Function Block
Introduction Guide

* You need Acrobat Reader 4.0 or greater versions in your PC to display the PDF file.
## Contents

### Chapter 1 OMRON FB Library
1. What is a Function Block? .................................................. 1-1
2. An Example of a Function Block ........................................ 1-2
3. Overview of the OMRON FB Library ................................... 1-3
  3-1. Benefits of the OMRON FB Library ................................. 1-3
  3-2-1. Example of using the OMRON FB Library - 1 .................. 1-4
  3-2-2. Example of using the OMRON FB Library - 2 .................. 1-5
3-3. Content of the OMRON FB Library ................................... 1-6
  3-3-1. OMRON FB Part Files .............................................. 1-6
  3-3-2. Library reference ................................................... 1-6
3-4. File Catalog and Where to Access the OMRON FB Library ....... 1-7
  3-4-1. Catalog of OMRON FB Library files .............................. 1-7
  3-4-2. CX-Programmer installation CD ................................. 1-7
  3-4-3. Accessing OMRON FB Library files from Web server ....... 1-7

### Chapter 2 How to use the OMRON FB Library
1. Explanation of the target program ...................................... 2-1
  1-1. Application Specifications ........................................... 2-1
  1-2. Specifications of the OMRON FB Part file ......................... 2-1
  1-3. Input program ....................................................... 2-2
2. Opening a new project and setting the Device Type .................. 2-3
3. Main Window functions .................................................. 2-4
4. Import the OMRON FB Part file ........................................ 2-5
5. Program Creation ........................................................ 2-6
  5-1. Enter a Normally Open Contact .................................... 2-6
  5-2. Entering an Instance .................................................. 2-7
  5-3. Entering Parameters ................................................... 2-7
6. Program Error Check (Compile) ......................................... 2-9
7. Going Online ............................................................. 2-10
8. Monitoring - 1 .............................................................. 2-11
9. Monitoring - 2 Change Parameter Current Value ...................... 2-12
10. Online Editing ............................................................ 2-13

### Chapter 3 Customize the OMRON FB Part file
1. Explanation of target program .......................................... 3-1
  1-1. Changing File Specifications ........................................ 3-1
  1-2. Changing the contents of the OMRON FB Part file ............... 3-1
2. Copy the OMRON FB Part file .......................................... 3-2
3. Add a variable to the Function Block .................................. 3-3
4. Changing the Function Block Ladder .................................. 3-4
  4-1. Entering a Contact ................................................... 3-4
5. Supplemental Information ............................................... 3-5
  5-1. How to delete unused Function Block definitions .............. 3-5
  5-2. Memory allocation for Function Blocks .......................... 3-5

### Chapter 4 How to use the ST (Structured Text) language
1. What is the ST Language? ................................................. 4-1
2. Explanation of the target program ...................................... 4-1
3. Create a Function Block using ST ...................................... 4-2
4. Entering Variables in to Function Blocks ............................. 4-3
5. Entry of ST program ..................................................... 4-4
6. Entering the FB to the Ladder Program and error checking ....... 4-5
7. Program Transfer .......................................................... 4-6
8. Monitoring the Function Block execution ............................ 4-7
Reference: Example of an ST program using IF-THEN-ELSE-END_IF 4-8

Useful Functions ............................................................. Appendix
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.

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.
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.

By enabling the input parameter to be editable, it is possible to allow an arbitrary time limit circuit.
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.
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)
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.
3-4. File Catalog and Where to Access the OMRON FB Library

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

<table>
<thead>
<tr>
<th>Type</th>
<th>Target components</th>
<th>Number of OMRON FB Part files (at the time of July '04)</th>
</tr>
</thead>
<tbody>
<tr>
<td>FA components</td>
<td>Temperature controller, Smart sensor, ID sensor, Vision sensor, 2 dimensions bar code reader</td>
<td>approx. 80</td>
</tr>
<tr>
<td>PLC</td>
<td>CPU unit, Memory card, Special CPU IO unit (Ethernet, Controller Link, DeviceNet unit, Temperature control unit)</td>
<td>approx. 95</td>
</tr>
</tbody>
</table>
| 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.

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.
Chapter 2
How to use the OMRON FB Library

Function Block
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. Explanation of the target program

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:

- Function description:
  - EN will be OFF for the time set in OFF time and then will be ON for the time set in ON time.
- EN input condition:
  - Connect the EN input to the Always ON flag (P_On).
- Restrictions on input variables:
  - If the input variables are out of range, the EN flag will turn OFF and the FB will not be processed.
  - Set the ON time and OFF time (input variables to between #000 and #9999 in BCD (100 ms units). If a setting is not within range, EN will be turned OFF.
- Application example:
  - In the following example, the EN flag will be positive to ON for 5s and OFF for 3s.
- Related FBs:
  - Use the clock FB for the time clock mode in the PLC in the setup.
  - Binary mode: Make ON Time/OFF Time Clock Pulse in Binary (CPU007_T8aClockPulse_B) BCD mode: Make ON Time/OFF Time Clock Pulse in BCD (CPU007_T8aClockPulse_BCD)
1-3. Input program

Create the following ladder program:-

![Ladder Diagram]

[Reference] If created as a straightforward ladder diagram, the program would be as below:-

![Straightforward Ladder Diagram]
2. Opening a new project and setting the Device Type

Click the toolbar button [New] in CX-Programmer.

Click the left mouse button.

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

Click the left mouse button to select CPU type.

Click [OK] to decide the selected CPU type.
3. Main Window functions

The main window functionality is explained here.

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

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.

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

Confirm cursor position is at the upper left of Ladder Window to start programming.

5-1. Enter a Normally Open Contact

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

“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

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

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

Applies a name for the specific process in the diagram.

Shows FB instance (invocation) 'WorkInputTimingGenerator'.

5-3. Entering Parameters

Move the cursor to the left of input parameter.

Enter the address. [d100]

Choose an address for the input parameter ‘OnTime’.
Enter the remaining parameters in the same way.

Please add the following prefix for entering constants as parameters:
- "#" (Hexadecimal/BCD)
- "&" (Decimal)
6. Program Error Check (Compile)

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

- Click on errors in the Output Window to display the error rung in red.
- 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.
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]
- The background color of the Ladder Window changes to gray.
- Scan time is displayed (except during Program Mode).
- The operating mode of the active PLC is shown.
- Click [OK]
- The CX-Simulator Console box is shown.
- Program transfer starts.

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.)
8. Monitoring - 1

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

Click Change the PLC (Simulator) to Monitor mode.

Click [Yes].

The monitored area is displayed in a specified color.

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.

••• toggles PLC monitoring on/off

The monitored area is displayed in a specified color.

The current values of parameters are shown.
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.

Click [Set]

Please add the following prefix for entering constants as parameters:

- "#" (Hexadecimal/BCD)
- "&" (Decimal)
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.

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

Shortcut: [Ctrl]+[E]

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

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

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

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

Double click
Chapter 3
Customize the OMRON FB Part file
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 pasted Function Block icon and click mouse right button.
-> Rename
[MakeClockPulse_BCD_INV]

Select pasted Function Block icon and right click the mouse button.
-> Property
Or

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

Note:
The user can’t create Function Block Definitions With name starting '_' (underscore).
Please use names not starting with '_'.

Note:
The user can’t create Function Block Definitions With name starting '_' (underscore).
Please use names not starting with '_'.

Explanation of target Program
Explanation of target Program
Copy of FB part
Change of FB Definition
Copy of FB part
Change of FB Definition
3. Add a variable to the Function Block

Open the Function Block Ladder Editor.

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

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.

Select BOOL for bit data.

Enter a new variable name.

Select BOOL for bit data.

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

Variable Table

Variable Name | Data Type | AT | Initial Value | Retain | Comment
---|---|---|---|---|---
END | BOOL | | FALSE | | Indicates successful execution...

Ladder Editor
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.

<table>
<thead>
<tr>
<th>Name</th>
<th>Data Type</th>
<th>AT</th>
<th>Initial Value</th>
<th>Rate (s)</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>FO, ON</td>
<td>BOOL</td>
<td></td>
<td></td>
<td></td>
<td>0.0 second pulse bit</td>
</tr>
<tr>
<td>PO, ON</td>
<td>BOOL</td>
<td></td>
<td></td>
<td></td>
<td>0.5 second pulse bit</td>
</tr>
<tr>
<td>PO, OFF</td>
<td>BOOL</td>
<td></td>
<td></td>
<td></td>
<td>0.5 second pulse bit</td>
</tr>
<tr>
<td>F, Time</td>
<td>BOOL</td>
<td></td>
<td></td>
<td></td>
<td>1 minute pulse bit</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Intervals</th>
<th>Inputs</th>
<th>Outputs</th>
<th>Externals</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

4-1. Entering a Contact

![Diagram of entering a contact]

![Diagram of new coil]

Explanation of target Program

Copy of FB part

Change of FB Definition
5. Supplemental Information

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

Are you sure you want to delete symbol 'aaa'?

Yes

No

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.
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)

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.
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)

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.

A New Function Block definition is created.

Change the Function Block definition name

Note: The user can’t create Function Block Definitions with names starting '_' (underscore). Please use names not starting with '_'.

Variable Table

ST Edit Field
4. Entering Variables into Function Blocks

Select Variable Table.

Enter data for the following.
- Name
- Data type
- Comment

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

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.
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;
```

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 ST(2004) *)
Score := (x + y + z) / 3.0;
```
6. Entering the FB to the Ladder Program and error checking

Enter the following FB into the ladder program.
Instance name: ThicknessAverage
Input parameters: D0, D2, D4
Output parameter: D6

Perform a programs check before transferring the program.

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.
7. Program Transfer

Go online to the PLC with CX-Simulator and transfer the program.

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

Refer to page 2-10 for steps to go online and transfer the program.

Confirm that the PLC is Monitor mode.
8. Monitoring the Function Block execution

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

1. Display the Watch Window.
   - Press Alt + 3

2. Open the Edit dialog.
   - Click Browse… button using the mouse left button.

3. Select REAL(32bit floating point)
   - Click the button using the left mouse button, then select the following:
     - Symbols of type [REAL]
     - Name or address [ThicknessAverage.x]

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

```
<table>
<thead>
<tr>
<th>Decide the average, thick, proper, or thin</th>
</tr>
</thead>
<tbody>
<tr>
<td>200.00</td>
</tr>
<tr>
<td>Average calc...</td>
</tr>
<tr>
<td>ThicknessDecision</td>
</tr>
<tr>
<td>OutputOfDecisionResult</td>
</tr>
<tr>
<td>(BOOL) (BOOL)</td>
</tr>
<tr>
<td>EN EN</td>
</tr>
<tr>
<td>20.00 23.00</td>
</tr>
<tr>
<td>Thick decision</td>
</tr>
<tr>
<td>Thin decision</td>
</tr>
<tr>
<td>Proper decision</td>
</tr>
</tbody>
</table>
```
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.

![Image]

Click or push [F4] key: all symbols / address having IO comment containing the text 'temperature' are listed. See below:

![Image]

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'.

![Image]