PowerSpan II™ Device Errata and Design Notes

80A1010_ER001_13

November 4, 2009

6024 Silver Creek Valley Road San Jose, California 95138

Telephone: (408) 284-8200 • FAX: (408) 284-3572

Printed in U.S.A.

1. Introduction

This document describes errata and design notes for both the revisions of the Single-PCI (CA91L8260B) and Dual-PCI (CA91L8200B) PowerSpan II devices. The first revision PowerSpan II devices have a Z at the end of the part number (that is CA91L8200B-100CEZ), the second revisions have a Z1, and the third have a Z2.

2. Errata Listing

Table 1 provides a summary list of the PowerSpan II device errata.

Table 1: Summary of Device Errata

Errata #	Description	Errata Corrected In Single PCI PowerSpan II (CA91L8260B)	Errata Corrected In Dual PCI PowerSpan II (CA91L8200B)
1	"Incorrect Discard Timer Operation" on page 5	No	No
2	"Incorrect Operation of P1_REQ64_ During 64-bit Reads" on page 6	No	No
3	"Incorrect Overwriting of EEPROM Devices Through I^2C" on page 6	No	No
4	"Incorrect Response During PCI Configuration Cycles with Px_BSI20" on page 7	No	No
5	"PowerSpan II as a Peripheral Device Does Not Tolerate FRAME_ and GNT_ Assertion on the Same Clock" on page 7 In version 80A1010_ER001_02 of the PowerSpan II Errata and Design Notes, the errata # 3.5 was covered in detail. However, further investigation shows the PowerSpan II is compatible with *PCI Specification 2.2*. The issue is no longer a PowerSpan II errata.	N/A	N/A
6	"Processor Bus Lockup When Using the Dual PCI PowerSpan II" on page 8	N/A	Yes
7	"JTAG TAP Issue in Bypass Mode" on page 9	Yes	Yes

Table 1: Summary of Device Errata

Errata #	Description	Errata Corrected In Single PCI PowerSpan II (CA91L8260B)	Errata Corrected In Dual PCI PowerSpan II (CA91L8200B)
8	“DMA Issues” on page 10	No	No
9	“PowerSpan II Initiates Incorrect Cycles on the PCI-2 Bus” on page 11	N/A	Yes
10	“Single PCI PowerSpan II PCI Parity Error Issue” on page 12 This errata was an issue for the prototype PowerSpan II. It has been corrected in the production version of the device. This errata description is for customer reference only.	N/A	N/A
11	“Single PCI PowerSpan II Initiates Incorrect Cycles on the PCI-1 bus” on page 13	Yes	N/A
12	“Clock Resolution of PCI Latency Timer” on page 13	No	No
13	“Overlapping PCI Address Images In Master-based Decode Mode” on page 14	No	No
14	“PCI AC Timing Specification Violations: P1_RST_ to P1_REQ64_ Minimum Hold Time” on page 15	No	No
15	“Incorrect Response to PCI-1 Fast Back-to-back Cycles Composed with a Different Transfer Size” on page 15	No	No

2.1 Errata Descriptions

This section describes errata and errata solutions for the PowerSpan II device.

2.1.1 Incorrect Discard Timer Operation

A processor may perform a single read to the PowerSpan II's PB slave channel to initiate a PCI memory or IO read. If the PCI target device, as a response to the memory or IO read, continuously retries PowerSpan II (delay >2^{15} PB_CLKs) and returns the data with a long delay PowerSpan II fails to return the data to the processor. The Transfer Acknowledge (TA_) signal is not asserted to the processor.

The internal discard timer within PowerSpan II times-out and causes the internal state machines to enter an error state. Because of this, PowerSpan II discards the data returned. Since the processor never sees a TA_, it continues to re-initiate the cycle on the processor bus causing PowerSpan II to generate another PCI memory or IO read. Only when this cycle completes within 2^{15} PB_CLKs does the PowerSpan II return the data correctly.

Software Work-around

Ensure that the PowerSpan II's internal discard timer does not time out. The system designer must make sure that the PCI Target is able to return data within 2^{15} PB_CLKs.

Alternatively, a watchdog timer could be used to reset the system within 2^{15} PB_CLKs.

2.1.2 Incorrect Operation of P1_REQ64_ During 64-bit Reads

When PowerSpan II is used in a mixed 32-bit and 64-bit PCI environment, and a write to a 32-bit PCI target is immediately followed by a 64-bit read to a 64-bit PCI target, PowerSpan II does not assert P1_REQ64_. The read is retried and PowerSpan II re-issues that 64-bit read and then correctly assert P1_REQ64_. This occurrence can confuse the 64-bit target into thinking the second read is a new read. If this happens an infinite retry condition could occur. The unknown condition depends on how the target responds to the signal combination (for example, the target could respond with a infinite retry).

Software Work-around for Single PCI PowerSpan II

When only a PB slave image is used to initiate reads or writes on the PCI-1 bus of the Single PCI PowerSpan II it is possible to use software control the transaction ordering on PB bus side. In a mixed 32-bit and 64-bit PCI environment ensure that any 32-bit write to a 32-bit PCI target at address offset 0x00 is followed by a 32-bit dummy write at address offset 0x04 before initiating a 64-bit read to a 64-bit PCI target.

Software Work-around for Dual PCI PowerSpan II

If the Dual PCI PowerSpan II is in a mixed 32-bit and 64-bit PCI environment there is no way to control the transaction ordering out of the PCI-1 master channel. An optional work-around is to use PowerSpan II PCI-1 bus in 32-bit mode only by disconnecting its P1_REQ64 signal from the bus and pulling-up P1_REQ64.

2.1.3 Incorrect Overwriting of EEPROM Devices Through I^2C

During an EEPROM load, if the PowerSpan II receives a reset during the write to the address field of the EEPROM, on the next rising edge of PO_RESET when performing the subsequent EEPROM load, PowerSpan II incorrectly overwrites the contents of the EEPROM.

The reason for this issue is because PowerSpan II I^2C interface does not reset the bus when PowerSpan II is reset. The I^2C bus can stay in write mode when the chip reset occurs. Because the I^2C bus is in write mode, when the first transaction occurs it is seen as a write and the EEPROM is overwritten.

Hardware Work-around

Some types of EEPROM provide a “Write Protect pin” for hardware data protection. (For example, ATMEL AT24C02A).The designer can implement external logic on the board to control the voltage on “Write Protect pin” to avoid unexpected writes during the runtime.

2.1.4 Incorrect Response During PCI Configuration Cycles with Px_BSI20

If an external agent is performing PCI configuration cycles to another PCI device and the base address programmed in Px_BSI20 of the PowerSpan II matches the PCI configuration address (such as AD[31:11]), then PowerSpan II incorrectly responds to that cycle.

For example, if the PowerSpan II P1_IDSEL_ signal is connected to AD[31] of the PCI bus and P1_BSI20 is programmed as 0x1000_0000, then PowerSpan II responds correctly to the configuration cycle when AD[31] is toggled high. However, the PowerSpan II also incorrectly respond to the configuration cycle when AD[28] is high, which corresponds to address programmed in P1_BSI20.

2.1.4.1 Software Work-around

The solution to this issue is to make sure that the value programmed in the P1_BSI20 (or P2_BSI20 in the Dual PCI PowerSpan II) registers does not match a valid PCI configuration address. This is done by programming more than one 1 in the bit fields [0:20] of P1_BSI20 (or P2_BSI20).

A second solution is to disable the PowerSpan II's register image by clearing the BSREG_BAR_EN bit in the P1_MISC_CSR (or P2_MISC_CSR) register prior to initiating a PCI configuration cycle.

2.1.5 PowerSpan II as a Peripheral Device Does Not Tolerate FRAME_ and GNT_ Assertion on the Same Clock

In version 80A1010_ER001_02 of the *PowerSpan II Errata and Design Notes*, errata # 5 was covered in detail. However, further investigation shows the PowerSpan II is compatible with *PCI Specification 2.2*. The issue is no longer a PowerSpan II errata.

2.1.6 Processor Bus Lockup When Using the Dual PCI PowerSpan II

When operating the Dual PCI PowerSpan II (CA91L8200B-100xxZ) under nominal voltage and temperature conditions, PowerSpan II does not reliably transfer data as a master to the processor (60x) bus. The failure to reliably transfer data applies to both PCI initiated cycles and DMA operation.

Symptoms of a failure include missed assertions of key signals and complete lockup of the processor and the PowerSpan II Processor Bus Interface. Operation of the device in modes which do not master the processor (60x) bus have not been shown to exhibit this problem.

Hardware Work-around

The problem has been shown to be dependant on the I/O and core voltages. This issue has been corrected in the Dual PCI PowerSpan II Revision Z1 device.

Work-arounds exist to adjust the I/O and/or core voltages as described in one of the scenarios below. When one of these work-arounds is implemented the failing cycles on the processor (60x) bus do not occur.

- An adjustment of the I/O Voltage from 3.3V to 3.0V, core remains at nominal voltage of 2.5V
- An adjustment of the core Voltage from 2.5V to 3.1V, I/O remains at the nominal voltage of 3.3V
- An adjustment of the core Voltage to 2.7V and the I/O Voltage to 3.1V

Note 1: IDT is unable to guarantee long term operation of the silicon when the core voltage is used out of specification. Therefore, if a work-around is implemented which uses the core voltage out of specification then the Dual PCI PowerSpan II device should only be used for internal design verification.

Note 2: These work-arounds assume room temperature operating conditions. We have observed that the issue is temperature dependant and failures are more likely to occur at elevated temperatures.

2.1.7 JTAG TAP Issue in Bypass Mode

The PowerSpan II JTAG TAP problem is a result of a timing sensitivity that exists between the input signals to the JTAG TDO output buffer. The PowerSpan II JTAG TAP can be configured to be in Bypass mode.

One state traversed by the state machine to get into Bypass mode is “capture-DR”. During capture DR a logic 0 is loaded, and this appears at TDO when the state machine reaches “shift-DR” state. The problem is in PowerSpan II this logic 0 does not appear at TDO, instead a logic 1 appears. If the TAP is made to stay in “shift-DR” state, then whatever is in TDI should appear at TDO one test clock (TCK) cycle later. In PowerSpan II, TDO does not always propagate the logic level on TDI when TDI is a constant high (1). There is no problem when TDI is low (0) or when TDI toggles between high and low.

Hardware Work-around

The hardware solution is to jumper TDI to TDO externally to the PowerSpan II on the board.

This issue has been corrected in the production version of the PowerSpan II.

2.1.8 DMA Issues

There are two DMA issues that have been discovered in PowerSpan II. Each issue is described in detail in the following sections.

2.1.8.1 DMA Source Bus and Destination Bus Lockup

When a DMA transaction begins, the individual interfaces involved in the transaction (source and destination busses) are blocked from receiving transactions from any other source until the DMA Block Size (DBS) is reached (see “Software Work-around”). In situations where a DMA transaction has been initiated and the data is not available (for example, DMA source bus gets continuous retries) the DMA channel will be locked until the transaction completes or until the retry counter expires if the source bus is the PCI bus. The PCI Target Channel will not be available until the DMA transaction completes.

Software Work-around

Change the value of the DMA Block Size (DBS) field in the PowerSpan II DMA General Control and Status register (DMA_GCSR). By default, the DMA engine always tries to transfer 128 bytes of data and blocks both the source and destination interface until the transaction is complete.

2.1.8.2 DMA Transaction and Retry Timer Expiration Issue

If the DMA transaction receives continuous retries on the source bus until the retry timer expires, the PowerSpan II will write 8 bytes on the destination bus at the original address where the transaction was retried. The data that is written to the destination bus is incorrect.

Software Work-around

In order to avoid experiencing this errata, the system must have the retry timer programed not to expire. PowerSpan II has a MAX_RETRY field in the PCI-1 and PCI-2 Miscellaneous Control and Status registers that enable the user to select the number of retries that are allowed on the PCI busses. The default setting is to retry the transaction forever.

2.1.8.3 DMA Deadlock Condition

A DMA deadlock condition can happen when two transactions occur in the following sequence:

1. PowerSpan II's DMA is used to read from PCI host memory to MPC8260 local memory
2. The PCI host directly writes to the MPC8260 local memory through PowerSpan II's PCI target channel.

Because the PCI host strictly follows PCI bus transaction ordering, it must complete the second write transaction (described above) before it provides the first read transaction data (described above). However, due to a PowerSpan II DMA errata (see “DMA Source Bus and Destination Bus Lockup” on page 10) the DMA blocks the PB master channel once the DMA starts. This causes the second write transaction to be unable to complete until the DMA request is fulfilled. When this occurs PowerSpan II’s DMA is deadlocked.

Work-around

The second write transaction must be completed with the following steps:

1. The PCI host prepares the data to be written for the second transaction to somewhere in its host memory space
2. The PCI host write to PowerSpan II’s internal register to setup DMA1 to fetch the prepared data from PCI host memory space to MPC8260 local memory.

2.1.9 PowerSpan II Initiates Incorrect Cycles on the PCI-2 Bus

When operating the Dual PCI PowerSpan II device under nominal voltage and temperature conditions, the PowerSpan II does not reliably transfer data as a PCI master to the 32-bit, PCI-2 bus. The symptom of a failure is the missed assertion of a P2_AD[xx] signal (most likely P2_AD[15]).

Both of the following criteria must occur in order to experience this errata:

- PowerSpan II retried the previous cycle on the PCI-2 bus
- The level of P2_AD[xx] was a logic high before PowerSpan II initiated the cycle.

Hardware Work-around

The problem has been shown to be dependant on the I/O and Core voltages.

The work-around consists of adjusting the I/O and/or Core voltages. When the following work-around is implemented, the failing cycles on the PCI-2 bus do not occur.

- An adjustment of the Core voltage and/or the I/O voltage so that the delta between the two voltages is less than, or equal to, 0.3V. (Core voltage must not exceed 3.0V).

Note 1: Adjusting the Core voltage out of the normal operating range has a long-term reliability impact on silicon.

This issue has been corrected in the production version of the PowerSpan II.

2.1.10 Single PCI PowerSpan II PCI Parity Error Issue

When the Single PCI PowerSpan II is the PCI master and performing writes, there is the possibility for PowerSpan II to drive the P1_PAR_ signal. This signal causes the target device to generate a parity error. The parity error is generated even though there are no errors occurring during the cycles.

Hardware Work-around

This issue was corrected in the Z1 version of the Single PowerSpan II. Replacing the Z version of the device with the Z1 version corrects this errata.

Hardware Work-around

The problem has been shown to be dependant on the I/O and Core voltages.

The work-around consists of adjusting the I/O and/or Core voltages. When the following work-around is implemented, the failing cycles on the PCI bus do not occur.

- An adjustment of the Core voltage and/or the I/O voltage so that the delta between the two voltages is less than, or equal to, 0.3V. (Core voltage must not exceed 3.0V).

Note: Adjusting the Core voltage out of the normal operating range has a long-term reliability impact on the silicon.

2.1.11 Single PCI PowerSpan II Initiates Incorrect Cycles on the PCI-1 bus

When the PowerSpan II PCI master interface is initiated by one of the PowerSpan II's DMA engines the PowerSpan II potentially does not reliably transfer data as a PCI master to the 64-bit PCI-1 bus. The symptom of a failure is the missed assertion of a P1_AD[xx] signal (most likely P1_AD[3], P1_AD[24], or P1_AD[27]).

The PowerSpan II PCI master interface must be initiated by one of the PowerSpan II DMA engine in order to experience this errata. An indication that a system is experiencing this errata is multiple PCI parity errors.

Hardware Work-around

This issue was corrected in the Z2 version of the Single PCI PowerSpan II. Replacing the Z or Z1 version of the device with the Z2 version corrects this errata.

Hardware Work-around

The problem has been shown to be dependant on the I/O and Core voltages.

The work-around consists of adjusting the I/O and/or Core voltages. When the following work-around is implemented, the failing cycles on the PCI bus do not occur.

- An adjustment of the Core voltage and/or the I/O voltage so that the delta between the two voltages is less than, or equal to, 0.3V. (Core voltage must not exceed 3.0V).

Note: Adjusting the Core voltage out of the normal operating range has a long-term reliability impact on the silicon.

2.1.12 Clock Resolution of PCI Latency Timer

If bit 3 of the 8-bit latency timer (LTIMER) field in the Px_MISC0 register is zero, then the resolution of the LTIMER field is eight PCI clocks instead of one PCI clock. For example, if the latency timer is programmed to 0x17, the actual value of the latency timer is 0x10. The lower 3 bits (LTIMER[2:0]) become zero. If the latency timer is programmed to 0x1F, the latency timer has the correct value of 0x1F because bit 3 is equal to one.

2.1.13 Overlapping PCI Address Images In Master-based Decode Mode

In a multiple PCI master PowerSpan II system, when the address images use overlapping memory space the Master-based decode feature of PowerSpan II must be enabled. When Master-based decode is enabled, the PCI target images only claim a transaction decoded for its specified physical address space if it originates from a specific PCI master. External bus masters are selected for a specific target image by setting the bits in the PCI x Target Image x Translation Address register. This feature can only be enabled when the PowerSpan II's PCI Arbiter is enabled.

However, when the PCI address images use overlapping memory space an issue can arise when two masters request the same data from PowerSpan II. PowerSpan II retries the first master and fetches the data from the destination bus. However, if a second master initiates a read transaction after PowerSpan II has the fetched data, and the transaction has the same address as the originally retried master,
PowerSpan II sends the fetched data that was for the first master to the second master. PowerSpan II fails to distinguish between bus masters after it retries the original PCI master, which is required in the Master-based decode mode. When the original master tries to complete the read transaction after PowerSpan II stops retrying the master (because it believes the transaction is complete), the requested transaction is treated as a new transaction by PowerSpan II.

This errata only impacts the PCI Master-based decode feature. The PowerSpan II Processor Bus Master-based decode functions as intended.

Software Work-around

This errata only affects systems that use overlapping memory images in multiple PCI master systems. In order to avoid this errata, do not use overlapping address images in a PowerSpan II application that has multiple PCI Masters.

When a multiple PCI master PowerSpan II system has address images that do not use overlapping memory space, there is no issue associated with enabling the Master-based decode feature; it will not adversely affect the system.

2.1.14 PCI AC Timing Specification Violations: P1_RST_ to P1_REQ64_ Minimum Hold Time

Each PowerSpan II PCI interface can be configured for operating frequencies of 66MHz by ensuring that the pin Px_M66EN is connected to logic one. The Table 2 is a sub-set of the PCI timing table found in the *PowerSpan II User Manual*. Table 2 is valid for operation in a 3.3 V signaling environment.

Table 2: PCI 33 MHz and PCI 66 MHz Timing Parameters

Timing Parameter	Description	CE/IE		Units
		Min	Max	
t_{206}[a]	P1_RST_ to P1_REQ64_ hold time (Adapter Scenario) *(PCI Specification (Revision 2.2)* value is 0 ns)	2.3		ns

a. In the adapter scenario an external agent controls both P1_REQ64_ and P1_RST_

2.1.15 Incorrect Response to PCI-1 Fast Back-to-back Cycles Composed with a Different Transfer Size

PowerSpan II may not correctly drive the address onto the 60x bus when responding to a PCI-1 Interface fast back-to-back transfer.

Work-around

Ensure the PCI master (or PCI-to-PCI bridge) never generates fast back-to-back cycles to the PowerSpan II. If the software driver enables fast back-to-back cycles, contact the vendor for the driver patch.

3. Design Notes

Design notes are suggestions or additional system information that help design PowerSpan II into a system. Table Figure 3 provides a summary of the PowerSpan II device design notes.

Table 3: Summary of Design Notes

Design Note	Description
1	"Power Sequencing" on page 16
2	"Clock and Reset Timing" on page 17
3	"PowerSpan II and MPC107 Incompatibility" on page 19
4	"PowerSpan II and External Arbitration on the Processor (60x) Bus" on page 20
5	"PowerSpan II I_2O Shell Functionality is Not Recommended for Customer Use" on page 20

3.1 Power Sequencing

The following conditions are recommended to ensure correct operation of the PowerSpan II:

- The input voltage (Vin) must not exceed the I/O voltage (VDD2) by more than 2.5V at any time, including during power-on reset.
- The I/O voltage (VDD2) must not exceed the core voltage (VDD and Px_VDDA) by more than 1.6 V at any time, including during power-on reset.
- The core voltage (VDD and Px_VDDA) must not exceed the I/O voltage (VDD2) by more than 0.4 V at any time, including during power-on reset.

3.2 Clock and Reset Timing

Systems must have the P1_CLK, P2_CLK, and PB_CLK clock frequencies stable before PO_RST_ is deasserted. If a frequency change is required, a new power-up sequence (with the assertion of PO_RST_) must be initiated in order for the PLLs to lock. As long as applications comply to the clock and reset timing restrictions, there is no clock timing issue.

Real Life Customer Application Example

A customer used PowerSpan II in a PCI adapter card application. In this particular application, PO_RST_ was generated by a local voltage monitor device, while P1_CLK was provided by the system controller board through the backplane. Under certain power-up events, P1_CLK only became stable after the local PO_RST_ was deasserted because of different power ramping between the adapter card (containing PowerSpan II) and the system controller card. As a result of the application violating the clock and reset timing restrictions, the PowerSpan II PCI interface does not claim any transactions after reset.

The clock and reset timing restrictions for PowerSpan II are shown in the “AC Timing Chapter” and “Reset, and Clock Timing Parameters” table of the *PowerSpan II User Manual*. The timing parameter that deals with this issue is t102 in the table.

Customer Application Work-around

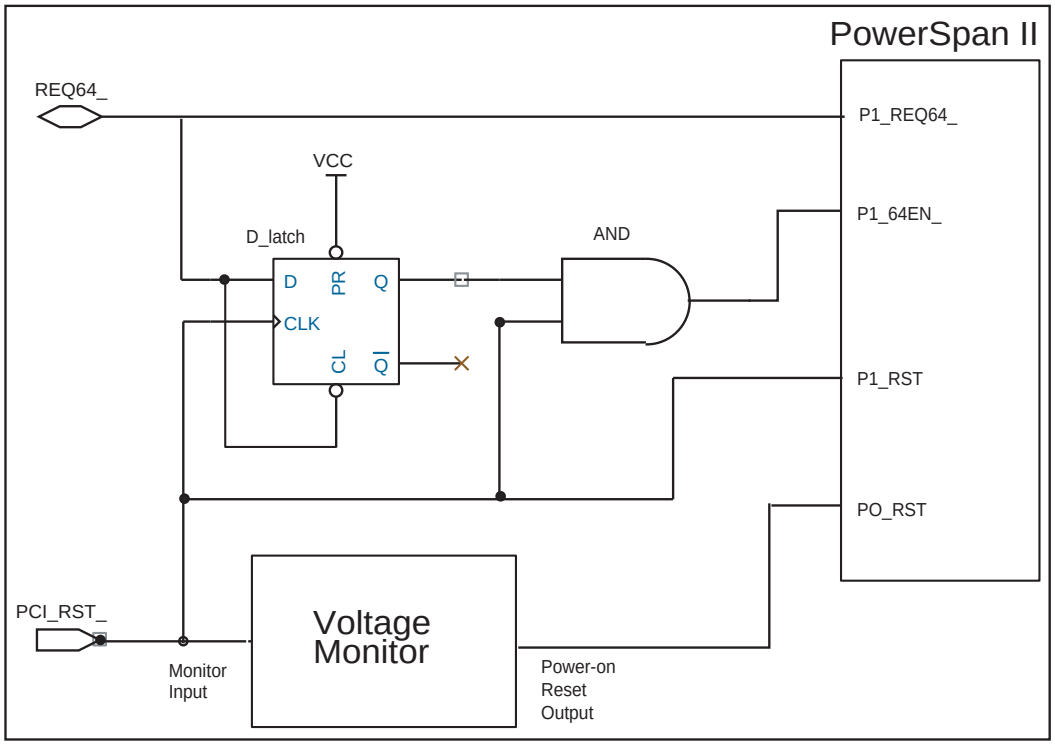
The proposed work-around to this situation is to delay deasserting PO_RST_ to PowerSpan II until the host deasserts the PCI_RST_ signal. Since the PCI_CLK is an output from the system controller along with PCI_RST_, the negation of PCI_RST_ implies the PCI_CLK is stable. The local voltage monitor uses PCI_RST_ to make sure that the PCI_CLK is stable before the deassertion of PO_RST_. A possible implementation of this circuit is shown in Figure 1.

By generating PO_RST_ off of PCI_RST_, there is a chance that the PowerSpan II will be unable to latch the state of the REQ64_ signal on deassertion of PCI_RST_. Therefore, an external D-Latch is used to capture the state of REQ64_ to determine the width of the bus (see Figure 1). The purpose of the asynchronous clear on D-latch is to prevent potential bus contention on the REQ64_ signal which is driven by PowerSpan II on the deassertion of P1_RST_ in a 32-bit system. The P1_64EN_ signal, which is originally used for CompactPCI Hot Swap applications, is used here to tell the PowerSpan II if this is a 32 or 64-bit application. The AND gate is AND-ing the output of D-latch with PCI_RST_ to ensure the P1_64EN_ is low when the D-latch power-up with uncertain output.

Each PCI_RST_ assertion leads to a corresponding assertion on PO_RST_. Therefore, the PowerSpan II Multiplex System Pin Mode Power-up option’s timing must be met with each toggle of the PCI_RST_ signal.

Figure 1 shows the logic required to delay deasserting PO_RST_ to PowerSpan II until the host deasserts the PCI_RST_ signal.

Figure 1: Logic Required to Delay Deasserting PO_RST_ to PowerSpan II



If the external glue logic shown in Figure 1 is implemented by a CPLD, make sure the CPLD output and Flip-flop state are not affected by any reset.

This work-around only applies to designs where PowerSpan II is in a PCI adapter card and is configured for a non-Hot Swap environment. Normally in this scenario, the PCI bus width is indicated by P1_REQ64_ signal (P1_64EN_ is pulled high). For designs where PowerSpan II is in a PCI host card or it is configured for a Hot Swap environment, the circuit shown in Figure 1 must not be used.

3.3 PowerSpan II and MPC107 Incompatibility

The Motorola MPC107 uses the Data Bus Grant (DBG) signal to pace the processor (60x) bus traffic. Whenever the MPC107 needs to hold on to its bus tenure in order to execute tasks on its other interfaces, its arbiter denies the DBG signal to other processor (60x) bus masters. This creates an issue for the MPC8260 and
PowerSpan II devices that do not use the DBG signal to determine whether they can take the ownership of the processor (60x) data bus.

The MPC8260 processor (60x) bus master ignores the DBG signal. If it does not get DBG, it simply continues with the data phase execution. The PowerSpan II processor (60x) bus master uses other signals to determine whether the data bus is free, but if the DBG signal is not asserted at the moment the PowerSpan II's processor (60x) bus master expects the data bus to be available, the PowerSpan II's processor (60x) bus master enters an error state and causes a live-lock on the processor (60x) bus.

The PowerSpan II and MPC107 should not be used together in a system.

3.4 PowerSpan II and External Arbitration on the Processor (60x) Bus

In systems where the PowerSpan II processor (60x) bus arbiter is disabled, PowerSpan II must sample the DBG_ signal (from the external arbiter) on the next clock cycle after the TS_ signal is asserted.

If the external arbiter used in the system is the MPC8260 processor (60x) bus arbiter, the PPC_ACR[DBGD] bit in MPC8260 registers must be set. This bit determines when the MPC8260 asserts DBG_ as system arbiter. Please refer to *MPC8260 User's Manual* for more information on the PPC_ACR[DBGD] bit.

If the PPC_ACR[DBGD] bit is not set when the MPC8260 arbiter is the external master in the system, the PowerSpan II processor (60x) bus master interface is not able to consistently master the processor (60x) bus. PowerSpan II continuously retries the external PCI access to the processor (60x) bus, but does not continue to assert Bus Request (BR_) signal. Because of the different system timing for the DBG_ signal, this issue can arise at various time. For example this issue can happen either the first time PowerSpan II accesses the processor (60x) bus, or several hours after the first access.

There is no system issue when using PowerSpan II's processor (60x) bus arbiter.

3.5 PowerSpan II I_2O Shell Functionality is Not Recommended for Customer Use

The I_2O shell functionality in PowerSpan II has not been extensively verified at the system level under various configurations. IDT does not recommend the use of the I_2O shell in PowerSpan II.