Universal Probe

Software Users Manual

Boundary-Scan Tool

No. E090963-03

Table of Contents

Notes 9

Operating Precautions 10

Abbreviations, Terms and Conventions 12

1. Overview of This Software 14

1.1. System Requirements 14

1.2. Characteristics 14

1.3. Turning on or off the Power 15

1.3.1. Connecting to the Target and Turning on the Power 15

1.3.2. Turning off the Power and Disconnecting from the Target 15

2. How to Operate 16

2.1. Workflow 16

2.1.1. Starting the Boundary-Scan Tool 17

2.1.2. Operation procedures in the Project view 18

2.1.3. Operation procedures in the Waveform Display window 19

2.1.4. Operation procedures in the Pin Status List window 20

2.1.5. Operation procedures in the Connection Test window 21

2.1.6. Operation procedures in the Script pane 22

2.2. Project View 23

2.2.1. Create a new project 23

2.2.2. Open a project 23

2.2.3. Save a project 24

2.2.4. Close a project 24

2.2.5. Add a device 26

2.2.6. Edit a device 27

2.2.7. Delete a device 28

2.2.8. Move a device upward 28

2.2.9. Move a device downward 28

2.2.10. Device properties 28

2.2.11. Connect a probe 31

2.2.12. Select a probe 31

2.2.13. Probe communication (automatic device recognition) 32

2.2.14. Disconnect a probe 34

2.2.15. Set a JTAG clock 35

2.2.16. Select a JTAG clock 36

2.2.17. Set a JTAG clock (low speed) 37
2.2.18. Consistency test 37
2.2.19. Add a port group 38
2.2.20. Edit a port group 39
2.2.21. Delete a port group 39
2.2.22. Set a device reference name 39
2.2.23. Reset a device reference name 40
2.2.24. Change a port name (per device) 41
2.2.25. Change a Port Name (per port) 41
2.2.26. Reset a port name (per device) 42
2.2.27. Reset a port name (per port) 42
2.2.28. Save a changed port name 42
2.3. Waveform Display window 43
2.3.1. Create a new waveform window 43
2.3.2. Open a waveform in viewer 43
2.3.3. Rename a waveform 44
2.3.4. Delete a waveform 45
2.3.5. Add a port (signal) 46
2.3.6. Create a port group (signal) 47
2.3.7. Switch the port group (signal) display between binary and hexadecimal notation 48
2.3.8. Change the display order of ports/port groups (signal) 48
2.3.9. Delete the ports/port groups (signal) 49
2.3.10. Select a sampling period 49
2.3.11. Start sampling (start measurement) 50
2.3.12. Stop sampling (stop measurement) 50
2.3.13. Zoom in 50
2.3.14. Zoom out 50
2.3.15. Set a cursor 51
2.3.16. Move a cursor 52
2.3.17. Display cursor at center 52
2.3.18. Set a marker 52
2.3.19. Move cursor to marker position 53
2.3.20. Display marker at the center 53
2.3.21. Print a waveform 54
2.3.22. Save a waveform 54
2.4. Pin Status List window 55
2.4.1. Create/open a new pin status list 55
2.4.2. Change a list name 55
2.4.3. Delete a list 56
2.4.4. Add a port 57

2.4.5. Create a port group....58
2.4.6. Switch the port group display between binary and hexadecimal notation....58
2.4.7. Change the display order of ports/port groups....59
2.4.8. Delete a port/port group....59
2.4.9. Execute the SAMPLE command....61
2.4.10. Execute the EXTEST command....61
2.4.11. Save to a CSV file....65
2.5. Connection Test window....66
2.5.1. Create a new Connection Test window/open a Connection Test window....66
2.5.2. Rename a test (list)....67
2.5.3. Delete a test (list)....67
2.5.4. Add a net....68
2.5.5. Edit a pin list....69
2.5.6. Delete a net....70
2.5.7. Change the order in which a net is listed....71
2.5.8. Open a net list....71
2.5.9. Create/edit/delete a group....72
2.5.10. Switch whether to show pin names or port names....74
2.5.11. Set output values....74
2.5.12. Set expected values....75
2.5.13. Execute a test....76
2.5.14. Switch between binary and hexadecimal notation....77
2.5.15. Save a net list....78
2.5.16. Save the connection text results....79
2.6. Script pane....81
2.6.1 Select a file....82
2.6.2. Execute (file)....83
2.6.3. Execute (command line)....84
2.6.4. Save (in text format)....85
2.6.5. Save (in script format)....85
2.6.6. Clear the display....85

3. Menu....86

3.1. File....86
3.1.1. Create a new project....86
3.1.2. Open a project....86
3.1.3. Save a project....87
3.1.4. Close a project....87
3.1.5. Create a new Waveform window....88
3.1.6. Save a waveform....89

3.1.7. Open a waveform in Viewer 89
3.1.8. Create/open a new pin status list 89
3.1.9. Save a pin status list in CSV format 89
3.1.10. Create a new Connection Test window 90
3.1.11. Open a net list 90
3.1.12. Save a net list 90
3.1.13. Save connection test results 90
3.1.14. Preview Print 90
3.1.15. Set a Printer 91
3.1.16. Print 91
3.1.17. Exit application 92
3.2. View 93
3.2.1. Toolbar and docking window 93
3.2.2. Status bar 94
3.2.3. Script 94
3.3. Project 95
3.3.1. Connect 95
3.3.2. Disconnect 95
3.3.3. Add a device 95
3.3.4. Consistency test 95
3.3.5. Add a script file 95
3.3.6. Add a port group 96
3.3.7. Set a JTAG clock 96
3.3.8. Forced output of JTAG signal 96
3.4. Waveform Display 97
3.4.1. Start sampling (start measurement) 97
3.4.2. Stop sampling (stop measurement) 97
3.5. Pin Status List 98
3.5.1. SAMPLE Command 98
3.5.2. Execute the EXTEST command 98
3.5.3. Display warning when executing EXTEST 98
3.6. Connection Test 99
3.6.1. Learn expected values 99
3.6.2. Connection test 99
3.6.3. Display warning when outputting signal 99
3.7. Tools 100
3.7.1. Manage BSDL 100
3.7.2. Obtain the probe license 101
3.7.3. Specify an editor 102
3.8. Help 103

3.8.1. Display help 103
3.8.2. Version Information 103

4. Windows 104

4.1. Project View window 104
4.1.1. Toolbar 105
4.1.2. Project name 105
4.1.3. Probe information 105
4.1.4. Waveform display tree 106
4.1.5. Pin status list tree 106
4.1.6. Connection test tree 106
4.1.7. Script tree 107
4.1.8. Port group tree 107
4.1.9. Board tree 107
4.2. Waveform Display window 109
4.2.1. Port group name list column 109
4.2.2. Cursor (green solid line) 109
4.2.3. Markers (blue and orange dashed lines) 109
4.3. Pin Status List window 110
4.3.1. Toolbar 110
4.3.2. Checkbox column 110
4.3.3. Pin column 111
4.3.4. Device Port Name column 111
4.3.5. Type column 111
4.3.6. I/O column 112
4.3.7. Output column 112
4.3.8. Input column 112
4.3.9. C-Cell (Control Cell) column 112
4.4. Connection Test window 113
4.4.1. Toolbar 113
4.4.2. Net No. column 113
4.4.3. Output Pin List column 113
4.4.4. Input Pin List column 114
4.4.5. Output Value column 114
4.4.6. Expected Value column 114
4.4.7. Input Value column 114
4.4.8. Result column 114
4.5. Edit Device/Manage BSDL dialog box 115
4.5.1. Manufacturer list 115
4.5.2. Device list 115

4.5.3. BSDL file edit 116
4.5.4. OK and Cancel 116
4.6. Set JTAG Clock dialog box 117
4.6.1 Select Clock 117
4.6.2 Low-speed Clock edit box 118
4.7. Script pane window 119
4.7.1. Toolbar 119
4.7.2. File Name edit box 119
4.7.3. Standard I/O edit box 120
4.8. Status bar 121

5. Script 122

5.1. List of Interface Functions 122
5.1.1. Functions related to device setting 122
5.1.2. Functions related to value setting 125
5.1.3. Functions related to connection test setting 128
5.1.4. Functions related to EXTEST setting 132
5.1.5. Functions related to execution of boundary scan test 134
5.1.6. Functions related to operation of the Waveform Display window 136
5.1.7. Functions related to operation in the Pin Status List window 138
5.1.8. Functions related to the Connection Test window 141
5.1.9. Functions related to environment setting 147
5.1.10. Others 148
5.2. Precautions for Use of Interface Functions 149
5.2.1. Calling of functions 149
5.2.2. Specification of a pin name 149
5.2.3. Pin value (data format) 149
5.2.4. Pin value (setting/acquisition by type) 150
5.2.5. Pin value (drive/update timing) 150
5.2.6. Device name 151
5.2.7. Clock setting 151

A. Appendix 152

A.1. Connection and Handling Method of Devices 152
A.1.1. Target not connected with Vref signal 152
A.1.2. When a device not contained in BSDL information is recognized 155
A.2. List of Messages 159
A.2.1. [ERROR] 159
A.2.2. [WARNING] 160
A.3. Troubleshooting 162

A.3.1. Troubles related to installation 162
A.3.2. When Project view is used 162

Revision History 163

Manufacturer Information 164

Notes

Thank you for purchasing the "Universal Probe" from Sohwa & Sophia Technologies. Correctly understand notes and other information in this document before using this product.

1. Permission from the Japanese government is required when exporting the product and technologies described in this document to which the "Foreign Exchange and Foreign Trade Act" applies or bringing them out of Japan.
2. The product described in this document is intended to be used for general electronics such as business equipment, communication equipment, measuring instruments and home electric appliances. Do not use this product in special-purpose equipment such as automobile, railway, marine, or aerospace equipment, transportation equipment, combustion equipment, safety devices, medical equipment, infrastructure equipment, or nuclear power that requires special quality and reliability and of which failure or malfunction may directly threaten human lives or injure human bodies. The customer should be responsible for using this product in such equipment.
3. It is strictly prohibited to reprint this document in whole or in part without our written permission.
4. The contents of this document are subject to change without notice.
5. The specifications in this document may lead to different results depending on your environment or measurement conditions.
6. Note that we bear no responsibility for consequence of operating this product.
7. "Operating Precautions" in this document are important notes to prevent damage on users and third parties as well as assets and to use the product safely and correctly. Be sure to read these precautions before using this product.
8. The product names and trade names in this document are trademarks or registered trademarks of respective owners.

Contact information can be obtained on the website of **Sohwa & Sophia Technologies Inc.**
URL > **http://www.ss-technologies.co.jp/en/index.html**

Operating Precautions

Failure to observe the following precautions may lead to human death or severe injury.

 **Do**	Avoid supplying voltage out of the range specified in the specifications of this product. Supplying the voltage out of the range may cause damage or fire.
 **Do**	When using the target equipped with the ground terminal, ensure that the ground terminals of the target and peripheral equipment are connected. Failure to do so may cause an equipment failure or electric shock. Avoid connecting the ground terminal to the gas pipe. This causes a fire or explosion.
 **Don't**	Do not transport this product with equipment connected. In particular, hold the plug when removing or inserting the cable. Failure to do so may damage the cable, causing a fire or electric shock.
 **Don't**	Observe the following points when handling the cable. Do not damage, process, forcibly bend, twist, pull, putting any object on or heat the cable, moving the cable close to the heating device, or touch the cable with a wet hand. Failure to observe these precautions may cause a fire or electric shock. If the cable is damaged, stop using it.
 **Don't**	When you hear thunders, do not touch the power plug. This causes an electric shock. If the product seems to be damaged by lightning strike, stop using it.
 **Don't**	Do not let a staple, clip or other metal items enter into the product. This may cause a fire or failure.
 **Don't**	Do not use or leave the product in direct sunlight, near heating devices, in an extremely hot or cold environment, under hard vibrations, in dusty area with a large amount of metal dust or oily dust, or noisy area full of spike noise. Do not give strong impact to the product.
 **Do not disassemble**	Do not disassemble, alter or repair the product. This may cause a fire or electric shock.
 **No wetting**	Do not use the product at a place where there is liquid or at a humid place such as bathroom or in vicinity to glasses. This may cause an electric shock. If liquid enters into this product, immediately turn it off and stop using it.
 **Caution**	Touching the energized product for a long time may cause low-temperature burns. Do not use this product covering with comforter or other cloth.
 **Pull out the plug.**	Immediately turn the power off if unusual smell, noise, smoke or fire is detected or if the product is or may be damaged due to a fall or strong shock. Continuing to use it may lead to a serious accident. Stop using the product.

- This software runs using the BSDL file disclosed by each device manufacturer as the information source.
  Note that this software recognizes the device and conducts a boundary scan test assuming that the contents of the BSDL file are correct.
- This software uses the BSDL file Sohwa & Sophia Technologies Inc. obtained in October 2011. Use the latest BSDL file.
- This software does not support the devices of which BSDL file is not disclosed.
- The device may be overloaded and **damaged** depending on the settings during the boundary scan test. Take extreme care when conducting the test.
  Note that Sohwa & Sophia Technologies Inc. bears no responsibility if the device is damaged due to inappropriate settings.

Abbreviations, Terms and Conventions

This section describes the abbreviations, terms and conventions used in this document.

- About numeric values ... All the numeric values are positive unless otherwise specified.
- K (capital letter) ... Represents 2^{10}=1024. (Example: 16K=16384)
- k (small letter) ... Represents 1000. (Example: 1kHz=1000Hz)
- [xxxxx] ... Represents the window title.

The annotations and notes used in this document are as shown in Figure 1.

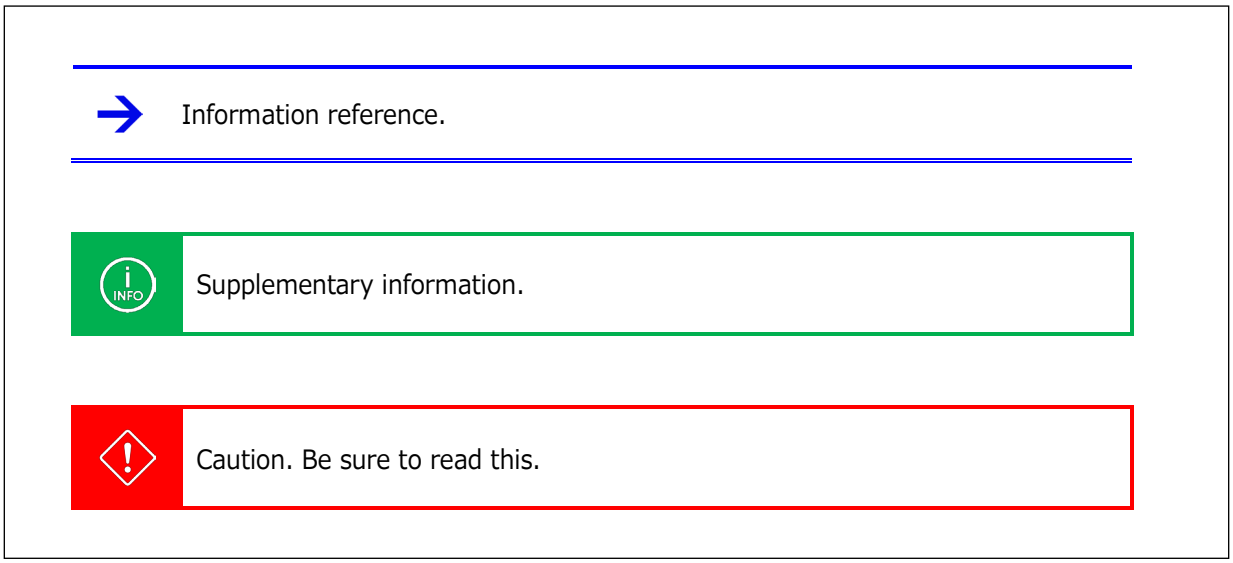


Figure 1

Abbreviations and terms are listed in Table 1.

Table 1

Abbreviations and terms	Description
This product	Universal Probe including accessories.
Probe	Universal Probe itself.
This software	Boundary-scan tool.
Serial No.	Stands for the serial number. Means the Probe ID.
Software Code	License required to issue the License Code for each software. Not required for the free edition.
License Code	Code to add the functions that can be used in this device.
Host PC	This means the PC used for controlling the Universal Probe.
Target	Object to be controlled or measured with the Universal Probe.
(N/A)	Stands for Not Assigned. Indicates that no information is assigned.
TCK	Stands for Test Clock. This is a clock for JTAG interface.
TMS	Stands for Test Mode Select. This controls the state of the connected JTAG TAP controller.
TDI	Stands for Test Data In. This is a data input to the JTAG interface. This is an output from this device.
TDO	Abbreviation of Test Data Out. This is a data output from the JTAG interface. This is an input to this device.
TRST(nTRST)	Stands for Test Reset. This is a signal that resets the JTAG TAP controller. (optional)
SRST(nSRST)	Stands for System Reset. This is usually connected to the main reset of a target board. (optional)
RTCK	Clock output from the connection destination. The debug probe aligns the TCK to this frequency. (optional)
DBGRQ	Stands for Debug Request. This request stops the target from the debug probe and activates the debug status. (optional)
DBGACK	Stands for Debug Acknowledge. This is changed when a target board is stopped by DBGRQ. (optional)
BSDL file	Stands for Boundary Scan Description Language. The BSDL file is usually obtained from the device manufacturer.

1. Overview of This Software

1.1. System Requirements

- PC running Microsoft Windows 7 or later
- CPU: 1GHz or faster (depending on the requirements of the used OS)
- Memory: 1GB or larger (depending on the requirements of the used OS)
- HDD: 500MB or larger free hard disk space
- OS: Windows 7 or later (32bit or 64bit versions are supported)
- One or more empty USB 2.0 ports

1.2. Characteristics

The Boundary-Scan Tool is a tool to check the status of and operate the device pins that use the JTAG mechanism running on Windows 7 or later.

The Boundary-Scan Tool has the following characteristics and functions.

- Waveform display
  - Displays the transition of status of the specified pin in the form of waveform.
  - Samples signals at the minimum interval of 10ms.
  - Saves measured waveforms to a file. Displays saved waveforms in the waveform viewer.
  - Prints measured waveforms.
  - Saves each setting to a file.

- Pin status list
  - Checks the status of the specified pin by using the SAMPLE command in the boundary scan test.
  - Tests relationships between pins by using the EXTEST command in the boundary scan test.
  - Saves the statuses of displayed pins to a CSV file.
  - Saves each setting to a file.

- Connection test
  - Displays the transition of status of the specified pin in the form of waveform.
  - Tests relationships between pins by using the EXTEST command in the boundary scan test. This is the function more specialized for relationships between pins than the pin status list.
  - Saves the pin connection settings to a file.
  - Saves the values output for connection test and their expected values to a file.

- Script execution
  - Displays the transition of status of the specified pin in the form of waveform.
  - Executes a variety of functions in this software by using Python scripts and interface functions.
  - Executes scripts from the command line or from a script file in batch.
  - Saves execution results to two types (text file and Python script file) of log files.
  - Saves the information on the used script file to a file.

1.3. Turning on or off the Power

1.3.1. Connecting to the Target and Turning on the Power

Connect the probe and the target with the following procedure:

1) Confirm that the target is turned off.
2) Connect the USB cable of the probe.
3) Connect the target and the probe.
4) Turn on the target.
5) Start this software and perform functions such as writing data.

1.3.2. Turning off the Power and Disconnecting from the Target

Disconnect the probe from the target with the following procedure:

1) Exit this software.
2) Turn off the target.
3) Remove the probe from the target.

→ For details on the hardware specifications of probe, target restrictions, connection and other items, refer to the “Hardware Users Manual.”

2. How to Operate

2.1. Workflow

The main process from starting the Boundary-Scan Tool, writing the user module, through saving the project file is shown below.

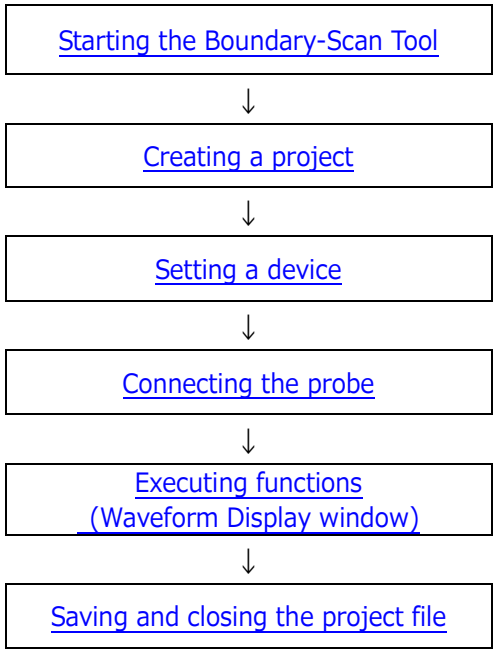


Figure 2

→ Executing functions has **"Project View"**, **"Waveform Display window"**, **"Pin Status List window"** and **"Connection Test window"**. As an example, shows "Waveform Display window" in the above flow.

2.1.1. Starting the Boundary-Scan Tool

Icon	Operations in Windows
	[Start] -> [All Program] -> [Universal Probe] -> [Boundary- Scan Tool]

The window shown in the figure below opens immediately after this software starts up.

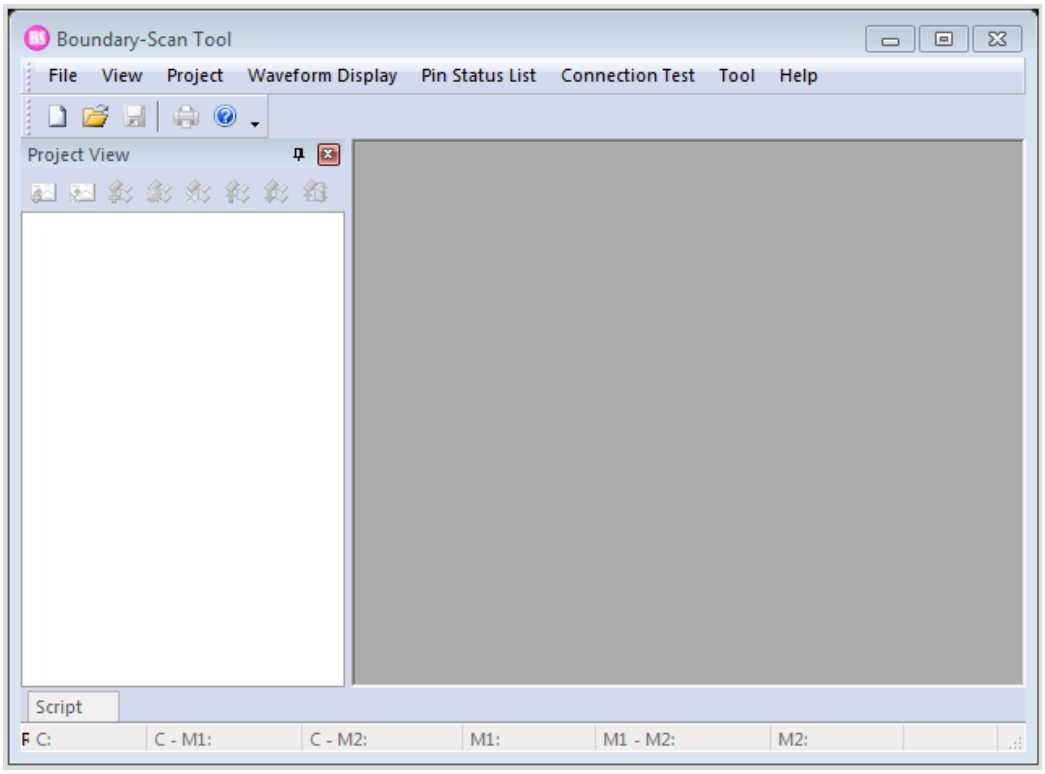


Figure 3

→ For detailed operation procedures of each function, refer to "Project View," "Waveform Display window," "Pin Status List window," "Connection Test window" and "Script Pane."

2.1.2. Operation procedures in the Project view

The main operation procedures in the Project view are listed below.

Table 2

	(1) **Creating or opening a project** Create an empty project and specify a variety of settings. Or, open the already setup project.
	((2) **Manual setting of a device (optional))** Configure the JTAG chain by using Add Device, Delete Device, Move Device Upward, Move Device Downward or Device Properties.
	(3) **Connecting the probe (automatic device recognition)** Start the probe and make it ready for operation. The software also tries to automatically recognize the device at the same time.
	((4) **Manual setting of a device (optional)**) Configure the JTAG chain if automatic recognition failed, the BSDL information was incorrect, or due to other reasons.
	(5) **Setting the JTAG clock** Set the clock frequency to conduct a boundary scan test.
	(6) **Consistency test** Conduct a test to see if the configured chain is proper, if the JTAG chain has been changed after the probe was connected.
	(7) **Grouping ports (optional)** Group multiple ports.
	(8) **Opening other windows** Open the Waveform Display window or the Pin Status List window and conduct a boundary scan test. **Waveform Display window** **Pin Status List window**
	(9) **Disconnecting the probe** Disconnect the probe.
	(10) **Saving and closing a project** Save a variety of settings in a project. Close the project for which test is complete.

2.1.3. Operation procedures in the Waveform Display window

The main operation procedures in the Waveform Display window are listed below.

Table 3

	(1) **Displaying the Waveform Display window** Create a new waveform display pattern or select an existing waveform display pattern and open the Waveform Display window. Saved waveforms can be also opened and displayed.
	(2) **Setting measurement conditions** Set the following sampling conditions before starting measurement. **Port group operations** **Sampling period selection**
	(3) **Waveform measurement** Sample data at the specified period and display the waveform.
	(4) **Verifying the waveform** Stop sampling or open a saved waveform and verify the waveform.
	(5) **Saving the waveform** Save the waveform to a file for later verification.

2.1.4. Operation procedures in the Pin Status List window

The main operation procedures in the Pin Status List window are listed below.

Table 4

	(1) **Opening the Pin Status List window** Create an empty pin status list and specify a variety of settings. Or, open the already setup list.
	(2) **Setting a port group** Configure the listed ports by using Add Port/Group, Delete Port/Group, Move Port/Group Upward, or Move Port/Group Downward.
	(3) **Conducting a boundary scan test** Conduct a boundary scan test by using the SAMPLE and EXTEST commands.
	(4) **Saving the current status to a CSV file** Save the current status displayed in the Pin Status List window to a CSV file.

2.1.5. Operation procedures in the Connection Test window

The main operation procedures in the Connection Test window are listed below.

Table 5

	(1) **Opening the Connection Test window** Create an empty connection test list and specify a variety of settings. Or, open the already setup list.
	(2) **Setting a pin connection (net)** Set output and input ports to create a "net list." A new net list can also be created from a previously created net list file.
	(3) **Conducting a connection test** Create a test pattern and conduct a connection test by using the EXTEST command in the boundary scan test.
	(4) **Saving connection settings and result** Save the net list and connection test result to a file.

2.1.6. Operation procedures in the Script pane

The main operation procedures in the Script pane are listed below.

Table 6

	(1) **Selecting a script file** Select the `Python` script file to be executed in batch.
	(2) **Executing a script file** Execute the script file selected at (1) to execute a variety of functions in this software in batch.
	(3) **Using the command line for execution** Enter a `Python` command line in the Standard I/O edit box for execution.
	(4) **Saving and clearing the history** Save the batch execution and command line execution history to a text file or `Python` script file, or clear the history.

2.2. Project View

2.2.1. Create a new project

Creates a new project.
When using the product for the first time, select the following tool button or menu item to create a project.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Create New Project**	Ctrl + N

Executing the above menu item creates an untitled project in the Project view.

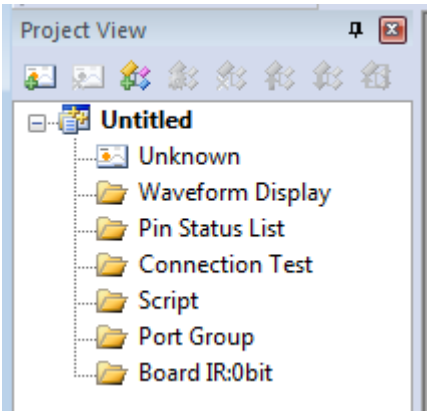


Figure 4

2.2.2. Open a project

Opens a previously created project file. The settings when you stopped the work lastly are restored.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Open Project**	Ctrl + O

Executing the above menu item displays the **Select File** dialog box. Select the project file you want to open and select **OK**. Information recorded in the project file is restored in the Project view.

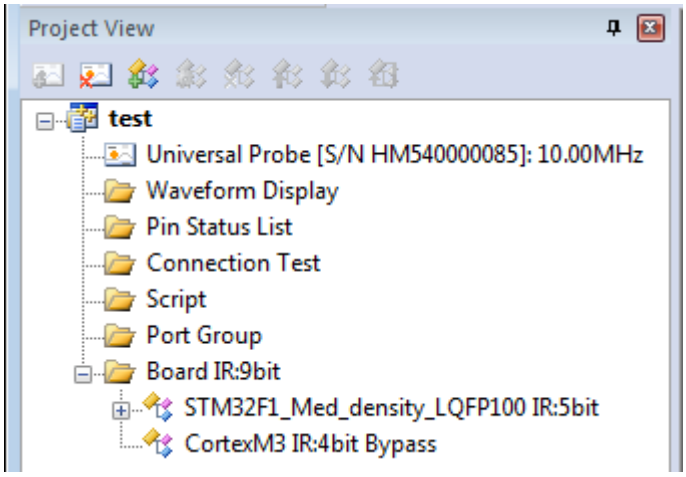


Figure 5

2.2.3. Save a project

Saves a variety of information currently set to the project file.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Project** **File** -> **Save Project as**	Ctrl + S

Executing the above menu item saves a variety of project settings to a file. If "Save" or "Save as" is selected when there is no file for the project to be saved, the **Select File** dialog box is displayed. Specify the saving destination and file name and select **Save** to save project settings to a file.

The extension of a project file is .bss.

The project file contains the following information:

- Individual ID of probe to be connected and clock frequency for boundary scan test (JTAG clock)
- Target device configuration for boundary scan test
- Port information registered in the Waveform Display window (waveform display pattern)
- Port information registered in the Pin Status List window (pin status list pattern) and the check status of each port
- Check status of **"Project >> Forced Output of JTAG Signal"** menu item

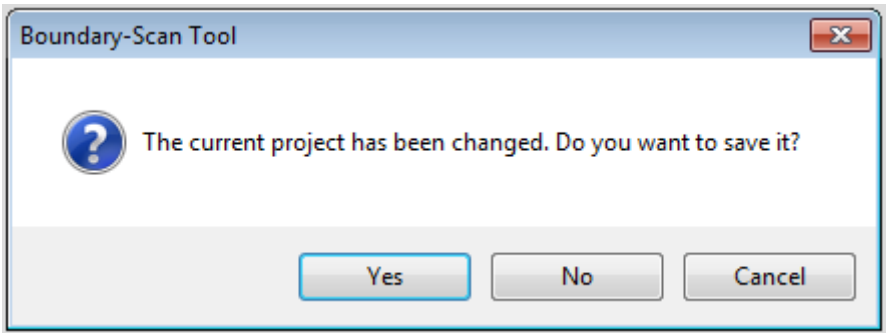
2.2.4. Close a project

Closes the currently open project.

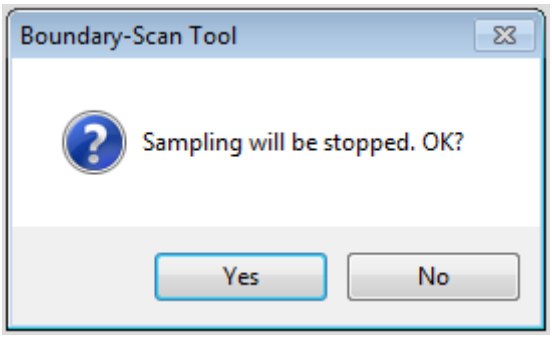
Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Close Project**	---

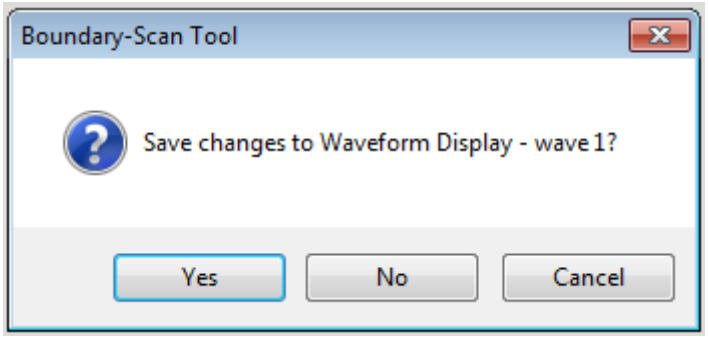
Executing the above menu item closes the currently open project and removes it from the Project view. The project is disconnected from the probe at this time if connected.

When the Create, Open or Close menu item is executed if a change is made to the already created project (project having an asterisk (*) at the right of its name), a message appears to ask if the currently open project is to be saved or not. Select **"Yes," "No" or "Cancel"** as needed and proceed to the next step.



If sampling is being performed in the Waveform Display window when the Create, Open or Close menu item is executed for a project, messages appear to ask **whether to stop measurement** and **whether to save the waveform**. Select **"Yes," "No" or "Cancel"** as needed and proceed to the next step.





2.2.5. Add a device

Set a device to the project and configure the boundary scan chain.
Use this menu item if automatic recognition cannot be performed because there is no target at hand, BSDL contains errors, or for other reasons.

Tool button	Operation on the menu bar	Shortcut key
	**Project** -> **Add Device** * Alternatively, Project View -> Board, right-click to open the context menu.	---

The device is added under the **board** tree in the Project view. **Unknown** device is added as the initial value.
Every time this menu item is selected, one device is added to the end of the board tree.

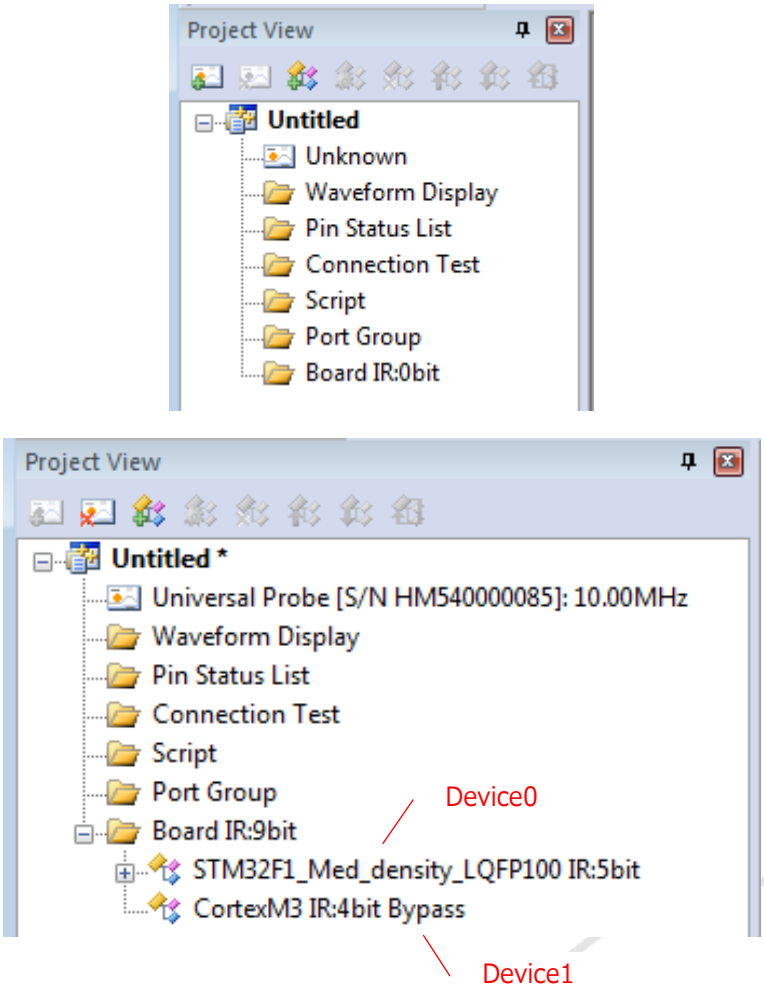
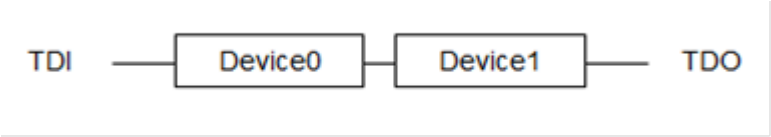


Figure 6

The device at the top of the board tree will become the closest one to the TDI of the probe.
The chain number of the device at the top of the board tree is always "0" and the device is displayed as "Dev0" in the Waveform Display window or the Pin Status List window.
Devices are sequentially numbered from the one closest to the TDI as shown in the figure below.



When registering devices in the Project view, register the closest one to the TDI first, and the farthest one last.
If the display order of devices in the Project view is different from the actual connection order, adjust the device positions with “Move Upward” or “Move Downward.”

2.2.6. Edit a device

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Project View -> Board -> Any device, right-click to open the context menu -> Edit Device	---

The **Edit Device** dialog box is displayed.
When you select a manufacturer in the manufacturer list, the devices registered in this software are listed. Select a device suitable for your target board.
If the suitable device is not included in the list, you can add the device to the list by loading the BSDL file.

Directly enter the path to the BSDL file in the **BSDL edit box** or click the [...] button and select the file.
Clicking the [Read] button loads the specified BSDL file and adds the device to the list.

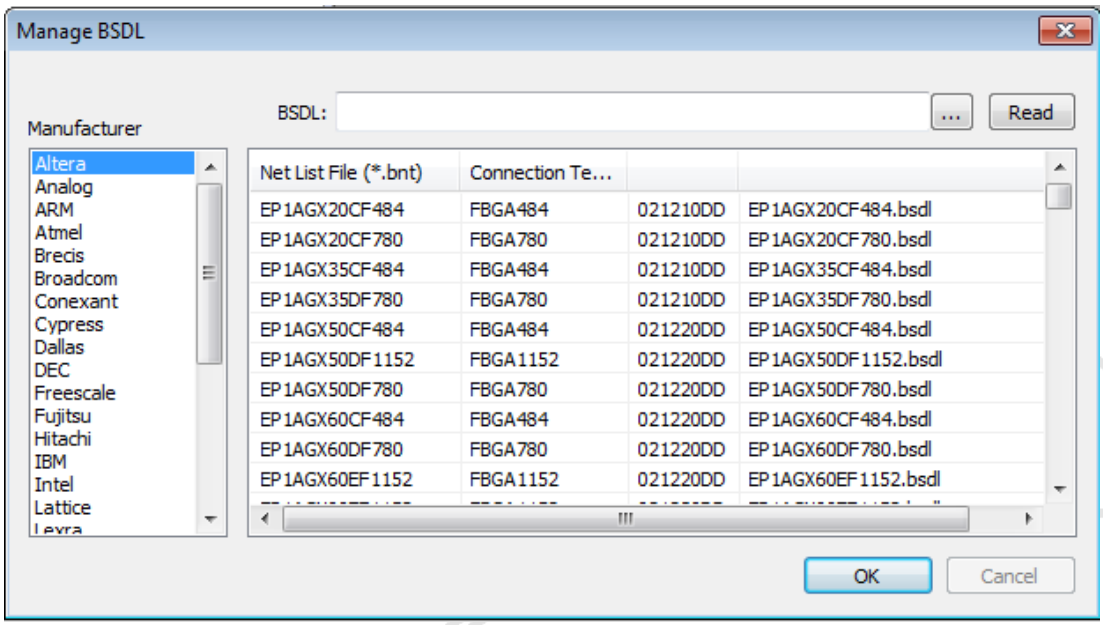


Figure 7

2.2.7. Delete a device

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Project View -> Board -> Any device, right-click to open the context menu -> Delete Device	---

The selected device is deleted from the tree in the Project view.

2.2.8. Move a device upward

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Project View -> Board -> Any device, right-click to open the context menu -> Move Device Upward	---

The selected device is switched with the device above it.
Note that **the chain number of each device is also switched**.

2.2.9. Move a device downward

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Project View -> Board -> Any device, right-click to open the context menu -> Move Device Downward	---

The selected device is switched with the device below it.
Note that **the chain number of each device is also switched**.

2.2.10. Device properties

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Any device, right-click to open the context menu -> Device Properties	---

Displays the properties of the selected device.
The bit length and bypass setting of the IR register in the device can also be edited.

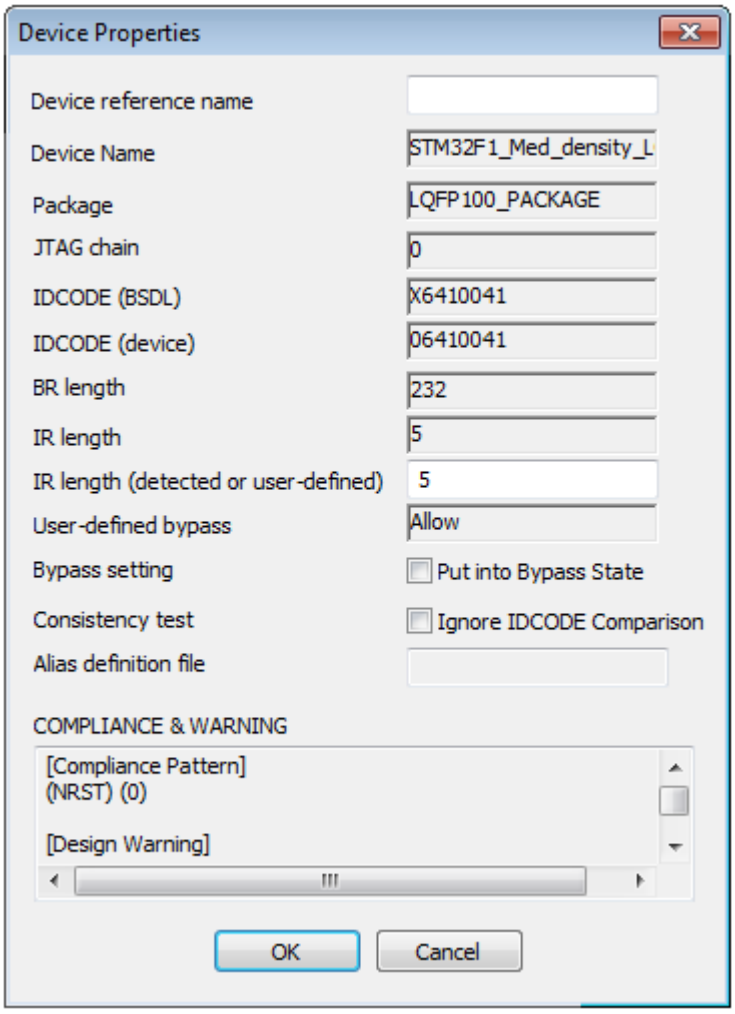


Figure 8

Table 7

Device reference name	Device name that the user can independently set. The initial value is "."
Device name	Device name. The initial value is "Unknown." This is acquired from the BSDL.
Package	Device package information. The initial value is "." This is acquired from the BSDL.
JTAG chain	Indicates the position (order) of the device in the boundary scan chain. The upper in the Project view tree is the device, the smaller is the number. The minimum value is "0."
IDCODE(BSDL)	Device-specific ID that can be acquired by using the boundary scan IDCODE command. The initial value is "." This is acquired from the BSDL.
IDCODE (device)	Device-specific ID that can be acquired by using the boundary scan IDCODE command. The initial value is "00000000." This is acquired by issuing the IDCODE command to the actual device. If the device implements the IDCODE command, it is updated when the consistency test is successful.
BR length	Bit length of device boundary scan register. The initial value is "0." This is acquired from the BSDL.

IR length (BSDL)	Bit length of device IR register. The initial value is "0." This is acquired from the BSDL.
IR length (detected or user-defined)	Bit length of device IR register. The initial value is "0." Any value can be set. This value is updated when the automatic device recognition is successful. If the consistency test fails even though the boundary scan device configuration is appropriate, the device may pass the test after this item is changed.
User-defined bypass setting	Indicates whether the bypass setting check described below is enabled or disabled. This is "Prohibited" if the BSDL does not contain port information. When this is "Prohibited," the device is forcibly set to the bypass status.
Bypass setting	When this checkbox is checked, the BYPASS command makes the device skip the boundary scan test.
Consistency test	When this checkbox is checkmarked, the consistency test is conducted without comparing the IDCODE read out of the device by using the IDCODE command and the IDCODE in the BSDL. Checkmark this checkbox if the information in the BSDL is different from the value read by using the IDCODE command.
Alias definition file	File name selected in the “Change Port Name” menu item.
COMPLIANCE & WORNING	The COMPLIANCE and WARNING information in the BSDL is displayed. Device-specific rule such as “Port N must output Low” is displayed.

2.2.11. Connect a probe

Tool button	Operation on the menu bar	Shortcut key
	**Project -> Connect**	Ctrl + C

Connect between this software and the probe. The software tries to automatically recognize the target boundary scan chain at connection time.

2.2.12. Select a probe

When a single probe is connected to the PC when you connect a probe, that probe is connected.
When two or more probes are connected, the Select Probe dialog box is displayed. Select the probe to use.

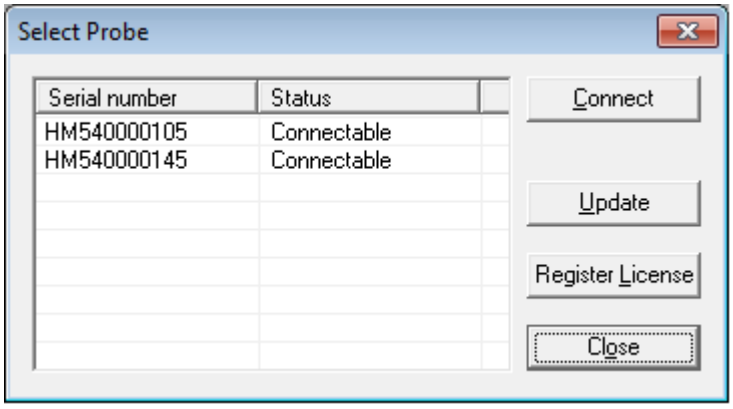


Figure 9

Click the probe to be used and click the Connect button.

Table 8

Serial number	Displays the serial number registered for the probe.	
Status	Displays the status of probe.	
	Connectable :	The probe can be connected to this software and used.
	Currently used :	The probe is already used in another application. This software cannot use this probe.
	License not registered :	The license is not registered. To use this probe, click the Register License button to register the license.
Connect	Connects to the selected probe. This button is enabled only when the probe of which status is “Connectable” is selected.	
Update	Searches for the currently connected probes again and updates the probe list.	

Register License	Displays the dialog box to register the license. This button is enabled when the probe of which status is “Connectable” or “License not registered” is selected. → For details, refer to "**Obtain the probe license**"
Close	Closes this dialog box.

2.2.13. Probe communication (automatic device recognition)

After connecting to the probe, this software tries to automatically recognize the target boundary scan chain.

When the target is connected, the software recognizes the number of devices and tries to acquire the BSDL information on each device.

If there are multiple candidate packages for a recognized device, the Select Device Package dialog box is displayed. Select the appropriate package.

If automatic recognition is successful, the device is registered under the "Board" tree in the Project view.

If automatic recognition fails, configure the boundary scan chain with “2.2.5. Add a device”.

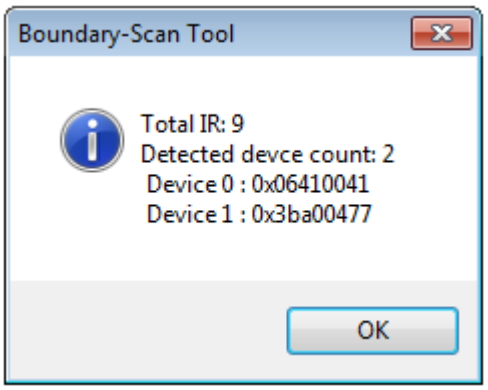


Figure 10

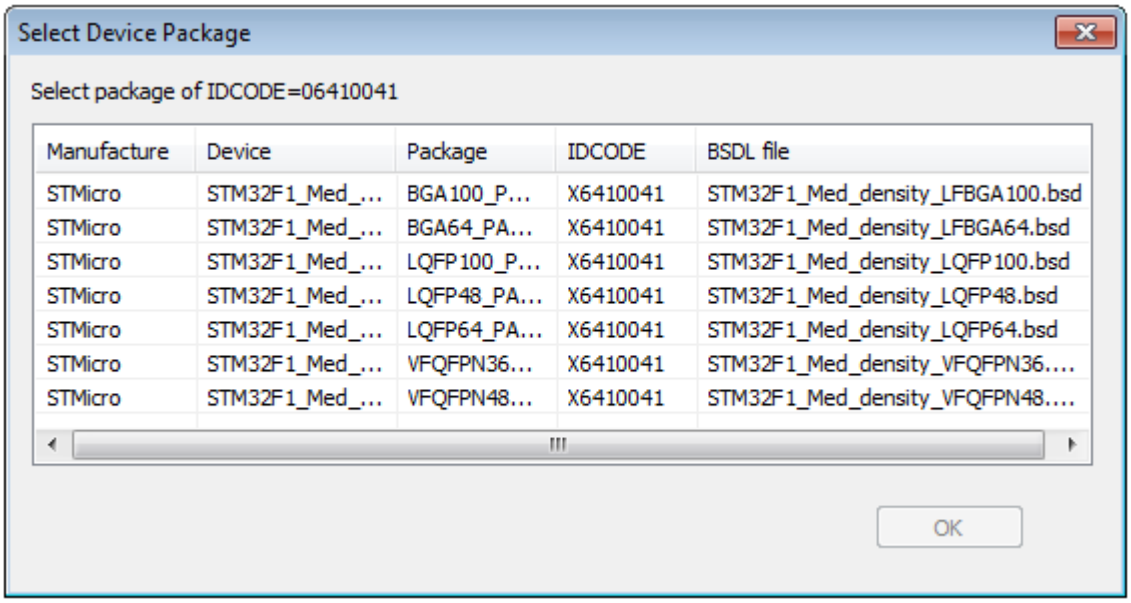


Figure 11

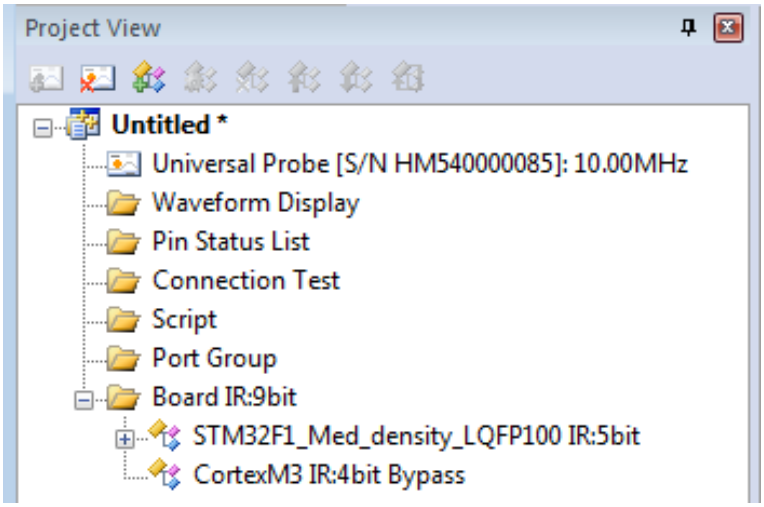
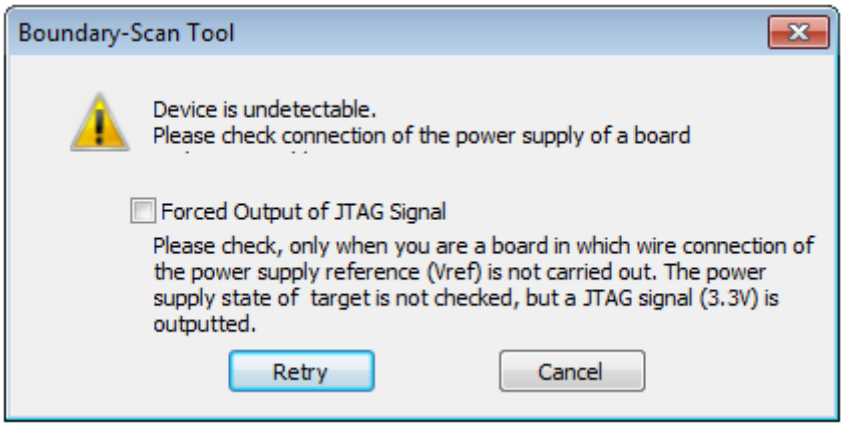


Figure 12

INFO If device configuration is already completed with "Manual setting of a device" or "Open Project File", the consistency test is conducted first. If the test fails, the software tries to automatically recognize the device again.

INFO If automatic recognition fails because the target power supply cannot be detected, a message box appears which indicates that **the device cannot be detected** and asks **whether to output the JTAG signal by ignoring the target power supply detection**. If the target which the power supply reference is not wired to is connected to the probe, checkmark and click Retry. If the target is not connected, connect the target and retry the operation or cancel the connection process with the Cancel button.



The voltage that should be output as the High level of JTAG signal is normally detected from the Vref signal. However, 3.3V is output as the High level when this is checkmarked.

2.2.14. Disconnect a probe

Tool button	Operation on the menu bar	Shortcut key
	**Project -> Disconnect**	Ctrl+Shift+C

Disconnect the probe from this software. A message asking whether to clear probe information from the project is displayed. Select “Yes,” “No” or “Cancel” as needed.

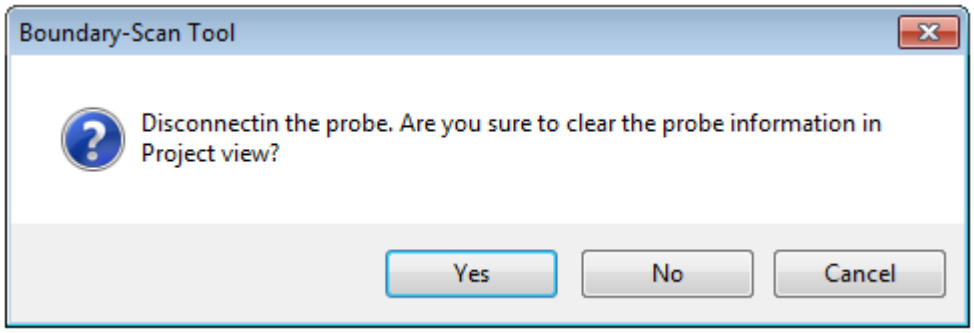
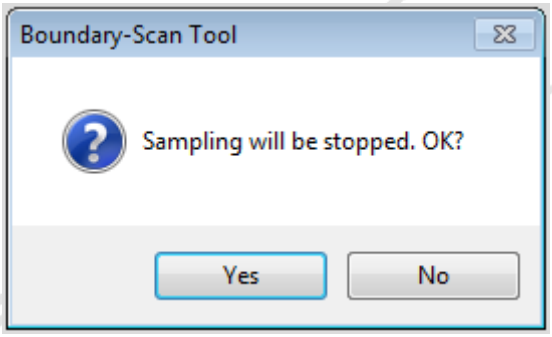
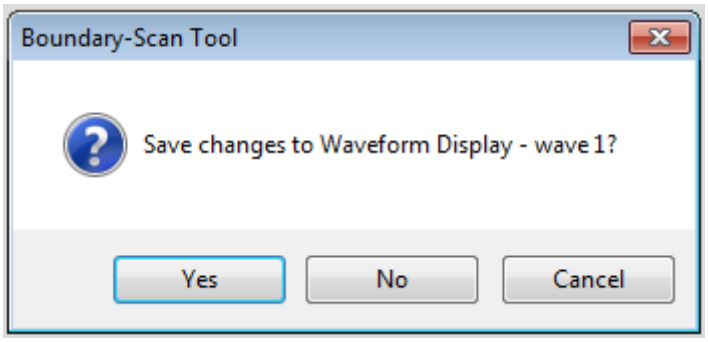


Figure 13

If sampling is being performed in the Waveform Display window when the Disconnect menu item is executed for a probe, messages appear to ask **whether to stop measurement** and **whether to save the waveform**. Select **“Yes,” “No” or “Cancel”** as needed and proceed to the next step.





2.2.15. Set a JTAG clock

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Set JTAG Clock** * Alternatively, Project View -> Probe, right-click to open the context menu.	---

Set a clock frequency between the probe and the target device when executing a boundary scan test. Selecting the above menu item displays the "Set JTAG Clock" dialog box. Set the clock frequency suitable for the target.

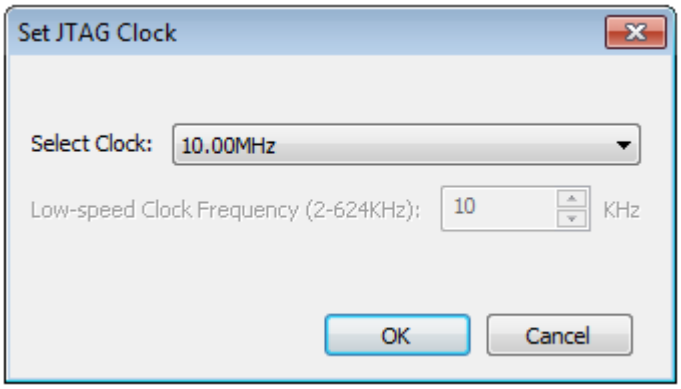


Figure 14

This software automatically recognizes the device and executes the consistency test at **"Low-speed Clock"** and **"10kHz"**. Then, increase the JTAG clock with **Auto Setting**.

2.2.16. Select a JTAG clock

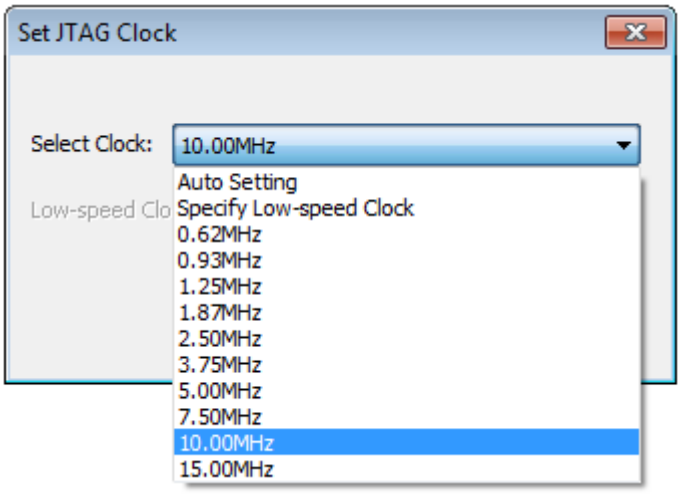


Figure 15

The frequencies that can be set for the probe are displayed in the pull-down list.

The initial value for project is "Specify Low-speed Clock."

Details of each item are as shown below.

Table 9

Auto Setting	Automatically sets the clock frequency. If the target that can be automatically recognized is connected or the device is set to the project, select the lowest frequency from the BSDL information on each device and set the highest frequency that is slower than that value and can be selected from the pull-down list. INFO For example, assume that there are devices A and B and 12MHz and 20MHz are included in the BSDL for each of them. In this case, 10MHz, which is the maximum clock frequency slower than 12MHz, is selected from the pull-down list in the above figure.
Specify a Low-speed Clock	Set a frequency slower than 625kHz in the Low-Speed Clock Frequency edit box below the pull-down list. INFO The initial value for the project is "Specify Low-speed Clock" and "10kHz."
Other frequencies	Internally conducts the communication test with the target at the selected frequency. If it works, set to that frequency. Otherwise, decrease the frequency until it works.

Because this value is set depending on whether it works or not regardless of BSDL information, a frequency faster than the operating frequency specified in the BSDL may be set.

2.2.17. Set a JTAG clock (low speed)

Perform JTAG communication with the probe and target device at a lower frequency. **"Specify Low-speed Clock"** from the Select Clock pull-down list enables this edit box. The value range is from 2 to 624 (unit: kHz).

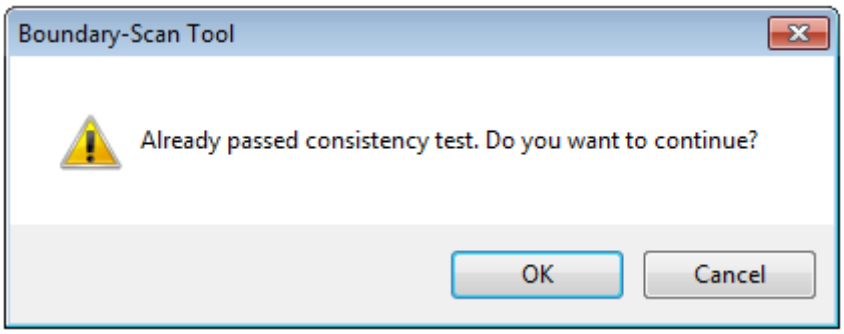
2.2.18. Consistency test

Tool button	Operation on the menu bar	Shortcut key
	**Project -> Consistency Test** * Project View -> Board -> Any device, right-click to open the context menu.	---

Conduct a test to check whether the boundary scan test can be performed by using the boundary scan chain configured in the Project view.
When the test is successful, the above menu item is grayed out and the following functions become usable.

- Waveform Display window
- Pin Status List window
- Connection Test window

If the "automatic device recognition" is successful, the consistency test is considered to be passed and the above menu item is grayed out.
If you try to execute "Add Device," "Edit Device," "Delete Device," "Move Device Upward," or "Move Device Downward" menu item after the consistency test is passed, the following warning message is displayed. The consistency test may fail if a project side device's configuration has been changed.



2.2.19. Add a port group

You can create groups in the “port group” tree in the Project view. Created groups can be used in the Waveform Display window, Pin Status List window or other windows.

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Add Port Group** * Alternatively, Project View -> Port group, right-click to open the context menu.	Ctrl + G

The group can be created by specifying a combination from all the ports on the board. Once created, the group can be used in the Waveform Display window or the Pin Status List window.

• **Create Group dialog box**

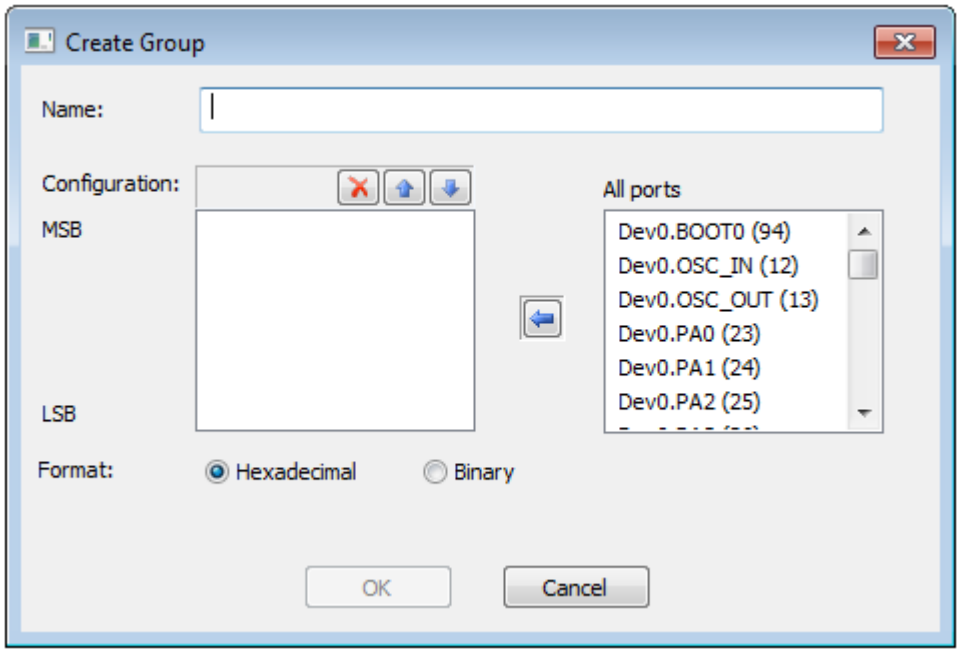


Figure 16

• **Specifying a name**
Specify a group name in “Name.”

• **Specifying a port to be configured**
All the ports on the board are listed in the “All ports” list. Select the port you want to add to the group from the list and click the [←] button. The group is added to the end of the “Configuration” list.

• **Changing the order of ports to be configured**
To change the order of a port in the "Configuration" list, select it from the list and change the order by clicking the [↑] or [↓] button.

• **Deleting a port to be configured**
To delete a port from the "Configuration" list, select it and click the [X] button.

• **Specifying a format in which values are displayed**
The format to display the values in the group can be specified by clicking “Hexadecimal” or “Binary.” This is used as the default display format to display values in the Waveform Display window or the Pin Status List window.

• **Adding a Group**
Click the [OK] button to add a group. The group is added to the end.

2.2.20. Edit a port group

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Port group -> Added port group, right-click to open the context menu -> Edit Group	---

The name, port configuration, and display format of the group added with “Add Group” can be changed. Selecting the above menu item opens the create group menu. Edit the group with the same procedure as “Add Group.”

2.2.21. Delete a port group

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Port group -> Added port group, right-click to open the context menu -> Delete	---

The group selected from the Project view can be deleted.

2.2.22. Set a device reference name

A device reference name can be set to a device. The set name can be used in the Connection Test window.

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added device, right-click to open the context menu -> Set Device Reference Name	---

The Set Device Reference Name dialog box opens. Enter a device reference name and click the [OK] button.

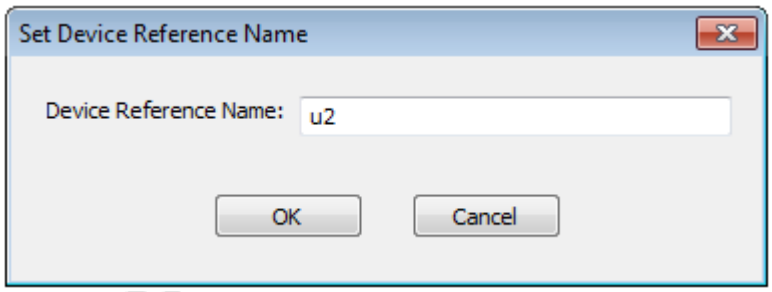


Figure 17

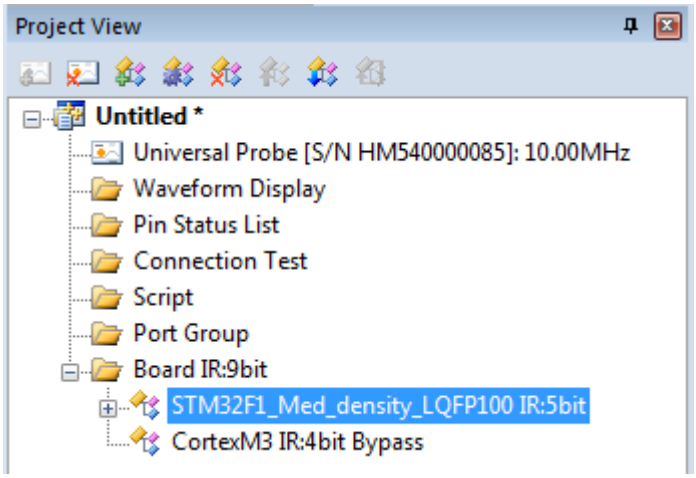


Figure 18

The device reference name can also be changed with the procedure below.

Project View -> Board → Added device, right-click to open the context menu. →Device Properties

The Device Properties dialog box opens. Edit the device reference name and click the [OK] button.

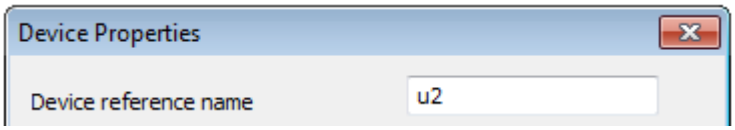


Figure 19

2.2.23. Reset a device reference name

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added device, right-click to open the context menu -> Reset Device Reference Name	---

The set device reference name is cleared.

2.2.24. Change a port name (per device)

The port name can be changed per device or per port to facilitate port identification. In addition, the changed port name can be saved.

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added device, right-click to open the context menu -> Change Port Name	---

The Change Port Name dialog box opens. Enter the name of the file where the port name is defined in the definition file field and click the [OK] button.

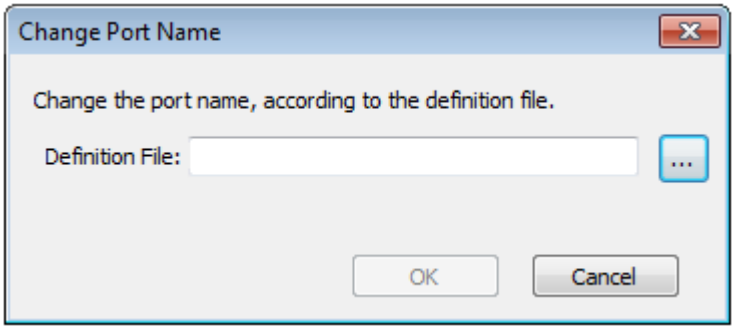


Figure 20

The port name definition file has the following format.

Table 10

[PortName]	List of pin names for each port and port names. The list consists of combinations of "Pin name, Port name." Example: [PortName] 23 , INT0 30 , INT2 32 , INT4 35 , INT3

Note that all the device port names you have tried to change are reset if the port name definition file contains errors.

2.2.25. Change a Port Name (per port)

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added device -> Port, right-click to open the context menu -> Change Port Name	---

The Change Port Name dialog box opens. Enter a port name in the port name field and click the [OK] button.

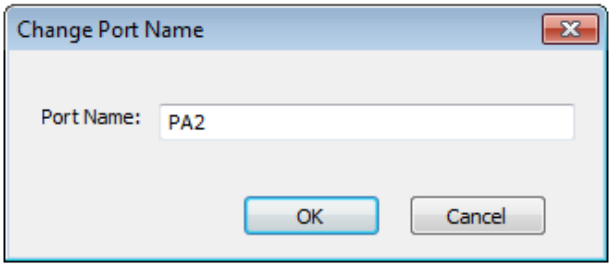


Figure 21

2.2.26. Reset a port name (per device)

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added device, right-click to open the context menu -> Reset Port Name	---

Changed port names are cleared per device. The port name is defined in the BSDL file.

2.2.27. Reset a port name (per port)

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added device -> Port, right-click to open the context menu -> Reset Port Name	---

Changed port names are cleared. The port name is defined in the BSDL file.

2.2.28. Save a changed port name

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added device, right-click to open the context menu -> Save Changed Port Name	---

The Save Changed Port Name dialog box opens. Enter a file name in the definition file field and click the [OK] button.

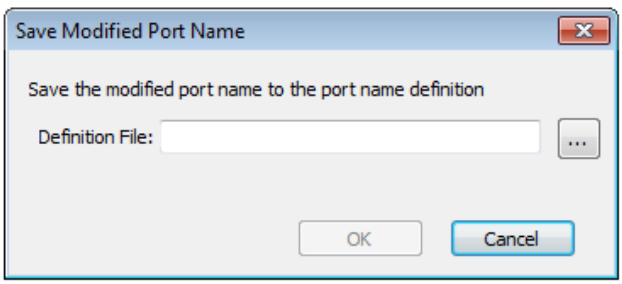


Figure 22

2.3. Waveform Display window

2.3.1. Create a new waveform window

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Create New Waveform window** *Project View -> Waveform Display -> Right-click to open the context menu -> Waveform Display	Ctrl + W

Executing the above menu item adds **"wave n"** to the Project view and opens the Waveform Display window ("n" represents a serial number starting with "1"). Every time the above menu item is executed, a new empty waveform named "wave n" is added and displayed in the window. However, if the above menu item is executed with "wave n" selected in the Project view, no new waveform is created but the selected waveform is displayed in the window. The name of the current list appears on the title bar of the Waveform Display window.

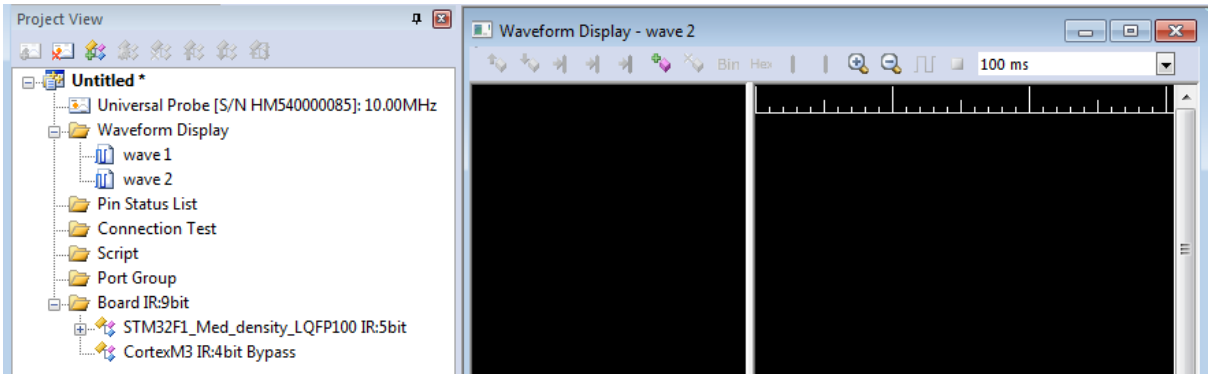


Figure 23

2.3.2. Open a waveform in viewer

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Open Waveform in Viewer**	---

Executing the above menu item displays the Select File dialog box. When you select the saved waveform file (with an extension of .bsw), that waveform is opened in the Waveform window. When a waveform is opened, the following function cannot be used.

- Add a Port
- Create a Group
- Delete a Port Group
- Start Sampling
- Stop Sampling
- Specify a Sampling Interval

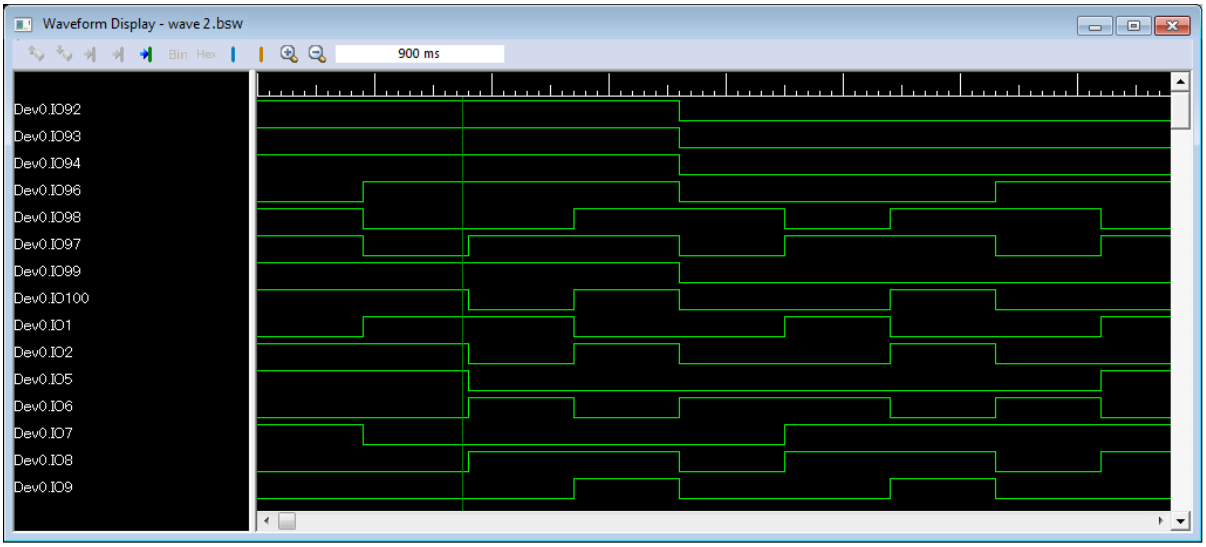


Figure 24

2.3.3. Rename a waveform

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added [wave n] Right-click to open the context menu -> Rename *Click the mouse or press the F2 key while selecting [wave n].	---

Selecting the above menu item makes the relevant items editable in the Project View. Change the name according to the purpose.

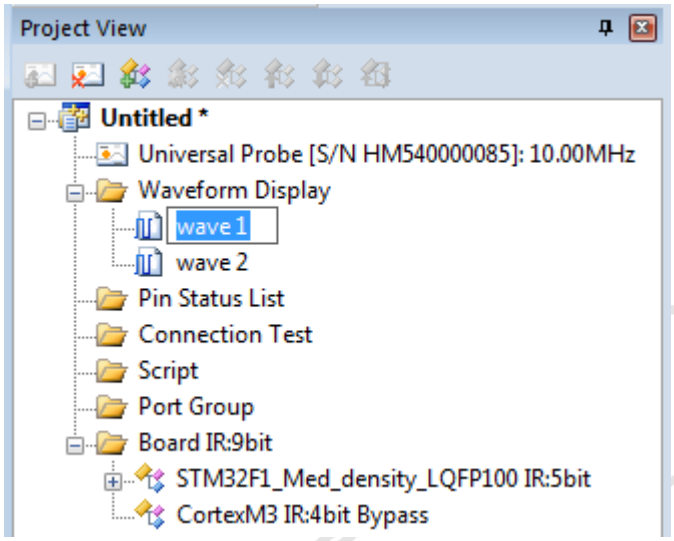


Figure 25

2.3.4. Delete a waveform

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> Added [wave n], right-click to open the context menu -> Delete	---

Selecting the above menu item deletes the relevant items from the Project view.

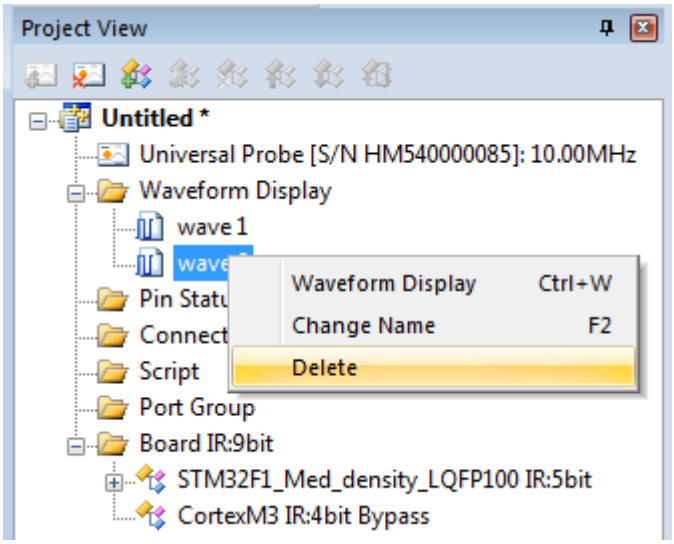
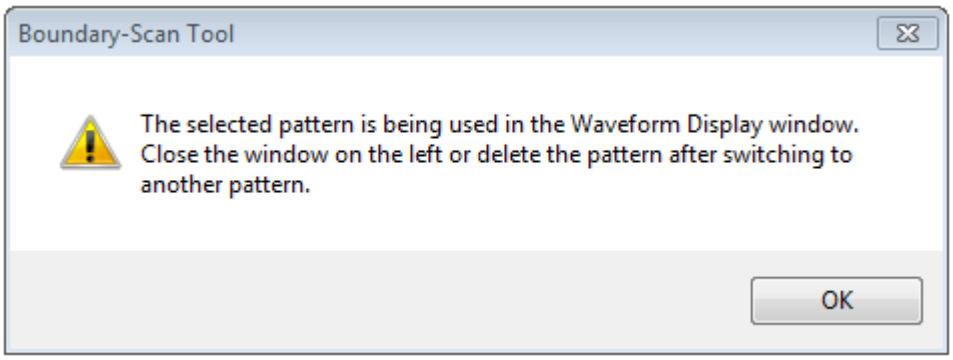


Figure 26

If the list displayed when selecting the above menu item coincides with the one displayed in the Waveform Display window, the following error message appears.



2.3.5. Add a port (signal)

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Port/port group name display area -> Right-click to open the context menu -> Add	---

Selecting the above menu item displays the **"Add Port"** dialog box. In the dialog box, all the ports of all the devices registered in the project are listed. Checkmark the desired port(s) to check the status and click the **"OK"** button. Multiple ports can be selected.

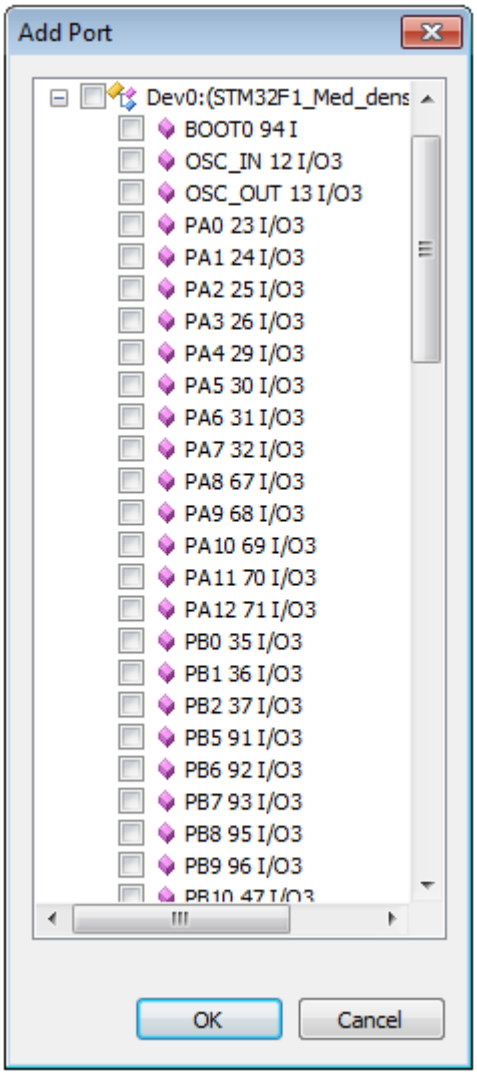


Figure 27

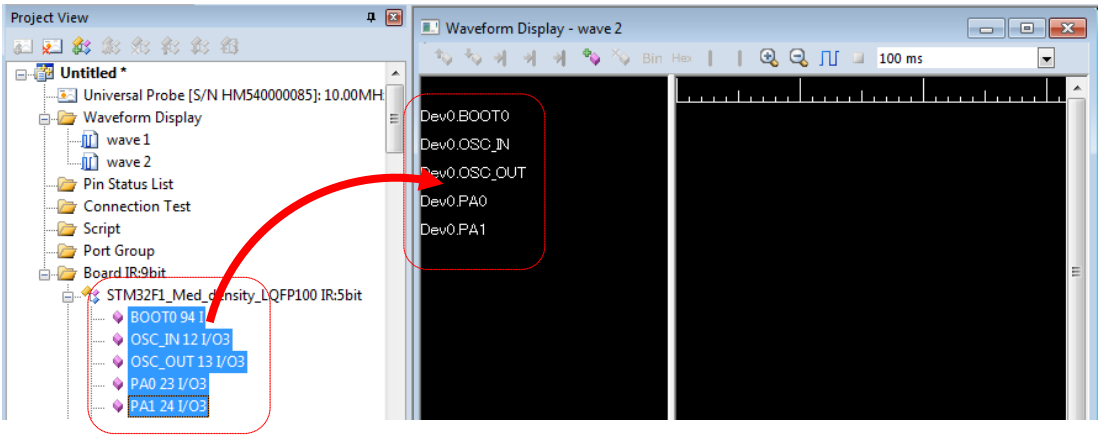
Clicking the checkbox for the **"Board"** tag switches all the ports of all the devices between selected and deselected. Clicking the checkbox for the **"Dev n"** tag ("n" represents a serial number in the boundary scan chain) switches all the ports of that device between selected and deselected.

Some ports share the control cell.

(Cells whose numbers' BSDL "function" is "control" are shared among two or more ports so that the I/O direction and whether to permit output are collectively managed).

When one of these ports is checkmarked, other ports that share the same cell are also checkmarked.

A port can also be added by dragging and dropping it in the Pin Status List window from the Project view. Multiple items can be selected in the Project view by **left-clicking the mouse while holding down the CTRL or SHIFT key**. If a port that shares the control cell is added, other ports that share the control cell are also added.



2.3.6. Create a port group (signal)

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. *Waveform Display window -> Port/port group name display area -> Right-click to open the context menu -> Create Group	---

The group can be created by specifying a combination of added ports.
Once a group is created, values can be specified and displayed in binary or hexadecimal notation.
The group created with this menu item is not added to the Project view.

For how to create a port group in the Create Group dialog box, refer to **2.2.19. Add a port** group.

The port group name display area displays the device reference name and port name for each port as shown in the figure below. Only the group name is displayed for the group.

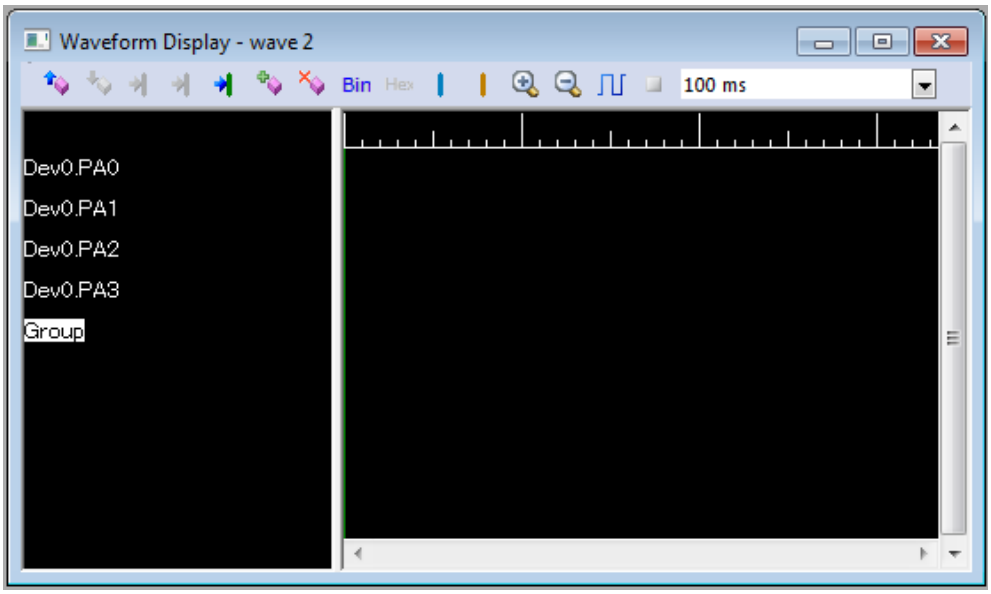


Figure 28

2.3.7. Switch the port group (signal) display between binary and hexadecimal notation

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Select a port group -> Right-click to open the context menu -> "Display in Binary Notation"	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Select a port group -> Right-click to open the context menu -> "Display in Hexadecimal Notation"	---

Selecting the above menu item switches whether to display the values in the selected group between binary and hexadecimal notations.

2.3.8. Change the display order of ports/port groups (signal)

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Select a port/port group -> Right-click to open the context menu -> “Up”	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Select a port/port group -> Right-click to open the context menu -> “Down”	---

A port or group can be moved by selecting it in the port group name display area.

2.3.9. Delete the ports/port groups (signal)

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Select a port/port group -> Right-click to open the context menu -> Delete	---

A port or group can be deleted by selecting it in the port group name display area.

2.3.10. Select a sampling period

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. *Waveform Display window -> Toolbar -> Sampling period specification combo box	---

Sampling is performed in the Waveform Display window per each selected sampling period.

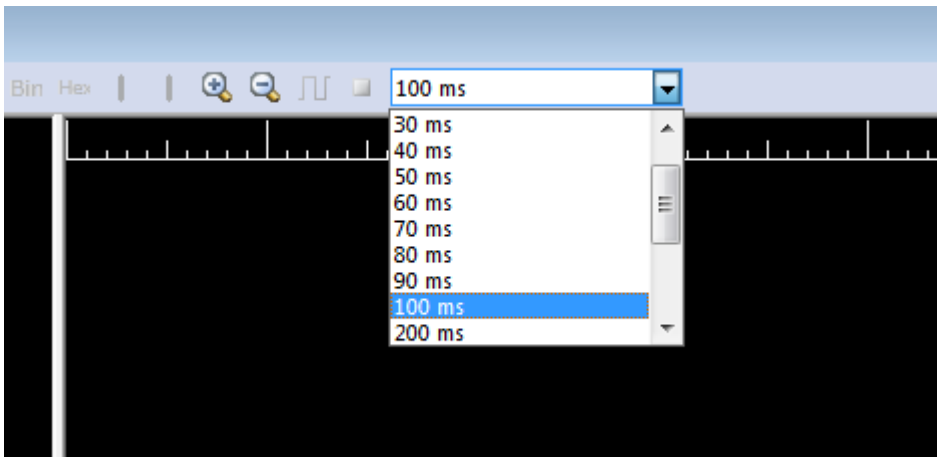


Figure 29

The following sampling periods can be selected:

- 10ms
- 20ms
- 30ms
- 40ms
- 50ms
- 60ms
- 70ms
- 80ms
- 90ms
- 100ms
- 200ms
- 300ms
- 400ms
- 500ms
- 600ms
- 700ms
- 800ms
- 900ms

The sampling periods that can be selected differ depending on the JTAG clock speed.

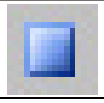
2.3.11. **Start sampling (start measurement)**

Starts sampling (measurement).

Tool button	Operation on the menu bar	Shortcut key
	**Waveform Display** -> **Start Sampling**	F7

2.3.12. **Stop sampling (stop measurement)**

Stops sampling (measurement).

Tool button	Operation on the menu bar	Shortcut key
	**Waveform Display** -> **Stop Sampling**	Ctrl + F7

2.3.13. Zoom in

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Waveform display area -> Right-click to open the context menu	---

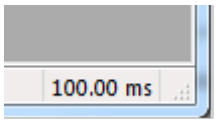
Expands the waveform displayed in the waveform display area.
When the cursor is in the waveform display area, the waveform is expanded around the cursor; otherwise around the center of current display range.

2.3.14. Zoom out

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Waveform display area -> Right-click to open the context menu	---

Reduces the waveform displayed in the waveform display area.
When the cursor is in the waveform display area, the waveform is reduced around the cursor; otherwise around the center of current display range.

The zooming ratios are 2, 5, 10... (1/2, 1/5, 1/10...).
There is no function to minimize or maximize the display at once.
In addition, the status bar displays the length of time per one graduation. The default value is 100msec.



2.3.15. Set a cursor

The cursor can be set by left-clicking in the waveform display area in the Waveform Display window. The **green** solid line indicates the cursor.

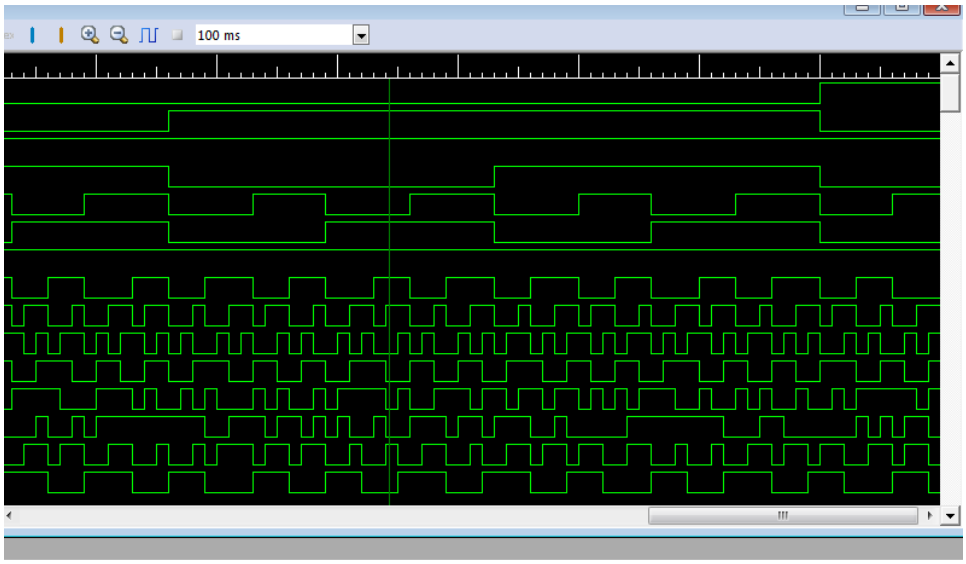


Figure 30

The status bar also displays the cursor position (C).

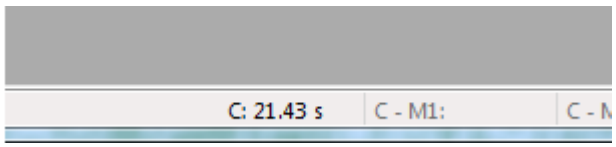
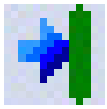


Figure 31

2.3.16. Move a cursor

The cursor can be moved by left-clicking or dragging.

2.3.17. Display cursor at center

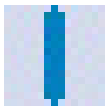
Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar.	---

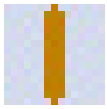
The cursor is displayed so that it is located at the center of the waveform display range.

Using with markers makes it possible to verify the time between the cursor and the marker.

2.3.18. Set a marker

Two positions can be marked in the waveform display area. (Markers 1 and 2)

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Waveform display area -> Ctrl + Left-click	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. *Waveform Display window -> Waveform display area -> Ctrl + Shift + Left-click	---

Reduces the waveform displayed in the waveform display area.
When the cursor is in the waveform display area, the waveform is reduced around the cursor; otherwise around the center of current display range.

The toolbar button can be used to set a marker to the cursor position. In addition, a marker can be set to any position in the waveform range by left-clicking the position while holding down the Ctrl key or the Ctrl and Shift keys.
Markers 1 and 2 are displayed by **blue** and **orange** broken lines, respectively.

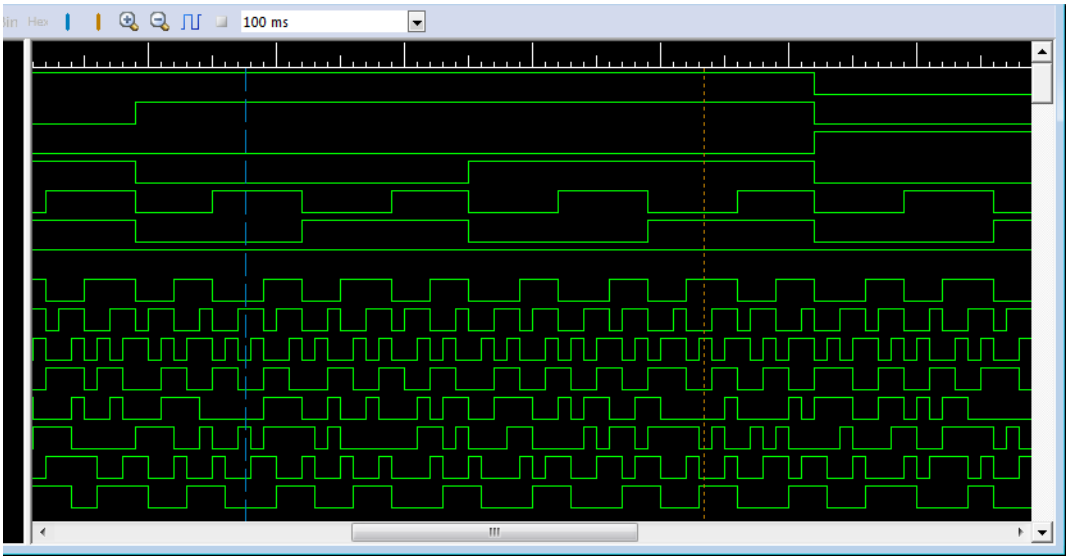


Figure 32

The status bar displays each marker position (M1 or M2). The status bar also displays difference between cursor and markers and between markers.
C-M1 : Difference between the cursor and Marker 1
C-M2 : Difference between the cursor and Marker 2
M1-M2 : Difference between Marker 1 and Marker 2

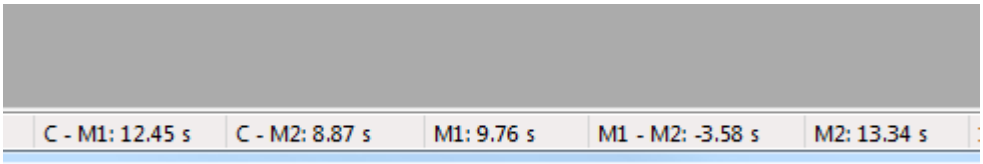


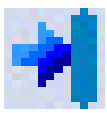
Figure 33

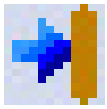
2.3.19. Move cursor to marker position

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. *Waveform Display window -> Waveform display area -> Right-click to open the context menu -> Move Cursor to Marker 1	---

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. *Waveform Display window -> Waveform display area -> Right-click to open the context menu -> Move Cursor to Marker 2	---

2.3.20. Display marker at the center

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar.	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar.	---

The marker is displayed so that it is located at the center of the waveform display range.

2.3.21. Print a waveform

The current display range in the Waveform Display window can be printed.
For details, refer to "**3.1.14. Preview Print**," "**3.1.15. Set a Printer**" and "**3.1.16. Print**."

2.3.22. Save a waveform

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Save Waveform**	---

The measured waveform data can be saved to a file.
The extension of file is .bsw.

The saved file can be loaded by selecting **File** -> **Open Waveform in Viewer** in the menu.

2.4. Pin Status List window

2.4.1. Create/open a new pin status list

Creates an empty pin status list and displays it in the Pin Status List window.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Create/Open New Pin Status List** * Alternatively, Project View -> Pin Status List, right-click to open the context menu.	Ctrl + L

Executing the above menu item adds **"list n"** to Project view and opens the Pin Status List window ("n" represents a serial number starting with "1"). Every time the above menu item is executed, a new empty list named "list n" is added and displayed in the window. However, if the above menu item is selected with "list n" selected in Project view, no new list is created, but the selected list is displayed in the window. The name of the current list appears on the title bar of the Pin Status List window.

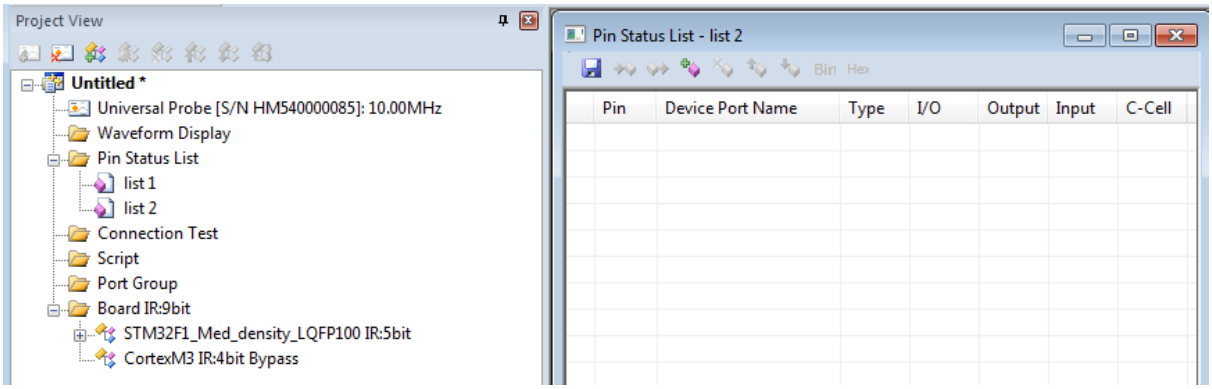


Figure 34

2.4.2. Change a list name

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Pin Status List -> Added [list n], right-click to open the context menu -> Rename *Click the mouse or press the F2 key while selecting [list n].	---

Selecting the above menu item makes the relevant items editable in the Project view. Change the name according to the purpose.

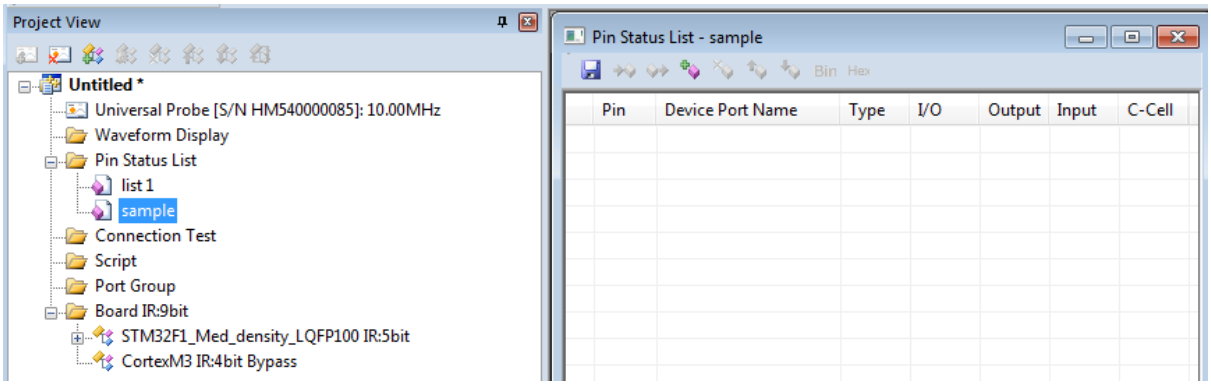


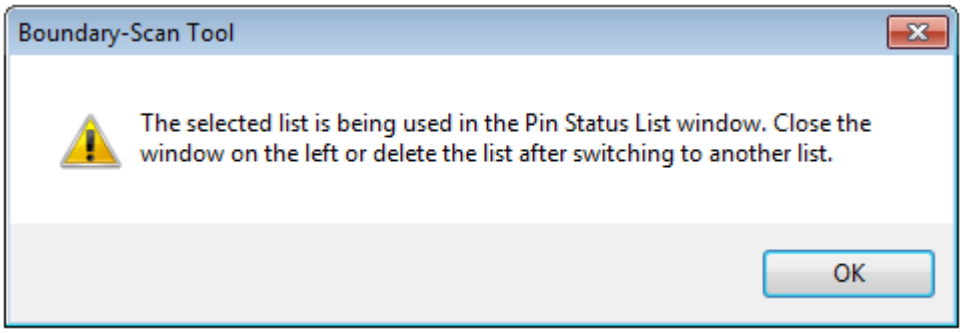
Figure 35

2.4.3. Delete a list

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Pin Status List -> Added [list n], right-click to open the context menu -> Delete	---

Selecting the above menu item deletes the relevant items from the Project view.

If the list displayed when selecting the above menu item coincides with the one displayed in the Pin Status List window, the following error message appears.



2.4.4. Add a port

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Pin Status List window -> Right-click to open the context menu.-> Add Port	---

Selecting the above menu item displays the "Add Port" dialog box. In the dialog box, all the ports of all the devices and the groups registered in the project are listed. Checkmark the desired port(s) or group(s) and click the "OK" button. Multiple ports can be selected.

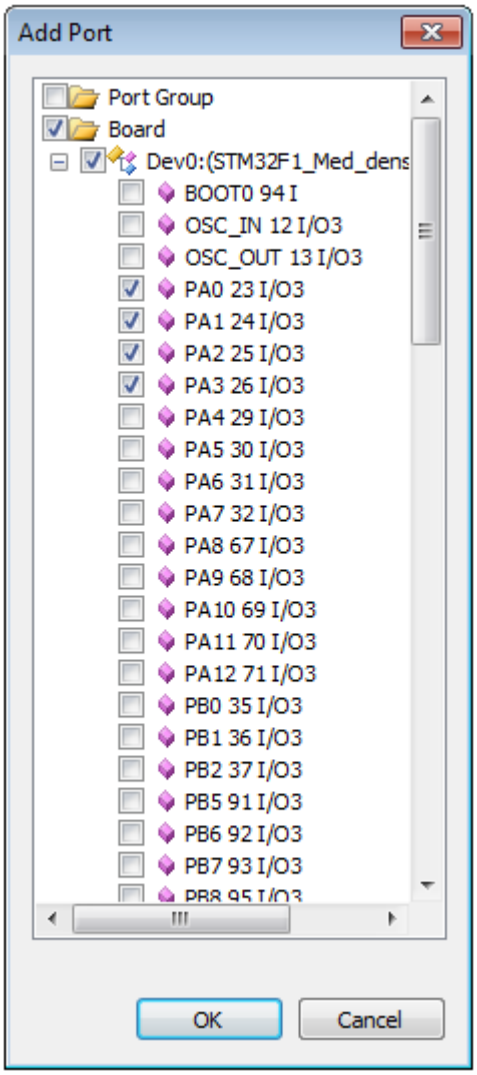
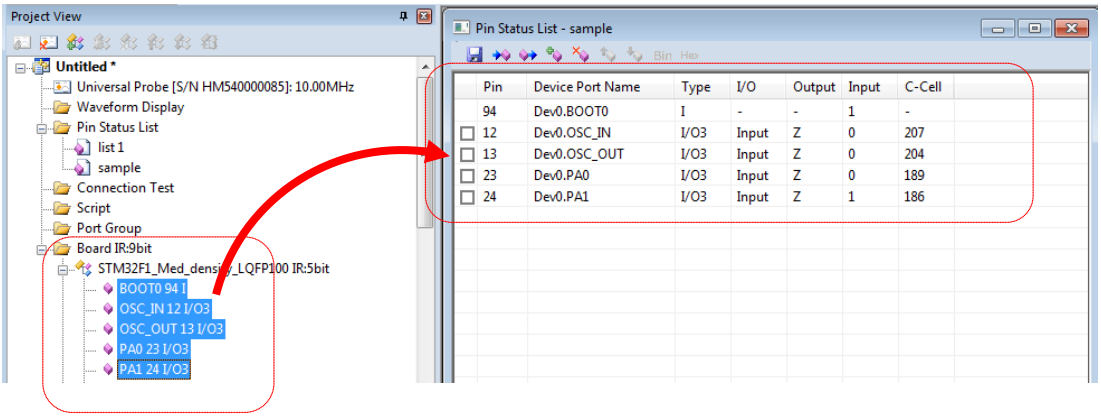


Figure 36

Clicking the checkbox for the **"Board"** tag switches all the ports of all the devices between selected and deselected. Clicking the checkbox for the **"Dev n"** tag ("n" represents a serial number in the boundary scan chain) switches all the ports of that device between selected and deselected.
Some ports share the control cell.
(Cells whose numbers' BSDL "function" is "control" are shared among two or more ports so that the I/O direction and whether to permit output are collectively managed).

When one of these ports is checkmarked, other ports that share the same cell are also checkmarked.
Clicking the checkbox for the Port Group tag switches all the groups between selected and deselected. When a group is added by clicking the OK button, the **ports that constitute the group** as well as **the ports that share the control cell with them** are added.
A port or group can also be added by dragging and dropping it in the Pin Status List window from the Project view. Multiple items can be selected in the Project view by **left-clicking the mouse while holding down the CTRL or SHIFT key**. If a port that shares the control cell is added, other ports that share the control cell are also added.
Dragging and dropping a group adds the ports that constitute the group as well as the ports that share the control cell with them.



2.4.5. Create a port group

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Pin Status List window -> Right-click to open the context menu -> Create Group	---

The group can be created by specifying a combination of added ports.
Once a group is created, values can be specified and displayed in binary or hexadecimal notation.
The group created with this menu item is not added to the Project view.

For how to create a port group in the Create Group dialog box, refer to **2.2.19. Add a port** group.

2.4.6. Switch the port group display between binary and hexadecimal notation

Tool button	Operation on the menu bar	Shortcut key
Bin	This operation cannot be performed from the menu bar. * Pin Status List window -> Select a port group. -> Right-click to open the context menu -> "Display in Binary Notation"	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Pin Status List window -> Select a port group. -> Right-click to open the context menu -> "Display in Hexadecimal Notation"	---

Selecting the above menu item switches whether to display the values in the selected group between binary and hexadecimal notations.

2.4.7. Change the display order of ports/port groups

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Pin Status List window -> Select a port/port group. -> Right-click to open the context menu -> “Up”	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Pin Status List window -> Select a port/port group. -> Right-click to open the context menu -> “Down”	---

A port or group can be moved by selecting it in the port group name display area.

2.4.8. Delete a port/port group

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Pin Status List window -> Select a port/port group. -> Right-click to open the context menu -> "Delete Port"	---

Selecting the above menu item displays the "Delete Port" dialog box. In the dialog box, all the ports and groups registered in the Pin Status List window are listed. Checkmark the port(s) and group(s) to be deleted and click the "OK" button. Multiple ports can be selected.

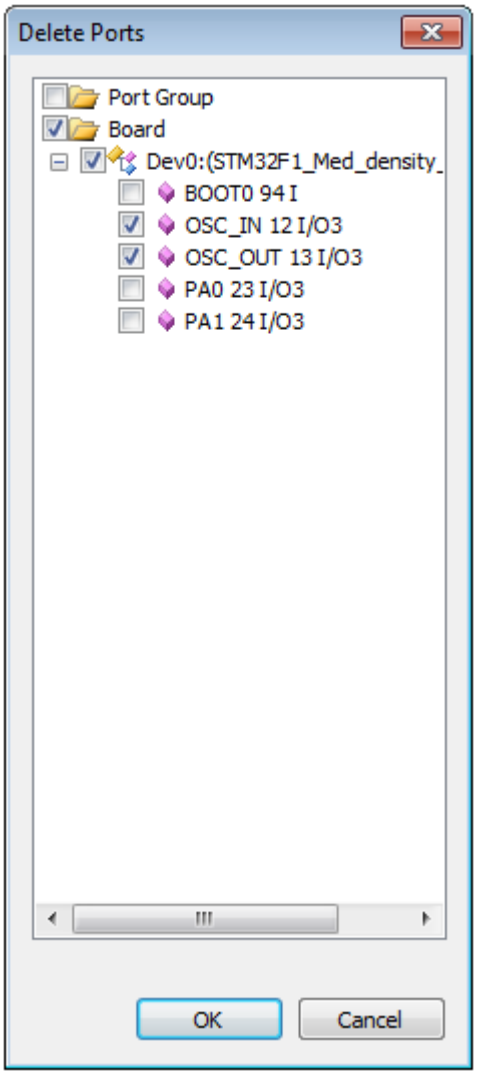


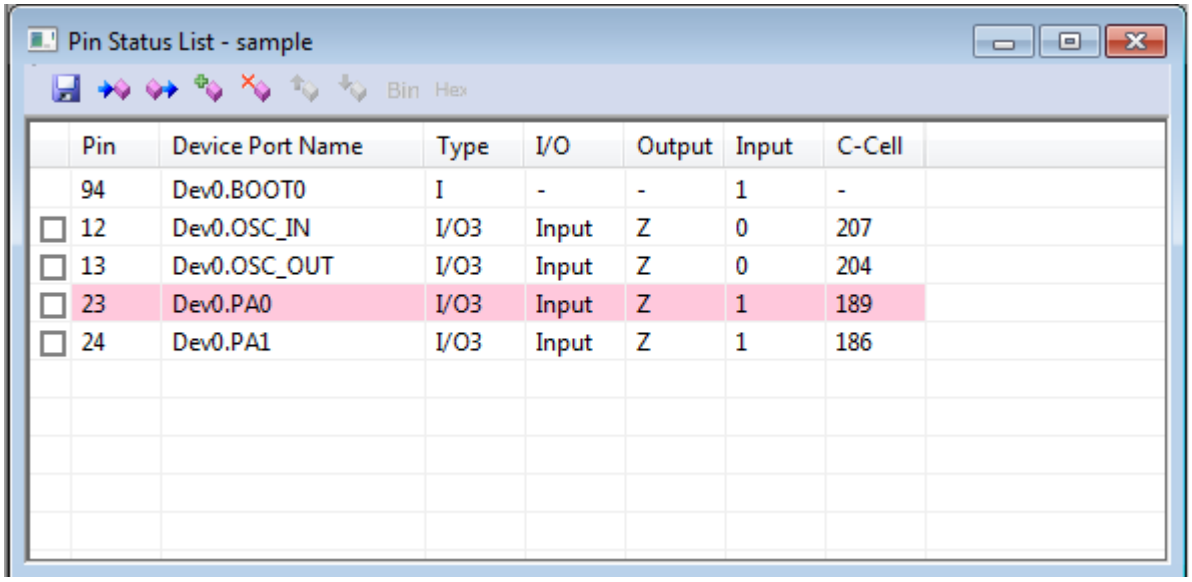
Figure 37

In the same way as for "Add Port," clicking the checkbox for the "Board" tag switches all the ports of all the devices between selected and deselected. Clicking the checkbox for the "Dev n" tag ("n" represents a serial number in the boundary scan chain) switches all the ports of that device between selected and deselected.
In the same way as for "Add Port," the ports that share the control cell with these ports are also switched between selected and deselected.
If a port that constitutes a group is deleted, that group is deleted regardless of whether it is checkmarked or not.
If a group is checkmarked but the ports that constitute it are not, the ports are not deleted.

2.4.9. Execute the SAMPLE command

Tool button	Operation on the menu bar	Shortcut key
	**Pin Status List -> SAMPLE Command** * Alternatively, Pin Status List window -> Right-click to open the context menu.	F5

The SAMPLE command is issued to the devices in the boundary scan chain and allows them to load the device I/O data to the boundary scan register. This software reads the loaded values and updates the list in the Pin Status List window accordingly. The rows containing a value changed by this update are displayed in a different color.



Pin	Device Port Name	Type	I/O	Output	Input	C-Cell
94	Dev0.BOOT0	I	-	-	1	-
12	Dev0.OSC_IN	I/O3	Input	Z	0	207
13	Dev0.OSC_OUT	I/O3	Input	Z	0	204
23	Dev0.PA0	I/O3	Input	Z	1	189
24	Dev0.PA1	I/O3	Input	Z	1	186

Figure 38

2.4.10. Execute the EXTEST command

Tool button	Operation on the menu bar	Shortcut key
	**Pin Status List -> EXTEST Command** * Alternatively, Pin Status List window -> Right-click to open the context menu.	Ctrl + F5

The EXTEST command is issued to the devices in the boundary scan chain and checks the connection status of the I/O pins of these devices. After the EXTEST command is executed, this software reads the status of the I/O pins of these devices and updates the list in the Pin Status List window. The rows containing a values changed by this update are displayed in a different color.
The detailed procedure is as follows:

1) Checkmark the port or group for which values are to be set.

- No checkbox appears for input ports (the Type column is "I").
  An output value can be set for output ports (the Type column is "O") by checkmarking the corresponding checkbox.
  An I/O direction and output value can be set for I/O ports (the Type column is "I/O") by checkmarking the corresponding checkbox.

- When a port is selected or deselected, other ports that share the same C-Cell value with that port are also selected or deselected.
  A checkbox appears for a group that contains one or more output or I/O ports.
  If a group is selected or deselected, all the ports that constitute the group are also selected or deselected.
  Other ports that share the same C-Cell value with these ports are also selected or deselected.

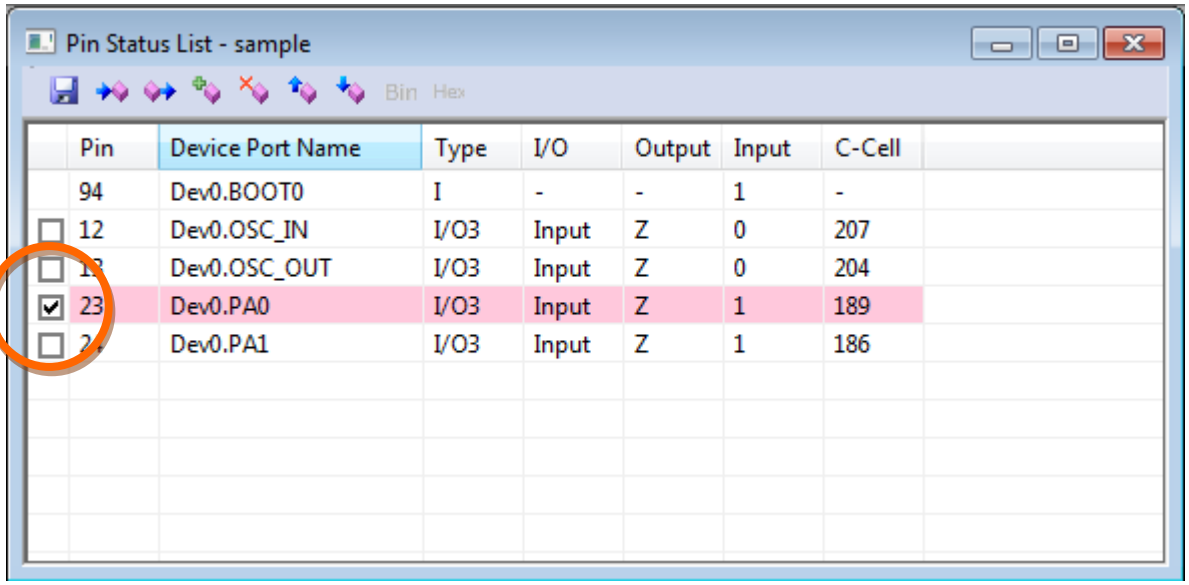


Figure 39

2) Click the I/O column and select an I/O direction (only for I/O ports).

- If Input is selected, that port is used as an input port.
  In this case, the output value is not changed.
  If Output is selected, set an output value in step 3.
- When the I/O direction of a port is changed, the I/O directions of other ports that share the same C-Cell value with that port are also changed.

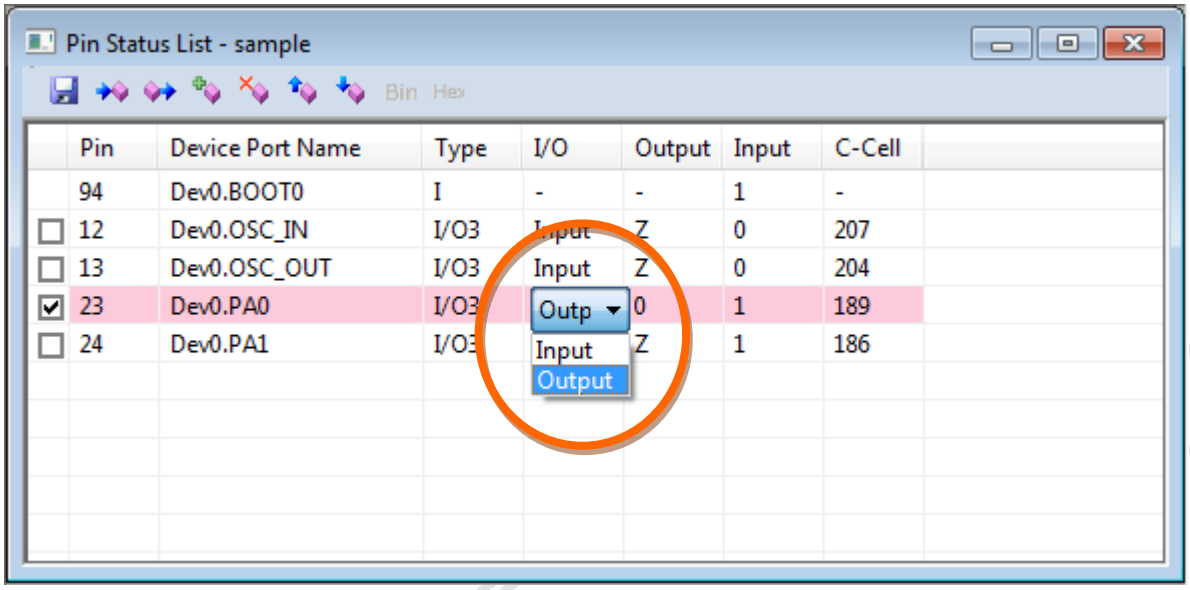


Figure 40

3) Click the Output column and select an output value.

- For an output only port with a control cell, "Z" can be selected to prohibit output.
  Select 0 (Low output) or 1 (High output) for other ports.
- When the output value for a port is changed from "0" or "1" to "Z" or from "Z" to another value, the output values for other ports that share the same C-Cell value are also changed.
- For a group, set a group value in the edit box.
  However, values in some bits may be ignored, depending on the type or I/O status of the port that constitutes the group.
  Whether a group value should be set in binary or hexadecimal notation depends on the current display format for the group.

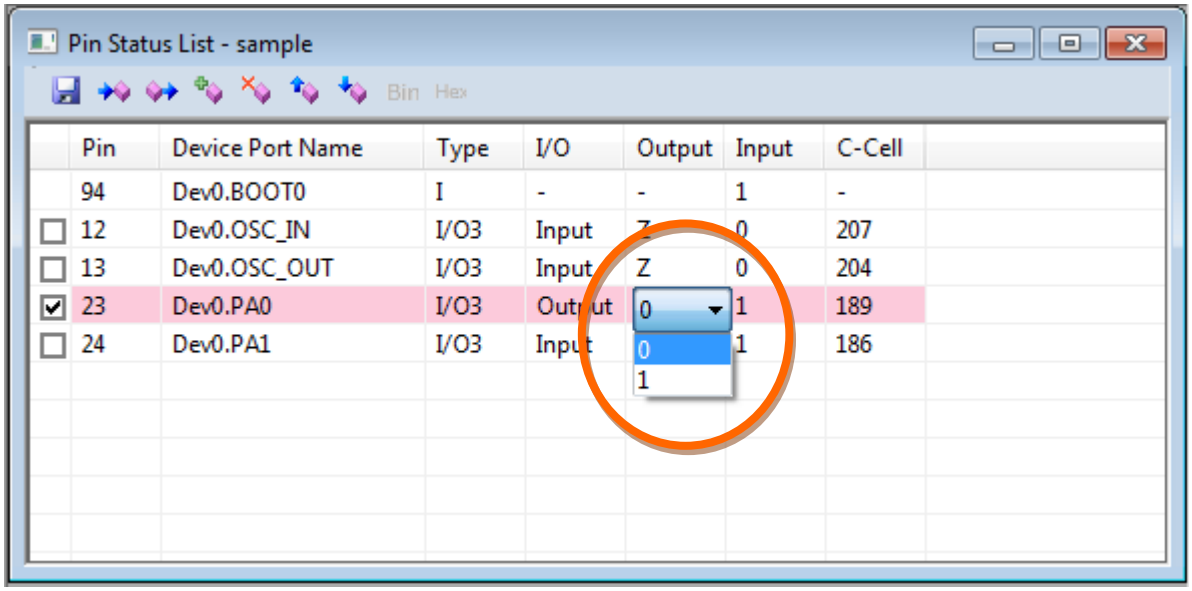


Figure 41

4) Repeat steps 1 to 3 for all ports for which values are to be specified.

- Deselected ports function as follows:
  - I/O ports >> Used as input ports.
  - Output ports (with a C-Cell value) >> Used with output prohibited.
  - Output ports (without C-Cell value) >> Output 0 (Low).

5) Execute EXTEST.

- When executing EXTEST, a warning message that asks you whether you really want to execute EXTEST with these settings appears. To execute the command, select **"OK."** To check the settings again, select **"Cancel."** If you do not want this warning message to appear again, checkmark the **"Do not show this message again"** checkbox. This check can also be set from the **"Pin Status List"** menu.

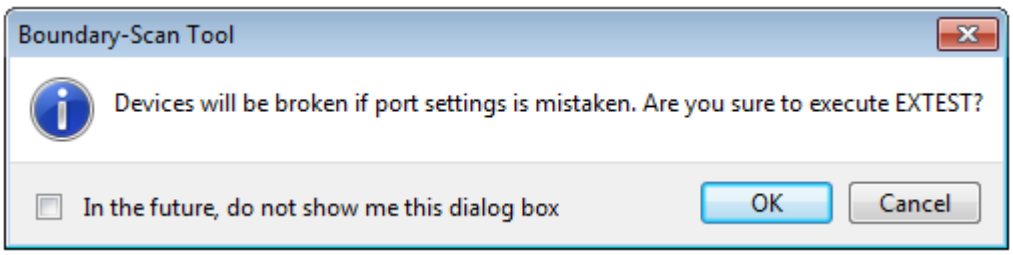


Figure 42

Depending on the I/O direction or output values of ports, the target board may be damaged under unexpected load. Check the settings carefully before executing this command.

The value of the port of devices set to bypass in **"Device Properties"** are not changed by executing a command, since they are skipped by the BYPASS command of boundary scan when the command is executed.
Those ports are displayed on gray-colored background.

2.4.11. Save to a CSV file

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Pin Status List in CSV File** * Pin Status List window -> Toolbar -> "Save in CSV"	---

Selecting the above menu item displays the Save File dialog box. Specify where to save the file and a file name. The initial file name is the name of the pin status list. Clicking Save creates a file as shown below **:**

	A	B	C	D	E	F	G	H
1	DISABLE	94	Dev0.BOOT0	I	–	–	1	–
2	FALSE	12	Dev0.OSC_IN	I/O3	Input	Z	0	207
3	FALSE	13	Dev0.OSC_OUT	I/O3	Input	Z	0	204
4	TRUE	23	Dev0.PA0	I/O3	Input	Z	1	189
5	FALSE	24	Dev0.PA1	I/O3	Input	Z	1	186
6								
7								

Figure 43

- The items in the CSV file correspond to **"Checkbox," "Pin," "Device Port Name," "Type," "I/O," "Output," "Input," and "C-Cell"** from left to right.
- For a row for a group, the pin numbers for the port that constitutes the group are output next to the "C-Cell" column.
- The checkbox status has the following meanings:
  - No Checkbox: DISABLE
  - ON: TRUE
  - OFF: FALSE

  The current values in the Pin Status List window are displayed in other columns.

2.5. Connection Test window

This section explains how to use the Connection Test window of this software.

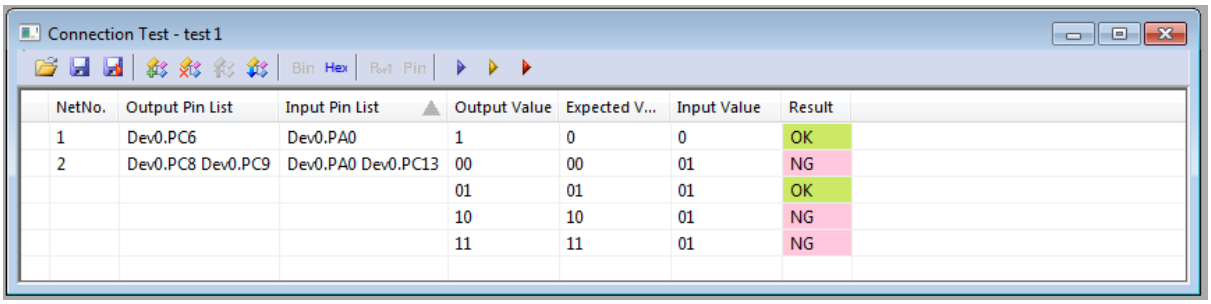


Figure 44

2.5.1. Create a new Connection Test window/open a Connection Test window

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Create new Connection Test window/Open Connection Test window** * Alternatively, Project View -> Connection Test -> Right-click to open the context menu.	Ctrl + T

Executing the above menu item adds "test n" to Project view and opens the Connection Test window ("n" represents a serial number starting with "1"). Every time the above menu item is selected, a new empty list named "test n" is added and displayed in the window. However, if the above menu item is selected with "test n" selected in Project view, no new list is created but the selected list is displayed in the window. The name of the current list appears on the title bar of the Connection Test window.

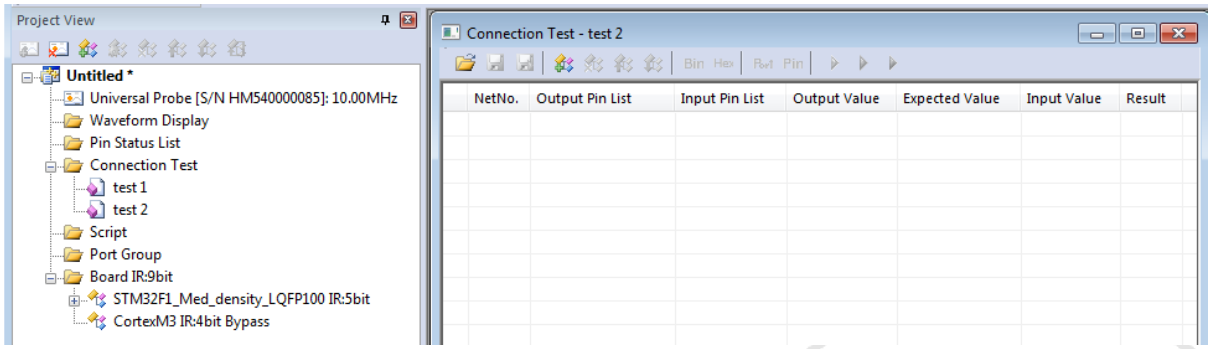


Figure 45

2.5.2. Rename a test (list)

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Connection Test -> Added [test n], right-click to open the context menu -> Rename * Click the mouse or press the F2 key while selecting [test n].	---

Selecting the above menu item makes the relevant items editable in the Project view. Change the name according to the purpose.

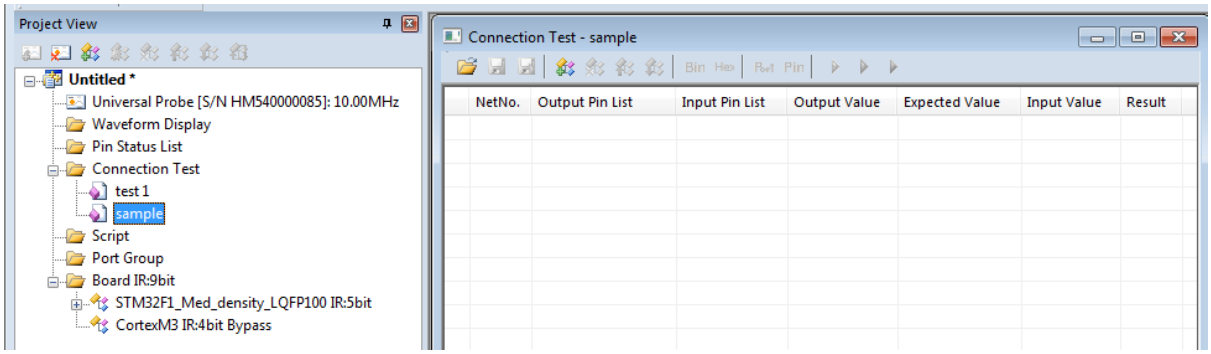


Figure 46

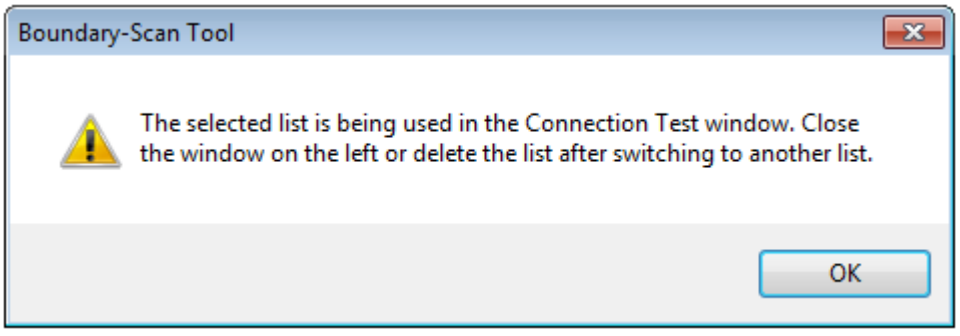
2.5.3. Delete a test (list)

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Connection Test -> Added [test n] Right-click to open the context menu -> Delete	---

Selecting the above menu item deletes the relevant items from the Project view.

INFO

If the test (list) displayed when selecting the above menu item coincides with the one displayed in the Connection Test window, the following error message appears.



2.5.4. Add a net

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Connection Test window -> Right-click to open the context menu -> "Add Net"	---

Selecting the above menu item adds an empty **"net n"** row to the Connection Test window ("n" represents a serial number starting with "1").

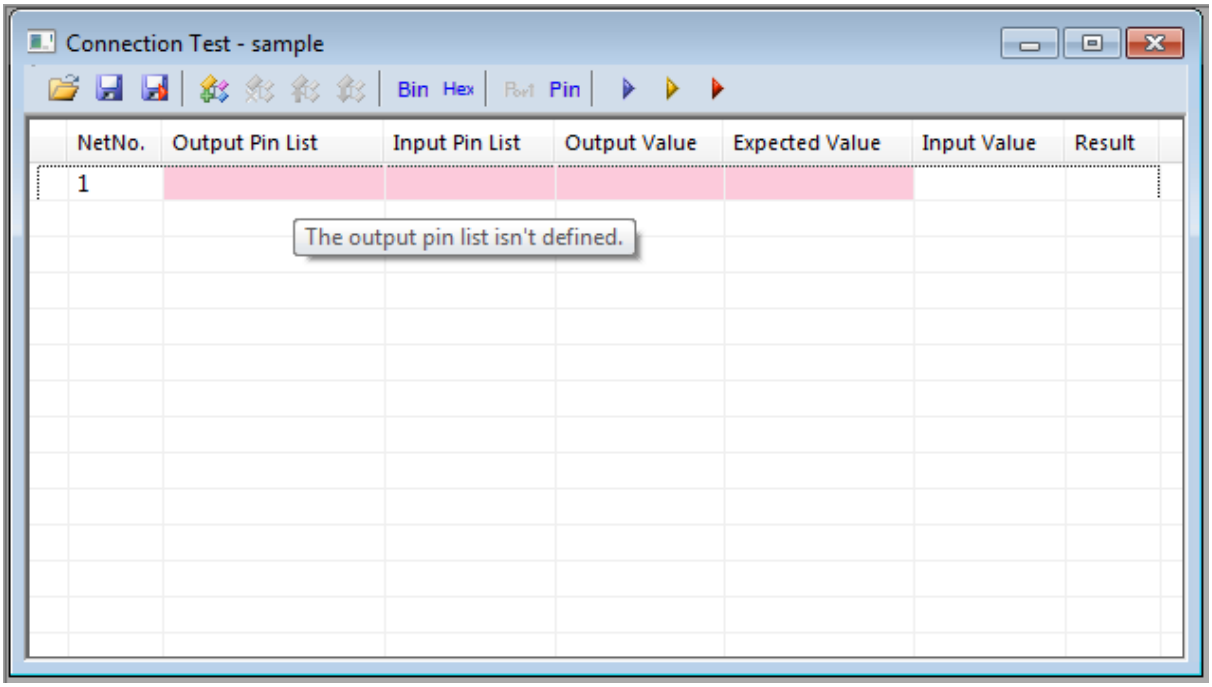


Figure 47

Columns with pink-colored background indicate that incorrect values are set in them. Placing the mouse cursor on these columns displays a pop-up that describes why they are incorrect. When the columns no longer have incorrect settings, the background color changes to blue-violet.

2.5.5. Edit a pin list

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Connection Test window -> Input/Output Pin List -> Right-click to open the context menu -> "Edit Input/Output Pin List"	---

Selecting the above menu item opens the **"Edit Pin List"** dialog box. Configure a list of pins on which a connection test is to be executed.

• **Edit Pin List dialog box**

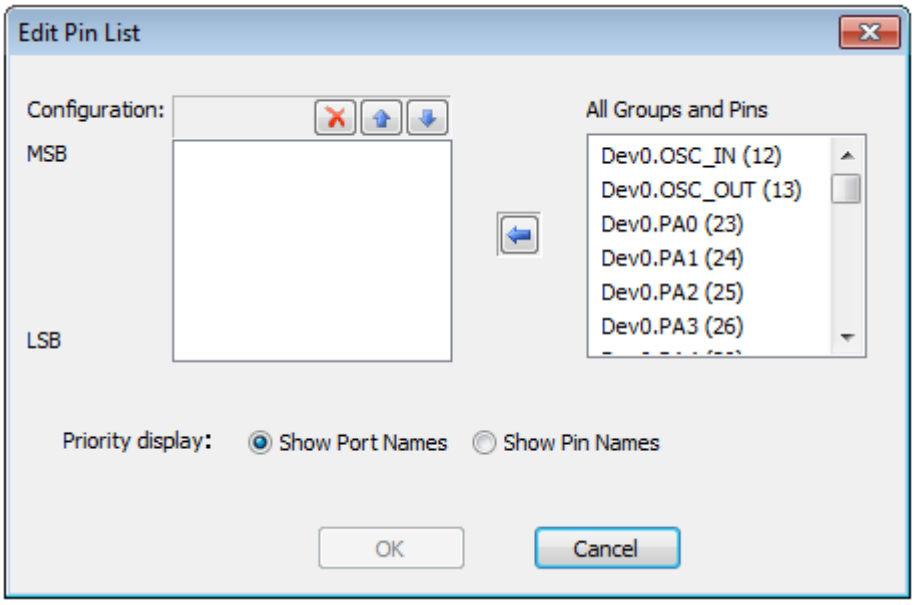


Figure 48

• **Specifying a port to be configured**
All the pins on the board and groups added are listed in the "All Groups and Pins" list. Select the pin or group you want to add to the group from the list and click the [←] button. The group is added to the end of the “Configuration” list.

• **Changing the order of ports to be configured**
To change the order of a pin/group in the "Configuration" list, select it from the list and change the order by clicking the [↑] or [↓] button.

• **Deleting a port to be configured**
To delete a pin/group from the "Configuration" list, select it and click the [X] button.

• **Switching whether to show port names or pin names**
Clicking "Show Port Names" or "Show Pin Names" switches whether to show port names or pin names in the Input/Output Pin List column.

• **Reflecting the edit**
Click the [OK] button to reflect the edited pin list.

If an output pin list contains an input only pin, the background color of the Output Pin List column changes to pink to indicate that there is an incorrect setting.
Similarly, if an input pin list contains an output only pin, the background color of the Input Pin List column changes to pink to indicate that there is an incorrect setting.

A pin list can also be edited by the following methods.

- By dragging and dropping a pin or group in Project view on the Output/Input Pin List column, it is registered.
- By left-clicking on the Output/Input Pin List column, a dialog box for editing opens. Then, edit the list directly. Delimit pin names with space characters.

If the settings of the list are correct, the background color of the edited column changes to blue-violet. Placing the mouse cursor on the Input/Output Pin List column displays all the constituent elements in a pop-up.

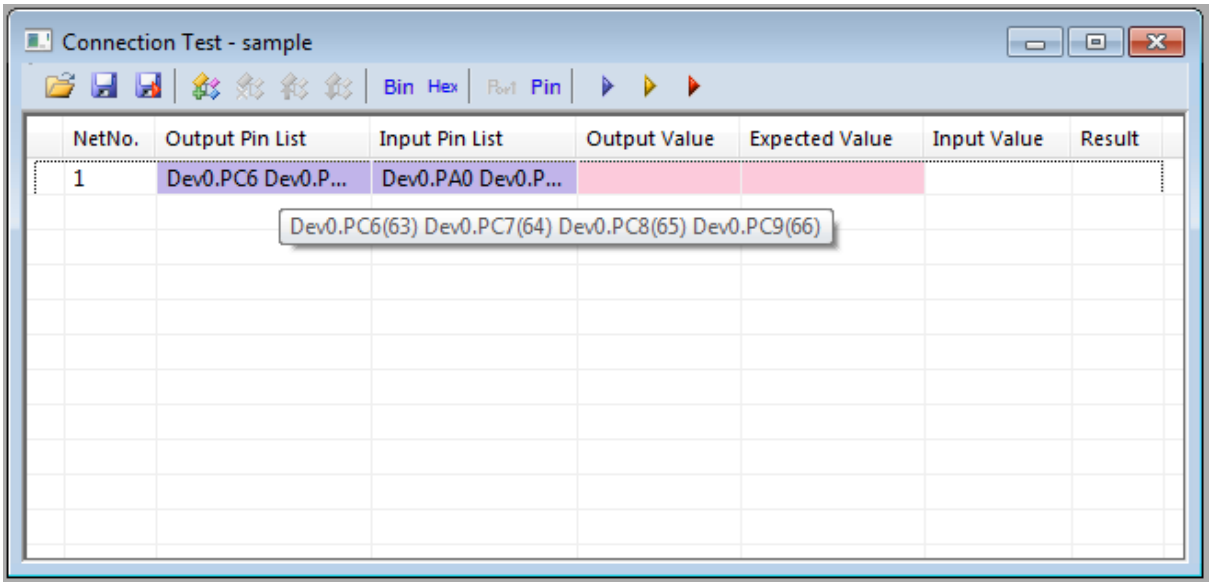


Figure 49

2.5.6. Delete a net

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Connection Test window -> Right-click to open the context menu -> "Delete Net"	---

Selecting the above menu item deletes the **"net n"** row. If two or more output value patterns have been registered, the corresponding rows are also deleted.

2.5.7. Change the order in which a net is listed

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Connection Test window -> Right-click to open the context menu -> "Move Net Upward"	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Connection Test window -> Right-click to open the context menu -> "Move Net Downward"	---

Selecting **"Move Net Upward"** allows the selected "net n" to move upward and replace the next net above in the connection test list.
Selecting **"Move Net Downward"** allows the selected "net n" to move downward and replace the next net below in the connection test list.

2.5.8. Open a net list

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Open Net List**	Ctrl+Shift+T

Selecting the above menu item opens the Select File dialog box. Select a net list file (bnt file). Create a net list file in the specified format in advance or use a file created in this window and saved in advance.

- If a file with the same name as a test name registered under the "Connection Test" tag in Project view is opened, a net is reconfigured in that test.
  If a file with a new name is opened, the test name in Project view changes to the test name of the file, and a net described in the file is reconfigured.
- If the net list file has an error, an error list file (ber file) is created in the folder of the net list file. Make use of the error list file to correct the net list file.

2.5.9. Create/edit/delete a group

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Connection Test window -> Right-click to open the context menu -> "Create/Edit/Delete Group"	---

Selecting the above menu item opens the Manage Group dialog box. In this dialog box, a group that can be used in the Connection Test window can be created, edited or deleted.

• **Manage Group dialog box**

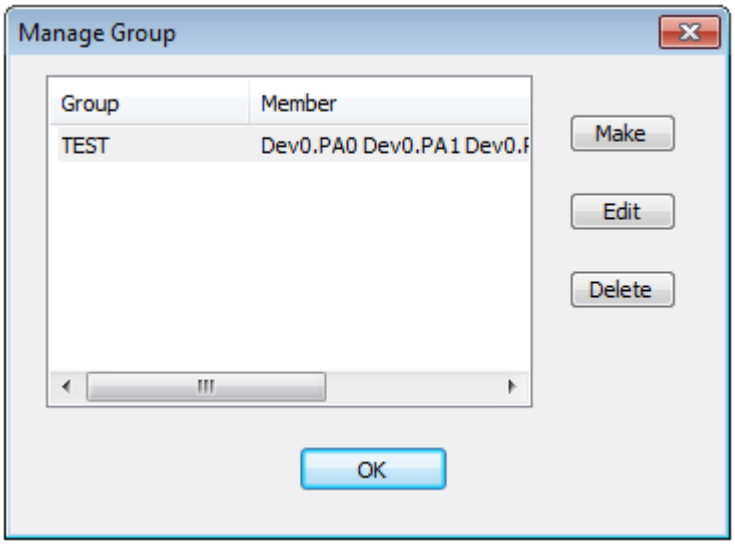


Figure 50

Clicking the Create button and Edit button also opens the Create Group dialog box. Create a group by specifying a combination from all the ports on the board.
The group created with this menu item is not added to the Project view.

Groups created in Project view also appear on the Manage Group dialog box. If the group is edited, the changes take effect only in the **Connection Test window**. They are not reflected in Project view. In addition, **these groups cannot be deleted**.

• **Create Group dialog box**

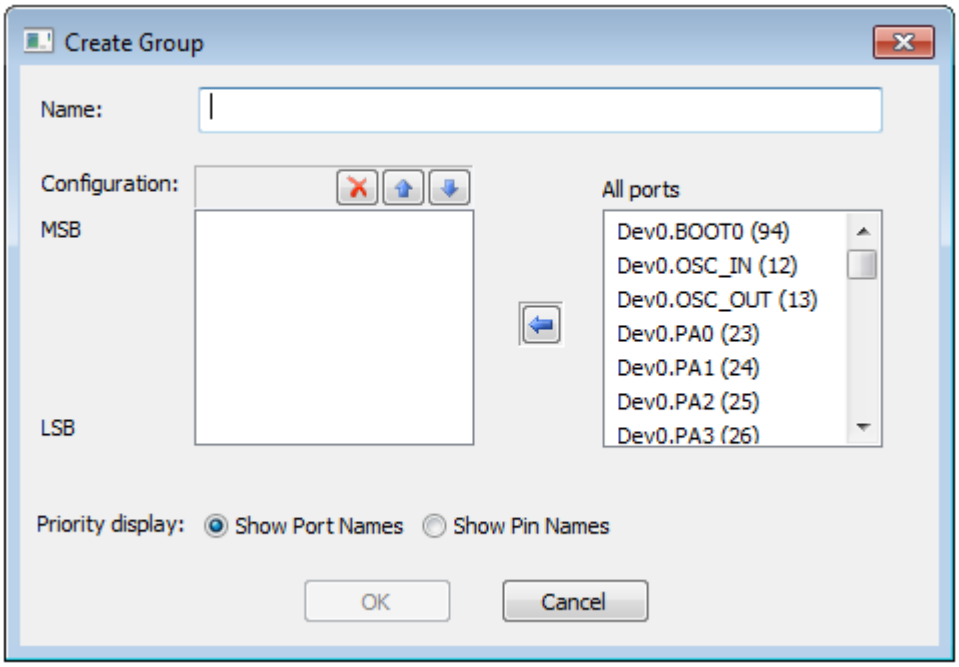


Figure 51

• **Specifying a name**
Specify a group name in “Name.”

• **Specifying a port to be configured**
All the ports on the board are listed in the “All ports” list. Select the port you want to add to the group from the list and click the [←] button. The group is added to the end of the “Configuration” list.

• **Changing the order of ports to be configured**
To change the order of a port in the "Configuration" list, select it from the list and change the order by clicking the [↑] or [↓] button.

• **Deleting a port to be configured**
To delete a port from the "Configuration" list, select it and click the [X] button.

• **Specifying a format in which pins are displayed**
By clicking "Show Port Names" or "Show Pin Names," the format to display the group configuration can be specified.

• **Adding a Group**
Click the [OK] button to add a group. A group added here can be selected in editing output/input pin lists.

2.5.10. Switch whether to show pin names or port names

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Connection Test window -> Input/Output Pin List -> Right-click to open the context menu -> "Show Port Names"	---

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Connection Test window -> Input/Output Pin List -> Right-click to open the context menu -> "Show Pin Names"	---

Switches whether to display port names or pin names for pins in the selected column.
Or, pressing the button on the toolbar without selecting any Output Pin List or Input Pin List column, the display format of all pins can be switched.

2.5.11. Set output values

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar. * Connection Test window -> Output Pin List -> Right-click to open the context menu -> "Create Output Patterns Automatically" * Connection Test window -> Output Value -> Right-click to open the context menu -> "Add/Delete Output Pattern"	---

Sets values output from the pins of the output pin list when executing a connection test.
Clicking the Create Output Patterns Automatically button inserts the four output patterns of "output 0 from all pins," "output 1 from all pins," "0101..."and" 1010...."
To set another output value, click a pattern created automatically and edit it directly, or select the "Add Output Pattern" menu item to add a new output pattern row and then edit it.

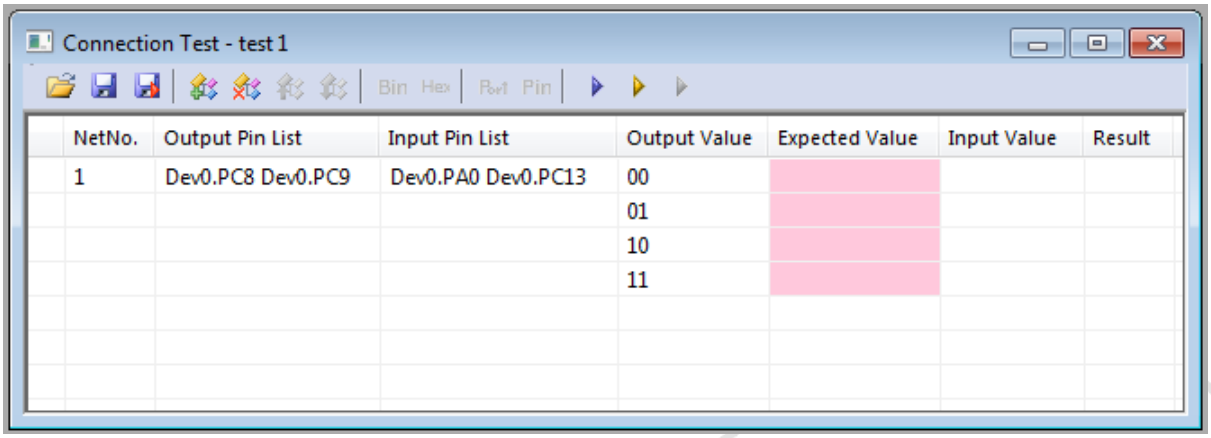


Figure 52

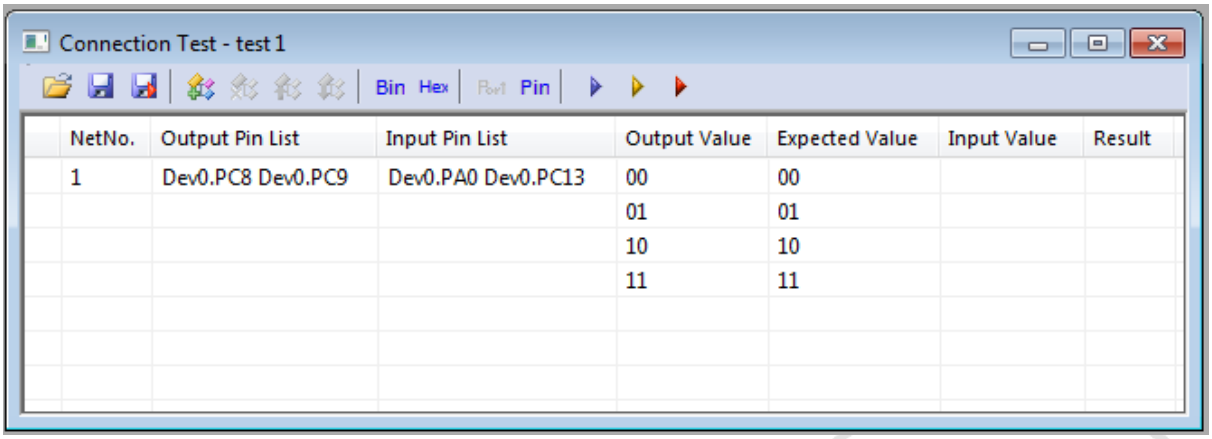
2.5.12. Set expected values

Tool button	Operation on the menu bar	Shortcut key
	**Connection Test** -> **Learn Expected Values** * Connection Test window -> Expected Value -> Enter values directly from the keyboard	---

Set values the pins of the input pin list are expected to have when a connection test is executed with the set output values.
If the letter "x" is set, the corresponding bit is not compared with the input value at the time of testing (it is always judged as OK).
The letter "x" set in hexadecimal notation means that all the four bits are ignored.

If the "Learn Expected Values" menu item is executed, the EXTEST command of a boundary scan test is executed, and the values are loaded and set as the expected values.
If you have a board that has already passed a connection test, that board can be used to create expected values.

The "Learn Expected Values" are executed for the following items:
・When the row is selected by clicking the net number column, all patterns of the selected net is executed.
・When the row is selected by clicking other columns (output pin list, input pin list, output value, expected value, input value, result), only the selected row is executed.
・When output pin list, input pin list, output value, expected value columns are being edited, or just after having edited the settings, only the row is executed.
・When no row is selected, e. g. by clicking a row without registered net, all the registered nets are executed.



NetNo.	Output Pin List	Input Pin List	Output Value	Expected Value	Input Value	Result
1	Dev0.PC8 Dev0.PC9	Dev0.PA0 Dev0.PC13	00	00		
			01	01		
			10	10		
			11	11		

Figure 53

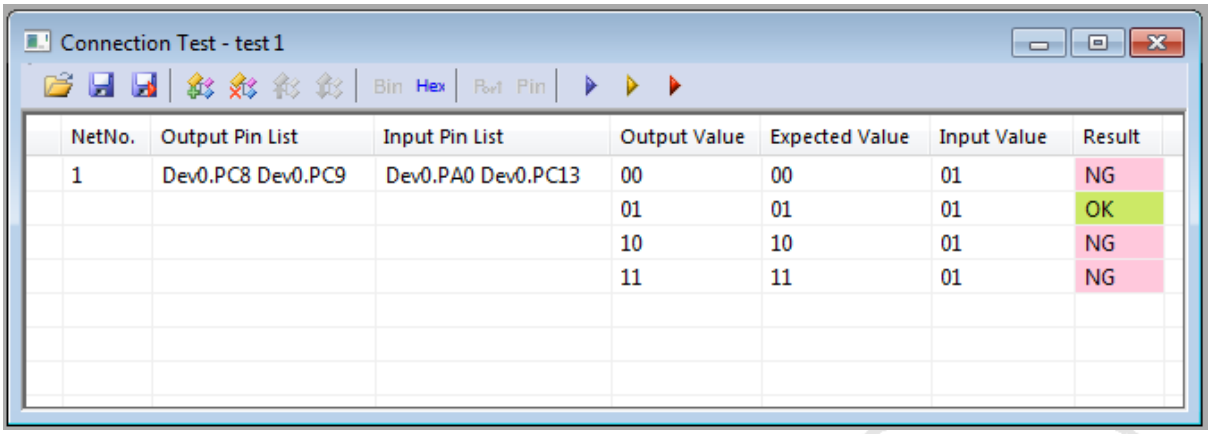
2.5.13. Execute a test

Tool button	Operation on the menu bar	Shortcut key
	**Connection Test** -> **Connection Test**	---

Use the EXTEST command of a boundary scan test to execute a connection test.
Rows with output values and expected values set can be tested.

The connection test is executed for the following items.
・When the row is selected by clicking the net number column, all patterns of the selected net is executed.
・When the row is selected by clicking other columns (output pin list, input pin list, output value, expected value, input value, result), only the selected row is executed.
・When output pin list, input pin list, output value, expected value columns are being edited, or just after having edited the settings, only the row is executed.
・When no row is selected, e. g. by clicking a row without registered net, all the registered nets are executed.

In an output test, the pins specified in "Output Pin List" are driven to the values specified in "Output Value" by the EXTEST command first.
Then, the values of the pins specified in Input Pin List are loaded and compared with the expected values.
If the loaded values are equal to the expected values, "OK" appears in the Result column displayed with green-colored background.
If the loaded values are not equal to the expected values, "NG" appears in the Result column displayed with a pink-colored background.

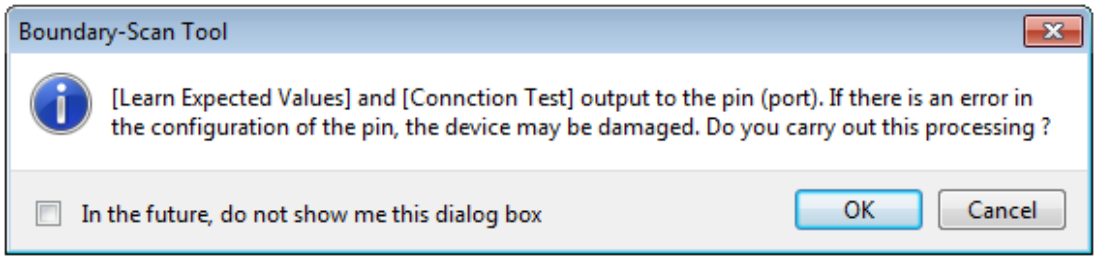


NetNo.	Output Pin List	Input Pin List	Output Value	Expected Value	Input Value	Result
1	Dev0.PC8 Dev0.PC9	Dev0.PA0 Dev0.PC13	00	00	01	NG
			01	01	01	OK
			10	10	01	NG
			11	11	01	NG

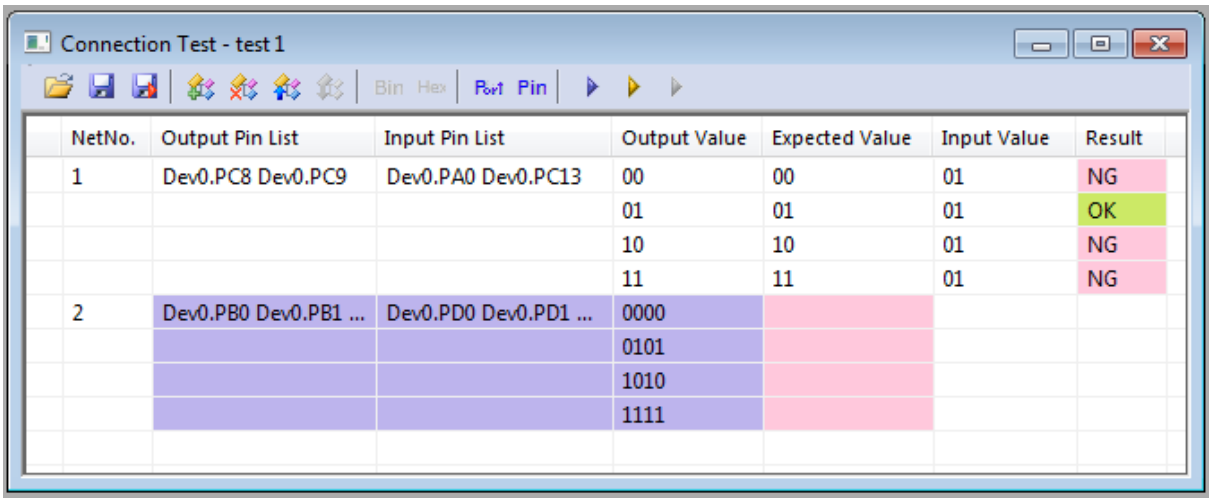
Figure 54

INFO

When executing EXTEST, a warning message that asks you whether you really want to execute EXTEST with these settings appears. To execute the command, select **"OK"**. To check the settings again, select **"Cancel."** If you do not want this warning message to appear again, checkmark the **"Do not show this message again"** checkbox. This check can also be set from the **"Connection Test"** menu.



For nets to add later, conduct tests by repeating steps 1 to 3 above.



NetNo.	Output Pin List	Input Pin List	Output Value	Expected Value	Input Value	Result
1	Dev0.PC8 Dev0.PC9	Dev0.PA0 Dev0.PC13	00	00	01	NG
			01	01	01	OK
			10	10	01	NG
			11	11	01	NG
2	Dev0.PB0 Dev0.PB1 ...	Dev0.PD0 Dev0.PD1 ...	0000			
			0101			
			1010			
			1111			

Figure 55

2.5.14. Switch between binary and hexadecimal notation

Tool button	Operation on the menu bar	Shortcut key
Bin	This operation cannot be performed from the menu bar. * Connection Test window -> Output Value/Expected Value/Input Value -> Right-click to open the context menu -> "Display in Binary Notation"	---

Tool button	Operation on the menu bar	Shortcut key
Hex	This operation cannot be performed from the menu bar. * Connection Test window -> Output Value/Expected Value/Input Value -> Right-click to open the context menu -> "Display in Hexadecimal Notation"	---

Place the mouse pointer over the "Output Value," "Expected Value," or "Input Value" column and switch whether to display the values in that column in binary notation or hexadecimal notation.
By pressing the button on the toolbar without selecting any column, the display format of output, expected and input values can be switched.
However, to switch expected values containing the letter "x" from binary to hexadecimal notation, the letter "x" must be 4 consecutive bits (1 digit in hexadecimal notation).

2.5.15. Save a net list

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Net List**	---

Selecting the above menu item displays the Save File dialog box. Specify where to save the file and a file name.
The initial file name is the name of the connection test.
Two files (.bnt file and .bnv file) are created.
The formats of these files are as follows:

Table 11

.bnt file	
[DeviceName]	Lists device reference names. The list consists of combinations of "device reference name, device initial name." If no device reference name has been set for any device, this tag is omitted. Example: [DeviceName] U1, Dev0 U2, Dev1
[PortNameFile]	Lists port name definition files per device. The list consists of combinations of "device name, port name definition." Device names may be device reference names or device initial names. If no port name change has been set for any device, this tag is omitted. Example: [PortNameFile] Dev0, test2_Dev0.pn
[GroupName]	Lists groups used in the connection test. The list consists of combinations of "group name, pin 1, pin 2, ..." Pins are delimited with a space. This tag is omitted if no group is used. Example: [GroupName] TESTGROUP, U1.1 U1.2 U1.3 U1.4
[Net]	Lists net settings. The list consists of combinations of "net number, output pin list, input pin list." Pins are delimited with a space.

	Example: [Net] 1, U1.1 U1.2 U1.3 U1.4…, U2.1 U2.2 U2.3 U2.4… 2, TESTGROUP, U1.200 U1.201 U1.202 U1.203…

Table 12

.bnv file	
[NetValue]	Lists output values, expected values, and input values per net. The list consists of combinations of "net number, output values, expected values, input values." Values displayed in hexadecimal notation in the Connection Test window are suffixed with "H." Bits that are not compared in the connection text (always judged as "OK") are represented by the letter "x." Example: [NetValue] 1, 00000000, xxxxxxxx, 11101001 1, 01010101, 01010101, 11111111 1, AAH, AAH, FFH 1, FFH, xFH, FFH 2, 0, 0101, 0101 2, 1, 1010, 1010

2.5.16. Save the connection text results

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Connection Test Results**	---

Selecting the above menu item displays the Save File dialog box. Specify where to save the file and the file name.
The initial file name is the name of the connection test. The file is saved in CSV format.

; Test Pattern : sample
; Net List : C:¥Query-J¥sample.bnt
; Data List : C:¥Query-J¥sample.bnv
; Date : 2012/02/27 20:21:10
;

1	U1.A0_PTK0 U1.A1 U1.A2 U1.A3 U	U1.D7 U1.D6 U1.D5 U1.D4 U1.D3	0	11100001	11101001	OK
1			1010101	11100001	11100001	OK
1			10101010	11100011	11100001	OK
1			11111111	11100011	11100011	OK
2	U1.A8 U1.A9 U1.A10 U1.A11	U1.D8 U1.D9 U1.D10 U1.D11	0	1101	1101	OK
2			101	1101	1101	OK
2			1010	1101	1101	OK
2			1111	1101	1101	OK

Figure 56

Test result data is listed below the header including test name and date/time. The test results are arranged in the following order from left to right:
"net number, (output pin list), (input pin list), output values, expected values, input values, result (OK or NG)."

2.6. Script pane

This section explains how to use the Script pane of this software.

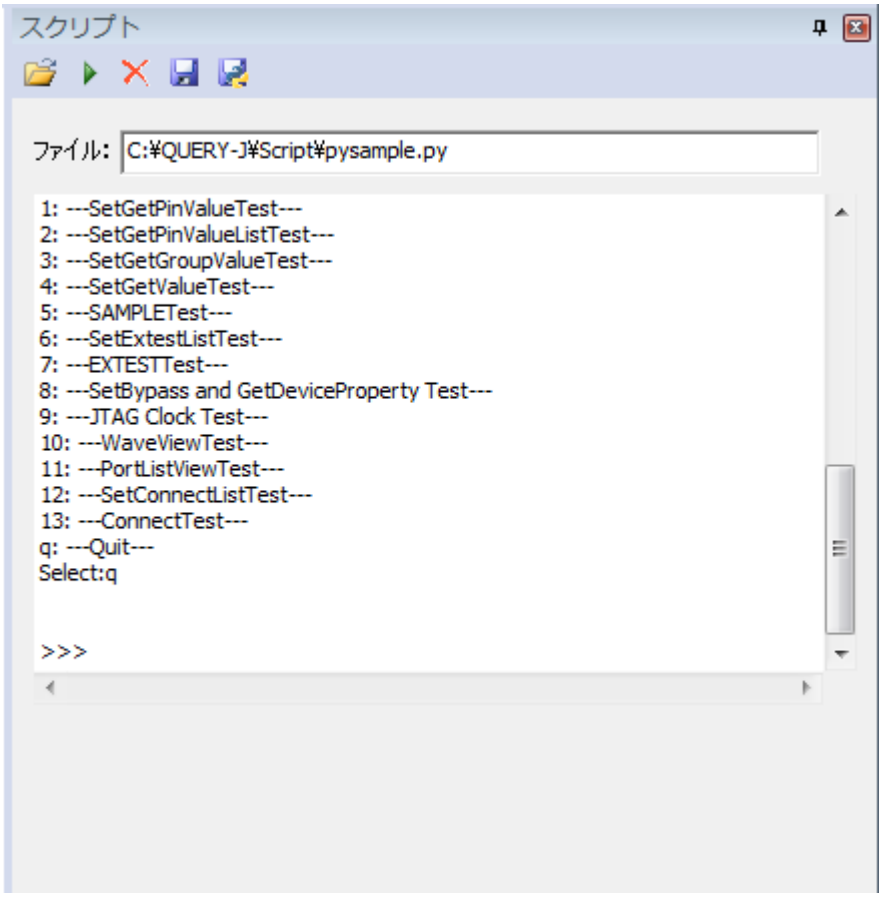


Figure 57

2.6.1 Select a file

Tool button	Operation on the menu bar	Shortcut key
	Project -> Add Script File * Project View -> Script -> Right-click to open the context menu -> "Add File"	---

Clicking the above-mentioned button on the toolbar in the Script pane opens the Select File dialog box. Select a Python script file created in advance and press the **OK** button on the Select File dialog box, and the name of the selected file appears on the **File:** edit box.

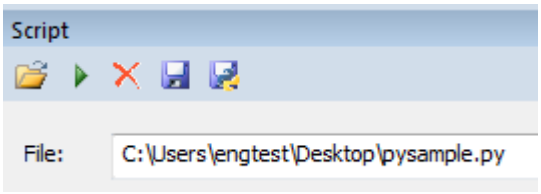


Figure 58

Operating the Project view opens the Select File dialog box. Select a Python script file created in advance and press the OK button on the Select File dialog box, and the name of the selected file is added under the Script tree in Project view.

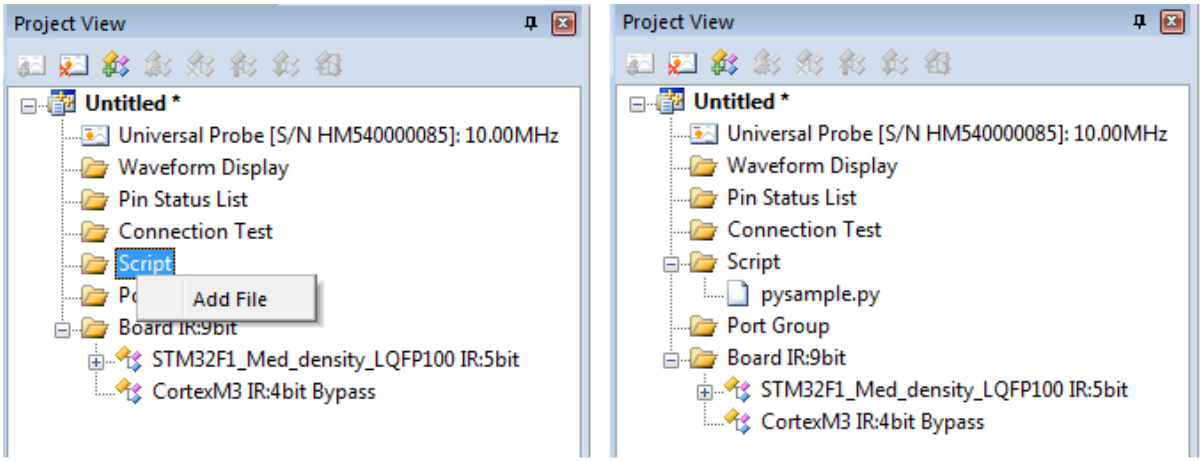


Figure 59

The Python interpreter (python32.dll) is used when a script file is executed. If you did not install Python when installing this software, provide an environment where Python can be executed. Use of Python 3.2 is assumed.
The version of the Python installer that comes with this software is 3.2.2.

2.6.2. Execute (file)

Tool button	Operation on the menu bar	Shortcut key
	Project -> Add Script File * Project View -> Script -> Right-click to open the context menu -> "Execute"	---

When clicking the above-mentioned button on the toolbar in the Script pane, the Python interpreter executes the file entered in the **File:** edit box. If the execution of the whole script file completes successfully, the file is registered in the "Script" tree in Project view unless it has already been registered. If an error occurs during the execution of the file, the file is not registered.

Descriptions of Python3 (except quit() and exit() commands), and interface functions for using the functions provided by this software can be used in a script file. For details of interface functions, refer to the "List of interface functions (by function)."

Selecting a script file from Project view also makes the Python interpreter execute the file.

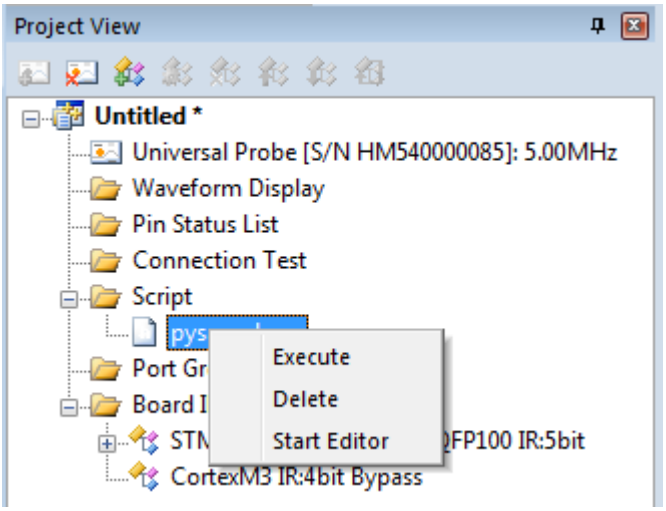


Figure 60

2.6.3. Execute (command line)

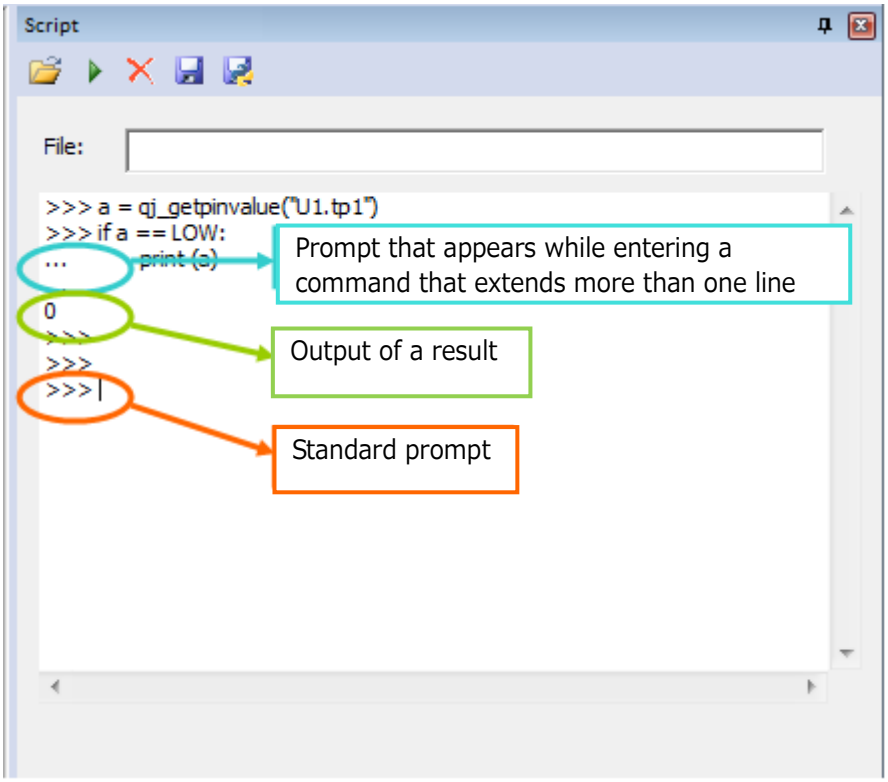


Figure 61

You can describe various Python commands in the standard I/O area on the Script pane and execute them in an interactive manner. They can be used after a new project is created or a project is opened following the startup of this software (However, most of the functions for this software cannot be used until a connection with a target is established). Generally, enter a command after the prompt ">>>." The prompt "..." appears in the middle of commands that extend over two or more lines such as an "if" statement or "for" statement. In describing scripts, be sure that they are indented correctly.

If character strings containing a line feed code are pasted on the standard I/O area, they are not immediately executed. They are executed when the Enter key is pressed after they are pasted.
If two or more lines of character strings are pasted and the Enter key is pressed, only the lowest character strings are executed.

The Python interpreter (python32.dll) is used when a command is executed. If you did not install Python when installing this software, provide an environment where Python can be executed. Use of Python 3.2 is assumed.
The version of the Python installer that comes with this software is 3.2.2.

Descriptions of Python3 (except quit() and exit() commands), and interface functions for using the functions provided by this software can be used in a script. For details of interface functions, refer to the "List of interface functions (by function)."

2.6.4. Save (in text format)

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar.	---

Clicking the above-mentioned button on the toolbar in the Script pane opens the Select File dialog box. The content in the Standard I/O edit box is written to the file with the specified name in text format.

2.6.5. Save (in script format)

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar.	---

Clicking the above-mentioned button on the toolbar in the Script pane opens the Select File dialog box. The content in the Standard I/O edit box is written to the file with the specified name. When the content is written, the character strings for prompts (">>>" and "...") are deleted from the command lines. Lines for execution results are prefixed with the character string ("#") to comment out. The file written here can be reused by describing in the **File Name:** edit box or by adding it to Project view.

2.6.6. Clear the display

Tool button	Operation on the menu bar	Shortcut key
	This operation cannot be performed from the menu bar.	---

Clicking the above-mentioned button on the toolbar in the Script pane clears the content in the Standard I/O edit box

3. Menu

This chapter explains the menus.

3.1. File

The "File" menu items are used to operate the files related to the project.

3.1.1. Create a new project

Creates a new project.
When using the product for the first time, select the following tool button or menu item to create a project.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Create New Project**	Ctrl + N

Executing the above menu item creates an untitled project in the Project view.

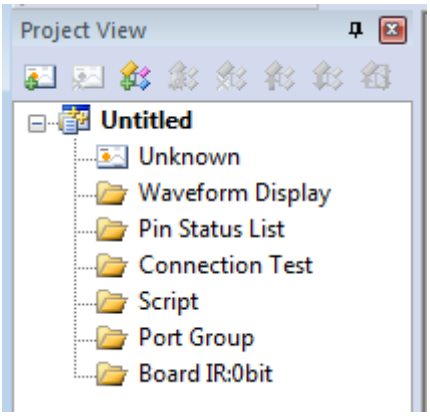


Figure 62

3.1.2. Open a project

Opens a previously created project file. The settings when you stopped the work lastly are restored.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Open Project**	Ctrl + O

Executing the above menu item displays the **Select File** dialog box. Select the project file you want to open and select **OK**. Information recorded in the project file is restored in the Project view.

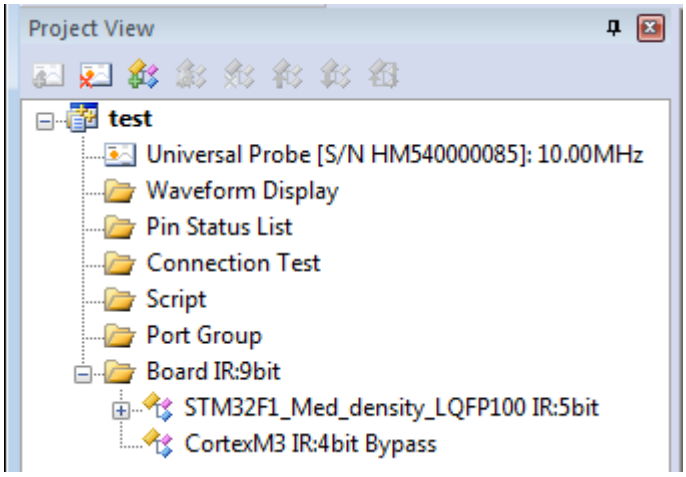


Figure 63

3.1.3. Save a project

Saves a variety of information currently set to the project file.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Project** **File** -> **Save Project as**	Ctrl + S

Executing the above menu item saves a variety of project settings to a file. If "Save" or "Save as" is selected when there is no file for the project to be saved, the **Select File** dialog box is displayed. Specify the saving destination and file name and select **Save** to save project settings to a file.

The extension of a project file is .bss.

The project file contains the following information:

- Individual ID of probe to be connected and clock frequency for boundary scan test (JTAG clock)
- Target device configuration for boundary scan test
- Port information registered in the Waveform Display window (waveform display pattern)
- Port information registered in the Pin Status List window (pin status list pattern) and the check status of each port
- Check status of **"Project >> Forced Output of JTAG Signal"** menu item

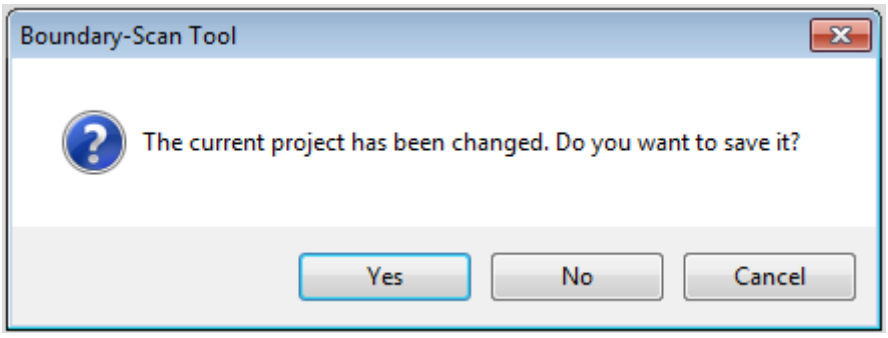
3.1.4. Close a project

Closes the currently open project.

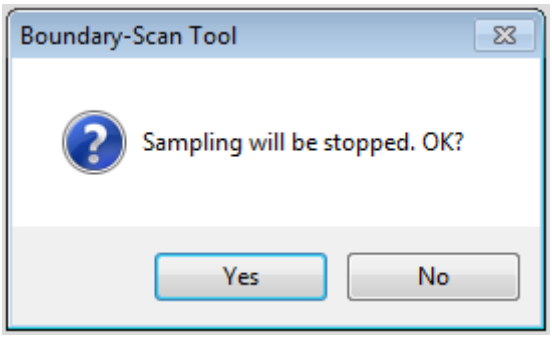
Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Close Project**	---

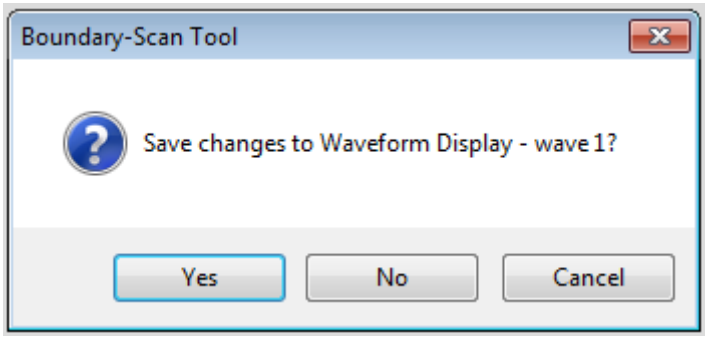
Executing the above menu item closes the currently open project and removes it from the Project view. The project is disconnected from the probe at this time if connected.

When the Create, Open or Close menu item is executed if a change is made to the already created project (project having an asterisk (*) at the right of its name), a message appears to ask if the currently open project is to be saved or not. Select **"Yes," "No" or "Cancel"** as needed and proceed to the next step.



If sampling is being performed in the Waveform Display window when the Create, Open or Close menu item is executed for a project, messages appear to ask **whether to stop measurement** and **whether to save the waveform**. Select **"Yes," "No" or "Cancel"** as needed and proceed to the next step.





3.1.5. Create a new Waveform window

Creates an empty waveform display pattern and displays it in the Waveform Display window.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Create New Waveform window**	Ctrl + W

3.1.6. Save a waveform

Saves observed waveform data to a waveform data file.
The extension of file is .bsw.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Save Waveform**	---

The saved file can be loaded by selecting **File** -> **Open Waveform in Viewer** in the menu.

3.1.7. Open a waveform in Viewer

Loads a saved waveform data file to the dedicated viewer.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Open Waveform in Viewer**	---

Executing the above menu item displays the Select File dialog box. When you select the saved waveform file (with an extension of .bsw), that waveform is opened in the Waveform window. When a waveform is opened, the following function cannot be used.

- Add a Port
- Create a Group
- Delete a Port Group
- Start Sampling
- Stop Sampling
- Specify a Sampling Interval

3.1.8. Create/open a new pin status list

Creates an empty pin status list and displays it in the Pin Status List window.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Create/Open New Pin Status List**	Ctrl + L

3.1.9. Save a pin status list in CSV format

Saves the current pin status list to a CSV file.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Pin Status List in CSV File**	---

3.1.10. Create a new Connection Test window

Creates an empty connection test and displays it in the Connection Test window.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Create new Connection Test Window/Open Connection Test Window**	Ctrl + T

3.1.11. Open a net list

Opens a dedicated net list file and displays the connection settings in the Connection Test window.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Open Net List**	Ctrl+Shift+T

3.1.12. Save a net list

Saves the connection settings in the Connection Test window to a dedicated net list file.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Net List**	---

3.1.13. Save connection test results

Saves the test results in the Connection Test window to a CSV file.

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Save Connection Test Results**	---

3.1.14. Preview Print

Displays a print preview.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Preview Print**	---

The currently displayed content is displayed in a single page. The sampling interval, printed range, length of time per graduation, time corresponding to the cursor, and times corresponding to the markers are shown below.

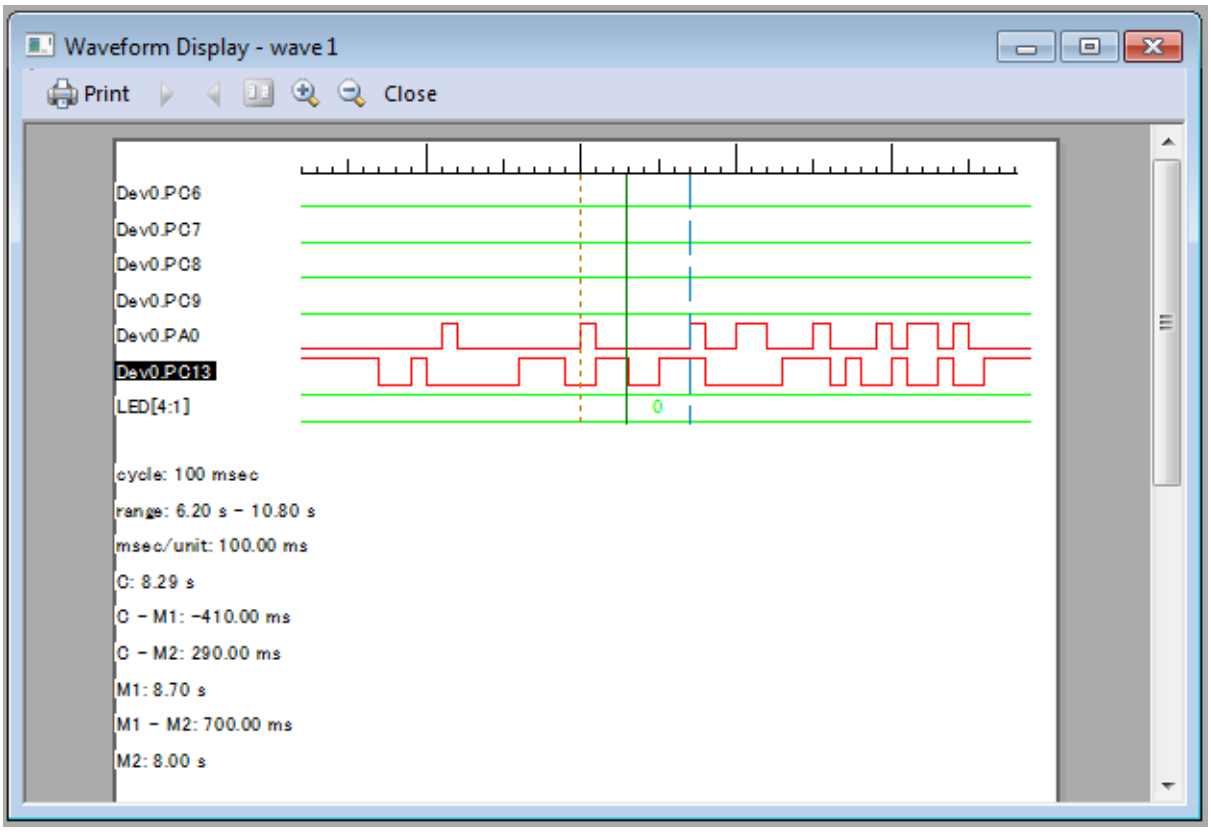


Figure 64

3.1.15. Set a Printer

Changes the settings of the printer to be used.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Set Printer**	---

3.1.16. Print

Prints observed waveforms. The currently displayed content is printed in a single page. The sampling interval, time corresponding to the cursor, time corresponding to the markers, and the length of time per one graduation are printed below.

Tool button	Operation on the menu bar	Shortcut key
	**File -> Print**	Ctrl + P

When the following dialog box appears, specify the printer, the range to be printed, the number of copies, and others.

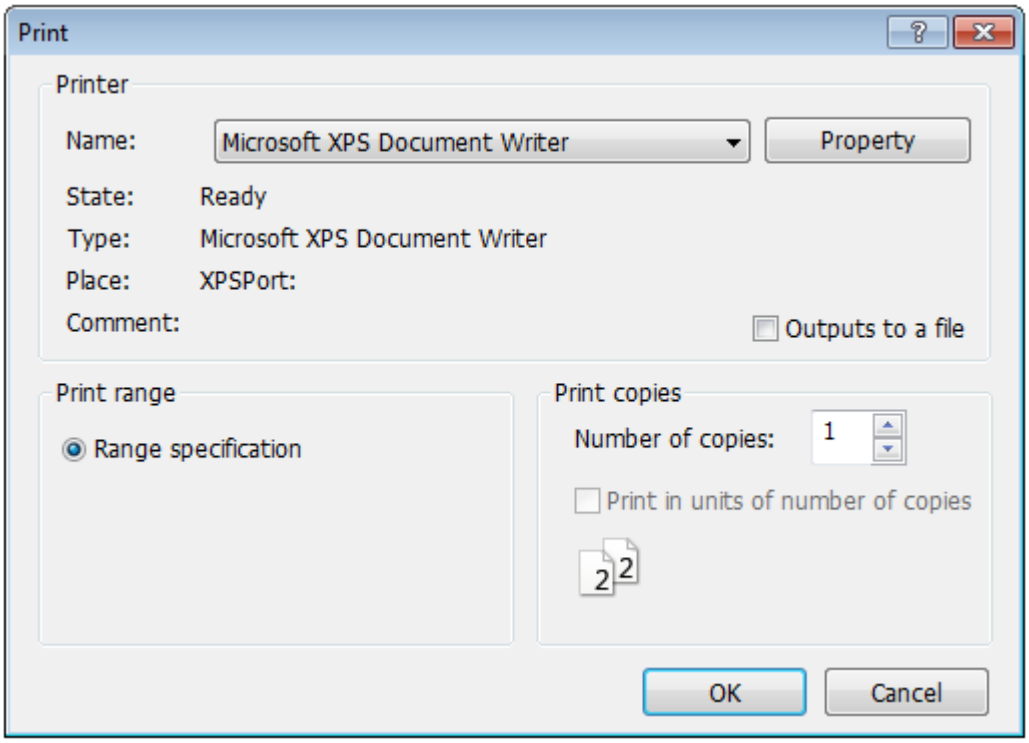


Figure 65

Preview the print before printing.

3.1.17. Exit application

Exits the Boundary-Scan Tool.

Tool button	Operation on the menu bar	Shortcut key
---	**File** -> **Exit Application**	---

3.2. View

The "View" menu contains items that are used to display/hide windows or customize the toolbars.

3.2.1. Toolbar and docking window

Sets display/hide of the toolbars and customize various items.

1) Standard

Specifies display/hide of the standard toolbar.

Tool button	Operation on the menu bar	Shortcut key
---	**View -> Toolbar and Docking window -> Standard**	---

2) Script

Specifies display/hide of the script pane.

Tool button	Operation on the menu bar	Shortcut key
---	**View -> Toolbar and Docking window -> Script**	---

3) Project view

Specifies display/hide of the toolbar on Project view.

Tool button	Operation on the menu bar	Shortcut key
---	**View -> Toolbar and Docking window -> Project View**	---

4) Customize

Customizes the toolbar.

Tool button	Operation on the menu bar	Shortcut key
---	**View -> Toolbar and Docking window -> Customize**	---

3.2.2. Status bar

Specifies display/hide of the status bar.

Tool button	Operation on the menu bar	Shortcut key
---	**View -> Status Bar**	---

3.2.3. Script

Specifies display/hide of the script pane.

Tool button	Operation on the menu bar	Shortcut key
---	**View -> Script**	---

3.3. Project

The "Project" menu sets communications with devices and the probe, and adds script files and port groups.

3.3.1. Connect

Connects between this software and the probe.

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Connect**	Ctrl + C

3.3.2. Disconnect

Disconnects between this software and the probe.

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Disconnect**	Ctrl+Shift+C

3.3.3. Add a device

Adds a device to the **"Board"** tree in Project view.

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Add Device**	---

3.3.4. Consistency test

Tests if a boundary scan test can be executed.

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Consistency Test**	---

3.3.5. Add a script file

Adds a Python script file to the **"Script"** tree in Project view.

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Add Script File**	---

3.3.6. Add a port group

Adds a group to the **"Port Group"** tree in Project view.

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Add Port Group**	Ctrl + G

3.3.7. Set a JTAG clock

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Set JTAG Clock**	---

Sets the frequency at which a boundary scan test is conducted. Selecting the above menu item displays the **"Set JTAG Clock"** dialog box. Select a desired frequency from the pull-down list or select **"Set Low-speed Clock"** from the pull-down list and then enter a frequency (slower than 625kHz) in the **"Low-speed Clock Frequency"** edit box. The specified frequency is set when **OK** is clicked.

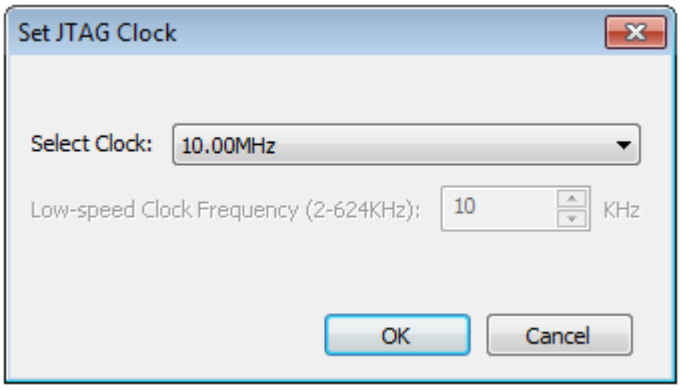


Figure 66

After OK is clicked, an operation test is conducted at the specified frequency. If the test is passed, that frequency is used. If the test is not passed, the frequency is decremented by one level until the test is passed. If **"Auto Setting"** is selected, the **lowest one** of the operating frequencies in the BSDL information of the devices is selected, and the highest frequency slower than the selected one is selected from the pull-down list and set.

3.3.8. Forced output of JTAG signal

Sets whether to output JTAG signals without checking the power supply status of the target when the device is automatically recognized or a boundary scan test is executed.
Checkmark this menu item when conducting a boundary scan test on a board whose power supply status cannot be detected because a power supply reference signal is not connected.

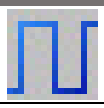
Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Forced Output of JTAG Signal**	---

3.4. Waveform Display

The "Display Waveform" menu contains items to control start/stop of the boundary scan.

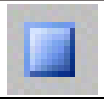
3.4.1. Start sampling (start measurement)

Starts measurement.

Tool button	Operation on the menu bar	Shortcut key
	**Display Waveform** -> **Start Sampling**	F7

3.4.2. Stop sampling (stop measurement)

Stops measurement.

Tool button	Operation on the menu bar	Shortcut key
	**Display Waveform** -> **Stop Sampling**	Ctrl + F7

3.5. Pin Status List

The "Pin Status List" menu contains items to control the Pin Status List window.

3.5.1. SAMPLE Command

The SAMPLE command is issued to the devices in the boundary scan chain to obtain the port status.

Tool button	Operation on the menu bar	Shortcut key
	**Pin Status List -> SAMPLE Command**	F5

3.5.2. Execute the EXTEST command

The EXTEST command is issued to the devices in the boundary scan chain to check the connection status of the I/O pins of these devices.

Tool button	Operation on the menu bar	Shortcut key
	**Pin Status List -> EXTEST Command**	Ctrl + F5

3.5.3. Display warning when executing EXTEST

When this menu item is checkmarked, a warning message that asks you whether you really want to execute EXTEST with these settings appears when executing the EXTEST command menu.

3.6. Connection Test

The "Connection Test" menu contains items to control the Connection Test window.

3.6.1. Learn expected values

Creates expected values according to the output values for each net in the Connection Test window.

Tool button	Operation on the menu bar	Shortcut key
	**Connection Test** -> **Learn Expected Values**	---

3.6.2. Connection test

Conducts a connection test on each net in the Connection Test window.

Tool button	Operation on the menu bar	Shortcut key
	**Connection Test** -> **Connection Test**	---

3.6.3. Display warning when outputting signal

When this menu item is checkmarked, a warning message that asks you whether you really want to execute EXTEST with these settings appears when executing Learn Expected Values and Connection Test menus.

3.7. Tools

The "Tools" menu contains items to set BSDL, license, or editor.

3.7.1. Manage BSDL

Tool button	Operation on the menu bar	Shortcut key
---	**Tools** -> **Manage BSDL**	Ctrl + B

Manages BSDL files used by this software.

Selecting the above menu item displays the "Manage BSDL" dialog box. When you select a manufacturer in the manufacturer list, devices of that manufacturer that have been registered in this software are listed. To add a new device, directly enter the path to the BSDL file in the "BSDL edit box" or click the [···] button and select the file. Clicking the [Read] button loads the specified BSDL file and adds the device to the list.

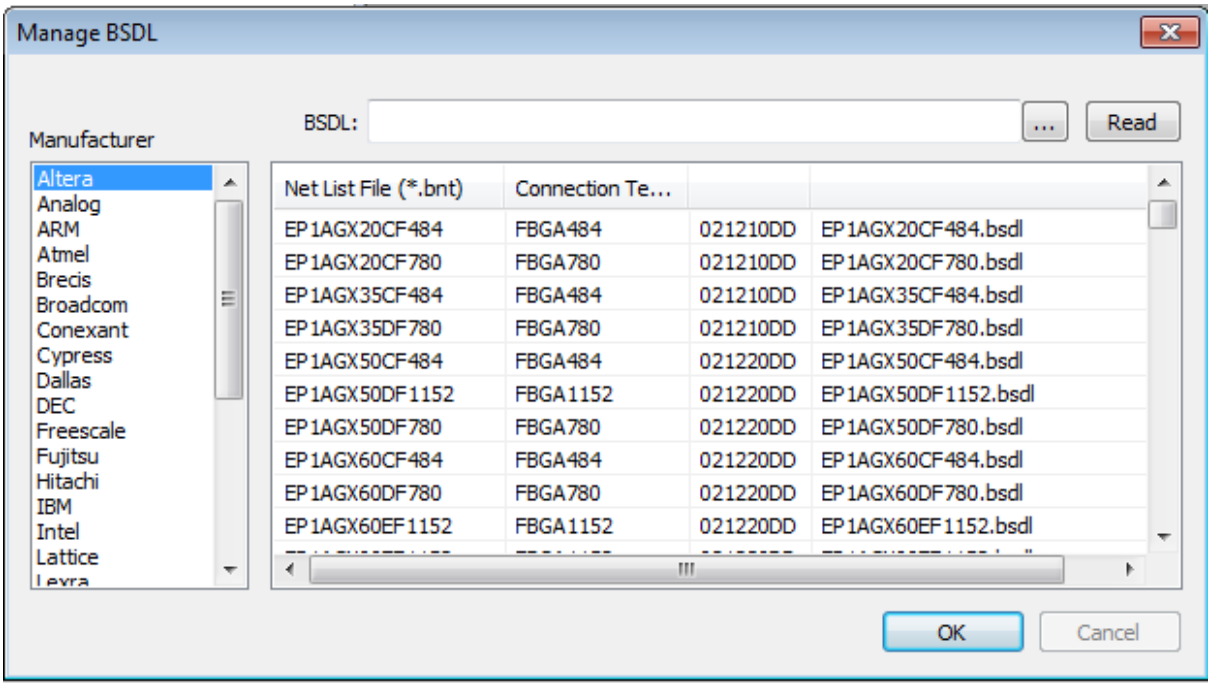


Figure 67

If the BSDL file for a device that has already been registered is loaded, the existing information is overwritten with the loaded information.

3.7.2. Obtain the probe license

Tool button	Operation on the menu bar	Shortcut key
---	**Tools** -> **Obtain Probe License**	---

This software employs the license system.
When a user notifies us the probe Serial No., the License Code will be issued.

(1) Entering the License Code

When this software starts, select a probe for which [Select Probe] dialog box shows “License not registered” or “Connectable” and click [Register License] button. The License Code entry window is displayed.
Enter the License Code received from us. This software becomes usable at the target probe.

Figure 68

(2) Notes on entering the License Code

Enter exactly the same License Code as the received information.

INFO If the License Code is rejected even though it matches the code in the received email, please contact us.

(3) Error display list

Table 13

Displayed Error	Meaning	Action
The License Code is incorrect.	Displayed when the License Code cannot be analyzed due to invalid format of License Code or for other reasons.	Check the email that includes the License Code and enter it again.
The License Code does not match the serial number. Please check if the License Code is matching the Serial number of Probe.	Displayed when the Serial No. of the used probe does not match the Serial No. of the probe to which the License Code is applied.	Check the email that includes the License Code and enter it again.

3.7.3. Specify an editor

Tool button	Operation on the menu bar	Shortcut key
---	**Tools** -> **Specify Editor**	---

Specifies an editor to be started from the context menu of a script file in Project view.

Selecting the above menu item displays the "Specify Editor" dialog box. The currently specified editor appears on the dialog box. The initial value is "Notepad" (notepad.exe). Enter the executable file for a desired editor directly in the edit box or click the file selection button ("...") and specify the editor in the Select File dialog box.
Specify the editor with the full path.

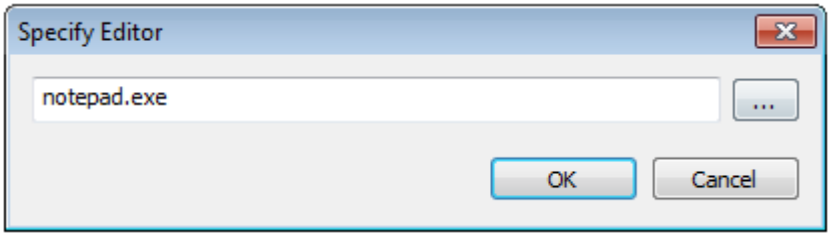


Figure 69

3.8. Help

The "Help" menu contains items to display help and version information.

3.8.1. Display help

Display the instruction manual for this software.
Selecting this command opens the page for instruction manual in our website.

Tool button	Operation on the menu bar	Shortcut key
---	**Help** → **Display Help**	F1

3.8.2. Version Information

Displays the version of this software.

Tool button	Operation on the menu bar	Shortcut key
	**Help** → **Version Information**	---

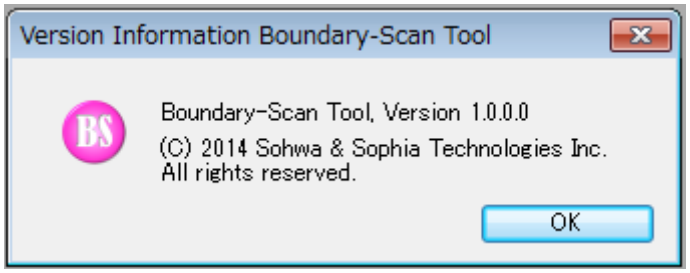


Figure 70

4. Windows

This chapter explains various windows.

4.1. Project View window

This window displays several settings for a project.
The items displayed in this window are described below.

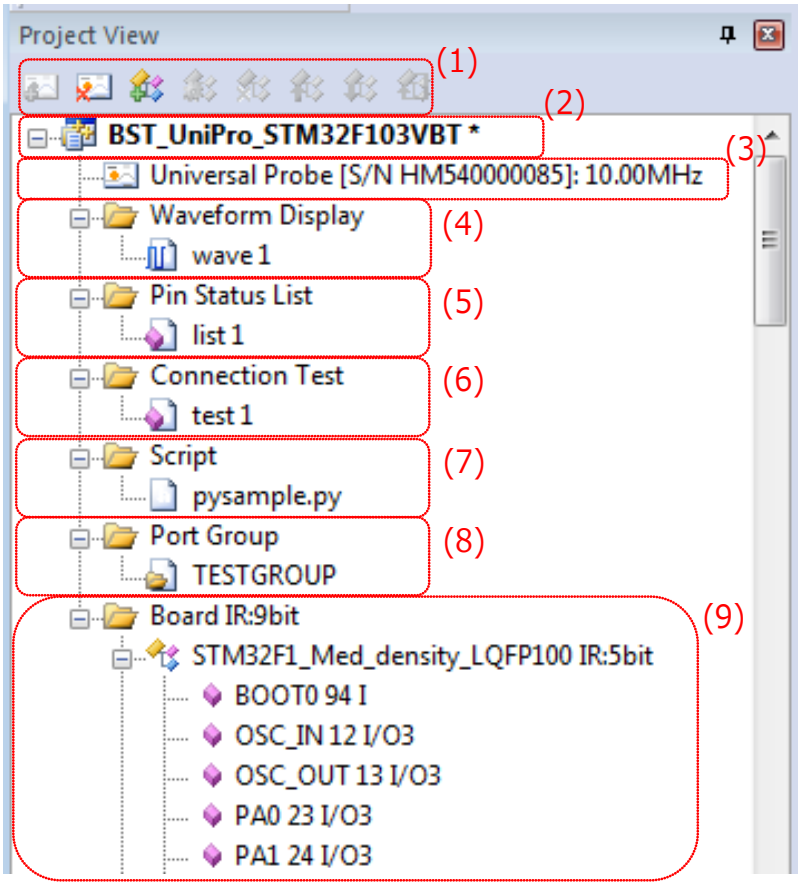


Figure 71

(1) Toolbar
(2) Project name
(3) Probe information
(4) Waveform display tree
(5) Pin status list tree
(6) Connection test tree
(7) Script tree
(8) Port group tree
(9) Board tree

4.1.1. Toolbar

Frequently used menus of Project view can be used from the toolbar.

4.1.2. Project name

The name of the current project is displayed. The project name is the character strings of the project file name excluding the extension.
If a change is made to the project, an asterisk **"*"** appears to the right of the project name.
This asterisk disappears when the project is saved.
It is judged that a change was made to the project if:

- the probe to be connected was decided or changed,
- the JTAG clock for the probe was changed,
- a waveform display pattern was newly created, renamed or deleted,
- the number of ports registered in a waveform display pattern was changed,
- the order of ports registered in a waveform display pattern was changed,
- a pin status list was newly created, renamed or deleted,
- the number of ports registered in a pin status list was changed,
- the order of ports registered in a pin status list was changed,
- the check status of a port(s) of a pin status list was changed,
- the devices that constitute the "Board" tag were changed, or
- the check status of "Forced Output of JTAG Signal" menu item was changed.

4.1.3. Probe information

Information on the probe to be connected is displayed.
The information displayed changes as follows, depending on the connection status to the probe.

Table 14

Initial value	Unknown
Probe connected	UNIVERSAL PROBE [probe serial No.] : JTAG clock
Probe disconnected	UNIVERSAL PROBE [probe serial No.] : JTAG clock: Not Connected

The probe information has the following context menus:

Table 15

Connect	Connects with the probe.
Disconnect	Disconnects from the probe.

4.1.4. Waveform display tree

"Waveform Display Pattern" is managed.
The "Waveform Display" column has the following context menu:

Table 16

Waveform Display	Creates a new "Waveform Display Pattern."

The default name of the new "Waveform Display Pattern" is "wave n" ("n" represents a serial number starting with "1").
"wave n" has the following context menu:

Table 17

Waveform Display	Displays the selected "wave n" in the Waveform Display window.
Change Name	Changes the name of the selected "wave n" to any name.
Delete	Deletes the selected "wave n" from the project.

4.1.5. Pin status list tree

"Pin Status List**"** is managed.
The "Pin Status List" column has the following context menu:

Table 18

Pin Status List	Creates a new "Pin Status List."

The default name of the new "Pin Status List" is **"**list n**"** ("n" represents a serial number starting with "1").
"list n" has the following context menu:

Table 19

Pin Status List	Displays the selected "list n" in the Pin Status List window.
Change Name	Changes the name of the selected "list n" to any name.
Delete	Deletes the selected "list n" from the project.

4.1.6. Connection test tree

"Connection Test List**"** is managed.
The "Connection Test" column has the following context menus:

Table 20

Connection Test	Creates a new "Connection Test List."
Open Net List	Opens a dedicated net list file to configure a connection test.

The default name of the new "Connection Test List" is **"**test n**"** ("n" represents a serial number starting with "1").
"test n" has the following context menus:

Table 21

Connection Test	Displays the selected "test n" in the Connection Test window.
Change Name	Changes the name of the selected "test n" to any name.
Delete	Deletes the selected "test n" from the project.

4.1.7. Script tree

"Script File" is managed.
The "Script" column has the following context menu:

Table 22

Add File	Adds a "Script File."

The name of the added Python script file appears under the "Script" tree.
"Script File" has the following context menus:

Table 23

Execute	Executes the selected script file. The results, etc. can be checked in the Script pane.
Delete	Deletes the selected script file from the project. The file itself is not deleted.
Start Editor	Opens the selected script file in an editor. Specify the editor from the Tools menu.

4.1.8. Port group tree

"Group" of any port is managed as a whole.
The "Port Group" column has the following context menu:

Table 24

Add Group	Opens the Create Group dialog box. Create a new "Group."

The group name set in the Create Group dialog box appears under the "Port Group" tree.
"Group" has the following context menus:

Table 25

Edit Group	Edits the selected "Group." When the Create Group dialog box opens, edit where changes are needed.
Delete	Deletes the selected "Group" from the project.

4.1.9. Board tree

The "Board" column displays the total bit length of all the devices obtained from the target board following automatic recognition of devices or a consistency test.
The initial value is "0" and is updated when the "Connect" or "Consistency Test" menu item is executed.
In addition, "Device" is managed.
What appears depends on the information known about the device.

Table 26

Immediately after the device is added	Unknown No BSDL information is available.

Immediately after the device information is edited	Device name IR:XXXX The device has been specified with "Edit Device," but no other details have been set. The IR register length has not been found.
After the IR register length is edited	Device name IR:n bit The device information appears in this format when the IR register length is manually changed after the device passed a consistency test or was automatically recognized.

The "Board" column has the following context menus:

Table 27

Add Device	Creates a new device in the "Board" tree.
Consistency test	Tests whether boundary scan test can be executed with the devices registered in the project.

The default name of a newly created "Device" is "Unknown."
"Device" has the following context menus:

Table 28

Edit Device	Opens the "Edit Device" dialog box. Updates the information on the selected device.
Delete Device	Deletes the selected device from the project.
Move Device Upward	The selected device is switched with the device above it. The position of the boundary scan chain in the project is also switched.
Move Device Downward	The selected device is switched with the device below it. The position of the boundary scan chain in the project is also switched.
Device Properties	Displays the information on the selected device. In the "Device Properties" dialog box, the bit length of the IR register can be changed and the bypass can be set.

Below each device appears the "Port" obtained from the BSDL information. Devices or ports can be registered in a "Waveform Display Pattern" and "Pin Status List" by dragging and dropping them in the Waveform Display window and the Pin Status List window.
The information on each port is displayed in the following order: "Port name," "Pin number," "Type."

4.2. Waveform Display window

This window displays observed waveforms. The time per one graduation appears at the end of the status bar. Waveforms for input ports, output ports, and high impedance ports are displayed in red, green, and blue, respectively.
The items displayed in this window are described below.

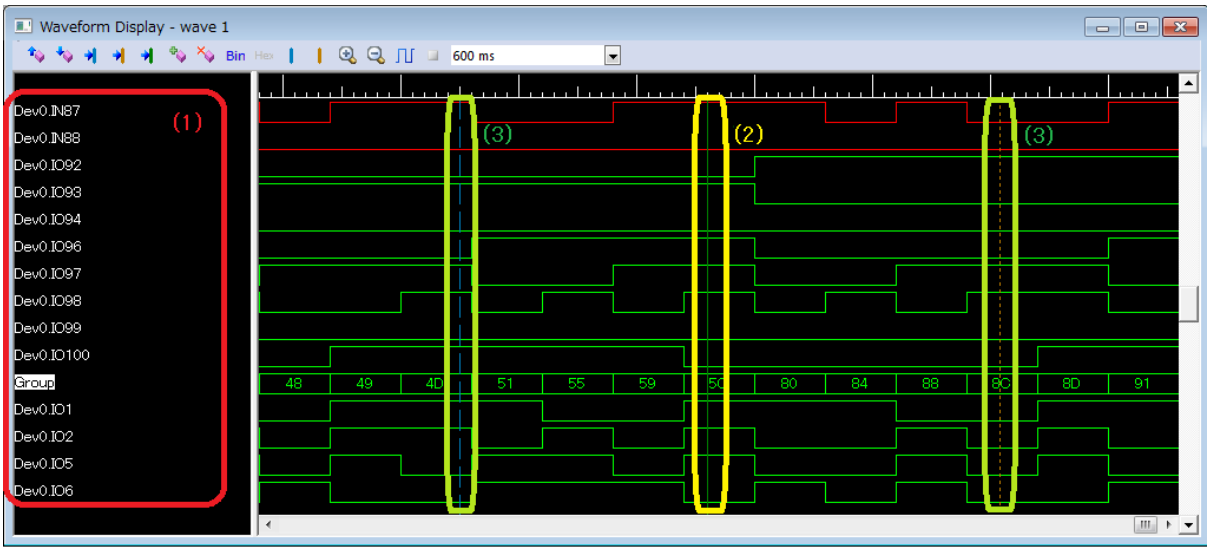


Figure 72

(1) Port Group Name List column
(2) Cursor (green solid line)
(3) Markers (blue and orange dashed lines)

4.2.1. Port group name list column

Port group names are listed.Ports are listed as "Dev" + JTAG chain No. + "." + port name.
Placing the mouse cursor on a group name displays the ports that constitute the bus from the MSB in a tooltip.
Select a port group and right-click on it to open the context menu, from which the following operations can be performed:

- Move
- Delete

4.2.2. Cursor (green solid line)

The position of the cursor is indicated by a green line.

4.2.3. Markers (blue and orange dashed lines)

The positions of markers are indicated.
Two markers can be set:

- Marker 1: Indicated by a blue dashed line.
- Marker 2: Indicated by an orange dashed line.

4.3. Pin Status List window

This window is used to monitor the pin status of each device and conduct a boundary scan test. The items displayed in this window are described below.

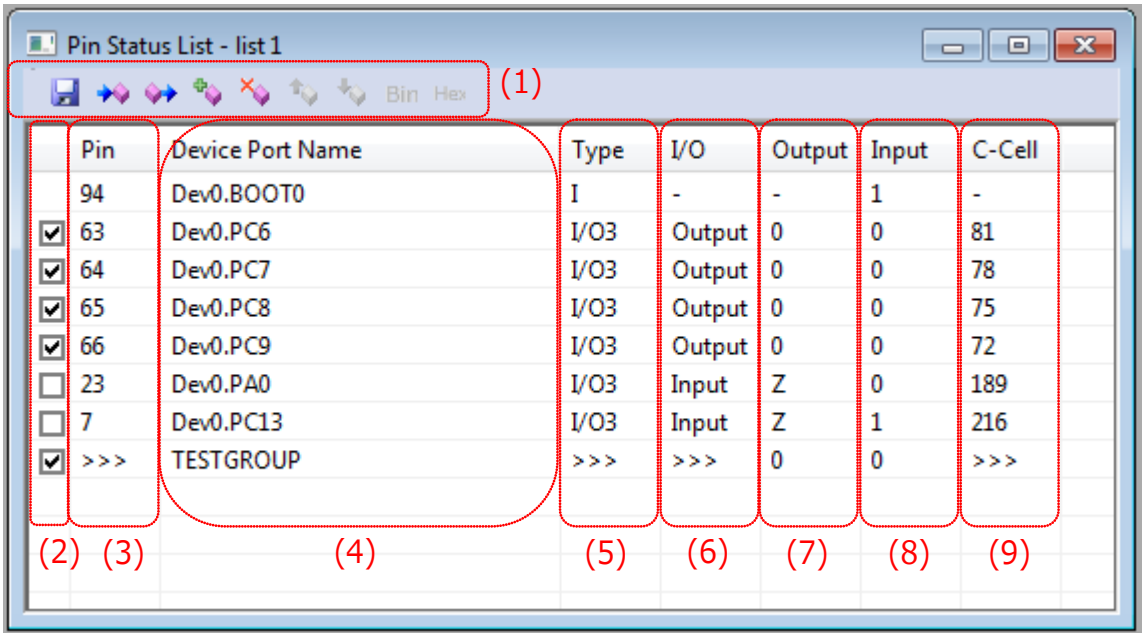


Figure 73

(1) Toolbar
(2) Checkbox column
(3) Pin column
(4) Device Port Name column
(5) Type column
(6) I/O column
(7) Output column
(8) Input column
(9) C-Cell (Control Cell) column

4.3.1. Toolbar

Frequently used menus of the Pin Status List window can be used from the toolbar.

4.3.2. Checkbox column

Checkbox for output only or I/O port appears here. No checkbox for input only ports appears.
A checkbox appears for a group that contains output only port or I/O port for at least one constituent port. No checkbox appears for a group that contains only input only ports. The following items can be set for the checkmarked ports.

Table 29

Output only port	By clicking the "Output" column, an output value can be set.
I/O port	By clicking the "I/O" column, an I/O direction of the port can be set. By setting the "I/O" column to "Output" and then clicking the "Output" column, an output value can be set.
Group	By clicking the "Output" column, output values can be set for each port.

Unchecked ports are treated as follows when a boundary scan test is conducted.

Table 30

Output only port	An output value of "Z" (Disable) is assumed (for a port with a control cell). An output value of "0" (Low) is assumed (for a port without control cell).
I/O port	Treated as an input port.

INFO

If the checkmark of the column has been changed, the checkmarks for other ports that have the same number as that port in the C-Cell column are also changed.
If the checkmark for a group has been changed, the checkmarks for all the ports that constitute the group are also changed.

4.3.3. Pin column

The pin numbers of the port on the device are displayed.
For a group, the symbol ">>>" is displayed. Placing the mouse cursor on the column displays the pin numbers of the port that constitutes the group in a pop-up.

4.3.4. Device Port Name column

Port names are displayed. The default display format is "Dev n.port name."
("n" represents the position of the device in the boundary scan chain starting with "0").
If the port name is changed by "Set Device Reference Name" or "Change Port Name," that name is displayed.
For a group, the group name is displayed. Placing the mouse cursor on the column displays the names of the port that constitutes the group in a pop-up.

4.3.5. Type column

The I/O type of the port is displayed. The following values are displayed:

Table 31

I	Input only port.
O	Output only port.
O3	Output only port. Port with 3-state output whose "function" is defined as "OUTPUT3" in the BSDL.
I/O	I/O port.
I/O3	I/O port. Port with 3-state output whose "function" is defined as "OUTPUT3" in the BSDL.
>>>	Group. Placing the mouse cursor on the column displays the types of the port that constitutes the group in a pop-up.

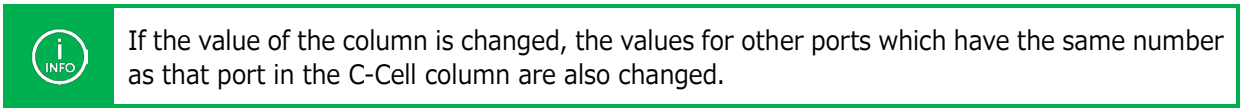
4.3.6. I/O column

The I/O direction of the port is displayed or set.
For checkmarked I/O port, the direction can be selected from "Input" or "Output."
For input only and output only ports, the value of this column is always **"-."**
For a group, the symbol ">>>" is displayed. The value cannot be selected.
Placing the mouse cursor on the column displays the I/O of the port that constitutes the group in a pop-up.

If the value is changed from "Input" to "Output" and vice versa, the value in the "Output" column is also changed as follows:

Table 32

"Input" > "Output"	The value in the "Output" column changes to "0" (Low output).
"Output" > "Input"	The value in the "Output" column changes to "Z" (high impedance).

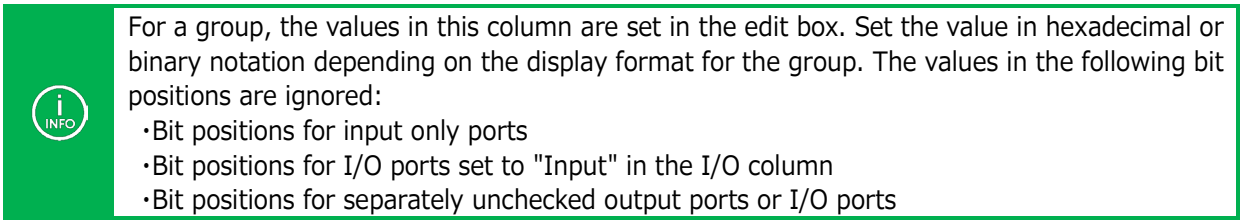


4.3.7. Output column

The output values of the port are displayed or set.
For checkmarked output only or I/O port, the output value can be selected.
For input only ports, the value of this column is always **"-."**
The following values can be selected as the output value:

Table 33

0	Outputs Low. This value can be selected for the port types "O," "O3," "I/O," and "I/O3."
1	Outputs High. This value can be selected for the port types "O," "O3," "I/O," and "I/O3."
Z	Outputs high impedance. This value can be selected for the "O3" port.



4.3.8. Input column

The input values of the port are displayed.
For output only ports, the value of this column is always **"-."**

4.3.9. C-Cell (Control Cell) column

The control cell numbers of the port defined in the BSDL are displayed.
For ports with no control cell number defined, the value of this column is always **"-."**
For a group, the symbol ">>>" is displayed. Placing the mouse cursor on the column displays the control cell numbers of the port that constitutes the group in a pop-up.

4.4. Connection Test window

This is a window to test connections between the signals of each device using the boundary scan test.
The items displayed in this window are described below.

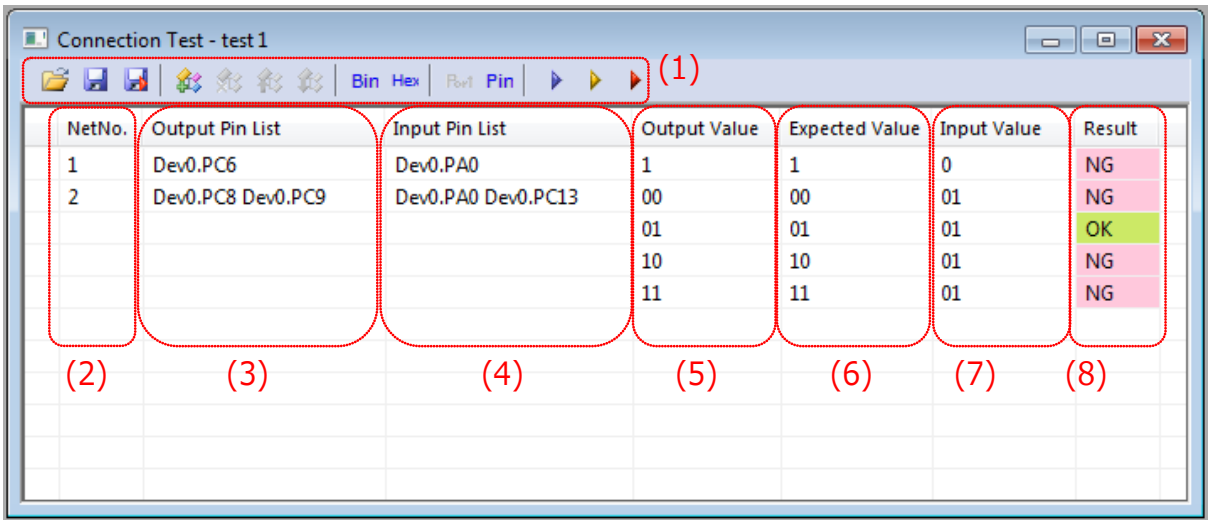


Figure 74

(1) Toolbar
(2) Net No. column
(3) Output Pin List column
(4) Input Pin List column
(5) Output Value column
(6) Expected Value column
(7) Input Value column
(8) Result column

4.4.1. Toolbar

Major menus of the Connection Test window can be used from the toolbar.

4.4.2. Net No. column

Serial numbers given to connection settings between pins. The net on the top is always given the number "1." When a net is added, deleted, or moved upward or downward, the nets are renumbered.

4.4.3. Output Pin List column

The list of pin or group names from which signals are output in the connection test.
If an input only pin is registered, the background color of the column changes to pink to indicate that there is an incorrect setting.
The Output Pin List column corresponds to the Output Value column one-to-one, for example, the first (leftmost) pin in the list corresponds to the MSB of the Output Value column.
Placing the mouse cursor on the column displays the pin numbers of the port that constitutes the group in a pop-up.

4.4.4. Input Pin List column

The list of pin or group names to which signals are input in the connection test.
If an output only pin is registered, the background color of the column changes to pink to indicate that there is an incorrect setting.
The Input Pin List column corresponds to the Expected Values and Input Values one-to-one, for example, the first (leftmost) pin in the list corresponds to the MSB of the Expected Value column and Input Value column.
Placing the mouse cursor on the column displays the names of the port that constitutes the group in a pop-up.

4.4.5. Output Value column

Sets values output from the pins of the output pin list when executing a connection test.
The Output Value column corresponds to the output pin list column one-to-one, for example, the MSB of the Output Value column corresponds to the head (leftmost) of the output pin list.
Values displayed in hexadecimal notation are suffixed with "H."
Editing the Output Pin List column changes the status to "not set."

4.4.6. Expected Value column

Sets values that each pin in the input pin list should take when executing the connection test.
The Expected Value column corresponds to the input pin list column one-to-one, for example, the MSB of the Expected Value column corresponds to the head (leftmost) of the input pin list.
If the letter "x" is set, the corresponding bit is not compared with the input value in the connection test (it is judged as OK).
Editing the Output Pin List, Input List Column, and Output Value columns changes the status to "not set."

4.4.7. Input Value column

The values taken by each pin in the input pin list at the time of the connection test are displayed.
The Input Value column corresponds to the input pin list column one-to-one, for example, the MSB of the Input Value column corresponds to the head (leftmost) of the input pin list.

4.4.8. Result column

The results of comparison of the input values with expected values at the time of the connection test of each row are displayed.
If these values are the same (the values expected from the output values were input), "OK" appears and the background color of the Result column changes to green.
If these values are different (the values expected from the output values were not input), "NG" appears and the background color of the Result column changes to pink.

4.5. Edit Device/Manage BSDL dialog box

This dialog box is used to select device information, called from the "Edit Device" menu item.
The dialog box to manage BSDL files, called from the "Manage BSDL" menu item, is also explained in this section.
The items displayed in this window are described below.

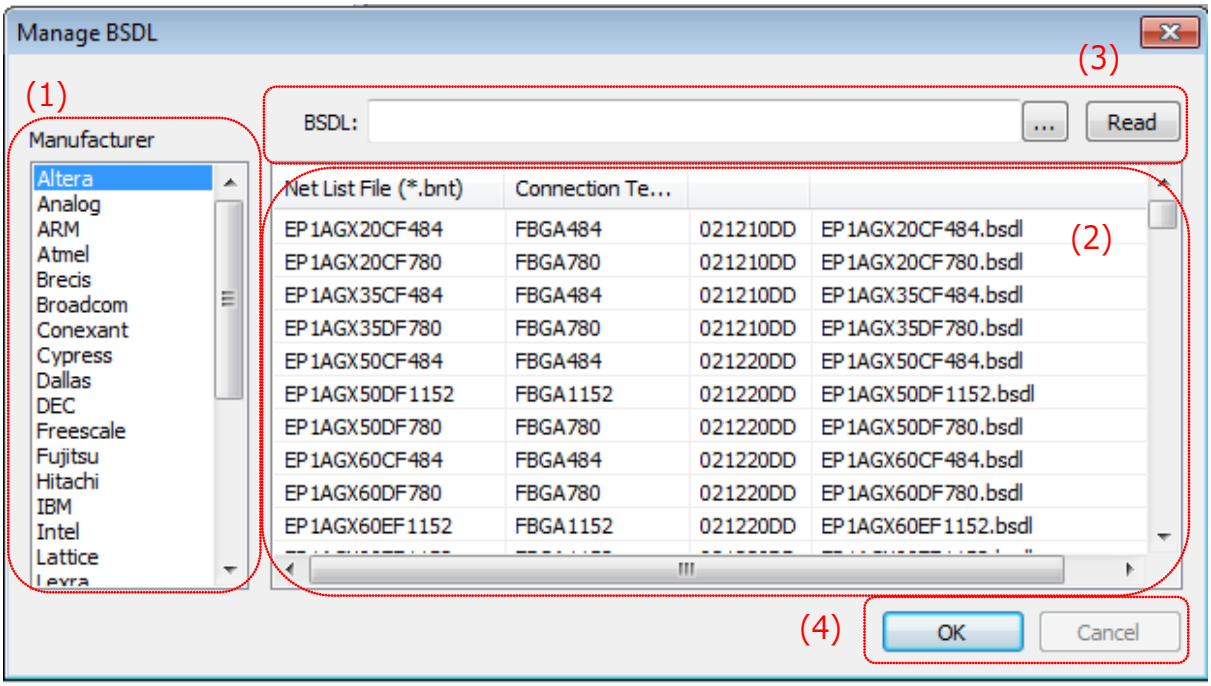


Figure 75

(1) Manufacturer list
(2) Device list
(3) BSDL file edit
(4) OK and Cancel

4.5.1. Manufacturer list

Device manufacturers are listed. Selecting a manufacturer lists devices from the manufacturer in the device list.
Devices are listed based on the "IDCODE" command in the BSDL. Devices whose manufacturers could not be identified by analyzing the BSDL are registered in "Unknown" in the manufacturer list.

4.5.2. Device list

Devices from the manufacturer selected in the manufacturer list are listed.
The columns of this list provide the following information.

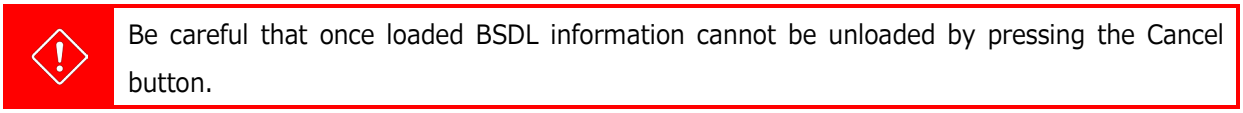
Table 34

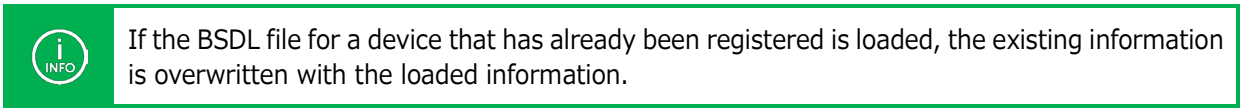
Device name	Device name obtained by analyzing the BSDL.
Package	QFP and other device package information obtained by analyzing the BSDL.
IDCODE	Value of the IDCODE command obtained by analyzing the BSDL.
BSDL file	Name of the BSDL file used to analyze the device information.

4.5.3. BSDL file edit

If the desired device is not found in the device list, the BSDL file for the device can be specified in this box to add to the list.
Follow the procedure below to add a device.

- Enter the full path to the BSDL file for the desired device in the BSDL: edit box.
  Alternatively, click the [⋯] button to call the Select File dialog box and then specify the BSDL file.
- Click the [Read] button to load the BSDL file.
- After the loaded BSDL file is analyzed successfully, that device is registered in the list of devices from the manufacturer.





4.5.4. OK and Cancel

These buttons on the "Edit Device" dialog box and on the "Manage BSDL" dialog box function as follows:

Table 35

Edit Device	OK	This button is enabled when a device is selected in the device list. Clicking this button closes the dialog box and updates the device information in Project view with the selected device information.
	Cancel	This button is always enabled. Clicking this button closes the dialog box. This button does not affect the project.
Manage BSDL	OK	This button is always enabled. Clicking this button closes the dialog box. This button does not affect the project.
	Cancel	This button is always grayed out.

4.6. Set JTAG Clock dialog box

This dialog window is used to set a clock frequency between the probe and the target device when executing the boundary scan test.
The items displayed in this window are described below.

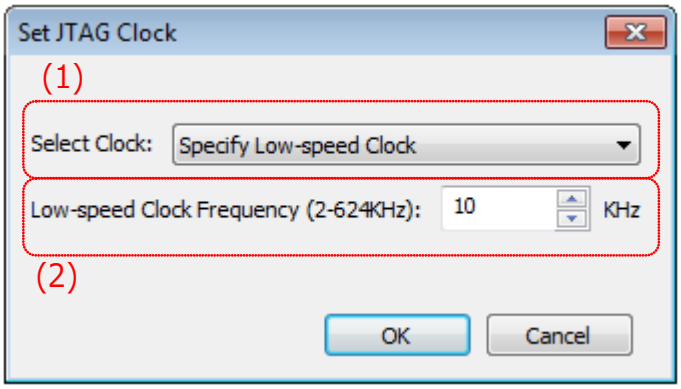


Figure 76

(1) Select Clock
(2) Low-speed Clock edit

4.6.1 Select Clock

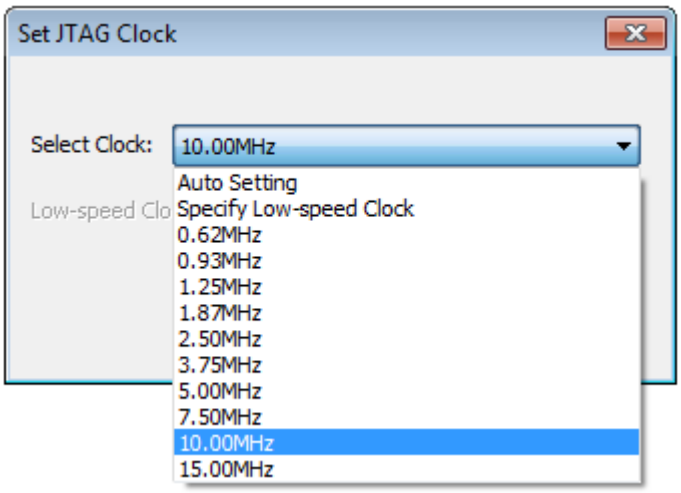


Figure 77

The frequencies that can be set for the probe are displayed in the pull-down list.
The initial value for project is "Specify Low-speed Clock."
Details of each item are as shown below.

Table 36

Auto Setting	Automatically sets the clock frequency. If the target that can be automatically recognized is connected or the device is set to the project, select the lowest frequency from the BSDL information on each device and set the highest frequency that is slower than that value and can be selected from the pull-down list.
Specify Low-speed Clock	Set a frequency slower than 1MHz in the **Low-speed Clock edit box** in the pull-down list. The initial value for the project is "Specify Low-speed Clock" and "10kHz."
Other frequencies	Because this value is set based on whether it works or not, regardless of BSDL information, a frequency faster than the operating frequency specified in the BSDL may be set. Because this value is set based on whether it works or not, regardless of BSDL information, a frequency faster than the operating frequency specified in the BSDL may be set.

4.6.2 Low-speed Clock edit box

JTAG communication between the probe and the target device uses a relatively low frequency slower than 1MHz. Selecting "Specify Low-speed Clock" from the **Select Clock** pull-down list enables this edit box. The value range is from 2 to 624 (unit: kHz).

4.7. Script pane window

This window is used to execute Python scripts.
Script can be executed by specifying a file or line by line on the command line.
The items displayed in this window are described below.

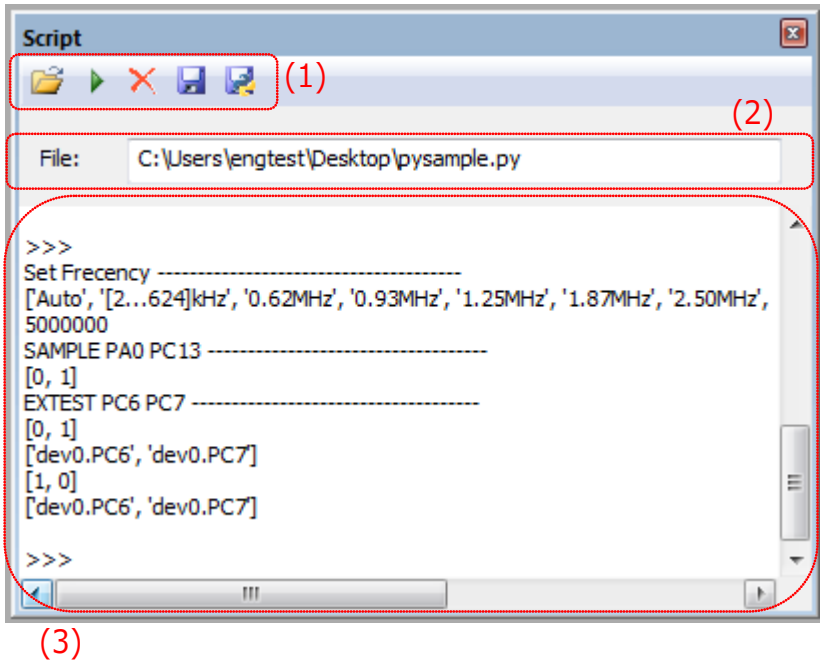


Figure 78

(1) Toolbar
(2) File Name edit box
(3) Standard I/O edit box

4.7.1. Toolbar

All the functions executed on the Script pane are used from the toolbar.

4.7.2. File Name edit box

Enter/display the name of a Python script file to be executed as a batch file.
Enter the file name by one of the following methods:

- Press the "Open File" button on the toolbar to select the desired file.
- Directly enter to the edit box.
- Select "Execute" from the context menu of the script file tag in Project view.

Pressing the Enter key while a file name is entered executes that file. (This is equivalent to the Execute button on the toolbar).
After the file has been executed successfully, the file is registered in the tree in Project view unless it has already been registered.
To enter a file name directly, enter the full path.

4.7.3. Standard I/O edit box

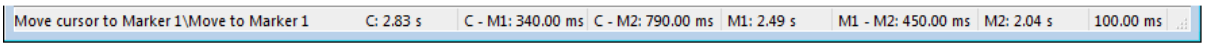
This edit box is used to print with the print() command and input with input() command in a Python script file or enter and execute scripts directly.
Descriptions of Python3 and the interface functions provided by this software can be used for script files or direct entry of commands.
The following operations can be performed in the edit box:

Table 37

Left and right arrow keys	Moves the cursor to the left or right. The cursor can move only to the right of prompt (">>>" or "...").
Up and down arrow keys	• Up: Tracks backward the directly entered command history. • Down: Tracks forward the directly entered command history.
Enter key	Sends the content following the prompt to the `Python` interpreter. It is executed if it conforms to the `Python` grammar. If not, error information is output.
Ctrl+C	Copies character strings in the selected area to the clipboard.
Ctrl+V	Pastes character strings copied to the clipboard. Be careful when pasting character strings that do not conform to the `Python` grammar.
Left mouse button	Used to place the cursor on a certain point or to select a range in the edit box.

4.8. Status bar

The status bar is displayed at the bottom of the window and indicates the explanation of each selection item, operation status, and length of time between points of trigger, cursor, and markers.



(1) (2) (3) (4) (5) (6) (7) (8)

Figure 79

(1) Displays brief explanation of each command.
(2) Displays the length of time from the beginning of the sampling result to the cursor.
(3) Displays the length of time between the cursor and the marker 1.
(4) Displays the length of time between the cursor and the marker 2.
(5) Displays the length of time from the beginning of the sampling result to the marker 1.
(6) Displays the length of time between the marker 1 and the marker 2.
(7) Displays the length of time from the beginning of the sampling result to the marker 2.
(8) Displays the length of time per one graduation.

5. Script

5.1. List of Interface Functions

To use each function of this software from the Python script, we offer interface functions. This page explains the meanings and examples of use of each interface function by categories.

5.1.1. Functions related to device setting

qj_setdevicereference(*"device", "newname"*)

Function: setting of device reference name Argument type: text string, text string Return value: none

Gives a reference name to the specified device. This is equivalent to the "Set Device Reference Name" function in the Project view. This function makes it possible to access the device by using its original name (dev n) or the name assigned with this function. The name set here is also displayed as the device reference name in Project view and the reference name in Device Properties.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified device does not exist on the board.

Example of use:

```
>>> qj_setdevicereference("dev0", "U1")
>>> a = qj_getpinvalue("U1.pin0")
>>> print (a)
0
```

qj_setpinname(*"device", "pin", "newname"*)

Function: change of pin name Argument type: text string, text string, text string Return value: none

Assigns "newname" to the name of the specified pin on the specified device. This is equivalent to the "Change Port Name" function in the Project view. This function makes it possible to access the device by using the original name of the pin or the name assigned with this function. The name set here is also displayed as the port name in Project view.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.

Example of use:

```
>>> qj_setpinname("dev0", "pin0", "addr0")
>>> a = qj_getpinvalue("dev0.addr0")
>>> print (a)
0
```

qj_setpinnamefile(*"device", "filename"*)

Function: change of pin name Argument type: text string, text string Return value: none

Changes the names of pins on the specified device altogether according to the contents of the specified file. This is equivalent to the file specification of the "Change Port Name" function in Project view. This function makes it possible to access the device by using the original name of the pin or the name assigned with this function. The name set here is also displayed as the port name in Project view. Specify the file name with the full path. However, due to the specifications of the escape sequence of Python, double the "¥" character for each directory.

Example of use:

```
>>> qj_setpinnamefile("dev0", "C:¥¥Query-J¥¥samplenamefile.txt")
>>> a = qj_getpinvalue("dev0.addr0")
>>> print (a)
0
```

qj_newgroup(*"group", ["pin", ...], local*)	**Function: group formation Argument type: text string, text string list, int Return value: none**

Forms a group of pins in the specified pin list with "group" as its name. If a group with the same name already exists, it is overwritten. The beginning of the list corresponds to the MSB of the group. If the third argument is "0," this function registers the group to Project view. Otherwise, the group is not registered to Project view.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2"]
>>> qj_newgroup("GROUP0", list, 0)
>>> qj_setgroupvalue("GROUP0", 5, 7)
>>> vallist = qj_getpinvaluelist(list)
>>> print (vallist)
[1, 0, 1]
```

qj_getgroupmember(*"group"*)	**Function: acquisition of names of pins constituting group Argument type: text string Return value: text string list**

Returns the list of names of pins that constitute the specified group.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified group does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2"]
>>> qj_newgroup("GROUP0", list, 0)
>>> namelist = qj_getgroupmember("GROUP0")
>>> print (namelist)
['dev0.pin0', 'dev0.pin1', 'dev0.pin2']
```

qj_setbypass(*"device", onoff*)	**Function: BYPASS setting of device Argument type: text string, int Return value: none**

Set BYPASS to the specified device. If the value of "onoff" is "1," the device will be handled as BYPASS in the subsequent boundary scan function such as qj_dosample() and qj_doextest(), and the value of each pin of the device will not be updated. If the value of "onoff" is "0," the device will be handled as a normal device and the value of each pin of the device will be updated. If the value of "onoff" is other than the above, it is handled as "true(1)" in accordance with the rule of Python.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified device does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "dev1.pin0"]
>>> qj_setextestlist(list)
>>> qj_setbypass("dev0", 1)
>>> qj_doextest()
>>> vallist = qj_getpinvaluelist(list)
```

qj_getdeviceproperty(*"device"*)	**Function: acquisition of device property Argument type: none Return value: dictionary**

Returns various information of the specified device. The values of dictionary type keys are as follows:

Table 38

"ref"	Key of device reference name. Its value is the "Device Reference Name" text string in the Device Properties dialog box.
"device"	Key of device name. Its value is the "Device Name" text string in the Device Properties dialog box.
"package"	Key of device package. Its value is the "Package" text string in the Device Properties dialog box.
"chain"	Key of device number in the boundary scan chain. Its value is the int value of "JTAG Chain" in the Device Properties dialog box.
"bsdlid"	Key of the device ID written in the BSDL file. Its value is the "IDCODE (BSDL)" text string in the Device Properties dialog box.
"readid"	Key of the ID read from the device by IDCODE command. Its value is the "IDCODE (device)" text string in the Device Properties dialog box.
"brlen"	Key of the boundary register length of the device. Its value is the int value of "BR Length" in the Device Properties dialog box.
"bsdlir"	Key of the IR register length of the device that is written in the BSDL file. Its value is the int value of "IR Length (BSDL)" in the Device Properties dialog box.
"readir"	Key of the IR register length of the device that is read from the boundary scan chain. Its value is the int value of "IR Length (detection/user setting)" in the Device Properties dialog box.
"bypass"	Key of the BYPASS setting of the device. Its value is a text string converted from the "BYPASS Setting" in the Device Properties dialog box. This function returns "true" when BYPASS setting is specified.

In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified device does not exist on the board.

Example of use:

```
>>> dic = qj_getdeviceproperty("dev0")
>>> print (dic)
{'readid': '001A200F', 'chain': 0, 'package': 'TBP_208A', 'bsdlir': 32, 'bsdlid': '001A200F', 'readir': 32, 'brlen': 385, 'bypass': 'false', 'device': 'SH7705', 'ref': 'U1'}
```

5.1.2. Functions related to value setting

qj_setpinvalue(*"pin"*, *value*)	**Function: setting of pin value Argument type: text string, int** **Return value: none**

Sets the specified value to pin value. This function is mainly used for setting the output value when driving by the qj_doextest() function. When an input only pin is specified, this function has no effect. When HIGH_Z(2) is set, output is prohibited. In particular, note that if HIGH_Z is set to the input/output pin, the input value will be returned to later function calls such as qj_getpinvalue().
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1"]
>>> qj_setextestlist(list)
>>> qj_setpinvalue("dev0.pin0", LOW)
>>> qj_doextest()
>>> vallist = qj_getpinvaluelist(list)
>>> print (vallist)
[0, 0]
```

qj_getpinvalue(*"pin"*)	**Function: acquisition of pin value Argument type: text string** **Return value: int**

Returns the value of the specified pin.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.

Example of use:

```
>>> qj_dosample()
>>> a = qj_getpinvalue("dev0.pin0")
>>> print (a)
0
```

qj_setpinvaluelist(*["pin", ...], [value, ...]*)	**Function: setting of pin value Argument type: text string list, int list** **Return value: none**

Sets each specified value to one or more specified pins. This function is mainly used for setting the output value when driving by the qj_doextest() function. When an input only pin is specified, this function has no effect. When HIGH_Z(2) is set, output is prohibited. In particular, note that if HIGH_Z is set to the input/output pin, the input value will be returned to later function calls such as qj_getpinvalue().
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.
- The number of elements differs between the pin list and the value list.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1"]
>>> vallist = [LOW, HIGH]
>>> qj_setpinvaluelist(list, vallist)
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2"]
>>> qj_setextestlist(list)
>>> qj_doextest()
>>> vallist = qj_getpinvaluelist(list)
>>> print (vallist)
[0, 1, 0]
```

qj_getpinvaluelist(*["pin", ...]*)	**Function: acquisition of pin value Argument type: text string Return value: int list**

Returns the value of the specified pin.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.

Example of use:

```
>>> qj_dosample()
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2"]
>>> vlist = qj_getpinvaluelist(list)
>>> print (vlist)
[0, 1, 0]
```

qj_setgroupvalue(*"group", value, mask*)	**Function: setting of group value Argument type: text string, int, int Return value: none**

Sets the specified value to specified group. This function expresses LOW, HIGH, and HIGH_Z of the pin corresponding to each bit position by combining the value value and the mask value. This function is mainly used for setting the output value when driving by the qj_doextest() function. Among group constitution pins, the value set to the input only pin is ignored. A bit whose mask value is set to "0" means output prohibition (HIGH_Z). In particular, note that if HIGH_Z is set to the input/output pin, the input value will be returned to later function calls such as qj_getpinvalue().
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified group does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2"]
>>> qj_newgroup("GROUP0", list, 0)
>>> qj_setgroupvalue("GROUP0", 5, 7)
>>> vallist = qj_getpinvaluelist(list)
>>> print (vallist)
[1, 0, 1]
```

qj_getgroupvalue(*"group"*)	**Function: acquisition of group value Argument type: text string Return value: int, int**

Returns the specified group value and the mask value.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified group does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2"]
>>> qj_newgroup("GROUP0", list, 0)
>>> qj_dosample()
>>> val, mask = qj_getgroupvalue("GROUP0")
>>> print (val, mask)
5 7
```

qj_setvalue(*["pin" or "group", ...], [value, ...], [mask, ...]*)	**Function: setting of pin/group value Argument type: text string list, int list, int list Return value: none**

Sets the specified value to specified pin/group. If a pin is specified, the mask value is ignored and the value value is set. If a group is specified, LOW, HIGH, and HIGH_Z of the pin corresponding to each bit position are set by combining the value value and the mask value. This function is mainly used for setting the output value when driving by the qj_doextest() function. The value specified to the input only pin is ignored. HIGH_Z(2) means output prohibition. In particular, note that if HIGH_Z is set to the input/output pin, the input value will be returned to later function calls such as qj_getpinvalue().

In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified group or pin does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2"]
>>> qj_newgroup("GROUP0", list, 0)
>>> list = ["GROUP0", "dev0.pin3", "dev0.pin4"]
>>> vallist = [5, LOW, HIGH]
>>> masklist = [7, 1, 1]
>>> qj_setvalue(list, vallist, masklist)
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2", "dev0.pin3", "dev0.pin4"]
>>> vallist = qj_getpinvaluelist(list)
>>> print (vallist)
[1, 0, 1, 0, 1]
```

qj_getvalue(*["pin" or "group", ...]*)	**Function: acquisition of pin/group value Argument type: text string list** **Return value: (int, int) list**

Acquires the specified pin/group values and the mask values in the list format. The list is structured as: [(value 0, mask 0), (value 1, mask 1), ...] If a pin is specified, the mask value is always "1." If a group is specified, the value is expressed by combining the value value and the mask value.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified group or pin does not exist on the board.

Example of use:

```
>>> qj_dosample()
>>> list = ["GROUP0", "dev0.pin3", "dev0.pin4"]
>>> vlist = qj_getvalue(list)
>>> print (vlist)
[(5, 7), (1, 1), (0, 1)]
```

5.1.3. Functions related to connection test setting

qj_setconnectlist(*["output", ...], ["input", ...]*)

Function: setting of pin for connection test Argument type: text string list, text string list Return value: none

Sets pin names or group names specified with arguments in the list of pin names or group names that is used in the subsequent qj_doconnecttest(). To the first argument, pass the names of the pins or groups on the output side as a list. To the second argument, pass the names of the pins or groups on the input side as a list. Every time this function is called, the existing setting information is discarded.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Specified pin or group does not exist on the board.
- The same pin is on both output side and input side.

Example of use:

```
>>> outlist = ["dev0.out0"]
>>> inlist = ["dev0.in0"]
>>> qj_setconnectlist(outlist, inlist)
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_addconnectlist(outlist, inlist)
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
['dev0.out0', 'dev0.out1', 'dev0.out2', 'OUTGROUP0'] ['dev0.in0', 'dev0.in1', 'dev0.in2']
>>> vallist = [LOW, LOW, LOW, 0x0]
>>> masklist = [1, 1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW, LOW]
>>> masklist = [1, 1, 1]
>>> qj_setexpectedvalue(inlist, vallist, masklist)
>>> result, errlist = qj_doconnecttest()
```

qj_addconnectlist(*["output", ...], ["input", ...]*)

Function: addition of pin for connection test Argument type: text string list, text string list Return value: none

Adds pin names or group names specified with arguments to the list of pin names or group names that is used in the subsequent qj_doconnecttest(). To the first argument, pass the names of the pins or groups on the output side as a list. To the second argument, pass the names of the pins or groups on the input side as a list.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Specified pin or group does not exist on the board.
- The same pin exists on both output side and input side in the added list and the existing list.

Example of use:

```
>>> outlist = ["dev0.out0"]
>>> inlist = ["dev0.in0"]
>>> qj_setconnectlist(outlist, inlist)
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_addconnectlist(outlist, inlist)
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
['dev0.out0', 'dev0.out1', 'dev0.out2', 'OUTGROUP0'] ['dev0.in0', 'dev0.in1', 'dev0.in2']
>>> vallist = [LOW, LOW, LOW, 0x0]
>>> masklist = [1, 1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW, LOW]
```

```
>>> masklist = [1, 1, 1]
>>> qj_setexpectedvalue(inlist, vallist, masklist)
>>> result, errlist = qj_doconnecttest()
```

qj_getconnectlist()

Function: acquisition of pin for connection test Argument type: none Return value: text string list, text string list

Returns the list of pin names or group names that is used in the subsequent qj_doconnecttest(). To the first return value, the list on the output side is returned. To the second return value, the list on the input side is returned.

Example of use:

```
>>> outlist = ["dev0.out0"]
>>> inlist = ["dev0.in0"]
>>> qj_setconnectlist(outlist, inlist)
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_addconnectlist(outlist, inlist)
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
['dev0.out0', 'dev0.out1', 'dev0.out2', 'OUTGROUP0'] ['dev0.in0', 'dev0.in1', 'dev0.in2']
>>> vallist = [LOW, LOW, LOW, 0x0]
>>> masklist = [1, 1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW, LOW]
>>> masklist = [1, 1, 1]
>>> qj_setexpectedvalue(inlist, vallist, masklist)
>>> result, errlist = qj_doconnecttest()
```

qj_removeconnectlist(*["output", ...], ["input", ...]*)

Function: exclusion of pin for connection test Argument type: text string list, text string list Return value: none

Excludes the pin names or group names specified with arguments from the list of pin names or group names that is used in the subsequent qj_doconnecttest(). To the first argument, pass the names of the pins or groups on the output side as a list. To the second argument, pass the names of the pins or groups on the input side as a list.

Example of use:

```
>>> outlist = ["dev0.out0", "dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in0", "dev0.in1", "dev0.in2"]
>>> qj_setconnectlist(outlist, inlist)
>>> outlist = ["dev0.out0"]
>>> inlist = []
>>> qj_removeconnectlist(outlist, inlist)
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
['dev0.out1', 'dev0.out2', 'OUTGROUP0'] ['dev0.in0', 'dev0.in1', 'dev0.in2']
>>> vallist = [LOW, LOW, LOW, 0x0]
>>> masklist = [1, 1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW, LOW]
>>> masklist = [1, 1, 1]
>>> qj_setexpectedvalue(inlist, vallist, masklist)
>>> result, errlist = qj_doconnecttest()
```

qj_clearconnectlist()	Function: clearing of pin for connection test Argument type: none Return value: none

Clears the list of pin names or group names that is used in the subsequent qj_doconnecttest().

Example of use:

```
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_setconnectlist(outlist, inlist)
>>> vallist = [LOW, LOW, 0x0]
>>> masklist = [1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW]
>>> masklist = [1, 1]
>>> qj_setexpectedvalue(inlist, vallist, masklist)
>>> result, errlist = qj_doconnecttest()
>>> qj_clearconnectlist()
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
[] []
```

qj_setexpectedpinvalue(*["pin", ...], [value, ...]*)	Function: setting of expected value of pin Argument type: text string list, int list Return value: none

Sets an expected value that is used for judging the result by qj_doconnecttest() to the specified pin. The pin name specified in this function needs to be specified in the input side of qj_setconnectlist().
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.
- The specified pin does not exist in the input side list of connection test.

Example of use:

```
>>> outlist = ["dev0.out0"]
>>> inlist = ["dev0.in0"]
>>> qj_setconnectlist(outlist, inlist)
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_addconnectlist(outlist, inlist)
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
['dev0.out0', 'dev0.out1', 'dev0.out2', 'OUTGROUP0'] ['dev0.in0', 'dev0.in1', 'dev0.in2']
>>> vallist = [LOW, LOW, LOW, 0x0]
>>> masklist = [1, 1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW, LOW]
>>> qj_setexpectedpinvalue(inlist, vallist)
>>> result, errlist = qj_doconnecttest()
```

qj_setexpectedvalue(*["pin" or "group", ...], [value, ...], [mask, ...]*)	Function: setting of expected value of pin/group Argument type: text string list, int list, int list Return value: none

Sets an expected value that is used for judging the result by qj_doconnecttest() to the specified pin or group. If a pin is specified, the mask value is ignored and the value value is set. If a group is specified, LOW, HIGH, and HIGH_Z of the pin corresponding to each bit position are set by combining the value value and the mask value. The pin name or group name specified in this function needs to be specified in the input side of qj_setconnectlist().
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified group or pin does not exist on the board.
- The specified group or pin does not exist in the input side list of connection test.

Example of use:

```
>>> outlist = ["dev0.out0"]
>>> inlist = ["dev0.in0"]
>>> qj_setconnectlist(outlist, inlist)
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_addconnectlist(outlist, inlist)
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
['dev0.out0', 'dev0.out1', 'dev0.out2', 'OUTGROUP0'] ['dev0.in0', 'dev0.in1', 'dev0.in2']
>>> vallist = [LOW, LOW, LOW, 0x0]
>>> masklist = [1, 1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW, LOW]
>>> masklist = [1, 1, 1]
>>> qj_setexpectedvalue(inlist, vallist, masklist)
>>> result, errlist = qj_doconnecttest()
```

5.1.4. Functions related to EXTEST setting

qj_setextestlist(*["pin or group", ...]*)	**Function: setting of pin for EXTEST Argument type: text string list** **Return value: none**

Sets pin names or group names specified with arguments to the list of pin names or group names that is used in the subsequent qj_doextest(). Every time this function is called, the existing setting information is discarded.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Specified pin or group does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin1", "dev0.pin2", "GROUP0"]
>>> qj_setextestlist(list)
>>> list = qj_getextestlist()
>>> print (list)
['dev0.pin1', 'dev0.pin2', 'GROUP0']
>>> qj_doextest()
```

qj_addextestlist(*["pin or group", ...]*)	**Function: addition of pin for EXTEST Argument type: text string list** **Return value: none**

Adds pin names or group names specified with arguments to the list of pin names or group names that is used in the subsequent qj_doextest().
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Specified pin or group does not exist on the board.

Example of use:

```
>>> list = ["dev0.pin0"]
>>> qj_setextestlist(list)
>>> list = ["dev0.pin1", "dev0.pin2", "GROUP0"]
>>> qj_addextestlist(list)
>>> list = qj_getextestlist()
>>> print (list)
['dev0.pin0', 'dev0.pin1', 'dev0.pin2', 'GROUP0']
>>> qj_doextest()
```

qj_getextestlist()	**Function: acquisition of pin for EXTEST Argument type: none** **Return value: text string list**

Returns the list of pin names or group names that is used in the subsequent qj_doextest().

Example of use:

```
>>> list = ["dev0.pin1", "dev0.pin2", "GROUP0"]
>>> qj_setextestlist(list)
>>> list = qj_getextestlist()
>>> print (list)
['dev0.pin1', 'dev0.pin2', 'GROUP0']
>>> qj_doextest()
```

qj_removeextestlist(*["pin or group", ...]*)	**Function: exclusion of pin for EXTEST Argument type: text string list Return value: none**

Excludes pin names or group names specified with arguments from the list of pin names or group names that is used in the subsequent qj_doextest().

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "dev0.pin2", "GROUP0"]
>>> qj_setextestlist(list)
>>> list = ["dev0.pin0"]
>>> qj_removeextestlist(list)
>>> list = qj_getextestlist()
>>> print (list)
['dev0.pin1', 'dev0.pin2', 'GROUP0']
>>> qj_doextest()
```

qj_clearextestlist()	**Function: clearing of pin for EXTEST Argument type: none Return value: none**

Clears the list of pin names or group names that is used in the subsequent qj_doextest().

Example of use:

```
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_setextestlist(list)
>>> qj_doextest()
>>> qj_clearextestlist()
>>> list = qj_getextestlist()
>>> print (list)
[]
```

5.1.5. Functions related to execution of boundary scan test

qj_doconnecttest()	Function: execution of connection test Argument type: none Return value: int, text string list

By using the pin and group that have been specified in advance in qj_setconnectlist(), executes the boundary scan test EXTEST command with result comparison. After executing the test, if the value of each specified pin on the input side is equal to the expected value set in qj_setexpectedvalue(), 1 (true) is returned to the first return value and a list containing a single blank text string only is returned to the second return value. After executing the test, if there is any pin whose value is different from the expected value, 0 (false) is returned to the first return value and the list of the pin names whose values are different from the expected value is returned to the second return value.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- Pins (input side) are not registered with qj_setconnectlist() or qj_addconnecttlist().
- There are pins or groups on the input side whose expected values are not set.
- The Waveform Display window is performing measurement.

Example of use:

```
>>> outlist = ["dev0.out0", "dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in0", "dev0.in1", "dev0.in2"]
>>> qj_setconnectlist(outlist, inlist)
>>> outlist, inlist = qj_getconnectlist()
>>> print (outlist, inlist)
['dev0.out0', 'dev0.out1', 'dev0.out2', 'OUTGROUP0'] ['dev0.in0', 'dev0.in1', 'dev0.in2']
>>> vallist = [LOW, LOW, LOW, 0x0]
>>> masklist = [1, 1, 1, 0xF]
>>> qj_setvalue(outlist, vallist, masklist)
>>> vallist = [LOW, LOW, LOW]
>>> masklist = [1, 1, 1]
>>> qj_setexpectedvalue(inlist, vallist, masklist)
>>> result, errlist = qj_doconnecttest()
>>> print (result, errlist)
0 ['dev0.in0']
```

qj_doextest()	Function: execution of boundary scan EXTEST command Argument type: none Return value: none

Operates the pins specified in advance in qj_setextestlist() or qj_addextestlist() by the boundary scan EXTEST command. If the pins are output pins or input/output pins to which HIGH or LOW is set, they are driven. If the pins are input pins, or input/output pins to which HI_Z is set, their status is acquired.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- Pins are not registered by qj_setextestlist() or qj_addextestlist().
- The Waveform Display window is performing measurement.

Example of use:

```
>>> list = ["dev0.pin1", "dev0.pin2", "GROUP0"]
>>> qj_setextestlist(list)
>>> list = qj_getextestlist()
>>> print (list)
['dev0.pin1', 'dev0.pin2', 'GROUP0']
>>> qj_doextest()
```

qj_dosample()	Function: execution of boundary scan SAMPLE command Argument type: none Return value: none

Updates values of all pins on the board by the boundary scan SAMPLE command.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.

Example of use:

```
>>> qj_dosample()
>>> a = qj_getpinvalue("dev0.pin0")
>>> print (a)
0
```

5.1.6. Functions related to operation of the Waveform Display window

qj_wv_open(*"name"*)	**Function: opening of waveform display pattern Argument type: text string Return value: none**

Opens the waveform display pattern whose name is specified by an argument. If the specified name does not exist in Project view, a project with that name is newly created. If the Waveform Display window is closed, this function opens it.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.

Example of use:

```
>>> qj_wv_open("wave 1")
```

qj_wv_close()	**Function: closing of Waveform Display window Argument type: none Return value: none**

Closes the Waveform Display window. When the window is already closed, this function has no effect.

Example of use:

```
>>> qj_wv_open("wave 1")
>>> qj_wv_close()
```

qj_wv_add(*["pin" or "group", ...]*)	**Function: addition of signal to Waveform Display window Argument type: text string list Return value: none**

Adds the specified pin or group to the Waveform Display window (pattern) that is currently open.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Specified pin or group does not exist on the board.
- The Waveform Display window is closed.
- The Waveform Display window is performing measurement.

Example of use:

```
>>> qj_wv_open("wave 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_wv_add(list)
```

qj_wv_remove(*["pin" or "group", ...]*)	**Function: exclusion of signal from Waveform Display window Argument type: text string list Return value: none**

Excludes the specified pin or group from the Waveform Display window (pattern) that is currently open.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Waveform Display window is closed.
- The Waveform Display window is performing measurement.

Example of use:

```
>>> qj_wv_open("wave 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_wv_add(list)
>>> list = ["dev0.pin0"]
>>> qj_wv_remove(list)
```

qj_wv_setinterval(*interval*)	**Function: setting of sampling interval Argument type: int Return value: int**

Sets the sampling interval of the Waveform Display window (pattern) that is currently open. The unit of sampling interval is millisecond (ms). When a value not listed in the combo box for sampling interval of the Waveform Display window is specified, this function sets the nearest value above the specified value. It is returned as an actually set value.

In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Waveform Display window is closed.
- The Waveform Display window is performing measurement.

Example of use:

```
>>> qj_wv_open("wave 1")
>>> a = qj_wv_getinterval()
>>> print (a)
100
>>> a = qj_wv_setinterval(250)
>>> print (a)
300
```

qj_wv_getinterval()	**Function: acquisition of sampling interval Argument type: none Return value: int**

Acquires the sampling interval of the Waveform Display window (pattern) that is currently open. The unit of sampling interval is millisecond (ms).
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Waveform Display window is closed.

Example of use:

```
>>> qj_wv_open("wave 1")
>>> a = qj_wv_getinterval()
>>> print (a)
100
```

qj_wv_start()	**Function: start of sampling Argument type: none Return value: none**

Starts measurement of the Waveform Display window (pattern) that is currently open. If measurement has been already started, no action is made.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Waveform Display window is closed.
- No signal is registered.

Example of use:

```
>>> qj_wv_open("wave 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_wv_add(list)
>>> qj_wv_start()
```

qj_wv_stop()	**Function: stopping of sampling Argument type: none Return value: none**

Stops measurement of the Waveform Display window (pattern) that is currently open. If the measurement has been already stopped, no action is made.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Waveform Display window is closed.

Example of use:

```
>>> qj_wv_open("wave 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_wv_add(list)
>>> qj_wv_start()
>>> input("Wait Input:   ")
Wait Input:
>>> qj_wv_stop()
```

5.1.7. Functions related to operation in the Pin Status List window

qj_pv_open(*"name"*)	**Function: opening of Pin Status List Argument type: text string Return value: none**

Opens the pin status list whose name is specified with an argument. If the specified name does not exist in Project view, a project with that name is newly created. If the Pin Status List window is closed, this function opens it.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.

Example of use:

```
>>> qj_pv_open("list 1")
```

qj_pv_close()	**Function: closing of Pin Status List window Argument type: none Return value: none**

Closes the Pin Status List window. When the window is already closed, this function has no effect.

Example of use:

```
>>> qj_pv_open("list 1")
>>> qj_pv_close()
```

qj_pv_add(*["pin" or "group", ...]*)	**Function: addition of signal to Pin Status List window Argument type: text string list Return value: none**

Adds the specified pin or group to the Pin Status List (list) that is currently open.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Specified pin or group does not exist on the board.
- The Pin Status List window is closed.

Example of use:

```
>>> qj_pv_open("list 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_pv_add(list)
```

qj_pv_remove(*["pin" or "group", ...]*)	**Function: exclusion of pin from Pin Status List window Argument type: text string list Return value: none**

Excludes the specified pin or group from the Pin Status List window (list) that is currently open.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Pin Status List window is closed.

Example of use:

```
>>> qj_pv_open("list 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_pv_add(list)
>>> list = ["dev0.pin0"]
>>> qj_pv_remove(list)
```

qj_pv_setpinvalue(*"pin", value*)	**Function: setting of pin value Argument type: text string, int Return value: none**

Sets the specified value to pin value. This function is mainly used for setting the output value when driving by the qj_pv_doextest() function. When an input only pin is specified, this function has no effect. When HIGH_Z(2) is set, output is prohibited. After execution, updates the display of the Pin Status List window.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Pin Status List window is closed.
- The specified pin does not exist on the board.

Example of use:

```
>>> qj_pv_open("list 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_pv_add(list)
>>> qj_pv_setpinvalue("dev0.pin0", LOW)
```

qj_pv_setpinvaluelist(*["pin", ...], [value, ...]*)

Function: setting of pin value Argument type: text string list, int list Return value: none

Sets each specified values to one or more specified pins. This function is mainly used for setting the output value when driving by the qj_pv_doextest() function. When an input only pin is specified, this function has no effect. When HIGH_Z(2) is set, output is prohibited. After execution, updates the display of the Pin Status List window.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Pin Status List window is closed.
- The specified pin does not exist on the board.
- The number of elements differs between the pin list and the value list.

Example of use:

```
>>> qj_pv_open("list 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_pv_add(list)
>>> list = ["dev0.pin0", "dev0.pin1"]
>>> vallist = [LOW, HIGH]
>>> qj_pv_setpinvaluelist(list, vallist)
```

qj_pv_dosample()

Function: execution of boundary scan SAMPLE command Argument type: none Return value: none

Executes the SAMPLE command in the Pin Status List window.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Pin Status List window is closed.
- The pin is not registered in the Pin Status List.

Example of use:

```
>>> qj_pv_open("list 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_pv_add(list)
>>> qj_pv_dosample()
```

qj_pv_doextest()

Function: execution of boundary scan EXTEST command Argument type: none Return value: none

Executes the EXTEST command in the Pin Status List window.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Pin Status List window is closed.
- The pin is not registered in the Pin Status List.
- The Waveform Display window is performing measurement.

Example of use:

```
>>> qj_pv_open("list 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_pv_add(list)
>>> qj_pv_setpinvalue("dev0.pin0", LOW)
>>> qj_pv_doextest()
```

qj_pv_savecsv()	Function: saving of Pin Status List window Argument type: none Return value: none

Saves the status of the Pin Status List in a CSV file. This is equivalent to pressing the "Save in CSV" button in the Pin Status List window.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The Pin Status List window is closed.

Example of use:

```
>>> qj_pv_open("list 1")
>>> list = ["dev0.pin0", "dev0.pin1", "GROUP0"]
>>> qj_pv_add(list)
>>> qj_pv_dosample()
>>> qj_pv_savecsv()
```

5.1.8. Functions related to the Connection Test window

qj_cv_open(*"name"*)	**Function: opening of Connection Test window Argument type: text string Return value: none**

Opens the connection test whose name is specified by an argument. If the specified name does not exist in Project view, a project with that name is newly created. If the Connection Test window is closed, this function opens it.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.

Example of use:

```
>>> qj_cv_open("test 1")
```

qj_cv_close()	**Function: closing of Connection Test window Argument type: none Return value: none**

Closes the Connection Test window. When the window is already closed, this function has no effect.

Example of use:

```
>>> qj_cv_open("test 1")
>>> qj_cv_close()
```

qj_cv_addconnectlist(*["output", ...], ["input", ...]*)	**Function: addition of net to Connection Test window Argument type: text string list, text string list Return value: none**

Adds net to the Connection Test window and sets the output pin list and input pin list that are specified by arguments. To the first argument, pass the names of the pins or groups on the output side as a list. To the second argument, pass the names of the pins or groups on the input side as a list.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- Specified pin or group does not exist on the board.
- In both of output pin list and input pin list, pins or groups having the same name are specified.
- An input only pin is included in the output pin list.
- An output only pin is included in the input pin list.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
```

qj_cv_removenet(*netnumber*)	**Function: deletion of net from Connection Test window Argument type: int Return value: none**

Deletes the net of the specified number from the Connection Test window. If a number that does not exist is specified, this function has no effect.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> qj_cv_removenet(1)
```

qj_cv_setoutput(*netnumber*, *[value, ...]*)

Function: setting of output value to net Argument type: int, int list Return value: none

Sets an output value to the net of the specified number in the Connection Test window. The existing output values are overwritten.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> values = [0, 1, 2, 3]
>>> qj_cv_setoutput(1, values)
```

qj_cv_addoutput(*netnumber*, *[value, ...]*)

Function: addition of output value to net Argument type: int, int list Return value: none

Adds an output value to the net of the specified number in the Connection Test window. The new output value is added to the line below the existing output values.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> values = [0, 1]
>>> qj_cv_setoutput(1, values)
>>> values = [2, 3]
>>> qj_cv_addoutput(1, values)
```

qj_cv_editoutput(*netnumber*, *valueindex*, *value*)

Function: change of output value in net Argument type: int, int, int Return value: none

Changes the output value of the specified index in the net of the specified number in the Connection Test window. The index of each net begins with 0.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.
- An index that does not exist is specified.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
```

```
>>> values = [0, 1, 2, 3]
>>> qj_cv_setoutput(1, values)
>>> qj_cv_editoutput(1, 1, 3)
```

qj_cv_removeoutput(*netnumber*, *valueindex*)

Function: deletion of output value from net Argument type: int, int Return value: none

Deletes the output value of the specified index in the net of the specified number in the Connection Test window. The index of each net begins with 0.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.
- An index that does not exist is specified.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> values = [0, 1, 2, 3]
>>> qj_cv_setoutput(1, values)
>>> qj_cv_removeoutput(1, 1)
```

qj_cv_setexpected(*netnumber*, *valueindex*, *value*, *mask*)

Function: setting of expected value to net Argument type: int, int, int, int Return value: none

Sets an expected value for the output value of the specified index in the net of the specified number in the Connection Test window. The index of each net begins with 0. The final value is determined by the combination of the value and the mask value. When the mask bit is 0, the corresponding bit ignores the result of value comparison in a connection test. (This is always handled as OK. Character "x" is displayed in the Connection Test window.) In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.
- An index that does not exist is specified.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> values = [0, 1, 2, 3]
>>> qj_cv_setoutput(1, values)
>>> qj_cv_setexpected(1, 0, 0b00, 0b11)
>>> qj_cv_setexpected(1, 1, 0b01, 0b01)
```

qj_cv_setexpectedlist(*netnumber*, *[value, ...]*, *[mask, ...]*)

Function: setting of expected value to net Argument type: int, int list, int list Return value: none

Sets an expected value for each output value of the net of the specified number in the Connection Test window. The final value is determined by the combination of the value and the mask value. When the mask bit is 0, the corresponding bit ignores the result of value comparison in a connection test. (This is always handled as OK. Character "x" is displayed in the Connection Test window.) If an expected value list containing more expected values than the number of output values is passed to the argument, the

excessive expected values are ignored.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.
- The number of expected values in the expected value list differs from the number of mask values in the mask value list.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> values = [0, 1, 2, 3]
>>> qj_cv_setoutput(1, values)
>>> values = [0, 1, 2, 3]
>>> masks = [0b11, 0b01, 0b11, 0b10]
>>> qj_cv_setexpectedlist(1, values, masks)
```

qj_cv_setvalue(*netnumber, [output, ...], [expect, ...], [mask, ...]*)	**Function: setting of output value and expected value to net Argument type: int, int list, int list, int list Return value: none**

Sets an output value and an expected value to the net of the specified number in the Connection Test window. The final expected value is determined by the combination of **expect** and **mask**. When the mask bit is 0, the corresponding bit ignores the result of value comparison in a connection test. (This is always handled as OK. Character "x" is displayed in the Connection Test window.) The existing settings of the output value and expected value are overwritten. If an expected value list containing more expected values than the number of output values is passed to the argument, the excessive expected values are ignored.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.
- The number of expected values in the expected value list differs from the number of mask values in the mask value list.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> values = [0, 1, 2, 3]
>>> expects = [0, 1, 2, 3]
>>> masks = [0b11, 0b01, 0b11, 0b10]
>>> qj_cv_setvalue(1, values, expects, masks)
```

qj_cv_autovalue(*netnumber or NULL*)	**Function: automatic creation of output pattern Argument type: int Return value: none**

Executes "Create Output Pattern Automatically" menu to the net of the specified number in the Connection Test window. When the net number is omitted, this menu is executed to all nets.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.

- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> qj_cv_autovalue()
```

qj_cv_learnexpected(*netnumber*, *valueindex*) or (*NULL*)	**Function: learning of expected value Argument type: int, int Return value: none**

Executes "Learn Expected Values" menu to the output value of the specified index in the net of the specified number in the Connection Test window. The index of each net begins with 0. When all arguments are omitted, "Learn Expected Values" menu is executed to all output values.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.
- An index that does not exist is specified.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> qj_cv_autovalue()
>>> qj_cv_learnexpected(1, 0)
```

qj_cv_dotest(*netnumber*, *valueindex*) or (*NULL*)	**Function: execution of connection test Argument type: int, int Return value: none**

Executes "Connection Test" menu to the output value of the specified index in the net of the specified number in the Connection Test window. The index of each net begins with 0. When all arguments are omitted, "Learn Expected Values" menu is executed to all output values.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.
- A net number that does not exist is specified.
- An index that does not exist is specified.
- At least one of the output value and the expected value is not set for the specified index.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> qj_cv_autovalue()
>>> qj_cv_learnexpected(1, 0)
>>> qj_cv_dotest()
```

qj_cv_opennetlist()	Function: opening of net list Argument type: none Return value: none

Executes "Open Net List" menu in the Connection Test window.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.

Example of use:

```
>>> qj_cv_open("test 1")
>>> qj_cv_opennetlist()
```

qj_cv_savenetlist()	Function: saving of net list Argument type: none Return value: none

Executes "Save Net List" menu in the Connection Test window.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> qj_cv_savenetlist()
```

qj_cv_saveresult()	Function: saving of connection test result Argument type: none Return value: none

Executes "Save Connection Test Results" menu in the Connection Test window.
In the following cases, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The board has not passed the consistency test.
- The Connection Test window is closed.
- No net is registered in the Connection Test window.

Example of use:

```
>>> qj_cv_open("test 1")
>>> outlist = ["dev0.out1", "dev0.out2", "OUTGROUP0"]
>>> inlist = ["dev0.in1", "dev0.in2"]
>>> qj_cv_addconnectlist(outlist, inlist)
>>> qj_cv_autovalue()
>>> qj_cv_learnexpected(1, 0)
>>> qj_cv_dotest()
>>> qj_cv_saveresult()
```

5.1.9. Functions related to environment setting

qj_getfrequencylist()	Function: acquisition of frequency list Argument type: none Return value: text string list

Returns a value that can be set as a boundary scan test frequency.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Not connected with probe.

Example of use:

```
>>> list = qj_getfrequencylist()
>>> print (list)
['Auto', '[2...624]kHz', '0.62MHz', '0.93MHz', '1.25MHz', '1.87MHz', '2.50MHz', '3.75MHz', '5.00MHz', '7.50MHz', '10.00MHz', '15.00MHz']
>>> a = qj_setfrequency("5.00MHz")
>>> print (a)
5000000
```

qj_setfrequency(*"frequency"*)	Function: setting of boundary scan frequency Argument type: text string Return value: int

Sets the frequency at which boundary scan test is conducted in a text string. Returns the int value of the actually set frequency as the return value. In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Not connected with probe.

Example of use:

```
>>> list = qj_getfrequencylist()
>>> print (list)
['Auto', '[2...624]kHz', '0.62MHz', '0.93MHz', '1.25MHz', '1.87MHz', '2.50MHz', '3.75MHz', '5.00MHz', '7.50MHz', '10.00MHz', '15.00MHz']
>>> a = qj_setfrequency("5.00MHz")
>>> print (a)
5000000
```

qj_getfrequency()	Function: Acquisition of boundary scan frequency Argument type: none Return value: int

Returns the frequency at which boundary scan test is conducted.
In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- Not connected with probe.

Example of use:

```
>>> list = qj_getfrequencylist()
>>> print (list)
['Auto', '[2...624]kHz', '0.62MHz', '0.93MHz', '1.25MHz', '1.87MHz', '2.50MHz', '3.75MHz', '5.00MHz', '7.50MHz', '10.00MHz', '15.00MHz']
>>> a = qj_getfrequency()
>>> print (a)
5000000
```

5.1.10. Others

qj_runthefile(*"filename" or NULL*)	**Function: execution of script file Argument type: text string Return value: none**

Executes the script file specified by an argument. This is equivalent to the "Execute button" function in the Script pane. When the argument is omitted, the file listed in the "File Name" edit box in the Script pane is executed. If a file name has been passed to the argument, the corresponding file is executed. Specify the file name with the full path. However, due to the specifications of the escape sequence of Python, double the **"¥"** character for each directory. If a file name that does not exist is specified, a message box saying that the specified file does not exist is displayed in the same manner as pressing "Execute button" in the Script pane.

Example of use:

```
>>> qj_runthefile("C:¥¥Query-J¥¥samplescript.py")
>>> qj_runthefile()
```

qj_setalias(*"pin", "newname"*)	**Function: setting of pin name alias Argument type: text string, text string Return value: none**

Assigns the newname to the pin specified by the argument "pin." Use this function if there is a pin for which you want to omit description of the device name in functions such as qj_getvalue(). If you specify a registered newname, the existing newname is overwritten.

In the following case, error information is output to the standard output and, if a script file is being executed, the execution is interrupted.

- The specified pin does not exist on the board.

Example of use:

```
>>> qj_setalias("dev0.pin0", "P0")
>>> a = qj_getpinvalue("P0")
>>> print (a)
0
```

5.2. Precautions for Use of Interface Functions

When you use the interface functions for Python script that are provided by this software, pay attention to the following issues.

5.2.1. Calling of functions

All the interface functions are defined in the queryj module and they are imported to the Python interpreter ("import queryj") when this software starts up.
At the same time, all the interface functions are registered as global functions ("from queryj import *").
Therefore, all the interface functions can be called with queryj.qj_xxxx() or qj_xxxx().
While this software is starting up, do not delete the queryj module. If it is deleted, the operation of further script execution functions is not guaranteed.

5.2.2. Specification of a pin name

To specify a pin name in the Python script, describe as follows:
"Device name.pin name (or port name)"
(Example: acquiring the value of "first pin" of "dev0 device" to int-type variable "value": value = qj_getpinvalue("dev0.1"))
(Example: setting the value of "first pin" of "dev0 device" to text string list-type variable "list": list = ["dev0.1"]

However, only for qj_setpinname(), specify device name and pin name separately as an exception.
(Example: giving name "data0" to "first pin" of "dev0 device": qj_setpinname("dev0", "1", "data0"))
After changing a name with qj_setpinname() or "Change Port Name," you can specify the pin with the name after change or the pin name, but you cannot specify the pin with the name before change. In other words, you can specify the pin only with the pin name or the port name that is currently displayed in Project view or other screens.

5.2.3. Pin value (data format)

・Use of pin only
0 >> LOW
1 >> HIGH
2 >> HI_Z (high impedance)
This software also offers variables LOW, HIGH, and HI_Z of the above values.
(Example: setting "LOW" to "first pin" of "dev0 device": qj_setpinvalue("dev0.1", LOW))
・Use in group
Use in combination with bit mask.
0 (mask 1) >> Low
1 (mask 1) >> High
0 (mask 0) >> High_Z
1 (mask 0) >> High_Z
For example, if value=0xAA and mask value=0xF0, bits are set as
High, Low, High, Low, High_Z, High_Z, High_Z, and High_Z from left to right.
(Example : list = ["GROUP0"]
vlist = [0xaa]
mlist = [0xf0]
qj_setvalue(list, vlist, mlist))

5.2.4. Pin value (setting/acquisition by type)

Note that handling of values in functions such as qj_setpinvalue(), qj_getpinvalue(), qj_setgroupvalue(), qj_getgroupvalue(), qj_setvalue(), and qj_getvalue() that handle value setting and acquisition is affected by the type of the corresponding pin.

Table 39

Input only pin	・ set function does nothing. Input side value is returned to get function.
Output only pin	・ set function sets an output value. Output side value is returned to get function.
Input/output pin	・ set function sets an output value. ・ When set function sets HIGH or LOW, output side value is returned to get function. ・ When set function sets HI_Z, input side value is returned to get function.

Therefore, when set function and get function are executed to the same pin, the set value and the get value may differ.
Example: LOW is set to input only pin, but HIGH is acquired.
Example: HI_Z is set to input/output pin, but LOW is acquired.
etc.
From the above, the relationships between the setting and acquisition of the value and mask value in group functions (qj_setgroupvalue(), qj_getgroupvalue()) are summarized as follows:

Table 40

Pin type	set (value, mask)	get (value, mask)
Input only pin	LOW(0, 1) * However, no operation.	LOW(0, 1) or HIGH(1, 1) * Status after the last SAMPLE command executed
	HIGH(1, 1) * However, no operation.	LOW(0, 1) or HIGH(1, 1) * Status after the last SAMPLE command executed
	HI_Z(0, 0) * However, no operation.	LOW(0, 1) or HIGH(1, 1) * Status after the last SAMPLE command executed
Output only pin	LOW(0, 1)	LOW(0, 1)
	HIGH(1, 1)	HIGH(1, 1)
	HI_Z(0, 0)	HI_Z(0, 0)
Input/output pin	LOW(0, 1)	LOW(0, 1)
	HIGH(1, 1)	HIGH(1, 1)
	HI_Z(0, 0)	LOW(0, 1) or HIGH(1, 1) * Status after the last SAMPLE command executed

5.2.5. Pin value (drive/update timing)

The signal on the target board is driven or taken in this software when a function related to boundary scan test execution such as qj_doextest(), qj_doconnecttest(), and qj_dosample() are executed.
For example, the signal on the target board is not driven when a value is set with a function related to value setting (e.g. qj_setpinvalue()).
(Even when HIGH is set on this software, the state on the target board may be LOW.)

Table 41

Pin type	set (value, mask)
qj_doextest()	When this function is executed, only the pin acquired by qj_getextestlist() is operated. Depending on the pin type, this function does the following.
	Input only pin: Takes in the status.
	Output only pin: Drives the value set with qj_setpinvalue() or the like.
	Input/output pin: Drives if HIGH or LOW has been set with qj_setpinvalue() or the like (output action). Takes in the status if HI_Z has been set (input action).
qj_doconnecttest()	When this function is executed, operates only the pin acquired with qj_getconnectlist(). Drives the pin registered on the output side to the value set in a function such as qj_setpinvalue() and takes in the status of the pin registered on the input side.
qj_dosample()	Takes in the status of all pins. Output signal is not driven.

To drive an output only pin or an input/output pin when qj_doextest() or qj_doconnecttest() is executed, a function for value setting such as qj_setpinvalue needs to be applied to the pin in advance.
(By setting an output value, the status becomes equivalent to "checkmarked" in the Pin Condition window.)

5.2.6. Device name

The initial value of device name is "devN" ("N" represents the order of device in boundary scan chain starting with 0).
Therefore, qj_setdevicereference() prohibits to set the text string "devN" as a new reference name in upper-case letters or lower-case letters.

5.2.7. Clock setting

For the frequency that is specified in the function qj_setfrequency(), use of the following text strings is allowed, for example:

Table 42

"Auto"	Example: qj_setfrequency("auto") >> 8.33MHz
Value without unit	Example: qj_setfrequency("10000") >> low-speed clock 10kHz
Numeric value with k(Hz)	Example: qj_setfrequency("100k") >> low-speed clock 100kHz qj_setfrequency("100kHz") >> low-speed clock 100kHz
Numeric value with M(Hz)	Example: qj_setfrequency("10M") >> 8.33MHz qj_setfrequency("8.33MHz") >> 8.33MHz

Units other than kHz and MHz are prohibited because the probe cannot output clocks in such units.

A. Appendix

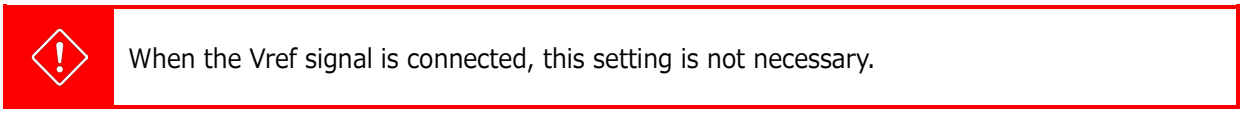
A.1. Connection and Handling Method of Devices

This section explains cases of special connection among this software, probe and target board.

A.1.1. Target not connected with Vref signal

When the target board is not equipped with signal that connects with the Vref signal from the JTAG connector of the probe, make the probe output 3.3V as the High level of the JTAG signal (example: SH7705).

* SH7705 is a trademark of Renesas Technology Corporation.



When the Vref signal is connected, this setting is not necessary.

1) Create a new project

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Create New Project**	Ctrl + N

Executing the above menu item creates an untitled project in the Project view.

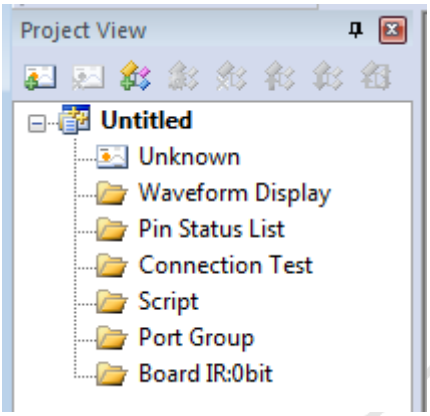


Figure 80

2) Forced output of JTAG signal

Tool button	Operation on the menu bar	Shortcut key
---	**Project -> Forced Output of JTAG Signal**	---

Click the above menu to checkmark.
A confirmation message is displayed when you checkmark. Select "OK."
Checkmarking the menu enables 3.3V to be output as the High level of JTAG signal.

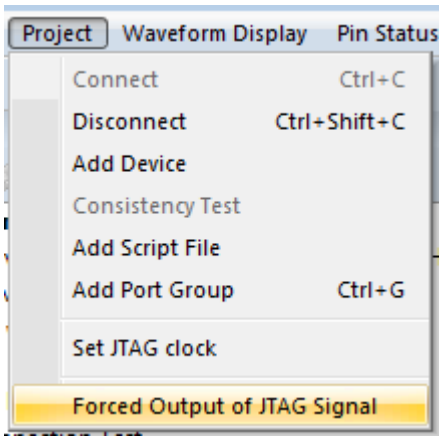


Figure 81

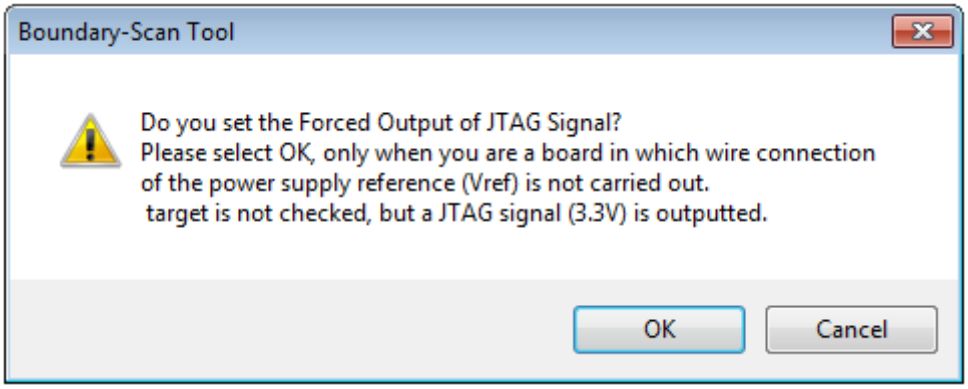


Figure 82

3) Connect

Tool button	Operation on the menu bar	Shortcut key
	**Project -> Connect**	Ctrl + C

Connect between this software and the probe. The software tries to automatically recognize the target boundary scan chain at connection time.

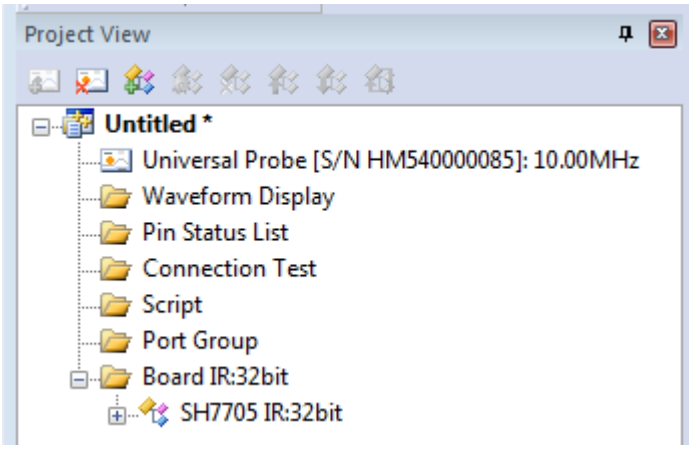
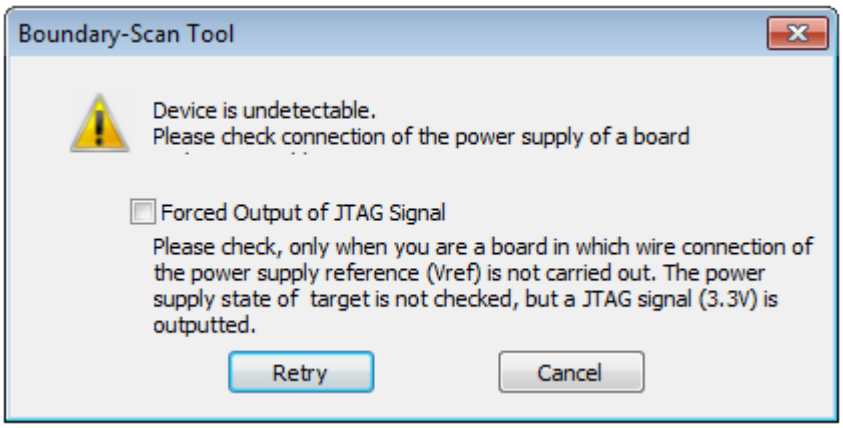


Figure 83

If you connect without checkmarking the "Forced Output of JTAG Signal" checkbox, the power supply to the target board cannot be detected and the following message is displayed.



Checkmarking the "Forced Output of JTAG Signal" checkbox and pressing the Retry button performs automatic recognition of device again.

A.1.2. When a device not contained in BSDL information is recognized

Although a single CPU as hardware, automatic recognition of boundary scan chain may detect multiple devices. In this case, boundary scan test may be conducted by following the procedures below (example: i.MX512).

*i.MX512 is a trademark of Freescale Corporation.

1) Create a new project

Tool button	Operation on the menu bar	Shortcut key
	**File** -> **Create New Project**	Ctrl + N

Executing the above menu item creates an untitled project in the Project view.

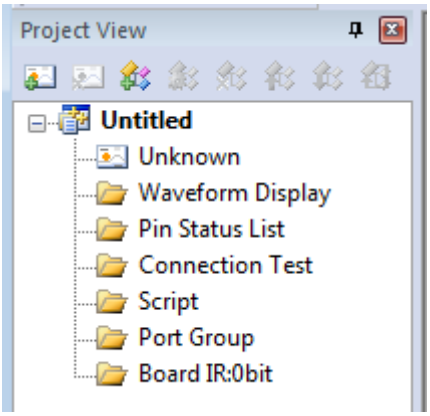


Figure 84

2) Connect

Tool button	Operation on the menu bar	Shortcut key
	**Project -> Connect**	Ctrl + C

Connect between this software and the probe. The software tries to automatically recognize the target boundary scan chain at connection time.

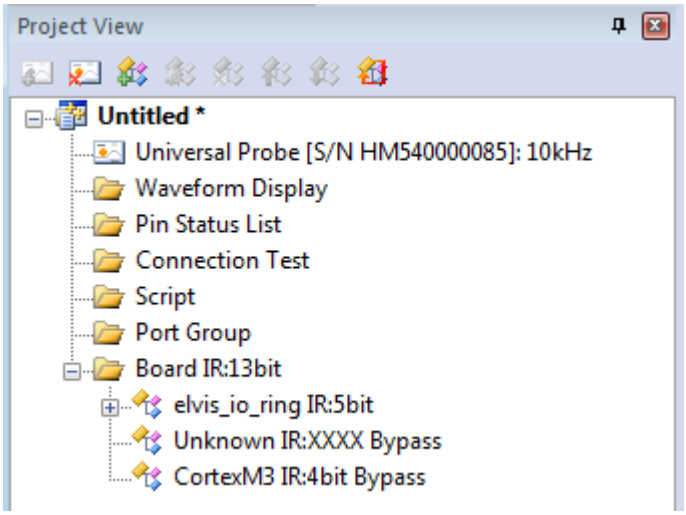


Figure 85

3) Properties of an "unknown" device

Tool button	Operation on the menu bar	Shortcut key
---	This operation cannot be performed from the menu bar. * Project View -> Board -> unknown device, right-click the context menu	---

This displays the properties of the selected "unknown" device.
In the "Board" column of Project view, the total IR bit length of the board is displayed. Edit the IR length in the Device Properties dialog box by comparing with the IR bit length of other devices so that the edited IR bit length matches the total IR bit length.

Also, checkmark the "Put into Bypass State" checkbox. This is to skip the test with the BYPASS command during boundary scan test because the "unknown" device has no port information.

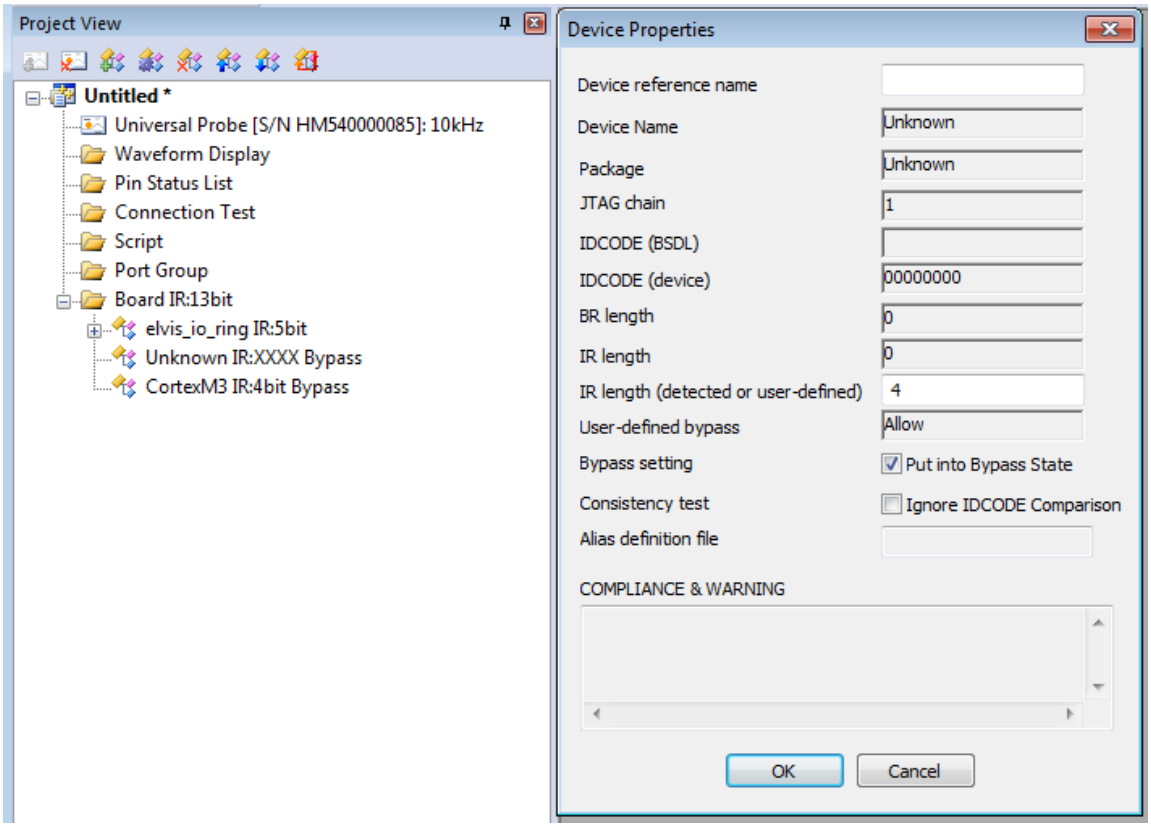


Figure 86

4) Set "Ignore IDCODE"

If the device's IDCODE that is actually acquired from the device differs from the IDCODE information described in BSDL, checkmark "Ignore IDCODE Comparison**"** checkbox in the Device Properties dialog box.

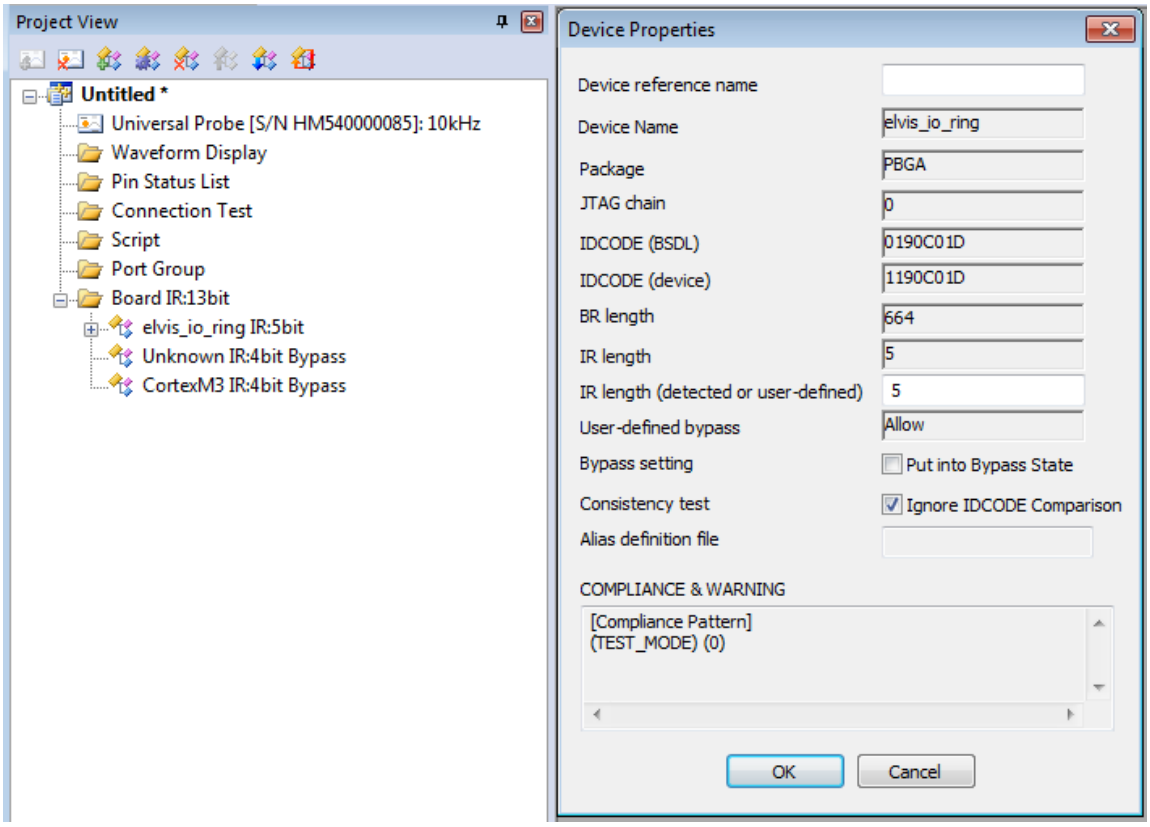


Figure 87

5) Consistency test

Tool button	Operation on the menu bar	Shortcut key
	**Project -> Consistency Test** * Project View -> Board -> Any device, right-click to open the context menu.	---

Test if the boundary scan test can be performed on the boundary scan chain configured in the Project view.
If the test is successful, the above menu is grayed out.

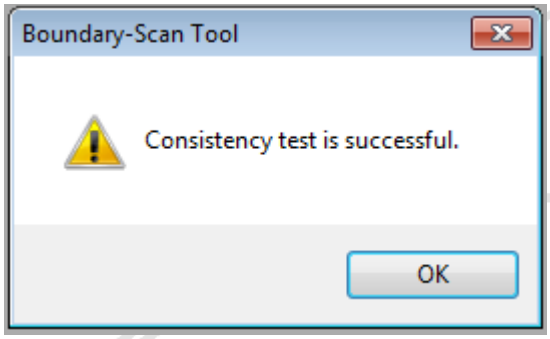


Figure 88

A.2. List of Messages

A.2.1. [ERROR]

This section provides error messages and their explanations.

Table 43

1	"Could not detect probe. Check USB port."	This message is displayed when this software could not detect the probe when "Connect" menu was executed. Probe may not be connected to PC, or failure occurred in USB cable or probe.
2	"Could not detect the probe with the specified serial number. Do you connect with other probe?"	This message is displayed when this software could not detect the probe that is registered in Project View when "Connect" menu was executed. Probe may not be connected to the PC, a different probe from the one registered in Project view may be connected to the PC, or a failure occurred in the USB cable or probe.
3	"Could not open project file."	This message is displayed when the contents of the project file could not be restored in the project during execution of "Open project" menu because of failure in accessing files or for other reasons.
4	"Failed to acquire port from BSDL file."	This message is displayed when a port group could not be created from the BSDL information of the specified BSDL file during execution of "Open project" and "Edit device" menus.
5	"Failed to analyze BSDL file."	This message is displayed when analysis of file failed after reading BSDL file in "Edit device dialog" and "Manage BSDL dialog."
6	"Failed to execute SAMPLE command."	This message is displayed when execution of "SAMPLE command" menu in the Pin Condition List window has failed.
7	"Failed to execute EXTEST command."	This message is displayed when execution of "EXTEST command" menu in the Pin Condition List window has failed.
8	"IR length is not set."	This message is displayed when the IR bit length is "0" during execution of "SAMPLE command" and "EXTEST command" menus in the Pin Condition List window.
9	"Specify Bypass setting to device whose BSDL has not been determined."	This message is displayed when a device set as "Unknown" in Project view is not set to bypass during execution of "SAMPLE command" and "EXTEST command" menus in the Pin Condition List window.
10	"Total IR length does not match the setting."	This message is displayed when the IR bit length set to each device in Project view and the IR bit length read from each device on the target board are different during execution of "SAMPLE command" and "EXTEST command" menus in the Pin Condition List window.
11	"Failed to acquire product ID. wpcpu.id in the installation folder may not exist or may be broken."	This message is displayed when the product ID could not be acquired from wpcpu.id file during execution of the "Connect" menu.

12	"Could not verify license. Check the combination of License No. and serial No. of probe."	This message is displayed when acquisition of license failed after License acquisition screen is displayed by executing the "Connect" and "Acquire probe license" menus.
13	"An error occurred in sampling. Sampling will be stopped."	This message is displayed when an error has occurred during measurement in the Waveform Display window.
14	The format of "file name" of waveform file is incorrect."	This message is displayed when a file with incorrect format is opened during execution of "Open with viewer" menu to open waveform.

A.2.2. [WARNING]

This section provides warning messages and their explanations.

Table 44

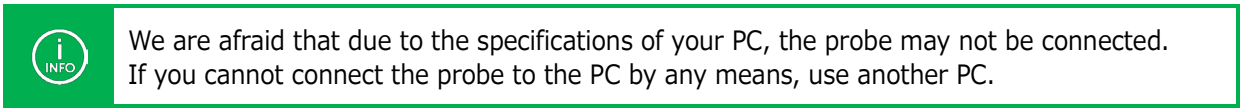
1	"Number of detected devices n Device 1 : IDCODE Device 2 : IDCODE ⋯"	This message is displayed when a device is automatically recognized during execution of the "Connect" menu.
2	"Probe will be disconnected. Do you clear the probe information in the Project view?"	This message is displayed when the "Disconnect" menu is executed asking you whether to keep the information of probe serial No. and JTAG frequency in the project.
3	"Sampling will be stopped. OK?"	When executing "Create new project," "Open project," "Close project," "Exit application," "Create new waveform display pattern," or "Save waveform" menu, this question is displayed if measurement is in progress in the Waveform Display window.
4	"A probe registered in project has been detected. Do you want to connect it?"	When "Open project" menu is executed, this question is displayed if a probe registered in the project file is connected to PC.
5	"The current project has been changed. Do you want to save it?"	When a project is displayed in Project view and the project name is appended with "*," this question is displayed when "Create New Project," "Open Project," "Close Project," or "Exit Application" is executed.
6	"Following devices have been registered. Manufacturer's name: Device 1 Device 2 . . ."	This message is displayed if the BSDL file read in the "Edit Device" dialog box and the "Manage BSDL" dialog box was successfully analyzed.
7	"Consistency test is successful."	Displays the execution result of "Consistency test" menu.
8	"Consistency test failed."	Displays the execution result of "Consistency test" menu.
9	"The selected list is being used in the Pin condition list window. Close the window on the left or delete the list after switching to another list."	This message is displayed if you attempt to delete a list that is opened in the Pin Condition List window from Project view.
10	"The selected pattern is being used in the Waveform Display window. Close the window on the left or delete the pattern after switching to another pattern."	This message is displayed if you attempt to delete a pattern that is opened in the Waveform Display window from Project View.
11	"Already passed consistency test. Do you	This question is displayed when "Add device,"

	want to continue?"	"Edit device," "Move device upward," or "Move device downward" menu is executed after passing consistency test (when boundary scan test is executable). If you continue the process, the target board may not pass consistency test and boundary scan test may not be able to be executed.
12	"Waveform will be cleared. OK?"	This question is displayed if a waveform data is displayed in the Waveform Display window when executing "Add signal," "Delete signal," "Start measurement," or "Change sampling interval" menu is executed.
13	"Running script is waiting for input. Cannot exit application."	This message is displayed when you attempt to execute an application exit menu while a command such as input() in Python script is waiting for input in the script pane. Complete execution of script and then exit the application.
14	"The specified group will be deleted because it contains a port of deleted device."	This message is displayed when a device is deleted from Project view if a port of the device is registered in the group.
15	"This script file has already been registered."	When a script file is selected in "Add script file" of the Project menu or in Project view, this message is displayed if the file has already been registered.
16	"Specified script file was not found."	This message is displayed if the file does not exist in the specified file path when the script file is executed.

A.3. Troubleshooting

A.3.1. Troubles related to installation

Question	Answer
Cannot detect probe.	Confirm the followings: ① This software does not detect any other probe than Universal Probe. ② Check again if probe is properly connected to PC. Also refer to **"Procedures for turning power on/off"**. ③ Installation of the software may have failed and the driver software may not be installed. Install the software again.



A.3.2. When Project view is used

Question	Answer
Automatic detection of device fails.	Confirm the followings: ① Check the connection of power supply to board and JTAG cable. If the board is not wired with the power supply reference signal (Vref) of JTAG cable, the board may be recognized by checkmarking "Project" >> "Forced Output of JTAG Signal" menu. ② Open "Manage BSDL" dialog from Tool menu and confirm that the desired device is registered. If it is not registered, acquire the corresponding BSDL file from the website of the device manufacturer or other sources and register the device from the above dialog box. For details, refer to **"Manage BSDL"** in "Tool." If the BSDL file is not disclosed, we are afraid that boundary scan test cannot be conducted on the device. ③ If the desired device is registered in step "②," confirm the IDCODE of the desired device in the "Manage BSDL" dialog. If the IDCODE column is blank, by registering the device by following **"Add Device"** and checkmarking the "Ignore IDCODE Comparison" checkbox on the Device Properties dialog box, boundary scan test may be executed. ④ If the device cannot be detected in "③," by changing "IR Bit Length" in the "Device Properties" dialog box, boundary scan test may be executed. • This method is effective only when the definition of IR bit length in the BSDL file differs from the IR bit length of the device. ⑤ If the trouble still persists, please contact us.

Revision History

Ver. No.	Revision date	Contents of revision
01	09/30/2014	Initial Release.
02	10/03/2014	Replaced figures.
03	11/26/2014	Correction of typographical errors.

Manufacturer Information

Sohwa & Sophia Technologies Inc.

[Headquarters]

6-2, Minami-kurokawa, Asao-ku, Kawasaki City, Kanagawa Pref.,
215-8588, JAPAN
HomePage: http://www.ss-technologies.co.jp/en/index.html

Subsidiary

SOHWA MALAYSIA SND. BHD.

Unit 5-2, Level 5, Tower 6, Avenue 5,
The Horizon, Bangsar South No.8, Jalan Kerinchi
59200, Kuala Lumpur, Malaysia

HomePage : http://www.sohwa-m.com.my/