

Agilent E2929A/B Windows and Dialog Boxes

**User Interface
Reference**



Agilent Technologies



Important Notice

All information in this document is valid for both Agilent E2929A and Agilent E2929B testcards.

Copyright

© 2001 Agilent Technologies. All rights reserved.

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Agilent Technologies Inc. as governed by United States and international copyright laws.

Author: Anja Schauer, t3 medien GmbH

Notice

The material contained in this document is subject to change without notice. Agilent Technologies makes no warranty of any kind with regard to this material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Agilent Technologies shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Contents

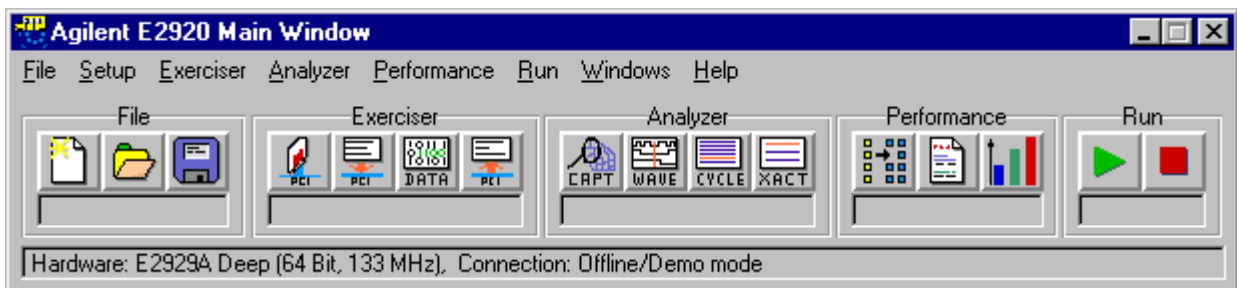
The Main Window	7
File Menu	9
Setup Menu	11
Testcard Configuration Dialog Box	12
PCI-X Selection List Dialog Box	14
Scan All Ports Dialog Box	14
Register Product Options Dialog Box	15
PCI-X Bus Dialog Box	15
Update Hardware Dialog Box	15
Exerciser Menu	17
Exerciser Generic Settings Dialog Box	19
Error Injection Page	19
Interrupt Injection Page	22
Trigger Injection Page	24
Initiator Page	25
Data Generator Page	26
Arbitration Page	28
Block and Behavior Editor Dialog Boxes	29
Requester-Initiator Block Editor Dialog Box	30
Requester-Initiator Behavior Editor Dialog Box	32
Completer-Target Behavior Editor Dialog Box	33
Completer-Initiator Behavior Editor Dialog Box	35
Requester-Target Behavior Editor Dialog Box	37
Configuration Space Window	39
Target Decode Window	41
Data Memory Editor Window	44

Select View Of Data Dialog Box	45
Initiator Conditional Start Dialog Box	46
Interrupt Generator Window	47
Analyzer Menu	49
<hr/>	
Protocol Check Window	50
Capture Dialog Box	51
Trigger Tab	51
Storage Tab	53
Pattern Editor Dialog Box	53
Selection List Dialog Box	54
Trigger Sequencer Dialog Box	55
Waveform Viewer Window	57
Arrange Signals Dialog Box	60
Bus Cycle Lister Window	61
Transaction Lister Window	62
Performance Menu	63
<hr/>	
Performance Setup Window	64
Completer Identification Tab	65
Requester Identification Tab	66
Pair Select Tab	66
Capture Tab	67
Performance Charts Window	68
PCI-X Usage Tab	69
Burst Usage Tab	70
Command Tab	71
Latencies Tab	71
Performance Report Window	72
Run Menu	73
<hr/>	
Run Options Window	73
Windows Menu	75
<hr/>	
Command Line Interface Window	75

PCI-X Real Time Performance	79
<hr/>	
Menus	80
Agilent RTP Real Time Performance Window	82
RTP Select Testcard Window	82
RTP Testcard Setup Window	83
Protocol Rule Masking Dialog Box	85
RTP Measurement Setup Dialog Box	86
RTP Options Dialog Box	87

The Main Window

The main window is either always visible or iconified while the application is running. It contains several menus accessible from the menubar and several groups of buttons. Which menus are enabled and which button groups are visible depends on which hardware and software components are licensed.



Menus The items of the different menus give access to all the features of this application and its components.

File menu As with many applications, you can select file handling actions from the File menu.

Setup menu All settings concerning the testcard type and connection are made in the Setup menu.

Exerciser menu If the PCI-X Exerciser is licensed with your testcard, you find all its settings and functions in the Exerciser menu.

Analyzer menu If the PCI-X Analyzer is licensed with your testcard, you find all its settings and functions in the Analyzer menu.

Performance menu If the PCI-X Performance Optimizer is licensed with your testcard, you find all its settings and functions in the Performance menu.

Windows menu If you need to open the Command Line Interface window or close all windows and dialog boxes, you can do so in the Windows menu.

Button Groups The main window contains a row of buttons that are grouped together. Some of them might not be visible unless the respective hardware or software component is licensed.

- The File button group far to the left is always visible.
- The component-specific groups are labeled *Exerciser*, *Analyzer* and *Performance*. They are only visible if you have the corresponding license.
- The Run group far to the right is always visible.

All groups contain various buttons. The function of each button is explained along with the menu items. Every group also contains a status text field, which provides information about the current status of the component.

Status Line In the status line at the bottom of the main window, you find information about the selected hardware and the current type of connection to the hardware, including the port number if applicable.



File Menu

The File menu provides access to all file handling actions. The possible options are:

New This selection starts a new session.

Load In order to load a previously saved session, select this item. Then choose the session file (bst file) from the file dialog box.

Save Clicking this item saves the current session.

Save As If you want to save the current test session with a new name and/or location, select this item.

Exit Selecting this item will close the application.

Setup Menu

The items of this menu are used to either setup, modify, or check the hardware and the connection, or simply to display the current test configuration.

Testcard Configuration Selecting this option opens the Testcard Configuration dialog box, where you can modify the current testcard configuration.

PCI-X Bus Selecting this option opens the PCI-X Bus dialog box where you can find the PCI-X bus properties.

Check Connection Selecting this option sends a test message to the testcard at the selected port. In reply, you get a message informing you if the testcard was found on this port. It also re-establishes the testcard.

Check Hardware This option compares the software (GUI) version to the current firmware version on the Agilent E2929A/B testcard. In case of a version mismatch a hardware update is necessary.

Update Hardware If the firmware version on the testcard is different from the GUI version, a hardware update is necessary. Select this item to open the Update Hardware dialog box.

Options The *Setup* menu provides the following Options::

- *Trigger Sequencer*

If this option is activated, you can set up the trigger sequence. You do this in the Trigger Sequencer dialog box, which you open by selecting *Capture...* in the *Analyzer* menu.

- *Auto Polling*

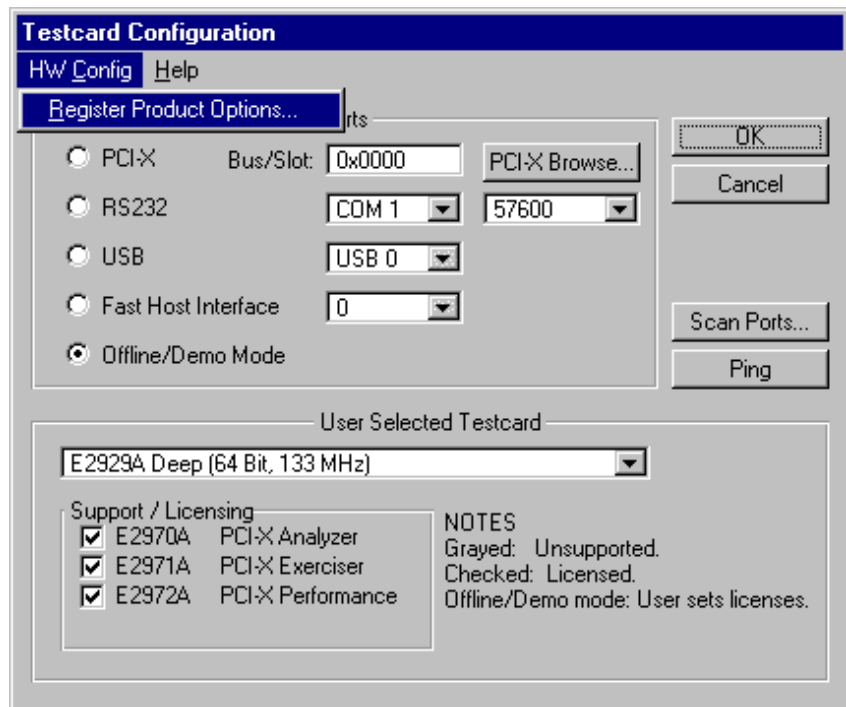
If this option is activated, the connection to the testcard is automatically checked.

- *Tooltips*

With this option you can activate or deactivate the tool tips. Tool tips are small yellow boxes briefly explaining the use of buttons and other window components when placing the cursor on them.

Testcard Configuration Dialog Box

The Testcard Configuration dialog box is used to set up the testcard configuration. This includes the selection of the connection type and the port as well as information specific to the Agilent E2929A/B testcard.



HW Config Menu If you want to register new software components for the testcard, open the Register Product Options dialog box from the HW Config menu.

Ports The type of connection between your control PC and the Agilent E2929A/B testcard can be selected with the four option buttons to the left. The possible choices are:

PCI-X The control software is running on the test system, and the testcard is accessed directly via the PCI-X bus. The Bus/Slot text field displays the port number. To view a list of all connected devices, click the *PCI-X Browse* button. This opens the PCI-X Selection List dialog box.

RS232 Mark this option if you have connected the testcard to the serial port of the control PC with the included RS232 cable. Select the COM port and the baud rate for the connection.

USB Port Mark this option if you have connected the testcard to the USB port of the control PC. The USB port of the testcard can be used in order to connect more than 4 testcards to one host without using PCI-X connections. With an USB hub, as much as 256 PCI-X testcards can be controlled simultaneously.

Fast Host Interface The fastest—and therefore recommended—connection to your testcard is via the included Fast Host Interface card (using the bi-directional Centronics cable). Starting at port 0, switch through the possible port numbers until the testcard type information section of this dialog displays “Testcard Found On Port”.

Offline/Demo Mode Choose this option if you want to simulate any of the available testcard types or if you temporarily want to disconnect from a testcard. In this mode, no connection to a testcard is established.

Testcard Information This section displays information about the type of the currently selected testcard (if any was found on the selected port). The check boxes can only be toggled manually when you are in Offline/Demo mode.

User Selected Testcard If a testcard was found on the selected port, its name, maximum bus width, and clock speed are displayed in the first line of this section. In Offline/Demo mode, you see a selection list where you choose which of the available testcards you want to simulate.

Support/Licensing Here you see which of the hardware and software components are enabled with your testcard. Grayed items are not supported with your testcard and cannot be licensed. The licensed components are checked. In Offline/Demo mode, you can select any component.

Scan Ports Button The Scan Ports button scans all ports (PCI-X, serial, Fast Host) and displays a list of all found testcards in the Scan All Ports dialog box.

NOTE If you are already connected to the testcard, this connection is not shown in the list.

Ping Button The Ping button sends a test message to the testcard via the selected port. In reply, a message window opens telling if the testcard responded on that port or if it could not be found (for example Offline/Demo mode selected). If a proper connection to the testcard is established, the testcard's LEDs flash.

PCI-X Selection List Dialog Box

This dialog box displays a list of all devices that are connected to the PCI-X bus of the host machine. If this host PC is also the system to be tested, the Agilent E2929A/B testcard is shown here, too, and it can be controlled directly via the PCI-X bus.

If you want to search for testcards on all ports, including the serial port, open the Scan All Ports dialog box by clicking the *Scan Ports* button instead of the *PCI-X Browse* button in the Testcard Configuration dialog box.

Scan All Ports Dialog Box

The Scan All Ports dialog box displays a list of all Agilent PCI-X testcards found on all ports. In contrast to the PCI-X Selection List dialog box, it does not show other devices than Agilent E2920 series testcards. In addition, it searches not only all PCI and PCI-X ports, but also all serial ports and the Fast Host Interface if available.

You can select a port/testcard connection by double-clicking it in the list.

Register Product Options Dialog Box

In case you want to add new hardware or software components to your existing configuration, you need to register them in the Register Product dialog box. Pick the new component from the selection list and enter the license key that was supplied by your vendor in the text field below. After clicking *OK*, this component's GUI functions are enabled.

PCI-X Bus Dialog Box

The PCI-X Bus dialog box is used to check the PCI-X bus properties. These properties are:

- The bus clock speed
- The bus bit width, either 32 bit or 64 bit

In Offline/Demo mode, you can modify these properties.

Update Hardware Dialog Box

After a software update, a version conflict with the onboard firmware can occur. In this case, the hardware must be updated.

When connecting to the testcard and during program start, the hardware check is performed automatically. However, you can also start the check with the *Check Hardware* item in the *Setup* menu. This can be necessary, for example, when switching to another testcard without restarting the software.

To update the hardware, select the control interface to be used and enter the correct port number. After pressing *OK* a message box pops up telling you, whether the update was successful.

Exerciser Menu

With the items of this menu you can access all windows and dialog boxes concerning the PCI-X Exerciser.

The *Exerciser* menu and the *Exerciser* button group are available only if the PCI-X Exerciser option has been installed.

Button Group The four buttons in the main window provide an additional way to access the most frequently used windows of the Exerciser. Which window each button opens is listed below together with the menu items.

Menu Items The following items can be selected:

Settings Here, you find different options for setting exerciser generic properties. Selecting one of these options opens the Exerciser Generic Settings dialog box. This dialog box contains tabs to switch between the options.

Transactions Selecting this item opens the Requester-Initiator Block Editor.

For quick access, click the Requester-Initiator Transaction button .

Behaviors Selecting this item allows you to access the behavior dialog boxes for the Requester-Initiator, the Completer-Target, the Completer-Initiator and the Requester-Target.

Config Space... Selecting this item opens the Configuration Space window.

Target Decode... Selecting this item opens the Target Decode window.

For quick access, click the Target Decode button .

Data Memory... Selecting this item opens the Data Memory Editor window.

For quick access, click the Data Memory button .

Conditional Start... Selecting this item opens the Initiator Conditional Start dialog box.

Interrupt Generation... Selecting this item opens the Interrupt Generator window.

Prog Selecting this item writes all Exerciser settings and properties to the testcard.

Run Selecting this item resets all initiator and target behavior counters and starts the requester-initiator transactions.

Stop Selecting this item completes the current transaction and then stops the requester-initiator.

Clicking the stop button in the main window stops the requester-initiator, too along with all other running components.

Break Selecting this item completes the current transaction, disables the target and resets the complete exerciser afterwards.

Continue Selecting this item continues requester-initiator transactions without resetting the behavior counters.

Reset Selecting this item resets all bus state machines (initiator and target) and clears the requester-initiator intention.

Exerciser Generic Settings Dialog Box

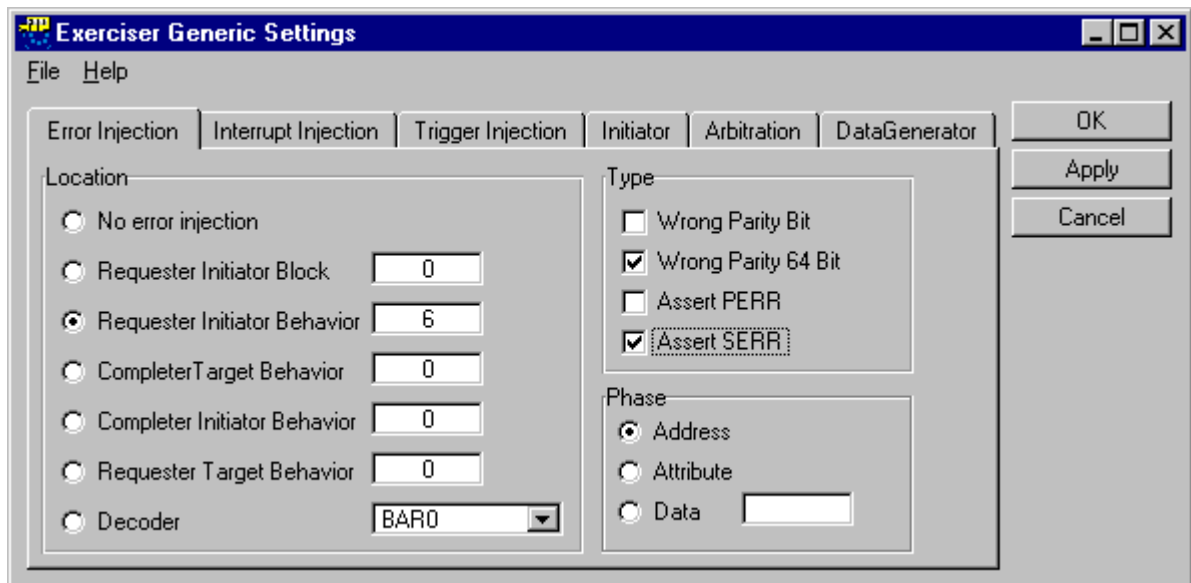
The Exerciser Generic Settings dialog box is used to set up generic exerciser and requester-initiator properties.

This dialog box is displayed after selecting one of the options in the submenu of the *Settings* command in the *Exerciser* menu.

Error Injection Page

The *Error Injection* page is displayed when pointing on the *Exerciser* menu to *Settings*, and then clicking the respective option.

Use this page to generate inverted parity (PAR and PAR64), to signal a parity error (PERR#) or a system error (SERR#) in any participated address phase, attribute phase or data phase.



Location In the *Location* area, you can select whether and from which resource an error is injected. The exact location within that resource can be specified.

Requester Initiator Block The error is injected from the requester-initiator block memory. To specify the block, enter the memory line number in the range 0 ... 255.

Requester Initiator Behavior The error is injected from the requester initiator behavior memory. To specify the behavior, enter the memory line number in the range 0 ... 255.

Completer Target Behavior The error is injected from the completer-target behavior memory. To specify the behavior, enter the memory line number in the range 0 ... 255.

Completer Initiator Behavior The error is injected from the completer-initiator behavior memory. To specify the behavior, enter the memory line number in the range 0 ... 255.

Requester Target Behavior The error is injected from the requester target behavior memory. To specify the behavior, enter the memory line number in the range 0 ... 255.

Decoder The error is injected from the selected decoder address range. You can select between *Bar0*, *Bar1* and *Bar2*.

Type In the *Type* area, you can select the errors that are to be injected.

Wrong Parity Bit/64 Bit A wrong parity (PAR/PAR64) is generated one clock cycle after the phase selected in the *Phase* area.

PERR PERR is asserted two clocks after the data phase selected in the *Phase* area.

SERR SERR is asserted two clocks after the phase selected in the *Phase* area for the duration of one clock.

Phase In the *Phase* area, you can determine when the possible exceptions are generated.

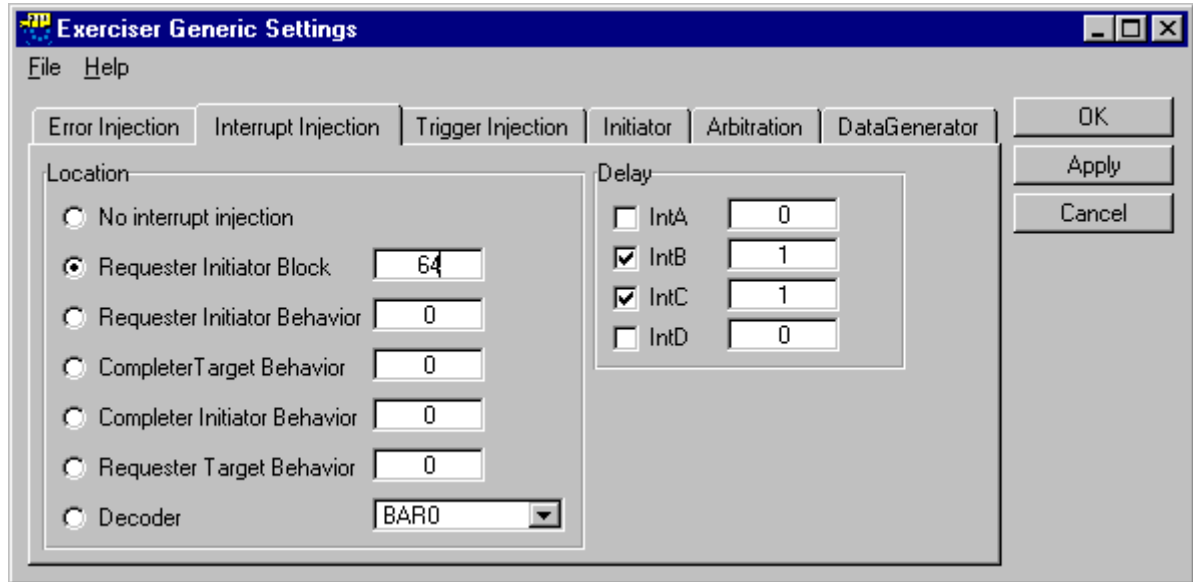
Address The exceptions are generated in the address phase.

Attribute The exceptions are generated in the attribute phase.

Data The exceptions are generated in the data phase with the specified number. The value range for the data phase number is 1 ... 512 (1024 in 32-bit systems).

Interrupt Injection Page

The *Interrupt Injection* page is displayed when pointing on the *Exerciser* menu to *Settings*, and then clicking the respective option.



Location In the *Location* area, you can select whether and when the involvement of the testcard in a particular transaction triggers the generation of interrupts. The exact location within that resource can also be specified.

Requester Initiator Block The interrupt is asserted at the start of the specified requester-initiator block. To specify the block, enter the memory line number in the range 0 ... 255.

Requester Initiator Behavior Interrupt is asserted at the start of the requester initiator behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

Completer Target Behavior Interrupt is asserted at the start of the completer-target behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

Completer Initiator Behavior Interrupt is asserted at the start of the completer initiator behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

Requester Target Behavior Interrupt is asserted at the start of the requester target behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

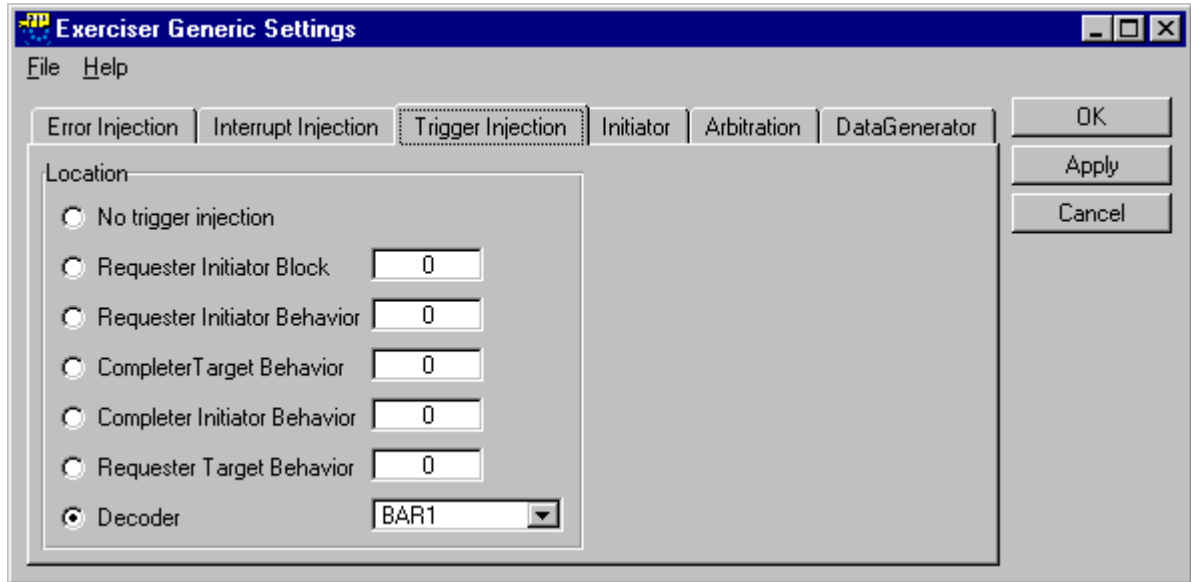
Decoder Interrupt is asserted when the selected decoder address range is accessed. You can select between *Bar0*, *Bar1* and *Bar2*.

Delay In the *Delay* area, you can select if and when the interrupts INTA / INTB / INTC / INTD are asserted.

The interrupt is asserted after the given number of clocks. The valid range is 0 ... 65535.

Trigger Injection Page

The *Trigger Injection* page is displayed when pointing on the *Exerciser* menu to *Settings*, and then clicking the respective option.



Location In the *Location* area, whether and when the involvement of the testcard in a particular transaction asserts the trigger output. The exact location within that resource can also be specified.

Requester Initiator Block Trigger output is asserted at the start of the specified requester-initiator block. To specify the block, enter the memory line number in the range 0 ... 255.

Requester Initiator Behavior Trigger output is asserted at the start of the specified requester initiator behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

Completer Target Behavior Trigger output is asserted at the start of the specified completer-target behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

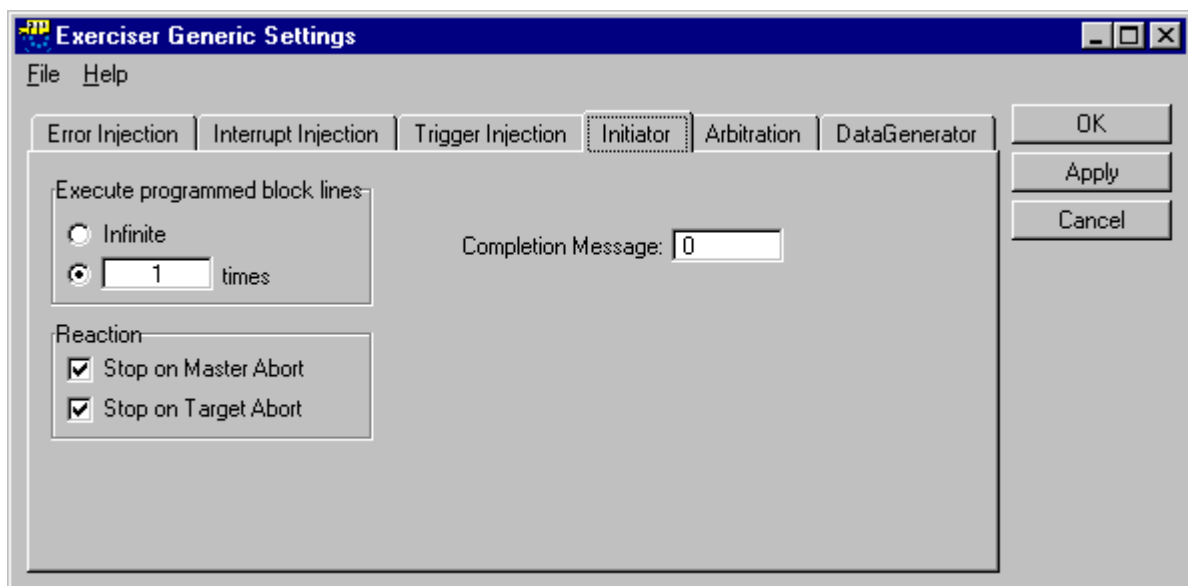
Completer Initiator Behavior Trigger output is asserted at the start of the specified completer initiator behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

Requester Target Behavior Trigger output is asserted at the start of the specified requester target behavior. To specify the behavior, enter the memory line number in the range 0 ... 255.

Decoder Trigger output is asserted when the selected decoder address range is accessed. You can select between *Bar0*, *Bar1* and *Bar2*.

Initiator Page

The *Initiator* page is displayed when pointing on the *Exerciser* menu to *Settings*, and then clicking the respective option.



Execute Programmed Block Lines In the *Execute Programmed Block Lines* area, you can define how often the programmed series of requester-initiator blocks is executed.

Infinite Execution of the programmed block lines will be infinitely repeated until the requester-initiator stop command is executed or another termination condition is met.

... times For a finite number of executions, enter a value in the range 1 ... $(2^{32} - 1)$.

Reaction In the *Reaction* area, you can define if and when execution of the programmed requester-initiator block lines will be stopped.

Stops on Initiator Abort Selecting this check box stops execution immediately after an initiator abort occurs.

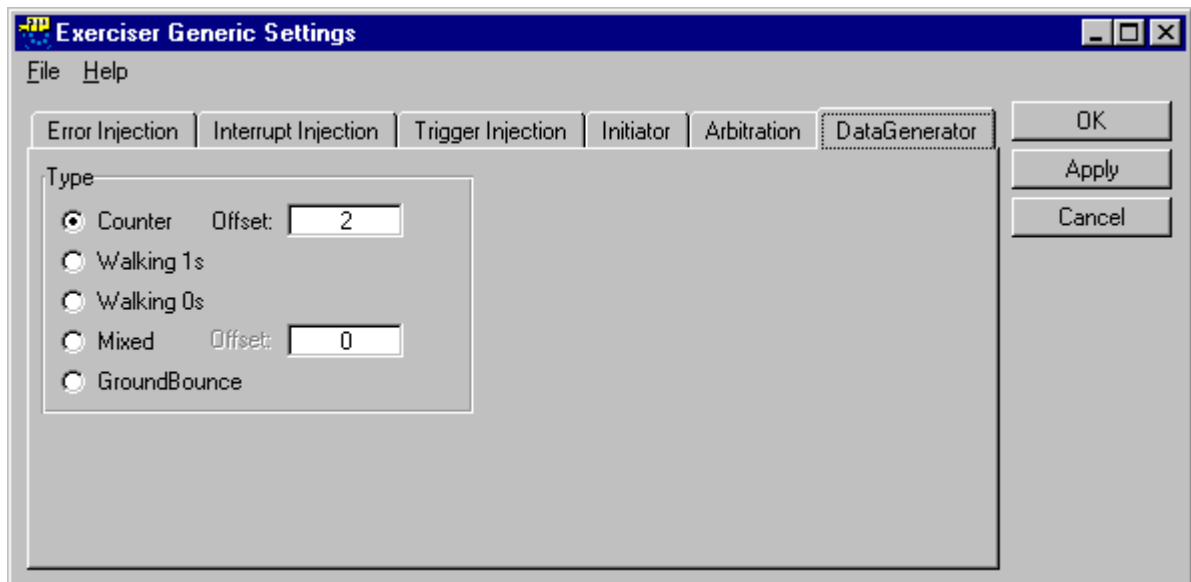
Stops on Target Abort Selecting this check box stops execution immediately after a target abort occurs.

Completion Message In the *Completion Message* text box, you can enter a 32-bit value that defines the content of the split completion message:

- Bits 19 ... 31 will be put on the AD bus during the data phase.
- Bits 0 ... 11 are determined by the remaining byte count. For more information about the byte count, see the “*Requester-Initiator Behavior Editor Dialog Box*” on page 32.

Data Generator Page

The *Data Generator* page is displayed when pointing on the *Exerciser* menu to *Settings*, and then clicking the respective option.



Type In the *Type* area, you can select the type of data generator.

NOTE This option has an effect only if the data generator was selected as data source in the requester-initiator block editor window or in the target decoder window.

Counter Counter with a programmable offset to the bus address. You can enter an offset in the range 0 ... ($2^{20} - 1$).

The counter creates a 64-bit wide data pattern, consisting of two count values from bit 0 ... 22 and from 32 ... 54. The remaining 18 bits can be preset with an arbitrary value or the initiator (requester-initiator and completer-initiator) identification to enable an easy identifier for any seen data in the system.

In the case that the counter is chosen with an initiator identification, the data compare can be switched off for the 18 fixed bits to allow at least unidirectional data verification.

Walking 1s This data generator creates walking 1s.

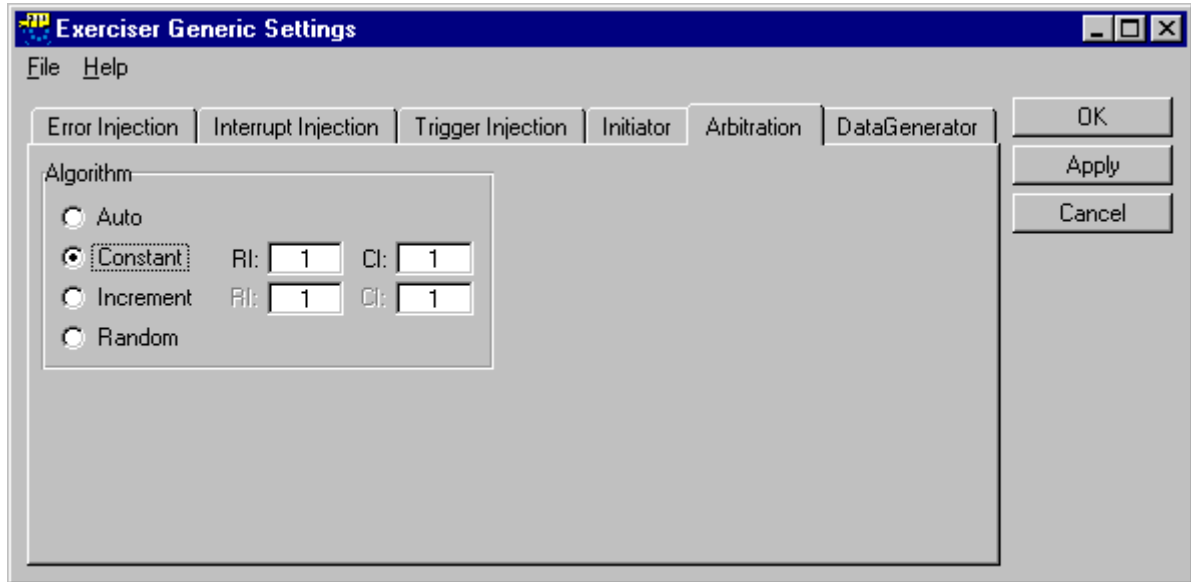
Walking 0s This data generator creates walking 0s.

Mixed Counter with shuffled bits. A 64-bit wide pattern changes with a period of 2^{21} (looks pseudo-random).

GroundBounce This data generator creates 0x00000000 and 0xffffffff data patterns successively.

Arbitration Page

The *Arbitration* page is displayed when pointing on the *Exerciser* menu to *Settings*, and then clicking the respective option.



Algorithm In the *Algorithm* area, you can determine how the internal arbiter decides if the next transaction is a requester-initiator transaction or a completer-initiator transaction.

Auto Requester-initiator transaction and completer-initiator transaction are selected one after the other.

Constant The arbiter takes a fixed number of requester-initiator transactions and then a fixed number of completer-initiator transactions.

You can enter the numbers of requester-initiator and completer-initiator transactions in the valid ranges 1 ... 254.

Increment The arbiter increments the number of requester-initiator transactions and then the number of completer-initiator transactions.

You can enter the numbers of requester-initiator and completer-initiator transactions in the valid ranges 1 ... 254.

With every cycle, both numbers are incremented by one.

Random For every cycle, random values in range of 1...31 for the numbers of requester-initiator and completer-initiator transactions are selected.

The arbiter takes the random number of requester-initiator transactions and then the random number of completer-initiator transactions.

Block and Behavior Editor Dialog Boxes

The block and behavior dialog boxes are structured as a table. The columns of each table represent the properties, the rows represent the lines of the block/behavior memories, where one line is one transaction/behavior.

NOTE You can only select one row. This is done by clicking on a line number in the leftmost column.

These dialog boxes provide the following buttons:

Copy Click the *Copy* button to make the currently selected line the source line that will be used for following Insert/Paste operations. The number of the source line is shown in brackets.

Insert Click the *Insert* button to insert the previously copied line below the selected line.

Paste Click the *Paste* button to overwrite the selected line with the previously copied line.

Default Click the *Default* button to set the selected line to defaults.

Delete Click the *Delete* button to remove the selected line.

For detailed information on each dialog box, refer to the following sections.

Requester-Initiator Block Editor Dialog Box

You can open the Requester-Initiator Block Editor dialog box by clicking *Transactions* in the *Exerciser* menu.

Requester-Initiator Transactions For one transaction, you can define the following properties:

Command To select a command, click the command box of the respective line and select one of the commands, for example, *IOWrite*, in the item list.

AD32/AD64 Enter the lower/upper bits of the bus address.

NumBytes Enter the total number of bytes transferred in this logical block transfer in the range 1 ... 0xffffffff.

NOTE It is possible to cross the 4-GB address boundaries.

Resource Select the data source for data leaving the testcard and entering the testcard: the data memory or the data generator.

IntAddr Enter the internal address of the testcard's internal data memory in the range 0x00000 ... 0xffffffff. The values must have the same QWORD alignment as *AD32*.

DataCmp You can switch the data compare on or off.

CondStart You can define whether the requester-initiator block execution starts unconditionally or whether a conditional start pattern must have occurred:

- *No*

Unconditional start.

- *Once1*

After the exerciser has been started, block execution waits until the conditional start pattern 1 has occurred at least once.

- *Wait1*

After the end of the previous requester-initiator sequence, block execution waits until the conditional start pattern 1 has occurred.

- *Once2*

After the exerciser has been started, block execution waits until the conditional start pattern 2 has occurred at least once.

Byten Enter the values for the byte enables in the range `0000\b ... 1111\b`. The values are shown on `C/BE[3::0]` and `C/BE[7::4]` during all data phases of the block that is written. For a block that is read, this property is ignored.

At the beginning and at the end of a sequence, the byte enables are masked. The mask is determined by the start address and the byte count.

Completion You can select whether all outstanding requests have been completed before the block can be scheduled:

- *NO*
Block execution does not wait.
- *YES*
Block execution waits until all outstanding requests have been completed.

Queue You can select the requester-initiator queue into which the block is put: queue *A*, queue *B* and *Auto*, where *Auto* selects any free queue.

RelaxOrder You can switch the relaxed ordering on or off. *NoSnoop* You can select if snoop will be done.

Reser31 Controls the value of AD[31] in the attribute phase. For programming a PCI-X-compliant behavior, do not set this property to 1.

Requester-Initiator Behavior Editor Dialog Box

You can open the Requester-Initiator Behavior Editor dialog box by clicking *Behaviors* → *Requester-Initiator* in the *Exerciser* menu.

Requester-Initiator Behaviors In the Requester-Initiator Behavior Editor, you can specify the following behaviors:

Queue You can select the requester-initiator queue from which the next sequence is generated: *A*, *B* or *Next*, where *Next* means that the next queue after the last used queue is selected.

Tag Associates a tag to the sequence. Execution waits as long as the desired tag is in use.

- *0 ... 31*

You can select one of these tags. The selected tag will be given to the sequence.

- *Auto*

Any free tag if possible will be given to the sequence.

ByteCount You can select a byte count for the sequence in the range **1 ... 4096**.

NOTE The desired byte count might actually be limited at runtime to the maximum byte count that is specified in the configuration space.

Disconnect You can select if and how often the requester-initiator disconnects its current sequence.

Delay You can select the number of clock cycles that are inserted between transactions. The valid range is **1 ... 65535**.

Steps You can select the number of address steps in the range **0 ... 6**. For programming a PCI-X-compliant behavior, do **not** use values 5 and 6.

The number of address steps is the number of clock cycles between the assertion of GNT# and the assertion of FRAME# plus two clock cycles, which are designed into the register-to-register interface of PCI-X.

According to the PCI-X specification, this property is ignored for configuration cycles. For address steps are always inserted.

RelReq You can define the number of clock cycles after which REQ# is deasserted. The valid range is **1 ... 2047**.

Repeat You can define how often the current behavior is applied before the next behavior is used. The valid range is 1 ... 256.

Skip You can control the address calculations for subsequent transfers.

– *No*

You can select this item to transfer data into a contiguous memory area.

– *1 ... 7*

You can select one of these skip-registers. The value of the skip-register is added to the next address phase.

Completer-Target Behavior Editor Dialog Box

You can open the Completer-Target Behavior Editor dialog box by clicking *Behaviors* → *Completer-Target* in the *Exerciser* menu.

Completer-Target Behaviors In the Completer-Target Behavior Editor, you can specify the following behaviors:

DecSpeed You can select the decode speed for the transaction.

Ack64 You can define whether the 64-bit data transfer acknowledge will be asserted.

Initial You can define how the completer-target responds after waiting the number of cycles specified under *Latency*:

– *Accept*

The target accepts the data transfer or signals a split response.

– *Single*

The target signals a single data phase disconnect.

– *Retry*

The target signals retry.

– *TAbort*

The target signals a target abort.

Latency You can define the number of wait cycles before the completer-target responses.

Subseq You can specify the target response in subsequent data phases, which comes only into effect when the corresponding *Initial* completer-target behavior was set to *Accept*.

– *Accept*

Accepts all subsequent data phases. No target termination is signaled.

– *Disconnect*

Signals a target disconnect in the data phase that can be specified under *SubseqPhase*.

SubseqPhase You can define a subsequent data phase that is used when you set the *Subseq* behavior to *Disconnect*.

SplitLatency You can define the number of wait cycles in the range 3 ... 18 before the split response is signaled as determined. If a split response is not to be signaled as determined (split condition is not met), *Initial* defines the initial behavior.

SplitEnable You can define whether a split response is generated.

To enable split response generation, you must set up the decoder accordingly (see “*Target Decode Window*” on page 41).

Repeat You can define how often the current behavior is applied before the next behavior is used. The valid range is 1 ... 65536.

Completer-Initiator Behavior Editor Dialog Box

You can open the Completer-Initiator Behavior Editor dialog box by clicking *Behaviors* → *Completer-Initiator* in the *Exerciser* menu.

Completer-Initiator Behaviors In the Completer-Initiator Behavior Editor, you can specify the following behaviors:

Queue You can select the request queue from which the next completion is generated.

– *Next*

The next queue is selected following the one selected previously. This gives every queue a chance to complete.

– *None*

No queue is selected. The requests will be accumulated in the request queues for out-of-order completion.

– *Auto*

Any non-empty available queue is selected.

– *ASkip ... DSkip*

Queue A/B/C/D is selected for the next completion transaction. If this queue is currently empty, the current behavior is skipped.

– *AWait ... DWait*

Queue A/B/C/D is selected for the next completion transaction. If this queue is currently empty, execution waits until this queue gets a request.

CAUTION

Be very careful using this option, because requests in other queues will not be completed if the selected queue does not receive a request.

For programming a PCI-X-compliant behavior, do not use these options. A device is not supposed to make a completion dependent on the receipt of another transaction

Addrxx You can switch on controlling the state of a reserved bit of the AD[] bus in the respective address phase by selecting *Yes*. If you select *No* this property is ignored and the corresponding bits are copied from the request. For programming a PCI-X-compliant behavior, do **not** select *No*.

Attrxx You can switch on controlling the state of a reserved bit of the AD[] bus in the respective attribute phase by selecting *Yes*. If you select *No* this property is ignored and the corresponding bits are copied from the request. For programming a PCI-X-compliant behavior, do **not** select *No*.

ErrMsg You can switch the generation of a split completion message on or off.

Partition You can define whether and how the size of the (partial) completion transaction is limited.

– *No*

The full byte count is transferred without disconnecting the completion.

– *1 ... 63*

The completion is disconnected at every n-th allowable disconnect boundary (ADB) after the current start address, where n is in the range 1 ... 63.

CondStart You can define if the completion starts unconditionally or if a conditional start pattern must have occurred.

– *No*

Unconditional start.

– *Once1*

After the exerciser has been started, block execution waits until the conditional start pattern 1 has occurred at least once.

– *Wait1*

After the end of the previous completer-initiator sequence, block execution waits until the conditional start pattern 1 has occurred.

– *Once2*

After the exerciser has been started, block execution waits until the conditional start pattern 2 has occurred at least once.

Delay You can select the number of clock cycles that are inserted between transactions. The valid range is 1 ... 65535.

Steps You can select the number of address steps in the range 0 ... 6. For programming a PCI-X-compliant behavior, do **not** select numbers 5 and 6.

The number of address steps is the number of clock cycles between the assertion of GNT# and the assertion of FRAME# plus two clock cycles, which are designed into the register-to-register interface of PCI-X.

According to the PCI-X specification, this property is ignored for configuration cycles. For address steps are always inserted.

Req64 You can select if the 64-bit data transfer request is asserted.

This property is not evaluated if it is used with a DWORD command. REQ64 will not be asserted for DWORD commands, regardless of the property setting.

RelReq You can define the number of clock cycles after that REQ# is deasserted. The valid range is 1 ... 2047.

Repeat You can define how often the current behavior is applied before the next behavior is used. The valid range is 1 ... 265.

Requester-Target Behavior Editor Dialog Box

You can open the Requester-Target Behavior Editor dialog box by clicking *Behaviors* → *Requester-Target* in the *Exerciser* menu.

Requester-Target Behaviors In the Requester-Target Behavior Editor, you can specify the following behaviors:

DecSpeed You can select the decode speed for the transaction.

Ack64 You can define whether the 64-bit data transfer acknowledge will be asserted.

Initial You can define how the requester-target responds after waiting the number of cycles specified under *Latency*.

– *Accept*

The target accepts the data transfer or signals a split response.

– *Single*

The target signals a single data phase disconnect.

– *Retry*

The target signals retry.

– *TAabort*

The target signals a target abort.

Latency You can define the number of wait cycles before the split response is signaled. If a split response is not to be signaled as determined (split condition is not met), *Initial* defines the initial behavior.

Subseq You can specify the target response in subsequent data phases, which comes only into effect when the corresponding *Initial* completer-target behavior was set to *Accept*.

– *Accept*

Accepts all subsequent data phases. No target termination is signaled.

– *Disconnect*

Signals a target disconnect in the data phase that can be specified with *SubseqPhase*.

SubseqPhase You can define a subsequent data phase that can be used when you set the *Subseq* behavior to *Disconnect*.

Repeat You can define how often the current behavior is applied before the next behavior is used. The valid range is 1 ... 65536.

Configuration Space Window

This window displays the configuration space header of the target.

On the Agilent E2929A/B testcard, the configuration space is implemented with two equally sized memories. One contains the actual value of the configuration space header, the other holds a mask, defining which bits are fixed (the value 0 or 1 is displayed) and which are changeable by the BIOS or another program routine (indicated by an x).

Basically any bit of the configuration space header is freely programmable. Nevertheless, a fair amount of knowledge is needed to make settings that are compliant to the PCI-X specification. The values in the Base Address register entries of the configuration space header can be modified by changing values in the Target Decode window.

Table Contents The lines of the table represent the different registers of the configuration space header. Some lines contain several registers, indicated by a vertical line separator in the *Register* column. The columns of the table show from left to right:

... The hexadecimal byte address relative to the first byte of the configuration space.

Register The name(s) of the register(s) according to the PCI-X specification.

Contents [BIOS view] The current values in the register(s) as shown to the BIOS.

Capabilities [x=changeable] These fields contain a representation of the registers' programming mask.

– 0 or 1

This bit is fixed (read-only) and its value cannot be changed by the BIOS or an other routine.

– x

This bit is programmable by the BIOS as well as by other routines.

- Respect Bios** If you select the *Respect Bios* check box, all settings that are relevant for the BIOS to handle the assignment of the address spaces to the different target devices on the PCI-X bus, cannot be changed.
- Read from card** If you click the *Read from card* button, the current settings in the configuration space header on the Agilent E2929A/B testcard—including the mask—are loaded into this window.
- Write to card** If you click the *Write to card* button, the current contents of this window will be written to the configuration space header and the mask memory of the Agilent E2929A/B testcard.
- If a bit has got a fixed value in the right column, a differing value in the *Contents* column will be ignored.
- Match Indicator** At the bottom of this window, the match indicator shows if the current settings of the testcard and the contents of this window match (Green=Yes, Red=No).
- To load the current testcard settings into the GUI, click the *Read from card* button. To write the edited settings onto the testcard, click the *Write to card* button. After that, the match indicator is green again.
- Both actions will also update the Target Decode window. Furthermore, they will set the Target Decode window's match indicator to red, because no guarantee can be given to the correctness of settings that are not related to the configuration space.
- When changes in the *Contents* column are made to bits that are set in the right column, these changes will be ignored. In this case, the match indicator keeps its color. If it was green, this may be confusing, because the entries in the two fields do not match anymore. Click the *Read from card* button to reload the correct values.

Target Decode Window

This window is used to view and edit the setup of the target decoders. These decoders are needed for the target to react to PCI-X transactions that are driven onto the bus by initiators. A decoder claims a transaction if the address on the bus is within its address range.

Decoder Types The Agilent E2929A/B testcard provides the following decoders:

- 3 programmable decoders responsible for decoding memory and I/O cycles.

These decoders are named *BAR 0-LO*, *BAR 0-HI*, *BAR 2-LO*, *BAR 2-HI*, *BAR 4* and *BAR 5*, which corresponds to the Base Address Registers 0 through 5 in the configuration space header.

- A standard non-programmable configuration space decoder for configuration cycles.
- An expansion ROM decoder for accesses to the expansion ROM. This decoder can be used, for example, to test whether BIOS correctly accesses this memory, or to test power-on-self-tests or interrupt service routines stored in this memory.
- A requester-target decoder for split responses.

Decoder Settings To change a setting, click the corresponding field in the table. Either a check box, a selection list or an editable text field turns up, allowing you to modify the setting. If a field is gray, it cannot be modified.

For the decoders the following settings are available:

Location, Prefetchable These parameters correspond to the last four bits in the BARs in the configuration space header and define where the BIOS must place the address range.

Resource Determines the resource to which the decoder is connected. Possible choices are:

- *resourcemem*
Selects the data memory.
- *resourcegen*
Selects the data generator.
- *resourcecfg*
Selects the configuration space.
- *resourcedbi*
Selects the internal DBI registers of the PCI-X Asic.

NOTE If initiator and target of the Agilent E2929A/B testcard are set up to transfer data between each other, the target has no access to the data memory. If the initiator writes to the target, data memory will remain unchanged. If the initiator reads from the target, it will receive dummy data.

Res. Base This specifies the start address of the data resource in the testcard's data memory.

Res. Size This value defines the size of the allocated resource memory. If this size is smaller than the size of the decoded address range, the resource contents will be overwritten/sent multiple times to fill the decoder address range.

Base Addr., Size Determines the address range of the decoder. It corresponds to the base address registers in the configuration space.

If *Base Addr.* is gray, these registers were set by the BIOS. To modify the settings, clear the *Respect Bios* check box.

CAUTION

If any changes to the address ranges were done manually, ensure that multiple targets do not decode overlapping address ranges. Multiple devices decoding overlapping address ranges may result in hardware damage.

Split Switches the split decoder on or off.

- Respect Bios** If you select the *Respect Bios* check box, the base address (*Base Addr.*) cannot be changed. These registers were set by the BIOS during power up.
- Read from card Button** Click this button to load the current settings from the Agilent E2929A/B testcard into this window. This will cause the match indicator to turn green.
- Write to card Button** Click this button to write the current window contents to the Agilent E2929A/B testcard. If the entries are successfully written to the testcard, the match indicator turns green. BIOS relevant entries cannot be stored with this button. They can only be saved as power-up settings for the next session.
- Write PU to card Button** Click this button if you want to save the current window contents as default values for the next power up (PU). If you have not already clicked the *Write to card* button with these settings, the match indicator now shows a yellow color.
- Match Indicator** In the bottom line of this window you see the match indicator. The indicator shows one of the following colors:
- Red
The current settings on the testcard and the contents of this window do not match.
 - Yellow
The power-up settings on the testcard match the window contents, but not the settings currently in use on the testcard.
 - Green
The current settings on the testcard match the window contents, but not necessarily the power-up settings for the next power up.

Data Memory Editor Window

The data memory on the Agilent E2929A/B testcard is used by both the master and the target, both as data source and as a destination for received data. It has a size of 1024 KB.

If master and target of the Agilent E2929A/B testcard are set up to transfer data between each other, the target has no access to the data memory. If the master writes to the target, data memory will remain unchanged. If the master reads from the target, it will receive dummy data.

Table Contents The two columns of the table contain:

Address Displayed is the lowest byte address of the bytes shown in the data column. The addresses are relative to the first byte of the data memory.

Data This column contains the data located at the corresponding address. The value in brackets in the column title tells the amount of data per row.

Buttons This window provides buttons that have the same functionality as the respective menu items.



- Select View Of Data

If you want to change the data display mode, click this button to open the Select View Of Data dialog box.



- Zero out Data Memory

Clicking this button will fill the complete data memory with zeros (0). This can take several minutes to finish.



- Refresh Data

Clicking this button reloads the data from the data memory on the testcard into this window.



- Print

Clicking this button opens a print dialog. Printed is always the currently visible part of the table only, not the complete data memory.

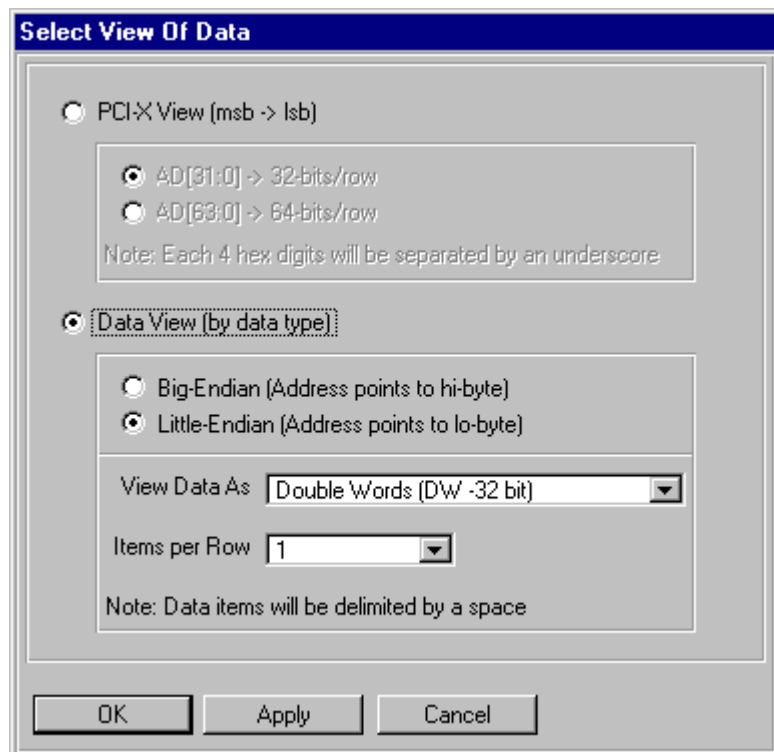
If you want to print more than the visible range, save the data to a file, open it with any word processor and print it from there.

- *Goto:*

With this feature, you can directly go to a specific address without using the scroll bar.

Select View Of Data Dialog Box

This dialog box is used to modify the view of the presented data in the Data Memory Editor window.



You can choose between two different ways available to display the data.

PCI-X View This view displays the data starting from the most significant bit (msb) down to the least significant bit (lsb). The data is grouped in words (four hex values) separated by underscores.

AD[31:0] Clicking this option shows 32 bit per row.

AD[63:0] Clicking this option to shows 64 bit per row.

Data View Select this view if you want to display the data grouped by the data type. There are several additional choices to modify the view to match your needs.

Big-Endian If you select this option, the byte with the lowest address in this row will be displayed first.

Little-Endian If you select this option, the byte with the highest address in this row will be displayed first.

View Data As Use this selection list to determine the type the data is to be displayed as: one byte, one word, one double-word, or one quad-word.

Items per Row Use this selection list to determine the amount of data displayed per row. The available options depend on the selected data type, the maximum amount per row is 64 bit or one quad-word. The different items in a row are separated by a space.

Initiator Conditional Start Dialog Box

With this dialog box, you can specify two different start conditions for the initiator transactions.

These start conditions can be selected in the Requester Initiator Block Editor. The Exerciser only starts the transactions when the specified pattern occurs on the PCI-X bus (or on the external trigger input ports if specified).

To edit the pattern expressions, click the Edit button. This opens the Pattern Editor dialog box.

After you finished specifying a start condition, it appears in the text field.

Interrupt Generator Window

With the use of this window you can directly assert and deassert any combination of the PCI-X interrupts.

- INTA# (default)
- INTB#
- INTC#
- INTD#

To assert certain interrupts, select the interrupts and click the *Assert* button. To deassert certain interrupts again, select them and click the *Deassert* button.

The reaction of your system under test to asserted interrupts strongly depends on its configuration.

CAUTION

Asserting interrupts can cause your system under test to crash. If you are running the control software on the same machine, all unsaved data will be lost.

Analyzer Menu


With the items of this menu, you can access all dialog boxes and windows concerning the PCI-X Analyzer.

NOTE Out of the Protocol Check, the Analyzer features are only available with Opt. 100 of the Agilent E2920 PCI-X series.


Button Group The four buttons in the main window provide an additional way to access the most frequently used windows of the Analyzer. Which window each button opens is listed below together with the menu items.

Menu Items *Protocol Check* Selecting this item opens the Protocol Check window.

Capture Selecting this item opens the Capture dialog box.

For quick access, click the Capture button .

Waveform Viewer Selecting this item opens the Waveform Viewer window.

For quick access, click the Waveform Lister button .

Bus Cycle Lister Selecting this item opens the Bus Cycle Lister window.

For quick access, click the Bus Cycle Lister button .

Transaction Lister Selecting this item opens the Transaction Lister window.

For quick access, click the Transaction Lister button .

Run Choosing the run item starts the PCI-X Analyzer, but none of the other components.

Stop Choosing the stop item stops the PCI-X Analyzer.

Protocol Check Window

The Agilent E2929A/B testcard checks the PCI-X bus for protocol violations in real time. Use this window to view the current protocol check results.

The table lists all 53 PCI and PCI-X protocol rules and Agilent rules that the Agilent E2929A/B testcard checks for. To get a brief description of a rule, double-click the name field. The *Mask* fields show whether a rule is enabled or disabled, while the *Status* fields show their status, either *OK* or *ERROR*.

Status In the *Status* section, the total number of violated rules is displayed, as well as the name of the rule that was violated first.

NOTE The protocol checker is always running on the Agilent E2929A/B testcard. The information presented in this window will be not up to date if errors occurred after the last update of this window.

To update the list and the total number of violated rules, click the *Read from card* button.

To reset the list (no more errors) and read again from the testcard, click the *Clear* button.

NOTE If a reported protocol error cannot be cleared with the *Clear* button, this indicates that the error still persists on the bus.

Enable/Disable If you want to trigger on a particular selection of protocol rules, you can enable and disable them by clicking on the fields in the *Mask* column.

NOTE A change in the mask column clears all displayed errors.

Disabled errors will not be reported. They will still be displayed (now yellow instead of red) but will be ignored. To enable a disabled signal, click the *Mask* field again.

Capture Dialog Box

NOTE This dialog box is only available if *Setup -> Options -> Trigger Sequencer* is unchecked.

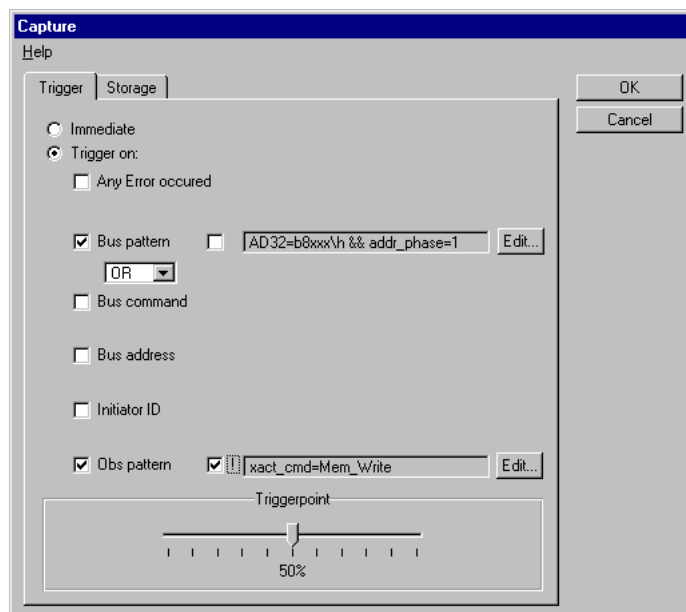
The Capture dialog box has two tabs in which you can define the settings for the trace memory.

Trigger Tab The trigger defines when to start the recording of the trace memory. To define the trigger qualifier, use the Trigger tab.

Storage Tab To specify which of the bus cycles on the PCI-X bus to record in the trace memory, use the Storage tab.

Trigger Tab

Use this tab to set the trigger qualifier. This qualifier then defines when to start the recording of the capture.



Immediate This is the default option. The capture will be triggered immediately after the Analyzer starts running.

Trigger On: This option allows you to select the events that trigger the capture. The events can be OR- or AND-combined.

– *Any Error occurred*

If you select this option, the capture is triggered on protocol-, data compare and split errors.

– *Bus pattern*

The bus pattern is a boolean expression that is compared to the bus signals. When the bus pattern occurs, the capture is triggered. To enter or edit the pattern, use the Pattern Editor that opens when you click the *Edit* button.

To negate the specified pattern, check the box in front of this pattern. This allows disjunction of the signals.

– *Bus command*

Selecting this option opens an item list where you can select the bus command. When the specified bus command is seen, the capture is triggered.

– *Bus address*

Selecting this option uncovers text boxes to enter a lower and an upper 32-bit value for the bus address. When the specified bus address is used, the capture is triggered.

– *Initiator ID*

Selecting this option allows you to enter the initiator identification (requester-initiator and completer-initiator) to enable an easy identifier for any seen data in the system. The value range is 18 bits.

– *Obs pattern*

The observer pattern is a boolean expression that is compared to the bus signals. When the observer pattern occurs, the capture is triggered. To enter or edit the pattern, use the Pattern Editor that opens when you click the *Edit* button.

To negate the specified pattern, check the box in front of this pattern. This allows disjunction of the signals.

Triggerpoint This slider defines the position of the triggerpoint in the trace memory.

- 0% means no pre-trigger bus traffic is kept in the trace memory.
- 100% means that only pre-trigger traffic is stored.

Storage Tab

With this tab, you can define which bus cycles are captured in the trace memory and which are ignored. The options are:

All Selecting this option causes all bus cycles to be captured in the trace memory.

Selected Transactions/States This option allows you to specify a boolean expression for the storage qualifier. Then only transactions matching this pattern will be stored.

To enter or edit the expression in the text field, open the Pattern Editor Dialog Box by clicking the *Edit* button.

If you select *Selected Transactions/States*, you can also suppress certain bus states. The two choices are:

Supress Idle Cycles This option filters out all idle cycles.

Supress Wait States This option causes that no wait states are stored.

Pattern Editor Dialog Box

This dialog box is used to specify patterns used for conditional expressions. It provides a table with two columns for the signals and their value of interest.

If specifying the storage qualifier, only four different signals appear in the list. The trigger qualifier, however, can be set up for a wide range of signal and bus state values.

Entering Values The method of entering the values depends on the type of the signal:

- If the signal is a bit vector, for example, the address and data bus signal *AD32*, you can enter the value directly into the value field. Possible formats are hexadecimal (entry ends with *\h*), decimal (*\d*) and binary (*\b*).
- If the signal is a single bit, for example, the initiator ready signal *IRDY*, you can select a value from the list in the value field. The possible values are 0, 1, and x (don't care).

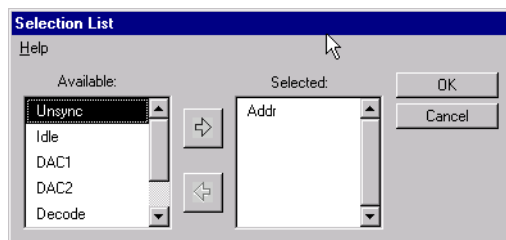
- If the signal is a type that can take several different values, for example, the bus state *bstate* or the transaction command *xact_cmd*, you can select a value. When clicking the value field the Selection List dialog box opens to present the possible options.

Clearing Values To remove a single value from the selected pattern, select *DON'T CARE* (or *x* for bits). To reset all signals click the *Clear* button.

Creating the Pattern The boolean expression that makes up the pattern is generated when leaving this dialog box by clicking the *OK* button. It then appears in the pattern line in the window from where you opened this dialog box.

Selection List Dialog Box

This dialog box helps you to specify a pattern in the Pattern Editor window. More specifically, it opens only when you are selecting the values of signals that are neither single bits nor bit vectors.



The value can be selected from the *Available* list which contains all the states that this signal can take.

Trigger Sequencer Dialog Box

NOTE This dialog box has two different appearances depending on the *Options* you have selected in the *Setup* menu. If you uncheck the *Trigger Sequencer* option, this dialog box is called Capture dialog box.

This dialog box is used to specify the sequencer that controls the trigger and storage qualifier depending on the occurrence of a sequence of patterns.

Default Buttons This dialog box provides two *Default* buttons that set conditions and patterns, or conditions only to defaults.

CAUTION

The *Default* buttons clear all entries you have done to specify transitions and/or patterns.

Transitions The sequencer description table consists of rows describing certain states and transitions as used in a state diagram. Usually there is more than one row for each state. Each row describes a transition to another state. A maximum of 64 transitions can be programmed for up to four states.

Each row holds the following information:

Cur State The identity number of the state described by the row. There may be more than one row for one state.

Next State The identity number of the state to be taken when the transaction condition turns true.

XAct Cond. If this transaction condition turns true, the state specified by *Next State* is taken. All transaction conditions belonging to one *Cur State* must be mutually exclusive.

Trig Cond., SQ Cond. Specify the output conditions for the trigger and storage qualifier.

FBA Dec. Cond., FBB Dec Cond. The feedback conditions are used to decrement the loop variables that start counting at the preload values *FBA* and *FBB* specified below the transition table. If these loop variables reach zero, the respective terminal count signals (*tc_fba*, *tc_fbb*) are set, which can be used as variables in the other conditions.

FBA Load Cond., *FBB Load Cond.* The load conditions define when the loop variables are reset to the preload values *FBA* and *FBB* specified below the transition table.

Feedback Counter The valid values for feedback counter A and B are: 00000000\h ...
ffffffff\h.

Patterns You can specify two bus signal patterns, up to four observer signal patterns and one error signal pattern.

bus0, *bus1* To specify the pattern, all bus signals and the external trigger signals can be used combined with a logical AND .

obs0 ... *obs3* To specify the pattern, all observer signals (bus state, decoding state and so forth) can be used combined with a logical AND.

err0 To specify the pattern, all error signals (protocol error, data compare error and split transaction error) can be used combined with a logical OR.

These different signal patterns can be used for the conditional transitions or outputs in the sequencer.

These patterns are boolean expressions that are set with the use of the Pattern Editor that opens when clicking the Edit button.

Triggerpoint This slider defines the position of the triggerpoint in the trace memory.

- 0 percent means no pre-trigger bus traffic is kept in the trace memory.
- 100 percent means that only pre-trigger traffic is stored.

Waveform Viewer Window

This window displays all signals stored in the trace memory in a waveform diagram. You can inspect the value of every signal at any point in the recorded capture. Two markers can be set in the timing diagram for easier navigation.

Signal List The left part of the diagram shows a list of all the displayed signals. For non-boolean signals the *Val(A)* column shows the value of this signal where marker A is set.

To rearrange the signals or to change the selection of displayed signals, open the Arrange Signals dialog box by selecting the *Arrange* item from the *Signals* menu.

Waveform Diagram The right part of the diagram shows the waveforms of the signals. Due to the large amount of data, only a small section of the diagram can be shown at a time. Use the scroll bars to scroll through the diagram.

The values of the non-boolean signals are shown, if the space is sufficient.

The text fields below the diagram show the first (left text field) and last (right text field) sample currently displayed. The selection list between the two is used to select the resolution. As an example, 5 *sa/div* means that there are 5 samples or clock cycles between two divisional marks in the top row of the diagram.

In the top row you can also see the markers A and B, and the triggerpoint T, if they are set in the currently displayed section.

The small white arrow buttons on both sides of the horizontal scroll bar are additional page increment/decrement buttons.

Vertical gray lines indicate gaps in the recorded data. This occurs if certain samples were filtered out by the storage qualifier during the capture. The storage qualifier is set in the Storage tab in the Capture dialog box.

Marker Positions Down to the left of the window you see three text fields. They contain the current positions of the two markers A and B and the distance between them. The unit *sa* stands for samples.

To modify them you, can either use the small plus and minus buttons or enter the desired position manually in the text fields.

Slider Bar Down to the right there is a slider bar representing the complete range of samples stored in the trace memory. It shows the triggerpoint T (always located at sample 0) and the positions of the markers A and B, if they are set.

While the trigger T is always located at sample 0, you can drag the markers A and B to new positions.

Buttons All functions provided by the buttons can also be called via the respective menu items (except for the Redraw button). They are listed from left to right and can be identified as follows:



- Run Analyzer
This button starts the Analyzer.



- Stop Analyzer
This button stops the Analyzer. This could be necessary if the trigger is not met at all or if the storage qualifier is limiting the captured data so that the trace memory is not filled within a reasonable time.



- Reload Trace Memory
This function reloads the waveforms from the memory.



- Goto Triggerpoint
Clicking this button displays the signals around the trigger with the current resolution.



- Goto Marker A
Clicking this button displays the signals around marker A with the current resolution.



- Goto Marker B
Clicking this button displays the signals around marker B with the current resolution.



- Move Marker A into View
This button moves marker A into the center of the current view.



- Move Marker B into View
This button moves marker B into the center of the current view.



- View Range between Marker A and B
This button displays the range between the two markers (if not too far apart).



- Move Marker B near to Marker A
This button sets marker B close to marker A. This means the distance then is 1 while the order of the markers is maintained as before.



- Zoom out



- Zoom in



- Cross Reference

Clicking the Cross Reference button will select the respective bus cycles in the range between the markers A and B in the Bus Cycle Lister and the Transaction Lister.



- Redraw

This button redraws the window.

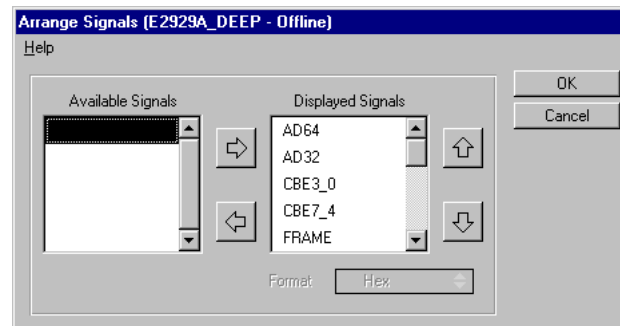


- Print

This button opens a print dialog box where you can print the window.

Arrange Signals Dialog Box

The Arrange Signals dialog box is used to rearrange the order of the signals in the Waveform Viewer window or to change the selection of the displayed signals.



Signal Lists The list on the left contains all the available signals that are not currently displayed in the waveform viewer.

The right list contains all the signals that are displayed in the waveform viewer.

Arrow buttons Use the left and right arrow buttons to move selected signals between the two signal lists.

Use the up and down arrow buttons to move a selected signal in the right list one position.

Format If a signal is selected that is not boolean, you can select the format in which the signal's value will be displayed in the waveform viewer. You can choose between displaying it as a decimal or hex number.

Bus Cycle Lister Window

This window presents the contents of the trace memory in terms of bus cycles. Each row represents one bus cycle and contains information such as the cycle number, the duration, the bus states, the PCI-X commands, and the detected address and data signals.

Which bus cycles are stored in the trace memory is defined by the storage qualifier that is set in the Storage tab in the Capture dialog box.

Buttons The first three buttons are used to run and stop the Analyzer, and to reload the trace memory.



- Goto Triggerpoint

With the Goto Triggerpoint button you can jump to the bus cycle numbered 0.



- Cross Reference

Clicking the Cross Reference button will select the respective bus cycles in the Waveform Viewer and the Transaction Lister.

- *Goto:*

The *Goto* text field can be used to jump to a particular bus cycle number.

You can select a range of bus cycles by clicking on one and dragging the mouse over the desired range with the left mouse button held down.

Export to Text File If you want to save the list of bus cycles as a text file, choose *Export to File* from the *File* menu. To save the currently selected range choose *Export selected Range*.

Search Menu With the items in the *Search* menu you can search for any string or for errors.

Transaction Lister Window

This window is used to display all the transactions that occurred on the PCI-X bus while the Analyzer was running and storing the capture. Because several bus cycles usually make up one transaction, this view can be seen as a higher level of abstraction than the view of the Bus Cycle Lister.

The presented transactions give information about the number of the first bus cycle, the PCI-X command type, the first address, data information (if applicable), and the detected attributes, including the occurred protocol errors.

Buttons The first three buttons are used to run and stop the Analyzer, and to reload the trace memory.



- Goto Triggerpoint

With the Goto Triggerpoint button, you can jump to the bus cycle numbered 0.



- Cross Reference

Clicking the Cross Reference button will select the respective bus cycles in the Waveform Viewer and the Bus Cycle Lister.



- Split Request Search

With this button, you can switch between split response and completion message transaction.

- *Goto:*

With this feature, you can directly go to a particular transaction number.

You can select a range of transactions by clicking on one and dragging the mouse over the desired range with the left mouse button held down.

Export to Text File If you want to save the list of transactions as a text file, choose *Export to File* from the *File* menu.

Search Menu With the items in the *Search* menu, you can search for any string or for errors.

Filter Menu With *Suppress Data Transfers* in the *Filter* menu, you can horizontally filter out all data transfers. This is useful to remove long bursts.

Performance Menu

The items of this menu provide access to the dialog boxes and windows concerning the Agilent E2972A Performance Optimizer.

The *Performance* menu is available only if the Performance Optimizer option has been installed.


Button Group The three buttons in the main window provide an additional way to access the most frequently used windows of the Performance Optimizer. Which window each button opens is listed below together with the menu items.

Menu Items The following list gives a brief overview of the use of the menu items.

Performance Setup Selecting this item opens the Performance Setup window.

For quick access, click the Performance Setup button  .

Performance Report To see the results of the performance measurements presented as a textual report, select this item. The Performance Report window opens.

For quick access, click the Performance Report button  .

Performance Charts To see the results of the performance measurements presented in charts and diagrams, select this item. The Performance Charts window opens.

For quick access, click the Performance Charts button  .

Run Selecting this item starts a performance measurement according to the settings in the performance setup. When the measurements are finished, the results are loaded into the respective windows.

Stop Selecting this item stops a running performance measurement and loads the recorded data into the result windows. The latter is done in any case, even if no measurement was running.

Performance Setup Window

This window is used to set up measurements to evaluate and optimize the performance of the system under test.

Setup Tabs The performance measurement setup is done in the four tabs of this window.

Completer Identification To evaluate particular completers, specify their identification in the Completer Identification tab.

Requester Identification To evaluate particular requesters, specify their identification in the Requester Identification tab.

Pair Select To evaluate the performance of a particular requester-completer pair, use the Pair Select tab.

Capture To specify conditions for the start and the end of the performance measurements, use the Capture tab.

Buttons This window also provides some buttons that have the same functionality as the respective menu items.

- Run

Clicking this button runs the Performance Optimizer. When the measurements are finished, the results are loaded into the GUI.

- Stop

Clicking this button stops running measurements and loads the results into the GUI.

- Rescan

Clicking this button recalculates the performance measurement results from the trace memory using the current setup and loads them into the GUI.

- Print

Clicking this button opens a dialog box for printing the current view of the Performance Setup window.

Completer Identification Tab

In this tab, you can specify the completer identification. You have to use this to do performance measurements on one or more particular completer devices. Every completer is identified by the address range that it decodes. You can specify up to ten different address ranges.

Table Contents The Completer Identification tab is structured as a table. Every row contains the identification of one completer decoder. The fields of the columns contain:

Completer Name Here you can enter a name for the completer decoder. This name will then appear in the report and the displays.

Base Address This is the base address of the completer's decoder. Use \h to indicate hex values.

Size This is the size of the completer decoder's address range. Use \h to indicate hex values.

Command This can be used to restrict the performance measurements on *I/O* commands or memory commands. *None* means that no measurements are made on this completer.

Requester Identification Tab

In this tab, you can specify the requester identification. You have to use this to do performance measurements on one or more particular requester devices. Every requester is identified by its bus, device, and function number. You can specify up to ten different requesters.

In PCI-X systems every slot has its own bus, device, and function number.

Table Contents The Requester Identification tab is structured as a table. Every row contains the identification of a requester device. The fields of the columns contain:

Requester Name Here you can enter an name for the requester. This name will then appear in the report and the displays.

Bus, Device and Function Here you have to enter the bus, device and function number of the requester device selected as device under test.

NOTE You can get these numbers when running the performance measurement and viewing the test report. Report section 1.1 lists all requesters found on the bus with their corresponding bus, device, and function numbers.

Pair Select Tab

You can use this tab to monitor the traffic between a particular requester-completer pair.

Requester Here you can select the requester you want to observe from the selection list. This list contains “All requesters” and the requesters that you have specified in the Requester Identification tab of this window.

Completer Here you can select the completer you want to observe from the selection list. This list contains “All completers” and the completers that you have specified in the Completer Identification tab of this window.

Command Type Here you can select which command types are of interest in your observation. You can either monitor read commands, write commands, or both types.

Title Enter an title that represents your requester-completer pair. This title will then appear in the report and the displays.

Capture Tab

You can use this tab to specify a trigger for the start of the performance measurements. Furthermore you can define how long the measurements will run.

Trigger The trigger defines when the performance measurements start. The trigger condition uses a boolean expression to determine if a certain pattern occurs on the PCI-X bus. To edit the pattern, click the *Edit* button. This opens the Trigger tab in the Capture dialog box.

Any changes made in the Storage tab of the Capture dialog box are ignored—the storage qualifier is set to *Suppress Idle Cycles*.

End of Analysis You have the following options for specifying the length of the performance measurements. The possible options are:

Stop at end of Tracememory The complete trace memory will be filled with data before the performance results are calculated.

Stop after x samples With this option, you can measure the performance on a certain number of samples. Note that idle states are not counted as samples. Thus, this value does not tell the number of bus cycles during this measurement. The maximum number of samples allowed is 1,048,575.

Stop after access to address With this option, the test is stopped after a certain address was accessed. This address must be entered in hex format in the range from 0x00000000 to 0xffffffff.

Additionally, you have to select the bus command that is supposed to access this address to stop the test.

NOTE If you store the waveform file after a repetitive test run, only the data of the most recent test will be stored. Later analysis of this file will possibly yield results other than the statistics on the complete set of tests.

Performance Charts Window

The Performance Charts window presents the results of the performance measurements in different charts and diagrams.

Chart Tabs This window provides the following four tabs:

PCI-X Usage General PCI-X usage statistics are presented in the PCI-X Usage tab.

Burst Usage The use of bursts in the examined traffic is displayed in the Burst Usage tab.

Command The distribution of the different bus commands in the examined traffic is shown in the Command tab.

Latency Information on the different types of latencies that might occur on the bus is found in the Latency tab.

Buttons This window also provides some buttons that have the same functionality as the respective menu items.

- Run

Clicking this button runs the Performance Optimizer. When the measurements are finished, the results are loaded into the GUI.

- Stop

Clicking this button stops running measurements and loads the results into the GUI.

- Rescan

Clicking this button recalculates the performance measurement results from the trace memory with the current setup and loads them into the GUI.

- Print

Clicking this button opens a dialog box for printing the current view of the Performance Charts window.

Setup To view or modify the performance measurement setup, open the Performance Setup window by selecting *Setup Window* from the *Setup* menu.

PCI-X Usage Tab

This tab displays the general statistics on the usage of the PCI-X bus. In the lower right corner you see which requester(s) and completer(s) were considered in these measurements. The value in parentheses indicates whether only write commands, read commands or both contributed to the statistics.

The Diagrams On this tab you see the following five charts:

Bus Utilization Corresponds to report section 8.2. This diagram shows the distribution of the following data:

- Idle cycles (blue)
- Overhead cycles (red)
- Split overhead (yellow)
- Data (green)

If the traffic of a particular requester-completer pair was measured, the communication of all other requester-completer-pairs is counted as idle here, too.

Split Statistics Corresponds to report section 8.4.1. This diagram shows the following data:

- The number of *Normal Transaction* (green)
- Transactions that are split with exactly one *Split Completion* (yellow)
- Transactions that are split with more than one *Split Completion* (blue)

PCI-X Efficiency Corresponds to the report section 8.3. This bar diagram shows the calculated efficiency of the communication of the selected requester-completer pair. This is the actual amount of transferred data divided by the maximum amount of data theoretically transmittable during the used number of bus cycles.

PCI-X Throughput This diagram shows the throughput in percent of the maximum theoretical possible amount of data that could be transferred (dependent on bus-width and speed).

Command Usage Corresponds to the report section 8.2.3. This chart shows the distribution of the different bus commands in the data traffic. The green section represents the data phases, while the red part is the transaction overhead. The command abbreviations are:

- IOR, IOW: I/O Read and I/O Write
- MRD, MWR: plain Memory Read and Memory Write
- CRD, CWR: Configuration Read and Configuration Write
- MRB, MWB: Memory Read Block and Memory Write Block

Burst Usage Tab

This tab provides a diagram that shows the distribution of the burst lengths that contributed to the bus traffic. This diagram corresponds to report section 8.3.1.

Burst lengths are displayed starting with 0, 1 and then increasing in ranges $\{2^n, 2^{(n+1)} - 1\}$, where n ranges from 1 ... 11.

The maximum burst length is 4096.

Furthermore, for every burst length, this diagram shows three small squares. These squares are described below:

Data Fraction (red) This square shows what percentage of all data phases is part of a burst of this particular length. The sum of the data fractions equals one.

Bus Fraction (green) This square shows what percentage of all bus cycles during bus activity (no idle states) is part of a burst of this particular length. The sum of the bus fractions equals one.

Efficiency (blue) This square shows the average efficiency of a burst of this particular length. The efficiency of all burst lengths will not sum up to one.

Command Tab

This tab provides a diagram that displays the distribution of the different bus commands during the traffic (compare the PCI-X Usage tab of this window). More specifically, it is split up with respect to the different burst lengths. This diagram corresponds to report section 8.2.5.

Burst lengths are displayed starting with 0, 1 and then increasing in ranges $\{2^n, 2^{(n+1)} - 1\}$, where n ranges from 1 ... 11.

The y-axis here shows what percentage of all transactions in the traffic were transactions of a particular burst length with a particular command. The sum of all bars in this diagram is one.

The key above the diagram shows the colors that represent each command.

Latencies Tab

This tab provides a diagram that displays the lengths of the *First Word Latency* (red) measured in number of clock cycles. This diagram corresponds to report section 8.5.

First Word Latency is the time from the start of the transaction until the begin of the first data phase.

The y-axis here shows what percentage of these latencies that occurred were of a particular type and length. The sum of all bars in this diagram is 100 percent.

Performance Report Window

This window displays a report as text of the performance measurements. The report is generated by evaluating the data captured in the trace memory.

Click on figures displayed in blue color shows the report section that dissolves this performance value in more detail.

Buttons The window provides some buttons that have the same functionality as the respective menu items.

- Run

Clicking this button starts the Performance Optimizer. When the data recording is finished, the measurement results are loaded into the GUI.

- Stop

Clicking this button stops running measurements and loads the results into the GUI.

- Rescan

Clicking this button recalculates the performance measurement results from the data in the trace memory and presents them in the GUI.

- Go back

With this button you can go back to the previously clicked blue value.

- Print

Clicking this button opens a dialog box for printing the whole report or only a selection of it.

File Menu In the *File* menu, you can find the following additional features that cannot be accessed with any of the buttons:

Load This item loads a waveform file (*wfm*-file) into the GUI.

Note, that the *wfm*-file needs to be saved in performance mode to allow performance calculations.

Save to File This item saves the contents of the trace memory as a *wfm*-file.

Export to File This item saves the performance report as a *txt*-file.

Close This item closes this window.

Run Menu

In the *Run*, menu you can make use of the following items:

Options If you want to specify some run options for your test, select this item to open the Run Options window.

Start This item starts the currently specified test with the selected options.

Stop Choose this item to stop the current test.

Run Options Window

This window is used to set options for the test run. The selected options then apply for the Run button in this window, the Run button in the main window, and the Start item in the *Run* menu.

Runs Analyzer Check this box if you want to run the PCI-X Analyzer in the test.

Runs Performance Check this box if you want to run the PCI-X Performance Optimizer in the test.

Runs Exerciser Check this box if you want to run the PCI-X Exerciser (requester-initiator) in the test.

Clears protocol errors Check this box if you want to clear the protocol error lists before starting the test.



Windows Menu

In the *Windows*, menu you can make use of the following items:

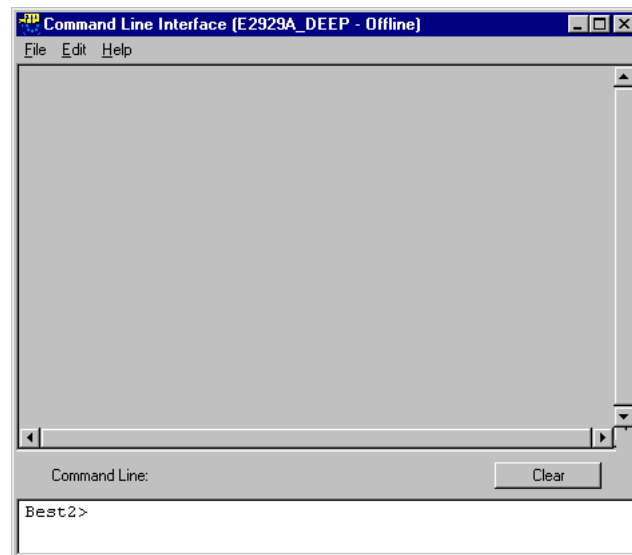
Command Line This item opens the Command Line Interface window.

Close All Choose this item to close all dialog boxes and windows, except for the main window.

Command Line Interface Window

This Command Line Interface (CLI) window can be used to directly call the Agilent E2929A/B testcard's C functions instead of writing a complete C program. For every C function, there is a corresponding CLI command. Additionally, for every command there is shortcut abbreviation provided for typing convenience.

Command Descriptions A complete description of all CLI commands and their abbreviations can be found in the *Agilent E2929A/B C-API/PPR Reference*.



Window Contents The window is divided into two areas. The large gray text area is the console output area. It is not editable and contains all your entries along with the results returned by the C functions.

The smaller white text area at the bottom holds your command line. Enter your function calls here with the required parameters and press return. This CLI also features command line history for reuse of previously entered commands. Press the up and down arrow keys to browse through the last 50 commands.

The *Clear* button to the right clears both areas.

Menu Items There are some more features provided in the menus of this window.

In the *File* menu you can find:

New Selecting this item clears both text areas.

Save As Selecting this item allows you to save the contents of the output area as a text file for later use.

Start Logging Selecting this item allows you to specify a file name, where all your CLI commands will be stored (not the results returned by the C functions) as a CLI script.

Stop Logging Selecting this item stops logging the commands in the CLI script file.

Run Script Selecting this item allows you to run a previously logged CLI script file. Note, that execution stops at the first error that occurs, for example due to missing parameters in a function call.

Close Selecting this item closes the CLI window.

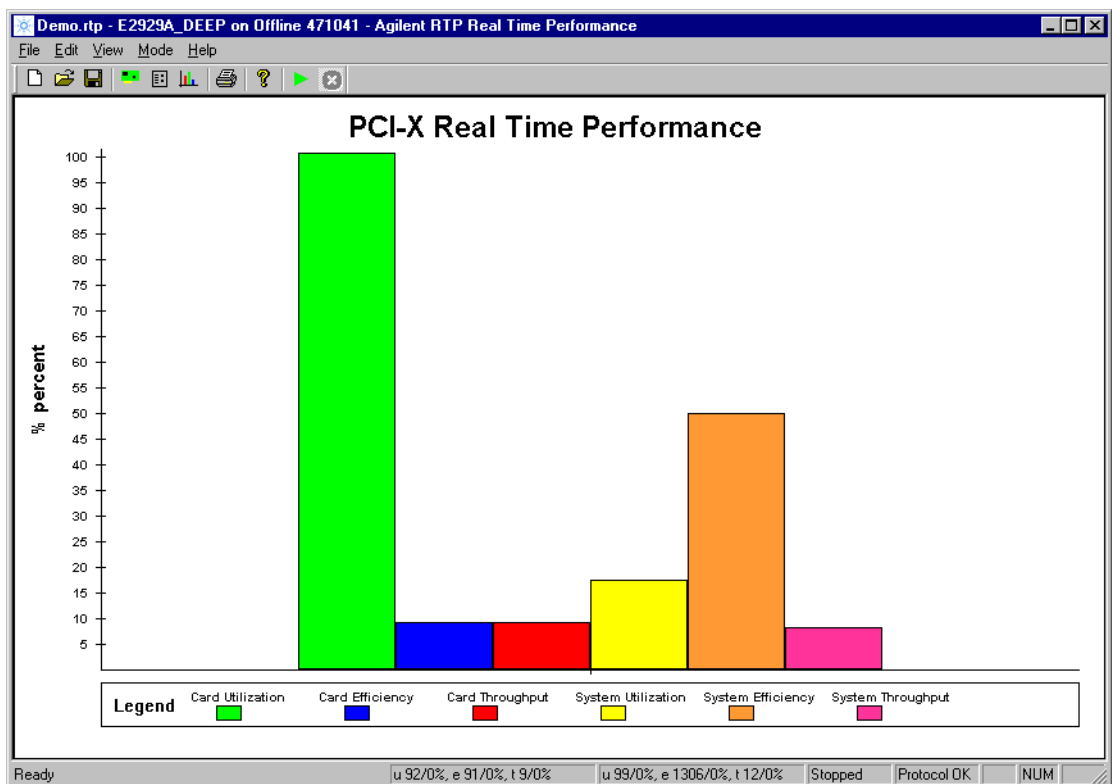
In the *Edit* menu, you can find the items *Copy* and *Paste*. You can reuse commands or parts of them by selecting them in the output area with your mouse and then selecting the *Copy* and *Paste* items of this menu.

PCI-X Real Time Performance

The Agilent E2920 PCI-X software provides additional features for real-time performance measurements. These features are available via a separate GUI, which can be accessed by selecting:

Start > Programs > Agilent E2920 PCI-X > PCI-X Real Time Performance GUI

The user interface for the RTP Real Time Performance measurements looks as follows:





Menus

This section describes the features that are unique to the Real Time Performance (RTP) application. All features are accessible within the menus in the menu bar or within the shortcut menus in the navigator.

File Menu

The *File* menu items can be used to handle file handling actions, and to run and stop the real-time performance measurements.

The *File* menu contains standard desktop features, and features unique to the RTP application, such as:

- *Open*
To load a previously saved session, click this item. The session file (.rtf file) can then be selected from the *Open File* dialog box.
- *Run*
To run the current measurement, click this item. When running in offline mode, random values are generated.
For quick access, click the *Run* icon  in the tool bar or use the shortcut menus.
- *Stop*
To stop the running measurement, click this item. For quick access, click the *Stop* icon  in the tool bar or use the shortcut menus.
- *Print ...*
To print the measurement results as shown on the display, click this item.

Edit Menu

The *Edit* menu can be used to set up the measurements.

Select Testcard... To view all connected testcards and to select one for the measurement, click this item. This opens the RTP Select Testcard window. See “*RTP Select Testcard Window*” on page 82.

For quick access, click the Select Testcard button  .

Setup Testcard... To view and modify testcard settings, select this item. This opens the RTP Testcard Setup window. See “*RTP Testcard Setup Window*” on page 83.

For quick access, click the Setup Testcard button  .

Setup Measurement ... To select one or two predefined standardized performance measurements, such as PCI-X Throughput, select this item. This opens the RTP Measurement Setup window. See “*RTP Measurement Setup Dialog Box*” on page 86.

For quick access, click the Setup Measurement button  .

View Menu

The *View* menu items can be used to switch the *Toolbar* or *Status Bar* on or off and to set display, measurement and report options.

The *View* menu contains standard features and features unique to the RTP application, such as:

- *Options ...*

To specify the update interval for the measurement, the report file and the display options, click this item. This opens the “*RTP Options Dialog Box*” on page 87.

Mode Menu

The *Mode* menu can be used to switch between online and offline mode.

Go Online To switch on the connection to the system under test, click this item. The online mode allows you to run the measurement and to generate a report.

Go Offline To switch off the connection to the system under test, click this item. Offline mode can be used for demonstration purposes, or to test various settings. When running in offline mode, random values are generated.

Agilent RTP Real Time Performance Window

The Agilent RTP Real Time Performance window is the user interface for setting up and running real-time performance measurements. The Agilent RTP Real Time Performance window consists of a menu bar, a tool bar and a status bar, and displays the results calculated by means of the performance counters.

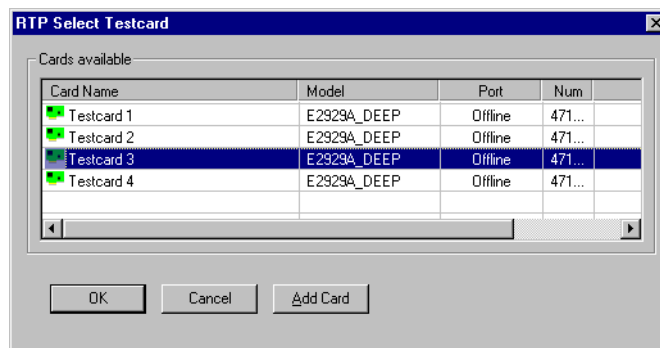
Start and Stop Buttons Use the *Start* and *Stop* buttons to control the performance measurement.

Status Bar The status bar shows the current measurement results and gives information about the current operation mode. The status bar also indicates whether protocol errors occurred during the measurement.

RTP Select Testcard Window

The Select Testcard window is displayed when you click the respective item in the *Edit* menu.

For quick access, click the Select Testcard button  .



By default, the software is in online mode and automatically scans the testcards connected to the system under test. In online mode, this window displays all available testcards and shows where they are installed.

Add Card Button If it is not possible for you to insert the required testcards into the system under test, you can set up a performance test in offline mode. In offline mode, the *Add Card* button in the Select Testcard window allows you to add testcards necessary for the test.

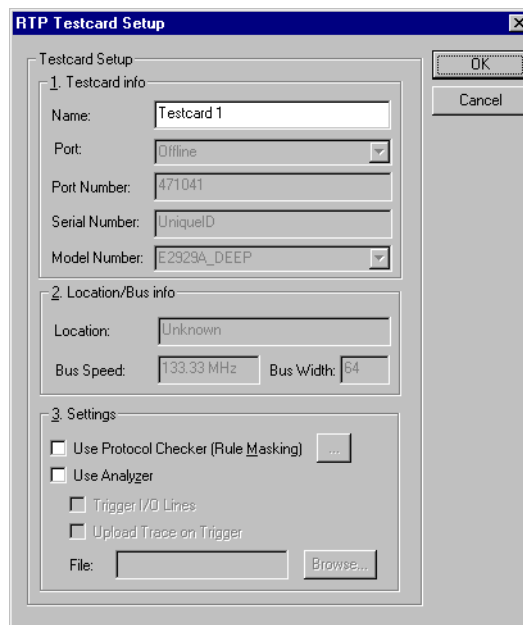
You can select the testcard required for the performance measurement. The selected testcard will be displayed in the RTP Setup Testcard window.

RTP Testcard Setup Window

The Setup Testcard window is displayed when you click the respective item in the *Edit* menu.

For quick access, click the Setup Testcard button .

The Setup Testcard window gives information about the testcard and the PCI-X bus it is plugged into. This window also allows you to modify testcard settings.



Testcard Info *Testcard info* contains product and installation identifiers. These entries cannot be modified in online mode. They are determined by the current card and its installation.

Location/Bus Info *Location/Bus info* identifies the testcard by its bus, device and function number and gives information about PCI-X bus properties.

Settings The *Settings* group contains several check boxes used to enable or disable the current card features in online and offline mode.

TIP It is a good idea to switch off any features that are not necessary for your test. This results in shorter run times.

The check boxes are:

- *Use Protocol Checker (Rule Masking)*

To use the protocol checker, this check box must be selected. Clicking the details button next to this check box opens the “*Protocol Rule Masking Dialog Box*” on page 85 where you can enable and disable protocol rules.

- *Use Analyzer*

This option allows you to use the Analyzer functionality of the testcard. If checked, the Analyzer triggers as soon as an error (data-compare error or protocol violation) occurs. To mask protocol rules, use the *Protocol Checker (Rule Masking)* feature described above.

The captured data is stored in the testcard's trace memory and can later be uploaded for analyzing.

The following check boxes are only available when the *Use Analyzer* check box is selected.

- *Trigger I/O Lines*

To enable cross-triggering with the test or to connect an external logic analyzer, this check box must be selected. Cross-triggering means that you connect the trigger I/O ports of two testcards.

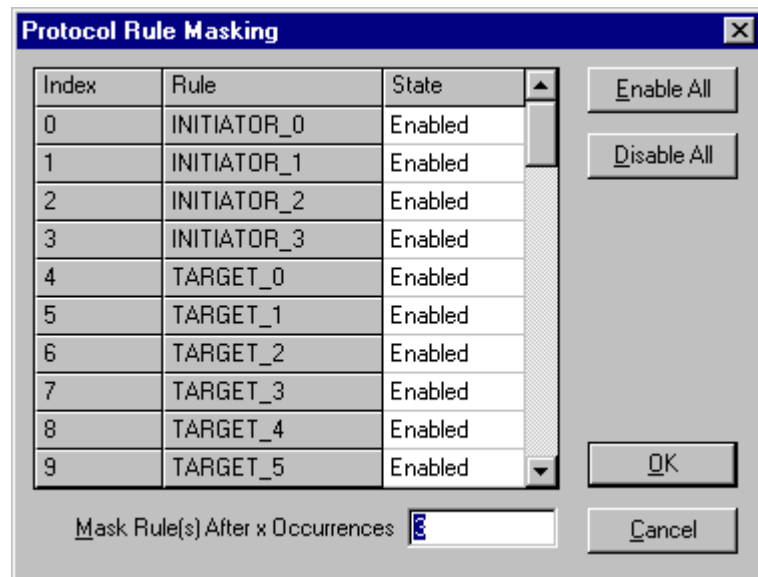
- *Upload Trace on Trigger*

To write the trace memory from the testcard to a *.wfm file (default is *tracemem*) after the testing is triggered, this check box must be selected. The trace memory waveform file can be used for in-depth root cause analysis. This option usually is not recommended, because it can take very long time to complete.

To change the name or the directory where the file should be stored, click the details button next to the *File* edit field.

Protocol Rule Masking Dialog Box

The Protocol Rule Masking dialog box lists all protocol rules defined by the PCI-X specification and some Agilent rules (see the *Agilent C-API/PPR Programming Reference*, which is delivered with the PCI-X testcard).



Each testcard has a protocol observer that monitors the PCI-X bus in real time to detect any protocol violations. This dialog box allows you to enable and to disable protocol rules.

Enable or Disable all Rules You can enable or disable all rules in the table by clicking the respective button. A right-click in the rule table opens a shortcut menu, where you can also select these features.

Enable or Disable one Rule You can enable or disable a selected rule by using the shortcut menu, which is available by right-clicking in the rule table. Clicking in the *State* column of the selected rule changes the current state of this rule.

Rule Description You can get detailed information by selecting *Rule Description* in the shortcut menu of a selected rule. Clicking on the rule identifier also shows the description of the selected rule.

For example, if you click on IRDY 0, the following description is shown:

```
BX_OBSRULE_IRDY_0:
IRDY# must be asserted two clocks after the attribute
phase. PCI-X Spec 1.10.2. Rule 3b
```

Rule Masking after a Specified Number of Occurrences

The Protocol Rule Masking dialog box allows you to mask rules after a specified number of occurrences, so that storage of identical rule violations in the test report is limited. The number can be entered in the *Mask Rule(s) After x Occurrences* field.

RTP Measurement Setup Dialog Box

The Setup Measurement window is displayed when you click *Setup Measurement...* in the *Edit* menu.

For quick access, click the Setup Measurement button .

This dialog box allows you to set up two performance measures to be displayed in the Agilent RTP Real Time Performance window.

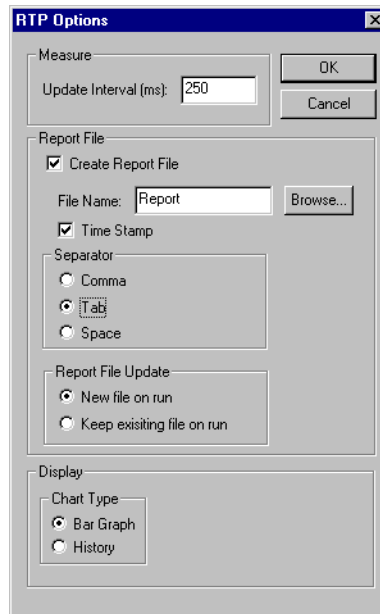
In the Setup Measurement window, you can select standardized predefined measurements to be run either for the testcard, or the system, or both.

The standardized predefined measurements are PCI-X throughput, PCI-X utilization, PCI-X efficiency, retry rate and split rate.

NOTE At a time, you can run only two measurements.

RTP Options Dialog Box

The RTP Options dialog box is displayed when clicking *Options* in the *View* menu. This dialog box allows you to specify measurement, report and display options.



Measure Options You can enter a time interval in the range 100 ... 7FFFFFFF\h. The update time is the time taken between two samples. It is used for both the screen update and the report file.

Report File Options Using the *Report File* options, you can create a report file containing all results of the measurements selected in the RTP Measurement Setup dialog box per time interval.

In addition, all protocol errors occurred during the measurement are listed in the *Trigger Observer* column.

Example:

Time	Utilization	Efficiency	Throughput	Utilization	Efficiency	Throughput	Trigger	Observer
0.25	0.789636	0.117744	0.092975	0.361126	0.296522	0.107082	""	""
0.501	0.863582	0.033210	0.028680	0.542405	0.226959	0.123104	""	""
0.751	0.081423	0.805472	0.065584	0.094668	0.341715	0.032350	""	""
1.001	0.232765	0.078701	0.018319	0.931639	0.010749	0.010014	""	""
1.252	0.058718	0.716151	0.042051	0.398602	0.135719	0.054098	""	""
1.502	0.837184	0.079765	0.066778	0.693533	0.071678	0.049711	""	""
1.752	0.004334	15.160211	0.065699	0.398694	0.075589	0.030137	""	""
2.003	0.255135	0.335123	0.085501	0.435499	0.255519	0.111278	""	""
2.253	0.940977	0.079910	0.075194	0.576678	0.030873	0.017804	""	""
2.503	0.383007	0.001394	0.000534	0.082247	1.002968	0.082492	""	""
2.754	0.064852	1.563294	0.101382	0.691488	0.145103	0.100337	""	""
3.004	0.685629	0.026028	0.017846	0.727897	0.133568	0.097217	""	""
3.255	0.868679	0.092744	0.080565	0.085452	0.807455	0.068999	""	""
3.505	0.058779	0.584761	0.034371	0.981780	0.078936	0.077498	""	""
3.755	0.922483	0.049802	0.045942	0.218635	0.089144	0.019490	""	""
4.006	0.521439	0.216296	0.112785	0.902646	0.061171	0.055216	""	""
4.256	0.782098	0.027456	0.021474	0.775872	0.140227	0.108798	""	""

To specify the report, the following options are available:

Time Stamp Check this option to display the measurement results subject to the time. The report contains an additional *Time* column.

Separator To prepare the results for further processing with other tools, you can select how the results are separated.

Report File Update By default, running a new measurement creates a new report file. To list several performance measurements in one file, select *Keep existing file on run*.

Display Options The following chart types are available:

- *Bar Graph* shows the results in a bar diagram. It always shows the results of the last measurement time interval specified under *Measure*.
- *History* shows the result history over time, which is especially useful to detect peaks.

Publication Number: 5988-4902EN

