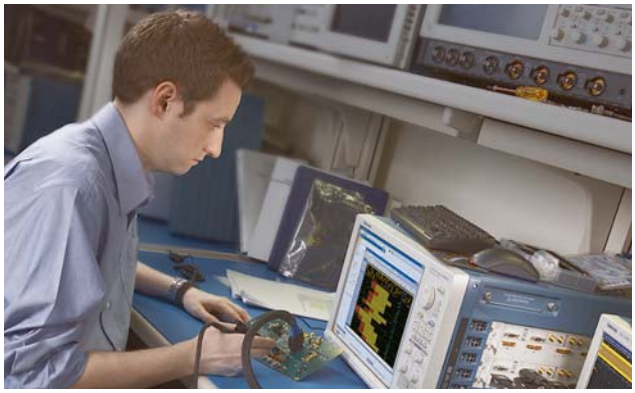
PCI Express Power State Management Validation and Debug Using the Tektronix Logic Protocol Analyzer

Debug the Toughest PCIe Challenges, Including Active State Power Management

PCI Express and Power Management



The importance of Active State Power Management (ASPM) has grown with the increasing emphasis on “green” products and energy conservation. ASPM plays a role not only in mobile devices but also in desktop and server systems as well. Consequently the validation of power management features in all types of computing platforms has taken on a new level of urgency.

Power management has been one of the most complex issues to debug in PCIe Gen1.0, Gen 2.0, and continues to be a challenge for PCIe Gen 3.0. The Logic Protocol Analyzer is the tool of choice for analyzing events that occur during the transitions to and from power management states. Inevitably some of the signals of interest can appear on one or more PCI Express buses within a system, which necessitates a Logic Protocol Analyzer that can carry out uncompromised acquisition of multiple PCI Express links or other busses.

Logic Protocol Analyzers

The preferred tool for measuring the logical sub-block of the PHY, the Data Link, Packet, and Transaction layers of PCI Express is the Logic Protocol Analyzer (LPA). Unlike real-time oscilloscopes, LPAs provide protocol disassembly of all layers of the link with packet level, symbol level, and link event triggering across all lanes of the link. The purpose of the LPA is to simplify acquisition and analysis of the digital aspects of the serial data. To carry out the serial bus debug mission, the LPA must deliver features consistent with the needs of high-speed buses with speeds up to 8GT/s: physical layer acquisition, deep memory, flexible triggering, synchronization with other system buses and more, all while offering low impact probing tools that provide direct insight into physical layer signals.

The most critical requirement for observing the change of power management states is the ability to synchronize quickly to the link as power states are exited. The transition passes quickly and a tool that is slow to respond might miss the very cycles in which a problem occurs.

PCI Express Power States

To understand the transitions from state to state there needs to be an understanding of the different power states.

L0 - Active normal operating mode state.

L0s - Energy saving “standby” state with fast recovery back to L0.

L1 - Lower power “standby” state but with a longer recovery than L0s.

L2 - Auxiliary-powered deep-energy-saving state.

L3 - Link Off state.

Power Management

Consider the scenario where a Transaction Layer Packet (TLP) Configuration Write is writing an incorrect value to a register, which in turn causes abnormal system behavior. This type of problem typically occurs during hardware/software integration and may only occur when power management is enabled. This is because power state changes are stressful to the transmitter and the receiver therefore can cause some link quality issues that show up in the form of logic errors. In order to capture this event, the analyzer must acquire all TLPs as the bus is exiting L0s and returning to L0. Since L0s is meant to be capable of fast recovery it is important to only send as many packets that are necessary to return to L0. Therefore any test equipment on that bus should lock on and acquire almost all of the advertised number of Fast Training Sequences (FTS) as the link exits out of electrical idle in order to not miss the first packet after returning to L0. The Tektronix LPA typically tracks exit from electrical idle after observing approximately four FTS at PCIe Gen 3 rates and within six FTS at PCIe Gen 1 and 2 rates even if electrical idle is longer that 2µs! Locking to data this quickly prevents losing valuable transactions that occur at the beginning of the L0 state. See Figure 1.

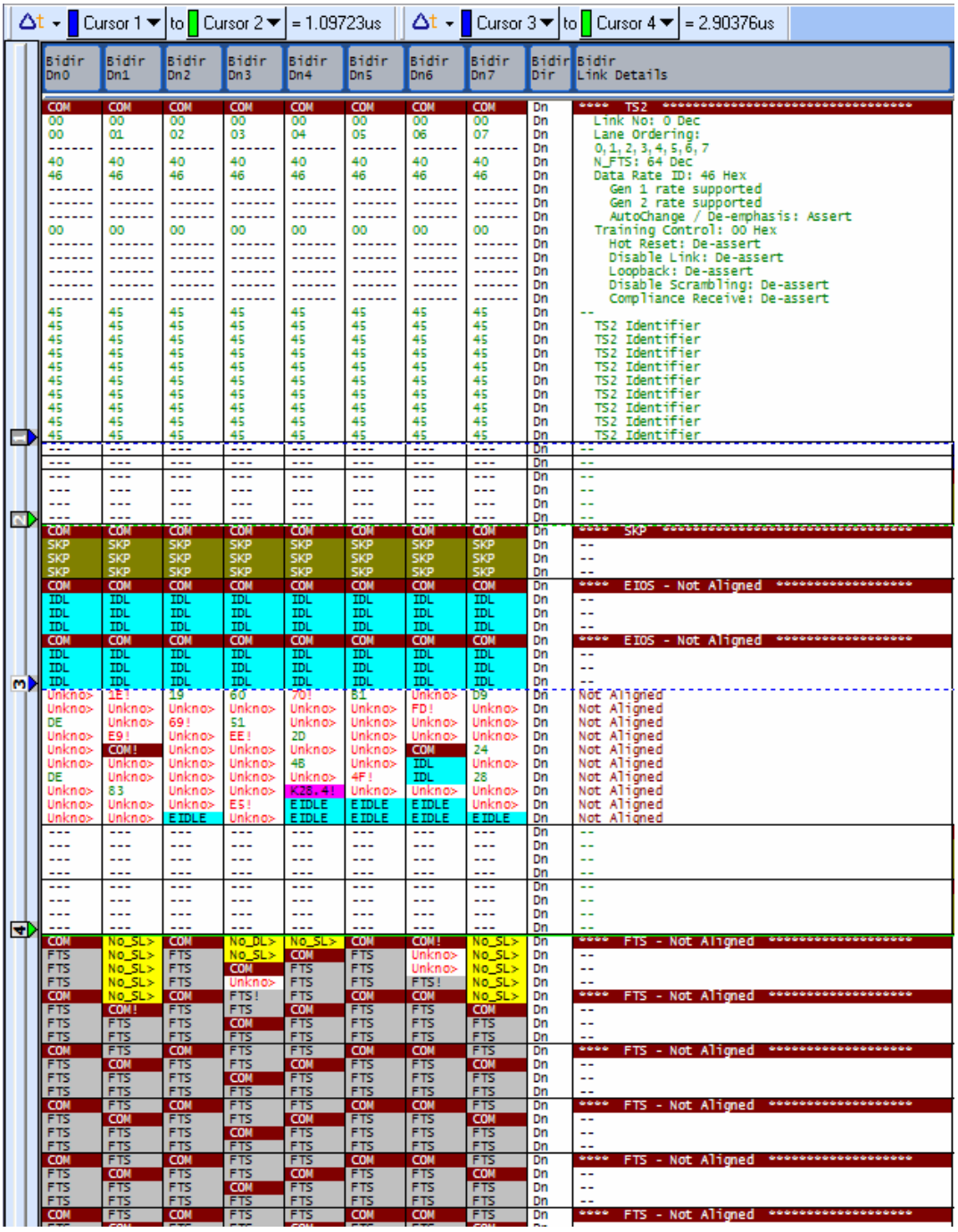


Figure 1. Illustrates the process to transition from L0 to L1. Link entering L0s at 5.0Gb/s.

Since L0s is meant to be a low latency, fast recovery state and independent of what the other direction is doing, the system does not need to prepare or do any communication with any other device. L1 on the other hand is more complicated but it saves more power than L0s.

As the link exits electrical idle, FTS are transmitted. The number of FTS that will be transmitted is determined during link training. If the analyzer was unable to lock to the data as the link exited from electrical idle, the TLP Configuration Read packet would not be acquired by the analyzer and it would be impossible to verify that the register value in the packet was the incorrect register. In the example illustrated in Figure 1 over 1.1µs passed between the last transaction (in this case a Training Sequence 2 or TS2) and L0s entry. The link was in L0s for 2.9µs before starting to wake up. From the Upstream TS2 (see Figure 2) we know that the Downstream needs to send 31 FTS to recover from L0s and in this case the Tektronix Logic Protocol Analyzer captured 29 of them.

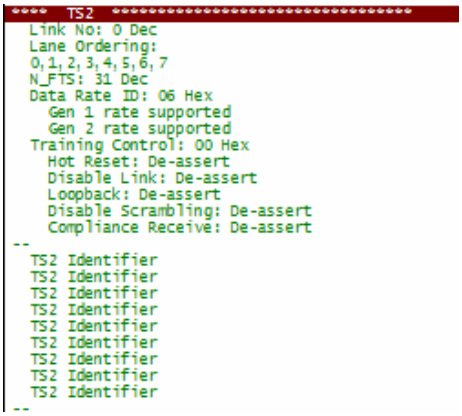


Figure 2. TS2 from the Upstream

Downstream	**Upstream**
	Begins L1 transition process. Accumulates minimum credits and blocks scheduling of new TLPs Waits to receive Ack for last TLP. PM_Enter_L1 DLLPs sent repeatedly.
Blocks scheduling of new TLPs.	
	Waits for PM_Request_Ack DLLP, acknowledging the PM_Enter_L1 DLLP.
Receives acknowledgment for last TLP. Sends PM_Request_Ack DLLP repeatedly until it sees electrical idle.	
	Sees PM_Request_Ack DLLP, disables DLLP, TLP transmission and brings Physical Layer to electrical idle.
Completes L1 transition: disables DLLP, TLP transmission and brings Physical Layer to electrical idle.	

Table 1. Transaction flow for a transition into L1.

Figure 3 shows the start of the transition from L0 to L1. First you can see the Upstream send an Update Flow Control packet. Once it completes the last Completion with Data and receives the Acknowledge, it starts sending out repetitive requests to go to L1. During the time that it takes to go from Figure 3 to Figure 4 the Downstream finally responds and within two packets the Upstream goes to L1 indicated by a blue line at the gap. The downstream then sends a few more L1 acknowledges before dropping to L1 itself.

TLA Timestamp		Link	PacketType	STP_Seq	AckNak_	Fmt	T
004:742:741	+	SA 1_Up	Ack		0x81F		
004:742:792	+	SA 1_Up	UpdateFC-NP				
004:743:051	+	SA 1_Up	CplD	0x0672		0x2	0x
004:743:347	+	SA 1_Dn	Ack		0x672		
004:744:148	+	SA 1_Dn	MRd(32)	0x0820		0x0	0x
004:744:169	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:181	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:182	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:183	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:184	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:196	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:197	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:198	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:199	+	SA 1_Up	PM_Active_State_Request_L1				

Figure 3. Start of the transition into L1.

TLA Timestamp		Link	PacketType	STP_Seq	AckNak_	Fmt	TC
004:744:638	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:639	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:640	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:641	+	SA 1_Up	PM_Active_State_Request_L1				
004:744:643	+	SA 1_Dn	PM_Request_Ack				
004:744:645	+	SA 1_Dn	PM_Request_Ack				
004:744:658	+	SA 1_Dn	PM_Request_Ack				
004:744:660	+	SA 1_Dn	PM_Request_Ack				
004:744:673	+	SA 1_Dn	PM_Request_Ack				
004:744:675	+	SA 1_Dn	PM_Request_Ack				
004:744:692	+	SA 1_Dn	PM_Request_Ack				
004:744:693	+	SA 1_Dn	PM_Request_Ack				
004:744:707	+	SA 1_Dn	PM_Request_Ack				
004:744:708	+	SA 1_Dn	PM_Request_Ack				
004:744:722	+	SA 1_Dn	PM_Request_Ack				
004:744:723	+	SA 1_Dn	PM_Request_Ack				

Figure 4. The downstream responding to the L1 request

Summary

ASPM continues to be one place where a lot of debug issues are showing up. With the demand for ASPM going up the need for a tool that can accurately capture these events goes up. Measurement tools, ranging from real-time oscilloscopes to logic protocol analyzers and signal sources, help engineers deal with PCI Express measurement challenges. These solutions deliver the performance to capture, display, and analyze the most complex serial signals. Thanks to these innovative, automated tools, engineers can perform debug and validate designs quickly and easily which enables faster time to market.