Instruction Manual

Tektronix

TMS817 and TMS818
PCIExpress Bus Supports

071-1214-00

www.tektronix.com

Tektronix, Inc., P.O. Box 500, Beaverton, OR 97077

SOFTWARE WARRANTY

Tektronix warrants that the media on which this software product is furnished and the encoding of the programs on the media will be free from defects in materials and workmanship for a period of three (3) months from the date of shipment. If a medium or encoding proves defective during the warranty period, Tektronix will provide a replacement in exchange for the defective medium. Except as to the media on which this software product is furnished, this software product is provided “as is” without warranty of any kind, either express or implied. Tektronix does not warrant that the functions contained in this software product will meet Customer’s requirements or that the operation of the programs will be uninterrupted or error-free.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period. If Tektronix is unable to provide a replacement that is free from defects in materials and workmanship within a reasonable time thereafter, Customer may terminate the license for this software product and return this software product and any associated materials for credit or refund.

THIS WARRANTY IS GIVEN BY TEKTRONIX IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX’ RESPONSIBILITY TO REPLACE DEFECTIVE MEDIA OR REFUND CUSTOMER’S PAYMENT IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

HARDWARE WARRANTY

Tektronix warrants that the products that it manufactures and sells will be free from defects in materials and workmanship for a period of one (1) year from the date of shipment. If a product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; c) to repair any damage or malfunction caused by the use of non-Tektronix supplies; or d) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Table of Contents

Getting Started

Operating Basics

Specifications

Reference

Replaceable Parts

Glossary

Index

List of Figures

List of Tables

General Safety Summary

Review the following safety precautions to avoid injury and prevent damage to this product or any products connected to it. To avoid potential hazards, use this product only as specified.

Only qualified personnel should perform service procedures.

While using this product, you may need to access other parts of the system. Read the *General Safety Summary* in other system manuals for warnings and cautions related to operating the system.

To Avoid Fire or Personal Injury

Use Proper Power Cord. Use only the power cord specified for this product and certified for the country of use.

Connect and Disconnect Properly. Do not connect or disconnect probes or test leads while they are connected to a voltage source.

Connect and Disconnect Properly. Connect the probe output to the measurement instrument before connecting the probe to the circuit under test. Disconnect the probe input and the probe ground from the circuit under test before disconnecting the probe from the measurement instrument.

Ground the Product. This product is grounded through the grounding conductor of the power cord. To avoid electric shock, the grounding conductor must be connected to earth ground. Before making connections to the input or output terminals of the product, ensure that the product is properly grounded.

Observe All Terminal Ratings. To avoid fire or shock hazard, observe all ratings and markings on the product. Consult the product manual for further ratings information before making connections to the product.

Do not apply a potential to any terminal, including the common terminal, that exceeds the maximum rating of that terminal.

Do Not Operate Without Covers. Do not operate this product with covers or panels removed.

Use Proper Fuse. Use only the fuse type and rating specified for this product.

Avoid Exposed Circuitry. Do not touch exposed connections and components when power is present.

Do Not Operate With Suspected Failures. If you suspect there is damage to this product, have it inspected by qualified service personnel.

Do Not Operate in Wet/Damp Conditions.

Do Not Operate in an Explosive Atmosphere.

Keep Product Surfaces Clean and Dry.

Provide Proper Ventilation. Refer to the manual's installation instructions for details on installing the product so it has proper ventilation.

Symbols and Terms

Terms in this Manual. These terms may appear in this manual:

WARNING. *Warning statements identify conditions or practices that could result in injury or loss of life.*

CAUTION. *Caution statements identify conditions or practices that could result in damage to this product or other property.*

Terms on the Product. These terms may appear on the product:

DANGER indicates an injury hazard immediately accessible as you read the marking.

WARNING indicates an injury hazard not immediately accessible as you read the marking.

CAUTION indicates a hazard to property including the product.

Symbols on the Product. The following symbols may appear on the product:

Service Safety Summary

Only qualified personnel should perform service procedures. Read this *Service Safety Summary* and the *General Safety Summary* before performing any service procedures.

Do Not Service Alone. Do not perform internal service or adjustments of this product unless another person capable of rendering first aid and resuscitation is present.

Disconnect Power. To avoid electric shock, switch off the instrument power, then disconnect the power cord from the mains power.

Use Care When Servicing With Power On. Dangerous voltages or currents may exist in this product. Disconnect power, remove battery (if applicable), and disconnect test leads before removing protective panels, soldering, or replacing components.

To avoid electric shock, do not touch exposed connections.

Preface

This instruction manual contains specific information about the TMS817 and TMS818 Bus support products and is part of a set of information on how to operate this product on compatible Tektronix logic analyzers.

If you are not familiar with operating support products, you need to supplement this instruction manual with introductory information about how to set up and run a support package on the logic analyzer. Go to the logic analyzer online help index under Microprocessor support packages.

For help in understanding terms in this manual that may be new to you, see the Glossary at the end of the manual.

Manual Conventions

This manual uses the following conventions:

- The term “disassembler” refers to the software that decodes bus cycles into instruction mnemonics and cycle types.
- The terms “Master” and “Slave” refer to modules that are located in numbered slots (see Figure 1–1 on page 1–2).

Contacting Tektronix

Phone	1-800-833-9200*
Address	Tektronix, Inc. Department or name (if known) 14200 SW Karl Braun Drive P.O. Box 500 Beaverton, OR 97077 USA
Web site	www.tektronix.com
Sales support	1-800-833-9200, select option 1*
Service support	1-800-833-9200, select option 2*
Technical support	Email: techsupport@tektronix.com 1-800-833-9200, select option 3* 6:00 a.m. - 5:00 p.m. Pacific time

*** This phone number is toll free in North America. After office hours, please leave a voice mail message.**
Outside North America, contact a Tektronix sales office or distributor; see the Tektronix web site for a list of offices.

Getting Started

Getting Started

This section contains the following information for the TMS817 and TMS818 PCIExpress Bus Support products:

- Product description
- Logic analyzer configuration
- Connecting the logic analyzer to the target system

Product Description

The TMS817 and TMS818 probe adapters are designed to connect to a midbus or slot target system. This allows you to acquire data from a PCIExpress Bus with little effect on the target system.

When a probe adapter is connected to the target system, the signals from the PCIExpress system flow through the probe head, into the preprocessor unit, and then through the LAI cables to the logic analyzer.

The TMS817 and TMS818 bus software provides decoding of the serial-data stream. There are five software packages, x1, x2, x4, x8, and x16, one for each of the PCIExpress bus-width links and three test software packages.

- TMS817 supports x1, x2, x4, x8, and x16 lane widths
- TMS818 supports x1, x2, and x4 lane widths

The probe adapter assumes that all lanes in a link are affected the same way by repeater chips. Specifically, if the repeater chip adds or deletes x SKIP symbol(s) from a SKIP packet on one lane it adds or deletes x SKIP symbols from the same SKIP packet on all lanes.

Trigger Support

Trigger libraries containing EasyTrigger programs are provided for each support package. See page 2–3 for a list of the trigger programs.

Logic Analyzer Software Compatibility

The label on the CD-ROM states that version 4.2 of the logic analyzer software is compatible with the TMS817 and TMS818 products.

Logic Analyzer Configuration

To use either of the probe adapters, you need a minimum module speed of 450 MHz. See Table 1–1 for the number and type of modules needed for different lane widths.

If you use more than one module, the modules must be merged. See the logic analyzer online help for how to merge your modules.

Table 1-1: Number and Type of modules

Lane width	Module type	Module number
x1, x2, x4	TLA7AX3* or TLA7AX4**	1
x8	TLA7AX4**	1
x16	TLA7AX4**	2 (merged)

* **102 channels**

** **136 channels**

The term Master (M) module refers to the middle module of a 5-wide module chassis. The term Slave1 (S1) module refers to the module to the right of the Master module of a 5-wide module chassis. Figure 1–1 shows the configuration for a 2-wide module merge (x16-wide link only).

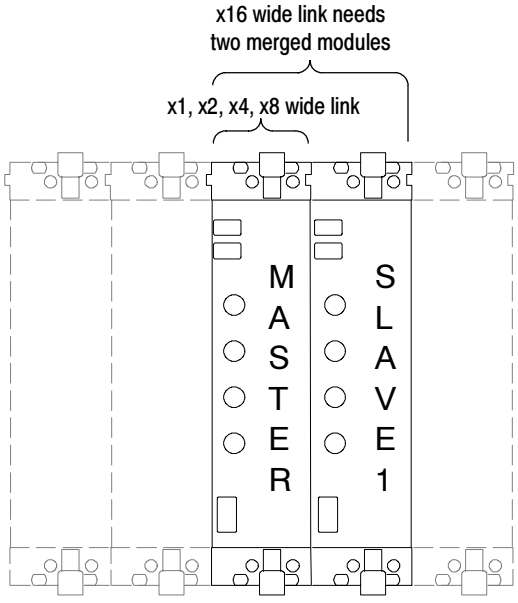

Figure 1-1: Configuration of the Master and Slave1 modules

LAI Cables

The LAI cables are specifically designed for use with the TMS817 and TMS818 probe adapter products. The link widths you choose require a specific number of LAI cables. For more detailed information on connecting your LAI cables to the preprocessor unit, see page 1–22.

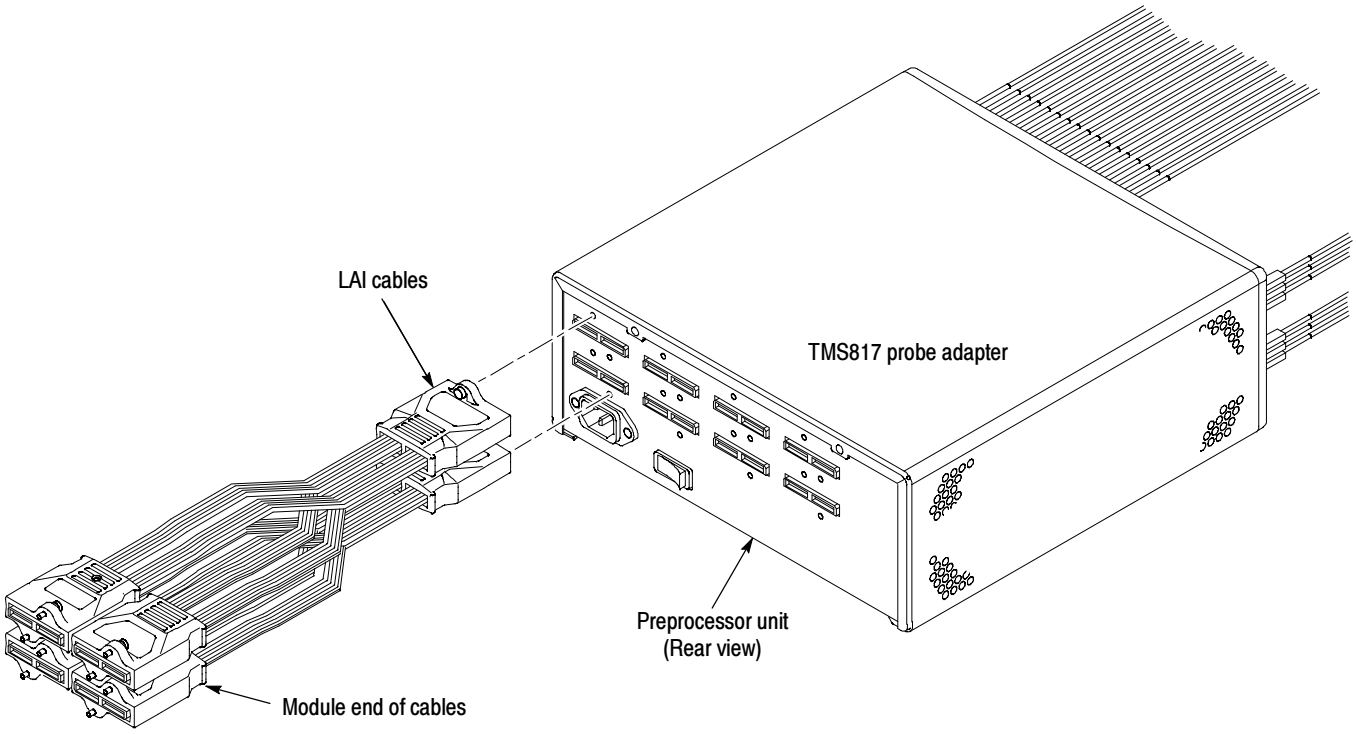

Figure 1-2: LAI Cables

Labeling LAI Cables

To apply labels to the LAI cables, see page 1–24.

Standard and Optional Accessories

A complete list of standard and optional accessories is provided in the *Replaceable Parts List* on page 5–3.

For the TMS817 product:

- The following options are available only when ordering a TMS817 product.

  Option 01 — midbus probe head

  Option 02 — x16 slot board

  Option 03 — x8 slot board

  Option 04 — LAI cables (4)

For an additional midbus probe head assembly with attaching parts order:

- TMSIC6 — midbus probe head for the TMS817

For the TMS818 product:

- The following options are available only when ordering a TMS818 product.

  Option 01 — midbus probe head

  Option 02 — x4 board

  Option 03 — x1 board

  Option 04— LAI cables (2)

For an additional midbus probe head assembly with attaching parts order:

- TMSIC8 — midbus probe head for TMS818

Probe Adapter Review

Review the electrical specifications beginning on page 3–1 as they relate to the target system, as well as the following descriptions of other product information.

System Data Rate

The probe adapters are designed to acquire data from a PCIExpress bus operating at 2.5 GT/s and have been tested at ±10% of the nominal frequency. These probe adapters are capable of acquiring spread spectrum data at 0.5 ppm at a rate of 33 KHz. An external clock is required for acquiring data at ±10% of 2.5 GT/s and for spread spectrum data.

Contact your Tektronix sales representative for current information on the fastest buses supported.

Nonintrusive Acquisition

The probe adapters do not modify or present signals back to the target system.

Storage Qualified Data

The disassembler is not designed to work with gaps in the acquisition data. Disassembly of storage qualified data can be indeterminate and incorrect.

Bus Width

The TMS817 and TMS818 products support bus link widths of x1, x2, x4, x,8, and x16. The TMS817 product supports bus link widths of x1, x2, and x4.

Linking of Requests to Completions

Linking of Requests to Completions across separate links and analysis of separate directions on the same link are not supported in the disassembler. Only timestamp correlation is available.

Lane Changes

On-the-fly lane reordering and link-width adjustments are not supported.

Detect Mechanism

The detect mechanism is not supported.

Triggering

Due to the logic analyzer trigger resource, triggering capabilities are different for different width links.

Training Packets

Training packets are captured and displayed as they are acquired from the link, but lane ordering and polarity settings need to be set manually.

Packet Payload

Only PCIExpress protocol tracking is performed. The disassembler does not perform decoding for any packet payloads.

Connect the Logic Analyzer to the Target System

Read the entire following section before beginning the installation procedure.

Tools

The following is a list of required tools:

- Flatbladed screwdriver (0.1 inch tip width) to adjust the Width or Mode switch
- POZIDRIV (PZ1) screwdriver to connect the clam shell housings on the preprocessor unit.
- **Optional:** A torque wrench helps to ensure reliable connections by meeting the nominal torque values that may be listed in these instructions. When attaching screws to the probe head use 4 in-lbs (0.451 Newton meters) of torque.

CAUTION. *To prevent static damage to the probe adapter, the LAI cables, and the module, handle components only in a static-free environment.*

Always wear a grounding wrist strap, heel strap, or similar device while handling a probe adapter.

Task Summary

Table 1–2 list the tasks that you must do to connect either a midbus probe adapter or a slot board to the logic analyzer.

Table 1-2: Task summary

Tasks	See page
Solder the retention fixtures and attach the interconnect strip to a midbus footprint on either the target system or the slot board.	1-8
Connect the midbus probe head to the midbus footprint.	1-9
(Slot board only) Connect the Slot board.	1-10
Connect the plugs on the midbus cables to the front of the preprocessor unit.	1-12
Connect External Clocking.	1-18
Adjust the preprocessor settings.	1-20
Connect the LAI cables between the preprocessor unit and logic analyzer.	1-23
Check the lane polarity and lane order.	1-27

Connecting the Midbus Probe Head

Follow these steps to connect the probe head to a midbus footprint:

1. Power off the preprocessor unit, if necessary. Refer to *Applying Power* and *Removing Power* on page 1–25.
2. Power off the target system. It is not necessary to power off the logic analyzer.

CAUTION. *To prevent static damage, handle these components only in a static-free environment. Static discharge can damage the probe adapter, the probes, and the logic analyzer module.*

Always wear a grounding wrist strap, heel strap, or similar device while handling the probe adapter.

3. To discharge the stored static electricity, touch the ground connector located on the back of the logic analyzer.
4. Make sure the target system is in a static-free environment.

CAUTION. *To prevent damage to the target board, soldering must be done by qualified service personnel.*

5. Locate the midbus footprint on the target system or slot board.
6. Remove the two retention fixtures from the hardware bag. Notice that one retention fixture is gold and is keyed, and the other is silver and is not keyed (see Figure 1–3 on page 1–8).
7. Correctly position one retention fixture on each side of the midbus footprint (see Figure 1–3 on page 1–8).

CAUTION. *To prevent resoldering a retention fixture, check that the retention fixture "nose" is pointing toward the midbus footprint as shown in Figure 1–3.*

8. Solder the four retention-fixture pins to the back of the circuit board.

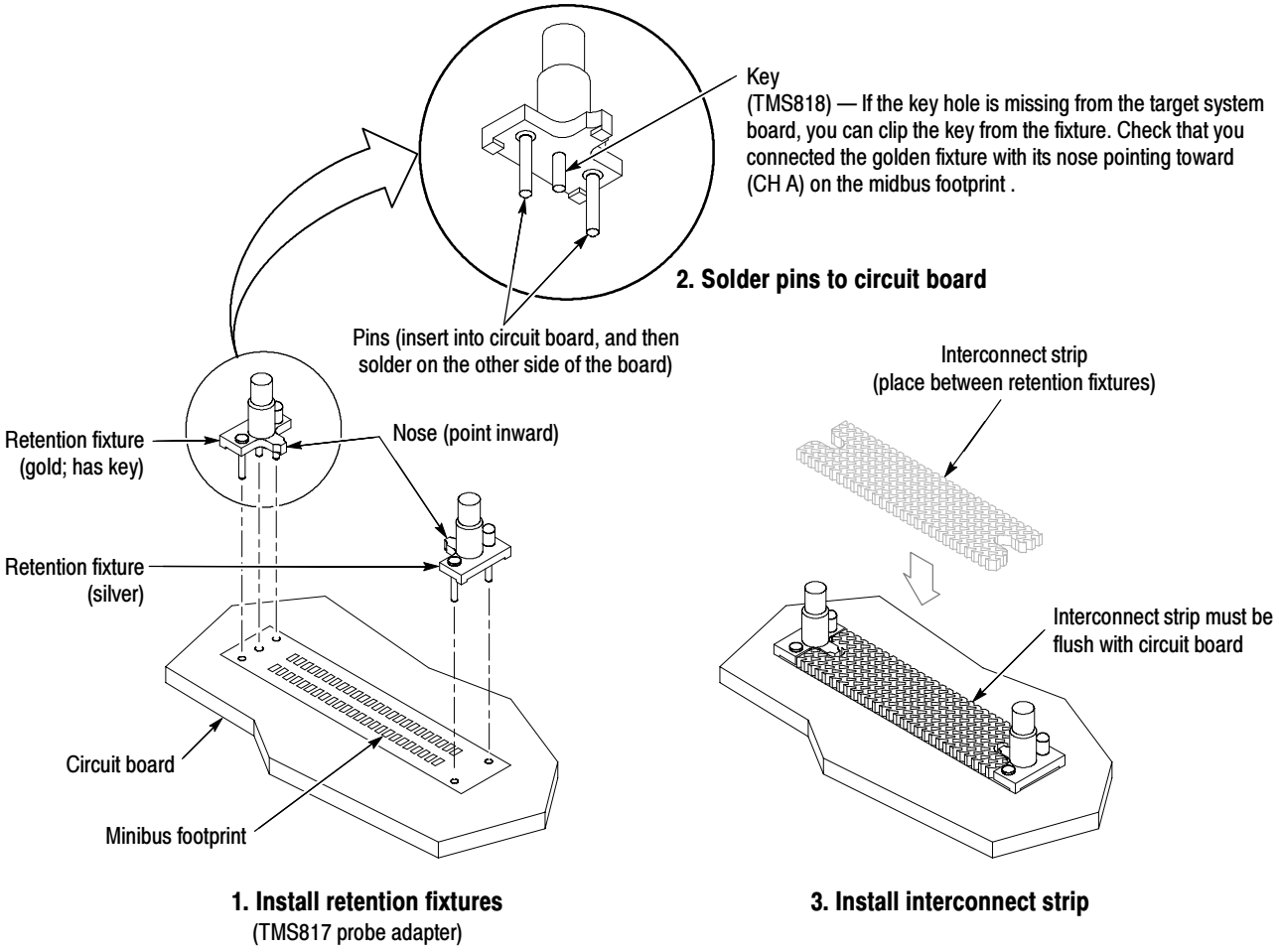

Figure 1-3: Connecting the retention fixtures

9. Remove the plastic interconnect strip from the hardware bag.

10. Place the interconnect strip on top of the midbus footprint and press into place.

NOTE*. The interconnect strip must lay flat against the circuit board; no gaps are allowed between the strip and the board.*

11. Remove the probe head from the protective packaging.

***NOTE**. When you attach the probe head, match the colors on either sides of the retention fixtures to the probe head. One side of the probe head is gold and the other is silver.*

12. Connect the probe head to the midbus footprint (see Figure 1–4).

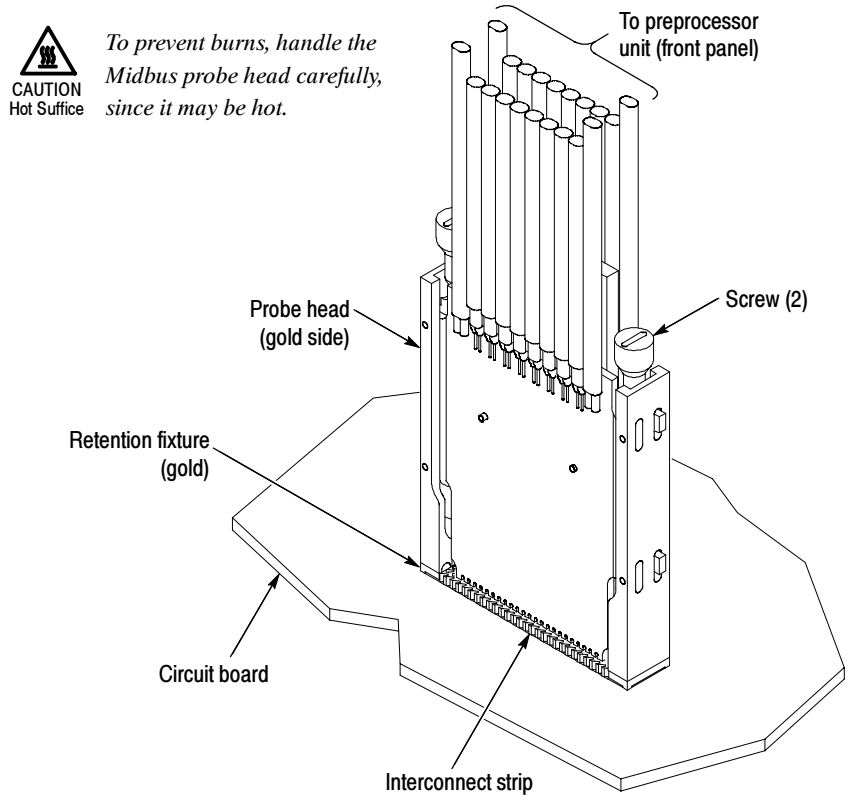

Figure 1- 4: Connecting the probe head to the retention fixtures

***CAUTION.** To prevent damage to the connector on the target system, always position the probe head perpendicular to the connector. Do not twist or bend the probe adapter while it is attached to the target system and support the probe-head cables in a way that keeps the mechanical forces on the connectors to a minimum.*

13. Tighten the screws (on both sides of the probe head) to the threaded studs on the retention fixtures.

Connecting the Slot Board

1. Remove the PCIExpress card from the target system.
2. Connect the midbus probe head to the slot board. See steps 5 through 12 starting on page 1–7.
3. Connect the PCIExpress card to the PCIExpress connect on the slot board.

   If you need more mechanical support for the PCIExpress card, attach the optional support bracket to the slot boar. Place the PCIExpress bracket on top of the Support bracket and attach it using the two supplied screws.

CAUTION. *To avoid damage to the PCIExpress connector, check that the PCIExpress bracket is on top of the support bracket.*

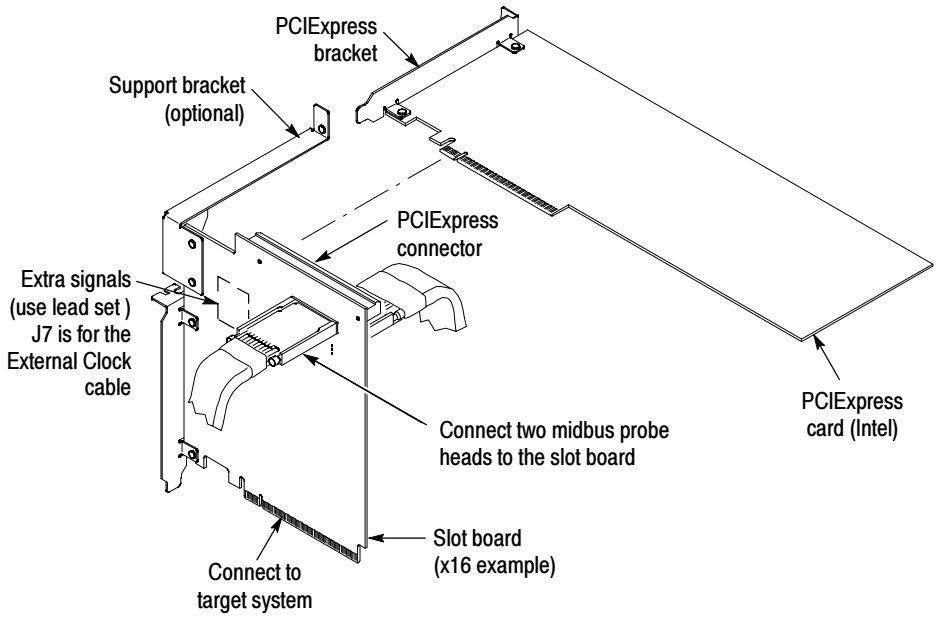

Figure 1-5: Connecting the Slot board and PCIExpress card

4. Connect the Slot board to your target system.

Extra Signals. The extra signals listed in Table 1–3 (except for J17) are not needed by the TMS817 and TMS818 support packages, but have been provided should you need to monitor them.

Use lead sets to connect the extra pins on the slot board to the logic analyzer. The extra signals need to be connected to a logic analyzer module separate from the modules that you are using for the TMS817 and TMS818 supports.

The slot board is labeled with an A and B side which lists these extra signal you may want to acquire. Table 1–3 lists the extra signals for each side of the slot board.

Table 1-3: Extra x16 link-width signals

Connector Pin	Signal name
J3-1	WAKE#
J3-2	GND
J4-1	PWRGD
J4-2	GND
J5-1	SMCLK
J5-2	GND
J6-1	SMDAT
J6-2	GND
J7-1*	REFCLK+
J7-2*	GND
J7-3*	REFCLK-

* **For Ext Clock cable**

5. Optional: Attach a tie-down strap(s) to the probe-head cables to stabilize the probe adapter while it is attached to the target system, if necessary.

Cable Configurations

Each preprocessor unit is capable of acquiring signals from a complete link. A complete link is two unidirectional links. The complete data link may be composed of 1, 2, 4, 8 or 16 channels (unidirectional lanes). Following are two examples of preprocessor-unit configurations.

Figure 1–6 shows an example of a preprocessor-unit configuration for two x8-width links.

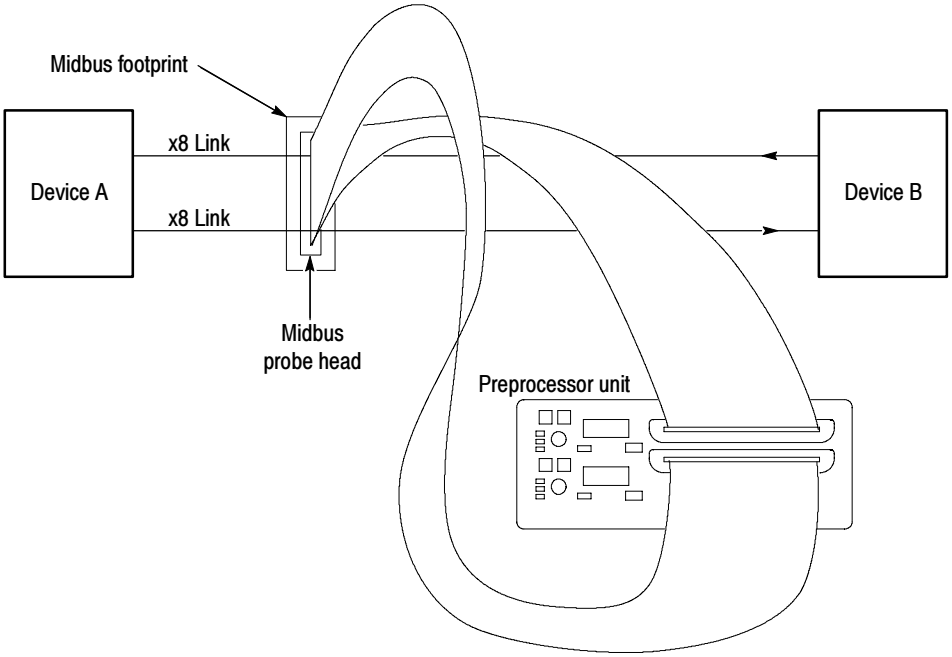

Figure 1- 6: Preprocessor unit configuration for two x8-wide unidirectional links

Figure 1–7 shows an example of a preprocessor-unit configuration for four unidirectional x4-width links (two x4 width links).

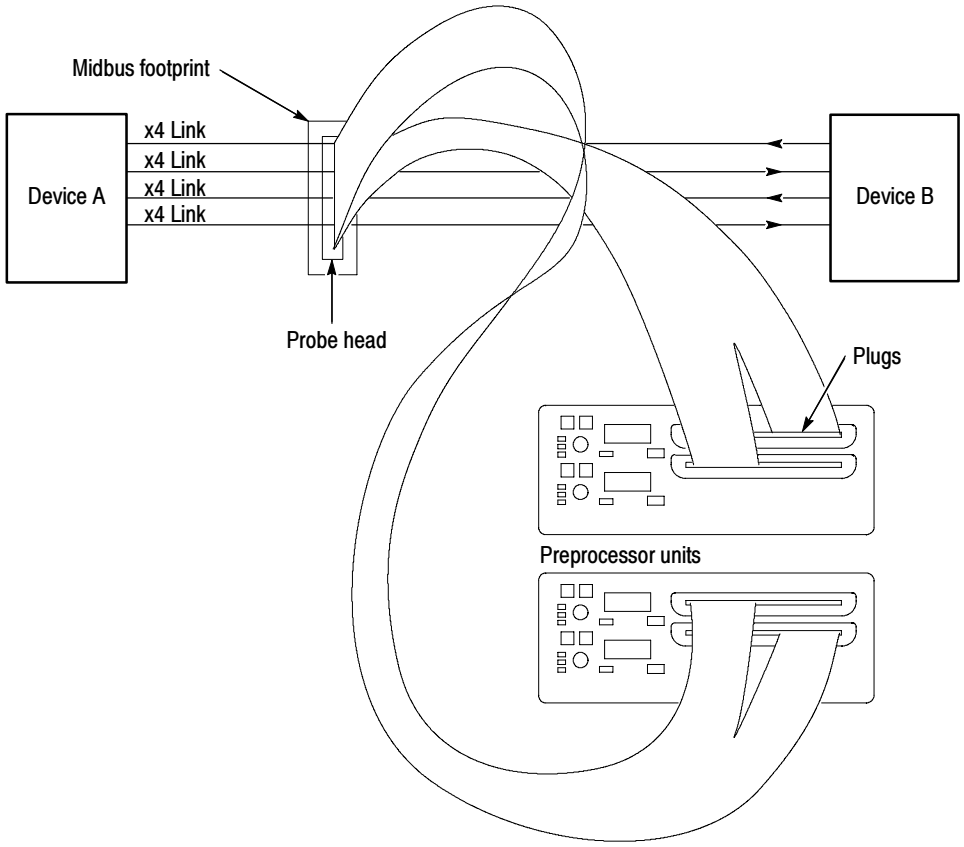

Figure 1-7: Preprocessor unit configuration for four x4-wide links

Create a Configuration Plan. Before connecting the probe-head plugs to the preprocessor unit, you must complete the following steps:

1. Determine the number of complete links (see examples of different link configurations on pages 1–12 and 1–13).
2. Assign the particular lanes that make up the complete links. These links go through specific pins on the midbus footprint.

***NOTE**. The recommended midbus footprint for the target system is shown in Figure 1–8. If the midbus footprint is different than Figure 1–8, you may need to create a cross reference for your configuration.*

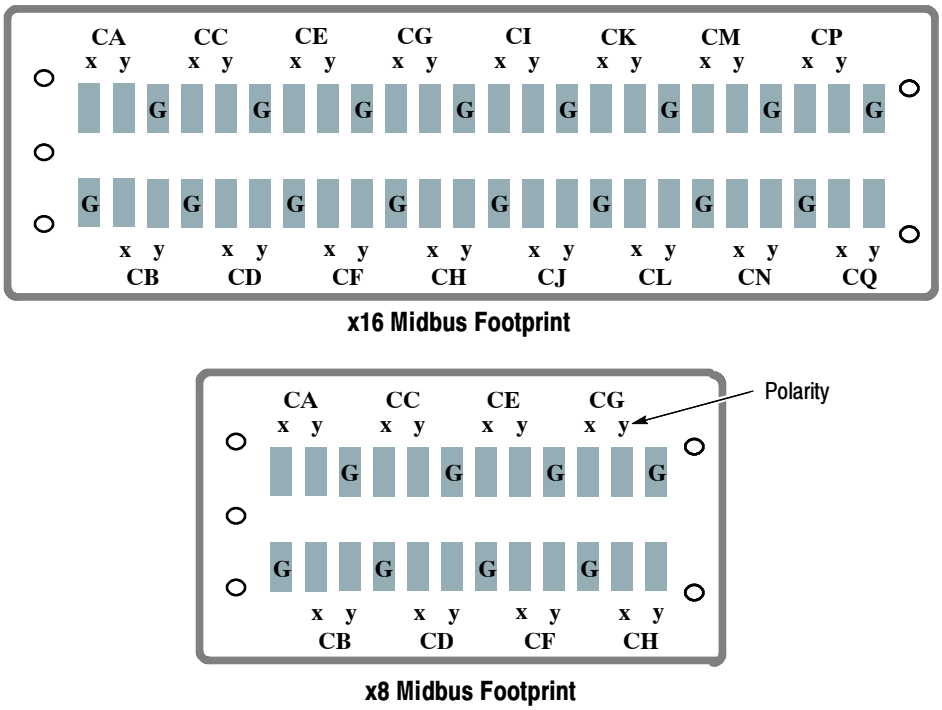

Figure 1-8: Preferred midbus footprint configurations

3. You must choose which upstream or downstream connector on the preprocessor unit your link attaches to (see Figure 1–9).

   The upstream and downstream connectors are located on the front of the preprocessor unit and it does not matter which connector you choose. The labels upstream and downstream do not refer to signal direction. These terms are used only to differentiate between the two connectors.

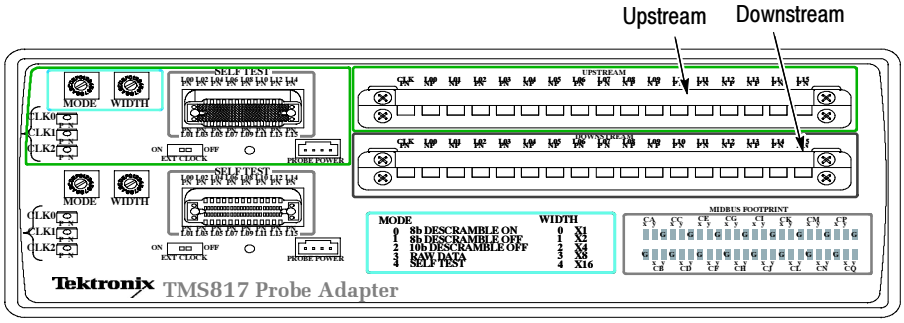

Figure 1-9: Preprocessor unit (front)

4. You must choose which side of the plug is positive or negative (X or Y) on the probe-head cable (see Figure 1–10 on page 1–16 for the location of the polarity references on the plug).

Connecting the Probe-Head Plugs. Use the following procedure to connect the probe head plugs to the front of the preprocessor unit:

NOTE. *All probe-head plugs must be connected to the preprocessor unit, regardless of the configuration. For example, if your configuration has four lanes, then L00–L03 have active channels, and L04–L15 have all inactive channel. All probe-head plugs will be connected. For the inactive channels, the order and polarity is not important.*

1. Power off the preprocessor unit. Refer to *Applying Power* and *Removing Power* on page 1–25. It is not necessary to power off the target system or the logic analyzer.
2. Attach the appropriate probe head cable plugs to the front of the preprocessor unit (see Figure 1–10 on page 1–16).

NOTE. *The name of the pad on the midbus footprint is printed on each probe-head plug for ease of use (see Figure 1–10 on page 1–16).*

3. Connect the probe head power plug(s) to the front of the preprocessor unit (see Figure 1–12 on page 1–20).

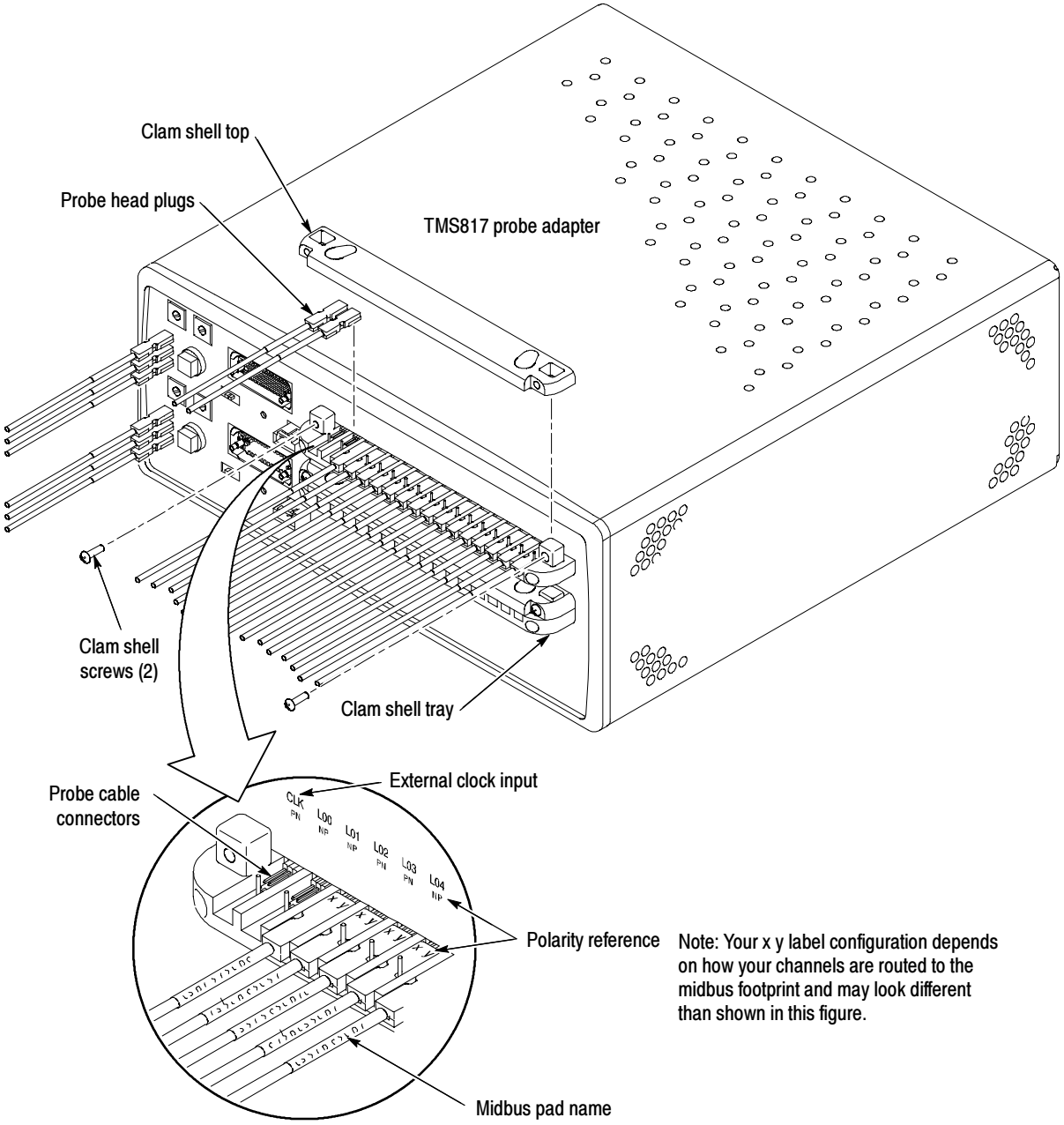

Figure 1- 10: Attaching the probe-head cables

What Next

Read the information on *External Clockin*g on page 1–18. For most situations you will need to use the External clock cable.

After completing the external clocking section, you are ready to:

1. Make the adjustments to the front of the preprocessor unit (see page 1–20)
2. Check the polarity and order of the lanes (see page 1–27)
3. Attach the LAI cables (see page 1–22)

***NOTE**. If you have trouble acquiring data from the target system after you complete these procedures, use the* Self Test *procedure on page 1–29 to ensure that the preprocessor unit is working correctly. Also, recheck your lane polarity and lane ordering.*

External Clocking

External clocking is the recommended mode of operation due to the frequency difference between the preprocessor unit and the target system. External clocking is required if the following two conditions apply:

- The target system is statically exceeding the ±100 ppm data-rate variation from 2.5 GT/s; for example, frequency margining.
- The target system is dynamically exceeding the ±100 ppm date-rate variation from 2.5 GT/s; for example, spread-spectrum clocking.

For both of the above conditions, you need to attach the included external clock input cable to the 100 MHz clock on the target system (see Figure 1–11).

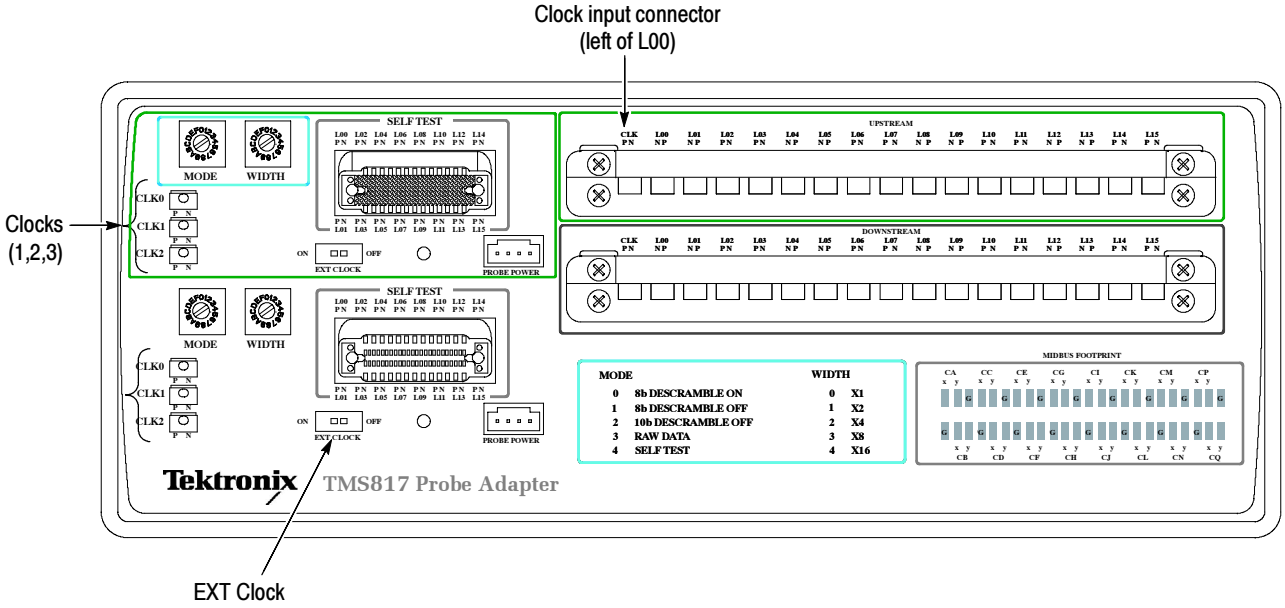

Figure 1-11: External clocking connections

Use the following procedure to connect the external clock input cable (the clock polarity does not matter):

***NOTE**. We recommend separate clock connection from the target system to the preprocessor unit for each unidirectional link.*

1. Power off both the probe adapter and the target system.

2. On the target system, attach the included external clock input cable to the three-pin 100 MHz (nominal) clock connector. The female end of the external clock input cable attaches to the male connector on the target system. The external clock polarity is not important.
3. On the front of the preprocessor unit, attach the male end of the external clock input cable to the CLK input connector (see Figure 1–11).
4. On the front of the preprocessor unit, set the EXT CLK switch to ON (see Figure 1–11).
5. If you are using both upstream and downstream portions of the preprocessor unit you can:
   - (Recommended) Use a separate clock input cable for each unidirectional link.
   - Attach another clock cable between the CLK (0, 1, 2) connector and the CLK input connector (to the left of the L00 connection) on the other portion of the preprocessor unit. Set the EXT Clock switch to ON.

   The CLK (0, 1, 2) connector is replicated from the CLK input connector.

   If you have multiple preprocessor units requiring the 100 MHz clock, then you can interconnect the preprocessor units with the provided clock cables, and use the above steps 3 through 5.

Use the following procedure to disconnect the external clock input:

1. Power off the probe adapter and the target system.
2. Disconnect all clock cables that connect the target system to the preprocessor unit, and disconnect the preprocessor-to-preprocessor cables. Retain these cables for future use.
3. Set all EXT CLK selector switches to OFF.

Adjust the Preprocessor Unit Settings

Use Figure 1-12 and Table 1-4 on page 1-20 to set up the preprocessor unit.

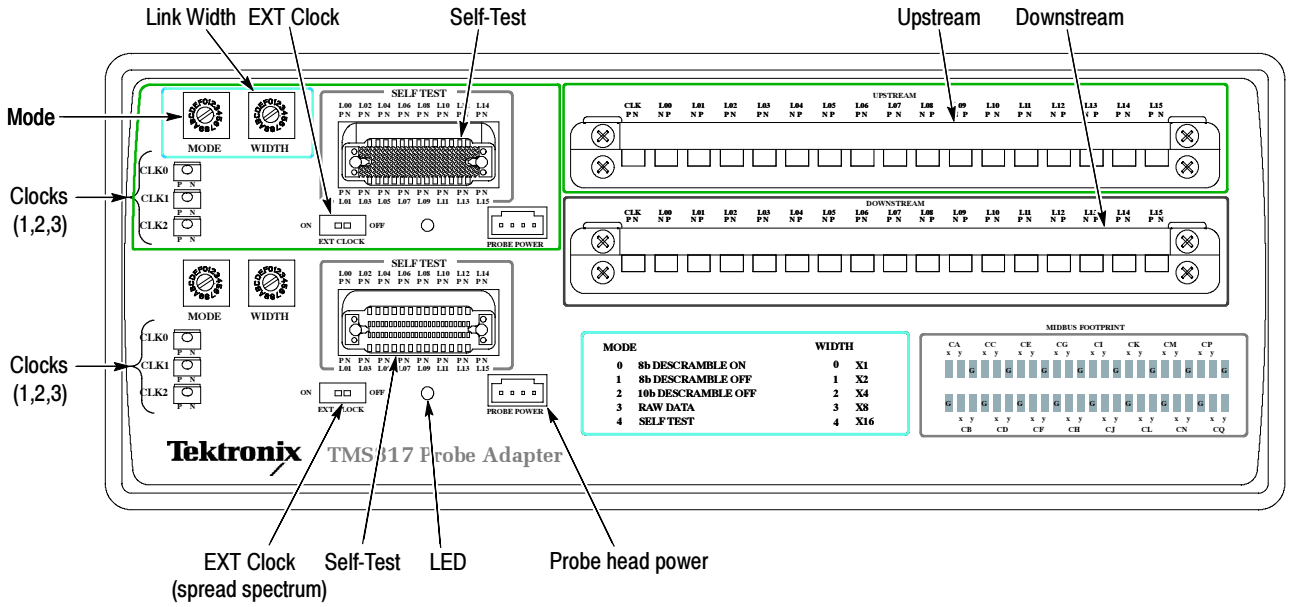

Figure 1-12: Adjust the preprocessor unit settings

CAUTION. *To prevent damage to the preprocessor unit, you must power off the preprocessor unit before changing any settings.*

Table 1-4: Preprocessor unit settings

Name	TMS817 settings and connections	TMS818 settings and connections	Description
Mode (rotary switch with multiple positions)	0 1.0a — descramble on 1 8b — descramble off 2 10b — descramble off 3 Raw data 4 Self Test 5 1.0 — descramble on	Same	Use the mode switch to select the type of data you are acquiring.

Table 1-4: Preprocessor unit settings (Cont.)

Name	TMS817 settings and connections	TMS818 settings and connections	Description
Self Test (midbus footprint) **Caution**, static sensitive.	Connect the probe head to the Self-Test connector on the front of the preprocessor unit where you are running the self test. Set the mode switch to Self Test (position 4).	Same	Use the Self Test connection to test that the preprocessor unit is operating properly.
Link Width (Rotatory switch with multiple positions) **Note:** For self test this switch has different settings. Refer to Table 1–8 on page 1–31.	0 x1 1 x2 2 x4 3 x8 4 x16	0 x1 1 x2 2 x4	Use the link width setting to specify the width of your link. (Use a flatbladed screwdriver to make the adjustments.)
Ext clock (Spread Spectrum slide switch)	On/off	Same	Use the spread spectrum clocking switch to specify if it is enabled on the target system.
Clocks (connector)	Three clock connections are available.	Same	If needed, use when the external clock and target system does not provide a separate clock for each unidirectional link.
Upstream (connector) Downstream (connector)	Connect the appropriate probe-head plugs to one of the connectors on the front of the preprocessor unit.	Same	Use to acquire bus information. Before connecting, you must determine which connector, upstream or downstream, to connect to the front of the preprocessor unit.

Connect the LAI Cables

The LAI cables connect the logic analyzer module(s) to the back of the preprocessor unit (see Figure 1–13). Before you connect the LAI cables, you need to determine the number of cables and then apply labels.

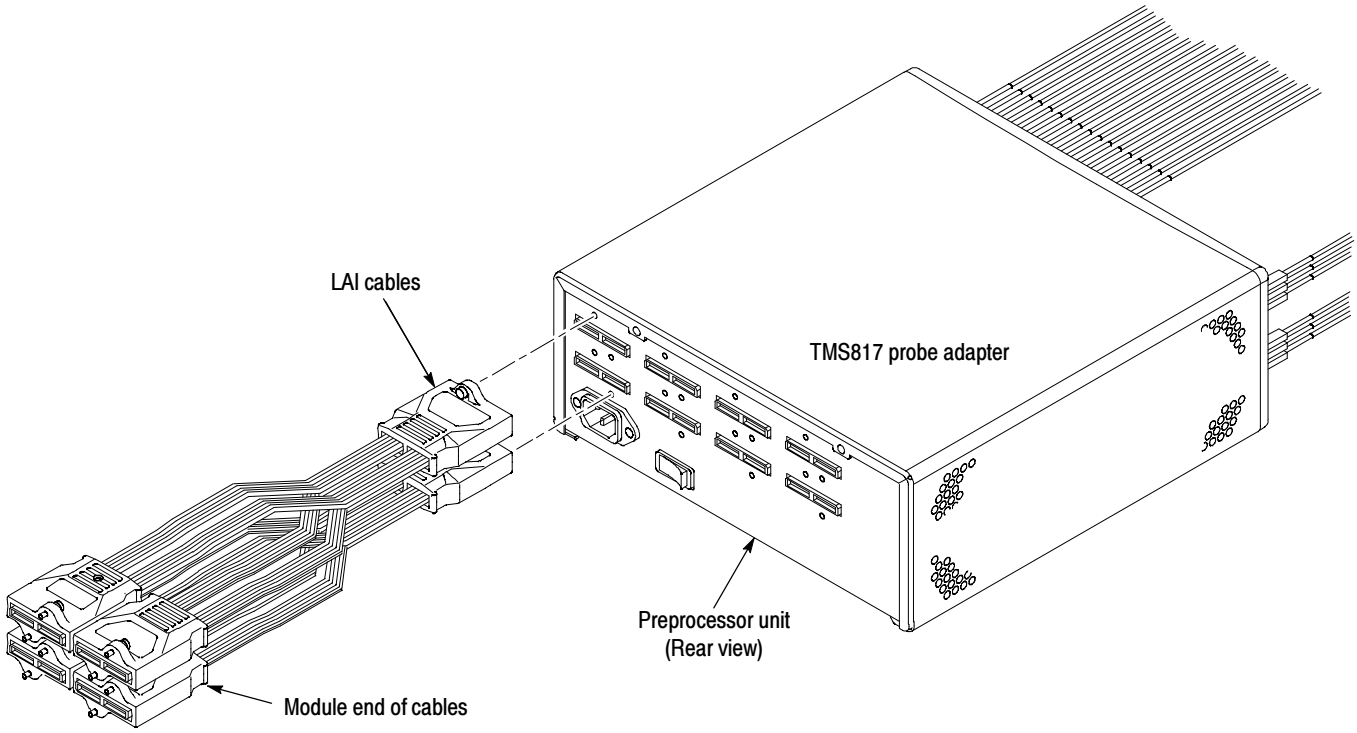

Figure 1-13: LAI cable and preprocessor unit

CAUTION. *To prevent static damage to the probe adapter, LAI cables, probes, and the module, handle components only in a static-free environment.*

Always wear a grounding wrist strap, heel strap, or similar device while handling probe adapter.

Number of LAI Cables. The link widths that you choose require a specific number of LAI cables. To determine the number of cables you need, see Table 1–5.

For example, a x1 unidirectional link and a x16 unidirectional link require six LAI cables, or two cables for x1 and four cables for x16 connections.

Table 1-5: LAI cable quantities

Link widths	LAI cable quantity
x1 unidirectional upstream	2
x1 bidirectional up/downstream	4
x2 unidirectional upstream	2
x2 bidirectional up/downstream	4
x4 unidirectional upstream	2
x4 bidirectional up/downstream	4
x8 unidirectional upstream	2
x8 bidirectional up/downstream	4
x16 unidirectional upstream	4
x16 bidirectional up/downstream	8

Figure 1–14 shows the back of the preprocessor unit.

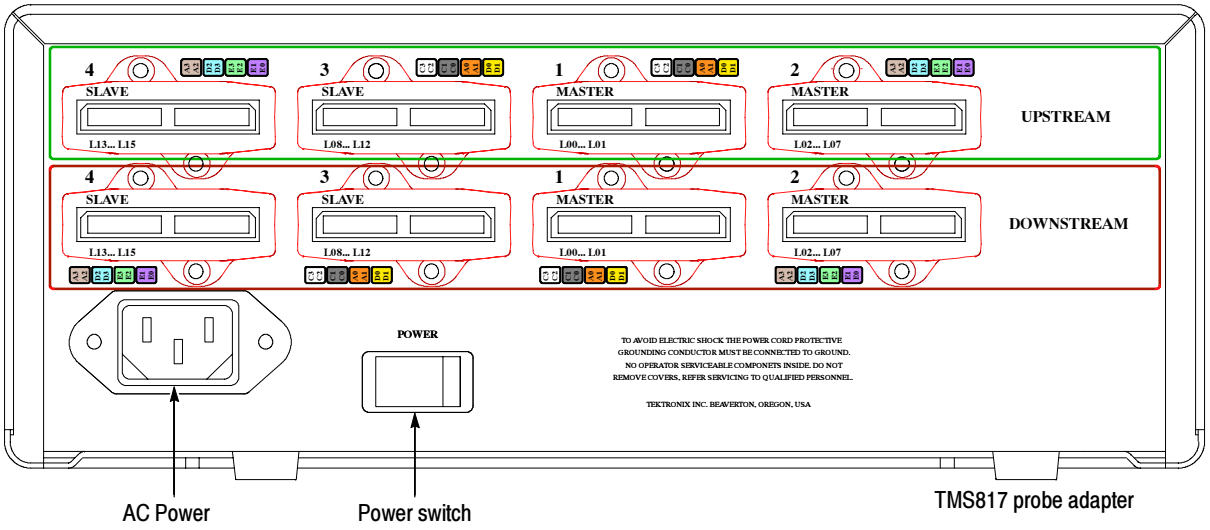

Figure 1-14: Preprocessor unit (back)

Applying Labels. You need to attach labels to the module end and the preprocessor-unit end of the LAI cables. Read the following note before you begin attaching labels.

NOTE. *Always use flat-nosed tweezers to remove the labels from the sheet of labels. Never peel labels with your fingers. The labels are made of soft vinyl and can stretch and distort easily. To avoid stretching the label, always grasp it from the top right corner while removing it from the sheet of labels.*

The adhesive on the vinyl labels is extremely strong. Carefully align the label to the indented outline on the module end and preprocessor unit end. Once labels are placed on the LAI Cables, they become very difficult to remove.

To attach labels, perform the following steps:

1. Determine which channel groups you are planning to use, and identify the matching labels.
2. Follow the steps in Figure 1–15 while attaching the labels.

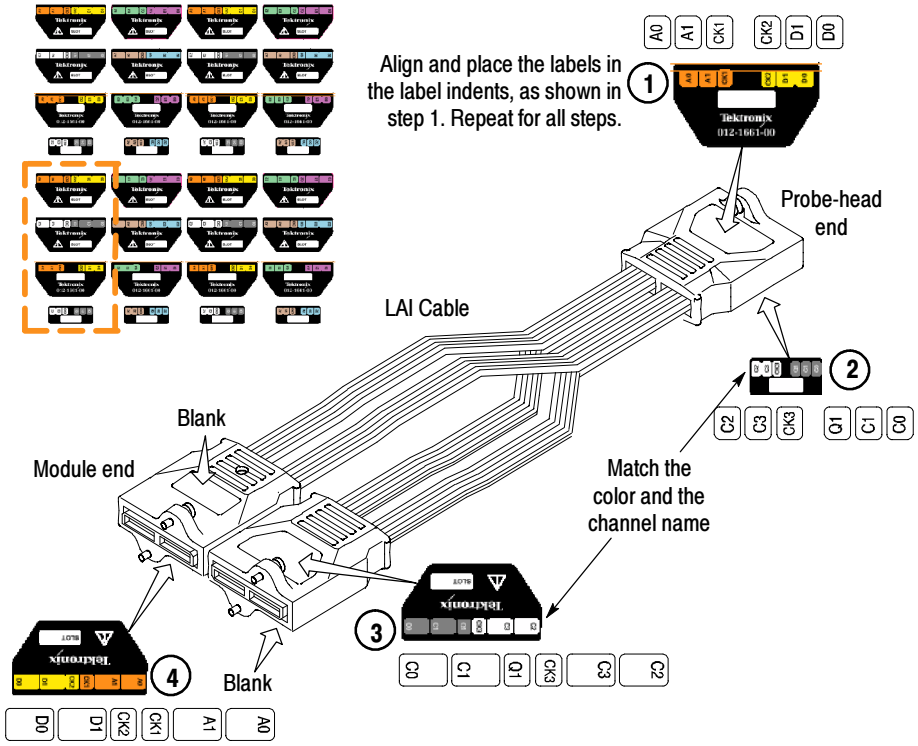

Figure 1-15: Apply LAI labels

LAI Cable Configuration. Following is the minimum configuration for disassembly support.

1. Match the A, D, C, and E LAI cables from the Master module with the corresponding D3/D2 and A3/A2, D1/D0 and A1/A0, C1/C0 and C3/C2, and E3/E2, and E1/E2 LAI connector labels on the preprocessor unit. Connect the LAI cables.

   Do not connect the E1/E2, E3/E2 connector if you are using an 102 channel module (for x1, x2, or x4 link widths).

2. (x16-wide link only) Repeat step 1 to make LAI cable connections between the Slave module and the preprocessor unit.

Figure 1–16 shows the configuration for the Master and Slave1 modules.

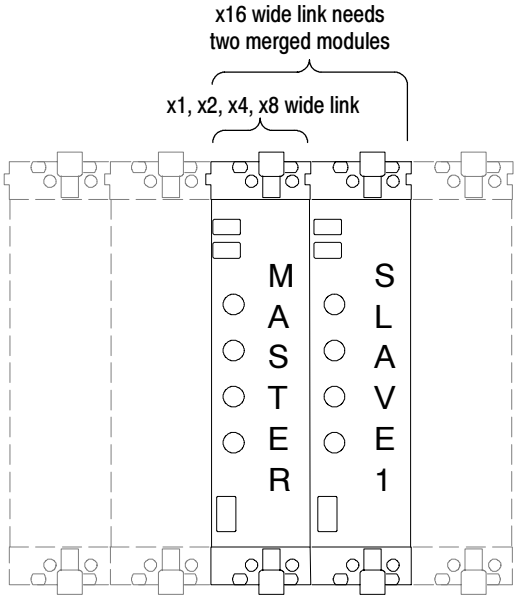

Figure 1-16: Configuration of the Master and Slave1 modules

Applying Power

To apply power to the probe adapter and target system, follow these steps:

WARNING. *To prevent personal injury or damage to the preprocessor unit, there are no operator serviceable parts inside the cover of the preprocessor unit. Refer servicing of parts in the preprocessor unit to Tektronix authorized personnel only.*

1. Make sure the power switch on the preprocessor unit is in the off position. The (zero) is depressed on the power switch, which is located on the rear panel of the preprocessor unit.
2. Plug the AC power cord into the IEC connector on the back of the preprocessor unit.
3. Plug the AC power cord into an electrical outlet that you know is working properly.
4. Power on the preprocessor unit. Two green LEDs light on the front of the preprocessor unit, indicating that the probe adapter is active.
5. Power on the target system.

Removing Power

To remove power from the target system and the probe adapter, follow these steps:

1. Power off the target system.
2. Power off the probe adapter at the back of the preprocessor unit.

Capture Training Sequence

For the logic analyzer to receive coherent data you must have the correct lane polarity and order. To determine the correct lane polarity and order, you need to load the training sequence EasyTrigger.

The training sequence EasyTrigger detects TS1/TS2 training packets. These packets, when displayed, show the polarity and order of the lanes (see Figure 1–17).

Follow these steps to verify the lane polarity and order:

***NOTE**. Always correct the lane polarity first and then the lane position. If you correct the lane position first, the polarity information will be incorrect.*

1. If you have not loaded the software support package, load it now (refer to page 2–3). The EasyTriggers are loaded when the software support is loaded.
2. In the EasyTrigger window, select **Trig_on_Training_Sequence** (refer to page 2–3).
3. On the logic analyzer, select **Run.**
4. On your target system, initiate a TS1/TS2 sequence.
5. After the logic analyzer triggers, scroll down to the last TS1/TS2 sequence (see Figure 1–17).

   Scrolling to the last TS1/TS2 sequence ensures that you are looking at the final polarity and lane order.
6. Check the last ten polarity symbols in each lane (see Figure 1–17). These symbols must be 45 or 4A indicating correct polarity.

   If any lane has a BA or a B5 polarity symbol, the polarity of that lane needs to be inverted. To invert the polarity of a lane:

   a. Unplug the lane from the preprocessor unit

   b. Invert the plug

   c. Reconnect the plug

7. After all the lane polarities are corrected, initiate a TS1/TS2 sequence on the target system and select RUN. After logic analyzer triggers, scroll down to the last TS1TS2 sequence. Check that there are no BA or B5 in the lane and that the lane ordering in the Link Details column is correct (see Figure 1–17).

   The lane ordering must be in an ascending order starting from lane 00. If a lane is not in the correct order, then unplug that lane from the preprocessor unit and plug it into the correct lane, taking care that the polarity remains the same.

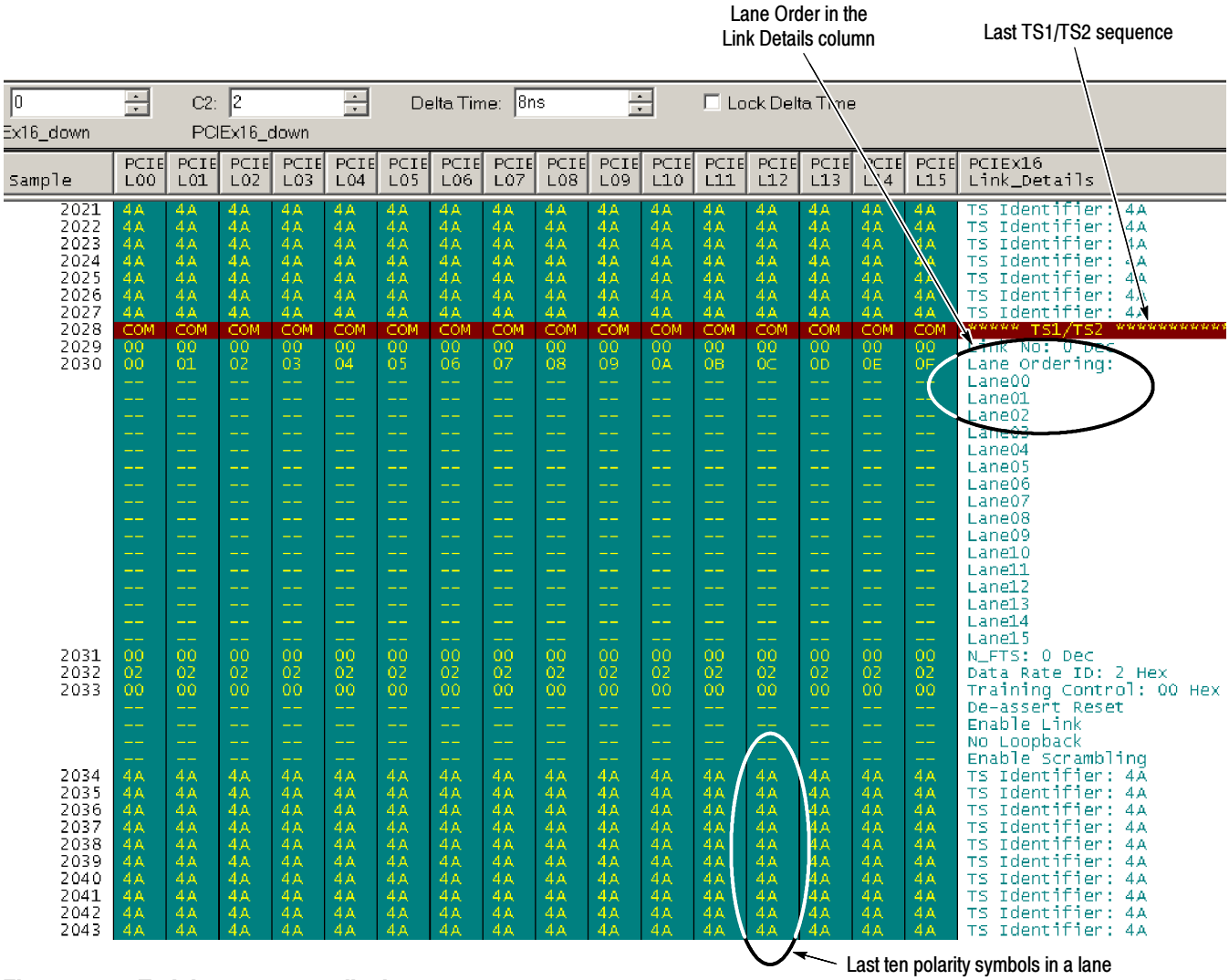

Figure 1-17: Training sequence display

Troubleshooting

If you cannot acquire data from the target system using the probe adapter, use the following self-test procedure to ensure that the preprocessor unit is working correctly.

Also, recheck your lane polarity and lane ordering (see *Load Training Sequence* on page 1–27).

Use the Trouble shooting check list on page 1–36 to check that the probe adapter is set up correctly.

WARNING. *To prevent personal injury or damage to the preprocessor unit, there are no operator serviceable parts inside the cover of the preprocessor unit. Refer servicing of internal parts in the preprocessor unit to Tektronix authorized personnel only.*

Self Test

Before you begin the self test example, check that power is supplied to the preprocessor unit by observing the lighted LED on the front of the case. If the LED is not lighted:

- Check that the power switch on the back of the preprocessor unit is powered on. If powered off, a 1 (one) is depressed on the switch.
- Check that the AC power cord is plugged into an electrical outlet that you know is working properly.
- If the LED is still not lighted, unplug the unit from the electrical outlet and call a Tektronix Service representative.

NOTE. *We recommend that you save the system. This ensures that system properties are retained for future use. However, care must be taken while restoring the system to exactly the same system as previously restored. Any changes (hardware or software) to the system causes the saved system not to restore. Saving the system should only be done on stable systems.*

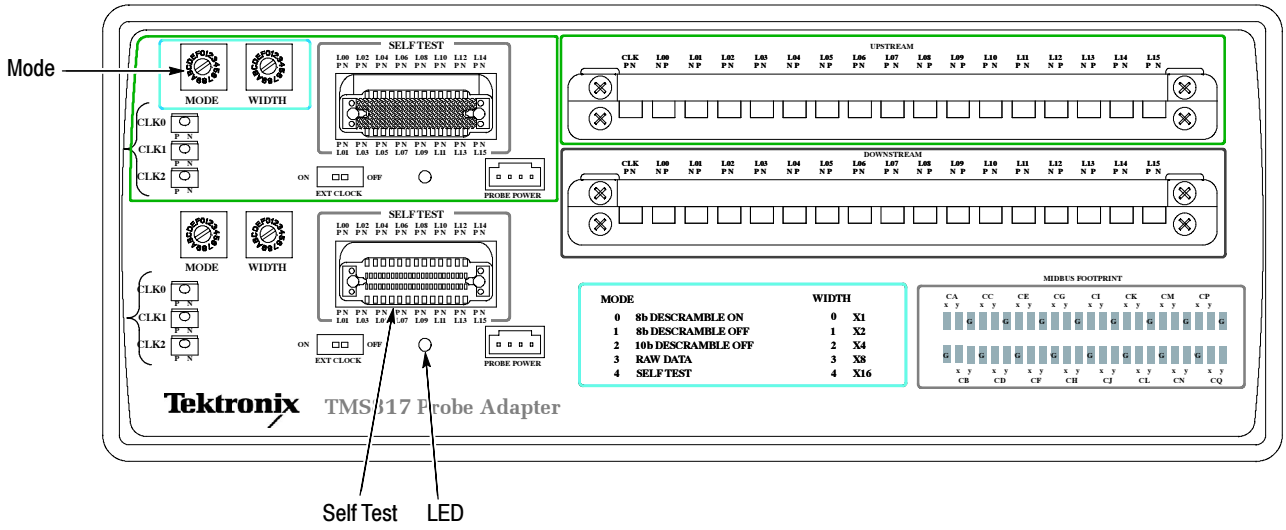

Figure 1-18: Self test connector

Follow these steps to run the self test:

1. Connect the midbus probe head to the self test midbus footprint on the preprocessor unit. Check that you also connected the interconnect strip correctly (see Figure 1–4 on page 1–9).

2. Connect the probe-head plugs to the front of the preprocessor unit as listed in Table 1–6. Connect the power connector to the front of the preprocessor unit.

Table 1-6: Probe-head channel connections

Probe head channel name	Preprocessor unit channel name	Probe head polarity
Channel A	L 00	X = N
Channel B	L 01	X = N
Channel C	L 02	X = N
Channel D	L 03	X = N
Channel E	L 04	X = N
Channel F	L 05	X = N
Channel G	L 06	X = N
Channel H	L 07	X = N
Channel I	L 08	X = P
Channel J	L 09	X = P
Channel K	L 10	X = P
Channel L	L 11	X = P

Table 1-6: Probe-head channel connections (Cont.)

Probe head channel name	Preprocessor unit channel name	Probe head polarity
Channel M	L 12	X = P
Channel N	L 13	X = P
Channel P	L 14	X = P
Channel Q	L 15	X = P

3. Choose a support package that matches your link width using Table 1–7.

Table 1-7: Test package load options

Support package	Lane width
PCIEx421_test	x1, x2, and x4
PCIEx8_test	x8
PCIEx16a_test	x16
PCIEx16b_test	x16
PCIEx16c_test	x16

***NOTE**. To test a complete x16 unidirectional link, all three of the x16 support packages need to be used.*

4. Set the front panel switches as listed in Table 1–8.

Table 1-8: Front panel switch settings

Name	Position
Link width switch (for self test)	Position 2 for x1,x2,and x4
	Position 3 for x8
	Position 4 for x16 (part a)
	Position 5 for x16 (part b)
	Position 3 for x16 (part c)
Mode switch	Set to switch position 4
External clocking switch	Set to OFF

5. Load the appropriate test support package on the module. For example, if testing a x8 link, load the PCIEx8_test support package. Table 1–9 lists module configurations for each test support package:

Table 1-9: Type and number of module for test package

Support package	Module
PCIEx421_test	One unmerged, 102- or 136 channel module
PCIEx8_test	One unmerged, 136 channel module
PCIEx16a_test	Two merged, 136 channel modules
PCIEx16b_test	Two merged, 136 channel modules
PCIEx16c_test	Two merged, 136 channel modules

6. Power off all modules other than the module that has the PCIEx8_test support package loaded on it.

***NOTE**. PCIEx8_test needs one unmerged 136 channel module.*

   a. Click on System, and then select System Trigger.

   b. Enable “System triggered by this module”, and select the module on which the test support package is loaded.

   c. Click OK.

7. Click on the **Setup** button, and then click on Define Compare.

   a. In the compare window, enable “Enable Data Compare”.

   b. Click on Add Data Source and browse to: C:\Program Files\TLA700\Supports\PCIEx8_Test\PCIEx8_Comparison-Data.tla (example for a x8 test)

   c. Click OK, and close the compare window.

***NOTE**. For the PCIEx421_test support package there are two comparison data files. Use the PCIEx421_102ch_ComparisonData.tla file for an 102 channel module and the PCIEx421_136ch_ComparisonData.tla file for an 136 channel module.*

8. Click on Setup and change the memory depth to 1024.

9. In the Setup window, click on Channel Compare in the Table Shows box, and then:

   a. Click on the check mark beside the clk channel. This disables the clock signal from being compared during self test.

   b. Minimize the Setup window.

10. Click on System, and then select Repetitive Properties in the pull down menu.

    a. Enable “Stop if compare with reference is not equal”.

    b. Click OK.

11. Add a new data listing window for the PCIEx8_Test module.

12. Right click on a group name in the Listing Window and select Properties.

    a. Click on the Listing Window tab.

    b. Enable Acq != Ref, and select the color red. Enable Acq = Ref, and select the color green. Click OK.

13. Click on System, and enable Repetitive.

14. Click on the **Status** button and use the scroll bar to scroll to the PCIEx8_test module.

15. In the logic analyzer system command bar, press the **Run** button.

16. In the Status window, look at Data Compare. If you see Data Compare:Equal it means that the preprocessor unit is working correctly and the self test has finished successfully. Press the **Stop** button.

    If the system stops by itself, you will see DataCompare:Not Equal. This means that there are bad channels and you will need to check the listing window for these bad channels. All bad data is displayed in red.

For TMS818. To test all eight channels of the probe head:

- Test Channel A, Channel B, Channel C, and Channel D using the procedure above. To test Channel E, Channel F, Channel G and Channel H connect to the preprocessor unit as shown in Table 1–10:

Table 1-10: Probe adapter connections for channel E, F, G, and H

Probe head channel name	Preprocessor unit channel name	Probe head Polarity
Channel E	L 00	X = N
Channel F	L 01	X = N
Channel G	L 02	X = N
Channel H	L 03	X = N

- Follow the Self Test procedure for a X4 link.

Test Characteristics. Following is a list of test characteristics:

- The system stops as soon as acquired data does not match the reference data. Acquired data can be seen in the listing window; incorrect data is red and correct data is green.
- If the system is manually stopped while it is running, errors are seen in the listing window because the trigger is not aligned to the trigger in the comparison data.
- The self test is effective at finding stuck bits, but it will not stop if one or more of the incoming lanes are dead. If there is a dead lane, 17C is shown on all lanes when manually stopped.
  - Power to the probe head is not connected. If this is the case, connect the power to the power connector on the probe head.
  - One or more of the incoming serial channels are dead.

To find out which channel is dead:

1. Load the appropriate support package on the module (in this example load the PCIEx8 support package), and change the mode switch setting to the appropriate position.
2. Attach the probe head to the target system (see page 1–4).
3. Take an acquisition.

If one or more lanes show 1BC in the listing window that means that those lanes are dead.

4. To verify that the bad channels are on the probe head and not the preprocessor unit, swap the bad lane(s) with a good lane(s).

   If the good lanes show 1BC, either the probe head channels are dead or the channels on the target system are dead. If you received 17C during the self test, the channel on the probe head is dead.

If performing the preceding steps did not reveal the problem, call a Tektronix sales representative.

Check list for Troubleshooting

Table 1-11: Troubleshooting checklist

✓	Description	Notes
	Check that the lane polarity and lane ordering is correct. See page 1-27.	
	Check that the interconnect strip is connected. See page 1-8.	
	Check that the midbus retention mechanisms are soldered correctly. See page 1-8.	
	Check that the power cable for the probe head is connected to the probe power on the front of the preprocessor unit. See Figure 1-12.	
	Check that the mode and the link width switches are set to the appropriate settings. See page 1-20.	
	Check that the correct software support package is loaded.	
	Check that the logic analyzer modules are the correct modules and that they have the maximum frequency setting. See page 1-2.	
	Check that the Self Test runs. See page 1-29.	
	Check that the external clock cable is connected. See Figure 1-11.	
	Check that the data eye of the target system meets the minimum requirements. See page 3-3.	
	Check that the external clock signal meets minimum requirements. See page 3-3.	

Care and Maintenance

Before cleaning this product, read the following information.

WARNING. *To prevent personal injury or damage to the preprocessor unit, refer servicing of internal parts in the preprocessor unit to Tektronix authorized personnel only. There are no user-serviceable parts inside the cover of the preprocessor unit.*

CAUTION. *Static discharge can damage the probe adapter, the probes, and the module. To prevent static damage, you must handle components only in a static-free environment.*

The probe adapter, consisting of the probe head and preprocessor unit, does not require scheduled or periodic maintenance. However, to keep good electrical contact and efficient heat dissipation, keep the probe adapter free of dirt, dust, and contaminants. When not in use, store the probe adapter in the original shipping bags and cardboard carton.

External Cleaning Only

Clean dirt and dust with a soft bristle brush. For more extensive cleaning, use only a damp cloth moistened with deionized water; do not use any other chemical cleaning agents.

WARNING. *To prevent harm to yourself or damage to the preprocessor unit, do not open the preprocessor unit for cleaning or allow any moisture inside the unit. Refer servicing of internal parts in the preprocessor unit to Tektronix authorized personnel only. External parts may be replaced by qualified service personnel.*

Ship the Probe Adapter

To commercially transport the TMS817 and TMS818 probe adapters, package as follows:

1. Use the existing cardboard shipping carton and cushioning material.

   If the existing shipping carton is not available, use a double-walled, corrugated cardboard shipping carton that allows a 3 inch (7.62 cm) minimum on all sides of the product.

2. If you are shipping a probe adapter to a Tektronix service center for Warranty service, attach a tag to the probe adapter showing the following:

   - Owner's name and address
   - Name of a person who can be contacted
   - Probe adapter type and serial number
   - Description of the problem

3. Place the midbus probe head in a separate static shielding bag and close with nonstatic generating tape.

CAUTION. *To prevent damage to the probe head and preprocessor unit, do not place the probe head in the large static shielding bag with the processor unit.*

Figure 1-19: Place the probe head in a static shielding bag

4. Place the preprocessor unit inside a static shielding bag.
5. Place the foam end caps on both sides of the preprocessor unit and place the preprocessor unit inside the cardboard carton (see Figures 1–20).

Figure 1-20: Place the end caps on the preprocessor

6. To stabilize the preprocessor unit, place the smaller cardboard carton next to the preprocessor unit (see Figure 1–21). If you are returning the slot board, place it inside a static shielding bag and inside the smaller carton.

7. Place the midbus probe head and cable on top of the preprocessor unit.

Figure 1-21: Place the preprocessor unit in the carton

8. Close and tape the cardboard carton.

Operating Basics

Setting Up the Support

This section provides information on how to set up the TMS817 and TMS818 PCIExpress bus support products with a Tektronix logic analyzer. The information covers software installation.

Before you acquire and display data, you must load the appropriate support and specify setups for clocking and triggering. The support provides default values for each of these setups, but you can change them on the Tektronix logic analyzer as needed.

Installing the Support Software

***NOTE**. Before you install any software, you should verify that the bus support software is compatible with the logic analyzer software.*

To install the software on your Tektronix logic analyzer, follow these steps:

1. Insert the CD-ROM in the CD drive.
2. Click the Windows Start button, point to Settings, and click Control Panel.
3. In the Control Panel window, double-click Add/Remove Programs.
4. Follow the instructions on the screen for installing the software from the CD.

To remove or uninstall software, follow the above instructions except select Uninstall. You must close all windows before you uninstall any software.

Support Packages

The probe adapter products install support packages on the Tektronix logic analyzer. Each support package can offer different clocking and display options.

PCIEx1 This support package provides disassembly support for a x1 unidirectional link.

PCIEx2 This support package provides disassembly support for a x2 unidirectional link.

PCIEx4 This support package provides disassembly support for a x4 unidirectional link.

PCIEx8 This support package provides disassembly support for a x8 unidirectional link (TMS817 only).

PCIEx16 This support package provides disassembly support for a x16 unidirectional link (TMS817 only).

PCIEx421_test This support package is used in the self-test procedure to verify correct operation of the preprocessor unit and cables when configured for x1, x2, or x4 link widths when using a 102 or 136 channel Tektronix logic analyzer modules.

PCIEx8_test This support package is used in the self-test procedure to verify correct operation of the preprocessor and cables when configured for a x8 link width when using a 136 channel Tektronix logic analyzer module.

PCIEx16a_test
PCIEx16b_test
PCIEx16c_test These support packages are used in the self-test procedure to verify correct operation of the preprocessor and cables when configured for a x16 link width and using two 136 channel Tektronix logic analyzer modules.

Custom Clocking Option

A special clocking program is loaded to the module every time you load one of the support packages. Each support package offers different clocking options.

Custom is one of three menu choices found in the Clocking Options menu (the other two options are External and Internal clocking).

All supports provide the following Custom clocking options:

- All Cycles Types
- Valid Packet Cycles Only
- TLPs and DLLPs Only
- TLPs Only

All Cycle Types All channels are acquired at each edge of the clock.

Valid Packet Cycles Only All channels are acquired at each edge of the clock when TS1/TS2, Fast Training Sequences, Skip packets, and TLPs or DLLPs are detected by the preprocessor unit.

TLPs and DLLPs Only All channels are acquired at each edge of the clock when TLPs (Transaction Layer Packets) or DLLPs (Data Link Layer Packets) are detected by the preprocessor unit.

TLPs Only All channels are acquired at each edge of the clock when a TLP is detected by preprocessor unit.

Reference Tables

The *Reference* section of this manual contains three groups of tables which are listed below:

Group Definitions Group definition tables are located on page 4–69 of the *Reference* section.

Symbols Symbol tables are located on page 4–1 of the *Reference* section.

Channel Assignments The software automatically creates channel assignments for each support package. Channel assignment tables are located on page 4–173 of the *Reference* section.

Installing Trigger Programs

The following trigger programs are installed along with the TMS817 and TMS818 PCIExpress bus support packages.

- Trigger Programs for the PCIEx1 support package are installed in the C:\Program Files\TLA700\Supports\PCIEx1\EasyTriggers folder.
- Trigger Programs for the PCIEx2 support package are installed in the C:\Program Files\TLA700\Supports\0PCIEx02\EasyTriggers folder.
- Trigger Programs for the PCIEx4 support package are installed in the C:\Program Files\TLA700\Supports\0PCIEx04\EasyTriggers folder.
- Trigger Programs for the PCIEx8 support package are installed in the C:\Program Files\TLA700\Supports\0PCIEx08\EasyTriggers folder.
- Trigger Programs for the PCIEx16 support package are installed in the C:\Program Files\TLA700\Supports\PCIEx16\EasyTriggers folder.

Loading Trigger Programs

To load a trigger program from a support package, follow these steps:

1. Load the support package.
2. From the system window, click the 

Trigger button.

   Figure 2–1 on page 2–4 shows an open trigger window.
3. Scroll through the EasyTrigger programs to find the trigger program that you need.

A description of an EasyTrigger program is provided in the Trigger window (see Figure 2–1 for the location of the description). In this description some programs include an illustration of the packet that highlights the packet fields the program uses as trigger criteria.

4. Select an EasyTrigger program from the list, and fill in the fields.

   You may not be able to fill out as many fields as you want in the Trigger window because some link widths do not allow all fields to be specified within the EasyTrigger program due to trigger resource limitations.

You are now ready to trigger on the acquired data. For additional information on triggering, refer to the logic analyzer online help.

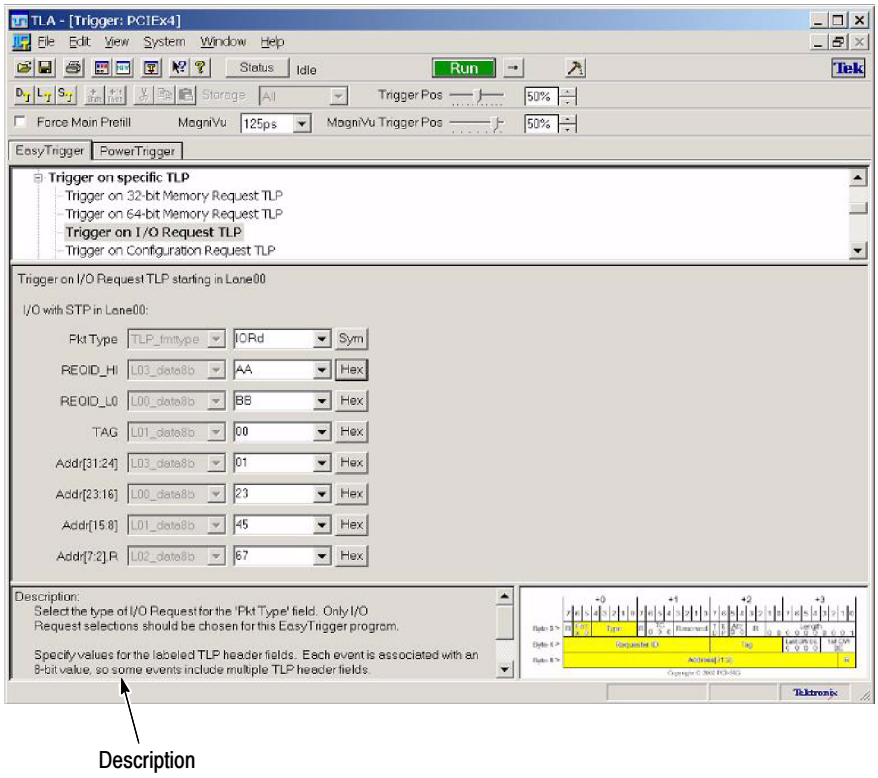

Figure 2-1: Trigger window

Trigger Programs

Following is a list of EasyTrigger programs:

Triggering on specific types of packets:

DLLP

- Trigger on Ack or Nak DLLP
- Trigger on Flow Control DLLP
- Trigger on Power Management DLLP
- Trigger on Vendor Specific DLLP

TLP

- Trigger on 32-bit Memory Request TLP
- Trigger on 64-bit Memory Request TLP
- Trigger on I/O Request TLP
- Trigger on Configuration Request TLP
- Trigger on Message Request TLP
- Trigger on Completion TLP

Triggering on packets by type:

DLLPs

- Trigger on DLLP
- Trigger on DLLP1 or DLLP2
- Trigger on DLLP1 followed by DLLP2
- Trigger on DLLP1 or DLLP2 or DLLP3
- Trigger on DLLP1 followed by DLLP2 followed by DLLP3

TLPs

- Trigger on TLP
- Trigger on TLP1 or TLP2
- Trigger on TLP1 followed by TLP2
- Trigger on TLP1 or TLP2 or TLP3
- Trigger on TLP1 followed by TLP2 followed by TLP3

TLPs and DLLPs

- Trigger on DLLP followed by TLP
- Trigger on TLP followed by DLLP
- Trigger on DLLP or TLP
- Trigger on DLLP1 followed by DLLP2 followed by TLP
- Trigger on DLLP1 followed by DLLP2 followed by TLP
- Trigger on TLP1 followed by TLP2 followed by DLLP

Trigger on Simple Events

- Trigger on Hot Reset (Reset bit Asserted)
- Trigger on rule violation
- Trigger on Electrical Idle
- Trigger on Training Sequence
- Trigger on FTS
- Trigger on skip packet

Acquiring and Viewing Disassembled Data

This section describes how to acquire data and view it disassembled.

Acquiring Data

Once you load the support package, choose a clocking mode, and specify the trigger, you are ready to acquire and disassemble data.

If you have problems acquiring data, refer to your logic analyzer online help for more information about acquiring data.

Viewing Disassembled Data

You can view data in two display formats:

- Waveform (state)
- Listing (disassembly)

Waveform Window Format

The waveform display shows each lane of data in rows. This data is not disassembled.

***NOTE**. If a channel group is not visible, you must use Add waveform in the waveform window to make the group visible. Refer to* Changing How Data is Displayed *on page 2-11 for more information.*

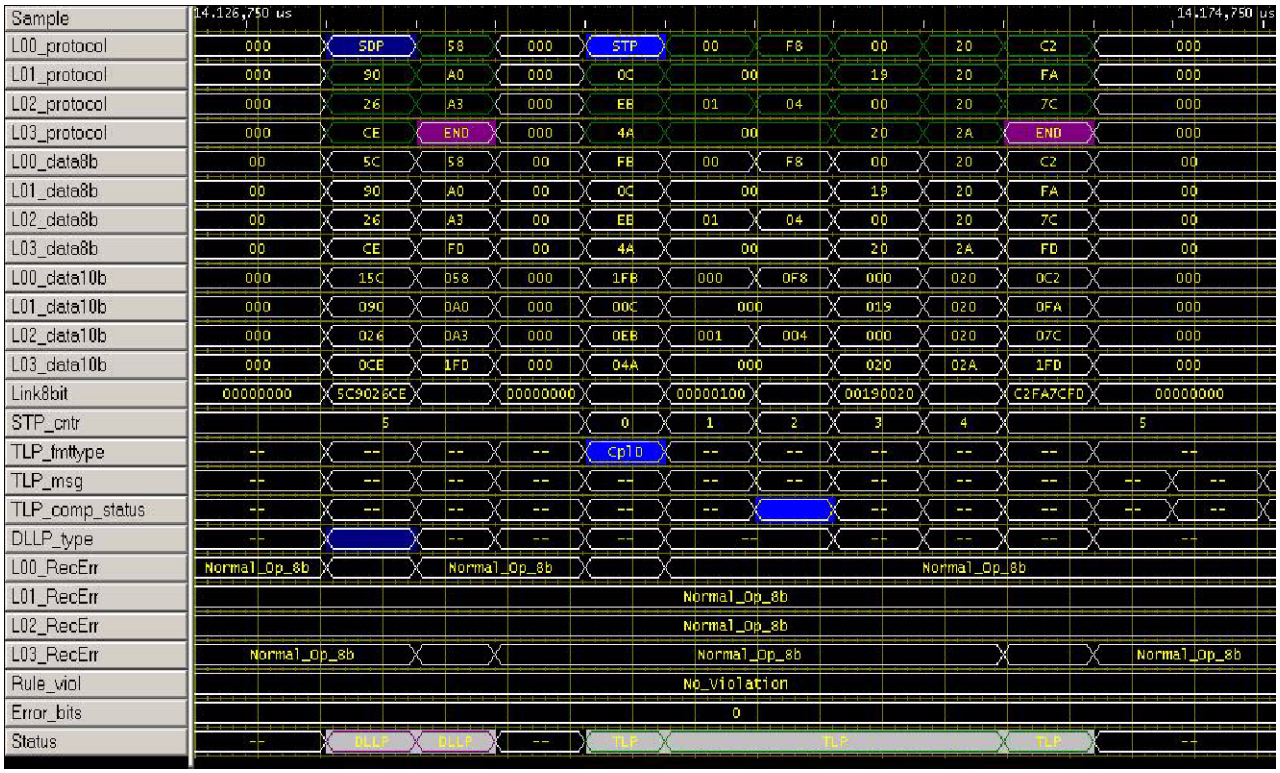

Figure 2-2: Waveform display

Listing Window Format

The listing display shows the packet fields as searchable columns. Figure 2-3 shows an example of a disassembled Listing-Window display format with the Extended Detail control ON.

***NOTE**. If a channel group is not visible, you must use Add Column in the Listing window to make the group visible. Refer to* Changing How Data is Displayed *on page 2-11.*

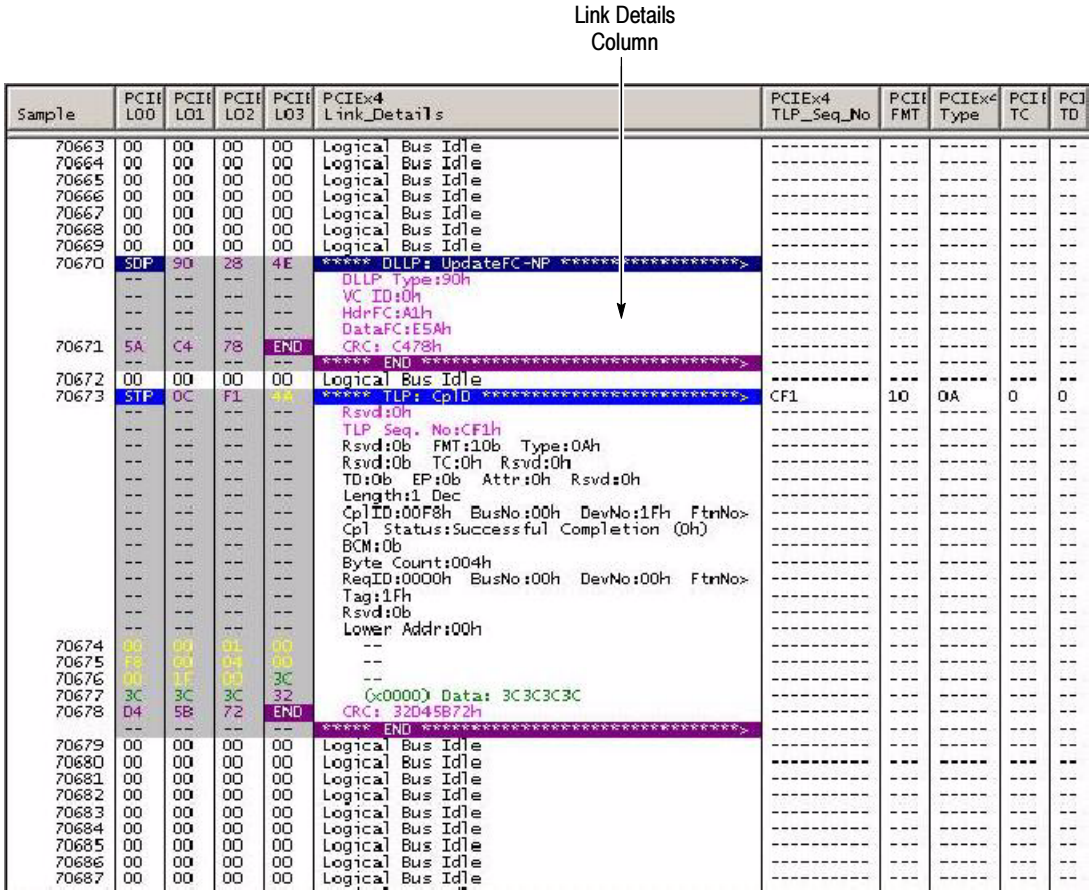

Figure 2-3: PCIEx4 display with Extended Details ON

The disassembler displays special characters and strings to indicate significant events. Table 2-1 describes these special characters and strings.

Table 2-1: Description of special characters in the display

Character or string	Definition
>>	Insufficient room on the screen to show all available data.
--	Invalid data or group, including read data.

Table 2–2 lists the groups displayed in the listing window for all link widths.

Table 2–2: Groups displayed in the listing window (for all link widths)

Group	Radix	Number of bits	Default state
L00 – L15*	TEXT	N/A	ON
Link_Details	TEXT	N/A	ON
TLP_Seq_No	HEX	8	ON
FMT	BIN	2	ON
Type	BIN	5	ON
TC	BIN	3	ON
TD	BIN	1	ON
EP	BIN	1	ON
Attr	BIN	2	ON
DataLength	DEC	10	ON
Requester_ID	HEX	16	ON
Req_Bus_No	HEX	8	ON
Req_Dev_No	HEX	5	ON
Req_Ftn_No	HEX	3	ON
Completer_ID	HEX	16	ON
Cpl_Bus_No	HEX	8	ON
Cpl_Dev_No	HEX	5	ON
Cpl_Ftn_No	HEX	3	ON
Bus_No	HEX	8	ON
Device_No	HEX	5	ON
Function_No	HEX	3	ON
Tag	HEX	8	ON
Last_DWBE	HEX	4	ON
First_DWBE	HEX	4	ON
Addr64	HEX	32	ON
Addr32	HEX	32	ON
Register_No	HEX	10	ON
Lower_Addr	HEX	6	ON
Msg_Code	HEX	8	ON
Vendor_ID	HEX	16	ON
Cpl_Stat	HEX	3	ON
BCM	BIN	1	ON
Byte_Count	HEX	12	ON

Table 2-2: Groups displayed in the listing window (for all link widths) (Cont.)

Group	Radix	Number of bits	Default state
TLP_Digest	HEX	32	ON
TLP_CRC	HEX	32	ON
AckNak_Seq_Num	HEX	12	ON
VC_ID	HEX	3	ON
HdrFC	HEX	8	ON
DataFC	HEX	12	ON
Vendor	HEX	24	ON
DLLP_CRC	HEX	16	ON

*** The actual number of lane columns that are displayed depends on the support package that is loaded.**

Invalid groups. The disassembler invalidates any group that is not valid for a given sample. The disassembler uses dashes to represent invalid groups.

Searching Through Data. Data searching is supported in the listing display. See *Hardware-Assisted Searching* on page 2-14 of this manual and the logic analyzer online help for additional search information.

Changing How Data is Displayed

There are other options listed in the Disassembly property page to control how you analyze and display disassembled data in the listing window.

For example, if you are searching for a TLP packet and do not want to see any other types of packets, choose Show TLPs Only.

In the listing window, the following bus specific fields allow you to further modify disassembled data to suit your needs.

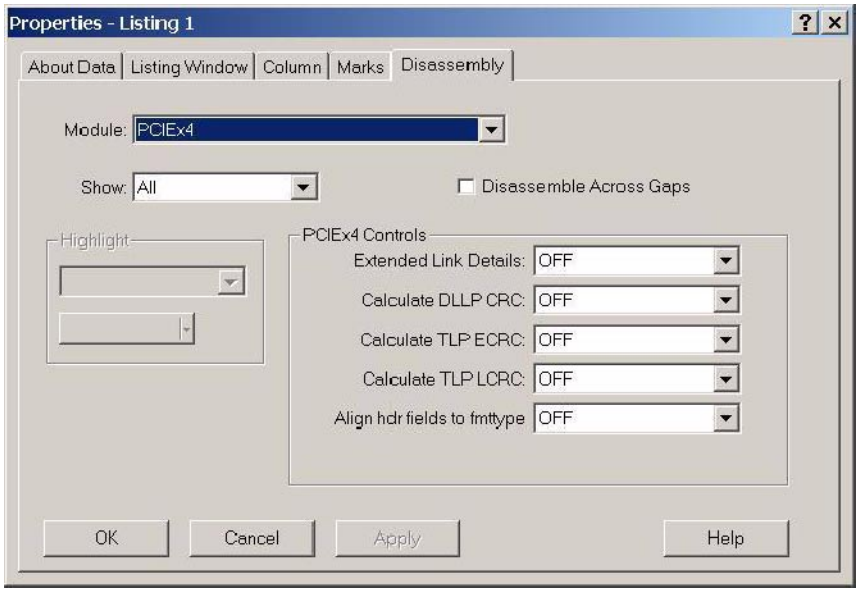

Figure 2-4: Disassembly tab in listing window

Table 2-3: Logic analyzer disassembly display options

Disassembly window	Option	Description
Show:	All (default)	All required data is disassembled and shown including logical idle samples.
	Non-Idle Samples	Logical idle samples are hidden.
	TLP/DLLPs Only	Only samples containing TLPs and DLLPs are shown.
	TLPs Only	Only samples containing TLPs are shown.
	TLP Headers Only	Only samples containing TLP headers are shown.
Highlight	None (default)	None (default)
Disassemble Across Gaps:	Yes No (default)	General listing window setting. (not recommended for PCIExpress data) No (default)

Any errors in link traffic detected by the disassembler are displayed regardless of the display option that you selected.

Bus Specific Fields

In the Controls submenus you can select any of the following controls to change the way data is displayed.

Extended Link Details. The Link_Details column of the listing window can be set to show or hide extended packet information by setting the Extended Link Details mode to ON or OFF (see page 2–9).

If set to OFF, the Link_Details column displays general packet information on a single line only.

If it is set to ON, the Link_Details column displays extended packet information on multiple lines. All packet fields are decoded and displayed in the Link_Details column. TLP payload data is displayed double word aligned along with the lower word address starting with the address acquired in the TLP header.

OFF (Default)

On

Calculate DLLP CRC. The disassembler calculates the CRC for DLLPs when this property is set to ON. If the calculated value differs from the value acquired from the link, an error message is displayed in the Link_Details column.

OFF (Default)

ON

Calculate TLP ECRC. The disassembler calculates the ECRC for TLPs when this property is set to ON. If the calculated value differs from the value acquired from the link, an error message is displayed in the Link_Details column.

Off (Default)

On

Calculate TLP LCRC. The disassembler calculates the LCRC for TLPs when this property is set to ON. If the calculated value differs from the value acquired from the link, an error message is displayed in the Link_Details column.

Off (Default)

On

Align hdr fields to fmttype. This property controls the placement of the TLP header and DLLP field values displayed in individual columns. The values for each packet are displayed on the same line despite the fact that they may be extracted from data spanning multiple samples. Displaying packets on the same line is done to make searching for packets by specifying the values of multiple packet fields possible.

By default, the field values are aligned with the sample containing the STP if it is a TLP or with the sample containing the SDP if it is a DLLP. By setting this property to ON, the packet field values are displayed on the same line as the TLP_fmttype and DLLP_type group values.

Off (Default)

On

10-bit Mode Acquisition

When the preprocessor unit is configured to acquire the link in 10-bit mode, the listing window displays the symbol encoding in the individual lane columns. No further link analysis is performed.

Hardware-Assisted Search

To use the hardware-assisted search you need software version 4.3 on your Tektronix logic analyzer. This section covers the following hardware-assisted search information:

- Behavior on page 2–15
- Rules for setting up hardware-assisted searches on page 2–15
- Example of a x16 link hardware-assisted search on page 2–19

Features

Following is a list of hardware-assisted search features:

- You can enter hardware search criteria in the Search Definition dialog box of the listing window.

  You can display disassembly columns in the listing window and use hardware-assisted searching.

- Elements of a search clause, such as groups and channels, do not need to be visible in the listing window.

- You can mix groups and channels (defined in the Search Definition dialog box) with disassembly groups in the same search clause, and then use the hardware-assisted search to search acquisition data.

Hardware-Assisted Search Behavior

The hardware-assisted search processes data in two parts. First, the hardware portion of the search finds samples that match the criteria defined by the setup groups. Then the software is used to determine if the sample matches the remaining criteria.

Once the hardware portion of the hardware-assisted search is started, the software processing is delayed until the hardware-assisted search finds a sample that matches the hardware portion of the search definition.

Hardware-Assisted Search Speed. If you follow the rules of ordering and logical combinations of setup groups, the hardware-assisted search can significantly reduce your search time and make it faster than using a software-driven search.

Review the following areas that may affect search speed:

- The setup groups defined in the Setup window contain channels that the hardware can search.

  Groups of channels from a single module are faster to process than groups from multiple modules that have their channels searched independently on more than one module. Multiple-module searches can result in false-positive matches (reported by individual modules) that need to be correlated with search results from other modules. A multiple-module search has the potential to slow the hardware-assisted search significantly.

- The type of data acquired and the type of search performed.

  If the module acquisition contains large amounts of data that is identical to the search criteria (from the perspective of the hardware), then the search speed is the same for the software-driven search and the hardware-assisted search. This is because the modules find many matches that the software needs to qualify.

  If the module acquisition contains data that is mostly different from the search criteria, hardware-assisted searches may be faster than software-driven searches.

Hardware-Assisted Search Rules

For the hardware-assisted search to occur in the listing window, you must follow the Module Search rules and the Clause Description rules.

Module Search Rules. Failure to meet these rules turns off hardware-assisted searching:

1. The module search expression must contain only *one active clause*. All other defined searches must be disabled (turned off).
2. The clause must use the *When Present* condition.

Clause DescriptionSearch Rules. The logical combination of hardware and software events determines when a hardware-assisted search is used. The search clause can contain events that are combined using AND and OR.

Events that are searched using hardware are called hardware events and events that can only be searched using software are called software events.

- Hardware events include user groups and channels without support package names, for example, L00_protocols and TLP_fmttype.
- Software events include sample, timestamp, anything events, and any group event using a disassembly group (groups prefixed with the support package name), for example, PCIEx8 L00 and PCIEx8 TLP_fmttype.

When evaluating the list of events in a clause, use the following rules:

***NOTE**. The first rule met is used to set up the hardware-assisted search. If no rule is met, software searching occurs.*

Evaluate the expression starting with the first event and ending with the last.

1. If all events in the expression are ORed together, for example, "(((A | B) | C) | D)...", then all must be hardware events to use hardware-assisted search. The use of any software event in this expression disables the hardware-assisted search.
2. When an OR expression is followed by one or more ANDs, for example, "(((A | B) & C) & D)" and if the OR contains all hardware events, then these are used for hardware-assisted search. The OR must precede any AND expression in the list of events.

Evaluate the expression starting with the last event, and ending with the first.

3. For all continuous ANDed events, any hardware events in this series can be used in hardware-assisted search. For example, "(((A & B) & C) & D)..." all software events within the ANDed set are ignored during hardware-assisted search. If none are hardware events, then a software search is used.
4. When OR expressions are to the left of one or more ANDs, for example, "(((A | B) & C) & D)...", and fail Clause Description search rule 2, only those hardware events that are ANDed are considered for hardware-assisted search.

5. For all contiguous OR event, expressions to the right determine if they are all hardware events. If they are, then look through the next set of ANDed events to the left for one hardware event and include this in the OR. For example, "(((A & B) | C) | D)...". Hardware-assisted search is then executed on this set of events.

Table 2-4: Clause description search rules

Rule	Expression	As evaluated from ...	Requirements for enabling hardware-assisted searching
1	"(((**A** | **B**) | **C**) | **D**)..."	Left to Right	All events must be hardware events
2	(((**A** | **B**) & C) & D)..."	Left to Right	All OR events must be hardware events
3	"(((A & B) & C) & D)..."	Right to Left	One or more events must be hardware events
4	"(((A | B) & **C**) & **D**)..."	Right to Left	One or more AND events must be hardware events
5	"(((A & B) | **C**) | **D**)..."	Right to left	All OR events must be hardware events. One or more AND events must be hardware vents.

Hardware-Assisted Search Example

Starting on page 2-17 is a hardware-assisted search example you can perform that uses a PCIExpress acquisition and contains data acquired from a x16 link. You can also use this example for other link widths, but you may need to use different groups depending on which support is loaded.

The TLPs can start in one of four lanes in a x16 link. Figure 2-5 shows the four TLP starting positions and the format type field locations for a memory write TLP.

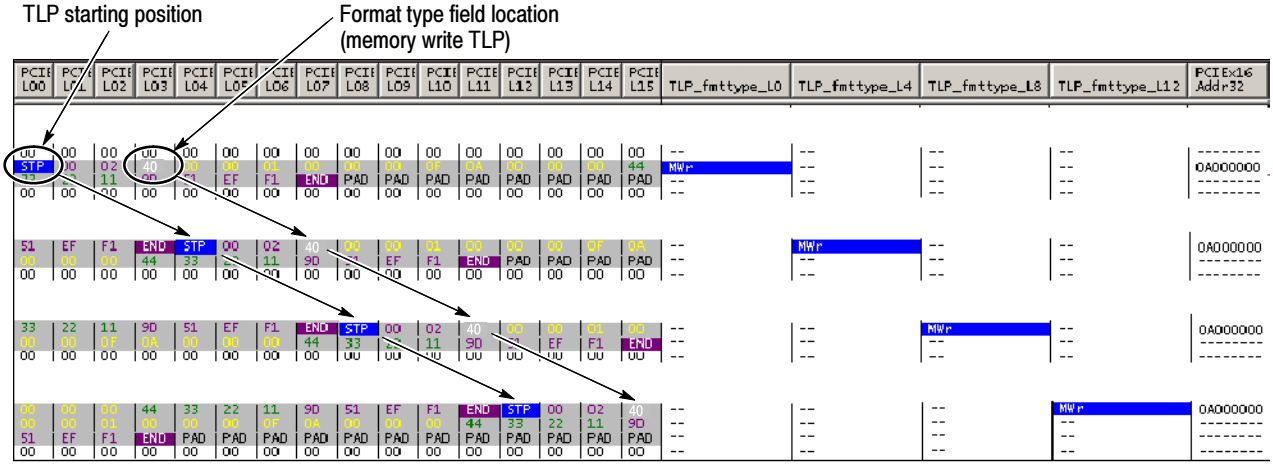

Figure 2-5: MWr TLP starting positions and relevant fields

To execute a hardware-assisted search for a memory write TLP at a specific address for a x16 link, follow this example:

1. In the Define Search dialog box, click the group arrow and add these groups:

   TLP_fmttype_L0

   TLP_fmttype_L4

   TLP_fmttype_L8

   TLP_fmttype_L12

   Using these groups in the Define Search dialog box enables a hardware-assisted search. The groups are defined in the setup window.

   These groups have symbol tables that describe the type of TLP. There is a TLP_fmttype group for each TLP starting lane in the x16 support package.

2. Click the AND/OR button and choose OR for each group. This step ORs the groups together.

3. Select the = comparison operator.

4. Click the value arrow and choose MWr for memory write TLPs starting in each starting lane.

   Now the modules are configured for a hardware-assisted search for memory write TLP packets starting in all four starting lanes.

5. Click the group arrow and select the PCIEx16 Addr32 group.

   The PCIEx16 Addr32 group contains the address value for packets starting in all four starting positions.

6. For the PCIEx16 Addr32 group, click the AND/OR button and select And. This adds PCIEx16 Addr32 group to the rest of the search clause.

7. Select the = comparison operator.

8. Enter the hexadecimal address, 0A000000, in the Hex field. This allows a search for a particular 32-bit address value for the memory write TLP.

9. Click the When condition arrow and select Present.

   When all the search criteria are entered, the Search Definition dialog box displays a Boolean expression describing the clause. See Figure 2–6. The search clause must match one of the search rules on page 2–16 for the hardware-assisted search to run.

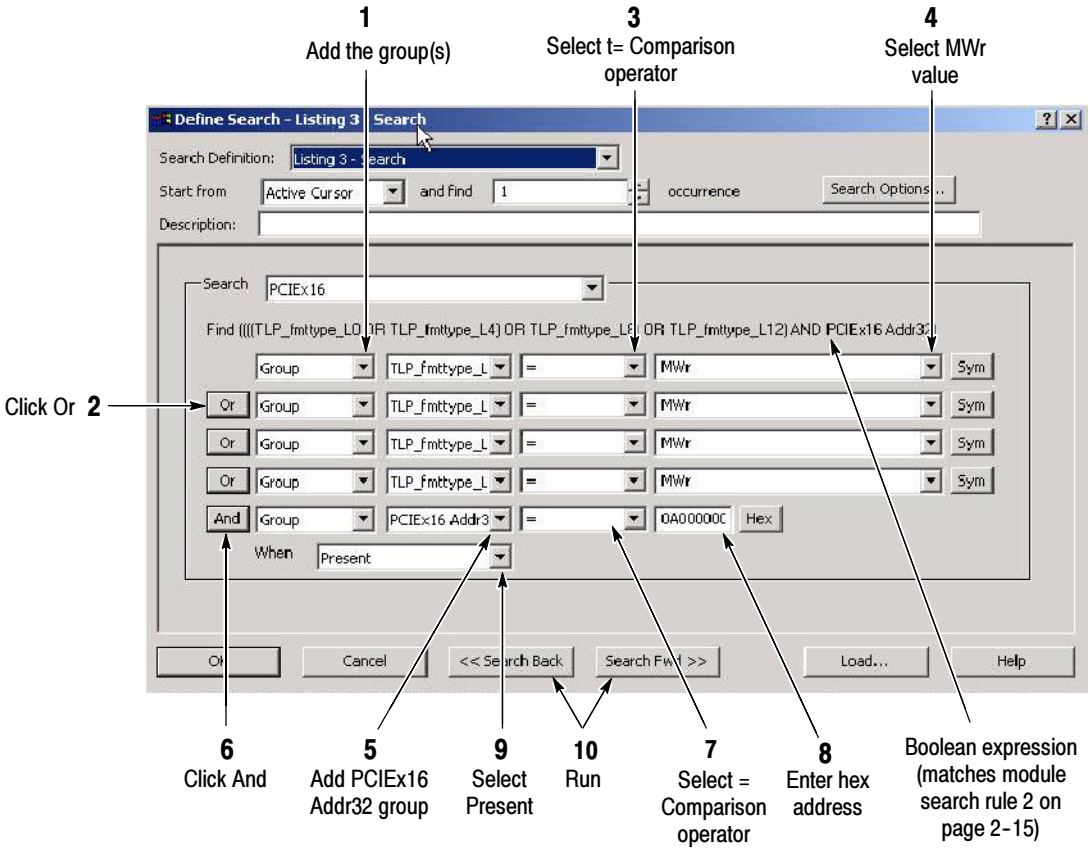

Figure 2-6: Search for a memory write to address 0A000000

In Figure 2-6, the Boolean expression matches rule 2 of the Module Search rules on page 2-16 and the hardware should be used to help find a match.

10. Click the <<Search Back or Search Forward>> buttons to start the hardware-assisted search.

Troubleshooting Hardware-Assisted Searches. If a hardware-assisted search takes too long, then the Boolean expression may not match one of the Search Clause Description rules or the Boolean expression may contain events that slow the hardware-assisted search.

- Check that your Boolean expression matches one of the rules listed on page 2-16.

  If the expression does not match one of the Search Clause Description rules, then the search is performed entirely in software.

- To increase the speed of the hardware-assisted search, see *Hardware-Assisted Search Speed* on page 2-15.

***NOTE**. The search requires all field values from all groups to be displayed in the same line. If one or more of the groups have values displayed on different lines, the search will fail to find a match, even if all the information is obtained from the same packet.*

For x1 and x2 link widths, packet field information from disassembly processing is displayed in columns aligned with the start of packet symbol, not the TLP_fmttype textual value. To perform a search similar to this example for these link widths, set the disassembly property field, *Align hdr fields to fmttype,* to ON (see page 2–14).

Special Messages

This section provides information about the special messages used in this software support. The disassembler uses special messages to indicate significant events. These messages are highlighted in red in the Link Details column of the listing window. Tables 2–5 through 2–12 show the messages and their descriptions.

The special messages are in addition to the errors detected by the preprocessor hardware listed in PCIEx_RuleViol.tsf and PCIEX_RecErr.tsf files. These files are located in the TLA700\Supports\PCIEx directories. These symbol tables are used by the Rule_viol and L??_RecErr groups.

Table 2–5: Training sequence messages

Message	Description
Error: Duplicate Lane Number Assignment	Indicates that more than one lane has been assigned the lane number.
Error: Lane No. value in 'L??' exceeds max link width	Indicates that the lane number as acquired in the training sequence is higher than the link width.
Lane??: Lane Polarity Inversion	Indicates that the lane is inverted. The corresponding cable at the input to the preprocessor needs to be inverted.

Table 2-6: Packet framing messages

Message	Description
Error: Malformed packet – Expected END symbol	Indicates that the END symbol was not found in the lane and sample as determined based on type of packet and packet starting lane.
Error: Malformed TLP?	Indicates that the TLP is not framed by STP and END or EDB symbols, or starts in an incorrect lane.
Error: Invalid symbol between END and SDP	Indicates that something other than PAD symbols was acquired between the END symbol from the 1st packet and the SDP symbol of next packet, or two packets are in one symbol time and the next packet does not start immediately following the END symbol of the 1st packet.
Error: SDP expected in lane 00	Indicates that an SDP symbol was found in a lane other than lane 00 when not following another packet, or when a previous packet ended in the previous sample.
Error: STP expected in lane 00	Indicates that an STP symbol was found in a lane other than lane 00 when not following another packet, or when a previous packet ended in the previous sample.
Error: Malformed packet preceding TLP???	Indicates that the TLP is not framed by STP and END or EDB symbols, starts in an incorrect lane.
Error: Invalid symbol between END and STP	Indicates that something other than PAD symbols was acquired between the END symbol from 1st packet and the STP symbol of next packet or when two packets are in one symbol time and the next packet does not start immediately following the END symbol of the 1st packet.
Error: Malformed packet - Missing END???	Indicates that no END/EDB symbol was found in the expected sample and lane.
Error: Abnormal packet termination	Indicates that the packet was interrupted and terminated by a skip ordered set, training sequence, or FTS, TLP, DLLP.
Non-Idle Bus	Indicates the link is supposed be in logical idle at this sample, but a nonzero value is found in one or more lanes.

Table 2-7: DLLP messages

Message	Description
Error reading DLLP	Indicates that a general error occurred while trying to decode the DLLP possibly caused by a gap/suppression of data.

Table 2-8: TLP header messages

Message	Description
Error reading TLP	Indicates a general error occurred while trying to decode the TLP header possibly caused by a gap/suppression of data.
Error Forwarding/Poisoned TLP	Indicates that the EP field of TLP header is HIGH.
Error: Invalid 1stDWBE/Length values for Req TLP	Indicates that the TLP length field > 1 and First DWBE field is 0 for request TLPs.
Error: Invalid LastDWBE/Length values for Req TLP	Indicates that the TLP length field is 1 and Last DWBE field is not 0, or TLP length field > 1 and Last DWBE field is 0 for request TLPs.
Zero Length Read Request -- Possible Flush	Indicates that the TLP length field is 1, Last DWBE is 0, First DWBE is 0, for Memory Read Request.
Error: Invalid Traffic Class for Message TLP	Indicates that the TC field was not zero.
Completion Status: + 'Unsupported Request', 'Config Req Rtry Stat', or 'Completer Abort'	Indicates completion status other than 'Successful Completion'.
TLP: Msg – + 'ERR_COR', 'ERR_NONFATAL', or 'ERR_FATAL'	Indicates that an error message TLP was acquired.

Table 2-9: TLP payload messages

Message	Description
Error: Expecting valid data in Lane??	Indicates that an invalid data such as a Kcode was found in a sample and lane that should be TLP payload data.
Error: Payload exceeds TLP Length Field: #	Indicates that the payload data exceeds what the length field indicated. Possible causes include: a missing END symbol, or the inclusion of TLP Digest field without the TD field HIGH.
Error: Packet data total: # DW – does not match length field	Indicates that the payload data exceeds what the length field indicated. Possible causes include: a missing END symbol, or the inclusion of TLP Digest field without the TD field HIGH.

Table 2-10: TLP digest messages

Message	Description
Error: Missing TLP Digest or Gap in acquisition	Indicates that the TLP digest field was not found in the sample expected.
Error: Expected TLP Digest starting in Lane??	Indicates that the TLP digest field was not found in the lane expected.

Table 2-11: CRC checking messages

Message	Description
Error: ECRC mismatch	ECRC value acquired in TLP digest field does not match the ECRC value calculated by applying the ECRC algorithm to the acquired data. Possible causes include incorrect ECRC at the transmitter, poor signal quality at probe head, different algorithm used between transmitter and TLA software, incorrect polarity or ordering of cables at the input of the preprocessor, problem with the cables or connecting the preprocessor to the TLA.
Error: CRC mismatch	The TLP or DLLP CRC acquired does not match the CRC value calculated by applying the CRC algorithm to the acquired data.

Table 2-12: General acquisition messages

Message	Description
Error: Missing Data - Gap in TLP header	Complete decode of TLP header was not possible due to a gap/suppression of data.
Error: Missing Data - Gap in DLLP	Complete decode of DLLP was not possible due to a gap/suppression of data.
Error: Missing Data - Gap in Training Sequence	Complete decode of Training Sequence ordered set was not possible due to a gap/suppression of data.
Error: Missing Data - Gap in packet	Complete decode of Training Sequence ordered set was not possible due to a gap/suppression of data.
Lane-to-Lane Deskew Error	The link was not properly deskewed by preprocessor unit. Also displayed when sample contains SKP (K28.0) symbols in one or more lanes but not all lanes of the link. This error message is displayed until a sample containing all SKP (K28.0) symbols is found. No further post processing of packets is performed when the link is not deskewed.

Viewing an Example of Disassembled Data

The demonstration system file (or demonstration reference memory), provided on your CD-ROM, lets you view an example of how the Bus cycles look when they are disassembled. Viewing this system file is not a requirement when preparing the module for use. You can view the system file without connecting the logic analyzer to your target system.

For information on how to view the file refer to the logic analyzer online help.

Specifications

Specifications

This section contains information regarding the specifications of the TMS817 and TMS818 products.

Circuit Description

The probe adapter preprocesses all signals on the PCIExpress bus before the PCIExpress bus signals are captured by the logic analyzer. Following is a list of probe head and preprocessor unit functions.

Probe head

The probe head performs the following functions:

- The probe head accesses the PCIExpress bus signals.
- The plugs at the end of the probe head cables allow you to acquire different combinations of lane positionings and polarities.

Preprocessor unit

The preprocessor unit preprocesses the PCIExpress bus signals for the logic analyzer in the following sequence:

1. Frames the serial data into 10 b symbols based on K28.5 patterns on the bus.
2. Converts the 10 b serial data to 10 b parallel symbols and then converts them into 8 b parallel symbols with a few extra control bits.
3. Descrambles the 8 b symbols. Then based on SKIP ordered sets that occurred on those lanes, aligns multiple lanes of 8 b symbols.
4. Finally, the preprocessor unit generates qualifiers for the logic analyzer to allow for easy triggering.

Derived Signals

Table 3–1 lists and defines the derived signals.

Table 3–1: Derived signals

Derived signal	Definition
STP_packet_ (0–3)	These signals indicate the the sample is part of a TLP, the position of the STP symbol, and the lane where the TLP started.
STP_cntr (0–12)	These signals indicate and track the number of symbol times that elapse on the PCIExpress bus since the beginning of the present TLP packet. Counter is cleared by STP.
SDP_packet_(0–3)	These signals indicate the sample is part of a DLLP, the position of the SDP symbol, and the lane where the DLLP started.
Rule_viol (0–2)	These signals indicate that a violation of a specific PCIExpress protocol rule.
SKP_detect	This signal indicates the sample is part of a skip ordered-set.
Elec_idle_flag	This signal indicates the link is in the electrical idle state.
TS_detect	This signal indicates that the sample is part of a training sequence ordered-set.
FTS_detect	This signal indicates that the sample is part of a fast-training sequence ordered-set.
END_EDB_detect	This signals indicates that either an END or EDB is present on a lane in this sample.

Specification Tables

Table 3–2 lists the electrical requirements the target system must produce for the probe adapter to acquire correct data.

Table 3–2: Electrical specifications

Characteristics	Requirements
Preprocessor unit AC power requirements	
Input Voltage	100 - 240 VAC ± 10% CAT-II
Input Frequency	50 - 60 Hz
Input Current	4.0 A (400 VA) Maximum

Table 3-2: Electrical specifications (cont.)

Characteristics	Requirements
Target system	
Specified data rate	2.5 Gb/s ±100 ppm maximum with no external clock 2.5 Gb/s ±10% ppm maximum with external clock
Spread Spectrum Frequence modulation rate	0-0.5% @ 33 kHz maximum (must use clock cable)
Minimum Eye requirement	190 ps wide, 100 m V_{p-p} differential trapezoidal (at midbus footprint with probe load)
External Clock input	100 MHz ±300 ppm maximum
External Clock differential swing	800 m V_{p-p} minimum, 4.0 V_{p-p} maximum (400 mV swing per side of differential pair)
External clock voltage range	0 to 2 V
Midbus probe head timing violation	
Insertion loss	-1.0 dB
Insertion jitter	40 ps p-p
Slot board with probe head timing violation	
Insertion loss	-2.7 dB
Insertion jitter	25 ps p-p

Table 3-3: Environmental specifications

Characteristic	Description
Temperature	
Maximum operating	+40° C (+104° F)
Minimum operating	10° C (50° F)
Nonoperating	-30° C to +70° C (-22° F to +158° F)
Humidity	10 to 90% relative humidity, noncondensing up to 30° C, maximum wet bulb temperature of 29° C to 40° C
Altitude	
Operating	3 km (10,000 ft) maximum
Nonoperating	12 km (40,000 ft) maximum
Electrostatic immunity	The probe adapter is static sensitive

Table 3-3: Environmental specifications (cont.)

Characteristic	Description
Required airflow clearances (preprocessor unit)	
Sides	2 in (5.08 cm)

Load Models

Figure 3-1 shows the load model for the midbus probe adapter PCIExpress signals.

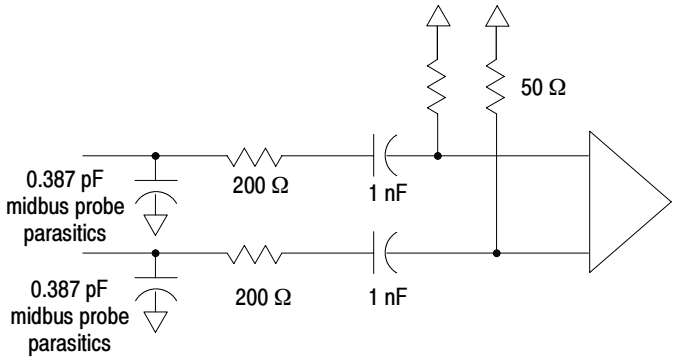

Figure 3-1: Midbus load model

Figure 3-2 shows the load model for the Slot board PCIExpress signals.

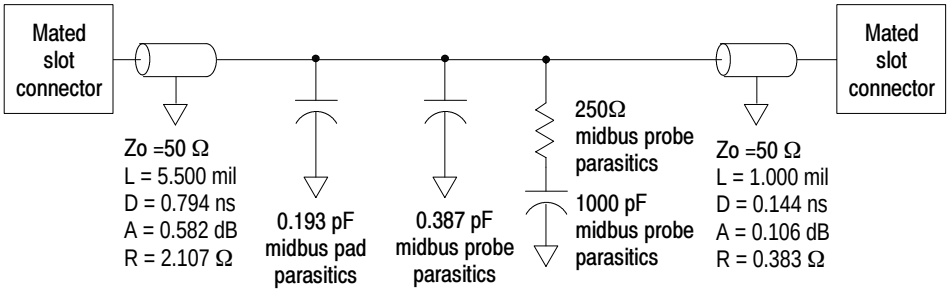

Figure 3-2: Slot load model

Table 3-4: Certifications and compliances

Category	Standards or description	
EC Declaration of Conformity – Low Voltage	Compliance was demonstrated to the following specification as listed in the Official Journal of the European Communities:	
	Low Voltage Directive 73/23/EEC, amended by 93/68/EEC	
	EN 61010-1/A2:1995	Safety requirements for electrical equipment for measurement control and laboratory use.
U.S. Nationally Recognized Testing Laboratory Listing	UL3111-1	Standard for electrical measuring and test equipment.
Canadian Certification	CAN/CSA C22.2 No. 1010.1	Safety requirements for electrical equipment for measurement, control, and laboratory use.
Installation (Overvoltage) Category Descriptions	Terminals on this product may have different installation (overvoltage) category designations. The installation categories are:	
	CAT III	Distribution-level mains (usually permanently connected). Equipment at this level is typically in a fixed industrial location.
	CAT II	Local-level mains (wall sockets). Equipment at this level includes appliances, portable tools, and similar products. Equipment is usually cord-connected.
	CAT I	Secondary (signal level) or battery operated circuits of electronic equipment.
Equipment Type	Test and measuring	
Safety Class	Class 1 (as defined in IEC 61010-1, Annex H) – grounded product	
Overvoltage Category	Mains input: Overvoltage Category II (as defined in IEC 61010-1, Annex J) All other inputs and outputs: 5 V maximum	
Pollution Degree Descriptions	Pollution Degree 2 (as defined in IEC 61010-1). Note: Rated for indoor use only.	

Dimensions

Figure 3-4 shows the dimensions of the TMS817 and TMS818 PCIExpress preprocessor unit.

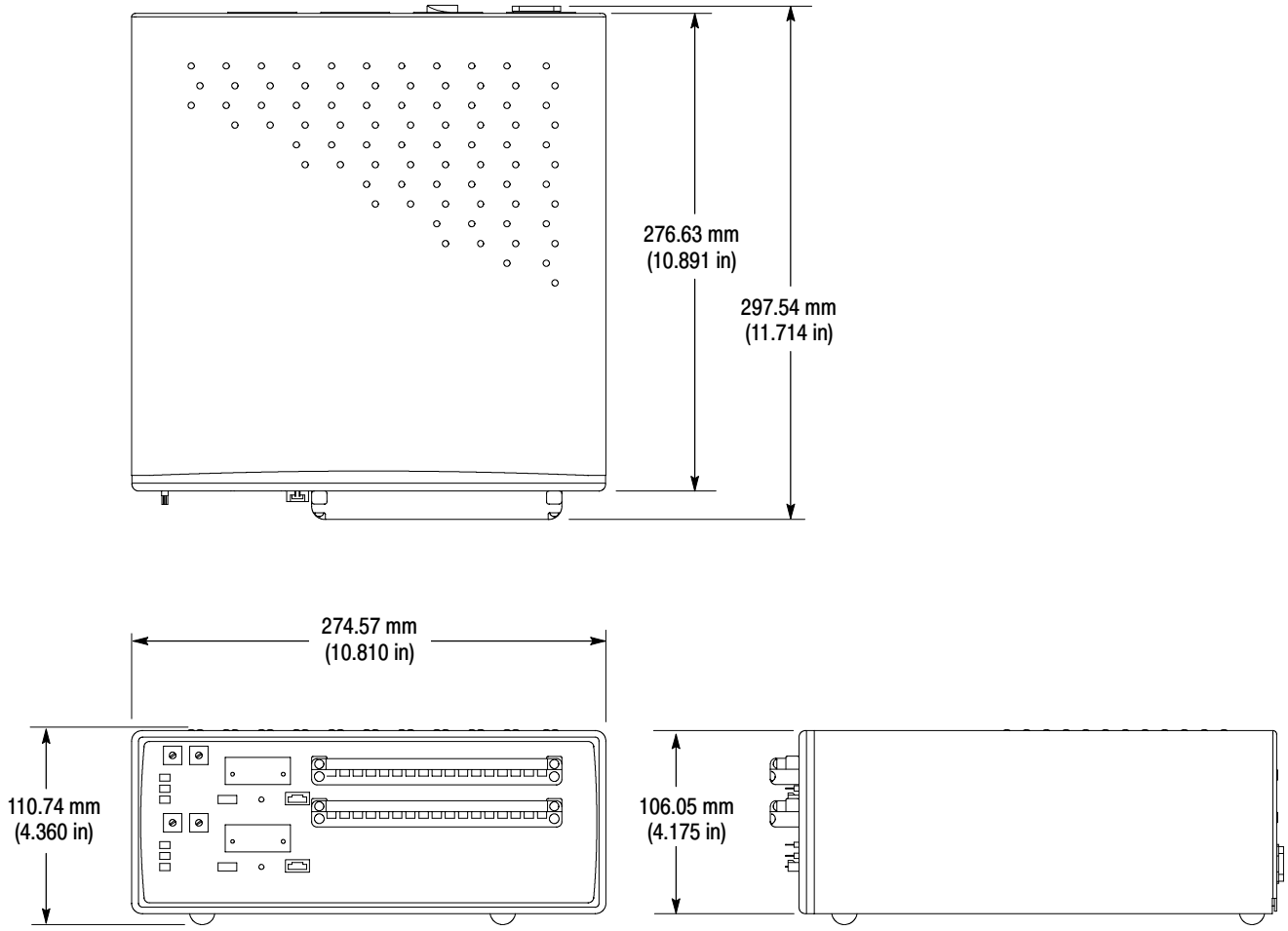

Figure 3-3: Dimensions of the preprocessor unit

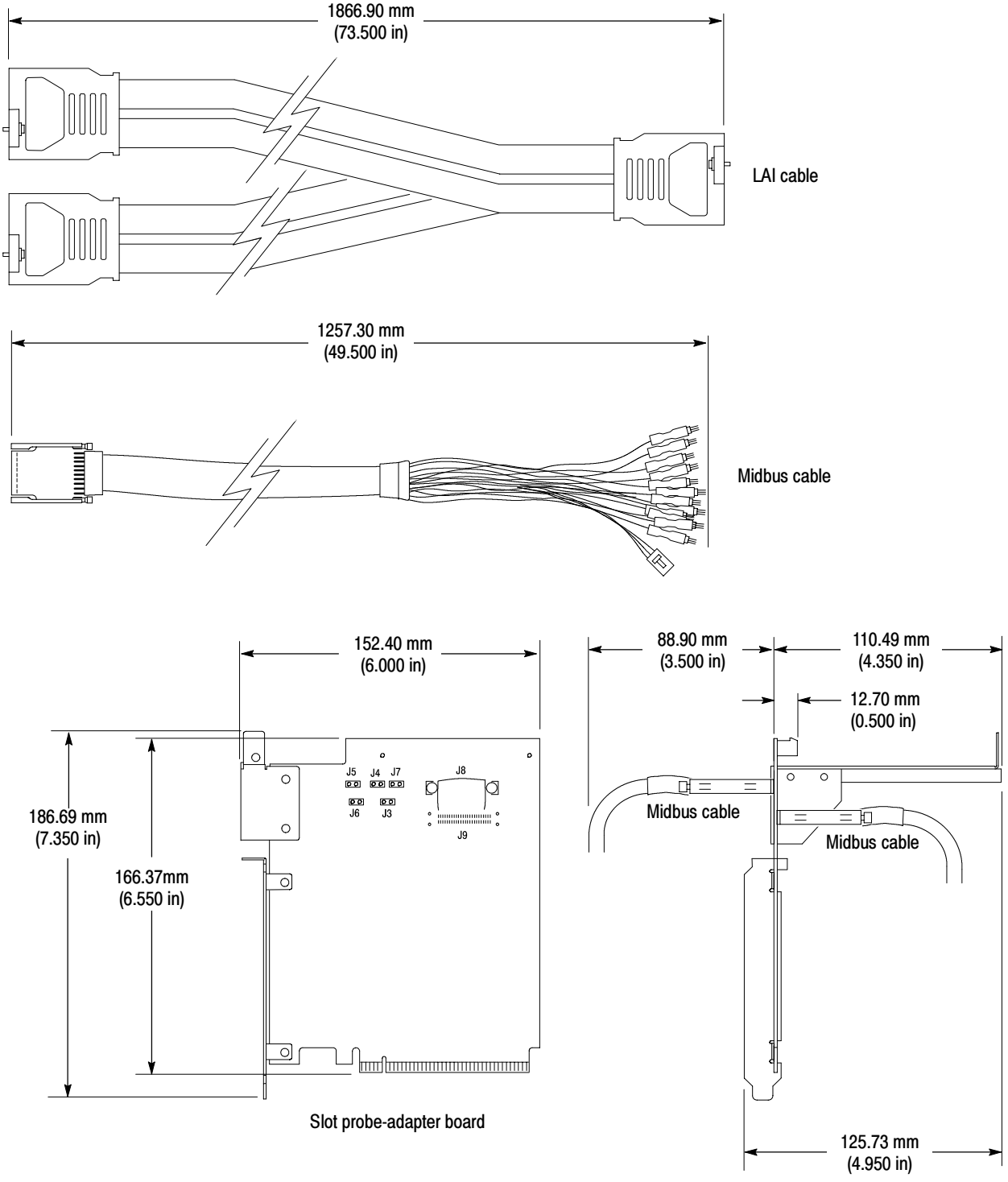

Figure 3-4: Dimensions of the cables and probe heads

Reference

Symbol Tables

The TMS817 and TMS818 software supplies symbol table files for all the PCIExpress channel groups. Information on basic operations in your logic analyzer online help menu describes how to use symbolic values for displaying channel groups symbolically.

For channel groups with an associated symbol table file, the display radix can be set to “Symbolic” and the proper symbol table can be selected by using the column properties dialog in the listing window display.

PCIEx1 Symbol Tables

Table 4- 1: PCIEx1_TLP_fmttype symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	------	Symbol_Table_Enable
	| +----------	------	------	Rule_viol_2
	| |+----------	------	------	Rule_viol_1
	| ||+----------	------	------	Rule_viol_0
	| ||| +----------	------	------	STP_packet_3
	| ||| | +----------	------	------	STP_cntr_12
	| ||| | |+----------	------	------	STP_cntr_11
	| ||| | ||+----------	------	------	STP_cntr_10
	| ||| | |||+----------	------	------	STP_cntr_09
	| ||| | ||||+----------	------	------	STP_cntr_08
	| ||| | |||||+----------	------	------	STP_cntr_07
	| ||| | ||||||+----------	------	------	STP_cntr_06
	| ||| | |||||||+----------	------	------	STP_cntr_05
	| ||| | ||||||||+----------	------	------	STP_cntr_04
	| ||| | |||||||||+----------	------	------	STP_cntr_03
	| ||| | |||||||||+----------	------	------	STP_cntr_02
	| ||| | ||||||||||+----------	------	------	STP_cntr_01
	| ||| | ||||||||||+----------	------	------	STP_cntr_00
	| ||| | |||||||||| +----------	------	------	L00_data_06 Fmt[1]
	| ||| | |||||||||| | +----------	------	------	L00_data_05 Fmt[0]

Table 4-1: PCIEx1_TLP_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | ||||||||||| | | +--------	------	------	L00_data_04 Type[4]
	| ||| | ||||||||||| | | | +------	------	------	L00_data_03 Type[3]
	| ||| | ||||||||||| | | | | +----	------	------	L00_data_02 Type[2]
	| ||| | ||||||||||| | | | | |+---	------	------	L00_data_01 Type[1]
	| ||| | ||||||||||| | | | | ||+--	------	------	L00_data_00 Type[0]
	| ||| | ||||||||||| | | | | |||			
MRd	1 000 1 0000000000011 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk	1 000 1 0000000000011 0 X 0 0 001			Mem Read Req. Locked
MWr	1 000 1 0000000000011 1 X 0 0 000	@white	@blue	Mem Write Request
IORd	1 000 1 0000000000011 0 0 0 0 010			I/O Read Request
IOWr	1 000 1 0000000000011 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0	1 000 1 0000000000011 0 0 0 0 100			Config. Read Type 0
CfgWr0	1 000 1 0000000000011 1 0 0 0 100	@white	@blue	Config. Write Type 0
CfgRd1	1 000 1 0000000000011 0 0 0 0 101			Config. Read Type 1
CfgWr1	1 000 1 0000000000011 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx	1 000 1 0000000000011 0 1 1 0 000			Msg Req.- Routed to Route complex
Msg-Rtd_by_addr	1 000 1 0000000000011 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID	1 000 1 0000000000011 0 1 1 0 010			- Routed by ID
Msg-Broadcast_fm_rt_cmplx	1 000 1 0000000000011 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Local_term_at_recv	1 000 1 0000000000011 0 1 1 0 100			- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx	1 000 1 0000000000011 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved	1 000 1 0000000000011 0 1 1 0 XXX			- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx	1 000 1 0000000000011 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr	1 000 1 0000000000011 1 1 1 0 001			- Routed by Address

Table 4-1: PCIEx1_TLP_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
MsgD-Rtd_by_ID	1 000 1 0000000000011 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx	1 000 1 0000000000011 1 1 1 0 011			- Broadcast from rt complex
MsgD-Local_term_at_recv	1 000 1 0000000000011 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx	1 000 1 0000000000011 1 1 1 0 101			- Gathered and routed to rt complex
MsgD-reserved	1 000 1 0000000000011 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS	1 000 1 0000000000011 0 1 1 1 XXX			Msg for Advanced Switching
MsgASD	1 000 1 0000000000011 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl	1 000 1 0000000000011 0 0 0 1 010			Completion wo/ data
CplD	1 000 1 0000000000011 1 0 0 1 010	@white	@blue	Completion w/data
CplLk	1 000 1 0000000000011 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read
CplDLk	1 000 1 0000000000011 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	1 XX1 X XXXXXXXXXXXXX X X X X XXX	@black	@red	
ERROR	1 X1X X XXXXXXXXXXXXX X X X X XXX	@black	@red	
ERROR	1 1XX X XXXXXXXXXXXXX X X X X XXX	@black	@red	
--	X XXX X XXXXXXXXXXXXX X X X X XXX			

Table 4-2: PCIEx1_TLP_msg symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------------------	------	-----	Symbol_Table_Enable
	| +----------------------------------	------	-----	Rule_viol_2
	| |+---------------------------------	------	-----	Rule_viol_1
	| ||+--------------------------------	------	-----	Rule_viol_0
	| ||| +------------------------------	------	-----	STP_packet_3
	| ||| | +----------------------------	------	-----	STP_cntr_12

Table 4-2: PCIEx1_TLP_msg symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | |+----------	------	-----	STP_cntr_11
	| ||| | ||+----------	------	-----	STP_cntr_10
	| ||| | |||+----------	------	-----	STP_cntr_09
	| ||| | ||||+----------	------	-----	STP_cntr_08
	| ||| | |||||+----------	------	-----	STP_cntr_07
	| ||| | ||||||+----------	------	-----	STP_cntr_06
	| ||| | |||||||+----------	------	-----	STP_cntr_05
	| ||| | ||||||||+----------	------	-----	STP_cntr_04
	| ||| | |||||||||+----------	------	-----	STP_cntr_03
	| ||| | ||||||||||+----------	------	-----	STP_cntr_02
	| ||| | |||||||||||+----------	------	-----	STP_cntr_01
	| ||| | ||||||||||||+----------	------	-----	STP_cntr_00
	| ||| | |||||||||||| +----------	------	-----	L00_data_04_L7 Fmt[0]
	| ||| | |||||||||||| |+----------	------	-----	L00_data_03_L7 Type[4]
	| ||| | |||||||||||| || +----------	------	-----	L00_data_02_L7 Type[2]
	| ||| | |||||||||||| || |+----------	------	-----	L00_data_01_L7 Type[1]
	| ||| | |||||||||||| || ||+----------	------	-----	L00_data_00_L7 Type[0]
	| ||| | |||||||||||| || ||| +----------	------	-----	L00_data_07 Code[7]
	| ||| | |||||||||||| || ||| |+----------	------	-----	L00_data_06 Code[6]
	| ||| | |||||||||||| || ||| ||+--------	------	-----	L00_data_05 Code[5]
	| ||| | |||||||||||| || ||| |||+-------	------	-----	L00_data_04 Code[4]
	| ||| | |||||||||||| || ||| ||| +-----	------	-----	L00_data_03 Code[3]
	| ||| | |||||||||||| || ||| ||| |+----	------	-----	L00_data_02 Code[2]
	| ||| | |||||||||||| || ||| ||| ||+---	------	-----	L00_data_01 Code[1]
	| ||| | |||||||||||| || ||| ||| |||+--	------	-----	L00_data_00 Code[0]
	| ||| | |||||||||||| || ||| ||| ||||			
Assert_INTA	1 000 1 0000000001010 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 1 0000000001010 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 1 0000000001010 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 1 0000000001010 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 1 0000000001010 10 100 0010 0100	@white	@blue	De-assert INTA
Deas-sert_INTB	1 000 1 0000000001010 10 100 0010 0101	@white	@blue	De-assert INTB

Table 4-2: PCIEx1_TLP_msg symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Deas-sert_INTC	1 000 1 0000000001010 10 100 0010 0110	@white	@blue	De-assert INTC
Deas-sert_INTD	1 000 1 0000000001010 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Ac-tive_State_Na k	1 000 1 0000000001010 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 1 0000000001010 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 1 0000000001010 10 011 0001 1001	@white	@blue	Broadcast down-stream
PME_TO_Ack	1 000 1 0000000001010 10 101 0001 1011	@white	@blue	Sent upstream by endpoint
ERR_COR	1 000 1 0000000001010 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFA-TAL	1 000 1 0000000001010 10 000 0011 0001	@black	@red	Non-fatal, uncorrect-able error
ERR_FATAL	1 000 1 0000000001010 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 000 1 0000000001010 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_ Limit	1 000 1 0000000001010 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_De-fined_Type0	1 000 1 0000000001010 10 XXX 0111 1110	@white	@blue	Vendor defined type 0
Vendor_De-fined_Type1	1 000 1 0000000001010 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indica-tor_On	1 000 1 0000000001010 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indica-tor_Blink	1 000 1 0000000001010 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indica-tor_Off	1 000 1 0000000001010 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indica-tor_On	1 000 1 0000000001010 10 100 0100 0101	@white	@blue	Power Indicator Com-mand set to 01b
PWR_Indica-tor_Blink	1 000 1 0000000001010 10 100 0100 0111	@white	@blue	Power Indicator Com-mand set to 10b
PWR_Indica-tor_Off	1 000 1 0000000001010 10 100 0100 0100	@white	@blue	Power Indicator Com-mand set to 11b
Attn_But-ton_Pressed	1 000 1 0000000001010 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	1 XX1 X XXXXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	1 X1X X XXXXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	

Table 4-2: PCIEx1_TLP_msg symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
ERROR	1 1XX X XXXXXXXXXXXX XX XXX XXXX XXXX			
--	X XXX X XXXXXXXXXXXX XX XXX XXXX XXXX			

Table 4-3: PCIEx1_TLP_comp_status symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------------------------	------	------	Symbol_Table_Enable
	| +---------------------------	------	------	Rule_viol_2
	| |+--------------------------	------	------	Rule_viol_1
	| ||+-------------------------	------	------	Rule_viol_0
	| ||| +----------------------	------	------	STP_packet_3
	| ||| | +---------------------	------	------	STP_cntr_12
	| ||| | |+--------------------	------	------	STP_cntr_11
	| ||| | ||+-------------------	------	------	STP_cntr_10
	| ||| | |||+------------------	------	------	STP_cntr_09
	| ||| | ||||+-----------------	------	------	STP_cntr_08
	| ||| | |||||+----------------	------	------	STP_cntr_07
	| ||| | ||||||+---------------	------	------	STP_cntr_06
	| ||| | |||||||+--------------	------	------	STP_cntr_05
	| ||| | ||||||||+------------	------	------	STP_cntr_04
	| ||| | |||||||||+----------	------	------	STP_cntr_03
	| ||| | ||||||||||+---------	------	------	STP_cntr_02
	| ||| | |||||||||||+-------	------	------	STP_cntr_01
	| ||| | ||||||||||||+-----	------	------	STP_cntr_00
	| ||| | |||||||||||| +----	------	------	L00_data_04_L6 Type[4]
	| ||| | |||||||||||| |+---	------	------	L00_data_03_L6 Type[3]
	| ||| | |||||||||||| || +-	------	------	L00_data_07 Cmp. Stat.
	| ||| | |||||||||||| || |+	------	------	L00_data_06 Cmp. Stat.
	| ||| | |||||||||||| || ||+	------	------	L00_data_05 Cmp. Stat.
	| ||| | |||||||||||| || |||			

Table 4-3: PCIEx1_TLP_comp_status symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Success-ful_Cmptl	1 000 1 0000000001001 01 000	@white	@blue	Successful Completion
Unsup-ported_Req	1 000 1 0000000001001 01 001	@black	@red	Unsupported Request
Cfg_Req_Rtry_Status	1 000 1 0000000001001 01 010	@black	@red	Configuration Request Retry Status
Complet-er_Abort	1 000 1 0000000001001 01 100	@black	@red	Completer Abort
ERROR	X XX1 X XXXXXXXXXXXXX XX XXX	@black	@red	
ERROR	X X1X X XXXXXXXXXXXXX XX XXX	@black	@red	
ERROR	X 1XX X XXXXXXXXXXXXX XX XXX	@black	@red	
--	X XXX X XXXXXXXXXXXXX XX XXX			

Table 4-4: PCIEx1_DLLP_type symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------	-------	-------	Symbol_Table_Enable
	| +-----------------	-------	-------	Rule_viol_2
	| |+----------------	-------	-------	Rule_viol_1
	| ||+---------------	-------	-------	Rule_viol_0
	| ||| +------------	-------	-------	STP_packet_2
	| ||| | +----------	-------	-------	L00_data_07
	| ||| | |+---------	-------	-------	L00_data_06
	| ||| | ||+--------	-------	-------	L00_data_05
	| ||| | |||+-------	-------	-------	L00_data_04
	| ||| | ||| +------	-------	-------	L00_data_03
	| ||| | ||| | +----	-------	-------	L00_data_02
	| ||| | ||| | |+---	-------	-------	L00_data_01
	| ||| | ||| | ||+--	-------	-------	L00_data_00
	| ||| | ||| | |||			
Ack	1 000 1 0000 0 000	@white	@navy	
Nak	1 000 1 0001 0 000	@white	@navy	
PM_Enter_L1	1 000 1 0010 0 000	@white	@navy	
PM_Enter_L23	1 000 1 0010 0 001	@white	@navy	

Table 4-4: PCIEx1_DLLP_type symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
PM_Active_state_Req_L1	1 000 1 0010 0 011	@white	@navy	
PM_Req_Ack	1 000 1 0010 0 100	@white	@navy	
Vendor_Specific	1 000 1 0011 0 000	@white	@navy	
InitFC1-P	1 000 1 0100 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC1-NP	1 000 1 0101 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC1-Cpl	1 000 1 0110 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-P	1 000 1 1100 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-NP	1 000 1 1101 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-Cpl	1 000 1 1110 0 XXX	@white	@navy	XXX specifies virt. channel
UpdataFC-P	1 000 1 1000 0 XXX	@white	@navy	XXX specifies virt. channel
UpdateFC-NP	1 000 1 1001 0 XXX	@white	@navy	XXX specifies virt. channel
UpdateFC-Cpl	1 000 1 1010 0 XXX	@white	@navy	XXX specifies virt. channel
ERROR	X XX1 X XXXX X XXX	@black	@red	
ERROR	X X1X X XXXX X XXX	@black	@red	
ERROR	X 1XX X XXXX X XXX	@black	@red	
--	X XXX X XXXX X XXX			

PCIEx2 Symbol Tables

Table 4-5: PCIEx2_fmttype symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+--------------------------------	-----	-----	Symbol_Table_Enable
	| +------------------------------	-----	-----	Rule_viol_2
	| |+-----------------------------	-----	-----	Rule_viol_1

Table 4-5: PCIEx2_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||+----------	-----	-----	Rule_viol_0
	| ||| +----------	-----	-----	STP_packet_3
	| ||| | +----------	-----	-----	STP_cntr_11
	| ||| | |+----------	-----	-----	STP_cntr_10
	| ||| | ||+----------	-----	-----	STP_cntr_09
	| ||| | |||+----------	-----	-----	STP_cntr_08
	| ||| | ||||+----------	-----	-----	STP_cntr_07
	| ||| | |||||+----------	-----	-----	STP_cntr_06
	| ||| | ||||||+----------	-----	-----	STP_cntr_05
	| ||| | |||||||+----------	-----	-----	STP_cntr_04
	| ||| | ||||||||+----------	-----	-----	STP_cntr_03
	| ||| | |||||||||+----------	-----	-----	STP_cntr_02
	| ||| | ||||||||||+----------	-----	-----	STP_cntr_01
	| ||| | |||||||||||+----------	-----	-----	STP_cntr_00
	| ||| | ||||||||||| +----------	-----	-----	L01_data_06 Fmt[1]
	| ||| | ||||||||||| | +----------	-----	-----	L01_data_05 Fmt[0]
	| ||| | ||||||||||| | | +----------	-----	-----	L01_data_04 Type[4]
	| ||| | ||||||||||| | | | +----------	-----	-----	L01_data_03 Type[3]
	| ||| | ||||||||||| | | | | +----------	-----	-----	L01_data_02 Type[2]
	| ||| | ||||||||||| | | | | |+----------	-----	-----	L01_data_01 Type[1]
	| ||| | ||||||||||| | | | | ||+----------	-----	-----	L01_data_00 Type[0]
	| ||| | ||||||||||| | | | | |||			
MRd	1 000 1 000000000001 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk	1 000 1 000000000001 0 X 0 0 001	@white	@blue	Mem Read Req. Locked
MWr	1 000 1 000000000001 1 X 0 0 000	@white	@blue	Mem Write Request
IORd	1 000 1 000000000001 0 0 0 0 010	@white	@blue	I/O Read Request
IOWr	1 000 1 000000000001 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0	1 000 1 000000000001 0 0 0 0 100	@white	@blue	Config. Read Type 0
CfgWr0	1 000 1 000000000001 1 0 0 0 100	@white	@blue	Config. Write Type 0
CfgRd1	1 000 1 000000000001 0 0 0 0 101	@white	@blue	Config. Read Type 1
CfgWr1	1 000 1 000000000001 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx	1 000 1 000000000001 0 1 1 0 000	@white	@blue	Msg Req.- Routed to Route complex

Table 4-5: PCIEx2_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Msg-Rtd_by_addr	1 000 1 000000000001 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID	1 000 1 000000000001 0 1 1 0 010	@white	@blue	- Routed by ID
Msg-Broad-cast_fm_rt_cmplx	1 000 1 000000000001 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Lo-cal_term_at_recv	1 000 1 000000000001 0 1 1 0 100	@white	@blue	- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx	1 000 1 000000000001 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved	1 000 1 000000000001 0 1 1 0 XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx	1 000 1 000000000001 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr	1 000 1 000000000001 1 1 1 0 001	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID	1 000 1 000000000001 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx	1 000 1 000000000001 1 1 1 0 011	@white	@blue	- Broadcast from rt complex
MsgD-Lo-cal_term_at_recv	1 000 1 000000000001 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx	1 000 1 000000000001 1 1 1 0 101	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved	1 000 1 000000000001 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS	1 000 1 000000000001 0 1 1 1 XXX	@white	@blue	Msg for Advanced Switching
MsgASD	1 000 1 000000000001 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl	1 000 1 000000000001 0 0 0 1 010	@white	@blue	Completion wo/ data
CplD	1 000 1 000000000001 1 0 0 1 010	@white	@blue	Completion w/data
CplLk	1 000 1 000000000001 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read

Table 4-5: PCIEx2_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
CplDLk	1 000 1 000000000001 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	X XX1 X XXXXXXXXXXX X X X X XXX	@black	@red	
ERROR	X X1X X XXXXXXXXXXX X X X X XXX	@black	@red	
ERROR	X 1XX X XXXXXXXXXXX X X X X XXX	@black	@red	
--	X XXX X XXXXXXXXXXX X X X X XXX			

Table 4-6: PCIEx2_msg symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	------	Symbol_Table_Enable
	| +----------	------	------	Rule_viol_2
	| |+----------	------	------	Rule_viol_1
	| ||+----------	------	------	Rule_viol_0
	| ||| +----------	------	------	STP_packet_3
	| ||| | +----------	------	------	STP_cntr_11
	| ||| | |+----------	------	------	STP_cntr_10
	| ||| | ||+----------	------	------	STP_cntr_09
	| ||| | |||+----------	------	------	STP_cntr_08
	| ||| | ||||+----------	------	------	STP_cntr_07
	| ||| | |||||+----------	------	------	STP_cntr_06
	| ||| | ||||||+----------	------	------	STP_cntr_05
	| ||| | |||||||+----------	------	------	STP_cntr_04
	| ||| | ||||||||+----------	------	------	STP_cntr_03
	| ||| | |||||||||+----------	------	------	STP_cntr_02
	| ||| | ||||||||||+----------	------	------	STP_cntr_01
	| ||| | |||||||||||+----------	------	------	STP_cntr_00
	| ||| | ||||||||||| +----------	------	------	L01_data_04_L4 Fmt[0]
	| ||| | ||||||||||| |+----------	------	------	L01_data_03_L4 Type[4]
	| ||| | ||||||||||| || +----------	------	------	L01_data_02_L4 Type[2]
	| ||| | ||||||||||| || |+----------	------	------	L01_data_01_L4 Type[1]

Table 4-6: PCIEx2_msg symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | |||||||||||| || ||+----------	------	------	L01_data_00_L4 Type[0]
	| ||| | |||||||||||| || ||| +----------	------	------	L00_data_07 Code[7]
	| ||| | |||||||||||| || ||| |+---------	------	------	L00_data_06 Code[6]
	| ||| | |||||||||||| || ||| ||+--------	------	------	L00_data_05 Code[5]
	| ||| | |||||||||||| || ||| |||+-------	------	------	L00_data_04 Code[4]
	| ||| | |||||||||||| || ||| |||| +-----	------	------	L00_data_03 Code[3]
	| ||| | |||||||||||| || ||| |||| |+----	------	------	L00_data_02 Code[2]
	| ||| | |||||||||||| || ||| |||| ||+---	------	------	L00_data_01 Code[1]
	| ||| | |||||||||||| || ||| |||| |||+--	------	------	L00_data_00 Code[0]
	| ||| | |||||||||||| || ||| |||| ||||			
Assert_INTA	1 000 1 000000000101 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 1 000000000101 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 1 000000000101 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 1 000000000101 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 1 000000000101 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 000 1 000000000101 10 100 0010 0101	@white	@blue	De-assert INTB
Deassert_INTC	1 000 1 000000000101 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 000 1 000000000101 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 000 1 000000000101 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 1 000000000101 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 1 000000000101 10 011 0001 1001	@white	@blue	Broadcast downstream
PME_TO_Ack	1 000 1 000000000101 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 000 1 000000000101 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFATAL	1 000 1 000000000101 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 000 1 000000000101 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 000 1 000000000101 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 000 1 000000000101 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_Defined_Type0	1 000 1 000000000101 10 XXX 0111 1110	@white	@blue	Vendor defined type 0
Vendor_Defined_Type1	1 000 1 000000000101 10 XXX 0111 1111	@white	@blue	Vendor defined type 1

Table 4-6: PCIEx2_msg symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Attn_Indicator_On	1 000 1 000000000101 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indicator_Blink	1 000 1 000000000101 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indicator_Off	1 000 1 000000000101 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indicator_On	1 000 1 000000000101 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indicator_Blink	1 000 1 000000000101 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indicator_Off	1 000 1 000000000101 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_Button_Pressed	1 000 1 000000000101 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 X XXXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X X XXXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX X XXXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
--	X XXX X XXXXXXXXXXXX XX XXX XXXX XXXX			

Table 4-7: PCIEx2_comp_status symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------------------------	-------	--------	Symbol_Table_Enable
	| +---------------------------	-------	--------	Rule_viol_2
	| |+--------------------------	-------	--------	Rule_viol_1
	| ||+-------------------------	-------	--------	Rule_viol_0
	| ||| +----------------------	-------	--------	STP_packet_3
	| ||| | +--------------------	-------	--------	STP_cntr_11
	| ||| | |+-------------------	-------	--------	STP_cntr_10
	| ||| | ||+------------------	-------	--------	STP_cntr_09
	| ||| | |||+-----------------	-------	--------	STP_cntr_08
	| ||| | ||||+----------------	-------	--------	STP_cntr_07
	| ||| | |||||+---------------	-------	--------	STP_cntr_06
	| ||| | ||||||+--------------	-------	--------	STP_cntr_05
	| ||| | |||||||+-------------	-------	--------	STP_cntr_04
	| ||| | ||||||||+------------	-------	--------	STP_cntr_03
	| ||| | |||||||||+----------	-------	--------	STP_cntr_02

Table 4-7: PCIEx2_comp_status symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | |||||||||+----------	-------	--------	STP_cntr_01
	| ||| | ||||||||||+----------	-------	--------	STP_cntr_00
	| ||| | ||||||||||| +-------	-------	--------	L01_data_04_LLL Type[4]
	| ||| | ||||||||||| |+------	-------	--------	L01_data_03_LLL Type[3]
	| ||| | ||||||||||| || +----	-------	--------	L01_data_07 Compl. Stat.
	| ||| | ||||||||||| || |+---	-------	--------	L01_data_06 Compl. Stat.
	| ||| | ||||||||||| || ||+--	-------	--------	L01_data_05 Compl. Stat.
	| ||| | ||||||||||| || |||			
Success-ful_Cmptl	1 000 1 000000000100 01 000	@white	@blue	Successful Completion
Unsup-ported_Req	1 000 1 000000000100 01 001	@black	@red	Unsupported Request
Cfg_Req_Rtry_ Status	1 000 1 000000000100 01 010	@black	@red	Configuration Request Retry Status
Complet-er_Abort	1 000 1 000000000100 01 100	@black	@red	Completer Abort
ERROR	X XX1 X XXXXXXXXXXXX XX XXX	@black	@red	
ERROR	X X1X X XXXXXXXXXXXX XX XXX	@black	@red	
ERROR	X 1XX X XXXXXXXXXXXX XX XXX	@black	@red	
--	X XXX X XXXXXXXXXXXX XX XXX			

Table 4-8: PCIEx2_DLLP_type symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+-------------------	------	------	Symbol_Table_Enable
	| +-----------------	------	------	Rule_viol_2
	| |+----------------	------	------	Rule_viol_1
	| ||+---------------	------	------	Rule_viol_0
	| ||| +------------	------	------	STP_packet_3
	| ||| | +----------	------	------	L01_data_07
	| ||| | |+---------	------	------	L01_data_06
	| ||| | ||+--------	------	------	L01_data_05

Table 4-8: PCIEx2_DLLP_type symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | |||+--------	------	------	L01_data_04
	| ||| | ||| +------	------	------	L01_data_03
	| ||| | ||| | +----	------	------	L01_data_02
	| ||| | ||| | |+---	------	------	L01_data_01
	| ||| | ||| | ||+--	------	------	L01_data_00
	| ||| | ||| | |||			
Ack	1 000 1 0000 0 000	@white	@navy	
Nak	1 000 1 0001 0 000	@white	@navy	
PM_Enter_L1	1 000 1 0010 0 000	@white	@navy	
PM_Enter_L23	1 000 1 0010 0 001	@white	@navy	
PM_Active_state_Req_L1	1 000 1 0010 0 011	@white	@navy	
PM_Req_Ack	1 000 1 0010 0 100	@white	@navy	
Vendor_Specific	1 000 1 0011 0 000	@white	@navy	
InitFC1-P	1 000 1 0100 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC1-NP	1 000 1 0101 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC1-Cpl	1 000 1 0110 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-P	1 000 1 1100 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-NP	1 000 1 1101 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-Cpl	1 000 1 1110 0 XXX	@white	@navy	XXX specifies virt. channel
UpdataFC-P	1 000 1 1000 0 XXX	@white	@navy	XXX specifies virt. channel
UpdateFC-NP	1 000 1 1001 0 XXX	@white	@navy	XXX specifies virt. channel
UpdateFC-Cpl	1 000 1 1010 0 XXX	@white	@navy	XXX specifies virt. channel
ERROR	X XX1 X XXXX X XXX	@black	@red	
ERROR	X X1X X XXXX X XXX	@black	@red	
ERROR	X 1XX X XXXX X XXX	@black	@red	
--	X XXX X XXXX X XXX			

PCIEx4 Symbol Tables

Table 4- 9: PCIEx4_fmttype symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------------	------	------	Symbol_Table_Enable
	| +-----------------------------	------	------	Rule_viol_2
	| |+----------------------------	------	------	Rule_viol_1
	| ||+---------------------------	------	------	Rule_viol_0
	| ||| +-------------------------	------	------	STP_packet_2
	| ||| | +-----------------------	------	------	STP_cntr_10
	| ||| | |+----------------------	------	------	STP_cntr_09
	| ||| | ||+---------------------	------	------	STP_cntr_08
	| ||| | |||+--------------------	------	------	STP_cntr_07
	| ||| | ||||+-------------------	------	------	STP_cntr_06
	| ||| | |||||+------------------	------	------	STP_cntr_05
	| ||| | ||||||+-----------------	------	------	STP_cntr_04
	| ||| | |||||||+----------------	------	------	STP_cntr_03
	| ||| | ||||||||+---------------	------	------	STP_cntr_02
	| ||| | |||||||||+--------------	------	------	STP_cntr_01
	| ||| | ||||||||||+-------------	------	------	STP_cntr_00
	| ||| | |||||||||| +-----------	------	------	L03_data_06 Fmt[1]
	| ||| | |||||||||| | +----------	------	------	L03_data_05 Fmt[0]
	| ||| | |||||||||| | | +--------	------	------	L03_data_04 Type[4]
	| ||| | |||||||||| | | | +------	------	------	L03_data_03 Type[3]
	| ||| | |||||||||| | | | | +----	------	------	L03_data_02 Type[2]
	| ||| | |||||||||| | | | | |+---	------	------	L03_data_01 Type[1]
	| ||| | |||||||||| | | | | ||+--	------	------	L03_data_00 Type[0]
	| ||| | |||||||||| | | | | |||			
MRd	1 000 1 00000000000 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk	1 000 1 00000000000 0 X 0 0 001	@white	@blue	Mem Read Req. Locked
MWr	1 000 1 00000000000 1 X 0 0 000	@white	@blue	Mem Write Request
IORd	1 000 1 00000000000 0 0 0 0 010	@white	@blue	I/O Read Request
IOWr	1 000 1 00000000000 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0	1 000 1 00000000000 0 0 0 0 100	@white	@blue	Config. Read Type 0
CfgWr0	1 000 1 00000000000 1 0 0 0 100	@white	@blue	Config. Write Type 0

Table 4-9: PCIEx4_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
CfgRd1	1 000 1 00000000000 0 0 0 0 101	@white	@blue	Config. Read Type 1
CfgWr1	1 000 1 00000000000 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx	1 000 1 00000000000 0 1 1 0 000	@white	@blue	Msg Req.- Routed to Route complex
Msg-Rtd_by_addr	1 000 1 00000000000 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID	1 000 1 00000000000 0 1 1 0 010	@white	@blue	- Routed by ID
Msg-Broad-cast_fm_rt_cmplx	1 000 1 00000000000 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Lo-cal_term_at_recv	1 000 1 00000000000 0 1 1 0 100	@white	@blue	- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx	1 000 1 00000000000 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved	1 000 1 00000000000 0 1 1 0 XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx	1 000 1 00000000000 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr	1 000 1 00000000000 1 1 1 0 001	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID	1 000 1 00000000000 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx	1 000 1 00000000000 1 1 1 0 011	@white	@blue	- Broadcast from rt complex
MsgD-Lo-cal_term_at_recv	1 000 1 00000000000 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx	1 000 1 00000000000 1 1 1 0 101	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved	1 000 1 00000000000 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS	1 000 1 00000000000 0 1 1 1 XXX	@white	@blue	Msg for Advanced Switching
MsgASD	1 000 1 00000000000 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl	1 000 1 00000000000 0 0 0 1 010	@white	@blue	Completion wo/ data

Table 4-9: PCIEx4_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
CplD	1 000 1 00000000000 1 0 0 1 010	@white	@blue	Completion w/data
CplLk	1 000 1 00000000000 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read
CplDLk	1 000 1 00000000000 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	X XX1 X XXXXXXXXXXX X X X X XXX	@black	@red	
ERROR	X X1X X XXXXXXXXXXX X X X X XXX	@black	@red	
ERROR	X 1XX X XXXXXXXXXXX X X X X XXX	@black	@red	
--	X XXX X XXXXXXXXXXX X X X X XXX			

Table 4-10: PCIEx4_msg symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	-----	Symbol_Table_Enable
	| +----------	------	-----	Rule_viol_2
	| |+----------	------	-----	Rule_viol_1
	| ||+----------	------	-----	Rule_viol_0
	| ||| +----------	------	-----	STP_packet_2
	| ||| | +----------	------	-----	STP_cntr_10
	| ||| | |+----------	------	-----	STP_cntr_09
	| ||| | ||+----------	------	-----	STP_cntr_08
	| ||| | |||+----------	------	-----	STP_cntr_07
	| ||| | ||||+----------	------	-----	STP_cntr_06
	| ||| | |||||+----------	------	-----	STP_cntr_05
	| ||| | ||||||+----------	------	-----	STP_cntr_04
	| ||| | |||||||+----------	------	-----	STP_cntr_03
	| ||| | ||||||||+----------	------	-----	STP_cntr_02
	| ||| | |||||||||+----------	------	-----	STP_cntr_01
	| ||| | |||||||||+----------	------	-----	STP_cntr_00
	| ||| | ||||||||| +----------	------	-----	L03_data_04_LL Fmt[0]
	| ||| | ||||||||| |+----------	------	-----	L03_data_03_LL Type[4]
	| ||| | ||||||||| || +----------	------	-----	L03_data_02_LL Type[2]

Table 4-10: PCIEx4_msg symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | |||||||||| || |+----------	------	-----	L03_data_01_LL Type[1]
	| ||| | |||||||||| || ||+----------	------	-----	L03_data_00_LL Type[0]
	| ||| | |||||||||| || ||| +----------	------	-----	L02_data_07 Code[7]
	| ||| | |||||||||| || ||| |+---------	------	-----	L02_data_06 Code[6]
	| ||| | |||||||||| || ||| ||+--------	------	-----	L02_data_05 Code[5]
	| ||| | |||||||||| || ||| |||+-------	------	-----	L02_data_04 Code[4]
	| ||| | |||||||||| || ||| |||| +-----	------	-----	L02_data_03 Code[3]
	| ||| | |||||||||| || ||| |||| |+----	------	-----	L02_data_02 Code[2]
	| ||| | |||||||||| || ||| |||| ||+---	------	-----	L02_data_01 Code[1]
	| ||| | |||||||||| || ||| |||| |||+--	------	-----	L02_data_00 Code[0]
	| ||| | |||||||||| || ||| |||| ||||			
Assert_INTA	1 000 1 00000000010 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 1 00000000010 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 1 00000000010 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 1 00000000010 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 1 00000000010 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 000 1 00000000010 10 100 0010 0101	@white	@blue	De-assert INTB
Deassert_INTC	1 000 1 00000000010 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 000 1 00000000010 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 000 1 00000000010 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 1 00000000010 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 1 00000000010 10 011 0001 1001	@white	@blue	Broadcast downstream
PME_TO_Ack	1 000 1 00000000010 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 000 1 00000000010 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFATAL	1 000 1 00000000010 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 000 1 00000000010 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 000 1 00000000010 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 000 1 00000000010 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_Defined_Type0	1 000 1 00000000010 10 XXX 0111 1110	@white	@blue	Vendor defined type 0

Table 4-10: PCIEx4_msg symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Vendor_Defined_Type1	1 000 1 00000000010 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indicator_On	1 000 1 00000000010 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indicator_Blink	1 000 1 00000000010 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indicator_Off	1 000 1 00000000010 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indicator_On	1 000 1 00000000010 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indicator_Blink	1 000 1 00000000010 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indicator_Off	1 000 1 00000000010 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_Button_Pressed	1 000 1 00000000010 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 X XXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X X XXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX X XXXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
--	X X XXXXXXXXXXX XX XXX XXXX XXXX			

Table 4-11: PCIEx4_comp_status symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+--------------------------	-------	-------	Symbol_Table_Enable
	| +------------------------	-------	-------	Rule_viol_2
	| |+-----------------------	-------	-------	Rule_viol_1
	| ||+----------------------	-------	-------	Rule_viol_1
	| ||| +--------------------	-------	-------	STP_packet_3
	| ||| | +------------------	-------	-------	STP_cntr_10
	| ||| | |+-----------------	-------	-------	STP_cntr_09
	| ||| | ||+----------------	-------	-------	STP_cntr_08
	| ||| | |||+---------------	-------	-------	STP_cntr_07
	| ||| | ||||+--------------	-------	-------	STP_cntr_06
	| ||| | |||||+-------------	-------	-------	STP_cntr_05
	| ||| | ||||||+------------	-------	-------	STP_cntr_04
	| ||| | |||||||+----------	-------	-------	STP_cntr_03
	| ||| | ||||||||+--------	-------	-------	STP_cntr_02

Table 4-11: PCIEx4_comp_status symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | ||||||||+----------	-------	-------	STP_cntr_01
	| ||| | |||||||||+---------	-------	-------	STP_cntr_00
	| ||| | ||||||||| +-------	-------	-------	L03_data_04_LL Type[4]
	| ||| | ||||||||| |+------	-------	-------	L03_data_03_LL Type[3]
	| ||| | ||||||||| || +----	-------	-------	L01_data_07 Compl. Stat.
	| ||| | ||||||||| || |+---	-------	-------	L01_data_06 Compl. Stat.
	| ||| | ||||||||| || ||+--	-------	-------	L01_data_05 Compl. Stat.
	| ||| | ||||||||| || |||			
Success-ful_Cmptl	1 000 1 00000000010 01 000	@white	@blue	Successful Comple-tion
Unsup-ported_Req	1 000 1 00000000010 01 001	@black	@red	Unsupported Re-quest
Cfg_Req_Rtry_Status	1 000 1 00000000010 01 010	@black	@red	Configuration Re-quest Retry Status
Complet-er_Abort	1 000 1 00000000010 01 100	@black	@red	Completer Abort
ERROR	X XX1 X XXXXXXXXXXX XX XXX	@black	@red	
ERROR	X X1X X XXXXXXXXXXX XX XXX	@black	@red	
ERROR	X 1XX X XXXXXXXXXXX XX XXX	@black	@red	
--	X XXX X XXXXXXXXXXX XX XXX			

Table 4-12: PCIEx4_DLLP_type symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+-------------------	-------	-------	Symbol_Table_Enable
	| +-----------------	-------	-------	Rule_viol_2
	| |+----------------	-------	-------	Rule_viol_1
	| ||+---------------	-------	-------	Rule_viol_0
	| ||| +-------------	-------	-------	STP_packet_2
	| ||| | +----------	-------	-------	L01_data_07
	| ||| | |+---------	-------	-------	L01_data_06
	| ||| | ||+--------	-------	-------	L01_data_05
	| ||| | |||+-------	-------	-------	L01_data_04
	| ||| | |||| +-----	-------	-------	L01_data_03
	| ||| | |||| | +---	-------	-------	L01_data_02

Table 4-12: PCIEx4_DLLP_type symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| | |||| | |+---	-------	-------	L01_data_01
	| ||| | |||| | ||+--	-------	-------	L01_data_00
	| ||| | |||| | |||			
Ack	1 000 1 0000 0 000	@white	@navy	
Nak	1 000 1 0001 0 000	@white	@navy	
PM_Enter_L1	1 000 1 0010 0 000	@white	@navy	
PM_Enter_L23	1 000 1 0010 0 001	@white	@navy	
PM_Active_state_Req_L1	1 000 1 0010 0 011	@white	@navy	
PM_Req_Ack	1 000 1 0010 0 100	@white	@navy	
Vendor_Specific	1 000 1 0011 0 000	@white	@navy	
InitFC1-P	1 000 1 0100 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC1-NP	1 000 1 0101 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC1-Cpl	1 000 1 0110 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-P	1 000 1 1100 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-NP	1 000 1 1101 0 XXX	@white	@navy	XXX specifies virt. channel
InitFC2-Cpl	1 000 1 1110 0 XXX	@white	@navy	XXX specifies virt. channel
UpdataFC-P	1 000 1 1000 0 XXX	@white	@navy	XXX specifies virt. channel
UpdateFC-NP	1 000 1 1001 0 XXX	@white	@navy	XXX specifies virt. channel
UpdateFC-Cpl	1 000 1 1010 0 XXX	@white	@navy	XXX specifies virt. channel
ERROR	X XX1 X XXXX X XXX	@black	@red	
ERROR	X X1X X XXXX X XXX	@black	@red	
ERROR	X 1XX X XXXX X XXX	@black	@red	
--	X XXX X XXXX X XXX			

PCIEx8 Symbol Tables

Table 4- 13: PCIEx8_fmttype symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	-----	Symbol_Table_Enable
	| +----------	------	-----	Rule_viol_2
	| |+----------	------	-----	Rule_viol_1
	| ||+----------	------	-----	Rule_viol_0
	| ||| +----------	------	-----	STP_packet_2
	| ||| |+----------	------	-----	STP_packet_1
	| ||| ||+----------	------	-----	STP_packet_0
	| ||| ||| +----------	------	-----	STP_cntr_09
	| ||| ||| |+----------	------	-----	STP_cntr_08
	| ||| ||| ||+----------	------	-----	STP_cntr_07
	| ||| ||| |||+----------	------	-----	STP_cntr_06
	| ||| ||| ||||+----------	------	-----	STP_cntr_05
	| ||| ||| |||||+----------	------	-----	STP_cntr_04
	| ||| ||| ||||||+----------	------	-----	STP_cntr_03
	| ||| ||| |||||||+----------	------	-----	STP_cntr_02
	| ||| ||| ||||||||+----------	------	-----	STP_cntr_01
	| ||| ||| |||||||||+----------	------	-----	STP_cntr_00
	| ||| ||| ||||||||| +----------	------	-----	L03_data_06 Fmt[1]
	| ||| ||| ||||||||| | +----------	------	-----	L03_data_05 Fmt[0]
	| ||| ||| ||||||||| | | +----------	------	-----	L03_data_04 Type[4]
	| ||| ||| ||||||||| | | | +----------	------	-----	L03_data_03 Type[3]
	| ||| ||| ||||||||| | | | | +----------	------	-----	L03_data_02 Type[2]
	| ||| ||| ||||||||| | | | | |+----------	------	-----	L03_data_01 Type[1]
	| ||| ||| ||||||||| | | | | ||+----------	------	-----	L03_data_00 Type[0]
	| ||| ||| ||||||||| | | | | ||| +----------	------	-----	L07_data_06 Fmt[1]
	| ||| ||| ||||||||| | | | | ||| | +----------	------	-----	L07_data_05 Fmt[0]

Table 4-13: PCIEx8_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| ||||||||| | | | | ||| | | +-------	------	-----	L07_data_04 Type[4]
	| ||| ||| ||||||||| | | | | ||| | | | +-----	------	-----	L07_data_03 Type[3]
	| ||| ||| ||||||||| | | | | ||| | | | | +---	------	-----	L07_data_02 Type[2]
	| ||| ||| ||||||||| | | | | ||| | | | | |+--	------	-----	L07_data_01 Type[1]
	| ||| ||| ||||||||| | | | | ||| | | | | ||+-	------	-----	L07_data_00 Type[0]
	| ||| ||| ||||||||| | | | | ||| | | | | |||			
MRd_L0	1 000 100 0000000000 0 X 0 0 000 X X X X XXX	@white	@blue	Mem Read Request
MRdLk_L0	1 000 100 0000000000 0 X 0 0 001 X X X X XXX	@white	@blue	Mem Read Req. Locked
MWr_L0	1 000 100 0000000000 1 X 0 0 000 X X X X XXX	@white	@blue	Mem Write Request
IORd_L0	1 000 100 0000000000 0 0 0 0 010 X X X X XXX	@white	@blue	I/O Read Request
IOWr_L0	1 000 100 0000000000 1 0 0 0 010 X X X X XXX	@white	@blue	I/O Write Request
CfgRd0_L0	1 000 100 0000000000 0 0 0 0 100 X X X X XXX	@white	@blue	Config. Read Type 0
CfgWr0_L0	1 000 100 0000000000 1 0 0 0 100 X X X X XXX	@white	@blue	Config. Write Type 0
CfgRd1_L0	1 000 100 0000000000 0 0 0 0 101 X X X X XXX	@white	@blue	Config. Read Type 1
CfgWr1_L0	1 000 100 0000000000 1 0 0 0 101 X X X X XXX	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx_L0	1 000 100 0000000000 0 1 1 0 000 X X X X XXX	@white	@blue	Msg Req.- Routed to Route complex
Msg-Rtd_by_addr_L0	1 000 100 0000000000 0 1 1 0 001 X X X X XXX	@white	@blue	- Routed by Address
Msg-Rtd_by_ID_L0	1 000 100 0000000000 0 1 1 0 010 X X X X XXX	@white	@blue	- Routed by ID
Msg-Broadcast_fm_rt_cmplx_L0	1 000 100 0000000000 0 1 1 0 011 X X X X XXX	@white	@blue	- Broadcast fm root
Msg-Local_term_at_recv_L0	1 000 100 0000000000 0 1 1 0 100 X X X X XXX	@white	@blue	- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx_L0	1 000 100 0000000000 0 1 1 0 101 X X X X XXX	@white	@blue	- gathered and routed to root complx

Table 4-13: PCIEx8_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Msg-reserved_L0	1 000 100 0000000000 0 1 1 0 XXX X X X X XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx_L0	1 000 100 0000000000 1 1 1 0 000 X X X X XXX	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr_L0	1 000 100 0000000000 1 1 1 0 001 X X X X XXX	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID_L0	1 000 100 0000000000 1 1 1 0 010 X X X X XXX	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx_L0	1 000 100 0000000000 1 1 1 0 011 X X X X XXX	@white	@blue	- Broadcast from rt complex
MsgD-Local_term_at_recv_L0	1 000 100 0000000000 1 1 1 0 100 X X X X XXX	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx_L0	1 000 100 0000000000 1 1 1 0 101 X X X X XXX	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved	1 000 100 0000000000 1 1 1 0 XXX X X X X XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS_L0	1 000 100 0000000000 0 1 1 1 XXX X X X X XXX	@white	@blue	Msg for Advanced Switching
MsgASD_L0	1 000 100 0000000000 1 1 1 1 XXX X X X X XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl_L0	1 000 100 0000000000 0 0 0 1 010 X X X X XXX	@white	@blue	Completion wo/ data
CplD_L0	1 000 100 0000000000 1 0 0 1 010 X X X X XXX	@white	@blue	Completion w/data
CplLk_L0	1 000 100 0000000000 0 0 0 1 011 X X X X XXX	@white	@blue	Compl. Wo/data for locked mem. Read
CplDLk_L0	1 000 100 0000000000 1 0 0 1 011 X X X X XXX	@white	@blue	Compl. W/data for locked mem. Read
MRd_L4	1 000 101 0000000000 X X X X XXX 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk_L4	1 000 101 0000000000 X X X X XXX 0 X 0 0 001	@white	@blue	Mem Read Req. Locked
MWr_L4	1 000 101 0000000000 X X X X XXX 1 X 0 0 000	@white	@blue	Mem Write Request
IORd_L4	1 000 101 0000000000 X X X X XXX 0 0 0 0 010	@white	@blue	I/O Read Request
IOWr_L4	1 000 101 0000000000 X X X X XXX 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0_L4	1 000 101 0000000000 X X X X XXX 0 0 0 0 100	@white	@blue	Config. Read Type 0
CfgWr0_L4	1 000 101 0000000000 X X X X XXX 1 0 0 0 100	@white	@blue	Config. Write Type 0

Table 4-13: PCIEx8_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
CfgRd1_L4	1 000 101 0000000000 X X X X XXX 0 0 0 0 101	@white	@blue	Config. Read Type 1
CfgWr1_L4	1 000 101 0000000000 X X X X XXX 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx_L4	1 000 101 0000000000 X X X X XXX 0 1 1 0 000	@white	@blue	Msg Req.- Routed to Route complex
Msg-Rtd_by_addr_L4	1 000 101 0000000000 X X X X XXX 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID_L4	1 000 101 0000000000 X X X X XXX 0 1 1 0 010	@white	@blue	- Routed by ID
Msg-Broadcast_fm_rt_cmplx_L4	1 000 101 0000000000 X X X X XXX 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Local_term_at_recv_L4	1 000 101 0000000000 X X X X XXX 0 1 1 0 100	@white	@blue	- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx_L4	1 000 101 0000000000 X X X X XXX 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved_L4	1 000 101 0000000000 X X X X XXX 0 1 1 0 XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx_L4	1 000 101 0000000000 X X X X XXX 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr_L4	1 000 101 0000000000 X X X X XXX 1 1 1 0 001	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID_L4	1 000 101 0000000000 X X X X XXX 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx_L4	1 000 101 0000000000 X X X X XXX 1 1 1 0 011	@white	@blue	- Broadcast from rt complex
MsgD-Local_term_at_recv_L4	1 000 101 0000000000 X X X X XXX 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx_L4	1 000 101 0000000000 X X X X XXX 1 1 1 0 101	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved_L4	1 000 101 0000000000 X X X X XXX 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver

Table 4-13: PCIEx8_fmttype symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
MsgAS_L4	1 000 101 0000000000 X X X X XXX 0 1 1 1 XXX	@white	@blue	Msg for Advanced Switching
MsgASD_L4	1 000 101 0000000000 X X X X XXX 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl_L4	1 000 101 0000000000 X X X X XXX 0 0 0 1 010	@white	@blue	Completion wo/ data
CplD_L4	1 000 101 0000000000 X X X X XXX 1 0 0 1 010	@white	@blue	Completion w/data
CplLk_L4	1 000 101 0000000000 X X X X XXX 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read
CplDLk_L4	1 000 101 0000000000 X X X X XXX 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	X XX1 XXX XXXXXXXXXX X X X X XXX X X X X XXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXXX X X X X XXX X X X X XXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXXX X X X X XXX X X X X XXX	@black	@red	
--	X XXX XXX XXXXXXXXXX X X X X XXX X X X X XXX			

Table 4-14: PCIEx8_msg_L0 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	-----	Symbol_Table_Enable
	| +----------	------	-----	Rule_viol_2
	| |+----------	------	-----	Rule_viol_1
	| ||+----------	------	-----	Rule_viol_0
	| ||| +----------	------	-----	STP_packet_3
	| ||| |+----------	------	-----	STP_packet_1
	| ||| ||+----------	------	-----	STP_packet_0
	| ||| ||| +----------	------	-----	STP_cntr_09
	| ||| ||| |+----------	------	-----	STP_cntr_08
	| ||| ||| ||+----------	------	-----	STP_cntr_07
	| ||| ||| |||+----------	------	-----	STP_cntr_06
	| ||| ||| ||||+----------	------	-----	STP_cntr_05
	| ||| ||| |||||+----------	------	-----	STP_cntr_04
	| ||| ||| ||||||+----------	------	-----	STP_cntr_03
	| ||| ||| |||||||+----------	------	-----	STP_cntr_02
	| ||| ||| ||||||||+----------	------	-----	STP_cntr_01
	| ||| ||| |||||||||+----------	------	-----	STP_cntr_00

Table 4-14: PCIEx8_msg_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| ||||||||| +----------	------	-----	L03_data_04_L Fmt[0]
	| ||| ||| ||||||||| |+----------	------	-----	L03_data_03_L Type[4]
	| ||| ||| ||||||||| || +----------	------	-----	L03_data_02_L Type[2]
	| ||| ||| ||||||||| || |+----------	------	-----	L03_data_01_L Type[1]
	| ||| ||| ||||||||| || ||+----------	------	-----	L03_data_00_L Type[0]
	| ||| ||| ||||||||| || ||| +----------	------	-----	L02_data_07 Code[7]
	| ||| ||| ||||||||| || ||| |+----------	------	-----	L02_data_06 Code[6]
	| ||| ||| ||||||||| || ||| ||+----------	------	-----	L02_data_05 Code[5]
	| ||| ||| ||||||||| || ||| |||+----------	------	-----	L02_data_04 Code[4]
	| ||| ||| ||||||||| || ||| ||| +----------	------	-----	L02_data_03 Code[3]
	| ||| ||| ||||||||| || ||| ||| |+----------	------	-----	L02_data_02 Code[2]
	| ||| ||| ||||||||| || ||| ||| ||+----------	------	-----	L02_data_01 Code[1]
	| ||| ||| ||||||||| || ||| ||| |||+----------	------	-----	L02_data_00 Code[0]
	| ||| ||| ||||||||| || ||| ||| ||||			
Assert_INTA	1 000 100 0000000001 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 100 0000000001 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 100 0000000001 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 100 0000000001 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 100 0000000001 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 000 100 0000000001 10 100 0010 0101	@white	@blue	De-assert INTB
Deassert_INTC	1 000 100 0000000001 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 000 100 0000000001 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 000 100 0000000001 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 100 0000000001 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 100 0000000001 10 011 0001 1001	@white	@blue	Broadcast downstream
PME_TO_Ack	1 000 100 0000000001 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 000 100 0000000001 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFATAL	1 000 100 0000000001 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 000 100 0000000001 10 000 0011 0011	@black	@red	Fatal, uncorrectable error

Table 4-14: PCIEx8_msg_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Unlock	1 000 100 0000000001 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 000 100 0000000001 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_Defined_Type0	1 000 100 0000000001 10 XXX 0111 1110	@white	@blue	Vendor defined type 0
Vendor_Defined_Type1	1 000 100 0000000001 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indicator_On	1 000 100 0000000001 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indicator_Blink	1 000 100 0000000001 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indicator_Off	1 000 100 0000000001 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indicator_On	1 000 100 0000000001 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indicator_Blink	1 000 100 0000000001 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indicator_Off	1 000 100 0000000001 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_Button_Pressed	1 000 100 0000000001 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 XXX XXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
--	X XXX XXX XXXXXXXXXX XX XXX XXXX XXXX			

Table 4-15: PCIEx8_msg_L4 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	-----	Symbol_Table_Enable
	| +----------	------	-----	Rule_viol_2
	| |+----------	------	-----	Rule_viol_1
	| ||+----------	------	-----	Rule_viol_0
	| ||| +----------	------	-----	STP_packet_3
	| ||| |+----------	------	-----	STP_packet_1
	| ||| ||+----------	------	-----	STP_packet_0
	| ||| ||| +----------	------	-----	STP_cntr_09

Table 4-15: PCIEx8_msg_L4 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| |+----------	------	-----	STP_cntr_08
	| ||| ||| ||+----------	------	-----	STP_cntr_07
	| ||| ||| |||+----------	------	-----	STP_cntr_06
	| ||| ||| ||||+----------	------	-----	STP_cntr_05
	| ||| ||| |||||+----------	------	-----	STP_cntr_04
	| ||| ||| ||||||+----------	------	-----	STP_cntr_03
	| ||| ||| |||||||+----------	------	-----	STP_cntr_02
	| ||| ||| ||||||||+----------	------	-----	STP_cntr_01
	| ||| ||| |||||||||+----------	------	-----	STP_cntr_00
	| ||| ||| ||||||||| +----------	------	-----	L07_data_04_L Fmt[0]
	| ||| ||| ||||||||| |+----------	------	-----	L07_data_03_L Type[4]
	| ||| ||| ||||||||| || +----------	------	-----	L07_data_02_L Type[2]
	| ||| ||| ||||||||| || |+----------	------	-----	L07_data_01_L Type[1]
	| ||| ||| ||||||||| || ||+----------	------	-----	L07_data_00_L Type[0]
	| ||| ||| ||||||||| || ||| +----------	------	-----	L06_data_07 Code[7]
	| ||| ||| ||||||||| || ||| |+----------	------	-----	L06_data_06 Code[6]
	| ||| ||| ||||||||| || ||| ||+----------	------	-----	L06_data_05 Code[5]
	| ||| ||| ||||||||| || ||| |||+----------	------	-----	L06_data_04 Code[4]
	| ||| ||| ||||||||| || ||| |||| +----------	------	-----	L06_data_03 Code[3]
	| ||| ||| ||||||||| || ||| |||| |+----------	------	-----	L06_data_02 Code[2]
	| ||| ||| ||||||||| || ||| |||| ||+----------	------	-----	L06_data_01 Code[1]
	| ||| ||| ||||||||| || ||| |||| |||+----------	------	-----	L06_data_00 Code[0]
	| ||| ||| ||||||||| || ||| |||| ||||			
Assert_INTA	1 000 101 0000000001 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 101 0000000001 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 101 0000000001 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 101 0000000001 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 101 0000000001 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 000 101 0000000001 10 100 0010 0101	@white	@blue	De-assert INTB
Deassert_INTC	1 000 101 0000000001 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 000 101 0000000001 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 000 101 0000000001 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 101 0000000001 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 101 0000000001 10 011 0001 1001	@white	@blue	Broadcast downstream

Table 4-15: PCIEx8_msg_L4 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
PME_TO_Ack	1 000 101 0000000001 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 000 101 0000000001 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFA-TAL	1 000 101 0000000001 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 000 101 0000000001 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 000 101 0000000001 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 000 101 0000000001 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_De-fined_Type0	1 000 101 0000000001 10 XXX 0111 1110	@white	@blue	Vendor defined type 0
Vendor_De-fined_Type1	1 000 101 0000000001 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indica-tor_On	1 000 101 0000000001 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indica-tor_Blink	1 000 101 0000000001 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indica-tor_Off	1 000 101 0000000001 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indica-tor_On	1 000 101 0000000001 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indica-tor_Blink	1 000 101 0000000001 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indica-tor_Off	1 000 101 0000000001 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_But-ton_Pressed	1 000 101 0000000001 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 XXX XXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXXX XX XXX XXXX XXXX	@black	@red	
--	X XXX XXX XXXXXXXXXX XX XXX XXXX XXXX			

Table 4-16: PCIEx8_comp_status symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	------	Symbol_Table_Enable

Table 4-16: PCIEx8_comp_status symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| +------------------------------	------	------	Rule_viol_2
	| |+------------------------------	------	------	Rule_viol_1
	| ||+------------------------------	------	------	Rule_viol_0
	| ||| +----------------------------	------	------	STP_packet_3
	| ||| |+----------------------------	------	------	STP_packet_1
	| ||| ||+----------------------------	------	------	STP_packet_0
	| ||| ||| +------------------------	------	------	STP_cntr_09
	| ||| ||| |+------------------------	------	------	STP_cntr_08
	| ||| ||| ||+------------------------	------	------	STP_cntr_07
	| ||| ||| |||+----------------------	------	------	STP_cntr_06
	| ||| ||| ||||+---------------------	------	------	STP_cntr_05
	| ||| ||| |||||+-------------------	------	------	STP_cntr_04
	| ||| ||| ||||||+------------------	------	------	STP_cntr_03
	| ||| ||| |||||||+-----------------	------	------	STP_cntr_02
	| ||| ||| ||||||||+----------------	------	------	STP_cntr_01
	| ||| ||| ||||||||+---------------	------	------	STP_cntr_00
	| ||| ||| |||||||| +-------------	------	------	L03_data_04_L Type[4]
	| ||| ||| |||||||| |+------------	------	------	L03_data_03_L Type[3]
	| ||| ||| |||||||| || +----------	------	------	L01_data_07 Compl. Stat.
	| ||| ||| |||||||| || |+----------	------	------	L01_data_06 Compl. Stat.
	| ||| ||| |||||||| || ||+---------	------	------	L01_data_05 Compl. Stat.
	| ||| ||| |||||||| || ||| +-------	------	------	L07_data_04_L Type[4]
	| ||| ||| |||||||| || ||| |+------	------	------	L07_data_03_L Type[3]
	| ||| ||| |||||||| || ||| || +----	------	------	L05_data_07 Compl. Status[2]
	| ||| ||| |||||||| || ||| || |+---	------	------	L05_data_06 Compl. Status[1]
	| ||| ||| |||||||| || ||| || ||+--	------	------	L05_data_05 Compl. Status[0]
	| ||| ||| |||||||| || ||| |||			
Successful_Cmptl_L0	1 000 100 0000000001 01 000 XX XXX	@white	@blue	Successful Completion
Unsupported_Req_L0	1 000 100 0000000001 01 001 XX XXX	@black	@red	Unsupported Request
Cfg_Req_Rtry_Status_L0	1 000 100 0000000001 01 010 XX XXX	@black	@red	Configuration Request Retry Status

Table 4-16: PCIEx8_comp_status symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Complet-er_Abort_L0	1 000 100 0000000001 01 100 XX XXX	@black	@red	Completer Abort
Success-ful_Cmptl_L4	1 000 101 0000000001 XX XXX 01 000	@white	@blue	Successful Completion
Unsup-ported_Req_L4	1 000 101 0000000001 XX XXX 01 001	@black	@red	Unsupported Request
Cfg_Req_Rtry_Status_L4	1 000 101 0000000001 XX XXX 01 010	@black	@red	Configuration Request Retry Status
Complet-er_Abort_L4	1 000 101 0000000001 XX XXX 01 100	@black	@red	Completer Abort
ERROR	X XX1 XXX XXXXXXXXXX XX XXX XX XXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXXX XX XXX XX XXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXXX XX XXX XX XXX	@black	@red	
--	X XXX XXX XXXXXXXXXX XX XXX XX XXX			

Table 4-17: PCIEx8_DLLP_type symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------------	------	------	Symbol_Table_Enable
	| +-----------------------------	------	------	Rule_viol_2
	| |+----------------------------	------	------	Rule_viol_1
	| ||+---------------------------	------	------	Rule_viol_0
	| ||| +-------------------------	------	------	STP_packet_2
	| ||| |+------------------------	------	------	STP_packet_1
	| ||| ||+-----------------------	------	------	STP_packet_0
	| ||| ||| +---------------------	------	------	L01_data_07
	| ||| ||| |+--------------------	------	------	L01_data_06
	| ||| ||| ||+-------------------	------	------	L01_data_05
	| ||| ||| |||+------------------	------	------	L01_data_04
	| ||| ||| |||| +----------------	------	------	L01_data_03
	| ||| ||| |||| | +--------------	------	------	L01_data_02
	| ||| ||| |||| | |+-------------	------	------	L01_data_01
	| ||| ||| |||| | ||+------------	------	------	L01_data_00
	| ||| ||| |||| | ||| +----------	------	------	L05_data_07
	| ||| ||| |||| | ||| |+---------	------	------	L05_data_06
	| ||| ||| |||| | ||| ||+--------	------	------	L05_data_05

Table 4-17: PCIEx8_DLLP_type symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| |||| | ||| |||+--------	------	------	L05_data_04
	| ||| ||| |||| | ||| |||| +------	------	------	L05_data_03
	| ||| ||| |||| | ||| |||| | +----	------	------	L05_data_02
	| ||| ||| |||| | ||| |||| | |+---	------	------	L05_data_01
	| ||| ||| |||| | ||| |||| | ||+--	------	------	L05_data_00
	| ||| ||| |||| | ||| |||| | |||			
Ack_L0	1 000 100 0000 0 000 XXXX X XXX	@white	@navy	
Nak_L0	1 000 100 0001 0 000 XXXX X XXX	@white	@navy	
PM_Enter_L1_L0	1 000 100 0010 0 000 XXXX X XXX	@white	@navy	
PM_Enter_L23_L0	1 000 100 0010 0 001 XXXX X XXX	@white	@navy	
PM_Active_state_Req_L1_L0	1 000 100 0010 0 011 XXXX X XXX	@white	@navy	
PM_Req_Ack_L0	1 000 100 0010 0 100 XXXX X XXX	@white	@navy	
Vendor_Specific_L0	1 000 100 0011 0 000 XXXX X XXX	@white	@navy	
InitFC1-P_L0	1 000 100 0100 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-NP_L0	1 000 100 0101 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-Cpl_L0	1 000 100 0110 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-P_L0	1 000 100 1100 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-NP_L0	1 000 100 1101 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-Cpl_L0	1 000 100 1110 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
UpdataFC-P_L0	1 000 100 1000 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-NP_L0	1 000 100 1001 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-Cpl_L0	1 000 100 1010 0 XXX XXXX X XXX	@white	@navy	XXX specifies virt. Chan.
Ack_L4	1 000 101 XXXX X XXX 0000 0 000	@white	@navy	-
Nak_L4	1 000 101 XXXX X XXX 0001 0 000	@white	@navy	-
PM_Enter_L1_L4	1 000 101 XXXX X XXX 0010 0 000	@white	@navy	-

Table 4-17: PCIEx8_DLLP_type symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
PM_Enter_L23_L4	1 000 101 XXXX X XXX 0010 0 001	@white	@navy	-
PM_Active_state_Req_L1_L4	1 000 101 XXXX X XXX 0010 0 011	@white	@navy	-
PM_Req_Ack_L4	1 000 101 XXXX X XXX 0010 0 100	@white	@navy	-
Vendor_Specific_L4	1 000 101 XXXX X XXX 0011 0 000	@white	@navy	-
InitFC1-P_L4	1 000 101 XXXX X XXX 0100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-NP_L4	1 000 101 XXXX X XXX 0101 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-Cpl_L4	1 000 101 XXXX X XXX 0110 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-P_L4	1 000 101 XXXX X XXX 1100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-NP_L4	1 000 101 XXXX X XXX 1101 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-Cpl_L4	1 000 101 XXXX X XXX 1110 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdataFC-P_L4	1 000 101 XXXX X XXX 1000 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-NP_L4	1 000 101 XXXX X XXX 1001 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-Cpl_L4	1 000 101 XXXX X XXX 1010 0 XXX	@white	@navy	XXX specifies virt. Chan.
ERROR	X XX1 XXX XXXX X XXX XXXX X XXX	@black	@red	-
ERROR	X X1X XXX XXXX X XXX XXXX X XXX	@black	@red	-
ERROR	X 1XX XXX XXXX X XXX XXXX X XXX	@black	@red	-
--	X XXX XXX XXXX X XXX XXXX X XXX	-	-	-

PCIEx16 Symbol Tables

Table 4-18: PCIEx16_fmttype_L0 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------------	------	-----	Symbol_Table_Enable

Table 4-18: PCIEx16_fmttype_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| +----------	------	-----	Rule_viol_2
	| |+----------	------	-----	Rule_viol_1
	| ||+----------	------	-----	Rule_viol_0
	| ||| +----------	------	-----	STP_packet_2
	| ||| |+----------	------	-----	STP_packet_1
	| ||| ||+----------	------	-----	STP_packet_0
	| ||| ||| +----------	------	-----	STP_cntr_08
	| ||| ||| |+----------	------	-----	STP_cntr_07
	| ||| ||| ||+----------	------	-----	STP_cntr_06
	| ||| ||| |||+----------	------	-----	STP_cntr_05
	| ||| ||| ||||+----------	------	-----	STP_cntr_04
	| ||| ||| |||||+----------	------	-----	STP_cntr_03
	| ||| ||| ||||||+----------	------	-----	STP_cntr_02
	| ||| ||| |||||||+----------	------	-----	STP_cntr_01
	| ||| ||| ||||||||+----------	------	-----	STP_cntr_00
	| ||| ||| |||||||| +----------	------	-----	L03_data_06 Fmt[1]
	| ||| ||| |||||||| | +----------	------	-----	L03_data_05 Fmt[0]
	| ||| ||| |||||||| | | +----------	------	-----	L03_data_04 Type[4]
	| ||| ||| |||||||| | | | +----------	------	-----	L03_data_03 Type[3]
	| ||| ||| |||||||| | | | | +----------	------	-----	L03_data_02 Type[2]
	| ||| ||| |||||||| | | | | |+----------	------	-----	L03_data_01 Type[1]
	| ||| ||| |||||||| | | | | ||+----------			
	| ||| ||| |||||||| | | | | |||			
MRd	1 000 100 000000000 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk	1 000 100 000000000 0 X 0 0 001	@white	@blue	Mem Read Req. Locked
MWr	1 000 100 000000000 1 X 0 0 000	@white	@blue	Mem Write Request
IORd	1 000 100 000000000 0 0 0 0 010	@white	@blue	I/O Read Request
IOWr	1 000 100 000000000 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0	1 000 100 000000000 0 0 0 0 100	@white	@blue	Config. Read Type 0
CfgWr0	1 000 100 000000000 1 0 0 0 100	@white	@blue	Config. Write Type 0
CfgRd1	1 000 100 000000000 0 0 0 0 101	@white	@blue	Config. Read Type 1
CfgWr1	1 000 100 000000000 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx	1 000 100 000000000 0 1 1 0 000	@white	@blue	Msg Req.- Routed to Route complex

Table 4-18: PCIEx16_fmttype_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Msg-Rtd_by_addr	1 000 100 000000000 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID	1 000 100 000000000 0 1 1 0 010	@white	@blue	- Routed by ID
Msg-Broad-cast_fm_rt_cmplx	1 000 100 000000000 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Lo-cal_term_at_recv	1 000 100 000000000 0 1 1 0 100	@white	@blue	- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx	1 000 100 000000000 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved	1 000 100 000000000 0 1 1 0 XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx	1 000 100 000000000 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr	1 000 100 000000000 1 1 1 0 001	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID	1 000 100 000000000 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx	1 000 100 000000000 1 1 1 0 011	@white	@blue	- Broadcast from rt complex
MsgD-Lo-cal_term_at_recv	1 000 100 000000000 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx	1 000 100 000000000 1 1 1 0 101	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved	1 000 100 000000000 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS	1 000 100 000000000 0 1 1 1 XXX	@white	@blue	Msg for Advanced Switching
MsgASD	1 000 100 000000000 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl	1 000 100 000000000 0 0 0 1 010	@white	@blue	Completion wo/ data
CplD	1 000 100 000000000 1 0 0 1 010	@white	@blue	Completion w/data
CplLk	1 000 100 000000000 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read

Table 4-18: PCIEx16_fmttype_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
CplDLk	1 000 100 000000000 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	X XX1 XXX XXXXXXXXX X X X X XXX	@black	@red	-
ERROR	X X1X XXX XXXXXXXXX X X X X XXX	@black	@red	-
ERROR	X 1XX XXX XXXXXXXXX X X X X XXX	@black	@red	-
--	X XXX XXX XXXXXXXXX X X X X XXX	-	-	-

Table 4-19: PCIEx16_fmttype_L4 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------------	------	------	Symbol_Table_Enable
	| +-----------------------------	------	------	Rule_viol_2
	| |+----------------------------	------	------	Rule_viol_1
	| ||+---------------------------	------	------	Rule_viol_0
	| ||| +-------------------------	------	------	STP_packet_2
	| ||| |+------------------------	------	------	STP_packet_1
	| ||| ||+-----------------------	------	------	STP_packet_0
	| ||| ||| +---------------------	------	------	STP_cntr_08
	| ||| ||| |+--------------------	------	------	STP_cntr_07
	| ||| ||| ||+-------------------	------	------	STP_cntr_06
	| ||| ||| |||+------------------	------	------	STP_cntr_05
	| ||| ||| ||||+-----------------	------	------	STP_cntr_04
	| ||| ||| |||||+----------------	------	------	STP_cntr_03
	| ||| ||| ||||||+---------------	------	------	STP_cntr_02
	| ||| ||| |||||||+--------------	------	------	STP_cntr_01
	| ||| ||| ||||||||+-------------	------	------	STP_cntr_00
	| ||| ||| |||||||| +------------	------	------	L07_data_06 Fmt[1]
	| ||| ||| |||||||| | +----------	------	------	L07_data_05 Fmt[0]
	| ||| ||| |||||||| | | +--------	------	------	L07_data_04 Type[4]
	| ||| ||| |||||||| | | | +------	------	------	L07_data_03 Type[3]
	| ||| ||| |||||||| | | | | +----	------	------	L07_data_02 Type[2]
	| ||| ||| |||||||| | | | | |+---	------	------	L07_data_01 Type[1]
	| ||| ||| |||||||| | | | | ||+--	------	------	L07_data_00 Type[0]

Table 4-19: PCIEx16_fmttype_L4 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| |||||||| | | | | |||			
MRd	1 000 101 000000000 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk	1 000 101 000000000 0 X 0 0 001	@white	@blue	Mem Read Req. Locked
MWr	1 000 101 000000000 1 X 0 0 000	@white	@blue	Mem Write Request
IORd	1 000 101 000000000 0 0 0 0 010	@white	@blue	I/O Read Request
IOWr	1 000 101 000000000 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0	1 000 101 000000000 0 0 0 0 100	@white	@blue	Config. Read Type 0
CfgWr0	1 000 101 000000000 1 0 0 0 100	@white	@blue	Config. Write Type 0
CfgRd1	1 000 101 000000000 0 0 0 0 101	@white	@blue	Config. Read Type 1
CfgWr1	1 000 101 000000000 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx	1 000 101 000000000 0 1 1 0 000	@white	@blue	Msg Req.- Routed to Route complex
Msg-Rtd_by_addr	1 000 101 000000000 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID	1 000 101 000000000 0 1 1 0 010	@white	@blue	- Routed by ID
Msg-Broadcast_fm_rt_cmplx	1 000 101 000000000 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Local_term_at_recv	1 000 101 000000000 0 1 1 0 100	@white	@blue	- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx	1 000 101 000000000 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved	1 000 101 000000000 0 1 1 0 XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx	1 000 101 000000000 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr	1 000 101 000000000 1 1 1 0 001	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID	1 000 101 000000000 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx	1 000 101 000000000 1 1 1 0 011	@white	@blue	- Broadcast from rt complex

Table 4-19: PCIEx16_fmttype_L4 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
MsgD-Local_term_at_recv	1 000 101 000000000 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx	1 000 101 000000000 1 1 1 0 101	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved	1 000 101 000000000 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS	1 000 101 000000000 0 1 1 1 XXX	@white	@blue	Msg for Advanced Switching
MsgASD	1 000 101 000000000 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl	1 000 101 000000000 0 0 0 1 010	@white	@blue	Completion wo/ data
CplD	1 000 101 000000000 1 0 0 1 010	@white	@blue	Completion w/data
CplLk	1 000 101 000000000 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read
CplDLk	1 000 101 000000000 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	X XX1 XXX XXXXXXXXX X X X X XXX	@black	@red	-
ERROR	X X1X XXX XXXXXXXXX X X X X XXX	@black	@red	-
ERROR	X 1XX XXX XXXXXXXXX X X X X XXX	@black	@red	-
--	X XXX XXX XXXXXXXXX X X X X XXX	-	-	-

Table 4-20: PCIEx16_fmttype_L8 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------------	------	-----	Symbol_Table_Enable_Slave
	| +----------------------------	------	-----	Rule_viol_2_Slave
	| |+----------------------------	------	-----	Rule_viol_1_Slave
	| ||+----------------------------	------	-----	Rule_viol_0_Slave
	| ||| +-------------------------	------	-----	STP_packet_2_Slave
	| ||| |+-------------------------	------	-----	STP_packet_1_Slave
	| ||| ||+-----------------------	------	-----	STP_packet_0_Slave
	| ||| ||| +---------------------	------	-----	STP_cntr_08_Slave
	| ||| ||| |+--------------------	------	-----	STP_cntr_07_Slave

Table 4-20: PCIEx16_fmttype_L8 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| ||+-------------------	------	-----	STP_cntr_06_Slave
	| ||| ||| |||+-------------------	------	-----	STP_cntr_05_Slave
	| ||| ||| ||||+-------------------	------	-----	STP_cntr_04_Slave
	| ||| ||| |||||+-------------------	------	-----	STP_cntr_03_Slave
	| ||| ||| ||||||+-------------------	------	-----	STP_cntr_02_Slave
	| ||| ||| |||||||+-------------------	------	-----	STP_cntr_01_Slave
	| ||| ||| ||||||||+-------------------	------	-----	STP_cntr_00_Slave
	| ||| ||| |||||||| +------------	------	-----	L11_data_06 Fmt[1]
	| ||| ||| |||||||| | +----------	------	-----	L11_data_05 Fmt[0]
	| ||| ||| |||||||| | | +--------	------	-----	L11_data_04 Type[4]
	| ||| ||| |||||||| | | | +------	------	-----	L11_data_03 Type[3]
	| ||| ||| |||||||| | | | | +----	------	-----	L11_data_02 Type[2]
	| ||| ||| |||||||| | | | | |+---	------	-----	L11_data_01 Type[1]
	| ||| ||| |||||||| | | | | ||+--	------	-----	L11_data_00 Type[0]
	| ||| ||| |||||||| | | | | |||			
MRd	1 000 110 000000000 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk	1 000 110 000000000 0 X 0 0 001	@white	@blue	Mem Read Req. Locked
MWr	1 000 110 000000000 1 X 0 0 000	@white	@blue	Mem Write Request
IORd	1 000 110 000000000 0 0 0 0 010	@white	@blue	I/O Read Request
IOWr	1 000 110 000000000 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0	1 000 110 000000000 0 0 0 0 100	@white	@blue	Config. Read Type 0
CfgWr0	1 000 110 000000000 1 0 0 0 100	@white	@blue	Config. Write Type 0
CfgRd1	1 000 110 000000000 0 0 0 0 101	@white	@blue	Config. Read Type 1
CfgWr1	1 000 110 000000000 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx	1 000 110 000000000 0 1 1 0 000	@white	@blue	Msg Req.- Routed to Route complex
Msg-Rtd_by_addr	1 000 110 000000000 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID	1 000 110 000000000 0 1 1 0 010	@white	@blue	- Routed by ID
Msg-Broad-cast_fm_rt_cmplx	1 000 110 000000000 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Lo-cal_term_at_recv	1 000 110 000000000 0 1 1 0 100	@white	@blue	- Local,term at rec

Table 4-20: PCIEx16_fmttype_L8 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Msg-gthrd_rtd_to_rt_cmplx	1 000 110 000000000 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved	1 000 110 000000000 0 1 1 0 XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx	1 000 110 000000000 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr	1 000 110 000000000 1 1 1 0 001	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID	1 000 110 000000000 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx	1 000 110 000000000 1 1 1 0 011	@white	@blue	- Broadcast from rt complex
MsgD-Local_term_at_recv	1 000 110 000000000 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx	1 000 110 000000000 1 1 1 0 101	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved	1 000 110 000000000 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS	1 000 110 000000000 0 1 1 1 XXX	@white	@blue	Msg for Advanced Switching
MsgASD	1 000 110 000000000 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl	1 000 110 000000000 0 0 0 1 010	@white	@blue	Completion wo/ data
CplD	1 000 110 000000000 1 0 0 1 010	@white	@blue	Completion w/data
CplLk	1 000 110 000000000 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read
CplDLk	1 000 110 000000000 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	X XX1 XXX XXXXXXXXX X X X X XXX	@black	@red	-
ERROR	X X1X XXX XXXXXXXXX X X X X XXX	@black	@red	-
ERROR	X 1XX XXX XXXXXXXXX X X X X XXX	@black	@red	-
--	X XXX XXX XXXXXXXXX X X X X XXX	-	-	-

Table 4-21: PCIEx16_fmttype_L12 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------------	------	-----	Symbol_Table_Enable_Slave
	| +----------------------------	------	-----	Rule_viol_2_Slave
	| |+---------------------------	------	-----	Rule_viol_1_Slave
	| ||+--------------------------	------	-----	Rule_viol_0_Slave
	| ||| +------------------------	------	-----	STP_packet_2_Slave
	| ||| |+-----------------------	------	-----	STP_packet_1_Slave
	| ||| ||+----------------------	------	-----	STP_packet_0_Slave
	| ||| ||| +---------------------	------	-----	STP_cntr_08_Slave
	| ||| ||| |+--------------------	------	-----	STP_cntr_07_Slave
	| ||| ||| ||+-------------------	------	-----	STP_cntr_06_Slave
	| ||| ||| |||+------------------	------	-----	STP_cntr_05_Slave
	| ||| ||| ||||+-----------------	------	-----	STP_cntr_04_Slave
	| ||| ||| |||||+----------------	------	-----	STP_cntr_03_Slave
	| ||| ||| ||||||+---------------	------	-----	STP_cntr_02_Slave
	| ||| ||| |||||||+--------------	------	-----	STP_cntr_01_Slave
	| ||| ||| ||||||||+-------------	------	-----	STP_cntr_00_Slave
	| ||| ||| |||||||| +-----------	------	-----	L15_data_06 Fmt[1]
	| ||| ||| |||||||| | +----------	------	-----	L15_data_05 Fmt[0]
	| ||| ||| |||||||| | | +--------	------	-----	L15_data_04 Type[4]
	| ||| ||| |||||||| | | | +------	------	-----	L15_data_03 Type[3]
	| ||| ||| |||||||| | | | | +----	------	-----	L15_data_02 Type[2]
	| ||| ||| |||||||| | | | | |+---	------	-----	L15_data_01 Type[1]
	| ||| ||| |||||||| | | | | ||+--	------	-----	L15_data_00 Type[0]
	| ||| ||| |||||||| | | | | |||			
MRd	1 000 111 000000000 0 X 0 0 000	@white	@blue	Mem Read Request
MRdLk	1 000 111 000000000 0 X 0 0 001	@white	@blue	Mem Read Req. Locked
MWr	1 000 111 000000000 1 X 0 0 000	@white	@blue	Mem Write Request
IORd	1 000 111 000000000 0 0 0 0 010	@white	@blue	I/O Read Request
IOWr	1 000 111 000000000 1 0 0 0 010	@white	@blue	I/O Write Request
CfgRd0	1 000 111 000000000 0 0 0 0 100	@white	@blue	Config. Read Type 0
CfgWr0	1 000 111 000000000 1 0 0 0 100	@white	@blue	Config. Write Type 0
CfgRd1	1 000 111 000000000 0 0 0 0 101	@white	@blue	Config. Read Type 1

Table 4-21: PCIEx16_fmttype_L12 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
CfgWr1	1 000 111 000000000 1 0 0 0 101	@white	@blue	Config. Write Type 0
Msg-Rt_to_rt_cmpx	1 000 111 000000000 0 1 1 0 000	@white	@blue	Msg Req.- Routed to Route complex
Msg-Rtd_by_addr	1 000 111 000000000 0 1 1 0 001	@white	@blue	- Routed by Address
Msg-Rtd_by_ID	1 000 111 000000000 0 1 1 0 010	@white	@blue	- Routed by ID
Msg-Broadcast_fm_rt_cmplx	1 000 111 000000000 0 1 1 0 011	@white	@blue	- Broadcast fm root
Msg-Local_term_at_recv	1 000 111 000000000 0 1 1 0 100	@white	@blue	- Local,term at rec
Msg-gthrd_rtd_to_rt_cmplx	1 000 111 000000000 0 1 1 0 101	@white	@blue	- gathered and routed to root complx
Msg-reserved	1 000 111 000000000 0 1 1 0 XXX	@white	@blue	- Reserved, term at receiver
MsgD-Rdt_to_rt_cmpx	1 000 111 000000000 1 1 1 0 000	@white	@blue	Msg Req. with data - Routed to rt cmpx
MsgD-Rtd_by_addr	1 000 111 000000000 1 1 1 0 001	@white	@blue	- Routed by Address
MsgD-Rtd_by_ID	1 000 111 000000000 1 1 1 0 010	@white	@blue	- Routed by ID
MsgD-Brdcast_fm_rt_cmplx	1 000 111 000000000 1 1 1 0 011	@white	@blue	- Broadcast from rt complex
MsgD-Local_term_at_recv	1 000 111 000000000 1 1 1 0 100	@white	@blue	- Local, term at rec.
MsgD-gthrd_rtd_to_rt_cmplx	1 000 111 000000000 1 1 1 0 101	@white	@blue	- Gathered and routed to rt complex
MsgD-reserved	1 000 111 000000000 1 1 1 0 XXX	@white	@blue	- Reserved. Term. At receiver
MsgAS	1 000 111 000000000 0 1 1 1 XXX	@white	@blue	Msg for Advanced Switching
MsgASD	1 000 111 000000000 1 1 1 1 XXX	@white	@blue	Msg. For Advanced Switching w/data
Cpl	1 000 111 000000000 0 0 0 1 010	@white	@blue	Completion wo/ data
CplD	1 000 111 000000000 1 0 0 1 010	@white	@blue	Completion w/data

Table 4-21: PCIEx16_fmttype_L12 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
CplLk	1 000 111 000000000 0 0 0 1 011	@white	@blue	Compl. Wo/data for locked mem. Read
CplDLk	1 000 111 000000000 1 0 0 1 011	@white	@blue	Compl. W/data for locked mem. Read
ERROR	X XX1 XXX XXXXXXXXX X X X X XXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXX X X X X XXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXX X X X X XXX	@black	@red	
--	X XXX XXX XXXXXXXXX X X X X XXX			

Table 4-22: PCIEx16_msg_L0 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+-----------------------------------	-------	-------	Symbol_Table_Enable
	| +----------------------------------	-------	-------	Rule_viol_2
	| |+---------------------------------	-------	-------	Rule_viol_1
	| ||+--------------------------------	-------	-------	Rule_viol_0
	| ||| +------------------------------	-------	-------	STP_packet_2
	| ||| |+-----------------------------	-------	-------	STP_packet_1
	| ||| ||+----------------------------	-------	-------	STP_packet_0
	| ||| ||| +--------------------------	-------	-------	STP_cntr_08
	| ||| ||| |+-------------------------	-------	-------	STP_cntr_07
	| ||| ||| ||+------------------------	-------	-------	STP_cntr_06
	| ||| ||| |||+-----------------------	-------	-------	STP_cntr_05
	| ||| ||| ||||+----------------------	-------	-------	STP_cntr_04
	| ||| ||| |||||+---------------------	-------	-------	STP_cntr_03
	| ||| ||| ||||||+--------------------	-------	-------	STP_cntr_02
	| ||| ||| |||||||+-------------------	-------	-------	STP_cntr_01
	| ||| ||| ||||||||+------------------	-------	-------	STP_cntr_00
	| ||| ||| |||||||| +-----------------	-------	-------	L03_data_04 Fmt[0]
	| ||| ||| |||||||| |+----------------	-------	-------	L03_data_03 Type[4]
	| ||| ||| |||||||| || +---------------	-------	-------	L03_data_02 Type[2]
	| ||| ||| |||||||| || |+--------------	-------	-------	L03_data_01 Type[1]
	| ||| ||| |||||||| || ||+-------------	-------	-------	L03_data_00 Type[0]
	| ||| ||| |||||||| || ||| +----------	-------	-------	L10_data_07 Code[7]
	| ||| ||| |||||||| || ||| |+---------	-------	-------	L10_data_06 Code[6]

Table 4-22: PCIEx16_msg_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| |||||||| || ||| ||+--------	-------	-------	L10_data_05 Code[5]
	| ||| ||| |||||||| || ||| |||+-------	-------	-------	L10_data_04 Code[4]
	| ||| ||| |||||||| || ||| ||| +-----	-------	-------	L10_data_03 Code[3]
	| ||| ||| |||||||| || ||| ||| |+----	-------	-------	L10_data_02 Code[2]
	| ||| ||| |||||||| || ||| ||| ||+---	-------	-------	L10_data_01 Code[1]
	| ||| ||| |||||||| || ||| ||| |||+--			
	| ||| ||| |||||||| || ||| ||| |||			
Assert_INTA	1 100 000000000 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 100 000000000 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 100 000000000 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 100 000000000 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 100 000000000 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 100 000000000 10 100 0010 0101	@white	@blue	De-assert INTB
Deassert_INTC	1 100 000000000 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 100 000000000 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 100 000000000 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 100 000000000 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 100 000000000 10 011 0001 1001	@white	@blue	Broadcast downstream
PME_TO_Ack	1 100 000000000 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 100 000000000 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFATAL	1 100 000000000 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 100 000000000 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 100 000000000 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 100 000000000 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_Defined_Type0	1 100 000000000 10 XXX 0111 1110	@white	@blue	Vendor defined type 0
Vendor_Defined_Type1	1 100 000000000 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indicator_On	1 100 000000000 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indicator_Blink	1 100 000000000 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b

Table 4-22: PCIEx16_msg_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Attn_Indicator_Off	1 100 000000000 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indicator_On	1 100 000000000 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indicator_Blink	1 100 000000000 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indicator_Off	1 100 000000000 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_Button_Pressed	1 100 000000000 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
--	X XXX XXXXXXXXX XX XXX XXXX XXXX			

Table 4-23: PCIEx16_msg_L4 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	------	Symbol_Table_Enable
	| +----------	------	------	Rule_viol_2
	| |+----------	------	------	Rule_viol_1
	| ||+----------	------	------	Rule_viol_0
	| ||| +----------	------	------	STP_packet_2
	| ||| |+----------	------	------	STP_packet_1
	| ||| ||+----------	------	------	STP_packet_0
	| ||| ||| +----------	------	------	STP_cntr_08
	| ||| ||| |+----------	------	------	STP_cntr_07
	| ||| ||| ||+----------	------	------	STP_cntr_06
	| ||| ||| |||+----------	------	------	STP_cntr_05
	| ||| ||| ||||+----------	------	------	STP_cntr_04
	| ||| ||| |||||+----------	------	------	STP_cntr_03
	| ||| ||| ||||||+----------	------	------	STP_cntr_02
	| ||| ||| |||||||+----------	------	------	STP_cntr_01
	| ||| ||| ||||||||+----------	------	------	STP_cntr_00
	| ||| ||| ||||||||| +----------	------	------	L07_data_04 Fmt[0]
	| ||| ||| ||||||||| |+----------	------	------	L07_data_03 Type[4]

Table 4-23: PCIEx16_msg_L4 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| |||||||| || +--------------	------	------	L07_data_02 Type[2]
	| ||| ||| |||||||| || |+-------------	------	------	L07_data_01 Type[1]
	| ||| ||| |||||||| || ||+------------	------	------	L07_data_00 Type[0]
	| ||| ||| |||||||| || ||| +----------	------	------	L14_data_07 Code[7]
	| ||| ||| |||||||| || ||| |+---------	------	------	L14_data_06 Code[6]
	| ||| ||| |||||||| || ||| ||+--------	------	------	L14_data_05 Code[5]
	| ||| ||| |||||||| || ||| |||+-------	------	------	L14_data_04 Code[4]
	| ||| ||| |||||||| || ||| ||| +-----	------	------	L14_data_03 Code[3]
	| ||| ||| |||||||| || ||| ||| |+----	------	------	L14_data_02 Code[2]
	| ||| ||| |||||||| || ||| ||| ||+---	------	------	L14_data_01 Code[1]
	| ||| ||| |||||||| || ||| ||| |||+--	------	------	L14_data_00 Code[0]
0	| ||| ||| |||||||| || ||| ||| |||			
Assert_INTA	1 000 101 000000000 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 101 000000000 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 101 000000000 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 101 000000000 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 101 000000000 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 000 101 000000000 10 100 0010 0101	@white	@blue	De-assert INTB
Deassert_INTC	1 000 101 000000000 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 000 101 000000000 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 000 101 000000000 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 101 000000000 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 101 000000000 10 011 0001 1001	@white	@blue	Broadcast downstream
PME_TO_Ack	1 000 101 000000000 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 000 101 000000000 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFATAL	1 000 101 000000000 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 000 101 000000000 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 000 101 000000000 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 000 101 000000000 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_Defined_Type0	1 000 101 000000000 10 XXX 0111 1110	@white	@blue	Vendor defined type 0

Table 4-23: PCIEx16_msg_L4 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Vendor_Defined_Type1	1 000 101 000000000 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indicator_On	1 000 101 000000000 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indicator_Blink	1 000 101 000000000 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indicator_Off	1 000 101 000000000 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indicator_On	1 000 101 000000000 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indicator_Blink	1 000 101 000000000 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indicator_Off	1 000 101 000000000 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_Button_Pressed	1 000 101 000000000 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 XXX XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
--	X XXX XXX XXXXXXXXX XX XXX XXXX XXXX			

Table 4-24: PCIEx16_msg_L8 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------------------------	------	------	Symbol_Table_Enable
	| +-------------------------	------	------	Rule_viol_2
	| |+------------------------	------	------	Rule_viol_1
	| ||+-----------------------	------	------	Rule_viol_0
	| ||| +---------------------	------	------	STP_packet_3
	| ||| |+--------------------	------	------	STP_packet_2_L
	| ||| ||+-------------------	------	------	STP_packet_1_L
	| ||| |||+------------------	------	------	STP_packet_0_L
	| ||| |||| +----------------	------	------	L11_data_04 Fmt[0]
	| ||| |||| |+---------------	------	------	L11_data_03 Type[4]
	| ||| |||| || +-------------	------	------	L11_data_02 Type[2]
	| ||| |||| || |+------------	------	------	L11_data_01 Type[1]
	| ||| |||| || ||+----------	------	------	L11_data_00 Type[0]

Table 4-24: PCIEx16_msg_L8 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| |||| || ||| +----------	------	------	L02_data_07 Code[7]
	| ||| |||| || ||| |+---------	------	------	L02_data_06 Code[6]
	| ||| |||| || ||| ||+--------	------	------	L02_data_05 Code[5]
	| ||| |||| || ||| |||+-------	------	------	L02_data_04 Code[4]
	| ||| |||| || ||| |||| +-----	------	------	L02_data_03 Code[3]
	| ||| |||| || ||| |||| |+----	------	------	L02_data_02 Code[2]
	| ||| |||| || ||| |||| ||+---	------	------	L02_data_01 Code[1]
	| ||| |||| || ||| |||| |||+--	------	------	L02_data_00 Code[0]
	| ||| |||| || ||| |||| ||||			
Assert_INTA	1 000 1110 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 1110 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 1110 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 1110 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 1110 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 000 1110 10 100 0010 0101	@white	@blue	De-assert INTB
Deassert_INTC	1 000 1110 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 000 1110 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 000 1110 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 1110 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 1110 10 011 0001 1001	@white	@blue	Broadcast downstream
PME_TO_Ack	1 000 1110 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 000 1110 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFATAL	1 000 1110 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 000 1110 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 000 1110 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 000 1110 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_Defined_Type0	1 000 1110 10 XXX 0111 1110	@white	@blue	Vendor defined type 0
Vendor_Defined_Type1	1 000 1110 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indicator_On	1 000 1110 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b

Table 4-24: PCIEx16_msg_L8 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Attn_Indicator_Blink	1 000 1110 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indicator_Off	1 000 1110 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indicator_On	1 000 1110 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indicator_Blink	1 000 1110 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indicator_Off	1 000 1110 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_Button_Pressed	1 000 1110 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 XXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X XXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX XXXX XX XXX XXXX XXXX	@black	@red	
--	X XXX XXXX XX XXX XXXX XXXX			

Table 4-25: PCIEx16_msg_L12 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	------	-----	Symbol_Table_Enable
	| +----------	------	-----	Rule_viol_3
	| |+----------	------	-----	Rule_viol_1
	| ||+----------	------	-----	Rule_viol_0
	| ||| +----------	------	-----	STP_packet_2
	| ||| |+----------	------	-----	STP_packet_1
	| ||| ||+----------	------	-----	STP_packet_0
	| ||| ||| +----------	------	-----	STP_cntr_08
	| ||| ||| |+----------	------	-----	STP_cntr_07
	| ||| ||| ||+----------	------	-----	STP_cntr_06
	| ||| ||| |||+----------	------	-----	STP_cntr_05
	| ||| ||| ||||+----------	------	-----	STP_cntr_04
	| ||| ||| |||||+----------	------	-----	STP_cntr_03
	| ||| ||| ||||||+----------	------	-----	STP_cntr_02
	| ||| ||| |||||||+----------	------	-----	STP_cntr_01
	| ||| ||| ||||||||+----------	------	-----	STP_cntr_00
	| ||| ||| |||||||| +----------	------	-----	L15_data_04_L Fmt[0]
	| ||| ||| |||||||| |+----------	------	-----	L15_data_03_L Type[4]
	| ||| ||| |||||||| || +----------	------	-----	L15_data_02_L Type[2]
	| ||| ||| |||||||| || |+----------	------	-----	L15_data_01_L Type[1]
	| ||| ||| |||||||| || ||+----------	------	-----	L15_data_00_L Type[0]
	| ||| ||| |||||||| || ||| +----------	------	-----	L06_data_07 Code[7]
	| ||| ||| |||||||| || ||| |+----------	------	-----	L06_data_06 Code[6]
	| ||| ||| |||||||| || ||| ||+----------	------	-----	L06_data_05 Code[5]
	| ||| ||| |||||||| || ||| |||+----------	------	-----	L06_data_04 Code[4]
	| ||| ||| |||||||| || ||| |||| +----------	------	-----	L06_data_03 Code[3]
	| ||| ||| |||||||| || ||| |||| |+----------	------	-----	L06_data_02 Code[2]
	| ||| ||| |||||||| || ||| |||| ||+----------	------	-----	L06_data_01 Code[1]
	| ||| ||| |||||||| || ||| |||| |||+----------	------	-----	L06_data_00 Code[0]
	| ||| ||| |||||||| || ||| |||| ||||			
Assert_INTA	1 000 111 000000001 10 100 0010 0000	@white	@blue	Assert INTA virt. wire
Assert_INTB	1 000 111 000000001 10 100 0010 0001	@white	@blue	Assert INTB virt. Wire
Assert_INTC	1 000 111 000000001 10 100 0010 0010	@white	@blue	Assert INTC virt. Wire
Assert_INTD	1 000 111 000000001 10 100 0010 0011	@white	@blue	Assert INTD virt. wire
Deassert_INTA	1 000 111 000000001 10 100 0010 0100	@white	@blue	De-assert INTA
Deassert_INTB	1 000 111 000000001 10 100 0010 0101	@white	@blue	De-assert INTB

Table 4-25: PCIEx16_msg_L12 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Deassert_INTC	1 000 111 000000001 10 100 0010 0110	@white	@blue	De-assert INTC
Deassert_INTD	1 000 111 000000001 10 100 0010 0111	@white	@blue	De-assert INTD
PM_Active_State_Nak	1 000 111 000000001 10 100 0001 0100	@white	@blue	Terminate at Reciever
PM_PME	1 000 111 000000001 10 000 0001 1000	@white	@blue	Sent upstream by PME-requesting cmpnt.
PME_Turn_Off	1 000 111 000000001 10 011 0001 1001	@white	@blue	Broadcast downstream
PME_TO_Ack	1 000 111 000000001 10 101 0001 1011	@white	@blue	Sent upstream by end-point
ERR_COR	1 000 111 000000001 10 000 0011 0000	@black	@red	Correctable error
ERR_NONFATAL	1 000 111 000000001 10 000 0011 0001	@black	@red	Non-fatal, uncorrectable error
ERR_FATAL	1 000 111 000000001 10 000 0011 0011	@black	@red	Fatal, uncorrectable error
Unlock	1 000 111 000000001 10 011 0000 0000	@white	@blue	Unlock completer
Set_Slot_Pwr_Limit	1 000 111 000000001 10 100 0101 0000	@white	@blue	Set slot power limit in upstream port.
Vendor_Defined_Type0	1 000 111 000000001 10 XXX 0111 1110	@white	@blue	Vendor defined type 0
Vendor_Defined_Type1	1 000 111 000000001 10 XXX 0111 1111	@white	@blue	Vendor defined type 1
Attn_Indicator_On	1 000 111 000000001 10 100 0100 0001	@white	@blue	Attention Indicator Control set to 01b
Attn_Indicator_Blink	1 000 111 000000001 10 100 0100 0011	@white	@blue	Attention Indicator Control set to 10b
Attn_Indicator_Off	1 000 111 000000001 10 100 0100 0000	@white	@blue	Attention Indicator Control set to 11b
PWR_Indicator_On	1 000 111 000000001 10 100 0100 0101	@white	@blue	Power Indicator Command set to 01b
PWR_Indicator_Blink	1 000 111 000000001 10 100 0100 0111	@white	@blue	Power Indicator Command set to 10b
PWR_Indicator_Off	1 000 111 000000001 10 100 0100 0100	@white	@blue	Power Indicator Command set to 11b
Attn_Button_Pressed	1 000 111 000000001 10 100 0100 1000	@white	@blue	Attention button pressed
ERROR	X XX1 XXX XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXX XX XXX XXXX XXXX	@black	@red	
--	X XXX XXX XXXXXXXXX XX XXX XXXX XXXX			

Table 4-26: PCIEx16_comp_status_L8 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+-------------------------	-------	-------	Symbol_Table_Enable
	| +------------------------	-------	-------	Rule_viol_2
	| |+-----------------------	-------	-------	Rule_viol_1
	| ||+----------------------	-------	-------	Rule_viol_0
	| +------------------------	-------	-------	STP_packet_2
	| ||| |+--------------------	-------	-------	STP_packet_1
	| ||| ||+-------------------	-------	-------	STP_packet_0
	| ||| ||| +-----------------	-------	-------	STP_cntr_08
	| ||| ||| |+----------------	-------	-------	STP_cntr_07
	| ||| ||| ||+---------------	-------	-------	STP_cntr_06
	| ||| ||| |||+--------------	-------	-------	STP_cntr_05
	| ||| ||| ||||+-------------	-------	-------	STP_cntr_04
	| ||| ||| |||||+------------	-------	-------	STP_cntr_03
	| ||| ||| ||||||+-----------	-------	-------	STP_cntr_02
	| ||| ||| |||||||+----------	-------	-------	STP_cntr_01
	| ||| ||| ||||||||+---------	-------	-------	STP_cntr_00
	| ||| ||| |||||||| +-------	-------	-------	L03_data[4] Type[4]
	| ||| ||| |||||||| |+------	-------	-------	L03_data[3] Type[3]
	| ||| ||| |||||||| || +----	-------	-------	L09_data[7] Compl. Stat.
	| ||| ||| |||||||| || |+---	-------	-------	L09_data[6] Compl. Stat.
	| ||| ||| |||||||| || ||+--	-------	-------	L09_data[5] Compl. Stat.
	| ||| ||| |||||||| || |||			
Success-ful_Cmptl	1 000 100 000000000 01 000	@white	@blue	Successful Completion
Unsup-ported_Req	1 000 100 000000000 01 001	@black	@red	Unsupported Request
Cfg_Req_Rtry_Status	1 000 100 000000000 01 010	@black	@red	Configuration Request Retry Status
Complet-er_Abort	1 000 100 000000000 01 100	@black	@red	Completer Abort
ERROR	X XX1 XXX XXXXXXXXX XX XXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXX XX XXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXX XX XXX	@black	@red	
--	X XXX XXX XXXXXXXXX XX XXX			

Table 4-27: PCIEx16_comp_status_L4 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+-------------------------	------	-----	Symbol_Table_Enable
	| +------------------------	------	-----	Rule_viol_2
	| |+-----------------------	------	-----	Rule_viol_1
	| ||+----------------------	------	-----	Rule_viol_0
	| ||| +--------------------	------	-----	STP_packet_2
	| ||| |+-------------------	------	-----	STP_packet_1
	| ||| ||+------------------	------	-----	STP_packet_0
	| ||| ||| +----------------	------	-----	STP_cntr_08
	| ||| ||| |+---------------	------	-----	STP_cntr_07
	| ||| ||| ||+--------------	------	-----	STP_cntr_06
	| ||| ||| |||+-------------	------	-----	STP_cntr_05
	| ||| ||| ||||+------------	------	-----	STP_cntr_04
	| ||| ||| |||||+----------	------	-----	STP_cntr_03
	| ||| ||| ||||||+---------	------	-----	STP_cntr_02
	| ||| ||| |||||||+-------	------	-----	STP_cntr_01
	| ||| ||| ||||||||+------	------	-----	STP_cntr_00
	| ||| ||| |||||||| +-----	------	-----	L07_data[4] Type[4]
	| ||| ||| |||||||| |+----	------	-----	L07_data[3] Type[3]
	| ||| ||| |||||||| || +---	------	-----	L013_data[7] Compl. Stat.
	| ||| ||| |||||||| || |+--	------	-----	L013_data[6] Compl. Stat.
	| ||| ||| |||||||| || ||+-	------	-----	L013_data[5] Compl. Stat.
	| ||| ||| |||||||| || |||			
Successful_Cmptl	1 000 101 000000000 01 000	@white	@blue	Successful Completion
Unsupported_Req	1 000 101 000000000 01 001	@black	@red	Unsupported Request
Cfg_Req_Rtry_Status	1 000 101 000000000 01 010	@black	@red	Configuration Request Retry Status
Completer_Abort	1 000 101 000000000 01 100	@black	@red	Completer Abort
ERROR	X XX1 XXX XXXXXXXXX XX XXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXX XX XXX	@black	@red	
ERROR	X 1XX XXX XXXXXXXXX XX XXX	@black	@red	
--	X XXX XXX XXXXXXXXX XX XXX			

Table 4-28: PCIEx16_comp_status_L8 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	-------	-------	Symbol_Table_Enable
	| +----------	-------	-------	Rule_viol_2
	| |+----------	-------	-------	Rule_viol_1
	| ||+----------	-------	-------	Rule_viol_0
	| ||| +----------	-------	-------	STP_packet_3
	| ||| |+----------	-------	-------	STP_packet_2_L
	| ||| ||+----------	-------	-------	STP_packet_1_L
	| ||| |||+----------	-------	-------	STP_packet_0_L
	| ||| ||| +-------	-------	-------	L11_data_04_L Type[4]
	| ||| ||| |+------	-------	-------	L11_data_03_L Type[3]
	| ||| ||| || +----	-------	-------	L01_data_07 Compl. Stat.
	| ||| ||| || |+---	-------	-------	L01_data_06 Compl. Stat.
	| ||| ||| || ||+--	-------	-------	L01_data_05 Compl. Stat.
	| ||| ||| || |||			
Success-ful_Cmptl	1 000 1110 01 000	@white	@blue	Successful Completion
Unsup-ported_Req	1 000 1110 01 001	@black	@red	Unsupported Request
Cfg_Req_Rtry_Status	1 000 1110 01 010	@black	@red	Configuration Request Retry Status
Completer_Abort	1 000 1110 01 100	@black	@red	Completer Abort
ERROR	X XX1 XXXX XX XXX	@black	@red	
ERROR	X X1X XXXX XX XXX	@black	@red	
ERROR	X 1XX XXXX XX XXX	@black	@red	
--	X XXX XXXX XX XXX			

Table 4-29: PCIEx16_comp_status_L12 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------	------	-----	Symbol_Table_Enable
	| +-----------------------	------	-----	Rule_viol_2
	| |+----------------------	------	-----	Rule_viol_1
	| ||+---------------------	------	-----	Rule_viol_0
	| ||| +--------------------	------	-----	STP_packet_3
	| ||| |+-------------------	------	-----	STP_packet_1
	| ||| ||+------------------	------	-----	STP_packet_0
	| ||| ||| +----------------	------	-----	STP_cntr_08
	| ||| ||| |+---------------	------	-----	STP_cntr_07
	| ||| ||| ||+--------------	------	-----	STP_cntr_06
	| ||| ||| |||+-------------	------	-----	STP_cntr_05
	| ||| ||| ||||+------------	------	-----	STP_cntr_04
	| ||| ||| |||||+-----------	------	-----	STP_cntr_03
	| ||| ||| ||||||+----------	------	-----	STP_cntr_02
	| ||| ||| |||||||+---------	------	-----	STP_cntr_01
	| ||| ||| ||||||||+-------	------	-----	STP_cntr_00
	| ||| ||| |||||||| +------	------	-----	L15_data_04_L Type[4]
	| ||| ||| |||||||| |+-----	------	-----	L15_data_03_L Type[3]
	| ||| ||| |||||||| || +---	------	-----	L05_data_07 Compl. Stat.
	| ||| ||| |||||||| || |+--	------	-----	L05_data_06 Compl. Stat.
	| ||| ||| |||||||| || ||+-	------	-----	L05_data_05 Compl. Stat.
	| ||| ||| |||||||| || |||			
Successful_Cmptl	1 000 111 000000001 01 000	@white	@blue	Successful Completion
Unsupported_Req	1 000 111 000000001 01 001	@black	@red	Unsupported Request
Cfg_Req_Rtry_Status	1 000 111 000000001 01 010	@black	@red	Configuration Request Retry Status
Completer_Abort	1 000 111 000000001 01 100	@black	@red	Completer Abort
ERROR	X XX1 XXX XXXXXXXXX XX XXX	@black	@red	
ERROR	X X1X XXX XXXXXXXXX XX XXX	@black	@red	

Table 4-29: PCIEx16_comp_status_L12 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
ERROR	X 1XX XXX XXXXXXXXX XX XXX			
--	X XXX XXX XXXXXXXXX XX XXX			

Table 4-30: PCIEx16_DLLP_type_L0 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+--------------------	--------	-------	Symbol_Table_Enable
	| +-------------------	--------	-------	Rule_viol_2
	| |+------------------	--------	-------	Rule_viol_1
	| ||+-----------------	--------	-------	Rule_viol_0
	| ||| +---------------	--------	-------	STP_packet_2
	| ||| |+--------------	--------	-------	STP_packet_1
	| ||| ||+-------------	--------	-------	STP_packet_0
	| ||| ||| +----------	--------	-------	L01_data_07
	| ||| ||| |+---------	--------	-------	L01_data_06
	| ||| ||| ||+--------	--------	-------	L01_data_05
	| ||| ||| |||+-------	--------	-------	L01_data_04
	| ||| ||| |||| +-----	--------	-------	L01_data_03
	| ||| ||| |||| | +---	--------	-------	L01_data_02
	| ||| ||| |||| | |+--	--------	-------	L01_data_01
	| ||| ||| |||| | ||+-	--------	-------	L01_data_00
	| ||| ||| |||| | |||			
Ack	1 000 100 0000 0 000	@white	@navy	
Nak	1 000 100 0001 0 000	@white	@navy	
PM_Enter_L1	1 000 100 0010 0 000	@white	@navy	
PM_Enter_L23	1 000 100 0010 0 001	@white	@navy	
PM_Active_state_Req_L1	1 000 100 0010 0 011	@white	@navy	
PM_Req_Ack	1 000 100 0010 0 100	@white	@navy	
Vendor_Specific	1 000 100 0011 0 000	@white	@navy	
InitFC1-P	1 000 100 0100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-NP	1 000 100 0101 0 XXX	@white	@navy	XXX specifies virt. Chan.

Table 4-30: PCIEx16_DLLP_type_L0 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
InitFC1-Cpl	1 000 100 0110 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-P	1 000 100 1100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-NP	1 000 100 1101 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-Cpl	1 000 100 1110 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdataFC-P	1 000 100 1000 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-NP	1 000 100 1001 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-Cpl	1 000 100 1010 0 XXX	@white	@navy	XXX specifies virt. Chan.
ERROR	X XX1 XXX XXXX X XXX	@black	@red	
ERROR	X X1X XXX XXXX X XXX	@black	@red	
ERROR	X 1XX XXX XXXX X XXX	@black	@red	
--	X XXX XXX XXXX X XXX			

Table 4-31: PCIEx16_DLLP_type_L4 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+--------------------	------	--------	Symbol_Table_Enable
	| +-------------------	------	--------	Rule_viol_2
	| |+------------------	------	--------	Rule_viol_1
	| ||+-----------------	------	--------	Rule_viol_0
	| ||| +---------------	------	--------	STP_packet_2
	| ||| |+--------------	------	--------	STP_packet_1
	| ||| ||+-------------	------	--------	STP_packet_0
	| ||| ||| +----------	------	--------	L05_data_07
	| ||| ||| |+---------	------	--------	L05_data_06
	| ||| ||| ||+--------	------	--------	L05_data_05
	| ||| ||| |||+-------	------	--------	L05_data_04
	| ||| ||| |||| +-----	------	--------	L05_data_03
	| ||| ||| |||| | +---	------	--------	L05_data_02
	| ||| ||| |||| | |+--	------	--------	L05_data_01
	| ||| ||| |||| | ||+-	------	--------	L05_data_00

Table 4-31: PCIEx16_DLLP_type_L4 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	| ||| ||| |||| | |||			
Ack	1 000 101 0000 0 000	@white	@navy	
Nak	1 000 101 0001 0 000	@white	@navy	
PM_Enter_L1	1 000 101 0010 0 000	@white	@navy	
PM_Enter_L23	1 000 101 0010 0 001	@white	@navy	
PM_Active_state_Req_L1	1 000 101 0010 0 011	@white	@navy	
PM_Req_Ack	1 000 101 0010 0 100	@white	@navy	
Vendor_Specific	1 000 101 0011 0 000	@white	@navy	
InitFC1-P	1 000 101 0100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-NP	1 000 101 0101 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-Cpl	1 000 101 0110 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-P	1 000 101 1100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-NP	1 000 101 1101 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-Cpl	1 000 101 1110 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdataFC-P	1 000 101 1000 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-NP	1 000 101 1001 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-Cpl	1 000 101 1010 0 XXX	@white	@navy	XXX specifies virt. Chan.
ERROR	X XX1 XXX XXXX X XXX	@black	@red	
ERROR	X X1X XXX XXXX X XXX	@black	@red	
ERROR	X 1XX XXX XXXX X XXX	@black	@red	
--	X XXX XXX XXXX X XXX			

Table 4-32: PCIEx16_DLLP_type_L12 symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+--------------------	-------	-------	Symbol_Table_Enable_Slave
	| +-------------------	-------	-------	Rule_viol_2_Slave
	| |+------------------	-------	-------	Rule_viol_1_Slave
	| ||+-----------------	-------	-------	Rule_viol_0_Slave
	| ||| +---------------	-------	-------	STP_packet_2_Slave
	| ||| |+--------------	-------	-------	STP_packet_1_Slave
	| ||| ||+-------------	-------	-------	STP_packet_0_Slave
	| ||| ||| +----------	-------	-------	L09_data_07
	| ||| ||| |+---------	-------	-------	L09_data_06
	| ||| ||| ||+--------	-------	-------	L09_data_05
	| ||| ||| |||+-------	-------	-------	L09_data_04
	| ||| ||| |||| +-----	-------	-------	L09_data_03
	| ||| ||| |||| | +---	-------	-------	L09_data_02
	| ||| ||| |||| | |+--	-------	-------	L09_data_01
	| ||| ||| |||| | ||+-	-------	-------	L09_data_00
	| ||| ||| |||| | |||			
Ack	1 000 110 0000 0 000	@white	@navy	
Nak	1 000 110 0001 0 000	@white	@navy	
PM_Enter_L1	1 000 110 0010 0 000	@white	@navy	
PM_Enter_L23	1 000 110 0010 0 001	@white	@navy	
PM_Active_state_Req_L1	1 000 110 0010 0 011	@white	@navy	
PM_Req_Ack	1 000 110 0010 0 100	@white	@navy	
Vendor_Specific	1 000 110 0011 0 000	@white	@navy	
InitFC1-P	1 000 110 0100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-NP	1 000 110 0101 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC1-Cpl	1 000 110 0110 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-P	1 000 110 1100 0 XXX	@white	@navy	XXX specifies virt. Chan.
InitFC2-NP	1 000 110 1101 0 XXX	@white	@navy	XXX specifies virt. Chan.

Table 4-32: PCIEx16_DLLP_type_L12 symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
InitFC2-Cpl	1 000 110 1110 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdataFC-P	1 000 110 1000 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-NP	1 000 110 1001 0 XXX	@white	@navy	XXX specifies virt. Chan.
UpdateFC-Cpl	1 000 110 1010 0 XXX	@white	@navy	XXX specifies virt. Chan.
ERROR	X XX1 XXX XXXX X XXX	@black	@red	
ERROR	X X1X XXX XXXX X XXX	@black	@red	
ERROR	X 1XX XXX XXXX X XXX	@black	@red	
--	X XXX XXX XXXX X XXX			

Common Symbol Tables

This symbol table is assigned to groups L00_RecErr through L15_RecErr.

Table 4-33: PCIEx_RecErr symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+--------	--------	--------	Ten_bit_mode(_Slave)
	|+-------	--------	--------	LN_data[11]
	|| +-----	--------	--------	LN_data[8]
	|| | +---	--------	--------	LN_data[10]
	|| | |			
Normal_Op_8b	00 0 0			Valid data character received
Overrun-Underrun_8b	00 0 1	@black	@red	Receiver overrun or underrun detected
Normal-Kcode_8b	00 1 0			Receiver operation, valid control char
Normal-Idle_8b	00 1 1			Receiver operation, valid IDLE char
Code_Error_8b	01 0 0	@black	@red	8B10B decoder detected illegal char
Disparity_Error_8b	01 0 1	@black	@red	8B10B decoder detected a disparity err

Table 4-33: PCIEx_RecErr symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
Not_Byte_Sync_8b	01 1 0	@black	@red	The receiver has lost byte alignment
Normal_Op_10b	10 X 0			Valid data character received
Normal-Idle_10b	10 X 1			Receiver operation, valid IDLE char
Not_byte_word_sync_10b	11 X 0	@black	@red	The receiver has lost byte alignment
Overrun-Underrun_10b	11 X 1	@black	@red	Receiver overrun or underrun detected
--	XX X X			

This symbol table is assigned to group rule_viol for each support.

Table 4-34: PCIEx_Rule_viol symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+--------	--------	--------	EDB_Detect
	| +------	--------	--------	LN_data[11]
	| |+-----	--------	--------	LN_data[8]
	| ||+----	--------	--------	LN_data[10]
	| ||			
No_Violation	X 000			No error occurred
STP_sym_time_viol	X 001	@black	@red	Multiple STP in the same symbol time on lane(s) 0,4,8, or 12
SDP_sym_time_viol	X 010	@black	@red	Multiple SDP in the same symbol time on lane(s) 0,4,8, or 12
STP_lane_viol	X 011	@black	@red	STP detected on a lane other than 0,4,8, or 12
SDP_lane_viol	X 100	@black	@red	SDP detected on a lane other than 0,4,8, or 12
STP_SDP_packet_viol	X 101	@black	@red	0Two SDP/STP symbols detected without an intervening END/EDB symbol
SDP_END_viol	X 110	@black	@red	END/EDB was not detected on eighth symbol following SDP
Multiple_viol	X 111	@black	@red	Multiple violations at the same time
--	X XXX			

Table 4-35: PCIEx_Rule_viol symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+----------	--------	--------	STP_packet_3
	| +----------	--------	--------	STP_packet_2
	| |+----------	--------	--------	STP_packet_1
	| ||+----------	--------	--------	STP_packet_0
	| ||| +----------	--------	--------	SDP_packet_3
	| ||| | +----------	--------	--------	SDP_packet_2
	| ||| | |+----------	--------	--------	SDP_packet_1
	| ||| | ||+----------	--------	--------	SDP_packet_0
	| ||| | |||+----------	--------	--------	END_EDB_detect
	| ||| | |||| +----------	--------	--------	Ten_bit_mode
	| ||| | |||| | +----------	--------	--------	Symbol_Table_Enable
	| ||| | |||| | | +----------	--------	--------	Elec_Idle_Flag
	| ||| | |||| | | | +----------	--------	--------	Rule_viol_3
	| ||| | |||| | | | | +----------	--------	--------	TS_Detect
	| ||| | |||| | | | | |+----------	--------	--------	SKP_Detect
	| ||| | |||| | | | | ||+			
	| ||| | |||| | | | | |||			
TLP	1 XXX 0 XXXX 0 1 0 X 000	@green	@silver	TLP
DLLP	0 XXX 1 XXXX 0 1 0 X 000	@purple	@silver	DLLP
TLP-DLLP	1 XXX 1 XXXX 0 1 0 X 000	@yellow	@silver	TLP-DLLP
TS1-TS2	0 XXX 0 XXXX 0 1 0 X 100	@black	@teal	TS1-TS2
SKIP_PKT	0 XXX 0 XXXX 0 1 0 X 010	@black	@olive	SKIP_PKT
FTS	0 XXX 0 XXXX 0 X 1 X XXX	@black	@gray	FTS
ELEC_IDLE	X XXX X XXXX X X 1 X XXX			
--	X XXX X XXXX X X X X XXX			

The name of the symbol table file is PCIEx_10bName. This symbol table will not be assigned to a group by default, but may be assigned to all LXY_data groups, where 'XY' is the particular lane number within the PCIEx link, when the acquistion has acquired 10-bit data. For the full version, see PCIEx_10bName.tsf on the support disc.

Table 4-36: PCIEx_10b Name symbol table

Symbol	Binary pattern	Definition
	+----------	LXY_data_09
	|+----------	LXY_data_08
	||+---------	LXY_data_07
	|||+--------	LXY_data_06
	||||+-------	LXY_data_05
	|||||+------	LXY_data_04
	||||||+-----	LXY_data_03
	|||||||+----	LXY_data_02
	||||||||+---	LXY_data_01
	|||||||||+--	LXY_data_00
	||||||||||	
D0.0-	0010111001	
…	…	
K30.7-	0001011110	
D0.+	1101000110	
…	…	
K30.7+	1110100001	

This symbol table is assigned to all LXY_protocol groups where ‘XY’ is the particular lane number within the PCIEx link. The purpose of this symbol table is to color the groups data and substitute assigned KCODE names for the HEX data. This symbol table contains several hundred symbols so an abridged version is shown below. For the full version, see PCIEx_LN_color.tsf on the support disc.

Table 4-37: PCIEx_LN_color symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
	+------------------------	--------	-------	Ten_bit_mode(_Slave)
	+------------------------	--------	-------	Symbol_Table_Enable(_Slave)
	| +----------------------	--------	-------	TS_Detect(_Slave)
	| | +---------------------	--------	-------	STP_packet_3(_Slave)
	| | | +------------------	--------	-------	SDP_packet_3(_Slave)
	| | | | +-----------------	--------	-------	LXY_data_11
	| | | | | +---------------	--------	-------	LXY_data_10
	| | | | | | +-------------	--------	-------	LXY_data_09
	| | | | | | | +-----------	--------	-------	LXY_data_08
	| | | | | | | | +---------	--------	-------	LXY_data_07
	| | | | | | | |+--------	--------	-------	LXY_data_06
	| | | | | | | ||+-------	--------	-------	LXY_data_05
	| | | | | | | |||+------	--------	-------	LXY_data_04
	| | | | | | | ||||+-----	--------	-------	LXY_data_03
	| | | | | | | |||||+----	--------	-------	LXY_data_02
	| | | | | | | ||||||+---	--------	-------	LXY_data_01
	| | | | | | | |||||||+--	--------	-------	LXY_data_00
	| | | | | | | ||||||||			
COM	0 1 X X X 0 1 0 1 10111100	@white	@maroon	COM symbol
SKP	0 1 X X X 0 0 0 1 00011100	@black	@olive	SKP symbol
FTS	0 1 X X X 0 0 0 1 00111100	@black	@gray	FTS symbol
SDP	0 1 X X X 0 0 0 1 01011100	@white	@navy	SDP symbol
IDL	0 1 X X X 0 0 0 1 01111100	@black	@cyan	IDL symbol
PAD	0 1 X X X 0 0 0 1 11110111	@black	@silver	PAD symbol
STP	0 1 X X X 0 0 0 1 11111011	@white	@blue	STP symbol
END	0 1 X X X 0 0 0 1 11111101	@white	@black	END symbol
EDB	0 1 X X X 0 0 0 1 11111110	@white	@magenta	EDB symbol
19C	0 1 X X X 0 0 0 1 10011100	@white	@purple	Other KCODE
1DC	0 1 X X X 0 0 0 1 11011100	@white	@purple	Other KCODE

Table 4-37: PCIEx_LN_color symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
1FC	0 1 X X X 0 0 0 1 11111100	@white	@purple	Other KCODE
00	0 1 1 X X 0 0 X X 00000000	@black	@teal	Part of training sequence
…	…	…	…	…
FF	0 1 1 X X 0 0 X X 11111111	@black	@teal	Part of training sequence
00	0 1 X 1 X 0 0 X X 00000000	@black	@green	Part of DLLP or TLP packet
…	…	…	…	…
FF	0 1 X 1 X 0 0 X X 11111111	@black	@green	Part of DLLP or TLP packet
00	0 1 X X 1 0 0 X X 00000000	@black	@green	Part of DLLP or TLP packet
…	…	…	…	…
FF	0 1 X X 1 0 0 X X 11111111	@black	@green	Part of DLLP or TLP packet
D0.0-	1 0 X X X 0 0 0 0 10111001			
K30.7+	1 0 X X X 0 0 1 1 10100001			
000	X X X X X X X 0 0 00000000			
…	…			
FFF	X X X X X X 1 1 11111111			

This symbol table is used by the disassembler for coloring Lane data columns and the Link_Details column.

Table 4-38: PCIEx_color symbol table

Symbol	Binary pattern	Text color	Bkgnd color	Definition
PCIEX_CLR_COM	000000	@yellow	@maroon	COMMA symbol
PCIEX_CLR_SKP	000001	@yellow	@olive	SKP symbol
PCIEX_CLR_FTS	000010	@black	@gray	FTS symbol
PCIEX_CLR_SDP	000011	@white	@navy	SDP symbol
PCIEX_CLR_IDL	000100	@black	0@cyan	IDL symbol
PCIEX_CLR_PAD	000101	@black	@silver	PAD symbol
PCIEX_CLR_STP	000110	@white	@blue	STP symbol
PCIEX_CLR_END	000111	@white	@purple	END symbol
PCIEX_CLR_EDB	001000	@white	@red	EDB symbol
PCIEX_CLR_KCODE	001001	@white	@magenta	Other KCODE symbols
PCIEX_CLR_TS	001010	@yellow	@teal	Training sequence
PCIEX_CLR_TLP_HDR	001011	@yellow	@silver	TLP packet header
PCIEX_CLR_DLL	001100	@purple	@silver	TLP data link layer/CRC/SeqNo.

Table 4-38: PCIEx_color symbol table (Cont.)

Symbol	Binary pattern	Text color	Bkgnd color	Definition
PCIEX_CLR_TLP_DATA	001101	@green	@silver	TLP packet data
PCIEX_CLR_DLLP	001110	@yellow	@silver	DLLP packet
PCIEX_CLR_ERROR	001111	@black	@red	0Error in sample
PCIEX_CLR_IDLE	010000			Logical idle
PCIEX_CLR_TLPTYPE	010001	@blue		TLP packet type for Link_Details column
PCIEX_CLR_DLLPTYPE	010010	@navy		DLLP packet type for Link_Details column
PCIEX_CLR_TLPHDRDET	010011	@yellow		TLP header info for Link_Details column
PCIEX_CLR_DLLPHDRDET	010100	@magenta		DLLP info for Link_Details column
PCIEX_CLR_FTSDET	010101	@gray		FTS_packet for Link_Details column
PCIEX_CLR_SKPDET	010110	@olive		SKP packet for Link_Details column
PCIEX_CLR_TSDET	010111	@teal		TS packet for Link_Details column
PCIEX_CLR_TLPDATADET	011000	@green		TLP data for Link_Details column
PCIEX_CLR_DLLDET	011001	@magenta		TLP/DLLP data link layer/CRC/SeqNo. For Link_Details column
PCIEX_CLR_PKTTEXT	011010	@black	@silver	TLP/DLLP data column color for multi-line samples
PCIEX_CLR_10B	011011	@black		10b data color for 10-bit mode

Group Definitions Tables

This section includes:

- Group Definition Tables 4–39 through 4–225
- Explanation of LXX_RecErr group
- Explanation of Rule Violation group

PCIEx1 Group Definitions

Channel groups for PCIEx1 are displayed on the screen in the order shown below.

Table 4-39: Channel groups for PCIEx1

Group name	Description	Display radix	Number of bits
L00_protocol	Lane00 data and qualifiers used to determine color in waveform windows. This group is intended for display only.	SYM	17
L00_data8b	Lane00 8-bit data intended for searching and trigger state machine pattern matching	HEX	8
L00_data10b	L00 10-bit data intended for searching and trigger state machine pattern matching.	HEX	10
Link8bit	8 bit data of all Lanes	HEX	8
STP_cntr	Counter synch to STP detection	DEC	13
TLP_fmttype	Fmt and Type fields for TLP	SYM	25
TLP_msg	Message types for TLP	SYM	31
TLP_comp_status	Completion Status for TLP	SYM	23
DLLP_type	DLLP type	SYM	13
L00_RecErr	Detects Code and Disparity Errors	SYM	4
Error_bits	Data[11] of Lane 00	HEX	1
Rule_viol	Rule checking of logical physical layer	SYM	4
Status	Indicates general status of link	SYM	17

Table 4-40: L00_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L00_data_11
	10	L00_data_10
	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-41: L00_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data8b group is OFF(HEX).

Table 4-42: L00_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data10b group is OFF(HEX).

Table 4-43: Link8bit group assignments

Bit order		PCIExpress signal name
MSB	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the Link8bit group is OFF(HEX).

Table 4-44: STP_cntr group assignments

Bit order		PCIExpress signal name
MSB	12	STP_cntr_12
	11	STP_cntr_11
	10	STP_cntr_10
	9	STP_cntr_09
	8	STP_cntr_08

Table 4-44: STP_cntr group assignments (Cont.)

	7	STP_cntr_07
	6	STP_cntr_06
	5	STP_cntr_05
	4	STP_cntr_04
	3	STP_cntr_03
	2	STP_cntr_02
	1	STP_cntr_01
LSB	0	STP_cntr_00

The default radix of the STP_cntr group is DEC

Table 4-45: TLP_fmttype group assignments

Bit order		PCIExpress signal name
MSB	24	Symbol_Table_Enable
	23	Rule_viol_2
	22	Rule_viol_1
	21	Rule_viol_0
	20	STP_packet_3
	19	STP_cntr_12
	18	STP_cntr_11
	17	STP_cntr_10
	16	STP_cntr_09
	15	STP_cntr_08
	14	STP_cntr_07
	13	STP_cntr_06
	12	STP_cntr_05
	11	STP_cntr_04
	10	STP_cntr_03
	9	STP_cntr_02
	8	STP_cntr_01
	7	STP_cntr_00
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03

Table 4-45: TLP_fmttype group assignments (Cont.)

Bit order		PCIExpress signal name
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the TLP_fmttype group is symbolic. The symbol table file is PCIEx1_TLP_fmttype.

Table 4-46: TLP_msg group assignments

Bit order		PCIExpress signal name
MSB	30	Symbol_Table_Enable
	29	Rule_viol_2
	28	Rule_viol_1
	27	Rule_viol_0
	26	STP_packet_3
	25	STP_cntr_12
	24	STP_cntr_11
	23	STP_cntr_10
	22	STP_cntr_09
	21	STP_cntr_08
	20	STP_cntr_07
	19	STP_cntr_06
	18	STP_cntr_05
	17	STP_cntr_04
	16	STP_cntr_03
	15	STP_cntr_02
	14	STP_cntr_01
	13	STP_cntr_00
	12	L00_data_04_L7
	11	L00_data_03_L7
	10	L00_data_02_L7
	9	L00_data_01_L7
	8	L00_data_00_L7
	7	L00_data_07
	6	L00_data_06

Table 4-46: TLP_msg group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the TLP_msg group is symbolic. The symbol table file is PCIEx1_TLP_msg.

Table 4-47: TLP_comp_status group assignments

Bit order		PCIExpress signal name
MSB	22	Symbol_Table_Enable
	21	Rule_viol_2
	20	Rule_viol_1
	19	Rule_viol_0
	18	STP_packet_3
	17	STP_cntr_12
	16	STP_cntr_12
	15	STP_cntr_12
	14	STP_cntr_12
	13	STP_cntr_12
	12	STP_cntr_12
	11	STP_cntr_12
	10	STP_cntr_12
	9	STP_cntr_12
	8	STP_cntr_12
	7	STP_cntr_12
	6	STP_cntr_12
	5	STP_cntr_12
	4	L00_data_04_L6
	3	L00_data_03_L6
	2	L00_data_07

Table 4-47: TLP_comp_status group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L00_data_06
LSB	0	L00_data_05

The default radix of the TLP_comp_status group is symbolic. The symbol table file is PCIEx1_TLP_comp_status.

Table 4-48: DLLP_type group assignments

Bit order		PCIExpress signal name
MSB	12	Symbol_Table_Enable
	11	Rule_viol_2
	10	Rule_viol_1
	9	Rule_viol_0
	8	SDP_packet_2
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the DLLP_type group is symbolic. The symbol table file is PCIEx1_DLLP_type.

Table 4-49: L00_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L00_data_11
	1	L00_data_08
LSB	0	L00_data_10

Table 4-50: Error_bits group assignments

Bit order		PCIExpress signal name
MSB	0	L00_data_11

The default radix of the Error_bits group is HEX.

Table 4-51: Rule_viol group assignments

Bit order		PCIExpress signal name
MSB	3	EDB_Detect
	2	Rule_viol_2
	1	Rule_viol_1
LSB	0	Rule_viol_0

The default radix of the Rule_viol group is symbolic. The symbol table file is PCIEx1_Rule_viol.

Table 4-52: Status group assignments

Bit order		PCIExpress signal name
MSB	15	STP_packet_3
	14	STP_packet_2
	13	STP_packet_1
	12	STP_packet_0
	11	SDP_packet_3
	10	SDP_packet_2
	9	SDP_packet_1
	8	SDP_packet_0
	7	END_EDB_Detect
	6	Ten_bit_mode
	5	Symbol_Table_Enable
	4	Elect_ Idle_Flag
	3	Rule_viol_3
	2	TS_Detect
	1	SKP_Detect
LSB	0	FTS_Detect

The default radix of the Status group is symbolic. The symbol table file is PCIEx_Status.

PCIEx2 Group Definitions

Channel groups for PCIEx2 are displayed on the screen in the order shown below.

Table 4-53: Channel groups for PCIEx2

Group name	Description	Display radix	Number of bits
L00_protocol – L01_protocol	Lane00 – L01 data and qualifiers used to determine color in waveform windows. This group is intended for display only.	SYM	17 ea
L00_data8b – L01_data8b	Lane00 – Lane01 8-bit data intended for searching and trigger state machine pattern matching	HEX	8 ea
L00_data10b – L01_data10b	Land00 – Lane01 10-bit data intended for searching and trigger state machine pattern matching	HEX	10 ea
Link8bit	8 bit data of all Lanes	HEX	16
STP_cntr	Counter synch to STP detection	DEC	12
TLP_fmttype	Fmt and Type fields for TLP	SYM	24
TLP_msg	Message types for TLP	SYM	30
TLP_comp_status	Completion Status for TLP	SYM	22
DLLP_type	DLLP type	SYM	13
L00_RecErr – L01_RecErr	Detects Code and Disparity Errors	SYM	4 ea
Error_bits	Data[11] of all Lanes	HEX	2
Rule_viol	Rule checking of logical physical layer	SYM	4
Status	Indicates general status of link	SYM	17

Table 4-54: L00_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L00_data_11
	10	L00_data_10

Table 4-54: L00_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-55: L01_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L01_data_11
	10	L01_data_10
	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-56: L00_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data8b group is OFF(HEX).

Table 4-57: L01_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data8b group is OFF(HEX).

Table 4-58: L00_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06

Table 4-58: L00_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data10b group is OFF(HEX).

Table 4-59: L01_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data10b group is OFF(HEX).

Table 4-60: Link8bit group assignments

Bit order		PCIExpress signal name
MSB	15:12	L01_data_07, L01_data_06, L01_data_05, L01_data_04
	11:8	L01_data_03, L01_data_02, L01_data_01, L01_data_00
	7:5	L00_data_07, L00_data_06, L00_data_05, L00_data_04
LSB	4:0	L00_data_03, L00_data_02, L00_data_01, L00_data_00

The default radix of the Link8bit group is HEX.

Table 4-61: STP_cntr group assignments

Bit order		PCIExpress signal name
MSB	11	STP_cntr_11
	10	STP_cntr_10
	9	STP_cntr_09
	8	STP_cntr_08
	7	STP_cntr_07
	6	STP_cntr_06
	5	STP_cntr_05
	4	STP_cntr_04
	3	STP_cntr_03
	2	STP_cntr_02
	1	STP_cntr_01
LSB	0	STP_cntr_00

The default radix of the STP_cntr group is DEC

Table 4-62: TLP_fmttype group assignments

Bit order		PCIExpress signal name
MSB	23	Symbol_Table_Enable
	22	Rule_viol_2
	21	Rule_viol_1
	20	Rule_viol_0
	19	STP_packet_3
	18	STP_cntr_11
	17	STP_cntr_10
	16	STP_cntr_09
	15	STP_cntr_08
	14	STP_cntr_07
	13	STP_cntr_06
	12	STP_cntr_05
	11	STP_cntr_04
	10	STP_cntr_03
	9	STP_cntr_02

Table 4-62: TLP_fmttype group assignments (Cont.)

Bit order		PCIExpress signal name
	8	STP_cntr_01
	7	STP_cntr_00
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the TLP_fmttype group is symbolic. The symbol table file is PCIEx2_TLP_fmttype.

Table 4-63: TLP_msg group assignments

Bit order		PCIExpress signal name
MSB	29	Symbol_Table_Enable
	28	Rule_viol_2
	27	Rule_viol_1
	26	Rule_viol_0
	25	STP_packet_3
	24	STP_cntr_11
	23	STP_cntr_10
	22	STP_cntr_09
	21	STP_cntr_08
	20	STP_cntr_07
	19	STP_cntr_06
	18	STP_cntr_05
	17	STP_cntr_04
	16	STP_cntr_03
	15	STP_cntr_02
	14	STP_cntr_01
	13	STP_cntr_00
	12	L01_data_04_L4
	11	L01_data_03_L4

Table 4-63: TLP_msg group assignments (Cont.)

Bit order		PCIExpress signal name
	10	L01_data_02_L4
	9	L01_data_01_L4
	8	L01_data_00_L4
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the TLP_msg group is symbolic. The symbol table file is PCIEx2_TLP_msg.

Table 4-64: TLP_comp_status group assignments

Bit order		PCIExpress signal name
MSB	21	Symbol_Table_Enable
	20	Rule_viol_2
	19	Rule_viol_1
	18	Rule_viol_0
	17	STP_packet_3
	16	STP_cntr_11
	15	STP_cntr_10
	14	STP_cntr_09
	13	STP_cntr_08
	12	STP_cntr_07
	11	STP_cntr_06
	10	STP_cntr_05
	9	STP_cntr_04
	8	STP_cntr_03
	7	STP_cntr_02
	6	STP_cntr_01
	5	STP_cntr_00

Table 4-64: TLP_comp_status group assignments (Cont.)

Bit order		PCIExpress signal name
	4	L01_data_04_LLL
	3	L01_data_03_LLL
	2	L01_data_07
	1	L01_data_06
LSB	0	L01_data_05

The default radix of the TLP_comp_status group is symbolic. The symbol table file is PCIEx2_TLP_comp_status

Table 4-65: DLLP_type group assignments

Bit order		PCIExpress signal name
MSB	12	Symbol_Table_Enable
	11	Rule_viol_2
	10	Rule_viol_1
	9	Rule_viol_0
	8	SDP_packet_2
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the DLLP_type group is symbolic. The symbol table file is PCIEx2_DLLP_type.

Table 4-66: L00_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L00_data_11
	1	L00_data_08
LSB	0	L00_data_10

The default radix of the L00_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-67: L01_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L01_data_11
	1	L01_data_08
LSB	0	L01_data_10

The default radix of the L01_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-68: Error_bits group assignments

Bit order		PCIExpress signal name
MSB	1	L01_data_11
LSB	0	L00_data_11

The default radix of the Error_bits group is HEX.

Table 4-69: Rule_viol group assignments

Bit order		PCIExpress signal name
MSB	3	EDB_Detect
	2	Rule_viol_2
	1	Rule_viol_1
LSB	0	Rule_viol_0

The default radix of the Rule_viol group is symbolic. The symbol table file is PCIEx1_Rule_viol.

Table 4-70: Status group assignments

Bit order		PCIExpress signal name
MSB	15	STP_packet_3
	14	STP_packet_2
	13	STP_packet_1
	12	STP_packet_0
	11	SDP_packet_3
	10	SDP_packet_2
	9	SDP_packet_1
	8	SDP_packet_0
	7	END_EDB_Detect
	6	Ten_bit_mode
	5	Symbol_Table_Enable
	4	Elect_Idle_Flag
	3	Rule_viol_3
	2	TS_Detect
	1	SKP_Detect
LSB	0	FTS_Detect

The default radix of the Status group is symbolic. The symbol table file is PCIEx1_Status.

PCIEx4 Group Definitions

Channel groups for PCIEx4 are displayed on the screen in the order shown below.

Table 4-71: Channel groups for PCIEx4

Group name	Description	Display radix	Number of bits
L00_protocol – L03_protocol	Lane00 – L01 data and qualifiers used to determine color in waveform windows. This group is intended for display only.	SYM	17 ea
L00_data8b – L03_data8b	Lane00 – Lane01 8-bit data intended for searching and trigger state machine pattern matching.	HEX	8 ea
L00_data10b – L03_data10b	Lane00 – Lane01 10-bit data intended for searching and trigger state machine pattern matching.	HEX	10 ea
Link8bit	8 bit data of all Lanes	HEX	32
STP_cntr	Counter synch to STP detection	DEC	11
TLP_fmttype	Fmt and Type fields for TLP	SYM	23
TLP_msg	Message types for TLP	SYM	29
TLP_comp_status	Completion Status for TLP	SYM	21
DLLP_type	DLLP type	SYM	13
L00_RecErr – L03_RecErr	Detects Code and Disparity Errors	SYM	4 ea
Error_bits	Data[11] of all Lanes	HEX	4
Rule_viol	Rule checking of logical physical layer	SYM	4
Status	Indicates general status of link	SYM	16

Table 4-72: L00_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L00_data_11
	10	L00_data_10

Table 4-72: L00_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-73: L01_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L01_data_11
	10	L01_data_10
	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-74: L02_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L02_data_11
	10	L02_data_10
	9	L02_data_09
	8	L02_data_08
	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_protocol group is symbolic. The symbol table file is PCIEx_LN_color

Table 4-75: L03_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L03_data_11
	10	L03_data_10
	9	L03_data_09

Table 4-75: L03_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	8	L03_data_08
	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-76: L00_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data8b group is OFF(HEX).

Table 4-77: L01_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03

Table 4-77: L01_data8b group assignments (Cont.)

Bit order		PCIExpress signal name
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data8b group is OFF(HEX).

Table 4-78: L02_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_data8b group is OFF(HEX).

Table 4-79: L03_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_data8b group is OFF(HEX).

Table 4-80: L00_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data10b group is OFF(HEX).

Table 4-81: L01_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data10b group is OFF(HEX).

Table 4-82: L02_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L02_data_09
	8	L02_data_08
	7	L02_data_07

Table 4-82: L02_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_data10b group is OFF(HEX).

Table 4-83: L03_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L03_data_09
	8	L03_data_08
	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_data10b group is OFF(HEX).

Table 4-84: Link8bit group assignments

Bit order		PCIExpress signal name
MSB	31:28	L03_data_07, L03_data_06, L03_data_05, L03_data_04
	27:24	L03_data_03, L03_data_02, L03_data_01, L03_data_00
	23:20	L02_data_07, L02_data_06, L02_data_05, L02_data_04

Table 4-84: Link8bit group assignments (Cont.)

Bit order		PCIExpress signal name
	19:16	L02_data_03, L02_data_02, L02_data_01, L02_data_00
	15:12	L01_data_07, L01_data_06, L01_data_05, L01_data_04
	11:8	L01_data_03, L01_data_02, L01_data_01, L01_data_00
	7:4	L00_data_07, L00_data_06, L00_data_05, L00_data_04
LSB	3:0	L00_data_03, L00_data_02, L00_data_01, L00_data_00

The default radix of the Link8bit group is HEX.

Table 4-85: STP_cntr group assignments

Bit order		PCIExpress signal name
MSB	10	STP_cntr_10
	9	STP_cntr_09
	8	STP_cntr_08
	7	STP_cntr_07
	6	STP_cntr_06
	5	STP_cntr_05
	4	STP_cntr_04
	3	STP_cntr_03
	2	STP_cntr_02
	1	STP_cntr_01
LSB	0	STP_cntr_00

The default radix of the STP_cntr group is DEC

Table 4-86: TLP_fmttype group assignments

Bit order		PCIExpress signal name
MSB	22	Symbol_Table_Enable
	21	Rule_viol_2
	20	Rule_viol_1

Table 4-86: TLP_fmttype group assignments (Cont.)

Bit order		PCIExpress signal name
	19	Rule_viol_0
	18	STP_packet_2
	17	STP_cntr_10
	16	STP_cntr_09
	15	STP_cntr_08
	14	STP_cntr_07
	13	STP_cntr_06
	12	STP_cntr_05
	11	STP_cntr_04
	10	STP_cntr_03
	9	STP_cntr_02
	8	STP_cntr_01
	7	STP_cntr_00
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the TLP_fmttype group is symbolic. The symbol table file is PCIEx4_TLP_fmttype.

Table 4-87: TLP_msg group assignments

Bit order		PCIExpress signal name
MSB	28	Symbol_Table_Enable
	27	Rule_viol_2
	26	Rule_viol_1
	25	Rule_viol_0
	24	STP_packet_3
	23	STP_cntr_10
	22	STP_cntr_10
	21	STP_cntr_10

Table 4-87: TLP_msg group assignments (Cont.)

Bit order		PCIExpress signal name
	20	STP_cntr_10
	19	STP_cntr_10
	18	STP_cntr_10
	17	STP_cntr_10
	16	STP_cntr_10
	15	STP_cntr_10
	14	STP_cntr_10
	13	STP_cntr_10
	12	L03_data_04_LL
	11	L03_data_03_LL
	10	L03_data_02_LL
	9	L03_data_01_LL
	8	L03_data_00_LL
	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the TLP_msg group is symbolic. The symbol table file is PCIEx4_TLP_msg.

Table 4-88: TLP_comp_status group assignments

Bit order		PCIExpress signal name
MSB	20	Symbol_Table_Enable
	19	Rule_viol_2
	18	Rule_viol_1
	17	Rule_viol_0
	16	STP_packet_3
	15	STP_cntr_10
	14	STP_cntr_09

Table 4-88: TLP_comp_status group assignments (Cont.)

Bit order		PCIExpress signal name
	13	STP_cntr_08
	12	STP_cntr_07
	11	STP_cntr_06
	10	STP_cntr_05
	9	STP_cntr_04
	8	STP_cntr_03
	7	STP_cntr_02
	6	STP_cntr_01
	5	STP_cntr_00
	4	L03_data_04_LL
	3	L03_data_03_LL
	2	L01_data_07
	1	L01_data_06
LSB	0	L01_data_05

The default radix of the TLP_comp_status group is symbolic. The symbol table file is PCIEx4_TLP_comp_status.

Table 4-89: DLLP_type group assignments

Bit order		PCIExpress signal name
MSB	12	Symbol_Table_Enable
	11	Rule_viol_2
	10	Rule_viol_1
	9	Rule_viol_0
	8	SDP_packet_2
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the DLLP_type group is symbolic. The symbol table file is PCIEx4_DLLP_type.

Table 4-90: L00_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L00_data_11
	1	L00_data_08
LSB	0	L00_data_10

The default radix of the L00_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-91: L01_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L01_data_11
	1	L01_data_08
LSB	0	L01_data_10

The default radix of the L01_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-92: L02_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L02_data_11
	1	L02_data_08
LSB	0	L02_data_10

The default radix of the L02_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-93: L03_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L03_data_11
	1	L03_data_08
LSB	0	L03_data_10

The default radix of the L03_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-94: Error_bits group assignments

Bit order		PCIExpress signal name
MSB	3	L03_data_11
0	2	L02_data_11
0	1	L01_data_11
LSB	0	L00_data_11

The default radix of the Error_bits group is HEX.

Table 4-95: Rule_viol group assignments

Bit order		PCIExpress signal name
MSB	3	EDB_Detect
	2	Rule_viol_2
	1	Rule_viol_1
LSB	0	Rule_viol_0

The default radix of the Rule_viol group is symbolic. The symbol table file is PCIEx_Rule_viol.

Table 4-96: Status group assignments

Bit order		PCIExpress signal name
MSB	15	STP_packet_3
	14	STP_packet_2
	13	STP_packet_1

Table 4-96: Status group assignments (Cont.)

Bit order		PCIExpress signal name
	12	STP_packet_0
	11	SDP_packet_3
	10	SDP_packet_2
	9	SDP_packet_1
	8	SDP_packet_0
	7	END_EDB_Detect
	6	Ten_bit_mode
	5	Symbol_Table_Enable
	4	Elect_Idle_Flag
	3	Rule_viol_3
	2	TS_Detect
	1	SKP_Detect
LSB	0	FTS_Detect

The default radix of the Status group is symbolic. The symbol table file is PCIEx_Status.

PCIEx8 Group Definitions

Channel groups for PCIEx8 are displayed on the screen in the order shown below.

Table 4-97: Channel groups for PCIEx8

Group name	Description	Display radix	Number of bits
L00_protocol – L07_protocol	Lane00 – L07 data and qualifiers used to determine color in waveform windows. This group is intended for display only.	SYM	17 ea
L00_data8b – L07_data8b	Lane00 – Lane07 8-bit data intended for searching and trigger state machine pattern matching.	HEX	8 ea
L00_data – L07_data	Lane00 – Lane07 10-bit data intended for searching and trigger state machine pattern matching.	HEX	10 ea
Link8bit	8 bit data of all Lanes	HEX	64
STP_cntr	Counter synch to STP detection	DEC	10

Table 4-97: Channel groups for PCIEx8 (Cont.)

TLP_fmttype	Fmt and Type fields for TLP	SYM	31
TLP_msg_L0	Message types for TLP – STP on L00	SYM	30
TLP_msg_L4	Message types for TLP – STP on L04	SYM	30
TLP_comp_status	Completion Status for TLP	SYM	27
DLLP_type	DLLP type	SYM	23
L00_RecErr – L07_RecErr	Detects Code and Disparity Errors	SYM	4 ea
Error_bits	Data[11] of all Lanes	HEX	8
Rule_viol	Rule checking of logical physical layer	SYM	4
Status	Indicates general status of link	SYM	16

Table 4-98: L00_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L00_data_11
	10	L00_data_10
	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-99: L01_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L01_data_11
	10	L01_data_10
	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_protocol group is symbolic. The symbol table file is PCIEx_LN_color

Table 4-100: L02_protocol group assignments

Bit order	PCIExpress signal name
9	L02_data_09
8	L02_data_08
7	L02_data_07
6	L02_data_06
5	L02_data_05
4	L02_data_04
3	L02_data_03
2	L02_data_02
1	L02_data_01
LSB 0	L02_data_00

The default radix of the L02_protocol group is symbolic. The symbol table file is PCIEx_LN_color

Table 4-101: L03_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L03_data_11
	10	L03_data_10
	9	L03_data_09
	8	L03_data_08
	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-102: L04_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L04_data_11
	10	L04_data_10
	9	L04_data_09
	8	L04_data_08
	7	L04_data_07
	6	L04_data_06

Table 4-102: L04_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L04_data_05
	4	L04_data_04
	3	L04_data_03
	2	L04_data_02
	1	L04_data_01
LSB	0	L04_data_00

The default radix of the L04_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-103: L05_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L05_data_11
	10	L05_data_10
	9	L05_data_09
	8	L05_data_08
	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the L05_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-104: L06_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L05_data_11
	10	L05_data_10
	9	L05_data_09
	8	L05_data_08
	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the L06_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-105: L07_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L07_data_11
	10	L07_data_10
	9	L07_data_09
	8	L07_data_08
	7	L07_data_07
	6	L07_data_06

Table 4-105: L07_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L07_data_05
	4	L07_data_04
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the L07_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-106: L00_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data8b Group is OFF(HEX).

Table 4-107: L01_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data8b group is OFF(HEX).

Table 4-108: L02_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_data8b group is OFF(HEX).

Table 4-109: L03_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_data8b group is OFF(HEX).

Table 4-110: L04_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L04_data_07
	6	L04_data_06
	5	L04_data_05
	4	L04_data_04
	3	L04_data_03
	2	L04_data_02
	1	L04_data_01
LSB	0	L04_data_00

The default radix of the L04_data8b group is OFF(HEX).

Table 4-111: L05_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the L05_data8b group is OFF(HEX).

Table 4-112: L06_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L06_data_07
	6	L06_data_06
	5	L06_data_05
	4	L06_data_04
	3	L06_data_03
	2	L06_data_02

Table 4-112: L06_data8b group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L06_data_01
LSB	0	L06_data_00

The default radix of the L06_data8b group is OFF(HEX).

Table 4-113: L07_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L07_data_07
	6	L07_data_06
	5	L07_data_05
	4	L07_data_04
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the L07_data8b group is OFF(HEX).

Table 4-114: L00_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data10b Group is OFF(HEX).

Table 4-115: L01_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data10b Group is OFF(HEX).

Table 4-116: L02_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L02_data_09
	8	L02_data_08
	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_data10b Group is OFF(HEX).

Table 4-117: L03_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L03_data_09
	8	L03_data_08
	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_data10b Group is OFF(HEX).

Table 4-118: L04_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L04_data_09
	8	L04_data_08
	7	L04_data_07
	6	L04_data_06
	5	L04_data_05
	4	L04_data_04
	3	L04_data_03
	2	L04_data_02
	1	L04_data_01
LSB	0	L04_data_00

The default radix of the L04_data10b Group is OFF(HEX).

Table 4-119: L05_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L05_data_09
	8	L05_data_08
	7	L05_data_07

Table 4-119: L05_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the L05_data10b Group is OFF(HEX).

Table 4-120: L06_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L06_data_09
	8	L06_data_08
	7	L06_data_07
	6	L06_data_06
	5	L06_data_05
	4	L06_data_04
	3	L06_data_03
	2	L06_data_02
	1	L06_data_01
LSB	0	L06_data_00

The default radix of the L06_data10b Group is OFF(HEX).

Table 4-121: L07_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L07_data_09
	8	L07_data_08
	7	L07_data_07
	6	L07_data_06
	5	L07_data_05
	4	L07_data_04

Table 4-121: L07_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the L07_data10b group is OFF(HEX).

Table 4-122: Link8bit group assignments

Bit order		PCIExpress signal name
MSB	63:60	L07_data_07, L07_data_06, L07_data_05, L07_data_04
	59:56	L07_data_03, L07_data_02, L07_data_01, L07_data_00
	55:52	L06_data_07, L06_data_06, L06_data_05, L06_data_04
	51:48	L06_data_03, L06_data_02, L06_data_01, L06_data_00
	47:44	L05_data_07, L05_data_06, L05_data_05, L05_data_04
	43:40	L05_data_03, L05_data_02, L05_data_01, L05_data_00
	39:36	L04_data_07, L04_data_06, L04_data_05, L04_data_04
	35:32	L04_data_03, L04_data_02, L04_data_01, L04_data_00
	31:28	L03_data_07, L03_data_06, L03_data_05, L03_data_04
	27:24	L03_data_03, L03_data_02, L03_data_01, L03_data_00
	23:20	L02_data_07, L02_data_06, L02_data_05, L02_data_04
	19:16	L02_data_03, L02_data_02, L02_data_01, L02_data_00
	15:12	L01_data_07, L01_data_06, L01_data_05, L01_data_04
	11:8	L01_data_03, L01_data_02, L01_data_01, L01_data_00

Table 4-122: Link8bit group assignments (Cont.)

Bit order		PCIExpress signal name
	7:4	L00_data_07, L00_data_06, L00_data_05, L00_data_04
LSB	3:0	L00_data_03, L00_data_02, L00_data_01, L00_data_00

The default radix of the Link8bit group is HEX.

Table 4-123: STP_cntr group assignments

Bit order		PCIExpress signal name
MSB	9	STP_cntr_09
	8	STP_cntr_08
	7	STP_cntr_07
	6	STP_cntr_06
	5	STP_cntr_05
	4	STP_cntr_04
	3	STP_cntr_03
	2	STP_cntr_02
	1	STP_cntr_01
LSB	0	STP_cntr_00

The default radix of the STP_cntr group is DEC

Table 4-124: TLP_fmttype group assignments

Bit order		PCIExpress signal name
MSB	30	Symbol_Table_Enable
	29	Rule_viol_2
	28	Rule_viol_1
	27	Rule_viol_0
	26	STP_packet_2
	25	STP_packet_1
	24	STP_packet_0
	23	STP_cntr_09
	22	STP_cntr_08

Table 4-124: TLP_fmttype group assignments (Cont.)

Bit order		PCIExpress signal name
	21	STP_cntr_07
	20	STP_cntr_06
	19	STP_cntr_05
	18	STP_cntr_04
	17	STP_cntr_03
	16	STP_cntr_02
	15	STP_cntr_01
	14	STP_cntr_00
	13	L03_data_06
	12	L03_data_05
	11	L03_data_04
	10	L03_data_03
	9	L03_data_02
	8	L03_data_01
	7	L03_data_00
	6	L07_data_06
	5	L07_data_05
	4	L07_data_04
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the TLP_fmttype group is symbolic. The symbol table file is PCIEx8_TLP_fmttype.

Table 4-125: TLP_msg_L0 group assignments

Bit order		PCIExpress signal name
MSB	29	Symbol_Table_Enable
	28	Rule_viol_2
	27	Rule_viol_1
	26	Rule_viol_0
	25	STP_packet_3
	24	STP_packet_1

Table 4-125: TLP_msg_L0 group assignments (Cont.)

Bit order		PCIExpress signal name
	23	STP_packet_0
	22	STP_cntr_09
	21	STP_cntr_08
	20	STP_cntr_07
	19	STP_cntr_06
	18	STP_cntr_05
	17	STP_cntr_04
	16	STP_cntr_03
	15	STP_cntr_02
	14	STP_cntr_01
	13	STP_cntr_00
	12	L03_data_04_L
	11	L03_data_03_L
	10	L03_data_02_L
	9	L03_data_01_L
	8	L03_data_00_L
	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the TLP_msg_L0 group is symbolic. The symbol table file is PCIEx8_TLP_msg_L0.

Table 4-126: TLP_msg_L4 group assignments

Bit order		PCIExpress signal name
MSB	29	Symbol_Table_Enable
	28	Rule_viol_2
	27	Rule_viol_1
	26	Rule_viol_0

Table 4-126: TLP_msg_L4 group assignments (Cont.)

Bit order		PCIExpress signal name
	25	STP_packet_3
	24	STP_packet_1
	23	STP_packet_0
	22	STP_cntr_09
	21	STP_cntr_08
	20	STP_cntr_07
	19	STP_cntr_06
	18	STP_cntr_05
	17	STP_cntr_04
	16	STP_cntr_03
	15	STP_cntr_02
	14	STP_cntr_01
	13	STP_cntr_00
	12	L07_data_04_L
	11	L07_data_03_L
	10	L07_data_02_L
	9	L07_data_01_L
	8	L07_data_00_L
	7	L06_data_07
	6	L06_data_06
	5	L06_data_05
	4	L06_data_04
	3	L06_data_03
	2	L06_data_02
	1	L06_data_01
LSB	0	L06_data_00

The default radix of the TLP_msg_L4 group is symbolic. The symbol table file is PCIEx8_TLP_msg_L4.

Table 4-127: TLP_comp_status group assignments

Bit order		PCIExpress signal name
MSB	26	Symbol_Table_Enable
	25	Rule_viol_2

Table 4-127: TLP_comp_status group assignments (Cont.)

Bit order		PCIExpress signal name
	24	Rule_viol_1
	23	Rule_viol_0
	22	STP_packet_3
	21	STP_packet_1
	20	STP_packet_0
	19	STP_cntr_09
	18	STP_cntr_08
	17	STP_cntr_07
	16	STP_cntr_06
	15	STP_cntr_05
	14	STP_cntr_04
	13	STP_cntr_03
	12	STP_cntr_02
	11	STP_cntr_01
	10	STP_cntr_00
	9	L03_data_04_L
	8	L03_data_03_L
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L07_data_04_L
	3	L07_data_03_L
	2	L05_data_07
	1	L05_data_06
LSB	0	L05_data_05

The default radix of the TLP_comp_status group is symbolic. The symbol table file is PCIEx8_TLP_comp_status.

Table 4-128: DLLP_type group assignments

Bit order		PCIExpress signal name
MSB	22	Symbol_Table_Enable
	21	Rule_viol_2
	20	Rule_viol_1

Table 4-128: DLLP_type group assignments (Cont.)

Bit order		PCIExpress signal name
	19	Rule_viol_0
	18	SDP_packet_2
	17	SDP_packet_1
	16	SDP_packet_0
	15	L01_data_07
	14	L01_data_06
	13	L01_data_05
	12	L01_data_04
	11	L01_data_03
	10	L01_data_02
	9	L01_data_01
	8	L01_data_00
	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the DLLP_type group is symbolic. The symbol table file is PCIEx8_DLLP_type.

Table 4-129: L00_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L00_data_11
	1	L00_data_08
LSB	0	L00_data_10

The default radix of the L00_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-130: L01_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L01_data_11
	1	L01_data_08
LSB	0	L01_data_10

The default radix of the L01_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-131: L02_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L02_data_11
	1	L02_data_08
LSB	0	L02_data_10

The default radix of the L02_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-132: L03_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L03_data_11
	1	L03_data_08
LSB	0	L03_data_10

The default radix of the L03_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-133: L04_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L04_data_11

Table 4-133: L04_RecErr group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L04_data_08
LSB	0	L04_data_10

The default radix of the L04_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-134: L05_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L05_data_11
	1	L05_data_08
LSB	0	L05_data_10

The default radix of the L05_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-135: L06_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L06_data_11
	1	L06_data_08
LSB	0	L06_data_10

The default radix of the L06_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-136: L07_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L07_data_11
	1	L07_data_08
LSB	0	L07_data_10

The default radix of the L07_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-137: Error_bits group assignments

Bit order		PCIExpress signal name
MSB	7	L07_data_11
	6	L06_data_11
	5	L05_data_11
	4	L04_data_11
	3	L03_data_11
	2	L02_data_11
	1	L01_data_11
LSB	0	L00_data_11

The default radix of the Error_bits group is HEX.

Table 4-138: Rule_viol group assignments

Bit order		PCIExpress signal name
MSB	3	EDB_Detect
	2	Rule_viol_2
	1	Rule_viol_1
LSB	0	Rule_viol_0

The default radix of the Rule_viol group is symbolic. The symbol table file is PCIEx_Rule_viol.

Table 4-139: Status group assignments

Bit order		PCIExpress signal name
MSB	15	STP_packet_3
	14	STP_packet_2
	13	STP_packet_1
	12	STP_packet_0
	11	SDP_packet_3
	10	SDP_packet_2
	9	SDP_packet_1
	8	SDP_packet_0
	7	END_EDB_Detect
	6	Ten_bit_mode

Table 4-139: Status group assignments (Cont.)

Bit order		PCIExpress signal name
	5	Symbol_Table_Enable
	4	Elect_Idle_Flag
	3	Rule_viol_3
	2	TS_Detect
	1	SKP_Detect
LSB	0	FTS_Detect

The default radix of the Status group is symbolic. The symbol table file is PCIEx_Status.

PCIEx16 Group Definitions

Channel groups for PCIEx16 will be displayed on the screen in the order shown below.

Table 4-140: Channel groups for PCIEx16

Group name	Description	Display radix	Number of bits
L00_protocol – L15_protocol	Lane00 – L15 data and qualifiers used to determine color in waveform windows. This group is intended for display only.	SYM	17 ea
L00_data8b - L15_data8b	Lane00 – Lane15 8-bit data intended for searching and trigger state machine pattern matching.	HEX	8 ea
L00_data10b – L15_data10b	Lane00 – Lane15 10-bit data intended for searching and trigger state machine pattern matching.	HEX	10 ea
Link8bit	8 bit data of all Lanes	HEX	128
STP_cntr	Counter synch to STP detection	DEC	9
TLP_fmttype_L0	Fmt and Type fields for TLP – STP on L00	SYM	23
TLP_fmttype_L4	Fmt and Type fields for TLP – STP on L04	SYM	23
TLP_fmttype_L8	Fmt and Type fields for TLP – STP on L08	SYM	23
TLP_fmttype_L12	Fmt and Type fields for TLP – STP on L12	SYM	23
TLP_msg_L0	Message types for TLP – STP on L00	SYM	29
TLP_msg_L4	Message types for TLP – STP on L04	SYM	29
TLP_msg_L8	Message types for TLP – STP on L08	SYM	21
TLP_msg_L12	Message types for TLP – STP on L12	SYM	29
TLP_comp_status_L0	Completion Status for TLP – STP on L00	SYM	21
TLP_comp_status_L4	Completion Status for TLP – STP on L04	SYM	21
TLP_comp_status_L8	Completion Status for TLP – STP on L08	SYM	13
TLP_comp_status_L12	Completion Status for TLP – STP on L12	SYM	21
DLLP_type_L0	DLLP type – SDP on L00	SYM	15
DLLP_type_L4	DLLP type – SDP on L04	SYM	15
DLLP_type_L8	DLLP type – SDP on L08	SYM	15
DLLP_type_L12	DLLP type – SDP on L12	SYM	15
L00_RecErr – L15_RecErr	Detects Code and Disparity Errors	SYM	4 ea
Error_bits	Data[11] of all Lanes	HEX	16
Rule_viol	Rule checking of logical physical layer	SYM	4
Status	Indicates general status of link	SYM	16

Table 4-141: L00_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L00_data_11
	10	L00_data_10
	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-142: L01_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L01_data_11
	10	L01_data_10
	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06

Table 4-142: L01_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-143: L02_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L02_data_11
	10	L02_data_10
	9	L02_data_09
	8	L02_data_08
	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_protocol group is symbolic. The symbol table file is PCIEx_LN_color

Table 4-144: L03_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L03_data_11
	10	L03_data_10
	9	L03_data_09
	8	L03_data_08
	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-145: L04_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L04_data_11
	10	L04_data_10
	9	L04_data_09
	8	L04_data_08
	7	L04_data_07
	6	L04_data_06

Table 4-145: L04_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L04_data_05
	4	L04_data_04
	3	L04_data_03
	2	L04_data_02
	1	L04_data_01
LSB	0	L04_data_00

The default radix of the L04_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-146: L05_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L05_data_11
	10	L05_data_10
	9	L05_data_09
	8	L05_data_08
	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the L05_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-147: L06_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L06_data_11
	10	L06_data_10
	9	L06_data_09
	8	L06_data_08
	7	L06_data_07
	6	L06_data_06
	5	L06_data_05
	4	L06_data_04
	3	L06_data_03
	2	L06_data_02
	1	L06_data_01
LSB	0	L06_data_00

The default radix of the L06_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-148: L07_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode
	15	Symbol_Table_Enable
	14	TS_Detect
	13	STP_packet_3
	12	SDP_packet_3
	11	L07_data_11
	10	L07_data_10
	9	L07_data_09
	8	L07_data_08
	7	L07_data_07
	6	L07_data_06

Table 4-148: L07_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L07_data_05
	4	L07_data_04
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the L07_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-149: L08_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L08_data_11
	10	L08_data_10
	9	L08_data_09
	8	L08_data_08
	7	L08_data_07
	6	L08_data_06
	5	L08_data_05
	4	L08_data_04
	3	L08_data_03
	2	L08_data_02
	1	L08_data_01
LSB	0	L08_data_00

The default radix of the L08_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-150: L09_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L09_data_11
	10	L09_data_10
	9	L09_data_09
	8	L09_data_08
	7	L09_data_07
	6	L09_data_06
	5	L09_data_05
	4	L09_data_04
	3	L09_data_03
	2	L09_data_02
	1	L09_data_01
LSB	0	L09_data_00

The default radix of the L09_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-151: L10_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L10_data_11
	10	L10_data_10
	9	L10_data_09
	8	L10_data_08
	7	L10_data_07
	6	L10_data_06

Table 4-151: L10_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L10_data_05
	4	L10_data_04
	3	L10_data_03
	2	L10_data_02
	1	L10_data_01
LSB	0	L10_data_00

The default radix of the L10_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-152: L11_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L11_data_11
	10	L11_data_10
	9	L11_data_09
	8	L11_data_08
	7	L11_data_07
	6	L11_data_06
	5	L11_data_05
	4	L11_data_04
	3	L11_data_03
	2	L11_data_02
	1	L11_data_01
LSB	0	L11_data_00

The default radix of the L11_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-153: L12_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L12_data_11
	10	L12_data_10
	9	L12_data_09
	8	L12_data_08
	7	L12_data_07
	6	L12_data_06
	5	L12_data_05
	4	L12_data_04
	3	L12_data_03
	2	L12_data_02
	1	L12_data_01
LSB	0	L12_data_00

The default radix of the L12_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-154: L13_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L13_data_11
	10	L13_data_10
	9	L13_data_09
	8	L13_data_08
	7	L13_data_07
	6	L13_data_06

Table 4-154: L13_protocol group assignments (Cont.)

Bit order		PCIExpress signal name
	5	L13_data_05
	4	L13_data_04
	3	L13_data_03
	2	L13_data_02
	1	L13_data_01
LSB	0	L13_data_00

The default radix of the L13_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-155: L14_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L14_data_11
	10	L14_data_10
	9	L14_data_09
	8	L14_data_08
	7	L14_data_07
	6	L14_data_06
	5	L14_data_05
	4	L14_data_04
	3	L14_data_03
	2	L14_data_02
	1	L14_data_01
LSB	0	L14_data_00

The default radix of the L14_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-156: L15_protocol group assignments

Bit order		PCIExpress signal name
MSB	16	Ten_bit_mode_Slave
	15	Symbol_Table_Enable_Slave
	14	TS_Detect_Slave
	13	STP_packet_3_Slave
	12	SDP_packet_3_Slave
	11	L15_data_11
	10	L15_data_10
	9	L15_data_09
	8	L15_data_08
	7	L15_data_07
	6	L15_data_06
	5	L15_data_05
	4	L15_data_04
	3	L15_data_03
	2	L15_data_02
	1	L15_data_01
LSB	0	L15_data_00

The default radix of the L15_protocol group is symbolic. The symbol table file is PCIEx_LN_color.

Table 4-157: L00_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data8b group is OFF(HEX).

Table 4-158: L01_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data8b group is OFF(HEX).

Table 4-159: 02_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_data8b group is OFF(HEX).

Table 4-160: L03_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02

Table 4-160: L03_data8b group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_data8b group is OFF(HEX).

Table 4-161: L04_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L04_data_07
	6	L04_data_06
	5	L04_data_05
	4	L04_data_04
	3	L04_data_03
	2	L04_data_02
	1	L04_data_01
LSB	0	L04_data_00

The default radix of the L04_data8b group is OFF(HEX).

Table 4-162: L05_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the L05_data8b group is OFF(HEX).

Table 4-163: L06_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L06_data_07
	6	L06_data_06
	5	L06_data_05
	4	L06_data_04
	3	L06_data_03
	2	L06_data_02
	1	L06_data_01
LSB	0	L06_data_00

The default radix of the L06_data8b group is OFF(HEX).

Table 4-164: L07_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L07_data_07
	6	L07_data_06
	5	L07_data_05
	4	L07_data_04
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the L07_data8b group is OFF(HEX).

Table 4-165: L08_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L08_data_07
	6	L08_data_06
	5	L08_data_05
	4	L08_data_04
	3	L08_data_03
	2	L08_data_02

Table 4-165: L08_data8b group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L08_data_01
LSB	0	L08_data_00

The default radix of the L08_data8b group is OFF(HEX).

Table 4-166: L09_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L09_data_07
	6	L09_data_06
	5	L09_data_05
	4	L09_data_04
	3	L09_data_03
	2	L09_data_02
	1	L09_data_01
LSB	0	L09_data_00

The default radix of the L09_data8b group is OFF(HEX).

Table 4-167: L10_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L10_data_07
	6	L10_data_06
	5	L10_data_05
	4	L10_data_04
	3	L10_data_03
	2	L10_data_02
	1	L10_data_01
LSB	0	L10_data_00

The default radix of the L10_data8b group is OFF(HEX).

Table 4-168: L11_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L11_data_07
	6	L11_data_06
	5	L11_data_05
	4	L11_data_04
	3	L11_data_03
	2	L11_data_02
	1	L11_data_01
LSB	0	L11_data_00

The default radix of the L11_data8b group is OFF(HEX).

Table 4-169: L12_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L12_data_07
	6	L12_data_06
	5	L12_data_05
	4	L12_data_04
	3	L12_data_03
	2	L12_data_02
	1	L12_data_01
LSB	0	L12_data_00

The default radix of the L12_data8b group is OFF(HEX).

Table 4-170: L13_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L13_data_07
	6	L13_data_06
	5	L13_data_05
	4	L13_data_04
	3	L13_data_03
	2	L13_data_02

Table 4-170: L13_data8b group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L13_data_01
LSB	0	L13_data_00

The default radix of the L13_data8b group is OFF(HEX).

Table 4-171: L14_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L14_data_07
	6	L14_data_06
	5	L14_data_05
	4	L14_data_04
	3	L14_data_03
	2	L14_data_02
	1	L14_data_01
LSB	0	L14_data_00

The default radix of the L14_data8b group is OFF(HEX).

Table 4-172: L15_data8b group assignments

Bit order		PCIExpress signal name
MSB	7	L15_data_07
	6	L15_data_06
	5	L15_data_05
	4	L15_data_04
	3	L15_data_03
	2	L15_data_02
	1	L15_data_01
LSB	0	L15_data_00

The default radix of the L15_data8b group is OFF(HEX).

Table 4-173: L00_data10b group Assignments

Bit order		PCIExpress signal name
MSB	9	L00_data_09
	8	L00_data_08
	7	L00_data_07
	6	L00_data_06
	5	L00_data_05
	4	L00_data_04
	3	L00_data_03
	2	L00_data_02
	1	L00_data_01
LSB	0	L00_data_00

The default radix of the L00_data10b group is OFF(HEX).

Table 4-174: L01_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L01_data_09
	8	L01_data_08
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the L01_data10b group is OFF(HEX).

Table 4-175: L02_data10b group Assignments

Bit order		PCIExpress signal name
MSB	9	L02_data_09
	8	L02_data_08
	7	L02_data_07

Table 4-175: L02_data10b group Assignments (Cont.)

Bit order		PCIExpress signal name
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the L02_data10b group is OFF(HEX).

Table 4-176: L03_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L03_data_09
	8	L03_data_08
	7	L03_data_07
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the L03_data10b group is OFF(HEX).

Table 4-177: L04_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L04_data_09
	8	L04_data_08
	7	L04_data_07
	6	L04_data_06
	5	L04_data_05
	4	L04_data_04

Table 4-177: L04_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	3	L04_data_03
	2	L04_data_02
	1	L04_data_01
LSB	0	L04_data_00

The default radix of the L04_data10b group is OFF(HEX).

Table 4-178: L05_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L05_data_09
	8	L05_data_08
	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the L05_data10b group is OFF(HEX).

Table 4-179: L06_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L06_data_09
	8	L06_data_08
	7	L06_data_07
	6	L06_data_06
	5	L06_data_05
	4	L06_data_04
	3	L06_data_03
	2	L06_data_02

Table 4-179: L06_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L06_data_01
LSB	0	L06_data_00

The default radix of the L06_data10b group is OFF(HEX).

Table 4-180: L07_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L07_data_09
	8	L07_data_08
	7	L07_data_07
	6	L07_data_06
	5	L07_data_05
	4	L07_data_04
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the L07_data10b group is OFF(HEX).

Table 4-181: L08_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L08_data_09
	8	L08_data_08
	7	L08_data_07
	6	L08_data_06
	5	L08_data_05
	4	L08_data_04
	3	L08_data_03
	2	L08_data_02
	1	L08_data_01
LSB	0	L08_data_00

The default radix of the L08_data10b group is OFF(HEX).

Table 4-182: L09_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L09_data_09
	8	L09_data_08
	7	L09_data_07
	6	L09_data_06
	5	L09_data_05
	4	L09_data_04
	3	L09_data_03
	2	L09_data_02
	1	L09_data_01
LSB	0	L09_data_00

The default radix of the L09_data10b group is OFF(HEX).

Table 4-183: L10_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L10_data_09
	8	L10_data_08
	7	L10_data_07
	6	L10_data_06
	5	L10_data_05
	4	L10_data_04
	3	L10_data_03
	2	L10_data_02
	1	L10_data_01
LSB	0	L10_data_00

The default radix of the L10_data10b group is OFF(HEX).

Table 4-184: L11_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L11_data_09
	8	L11_data_08
	7	L11_data_07
	6	L11_data_06
	5	L11_data_05
	4	L11_data_04
	3	L11_data_03
	2	L11_data_02
	1	L11_data_01
LSB	0	L11_data_00

The default radix of the L11_data10b group is OFF(HEX).

Table 4-185: L12_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L12_data_09
	8	L12_data_08
	7	L12_data_07
	6	L12_data_06
	5	L12_data_05
	4	L12_data_04
	3	L12_data_03
	2	L12_data_02
	1	L12_data_01
LSB	0	L12_data_00

The default radix of the L12_data10b group is OFF(HEX).

Table 4-186: L13_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L13_data_09
	8	L13_data_08
	7	L13_data_07

Table 4-186: L13_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	6	L13_data_06
	5	L13_data_05
	4	L13_data_04
	3	L13_data_03
	2	L13_data_02
	1	L13_data_01
LSB	0	L13_data_00

The default radix of the L13_data10b group is OFF(HEX).

Table 4-187: L14_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L14_data_09
	8	L14_data_08
	7	L14_data_07
	6	L14_data_06
	5	L14_data_05
	4	L14_data_04
	3	L14_data_03
	2	L14_data_02
	1	L14_data_01
LSB	0	L14_data_00

The default radix of the L14_data10b group is OFF(HEX).

Table 4-188: L15_data10b group assignments

Bit order		PCIExpress signal name
MSB	9	L15_data_09
	8	L15_data_08
	7	L15_data_07
	6	L15_data_06
	5	L15_data_05
	4	L15_data_04

Table 4-188: L15_data10b group assignments (Cont.)

Bit order		PCIExpress signal name
	3	L15_data_03
	2	L15_data_02
	1	L15_data_01
LSB	0	L15_data_00

The default radix of the L15_data10b group is OFF(HEX).

Table 4-189: Link8bit group assignments

Bit order		PCIExpress signal name
MSB	127:124	L15_data_07, L15_data_06, L15_data_05, L15_data_04
	123:120	L15_data_03, L15_data_02, L15_data_01, L15_data_00
	119:116	L14_data_07, L14_data_06, L14_data_05, L14_data_04
	115:112	L14_data_03, L14_data_02, L14_data_01, L14_data_00
	111:108	L13_data_07, L13_data_06, L13_data_05, L13_data_04
	107:104	L13_data_03, L13_data_02, L13_data_01, L13_data_00
	103:100	L12_data_07, L12_data_06, L12_data_05, L12_data_04
	99:96	L12_data_03, L12_data_02, L12_data_01, L12_data_00
	95:92	L11_data_07, L11_data_06, L11_data_05, L11_data_04
	91:88	L11_data_03, L11_data_02, L11_data_01, L11_data_00
	87:84	L10_data_07, L10_data_06, L10_data_05, L10_data_04
	83:80	L10_data_03, L10_data_02, L10_data_01, L10_data_00
	79:76	L09_data_07, L09_data_06, L09_data_05, L09_data_04
	75:72	L09_data_03, L09_data_02, L09_data_01, L09_data_00

Table 4-189: Link8bit group assignments (Cont.)

Bit order		PCIExpress signal name
	71:68	L08_data_07, L08_data_06, L08_data_05, L08_data_04
	67:64	L08_data_03, L08_data_02, L08_data_01, L08_data_00
	63:60	L07_data_07, L07_data_06, L07_data_05, L07_data_04
	59:56	L07_data_03, L07_data_02, L07_data_01, L07_data_00
	55:52	L06_data_07, L06_data_06, L06_data_05, L06_data_04
	51:48	L06_data_03, L06_data_02, L06_data_01, L06_data_00
	47:44	L05_data_07, L05_data_06, L05_data_05, L05_data_04
	43:40	L05_data_03, L05_data_02, L05_data_01, L05_data_00
	39:36	L04_data_07, L04_data_06, L04_data_05, L04_data_04
	35:32	L04_data_03, L04_data_02, L04_data_01, L04_data_00
	31:28	L03_data_07, L03_data_06, L03_data_05, L03_data_04
	27:24	L03_data_03, L03_data_02, L03_data_01, L03_data_00
	23:20	L02_data_07, L02_data_06, L02_data_05, L02_data_04
	19:16	L02_data_03, L02_data_02, L02_data_01, L02_data_00
	15:12	L01_data_07, L01_data_06, L01_data_05, L01_data_04
	11:8	L01_data_03, L01_data_02, L01_data_01, L01_data_00
	7:4	L00_data_07, L00_data_06, L00_data_05, L00_data_04
LSB	3:0	L00_data_03, L00_data_02, L00_data_01, L00_data_00

The default radix of the Link8bit group is HEX.

Table 4-190: STP_cntr group assignments

Bit order		PCIExpress signal name
MSB	8	STP_cntr_08
	7	STP_cntr_07
	6	STP_cntr_06
	5	STP_cntr_05
	4	STP_cntr_04
	3	STP_cntr_03
	2	STP_cntr_02
	1	STP_cntr_01
LSB	0	STP_cntr_00

The default radix of the STP_cntr group is DEC

Table 4-191: TLP_fmttype_L0 group assignments

Bit order		PCIExpress signal name
MSB	22	Symbol_Table_Enable
	21	Rule_viol_2
	20	Rule_viol_1
	19	Rule_viol_0
	18	STP_packet_2
	17	STP_packet_1
	16	STP_packet_0
	15	STP_cntr_08
	14	STP_cntr_07
	13	STP_cntr_06
	12	STP_cntr_05
	11	STP_cntr_04
	10	STP_cntr_03
	9	STP_cntr_02
	8	STP_cntr_01
	7	STP_cntr_00
	6	L03_data_06
	5	L03_data_05
	4	L03_data_04
	3	L03_data_03

Table 4-191: TLP_fmttype_L0 group assignments (Cont.)

Bit order		PCIExpress signal name
	2	L03_data_02
	1	L03_data_01
LSB	0	L03_data_00

The default radix of the TLP_fmttype_L0 group is symbolic. The symbol table file is PCIEx16_TLP_fmttype_L0.

Table 4-192: TLP_fmttype_L4 group assignments

Bit order		PCIExpress signal name
MSB	22	Symbol_Table_Enable
	21	Rule_viol_2
	20	Rule_viol_1
	19	Rule_viol_0
	18	STP_packet_2
	17	STP_packet_1
	16	STP_packet_0
	15	STP_cntr_08
	14	STP_cntr_07
	13	STP_cntr_06
	12	STP_cntr_05
	11	STP_cntr_04
	10	STP_cntr_03
	9	STP_cntr_02
	8	STP_cntr_01
	7	STP_cntr_00
	6	L07_data_06
	5	L07_data_05
	4	L07_data_04
	3	L07_data_03
	2	L07_data_02
	1	L07_data_01
LSB	0	L07_data_00

The default radix of the TLP_fmttype_L4 group is symbolic. The symbol table file is PCIEx16_TLP_fmttype_L4.

Table 4-193: TLP_fmttype_L8 group assignments

Bit order		PCIExpress signal name
MSB	22	Symbol_Table_Enable_Slave
	21	Rule_viol_2_Slave
	20	Rule_viol_1_Slave
	19	Rule_viol_0_Slave
	18	STP_packet_2_Slave
	17	STP_packet_1_Slave
	16	STP_packet_0_Slave
	15	STP_cntr_08_Slave
	14	STP_cntr_07_Slave
	13	STP_cntr_06_Slave
	12	STP_cntr_05_Slave
	11	STP_cntr_04_Slave
	10	STP_cntr_03_Slave
	9	STP_cntr_02_Slave
	8	STP_cntr_01_Slave
	7	STP_cntr_00_Slave
	6	L11_data_06
	5	L11_data_05
	4	L11_data_04
	3	L11_data_03
	2	L11_data_02
	1	L11_data_01
LSB	0	L11_data_00

The default radix of the TLP_fmttype_L8 group is symbolic. The symbol table file is PCIEx16_TLP_fmttype_L8.

Table 4-194: TLP_fmttype_L12 group assignments

Bit order		PCIExpress signal name
MSB	22	Symbol_Table_Enable_Slave
	21	Rule_viol_2_Slave

Table 4-194: TLP_fmttype_L12 group assignments (Cont.)

Bit order		PCIExpress signal name
	20	Rule_viol_1_Slave
	19	Rule_viol_0_Slave
	18	STP_packet_2_Slave
	17	STP_packet_1_Slave
	16	STP_packet_0_Slave
	15	STP_cntr_08_Slave
	14	STP_cntr_07_Slave
	13	STP_cntr_06_Slave
	12	STP_cntr_05_Slave
	11	STP_cntr_04_Slave
	10	STP_cntr_03_Slave
	9	STP_cntr_02_Slave
	8	STP_cntr_01_Slave
	7	STP_cntr_00_Slave
	6	L10_data_06
	5	L10_data_05
	4	L10_data_04
	3	L10_data_03
	2	L10_data_02
	1	L10_data_01
LSB	0	L10_data_00

The default radix of the TLP_fmttype_L12 group is symbolic. The symbol table file is PCIEx16_TLP_fmttype_L12.

Table 4-195: TLP_msg_L0 group assignments

Bit order		PCIExpress signal name
MSB	28	Symbol_Table_Enable
	27	Rule_viol_2
	26	Rule_viol_1
	25	Rule_viol_0
	24	STP_packet_2
	23	STP_packet_1
	22	STP_packet_0

Table 4-195: TLP_msg_L0 group assignments (Cont.)

Bit order		PCIExpress signal name
	21	STP_cntr_08
	20	STP_cntr_07
	19	STP_cntr_06
	18	STP_cntr_05
	17	STP_cntr_04
	16	STP_cntr_03
	15	STP_cntr_02
	14	STP_cntr_01
	13	STP_cntr_00
	12	L03_data_04
	11	L03_data_03
	10	L03_data_02
	9	L03_data_01
	8	L03_data_00
	7	L10_data_07
	6	L10_data_06
	5	L10_data_05
	4	L10_data_04
	3	L10_data_03
	2	L10_data_02
	1	L10_data_01
LSB	0	L10_data_00

The default radix of the TLP_msg_L0 group is symbolic. The symbol table file is PCIEx16_TLP_msg_L0.

Table 4-196: TLP_msg_L4 group assignments

Bit order		PCIExpress signal name
MSB	28	Symbol_Table_Enable
	27	Rule_viol_2
	26	Rule_viol_1
	25	Rule_viol_0
	24	STP_packet_2
	23	STP_packet_1

Table 4-196: TLP_msg_L4 group assignments (Cont.)

Bit order		PCIExpress signal name
	22	STP_packet_0
	21	STP_cntr_08
	20	STP_cntr_07
	19	STP_cntr_06
	18	STP_cntr_05
	17	STP_cntr_04
	16	STP_cntr_03
	15	STP_cntr_02
	14	STP_cntr_01
	13	STP_cntr_00
	12	L07_data_04
	11	L07_data_03
	10	L07_data_02
	9	L07_data_01
	8	L07_data_00
	7	L14_data_07
	6	L14_data_06
	5	L14_data_05
	4	L14_data_04
	3	L14_data_03
	2	L14_data_02
	1	L14_data_01
LSB	0	L14_data_00

The default radix of the TLP_msg_L4 group is symbolic. The symbol table file is PCIEx16_TLP_msg_L4.

Table 4-197: TLP_msg_L8 group assignments

Bit order		PCIExpress signal name
MSB	20	Symbol_Table_Enable
	19	Rule_viol_2
	18	Rule_viol_1
	17	Rule_viol_0
	16	STP_packet_3

Table 4-197: TLP_msg_L8 group assignments (Cont.)

Bit order		PCIExpress signal name
	15	STP_packet_2_L
	14	STP_packet_1_L
	13	STP_packet_0_L
	12	L11_data_04_L
	11	L11_data_03_L
	10	L11_data_02_L
	9	L11_data_01_L
	8	L11_data_00_L
	7	L02_data_07
	6	L02_data_06
	5	L02_data_05
	4	L02_data_04
	3	L02_data_03
	2	L02_data_02
	1	L02_data_01
LSB	0	L02_data_00

The default radix of the TLP_msg_L8 group is symbolic. The symbol table file is PCIEx16_TLP_msg_L8.

Table 4-198: TLP_msg_L12 group assignments

Bit order		PCIExpress signal name
MSB	28	Symbol_Table_Enable
	27	Rule_viol_2
	26	Rule_viol_1
	25	Rule_viol_0
	24	STP_packet_2
	23	STP_packet_1
	22	STP_packet_0
	21	STP_cntr_08
	20	STP_cntr_07
	19	STP_cntr_06
	18	STP_cntr_05
	17	STP_cntr_04

Table 4-198: TLP_msg_L12 group assignments (Cont.)

Bit order		PCIExpress signal name
	16	STP_cntr_03
	15	STP_cntr_02
	14	STP_cntr_01
	13	STP_cntr_00
	12	L15_data_04_L
	11	L15_data_03_L
	10	L15_data_02_L
	9	L15_data_01_L
	8	L15_data_00_L
	7	L06_data_07
	6	L06_data_06
	5	L06_data_05
	4	L06_data_04
	3	L06_data_03
	2	L06_data_02
	1	L06_data_01
LSB	0	L06_data_00

The default radix of the TLP_msg_L12 group is symbolic. The symbol table file is PCIEx16_TLP_msg_L12.

Table 4-199: TLP_comp_status_L0 group assignments

Bit order		PCIExpress signal name
MSB	20	Symbol_Table_Enable
	19	Rule_viol_2
	18	Rule_viol_1
	17	Rule_viol_0
	16	STP_packet_2
	15	STP_packet_1
	14	STP_packet_0
	13	STP_cntr_08
	12	STP_cntr_07
	11	STP_cntr_06
	10	STP_cntr_05

Table 4-199: TLP_comp_status_L0 group assignments (Cont.)

Bit order		PCIExpress signal name
	9	STP_cntr_04
	8	STP_cntr_03
	7	STP_cntr_02
	6	STP_cntr_01
	5	STP_cntr_00
	4	L03_data_04
	3	L03_data_03
	2	L09_data_07
	1	L09_data_06
LSB	0	L09_data_05

The default radix of the TLP_comp_status_L0 group is symbolic. The symbol table file is PCIEx16_TLP_comp_status_L0.

Table 4-200: TLP_comp_status_L4 group assignments

Bit order		PCIExpress signal name
MSB	20	Symbol_Table_Enable
	19	Rule_viol_2
	18	Rule_viol_1
	17	Rule_viol_0
	16	STP_packet_2
	15	STP_packet_1
	14	STP_packet_0
	13	STP_cntr_08
	12	STP_cntr_07
	11	STP_cntr_06
	10	STP_cntr_05
	9	STP_cntr_04
	8	STP_cntr_03
	7	STP_cntr_02
	6	STP_cntr_01
	5	STP_cntr_00
	4	L07_data_04
	3	L07_data_03

Table 4-200: TLP_comp_status_L4 group assignments (Cont.)

Bit order		PCIExpress signal name
	2	L13_data_07
	1	L13_data_06
LSB	0	L13_data_05

The default radix of the TLP_comp_status_L4 group is symbolic. The symbol table file is PCIEx16_TLP_comp_status_L4.

Table 4-201: TLP_comp_status_L8 group assignments

Bit order		PCIExpress signal name
MSB	12	Symbol_Table_Enable
	11	Rule_viol_2
	10	Rule_viol_1
	9	Rule_viol_0
	8	STP_packet_3
	7	STP_packet_2_L
	6	STP_packet_1_L
	5	STP_packet_0_L
	4	L11_data_04_L
	3	L11_data_03_L
	2	L01_data_07
	1	L01_data_06
LSB		L01_data_05

The default radix of the TLP_comp_status_L8 group is symbolic. The symbol table file is PCIEx16_TLP_comp_status_L8.

Table 4-202: TLP_comp_status_L12 group assignments

Bit order		PCIExpress signal name
MSB	20	Symbol_Table_Enable
	19	Rule_viol_2
	18	Rule_viol_1
	17	Rule_viol_0
	16	STP_packet_2

Table 4-202: TLP_comp_status_L12 group assignments (Cont.)

Bit order		PCIExpress signal name
	15	STP_packet_1
	14	STP_packet_0
	13	STP_cntr_08
	12	STP_cntr_07
	11	STP_cntr_06
	10	STP_cntr_05
	9	STP_cntr_04
	8	STP_cntr_03
	7	STP_cntr_02
	6	STP_cntr_01
	5	STP_cntr_00
	4	L15_data_04_L
	3	L15_data_03_L
	2	L05_data_07
	1	L05_data_06
LSB	0	L05_data_05

The default radix of the TLP_comp_status_L12 group is symbolic. The symbol table file is PCIEx16_TLP_comp_status_L12.

Table 4-203: DLLP_type_L0 group assignments

Bit order		PCIExpress signal name
MSB	14	Symbol_Table_Enable
	13	Rule_viol_2
	12	Rule_viol_1
	11	Rule_viol_0
	10	SDP_packet_2
	9	SDP_packet_1
	8	SDP_packet_0
	7	L01_data_07
	6	L01_data_06
	5	L01_data_05
	4	L01_data_04
	3	L01_data_03

Table 4-203: DLLP_type_L0 group assignments (Cont.)

Bit order		PCIExpress signal name
	2	L01_data_02
	1	L01_data_01
LSB	0	L01_data_00

The default radix of the DLLP_type_L0 group is symbolic. The symbol table file is PCIEx16_DLLP_type_L0.

Table 4-204: DLLP_type_L4 group assignments

Bit order		PCIExpress signal name
MSB	14	Symbol_Table_Enable
	13	Rule_viol_2
	12	Rule_viol_1
	11	Rule_viol_0
	10	SDP_packet_2
	9	SDP_packet_1
	8	SDP_packet_0
	7	L05_data_07
	6	L05_data_06
	5	L05_data_05
	4	L05_data_04
	3	L05_data_03
	2	L05_data_02
	1	L05_data_01
LSB	0	L05_data_00

The default radix of the DLLP_type_L4 group is symbolic. The symbol table file is PCIEx16_DLLP_type_L4.

Table 4-205: DLLP_type_L8 group assignments

Bit order		PCIExpress signal name
MSB	14	Symbol_Table_Enable_Slave
	13	Rule_viol_2_Slave
	12	Rule_viol_1_Slave

Table 4-205: DLLP_type_L8 group assignments (Cont.)

Bit order		PCIExpress signal name
	11	Rule_viol_0_Slave
	10	SDP_packet_2_Slave
	9	SDP_packet_1_Slave
	8	SDP_packet_0_Slave
	7	L09_data_07
	6	L09_data_06
	5	L09_data_05
	4	L09_data_04
	3	L09_data_03
	2	L09_data_02
	1	L09_data_01
LSB	0	L09_data_00

The default radix of the DLLP_type_L8 group is symbolic. The symbol table file is PCIEx16_DLLP_type_L8.

Table 4-206: DLLP_type_L12 group assignments

Bit order		PCIExpress signal name
MSB	14	Symbol_Table_Enable_Slave
	13	Rule_viol_2_Slave
	12	Rule_viol_1_Slave
	11	Rule_viol_0_Slave
	10	SDP_packet_2_Slave
	9	SDP_packet_1_Slave
	8	SDP_packet_0_Slave
	7	L13_data_07
	6	L13_data_06
	5	L13_data_05
	4	L13_data_04
	3	L13_data_03
	2	L13_data_02
	1	L13_data_01
LSB	0	L13_data_00

The default radix of the DLLP_type_L12 group is symbolic. The symbol table file is PCIEx16_DLLP_type_L12.

Table 4-207: L00_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L00_data_11
	1	L00_data_08
LSB	0	L00_data_10

The default radix of the L00_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-208: L01_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L01_data_11
	1	L01_data_08
LSB	0	L01_data_10

The default radix of the L01_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-209: L02_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L02_data_11
	1	L02_data_08
LSB	0	L02_data_10

The default radix of the L02_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-210: L03_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L03_data_11
	1	L03_data_08
LSB	0	L03_data_10

The default radix of the L03_RecErr group is symbolic. The symbol table file is PCIEx_RecErr. See

Table 4-211: L04_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L04_data_11
	1	L04_data_08
LSB	0	L04_data_10

The default radix of the L04_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-212: L05_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L05_data_11
	1	L05_data_08
LSB	0	L05_data_10

The default radix of the L05_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-213: L06_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L06_data_11

Table 4-213: L06_RecErr group assignments (Cont.)

Bit order		PCIExpress signal name
	1	L06_data_08
LSB	0	L06_data_10

The default radix of the L06_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-214: L07_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode
	2	L07_data_11
	1	L07_data_08
LSB	0	L07_data_10

The default radix of the L07_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-215: L08_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L08_data_11
	1	L08_data_08
LSB	0	L08_data_10

The default radix of the L08_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-216: L09_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L09_data_11
	1	L09_data_08
LSB	0	L09_data_10

The default radix of the L09_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-217: L10_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L10_data_11
	1	L10_data_08
LSB	0	L10_data_10

The default radix of the L10_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-218: L11_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L11_data_11
	1	L11_data_08
LSB	0	L11_data_10

The default radix of the L11_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-219: L12_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L12_data_11
	1	L12_data_08
LSB	0	L12_data_10

The default radix of the L12_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-220: L13_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L13_data_11
	1	L13_data_08
LSB	0	L13_data_10

The default radix of the L13_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-221: L14_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L14_data_11
	1	L14_data_08
LSB	0	L14_data_10

The default radix of the L14_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-222: L15_RecErr group assignments

Bit order		PCIExpress signal name
MSB	3	Ten_bit_mode_Slave
	2	L15_data_11
	1	L15_data_08
LSB	0	L15_data_10

The default radix of the L15_RecErr group is symbolic. The symbol table file is PCIEx_RecErr.

Table 4-223: Error_bits group assignments

Bit order		PCIExpress signal name
MSB	15	L15_data_11
0	14	L14_data_11
0	13	L13_data_11

Table 4-223: Error_bits group assignments (Cont.)

Bit order		PCIExpress signal name
0	12	L12_data_11
0	11	L11_data_11
0	10	L10_data_11
0	9	L09_data_11
0	8	L08_data_11
0	7	L07_data_11
0	6	L06_data_11
0	5	L05_data_11
0	4	L04_data_11
0	3	L03_data_11
0	2	L02_data_11
0	1	L01_data_11
LSB	0	L00_data_11

The default radix of the Error_bits group is HEX.

Table 4-224: Rule_viol group assignments

Bit order		PCIExpress signal name
MSB	3	EDB_Detect
	2	Rule_viol_2
	1	Rule_viol_1
LSB	0	Rule_viol_0

The default radix of the Rule_viol group is symbolic. The symbol table file is PCIEx_Rule_viol.

Table 4-225: Status group assignments

Bit order		PCIExpress signal name
MSB	15	STP_packet_3
	14	STP_packet_2
	13	STP_packet_1
	12	STP_packet_0
	11	SDP_packet_3

Table 4-225: Status group assignments (Cont.)

Bit order		PCIExpress signal name
	10	SDP_packet_2
	9	SDP_packet_1
	8	SDP_packet_0
	7	END_EDB_Detect
	6	Ten_bit_mode
	5	Symbol_Table_Enable
	4	Elect_Idle_Flag
	3	Rule_viol_3
	2	TS_Detect
	1	SKP_Detect
LSB	0	FTS_Detect

The default radix of the Status group is symbolic. The symbol table file is PCIEx_Status.

Table 4-226: Explanation of LXX_RecErr group

Ten_bit_mode	Lxx_data11	Lxx_data08	Lxx_data10	Priority	Description
0	0	0	0	8	Normal Operation, valid data character received
0	0	0	1	3	Overrun/Underrun: The receiver interface synchronization logic has detected an overrun/underrun condition.
0	0	1	0	7	Normal Operation, valid control character received
0	0	1	1	6	Normal Operation, valid K28.5 character received
0	1	0	0	4	Code error: The 8b/10b decoder detected an illegal character
0	1	0	1	5	Disparity Error: The 8b/10b decoder detected a disparity error
0	1	1	0	1	Not Byte synch: The receiver is in start-up or has lost symbol alignment and is searching for alignment
0	1	1	1	2	Not Word synch: Not applicable
1	0	bit 8 of 10 bit data	0	4	Normal Operation, valid data character received
1	0	bit 8 of 10 bit data	1	3	Normal Operation, K28.5 character received

Table 4-226: Explanation of LXX_RecErr group (Cont.)

Ten_bit_mode	Lxx_data11	Lxx_data08	Lxx_data10	Priority	Description
1	1	bit 8 of 10 bit data	0	1	Not Byte synch: The receiver is in start-up or has lost symbol alignment and is searching for alignment
1	1	bit 8 of 10 bit data	1	2	Overrun/Underrun: The receiver interface synchronization logic detected an overrun/underrun condition.

Table 4-227: Explanation of Rule Violation Group

Rule_Viol_2	Rule_Viol_1	Rule_Viol_0	Description
0	0	0	No error
0	0	1	Multiple STP in the same symbol time on lane(s) 0, 4, 8 or 12
0	1	0	Multiple SDP in the same symbol time on lane(s) 0, 4, 8 or 12
0	1	1	STP detecton on lane other than lanes 0, 4, 8 or 12
1	0	0	SDP detected on lane other than lanes 0, 4, 8 or 12
1	0	1	Two SDP/STP symbols detected without an intervening END/EDB symbol
1	1	0	END/EDB not detected on eigth symbols following SDP
1	1	1	Multiple violation at the same time

Table 4-228: Explanation of STP_Packet_0 and STP_Packet_1 signals

STP_Packet_1	STP_Packet_0	Description
0	0	Most recent TLP packet started on Lane 00 - valid until next STP symbol found
0	1	Most recent TLP packet started on Lane 04 - valid until next STP symbol found
1	0	Most recent TLP packet started on Lane 08 - valid until next STP symbol found
1	1	Most recent TLP Packet started on Lane 12 - valid until next STP symbol found

Table 4-229: Explanation of SDP_Packet_0 and SDP_Packet_1 signals

SDP_Packet_1	SDP_Packet_0	Description
0	0	Most recent DLLP packet started on Lane 00 - valid until next SDP symbol found
0	1	Most recent DLLP packet started on Lane 00 - valid until next SDP symbol found
1	0	Most recent DLLP packet started on Lane 08 - valid until next SDP symbol found
1	1	Most recent DLLP packet started on Lane 12 - valid until next SDP symbol found

Table 4-230: Explanation of STP_Packet_3

STP_Packet_3	Description
0	No TLP in progress
1	Bit is enabled at STP symbol until an EDB/ END arrives

Table 4-231: Explanation of SDP_Packet_3

SDP_Packet_3	Description
0	No DLLP in progress
1	Bit is enabled at SDP symbol and remains high for eight symbols regardless if an EDB/ END arrives

Table 4-232: Explanation of STP_Packet_2

STP_Packet_2	Description
0	No STP symbol in that symbol time
1	STP symbol detected in that symbol time

Table 4-233: Explanation of SDP_Packet_2

SDP_Packet_2	Description
0	No SDP symbol in that symbol time
1	SDP symbol detected in that symbol time

Channel Assignment Tables

Conventions

Channel assignments listed in the following tables use the following conventions:

- All signals are required by the support unless indicated otherwise.
- Channels are listed starting with the most significant bit (MSB) descending to the least significant bit (LSB), from top to bottom in each table.
- Channel group assignments are for all modules unless otherwise noted.
- The portable logic analyzer has the lower-numbered slot on the top, and the benchtop logic analyzer has the lower-numbered slot on the left.
- The term “master” module refers to the middle module of a five-wide merged module. The term “slave” module refers to the modules to the left or right of the master module of a five-wide merged module. Figure 4–1 shows the configuration for a two-wide module merge (x16 width link only). The modules must be configured as shown in Figure 4–1.

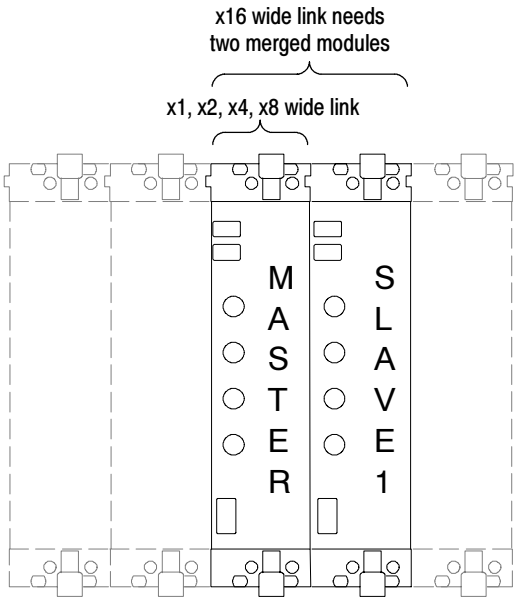

Figure 4-1: Configuration of the Master and Slave1 modules

PCIEx1

Table 4-234: PCIEx1 Clock channels (also stored as acquisition data)

Logic analyzer clock channel	PCIExpress support package channel name	Description
M_Clock:3	SDP_packet_3	HIGH for DLLP
M_Clock:2	TS_Detect	HIGH for training sequences TS1/TS2
M_Clock:1	clk	Clock Sync. To parallel data
M_Clock:0	SKP_Detect	HIGH for Skip-Ordered-Sets

Table 4-235: PCIEx1 QUAL channels (also stored as acquisition data)

Logic analyzer qualifier channel	PCIExpress support package channel name	Description
M_QUAL:3	-	Unused
M_QUAL:2	-	Unused
M_QUAL:1	STP_packet_3	HIGH for TLPs
M_QUAL:0	FTS_Detect	HIGH for Fast-Training-Sequences

Table 4-236: PCIEx1 A channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A3:7	L00_data_02_L7	Lane 00 [02] delayed 7 clocks
M_A3:6	-	Unused
M_A3:5	-	Unused
M_A3:4	-	Unused
M_A3:3	-	Unused
M_A3:2	-	Unused
M_A3:1	-	Unused
M_A3:0	-	Unused
M_A2:7	-	Unused
M_A2:6	-	Unused
M_A2:5	-	Unused

Table 4-236: PCIEx1 A channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A2:4	-	Unused
M_A2:3	-	Unused
M_A2:2	-	Unused
M_A2:1	-	Unused
M_A2:0	L00_data_01_L7	Lane 00[01] delayed 7 clocks
M_A1:7	-	Unused
M_A1:6	-	Unused
M_A1:5	-	Unused
M_A1:4	Rule_viol_3	Indicates EDB symbol in sample
M_A1:3	Rule_viol_2	Physical layer rule checking
M_A1:2	Rule_viol_1	Physical layer rule checking
M_A1:1	Rule_viol_0	Physical layer rule checking
M_A1:0	STP_packet_2	HIGH when STP symbol in sample
M_A0:7	STP_packet_1	Combined with STP_packet_0
M_A0:6	STP_packet_0	indicate lane where STPsymbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:5	SDP_packet_2	HIGH when SDP symbol in sample
M_A0:4	SDP_packet_1	Combined with SDP_packet_0
M_A0:3	SDP_packet_0	Indicated lane where SDP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:2	Symbol_Table_Enable	HIGH when preprocessor indicates data is in 8-bit, descrambled mode and not in electrical idle.
M_A0:1	-	Unused
M_A0:0	END_EDB_detect	HIGH when sample contains an END or EDB symbol

Table 4-237: PCIEx1 D channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_D3:7	L00_data_03_L7	Lane 00 [03] delayed 7 clocks
M_D3:6	L00_data_04_L7	Lane 00 [04] delayed 7 clocks
M_D3:5	STP_cntr_03	STP counter bit 3
M_D3:4	STP_cntr_02	STP counter bit 2
M_D3:3	STP_cntr_01	STP counter bit 1
M_D3:2	STP_cntr_00	STP counter bit 0
M_D3:1	L00_data_00_L7	Lane 00[00] delayed 7 clocks
M_D3:0	STP_cntr_09	STP counter bit 9
M_D2:7	STP_cntr_08	STP counter bit 8
M_D2:6	STP_cntr_07	STP counter bit 7
M_D2:5	STP_cntr_06	STP counter bit 6
M_D2:4	STP_cntr_05	STP counter bit 5
M_D2:3	STP_cntr_04	STP counter bit 4
M_D2:2	STP_cntr_12	STP counter bit 12
M_D2:1	STP_cntr_11	STP counter bit 11
M_D2:0	STP_cntr_10	STP counter bit 10
M_D1:7	-	Unused
M_D1:6	-	Unused
M_D1:5	-	Unused
M_D1:4	-	Unused
M_D1:3	-	Unused
M_D1:2	-	Unused
M_D1:1	-	Unused
M_D1:0	-	Unused
M_D0:7	-	Unused
M_D0:6	-	Unused
M_D0:5	-	Unused
M_D0:4	-	Unused
M_D0:3	-	Unused
M_D0:2	-	Unused
M_D0:1	-	Unused
M_D0:0	-	Unused

Table 4-238: PCIEx1 C channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C3:7	Ten_bit_mode	Indicates the link was acquired in 10b mode
M_C3:6	-	Unused
M_C3:5	-	Unused
M_C3:4	-	Unused
M_C3:3	L00_data_00	Lane 00 [00]
M_C3:2	L00_data_01	Lane 00 [01]
M_C3:1	L00_data_10	Lane 00 [10]
M_C3:0	L00_data_11	Lane 00 [11]
M_C2:7	L00_data_02	Lane 00 [02]
M_C2:6	L00_data_03	Lane 00 [03]
M_C2:5	L00_data_04	Lane 00 [04]
M_C2:4	L00_data_05	Lane 00 [05]
M_C2:3	L00_data_06	Lane 00 [06]
M_C2:2	L00_data_07	Lane 00 [07]
M_C2:1	L00_data_08	Lane 00 [08]
M_C2:0	L00_data_09	Lane 00 [09]
M_C1:7	Elec_Idle_Flag	HIGH when in electrical idle
M_C1:6	L00_data_04_L6	Lane 00 [04] delayed 6 clocks
M_C1:5	L00_data_03_L6	Lane 00 [03] delayed 6 clocks
M_C1:4	-	Unused
M_C1:3	-	Unused
M_C1:2	-	Unused
M_C1:1	-	Unused
M_C1:0	-	Unused
M_C0:7	-	Unused
M_C0:6	-	Unused
M_C0:5	-	Unused
M_C0:4	-	Unused
M_C0:3	-	Unused
M_C0:2	-	Unused

Table 4-238: PCIEx1 C channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C0:1	-	Unused
M_C0:0	-	Unused

PCIEx2

Table 4-239: PCIEx2 Clock channels (also stored as acquisition data)

Logic analyzer Clock channel	PCIExpress support package channel name	Description
M_Clock:3	SDP_packet_3	HIGH for DLLP
M_Clock:2	TS_Detect	HIGH for training sequences TS1/TS2
M_Clock:1	clk	Clock Sync. To parallel data
M_Clock:0	SKP_Detect	HIGH for Skip-Ordered-Sets

Table 4-240: PCIEx2 QUAL channels (also stored as acquisition data)

Logic analyzer qualifier channel	PCIExpress support package channel name	Description
M_QUAL:3	-	Unused
M_QUAL:2	-	Unused
M_QUAL:1	STP_packet_3	HIGH for TLPs
M_QUAL:0	FTS_Detect	HIGH for Fast-Training-Sequences

Table 4-241: PCIEx2 A channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A3:7	L01_data_02_L4	Lane 01 [02] delayed 4 clocks
M_A3:6	-	Unused
M_A3:5	-	Unused
M_A3:4	-	Unused
M_A3:3	-	Unused
M_A3:2	-	Unused
M_A3:1	-	Unused
M_A3:0	-	Unused
M_A2:7	-	Unused
M_A2:6	-	Unused
M_A2:5	-	Unused

Table 4-241: PCIEx2 A channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A2:4	-	Unused
M_A2:3	-	Unused
M_A2:2	-	Unused
M_A2:1	-	Unused
M_A2:0	L01_data_01_L4	Lane 01[01] delayed 4 clocks
M_A1:7	L01_data_10	Lane 01 [10]
M_A1:6	L01_data_01	Lane 01[01]
M_A1:5	L01_data_00	Lane 01 [00]
M_A1:4	Rule_viol_3	Indicates EDB symbol in sample
M_A1:3	Rule_viol_2	Physical layer rule checking
M_A1:2	Rule_viol_1	Physical layer rule checking
M_A1:1	Rule_viol_0	Physical layer rule checking
M_A1:0	STP_packet_2	HIGH when STP symbol in sample
M_A0:7	STP_packet_1	Combined with STP_packet_0
M_A0:6	STP_packet_0	Indicates lane where STP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:5	SDP_packet_2	HIGH when SDP symbol in sample
M_A0:4	SDP_packet_1	Combined with SDP_packet_0
M_A0:3	SDP_packet_0	Indicates lane where SDP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:2	Symbol_Table_Enable	HIGH when preprocessor determines data is in 8-bit, descrambled mode and not in electrical idle.
M_A0:1	-	Unused
M_A0:0	END_EDB_detect	HIGH when sample contains an END or EDB symbol

Indicates the signal is asserted low.

Table 4-242: PCIEx2 D channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_D3:7	L01_data_03_L4	Lane 01 [03] delayed 4 clocks
M_D3:6	L01_data_04_L4	Lane 01 [04] delayed 4 clocks
M_D3:5	STP_cntr_03	STP counter bit 3
M_D3:4	STP_cntr_02	STP counter bit 2
M_D3:3	STP_cntr_01	STP counter bit 1
M_D3:2	STP_cntr_00	STP counter bit 0
M_D3:1	L01_data_00_L4	Lane 01[00] delayed 4 clocks
M_D3:0	STP_cntr_09	STP counter bit 9
M_D2:7	STP_cntr_08	STP counter bit 8
M_D2:6	STP_cntr_07	STP counter bit 7
M_D2:5	STP_cntr_06	STP counter bit 6
M_D2:4	STP_cntr_05	STP counter bit 5
M_D2:3	STP_cntr_04	STP counter bit 4
M_D2:2	STP_cntr_12	STP counter bit 12
M_D2:1	STP_cntr_11	STP counter bit 11
M_D2:0	STP_cntr_10	STP counter bit 10
M_D1:7	L01_data_11	Lane 01 [11]
M_D1:6	L01_data_02	Lane 01 [02]
M_D1:5	L01_data_03	Lane 01 [03]
M_D1:4	L01_data_04	Lane 01 [04]
M_D1:3	L01_data_05	Lane 01 [05]
M_D1:2	L01_data_06	Lane 01 [06]
M_D1:1	L01_data_07	Lane 01 [07]
M_D1:0	L01_data_08	Lane 01 [08]
M_D0:7	L01_data_09	Lane 01 [09]
M_D0:6	-	Unused
M_D0:5	-	Unused
M_D0:4	-	Unused
M_D0:3	-	Unused
M_D0:2	-	Unused
M_D0:1	-	Unused
M_D0:0	-	Unused

Table 4-243: PCIEx2 C channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C3:7	Ten_bit_mode	Indicates the data was acquired in 10b mode.
M_C3:6	-	Unused
M_C3:5	-	Unused
M_C3:4	-	Unused
M_C3:3	L00_data_00	Lane 00 [00]
M_C3:2	L00_data_01	Lane 00 [01]
M_C3:1	L00_data_10	Lane 00 [10]
M_C3:0	L00_data_11	Lane 00 [11]
M_C2:7	L00_data_02	Lane 00 [02]
M_C2:6	L00_data_03	Lane 00 [03]
M_C2:5	L00_data_04	Lane 00 [04]
M_C2:4	L00_data_05	Lane 00 [05]
M_C2:3	L00_data_06	Lane 00 [06]
M_C2:2	L00_data_07	Lane 00 [07]
M_C2:1	L00_data_08	Lane 00 [08]
M_C2:0	L00_data_09	Lane 00 [09]
M_C1:7	Elec_Idle_Flag	HIGH when in electrical idle
M_C1:6	L01_data_04_LLL	Lane 01 [04] delayed 3 clocks
M_C1:5	L01_data_03_LLL	Lane 01 [03] delayed 3 clocks
M_C1:4	-	Unused
M_C1:3	-	Unused
M_C1:2	-	Unused
M_C1:1	-	Unused
M_C1:0	-	Unused
M_C0:7	-	Unused
M_C0:6	-	Unused
M_C0:5	-	Unused
M_C0:4	-	Unused
M_C0:3	-	Unused
M_C0:2	-	Unused

Table 4-243: PCIEx2 C channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C0:1	-	Unused
M_C0:0	-	Unused

PCIEx4

Table 4-244: PCIEx4 Clock channels (also stored as acquisition data)

Logic analyzer clock channel	PCIExpress support package channel name	Description
M_Clock:3	SDP_packet_3	HIGH for DLLP
M_Clock:2	TS_Detect	HIGH for training sequences TS1/TS2
M_Clock:1	Clk	Clock Sync. To parallel data
M_Clock:0	SKP_Detect	SKP detect

Table 4-245: PCIEx4 QUAL channels (also stored as acquisition data)

Logic analyzer qualifier channel	PCIExpress support package channel name	Description
M_QUAL:3	-	Unused
M_QUAL:2	-	Unused
M_QUAL:1	STP_packet_3	HIGH for TLPs
M_QUAL:0	FTS_Detect	HIGH for Fast-Training-Sequences

Table 4-246: PCIEx4 A channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A3:7	L03_data_04_LL	Lane 03 [04] delayed 2 clocks
M_A3:6		Unused
M_A3:5		Unused
M_A3:4		Unused
M_A3:3		Unused
M_A3:2		Unused
M_A3:1		Unused
M_A3:0		Unused
M_A2:7		Unused
M_A2:6		Unused
M_A2:5		Unused

Table 4-246: PCIEx4 A channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A2:4		Unused
M_A2:3		Unused
M_A2:2		Unused
M_A2:1		Unused
M_A2:0	L03_data_03_LL	Lane 03[04] delayed by 2 clocks
M_A1:7	L01_data_10	Lane 01 [10]
M_A1:6	L01_data_01	Lane 01 [01]
M_A1:5	L01_data_00	Lane 01 [00]
M_A1:4	Rule_viol_3	Indicates EDB symbol in sample
M_A1:3	Rule_viol_2	Physical layer rule checking
M_A1:2	Rule_viol_1	Physical layer rule checking
M_A1:1	Rule_viol_0	Physical layer rule checking
M_A1:0	STP_packet_2	HIGH when STP symbol in sample
M_A0:7	STP_packet_1	Combined with STP_packet_0
M_A0:6	STP_packet_0	Indicates lane where STP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:5	SDP_packet_2	HIGH when SDP symbol in sample
M_A0:4	SDP_packet_1	Combined with SDP_packet_0
M_A0:3	SDP_packet_0	Indicates lane where SDP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:2	Symbol_Table_Enable	HIGH when preprocessor determines data is in 8-bit, descrambled mode and not in electrical idle.
M_A0:1	L03_data_02	Lane 03 [02]
M_A0:0	END_EDB_detect	HIGH when sample contains an END or EDB symbol

Table 4-247: PCIEx4 D channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_D3:7	-	Unused
M_D3:6	-	Unused
M_D3:5	STP_cntr_03	STP counter bit 3
M_D3:4	STP_cntr_02	STP counter bit 2
M_D3:3	STP_cntr_01	STP counter bit 1
M_D3:2	STP_cntr_00	STP counter bit 0
M_D3:1	L03_data_02_LL	Lane 03[02] delayed 2 clocks
M_D3:0	STP_cntr_09	STP counter bit 9
M_D2:7	STP_cntr_08	STP counter bit 8
M_D2:6	STP_cntr_07	STP counter bit 7
M_D2:5	STP_cntr_06	STP counter bit 6
M_D2:4	STP_cntr_05	STP counter bit 5
M_D2:3	STP_cntr_04	STP counter bit 4
M_D2:2	-	Unused
M_D2:1	-	Unused
M_D2:0	STP_cntr_10	STP counter bit 10
M_D1:7	L01_data_11	Lane 01 [11]
M_D1:6	L01_data_02	Lane 01 [02]
M_D1:5	L01_data_03	Lane 01 [03]
M_D1:4	L01_data_04	Lane 01 [04]
M_D1:3	L01_data_05	Lane 01 [05]
M_D1:2	L01_data_06	Lane 01 [06]
M_D1:1	L01_data_07	Lane 01 [07]
M_D1:0	L01_data_08	Lane 01 [08]
M_D0:7	L01_data_09	Lane 01 [09]
M_D0:6	L02_data_00	Lane 02 [00]
M_D0:5	L02_data_01	Lane 02 [01]
M_D0:4	L02_data_10	Lane 02 [10]
M_D0:3	L02_data_11	Lane 02 [11]
M_D0:2	L02_data_02	Lane 02 [02]
M_D0:1	L02_data_03	Lane 02 [03]
M_D0:0	L02_data_04	Lane 02 [04]

Table 4-248: PCIEx4 C channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C3:7	Ten_bit_mode	Indicates the data was acquired in 10b mode.
M_C3:6	L03_data_09	Lane 03 [09]
M_C3:5	L03_data_08	Lane 03 [08]
M_C3:4	L03_data_07	Lane 03 [07]
M_C3:3	L00_data_00	Lane 00 [00]
M_C3:2	L00_data_01	Lane 00 [01]
M_C3:1	L00_data_10	Lane 00 [10]
M_C3:0	L00_data_11	Lane 00 [11]
M_C2:7	L00_data_02	Lane 00 [02]
M_C2:6	L00_data_03	Lane 00 [03]
M_C2:5	L00_data_04	Lane 00 [04]
M_C2:4	L00_data_05	Lane 00 [05]
M_C2:3	L00_data_06	Lane 00 [06]
M_C2:2	L00_data_07	Lane 00 [07]
M_C2:1	L00_data_08	Lane 00 [08]
M_C2:0	L00_data_09	Lane 00 [09]
M_C1:7	Elec_Idle_Flag	HIGH when in electrical idle
M_C1:6	L03_data_01_LL	Lane 03 [01] delayed 2 clocks
M_C1:5	L03_data_00_LL	Lane 03 [00] delayed 2 clocks
M_C1:4	L03_data_11	Lane 03 [11]
M_C1:3	L03_data_10	Lane 03 [10]
M_C1:2	L03_data_01	Lane 03 [01]
M_C1:1	L03_data_00	Lane 03 [00]
M_C1:0	L02_data_09	Lane 02 [09]
M_C0:7	L02_data_08	Lane 02 [08]
M_C0:6	L03_data_06	Lane 03 [06]
M_C0:5	L03_data_03	Lane 03 [03]
M_C0:4	L02_data_07	Lane 02 [07]
M_C0:3	L02_data_06	Lane 02 [06]
M_C0:2	L02_data_05	Lane 02 [05]

Table 4-248: PCIEx4 C channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C0:1	L03_data_04	Lane 03 [04]
M_C0:0	L03_data_05	Lane 03 [05]

PCIEx8

Table 4-249: PCIEx8 Clock channels (also stored as acquisition data)

channel	PCIExpress support package channel name	Description
M_Clock:3	SDP_packet_3	HIGH for DLLP
M_Clock:2	TS_Detect	HIGH for training sequences TS1/TS2
M_Clock:1	clk	Clock Sync. To parallel data
M_Clock:0	SKP_Detec	Lane 03 [02] delayed one clock

Table 4-250: PCIEx8 QUAL channels (also stored as acquisition data)

Logic analyzer qualifier channel	PCIExpress support package channel name	Description
M_QUAL:3	L06_data_05	Lane 06 [05]
M_QUAL:2	L05_data_10	Lane 05 [10]
M_QUAL:1	STP_packet_3	HIGH for TLPs
M_QUAL:0	FTS_Detect	HIGH for Fast-Training-Sequences

Table 4-251: PCIEx8 A channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A3:7	L03_data_04_L	Lane 03 [04] delayed 1 clock
M_A3:6	L04_data_00	Lane 04 [00]
M_A3:5	L04_data_01	Lane 04 [01]
M_A3:4	L04_data_10	Lane 04 [10]
M_A3:3	L04_data_11	Lane 04 [11]
M_A3:2	L04_data_02	Lane 04 [02]
M_A3:1	L04_data_03	Lane 04 [03]
M_A3:0	L04_data_04	Lane 04 [04]
M_A2:7	L04_data_05	Lane 04 [05]
M_A2:6	L04_data_06	Lane 04 [06]

Table 4-251: PCIEx8 A channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A2:5	L04_data_07	Lane 04 [07]
M_A2:4	L04_data_08	Lane 04 [08]
M_A2:3	L04_data_09	Lane 04 [09]
M_A2:2	L05_data_00	Lane 05 [00]
M_A2:1	L05_data_01	Lane 05 [01]
M_A2:0	L03_data_03_L	Lane 03 [03] delayed 1 clock
M_A1:7	L01_data_10	Lane 01 [10]
M_A1:6	L01_data_01	Lane 01 [01]
M_A1:5	L01_data_00	Lane 01 [00]
M_A1:4	Rule_viol_3	Indicates EDB symbol in sample
M_A1:3	Rule_viol_2	Physical layer rule checking
M_A1:2	Rule_viol_1	Physical layer rule checking
M_A1:1	Rule_viol_0	Physical layer rule checking
M_A1:0	STP_packet_2	HIGH when STP symbol in sample
M_A0:7	STP_packet_1	Combined with STP_packet_0
M_A0:6	STP_packet_0	Indicates lane where STP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:5	SDP_packet_2	HIGH when SDP symbol in sample
M_A0:4	SDP_packet_1	Combined with SDP_packet_0
M_A0:3	SDP_packet_0	Indicates lane where SDP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:2	Symbol_Table_Enable	HIGH when preprocessor determines data is in 8-bit, descrambled mode and not in electrical idle.
M_A0:1	L03_data_02	Lane 03 [02]
M_A0:0	END_EDB_detect	HIGH when sample contains an END or EDB symbol

Table 4-252: PCIEx8 D channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_D3:7	L07_data_00_L	Lane 07 [00] delayed 1 clock
M_D3:6	L07_data_01_L	Lane 07 [01] delayed 1 clock
M_D3:5	STP_cntr_03	STP counter bit 3
M_D3:4	STP_cntr_02	STP counter bit 2
M_D3:3	STP_cntr_01	STP counter bit 1
M_D3:2	STP_cntr_00	STP counter bit 0
M_D3:1	L03_data_02_L	Lane 03 Data 02 delayed by 1 clk
M_D3:0	STP_cntr_09	STP counter bit 9
M_D2:7	STP_cntr_08	STP counter bit 8
M_D2:6	STP_cntr_07	STP counter bit 7
M_D2:5	STP_cntr_06	STP counter bit 6
M_D2:4	STP_cntr_05	STP counter bit 5
M_D2:3	STP_cntr_04	STP counter bit 4
M_D2:2	L07_data_04_L	Lane 07 [04] delayed 1 clock
M_D2:1	L07_data_03_L	Lane 07 [03] delayed 1 clock
M_D2:0	L07_data_02_L	Lane 07 [02] delayed 1 clock
M_D1:7	L01_data_11	Lane 01 [11]
M_D1:6	L01_data_02	Lane 01 [02]
M_D1:5	L01_data_03	Lane 01 [03]
M_D1:4	L01_data_04	Lane 01 [04]
M_D1:3	L01_data_05	Lane 01 [05]
M_D1:2	L01_data_06	Lane 01 [06]
M_D1:1	L01_data_07	Lane 01 [07]
M_D1:0	L01_data_08	Lane 01 [08]
M_D0:7	L01_data_09	Lane 01 [09]
M_D0:6	L02_data_00	Lane 02 [00]
M_D0:5	L02_data_01	Lane 02 [01]
M_D0:4	L02_data_10	Lane 02 [10]
M_D0:3	L02_data_11	Lane 02 [11]
M_D0:2	L02_data_02	Lane 02 [02]

Table 4-252: PCIEx8 D channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_D0:1	L02_data_03	Lane 02 [03]
M_D0:0	L02_data_04	Lane 02 [04]

Table 4-253: PCIEx8 C channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C3:7	Ten_bit_mode	Indicates the data was acquired in 10b mode.
M_C3:6	L03_data_09	Lane 03 [09]
M_C3:5	L03_data_08	Lane 03 [08]
M_C3:4	L03_data_07	Lane 03 [07]
M_C3:3	L00_data_00	Lane 00 [00]
M_C3:2	L00_data_01	Lane 00 [01]
M_C3:1	L00_data_10	Lane 00 [10]
M_C3:0	L00_data_11	Lane 00 [11]
M_C2:7	L00_data_02	Lane 00 [02]
M_C2:6	L00_data_03	Lane 00 [03]
M_C2:5	L00_data_04	Lane 00 [04]
M_C2:4	L00_data_05	Lane 00 [05]
M_C2:3	L00_data_06	Lane 00 [06]
M_C2:2	L00_data_07	Lane 00 [07]
M_C2:1	L00_data_08	Lane 00 [08]
M_C2:0	L00_data_09	Lane 00 [09]
M_C1:7	Elec_Idle_Flag	HIGH when in electrical idle
M_C1:6	L03_data_01_L	Lane 03 [01] delayed 1 clock
M_C1:5	L03_data_00_L	Lane 03 [00] delayed 1 clock
M_C1:4	L03_data_11	Lane 03 [11]
M_C1:3	L03_data_10	Lane 03 [10]
M_C1:2	L03_data_01	Lane 03 [01]
M_C1:1	L03_data_00	Lane 03 [00]
M_C1:0	L02_data_09	Lane 02 [09]

Table 4-253: PCIEx8 C channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C0:7	L02_data_08	Lane 02 [08]
M_C0:6	L03_data_06	Lane 03 [06]
M_C0:5	L03_data_03	Lane 03 [03]
M_C0:4	L02_data_07	Lane 02 [07]
M_C0:3	L02_data_06	Lane 02 [06]
M_C0:2	L02_data_05	Lane 02 [05]
M_C0:1	L03_data_04	Lane 03 [04]
M_C0:0	L03_data_05	Lane 03 [05]

Table 4-254: PCIEx8 E channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_E3:7	L06_data_04	Lane 06 [04]
M_E3:6	L06_data_03	Lane 06 [03]
M_E3:5	L06_data_02	Lane 06 [02]
M_E3:4	L06_data_11	Lane 06 [11]
M_E3:3	L06_data_10	Lane 06 [10]
M_E3:2	L06_data_01	Lane 06 [01]
M_E3:1	L06_data_00	Lane 06 [00]
M_E3:0	L05_data_09	Lane 05 [09]
M_E2:7	L05_data_08	Lane 05 [08]
M_E2:6	L05_data_07	Lane 05 [07]
M_E2:5	L05_data_06	Lane 05 [06]
M_E2:4	L05_data_05	Lane 05 [05]
M_E2:3	L05_data_04	Lane 05 [04]
M_E2:2	L05_data_03	Lane 05 [03]
M_E2:1	L05_data_02	Lane 05 [02]
M_E2:0	L05_data_11	Lane 05 [11]
M_E1:7	L06_data_07	Lane 06 [07]
M_E1:6	L06_data_08	Lane 06 [08]
M_E1:5	L06_data_09	Lane 06 [09]

Table 4-254: PCIEx8 E channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_E1:4	L07_data_00	Lane 07 [00]
M_E1:3	L07_data_01	Lane 07 [01]
M_E1:2	L07_data_10	Lane 07 [10]
M_E1:1	L07_data_11	Lane 07 [11]
M_E1:0	L07_data_02	Lane 07 [02]
M_E0:7	L07_data_03	Lane 07 [03]
M_E0:6	L07_data_04	Lane 07 [04]
M_E0:5	L07_data_05	Lane 07 [05]
M_E0:4	L07_data_06	Lane 07 [06]
M_E0:3	L07_data_07	Lane 07 [07]
M_E0:2	L07_data_08	Lane 07 [08]
M_E0:1	L07_data_09	Lane 07 [09]
M_E0:0	L06_data_06	Lane 06 [06]

PCIEx16

Table 4-255: PCIEx16 Clock channels (also stored as acquisition data)

Logic analyzer Clock channel	PCIExpress support package channel name	Description
M_Clock:3	SDP_packet_3	HIGH for DLLP
M_Clock:2	TS_Detect	HIGH for training sequences TS1/TS2
M_Clock:1	clk	Clock Sync. To parallel data
M_Clock:0	SKP_Detect	HIGH for SKP-ordered sets

Table 4-256: PCIEx16 QUAL channels (also stored as acquisition data)

Logic analyzer qualifier channel	PCIExpress support package channel name	Description
M_QUAL:3	L06_data_05	Lane 06 [05] data
M_QUAL:2	L05_data_10	Lane 05 [10] data
M_QUAL:1	STP_packet_3	HIGH for TLPs
M_QUAL:0	FTS_Detect	HIGH for Fast-Training-Sequences

Table 4-257: PCIEx16 A channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A3:7	L11_data_04_L	Lane 11 [04] delayed 1 clock
M_A3:6	L04_data_00	Lane 04 [00]
M_A3:5	L04_data_01	Lane 04 [01]
M_A3:4	L04_data_10	Lane 04 [10]
M_A3:3	L04_data_11	Lane 04 [11]
M_A3:2	L04_data_02	Lane 04 [02]
M_A3:1	L04_data_03	Lane 04 [03]
M_A3:0	L04_data_04	Lane 04 [04]
M_A2:7	L04_data_05	Lane 04 [05]
M_A2:6	L04_data_06	Lane 04 [06]
M_A2:5	L04_data_07	Lane 04 [07]
M_A2:4	L04_data_08	Lane 04 [08]
M_A2:3	L04_data_09	Lane 04 [09]
M_A2:2	L05_data_00	Lane 05 [00]
M_A2:1	L05_data_01	Lane 05 [01]
M_A2:0	L11_data_03_L	Lane 11 [03] delayed 1 clock
M_A1:7	L01_data_10	Lane 01 [10]
M_A1:6	L01_data_01	Lane 01 [01]
M_A1:5	L01_data_00	Lane 01 [00]
M_A1:4	Rule_viol_3	Indicates EDB symbol in sample
M_A1:3	Rule_viol_2	Physical layer rule checking
M_A1:2	Rule_viol_1	Physical layer rule checking

Table 4-257: PCIEx16 A channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_A1:1	Rule_viol_0	Physical layer rule checking
M_A1:0	STP_packet_2	HIGH when STP symbol in sample
M_A0:7	STP_packet_1	Combined with STP_packet_0
M_A0:6	STP_packet_0	Indicates lane where STP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:5	SDP_packet_2	HIGH when SDP symbol in sample
M_A0:4	SDP_packet_1	Combined with SDP_packet_0
M_A0:3	SDP_packet_0	Indicates lane where SDP symbol was detected. IE: 00 – Lane00, 01 – Lane04, 10 – Lane08, 11 – Lane12
M_A0:2	Symbol_Table_Enable	HIGH when preprocessor determines data is in 8-bit, descrambled mode and not in electrical idle.
M_A0:1	L03_data_02	Lane 03 [02]
M_A0:0	END_EDB_detect	HIGH when sample contains an END or EDB symbol
# indicates the signal is asserted low.		

Table 4-258: PCIEx16 D channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_D3:7	L15_data_00_L	Lane 15 [00] delayed 1 clock
M_D3:6	L15_data_01_L	Lane 15 [01] delayed 1 clock
M_D3:5	STP_cntr_03	STP counter bit 3
M_D3:4	STP_cntr_02	STP counter bit 2
M_D3:3	STP_cntr_01	STP counter bit 1
M_D3:2	STP_cntr_00	STP counter bit 0
M_D3:1	L11_data_02_L	Lane 11 [02] delayed 1 clock
M_D3:0	0	Unused

Table 4- 258: PCIEx16 D channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_D2:7	STP_cntr_08	STP counter bit 8
M_D2:6	STP_cntr_07	STP counter bit 7
M_D2:5	STP_cntr_06	STP counter bit 6
M_D2:4	STP_cntr_05	STP counter bit 5
M_D2:3	STP_cntr_04	STP counter bit 4
M_D2:2	STP_packet_1_L	STP_packet_1 delayed 1 clock
M_D2:1	STP_packet_0_L	STP_packet_0 delayed by 1 clock
M_D2:0	STP_packet_2_L	STP_packet_2 delayed 1 clock
M_D1:7	L01_data_11	Lane 01 [11]
M_D1:6	L01_data_02	Lane 01 [02]
M_D1:5	L01_data_03	Lane 01 [03]
M_D1:4	L01_data_04	Lane 01 [04]
M_D1:3	L01_data_05	Lane 01 [05]
M_D1:2	L01_data_06	Lane 01 [06]
M_D1:1	L01_data_07	Lane 01 [07]
M_D1:0	L01_data_08	Lane 01 [08]
M_D0:7	L01_data_09	Lane 01 [09]
M_D0:6	L02_data_00	Lane 02 [00]
M_D0:5	L02_data_01	Lane 02 [01]
M_D0:4	L02_data_10	Lane 02 [10]
M_D0:3	L02_data_11	Lane 02 [11]
M_D0:2	L02_data_02	Lane 02 [02]
M_D0:1	L02_data_03	Lane 02 [03]
M_D0:0	L02_data_04	Lane 02 [04]

Table 4-259: PCIEx16 C channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C3:7	Ten_bit_mode	Indicates the data was acquired in 10b mode.
M_C3:6	L03_data_09	Lane 03 [09]
M_C3:5	L03_data_08	Lane 03 [08]
M_C3:4	L03_data_07	Lane 03 [07]
M_C3:3	L00_data_00	Lane 00 [00]
M_C3:2	L00_data_01	Lane 00 [01]
M_C3:1	L00_data_10	Lane 00 [10]
M_C3:0	L00_data_11	Lane 00 [11]
M_C2:7	L00_data_02	Lane 00 [02]
M_C2:6	L00_data_03	Lane 00 [03]
M_C2:5	L00_data_04	Lane 00 [04]
M_C2:4	L00_data_05	Lane 00 [05]
M_C2:3	L00_data_06	Lane 00 [06]
M_C2:2	L00_data_07	Lane 00 [07]
M_C2:1	L00_data_08	Lane 00 [08]
M_C2:0	L00_data_09	Lane 00 [09]
M_C1:7	Elec_Idle_Flag	HIGH when in electrical idle
M_C1:6	L11_data_01_L	Lane 11 [01] delayed 1 clock
M_C1:5	L11_data_00_L	Lane 11 [00] delayed 1 clock
M_C1:4	L03_data_11	Lane 03 [11]
M_C1:3	L03_data_10	Lane 03 [10]
M_C1:2	L03_data_01	Lane 03 [01]
M_C1:1	L03_data_00	Lane 03 [00]
M_C1:0	L02_data_09	Lane 02 [09]
M_C0:7	L02_data_08	Lane 02 [08]
M_C0:6	L03_data_06	Lane 03 [06]
M_C0:5	L03_data_03	Lane 03 [03]
M_C0:4	L02_data_07	Lane 02 [07]
M_C0:3	L02_data_06	Lane 02 [06]
M_C0:2	L02_data_05	Lane 02 [05]

Table 4-259: PCIEx16 C channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_C0:1	L03_data_04	Lane 03 [04]
M_C0:0	L03_data_05	Lane 03 [05]

Table 4-260: PCIEx16 E channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_E3:7	L06_data_04	Lane 06 [04]
M_E3:6	L06_data_03	Lane 06 [03]
M_E3:5	L06_data_02	Lane 06 [02]
M_E3:4	L06_data_11	Lane 06 [11]
M_E3:3	L06_data_10	Lane 06 [10]
M_E3:2	L06_data_01	Lane 06 [01]
M_E3:1	L06_data_00	Lane 06 [00]
M_E3:0	L05_data_09	Lane 05 [09]
M_E2:7	L05_data_08	Lane 05 [08]
M_E2:6	L05_data_07	Lane 05 [07]
M_E2:5	L05_data_06	Lane 05 [06]
M_E2:4	L05_data_05	Lane 05 [05]
M_E2:3	L05_data_04	Lane 05 [04]
M_E2:2	L05_data_03	Lane 05 [03]
M_E2:1	L05_data_02	Lane 05 [02]
M_E2:0	L05_data_11	Lane 05 [11]
M_E1:7	L06_data_07	Lane 06 [07]
M_E1:6	L06_data_08	Lane 06 [08]
M_E1:5	L06_data_09	Lane 06 [09]
M_E1:4	L07_data_00	Lane 07 [00]
M_E1:3	L07_data_01	Lane 07 [01]
M_E1:2	L07_data_10	Lane 07 [10]
M_E1:1	L07_data_11	Lane 07 [11]
M_E1:0	L07_data_02	Lane 07 [02]
M_E0:7	L07_data_03	Lane 07 [03]

Table 4-260: PCIEx16 E channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
M_E0:6	L07_data_04	Lane 07 [04]
M_E0:5	L07_data_05	Lane 07 [05]
M_E0:4	L07_data_06	Lane 07 [06]
M_E0:3	L07_data_07	Lane 07 [07]
M_E0:2	L07_data_08	Lane 07 [08]
M_E0:1	L07_data_09	Lane 07 [09]
M_E0:0	L06_data_06	Lane 06 [06]

Table 4-261: PCIEx16 Clock channels (also stored as acquisition data)

channel	PCIExpress support package channel name	Description
S_Clock:3	L09_data_10	Lane 09 [10]
S_Clock:2	L12_data_00	Lane 12 [00]
S_Clock:1	L11_data_09	Lane 11 [09]
S_Clock:0	L14_data_08	Lane 14 [08]
# indicates the signal is asserted low.		

Table 4-262: PCIEx16 QUAL channels (also stored as acquisition data)

Logic analyzer qualifier channel	PCIExpress support package channel name	Description
S_QUAL:3	STP_cntr_06_Slave	Copy of STP counter bit 6
S_QUAL:2	STP_cntr_07_Slave	Copy of STP counter bit 7
S_QUAL:1	L09_data_01	Lane 09 [01]
S_QUAL:0	L14_data_07	Lane 14 [07]

Table 4-263: PCIEx16 Slave A channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
S_A3:7	L14_data_09	Lane 03 [09]
S_A3:6	L15_data_00	Lane 15 [00]
S_A3:5	L15_data_01	Lane 15 [01]
S_A3:4	L15_data_10	Lane 15 [10]
S_A3:3	L15_data_11	Lane 15 [11]
S_A3:2	L15_data_02	Lane 15 [02]
S_A3:1	L15_data_03	Lane 15 [03]
S_A3:0	L15_data_04	Lane 15 [04]
S_A2:7	L15_data_05	Lane 15 [05]
S_A2:6	L15_data_06	Lane 15 [06]
S_A2:5	L15_data_07	Lane 15 [07]
S_A2:4	L15_data_08	Lane 15 [08]
S_A2:3	L15_data_09	Lane 15 [09]
S_A2:2	-	Unused
S_A2:1	-	Unused
S_A2:0	-	Unused
S_A1:7	L11_data_08	Lane 11 [08]
S_A1:6	L11_data_07	Lane 11 [07]
S_A1:5	L11_data_06	Lane 11 [06]
S_A1:4	L11_data_05	Lane 11 [05]
S_A1:3	L11_data_04	Lane 11 [04]
S_A1:2	L11_data_03	Lane 11 [03]
S_A1:1	L11_data_02	Lane 11 [02]
S_A1:0	L11_data_11	Lane 11 [11]
S_A0:7	L11_data_10	Lane 11 [10]
S_A0:6	L11_data_01	Lane 11 [01]
S_A0:5	L11_data_00	Lane 11 [00]
S_A0:4	L10_data_09	Lane 10 [09]
S_A0:3	L10_data_08	Lane 10 [08]
S_A0:2	L10_data_07	Lane 10 [07]
S_A0:1	L10_data_06	Lane 10 [06]
S_A0:0	L10_data_05	Lane 10 [05]

Table 4-264: PCIEx16 Slave D channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
S_D3:7	L14_data_06	Lane 14 [06]
S_D3:6	L14_data_05	Lane 14 [05]
S_D3:5	L14_data_04	Lane 14 [04]
S_D3:4	L14_data_03	Lane 14 [03]
S_D3:3	L14_data_02	Lane 14 [02]
S_D3:2	L14_data_11	Lane 14 [11]
S_D3:1	L14_data_10	Lane 14 [10]
S_D3:0	L14_data_01	Lane 14 [01]
S_D2:7	L14_data_00	Lane 14 [00]
S_D2:6	L13_data_09	Lane 13 [09]
S_D2:5	L13_data_08	Lane 13 [08]
S_D2:4	L13_data_07	Lane 13 [07]
S_D2:3	L13_data_06	Lane 13 [06]
S_D2:2	L13_data_05	Lane 13 [05]
S_D2:1	L13_data_04	Lane 13 [04]
S_D2:0	L13_data_03	Lane 13 [03]
S_D1:7	L12_data_01	Lane 12 [01]
S_D1:6	L12_data_10	Lane 12 [10]
S_D1:5	L12_data_11	Lane 12 [11]
S_D1:4	L12_data_02	Lane 12 [02]
S_D1:3	L12_data_03	Lane 12 [03]
S_D1:2	L12_data_04	Lane 12 [04]
S_D1:1	L12_data_05	Lane 12 [05]
S_D1:0	L12_data_06	Lane 12 [06]
S_D0:7	L12_data_07	Lane 12 [07]
S_D0:6	L12_data_08	Lane 12 [08]
S_D0:5	L12_data_09	Lane 12 [09]
S_D0:4	L13_data_00	Lane 13 [00]
S_D0:3	L13_data_01	Lane 13 [01]
S_D0:2	L13_data_10	Lane 13 [10]

Table 4-264: PCIEx16 Slave D channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
S_D0:1	L13_data_11	Lane 13 [11]
S_D0:0	L13_data_02	Lane 13 [02]

Table 4-265: PCIEx16 C channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
S_C3:7	L09_data_11	Lane 09 [11]
S_C3:6	L09_data_02	Lane 09 [02]
S_C3:5	L09_data_03	Lane 09 [03]
S_C3:4	L09_data_04	Lane 09 [04]
S_C3:3	L09_data_05	Lane 09 [05]
S_C3:2	L09_data_06	Lane 09 [06]
S_C3:1	L09_data_07	Lane 09 [07]
S_C3:0	L09_data_08	Lane 09 [08]
S_C2:7	L09_data_09	Lane 09 [09]
S_C2:6	L10_data_00	Lane 10 [00]
S_C2:5	L10_data_01	Lane 10 [01]
S_C2:4	L10_data_10	Lane 10 [10]
S_C2:3	L10_data_11	Lane 10 [11]
S_C2:2	L10_data_02	Lane 10 [02]
S_C2:1	L10_data_03	Lane 10 [03]
S_C2:0	L10_data_04	Lane 10 [04]
S_C1:7	L09_data_00	Lane 09 [00]
S_C1:6	L08_data_09	Lane 08 [09]
S_C1:5	L08_data_08	Lane 08 [08]
S_C1:4	L08_data_07	Lane 08 [07]
S_C1:3	L08_data_06	Lane 08 [06]
S_C1:2	L08_data_05	Lane 08 [05]
S_C1:1	L08_data_04	Lane 08 [04]
S_C1:0	L08_data_03	Lane 08 [03]
S_C0:7	L08_data_02	Lane 08 [02]

Table 4-265: PCIEx16 C channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
S_C0:6	L08_data_11	Lane 08 [11]
S_C0:5	L08_data_10	Lane 08 [10]
S_C0:4	L08_data_01	Lane 08 [01]
S_C0:3	L08_data_00	Lane 08 [00]
S_C0:2	L15_data_04_L	Lane 15 [04] delayed 1 clock
S_C0:1	L15_data_03_L	Lane 15 [03] delayed 1 clock
S_C0:0	L15_data_02_L	Lane 15 [02] delayed 1 clock

Table 4-266: PCIEx16 E channels

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
S_E3:7	STP_cntr_05_Slave	Copy of STP_cntr_05
S_E3:6	STP_cntr_04_Slave	Copy of STP_cntr_04
S_E3:5	STP_cntr_03_Slave	Copy of STP_cntr_03
S_E3:4	STP_cntr_02_Slave	Copy of STP_cntr_02
S_E3:3	STP_cntr_01_Slave	Copy of STP_cntr_01
S_E3:2	STP_cntr_00_Slave	Copy of STP_cntr_00
S_E3:1	SDP_packet_3_Slave	Copy of SDP_packet_3
S_E3:0	SDP_packet_2_Slave	Copy of SDP_packet_2
S_E2:7	SDP_packet_1_Slave	Copy of SDP_packet_1
S_E2:6	SDP_packet_0_Slave	Copy of SDP_packet_0
S_E2:5	Rule_viol_3_Slave	Copy of Rule_viol_3
S_E2:4	Rule_viol_2_Slave	Copy of Rule_viol_2
S_E2:3	Rule_viol_1_Slave	Copy of Rule_viol_1
S_E2:2	Rule_viol_0_Slave	Copy of Rule_viol_0
S_E2:1	-	Unused
S_E2:0	-	Unused
S_E1:7	STP_cntr_08_Slave	Copy of STP_cntr_08
S_E1:6	STP_cntr_09_Slave	Copy of STP_cntr_09
S_E1:5	STP_packet_0_Slave	Copy of STP_packet_0
S_E1:4	STP_packet_1_Slave	Copy of STP_packet_1

Table 4-266: PCIEx16 E channels (Cont.)

Logic analyzer acquisition channel	PCIExpress support package channel name	Description
S_E1:3	STP_packet_2_Slave	Copy of STP_packet_2
S_E1:2	STP_packet_3_Slave	Copy of STP_packet_3
S_E1:1	Symbol_Table_Enable_Slave	Copy of Symbol_Table_Enable
S_E1:0	Ten_bit_mode_Slave	Copy of Ten_bit_mode
S_E0:7	TS_Detect_Slave	Copy of TS_Detect
S_E0:6	-	Unused
S_E0:5	-	Unused
S_E0:4	-	Unused
S_E0:3	-	Unused
S_E0:2	-	Unused
S_E0:1	-	Unused
S_E0:0	-	Unused

Replaceable Parts

Replaceable Parts

This section contains a list of the replaceable parts for the TMS817 and TMS818 PCIExpress support product.

Parts Ordering Information

Replacement parts are available through your local Tektronix field office or representative.

Changes to Tektronix products are sometimes made to accommodate improved components as they become available and to give you the benefit of the latest improvements. Therefore, when ordering parts, it is important to include the following information in your order.

- Part number
- Instrument type or model number
- Instrument serial number
- Instrument modification number, if applicable

If you order a part that has been replaced with a different or improved part, your local Tektronix field office or representative will contact you concerning any change in part number.

Abbreviations

Abbreviations conform to American National Standard ANSI Y1.1–1972.

Mfr. Code to Manufacturer Cross Index

The table titled Manufacturers Cross Index shows codes, names, and addresses of manufacturers or vendors of components listed in the parts list.

Manufacturers cross index

Mfr. code	Manufacturer	Address	City, state, zip code
060D9	TENSOLITE COMPANY	PRECISION HARNESS AND ASSEMBLY~3000 COLUMBIA HOUSE BLVD~#120	VANCOUVER, WA 98661
0KB01	STAUFFER SUPPLY CO	810 SE SHERMAN	PORTLAND, OR 97214-4657
0ZJL1	INTERCON SYSTEMS INC	2800 COMMERCE DRIVE	HARRISBURG, PA 17110
5Y400	TRIAX METAL PRODUCTS INC	1880 SW MERLO DRIVE	BEAVERTON, OR 97006
80009	TEKTRONIX INC	14150 SW KARL BRAUN DR~PO BOX 500	BEAVERTON, OR 97077-0001
TK0588	UNIVERSAL PRECISION PRODUCT	1775 NW CORNELIUS PASS RD	HILLSBORO, OR 97124
TK2548	XEROX CORPORATION	14181 SW MILLIKAN WAY	BEAVERTON, OR 97005
TK6585	TRESKE PRECISION MACHINE INC	14140 SW GALBREATH DRIVE	SHERWOOD, OR 97140
0J9P9	GEROME MFG CO INC	PO BOX 737~403 NORTH MAIN	NEWBERG, OR 97132

Fig. & index number	Tektronix part number	Serial no. effective	Serial no. discont'd	Qty	Name & description	Mfr. code	Mfr. part number
6-1-1	012-1661-50			1	CABLE ASSEMBLY W/LABELS;INTCON,SHLD RIBBON,72L;TMS817	80009	012-1661-50
-2	426-2619-00			2	MOUNTING ASSEMBLY;KEYED ASSY, GOLD & UNKEYED ASSY,TIN;TMS817	TK0588	426-2619-00
-3	131-7433-00			1	CONTACT;SPRING CASING;TMS817	0ZJL1	8305-003
-	131-7502-00*			1	CONTACT;SPRING CASING;TMS818	0ZJL1	131750200
-4	020-2508-00			1	ACCESSORY KIT; SLOT X16;TMS817P	80009	020-2508-00
-	020-2509-00*			1	ACCESSORY KIT; SLOT X1;TMS817P	80009	020-2509-00
-	020-2510-00*			1	ACCESSORY KIT; SLOT X4;TMS817P	80009	020-2510-00
-	020-2511-00*			1	ACCESSORY KIT; SLOT X8;TMS817P	80009	020-2511-00
-5	407-4947-00			1	BRACKET,SUPPORT; 0.040 ELECTRO GALVANIZED STEEL,SAFETY CONTROLLED	0J9P9	407-4947-00
-6	211-0373-00			2	SCREW,MACHINE; 4-40 X 0.250,PNH,STL CD PLT,T10	0KB01	211-0373-00
-7 **	672-5668-50			1	CIRCUIT BD ASSY; TMS817 MID-BUS PROBE W/CABLES;TESTED	80009	672-5668-50
-	672-5758-50*			1	CIRCUIT BD ASSY;TMS818 MID-BUS PROBE W/CABLES,TESTED	80009	672-5758-50
-8	174-4826-00			1	CABLE ASSY; REF CLOCK;48 " LONG;TMS817 & TMS818	060D9	174-4826-00
-9	344-0604-00			1	CLIP,CABLE;X16,BOTTOM;ABS PLASTIC;TMS817	5Y400	344-0604-00
-10	344-0603-00			1	CLIP,CABLE;X16,TOP;ABS PLASTIC;TMS817	5Y400	344-0603-00
-11	211-0012-00			4	SCREW,MACHINE; 4-40 X 0.375,PNH,STL CD PL,POZ.	0KB01	211-0012-00
-	344-0605-00*			1	CLIP,CABLE;TOP;ABS PLASTIC;TMS818	TK6585	344-0605-00
-	344-0606-00*			1	CLIP,CABLE;BOTTOM;ABS PLASTIC;TMS818	TK6585	344-0606-00
					STANDARD ACCESSORIES		
	071-1214-XX			1	MANUAL,TECH; INSTRUCTION,3GIO SOFTWARE/ HARDWARE;TMS817 & TMS818	TK2548	071-1214-XX
	161-0104-00			1	CABLE ASSEMBLY:3,18 AWG, 98 L, 250V/10AMP, RIGHT ANGEL, IEC320, NEMA 15-5P, WITH CORD GRIP, US,	S3109	ORDER BY DESCRIPTION
					OPTIONAL ACCESSORIES		
	161-0104-05			1	CA ASSY,PWR:3,1.0MM SQ,250V/10A,2.5 METER,RTANG,IEC320,RCPT,AUSTRALIA	TK1373	ORDER BY DESCRIPTION
	161-0104-06			1	CA ASSY,PWR:3,1.0MM SQ,250V/10A,2.5 METER,RTANG,IEC320,RCPT,EUROPEAN	TK1373	ORDER BY DESCRIPTION
	161-0104-07			1	CA ASSY,PWR:3,1.0MM SQ,240V/10A,2.5 METER,RTANG,IEC320,RCPT X 13A,FUSED,UK PLUG,(13A FUSE),U	TK2541	ORDER BY DESCRIPTION
	161-0167-00			1	CA ASSY,PWR:3,0.75MM SQ,250V/10A,2.5 METER,RTANG,IEC320,RCPT,SWISS,NO CORD GRIP	S3109	ORDER BY DESCRIPTION

*** Not shown**

**** For exchange only; for an additional midbus probe head with attaching parts, see page 1-4 for ordering information.**

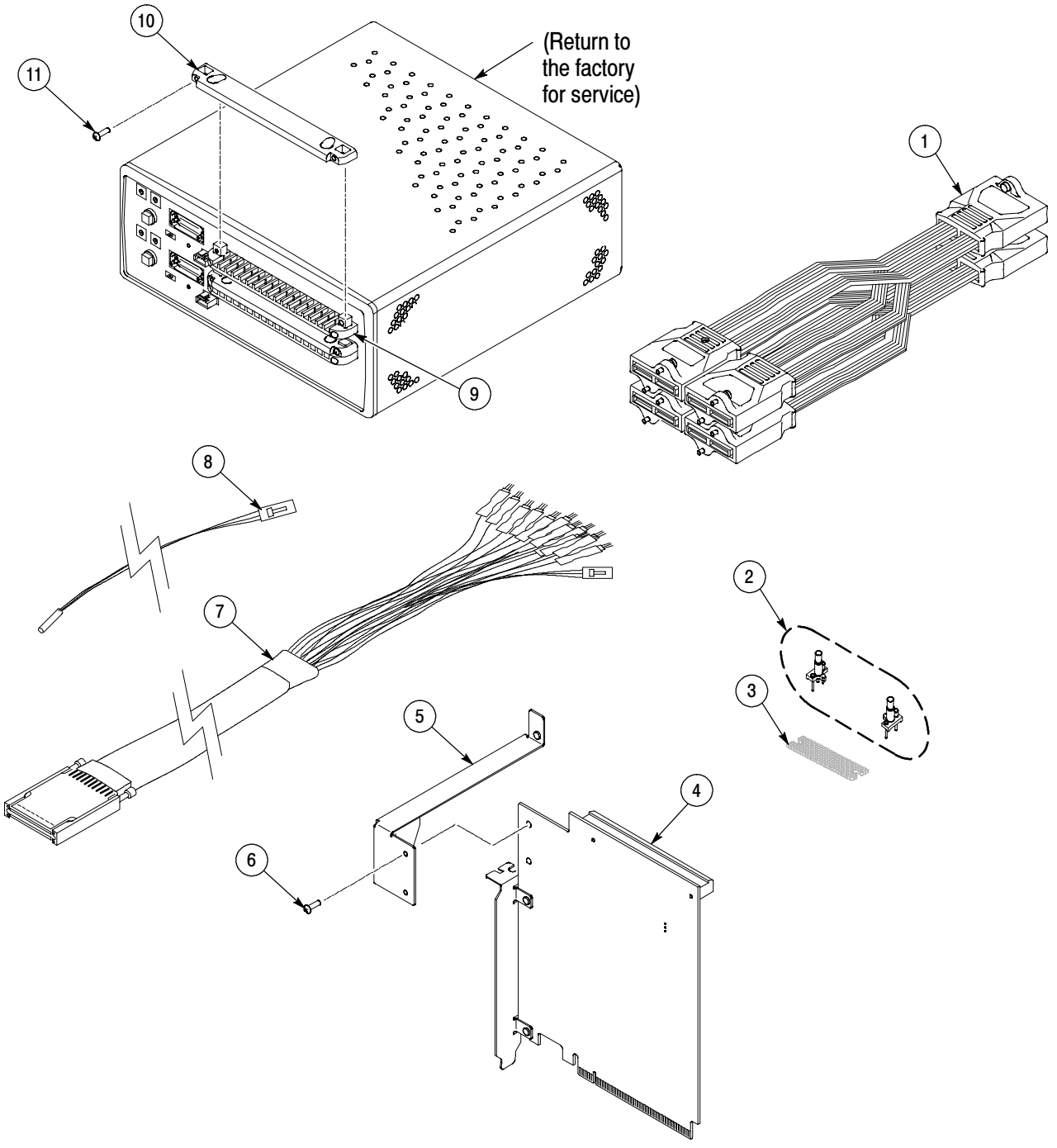

Figure 5- 1: Probe adapter exploded view

Glossary

Glossary

by-1, x1
A Link or Port with one physical Lane.

by-8, x8
A Link or Port with eight physical Lanes.

DLLPs (Data Link Layer Packet)
A packet generated in the data link layer to support link management functions.

Downstream
Indicates the information is flowing away from the Root Complex.

Lane
A set of differential signal pairs, one pair for transmission and one pair for reception.

Link
The collection of two ports and their interconnecting Lanes. A link is a dual-simplex communications path between two components.

Root Complex
An entity that includes a host bridge and one or more Root Ports.

Root Port
A PCI Express Port on a Root Complex that maps a portion of the hierarchy through an associated virtual PCI-PCI Bridge.

TLPs (Transaction Layer Packet)
A packet generated in the transaction layer to convey a request or transaction completion.

Upstream
Indicates information flows towards the Root Complex.

Index

Index

L

M

P

R

S

T

U

V

W

X