

Chapter 9.Debugging

The Debugging detects an error and removes it, so that the PLC program created by user may be operated normally, the ways finding a program error are following as.

1) Step Run

Detecting an error with running the program step by step.

2) Break Run

Detecting an error after stopping the program on a specified step with break.

3) Data Break Run

Detecting an error with break on the access conditions of the specified scan account.

4) Bread Run with Specifying Scan Account

Detecting an error after making the program in execution as much as specified scan account.

In Step Run mode, it is impossible to access the program with task, but in Break Run mode, you can access the task program. Maximum number of break point is 32 points for **Program Break** and only 1 point for **Data Break**.

If the program for debugging is opened, Step Run starts from the first line of its program. If not, it starts from the first line of the first scan program assigned in project. Step run ends at the last line of the program. Break point is displayed in red color in each program, and in blue for the execution of break to the point.

If you change to debugging mode, program monitoring starts automatically.

Note

Change to debugging mode is available only in Stop mode.

In case of disconnected with PLC

- ◆ Select **Online-Connect** in menu.
- ◆ Select **Debug-Debug Start** in menu.
- ◆ Or select **Online-Mode Change-Debug** in menu.

In case of connected with PLC

- ◆ Select **Debug-Debug Start** in menu.
- ◆ or select **Online-Mode Change-Debug** in menu.

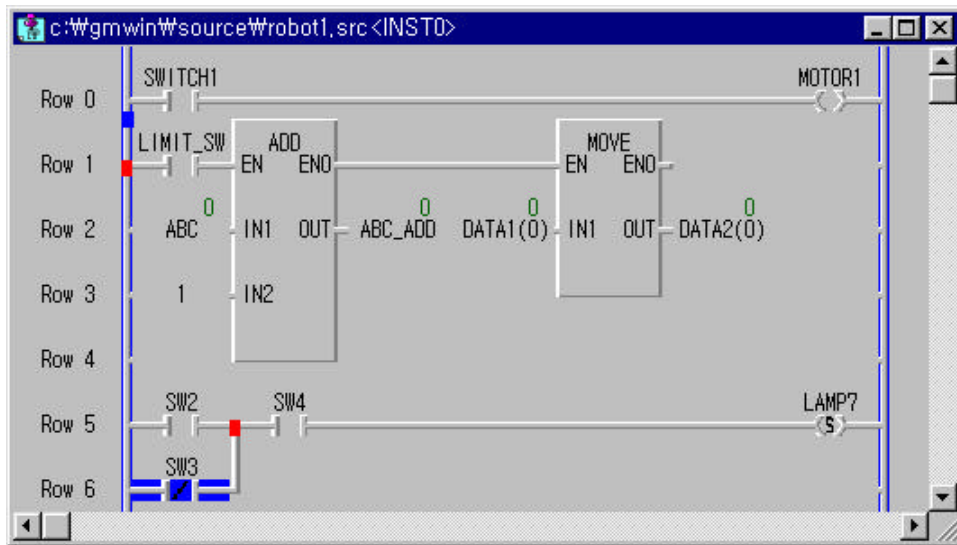
9.1. LD Debugging

9.1.1. Break Run

1) Select Break Run

- ◆ Move the cursor to the position that you want to assign as break point.

- ◆ Select the **Debug-Add/Remove Break Point**(Ctrl+F5) in the pull-down menu.



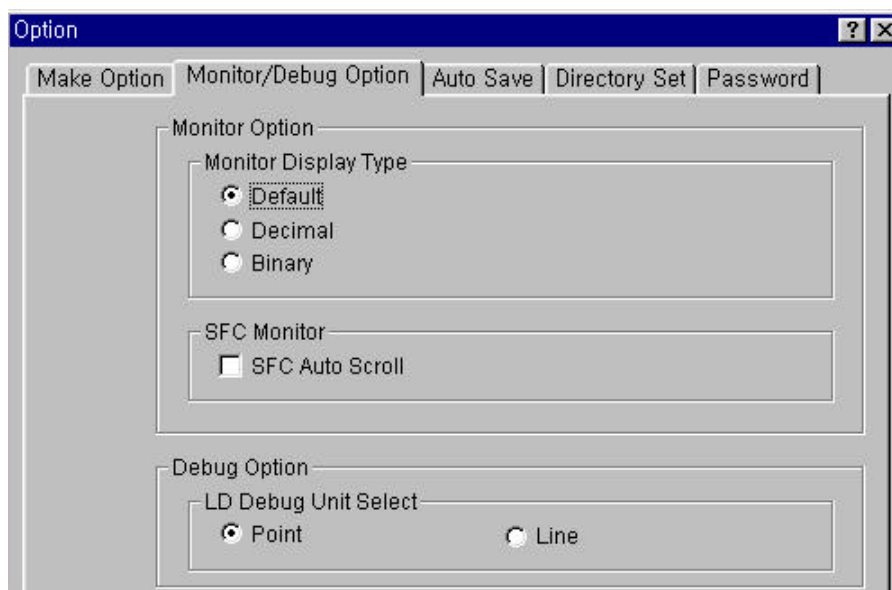
- ◆ Select the **Debug-Go**(Ctrl+F9) in menu.
As the above figure, a red square is displayed on the lower end of the left side of the position of assigned a break point and a blue square is displayed on the upper part of the left side of the position of break is on.
Then PLC program stops at the line set as break point(row 5 in above figure)

2) Run to cursor position

- ◆ Move the cursor to the position that you want to assign as break point
- ◆ Select the **Debug-Go to Cursor**(Ctrl+F2) in menu.

9.1.2. Step Run

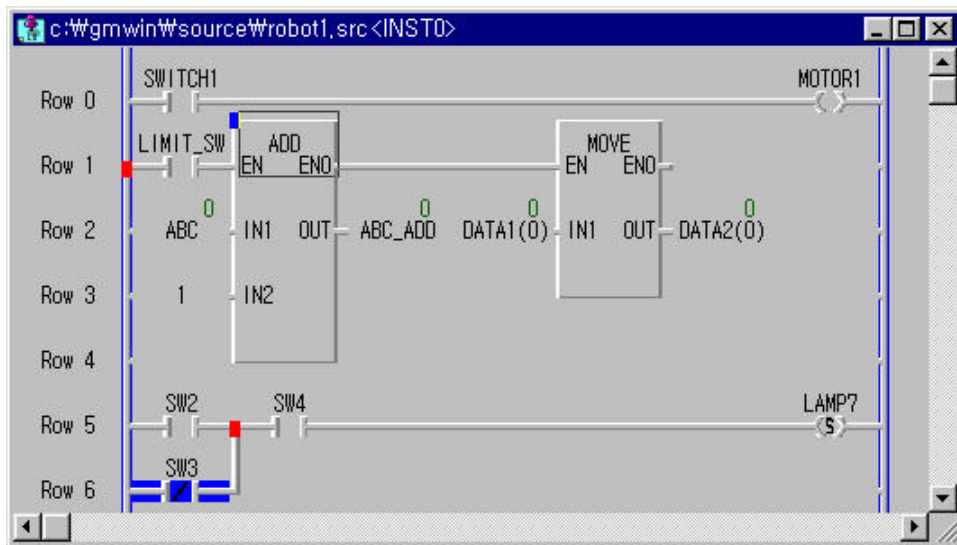
You can use two types of **Step over** in LD debugging. One is step over by Point unit, the other one is step over by Line unit. It can be chosen at the **Monitor/Debug Option** folder in Option dialog box of **Project** menu..



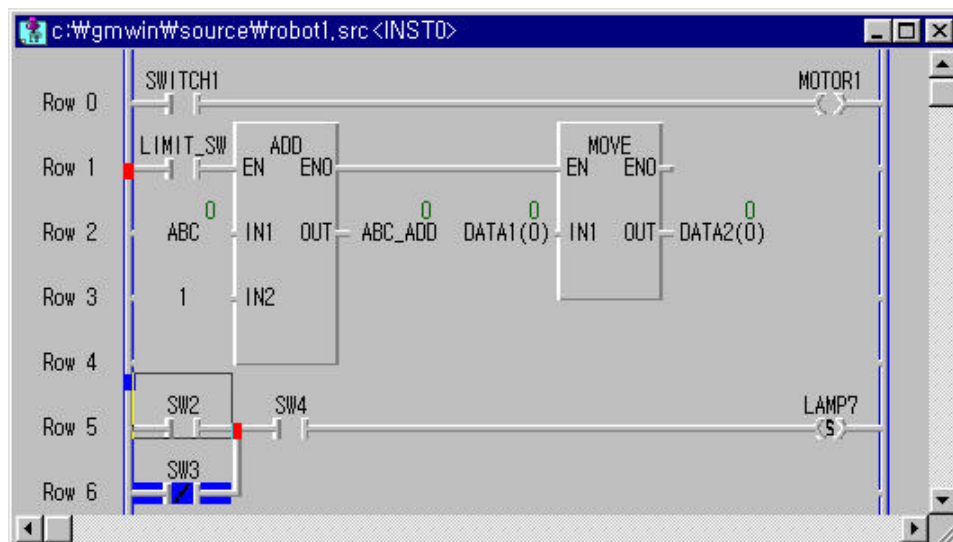
- ◆ Select the **Debug- Step Over**(Ctrl+F8) in menu.

Chapter 9. Debugging

When you run Step Over by **Point unit** for the program in 9.1.1 Break Run, the result is as below.



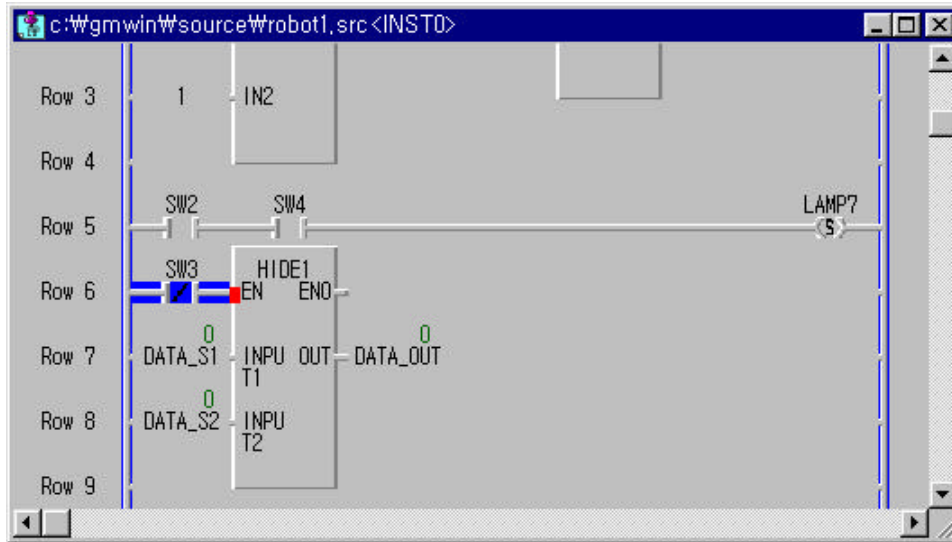
When you run Step Over by **Line unit** for the program in 9.1.1 Break Run, the result is as below .



9.1.3. Step Into

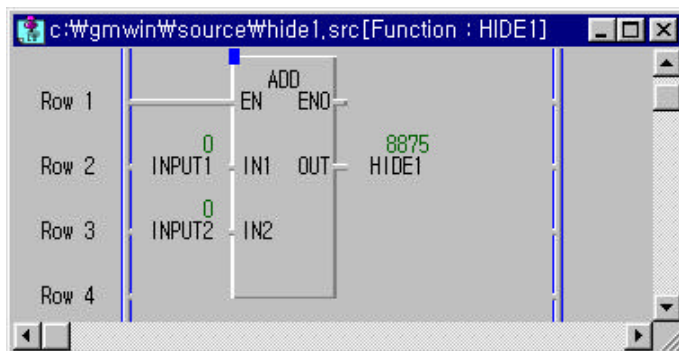
In case of Break is executed on the **User Define Function/Function Block**, you can debug a **User Define Function/Function Block** using **Step Into** function.

- ◆ Add the Break point for debugging on the User Defined Function/Function Block.



HIDDEN 1 is **User Define Function** on the above figure.

- ◆ Select the **Debug-Step Into**(Ctrl+F7) in menu.



If the source program of User Defined Function/Function Block exists, you can monitor/debug it by **Step into** function.

9.1.4. Step Out

To return to the program block from the **User defined Function/Function Block** after debugging.

- ◆ Select **Debug-Step Out** in menu.

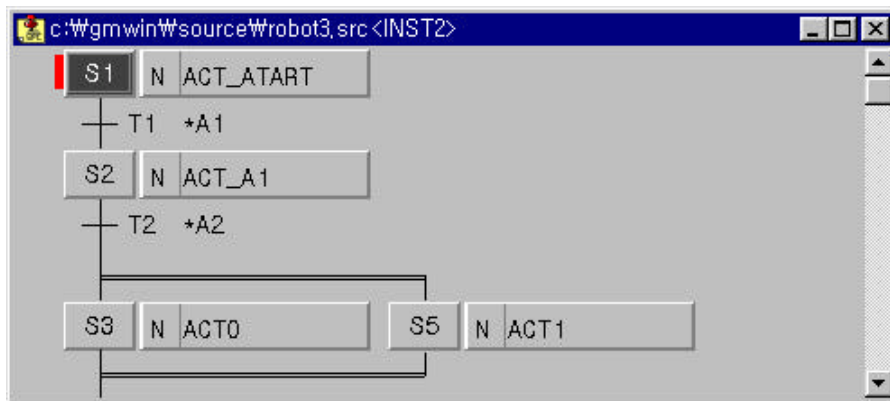
9.2. SFC Debugging

9.2.1. Break Run

In SFC, you can execute a **Break Run** function for only Step.

1) Set a Break Point

- ◆ Move the cursor to the Step that you want to assign as Break Point.
- ◆ Select the **Debug-Add/Remove break Point(Ctrl+F5)** in menu.



- ◆ Select the **Debug-Go(Ctrl+F9)** in menu.
Then **ACTION1** is executed, after **Trans1** condition is satisfied, at the moment **Step2** set as Break Point is in active.

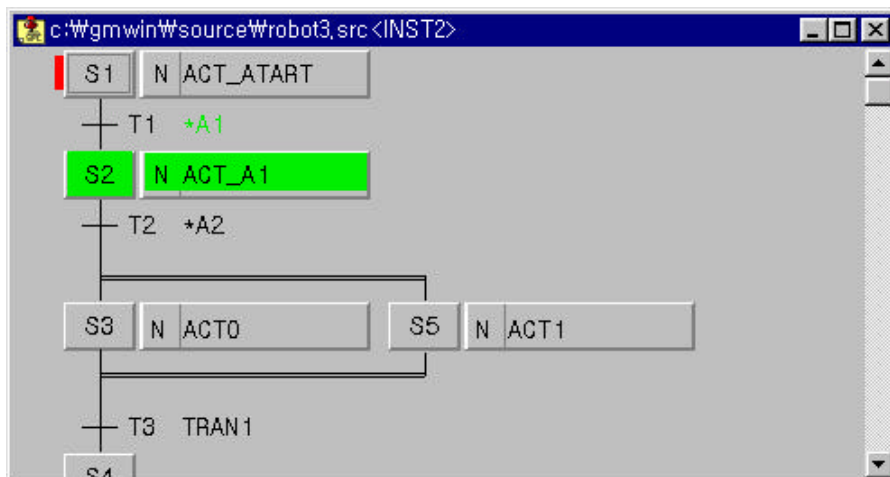
2) Run to cursor position

- ◆ Move the cursor to the step that you want to set as Break Point
- ◆ Select the **Debug-Go to Cursor(Ctrl+F2)** in menu.
At the moment Step that cursor is placed on is in active, PLC is stopped immediately.

9.2.2. Step Run

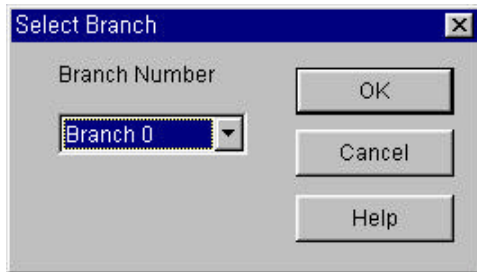
It executes a program by one step(S1, S2, S3).

- ◆ Select the **Debug-Step Over(Ctrl+F8)** in menu.



Step S3 connected with Step S2 on break is executed when SEL1 transition condition is satisfied. At the moment Step S3 is activated, PLC is stopped immediately. If you execute Step Run here, ACTION3 connected with Step S3 is executed when TRANS3 condition is satisfied. Stop it with the next connected step is activated.

When you select **Step Run**, if the next step es a parallel branch, then you should select the branch to execute **Step Run**.



- ◆ Select a branch number(select among the 1,2 and 3 in the above figure) in the **Branch Select** dialog box, and Click **OK** button.

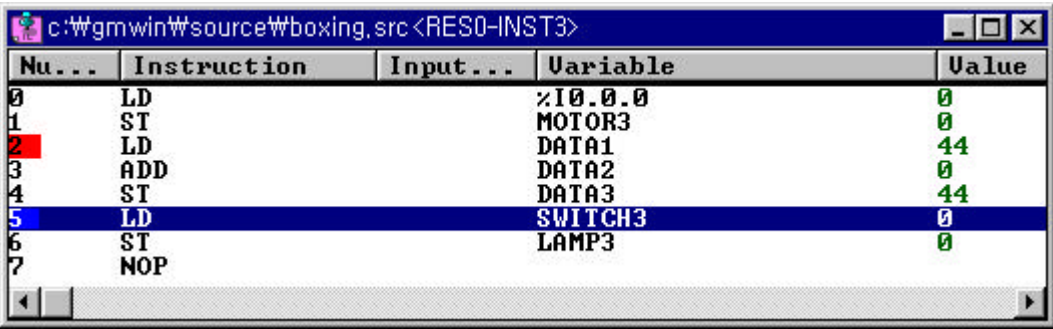
9.2.3. Step Into

When Step is under break condition, if you select **Step Into**, a break occurs at the first line of connected action.

9.2.4. Action Debugging

It allows you to debug the program by **Zoom** function for the action program.
It is the same as IL, LD debugging.

9.3. IL Debugging



Nu...	Instruction	Input...	Variable	Uvalue
0	LD		%I0.0.0	0
1	ST		MOTOR3	0
2	LD		DATA1	44
3	ADD		DATA2	0
4	ST		DATA3	44
5	LD		SWITCH3	0
6	ST		LAMP3	0
7	NOP			

9.3.1. Break Run

To execute a program break, you can use 2 ways as below.

1) Set a Break Point

- ◆ Move the cursor to the line that you want to set as a break.
- ◆ Select the **Debug-Add/Remove Break P0int(Ctrl+F5)** in menu.
- ◆ Select the **Debug-Go(Ctrl+F9)** in menu.

2) Run to Cursor Position

- ◆ Move the cursor to the line that you want to set as a break.
- ◆ Select the **Debug-Run to Cursor(Ctrl+F2)** in menu.

9.3.2. Step Run

It runs a program by one step.

- ◆ Select the **Debug-Step Over(Ctrl+F8)** in menu.

9.3.3. Step Into

It debugs a program in the Function or Function Block.

- ◆ Select the **Debug-Step Into(Ctrl+F7)** in menu.

9.3.4. Step Out

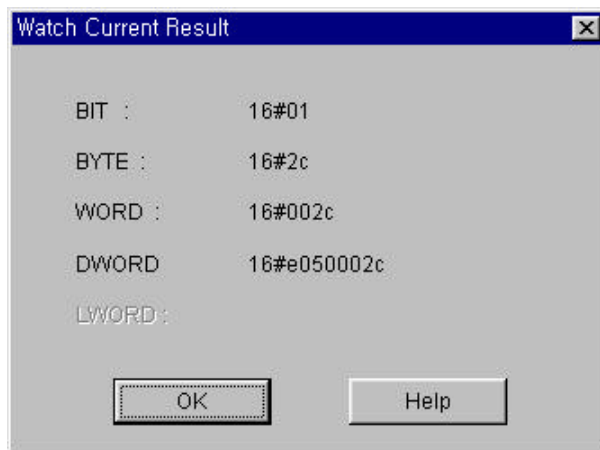
It returns from Function or Function Block to original program.

- ◆ Select the **Debug-Step Out** in menu.

9.3.5. Watch Current Result

It allows you to view the current result in the program that is under break.

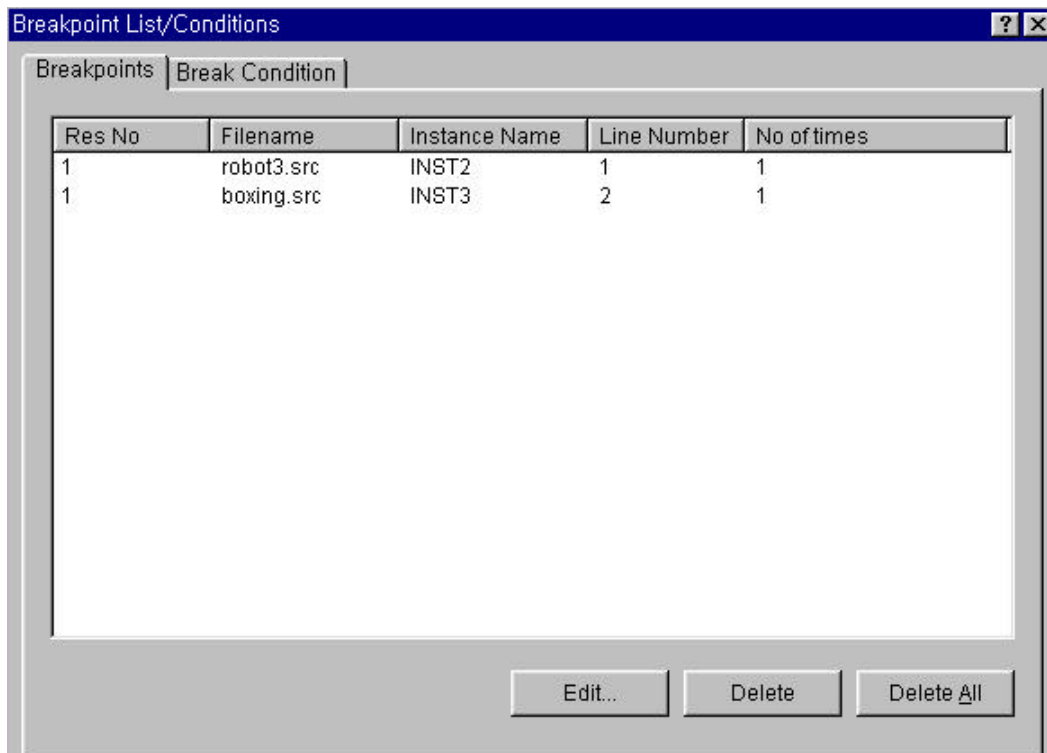
- ◆ Select the **Toolbox-Watch Current Result** in menu..



9.4. View Break Point

You can view and delete the break points in the Break Point List/Condition dialog box. GMWIN saves the break point by Project unit. That is, when you reload a project, the break points of the project are same as you set previously. If you delete a break point in the Break Point List/Condition dialog box, the break point set in the program is deleted.

- ◆ Select the **Debug-Break Point List/Condition** in menu.
- ◆ Select the **Break Points** Tab.



- ◆ You can view and delete the entire break points. Moreover you can set the number of break count.
- ◆ Click **OK** button.

1) Delete a Break Point

- ◆ Select the break point to delete in dialog box.
- ◆ Click **Delete** button.

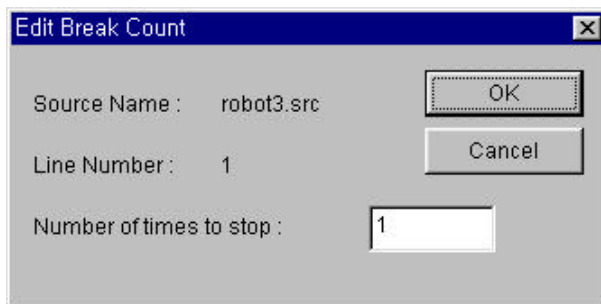
2) Delete the entire Break Point

- ◆ Click **Delete All** button.

3) Edit the number of Break

After executing the breaks as many as you set the number of break count, the PLC is stopped.

- ◆ Select the break point to edit the number of break count in dialog box.
- ◆ Click Edit.. button.

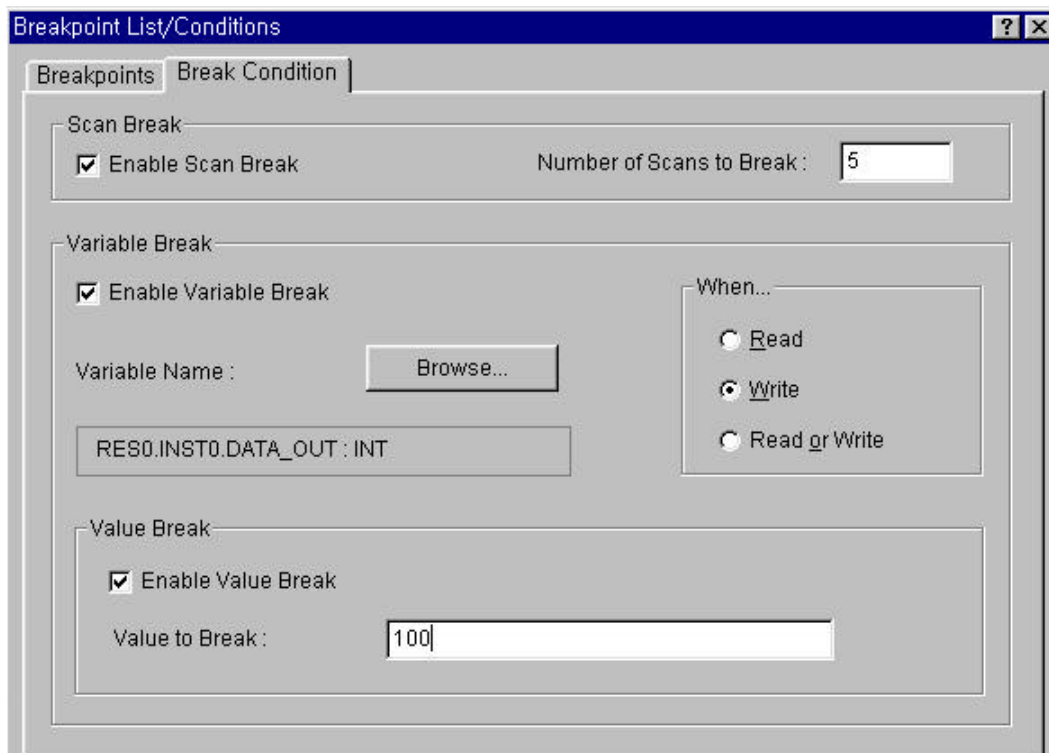


- ◆ Enter the number of Break count that want to break, and click **OK** button.

9.5. Set a Break Condition

In **Break Condition** dialog box, you can set the break condition, which is the condition to break by the state of assigned variable or number of the scan times.

- ◆ Select the **Debug-Break Point List/Condition...** in menu.
- ◆ Select the **Break Condition** Tab.



9.5.1. Break Scan

It stops the program after PLC runs as many as assigned number of scan times.

- ◆ Select the **Enable Scan Break** option check box.
- ◆ Enter the number at **Number of Scan to Break** input box.

9.5.2. Break Variable

If you read a variable or write an arbitrary value to assigned variable, PLC is stopped.

- ◆ Select the **Enable Variable Break** option check box.
- ◆ Select **Browse** button.
- ◆ Select a variable in **Register Variable** dialog box, click **OK** button.

The **Register Variable** dialog box contains the following elements:

- Kind** section with four radio buttons:
 - ☐ Configuration Global Variable
 - ☐ Resource Global Variable
 - ☒ Instance Variable
 - ☐ Direct Variable (with an adjacent text box containing "Ex) %IX0.0.0")
- Resource** dropdown menu showing "Resource0".
- Instance** dropdown menu showing "INST0".
- Variable List** table:

Variable List			
ABC	VAR	INT	
ABC_ADD	VAR	INT	
DATA_OUT	VAR	INT	
DATA_S1	VAR	INT	
DATA_S2	VAR	INT	
DATA1	VAR	ARRAY	
DATA2	VAR	ARRAY	
F1	VAR	INT	

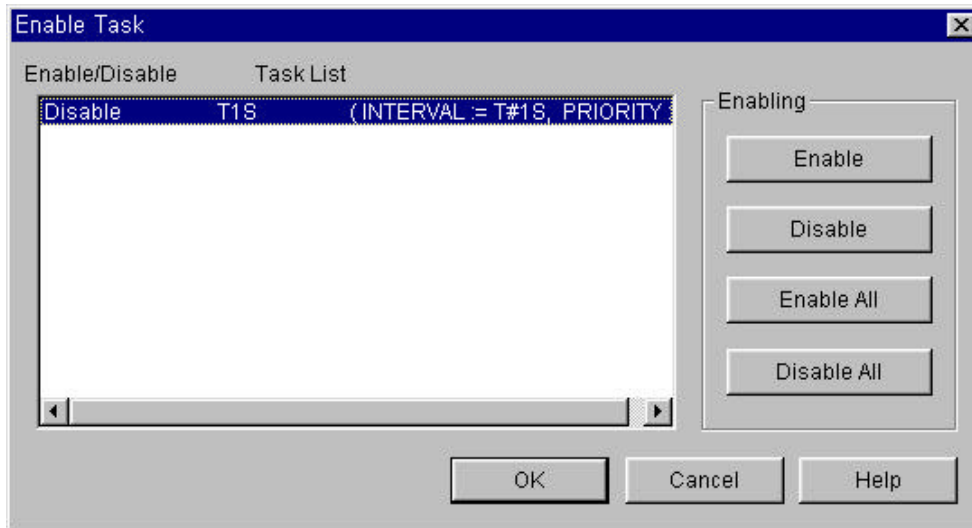
Buttons: OK, Cancel, Help.

- ◆ Choose one option in **Variable...** option radio box.
 - Read** : stops, when it reads the variable.
 - Write** : stops, when it writes an arbitrary value to variable.
 - Read/Write** : stops, when it reads a variable or writes an arbitrary value to variable.
- ◆ If you want to stop when reached to a specific value, select the **Enable Variable Break** option check box, and enter the value in the input text box.

9.6. Enable Task Turnover

In a program instance with a task, if you run the program after debugging by one step, the **Task Enable...** allows you to choose either Enable or Disable of the task turnover.

- ◆ Select the **Debug-Task Enable...** in the pull-down menu.



- ◆ Select a task to Enable/Disable in the **Enable Task** box.
- ◆ Click **Enable/Disable** button.
- ◆ When you want to set the entire lists as a enable/disable, click the **Enable/Disable All** button.
- ◆ Click **OK** button.