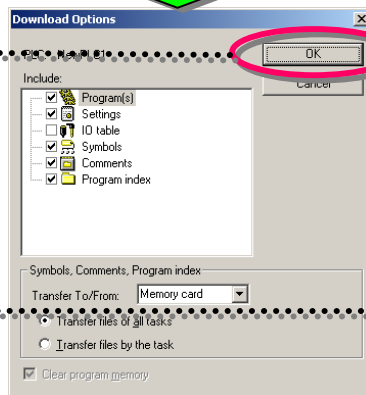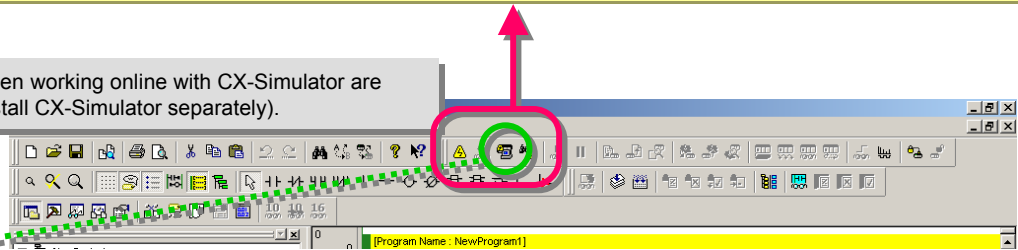
📇 Online with Simulator. Enables you to go online with CX-Simulator with one button (CX-Simulator must be installed.)

Online/debug functions when working online with CX-Simulator are explained in this guide (Install CX-Simulator separately).

Click 📇

Click [OK]

Download Options

Include:
- ☑ Program(s)
- ☑ Settings
- ☐ IO table
- ☑ Symbols
- ☑ Comments
- ☑ Program index

Symbols, Comments, Program index

Transfer To/From: Memory card

Program transfer starts.

Download

Program Download to PLC NewPLC1

Download successful

Click [OK]

- ◉ Transfer files of all tasks
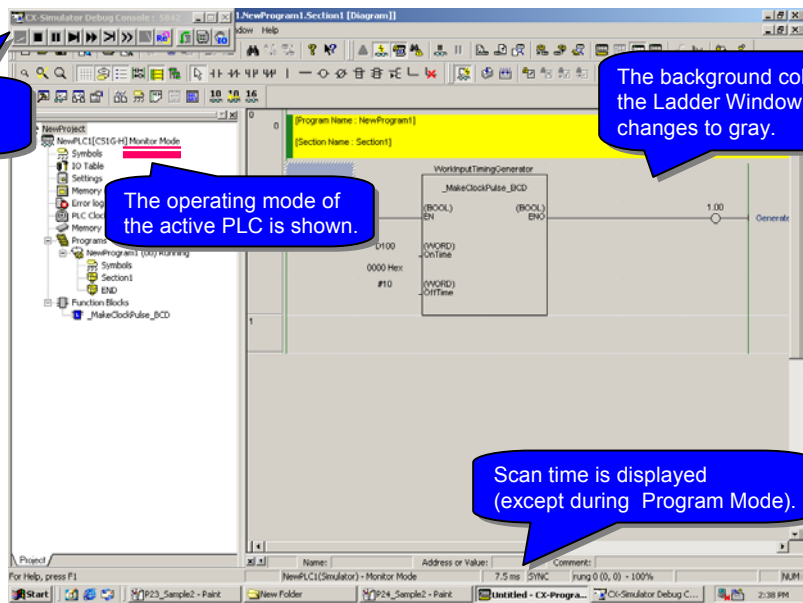- ○ Transfer files by the task
- ☑ Clear program memory
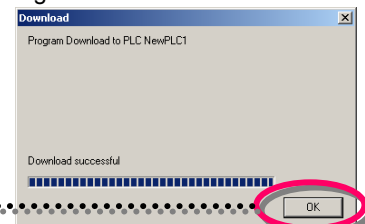
The CX-Simulator Console box is shown.

The operating mode of the active PLC is shown.

The background color of the Ladder Window changes to gray.

Scan time is displayed (except during Program Mode).
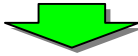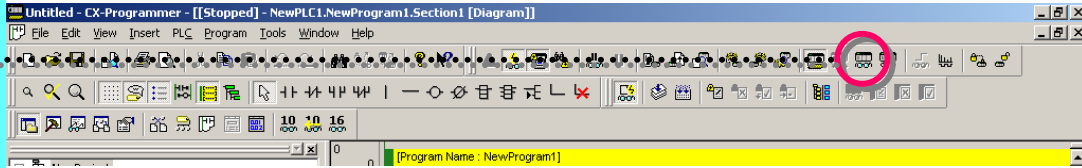
Online
to transfer

Monitoring

Online
Edit

8. Monitoring - 1
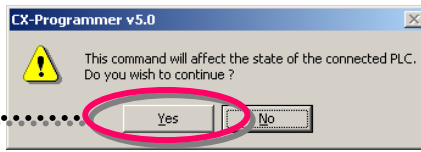
Change the PLC (Simulator) to Monitor mode.

The on/off status of contacts and coils can be monitored.

Click

If your program has a large volume of data, the scroll speed of the screen may become slow when monitoring.
To resolve this, click the icon below to cancel monitoring, scroll to the address you want to monitor, then restart the monitor mode.

CX-Programmer v5.0

⚠ This command will affect the state of the connected PLC.
Do you wish to continue ?

Yes   No

Click [Yes].
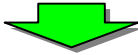
··· toggles PLC monitoring on/off

Untitled - CX-Programmer - [[Stopped] - NewPLC1.NewProgram1.Section1 [Diagram]]

File Edit View Insert PLC Program Tools Window Help

NewProject
NewPLC1[CS1G-H] Stop/Program Mode
Symbols
IO Table
Settings
Memory card
Error log
PLC Clock
Memory
Programs
NewProgram1 (00) Stopped
Symbols
Section1
END
Function Blocks
_CPU007_MakeClockPulse

[Program Name : NewProgram1]

[Section Name : Section1]

The monitored area is displayed in a specified color.

WorkInputTimingGenerato

_CPU007_MakeClockPulse_BCD

P_On
Always ON Flag

(BOOL)
EN

(BOOL)
ENO

1.00

D100
OnTime
0000 Hex

(WORD)
OnTime

#10

(WORD)
OffTime

The current values of parameters are shown.

Project

For Help, press F1

NewPLC1(Simulator) - Stop/Program Mode

SYNC   rung 1 (0, 0) - 100%

9. Monitoring - 2 Change Parameter Current Value

Change the current value of contact/coils or word data in the Ladder Window.
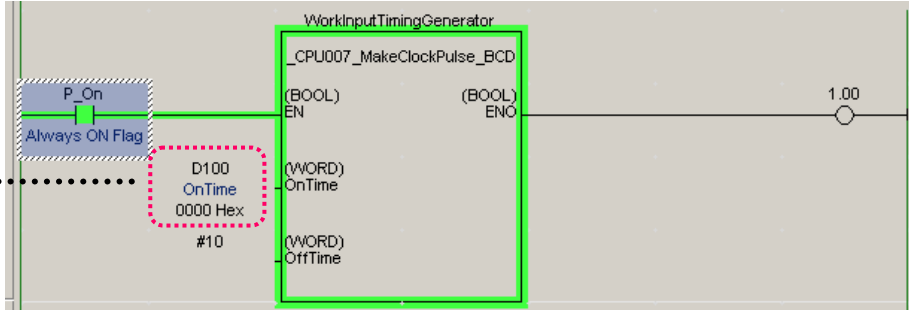
Move the cursor to the input parameter 'D100'.

Click mouse right button and select the menu item
[Set/Reset(S)]
->[Setting Value(V)]

Or

Double click mouse left button.

Or

ENT
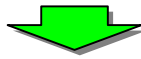
Change the current value of Input parameter.

Click [Set]

Please add the following prefix for entering constants as parameters:
 "#" (Hexadecimal/BCD)
 Or
 "&" (Decimal)

Set New Value

Address: D100   Set

Data type: UINT   Cancel

Value: #0020

#0 to #FFFF (1CH)

Online to transfer	Monitoring	Online Edit

10. Online Editing

Move the cursor to the rung requiring modification.

You can also select multiple rungs by using the Drag & Drop facility with the mouse.

[Program Name : NewProgram1]

[Section Name : Section1]

WorkInputTimingGenerator
_CPU007_MakeClockPulse_BCD
P_On (BOOL) (BOOL) 1.00
EN ENO
Always ON Flag
D100 (WORD)
OnTime OnTime
0020 Hex
#10 (WORD)
OffTime

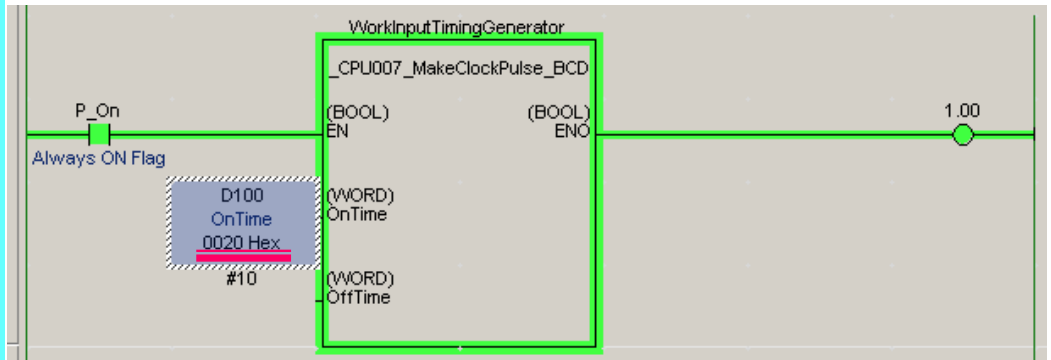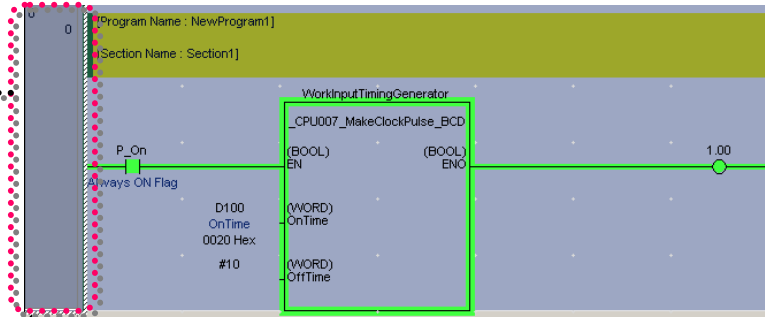Select [Program] ->[Online Edit] -> [Begin]

Shortcut: [Ctrl]+[E]

Move the cursor to a instruction you want to modify. Double click the left mouse button.

[Program Name : NewProgram1]

[Section Name : Section1]

WorkInputTimingGenerator
_CPU007_MakeClockPulse_BCD
P_On (BOOL) (BOOL) 1.00
EN ENO
Always ON Flag
D100 (WORD)
OnTime OnTime
0020 Hex
#10 (WORD)
OffTime

Double click
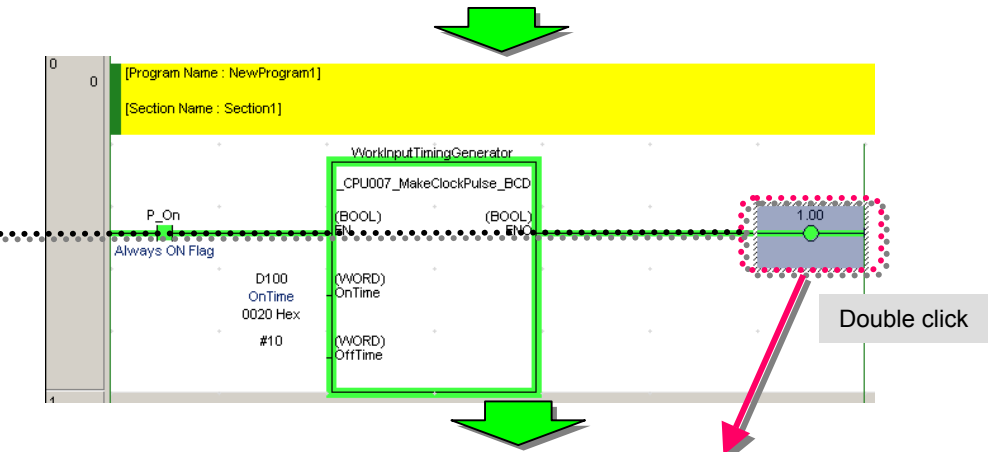
Edit the address to the required bit number (4.11 in the example)

-( )- Edit Coil

4.11 | Detail | OK | Cancel

Select [Program] ->[Online Edit] -> [Send Change]

Shortcut: [Ctrl]+{Shift]+[E]

WorkInputTimingGenerator
_CPU007_MakeClockPulse_BCD
P_On (BOOL) (BOOL) 4.11
EN ENO
Always ON Flag
D100 (WORD)
OnTime OnTime
0020 Hex
#10 (WORD)
OffTime
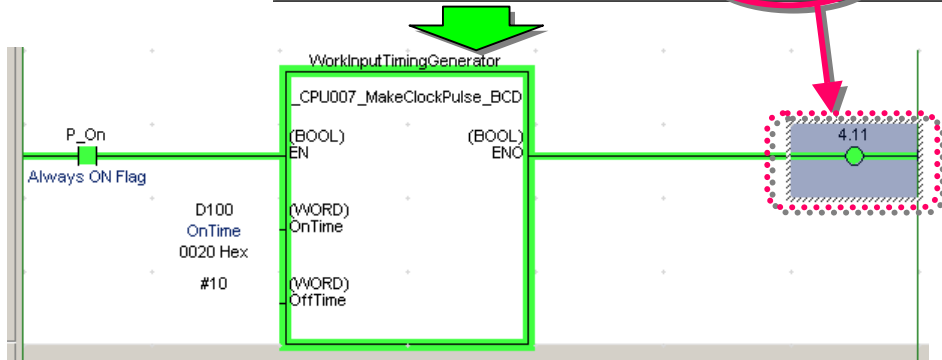
End

Chapter 3
Customize the OMRON FB Part file

Function Block

1. Explanation of target program

This chapter describes how to customize the OMRON FB Library using the OMRON FB Part file 'Make ON Time/OFF Time Clock Pulse in BCD'.
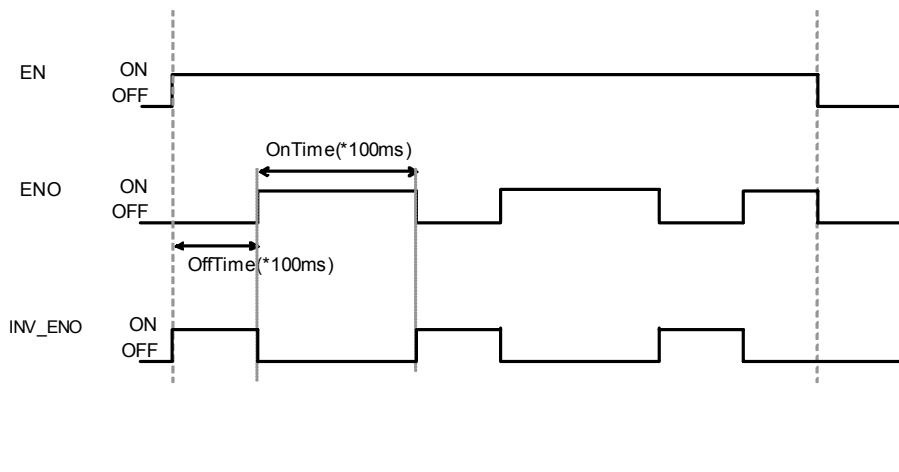
1-1. Changing File Specifications

The OMRON FB Part file 'Make ON Time/OFF Time Clock Pulse in BCD' is designed to repeatedly turn off the ENO for the specified OffTime (unit: 100 msec) and on for the specified OnTime (unit: 100 msec). In this example, the OMRON FB Part file will be changed to output an invert signal by adding the output parameter 'INV_ENO'.

1-2. Changing the contents of the OMRON FB Part file

To satisfy the requirement described above, the following changes must be made to OMRON FB Part file 'Make ON Time/OFF Time Clock Pulse in BCD'
1. Add an output parameter 'INV_ENO'.
2. Add ladder program to output the ENO for inverting the signal.

Caution
OMRON cannot guarantee the operation of a customized OMRON FB parts. Please be sure to check the process of your FB part sufficiently before customization and confirm the operation of each FB parts thoroughly after that.
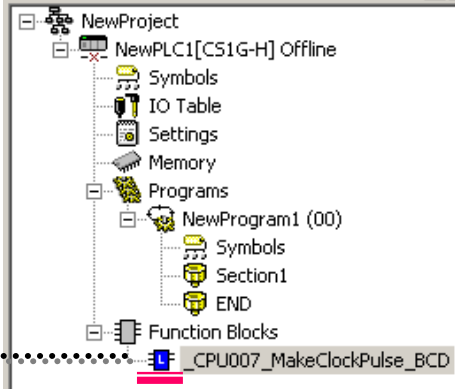
2. Copy the OMRON FB Part file

Import the 'Make ON Time/OFF Time Clock Pulse in BCD' Function Block Part file as explained in Chapter 1 (FB definition name: _CPU007_MakeClockPulse_BCD)

Select the OMRON FB Part icon then right click the mouse.
-> Copy
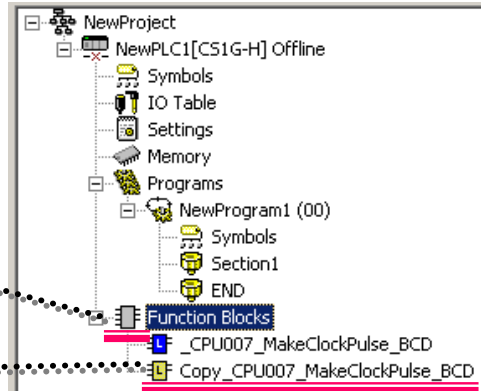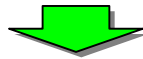
Select Function Block Definition icon and right click the mouse.
-> Paste

The OMRON FB Part file is pasted.

Change the FB definition name.
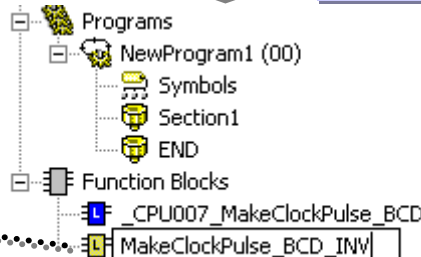
Select pasted Function Block icon and click mouse right button.
-> Rename
[MakeClockPulse_BCD_INV]

Note:
The user can't create Function Block Definitions
With name starting '_' (underscore).
Please use names not starting with '_'.
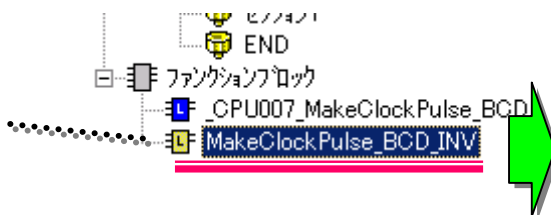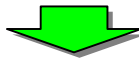
Select pasted Function Block icon and right click the mouse button.
-> Property

Or

ALT + ENT

Enable editing of the internal FB Program code.
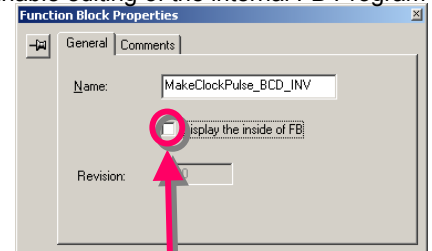
Tick the check box using the left mouse click.

3. Add a variable to the Function Block

Variable Table

Open the Function Block Ladder Editor.

Opens the Function Block Ladder Editor.

Select the Function Block icon using the mouse cursor and double click the left mouse button.

The original OMRON FB Part file is also able to display its ladder program, but cannot be edited.

Ladder Editor

Variable table

Name	Data Type	AT	Initial Value	Retai...	Comment
ENO	BOOL		FALSE		Indicates successful execution ...

| Internals | Inputs | Outputs | Externals |

Select Output tab in Variable Table using the mouse cursor And click the left mouse button.

Click the left mouse button and select Insert Variable(I).

Enter a new variable name.

New Variable

Name: INV_ENO

Data Type: BOOL

Usage: Output

Initial Value: FALSE

☐ Retain

Comment: Inverting output of ENO

OK

Cancel

Advanced...

Select BOOL for bit data.

Confirm the entered variable is correct.

Name	Data Type	AT	Initial Value	Retai...	Comment
ENO	BOOL		FALSE		Indicates successful execution ...
INV_ENO	BOOL		FALSE		Inverting output of ENO

4. Changing the Function Block Ladder

Add the required ladder diagram on Function Block Ladder edit field.
Move the cursor to the left column of the next rung.

Name	Data Type	AT	Initial Value	Retai...	Comment	
P_0_02s	BOOL				0.02 second clock pulse bit	
P_0_1s	BOOL				0.1 second clock pulse bit	
P_0_2s	BOOL				0.2 second clock pulse bit	
P_1min	BOOL				1 minute clock pulse bit	

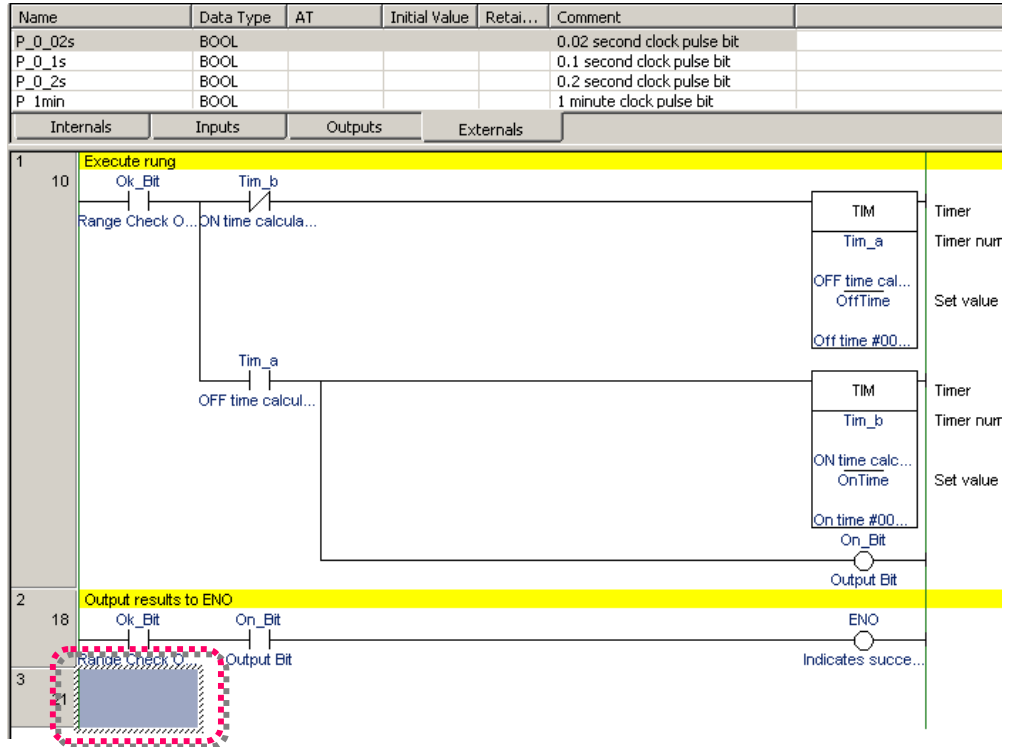| Internals	Inputs	Outputs	Externals

1 — Execute rung

```
10   Ok_Bit        Tim_b                                          TIM        Timer
     ┤ ├           ┤/├                                            Tim_a      Timer num
     Range Check O...  ON time calcula...
                                                                  OFF time cal...
                                                                  OffTime    Set value
                                                                  Off time #00...

              Tim_a                                               TIM        Timer
              ┤ ├                                                 Tim_b      Timer num
              OFF time calcul...
                                                                  ON time calc...
                                                                  OnTime     Set value
                                                                  On time #00...
                                                                  On_Bit
                                                                  ─○─
                                                                  Output Bit
```

2 — Output results to ENO

```
18   Ok_Bit        On_Bit                                         ENO
     ┤ ├           ┤ ├                                            ─○─
     Range Check O...  Output Bit                                 Indicates succe...
```

```
3
21   [         ]
```

4-1. Entering a Contact

-|/|- New Closed Contact ☒

`ENO` ▼ Detail >> | OK | Cancel


```
     Area check OK...  Bit for output
          ENO
21   ─┤/├──           [            ]
     Indicates succe...

22
```

-( )- New Coil ☒

`INV_ENO` ▼ Detail >> | OK | Cancel

```
     Area check OK...  Bit for output                          Indicates succe...
          ENO                                                      INV_ENO
21   ─┤/├──────────────────────────────────────────────────────────○─
     Indicates succe...                                        Inverting output...
```

Left side key sequence:

/

ENO

ENT

O

INV_ENO

ENT

5-1. How to delete unused Function Block definitions

When you delete unused Function Block definitions, it is not enough just to delete the Function Block invocations (instructions).
This is because the Function Block instance definitions are registered in the global symbol table.
At this situation, when the compile (program check) is done, then the unused function block instances will be shown on the output window. You can identify the unused function block instance definitions and delete them easily.
The Function Block definitions and Function Block instances are a part of user program in the CPU unit even if they are not called, so it is recommended to delete unused Function Block definitions and instances before transferring the program to the CPU unit.

Execute Compile **F7** key
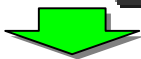
Result of Compilation

Double click mouse left button

Del key

Confirm Symbol Delete

Are you sure you want to delete symbol aaaa?

Yes   No

Click mouse left button

Function Block definition will be deleted.

5-2. Memory allocation for Function Blocks

It is necessary to allocate required memory for each function block instances to execute Function Blocks.
CX-Programmer allocates the memory automatically based on the following setting dialog information.
(PLC menu -> Function Block Memory -> Function Block Memory Allocation)
There are 4 types of areas, 'Not retain', 'Retain', 'Timers', and 'Counters'. Please change the settings if requires.

- Notice when changing the settings

    If you change the 'Not retain' or 'Retain' area, please consider the allocated memory areas for the special IO unit and CPU SIO unit.

- Special memory area for the Function Blocks

    CS1/CJ1-H/CJ1M CPUs (unit version: 3.0 or higher) have a special memory area which is extended hold (H) relay area.
    The address of the area is from H512 to H1535. CX-Programmer sets the area as a default.
    Please note that the area cannot be used for the operands of ladder instructions.

Function Block Memory Allocation [NewPLC1]

FB Instance Area	Start Address	End Address	Size	
Non Retain	H512	H1407	896	OK
Retain	H1408	H1535	128	Cancel
Timers	T3072	T4095	1024	Edit...
Counters	C3072	C4095	1024	Default
				Advanced...

Chapter 4

How to use
the ST(Structured Text)language

1. What is the ST Language?

The ST (Structured Text) language is a high-level language code for industrial controls (mainly PLCs) defined by the IEC 61131-3 standard.
It has many control statements, including IF-THEN-ELSE-END_IF, FOR / WHILE loop, and many mathematical functions such as SIN / LOG. it is suitable for mathematical processing.
The ST language supported by CX-Programmer is in conformance with IEC 61131-3 standard.
The arithmetic functions in CX-Programmer Ver.5.0 are as follows:

sine (SIN), cosine (COS), tangent (TAN), arc-sine (ASIN), arc-cosine (ACOS), arc-tangent (ATAN), square root (SQRT), absolute value (ABS), logarithm (LOG), natural-logarithm (LN), natural-exponential (EXP), exponentiation (EXPT)

```
(* Initial Settings *)
XMT[1] := 2;
XMT[2] := 7;
N := 2;

(* CRC16 *)
CRCTMP := 16#FFFF;
FOR I := 1 TO N DO
    CRCTMP := CRCTMP XOR XMT[1];
    FOR J := 1 TO 8 DO
        CT := CRCTMP AND 1;
        IF CRCTMP < 0 THEN
            CH := 1;
            CRCTMP := CRCTMP AND 16#7FFF; (* CRCTMP & 0x7FFF *)
        ELSE
            CH := 0;
        END_IF;
        UINT_CRCTMP := WORD_TO_UINT(CRCTMP) / 2;
        CRCTMP := UINT_TO_WORD(UINT_CRCTMP);
        IF CH = 1 THEN
            CRCTMP := CRCTMP OR 16#4000;        (* CRCTMP OR 0x4000 *)
        END_IF;
        IF CT = 1 THEN
            CRCTMP := CRCTMP XOR 16#A001; (* CRCTMP XOR 0xA001 *)
        END_IF;
    END_FOR;
END_FOR;

IF CRCTMP < 0 THEN
    CL := 1;
    CRCTMP := CRCTMP AND 16#7FFF;       (* CRCTMP & 0x7FFF *)

ELSE
    CL := 0;
END_IF;

C_1 := CRCTMP AND 16#FF;                (* CRCTMP & 0xFF *)
CRCTMP := CRCTMP AND 16#7F00;      (* CRCTMP & 0x7F00 *)
UINT_CRCTMP := WORD_TO_UINT(CRCTMP) / 256;
C_2 := UINT_TO_WORD(UINT_CRCTMP);
```

Reference: The IEC 61131 standard is an international standard for programming Programmable Logic Controllers (PLC), defined by the International Electro-technical Commission (IEC).
The standard consists of 7 parts, with part 3 defining the programming of PLCs.

2. Explanation of the target program

This example describes how to create an ST program in a Function Block to calculate the average value of a measured thickness.

ThicknessAverage
AverageCalc_3value

(BOOL) EN	(BOOL) ENO
(REAL) x	(REAL) score
(REAL) y	
(REAL) z	
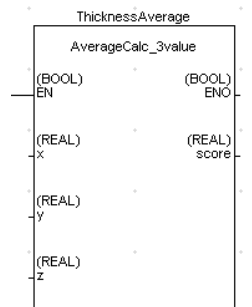
The data type should be set to REAL to store the data.
REAL type allows values with 32 bits of length, see range below:-
$-3.402823 \times 10^{38} \sim -1.175494 \times 10^{-38.}$, 0,
$+1.175494 \times 10^{-38} \sim +3.402823 \times 10^{38}$

FB definition name AverageCalc_3Value

Input symbols   x(REAL type), y(REAL type), z(REAL type)

Output symbol   score(REAL type)

ST Program definition   score := (x + y + z) / 3.0;

Substitute a value to a symbol is expressed by " := ".

Enter " ; " (semicolon) to complete the code.

Explanation of target Program		Create new FB Definition		Entering Symbols		Creating ST Program		Creating Ladder Program and check

3. Create a Function Block using ST

Create a Function Block using Structured Text.
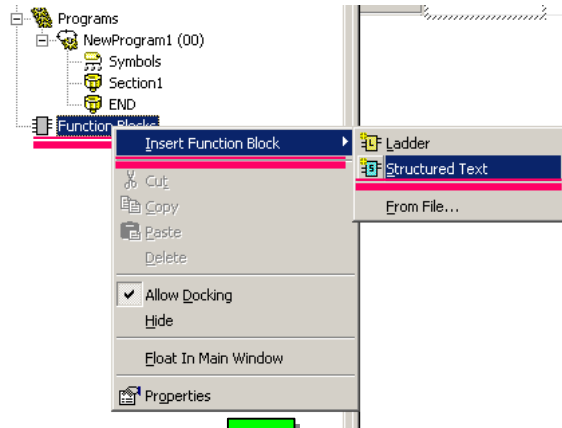
Select the Function Block icon using a mouse cursor, and click the right mouse button.
-> Insert Function Block(I)
->Structured Text(S)

A New Function Block definition is created.

Change the Function Block definition name
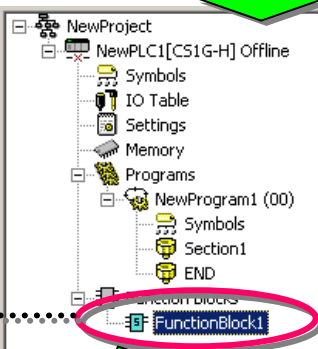
Select the Function Block definition icon using the mouse cursor and right click the mouse button.
Select Paste.
-> Rename
Enter
[AverageCalc_3value]
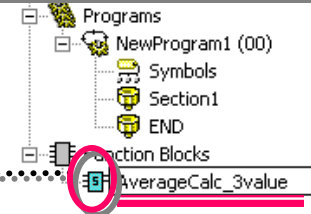
Open Function Block ST Editor

Select Function Block definition Icon by mouse cursor and double click the left mouse button.

Variable Table

ST Edit Field

Explanation of target Program	Create new FB Definition	Entering Symbols	Creating ST Program	Creating Ladder Program and check

4. Entering Variables in to Function Blocks

Select Variable Table.

Name	Data Type	AT	Initial Value	Retai...	Comment
EN	BOOL		FALSE		Controls execution of the Func...

Select the Input tab using the mouse cursor.

Internals · **Inputs** · Outputs · Externals

Select Insert from the Pop-up menu.

Enter data for the following.
Name
Data type
Comment

New Variable

Name: x → Enter a variable name

Data Type: REAL → Select REAL

Usage: Input

Initial Value: 0   ☐ Retain

Comment: Input value → Enter and applicable comment

OK   Cancel   Advanced...
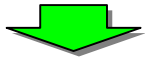
Enter input symbol x, output symbols y,z by repeating the process above.

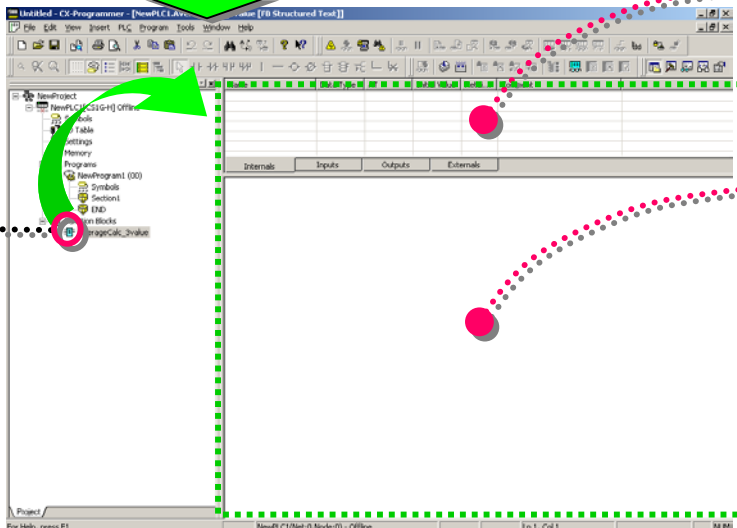| Name	Data Type	AT	Initial Value	Retai...	Comment
EN	BOOL		FALSE		Controls execution of the Func...
x	REAL		0.0		Input value 1
y	REAL		0.0		Input value 2
z	REAL		0.0		Input value 3

Input Variables

Name	Data Type	AT	Initial Value	Retai...	Comment
ENO	BOOL		FALSE		Indicates successful execution ...
score	REAL		0.0		Avrage

Output Variables

Reference: The copy and paste operation is available in FB Header.

Reference: The order of the variables in the FB table becomes the order of parameters on FB instance (invocation) in the normal ladder view.
To change the order, it is possible to drag & drop variables within the table.

ThicknessAverage

AverageCalc_3value

Name	Data Type	A
EN	BOOL	
x	REAL	
y	REAL	
z	REAL	

Name	Data Ty
ENO	BOOL
score	REAL

(BOOL) EN   (BOOL) ENO
(REAL) x   (REAL) score
(REAL) y
(REAL) z
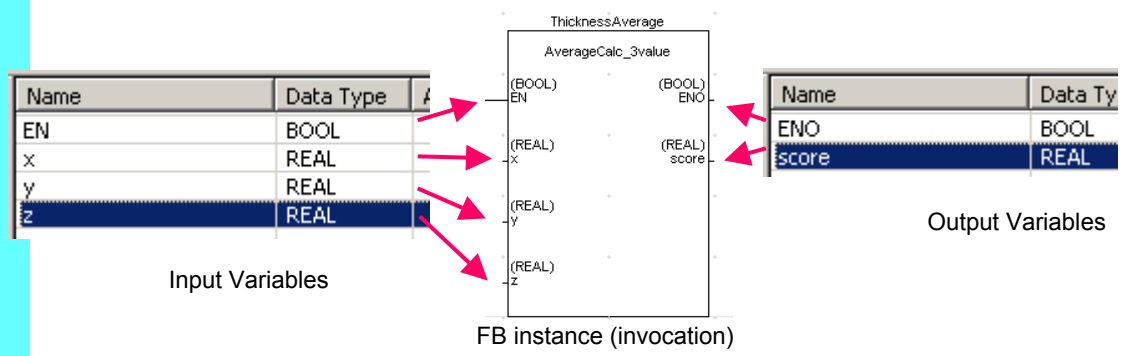
Input Variables

Output Variables

FB instance (invocation)

Explanation of target Program	Create new FB Definition	Entering Symbols	Creating ST Program	Creating Ladder Program and check

5. Entry of ST program

Select the ST Editor text field in the Function Block ST Editor window.
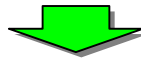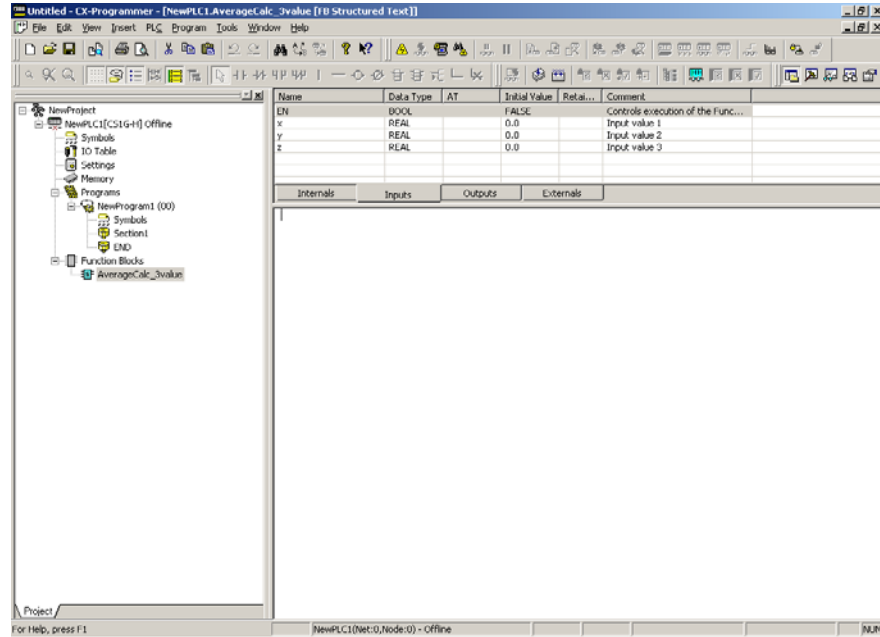
Enter text into the field: "score := (x + y + z) / 3.0;".

```
score := (x + y + z) / 3.0;
```

When the input expression is a real type calculation, please enter the constant value with decimal point and zero for single decimal places, e.g. '3.0'.

Reference: User may type Comments in the ST program.
Enter '(*' and '*)' both ends of comment strings, see below.
This is useful for recording change history, process expressions, etc.

```
(* Created by Suzuki 5/21/2004 *)
score := (x + y + z) / 3.0;
```

Explanation of target Program	Create new FB Definition	Entering Symbols	Creating ST Program	Creating Ladder Program and check

6. Entering the FB to the Ladder Program and error checking

Enter the following FB into the ladder program.
Instance name: ThicknessAvarage
Input parameters: D0, D2, D4
Output parameter: D6
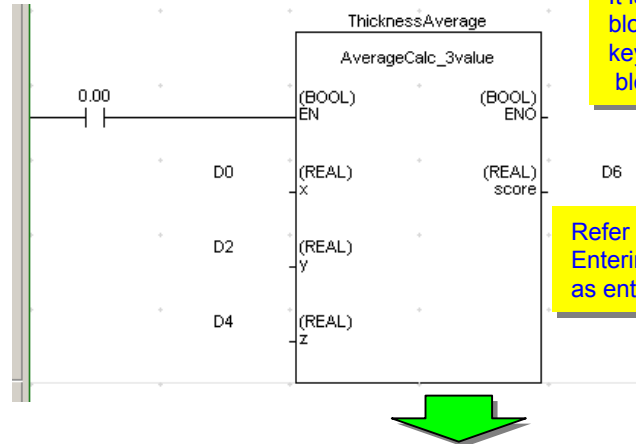
It is able to jump the referred function block definition by entering [Shift]+[F] key when the cursor is in the function block instance.

Refer page 2-7 for entering FB instances. Entering ST FB instances is the same as entering FB Ladder instances.

Perform a programs check before transferring the program.
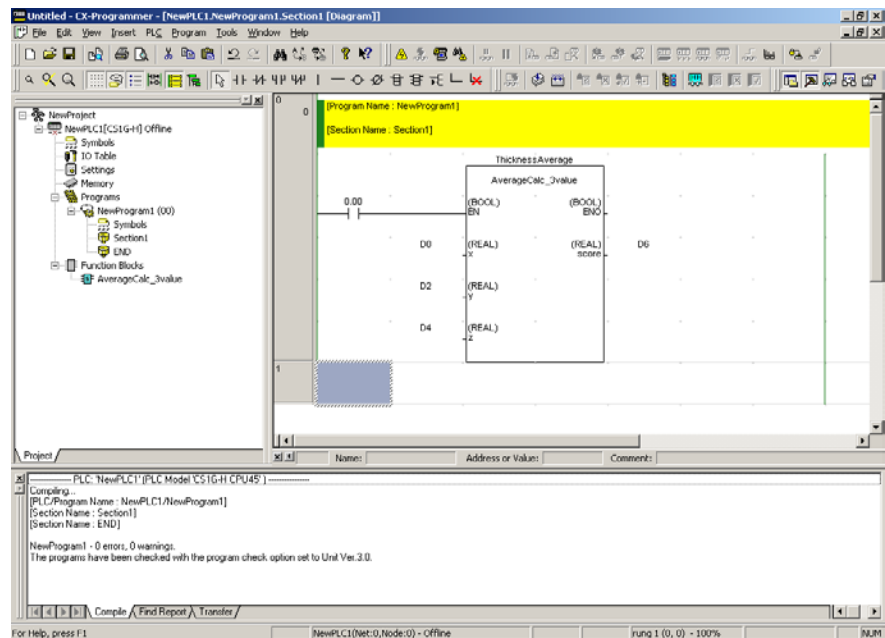
Refer page 2-9 for program checking.
The functionality is the same as for Function Block Ladder instances.

It is possible to change or add variables in the Function Block after inputting FB instance into the ladder editor. If modified, the Ladder editor changes the color of the left bus-bar of the rung containing the changed Function Block.
When this occurs, please select the instance in the Ladder Editor using the mouse cursor, and select Update Function Block Instance (U) from the pop-up menu.

Transfer Program ➤ Monitoring

7. Program Transfer

Go online to the PLC with CX-Simulator and transfer the program.
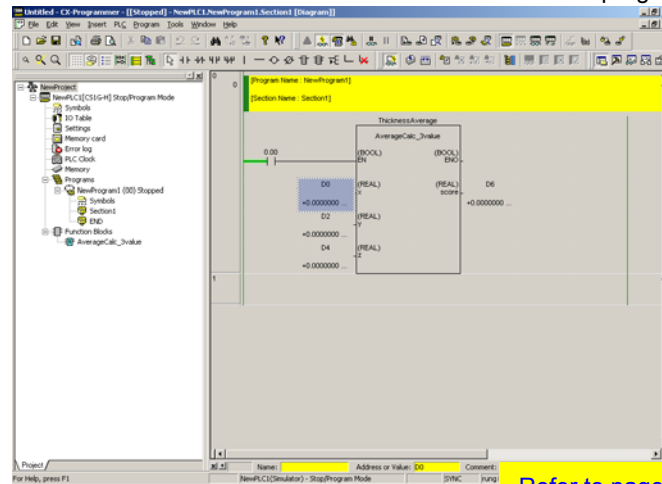
Refer to page 2-10 for steps to go online and transfer the program.
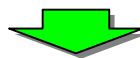
Change the PLC (Simulator) to Monitor mode.

Click 📟

The on/off status of contacts and coils can be monitored.

Click [Yes]

CX-Programmer v5.0

⚠ This command will affect the state of the connected PLC. Do you wish to continue ?

Yes   No
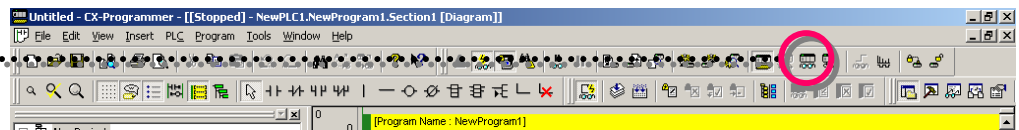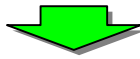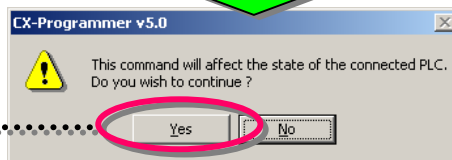
Confirm that the PLC is Monitor mode.
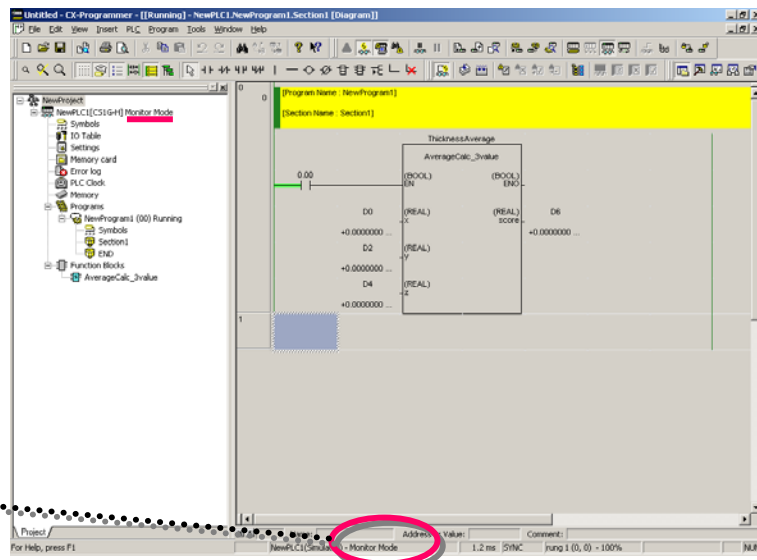
Transfer Program → Monitoring

8. Monitoring the Function Block execution

Monitors the present value of parameters in the FB instance using the Watch Window.
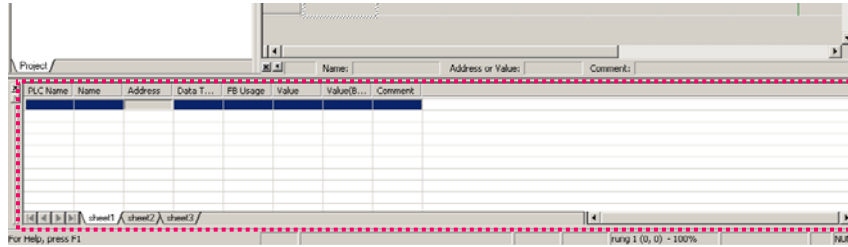
Display the Watch Window.

Alt + 3
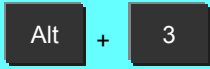
Open the Edit dialog.

ENT

Click Browse… button using the mouse left button.

Edit dialog

PLC: NewPLC1

Name or address: [ ] Browse…

Data Type / Format: BOOL (On/Off,Contact)

OK   Cancel

Click the ▼ button using the left mouse button, then select the following:
[Symbols of type]
[Name or address]

Find Symbol (Read only) - Any

Look in: NewPLC1

Symbols of type: BOOL

Name or address: [ ]   Edit Symbol

Symbol Information

Look in: NewPLC1

Symbols of type: REAL

dress: ThicknessAverage.x   Edit S

formation

Select REAL(32bit floating point)

BOOL
CHANNEL
DINT
DWORD
INT
LINT
LREAL
LWORD
REAL
UDINT
UDINT_BCD
UINT
UINT_BCD
ULINT
ULINT_BCD

Select ThicknessAvarage.x

ThicknessAverage.score
ThicknessAverage.x
ThicknessAverage.y
ThicknessAverage.z

Edit dialog

PLC: NewPLC1

Name or address: ThicknessAverage.x   Browse…

Data Type / Format: REAL (Floating Point,Double length)

OK   Cancel

Click [OK] button using the left mouse button.

When monitoring internal variables at debug phase, always use the Watch Window.
It is not possible to monitor inside a Function Block Definition.

PLC Name	Name	Address	Data Type / Format	FB Usage	Value	Value(Binary)	Comment
NewPLC1	ThicknessAverage.x	H513	REAL (Floating Point,Double length)	Input	+0.0000000 Float	+0.0000000 Float	Input value 1

sheet1 sheet2 sheet3
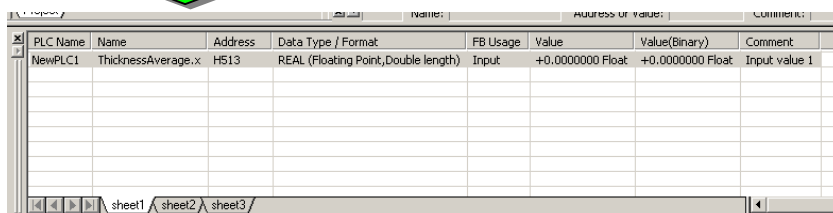
Reference: Example of an ST program using IF-THEN-ELSE-END_IF

The following ST program checks the average value calculated by the example of page 4-7 against a range (upper limit or lower limit).

FB Definition: OutputOfDecisionResult
Input symbols: score(REAL type), setover(REAL type), setunder(REAL type)
Output symbols: OK (BOOL type), overNG(BOOL type), underNG(BOOL type)

ST program:

```
IF score > setover THEN        (* If score > setover, *)
  underNG := FALSE;            (* Turn off underNG *)
  OK := FALSE;                 (* Turn off OK *)
  overNG := TRUE;             (* Turn on overNG *)

ELSIF score < setunder THEN (* if score =< setover  and score < setunder then *)
  overNG := FALSE;            (* Turn on overNG *)
  OK := FALSE;                 (* Turn off OK *)
  underNG := TRUE;            (* Turn on underNG *)

ELSE                           (* if setover > score > setunder then*)
  underNG := FALSE;           (* Turn off underNG *)
  overNG := FALSE;            (* Turn off overNG *)
  OK := TRUE;                 (* Turn off OK  *)

END_IF;                        (* end of IF section*)
```

Example of an FB instance (the instance name is 'ThicknessDecision')
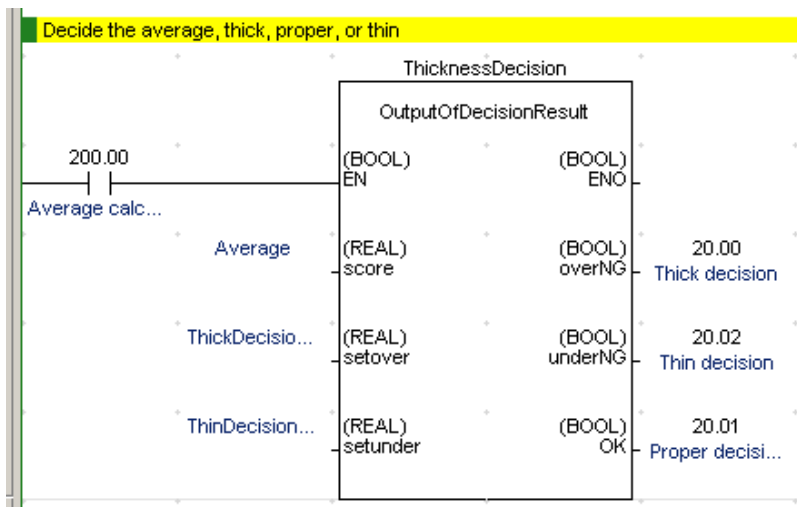
Useful Functions

It is possible to automatically display a list of symbol names or IO comments when entering the operands of instructions.
When entering the operand for contact or output (or special instructions), enter a string, and the dropdown list is automatically updated to display in symbol names or IO Comments using the defined string. Selecting the item from the list defines the operand information..
This is an efficient way of entering registered symbol information into the ladder.

Example: Enter text "Temperature" to the edit field in the operand dialog.

Click ▼ or push [F4] key; all symbols / address having IO comment containing the text 'temperature are listed. See below:-
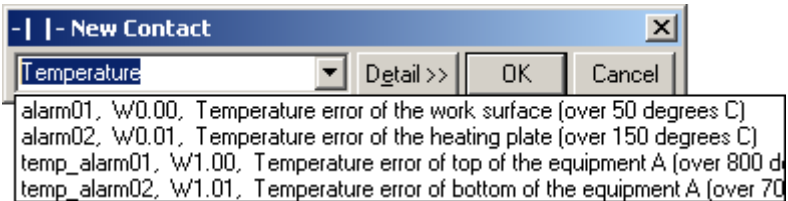
For instance, select 'temp_alarm01, W1.00, Temperature error of upper case of MachineA', from the list. The operand is set to be using symbol 'alarm01'.

Certain Terms and Conditions of Sale

1. <u>Offer; Acceptance.</u> These terms and conditions (these "<u>Terms</u>") are deemed part of all catalogs, manuals or other documents, whether electronic or in writing, relating to the sale of goods or services (collectively, the "<u>Goods</u>") by Omron Electronics LLC and its subsidiary companies ("<u>Seller</u>"). Seller hereby objects to any terms or conditions proposed in Buyer's purchase order or other documents which are inconsistent with, or in addition to, these Terms. Please contact your Omron representative to confirm any additional terms for sales from your Omron company.

2. <u>Prices.</u> All prices stated are current, subject to change without notice by Seller. Buyer agrees to pay the price in effect at time of shipment.

3. <u>Discounts.</u> Cash discounts, if any, will apply only on the net amount of invoices sent to Buyer after deducting transportation charges, taxes and duties, and will be allowed only if (i) the invoice is paid according to Seller's payment terms and (ii) Buyer has no past due amounts owing to Seller.

4. <u>Orders.</u> Seller will accept no order less than $200 net billing.

5. <u>Governmental Approvals.</u> Buyer shall be responsible for, and shall bear all costs involved in, obtaining any government approvals required for the importation or sale of the Goods.

6. <u>Taxes.</u> All taxes, duties and other governmental charges (other than general real property and income taxes), including any interest or penalties thereon, imposed directly or indirectly on Seller or required to be collected directly or indirectly by Seller for the manufacture, production, sale, delivery, importation, consumption or use of the Goods sold hereunder (including customs duties and sales, excise, use, turnover and license taxes) shall be charged to and remitted by Buyer to Seller.

7. <u>Financial.</u> If the financial position of Buyer at any time becomes unsatisfactory to Seller, Seller reserves the right to stop shipments or require satisfactory security or payment in advance. If Buyer fails to make payment or otherwise comply with these Terms or any related agreement, Seller may (without liability and in addition to other remedies) cancel any unshipped portion of Goods sold hereunder and stop any Goods in transit until Buyer pays all amounts, including amounts payable hereunder, whether or not then due, which are owing to it by Buyer. Buyer shall in any event remain liable for all unpaid accounts.

8. <u>Cancellation; Etc.</u> Orders are not subject to rescheduling or cancellation unless Buyer indemnifies Seller fully against all costs or expenses arising in connection therewith.

9. <u>Force Majeure.</u> Seller shall not be liable for any delay or failure in delivery resulting from causes beyond its control, including earthquakes, fires, floods, strikes or other labor disputes, shortage of labor or materials, accidents to machinery, acts of sabotage, riots, delay in or lack of transportation or the requirements of any government authority.

10. <u>Shipping; Delivery.</u> Unless otherwise expressly agreed in writing by Seller:
   a. Shipments shall be by a carrier selected by Seller;
   b. Such carrier shall act as the agent of Buyer and delivery to such carrier shall constitute delivery to Buyer;
   c. All sales and shipments of Goods shall be FOB shipping point (unless otherwise stated in writing by Seller), at which point title to and all risk of loss of the Goods shall pass from Seller to Buyer, provided that Seller shall retain a security interest in the Goods until the full purchase price is paid by Buyer;
   d. Delivery and shipping dates are estimates only.
   e. Seller will package Goods as it deems proper for protection against normal handling and extra charges apply to special conditions.

11. <u>Claims.</u> Any claim by Buyer against Seller for shortage or damage to the Goods occurring before delivery to the carrier must be presented in writing to Seller within 30 days of receipt of shipment and include the original transportation bill signed by the carrier noting that the carrier received the Goods from Seller in the condition claimed.

12. <u>Warranties.</u> (a) <u>Exclusive Warranty.</u> Seller's exclusive warranty is that the Goods will be free from defects in materials and workmanship for a period of twelve months from the date of sale by Seller (or such other period expressed in writing by Seller). Seller disclaims all other warranties, express or implied. (b) <u>Limitations.</u> SELLER MAKES NO WARRANTY OR REPRESENTATION, EXPRESS OR IMPLIED, ABOUT NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE OF THE GOODS. BUYER ACKNOWLEDGES THAT IT ALONE HAS DETERMINED THAT THE GOODS WILL SUITABLY MEET THE REQUIREMENTS OF THEIR INTENDED USE. Seller further disclaims all warranties and responsibility of any type for claims or expenses based on infringement by the Goods or otherwise of any intellectual property right. (c) <u>Buyer Remedy.</u> Seller's sole obligation hereunder shall be to replace (in the form originally shipped with Buyer responsible for labor charges for removal or replacement thereof) the non-complying Good or, at Seller's election, to repay or credit Buyer an amount equal to the purchase price of the Good; provided that in no event shall Seller be responsible for warranty, repair, indemnity or any other claims or expenses regarding the Goods unless Seller's analysis confirms that the Goods were properly handled, stored, installed and maintained and not subject to contamination, abuse, misuse or inappropriate modification. Return of any goods by Buyer must be approved in writing by Seller before shipment. Seller shall not be liable for the suitability or unsuitability or the results from the use of Goods in combination with any electrical or electronic components, circuits, system assemblies or any other materials or substances or environments. Any advice, recommendations or information given orally or in writing, are not to be construed as an amendment or addition to the above warranty.

13. <u>Damage Limits; Etc.</u> SELLER SHALL NOT BE LIABLE FOR SPECIAL, INDIRECT OR CONSEQUENTIAL DAMAGES, LOSS OF PROFITS OR PRODUCTION OR COMMERCIAL LOSS IN ANY WAY CONNECTED WITH THE GOODS, WHETHER SUCH CLAIM IS BASED IN CONTRACT, WARRANTY, NEGLIGENCE OR STRICT LIABILITY. Further, in no event shall liability of Seller exceed the individual price of the Good on which liability is asserted.

14. <u>Indemnities.</u> Buyer shall indemnify and hold harmless Seller, its affiliates and its employees from and against all liabilities, losses, claims, costs and expenses (including attorney's fees and expenses) related to any claim, investigation, litigation or proceeding (whether or not Seller is a party) which arises or is alleged to arise from Buyer's acts or omissions under these Terms or in any way with respect to the Goods. Without limiting the foregoing, Buyer (at its own expense) shall indemnify and hold harmless Seller and defend or settle any action brought against Seller to the extent that it is based on a claim that any Good made to Buyer specifications infringed intellectual property rights of another party.

15. <u>Property; Confidentiality.</u> The intellectual property embodied in the Goods is the exclusive property of Seller and its affiliates and Buyer shall not attempt to duplicate it in any way without the written permission of Seller. Notwithstanding any charges to Buyer for engineering or tooling, all engineering and tooling shall remain the exclusive property of Seller. All information and materials supplied by Seller to Buyer relating to the Goods are confidential and proprietary, and Buyer shall limit distribution thereof to its trusted employees and strictly prevent disclosure to any third party.

16. <u>Miscellaneous.</u> (a) <u>Waiver.</u> No failure or delay by Seller in exercising any right and no course of dealing between Buyer and Seller shall operate as a waiver of rights by Seller. (b) <u>Assignment.</u> Buyer may not assign its rights hereunder without Seller's written consent. (c) <u>Amendment.</u> These Terms constitute the entire agreement between Buyer and Seller relating to the Goods, and no provision may be changed or waived unless in writing signed by the parties.
(d) <u>Severability.</u> If any provision hereof is rendered ineffective or invalid, such provision shall not invalidate any other provision. (e) <u>Setoff.</u> Buyer shall have no right to set off any amounts against the amount owing in respect of this invoice. (f) As used herein, "<u>including</u>" means "including without limitation".

Certain Precautions on Specifications and Use

1. <u>Suitability of Use.</u> Seller shall not be responsible for conformity with any standards, codes or regulations which apply to the combination of the Good in the Buyer's application or use of the Good. At Buyer's request, Seller will provide applicable third party certification documents identifying ratings and limitations of use which apply to the Good. This information by itself is not sufficient for a complete determination of the suitability of the Good in combination with the end product, machine, system, or other application or use. The following are some examples of applications for which particular attention must be given. This is not intended to be an exhaustive list of all possible uses of this Good, nor is it intended to imply that the uses listed may be suitable for this Good:
   (i) Outdoor use, uses involving potential chemical contamination or electrical interference, or conditions or uses not described in this document.
   (ii) Energy control systems, combustion systems, railroad systems, aviation systems, medical equipment, amusement machines, vehicles, safety equipment, and installations subject to separate industry or government regulations.
   (iii) Systems, machines and equipment that could present a risk to life or property. Please know and observe all prohibitions of use applicable to this Good.
   NEVER USE THE PRODUCT FOR AN APPLICATION INVOLVING SERIOUS RISK TO LIFE OR PROPERTY WITHOUT ENSURING THAT THE SYSTEM AS A WHOLE HAS BEEN DESIGNED TO ADDRESS THE RISKS, AND THAT THE SELLER'S PRODUCT IS PROPERLY RATED AND INSTALLED FOR THE INTENDED USE WITHIN THE OVERALL EQUIPMENT OR SYSTEM.

2. <u>Programmable Products.</u> Seller shall not be responsible for the user's programming of a programmable Good, or any consequence thereof.

3. <u>Performance Data.</u> Performance data given in this catalog is provided as a guide for the user in determining suitability and does not constitute a warranty. It may represent the result of Seller's test conditions, and the user must correlate it to actual application requirements. Actual performance is subject to the Seller's Warranty and Limitations of Liability.

4. <u>Change in Specifications.</u> Product specifications and accessories may be changed at any time based on improvements and other reasons. It is our practice to change part numbers when published ratings or features are changed, or when significant construction changes are made. However, some specifications of the Good may be changed without any notice. When in doubt, special part numbers may be assigned to fix or establish key specifications for your application. Please consult with your Seller's representative at any time to confirm actual specifications of purchased Good.

5. <u>Errors and Omissions.</u> The information in this catalog has been carefully checked and is believed to be accurate; however, no responsibility is assumed for clerical, typographical or proofreading errors, or omissions.

OMRON®

OMRON ELECTRONICS LLC

1 Commerce Drive
Schaumburg, IL 60173
847.843.7900
For US technical support or
other inquiries: 800.556.6766

OMRON CANADA, INC.

885 Milner Avenue
Toronto, Ontario M1B 5V8
416.286.6465

OMRON ON-LINE

Global - http://www.omron.com
USA - http://www.omron.com/oei
Canada - http://www.omron.ca

UNITED STATES

To locate a Regional Sales Office, local Distributor or
to obtain product information, call: 847.843.7900

CANADA REGIONAL SALES OFFICES

Ontario	Toronto	416.286.6465
	Kitchener	519.896.1144
	Kingston	613.376.3968
Quebec	Montreal	514.636.6676
British Columbia	Vancouver	604.522.8855
Alberta	Edmonton	403.440.0818
	Calgary	403.257.3095

BRAZIL SALES OFFICE

Sao Paulo	55.11.5564.6488

ARGENTINA SALES OFFICE

Cono Sur	54.114.787.1129

MEXICO SALES OFFICES

Florida 954.227.2121	Ciudad Juarez 656.623.7083
Mexico, D.F. 555.534.1195	Monterrey, N.L. 818.377.4281