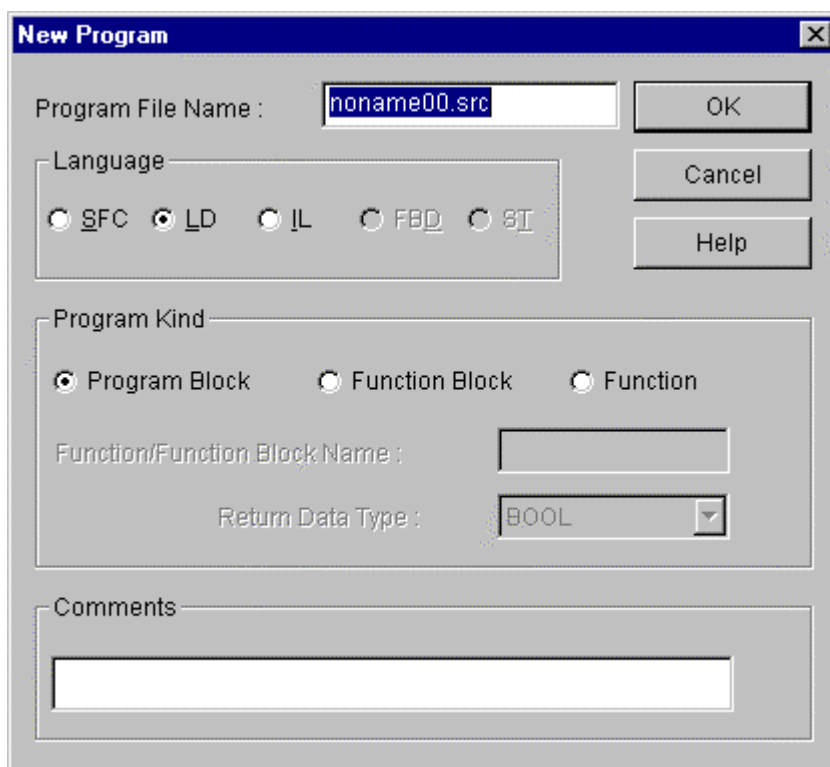# Chapter 5. Edit a Program

This program is a unit to create an order language for the PLC system is run. This chapter shows the methods; create a program and save it to file, edit each language, and etc.

5.1. Create a Program

To create a new program,

◆   Select **Program-New Program...**(🔲) in the menu..

◆   Enter Program File name.
◆   Select a language in **Language** option button.(GLOFA PLC supports SFC, LD, and IL. FBD and ST will be supported later.)
◆   Select a program in **Program Kind** option button. If you select **Function/Function Block** to create a user-defined function/function block, refer to the chapter 10 Library. Select **Program Block** option in this chapter.

Note
You can select a **Function** option button in only case of LD or IL language. You can not create a user-defined function by SFC.

Note
Function/Function Block name is only in active when **Function/Function Block** option button is selected and **Output Data Type** is in active only when **Function** option button is selected. (Refer to 10.3. Making a library file defined by user' for detail comment)

◆ Enter Comment in **Comment** box.
◆ Click **OK** button.
◆ If a new program is not existed in the project, the following dialog box appears.
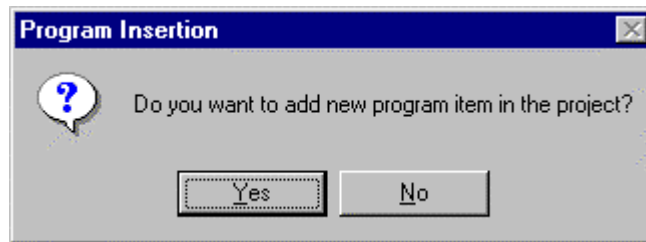

◆ Click **Yes** button. Then the **Define Program** dialog box appears.(Refer to 4.4.8.)
◆ If you finished define a program, it is inserted in the project as an item..
(In case of GM1 and there are more than two resources, you can select a resource it can introduce the definition of the program.)

5.2. Open a Program

You can open a saved program here. There are two ways to open an existing program. One is selecting in the menu and the other is opening the designated program in the project directly.

1) Open by selecting in the menu

◆ Select **Program-Open...**( ) in the menu.
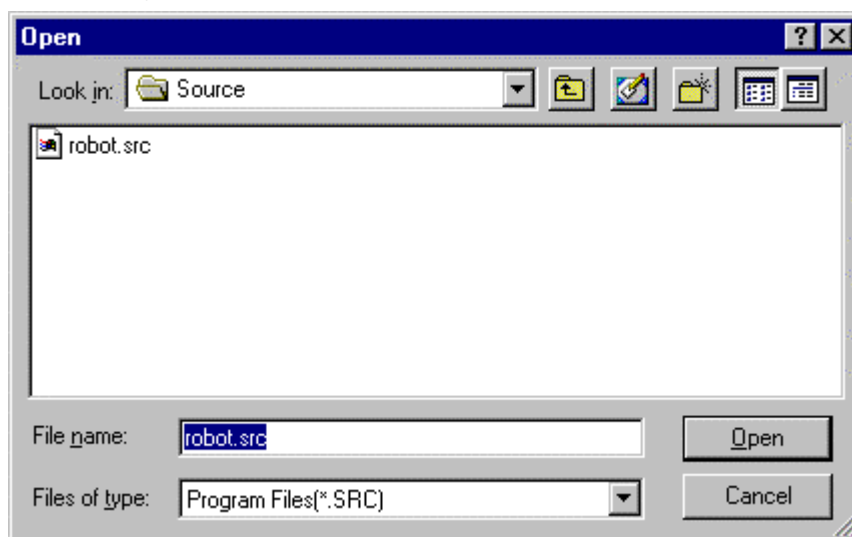
◆ Click **OK** button after selecting the desired program file.
◆ A selected program is created on the window screen.

2) Opening a program file included in Project

◆ If you want to open a program file included in Project, double-click a mouse on the appropriate program item in Project window list.
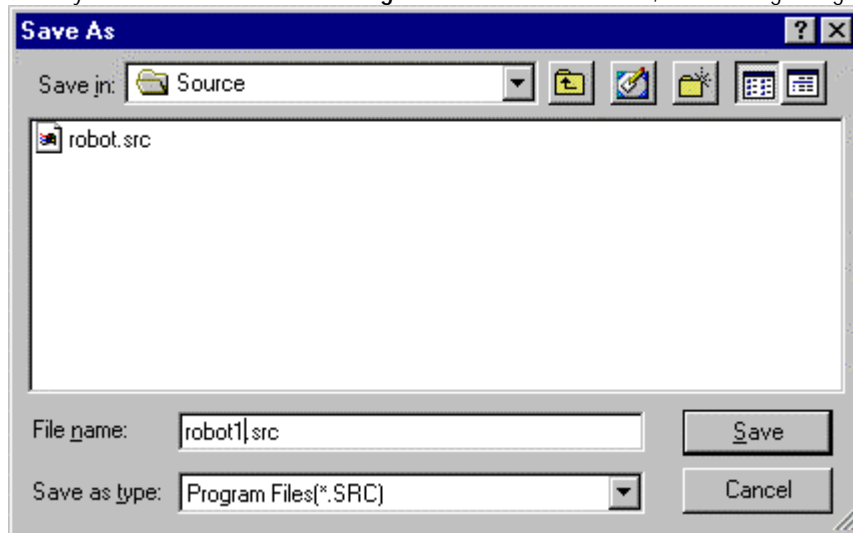◆ Then a selected program is created on the window screen.

> **Note**
>
> You can not open the same program more than two in Edit or Debug mode. In case of monitor mode, if you can not see a long program at the same time, please open the same program more than two and monitor a different part in each program.

5.3. Save a program

To save a program as file.

◆ Select Program-Save…([icon]) in the menu. In case of save an existing file, there is no message for saving.
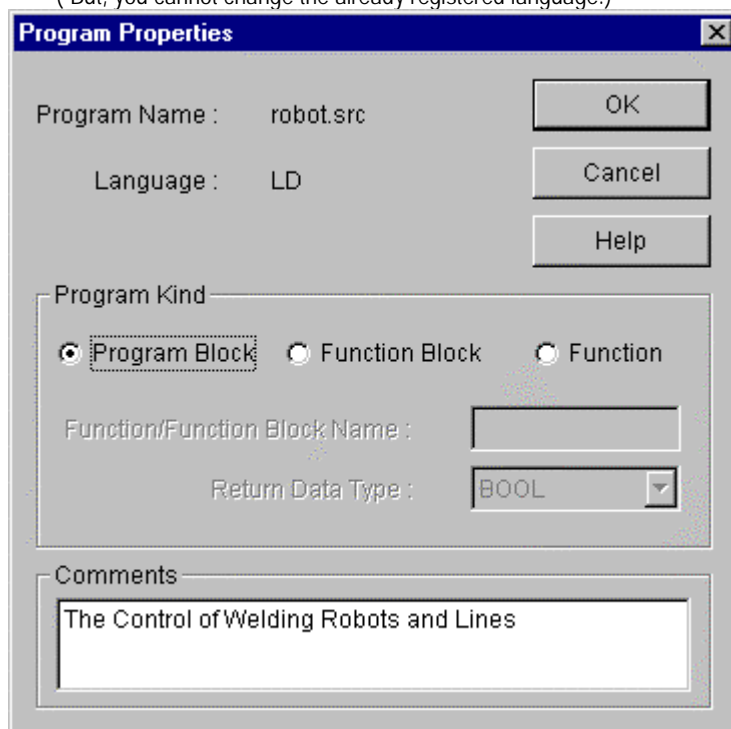◆ If you save a new file or select **Program-Save as…** in the menu, the following dialog box appears.

◆ Click **OK** button after entering the desired file name.
◆ If you want save all programs on the active window, select **Save All** in Menu.

5.4. Program Properties

You can change the kind or comments of program on editing.

◆ Select **Program-Properties…** in Menu. Then the dialog box appears as similar as the case of creating a program.
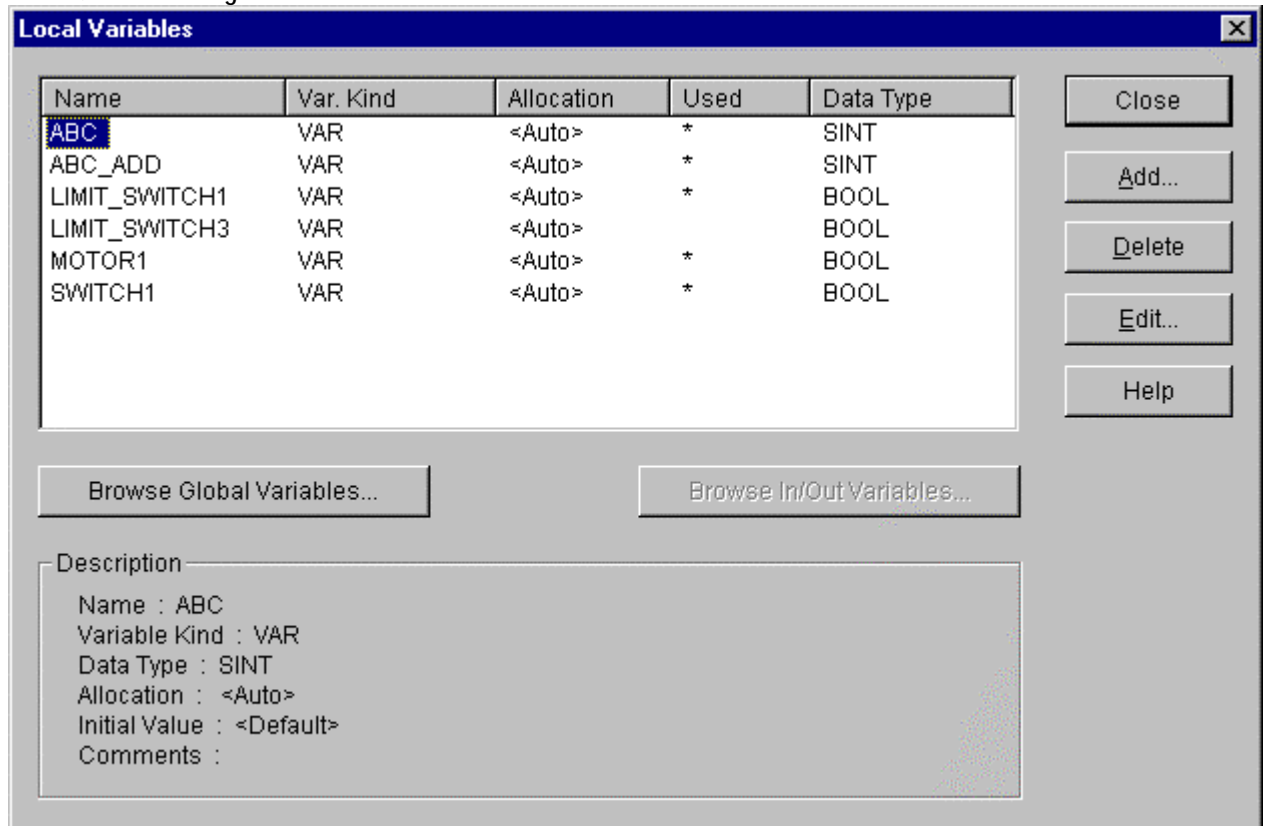(But, you cannot change the already registered language.)

◆ Click **OK** button after editing a desired item.

5.5. Edit a Variable

You can edit Local Variable it appropriates to the activate program window.

◆ Select **Program-Local Variables**....

◆ Click Close button after adding, deleting and editing each variable.

Note

In the list box of Local Variables, you can see another variables besides local variables that user enter. In case of function/function block, you can see I/O variable and output variable name to return and in case of SFC, you can see a variable 'TRANS' it contained executive conditions. But it is impossible to edit and delete these variables. Examples of these variables are as below.

RETVAL	\<Return variable\>	INT	- output variable name to be returned in function
IN1	VAR_INPUT	BOOL	- input variable in function/function block
IN2	VAR_INPUT	INT	- input variable in function/function block
OUT1	VAR_OUTPUT	BOOL	- output variable in function
TRANS	\<SFC transition\>	BOOL	- SFC execution condition variable

1) Add a Variable

◆ Open **Add/Edit Variables** dialog box by clicking **Add…** button.

◆ Enter the variable name in **Variable Name** box

Note

Maximum size for Variable name is 16 English characters.
However, in case of FB Instance type, the size limited to 8 English characters.

◆ Select Variable Kind in **Variable Kind** list box
◆ Select one type of **Data Type** option button (**Elementary Data Type/Function Block Instance**) and its type in that list box.
◆ Select one of **Memory Allocation** option button (**Auto** and **Assign(AT)**), and in case of selecting **Assign** enter the exact position in the text box.

> **Auto** : When you compile a program, declared variable is automatically assigned to
>   data area type
> **User Define** : User assigns the declared variable to I,Q,M area.

◆   In case of User Define, designation of I,Q,M area to input is as follows.

Prefix	Comments
I	Input
Q	Output
M	Internal Memory
X, none	1 bit
B	1 byte (8 bit)
W	1 word(16 bit)
D	2 word(32 bit)
L	4 word(64 bit)

Ex.) %QX3.1.4 or %Q3.1.4   : 4th bit of slot 1 on base 3.
   %IW2.4.2   : 2nd word of slot 4 on base 2.
   %MD48   : 48th double words of internal memory (start from 0).
   %MW10.3   : 3rd bit of 10th word in the internal memory.
                    (There is no general concept about base and slot in the
                    internal memory)

◆   If there is an initial value, enter the value in the box of **Initial Value**.
◆   Enter the comment in **Comments** box.
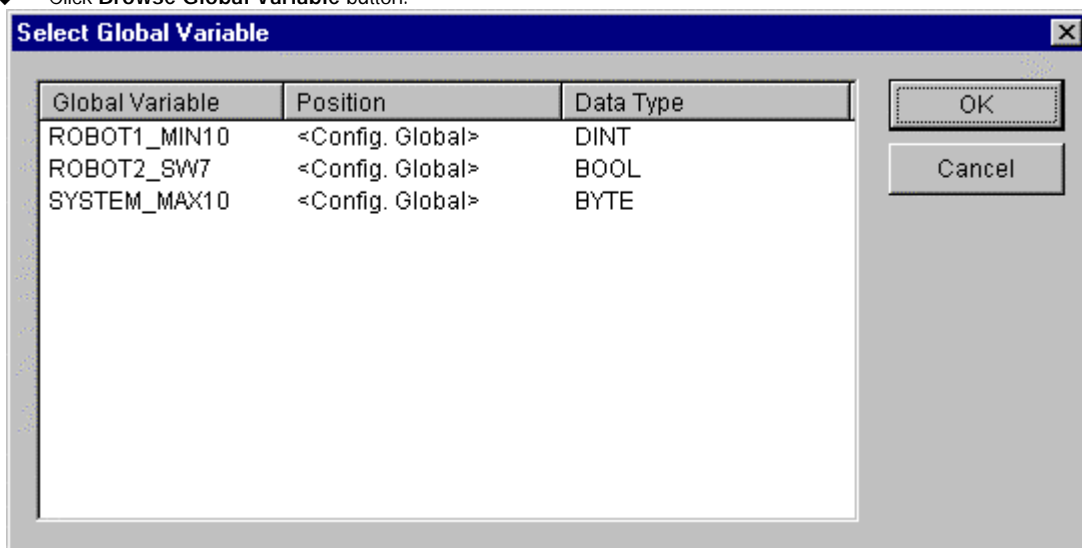◆   Click **OK** button.

2) Delete a Variable
   ◆   Select a variable to delete in the list box of **Local Variables**.
   ◆   Click **Delete** button

3) Edit a Variable
   ◆   Select a variable to edit in the list box of Local Variables.
   ◆   In the list box of **ADD/Edit variable**, edit a variable with the same way as inserting a variable and click **OK** button.

4) Browse Global Variable
   ◆   This function is used when you use a declared global variable in the program.
   ◆   Click **Browse Global Variable** button.

◆ Select a global variable to use in the program and click **OK** button.
◆ A selected global will be declared as **VAR_EXTERNAL** automatically and be added in **Local Variable List**.

Note

When you declare a variable, if the program belongs to the project, the variable declared as **VAR_EXTERNAL** is registered to **VAR_GLOBAL** automatically.

---

5) Browse In/Out variables
◆ This function is used when you want refer to the currently declared I/O variable without closing the **Local Variable** dialog box
◆ Click **Browse In/Out Variable** button. Then Dialog box appears as same as when you select **Program-I/O Variable** in the menu. You can add, edit or delete the I/O variable in the same way.


Note

Refer to the section 10.3.2 " Create I/O Variable" for " Edit I/O variable".

5.6. Edit LD

LD program creates a PLC program with a graphic symbol like coil or connects, which is used in relay logic diagram. In the below picture, ' **Rung Comment**' is the comment for the appropriate rung.
Rung is a unit of continuous line, which is connected vertically. I.e., in the below picture, a rung is from Row 1 to Row 4 and Row 5 is also a rung.
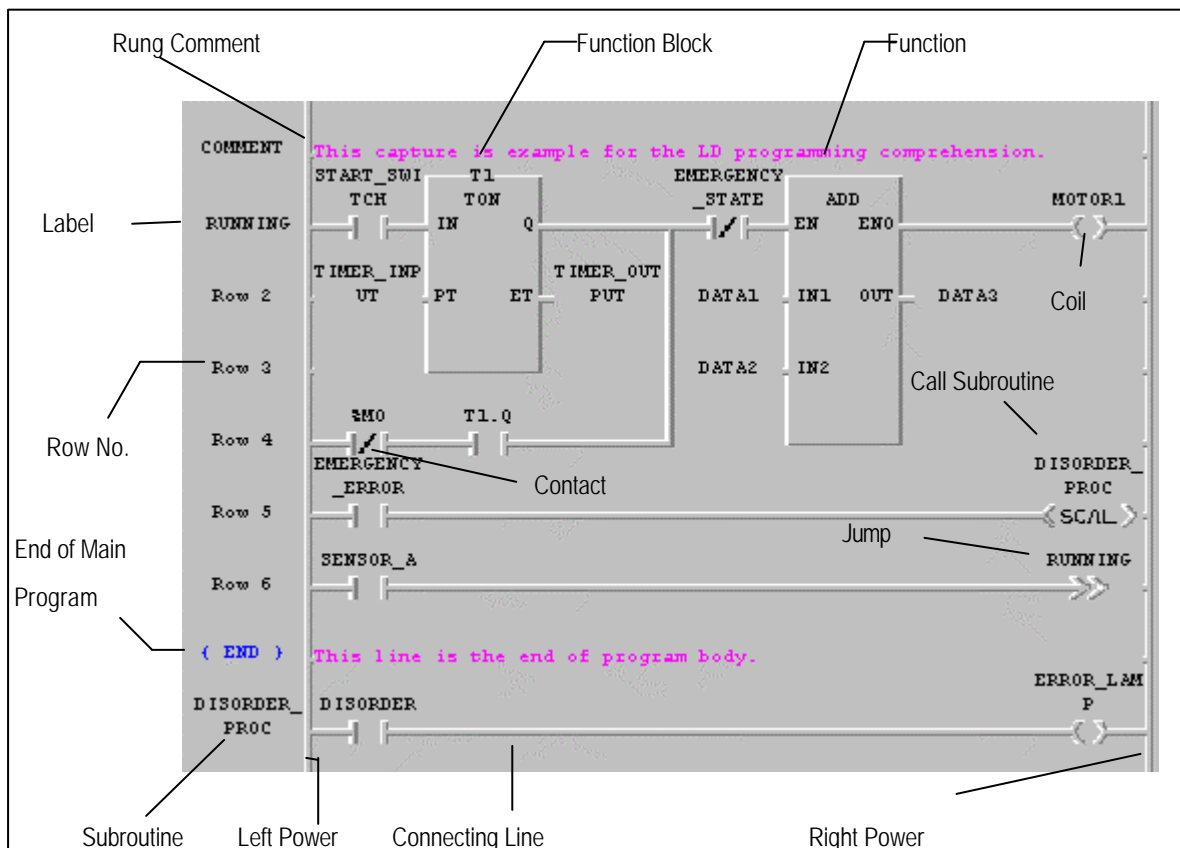
{ **END** } in Row 7 express the end of the main program. Error management, which is continued in the next column, is one type of sub-program and calls this subroutine in Row 1.

5-8

It is shows Tools and those names in the Toolbox.

Toolbox for ladder program
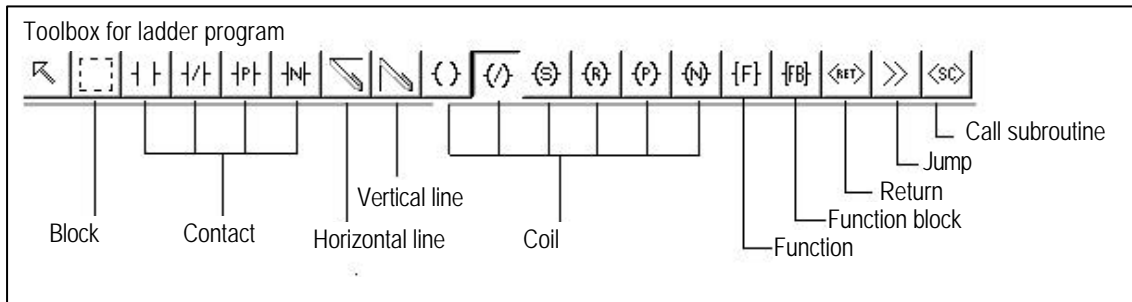
The shape and meaning of each element is as below.

Symbol	Shortcut Key	Comments
⊣ ⊢	F2	Normally open contact
⊣/⊢	F3	Normally closed contact
⊣P⊢	Shift-F1	Positive transition-sensing contact
⊣N⊢	Shift-F2	Negative transition-sensing contact
( )	F6	Coil
(/)	F7	Reverse coil
(S)	Shift-F3	Latched coil
(R)	Shift-F4	Unlatched coil
(P)	Shift-F5	Positive transition-sensing coil
(N)	Shift-F6	Negative transition-sensing coil
◣	F4	Horizontal line
◳	F5	Vertical line
{F}	F8	Function
{FB}	F9	Function Block
⟨RET⟩	Shift-F7	End command of LD program and Subroutine
≫	Shift-F8	Jump command for the branch of LD program(Label position)
⟨SC⟩	Shift-F9	Call a subroutine

If you select an arbitrary element in toolbox, the shape of the mouse is changed to that of the element.

5.6.1. Create a contact

Select a desired contact(⊣ ⊢ , ⊣/⊢ , ⊣P⊢ , ⊣N⊢ , ◿ , ◺)in toolbox.

◆ In LD program window, move the mouse to a desired position and click the left button.

◆ When you draw a line(◿ ◺), after moving the mouse from your desired position, drag a mouse by clicking the left button and you can draw a long line.

◆ When you draw a continuous line as the above way, if there is another contact under the mouse position, that contact is conserved.

◆ Move the cursor to the position that you want to input a contact in..

◆ Select a desired contact and line (F2, F3, Shft-F1, Shft-F2, F4, F5) of commands in **Toolbox**.

In case of contact, you can insert it between column 1 of program and column 30. In case of vertical line, you can insert it between column 2 and column 29.

5.6.2. Create a Coil

◆ Select a desired coil(( ) , (/) , (S) , (R) , (P) , (N))in **Toolbox**.

◆ In LD program window, move the mouse to a desired position and click the left button.

◆ Move a cursor to the position that you want to insert a coil in.

◆ Select a desired coil(F6, F7, Shft-F3, Shft-F4, Shft-F5, Shft-F6) of commands in **Toolbox**.

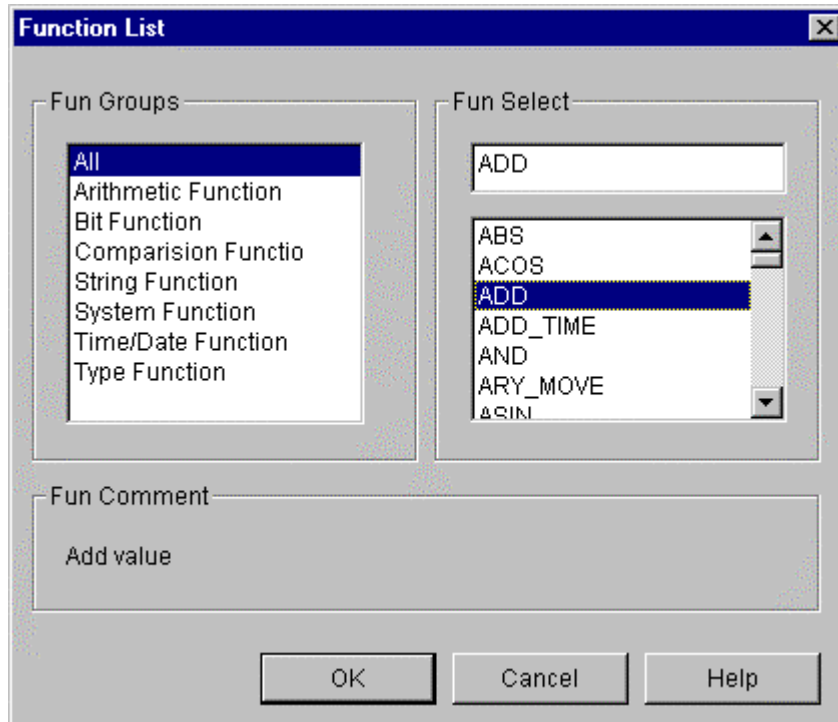Coil is inserted in the last (31st) column of LD program automatically.

5.6.3. Create a Function

◆ Select {F} in **Toolbox**.

◆ In LD program window, move the mouse to a desired position and click the left button.

◆ Move the cursor to the position that you want to a function in.

◆ Select a function (F8) of commands in Toolbox of the menu

Function List

Fun Groups

All
Arithmetic Function
Bit Function
Comparision Functio
String Function
System Function
Time/Date Function
Type Function

Fun Select

ADD

ABS
ACOS
ADD
ADD_TIME
AND
ARY_MOVE
ASIN

Fun Comment

Add value

OK   Cancel   Help

◆   Select a function group in **Function List** dialog box. Then only appropriate functions appear in the list.
◆   Select a desired function in **Function List** dialog box and click **OK** button.
◆   If the selected function is an Extensible Function, then the following dialog box appears automatically.

Number of Inputs

Maximum number of variable
inputs is :   8

How many variable inputs do you
want ?   2

OK

Cancel

Help

◆   Enter the desired input number
     You can enter a function between column 2 and column 30 in LD program.

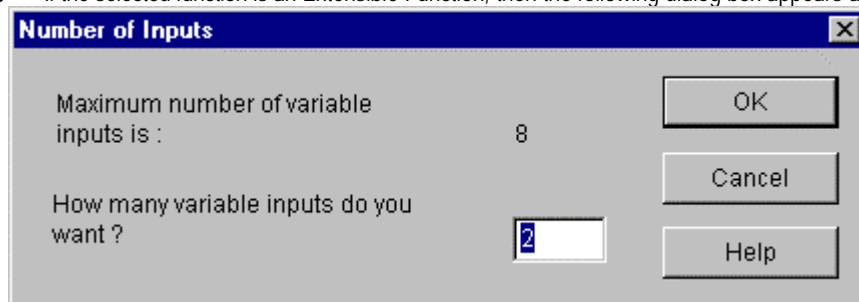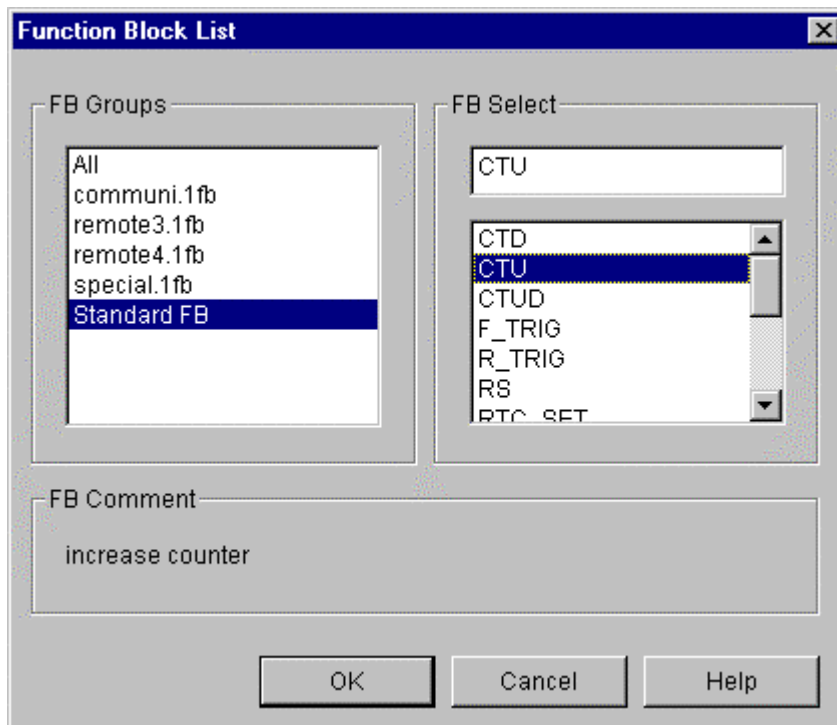### 5.6.4. Create a Function Block

◆   Select [FB] in **Toolbox**
◆   In LD program window, move the cursor to a desired position and click the left button.

◆   Move the cursor to the position that you want to insert a function in.
     Select a function block (F9) of commands in **Toolbox**.

◆ Select a function block group in **Function Block List** dialog box. Then only appropriate function blocks appear in the list.
◆ Select a desired function block in **Function Block List** dialog box and click **OK** button.
◆ Insert a desired instance name of function block in **Add Instance Name** dialog box.

You can insert a function block between column 2 and column 30 in LD program.

5.6.5. Return

◆ Select [<RET>] in Toolbox.
◆ In LD program window, move the mouse to a desired position and click the left button.

◆ Move the cursor to the position that you want to insert Return in..
◆ Select Return (Shft-F7) of commands in Toolbox.

Return is inserted in the last (31st) column of LD program.

5.6.6. Jump and Label

In LD program, Jump command allows you to go directly to the position that you want to branch out. Destination is expressed as a label. Position of label must be inserted on the column ' 0' . In LD program having subroutine, the label included in the main program must be inserted into jump on the main program and the jump in the subroutine area can branch out to the label in subroutine only.

1) Jump

◆ Select [≫] in **Toolbox**.
◆ In LD program window, click the left button after moving a mouse to the position that you want.

◆ In LD program window, move the cursor to the position that you want to insert a jump in.
◆ Select a jump (Shft-F8) of commands in **Toolbox**.

Jump is inserted in the last (31st) column of LD program.

2) Label

◆ Select [↖] in **Toolbox**.
◆ In LD program window, move the mouse to column ' 0' and double-click the left button.
◆ Select label in the **Label/Rung Comment/The End of Program Body** dialog box.
◆ Input a label name in Add Label dialog box. (Maximum 16 characters)

◆ In LD program window, move the cursor to column ' 0' .
◆ Press **Enter** key.
◆ Select Label in the Label/Rung Comment/The End of Program Body dialog box.
◆ Input a label name in **Add Label** dialog box.

When you designate a label in the jump command, input a label name on the position of jump command.
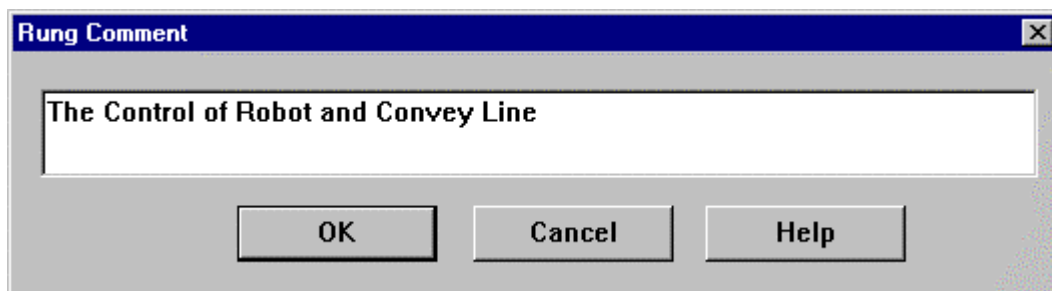Example)

5.6.7. Rung Comment

◆ Select [🔍] in **Toolbox**.
◆ In LD program window, move the mouse to column '0' of the first line in a desired rung and double-click the left button. But, if you want to edit and already existing rung comment move the mouse to that comment and double-click the left button on it.
◆ Select **Rung Comment** in dialog box.

Label or Rung Comment ☒

○ Label

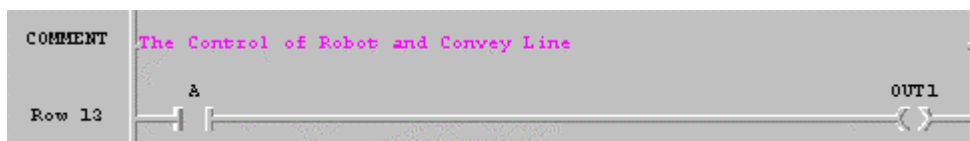⦿ Rung Comment

○ The End of Program Body

OK

Cancel

Help
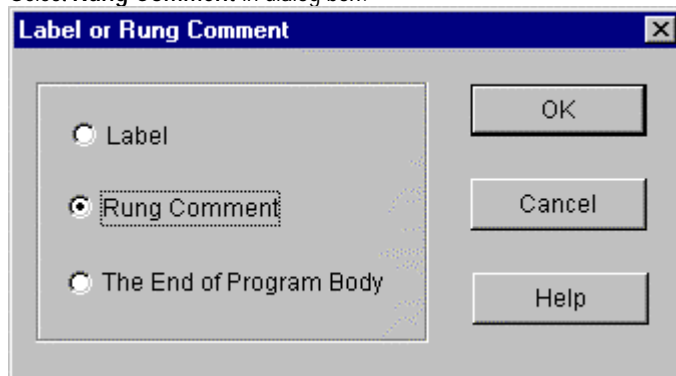
Enter the contents of rung comment in **Rung Comment** dialog box. (Maximum 170 characters)

◆ In LD program window, move the cursor to column '0' of the first line in desired rung. But, if you want to edit a created rung comment, move a cursor to that comment.
◆ Press **Enter** key.
◆ Select **Rung Comment** in **Label/Rung Comment** dialog box.
◆ Input a rung comment into the **Rung Comment** dialog box.

Rung Comment ☒

The Control of Robot and Convey Line

OK   Cancel   Help

Rung comment is displayed in all through on line as below.

```
COMMENT   The Control of Robot and Convey Line

                A                                      OUT1
Row 13     ──┤ ├──────────────────────────────────────( )──
```

5.6.8. Subroutine Call

◆ Select <img> in **Toolbox**.
◆ In LD program window, move the mouse to a desired position and click the left button.

◆ In LD program window, move the cursor to the position that you want to insert a subroutine name in.
◆ Select **Toolbox-Subroutine Call <SCAL>** (Shft-F9) in the menu.
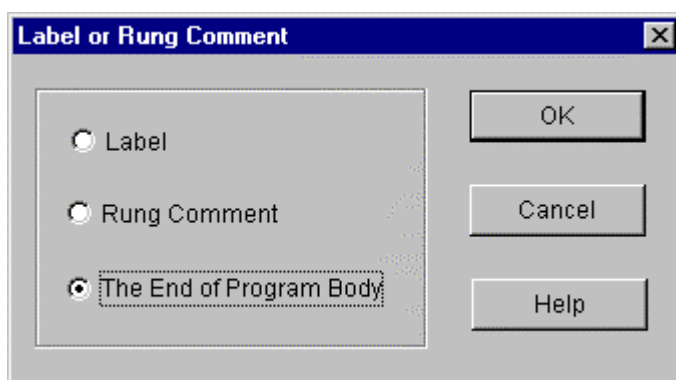
Insert a subroutine name on the **Subroutine Call**.
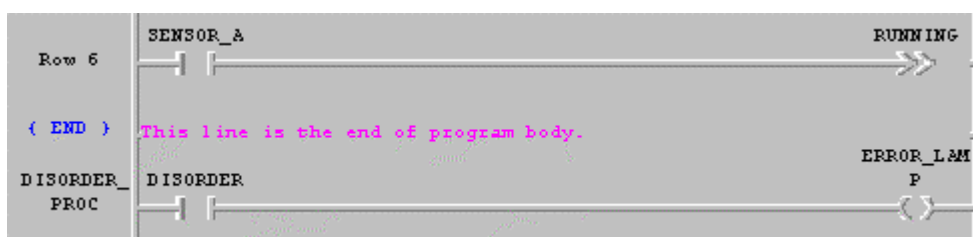
5.6.9. End of Main Program

◆ Select <img> in **Toolbox**.
◆ In LD program window, move the cursor to column ‘ 0’ of the row to designate as the end of main program.
◆ Select **The end of Program Body** in **Label** or **Rung Comment** dialog box.

◆ Move the cursor to column ‘ 0’ of the row to designate as the end of main program.
◆ Press **Enter** key.
◆ Select **The End of Program Body** in **Label or Rung Comment** dialog box.

In the end of main program, like below picture, **{END}** is displayed in column ‘ 0’ of the appropriate row and “ This line is the

end of main program” is displayed from column ‘ 1’ .

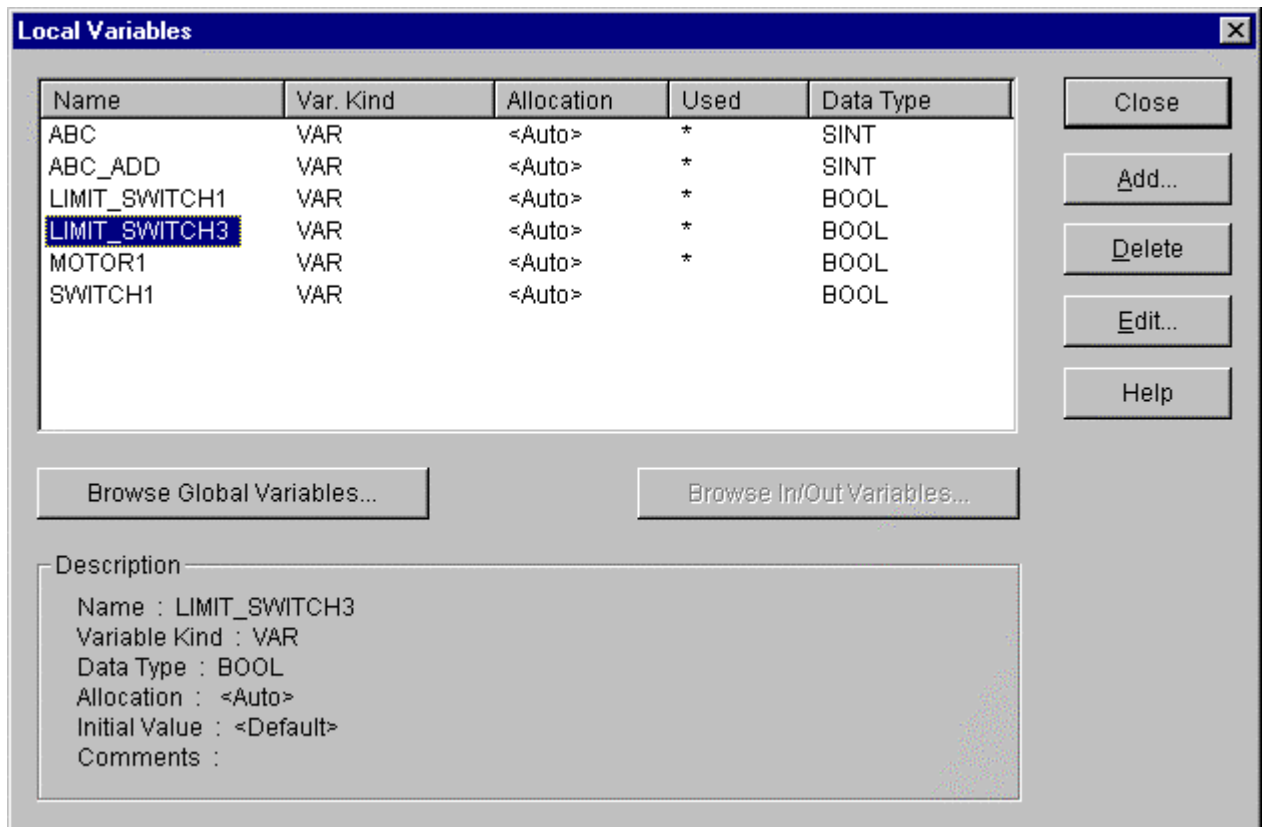Program logic under the end sign of the main program is regarded as subroutine area.
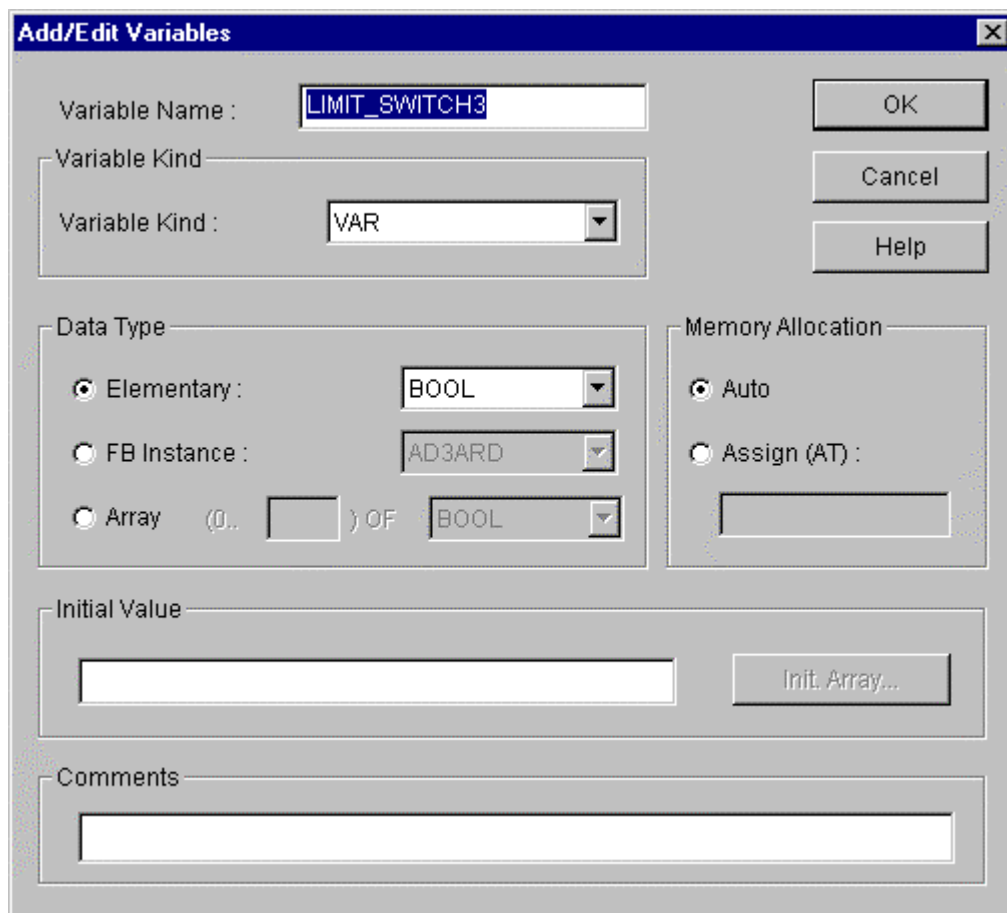
5.6.10. Variable Input

◆　Select ⬚ in toolbox.
◆　In LD program window, move the mouse on a desired contact, coil or I/O position of
　　Function block and double-click the left button.

◆　In LD program window, move the cursor a desired contact, coil or I/O position of
　　Function block and double-click the left button
◆　Call a variable dialogue box by pressing **Enter** key.

Local Variables ✕

Name	Var. Kind	Allocation	Used	Data Type
ABC	VAR	<Auto>	*	SINT
ABC_ADD	VAR	<Auto>	*	SINT
LIMIT_SWITCH1	VAR	<Auto>	*	BOOL
LIMIT_SWITCH3	VAR	<Auto>	*	BOOL
MOTOR1	VAR	<Auto>	*	BOOL
SWITCH1	VAR	<Auto>		BOOL

Close
Add...
Delete
Edit...
Help

Browse Global Variables...　　Browse In/Out Variables...

Description
　Name : LIMIT_SWITCH3
　Variable Kind : VAR
　Data Type : BOOL
　Allocation : <Auto>
　Initial Value : <Default>
　Comments :

◆　You can add, delete or edit a variable in **Variables** dialogue box.
◆　In **Variables** dialogue box, enter the variable name to be used and click **OK** button. If you enter an already declared variable name, then variable input is over here.
◆　If you enter a new variable name, **Add/Edit Variables** dialogue box, which determines the variable properties, appears.

◆ Refer to the '5.5 Edit a Variable' for detail information about **Add/Edit Variables** dialog box.
◆ Press **OK** button.

5.6.11. Edit a Block

Before editing a block, you must setup the range to be applied as a block.
Namely, if you want to copy or delete a continuous program, you must designate a range
desired s a block before selecting a block editing command.

◆ Select ⌑ in toolbox.
◆ In LD program window, after moving a mouse to the start point of a block by pressing the left button, drag the selected
range to the last row of the range to select.

◆ In LD program window, move the cursor to the start point of a block.
◆ With pressing **Shift** key, move the cursor with arrow key to the last row of range to select.

Note
When you setup a block, if the end of block range is in a function or function block,.
the function or function block is automatically included in the block.

1) Cut

After selecting a block,

◆ Select **Edit-Copy(Ctrl+X,** [icon] **)** .

2) Copy

After selecting a block,

◆ Select **Edit-Copy(Ctrl+C,** [icon] **)**.

3) Paste

After copying or cutting a block,

◆ In LD program window, move the cursor to the position that you want to paste on.

◆ Select **Edit-Paste(Ctrl+V,** [icon] **)**.
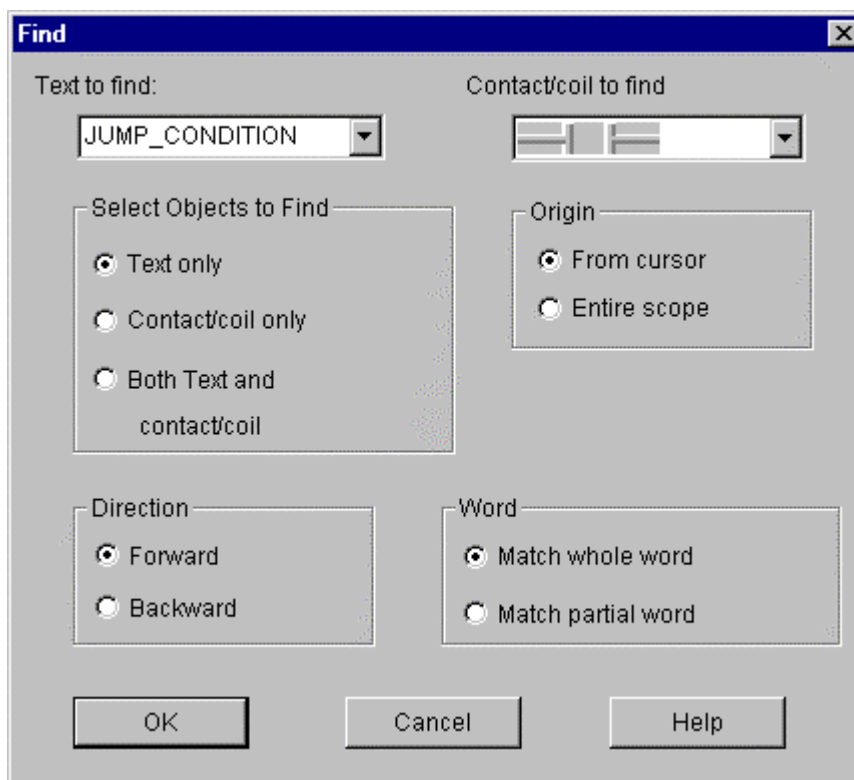
4) Delete

After selecting a block,

◆ Select **Edit-delete(Delete,** [icon] **)**.

5.6.12. Find

This command allows you to find a variable name, position of contact and etc, which are used in the program
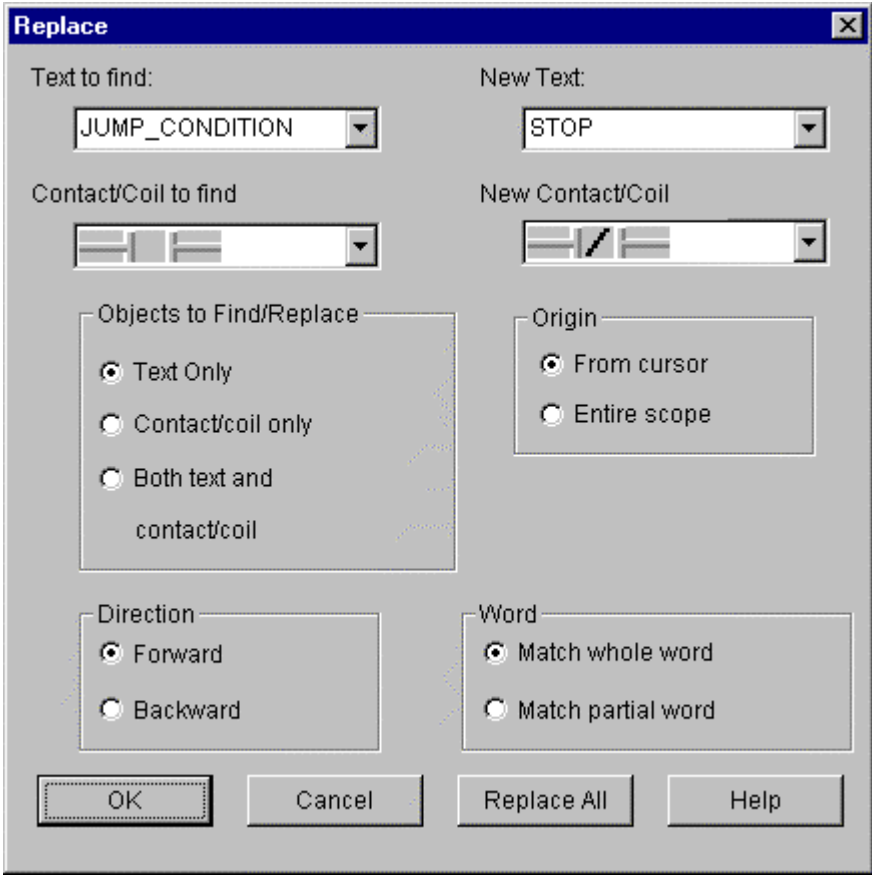
Call Find dialog box by selecting Edit-Find ([icon]).

◆   In **Text to find** edit box of **Find** dialog box, enter the string that you want
        to find.
◆   Select the following options to find and click **OK** button.

(1) Contact/Coil to find
    Select the contact or coil that you want to find.
(2) Select Objects to find
    Text only : Find only the content o character.
    Contact/Coil only : Find only LD program elements like contact, coil or etc.
    Both text and contact/Coil : Find a content of character and LD program elements
        like contact, coil or etc. simultaneously.
(3) Origin
    Select the range where find command is executed.
      From cursor : Find from current cursor position.
    Entire scope : Find through the whole program.
(4) Direction
    Select the direction that find command is executed.
    Down : Find command is executed downward.
    Up : Find command is executed upward
(5) Sensitivity
    Select an accordance degree in finding a content of character.
      Match whole word only : Find only an exactly corresponding word.
        Regular expression : Find also a partially corresponding word.

5.6.13. Replace

Call **replace** dialog box by selecting **Edit-Replace(**⬜**)**.

◆ Enter the character that you want to change in **Text to find** edit box.
◆ Select the element of LD program to change in **Contact/coil** list box.

◆ Enter the character that you want to replace in **New text** edit box.
◆ Select the element of LD program to replace in **New Contact/Coil** list box.
◆ Select the options in the same way of find command and click **OK** button.

◆ Click **OK** button.
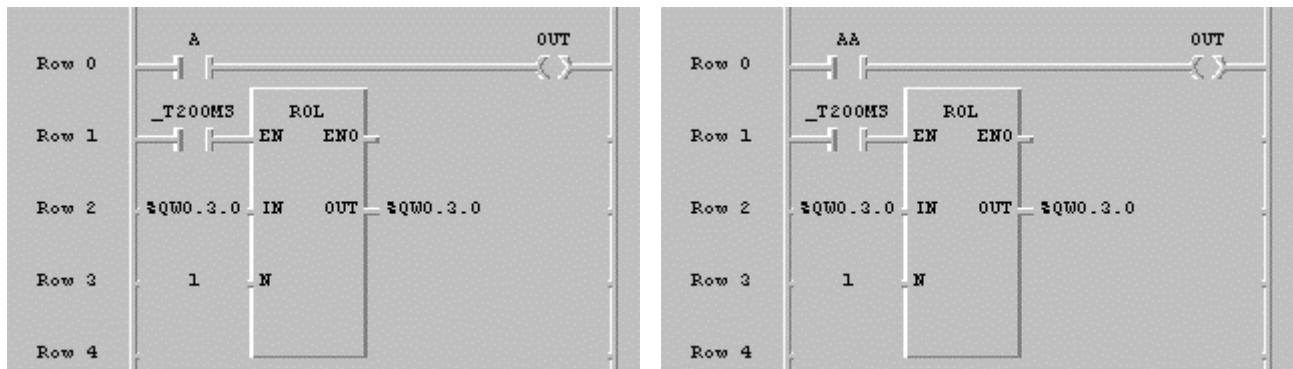
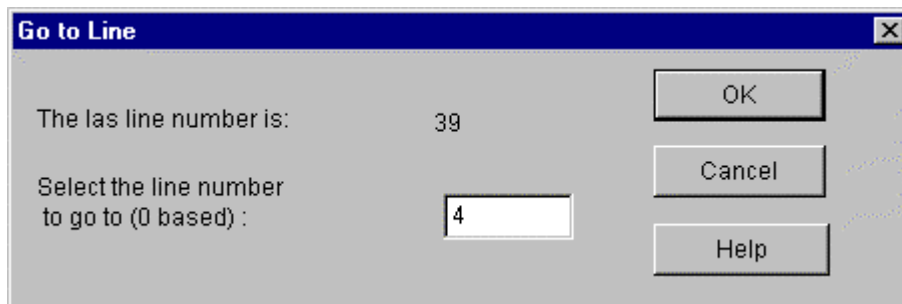Before   After

5.6.14. Find/ Replace Again

If you executed Find/Replace command before, execute that command again with the condition that designated before.
◆ Select **Edit-Again(Ctrl+F3)**.

5.6.15. Go to

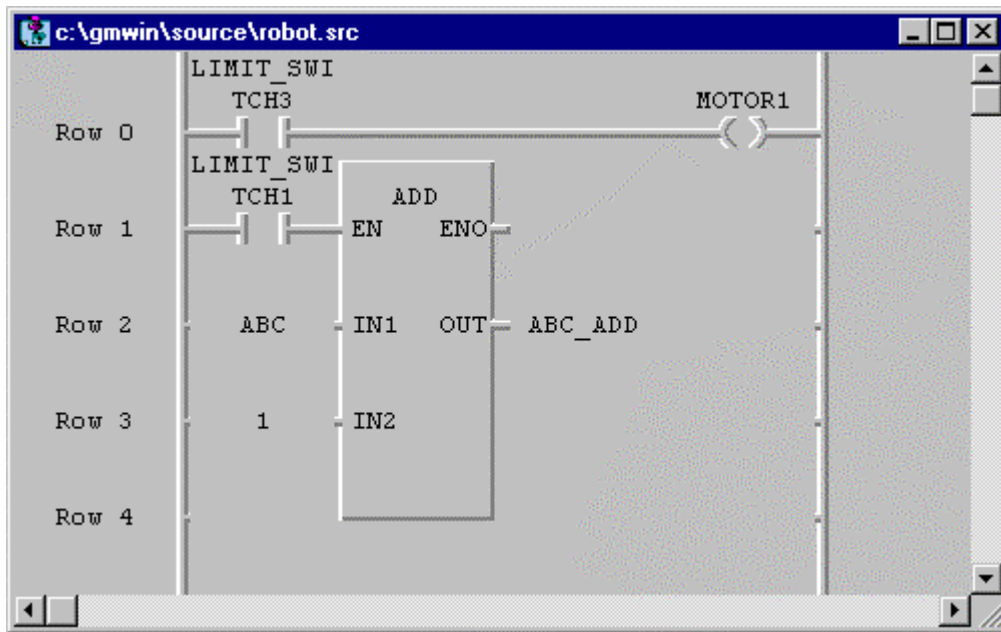◆ Select **Edit-Go to**.

◆ In **Go to line** dialog box, enter the row number that you want to find.
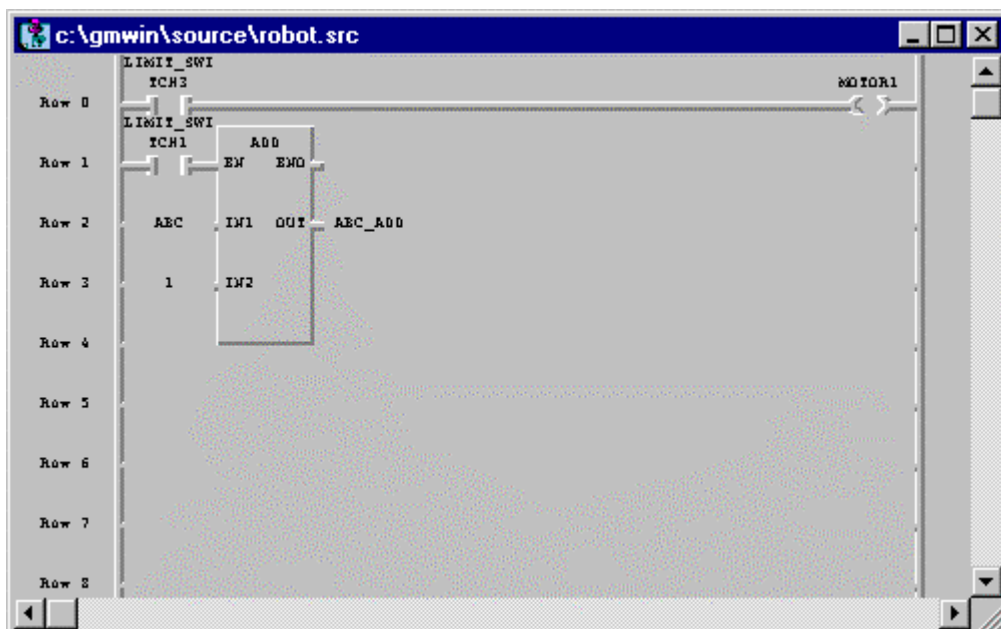
◆ Click **OK** button.

5.6.16. Zoom In/ Out

This command allows you to adjust the size of LD Program element.
◆　　　Call a dialog box by selecting **Edit-Condense**.
◆　　Select a desired screen size in **Condense Level** dialog box.
　　(100%, 80%, 70%, 50%, 25%, 12%).

Example) In case that compression rate is 100%

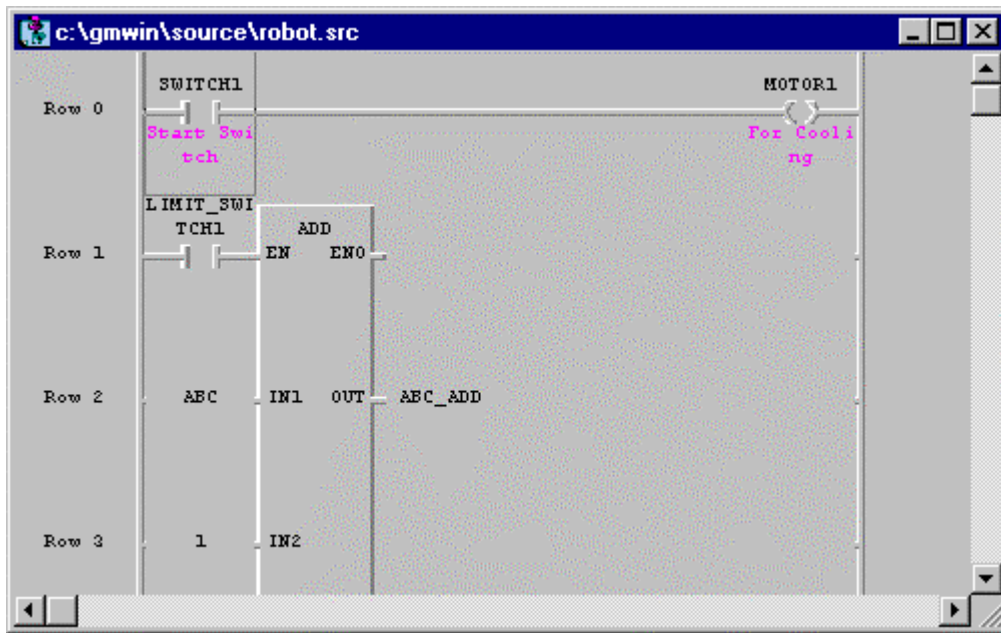In case that compression rate is 100%

Note

In case that compression rate is 25%, 12%, LD program is not displayed as 3D type.

5.6.17. Variable Comment

LD program supports the function that indicates the comment of each variable.
◆   Select **Edit-Variable comment (Ctrl+M)** in the pull-down menu.
◆   If there is a check mark at the left side of **Edit-Variable comment (Ctrl+M)**,
      a current window is in the state that shows a variable comment and if there is
      not, the window is in the state that does not show the comment.

Example) In case that express a variable comment

5.6.18. Delete Row

◆   Select ( ) in toolbox.
◆   In LD program window, move the mouse to the row that you want to delete and move the cursor with clicking a left
      button.
◆   Select **Edit-Delete Line**.

◆   In LD program window, move the cursor to the row that you wish to delete.
◆   Select **Edit-Delete Line (Ctrl+D)**.

5.6.19. Delete Cell

Delete the cell that cursor is on.

If the current position of the cursor is empty space, this function deletes the empty cell by one column.

◆   Select ( ) in toolbox.
◆   In LD program window, move the mouse to the cell you want to delete and move the cursor with clicking a left button.
◆   Select **Edit-Delete Cell**.

◆ In LD program window, move a cursor to the cell to be deleted.
◆ Select **Edit-Delete Cell (Delete)**.

5.6.20. Insert Line

◆ Select (⬉) in toolbox.
◆ In LD program window, move the mouse to the position that you want to insert the row and move the dursor with clicking a left button.
◆ Select **Edit-Insert Line(Ctrl+N)**.

◆ In LD program window, move the mouse to the position that you want to enter the row.
◆ Select **Edit-Insert Line(Ctrl+N)**.

5.6.21. Insert Cell

Insert an empty cell in the position where a cursor is on.

◆ Select (⬉) in toolbox.
◆ In LD program window, move the mouse to the position where you want to insert a cell and clicking a left button.
◆ Select **Edit-Insert Cell(Ctrl+I)**.

◆ In LD program window, move the cursor to the position where you want to insert the cell.
◆ Select **Edit-Insert Cell(Ctrl+I)**.

5.6.22. Move/Copy Cell

(for copy, mouse and keyboard are used simultaneously)
◆ Select (⬉) in toolbox.
◆ In LD program window, move the mouse to the cell to move/copy.
◆ Drag to a desired position with clicking a left button of mouse.
◆ In case of moving, release the ldft button of the mouse.
In case of copy, release a left button of mouse with pressing **CTRL** key.

5.6.23. Pop-Up Menu

◆ Click a right button of the mouse on an arbitrary position or desired position.

◆　Select a desired command in menu that is appeared in right side.
Pop-Up Menu are various according to the position of the mouse. You can select frequently used functions in Pop-Up Menu.

To Arrow mode	Ctrl+A
Undo (Variable Edit)	Ctrl+Z
Cut	Ctrl+X
Copy	Ctrl+C
Paste	Ctrl+V
Delete	Del
Find...	Ctrl+F
Replace...	Ctrl+H
Again	Ctrl+F3
Goto...	
Condense...	Ctrl+E
Variable comment...	Ctrl+M
Delete line	Ctrl+D
Delete cell(H)	Del
Insert Line	Ctrl+L
Insert cell	Ctrl+I

If you use a Pop-up menu, then you don't need to use each command through the pull-down menu.
In the upper menu, To Arrow Mode is the command that changes the input state of a current
mouse into arrow mode to select or enter the data or elements of the program.

5.6.24. Undo

This command allows you to cancel previously edited action.
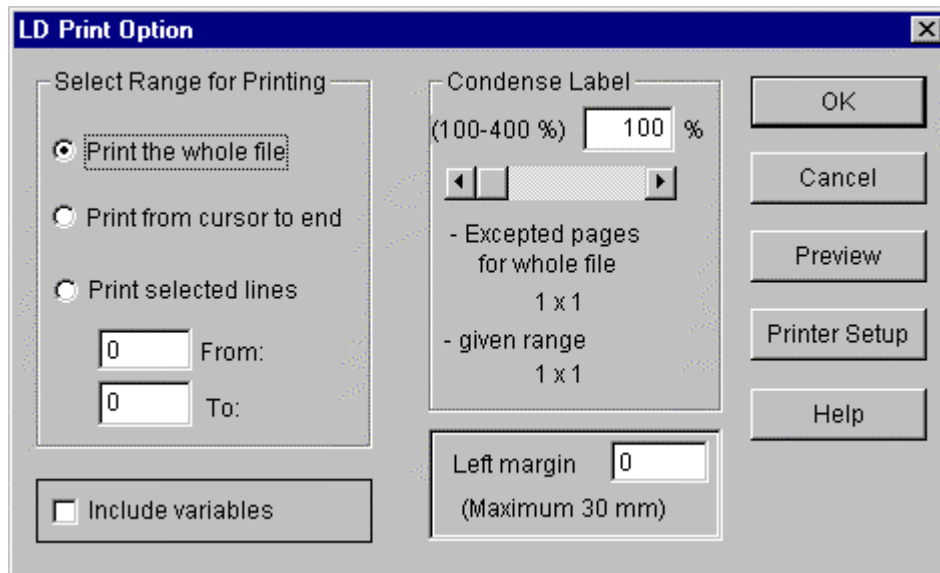
◆　Select **Undo(Ctrl+Z** ; ⟳ **)**.

◆　Select **Undo(Ctrl+Z)**.

5.6.25. Print and Preview   5-26

Print
◆   Call a LD Print Option dialog box by selecting **Project-Print**.

◆   In **Select Range for Printing** group box of **LD Print Option** dialog box, select the range of LD program to print.
◆   In **Left Margin** group box, enter the blank in mm.(Maximum size is 30mm).
◆   In **Condense Label** group box, enter the compression rate of content to print within the range that is from 100% to 400%. Indicates the required page number of printing the whole file and the required page number of printing the given range that user designate in **Select Range for Printing.**

Preview

◆   **Preview** button of **LD Print Option** dialog box is used when you want to see the
    outline of contents to print before printing actually on paper.
    In **Preview** dialog box, you can determine either showing several pages
    simultaneously in a screen or by one page.