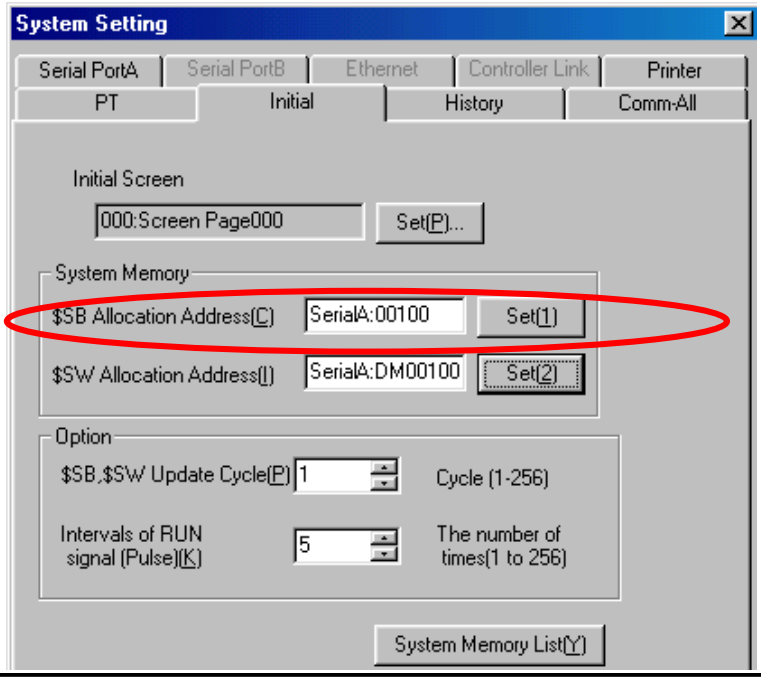
PT Status Control Area, PT Notification Area, and Windows Control Area

BEFORE CONVERSION	The PT can be controlled using the ON/OFF status or numeric contents of PLC addresses allocated as the PT Status Control Area and the Windows Control Area. The PLC addresses allocated as the PT Notification Area provide notification of the PT's operating status.
AFTER CONVERSION	The NS-series PTs do not have the control and notification areas defined in the NT-series PTs, so they are not converted. Instead, the NS-series PTs provide system memory bits ($SB) and system memory words ($SW). The beginning address of the system memory words $SW and the beginning address of the PT Status Control Area are both "screen switching," so this address is allocated for the first word of $SW during conversion.
COUNTERMEASURE	Select ***Setting - System Setting - Initial*** from the menus and set $SB.



Note: The default internal memory address $B will be allocated for $SB during conversion. Use the NS-Designer to set a PLC address for $SB or to set both $SB and $SW in internal memory. If this setting is not made, normal operation may not be possible even if screen data is transferred to the PT.

Objects and Overall Operation

Fixed Displays, Numeral Displays, Numeral Inputs, String Displays, String Inputs, Lamps, and Touch Switches

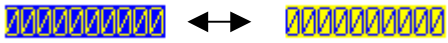
♦Function Settings: Display Attributes

BEFORE CONVERSION

Normal: Character colors and background colors appear as specified.

Flashing: The display is switched repeatedly between the normal display and background color.

Inverse/Flashing: The character and background colors are repeatedly switched.



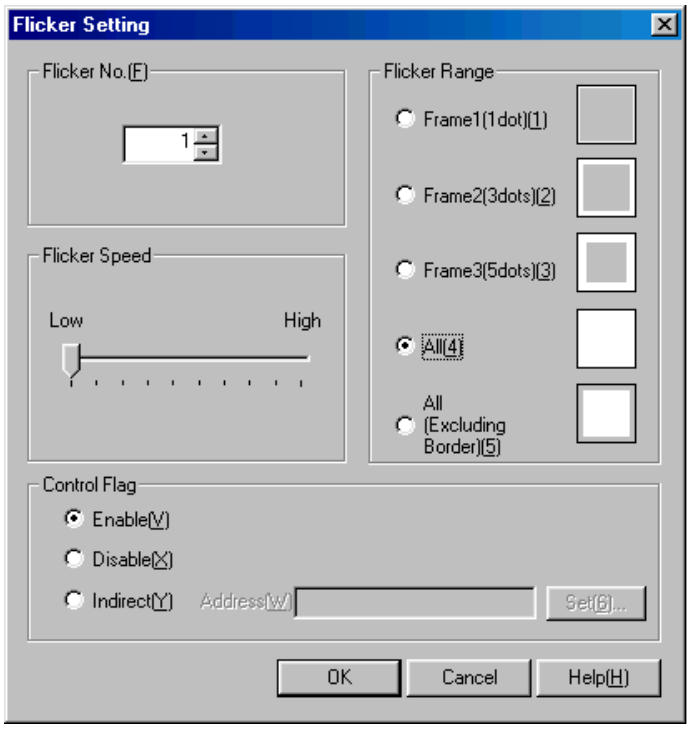
AFTER CONVERSION

Flashing and inverse/flashing are set to Flicker No. 1, and the operation for inverse/flashing will be the same as for flashing.

COUNTERMEASURE

Define flicker settings separately from functional objects in advance, and then allocate a flicker setting to each functional object. To change the flicker method for individual objects, select ***Settings - Flicker Setting*** from the NS-Designer menus, and make the settings for the required flicker numbers. Then allocate a flicker number for each functional object.

SETTINGS



Display Operation

Numeral Input and String Input

Function Settings: Cursor Frame Attributes

BEFORE CONVERSION — The selected input field uses the flashing or inverse operation set in the cursor attributes.

AFTER CONVERSION — Input becomes possible in the selected input field as shown in the following diagram, but flashing and inverse operation are not possible.

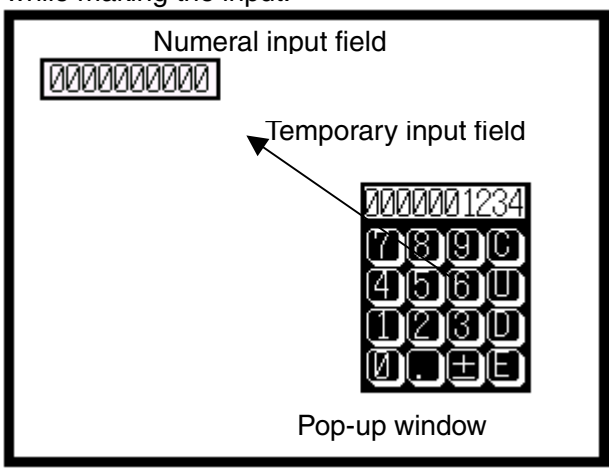
COUNTERMEASURE — No functionality is provided for flashing or inverse display of input fields.

♦Function Settings: Temporary Input Fields

BEFORE CONVERSION

Temporary input fields are used prior to inputting the data into the input field on the base screen.

The current setting can be confirmed in the temporary input field while making the input.

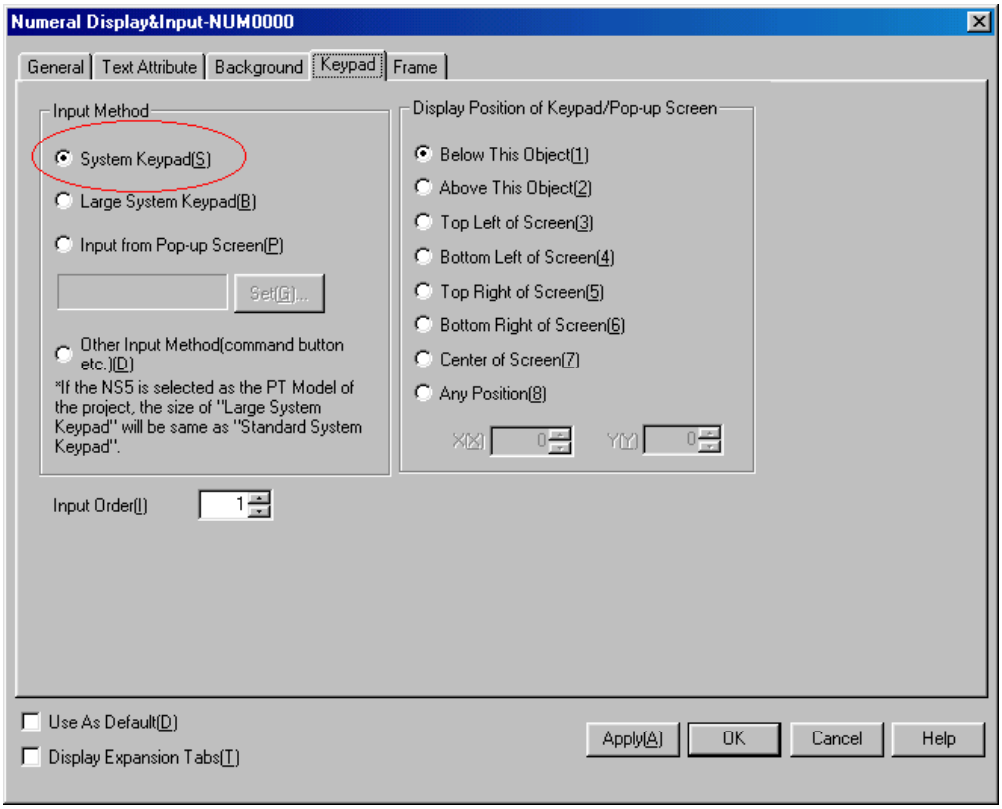


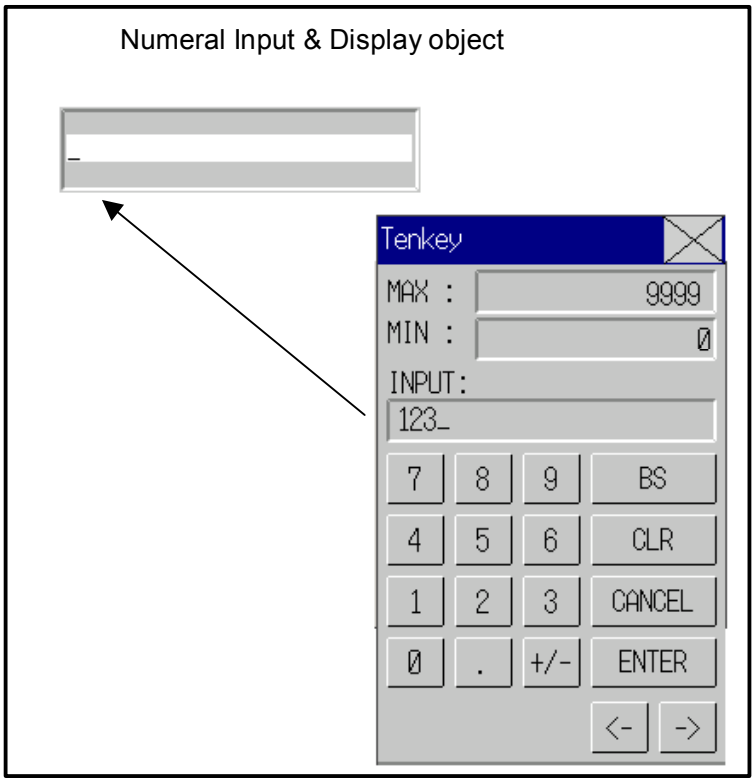
AFTER CONVERSION

The temporary input field in the pop-up window is deleted and the data is input directly to the input field on the base screen.

COUNTERMEASURE

Select ***System Keypad*** in the keypad tab of the numeral input and display object, turn the $SB18 ON (Display keypad with temporary input), to enable inputting numeral values. Then you can input values confirming the entered information.


Numeral Display&Input-NUM0000
General
Text Attribute
Background
Keypad
Frame
Input Method
System Keypad(S)
Large System Keypad(B)
Input from Pop-up Screen(P)
Set(G)...
Other Input Method(command button etc.)(D)
*If the NS5 is selected as the PT Model of the project, the size of "Large System Keypad" will be same as "Standard System Keypad".
Display Position of Keypad/Pop-up Screen
Below This Object(1)
Above This Object(2)
Top Left of Screen(3)
Bottom Left of Screen(4)
Top Right of Screen(5)
Bottom Right of Screen(6)
Center of Screen(7)
Any Position(8)
X(X)
Y(Y)
Input Order(I)
Use As Default(D)
Display Expansion Tabs(T)
Apply(A)
OK
Cancel
Help


Numeral Input & Display object
Tenkey
MAX :
9999
MIN :
0
INPUT:
123_
7
8
9
BS
4
5
6
CLR
1
2
3
CANCEL
0
.
+/-
ENTER
<-
->

Graphs

♦Function Settings: Percentage Displays

BEFORE CONVERSION The contents of the numeral memory tables for the specified values were displayed as percentages.

AFTER CONVERSION Percentage displays are not supported and this functionality is not converted.

COUNTERMEASURE Use a macro when changing graph values to calculate and display percentages based on the current value and the 100% value.

Broken-line Graphs

♦Function Settings: Number of Display Points

BEFORE CONVERSION Up to 320 points can be displayed.

AFTER CONVERSION Only up to 256 points can be displayed.

COUNTERMEASURE Create a separate broken-line graph for points 257 to 320.

♦Function Settings: Monitor Addresses

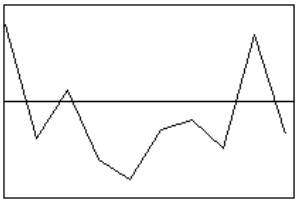
BEFORE CONVERSION The contents of multiple consecutive numeral memory table entries are displayed.

AFTER CONVERSION The contents of multiple consecutive addresses in the same channel are displayed.
It is not possible to display addresses from different channels or non-consecutive addresses on the same line.

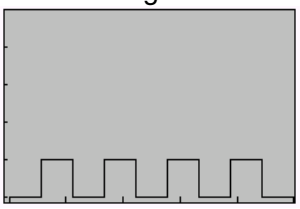
COUNTERMEASURE Allocate consecutive addresses.

♦Function Settings: Connections between Points

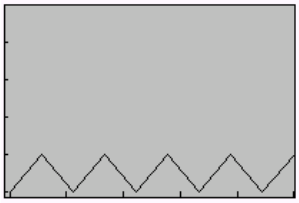
BEFORE CONVERSION Points are connected with straight lines.



AFTER CONVERSION The origin is connected by vertical and horizontal lines.



COUNTERMEASURE Remove the checkmark from the setting for step display in the line settings.



I/O Comments

Touch Switches and Lamps

BEFORE CONVERSION I/O comments can be input for objects. (I/O comments are accessed from I/O comment tables.)

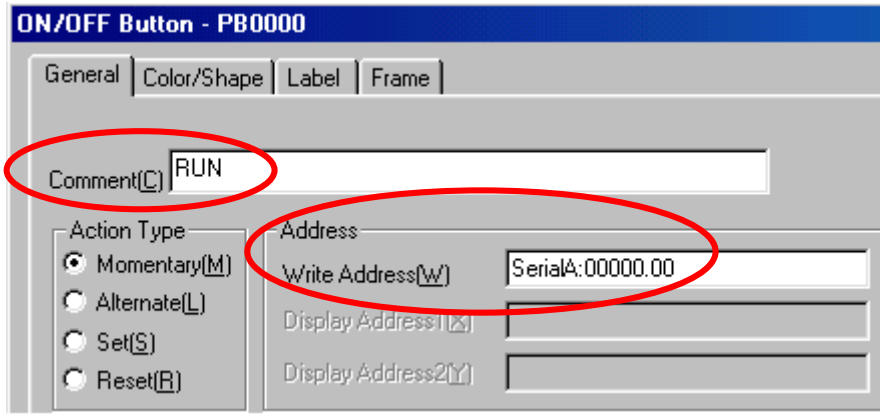
AFTER CONVERSION Although a comment field is supported, there is no I/O comment field. (The I/O comments will be converted to CSV format and output to a file.)

COUNTERMEASURE Input a suitable comment in the comment field for each object.

TSW1

Address: SerialA:00000.00
Comment: Start operation.

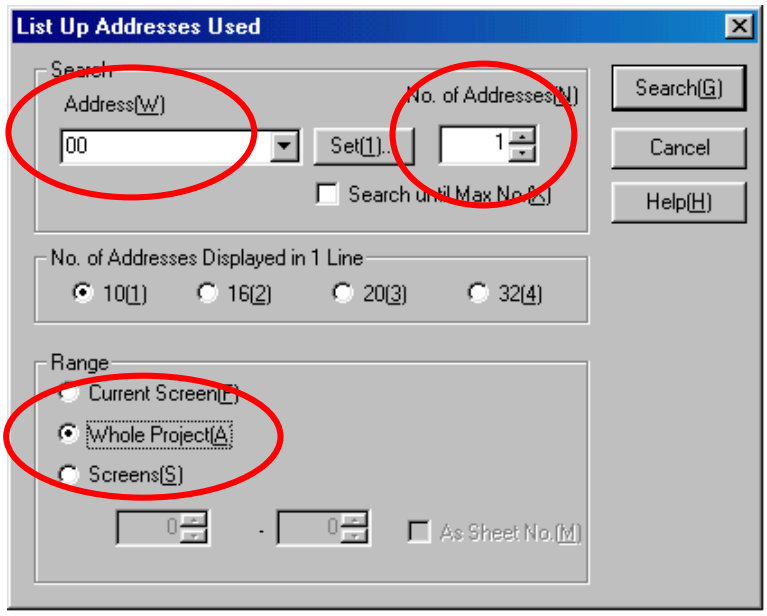
Open the CSV file containing the I/O comments, copy the I/O comment for SerialA:00000.00 and paste it into the Comment Field on the General Tab Page of the object properties.



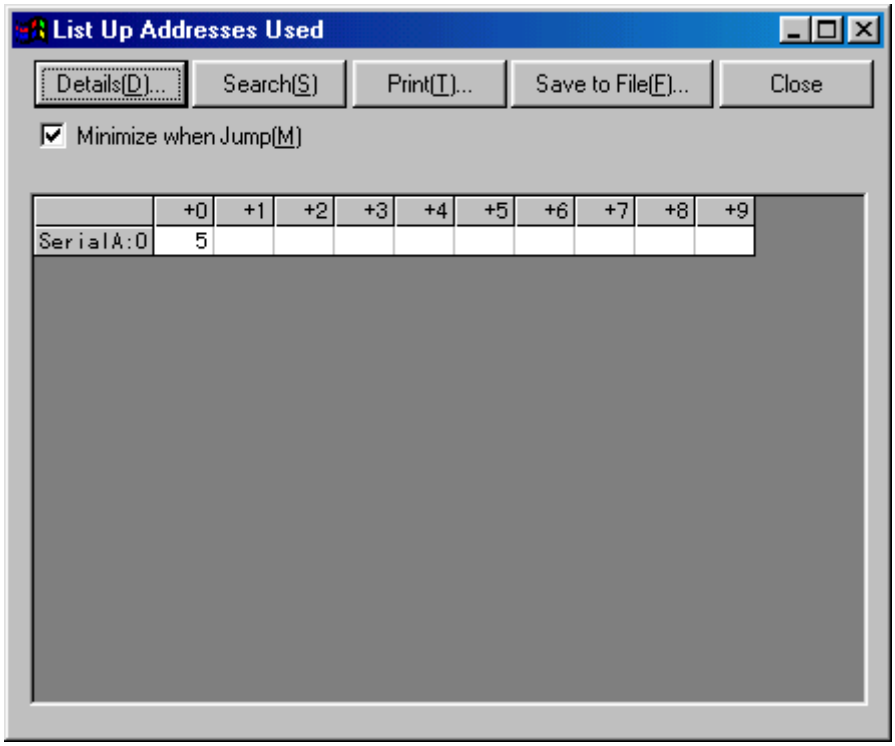
♦**Use the following procedure to add the same comment to multiple objects with the same address allocated to them.**

1. Select Tools - List Up Addresses Used.

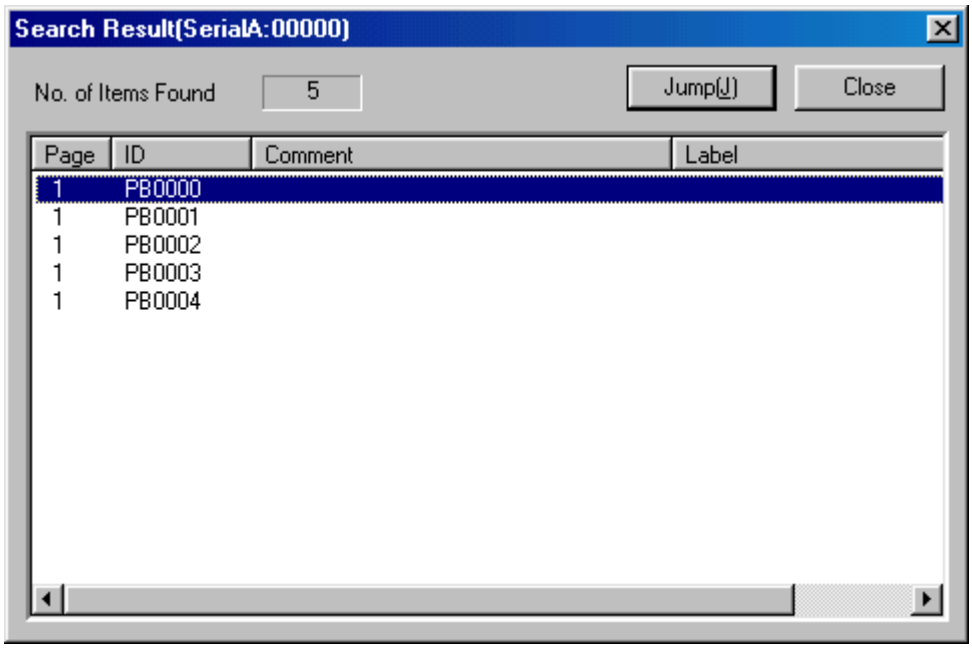
2. Set SerialA:00000.00 as the search address, set the number of addresses to 1, and set the search range to the *Whole Project.*



3. The following display will appear, showing where the address is used. Click the **Details** Button.



4. A list of ID numbers using SerialA:00000.00 will be displayed. Select one of these and click the **Jump** Button. An editing window will be displayed for the object.



5. Input the comment in the object properties.

6. Repeat steps 4 and 5 to input comments for all required objects.

Alarm Lists

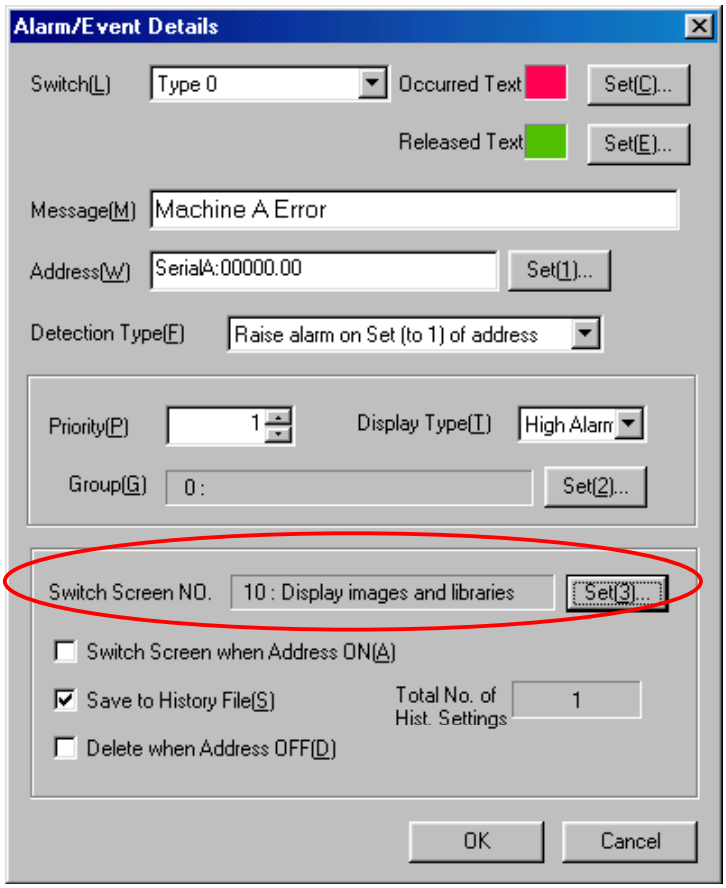
Image and Library Displays

BEFORE CONVERSION Images or library objects could be displayed by touching an alarm item.

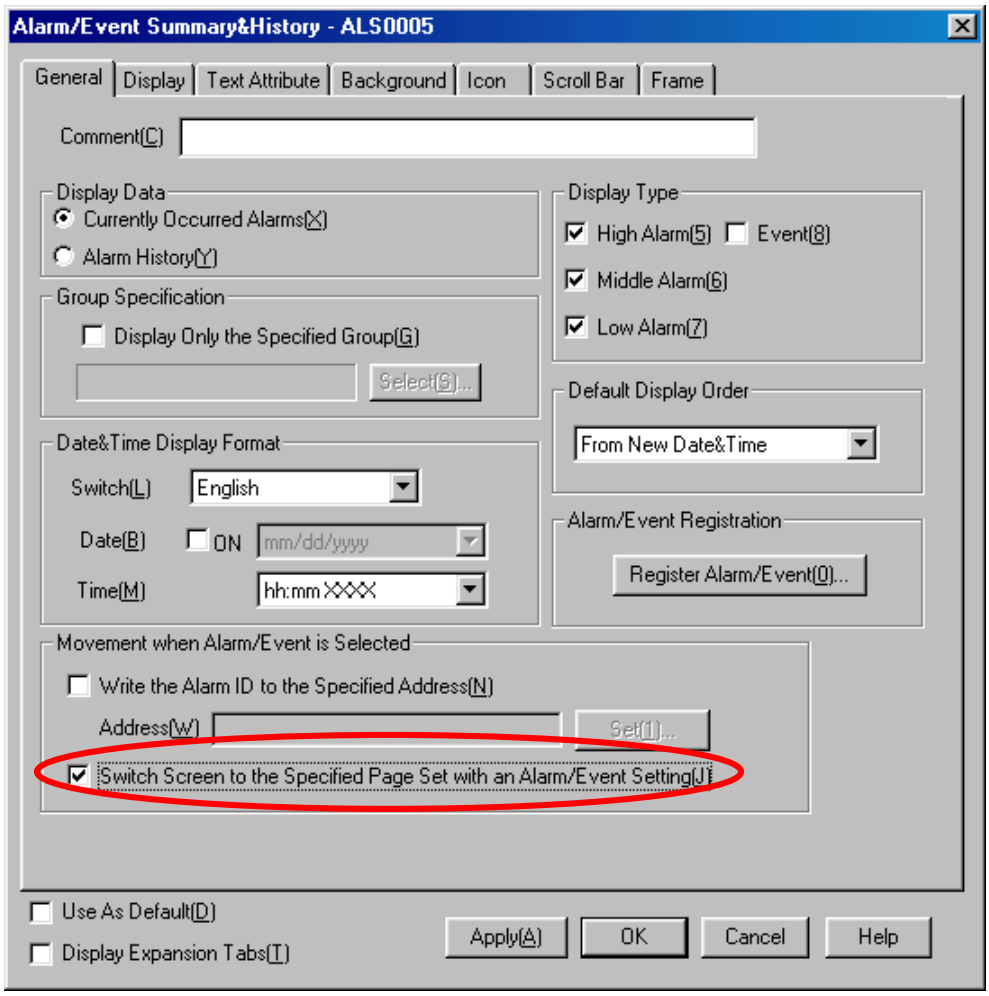
AFTER CONVERSION Images or library objects are not displayed even if an alarm item is touched.

COUNTERMEASURE Images or library objects can be displayed by using window screens.

1. Convert the images and library objects to bitmap (bmp) and vector (vec) files using the Conversion Support Tool
2. Create a window screen and display the desired image or library object using the bitmap object or a shape-specified lamp.
3. Double-click the alarm/event setting items (or select corrections). The following Alarm/Event Details Dialog Box will be displayed.
4. Set the screen number of the window screen that was just created as the switch screen number.



5. Turn ON Switch Screen to the Specified Page Set with an Alarm/Event Setting in the Alarm/Event Summary & History properties.



With the above settings, the window created in step 2 will be displayed when the alarm item is touched.

Current Alarm Display

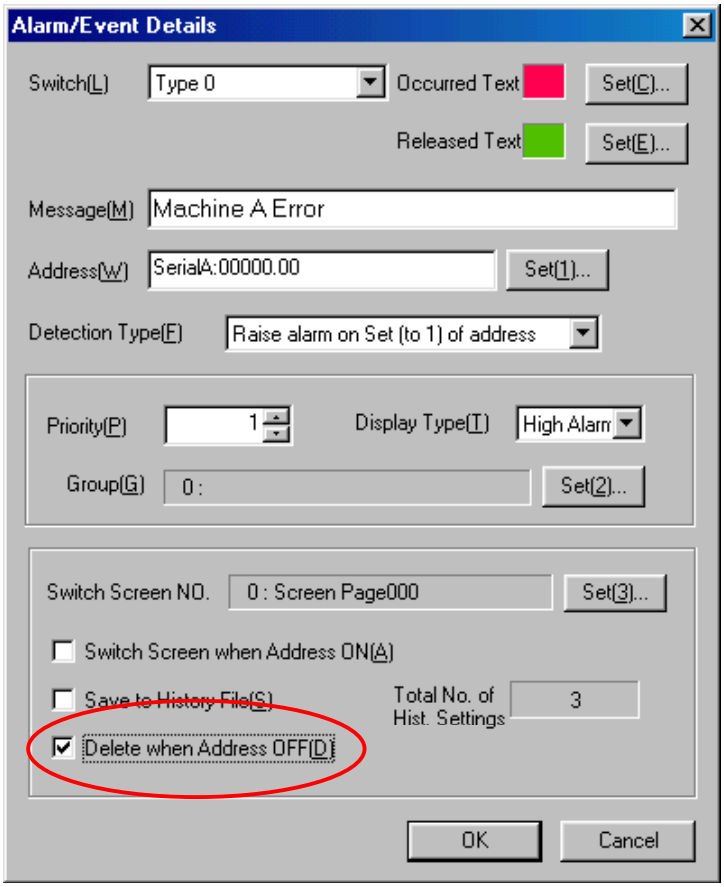
BEFORE CONVERSION — Only current alarms are displayed. An alarm is automatically removed from the display when the alarm is cleared.

AFTER CONVERSION — Alarms are not removed from the display even when they have been cleared. Alarms that have been cleared can be confirmed using the status colors displayed to the left of the alarms.

COUNTERMEASURE — The following method can be used to display only current alarms. If this method is used, however, history data will not be saved in the alarm history.

Change the following settings in the *Alarm/Event Details* properties.

- Remove the checkmark from *Save to History File*.
- Add a checkmark to *Delete when Address OFF*.



With the above settings, the alarm will be automatically deleted from the display when it is cleared (i.e., when the address turns OFF).

Alarm Registration

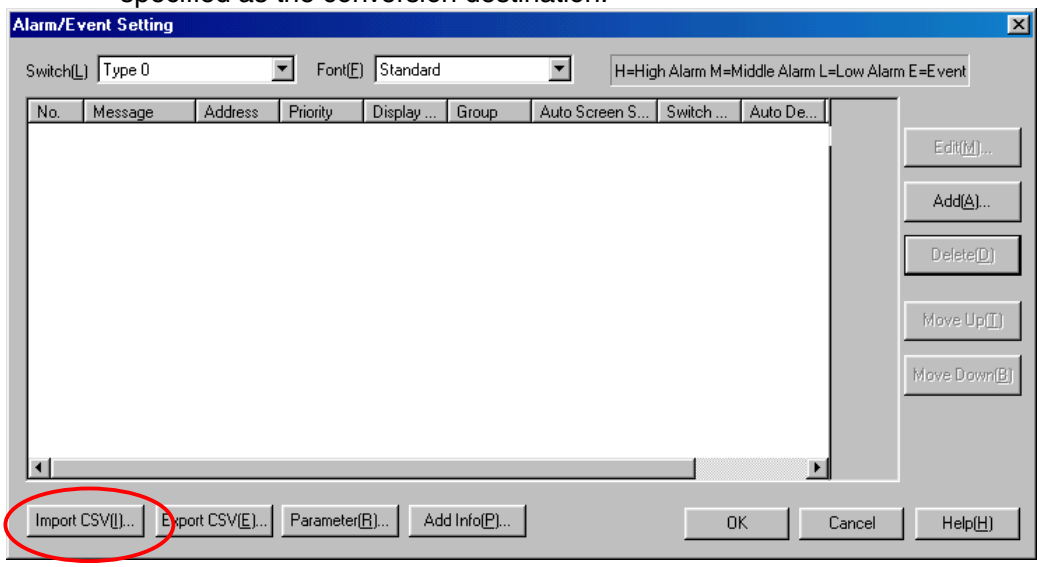
BEFORE CONVERSION Alarms are registered in the bit memory table.

AFTER CONVERSION The bit memory table is not converted per se, but the contents of the table are converted to a CSV file (Bitmem.csv). Alarms and events are not registered for Alarm/Event Summary and History objects by the Conversion Support Tool, so alarms will not be output even when they occur.

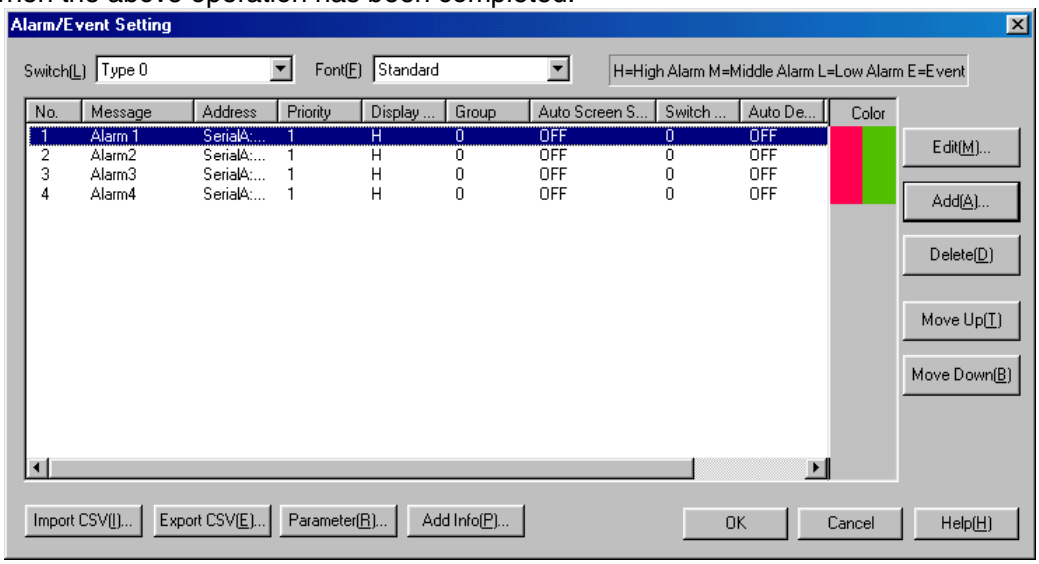
COUNTERMEASURE The following method can be used to register alarms.

1. Select ***Setting*** - ***Alarm/Event Setting*** from the menus.

2. Click the **Import CSV** Button and open the Bitmem.csv file. Bitmem.csv will be created in the project folder under the folder specified as the conversion destination.



The alarms that have been registered will be displayed in the Alarm/Event Setting Dialog Box when the above operation has been completed.



Note: Although up to 1,024 alarms can be registered with NT-series PTs, only up to 500 (1000 for the system program ver.5 onwards) can be registered with NS-series PTs. If there are more than 500 (1000) alarms registered before conversion, edit the CSV file and reduce the number to 500 (1000) or less.

I/O Comments

BEFORE CONVERSION	I/O comments can be input for alarm bits.
AFTER CONVERSION	Neither I/O comments nor normal comments can be added.
COUNTERMEASURE	If only label 0 is used, the column to the right of the last column in Bitmem.csv (i.e., the column for label 0 alarm messages) can be used to input comments. (This will be ignored during alarm registration.)

Initialization

BEFORE CONVERSION	Initialization of numeral and string memory tables at startup could be enable or disabled.
AFTER CONVERSION	The contents of the referenced addresses are always displayed.
COUNTERMEASURE	The NS-series PTs do not provide the functionality to assign default values for addresses, so the status when power is turned OFF (resume ON and initialize ON) or PT default values (resume OFF and initialize ON) cannot be displayed. Initialization, however, is possible by using macros to set default values for each address.

I/O Comments

BEFORE CONVERSION	I/O comments can be input for numerals.
AFTER CONVERSION	Nothing is displayed in the comment field.
COUNTERMEASURE	Input an I/O comment in the comment field for each object as required.

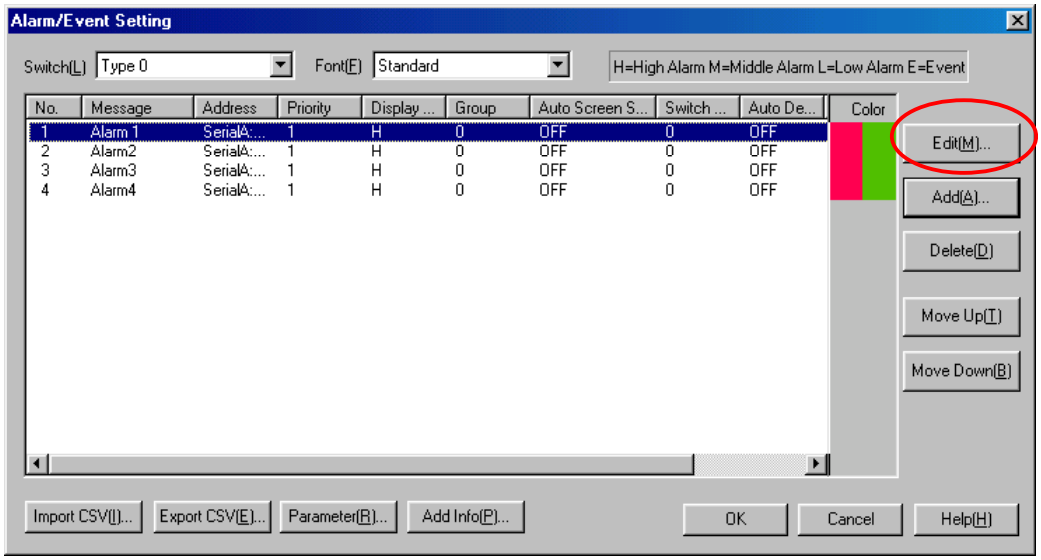
Bit Memory Tables

Screen Switches

BEFORE CONVERSION — The screen is changed to the specified screen when the bit turns ON.

AFTER CONVERSION — The screen will not changed to the specified screen even if the bit turns ON.

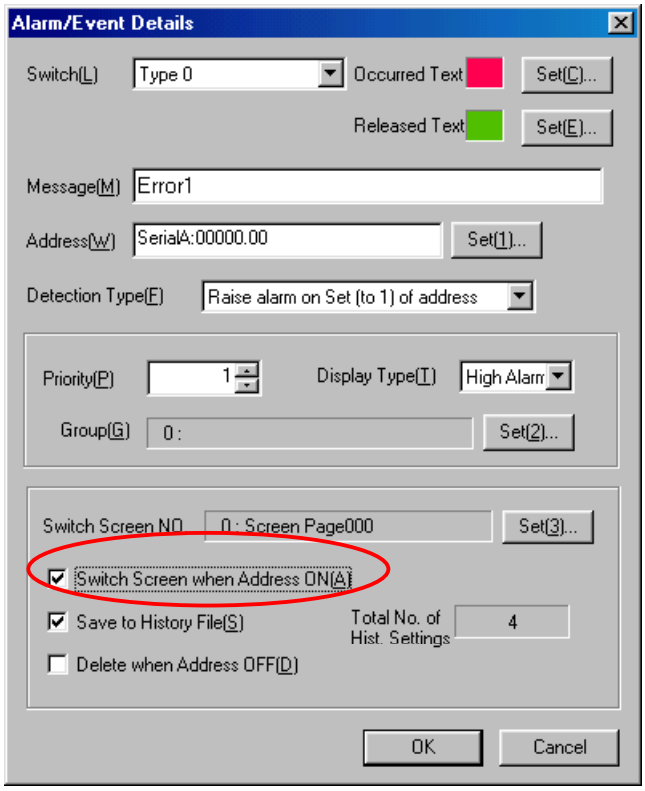
COUNTERMEASURE — The following procedure can be used to register alarms in advance in the Alarm/Event Details Dialog Box to switch screens.



1. Click the **Edit** Button.

2. Specify the bit in the alarm/event settings and turn ON *Switch Screen when Address ON.*

3. Input the number of the screen to switch to using the **Set** Button by *Switch Screen NO.*



With the above settings, the screen will change to the specified screen when the bit turns ON.

Comments

BEFORE CONVERSION Comments can be added to library objects.

AFTER CONVERSION Library objects are converted to vector (vec) files, but comments are not attached.

COUNTERMEASURE Use library object codes to differentiate.

Conversion Tables

Conversion Tables

BEFORE CONVERSION	Up to five calculations were possible.
AFTER CONVERSION	The NS-series PTs do not support this function and if conversions are used, display values will not be correct and the correct values will not be written to the PLC.
COUNTERMEASURE	Use the unit and scale settings to scale numeric values. Use macros for other conversions. Constant conversions are not possible.

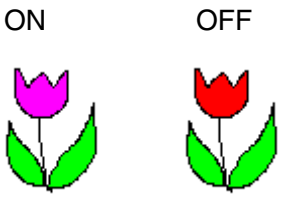
Image Lamps

Image Lamps

BEFORE CONVERSION	Different images can be created and specified for ON and OFF status.
AFTER CONVERSION	The Conversion Support Tool converts image and library object lamps to an ON bit lamp and an OFF bit lamp. If a library object was used, the bit lamp shape specification can be used to select the vector (vec) files converted from the library object, but selection will not be possible for images. Images will thus be converted to two transparent bit lamps.
COUNTERMEASURE	Images will be converted to bitmap (bmp) files. Indirect specification can be used for bit lamp objects to turn ON and OFF images (bitmap files). To achieve this, however, the address allocated for ON/OFF status must be converted to a word, requiring changes to the ladder program.

Example:

1. A bitmap image will be created in the project folder under the folder specified as the conversion destination.

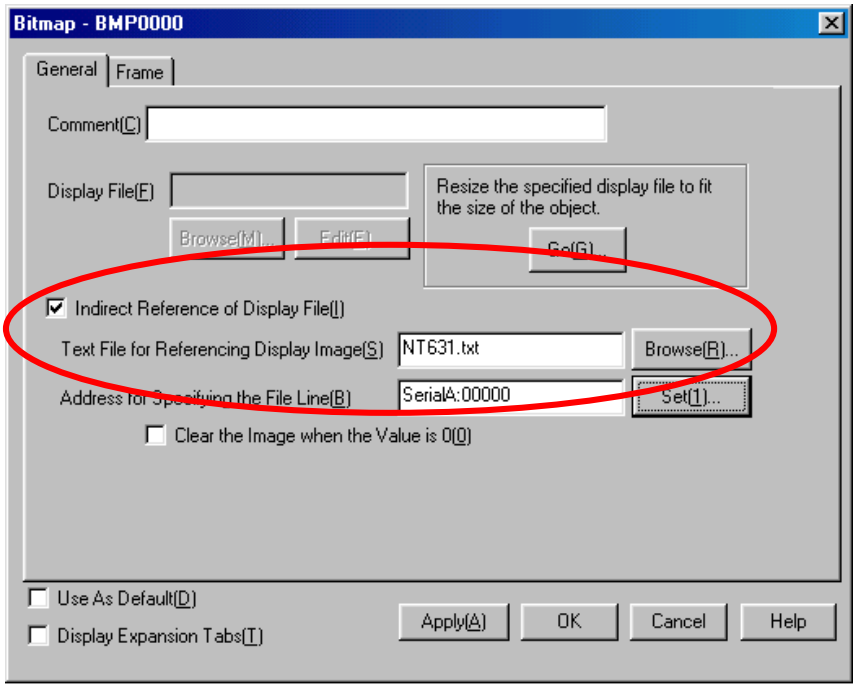


Img0001.bmp Img0002.bmp

2. Create a bitmap object on the NS-Designer.

3. Select the bitmap property settings.

- Turn ON Indirect Reference of Display.
- Set the *Test File for Referencing Display Image* to a text (txt) file in the project folder under the folder specified as the conversion destination
- Set the *Address for Specifying File Line* to any value. (Set to DM 0000 in this example.)


Bitmap - BMP0000
General
Frame
Comment(C)
Display File(F)
Browse(M)...
Edit(E)
Resize the specified display file to fit the size of the object.
Go(G)...
Indirect Reference of Display File(I)
Text File for Referencing Display Image(S)
NT631.txt
Browse(R)...
Address for Specifying the File Line(B)
SerialA:00000
Set(1)...
Clear the Image when the Value is 0(0)
Use As Default(D)
Display Expansion Tabs(T)
Apply(A)
OK
Cancel
Help

Screen Attributes

History

BEFORE CONVERSION	If the history has been enabled, the screen will be registered in the screen display history when the screen is displayed.
AFTER CONVERSION	There is no screen display history in the screen attributes, and this function is not converted.
COUNTERMEASURE	Use the screen display log in the operation logs instead. With this function, the screens that are displayed will be registered as long as $SB40 in system memory is ON.

Perform the following procedure to use operation logs (functional object operation and macro execution).

1. Make operation log settings.
   Select *Operation Log History* in the Write Settings Dialog Box under functional object properties settings. This sets the object for operational log recording.
   There are 5 functional objects to which operation logs can be set: ON/OFF buttons, word buttons, numeral display and input objects, string display and input objects, and thumbwheel switches.

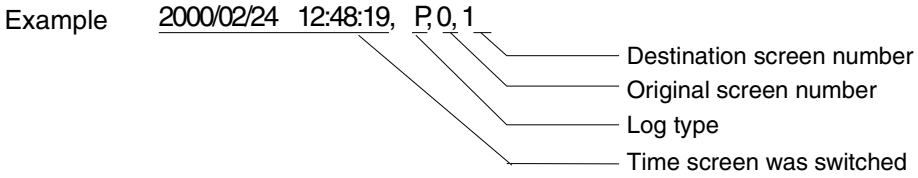
2. Activating the operation logs.
   Whether records will be kept for functional object operations, screen displays, or macro execution will be determined by the contents of system memory for each object.
   a) Functional object operation log: $SB39 ON: Log, OFF: Do not log
   b) Screen display log: $SB40 ON: Log, OFF: Do not log
   c) Macro execution log: $SB41 ON: Log, OFF: Do not log

3. Operation Log Format
   The data listed below is recorded as the operation log. This data can be saved as a CSV file to a Memory Card by turning ON system memory bit $SB38. The CSV file name will be Operat.CSV. Up to 1,024 operations can be recorded for operation logs.

Screen display log and time

- Log type (“P”)
- Original screen number
- Destination screen number

Example: Reading screen display histories saved as CSV files

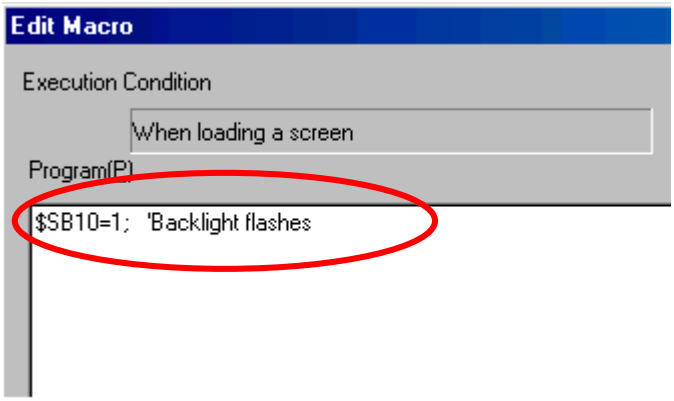


Meaning: Switched from screen 0 to screen 1 on 2000/02/24 at 12:48:19.

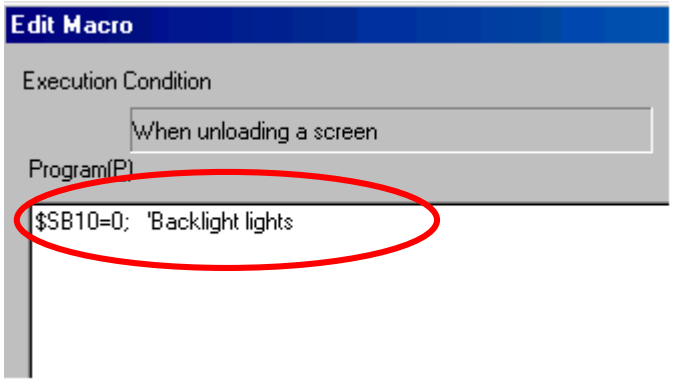
Backlight

BEFORE CONVERSION	Selection was possible between lighting and flashing.
AFTER CONVERSION	There is no backlight setting for lighting and flashing in the screen attributes, and this function is not converted.
COUNTERMEASURE	Bit $SB10 in system memory (Backlight Flashing Control Bit) can be manipulated by macros when loading and unloading screens to achieve the same results.

♦Turn ON $SB10 with the macro for screen loading.



♦Turn OFF $SB10 with the macro for screen unloading.



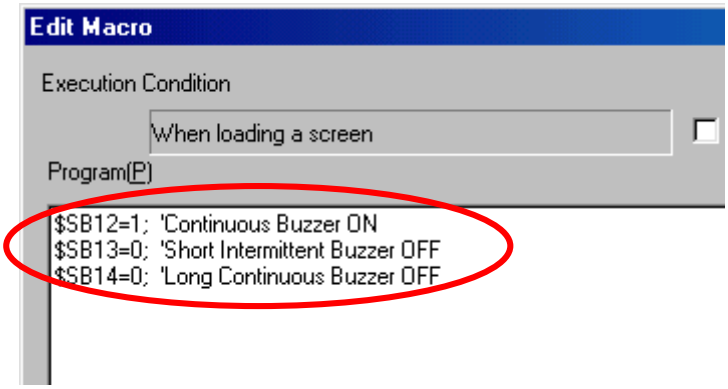
Buzzer

BEFORE CONVERSION	Continuous, short intermittent, or long intermittent buzzing could be selected in the screen attributes.
AFTER CONVERSION	There is no attribute to set the buzzer in the screen attributes, and this function is not converted.
COUNTERMEASURE	Bits $SB12 (Continuous Buzzer), $SB13 (Short Intermittent Buzzer), and $SB14 (Long Intermittent Buzzer) in system memory can be manipulated by macros when loading and unloading screens to achieve the same results.

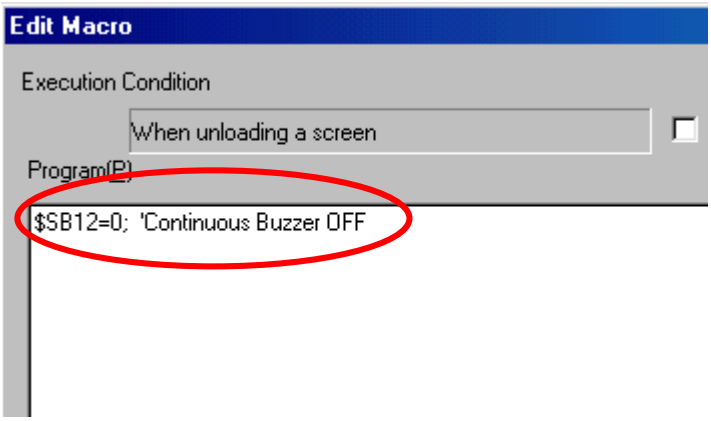
> Note: If more than one of these bits turns ON, the priority order for the buzzers will be $SB12 (Continuous Buzzer), $SB13 (Short Intermittent Buzzer), and $SB14 (Long Intermittent Buzzer).

Setting a Continuous Buzzer

♦**Turn ON $SB12 with the macro for screen loading.**

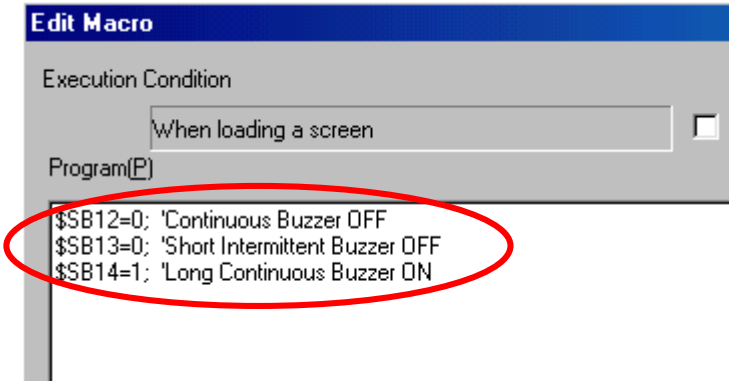


♦Turn OFF $SB12 with the macro for screen unloading.

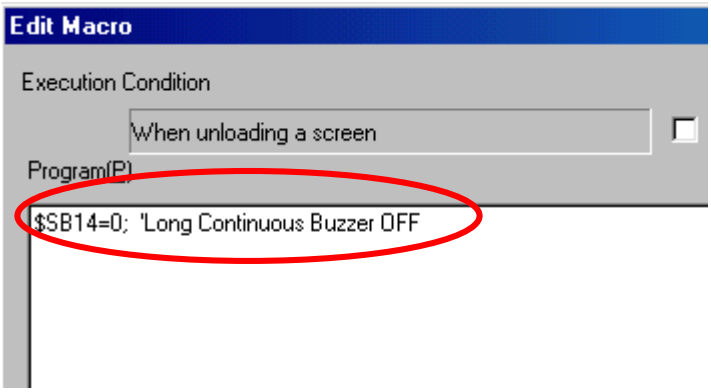


Setting a Short Intermittent Buzzer

♦Turn ON $SB14 with the macro for screen loading.



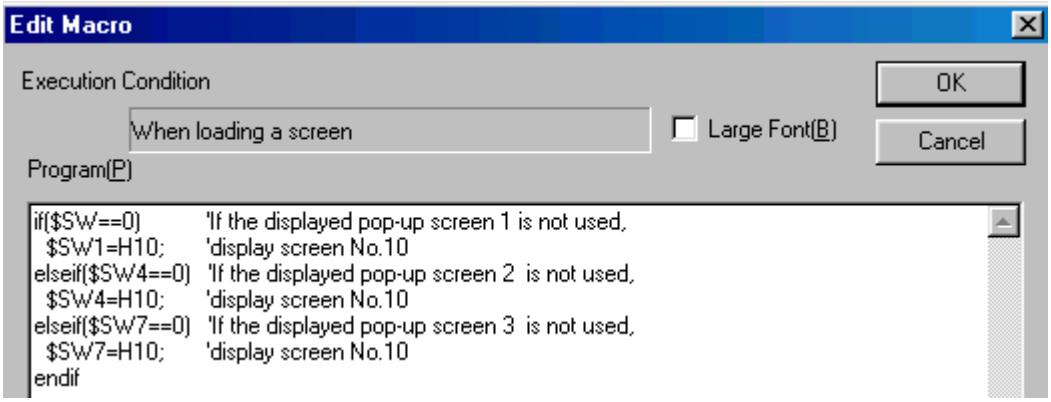
♦Turn OFF $SB14 with the macro for screen unloading.



Window Displays

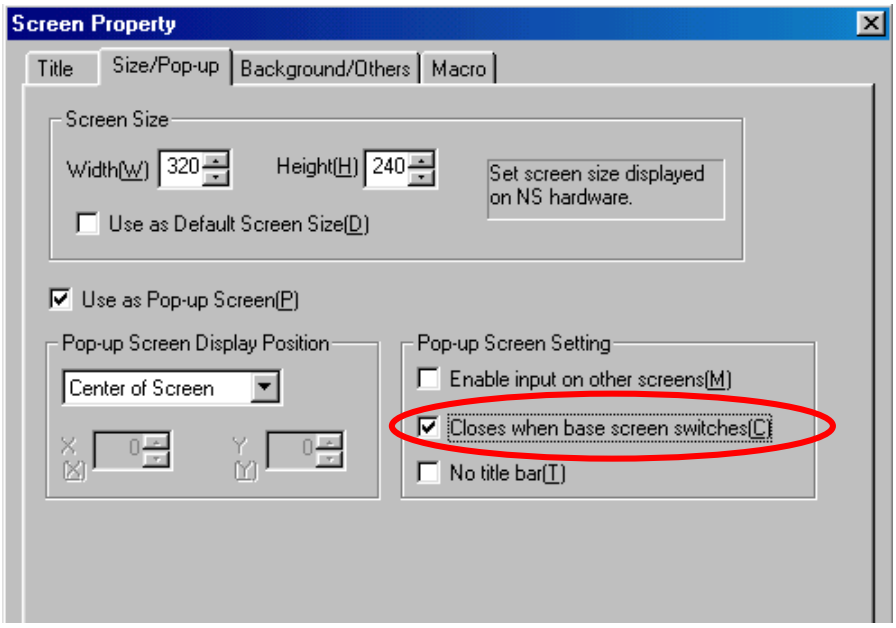
BEFORE CONVERSION	A local window could be displayed at the same time as the screen by setting a local window in the screen attributes.
AFTER CONVERSION	There is no local window setting in the screen attributes, and this function is not converted.
COUNTERMEASURE	The screen number of a window can be written to words $SW1 (Displayed Pop-up Screen 1), $SW4 (Displayed Pop-up Screen 2), and $SW7 (Displayed Pop-up Screen 3) in system memory with a macro when loading the screen.

1. Use the following macro when loading the screen.
   (Example for displaying screen 10.)



Note: If three global windows are already displayed with the base screen, screen 10 will not be displayed for the above macro. In this case, the above macro will need to be altered.

2. Select *Closes when Base Screen Switches* on the Size/Pop-Up Tab Page in the pop-up screen properties.



Grids

A memory map image file (.mmi) is used in the file before conversion, so the grid setting information in this file will be lost. The grid setting information will not be saved after conversion with the Conversion Support Tool.

COUNTERMEASURE A grid must be set separately for each screen.

> Note: The object group information is also stored in an .mmi file and is thus lost.

Host Connection Screen

Screen 9000

BEFORE CONVERSION	Screen 9000 could be created as a host connection screen.
AFTER CONVERSION	A host connection screen cannot be created and so screen 9000 is not converted.
COUNTERMEASURE	The functionality to create a host connection screen is not supported.

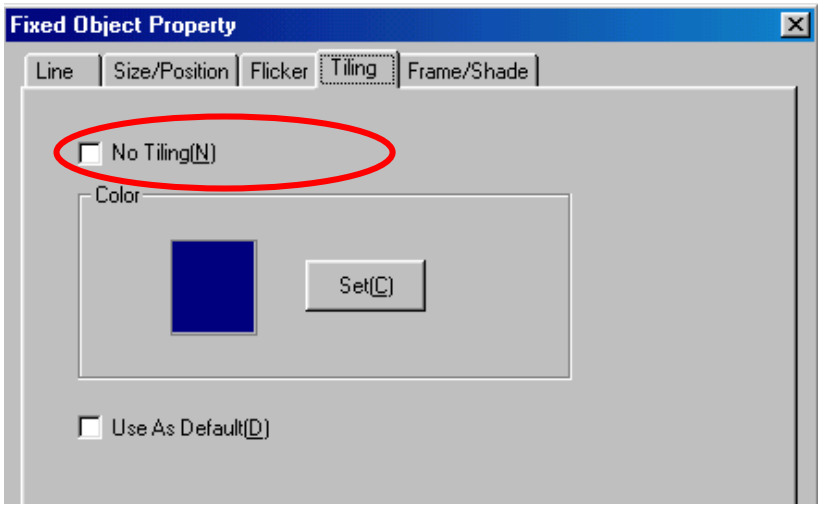
Fixed Displays

Filling (Tiling)

BEFORE CONVERSION	Filled fixed displays could be filled to include the border as well.
AFTER CONVERSION	Filling is not converted for objects.
COUNTERMEASURE	The countermeasure depends on whether the filled region is a single object or not.

♦Single Objects

1. Open the Tiling Tab Page in the properties of the converted object and remove the checkmark from *No Tiling.*



2. Set the desired color under *Color.*

♦Multiple Objects

Recreate the area to be filled as a new object (rectangle or polygon) and fill it with the desired color.

Fixed Display Images

Indirect References

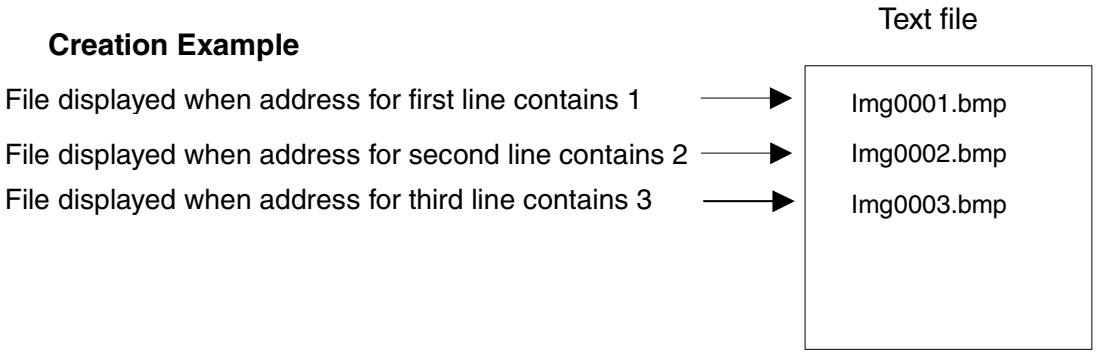
BEFORE CONVERSION An image can be displayed using indirect referencing by specifying an image number in a numeral memory table.

AFTER CONVERSION The object is not converted when indirect specification is used.

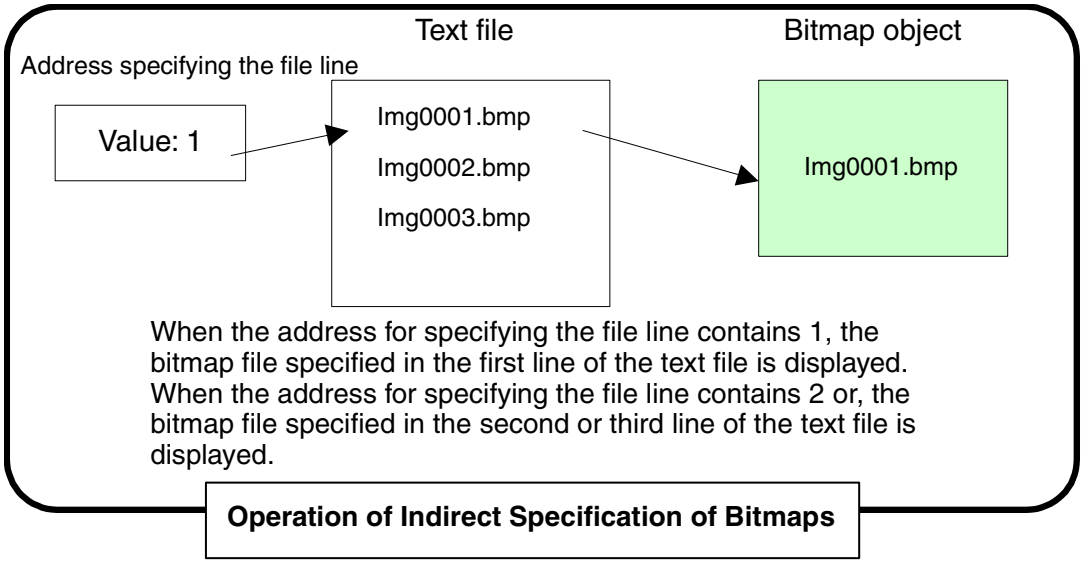
COUNTERMEASURE The following procedure can be used for indirect referencing with bitmap displays.

1. The image is saved by the Conversion Support Tool as a bitmap file in the project folder under the folder specified as the conversion destination. Check the image.
2. Create a bitmap object where the images are to be displayed.
3. Create a text file that will be used for indirect specification. In the text file, input the following for the bitmap file name converted by the Conversion Support Tool.
   Save the text (txt) file in the project folder under the folder specified as the conversion destination.

Creation Example



4. Open the bitmap object properties and select indirect specification of the display file.
5. Click the **Browse** Button next to *Display image file* and specify the text file created in step 3.
6. Specify the address for specifying file lines. If necessary, select *Clear display when 0*.



Operation of Indirect Specification of Bitmaps

Indirect References

BEFORE CONVERSION	An image can be displayed using indirect referencing by specifying an library object number in a numeral memory table.
AFTER CONVERSION	The object is not converted when indirect specification is used.
COUNTERMEASURE	The Conversion Support Tool converts library objects to vector (vec) files, but indirect specification for vector files is not possible for NS-series PTs. If less than 11 library objects are to be indirectly specified, then the shape specifications for a word lamp can be used instead. An operation similar to indirect specification is achieved by specifying a vector file for each of the values 0 to 9 of the specified address.

Opening Pop-up Windows and Keyboards

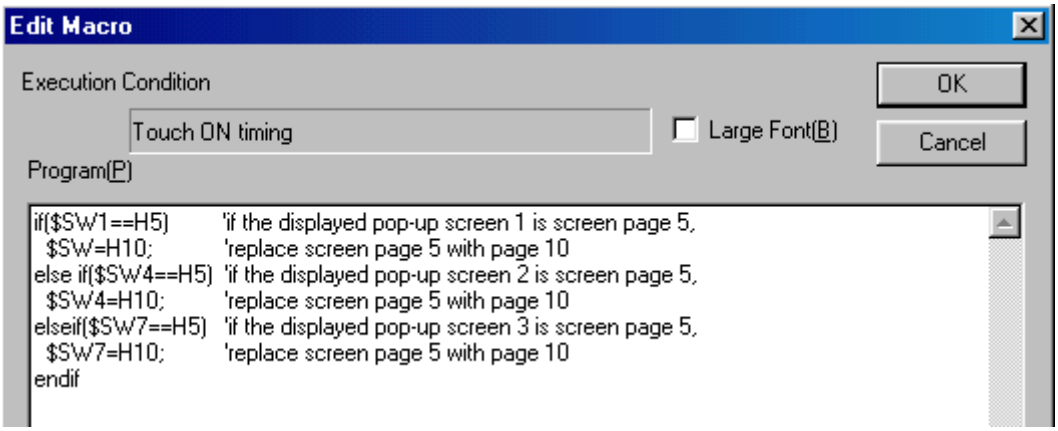
BEFORE CONVERSION	Two local windows can be opened from the base screen.
AFTER CONVERSION	The conversion results will contain a setting to disable inputting on another screen when one is already open. If there is a touch switch that opens multiple windows on the base screen and a window is open, inputs to the base screen will be disabled and it will not be possible open multiple windows.
COUNTERMEASURE	Add a checkmark to *Enable Input on Other Screens* on the Size/Pop-Up Tab Page in the pop-up screen properties.

BEFORE CONVERSION	A function to replace the local window can be used to close the open local window and open another one.
AFTER CONVERSION	The data is converted to a command button to open a window (pop-up screen). This enables opening another window without closing the currently open window. If, however, there is a touch switch that opens multiple windows on the base screen, the previously described countermeasure must be taken.
COUNTERMEASURE	The screen number of a window can be written to words $SW1 (Displayed Pop-up Screen 1), $SW4 (Displayed Pop-up Screen 2), and $SW7 (Displayed Pop-up Screen 3) in system memory with the macro for when the touch switch is touched to turn it ON.

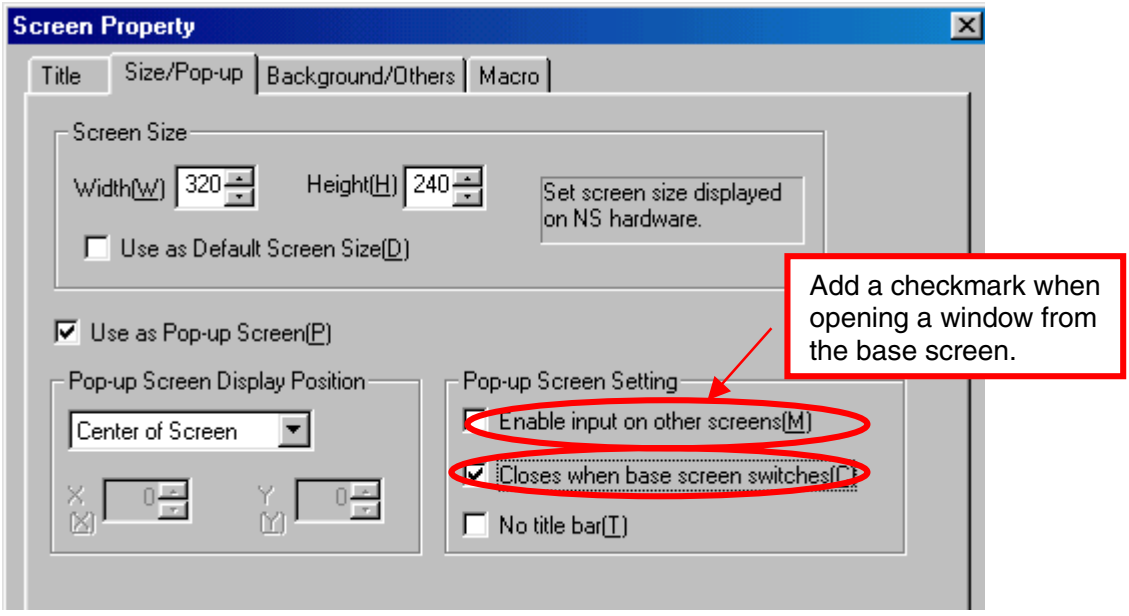
1. Create a command button and select no processing for it.

2. Use the following macro when the touch switch is touched to turn it ON.

(Example for replacing screen 5 with screen 10.)



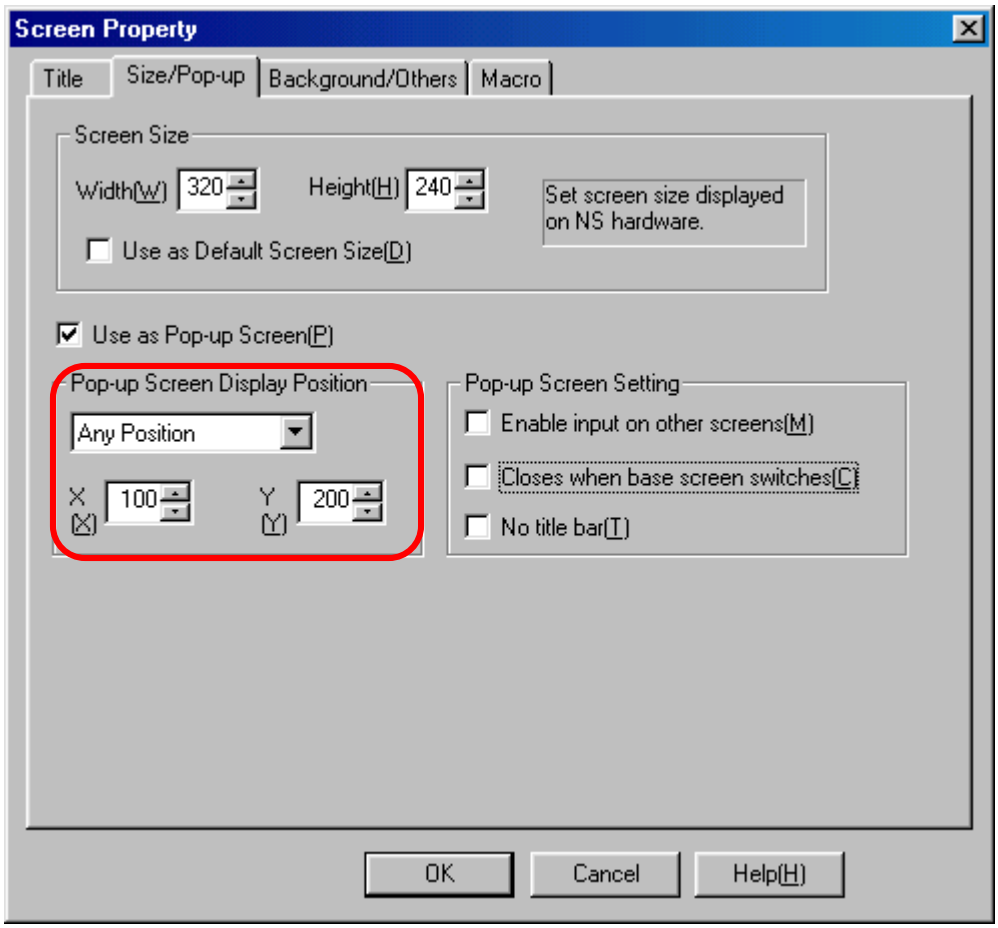
3. Select *Closes when base screen switches* on the Size/Pop-up Tab Page in the pop-up screen property settings of the screen properties for the open window (screen 10).



BEFORE CONVERSION The X and Y coordinates can be specified for opening a window.

AFTER CONVERSION There is no attribute to set the X and Y coordinates for a window open command button for NS-series PTs and this function is not converted. The window will be displayed in the center of the screen (the default position for windows (pop-up screens)).

COUNTERMEASURE Change the *Pop-up Window Display Position* in the screen properties for the window (pop-up screen) to *Any Position* and set the X and Y coordinates.



Note: With NT-series PTs, the lower left coordinates of the window are specified for the window display position. With NS-series PTs, the upper left coordinates of the window are specified for the window display position.

Closing Pop-up Windows and Keyboards

BEFORE CONVERSION Open windows can be closed with touch switches.

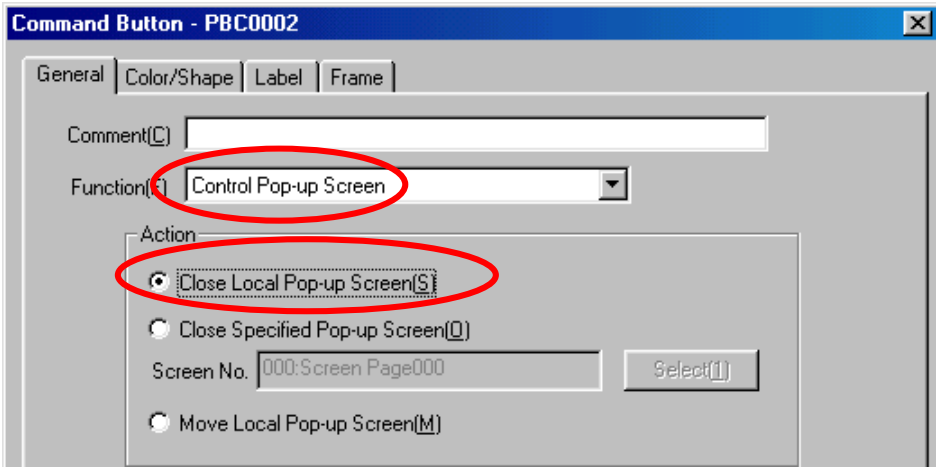
AFTER CONVERSION Touch switches to close windows are not converted.

COUNTERMEASURE Create the command button described below to close the window or use the close button in the window active bar to close the window.

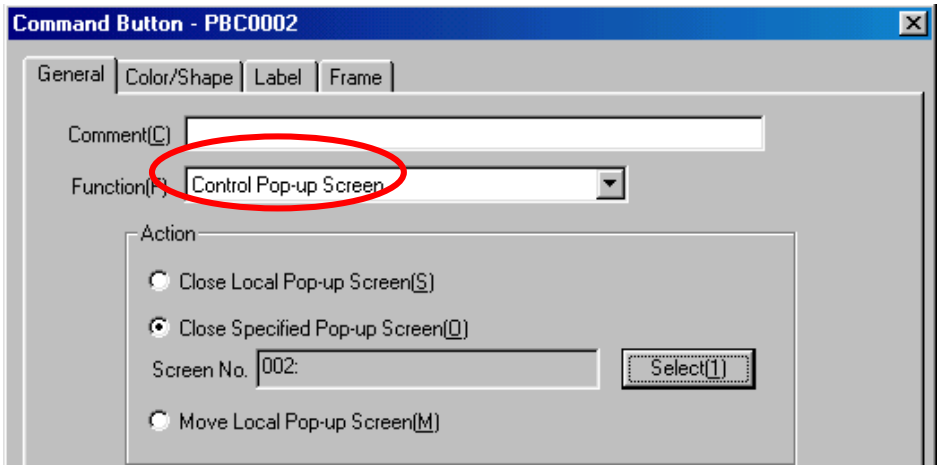
Creating a Command Button to Close the Window

1. Create a command button and select *Control Pop-up Screen* for it.
2. Use one of the following methods.
   a) Closing a Window with the Close Window Button Located on It
   b) Closing a Window Regardless of the Location of the Close Window Button (The local screen can also be specified.)

a) Add a checkmark to *Close Local Pop-up Screen.*



b) Add a checkmark to *Close Specified Pop-up Screen* and specify the screen using the **Select** Button. (Example for closing screen 2.)

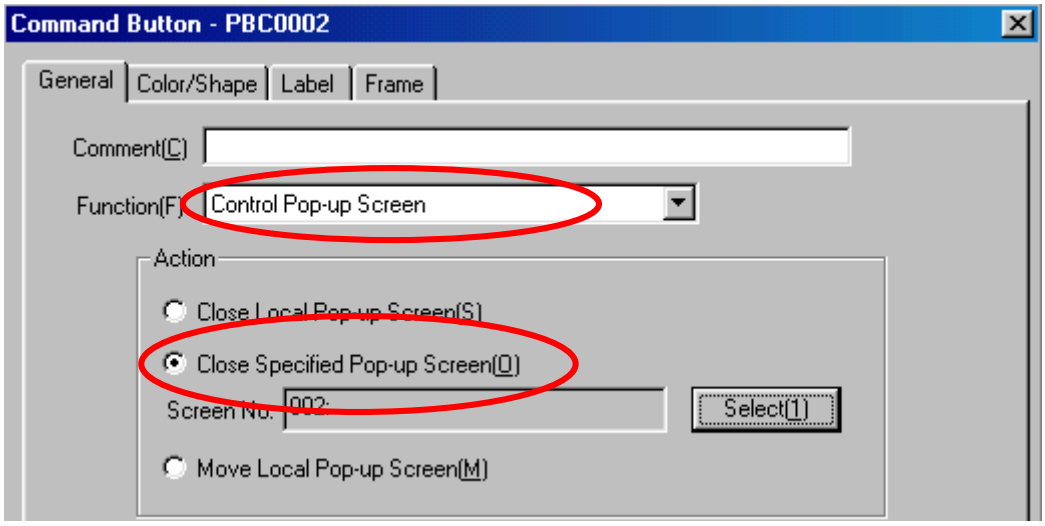


BEFORE CONVERSION Windows can be opened and closed using the ON/OFF status of touch switches.

AFTER CONVERSION The data is converted to a command button to open a window (pop-up screen).
Closing is not possible.

COUNTERMEASURE Add a command button set to *Control Pop-up Screen.*

1. Create a command button and select *Control Pop-up Screen* for it.
2. Add a checkmark to *Close Specified Pop-up Screen* and specify the screen using the **Select** Button.
(Example for closing screen 2.)



Touch Switches

Copy Setting

BEFORE CONVERSION — Copying was possible between entries in two numeral memory tables or two string memory tables.

AFTER CONVERSION — Memory tables are not converted so the copy function between tables is not converted.
Only the function to copy constants (numerals) at the position of the cursor (i.e., the input field with the focus) is converted.

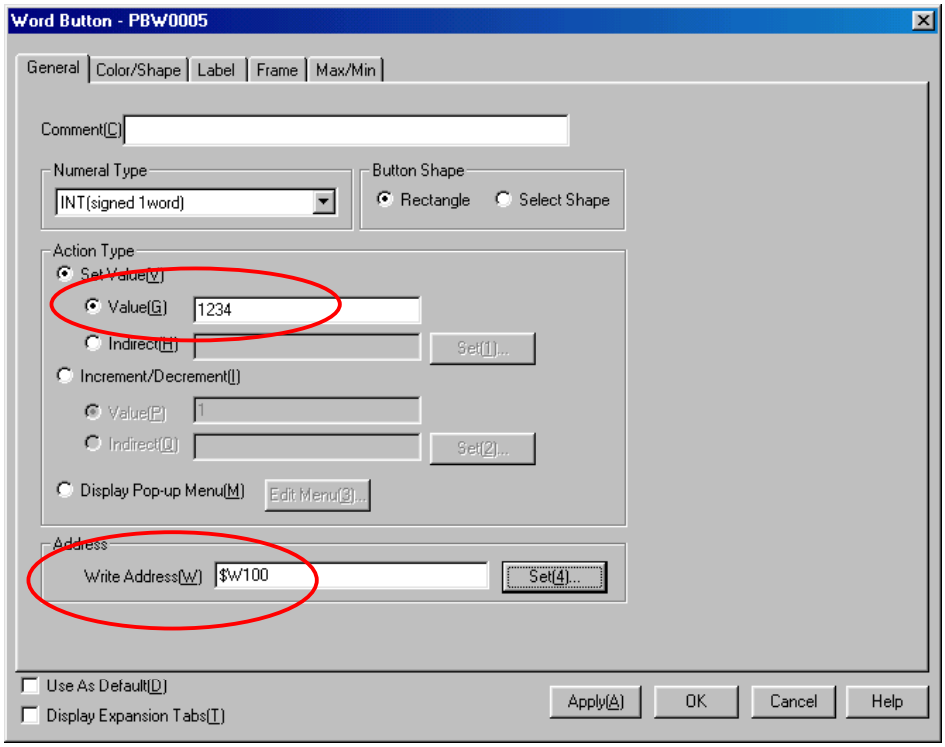
COUNTERMEASURE — Word buttons can be used instead if one of the following conditions is satisfied.

a) Copying a constant to PT memory or PLC memory
b) Copying from PT memory or PLC memory to PT memory or PLC memory

1. Create a word button. Input the destination address in PT or PLC memory into the *Write Address* Field.
2. Set the Action Type to Set Value.

a) Copying a Constant to PT Memory or PLC Memory

Set the constant as the *Value* under *Set Value*.
(Example for copying 1234 to $W100 in PT memory.)



b) Copying from PT Memory or PLC Memory to PT Memory or PLC Memory

Set the PT or PC memory address in the *Indirect* Field under *Set Value.*
(Example for copying the contents of CIO 00120 to DM 00100.)